

Universidad Autónoma Metropolitana Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería

Licenciatura en Ingeniería en Computación

Reporte de proyecto terminal

Gestión del archivo digital de la Coordinación de Posgrados de CBI mediante un sistema de información web

Hernández Báez Antonio 202314716

Neria Martínez Francisco Javier 206304789

Trimestre 13 - I

Abril de 2013

Asesora: Dra. María Lizbeth Gallardo López  
Profesor-investigador  
Departamento de Sistemas

---

## Agradecimientos

---

Agradecemos de manera muy especial a nuestra asesora de proyecto terminal:  
Dra. María Lizbeth Gallardo López  
por su paciencia y apoyo.

A los profesores y personal del Posgrado de CBI que nos brindaron su amistad y ayuda:  
Dra. Margarita Juárez Nájera  
D.G. Perla Adriana Velasco Santos  
Dr. José Luis Cardoso Cortés  
Dr. José Luis Hernández Ávila  
Dr. Francisco Javier Zaragoza Martínez

A los profesores:  
M. en C. María Antonieta García Galván  
Dra. Beatriz Adriana González Beltrán  
Ing. Jorge Damaso Hanel del Valle  
Dr. Alejandro Kunold Bello  
Dr. Juan Villegas Cortez  
Por sus valiosos consejos durante nuestra formación profesional.

Azcapotzalco, Distrito Federal, México.

Hernández Báez Antonio  
Neria Martínez Francisco Javier

Abril de 2013

---

# Índice de contenido

---

Agradecimientos.....	II
1 Introducción.....	1
2 Objetivos.....	2
2.1 Objetivo general.....	2
2.2 Objetivos específicos.....	2
3 Antecedentes.....	3
3.1 Referencias internas.....	3
3.2 Referencias externas.....	4
4 Justificación.....	6
5 Metodología.....	7
6 Desarrollo.....	9
6.1 Diseño.....	9
6.1.1 Casos de uso.....	9
6.1.2 Escenarios.....	10
6.1.3 Tarjetas CRC.....	12
6.1.4 Modelo de dominio.....	14
6.1.5 Diagrama de secuencias.....	14
6.1.6 Modelo de clases.....	16
6.1.7 Modelo de la base de datos.....	20
6.1.8 Arquitectura.....	25
6.1.9 Interfaz gráfica de usuario.....	26
6.2 Implementación.....	29
6.2.1 Tecnología empleada.....	29
6.2.1.1 Hardware.....	29
6.2.1.2 Software.....	30
6.2.2 Funcionalidad.....	31
6.2.2.1 Capa SSL.....	31
6.2.2.2 Implementación del modelo.....	32
6.2.2.3 Implementación de la vista.....	35
6.2.2.4 Implementación del controlador.....	37
6.2.3 Pruebas.....	38
7 Conclusiones y trabajo futuro.....	41
8 Anexo.....	43
8.1 Claves y nomenclatura de archivos.....	44
8.2 Nomenclatura de archivos.....	47
Glosario.....	49
Bibliografía.....	50

---

## Índice de figuras

Figura 5.1: Proceso unificado visto en dos dimensiones [8].....	7
Figura 6.1.1.1: Diagrama de casos de uso general del SIP-CBI.....	9
Figura 6.1.2.1: Subcasos de uso del caso de uso "Gestionar alumnos".....	10
Figura 6.1.2.2: Escenario "GA1: Realizar alta de alumno".....	12
Figura 6.1.4.1: Tarjetas CRC para la entidad alumnos.....	14
Figura 6.1.5.1: Modelo de dominio de las entidades asociadas a la entidad alumno.....	15
Figura 6.1.6.1: Diagrama de secuencias para la operación "alta".....	16
Figura 6.1.6.2: Diagrama de clases para alumnos.....	18
Figura 6.1.6.3: Diagrama de clases general del SIP-CBI.....	19
Figura 6.1.7.1: Base de datos relacional diseñada a partir del diagrama de clases de la Figura 6.1.6.2.....	23
Figura 6.1.7.2: Base de datos relacional del SIP-CBI.....	24
Figura 6.1.8.1: Funcionamiento del patrón MVC.....	25
Figura 6.1.9.1: Prototipo de interfaz gráfica.....	26
Figura 6.1.9.2: Interfaz gráfica final del SIP-CBI.....	26
Figura 6.1.9.3: Propuesta de navegabilidad para el proceso de alta.....	27
Figura 6.1.9.4: Proceso de alta visto desde la interfaz gráfica.....	28
Figura 6.1.9.5: Propuesta de navegabilidad para el proceso de búsqueda y cambio de información.....	28
Figura 6.1.9.6: Proceso de búsqueda y cambio de información visto desde la interfaz gráfica.....	29
Figura 6.2.2.1.1: Verificación del funcionamiento de la capa SSL por medio del navegador Chromium.....	32
Figura 6.2.2.2.1: Interfaz gráfica para el proceso de alta de un alumno.....	33
Figura 6.2.2.2.2: La interfaz muestra un mensaje que indica el procesamiento del formulario de manera exitosa.....	33
Figura 6.2.2.3.1: Interfaz gráfica del proceso de alta de un alumno.....	35
Figura 6.2.3.1: Prueba de integridad referencial a la tabla alumnos.....	38
Figura 6.2.3.2: Prueba de integridad referencial a la tabla posgrados.....	39
Figura 6.2.3.3: Pruebas realizadas al proceso de autenticación de usuarios.....	39
Figura 6.2.3.4: Validación de los campos matrícula, nombre y apellido paterno.....	40
Figura 6.2.3.5: Validación de datos en el formulario para cambiar datos de un alumno.....	40
Figura 8.1: Formato de control de medios de verificación del PNPC.....	43
Figura 8.2.1: Propuesta de nomenclatura para el archivo de fotografía de usuario.....	47
Figura 8.2.2: Propuesta de nomenclatura de archivos asociados a la entidad alumnos.....	48

---

## Índice de tablas

---

Tabla 7.1: Logros alcanzados durante el proyecto.....	42
Tabla 8.1.1 Claves para posgrados.....	44
Tabla 8.1.2 Tipos de estatus de las entidades trimestresalumnos, tesis, docentes, posgrados, planestudios y usuarios.....	44
Tabla 8.1.3 Claves para estatus de alumnos por trimestre.....	46
Tabla 8.1.4 Claves para estatus de docentes.....	46
Tabla 8.1.5 Claves para estatus de tesis.....	46
Tabla 8.1.6 Claves para estatus de posgrados.....	46
Tabla 8.1.7 Claves para estatus de planes de estudios.....	46
Tabla 8.1.8 Claves para estatus de usuarios.....	46
Tabla 8.1.9 Claves para documentos personales.....	46
Tabla 8.1.10 Claves para documentos asociados al alumno.....	47
Tabla 8.1.11 Clave para proyectos de investigación dirigidos por docentes.....	47
Tabla 8.1.12 Clave para planes de estudios.....	47
Tabla 8.2.1: Tabla de definiciones del glosario.....	49

---

# 1 Introducción

---

El proyecto tiene por objetivo crear un sistema de información con el cual se facilitará la gestión de la información de los docentes, alumnos y planes de estudios pertenecientes a la Coordinación de Estudios de Posgrados de Ciencias Básicas e Ingeniería. El sistema también deberá ser capaz de generar reportes y almacenarlos en una imagen *ISO* de acuerdo a los parámetros establecidos por el PNPC<sup>1</sup> del Consejo Nacional de Ciencia y Tecnología.

---

1 PNPC: El Programa Nacional de Posgrados de Calidad de CONACyT solicita a las instituciones registradas en él un *CD* con un reporte que contenga datos académicos de alumnos, docentes y planes de posgrados, entre otros.

---

## 2 Objetivos

---

### 2.1 Objetivo general

- Diseñar e implementar un sistema de información web para la gestión del archivo digital de la Coordinación de Posgrados de la División de Ciencias Básicas e Ingeniería.

### 2.2 Objetivos específicos

- Diseñar e implementar un módulo que permita la gestión de administradores.
- Diseñar e implementar un módulo que permita la gestión de alumnos.
- Diseñar e implementar un módulo que permita la gestión de docentes.
- Diseñar e implementar un módulo que permita la gestión de planes de estudio de posgrado.
- Diseñar e implementar un módulo que permita obtener informes de acuerdo a los requerimientos establecidos por el PNPC del Consejo Nacional de Ciencia y Tecnología.
- Diseñar e implementar una base de datos que soporte la información correspondiente de los módulos a desarrollar.
- Diseñar una interfaz gráfica de usuario que permita utilizar el sistema web.
- Integrar los módulos y la interfaz gráfica de usuario.

---

## 3 Antecedentes

---

Se han realizado esfuerzos por crear sistemas de información que ayuden a gestionar el archivo del posgrado y el proceso de preregistro de los aspirantes. En el periodo de 2008 a 2010 se implantó un sistema, el cual fue desarrollado por personal de la División de Ciencias Sociales y Humanidades de la UAM Azcapotzalco. Este sistema permite actualizar los datos personales de docentes, el contenido del sitio web y el preregistro de aspirantes a los posgrados. De acuerdo a los comentarios expresados por la Coordinación de Posgrados, este sistema carece de manuales de usuario, manuales de instalación y documentación sobre su diseño. El mismo sistema fue desarrollado en el lenguaje *ASP*<sup>2</sup> e implantado en un equipo con sistema operativo *Windows 2003 Server* y servidor web *IIS*<sup>3</sup>, y no se especificó si se contaban con las licencias originales. El sistema tampoco cuenta con cifrado de contraseñas, ni está soportado sobre una capa segura que proteja la integridad de las transacciones entre el cliente y el servidor. Por lo anterior, este sistema fue retirado en el trimestre 10-P.

Existen proyectos dentro y fuera de la Universidad que tienen como propósito la gestión de información académica de alumnos y docentes, y la gestión de archivos.

### 3.1 Referencias internas

- a) **Sistema automatizado de información para las oficinas de la Coordinación Divisional de Docencia y Atención a Alumnos de CBI para el Área de Posgrado.** En este proyecto se diseñó un sistema para gestionar la información académica de los alumnos de posgrado de la División de Ciencias Básicas e Ingeniería de la UAM Iztapalapa. Se diseñó el sistema siguiendo el paradigma de programación orientada a objetos, auxiliándose con el lenguaje unificado de modelado<sup>4</sup>. Este proyecto sugiere las tecnologías: *Java Applet*<sup>5</sup>, lenguaje de marcado de hipertexto con *CGI*<sup>6</sup> y conexión a una base de datos mediante *DBSP*<sup>7</sup>, para implementar el sistema en web [1].

En nuestro proyecto se diseñó e implementaron los módulos descritos en los objetivos particulares, mediante el uso del proceso unificado<sup>8</sup> y el lenguaje unificado de modelado.

---

2 *ASP (Active Server Pages)*: Tecnología de *Microsoft* para desarrollar páginas web dinámicas.

3 *IIS (Internet Information Services)*: Servidor web desarrollado por *Microsoft* para el sistema operativo *Windows*.

4 Lenguaje unificado de modelado: Lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

5 *Java Applet*: Componente de una aplicación que se ejecuta en el contexto de otro programa escrito en lenguaje *Java*.

6 *CGI (Common Gateway Interface)*: Tecnología que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web.

7 *DBSP (Data Base Scripting Pages)*: Tecnología que se ejecuta en el servidor diseñada para acelerar el proceso de construcción de sitios *B2B (Business to Business)*, *B2C (Business to Client)*, cualquier sistema *e-business* o de información que requiera la explotación de datos a través de Internet.

8 Proceso unificado: Marco de desarrollo de software que se caracteriza por estar dirigido por casos de uso, centrado en la arquitectura y por ser iterativo e incremental.

Su implementación se realizó en el lenguaje *PHP*<sup>9</sup> orientado a objetos. La base de datos se desarrolló en *MySQL*<sup>10</sup>.

- b) **Sistema de información de la producción académica de los profesores de la División de Ciencias Básicas e Ingeniería.** Este proyecto terminal gestiona la información académica, estudios realizados, especialidades, artículos, libros y congresos de los profesores adscritos a la División de CBI de la UAM Iztapalapa. Este proyecto fue implementado en *Clipper*<sup>11</sup> con una base de datos elaborada en *DbaseIII+*<sup>12</sup>. El proyecto fue diseñado para instalarse en una o varias computadoras pero sin la posibilidad de sincronizar la información en una base de datos única [2].
- c) **Análisis y diseño de un sistema de información para el Departamento de Economía. Este proyecto terminal realiza la gestión de información académica de profesores adscritos al Departamento de Economía de la UAM Iztapalapa.** La gestión incluye publicaciones, proyectos de investigación, cursos impartidos, inventario de equipo de cómputo y convenios con otras instituciones. El sistema fue diseñado siguiendo el paradigma orientado a objetos, auxiliándose del lenguaje unificado de modelado. Las interfaces gráficas de la aplicación fueron desarrolladas en *Visual Basic*<sup>13</sup> y únicamente propone el modelo relacional de la base de datos (no realizó su implementación) [3].

En los dos proyectos anteriores se considera únicamente la gestión de información de profesores. Nuestro proyecto considera la gestión de información de profesores, alumnos, planes de estudio de posgrado y reportes. Además, se usó una base de datos relacional y centralizada en la cual se almacena la información.

## 3.2 Referencias externas

- a) **Dropbox.** Es un servicio de alojamiento de archivos basado en web. Utiliza el modelo de red *cloud storage*<sup>14</sup> para que sus usuarios almacenen y compartan archivos, y/o directorios, a través de Internet. El servicio es gratuito hasta los 2GB de almacenamiento; si se requiere de mayor capacidad debe pagarse una cuota mensual. Los usuarios registrados en el servicio pueden gestionar sus archivos desde el sitio web de la compañía; o bien, descargar una aplicación cliente en su computadora local. Está disponible para distintos sistemas operativos (*Windows, MacOS, Linux* y teléfonos móviles) [4].

---

9 *PHP (PHP Hypertext Pre-processor):* Es un lenguaje de programación interpretado diseñado para la creación de páginas web dinámicas.

10 *MySQL:* Sistema de gestión de bases de datos relacional, multihilo y multiusuario.

11 *Clipper:* Lenguaje de programación procedural e imperativo creado en 1985 por *Nantucket Corporation*.

12 *DbaseIII+:* Sistema de gestión de bases de datos desarrollado en 1960.

13 *Visual Basic:* Lenguaje de programación dirigido por eventos, desarrollado por el alemán Alan Cooper para *Microsoft*.

14 *Cloud storage:* Modelo de red de almacenamiento en línea donde los datos se almacenan en repositorios virtualizados que son generalmente organizados por terceros.

### 3 Antecedentes

- b) **Owl**. Software para crear un repositorio de documentos basado en web que permite a múltiples usuarios guardar documentos y compartirlos con otros usuarios inscritos a *Owl*. Su licencia es de tipo *GNU/GPL*<sup>15</sup> [5].
- c) **OpenKM**. Software para gestión documental orientado a la pequeña y mediana empresa. Posee una interfaz de usuario web para realizar operaciones como: compartir, establecer la seguridad, auditar y localizar documentos. Este software se encuentra disponible para los sistemas operativos *Windows*, *Linux*, *MacOS* y *Unix*. El software es de código abierto<sup>16</sup> y se distribuye mediante dos licencias: de paga y licencia *GNU/GPL* [6].

