



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

SISTEMA DE CONTROL DE BECAS PARA LA INTRANET  
DEL INSTITUTO DE INGENIERÍA.

T E S I S

QUE PARA OBTENER EL TÍTULO DE:  
INGENIERO EN COMPUTACIÓN.

P R E S E N T A N :  
GUILLERMO MORA PÉREZ.  
ERIKA ESTHER TORRES VERDÍN.

DIRECTOR: ING. MARCO AMBRÍZ MAGUEY.



CIUDAD UNIVERSITARIA, MÉXICO D.F.

MAYO DE 2002

TESIS CON  
FALLA DE OR.GEN



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

*A ti mamá por tus inagotables muestras de apoyo, porque te desamañabas conmigo para tener todo listo cuando yo tenía que irme temprano a la escuela. Muchas gracias por todos los sacrificios que has hecho por mí, tú eres una de las personas más importantes de mi vida. A ti te debo la tenacidad que me ha ayudado a cumplir con los objetivos que me he propuesto.*

*¡Te quiero mucho mamita!*

*A ti papá por enseñarme a no darme por vencida y a trabajar arduamente para alcanzar mis objetivos. Muchas gracias por todo el apoyo que me diste cuando estaba estudiando, por tus consejos y por la paciencia que me demostraste cuando me ayudabas con mis tareas escolares. Tú eres muy importante para mí.*

*¡Te quiero mucho papi!*

*A mis hermanas María Amparo y María Isabel porque ambas me han apoyado y aconsejado en los momentos cruciales de mi vida y se han preocupado por mi bienestar desde que nací. También les agradezco mucho el haber celebrado, junto conmigo, el cumplimiento de mis objetivos. Son las mejores hermanas del mundo, las más divertidas, las más cariñosas y también las más apoyadoras. No se que haría sin ustedes, las quiero muchísimo.*

*A mis hermanos Víctor, Carlos y Francisco Javier, porque cada uno de ustedes contribuyó en forma diferente para que yo pudiera alcanzar mis metas. Les agradezco la paciencia infinita que siempre mostraron cuando me ayudaban con mis tareas y mis proyectos de la escuela. Les reconozco el apoyo incondicional que me han ofrecido durante toda mi vida. Los quiero mucho.*

*A mis sobrinos Miguel Ángel, Raúl Andrés y Gustavo Francisco, porque su alegría, su ternura y su sinceridad me han ayudado a valorar las cosas importantes de la vida y han hecho que mi existencia sea más divertida y agradable. Recuerden que "mag\_eri" es en honor a los tres. Los quiero una enormidad.*

*A mis amigos y compañeros de la Facultad de Ingeniería y del Instituto de Ingeniería por haber hecho de esta etapa una de las más felices y gratificantes de mi vida.*

*Erika.*

*Dedico este trabajo a mis padres, que con su esfuerzo y sacrificio diario, se comprometieron para que yo obtuviera la formación profesional a la que ellos no tuvieron acceso. Además les agradezco que me hayan enseñado que el título no hace a la persona. No existen palabras para demostrarles lo agradecido que estoy con ustedes por darme la oportunidad de alcanzar las metas que me he propuesto. Muchas gracias mamá y papá.*

*Agradezco a mi hermana y a mis hermanos por igual, porque desde que tengo uso de razón, me han apoyado en todas las formas posibles para que alcance mis objetivos y mis metas en la vida. Muchas gracias hermanos.*

*A mis compañeros y amigos de la Facultad de Ingeniería, del Instituto de Ingeniería y del Programa de Tecnología en Computación, que de alguna manera me ayudaron en la realización de este trabajo, les doy las gracias.*

*Guillermo.*

*Gracias a la Universidad Nacional Autónoma de México, en especial a todos los profesores de la Facultad de Ingeniería, quienes contribuyeron en nuestra formación como ingenieros.*

*Agradecemos el apoyo de los funcionarios del Instituto de Ingeniería de la U.N.A.M., particularmente al M.I. Roberto Magallanes Negrete, por la ayuda y el tiempo que nos dedicó durante el desarrollo del Sistema de Control de Becas.*

*Damos nuestro sincero agradecimiento al Ing. Marco Ambríz Maguey, jefe de la Coordinación de Sistemas de Cómputo del Instituto de Ingeniería, por la confianza, el apoyo, la paciencia, las múltiples asesorías y los consejos que nos ofreció durante la realización del presente trabajo.*

*Erika y Guillermo.*

## Tabla de Contenidos

<b>INTRODUCCIÓN.</b>	<b>1</b>
<b>CAPÍTULO UNO.</b>	
<b>1. Antecedentes Generales.</b>	<b>3</b>
1.1 Sobre el Instituto de Ingeniería.	3
1.2 Finalidad y orientación del Instituto de Ingeniería.	4
1.3 La Secretaría Académica del Instituto de Ingeniería.	5
1.4 Programa de Becas del Instituto de Ingeniería (PBII).	5
1.5 Reglamento del programa de becas.	6
1.6 Definición del problema de control de becas.	8
1.7 Objetivo general.	9
1.8 Objetivos específicos.	9
1.9 Alcances.	9
<b>CAPÍTULO DOS.</b>	
<b>2. Entorno del Sistema.</b>	<b>11</b>
2.1 Sistema de Control de Becas Anterior.	11
2.1.1 Características Generales.	11
2.1.2 Ventajas del sistema anterior.	21
2.1.3 Desventajas del sistema anterior.	22
2.1.4 Problemas encontrados al sistema.	23
2.2 Sistema de Recursos Humanos del Instituto de Ingeniería (SRHII).	24
2.2.1 Sobre el Sistema de Recursos Humanos.	24
2.2.2 Evaluación de la organización de la información del SRHII.	24
2.2.3 Alternativas de desarrollo del sistema utilizando la base de datos del SRHII.	25
<b>CAPÍTULO TRES.</b>	
<b>3. Fundamentos Tecnológicos.</b>	<b>27</b>
3.1 Tecnologías Existentes en el Mercado.	27
3.2 Sistemas operativos de los servidores de web.	27
3.3 Servidores de web.	29
3.4 Arquitecturas de computadoras más utilizadas.	29
3.5 Manejadores de Bases de Datos.	30
3.6 Herramientas de Programación de Tecnología Web.	33
3.6.1 PHP.	34
3.6.2 ASP.	36
3.6.3 JSP.	38
3.7 Infraestructura del Instituto de Ingeniería,	41
3.7.1 Red Local del Instituto de Ingeniería.	41
3.7.2 Servidores.	41
3.7.3 Manejadores de Bases de Datos.	42
3.7.4 Servidores de web.	42

**CAPÍTULO CUATRO.**

<b>4. Elementos y Herramientas de desarrollo del Sistema.</b>	<b>43</b>
4.1 Elección de los elementos y herramientas de desarrollo.	43
4.1.1 Manejador de Bases de Datos.	43
4.1.2 Servidor de web.	44
4.1.3 Herramientas de Programación.	45
4.1.3.1 Tecnología PHP.	45
4.1.3.2 Tecnología ASP.	46
4.1.3.3 Tecnología JSP.	47
4.1.3.4 Comparación entre las tecnologías PHP, ASP y JSP.	48
4.1.3.5 Herramienta de Programación Seleccionada.	49
4.2 Hardware y Arquitectura de Desarrollo.	53
4.3 Propuesta de despliegue de los componentes del sistema para su operación.	53
4.3.1 Desarrollo de la Propuesta.	53

**CAPÍTULO CINCO.**

<b>5. Análisis, Diseño e Implementación del Sistema.</b>	<b>57</b>
5.1 Recopilación de Requerimientos	57
5.1.1 Requerimientos del Programa de Becas del Instituto de Ingeniería.	57
5.1.2 Requerimientos del registro de los prestadores de Servicio Social.	66
5.1.3 Requerimientos del registro de los estudiantes sin estipendio.	69
5.2 Modelado de Datos	71
5.2.1 Diccionario de Datos.	71
5.2.2 Diagrama Entidad Relación.	75
5.3 Análisis del Sistema.	76
5.3.1 Casos de Uso.	76
5.3.1.1 Actores del sistema.	76
5.3.1.2 Acciones de los actores.	76
5.3.1.3 Descripción de los casos de uso.	77
5.3.2 Diagramas de Casos de Uso.	88
5.3.3 Diagramas de Actividades.	90
5.4 Diseño del Sistema	92
5.4.1 Diagramas de Clases.	92
5.4.2 Diagramas de Secuencia.	96
5.5 Implementación del Sistema.	100
5.5.1 Diagramas de Componentes.	100

**CAPÍTULO SEIS.**

<b>6. Pruebas y resultados finales.</b>	<b>103</b>
6.1 Descripción y resultados de las pruebas realizadas al sistema.	103
6.1.1 Pruebas de funcionalidad.	103
6.1.2 Pruebas de regresión.	104
6.1.3 Pruebas de navegación.	105
6.1.4 Pruebas de compatibilidad.	105
6.1.5 Pruebas de utilización.	106
6.1.6 Pruebas de desempeño.	107
6.2 Descripción del Sistema Terminado.	110

6.2.1 Registro de un estudiante nuevo.	110
6.2.2 Registro de los datos de un estudiante.	111
6.2.3 Asignación de una beca.	115
6.2.4 Proceso de aprobación de una beca.	118
6.2.5 Información generada para las Secretarías Académica y Administrativa.	120
6.3 Resultados Generales.	121
6.4 Propuestas para la siguiente versión.	121
<b>CAPÍTULO SIETE.</b>	
<b>7. Conclusiones.</b>	<b>123</b>
<b>ANEXOS.</b>	
A. Fragmento del Reglamento Interno del Instituto de Ingeniería.	127
B. Reglamento del PBII.	133
C. Propuesta del reglamento para el uso del SCBII.	138
<b>BIBLIOGRAFÍA.</b>	<b>143</b>

# INTRODUCCIÓN



## Introducción.

La necesidad de desarrollar el sistema de información que se describe en el presente trabajo, surgió como consecuencia de la falta de un mecanismo automatizado para que el Instituto de Ingeniería pudiera agilizar el proceso de evaluación de las becas que otorga.

Anteriormente se implementó un sistema de información que pretendió alcanzar el objetivo citado; sin embargo, ese sistema cumplía parcialmente las expectativas del Instituto, por lo que no tuvo una buena aceptación inicial y no se utilizó.

Lo anterior fue determinante para decidir nuestra participación en el desarrollo de un sistema de información completamente nuevo, que además de contemplar el control del proceso de aprobación de las becas, permitiera modificar, guardar y consultar los datos personales y académicos, así como la trayectoria de cada uno de los becarios y los exbecarios dentro del Instituto. Asimismo, era una necesidad primordial considerar diferentes niveles de seguridad, dependiendo de la jerarquía del usuario que accediera al sistema.

Tomando en cuenta lo sucedido al sistema anterior, decidimos que antes de empezar el desarrollo del nuevo sistema, requeriéramos estudiar y analizar las necesidades actuales del Instituto relativas al control de las becas. Inicialmente consultamos el Reglamento del Programa de Becas y el Reglamento Interno del Instituto, con el fin de establecer los requisitos generales; posteriormente sostuvimos diversas entrevistas con algunos funcionarios que nos permitieron percibir las expectativas y las necesidades que el sistema debería cubrir para su utilización.

En el presente trabajo se muestran las diferentes fases que conforman el desarrollo del Sistema de Control de Becas del Instituto de Ingeniería.

En el capítulo uno se describen los antecedentes generales del sistema y se define la importancia del Programa de Becas del Instituto de Ingeniería (PBII), además de sus alcances y sus objetivos, así como el problema del control de becas.

El capítulo dos se refiere al anterior sistema de control de becas y se analiza la base de datos en donde actualmente se guarda toda la información referente al personal de Instituto, el cual incluye a los becarios.

En el capítulo tres, se realiza un breve estudio de las tendencias actuales del mercado, en cuanto a las tecnologías y las herramientas para desarrollar sistemas de información. Asimismo, se describe la infraestructura de cómputo que actualmente tiene el Instituto de Ingeniería.

En el capítulo cuatro se hace una comparación entre las diferentes tecnologías y las herramientas existentes en el mercado, con los recursos de

cómputo que tiene el Instituto. Finalmente, se toma una decisión en cuanto a los elementos que se utilizaron para el desarrollo del sistema, tomando en cuenta el estudio de capítulo tres y la comparación realizada en el capítulo 4.

El capítulo cinco contempla las partes más importantes del proceso de análisis, diseño e implementación del sistema. En él se describen y se analizan los requerimientos del sistema y se muestran los diagramas de UML necesarios para visualizarlo.

En el capítulo seis se explican las pruebas que se hicieron al sistema; se muestra un breve recorrido del mismo, se plantea el trabajo que se espera realizar en el futuro y se describen los resultados obtenidos.

Finalmente en el capítulo siete se exponen las conclusiones a las que llegamos en esta tesis.

# CAPÍTULO UNO

## Antecedentes Generales

## **1. Antecedentes Generales.**

### **1.1 Sobre el Instituto de Ingeniería.**

El Instituto de Ingeniería de la Universidad Nacional Autónoma de México es el centro de investigación en diversas áreas de la Ingeniería más productivo del país. Integra a una comunidad de aproximadamente 740 personas, constituida por 95 investigadores, 95 técnicos académicos, 370 becarios de licenciatura y postgrado y 180 empleados administrativos. El 94% de los investigadores cuenta con estudios de postgrado, 70% con doctorado y 24% con maestría.

Desde su fundación, en 1956, la política del Instituto ha sido realizar investigación orientada a problemas generales de la Ingeniería, así como colaborar con entidades públicas y privadas para mejorar la práctica de la ingeniería en el ámbito nacional, al aplicar los resultados de las investigaciones a problemas específicos. Como consecuencia, algunos proyectos son financiados con recursos que la UNAM otorga, y otros, mediante contratos de investigación con empresas o corporaciones solicitantes. El ingreso promedio generado por este concepto, ha alcanzado en los últimos años, 140 millones de pesos anuales.

Las principales organizaciones para las que se han realizado proyectos de investigación, desarrollo tecnológico y asistencia técnica, son:

- Agencia Internacional para el Desarrollo (USAID).
- Asociación Mexicana de Instituciones de Seguros (AMIS).
- Banco Interamericano de Desarrollo.
- Cementos Mexicanos (CEMEX).
- Comisión Federal de Electricidad.
- Comisión Nacional del Agua.
- Comunidad Europea.
- Consejo Nacional de Ciencia y Tecnología (CONACYT).
- E&M Inc. USA.
- Fundación México-Estados Unidos para la Ciencia, AC.
- Gobierno del Distrito Federal.
- Hayes Wheels Acero, SA de CV.
- Instituto Mexicano del Petróleo.
- Instituto Nacional para la Educación de los Adultos.
- Kelvinator de México, SA de CV.
- Petróleos Mexicanos (PEMEX).
- Secretaría de Comunicaciones y Transportes (SCT).
- Secretaría de Educación Pública (SEP).
- SEMARNAP.

## **1.2 Finalidad y orientación del Instituto de Ingeniería.**

El Instituto de Ingeniería es parte del Subsistema de Investigación Científica de la Universidad Nacional Autónoma de México y orgánicamente se encuentra dentro de la Coordinación de la Investigación Científica.

Los principales objetivos del Instituto son desarrollar investigación para mejorar los conocimientos, métodos y criterios en Ingeniería, contribuir a la formación de expertos en esta rama del saber, así como promover la más alta calidad en la práctica profesional.

Las principales funciones del Instituto son:

- Realizar investigación para mejorar los conocimientos, métodos y criterios en Ingeniería, tanto fundamental como aplicada.
- Formación de investigadores en Ingeniería.
- Apoyar al desarrollo tecnológico y análisis de los requerimientos sociales a cuya solución puede aportar la Ingeniería.
- Proporcionar servicios de Ingeniería a los diversos sectores de la sociedad con el propósito de contribuir al avance de los objetivos propios de la Universidad.
- Apoyar la formación de profesores y las tareas docentes de la Facultad de Ingeniería.
- Estudiar problemas de interés nacional.
- Difundir los resultados de sus investigaciones.
- Llevar a cabo las actividades necesarias para realizar las funciones precedentes.

En el desempeño de estas funciones, el Instituto colabora con otras instituciones afines, técnicas, culturales y científicas, del país y en el extranjero.

Las actividades que se llevan a cabo en el Instituto son: investigación técnica y aplicada, apoyo al desarrollo tecnológico y análisis de los requerimientos sociales a cuya solución puede aportar la Ingeniería. Asimismo, se proporcionan servicios de Ingeniería a los diversos sectores de la sociedad con el propósito de contribuir al avance de los objetivos propios de la universidad.

Es importante enfatizar que gran parte del futuro del Instituto se encuentra cimentado en el proyecto de la Torre de Ingeniería (Torre del II). Este proyecto

responde a los renovados criterios de operación del Instituto, dentro de los cuales hace algunos años se considera, además del impulso a la investigación teórica, el establecimiento de convenios de trabajo y la asociación temporal con empresas privadas o instituciones de gobierno, con el fin de estimular la investigación aplicada.

Las actividades del Instituto generan infinidad de aplicaciones prácticas que contribuyen al desarrollo de la industria y tecnología mexicanas, a partir del conocimiento, talento y habilidad de los investigadores universitarios en todas las ramas de la ingeniería. Para favorecer la labor de investigación estrechamente vinculada a las empresas e instituciones interesadas en promover y aprovechar los avances en Ingeniería, se creó la Torre del II, que proporcionará un espacio adecuado para este propósito.

### **1.3 La Secretaría Académica del Instituto de Ingeniería.**

La Secretaría Académica colabora con la Dirección en la gestión de recursos humanos, materiales y financieros para cumplir con las funciones del Instituto. Estas funciones son investigar, formar, enseñar, difundir y vincular. Las actividades de gestión implican labores de planeación, organización, operación, regulación, información y evaluación.

Para el sistema que se planea desarrollar, la colaboración con la Secretaría Académica es muy importante durante la fase de requerimientos, debido a que promueve la formación de futuros profesionistas y la preparación de ingenieros a través del Programa de Becas del Instituto de Ingeniería (PBII). Realiza también los trámites relacionados con períodos sabáticos, promociones, permisos, becas al extranjero, intercambios académicos y repatriaciones, entre otros.

### **1.4 Programa de Becas del Instituto de Ingeniería (PBII).**

En congruencia con las actividades que promueve, el Instituto confiere relevancia especial a su tarea de formación de personal especializado, para lo cual incorpora en sus proyectos numerosos estudiantes de licenciatura y postgrado. Estos estudiantes se benefician al completar su formación con la práctica de la investigación en Ingeniería, y contribuyen a los resultados de los proyectos. El Instituto cuenta con un programa propio de becas que complementa otros programas vigentes en la Universidad y que permite enfatizar la importancia de la actividad docente.

El Programa de Becas del Instituto (PBII) integra actualmente a dos tipos de estudiantes:

- Becarios que colaboran en un proyecto de investigación y reciben un estipendio mensual cuyo monto se determina de acuerdo al nivel de estudios del estudiante (desde licenciatura hasta doctorado).

- Prestadores de Servicio Social que colaboran en algún proyecto para liberar este trámite. Estos estudiantes no reciben estipendio por su labor.

Todos los estudiantes, sobre todo de la Facultad de Ingeniería de la UNAM, que reúnan los requisitos estipulados por el reglamento del PBII, podrán ser becarios del Instituto para realizar actividades escolares, participar activamente en proyectos de investigación del mismo, contar con tutoría académica individual y en su caso recibir un estipendio.

La importancia del PBII radica en que su objetivo central es coadyuvar a la formación de personal de alto nivel académico en las especialidades de la Ingeniería que se cultivan en el Instituto para apoyar a la investigación, la docencia y la práctica profesional

### **1.5 Reglamento del programa de becas.**

Del Reglamento General de Becas se pueden obtener los siguientes puntos:

1. - Al PBII pueden ser aceptados estudiantes de cualquier nivel superior (licenciatura, maestría y doctorado), sin importar la Institución de procedencia.<sup>1</sup>
2. - Existen dos tipos de becas:
  - a) Regulares
    - De estudio para cursar asignaturas y participar en proyectos de investigación.
    - De tesis para realizar solamente actividades de Investigación.
  - b) Especiales que se otorgarán, en casos excepcionales, para lograr propósitos complementarios a la obtención del grado, tales como: prestar servicio social, cumplir requisitos de admisión, participar en programas de capacitación e intercambio u otros aprobados por el Consejo Interno.<sup>2</sup>
3. - El PBII acepta estudiantes:
  - a) Sin beca externa.
  - b) Con beca externa (cuyo monto se complementará en caso de ser necesario).<sup>3</sup>

<sup>1</sup> Capítulo 1 del Reglamento del PBII

<sup>2</sup> Capítulo 2 del Reglamento del PBII

<sup>3</sup> Capítulo 3, tercer párrafo del PBII

4. - La duración las becas es de un semestre. Las becas regulares sólo pueden renovarse un máximo de cuatro veces en licenciatura y maestría, y hasta nueve veces en el caso de doctorado con maestría incluida.

Las becas especiales pueden ser renovadas el número de veces que sean necesarias.<sup>4</sup>

5. - Se puede cambiar de tutor o de área de investigación después de un semestre y por una sola ocasión.<sup>5</sup>

6. - Se puede apelar ante la Dirección del Instituto la cancelación o no renovación de la beca, así como la suspensión temporal del estipendio.<sup>6</sup>

7. - El procedimiento de primer ingreso al PBII.<sup>7</sup>

- a) El becario junto con su tutor llena la solicitud, la cual será entregada al coordinador correspondiente, después de haber cumplido con los requisitos establecidos por el PBII
- b) El coordinador junto con el subdirector evalúa la solicitud.
- c) El director autoriza el ingreso del solicitante de acuerdo al resultado de evaluación.

Dentro del proceso de alta deben de anexarse a la solicitud los siguientes documentos:

- El historial académico.
- El currículum.
- El horario de trabajo.
- El programa de actividades.
- En su caso, la copia de la carta de aprobación del tema de tesis.

8. - El procedimiento de renovación al PBII.<sup>8</sup>

- a) El becario junto con su tutor llenan la solicitud, la cual será entregada al coordinador correspondiente, después de haber cumplido con los requisitos establecidos por el PBII
- b) El coordinador junto con el subdirector evalúan la solicitud.
- c) El director autoriza el reingreso del solicitante de acuerdo al resultado de evaluación.

<sup>4</sup> Capítulo 4 del Reglamento del PBII

<sup>5</sup> Capítulo 6, inciso h del Reglamento del PBII

<sup>6</sup> Capítulo 6, inciso i del Reglamento del PBII

<sup>7</sup> Capítulo 9, punto 9.1 del Reglamento del PBII

<sup>8</sup> Capítulo 9, punto 9.2 del Reglamento del PBII



Dentro del proceso de renovación deben de anexarse a la solicitud los siguientes documentos:

- El historial académico.
- El currículum.
- El horario de trabajo.
- El programa de actividades del próximo semestre.
- El informe de actividades del semestre anterior.

### **1.6 Definición del problema de control de becas.**

Actualmente para solicitar una beca al PBII, se debe de llenar un formato que incluye los datos personales del estudiante, sus antecedentes académicos y las fuentes de financiamiento de la beca. A esta solicitud se le debe de anexar el currículum, el historial académico, el horario, el plan de trabajo del siguiente semestre y el informe de actividades del estudiante.

El principal problema del control actual de las becas, radica en que el llenado del formato descrito anteriormente se realiza en forma manual por parte de los becarios y posteriormente es evaluado de la misma forma por los directivos del Instituto. Este proceso es administrado por la Secretaría Académica del organismo.

Anteriormente y a petición de la Secretaría Académica del Instituto, se implementó un Sistema de Control de Becas que cumplía parcialmente con las necesidades que entonces requería el PBII. Además de no alcanzar las expectativas, el sistema presentaba fallas constantes en su funcionamiento. Los problemas, ventajas y desventajas que se encontraron en este sistema, se describen detalladamente en el siguiente capítulo.

Actualmente, debido a la importancia del PBII, la necesidad de un sistema de control de estudiantes es persistente. Es por esto, que la Secretaría Académica del Instituto ha solicitado el desarrollo de un nuevo sistema de cómputo que facilite y agilice las labores de registro y control de los estudiantes que desean colaborar en el Instituto (como becarios, prestadores de servicio social, tesisistas o personal de apoyo), pero sobretodo que cuente con un acceso permanente a los datos de estos estudiantes y a la información de su trayectoria dentro de la institución. Asimismo, para los funcionarios involucrados en el proceso de solicitud de una beca, es necesario tener acceso a un sistema que les pueda proporcionar estadísticas que los ayuden a tomar decisiones de acuerdo a sus necesidades y a su rango en la Institución.

La nueva versión del Sistema de Control de Becas, cuyo análisis y desarrollo es el tema central del presente trabajo, puede verse como el

primero de diversos módulos de un sistema de información integral que a largo plazo ofrezca la vinculación con otros sistemas manejados por el Instituto, tal y como lo es el Sistema de Recursos Humanos, administrado por la Unidad Central de Procesos de la Secretaría Administrativa, cuya función principal es el manejo de los pagos del personal que labora en el Instituto, incluyendo a los becarios.

### **1.7 Objetivo General.**

El objetivo principal de este trabajo es analizar, diseñar e implementar un sistema para el control de las becas otorgadas por el Instituto de Ingeniería, el cual pueda ser utilizado únicamente a través de la red interna de la institución, por medio de una interfaz de web y que ofrezca diferentes niveles de acceso a la información de acuerdo con la jerarquía del usuario.

### **1.8 Objetivos Específicos.**

A continuación se enlistan los objetivos específicos de esta tesis:

- Recopilación de los requerimientos del Sistema de Control de Becas con diferentes funcionarios del Instituto.
- Análisis de los requerimientos del sistema, utilizando como herramienta al Lenguaje Unificado de Modelado.
- Investigación sobre las herramientas y elementos de desarrollo que existen en el mercado para la creación de la aplicación. Asimismo, hacer una descripción de la infraestructura y recursos con los que cuenta el Instituto para el desarrollo del sistema.
- Propuesta de las herramientas y elementos de desarrollo del sistema, considerando los recursos del Instituto.
- Desarrollo del Sistema de Control de Becas.
- Documentación del diseño, implementación y pruebas del sistema.

### **1.9 Alcances.**

- El Sistema de Control de Becas que se describe en esta tesis, contempla la inscripción de nuevos becarios al PBII, la actualización de los datos de los becarios existentes y el proceso de aprobación de una beca. En versiones futuras del sistema, se planea desarrollar los módulos para los prestadores de servicio social y los estudiantes sin estipendio.
- La captura de los requerimientos del sistema contempla a los becarios y a los funcionarios involucrados en el proceso de aprobación de una beca. Asimismo, los requerimientos incluyen los apartados para el registro de

prestadores de servicio social y de estudiantes sin estipendio. De acuerdo con los objetivos establecidos anteriormente, esta tesis únicamente describe el análisis de los requerimientos relacionados con el Programa de Becas del Instituto de Ingeniería (PBII).

- El diseño e implementación de la base de datos del sistema, contempla además de la información relacionada con los becarios, la de los prestadores de servicio social y los estudiantes sin estipendio.
- Se presenta una descripción de la implementación del sistema, únicamente para su etapa de desarrollo. En cuanto a la etapa de producción del sistema, se hace una propuesta de la arquitectura en la que residirá.
- Las pruebas que se describen en este trabajo, fueron hechas únicamente dentro de la Coordinación de Sistemas de Cómputo. Las versiones futuras del sistema podrán ser probadas en el resto de las unidades organizacionales del Instituto de Ingeniería.
- Las limitaciones que presenta la organización de la información del Sistema de Recursos Humanos de la Secretaría Administrativa, que es el encargado de generar los pagos de los becarios, no permiten establecer una comunicación directa con la base de datos del Sistema de Control de Becas.
- El SCBII generará un archivo con la información requerida por el Sistema de Recursos Humanos, para expedir los pagos a los becarios del Instituto.
- Con el objetivo de establecer las políticas de uso del SCBII, en esta tesis se presenta una propuesta del reglamento del sistema.

# CAPÍTULO DOS

## Entorno del Sistema

## **2. Entorno del Sistema.**

En el capítulo uno se describió el contexto institucional del Sistema de Control de Becas (SCBII) y se explicó la importancia del mismo para el Instituto de Ingeniería. Asimismo, se mencionó que anteriormente se desarrolló un sistema que cumplía parcialmente con los requerimientos del PBII.

En la primera parte de este capítulo se describen las características generales del sistema anterior, así como sus ventajas y desventajas. La segunda parte consiste en la descripción y la evaluación general de la base de datos del Sistema de Recursos Humanos del Instituto. Este estudio se realizó a petición del Secretario Académico, con el objetivo de analizar la factibilidad de utilizar esta base de datos para el desarrollo del nuevo SCBII.

### **2.1 Sistema de Control de Becas Anterior.**

#### **2.1.1 Características Generales.**

El sistema anterior se basaba en una arquitectura cliente/servidor de tres capas lógicas, utilizando como cliente un navegador web y un servidor de web y un servidor de bases de datos como su contraparte. Para tener acceso a los datos del sistema se utilizó el manejador de bases de datos Postgresql, la interfaz web fue programada en Perl y además funcionaba sobre el sistema operativo Linux.

El servidor de web, el de bases de datos y el código fuente se alojaban en el mismo equipo. En cuanto a este equipo, se puede mencionar que es una PC "HP Brio" con procesador Intel Pentium III de 550 MHz ,64 MB en RAM y disco duro con capacidad de 8GB.

El procedimiento de alta o de renovación de una beca se dividía en 2 procesos principales. El primero de estos procesos consistía en el ingreso de los datos académicos y personales del becario o aspirante a becario. El segundo proceso consistía en la aprobación de la solicitud de beca por parte de su tutor, el coordinador, el subdirector, y el director del área en donde el becario pretendía realizar el proyecto.

El ingreso de los datos del becario, estaba conformado por varios formularios, a los cuales el becario tenía acceso proporcionando su número de cuenta de la UNAM o su RFC (ver figura 1).

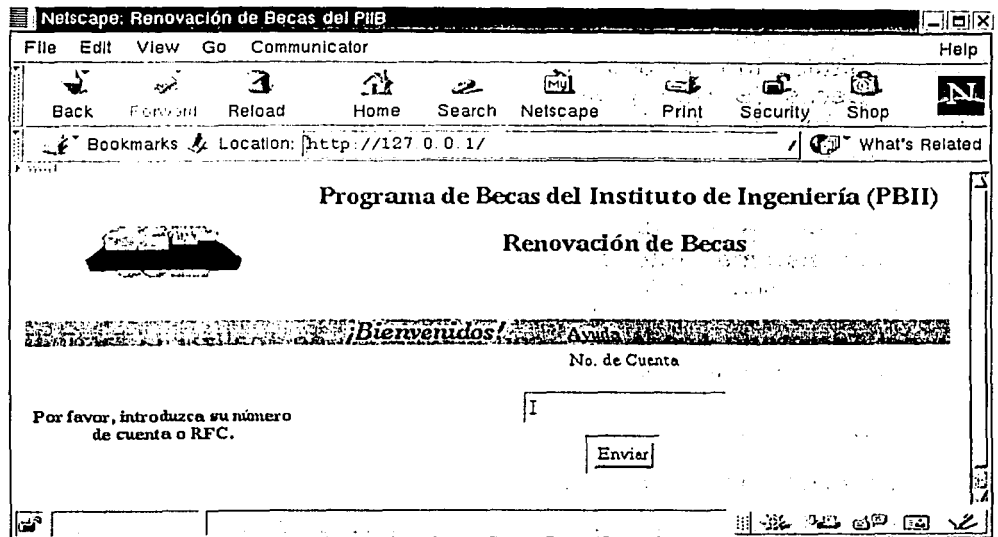


Figura No. 1

La información requerida por la primera forma eran los datos personales y los antecedentes académicos del interesado (ver figura 2). En la sección de datos personales, todos los campos eran obligatorios, excepto el de teléfono. En la sección de antecedentes académicos, los datos de institución, créditos y promedio correspondientes a la licenciatura eran obligatorios, los datos para maestría y doctorado dependían del grado de estudios del solicitante.

**Netscape: Verificación de Datos**

File Edit View Go Communicator

Bookmarks Location: <http://132.248.156.90/cgi-bin/datos1.pl> / What's Related

### Programa de Becas del Instituto de Ingeniería (PBII)

#### Renovación de Becas

**DATOS PERSONALES Y ANTECEDENTES ACADEMICOS**

**Nombre:** LUIS G. **Sexo:** M  
**Apellidos:** VELASQUILLO MARTINEZ

**Domicilio:**

**Calle y Número:** TOLTECAS 20 MZ 13 **Lugar de Nacimiento:**  
MEXICO  
**Colonia:** SANTO DOMINGO  
**Población:** COYOACAN D.F. **Fecha de Nacimiento:**  
1961  
**Código Postal:** 04360  
**Teléfono:**

**ANTECEDENTES ACADEMICOS**

Institución	Créditos	Cuadro	Promedio
<b>Licenciatura:</b> INAH	100.0		70.00
<b>Maestría:</b> I	70.00		70.00
<b>Doctorado:</b> INS DE PHYSIQU	100.0		70.00

**Continuar**

Figura No. 2

La tercera forma hace referencia a los datos del período de becas anterior, la cual asumía que el becario ya había estado inscrito anteriormente en el programa. En este formulario se incluían el nombre, los apellidos del becario, el número de materias que cursó durante el ciclo precedente, su avance de créditos, así como todos los datos relacionados con el proyecto anterior, tales como su número, su nombre y el período de vigencia, incluyendo los datos del tutor y de la coordinación a cargo del proyecto. Además, se debía de anexar el reporte de actividades del becario. Finalmente se tenía que introducir el número de proyecto que se pretendía realizar para el siguiente semestre (ver figura 3).

Netscape: Verificación de Datos

File Edit View Go Communicator Help

Bookmarks Location: http://132.248.156.90/cgi-bin/datos2/ What's Related

### Programa de Becas del Instituto de Ingeniería (PBII)

#### Renovación de Becas

---

**INFORME SEMESTRE ANTERIOR** Ayuda

**Nombres:** LUIS G.

**Apellidos:** VELASQUILLO MARTINEZ

**Programa vigente del:** /199/ (del 7/0/20/)

**Número de Proyecto:** 8562      **Nombre:** Neotectónica del Itsmo de Tehuantepec y su impacto en el riesgo sísmico de la región

**Tutor:** CHAVEZ GONZALEZ MARIO      **Número de asignaturas:** 0.00

**Coordinación:** Ingeniería Simulógica      **Créditos:** 0.00

**Reporte de actividades:**

El desarrollo del proyecto que hice el semestre pasado no arrojó los resultados

---

**Programa para el próximo semestre:**

**Número de Proyecto:**

Figura No. 3

El cuarto formulario se refería a la información del programa para el siguiente semestre, la cual se basaba en el número de proyecto que se introdujo en el formulario anterior. En esta parte de la solicitud, el becario podía seleccionar a su tutor y además debía de incluir un programa de actividades para el próximo período. El becario también actualizaba su promedio y avance de créditos. Este formulario concluía el proceso de solicitud de inscripción al programa de becas (ver figura 4).



Netscape: Continúa el proceso

File Edit View Go Communicator Help

Bookmarks Location: http://132.248.156.90/cgi-bin/tutor.pl What's Related

## Programa de Becas del Instituto de Ingeniería (PBII)

### Renovación de Becas

---

**SELECCIÓN DE PROGRAMA PARA EL PROXIMO SEMESTRE**

Nombre: \_\_\_\_\_

Número de Proyecto: **8562**      Neotectónica del Istmo de Tehuantepec y su impacto en el riesgo sísmico de la región.

Selección de tutor: **CHAVEZ GONZALEZ MARIO**

Asignaturas:       Créditos:

---

**Programa de actividades para el próximo semestre:**

Figura No. 4

Como se mencionó anteriormente, el proceso de alta o de renovación de una beca se dividía en 2 procesos principales. El primero de estos procesos es el que consistía en el ingreso de los datos académicos y personales, lo cual se acaba de describir. El segundo consistía en la aprobación de la beca por parte de cada una de las autoridades involucradas, como lo son el tutor, el coordinador, el subdirector y el director. Para poder tener acceso al segundo proceso del sistema, la autoridad correspondiente tenía que introducir su rfc, su contraseña e indicar el puesto que desempeñaba dentro del Instituto. (ver figura 5).

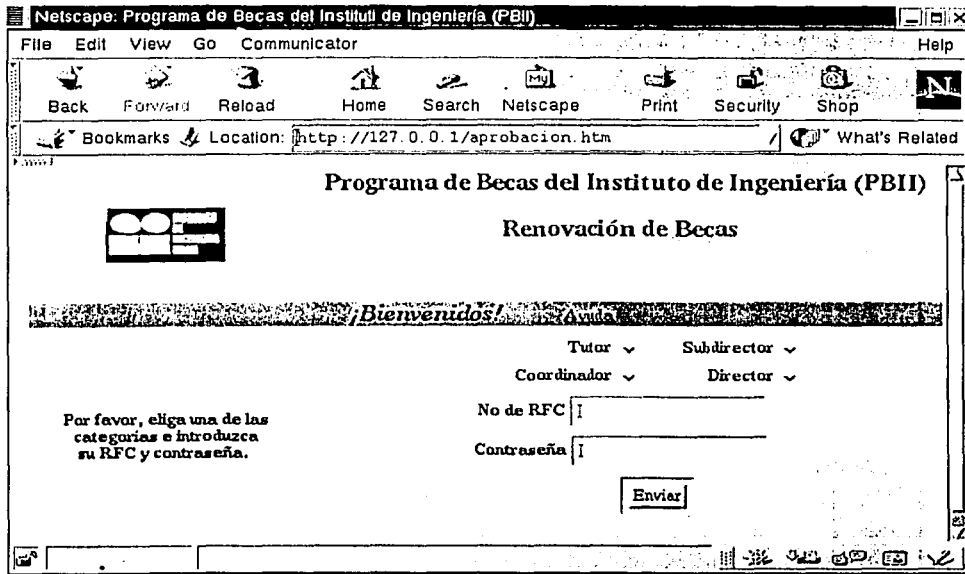


Figura No. 5

Una vez que los datos requeridos eran introducidos en el formulario anterior, se desplegaba una nueva forma, y en caso de que la autoridad involucrada tuviera becas pendientes por aprobar, hacía un desplegado de los nombres de los becarios que esperaban por su aprobación, para que se eligiera una solicitud a la vez (ver figura 6).

