



Proyecto:	Simulador discreto para uso académico (SimuReD)
Autor:	Sergio Viera
Fecha:	24/04/2008
Cátedra:	Proyecto
Año:	2007
Profesor:	Ing. Walter Rodriguez Esquivel (MBA)
Ayudantes:	Ing. José Luís Gobbe Ing. Silvio Zurzolo

El presente trabajo es una obra de Sergio Viera, con el cual accede al título de Ingeniero en Sistemas de Información, graduado de la Facultad Regional Delta perteneciente a la Universidad Tecnológica Nacional (Argentina).

SimuReD, todos los derechos reservados (2009).

Se autoriza utilizar cualquier parte de este proyecto, siempre y cuando se cite la fuente.
Para solicitar información o consultar el avance del proyecto, visite www.simured.edu.ar

Indice

CAPITULO 1 – CONCEPTUALIZACIONES BÁSICAS	1
INTRODUCCIÓN	3
DIAGNÓSTICO DE SITUACIÓN	3
FUNDAMENTACIÓN	4
OBJETIVO GENERAL	5
OBJETIVOS ESPECÍFICOS	5
ALCANCE	5
RECURSOS NECESARIOS	5
RESULTADOS ESPERADOS	6
DETERMINACIÓN DE PLAZOS	7
CRONOGRAMA DE EJECUCIÓN	7
MARCO INSTITUCIONAL Y GERENCIA DEL PROYECTO	11
SUSTENTABILIDAD Y REPLICABILIDAD	11
MONITOREO Y EVALUACIÓN	12
CAPITULO 2 – ESTUDIO DE MERCADO	13
INTRODUCCIÓN	15
ANTECEDENTES	16
PRESENTACIÓN DEL PRODUCTO	19
MERCADO INTERNACIONAL	20
MERCADO NACIONAL	22
DESCRIPCIÓN DE LA OFERTA Y DEMANDA	25
COMPARACIÓN DE PRODUCTOS	36
CONCLUSIONES	37
CAPITULO 3 – TECNOLOGÍAS	39
INTRODUCCIÓN	40
METODOLOGÍAS	40
ARQUITECTURAS	46
HERRAMIENTAS DE DESARROLLO	49
GRÁFICOS Y ANIMACIÓN	53
WEB SERVICES	55
REQUERIMIENTOS OPERATIVOS	56
CONCLUSIONES	57
BIBLIOGRAFÍA	57
CAPITULO 4 – LOCALIZACIÓN	59
INTRODUCCIÓN	61
POSIBLES UBICACIONES	61
MACROLOCALIZACIÓN	62
MICROLOCALIZACIÓN	64
CONCLUSIÓN	65
CAPITULO 5 – INVERSIÓN	67
INTRODUCCIÓN	69
INVERSIONES PREVIAS A LA PUESTA EN MARCHA	70
INVERSIONES DURANTE EL DESARROLLO	70
CONCLUSIÓN	71

CAPITULO 6 – COSTOS	73
INTRODUCCIÓN	75
ESTIMACIÓN DEL ÁMBITO DEL SOFTWARE	76
ESTIMACIÓN DE LOS RECURSOS	78
ESTIMACIÓN DE LOS PUNTOS DE FUNCIÓN	80
APLICACIÓN DEL MODELO COCOMO II	88
APLICACIÓN DEL SOFTWARE CALIBRADO POR EXPERIENCIA PROFESIONAL	91
CONCLUSIONES	96
CAPITULO 7 – FINANCIAMIENTO	97
INTRODUCCIÓN	99
ENTIDADES BANCARIAS	99
OTRAS OPCIONES DE FINANCIAMIENTO	102
CONCLUSIÓN	105
CAPITULO 8 - RENTABILIDAD	107
INTRODUCCIÓN	109
ANÁLISIS COSTO-BENEFICIO	109
CÁLCULO DE INDICADORES	112
CONCLUSIÓN	114
CAPITULO 9 – LEGALES	117
INTRODUCCIÓN	119
REGISTRO DE LA MARCA	119
REGISTRO DE DOMINIOS	120
LA INDUSTRIA DEL SOFTWARE	121
REGLAMENTO DE BECAS DE INVESTIGACIÓN Y SERVICIO	122
PROPIEDAD INTELECTUAL	122
LICENCIAS DE SOFTWARE	123
CONCLUSIÓN	124
CAPITULO 10 – MARCO TEÓRICO TÉCNICO	125
INTRODUCCIÓN	127
MAPA CONCEPTUAL DE LA FUNDAMENTACIÓN TEÓRICA DE LA CARRERA	127
DIAGRAMA ESQUEMÁTICO DEL MARCO TEÓRICO TÉCNICO DEL PROYECTO	131
ANEXOS	133
ANEXO 2A – CARRERAS UTN	135
ANEXO 2B – DIFERENCIA ENTRE SIMULACIÓN Y OPTIMIZACIÓN	137
ANEXO 2C – DESCRIPCIÓN DE PRODUCTOS.	141
ANEXO 3A – SISTEMA OPERATIVO LINUX, DISTRIBUCIÓN CENTOS	159
ANEXO 3B – SERVIDOR WEB, APACHE TOMCAT	161
ANEXO 3C – VERSIONADO, CVS	163
ANEXO 3D – MYSQL	167
ANEXO 3E – FLASH & ACTIONSCRIPT	171
ANEXO 3F - WEB SERVICES – PROTOCOLO SOAP	173
ANEXO 4A – CAMPANA	177
ANEXO 4B – SANTA FE	185
ANEXO 4C – PLANO DE LA FACULTAD REGIONAL DELTA	189
ANEXO 4D – CAMPANA CIUDAD INTELIGENTE	191
ANEXO 6A – DESARROLLO DE LA TÉCNICA DE PUNTOS DE FUNCIÓN	193
ANEXO 6B – MÉTODO DE ESTIMACIÓN COCOMO	209

ANEXO 6C – DETALLES DEL SISTEMA DE BECAS DE LA UTN	217
ANEXO 7A - TIPOS DE PROYECTOS QUE FINANCIAN EL FONAR	219
ANEXO 9A – LEY NACIONAL DE PROMOCIÓN DE LA INDUSTRIA DEL SOFTWARE	223
ANEXO 9B – EL SOFTWARE COMO ACTIVIDAD INDUSTRIAL	229
ANEXO 9C – REGLAMENTO DE BECAS	231
ANEXO 9D – LEY DE MARCAS	249
ANEXO 9E – REGISTRO DE DOMINIOS EDU.AR	251

Capítulo 1 – Conceptualizaciones Básicas

INTRODUCCIÓN	3
DIAGNÓSTICO DE SITUACIÓN	3
FUNDAMENTACIÓN	4
OBJETIVO GENERAL	5
OBJETIVOS ESPECÍFICOS	5
ALCANCE	5
RECURSOS NECESARIOS	5
RESULTADOS ESPERADOS	6
DETERMINACIÓN DE PLAZOS	7
CRONOGRAMA DE EJECUCIÓN	7
MARCO INSTITUCIONAL Y GERENCIA DEL PROYECTO	11
SUSTENTABILIDAD Y REPLICABILIDAD	11
MONITOREO Y EVALUACIÓN	12

Introducción

El desarrollo constante de nuevos y evolucionados sistemas de simulación pone a disposición de profesionales, empresarios y científicos excelentes herramientas de análisis y toma de decisiones.

La complejidad de los sistemas del mundo moderno es una característica distintiva de la realidad actual. La simulación es la herramienta básica para modelar y analizar la complejidad de los sistemas reales.

Los métodos analíticos a menudo no brindan una solución directa a los complejos modelos que se obtienen de representar la realidad. Es por ello que las técnicas de simulación son alternativas y complementarias a estos modelos, ya que se basan en aproximaciones.

La simulación consiste en modelar el comportamiento de fenómenos físicos mediante la discretización del estudio de su comportamiento. Se llega a una solución mediante una secuencia de eventos en una escala temporal, partiendo de un estado conocido a otro estado final. Éste estado final se obtiene a través de eventos consecutivos aproximando valores mediante la aplicación de valores matemáticos, utilizando la probabilidad de eventos y la estadística, o la aplicación de valores tomados de la experimentación previa.

Seleccionar el modo aleatorio o determinístico dependerá de las necesidades de cada caso particular. Las variables que influyen en un modelo físico son de orígenes variados. Propiedades internas del proceso o externas del medio, sumadas a la interacción entre ambos son las características claves del sistema.

Los modelos discretos pueden representar los sistemas a partir de las características esenciales y las interrelaciones realimentadas. Esta visión simplifica el modelado y elimina el acoplamiento presente en los fenómenos naturales.

Diagnóstico de situación

El crecimiento en la calidad de las computadoras y del software asociado, ha incrementado la capacidad para estudiar los fenómenos de la naturaleza y de los sistemas concebidos por el hombre, omitiendo la necesidad de sobre-simplificar los problemas para hacerlos accesibles a su análisis.

La incorporación de estas herramientas y metodologías en los procesos de diseño, modificación, mantenimiento y aseguramiento de la confiabilidad de los componentes y estructuras se refleja en los nuevos enfoques incluidos en los códigos y normas de diseño, publicaciones científicas y técnicas, presentaciones en congresos de las distintas especialidades, especificaciones técnicas de compra de equipos, entre otros.

El beneficio inmediato de esta tendencia es la capacidad de producir diseños más eficientes y confiables, en un contexto menos restringido por la necesidad de concebir para poder calcular o de someterse a la necesidad ensayos que por razones de costo, en general no permiten agotar el proceso de optimización del diseño.

La posibilidad de representar las complejidades del mundo en el entorno virtual de una computadora eliminó la necesidad de forzar descripciones lineales (de un mundo esencialmente no-lineal) por limitaciones de los medios de cálculo, poniendo los límites del lado del profesional antes que en las herramientas y metodologías disponibles.

Concretamente esto significa que los profesionales que pretendan ser competentes en este nuevo contexto deben estar formados para utilizar de un modo solvente las herramientas disponibles. Resulta ocioso señalar que cualquier pretensión de desarrollo tecnológico independiente implica la capacidad para desarrollar las propias herramientas.

Si se analiza este fenómeno desde la perspectiva de las universidades, se puede concluir que, debido a fundadas razones de eficiencia y generalidad, los diseños curriculares de las distintas carreras de la ingeniería no incluyen ni pueden incluir la variedad de temas necesarios para garantizar una formación adecuada en estas disciplinas, requiriendo de acciones concretas que tiendan a completar una formación profesional que garantice el buen desempeño, con competencia y perspectivas de desarrollo.

En tal sentido y observando las tendencias de otros países que han partido de un contexto socio-económico y cultural semejante al nuestro y que hoy disponen de profesionales con un nivel de formación apropiado para sostener sus planes de desarrollo industrial, se concluye que es imprescindible ofrecer formación de posgrado que provean de los conocimientos, aptitudes y competencias que permitan expandir nuestro campo científico y cultural, producir bienes científicos y tecnológicos y fortalecer el desarrollo económico y social.

Fundamentación

En el contexto antes descrito, las organizaciones requieren de profesionales altamente capacitados y actualizados en el área de la simulación computacional, área del conocimiento en permanente transformación y evolución tecnológica de tal magnitud, que la desactualización de los conocimientos en ciertos temas se pueden medir en breves lapsos de tiempo.

La Universidad tiene la responsabilidad de crear un contexto orgánico en el que los recursos formativos actualmente existentes en ésta universidad y en el conjunto de las universidades públicas, actuando concurrente y coordinadamente, contribuyan en la formación de profesionales de alto nivel de competencia capaces de atender las necesidades crecientes de innovación e investigación tecnológica con base en la capacidad industrial instalada.

Objetivo General

Ofrecer una herramienta de software para el desarrollo de modelos de simulación discreta adaptable a diversos tipos de escenarios.

Objetivos Específicos

- 1 - Promover el uso de simulaciones, entregando una herramienta de calidad para la formación académica en las especialidades que se requieran.
- 2 - Ofrecer un software gratuito para uso académico y sin restricciones en cuanto a su funcionalidad.
- 3 - Confeccionar un software sencillo y compacto, que permita la facilidad de uso y el mínimo de requisitos para su ejecución.
- 4 - Evitar conflictos referentes a la plataforma utilizada por el cliente, permitiendo su ejecución sobre cualquier Sistema Operativo y configuración de hardware.
- 5 - Permitir la ejecución de los modelos de simulación utilizando el formato Web2.0.
- 6 - El software de simulación debe utilizar la teoría de colas para el procesamiento y modelado.

Alcance

- El sistema está orientado al uso exclusivo de la UTN (Argentina), extendiéndose a todas sus Facultades y Unidades Académicas, para todas las materias que sea requerido.
- Quedan excluidos todos los cursos extracurriculares que se dicten en las distintas filiales de la UTN y otras entidades educativas.
- En una segunda etapa se prevé la extensión del software para uso comercial, destinado tanto a Pymes como a Universidades privadas.
- Se excluye la etapa de capacitación en simulación, dado que el software está orientado al uso académico y por lo tanto excede el alcance de este proyecto.

Recursos necesarios

El desarrollo de SimuReD será realizado a través del sistema de Becas de Investigación y Servicio, provenientes del rectorado de la Universidad Tecnológica Nacional. Y se utilizarán las instalaciones en horarios en los que usualmente se encuentran desocupadas. Por tal motivo, los recursos financieros que demandará el proyecto se evalúan en base a un costo de oportunidad de utilización de tales recursos.

El desarrollo de SimuReD implica la utilización de recursos materiales y económicos disponibles en la facultad, los cuales deben ser asignados al presente proyecto.

Los recursos materiales necesarios son todos los elementos de oficina, los cuales están disponibles en el aula 19 o en el laboratorio de sistemas. En cuanto a los recursos del servidor, los mismos están disponibles en Gesin y listos para usarse en proyectos de este tipo.

En cuanto a los recursos económicos, se deben asignar 3 módulos de las becas de investigación y servicio a este proyecto, los cuales se encuentran disponibles.

El personal necesario para el desarrollo de SimuRed se reclutará a partir del alumnado que concurre a la Facultad Regional Delta. Además se contará con la dirección del autor del presente proyecto, supervisado por el Director de la carrera de Ingeniería en Sistemas Ing. Luis Perna y con la asistencia técnica necesaria por parte del plantel docente que presta sus servicios en dicha institución.

Los roles a desempeñar por el equipo interno serán:

- 1 Project Leader, cuyas tareas serán: la administración de los recursos, control del avance del proyecto y configuración de la infraestructura necesaria para el desarrollo.
- 1 Analista funcional, se encargará de la documentación asociada al negocio de la aplicación.
- 1 Analista programador gráfico, con la tarea de realizar todo el entorno gráfico de la aplicación.
- 1 Analista programador Java, quien se encargará de las tareas de desarrollo del negocio.

Resultados esperados

Proporcionar un simulador online, con interfaz gráfica, sin restricciones de uso para el ambiente académico Nacional.

Se espera que el uso de SimuReD sea superior al 50% de todas las cátedras de simulación de la UTN.

Un resultado paralelo es que se promocióne la capacidad que posee la FRD de desarrollar proyectos de gran magnitud a nivel nacional e internacional.

El resultado principal espedado de este proyecto es obtener un simulador con características académicas, pero con un potencial mercado, ya que las técnicas de simulación están bajo constante evolución y todos los indicadores muestran que los simuladores se convertirán en una de las principales fuentes de información para la toma de desiciones.

El entrenamiento que perciban los alumnos que participen del proyecto les dará una excelente experiencia profesional lo cual les permitirá insertarse en el creciente mercado laboral en materia de software.

Este entrenamiento beneficiará a la Facultad Regional Delta, ya que la excesiva demanda que hay en la zona Zárate-Campana de programadores experimentados en nuevas tecnologías, no puede ser cubierta por esta casa de altos estudios. De esta manera, se está contribuyendo a aumentar la calidad de la mano de obra en la zona.

El desarrollo de un simulador para uso académico, con las características de SimuReD, es un evento sin precedentes, por lo cual la concreción del presente proyecto deberá traer

consigo una divulgación por medios de prensa local y nacional, ya que el mismo será utilizado en todas las facultades de al UTN. La divulgación de este proyecto llamará la atención de los empresarios a los cuales se deberá invitar a acercarse a la Facultad Regional Delta para asesorarlos sobre el uso de las técnicas de simulación.

Obtener una herramienta como SimuReD, permitirá mejorar el desarrollo de habilidades en aquellos alumnos que se interesen por la simulación. Esto hoy no es posible dadas las limitaciones impuestas en los softwares utilizados bajo licencias académicas.

Determinación de plazos

Con los recursos humanos anteriormente descriptos y considerando la baja carga horaria mensual que tienen las becas de investigación y servicio, la concreción del presente proyecto tiene una duración de dos años académicos. Esto es, iniciando actividades en Abril y terminando las mismas en Noviembre.

Cronograma de ejecución

El cronograma tentativo para la ejecución del proyecto es el estipulado en el diagrama de Gannt siguiente.

Mes	Abr-09	May-09	Jun-09	Jul-09	Ago-09	Sep-09	Oct-09	Nov-09	Dic-09	Ene-10	Feb-10	Mar-10	Abr-10	May-10	Jun-10	Jul-10	Ago-10	Sep-10	Oct-10	Nov-10	Dic-10
	1	2	3	4	5	6	7	8	9				1	1	1	1	1	1	1	1	1
Reclutamiento.	█												█								
Organización Física.	█	█	█										█								
Formación del equipo.		█	█	█										█	█	█					
Análisis de sistema		█	█	█	█	█	█	█	█					█	█	█	█	█			
Desarrollo					█	█	█	█	█					█	█	█	█	█	█	█	█
Pruebas y QA							█	█	█						█	█	█	█	█	█	█
Distribución									█												█

Tabla 1 – Cronograma Tentativo de Ejecución

Las actividades que se desarrollarán en cada una de estas etapas se describen en la tabla 2. En dicha tabla se puede ver la estimación de los tiempos para cada subtarea evaluada a través de la estimación de los tiempos optimistas, pesimistas y normales, y a partir de ellos calcular el tiempo medio de ejecución de la tarea.

Reclutamiento de los integrantes del equipo.

ACTIVIDADES	DETALLE	Tiempos estimados			
		to	tn	tp	tm
Asignación de los roles.	En las entrevistas con los postulantes se evalúa el rol a ocupar.	1	1	3	2
Preparación de las entrevistas.	Realizar planillas, test, métodos de evaluación, metodos de selección. Definir el lugar dónde se realizarán las entrevistas	2	4	6	4
Publicación de solicitada para reclutar el personal.	En el 2009 se realizó una feria para dar a conocer a los alumnos los proyectos activos en la facultad y donde se presentó SimuReD.	1	1	3	2
Realización de entrevistas a los postulantes.		1	3	10	5
Analisis y selección de los integrantes del proyecto.	Mediante el método de selección elegido y realizando las comparaciones necesarias de acuerdo a los postulantes. Seleccionar titular del puesto y suplentes.	1	3	4	3
Confirmar con los postulantes la integración al proyecto.	Realizar una segunda entrevista al momento de la confirmación para describir el rol a desarrollar.	1	1	5	2

Organización Física del lugar de trabajo y de los materiales.

ACTIVIDADES	DETALLE	Tiempos estimados			
		to	tn	tp	tm
Acondicionamiento del lugar de desarrollo del proyecto.	Se deberá comprobar el estado de la iluminación, ventilación, instalación eléctrica, pintura, pisos, puertas y ventanas, ruidos externos, olores externos, control del mobiliario, carteles indicadores de salida, matafuegos.				
	Exponer cuadros con mensajes alentadores y figuras agradables. Incorporar plantas al ambiente de trabajo.	1	3	8	4
Adquisición del hardware necesario.	Comprobar las PCs con los requerimientos mínimos.				
	Comprobar el funcionamiento de la red, servidores y el acceso a Internet.				
	Verificar el acceso a impresoras.	1	2	5	3
Adquisición de los recursos de software necesarios	Comprobar si las PCs disponibles contienen el software necesario, en caso de no tenerlo instalado proceder a la instalación de los mismos previa descarga de los instaladores correspondientes (nota: se usará software bajo licencia GNU).	0	2	3	2

Seguridad física de la oficina.	Establecer políticas de seguridad, acceso, verificar trabas en puertas y ventanas. Establecer quién accede y cómo se resguardarán los bienes dentro de la oficina.	1	3	5	3
Integración del equipo de trabajo.	Se destinará el tiempo necesario a la integración y socialización de todas las personas involucradas. Se dará énfasis en la presentación profesional de cada integrante. Esta actividad es deseable realizarla con un refrigerio (desayuno, merienda, café, etc).	1	1	2	1
Análisis del objetivo del proyecto.	Lectura del objetivo general del proyecto y análisis grupal.	1	1	1	1

Conformación del equipo y siembra de la semilla para definir la cultura organizacional.

ACTIVIDADES	DETALLE	Tiempos estimados			
		to	tn	tp	tm
Definición de roles dentro del proyecto.	En esta actividad se dará una descripción de los roles y una justificación del porque la persona seleccionada ocupará ese rol. También se dará la oportunidad de que el implicado realice un feedback sobre las tareas que espera realizar en el proyecto. Los roles ya han sido debidamente notificados a cada integrante en forma individual al momento de la contratación (preferentemente en forma escrita).	1	1	1	1
Definición de la misión del grupo.	Se redactará en forma conjunta la misión del proyecto, poniendo especial atención a que todos los integrantes participen activamente de esta actividad.	1	1	3	2
Capacitación / Repaso de conceptos.	Teoría de colas.				
	Sistemas de simulación discretos.	2	5	10	6
Definición de políticas de QA	Definir las métricas y políticas de calidad para adoptar buenas prácticas de trabajo.	5	10	15	10

Análisis del Sistema.

ACTIVIDADES	DETALLE	Tiempos estimados			
		to	tn	tp	tm
Análisis tecnológico	Se realizará un análisis de las herramientas disponibles en las distintas facultades para adaptar el desarrollo a los estándares disponibles.	1	5	8	5

Análisis de requerimientos	Se realizará un análisis minucioso, mediante entrevistas y encuestas a los profesores de las materias asociadas que desean utilizar esta herramienta para determinar los requerimientos específicos.	12	15	20	16
Arquitectura de desarrollo	Seleccionar un framework o base para que se ajuste a la arquitectura propuesta.	4	5	9	6
Análisis técnico	Realizar un análisis técnico con las metodologías a adoptar para la implementación del motor de simulación.	14	15	16	15
Diseño Funcional	Redacción y confección de los CU para el desarrollo.	7	20	25	17
Diseño IT	Implementación de la arquitectura seleccionada. Confeccionar interrelación entre el SO y decidir la estructura del directorio asociado al desarrollo.	10	25	30	22

Diseño del sistema.

ACTIVIDADES	DETALLE	Tiempos estimados			
		to	tn	tp	tm
Diseño técnico	Seleccionar o realizar el core que dará soporte al desarrollo, producto del análisis funcional. Esto se realizará tanto en la plataforma seleccionada como en la base de datos.	5	20	25	17
Implementación de procedimientos de BD	Configuración y seteos iniciales de la base de datos. Incluye las definiciones de los procedimientos que interactuarán con el sistema.	8	10	15	11
Codificación y documentación	Implementación del código en el lenguaje seleccionado. Documentación interna del código.	5	15	20	13

Desarrollo

ACTIVIDADES	DETALLE	Tiempos estimados			
		to	tn	tp	tm
Manuales de usuario y documentación	Escritura y redacción de los manuales de usuario y de la documentación técnica externa.	5	10	20	12
Pruebas de usuario	Pruebas de funcionalidad técnica.	15	30	35	27
Pruebas de QA	Verificar estilos de diseño definidos en pantallas y reportes.	3	10	12	8

Pruebas y QA

ACTIVIDADES	DETALLE	Tiempos estimados			
		to	tn	tp	tm
Refactoring	Corrección de errores detectados en esta etapa.	0	5	10	5
Publicación y difusión del sistema	Publicar el software y realizar las notificaciones necesarias	1	5	15	7
Pruebas de usuario	Se realizan las pruebas de usuario, pero en esta ocasión en modo operativo	0	0	0	0

Distribución

ACTIVIDADES	DETALLE	Tiempos estimados			
		to	tn	tp	tm
Actualización de la página web	La misma será conformada por el equipo participante dando un marco adicional para que cada uno cuente su experiencia en el grupo.	5	6	7	6

Marco Institucional y Gerencia del Proyecto

Este proyecto es desarrollado como trabajo final de la carrera de ingeniería en sistemas de información del alumno Sergio Viera y con la presentación formal del mismo su servidor obtendrá el título de Ingeniero.

La cátedra Proyecto, dirigida por el Ing. Walter Rodríguez Esquivel (MBA) y con la participación de los Ingenieros Jose Luis Gobbe y Silvio Zurzolo, es el principal motor que dio las herramientas y el impulso necesario para la formulación del presente proyecto.

También es destacable la aprobación por parte de la secretaría académica de sistemas y el apoyo incondicional del laboratorio de sistemas, Gesin y la secretaría de asuntos estudiantiles.

La gerencia del proyecto estará a cargo del flamante Ingeniero, autor del proyecto.

Sustentabilidad y replicabilidad

SimuReD es sustentable, dado que se desarrollará toda la documentación necesaria bajo estándares actuales del mercado para su mantenimiento y extensión de requerimientos acorde a las necesidades posteriores a la finalización del presente proyecto.

Dadas las características de SimuReD no es necesaria la replicabilidad del proyecto, ya que se puede extender su uso a los usuarios que sea necesario. De esta manera se podría prestar servicio desde el mismo servidor a potenciales clientes.

De ser necesario, se puede instalar un servidor en una empresa, con el software y las características técnicas descritas en el Capítulo 3 – Tecnologías y montar los servlets necesarios en dicho servidor para uso dentro de una intranet.

Monitoreo y Evaluación

El monitoreo del personal a lo largo del desarrollo del proyecto estará a cargo de la SAE (Secretaría de Asuntos Estudiantiles), que es quien realiza el control de asistencia y feedback de los becarios que prestan el servicio.

La evaluación del desarrollo estará medido por el profesor y ayudante de la cátedra de Simulación de la Regional Delta y de los alumnos que cursan la materia. También se tiene el apoyo de la Regional Santa Fe, cuyos docentes accedieron a realizar el testing de la aplicación y a reportar los bugs encontrados.

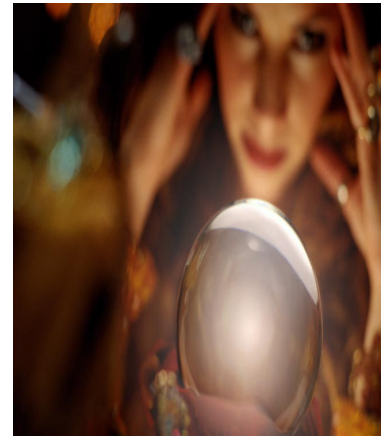
Capitulo 2 – Estudio de Mercado

INTRODUCCIÓN	15
ANTECEDENTES	16
PRESENTACIÓN DEL PRODUCTO	19
MERCADO INTERNACIONAL	20
MERCADO NACIONAL	22
DESCRIPCIÓN DE LA OFERTA Y DEMANDA	25
COMPARACIÓN DE PRODUCTOS	36
CONCLUSIONES	37

Introducción

Durante varias generaciones, y desde tiempos ancestrales, el hombre siempre quiso tener el poder de adivinar el futuro. Saber que cosas ocurrirán dentro de algún tiempo es un anhelo de toda la humanidad.

Imagine por un momento si pudiera saber con anticipación la hora exacta que debe salir de su casa para llegar a tiempo a su trabajo, previendo cualquier imprevisto del viaje; o por ejemplo, si pudiera programar sus vacaciones en el lugar con el clima apropiado para que no llueva ni un día; o conocer que cantidad exacta de productos venderá su empresa en el próximo trimestre.



La posibilidad de conocer eventos futuros trae aparejada una serie de ventajas para anticiparse a los hechos y actuar en consecuencia. Esto es importante para cualquier desarrollo social, ya que la información es el activo más importante que cualquier Organización pueda tener.

Con el desarrollo de la matemática estadística, la lógica y la Ingeniería en Sistemas, se tienen los elementos que dan origen a una herramienta muy eficiente para la predicción de escenarios futuros, la simulación.

Una definición formal de un sistema de simulación es la formulada por R.E. Shannon¹:

"La simulación es el proceso de diseñar un modelo de un sistema real y llevar a término experiencias con él, con la finalidad de comprender el comportamiento del sistema o evaluar nuevas estrategias -dentro de los límites impuestos por un cierto criterio o un conjunto de ellos- para el funcionamiento del sistema".

También se puede mencionar la formulada por Tomas H. Naylor²:

"Simulación es el proceso de diseñar y desarrollar un modelo computarizado de un sistema o proceso y conducir experimentos con este modelo con el propósito de entender el comportamiento del sistema o evaluar varias estrategias con las cuales se puede operar el sistema".

¹ "Simulación de Sistemas, Diseño Desarrollo e Implantación" (1999), Ed. Trillas.

² "Experimentos de Simulación en Computadoras con Modelos de Sistemas Económicos" (1977), ed. Limusa.

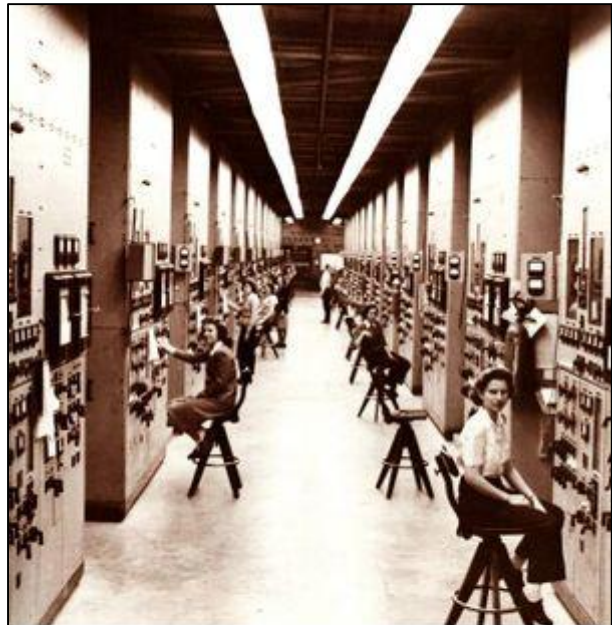
Antecedentes

Un poco de historia

El término simulación comenzó a utilizarse por primera vez durante los años 40 en el Proyecto Manhattan³ en la Segunda Guerra Mundial para modelar el proceso de la detonación nuclear.

John Von Neumann, era el encargado del diseño del dispositivo Trinity Test, junto a él se encontraba el científico polaco Stanislaw Ulam. Ambos introdujeron el término Simulación cuando trabajaban en el laboratorio Nacional Los Álamos California, Estados Unidos.

Éstos físicos se enfrentaron a problemas que eran demasiado caros para ser resueltos experimentalmente, o demasiado complejos para ser resueltos analíticamente. A raíz de esto, Von Neumann y Ulam reconocieron el potencial de las computadoras electrónicas para automatizar el proceso matemático que permitía verificar las soluciones propuestas. Ulam y Von Neumann, junto a Nicholas Metrópolis, desarrollaron algoritmos de implementación y encontraron formas de convertir problemas no aleatorios en formas aleatorias para ser solucionados vía muestreo estadístico.



Paneles de control y operadores del sistema de cálculo principal del Proyecto Manhattan situado en Oak Ridge, Tennessee. Durante el proyecto Manhattan los operadores eran mujeres que trabajaban en turnos de 24 horas sin conocer el propósito o consecuencias de su trabajo.

En su investigación acerca de la difusión de los neutrones, los científicos sugirieron una solución que equivalía a someter el problema a una ruleta. Paso a paso las probabilidades de los diversos acontecimientos se fueron unieron en una imagen que dio una solución aproximada del problema. Von Neumann dio el nombre clave de Monte Carlo⁴ a los trabajos secretos de los Álamos como una codificación por el proyecto secreto del departamento de defensa de los EE.UU. que ellos conducían.



El llamado *muestreo de Monte Carlo* es fundamental para el concepto de los sistemas de simulación que contienen elementos estocásticos o probabilísticos. El método fue tan

³ El Proyecto Manhattan era el nombre en clave de un proyecto de investigación científico llevado a cabo durante la Segunda Guerra Mundial por los Estados Unidos con ayuda parcial del Reino Unido y Canadá. El objetivo final del proyecto era el desarrollo de la primera bomba atómica. (Fuente: Wikipedia)

⁴ Haciendo referencia a la ciudad de Monte Carlo, Mónaco.

exitoso que su popularidad se extendió a varios campos y el término casi se ha vuelto sinónimo de simulación en la mente de muchas personas.

En la actualidad, gracias a la evolución del hardware y al desarrollo de software existen herramientas de propósito general y específicas para la simulación en diversos escenarios, con características interesantes, tales como la facilidad de uso, ambiente gráfico e interactivo, poderosas instrucciones para el manejo de fenómenos de líneas de espera, procesos de fabricación, trabajo en equipo y otras. Además dicho software ofrece la posibilidad de diseñar e implementar interfaces con otras aplicaciones, gracias al enfoque orientado a objetos en que se fundamentan. Algunos de estos productos son: SIMAN, ARENA, PROMODEL, MASTERCAM.

La ventaja que introdujo la simulación es su adaptabilidad a disciplinas variadas, como ser: administración, transporte, producción, logística y distribución, y un sin fin mas que puedan plantearse como sistemas globales de datos relacionados. Esto introdujo estudios específicos a muy bajo costo (prácticamente el del software asociado y las horas hombre disponibles) que llevados a la dinámica empresarial hicieron que su implementación sea atractiva frente a los beneficios que se pueden obtener. Pues el modelar situaciones conocidas para crear escenarios probables para sacar conclusiones en tiempo real evidencia una ventaja competitiva que una Empresa no puede despreciar. Algunas empresas, en particular muy pocas en la Argentina, implementan esta tecnología en los llamados Laboratorios de Aprendizaje e involucran a su personal en los cambios que esta introduce. En otras partes del mundo (especialmente en EE.UU.) el MIT⁵ coloca a este tipo de prácticas como normales dentro del esquema de transferencia tecnológica que posee la Universidad y la Empresa. Mientras que en nuestro país, recién en la actualidad comienza a aplicarse con mayor énfasis en algunas áreas.



La simulación está basada en datos estadísticos que recrean condiciones y soluciones probables. Esta herramienta, solo nos proporciona una ayuda para una interpretación sistémica de la solución a acontecimientos que están interrelacionados unos con otros. A su vez crean entrenamiento e instrucción en el personal afectado a su desarrollo e implementación. Todo proyecto a modelar, y particularmente los que llevan adelante profesionales líderes en simulación numérica, deben saber combinar parámetros, variables, relaciones funcionales, y restricciones que formen los componentes que desarrollan la función objetivo.



Estos modelos tienen un excelente y amplio campo de aplicabilidad en las empresas, lo cual permite recrear condiciones y establecer políticas de decisión, a muy bajo costo de implementación. Lo que aventaja en forma sustancial a la Empresa que lo desarrolla frente a

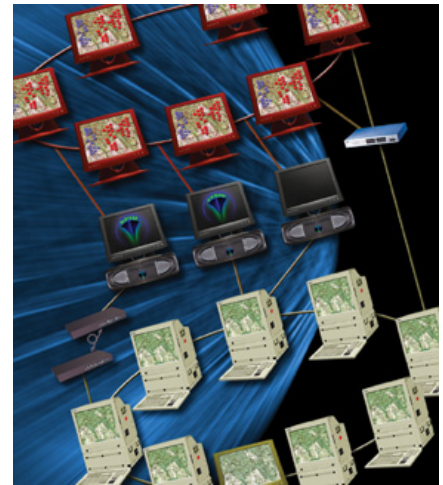
⁵ Instituto Tecnológico de Massachusetts.

sus competidoras. Pues incorpora no solo un ordenamiento en la toma de decisiones, sino también pautas y actitudes mas abiertas en las personas afectadas en las tareas específicas. Dejando de lado aspectos que hacen a la antigua y estática empresa del siglo pasado e incorporando pautas de la Empresa dinámica del siglo XXI que deja de lado los principios de la división del trabajo (que en su época dieron resultados y en la actualidad retrasan la producción), para aplicar aspectos empresariales que muestran al trabajo final como parte de un sistema global productivo y no como mero resultado de acciones individuales, interiorizando a cada individuo de los alcances y objetivos de la empresa dentro del entorno productivo mundial.

Existen varias herramientas para realizar simulaciones, las cuales pueden clasificarse de la siguiente manera:

- Según su alcance:

- Específicos: están desarrollados para un fin concreto, como puede ser la simulación del comportamiento de un fluido atravesando una tubería, donde pueden cambiarse las opciones (por ejemplo, diámetro de la tubería, densidad del fluido) pero no puede cambiarse el modelo.
- Genéricos: están desarrollados para múltiples usos, en este caso se puede cambiar el modelo. En algunos casos, los sistemas de simulación genéricos, ofrecen la arquitectura de sistemas predefinidos que pueden tomarse como inicio para un desarrollo a medida.



- Según su destino:

- Recreativos: en esta clasificación podemos mencionar aquellos simuladores que fueron diseñados con fines de ocio, es el caso de los simuladores de vuelo o de navegación que no tienen fines educativos ni de entrenamiento.
- Aplicativos: son los que se utilizan como herramientas de apoyo a decisiones y análisis. A través de ellos se realizan simulaciones para determinar el comportamiento de un escenario a diversos cambios.

- Según sus atributos:

- Discretos: cada una de las entidades que pertenecen al sistema se puede individualizar, y ésta, a su vez, funciona de manera independiente. En este caso podemos mencionar la simulación de una línea embotelladora, o la simulación de la distribución de paquetes a varias localidades con distintos vehículos.
- Continuos: en este caso las entidades no pueden ser individualizadas, o su comportamiento denota cierta correspondencia. Aquí se pueden mencionar simulaciones sobre sistemas de fluidos.

Tomando en cuenta la breve descripción anterior y enfocado, desde el punto de vista de tomar a los simuladores como herramientas, éstos ofrecen un gran potencial para el desarrollo y análisis de sistemas altamente complejos, que de otro modo serían imposibles de analizar.

Otro de los puntos fuertes de los sistemas de simulación, es que la escala de tiempo puede ser alterada según la necesidad de estudio. Esto quiere decir que un proceso muy veloz (supongamos explosivo) puede frenarse para que su ejecución tarde lo suficiente como para ver ciertos procesos, que de otra manera, permanecen ocultos, o se puede acelerar un proceso que dura miles de años para ser ejecutado en pocos segundos. También permite analizar un instante de tiempo crítico o avanzar y retroceder a través del tiempo.

Con estas premisas, se puede comenzar a vislumbrar la importancia del desarrollo de estas herramientas, con el potencial agregado que el costo de operación es reducido, comparado con el costo de la metodología “práctica y error” utilizada en reemplazo de la simulación.

Por otra parte, si nos referimos a la Optimización como herramienta para la resolución de problemas, veremos que ésta no satisface las necesidades de complejidad de los sistemas de hoy en día, y en este caso también debe considerarse la simulación como la mejor alternativa para la mayoría de los casos (Para conocer en detalle las diferencias puntuales entre estos métodos remítase al Anexo B).

Presentación del producto

Actualmente, las especialidades que utilizan sistemas de simulación van desde la Ingeniería y las ciencias básicas hasta las finanzas y los negocios.

Cabe destacar que la simulación es una técnica prácticamente nueva, y su desarrollo aún no llegó a su máximo nivel, por lo que se encontrarán importantes oportunidades para el desarrollo de patentes industriales (bajo el método científico que aporten a la ciencia).

En la actualidad existe un reducido número de opciones de software para el modelado de sistemas de simulación. Aquí nos enfocamos en software para simulaciones del tipo discreto y de uso genérico. La falta de este tipo de software, con licencias gratuitas para uso académico muestra una clara oportunidad en el mercado, ya que en su gran mayoría, los profesionales tienden a utilizar las mismas herramientas con la que fueron capacitados.

La Universidad Tecnológica Nacional forma profesionales que, luego de egresados, ocupan posiciones estratégicas en todo tipo de Organizaciones, tanto a nivel Nacional como Internacional. A raíz de esto se puede prever extender la utilización mediante la venta de licencias, el soporte técnico y la consultoría las cuales se convertirán en los principales servicios, siendo estos los puntos clave para formular nuevos proyectos futuros una vez concretado éste.

Otro punto a destacar es el cambio tecnológico de los últimos 3 años, que ubican a cualquier proyecto nuevo ante igualdad de condiciones frente a la competencia. Entiéndase por cambio tecnológico a los siguientes puntos:

- La orientación del software a la Web 2.0, esto es, permitir crear comunidades virtuales en torno a la aplicación.
- La alta variedad de equipos donde ejecutar programas, extensibles desde PCs, notebook, teléfonos celulares, BlueBerries y otros dispositivos que se incorporen en los próximos años, como TV Digitales.



Hoy en día gran parte del software de simulación se desarrolló bajo un concepto comercial, dejando de lado el ambiente académico. Gran parte de estas empresas ofrecen servicios de capacitación en la utilización del software, pero no ofrecen capacitación en la utilización de técnicas de modelado de simulación.

SimuRed nace a partir de una necesidad académica, libre y sin restricciones para tal fin dentro de la UTN, dando una característica distintiva dentro del mercado.

Mercado Internacional

El mercado internacional de los simuladores comerciales se encuentra en un estado de crecimiento sin precedentes. Este crecimiento es debido a la demanda generada por las distintas áreas, donde la utilización de los simuladores es altamente valorado por su eficiencia y eficacia. La globalización de los mercados incrementa la competitividad, por ello, una decisión rápida y correcta puede hacer la diferencia entre seguir en el mercado o quedar excluido del mismo. Para tomar esas decisiones, son muy valorados los simuladores como herramientas de análisis.

Ante esta imperiosa necesidad fueron varios los simuladores que se comenzaron a desarrollar y comercializar. El origen de estas nuevas soluciones puede clasificarse en 4 grandes grupos.

- Empresas dedicadas a la Ingeniería y/o Automatización.
- Empresas nuevas dedicadas a Simuladores.
- Empresas de cualquier rubro que necesitan de un Simulador.
- Instituciones públicas o privadas orientadas a la Investigación y Desarrollo.

Anualmente, todas las Empresas dedicadas a simulación, se congregan en la WSC (Winter Simulation Conference), donde exponen los adelantos logrados y presentan nuevos productos al mercado. Esta Convención se realiza en Miami, Florida y nuclea tanto a Empresas como a Instituciones educativas de todo el mundo. Si bien se realizan otras Conferencias Internacionales, la WSC es la más importante dado que la mayoría de las Empresas dedicadas al desarrollo de sistemas de simulación se encuentran en EE.UU.

En la última Convención realizada, Rockwell Automation⁶ presentó nuevos productos y adelantos para su producto Arena Simulation Software, el cual es líder en el mercado del software de simulación. Este liderazgo se mantiene desde hace varios años, pero hoy en día el crecimiento sostenido de Simuladores tales como Promodel y Flexim⁷ hacen que la diferencia de liderazgo sea cada vez menor.

Las empresas que comenzaron a desarrollar herramientas a medida, simuladores dedicados, en muchas ocasiones vieron la oportunidad de lanzar al mercado su producto y/o servicio. Un caso de éxito es el logrado por la empresa Boeing, dedicada a la construcción de aeronaves, la cual desarrolló un simulador que emula varios tipos de aeronaves y son utilizados actualmente para el entrenamiento y evaluación de pilotos de distintas aerolíneas del mundo.

Instituciones académicas de todo el mundo compiten activamente en el mercado, tal es el caso de los simuladores 20Sim, DEVS Java, Octave, Taylor y Tutsim, los cuales todos tienen como común denominador que fueron incubados en Universidades de distintos países.

Octave fue creado alrededor del año 1988, en la Universidad de Wisconsin, pero con una finalidad diferente a un simulador: ser utilizado en un curso de diseño de reactores químicos. Posteriormente en el año 1992, se decide extenderlo y comienza su desarrollo a cargo de John Eaton. La primera versión alpha fue lanzada el 4 de enero de 1993. Un año más tarde, el 17 de febrero, 1994 aparece la versión 1.0. El nombre surge del nombre de un profesor de uno de los autores conocido por sus buenas aproximaciones por medio de cálculos mentales a problemas numéricos. Octave es un firme competidor de Matlab, software muy completo para cálculos de Ingeniería y fue completamente desarrollado en una Universidad.

DEVS Java más que un simulador es un nuevo formalismo para el modelado y diseño de simulaciones. Este formalismo se basa en teorías bien definidas que permite expresar jerarquías de módulos, donde cada uno de esos módulos es un modelo de simulación discreta. La Universidad de Arizona es la sede de este proyecto, actualmente DEVS Java es Open Source, por lo que se está creando una comunidad que impulsa el desarrollo.

La Universidad de Twente (Países Bajos) desarrolló Tutsim en los años 70. Una vez lograda la madurez del proyecto fueron vendidos los derechos a una empresa privada para su comercialización. En la actualidad este producto es 20Sim, el cual presenta excelente rendimiento y calidad.

Google nos ofrece una herramienta de análisis llamado Google Trends⁸. Utilizando esta herramienta se realizó un análisis de 5 softwares de simulación para compararlos. Dentro

⁶ Rockwell Automation fué fundada en 2001 en EEUU, su principal servicio son tecnologías orientadas a la automatización de procesos industriales, donde ocupa una importante posición en el mercado de microcontroladores industriales, y es dueño de la popular línea Foxboro, famosa por sus constantes innovaciones.

⁷ Se puede encontrar una descripción de estos productos en el Anexo C – Descripción de productos.

⁸ Las gráficas de Google Trends representan con cuanta frecuencia se realiza una búsqueda particular. El eje horizontal de la gráfica representa el tiempo, y el eje vertical representa la frecuencia con la que se ha buscado el término en todo el mundo y en todos los idiomas.

de éste análisis se excluyó Arena Simulation Software, dado que la popularidad del mismo hace que la gráfica no quede apta para un análisis cualitativo.

Se eligieron PowerSim, GPSS, VenSim, VisSim y Promodel dado que son los softwares más utilizados en la UTN, tal cual podrá verse al final de este capítulo en el análisis de la encuesta.

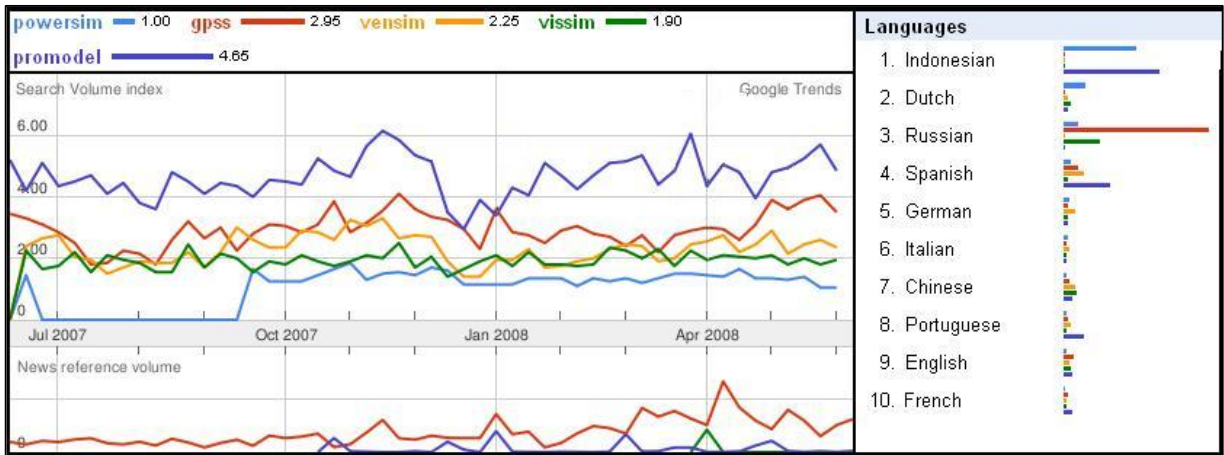


Figura 1 –Resultados de la consulta en Google Trends.

Aquí vemos que Promodel es el que mayor cantidad de búsquedas tuvo en los últimos 12 meses, seguido por GPSS, esto se traduce en la popularidad de cada uno.

Mercado Nacional

La Universidad Tecnológica Nacional (UTN.) fue creada el 14 de octubre de 1959 por medio de la ley 14.855, integrando desde ese entonces, el sistema universitario nacional. La UTN surge como Universidad Nacional con la función específica de crear, preservar y transmitir la técnica y la cultura universal en el campo de la tecnología, siendo la única Universidad Nacional del país cuya estructura académica tiene a las Ingenierías como objetivo central.

En 1948, se implementa el segundo nivel orientado a la especialización técnica para los trabajadores del ciclo de aprendizaje y capacitación de las escuelas dependientes de la Comisión Nacional de Aprendizaje y Orientación Profesional. Se establece la creación de la Universidad Obrera Nacional como institución superior de enseñanza técnica, dependiente de la citada Comisión, con el objeto de formar integralmente profesionales de origen laboral, destinados a satisfacer las necesidades de la industria argentina. Al egresado se le otorgaba el título de Ingeniero de Fábrica en las especialidades correspondientes. Para el ingreso a la Universidad se estableció que los alumnos, en todos los casos, debían desarrollar simultáneamente una actividad afín con la carrera que cursaran. Durante casi una década se produjo un rápido desarrollo a lo largo del país de esta modalidad educativa, que acompañaba la expansión industrial argentina de aquellas épocas. Este proceso requería un nivel cada vez más profundo en la formación de los egresados, lo que se tradujo en importantes cambios académicos y organizativos.

Luego de su creación, las características organizativas y académicas continuaron atrayendo a un número cada vez más elevado de alumnos, lo que llevó a la creación de nuevas Facultades Regionales. Hacia el año 1953 la institución ya tenía la estructura académica de Universidad, existiendo en ese entonces las Facultades Regionales de Buenos Aires, Córdoba, Mendoza, Rosario y Santa Fe, a las que se sumaron, en 1954, las Facultades Regionales de Bahía Blanca, La Plata, Tucumán y Avellaneda.

El 31 de agosto de 1962, la Asamblea Universitaria aprobó el Primer Estatuto de UTN., que al momento contaba con once Facultades Regionales.

A inicios de 1984, Argentina retoma definitivamente el camino de la democracia institucional. En el mes de junio de ese año, el Congreso de la Nación aprueba la Ley 23.068, destinada a normalizar las Universidades Nacionales, seriamente afectadas por las medidas tomadas en todos los campos por el ex gobierno de facto. Ésta ley permitió iniciar los trabajos destinados a devolver a la UTN. toda la fuerza creadora que necesitaba para reasumir su papel en la reconstrucción de la educación Universitaria.



Universidad Obrera Nacional, Actual UTN - FRBA

La UTN. viene participando activamente en el desarrollo de la Argentina, a través de su íntima vinculación con el sistema productivo. Desde su creación en el año 1959, han egresado más de 30.000 profesionales de sus carreras de Ingeniería.

La Universidad Tecnológica Nacional presenta características que la distinguen del resto del sistema Universitario Nacional.

- Es la única Universidad del país cuya estructura académica tiene a las Ingenierías como objetivo prioritario. Además de Ingeniería, se dictan Licenciaturas, Carreras de postgrado, como Doctorados y Maestrías, y varias carreras cortas.⁹
- Tiene carácter Federal, ya que cuenta con 24 Facultades Regionales y con 5 unidades académicas, distribuidas de manera estratégica abarcando todas las regiones de la Argentina, esto se traduce en una permanente e íntima vinculación con los sistemas productivos regionales y un activo intercambio académico a nivel Nacional. A raíz de su extensión geográfica la capacidad de absorción de alumnado es muy alta, actualmente alrededor de 70.000 cursantes, que equivale a más del 50% de todos los estudiantes de Ingeniería del país.

La UTN se ha convertido en el referente tecnológico Nacional. La visión y dedicación de sus autoridades y docentes, y el alto nivel académico logrado por los alumnos y los grupos

⁹ Para un informe detallado de las carreras ofrecidas en la UTN remítase al anexo A.

de investigación y desarrollo han hecho esto posible, posicionando a la UTN en el liderazgo en cuanto a formación de profesionales tecnológicos a nivel nacional.

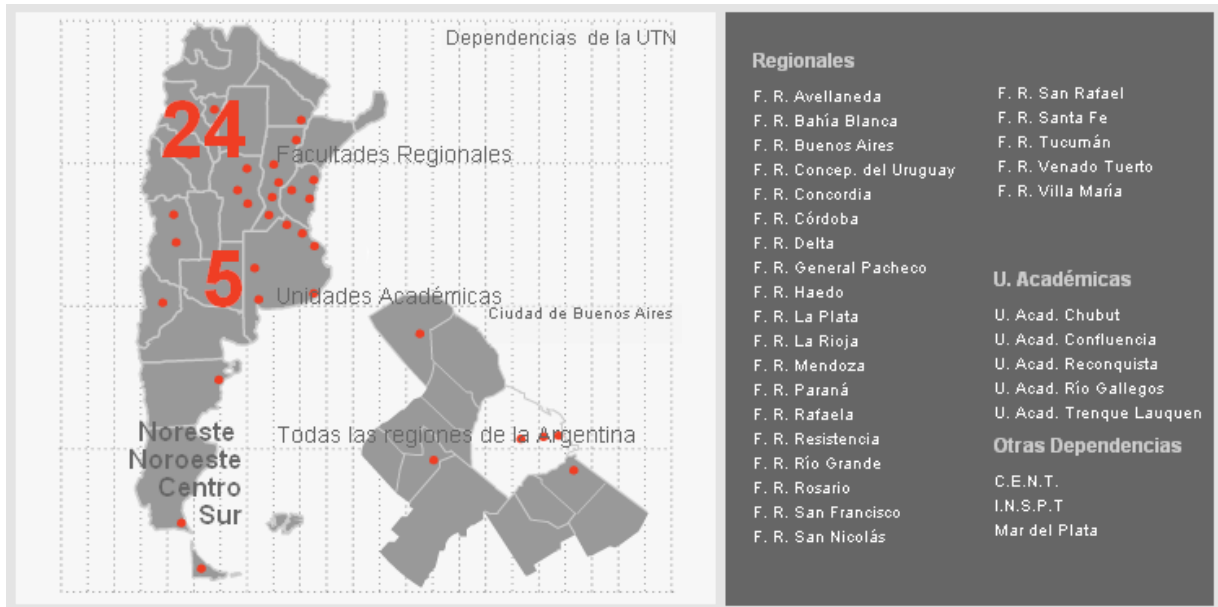


Figura 2 – Distribución geográfica de la UTN

La finalidad de este proyecto es cubrir una necesidad latente para el desempeño académico de todas las materias y cursos donde es necesaria la utilización de una herramienta de simulación práctica, simple y sin restricciones.

La UTN cuenta con un nivel académico excelente, y con las herramientas y el apoyo técnico necesario para lograr el desarrollo de la herramienta propuesta. El dictado de la carrera Ingeniería en Sistemas de Información facilita la tarea, dado que en la misma se ven las materias que tienen como eje a la matemática y el modelado, que dan los conocimientos básicos para la simulación.

Este proyecto dará la posibilidad de desarrollo profesional a los estudiantes. Si bien la dirección del mismo se llevará a cabo con profesionales dedicados, se contempla que todos los talentos necesarios para el desarrollo del mismo sea a través de mecanismos¹⁰ que permitan a los alumnos formar parte del desarrollo.

A través de este enfoque para el desarrollo, se obtendrá doble beneficio, por un lado solucionar un problema latente, que es la falta de una herramienta de simulación sin restricciones y con todas las ventajas que implica para la Institución ser propietario de la misma. Y por otro lado, el enriquecimiento académico que se logrará al desarrollarlo con los mismos estudiantes, esto último tendrá un impacto social muy positivo en la imagen de la UTN.

¹⁰ Entiéndase becas, pasantías o prácticas profesionales.

Descripción de la oferta y demanda

Para conocer la realidad del mercado objetivo, al que apunta este proyecto, se optó por recolectar información de toda la estructura académica de la UTN. Dadas las distancias entre las distintas facultades se llegó a la decisión de realizar una encuesta a todos los docentes involucrados. De la encuesta se obtuvo valiosa información que ayudó a entender mejor la complejidad de la oferta y la demanda.

Elección de la herramienta.

La encuesta se realizó con la versión gratuita de una herramienta on-line llamada e-encuesta (www.e-encuesta.com). La restricción que tiene esta versión es que se pueden realizar encuestas con un tope máximo de 100 respuestas, con la opción de comprar el excedente. Como en este caso la población es menor a ese número, la herramienta se adapta perfectamente para esta tarea.



Otro punto interesante encontrado para seleccionar esta herramienta, son las opciones de análisis que nos brinda. Por un lado nos permite crear una lista con las direcciones de correo electrónico y gestionar el envío de e-mails, con esta opción se envía un link distinto a cada participante para identificar quienes han respondido y quienes faltan, pudiendo enviar un recordatorio, cuando se crea conveniente, sólo a aquellos que no han respondido. Y por el otro, posee varios modelos de encuestas y preguntas para utilizar.

Por el lado del informe de resultados, nos encontramos con que las opciones son muy flexibles y simples a la vez, nos da varias posibilidades para realizar el análisis de la información recolectada. Además, podemos acceder a cada uno de los cuestionarios independientemente.

Elección de la población objetivo

El software de simulación de eventos discretos que abarca este proyecto, está destinado al uso académico dentro de la UTN para las carreras de grado que sean necesarias. Por este motivo, primero se realizó un relevamiento de todas las facultades regionales y unidades académicas para obtener información sobre las carreras de grado que se dictan en cada una de ellas. Un detalle completo de este relevamiento se muestra en el Cuadro 1.



Allí podemos ver que en las 24 facultades regionales y 5 regionales académicas se dictan 13 especialidades en Ingeniería¹¹. En el cuadro se incluye también el título intermedio Analista de Sistemas dado que la Regional Académica Trenque Lauquen dicta hasta este título, pudiendo terminar la cursada de Ingeniería en Sistemas en otra Facultad Regional.

¹¹ En este caso se omitieron las licenciaturas y los postgrados dado que en ninguna de ellas se encontró orientación alguna a la técnica de Simulación. Esto omite la posibilidad de que en alguna de ellas se pueda utilizar simuladores específicos.

Al mismo tiempo que se relevó esta información, se fueron tomando las estructuras curriculares de cada una de las carreras con el fin de individualizar las cátedras potenciales a utilizar un software de simulación con las características de este proyecto.

	Aeronáutica	Analista de sistemas	Civil	Eléctrica	Electromecánica	Electrónica	Industrial	Mecánica	Metalúrgica	Naval	Pesquera	Química	Sistemas de Información	Textil
F. R. Avellaneda			X	X		X	X	X				X		
F. R. Buenos Aires		X	X	X		X	X	X		X		X	X	X
F. R. Bahía Blanca			X	X		X		X						
F. R. Córdoba		X	X	X		X	X	X	X			X	X	
F. R. Concordia			X	X										
F. R. Concep. del Uruguay		X	X		X								X	
F. R. Delta		X		X				X				X	X	
F. R. General Pacheco			X	X				X						
F. R. Haedo	X					X	X	X						
F. R. La Plata		X	X	X			X	X		X		X	X	
F. R. La Rioja			X		X	X								
F. R. Mendoza		X	X		X	X						X	X	
F. R. Paraná			X		X	X								
F. R. Rafaela			X		X		X							
F. R. Resistencia		X			X							X	X	
F. R. Río Grande							X				X			
F. R. Rosario		X	X	X				X				X	X	
F. R. Santa Fe		X	X	X			X	X					X	
F. R. San Francisco		X			X	X						X	X	
F. R. San Nicolás				X		X	X	X	X					
F. R. San Rafael			X		X		X							
F. R. Tucumán		X	X	X		X		X					X	
F. R. Villa María		X				X		X				X	X	
F. R. Venado Tuerto			X		X									
R. Acad. Confluencia						X						X		
R. Acad. Chubut											X			
R. Acad. Río Gallegos					X		X							
R. Acad. Reconquista					X									
R. Acad. Trenque Lauquen		X												

Tabla 1 – Ingenierías por F.Regional y R.Académica.

Luego de procesar y analizar esta información, se llegó a la conclusión de que la cátedra que se ajusta es Simulación, la cual es dictada dentro de la currícula de Ingeniería en Sistemas de Información como obligatoria en el cuarto nivel de la carrera. Además se encontró esta cátedra en la carrera de Ingeniería Industrial en la FRBA como optativa.

Cabe aclarar que las cátedras *Diseño y Simulación de sistemas con redes PETRI*¹² y *Simulación y Optimización*¹³, no podrían utilizar un software con las características de

¹² De la carrera Ing. en Sistemas de Información dictada en F.R.Mendoza como electiva de 4to año.

¹³ De la carrera Ing. Industrial dictada como electiva de 5to nivel en F.R.Rafaela.

SimuRed, ya que el programa está orientado al análisis de modelos específicos, por tal motivo los sistemas de simulación que utilizan son de otro tipo.

Es muy importante tener en cuenta que el desarrollo de SimuRed dará la posibilidad de crear modelos estándares que podrán ser utilizados en cátedras como las anteriores, ya que al desarrollar el motor de simulación y tener el *Know How* dentro de la UTN se podrá continuar con desarrollos de este tipo, en base a este proyecto. Como ejemplo de esto último podemos mencionar las cátedras de Investigación Operativa (curricular de Ing. en Sistemas), Tránsito y Transporte (electiva de Ing. Civil), Robótica (curricular en Ing. Electrónica y electiva en Electromecánica), Automatización en Logística (curricular en Ing. Industrial) y otras tantas que, junto a los docentes de las mismas, podrían desarrollarse simuladores específicos para uso dentro del cursado de la cátedra, con un esfuerzo sorprendentemente bajo.

Luego de individualizar las cátedras potenciales a utilizar un software de simulación con las características de este proyecto, se procedió a obtener los nombres de los docentes, para ello se consultó en las páginas web de las respectivas instituciones, en ellas se encontró los nombres del 50% de los docentes. Para obtener el resto de los docentes se enviaron e-mails a las distintas oficinas de bedelía o al departamento de sistemas de las facultades faltantes. Inclusive hubo casos en los que no recibimos respuesta, y en éstos casos se obtuvo la información de los docentes a través de alumnos que tenían publicados apuntes on-line o a través de los grupos de Yahoo destinados a la cátedra.

La población total de todos los docentes de la cátedra de Simulación de todas las Facultades Regionales de la UTN resultó en 26, que incluye todos los cargos (profesores titulares, profesores adjuntos, JTPs, ayudantes), en todas las modalidades de contratación (Ordinario, Interino). Esta población está dividida en 12 Facultades Regionales.