Nuestro proyecto no es un servicio para alojar y compartir archivos en red, como los tres mencionados anteriormente. Nuestro sistema se centra en la gestión de un archivo digital, por medio de un sistema web. La información de los estudiantes, profesores y personal administrativo son alojados en una base de datos y los documentos como: tesis, actas, solicitudes de becas, artículos de investigación, entre otros, son almacenados en forma de archivos. Esta información podrá ser accedida únicamente por los roles que tengan derecho sobre ella. Por lo tanto, no se considera la posibilidad de compartir la información, tal y como lo hacen los sistemas tipo *Dropbox*, *Owl* y *OpenKM*.

---

15 *GNU/GPL (GNU General Public License)*: Licencia creada por la Fundación de Software Libre orientada a proteger la libre distribución, modificación y uso del software.

16 Código abierto: Término con el que se conoce al software distribuido y desarrollado libremente.

---

## 4 Justificación

---

Los Coordinadores de los posgrados de la División de Ciencias Básicas e Ingeniería tienen la necesidad de llevar el control de la información de los planes de estudio, alumnos y docentes. Esta información se encuentra distribuida en documentación impresa y en archivos en formato electrónico; a los cuales nos referiremos como archivo del posgrado. Actualmente estos archivos se encuentran almacenados en diferentes computadoras lo que dificulta su rápida localización y administración, además, no se lleva un control de las personas que acceden a esta información.

En este proyecto terminal se propuso crear un nuevo sistema de gestión del archivo del posgrado que permita incluir una base de datos para almacenar los datos de los usuarios, y una base de documentos para toda la información: tesis, actas, solicitudes de becas, *kárdex*, y artículos de investigación, entre otros. Para garantizar la seguridad de las transacciones entre el cliente y el servidor, el sistema se implementó sobre una capa de conexión segura, además se usó contraseñas con cifrado. Los Coordinadores de los Posgrados y asistentes administrativos utilizan computadoras con distintos sistemas operativos (*Linux*, *Windows* o *MacOS*). Ante esta situación y con el fin de que puedan gestionar el archivo del posgrado desde cualquier computadora, sin importar el tipo de sistema operativo o instalar software de terceros, fue implementado como un sistema de información web.

El sistema, el cual denominaremos Sistema de Información de los Posgrados de CBI o SIP-CBI, ofrece los siguientes beneficios: se automatizan los procesos de búsqueda de información, generación de reportes, se tiene la información de los alumnos y docentes centralizada, el acceso y transferencias de información se efectúan de forma segura, cuenta con una interfaz intuitiva y fácil de utilizar desde una aplicación cliente (navegador). Además, permite una administración adecuada de la información de los alumnos y docentes del posgrado. Este sistema cuenta con la documentación del análisis, diseño e implementación, lo que permite que el sistema sea escalable y adaptable a nuevos requerimientos. También, se cuenta con un manual de usuario e instalación.

## 5 Metodología

Para diseñar el SIP-CBI se utilizó la metodología orientada a objetos llamada proceso unificado y el lenguaje unificado de modelado para representar los sistemas de información [7]. Una de nuestras inquietudes fue reforzar nuestros conocimientos en el paradigma orientado a objetos, ya que en el transcurso de nuestra formación profesional el tipo de paradigma que más desarrollamos y trabajamos fue el paradigma estructurado.

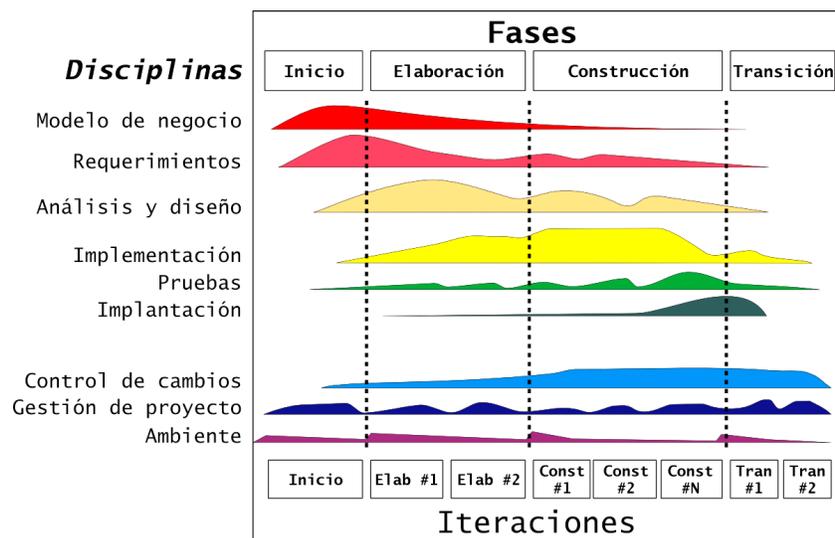


Figura 5.1: Proceso unificado visto en dos dimensiones [8].

El proceso unificado se compone de artefactos a su vez formados por una serie de disciplinas que cambian a lo largo del tiempo (ver Figura 5.1). El tiempo se encuentra clasificado en cuatro fases: inicio, elaboración, construcción y transición. A continuación se describe las actividades realizadas durante cada fase, así como los artefactos desarrollados:

- **Inicio:** Se inició la elaboración de los documentos de visión, casos de uso, especificación complementaria, glosario y lógica de negocio (actual y propuesto). Se instaló el sistema operativo *Debian* en una máquina virtual en el software *VirtualBox*. En esta máquina virtual se instaló *Apache* (servidor web), *OpenSSL* (capa segura), *MySQL* (base de datos), *PHP* (intérprete del lenguaje de programación), *CakePHP* (*framework*) y el software para la elaboración de diagramas UML (*Dia*, *StarUML* y *MySQL Workbench*). Para determinar los requerimientos del cliente se realizaron una serie de entrevistas con algunos Coordinadores del Posgrado de CBI y personal administrativo. Además, se consultaron documentos como el denominado “Formato de control de medios de verificación del PNPC” que puede verse en el Anexo.
- **Elaboración:** Durante esta fase se refinaron los documentos de visión, casos de uso y especificación complementaria. Este refinamiento consistió en revisar y realizar cambios con el cliente de los requerimientos obtenidos durante la fase de inicio. Se crearon los

documentos de escenarios de uso y tarjetas *CRC*<sup>17</sup> para identificar las entidades a implementar. Una vez identificadas las entidades se crearon los documentos de modelo de dominio, diagramas de secuencias y diagramas de clases. Los documentos de escenarios, tarjetas *CRC* y modelo de dominio fueron revisados continuamente para definir las entidades a convertir en clases de software. Se definió la arquitectura a implementar (MVC). Con ayuda del modelo de clases se obtuvo el modelo de base de datos (esquema lógico). Se inició la implementación parcial del módulo de usuarios. Esta implementación consistió en escribir el código fuente de las clases del módulo de usuarios y realizar pruebas en el servidor web en una capa *SSL*<sup>18</sup>.

- **Construcción:** En esta fase se refinaron los documentos modelo de dominio, diagramas de secuencias, diagramas de clases y modelo de base de datos. Se inició la implementación de los módulos en el *framework CakePHP* pero su uso fue descartado posteriormente debido a su poca flexibilidad en el manejo de los datos, lo cual corrompía el diseño original; por lo tanto, la implementación de las clases del sistema no extienden ni heredan de las clases de alguna *framework*. Se implementaron los módulos de usuarios, alumnos, planes de estudios, docentes y adicionalmente un módulo de sistema.
- **Transición:** Se continuó la implementación del SIP-CBI corrigiendo errores obtenidos en las pruebas. Una vez concluida la implementación de los módulos se instaló el SIP-CBI en el servidor de las instalaciones del Posgrado de CBI. Se realizaron pruebas y ajustes en el código fuente y en los documentos de diagramas de clases y modelo de la base de datos.

El ciclo de vida del modelo empleado es iterativo e incremental, por lo tanto, se realizó una gestión del proyecto durante todas las fases.

---

17 Formalmente las tarjetas *CRC* (*Class-Responsibility-Collaboration*) no forman parte del lenguaje unificado de modelado, pero son un mecanismo útil para distinguir responsabilidades, atributos y colaboraciones [9].

18 *SSL* (*Secure Socket Layer*): Protocolo criptográfico que proporciona comunicaciones seguras en una red.

### 6.1 Diseño

Esta sección tiene por objetivo mostrar las técnicas empleadas para el diseño del sistema y la selección de la arquitectura, para lo cual se desarrollaron las siguientes secciones: 6.1.1 Casos de uso, 6.1.2 Escenarios, 6.1.3 Tarjetas CRC, 6.1.4 Modelo de dominio, 6.1.5 Diagrama de secuencias, 6.1.6 Modelo de clases, 6.1.7 Modelo de la base de datos, 6.1.8 Arquitectura y 6.1.9 Interfaz gráfica de usuario. Se ejemplificará cada sección con el módulo de alumnos, en específico, con la operación *alta de un alumno*. En el documento de diseño del sistema se describen los módulos restantes.

#### 6.1.1 Casos de uso

Como resultado del análisis de requerimientos se detectaron cuatro grandes casos de uso: gestionar usuarios, gestionar alumnos, gestionar planes de estudio y gestionar docentes. Los casos de uso incluyen al “gestionar autenticación de usuarios” ya que es necesario estar autenticado dentro del sistema, con un nombre de usuario y contraseña, para poder utilizar los demás casos de uso. También se identificaron dos tipos de actores primarios que interactúan con el sistema: administrador tipo A y tipo B. Ambos administradores podrán realizar operaciones de alta, baja, cambios y consultas en los cuatro casos de uso principales a excepción del administrador tipo B quién no podrá gestionar usuarios. El SIP-CBI, actores primarios y casos de uso se representan gráficamente en la Figura 6.1.1.1.

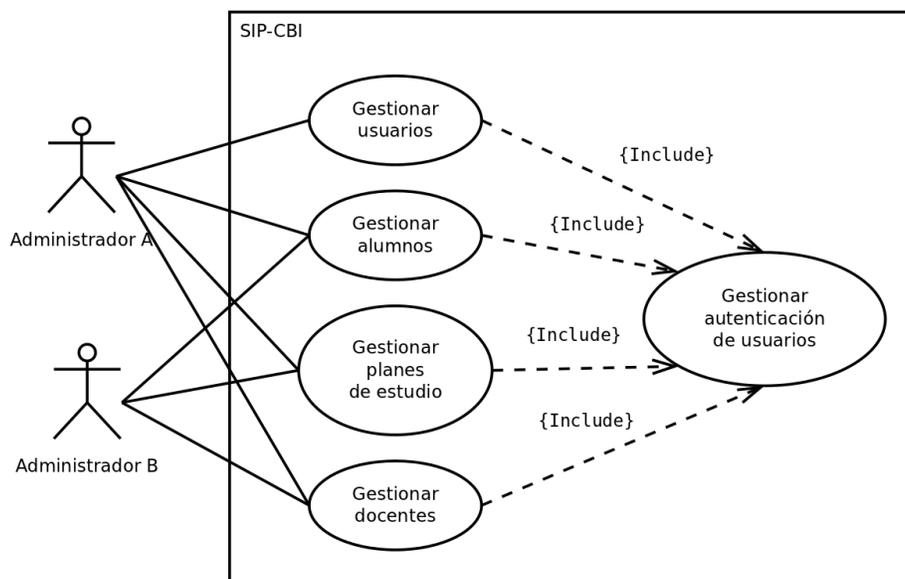


Figura 6.1.1.1: Diagrama de casos de uso general del SIP-CBI.

Cada caso de uso representado en la Figura 6.1.1.1 contiene subcasos de uso que representan las operaciones alta, baja, cambios y consultas. En la Figura 6.1.2.1 se muestran los subcasos de uso que forman el caso "Gestionar alumnos".

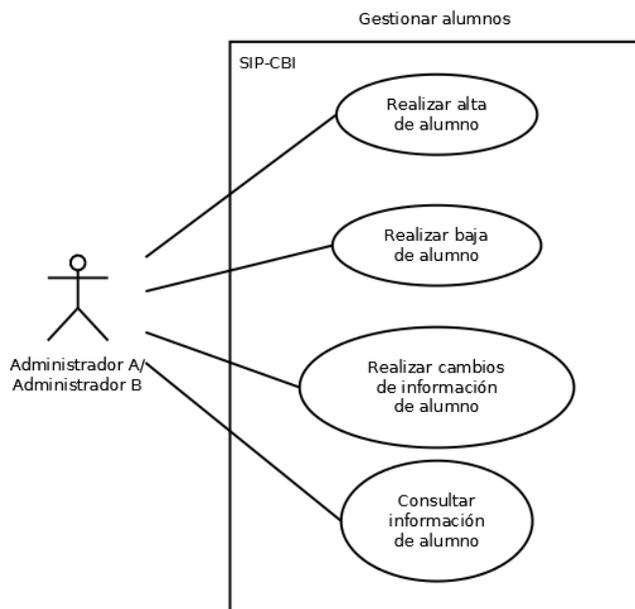


Figura 6.1.2.1: Subcasos de uso del caso de uso "Gestionar alumnos".

## 6.1.2 Escenarios

El siguiente paso en nuestro diseño fue determinar los escenarios en base a los casos de uso planteados anteriormente. A cada escenario se le asignó una clave y un número. Los escenarios del SIP-CBI son:

- GAU: Gestionar autenticación de usuarios.
  - GAU1: Iniciar sesión de usuario.
  - GAU2: Finalizar sesión de usuario.
- GU: Gestionar usuarios.
  - GU1: Realizar alta de usuario.
  - GU2: Realizar baja de usuario.
  - GU3: Realizar cambios de información de usuario.
  - GU4: Consultar información de usuario.

- GA: Gestionar alumnos.
  - GA1: Realizar alta de alumno.
  - GA2: Realizar baja de alumno.
  - GA3: Realizar cambios de información de alumno.
  - GA4: Consultar información de alumno.
- GPE: Gestionar planes de estudios.
  - GPE1: realizar alta de un plan de estudios.
  - GPE2: realizar baja de un plan de estudios.
  - GPE3: realizar cambios de información un plan de estudios.
  - GPE4: Consultar información de un plan de estudios.
- GD: Gestionar docentes.
  - GD1: Realizar alta de docente.
  - GD1: Realizar baja de docente.
  - GD1: Realizar cambios en información de docente.
  - GD1: Consultar información de docente.

La Figura 6.1.2.2 muestra el escenario de uso GA1: Realizar alta de usuario.