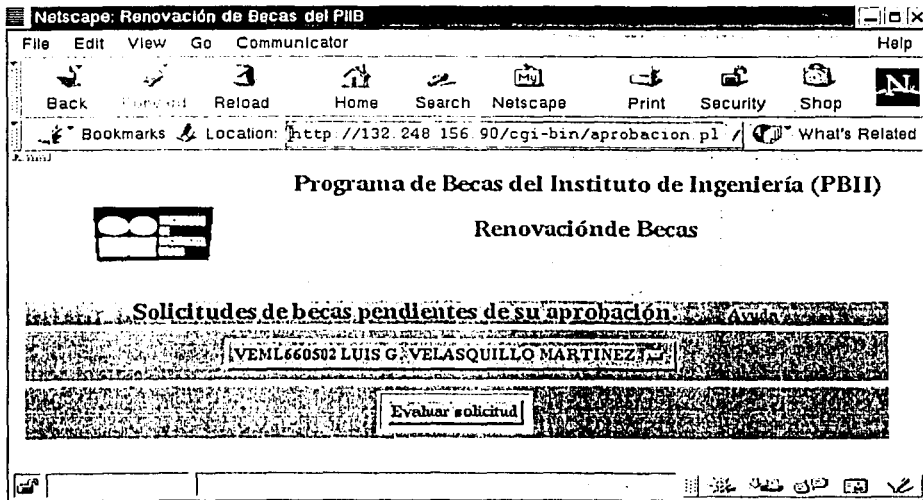


Figura No. 6

En caso de que la autoridad correspondiente no tuviera solicitudes pendientes, se mostraba una forma que explicaba esta situación (ver figura 7).

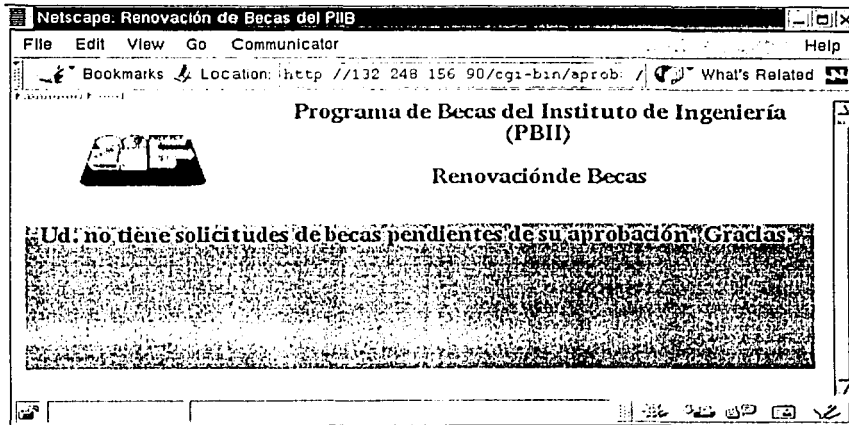


Figura No. 7

En caso de que la autoridad correspondiente tuviera solicitudes por aprobar, eligiera alguna de ellas y presionara el botón de evaluar solicitud, se desplegaba una nueva forma, en la cual aparecían los datos personales y académicos del solicitante, también se podía tener acceso al informe del periodo anterior y al plan de trabajo para el siguiente periodo (ver figura 9). Asimismo se visualizaban los datos de nómina, que incluían el periodo de inicio y término de la beca, el monto de cada una de las instituciones que podían financiar la beca, así como el nivel de estudios del solicitante. En caso de que la autoridad que ingresaba a este formulario fuera el tutor del becario, éste tenía que llenar los campos asociados al tipo y monto de la beca (ver figura 8a). Si la autoridad que accedía al formulario tenía mayor jerarquía en el proceso de aprobación, sólo podía verificar la información sin modificarla (ver figura 8b). Finalmente, en ambos casos, se observaban tres opciones a elegir: aprobar, rechazar y pendiente.

File Edit View Go Communicator

Bookmarks Location: http://132.248.156.90/cgi-bin/preedi.pl

### Programa de Becas del Instituto de Ingeniería (PBII)

#### Renovación de Becas

**Nombre:** LUIS G.

**Apellidos:** VELASQUILLO MARTINEZ

**Domicilio:**

**Calle y Número:** TOLTECAS 10 MZ. 13

**Colonia:** SANTO DOMINGO

**Población:** COYOACAN D.F.

**Código Postal:** 04360

**Teléfono:**

Institución	Créditos Cubiertos	Promedio
Licenciaturas: UNAM	100.00	0.00
Maestrías:	0.00	0.00
Doctorado: INS. DE PHYSIQU	100.00	0.00

[Ver informe y programa de actividades](#)

**Datos de nómina:**

Inicio: 01/03/2000    Term: 29/09/2000    Categoría-nivel: 1 becario de licenciatura

DGAPA: 100    CONACYT: 200    EXTERNA: 300    PBII: 400

Figura No. 8a

Netscape: Verificación de Datos

File Edit View Go Comunicador Help

### Programa de Becas del Instituto de Ingeniería (PBII)

#### Renovación de Becas

**Datos personales y antecedentes académicos** [Ayuda](#)

**Nombres:** JOSE FERNANDO

**Apellidos:** MALDONADO SALGADO

**Domicilio:**

**Calle y Número:** CERRO SN. ANDRES NO. 188

**Colonia:** CAMPESTRE CHURUBUSCO

**Población:** MEXICO D.F.

**Código Postal:** 04200

**Teléfono:** 5494111

**Lugar de Nacimiento:** MEXICO

**Fecha de Nacimiento:** 27/08/70

Institución	Creditos Completados	Promedio
Licenciatura: ENEP ARAGON	36.00	8.00
Maestría:	0.00	0.00
Doctorado:	0.00	0.00

**Historial**

Inicio: /030/    Term.: /092/    Categoría-nivel:1 becario de licenciatura  
 DGAPA: 100    CONACYT: 200    EXTERNA: 300    PBII: 400

[Aprobar](#) [Rechazar](#) [Pendiente](#)

[Terminar](#)

Figura No. 8b

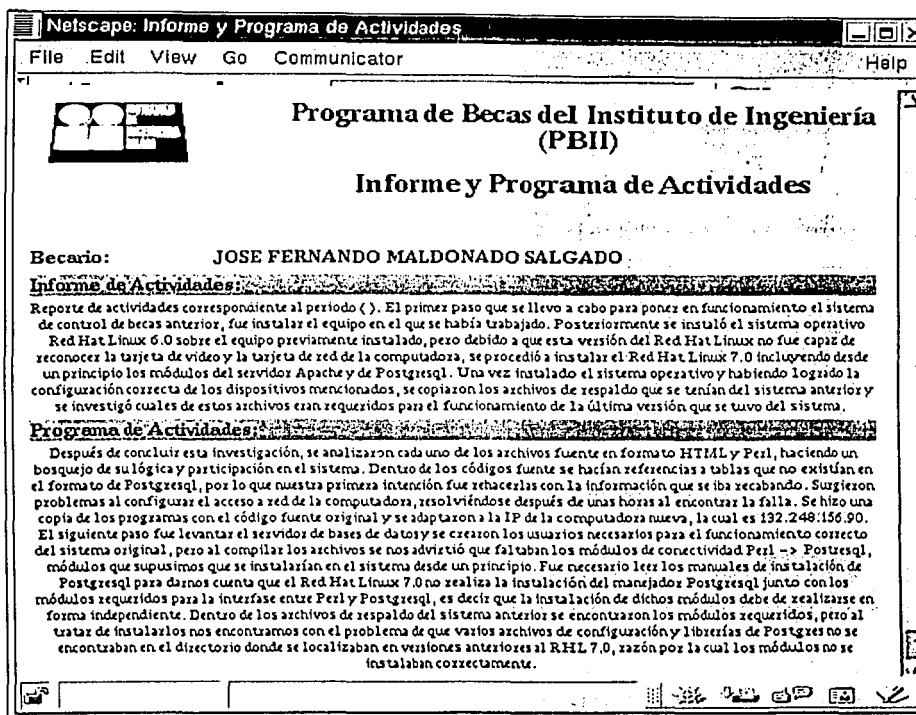


Figura No. 9

En caso de escoger la opción de aprobar o la de rechazar (sólo una a la vez), se notificaba a la autoridad correspondiente el resultado de la elección (ver figura 10). Cuando un tutor aprobaba una solicitud, su coordinador correspondiente era notificado. Cuando el coordinador aprobaba dicha solicitud, la decisión se transmitía al subdirector encargado de la coordinación. De la misma forma, el director era notificado de la decisión de aprobación del subdirector.

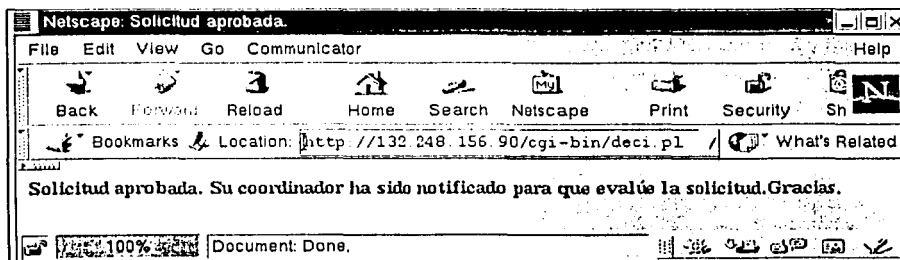


Figura No. 10

En caso de que cualquier autoridad rechazara una solicitud, el tutor responsable era el único en ser notificado (ver figura 11).

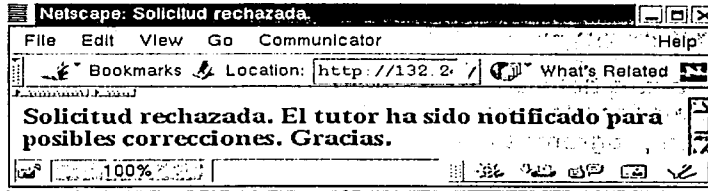


Figura No.11

En caso de escoger la opción de pendiente, la solicitud quedaba en espera de alguna de las otras dos alternativas.

### 2.1.2 Ventajas del sistema anterior.

1. Por ser una aplicación con arquitectura cliente/servidor el sistema ofrece las siguientes ventajas:
  - Proporciona cierta distribución de la carga de trabajo que sufre el sistema.
  - La interfaz gráfica del usuario reduce el tiempo de aprendizaje del proceso de inscripción y de aprobación de becas.
  - La aplicación puede ser accedida en cualquier momento y desde cualquier lugar, incluso fuera de la red interna del Instituto.
  - Permite un mejor aprovechamiento de los programas existentes de los clientes y con los cuales están familiarizados los usuarios.
  - La interfaz y la arquitectura de los clientes es independiente de la plataforma en la que reside el sistema.
  - Varios clientes pueden tener acceso al sistema al mismo tiempo.
  - Reduce el tráfico de la red transfiriendo únicamente la información necesaria.
  - La falla en cualquier computadora cliente no afecta al resto del sistema, teniendo un mayor control del mismo.
2. El uso de CGI's permite ejecutar en tiempo real las consultas, por lo que reduce la inconsistencia en la información y disminuye la carga de trabajo para el servidor. Los CGI's son también herramientas muy eficaces para la manipulación de una base de datos por parte del usuario sin importar el tipo de manejador de bases de datos que se utilice.

3. Debido a que el sistema reside dentro de una plataforma Linux, los módulos del servidor de bases de datos Postgresql, los del servidor de web Apache y los CGI's de Perl tienen un mejor desempeño en su funcionamiento.
4. El costo del software del sistema es nulo, debido a que el sistema operativo Linux, el manejador de bases de datos Postgresql y el intérprete del lenguaje Perl son programas con licencia pública y gratuita.
5. Linux en comparación con Windows NT, necesita de menos recursos de hardware para mantener el sistema funcionando a una velocidad mayor, es decir, Linux ofrece un menor tiempo de procesamiento. Además, el nivel de seguridad proporcionado por Linux es mejor.
6. Debido a que el sistema de becas reside en una única PC, tanto el costo del hardware y como el de su mantenimiento es mínimo.

### **2.1.3 Desventajas del sistema anterior.**

1. Una aplicación cliente/servidor, como lo es el sistema de becas, requiere de un fuerte rediseño de todos los elementos involucrados (modelos de datos, procesos, interfaces, comunicaciones, almacenamiento de datos, etc.). Además, en la actualidad existen pocas herramientas que ayuden a determinar la mejor forma de dividir las aplicaciones entre la parte cliente y la parte servidor.
2. El sistema fue realizado en un lenguaje de programación estructurado, por lo que su funcionamiento queda limitado exclusivamente al proceso de solicitud y aprobación de becas. Esto significa que el código no puede ser reutilizado para otras aplicaciones con finalidades similares, dificultando el desarrollo de sistemas integrados institucionalmente. Esta característica da como resultado un sistema cuyo mantenimiento es complejo.
3. El manejador de bases de datos Postgresql, no puede residir en plataformas diferentes a Unix, haciéndolo dependiente del sistema operativo.
4. El sistema operativo Linux y el manejador de bases de datos Postgresql, tienen la desventaja de que al ser gratuitos carecen de un servicio de soporte técnico.
5. Es muy difícil asegurar un buen nivel de seguridad en un sistema centralizado en una sola computadora.
6. Aunque la arquitectura lógica es de tres capas, todas las consultas del cliente hacia la base de datos son directas, sin que la carga de trabajo pueda ser compartida entre el servidor de web y el de bases de datos.



7. Al residir el sistema en una computadora personal, el hardware carece de los recursos necesarios para atender una carga de trabajo extrema, la cual puede ser causada debido a que en esta misma máquina se controla tanto al servidor de bases de datos como al servidor de páginas web, obteniéndose así un sistema lento y altamente propenso a sufrir interrupciones. Este carácter centralizado del sistema junto con la carencia de servidores redundantes, son factores determinantes que pueden provocar la pérdida o la inconsistencia de la información.
8. El sistema es totalmente dependiente de la intranet del Instituto, por lo que en caso de presentarse algún problema en el funcionamiento de ésta, el sistema se vuelve inaccesible.

#### **2.1.4 Problemas.**

En particular, el sistema actual presenta los siguientes problemas de programación y de lógica:

1. Cada una de las conexiones realizadas a la base de datos en los scripts de Perl no fueron cerradas, lo que puede llegar a producir la inminente caída del sistema.
2. El cliente hace conexiones innecesarias al servidor de web para hacer la validación de campos, lo que disminuye el rendimiento del sistema.
3. Los scripts de los CGI's repiten código que pudo haber sido escrito una sola vez, lo que dificulta los cambios a futuro que requieran las formas del sistema, es decir su mantenimiento.
4. El diseño de la base de datos no fue normalizado, por lo que la consulta a las tablas no es óptima y puede haber inconsistencia de datos.
5. Los informes y los reportes de actividades no pueden exceder de 8104 caracteres (aproximadamente menos de una página en formato de Word), debido a que la base de datos no permite tener campos mayores a esta longitud.
6. En el sistema se asume que todas las solicitudes de beca son para renovación, no se considera el caso de nuevo ingreso al programa.
7. El becario es el único que puede verificar el estado de la aprobación de su beca, sin que las demás autoridades involucradas tengan este conocimiento.
8. El sistema permite abrir dos sesiones diferentes al mismo tiempo, utilizando una misma clave de identificación, lo que puede provocar inconsistencia en los datos.

9. Cada vez que se rechaza una solicitud de beca, se desconoce la razón de esta decisión.
10. El tutor no puede monitorear el estado de las becas que ha promovido.
11. El password de las autoridades no está encriptado en la base de datos.
12. El sistema actual carece de la documentación necesaria para su instalación, funcionamiento y mantenimiento.

## **2.2 Sistema de Recursos Humanos del Instituto de Ingeniería (SRHII).**

### **2.2.1 Sobre el Sistema de Recursos Humanos.**

El Sistema de Recursos Humanos del Instituto de Ingeniería tiene la función de controlar el manejo del personal de la institución. Los becarios forman parte del personal del Instituto, por lo tanto el Sistema de Recursos Humanos contiene información que podría ser empleada por el Sistema de Control de Becas.

El Sistema de Recursos Humanos es utilizado internamente por la Secretaría Administrativa del Instituto. El Secretario Académico es el único funcionario externo a la Secretaría Administrativa que puede tener acceso al Sistema de Recursos Humanos y sólo puede hacer cambios y consultas de la parte académica del mismo.

### **2.2.2 Evaluación de la organización de la información del Sistema de Recursos Humanos.**

Analizando la base de datos del Sistema de Recursos Humanos se identificaron varias tablas cuya información es útil para el Sistema de Control de Becas. Desafortunadamente desde el punto de vista de cómputo, la estructura de la información encontrada no cumple con las normas básicas requeridas en el diseño de bases de datos.

Dentro de los problemas que presenta la base de datos en cuanto al manejo de la información, se pueden mencionar los siguientes:

- a) No se respeta la estructura organizacional del Instituto.
- b) Se repite información innecesariamente.
- c) La información en la base de datos está mezclada, un ejemplo de esto es que no se hace una distinción clara entre la información de los becarios y la del resto del personal.
- d) Existen diversos campos de información que ya no se utilizan y ocupan espacio innecesario.

Es un hecho que el sistema funciona correctamente, a pesar de las deficiencias que presenta en la organización de su base de datos. Aun tomando en cuenta este punto, no es adecuado crear un sistema de información que utilice una base de datos que presenta problemas de diseño.

En el siguiente apartado se presentan tres alternativas de desarrollo para el SCBII, tomando en cuenta las características de la base de datos del Sistema de Recursos Humanos.

### **2.2.3 Alternativas de desarrollo del sistema utilizando la base de datos del SRHII.**

De acuerdo con el análisis de la base de datos del Sistema de Recursos Humanos del Instituto expuesto anteriormente, se definieron las tres alternativas de desarrollo del Sistema de Control de Becas que se describen a continuación:

#### **Primera Alternativa.**

Consiste en pasar por alto las deficiencias de diseño que presenta la base de datos del Sistema de Recursos Humanos y adecuarla para el funcionamiento del sistema propuesto.

#### **Ventajas:**

- Al tener acceso a una base de datos que ya está funcionando, se agilizaría el desarrollo del Sistema de Control de Becas.

#### **Desventajas:**

- Para cumplir con algunos de los requerimientos del Sistema de Control de Becas, la base de datos del Sistema de Recursos Humanos necesitaría ser modificada, lo cual implica sólo resolver el problema a corto plazo, pero en lo sucesivo haría más evidente la deficiencia en la estructura lógica de la base de datos.
- Existiría redundancia e inconsistencia en la información debido a que la base de datos del Sistema de Recursos Humanos no cumple con las reglas de normalización básicas.

#### **Segunda Alternativa.**

La segunda alternativa consiste en diseñar una base de datos independiente para el Sistema de Control de Becas, la cual contendría toda la información que este sistema requiere y podría ser vista como el punto de partida, para el rediseño de toda la base de datos del sistema utilizado por la Secretaría Administrativa.

**Ventajas:**

- La base de datos cumpliría con todas las normas establecidas para su diseño. Esto ayudaría a garantizar el buen funcionamiento y el correcto desempeño del sistema.
- La base de datos formaría parte del primer módulo de un sistema integral de información que contendría en un futuro, toda la información administrativa del Instituto.

**Desventajas:**

- Hasta el momento en que se rediseñara el Sistema de Recursos Humanos del Instituto, habría redundancia con respecto a la información existente en la base de datos de este sistema.
- Debido a la forma en que actualmente está estructurado el Sistema de Recursos Humanos, no podría haber una interacción directa entre éste y el Sistema de Control de Becas.

**Tercera Alternativa.**

La tercera opción consiste en rediseñar por completo la estructura de la base de datos actual del Sistema de Recursos Humanos.

**Ventajas:**

- A largo plazo representaría la mejor opción, porque permitiría tener almacenada en una sola base de datos toda la información concerniente al Instituto, además de que cumpliría con los requerimientos de diseño mínimos que toda base de datos debe de tener.

**Desventajas:**

- Diseñar nuevamente la base de datos que utiliza el Sistema de Recursos Humanos requiere de un largo periodo de desarrollo, lo que implica posponer el desarrollo del Sistema de Control de Becas.
- Su desarrollo rebasa los objetivos establecidos en la presente tesis.

**Alternativa seleccionada.**

Debido a la necesidad casi inmediata de un Sistema de Control de Becas para el Instituto, se decidió trabajar sobre la segunda alternativa. La segunda opción, ofrece la seguridad de desarrollar un sistema cuya base de datos cumpla con los requerimientos de diseño básicos.

# CAPÍTULO TRES

## Fundamentos Tecnológicos

### 3. Fundamentos Tecnológicos

En este capítulo se estudia la disponibilidad en el mercado de los elementos tecnológicos que se necesitan para desarrollar el sistema y se analizan los recursos actuales del Instituto de Ingeniería. En el siguiente capítulo se hace una comparación entre ambos y posteriormente se propone una solución para la creación e implementación del SCBII.

#### 3.1 Tecnologías Existentes en el Mercado.

Es difícil determinar con exactitud cifras referentes a las tecnologías de cómputo más utilizadas en la actualidad. Cada compañía involucrada en el negocio afirma ser la mejor en diferentes aspectos o productos, presentando estadísticas que nunca dejarán de ser dudosas. No existen entonces verdades absolutas en este tema por lo que hay que tomarlo con el mayor cuidado posible.

Las cifras presentadas en los apartados correspondientes a servidores de web, sistemas operativos y arquitecturas de computadoras más usadas se basan en los estudios realizados por la compañía inglesa Netcraft.com, dedicada entre otros aspectos a la consultoría y al análisis de tecnologías de Internet. Mediante metodologías tecnológicamente aceptables, esta compañía ha hecho un estimado<sup>1</sup> de las tendencias actuales en el uso de sistemas operativos, servidores de web y arquitecturas de computadoras conectadas y accesibles públicamente desde Internet.

Las cifras correspondientes a los manejadores de bases de datos más usados se basan en la investigación realizada por Dataquest Inc., compañía que forma parte de Gartner Inc, firma dedicada a la investigación y consultoría de soluciones tecnológicas empresariales en Estados Unidos.

#### 3.2 Sistemas operativos de los servidores de web.

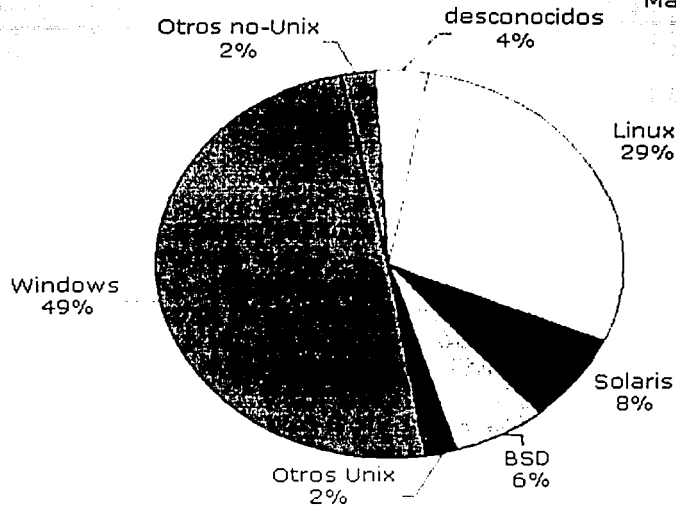
De un total de 29,302,656 sitios de web contabilizados a marzo de 2002, la siguiente figura esquematiza los sistemas operativos que utilizan los servidores de web que son accedidos públicamente desde Internet.

<sup>1</sup> Las cifras son estimados debido a que una IP o un grupo de ellas puede contener cientos o hasta decenas de miles de sitios web, además de que cada sitio web puede tener una serie de sinónimos en los servidores de nombres. Las cifras resultantes del estudio de Netcraft.com fueron parcialmente filtradas con estos criterios para dar una mayor aproximación a la realidad. Aun así se contempla un margen de error que no incluye a aquellos servidores que no pudieron ser contados por estar detrás de un firewall o de un servidor virtual.

TRABAJOS CON  
FALTA DE ORIGEN

### Conteo Mundial de Sistemas Operativos de los Servidores Web

Marzo 2001



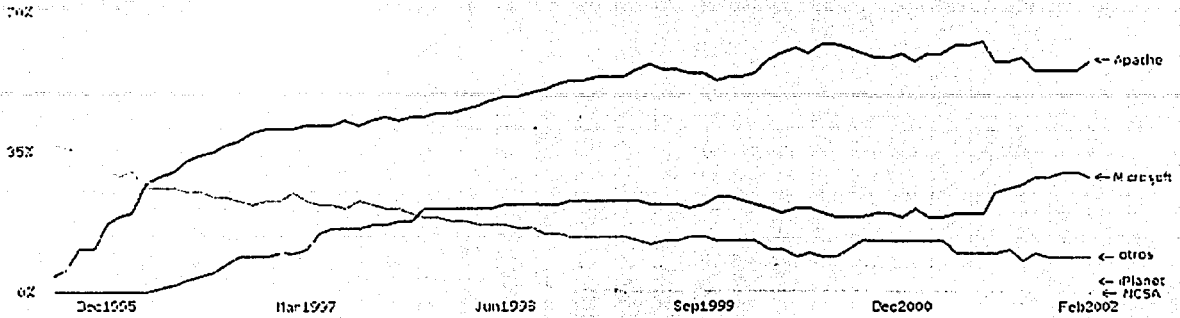
Grupo de Sistemas Operativos	Porcentaje	Composición
<b>Windows</b>	49.2%	Windows 2000, NT4, NT3, Windows 95, Windows 98
<b>Linux</b>	28.5%	Todas las distribuciones de Linux
<b>Solaris</b>	7.6%	Solaris 2, Solaris 7, Solaris 8
<b>BSD</b>	6.3%	BSDI BSD/OS, FreeBSD, NetBSD, OpenBSD
<b>Otros Unix</b>	2.4%	AIX, Compaq Tru64, HP-UX, IRIX, SCO Unix, SunOS 4 y otros
<b>Otros no-Unix</b>	2.5%	MacOS, NetWare y sistemas IBM
<b>Desconocidos</b>	3.6%	Desconocidos por el detector de sistemas operativos.

Fuente: netcraft.com

Cabe destacar que si se filtran los resultados con base al dominio al que pertenece el servidor, las cifras no tienen el mismo comportamiento, superando en varios casos Linux a Windows (sobre todo en instituciones educativas).

### 3.3 Servidores de web.

De un total de 38,444,856 sitios web, la siguiente figura muestra el uso de los servidores de web a través del tiempo, comenzando desde el año de 1995 hasta Febrero de 2002.



La gráfica anterior muestra el porcentaje actual del uso de servidores de web.

Grupo de Servidores	Porcentaje	Composición
<b>Apache</b>	56.87%	Apache-1.3.*, Apache-AdvancedExtranetServer, Apache-NeoWebScript, Apache-SSL, Apache-SSL-US, Apache4AME, ApacheFP, ApachePro, ApacheSSL, ApacheSSL-DWEB, Apache-1.3.14-(Unix), Citysearch-Apache, Houselit-Apache, InfoWest.COM-Custom-Apache, Rapidsite, Red, Stronghold, Superboard-Apache, Telesat-Apache, etc
<b>Microsoft</b>	30.25%	Microsoft-IIS, Microsoft-IIS-W, Microsoft-PWS-95, & Microsoft-PWS
<b>iPlanet</b>	3.60%	iPlanet-Enterprise, Netscape-Enterprise, Netscape-FastTrack, Netscape-Commerce, Netscape-Communications, Netsite-Commerce y Netsite-Communications.
<b>Otros</b>	9.28%	Zeus, Rapidsite, thttpd, webSitePro, StrongHold, WebStar, WebLogig, Roxen-WWW-Server, Roxen-Webserveretc.

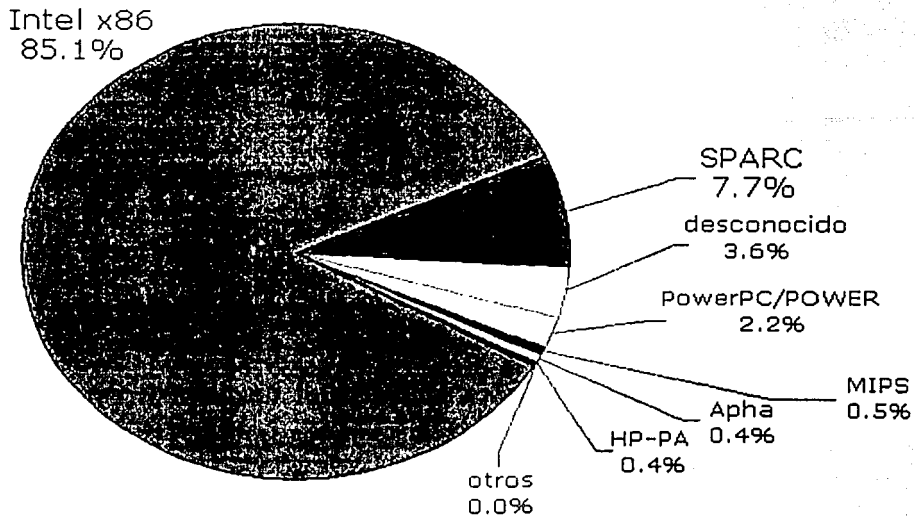
Fuente: netcraft.com

### 3.4 Arquitecturas de computadoras más utilizadas.

Las arquitecturas de procesadores utilizadas por los servidores de web que fueron rastreados por Netcraft.com presentan la distribución mostrada por la siguiente gráfica.



## Arquitecturas de Microprocesadores usadas por servidores web públicos



Fuente: netcraft.com

### 3.5 Manejadores de Bases de Datos

Los DMBS (Database Management System o Sistema Manejador de Bases de Datos) son programas que organizan la información de manera que su consulta, almacenamiento y modificación se lleve a cabo de la manera más eficiente posible. Los dos principales tipos de DBMS que se encuentran disponibles en el mercado son los siguientes:

- a) Los relacionales conocidos como RDBMS (Relational Database Management System o Sistema Manejador de Bases de Datos Relacional).
- b) Los orientados a objetos.

En la actualidad, el uso de los manejadores de bases de datos relacionales es mayor al de los relativamente nuevos manejadores orientados a objetos. Esto se debe a que los estándares para las bases de datos relacionales están totalmente definidos, lo que significa que al utilizarlos se dispone de una mayor variedad de DBMS de donde elegir y de una mayor cantidad de bibliografía.

Las diferencias entre un RDBMS y otro, radican principalmente en la forma de administrar la base de datos, debido a que cada compañía implementa características adicionales que extienden la funcionalidad del RDBMS.

Para que un DBMS pueda ser reconocido como relacional debe de manejar SQL (Structured Query Language; Lenguaje Estructurado de Consulta) como lenguaje para consultar y manipular la información de la base de datos.

En la siguiente tabla se muestran algunos de los RDBMS disponibles en el mercado.

Nombre	Plataformas sobre las que está disponible	Licencias de Uso
<b>Oracle</b>	Solaris Linux Windows NT	Gratuito con fines de desarrollo Gratuito con fines de desarrollo Gratuito con fines de desarrollo
<b>Adaptative Server (Sybase)</b>	Solaris Linux Windows NT	Necesita Licencia Comercial Gratuito con fines de desarrollo Necesita Licencia Comercial
<b>MS SQL Server</b>	Windows	Necesita Licencia Comercial
<b>My SQL</b>	Unix Windows	Distribución Libre Necesita Licencia Comercial
<b>PostgreSQL</b>	Unix	Distribución Libre

Debido a la dificultad que presenta el rastreo de manejadores de bases de datos conectados (DBMS) en Internet, fue necesario utilizar una fuente de información distinta a la utilizada con los servidores de web, la cual permitiera mostrar las tendencias en la utilización de productos de bases de datos. Un indicador confiable de estas tendencias son las ganancias anuales que presentan las compañías de software por la venta de licencias de sus productos de bases de datos. Desafortunadamente estas cifras no representan la utilización de las bases de datos de distribución gratuita.

El estimado en ventas de licencias de manejadores de bases de datos registradas en los Estados Unidos durante el año 2000, así como las cifras de 1999 se muestra en la siguiente tabla

Compañía	% de mercado en 1999	% de mercado en 2000
<b>Oracle</b>	31.4	33.8
<b>IBM</b>	29.9	30.1
<b>Microsoft</b>	13.1	14.9
<b>Sybase</b>	3.3	3.2
<b>Informix</b>	5.0	3.0
<b>Otras</b>	17.3	15.0

Fuente: Gartner Dataquest (Mayo 2001)<sup>2</sup>

Otros datos obtenidos de la misma fuente revelan que el 80% del mercado es dominado por los manejadores de bases de datos relacionales (RDBMS), mientras el resto es ocupado por los manejadores de bases de datos orientados a objetos (OODBMS).

<sup>2</sup> Consulte: [www.itworld.com/AppDev/119/DataquestOracleexte413](http://www.itworld.com/AppDev/119/DataquestOracleexte413)

Las tendencias cambian dependiendo de la plataforma en la que son usados los DBMS. Para plataformas Windows NT, Oracle es ligeramente superada por Microsoft, representando el 38% y 37.3% respectivamente de las ventas totales en este segmento, dejando en tercer lugar a IBM con el 18.5% del mercado. Para plataformas Unix, Oracle representa el 66.2%, IBM el 14.4% e Informix el 6.3%.

En cuanto a los DBMS de distribución libre, se pueden mencionar los siguientes:

- Beagle SQL
- Diamond
- GNU SQL Server
- Internase (para Linux)
- LEAP
- MDBMS
- mSQL (MiniSQL)
- MySQL
- Pql
- PostgreSQL
- PREDATOR
- SQRL

Es difícil determinar cual de estos DBMS es el más utilizado, debido a que por ser de distribución libre, no es posible rastrear las tendencias de uso a través de registros de ventas. Asimismo, las compañías encargadas de realizar las estadísticas relacionadas con las ventas y utilización de software, sólo se interesan en las que tienen que ver con DBMS comerciales.

Los mismos creadores de DBMS como PostgreSQL<sup>3</sup>, afirman que debido que no manejan un esquema de licencias o de registro, es difícil establecer un número fijo de los usuarios del DBMS aunque se cree que podrían ser cientos, mismos que han podido detectar a través de los correos que les envían.<sup>4</sup>

Debido a esta situación, resulta muy complicado encontrar estadísticas que incluyan a los DBMS comerciales y a los DBMS gratuitos. Después de diversas búsquedas en distintos sitios de Internet se encontraron los resultados de una encuesta que la revista de tecnología de la información, eWeek<sup>5</sup>, hizo a sus lectores para determinar que DBMS resulta más importante para su empresa.

Los resultados de la encuesta se muestran a continuación:

DBMS	%
IBMDB2	6.25
Informix	2.24

<sup>3</sup> Computer Science Div., Dept. of EECS University of California at Berkeley.

<sup>4</sup> Consulte: <http://postgresql.org>, <http://www.math.eku.edu/tools/FAQ-Postgres.html>

<sup>5</sup> Consulte: [http://www.eWeek.com/poll\\_archive/0,3666,p=1144&bn=1,00.asp](http://www.eWeek.com/poll_archive/0,3666,p=1144&bn=1,00.asp)

Microsoft SQL Server	15.03
MySQL	23.70
Ingres	0.59
Oracle	30.18
PostgreSQL	15.92
Sybase Adaptive Server Enterprise	3.87
Sybase SQL Anywhere	0.88
Otros(Poet, Jasmine, Versant, Matisse)	0.34

Es importante observar que esta encuesta incluye a DBMS de distribución libre como PostgreSQL y MySQL, que son los más conocidos de su tipo. También se puede observar que el DBMS más popular es Oracle, mientras que los DBMS de Sybase tienen una menor aceptación que PostgreSQL y MySQL.

### 3.6 Herramientas de Programación de Tecnología Web.

Uno de los ejes fundamentales que diferencian a Internet de otros medios de comunicación es la interacción y la personalización de la información con el usuario, que permite desarrollar contenido en línea dependiendo de, por ejemplo, el perfil del visitante o los datos que éste completó en un formulario. Estas aplicaciones se conocen como aplicaciones de Web dinámicas que actualmente prevalecen en sitios de comercio electrónico, en donde la información desplegada se genera a partir de la información obtenida a través de una base de datos o de otra fuente externa.

El lenguaje para programar páginas HTML, es en sí mismo estático. Para crear páginas interactivas han surgido lenguajes específicos de programación para la Web, los cuales podemos diferenciarlos en dos tipos:

- 1) Los que trabajan del lado del cliente (en el navegador del usuario).

Algunos ejemplos de este tipo de lenguajes son Javascript, Java (applets) y todos los plug-ins o ActiveX que posea el navegador, tales como Flash o VRML.

- 2) Los que trabajan en el servidor (donde está alojada la página).

La programación en el servidor es mucho más potente ya que no estamos dependiendo del poder de procesamiento de la computadora que posea el usuario que está visualizando la página. La ventaja principal de las tecnologías dependientes del servidor radica en la seguridad que tiene el programador sobre su código, ya que éste se encuentra únicamente en los archivos del servidor que al ser solicitado a través del web es ejecutado, por lo que los usuarios no tienen acceso más que a la página resultante en su navegador.

Estas tecnologías pueden ser subdivididas en:

a) HTML incrustado en programas (stand alone programs): CGI's y Servlets. Generan código HTML en la salida estándar. Una diferencia importante entre los CGI's y los Servlets radica en que los CGI's son scripts que se interpretan en tiempo de ejecución, en cambio los Servlets son programas parcialmente compilados que requieren menos recursos del sistema para su ejecución.

b) Programas incrustados en HTML (server side scripting): PHP, ASP y JSP. Mezclan código HTML con el código específico de la aplicación. Cuando estas páginas son requeridas, se analiza y separa el código HTML del código del lenguaje de aplicación, a este último lo compila y al código HTML lo deja intacto. El Código compilado generará HTML dinámico, el cual se mezclará a la salida con el código HTML estático.