Elección de las preguntas

El objetivo principal de la encuesta es el de conocer la realidad del dictado de la cátedra Simulación en cada una de las regionales. Para la redacción de las preguntas se utilizó el asistente de e-encuestas.com, se corroboró las mismas con dos profesionales de sistemas y se realizaron dos encuestas previas para verificar la comprensión de todas las preguntas.

La encuesta está dividida en 4 secciones, cada sección tiene una finalidad específica y las preguntas incluidas en ella fueron optimizadas para reducir el tiempo de respuesta. De esta manera garantizamos que cada persona que se dispone a responder concluya la misma.

La primer sección está destinada a identificar la persona que está respondiendo. Si bien e-encuestas tiene un gestor de usuarios, por el cual envía un e-mail y registra quien ha respondido, éste no contempla los casos en los cuales se responde a través de un acceso público. En ciertos casos se ha enviado el link de acceso público desde el e-mail personal, para que de esta manera el mensaje sea mas directo e individual, intentando así que sea mejor recibido por el destinatario. Otro punto de esta sección fue el de corroborar la información obtenida, como el cargo que desempeña.

La segunda sección nos da información sobre la marca, que software se utiliza, con que frecuencia y el tiempo que se lleva utilizándolo. Aquí lo que se trata es de obtener es la importancia que tiene el software de simulación dentro de la cátedra. Una cátedra que utiliza una sola marca, hace mas de 3 años y es usado por los alumnos todas las semanas nos está hablando de un software ya establecido, con un fuerte perfil dentro de la cátedra, y con pocas posibilidades de cambio. Por el contrario, una cátedra que esté utilizando un software desde hace poco tiempo nos habla de una cátedra sin un software establecido, en busca de uno. También vemos el uso que se tiene del software, ya que la cátedra puede recomendar alguno, pero no fomentar su uso (a través de trabajos prácticos o investigaciones).

La tercer sección busca la importancia que cada cátedra da a ciertos atributos del software utilizado. En este caso se decidió crear una matriz para facilitar las respuestas. La cuarta sección intenta encontrar el grado de satisfacción obtenido por el software utilizado y se recolectan las sugerencias que el encuestado crea necesarias.

En la cuarta sección, se agregó un campo para que, si la persona está interesada, pueda dejarnos su e-mail. De esta manera verificamos el e-mail mas usado por el docente y vemos si el docente tiene interés en nuestro proyecto.

Resultados

La respuesta de los encuestados fue bastante rápida. el 50% de las respuestas se consiguió en los primeros 3 días, esto nos da una pauta de la predisposición por parte de los docentes involucrados. Del total de encuestados respondieron el 58%, que se traduce en un 83% de las facultades regionales involucradas.

	Docentes	Regionales
Encuestados	26	12
Respuestas	15	10
Proporción	58%	83%

Tabla 2 – Respuestas de la encuesta.

Con respecto al cargo que desempeña cada uno de los encuestados, podemos ver que la mayoría se desempeña como profesor, un 64%, mientras que sólo el 14% de las respuestas fueron de ayudantes de cátedra. Cabe aclarar que en muchas regionales se dispone de un único representante de la cátedra, en este caso podemos mencionar la Facultad Regional San Francisco, y en otros casos tenemos varias cátedras de la misma materia, como es el caso de la Facultad Regional Santa Fe.

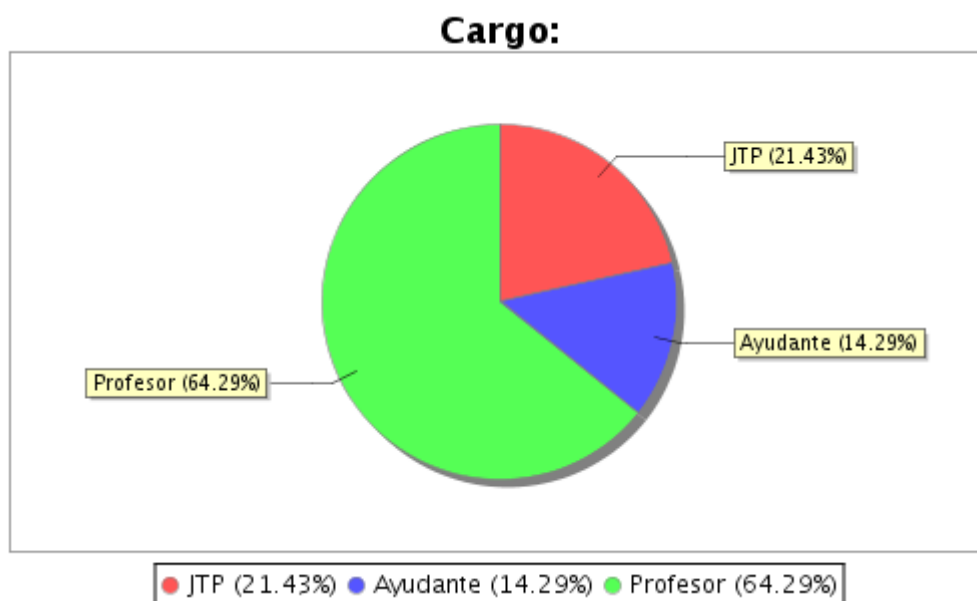


Figura 3 – Cargos ocupados por los encuestados.

En el 71% de los casos el tiempo que hace que se está usando el software de simulación elegido es de más de 3 años, lo que nos indica que una vez seleccionado el software e incorporado en la práctica de la cátedra éste se mantiene por el tiempo. Por otro lado podemos ver que sólo un 7% no utiliza software de simulación.

¿Cuánto tiempo lleva utilizando software de simulación para su curso?

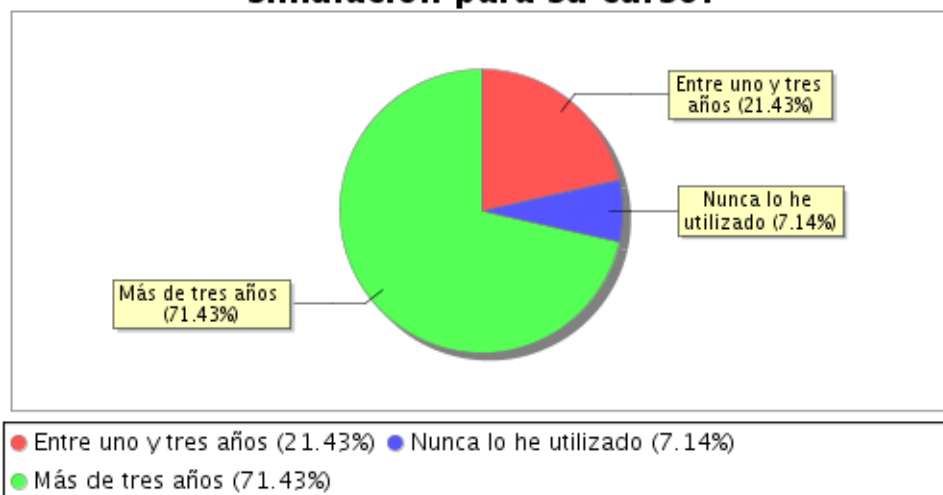


Figura 4 – Tiempos de uso.

Si bien la cátedra puede recomendar el uso de un software específico, es importante saber con qué frecuencia se utiliza, dado que a mayor cantidad de uso, mayor será la importancia que tenga el software dentro de la cátedra. Como podemos ver en los resultados, la utilización del software es alta, alcanzando el 71% de los casos a utilizarlo todos los meses de la cursada. Sólo el 14% de los casos no utilizan el software recomendado.

¿Con qué frecuencia utilizan sus alumnos el software recomendado?

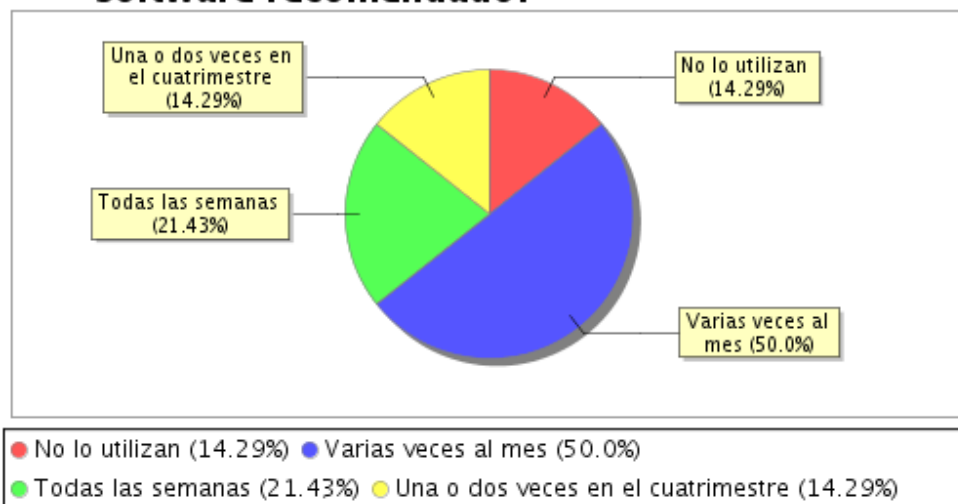


Figura 5 – Frecuencia de uso.

En torno a la satisfacción percibida por los docentes podemos ver que el 100% de los mismos se encuentra satisfecho. Vemos que un 20% está muy satisfecho, esto podría llegar a interpretarse como que estos docentes se encuentran cómodos y confiados en el software actualmente utilizado y que proponer un cambio puede no llegar a ser bienvenido.

¿Cuál es su grado de satisfacción en general con el software que utiliza actualmente?

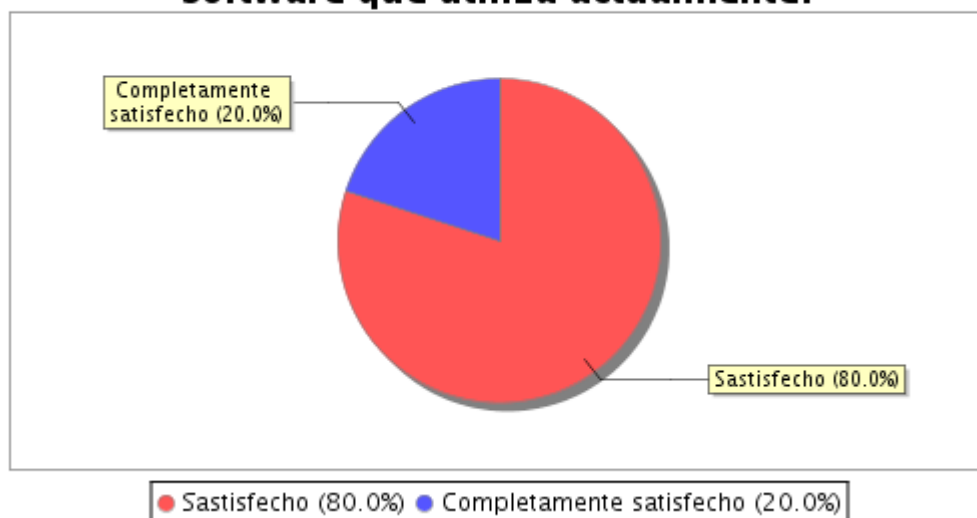


Figura 6 –Satisfacción.

En contraposición con la satisfacción, hubo 5 comentarios (un 33%) que resaltaron cambios en el software que están utilizando actualmente. Los cambios propuestos tienen que ver con la interfase y con las limitaciones impuestas en las versiones académicas. Las recomendaciones son:

Si pudiese cambiar algo del software que utiliza actualmente ¿qué cambiaría?	
1	La interfase
2	No tener la limitaciones que se presentan (por ejemplo cantidad de líneas de código en el GPSS) dado que la mayoría de los Sw son versiones estudiantiles o free
3	Limitaciones en el tamaño de los modelos (cantidad de objetos y entidades) en la versión académica.
4	La interfaz, que debería ser gráfica. Aumentar su capacidad de nodos. Facilitar la programación de líneas de software adicionales a los nodos (if then else, etc).
	En Octave, el cambio importante necesario sería la documentación de base.
5	Matlab es un soft 100% útil y sin problemas. Powersim, nuevamente la documentación respaldatoria

Tabla 3 –Cambios deseados.

En la pregunta: “A la hora de seleccionar el software de simulación, ¿cuál fue la importancia que le dio usted a cada uno de los siguientes aspectos?” se realizó una ponderación de las mismas para obtener un valor representativo de cada uno de los aspectos evaluados. Así las repuestas obtenidas fueron orientadas hacia el punto máximo de respuestas y se calculó la media en base al porcentaje obtenido.

Aspecto Evaluado	Resultado
Tipos de Licencia	Muy Importante (60%)
Portabilidad	Poco Importante (55%)
Interfaz	Importante (66%)
Calidad	Muy Importante (65%)
Funcionalidad	Importante (60%)
Instalación	Poco Importante (90%)
Recursos necesarios	Importante (48%)
Manual de usuario	Poco Importante (63%)
Soporte técnico	Importante (67%)
Empresa distribuidora	Nada Importante (62%)

Tabla 4 – Importancia sobre aspectos evaluados.

De este resultado, podemos observar que los aspectos que más se destacan son los tipos de licencia y la calidad del software. En menor medida se evalúa la interfaz, funcionalidad, recursos necesarios y soporte técnico.

Por el contrario, se ven como poco importantes la portabilidad, la instalación y el manual de usuario. Finalmente a la mayoría no le interesa la empresa distribuidora.

Con respecto al software utilizado, se relevaron las marcas que se detallan en la Tabla 5. En esta tabla podemos ver que hay 2 marcas utilizadas en 3 Regionales, 4 marcas utilizadas en 2 regionales y 11 marcas que sólo son usadas en una regional¹⁴.

¹⁴ Para un detalle de estos productos remitirse al Anexo C.

Marca	# FR	Regionales
GPSS	3	FRM / FRSF / FRVM
Vensim	3	FRM / FRRE / FRSF
Arena - Rockwell	2	FRVM / FRBA
MATLAB	2	FRC / FRLP
Powersim	2	FRLP / FRSFCO
Promodel	2	FRM / FRRE
20Sim	1	FRSFCO
Desarrollo propio	1	FRC
DEVS JAVA	1	FRSF
Excel	1	FRC
EXTEND	1	FRM
Octave	1	FRLP
Quest - Dassault	1	FRBA
Simnet	1	FRD
Taylor	1	FRSF
Tutsim	1	FRSFCO
Vissim	1	FRSFCO

Regionales	Software que utilizan
FRBA	Arena - Rockwell
FRBA	Quest - Dassault
FRC	Desarrollo propio
FRC	Excel
FRC	Matlab
FRD	Simnet
FRLP	Matlab
FRLP	Octave
FRLP	Powersim
FRM	Extend
FRM	GPSS
FRM	Promodel
FRM	Vensim
FRRE	Promodel
FRRE	Vensim
FRSF	DEVS JAVA
FRSF	GPSS
FRSF	Taylor
FRSF	Vensim
FRSFCO	20Sim
FRSFCO	Powersim
FRSFCO	Tutsim
FRSFCO	Vissim
FRVM	Arena
FRVM	GPSS
FRT	NO USA

Tabla 5 – Software utilizado.

Para culminar se realizó un estudio de cada uno de los diferentes productos utilizados destacando las ventajas y desventajas de cada uno.

Producto	Ventajas	Desventajas
GPSS	<ul style="list-style-type: none"> - Muy simple de usar. - Tiene una fuerte orientación académica. - Hay mucha documentación on-line disponible. 	<ul style="list-style-type: none"> - El valor de la licencia es por cada usuario (envían un solo CD). - GPSS es un lenguaje muy antiguo y no posee características de modelado modernas. - Restricciones propias del lenguaje.
Vensim	<ul style="list-style-type: none"> - Gratis para uso académico (versión PLE). - Muy fácil de utilizar. - Excelente manual de usuario, con opción de envío de un manual impreso. 	<ul style="list-style-type: none"> - No provee un informe detallado para el análisis de las características del modelo. - Interfaz de usuario desactualizada. - En su versión PLE no incluye funciones básicas.

Arena - Rockwell	<ul style="list-style-type: none"> - Líder del mercado. - Excelente respuesta. - Gran cantidad de gráficos disponibles. 	<ul style="list-style-type: none"> - La versión académica no está disponible para Latino América. - Es una herramienta muy completa, por lo que se necesita un aprendizaje previo del software. - No permite crear un modelo desde cero, todos los modelos se deben crear a partir de Templates (salvo en la edición profesional).
MATLAB	<ul style="list-style-type: none"> - Software de Investigación y Desarrollo para aplicaciones de Ingeniería. - Gran cantidad de herramientas disponibles. - Posibilidad de crear bibliotecas de usuario. - Articulación con otros lenguajes de programación. 	<ul style="list-style-type: none"> - Simulink es el software para simulación, que se encuentra embebido en Matlab. Por lo tanto se debe instalar Matlab para usar Simulink. - Software muy complicado para utilizar, se debe tener una capacitación previa. - La salida gráfica del sistema es sólo a través de diagramas de ejes. - Posee un lenguaje de programación propio.
Powersim	<ul style="list-style-type: none"> - Trayectoria de 15 años en el mercado. - Muy buena interfaz de usuario. - Facilidad de uso. - Orientado al modelo. 	<ul style="list-style-type: none"> - Restricciones en el modelo para la versión académica. - No posee soporte técnico para la versión académica.
Promodel	<ul style="list-style-type: none"> - No requiere programación adicional para sus modelos. - Posee un módulo de optimización que orienta al analista a realizar cambios. - Integración con Excel y otros productos Microsoft. 	<ul style="list-style-type: none"> - Posee licencias académicas pero con un precio anual, este valor no incluye a Argentina. - La versión académica tiene grandes restricciones de uso.
20Sim	<ul style="list-style-type: none"> - Excelente interfaz de diseño, tiene las características mas sobresalientes de los mejores IDEs de desarrollo. - Tiene nodos específicos de Ingeniería, como masas, resortes y amortiguadores, con su comportamiento preestablecido. - Gráficos de salida. 	<ul style="list-style-type: none"> - Precio. - Renovación de licencias anualmente.
DEVS JAVA	<ul style="list-style-type: none"> - Nuevo concepto en modelado de Simulación. - Gran cantidad de desarrollos bajo este concepto. - Al estar desarrollado en Java es OpenSource, o sea que se puede obtener el código fuente. 	<ul style="list-style-type: none"> - Se trata de un paquete de desarrollo bajo Java, por lo que se necesita conocer el lenguaje de programación. - Se debe desarrollar una aplicación que implemente el paquete DEVS Java.
Excel	<ul style="list-style-type: none"> - Amplia aceptación de los usuarios. - Flexibilidad en el análisis de datos. 	<ul style="list-style-type: none"> - Problemas de compatibilidad con las librerías utilizadas en las macros. - Imposibilidad de realizar animaciones. - Demanda mucho tiempo en el desarrollo del modelo y conocimientos avanzados del paquete de Office.
ExtendSim	<ul style="list-style-type: none"> - La arquitectura implementada es muy flexible. - Permite el trabajo en Grupo. - Disponibilidad de recursos obligada por el tipo de licencia. 	<ul style="list-style-type: none"> - Limitaciones estrictas en cuanto a la cantidad de recursos disponibles en las versiones académicas.

Octave	<ul style="list-style-type: none"> - Licencias GNU. - Tiene todas las características de Matlab. - Desarrollado en una Universidad. - Existe una comunidad activa en Internet, donde pueden realizarse consultas. 	<ul style="list-style-type: none"> - Utiliza un lenguaje de programación propio (distinto a Matlab). - Para utilizar gráficos se necesita instalar otra aplicación. - Interfaz muy poco cuidada.
Dassault	<ul style="list-style-type: none"> - Modelado 100% gráfico. - Excelente calidad y exactitud gráfica. 	<ul style="list-style-type: none"> - Funciona integrado con otros productos de la empresa, formando un paquete de aplicaciones. - Usa innovación tecnológica, con lo cual las técnicas clásicas de Simulación podrían estar omitidas.
Simnet II	<ul style="list-style-type: none"> - Utiliza teoría de colas en todos sus nodos. - No necesita de instalación para correr. - La codificación del modelo se resolvió mediante un pseudo-lenguaje muy práctico y sencillo. - Muy estable en situaciones de saturación. 	<ul style="list-style-type: none"> - No posee una interfaz amigable, la misma se ejecuta fuera del entorno visual, inhabilitando el uso del mouse. - Si bien los resultados son fáciles de entender, los mismos se presentan en texto plano, sin ningún formato, lo que dificulta la lectura del mismo. - Es muy difícil conseguir un manual que posibilite la consulta a la hora de transferir el modelo. - Tiene limitaciones en cuanto a la cantidad de entidades. - La funcionalidad adicional debe ser programada en FORTRAN, lenguaje potente y simple de aprender pero actualmente no es muy popular.
Taylor	<ul style="list-style-type: none"> - Buen programa académico. - Información adicional para estudiantes. - Capacitación para docentes. 	<ul style="list-style-type: none"> - Está limitado a 30 átomos (esto es la creación de 30 entidades dentro del modelo).
Tutsim	<ul style="list-style-type: none"> - Tiene nodos específicos de Ingeniería, como masas, resortes y amortiguadores, con su comportamiento preestablecido. 	<ul style="list-style-type: none"> - No tiene actualizaciones en los últimos años.
Vissim	<ul style="list-style-type: none"> - VisSim posee un programa de uso académico. - Con una sola licencia se cubren todas las personas del departamento que solicitó la licencia. - Los integrantes del equipo pueden instalar libremente el software en otras máquinas, inclusive externas a la institución. - Tiene soporte técnico. - Buena calidad de gráficos en las simulaciones. 	<ul style="list-style-type: none"> - El programa de licenciamiento académico no está disponible para Argentina. - Hay que renovar licencias todos los años. - El uso académico está sólo disponible para una versión que no tiene actualizaciones desde hace años. Si se desea usar una más moderna se debe pagar un extra. - El soporte técnico esta sujeto a disponibilidad.

Tabla 6 – Principales características del software utilizado.

Finalmente, la cantidad de encuestados que nos dejaron su dirección de correo electrónico porque encontraron interesante este proyecto son 9, lo que se traduce en un 60% de los encuestados.

Comparación de productos

La totalidad del software expuesto en este capítulo es extranjero. Países como Estados Unidos, Suiza y Netherland son los principales proveedores de paquetes comerciales de simuladores sin fines específicos. Las opciones disponibles son de excelente calidad, pero como se comentó anteriormente, no tienen programas académicos libres de restricciones.

La comercialización del software de simulación está tan globalizada, y son tan pocas las opciones, que la competencia se realiza sin importar el origen de la empresa que lo provee. Así, un producto como PowerSim, de origen noruego, está presente en el mercado norteamericano de la misma manera que Rockwell Arena o Vensim.

A nivel nacional no existen productos de software con las características de SimuReD. Esta característica se extiende al Continente Americano. Una excepción a este punto es el caso de ProModel, que posee oficinas comerciales en San Pablo, Brasil, como representante para todo Latino-America.

Por un lado esto podría parecer lógico, después de todo, como vimos anteriormente, la competencia en materia de simuladores no reconoce fronteras. Pero por otro lado, al utilizar Google Trends¹⁵ para realizar una comparación de el volumen de búsquedas de la palabra Simulador en inglés y castellano, se puede observar que a partir del año 2006 las búsquedas de simuladores en español superan a las búsquedas en inglés.

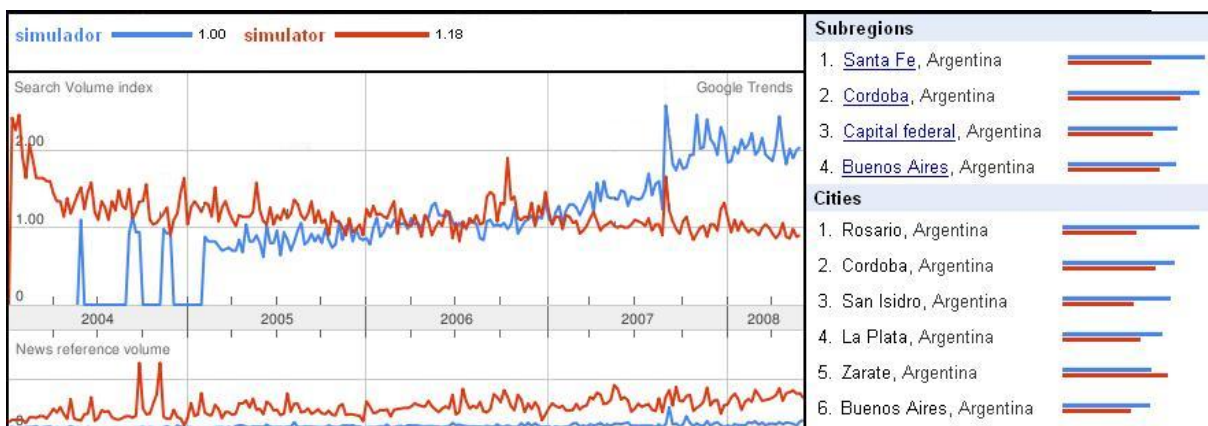


Figura 7 – Volumen de búsquedas en Google

Esto último se traduce en que el mercado de los simuladores en America latina está creciendo mucho más que el mismo mercado en países anglosajones.

En un nivel inicial, SimuReD será un verdadero éxito a nivel académico dentro de la UTN, pero luego podría convertirse en un producto de competencia equitativa en el mercado internacional, dando a la UTN y a la facultad que lo desarrolle, un status y reconocimiento acorde.

¹⁵ Herramienta gratuita de google la cual mide el volumen de búsquedas realizadas con la palabra seleccionada. También permite realizar comparaciones entre dos palabras clave.

Conclusiones

SimuReD nació a partir de una necesidad dentro del ambiente académico de la UTN-FRD. Se pensó que desarrollar un software de simulación discreto para uso académico podría ser un gran aporte para la institución. Consultando el mercado internacional se vió que muchas de las herramientas disponibles nacieron a partir de proyectos incubados dentro de Universidades, pero lamentablemente, sólo unos pocos tienen orientación académica o planes de licencias acorde a las necesidades Nacionales.

El relevamiento realizado en las distintas filiales de la UTN mostró una clara aceptación del producto y un gran interés en el desarrollo del mismo por parte de los docentes de las cátedras que utilizan un software de este tipo.

Teniendo en cuenta que no se trata de un desarrollo comercial, la ganancia de este producto se medirá en el conocimiento adquirido y en el valor agregado a la Institución.

Capítulo 3 - Tecnologías

Introducción	40
Metodologías	40
Arquitecturas	46
Herramientas de Desarrollo	49
Gráficos y Animación	53
Web Services	55
Requerimientos Operativos	56
Conclusiones	57
Bibliografía	57

Introducción

Seleccionar las Tecnologías para el desarrollo de SimuReD es seleccionar las herramientas tecnológicas de diseño, soporte e infraestructura para un producto de software. Este software debe tener exactitud en los cálculos matemáticos, ya que un simulador exige alta precisión en los resultados y se debe contemplar que un desarrollo de este tipo posee una alta carga de acceso a cálculos probabilísticas, los cuales deben realizarse en un tiempo aceptable y no conviertan a SimuReD en un fracaso operativo.

El equipo no va a ser estable durante el desarrollo, ya que los integrantes se reclutarán a través del sistema de becas y/o pasantías de la facultad, por este motivo debe existir una herramienta de versionado de documentos para un control y seguimiento de los mismos.

Dado que el sistema de becas tiene una carga horaria inferior a una jornada de trabajo normal¹, a la hora de seleccionar las tecnologías a utilizar se deberá tener en cuenta las herramientas utilizadas en las cátedras de la UTN-FRD para reducir el tiempo de entrenamiento y adaptación de los miembros que ingresan al equipo de trabajo.

Finalmente, se analizarán los requerimientos de infraestructura necesarios para cada caso.

Metodologías

Para el desarrollo de SimuReD se necesita adoptar una metodología que contemple las características más distintivas de este tipo de software.

Se prioriza el trabajo en equipo, en este caso, de un equipo formado por alumnos de la facultad, que en su gran mayoría no poseen experiencia previa en el desarrollo de software a nivel laboral, pero si los conocimientos académicos necesarios, obtenidos a través de las prácticas obligatorias en las cátedras de la carrera. Por otro lado, la disponibilidad de recursos se verá reducida (al utilizar los métodos de contratación de becas / pasantías disponibles en la facultad), por lo que un desarrollo ágil, rápido y dinámico, que no requiera muchos formalismos es un factor deseable, pero a la vez deberá tener todas las características necesarias para soportar la rotación de los miembros del equipo.

La evolución del proyecto debe estar debidamente documentada y versionada, para evitar la pérdida de trabajo y para facilitar el ingreso de nuevos integrantes al equipo (los integrantes cambiarán a lo largo del proyecto al menos una vez al año).

Dentro de las alternativas disponibles, tenemos 3 metodologías de interés: RAD (Rapid Application Development), RUP (Rational Unified Process) y T4G (Técnicas de Cuarta Generación). A continuación se analiza cada una de ellas.

¹ Se considera una jornada de 8 horas diarias, de Lunes a Viernes, lo que completa 40 horas efectivas de trabajo semanal contra 6 horas efectivas de trabajo semanal en la Beca.

RAD (Rapid Application Development).

Esta metodología es nueva y un tanto revolucionaria. Algunos autores hablan de un cambio en el paradigma de las metodologías de desarrollo de software. Lo cierto es que RAD propone cambios radicales, y muchos de estos nuevos enfoques pueden ser aplicables al desarrollo de SimuReD.

Una característica muy interesante de RAD, posible de aplicar en este proyecto, es la técnica denominada Programación Extrema (Extreme Programming o XP). A continuación se transcribe una explicación de esta metodología.

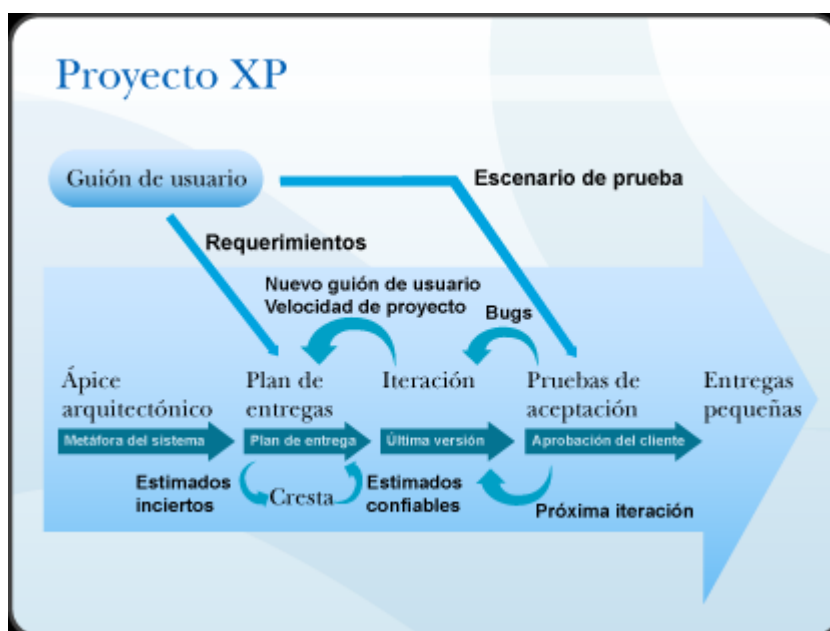


Figura 1 – Diagrama de la Programación Extrema

“En la programación extrema se da por supuesto que es imposible prever todo antes de empezar a codificar. Es bastante normal hacer un diseño, ponerse a codificar, ver que hay faltantes o errores en el diseño, empezar a codificar fuera del diseño y al final el código y el diseño, o no se parecen, o hemos destinado valioso tiempo en cambiar la documentación de diseño para que se parezca al código.

En vez de tratar de luchar contra todo eso, se lo asume y se busca una forma de trabajar que se adapte fácilmente a esas circunstancias. Básicamente la idea de la programación extrema consiste en trabajar estrechamente con el cliente, haciéndole mini-versiones con mucha frecuencia (cada dos semanas). En cada mini-versión se debe hacer el mínimo de código y lo más simple posible para que funcione correctamente. El diseño se hace sobre la marcha, haciendo un mini-diseño para la primera mini-versión y luego modificándolo en las siguientes mini-versiones. Además, no hay que hacer una documentación para el diseño, no hay mejor documentación que el mismo código. El código, por tanto, también se modifica continuamente de mini-versión en mini-versión, añadiéndole funcionalidad y extrayendo sus partes comunes.”²

² Extraído de <http://www.chuidiang.com/ood/metodologia/extrema.php> (21/07/2008)

Como se puede ver, esta metodología es ágil y tiene las características para un desarrollo rápido y dinámico. Además favorece la reutilización de componentes y el desarrollo incremental, favoreciendo la corrección de errores en forma progresiva.

Para utilizar esta metodología, se comienza por una reunión, donde se establecen las necesidades primarias con el usuario, se estima el tiempo de desarrollo y se comienza a trabajar sobre el plan de pruebas (antes de codificar). Estas pruebas deben ser automáticas, de forma que haya un programa de pruebas que ejecutemos y nos diga si los requerimientos se cumplen.

Los programadores trabajan en parejas (dos personas por PC), el código se hace de la forma más sencilla posible, lo mínimo imprescindible, para que se pasen las pruebas automáticas. Las parejas de programadores se deben intercambiar de forma que todos trabajan con todos, de esta manera se incrementa la calidad del código, ya que el mismo no está sujeto al criterio de una sola persona. Además, los programadores juniors³ aprenderán de los seniors⁴ al trabajar con ellos. Todo el código que realiza una pareja tendrá mas probabilidad de ser reutilizado ya que hay dos programadores que lo saben y que lo reutilizarán cuando puedan, enseñándoselo a sus nuevos compañeros. De esta manera el conocimiento del código ya hecho se propaga de forma natural entre todos los programadores del equipo. Además, cualquier pareja puede tocar el código ya hecho por otras siempre y cuando sea para optimizar el proceso.

Una de las mayores objeciones a esta metodología es la falta de documentación, la cual se hace absolutamente necesaria para un proyecto con las características de SimuReD. Se recuerda que en este caso, los integrantes del equipo saldrán del proyecto todos al mismo tiempo, con lo que se perdería parte de la esencia de la programación extrema, que es transmitir los conocimientos entre todos los integrantes del equipo. Asimismo, se deberá considerar que al finalizar el proyecto deberá haber disponible un manual de usuario, para el cual es importante contar con la documentación del desarrollo.

RUP (Rational Unified Process)

RUP es tal vez la metodología mas utilizada en la actualidad, ya que está orientada al desarrollo OO (Orientado a Objetos) y utiliza UML (Unified Modeling Lenguaje) como herramienta de análisis y diseño.

Los principios básicos⁵ de RUP son 5:

- Adaptar el proceso: El proceso deberá adaptarse a las características propias del proyecto u organización. El tamaño del mismo, así como su tipo o las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto.
- Balancear prioridades: Los requerimientos de los diversos participantes pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un balance que satisfaga los deseos de todos.

³ Seniority asociado a un programador inexperto o en la etapa inicial dentro de la tecnología.

⁴ Seniority asociado a un programador experto, usualmente con mas de 3 años de experiencia en la tecnología asociada.

⁵ Extraídas de Wikipedia: <http://es.wikipedia.org/wiki/RUP> (22/07/2008)

- Demostrar valor iterativamente: Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los usuarios, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.
- Elevar el nivel de abstracción: Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o marcos de referencia (Framework) por nombrar algunos. Esto evita pasar directamente de los requisitos a la codificación, sin saber con certeza qué codificar para satisfacer de la mejor manera los requerimientos y sin comenzar desde un principio pensando en la reutilización del código. Un alto nivel de abstracción también permite discusiones sobre diversos niveles y soluciones arquitectónicas. Éstas se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con el lenguaje UML.
- Enfocarse en la calidad: El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente.

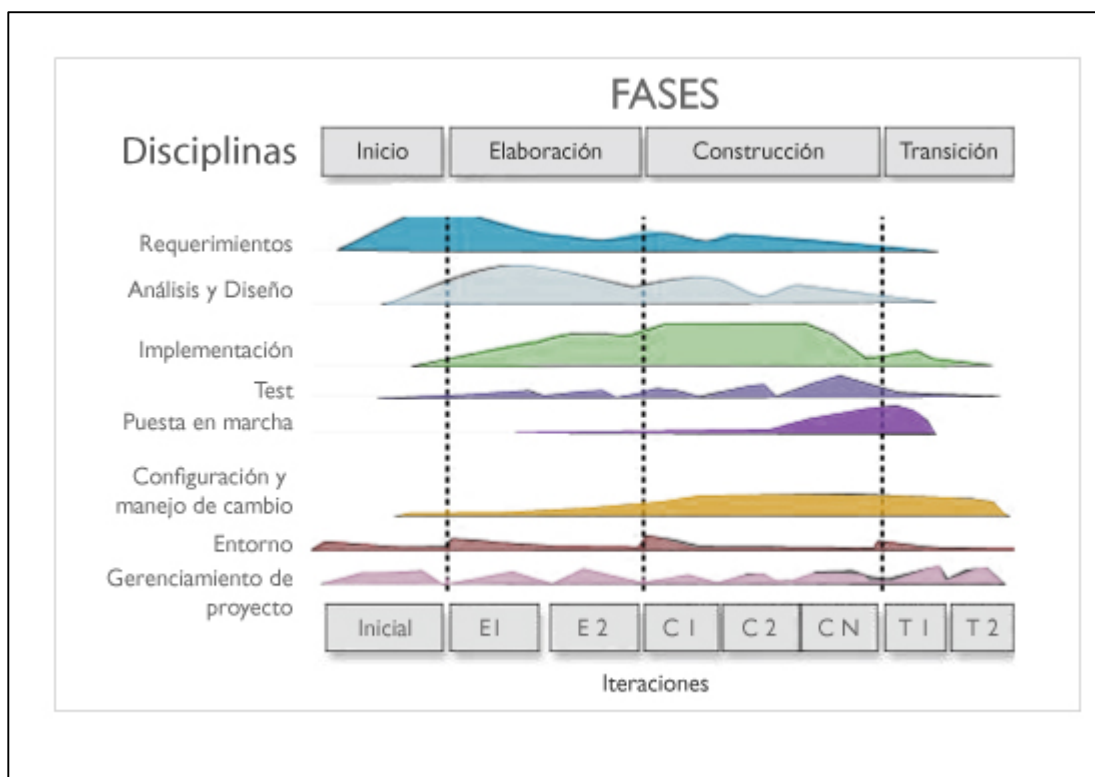


Figura 2 – Fases y Disciplinas de RUP

Las etapas básicas para un desarrollo son esencialmente 4. En cada una de estas etapas se encuentran herramientas de UML adecuadas, con las cuales pueden realizarse un correcto relevamiento que sirven de soporte a la siguiente etapa, a otro grupo del equipo y hasta inclusive al usuario. Cada una de las etapas tiene las siguientes características generales:

- **Análisis de Requerimientos:** En esta etapa se logra claridad sobre lo que desea el usuario y la forma en la cual se le va a presentar la solución que está buscando. Dentro de esta etapa se encuentran las siguientes actividades:

- Identificar los Casos de Uso⁶ (CU) del sistema: aquí se deben definir los actores (usuarios que interactuarán con el sistema), los casos de uso y las relaciones entre ellos.
 - Detallar los CU descriptos: definir su objetivo y dar una descripción detallada de la funcionalidad. Realizar los diagramas de interacción necesarios⁷.
 - Definir la interfaz: dibujar las pantallas de la aplicación.
 - Definir el modelo del mundo: se define la estructura de Clases, los métodos y los paquetes de clases. Aquí se presta especial atención a la reutilización de componentes y a los patrones de software.
 - Validar los modelos obtenidos.
- Diseño del Sistema: Se define una subdivisión en aplicaciones del sistema y la forma en que se comunican, esto puede expresarse por medio de asociaciones de dependencia entre los nodos, componentes o aplicaciones. También, se identifica y valida la arquitectura del sistema.
 - Diseño Detallado: En esta etapa se analiza el Framework⁸ y se realizan todos los diagramas de clases y las divisiones en paquetes. Se desarrolla el modelo de interfaz, así como los modelos de control, persistencia y comunicaciones. Se definen excepciones y se incorporan los patrones de diseño convenientes.
 - Implementación y Pruebas: Es la etapa final y aquí se desarrolla el código cumpliendo con los estándares de programación, utilizando toda la documentación generada. También se deberán definir los casos de prueba, los cuales deberán realizarse antes de empezar con el desarrollo del código.

La utilización de RUP, supone la administración de gran cantidad de documentación y de recursos. Esto debe ser analizado en relación al tamaño del proyecto, ya que para proyectos demasiado chicos puede ser una metodología que resulte contra productiva. Pero también se debe evaluar que la documentación obtenida en todas las etapas de desarrollo es muy valiosa y facilita la incorporación de personas al proyecto.

T4G (Técnicas de Cuarta Generación)

El término Técnicas de Cuarta Generación (T4G) se trata del uso de una herramienta de software donde ha de especificarse algunas características de alto nivel. Luego la herramienta genera automáticamente el código fuente basándose en la especificación detallada. T4G se orienta hacia la habilidad de especificar software a un nivel que sea más próximo al lenguaje natural o a una notación que proporcione funciones significativas, alejándose de esta manera de la abstracción de lenguaje de programación.

T4G comienza con la recolección de requisitos. Idealmente el cliente debe describir los requerimientos y estos deben traducirse directamente en un prototipo operacional pero esto no funciona así. Por lo general, el cliente no está seguro de lo que necesita, puede ser ambiguo en la especificación de hechos que son conocidos y puede ser incapaz o no desear especificar la

⁶ Caso de Uso es un documento donde se detalla una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema.

⁷ Diagramas de Casos de Uso, Diagramas de Secuencia, Diagramas de Colaboración.

⁸ Un Framework, es una estructura de software que trae implementadas soluciones a los problemas comunes entre aplicaciones, como ser conexión a bases de datos, emisión de reportes, etc.

información en la forma que una herramienta T4G la necesita. Para aplicaciones chicas puede ser posible ir directamente desde la recolección de requisitos a la implementación, usando un lenguaje de cuarta generación no procedimental.

Hay algunos méritos en las razones de cada parte. Aunque es algo difícil separar los hechos de las suposiciones es posible resumir el estado actual de los métodos T4G:

1. Con muy pocas excepciones el dominio de aplicación actual de las T4G esta limitada a las aplicaciones de sistemas de información comerciales, específicamente al análisis de información y de la obtención de informes en las grandes bases de datos. Hasta ahora T4G se han usado muy poco en productos de ingeniería y áreas de aplicación de sistemas.

2. La recolección de datos preliminares que acompañan al uso de T4G parece indicar que el tiempo requerido para producir software se reduce mucho para aplicaciones pequeñas de trabajo medio así como también la cantidad e análisis y diseño.

3. Sin embargo el uso de T4G para grandes trabajos de desarrollo de software exige el mismo o más tiempo de análisis, diseño y prueba perdiéndose así un tiempo sustancial que se ahorra mediante la eliminación de la codificación.



Figura 3 - Etapas de la metodología T4G

Para resumir las técnicas de cuarta generación se describen una serie de ventajas y desventajas referentes a la adopción de la misma.

Ventajas:

- Reducción drástica en el tiempo de desarrollo.
- Mejora significativa en la productividad.

Desventajas:

- Las herramientas actuales de T4G no son más fáciles de utilizar que los lenguajes de programación, y el código generado es "ineficiente".

- El mantenimiento de grandes sistemas desarrollados mediante T4G, es cuestionable.
- No se ha utilizado en desarrollos de Ingeniería.

Arquitecturas

“La arquitectura de software, tiene que ver con el diseño y la implementación de estructuras de software de alto nivel. Es el resultado de ensamblar un cierto número de elementos arquitectónicos de forma adecuada para satisfacer la mayor funcionalidad y requerimientos de desempeño de un sistema, así como requerimientos no funcionales, como la confiabilidad, escalabilidad, portabilidad y disponibilidad.”⁹

La arquitectura de software define, de manera abstracta, los componentes que llevan a cabo alguna tarea, sus interfaces y la comunicación ente ellos. Toda arquitectura debe ser implementable en una arquitectura física, que consiste simplemente en determinar qué computadora tendrá asignada cada tarea.

Para definir la arquitectura de SimuReD, se debe tener en cuenta dos puntos importantes, que están definidos en los objetivos específicos. Por un lado, la aplicación debe ser multiplataforma y por el otro lado se debe contemplar la infraestructura necesaria para que pueda compartir conocimientos a través de una aplicación del tipo Web 2.0.

Hay dos arquitecturas aplicables (según las características de SimuReD), que uniéndolas se puede obtener un combo ideal para el desarrollo de este proyecto. A continuación se dará una descripción de ambas arquitecturas.

Cliente-servidor¹⁰

Esta arquitectura consiste básicamente en un programa denominado el cliente, que realiza peticiones a otro programa denominado el servidor, el cual le da una respuesta al cliente que lo ha solicitado. Aunque esta idea se puede aplicar a programas que se ejecutan sobre una sola computadora tiene mayores ventajas en un sistema operativo multiusuario distribuido a través de una red de computadoras.

En esta arquitectura la capacidad de procesamiento está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa. Los tipos específicos de servidores incluyen los servidores Web, los servidores de archivo, los servidores del correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma.

⁹ Definición publicada por Philippe Kruchten en su libro The Rational Unified Process Made Easy : A Practitioner's Guide to the RUP (Addison-Wesley – 2003)

¹⁰ Adaptado del texto publicado en Wikipedia.org

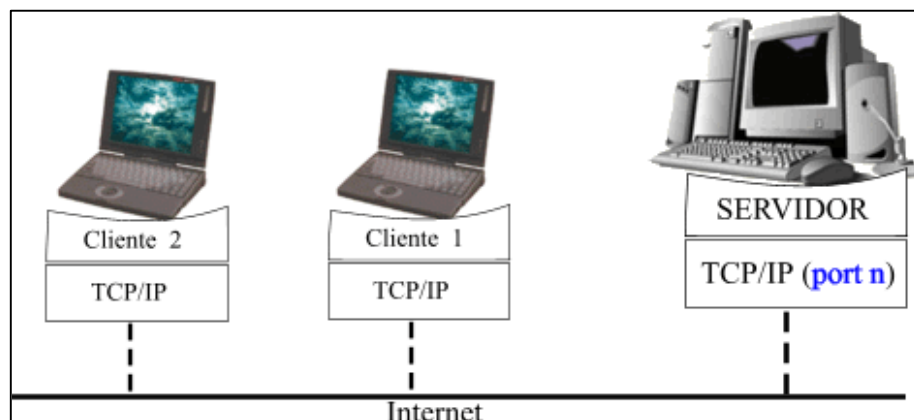


Figura 4 – Diagrama de una arquitectura Cliente-Servidor

Una disposición muy común son los sistemas multicapa en los que el servidor se descompone en diferentes programas que pueden ser ejecutados por diferentes computadoras aumentando así el grado de distribución del sistema.

Multinivel

Las arquitecturas multinivel se basan en la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.

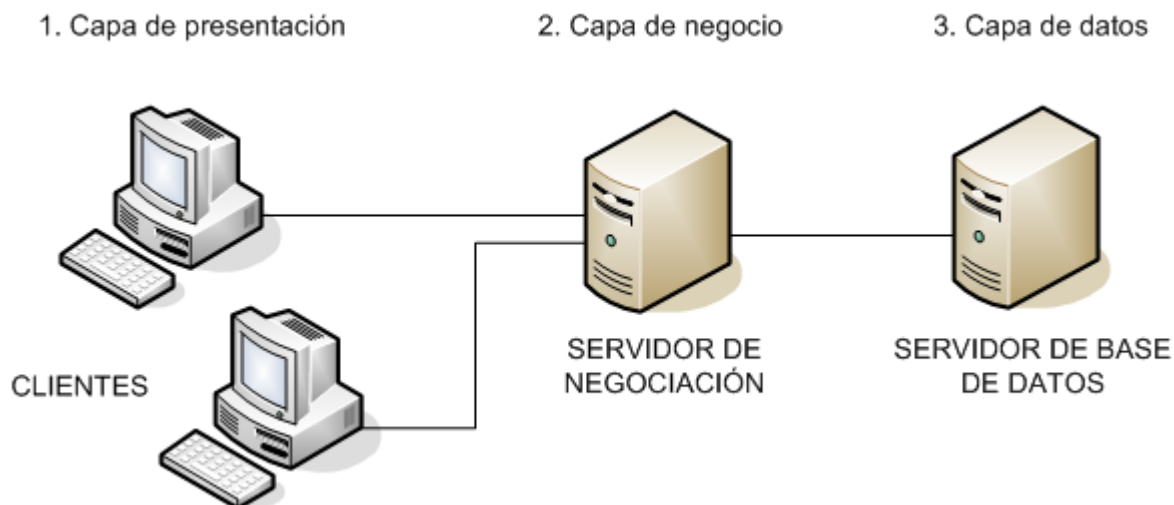


Figura 5 – Diagrama de una arquitectura Multinivel

La principal ventaja de esta arquitectura es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre todos los componentes. Además, permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API¹¹ que existe entre

¹¹ Application Programming Interface - Interfaz de Programación de Aplicaciones

niveles. A cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

Un diseño aplicable al desarrollo de SimuReD es el diseño en tres niveles (o en tres capas), el cual estaría conformado de la siguiente manera:

- Capa de presentación: es la que ve el usuario (también se la puede llamar capa de usuario), y es la que presenta el sistema al usuario, le muestra la información y captura los datos del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser entendible y fácil de usar para el usuario.
- Capa de negocio: es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio o capa de lógica del negocio, porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.
- Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio. En nuestro caso aquí se incluirían los generadores de números aleatorios y los generadores matemáticos de distribuciones, lo que denominaremos el núcleo o core del simulador.

Todas estas capas pueden ser alojadas en una única computadora, si bien lo más usual es que haya varias. Se puede separar a los clientes en varias terminales (ya sean PC u otros dispositivos) e instalar las capas de negocio y de datos en el mismo servidor, y si el crecimiento de las necesidades lo aconseja se pueden separar posteriormente en dos o más servidores.

En una arquitectura de tres niveles, los términos capas y niveles no significan lo mismo ni son similares. El término capa hace referencia a la forma como una solución es segmentada desde el punto de vista lógico, en cambio, el término nivel corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física. Por ejemplo:

- Una solución de tres capas (presentación, lógica, datos) que residen en una sola computadora (Presentación+lógica+datos). Se dice que la arquitectura de la solución es de tres capas y *un nivel*.
- Una solución de tres capas (presentación, lógica, datos) que residen en dos computadoras (presentación+lógica, lógica+datos). Se dice que la arquitectura de la solución es de tres capas y *dos niveles*.
- Una solución de tres capas (presentación, lógica, datos) que residen en tres computadoras (presentación, lógica, datos). La arquitectura que la define es: solución de tres capas y *tres niveles*.

Con lo anteriormente expuesto se puede ver que la implementación de una arquitectura Cliente-Servidor combinada con una arquitectura Multinivel es ideal para el desarrollo de

SimuReD, ya que por un lado, se podrá trabajar sobre un núcleo o core que resuelva todos los problemas matemáticos, que será independiente de la interfaz de usuario. Por otro lado, se desarrollarán diferentes interfaces de usuario sin afectar la lógica del sistema.

Herramientas de Desarrollo

Para el desarrollo de SimuReD se necesitarán herramientas para administración de datos (Base de Datos), para codificación (lenguaje de programación) y para el trabajo en equipo (herramienta de versionado o CVS¹²). A la hora de seleccionar las tecnologías se tuvo en cuenta la capacitación con la que cuentan los alumnos de la FRD, ya que si se seleccionan herramientas que son utilizadas en las distintas cátedras se evita la etapa de capacitación de los integrantes del equipo.

Base de Datos

Los datos que maneja un simulador son generados a partir de cálculos matemáticos, en este sentido la administración de datos está reducida a parámetros, los cuales pueden ser mantenidos a través de archivos de propiedades (técnicamente denominados archivo properties). Éstos son simples archivos secuenciales que guardan la información necesaria para inicializar el sistema. También pueden administrarse a través de archivos XML. Entonces podemos ver que para que el simulador funcione no tenemos necesidad de implementar un gestor de base de datos.

Analizando la gestión de la documentación generada, o sea, cómo se guardarán los modelos creados por los usuarios. Éstos modelos podrían ser administrados por el usuario y que él decida dónde alojar el archivo (ya sea en su propia PC o en algún servidor compartido). Pero existirá un repositorio de documentos, para favorecer la comunicación entre distintos usuarios y de esta manera promover la comunidad en torno a la aplicación (Web2.0). Éste repositorio deberá ser administrado y organizado para tener un fácil acceso a los documentos por parte de los usuarios.

Para esta administración, no se necesitará gran cantidad de procesamiento en el motor de base de datos, pero sí de una herramienta estable y que permita instalarse en un servidor para poder ser accesible por los distintos usuarios. Para este caso se recomienda el uso de un SGBD¹³ muy utilizado en proyectos Web como lo es MySQL¹⁴. Este motor de base de datos se caracteriza por ser muy potente, estar bajo licencia GNU y por demandar pocos recursos para su instalación¹⁵.

Una característica interesante de MySQL es que puede integrarse con una herramienta de versionado llamada Bonsai CVS¹⁶. De esta manera el motor de base de datos no sólo se utilizará en la etapa de pruebas y en el uso normal del sistema, sino que también podrá utilizarse en la etapa de desarrollo para administrar, compartir y versionar toda la documentación del proyecto.

¹² Concurrent Versioning System (Sistema de Versiones Concurrentes)

¹³ Sistema de Gestión de Base de Datos.

¹⁴ www.mysql.com

¹⁵ comparado con productos similares, como ser PostgreSQL y Oracle.

¹⁶ www.mozilla.org/projects/bonsai

Control de versiones

CVS es un sistema de control de versiones. A medida que avanzamos con el proyecto, se van generando versiones de los documentos y del código fuente, según vamos añadiendo mejoras, corrigiendo errores o completando los mismos. CVS nos permite guardar todas estas versiones de una forma cómoda, facilitándonos comparar los archivos de una versión con otra, volver a una versión anterior y actualizar los cambios.

Con CVS, cada vez que modificamos un archivo, se guarda el nombre de la persona que lo ha modificado, la fecha/hora y un comentario sobre la modificación. Podemos ver sobre un archivo concreto el historial de cambios a lo largo de todo el proyecto.

Para usar CVS se requiere instalar dos programas. Por un lado un servidor que es el que guardará todas las versiones de todos los archivos. Es al que se le envían los archivos cuando están listos y el que devuelve dichos archivos cuando son solicitados. Por otro lado, en cada estación de trabajo necesitamos un cliente de CVS. Este es un programa capaz de conectarse con el servidor y pedirle o enviarle archivos.

Bonsai CVS es un proyecto de mozilla.org¹⁷. Esta herramienta nos provee de un servicio de versionado para documentos y código fuente. Hay una comunidad activa detrás de este proyecto, la cual provee soporte técnico gratuito ante posibles inconvenientes. Este desarrollo funciona bajo el sistema operativo Linux.

Como alternativa y corriendo bajo sistemas operativos Microsoft tenemos cvsnt (www.cvsnt.com). Este sistema, además de funcionar bajo Windows tiene licencia GPL¹⁸. Desde allí se baja el servidor de CVS el cual se instalará en la maquina donde todos los integrantes del equipo se conectarán.

Entre las alternativas de clientes para esta herramienta tenemos varios IDEs, como Eclipse, los cuales pueden configurarse fácilmente y son compatibles. Por otro lado se tiene el cliente de cvsnt, el cual puede bajarse desde www.wincvs.org¹⁹.

Lenguajes de programación

En cuanto a los lenguajes de programación debemos analizar por separado las necesidades. En primer lugar se necesitará un lenguaje que permita el procesamiento de cálculos probabilísticos, de una manera sencilla, pero también exacta. También deberá ser flexible en la generación de las pantallas y en lo posible que pueda desarrollarse para funcionar en diferentes plataformas. Se planteó también, que era condicionante que el lenguaje adoptado debería ser conocido por los alumnos, de modo que debería estar incluido en la currícula de la carrera. Finalmente el lenguaje debe ser Orientado a Objetos, para conservar una tecnología moderna, acorde a las metodologías y arquitecturas anteriormente propuestas.

¹⁷ Comunidad de desarrolladores on-line creadores de Mozilla Firefox, uno de los navegadores líderes a nivel mundial

¹⁸ GPL: GNU General Public License

¹⁹ Consultar manual de instalación en www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=cvsnt

Ante estas circunstancias se cree en la necesidad de adoptar distintos lenguajes de programación, uno para cada capa de la aplicación. A continuación se ofrece un breve análisis de los lenguajes R, Java, HTML, JavaScript y ActionScript.

Generación de Valores Estocásticos

R es un lenguaje estadístico muy extendido en el ámbito universitario debido a su alta calidad y libre distribución. El lenguaje R es orientado a objetos e incorpora gran número de rutinas de cálculo estadístico, en este sentido es el más completo de los que se presentan en el mercado. R es GNU y fue desarrollado a partir de S, un lenguaje similar pero bajo licencia.

Programación de la Capa de Negocio

Java es un lenguaje moderno y muy utilizado en el ambiente laboral por su característica multiplataforma. Además de ser de uso gratuito, es uno de los lenguajes más utilizados hoy en día. En la FRD hay un taller de Java que se dicta en el segundo cuatrimestre, por lo que los alumnos ya cuentan con la capacitación necesaria a partir del segundo nivel de cursada.

La forma más usual de ejecutar aplicaciones Cliente-Servidor con Java es mediante la implementación de Servlets. Los Servlets son objetos gestionados y ejecutados dentro de un contenedor de servlets el cual debe estar instalado en el servidor. El contenedor más utilizado es Apache Tomcat, del cual se dará una descripción mas adelante en este capítulo.

Los Servlets se ejecutan en el servidor y se encargan de ser un nexo entre la información interna del sistema y el usuario. Resuelven la lógica, seguridad y comunicación del sistema en el servidor entregando a la interfaz de usuario la información necesaria ya procesada.

Una característica de los servlets es que no poseen interfaz gráfica, ya que el objetivo de los mismos no es realizar todas las tareas ocultas del sistema.

Otro lenguaje que es enseñado en la FRD es C++, el cual tiene todas las características necesarias para SimuReD. Sin embargo este lenguaje fue superado por C# el cual está incorporado dentro del paquete .NET de Microsoft.

En la cátedra de paradigmas de programación, se dedica un cuatrimestre al desarrollo con SmallTalk. Si bien SmallTalk tiene varias versiones, hay algunas que son libres y permiten la generación de archivos ejecutables.

GNU SmallTalk es una implementación libre desarrollada a partir de SmallTalk-80. Por otro lado tenemos Squeak que cuenta con las capacidades y herramientas típicas de un sistema SmallTalk y se encuentra entre los sistemas de SmallTalk que se pueden ejecutar prácticamente en casi todas las plataformas de hardware. Squeak se puede utilizar con una implementación de Morphic, un Framework de manipulación directa de objetos basado en Self. Hasta la fecha es la herramienta preferida por los principales desarrolladores de SmallTalk.

Interfaz de Usuario

El desarrollo de la capa de usuario o interfaz de usuario, debe adaptarse a las múltiples terminales a ser utilizadas por los usuarios, tales como PC de escritorio con variados sistemas operativos, teléfonos celulares, SmartPhones y cualquier dispositivo nuevo que se incorpore al mercado. Para ello, el desarrollo de la interfaz debe realizarse bajo una tecnología que soporte dichos cambios.

HTML y JavaScript son lenguajes, del tipo interpretados, que pueden ser corridos en cualquier explorador Web, lo que los convierte en excelentes creadores de interfaces multiusuarios. La mayoría de los dispositivos actuales tiene un software incorporado que soporta estos lenguajes.

HTML es un lenguaje basado en tags y por su naturaleza es un lenguaje estático. Esto quiere decir que todo lo que hace HTML es dar formato y ubicar diferentes tipos de objetos gráficos por la pantalla. A raíz de la creciente demanda y la evolución de Internet, este lenguaje se convirtió en el eje principal de los desarrollos web y ha tenido una evolución muy favorable. Actualmente se denomina a DHTML al perfil dinámico del lenguaje, aunque la estructura esencial es la misma.

La potencialidad de HTML está dada por el uso de varias tecnologías, que implementadas en conjunto crean una herramienta muy versátil y excelente, tanto en prestaciones como en eficiencia. Una tecnología que le brinda cierto dinamismo a HTML es JavaScript.

Con JavaScript se agrega funcionalidad a la pantalla creada con HTML. En un principio se utilizaba sólo para realizar cierto tipo de validaciones, pero la propia evolución hizo de este hijo de C, una herramienta tan poderosa como sorprendente.

A un nivel avanzado la tecnología AJAX es posible gracias a la evolución de JavaScript. AJAX permite realizar cambios en la pantalla (reescribiendo el código HTML) sin necesidad de recargar la misma completamente. Esta acción marcó un cambio radical en Internet, ya que la sensación percibida con AJAX es similar a una aplicación de escritorio, dejando de lado los grandes vacíos que se generaban en las páginas web cuando se realizaba una consulta al servidor.

En Internet pueden conseguirse cuantiosas librerías de uso libre, que resuelven muchas funcionalidades posibles de hacer con JavaScript.

Otro avance importante es el uso de los archivos de estilo CSS. Estos archivos, confinan en un solo lugar el formato definido de todo un sitio. A primeras luces, podría parecer que esta tecnología no aporta demasiado a la flexibilidad. Pero visto desde una perspectiva técnica, el uso de CSS posibilita realizar modificaciones o cambios radicales en el sitio web tan fácil como rápido.

La generación de páginas en forma dinámica es otro de los avances importantes que mencionar. Consiste en la generación del código HTML, con todos los adicionales necesarios para mostrarse en el cliente, en el servidor de la aplicación. Para esto se utiliza la tecnología denominada ASP (licenciada por Microsoft) y JSP (propiedad de SUN Microsystem). Como se mencionó anteriormente, el lenguaje óptimo para SimuReD es Java, por lo que, para adaptar esta tecnología se opta por JSP.

La tecnología JSP consiste en incorporar código de Java en el código HTML de la página a enviar desde el servidor al usuario. Cuando el usuario realiza la solicitud al servidor, primero se ejecuta el código Java embebido en el archivo HTML y después se envía dicho archivo (sólo con HTML y los Scripts correspondientes). Mediante la lógica incorporada con Java y con el correspondiente acceso a la memoria del servidor (y a los objetos que viven en él) pueden realizarse llenado de grillas con datos obtenidos de una consulta en la base de datos, o llenar ciertos combos que se mostrarán en la aplicación. También se puede incorporar funciones de JavaScript dependiendo de cierta lógica o cambiar la funcionalidad de botones de acción.

La potencialidad que se obtiene al trabajar con JSPs es excelente, además es invisible al usuario. También permite la reutilización de pantallas, ya que el mismo código HTML sirve para mostrar distintos datos, ya sean los datos de todos los clientes, o los datos de todos los productos de una Empresa.