<p><b>GA: Gestionar alumnos</b></p> <p><b>GA1: Realizar alta de alumno</b></p> <p><b>Descripción</b> Caso de uso que permite realizar alta de un alumno en el sistema.</p> <p><b>Actores</b> Primarios: administrador tipo A, administrador tipo B.</p> <p><b>Disparador</b> El actor primario solicita realizar el alta de un alumno</p> <p><b>Precondiciones</b> El alumno no se encuentra registrado en el sistema.</p> <p><b>Postcondiciones</b> El alumno se encuentra registrado en el sistema.</p> <p><b>Flujo principal</b></p> <ol style="list-style-type: none"> <li>1. El actor primario ingresa al módulo para realizar la alta de un alumno.</li> <li>2. El actor primario selecciona la opción “alta de un alumno”.</li> <li>3. El sistema muestra un formulario para ingresar los datos del alumno.</li> <li>4. El actor primario ingresa los datos solicitados en el formulario.</li> <li>5. El actor primario selecciona la opción “agregar nuevo registro”.</li> <li>6. El sistema solicita confirmación.</li> <li>7. El actor primario confirma la solicitud.</li> <li>8. El sistema realiza el registro.</li> <li>9. El sistema indica al actor primario que su solicitud a tenido éxito.</li> <li>10. El sistema indica al actor primario si desea realizar otro registro.</li> <li>11. El actor primario sale del módulo.</li> </ol> <p><b>Flujos alternativos</b></p> <ol style="list-style-type: none"> <li>3.1. El actor primario cancela la operación. <ol style="list-style-type: none"> <li>3.1.1. Regresa al punto 2 del flujo principal.</li> </ol> </li> <li>4.1. El actor primario cancela la operación. <ol style="list-style-type: none"> <li>4.1.1. Regresa al punto 3 del flujo principal.</li> </ol> </li> <li>7.1. El sistema indica que ya existe un registro igual. <ol style="list-style-type: none"> <li>7.1.1. El sistema muestra el formulario con los datos suministrados.</li> <li>7.1.2. El actor primario suministra un registro con datos distintos.</li> <li>7.1.3. Regresa al punto 5 del flujo principal.</li> </ol> </li> </ol> <p><b>Requerimientos no funcionales</b> Este proceso no debe de tardar más de 2 minutos.</p>
---

Figura 6.1.2.2: Escenario “GA1: Realizar alta de alumno”.

### 6.1.3 Tarjetas CRC

Una vez definidos los escenarios se procedió a determinar los prototipos de entidades que potencialmente se convertirán en clases de software. Con ayuda de las tarjetas CRC se determinaron entidades primarias y sus respectivas entidades secundarias. Las entidades son:

- Autenticación de usuarios.
- Usuarios.
- Trimestres.
- Alumnos.
- Datos personales.



Este modelo considera las correspondencias directas entre entidades. En la Figura 6.1.5.1 se muestra el modelo de dominio para las entidades relacionadas con alumnos. En el caso particular de las entidades relacionadas con alumnos, estas presentan una relación de agregación, es decir, están contenidas dentro de la entidad alumnos, pero no son parte fundamental para su existencia.

### 6.1.5 Diagrama de secuencias

A continuación se elaboraron diagramas de secuencias que muestran la interacción entre los objetos del modelo de análisis. En estos diagramas se representa la comunicación entre objetos a través del tiempo. Los elementos mostrados en estos diagramas provienen de los escenarios de alta, baja, cambios y consultas y el modelo de dominio. En la Figura 6.1.6.1 se muestra el diagrama de secuencias asociado a la operación *alta*.

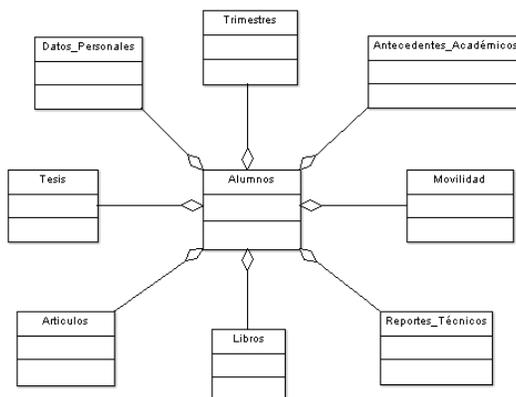


Figura 6.1.5.1: Modelo de dominio de las entidades asociadas a la entidad alumno.

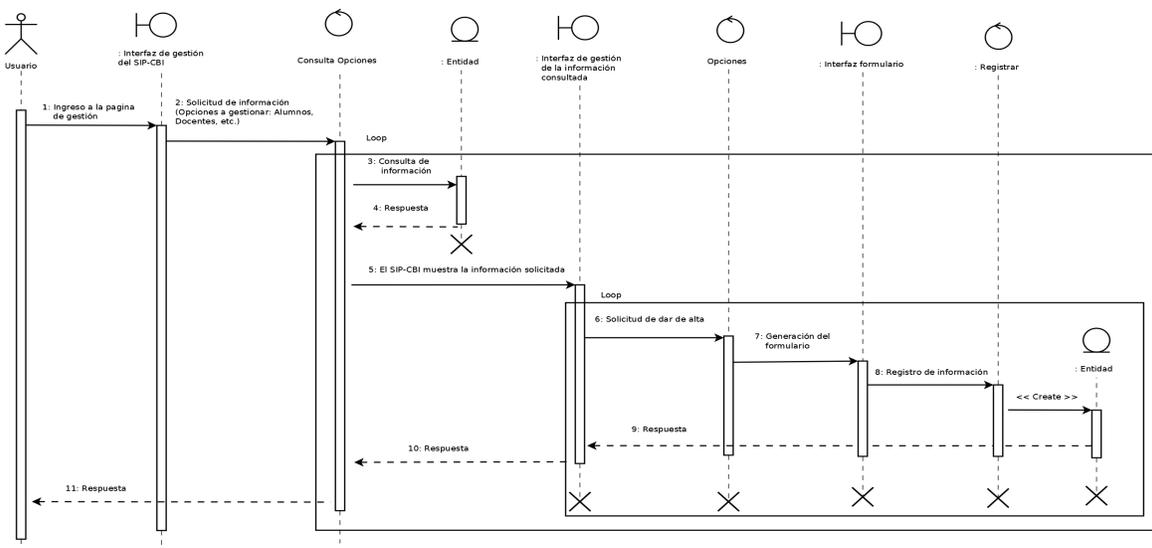


Figura 6.1.6.1: Diagrama de secuencias para la operación “alta”.

### 6.1.6 Modelo de clases

Con ayuda del modelo de dominio y los diagramas de secuencias se construyó el diagrama de clases del SIP-CBI. En este diagrama se muestran las clases, relaciones, métodos, atributos, y navegabilidad entre todas las entidades del sistema. Las clases definidas para el SIP-CBI son: *Autenticacion*, *Usuarios*, *Trimestres*, *Alumnos*, *DatosPersonales*, *AntecedentesAcad*, *Tesis*, *Libros*, *Articulos*, *Posgrados*, *Movilidad*, *PlanEstudios*, *Docentes* y *Proyectos*.

A continuación describiremos las clases determinadas para el módulo de alumnos:

- **Alumnos:** Es una clase principal o primaria del SIP-CBI. Nos permite realizar operaciones de alta, baja, cambios y consultas de alumnos en la base de datos. Contiene los atributos: *matricula*, *trimestreingreso*, *nombre*, *apellidop*, *apellidom* y *foto*. *Alumnos* contiene a las clases *DatosPersonales*, *Trimestres*, *AntecedentesAcad*, *Movilidad*, *ReporteTecnico*, *Libros*, *Articulos* y *Tesis* mediante relaciones de agregación.
- **Trimestres:** Clase secundaria que gestiona los trimestres. Tiene una relación de agregación con la clase *Alumnos*. Esta formada por los atributos: *trimestre*, *alias*, *fechainicio*, *fechafin* y *comentario*.
- **DatosPersonales:** Esta clase permite realizar operaciones de alta, baja, cambios y consultas en los datos personales de los alumnos que serán almacenados en la base de datos. Tiene una relación de agregación con las clases *Alumnos* y *Docentes*. Los atributos asociados esta clase son: *direccion*, *telefono*, *celular*, *correoinst*, *correoext*, *nacionalidad*, *fechanacimiento*, *paisnacimiento*, *lugarnacimiento*, *curp*, *cvupdf* e *idiompdf*.

- **AntecedentesAcad:** Permite las operaciones de alta, baja, cambios y consultas en los antecedentes académicos de los alumnos. Tiene una relación de agregación con las clases *Alumnos* y *Docentes*. Los atributos que la forman son: *grado*, *pais*, *AAegreso*, *promedio*, *institucion* y *titulopdf*.
- **Tesis:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de tesis del alumno. Tiene una relación de agregación con la clase *Alumnos* y *Docentes*. Contiene a los atributos: *titulo*, *coasesores*, *fecha* y *tesispdf*.
- **ReporteTecnico:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de reportes técnicos del alumno. Tiene una relación de agregación con la clase *Alumnos*. Esta formada por los atributos: *titulo*, *instanciapresentarreporte*, *descripcion*, *numeropaginas*, *fecha*, *objetivosreporte*, *coautores* y *reportepdf*.
- **Libros:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de libros del alumno. Tiene una relación de agregación con la clase *Alumnos*. Esta formada por los atributos: *ISBN*, *titulo*, *editorial*, *edicion*, *AApublicacion*, *volumen*, *paginas*, *tiraje*, *idioma*, *palabrasclave* y *libropdf*.
- **Artículos:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de artículos del alumno. Tiene una relación de agregación con la clase *Alumnos*. Contiene los atributos: *tipoarticulos*, *titulo*, *AApublicacion*, *volumen*, *numerovolumen*, *revistapublicacion*, *delapag*, *alapag*, *coautores*, *posicionautor* y *articulo*pdf.
- **Movilidad:** Esta clase permite realizar operaciones del alta, baja, cambios y consultas en la información de movilidad de un alumno. Tiene una relación de agregación con la clase *Alumno*. Esta formada por los atributos: *institucion*, *direccion*, *fechainicio*, *fechafin*, y *movilidadpdf*.

En la Figura 6.1.6.2 se muestra el diagrama de clases de las clases asociadas al módulo de alumnos.

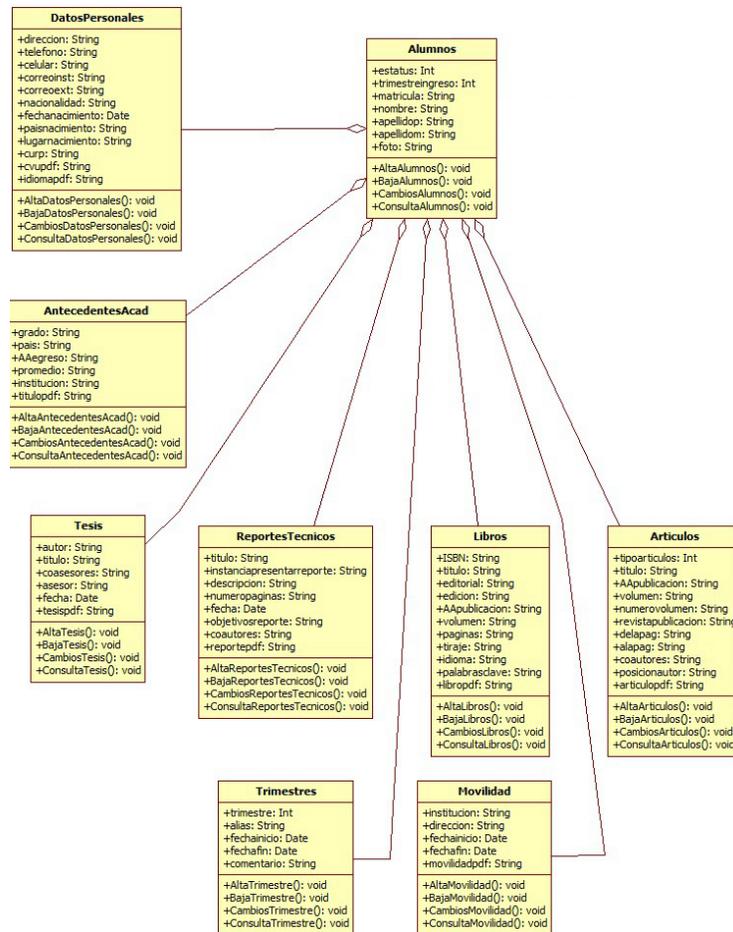


Figura 6.1.6.2: Diagrama de clases para alumnos.

La Figura 6.1.6.3 muestra el diagrama de clases general del SIP-CBI.

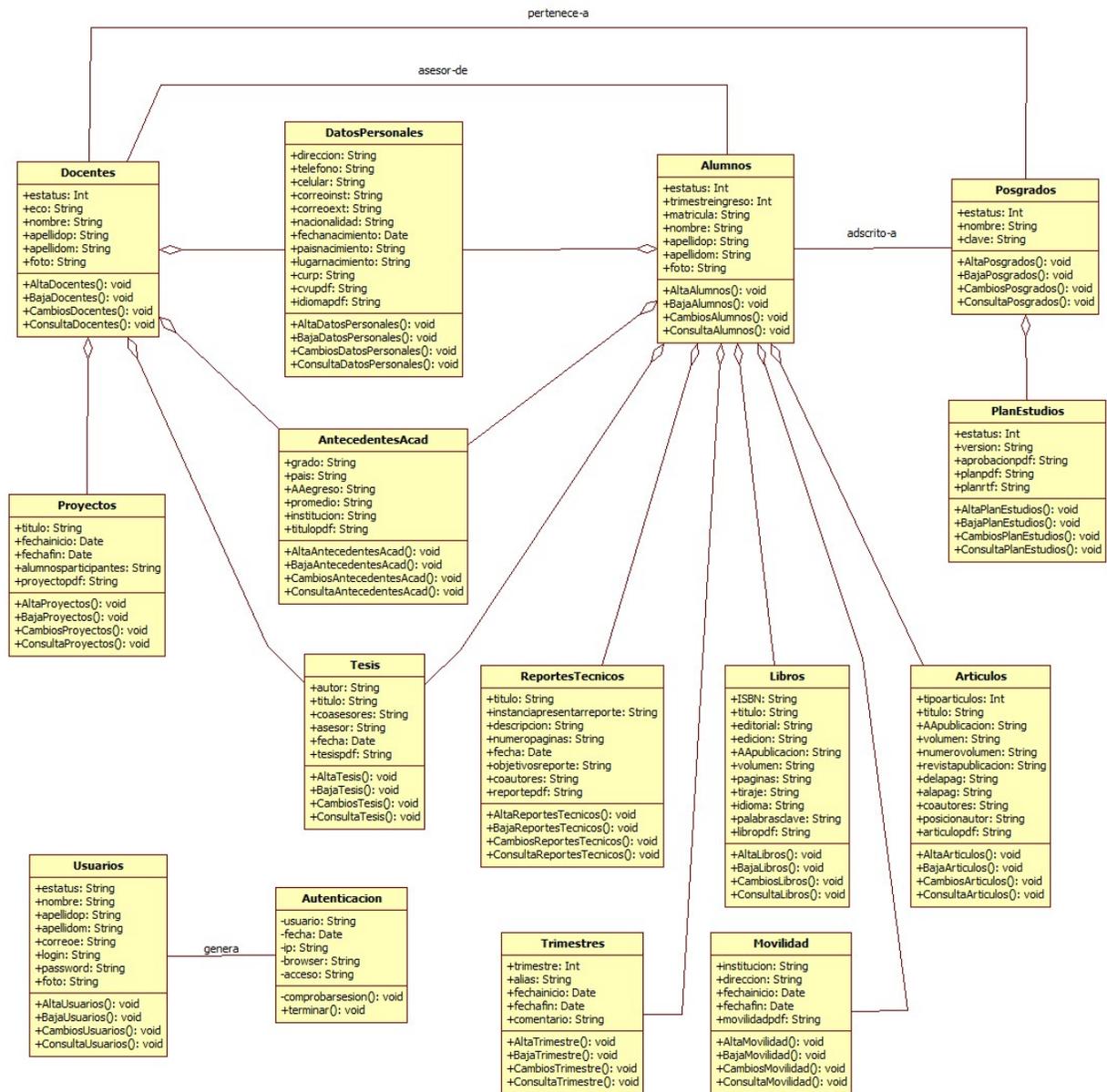


Figura 6.1.6.3: Diagrama de clases general del SIP-CBI.