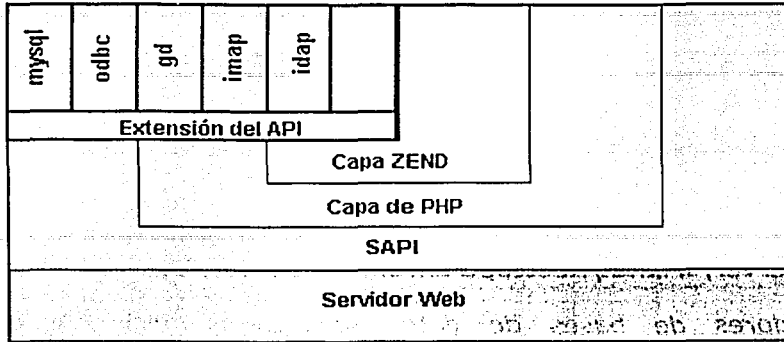
### **3.6.1 PHP.**

Originalmente creado en el año de 1994 por Rasmus Lerdorf, PHP (Hypertext Preprocessor) es un lenguaje interpretado, el cual se considera un híbrido de C, JavaScript, UNIX Shell y Perl. Con la diferencia de que PHP se encuentra incrustado dentro de los códigos HTML existentes del lado del servidor de web.

#### *Componentes de PHP4.*

PHP cuenta con un componente intrínseco denominado motor Zend, el cual es el intérprete del lenguaje PHP. Asimismo, los módulos de funciones llamados extensiones PHP, también están incluidos dentro del paquete. Un componente de gran utilidad es la capa de abstracción del servidor de web, llamada SAPI, que simplifica la tarea de añadir soporte nativo para nuevos servidores de web. SAPI también tiene la ventaja de añadir estabilidad a PHP4, ya que soporta servidores de web multihilo.

La siguiente figura presenta la arquitectura de PHP. El borde superior de la figura representa la interfaz con el programador.



Actualmente, SAPI tiene implementaciones para los servidores Apache, Roxen, Java, ISAPI, AOLServer y CGI.

Las funciones PHP son parte de una de las capas que se muestran en la esquina superior izquierda de la figura anterior. Muchas funciones, tales como el soporte para MySQL, son proporcionadas mediante extensiones. La mayoría de las extensiones son opcionales, éstas se pueden enlazar dentro de PHP en tiempo de compilación o construirse dinámicamente cargándose bajo demanda.

**Características:**

- *Rápido aprendizaje.* Por ser similar al lenguaje C y no basar su funcionamiento en algún otro lenguaje, es rápido de aprender y relativamente simple.
- *Escalabilidad limitada.* PHP puede ser escalado, la limitante que existe se debe a que los aspectos de escalabilidad los maneja el servidor de web en el que reside, esto es, la escalabilidad depende de componentes externos a PHP.
- *Características del lenguaje.* PHP puede ser utilizado para manejar contenido dinámico, bases de datos, sesiones, archivos, cookies, etc. Todo esto orientado a páginas web en formatos HTML y DHTML. Además es capaz de manipular formatos XML e interactuar con archivos de Flash. Es capaz de generar múltiples formatos de salida, archivos binarios, incluyendo formatos como JPEG, PNG, GIF y PDF, soportando también la carga y descarga de archivos binarios y de texto. Asimismo PHP puede interactuar con otros servicios utilizando protocolos como IMAP, SNMP, NNTP, POP3, HTTP, entre otros

- *Portabilidad.* PHP es multiplataforma gracias a su gran variedad de librerías que le permiten residir en sistemas Linux, Solaris, Windows NT, MacOS BSD.
- *Componentes soportados.* Soporta componentes COM, además soporta la característica de herencia de los Lenguajes Orientados a Objetos, además de poder manipular objetos de Java como si fuesen propios de PHP, permitiendo la ejecución de Servlets.
- *Servidores Web Soportados.* Corre en servidores Apache, IIS, Zeus, AOLServer, iPlanet/Nestcape Enterprise Server, etc.
- *Servidores de bases de datos soportados.* PHP permite ejecutar comandos SQL incrustados en el código HTML, soportando manejadores de bases de datos de Sybase, Oracle, MySQL, ODBC, PostgreSQL, MS-SQLServer, Informix, dBase entre otros.
- *Mantenimiento del código.* Ocultando funciones complejas y relegándolas a capas inferiores dentro de su arquitectura, hace más simples las operaciones que en otros lenguajes llevarían más líneas de código.
- *Costo.* Por ser software libre, el costo es nulo.

### 3.6.2 ASP.

Microsoft introdujo esta tecnología llamada Active Server Pages (ASP) en diciembre de 1996.

El principio de la tecnología ASP es el VBScript, pero existe otra diversidad de lenguajes de programación que pueden ser utilizados como lo es Perl, JScript, etc. ASP es una tecnología dinámica funcionando del lado del servidor, lo que significa que cuando el usuario solicita un documento ASP, las instrucciones de programación dentro del script son ejecutadas para enviar al navegador únicamente el código HTML resultante.

En realidad ASP es solamente una implementación específica de un ISAPI (Internet Server API), que en este caso oculta la complejidad de los protocolos inherentes y las interfaces del sistema. El manejador "ASP.dll" es una extensión ISAPI, que es llamado cada vez que se accede a una página con extensión ASP.

*Los objetos y componentes de ASP.*

Los ASP's proporcionan objetos y componentes. La función de ambos es manejar la interacción entre el servidor de web y el navegador de web. Un objeto es un conjunto de funciones relacionadas y piezas de información. Los componentes son archivos con extensión ".dll", que vienen junto con un ASP. Los componentes contienen una amplia variedad de objetos. Dentro de los

ASP's existen objetos intrínsecos que están relacionados con distintos aspectos de interactividad. Existen 7 objetos ASP intrínsecos, los cuales se describen a continuación.

Los siete objetos built-in son:

- Petición: Reúne la información solicitada por el usuario.
- Respuesta: Manda al usuario la información solicitada.
- Servidor: Controla el IIS.
- Aplicación: Comparte información entre usuarios. Es utilizado para representar la aplicación de web.
- Sesión: Es utilizado para representar la sesión del usuario y guardar información relacionada con esa sesión.
- Object Context: Es utilizado para ejecutar o cancelar transacciones.
- ASPError: Contiene detalles relacionados con cualquier tipo de error.

#### *Funcionamiento de los ASP's.*

Los ASP's son creados con la extensión ".asp". En el momento en que el Internet Information Server (IIS) recibe una petición para cargar una página con la extensión ".asp", el servidor de web accede y compila el script contenido dentro de la página y carga el código compilado en memoria. Después el script lleva a cabo algunos procesamientos de datos, que generalmente generan el código HTML que se escribe sobre el ASP. Para poder llevar a cabo estos procedimientos, el script tiene que acceder a los objetos built-in de ASP, que contienen información considerable sobre el ambiente ASP y que le permiten al script comunicarse nuevamente con su ambiente. El código estático de HTML que se generó a partir del ASP es regresado al cliente utilizando una transacción HTTP común. Para el usuario, la página luce igual que cualquier página html, excepto por la extensión ".asp". Esto se debe a que todo el procesamiento del script tuvo lugar en el servidor.

El IIS es capaz de manejar scripts cuando un ASP recibe información a través de la petición para cargar una página web. El servidor de web examina que la extensión del archivo sea ".asp". Posteriormente el host del script ASP se presenta en el servidor de web en la forma del archivo asp.dll. Cuando se tiene el archivo asp.dll se puede asegurar que todo el código ASP es interpretado.



### *Portabilidad.*

ASP es una tecnología basada totalmente en Microsoft, es decir que sólo funciona bajo plataformas de Microsoft. ASP es parte del Internet Information Server (IIS) desde la versión 3.0. El IIS está en el Windows 2000 o NT, en cambio, para los que tiene Windows 95/98 deberán limitarse a usar el PWS (Personal Web Server).

La tecnología ASP puede correr en otras plataformas distintas a las de Microsoft, pero solamente con el uso de software auxiliar como el Chili!ASP, un producto de la compañía Sun Chili!Soft.

### *Bases de datos con las que interactúan los ASP's:*

Los ASP's trabajan con cualquier base de datos compatible con las tecnologías OLEDB u ODBC.

### **3.6.3 JSP.**

Java Server Pages (JSP) es una tecnología cuya función es controlar el contenido o la apariencia de las páginas Web a través del uso de servlets, que son pequeños programas específicos dentro de la página web que se ejecutan en el servidor de web para modificar la página antes de que sea enviada al usuario que la solicitó. Sun Microsystems, el desarrollador de Java, también se refiere a la tecnología JSP como la interfaz del programa de aplicación de los servlets (servlet's API). Un JSP llama a un programa Java que es ejecutado por el servidor de web, a diferencia de un ASP, el cual contiene un script que es interpretado por un traductor de scripts (como VBScript o Jscript) antes de que la página sea enviada al usuario.

La tecnología JSP utiliza etiquetas de tipo XML y scriptlets escritos en el lenguaje de programación Java para encapsular la lógica que genera el contenido de una página. Adicionalmente, la lógica de la aplicación puede residir en fuentes del lado del servidor a los que la página tiene acceso con estas etiquetas y scriptlets.

Los scriptlets son pequeños scripts que pueden incluirse en páginas JSP. Un scriptlet es un fragmento de código, que se ejecuta al tiempo de procesamiento de una petición. Los scriptlets pueden combinarse con elementos estáticos de la página para crear una página generada en forma dinámica.

La tecnología JSP es una extensión de la tecnología de servlets de Java. Los servlets son independientes de la plataforma, son módulos del lado del servidor de Java que encajan en el ambiente del servidor de web y pueden ser utilizados para extender las capacidades del servidor de web con sobrecarga, mantenimiento y soporte mínimos.

### *Funcionamiento de la tecnología JSP.*

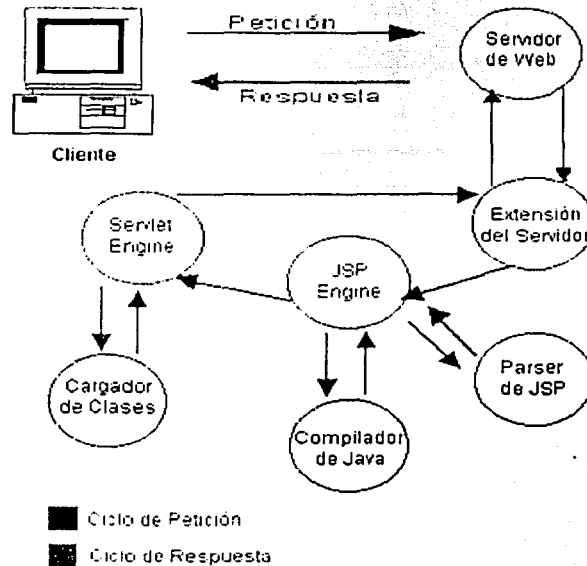
El código fuente de una página JSP es esencialmente HTML (texto o incluso XML), que contiene etiquetas JSP y/o código de Java dentro de éstas. La extensión ".jsp" de este tipo de documentos le indica al servidor que se requiere de un manejo especial de la página.

Este manejo especial que realiza el servidor de web, incluye los 4 pasos que se describen a continuación:

1. El JSP engine (que se instala en el servidor de Web) analiza y separa los componentes de la página y crea un archivo de código fuente de Java. El JSP engine es el que recibe las peticiones de un cliente a la página JSP y genera las respuestas de la página al cliente.
2. El JSP engine compila el archivo creado en el paso 1 y se crea un archivo ".class" de Java. El archivo ".class" creado en el paso 2 es un servlet, y a partir de este momento el servlet engine se hace cargo del archivo ".class" en la misma forma que maneja cualquier otro servlet.
3. El servlet engine carga la clase Servlet generada en el paso 2 para la ejecución.
4. El Servlet se ejecuta y regresa los resultados al cliente que los solicitó.

Aunque puede parecer que el proceso descrito arriba consume mucho tiempo y recursos, es mucho más eficiente de lo que aparenta. Los pasos 1 y 2 ocurren sólo una vez, cuando se despliega por primera vez el JSP o cuando se actualiza. El servlet engine ejecuta el paso 3 solamente sobre la primera petición de ese servlet hasta que el último servlet se reinicie. Después de esto el cargador de clases, carga la clase correspondiente sólo una vez y la deja disponible durante todo el ciclo de vida del actual JVM (Java Virtual Machine). Finalmente, algunos servidores de aplicación ofrecen captura de páginas, lo cual puede mejorar el desempeño y reducir el tiempo de ejecución de la petición. Con la captura de páginas, incluso el paso 4 puede ejecutarse sólo una vez dependiendo de qué tan dinámicos son los datos de la página.

La siguiente figura esquematiza el proceso anterior.



### *La arquitectura MVC (Model-View-Controller) o la arquitectura Modelo 2.*

JSP sigue el modelo de separar la lógica de programación del diseño de la página a través del uso de componentes como los JavaBeans, Enterprise JavaBeans (EJB) y las etiquetas JSP.

Al incrementarse la complejidad, las páginas se sobrecargarían con scriptlets y código de procesamiento, haciendo difícil depurar y hacer cambios en los datos. Para solucionar este problema se introdujo la arquitectura MVC para Servlets/JSP. Esta arquitectura es un alcance híbrido que sirve al contenido dinámico de las páginas y combina el uso de Servlets y JSP. Se toma como ventaja la fuerza de ambas tecnologías: Los JSP's son usados para generar la capa de presentación y los Servlets para ejecutar las tareas de procesamiento intensivo. El Servlet actúa como el controlador que procesa las peticiones, creando "beans" (modelos) u objetos utilizados por el JSP, y dependiendo de las acciones de usuario, se le envía a éste la página JSP adecuada. La página JSP (el visualizador), no contiene la lógica de procesamiento, es simplemente la responsable de extraer e insertar el contenido dinámico proveniente del Servlet dentro de los templates estáticos.

### *Portabilidad.*

Los JSP's son independientes de la plataforma, pueden correr sobre cualquier sistema operativo incluyendo UNIX, Microsoft Windows, Linux, Solaris y Mac OS. Pueden trabajar sobre cualquier servidor de Web incluyendo Apache, Netscape y IIS.

### *Bases de datos con las que interactúa.*

Los JSP's trabajan con cualquier base de datos compatible con las tecnologías ODBC y JDBC.

## **3.7 Infraestructura del Instituto de Ingeniería.**

### **3.7.1 Red Local del Instituto de Ingeniería.**

El Instituto de Ingeniería cuenta con una red local de cómputo que proporciona un enlace a RedUNAM. Esta red tiene topología de estrella con el backbone colapsado y con troncales de fibra óptica multimodo, con ello se garantiza un crecimiento gradual del ancho de banda de la red para adaptarse a los requerimientos futuros de las nuevas aplicaciones que ofrecerá Internet 2, como son ambientes virtuales, videoconferencia, telepresencia, visualización científica, procesamiento distribuido, paralelo, entre otras. La red local de cómputo ha crecido notablemente. Actualmente está integrada por más de 1000 equipos entre servidores UNIX, NT, PCs, periféricos, 25 equipos de telecomunicaciones y 700 puertos de red distribuidos en 10 edificios interconectos a través de un red de fibra óptica y utilizando tecnologías ATM, FDDI, CDDI, Fast Ethernet y Ethernet a una velocidad máxima en el backbone de 155 Mbps. Así mismo actualmente se trabaja también en el diseño de la red de cómputo de la Torre de Ingeniería sobre la base de conceptos de sistemas de cableado estructurado para edificios inteligentes con fibra óptica, tecnologías Fast Ethernet/Gigabit-Ethernet, redes virtuales, videoconferencia, energía no interrumpida y monitoreo remoto. El laboratorio de planeación y operación de la Red-II es el encargado de diseñar, desarrollar, planear y mantener en operación continua la red local de cómputo del Instituto y su enlace a RedUNAM.

### **3.7.2 Servidores.**

El laboratorio de sistemas operativos distribuidos perteneciente a la coordinación de Sistemas de Cómputo del Instituto, cuenta con dos áreas: en la primera se diseña, instala, administra y mantiene en operación 15 servidores secundarios de aplicación Windows NT y tres servidores centrales: Altair (principal de dominio), Imsai (de respaldo) y Antares (de acceso remoto RAS), estos equipos dan apoyo a 438 usuarios y a 10 servidores de impresión remota. La segunda área diseña, instala, administra y mantiene en operación 66 sistemas UNIX de los cuales dos son los servidores principales de Internet: Pumas y Tonatiuh, con 700 claves de usuario para e-mail, telnet con shell

UNIX, web, FTP, bases de datos SQL, compiladores y diversas aplicaciones, además se cuenta con 85 claves de acceso remoto telefónico vía módem. (Red-II en su casa.)

### **3.7.3 Manejadores de Bases de Datos.**

En la coordinación de Sistemas de Cómputo del Instituto de Ingeniería se utilizan los siguientes manejadores de bases de datos:

- Sybase 11.01, para la Base de Datos Académica del Instituto de Ingeniería (BDAII).
- Microsoft SQL Server, para aplicaciones de web en proceso de desarrollo.

### **3.7.4 Servidores de web.**

El servidor de web utilizado para el sitio del Instituto de Ingeniería es el Apache Web Server versión 1.3.24 el cual funciona sobre una plataforma SPARC de Sun. En forma secundaria, el Microsoft Internet Information Server 5.0 es utilizado internamente por la Coordinación de Sistemas de Cómputo.

# CAPÍTULO CUATRO

## Elementos y Herramientas de Desarrollo del Sistema

## **4. Elementos y herramientas de desarrollo del Sistema.**

En los capítulos anteriores analizamos la importancia del Sistema de Control de Becas (SCBII), así como las tecnologías disponibles en el mercado para su desarrollo y los recursos de cómputo con los que cuenta el Instituto. Es importante mencionar que estos aspectos, además del presupuesto del Instituto, constituyen los parámetros esenciales sobre los cuales se propone una solución.

### **4.1 Elección de los elementos y herramientas de desarrollo.**

Durante la etapa de diseño de un sistema de información es necesario elegir el servidor de web, el manejador de bases de datos, las herramientas de programación que se van a utilizar y el sistema operativo sobre el cual residirán estos componentes en forma independiente. La elección no es simple, debido a que cada uno de estos elementos está relacionado con los demás y a que no todas las herramientas de programación están disponibles para Unix y NT, asimismo ocurre para los manejadores de bases de datos y los servidores de web.

#### **4.1.1 Manejador de Bases de Datos.**

Para elegir el manejador de bases de datos (DBMS) adecuado, se tienen dos opciones: DBMS de distribución libre y DBMS comerciales.

En cuanto a los DBMS de distribución libre, los más conocidos son PostgreSQL y MySQL. A pesar de tener un buen desempeño, no existen estadísticas confiables que nos permitan saber quienes son sus usuarios y para que tipo de sistemas se han utilizado. Además se cuenta con la desventaja de que no existe una empresa que se haga cargo del soporte técnico o que se responsabilice por "bugs" en el DBMS. PostgreSQL es completamente gratuito, mientras que MySQL requiere de un pago mínimo por cada equipo en donde se instale.

Por su parte, los DBMS comerciales ofrecen la ventaja de ofrecer soporte técnico confiable, sin embargo el costo de las licencias de cada uno de ellos es elevado.

Todos los DBMS mencionados anteriormente siguen el estándar ANSI SQL92, cumpliéndolo total o parcialmente. Por ejemplo, los DBMS comerciales como Oracle 8, Sybase 11 y MS SQL Server 2000 cumplen con el estándar completamente, mientras que los de distribución libre como MySQL y PostgreSQL 7 omiten algunas funcionalidades. Actualmente los DBMS, tratan de cumplir con un nuevo estándar, el ANSI SQL99, el cual ya es soportando por Oracle, Sybase y Microsoft SQL Server.

Debido a que los DBMS de distribución libre cumplen parcialmente con el estándar de SQL y a la falta de soporte técnico, nuestra selección de DBMS se inclina hacia uno de tipo comercial.

Tal y como se mencionó en el capítulo anterior, la coordinación de Sistemas de Cómputo cuenta con las licencias de uso del DBMS Microsoft SQL Server 2000 y Sybase 11.01. Por esta razón, nuestra selección se restringe hacia las opciones mencionadas.

Debido a los problemas que ha tenido últimamente la representación de Sybase en México para dar soporte a todos sus clientes, nuestra decisión final es la de utilizar a Microsoft SQL Server 2000 como el DBMS del Sistema de Becas, debido a la confiabilidad y disponibilidad de soporte técnico que éste requiere.

Aunque el manejador seleccionado requiere al sistema operativo Microsoft Windows, se puede obtener en forma gratuita a través de Internet, un driver que permite la interfaz entre un sistema operativo UNIX y Microsoft SQL Server<sup>1</sup>.

#### **4.1.2 Servidor de web.**

Debido a que el desarrollo del SCBII requiere economizar y hacer uso de los recursos disponibles, es necesario elegir un servidor de web con el que cuente la coordinación de Sistemas de Cómputo o bien, un servidor de distribución libre. Asimismo, es muy importante que el servidor de web elegido, sea capaz de soportar distintas tecnologías de programación.

En el capítulo anterior, se mostraron las estadísticas de los servidores de web más utilizados. El grupo de servidores de Apache y el de Microsoft resultaron ser los preferidos de los sitios web encuestados, el primero con el 56.87% y el segundo con el 30.25%. El resto de los servidores comparten menos del 13%, razón por la cual nuestra selección se inclina hacia los dos más populares.

Dentro de los grupos de servidores elegidos, los más conocidos de cada uno de ellos son Apache Web Server y Microsoft IIS, respectivamente. Ambos permiten utilizar ciertas herramientas de programación, con la finalidad de generar contenido dinámico en las páginas de Internet y realizar conexiones a bases de datos, para manejar información e incluirla dentro del contenido de las páginas web. Estas herramientas incluyen Servlets, JSP's, ASP's PHP, Perl entre otros.

El servidor IIS de Microsoft es gratuito en la compra del paquete de Windows NT/2000. En cuanto a las tecnologías de programación soportadas, el API de ASP's viene incluido con el servidor de Microsoft, pero para el resto de

---

<sup>1</sup> Consulte: [http://www.atinet.com/Products/OpenRDA/ODBC\\_UNIX/ODBC\\_UNIX\\_sql.asp](http://www.atinet.com/Products/OpenRDA/ODBC_UNIX/ODBC_UNIX_sql.asp)



las herramientas, tales como PHP, JSP y Java Servlets, es necesario utilizar software de terceros, que no siempre puede adquirirse en forma gratuita. Este servidor es totalmente dependiente de plataformas de Microsoft.

En cuanto al servidor Apache, éste puede adquirirse en forma gratuita a través del sitio web de Apache. De la misma forma se pueden obtener los módulos para soportar las tecnologías Perl, PHP, JSP y Java Servlets, ya que esta organización ha desarrollado un proyecto para cada una de ellas. Para la tecnología ASP se requiere instalar y configurar adecuadamente el módulo de Perl<sup>2</sup>. El servidor Apache puede ser instalado sobre las tecnologías Microsoft y UNIX en sus diferentes tipos; sin embargo el servidor de web Apache ha sido mayormente probado sobre plataformas UNIX.

En el aspecto referente al soporte técnico que incluyen estos productos, el servidor de web de Microsoft cuenta con soporte gratuito por parte de su fabricante. Para el caso del servidor de web Apache, el soporte no es directo y se reduce a foros de discusión a través de Internet.

Aunque en la coordinación de Sistemas de Cómputo, se cuenta con Microsoft IIS, la página principal del Instituto utiliza a Apache como servidor de web, razón por la cual se tiene mayor experiencia en el uso de éste. Asimismo, Apache Web Server no necesita de software de terceros para poder soportar a la mayoría de las tecnologías de programación. Por los motivos expuestos, incluyendo el hecho de que Apache Web Server es multiplataforma, nuestra elección final se inclina hacia este servidor.

#### **4.1.3 Herramientas de Programación.**

En el capítulo anterior se realizó una breve descripción sobre tres herramientas de programación de tecnología web: ASP, PHP y JSP. A continuación se describen las ventajas y desventajas de cada una de las alternativas mencionadas, se comparan y finalmente se escoge una de ellas como herramienta de desarrollo del SCBII.

##### **4.1.3.1 Tecnología PHP.**

*Ventajas.*

- Fácil de aprender, debido a que su código es legible.
- Es orientado a tecnología Web y soporta múltiples protocolos de Internet.
- Es multiplataforma y soportado por los servidores de web más comunes.
- Ofrece el manejo de clases de Java y objetos COM.

---

<sup>2</sup> Consulte: <http://freshmeat.net/projects/apacheasp/>

- Soporta múltiples DBMS.
- El costo es nulo.

*Desventajas.*

- La escalabilidad es dependiente de terceros.
- No es recomendado para aplicaciones grandes.
- Por ser de distribución libre, no existe un organismo que se responsabilice por fallas o que promueva actualizaciones.

**4.1.3.2 Tecnología ASP.**

*Ventajas.*

- Es fácil de aprender.
- La sintaxis del código ASP es relativamente sencilla lo cual facilita su mantenimiento.
- El servidor de web procesa el script de ASP y envía al cliente únicamente código HTML, que entonces es interpretado por el navegador. Esto facilita la seguridad del código ASP.
- ASP utiliza los componentes de servidor ActiveX que corren en el servidor local (u otro servidor en la red) para tener acceso a bases de datos y a aplicaciones para procesar la información. Permite utilizar componentes en cualquier lenguaje, incluyendo Visual Basic, C++, COBOL y Java.
- El servidor de web que utiliza ASP, el IIS, contiene un conjunto de componentes, incluyendo los componentes ActiveX Data Objects (ADO), que ofrecen un acceso sencillo a fuentes de datos compatibles con cualquier driver OLE/DB u ODBC.
- ASP soporta la sintaxis de VBScript o de JScript desde la instalación inicial. Pueden añadirse otros lenguajes como Perlscript.
- ASP viene incluido en los sistemas operativos Windows NT/2000 sin costo adicional.

*Desventajas.*

- Solamente corre en plataformas de Microsoft:
  - a) Para Windows NT y Windows 2000 se requiere el uso del IIS (Internet Information Server)

b) Para Windows 95/98 se requiere el uso del PWS (Personal Web Server)

- Necesita de productos de otras empresas distintas a Microsoft para ser multiplataforma.
- Limitado a utilizar sus componentes para aplicaciones específicas. ASP está diseñado para aplicaciones pequeñas y sencillas. No permite construir aplicaciones complejas y modularizables.
- Por cada petición que se le hace a una página ASP, ésta tiene que ser interpretada.
- No es tolerante a fallas del sistema.

#### **4.1.3.3 Tecnología JSP.**

##### *Ventajas.*

- Es independiente de la plataforma (sistema operativo y servidor de web).
- Se basa en el lenguaje Java que proporciona un desempeño superior al que tienen los lenguajes interpretados como VBScript o Jscript. Asimismo el lenguaje Java ofrece seguridad contra fallas del sistema.
- Cuando una página JSP es llamada por primera vez, ésta todavía no existe, se compila dentro de la clase Java Servlet y se guarda en la memoria del servidor. Esto permite respuestas rápidas para llamadas subsecuentes a la página y soluciona el problema de crear un nuevo proceso para cada petición.
- Debido a que los JSP's son una extensión de la tecnología de Servlets de Java, la escalabilidad de los JSP's es un resultado directo de los Servlets. Por lo tanto, los JSP's permiten la creación de aplicaciones, grandes, complejas y escalables.
- La tecnología JSP hace uso total del lenguaje Java. Esto hace a JSP un lenguaje muy poderoso para la creación de aplicaciones Web. Java ofrece las características asociadas con un lenguaje maduro, estable y bien desarrollado, tales como herencia, manejo de excepciones y una total orientación a objetos. Al ser JSP un subconjunto de Java, al evolucionar este último en forma paralela lo harán los JSP's.
- Por ser una extensión del lenguaje de programación Java, los JSP's permiten desarrollar aplicaciones estables, aun cuando la demanda de usuarios sea crítica. Asimismo permiten desarrollar aplicaciones en forma modular.

**Desventajas.**

- El desarrollo de JSP's requiere de conocimientos previos de Java y Java Servlets.
- Los JSP's son difíciles de mantener, debido a que una sola sentencia de su código tiende a agrupar varias instrucciones en una y a hacer uso de clases de Java, incluyendo Servlets. El mantenimiento de los JSP's requieren de personal que tenga conocimientos de Java.
- La tecnología JSP requiere que se instale una Máquina Virtual de Java en el servidor de aplicación, así como un contenedor de JSP's.

**4.1.3.4 Comparación entre las tecnologías PHP, ASP y JSP.**

A continuación se muestra una tabla comparativa de las características fundamentales de las tecnologías PHP (distribución libre), ASP (Microsoft) y JSP (Sun Microsystems).

	<b>PHP</b>	<b>ASP</b>	<b>JSP</b>
<b>Facilidad de desarrollo</b>	Es un lenguaje relativamente simple y fácil de aprender	Se basa en Jscript y VBScript	Es esencial tener conocimientos de Java y Java Servlets.
<b>Escalabilidad</b>	La escalabilidad de PHP depende del servidor de web.	Es buena para aplicaciones pequeñas que sólo sirven para un fin específico.	Excelente debido a que es un resultado directo de la escalabilidad de los Servlets.
<b>Características del Lenguaje</b>	Tiene varias funciones estándar útiles, para cada aspecto de la programación web.  (PHP4)	Cuenta con bastantes características de JScript y VBScript que proporcionan objetos útiles y métodos para el manejo de datos.	Puede hacer uso de todas las características del lenguaje.  (Java)
<b>Portabilidad</b>	Sobre plataformas Linux, UNIX, Windows9x/2000, MacOS, OS/2	Windows 95/98/2000 y Windows NT.	UNIX, Microsoft Windows, MacOS, Linux.
<b>Componentes Soportados</b>	COM, clases Java	Componentes COM	Java Beans, Enterprise Java Beans

<b>Servidores de Web Soportados</b>	Apache, iPlanet/Netscape Enterprise Server, MS IIS, Apache, Zeus, etc.	IIS, Personal Web Server (PWS), Apache <sup>3</sup>	Cualquier Servidor de Web, incluyendo Apache, Netscape y IIS <sup>4</sup>
<b>Bases de Datos Soportadas</b>	MySQL, mSQL, ODBC, Oracle, Informix, Sybase:	Cualquier base de datos compatible con ODBC.	Cualquier base de datos compatible con ODBC o JDBC.
<b>Mantenimiento del Código</b>	La estructura de PHP es razonablemente simple y legible.	La sintaxis es relativamente simple y esto facilita su mantenimiento.	Pueden ser difíciles de mantener si no se está familiarizado con la estructura de Java.
<b>Costo</b>	Gratuito.	Gratuito con la compra de WindowsNT/2000.	Gratuito.

#### 4.1.3.5 Herramienta de Programación Seleccionada.

Antes de elegir una herramienta de programación determinada, debemos de recordar que se pretende considerar al SCBII como el módulo de un sistema integral de información administrativa del Instituto de Ingeniería, lo que significa que el SCBII debe de ser flexible a los cambios y las adaptaciones que se lleven a cabo en el futuro. Por lo anterior, la herramienta de programación seleccionada debe de cumplir al menos con las siguientes características:

- Ser independiente de la plataforma
- Ser escalable
- Soportar aplicaciones a las que tengan acceso múltiples usuarios con distintas tareas.

La escalabilidad de la tecnología PHP es dependiente de terceros, además de que se usa en aplicaciones simples que no requieren expansión a futuro. Por su parte, la tecnología ASP es dependiente de plataformas Microsoft y únicamente sirve para aplicaciones con un fin específico. Las ventajas que le brinda tener como base un lenguaje orientado a objetos como Java, hacen de la tecnología JSP la idónea para desarrollar el SCBII y no sólo porque cumple con las características mencionadas anteriormente, sino porque permite organizar el código en tres niveles:

1. *Clases de Java*: Constituyen el esqueleto o estructura básica del sistema.
2. *Servlets*: Permiten manipular la información que el cliente introduce y enviarla a la base de datos, haciendo este proceso invisible para el usuario. Asimismo permiten manipular gestiones de seguridad, que en nuestro caso puede aplicarse para ofrecer distintos niveles de seguridad de acuerdo con la jerarquía del usuario dentro del Instituto de Ingeniería y restringir el acceso al

<sup>3</sup> Requiere de módulos adicionales de proyectos de Apache.

<sup>4</sup> El servidor web Microsoft IIS requiere de componentes de terceros para la ejecución de JavaServer Pages (JSP)

sistema, únicamente a computadoras que pertenezcan a la Intranet del mismo.

3. *JSP's*: Generan la interfaz con el usuario, permiten crear código en forma dinámica y pueden comunicarse directamente con los Servlets y las clases de Java.

En general, Java es una buena alternativa para aquellas instituciones en las cuales se requiere implementar un conjunto de sistemas homogéneos, que representen los esquemas y políticas sobre los cuales se rigen las organizaciones. La naturaleza modular de Java hace posible el crecimiento de cualquier sistema de información.

Habiendo decidido que la tecnología que se va a utilizar es la de JSP, es importante describir y explicar el funcionamiento de los Servlets de Java.

### ***Java Servlets.***

Los Servlets son código de Java que se ejecutan en los servidores de aplicación y aunque no están limitados a manejar peticiones http, se utilizan comúnmente para generar páginas web con contenido dinámico; esto basado en la información de la petición de los clientes y sin las limitaciones de la programación en CGI's. Los Servlets se componen de clases estándar e independientes dentro del API de Java, por lo que tienen acceso a cualquier clase de este lenguaje de programación. Los Servlets se encuentran dentro de un paquete de Java (javax.servlet) y son manejados por lo que se denomina "Contenedor de Servlets".

El Contenedor de Servlets, en conjunción con el servidor de web o de aplicación, provee los servicios de red sobre los cuales se establecen las peticiones y las respuestas, decodifica peticiones y genera respuestas basadas en MIME (Multi-Purpose Internet Mail Extensions). El Contenedor de Servlets también da soporte y manipula los Servlets a través de su ciclo de vida. El Contenedor puede ser nativo del servidor de web o ser instalado como un componente a través del API del mismo servidor, incluso puede ser nativo o agregarse a servidores de aplicación. Existen contenedores que no requieren de un servidor de web y se ejecutan como aplicaciones independientes. Ya sean independientes o en cooperación con algún servidor de web, ambas opciones de instalación necesitan contar con una Máquina Virtual de Java que de soporte a las clases propias del lenguaje. El Contenedor debe soportar el protocolo HTTP para las peticiones y respuestas, incluso puede interactuar con protocolos como el HTTPS (HTTP sobre SSL).

El API Servlet no incluye información acerca de cómo son cargados los Servlets, el ambiente en el cual corren y el protocolo usado para transmitir los datos del usuario. Esto permite a los Servlets poder ser usados por diferentes servidores de web.

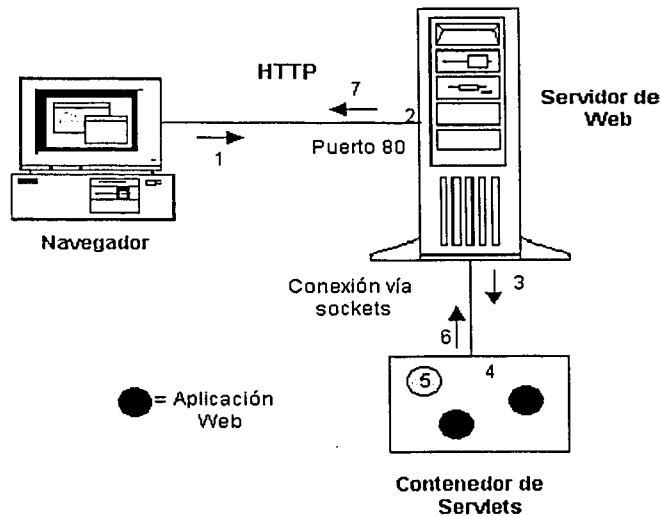
Un Servlet puede comunicarse y ejecutar otros Servlets, incluso ejecutar métodos concretos de éstos. Esto ayuda a distribuir eficientemente el trabajo, incluso entre diferentes servidores.

### *Funcionamiento de los Servlets.*

Una vez que el Servlet ha sido escrito y compilado como cualquier otra clase de Java, es suficiente con que el contenedor de Servlets esté configurado y funcionando para poder generar la respuesta del Servlet. En caso de que el contenedor trabaje cooperativamente con el servidor de web, el proceso de solicitud/respuesta es el siguiente:

- 1) Se emite una petición HTTP desde un navegador hacia el servidor web.
- 2) El servidor web recibe la petición.
- 3) Si se solicita una página que es procesada por algún Servlet, el servidor web (a través de un socket) envía la dirección URL al contenedor. Esto es debido a que el servidor de web está configurado para enviar direcciones URL específicas hacia el contenedor.
- 4) Si el Servlet al cual se hace referencia no se encuentra cargado, se carga en la memoria reservada para su funcionamiento sobre la Máquina Virtual de Java.
- 5) Si el Servlet ya está cargado en memoria, éste le asigna un hilo a la petición, la procesa y genera la respuesta en el formato preestablecido en el Servlet, el cual puede ser HTML.
- 6) El contenedor regresa la respuesta HTML al servidor web.
- 7) El servidor web reenvía la respuesta al navegador que hizo la petición.

La siguiente figura esquematiza el proceso descrito anteriormente.



Los Servlets pueden ser permanentes o temporales. Los permanentes se inicializan al levantar el demonio del servidor de web y permanecen hasta que este demonio se detiene. Los temporales se cargan en demanda. Ambos se pueden acceder indistintamente de manera local o remota.

Cuando un Servlet es llamado desde un cliente, éste recibe dos objetos: ServletRequest y ServletResponse. La interfaz ServletRequest se encarga de la comunicación desde el cliente al servidor, mientras que la interfaz ServletResponse atiende la comunicación desde Servlet al cliente.

La interfaz ServletRequest permite al Servlet acceder a cierto tipo de información incluyendo los nombres de parámetros pasados por el cliente, el protocolo usado por el cliente, los nombres de las computadoras cliente remotas que hacen la solicitud y el servidor que la recibe. Esta interfaz les facilita a los Servlets el acceso a métodos que permiten manejar la presentación de la respuesta como salida en el navegador, a través de los cuales consiguen los datos desde el cliente que usa protocolos como HTTP POST, etc.

La interfaz ServletResponse proporciona al Servlet los métodos para contestarle al cliente. Permite al Servlet configurar la forma de salida de los datos para el cliente. Las subclases de ServletResponse le dan más capacidad al Servlet para responder. Estas clases e interfaces descritas conforman a un Servlet básico. Pero existen métodos adicionales que provee el API con la capacidad para controlar sesiones o múltiples conexiones, entre muchas más aplicaciones.

#### *Usos más comunes de los Servlets.*

- Procesar y/o salvar la información generada por una forma HTML.
- Proveer contenido dinámico a páginas web, actuando como un enlace entre el cliente y una o varias bases de datos en una arquitectura cliente servidor de tres capas.
- Realizar tareas de proxy para un applet. Debido a las restricciones de seguridad de los applets éstos no pueden acceder directamente a un servidor de bases de datos de computadoras remotas.
- Control de recepción de e-mails, sistemas de news, chats, archivos, etc.
- Abrir sockets entre el cliente y una tercer computadora.
- La escritura/lectura de cookies y el rastreo de sesiones, todo esto para mantener y guardar información durante la sesión de un cliente.