En el caso de SimuReD, la generación dinámica de HTML permite la comunicación con el usuario en forma dinámica e interactiva y minimiza el trabajo por parte del equipo al favorecer la reutilización de componentes.

Gráficos y Animación

Una parte importante de un simulador es la capacidad de presentar la información resultante del proceso en forma clara y sencilla, con elementos de análisis tan diversos como gráficos de barra o indicadores tipo reloj. Llega a ser tan importante que puede marcar el éxito o fracaso del proyecto.

Si bien la animación no es esencial para mostrar el resultado de la simulación, da un elemento de análisis adicional muy codiciado. La creación del modelo en forma gráfica utilizando el método “arrastrar y colocar” (Drag & Drop) también genera un valor agregado y hace mas intuitivo el uso del sistema.

Hay disponibles dos herramientas factibles de utilizar con SimuReD, una de ellas es el uso de Applets y otro es a través del uso de Flash.

Los Applets son componentes de Java que permiten enviar un programa Java al cliente embebido en la página HTML. Este componente funciona dentro del navegador del cliente en forma independiente. Independientemente del sistema operativo, equipo y tipo de conexión que tenga el usuario el Applet siempre se ejecutará de la misma manera en todas las computadoras. De esta manera se garantiza que los usuarios podrán ejecutar las animaciones independientemente del hardware y software del que dispongan.

Con muy poco esfuerzo, un Applet puede convertirse en un Midlet. Este componente es similar al anterior pero se ejecuta en dispositivos móviles.

Así, diseñando la animación con Java (mediante la utilización de Applets y Midlets) se obtiene una utilidad multiplataforma que además facilita la comunicación entre la capa de negocio y la de usuario, al unificar el lenguaje.

El uso de Java para realizar las animaciones parece ser una muy buena opción, pero la capacidad de programar con los paquetes de AWT o SWING es determinante para el éxito del

mismo. Estas dos librerías son las encargadas de administrar los gráficos en Java y son bases para utilizar mecanismos de animación como Sprites²⁰ y los Frames.

La alternativa que evita este problema es el uso de Flash para realizar las animaciones.

Actualmente mas del 50% de las animaciones que se encuentran en Internet están realizadas con Flash. Macromedia Flash es una herramienta de diseño bajo licencia de Adobe.

La popularidad de esta herramienta fue posible gracias a la facilidad de uso y a la calidad de las animaciones obtenidas. Sumando la flexibilidad que se obtiene a través de la implementación de fragmentos de código para agregar lógica utilizando ActionScript. ActionScript está basado en JavaScript, con la gran diferencia que ActionScript es Orientado a Objetos.

La arquitectura de funcionamiento es igual a los Applets de Java, se descarga un archivo desde el servidor que se ejecuta en el cliente a través de un browser o navegador. También es soportado en diferentes dispositivos móviles, como teléfonos celulares y smart-phones. Y como opción adicional, existe un reproductor Flash que puede ser descargado gratuitamente, para reproducir los archivos en forma independiente.

La tecnología Flash utiliza fotogramas²¹ para realizar la animación. Soporta graficos vectoriales, sonidos y video que pueden ser configurados de manera bidireccional, esto significa que desde el equipo del usuario se puede enviar al servidor Streams²² de audio y video.

La versatilidad es tal que se utiliza para realizar desde simples anuncios en páginas web, hasta Aplicaciones Ricas de Internet (RIA – Rich Internet Applications). Este último tipo de aplicaciones consiste en descargar toda la aplicación en el equipo del usuario de modo que se evite la alta carga de tráfico entre el cliente y el servidor. Sólo se realizan consultas al servidor cuando es absolutamente necesario, por ejemplo para consultas en archivos o base de datos.

Con Flash se pueden crear varios formatos de archivos de salida, entre ellos, archivos ejecutables. De los archivos compilados puede obtenerse el código fuente y los elementos muy fácilmente mediante descompiladotes. Esto podría causar un grave problema en la seguridad de la aplicación, ya que podría incorporarse código malicioso dentro de la misma y redistribuirse. Por este motivo, es importante utilizar la ofuscación²³ de los archivos, para evitar este tipo de problemas.

²⁰ Imagen alojada en la memoria con una secuencia completa del dibujo animado. La técnica consiste en utilizar este componente como un objeto independiente a los movimientos que realice en la pantalla. Por ejemplo en un video juego, se utiliza un Sprite para mostrar la animación del personaje, independientemente del movimiento que tenga en la pantalla.

²¹ Un fotograma es una imagen. Al cambiar los distintos fotogramas, a una velocidad adecuada, se obtiene el movimiento de las diferencias entre las imágenes. Esta técnica es básica en la creación de dibujos animados y en el cine.

²² Cadena de información que se puede ejecutar mientras se va descargando.

²³ se refiere a encubrir el significado de una comunicación haciéndola más confusa y complicada de interpretar.

Web Services

Un problema que se presenta al trabajar con tecnologías distintas entre si, es la comunicación entre las capas y aplicaciones. ¿Como un servlet, alojado en el servidor y que entiende sólo Java, puede enviarle una orden a un componente gráfico que sólo entiende Flash y el cual se encuentra embebido en HTML? La respuesta a este intrincado interrogante es muy sencilla... Web Services.

Los Web Services o Servicios Web, proveen un estándar de comunicación entre los distintos elementos que componen una aplicación de este tipo. La comunicación se basa en archivos XML que son los puntos de acceso entre un módulo y otro. La lectura y escritura de los archivos XML debe estar sincronizada.

Cada componente debe implementar la lectura/escritura de los objetos que están en la memoria, de manera tal que otro componente de la aplicación, desarrollado bajo otro tipo de tecnología, pueda interpretar este archivo y replicar dichos objetos para su posterior uso y procesamiento de los mismos. A este mecanismo se lo denomina interoperatividad.

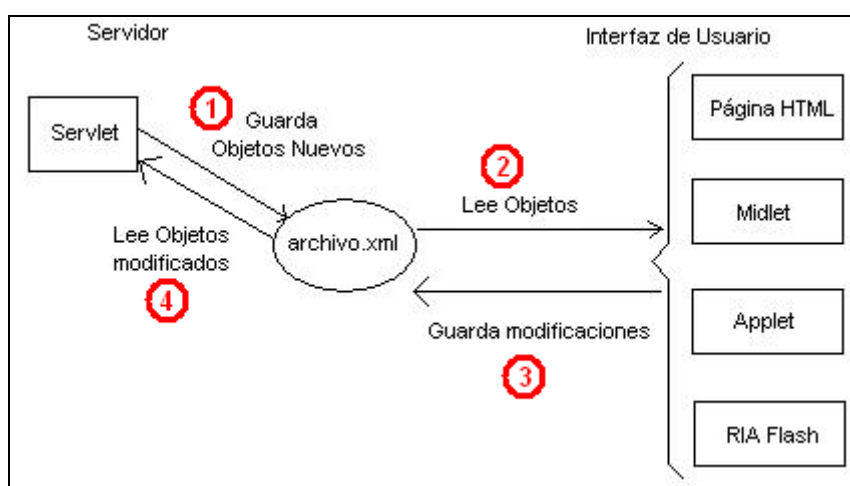


Figura 6 – Circuito básico de comunicación utilizando XML

La interoperatividad se consigue gracias a la adopción de estándares abiertos. Una definición muy buena se expresa en wikipedia.org:

“Estándar abierto es una especificación disponible públicamente para lograr una tarea específica. La especificación debe haber sido desarrollada en proceso abierto a toda la industria y también debe garantizar que cualquiera la puede usar sin necesidad de pagar regalías o rendir condiciones a ningún otro. Al permitir a todos el obtener e implementar el estándar, pueden incrementar y permitir la compatibilidad e interoperabilidad entre distintos componentes de hardware y software, ya que cualquiera con el conocimiento técnico necesario y recursos puede construir productos que trabajen con los de otros vendedores, los cuales comparten en su diseño base el estándar.”

Los estándares abiertos tienden a generar un mercado libre y muy dinámico, porque al no haber restricciones en su uso lo común es que sobre unos estándares abiertos se edifiquen otros y así sucesivamente. Es el caso de los estándares más comunes de Internet y por tanto de la Internet misma.

Estos estándares se implementan a través de protocolos. Existen varios, pero en el caso de SimuReD se debe optar por SOAP (Simple Object Access Protocol), ya que es el protocolo altamente recomendado por expertos para comunicar las tecnologías aquí descriptas.

Requerimientos Operativos

Las PCs utilizadas por los miembros del equipo deberán tener instalada la aplicación cliente del CVS, que deberá ser descargada del sitio nombrado anteriormente. También deberán tener instalado un paquete de herramientas ofimáticas para la redacción de los documentos (Casos de Uso, Diagramas, Casos de prueba, etc.), que podría ser el paquete de Microsoft (MS-Office), el paquete de OpenOffice.org o el paquete que ofrece Google (Google Docs). Además se tendrá que tener instalada la herramienta de desarrollo seleccionada (Eclipse + Java o Squeak).

Los requerimientos de hardware mínimos para instalar este software son:

- Sistema de archivos NTFS.
- Procesador:
 - § 1000 Mhz. (limitado por Squeak).
- Memoria:
 - § 256 Mb (limitado por Java).
- Espacio en Disco Rígido:
 - § 250 Mb para MS-Office.
 - § 125 Mb para Eclipse.
 - § 50 Mb para Java.
 - § 45 Mb para Squeak.

Debe haber una PC que funcione como servidor, que debe tener instalado el servidor CVS y Tomcat (en caso de utilizar Java). Aquí también deberá instalarse MySQL. Esta máquina puede ser una de las utilizadas por algún miembro del equipo, pero deberá tener instalado este software específico y estar disponibles para el resto del equipo.

Los requerimientos mínimos del hardware son:

- Sistema de archivos NTFS.
- Procesador:
 - § 1000 Mhz.
- Memoria:
 - § 256 Mb (limitado por Java).
- Espacio en Disco Rígido:
 - § 50 Mb para Java.
 - § 45 Mb para Squeak.
 - § 50 Mb para Tomcat.

Conclusiones

Con respecto a la metodología a adoptar para el desarrollo de SimuReD se cree que la mejor alternativa es RUP dado que, por un lado tenemos el respaldo de todo un sistema de documentación que optimiza el valor horas/hombre y por el otro, tenemos como valor agregado la capacitación que obtendrán los becados en el uso de una herramienta nueva muy utilizada en el ambiente laboral.

En cuanto a la elección de la arquitectura de software, se dio prioridad a una que permita el intercambio de información entre los usuarios, por lo que la arquitectura Cliente-Servidor es la ideal. Desde el punto de vista del desarrollo había que considerar que se deberían desarrollar etapas o fases bien distintivas, para evitar que el cambio de equipo alargara los tiempos de desarrollo. Aquí entra en juego la arquitectura multinivel, la cual nos permite trabajar en cada una de las capas independientemente de las otras, pudiendo así enfocarse en el núcleo, en las comunicaciones o en la interfaz de usuario como si fueran aplicaciones diferentes.

La implementación de un sistema CVS para el control de las versiones de todos los documentos del proyecto se vuelve imprescindible, dado que se espera una rotación en los integrantes del equipo. La instalación del servidor CVS puede realizarse en una de las PCs destinadas al proyecto, debiéndose tener la precaución de realizar los backups correspondientes. Para el desarrollo, y basado en el análisis de los lenguajes, podemos ver que Squeak es una de las opciones que mas ventajas tiene, ya que los alumnos tienen la capacitación necesaria para utilizarlo. No hay que descartar la posibilidad de utilizar Java, ya que la única imposición que tiene es en la instalación de un servidor dedicado.

Bibliografía

Ingeniería de Sistemas de Software, Gonzalo León Serrano (Ed. Isdefe - 1996)
An introduction to Software Architecture, David Garlan y Mary Shaw ()
UML y Patrones, Introducción al análisis y diseño orientado a objetos, Craig Larman (Prentice Hall - 1999).
www.adictosaltrabajo.com
www.desarrolloweb.com
www.programacion.com
www.wikipedia.org

Capítulo 4 – Localización

INTRODUCCIÓN	61
POSIBLES UBICACIONES	61
MACROLOCALIZACIÓN	62
MICROLOCALIZACIÓN	64
CONCLUSIÓN	65

Introducción

Definir la localización de un proyecto es un factor muy importante, ya que puede determinar el éxito o fracaso del mismo. Una elección errónea trae problemas como la dificultad para contratar personal idóneo o elevados costos de transporte en la obtención de la materia prima.

La localización de un proyecto de software, como lo es SimuReD, dependerá de varios factores que serán analizados en este capítulo. A través de un exhaustivo análisis de las fuerzas locacionales¹ se obtendrá una posible ubicación para desarrollar el proyecto.

Posibles ubicaciones

La localización por defecto es la Facultad Regional Delta de la UTN, ubicada en Campana, provincia de Buenos Aires. Esta locación es preferencial por ser cuna de este proyecto, de hacerse en otra ubicación el sentido del nombre del producto deberá cambiar de SimuReD (Simulador de la Regional Delta) a SimuRed (Simulador basado en Redes). Obviamente este no es un factor determinante, ya que los objetivos del proyecto no cambian y al momento de tomar una decisión se deberá velar por la concreción del proyecto y no dejarse llevar por un factor emocional.

En la Facultad Regional Delta, contamos con el apoyo incondicional de docentes con una amplia experiencia en el modelado, simulación e investigación operativa, como así también con las herramientas y la infraestructura necesaria para el desarrollo.

Una alternativa de localización es la Facultad Regional Santa Fe, donde se encontró un gran interés en los docentes de dicha facultad cuando se realizó la encuesta del estudio de mercado (Capítulo 2). Allí fueron 5 los docentes que nos solicitaron mayor información de SimuReD, la cual se envía asiduamente informando los avances del presente proyecto.

En la Facultad Regional Santa Fe, la cátedra de Simulación se dicta en 3 turnos (matutino, vespertino y nocturno), contando con el mayor staff de docentes de Simulación en todas las Regionales de la UTN, esto es una ventaja alentadora sobre las otras regionales.

Para tomar una decisión acertada dentro de estas dos alternativas se evalúan ambas mediante el Método Cualitativo de Locación, el cual consiste en definir los principales factores determinantes de una localización, para asignarle valores ponderados de peso relativo, de acuerdo con la importancia que se les atribuye. De esta manera definimos la macrolocalización.

¹ Los factores o fuerzas locacionales son las características de las alternativas de ubicación, que permiten establecer la diferenciación entre ellas.

Macrolocalización

Los factores tomados en cuenta para evaluar y tomar la decisión correcta fueron seleccionados luego de una ardua tarea de análisis, teniendo la precaución de ser imparciales sobre los factores para no favorecer ninguna de las dos alternativas y velando por la factibilidad del proyecto (Equilibrio Parcial).

A continuación se enumeran los factores evaluados y se da una breve descripción del enfoque o perspectiva que representa cada uno de ellos.

- **Respaldo de la administración:** Se evalúa el interés demostrado por la administración de la Regional. En el caso de la FRD se tiene el apoyo del Director del Departamento de Sistemas, mientras que en la FRSF no se obtuvo respuesta de alguno de los representantes de las distintas direcciones, pero si se obtuvo el apoyo de la jefatura de la cátedra.
- **Apoyo tecnológico:** El apoyo tecnológico refiere al soporte adicional al grupo de desarrollo ante posibles contingencias. Tanto la FRD como la FRSF cuentan con el soporte técnico adecuado.
- **Agilidad burocrática:** Evalúa la flexibilidad con la que pueden conseguirse los recursos necesarios para el desarrollo de SimuReD.
- **Cantidad de proyectos** de desarrollo de software llevados a cabo dentro de la Regional.
- **Apoyo profesional** por parte de los docentes
- **Nivel académico para el desarrollo de software:** Se evalúa la capacitación ofrecida para el desarrollo de software. Aquí se analizan los talleres adicionales a la currícula que poseen las Facultades Regionales como así también la orientación práctica que tienen las cátedras orientadas al desarrollo de software (Análisis de sistemas, Paradigmas de Programación, Sintaxis y semántica del lenguaje, etc.).
- **Interés empresarial:** Evalúa el posible interés de las empresas de la zona en este proyecto. Para ello se evalúan los usos actuales a nivel empresarial de los simuladores comerciales y se pondera en base a la cantidad de industrias en la zona con esas características.
- **Interés Gubernamental:** Evalúa en que grado podría impactar el desarrollo de éste simulador a nivel gubernamental, ya sea municipal o provincial.
- **Impacto en los medios:** Evalúa en que grado podría impactar el desarrollo de éste simulador en los medios de prensa local, zonal y provincial.
- **Condiciones sociales y culturales:** para obtener un perfil de la sinergia posible de lograr en el equipo.
- **Mano de obra disponible:** Analiza el grado de ocupación de los estudiantes. La facultad que posee mayor cantidad de alumnos con trabajo es la que menor ponderación tiene.
- **Disponibilidad de becas:** Se consultó con la secretaría a cargo de la administración de becas la posibilidad de conseguir la asignación de recursos a este proyecto. Se ponderó la misma con 0 (en el caso de no disponer del recurso), 10 (en caso de tener la seguridad de disponibilidad).

- **Infraestructura disponible:** Evalúa la posibilidad de conseguir todos los recursos necesarios al servicio del desarrollo de este proyecto, tales como lugar físico, mobiliario, equipamiento, etc.
- **Medios de comunicación:** disponibilidad de cuentas de e-mail, teléfono y acceso a publicar informes en la intranet.

FACTOR RELEVANTE	PESO	FRD		FRSF	
		CALIF.	CALIF. POND	CALIF.	CALIF. POND
INSTITUCIONALES					
RESPALDO DE LA ADMINISTRACIÓN	0.15	10	0.150	8	0.120
APOYO TECNOLÓGICO	0.05	7	0.035	7	0.035
AGILIDAD BUROCRÁTICA	0.05	5	0.025	5	0.025
CANTIDAD DE PROYECTOS	0.03	6	0.018	9	0.027
APOYO PROFESIONAL	0.08	10	0.080	10	0.080
NIVEL ACADÉMICO PARA EL DESARROLLO DE SOFTWARE	0.09	9	0.081	9	0.081
SOCIALES					
INTERES EMPRESARIAL	0.08	10	0.080	7	0.056
INTERES GUBERNAMENTAL	0.06	4	0.024	6	0.036
IMPACTO EN LOS MEDIOS	0.06	5	0.030	8	0.048
CONDICIONES SOCIALES Y CULTURALES	0.05	8	0.040	9	0.045
ECONÓMICOS					
MANO DE OBRA DISPONIBLE	0.10	9	0.090	9	0.090
DISPONIBILIDAD DE BECAS	0.05	10	0.050	8	0.040
INFRAESTRUCTURA DISPONIBLE	0.10	10	0.100	8	0.080
MEDIOS DE COMUNICACION	0.05	6	0.030	6	0.030
SUMA	1.00		0.833		0.793

Tabla 1 – Ponderación para la Macrolocalización.

Como resultado podemos observar la pequeña diferencia encontrada entre ambas opciones. La Facultad Regional Delta se muestra más favorable ante la Facultad Regional Santa Fe.

Los puntos en los cuales la FRD se destaca para la localización de este proyecto son en el apoyo que se tiene de la administración, tanto de la cátedra Proyecto, como del departamento de Sistemas para la concreción de SimuReD. Este punto facilita la obtención de las becas necesarias para el reclutamiento de los integrantes del equipo de trabajo.

Otro punto fuerte que encuentra a Campana con diferencia favorable sobre Santa Fe es la perspectiva de crecimiento industrial de la zona. Si bien este proyecto está orientado al uso académico exclusivo para todas las regionales de la UTN, estar inmerso en un ambiente de desarrollo industrial puede atraer el interés de diferentes empresas de la zona. Estas empresas podrían solicitar servicio de capacitación en simulación estratégica para su negocio, obteniendo la Regional un beneficio extra.

Si bien los medios de comunicación de Campana aún no tienen un desarrollo acorde al crecimiento de la zona, la ciudad tiene influencia en los medios de Capital Federal. Este es otro punto interesante, ya que si se llega a un medio de Capital Federal el alcance de la noticia es a nivel nacional. Dado que esto es más difícil de alcanzar, la ponderación favorece a la Regional Santa Fe.

En la FRD hay un convenio con Tenaris-Siderca por el cual esta empresa se compromete a donar a la Regional todo el material tecnológico en desuso. Por tal motivo, el equipamiento tecnológico de la Regional Delta es el mejor de todas las regionales del País.

Por último, y contribuyendo a favor de la localización en la Ciudad de Campana, al momento de cerrar el presente análisis, se concretó el plan denominado Campana Ciudad Inteligente², mediante el cual se realizará una red inalámbrica en la ciudad con posibilidad de acceso gratuito para todos los habitantes de la ciudad. Este proyecto ubica a Campana entre las ciudades con mayor desarrollo tecnológico a nivel nacional. Este plan demuestra el compromiso de las autoridades municipales y académicas por el desarrollo de proyectos tecnológicos dentro del distrito, favoreciendo en gran medida el desarrollo de SimuReD dentro de esta jurisdicción.

Microlocalización

El desarrollo de SimuRed se llevará a cabo en las instalaciones de la Facultad Regional Delta. Este edificio se ubica a 100 metros del arco de entrada a la ciudad de Campana. La cercanía con la ruta Nacional Nro 9 (ruta Panamericana) hace que el acceso al edificio sea excelente.

El edificio cuenta con aproximadamente 2600 mts² cubiertos, distribuidos en dos plantas. Esta superficie contempla aulas, oficinas, laboratorios, patios internos, baños, dependencias auxiliares y el aula magna (con capacidad para 200 personas).

Se encuentra en etapa de licitación y construcción la ampliación de la Regional. Esta ampliación está estipulada para desarrollarse durante los años venideros hasta el 2011.

El espacio físico necesario para el desarrollo de este proyecto se reduce a un máximo de 2 computadoras de escritorio con acceso a Internet. Consultando con el responsable de los laboratorios de Sistemas (Pedro Assis) y con el Jefe del departamento de Sistemas (Luis Perna), se consiguió ubicación en dos lugares alternativos a definir al momento de iniciar con el desarrollo.

Por un lado se encuentra el laboratorio denominado PC4, donde es posible su uso previa solicitud formal en una planilla. Esta solicitud puede extenderse a un día y un horario fijo todas las semanas. Dado que el proyecto será desarrollado con alumnos de esta Regional, los horarios a solicitar el laboratorio son fuera del horario normal de cursada (de 18:30 hs. a 23:15 hs.), por lo que se garantiza que el mismo se encontrará libre en al menos un horario

² Para más detalles del plan remitirse al anexo 4D – Campana Ciudad Inteligente

semanal. En este caso las PCs a utilizar son públicas y se encuentran freezadas³. Esto no interfiere en el desarrollo, ya que toda la documentación generada será resguardada en los respectivos servidores⁴.

Por otro lado, se encuentra disponible el laboratorio de Hardware, donde se compartiría el espacio físico con el equipo de trabajo que desarrolla sus actividades allí. En este caso, se cuenta con una computadora asignada para el proyecto y la cual será utilizada sólo por los miembros del equipo de SimuReD.

La selección adecuada de uno u otro caso dependerá, en gran medida, de la disponibilidad horaria de los integrantes del equipo. Si los integrantes del equipo trabajan todos juntos en un mismo horario es aconsejable la primer opción, en cambio, si los horarios son dispares se aconseja la segunda opción.

Ambas oficinas (PC4 y Laboratorio de Hardware) se hallan ubicadas en el segundo piso de la Regional, tal como se indica en la figura.

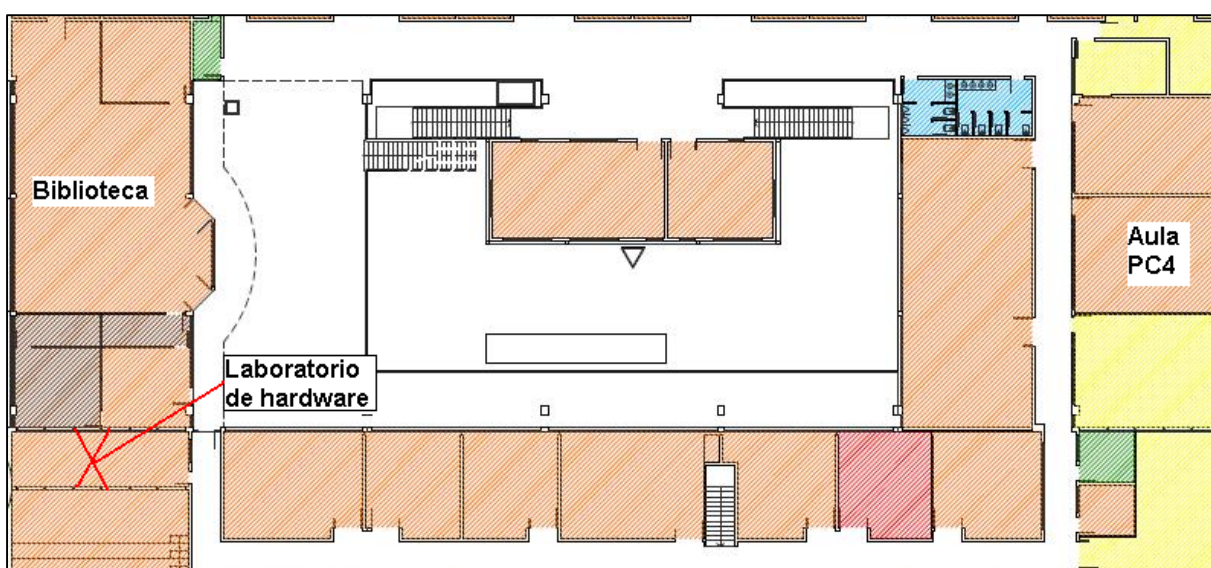


Figura 1 – Ubicación en Planta alta

Conclusión

Como resultado del análisis de macrolocalización puede observarse que la FRD es la mejor opción para el desarrollo de SimuReD. No debe descartarse la opción de Santa Fe, ya que la diferencia es muy pequeña y ante posibles complicaciones con la implementación en la Facultad Regional Delta en Campana se deberá optar por esta alternativa.

³ Una computadora Freezada tiene la capacidad de iniciar el sistema operativo en idénticas condiciones cada vez que se enciende la máquina (esto incluye el sistema de archivos). Por este motivo, cualquier documento guardado será eliminado cuando la PC sea apagada.

⁴ Tal lo descrito en el capítulo 3 - Tecnologías

Campana cuenta con un crecimiento mayor que cualquier otro centro industrial del País, aprovechando este crecimiento SimuReD tiene mayores posibilidades de difusión, ya que la zona comienza a tener eco en los medios de Capital Federal.

En la Regional Delta, se encontró un gran interés por este proyecto desde la jefatura del departamento de Sistemas, brindando todo su apoyo para la concreción de este proyecto.

Desde el departamento de Sistemas, se consiguió el lugar para instalar el equipo de trabajo, así como el uso de las comodidades necesarias (muebles, computadoras). El lugar mas aconsejable es el laboratorio de hardware que cuenta con instalaciones reservadas ideal para reuniones con el equipo de trabajo.

Capítulo 5 – Inversión

INTRODUCCIÓN	69
INVERSIONES PREVIAS A LA PUESTA EN MARCHA	70
INVERSIONES DURANTE EL DESARROLLO	70
CONCLUSIÓN	71

Introducción

El objetivo general del análisis de inversión es el descripto por la cátedra de Proyecto en una presentación disertada por el Ing. Walter Rodriguez Esquivel, dice:

“Analizar cómo la información que proveen los estudios de mercado, técnico y organizacional para definir la cuantía de las inversiones de un proyecto, debe sistematizarse, a fin de ser incorporada como un antecedente más en la proyección del flujo de caja que posibilite su posterior evaluación”

El desarrollo de SimuReD implica el uso de recursos, los cuales deben ser debidamente identificados para evitar problemas a lo largo de la implementación del proyecto. La inversión son los recursos económicos necesarios para realizar el proyecto y establecer un indicador estándar en la toma de decisiones.

Cuando se determina la inversión necesaria para el proyecto se tiene que tener cuidado de no caer en la subvaluación o en la sobrevaluación ya que ambos extremos no representarán la inversión real del proyecto y traerá problemas económicos o financieros a lo largo de la implementación.

La inversión en SimuReD está destinada a promover el desarrollo económico-social que es una definición más amplia que la de crecimiento económico; el desarrollo además del enfoque cuantitativo, requiere también del cualitativo. El desarrollo no solo implica mejorar el nivel de vida, sino también una mejor educación, un ambiente más limpio, igualdad de oportunidades, y una vida cultural más rica. En este marco, la inversión, es la base del desarrollo sin la cual no es posible el progreso, y es vital para que este sea sustentable.

En cuanto a la clasificación de las inversiones existen muchos criterios. En lo que respecta al desarrollo de SimuReD se puede encuadrarlo dentro de las siguientes clasificaciones:

- De acuerdo a la función es una inversión de modernización o innovación, ya que se trata de un nuevo producto que mejora los productos existentes.
- De acuerdo al tiempo de la inversión se trata de una inversión a largo plazo, si consideramos que el tiempo de desarrollo durará 2 años calendarios¹.
- Con respecto a la relación que mantiene con las demás inversiones de la UTN es una inversión complementaria, ya que facilita el dictado de la cátedra, lo cual optimiza la inversión en docencia.
- De acuerdo al origen del capital, SimuReD es una inversión pública, ya que los fondos provienen del estado.

¹ Si se desarrollara bajo condiciones normales de contratación (jornada de 8hs de lunes a viernes) es un proyecto que se concretaría en poco más de 5 meses con 3 personas, lo cual lo encuadraría en una inversión a corto plazo.

La inversión requerida para concretar la realización del proyecto se divide en dos partes, la inversión previa a la puesta en marcha, y la inversión durante la operación.

Inversiones Previas a la puesta en marcha

La inversión previa a la puesta en marcha consiste en todas las inversiones necesarias para comenzar con las actividades normales de desarrollo. Las inversiones previas a la puesta en marcha se pueden agrupar en:

- activos fijos
- activos intangibles
- capital de trabajo

Las inversiones en activos fijos son todas aquellas que se realizan en los bienes tangibles que se utilizarán en el proceso de transformación de los insumos o que sirvan de apoyo a la operación normal del proyecto. Ejemplo: equipamientos, oficinas, terrenos, infraestructura de servicios.

Las inversiones en activos intangibles son aquellas que se realizan sobre activos constituidos por los servicios o derechos para la puesta en marcha del proyecto. Incluyen:

- Gastos de Organización (gastos de administración, patentes, licencias)
- Gastos de Puesta en Marcha (gastos realizados antes del inicio de operación)
- Gastos de Capacitación

Las inversiones en capital de trabajo están formadas por el conjunto de recursos necesarios en la forma de activos corrientes para la operación normal del proyecto durante un ciclo productivo², para una capacidad y tamaño determinados. Se considera como una inversión a largo plazo, ya que forma parte del monto permanente de los activos corrientes necesarios para asegurar la operación del proyecto.

Inversiones durante el desarrollo

Las inversiones durante el desarrollo son aquellas necesarias para el reemplazo de activos o ampliación. Comprende los bienes necesarios para etapas de expansión posteriores, como la compra de vehículos, equipamiento, mobiliario o licencias.

Para el desarrollo de SimuReD no se prevén inversiones durante el desarrollo, pero se recomienda el análisis de futuras etapas a partir de este proyecto, ya que se invertirá tiempo en dejar toda la documentación necesaria para continuar expandiendo SimuReD y agregando funcionalidades nuevas, que puedan llegar a posicionarlo dentro de los primeros softwares de simulación educativos de habla hispana. Pero para llegar se necesitarán inversiones futuras que deben ser propuestas bajo el planteo de nuevas estrategias.

² Se denomina ciclo productivo al proceso que se inicia con el primer desembolso para cancelar los insumos de la operación y termina cuando se venden los insumos, transformados en productos terminados, se percibe el producto de la venta y queda disponible para cancelar nuevos insumos.

Conclusión

El desarrollo de SimuReD requiere de inversiones por parte de la Facultad Regional Delta que no implican consumo de recursos clave, ni tampoco implican un costo de oportunidad importante para el desarrollo de otras áreas de la mencionada regional.

Para la puesta en marcha del proyecto, se utilizarán recursos tangibles que ya pertenecen al activo de la facultad, sin que esto origine una inversión extra. Tales recursos pueden consultarse en el capítulo 6 – Costos.

Los activos intangibles necesarios, tampoco sustituyen una inversión, dado que los equipos en los cuales se trabajará y se utilizarán como terminales, ya disponen de las licencias necesarias. En cuanto al software nuevo de desarrollo y el necesario en el servidor, son todos gratuitos y no demandan ningún tipo de inversión más que su configuración.

El capital de trabajo consiste en la asignación de los módulos necesarios para el desarrollo, por el período solicitado. Tal como se describió en el capítulo 6, los módulos necesarios para la implementación son 6 (que se traducen en 3 personas) por el período de 2 ciclos lectivos, o sea, dos años consecutivos. Estos módulos serán tomados de los asignados al departamento de Ingeniería de Software, ya que este proyecto contribuirá fehacientemente al desarrollo de las capacidades de los integrantes dotándolos de experiencia laboral, el cual es el objetivo primordial de las becas.

Durante el desarrollo de SimuRed no será necesario reinversiones. Al término del proyecto deberá realizarse un nuevo estudio de factibilidad para analizar la opción de continuar con el desarrollo ampliando la funcionalidad del mismo.

Se puede observar que la inversión en SimuReD no requiere de un desembolso de efectivo, sino de utilizar recursos disponibles dentro de la Facultad Regional Delta.

Fuentes:

<http://www.fao.org/DOCREP/003/V8490S/v8490s05.htm>

Capitulo 6 – Costos

INTRODUCCIÓN	75
ESTIMACIÓN DEL ÁMBITO DEL SOFTWARE	76
ESTIMACIÓN DE LOS RECURSOS	78
ESTIMACIÓN DE LOS PUNTOS DE FUNCIÓN	80
APLICACIÓN DEL MODELO COCOMO II	88
APLICACIÓN DEL SOFTWARE CALIBRADO POR EXPERIENCIA PROFESIONAL	91
CONCLUSIONES	96

Introducción

El costo de un proyecto de software es uno de los puntos más complicados de definir para la elaboración de un proyecto. Tal como menciona Roger S. Pressman en su libro *Ingeniería del Software*, un enfoque práctico al introducir al tema:

“Aunque la estimación es mas un arte que una ciencia, es una actividad importante que no debe llevarse a cabo de forma descuidada. Existen técnicas útiles para la estimación de costos y de tiempos. Y, dado que la estimación es la base de todas las demás actividades de planificación del proyecto y sirve como guía para una buena ingeniería del software, no es en absoluto aconsejable embarcarse sin ella.”

Para poder estimar el costo del software existen varias técnicas utilizadas por los profesionales de sistemas, pero hay una que se volvió muy popular en los últimos tiempos y se está volviendo en un estándar, ya que está teniendo gran relevancia entre los líderes de proyectos de software del mundo. Éste método de estimación de costos es COCOMO II.

COCOMO II es una adaptación mejorada del método antecesor COCOMO, el cual basaba sus cálculos dependiendo de las líneas de código. Como se puede suponer, la estimación de las líneas de código en una etapa tan temprana como es la planificación de un proyecto de software, puede ser una tarea muy tediosa, requiriendo mucha experiencia y con una obtención de error muy alta, con el cual, podría llegar a obtenerse una estimación demasiado alejada de la realidad.

El nuevo método de estimación se basa en puntos de función (PF), a través de la definición de los puntos de función se independiza al proyecto de la tecnología de desarrollo y de la habilidad y experiencia del profesional de sistemas encargado de esta tarea en la planificación del proyecto.

Los puntos de función dan una estimación del tamaño del software, y permiten obtener indicadores para la medición y control del desarrollo, como ser, indicadores de eficiencia, de calidad¹.

A partir de la estimación del costo del software se realiza todo el planeamiento financiero y económico. La importancia de la precisión en la estimación del costo es crucial, dado que una buena aproximación dará por resultado un proyecto exitoso, mientras que una mala estimación podrá hacer fracasar al proyecto en su etapa previa al desarrollo o inclusive puede determinar una suspensión del proyecto antes de terminado, dado que no cumple con los costos estipulados.

¹ Para profundizar en los temas de métricas orientadas a la función y el modelo de estimación COCOMO II remítase al anexo A y B.

Para obtener un segundo enfoque en la estimación del costo se utilizará también una técnica alternativa. Esta alternativa es generada a partir de un software creado y calibrado por una empresa dedicada al desarrollo de software desde hace más de 10 años. Se ha conseguido el uso de esta aplicación calibrada a través de un permiso especial del profesional de sistemas titular de la empresa en cuestión.

Estimación del Ámbito del Software

La especificación del ámbito del software nos lleva a evaluar la función y el rendimiento de SimuReD. Para definir esto se repasarán los objetivos y limitaciones del software en cuestión.

SimuReD es un software de simulación de eventos discretos que utiliza la teoría de colas como fundamento matemático para la resolución de sus nodos. Es un software netamente académico y para uso exclusivo de la UTN (Universidad Tecnológica Nacional Argentina) en todas sus Facultades Regionales y Unidades Académicas, especialmente diseñado para cumplir con las exigencias de la cátedra Simulación, correspondiente al cuarto nivel de la carrera de Ingeniería en Sistemas.

El simulador no tendrá restricciones de ningún tipo y será gratuito para la UTN. Será una aplicación bajo la arquitectura cliente-servidor, por lo cual deberá haber una gestión de usuarios. El servidor que contiene la aplicación estará situado en el centro de cómputos de la UTN-FRD bajo la administración técnica del Gesin.

La interfaz de usuario será simple, con una columna que contenga los nodos disponibles para realizar el modelo de red de la simulación y las funciones básicas del manejo de archivos. La funcionalidad requerida para construir el modelo será del tipo arrastrar y colocar, para la cual se utilizará la tecnología Flash. La gestión de los archivos será a través de una gestión estándar de archivos, utilizando las opciones APIs disponibles de la herramienta de desarrollo.

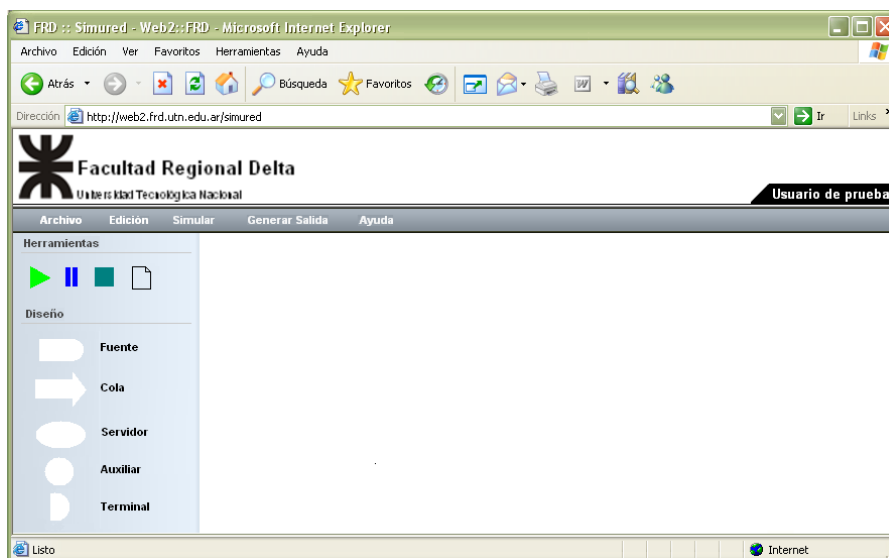


Figura 1 – Pantalla principal de SimuReD

Tendrá tres informes de salida, reporte, gráfico y salida en archivo de planilla de datos. Los primeros dos serán entregados por la pantalla del explorador luego de realizar el proceso correspondiente en el servidor, mientras que la salida por archivo se realizará mediante la gestión del archivo generado en el servidor y será transmitido al cliente mediante ftp.

De acuerdo al resumen de la aplicación, y aplicando la técnica de descomposición se obtienen las siguientes funciones:

- Autenticación de usuarios
 - Login / Acceso al sistema
 - Logout/Salida al sistema
- Proceso matemático
 - Simulación
 - Generacion de números aleatorios
 - Distribución Normal
 - Distribución Exponencial
 - Distribución Uniforme
- Definición de Nodos
 - Nodo Fuente
 - Nodo Servidor
 - Nodo Cola
 - Nodo Rama
 - Nodo Auxiliar
 - Atributos
 - Entidades
- Salidas del sistema
 - Gestión de archivos del proyecto
 - Graficos Salida
 - Informe Salida

Exportar Datos
Edición de Red

Estimación de los Recursos

Recursos Humanos

El número de personas necesarias para el desarrollo de SimuReD se estableció en tres becados y un líder de proyecto. Esta cantidad se estableció de esta manera para evitar que el proyecto se extienda más de un año.

El líder del proyecto tendrá a su cargo las tareas de infraestructura, tecnología, comunicaciones y revisión de los Casos de Uso. Así como también el control del avance del proyecto.

Los integrantes que se encuentren bajo el régimen de becas, abocados al desarrollo de SimuReD, tendrán a su cargo las tareas de documentación y desarrollo del software. Los roles que deberán desarrollar serán de analista programador en las distintas tecnologías que se implementan en SimuReD.

El analista programador Java, tendrá a su cargo el desarrollo de todas los componentes de software que correrán en el servidor.

El analista programador Flash, tendrá a su cargo el desarrollo de todo lo referente a la interfaz de usuario.

El analista programador Web, tendrá a su cargo el diseño de la estructura HTML como así también toda la interfaz de comunicación entre procesos (archivos XML).

El líder del proyecto trabajará bajo la modalidad Ad-honorem dado que el reconocimiento público que dará el desarrollo de este proyecto pagará con creces el trabajo realizado.

Las becas son gestionadas a través de la Secretaría de Asuntos Estudiantiles, bajo la dirección de Anabela Segovia, quien luego de una entrevista brindó todos los detalles del sistema de reclutamiento de los alumnos para adquirir las becas disponibles². Según trascendió el costo de cada módulo es de \$ 195.- por tal motivo el costo total de mano de obra mensual es de \$ 585.-

² Para un detalle de esta entrevista remitirse al Anexo 6C

Recursos de Software Reutilizables

La reutilización de componentes de software disminuye los costos de desarrollo pero aumenta los costos de diseño. El resultado final siempre termina en una disminución del tiempo total de desarrollo.

Las tecnologías utilizadas para SimuReD son de última generación y soportan el paradigma de objetos, en este paradigma la reutilización es uno de los pilares. Se debe invertir esfuerzo en el desarrollo de los diagramas UML para maximizar la reutilización de componentes.

La utilización de los paquetes de software, que vienen incluidos en la máquina virtual de Java, son suficientes para las exigencias matemáticas de SimuRed.

De la misma manera, las herramientas de software incluidas en Flash, minimizan el tiempo de desarrollo de las interfaces.

Herramientas de Software y Hardware

Las herramientas de Hardware a utilizar para el desarrollo de SimuReD están disponibles en la facultad. Se prevee la utilización de las máquinas del aula PC4 en un horario anterior al inicio de la jornada académica, en al menos 2 veces a la semana por no mas de 4 horas.

El sistema para la reserva del aula está a cargo del Profesor Pedro Asis, quien toma las reservas y las registra en una planilla. Se puede solicitar el aula para utilizarse siempre los mismos días por todo el cuatrimestre, o se puede hacer una reserva con una semana de anticipación.

Fuera del horario académico, la demanda del aula PC4 es muy baja, sin embargo existe la posibilidad de utilizar las máquinas del laboratorio de hardware, si esto fuera necesario.

Como servidor, se utilizarán los equipos disponibles en la regional. Cabe aclarar que estos equipos son de última tecnología y fueron donados a través de un convenio con la empresa Tenaris, por el cual todos los equipos tecnológicos que Tenaris deja de utilizar (por renovación tecnológica) pasan a manos de la FRD.

La gestión y mantenimiento de los servidores está a cargo del Gesin. En una reunión con el responsable técnico del Gesin, Cristian Pacheco, se corroboró la información. En dichos equipos puede instalarse un sistema operativo para ser administrado por el equipo de desarrollo de SimuReD, donde se instalará el software necesario. Además se puede disponer del espacio en disco necesario para alojar la documentación.

El software necesario para el desarrollo no requiere de costos adicionales, ya que todo lo necesario se distribuye bajo licencias GNU. Para mayor detalle del software, remitirse al Capítulo 3 – Tecnologías.

Estimación de los Puntos de Función

Para aplicar una metodología de estimación de esfuerzo, primero hay que saber el tamaño del software a desarrollar. La metodología de Puntos de Función, nos ofrece un indicador muy utilizado en el desarrollo de software. A partir de un indicador del tamaño del software podremos estimar los costos del desarrollo de SimuReD.

Los puntos de función (PF) son determinados a partir de cinco características clave del sistema.

Archivos Lógicos Internos y Externos

Primero se realiza la identificación de los Archivos Lógicos Internos, los cuales se ponderan según la complejidad mostrada en la Tabla 1

	1 a 19 ítems de datos referenciados	20 a 50 ítems de datos referenciados	51 o más ítems de datos referenciados
1 registro lógico	SIMPLE	SIMPLE	MEDIA
2 a 5 registros lógicos	SIMPLE	MEDIA	COMPLEJA
6 o más registros lógicos	MEDIA	COMPLEJA	COMPLEJA

Tabla 1 - Complejidad de archivos lógicos internos

Los Archivos Lógicos Internos, de acuerdo al diagrama de clases mostrado en el Capítulo 3 - Tecnologías, son los siguientes:

1. Fuente
 - a. Datos elementales referenciados: 5
 - b. Registros lógicos referenciados: 2
 - c. Grado: Simple
2. Cola
 - a. Datos elementales referenciados: 5
 - b. Registros lógicos referenciados: 2
 - c. Grado: Simple
3. Servidor
 - a. Datos elementales referenciados: 7
 - b. Registros lógicos referenciados: 4
 - c. Grado: Simple
4. Terminal
 - a. Datos elementales referenciados: 3
 - b. Registros lógicos referenciados: 1
 - c. Grado: Simple
5. Auxiliar
 - a. Datos elementales referenciados: 4
 - b. Registros lógicos referenciados: 2

- c. Grado: Simple
- 6. Rama
 - a. Datos elementales referenciados: 5
 - b. Registros lógicos referenciados: 3
 - c. Grado: Simple

Para identificar los Archivos de Interfase Externa y evaluar su complejidad se toma como referencia nuevamente la tabla 1.

- 1. Informe
 - a. Datos elementales referenciados: 25
 - b. Registros lógicos referenciados: 6
 - c. Grado: Compleja
- 2. Modelo
 - a. Datos elementales referenciados: 5
 - b. Registros lógicos referenciados: 7
 - c. Grado: Media
- 3. Ayuda
 - a. Datos elementales referenciados: 4
 - b. Registros lógicos referenciados: 1
 - c. Grado: Simple
- 4. Trazabilidad (detalle de la simulación)
 - a. Datos elementales referenciados: 29
 - b. Registros lógicos referenciados: 7
 - c. Grado: Compleja

Entradas de usuario: se cuenta cada entrada de usuario que proporciona datos a la aplicación

Dentro de SimuReD encontramos que las entradas externas o de usuario son aquellas pantallas en las que se ingresan los datos de los nodos para describir el comportamiento de cada uno de ellos en el modelo.

Para determinar la complejidad de cada entrada se utiliza la tabla 2.

	1 a 4 datos elementales referenciados	5 a 15 datos elementales referenciados	16 o más datos elementales referenciados
0 ó 1 archivo lógico referenciado	SIMPLE	SIMPLE	MEDIA
2 archivos lógicos referenciados	SIMPLE	MEDIA	COMPLEJA
3 ó más archivos lógicos referenciados	MEDIA	COMPLEJA	COMPLEJA

Tabla 2 - Complejidad de entradas externas

- 1. Propiedades de Fuente
 - a. Datos elementales referenciados: 6

- b. Archivos lógicos referenciados: 2
 - c. Grado de la función: Media
- 2. Propiedades de Cola
 - a. Datos elementales referenciados: 6
 - b. Archivos lógicos referenciados: 2
 - c. Grado de la función: Media
- 3. Propiedades de Servidor
 - a. Datos elementales referenciados: 8
 - b. Archivos lógicos referenciados: 2
 - c. Grado de la función: Media
- 4. Propiedades de Terminal
 - a. Datos elementales referenciados: 4
 - b. Archivos lógicos referenciados: 2
 - c. Grado de la función: Simple
- 5. Propiedades de Auxiliar
 - a. Datos elementales referenciados: 4
 - b. Archivos lógicos referenciados: 2
 - c. Grado de la función: Simple
- 6. Propiedades de Rama
 - a. Datos elementales referenciados: 6
 - b. Archivos lógicos referenciados: 3
 - c. Grado de la función: Compleja
- 7. Configuración de la Simulación
 - a. Datos elementales referenciados: 19
 - b. Archivos lógicos referenciados: 2
 - c. Grado de la función: Compleja

Salidas de usuario: se cuenta cada salida que la aplicación proporciona al usuario (informes, pantallas, mensajes).

Las salidas de usuario se ponderan de acuerdo a la tabla 3.

	1 a 5 datos elementales referenciados	6 a 19 datos elementales referenciados	20 o más datos elementales referenciados
0 ó 1 archivo lógico referenciado	SIMPLE	SIMPLE	MEDIA
2 a 3 archivos lógicos referenciados	SIMPLE	MEDIA	COMPLEJA
4 ó más archivos lógicos referenciados	MEDIA	COMPLEJA	COMPLEJA

Tabla 3 - Complejidad de salidas externas

- 1. Informe de la simulación.
 - a. Datos elementales referenciados: 25
 - b. Archivos lógicos referenciados: 6
 - c. Grado de la función: Compleja

2. Trazabilidad (detalle de la simulación).
 - a. Datos elementales referenciados: 30
 - b. Archivos lógicos referenciados: 8
 - c. Grado de la función: Compleja

Peticiones de usuario: entrada interactiva que produce alguna respuesta del software inmediata como salida interactiva.

Para analizar la complejidad de las peticiones de usuario o consultas externas, se analizan por separado la entrada y la salida, tomando como resultado la mayor de las complejidades. Las tablas 4 y 5 muestran la ponderación de la complejidad de acuerdo a los parámetros.

	1 a 4 datos elementales referenciados	5 a 15 datos elementales referenciados	16 o más datos elementales referenciados
0 ó 1 archivo lógico referenciado	SIMPLE	SIMPLE	MEDIA
2 archivos lógicos referenciados	SIMPLE	MEDIA	COMPLEJA
3 ó más archivos lógicos referenciados	MEDIA	COMPLEJA	COMPLEJA

Tabla 4 - Complejidad de la parte de entrada de una consulta externa

	1 a 5 datos elementales referenciados	6 a 19 datos elementales referenciados	20 o más datos elementales referenciados
0 ó 1 archivo lógico referenciado	SIMPLE	SIMPLE	MEDIA
2 a 3 archivos lógicos referenciados	SIMPLE	MEDIA	COMPLEJA
4 ó más archivos lógicos referenciados	MEDIA	COMPLEJA	COMPLEJA

Tabla 5 - Complejidad de la parte de salida de una consulta externa

1. Construcción del modelo (Entrada)
 - a. Datos elementales referenciados: 5
 - b. Archivos lógicos referenciados: 7
 - c. Grado de la función Entrada: Compleja
2. Construcción del modelo (Salida)
 - a. Datos elementales referenciados: 29
 - b. Archivos lógicos referenciados: 7
 - c. Grado de la función Salida: Compleja

Grado: Compleja.

Luego de analizar los cinco parámetros clave para el cálculo de los puntos de función, se realiza el conteo de dichos parámetros y se completa la siguiente tabla:

Parámetro de medición	Factor de ponderación						
	Simple		Media		Compleja		Total
	Cuenta	Factor	Cuenta	Factor	Cuenta	Factor	
Número de entradas de usuario	2	3	3	4	2	6	30
Número de salidas de usuario	0	4	0	5	2	7	14
Número de peticiones de usuario	0	3	0	4	1	6	6
Número de archivos	7	7	1	10	2	15	89
TOTAL Ponderación							139

A continuación se describe el análisis del grado de influencia en la determinación del factor de ajuste de acuerdo a estándares establecidos³.

1- Comunicación de datos

Los datos e informaciones de control utilizados por la aplicación son enviados o recibidos a través de recursos de comunicación de datos. Todos los dispositivos de comunicación utilizan algún tipo de protocolo de comunicación.

Grado de influencia 4 (La aplicación es más que una entrada *on-line*, y soporta sólo un protocolo de comunicación).

2- Procesamiento distribuido

Datos o procesamiento distribuidos entre varias unidades de procesamiento (CPUs) son características generales que pueden influenciar en la complejidad de la aplicación.

Grado de influencia 4 (El procesamiento distribuido y la transferencia de datos son *on-line*, en ambas direcciones).

3- Performance

Los objetivos de la performance del sistema, establecidos y aprobados por el usuario en términos de respuesta, influyen o podrían influir en el proyecto, desarrollo, implementación o soporte de la aplicación.

Grado de influencia 1 (Requerimientos de performance y de diseño fueron establecidos y previstos, sin embargo ninguna acción especial fue requerida).

4- Configuración del equipamiento

Esta característica representa la necesidad de realizar consideraciones especiales en el diseño de los sistemas para que la configuración del equipamiento no sea sobrecargada.

³ Los estándares fueron tomados de la dirección www.geocities.com/gimenezpy, página web dedicada a la métrica de Puntos de Función escrita por el Ingeniero en Sistemas José Giménez, un reconocido profesional de Paraguay.

Grado de influencia 2 (Algunas consideraciones de ajuste de performance y seguridad son necesarias).

5- Volumen de transacciones

El nivel de transacciones es alto y tiene influencia en el diseño, desarrollo, implementación y mantenimiento de la aplicación.

Grado de influencia 0: No están previstos periodos picos de volumen de transacción.

6- Entrada de datos on-line

Esta característica cuantifica la entrada de datos on-line proveída por la aplicación.

Grado de influencia 5: Más del 30% de las transacciones son entradas de datos *on-line*.

7- Interfase con el usuario

Las funciones on-line del sistema hacen énfasis en la amigabilidad del sistema y su facilidad de uso, buscando aumentar la eficiencia del usuario final. El sistema posee los items marcados en negrita:

- Ayuda para la navegación (teclas de función, accesos directos y menús dinámicos)
- **Menús**
- Documentación y ayuda on-line
- Movimiento automático del cursor
- **Scrolling vertical y horizontal**
- Impresión remota (a través de transacciones on-line)
- Teclas de función preestablecidas
- **Ejecución de procesos batch a partir de transacciones on-line**
- **Selección de datos vía movimiento del cursor en la pantalla**
- Utilización intensa de campos en video reverso, intensificados, subrayados, coloridos y otros indicadores
- Impresión de la documentación de las transacciones on-line por medio de hard copy
- **Utilización del mouse**
- **Menús pop-up**
- **El menor número de pantallas posibles para ejecutar las funciones del negocio**
- Soporte bilingüe (el soporte de dos idiomas, cuente como cuatro items)
- Soporte multilingüe (el soporte de más de dos idiomas, cuente como seis items)

Grado de influencia 5 (Más de cinco de los items descriptos fueron establecidos requerimientos en cuanto a la amigabilidad suficientes para utilizar herramientas especiales y procesos especiales para demostrar anticipadamente que los objetivos fueron alcanzados)

8- Actualización on-line

La aplicación posibilita la actualización on-line de los archivos lógicos internos.

Grado de influencia 0 (Ninguna).

9- Procesamiento complejo

El procesamiento complejo es una de las características de la aplicación, los siguientes componentes están presentes:

- Procesamiento especial de auditoria y/o procesamiento especial de seguridad
- Procesamiento lógico extensivo
- **Procesamiento matemático extensivo**
- Gran cantidad de procesamiento de excepciones, resultando en transacciones incompletas que deber ser procesadas nuevamente.
- Procesamiento complejo para manipular múltiples posibilidades de entrada/salida.

Grado de influencia 1 (uno de los items descriptos – marcado en negrita).

10 - Reusabilidad

La aplicación y su código serán o fueron proyectados, desarrollados y mantenidos para ser utilizados en otras aplicaciones.

Grado de influencia 1 (Código reutilizado fue usado solamente dentro de la aplicación).

11- Facilidad de implementación

La facilidad de implementación y conversión de datos son características de la aplicación. Un plan de conversión e implementación y/o herramientas de conversión fueron proveídas y probadas durante la fase de prueba de la aplicación.

Grado de influencia 0 (Ninguna consideración especial fue establecida por el usuario y ningún procedimiento especial fue necesario en la implementación).

12- Facilidad de operación

La facilidad de operación es una característica del sistema. Procedimientos de inicialización, respaldo y recuperación fueron proveídos y probados durante la fase de prueba del sistema. La aplicación minimiza la necesidad de actividades manuales, tales como montaje de unidades de almacenamiento, uso de papel e intervención del operador.

Grado de influencia 0 (Ninguna consideración especial de operación, además del proceso normal de respaldo establecido por el usuario).

13- Múltiples locales

La aplicación fue específicamente proyectada, diseñada y mantenida para ser instalada en múltiples locales de una organización o para múltiples organizaciones.

Grado de influencia 3 (La necesidad de múltiples locales fue considerada en el proyecto y la aplicación está separada para trabajar sobre diferentes ambientes de hardware y/o software).

14- Facilidad de cambios

La aplicación fue específicamente proyectada y diseñada con vistas a facilitar su mantenimiento. Las siguientes características pueden ser atribuidas a la aplicación:

- Están disponibles facilidades como consultas e informes flexibles para atender necesidades simples (contar 1 ítem).
- Están disponibles facilidades como consultas e informes flexibles para atender necesidades de complejidad media (contar 2 ítems)
- **Están disponibles facilidades como consultas e informes flexibles para atender necesidades complejas (contar 3 ítems)**
- Datos de control son almacenados en tablas que son mantenidas por el usuario a través de procesos on-line, pero los cambios se hacen efectivos solamente al día siguiente
- Datos de control son almacenados en tablas que son mantenidas por el usuario a través de procesos on-line, pero los cambios se hacen efectivos inmediatamente (contar 2 ítems)

Grado de influencia 3 (Tres de los ítems descriptos).

En resumen, los niveles de influencia obtenidos para las 14 características son:

Características generales del sistema	Nivel de influencia
1- Comunicación de datos	4
2- Procesamiento distribuido	4
3- Performace (desempeño)	1
4- Configuración del equipamiento	2
5- Volumen de transacciones	0
6- Entrada de datos on-line	5
7- Interfase con el usuario	5
8- Actualización on-line	0
9- Procesamiento complejo	1
10- Reusabilidad	1
11- Facilidad de implementación	0
12- Facilidad de operación	0
13- Múltiples locales	3
14- Facilidad de cambios	3
Nivel de influencia	29

A partir del nivel de influencia obtenemos el factor de ajuste aplicando la fórmula

$$\text{Factor de ajuste} = (\text{Nivel de influencia} * 0,01) + 0,65$$

Aplicando los valores estimados obtenemos un Factor de ajuste de **0,94**

Calculo de Puntos de Función

Ahora ya se tienen todos los valores para calcular los puntos de función finales de SimuReD. Aplicando las fórmulas se obtiene:

$$\begin{aligned} PF &= [PF \text{ Brutos}] * [\text{Factor de ajuste}] \\ PF &= 139 * 0,94 \\ \mathbf{PF} &= \mathbf{130,66} \end{aligned}$$

O sea que el tamaño de SimuReD es de 130 puntos de función.

Aplicación del modelo COCOMO II

Los puntos de función sirven como indicador del tamaño del software. COCOMO II no trabaja con puntos de función, sino con Líneas de Código (LDC) o miles de líneas de código KLDC.

Para convertir los puntos de función en KLDC hay un estándar donde se tabulan las equivalencias entre los puntos de función y las KLDC para cada uno de los lenguajes de programación⁴. Allí encontramos que la equivalencia para los lenguajes de programación utilizados en el desarrollo de SimuReD son los mostrados en la tabla junto al porcentaje de uso de cada lenguaje en SimuReD.

Lenguaje	% Uso	LDC x PF
HTML	10	16
Java	85	53
SQL	5	13

De acuerdo a lo descripto podemos calcular las KLDC a partir de los PF calculados anteriormente (PF=130.66).

$$LDC = [(130.66*0.1)*16] + [(130.66*0.85)*53] + [(130.66*0.05)*13] = 6180.218$$

$$\mathbf{KLDC = 6.18}$$

De acuerdo a las características de SimuReD el Modo COCOMO a implementar es **Orgánico**, ya que se trata de un desarrollo relativamente chico (COCOMO contempla un umbral aproximado de 50 KLDC para considerar un proyecto pequeño). Además el entorno donde será desarrollado es estable.

Para el cálculo se utilizará el **modelo intermedio** de COCOMO ya que la aproximación es mayor. Para ello se analizan los 15 atributos (Las especificaciones de los atributos pueden consultarse en el Anexo 6C – Método de Estimación COCOMO).

Atributos del producto software

⁴ La tabla completa de equivalencias puede consultarse en <http://www.spr.com/library/0langtbl.htm>

- RELY: Fiabilidad requerida del software. Muy bajo, un error en la ejecución no produce ningún tipo de pérdidas.
- DATA: Tamaño de la base de datos. Bajo, sólo se utiliza para la gestión de usuarios y algunos parámetros de configuración general.
- CPLX: Complejidad del producto. Alto, dada la complejidad en las funciones matemáticas del núcleo del simulador.

Atributos del hardware

- TIME: Limitaciones en el tiempo de ejecución. Nominal, se utilizará un servidor exclusivo.
- STOR: Limitaciones de memoria principal. Nominal, se utilizará un servidor exclusivo.
- VIRT: Volatilidad de la máquina virtual. Bajo, no se prevén cambios en los servidores.
- TURN: Frecuencia de cambio en el modelo de explotación del ordenador. Bajo, SimuReD es un sistema interactivo.

Atributos de personal

- ACAP: Capacitación de los analistas. Nominal, se trata de un equipo con formación académica en la FRD.
- AEXP: Experiencia en aplicaciones. Muy bajo, los alumnos que solicitan las becas rara vez poseen experiencia previa.
- PCAP: Capacitación de los programadores. Nominal, se trata de un equipo con formación académica en la FRD.
- VEXP: Experiencia en la máquina virtual. Muy bajo, la experiencia en la máquina es menor de un mes.
- LEXP: Experiencia en el lenguaje de programación. Nominal, los alumnos reciben capacitación en Java a partir de segundo año.

Atributos del proyecto

- MODP: Prácticas Modernas de programación. Muy alto, se utilizarán herramientas UML y metodologías de objetos.
- TOOL: Uso de herramientas para el desarrollo de software. Nominal, se utilizarán herramientas estándar de uso actual.
- SCED: Limitaciones en la planificación. Nominal.