### 6.1.7 Modelo de la base de datos

Para construir el modelo de la base de datos a partir del modelo de dominio y el diagrama de clases se aplicó la siguiente estrategia:

1. Convertir cada entidad del modelo de dominio en una tabla.
2. Convertir los atributos de las clases en renglones (campos) de la tabla.
3. Especificar llaves primarias e índices en cada tabla.
4. Relacionar las tablas de manera adecuada para evitar relaciones del tipo M:N. En las relaciones 1:N la relación se representa con una llave foránea que se inserta en la tabla que tiene la cardinalidad N. En caso de tener cardinalidades múltiples en ambos lados, M:N, debe introducirse una tabla adicional que contenga las llaves primarias de ambas entidades.

Las tablas de bases de datos asociadas al módulo de alumnos son:

A continuación se describe cada una de la entidades de bases de datos que forman el SIP-CBI:

- **trimestres:** Tabla que almacena la información de los trimestres. Esta formada por los campos: *id\_trimestres*, *trimestre\_trimestres*, *alias\_trimestres*, *fechainicio\_trimestres*, *fechafin\_trimestres* y *comentario\_trimestres*.
- **trimestresalumnos:** Contiene la información de la actividad trimestral del alumno. La tabla esta formada por los campos: *id\_trimestresalumnos*, *id\_trimestres\_trimestresalumnos*, *id\_alumnos\_trimestresalumnos* e *id\_estatus\_trimestresalumnos*. Los campos *id\_trimestres\_trimestresalumnos*, *id\_alumnos\_trimestresalumnos* e *id\_estatus\_trimestresalumnos* son llaves foráneas que direccionan hacia el *id* de las tablas *trimestres*, *alumnos* y *estatus*.
- **alumnos:** Permite almacenar la información de los alumnos. Contiene los campos: *id\_alumnos*, *matricula\_alumnos*, *id\_trimestreingreso\_alumnos*, *nombre\_alumnos*, *apellidop\_alumnos*, *apellidom\_alumnos* y *foto\_alumnos*. El campo *id\_trimestreingreso\_alumnos* es una llave foránea que apunta hacia el campo *id\_trimestres* de la tabla *trimestres*.
- **alumnoadscrito:** Almacena la información que relaciona a un alumno con un posgrado. La tabla esta formada por los campos: *id\_alumnoadscrito*, *id\_alumnos\_alumnoadscrito* e *id\_posgrados\_alumnoadscrito*. Los campos *id\_alumnos\_alumnoadscrito* e *id\_posgrados\_alumnoadscrito* son llaves foráneas de los *id* de las tablas *alumnos* y *posgrados*.
- **alumnoplanestudios:** Contiene la información que relaciona a un alumno con un plan de

estudios de algún posgrado. Esta tabla se forma con los campos: *id\_alumnoplanestudios*, *id\_alumnoadscrito\_alumnoplanestudios* e *id\_planestudios\_alumnoplanestudios*. Los campos *id\_alumnoadscrito\_alumnoplanestudios* e *id\_planestudios\_alumnoplanestudios* funcionan como llaves foráneas que direccionan al *id* de las tablas *alumnoadscrito* y *planestudios*.

- **datospersonales:** Esta tabla almacena datos personales adicionales de alumnos y docentes. Los campos asociados esta tabla son: *id\_datospersonales*, *direccion\_datospersonales*, *telefono\_datospersonales*, *celular\_datospersonales*, *correoinst\_datospersonales*, *correoext\_datospersonales*, *nacionalidad\_datospersonales*, *fechanacimiento\_datospersonales*, *paisnacimiento\_datospersonales*, *lugarnacimiento\_datospersonales*, *curp\_datospersonales*, *cvupdf\_datospersonales* e *idiomapdf\_datospersonales*.
- **alumnospersonales:** Esta tabla permite relacionar la tabla *alumnos* con la tabla *datospersonales*. Esta formada por los campos: *id\_alumnospersonales*, *id\_alumnos\_alumnospersonales* e *id\_datospersonales\_alumnospersonales*. Los campos *id\_alumnos\_alumnospersonales* e *id\_datospersonales\_alumnospersonales* son llaves foráneas provenientes de las *id* de las tablas *alumnos* y *datospersonales*, respectivamente.
- **antecedentesacad:** Tabla que almacena los antecedentes académicos de *alumnos* y *docentes*. Esta formada por los campos: *id\_antecedentesacad*, *grado\_antecedentesacad*, *pais\_antecedentesacad*, *AAegreso\_antecedentesacad*, *promedio\_antecedentesacad*, *institucion\_antecedentesacad* y *titulopdf\_antecedentesacad*.
- **alumnosantecedentes:** Relaciona las tablas *alumnos* y *antecedentesacad*. Contiene los campos: *id\_alumnosantecedentes*, *id\_alumnos\_alumnosantecedentes* e *id\_antecedentesacad\_alumnosantecedentes*. Los campos *id\_alumnos\_alumnosantecedentes* e *id\_antecedentesacad\_alumnosantecedentes* son llaves foráneas provenientes de las *id* de las tablas *alumnos* y *antecedentesacad*, respectivamente.
- **tesis:** Almacena información relacionada con las tesis de alumnos. Contiene los campos: *id\_tesis*, *id\_estatus\_tesis*, *id\_alumnos\_tesis*, *id\_docentes\_tesis*, *titulo\_tesis*, *coasesores\_tesis*, *fecha\_tesis* y *tesispdf\_tesis*. Los campos *id\_estatus\_tesis*, *id\_alumnos\_tesis* e *id\_docentes\_tesis* son llaves foráneas del *id* las tablas *estatus*, *alumnos* y *docentes*, respectivamente.
- **reportetecnico:** Tabla que almacena información sobre los reportes técnicos del alumno. Esta formada por los campos: *id\_reportetecnico*, *id\_alumnos\_reportetecnico*, *titulo\_reportetecnico*, *instanciapresentarreporte\_reportetecnico*, *descripcion\_reportetecnico*, *numeropaginas\_reportetecnico*, *fecha\_reportetecnico*, *objetivosreporte\_reportetecnico*, *coasesores\_reportetecnico* y *reportepdf\_reportetecnico*.

El campo *id\_alumnos\_reportetecnico* es una llave foránea que apunta hacia el *id* de la tabla *alumnos*.

- **libros:** Almacena la información de los libros del alumno. Esta formada por los campos: *id\_libros*, *id\_alumnos\_libros*, *ISBN\_libros*, *titulo\_libros*, *editorial\_libros*, *edicion\_libros*, *AApublicacion\_libros*, *volumen\_libros*, *paginas\_libros*, *tiraje\_libros*, *idioma\_libros*, *palabrasclave\_libros* y *libropdf\_libros*. El campo *id\_alumnos\_libros* es una llave foránea que apunta hacia el *id* de la tabla *alumnos*.
- **articulos:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de artículos del alumno. Tiene una relación de agregación con la clase *Alumnos*. Contiene los atributos: *id\_articulos*, *id\_tipoarticulos\_articulos*, *id\_alumnos\_articulos*, *titulo\_articulos*, *AApublicacion\_articulos*, *volumen\_articulos*, *numerovolumen\_articulos*, *revistapublicacion\_articulos*, *delapag\_articulos*, *alapag\_articulos*, *coautores\_articulos*, *posicionautor\_articulos* y *articulo-pdf\_articulos*. Los campos *id\_articulos* e *id\_tipoarticulos\_articulos* son llaves foráneas de los *id* de las tablas *alumnos* y *tipoarticulos*, respectivamente.
- **tipoarticulos:** Tabla que permite asignar un tipo de artículo. Esta formada por los campos *id\_tipoarticulo*, *tipo\_tipoarticulo*, *clave\_tipoarticulo* y *nombre\_tipoarticulo*.
- **posgrados:** Tabla que contiene el nombre y clave de los posgrados. Esta formada por los campos: *id\_posgrados*, *id\_estatus\_posgrados*, *nombre\_posgrados* y *clave\_posgrados*. El campo *id\_estatus\_posgrados* es una llave foránea que apunta hacia el *id* de la tabla *estatus*.
- **movilidad:** Contiene la información sobre las movilidades del alumno. Esta formada por los campos: *id\_movilidad*, *id\_alumnos\_movilidad*, *institucion\_movilidad*, *direccion\_movilidad*, *fechainicio\_movilidad*, *fechafin\_movilidad*, y *movilidadpdf\_movilidad*. El campo *id\_alumnos\_movilidad* funciona como llave foránea hacia el *id* de la tabla *estatus*.

La Figura 6.1.7.1 muestra las tablas de bases de datos obtenidas a partir del modelo de clases presentado en la Figura 6.1.6.2.





### 6.1.8 Arquitectura

La implementación del SIP-CBI se llevó a cabo mediante el uso del patrón MVC. El MVC es un patrón de arquitectura de software que permite separar la lógica de negocio de la interfaz gráfica de usuario. Agiliza la funcionalidad, mantenibilidad y escalabilidad de un sistema.

El MVC se divide en tres niveles de abstracción:

- **Modelo:** representa la lógica del negocio. Actúa como un intermediario para acceder de forma directa a la información almacenada en la base de datos.
- **Vista:** elemento encargado de mostrar la información al usuario de forma gráfica.
- **Controlador:** intermediario entre la vista y el modelo. Gestiona las solicitudes del usuario hacia el modelo y regresa datos a la vista para presentarlos al usuario.

El funcionamiento del patrón MVC puede describirse de la siguiente forma:

- El usuario realiza una petición.
- El controlador recibe esta petición.
- El controlador realiza una llamada al modelo.
- El modelo interactúa con la base de datos directamente o mediante servicios web y regresa datos al controlador.
- El controlador recibe los datos y los envía a la vista.
- La vista recibe los datos, los procesa y crea una capa de abstracción para el proceso de datos y otra para generar la interfaz gráfica de usuario.
- La vista mostrará los datos procesados al usuario.

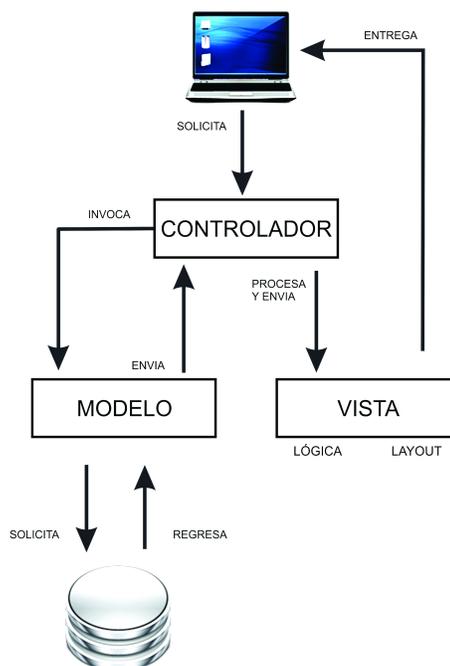


Figura 6.1.8.1: Funcionamiento del patrón MVC.

En la Figura 6.1.8.1 se muestra el flujo de datos a través del patrón MVC.

### 6.1.9 Interfaz gráfica de usuario

Con asesoría de la D.G. Perla Adriana Velasco Santos, se diseñó una interfaz gráfica dividida en cuatro secciones para organizar la información. Tres secciones se mantienen fijas: *encabezado*, *menú* y *pie de página*. La sección *contenido* permite visualizar la información solicitada al SIP-CBI. La sección *menú* es desplegable con el fin de no saturar visualmente la interfaz. En la Figura 6.1.9.1 se observa el prototipo de interfaz gráfica y en la Figura 6.1.9.2 la interfaz gráfica final.

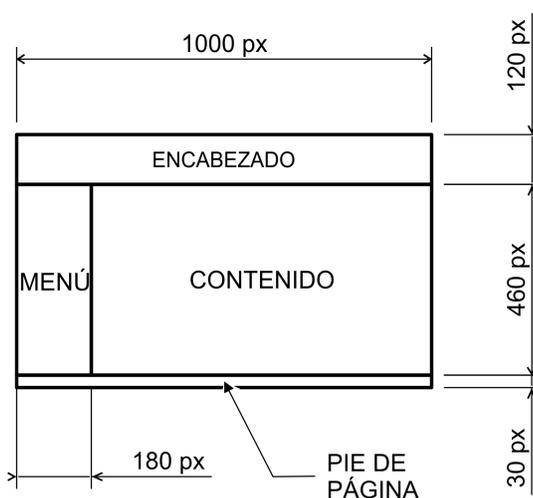


Figura 6.1.9.1: Prototipo de interfaz gráfica



Figura 6.1.9.2: Interfaz gráfica final del SIP-CBI.

Se implementó un diseño con navegabilidad de tres niveles. El menú principal permite cargar dos opciones en la zona de *contenido*:

- **Alta.** Genera un formulario para operaciones de: alta de usuarios, alumnos, docentes o posgrados. Después de procesar el formulario se muestra un mensaje de confirmación así como un menú de opciones. En la Figura 6.1.9.3 se muestra la propuesta de navegabilidad para el proceso de alta y en la Figura 6.1.9.4 la interfaz gráfica final.
- **Búsqueda.** Genera una lista de coincidencias las cuales pueden ser: datos de usuarios, alumnos, docentes o posgrados. Estos datos pueden editarse, para lo cual, se genera en otro nivel un formulario. Después de procesar el formulario se muestra un mensaje de confirmación así como un menú de opciones. En la Figura 6.1.9.5 se muestra la propuesta de navegabilidad para el proceso de alta de un alumno y en la Figura 6.1.9.6 la interfaz gráfica final.

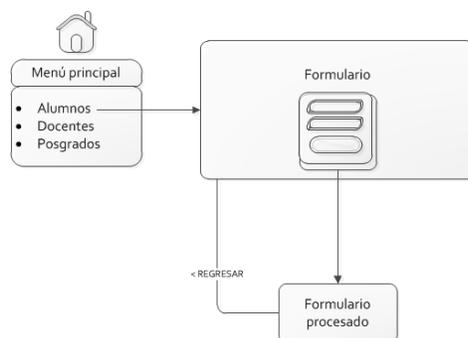


Figura 6.1.9.3: Propuesta de navegabilidad para el proceso de alta.