## **4.2 Hardware y Arquitectura de Desarrollo.**

Para la etapa de desarrollo del sistema, la coordinación de Sistemas de Cómputo nos proporcionó dos computadoras personales con las siguientes características:

- Memoria RAM: 128 KB.
- Procesador: Intel Pentium 3, a 800 MHz.
- Sistema Operativo: Windows NT.
- Acceso a Internet: A través de la red del Instituto a una velocidad compartida de 10 Mbps.

Durante esta fase, el sistema fue distribuido en una arquitectura cliente/servidor de tres capas lógicas y físicas. Las características de estas capas se describen a continuación:

- a) Cliente :Es cualquier computadora con acceso a la intranet del Instituto y con los siguientes requisitos mínimos de software:
  - Sistema Operativo: Windows 95/98/2000/NT/\* o UNIX en sus diferentes versiones.
  - Explorador de Web: Microsoft Explorer versión 4 en adelante.
- b) Servidor de web: Utiliza Apache Tomcat 4.0, alojado en la PC con dirección IP 132.248.156.90. Para comunicarse con el servidor de bases de datos, el servidor de web utiliza el SQL Server 2000 JDBC Driver Beta 1.
- c) Servidor de bases de datos: Utiliza como manejador a Microsoft SQL Server 2000 y se encuentra alojado en la PC con dirección IP 132.248.156.246.

## **4.3 Propuesta de despliegue de los componentes del sistema para su operación.**

En este apartado se presenta una propuesta sobre la arquitectura en donde residirá el sistema, después de que éste haya superado la etapa de desarrollo y se decida iniciar su fase de utilización dentro del Instituto.

### **4.3.1 Desarrollo de la Propuesta.**

Durante el diseño de un sistema es necesario decidir el esquema de la arquitectura cliente/servidor a emplear. La elección del número de capas de la arquitectura debe de tomar en cuenta el número de usuarios que van a utilizar el sistema, la velocidad esperada de las transacciones entre el cliente y la base de datos y la escalabilidad que el sistema requiere a largo plazo.

Actualmente, diversos sistemas de información utilizan las arquitecturas cliente/servidor que operan en sistemas de dos capas, tres capas o n capas.

Estas capas describen la partición lógica o física de la capa de escritorio (cliente), de la de aplicación y de la de información (manejadores de bases de datos) para establecer la comunicación entre un servidor y un cliente. El componente de escritorio proporciona el despliegue y la interacción del usuario con la aplicación. El componente de aplicación lleva a cabo las peticiones entre el cliente y la base de datos. La base de datos almacena la información a partir de los procesos de aplicación

La arquitectura que se propone para el Sistema de Control de Becas es una arquitectura cliente/servidor de tres capas (a nivel del hardware y a nivel de software).

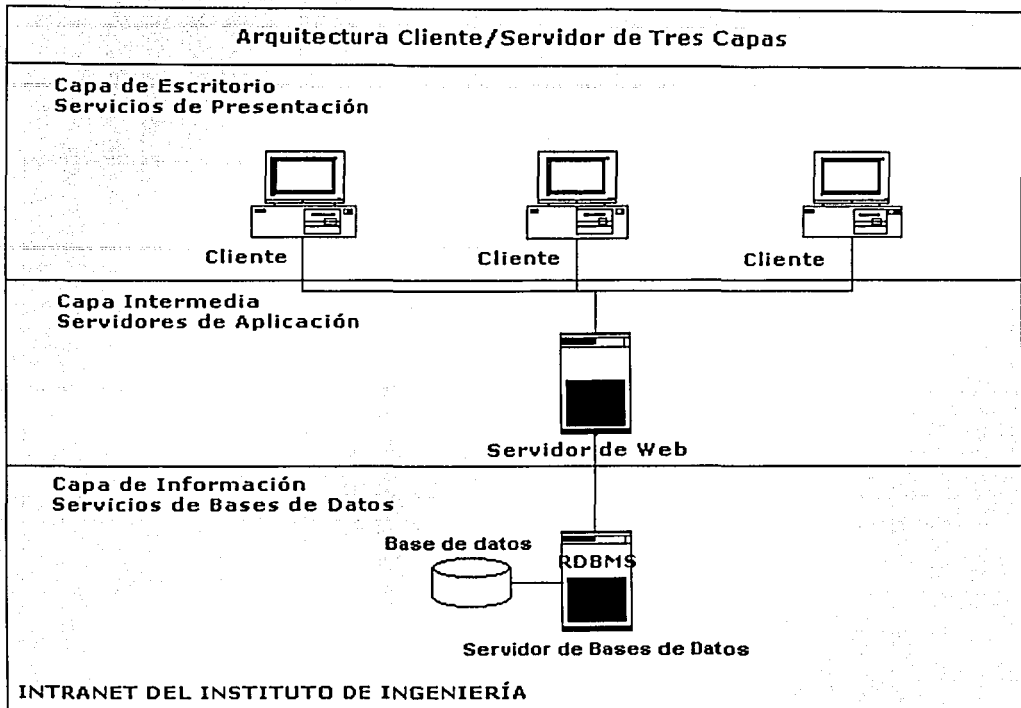
La decisión de utilizar esta arquitectura en lugar de una arquitectura de dos capas, se debe a que esta última puede ser más demandante en la infraestructura de la red debido a la necesidad del cliente para descargar datos directamente de la base de datos, resultando esto en requerimientos de ancho de banda más altos.

La simplicidad y la facilidad para el desarrollo de pequeños sistemas, es la ventaja más grande de un modelo cliente/servidor de dos capas, que se ajusta bien para aplicaciones departamentales. De cualquier forma, los sistemas críticos de negocios que abarcan toda una organización o institución como es nuestro caso, necesitan una escalabilidad superior a dos capas.

La arquitectura cliente/servidor de tres capas reduce el tráfico de información en la red por lo que mejora el rendimiento de los sistemas (especialmente respecto de la estructura en dos planos). Asimismo, brinda una mayor flexibilidad en el desarrollo y la elección de plataformas sobre la cual montar las aplicaciones y provee escalabilidad horizontal y vertical.

La arquitectura de tres capas da soporte a una mejor integración entre aplicaciones y permite mantener la independencia entre el código de la aplicación (reglas y conocimiento del negocio) y los datos almacenados, mejorando la portabilidad de las aplicaciones.

La distribución de la arquitectura propuesta para el SCBII se esquematiza en la siguiente figura:



En la figura anterior se pueden observar tres capas:

1. Capa de escritorio: Corresponde a las computadoras cliente, en nuestro caso, se trata de la interfaz del usuario desde la cual los becarios y las autoridades del Instituto tendrán acceso al Sistema de Control de Becas.
2. Capa intermedia: Corresponde a los servicios de aplicación, en esta parte se pretende alojar al servidor de web.
3. Capa de información: Corresponde a los servicios de datos, en este nivel se monta el servidor de bases de datos.

En este tipo de arquitectura, el servidor de web cumple la función de puente entre la parte cliente y el servidor de bases de datos que posee la información.

En el lado del cliente tenemos una máquina en la que un navegador ejecuta un JSP que muestra un formulario o despliega alguna búsqueda. En la segunda capa tenemos una máquina servidora, en la que un software servidor de web, recibe las peticiones de los clientes, y desencadena la ejecución del Servlet

generado a partir del JSP. Este Servlet trabajara con una base de datos alojada en otro servidor dedicado exclusivamente a esa función.

# CAPÍTULO CINCO

## Análisis, Diseño e Implementación del Sistema

## **5. Análisis, Diseño e Implementación del Sistema.**

### **5.1 Recopilación de Requerimientos.**

Teniendo como objetivo la recopilación de los requerimientos generales para el SCBII, se hizo una propuesta de éstos en cooperación con el jefe de la coordinación de Sistemas de Cómputo del Instituto, la cual fue presentada al Secretario Académico y al subdirector del área de Electromecánica y Cómputo para su aprobación. Los funcionarios mencionados, modificaron y complementaron la propuesta con distintas observaciones, comentarios y sugerencias.

En entrevistas posteriores con el Secretario Académico y el coordinador de Sistemas de Cómputo se establecieron los requerimientos generales del SCBII, mismos que se detallan en este capítulo.

#### **5.1.1 Requerimientos del Sistema de Control de Becas.**

En todos los puntos establecidos por los requerimientos que se describen a continuación, se respeta el Reglamento Interno del Instituto de Ingeniería.<sup>1</sup>

#### **Sobre el PBII.**

##### **Generales.**

- a) En todo aspecto relacionado con los requerimientos de suscripción, duración, término y renovación de una beca se debe de respetar el Reglamento General de Becas del Instituto de Ingeniería.<sup>2</sup>
- b) Se debe de considerar el registro de un candidato a becario en el sistema, así como contemplar su baja del PBII. En este último caso se deben de conservar los datos del becario en registros históricos. Esto debido a que el anterior sistema no contemplaba tales funcionalidades.
- c) Se deben de considerar dos tipos de solicitud de beca, ya sea renovación o de nuevo ingreso. Asimismo, el sistema debe ser capaz de cancelar una beca.
- d) Los datos académicos del becario sólo pueden ser cambiados hasta el final del periodo de la beca. El horario de trabajo del becario, así como sus datos personales pueden cambiarse en cualquier momento siempre y cuando se tenga la autorización del tutor.

---

<sup>1</sup> Consultar Anexo A.

<sup>2</sup> Consultar Anexo B.

**Responsabilidades del becario.**

a) Introducir y actualizar la siguiente información:

Datos Personales.

- CURP (opcional y actualizable).
- Domicilio (actualizable).
- Teléfono (opcional y actualizable).
- País de procedencia.
- Estado Civil (actualizable).
- Correo Electrónico (actualizable).

Datos Académicos.

- Matrícula de la Universidad a la que pertenece (opcional y actualizable).
- Institución donde realizó o realiza la Licenciatura y/o Maestría y/o Doctorado (los campos de nivel maestría y doctorado son opcionales y actualizables).
- Nombre completo de la licenciatura, maestría o doctorado que cursa actualmente o que haya terminado.
- El promedio actual y/o terminal por cada nivel académico en una escala de 0 a 100 (actualizable).
- El porcentaje de avance de créditos por nivel académico en una escala de 0 a 110 (actualizable).
- El estado actual de cada uno de los grados registrados (carrera en curso, pasante o grado obtenido).
- En su caso la fecha del examen profesional de cada grado académico obtenido (opcional y actualizable)
- En caso de tener una beca externa al PBII, especificar el monto y la institución (opcional).

Información adicional requerida por el PBII.

- En caso de primer ingreso, se requiere incluir un plan general de trabajo del becario.
- Tutor responsable de la beca.
- Plan de trabajo para el periodo de vigencia de la beca.
- En caso de renovación, se debe de incluir el informe de actividades correspondiente a la beca más reciente.
- En la parte de plan de trabajo, el estudiante también debe de especificar el nombre del proyecto en el que trabajará. Esta información será incluida en el registro histórico del becario.
- Curriculum (actualizable).
- Historia Académica (actualizable).
- Horario de Trabajo (actualizable).

- b) Solicitar a su tutor la clave y la contraseña de acceso al sistema, manteniendo la confidencialidad de estos datos.
- c) El becario puede solicitar el cambio de tutor después de un semestre y por una sola vez. Es decir, que en caso de que el becario tome la decisión de trabajar con otro tutor diferente al actual, se realizará la transferencia correspondiente. Es importante hacer notar que a pesar de que para el nuevo tutor es la primer vez que se inscribe al becario, el PBII consideraría esta inscripción como una renovación de beca.

### ***Responsabilidades del Tutor.***

- a) Verificar que el candidato a becario cumpla con lo estipulado por el Reglamento General de Becas, específicamente en lo que se refiere a requisitos (capítulo 5), duración (capítulo 4) y terminación (capítulo 5) de una beca.
- b) Verificar que cuente con suficientes recursos en su fondo de becas.
- c) Dar de alta a un aspirante a becario en la base de datos del sistema, introduciendo el nombre completo y la fecha de nacimiento del candidato a becario. El sistema es responsable de generar su clave y contraseña.
- d) Consultar la contraseña y clave de cada uno de sus becarios.
- e) Registrar los siguientes datos en el proceso de aprobación de una beca:
  - La vigencia de la beca.
  - El nombre y número de proyecto al que estará asignado el estudiante al menos durante el principio del periodo de la beca.
  - El nivel de la beca en caso de ser regular (condicional).
  - El monto de una beca especial, así como la justificación de la misma (condicional).
  - La institución o instituciones que financian becas externas así como los montos correspondientes (opcional).
- f) No debe de asignar una beca a un estudiante que no ha llenado su plan de trabajo.
- g) En caso de nuevo ingreso, el tutor, además de evaluar los antecedentes académicos del candidato a becario, deberá de estudiar su plan de trabajo para el próximo semestre. Una vez terminada la evaluación, el tutor tomará la decisión de aprobar o rechazar la solicitud de beca.
- h) En caso de renovación, el tutor además de evaluar los datos académicos y personales del candidato a becario, deberá de analizar el informe de actividades correspondiente al periodo de la beca más reciente.



- i) Modificar, si es necesario, los planes de trabajo e informes de sus becarios durante el periodo de aprobación de becas.
- j) En caso de que una solicitud de beca sea rechazada por algún funcionario (coordinador, subdirector, director) el tutor es el responsable de analizar la razón del rechazo y en su caso, comunicar el resultado del análisis al candidato a becario para que conjuntamente hagan las correcciones pertinentes.
- k) Dar de baja del sistema a un aspirante a becario, en caso de que la solicitud de beca haya sido rechazada en forma irrevocable.
- l) Dar de baja en el sistema a un becario del programa, siempre y cuando especifique la razón de la baja, la fecha, los logros alcanzados durante su estancia, nivel académico al ingresar y al salir del PBII, y la información referente a sus actividades futuras. Esto se notificará a los funcionarios correspondientes.

#### ***Responsabilidades del Coordinador.***

- a) Verificar que el becario cumpla con los requisitos establecidos en los capítulos 4,5 y 8 del Reglamento General de Becas.
- b) Evaluar los antecedentes académicos del becario así como su programa de actividades. En caso de renovación, también debe de evaluar el informe de actividades del semestre anterior.
- c) Verificar si existe congruencia entre el programa de actividades del becario, en su caso el informe del semestre anterior y el trabajo que realiza el tutor.
- d) Verificar el monto y el nivel de la beca asignada al becario.
- e) Tomar la decisión de aprobación o rechazo de la solicitud de beca. En caso de rechazo se debe de comunicar la razón de la decisión al tutor. En el caso de aprobación, puede anexar algún comentario a la beca.
- f) Verificar cuánto dinero gasta en becas cada uno de los tutores dentro de su coordinación.
- g) Verificar cuántos becarios tiene cada uno de sus tutores.
- h) Verificar cuántos becarios de su coordinación han obtenido el grado que les corresponde.

#### ***Responsabilidades del Subdirector.***

- a) Verificar el monto y el nivel de la beca asignada a cada becario.

- b) Revisar ocasionalmente algún plan o reporte de actividades.
- c) Tomar la decisión de aprobación o rechazo de la solicitud de beca. En caso de rechazo se debe de especificar la razón de la decisión. En el caso de aprobación, puede anexar algún comentario a la beca.
- d) Analizar el avance académico que han tenido los becarios del programa en cada una de las coordinaciones.
- e) Verificar si el Instituto cuenta con los recursos suficientes para becar a un determinado estudiante.
- f) Verificar si el becario amerita la renovación de acuerdo con el desarrollo académico que tenga y el desempeño de sus actividades.
- g) Verificar cuánto dinero gasta en becas cada una de sus coordinaciones.

***Responsabilidades del Director.***

- a) Verificar cuánto dinero se gasta en becas en cada coordinación y en cada subdirección del Instituto.
- b) Autorizar el ingreso de un solicitante y su estipendio de acuerdo con el resultado de la evaluación, recursos disponibles y programas de investigación.
- c) En caso de rechazar una solicitud de beca, el director deberá comunicar la razón de esta decisión al tutor. En el caso de aprobación, puede anexar algún comentario a la beca.
- d) Analizar y evaluar el desempeño académico de los becarios del Instituto en forma global (promedio, avance de créditos, grado académico obtenido).
- e) Consultar los registros históricos que le permitan mantener la vinculación con exbecarios.

***Responsabilidades del Secretario Académico.***

Las responsabilidades del Secretario Académico se definen a continuación:

- a) Dar de alta y baja en el sistema a cada uno de los tutores, coordinadores, subdirectores, secretarios y al director del Instituto, introduciendo la fecha de nacimiento y el nombre completo de cada académico.
- b) Mantener actualizada la información de las funciones, que desempeñan cada uno de los académicos registrados en el sistema.

- c) Mantener actualizados los tabuladores de los montos de las becas regulares.
- d) Mantener actualizada la información de las unidades organizacionales del Instituto (coordinaciones, subdirecciones y secretarías).
- e) Mantener actualizada la información de los catálogos (universidades, períodos de becas, instituciones externas de financiamiento, justificaciones para la autorización de becas especiales, motivos de rechazo de becas, etc.).
- f) Generar los reportes estadísticos.
- g) Autorizar las modificaciones al sistema.

### ***Responsabilidades del Operador del Sistema.***

Es necesaria la figura de un administrador eventual del sistema quien se hará responsable de las siguientes actividades:

- a) Registrar al Secretario Académico.
- b) Asegurar el funcionamiento del sistema, lo cual incluye el mantenimiento del hardware y el de la infraestructura requerida.
- c) Realizar los respaldos de la base de datos y del código fuente del sistema.
- d) Hacer las modificaciones que se requieran en el código del sistema, con previa autorización del Secretario Académico.

### ***Sobre las becas.***

- a) Las becas del PBII se otorgan por períodos máximos de un semestre y sólo pueden renovarse semestralmente.
- b) De acuerdo con el reglamento del PBII, las becas regulares se renuevan hasta cuatro veces en los niveles de licenciatura o maestría y hasta nueve veces en el de doctorado, incluyendo la maestría.
- c) Las solicitudes de becas de primer ingreso deben de considerar los siguientes puntos:
  - a. En caso de ser asignadas fuera de un periodo de renovación, deben de ser promovidas por el tutor al menos 5 días antes de la fecha de inicio de la beca.

- b. La fecha de inicio debe de ser el primer día de cada quincena que esté dentro del período de becas.
- d) Con respecto a las solicitudes de renovación de beca, se deben de tomar en cuenta los siguientes aspectos:
  - a. Deben de ser promovidas por el tutor al menos 15 días antes de la fecha de inicio del período.
  - b. Para las becas regulares, la fecha de inicio debe de ser la misma que marca el período de becas correspondiente.
- e) En cuanto a las becas especiales, éstas sólo se deben otorgar en casos excepcionales. Estos casos pueden comprender los siguientes:
  - El estudiante tiene un promedio por debajo del requerido por el PBII.
  - El estudiante ha excedido el número de renovaciones de beca regular.
  - El estudiante ofrece servicios temporales al Instituto.
  - El estudiante solicitó una renovación después del período de inscripción.
  - El estudiante realiza una actividad que requiere de mayor responsabilidad.
  - El estudiante trabaja un mayor número de horas a la semana que las requeridas.

En caso de que el tutor actual sea sustituido por otro cuando el período de becas ya ha iniciado, no será necesario solicitar una renovación de beca, sólo se registrarán los cambios pertinentes y la beca actual continuará vigente hasta el término del período. Lo anterior se hace con el objetivo de que el becario no pierda una de sus oportunidades de renovación.

### ***Sobre el formato de la solicitud.***

Al formato original de la solicitud del PBII se le han hecho algunas modificaciones, entre los cuales destaca la anulación de los campos correspondientes al número y nombre del proyecto. Esto último se debe a que actualmente la información correspondiente al dinero asignado a cada becario se maneja con el Fondo de Becas.

Además de estas modificaciones se proponen otras que incluyen agregar algunos campos:

- a) RFC
- b) CURP
- c) Correo electrónico
- d) Nombre completo de la licenciatura, maestría o doctorado que cursa actualmente.
- e) Avance de créditos.

- f) País de procedencia en lugar de la Nacionalidad.
- g) Monto de la beca especial
- h) Justificación de la beca especial

**Sobre los tipos de estudiantes.**

Es necesario considerar el registro de distintos tipos de estudiantes:

- a) **Becarios:** Reciben estipendio y pertenecen al PBII. El proceso de aprobación de un becario involucra al tutor, al coordinador, al subdirector y al director.
- b) **Prestadores de Servicio Social:** Son estudiantes que desean realizar su servicio social en el Instituto de Ingeniería y no reciben estipendio. El proceso de aprobación de un prestador de servicio social involucra al tutor y al Secretario Académico. Las especificaciones para los prestadores de servicio social se explican más adelante.
- c) **Becarios que además son prestadores de servicio social:** Son becarios que ingresaron al PBII por medio del proceso de aprobación descrito anteriormente y que posteriormente decidieron, además de ser becarios, realizar su servicio social en el Instituto de Ingeniería. Por esta última actividad los becarios no reciben estipendio.
- d) **Estudiantes sin estipendio:** Pueden ser tesistas sin beca o personal de apoyo del Instituto. El proceso de registro de este tipo de estudiantes incluye al tutor y al coordinador correspondiente.

Clasificación general de los estudiantes que laboran en el Instituto de Ingeniería.

Los tipos de estudiantes descritos anteriormente, de acuerdo a su nivel académico y al estipendio que reciben pueden clasificarse de la siguiente forma:

		Nivel Académico						
		Serv. Soc.	L1	L2	M1	M2	D1	D2
Estipendio	0	LS	L10	L20	M10	M20	D10	D20
	<	LS<	L1<	L2<	M1<	M2<	D1<	D2<
	=	LS=	L1=	L2=	M1=	M2=	D1=	D2=
	>	LS>	L1>	L2>	M1>	M2>	D1>	D2>

- LS: Servicio Social (únicamente a nivel L1).
- L1: Estudiante de licenciatura sin tema de tesis aprobado.
- L2: Estudiante de licenciatura con tema de tesis aprobado.
- M1: Estudiante de maestría sin tema de tesis aprobado.
- M2: Estudiante de maestría con tema de tesis aprobado.

D1: Estudiante de doctorado sin tema de tesis aprobado.

D2: Estudiante de doctorado con tema de tesis aprobado.

0 : El estudiante no recibe estipendio.

< : El estudiante recibe una beca *menor* a la cantidad que indica el tabulador.

= : El estudiante recibe una beca *igual* a la cantidad que indica el tabulador.

> : El estudiante recibe una beca *mayor* a la cantidad que indica el tabulador.

	Becas Especiales
	Becas sin estipendio*
	Becas Regulares

\*No afecta al fondo de becas

### ***Sobre los registros históricos.***

Los registros históricos de los becarios que han sido dados de baja deben de contener la siguiente información:

- a) Nombre completo.
- b) RFC
- c) Institución educativa de procedencia.
- d) País de procedencia.
- e) Fecha de Ingreso al PBII.
- f) Fecha de baja del PBII.
- g) Períodos de becas en los que ha estado inscrito.
- h) Proyectos realizados por períodos.
- i) Tutores y coordinaciones a las que perteneció en cada uno de los períodos de beca dentro del PBII.
- j) Grado de estudios obtenido al ser dado de baja del PBII.
- k) Promedio General Obtenido.
- l) Razón de la baja.
- m) Información adicional sobre las actividades que el becario desarrollará después de haber sido dado de baja del PBII.

### ***Sobre los tipos de bajas del PBII.***

Los motivos de baja de un becario del PBII pueden incluir a los siguientes:

- a) El becario cumplió con su programa de trabajo:
  - El becario ha ingresado a la fuerza laboral.
  - El becario inició estudios de postgrado en el territorio nacional o en el extranjero
  - El becario regresó a su país de origen (si es extranjero).

b) El becario no cumplió:

- El becario no reunió los requisitos académicos.
- El becario no cumplió con los objetivos de su programa de trabajo.

### **5.1.2 Requerimientos del registro de los prestadores de Servicio Social.**

#### ***Generalidades.***

- a) El sistema debe de considerar el registro de un prestador de servicio social.
- b) El sistema debe considerar la baja definitiva de un prestador de servicio social, ya sea cuando se de por terminado el servicio social o cuando exista alguna circunstancia que lo justifique.
- c) El sistema debe de permitir cambios a la información académica y al horario de trabajo del estudiante. Estos cambios podrán realizarse al inicio de cada semestre escolar.

#### ***Sobre el Proceso de Aprobación.***

- a) Se debe verificar que el Instituto de Ingeniería tenga vigente uno o varios programas de servicio social aprobados por la Dirección General de Orientación y Servicios Escolares (DGOSE).
- b) Para aceptar a un estudiante como prestador de servicio social del Instituto de Ingeniería se respetará el reglamento de servicio social de cada Escuela o Facultad.
- c) El proceso de aprobación de la solicitud de un estudiante que desea realizar el servicio social en el Instituto de Ingeniería, involucra al tutor y al Secretario Académico.

#### ***Sobre el formato de registro.***

Se debe de establecer un formato especial para el registro de los prestadores de servicio social. Este formato debe de incluir al menos los siguientes registros:

- a) Datos Personales:
  - Nombre completo del estudiante.
  - RFC.
  - CURP.
  - Domicilio (actualizable).
  - Teléfono (opcional y actualizable).

- Nacionalidad.
- Fecha de Nacimiento.
- Estado Civil (actualizable).
- Correo Electrónico (actualizable).

b) Datos Académicos:

- Número de matrícula de la Universidad a la que pertenece.
- Institución educativa de donde proviene.
- Nombre de la carrera que cursa.
- Avance de créditos.
- Promedio de calificaciones.

c) Datos sobre el servicio social:

- Número de registro del programa de servicio social.
- Nombre del programa de servicio social.
- Nombre del responsable directo.
- Fecha de inicio del servicio social.
- Fecha de terminación del servicio social.
- Número de horas a la semana.
- Carreras Solicitadas.
- Breve descripción de las actividades a desarrollar por carrera.

***Responsabilidades del estudiante.***

- a) Introducir su información académica y personal en el sistema.
- b) Cumplir con un avance de créditos mayor o igual al 70%.
- c) Cumplir con las horas semanales comprometidas con su tutor para la realización del servicio social.

***Responsabilidades del tutor.***

- a) Elaborar la propuesta de un nuevo programa de servicio social y llenar el formato correspondiente expedido por la DGOSE. Una vez terminada la propuesta, el tutor debe de entregarla al Secretario Académico para que éste apruebe el programa y lo firme. Posteriormente el tutor debe de llevar el formato firmado por él mismo (en calidad de responsable directo) y por el Secretario Académico (en calidad de secretario administrativo) a la DGOSE y esperar su aprobación definitiva. El tutor debe de informar al Secretario Académico de la aprobación del programa de servicio social para que lo registre en la base de datos. En caso de que el programa no sea aprobado, el tutor debe de realizar las correcciones pertinentes y repetir el proceso anterior.



- b) El tutor debe de ser el responsable directo del programa de servicio social al que ingrese el estudiante.
- c) Verificar que el programa de servicio social del cual es responsable directo, incluya la carrera del solicitante. También debe de constatar que el estudiante cumpla con los requisitos para realizar servicio social estipulados por su Institución educativa.
- d) Verificar los datos del aspirante y aprueba o rechaza la información.
- e) El tutor y el estudiante deben de fijar conjuntamente el tiempo de liberación del Servicio Social. Este período sólo puede ser de 6 meses o 1 año.
- f) Registrar en el sistema las actividades específicas que va a desarrollar el estudiante como prestador de servicio social.
- g) Verificar que el estudiante cumpla con la cantidad de horas semanales comprometidas para la realización del servicio social.
- h) Verificar que el estudiante cumpla con el plan de trabajo al que se ha comprometido.
- i) Firmar los reportes bimestrales del estudiante.
- j) El tutor tiene la autoridad para dar de alta a un posible prestador de servicio social en la base de datos del sistema, introduciendo el RFC y el nombre completo del estudiante.
- k) En caso de que la solicitud del posible prestador de servicio social sea rechazada, el tutor debe dar de baja del sistema al estudiante.
- l) El tutor tiene la autoridad para dar de baja del sistema y cancelar el servicio social de cualquier estudiante a su cargo que no cumpla con las obligaciones comprometidas.
- m) En el momento de la liberación del servicio social, el tutor debe de certificar que el estudiante cumplió con el número de horas comprometidas y las actividades asignadas, dando aviso de ello al Secretario Académico para generar la constancia correspondiente.
- n) Firmar la carta de liberación de servicio social de un estudiante.

### ***Responsabilidades del Secretario Académico.***

- a) El Secretario Académico es el responsable administrativo de cualquier programa de servicio social del Instituto de Ingeniería registrado ante la DGOSE.

- b) Una vez que el programa de servicio social ha sido aprobado y registrado por la DGOSE, el Secretario Académico es el responsable de registrar en la base de datos el programa incluyendo la siguiente información:
- Número de registro del programa.
  - Nombre del programa.
  - Nombre del responsable directo.
  - Período de Vigencia.
  - Carreras que incluye.
  - Actividades Generales.
- c) Durante el proceso de aceptación del posible prestador de servicio social, el Secretario Académico es el responsable de verificar que el tutor esté asignado al programa de servicio social al cual se pretende inscribir al estudiante. De igual manera debe de aprobar el inicio y fin del periodo de trabajo.
- d) El Secretario Académico junto con el tutor es el responsable de firmar la carta de liberación de servicio social del estudiante.

### **5.1.3 Requerimientos del registro de los estudiantes sin estipendio.**

#### ***Misión de los funcionarios involucrados.***

Tal y como se definió anteriormente, los funcionarios responsables de la aprobación de la solicitud de un estudiante sin estipendio son el tutor y el coordinador correspondiente.

#### ***Responsabilidades del Estudiante.***

- a) Registrar su información académica y personal. Esta información es la misma que se utiliza para el registro de becarios.
- b) Actualizar su información académica y su horario de trabajo al inicio de cada semestre escolar.

#### ***Responsabilidades del Tutor.***

- a) Dar de alta en la base de datos a un estudiante sin estipendio, introduciendo su RFC y su nombre completo.
- b) Aprobar los datos personales y académicos que introdujo el estudiante en el sistema.

- c) Asignar al estudiante las actividades y el proyecto que va a desarrollar, así como la duración aproximada del mismo.
- d) En caso de ser tutor de un tesista, el tutor debe de verificar que el estudiante haya liberado el Servicio Social.
  
- e) Asimismo para el caso de un tesista, el tutor debe de ingresar al sistema la siguiente información adicional:
  - Nombre de la tesis (actualizable)
  - Especificar si la tesis está registrada como seminario y/o tesis.
  - Horario de trabajo
  - Fecha de Registro de la tesis.
  - Estado actual de la tesis (inicio, terminada, etc.)
  - Especificar si el estudiante recibió mención honorífica.

Al concluir la tesis, se debe de introducir la siguiente información:

- Fecha de terminación.
  - Duración total de la tesis.
  - Fecha de examen profesional.
  - Grado obtenido por el tesista.
- 
- f) Informar al coordinador del ingreso de un estudiante sin estipendio, ya sea personal de apoyo o tesista.
  
  - g) En el caso de un tesista, el tutor debe de informar al Secretario Académico del ingreso del estudiante.

#### ***Responsabilidades del Coordinador.***

- a) Consultar el número de estudiantes sin estipendio que están trabajando en su coordinación.

#### ***Responsabilidades del Secretario Académico.***

- a) Consultar el número de tesis que tiene el Instituto.

## 5.2 Modelado de datos.

### 5.2.1 Diccionario de Datos.

En este apartado se muestran las características generales las principales tablas que conforman la base de datos del sistema, las demás pueden ser consultadas en el manual técnico que pertenece al Instituto de Ingeniería.

**TABLA ESTUDIANTES**

Columna	EST RFC	NOMBRE	AP PAT	AP MAT	CURP
<b>Tipo llave</b>	PK				
<b>Tipo de dato</b>	varchar	varchar	varchar	varchar	varchar
<b>Longitud</b>	13	30	30	30	30
<b>Nulos / únicos</b>	NN,U	NN	NN	NN	
<b>Descripción</b>	RFC del estudiante	Nombre(s) del estudiante	Apellido Paterno	Apellido Materno	Clave única de población
<b>Ejemplo</b>	MOPG770419	Guillermo	Mora	Pérez	MOPG772125KSA

Columna	CALLE	NUMERO	NUM_INT	COLONIA
<b>Tipo llave</b>				
<b>Tipo de dato</b>	varchar	varchar	varchar	Varchar
<b>Longitud</b>	40	5	5	30
<b>Nulos / únicos</b>				
<b>Descripción</b>	Calle de residencia	Número de la calle	Número interior	Colonia
<b>Ejemplo</b>	Cerro de las Flores	500	2	Las Flores

Columna	DELEG	CP	CIUDAD	FECHA_NAC	EDOCIV_ID
<b>Tipo llave</b>					FK2
<b>Tipo de dato</b>	varchar	varchar	varchar	smalldatetime	Int
<b>Longitud</b>	30	10	30	4	4
<b>Nulos / únicos</b>					
<b>Descripción</b>	Delegación o municipio	Código postal	Ciudad de residencia	Fecha de nacimiento	identificador de estado civil
<b>Ejemplo</b>	Coyoacán	08810	México	19/04/1977	1

Columna	SEX_ID	TELEFONO	EMAIL	LOGIN_II
<b>Tipo llave</b>	FK1			
<b>Tipo de dato</b>	int	varchar	varchar	Varchar
<b>Longitud</b>	4	10	30	30
<b>Nulos / únicos</b>				
<b>Descripción</b>	Identificador de sexo	Teléfono	Correo electrónico	Login de acceso para Windows NT
<b>Ejemplo</b>	2	55173835	gmorap@iingen.mx	GMoraP

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DEL SISTEMA.

<b>Columna</b>	PAIS_ID	HOR_TRAB	CURRIC	HISTORIAL	STAT_ID	FECHA_MOD	PASSWD
<b>Tipo llave</b>	FK3				FK4		
<b>Tipo de dato</b>	int	varchar	varchar	varchar	int	smalldatetime	varchar
<b>Longitud</b>	4	504	30	30	4	4	20
<b>Nulos / únicos</b>							
<b>Descripción</b>	identificador del país de procedencia	Horario de trabajo	Currículo	Historial académico del nivel en curso	Identificación del estatus del estudiante	Fecha de actualización	Contraseña del usuario.
<b>Ejemplo</b>	75	131162113 143184	Referencia a archivo	Referencia a archivo	1	20/03/2002	aGb7142

**TABLA ACADÉMICOS**

<b>Columna</b>	ACA_RFC	NOMBRE	AP_PAT	AP_MAT	FECHA_NAC	SEX_ID
<b>Tipo llave</b>	PK					FK1
<b>Tipo de dato</b>	varchar	varchar	varchar	varchar	smalldatetime	int
<b>Longitud</b>	13	30	30	30	4	4
<b>Nulos / únicos</b>	NN,U	NN	NN	NN		
<b>Descripción</b>	RFC del académico	Nombre	Apellido paterno	Apellido materno	Fecha de Nacimiento	Identificador del sexo
<b>Ejemplo</b>	AMMM611015	Marco	Ambríz	Maguey		2

<b>Columna</b>	TELEFONO	EMAIL	LOGIN_II	PAIS_ID	GRAD_ID	STAT_ID
<b>Tipo llave</b>				FK2	FK2	
<b>Tipo de dato</b>	Varchar	varchar	Varchar	int	int	int
<b>Longitud</b>	30	50	30	4	4	4
<b>Nulos / únicos</b>						NN
<b>Descripción</b>	Teléfono Personal	Correo electrónico	Login de acceso para Windows NT	Identificador del país	Identificador del Máximo Grado de Estudios.	Identificador de l estado del acceso del académico.
<b>Ejemplo</b>	56565453	mam@pumas.unam.mx	MAmbrizM	75	1	22

<b>Columna</b>	FECHA_MOD	PASSWD
<b>Tipo llave</b>		
<b>Tipo de dato</b>	smalldatetime	Varchar
<b>Longitud</b>	4	20
<b>Nulos / únicos</b>	NN	
<b>Descripción</b>	Fecha de Actualización de los datos.	Contraseña de acceso.
<b>Ejemplo</b>	30/02/2002	Ab5w1t67E

TABLA REGISTROS\_ESTUDIOS

Columna	EST RFC	UNIV_ID	GRADO_ID	CARR_ID	MATRICULA	CREDITOS
Tipo llave	PK,FK1	PK,FK2	PK,FK3	PK,FK4		
Tipo de dato	varchar	int	int	int	varchar	varchar
Longitud	13	4	4	4	20	6
Nulos / únicos	NN,U1	NN,U1	NN,U1	NN,U1		NN
Descripción	RFC del estudiante	identificador de la universidad donde se cursa o cursó la carrera	identificador del grado de estudios	identificador de la carrera cursada	Matricula o número de cuenta del estudiante en la universidad	Créditos cubiertos
Ejemplo	MOPG770419	110	1	10	0933301691	100

Columna	PROMED	FECHA_TIT	STAT_ID
Tipo llave			FK5
Tipo de dato	varchar	smalldatetime	Int
Longitud	5	4	4
Nulos / únicos	NN		
Descripción	Promedio actual o final	Fecha de titulación	identificador del estatus del registro de estudios
Ejemplo	8.4	01/01/2040	1

TABLA BECAS\_REGULARES

Columna	EST RFC	PER_BEC_ID	ACA RFC	BEC_NIV_ID	FECHA_SOL	FECHA_INI
Tipo llave	PK,FK1	PK,FK2	FK3	FK4		
Tipo de dato	varchar	int	varchar	int	smalldatetime	smalldatetime
Longitud	13	4	13	4	4	4
Nulos / únicos	NN,U1	NN,U1	NN	NN	NN	NN
Descripción	RFC del estudiante	Identificador del periodo de becas	RFC del académico responsable	Identificador del nivel de beca	Fecha de solicitud de la beca	Fecha de inicio de la beca
Ejemplo	MOPG770419	15	AMMM611015	L2	27/01/2001	01/03/2001

Columna	FECHA_FIN	PROY_NOM	PLAN_TRAB	INFORM	STAT_ID
Tipo llave					
Tipo de dato	smalldatetime	Varchar	text	text	int
Longitud	4	60	16	16	4
Nulos / únicos	NN	NN			NN
Descripción	Fecha de terminación de la beca	Nombre del proyecto al cual estará asignado	Plan de trabajo para el periodo	Informe final de actividades.	Identificador del estado de la beca.
Ejemplo	31/08/2001	Sistema de Becas.	<texto indefinido>	<texto indefinido>	1

ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DEL SISTEMA.