Los valores obtenidos son los siguientes:

VARIABLE	Muy bajo	Bajo	Valores Nominal	Alto	Muy alto	Extra alto
RELY	0.75	0.88	1.0	1.15	1.40	
DATA		0.94	1.0	1.08	1.16	
CPLX	0.70	0.85	1.0	1.15	1.30	1.65
TIME			1.0	1.11	1.30	1.65
STOR			1.0	1.06	1.21	1.56
VIRT		0.87	1.0	1.15	1.30	
TURN		0.87	1.0	1.07	1.15	
ACAP	1.46	1.19	1.0	0.86	0.71	
AEXP	1.29	1.13	1.0	0.91	0.82	
PCAP	1.42	1.17	1.0	0.86	0.70	
VEXP	1.21	1.10	1.0	0.90		
LEXP	1.14	1.07	1.0	0.95		
MODP	1.24	1.10	1.0	0.91	0.82	
TOOL	1.24	1.10	1.0	0.91	0.83	
SCED	1.23	1.08	1.0	1.04	1.10	

Lo cual arroja un FAE = **0,785**

Cálculos Resultantes

Cálculo del esfuerzo de desarrollo⁵:

$$E = a \text{ KLDC}^b * \text{FAE} = 3,2 * (6,18)^{1,05} * 0,785 = \mathbf{17 \text{ personas /mes}}$$

Cálculo del tiempo de desarrollo⁶:

$$T = c \text{ Esfuerzo}^d = 2,5 * (17)^{0,38} = \mathbf{16,15 \text{ meses}}$$

Personal promedio:

$$P = E/T = 17 / 16,15 = \mathbf{1,05 \text{ personas}}$$

Resultado

Para el desarrollo de SimuReD se necesitará 1 persona a jornada completa, trabajando alrededor de 16 meses.

Una jornada completa equivale a 8hs diarias (160hs mensuales) y un becario trabaja 24hs mensuales, por lo que para completar las horas de una jornada completa se necesitarán 7 becados para concretar el proyecto en 16 meses.

Dado que el ciclo lectivo cuenta con períodos de 10 meses y la asignación de becas es en forma anual, el tiempo de desarrollo debe ser múltiplo de 10 para optimizar los recursos. De esta manera, extendiendo el tiempo a 20 meses de desarrollo (multiplicando por un factor de 1,25), obtenemos un valor de 5,6 personas para el desarrollo. 3 becas son las que se consiguen, y el project leader deberá trabajar por una carga horaria correspondiente a 2,6 módulos (16 horas semanales) para cumplir con el tiempo.

⁵ Las constantes a y b fueron tomadas según se muestra en el Anexo 6B, de acuerdo al modo Orgánico

⁶ Las constantes c y d fueron tomadas según se muestra en el Anexo 6B, de acuerdo al modo Orgánico

Aplicación del software calibrado por experiencia profesional

Basado en los métodos anteriormente aplicados, se obtuvo autorización para el uso de una planilla que abstrae al planificador de la mayoría de los cálculos minimizando el tiempo de estimación.

Esta planilla de datos fue realizada por expertos en la materia, y continuamente modificada hasta obtener valores estables que converjan en una estimación de tiempo dentro de valores aceptables.

Los resultados arrojados fueron comparados con los resultados reales, obtenidos al término de cada proyecto, para poder realizar un ajuste en los parámetros de cálculos. De esta manera se llegó a obtener una estimación estándar que no supera el 10% de error en el tiempo estimado del proyecto.

La información necesaria para estimar el tiempo del proyecto se divide en 3 grupos: factores externos, especificaciones de cambio y tareas especiales.

Los factores externos refieren a las personas que participan activamente en el proyecto y el tiempo que le podrán dedicar. Dentro de este grupo podemos distinguir:

- **Equipo de trabajo:** Cantidad de personas, cuanto conocimiento tiene el equipo del tema en promedio y en que otros temas participan los integrantes en promedio, esto último se da sólo cuando los integrantes también tienen el mantenimiento de otros sistemas y el tiempo que demandan las incidencias.
- **Referentes:** Cantidad de usuarios o personas del cliente que participan del proyecto, que nivel de conocimiento y dedicación tendrán con el proyecto.
- **Aplicación:** Es el proyecto en sí, cuanto más desconocido más complejo y cuanto más conocimiento y documentación disponible menos esfuerzo.
- **Contexto:** Este último está relacionado con los ambientes de prueba, pérdida de tiempo por equipos lentos o con dificultades para ejecutar procesos, por ejemplo: si no se pudieran ejecutar ciertos procesos y se debe solicitar la ejecución a un operador, en este caso hay una pérdida de tiempo adicional. En cuanto a los ambientes de prueba se evalúa la dificultad para armarlos, esto es evaluar el espacio y el muestreo de datos, si el espacio no limita, se bajan los datos completos desde producción entonces no habrá inconsistencias en dichos datos. Caso contrario, se deben crear consultas en la base de datos crear un contexto apropiado para obtener resultados concretos. También se podría crear el ambiente inicial y disponer de espacio para backups, así cualquier problema se puede volver el ambiente al estado anterior.

Los valores seteados para SimReD son los siguientes (Columna Dato):

	Descripcion	Dato	Factor
Equipo de Trabajo			
F1	Personas afectadas al requerimiento	4+	10%
F2	Nivel de experiencia promedio	Bajo	10%
F3	Tiempo dedicado a otras tareas	Nulo	-10%
Referente			
F4	Número de referentes involucrados	1	0%
F5	Nivel de conocimiento promedio	Alto	-10%
F6	Nivel de disponibilidad	Nulo	10%
Aplicación			
F7	Nivel de complejidad	Medio	10%
F8	Nivel de conocimiento	Bajo	10%
F9	Nivel de documentación	Medio	5%
F10	Dificultad para construir el ambiente de prueba	Bajo	10%
F11	Dificultad para ejecutar la prueba	Medio	20%
Contexto			
F12	Calidad del entorno de desarrollo	Bueno	5%
F13	Calidad del entorno de prueba	Bueno	5%
F14	Tiempo improductivo de maquina	Nulo	0%
F15	Nivel de riesgo	Nulo	0%

Valores para los Factores Externos

Justificación de los valores seleccionados:

- F1 - Personas afectadas al requerimiento: 3 alumnos becados + 1 Project Leader
- F2 - Nivel de experiencia promedio: El promedio de alumno que solicita becas no tiene experiencia previa.
- F3 - Tiempo dedicado a otras tareas: Las becas serán asignadas exclusivamente a este proyecto.
- F4 - Número de referentes involucrados: El único referente será Luís Perna, quien es el profesor titular de la materia Simulación en la FRD.
- F5 - Nivel de conocimiento promedio: del referente.
- F6 - Nivel de disponibilidad: dada la carga horaria de Luís Perna, se lo podrá encontrar previa solicitud de entrevista.
- F7 - Nivel de complejidad: no se trata de un software de alta complejidad.
- F8 - Nivel de conocimiento: Se involucra el nivel de conocimiento del negocio (dado por la cátedra Simulación) y el nivel de conocimiento en programación.
- F9 - Nivel de documentación: se tiene amplia disponibilidad bibliográfica.
- F10 - Dificultad para construir el ambiente de prueba: No se presentan dificultades.
- F11 - Dificultad para ejecutar la prueba: No se presentan dificultades, pero se van a necesitar varios modelos para simular.
- F12 - Calidad del entorno de desarrollo: Se disponen de servidores de última generación con el apoyo técnico necesario por parte del Gesin.
- F13 - Calidad del entorno de prueba: se dispone de espacio suficiente para realizar backups de datos.
- F14 - Tiempo improductivo de maquina: No aplica.

F15 - Nivel de riesgo: No existen riesgos del contexto que afecten al proyecto.

En las especificaciones de cambio se carga toda la información correspondiente a los cambios a realizar o, en nuestro caso, a la especificación del desarrollo. La carga de información se divide en dos grupos, por un lado se detallan los datos del componente y por el otro los datos del desarrollo.

Para definir los datos del componente, primero se deben definir cada uno de los componentes que integrarán el software. Para ello se definieron los casos de uso (CU) que integrarán la aplicación. También se debe definir la tecnología con la cual se realizará el desarrollo y asignar una complejidad a cada uno de los componentes. La información cargada en dicha planilla es:

Nombre	Lenguaje	Complejidad
CU-001: Login/Acceso al sistema	PL-SQL	Baja
CU-002: Generador de números aleatorios	Java	Alta
CU-003: Distribución Normal	Java	Alta
CU-004: Distribución Exponencial	Java	Alta
CU-005: Distribución Uniforme	Java	Alta
CU-006: Gestión de documentos	Html	Media
CU-007: Nodo Fuente	Web-Paginas Dinamicas Forms	Media
CU-008: Nodo Servidor	Web-Paginas Dinamicas Forms	Media
CU-009: Nodo Cola	Web-Paginas Dinamicas Forms	Media
CU-010: Nodo Rama	Web-Paginas Dinamicas Forms	Media
CU-011: Nodo Auxiliar	Web-Paginas Dinamicas Forms	Media
CU-012: Atributos	Java	Media
CU-013: Entidades	Java	Baja
CU-014: Simulación	Java	Alta
CU-015: Gráficos Salida	Web-Paginas Dinamicas Reports	Media
CU-016: Informe Salida	Web-Paginas Dinamicas Reports	Media
CU-017: Exportar Datos	Html	Baja
CU-018: Edición del modelo de Red	Web-Paginas Dinamicas Forms	Alta
CU-019: Logout/Salida al sistema	PL-SQL	Baja

En los datos del desarrollo, se debe definir la complejidad del cambio o especificar si es un nuevo desarrollo. Para nuestro caso será 100% componentes nuevos. En la última columna se estima un porcentaje de reusabilidad de código. En nuestro caso, y al utilizar tecnologías OO (Orientadas a Objeto) la reutilización es alta.

La información cargada en cada uno de las especificaciones solicitadas son las que se muestran en el cuadro siguiente:

Datos del Desarrollo												
Nombre	Tipo	Horas Estimadas del componente	Diseño funcional					Prueba de usuario	Seguimiento y Control			% Líneas Reusables
			Diseño técnico	Programación	Prueba unitaria	Preparación de la prueba	Prueba de integración		Capacitación	Implantación	Seguimiento y Control	
			E1	E2	E3	E4	E5	E6	E7	E8	E9	
CU-001	Nuevo	27	2	3	12	2	4		1	1	3	0%
CU-002	Nuevo	133	11	13	60	11	19		4	3	13	0%
CU-003	Nuevo	133	11	13	58	11	19		4	3	13	5%
CU-004	Nuevo	133	11	13	58	11	19		4	3	13	5%
CU-005	Nuevo	133	11	13	58	11	19		4	3	13	5%
CU-006	Nuevo	67	5	7	30	5	9		2	1	7	0%
CU-007	Nuevo	67	5	7	28	5	9		2	1	7	10%
CU-008	Nuevo	67	5	7	28	5	9		2	1	7	10%
CU-009	Nuevo	67	5	7	28	5	9		2	1	7	10%
CU-010	Nuevo	67	5	7	28	5	9		2	1	7	10%
CU-011	Nuevo	67	5	7	28	5	9		2	1	7	10%
CU-012	Nuevo	67	5	7	29	5	9		2	1	7	5%
CU-013	Nuevo	27	2	3	12	2	4		1	1	3	5%
CU-014	Nuevo	133	11	13	60	11	19		4	3	13	0%
CU-015	Nuevo	67	5	7	30	5	9		2	1	7	0%
CU-016	Nuevo	67	5	7	30	5	9		2	1	7	0%
CU-017	Nuevo	27	2	3	12	2	4		1	1	3	0%
CU-018	Nuevo	133	11	13	60	11	19		4	3	13	0%
CU-019	Nuevo	27	2	3	10	2	4		1	1	3	20%

De acuerdo a toda la información requerida ya se puede tener una estimación inicial. En nuestro caso este valor es de 2047 horas.

Etapas del requerimiento	Porc. Distrib.	Horas Estimadas	Horas Ajustadas	Total	Factores Externos																			
					Equipo			Referente			Aplicación					Contexto								
					F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15					
E1	Diseño funcional	7%	121	151	25%		10%	-10%	0%	-10%	10%	10%	10%	5%							0%			
E2	Diseño tecnico	9%	151	189	25%		10%	-10%	0%			10%	10%	5%							0%	0%		
E3	Programacion y Prueba unitaria	43%	659	890	35%	10%	10%	-10%				10%	10%							5%		0%	0%	
E4	Preparacion de la prueba	9%	121	182	50%	10%	10%	-10%				10%	10%	5%	10%						5%	0%	0%	
E5	Prueba de integracion	16%	211	338	60%	10%	10%	-10%				10%	10%	5%						20%		5%	0%	0%
E6	Prueba de usuario	3%	32	54	70%	10%	10%	-10%	0%	-10%	10%	10%	10%	5%	10%	20%					5%	0%	0%	
E7	Capacitacion	3%	45	61	35%		10%		0%	-10%	10%	10%	10%	5%										
E8	Implantacion	2%	30	35	15%									10%	5%								0%	
E9	Seguimiento y Control	8%	151	174	15%	10%		-10%	0%					10%							5%		0%	
				Estimación inicial:		2074																		

Por último, se deben cargar tareas especiales, en caso que sea necesario. Estas son las tareas propias de la aplicación y/o cliente que no esté definida dentro de los estándares. Además, hay un dato que se debe cargar manualmente y es el tiempo que daremos al usuario para que realice las pruebas de acuerdo al plan de pruebas desarrollado.

Con la estimación de las tareas especiales, ya tenemos la estimación del esfuerzo, medido en horas, que demandará el desarrollo de SimuReD.

Tareas Especiales	Horas
Prueba de usuarios	32
Diseño Web	60
Estimación total:	2134

El resultado arrojado es de 2134 horas.

Esas horas son horas netas, entonces si en SimuReD trabajarán 3 personas con beca y el Project Leader con el doble de asignación de tiempo, cada integrante del equipo tendrá una carga horaria de:

$$\text{Carga Horaria} = 2134 \text{ [horas]} / 5 \text{ [personas]} = 426,8 \text{ [horas/persona]}$$

Como cada integrante trabaja bajo el sistema de becas de la facultad, y cada módulo de becas exige el cumplimiento de 6 horas semanales, podemos calcular las semanas que demanda el proyecto.

$$\text{Semanas de duración} = 426,8 \text{ [horas/persona]} / 6 \text{ [horas]} = 71,10 \text{ [semanas/persona]}$$

Resultado final:

- Becarios: 3
- Project Leader (PL): 1
- Carga horaria semanal: 6 hs becarios – 12 hs PL
- Duración del proyecto (en semanas): 71
- Cada ciclo lectivo tiene 10 meses, descontando los períodos de inactividad, la equivalencia es 35 semanas.

Conclusiones

De acuerdo a los diversos métodos utilizados para calcular el costo de desarrollo de SimuReD se obtienen las siguientes conclusiones:

- Personas en el equipo: 3 becados + 1 project leader.
- Tiempo estimado de desarrollo: 20 meses (2 años lectivos)

Tipo	Descripción	Costo (\$ ARG)
Recursos de Software	Se utilizará 100% software libre, no se prevee el pago de licencias de ningún tipo	0.-
Recursos de Hardware	Se utilizan equipos disponibles en la FRD.	0.-
Recursos Humanos	Se contratarán 3 becados por 2 años lectivos (20 meses) y el project leader trabajará ad-honorem	35.100.-
Otros Recursos	Se estima el costo del consumo eléctrico ⁷ de un equipo promedio durante 30 hs semanales (42 kW/mes). Agua, limpieza, mobiliario y demás se consideran despreciables ya que son servicios que están disponibles inclusive si el proyecto no se concreta.	162.-
Costo Actual de SimuReD		35.262.-

⁷ Precio del kWh en Campana por la empresa Eden SA es de 0.1931\$/kWh

Capítulo 7 – Financiamiento

INTRODUCCIÓN	99
ENTIDADES BANCARIAS	99
OTRAS OPCIONES DE FINANCIAMIENTO	102
CONCLUSIÓN	105

Introducción

El objetivo de este capítulo es describir la fuente de financiamiento que permita el desarrollo de SimuReD. Como ya hemos mencionado este proyecto está destinado a obtener un beneficio tecnológico, no tangible y no económico.

El financiamiento de SimuReD estará costado por el Programa de Becas de Investigación y Servicio de la UTN. Cabe recordar que para el desarrollo de SimuReD los recursos económicos necesarios están 100% ligados a la mano de obra, ya que el resto de los recursos necesarios se encuentran a disposición dentro de la FRD y no requieren inversión extra¹.

A continuación se realiza un análisis de las distintas fuentes de financiamiento alternativas para verificar si la opción elegida es la que mejor se adapta a las circunstancias.

Entidades Bancarias

El Banco Santander Río cuenta con un programa de impulso a la Educación Superior, por el cual se destinan fondos dirigidos a docentes, investigadores y alumnos de las Universidades que tienen convenio con el programa Santander Río Universidades, en el cual la UTN es participante.



La segunda edición del Plan comprende el período 2007-2009, el cual durante el año 2007 contempló las siguientes iniciativas:

- Becas de Grado y Post-grado Santander 150 Aniversario
- Becas de movilidad interna Alumnos, Docentes y Personal Técnico
- Premio Santander Río al Mejor Egresado 2006
- Premio Santander Río al Mérito Académico
- Colaboración Científica
- Becas de Investigación aplicada a la Economía Regional
- Programa de incubación de Empresas
- Becas de transferencia de tecnología
- Premio Jóvenes emprendedores
- Becas de Ayuda Social

Además este banco posee otro programa interesante, dentro de su Plan de Desarrollo Estratégico de Emprendimientos, llamado Nexo Emprendedor.

¹ Para mayor información, remitirse a los capítulos 5 (Inversión) y 6 (Costos).

La explicación de este mecanismo se transcribe desde la página del banco, donde se explican sus detalles:

Fondo de Innovación Tecnológica (FIT)

Nexo Emprendedor es el *fund manager* del FIT, un Fondo de Capital de Riesgo lanzado el 1ero de enero 2007, que busca realizar inversiones de capital en *start-ups* y proyectos de investigación y desarrollo surgidos del sistema de Ciencia y Tecnología argentina.

Filosofía del Fondo de Innovación Tecnológica

- Invertir en empresas argentinas con potencial y visión para desembarcar en mercados internacionales.
- Financiar proyectos argentinos de I+D+i (Investigación, Desarrollo e Innovación) que propongan innovaciones tecnológicas globales.
- Apoyar equipos con capacidades para liderar el proceso de expansión de sus emprendimientos.
- Sumar inversores que aporten *know-how*, redes de contacto y otros recursos estratégicos a las compañías del porfolio.

Perfil de los proyectos

- Ser innovador, de alto valor agregado y tener un perfil competitivo a nivel internacional.
- Tener un plan para encarar una etapa de crecimiento significativa.
- Contar con un *management* con capacidades para liderar el proceso de expansión.
- Preferentemente, estar avalado por una Universidad o Centro de Investigación.
- Ser generador de empleo calificado y socialmente responsable.
- Requerir una inversión entre U\$S 30.000 y U\$S 1.000.000.
- Para empresas en la fase de *start-up*, estar actualmente facturando. Para proyectos I+D+i, haber superado la fase de validación tecnológica.

Como podemos apreciar se trata de un fondo destinado a impulsar desarrollos con un fuerte perfil innovador y tecnológico. Destina el beneficio principalmente a desarrollos de interés global y a equipos con la capacidad de soportar una amplia expansión. También vemos que este proyecto tiene el perfil requerido.

Continuando con el análisis de este programa encontramos dos cuestiones interesantes que tienen que ver con los objetivos de este plan, los cuales transcribimos a continuación:

Fomento al I+D+i (Investigación, Desarrollo e Innovación)

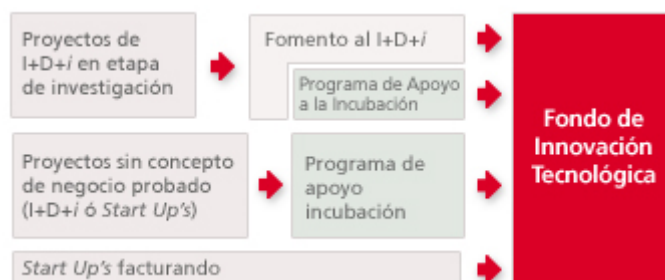
El objetivo es impulsar la creación de empresas argentinas de base tecnológica, competitivas globalmente, financiando estudios de validación técnica y productiva.

- Estudio de validación científica: Son estudios externos independientes que evalúan la factibilidad de la tecnología propuesta.
- Estudio de validación productiva: Son estudios externos independientes que analizan la factibilidad de realizar el proyecto a escala industrial de producción, incluyendo costo de fabricación, activos y maquinarias involucradas en el proceso, estudio de proveedores y distribuidores.

Apoyo a la Incubación

El objetivo es facilitar a los emprendedores de alto potencial el proceso de planificación de sus negocios, financiando el proceso de incubación de sus proyectos seleccionados en incubadoras universitarias homologadas.

Programa Fomento al I+D+i y Apoyo a la Incubación



Vemos entonces que el objetivo del plan contribuye en gran manera al objetivo de este proyecto, por lo que se convierte en una opción interesante.

Sería muy apresurado (o suerte de principiantes) tomar esta opción sin analizar otras, por lo que al finalizar con todas se tomará la más correcta para el caso.

Continuando con las instituciones bancarias analizamos el caso del Banco de la Provincia de Buenos Aires. Vía e-mail comunicaron que los mecanismos crediticios con los que cuentan no aplican a proyectos de este tipo. Sin embargo, recomendaron un sistema de créditos que tiene implementada la provincia de Buenos Aires a través de la Lotería de la Provincia por el cual se impulsan emprendimientos y proyectos desde su etapa inicial por medio de microcréditos.



Si bien este mecanismo de financiación encaja dentro del marco tecnológico de este proyecto, no aplica al marco geográfico, ya que el alcance se limita a proyectos dentro de la provincia y no a nivel Nacional, como este caso.

El Banco de la Nación Argentina no da soporte directo como impulsor de proyectos, sino que los mismos se implementan a través de otros organismos dependientes de la Presidencia de la Nación. Estos casos serán analizados posteriormente cuando terminemos con las opciones bancarias.



Por otra parte se consultó otras instituciones bancarias como BBVA Banco Francés y Bank Boston pero estas instituciones no poseen programas financieros para el impulso de proyectos tecnológicos desde sus etapas iniciales.

Otras Opciones de Financiamiento

La Presidencia de la Nación difunde desde hace unos años, un concurso de innovación y tecnología destinado a proyectos. El concurso es anual, diferenciándose tres períodos, uno de presentación de proyectos, otro de análisis de los mismos y un tercero de difusión. Los premios son en efectivo y sus montos varían de acuerdo a las distintas categorías.



Se establecen 7 categorías² con premios específicos y un premio general, con las siguientes asignaciones:

- *Producto Innovador: Categoría abierta, dirigida a personas físicas mayores de 18 años sin distinción de ocupación y personas jurídicas.* En nuestro caso no corresponde, dado que el producto no es innovador, sino que es el desarrollo de una herramienta con conceptos previamente desarrollados.
- *Diseño Industrial: Categoría dirigida a Diseñadores Industriales (DI) y estudiantes de facultades de diseño de todo el país.* Podría establecerse dentro de esta categoría, ya que al tratarse de un desarrollo para uso académico de una Universidad Tecnológica se extiende su beneficio al uso industrial.
- *Diseño Gráfico.* No aplica para esta categoría.
- *Investigación aplicada: categoría dirigida a grupos de investigación.* Si bien se debe hacer investigación para el desarrollo, no es el principal objetivo del proyecto.
- *Innovaciones en el agro: Categoría abierta, dirigida a personas físicas mayores de 18 años, sin distinción de ocupación, y personas jurídicas.* No aplica para este proyecto.
- *Software aplicado al diseño de videojuegos: Categoría dirigida a Diseñadores de video juegos, profesionales de sistemas, estudiantes de facultades de diseño y/o sistemas de todo el país, y otras personas físicas mayores de 18 años y empresas dedicadas a la industria del software.* Esta categoría podría ser una de las más propicias para este proyecto, ya que el desarrollo de un simulador discreto puede usarse como motor de un software de videojuegos.
- *INET: Categoría dirigida a escuelas técnicas y agrotécnicas.* No aplica a este proyecto.

El premio general, denominado INNOVAR, consiste en un premio adicional de \$20.000.- al mejor proyecto seleccionado entre los ganadores de todas las categorías anteriores.

Este proyecto podría incorporarse en alguna de las categorías antes mencionadas pero el objetivo real y el alcance del mismo hace que las categorizaciones sean muy débiles.

² Extraídas de www.innovar.gov.ar

FONTAR

Continuando con los ofrecimientos del Gobierno Nacional, nos encontramos con el FONTAR (Fondo Tecnológico Argentino) dependiente del Ministerio de Ciencia, Tecnología e Innovación Productiva que a través de la Agencia Nacional de Promoción Científica y Tecnológica promueve la financiación de proyectos bajo el lema “2008 – Año De La Enseñanza De Las Ciencias”, destinando un fondo de más de cuarenta y cinco millones de pesos para el año en curso.

Dentro de los tipos de proyectos que contempla este plan³ podemos destacar dos: “Servicios Tecnológicos para Instituciones” y “Capacitación”.

Los Servicios Tecnológicos para Instituciones son proyectos destinados a ofrecer un respaldo a instituciones cuyos laboratorios y centros de investigación necesiten fortalecer su aptitud para prestar servicios técnicos al sector privado. Se contemplan en estos proyectos gastos de infraestructura, equipamiento y capacitación. Los créditos promueven la mejora o instalación de servicios altamente calificados. Las instituciones beneficiarias deberán contar con una organización técnica y administrativa que garantice la adecuada utilización de los recursos adjudicados. Las entidades interesadas podrán presentarse individualmente o en forma asociada. Podrán ser beneficiarias Instituciones públicas o privadas del Sistema Científico-Tecnológico, Universidades Nacionales y Cámaras gremiales empresarias.

Los proyectos Capacitación son los que tienen por objetivo la capacitación y el reentrenamiento del personal de la empresa en nuevas tecnologías de producción o de gestión. Estas actividades pueden tener una vinculación directa con la demanda de recursos humanos capacitados, que surja de un proyecto de desarrollo tecnológico o modernización tecnológica. También pueden responder a un propósito más general, orientado al desarrollo de capacidades en sectores industriales o tecnológicos.

Aportes No Reembolsables (ANR 600)

La Agencia Nacional de Promoción Científica y Tecnológica, a través del Fondo Tecnológico Argentino (FONTAR), cuenta con un sistema de financiación de Proyectos de Innovación Tecnológica para la adjudicación de Aportes No Reembolsables destinados a proyectos de desarrollo tecnológico presentados por empresas PyMEs en el marco del Programa de Modernización Tecnológica.

El programa tiene como objetivo financiar parcialmente proyectos que tengan como meta mejorar las estructuras productivas y la capacidad innovadora de las empresas productoras de bienes y servicios de distintas ramas de actividad, mediante la ejecución de proyectos de desarrollo tecnológico.

³ Para una descripción completa de todos los tipos de proyectos que financia FONTAR, remitirse al Anexo 7A o bien visitar el sitio web www.agencia.gov.ar

Son beneficiarios del programa empresas productoras de bienes y servicios que satisfagan la condición de PyME, constituidas como tales al momento de la presentación de la solicitud y radicadas en el territorio nacional. No podrán ser beneficiarios Instituciones sin fines de lucro, ni dependencias gubernamentales.

Los recursos administrados en el marco de la presente Convocatoria consistirán en Aportes No Reembolsables de hasta seiscientos mil Pesos (\$600.000.-). En ningún caso estas subvenciones podrán exceder el 50% del costo total del proyecto, debiendo la empresa beneficiaria aportar el resto.

FONSOFT

El Fonsoft es el fondo fiduciario de promoción de la industria del software que se creó en el 2004, a partir de la sanción de la Ley de Promoción de la Industria del Software (Ley 25.922). El Fonsoft promueve el fortalecimiento de las actividades de producción de software a nivel nacional.

El FONSOFT financia:

- Proyectos de investigación y desarrollo relacionados a las actividades comprendidas en el régimen de promoción (creación, diseño, desarrollo, producción e implementación y puesta a punto de los sistemas de software).
- Programas de nivel terciario o superior para la capacitación de recursos humanos.
- Programas para la mejora en la calidad de los procesos de creación, diseño, desarrollo y producción de software.
- Programas de asistencia para la constitución de nuevos emprendimientos.

Los medios de financiación que ofrece el FONSOFT son a través de convocatorias públicas y abiertas que se realizan anualmente. Luego de la presentación de los proyectos (por los cuales hay que pagar un canon), integrantes del FONSOFT analizan cada una de las propuestas recibidas, seleccionando los proyectos beneficiarios del plan. Por otro lado, se encuentra la denominada “Ventanilla Instrumentos”, por la cual se pueden gestionar créditos directos con la institución.

Medios De Financiación FONTAR

1. Ventanilla Instrumentos:
 - a. Créditos Exporta son créditos para la iniciación o consolidación en la actividad exportadora de empresas PyMEs productoras de software.
2. Convocatorias FONSOFT:
 - a. ANR FONSOFT están destinados al fortalecimiento de la industria del software, para su financiamiento parcial a través de Aportes No Reembolsables (ANR). Se dividen en las categorías:
 - i. Certificación de Calidad.
 - ii. Desarrollo de nuevos productos y procesos de software.
 - iii. Investigación y Desarrollo precompetitivo de productos y procesos de software.

- b. FONSOFT Emprendedores: comprende a todas las personas físicas, empresas unipersonales y sociedades, los cuales pueden presentar sus Proyectos de Desarrollo de Productos de Software y Servicios Informáticos, para su financiamiento parcial a través de Subsidios.

FONCAP

FONCAP es un programa impulsado a través del Ministerio de Desarrollo Social - Secretaría de Políticas Sociales y Desarrollo Humano que tiene por objetivo “eliminar las barreras de acceso al crédito para el sector de la microempresa de menores recursos económicos, apoyando la organización representativa y su articulación con los demás sectores de la sociedad. Otorgar créditos al Sector Privado o Social a través de: organizaciones financieras y comerciales, organizaciones sin fines de lucro, legalmente constituidas.”

El FONCAP está orientado a instituciones u organizaciones privadas o sociales prestadoras de servicios financieros o no financieros de todo el país. Quedan excluidos los organismos plenamente estatales.

Además el plan ofrece los siguientes servicios:

- Asistencia financiera a sector microempresarial.
- Diseño de ingenierías financieras específicas.
- Asistencia técnica a instituciones de financiamiento para micro empresas en funcionamiento.
- Capacitación para el gerenciamiento e implementación de operaciones financieras para micro empresas.
- Apoyo técnico para la puesta en marcha de operaciones financieras para micro empresas.
- Asesoramiento en la formulación de proyectos de financiamiento y planes de negocios.
- Seguimiento y evaluación de operaciones para micro empresas.
- Asesoramiento a instituciones públicas o privadas en el diseño, desarrollo e implementación de políticas de financiamiento para micro empresas.

Conclusión

Las opciones de financiamiento para el desarrollo de este tipo de proyectos es muy variada y se encuentran disponibles tanto por instituciones públicas como privadas.

Los medios de financiamiento que mejor se podrían adaptar son los otorgados por el Banco Santander Rio y los otorgados por el FONSOFT. Sin embargo, dada las características de este proyecto, que proponen el desarrollo de SimuReD en su primera etapa y la metodología de desarrollo que propone una generación de valor en la institución, la mejor opción de financiamiento es la inicialmente propuesta (a través del sistema de Becas de la UTN).

La información aquí presentada no debe ser descartada, ya que en futuras ampliaciones de SimuReD podrían utilizarse alguno de los instrumentos de financiamiento antes descriptos dado que el proyecto encaja muy bien dentro de las clasificaciones y tiene muchas posibilidades de calificar para los créditos, e inclusive, para los ANR (ya sea por parte del FONTAR o por FONSOFT).

Fuentes:

www.mincyt.gov.ar

www.agencia.gov.ar

www.santanderrio.com.ar

www.bna.com.ar

www.bpba.com.ar

www.bapro.com.ar

Capítulo 8 - Rentabilidad

INTRODUCCIÓN	109
ANÁLISIS COSTO-BENEFICIO	109
CÁLCULO DE INDICADORES	112
CONCLUSIÓN	114

Introducción

El problema de la determinación de la rentabilidad de un conjunto de oportunidades de inversión es tan antiguo como la propia Ciencia Económica.

La rentabilidad es la medida del rendimiento que, en un determinado periodo de tiempo, producen los capitales utilizados en una inversión.

De acuerdo al tipo de inversión se tiene el tipo de beneficio o rendimiento que se obtiene. Cuando una inversión es privada se busca incrementar el beneficio, o sea, incrementar el valor del capital en juego, para este tipo de inversiones se estudia la rentabilidad a través de la teoría de la inversión. Por el contrario, cuando la inversión es pública se lleva a cabo para aumentar el bienestar social, en este caso, la rentabilidad es estudiada a través del análisis costo-beneficio.

En el caso de SimuReD, la rentabilidad está asociada a un bienestar social, ya que no se busca un beneficio económico sino un beneficio institucional. Por ello se realizará el análisis costo-beneficio.

Este tipo de análisis ha tomado muchas ideas, conceptos y métodos de la teoría de la inversión. Su necesidad es debida a la disparidad entre el beneficio neto privado y el social. Si no existiera esta disparidad los métodos de valoración y selección aplicables serían los mismos para cualquier tipo de inversión. La teoría de la inversión no es capaz de valorar servicios públicos fundamentales como seguridad, defensa, educación, etc.

Existen dos tendencias en el análisis costo-beneficio:

- Valorar los efectos indirectos, positivos y negativos sobre el bienestar social.
- Suplantar el sistema de precios de mercado por otro sistema de precios (denominados teóricos, sombra o de referencia) que respondan a una situación de bienestar social.

Luego de valorar el flujo de ingresos y gastos de una inversión pública se aplican los criterios de decisión de inversiones públicas que son similares a los que se utilizan en la empresa privada.

El objetivo principal de SimuReD es agregar valor a la UTN y en mayor medida a la Regional Delta.

Análisis Costo-Beneficio

Diferentes métodos pueden ser utilizados para calcular la relación Costo/Beneficio. Los métodos más sofisticados consideran el tiempo - valor del dinero como parte del análisis del Costo/Beneficio. El tiempo - valor del dinero, también conocido como factor de descuento, es simplemente un método utilizado para convertir el valor futuro del dinero en

valor presente. Se basa en que un peso hoy tiene más valor que un peso en un tiempo futuro debido a los intereses o a la ganancia que se pueda obtener. Incluir el tiempo – valor del dinero puede ser crucial para una organización ya que los esfuerzos por mejorar pueden requerir compromisos de capital por un período prolongado. Los métodos mas utilizados son:

- Punto de equilibrio: es el tiempo que se tomaría para eu el total de los ingresos incrementados o la reducción de gastos sea igual al costo total. En este caso no se toma en cuenta el valor del dinero en el tiempo.
- Período de devolución: es el tiempo requerido para recuperar el monto inicial de una inversión de capital. Este método calcula la cantidad de tiempo que se tomaría para lograr un flujo de caja positivo igual a la inversión total. Este método indica la liquidez del esfuerzo por mejorar un proceso en lugar de su rentabilidad y no tiene en cuenta el valor del dinero en el tiempo.
- Valor Actuan Neto (VAN): representa el valor presente de los flujos de caja salientes menos la inversión inicial. El valor presente del flujo de caja futuro es calculado utilizando el costo del capital como un factor de descuento.
- Tasa Interna de Retorno (TIR): es la tasa de interés que hace la ecuación de la inversión inicial con el valor presente de los futuros flujos de caja entrantes.

Para aplicar cualquiera de los indicadores descriptos se deberá estimar en valor económico de los beneficios, tarea que como se ha dicho antes, es necesaria para realizar el análisis de rentabilidad.

Como primer beneficio directo contamos con el uso de SimuReD dentro de la Regional Delta. El uso del simulador para la cátedra, por un lado brindará una herramienta más versátil, con menor tiempo de aprendizaje, lo cual se traduce en una mejora significativa en el desempeño de la cátedra. Por otro lado, la falta de limitaciones para el modelado beneficiará a aquellos alumnos que se sientan motivados por la simulación pudiendo llevar los modelos propuestos en la cátedra a un nivel profesional.

Estos dos beneficios tienen beneficios secundarios, que tal vez pueden demostrarse en tiempos futuros, cuando los alumnos formen parte de equipos para la toma de decisiones. Allí, los alumnos, más capacitados en el arte de la simulación, estarán mejor preparados para el análisis mediante el modelado, dado que se ha ganado calidad y velocidad en el aprendizaje de esta técnica. Se estima que la primera versión operativa estará disponible en el primer año de desarrollo, por este motivo el beneficio comienza a calcularse a partir del mes 11. El valor estimado es calculado a partir de una media entre todos los productos y se estima en un costo de U\$s 200.- (\$ 700.- ARG).

A partir del segundo cuatrimestre del segundo año (mes 15) se estima que la versión estará operativa para todas las regionales de la UTN. Dado que el beneficio es igual para todas las regionales se calcula el valor para las 11 instituciones donde se dicta Ingeniería en Sistemas de Información¹, esto es \$ 7700.-

¹ Para conocer el detalle de todas las regionales habilitadas para utilizar SimuReD, consultar el capítulo 2 (Estudio de Mercado)

Si bien el alcance de este proyecto no pone a SimuReD como un desarrollo comercial, éste tiene un valor en el mercado. Ese valor debe considerarse como un activo, ya que es un software desarrollado con fondos de la UTN y por lo tanto, agrega valor económico a la Facultad Regional Delta.

Considerar a SimuReD como un activo de la FRD significa que en un futuro la regional podría llegar a licenciar el software con el consiguiente ingreso de fondos que ello significa. Para estimar un valor económico en el mercado de SimuReD se consultaron los valores actuales del mercado para los productos detallados en el capítulo 2.

Producto	Precio (U\$s)	Condiciones
Arena - Rockwell	\$795.00	Basic edition - 1 licencia x 12 meses
ExtendSim	\$995.00	CP (Version básica) - Adicional por servicio tecnico: \$199/anual
Flexim	\$15,500.00	Precio por licencia de la Profesional Edition.
GPSS	\$1,750.00	Proof 5 - version minima
PowerSim	\$65.00	Versión especial por promoción - restringida

fuelle: <http://www.wior.uni-karlsruhe.de/bibliothek/Simulation/com>

Tabla 1 – Valores de software de simulación (Enero/09)

De acuerdo a las características de SimuReD y a los productos presentados en la Tabla 1, podríamos estimar que el valor por medio de un sistema de licencias anuales, rondaría los U\$s 300.- como mínimo. Si, en cambio, se estima el valor del software éste podría ascender a los U\$s 2000.- en la versión que se entregará al culminar este proyecto.

Si bien estos valores son estimativos, se consideraron los siguientes puntos:

- SimuReD no tiene la suficiente trayectoria comercial como Arena o Flexim.
- Se tiene el respaldo de la UTN y el apoyo de varios profesionales especialistas en ciencias exactas, TI y modelado (a nivel UTN como dentro de la Regional Delta).
- El software generado cumplirá con las mas modernas técnicas de desarrollo por lo que se prevee calidad identica o superior a los productos en el mercado.
- Acceso a la infraestructura necesaria para el desarrollo y distribución.
- SimuReD podrá trabajar on-line y se lo podría instalar en una intranet para uso compartido, optimizando de esta manera el intercambio de información y el tiempo de procesamiento.

Ahora bien, para calcular el beneficio en unidades monetarias se empieza a considerar el valor de SimuReD a partir de generada la documentación (Casos de Uso y Anexos Técnicos). Se estima que esta documentación estará lista a partir del segundo cuatrimestre del primer año (mes 5). Como el valor comercial de SimuReD al término del proyecto se estima en U\$s 2.000.- (\$ 7.000.- ARG) se realiza un incremento en el valor desde el mes 5 hasta finalizar el proyecto. Esto se logra incrementando el valor en un 10% por mes que es el valor que se agrega al proyecto.

Los costos fueron calculados en el capítulo 6 y consisten en el pago de las becas y los gastos adicionales que implicaría el desarrollo del proyecto que es el uso de la electricidad que consumen las PC en el tiempo de desarrollo.

Los Costos y Beneficios convertidos en un valor económico se traducen en el siguiente cuadro.

Mes	Costo			Beneficio			
	Costo Becas	Otros Gastos	Total Gastos	Uso en FRD	Uso en UTN	Valor Comercial	Total Beneficios
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	585.00	8.11	593.11	0.00	0.00	0.00	0.00
2	585.00	8.11	593.11	0.00	0.00	0.00	0.00
3	585.00	8.11	593.11	0.00	0.00	0.00	0.00
4	585.00	8.11	593.11	0.00	0.00	0.00	0.00
5	585.00	8.11	593.11	0.00	0.00	1500.00	1500.00
6	585.00	8.11	593.11	0.00	0.00	1650.00	1650.00
7	585.00	8.11	593.11	0.00	0.00	1815.00	1815.00
8	585.00	8.11	593.11	0.00	0.00	1996.50	1996.50
9	585.00	8.11	593.11	0.00	0.00	2196.15	2196.15
10	585.00	8.11	593.11	0.00	0.00	2415.77	2415.77
11	585.00	8.11	593.11	700.00	0.00	2657.34	3357.34
12	585.00	8.11	593.11	0.00	0.00	2923.08	2923.08
13	585.00	8.11	593.11	0.00	0.00	3215.38	3215.38
14	585.00	8.11	593.11	0.00	0.00	3536.92	3536.92
15	585.00	8.11	593.11	0.00	0.00	3890.61	3890.61
16	585.00	8.11	593.11	0.00	7700.00	4279.68	11979.68
17	585.00	8.11	593.11	0.00	0.00	4707.64	4707.64
18	585.00	8.11	593.11	0.00	0.00	5178.41	5178.41
19	585.00	8.11	593.11	0.00	0.00	5696.25	5696.25
20	585.00	8.11	593.11	0.00	0.00	6265.87	6265.87
21	0.00	0.00	0.00	0.00	0.00	7000.00	7000.00

Tabla 2 – Costos / Beneficios

El próximo paso consiste en calcular los indicadores Valor Futuro, VAN y TIR para evaluar la rentabilidad del proyecto.

Cálculo de indicadores

Para calcular el valor futuro de SimuReD se utiliza el método de Interés Compuesto, ya que en una inversión a Interés Compuesto, los intereses de un período se suman al capital inicial y son reinvertidos/capitalizados para obtener más intereses en los siguientes períodos.

La fórmula para t períodos de tiempo, es:

$$VF = VA * (1 + i)^t$$

Donde:

- VF: Valor Futuro.
- VA: Valor Actual o vigente.
- I: tasa de interés anual.
- T: número de períodos en años, de la inversión.

Ahora debemos calcular el valor futuro para el uso del software dentro de la FRD, a nivel UTN y su valor comercial al término del proyecto.

Aplicando la fórmula para el uso en la FRD, 1 año y un interés del 10% obtenemos:

$$VF = 700 * (1 + 0.10)^1 = 770$$

Aplicando la fórmula para 1.5 años y un interés del 10% obtenemos:

$$VF = 7.700 * (1 + 0.10)^{1.5} = 8.883$$

Aplicando la fórmula para 2 años y un interés del 10% obtenemos:

$$VF = 7.000 * (1 + 0.10)^2 = 8.470$$

Por lo tanto el valor futuro de SimuReD es:

$$VF = \$18.123.-$$

Para calcular el VAN tomamos los egresos discriminados entre el primer año y el segundo y también tomamos como ingresos los beneficios (en unidades monetarias) en el segundo año. Esto es considerar el valor del activo al finalizar el proyecto. Los valores finales para calcular el VAN son:

$$VAN = -C_0 + \frac{C_1}{(1+r)} + \frac{C_2}{(1+r)^2}$$

C_0 = Capital Inicial = \$ 0.-

C_1 = Flujo neto (Ingresos – Egresos) del primer año = \$ - 5931.10

C_2 = Flujo neto en el segundo año = \$ (18123 - 5931.10) = \$ 12191.90

Tasa = 10%

$$VAN = 0 + \frac{-5931.10}{(1.10)} + \frac{12191.90}{(1.10)^2}$$

$$VAN = 4684.05$$

Para calcular la TIR (Tasa Interna de Retorno) se debe igualar el VAN a cero para obtener el valor del rendimiento r . Esto se traduce en definir cual es el porcentaje mínimo requerido para que el proyecto comience a ser rentable.

$$0 = -C_0 + \frac{C_1}{(1+r)} + \frac{C_2}{(1+r)^2}$$

Reemplazando los valores en la fórmula antecedente y realizando los calculos correspondientes obtenemos:

$$TIR = 1.05$$

Conclusión

Analizando el Valor Actual Neto y la Tasa Interna de Retorno, podemos observar que el proyecto es económicamente factible.

El VAN = 4684.05 (o sea, mayor a cero) indica que el proyecto es económicamente viable.

La TIR trata de considerar un número en los méritos de un proyecto. Dicho número no depende de la tasa de interés que rige el mercado de capitales. El número es interno al proyecto y solo depende de los flujos de caja del proyecto.

Una inversión es aceptable si su TIR excede al rendimiento requerido. Según muchos analistas, debería aceptarse cualquier proyecto cuya TIR sea superior al COC (El COC se define como la rentabilidad que se deja de percibir en una alternativa de inversión por haber elegido otra)

- Si la TIR es mayor al COC significa que el VAN descontado con ese COC debe ser necesariamente positivo, en consecuencia la inversión es aceptable.
- Si fuera igual el VAN igualaría a cero. La inversión bajo análisis no sería diferente a la alternativa no elegida de la que obtuvimos el COC.
- Si fuera menor, el VAN sería negativo, en cuyo caso la inversión debería desecharse.

Por lo tanto, las inversiones viables son aquellas que tienen TIR superior a la tasa de descuento. Ese excedente es lo que ubica al proyecto en una rentabilidad superior al estándar elegido, que está representado por el COC².

El resultado calculado de la TIR es de 1.05, o sea 105% de tasa. Como la tasa utilizada para calcular el VAN fue del 10% se concluye que la tasa es mayor a la esperada por lo que el proyecto es aceptable.

Como resultado del análisis de los indicadores calculados se concluye que SimuReD tiene una rentabilidad favorable, lo que se traduce en una buena inversión para la UTN.

Más allá de la rentabilidad económica se obtendrán beneficios concretos adicionales que no deben desestimarse, a continuación se describen alguno de ellos.

- Mayor capacitación en el alumnado, esto se traduce en una mejora en la calidad académica de los egresados de la facultad, brindando a la casa de altos estudios una mejor imagen ante las empresas de la zona.
- Incentivo a la investigación y desarrollo de sistemas dentro del ámbito de la facultad.
- Mejora en la utilización de los modelos de simulación, clave para el desarrollo de la toma de decisiones de los nuevos profesionales.
- Mayor prensa local que ayuda a despertar el interés de futuros estudiantes.

² Extraído del libro Evaluación de Proyectos de Gabriel Baca Urbina, ed McGraw-Hill (2001)

- Colabora en la integración de las diferentes regionales donde se dicta la carrera de Ingeniería en Sistemas de Información, tanto a nivel docente, como a nivel alumnado.

Fuente:

- wikipedia.org
- zonaeconomica.com
- Preparación y Evaluación de Proyectos, de Nassir Sapag Chain y Reinaldo Sapag Chain, Ed. McGraw-Hill (1989).
- Evaluación de Proyectos de Gabriel Baca Urbina, Ed. McGraw-Hill (2001).

Capítulo 9 – Legales

CAPITULO 9 – LEGALES	117
INTRODUCCIÓN	119
REGISTRO DE LA MARCA	119
REGISTRO DE DOMINIOS	120
LA INDUSTRIA DEL SOFTWARE	121
REGLAMENTO DE BECAS DE INVESTIGACIÓN Y SERVICIO	122
PROPIEDAD INTELECTUAL	122
LICENCIAS DE SOFTWARE	123
CONCLUSIÓN	124

Introducción

El desarrollo de cualquier tipo de proyecto está regulado por un marco jurídico e institucional que regula los derechos y deberes de las relaciones entre los miembros del equipo afectado a éste.

La importancia del análisis del marco legal en una etapa temprana del proyecto es crucial, dado que se deben conocer las leyes, decretos, ordenanzas, acuerdos y resoluciones que afecten el desarrollo en una etapa temprana. Esto evitará incurrir en faltas legales por desconocimiento evitando multas o penalidades que hasta puedan hacer fracasar el proyecto. Por otro lado se podrían desperdiciar mecanismos legales que beneficien directamente el desarrollo del mismo.

Este capítulo trata todo lo referente al marco legal donde se encuadra el desarrollo de SimuReD. El objetivo del presente capítulo es analizar los alcances legales que rigen para el entorno de desarrollo de software dentro de una institución pública, como lo es la UTN.

Con lo anteriormente expuesto se define el alcance de este capítulo que abarca el ámbito académico dentro de la República Argentina.

Registro de la Marca

Dentro de los activos de toda Organización uno de los activos más preciados de los que se dispone son las ideas. Tal vez uno de los activos menos tangibles pero a la vez mas valiosos. A través de las ideas se concretan los hechos y es muy importante poder transmitir las ideas para la concreción de objetivos precisos.

El éxito de un proyecto depende en gran medida de la imagen que el usuario perciba. El mensaje transmitido para asegurar que un producto sea reconocido en el mercado debe ser protegido tan eficientemente como sea posible.

Excelentes productos o servicios pueden ser ofrecidos, pero si el público es incapaz de diferenciarlos claramente, pasaran inadvertidos a la sombra de aquellos con mayor presencia en el mercado.

Los consumidores son atraídos por las marcas que representan calidad y confiabilidad, y es por esta razón que las marcas comerciales, que son las que permiten a los productos y servicios ser distinguidos en el mercado, deben ser protegidas cuidadosamente.

El medio de proteger la identidad es a través del registro de su marca comercial. El registro de la marca comercial que identifica a su compañía le dará el derecho exclusivo de usar la marca comercial para identificar todos los productos y servicios ofrecidos por la compañía.

La Ley 22.362 y el Decreto Reglamentario 558/8, modificado por Decreto 1141/03, garantizan el proceso bajo el cual es posible registrar una marca comercial. La autoridad de aplicación de la Ley de Marcas es la Dirección de Marcas, del Instituto Nacional de Propiedad Industrial (I.N.P.I).¹

De acuerdo a las especificaciones de la Ley de marcas² SimuReD es registrable, para ello hay que completar la solicitud pertinente y realizar los pagos correspondientes a tal fin los cuales pueden consultarse en la página oficial del ente regulador de marcas www.inpi.gov.ar/templates/marcas_registradas.asp

Una vez que este proceso ha sido completado y la solicitud cumple con todos los requisitos de la legislación vigente, la solicitud es concedida y el titular obtiene el Título a la Marca Registrada expedido por el I.N.P.I.

El arancel inicial, es de doscientos pesos (\$ 200.-). No mediando objeciones (vistas administrativas u oposiciones.) sólo se deberá abonar un arancel adicional mínimo por el retiro de título. El trámite del registro lleva aproximadamente 18 meses, siempre y cuando no se produzca ninguna objeción a la solicitud. El plazo del derecho adquirido al registrar la marca es de 10 años contados desde la fecha de concesión del registro, renovable por períodos iguales y sucesivos.

Las marcas tienen validez territorial, por esta razón, si se desea registrar la marca en otros países se deben gestionar los registros correspondientes en cada región.

Registro de Dominios

En la era informática en la que vivimos, la marca del producto está directamente asociada a la disponibilidad de dominios existentes en Internet. Es por ello que es pertinente verificar y registrar los dominios de forma prematura para evitar perder la tutoría de los mismos.

La presencia en Internet se está volviendo cada día más en un requisito fundamental para una marca. Tal es el caso que, según críticos del medio tecnológico, reconocen que las marcas que no tienen presencia en la Red tienden a la desaparición.

SimuReD es una herramienta tecnológica educativa, por lo que la presencia en Internet, bajo un dominio propietario, se torna fundamental. Es por ello que se registró el dominio www.simured.com.ar, se reservó el subdominio www.frd.utn.edu.ar/simured y se está utilizando el dominio web2.frd.utn.edu.ar/simured, este último es exclusivo de la herramienta de colaboración instalada en la Regional Delta.

A partir de la aprobación de este proyecto, se comienza con el registro del dominio clave para la aplicación www.simured.edu.ar, el mismo se debe registrar a través de la ARUI desde su página de internet www.riu.edu.ar.

¹ Texto adaptado de la página oficial de la dirección de marcas: www.inpi.gov.ar

² Para consultar la Ley remitirse al anexo 9D.

La ARIU (Asociación Redes de Interconexión Universitaria) es un emprendimiento conjunto de las universidades nacionales e institutos universitarios integrantes del CIN (Consejo Interuniversitario Nacional) con el propósito de llevar adelante la gestión de redes para facilitar la comunicación informática a nivel nacional e internacional de las universidades nacionales, promoviendo la investigación informática, tecnológica, educativa y el desarrollo cultural en el área de las tecnologías de información y comunicaciones.

Para realizar el registro de este dominio se debe completar el formulario correspondiente³, el cual se puede encontrar en la página de referencia, y enviar una nota oficial tal como se describe en el Art.4 de las reglas de registro.

“Las solicitudes de instituciones educativas oficiales tendrán aceptación definitiva cuando la autoridad competente del organismo registrante, titular del CUIT y/o CUE, haga llegar a la RIU una nota oficial, con membrete de la dependencia y firma original del funcionario a cargo de la misma, en la que se solicite el nombre de dominio en cuestión para dicho organismo.”

La Industria del Software

A nivel nacional rigen dos leyes que benefician al desarrollo del software. Una de ellas, la Ley 25.856, define que el desarrollo de software debe ser considerado como una actividad industrial. Esta medida afecta a las empresas de desarrollo de software en el sentido en que las mismas son consideradas para los distintos beneficios económicos y financieros ofrecidos para la industria en general, tal como lo describe en su artículo 1°.

ARTICULO 1° - Establécese que la actividad de producción de software debe considerarse como una actividad productiva de transformación asimilable a una actividad industrial a los efectos de la percepción de los beneficios impositivos, crediticios y de cualquier otro tipo que se fijen para la industria por parte del Gobierno nacional.

A modo de complemento, la Ley 95.922, promueve la industria del software mediante beneficios adicionales a todas las empresas de desarrollo que estén debidamente inscriptas. Los beneficios rigen por un lapso de 10 años y básicamente constituyen en beneficios sobre los impuestos. La reducción de los pagos se considera estabilidad fiscal, y se describen en el capítulo II de la ley en los artículos 7, 8 y 9 que se transcriben a continuación:

ARTICULO 7° - Los sujetos que adhieran a este régimen gozarán de estabilidad fiscal por el término de diez (10) años contados a partir del momento de la entrada en vigencia de la presente ley. La estabilidad fiscal alcanza a todos los tributos nacionales, entendiéndose por tales los impuestos directos, tasas y contribuciones impositivas que tengan como sujetos pasivos a los beneficiarios inscriptos. La estabilidad fiscal significa que los sujetos que desarrollen actividades de producción de software no podrán ver incrementada su carga tributaria total nacional al momento de la incorporación de la empresa al presente marco normativo general.

ARTICULO 8° - Los beneficiarios del régimen de la presente ley que desempeñen actividades de investigación y desarrollo en software y/o procesos de certificación de calidad de software desarrollado en el territorio nacional y/o exportaciones de software (asegurando a los trabajadores de la actividad la legislación laboral vigente), podrán

³ Formulario de registro on-line: <http://www.riu.edu.ar/fd.html>

convertir en un bono de crédito fiscal intransferible hasta el 70% (setenta por ciento) de las contribuciones patronales que hayan efectivamente pagado sobre la nómina salarial total de la empresa con destino a los sistemas y subsistemas de seguridad social previstos en las leyes 19.032 (INSSJyP), 24.013 (Fondo Nacional de Empleo) y 24.241 (Sistema Integrado de Jubilaciones y Pensiones). Los beneficiarios podrán utilizar dichos bonos para la cancelación de tributos nacionales que tengan origen en la industria del software, en particular el impuesto al valor agregado (IVA) u otros impuestos nacionales y sus anticipos, en caso de proceder, excluido el impuesto a las ganancias. El bono no podrá utilizarse para cancelar deudas anteriores a la efectiva incorporación del beneficiario al régimen de la presente ley y, en ningún caso, eventuales saldos a su favor harán lugar a reintegros o devoluciones por parte del Estado.

ARTICULO 9° - Los sujetos adheridos al régimen de promoción establecido por la presente ley tendrán una desgravación del sesenta por ciento (60%) en el monto total del impuesto a las ganancias determinado en cada ejercicio. Este beneficio alcanzará a quienes acrediten gastos de investigación y desarrollo y/o procesos de certificación de calidad y/ o exportaciones de software, en las magnitudes que determine la autoridad de aplicación.

En el caso de SimuReD, estos beneficios no son aplicables, dado que la UTN no debe pagar ningún tipo de aporte por el desarrollo del mismo.

Reglamento de Becas de Investigación y Servicio

El reglamento completo del sistema de Becas de investigación y servicio puede consultarse en el anexo 9C, allí se puede ver que la asignación de las becas debe pasar por una serie de preselección entre los candidatos previamente inscriptos.

La clasificación del rectorado para las becas son 3: becas de servicio, becas de investigación y becas de ayuda social económica.

A cada regional se le asigna una cantidad de módulos, determinada por diversos parámetros, como la población y el nivel socio-económico del entorno. La regional dispone y distribuye de los módulos de acuerdo a las necesidades de cada caso. La gestión y distribución de los módulos⁴ se realiza a través de la SAE (Secretaría de Asuntos Estudiantiles).

Para SimuReD las becas serán para desarrollar técnicas no administrativas que cubran necesidades de la unidad académica, por eso encuadran en las becas de servicio.

Propiedad Intelectual

La propiedad intelectual protege las obras de sus autores para garantizar sus derechos sobre los esfuerzos realizados y evitar que otras personas puedan apoderarse de ella para generar un lucro o autoría sobre el bien.

En la Constitución Nacional se reconoce la propiedad intelectual en el Artículo 17:

“Todo autor o inventor es propietario exclusivo de su obra, invento o descubrimiento, por el término que le acuerde la ley”

⁴ Para un detalle del circuito de selección de beneficiarios de las becas remitirse al anexo 6C

El software es una creación, una obra intelectual, y como tal, protegida por la Ley 11.723 de Propiedad Intelectual. Según el artículo 2 de dicha ley el derecho de propiedad de una obra científica, literaria o artística, comprende para su autor la facultad de disponer de ella, de publicarla, de ejecutarla, de representarla, y exponerla en público, de enajenarla, de traducirla, de adaptarla o de autorizar su traducción y de reproducirla en cualquier forma.

El no cumplimiento de la ley 11.723 aplica la pena del delito de estafa, que es de 1 mes a 6 años de prisión, pudiendo el autor denunciar el delito mediante los canales legales ordinarios.

Licencias de Software

Una licencia de Software es el contrato por el cual se transmite el derecho de uso de un software, generalmente a cambio de dinero. En la industria del software existen varios tipos de licencias, en los cuales algunos son pagos y otros gratis.

Una nueva corriente de desarrollos en comunidades está haciendo de las licencias gratis una opción de muy buena calidad, igualando o superando a veces, las licencias comerciales. A modo de ejemplo podemos mencionar Mozilla Firefox frente a Internet Explorer.

Los tipos de licencia gratuitos, que son bajo los cuales se desarrolla SimuReD, más utilizados en la actualidad son los freeware y shareware. La diferencia radica en que las licencias freeware se aplican a todo el software en cuestión y sin restricciones en su uso, mientras que las licencias shareware proporcionan limitaciones en el uso gratuito.

Las licencias freeware, por lo general, permiten la redistribución del software pero con ciertas restricciones, como por ejemplo no agregar modificaciones o no revender las licencias. No hay que confundir una licencia freeware con el término open source o código abierto, ya que esto es un error.

Open source, se refiere a aquel software que se distribuye con el código fuente y por lo tanto, se permite su modificación. El software opensource está basado en la filosofía del software libre, impulsada por Richard Stallman⁵ bajo el proyecto GNU. Y por la cual se establece que todo el software debe ser libre, esto no quiere decir que sea gratis, sino que se debe disponer del código para realizar mejoras en forma incremental.

En la actualidad existe una organización llamada Free Software Foundation, o Fundación para el Software Libre, que introdujo el concepto de licencia GPL (General Public License) y que establece los derechos de uso del Software Libre. Los programas adquiridos bajo la licencia GPL pueden ser usados, copiados, modificados y redistribuidos libremente.

El Shareware es otra modalidad de comercialización todavía más extendida, el programa se distribuye con limitaciones, bien como versión de demostración o evaluación,

⁵ Richard Matthew Stallman (Manhattan, Nueva York, 16 de marzo de 1953), es un programador estadounidense y figura relevante del movimiento por el software libre en el mundo.

con funciones o características limitadas o con un uso restringido a un límite de tiempo establecido (por ejemplo 30 días). Así, se le da al usuario la oportunidad de probar el producto antes de comprarlo y, más tarde, adquirir la versión completa del programa. Un ejemplo muy claro de este tipo es el software antivirus, estas compañías suelen permitir la descarga de sus productos de evaluación que sólo son válidos para un determinado número de días. Una vez superado el máximo, el programa se bloquea y es necesario comprar el producto si se desea seguir utilizándolo.

Conclusión

De acuerdo a las disposiciones legales vigentes para el territorio de la República Argentina, el desarrollo de SimuRed no presenta conflictos o contradicciones al sistema jurídico que lo afecta.

En el entorno nacional, existen leyes de promoción y apoyo a los desarrollos de software. Si bien, en el caso de SimuReD no son aplicables, ya que el alcance de dichas leyes rige para el ambiente empresarial, es importante conocer el alcance y los beneficios de las mismas, ya que en un futuro SimuReD podría continuar su desarrollo en manos privadas.

De acuerdo a las reglamentaciones sobre la propiedad del proyecto, podemos observar que al desarrollarse en la UTN, ésta casa de altos estudios es la propietaria del mismo, teniendo la facultad de disponer del software a través de sus representantes legales.

El software utilizado para el desarrollo se encuentra bajo licencias freeware. Estas licencias permiten el uso del software completo en forma gratuita y con alcance al presente proyecto.