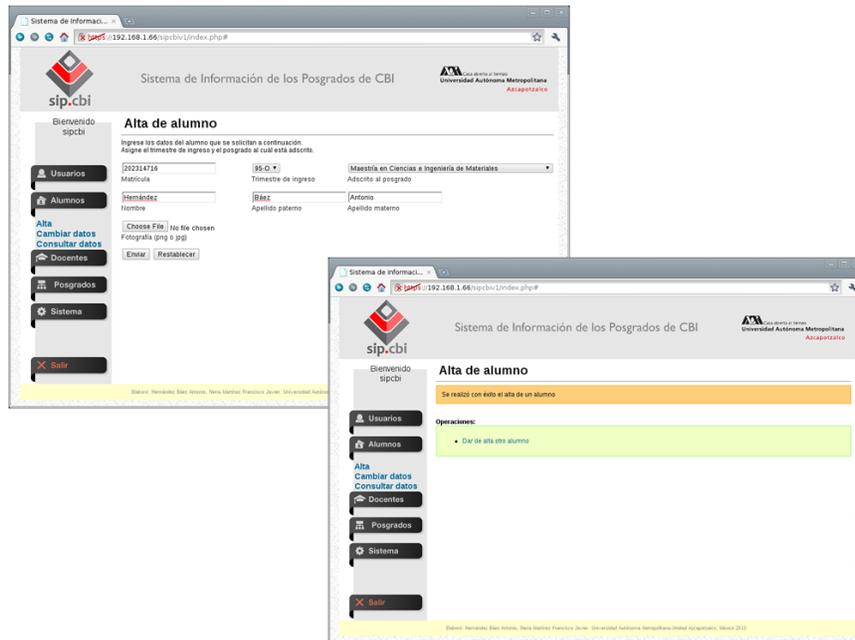


Figura 6.1.9.4: Proceso de alta visto desde la interfaz gráfica.

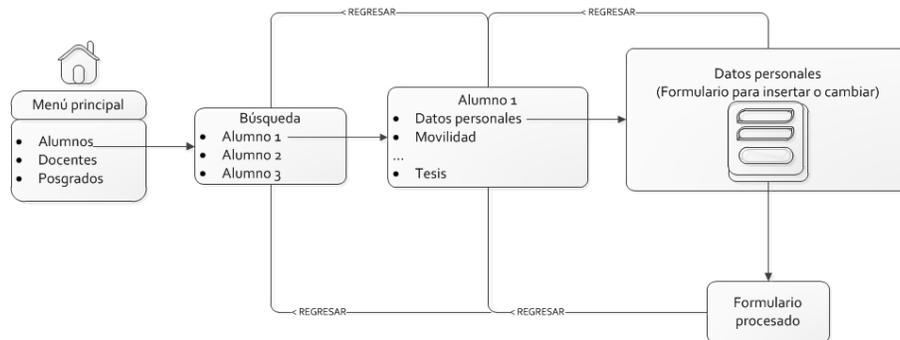


Figura 6.1.9.5: Propuesta de navegabilidad para el proceso de búsqueda y cambio de información.

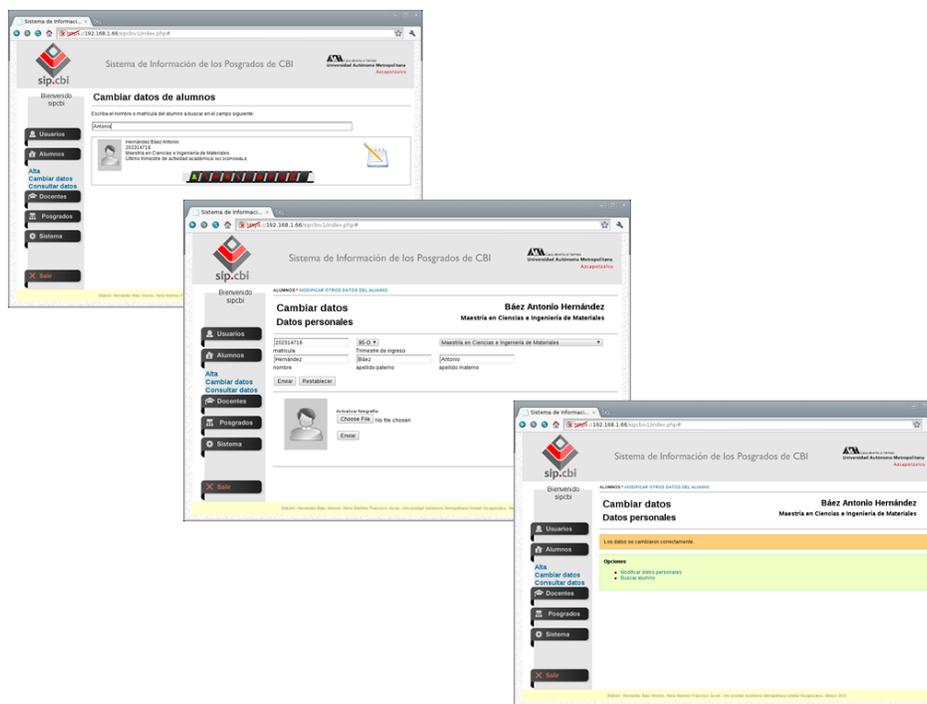


Figura 6.1.9.6: Proceso de búsqueda y cambio de información visto desde la interfaz gráfica.

## 6.2 Implementación

En esta etapa del desarrollo del software convertimos los artefactos obtenidos en la etapa de diseño en un sistema ejecutable.

En esta sección describiremos: tecnologías empleadas, software y hardware (sección 6.2.1); funcionalidades del sistema, clases y código fuente (sección 6.2.2); y pruebas, validaciones (sección 6.2.3).

### 6.2.1 Tecnología empleada

#### 6.2.1.1 Hardware

Para implementar el SIP-CBI se utilizó el siguiente hardware:

- Equipo 1: *Laptop Compaq-Presario CQ43*, procesador *Intel® Celeron® Dual-Core T3500 2.10 GHz*, 6 GB en RAM, disco duro de 320 GB, sistema operativo *Windows 7*.
- Equipo 2: Computadora de escritorio con procesador *Intel® Pentium® Dual Core E5200*

2.50 GHz, 2 GB en RAM, disco duro de 400 GB, sistema operativo *Windows XP SP2*.

### 6.2.1.2 Software

EL SIP-CBI fue implementado en software libre. A continuación se menciona el software utilizado en la implementación y los criterios de selección:

- Infraestructura *LAMP* (*Linux*, *Apache*<sup>19</sup>, *MySQL*, *PHP*). La infraestructura LAMP para el SIP-CBI esta formada por:
  - Sistema operativo *Debian 6* con *kernel 2.6.32.-5-486*, licencia *GNU/GPL* versión 3. Este sistema operativo ofrece corregir los errores de seguridad en sus paquetes mediante su equipo de auditoria de seguridad<sup>20</sup> y el denominado *Debian Security Tracker*<sup>21</sup>. Ofrece un servicio de listas de correo en donde informa sobre los últimos avisos de seguridad y comandos mínimos para obtener las últimas actualizaciones de sus paquetes<sup>22</sup>. La versión utilizada fue la más reciente al momento de iniciar la implementación.
  - Servidor web *HTTP Apache server 2.2.16-6*, licencia *Apache License 2.0*<sup>23</sup>. Uno de los servidores más utilizados a nivel mundial. De acuerdo al sitio web *news.netcraft.com* el 56% de 625,329,303 sitios web encuestados, en noviembre de 2012, utiliza *Apache* como servidor web<sup>24</sup>.
  - Gestor de bases de datos *MySQL server 5.1.49-3*, licencia *GNU/GPL* versión 2. Posee un amplio subconjunto de sentencias del lenguaje *SQL*, soporta transacciones y llaves foráneas, conectividad segura y la búsqueda e indexación de campos de texto.
  - Lenguaje de programación *PHP server 5.3.3-7+squeeze1*, licencia *PHP License* versión 3.0<sup>25</sup>. Fue seleccionado para desarrollar el SIP-CBI por ser un lenguaje orientado al desarrollo de aplicaciones web dinámicas, tiene capacidad de conexión con el gestor de bases de datos *MySQL*, permite utilizar técnicas de programación orientada a objetos y permite escribir código ordenado, estructurado y manejable (ej. patrón MVC).
- Seguridad

---

19 *HTTP Apache*: Servidor web desarrollado por la *Apache Software Foundation*.

20 <http://www.debian.org/security/audit/>

21 <http://security-tracker.debian.org/>

22 <http://www.debian.org/security/index.es.html#keeping-secure>

23 *Apache License 2.0*: Licencia de software libre creada por la *Apache Software Foundation* compatible con la *GNU General Public License*.

24 <http://news.netcraft.com/archives/2012/11/01/november-2012-web-server-survey.html>

25 *PHP License*: Licencia de código abierto creada por el *PHP Group*.

- Cifrado de contraseñas mediante el algoritmo *sha512*<sup>26</sup>. El algoritmo *MD5*<sup>27</sup> es utilizado en gran variedad de aplicaciones de seguridad y para verificar la integridad de archivos. En marzo de 2005 se demostró una colisión de *hash* provocada por la generación de dos llaves públicas distintas con el mismo *MD5 hash*. Por tal motivo se decidió cambiar el algoritmo de cifrado de contraseñas del SIP-CBI por *sha512*, el cuál, hasta el momento de desarrollar este proyecto terminal no ha presentado colisiones de *hash* y está soportado por *PHP* [10].
- Capa segura *OpenSSL 0.9.8o4squeeze7*, licencia *Apache License 1.0*. Este software permite operar el servidor web *HTTP Apache* sobre una capa de seguridad cifrada.

La infraestructura *LAMP*, la capa de seguridad y la plataforma de desarrollo se instalaron en una máquina virtual en el software *VirtualBox 4.1.8* (licencia *GNU/GPL* versión 2) con el objetivo de implementar en el sistema operativo invitado (*Debian*) sin necesidad de modificar las particiones de los discos duros del hardware seleccionado para el desarrollo.

Las licencias *GNU/GPL*, *Apache License 2.0* permiten usar los programas para cualquier propósito.

### 6.2.2 Funcionalidad

#### 6.2.2.1 Capa SSL

Para comprobar que el servidor web se encuentra ya sobre la capa de conexión segura, se solicitó a través de un navegador (en este caso particular *Chromium*) la *URL* con el protocolo *HTTPS*. Por tratarse de un certificado autofirmado el navegador enviará un mensaje de advertencia indicando que “el certificado del servidor no es de confianza”. Al solicitar los datos del certificado nos muestra los datos suministrados durante la generación del certificado, vigencia y las huellas digitales cifradas con el algoritmo *sha* (ver Figura 6.2.2.1.1).

---

<sup>26</sup> *SHA*: Es un sistema de funciones *hash* criptográficas.

<sup>27</sup> *MD5 (Message-Digest Algorithm 5)*: Algoritmo de reducción criptográfico de 128 bits.

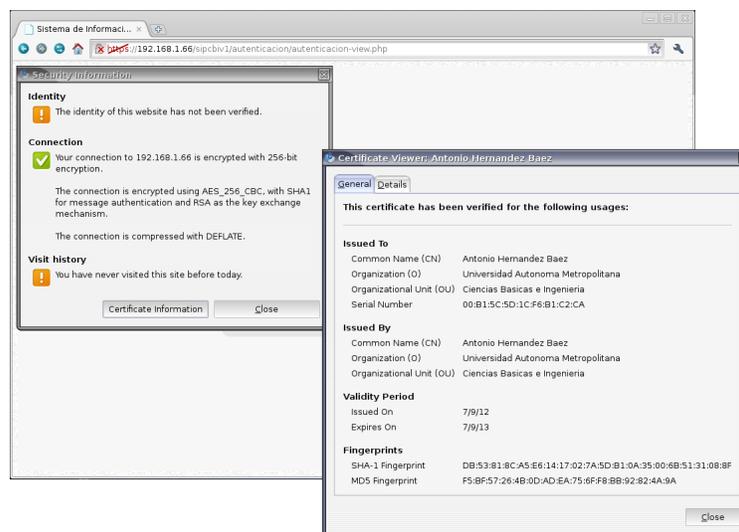


Figura 6.2.2.1.1: Verificación del funcionamiento de la capa SSL por medio del navegador *Chromium*.

En el manual técnico se describe la instalación, configuración y generación del certificado SSL.

### 6.2.2.2 Implementación del modelo

Se implementó un modelo para cada módulo del sistema. Cada modelo incluye las operaciones *select*, *update*, *insert* y *delete* realizadas a la base de datos. La Figura 6.2.2.2.1 muestra la interfaz gráfica para la operación de alta de un alumno. Esta interfaz gráfica presenta el formulario con los datos solicitados por el SIP-CBI. Al ser procesado el formulario se muestra un mensaje como se observa en la Figura 6.2.2.2.2. En el Listado 1 se muestra parcialmente del código perteneciente a la clase *Alumnos()* contenida en el archivo *alumnos-model.php* que corresponde a la operación de alta de un alumno.

## 6 Desarrollo

The screenshot shows a web browser window with the URL `http://192.168.1.66/sipcbv1/index.php#`. The page title is "Sistema de Información de los Posgrados de CBI". The main heading is "Alta de alumno". Below the heading, there is a sub-heading "Ingrese los datos del alumno que se solicitan a continuación. Asigne el trimestre de ingreso y el posgrado al cuál está adscrito." The form contains several input fields: "Matrícula" (202314716), "Trimestre de Ingreso" (95-0), "Adscrito al posgrado" (Maestría en Ciencias e Ingeniería de Materiales), "Nombre" (Hernández), "Apellido paterno" (Báez), and "Apellido materno" (Antonio). There is also a "Fotografía" field with a "Choose File" button and a "No file chosen" message. At the bottom of the form, there are "Enviar" and "Restablecer" buttons. On the left side, there is a navigation menu with buttons for "Usuarios", "Alumnos", "Docentes", "Posgrados", "Sistema", and "Salir". The footer contains the text: "Dibardí, Hernández Báez Antonio, Peña Martínez Francisco Javier. Universidad Autónoma Metropolitana Unidad Azcapotzalco, México 2013."

Figura 6.2.2.2.1: Interfaz gráfica para el proceso de alta de un alumno.

The screenshot shows the same web browser window as Figure 6.2.2.2.1. The page title is "Sistema de Información de los Posgrados de CBI". The main heading is "Alta de alumno". Below the heading, there is a sub-heading "Se realizó con éxito el alta de un alumno" displayed in a yellow box. Below this, there is a section titled "Operaciones:" with a list containing one item: "Dar de alta otro alumno" displayed in a green box. The navigation menu on the left side is the same as in Figure 6.2.2.2.1. The footer contains the text: "Dibardí, Hernández Báez Antonio, Peña Martínez Francisco Javier. Universidad Autónoma Metropolitana Unidad Azcapotzalco, México 2013."