<b>Columna</b>	FECHA_ACT	PROY_NUM	STAT_INFORME
<b>Tipo llave</b>			
<b>Tipo de dato</b>	smalldatetime	varchar	varchar
<b>Longitud</b>	4	4	1
<b>Nulos / únicos</b>			
<b>Descripción</b>	Fecha de actualización	Número del proyecto (informativo)	Estado del informe de actividades.
<b>Ejemplo</b>	30/01/2001	2424	1

**TABLA BECAS\_ESPECIALES**

<b>Columna</b>	EST RFC	PER_BEC_ID	ACA RFC	FECHA_SOL	FECHA_INI	FECHA_FIN
<b>Tipo llave</b>	PK,FK1	FK2	FK3			
<b>Tipo de dato</b>	varchar	int	varchar	smalldatetime	smalldatetime	smalldatetime
<b>Longitud</b>	13	4	13	4	4	4
<b>Nulos / únicos</b>	NN,U1	NN	NN	NN	NN,U1	NN
<b>Descripción</b>	RFC del estudiante	identificador del periodo de becas	RFC del académico responsable	Fecha de solicitud de la beca	Fecha de inicio de la beca	Fecha de terminación de la beca
<b>Ejemplo</b>	MOPG770419	16	AMMM611015	27/08/2001	01/09/2001	28/02/2002

<b>Columna</b>	PROY_NOM	PLAN_TRAB	INFORM	JUST_ID
<b>Tipo llave</b>				FK4
<b>Tipo de dato</b>	varchar	Text	text	int
<b>Longitud</b>	60	16	16	4
<b>Nulos / únicos</b>	NN			NN
<b>Descripción</b>	Nombre del proyecto a desarrollar	Plan de trabajo para el periodo	Informe final	Identificador de la justificación de beca
<b>Ejemplo</b>	Sistema de Becas	Desarrollo del Sistema de Becas.	Se hicieron las pruebas del sistema.	6

<b>Columna</b>	STAT_INFORME	STAT_ID	FECHA_ACT	PROY_NUM
<b>Tipo llave</b>				
<b>Tipo de dato</b>	varchar	Int	smalldatetime	varchar
<b>Longitud</b>	1	4	4	4
<b>Nulos / únicos</b>		NN		
<b>Descripción</b>	Estado del informe de actividades.	Identificador del estado de la beca.	Fecha de actualización	Número del proyecto (informativo)
<b>Ejemplo</b>	1	1	30/08/2002	2424

**TABLA FUNCIONES**

Columna	ACA RFC	CATEG_ID	ORG_UNIT_ID	FECHA_ACT
Tipo llave	PK,FK1	PK,FK2	FK3	
Tipo de dato	varchar	int	int	Smalldatetime
Longitud	13	4	4	4
Nulos / únicos	NN,U1	NN,U1	NN	NN
Descripción	RFC del academico	Identificador de la categoría del academico	Identificador de la unidad organizacional donde desempeña la categoría	Fecha de actualización
Ejemplo	AMMM611015	1	20	12/12/2001

**TABLA UNIDADES\_ORGANIZACIONALES**

Columna	ORG_UNIT_ID	NOMBRE	UNIT_SUPER_ID
Tipo llave	PK,FK		FK
Tipo de dato	int	varchar	int
Longitud	4	60	4
Nulos / únicos	NN,U	NN	NN
Descripción	Identificador de la unidad organizacional	Nombre de la unidad organizacional	Identificador de la unidad organizacional superior
Ejemplos	20	Coordinación de Sistemas de Cómputo	4
	4	Subdirección de Electromecánica y Cómputo.	1
	1	Dirección	1

**5.2.2 Diagrama Entidad-Relación.**

El diagrama que esquematiza la estructura de todas las tablas incluidas en la base de datos del sistema, se muestra en la siguiente página.





## 5.3 Análisis del Sistema.

### 5.3.1 Casos de Uso.

Un caso de uso es una descripción de un conjunto de secuencias de acciones, incluyendo las variantes, que ejecuta un sistema para producir un resultado observable de valor para un actor.

Un actor representa un conjunto coherente de roles que los usuarios de los casos de uso juegan al interactuar con éstos. Normalmente, un actor representa un rol que es jugado por una persona, un dispositivo de hardware o incluso otro sistema al interactuar con nuestro sistema.

Los actores sólo se pueden conectar a los casos de uso a través de asociaciones. Una asociación entre un actor y un caso indica que el actor y el caso de uso se comunican entre sí, y cada uno puede enviar y recibir mensajes.

En términos generales un caso de uso describe qué hace un sistema, pero no especifica cómo lo hace. Gráficamente, un caso de uso se representa como una elipse.

Para el análisis de requerimientos del SCBII, fue necesario identificar a los actores que iban a interactuar con él y las acciones (casos de uso) que iban a ejecutar.

#### 5.3.1.1 Actores del sistema.

¿Quién va a usar el sistema?

- Estudiantes
- Funcionarios (Tutores, Coordinadores, Subdirectores, Director y Secretario Académico)
- Administrador del sistema.

#### 5.3.1.2 Acciones de los actores.

¿Qué pueden hacer los actores con el sistema?

a) Becarios:

- Registrar su información personal y académica.
- Consultar el estado de su solicitud de beca.

b) Tutores

- Dar de alta a un estudiante nuevo en el sistema.
- Evaluar los datos académicos y personales del estudiante.
- Asignar una propuesta de beca al estudiante.

- Cancelar una beca.
- Dar de baja del sistema a los becarios o candidatos a becario.
- Monitorear el estado de sus propuestas de beca.
- Actualizar sus datos personales.

c) Funcionarios (Coordinador, Subdirector, Director.)

- Evaluar los datos académicos y personales del estudiante.
- Evaluar una solicitud de beca.
- Actualizar Datos Personales.

d) Secretario Académico.

- Dar de alta al director actual del Instituto.
- Dar de alta a los subdirectores.
- Dar de alta a los secretarios.
- Dar de alta a los coordinadores y tutores.
- Mantener actualizada la información académica.

**5.3.1.3 Descripción de los casos de uso.**

**Caso de Uso: Validar Usuario.**

**Actores:** Estudiante, Tutor, Funcionario (Coordinador, Subdirector, Director).

**Resumen:** El usuario se identifica en el sistema y éste lo envía a otra página de acuerdo a su jerarquía.

**Curso Normal de los eventos.**

Acción de los actores	Acción del sistema
1. El usuario solicita el acceso a la página principal del sistema.	2.- Verifica la dirección IP de la computadora del alumno. Si la IP pertenece al II le envía la página principal de acceso al sistema.
3. El usuario introduce su RFC y su contraseña.	4. Verifica el RFC y la contraseña. Envía a otra página de acuerdo a la categoría del usuario.

**Cursos Alternos:**

Línea 2: La dirección IP es inválida y envía una página de acceso no autorizado.

Línea 4: El RFC o la contraseña son incorrectos y regresa a la página principal del sistema.

**Caso de Uso: Registrar Estudiante.****Casos de uso que incluye:** Validar Usuario.**Actores:** Tutor**Resumen:** Un tutor registra a un estudiante nuevo en el sistema**Curso normal de los eventos.**

<b>Acción de los actores</b>	<b>Acción del sistema</b>
1. Dentro del menú, escoge la opción de "Estudiante Nuevo"	2. El sistema verifica la categoría del usuario y permite el ingreso al tutor.
3. El tutor introduce el nombre completo y fecha de nacimiento del estudiante que intenta dar de alta. A continuación, envía la información.	4. El sistema valida los datos y genera la clave y la contraseña del estudiante. Finalmente regresa al tutor una página para que verifique la información.
5. El tutor verifica y acepta la información.	6. El sistema introduce a la base de datos la información y regresa al tutor una página con el estado de la transacción realizada.

**Cursos Alternos:**

Línea 4: El sistema detecta algún error al verificar los datos y envía la descripción del error, para que el tutor lo corrija.

Línea 5: El tutor decide cambiar la información.

**Caso de Uso: Llenar Datos Personales y Académicos.****Casos de uso que incluye:** Validar Usuario, Registrar Estudiante.**Actores:** Estudiante.**Resumen:** El estudiante llena sus datos personales y académicos además de información requerida por el Instituto.**Curso normal de los eventos.**

<b>Acción de los actores</b>	<b>Acción del sistema</b>
1. El estudiante llena por primera vez sus datos personales y envía la información.	2. El sistema valida los campos y envía al estudiante una segunda página para que llene sus datos académicos
3. El estudiante llena por primera vez, los datos académicos requeridos y envía la información.	4. El sistema valida los campos y envía al estudiante una tercera página para que llene el horario de asistencia al Instituto y el plan de trabajo.
5. El estudiante llena los datos requeridos y los envía.	6. El sistema verifica la información y regresa una página para que el estudiante verifique por última vez los datos que introducirá.
7. El estudiante aprueba la	8. El sistema introduce la información

información y envía los datos.	a la base de datos y bloquea el acceso al estudiante para que no pueda cambiar sus datos. Finalmente, el sistema muestra una página con el resultado de la transacción.
--------------------------------	---

**Cursos Alternos:**

Línea 2,4 y 6: El sistema detecta algún error en los datos introducidos y no permite continuar el proceso.

Línea 7: El estudiante cambia algún dato.

**Caso de Uso: Actualizar Datos Personales y Académicos.**

**Casos de uso que incluye:** Validar Usuario.

**Actores:** Estudiante.

**Resumen:** El estudiante actualiza sus datos personales y académicos además de información requerida por el instituto.

**Curso normal de los eventos.**

Acción de los actores	Acción del sistema
1. El estudiante actualiza sus datos personales y envía la información.	2. El sistema valida los campos y envía al estudiante una segunda página para que actualice y/o agregue sus datos académicos.
3. El estudiante actualiza y/o agrega los datos académicos requeridos y envía la información.	4. El sistema valida los campos y envía al estudiante una tercer página para que actualice el horario de asistencia al Instituto, el plan de trabajo y/o informe de actividades.
5. El estudiante actualiza y/o agrega los datos requeridos y los envía.	6. El sistema verifica la información y regresa una página para que el estudiante verifique por última vez los datos que introducirá.
7. El estudiante aprueba la información y envía los datos.	8. El sistema actualiza la información en la base de datos y bloquea el acceso al estudiante para que no pueda cambiar sus datos. Finalmente, el sistema muestra una página con el resultado de la transacción.

**Cursos Alternos:**

Línea 2,4 y 6: El sistema detecta algún error en los datos introducidos y no permite continuar el proceso.

Línea 7: El estudiante cambia algún dato.

**Caso de Uso: Dar de baja a Estudiante.**

**Casos de uso que incluye:** Validar Usuario.

**Actores:** Tutor.

**Resumen:** El tutor escoge uno de los estudiantes a su cargo y lo da de baja.

**Curso normal de los eventos.**

Acción de los actores	Acción del sistema
1. Dentro del menú, el tutor escoge la opción de "Baja -> Estudiante"	2. El sistema muestra a todos los estudiantes que el tutor tiene a su cargo.
3. El tutor selecciona al estudiante que desea dar de baja.	4. El sistema muestra los datos personales y académicos del estudiante, así como su trayectoria dentro del PBII.
5. El tutor verifica los datos del estudiante y llena la información requerida para poderlo dar de baja.	6. El sistema verifica la información introducida por el tutor, la introduce a la base de datos y desasocia al alumno del tutor. El sistema muestra de nuevo la página donde el tutor puede escoger alguno de sus becarios para dalo de baja.

**Cursos Alternos:**

Línea 4:

- a) El sistema detecta que el estudiante tiene una beca activa o aprobada bajo la responsabilidad del tutor y le manda un mensaje de advertencia.
- b) El sistema detecta que el estudiante tiene una solicitud de beca bajo la responsabilidad del tutor y le manda un mensaje de advertencia.

Línea 6:

- a) El sistema detecta algún error en los datos introducidos y no permite continuar el proceso.
- b) El sistema detecta que el estudiante tiene una beca activa bajo la responsabilidad del tutor y la cancela.
- c) El sistema detecta que el estudiante tiene una beca aprobada no activa bajo la responsabilidad del tutor y la cancela.
- d) El sistema detecta que el estudiante tiene una solicitud de beca bajo la responsabilidad del tutor y le asigna a la solicitud el estado de rechazada definitivamente.

**Caso de Uso: Asignar Beca.****Casos de uso que incluye:** Validar Usuario**Actores:** Tutor**Resumen:** Un tutor selecciona un estudiante asignado a él, revisa su plan de trabajo y le asigna una beca.**Curso normal de los eventos.**

Acción de los actores	Acción del sistema
1. Dentro del menú, escoge la opción de "Asignación y renovación de becas"	2. El sistema verifica la categoría del usuario y permite el ingreso al tutor.
3. El tutor selecciona a un estudiante de entre los que están asignados a él y que tienen derecho de renovación.	4. Envía a la página que contiene los datos del estudiante.
5. Evalúa los datos del estudiante y escoge el tipo de beca.	6. Despliega el formulario para el tipo de beca seleccionada.
7. Llena los datos de la beca asignada.	8. Verifica que los datos sean correctos y muestra una vista previa de la información que se va a enviar.
9. Aprueba la beca y envía los datos.	10. Registra los datos de la beca y le establece a ésta el estado de <i>aprobada por el tutor</i> .

**Cursos Alternos:**

Línea 5:

- a) El estudiante no ha llenado su plan de trabajo y/o informe de actividades y el tutor no puede asignar una beca.
- b) El tutor no está de acuerdo con los datos del estudiante y no asigna una beca.

Línea 8: Los datos de la beca son incorrectos y el tutor los corrige.

**Caso de Uso: Evaluar solicitud de beca.****Actores:** Funcionario**Casos de uso que incluye:** Validar Usuario.**Resumen:** Un funcionario selecciona una solicitud de beca, la evalúa, posteriormente la aprueba y el sistema la envía al siguiente funcionario en la cadena de evaluación.**Curso Normal de los Eventos:**

Acción de los actores	Respuesta del sistema
1. El funcionario escoge del menú principal, la opción de "Revisión de	2. Verifica la categoría y permite el acceso al funcionario. Muestra un

Becas".	menú con los nombres de los tutores y con el número de becas pendientes de revisión por cada uno de ellos.
3. El funcionario selecciona a un tutor que tenga becas pendientes.	4. Despliega una página con los datos generales de las solicitudes de beca asociadas al tutor.
5. El funcionario escoge una solicitud de beca.	6. Despliega los datos académicos y personales del solicitante de la beca seleccionada. Asimismo, muestra los datos de la beca propuesta.
7. Evalúa los datos desplegados. Aprueba la solicitud.	8. Despliega una página indicando que la solicitud de beca ha sido aprobada y cambia el estado de la solicitud para que el siguiente funcionario en la jerarquía pueda evaluarla.

**Cursos alternos:**

Línea 3: El funcionario no tiene becas pendientes y sale del sistema.

Línea 7: El funcionario rechaza la solicitud y especifica el motivo de rechazo.

Línea 8: Si la beca es rechazada, el sistema cambia el estado de la solicitud a *rechazada*.

**Caso de uso: Registrar Académico.  
Sección Principal.**

**Actores:** Secretario Académico.

**Casos de uso que incluye:** Validar Usuario.

**Resumen:** El Secretario Académico selecciona el tipo de funcionario que va a dar de alta en el sistema, introduce los datos personales del académico que va a registrar y le asigna una función dentro de una unidad organizacional.

**Curso normal de los eventos.**

Acción de los actores	Respuesta del sistema
<p>1. El Secretario Académico escoge en el menú de altas, la opción correspondiente al registro del académico, según la función que éste va a desempeñar:</p> <ul style="list-style-type: none"> <li>a) Si es alta de tutor, ver la sección <i>Registrar Tutor</i>.</li> <li>b) Si es alta de coordinador, ver la sección <i>Registrar Coordinador</i>.</li> <li>c) Si es alta de subdirector, ver la sección de <i>Registrar</i></li> </ul>	



<p><i>Subdirector.</i> d) Si es alta de director, ver la sección <i>Registrar Director.</i></p>	
	<p>2. Genera la clave y verifica que el académico no esté registrado en el sistema. El sistema le asigna al académico, la función especificada. Muestra una página con los datos que van a ser introducidos</p>
<p>3. El Secretario Académico verifica la información y la envía.</p>	<p>4. Introduce la información en la base de datos y muestra una página con el estado de la transacción.</p>

**Cursos alternos:**

Línea 2: El académico ya está registrado en el sistema y éste manda un mensaje de error.  
Línea 3: El funcionario decide cambiar la información y regresa a la página anterior.

**Sección: Registrar Tutor**

**Curso normal de los eventos.**

<b>Acción de los actores</b>	<b>Respuesta del sistema</b>
<p>1. El Secretario Académico selecciona la opción de "Registrar Tutor".</p>	<p>2. El sistema verifica que la categoría del usuario, muestra el formulario correspondiente al registro de un tutor.</p>
<p>3. El Secretario introduce el nombre, apellidos y la fecha de nacimiento del tutor.</p>	
<p>4. Selecciona la coordinación en la que trabajará el tutor.</p>	

**Cursos alternos:**

Línea 2: La categoría del usuario no es válida y el sistema regresa la página de inicio.

**Sección: Registrar Coordinador.**

**Curso normal de los eventos.**

<b>Acción de los actores</b>	<b>Respuesta del sistema</b>
<p>1. El Secretario selecciona la opción de "Registrar Coordinador".</p>	<p>2. El sistema verifica que la categoría del usuario válida y muestra el</p>

	formulario correspondiente al registro de coordinador
3. El Secretario introduce el nombre, apellidos y la fecha de nacimiento del coordinador.	
4. El secretario selecciona la coordinación que dirigirá el coordinador.	

**Cursos alternos:**

Línea 2: La categoría del usuario no es válida y el sistema regresa la página de inicio.

**Sección: Registrar Subdirector.****Curso normal de los eventos.**

<b>Acción de los actores</b>	<b>Respuesta del sistema</b>
1. El Secretario Académico selecciona la opción de "Registrar Subdirector".	2. El sistema verifica que la categoría del usuario sea la de Secretario Académico y muestra el formulario para el registro del subdirector.
3. El Secretario introduce el nombre, apellidos y fecha de nacimiento del subdirector.	
4. El Secretario selecciona la subdirección que estará cargo del subdirector.	

**Cursos alternos:**

Línea 2: La categoría del usuario no es válida y el sistema regresa la página de inicio.

**Sección: Registrar Director.**

**Actor:** Secretario Académico.

**Curso normal de los eventos.**

<b>Acción de los actores</b>	<b>Respuesta del sistema</b>
1. El Secretario Académico selecciona la opción de "Registrar Director".	2. El sistema verifica que la categoría del usuario sea la de Secretario Académico y muestra la sección de registro de director. Implícitamente, el sistema le asigna al académico la función de director.

**Cursos alternos:**

Línea 2: La categoría del usuario no es válida y el sistema regresa la página de inicio.

**Caso de Uso: Actualizar Datos Personales.**

**Actores:** Tutor, Funcionario (Coordinador Subdirector) y Secretario Académico.

**Casos de Uso que incluye:** Validar Usuario.

**Resumen:** Un funcionario entra al sistema para actualizar sus datos personales.

**Curso Normal de los Eventos.**

<b>Acción de los actores</b>	<b>Respuesta del sistema</b>
1. El funcionario escoge del menú de cambios, la opción de "Actualizar datos personales".	2. Verifica la categoría del usuario y muestra la página con el RFC, fecha de nacimiento y nombre completo del académico. Asimismo, se muestra el formulario con los datos que faltan por llenarse.
3. El funcionario llena el formulario de datos y envía la información.	4. Verifica la información y muestra una vista previa de los datos.
5. El funcionario verifica la información y la envía.	6. Introduce la información en la base de datos y muestra una página con el estado de la transacción.

**Cursos alternos:**

Línea 2: El usuario no es un académico y el sistema regresa la página de inicio.

Línea 4: Existen errores en los datos y se muestra una página que los describe para que sean corregidos.

Línea 5: El académico decide cambiar la información y regresa a la página anterior.

**Caso de Uso: Monitorear estado de becas.**

**Actores:** Tutor.

**Casos de Uso que incluye:** Validar Usuario.

**Resumen:** Un tutor entra al sistema para ver el estado de las becas que asignó.

**Curso Normal de los Eventos.**

<b>Acción de los actores</b>	<b>Respuesta del sistema</b>
1. El tutor escoge del menú de cambios, la opción de "Monitoreo del Proceso de Aprobación".	2. El sistema busca todas aquellas solicitudes de beca que el tutor promovió con anterioridad y cuya fecha de inicio es futura. Las clasifica

	por su estado y las muestra al interesado. En el caso de las becas que han sido rechazadas muestra el motivo del rechazo.
--	---

**Caso de Uso: Actualizar datos de becas rechazadas.**

**Actores:** Tutor.

**Casos de Uso que incluye:** Validar Usuario, Monitorear estado de becas.

**Resumen:** Un tutor detecta dentro del monitoreo de becas que una beca fue rechazada, analiza el motivo de rechazo y cambia los datos a la beca.

**Curso Normal de los Eventos.**

<b>Acción de los actores</b>	<b>Respuesta del sistema</b>
1. El tutor escoge dentro de la página del monitoreo de las becas una solicitud de beca que fue rechazada.	2. El sistema muestra los datos de la solicitud de beca permitiendo que el tutor pueda cambiar la información de la misma.
3.- El tutor hace las modificaciones correspondientes a los datos de la solicitud y la envía para su validación	4. El sistema verifica los datos que el tutor introdujo y los muestra para que éste los confirme o los modifique.
5. El tutor aprueba los datos y los envía.	6. El sistema realiza los cambios y envía la solicitud de beca al coordinador correspondiente.

**Cursos alternos:**

Línea 3: El tutor verifica que los datos a modificar son responsabilidad del estudiante y rechaza la información.

Línea 4:

- a) El tutor rechaza los datos y el sistema permite que el estudiante entre a modificar sus datos personales, académicos, plan de trabajo y/o informe de actividades.
- b) El sistema detecta errores en los datos introducidos y los muestra para su corrección.

Línea 5: El tutor decide realizar cambios a los datos que introdujo.

**Caso de Uso: Actualizar catálogos académicos.**

**Actores:** Secretario Académico.

**Casos de Uso que incluye:** Validar Usuario.

**Resumen:** El Secretario Académico escoge la tabla que quiere modificar, hace los cambios necesarios y sale del sistema.

**Curso Normal de los Eventos.**

<b>Acción de los actores</b>	<b>Respuesta del sistema</b>
1. El tutor escoge la tabla que desea modificar.	2. El sistema muestra los datos de la tabla.
3. El tutor hace cambios a la tabla y los envía.	4. El sistema verifica la validez de los datos y muestra los cambios realizados.

**Cursos alternos:**

Línea 4: El sistema detecta errores en los datos introducidos y no permite continuar con la actualización.

### 5.3.2 Diagrama de Casos de Uso.

Un diagrama de casos de uso representa un conjunto de casos de uso, actores y sus relaciones. Los diagramas de casos de uso se utilizan para describir la vista estática de un sistema. Los diagramas de casos de uso son especialmente importantes para organizar y modelar el comportamiento de un sistema.

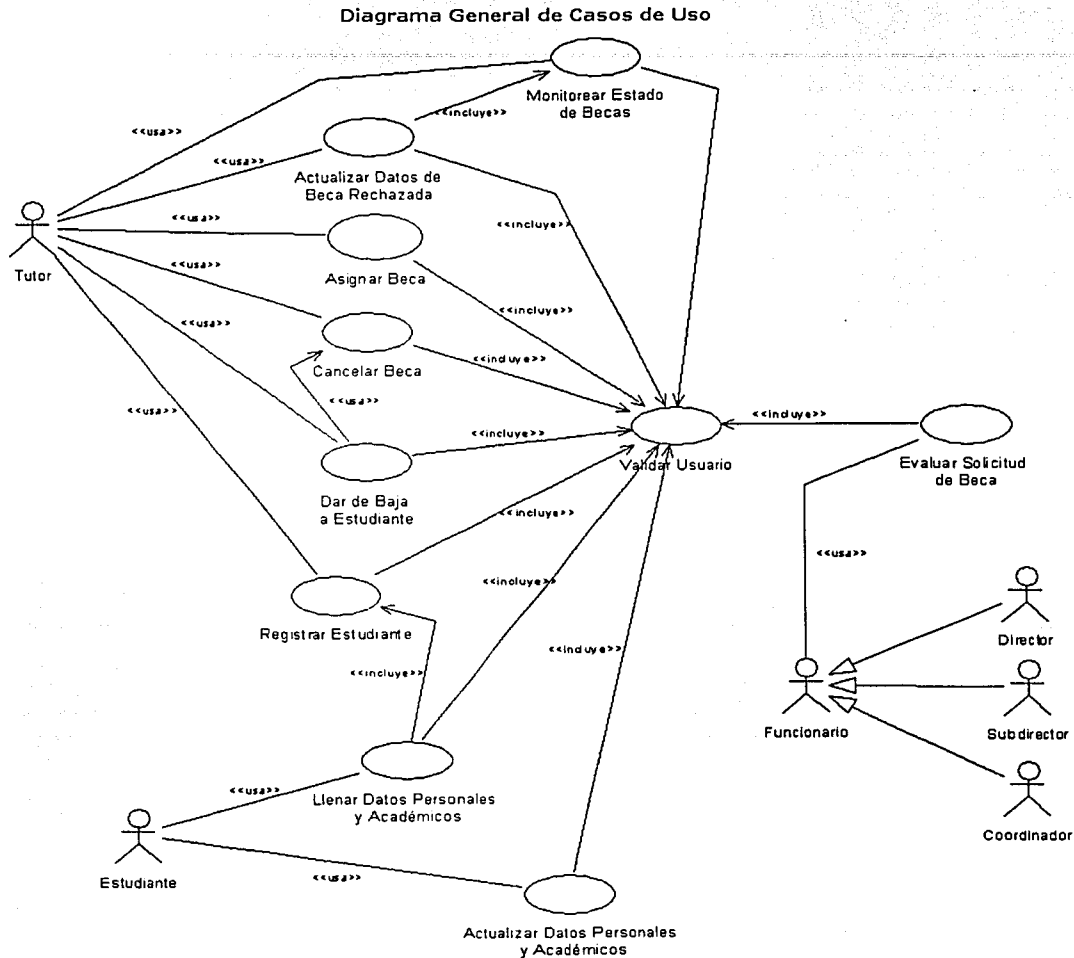
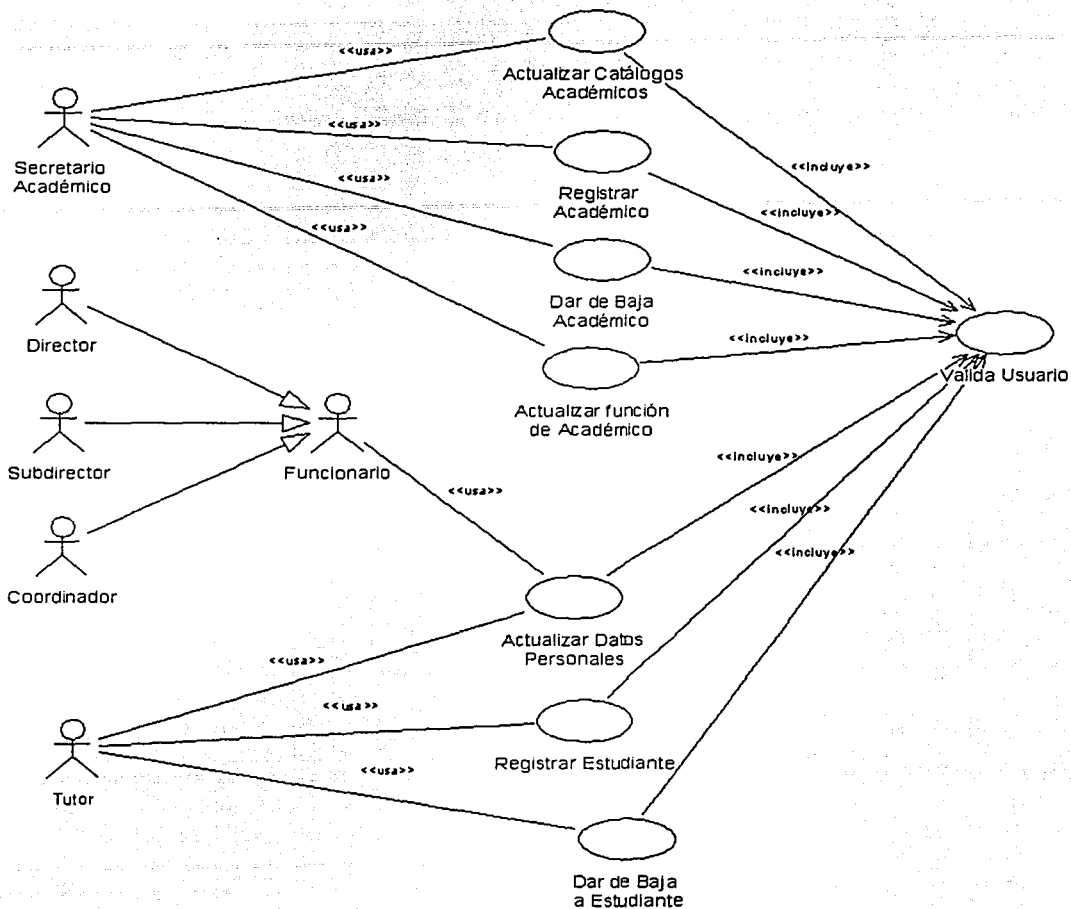


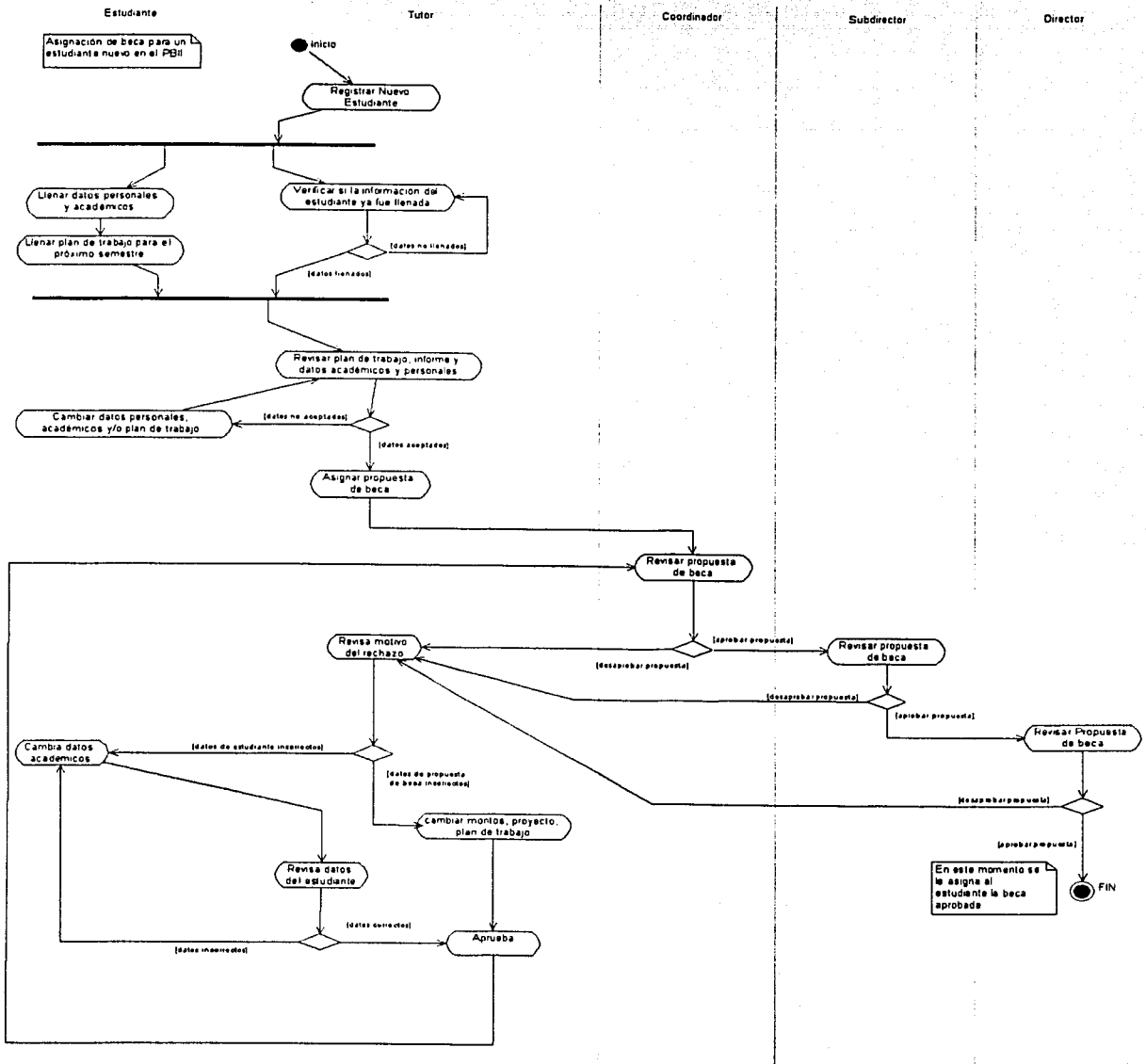
Diagrama de Casos de Uso Administrativos



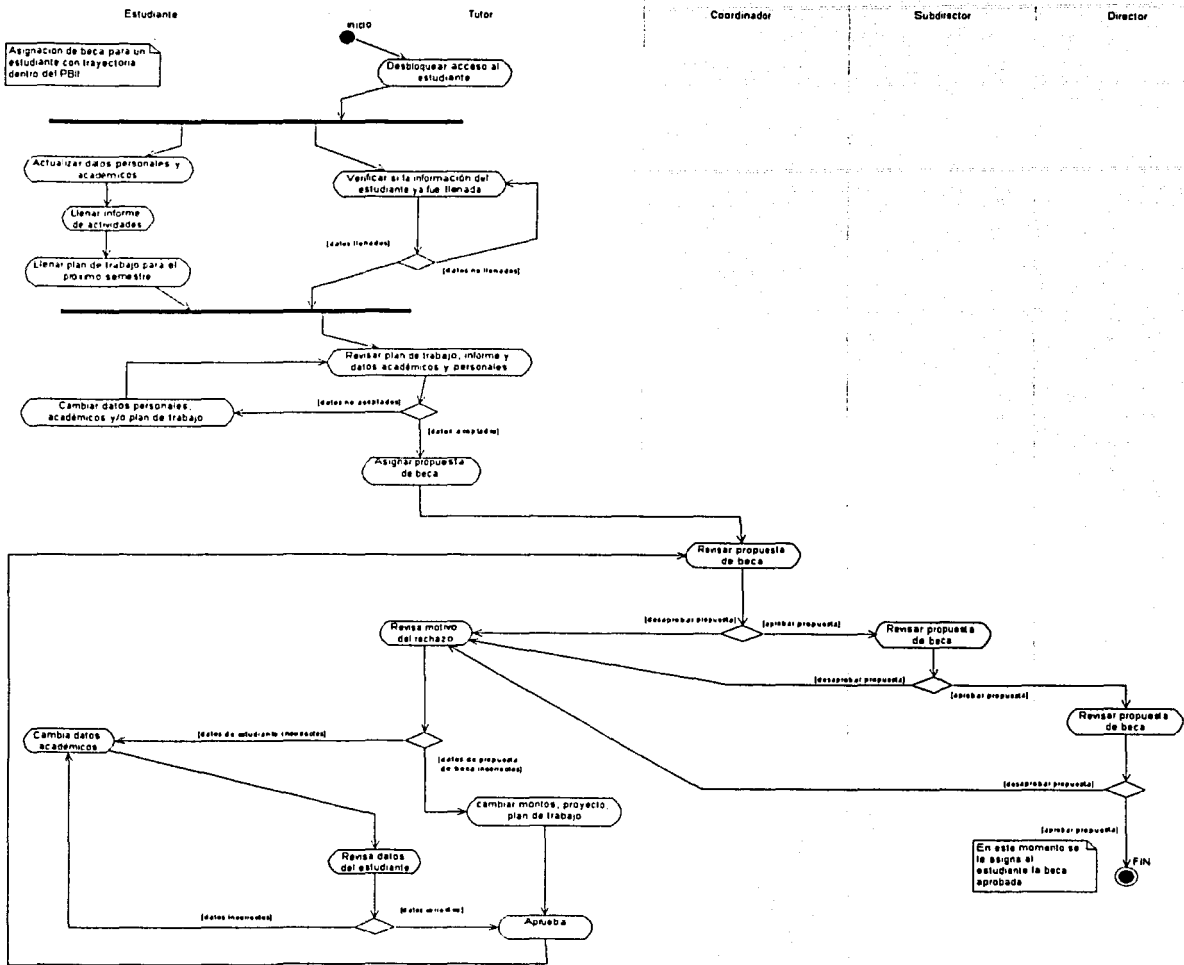
### 5.3.3 Diagramas de Actividades.

Un diagrama de actividades muestra el flujo de actividades de un sistema. Una actividad muestra un conjunto de actividades, el flujo secuencial o ramificado de actividades, y los objetos que actúan y sobre los que se actúa. Los diagramas de actividades se utilizan para ilustrar la vista dinámica de un sistema.

A continuación se muestran los diagramas de actividades del SCBII.







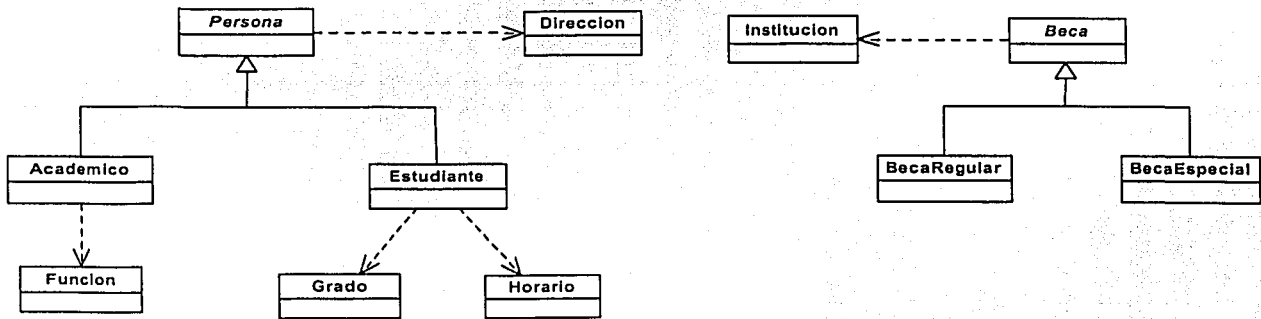
## 5.4 Diseño del Sistema.