Fuentes:

www.gnu.org

fsfeurope.org

www.inpi.gov.ar

www.mincyt.gov.ar

www.portaldeabogados.com.ar

www.presidencia.gov.ar

Capítulo 10 – Marco Teórico Técnico

<u>CAPITULO 10 – MARCO TEÓRICO TÉCNICO</u>	125
---	------------

INTRODUCCIÓN	127
MAPA CONCEPTUAL DE LA FUNDAMENTACIÓN TEÓRICA POR MATERIA DE CARRERA.	127
DIAGRAMA ESQUEMÁTICO DEL MARCO TEÓRICO TÉCNICO DEL PROYECTO	131

Introducción

El conocimiento adquirido a través del desarrollo de carrera de Ingeniería en Sistemas de Información, me aportó gran cantidad de conocimientos sin los cuales no hubiera podido ser posible la concreción del presente proyecto.

A lo largo de este capítulo se analizará en que grado aportó cada una de las cátedras en referencia a SimuReD.

Cabe aclarar que en los momentos que fue necesario consultar a alguno de los profesores de la Regional Delta, los mismos se ocuparon del tema de manera entusiasta y con ánimos de colaboración. Las consultas de índole técnica y teórica colaboraron con muchas de las definiciones aquí presentes.

Mapa Conceptual de la Fundamentación Teórica por Materia de Carrera.

Cada materia de la carrera de Ingeniería en Sistemas de Información, contribuyó en mayor o en menor medida para la realización de este proyecto. A continuación se ofrece un cuadro donde puede observarse una breve descripción de la materia de la carrera y el aporte que se le atribuye en este proyecto.

Materia	Descripción	Aportes
Administración de Recursos	En todo proyecto el factor limitante siempre son los recursos. A través de este curso se obtuvo herramientas para análisis y optimización de los recursos disponibles. En la segunda parte de la materia, se analizaron exclusivamente los recursos humanos, con técnicas y métodos actuales de selección y motivación.	Conocimiento de herramientas de medición de calidad de administración. Métodos de selección y motivación de personal. Gestión de las personas.
Administración Gerencial	Estructura organizativa desde el punto de vista de la toma de decisiones. Se trataron temas como tablero de control y FODA, herramientas imprescindibles para la toma de decisiones.	Claridad en la toma de decisiones de los distintos niveles jerárquicos. Estructura interna del impacto en la toma de decisiones.
Álgebra y Geometría Analítica	Elementos para la resolución de problemas con indeterminaciones. Representación de las figuras geométricas y su correspondencia matemática.	Herramientas matemáticas aplicables en el cálculo para la simulación de procesos.
Algoritmos y Estructura de Datos	Implementación de soluciones procedurales a través de gráficos y pseudo código.	Imprescindible para la implementación de todo desarrollo de software.
Análisis de Sistemas	Gestión previa de un proyecto de software. Utilización práctica de diagramas de análisis sistémicos y herramientas de recolección de información como son las encuestas, entrevistas y diseño de formularios.	De gran ayuda para realizar el estudio de mercado por las técnicas desarrolladas para la formulación de preguntas y tratamiento de las personas a la hora de recolectar información.

Materia	Descripción	Aportes
Análisis Matemático I	Desarrollo teórico de funciones y su tratamiento en el plano bidimensional. Aplicaciones de las operaciones matemáticas sobre tales funciones y los teoremas y axiomas que las respaldan.	Las definiciones y aplicaciones adquiridas son básicas para la implementación de modelos.
Análisis Matemático II	Desarrollo teórico de funciones en n dimensiones, extendiendo la aplicación matemática adquirida en Análisis Matemático I.	Extensión de la teoría adquirida en Análisis Matemático I que amplía el campo de modelado.
Arquitectura de Computadoras	Descripción de los componentes básicos y elementales concernientes al hardware del procesamiento electrónico.	Amplía la capacidad técnica de modelado
Arquitectura de Computadoras II	Configuración electrónica y aplicación de los mecanismos de procesamiento interno de una computadora.	Amplía la capacidad técnica de modelado
Calor, Electricidad y Magnetismo	Fundamentación teórica de los eventos físicos relacionados con el calor, la electricidad y el magnetismo.	Amplía la capacidad técnica de modelado
Comunicaciones	Estudio del medio y los distintos modos de comunicaciones entre equipos utilizando todas las tecnologías disponibles.	Optimización de las comunicaciones entre procesos del sistema interno. Comprensión de los modelos compuestos por líneas de comunicación.
Diseño de Sistemas	Diseño de los distintos niveles de un sistema, desde la aplicación de las formas normales para el diseño de la base de datos, hasta la definición de los estándares de diseño de pantallas y otras interfaces de usuario.	Contribuye para el diseño de las interfaces y la optimización de la base de datos.
Economía	Teoría del modelo económico actual.	Amplía la capacidad técnica de modelado
Estadística Aplicada	Recolección de datos y análisis de los mismos para evaluar a que tipo de modelo matemático pertenecen los mismos.	Efectiva para la adquisición de datos y fundamental para la calidad del modelo.
Física	Fundamentos básicos de la naturaleza física del medio.	Amplía la capacidad técnica de modelado
Fundamentos p/el Anál. de Señales	Desarrollo teórico de funciones en el plano complejo y su aplicación para el tratamiento de señales.	Amplía la capacidad técnica de modelado
Gestión de Datos	Diseño y administración eficiente de la base de datos para obtener información de ella y resguardar datos agregando valor al negocio.	Mejora el desempeño técnico del desarrollo al aportar herramientas esenciales para la información contenida.
Gestión Ingenieril	La Organización desde un punto de vista Gerencial.	Excelente para obtener la visión de la Organización desde el punto de vista del tomador de decisiones.
Ingeniería de Software	Aplicación de metodologías de última generación para el desarrollo de software.	Altamente favorable y contribuye con todo lo referente a la definición de infraestructura del software y fundamentalmente en las definiciones tecnológicas del proyecto.
Ingeniería y Sociedad	Muestra el compromiso social del Ingeniero y la importancia de su buen desempeño.	Agrega valor a los objetivos del proyecto.

Materia	Descripción	Aportes
Inglés I	Técnicas de lectura	Permite acceder a mayor cantidad de información para cualquiera de los usos.
Inglés II	Técnicas de lectura	Permite acceder a mayor cantidad de información para cualquiera de los usos.
Inteligencia Artificial	Fundamentos esenciales de la inteligencia creada por el hombre.	Recorre los límites a los cuales se puede llegar con la simulación.
Inteligencia Artificial II	Aplicación de modelos que permiten crear entidades que simulan tener inteligencia.	Amplifica la visión del modelado, llevando el mismo a un cambio paradigmático.
Investigación Operativa	Aplicación de modelos matemáticos a problemas usuales que permiten la optimización de recursos.	Ofrece herramientas matemáticas que se pueden aplicar dentro de un modelo de simulación.
Legislación	La importancia de las leyes y el entorno jurídico Nacional.	Permite obtener oportunidades legales y evitar posibles infracciones a las leyes vigentes.
Matemática Discreta	Modelos lógicos con implicancia matemática.	Herramienta invaluable para el análisis de casos en los cuales la lógica es aplicable.
Modelos Numéricos	Modelización de escenarios reales a través de métodos matemáticos.	Esencial para la abstracción de la realidad y la aplicabilidad de modelos predefinidos para tal tarea.
Paradigmas de Programación	Aplicación de distintos paradigmas para la resolución de problemas aplicando software.	Contribuye con la aplicación de soluciones alternativas y más económicas para problemas complejos.
Práctica Profesional Supervisada	200 horas de aplicación en el ambiente laboral.	La práctica de campo afianza los conocimientos y permite mejorar la migración desde el mundo académico al laboral.
Probabilidades y Estadística	Fundamentos matemáticos que aplican a los modelos estadísticos. Captación del comportamiento aleatorio de la naturaleza y su medición a través de las probabilidades de ocurrencia de eventos.	Es uno de los pilares básicos de un simulador discreto, ya que su motor se basa en eventos aleatorios abstraídos de la realidad.
Procesamiento Digital de Imágenes	Aplicación de métodos, matemáticos y sistémicos, para obtener información a partir de imágenes.	Agrega valor para la interfaz de usuario.
Proyecto	Herramientas técnicas y administrativas para la formulación y gestión de proyectos.	Es la materia que sembró la semilla de este proyecto, y con su aporte se logró la concreción del mismo.
Química	Fundamentos básicos de la naturaleza desde el punto de vista de la química.	Amplía la capacidad técnica de modelado
Redes de Información	La interconexión entre computadoras y las distintas configuraciones que pueden adoptarse.	Contribuye a la implementación de la arquitectura cliente-servidor y permite el diseño de la red para el desarrollo del proyecto.
Simulación	Teoría y aplicación práctica de modelos de simulación.	Ofrece el conocimiento técnico para construir un simulador.
Sintaxis y Semántica del Lenguaje	Análisis y aplicación de lenguajes de programación basados en archivos de registros de longitud variable y planos.	El núcleo de un simulador carga los datos iniciales de una manera similar a la lectura de un archivo plano. Esta materia ayuda a definir los detalles de dicho proceso.

Materia	Descripción	Aportes
Sistemas de Gestión I	Definición de métodos analíticos para la resolución de problemas puntuales, como stocks, transporte y transbordo.	Estos modelos analíticos ayudan en la definición de las entidades del simulador.
Sistemas de Gestión II	Definición de procesos de producción y su impacto en la empresa. Mejora continua, kaizen, just-in-time.	Los modelos de proceso ayudan a crear planes de prueba para estos casos.
Sistemas de Representación	Representación gráfica de objetos en diferentes planos a través de láminas y del uso de autocad.	La representación gráfica es de ayuda para el diseño de salidas animadas.
Sistemas Informáticos Industriales	Aplicación práctica de los sistemas realimentados de control.	Colabora con el modelado de sistemas de producción complejos.
Sistemas Operativos	Análisis de los mecanismos de control y acción interno del software base de toda unidad de procesamiento.	Ofrece una amplia visión del procesamiento a muy bajo nivel del software y su interrelación con los periféricos.
Sistemas y Organizaciones	Teoría de la Organización y su perspectiva sistémica.	Permite la implementación de modelos organizacionales para la simulación.
Teoría de Control	Análisis de los sistemas realimentados de control, su origen y los criterios de estabilidad temporal.	Colabora con el análisis cuantitativo en el modelado de sistemas de producción complejos.

En cada una de las materias se dio una breve descripción con los temas más relevantes que hicieron a la cursada de la misma en su momento. El contenido actual de cada materia dependerá en gran medida del profesor que actualmente esté a cargo y del programa académico vigente.

El plan de estudios que rige para estas descripciones es el plan 95 con la adaptación del plan 2001.

Diagrama Esquemático del Marco Teórico Técnico del Proyecto

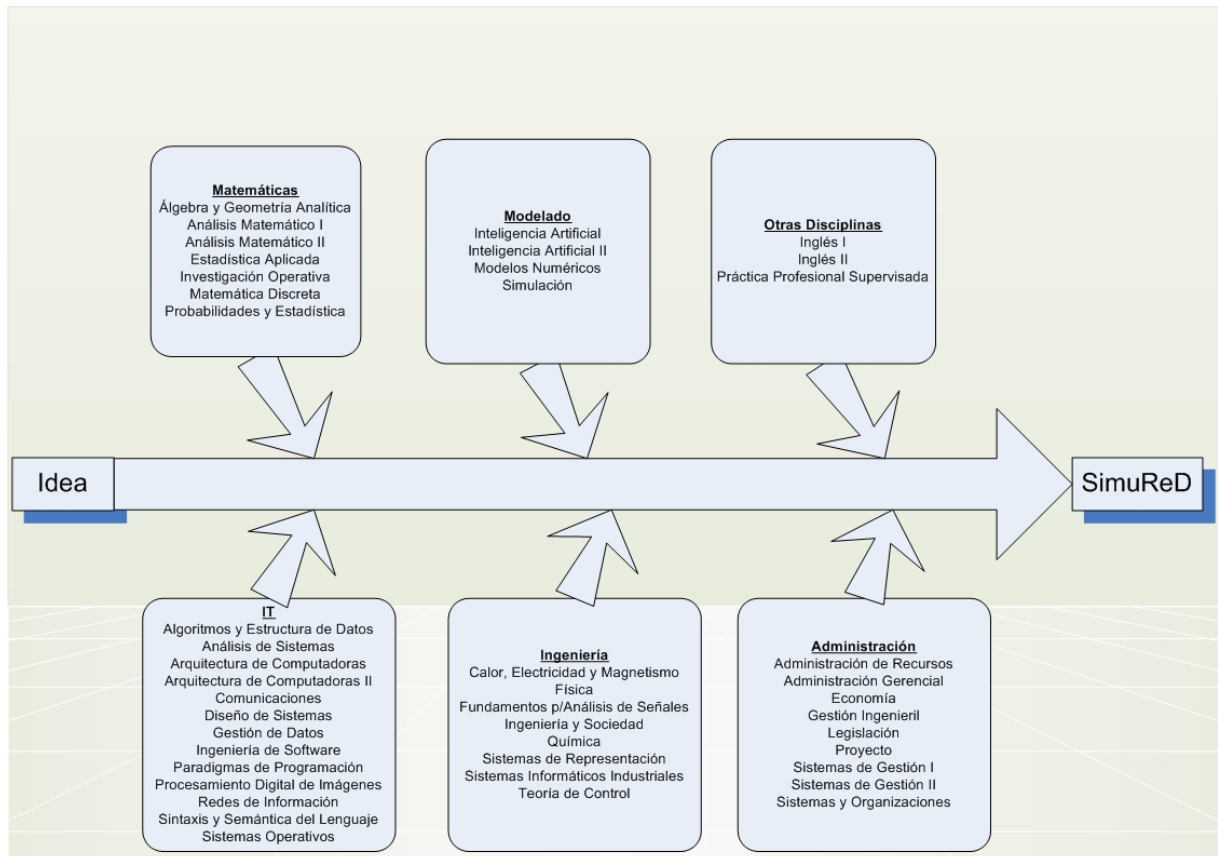


Figura 10.1 – Diagrama de contribución

ANEXOS – índice

ANEXO 2A – CARRERAS UTN	135
ANEXO 2B – DIFERENCIA ENTRE SIMULACIÓN Y OPTIMIZACIÓN	137
ANEXO 2C – DESCRIPCIÓN DE PRODUCTOS.	141
ANEXO 3A – SISTEMA OPERATIVO LINUX, DISTRIBUCIÓN CENTOS	159
ANEXO 3B – SERVIDOR WEB, APACHE TOMCAT	161
ANEXO 3C – VERSIONADO, CVS	163
ANEXO 3D – MYSQL	167
ANEXO 3E – FLASH & ACTIONSCRIPT	171
ANEXO 3F - WEB SERVICES – PROTOCOLO SOAP	173
ANEXO 4A – CAMPANA	177
ANEXO 4B – SANTA FE	185
ANEXO 4C – PLANO DE LA FACULTAD REGIONAL DELTA	189
ANEXO 4D – CAMPANA CIUDAD INTELIGENTE	191
ANEXO 6A – DESARROLLO DE LA TÉCNICA DE PUNTOS DE FUNCIÓN	193
ANEXO 6B – MÉTODO DE ESTIMACIÓN COCOMO	209
ANEXO 6C – DETALLES DEL SISTEMA DE BECAS DE LA UTN	217
ANEXO 7A - TIPOS DE PROYECTOS QUE FINANCIAN EL FONTAR	219
ANEXO 9A – LEY NACIONAL DE PROMOCIÓN DE LA INDUSTRIA DEL SOFTWARE	223
ANEXO 9B – EL SOFTWARE COMO ACTIVIDAD INDUSTRIAL	229
ANEXO 9C – REGLAMENTO DE BECAS	231
ANEXO 9D – LEY DE MARCAS	249
ANEXO 9E – REGISTRO DE DOMINIOS EDU.AR	251

Anexo 2A – Carreras UTN

Carreras Cortas

Acuicultura y procesamiento pesquero
 Administración
 Bromatología y medio ambiente
 Energía Renovable y Medio Ambiente
 Gestión Empresas Hoteleras
 Gestión Empresas Turísticas
 Higiene y Seguridad en el Trabajo
 Industrias Alimentarias
 Mantenimiento Industrial
 Mecatrónica
 Negociación de Bienes
 Periodismo
 Procesos industriales
 Programación
 Recupero Crediticio
 Seguridad Vial
 Sistemas Informáticos
 Supervisión de Procesos Industriales con

Orientación Petroquímica
 Tecnología de la Información

Carreras de Grado

Ingeniería

Aeronáutica
 Civil
 Eléctrica
 Electromecánica
 Electrónica
 Industrial
 Mecánica
 Metalúrgica
 Naval
 Pesquera
 Química
 Sistemas de Información
 Textil

Licenciatura

Administración Rural
 Organización Industrial
 Ciclos de Licenciaturas
 Ciencias Aplicadas
 Enología
 Enseñanza de la Física
 Enseñanza de la Matemática
 Enseñanza de la Química
 Gestión de Empresas Turísticas
 Gestión de la Producción Gráfica
 Higiene y Seguridad en el Trabajo

Lic. en Tec. Educativa (a distancia)
 Tecnología Educativa

Postgrados

Doctorado

Doctorado
 Doctorado en Ingeniería
 Especialización
 Control e Inspección de Construcciones

Soldadas

Higiene y Seguridad en el Trabajo
 Ingeniería Computacional
 Ingeniería en Mantenimiento
 Ingeniería Estructural
 Ingeniería Clínica
 Prep. y eval. de Proyectos
 Soldadura

Maestría y Especialización

Acústica Sonido
 Administración de Empresas
 Administración de Negocios
 Control Automático
 Cs de la Comp orient. B. D.
 Desarrollo Sust. Del Hábitad Humano
 Docencia Universitaria
 Estructura Sismorresistente
 Gestión de la Educación Superior
 Ingeniería Ambiental
 Ingeniería en Calidad
 Ingeniería en Sistemas de Información
 Radio Química
 Reactores Nucleares
 Siderurgia
 Simulación y Mecánica Computacional
 Aplicada al diseño en Ing.
 Sistemas de Información
 Tecnología Aeroespacial
 Tecnología en Alimentos
 Transporte y Logística

Trayectos de Articulación

Administración de Empresas
 Auditoría y Gestión Ambiental
 Desarrollo de Economías Regionales
 Educación Física
 Gestión de Negocios Agroalimentarios
 Lengua Inglesa
 Lic. en Comercialización
 Profesorado Universitario en Tec. Médica

Anexo 2B – Diferencia entre Simulación y Optimización

Artículo tomado del Newsletter de Perspectiva Logística. www.vaticgroup.com el 01/02/2008

V. A T L C.  G. R. O. U. P.

Simulación vs. Optimización

¿Cuál es la herramienta adecuada?

Para entrar en contexto, la *simulación* es la posibilidad de replicar situaciones reales o que podrían serlo para determinar estadísticamente su comportamiento. La *optimización* en cambio permite escoger entre una multitud de opciones la que más se adapte a las necesidades del problema. Aunque existen más consideraciones, a continuación presento algunas de las principales consideraciones que servirán de guía en el momento de escoger entre estas dos herramientas.

¿Quién no se ha visto enfrentado a situaciones problemáticas en nuestras organizaciones que parecen no tener una forma de análisis y respuesta evidente? ¿Quién no ha sentido que a veces las mejores prácticas no parecen tener el impacto que necesitamos o que nos exige la gerencia? Estas son algunas situaciones que pueden enfrentarse usando técnicas de Investigación de Operaciones. Sin embargo, debe hacerse un cuidadoso análisis de las opciones existentes en el mercado y la academia para no tomar una decisión apresurada.



En efecto, durante la última década, los casos de éxito a nivel mundial de la Investigación de Operaciones han generado un importante auge en su uso para la toma de decisiones estratégicas en logística. Esta tendencia o "moda" suscita una serie de interrogantes que deben ser estudiados con cautela antes de embarcarnos en un proyecto en este sentido. Claro está, un proyecto bien estructurado ofrecerá beneficios que compensarán ampliamente la inversión, pero este beneficio se obtendrá solamente si la herramienta que estamos empleando es la adecuada.

La simulación y la optimización son excelentes herramientas de toma de decisiones que cada vez son más usadas por todo tipo de organizaciones

A lo largo de esta serie de artículos de investigación de operaciones explicaremos algunas de las consideraciones que deben tenerse en el momento de determinar qué herramienta se ajusta a la problemática actual y a los resultados esperados. En esta edición, trataremos los puntos principales encontrados en la práctica y la academia que determinan en qué casos es más conveniente emplear Optimización o Simulación.

1.

Situación estática en el tiempo: dependiendo de si la problemática y las decisiones a tomarse no varían en el corto y mediano plazo, se tenderá a usar la *optimización*. Un ejemplo de problemática es cuando queremos definir el número de centros de distribución con los que debe operar nuestra organización. Esta situación se considera de tipo estática en el tiempo pues es una decisión que se tomará una sola vez en el corto y mediano plazo. Otros ejemplos incluirían decisiones como ubicación de plantas, determinación de precios (revenue management), determinación de la ubicación de los productos en el frontal de las góndolas.

2.

Inestabilidades y variabilidad en el sistema: cuando hay inestabilidades en el sistema (es decir el sistema no se estabiliza por efecto de las tasas de entrada y de salida por ejemplo) se debe usar *simulación* pues la optimización no lograría un resultado óptimo. En el caso del estudio de un centro de atención de un banco podría arrojar como resultado que se necesita un número muy elevado de cajeros si el sistema “explota”. Así mismo, la *simulación* se vuelve una valiosa herramienta cuando existen numerosas relaciones entre variables, y más aún cuando estas relaciones no son perceptibles. Un ejemplo de esta relación podría ser en manufactura la tasa de producción de una máquina al comienzo de la cadena que afectará sin duda el throughput del final de la cadena. Otro ejemplo de esta situación podría ser cuando en su empresa buscan implementar un sistema Kanban con todas las de la ley. Debido al número tan elevado de interacciones de este sistema, sólo han podido lograrse aproximaciones muy simplificadas del modelo a través de métodos como optimización, por eso la simulación resulta una muy buena opción. Un último ejemplo de esta situación es el manejo de inventarios de la industria automotriz donde una multitud de variables inciden sobre el manejo de los inventarios (algunas: manejo del efecto látigo, pluralidad de orígenes y destinos,...).

3.

Varios objetivos esenciales: cuando una situación debe resolverse siguiendo varios objetivos (Ej.: objetivos financieros, recursos, eficiencia, productividad y calidad) que en muchos casos son contradictorios es preferible usar *simulación*. Aunque existen métodos sofisticados de optimización (multiobjetivos o mediante algoritmos genéticos con su respectiva función de “fitness”), estas técnicas corren un alto riesgo de obtener un resultado incorrecto por la forma como se parametrizan. Incluso, en muchos casos no es posible definir de manera sencilla qué objetivo tiene mayor prelación y en que proporción en comparación con los demás objetivos. Es por esto que se logra una mejor visualización de la situación usando *simulación*. (Incluso puede pensarse en combinar simulación y optimización de manera combinada como lo veremos más adelante en el artículo). Ejemplos de esto podrían ser las necesidades contradictorias de servicio al cliente y reducción de costos en servicios de atención al cliente como call centers y sucursales bancarias. Otro caso típico en sistemas de producción es la clásica disyuntiva entre tasa de producción y material en proceso, donde una mayor tasa de producción incrementa el producto en proceso. Idealmente se quería minimizar el trabajo en proceso y maximizar la tasa de producción, lo que los hace variables que cambian de forma indirectamente proporcional.



Simulación de red de distribución en Flexsim

Las características del problema definen de manera conclusiva cual de las dos herramientas es la más adecuada. Es importante no tomar una decisión a la ligera.

4.

Funciones objetivo y restricciones no lineales: cuando la función objetivo y las restricciones no son lineales la *simulación* es una excelente herramienta que permite evitar las dificultades que presenta resolver este tipo de complejidades a través de la optimización. Algunos incluso no pueden resolverse usando optimización y deben reformularse para conseguir un planteamiento “lineal”. Algunos ejemplos son la optimización de portafolios de inversión, en los cuales hay funciones objetivo que consideran el VAR e incluso otras medidas de riesgo distintas y no lineales, o también restricciones de este estilo ante un objetivo de maximización del retorno.



Simulación en Flexsim de bodega

5.

Falta de datos: en los casos en los que en su organización no existan datos generosos del sistema,

puede ser una mejor opción la *simulación* puesto que puede emplear distribuciones como la triangular o la Beta que aunque no le darán un resultado de la mayor precisión, sí podrán darle una idea del funcionamiento del sistema de forma tal que pueda definir y evaluar políticas. Es importante aclarar que estas deben ser utilizadas con mesura y teniendo en cuenta siempre un adecuado análisis de sensibilidad a los parámetros en el sistema.

6. **Situaciones de asignación:** siempre y cuando no estemos en uno de los casos anteriores, los problemas de asignación obtienen grandes beneficios al ser tratados con **optimización**. El uso de esta herramienta le asegura obtener un resultado óptimo que hará más eficiente el uso de sus recursos ya sea desde el punto de vista de costos y utilización entre otros. De este tipo de problemas existen una gran variedad que incluyen algunos de los siguientes que se han trabajado por esta vía: asignación de tamaño de flotas, turnos en callcenters, tripulación a flotas aéreas, cargas de trabajo en el balanceo de línea en un sistema de producción o en un sistema de procesos documentales (como lo es el caso de una aduana).

7. **Estudio de escenarios y contingencias:** este tipo de situaciones pueden mirarse desde dos puntos de vista según sus características. Una es mediante optimización en cuyo caso se evalúan sensibilidades del planteamiento. Es decir se estudia qué tanto varía la respuesta óptima ante el cambio en cuestión. La otra opción es usar simulación para comparar escenarios alternativos cuando se tiene noción de que una de las dos opciones es un óptimo, pues de lo contrario se llegaría a un punto subóptimo.

En ciertos casos como el estudio de escenarios y contingencias la decisión es más difícil pues se puede usar cualquier de las dos herramientas según las condiciones de problema obteniendo respuestas por enfoques diferentes.

Esta guía da una idea de cuándo usar una u otra herramienta, sin embargo el uso adecuado de cada herramienta dependerá en gran medida de las condiciones específicas del problema a estudiarse. Así mismo, una técnica muy interesante para nuestras organizaciones es la que combina simulación con optimización. Esto ayuda en gran medida a mejorar los resultados que podrán obtenerse.

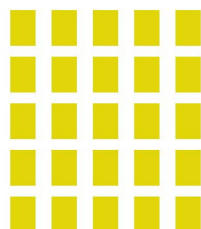


Figura 1: Espacio factible

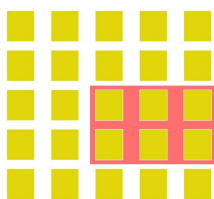


Figura 2: Optimización evalúa espacio factible

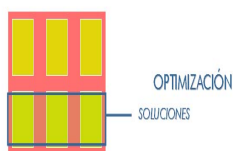


Figura 3: Optimización obtiene soluciones de la frontera eficiente

Por ejemplo piense en el caso de la determinación de tamaño de flota para operaciones de distribución. A priori, parecería ser una situación de asignación. Sin embargo, en la operación diaria de la flota, existen imprevistos y variabilidades como lo son la variación de la demanda a ser transportada, accidentes, cierre de vías, entre otras situaciones que pueden poner en entredicho la asignación obtenida por optimización. Piense ahora en las bondades de tener la posibilidad de ejecutar un modelo de simulación que toma sus parámetros de la optimización.

En la figura 1, podemos esquematizar el espacio existente del problema. En el caso del ejemplo, obtendríamos las posibles flotas para la organización. Por ejemplo entre 100 y 200 vehículos con cierta secuenciación cada una.

Como puede verse en la figura 2, la optimización corre sobre el espacio factible, es decir sobre las flotas que cumplen con las restricciones del problema.

En el momento de correr la optimización esta corre sobre las flotas factibles y se obtienen soluciones óptimas que hacen parte de la frontera eficiente. De manera sencilla, es un conjunto de puntos que son óptimos del problema. Como se ve resaltado en la figura 3, esta zona corresponde a algunos

puntos no necesariamente consecutivos. Un error frecuente es pensar que un problema de optimización sólo llega a una solución única. Siguiendo con nuestro ejemplo esto podría ser 150 y 162 vehículos.

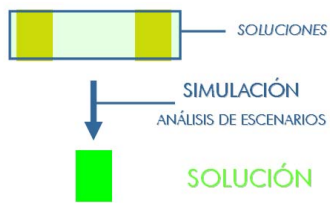


Figura 4: Muestra repercusiones de esquinas en los crossdocks

Pero entonces, ¿cómo seleccionar entre estas soluciones "óptimas"? Existen varias metodologías de las cuales la más recomendable es analizar cada una teniendo en cuenta parámetros estratégicos y de costos de operación, niveles de servicio y costos de oportunidad en el caso del tamaño de flota del ejemplo. Sin embargo en el caso del ejemplo, la determinación del tamaño óptimo de flota implica tener en cuenta un sinnúmero de variables e interacciones entre estas (ejemplo las que veíamos antes, cierres de vías, accidentes, ...). Por eso aplicamos simulación, tomando cada solución como un escenario como puede verse en la figura 4. De esta manera

obtenemos un análisis muy completo que puede incluso tener en cuenta contingencias como lo son cierres de vía.

Con este esquema de solución podrán lograrse los beneficios de cada herramienta, pudiendo obtener un resultado más preciso. Esta combinación muestra como la investigación de operaciones es una disciplina que permite un gran apoyo en la toma de decisiones, y la empresa o persona experta en este tema que lo asesore empleará la o las herramientas adecuadas de acuerdo a su problemática.

En conclusión, las características del problema darán en buena medida una idea de si la herramienta más conveniente es la simulación o la optimización. Sin embargo como usted se dará cuenta, la decisión de que herramienta utilizar depende en gran medida de las condiciones precisas de la situación. Además, como podrá ver existe una amplia gama de posibilidades que deben ser evaluadas para decidir con cada herramienta cual es la forma más adecuada de emplearla. Actualmente en el mercado existen paquetes de última generación tanto en simulación como en optimización que permitirán a su organización sacar provecho de la investigación de operaciones. Así mismo, una opción que siempre está disponible a nuestras organizaciones es contratar consultores expertos en el uso de estas herramientas para facilitar el proceso de aprendizaje y reducir los costos de emprender un proyecto de esta envergadura.

Anexo 2C – Descripción de productos.

GPSS

Origen: Suiza 

GPSS significa Sistema de Simulación de Propósito General (del inglés General Purpose Simulation System) y fue creado por Geoffrey Gordon. GPSS es un lenguaje de simulación de tiempo discreto donde el tiempo de la simulación avanza por pasos definidos. Un sistema es modelado a través de una red en la cual las transacciones generadas son pasadas desde un servicio (representado mediante bloques) a otro. Este sistema es particularmente bueno para problemas como producción.

GPSS fue muy popular en los finales de la década del 60 y actualmente es muy poco usado. GPSS es menos flexible que otros lenguajes de simulación, como Simula o SimScript.

WebGPSS es la última versión de GPSS, y está disponible en Internet y en CD una versión para Windows. WebGPSS está intentando posicionarse para ser el comienzo en el aprendizaje de simulación. GPSSWeb ha sido usado en varios colegios y universidades en el Norte de Europa y también en algunas universidades de Estados Unidos, como herramienta de aprendizaje en cursos de negocios e Ingenierías.

La versión académica básica en CD tiene un costo de 40 Euros por cada persona que lo utilice, con un mínimo de 100 Euros para una entrega.

Los modelos que corren bajo esta versión pueden contener hasta 150 bloques y 1500 transacciones simultáneamente. Esto permite realizar varios proyectos a los estudiantes.

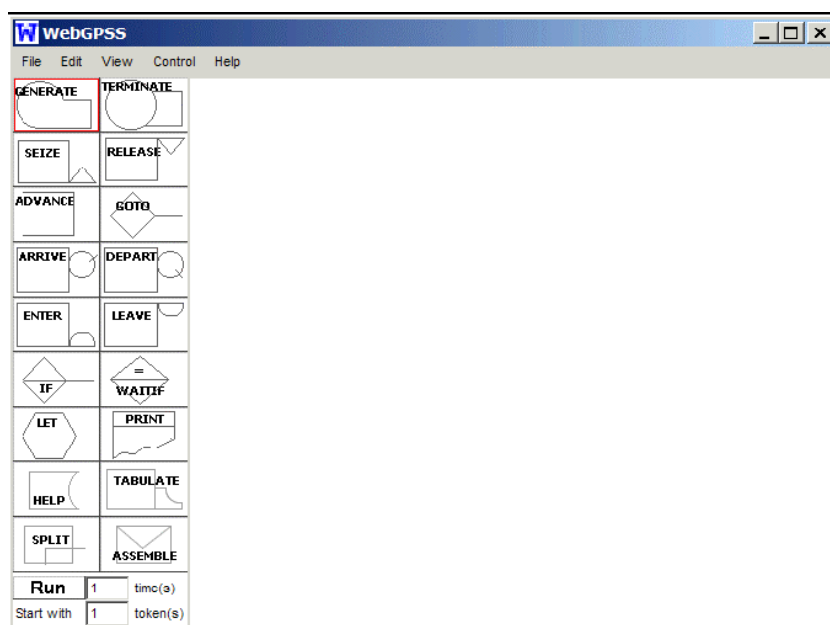


Figura 1 - Pantalla principal de GPSS

En el sitio oficial (www.webgpss.com) hay documentación con tips para docentes y alumnos. Como así también documentación relacionada, todo en Inglés.

Vensim Simulation Software

Origen: USA 

Es comercializado por Ventana Systems que ofrece productos y servicios para resolver problemas y mejorar el desempeño organizacional. Esta empresa comercializa Vensim que se utiliza para la construcción de modelos de simulación de negocios, científicos, ambientales y sistemas sociales.

Vensim es una herramienta gráfica de creación de modelos de simulación que permite conceptualizar, documentar, simular, analizar y optimizar modelos de Dinámica de Sistemas. Vensim proporciona una forma simple y flexible de crear modelos de simulación, sean con diagramas causales o con diagramas de flujos.

Las relaciones entre los elementos del sistema representan las relaciones causales, que se muestran mediante la conexión de palabras con flechas. Esta información se usa después por el Editor de Ecuaciones para crear el modelo de simulación. Se puede analizar el modelo en el proceso de construcción teniendo en cuenta las causas y el uso de las variables, y también estudiando los ciclos relacionados con una variable. Mientras que se construye un modelo que puede ser simulado, Vensim permite explorar el comportamiento del modelo.

Vensim es compatible con Windows 95/98/NT/2000/XP/Vista y Power Macintosh System 7 o superior (modo Clásico en OSX). Vensim requiere cerca de 20 Mb de espacio de disco para una instalación completa, pero la ejecución de un modelo simple puede requerir unos 100 Mb de espacio en el disco. El uso de la memoria es similar, requiriendo mayor cantidad cuando el modelo aumenta.

Vensim PLE (Personal Learning Edition) es una versión que ha sido diseñada para iniciarse en el modelado. Es gratis para uso académico y tiene licencia shareware para uso comercial. Esta versión está limitada en opciones de menú y funciones complementarias, centrando la atención en el modelado.

Ventana System también implementó un programa académico que ofrece precios especiales sobre sus productos comerciales a entidades educativas o personas que acrediten ser miembros de instituciones educativas. La versión mas económica es la PLE Plus con un valor de US\$ 89.-¹ que incluye un año de mantenimiento, soporte técnico, documentación electrónica y la posibilidad de extender la licencia a 18 meses. La renovación de esta licencia por un año más tiene un valor actual de US\$ 49.-¹³.

¹ Valores al 18 de Junio de 2008 (ref.: \$ 3,14 (ARG) = \$ 1.- (USA))

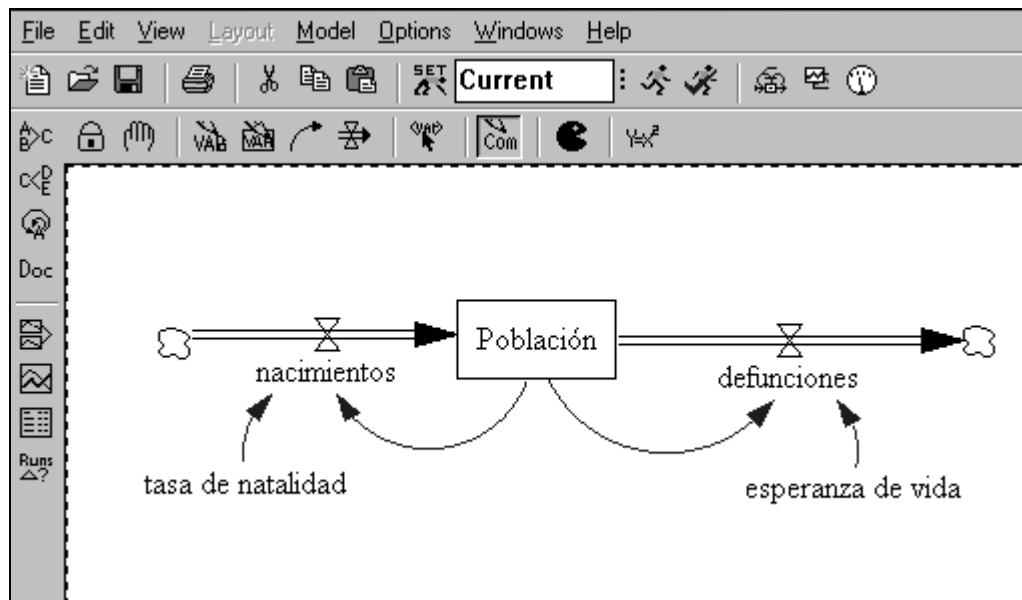


Figura 2 - Vista de la pantalla de diseño de Vensim

Arena – Rockwell

Arena es comercializado por Rockwell Software Inc. que es una división de Rockwell Automation Control Systems, empresa Norteamericana líder a nivel mundial en el desarrollo e implementación de hardware y software de automatización y simulación.

El Software Arena es la herramienta de simulación de eventos discretos y continuos líder a nivel mundial.

Arena está diseñado para su uso en toda una empresa, apoya el análisis en profundidad de una determinada área funcional (por ejemplo, producción, logística, servicio al cliente) y el análisis de los procesos que abarcan varias áreas funcionales. Arena utiliza una metodología de modelado denominada Templates, esta metodología es propia y consiste en la construcción de diagramas de flujo utilizando objetos denominados módulos. Una colección de módulos define un template.

Arena fue oficialmente reconocido como líder del mercado en la Winter Simulation Conference, realizada todos los años en Florida, Estados Unidos. En el año 2006 contaba con casi el 50% del mercado total.

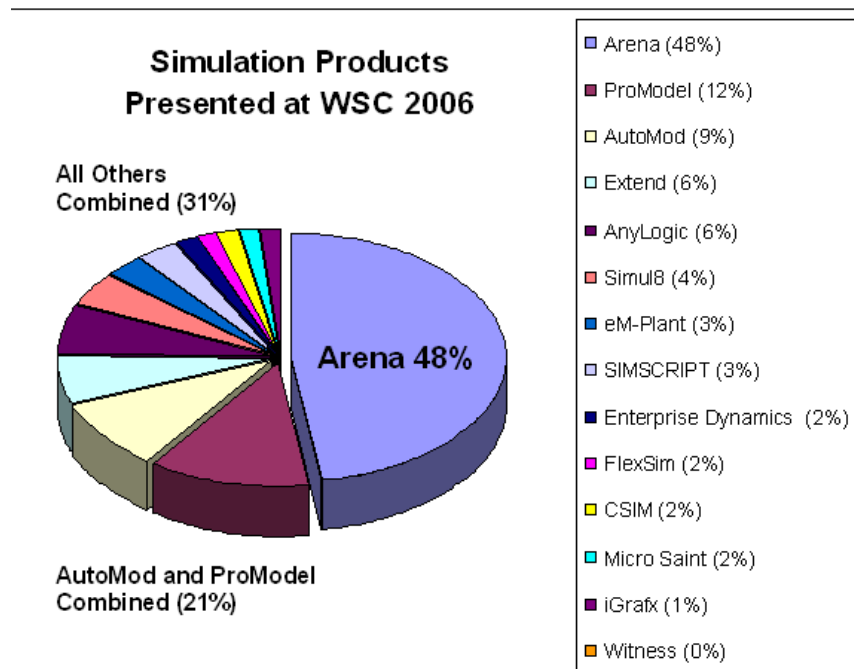


Figura 3 – Porcentaje de ocupación del mercado en 2006

La característica distintiva de Arena, es la arquitectura basada en templates. Si bien esto es una gran ventaja para el uso profesional, no favorece la comprensión y la práctica para la generación del modelo, ya que para crear el modelo se debe adaptar alguno de los disponibles (sólo las versiones profesionales de Arena cuentan con la posibilidad de crear templates).

MATLAB

Origen: USA 

MATLAB es la abreviatura de *MATrix LABoratory* (laboratorio de matrices). Se trata de un software matemático muy versátil que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M). Está disponible para las plataformas Unix, Windows y Apple Mac OS X.

Entre sus prestaciones básicas se hallan: la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos hardware. El paquete MATLAB dispone de dos herramientas adicionales que expanden sus prestaciones, a saber, Simulink (plataforma de simulación multidominio) y GUIDE (editor de interfaces de usuario - GUI). Además, se pueden ampliar las capacidades de MATLAB con las *cajas de herramientas (toolboxes)*; y las de Simulink con los *paquetes de bloques (blocksets)*.

Es un software muy usado en universidades y centros de investigación y desarrollo. En los últimos años ha aumentado el número de prestaciones, como la de programar directamente procesadores digitales de señal o crear código VHDL.

Simulink puede simular cualquier sistema que pueda ser definido por ecuaciones diferenciales continuas y ecuaciones diferenciales discretas. Esto significa que se puede modelar sistemas continuos en el tiempo, discretos en el tiempo o sistemas híbridos.

Posee una interfaz de usuario gráfica (GUI), con diagramas de bloques para construir los modelos utilizando operaciones con el ratón del tipo pulsar y arrastrar. De esta forma, podemos dibujar los modelos de la misma forma que lo haríamos con lápiz y papel. Esto supone un cambio radical respecto a los anteriores paquetes de simulación. Una vez construidos los diagramas de bloques, podemos ejecutar simulaciones y analizar los resultados, también de forma gráfica.

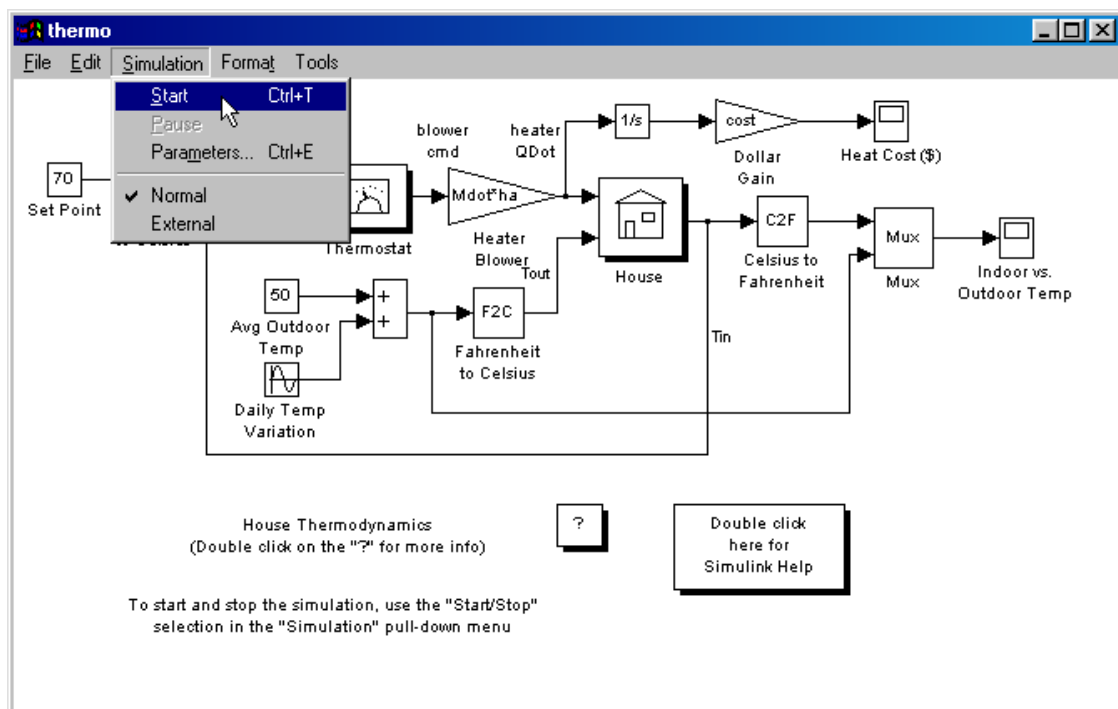


Figura 4 - Vista de la pantalla de diseño de Simulink

En SIMULINK podemos encontrar una amplia biblioteca de bloques de sumideros, fuentes, componentes lineales y no lineales y conectores. También podemos personalizar y crear nuestros propios bloques.

Los modelos son jerárquicos, de forma que podemos ver un sistema desde un nivel superior y entrando en los bloques podemos ir descendiendo a través de los niveles para ver con más detalle el modelo.

Powersim

Origen: Noruega 

Powersim es el resultado de más de 15 años de trabajo. La primera versión de este *software*, SimTek, se lanzó en el año 1987.

El lenguaje gráfico de modelado de Powersim hace muy sencillo seguir y explicar a otros los modelos de Dinámica de Sistemas. En los modelos de Powersim Studio, las variables se presentan mediante iconos conectados mediante flechas que representan flujos de información o de materiales. La estructura transparente de estos modelos constituye una de sus principales ventajas.

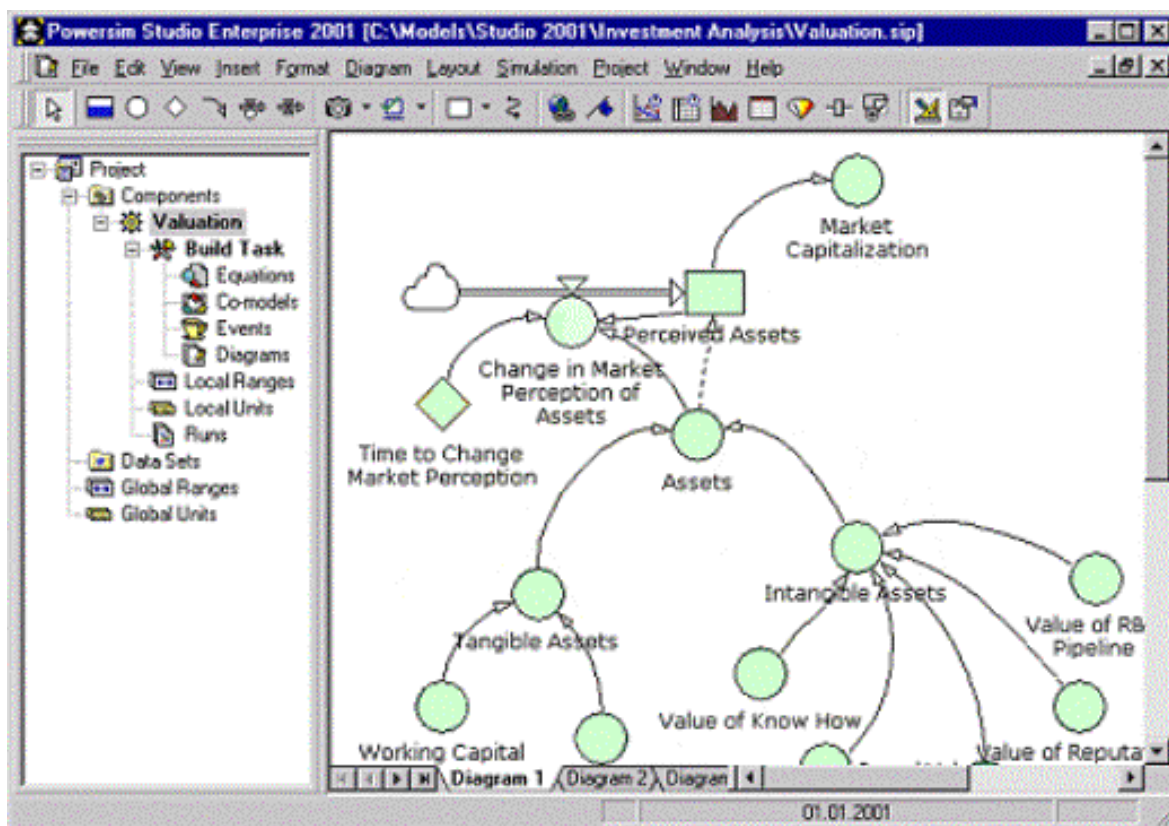


Figura 5 - Vista de la pantalla de diseño de Powersim

Desde el manejo de una operación específica hasta las estrategias que le dan forma a las empresas, quienes toman decisiones utilizan modelos todos los días para la toma de decisiones. Cuando nos enfrentamos a un problema, utilizamos nuestros modelos mentales para formular hipótesis sobre cual cadena de causa y efecto podemos utilizar para alcanzar el resultado deseado. La complejidad de nuestros "modelos mentales" exceden nuestra capacidad para anticipar las implicaciones de nuestras decisiones basadas en estos modelos.

La tecnología Powersim posee las capacidades para modelar cualquier tipo de problema complejo, y le provee de un acceso de fácil uso a sus datos históricos. Estos modelos son explícitos y transparentes, utilizando objetos gráficos que permiten una fácil inspección y verificación.

Las simulaciones basadas en la tecnología Powersim son típicamente utilizadas en Soporte a la toma de Decisiones, Formación Gerencial y Comunicación de Valor en niveles estratégicos, gerenciales y operacionales de una organización.

Características para Usuarios Académicos

- Mayor funcionalidad, facilidad en el uso y accesibilidad hacen de Powersim una opción excelente para instituciones de investigación y educación acreditadas.
- La minuciosidad y claridad conceptual hacen de Powersim una poderosa herramienta para enseñar dinámica de sistemas en todos los niveles.
- La simulación es además una excelente herramienta para la investigación y educación en otras disciplinas, como por ejemplo negocios o investigaciones como gobierno o corporaciones.
- La perspectiva holística de los modelos de simulación se prestan por sí solos para proyectos que abarquen áreas multidisciplinarias.

Powersim se comunica directamente con Microsoft Excel y SAP SEM (Strategic Enterprise Management, módulo de gestión estratégica de empresas de SAP). Esto permite importar datos reales de negocio y exportar escenarios futuros. También es posible la comunicación con cualquier otra base de datos a través de Powersim Studio SDK.

Es muy sencillo crear interfaces amigables directamente con Powersim, incluso para no programadores. Además, es posible exportar estos modelos en Internet. Hay varias versiones de Powersim con diferentes capacidades de modelado y de conectividad.

Promodel

Origen: USA 

Creado y comercializado por ProModel Corporation se nombra como el líder en el mercado de simulación de procesos concernientes a la salud.

ProModel es un simulador con animación para PC. Permite simular cualquier tipo de sistemas de producción, logística, manejo de materiales, del tipo discreto. Este software es un paquete de simulación que no requiere programación para realizar modelos simples, aunque si es necesario está disponible programar ciertos comportamientos. Tiene la combinación perfecta entre facilidad de uso y flexibilidad para aplicaciones complejas.

Una vez hecho el modelo, éste puede ser optimizado para encontrar los valores óptimos de los parámetros claves del modelo. Algunos ejemplos incluyen determinar la mejor combinación de factores para maximizar producción, minimizando costo, minimizar el número de camiones sin penalizar el servicio, etc. El módulo de optimización, ayuda a encontrar rápidamente la solución óptima, en lugar de solamente hacer prueba y error. ProModel cuenta con 2 optimizadores disponibles y permite de esta manera explotar los modelos de forma rápida y confiable.

- Único software de simulación con Optimización plenamente integrada
- Creación de modelos rápida, sencilla y flexible.
- Modelos optimizables.
- Elementos de Logística, Manejo de Materiales, y Operaciones incluidas. (Bandas de transporte, Grúas Viajeras, Operadores).
- Entrenamiento en Español.
- Resultados probados.
- Importación del Layout de Autocad, y cualquier herramienta de CAD / CAE / Diseño, así como de fotografías digitales.
- Soporte Técnico 24 horas al día, 365 días del Año.
- Integración a Excel, Lotus, Visual Basic y herramientas de Microsoft.
- Genera en automático las gráficas en 3 dimensiones para visualización en el espacio tridimensional.

Promodel ofrece un sistema de licencias orientado al uso académico. En este caso el administrador de estas licencias es la empresa Belge que se encuentra en San Pablo (Brasil) y es el único representante de ProModel para el MERCOSUR, la cual nos cuenta que las licencias de uso académico disponibles en Argentina no son las mismas que las publicadas para USA y Canadá, sin embargo los precios de las licencias para Universidades son mucho mas baratas que las licencias comerciales. Para nuestro mercado tienen otro precio de lista y hay que pagar, además, impuestos de importación y el envío, que se realiza por Fedex.

Las licencias publicadas, para USA y Canadá del Student Package, consta del envío del software en CD tiene un valor de US\$ 30.-, al cual puede agregarse un manual de usuario pagando un importe adicional.

El software provisto tiene limitaciones de uso en cuanto a los modelos soportados. Las capacidades limitadas para esta versión son:

- 20 locaciones
- 8 entidades
- 8 recursos
- 5 atributos
- 15 macros

20Sim

Origen: Países Bajos

20-SIM es un programa de modelaje y simulación que funciona en la plataforma de Windows. Con 20-SIM, se puede simular el comportamiento de sistemas dinámicos, como sistemas eléctricos, mecánicos, hidráulicos y cualquier combinación de estos.

20-SIM soporta completamente el modelaje grafico, de manera que se pueden diseñar y analizar sistemas dinámicos de manera muy intuitiva, sin tener que comprometer la potencia

del equipo. 20-SIM también funciona con el uso de componentes, lo cual le permite modelar sistemas como si fuesen bosquejos de problemas de ingeniería. Al escoger los componentes que necesita de librería, los podrá concatenar para reconstruir su diagrama de ingeniería, sin tener que escribir una sola línea de matemáticas.

20-SIM es compatible y corre con mayor eficiencia sobre la plataforma Windows Vista. Existe la posibilidad de cambiar el lenguaje utilizado en la interfaz. También tiene la particularidad que los modelos pueden tener múltiples implementaciones.

Las graficas adicionales que soporta poseen sus propios menús y botones. Tiene la posibilidad de utilizar el comando Export, que permite exportar gráficos, parámetros, archivos de datos, etc.

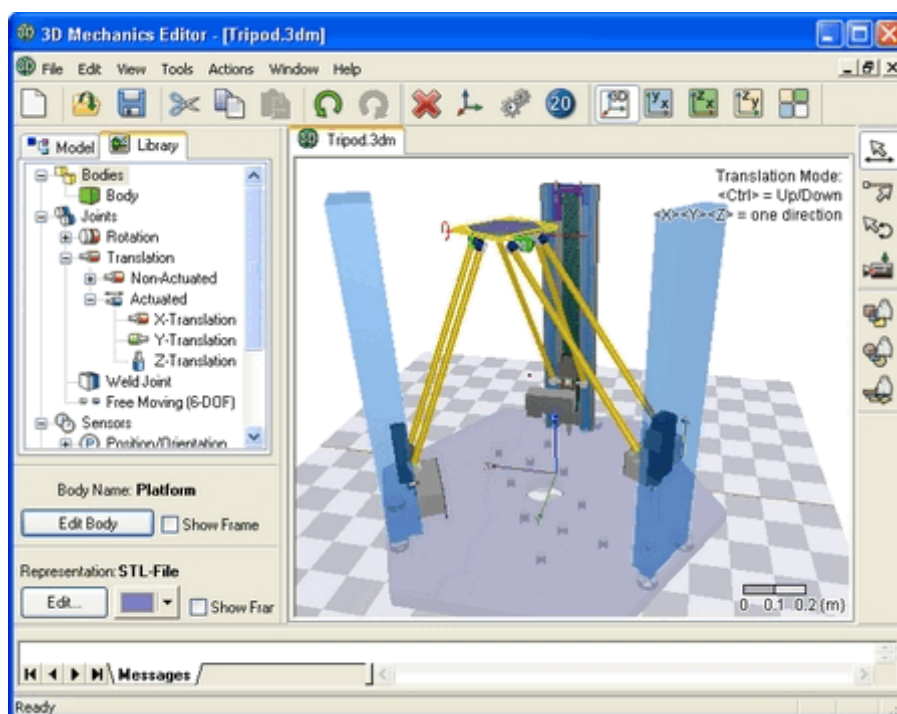


Figura 6 - Vista de la pantalla de resultados de 20Sim

20Sim tiene una versión académica denominada Classroom Kit, el cual se puede obtener una licencia de la versión profesional o de la versión estándar con ciertas restricciones a un precio reducido². Esta licencia cubre a todos los miembros de la institución y puede ser instalado en todas las máquinas de la institución, como así también en los hogares de los miembros.

Las restricciones impuestas son 2, el uso está limitado a uso en laboratorios y las licencias caducan al año de haberse adquirido.

² El valor del Classroom Kit de la versión profesional es de 4000 Euros, y el de la estándar es de 2000 Euros.

DEVS JAVA

Origen: USA 

DEVS, que viene del inglés para referirse a Discrete Event System Specification (Especificación de Sistemas de Eventos Discretos), es un formalismo para el modelado y análisis de sistemas discretos.

El término es ahora estándar en el campo de la Simulación para referirse a un formalismo modular y jerárquico para modelar y analizar sistemas de diversos tipos. En particular, sistemas de eventos discretos, sistemas de ecuaciones diferenciales (o sistemas continuos) y sistemas híbridos continuos y discretos.

El formalismo DEVS fue inventado por el Prof. Bernard P. Zeigler, actualmente trabajando en la Universidad de Arizona, y fue presentado al público en su primer libro Teoría de Modelado y Simulación en 1976 cuando era profesor adjunto en la Universidad de Michigan.

DEVS puede considerarse como una extensión de un autómata añadiendo el valor real en base al tiempo, el concepto modular con entrada y salida de eventos y el concepto jerárquico con la operación de acoplamiento.

Zeigler propuso un algoritmo jerárquico para un modelo de simulación DEVS en 1984 que se publicó en la revista Simulation Journal en 1987. Desde entonces, muchos han ampliado el formalismo DEVS introduciendo sus propios fines: DESS / DEVS combinó los sistemas de eventos continuos y discretos, DEVS-P para modelos paralelos de DESS, G-DEVS para el modelado lineal de trayectorias de piezas, RT-DEVS para desarrollos en tiempo real, y otras tantas versiones que resultaron de la original. Además de estas extensiones, hay algunas subclases como SP-DEVS y FD-DEVS que han sido desarrollados para investigar propiedades de sistemas específicos.

Debido al modelado jerárquico y modular, así como también por su simulación basada en el análisis de la capacidad, el formalismo DEVS y su variantes se han utilizado en muchas aplicaciones de ingeniería (como el diseño de hardware, software, sistemas de comunicación, sistemas de fabricación) y en la ciencia (como la biología y la sociología).

DEVS JAVA, es un paquete desarrollado en el popular lenguaje de programación Java, en el cual se implementan gran parte de los formalismos originales de DEVS.

Excel

Origen: USA 

Excel es un utilitario desarrollado y comercializado por Microsoft el cual es muy usado sobre todo en tareas administrativas.

Las herramientas con que cuenta este software son muchas, y abarcan un gran campo de acción. Entre otras, tiene una amplia variedad de generadores de números aleatorios y distribuciones probabilísticas, con los cuales pueden crearse modelos matemáticos.

Otra característica interesante es que pueden crearse macros. Las macros son fragmentos de código (en Visual Basic) que pueden asignarse a diferentes objetos o celdas dentro de cada planilla.

A partir de estas macros pueden desarrollarse aplicaciones que pueden ser fácilmente compartidas con otros usuarios de Excel. Como ejemplo de éstos tenemos el caso del docente de la Regional Córdoba que nos comentó de un desarrollo propio. También tenemos el caso de Simulación 4.0, que fue desarrollado por un profesional de la Universidad Nacional de Tucumán.

ExtendSim

ExtendSim es una poderosa herramienta de simulación, con el cual se pueden desarrollar modelos dinámicos de procesos reales en una amplia variedad de campos. Puede ser usado para crear modelos de bloques, explorar los procesos que intervienen, y ver cómo se relacionan, para finalmente, llegar a una solución óptima.

ExtendSim es fácil de utilizar y ayuda a entender sistemas complejos para producir mejores resultados con mayor rapidez. ExtendSim permite simular cualquier sistema o proceso mediante la creación de una representación lógica a través de sus módulos disponibles en un formato fácil de utilizar.

La salida puede ser animada, en 2D o 3D y posee la capacidad de ajustar la configuración mientras la simulación se encuentra corriendo. Una arquitectura robusta le añade características de escalabilidad en el sistema de simulación. Por medio de ésta, se puede pasar de un modelo continuo a uno discreto en tiempo de ejecución. También posee una herramienta por la cual pueden reutilizarse distintos bloques previamente armados, estos bloques pueden trabajar en conjunto a través del paso de mensajes entre ellos. Por otro lado es compatible con diversos lenguajes de programación, entre ellos C, Delphi, VB.

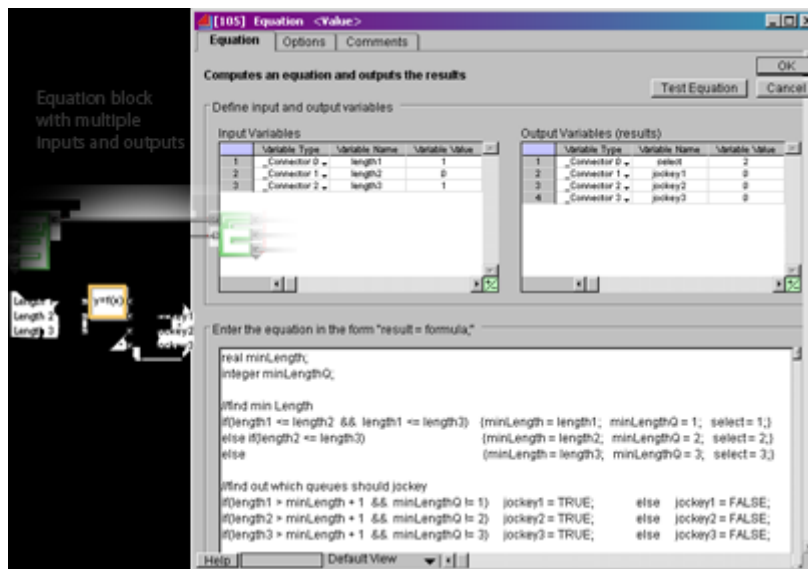


Figura 7 - Vista de la pantalla de Extend

Las licencias disponibles de Extend están catalogadas por niveles de uso. En los niveles mas altos se pueden llegar a crear modelos por separado, para luego integrarlos en una aplicación final, esto contribuye al trabajo en equipo. También permite la modificación de los bloques que el sistema trae incorporado. Otra característica interesante es la creación de funciones secuenciales para reutilizarlas en algún proyecto de gran envergadura.