Figura 6.2.2.2.2: La interfaz muestra un mensaje que indica el procesamiento del formulario de manera exitosa.

```

function altaalumnos($trimestre, $matricula, $nombre, $ap, $am, $posgrado,$foto)
{
    if($_FILES["foto"]["error"] != UPLOAD_ERR_OK)
    {
        $sql="INSERT INTO sipcbi.alumnos (id alumnos, id trimestreingreso alumnos, matricula alumnos, nombre alumnos,
apellidop_alumnos, apellidom_alumnos) VALUES (NULL, ".$trimestre.", ".$matricula.", ".$nombre.", ".$ap.", ".$am."");
mysql_query($sql,Conectar::con());

        $sqlultimoid="SELECT MAX(id_alumnos) FROM sipcbi.alumnos";
        $arregloutimoid=mysql_query($sqlultimoid,Conectar::con());
        $elementoutimoid = mysql_fetch_array($arregloutimoid);
        $ultimoid=$elementoutimoid[0];
        $sql2="INSERT INTO sipcbi.alumnoadscrito (id_alumnoadscrito, id_alumnos_alumnoadscrito, id_posgrados_alumnoadscrito)
VALUES (NULL, ".$ultimoid.", ".$posgrado."");
mysql_query($sql2, Conectar::con());
mysql_close();
    }
    else
    {
        $sql="INSERT INTO sipcbi.alumnos (id alumnos, id trimestreingreso alumnos, matricula alumnos, nombre alumnos,
apellidop_alumnos, apellidom_alumnos) VALUES (NULL, ".$trimestre.", ".$matricula.", ".$nombre.", ".$ap.", ".$am."");
mysql_query($sql,Conectar::con());
        $sqlultimoid="SELECT MAX(id_alumnos) FROM sipcbi.alumnos";
        $arregloutimoid=mysql_query($sqlultimoid,Conectar::con());
        $elementoutimoid = mysql_fetch_array($arregloutimoid);
        $ultimoid=$elementoutimoid[0];
        $sql2="INSERT INTO sipcbi.alumnoadscrito (id_alumnoadscrito, id_alumnos_alumnoadscrito, id_posgrados_alumnoadscrito)
VALUES (NULL, ".$ultimoid.", ".$posgrado."");
mysql_query($sql2, Conectar::con());

        $tipomime=$_FILES['foto']['tmp_name'];
        $tipomime2=mime_content_type($tipomime);
        if ($tipomime2=="image/jpeg")
        {
            $extension=".jpg";
        }
        else if($tipomime2=="image/png")
        {
            $extension=".png";
        }
        else if ($tipomime2=="image/gif")
        {
            $extension=".gif";
        }
        else
        {
            $extension=".txt";
        }
        $sqlclaveposgrado="SELECT clave_posgrados FROM sipcbi.posgrados WHERE id_posgrados=".$posgrado."";
        $arregloclaveposgrado=mysql_query($sqlclaveposgrado,Conectar::con());
        $elementoclaveposgrado = mysql_fetch_array($arregloclaveposgrado);
        $claveposgrado=$elementoclaveposgrado[0];

        $AA=date ("Y");
        if(is_uploaded_file($_FILES['foto']['tmp_name']))
        {
            $nombreDirectorio = "../archivo/fotos/";

            $nombreArchivo = "FOTO-".$matricula."-".$claveposgrado."-".$AA."-".$ultimoid."".$extension;
            if(move_uploaded_file($_FILES['foto']['tmp_name'], $nombreDirectorio.$nombreArchivo))
            {
                echo "Archivo subido correctamente";
            }
            else
            {
                echo "No puede mover el archivo a su ubicación final";
            }
        }
        else
        {
            echo "No se ha podido subir el archivo";
        }
        $sqlnombearchivo="UPDATE sipcbi.alumnos SET foto_alumnos='".$nombreArchivo.'" WHERE id_alumnos=".$ultimoid."";
        mysql_query($sqlnombearchivo,Conectar::con());
        mysql_close();
    }
}

```

Listado 1: vista parcial de la clase *Alumnos* contenida en el archivo *alumnos-model.php*

### 6.2.2.3 Implementación de la vista

Se implementaron vistas para cada módulo del sistema. Las vistas incluyen: *HTML*, *JavaScript*, *Jquery*, *AJAX* y controladores propios en *PHP*. La Figura 6.2.2.3.1 muestra la interfaz gráfica para el proceso de alta de un alumno.

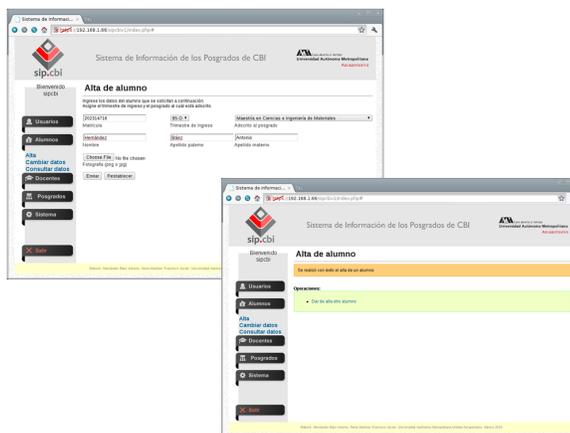


Figura 6.2.2.3.1: Interfaz gráfica del proceso de alta de un alumno.

La vista tiene en sí misma un controlador, el cual está encargado de asignar nombres a las secciones que la vista deberá mostrar al usuario. En el caso específico del alta de alumno se muestran dos secciones: la primera denominada *alta*, que presenta el formulario para dar de alta al alumno, y la segunda, llamada *mensajealta*, que muestra un mensaje que informa del registro exitoso del alumno en el sistema.

En el Listado 2 se observa parcialmente el código de del archivo *alumnos-alta-view.php*.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Vista alta</title>
...
</head>
<body>
<?php
if ($seccion=="alta")
{
...
}
else if ($seccion=="mensajealta")
{
...
}
?>
</body>
</html>
```

Listado 2: Secciones principales de la vista (archivo *alumnos-alta-view.php*).

En el Listado 3 se muestra parcialmente el código fuente que genera la vista del formulario.

```

<h1>Alta de alumno</h1><hr /><div>Ingrese los datos del alumno que se solicitan a continuación.</div><div>Asigne el trimestre
de ingreso y el posgrado al cuál está adscrito.</div><br />
<form action="alumnos-alta-controller.php?operacion=1" method="post" enctype="multipart/form-data" name="form1" id="form1"
class="formular">
<table width="780" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="33%"><input class="validate[required,custom[integer] text-input" name="matricula" type="text" id="matricula"
placeholder="Matrícula" size="20" maxlength="9" /></td>
<td width="33%">
<select name="trimestre" id="trimestre">
<option value="">---</option>
<?php foreach($arreglo as $lista): ?>
<option value="<?php echo $lista['id_trimestres']; ?>">
<?php
echo $lista['alias_trimestres'];
?></option>
<?php
endforeach;
?>
</select>
</td>
<td width="33%">
<select name="posgrado" id="posgrado">
<option value="">---</option>
<?php foreach($arreglo2 as $lista2): ?>
<option value="<?php echo $lista2['id_posgrados']; ?>">
<?php
echo $lista2['nombre_posgrados'];
?></option>
<?php
endforeach;
?>
</select>
</td>
</tr>
<tr>
<td width="33%"><label for="matricula" class="rotulosformularios">Matrícula</label></td>
<td width="33%"><label for="trimestreingreso" class="rotulosformularios">Trimestre de ingreso</label></td>
<td width="33%"><label for="posgrado" class="rotulosformularios">Adscrito al posgrado</label></td>
</tr>
<tr>
<td width="33%">&nbsp;</td><td width="33%">&nbsp;</td><td width="33%">&nbsp;</td>
</tr>
<tr>
<td width="33%"><input class="validate[required] text-input" type="text" name="nombre" type="text" id="nombre" size="20"
placeholder="Nombre" maxlength="50" /></td>
<td width="33%"><input class="validate[required] text-input" type="text" name="ap" type="text" id="apellido paterno"
size="20" placeholder="Apellido paterno" maxlength="50" /></td>
<td width="33%"><input name="am" type="text" id="apellido materno" size="20" placeholder="Apellido materno"
maxlength="50" /></td>
</tr>
<tr>
<td width="33%"><label for="nombre" class="rotulosformularios">Nombre</label></td><td width="33%"><label for="apellido
paterno" class="rotulosformularios">Apellido paterno</label></td><td width="33%"><label for="apellido materno"
class="rotulosformularios">Apellido materno</label></td></tr>
<tr>
<td width="33%">&nbsp;</td><td width="33%">&nbsp;</td><td width="33%">&nbsp;</td></tr>
<tr>
<td width="33%"><input name="foto" type="file" id="foto" size="15" name="foto" placeholder="Fotografía" /></td><td
width="33%">&nbsp;</td><td width="33%">&nbsp;</td></tr>
<tr>
<td width="33%">&nbsp;</td><td width="33%" class="rotulosformularios">Fotografía (png o jpg)</td><td width="33%">&nbsp;</td></tr>
</table>
<?php if ($cantidad==0)
{
?>
<input type="submit" name="enviar" id="enviar" value="Enviar" disabled="" />
<input type="reset" name="cancelar" id="cancelar" value="Restablecer" disabled="" />
 No se pueden agregar alumnos.<br />Es necesario dar de alta trimestres
<?php
}
else
{
?>
<input type="submit" name="enviar" id="enviar" value="Enviar" />
<input type="reset" name="cancelar" id="cancelar" value="Restablecer" />
<?php
}
?>
</form>

```

Listado 3: Sección *alta* contenida en el archivo *alumnos-alta-view.php*

En el Listado 4 se muestra el código que genera la sección *mensajealta*.

```
<h1>Alta de alumno</h1>
<hr />
<div class="aviso">
Se realizó con éxito el alta de un alumno
</div>
<br>
<br>
<strong>Operaciones:</strong>
<div class="agregar">
<ul>
<li><a href="alumnos-alta-controller.php?operacion=0">Dar de alta otro alumno</a></li>
</ul>
</div>
```

Listado 4: Sección *mensajealta* contenida en el archivo *alumnos-alta-view.php*.

### 6.2.2.4 Implementación del controlador

Se implementaron controladores para cada módulo. Estos controladores tienen las siguientes funciones:

- Crean los objetos a partir de las clases contenidas en el modelo.
- Invocan a los métodos de los objetos creados.
- Controlan las secciones que el usuario podrá observar en la interfaz gráfica.
- Reciben información proveniente de la interfaz gráfica.
- Reciben información proveniente del modelo.

En el Listado 5 se observa parcialmente el código fuente del controlador contenido en el archivo *alumnos-alta-controller.php*. Se observa: el control de las secciones con el uso de sentencia *case*, la inclusión del modelo y la vista, la creación del objeto a partir de una clase, la llamada a métodos de la clase y la destrucción del objeto.

```
<?php
$operacion=$_GET['operacion'];
switch ($operacion)
{
    case 0:
        include("alumnos-model.php");
        $seccion="alta";
        $trim = new Alumnos();
        $arreglo=$trim -> listatrimestres();
        $arreglo2=$trim -> listaposgrados();
        include("alumnos-alta-view.php");
        unset($trim);
        break;
    case 1:
        include("alumnos-model.php");
        $seccion="mensajealta";
        $alta= new Alumnos();
        $alta -> altaalumnos($_POST["trimestre"], $_POST["matricula"],$_POST["nombre"],$_POST["ap"], $_POST["am"],
        $_POST["posgrado"], $_FILES['foto']);
        include("alumnos-alta-view.php");
        unset($alta);
        break;
    ...
    ...
}
```

Listado 5: *alumnos-alta-controller.php*

### 6.2.3 Pruebas

Se realizaron pruebas de integridad referencial al modelo de bases de datos implementado. Al realizar el alta de un alumno no solo se realiza la inserción del registro en la tabla *alumnos*, también se inserta el posgrado al cual se encuentra adscrito el alumno en la tabla *alumnoadscrito*. Esto crea una relación entre las tablas *alumnos* y *posgrados* a través de la tabla *alumnoadscrito*.

Para comprobar la integridad referencial entre las tablas se realizó lo siguiente:

1. Mostrar la lista de alumnos del SIP-CBI
2. Seleccionar el *id* de un alumno.
3. Mostrar la lista de alumnos adscritos.
4. Identificar el *id* del alumno y el *id* del posgrado al cual está adscrito
5. Borrar de la tabla de *alumnos* al alumno con ayuda del *id*.

Al intentar borrar el registro del alumno seleccionado el sistema envió un mensaje de error indicando que el registro no puede ser borrado ya que forma parte de una llave foránea en la tabla *alumnoadscrito*. En la Figura 6.2.3.1 se muestra el procedimiento antes descrito y el mensaje de error al intentar borrar el registro.

```

root@debian6p: ~
File Edit View Terminal Help
mysql>
mysql>
mysql> SELECT * FROM alumnos;
+-----+-----+-----+-----+-----+-----+-----+
| id_alumnos | id_trimestreingreso_alumnos | matricula_alumnos | nombre_alumnos | apellidop_alumnos | apellidom_alumnos | foto_alumnos |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 9 | 202314716 | Antonio | Hernández | Báez | NULL |
| 2 | 66 | 95303250 | José | Hernández | Báez | NULL |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> SELECT * FROM alumnoadscrito;
+-----+-----+-----+
| id_alumnoadscrito | id_alumnos_alumnoadscrito | id_posgrados_alumnoadscrito |
+-----+-----+-----+
| 1 | 1 | 2 |
| 2 | 2 | 4 |
+-----+-----+-----+
2 rows in set (0.01 sec)

mysql> DELETE FROM posgrados WHERE id_posgrados='2';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('sipcbi`.`alumnoadscrito', CONSTRAINT `fk_alumnoadscrito_posgrados1` FOREIGN KEY (`id_posgrados_alumnoadscrito`) REFERENCES `posgrados` (`id_posgrados`) ON DELETE NO ACTION ON UPDATE NO ACTION)
mysql>

```

Figura 6.2.3.1: Prueba de integridad referencial a la tabla *alumnos*.

Del mismo modo se realizó una prueba con la tabla de posgrados. Al intentar borrar un posgrado que se encuentra relacionado con un alumno en la tabla *alumnoadscrito* el sistema enviará un mensaje de error (ver Figura 6.2.3.2).