### 5.4.1 Diagramas de Clases.

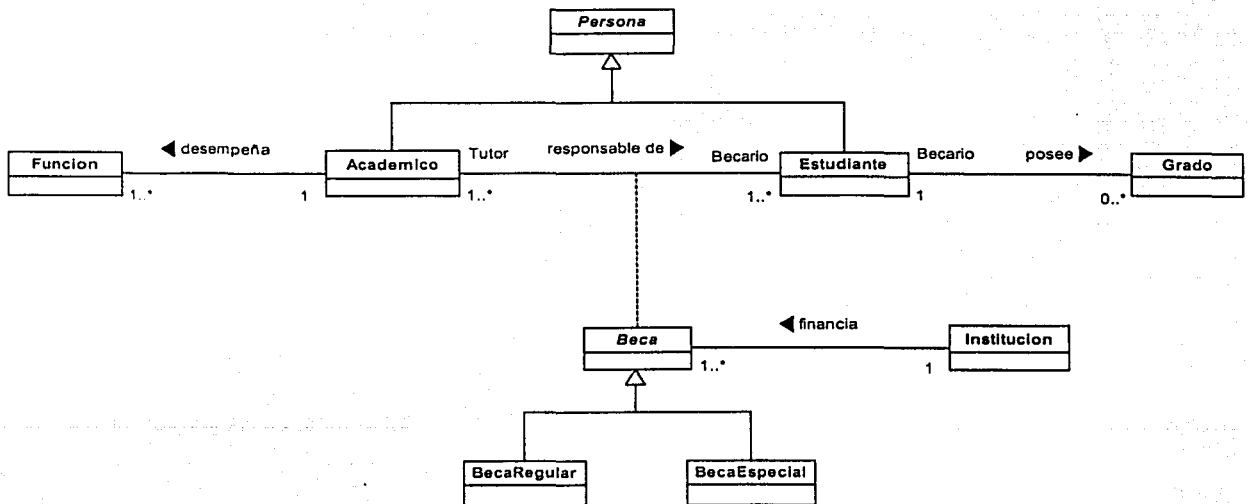
Un diagrama de clases presenta un conjunto de clases, interfaces y colaboraciones, y las relaciones entre ellas. Los diagramas de clases son los diagramas más comunes en el modelado de sistemas orientados a objetos. Los diagramas de clases se utilizan para describir la vista de diseño estática de un sistema.

En este apartado se muestran los diagramas de clases del SCBII. Asimismo se describen las características de las clases entidad más importantes. En el manual técnico, se incluyen el resto de las clases entidad, las clases de utilidad y las clases de control.

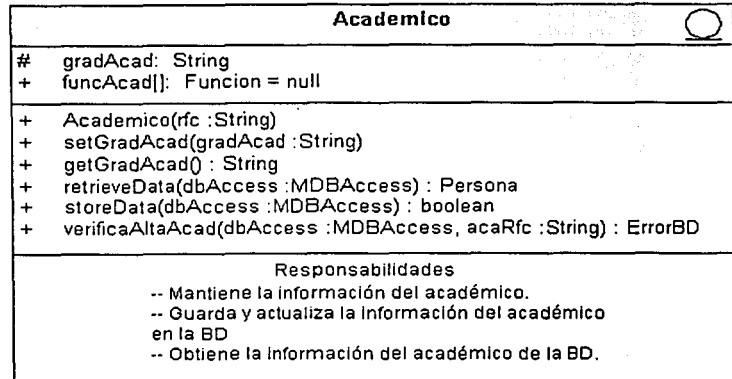
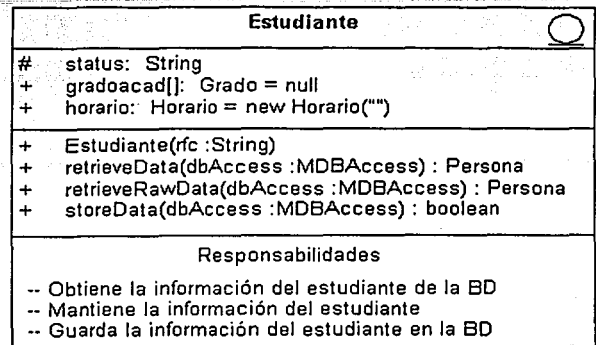
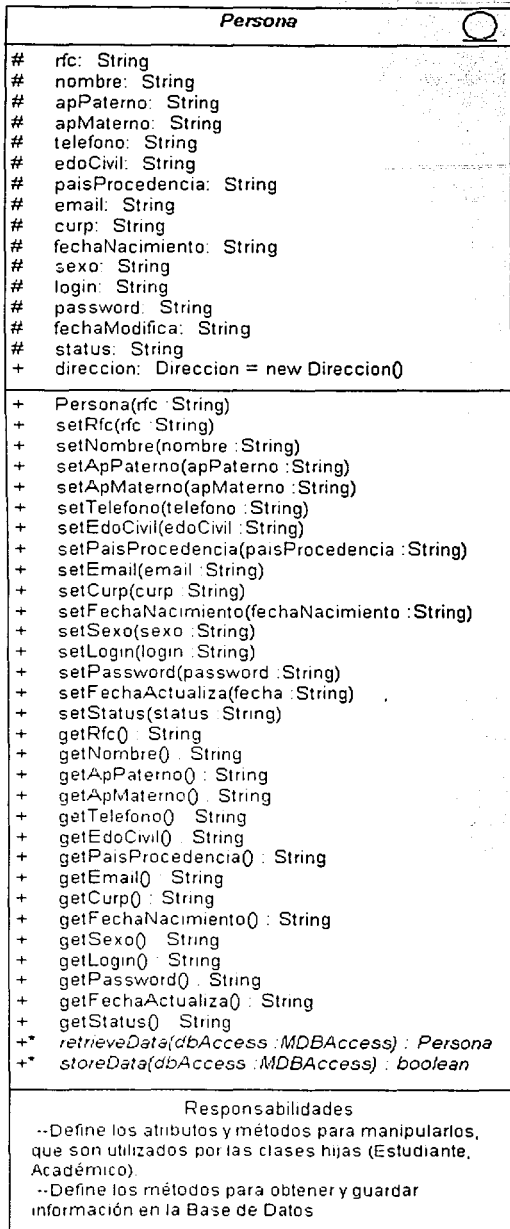
a) Diagrama de dependencia y generalización de las clases entidad.

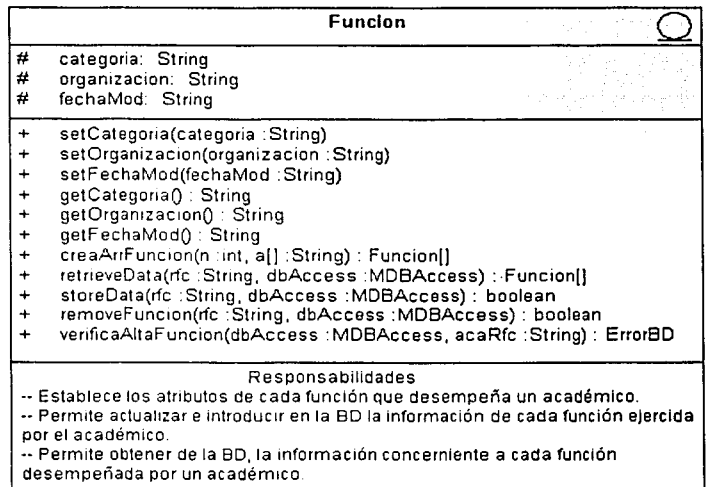
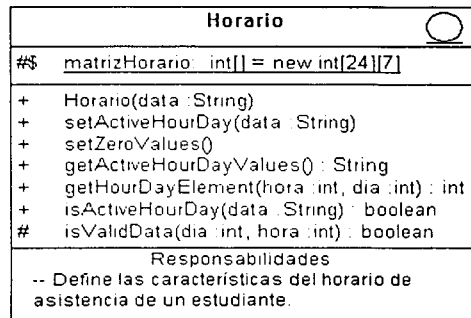
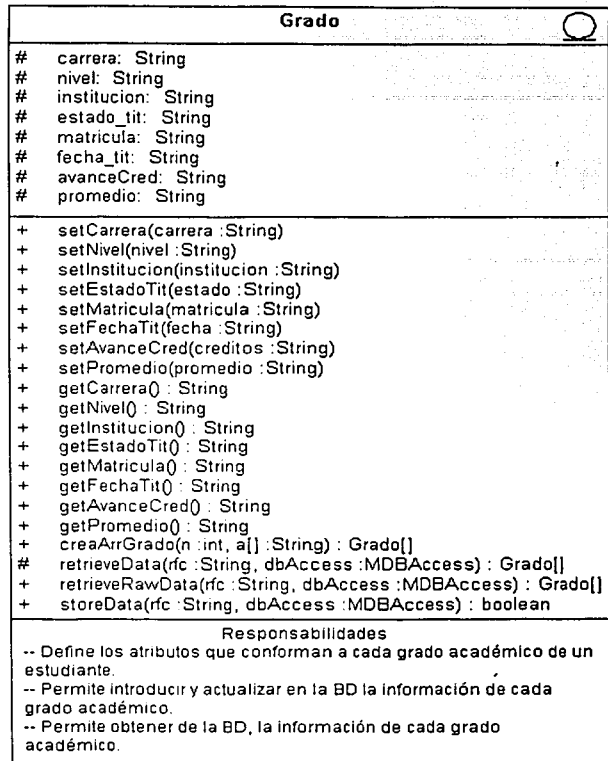
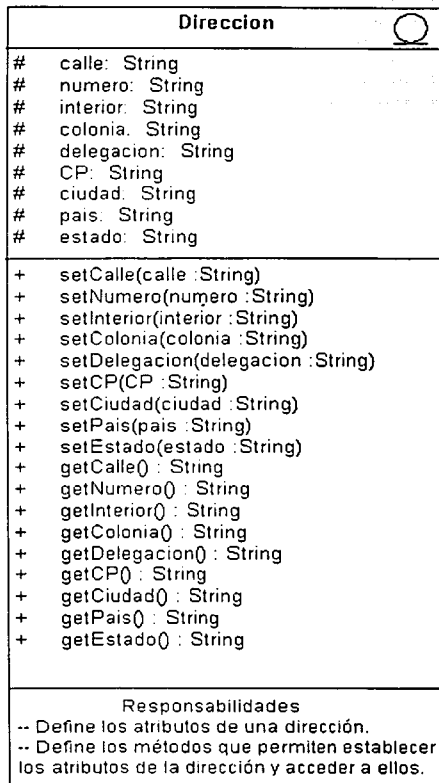


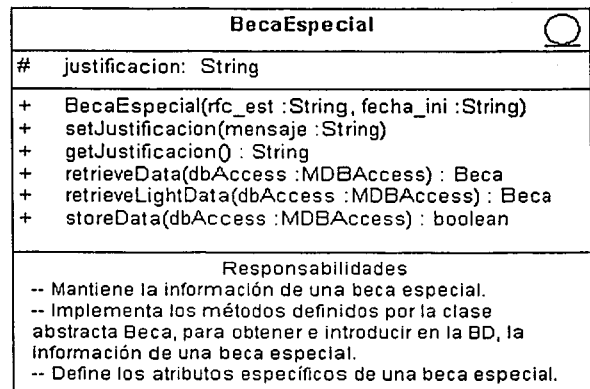
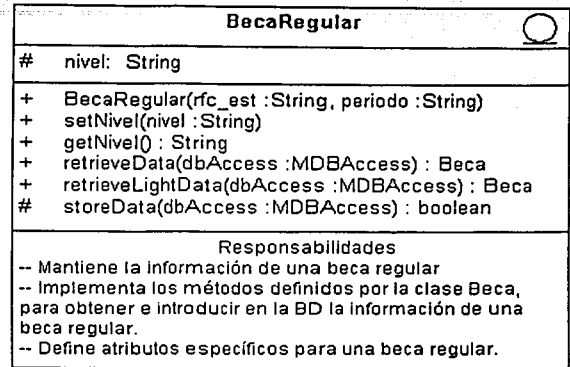
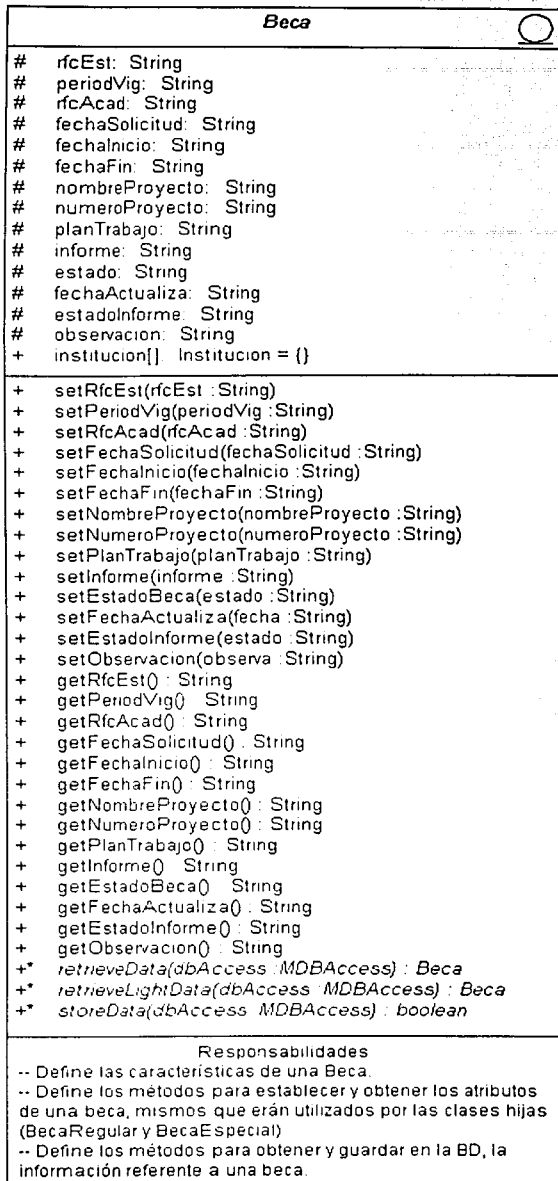
b) Diagrama de asociación de las clases entidad.

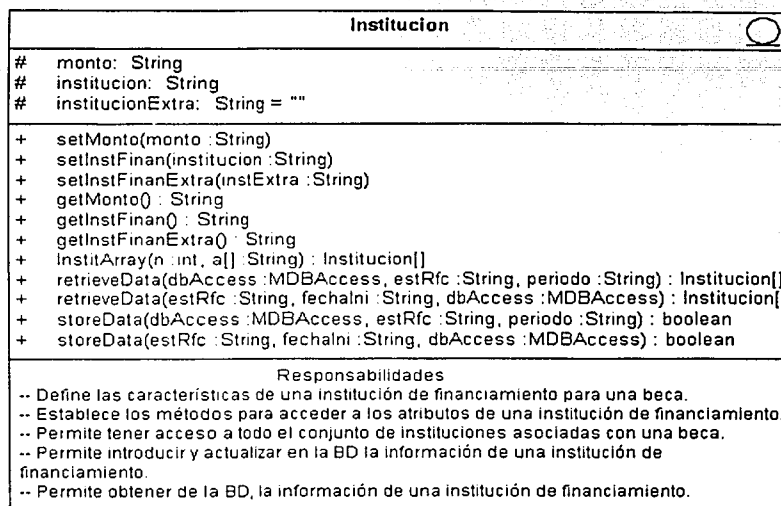


c) Atributos, métodos y responsabilidades de las clases entidad.







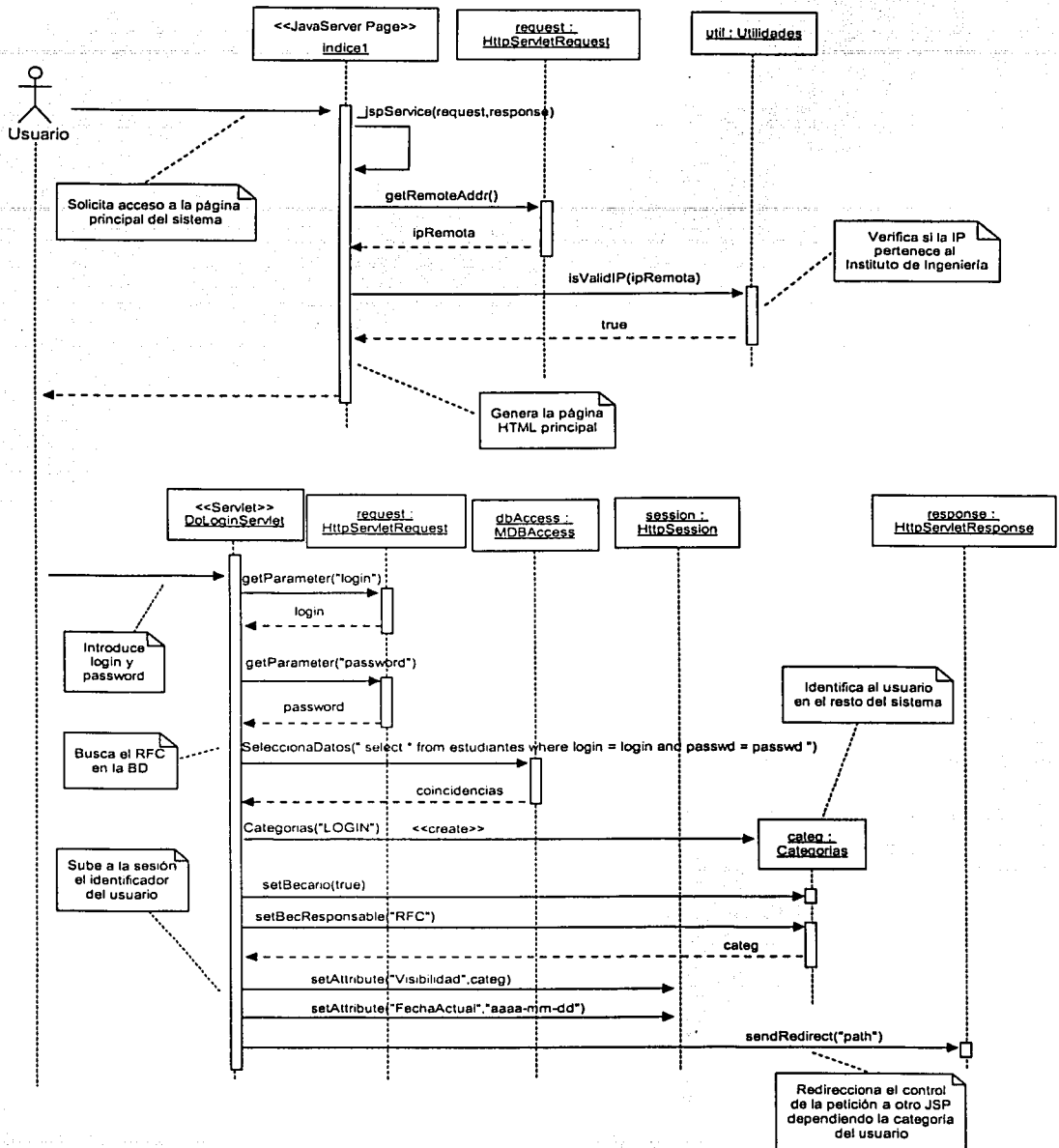


#### 5.4.2 Diagramas de Secuencia.

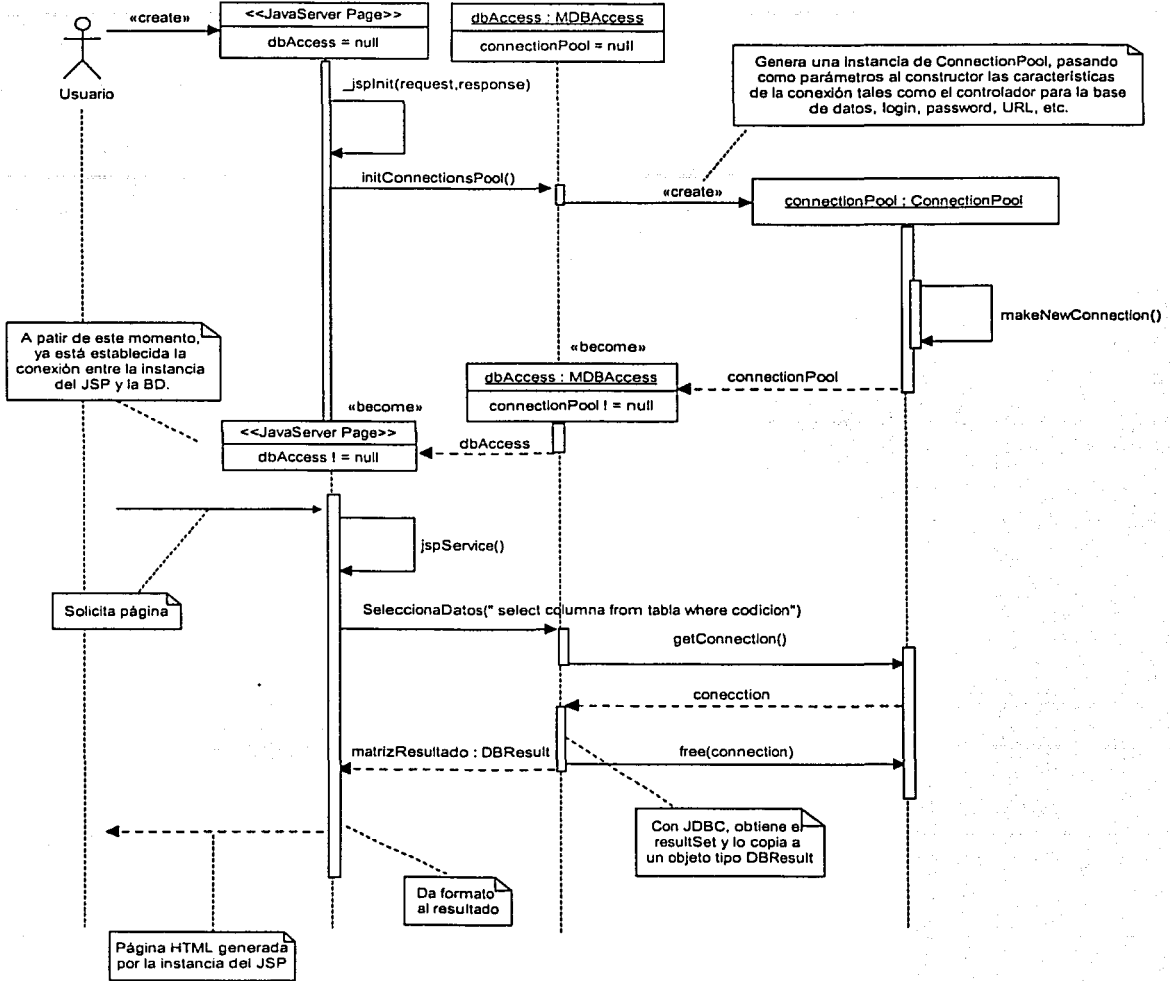
Un diagrama de secuencia es un diagrama de interacción que resalta la ordenación temporal de los mensajes. Un diagrama de secuencia presenta un conjunto de objetos y los mensajes enviados y recibidos por ellos. Los objetos suelen ser instancias con nombres o anónimas de clases, pero también pueden representar instancias de otros elementos, tales como colaboraciones, componentes y nodos. Los diagramas de secuencia se utilizan para describir la vista dinámica de un sistema.

En este apartado se muestran 3 diagramas de secuencia, el resto de los diagramas del sistema se incluyeron en el manual técnico.

Caso de Uso: Validar Usuario

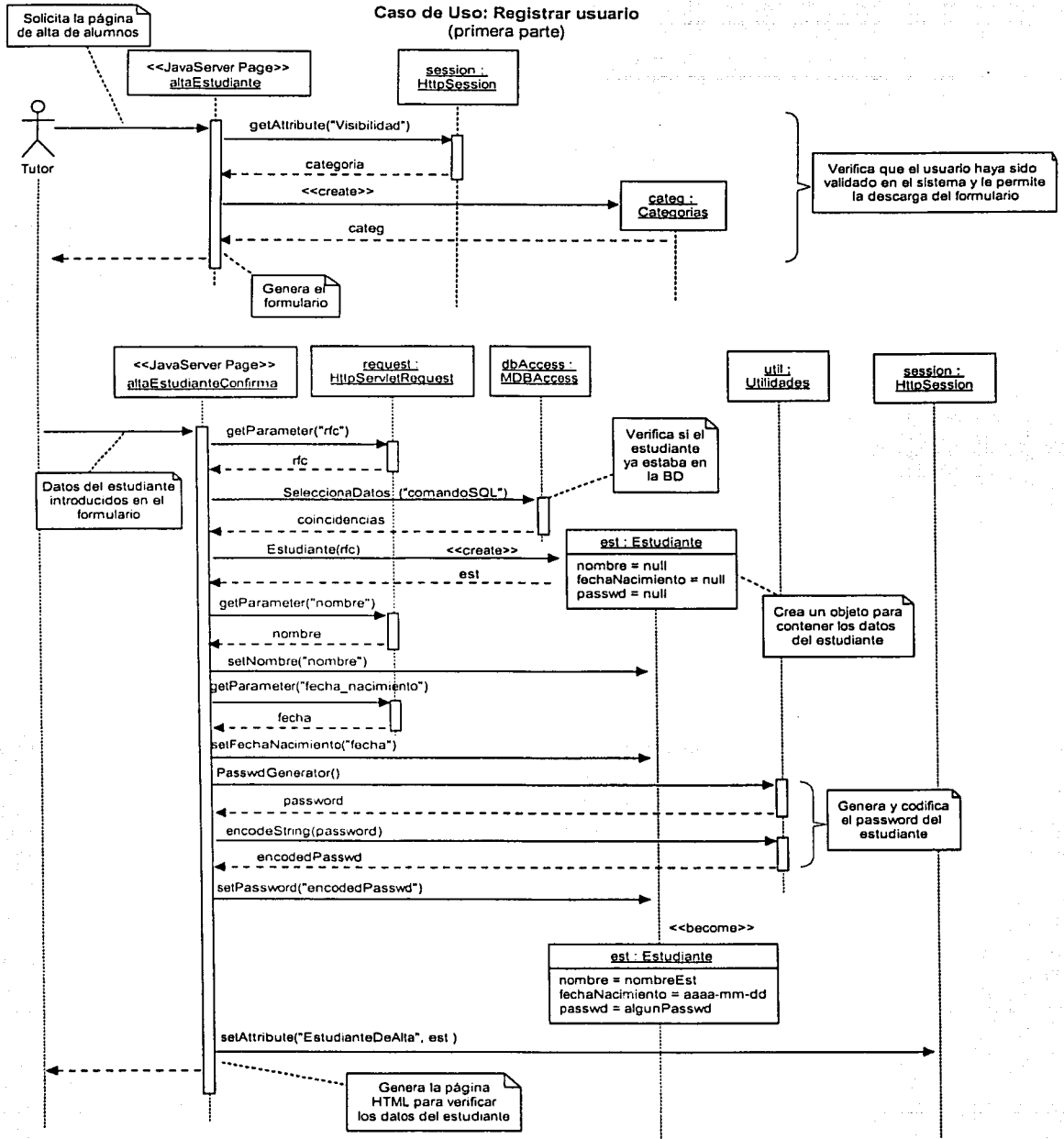


**Conexión directa de cualquier JSP con la BD**  
 (desde la creación de las instancias involucradas hasta la utilización de la conexión)



Nota: El proceso de generar una conexión nueva se ejecuta únicamente cuando un JSP es cargado por el contenedor de JSP's (cuando se ejecuta el jspInit). La conexión generada es reutilizada por todas las peticiones que el JSP maneje.





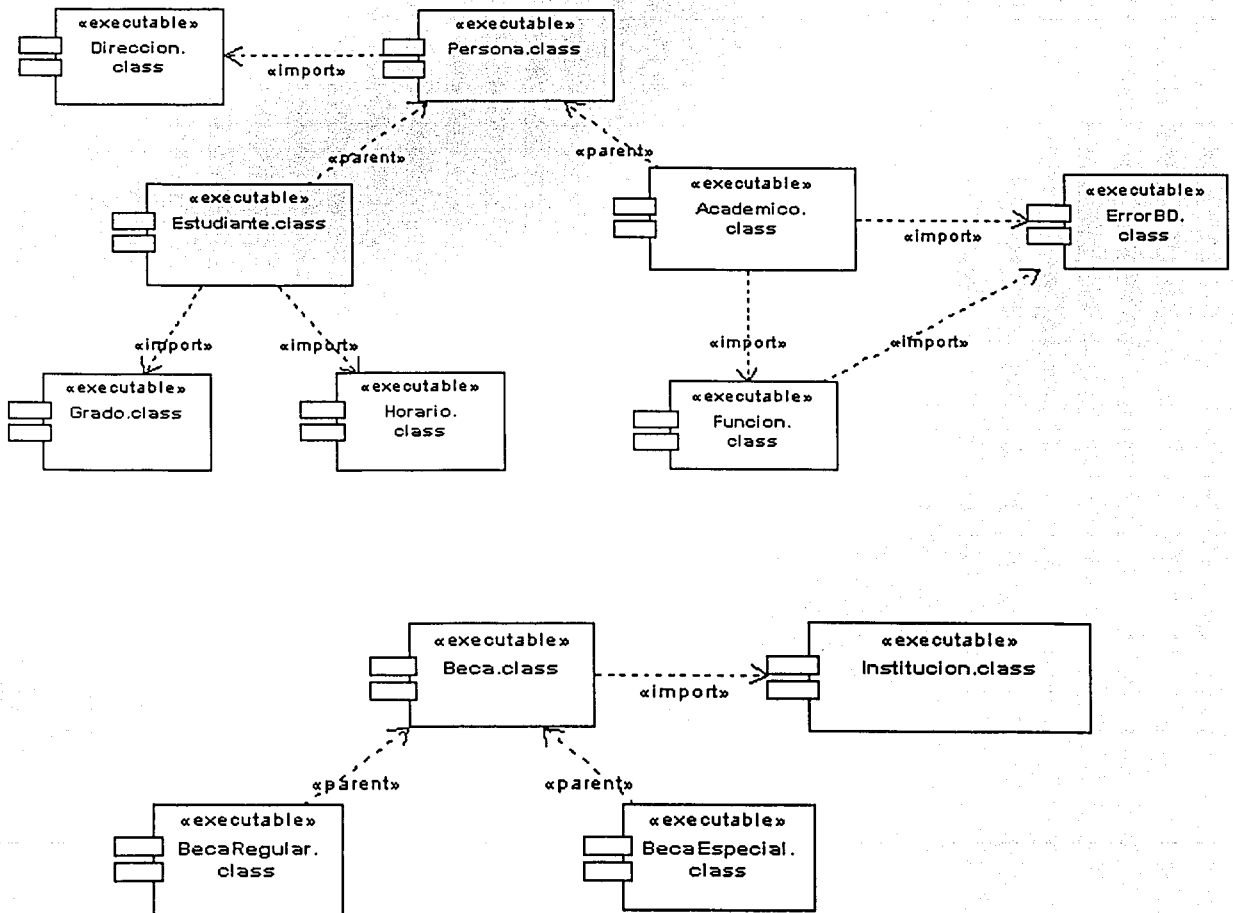
## 5.5 Implementación del Sistema.

### 5.5.1 Diagramas de Componentes.

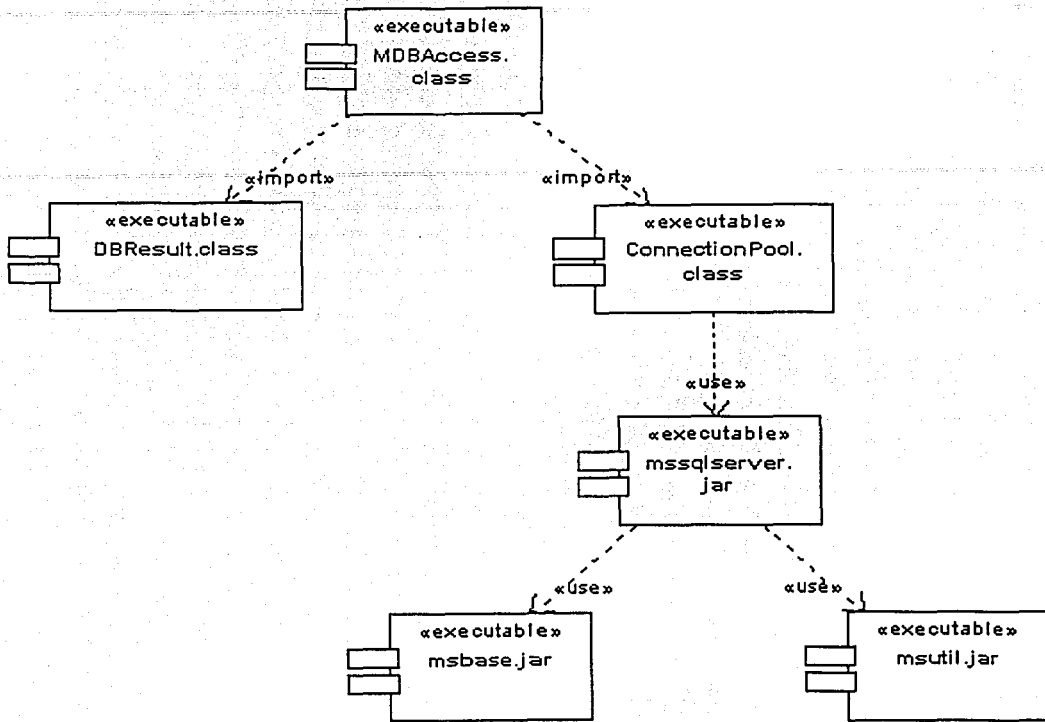
Un diagrama de componentes muestra un conjunto de nodos y sus relaciones. Los diagramas de componentes se utilizan para describir la vista de implementación estática de un sistema. Los diagramas de componentes se relacionan con los diagramas de clases en que un componente normalmente se corresponde con una o más clases, interfaces o colaboraciones.

En este apartado se muestran algunos de los componentes relacionados con las clases entidad, las clases de control y los Servlets de Java.

#### a) Componentes relacionados con las clases entidad.



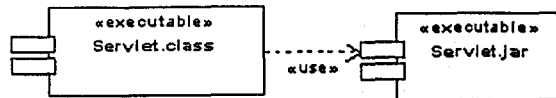
**b) Componentes relacionados con las clases de control.**



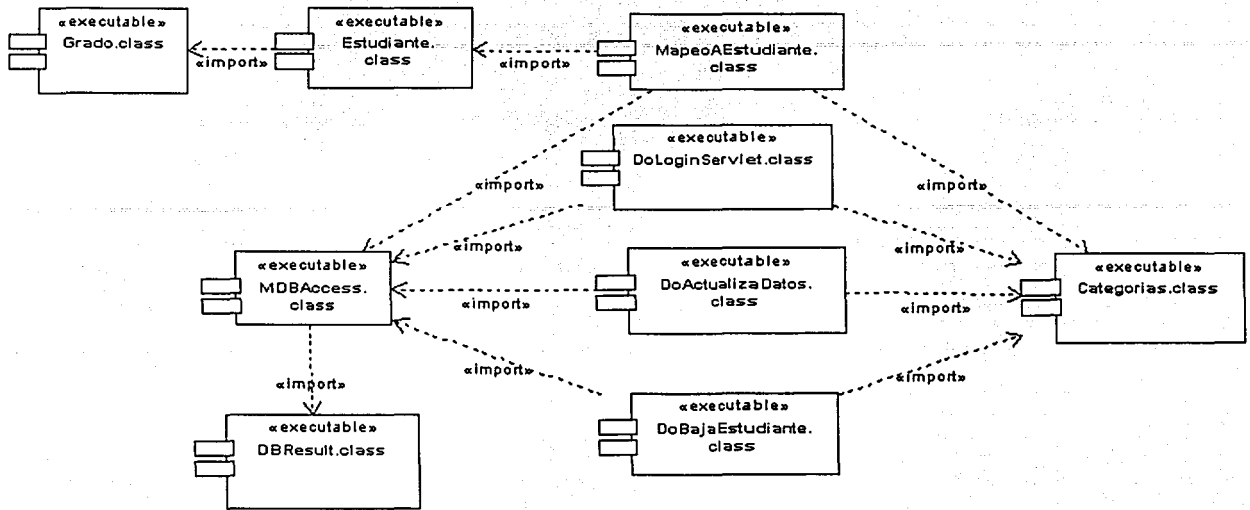
**c) Componentes de los Servlets de Java.**

Diagrama General.

Diagrama General para todos los Java Servlets utilizados en el código del sistema



## Diagrama de Servlets.



# CAPÍTULO SEIS

## Pruebas y Resultados Finales

## **6. Pruebas y Resultados Finales.**

### **6.1 Descripción y resultados de las pruebas realizadas al sistema.**

Los resultados que se describen en este apartado se obtuvieron principalmente de un simulacro de inscripción de becas realizado por los integrantes de la Coordinación de Sistemas de Cómputo. La función de los usuarios durante dicho simulacro fue utilizar el sistema sin tener algún conocimiento del mismo. Como candidatos a becarios participaron los estudiantes de todas las áreas que integran la coordinación, como tutores intervinieron los jefes de área y las funciones de coordinador, de subdirector y de director las desempeñó el responsable de la coordinación.

Los aspectos a evaluar variaron desde la funcionalidad del sistema, hasta el nivel de facilidad de uso y entendimiento de la interfaz del usuario.

A continuación, se muestran los resultados obtenidos para cada una de las pruebas aplicadas.

#### **6.1.1 Pruebas de funcionalidad.**

Las pruebas de funcionalidad se realizan para verificar que las aplicaciones de software cumplan con los requerimientos de los clientes. Este tipo de pruebas pueden ser guiadas por los casos de uso obtenidos en las etapas tempranas del ciclo de vida del sistema y pueden ser realizadas por los usuarios finales. Lo anterior se debe a que para utilizar el sistema, no es necesario conocer la forma en la que está estructurado y la manera en que fue programado.