Bajo la licencia ASP, el usuario está obligado a distribuir la funcionalidad desarrollada a otros usuarios, de esta manera se fomenta el uso del producto y aumenta la cantidad de bloques disponibles.

Octave

Origen: USA 

Octave o GNU Octave es un programa libre para realizar cálculos numéricos. Como indica su nombre es parte del proyecto GNU. MATLAB es considerado su equivalente comercial. Entre varias características que comparten se puede destacar que ambos ofrecen un intérprete permitiendo ejecutar órdenes en modo interactivo. Nótese que Octave no es un sistema de álgebra computacional como podría ser Máxima, sino que usa un lenguaje que está orientado al análisis numérico.

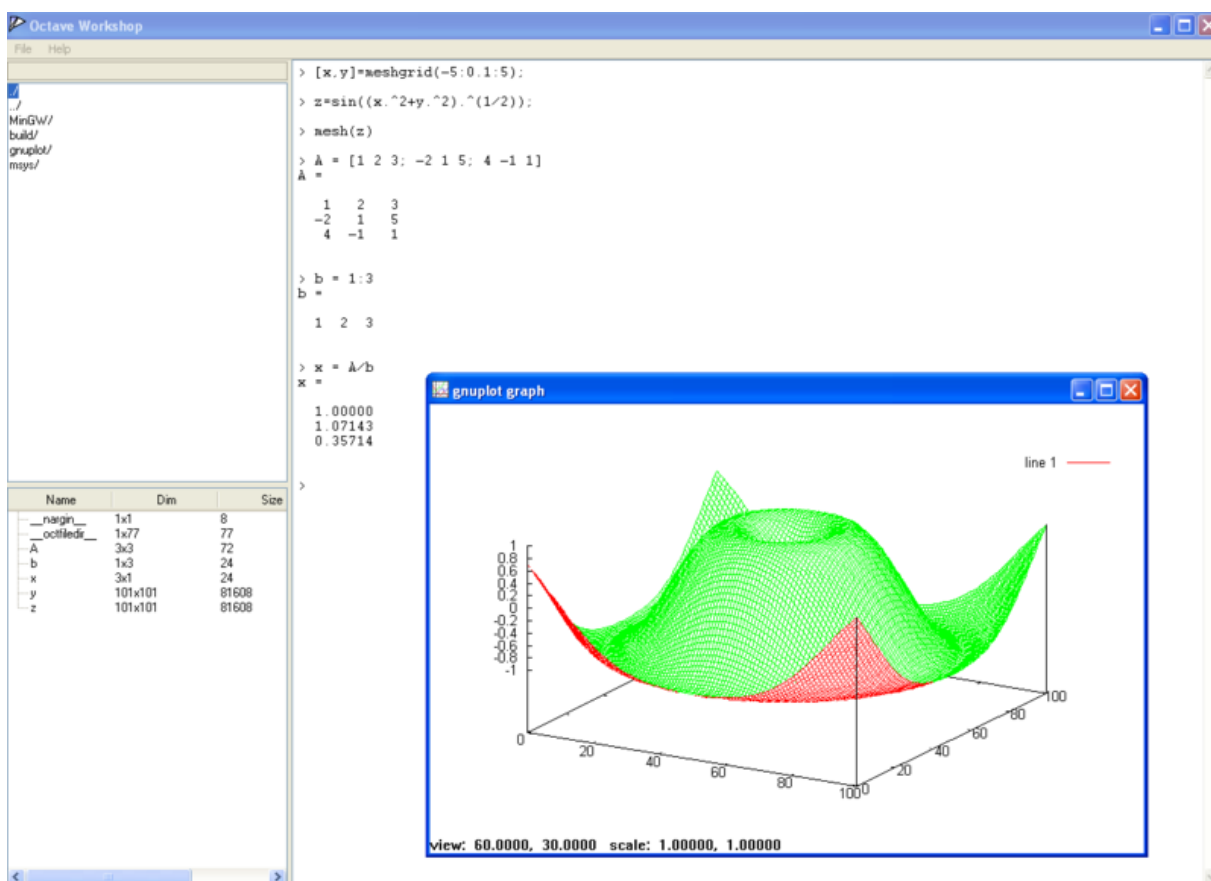


Figura 8 - Vista de la pantalla de resultados de Octave.

El proyecto fue creado alrededor del año 1988, en la Universidad de Wisconsin, pero con una finalidad diferente: ser utilizado en un curso de diseño de reactores químicos. Posteriormente en el año 1992, se decide extenderlo y comienza su desarrollo a cargo de John Eaton. La primera versión alpha fue lanzada el 4 de enero de 1993. Un año más tarde, el 17 de febrero, 1994 aparece la versión 1.0. El nombre surge del nombre de un profesor de unos de los autores conocido por sus buenas aproximaciones por medio de cálculos mentales a problemas numéricos.

Octave tiene un intérprete de su propio lenguaje (de sintaxis similar a Matlab), y permite una ejecución interactiva o por lotes. Puede extenderse el lenguaje con funciones y procedimientos por medios de módulos dinámicos. Utiliza otros programas GNU para ofrecer al usuario crear gráficos para luego imprimirlos o guardarlos (Grace). Además de correr en plataformas Unix también lo hace en Windows. Reconoce el formato de archivos de Matlab de extensión .m.

Dassault Systemes

Origen: Francia 

Esta empresa posee varios sistemas de modelado y es el principal innovador de sistemas PLM. El PLM es una estrategia comercial que consiste en aplicar soluciones industriales colaborativas al desarrollo de productos, desde la fase de diseño hasta la fabricación y el reciclaje, en toda la empresa extendida.

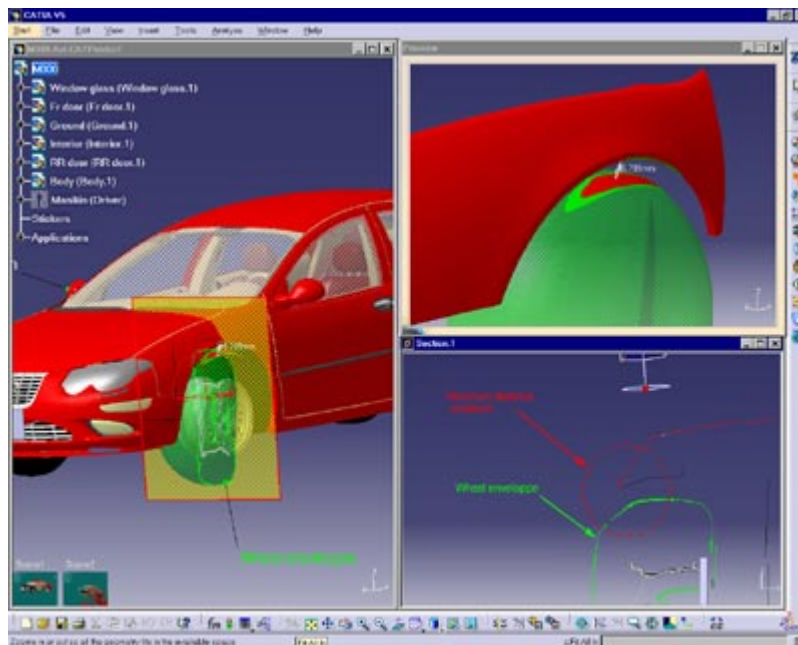


Figura 9 - Vista de la pantalla de Quest - Dassault

SIMULIA es un producto de Dassault que define nuevos estándares, para hacer de la simulación un proceso industrial en sí mismo, dentro de la cadena de valor de la ingeniería. Ofrece una cartera de soluciones escalables para la simulación realista, que incluye las aplicaciones de análisis CATIA, Abaqus para el análisis unificado de elementos finitos, soluciones multifísicas para una mejor comprensión de los retos que se plantean al ingeniero y soluciones para la gestión del ciclo de vida, destinadas a administrar los datos relativos a los procesos, simulación y propiedad intelectual.

Construida a partir de tecnologías probadas, reconocida calidad y destacado servicio de atención al cliente, SIMULIA hace de la simulación realista una práctica industrial crucial, apta para mejorar las prestaciones del producto, reducir los prototipos físicos e impulsar la innovación.

Simnet II

Desarrollado por Hamdy A. Taha, el Simnet II 3 es un sistema que tiene una versión académica que acompaña su libro Investigación de Operaciones Este sistema de simulación cumple casi todos los requisitos, pero presenta las desventajas que se detallan a continuación:

- No posee una interfaz amigable, la misma se ejecuta fuera del entorno visual, inhabilitando el uso del Mouse.
- Si bien los resultados son fáciles de entender, los mismos se presentan en texto plano, sin ningún formato, lo que dificulta la lectura del mismo.
- Es muy difícil conseguir un manual que posibilite la consulta a la hora de transferir el modelo.
- El uso del software se restringe a la compra del libro.
- Tiene limitaciones en cuanto a la cantidad de entidades.
- La funcionalidad adicional debe ser programada en FORTRAN, lenguaje potente y simple de aprender pero actualmente no es muy popular.

A la vez el SIMNET II cuenta con las siguientes ventajas:

- Utiliza teoría de colas en todos sus nodos.
- No necesita de instalación para correr.
- La codificación del modelo se resolvió mediante un pseudo-lenguaje muy práctico y sencillo.
- Muy estable en situaciones de saturación.

Taylor



Origen: Países Bajos

Los sistemas de simulación discretos son incorporados cada vez más al currículum académico. Enterprise Dynamics es una de las plataformas de simulación más utilizadas en estos cursos. No sólo es fácil de aprender (por lo que no pasan demasiado tiempo de aprendizaje del programa en sí), sino que también viene con muchos objetos de simulación (llamados átomos) que ayudan al estudiante en la construcción de un modelo de simulación bien construido para prácticamente cualquier tipo de tema.

Además hay muchos materiales disponibles para ayudar a las universidades y otras instituciones académicas en sus esfuerzos por enseñar simulación:

- La Educational Suite es el mismo software que está normalmente disponible para la industria, pero con materiales de capacitación.
- Una versión libre para los estudiantes de institutos que tengan una licencia de Educación Suite.

³ Publicado con su libro Investigación de Operaciones – Prentice Hall (1988).


- Tutoriales estudiantiles en Inglés, alemán, francés y holandés.
- Un entrenamiento especial de formación de maestros para ayudar al profesor a empezar con un curso de simulación utilizando Enterprise Dynamics.

Tutsim

Origen: Países Bajos 

Tutsim es un proyecto desarrollado en la Universidad de Twente, que fue vendido en los años 70. En la actualidad se está comercializando 20Sim, que es el sucesor de éste. Durante los últimos siete años no se han reportado actualizaciones de este software.

VisSim

Origen: USA 

Visual Solutions es la empresa dedicada a la comercialización de VisSim. VisSim es un lenguaje de modelado visual para la creación de sistemas dinámicos no lineales complejos. Tiene un potente generador de código C, el cual lo hace una gran plataforma para el modelo basado en el desarrollo de sistemas embebidos. Para problemas especializados, VisSim ofrece un conjunto de productos para su resolución, como sistemas en tiempo real, redes neuronales, etc.

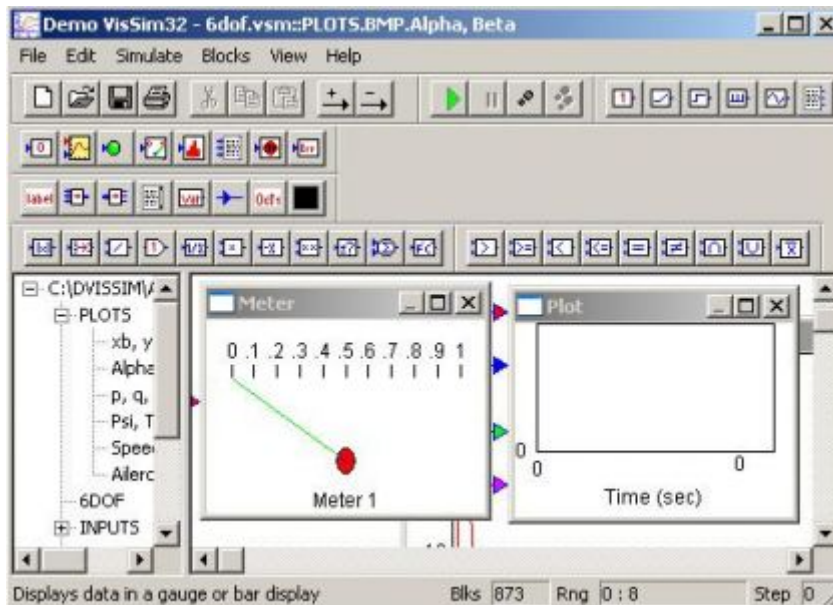


Figura 10 - Vista de diseño en VisSim

Free Academic Program es un programa que provee uso ilimitado de VisSim versión 3.0 por todos los miembros de la institución académica. Está disponible para los departamentos acreditados de la institución que desarrollen modelado y simulación usando VisSim.

El proceso de acreditación es simple. Un profesor del departamento interesado debe enviar una carta a VisSim Academic Program (por correo, fax: 978-692-3102 o e-mail: info@vissol.com) identificando la institución, departamento, especificaciones del programa de la materia, requerimientos del software y la información de distribución. La carta también debe hacer mención a que el software no será usado para actividades lucrativas. Una vez validada la información, Visual Solutions concederá la licencia al departamento y enviará el software. Tenga en cuenta que este programa puede no estar disponible en algunos países.

Productos disponibles:

- Professional VisSim v3.0™, es una herramienta para el modelado en bloques y la posterior simulación
- VisSim/Analyze v3.0™, herramienta de análisis y control del sistema de diseño.

Esta licencia incluye:

- Un año de licencia para usar el software en todas las computadoras que sean necesarias del departamento. La licencia para el paquete académico de VisSim será renovado en forma gratuita mediante una extensión de la licencia al finalizar el primer año.
- Un manual de usuario por cada licencia.
- Todos los integrantes del departamento, estudiantes y profesores, pueden hacer copias del software para usarlos dentro y fuera del campus.
- El soporte técnico está disponible para todos los miembros, sólo vía e-mail (support@vissol.com). Las preguntas serán respondidas en un plazo disponible.

El software entregado bajo este programa no será actualizado ni mantenido por Visual Solutions Inc. Sin embargo la institución podrá actualizar a la versión mas reciente mediante un pago extra. Para solicitar los precios llamar al 1-800-VISSIM-1 o enviar e-mail a info@vissol.com.

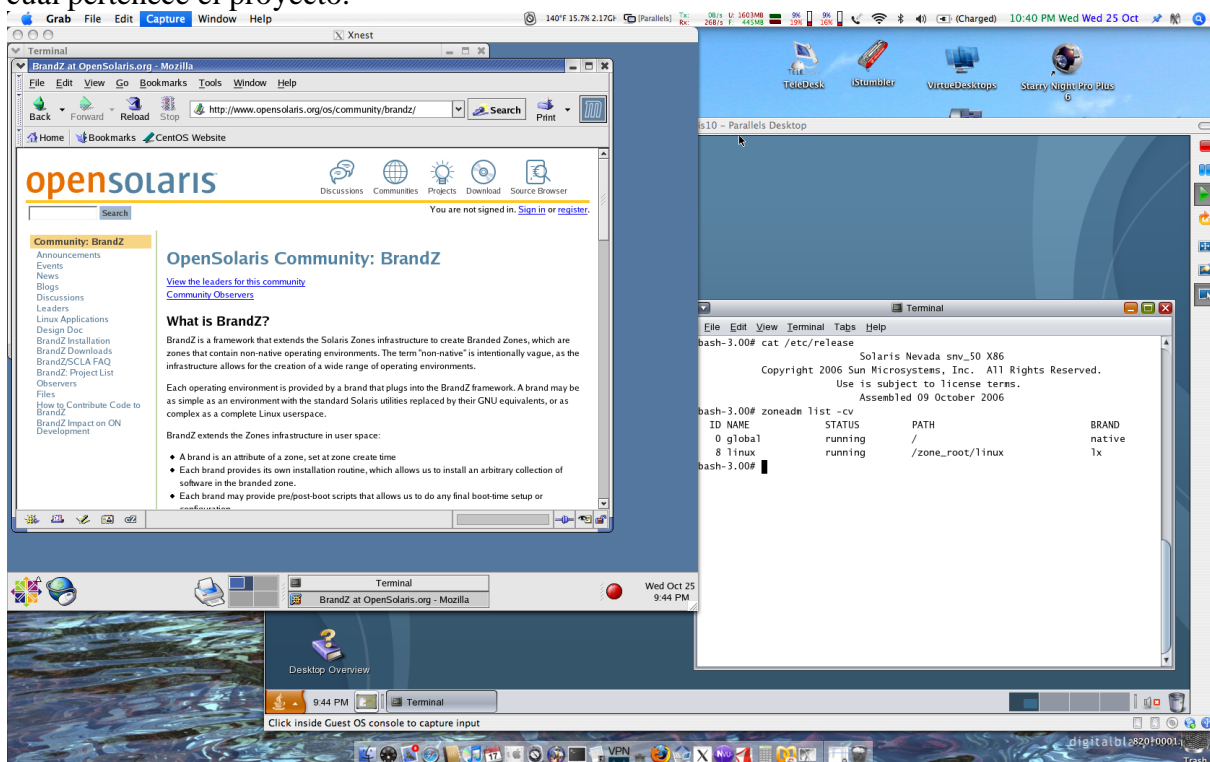
Anexo 3A – Sistema Operativo Linux, Distribución CentOS



CentOS (Community ENTERprise Operating System) es un clon a nivel binario de la distribución Linux Red Hat Enterprise Linux RHEL, compilado por voluntarios a partir del código fuente liberado por Red Hat.

Red Hat Enterprise Linux se compone de software libre y código abierto, pero se publica en formato binario usable (CD-ROM o DVD-ROM) solamente a suscriptores. Como es requerido, Red Hat libera todo el código fuente del producto de forma pública bajo los términos de la Licencia pública general de GNU y otras licencias. Los desarrolladores de CentOS usan ese código fuente para crear un producto final que es muy similar al Red Hat Enterprise Linux y está libremente disponible para ser bajado y usado por el público, pero no es mantenido ni asistido por Red Hat. Existen otras distribuciones también derivadas de las fuentes de Red Hat.

CentOS usa yum para bajar e instalar las actualizaciones, herramienta también utilizada por Fedora. Se ajusta plenamente a la política de redistribución y apunta a ser 100% compatible. Su uso es gratuito, sin embargo, se puede colaborar con la Fundación Caos, a la cual pertenece el proyecto.



Captura de pantalla del escritorio de CentOS

Anexo 3B – Servidor Web, Apache Tomcat

Apache Tomcat es una aplicación para el uso de tecnologías Java Servlet y JavaServer Pages (JSP). Las especificaciones Java Servlet y JavaServer Pages están elaboradas en el marco del Java Community Process.

Apache Tomcat es desarrollado en forma abierta y participativa del medio ambiente y puesto en libertad en virtud de la Apache Software License. Apache Tomcat está destinada a ser una colaboración de los mejores desarrolladores de todo el mundo.

Tomcat es el Servidor Web más utilizado a la hora de trabajar con Java en entornos Web; Tomcat es una implementación completamente funcional de los estándares de JSP y Servlets. Tomcat también puede especificarse como el manejador de las peticiones de JSP y servlets recibidas por servidores Web populares, como el servidor Apache HTTP de la Fundación de software de Apache o el servidor Microsoft Internet Information Server (IIS). Tomcat está integrado en la implementación de referencia Java 2 Enterprise Edition (J2EE) de Sun Microsystems.

Anexo 3C – Versionado, CVS

El Concurrent Versions System (CVS), también conocido como Concurrent Versioning System, es una aplicación informática que implementa un sistema de control de versiones: mantiene el registro de todo el trabajo y los cambios en los ficheros (código fuente principalmente) que forman un proyecto y permite que distintos desarrolladores (independientemente de su posición geográfica) colaboren. CVS se ha hecho popular en el mundo del software libre. Sus desarrolladores difunden el sistema bajo la licencia GPL.

CVS utiliza una arquitectura cliente-servidor: un servidor guarda las versiones actuales del proyecto y su historial. Los clientes se conectan al servidor para sacar una copia completa del proyecto. Esto se hace para que eventualmente puedan trabajar con esa copia y más tarde ingresar sus cambios con comandos GNU.

Típicamente, cliente y servidor se conectan utilizando Internet, pero con el sistema CVS el cliente y servidor pueden estar en la misma máquina. El sistema CVS tiene la tarea de mantener el registro de la historia de las versiones del programa de un proyecto solamente con desarrolladores locales. Originalmente, el servidor utilizaba un sistema operativo similar a Unix, aunque en la actualidad existen versiones de CVS en otros sistemas operativos, incluido Windows. Los clientes CVS pueden funcionar en cualquiera de los sistemas operativos más difundidos.

Varios clientes pueden sacar copias del proyecto al mismo tiempo. Posteriormente, cuando actualizan sus modificaciones, el servidor trata de acoplar las diferentes versiones. Si esto falla, por ejemplo debido a que dos clientes tratan de cambiar la misma línea en un archivo en particular, entonces el servidor deniega la segunda actualización e informa al cliente sobre el conflicto, que el usuario deberá resolver manualmente. Si la operación de ingreso tiene éxito, entonces los números de versión de todos los archivos implicados se incrementan automáticamente, y el servidor CVS almacena información sobre la actualización, que incluye una descripción suministrada por el usuario, la fecha y el nombre del autor y sus archivos Log.

Los clientes pueden también comparar diferentes versiones de archivos, solicitar una historia completa de los cambios, o sacar una foto histórica del proyecto tal como se encontraba en una fecha determinada o en un número de revisión determinado. Muchos proyectos de código abierto permiten el acceso de lectura anónimo, significando que los clientes pueden sacar y comparar versiones sin necesidad de teclear una contraseña; solamente el ingreso de cambios requiere una contraseña en estos casos.

Los clientes también pueden utilizar la orden de actualización con el fin de tener sus copias al día con la última versión que se encuentra en el servidor. Esto elimina la necesidad de repetir las descargas del proyecto completo.

CVS también puede mantener distintas ramas de un proyecto. Por ejemplo, una versión difundida de un proyecto de programa puede formar una rama y ser utilizada para corregir errores. Todo esto se puede llevar a cabo mientras la versión que se encuentra actualmente en desarrollo y posee cambios mayores con nuevas características se encuentre en otra línea formando otra rama separada.

Los archivos en el repositorio sobre la plataforma CVS no pueden ser renombrados, estos deben ser agregados con otro nombre y luego eliminados. El protocolo CVS no provee una manera de que los directorios puedan ser eliminados o renombrados, cada archivo en cada subdirectorío debe ser eliminado y re-agregado con el nuevo nombre. Soporte limitado para archivos Unicode con nombres de archivo no ASCII.

Un sistema de control de versiones proporciona:

- Mecanismo de almacenaje de los elementos que deba gestionar (ej. archivos de texto, imágenes, documentación.)
- Posibilidad de realizar cambios sobre los elementos almacenados (ej. modificaciones parciales, añadir, borrar, renombrar o mover elementos)
- Registro histórico de las acciones realizadas con cada elemento o conjunto de elementos (normalmente pudiendo volver o extraer un estado anterior del producto)

Aunque no es estrictamente necesario, suele ser muy útil la generación de informes con los cambios introducidos entre dos versiones, informes de estado, marcado con nombre identificativo de la versión de un conjunto de ficheros, etcétera.

La principal clasificación que se puede establecer está basada en el almacenamiento del código:

- Centralizados: existe un repositorio centralizado de todo el código, del cual es responsable un único usuario (o conjunto de ellos). Se facilitan las tareas administrativas a cambio de reducir la potencia y flexibilidad, pues todas las decisiones fuertes (como crear una nueva rama) necesitan la aprobación del responsable. Algunos ejemplos son CVS y Subversión
- Distribuidos: se aumenta la capacidad de decisión distribuida. Esto da más flexibilidad pero puede dificultar bastante la sincronización. Ejemplos: Git y GNU Arch

Todos los sistemas de control de versiones se basan en disponer de un repositorio, que es el conjunto de información gestionada por el sistema. Este repositorio contiene el historial de versiones de todos los elementos gestionados.

Cada uno de los usuarios puede crearse una copia local duplicando el contenido del repositorio para permitir su uso. Es posible duplicar la última versión o cualquier versión almacenada en el historial. Este proceso se suele conocer como check out o desproteger. Para modificar la copia local existen dos semánticas básicas:

- Exclusivos: para poder realizar un cambio es necesario marcar en el repositorio el elemento que se desea modificar y el sistema se encargará de impedir que otro usuario pueda modificar dicho elemento.
- Colaborativos: en el que cada usuario se descarga la copia la modifica y el sistema automáticamente mezcla las diversas modificaciones. El principal problema es la posible aparición de conflictos que deban ser solucionados manualmente o las

posibles inconsistencias que surjan al modificar el mismo fichero por varias personas no coordinadas. Además, esta semántica no es apropiada para ficheros binarios.

Tras realizar la modificación es necesario actualizar el repositorio con los cambios realizados. Habitualmente este proceso se denomina commit, check in o proteger.

Bonsai CVS es un software diseñado para ayudar a gestionar de forma productiva grandes proyectos de programación que utilicen el sistema de control de versiones CVS. Fue desarrollado inicialmente por la Fundación Mozilla para que múltiples programadores pudieran editar su extremadamente grande base de software.

Entre otras características, Bonsái permite realizar consultas complejas al servidor CVS, así como realizar actualizaciones casi en tiempo real al mismo.

Bonsái fue fundado por Terry Weissman. En principio se escribió en Tcl, y más tarde fue portado a Perl. Se ejecuta sobre una base de datos MySQL, y puede ser utilizado mediante cualquier servidor que soporte scripts en Perl, como por ejemplo Apache.

Adaptado de <http://es.wikipedia.org/wiki/CVS> y <http://www.programacion.com>

Anexo 3D – MySQL

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB (desde enero de 2008 una subsidiaria de Sun Microsystems) desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado se ofrece bajo la GNU-GPL para cualquier uso compatible con esta licencia, pero las empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario que proyectos como Apache, donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual, MySQL es propiedad y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson, y Michael Widenius.

SQL (Structure Query Lenguaje - Lenguaje de Consulta Estructurado) fue comercializado por primera vez en 1981 por IBM, el cual fue presentado a ANSI y desde ese entonces ha sido considerado como un estándar para las bases de datos relacionales. Desde 1986, el estándar SQL ha aparecido en diferentes versiones como por ejemplo: SQL:92, SQL:99, SQL:2003. MySQL es una idea originaria de la empresa opensource MySQL AB establecida inicialmente en Suecia en 1995 y cuyos fundadores son David Axmark, Allan Larsson, y Michael "Monty" Widenius. El objetivo que persigue esta empresa consiste en que MySQL cumpla el estándar SQL, pero sin sacrificar velocidad, fiabilidad o usabilidad.

Michael Widenius en la década de los 90 trató de usar mSQL para conectar las tablas usando rutinas de bajo nivel ISAM, sin embargo, mSQL no era rápido y flexible para sus necesidades. Esto lo conllevó a crear una API SQL denominada MySQL para bases de datos muy similar a la de mSQL pero más portable.

La procedencia del nombre de MySQL no es clara. Desde hace más de 10 años, las herramientas han mantenido el prefijo My. También, se cree que tiene relación con el nombre de la hija del cofundador Monty Widenius quien se llama My.

Por otro lado, el nombre del delfín de MySQL es Sakila y fue seleccionado por los fundadores de **MySQL** AB en el concurso "Name the Dolphin". Este nombre fue enviado por Ambrose Twebaze, un desarrollador de Open source Africano, derivado del idioma SiSwate, el idioma local de Swazilandia y corresponde al nombre de una ciudad en Arusha, Tanzania, cerca de Uganda la ciudad origen de Ambrose.

Existen varias APIs que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Pascal, Delphi (via dbExpress), Eiffel, Smalltalk, Java (con una implementación nativa del driver de Java), Lisp, Perl, PHP, Python, Ruby, Gambas, REALbasic (Mac), FreeBASIC, y Tcl; cada uno de estos utiliza una API específica. También existe un interfaz ODBC, llamado MyODBC que

permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL. También se puede acceder desde el sistema SAP, lenguaje ABAP.

MySQL es muy utilizado en aplicaciones web como , Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

Características:

- Usa GNU Automake, Autoconf, y Libtool para portabilidad
- Uso de multihilos mediante hilos del kernel.
- Usa tablas en disco b-tree para búsquedas rápidas con compresión de índice
- Tablas hash en memoria temporales
- El código MySQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL
- Completo soporte para operadores y funciones en cláusulas select y where.
- Completo soporte para cláusulas group by y order by, soporte de funciones de agrupación
- Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
- Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).
- Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix.
- En MySQL 5.0, los clientes y servidores Windows se pueden conectar usando memoria compartida.
- MySQL contiene su propio paquete de pruebas de rendimiento proporcionado con el código fuente de la distribución de MySQL.

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad.

Poco a poco los elementos de los que carecía MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

La licencia GNU GPL de MySQL obliga a que la distribución de cualquier producto derivado (aplicación) se haga bajo esa misma licencia. Si un desarrollador desea incorporar MySQL en su producto pero desea distribuirlo bajo otra licencia que no sea la GNU GPL, puede adquirir una licencia comercial de MySQL que le permite hacer justamente eso.

Anexo 3E – Flash & ActionScript

Adobe Flash es una aplicación en forma de estudio de animación que trabaja sobre Fotogramas destinado a la producción y entrega de contenido interactivo para diferentes audiencias alrededor del mundo sin importar la plataforma. Es actualmente escrito y distribuido por Adobe Systems, y utiliza gráficos vectoriales e imágenes ráster, sonido, código de programa, flujo de vídeo y audio bidireccional (el flujo de subida sólo está disponible si se usa conjuntamente con Macromedia Flash Communication Server). En sentido estricto, Flash es el entorno y Flash Player es el programa de máquina virtual utilizado para ejecutar los archivos generados con Flash.

Los archivos de Flash, que tienen generalmente la extensión de archivo SWF, pueden aparecer en una página web para ser vista en un navegador, o pueden ser reproducidos independientemente por un reproductor Flash. Los archivos de Flash aparecen muy a menudo como animaciones en páginas Web y sitios Web multimedia, y más recientemente Aplicaciones de Internet Ricas. Son también ampliamente utilizados en anuncios de la web.

En versiones anteriores, Macromedia amplió a Flash más allá de las animaciones simples, convirtiéndolo en una herramienta de desarrollo completa, para crear principalmente elementos multimedia e interactivos para Internet.

Fue hasta 2005 perteneciente a la empresa Macromedia conocido hasta entonces como Macromedia Flash® y adquirido por Adobe Systems (desde entonces conocido como Adobe Flash) ampliando con ello su portafolio de productos dentro del mercado.

ActionScript es el lenguaje de programación para crear scripts en Flash. Utilizando ActionScript en ciertos fotogramas clave podemos crear elementos interactivos como:

- La botonera clásica de navegación, con botones que reaccionan a clics del ratón y envían la cabecera de la película a un fotograma concreto.
- Contenido que se anima basándose en movimientos del ratón.
- Objetos que pueden ser movidos por el ratón o el teclado.
- Campos de texto que permiten a los usuarios entrar datos a la película como en un formulario.
- Controlar el contenido audiovisual.
- Cambiar el color y la localización de un objeto.
- Reducir el volumen de un sonido.
- Especificar la tipografía de un bloque de texto.
- Campos de texto que permiten a los usuarios entrar datos a la película como en un formulario.
- Modificar las propiedades repetidamente produciendo comportamientos únicos como son los movimientos basados en la física y la detección de colisiones.

Con ActionScript podemos generar contenido directamente desde la biblioteca de la película o duplicar contenido existente en el Escenario. El contenido generado en forma de programa puede servir como

- Un elemento estático.
- Una plantilla visual aleatoria.
- Un elemento interactivo, por ejemplo una nave en un juego espacial.
- Una opción de un menú que se abre cuando la presionan (pull-down menú).

ActionScript provee de una amplia variedad de herramientas para enviar y recibir información del servidor. Ejemplos de comunicación con el servidor son:

- Enlace a una página web.
- Libro de visitas.
- Aplicación de chat.
- Juego multijugadores a través de la red.
- Transacción de e-comercio.
- Sitio personalizado con nombre de usuario y contraseña.

FUENTE: <http://www.programacion.com/tutorial/actionscript/1/>

Anexo 3F - Web services – Protocolo SOAP

Los servicios web son la revolución informática de la nueva generación de aplicaciones que trabajan colaborativamente en las cuales el software está distribuido en diferentes servidores.

La informática se inició con programas monousuarios implantados en grandes servidores. Posteriormente estas primeras aplicaciones alcanzaron la capacidad de atender a diferentes usuarios. Pasaron los años y llegó la arquitectura cliente-servidor, que gracias a este modelo de desarrollo, la aplicación se dividía en una parte que interactuaba con el usuario y otra parte destinada al procesamiento de información. En este acercamiento se consiguió que cada una de las partes que constituían la aplicación pudiera residir en computadoras distintas.

Con el paso del tiempo, la computación aumentó y llegó la era de las aplicaciones distribuidas en las cuales los procesos se realizaban en diferentes unidades. De este paso surgió la tecnología Internet para solventar las problemáticas asociadas a fallo de aplicación centralizado.

Como punto final a esta cronología, los web services son un paso adelante en la computación ya que de esta forma un ordenador ya no se considerara como un núcleo de cómputo sino como un repositorio de servicios de n aplicaciones distribuidas por internet. Para comprender este concepto pondremos el ejemplo siguiente:

Imaginemos el supuesto de un servidor X que da 2 servicios, el servicio A y el servicio B. También nos dan el caso de que tenemos otro servidor Y que ofrece el servicio C y el servicio D. Por último también analizamos el servidor Z que da 2 servicios el servicio E y el servicio F.

Imaginar que el servidor W ofrece una aplicación on-line que se compone de su servicio principal G y de la interoperación con A, B, C, D, por tanto se da que los servicios A, B, C, D dan servicio a G independiente de su localización en G pero también están disponibles para el resto de internet si así se ha diseñado.

Por tanto viendo los datos anteriores vemos que las posibilidades combinatorias son grandes y sobre todo por primera vez y de manera más transparente que con el uso de componentes, el uso del software se hace más reutilizable ya que diferentes servicios pueden ser utilizados como componentes en una aplicación. Microsoft para conseguir este propósito con su tecnología .NET emplea como protocolo de comunicación, una aplicación XML, llamada SOAP.

SOAP son las siglas de Simple Object Access Protocol. Este protocolo deriva de un protocolo creado por David Winer, XML-RPC en 1998. En su sitio web, Userland, <http://www.userland.com> se puede encontrar multitud de documentación acerca de este primer protocolo de comunicación bajo http mediante XML. Con este protocolo se podían realizar RPC o remote procedure calls, es decir, podíamos bien en cliente o servidor realizar peticiones mediante http a un servidor web. Los mensajes debían tener un formato determinado empleando XML para encapsular los parámetros de la petición. Con el paso del tiempo el

proyecto iniciado por David Winer interesó a importantes multinacionales entre las que se encuentran IBM y Microsoft y de este interés por XML-RPC se desarrolló SOAP.

SOAP es un protocolo más completo que XML-RPC pero cabe decir que más complejo. La siguiente tabla comparativa muestra las diferencias entre ambos protocolos:

Características	XML-RPC	SOAP
Escalares básicos.	yes	yes
Estructuras.	yes	yes
Arrays.	yes	yes
Estructuras nombradas y Arrays.	no	yes
Manejo de fallos.	yes	yes
Curva de aprendizaje.	yes	no
Conjunto de caracteres.	no	yes (US-ASCII, UTF-8, UTF-16)
Tipos de datos definidos por usuario.	no	yes
Requiere entendimiento del cliente.	no	yes
Instrucciones de procesamiento Específicas.	no	yes

Uso de Web Services con Java y Flash

Los Web Services o servicios webs son una forma de comunicar aplicaciones diferentes usando XML. Esto resulta muy útil ya que nos brinda una opción de comunicación estándar que podemos usar entre aplicaciones realizadas con distintos lenguajes como ActionScript, C# o Java.

Como ejemplo concreto vamos a ver cómo podemos comunicar aplicaciones ActionScript (realizadas con Macromedia Flash) y aplicaciones Java (realizadas con JBuilder), en cuyo caso se usa SOAP, el uso de XML va a ser completamente transparente al programador pero no nos debe preocupar el protocolo.

Lo primero que hay que hacer es crear un proyecto con un método en Java, este método será llamado por Flash mediante webservices, puede hacer cualquier cosa que se pueda hacer en Java pero si envía o recibe parámetros estos deben ser serializables, es decir, tienen que ser objetos que contengan cadenas o vectores con cadenas (o vectores de vectores con cadenas) pero no estructuras más complejas como tablas hash, árboles, etc.

Una vez creado el método en JBuilder hacemos click derecho sobre él y seleccionamos "Export as a Web Service", si no tenemos ningún otro Web Service en el proyecto o queremos que se ejecute en uno aparte seleccionamos "Create web services module", en web module seleccionamos "New", ahora seleccionamos "Single server for all services in project" y elegimos Tomcat 4.1 o 5 según la versión que vayamos a utilizar. Finish. Finish. Ahora nos aparecerá el Web Services Designer donde podemos ver gráficamente nuestro Web Service. Para lanzarlo: botón derecho en la clase y pulsamos sobre "Rebuild", una vez ha acabado pulsamos sobre "Run Project" (un botón similar al de play pero en verde). En este momento ya tenemos Tomcat y nuestro Web Service ejecutándose, solo queda diseñar la parte de flash.

Abrimos Macromedia Flash y seleccionamos "Archivo", "Nuevo", "Documento de Flash" o abrimos un archivo flash que queramos conectar con nuestra función java. Ahora, en la derecha, en el panel de "Componentes" arrastramos el componente "WebServiceConnector" a nuestra área de trabajo. Para poner a punto el Web Service Connector le damos un nombre (pestaña "Propiedades", label "Componente") por ejemplo WSC, introducimos la WSDLURL (suele ser una url del tipo *http://10.142.85.97:8080/WebModule1/services/pruebaWS?wsdl*, la podemos consultar mediante la ventana de navegación que se nos ha abierto en el JBuilder) y el nombre del método al que queremos llamar, por ejemplo enviar cadena. Todos estos campos se pueden cambiar en tiempo de ejecución mediante `_root.WSC`. Para activar el Web Service podemos hacerlo cuando se carga el archivo swf, cuando se pulsa un botón, etc. Lo importante es poner el siguiente código para enviar y recibir los parámetros:

```
//Para el listener
var wscListener:Object = new Object();
wscListener.result = function(evt:Object) {
//en evt.target.results tenemos las variables devueltas por el método java
trace("Cadena obtenida: " + evt.target.results);
};
//Anyadimos el listener
_root.WSC.addEventListener("result", wscListener);
//en _root.WSC.params introducimos los datos que le llegan al método java
_root.WSC.params=["Te envío esta cadena"];
//que no se nos olvide activar el Web Service! :D
_root.WSC.trigger();
```

Conclusión.

Y eso es todo, a pesar de la simplicidad del ejemplo los Web Services son una herramienta muy potente que con el tiempo substituirá a otras interfaces entre aplicaciones. Además, el hecho de que esté basado en XML mediante SOAP nos asegura la compatibilidad con cualquier lenguaje que soporte dicho protocolo.

FUENTE: <http://www.cristalab.com/>

<http://www.desarrolloweb.com/articulos/957.php>



Anexo 4A – Campana

El partido de Campana posee una ubicación geográfica privilegiada ya que está a sólo 75 Km. de la ciudad de Buenos Aires, la autopista panamericana la une con la ciudad capital, y también con el centro y norte de la República Argentina.

Posee acceso fluvial a través del río Paraná de las Palmas, vinculándose por esta vía con Buenos Aires, Rosario y todo el litoral argentino. También cuenta con acceso ferroviario, de gran importancia en el desarrollo económico y social para el Partido. Está compuesto por las localidades de: Campana, Colinas de Otamendi, Lomas de Río Luján y Alto Los Cardales.



La ciudad de Campana, cabecera del Partido, se asienta sobre la margen derecha del Río Paraná de las Palmas. La diversidad de paisajes hace de Campana un lugar de gran atractivo turístico ya que posee actividades relacionadas con el río, el agro, la ecología, el deporte y la cultura.

Historia

1680, una estancia adquirida por el capitán Luis de Águila, ubicada en la zona de la "Cañada de la Cruz", corresponde al dato más antiguo que se conoce del lugar. La propiedad tuvo varios dueños

19 de mayo de 1759, la propiedad fue comprada por el comerciante andaluz Francisco Álvarez Campana. Desde entonces el lugar se comenzó a llamar "Rincón de Campana"

En 1860 las tierras fueron adquiridas, después de numerosas transacciones, por los hermanos Eduardo Costa y Luis Costa, que edificaron la "Estancia Vieja"

18 de abril de 1875, Luis Costa funda el pueblo. Se procede al loteo y creación del pueblo de Campana. Como dato interesante cabe señalar el hecho de que ya por entonces Campana se perfilaba como un importante polo de desarrollo; el puerto de ultramar y la creación de la línea férrea, ratificarían ese destino de progreso.

1876, se realiza el primer viaje de ferrocarril que uniría Retiro con Campana. Numerosas instituciones, como el correo, la policía, el consejo escolar y las escuelas, hacen pronta aparición en el pueblo que comienza a poblarse de numerosos inmigrantes.

En 1885 el pueblo se separa del partido de Exaltación de La Cruz, creándose el de Campana, el 6 de julio del mismo año. Don Alfredo Dabble levanta el primer frigorífico

Argentino. Se instalan así numerosas industrias como El Molino Harinero de Morixe y la Destilería de Alcohol.




Hasta 1924, el Frigorífico, dio empleo a miles de trabajadores campanenses. Ese año un incendio destruye sus instalaciones. Desde entonces, Campana entra en una pronunciada decadencia que se prolonga hasta la instalación en nuestra ciudad de la empresa Dalmine SAFTA, en 1954.



En 1925 se construye la nueva estación de ferrocarril.

A partir de 1954 se instalan varias empresas, entre ellas Dalmine SAFTA (hoy Tenaris-Siderca), la destilería Esso Sapa, Cometarsa, Carborundum, Pasa y Rhasa.

Información general

País	 Argentina
Provincia	 Buenos Aires
Partido	Campana
Ubicación	 34°10'37"S 58°55'15"O
Altitud	2 m.s.n.m.
Superficie	982 km ² (644 Km ² en sector insular)
Fundación	6 de julio de 1885 (por Luís y Eduardo Costa)
Población	83.698 hab.-INDEC, 2001 (900 hab. en Islas –Prefectura, 2000)
Densidad	85.2 hab/Km ²
Tasa de Crecimiento	17,1 % (1991, 2001) %



Según estimaciones del INDEC para junio de 2008 la población del partido alcanzará los 94.179 habitantes.

Evolución poblacional del partido de Campana según los distintos censos nacionales y variación intercensal en porcentaje								
	<u>1895</u>	<u>1914</u>	<u>1947</u>	<u>1960</u>	<u>1970</u>	<u>1980</u>	<u>1991</u>	<u>2001</u>
Población	7.733	15.470	17.250	30.747	44.297	57.839	71.464	83.698
Variación	-	+100,05%	+11,50%	+78,24%	+44,06%	+30,57%	+23,55%	+17,12%

Geografía

Se encuentra adyacente a la autopista Buenos Aires - Rosario - Córdoba RN 9, sobre la red troncal del Ferrocarril General Bartolomé Mitre.

En conjunto con Zárate conforman una importante área industrial de la conurbación megalópolis argentina que se extiende sobre las riberas del Plata - Paraná, entre el Gran La Plata y el Gran Rosario. La población es de 77.838 habitantes (INDEC, 2001) (50,88% mujeres, 49,16% hombres). Junto con la ciudad de Zárate, forman la aglomeración llamada Zárate - Campana siendo la población total de esta de 164.524 habitantes (INDEC, 2001).



El trazado de la ciudad se caracteriza por sus diagonales, teniendo una gran similitud con la ciudad de La Plata (ciudad capital de la provincia de Buenos Aires).

Ciudad de características eminentemente industriales, es sede de variadas e importantes industrias, en varios casos líderes mundiales en sus respectivos rubros.

En los últimos años, la ciudad de Campana ha alcanzado relevancia y reconocimiento internacional debido a la pujanza de algunas de sus industrias.

La industria

La puesta en marcha del puerto y la llegada del ferrocarril fueron el punto de partida del desarrollo de Campana, que se fue acrecentando con el correr de los años, en base a la excelente ubicación de la ciudad y su puerto magníficamente trazado.

Con un calado estimado en 32 pies durante todo el año, este puerto, en virtud del canal Emilio Mitro, tiene una fluida comunicación con todo el mundo y se transforma en el lugar ideal para establecer un parque industrial de excepcional importancia, tal como lo confirman las empresas que se han establecido en la zona y continúan sumándose permanentemente.



No es ningún secreto que el eje industrial desde La Plata hasta Rosario, genera en sí mismo el 70 % de las riquezas del país, y que el parque industrial que forma un cinturón en torno al antiguo pueblo fundado por los hermanos Costa, se ha convertido en uno de los más importantes puntos de desarrollo de ese eje.

Así, luego del desaparecido frigorífico (debido a un incendio en 1924) se afincaron la destilería petrolera ESSO y el Complejo Industrial Siderúrgico Siderca, creado por el visionario Agustín Rocca, que se encuentra en permanente desarrollo, aumentando las exportaciones argentinas de productos no tradicionales con el valor agregado de mano de obra nacional.

Otras empresas de jerarquía han instalado sus plantas de producción en Campana aprovechando su inmejorable ubicación, tales como Pasa, Rhasa, Sol Petróleo, Cabot, Carborundum, la mayoría de ellas líderes en sus rubros.

El puerto de Campana es hoy el tercero en recaudación fiscal en todo el país, lo que da una idea fehaciente de su importancia.

Zárate, Campana y Lima, conforman uno de los polos industriales más importantes de la provincia, sólo superado por el gran La Plata y el polo industrial de Bahía Blanca.

Zárate está rodeada de tres empresas cerveceras, Quilmes, Isenbeck y Budweiser, una gran empresa automotriz en proceso de expansión: Toyota Argentina, empresas vinculadas a la biotecnología, como Monsanto, Petrobrás, Merisant y Pecom; cinco papeleras: Celulosa, Witcel, Papelera del Plata, Valot y Celulosa Campana, una Central Nuclear operando y dos centrales de energía en construcción, más otras empresas de envergadura como la fabricante de tableros laminados Faplac, las químicas Lanxess (ex Bayer Argentina), Ciba Geigy, Rohm & Haas, TFL, Voridian (ex Eastman), Soluciones Químicas, la cementera Minetti, y la actividad portuaria que brindan Auto Terminal Zárate, Zárate Port, Murchison y Delta Dock. Zárate además es reconocida por su pujante actividad comercial y mercantil, al punto de que

las principales franquicias de productos de consumo masivo se han instalado en la zona sumándose al competitivo comercio local, contando con 500 personas adheridas al Centro de Comercio e Industrias de Zárate, abriendo el abanico de ofertas para los consumidores.

Campana y Zárate están separadas sólo por 8 kilómetros, y ambas ciudades se comunican entre si a través de la flamante Autovía 6 “Ruta del Mercosur” (ex Ruta Provincial Nro 6), la ruta Panamericana, el ramal principal del FFCC Mitre y el rio Paraná de las Palmas.

La ubicación estratégica de este polo industrial lo sitúa entre las vías de comercialización más importantes del MERCOSUR, con vías terrestres, ferroviarias y fluviales que la acercan a Uruguay, Brasil, Paraguay y al Mercado Común Europeo a través del Océano Atlántico.

Esta situación de privilegio convierte a Zárate-Campana y a su zona de influencia en un centro de producción de bienes y servicios único en su género, atrayendo las miradas de los inversores del país y del mercado internacional.

Hasta la década de los noventa otro importante predio industrial de Campana fueron los talleres ferroviarios, linderos a la estación ferroviaria Campana. Los mismos fueron y permanecen cerrados, aunque una parte de los mismos fue rescatado por una cooperativa metalurgica (CO.DE.ME).

Universidad Tecnológica Nacional Regional Delta

La idea de lograr la formación integral de sus estudiantes, despertó en Campana hacia los años 70, un sentimiento movilizador que con igual intención repercutió en Zárate.

La comunidad toda desde sus entidades representativas: Cámara Unión del Comercio e Industria, Club de Leones, Rotary Club, empezó a trabajar en orden a la creación de un centro universitario que sirviera a la realidad industrial de la zona.

Se compartieron propuestas, se realizó un excelente trabajo de fundamentación para ser presentado ante las autoridades (coordinado por el Arq. Ricardo Abel Premat) con las que se efectuaron numerosas entrevistas, manteniéndose continuamente informada a la opinión pública.



La intensa tarea demandó muchos meses, culminando en 1973, cuando gracias al invaluable apoyo del entonces comisionado municipal Wesser Masón (que ofreció el subsidio) y de la directora de la escuela N° 9 Sra. L. de Boéchat que aún a sabiendas de los problemas que se presentarían, cedió las instalaciones de la escuela para que allí comenzaran los esperados cursos.

Posteriormente, el continuo crecimiento obligó a trasladarse a un edificio alquilado en la esquina de Belgrano y Mitre, que durante un tiempo se compartió con la Asociación Dante Alighieri.

Los laboratorios funcionaron en la Escuela Normal Dr. Eduardo Costa, que sus autoridades cedieron con agrado. En la vecina ciudad de Zárate, se repetía la situación.

Con el tiempo, el Rectorado Nacional de la UTN, presenta a ambas comunidades la necesidad de unificar en un solo edificio los cursos que se realizaban en las dos localidades, con el consiguiente traslado de los alumnos de un lugar a otro.

Las fuerzas vivas de Campana unidas en una comisión, que también integraban alumnos, impulsadas por la Cámara Empresaria, consiguen que la intendencia a cargo del Sr. Calixto Dellepiane, done una manzana de terreno de su propiedad, solicitando a los directivos de las empresas radicadas en Campana, el conseguir los fondos necesarios para su construcción.

Finalmente el milagro se produce, la empresa Esso SAPA, que conmemoraba por aquel entonces el 75° Aniversario de su radicación en la ciudad, gracias al impulso del presidente de Exxon Ing. Juan Yañez, que había trabajado en Campana, y la colaboración del Jefe de RR.PP. de la Refinería Campana Don Eduardo Saccone, se hace cargo de la construcción de la moderna sede, donde hoy cursan sus estudios más de un millar de alumnos de la zona.

La sede fue inaugurada en 1981, siendo bendecida por el primer obispo de la diócesis Campana-Zárate Monseñor Espósito Castro.

El 1° de abril de 1980 por resolución 340 del rectorado de la Universidad, pasó a ser Escuela de Ingeniería Delta.

Es de destacar que esa casa de estudios durante el período de normalización de las universidades que se inicia con el restablecimiento de la democracia, se le otorgó el rango de Facultad Regional mediante la resolución 2046 del Ministerio de Educación de la Nación, el 17 de septiembre de 1984.

En 1987 comenzó a dictarse la carrera de Seguridad e Higiene Industrial y en 1993 se creó la carrera de Ingeniería de Sistemas de Información, que contribuyó a dar nuevo impulso a las distintas actividades y determinó el incremento sostenido de alumnos inscriptos, el cual actualmente asciende alrededor de 1100.

El patrimonio en instrumentos, máquinas y equipos se vio incrementado significativamente todos estos años, llevándose a cabo varias ampliaciones en el edificio, en los sectores de aulas, laboratorios, biblioteca, centro de cómputos, oficinas, servicios y Centro de Energía Ambiente.

En 1995 se produjo la apertura de inscripción a término en la carrera de Licenciatura en Administración Rural; y en 1996 comienza el nivel cuaternario, con el inicio de la carrera de Ingeniería Laboral, incorporándose el mismo año la carrera de Ingeniería Gerencial.

En 1997 se inició la carrera de posgrado de Especialista en Docencia Universitaria, con más de treinta docentes de la casa cursando la misma.

Finalmente, cabe destacar también el crecimiento permanente de las actividades de los grupos de investigación, así como los de los correspondientes a servicios de extensión hacia la comunidad, con múltiples implicancias en el desarrollo de esta Facultad Regional.

El origen, el contexto y su devenir, y la pertinencia institucional han ido modelando este presente de la Facultad Regional Delta, caracterizado por su particular desarrollo académico, sus líneas de investigación y su modo de relación con el medio.

Una característica distintiva de esta Institución es una adecuada proporción docentes-alumnos, la cual determina en gran medida estilos de interacción que favorecen las relaciones persona a persona.

Por otro lado, y dado el contexto regional de la facultad con un elevado número de industrias radicadas en la zona, se presenta como característica relevante el perfil docente general, ligado a la especialidad, con amplia experiencia profesional en las actividades relacionadas con las carreras, permitiendo de ese modo realizar el nexo entre el medio productivo y las disciplinas especializadas.

Fuente:

- [http://es.wikipedia.org/wiki/Campana_\(Buenos_Aires\)](http://es.wikipedia.org/wiki/Campana_(Buenos_Aires))
- encampana.com
- zarateinforma.com.ar
- campana.gov.ar
- Instituto Nacional de Estadísticas y Censos, INDEC

Anexo 4B – Santa Fe

La ciudad de Santa Fe de la Vera Cruz es la capital de la provincia de Santa Fe, y se encuentra en el sector centro-este de Argentina. Posee una población de 369.589 hab. (censo 2001). Es también capital del departamento La Capital de dicha provincia.

La ciudad de Santa Fe es la cuna de la Constitución de la Nación Argentina de 1853 y sede de la Convención reformadora de 1994.



Esta ciudad se caracteriza por tener una función principalmente administrativa, pero también constituye un polo importante de educación terciaria y universitaria, atrayendo miles de estudiantes de todas las regiones del país. Además comienza a ser un atractivo turístico importante no sólo por su rica historia y variedad arquitectónica, sino por su reciente modernización con la inclusión de hoteles, shoppings y comercios importantes.



La postal de la ciudad es su Puente Colgante santafesino, construido en la década del 20, y reconstruido en 2003, luego de su parcial destrucción durante la inundación de 1983.

El conglomerado Gran Santa Fe es uno de los polos urbanos más importantes del país, dicho sector interurbano supera los 500.000 hab.

Información General

País	 Argentina
Provincia	Santa Fe
Departamento	La Capital
Ubicación	 31° 38' 22" S 60° 41' 43" O
Altitud	19 m.s.n.m.
Superficie	3055 km ²
Fundación	15 de noviembre de 1573 (por Juan de Garay)
Población	369.589 hab.(INDEC, 2001)
Densidad	493.37 hab/Km ²
Tasa de Crecimiento	5,8 % (1991, 2001) %

También hay que destacar que para las ciudades de Santa Fe y Paraná (distanciadas por 25 km) se prevé una muy pronta conurbación. La ciudad de Paraná y sus zonas de influencia cuentan con 300.000 habitantes, elevando a más de 800.000 personas la conurbación.

A mayor distancia de Santa Fe, se observan una serie de núcleos urbanos importantes con un alto grado de autonomía, pertenecientes a los departamentos La Capital, Las Colonias y San Jerónimo. Estas poblaciones son: Esperanza, Coronda, San Carlos, San Jerónimo Norte, Franck, Nelson y Laguna Paiva, por citar las más importantes.

Todas estas ciudades, en un radio de no más de 100km con centro en la ciudad de Santa Fe, constituyen un polo regional de casi 1 millón de habitantes.

Geografía

El Puerto de Santa Fe, se sitúa en el corazón de la Hidrovía Paraguay - Paraná (km 584 del Río Paraná), siendo, aguas arriba, el último Puerto de ultramar apto para operaciones con buques oceánicos. Su ubicación estratégica lo convierte en el eslabón adecuado para unir los nodos de transportes (terrestre-fluvial-oceánico), permitiendo el desarrollo de operaciones de cabotaje nacional e internacional y marítimas internacionales, para cargas unitizadas, containerizadas, graneles y general, integrado por las Regiones Centro, NOA y NEA de Argentina. Su posición privilegiada lo perfila geográficamente como el centro obligado de transferencias de cargas desde y hacia los países situados en la Hidrovía.

Al Puerto se accede desde el Mar Argentino por la ruta denominada "Río Paraná de las Palmas", conformada por el Río de la Plata (Canal de acceso al Puerto de Buenos Aires, Canal Mitre), Río Paraná de las Palmas, Río Paraná Inferior, o por la ruta "Río Paraná Guazú - Paraná Bravo", integrada por el Río de la Plata (Canal de acceso al Puerto de Buenos Aires, Canal Martín García), Río Paraná Guazú, Río Bravo, Río Paraná Inferior. Aguas arriba, se vincula por los Ríos Paraná y Paraguay con Paraguay (Puerto de Asunción), Bolivia (Puertos Aguirre, Suarez) y Brasil (Puertos Ladario, Corumba, Cáceres - km 3432 -)

Industria

El Puerto de Santa Fe, ha iniciado un "Proceso de Reconversión Portuaria", con el objeto de adecuar y modernizar su infraestructura para hacer frente a las necesidades del creciente intercambio comercial de cargas. Este proceso ha comenzado con la elaboración de un Planamiento Estratégico que contemple: la profundización de las vías navegables en la ruta troncal de la Hidrovía Paraguay-Paraná; la potencialidad de cargas extra zona; el análisis de viabilidad técnico-económico de la eventual relocalización

del puerto; la factibilidad de adaptar las actuales estructuras a las formas de transporte intermodal; la creación de zonas de actividades logísticas y la compatibilidad de espacios comunes Puerto-Ciudad. Este proceso de reconversión responderá a una estrategia de conjunto, diseñada y ejecutada por los principales actores político-económico y la dirigencia de la ciudad santafesina.



Entre 2003 y 2008 hubo un fuerte crecimiento productivo permitió la generación de empleo y una recomposición del trabajo técnico calificado, según un informe elaborado por el Centro de Estudios para la Producción (CEP) dependiente de la Secretaría de Industria, Comercio y PyMEs.

Esta situación tuvo un claro correlato en la fuerte caída registrada en el número de desocupados durante el último trimestre del año pasado, cuando se ubicó en el 8,7%, tras haberse ubicado por encima del 20% tras la crisis desatada a fines de 2001.

Según el trabajo, el crecimiento de la industria impacta en la actualidad en la generación de empleo, contrastando con la dinámica de la década del '90, cuando la desarticulación del tejido productivo nacional se tradujo en una precipitada caída de los niveles de ocupación.

La destrucción de empleo durante los '90 fue muy pronunciada aún en aquellos años de crecimiento del producto, ya que la concentración de la actividad en grandes segmentos perjudicó especialmente a las ramas trabajo-intensivas y a las pymes, señaló el CEP.

Durante el período 1991-2001 el crecimiento y la generación de empleo arrojaba una elasticidad negativa de 0,58 puntos. Esto significa que por cada punto de incremento en el PIB caía 0,58 puntos el empleo formal en la Argentina.



Ese cociente pasó a ser de 0,60 puntos "positiva" durante el lapso 2003-2008, implicando una generación de más de medio punto por cada incremento adicional del Producto.

“El incremento de la ocupación industrial es a su vez acompañada por la reversión del proceso de precarización laboral y por la recomposición del trabajo técnico calificado”, destaca la entidad que funciona bajo la órbita del Ministerio de Economía y Producción.

En los últimos cuatro años se crearon unos 311.000 empleos formales en la industria, cifra que, en forma positiva, triplica la cantidad de puestos de trabajo que se vieron afectados entre 1994 y 2001 (estimados en alrededor de 83.000 empleos perdidos).

La inversión no escapa, en tanto, a la relación con la creación de empleo: en el período comprendido entre los años 2003 y 2008 creció 96 por ciento permitiendo un crecimiento del 30 por ciento en el empleo.

Entre 2002 y 2008, en el rubro alimentos y bebidas se crearon 59.000 puestos de trabajo; en textiles y cueros, 54.000 y en el sector metalúrgico unos 46.000 nuevos empleos.

Por el contrario, entre 1994 y 2001, en el rubro textiles y cueros se perdieron 27.000 puestos de trabajo; en material de transporte, 26.000 y en materiales básicas 20.000.

Al analizar la volumen de producción y el número de obreros ocupados por la industria, el informe señala que entre 1993 y 1998, mientras la producción crecía en algunos

períodos hasta por encima del 5 por ciento, el aumento en el número de obreros ocupados era nulo o negativo.

Durante la recesión -que se extendió entre el tercer trimestre del '98 e igual lapso de 2002- tanto la cantidad de obreros ocupados como el nivel de producción fue francamente negativo.

En el último trimestre de 2002, comenzó a observarse una recuperación de la producción, mientras que desde el primer trimestre de 2003 en adelante el crecimiento de la industria tuvo su correlato en la creación de empleo.

Instituciones Educativas

En cuanto a carreras universitarias de grado y posgrado, la ciudad cuenta con una extensa oferta. Se destaca por ser la sede de la Universidad Nacional del Litoral (una de las más antiguas del país), la Universidad Tecnológica Nacional (Facultad Regional Santa Fe) que data de mitad del siglo XX, y la Universidad Católica de Santa Fe.

En la Universidad Nacional del Litoral se instauró por primera vez en toda Latinoamérica la carrera de Ingeniero Químico. La Facultad de Ingeniería Química y sus institutos (especialmente INTEC dependiente del CONICET y la Universidad Nacional del Litoral) gozan de gran prestigio internacional. La UNL recibe alrededor de 25 mil alumnos de los cuales 11 mil provienen de localidades que no pertenecen al Departamento La Capital.

En la UTN concurren algo más de 1600 estudiantes de los cuales 500 son del interior.