```

root@debianopt: ~
File Edit View Terminal Help
+-----+-----+-----+-----+
| id_posgrados | id_estatus_posgrados | nombre_posgrados | clave_posgrados |
+-----+-----+-----+-----+
| 1 | 3 | Maestría en Ciencias e Ingeniería Ambientales | MCIA |
| 2 | 3 | Maestría en Ciencias de la Computación | MCC |
| 3 | 3 | Maestría en Ingeniería Estructural | MIE |
| 4 | 3 | Maestría en Ciencias e Ingeniería de Materiales | MCIM |
| 5 | 3 | Maestría en Optimización | MO |
| 6 | 3 | Maestría en Ingeniería de Procesos | MIP |
| 7 | 3 | Doctorado en Ciencias e Ingeniería (Ambiental, de Materiales) | DCIAM |
| 8 | 3 | Doctorado en Ingeniería Estructural | DIE |
| 9 | 3 | Doctorado en Optimización | DO |
| 10 | 3 | Doctorado en Ingeniería de Procesos | DIP |
+-----+-----+-----+-----+
10 rows in set (0.08 sec)

mysql> DELETE FROM posgrados WHERE id_posgrados='2';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('sipcbi`.`alumnoadscrio`, CONSTRAINT `fk_alumnoadscrio_posgrados1` FOREIGN KEY ('id_posgrados_alumnoadscrio') REFERENCES `posgrados` ('id_posgrados')) ON DELETE NO ACTION ON UPDATE NO ACTION)
mysql>

```

Figura 6.2.3.2: Prueba de integridad referencial a la tabla *posgrados*.

También se realizaron pruebas de validación al SIP-CBI. La primer prueba se realizó en el formulario de autenticación de usuarios. Se intentó acceder con los campos de usuario y contraseña vacíos pero el sistema envió un mensaje advirtiendo que faltaba completar ambos campos. En otro caso se intentó acceder con un usuario y contraseña no existentes en la tabla *usuarios*. Para este caso, el sistema envió un mensaje de error en la autenticación. En la Figura 6.2.3.3 se muestra el resultado de la prueba.

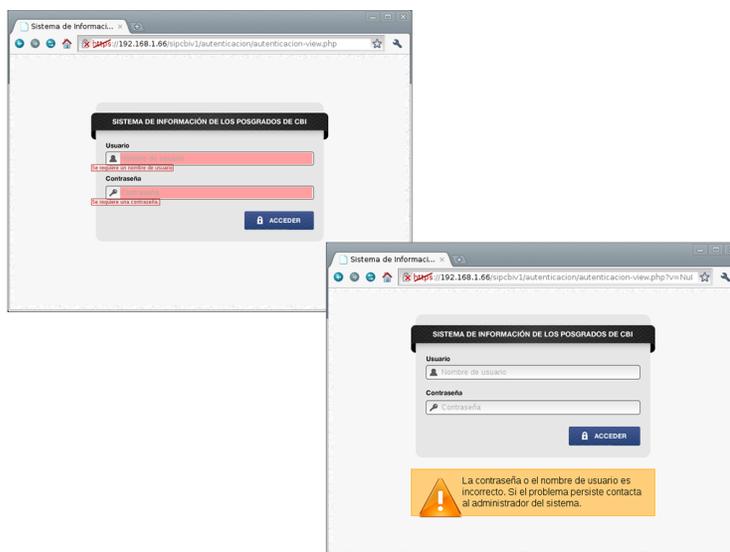


Figura 6.2.3.3: Pruebas realizadas al proceso de autenticación de usuarios.

Para el módulo de alumnos se realizó una prueba de validación en los campos. Se dejaron vacíos intencionalmente los campos matrícula, nombre y apellido paterno. Al procesar el formulario el sistema envía una serie de mensajes que indican los campos que deben llenarse (ver

Figura 6.2.3.4).

Figura 6.2.3.4: Validación de los campos matrícula, nombre y apellido paterno.

Para el formulario que actualiza los datos del alumno se realizó una prueba similar. En este caso se eliminó el nombre del alumno y al intentar procesar el formulario se mostró un mensaje que indicó el campo necesario a llenar para procesar el formulario.

Figura 6.2.3.5: Validación de datos en el formulario para cambiar datos de un alumno.

---

## 7 Conclusiones y trabajo futuro

---

El Sistema de Información de los Posgrados de CBI, SIP-CBI, es un sistema de información web, en proceso de desarrollo, que permite la gestión del archivo digital del posgrado. El archivo digital está formado por información distribuida en documentación impresa y archivos en formato electrónico sobre planes de estudio, alumnos y docentes. Se elaboró en su totalidad el diseño del sistema. Se implementaron los módulos de usuarios, alumnos, planes de estudios, docentes y adicionalmente un módulo de sistema, de los cuales, se logró implementar totalmente a usuarios, alumnos, planes de estudios y sistema.

En la Tabla 7.1 se muestran los objetivos específicos, alcance y logros.

La elaboración de este proyecto nos permitió: reforzar los conocimientos del paradigma orientado a objetos mediante el uso del proceso unificado y lenguaje unificado de modelado; administrar un sistema operativo multiusuario, basado en *Unix*, como servidor; aprender el uso del patrón MVC y manejar un *framework* para la implementación del proyecto.

Durante el proceso de implementación se descartó el uso del *framework CakePHP 2.3* debido a que: solicitaba reestructurar el diseño de la base de datos mediante asociaciones *hasOne*, *hasMany*, *belongsTo* y *hasAndBelongsToMany* (el modelo relacional elaborado en *MySQL* ya contiene estas relaciones); usar listas de control de acceso, en archivos de texto, para controlar la autenticación, comprometiendo la seguridad del sistema, y no presentar la documentación suficiente para adaptar la vista a un diseño gráfico personalizado. Estos factores fueron determinantes para detener la implementación por medio del *framework*.

Se decidió iniciar la elaboración del diseño gráfico durante la fase de modelo de dominio para establecer la forma en que la interfaz gráfica mostraría la información al usuario final (imágenes o texto). Esta decisión evitó en gran medida reescribir código de los modelos y los controladores de la vista. Además, proponer mapas de navegabilidad en la especificación complementaria permitió diseñar un modelo de interfaz gráfica fácil de usar y que fue fácilmente reproducido en todo el sistema.

El diseño del sistema se realizó en su totalidad, sin embargo, podrían implementarse otras funcionalidades :

- Establecer políticas para el manejo de los historiales de la documentación en formato *PDF*.
- Emplear *AJAX*, *JQuery* o tecnologías similares para realizar validaciones sin tener la necesidad de procesar los formularios en su totalidad.
- Implementar niveles de aislamiento para evitar que más de un usuario realice transacciones simultáneas sobre la misma información.
- Diseñar módulos para gestionar información sobre la infraestructura, servicios y seguimiento de egresados del posgrado.

## 7 Conclusiones y trabajo futuro

- Implementar la gestión de recursos en una arquitectura distribuida para manejar grandes cantidades de información.

Alcance	Objetivo	Logro
Total	Diseñar e implementar un módulo que permita la gestión de administradores.	Cumplimos con el diseño del módulo que permite gestionar a los administradores tipo A, B y C.
Total	Diseñar e implementar un módulo que permita la gestión de alumnos.	Se implementó en su totalidad un módulo para la gestión de información de alumnos.
Parcial	Diseñar e implementar un módulo que permita la gestión de docentes.	Se implementó parcialmente para garantizar el funcionamiento del módulo de alumnos.
Total	Diseñar e implementar un módulo que permita la gestión de planes de estudio de posgrado.	Se implementó en su totalidad.
Parcial	Diseñar e implementar un módulo que permita obtener informes de acuerdo a los requerimientos establecidos por el PNPC del Consejo Nacional de Ciencia y Tecnología.	Se implementó parcialmente: la consulta de documentos en formato <i>PDF</i> puede realizarse desde el módulo de alumnos o plan de estudios. El interesado puede descargar los archivos que necesite para armar el <i>CD</i> de acuerdo a los requerimientos mostrados en el Anexo.
Total	Diseñar e implementar una base de datos que soporte la información correspondiente de los módulos a desarrollar.	Se diseñó la totalidad del modelo relacional de la base de datos del sistemas.
Total	Diseñar una interfaz gráfica de usuario que permita utilizar el sistema web.	Se logró desarrollar la interfaz gráfica de usuario.
Total	Integrar los módulos y la interfaz gráfica de usuario.	El patrón arquitectónico MVC permitió la integración de los módulos y la interfaz gráfica desde el inicio de la implementación.
Total	Objetivo extra: Diseñar e implementar un módulo que permita la gestión de los estatus de usuarios, alumnos, docentes y posgrados.	Se implementó en su totalidad un módulo para la gestión de los estatus de alumnos, docentes y posgrados.
Total	Objetivo extra: Diseñar e implementar un módulo que permita la gestión de trimestres.	Se implemento en su totalidad un módulo para la gestión de los trimestres.

Tabla 7.1: Logros alcanzados durante el proyecto.

# 8 Anexo

RECTORÍA GENERAL APOYO INSTITUCIONAL AL POSGRADO CONTROL DE MEDIOS DE VERIFICACIÓN PNPC					
NOMBRE DEL PROGRAMA	UNIDAD	CONTENIDO	REFERENCIA		
			ORIENTACIÓN		
No	CARPETA		1	2	3
1	ESTRUCTURA DEL PROGRAMA				
1	Plan de estudios	1.1 Plan de estudios completo y sellado 1.2 Justificación del programa 1.3 Objetivos y metas 1.4 Perfil de ingreso 1.5 Perfil de egreso 1.6 Congruencia del plan de estudios 1.7 Mapa curricular 1.8 Actualización del plan de estudios 1.9 Opciones de graduación 1.10 Idioma			
2	Proceso de enseñanza-aprendizaje	2.1 Relación de medios de evaluación utilizados 2.2 Evaluación de los instrumentos de evaluación 2.3 Participación de estudiantes en eventos académicos internos			
2	ESTUDANTES				
3	Ingreso de estudiantes	3.1 Selección de estudiantes (requisitos de ingreso)			
4	Trajectoria escolar	4.1 Estadísticas de la Trayectoria Escolar 4.2 Opción y mecanismos para la obtención de grado 4.3 Actas de examen de grado			
5	Movilidad e intercambio de estudiantes	5.1 Resultados del programa de movilidad (nacional y en el extranjero) 5.2 Ejemplos de los mecanismos usados para fomentarla 5.3 Proyectos terminales o tesis codirigidas			
6	Tutorías y asesorías	6.1 Relación de estudiantes y tutores 6.2 Relación de estudiantes por director de tesis			
7	Dedicación exclusiva	7.1 Relación de estudiantes de TP vs TC			
3	PERSONAL ACADÉMICO				
4	Núcleo académico básico	4.1 Perfil del núcleo académico 4.2 Tiempo de dedicación 4.3 Distinciones académicas 4.4 Organización académica 4.5 Programa de superación 4.6 Evaluación académica			
8	PLAN DE MEJORAS Y ÚLTIMA EVALUACIÓN (reportes electrónicos)	Estructura del programa + FODA Estudiantes + FODA Personal Académico + FODA Infraestructura y Servicios + FODA Resultados + FODA Cooperación con otros actores de la sociedad + FODA Matriz FODA General			
9	INFORMACIÓN ESTADÍSTICA DEL PROGRAMA (REPORTES GENERADOS POR LA APLICACIÓN)	Estructura del programa Estudiantes Personal Académico Infraestructura y Servicios Resultados Cooperación con otros actores de la sociedad 20.1 Acciones basadas en las recomendaciones 20.2 Plan de Mejora Formato Libre de texto (máx. 10 cuartillas)			
	Datos del programa	Reporte de Datos Generales			
	Última evaluación	Reporte de última evaluación			
	Plan de estudios	Reporte de plan de estudios			
	Personal Académico	Reporte de académicos Reporte de LGAC Reporte de movilidad			
	Estudiantes	Reporte de estudiantes Reporte de movilidad Reporte de tutorías Reporte de LGAC de estudiantes			
	Vínculos	Reporte de vinculación			
	Productividad académica	Reportes de productividad académica Distinciones y premios Experiencia Profesional Grados Académicos			
10	PÁGINA WEB	Hacer un documento breve donde se describa el contenido de la página WEB del programa. a) Perfil de ingreso b) Objetivos generales y particulares del programa posgrado.			
9	Lineas de generación y/o aplicación del conocimiento	9.1 Congruencia de las LGAC y aplicación del conocimiento 9.2 Participación de los estudiantes en proyectos de investigación			
4	INFRAESTRUCTURA Y SERVICIOS				
10	Espacios y equipamiento	10.1 Disponibilidad de aulas (foto y descripción) 10.2 Espacios para profesores y estudiantes			
11	Laboratorios y talleres	11.1 Espacios, equipos y servicios 11.2 Materiales y suministros 11.3 Programación y utilización			
12	Información y documentación	12.1 Biblioteca e instalaciones 12.2 Asesorios y servicios			
13	Tecnologías de información y comunicación	13.1 Equipo e instalaciones 13.2 Redes 13.3 Atención y servicios			
5	RESULTADOS				
14	Trascendencia, cobertura y evolución del programa	14.1 Alcance y extensión de los resultados del programa 14.2 Cobertura del programa 14.3 Evolución y pertinencia del programa			
15	Pertinencia del programa (Seguimiento de egresados)	15.1 Satisfacción de los egresados 15.2 Proyección del programa			
16	Efectividad del posgrado	16.1 Análisis de la eficiencia terminal 16.2 Estrategias para incrementar graduación			
17	Contribución al conocimiento	17.1 Investigación y Desarrollo (Listado de productivos) 17.2 Listado de profesores en las LGAC 17.3 Dirección de tesis (participación de estudiantes) 17.4 Publicación de resultados de tesis de doctorado 17.5 Participación de estudiantes y profesores en encuentros académicos 17.6 Retroalimentación de la investigación y del trabajo profesional			
6	COOPERACIÓN CON OTROS ACTORES DE LA SOCIEDAD				
18	Vinculación	18.1 Descripción de mecanismos de vinculación 18.2 Copia de convenios o contratos vigentes 18.3 Estudios o informes relacionados con la vinculación			
19	Financiamiento	19.1 Recursos aplicados 19.2 Ingresos extraordinarios			
7	PLANEACIÓN INSTITUCIONAL (REPORTES GENERADOS POR LA APLICACIÓN)	7.1 Planeación institucional del posgrado 7.2 Autoevaluación del programa de posgrado			
	c) Estructura del plan de estudios				
	d) Número de alumnos matriculados por cohorte generacional.				
	e) Núcleo académico básico (señalar con una breve reseña curricular de los participantes).				
	f) Líneas generadoras y/o aplicación del conocimiento del programa.				
	g) Tutorías				
	h) Productividad académica relevante del programa de posgrado.				
	i) Vinculación con otros sectores de la sociedad.				
	j) Procesos administrativos				
	k) En los casos de programas con participación de varias universidades, la información deberá figurar en la pag. De cada una de ellas				

Figura 8.1: Formato de control de medios de verificación del PNPC.

## 8.1 Claves y nomenclatura de archivos

MCIA	Maestría en Ciencias e Ingeniería Ambientales.
MCC	Maestría en Ciencias de la Computación.
MIE	Maestría en Ingeniería Estructural.
MCIM	Maestría en Ciencias e Ingeniería de Materiales.
MO	Maestría en Optimización.
MIP	Maestría en Ingeniería de Procesos.
DCIAM	Doctorado en Ciencias e Ingeniería (Ambiental, de Materiales).
DIE	Doctorado en Ingeniería Estructural.
DO	Doctorado en Optimización.