Los casos de uso que se probaron fueron los siguientes:

- Registrar Estudiante.
- Llenar datos personales y académicos.
- Asignar beca.
- Evaluar solicitud de beca.
- Registrar académico.
- Monitorear Estado de Becas.

En la realización de un escenario particular del caso de uso "Llenar datos personales y académicos", se detectó un error de validación en el campo de porcentaje de créditos de los grados académicos. Este error se presentaba al intentar agregar un formulario para registrar un grado académico.

A excepción del error antes mencionado, el cual fue corregido y probado con resultados exitosos, cada caso de uso cumplió con el objetivo para el cual había sido programado.

Algunas de las variaciones de las pruebas de funcionalidad comprenden a las *Pruebas de Regresión* y a las *Pruebas de Navegación*.

### **6.1.2 Pruebas de regresión.**

Las pruebas de este tipo verifican que los cambios que se le hagan a una parte del código del sistema no afecten el funcionamiento del resto.

Durante el proceso de desarrollo del sistema, se hicieron diversas pruebas de este tipo debido a que los requerimientos fueron modificados a lo largo del proyecto y fue necesario hacer múltiples cambios en el código del sistema, no solamente en los archivos relacionados con la interfaz del usuario (JavaServer Pages), sino también en los archivos de clases y en la base de datos.

Las modificaciones más representativas hechas al sistema fueron las siguientes:

1. Inicialmente el sistema permitía asignar una beca a un estudiante aun cuando éste no hubiera llenado el plan de trabajo. Posteriormente se modificó esta característica para que en caso de que el candidato a becario no presente la propuesta de su plan de trabajo, el sistema no permita que el tutor promueva una beca.
2. En una versión temprana del sistema se contemplaba que el monto total de una beca regular, lo constituía la suma de las cantidades proporcionadas por las instituciones externas y por el PBII. Actualmente, se considera que el monto total de la beca es proporcionado por el fondo de becas del PBII, debido a que el dinero asignado por las instituciones externas se deposita dentro de este mismo fondo.
3. Después de haber terminado la primera fase de desarrollo del proceso de aprobación de una beca, surgieron nuevos requerimientos de los cuales se destaca la agregación de nuevos campos para incluir motivos de rechazo. Asimismo, se solicitó que para la aprobación de una beca, se pudieran escribir observaciones en un campo de texto libre.

Además de las modificaciones citadas existieron otras con un impacto menor sobre la estructura del sistema, tales como la creación de menús, la distribución de los formularios de los datos académicos y personales de los estudiantes en diferentes páginas y la agregación de campos de carácter informativo, entre otros.

Para cada uno de los aspectos descritos, se verificó que la funcionalidad del sistema lograda hasta ese momento no se viera afectada por las

modificaciones realizadas, obteniéndose resultados exitosos para todos los casos.

### **6.1.3 Pruebas de navegación.**

Las pruebas de navegación aseguran que todas las ligas dentro del sitio funcionan correctamente.

La realización de estas pruebas consistió en verificar que cada página generada por los JavaServer Pages tuviera las ligas necesarias de navegación, ya sea para regresar a páginas anteriores, enviar datos, abrir o cerrar ventanas secundarias y salir del sistema.

En el caso de las ligas de envío de datos de todos los formularios utilizados se verificó que la información fuera enviada correctamente, ya sea para su manipulación o para su almacenamiento en la base de datos.

Para las ligas responsables de abrir ventanas secundarias que contienen datos tales como el plan de trabajo, el informe de actividades, el horario de asistencia y el historial de becas de cada estudiante, se verificó la congruencia de esta información con la mostrada en la página de origen.

En el caso de la liga para salir del sistema, se verificó que la sesión del usuario realmente adquiriera el estado de inválida.

### **6.1.4 Pruebas de compatibilidad.**

Las pruebas de compatibilidad miden la capacidad del sistema para funcionar correctamente sobre todas las plataformas y configuraciones utilizadas por los usuarios finales. En el caso de las pruebas relacionadas con aplicaciones de web, se verifica la capacidad del sistema para ser desplegado en cada una de las versiones de los diferentes tipos de navegadores. Asimismo, se verifica la compatibilidad del sistema con las diversas arquitecturas de computadoras.

Se hicieron pruebas de compatibilidad con los siguientes navegadores:

- Netscape Navigator Versión 3.04 Gold para Windows 95/NT (liberado en octubre de 1997): Resultó incompatible debido a la antigüedad de la versión de JavaScript que maneja el navegador.
- Netscape Communicator Versión 4.08 para Windows 95/NT (liberado en noviembre de 1998): Resultó incompatible con la versión de JavaScript que maneja.
- Netscape Communicator Versión 4.79 para Windows 95/NT (liberado en noviembre de 2001): Resultó incompatible con la versión de JavaScript que maneja.



- Netscape versión 6.2.2 para Windows 95/NT (liberado en marzo de 2002): Presenta problemas únicamente con una sentencia de JavaScript que no ejecuta el navegador y de la cual dependen las validaciones de los campos de texto de los formularios.
- Internet Explorer versión 4.0 para Windows 95/NT (liberado en octubre de 1997): No presentó problemas.
- Internet Explorer versión 5.0 para Windows 95/NT (liberado en marzo de 1999): No presentó problemas.
- Internet Explorer versión 6.0 para Windows 95/NT/2000/XP (liberado en octubre de 2001): No presentó problemas.

De los resultados anteriores se puede concluir que el sistema es totalmente compatible con Microsoft Explorer. En el caso de Netscape, el mayor grado de compatibilidad se presenta con la última versión del mismo; sin embargo, existen problemas con una sentencia de JavaScript que no permiten que la compatibilidad sea completa. Esta situación podría ser resuelta creando una versión del sistema únicamente para Netscape 6.0, debido a que el resto de las versiones del navegador tienen más de 3 años de antigüedad por lo que se consideran obsoletas.

El uso de Netscape es muy reducido dentro del Instituto, lo que significa que por el momento la creación de una versión especial para este navegador no es necesaria, sobretodo porque retrasaría la liberación del sistema.

#### **6.1.5 Pruebas de utilización.**

Las pruebas de utilización se encargan de evaluar que tan intuitiva y fácil de navegar es la interfaz del usuario. Por esta razón, a las personas involucradas no se les dio alguna explicación previa del uso del sistema.

Dentro de los comentarios y observaciones de los usuarios se pueden destacar los siguientes:

- Hacer más explícita la forma en la que se deben de agregar los datos académicos de los estudiantes.
- Establecer una clara diferencia entre los campos de datos y sus valores.
- Hacer más evidente la última oportunidad que tienen los usuarios para cambiar los datos antes de enviarlos.
- Especificar el tipo y el formato de los datos que deben de recibir algunos campos.

**FALTA  
PAGINA**

**107**

**FALTA  
PAGINA**

**108**

7	$3 < t < 4$	- Evaluación de los datos del estudiante para asignar una beca. Este proceso lo realizan únicamente los tutores.
---	-------------	--

*Nota: La variable "t" representa el tiempo transcurrido desde que el cliente solicita la página hasta que ésta es desplegada completamente en el navegador*

Las páginas utilizadas por los estudiantes son las más solicitadas al sistema, y se ubican dentro del segmento de páginas cuyo tiempo de respuesta varía entre 1 y 3 segundos.

Las páginas que corresponden al 7% son aquellas que contienen los datos de los estudiantes y la información asociada a las becas, tal y como el periodo de vigencia, el monto, el plan de trabajo y el informe de actividades. A este segmento del sistema solamente tienen acceso los tutores, los coordinadores, los subdirectores y el director, por lo que se consideran de uso menor.

Los resultados obtenidos nos permitieron observar que los tiempos de respuesta del sistema, son directamente proporcionales a la cantidad de información solicitada a la base de datos.

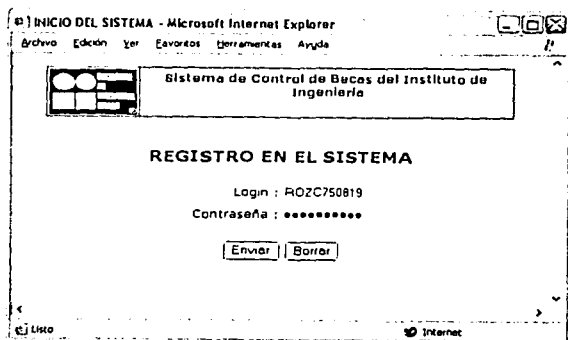
Además de las pruebas relacionadas con el tiempo de respuesta del sistema, se realizó una prueba de carga de información en la cual se detectaron errores de inserción de datos, particularmente en el caso de la introducción de planes de trabajo con un tamaño aproximado de 7 hojas. Este problema se debió a que el manejador de la base de datos tenía dificultades al tratar de reemplazar la información de algunas llaves primarias, al mismo tiempo que intentaba actualizar un campo de texto con la magnitud citada. Para resolver este problema, se dividió la acción en dos partes y se obtuvieron los resultados esperados originalmente.

## 6.2 Descripción del sistema terminado.

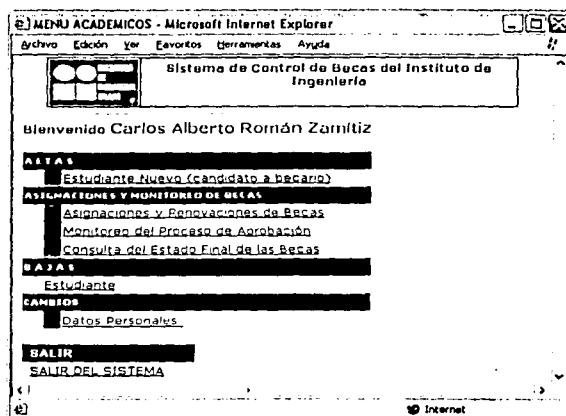
En este apartado se describen algunos de los procesos implementados por el sistema, tales como el registro de un estudiante nuevo y sus datos académicos y personales, la asignación de una beca y el proceso de aprobación de la misma.

### 6.2.1 Registro de un estudiante nuevo.

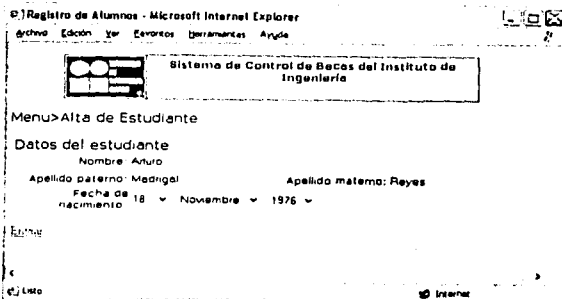
A continuación se muestran las pantallas correspondientes al registro de un estudiante de nuevo ingreso al PBII. Esta labor es realizada por el tutor correspondiente.



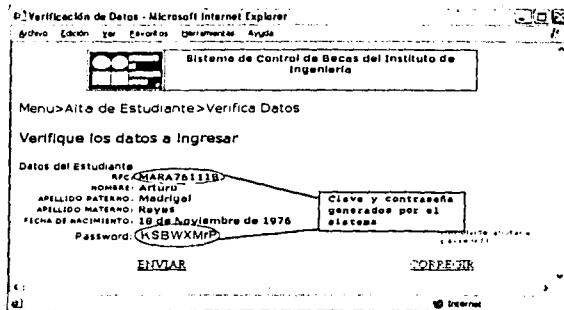
1. El tutor se identifica en el sistema.



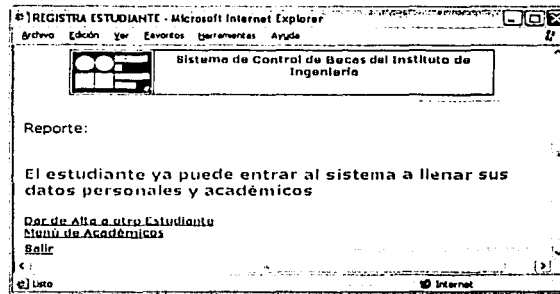
2. En el menú para académicos, el tutor selecciona la opción de Altas/Estudiante Nuevo.



3. El tutor introduce el nombre completo y la fecha de nacimiento del estudiante.



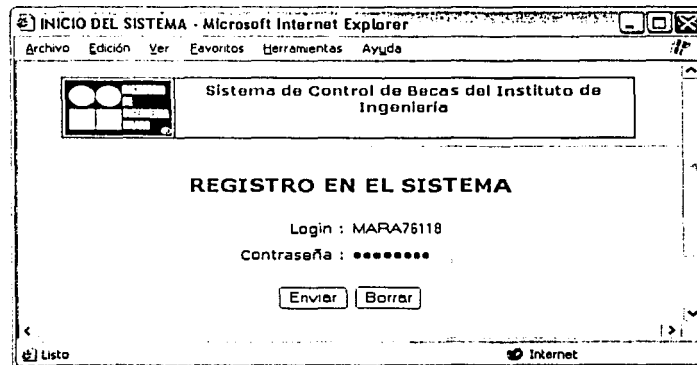
4. El sistema muestra los datos para que sean verificados por el tutor.



5. Después de enviar los datos, el sistema muestra el resultado del proceso de registro del estudiante.

### 6.2.2 Registro de los datos de un estudiante.

A continuación se muestran las pantallas correspondientes al registro de los datos personales y académicos de un estudiante, así como su plan de trabajo y su horario de asistencia al Instituto.



1. El estudiante se identifica en el sistema.

TECIS CON  
FALLA DE ORIGEN

Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

**Sistema de Control de Becas del Instituto de Ingeniería**

Introducir datos personales

**Datos Personales**

RFC: MARA761118 CURP: \_\_\_\_\_  
 Nombre: Arturo  
 Apellido Paterno: Madrigal Fecha de nacimiento: 18 de Noviembre de 1976  
 Apellido Materno: Reyes País de origen: MEXICO  
 Teléfono de casa: 55555555 Sexo: Masculino  
 Estado civil: Soltero  
 e-Mail: arturo@hotmail.com

Dirección:  
 Calle: Constitución Número: 256 Interior: \_\_\_\_\_  
 Colonia: Lomas Estrella Delegación ó municipio: Itzapalapa  
 Ciudad: México Estado: DF  
 CP: 06540

CONTINUAR  
 SALIR

Inicio Internet

2. Introduce sus datos personales.

http://127.0.0.1:8080/IntroDatosAcademicos.jsp?runend=showSecondSchoolTab&idcurp=81a1a&idtelefono=55355 - Mic...

Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

**Sistema de Control de Becas del Instituto de Ingeniería**

Introducir datos académicos

**Datos Académicos**

Universidad: UNAM Nivel: Licenciatura  
 Carrera: Ingeniería en Computación Matricula: 093564659  
 Promedio: 83.5 (1-100) % de avance: 100 %  
 fecha titulación: Día Mes Año  
 Estado: Carrera finalizada

<< SELECCIÓN PARA REGISTRAR LOS DATOS ANTERIORES

Universidad: ITESM Nivel: Maestro  
 Carrera: Ciencias de la Computación Matricula: MAS672001  
 Promedio: 90.5 (1-100) % de avance: 25 %  
 fecha titulación: Día Mes Año  
 Estado: Carrera en curso

<< SELECCIÓN PARA AÑADIR LOS DATOS ANTERIORES

QUIERE REGISTRAR LOS DATOS DE OTRA INSTITUCIÓN?

CONTINUAR  
 CORREGIR DATOS PERSONALES  
 SALIR

Inicio Internet

3. Introduce sus datos académicos.

llenar Datos - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

**Sistema de Control de Becas del Instituto de Ingeniería**

Caso 1. Llenar información para el siguiente periodo de becas.

Horario de Asistencia al Instituto de Ingeniería						
hora	Lun	Mar	Mie	Jue	Vie	Sab
7-8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8-9	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9-10	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10-11	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11-12	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
12-13	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
13-14	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
14-15	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
15-16	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
16-17	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
17-18	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
18-19	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19-20	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20-21	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Plan de Trabajo**

Durante el siguiente periodo de becas, voy a ser parte del equipo de desarrollo de servidores UNIX, trabajare en la administración de los servidores de correos y cuentas de los usuarios del Instituto.

CONTINUAR  
REGRESAR  
SALIR

Internet

4. Introduce el horario de asistencia al Instituto y el plan de trabajo.



VERIFIQUE SUS DATOS

**Datos Personales**

RFC: MARA761110 CURP: \_\_\_\_\_  
 Nombre: Arturo  
 Apellido Paterno: Madrigal Apellido Materno: Reyes  
 Teléfono: 55553555 País de origen: MEXICO  
 Estado civil: Soltero Sexo: Masculino  
 Fecha de nacimiento: 18 de Noviembre de 1976 e-Mail: arturo@hotmail.com

**Dirección**

Calle: Constitución Número: 256 Interior: \_\_\_\_\_  
 Colonia: Lomas Estrella Delegación: Iztapalapa  
 Ciudad: Mexico Estado: D.F.  
 CP: 08540

**Datos Académicos**

Universidad: UNAM Grado: Licenciatura  
 Carrera: Ingeniería en Computación Matrícula: 093564659  
 Promedio: 93.5 % de avance: 100  
 fecha titulación: desconocida Estado: Carrera finalizada

Universidad: ITEBM Grado: Maestría  
 Carrera: Ciencias de la Computación Matrícula: MA3672001  
 Promedio: 90.5 % de avance: 25.4  
 fecha titulación: desconocida Estado: Carrera en curso

**Horario de Asistencia**

Hora	Lun	Martes	Jue	Vie
7 - 8				
8 - 9				
9 - 10	X	X	X	X
10 - 11	X	X	X	X
11 - 12				
12 - 13	X	X	X	X
13 - 14	X	X	X	X
14 - 15				
15 - 16	X	X	X	X
16 - 17	X	X	X	X
17 - 18	X	X	X	X
18 - 19				
19 - 20				
20 - 21				

**Periodo de Trabajo**

Durante el siguiente periodo de becas, voy a ser parte del equipo de desarrollo de servidores UNIX, trabajaré en la administración de los servidores de correos y cuentas de los usuarios del Instituto.

ENVIAR      COPIAR

5. El sistema muestra los datos introducidos por el alumno para su verificación.

REGISTRA ESTUDIANTE - Microsoft Internet Explorer

Sistema de Control de Becas del Instituto de Ingeniería

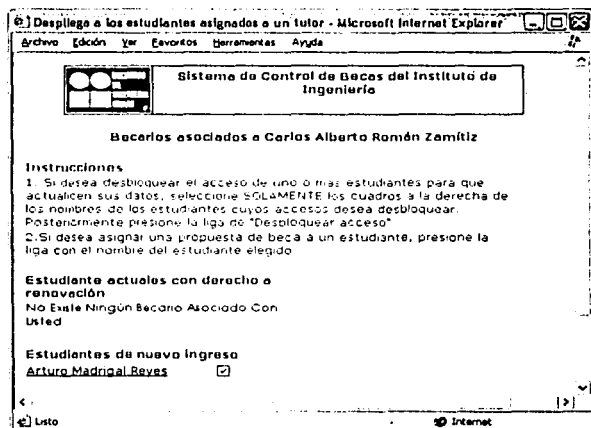
Reporte:

- Recuerde verificar el estado de su beca
- Su acceso ha sido bloqueado
- La actualización de sus datos se realizó con éxito

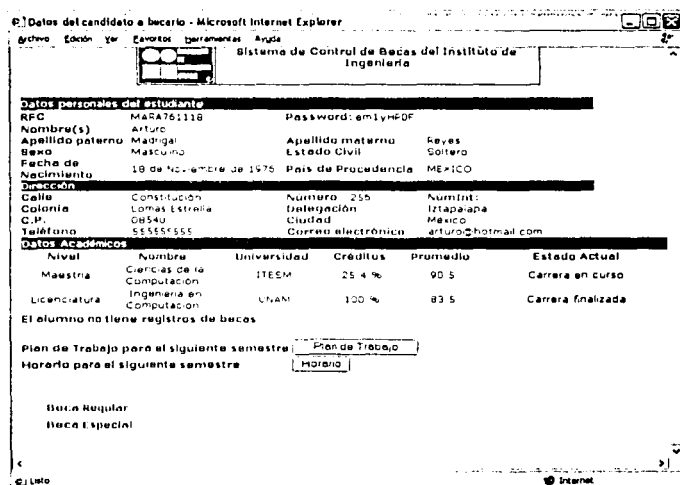
6. El sistema muestra el reporte del proceso de la introducción de los datos del estudiante.

### 6.2.3 Asignación de una beca.

El tutor se identifica en el sistema. En el menú de académicos, previamente mostrado, el tutor selecciona la opción de "Asignaciones y Renovaciones de Becas". A continuación se despliega la siguiente pantalla:




1. El tutor selecciona un estudiante nuevo para asignarle una propuesta de beca.



2. El sistema despliega los datos personales y académicos del estudiante. Asimismo se puede tener acceso al horario de asistencia y al plan de trabajo del estudiante. Posteriormente, el tutor selecciona el tipo de beca que le va a asignar al estudiante, es entonces cuando el sistema muestra la siguiente pantalla.

2) Datos del candidato a becarlo - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

 Sistema de Control de Becas del Instituto de Ingeniería

**Datos personales del estudiante**

RFC: MARA761118 Password: em1yHPOF  
 Nombre(s): Arturo  
 Apellido paterno: Madrigal Apellido materno: Reyes  
 Sexo: Masculino Estado Civil: Soltero  
 Fecha de Nacimiento: 18 de Noviembre de 1976 País de Procedencia: MEXICO

**Dirección**

Calle: Constitución Número: 256 NúmInt: Iztapalapa  
 Colonia: Lomas Estrella Delegación: Iztapalapa  
 C.P.: 08540 Ciudad: México  
 Teléfono: 55555555 Correo electrónico: arturo@hotmail.com

**Datos Académicos**

Nivel	Nombre	Universidad	Créditos	Promedio	Estado Actual
Maestría	Ciencias de la Computación	ITESM	25.4 %	90.5	Carrera en curso
Licenciatura	Ingeniería en Computación	UNAM	100 %	83.5	Carrera finalizada

El alumno no tiene registros de becas

Plan de Trabajo para el siguiente semestre   
 Horario para el siguiente semestre

**Datos generales de la beca**


Fecha de inicio: 01 Mayo 2002  
 Fecha de fin: 31 Agosto 2002  
 Nombre del Proyecto: Servidores Unix  
 Número de Proyecto: 4563

**Datos de la Beca Regular**

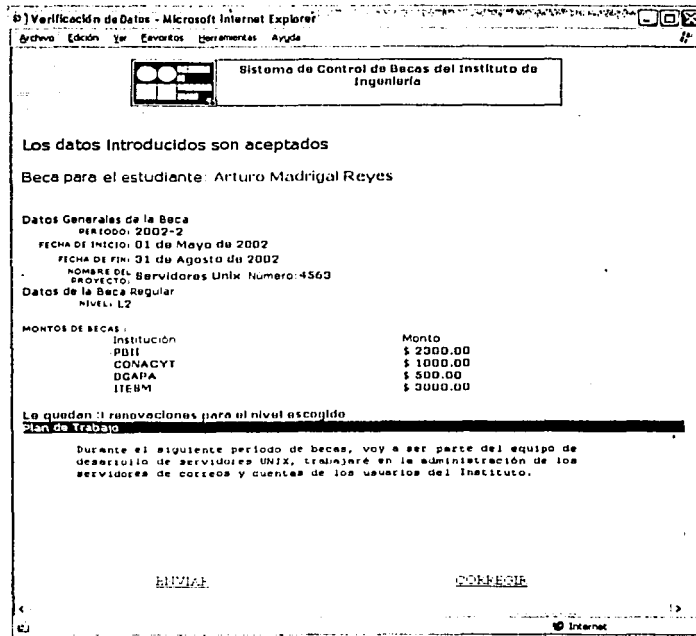
PIII  
 Nivel Descripción Monto  
 L2 - Beca de Licenciatura con tesis - \$2300 00

**Becas externas que tiene el estudiante (No afectan al fondo de Becas)**

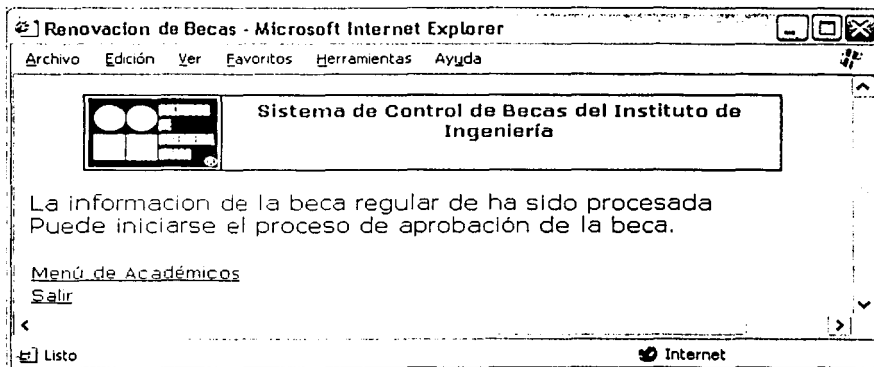
Institución	Monto \$	Acción
CONACYT	1000 .00	<<< <input checked="" type="checkbox"/> Agregar
DGAPA	500 .00	<<< <input checked="" type="checkbox"/> Agregar
Otra Institución ITESM	3000 .00	<<< <input checked="" type="checkbox"/> Agregar

Listo 

3. En esta pantalla se observan los mismos datos personales y académicos mostrados anteriormente. Asimismo, se despliega el formulario correspondiente al tipo de beca seleccionada. En este caso, el tutor escogió asignar una beca regular.



4. El sistema muestra una vista previa de los datos introducidos para que el tutor los verifique. Asimismo, muestra el número de renovaciones a las que el estudiante tiene derecho, en el caso de que su beca regular sea aprobada. En el caso de las becas especiales no se contabilizan las renovaciones.



5. El sistema muestra una pantalla indicando el estado del proceso, después de que el tutor envió la información.

### 6.2.4 Proceso de aprobación de una beca.

En esta sección, se describe brevemente el proceso de aprobación de una beca por parte del coordinador del área correspondiente. La secuencia que se muestra es la misma que se sigue para el caso del subdirector y el director.

Inicialmente el coordinador se identifica en el sistema, enseguida selecciona del menú de académicos la opción de "Revisión de Becas". Posteriormente el sistema despliega la siguiente pantalla:

Manú de Tutores Asociados - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

**Sistema de Control de Becas del Instituto de Ingeniería**

Usted es el coordinador de la Coordinación de Sistemas de Cómputo

Los siguientes tutores están asociados a su coordinación:

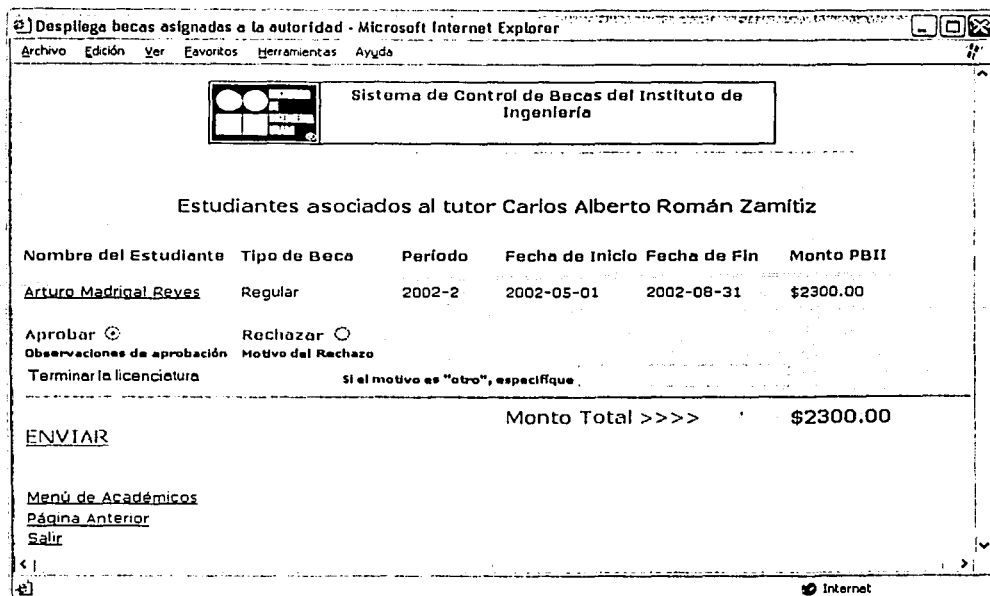
	Becas Regulares Pendientes	Becas Especiales Pendientes
<input checked="" type="checkbox"/> <a href="#">Marco Ambros Maguay</a>	0	0
<input checked="" type="checkbox"/> <a href="#">Luis Alberto Arellano Figueroa</a>	0	0
<input checked="" type="checkbox"/> <a href="#">Javier Enrique Hernández Capilla</a>	3	0
<input checked="" type="checkbox"/> <a href="#">Marcos Huerta Carrizo</a>	0	0
<input checked="" type="checkbox"/> <a href="#">Araceli Martínez Lorenzana</a>	0	0
<input checked="" type="checkbox"/> <a href="#">Guendaviani Rodríguez Benitez</a>	0	0
<input checked="" type="checkbox"/> <a href="#">Carlos Alberto Román Zamita</a>	1	0
<input checked="" type="checkbox"/> <a href="#">Jonathan Velázquez García</a>	0	0

**APROBAR**

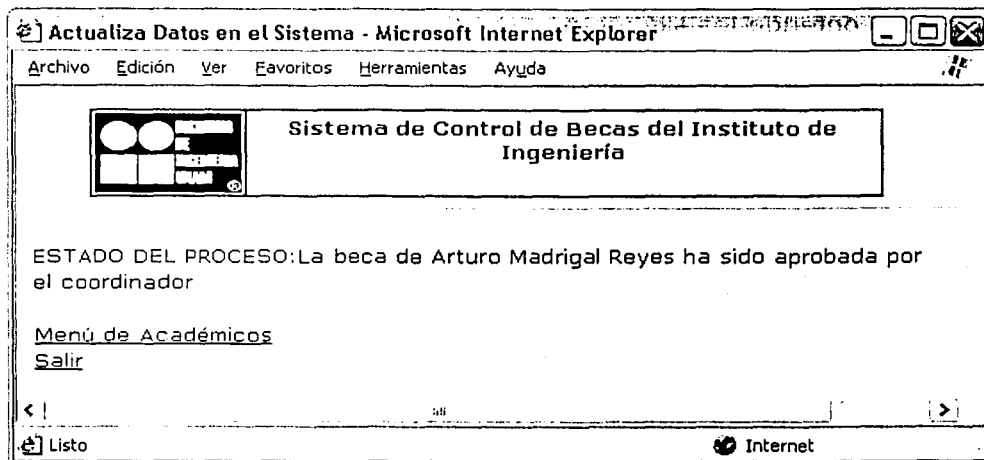
Menú de Académicos  
Salir

Internet

1. En esta pantalla, se muestran a los tutores asociados a la coordinación que preside el académico, así como las becas regulares y especiales pendientes de revisión por cada uno de los tutores. En este caso, el coordinador selecciona al tutor "Carlos Alberto Román" y se despliega la siguiente pantalla.



2. El tutor seleccionado solamente tiene 1 beca regular pendiente de revisión. Se pueden observar los principales datos de la beca, tales como el nombre del estudiante, el tipo de beca, el periodo, la fecha de inicio, la fecha de fin y el monto mensual que destinará el Instituto para pagar la beca. El coordinador decide aprobar la propuesta de beca anotando una observación.



3. El sistema muestra el estado del proceso de evaluación de la beca que acaba de aprobar el coordinador.

### **6.2.5 Información generada para las Secretarías Académica y Administrativa.**

Actualmente la información relacionada con los pagos de los cheques de los becarios se almacena en la base de datos del Sistema de Recursos Humanos (SRHII), que es administrado por la Secretaría Administrativa del Instituto. En estos momentos resulta difícil establecer una interfaz directa entre el Sistema de Control de Becas (SCBII) y el SRHII debido a las razones que se exponen a continuación:

- El SRHII no posee una arquitectura cliente/servidor que proporcione un acceso directo a las tablas de la base de datos para poder manipularlas.
- La información de las tablas del SCBII y las del SRHII no son compatibles, debido a que el primer sistema maneja el aspecto académico del Instituto, mientras que el segundo se ocupa de la parte administrativa.

Cada vez que termina un período de evaluación de becas, los resultados finales se registran manualmente en cada una de las solicitudes y éstas son enviadas a la Secretaría Administrativa. De toda la información que contienen los documentos citados, la Secretaría Administrativa solamente necesita los datos relacionados con los pagos, por lo que antes de capturarlos se requiere realizar una selección de los mismos.

Para ofrecer una alternativa que agilice el proceso descrito, se ha propuesto generar un archivo que concentre únicamente la información necesaria para registrar los pagos de las becas, cuyo formato sugerido es el de una hoja de cálculo que contenga los siguientes datos:

- Nombre completo del becario.
- Tutor asociado al becario.
- Monto total de la beca.
- Vigencia de la beca.

Para generar el archivo requerido, se puede extraer la información de la base de datos por medio de algunas funciones específicas de los JavaServer Pages (JSP).

Como consecuencia de la incompatibilidad que existe entre los sistemas mencionados se está definiendo, en conjunto con las Secretarías Académica y Administrativa, un mecanismo que permita generar los cheques de los becarios a partir de los resultados obtenidos durante el período de evaluación de las becas.

### 6.3 Resultados Generales.

Actualmente el sistema de control de becas se encuentra implementado sobre la arquitectura de desarrollo mostrada en el capítulo 4. Los requerimientos descritos en el capítulo 5 han sido completados en su totalidad. Sin embargo, durante el desarrollo del sistema surgieron nuevas necesidades algunas de las cuales ya han sido implementadas, mientras que las demás serán contempladas en la siguiente versión. Una descripción breve de los nuevos requerimientos se presenta en el apartado de "Propuestas para la siguiente versión" de este capítulo.

Teniendo como objetivo mantener un control en el uso del sistema, se ha escrito una propuesta del reglamento para el uso del mismo, la cual puede ser consultada en el Apéndice C de este trabajo. Dentro de los aspectos contemplados por el reglamento citado, se encuentra la labor de la Coordinación de Sistemas de Cómputo en lo que se refiere al mantenimiento del sistema. Las principales funciones de la Coordinación son asegurar el funcionamiento del sistema, generar los respaldos y las actualizaciones correspondientes a la base de datos y al código de programación.

En el siguiente y último capítulo, se exponen las conclusiones que obtuvimos al finalizar este trabajo.

### 6.4 Propuestas para la siguiente versión del sistema.

Aunque un sistema haya alcanzado sus objetivos de funcionamiento iniciales, la labor de desarrollo no puede considerarse concluida, debido a que siempre quedan requerimientos técnicos y funcionales por cubrir que hacen posible la continua evolución del sistema.

El proceso de desarrollo de software se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema. Cada ciclo concluye una versión interna<sup>1</sup> del producto (o en algunos casos una versión del producto lista para ser liberada), la cual cumple con los objetivos establecidos originalmente y expone los nuevos requerimientos que surgieron a lo largo del ciclo.

En nuestro caso el SCBII ha pasado por un par de versiones internas. Conforme cada versión ha sido analizada por los interesados, los requerimientos se han modificado. El simulacro interno que se llevo a cabo en la coordinación fue de gran ayuda para exponer nuevos requerimientos y problemas de funcionamiento del sistema, los cuales habrían sido difíciles de detectar sin un ejercicio de este tipo.

A continuación se mencionan los requerimientos obtenidos en la última fase de la versión actual del sistema y los cuales aún no han sido implementados.

---

<sup>1</sup> La versión interna del producto, es aquella que sólo sirve como iteración para los desarrolladores en su objetivo de ir perfeccionando el sistema.



- Es necesario implementar una serie de alertas para que a través del correo electrónico, se pueda dar aviso a las personas involucradas de la ocurrencia de un evento en particular. Un ejemplo de esto, es que el sistema sea capaz de avisar al tutor y al becario afectados de la proximidad de la fecha de fin de una beca.
- Toda la documentación necesaria para la aprobación de una beca, principalmente el plan de trabajo, el informe de actividades y el currículum de cada estudiante, debe de ser introducida al sistema como archivos adjuntos.
- En un futuro el sistema debe de permitir la consulta o la integración de la información del Sistema de Proyectos, lo cual ayudará a manipular el dinero destinado al pago de los cheques de los becarios. Actualmente, cada uno de los tutores puede consultar el estado de su fondo de becas y de esta manera, determinar si tiene recursos suficientes para patrocinar a cada una de las becas que promueva.
- Se requiere hacer un estudio de las páginas que requieren encriptar la información que envían o reciben, y posteriormente asociarlas a una conexión segura.

Además de los puntos antes mencionados, es importante considerar que antes de continuar con la evolución del sistema se debe de hacer una revisión exhaustiva y profunda de todos los elementos estructurales del mismo. Lo anterior se debe a que durante el desarrollo del sistema, éste estuvo expuesto a pequeñas modificaciones en los requerimientos originales, así como a la adición de nuevas características y comportamientos. Esta revisión ayudará a establecer bases aún más sólidas para continuar con las siguientes etapas de desarrollo del ciclo de vida del sistema.

# CAPÍTULO SIETE

## Conclusiones

## 7. Conclusiones.

El Instituto de Ingeniería, por ser uno de los centros de investigación más importantes de la UNAM, está consciente de la gran necesidad de mantenerse actualizado en toda la infraestructura tecnológica que lo apoya en la realización de sus funciones primordiales. Dentro de esas tareas principales se encuentra la labor desempeñada por el Programa de Becas del Instituto de Ingeniería (PBII), cuyo objetivo es la formación de recursos humanos principalmente de los estudiantes de licenciatura, de maestría y de doctorado quienes, a cambio del trabajo realizado dentro de la Institución, reciben un apoyo económico.

Dada la importancia que tiene el PBII y la obligación del Instituto por permanecer a la vanguardia en cuanto a recursos tecnológicos e informáticos, surgió la necesidad de crear un mecanismo automatizado para agilizar el proceso de evaluación de las becas otorgadas a los estudiantes, además de concentrar la información de su avance académico y de su trayectoria dentro del PBII.

Nuestra labor en la realización del proyecto descrito en esta tesis, consistió en el desarrollo y la implementación del mecanismo automatizado antes citado, el cual dio como resultado la primera versión del Sistema de Control de Becas. Para alcanzar nuestro objetivo fue necesario considerar tecnologías informáticas de punta; por ello realizamos una investigación que nos permitió conocer las herramientas de desarrollo y de programación de mayor actualidad en el mercado. Con los resultados de esa investigación, tomamos la decisión que a nuestro juicio era la más adecuada para la creación del sistema.