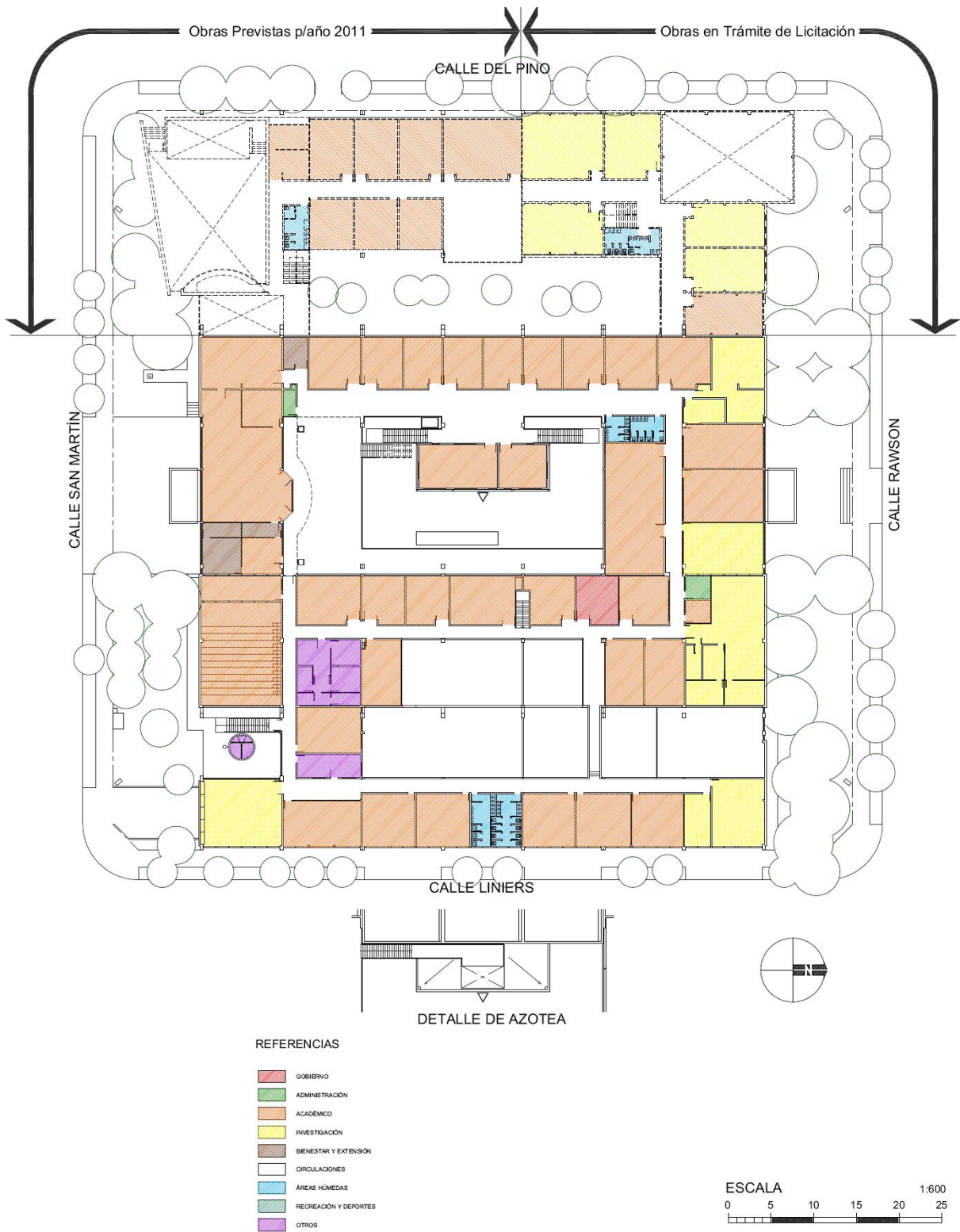
La ciudad cuenta con 24 jardines de infantes, 58 escuela primarias, 12 escuelas primarias nocturnas, 9 escuelas especiales, 22 talleres manuales, 10 Centros Educativos de Capacitación Laboral para Adultos, 10 Centros de Educación para Adultos, 13 escuelas de enseñanza media, 9 de enseñanza media para adultos y 23 escuelas de Educación Técnica.

En cuanto a escuelas privadas, funcionan 27 en el nivel inicial, 47 en la EGB, 29 en Polimodal, 13 Especiales, 8 Técnicas y 19 Institutos Superiores.

Anexo 4C – Plano de la Facultad Regional Delta



Plano de la Planta Baja con extensión prevista para el 2011.



Plano de la Planta Alta con extensión prevista para el 2011.

Anexo 4D – Campana Ciudad Inteligente

Fuente: edición digital de La Auténtica Defensa (05/11/2008)

Se puso en marcha el Plan "Campana Ciudad Inteligente":
Campana será una de las primeras ciudades digitales del país
Un plan de gran envergadura, que permitirá posicionar a nuestra ciudad entre las más desarrolladas tecnológicamente mejorando la calidad de vida de sus habitantes

En la mañana de ayer y en el salón del Honorable Concejo Deliberante, la Intendente Municipal Stella Maris Giroldi presentó el Plan "Campana Ciudad Inteligente", el cual permitirá fortalecer cuatro ejes fundamentales de la sociedad como son la Educación, la Seguridad, el Gobierno y la Producción a través de tecnologías de la información y la comunicación (TIC´S).



La Intendenta Stella Giroldi junto al Decano de la UTN Gustavo Bauer y el Secretario de Gabinete Pedro Orquiguil.

Además, estuvo acompañada por el Secretario General de Gabinete, Dr. Pedro Orquiguil; el Vicepresidente Primero del Honorable Concejo Deliberante, Alejandro Feliz Sánchez; el Decano de la Universidad Tecnológica Nacional Regional Delta, Ing. Gustavo Bauer y el Vicedecano, Miguel Ángel Sosa; la Fiscal General, Dra. Lilliana Maero y Cuerpo de Fiscales; Representantes del Foro Municipal de Seguridad y el Jefe Prefecto de Campana, Prefecto Principal Julio Pereyra.

Este plan se concretará por medio del desarrollo, implementación y puesta en marcha de una infraestructura troncal inalámbrica que permite interconectar diferentes sectores de la comunidad entre sí formando una red mallada, homogénea, estable y de alta calidad.

La digitalización de la ciudad abrirá un gran número de posibilidades tecnológicas, brindándole a la comunidad información y servicios necesarios las 24 hs. Se destaca, el video-vigilancia ciudadana o CIMOPU, que permitirá mayor seguridad mediante cámaras ubicadas en puntos estratégicos de la ciudad conectadas a organismos tales como policía, bomberos, prefectura naval, los cuales pueden actuar con mayor celeridad ante actos delictivos, incendios, accidentes de tránsito, etc.



Momento de la firma de convenio entre el Municipio y la UTN.

Al respecto, el Secretario General de Gabinete, Dr. Pedro Orquiguil expresó estar muy contento por esta puesta en marcha que traerá muchos beneficios "Hoy ponemos en funcionamiento un plan muy importante para nuestra ciudad y que traerá muchos beneficios ya que permitirá fortalecer cuatro ejes fundamentales como son la Seguridad, Educación, Gobierno y Producción".

Posteriormente el Dr. Orquiguil enumeró los puntos sobresalientes que se destacan en el plan y las ventajas del mismo. "El Plan incluye: internet gratuito para todos los establecimientos educativos; wi-fi en espacios públicos; servicio de interconexión de aéreas municipales, empresas y bibliotecas; control de tránsito; monitoreo ambiental; plataforma; comercio electrónico comunal; comunicaciones IP, entre otros servicios".

"Mientras que los beneficios son: mejorar la imagen de la ciudad postulándolo como polo tecnológico; incorporar ventajas competitivas para atraer industrias y negocios; ser modelo para otras municipalidades; mejorar la calidad de los habitantes; potencial el desarrollo tecnológico; mejorar la competitividad de la economía regional; establecer un medio de comunicación entre la Municipalidad y la comunidad; crecimiento económico y productividad; potenciar el uso tecnológicos municipales; fomentar la integración entre las empresas y el gobierno municipal; aumentar la seguridad ciudadana, entre otras ventajas", sintetizó el Secretario de Gabinete.

Finalizando su presentación, destacó que este plan "Campana Ciudad Inteligente", ubicará a nuestra ciudad entre las mejores de la Argentina porque todo el partido de Campana estará conectada a este servicio a través de distintas etapas desde el centro hasta los barrios. "Es un emprendimiento muy costoso que la Municipalidad de Campana esta dispuesto a solventar porque creemos en la importancia y en los beneficios que traerá".

"Hoy estamos poniendo en marcha un plan que propusimos como eje fundamental en la campaña electoral y que hoy estamos concretándolo. Un sueño que engloba a una institución educativa tan importante como es la UTN y la Municipalidad de Campana", concluyó el funcionario.



Uno de los responsables del proyecto explicó sus alcances.

El Decano de la UTN Regional Delta, Ing. Gustavo Bauer, destacó el compromiso y el trabajo que esta llevando adelante la UTN Regional Delta con la comunidad, insertándose y preocupándose para mejorar la calidad de vida de los ciudadanos. Asimismo, agradeció el apoyo constante de la Municipalidad de Campana y la Intendente Stella Maris Giroldi, además de las empresas y entidades que trabajan mancomunadamente para poder concretar los proyectos de la Universidad.

"Campana Ciudad Inteligente surgió de un grupo de alumnos y docentes que trabaja en el área de sistemas y que oportunamente se lo presentamos al entonces intendente Jorge Varela, que tomo con muchísimo entusiasmo esta propuesta. Luego, se agregó la idea de las cámaras para mejorar la seguridad, convirtiéndose en un proyecto más grande e importante aún", sostuvo el Decano de la UTN Regional Delta.

Por su parte, el Director General de Informática y Comunicaciones de la Municipalidad de Campana, Omar Di Gennearo, subrayó la importancia que le da el municipio a la tecnología, invirtiendo desde hace varios años. "Este plan es un proyecto de alta tecnología que no van a encontrar en muchas ciudades de las características de Campana. Asimismo, ayudará a disminuir la brecha tecnología que existe entre los países desarrollados mejorando la calidad de vida de todos los ciudadanos de nuestro partido".



Funcionarios del Gobierno Municipal.

Posteriormente, Cristian Pacheco de la UTN, se refirió a las características técnicas del proyecto.

Rubricando la presentación oficial, se firmó un convenio de colaboración entre la Municipalidad de Campana y la Facultad Regional Delta Campana, para el Plan Integral Campan Ciudad Digital - CIMOPU.

Anexo 6A – Desarrollo De La Técnica De Puntos De Función

Fuente: Estimación de Proyectos de Software (Lic. Ana Moreno Sánchez Capuchino – ITBA/UPM, 2006)

Los puntos de función miden el software cualificando la funcionalidad que proporciona externamente, basándose en el diseño lógico del sistema. Por lo tanto, en el caso de subsistemas diseñados independientemente los puntos de función se calcularán para cada uno de ellos, y luego se sumarán. Por ejemplo, cuando un sistema que proporcione por un lado una funcionalidad on-line y por otro lado una funcionalidad batch⁴, se han diseñado independientemente los dos subsistemas que proporcionan cada funcionalidad.

Los objetivos de los puntos de función son:

- Medir lo que el usuario pide y lo que el usuario recibe.
- Medir independientemente de la tecnología utilizada en la implantación del sistema.
- Proporcionar una métrica de tamaño que dé soporte al análisis de la calidad y la productividad.
- Proporcionar un medio para la estimación del software.
- Proporcionar un factor de normalización para la comparación de distintos software.

Además de estos objetivos el proceso de contabilizar los puntos de función debería ser:

- Simple como para minimizar la carga de trabajo de los procesos de medida.
- Conciso en sus resultados.

El análisis de los puntos de función se desarrolla considerando cinco parámetros básicos externos del sistema:

1. Entrada (EI, del inglés External Input).
2. Salida (EO, del inglés External Output).
3. Consultas (EQ, del inglés External Query).
4. Grupos de datos lógicos internos (ILF, del inglés Internal Logic File).
5. Grupos de datos lógicos externos (EIF, del inglés External Interface File).

Con estos parámetros, se determinan los puntos de función sin ajustar. A este valor, se le aplica un factor de ajuste obtenido en base a unas valoraciones subjetivas sobre la aplicación y su entorno. Es decir, las características generales del sistema.

La aplicación de la técnica de los puntos de función comprende los siguientes pasos:

- Definición de los límites del sistema.
- Definición de parámetros.
- Valoración de la complejidad.
- Análisis de las características generales del sistema.

⁴ Proceso por lotes

Definición De Los Límites Del Sistema

El límite es utilizado para definir el alcance del sistema y ayudar a identificar los parámetros externos. Existen tres visiones de los límites del sistema, dependiendo de la utilización que quiera realizarse de la técnica:

La aplicación límite del producto: abarca la totalidad de la aplicación y se realiza la cuenta de puntos al final del desarrollo del proyecto cuando se gestiona el grupo de mantenimiento o cuando la organización inicia el uso de FPA. Este tipo de cuenta puede ser también obtenida de un sistema en funcionamiento.

Límite inicial del proyecto a desarrollar: es un tipo de conteo similar al anterior, la diferencia está en que se deriva de los requisitos de un sistema que no existe aún.

Límite del proyecto de mejora: esta situación surge cuando ya existe el sistema y se trata de obtener nuevas versiones del mismo. La utilización de FPA en proyectos de mejora difiere de las anteriores en que se consideran adiciones, modificaciones o anulaciones de funcionalidades, en lugar de la totalidad del sistema. No se puede caer en la trampa de calcular los puntos del sistema total antes y después de las mejoras y luego abstraer uno de otro. Existe un elemento subjetivo en la determinación de los límites del sistema y obviamente un cambio en ellos cambiará el total de puntos de función. Aunque esto podría parecer una aproximación poco científica, en la práctica la orientación que el analista debería seguir es considerar el problema como un todo discreto. La fórmula que permite calcular los puntos de función de un nuevo desarrollo es la siguiente:

$$FPA = FP \times AF$$

Donde:

FP = es el número de puntos de función sin ajustar de la aplicación.

AF = es el factor de ajuste de la aplicación.

El cálculo de los puntos de función de un proyecto de mejora se puede obtener mediante la fórmula:

$$(ADD + CHGA) \times VAFA + (DEL \times VAFB) = EFP$$

Donde:

EFP = es el número de puntos de función del proyecto de mejora.

VAFB = es el factor de ajuste de la aplicación antes del proyecto de mejora.

ADD = es el número de puntos de función de aquellas funciones que se añadirán al proyecto como consecuencia de la mejora.

CHGA = es el número de puntos de función sin ajustar de aquellas funciones que serán modificadas por el proyecto de mejora. Este número refleja las funciones después de la modificación.

DEL = es el número de puntos de función sin aquellas funciones que serán eliminadas en el proceso de mejora.

VAFA = es el factor de ajuste de la aplicación después del proyecto de mejora.

Definición De Parámetros

Para poder determinar la existencia de los componentes que contribuirán al total final hay que definirlos previamente. La definición que vamos a considerar aquí es la realizada por IFPUG en 1994, última versión. Estos componentes pueden ser clasificados como tipos de funciones y son de dos clases: función datos o función transacciones.

Función Datos

Los tipos de función datos representan la funcionalidad proporcionada a los usuarios para cumplir con sus requisitos de datos internos y externos. Son de dos tipos: archivos lógicos internos y archivos de interfase externos.

Los archivos lógicos internos (ILF) son un grupo de datos lógicamente relacionados, identificables por los usuarios o información de control, mantenidos y utilizados dentro de los límites de la aplicación. Ejemplos de ILF son:

- Archivos maestros.
- Aplicaciones de seguridad de datos.
- Datos de auditoría.
- Mensajes help.
- Mensajes de error.
- Datos de back-up, si el usuario lo requiere.
- Archivos internos lógicos mantenidos por más de una aplicación.

Los archivos interfase externos representan un grupo de datos relacionados lógicamente identificables por el usuario o información de control utilizada por la aplicación, pero mantenida por otra aplicación.

Función transacción

El tipo de función transacción comprende tres tipos de función:

- Entradas externas (EI): mantiene datos almacenados internamente.
- Salidas externas (EO): datos de salida de la aplicación.
- Consultas externas (EQ): combinación de una entrada (pregunta) y de una salida (respuesta).

Vamos a comentar brevemente cada una de ellas.

Entradas externas

Las entradas externas son datos o información de control que se introducen en la aplicación desde fuera de sus límites. Estos datos mantienen un archivo lógico interno. La información de control está constituida por datos utilizados por un proceso dentro de los límites de la aplicación para asegurar el cumplimiento de los requisitos del negocio definidos

por los usuarios. Esta información de control puede mantener directamente un archivo lógico interno. Una entrada externa debería ser considerada única si tiene un formato distinto de las demás o el diseño lógico requiere una lógica de procesamiento distinta de otra entrada externa del mismo formato. En otras palabras, una entrada externa se considera única si los datos en un archivo lógico interno (ILF) y el formato de entrada son únicos o la lógica del proceso es única. Para cada proceso identificado que actualiza un archivo lógico interno:

- Hay que considerar cada formato de entrada como un proceso distinto, los datos utilizados por el proceso pueden tener distintos formatos.
- Hay que sumar una unidad a cada entrada externa por cada actividad de mantenimiento de datos realizada (sumar, cambiar, borrar).

Ejemplos de este tipo de función son:

- Las transacciones: datos introducidos para mantener archivos lógicos internos.
- Las pantallas de entrada: hay que añadir una unidad a entradas externas por cada función que mantiene un archivo lógico interno. Por ejemplo, si los datos introducidos en esa pantalla pueden añadir, cambiar y borrar información en un archivo lógico interno, se contarían tres entradas externas.
- Las entradas por lotes: por cada proceso único que mantiene un archivo lógico interno se debe añadir una entrada externa por cada adición, modificación o borrado de datos.
- Un archivo físico de entrada, cuando se analiza lógicamente, corresponde a una entrada externa o varias, dependiendo de los tipos de registros contenidos y del proceso requerido para su tratamiento.
- Asimismo dos o más archivos físicos de entrada pueden corresponder a una entrada externa si el proceso lógico y el formato son idénticos para cada uno de los archivos.
- Las entradas externas duplicadas: si distintos procesos de entrada solicitados expresamente por el usuario duplican una entrada externa, son contados independientemente cada uno. Un ejemplo puede ser un ingreso en una cuenta bancaria que se puede hacer por un cajero automático o a través de una operación normal, siendo el mismo tipo de entrada.

En cambio, no son entradas externas:

- Los datos referenciados utilizados por la aplicación pero no mantenidos como archivos lógicos internos.
- La entrada de una consulta.
- Los generadores de informes.
- Las pantallas de conexión que no mantengan un archivo lógico interno.

Salidas externas

Las salidas externas son datos o información de control que sale de los límites de la aplicación. Esta salida debe ser considerada única si tiene un formato único o si el diseño lógico requiere un proceso lógico distinto de otras salidas del mismo formato.

Deben considerarse salidas externas:

- La transferencia de datos a otras aplicaciones: datos que residen en un archivo lógico interno que son formateados y procesados para ser utilizados por otra aplicación.
- Las salidas son identificadas basadas en los procesos requeridos para el tratamiento de los datos. Un archivo físico de salida, cuando se analiza lógicamente, puede corresponder a varias salidas. De una manera similar, dos o más archivos físicos de salida pueden corresponder a una salida externa si el proceso lógico y los formatos son idénticos para cada uno de ellos. Un método para identificar múltiples salidas externas es ver cuántos tipos de registros distintos hay en el archivo.
- Los informes: cada informe producido por la aplicación se cuenta como una salida externa.
- Dos informes que tengan el mismo formato pero los datos de origen sean distintos, ya que se generan en distintos procesos, se contarán como dos salidas. Dos informes idénticos, producidos en diferentes soportes debido a requisitos específicos de los usuarios, se cuentan como salidas distintas.
- Los informes on-line que no corresponden a la salida de una consulta, se contarán también como una salida.
- Los mensajes de error/configuración: no se contarán si están asociados a una consulta.
- Los gráficos: cada gráfico distinto, solicitado por el usuario, debería ser contado como una salida. Así, si unos datos estadísticos se presentan en formato de tabla, diagrama de barras, y trata se contarán como tres salidas.
- Los generadores de informes: una salida desarrollada por el usuario con un generador de informes debería ser contada como una salida para cada tipo de informe especificado. Si el usuario solicita una facilidad de generación de informes como parte de la aplicación para ser confeccionados por él mismo, la cuenta será la siguiente:
 - Debería contarse una entrada externa por cada parámetro para la definición de informes o comando (ej. select, compare, sort merge, calculase, format, etcétera) utilizado por el usuario para controlar la generación del informe.
 - Debería contarse una salida por el informe total.
 - Debería contarse un archivo lógico si se crea un nuevo archivo y éste se salva.

No se deben contar como salidas:

- Las ayudas.
- Las distintas formas de invocar la misma salida lógica.
- Los mensajes de error/confirmación asociados con tipos de función distintos de entradas externas. Por ejemplo, no se contabilizarán como salida los mensajes de error/confirmación asociados a una consulta externa.
- Los informes múltiples/valores únicos de datos: informes idénticos con el mismo formato y la misma lógica de proceso, pero que existen debido a distintos valores de datos, no se cuentan como salidas distintas. Por ejemplo, dos informes idénticos en formato y construcción, el primero de los cuales contiene nombres comenzando desde la A a la L y el segundo desde la M a la Z se cuenta como una única salida.

- Las totalizaciones: los informes de totales no constituyen una salida.
- Los informes ad hoc: cuando el usuario dirige y es responsable de la creación mediante la utilización de lenguajes como FOCUS o SQL de un número indefinido de informes, no se cuentan como salidas.

Consultas externas

Las consultas representan los requisitos de información a la aplicación en una combinación única de entrada / salida que se obtiene de una búsqueda de datos, no actualiza un archivo lógico interno y no contiene datos derivados. Una consulta se considera única si tiene un formato distinto de otras consultas, ya sea en entrada o salida, o si el diseño lógico requiere ediciones distintas a las de otras consultas. Se entiende por datos derivados aquellos que requieren un proceso distinto a la búsqueda directa, edición o clasificación de información de archivos lógicos internos o archivos interfases externos.

Ejemplos de consultas son:

- La búsqueda inmediata de datos.
- Las consultas no explícitas: las pantallas de modificación/borrado que proporcionan capacidad de búsqueda de datos antes de la funcionalidad de cambio/borrado se consideran como consultas. Si la entrada y salida de la consulta son idénticas en las funciones de modificación y borrado, se contará una sola consulta.
- Los menús con consultas implícitas: las pantallas de menú que proporcionan una selección de pantallas y entradas para la búsqueda de datos para la pantalla llamada, se cuenta como una consulta.
- Pantallas de conexión: las pantallas de logon que proporcionarían seguridad se cuentan como una consulta.
- Las ayudas: son una consulta donde la entrada y la salida (texto) son únicas. Un texto que puede ser accedido o mostrado en pantalla mediante distintas peticiones o diferentes áreas de una aplicación, se cuenta como una consulta. Se pueden distinguir dos categorías de ayudas como son consultas típicas:
 - a) Ayudas a plena pantalla: es una facilidad que proporciona una salida a pantalla como consecuencia de una llamada, también a través de una pantalla. Se cuenta como una consulta de baja complejidad por aplicación, sin tener en cuenta el número de pantallas devueltas.
 - b) Ayudas por campos: es una facilidad que se proporciona dependiendo de la posición del cursor o algún otro método de identificación, mostrándose documentación específica a dicho campo. Se cuenta como una consulta de baja complejidad por aplicación.
- Las salidas duplicadas: consultas iguales que producen una salida en diferentes soportes, como consecuencia de especificaciones del usuario, se cuentan como consultas distintas.

- Las salidas gráficas: cada salida gráfica diferente solicitada por el usuario, debería contarse como una consulta.
- Tutoriales: los sistemas de software relativos a la formación de usuarios deberían contarse como un sistema distinto.

No se consideran como consultas:

- Los mensajes de error/confirmación.
- La utilización de distintos métodos de llamada a la misma consulta. Puede ocurrir que en una organización en particular surja una situación que no esté cubierta por las guías existentes para contar puntos de función. Este método refleja que ésta es una técnica en evolución. En tales casos el técnico debe tomar la decisión de formular una regla, basada en su experiencia personal, así como en la de otros. Lo más importante es documentar la regla y aplicarla consistentemente.

VALORACIÓN DE LA COMPLEJIDAD

Para cada uno de los parámetros externos se ha de indicar su complejidad como baja, media o alta. Para las entradas, salidas y consultas, se puede evaluar su complejidad en función del número de campos que contengan y del número de archivos a los que hagan referencia. Para los archivos, por el contrario, su complejidad vendrá dada en función del número de registros y de campos que contengan.

Una vez definida la complejidad de cada parámetro se aplica el cálculo de la tabla 3:

Parámetro	Complejidad	X	Peso	Total
Entrada	Alta	x	6	=
	Media	x	4	=
	Baja	x	3	=
Salida	Alta	x	7	=
	Media	x	5	=
	Baja	x	4	=
Fichero lógico interno	Alta	x	15	=
	Media	x	10	=
	Baja	x	7	=
Fichero lógico Externo	Alta	x	10	=
	Media	x	7	=
	Baja	x	5	=
Consultas	Alta	x	6	=
	Media	x	4	=
	Baja	x	3	=

Tabla 3: Cálculo de complejidad de los puntos de función

La suma total da como resultado los puntos de función sin ajustar del sistema. A continuación se describe un ejemplo:

Supongamos que deseamos desarrollar un software que tiene los siguientes componentes:

- 2 entradas simples
- 3 entradas medias
- 1 salida compleja
- 2 interfases externas medias
- 10 consultas simples

El total de puntos de función sin ajustar sería:

$$FP = (2 \times 3) + (3 \times 4) + (1 \times 7) + (2 \times 7) + (10 \times 3) = 69$$

ANÁLISIS DE LAS CARACTERÍSTICAS GENERALES DEL SISTEMA

Una vez obtenidos el total de puntos de función sin ajustar, debe realizarse un ajuste del mismo en función de las características generales del sistema. Estas características son:

1. Comunicación de datos.
2. Funciones distribuidas.
3. Rendimiento.
4. Configuraciones fuertemente utilizadas.
5. Frecuencia de transacciones.
6. Entrada on-line de datos.
7. Diseño para la eficiencia del usuario final.
8. Actualización on-line.
9. Procesos complejos.
10. Utilización en otros sistemas.
11. Facilidad de instalación.
12. Facilidad de operación.
13. Instalación de múltiples sitios.
14. Facilidad de cambio.

En función de estas catorce características se calcula el grado de influencia, del modo que se mostrará a continuación. Una vez calculado el grado de influencia de las

características, TDI (del inglés Total Degree of Influence), se puede llegar al valor del factor de ajuste mediante la fórmula:

$$AF = (TDI \times 0,01) + 0,65$$

El valor final de puntos de función ajustados será:

$$FPA = FP \times AF$$

Existe un debate general sobre las características generales del sistema, ya que en gran parte su evaluación es subjetiva, y por otro lado su valor como multiplicador es muy bajo. Sin embargo, forma parte de la técnica y en el futuro se prevé que sea uno de los aspectos más importantes en la evolución de la misma. Vamos a estudiar la valoración que hace de estas características el IFPUG. Cada una de ellas se evaluará de 0 a 5, según las guías que se indican a continuación, el TDI se obtendrá como suma de los valores de cada una de ellas.

1. Comunicación de datos.

Los datos e información de control utilizados en la aplicación se reciben o son enviados a través de medios de telecomunicación. Los terminales conectados localmente a la unidad de control son considerados como medios de comunicación. Los valores utilizados son los que se muestran a continuación.

0. La aplicación es por lotes o utilizando una computadora personal.
1. La aplicación es por lotes pero existe una entrada de datos o impresiones remotas.
2. La aplicación es por lotes pero son remotas la entrada de datos o la impresión.
3. Entrada on-line de datos a un proceso por lotes o sistema de consultas.
4. Más de un equipo front-end, y la aplicación soporta un solo tipo de protocolo de comunicaciones.
5. Más de un equipo front-end, y la aplicación soporta más de un tipo de protocolo de comunicaciones

2. Funciones distribuidas

Son características de la aplicación que permiten la existencia de datos o procesos distribuidos dentro del límite de la aplicación. Los valores posibles son los siguientes:

0. No existe este tipo de funciones en la aplicación.
1. La aplicación prepara datos para que el usuario final los procese en otro componente del sistema.
2. Los datos se transfieren y procesan en otro componente del sistema, pero no por el usuario final.
3. El proceso distribuido y la transferencia de datos son on-line y sólo en una dirección.
4. El proceso distribuido y la transferencia de datos son on-line en ambas direcciones.
5. Los procesos se desarrollan dinámicamente en el componente más apropiado del sistema.

3. Rendimiento

Los objetivos de rendimiento del sistema, definidos o aprobados por el usuario, tanto en tiempo de respuesta como en volumen de datos a procesar se verán influenciados por el diseño, desarrollo, instalación y soporte de la aplicación. A continuación se muestra la valoración del rendimiento.

0. No existen requisitos específicos de rendimiento.
1. Rendimiento y requisitos de diseño han sido definidos pero no requieren ninguna acción especial.
2. El tiempo de respuesta o la capacidad de proceso es crítico durante las horas pico. No se requiere ningún diseño especial para la utilización de la Unidad Central de Proceso (UCP) del equipo. Los procesos demorados se ejecutan al día siguiente.
3. El tiempo de respuesta o la capacidad de proceso es crítico durante todas las horas de operación. No se requiere un diseño especial para la utilización de la UCP.
4. Los requisitos de rendimiento por parte de los usuarios son suficientemente estrictos como para requerir un análisis de rendimiento en la fase de diseño.
5. Además, hay que utilizar herramientas para el análisis de rendimiento durante el diseño, desarrollo y/o fase de implementación para verificar los requisitos de rendimiento.

4. Configuraciones fuertemente utilizadas

Es una característica de la aplicación que requiere consideraciones especiales de diseño debido a las limitaciones de los equipos a utilizar. Los valores son:

- 0 No existen restricciones de ningún tipo.
- 1 Existen restricciones operativas, pero no requieren un esfuerzo especial para conseguirlas.
- 2 Existen algunas restricciones de seguridad o tiempo.
- 3 Existen requisitos específicos de procesador para algunas partes de la aplicación.
- 4 Las restricciones definidas en el equipo central o procesador dedicado obligan a limitaciones en la aplicación.
- 5 Además de las características del punto 4, existen limitaciones en los componentes distribuidos del sistema.

5. Frecuencia de transacciones

La frecuencia de transacciones es alta e influye sobre el diseño, desarrollo, instalación y soporte de la aplicación. Los valores a asignar son:

0. No existe una definición del período pico de transacciones.
1. Se conoce el período pico (mensual, trimestral, estacional, anual).
2. Se conoce el período semanal.
3. Se conoce el período pico diario.
4. La frecuencia de transacciones definida por el usuario en los requisitos de la aplicación o acuerdos de nivel de servicio son suficientemente altos como para requerir análisis de rendimiento de tareas durante la fase de diseño.
5. La frecuencia de transacciones definida por el usuario en los requisitos de la aplicación o acuerdos de nivel de servicio son suficientemente altos como para requerir el uso de análisis de rendimiento de tareas y de herramientas de medida del rendimiento en el diseño, desarrollo y/o fase, de instalación.

6. Entrada de datos on-line

Los valores posibles son:

0. Todas las transacciones se procesan por lotes.
1. 1% al 7% de las transacciones son interactivas.
2. 8% al 15% de las transacciones son interactivas.
3. 16% al 23% de las transacciones son interactivas.
4. 24% al 30% de las transacciones son interactivas.
5. Más del 30% de las transacciones son interactivas.

7. Eficiencia del usuario final

Las funciones on line proporcionadas ponen énfasis en un diseño que incremente la eficiencia del usuario final. Estas funciones pueden ser:

- Ayudas a la navegación.
- Menús.
- Ayudas/Documentación on-line
- Movimiento automático del cursor.
- Scrolling.
- Impresión remota.
- Teclas de función preasignadas.
- Selección mediante cursor de datos en pantalla.
- Uso amplio de facilidades de video.
- Interfase del mouse.
- Ventanas.
- Racionalización del uso de pantallas para realizar una función de negocio.
- Soporte de dos lenguajes (contar como cuatro funciones).
- Soporte multi-lenguaje (contar como seis funciones).

Los valores son:

0. Nada de lo anterior.
1. 1-3 de las funciones anteriores.
2. 4-5 de las funciones anteriores
3. 6 o más, pero no existen requisitos del usuario respecto a la eficiencia.
4. 6 o más, pero están definidos los requisitos de eficiencia del usuario que obligan a diseñar tareas que tienen en cuenta factores humanos; por ejemplo, minimizar el número de tecleos o uso de máscaras.
5. 6 o más, y hay requisitos del usuario sobre eficiencia que obligan a utilizar herramientas especiales y procesos para demostrar que los objetivos se han alcanzado.

8. Actualización on-line

La aplicación proporciona actualización on-line de los archivos lógicos internos. Los valores se muestran a continuación:

0. Ninguno.
1. Actualización on-line de 1 a 3 archivos. El volumen de actualización es bajo y la recuperación fácil.
2. Actualización on-line de 4 o más archivos. El volumen de actualización es bajo y la recuperación es baja.
3. Actualización importante de los archivos lógicos internos.
4. Además de la protección contra la pérdida de datos es esencial y ha sido especialmente diseñada y programada en el sistema.
5. Además del punto 4, los altos volúmenes de transacciones requieren que sea considerado el costo de los procesos de recuperación. Los procedimientos de recuperación están altamente automatizados con intervención mínima del operador.

9. Procesos complejos

Es una característica de la aplicación. Las categorías existentes son:

- Controles especiales (procesos de auditoría) y/o aplicaciones de seguridad.
- Proceso lógico complejo.
- Procesos matemáticos complejos.
- Excesivas excepciones de proceso dando lugar a transacciones incompletas que deben ser procesadas de nuevo; por ejemplo, transacciones incompletas en cajeros automáticos, falta de datos obligatorios, etc.
- Manejo de dispositivos complejos; por ejemplo, multimedia, independencia de dispositivos, etc.

Los valores son:

0. Nada de lo anterior.
1. Uno de los anteriores.
2. Dos de los anteriores.
3. Tres de los anteriores.
4. Cuatro de los anteriores.
5. Todos los anteriores.

10. Reutilización

La aplicación y el código han sido diseñados específicamente desarrollados y soportados para ser utilizados en otras aplicaciones. Los valores aparecen en la tabla 13.

0. No reusable,
1. Se utiliza código reusable dentro de la aplicación.
2. Menos del 10% de la aplicación tiene en cuenta las necesidades de más de un usuario.
3. El 10% o más de la aplicación tiene en cuenta las necesidades de más de un usuario.
4. La aplicación fue empaquetada expresamente y/o documentada para ser fácilmente reusable. La aplicación es adaptada por el usuario a nivel de código fuente.
5. La aplicación fue empaquetada expresamente y/o documentada para ser fácilmente reusable. La aplicación es adaptada por el usuario por medio de parámetros de mantenimiento.

11. Facilidad de instalación

La facilidad de conversión e instalación son características de la aplicación. Durante la fase de pruebas del sistema se proporcionarán y probarán un plan de conversión e instalación, así como herramientas para la conversión. Los valores son:

0. No se realizaron consideraciones ni se requirieron desarrollos especiales para la instalación por parte del usuario.
1. No se realizaron consideraciones especiales por el usuario pero se requirieron desarrollos especiales de instalación.
2. Los requisitos de conversión e instalación fueron definidos por el usuario y las guías para la conversión e instalación fueron desarrolladas y probadas. El impacto de la conversión en el proyecto no se considera importante.
3. Los requisitos de conversión e instalación fueron definidos por el usuario y las guías para la conversión e instalación fueron proporcionadas y probadas.
4. Además del punto 2, se proporcionarán y probarán la conversión automática y herramientas para la instalación.

5. Además del punto 3, se proporcionarán y probarán la revisión automática y las herramientas para la instalación.

12. Facilidad de operación

La facilidad de Operación es una característica de la aplicación. Se proporcionarán y probarán durante la fase de pruebas del sistema un arranque eficaz, procedimientos de respaldo y recuperación. La aplicación minimiza la necesidad de actividades manuales, tales como manejo de papel o intervención manual durante la operación del sistema. Los valores se muestran a continuación:

0. No se definieron por parte del usuario necesidades especiales de operación o respaldo distinto de las normales.
- 1-4. Seleccionar, valorando como uno, cada una de las siguientes solicitudes realizadas a la aplicación:
 - Procesos eficaces de arranque, respaldo y recuperación pero con intervención del operador (contar como 2).
 - La aplicación minimiza la necesidad de montajes de discos u otros dispositivos de almacenamiento.
 - La aplicación minimiza la necesidad de manejo de papel.
5. La aplicación debe diseñarse sin intervención de operadores; es decir, el operador no debe intervenir más que para arrancar y parar la aplicación. Uno de los elementos de la aplicación es la recuperación automática de errores.

13. Instalación en distintos lugares

La aplicación se diseñará y desarrollará para ser instalada y, mantenida en distintos lugares por distintas organizaciones.

0. No existen requisitos del usuario para considerar la necesidad de más de un usuario o lugar de instalación.
1. Se necesita diseñar la aplicación para ser utilizada en múltiples lugares pero funcionará bajo entornos idénticos de hardware y software.
2. Se necesita diseñar la aplicación para ser utilizada en múltiples lugares y funcionará bajo un entorno de hardware y software similares.
3. Se necesita diseñar la aplicación para ser utilizada en distintos lugares y funcionará bajo entornos distintos de hardware y software.
4. Deberán ser proporcionados y probados la documentación y los planes de soporte de la aplicación para ser utilizados en distintos lugares, en el modo que se indicó en los apartados 1 y 2.
5. Deberán ser proporcionados y probados la documentación y los planes de soporte de la aplicación para ser utilizada en distintos lugares, en el modo que se indicó en el apartado 3.

14. Facilidad de cambios

La aplicación debe ser específicamente diseñada, desarrollada y mantenida para facilitar el cambio. Los siguientes son ejemplos de facilidades de cambios:

- Capacidad para proporcionar flexibilidad en las consultas y obtención de informes.
- Los datos de la aplicación relativos al negocio se mantienen en tablas por parte de los usuarios.

0 No existe ninguna especificación Por parte de los usuarios en este sentido.

1-5 Se seleccionará alguna de estas opciones:

- Facilidad para realizar consultas o informes simples tales como la utilización de operadores lógicos AND/OR sobre un archivo lógico interno (se contará como 1).
- Facilidad para realizar consultas o informes de complejidad media tales como la utilización de operadores lógicos AND/OR sobre más de un archivo lógico interno (se contará como 2).
- Facilidad para realizar consultas/informes complejos (se contarán como 3).
- Se mantendrán datos de control en tablas que serán mantenidas por los usuarios a través de procesos interactivos on-line, pero los cambios no serán efectivos hasta el siguiente día de funcionamiento de la aplicación (se contará como 1).
- Igual que el caso anterior, pero los cambios serán efectivos inmediatamente (se contará como 2).

Ejemplo

Supongamos que el software para el que calculamos los puntos de función sin ajustar tiene los siguientes valores para las características generales.

1.	Comunicación de datos		3
2.	Funciones distribuidas		0
3.	Rendimiento		3
4.	Configuraciones fuertemente utilizadas	0	
5.	Frecuencia de transacciones	1	
6.	Entrada <i>on-line</i> de datos,		5
7.	Diseño para la eficiencia del usuario final		2
8.	Actualización <i>on-line</i> .		1
9.	Procesos complejos	2	
10.	Utilización en otros sistemas	0	
11.	Facilidad de instalación		0
12.	Facilidad de operación		3
13.	Instalación de múltiples sitios	0	
14.	Facilidad de cambio	2	
			23

Cálculo del factor de ajuste:

$$AF = 0,65 + (0,01 \times 23) = 0,88$$

Medida de puntos de función:

$$FPA = FP \times AF = 69 \times 0,88 = 61$$

Para usar eficientemente los puntos de función, se emplean unos ratios relativos a las siguientes métricas:

- Productividad: indica el número de puntos de función que puede desarrollar una persona en un mes.
- Calidad: indica el número de errores que supuestamente se cometerán por punto de función.
- Costo: indica el valor monetario que costará a la empresa el desarrollo de un punto de función.
- Documentación: indica el número de páginas de documentación que se generará por punto de función.
- Líneas de código: indica el número de líneas de un determinado lenguaje de programación, que se escribirán por punto de función.

Estos ratios vendrán medidos en:

-
- Productividad = puntos función/persona-mes
 - Calidad = errores/punto función
 - Costo = pesos/punto función
 - Documentación = páginas/punto función
 - Líneas de código = líneas/punto función

La clave de la utilización de esta técnica radica en la obtención de estos ratios que serán específicos de cada organización, y que nos darán información sobre el tamaño de la aplicación. Estos ratios se obtendrán de proyectos anteriores que se hayan desarrollado en la organización.

Anexo 6B – Método De Estimación Cocomo

Fuente: Estimación de Proyectos de Software (Lic. Ana Moreno Sánchez Capuchino – ITBA/UPM, 2006)

Este modelo se aplica a los desarrollos que siguen el ciclo de vida en cascada (en nueve etapas), y corresponde a las siguientes fases:

- Planificación y definición de requisitos.
- Diseño de producto.
- Diseño detallado.
- Codificación y pruebas unitarias.
- Integración y pruebas.
- Implantación.
- Explotación y Mantenimiento.
- Verificación y validación.
- Gestión de configuración.

Existen tres modos de desarrollo de software según COCOMO: orgánico, semilibre y semirígido, según las características de la aplicación y del entorno de desarrollo. A cada uno de estos modelos se le puede aplicar tres métodos de estimación distintos: básico, intermedio y detallado. Cuando un ingeniero de software está ante un proyecto a estimar, lo primero que debe hacer para aplicar COCOMO es situar su proyecto en el espacio de dos dimensiones (modo, modelo) de la figura 1.

En la figura 1, P_x es un proyecto que va a ser desarrollado según un modelo semilibre, dado el tipo de aplicación y el equipo del que se dispone, y se va a estimar según COCOMO detallado, dada la exactitud que se requiere de la estimación.

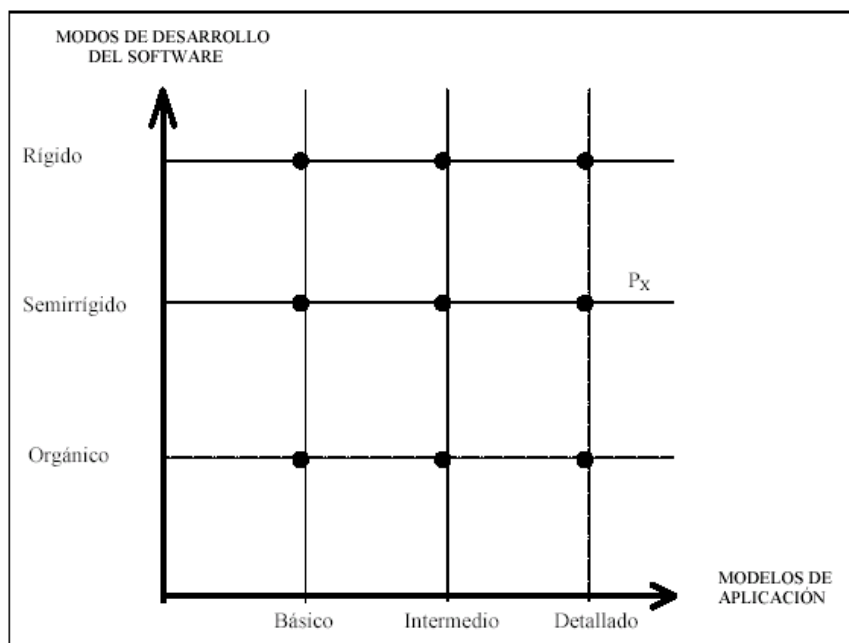


Figura 5. Espacio para situar un proyecto a estimar por COCOMO

MODOS COCOMO

Modo Orgánico

- El proyecto se desarrolla en equipos relativamente pequeños dentro de la propia empresa.
- Muchas personas relacionadas con el proyecto tienen amplia experiencia trabajando en sistemas similares dentro de la propia organización y tienen un buen conocimiento de cómo el sistema que se está desarrollando contribuirá a los objetivos de la organización.
- Esto significa que muchas personas podrán contribuir al proyecto desde las primeras etapas, sin generar una sobrecarga de comunicación importante.
- El proyecto se desarrolla en un entorno relativamente relacionado en cuanto a exigencia por parte de los usuarios para que el software cumpla las especificaciones y pueda ser desarrollado más fácilmente. Ésta es otra razón para una mayor productividad.
- Se desarrolla en un entorno generalmente estable, con muy pequeña probabilidad de coincidencia en el desarrollo de un nuevo hardware u operaciones desconocidas.
- Mínima motivación para terminar el proyecto antes de lo previsto.
- Proyectos de tamaño relativamente pequeño. Como máximo 50 KDSI (miles de líneas de código).

Modo Semilibre

- El equipo del proyecto tiene un nivel medio de experiencia en proyectos similares.
- El equipo es una combinación de personal experto e inexperto.
- Algunos miembros del proyecto tienen experiencia en aspectos del proyecto y otros no.
- El tipo de proyectos representativo podría ser un sistema de proceso de transacciones con interfases muy poco rigurosas en algunos casos y muy rígidas en otros.
- El tamaño del producto llega a las 300 KDSI.

Modo Rígido

- Proyectos que deben desarrollarse dentro de unas limitaciones muy estrictas.
- El producto debe explotarse dentro de un entorno muy acoplado de hardware, software, normativa y procedimientos operativos, tales como sistemas de transferencia electrónica de fondos o control de tráfico aéreo.
- Los costos de cambiar algo en este complejo entramado son tan altos, que sus características se consideran inmodificables, así que el software debe realizarse estrictamente conforme a las especificaciones.
- Como resultado, estos proyectos no admiten negociar cambios en el software modificando los requisitos e interfases del usuario, por lo que el esfuerzo en verificaciones y validación así como la gestión de configuraciones, es muy alto.
- Estos proyectos se desarrollan en áreas generalmente desconocidas, lo cual lleva inicialmente a equipos pequeños de analistas y, a una sobrecarga de comunicación importante durante el desarrollo.
- Una vez que el proyecto ha terminado la fase de diseño, la mejor estrategia para continuar el desarrollo es constituir un equipo grande de programación para realizar el diseño

- detallado, codificación y, pruebas en paralelo. Esta estrategia conduce a puntas en cuanto a personal y a un mayor consumo de esfuerzo mayor que en los restantes niveles.
- Se aplica a proyectos de cualquier tamaño.

MODELOS COCOMO

Modelo Básico

Se suele aplicar en los desarrollos de productos pequeños/medios, desarrollados por personal de la propia empresa en modo organización. Aunque también puede aplicarse al resto de los modos. Las ecuaciones de estimación de esfuerzo y tiempo de desarrollo para cada modo de desarrollo:

Orgánico:	MM =	2,4 (KDSI) ^{1,05}
	TDEV =	2,5 (MM) ^{0,38}
Semilibre:	MM =	3,0 (KDSI) ^{1,12}
	TDEV =	2,5 (MM) ^{0,35}
Rígido:	MM =	3,6 (KDSI) ^{1,20}
	TDEV =	2,5 (MM) ^{0,32}

Donde,

KDSI = número de instrucciones de código en miles.

MM = esfuerzo medido en meses/hombre.

TDEV = duración en meses.

Ejemplo

Supongamos que queremos desarrollar un programa que se ha estimado tendrá 32.000 instrucciones; y en base a las características de la aplicación decidimos tratarlo en el modo orgánico.

¿Cuáles serán el esfuerzo, tiempo y recursos requeridos para desarrollar dicha aplicación?

$$\begin{aligned} \text{Esfuerzo: MM} &= 2,4 \times (32)^{1,05} = 91 \text{ Meses/Hombre} \\ \text{Tiempo: TDEV} &= 2,5 \times (91)^{0,38} = 14 \text{ Meses} \end{aligned}$$

$$\text{Nº. medio de empleado: } 91 / 14 = 6,5 \text{ Personas}$$

La mayor limitación del modelo básico es que no incorpora el efecto de los factores, como la experiencia de los recursos, que influyen sobre el costo y en el mantenimiento del producto.

Modelo Intermedio

El modelo intermedio incorpora 15 variables de predicción que influyen en el costo del proyecto. Estas variables se agrupan en cuatro categorías: atributos del producto software, atributos del ordenador, atributos de personal y atributos del proyecto. Vamos a comentar cada uno de ellos.

Atributos del producto software

- **RELY:** Fiabilidad requerida del software. Podemos definir la fiabilidad como la probabilidad de que el software realice sus funciones satisfactoriamente en su próxima ejecución durante un período dado de tiempo. La influencia se clasifica en: muy alto, alto, nominal, bajo y muy bajo, en función del efecto que tenga un fallo del producto. Un rango muy bajo se usa cuando el defecto tenga que ser eliminado por los desarrolladores pero sin ninguna otra consecuencia; cuando haya posibles pérdidas de vidas humanas indicaremos un rango muy alto.

- **DATA:** Tamaño de la base de datos. Señala el tamaño y complejidad de la base de datos. Se expresa mediante la proporción:

$$[\text{Tamaño de la base de datos en caracteres}] / [\text{Tamaño del programa en DSI}]$$

Este parámetro puede tomar los valores bajo, nominal, alto y, muy alto, en función de cuatro segmentos determinados por los ratios: 10- 100- 1.000.

- **CPLX:** Complejidad del producto. Mide la complejidad en función de las funciones de control, cálculos, gestión de datos y operaciones dependientes de dispositivos. El rango de este parámetro puede variar desde muy bajo si el módulo utiliza expresiones matemáticas simples a extra alto si se emplean varios módulos con ejecución dinámica.

Atributos del ordenador

- **TIME:** Limitaciones en el tiempo de ejecución. Se refiere a las limitaciones de uso de máquina del producto considerado. Se expresa por la relación entre el porcentaje del tiempo de ejecución que se espera utilice el producto y cualquier otros producto/s compitiendo con él. Es nominal cuando su porcentaje es del 50 por 100 y extra alto cuando la restricción es del 95 por 100.

- **STOR:** Limitaciones de memoria principal. Se expresa en términos del porcentaje de memoria que se espera use el producto con relación al utilizado por otros productos. El rango varía desde nominal si se espera una reducción de memoria de menos del 50 por 100 hasta extra alto si la reducción es del 95 por 100.

- **VIRT:** Volatilidad de la máquina virtual. Se entiende por máquina virtual el conjunto de hardware y software que el producto utiliza para realizar su tarea. Durante el desarrollo esta

máquina puede sufrir cambios. El rango de su variabilidad va desde bajo hasta muy alto, en función de estos cambios.

- **TURN:** Frecuencia de cambio en el modelo de explotación del ordenador. Señala el nivel del tiempo de respuesta experimentado por el equipo que desarrolla el proyecto. Se define por el tiempo medio de respuesta en horas desde que el desarrollador introduce un trabajo en el ordenador hasta que obtiene los resultados del proceso. Estos factores han perdido parte de su importancia en los entornos actuales de desarrollo y explotación ya que estas limitaciones se producen sólo en el desarrollo de productos en que no es posible utilizar herramientas de productividad o desarrollos en entornos batch. El rango varía desde bajo para un sistema interactivo hasta muy alto cuando el tiempo de respuesta es mayor de 12 horas.

Atributos de personal

- **ACAP:** Capacitación de los analistas. Expresa en términos de percentiles con relación al conjunto de analistas los siguientes atributos:

- Habilidad para el análisis.
- Eficiencia y calidad en el trabajo.
- Habilidad para comunicarse y cooperar.

Se evalúa su eficiencia trabajando en equipo. Cuanto más capaz sea el equipo de analistas, menor será el esfuerzo necesario. El rango de este parámetro puede variar entre muy bajo y muy alto.

- **AEXP:** Experiencia en aplicaciones. Indica el nivel de experiencia en aplicaciones del equipo de desarrollo de proyectos. El rango varía desde muy bajo (menos de cuatro meses de experiencia) y muy alto (más de 12 años).

- **PCAP:** Capacitación de los programadores. Expresa similares atributos que el parámetro ACAP pero para los programadores.

- **VEXP:** Experiencia en la máquina virtual. Es el tiempo de experiencia en el entorno hardware y software del equipo que desarrolla el software. No se considera el lenguaje de programación. Este parámetro puede tomar los valores desde muy bajo (si la experiencia en la máquina es menor de un mes) hasta alto (si es mayor de tres años).

- **LEXP:** Experiencia en el lenguaje de programación. Un equipo de programadores con amplia experiencia en un lenguaje determinado programará de una forma más segura, disminuyendo incluso el número de errores. El rango de este parámetro puede variar desde muy bajo hasta alto para un equipo con un mes hasta tres años de experiencia.

Atributos del proyecto

- **MODP:** Prácticas Modernas de programación. Señala el grado de utilización de prácticas modernas de programación entendiendo por tal:

- Análisis de requisitos y diseño top-down.
- Diseño estructurado.
- Desarrollo incremental.
- Revisiones o inspecciones de diseño y código.
- Programación estructurada.
- Librerías de programas.

Se valorará el grado de utilización de estas prácticas desde muy bajo hasta muy alto.

• **TOOL:** Uso de herramientas para el desarrollo de software. Señala el grado de utilización de herramientas en el desarrollo al software. Se identifican cinco niveles de herramientas:

- Herramientas básicas de microprocesador.
- Herramientas básicas de microcomputador.
- Herramientas potentes de microcomputador.
- Herramientas potentes de ordenador central.
- Herramientas avanzadas.

El rango de este parámetro varía entre muy bajo, cuando sólo se usan herramientas básicas, hasta muy alto cuando se usan herramientas de propósito especial como CASE (Computer Aided Software Engineering).

• **SCED:** Limitaciones en la planificación. Se define mediante el porcentaje de retraso o aceleración con respecto a la planificación nominal impuesta al equipo de desarrollo. Cualquier aceleración (muy bajo) o retraso (muy, alto) requerirá mayor esfuerzo.

Cálculo de la estimación con el modelo intermedio

Estas 15 variables van a influir sobre la estimación de esfuerzo calculada. El esfuerzo calculado se ajusta multiplicándolo por el resultado de multiplicar entre sí los valores obtenidos de las tablas de atributos en función de los valores identificados en la definición del proyecto. La tabla 18 muestra los multiplicadores de esfuerzos, donde la primera columna muestra las variables, y las restantes el multiplicador a considerar para cada rango de valores desde muy bajo hasta extra alto.

VARIABLE	Muy bajo	Bajo	Valores Nominal	Alto	Muy alto	Extra alto
RELY	0.75	0.88	1.0	1.15	1.40	
DATA		0.94	1.0	1.08	1.16	
CPLX	0.70	0.85	1.0	1.15	1.30	1.65
TIME			1.0	1.11	1.30	1.65
STOR			1.0	1.06	1.21	1.56
VIRT		0.87	1.0	1.15	1.30	
TURN		0.87	1.0	1.07	1.15	
ACAP	1.46	1.19	1.0	0.86	0.71	
AEXP	1.29	1.13	1.0	0.91	0.82	
PCAP	1.42	1.17	1.0	0.86	0.70	
VEXP	1.21	1.10	1.0	0.90		
LEXP	1.14	1.07	1.0	0.95		
MODP	1.24	1.10	1.0	0.91	0.82	
TOOL	1.24	1.10	1.0	0.91	0.83	
SCED	1.23	1.08	1.0	1.04	1.10	

Tabla 18. Multiplicadores de esfuerzo

La estimación de esfuerzo aplicando este modelo es:

Modo orgánico: $MM = 3,2 (KDSI)^{1,05}$

Modo semilibre: $MM = 3,0 (KDSI)^{1,12}$

Modo rígido: $MM = 2,8 (KDSI)^{1,20}$

El tiempo de desarrollo TDEV se calcula como en el modelo básico.

Modelo Detallado

El modelo intermedio tiene dos limitaciones que pueden ser significativas en la estimación detallada de costo en grandes proyectos de software:

- La distribución de esfuerzo por fases puede ser inadecuada.
- Puede ser muy engorroso utilizarlo en un producto con muchos componentes.

El modelo detallado presenta dos funcionalidades que resuelven las limitaciones de COCOMO intermedio:

Multiplicadores de esfuerzo por fases

En el modelo COCOMO intermedio, la distribución de esfuerzo por fase se determina únicamente por el tamaño del producto. En la práctica, factores como la fiabilidad requerida, la experiencia en aplicaciones y desarrollos interactivos afectan a unas fases más que a otras. El modelo detallado proporciona un conjunto de multiplicadores de esfuerzo para cada atributo en cada fase. Estos multiplicadores determinan el esfuerzo requerido para completar cada fase.

Descomposición Jerárquica del producto a tres niveles

En el COCOMO intermedio, los factores de ajuste del costo se calculaban para distintos componentes del producto. Este proceso puede ser muy tedioso e innecesariamente repetitivo si ciertos componentes son agrupados en subsistemas con prácticamente el mismo factor de ajuste. El COCOMO detallado evita este problema proporcionando una jerarquización del producto a tres niveles:

- El nivel **módulo** se describe por el número de instrucciones (DSI) producidas y por aquellos factores que tienden a modificar dicho nivel: complejidad del módulo y adaptación a partir del software existente y de la capacidad y experiencia de los programadores que desarrollarán el módulo en el lenguaje y en la máquina virtual.

- El nivel **subsistema** queda descrito por los restantes factores (limitaciones en tiempo y memoria, capacidad de los analistas, herramientas, planificación, etc.) que tienden a variar de un subsistema a otro, pero que son iguales para todos los módulos dentro de un subsistema.

- El nivel **sistema** se define mediante los factores correspondientes al conjunto del proyecto, como son el esfuerzo nominal y la planificación de tiempos.

No se desarrolla este modelo dado que su aplicación queda reservada a grandes proyectos no muy frecuentes. El desarrollo completo de este modelo puede estudiarse en el libro «Software Engineering Economics» de B. Böehm, editorial Prentice Hall. 1981.

Anexo 6C – Detalles del sistema de Becas de la UTN

Según una entrevista realizada con la directora de la Secretaría de Asuntos Estudiantiles (SAE) de la Facultad Regional Delta, Anabella Segovia, se pudo obtener la información de cómo se gestiona la asignación de becas de la UTN.

A nivel nacional, existe el Programa de Becas de Investigación y Servicio de la UTN. A través de dicho programa, el rectorado de la UTN asigna módulos a las diferentes facultades regionales y de esta manera cada facultad puede disponer de dichos módulos para administrarlo entre los alumnos.

Cada módulo requiere una dedicación por parte del beneficiado, de 6 horas semanales, las cuales deben ser desarrolladas dentro del establecimiento y en beneficio de la institución. Los módulos pueden asignarse a tareas dentro de los diferentes laboratorios de investigación que funcionan en todas las facultades de la UTN del país o pueden ser asignados a servicios que cada facultad necesite (estos servicios pueden ser administrativos o técnicos).

Además, cada beneficiario del programa recibirá la suma de \$ 1040.⁵ anuales, pagaderos en cuotas de \$ 130.- mensuales desde la asignación de la beca (Abril), hasta la finalización de las mismas (Noviembre).

Para hacerse adjudicatario del programa, cada aspirante debe ser alumno regular del establecimiento y debe llenar una solicitud de beca (al comienzo del año lectivo) donde deberá contestar varias preguntas, entre ellas cual es la beca de su interés (puede seleccionar hasta 3 opciones).

En base a esos datos y mediante una comisión de becas creada exclusivamente para la ocasión se realiza una serie de cálculos para evaluar los alumnos aspirantes.

⁵ Pesos Argentinos. Valuación para el ciclo lectivo 2009.

COMISIÓN DE BECAS

ARTICULO 8 - La Comisión de Becas se integrará del siguiente modo: el Secretario de Asuntos Estudiantiles quien presidirá las actividades de la comisión, garantizado su efectividad, dos estudiantes, dos docentes, un graduado y un no docente, todos designados por el Consejo Directivo. Esta comisión podrá ser asesorada por el personal idóneo que se considere necesario.

ARTICULO 9 - Son funciones de la Comisión de Becas:

- a) Coordinar el llamado y la realización del concurso de becas.
- b) Informar a los postulantes los requisitos que deberán cumplir para acceder a la beca.
- c) Determinar la fecha y forma de presentación de las solicitudes de los postulantes.
- d) Analizar las solicitudes presentadas.
- e) Efectuar un relevamiento por Departamentos Académicos. Grupos de investigación y áreas de la Facultad Regional o Regional Académica con el objeto de informarse sobre las necesidades de becas de cada sector.
- f) Establecer el orden de mérito de los postulantes.
- g) Realizar el seguimiento sobre el desempeño de los becarios.
- h) Proponer el reemplazo de los becarios que no cumplan con las obligaciones establecidas.
- i) Adecuar la presente reglamentación a las características propias de la Facultad Regional o Regional Académica.
- j) Contemplar las pautas globales que por Circular determine el Rectorado en los programas de Becas Anuales

ARTICULO 16 - El dictamen deberá ser explícito y fundado. El acta, que firmaran sus integrantes deberá contener:

- a) Nómina de aspirantes.
- b) Análisis y evaluación de todos los elementos de juicio considerados expresados a través de la correspondiente rutina de cálculo.
- c) Orden de mérito de los postulantes.

Reglamento de Becas UTN

Luego, se envían los informes de los aspirantes que obtuvieron mayor puntaje al responsable de cada área donde se desarrollarán las tareas y se invita al aspirante a una reunión con dicho responsable quien lo evaluará y dará un puntaje por cada candidato.

Nuevamente en la SAE, se recalculan los puntajes y se seleccionan los nuevos beneficiarios.

La FRD cuenta con 96 módulos, los cuales han sido modificados a un módulo y medio (1,5 modulo) por cada beca. De esta manera los beneficiarios reciben el valor de \$ 195.- mensual desde Abril hasta Noviembre. Esta modificación es local y sólo se exige el cumplimiento horario de un módulo (6 horas).

De los 64 módulos disponibles para la FRD, 11 son asignados al departamento de Sistemas, a cargo del Ingeniero Luis Perna. Luego de varias entrevistas y de acuerdo al plan previsto para la asignación de becas para el año 2009, se llegó a un acuerdo por el cual 3 becas, que son asignadas para desarrollo de software a cargo del Licenciado Pedro Assis, serán asignadas al desarrollo de SimuReD.

Anexo 7A - Tipos de proyectos que financia el FONTAR

AGENCIA



Agencia Nacional de Promoción
Científica y Tecnológica

Fondo Tecnológico Argentino
(FONTAR)



Desarrollo Tecnológico

Proyectos dirigidos a elevar el nivel tecnológico de una empresa. Se financian a través de créditos, incentivos fiscales y subsidios, que pueden incluir propuestas para:

- Desarrollo de nuevos productos, dispositivos, materiales, procesos o servicios.
- Construcción de prototipos.
- Realización de ensayos a escala piloto.

Las actividades de desarrollo financiables pueden ser ejecutadas por las propias empresas o bien estar a cargo de centros universitarios, institutos del sistema público de ciencia y tecnología o instituciones privadas. Los beneficiarios deben comprometerse a aportar recursos de contrapartida y asumir la responsabilidad de reembolso de los recursos.

Podrá acceder al financiamiento cualquier empresa productiva del país, con independencia de su tamaño y sector de actividad, que cuente con un proyecto correctamente formulado, capacidad técnica y administrativa para ejecutarlo, y aptitud comercial para colocar el producto resultante en el mercado.

Modernización Tecnológica

Proyectos destinados a potenciar la competitividad de las empresas a través de:

- Modificación o mejoras de tecnologías de productos o procesos actualmente en uso.
- Construcción de plantas piloto, desarrollo y producción de prototipos de productos y de series de productos.
- Introducción de tecnologías de gestión de la producción.
- Desarrollos tecnológicos necesarios para pasar de la etapa piloto a la etapa industrial.
- Certificación de calidad.

Podrá acceder al financiamiento cualquier empresa productiva del país, con independencia de su tamaño y sector de actividad, que cuente con un proyecto correctamente formulado, capacidad técnica y administrativa para ejecutarlo, y aptitud comercial para colocar el producto resultante en el mercado.

Gastos de Patentamiento

La Agencia responde a las demandas crecientes del sector industrial, en cuanto a la apropiación de los conocimientos y los resultados producto de las investigaciones y desarrollos realizados en las empresas PyMES.

Por ello, mediante el instrumento ANR PATENTES, se ha creado una herramienta de financiamiento destinada a las PyMEs. Este instrumento tiene por objetivo la presentación de solicitudes de patentes (solicitudes nacionales y/o solicitudes en el extranjero) para promover la protección de los resultados innovadores tanto sea de producto, de procedimientos, como de los desarrollos científicos susceptibles de ser protegidos en diferentes áreas (según cada convocatoria).

Servicios Tecnológicos para Instituciones

Proyectos destinados a ofrecer un respaldo a instituciones cuyos laboratorios y centros de investigación necesiten fortalecer su aptitud para prestar servicios técnicos al sector privado. Se contemplan en estos proyectos gastos de infraestructura, equipamiento y capacitación.

Los créditos promueven la mejora o instalación de servicios altamente calificados. Las instituciones beneficiarias deberán contar con una organización técnica y administrativa que garantice la adecuada utilización de los recursos adjudicados. Las entidades interesadas podrán presentarse individualmente o en forma asociada.

Podrán ser beneficiarias Instituciones públicas o privadas del Sistema Científico-Tecnológico, Universidades Nacionales y Cámaras gremiales empresarias.

Servicios Tecnológicos para Pymes

Proyectos cuyo objetivo es financiar servicios tecnológicos demandados por las empresas y llevados a cabo en Instituciones Científico Tecnológicas o Universidades.

Algunos ejemplos de servicios financiables son: desarrollos tecnológicos específicos, metrología, certificaciones, análisis específicos, etc.

Capacitación

Proyectos que tienen por objetivo la capacitación y el reentrenamiento del personal de la empresa en nuevas tecnologías de producción o de gestión.

Estas actividades pueden tener una vinculación directa con la demanda de recursos humanos capacitados, que surja de un proyecto de desarrollo tecnológico o modernización tecnológica. También pueden responder a un propósito más general, orientado al desarrollo de capacidades en sectores industriales o tecnológicos.

Asistencia Técnica

Los proyectos de asistencia técnica tienen como fin apoyar a las empresas mediante la asistencia de consultores especializados.

El objetivo que se persigue es lograr la adaptación a nuevos niveles tecnológicos y la ejecución de proyectos de I+D; como así también generar desarrollo tecnológico, modernización o capacitación. Pueden comprender también la asistencia técnica para el diseño e implementación de tales proyectos y el desarrollo de planes de negocios originados en I+D.

Programa de Consejerías Tecnológicas

Este Programa tiene como objetivo establecer un mecanismo de fortalecimiento tecnológico para las pequeñas y medianas empresas (PyMES).

El Programa propone, a través de la creación de intermediarios denominados Consejeros Tecnológicos, estimular el desarrollo tecnológico de las empresas PyMES y favorecer un mercado adecuado para dinamizar de forma permanente el proceso de mejoramiento de las capacidades de innovación. La Consejería puede estar destinada a:

- Diagnosticar problemas tecnológicos y de gestión en materia de procesos y productos, organización de la producción y necesidades de capacitación.
- Formular proyectos de innovación, desarrollo y modernización tecnológica, sistemas de calidad y de gestión tecnológica.
- Identificar firmas de ingeniería, organismos tecnológicos y proveedores de servicios técnicos que contribuyan a solucionar los problemas y necesidades de las empresas.
- Desarrollar en la empresa una mayor capacidad de autodiagnóstico e información tecnológica que permita mejorar su competitividad.
- Apoyar en la implementación, seguimiento y evaluación de las mejoras, innovaciones y tareas de innovación.
- Apoyar la creación de estrategias asociativas de desarrollo productivo que provoquen cambios en las empresas participantes, tanto internos (productividad, gestión empresarial, etc.), como también externos (competitividad, alianzas estratégicas, encadenamientos productivos, etc.)
- Apoyar y mejorar la gestión empresarial mediante la incorporación de tecnologías superadoras tales como la aplicación de sistemas informáticos y electrónicos para la organización de la producción, la administración y la comercialización; la modernización de sistemas de logística; la gestión financiera y flujos de fondos; sistemas de información gerencial y otros similares.

Incubadoras de empresas, parques y polos tecnológicos

Proyectos que tienen como objetivo la planificación, implantación, desarrollo y mejoramiento de Incubadoras de Empresas de Base Tecnológica, Parques y/o Polos Tecnológicos.

Anexo 9A – Ley Nacional de Promoción de la Industria del Software

Ley 25.922 - LEY DE PROMOCION DE LA INDUSTRIA DEL SOFTWARE - Definición, ámbito de aplicación y alcances.

Tratamiento fiscal para el sector. Importaciones.

Fondo Fiduciario de Promoción de la Industria del Software (FONSOFT). Infracciones y sanciones.

Disposiciones generales

Sancionada: 18/08/2004

Promulgada Parcialmente: 07/09/2004

Publicación en B.O.: 09/09/2004

El Senado y Cámara de Diputados de la Nación Argentina reunidos en Congreso, etc. Sancionan con fuerza de Ley:

LEY DE PROMOCION DE LA INDUSTRIA DEL SOFTWARE

CAPITULO I Definición, ámbito de aplicación y alcances

ARTICULO 1° - Créase un Régimen de Promoción de la Industria del Software que regirá en todo el territorio de la República Argentina con los alcances y limitaciones establecidas en la presente ley y las normas reglamentarias que en su consecuencia dicte el Poder Ejecutivo nacional. El presente régimen estará enmarcado en las políticas estratégicas que a tal efecto establezca el Poder Ejecutivo nacional a través de sus organismos competentes y tendrá vigencia durante el plazo de diez años a partir de su aprobación.

ARTICULO 2° - Podrán acogerse al presente régimen de promoción las personas físicas y jurídicas constituidas en la República Argentina cuya actividad principal sea la industria del software, que se encuentren habilitadas para actuar dentro de su territorio con ajuste a sus leyes, debidamente inscritas conforme a las mismas y desarrollen en el país y por cuenta propia las actividades definidas en el artículo 4°.

ARTICULO 3° - Los interesados en acogerse al presente régimen deberán inscribirse en el registro habilitado por la autoridad de aplicación. Facúltase a la autoridad de aplicación a celebrar los respectivos convenios con las provincias que adhieran al presente régimen, con el objeto de facilitar y garantizar la inscripción de los interesados de cada jurisdicción provincial en el registro habilitado en el párrafo anterior.

ARTICULO 4° - Las actividades comprendidas en el régimen establecido por la ley son la creación, diseño, desarrollo, producción e implementación y puesta a punto de los sistemas de software desarrollados y su documentación técnica asociada, tanto en su aspecto básico como aplicativo, incluyendo el que se elabore para ser incorporado a procesadores utilizados en bienes de diversa índole, tales como consolas, centrales telefónicas, telefonía celular, máquinas y otros dispositivos. Queda excluida del régimen establecido en la presente ley la actividad de autodesarrollo de software.

ARTICULO 5° - A los fines de la presente ley, se define el software como la expresión organizada de un conjunto de órdenes o instrucciones en cualquier lenguaje de alto nivel, de nivel intermedio, de ensamblaje o de máquina, organizadas en estructuras de diversas secuencias y combinaciones, almacenadas en medio magnético, óptico, eléctrico, discos, chips, circuitos o cualquier otro que resulte apropiado o que se desarrolle en el futuro, previsto para que una computadora o cualquier máquina con capacidad de procesamiento de información ejecute una función específica, disponiendo o no de datos, directa o indirectamente.

CAPITULO II Tratamiento fiscal para el sector

ARTICULO 6° - A los sujetos que desarrollen las actividades comprendidas en el presente régimen de acuerdo a las disposiciones del capítulo I les será aplicable el régimen tributario general con las modificaciones que se establecen en el presente capítulo. Los beneficiarios que adhieran al presente régimen deberán estar en curso normal de cumplimiento de sus obligaciones impositivas y previsionales.

ARTICULO 7° - Los sujetos que adhieran a este régimen gozarán de estabilidad fiscal por el término de diez (10) años contados a partir del momento de la entrada en vigencia de la presente ley. La estabilidad fiscal alcanza a todos los tributos nacionales, entendiéndose por tales los impuestos directos, tasas y contribuciones impositivas que tengan como sujetos pasivos a los beneficiarios inscriptos. La estabilidad fiscal significa que los sujetos que desarrollen actividades de producción de software no podrán ver incrementada su carga tributaria total nacional al momento de la incorporación de la empresa al presente marco normativo general.

ARTICULO 8° - Los beneficiarios del régimen de la presente ley que desempeñen actividades de investigación y desarrollo en software y/o procesos de certificación de calidad de software desarrollado en el territorio nacional y/o exportaciones de software (asegurando a los trabajadores de la actividad la legislación laboral vigente), podrán convertir en un bono de crédito fiscal intransferible hasta el 70% (setenta por ciento) de las contribuciones patronales que hayan efectivamente pagado sobre la nómina salarial total de la empresa con destino a los sistemas y subsistemas de seguridad social previstos en las leyes 19.032 (INSSJyP), 24.013 (Fondo Nacional de Empleo) y 24.241 (Sistema Integrado de Jubilaciones y Pensiones). Los beneficiarios podrán utilizar dichos bonos para la cancelación de tributos nacionales que tengan origen en la industria del software, en particular el impuesto al valor agregado (IVA) u otros impuestos nacionales y sus anticipos, en caso de proceder, excluido el impuesto a las ganancias. El bono no podrá utilizarse para cancelar deudas anteriores a la efectiva incorporación del beneficiario al régimen de la presente ley y, en ningún caso, eventuales saldos a su favor harán lugar a reintegros o devoluciones por parte del Estado.

ARTICULO 9° - Los sujetos adheridos al régimen de promoción establecido por la presente ley tendrán una desgravación del sesenta por ciento (60%) en el monto total del impuesto a las ganancias determinado en cada ejercicio. Este beneficio alcanzará a quienes acrediten gastos de investigación y desarrollo y/o procesos de certificación de calidad y/ o exportaciones de software, en las magnitudes que determine la autoridad de aplicación.

ARTICULO 10. - A los efectos de la percepción de los beneficios establecidos en los artículos precedentes, los sujetos que adhieran al presente régimen deberán cumplir con alguna norma de calidad reconocida aplicable a los productos de software. Esta exigencia comenzará a regir a partir del tercer año de vigencia del presente marco promocional.

ARTICULO 11. - Los sujetos que adhieran a los beneficios establecidos en la presente ley, que además de la industria del software como actividad principal desarrollen otras de distinta naturaleza, llevarán su contabilidad de manera tal que permita la determinación y evaluación en forma separada de la actividad promovida del resto de las desarrolladas. La imputación de gastos compartidos con actividades ajenas a las promovidas se atribuirán contablemente respetando criterios objetivos de reparto, como cantidad de personal empleado, monto de salarios pagados, espacio físico asignado u otros, siendo esta enumeración meramente enunciativa y no limitativa. Serán declarados y presentados anualmente a la autoridad de aplicación en la forma y tiempo que ésta establezca los porcentuales de apropiación de gastos entre las actividades distintas y su justificativo.

CAPITULO III Importaciones

ARTICULO 12. - Las importaciones de productos informáticos que realicen los sujetos que adhieran al presente régimen de promoción quedan excluidas de cualquier tipo de restricción presente o futura para el giro de divisas que se correspondan al pago de importaciones de hardware y demás componentes de uso informático que sean necesarios para las actividades de producción de software.

CAPITULO IV Fondo Fiduciario de Promoción de la Industria del Software (Fonsoft)

ARTICULO 13. - Créase el Fondo Fiduciario de Promoción de la Industria del Software (Fonsoft), el cual será integrado por:

1. Los recursos que anualmente se asignen a través de la ley de presupuesto.
2. Los ingresos por las penalidades previstas ante el incumplimiento de la presente ley.
3. Ingresos por legados o donaciones.
4. Fondos provistos por organismos internacionales u organizaciones no gubernamentales.

ARTICULO 14. - Facúltase al Jefe de Gabinete de Ministros a efectuar las modificaciones presupuestarias que correspondan, previendo para el primer año un monto de pesos dos millones (\$ 2.000.000) a fin de poder cumplir con lo previsto en el inciso 1 del artículo 13.

ARTICULO 15. - La Secretaría de Ciencia, Tecnología e Innovación Productiva, a través de la Agencia Nacional de Promoción Científica y Tecnológica, será la autoridad de aplicación en lo referido al Fonsoft y actuará como fiduciante frente al administrador fiduciario.

ARTICULO 16. - La autoridad de aplicación definirá los criterios de distribución de los fondos acreditados en el Fonsoft los que serán asignados prioritariamente a universidades, centros de investigación, pymes y nuevos emprendimientos que se dediquen a la actividad de desarrollo de software. A los efectos mencionados en el párrafo anterior la autoridad de aplicación convendrá con las provincias que adhieran al régimen de la presente

ley, la forma y modo en que éstas, a través de sus organismos pertinentes, se verán representadas en la Agencia Nacional de Promoción Científica y Tecnológica.

ARTICULO 17. - La autoridad de aplicación podrá financiar a través del Fonsoft:

1. Proyectos de investigación y desarrollo relacionados a las actividades definidas en el artículo 4° de la presente.
2. Programas de nivel terciario o superior para la capacitación de recursos humanos.
3. Programas para la mejora en la calidad de los procesos de creación, diseño, desarrollo y producción de software.
4. Programas de asistencia para la constitución de nuevos emprendimientos.

ARTICULO 18. - La autoridad de aplicación otorgará preferencia en la asignación de financiamientos a través del Fonsoft, según lo definido en el artículo 16, a quienes:

- a) Se encuentren radicados en regiones del país con menor desarrollo relativo
- b) Registren en la República Argentina los derechos de reproducción de software según las normas vigentes;
- c) Generen mediante los programas promocionados un aumento cierto y fehaciente en la utilización de recursos humanos;
- d) Generen mediante los programas promocionados incrementales de exportación;
- e) Adhieran al presente régimen de promoción.

ARTICULO 19. - Las erogaciones de la autoridad de aplicación relacionadas a la administración del Fonsoft no deberán superar el cinco por ciento (5%) de la recaudación anual del mismo.

CAPITULO V Infracciones y sanciones

ARTICULO 20. - El incumplimiento de las normas de la presente ley y de las disposiciones de la autoridad de aplicación referidas a los beneficios establecidos en el capítulo II por parte de las personas físicas y jurídicas que se acojan al régimen de promoción de la presente ley, determinará la aplicación por parte de la autoridad de aplicación de las sanciones que se detallan a continuación:

1. Revocación de la inscripción en el registro establecido en el artículo 3° y de los beneficios otorgados por el capítulo II.
2. Pago de los tributos no ingresados con motivo de lo dispuesto en el capítulo II, con más los intereses, en relación con el incumplimiento específico determinado.
3. Inhabilitación para inscribirse nuevamente en el registro establecido en el artículo 3°.

CAPITULO VI Disposiciones generales

ARTICULO 21. - La autoridad de aplicación de la presente ley será la Secretaría de Industria, Comercio y de la Pequeña y Mediana Empresa dependiente del Ministerio de Economía y Producción, con excepción de lo establecido en el capítulo IV y sin perjuicio de lo establecido por el artículo 6° del decreto 252/2000, según texto ordenado por el decreto 243/2001.

ARTICULO 22. - La Secretaría de Industria, Comercio y de la Pequeña y Mediana Empresa deberá publicar en su respectiva página de Internet el registro de los beneficiarios del presente régimen, así como los montos de beneficio fiscal otorgados a los mismos.

ARTICULO 23. - A los fines de la presente ley quedan excluidas como actividades de investigación y desarrollo de software la solución de problemas técnicos que se hayan superado en proyectos anteriores sobre los mismos sistemas operativos y arquitecturas informáticas. También el mantenimiento, la conversión y/o traducción de lenguajes informáticos, la adición de funciones y/o preparación de documentación para el usuario, garantía o asesoramiento de calidad de los sistemas no repetibles existentes. Quedan también excluidas las actividades de recolección rutinarias de datos, la elaboración de estudios de mercado para la comercialización de software y aquellas otras actividades ligadas a la producción de software que no conlleven un progreso funcional o tecnológico en el área del software.

ARTICULO 24. - La autoridad de aplicación realizará auditorías y evaluaciones del presente régimen, debiendo informar anualmente al Congreso de la Nación los resultados de las mismas. Dicha información deberá realizarse a partir del tercer año de vigencia de la ley.

ARTICULO 25. - Los beneficios fiscales contemplados en la presente ley, mientras subsista el sistema de coparticipación federal de impuestos vigente, se detraerán de las cuantías de los recursos que correspondan a la Nación.

ARTICULO 26. - El cupo fiscal de los beneficios a otorgarse por el presente régimen promocional será fijado anualmente en la ley de Presupuesto general de gastos y cálculo de recursos de la Administración nacional. A partir de la vigencia de la presente ley y durante los tres primeros ejercicios fiscales posteriores, el cupo correspondiente se otorgará en función de la demanda y desarrollo de las actividades promocionadas.

ARTICULO 27. - Invítase a las provincias, a la Ciudad Autónoma de Buenos Aires y a los municipios a adherir al presente régimen mediante el dictado de normas de promoción análogas a las establecidas en la presente ley.

ARTICULO 28. - Comuníquese al Poder Ejecutivo.

DADA EN LA SALA DE SESIONES DEL CONGRESO ARGENTINO, EN BUENOS AIRES, A LOS DIECIOCHO DIAS DEL MES DE AGOSTO DEL AÑO DOS MIL CUATRO.

- REGISTRADA BAJO EL N° 25.922 - EDUARDO O. CAMAÑO. - MARCELO A. GUINLE. - Eduardo D. Rollano. - Juan Estrada.

Decreto 1182/2004 Bs. As., 7/9/2004 VISTO el Expediente N° S01:02 02385/2004 del Registro del MINISTERIO DE ECONOMIA Y PRODUCCION y el Proyecto de Ley registrado bajo el N° 25.922, sancionado por el HONORABLE CONGRESO DE LA NACION el 18 de agosto de 2004, y CONSIDERANDO:

Que mediante el Proyecto de Ley registrado bajo el N° 25.922, se creó un Régimen de Promoción de la Industria del Software que regirá en todo el territorio de la REPUBLICA ARGENTINA, previendo una serie de estímulos de carácter impositivo.

Que los referidos estímulos comprenden al beneficio de la estabilidad fiscal por el término de DIEZ (10) años con alcance a tributos nacionales, a bonos de crédito fiscal por hasta un SETENTA POR CIENTO (70%) de las contribuciones patronales efectivamente pagadas, para ser aplicados a la cancelación de tributos nacionales, y a una desgravación del SESENTA POR CIENTO (60%) en el monto total del Impuesto a las Ganancias determinado en cada ejercicio.

Que a través de su Artículo 25, el Proyecto de Ley establece que los beneficios fiscales que contempla, mientras subsista el sistema de coparticipación federal de impuestos vigente, se detraerán de las cuantías de los recursos que correspondan a la Nación.

Que la distribución de los recursos entre la Nación, las Provincias y la CIUDAD AUTONOMA DE BUENOS AIRES, debe responder, según lo ordena la CONSTITUCION NACIONAL, a criterios de equidad y solidaridad, debiendo dar prioridad al logro de un grado equivalente de desarrollo, calidad de vida e igualdad de oportunidades en todo el Territorio Nacional.

Que, por lo tanto, no resulta equitativo imponer una detracción a los recursos de la Nación para sostener un régimen al cual podrán tener acceso, en virtud de su carácter federal, todas las jurisdicciones políticas.

Que la incidencia de la medida aprobada en los recursos de la Nación dificultará, en la medida de la merma, el cumplimiento por parte del ESTADO NACIONAL de sus obligaciones específicas.

Que en función de los argumentos expuestos se estima conveniente observar el Artículo 25 del Proyecto de Ley registrado bajo el N° 25.922.

Que la medida que se propone no altera el espíritu ni la unidad del Proyecto de Ley sancionado por el HONORABLE CONGRESO DE LA NACION.

Que la Dirección General de Asuntos Jurídicos del MINISTERIO DE ECONOMIA Y PRODUCCION ha tomado la intervención que le compete.

Que el PODER EJECUTIVO NACIONAL tiene competencia para el dictado del presente decreto de acuerdo con lo dispuesto por el Artículo 80 de la CONSTITUCION NACIONAL.

Por ello, EL PRESIDENTE DE LA NACION ARGENTINA EN ACUERDO GENERAL DE MINISTROS DECRETA:

Artículo 1° - Obsérvase el Artículo 25 del Proyecto de Ley registrado bajo el N° 25.922.

Artículo 2° - Con la salvedad establecida en el artículo precedente, cúmplase, promúlgase y téngase por Ley de la Nación el Proyecto de Ley registrado bajo el N° 25.922.

Artículo 3° - Dése cuenta al HONORABLE CONGRESO DE LA NACION.

Artículo 4° - Comuníquese, publíquese, dése a la Dirección Nacional del Registro Oficial y archívese.

- KIRCHNER. - Alberto A. Fernández. - Julio M. De Vido. - José J. B. Pampuro. - Alicia M. Kirchner.

- Aníbal D. Fernández. - Roberto Lavagna. - Carlos A. Tomada. - Rafael A. Bielsa. - Horacio D. Rosatti. - Ginés González García

Anexo 9B – El Software Como Actividad Industrial

Ley 25.856: Establécese que la actividad de producción de software debe considerarse como una actividad productiva de transformación asimilable a una actividad industrial, a los efectos de la percepción de beneficios impositivos, crediticios y de cualquier otro tipo.

Sancionada: Diciembre 4 de 2003

Promulgada: Enero 6 de 2004

El Senado y Cámara de Diputados de la Nación Argentina reunidos en Congreso, sancionan con fuerza de Ley:

Consideración de la producción de software como actividad industrial

ARTICULO 1° - Establécese que la actividad de producción de software debe considerarse como una actividad productiva de transformación asimilable a una actividad industrial a los efectos de la percepción de los beneficios impositivos, crediticios y de cualquier otro tipo que se fijen para la industria por parte del Gobierno nacional.

ARTICULO 2° - Se instruye al Poder Ejecutivo para que en las políticas de promoción productiva vigente o a regir en el futuro se considere el diseño, el desarrollo y la elaboración de software como una actividad productiva de transformación pasible de ser promocionada.

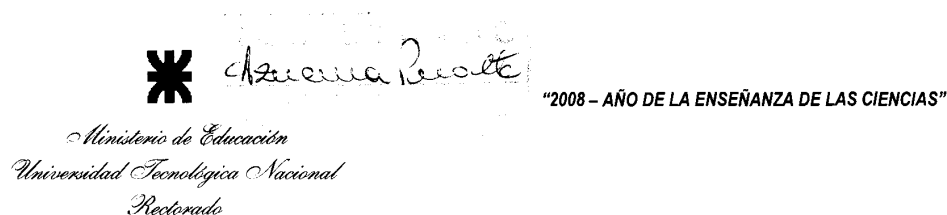
ARTICULO 3° - Se invita a los gobiernos provinciales y municipales a adoptar igual criterio al establecido en los artículos anteriores a los efectos de la extensión de los beneficios que se establezcan para las actividades industriales a las actividades productoras de software.

ARTICULO 4° - Comuníquese al Poder Ejecutivo. DADA EN LA SALA DE SESIONES DEL CONGRESO ARGENTINO, EN BUENOS AIRES, A LOS CUATRO DIAS DEL MES DE DICIEMBRE DEL AÑO DOS MIL TRES.

- REGISTRADA BAJO EL N° 25.856 –

EDUARDO O. CAMAÑO. - DANIEL O. SCIOLI. - Eduardo D. Rollano. - Juan Estrada.

Anexo 9C – Reglamento de Becas



Ordenanza N° 1180

Anexo I

REGLAMENTO DE BECAS

Artículo 1°: La designación de becarios en la Universidad Tecnológica Nacional se regirá por las disposiciones del presente Reglamento de Becas.

Artículo 2°: Las Becas de la Universidad Tecnológica Nacional son las siguientes:

- a) **Becas de Investigación:** para desarrollar proyectos de investigación propuestos por la Secretaría de Ciencia y Tecnología.
- b) **Becas de Servicio:** para desarrollar técnicas no administrativas que cubran necesidades de la Unidad Académica
- c) **Becas de Ayuda Social Económica:** para satisfacer necesidades básicas de los estudiantes, como transporte, apuntes, residencia y todas aquellas que limiten directamente sus posibilidades de estudio.

Cualquier nuevo plan de becas a implementarse en la Universidad deberá encuadrarse dentro de los incisos anteriores.

I. Disposiciones Generales:

Artículo 3°: Las Becas se proveerán por concurso de antecedentes y oposición para los casos que así lo requieran.

Artículo 4°: Podrán presentarse a concurso todos los alumnos de la Universidad Tecnológica Nacional que reúnan los siguientes requisitos:

- Cursar una carrera en la Universidad Tecnológica Nacional
- No poseer título universitario expedido por cualquier Universidad de gestión estatal o privada.

Artículo 5°: En cada Facultad Regional o Regional Académica funcionará una comisión de becas, presidida por el Secretario de Asuntos Estudiantes, a los efectos de poner en práctica lo dispuesto por este reglamento.

Artículo 6°: Los módulos de becas se adjudicarán por Resolución del Decano o Director Ad – Referéndum de su Consejo Directivo de acuerdo al dictamen elevado por la Comisión de Becas, junto a los antecedentes de todos los aspirantes.

Artículo 7°: El tipo, monto, número y distribución del ochenta por ciento de la totalidad de las becas, serán establecidas anualmente por Rectorado a propuesta



Luciana Pico

"2008 – AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo I

de la Secretaría de Asuntos Estudiantiles, por Resolución Ad – Referéndum del Consejo Superior, con fecha anterior a las mismas. El veinte por ciento restante será distribuido por el Consejo Superior en función de las propuestas elevadas por los Consejos Académicos o Directivos y la Secretaría de Asuntos Estudiantiles del Rectorado.

II. Comisión de Becas:

Artículo 8°: La Comisión de Becas se integrará del siguiente modo: el Secretario de Asuntos Estudiantiles quien presidirá las actividades de la comisión, garantizado su efectividad, dos estudiantes, dos docentes, un graduado y un no docente, todos designados por el Consejo Directivo.

Esta comisión podrá ser asesorada por el personal idóneo que se considere necesario.

Artículo 9°: Son funciones de la Comisión de Becas:

- a) Coordinar el llamado y la realización del concurso de becas.
- b) Informar a los postulantes los requisitos que deberán cumplir para acceder a la beca.
- c) Determinar la fecha y forma de presentación de las solicitudes de los postulantes.
- d) Analizar las solicitudes presentadas.
- e) Efectuar un relevamiento por Departamentos Académicos. Grupos de investigación y áreas de la Facultad Regional o Regional Académica con el objeto de informarse sobre las necesidades de becas de cada sector.
- f) Establecer el orden de mérito de los postulantes.
- g) Realizar el seguimiento sobre el desempeño de los becarios.
- h) Proponer el reemplazo de los becarios que no cumplan con las obligaciones establecidas.
- i) Adecuar la presente reglamentación a las características propias de la Facultad Regional o Regional Académica.



Asunción

"2008 – AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado

Ordenanza N° 1180

Anexo I

- j) Contemplar las pautas globales que por Circular determine el Rectorado en los programas de Becas Anuales.

III. Concurso:

Artículo 10°: LLAMADO. El llamado a concurso para la designación de becarios lo implementará el área de Asuntos Estudiantiles en cada Facultad Regional o Regional Académica, declarando abierta la inscripción por el término de diez (10) días y facilitando a los postulantes las solicitudes de inscripción respectivas.

Artículo 11°: PUBLICIDAD. La difusión del llamado a concurso se efectuará mediante carteleras en la dependencia con no menos de diez (10) días de anticipación a la apertura de la inscripción. Se especificará la cantidad de becas a concursar, los grupos de investigación o los servicios que se desea cubrir y las condiciones exigidas a esos efectos.

Artículo 12°: INFORMACION. Durante el plazo de difusión e inscripción el área de Asuntos Estudiantes pondrá a disposición de los interesados, Departamentos y Direcciones, la información con relación a las becas que se concursan.

Artículo 13°: INSCRIPCION. Las solicitudes de inscripción que los postulantes completarán y firmarán, se adjuntan como ANEXO II (Becas de Investigación y Servicio) y ANEXOS III y IV (Becas de Ayuda Económica), serán facilitadas por el área de Asuntos Estudiantiles de cada dependencia a los interesados.

Artículo 14°: COLOQUIO. La Comisión de Becas junto con los jefes de los grupos de investigación o servicios mantendrán un coloquio con los aspirantes a cada grupo específico, con el objeto de completar el perfil del estudiante, incorporando este informe a la documentación respectiva.

IV. Dictamen de la Comisión de Becas:

Artículo 15°: PLAZO. La Comisión de Becas, deberá expedirse dentro de los diez (10) días de haber finalizado la inscripción.

Artículo 16°: CONTENIDO. El dictamen deberá ser explícito y fundado. El acta, que firmaran sus integrantes deberá contener:

- a) Nómina de aspirantes.



Comisión de Becas

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo I

- b) Análisis y evaluación de todos los elementos de juicio considerados expresados a través de la correspondiente rutina de cálculo.
- c) Orden de mérito de los postulantes.

Artículo 17°: En caso de paridad de méritos, una entrevista personal con los aspirantes y la consideración de su situación económica, permitirá a la Comisión de Becas y a los jefes de grupos determinar el mérito de cada uno en particular.

Artículo 18°: El dictamen de la Comisión de Becas será exhibido en cartelera. De existir defectos de forma o procedimiento a juicio de los postulantes, en los tres (3) días sucesivos deberá interponerse y fundarse la apelación respectiva ante el Decano, quien consultará con la Comisión de Becas y al Secretario de Asuntos Estudiantiles.

V. Resolución:

Artículo 19°: Por Resolución del Decano o Director Ad-Referéndum del Consejo Directivo, se establecerá:

- a) Aprobar el dictamen de la Comisión de Becas.
- b) Declarar desierto el concurso, fundamentado tal decisión.
- c) Dejar sin efecto el concurso, acompañando los antecedentes que obligaron a tal determinación.

Artículo 20°: Notificado de su designación en un plazo de cinco (5) días, el becario se hará cargo de sus funciones inmediatamente. Si así no ocurriera y no media causa justificada, la beca será otorgada al segundo postulante en el orden de mérito elevado por la Comisión de Becas.

Artículo 21°: Las adjudicaciones de becas serán informadas a Rectorado para que se arbitren las medidas administrativas pertinentes.

VI. Informes:

Artículo 22°: Trimestralmente, la Comisión de Becas pedirá informes a los jefes de grupos de investigación y servicio sobre el desempeño de los becarios, a los efectos



2074-150
Luciana Piccolo

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo I

de tomar conocimientos sobre las tareas desarrolladas, y a la experiencia adquirida por los estudiantes.

Artículo 23°: La Comisión de Becas mantendrá reuniones periódicas con los alumnos que tengan becas de ayuda económica, informándose sobre el desarrollo de su plan de estudios y su problemática particular.

Artículo 24°: El Secretario de Asuntos Estudiantiles remitirá un informe pormenorizado a la Secretaría de Asuntos Estudiantiles del Rectorado antes de finalizar el período lectivo sobre la marcha del programa de becas, lo que permitirá volcar la experiencia evaluada en nuevas acciones.

VII. Obligaciones de los alumnos becados:

Artículo 25°: Los alumnos que perciben una beca deberán cursar, como mínimo, tres (3) materias durante el año que dura el beneficio y cumplir con las exigencias de trabajo en cada caso determine el jefe de grupo, la Comisión de Becas o quien corresponda, en función del tipo de beca de que se trate. Las únicas excepciones admitidas y que deberá formular el alumno por escrito, corresponderán a:

- a) Haber terminado o estar terminando de cursar las últimas asignaturas de la carrera.
- b) Causas debidas y fehacientemente justificadas a juicio de la Comisión de Becas.

Artículo 26°: Los alumnos becados deberán conocer, aceptar y cumplir las disposiciones de la presente reglamentación y las de carácter interno que dicte la Comisión de Becas.

Artículo 27°: Los alumnos becados deberán informar a la Secretaría de Asuntos Estudiantiles de la Facultad Regional o Regional Académica, por escrito, toda variación de su situación académica o económica que ocurriera durante el período que dure su beca.

Artículo 28°: Será obligación de los alumnos beneficiados con una beca de investigación o de servicio, presentar a la Comisión de Becas un informe escrito anual en las fechas en que la Comisión determine, acerca de las tareas desarrolladas experiencias adquiridas y posibilidades de mejorar los métodos de



Alicia Beato

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo I

trabajo implementados, junto con la certificación del jefe de grupo o área.

Artículo 29°: Los alumnos beneficiados deberán cumplir como máximo una carga horaria de 6 (seis) horas semanales por cada modulo de beca.

VIII. Situaciones derivadas:

Artículo 30°: RENOVACION DE LA BECA. A los efectos de ampliar las posibilidades de acceso a los alumnos a los grupos de investigación y servicios de la Universidad, los grupos que cuenten con más de tres (3) becarios renovaran un tercio de los mismos todos los años.

Los grupos con menos de tres (3) becarios podrán renovar la beca solamente en nuevo período, haciendo los reemplazos en años alternados.

Artículo 31°: BECAS VACANTES. Las becas que quedaran vacantes durante el período de su vigencia deberán ser adjudicadas a los postulantes que sigan en orden de méritos según el dictamen de la Comisión de Becas, excepto que no hubiera postulantes por lo que deberá llamarse nuevamente a concurso.

Artículo 32°: SUSPENSION DE LA BECA. El incumplimiento comprobado de cualquiera de las obligaciones de la presente reglamentación por parte del becario, o el haber falseado la información solicitada para poder acceder a la beca, dará lugar a la suspensión del pago de su beca por un período no mayor a treinta (30) días durante el cual la Comisión de Becas deberá resolver su situación.

Artículo 33°: CESE DE LA BECA. De acuerdo con el análisis efectuado por la Comisión de Becas, ésta determinará el cese de la beca y pasará su dictamen a Resolución del Decano o Director Ad-Referéndum de su Consejo Directivo. Son motivos para el cese de una beca:

- a. La desaparición o sustancial modificación de las causas que justificaron su otorgamiento.
- b. El incumplimiento injustificado de sus obligaciones por parte del becario.
- c. El abandono de sus estudios.
- d. La renuncia voluntaria a la beca.
- e. La graduación del alumno.



RECTORADO
Buenos Aires

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado

Ordenanza N° 1180

Anexo I

- f. No presentarse a cobrar la beca en un plazo de sesenta (60) días y no medie causa justificada.
- g. Toda otra causa que, a juicio de la Comisión de Becas, justifique la cesación.

Con estas becas vacantes, se procederá según el artículo 31.

Artículo 34°: SANCIONES. Cuando se compruebe que un becario obtuvo su beneficio adulterando documentación o faltando a la verdad, se le suspenderá inmediatamente la beca, elevándose los antecedentes a la Secretaría Académica del Rectorado a fin de que en el Consejo Superior se determinen las sanciones a aplicar de acuerdo con el estatuto de la Universidad. El alumno está obligado a devolver actualizados los importes percibidos hasta ese momento.

IX. Incompatibilidades:

Artículo 35°: Es incompatible el desempeño de un cargo rentado en la Universidad Tecnológica Nacional y el beneficio de una beca, con excepción de aquellos estudiantes que sean designados como ayudantes de segunda y presten un servicio de interés para la Facultad Regional o Regional Académica.

X. Becas de Investigación y Servicio

Artículo 36°: Además de las consideraciones generales ya expresadas en el desarrollo del presente reglamento, se seguirá la siguiente rutina de cálculo para la selección de los aspirantes:



Luciana Bualto

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo I

PROMEDIO GENERAL

Más

N° DE MATERIAS APROBADAS x 0.1

Más

N° DE MATERIAS CURSADAS x 3 / TOTAL DE MATERIAS DE LA CARRERA

Más

2

1 + N° APLAZOS

Más

**ANTECEDENTES PERSONALES (situación económica, becas anteriores,
trabajos afines, etc.). De 0 a 5 puntos.**

XI. Becas de Ayuda Económica

Artículo 37°: Este beneficio se otorgará sin necesidad de ninguna contraprestación del beneficiario en la Facultad Regional o Regional Académica, con la única exigencia de cumplir con el plan de desarrollo académico propuesto por el propio alumno con respecto al avance de la carrera.

Artículo 38°: La renovación de la beca podrá efectuarse, siempre que se haya cumplido con los objetivos propuestos en el plan de desarrollo académico.

Artículo 39°: La ponderación de las variables que se tendrán en cuenta para determinar el orden de selección de los becarios de ayuda social, que se adjuntan como Anexo N° V de este reglamento es la siguiente:



Luciana Balle

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo I

DESCRIPCION	PONDERACION
NIVEL DE INGRESOS	30
CONDICION DE ACTIVIDAD LABORAL	5
VIVIENDA	5
SALUD	5
TASA DE DEPENDENCIA	5
PROMEDIO GENERAL (incluyendo aplazos)	35
RENDIMIENTO ACADEMICO	10
PLAN DE DESARROLLO ACADEMICO	5
TOTAL	100

Artículo 40°: Las disposiciones internas a las que se ajustara el funcionamiento del Programa de Becas en cada Facultad Regional o Regional Académica, regulará los aspectos no previstos en el presente reglamento y en particular, establecerá:

- a. Los compromisos que debe cumplir el becario para mantener el beneficio durante el período en que sea otorgado, conforme a las características del sistema de enseñanza de la Facultad Regional o Regional Académica.
- b. La remoción de los integrantes de la Comisión de Becas que no cumplan con responsabilidad la tarea asignada.

Artículo 41°: El becario menor de edad o discapacitado, en el caso que así lo decida, podrá percibir el importe correspondiente al beneficio de la beca a través de su padre, madre o tutor.

Artículo 42°: A los efectos de la aplicación de este Reglamento, los plazos contados serán en días hábiles.

Artículo 43°: Toda situación no prevista en el presente reglamento, será resuelta por la Comisión de Becas.



Facultad Delta

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo II

Solicitud de Beca

DATOS DEL ESTUDIANTE

Apellido:..... Nombres:

DNI N°:..... Nacionalidad:..... Fecha de Nacimiento:/...../.....

Sexo: Estado civil: Teléfono: E-mail:

Domicilio: N°: Piso: Dpto.:

Código Postal: Localidad: Provincia:

Si el solicitante no viviera con su familia, completar el siguiente cuadro:

Domicilio de su familia:

Calle: N°: Piso: Dpto.:

Código Postal: Localidad: Provincia: Teléfono:

DATOS DEL GRUPO FAMILIAR

El solicitante debe incluirse entre los miembros del grupo.

Apellido y Nombres	Edad	Parentesco	Ocupación	Ver (1)	Ingresos (2)
Total de ingresos mensuales					

(1) Indicar en esta columna, marcando con una x, al miembro del grupo familiar que viva en el mismo domicilio.



Azuena Beato

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

Ministerio de Educación
 Universidad Tecnológica Nacional
 Rectorado

Ordenanza N° 1180

Anexo II

(2) Consignar el monto mensual neto (de bolsillo) que percibe cada uno de los miembros del grupo familiar en concepto de salario, ingresos por actividad autónoma, renta percibida por alquileres, cuota alimenticia, jubilaciones, pensiones u otras formas de ingresos regulares.

Observaciones referidas a este ítem.....

VIVIENDA

En referencia a la vivienda que habita el estudiante, indicar (marcando con un círculo) la situación de la misma en relación a alguno de los miembros de su grupo familiar:

Propietario sin deuda hipotecaria	SI / NO
Propietario con deuda hipotecaria	SI / NO
Inquilino u ocupante	SI / NO

Observaciones referidas a este ítem:

CONDICIÓN DE ACTIVIDAD LABORAL

Indicar (marcando con un círculo) la condición de actividad del jefe del grupo familiar:

Ocupado autónomo o dependiente, incluido jubilado o pensionado.	SI / NO
Sub-ocupado (incluye subsidio por desempleo)	SI / NO
Desocupado	SI / NO

Observaciones referidas a este ítem:

SALUD

Indicar (marcando con un círculo) la cobertura de salud que posee.

Si tiene obra social	SI / NO
Cobertura de salud parcial	SI / NO
No tiene obra social	SI / NO



Azuara

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo II

Observaciones referidas a este ítem:

.....
.....

RENDIMIENTO ACADÉMICO

Indique el número total de materias que ha aprobado al día de la fecha, por promoción o examen final, sin contar equivalencias: año en que ingresó a la Universidad:

Cantidad de Aplazos: Promedio General incluyendo aplazos:

DOCUMENTACION A PRESENTAR CON LA SOLICITUD

a) En todos los casos:

Fotocopia de D.N.I.

Fotocopia de Desocupado, (en caso de corresponder) expedido por la ANSES.

Fotocopia de los comprobantes de ingresos mensuales personales y del grupo familiar.

Fotocopia de las últimas facturas por consumo de agua corriente y electricidad, de la vivienda en la que habita.

Fotocopia de boletas pagas o impagas del último bimestre del Impuesto inmobiliario de la vivienda en la que habita.

Si es soltero y tiene hermanos menores de 18 años, fotocopia de la libreta de casamiento de los padres o, en su defecto, partidas de nacimiento de los hermanos.

Si es casado, fotocopia de la libreta de casamiento y/o partida de nacimiento de los hijos.

b) Cuando corresponda:

Fotocopia del contrato de alquiler y fotocopia del último recibo de pago.

Fotocopia del pago mensual del crédito hipotecario.

c) Importante: La información que se detalla en esta solicitud debe ser exacta, supone el conocimiento del Reglamento de Becas de Ayuda Económica y tiene carácter de **DECLARACIÓN JURADA**. La Solicitud deberá ser entregada **ÚNICAMENTE** en la Secretaría de Asuntos Estudiantiles de la Facultad Regional o Regional Académica.

Fecha de recepción:

Firma Secretaría:

Aclaración y sello:

Firma del solicitante:



Asunción Revollo

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

Ministerio de Educación
 Universidad Tecnológica Nacional
 Rectorado

Ordenanza N° 1180

Anexo III

Plan de Desarrollo Académico Propuesto

Apellido y Nombre y del Alumno:.....

Legajo: DNI: Fecha:

Carrera que cursa: F.R. /R.A:

Materia a cursar:

Nombre de la materia	Nivel al que pertenece	Régimen (cuatrimestral o anual)

Finales a rendir:

Nombre de la materia	Nivel al que pertenece	Mes mesa examinadora

Observaciones:

.....

 Por la comisión de becas

 Fecha y Firma del postulante



Universidad Tecnológica Nacional

"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo IV

Planilla de seguimiento

Apellido y Nombre y del Alumno:

Legajo: DNI: Fecha:

Carrera que cursa: F.R. /R.A:.....

Desarrollo de cursadas:

Nombre de la materia	Asistencia (S/N)	Observación	Regularizada S / N

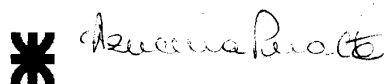
Seguimiento de finales:

Nombre de la materia	Fecha	Observación	Aprobada S / N

Observaciones:

.....
.....
.....

Fecha y Firma
Por la comisión de becas



"2008 - AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo V

Ponderación de variables para la adjudicación de Becas

I). Situación Socioeconómica: Participación en la ponderación: 50

Para la determinación de la situación socioeconómica, se considera el **nivel de ingreso como el gran indicador y a los demás como correctores:**

INDICADOR	DESCRIPCION	PONDERACION
A	NIVEL DE INGRESOS	30
B	VIVIENDA	5
C	CONDICION DE ACTIVIDAD LABORAL	5
D	SALUD	5
E	TASA DE DEPENDENCIA	5

II). Desempeño Académico: Participación en la ponderación: 50 %

Para la determinación del desempeño académico, se considera el **promedio general como el gran indicador y a los demás como correctores:**

INDICADOR	DESCRIPCION	PONDERACION
F	PROMEDIO GENERAL (incluyendo aplazos)	35
G	RENDIMIENTO ACADÉMICO	10
H	PLAN DE DESARROLLO ACADEMICO	5

CUADRO GENERAL DE PONDERACIÓN DE TODOS LOS INDICADORES

INDICADOR	DESCRIPCION	PONDERACION
A	NIVEL DE INGRESOS	30
B	CONDICION DE ACTIVIDAD LABORAL	5
C	VIVIENDA	5
D	SALUD	5
E	TASA DE DEPENDENCIA	5
F	PROMEDIO GENERAL (incluyendo aplazos)	35
G	RENDIMIENTO ACADÉMICO	10
H	PLAN DE DESARROLLO ACADEMICO	5
	TOTAL	100



Luciana Ratto

"2008 – AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado

Ordenanza N° 1180

Anexo V

MEDICION DE INDICADORES Y TABLA DE ASIGNACIÓN DE PUNTOS

A – NIVEL DE INGRESOS.

Se consideran los ingresos mensuales per cápita (X_i) del grupo familiar. Para su ponderación se considera un piso mínimo y una función continua. (Entre K_1 y K_2 pesos de ingreso)

$X_i =$ Suma Total de Ingresos del Grupo Familiar

Cantidad Total de Miembros del Grupo Familiar

$K_1 =$ Valor canasta familiar de la región: 30 puntos

$K_2 =$ K_1 multiplicado por 2 (dos): 0 puntos

Para valores intermedios de (K) se utiliza una Función continua entre K_1 y K_2 pesos:

$30 - 0,15 (X_i - K_1) = \text{Puntaje}$
--

B – CONDICION DE ACTIVIDAD LABORAL.

Para categorizar a los miembros económicamente activos (mayores de 18 años) del hogar.

1. *Personas con actividad regular rentada, por cuenta propia y/o relación de dependencia. Se incluyen en este grupo a jubilados y pensionados.*
2. *Personas con actividades rentadas inestables con ingresos no regulares, incluido subsidio por desempleo.*
3. *Persona que no recibe renta por ningún concepto.*

	CATEGORIA	VALOR
(1)	Ocupado	0
(2)	Sub-ocupado	3
(3)	Desocupado	5

Puntaje = Suma de Puntos Totales de Miembros Económicamente Activos
Cantidad de Miembros Económicamente Activos

C – VIVIENDA.

Se tomará como indicador, la situación jurídica del inmueble que habita el postulante.



Alicia Beute

"2008 – AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

Ministerio de Educación
 Universidad Tecnológica Nacional
 Rectorado

Ordenanza N° 1180

Anexo V

CATEGORIA	VALOR
Propietario sin deuda.	0
Propietario con deuda.	3
Inquilino u Ocupante.	5

D – SALUD.

Se considera la condición de cobertura de salud del estudiante exclusivamente. Dentro de obra social, se considera al servicio de medicina pre-paga con cobertura integral.

CATEGORIA	VALOR
Si tiene obra social.	0
Cobertura parcial.	3
No tiene obra social.	5

E – TASA DE DEPENDENCIA.

Se mide de acuerdo a la cantidad de hijos menores de 18 años que residan en el hogar, incluyendo los miembros discapacitados sin límite de edad.

CATEGORIA	VALOR
0 hijos	0
1 - 2 hijos	3
+ 2 hijos	5

F – PROMEDIO GENERAL.

Se multiplica el promedio general (incluyendo aplazos) por 3,5 lo que representa una función continua.

$\text{Promedio} \times 3,5 = \text{Puntaje}$

En caso de Ingresantes, el puntaje será por defecto 14 (4 x 3,5).

G – RENDIMIENTO ACADÉMICO.

Rendimiento Académico = (Materias Aprobadas / Materias Regularizadas) x 10



Asesora Beato

"2008 – AÑO DE LA ENSEÑANZA DE LAS CIENCIAS"

*Ministerio de Educación
Universidad Tecnológica Nacional
Rectorado*

Ordenanza N° 1180

Anexo V

En caso de Ingresantes, el puntaje será por defecto 4.

H – PLAN DE DESARROLLO ACADEMICO.

El mismo será evaluado entre 0 y 5 puntos a criterio de la Comisión de Becas Local.

Anexo 9D – Ley de Marcas

Ley de Marcas (Ley No. 22.362 -B.O. 2/1/81-)

CAPITULO I - DE LAS MARCAS

SECCION I - Derecho de propiedad de las marcas

ARTICULO 1 - Pueden registrarse como marcas para distinguir productos y servicios: una o más palabras con o sin contenido conceptual; los dibujos; los emblemas; los monogramas; los grabados; los estampados; los sellos; las imágenes; las bandas; las combinaciones de colores aplicadas en un lugar determinado de los productos o de los envases; los envoltorios; los envases; las combinaciones de letras y de números; las letras y números por su dibujo especial; las frases publicitarias; los relieves con capacidad distintiva y todo otro signo con tal capacidad.

ARTICULO 2 - No se consideran marcas y no son registrables:

- a) los nombres, palabras y signos que constituyan la designación necesaria o habitual del producto o servicio habitual a distinguir, o que sean descriptivos de su naturaleza, función, cualidades u otras características;
- b) los nombres; palabras, signos y frases publicitarias que hayan pasado al uso general antes de su solicitud de registro;
- c) la forma que se dé a los productos;
- d) el color natural o intrínseco de los productos o un solo color aplicado sobre los mismos.

ARTICULO 3 - No pueden ser registrados:

- a) una marca idéntica a una registrada o solicitada con anterioridad para distinguir los mismos productos o servicios;
- b) las marcas similares a otras ya registradas o solicitadas para distinguir los mismos productos o servicios;
- c) las denominaciones de origen nacionales o extranjeras.

Se entiende por denominación de origen el nombre de un país, de una región, de un lugar o área geográfica determinados que sirve para designar un producto originario de ellos, y cuyas cualidades y características se deben exclusivamente al medio geográfico. También se considera denominación de origen la que se refiere a un área geográfica determinada para los fines de ciertos productos;

- d) las marcas que sean susceptibles de inducir a error respecto de la naturaleza, propiedades, mérito, calidad, técnicas de elaboración, función, origen, precio u otras características de los productos o servicios a distinguir;
- e) las palabras, dibujos y demás signos contrarios a la moral y a las buenas costumbres;
- f) las letras, palabras, nombres, distintivos, símbolos, que usen o deban usar la Nación, las provincias, las municipalidades, las organizaciones religiosas y sanitarias;
- g) las letras, palabras, nombres o distintivos que usen las naciones extranjeras y los organismos internacionales reconocidos por el gobierno argentino;
- h) el nombre, seudónimo o retrato de una persona, sin su consentimiento o el de sus herederos hasta el cuarto grado inclusive;
- i) las designaciones de actividades incluyendo nombres y razones sociales, descriptivas de una actividad, para distinguir productos. Sin embargo, las siglas, palabras y demás signos con capacidad distintiva, que formen parte de aquéllas, podrán ser registrados para distinguir productos o servicios;
- j) las frases publicitarias que carezcan de originalidad.

ARTICULO 4 - La propiedad de una marca y la exclusividad de uso se obtienen con un registro. Para ser titular de una marca o ejercer el derecho de oposición a su registro o de su uso, se requiere un interés legítimo del solicitante o del oponente.

ARTICULO 5 - El término de duración de la marca registrada será de Diez (10) años. Podrá ser renovada indefinidamente por períodos iguales si la misma fue utilizada dentro de los Cinco (5) años previos a cada vencimiento, en la comercialización de un producto, en la prestación de un servicio, o como parte de la designación de una actividad.

ARTICULO 6 - La transferencia de la marca registrada es válida respecto de terceros, una vez inscripta en la Dirección Nacional de la Propiedad Industrial.

ARTICULO 7 - La cesión o venta del fondo de comercio comprende la de la marca, salvo estipulación en contrario.

ARTICULO 8 - El derecho de prelación para la propiedad de una marca se acordará por el día y hora en que se presente la solicitud, sin perjuicio de lo establecido en los tratados internacionales aprobados por la República Argentina.

ARTICULO 9 - Una marca puede ser registrada conjuntamente por Dos (2) o más personas. Los titulares deben actuar en forma conjunta para licenciar, transferir y renovar la marca; cualquiera de ellos podrá deducir oposición contra el registro de una marca, iniciar las acciones previstas en esta ley en su defensa y utilizarla, salvo estipulación en contrario.

Anexo 9E – Registro de dominios edu.ar

PRINCIPIOS BÁSICOS - ENUNCIADOS GENERALES

"La presente reglamentación y las modificaciones que oportunamente se introduzcan, tendrán vigencia a partir de su aparición en el sitio que la RIU (Redes de Interconexión Universitaria) posee en Internet, www.riu.edu.ar, y se aplicará a todas las soluciones pendientes de registro".

"No son susceptibles de registro las denominaciones contrarias a la moral, el orden público y las buenas costumbres".

REGLAS DEL REGISTRO

Art.1:- El registro de un determinado nombre de dominio se otorgará a la persona física o jurídica que primero lo solicite (en adelante registrante). Puede solicitar también el registro una persona física o jurídica diferente (en adelante el solicitante del registrante).

Art.2:- Podrán ser registrantes instituciones educativas oficiales y privadas o asociaciones de ellas. A tales efectos se considera institución educativa a "aquella entidad donde se cursan estudios de cualquier nivel y modalidad, que otorga títulos, certificados oficiales o con reconocimiento oficial que acreditan habilidades o condiciones para continuar estudios en otras instituciones".

Podrán ser registrantes también, organismos oficiales de los cuales dependen directamente instituciones educativas, con el único fin otorgar dominios de nivel superior a las instituciones educativas dependientes, o a otras instituciones educativas que cumplan con las mismas condiciones establecidas precedentemente para los registrantes de un subdominio de tercer nivel.

Art. 3:- Tanto el registrante como el solicitante, para solicitar un registro de nombre de dominio, deberán completar el formulario electrónico de la página Web de la RIU. El hecho de completar tal formulario supone una manifestación de conocimiento y aceptación de las reglas, procedimientos e instrucciones vigentes.

Art. 4.- A los fines de solicitar el registro del nombre de dominio, el registrante deberá proporcionar el número de CUIT (Código Único de Identificación Tributaria) y toda la información que se solicita en dicho formulario electrónico. En el caso de instituciones que integren el sistema de educación formal, el CUIT será reemplazado por el CUE (Código Único de Establecimiento).

Las solicitudes de instituciones educativas oficiales tendrán aceptación definitiva cuando la autoridad competente del organismo registrante, titular del CUIT y/o CUE, haga llegar a la RIU una nota oficial, con membrete de la dependencia y firma original del funcionario a cargo de la misma, en la que se solicite el nombre de dominio en cuestión para dicho organismo.

En el caso de instituciones privadas la solicitud presentada podrá tener aceptación definitiva cuando haga llegar a la RIU la documentación probatoria que acredite el reconocimiento oficial.

Todos los datos consignados en la solicitud de registro tendrán carácter de declaración jurada. Por tanto, al completar el formulario electrónico, el registrante, y/o solicitante, declara y garantiza que, a su leal saber y entender, toda la información proporcionada en la solicitud de nombre de dominio es correcta y verdadera. La RIU está facultada para rechazar una solicitud de registro de dominio, en caso de verificarse que la misma contiene datos falsos o erróneos.

Art. 5:- El registro del nombre de dominio tendrá una validez de un año computado a partir de la fecha de inscripción, y será renovable a petición de parte. La renovación deberá solicitarse durante el último mes de vigencia del registro. En el caso de que el registrante no la solicitara antes del cumplimiento de dicho período, se producirá la baja automática del nombre.

IMPORTANTE: La aplicación de esta regla esta suspendida hasta nuevo aviso.

Art.6:- Al efectuar el registro de un nuevo nombre de dominio, el registrante, designará un representante administrativo. Este quedará autorizado para efectuar requerimientos ulteriores sobre ese nombre de dominio por los medios previstos. La Entidad Registrante deberá comunicar inmediatamente a la RIU cualquier cambio de la Persona Responsable mediante los formularios existentes al efecto en la Web RIU.

Art.7: - Las denominaciones que contengan las palabras, letras, o nombres distintivos que usen o deban usar la Nación, las provincias y los municipios, sólo podrán ser registradas por las entidades públicas que correspondan.

En todos los casos, el nombre a registrar debe permitir identificar fácil y unívocamente a la institución que solicite el nombre de dominio, a los efectos de evitar confusiones con otras instituciones de similares denominaciones. Por ejemplo, no podrá utilizarse el nombre de próceres ni fechas patrias que identifican a una institución educacional, pues éste se repite en otras instituciones de una misma ciudad y en distintas jurisdicciones.

No serán admisibles como nombres de dominio las siglas que correspondan con nombres genéricos del sistema de nombres de dominio, por ejemplo: "telnet", "ftp", "www", "web", "smtp", "http", "tcp", "dns", "wais", "news", etc.

Art.8: - La RIU no actuará como mediador ni como árbitro, ni intervendrá de ninguna manera, en los conflictos que eventualmente se susciten entre los registrantes y/o solicitantes y/o terceros, relativos al registro o uso de un nombre de dominio.

Art.9: - El registrante es el único responsable civil y penalmente por las consecuencias que pudieran derivar de la registración, para sí y respecto de terceros, que pueda acarrear la selección de su nombre de dominio. En el caso que el registro haya sido solicitado por una persona física o jurídica diferente del registrante, ésta (denominado solicitante) será responsable solidariamente con el registrante. La RIU se limita exclusivamente a registrar el nombre de dominio indicado por el registrante y/o solicitante.

Art.10:- El hecho que la RIU registre un nombre de dominio a favor de un registrante, no implica que asuma responsabilidad alguna respecto de la legalidad de ese registro ni del uso del nombre de dominio por el registrante. No es responsabilidad de la RIU, y en virtud de ello, no le corresponde evaluar si el registro o el uso del nombre de dominio puede violar derechos de terceros. La RIU no acepta ninguna responsabilidad por cualquier conflicto por marcas registradas o sin registrar, o por cualquier otro tipo de conflicto de propiedad intelectual.

Art.11:- El registrante, y/o solicitante, que requiera el registro de un nombre de dominio en representación de una persona jurídica, declarará bajo juramento que tiene autorización del mismo para realizar la solicitud, y será responsable por cualquier error, falsedad u omisión en la información suministrada a la RIU.

Sin perjuicio de ello, la RIU se encuentra facultada para denegar o revocar un nombre de dominio en caso de que el mismo, a su criterio, se refiera a una persona física o jurídica de trascendencia y/o notoriedad pública si el registrante y/o solicitante no pudiera demostrar, a satisfacción de la RIU, que se encuentra debidamente autorizado por esa persona a efectuar tal solicitud.

Art.12:- El registrante y/o el solicitante, en el caso de tratarse de personas distintas, deben declarar bajo juramento que, de su conocimiento, el registro y uso del nombre de dominio solicitado no interfieren ni afectan derechos de terceros.

Art.13:- El registrante, y/o el solicitante en caso de tratarse de personas distintas, deben declarar bajo juramento que el registro del nombre de dominio solicitado no se realiza con ningún propósito ilegal ni viola ninguna legislación, y que todos los datos suministrados son verdaderos, no habiendo ocultado, simulado u omitido ninguna información que la RIU podría haber considerado esencial para su decisión de aceptar la solicitud del nombre de dominio. Asimismo, el registrante, y/o solicitante, se obligan a comunicar inmediatamente a la RIU cualquier modificación de los datos que se produzca.

El incumplimiento de la presente regla faculta a la RIU a rechazar la solicitud o proceder a dar inmediata baja al nombre de dominio registrado.

Art.14:- Cuando cualquier persona notifique que existe una inexactitud en la información proporcionada en la solicitud de registro de nombre de dominio, la RIU tomará las medidas razonables para investigar esa supuesta inexactitud. En caso que se determine que se ha proporcionado información inexacta, la RIU tomará las medidas razonables para corregir cualquier inexactitud, siempre que la misma no haya violado alguna de las reglas en cuyo caso denegará la solicitud o revocará el nombre de dominio.

Art.15:- La RIU podrá revocar el registro de un nombre de dominio cuando, por razones técnicas o de servicio ello sea conveniente, notificando electrónicamente al registrante. En el caso de que la revocación se realice por orden judicial, será efectivizada en el plazo que en la misma se establezca.

Art.16:- La RIU no es responsable por la eventual interrupción de los negocios, ni por los daños y perjuicios de cualquier índole que el rechazo de una solicitud, la revocación o pérdida del registro pudiera causar al registrante y/o al solicitante.

Art. 17:- El registrante y el solicitante asumen plenamente el compromiso de no responsabilizar en ningún caso a la RIU por cualquier daño y/o perjuicio que pudieran sufrir directa o indirectamente por el hecho del registro o uso del nombre de dominio.

Art. 18.- El registrante y/o solicitante reconocen que resulta técnicamente imposible suministrar un servicio libre de errores y que RIU no se compromete a ello.

Art. 19.- El registrante que hubiera obtenido de la RIU la delegación del subdominio de tercer nivel registrado, se compromete a su vez a otorgar subdominios de nivel mayor exclusivamente dentro de su propia estructura funcional o a instituciones externas que cumplen con los requisitos establecidos en la presente reglamentación.

DE LAS TRANSFERENCIAS

Art. 20.- Únicamente el registrante de un nombre de dominio podrá transferir el mismo a otra persona jurídica que reúna las condiciones y cumpla con los requerimientos establecidos en esta reglamentación y en la planilla electrónica de transferencias.

Previamente, y a tal fin, se deberá hacer llegar el acto de transferencia por instrumento público o privado, con certificación de firmas ante escribano público en donde conste:

Que el que transfiere, es efectivamente la entidad registrante, sea esta una persona jurídica o física. En este último caso, que el acto se efectúa a través de su representante legal. A tal fin, deberá acreditar ante el escribano interviniente, su calidad de representante legal de la entidad registrante como así también que posee facultades suficientes para llevar adelante la operación, constatadas y certificadas expresamente por el escribano interviniente.

En el documento de transferencia deberán constar los números de, C.U.I.T. de la entidad que transfiere el dominio y de la nueva entidad registrante.”

Art. 21.- La transferencia se operará a partir de la presentación de una solicitud de baja por transferencia por parte de la entidad registrante a través del contacto establecido para el nombre de dominio, y de la presentación subsecuente de una solicitud de registro por transferencia de parte de la nueva entidad registrante. El registro por transferencia operará como registro de un nuevo nombre de dominio a todos los efectos.