El sistema de control de becas que desarrollamos corresponde a una primera versión. Se trata de un sistema al que únicamente tienen acceso las computadoras que se encuentran dentro de la intranet del Instituto y que contempla distintos niveles de seguridad, dependiendo de la jerarquía del usuario. Asimismo, entre sus principales funciones, dicho sistema permite realizar el registro de nuevos estudiantes, tener acceso a la trayectoria de los becarios actuales del Instituto, registrar al personal académico, asignar y evaluar propuestas de becas, monitorear el estado del proceso de aprobación de una beca, registrar las bajas de los estudiantes y mantener actualizadas las tablas de catálogos.

Dentro de las principales ventajas que ofrece el Sistema de Control de Becas al Instituto de Ingeniería se pueden destacar las siguientes:

- Seguridad en el acceso, porque el uso del sistema está permitido únicamente a los funcionarios involucrados en el proceso. Además, el contenido de las páginas se genera en forma dinámica dependiendo de la jerarquía del usuario.
- Reducción de los tiempos de captura y de evaluación de las solicitudes, lo cual permite que el personal académico y los estudiantes beneficiados

puedan dedicarle más tiempo al desempeño de sus tareas prioritarias dentro del Instituto.

- Posibilidad de la creación de estadísticas, que permiten a los funcionarios del Instituto evaluar el desempeño académico de los estudiantes. El muestreo puede hacerse bajo diferentes criterios, conforme al promedio, al avance de créditos, al grado académico y a la unidad organizacional a la que pertenecen los becarios, entre otros.
- Monitoreo del proceso de evaluación de las becas asignadas, de manera que el tutor puede conocer en tiempo real, el estado actual de las becas asignadas y en caso de que alguna de ellas haya sido rechazada, puede saber cual fue el motivo de esa decisión.
- Además del tutor, los estudiantes también tienen la posibilidad de monitorear el estado de la evaluación de su solicitud de beca. Es importante mencionar que durante este proceso, los estudiantes conocen toda la información asociada a la beca con excepción del monto, el cual se revela hasta que la solicitud es aprobada.
- Control de las bajas de los estudiantes, lo que permite suspender la expedición de los pagos a personas que han terminado su labor como becarios del PBII. Además, por medio de este mecanismo se le da continuidad al contacto con los exbecarios del programa.
- Acceso inmediato a los registros históricos, que permite a los funcionarios consultar la trayectoria de los becarios y exbecarios del PBII.
- Ahorro de recursos materiales y humanos, así como la reducción del tiempo que dedica el personal administrativo del Instituto al apoyo del proceso de evaluación de las becas.

En la actualidad la red del Instituto cuenta con 700 nodos distribuidos en 10 edificios, los cuales se comunican con una velocidad máxima de 155 Mb/s. La velocidad hacia los servidores es de 100 Mb/s y hacia los clientes es de 10 Mb/s. Por lo anterior, consideramos que el Instituto cuenta con la infraestructura de red capaz de dar soporte a los más de 400 usuarios que van a utilizar el sistema.

Dentro de los planes a futuro del Instituto, se pretende adquirir la infraestructura necesaria para montar un centro de datos que reúna a las bases de datos de los sistemas de información del Instituto, incluyendo el sistema de control de becas.

En cuanto a los recursos humanos necesarios para mantener en operación el sistema, se puede mencionar que en la Coordinación de Sistemas de Cómputo se cuenta con diferentes grupos de trabajo, dentro de los cuales se

pueden mencionar los destinados al desarrollo y al mantenimiento de bases de datos, administración de servidores de web y soporte de la red del Instituto. Todas las áreas mencionadas pueden trabajar conjuntamente para asegurar el desempeño y el funcionamiento adecuados del sistema.

Aunque los resultados y las expectativas mencionadas pueden ser muy alentadores, probablemente el obstáculo más importante que enfrentará el uso del sistema, es la dificultad que tiene el personal académico para adecuarse a los cambios. Esto se debe a que desde que el PBII existe, el control del proceso de aprobación de las becas se ha hecho en forma manual, lo que ha dado lugar al arraigo de determinadas costumbres de parte del personal académico, ocasionadas principalmente por el exceso de trabajo en sus proyectos de investigación. Esta situación ha dado como consecuencia el retraso y la falta de eficiencia del proceso porque no se respetan las fechas límite para la entrega y la evaluación de las propuestas de becas.

Como consecuencia, es necesario solicitar el apoyo de la institución no sólo para promover y resaltar las ventajas que se obtendrán a partir del uso del sistema, sino también para apoyar su evolución y su perfeccionamiento.

En el ámbito personal este proyecto nos permitió percatarnos de que la etapa más importante durante el análisis de un sistema de información como el que hemos desarrollado, es la recopilación de los requerimientos. Esto se debe a que inicialmente las personas que van a utilizar el sistema sólo tienen una idea general y a veces imprecisa de los resultados que deben obtenerse.

Un aspecto muy importante para alcanzar los objetivos del sistema, consiste en realizar entrevistas exhaustivas con personas de todas las jerarquías involucradas, con el fin de que expongan sus ideas y necesidades, de manera que tanto los usuarios como los analistas logren tener una imagen lo más real posible de lo que será el producto final. Asimismo, es importante que las personas beneficiadas con el uso del sistema participen periódicamente en la revisión de los avances del mismo, con el fin de evitar que los desarrolladores tengan interpretaciones erróneas de los requerimientos establecidos originalmente.

En nuestro caso particular, tuvimos la oportunidad de mostrar el prototipo del sistema a los funcionarios interesados y se aclararon algunos requerimientos iniciales que habíamos malinterpretado, lo que dio lugar a la presentación de nuevas propuestas que modificaron en forma importante la estructura del sistema. De no haberse realizado esta revisión hubiera resultado más difícil efectuar los cambios en etapas posteriores.

Por lo anterior, aprendimos que un sistema necesita de múltiples versiones durante su ciclo de vida. A veces nos resultaba frustrante y difícil de comprender que el sistema no estaría completamente terminado desde su primera versión, debido a que las expectativas de los usuarios fueron aumentando conforme avanzaba el desarrollo del proyecto.

Para la creación del sistema de control de becas, fue necesario reafirmar y profundizar nuestros conocimientos de programación orientada a objetos, así como de diseño e implementación de bases de datos. Además, se requirió del aprendizaje de nuevos temas, tales como el análisis de sistemas, utilizando a UML como herramienta de visualización y la programación de Java Servlets y JavaServer Pages.

El éxito de la evolución del sistema dependerá, en gran parte, de que las personas que posteriormente se hagan cargo del proyecto tengan acceso a la información relacionada con las tecnologías mencionadas, a través de la documentación del sistema, cursos de capacitación, asesorías técnicas y libros especializados.

Finalmente, se puede concluir que los objetivos establecidos en el primer capítulo de esta tesis fueron cumplidos, incluyendo la elaboración de un reglamento para el uso del sistema. En lo que se refiere a las funcionalidades del sistema, es conveniente tomar en cuenta las propuestas para la siguiente versión descritas en el último capítulo de este trabajo.

# APÉNDICE A

## **A. Fragmento del Reglamento Interno del Instituto de Ingeniería.**

### **CAPITULO III. ORGANIZACIÓN**

#### **Artículo 4. Integración.**

El Instituto de Ingeniería está integrado por el Director, funcionarios, personal académico y personal administrativo. El Director es una autoridad universitaria. Son funcionarios académico-administrativos el Secretario Académico y los subdirectores. Son funcionarios administrativos el Secretario Administrativo y los Secretarios Técnicos. Además, en congruencia con sus funciones de formación, recibe estudiantes a través de su programa de becas o de otros programas institucionales, en los términos de los reglamentos respectivos.

El Instituto de Ingeniería tiene un Consejo Interno, una Comisión Dictaminadora, un Colegio Académico y participa en el Consejo Universitario, en el Consejo Académico de las Áreas de las Ciencias Físico-Matemáticas y de las Ingenierías y en el Consejo Técnico de la Investigación Científica, en los términos de la legislación correspondiente.

#### **Artículo 5. Organización.**

La unidad de organización del trabajo académico en el Instituto de Ingeniería es el **proyecto**, entendido como el conjunto de actividades que se realizan para la consecución de objetivos congruentes con las funciones del Instituto. Cada proyecto se desarrolla según una propuesta, previamente aprobada por el Director, que incluye objetivos, alcance, método de trabajo, programa de actividades, personal, recursos materiales y presupuesto. Está encabezado por un jefe de proyecto a quien corresponde elaborar la propuesta, dirigir y realizar el proyecto de acuerdo con ésta, informar al Coordinador y al Subdirector correspondientes, así como al Director y al Consejo Interno sobre su desarrollo y difundir oportunamente los resultados.

El personal que participa en proyectos con temas afines se agrupa en una Coordinación, y cierto número de coordinaciones, en una Subdirección.

El personal de apoyo a proyectos de diferentes coordinaciones se agrupa en unidades de servicio. Cada unidad de servicio tiene un jefe, quien depende del Secretario Académico, los Técnicos o el Administrativo, según el caso. Corresponde a los jefes de las unidades de servicio realizar y supervisar los trabajos que se les encomienden, mantener en buen estado las instalaciones a su cargo, así como dirigir y supervisar al personal asignado a su unidad.



El número de subdirecciones y coordinaciones, de secretarías técnicas y unidades de servicio es variable de acuerdo con los programas del Instituto, perspectivas de desarrollo y coherencia con los objetivos institucionales.

#### **CAPITULO IV. AUTORIDAD, FUNCIONARIOS Y COORDINADORES**

##### **Artículo 6. Del Director.**

El Director del Instituto es una autoridad universitaria, según lo señalan los artículos 3 de la Ley Orgánica y 12 del Estatuto General. Debe satisfacer los requisitos del artículo 52 del Estatuto General y ser designado por la Junta de Gobierno de una terna propuesta por el Rector, como lo establecen los artículos 6 y 11 de la Ley Orgánica y el artículo 34, fracción 7, del Estatuto General de la UNAM. Sus funciones son:

- a) Representar al Instituto.
- b) Dirigir y coordinar las actividades del Instituto, elaborar el proyecto de presupuesto y supervisar su ejercicio una vez autorizado.
- c) Participar, con voz y voto, en el Consejo Universitario y en el Consejo Técnico de la Investigación Científica.
- d) Proponer, ante el Consejo Técnico de la Investigación Científica, los asuntos del Instituto que sean de su competencia.
- e) Proponer al Rector la designación del Secretario Académico.
- f) Decidir, asesorado por el Consejo Interno, la formación de Subdirecciones, Coordinaciones, Secretarías Técnicas y Unidades de Servicio.
- g) Designar y remover a los Subdirectores y Coordinadores, auscultando la opinión del personal académico involucrado.
- h) Designar y remover a los Secretarios Técnicos y Administrativo y jefes de unidades de servicio.
- i) Convocar al Consejo Interno.
- j) Planear, asesorado por el Consejo Interno, las actividades del Instituto.
- k) Convocar a la elección de los representantes del personal académico ante los diversos cuerpos colegiados en los términos de lo señalado en la Legislación de la UNAM y en este Reglamento.

- l) Informar al Rector y al Consejo Técnico de la Investigación Científica, anualmente o cada vez que lo soliciten, sobre los planes de trabajo y el desarrollo de los mismos.
- m) Presentar públicamente el informe anual de actividades del Instituto e informar al Consejo Interno, cuando éste lo solicite, sobre el desarrollo de dichas actividades.
- n) Vigilar que la firma de convenios y contratos se ajusten a los lineamientos que fija el artículo 2 de este Reglamento y otros ordenamientos universitarios.
- o) Resolver, de acuerdo con las normas de la UNAM y asesorado por el Consejo Interno, los asuntos no previstos en este Reglamento, incluyendo los que miembros del Instituto sometan a su consideración.
- p) Desempeñar las demás funciones que le fije la Legislación Universitaria y los reglamentos respectivos.
- q) Realizar labores de investigación.

Quando deba designarse al Director del Instituto, el Consejo Interno, sin perjuicio de los diversos procedimientos de auscultación al personal académico, sugerirá al Rector una lista de candidatos para ocupar el cargo, con los nombres de las personas que hayan ocupado los primeros lugares en una votación directa y secreta de los investigadores titulares definitivos del Instituto que tengan más de un año de antigüedad en el mismo. El proceso de votación para integrar la lista será convocado y supervisado por el Consejo Interno.

#### **Artículo 7. Del Secretario Académico.**

El Secretario Académico es un funcionario académico-administrativo que debe cumplir con los mismos requisitos que el Director, exceptuando el de nacionalidad. Es designado por el Rector a propuesta del Director del Instituto, preferentemente de entre los investigadores titulares definitivos del Instituto. El Secretario Académico puede representar o sustituir al Director, según lo indican los artículos 51 y 54 del Estatuto General de la UNAM. Sus funciones son:

- a) Colaborar con el Director en la planeación, supervisión, coordinación y ejecución de las actividades de dirección del Instituto.
- b) Coordinar el uso eficiente de los recursos que se hayan asignado a la Secretaría Académica.
- c) Organizar y supervisar el programa de becas del Instituto.

- d) Coordinar las labores de formación y superación del personal académico del Instituto.
- e) Procurar la continua actualización y organización de las tareas académicas del Instituto.
- f) Opinar ante el Consejo Interno sobre los estímulos y remuneraciones adicionales de su personal.
- g) Realizar labores de investigación.

#### **Artículo 8. De los subdirectores.**

Los subdirectores son funcionarios académico-administrativos que deben cumplir con los mismos requisitos del Director, exceptuando el de nacionalidad, y son designados por éste en los términos del artículo 5 inciso g de este Reglamento. Sus funciones son:

- a) Colaborar con el Director en la planeación de las actividades del Instituto.
- b) Coordinar el uso eficiente de los recursos que se hayan asignado a su Subdirección.
- c) Proponer al Director la aprobación de proyectos de investigación y la firma de convenios de colaboración con diversas instituciones nacionales y del extranjero.
- d) Proponer al Director, después de consultar con el personal académico del área correspondiente, el nombramiento de coordinadores.
- e) Opinar ante el Consejo Interno sobre los estímulos y remuneraciones adicionales del personal académico de su subdirección.
- f) Realizar labores de investigación.

#### **Artículo 9. De los coordinadores.**

Los coordinadores son designados por el Director según el artículo 5 inciso g de este Reglamento. Sus funciones son:

- a) Orientar y coordinar las actividades académicas bajo su supervisión.
- b) Proponer al Subdirector la aprobación de proyectos de investigación y la firma de convenios de colaboración con diversas instituciones nacionales y del extranjero.

- c) Colaborar con el Subdirector en la organización y supervisión de las labores del Instituto correspondientes a su coordinación.
- d) Informar al Subdirector correspondiente sobre el desarrollo y los requerimientos de los proyectos bajo su supervisión.
- e) Fomentar la formación y superación del personal académico y estudiantes bajo su supervisión.
- f) Opinar ante el Subdirector sobre los estímulos y las remuneraciones adicionales del personal académico de su coordinación.
- g) Realizar labores de investigación.

#### **Artículo 10. Del Secretario Administrativo.**

El Secretario Administrativo es un funcionario administrativo designado por el Director en los términos del artículo 5, inciso h de este Reglamento. Las funciones del Secretario Administrativo son:

- a) Atender los servicios administrativos y coordinar el uso eficiente de los recursos que le hayan sido asignados.
- b) Facilitar a los funcionarios y al personal académico la información administrativa que requieran para el buen desempeño de sus funciones.
- c) Plantear al Director las necesidades de personal y recursos materiales para la realización de las labores que corresponden a la Secretaría Administrativa.
- d) Proponer al Director el nombramiento de los jefes de las unidades de servicio correspondientes.
- e) Colaborar con el Director en los asuntos que éste le encomiende.

#### **Artículo 11. De los Secretarios Técnicos.**

Los Secretarios Técnicos son funcionarios administrativos del Instituto. Su número dependerá del tipo y calidad de los servicios que requieran los programas del Instituto y su número es decidido por el Director, previa consulta al Consejo Interno. Las funciones específicas de cada Secretario Técnico son definidas por el Director; pero sus atribuciones genéricas son:

- a) Atender los servicios técnicos, coordinando el uso eficiente de los recursos que les hayan sido asignados.
- b) Proponer al Director el nombramiento de los jefes de unidades de servicio correspondientes.

c) Colaborar con el Director en los asuntos que éste les encomiende.

# APÉNDICE B

## **B. Reglamento del Programa de Becas.**

### **Capítulo 1. Objetivo**

El objetivo central del Programa de Becas del Instituto de Ingeniería (PBII) es coadyuvar a la formación de personal de alto nivel académico en las especialidades de la ingeniería que se cultivan en el Instituto para apoyar a la investigación, la docencia y la práctica profesional. Para ello, aquellos estudiantes, sobre todo de la Facultad de Ingeniería de la UNAM, que reúnan los requisitos señalados en este reglamento, podrán obtener una beca para realizar actividades escolares, participar activamente en proyectos de investigación del Instituto, contar con tutoría académica individual, recibir un estipendio y gozar de las demás prerrogativas indicadas en este reglamento.

### **Capítulo 2. Tipos**

Las becas pueden ser regulares y especiales. Las becas regulares son las que tienen como propósito obtener un grado académico; se otorgarán para los ciclos de licenciatura, maestría y doctorado en dos niveles:

- De estudio para cursar asignaturas y participar en proyectos de investigación.
- De tesis para realizar solamente actividades de investigación.

Las becas especiales se otorgarán, en casos excepcionales, para lograr propósitos complementarios a la obtención del grado, tales como: prestar servicio social, cumplir requisitos de admisión, participar en programas de capacitación e intercambio u otros aprobados por el Consejo Interno.

### **Capítulo 3. Estipendios**

Los estipendios de las becas regulares serán revisados y, en su caso, retabulados por el Consejo Interno al menos una vez por año, considerando los recursos presupuestales del Instituto y la congruencia con los indicadores pertinentes. Los de las becas especiales serán fijados casuísticamente de acuerdo al programa académico y los antecedentes del candidato.

Además, el Instituto realizará los trámites requeridos para que los estudiantes nacionales de la UNAM no desembolsen las cuotas de inscripción, colegiatura, revalidación de estudios, registro de tesis y otras de la misma naturaleza.

El Instituto acepta incorporar al PBII a estudiantes sin estipendio y a los becados o comisionados por otras instituciones. En este último caso, podrá asignar un estipendio complementario para alcanzar el que corresponda al del PBII.

## **Capítulo 4. Duración**

Las becas se otorgan por periodos máximos de un semestre y se renuevan semestralmente hasta cumplir con su propósito. Las becas regulares podrán renovarse hasta cuatro veces en los niveles de licenciatura o maestría y hasta nueve veces en el de doctorado, incluyendo la maestría. Las becas especiales solo podrán renovarse mientras se satisfagan los plazos mínimos requeridos para cumplir con los propósitos que les dieron origen.

## **Capítulo 5. Requisitos**

### *5.1 Primer ingreso*

a) Según el tipo de beca solicitada, satisfacer:

Para becas regulares de estudio, estar inscrito oficialmente en el nivel que corresponda y tener promedio general no inferior a 80/100.

Para becas regulares de tesis, haber cubierto la totalidad del programa en el nivel que corresponda con promedio general no inferior a 80/100, tener la aceptación oficial del tema de tesis y, en su caso, haber acreditado exámenes predoctorales.

Para becas especiales, tener el nivel académico requerido.

b) Elaborar conjuntamente con su Tutor un Programa General que conduzca a lograr el propósito de la beca en el tiempo reglamentado. Así mismo, elaborar el primer Programa Semestral de actividades.

### *5.2 Renovación*

a) Haber cumplido satisfactoriamente con las actividades del Programa Semestral, acreditar, en su caso, todas las asignaturas en que se haya inscrito, sostener un promedio general no menor a 80/100 y redactar el informe Semestral correspondiente.

b) Revisar, conjuntamente con su Tutor, el avance del Programa General de actividades y ajustarlo, en su caso, para lograr el propósito de la beca en el tiempo previsto. De acuerdo a ello, definir y redactar el siguiente Programa Semestral de actividades.

## **Capítulo 6. Derechos y obligaciones**

### *6.1 Derechos*

a) Recibir, en su caso, el estipendio aprobado; excepto por lo dispuesto en el capítulo 8.



- b) Tener tutoría académica adecuada para cumplir con su programa.
- c) Colaborar en proyectos de investigación en los que participe su Tutor, de acuerdo con los alcances de su programa individual.
- d) Dedicar parte del tiempo de investigación a preparar exámenes finales durante las fechas oficiales, previo acuerdo con su Tutor.
- e) Usar los resultados de su trabajo personal de investigación en su tesis y eventualmente publicarlos, previa aprobación de su Tutor, Coordinador y Subdirector.
- f) Solicitar a la Coordinación Editorial del Instituto la impresión de quince ejemplares de su trabajo de tesis.
- g) Participar en cursos extracurriculares para familiarizarse con técnicas específicas o en eventos congruentes con su formación académica. Cuando así lo considere su programa académico y existan los recursos podrán disfrutar de apoyo económico para inscripciones y viáticos.
- h) Solicitar fundadamente al Coordinador, Subdirector o Director, según corresponda, cambio de tutor o área de investigación después de un semestre y por una sola vez.
- i) Apelar ante la Dirección del Instituto la cancelación o no renovación de la beca, así como la suspensión temporal del estipendio.
- j) Tramitar una credencial que los identifique como miembros del PBII en el nivel correspondiente y, en su caso, una constancia de su estancia.
- k) Obtener apoyo institucional para disfrutar, eventualmente, de los instrumentos de formación del personal académico.

## 6.2 Obligaciones

- a) Cumplir los requisitos del capítulo 5.
- b) Dedicar tiempo completo a su programa académico.
- c) Comunicar al Instituto la existencia de otras becas o remuneraciones.
- d) Incluir un reconocimiento al Instituto en toda publicación o tesis producto de la beca y entregar una copia a la Secretaría Académica
- e) Solicitar semestralmente el formato de renovación.
- f) Acatar las disposiciones de este Reglamento y las de tipo administrativo del Instituto.

## **Capítulo 7. Tutores**

Fungirán como tutores los miembros del personal académico del Instituto que se comprometan a orientar y asesorar al becario en la definición, estructuración y conducción de su programa académico de estudios e investigación, así como a evaluar semestralmente su desempeño y formular las recomendaciones pertinentes.

## **Capítulo 8. Terminación**

### *8.1 Las becas se darán por terminadas cuando:*

- a) Se cumplan los propósitos de la beca o lo solicite el becario.
- b) Se tenga insuficiencia presupuestal o existan causas de fuerza mayor.

### *8.2 Las becas no serán renovadas cuando:*

- a) No se cumplan los requisitos del capítulo 5.
- b) Se agote el número de renovaciones reglamentado en el capítulo 4.

### *8.3 Las becas serán canceladas cuando:*

- a) Se violen las disposiciones de este Reglamento.
- b) Lo indique la legislación universitaria.

El pago del estipendio podrá suspenderse temporalmente cuando no se entreguen, sin causa justificada, los documentos probatorios del cumplimiento de los requisitos establecidos en el capítulo 5.

## **Capítulo 9. Procedimientos**

### *9.1 Primer ingreso*

a) El interesado obtendrá en la Secretaría Académica del Instituto una copia de este Reglamento y un formato de solicitud. Acudirá con el Coordinador del grupo de investigación en que desee participar, quien de acuerdo a las necesidades y recursos del grupo, le dará información complementaria y, según el caso, le aprobará o sugerirá un *Tutor*.

b) El estudiante leerá este reglamento y conjuntamente con su *Tutor*, elaborará su Programa General de actividades para no exceder las duraciones previstas, definirá su Programa Semestral, llenará su solicitud, reunirá los documentos probatorios de los requisitos establecidos en el inciso 5.1 y los entregará al Coordinador.

c) El Coordinador, atendiendo a los antecedentes académicos del solicitante, a su programa de actividades y a las observaciones del Tutor evaluará la solicitud conjuntamente con el Subdirector quien, en su caso, propondrá su ingreso al Director.

d) El Director autorizará el ingreso del solicitante y su estipendio de acuerdo con el resultado de la evaluación, recursos disponibles y programas de investigación.

e) El Subdirector comunicará el resultado de la gestión al Coordinador quien la comunicará al Tutor y éste al solicitante. Así mismo turnará el expediente a la Secretaría Académica para fines de seguimiento y gestiones administrativas subsecuentes.

### *9.2 Renovación*

a) El becario llenará y satisfará los requisitos de la solicitud que le hará llegar la Secretaría Académica y la entregará al Coordinador junto con el informe de sus actividades del último semestre, su programa de trabajo para el siguiente y, en su caso, las modificaciones al Programa General, todo avalado por su Tutor.

b) La evaluación y aprobación de la solicitud de renovación se harán de acuerdo con el procedimiento establecido en los puntos (c), (d) y (e) del inciso 9.1 y requisitos del inciso 5.2.

c) Los documentos probatorios sobre el cumplimiento de los requisitos deberán anexarse a la solicitud de renovación o, si por el momento no se dispone de ellos, mencionar su cumplimiento en el informe o programa semestral y entregarlos a la brevedad posible en la Secretaría Académica.

# APÉNDICE C

## **C. Propuesta del reglamento para el uso del SCBII.**

### **Generales.**

#### **1. Seguridad.**

- a) Si un usuario deja de enviar información al sistema, su sesión será cancelada 30 minutos después, contados a partir del momento en que dejó de utilizarla.
- b) Cada usuario debe de mantener la total confidencialidad de su clave y contraseña de acceso al sistema.
- c) Ningún usuario puede dejar abierta su sesión en el sistema.
- d) No se debe hacer uso indebido de los formularios del sistema.

#### **2. Sobre el PBII.**

- a) Se debe de respetar el reglamento del PBII, para cualquier transacción relacionada con la asignación, revisión y aprobación de una solicitud de beca.
- b) Para el registro y actualización de los datos y de las funciones de los académicos involucrados en el proceso de aprobación de becas, se debe de respetar el Reglamento Interno del Instituto de ingeniería.
- c) Dentro del SCBII, la Secretaría Académica tendrá la función de "subdirección", teniendo como "coordinaciones" adscritas a la Secretaría Técnica y a la Secretaría Administrativa. Lo anterior tiene como finalidad, respetar la cadena del proceso de aprobación de una beca.

### **Usuarios.**

#### **1. Estudiantes.**

Las siguientes son las responsabilidades generales de todos los estudiantes que utilicen el sistema.

##### *1.1 Registro de Datos*

- a) Proporcionar información estrictamente personal. Ningún estudiante está autorizado para introducir los datos de otro estudiante.
- b) Proporcionar toda la información que les sea requerida en forma actualizada, completa, verídica y comprobable.
- c) Notificar a su tutor si actualmente recibe otras becas adicionales a la del PBII.

- d) Redactar el documento correspondiente al plan de trabajo y/o al informe de actividades en formato de texto plano.
- e) En caso de que sus datos personales o académicos sean rechazados, el estudiante debe de hacer las correcciones que su tutor le indique.

*1.2 Seguridad.*

- a) Reportar al tutor cualquier anomalía en el uso de las claves de acceso.

*1.3 Consultas.*

- a) Monitorear el proceso de aprobación de su beca.

**2. Responsabilidades del estudiante de nuevo ingreso al PBII.**

*2.1 Seguridad y Uso del Sistema.*

- a) Avisar a su tutor si anteriormente había sido dado de alta en el SCBII.
- b) Pedir a su tutor que le proporcione la clave y la contraseña para su acceso al SCBII.

**3. Responsabilidades del estudiante inscrito y con derecho a renovación en el PBII.**

*3.1 Registro de Datos.*

- a) Recordarle con anticipación a su tutor de la fecha de terminación de su beca para que pueda entrar al sistema a llenar su informe de actividades.
- b) Actualizar sus datos personales y académicos cada vez que sea necesario.
- c) Actualizar su horario de asistencia al Instituto en cada cambio de semestre escolar.

*3.2 Seguridad.*

- a) En caso de que el estudiante no recuerde la clave y la contraseña de acceso al sistema, debe de solicitar a su tutor que le proporcione esta información.

#### 4. Tutores.

Las siguientes son las responsabilidades de cada uno de los tutores del PBII:

##### 4.1 Registro de datos.

- a) Registrar a los estudiantes de nuevo ingreso, introduciendo el nombre completo y la fecha de nacimiento. Es muy importante mencionar que estos datos, deben de ser cuidadosamente verificados antes de introducirlos al sistema.
- b) Verificar la validez de la información introducida por sus becarios o candidatos a becario.

##### 4.2 Control de las asignaciones de becas.

- a) Respetar los tiempos estipulados para dar de alta una solicitud de beca.
- b) No debe de asignar una beca a un estudiante que no haya llenado su plan de trabajo y en su caso el informe de actividades.
- c) Verificar y analizar cuidadosamente los datos relacionados a las becas que promueve.
- d) Verificar en su fondo de becas, si tiene el presupuesto suficiente para financiar la duración completa de cada una de las becas que promueva.
- e) Monitorear el proceso de las becas que ha asignado.
- f) Verificar que ninguna de sus solicitudes de beca quede sin revisión.
- g) En cuanto a los datos introducidos por un alumno, el tutor sólo puede hacer cambios al plan de trabajo y al informe de actividades.
- h) En caso de presentarse el rechazo de una solicitud beca, el tutor debe de analizar el motivo conjuntamente con el estudiante, y hacerle los cambios necesarios a la propuesta para poder continuar con el proceso de aprobación.

##### 4.3 Seguridad.

- a) Proporcionar a los estudiantes la clave y la contraseña que el sistema genera al introducir los datos de registro. En caso de ser necesario cambiar la contraseña de algún estudiante, el tutor es el único que tiene la autoridad para realizar este cambio.
- b) Permitir el acceso al sistema a los estudiantes con derecho a renovación, o que serán dados de baja.
- c) Cuidar la confidencialidad de las contraseñas de sus becarios.

#### *4.4 Bajas y Cancelaciones.*

- a) Dar de baja a un estudiante cuando éste termine su labor dentro del Instituto, introduciendo cada uno de los datos requeridos. El alumno debe de haber llenado su informe final de actividades para poder ser dado de baja del PBII.
- b) Cancelar una beca cuando se haya violado alguna de las normas establecidas por este reglamento o por el del PBII y con previa autorización de los funcionarios correspondientes.

### **5. Responsabilidades generales de los funcionarios (coordinadores, subdirectores, director).**

- a) Mantener actualizada su información personal.
- b) En el caso de rechazar una beca, deben de especificar la razón.
- c) Prestar mayor atención a la revisión de las becas especiales.

### **6. Responsabilidades del Secretario Académico.**

#### *6.1 Registro y Actualizaciones de datos.*

- a) Registrar a cada uno de los tutores del PBII.
- b) Registrar a cada uno de los coordinadores del Instituto.
- c) Registrar a cada uno de los subdirectores y al director del Instituto.
- d) Registrar a los secretarios del Instituto.
- e) Mantener actualizados los tabuladores de los montos de las becas regulares.
- f) Mantener actualizada la información de las unidades organizacionales del Instituto (coordinaciones, subdirecciones y secretarías).
- g) Mantener actualizada la información de las funciones que desempeñan cada uno de los académicos del Instituto.
- h) Mantener actualizada la información de los catálogos (universidades, períodos de becas, instituciones externas de financiamiento, justificaciones para la autorización de becas especiales, motivos de rechazo de becas, etc.).
- i) Generar la información de los reportes estadísticos.

#### *6.2 Seguridad.*

- a) Autorizar las modificaciones al sistema.
- b) Autorizar el archivo generado para uso de la Secretaría Administrativa.



## **7. Responsabilidades de la Coordinación de Sistemas de Cómputo.**

### *7.1 Registro de datos.*

- a) Registrar al Secretario Académico.

### *7.2 Mantenimiento del Sistema.*

- a) Asegurar el funcionamiento del sistema, lo cual incluye el mantenimiento del hardware y el de la infraestructura requerida.
- b) Realizar los respaldos de la base de datos y del código fuente del sistema.
- c) Hacer las modificaciones que se requieran en el código del sistema, con previa autorización del Secretario Académico.

# BIBLIOGRAFÍA

**Capítulo 1.***Páginas de Internet.*

- Instituto de Ingeniería UNAM, [www.iingen.unam.mx/homeesp.html](http://www.iingen.unam.mx/homeesp.html)
- Instituto de Ingeniería UNAM, "Reglamento del Programa de Becas", [www.iingen.unam.mx/infins/becas/index.html](http://www.iingen.unam.mx/infins/becas/index.html)

**Capítulo 2.***Páginas de Internet.*

- CISCO Systems, Solutions Guide, "Building an ERP-Ready Network", [www.cisco.com/warp/public/cc/so/cuso/epso/entdes/ernp\\_wp.pdf](http://www.cisco.com/warp/public/cc/so/cuso/epso/entdes/ernp_wp.pdf).

**Capítulo 3.***Libros.*

- Marty Hall, "Servlets y JavaServer Pages, Guía Práctica", PEARSON EDUCACIÓN, México, 2001.

*Páginas de Internet.*

- Bruce Momjian, "Frequently Asked Questions (FAQ) for PostgreSQL", [www.math.eku.edu/tools/FAQ-Postgres.html](http://www.math.eku.edu/tools/FAQ-Postgres.html)
- Diego Pino García, "Cliente de Correo Web, Apéndice C: PHP Hypertext Preprocessor ", Dic 2000 , [couso.ei.uvigo.es/manualestecnicos/documentos/php.pdf](http://couso.ei.uvigo.es/manualestecnicos/documentos/php.pdf)
- eWeek, Poll Archive, [www.eweek.com/poll\\_archive/0,3666,p=1144&bn=1,00.asp](http://www.eweek.com/poll_archive/0,3666,p=1144&bn=1,00.asp)
- Java Server Pages, "Dinamically Generated Web Content", [www.javasoft.com/products/jsp/index.html](http://www.javasoft.com/products/jsp/index.html)
- Jason Hunter, "The Problems with JSP", [www.servlets.com/soapbox/problems-jsp.html](http://www.servlets.com/soapbox/problems-jsp.html)
- Keerthi Thomas, "A Web Developer's Perspective: from ASP to JSP", [www.indiawebdevelopers.com/technology/java/jsp.asp](http://www.indiawebdevelopers.com/technology/java/jsp.asp)
- Mark Jorgenson, "Web Development", [komar.cs.stthomas.edu/qm425/00f\\_01/Jorgenson1.htm](http://komar.cs.stthomas.edu/qm425/00f_01/Jorgenson1.htm)
- Netcraft: [www.netcraft.com](http://www.netcraft.com)

- Peter Sayer, "Oracle extends lead in database sales", [www.itworld.com/AppDev/119/DataquestOracleexte413](http://www.itworld.com/AppDev/119/DataquestOracleexte413)
- Sun Chill!Soft "Why Active Servers?" [www.chillsoft.net/whitepeppers/Why%20Active%20Server%20Pages.pdf](http://www.chillsoft.net/whitepeppers/Why%20Active%20Server%20Pages.pdf)
- Sudhanshu Arora, "Dynamic Scripting Language PHP", Oct. 2000, [www.eecs.umich.edu/~aprahash/585/html/php.pdf](http://www.eecs.umich.edu/~aprahash/585/html/php.pdf)
- Vince Logozzo, "User-Friendly Online Database" [www.muc.edu/~logozzva/sce/revision1.html](http://www.muc.edu/~logozzva/sce/revision1.html)

#### Capítulo 4.

##### *Páginas de Internet.*

- CISCO Systems, Solutions Guide, "Building an ERP-Ready Network", [www.cisco.com/warp/public/cc/so/cuso/epso/entdes/erpn\\_wp.pdf](http://www.cisco.com/warp/public/cc/so/cuso/epso/entdes/erpn_wp.pdf).
- Java Developer Connection, "Fundamentals of Java Servlets: The Java Servlet API", [developer.java.sun.com/developer/onlineTraining/Servlets/Fundamentals/Servlets.html](http://developer.java.sun.com/developer/onlineTraining/Servlets/Fundamentals/Servlets.html).
- John M. Thompson, "Java Servlets – Focus on Encapsulation and Extension of CGI", [www.iwaypublishing.com/jt/javaservlets0401.html](http://www.iwaypublishing.com/jt/javaservlets0401.html)
- Joshua Chamas, Apache :: ASP , [freshmeat.net/projects/apacheasp/](http://freshmeat.net/projects/apacheasp/)
- KUSANAGINET.COM , "Apache 1.3 Vs Roxen 2.2", [www.kusanaginet.com/article.php3?id\\_article=7](http://www.kusanaginet.com/article.php3?id_article=7)

#### Capítulo 5.

##### *Libros.*

- Craig Larman "UML y patrones, Introducción al análisis y diseño orientado a objetos", Prentice- Hall, México 1999
- G. Booch, J. Rumbaugh, I. Jacobson, "The Unified Modeling Language User Guide", Addison- Wesley object technology series, USA 1999.
- G. Booch, J. Rumbaugh, I. Jacobson, "El proceso unificado de desarrollo de software", Pearson Educación, S.A., Madrid, 2000.
- Laura Lemay, Charles L. Perkins, "Aprendiendo Java 1.1 en 21 días", Prentice-Hall Hispanoamericana, S.A., México 1998.

- Microsoft Training and Certification, "Programming a Microsoft SQL Server 2000 Database", Classroom Setup Guide, Microsoft Corporation 2000.
- Mike Morgan, "Descubre Java 1.2", Prentice-Hall Iberia, Madrid 1999.
- Richard Barker, "Case Method, Entity Relationship Modelling", Oracle Manual, Addison-Wesley Publishing Company, EUA 1990.

*Páginas de Internet.*

- Instituto de Ingeniería, UNAM, "Reglamentos",  
[www.iingen.unam.mx/infins/reglas.html](http://www.iingen.unam.mx/infins/reglas.html)

**Capítulo 6.**

*Páginas de Internet.*

- Dotcomfidence.com, "Knowledge Base",  
[www.dotcomfidence.com/htmls/Knowledge\\_base/KnowledgeBase](http://www.dotcomfidence.com/htmls/Knowledge_base/KnowledgeBase)
- Philosophie.com, "Types of Tests", USA 1999,  
[www.philosophie.com/testing/tests.html](http://www.philosophie.com/testing/tests.html)
- The Test Company, "Load and Stress Testing FAQ",  
[www.ttcx.com/sl\\_faq.shtml#whatis](http://www.ttcx.com/sl_faq.shtml#whatis)