DIP	Doctorado en Ingeniería de Procesos.

Tabla 8.1.1 Claves para posgrados.

trimes	Trimestres
tesis	Tesis
docent	Docentes
alumno	Alumnos
posgra	Posgrados
planes	Planes de estudios
usuari	Usuarios

Tabla 8.1.2 Tipos de estatus de las entidades trimestresalumnos, tesis, docentes, posgrados, planesestudios y usuarios.

## 8 Anexo

insact	Inscrito Activo	o	Alumno con carga académica (UEA).
norein	No reinscrito		Alumno que no se inscribió a ninguna UEA, ni tampoco se inscribió "en blanco".
suspen	Suspendido		Alumno sancionado temporalmente por cometer una falta a la Universidad. Lo anterior, con base a un dictamen de órgano colegiado.
bajade	Baja definitiva		Recurso que tiene el alumno para solicitar su baja voluntaria en la Universidad.
egresa	Egresado		Alumno que concluyó los créditos del plan de estudios, cubrió actividades académicas aunque no tengan valor en créditos y solicitó su certificado de estudios de especialización, maestría o doctorado.
bajare	Baja reglamentaria		Es la pérdida de la calidad de alumno. Entre algunas de las causas está el no haber concluido sus estudios en el plazo máximo establecido en el plan de estudios. Así también en algunos planes de estudio se establece que el alumno sólo tiene dos oportunidades para aprobar una UEA.
egreaa	Egresado "a"		Egresado potencial ; es decir, es el alumno activo que está por concluir sus créditos del plan de estudios. Se le pone este estado para no afectar su historial académico, cuando su plan de estudios se encuentra en un proceso de conversión.
bajadi	Baja por dictamen de órgano colegiado		Expulsión definitiva del alumno dictada por órgano colegiado
insbla	Inscrito en blanco		Alumno que hizo el trámite de reinscripción sin "carga" de UEA. Lo anterior, para tener derecho a utilizar los servicios de biblioteca, actividades deportivas y salas de cómputo, etc.
acenua	Aceptado nuevo ingreso		Estado académico temporal que se utiliza para identificar a los aspirantes seleccionados de primer ingreso. Sólo se tiene unos cuantos días ya que después cambia conforme el admitido tiene o no actividad escolar.
credcu	Créditos cubiertos		Alumno que ya concluyó los créditos del plan de estudios y actividades académicas requeridas aunque éstas no tengan valor en créditos. No ha solicitado su certificado de estudios de especialización, maestría o doctorado en la UAM, ni ha obtenido diploma de especialización ni grado académico de maestría o doctorado.
nueinp	Nuevo ingreso no presentado		Aspirante seleccionado de primer ingreso que no hizo ningún trámite de inscripción. Este estado empezó a utilizarse a partir del año de 1996.
abseis	Abandono de más de seis trimestres consecutivos		No aplica para posgrado. Se incluyó en este documento por compatibilidad con el catálogo de licenciatura.
alumdi	Alumno con diploma		Egresado de una especialización que obtuvo el grado académico correspondiente.
alugra	Alumno con		Egresado de una maestría o doctorado que obtuvo el grado académico correspondiente.

	grado	
--	-------	--

Tabla 8.1.3 Claves para estatus de alumnos por trimestre.

activo	Activo.
sabati	En período sabático.

Tabla 8.1.4 Claves para estatus de docentes.

proces	En proceso.
finali	Finalizada.

Tabla 8.1.5 Claves para estatus de tesis.

activo	Activo.
inacti	Inactivo.
proact	En proceso de aprobación.

Tabla 8.1.6 Claves para estatus de posgrados.

activo	Activo.
inacti	Inactivo.
proact	En proceso de aprobación.

Tabla 8.1.7 Claves para estatus de planes de estudios.

activo	Activo.
inacti	Inactivo.
inhabi	Inhabilitado.

Tabla 8.1.8 Claves para estatus de usuarios.

FOTO	Fotografía.
TITULO	Título de licenciatura o grado.
CVU	Currículum Vitae Único de Conacyt.

Tabla 8.1.9 Claves para documentos personales

TESIS	Tesis de maestría o doctorado.
LIBRO	Libros.
ARTICULO	Artículos de investigación.
REPORTE	Reportes técnicos.
MOVILIDAD	Movilidad de alumnos.

Tabla 8.1.10 Claves para documentos asociados al alumno.

PROYECTOS	Proyectos de investigación
-----------	----------------------------

Tabla 8.1.11 Clave para proyectos de investigación dirigidos por docentes.

PLAN	Planes de estudios
------	--------------------

Tabla 8.1.12 Clave para planes de estudios

## 8.2 Nomenclatura de archivos

La Figura 8.2.1 muestra la nomenclatura para el archivo de fotografía de la entidad usuarios.



Figura 8.2.1: Propuesta de nomenclatura para el archivo de fotografía de usuario.

La Figura 8.2.2 muestra la nomenclatura para los archivos correspondientes a las claves: CVU, TITULO, FOTO, PLAN, REPORTE, MOVILIDAD, LIBRO, TESIS, ARTICULO asociadas a la entidad alumnos.

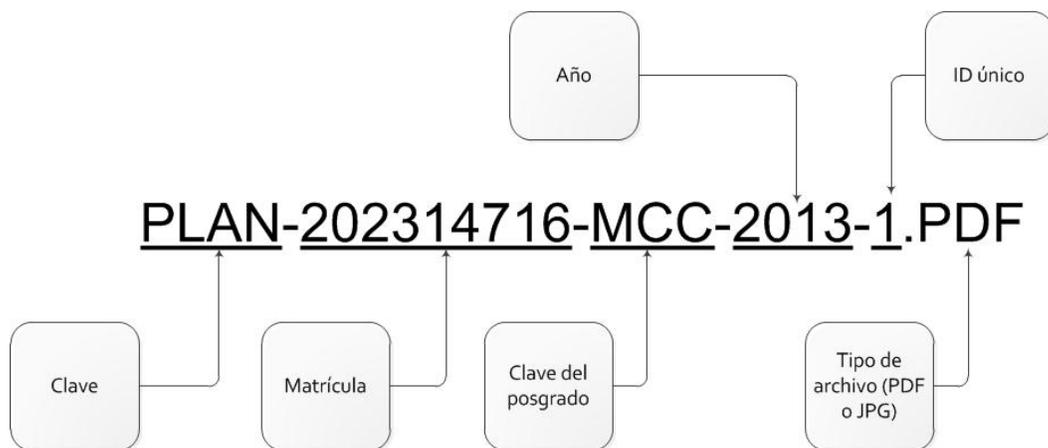


Figura 8.2.2: Propuesta de nomenclatura de archivos asociados a la entidad alumnos.

---

## Glosario

---

<b>Término</b>	<b>Definición e información</b>	<b>Alias</b>
Baja	Representa la acción de inhabilitar. Los usuarios, alumnos, docentes y planes de estudios no se eliminarán, únicamente se inhabilitará su acceso.	Eliminar
Usuario	Actor primario que gestiona la información del sistema.	Administrador tipo A o B
ACL	Lista de control de acceso	Lista

Tabla 8.2.1: Tabla de definiciones del glosario.

---

## Bibliografía

---

- [1] J. A. Morales Orea, “Sistema automatizado de información para las oficinas de la Coordinación Divisional de Docencia y Atención a Alumnos de CBI para el Área de Posgrado,” proyecto terminal, Licenciatura en Computación, División de CBI, Universidad Autónoma Metropolitana Iztapalapa, México, 2002.
- [2] J. Rivera Campuzano, “Sistema de información de la producción académica de los profesores de la División de Ciencias Básicas e Ingeniería,” proyecto terminal, Licenciatura en Computación, División de CBI, Universidad Autónoma Metropolitana Iztapalapa, México, 1992.
- [3] G. P. Hernández Trejo y L. Pérez Peña, “Análisis y diseño de un sistema de información para el Departamento de Economía,” proyecto terminal, Licenciatura en Computación, División de CBI, Universidad Autónoma Metropolitana Iztapalapa, México, 2001.
- [4] Dropbox. (2012, febrero 24). *Recorrido*. [En línea]. Disponible en : <http://www.dropbox.com/tour>
- [5] Owl. (2012, marzo 28). *Inicio*. [En línea]. Disponible en: <http://www.doxbox.ca/>
- [6] OpenMK. (2012, febrero 25). *Inicio*. [En línea]. Disponible en: <http://www.openkm.com/>
- [7] S. R. Schach, “The Object-Oriented Paradigm, UML, and the Unified Process,” en *Introduction to object oriented analysis and design with UML and the unified process*, P. Ducham, Ed. New York: McGraw-Hill, 2004, pp. 47-48.
- [8] P. Kruchten, “The Rational Unified Process” en *The Rational Unified Process an Introduction*, 3a ed. Boston, MA: Addison-Wesley, 2004, pp. 22.
- [9] B. Holmes y D.T. Joyce, “Objects working together” en *Object-oriented programming with java*, J. H. Hauck, Ed., Sudbury, MA: Jones and Bartlett Publishers, Inc., 2001, pp. 582-586.
- [10] P. K. Pachghare, “Authentication” en *Cryptography And Information Security*, A. K. Ghosh, Ed., New Delhi: PHI Learning Private Limited, 2009, pp.172-176.

Universidad Autónoma Metropolitana Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería

Licenciatura en Ingeniería en Computación

Propuesta de proyecto terminal

Gestión del archivo digital de la Coordinación de Posgrados de CBI mediante un sistema de información web

Hernández Báez Antonio                      202314716                      \_\_\_\_\_

Neria Martínez Francisco Javier                      206304789                      \_\_\_\_\_

Trimestre 12 - I

10 de abril de 2012

Tercera versión

Asesora: Dra. María Lizbeth Gallardo López  
Profesor-investigador  
Departamento de Sistemas

\_\_\_\_\_

## Objetivo general

Diseñar e implementar un sistema de información web para la gestión del archivo digital de la Coordinación de Posgrados de la División de Ciencias Básicas e Ingeniería.

## Objetivos específicos

- Diseñar e implementar un módulo que permita la gestión de administradores.
- Diseñar e implementar un módulo que permita la gestión de alumnos.
- Diseñar e implementar un módulo que permita la gestión de docentes.
- Diseñar e implementar un módulo que permita la gestión de planes de estudio de posgrado.
- Diseñar e implementar un módulo que permita obtener informes de acuerdo a los requerimientos establecidos por el PNPC<sup>1</sup> del Consejo Nacional de Ciencia y Tecnología.
- Diseñar e implementar una base de datos que soporte la información correspondiente de los módulos a desarrollar.
- Diseñar una interfaz gráfica de usuario que permita utilizar el sistema web.
- Integrar los módulos y la interfaz gráfica de usuario.

Es importante señalar que cuando hablamos de "gestión", nos referimos a las operaciones: alta, consulta, cambios e inhabilitación de la información concerniente a: administradores, alumnos, docentes y planes de estudio.

## Introducción

Se han realizado esfuerzos por crear sistemas de información que ayuden a gestionar el archivo del posgrado y el proceso de preregistro de los aspirantes. En el periodo de 2008 a 2010 se implantó un sistema, el cual fue desarrollado por personal de la División de Ciencias Sociales y Humanidades de la UAM Azcapotzalco. Este sistema permite actualizar los datos personales de docentes, el contenido del sitio web y el preregistro de aspirantes a los posgrados. De acuerdo a los comentarios expresados por la Coordinación de Posgrados, este sistema carece de manuales de usuario, manuales de instalación y documentación sobre su diseño. El mismo sistema fue desarrollado en el lenguaje ASP<sup>2</sup> e implantado en un equipo con sistema operativo *Windows 2003 Server* y servidor web *IIS*<sup>3</sup>, y no se especificó si se contaban con las licencias originales. El sistema tampoco cuenta con cifrado de contraseñas, ni está soportado sobre una capa segura que proteja la integridad de las transacciones entre el cliente y el servidor. Por lo anterior, este sistema fue retirado en el trimestre 10-P.

En este proyecto terminal se propone crear un nuevo sistema de información web mediante el uso de un marco de desarrollo de software. En su diseño se considerarán aspectos sobre seguridad: cifrado de contraseñas; usabilidad: no sobrecargar las páginas de presentación, emplear 3 niveles en la navegabilidad, y respetar el diseño del sitio web de la Unidad; capacidad de mantenimiento: documentar requerimientos, diseño e implementación. Se tendrá particular interés en el uso de software libre para su desarrollo.

## Justificación

Los Coordinadores de los posgrados de la División de Ciencias Básicas e Ingeniería tienen la necesidad de llevar el control de la información de los planes de estudio, alumnos y docentes. Esta información se encuentra distribuida en documentación impresa y en archivos en formato electrónico; a los cuales nos

1 PNPC: El Programa Nacional de Posgrados de Calidad de CONACyT solicita a las instituciones registradas en él un CD con un reporte que contenga datos académicos de alumnos, docentes y planes de posgrados, entre otros.

2 ASP (*Active Server Pages*): Tecnología de *Microsoft* para desarrollar páginas web dinámicas.

3 IIS (*Internet Information Services*): Servidor web desarrollado por *Microsoft* para el sistema operativo *Windows*.

referiremos como archivo del posgrado. Actualmente estos archivos se encuentran almacenados en diferentes computadoras lo que dificulta su rápida localización y administración, además, no se lleva un control de las personas que acceden a esta información.

En este proyecto terminal proponemos crear un nuevo sistema de gestión del archivo del posgrado que permita incluir una base de datos para almacenar los datos de los usuarios, y una base de documentos para toda la información: tesis, actas, solicitudes de becas, *kárdex*, y artículos de investigación, entre otros. Para garantizar la seguridad de las transacciones entre el cliente y el servidor, el sistema se implementará sobre una capa de conexión segura, además se usarán contraseñas con cifrado. Se podrá tener acceso al sistema a través de una aplicación cliente (navegador) en diferentes sistemas operativos. Se diseñará una interfaz gráfica de usuario intuitiva, que permita su fácil uso. Además, se podrán generar reportes para el PNPC de CONACyT.

El sistema ofrecerá los siguientes beneficios: se automatizarán los procesos de búsqueda de información, generación de reportes y trámites, se tendrá la información de los alumnos y docentes centralizada, el acceso y transferencias de información se efectuarán de forma segura, contará con una interfaz intuitiva y fácil de utilizar. Además, permitirá una administración adecuada de la información de los alumnos y docentes del posgrado. Este sistema contará con la documentación del análisis, diseño e implementación, lo que permitirá que el sistema sea escalable y adaptable a nuevos requerimientos. También, se contará con un manual de usuario e instalación. El sistema será implementado en su totalidad en software libre.

## **Antecedentes**

Existen proyectos dentro y fuera de la Universidad que tienen como propósito la gestión de información académica de alumnos y docentes, y la gestión de archivos.

### **Referencias internas**

Sistema automatizado de información para las oficinas de la Coordinación Divisional de Docencia y Atención a Alumnos de CBI para el Área de Posgrado. En este proyecto se diseñó un sistema para gestionar la información académica de los alumnos de posgrado de la División de Ciencias Básicas e Ingeniería de la UAM Iztapalapa. Se diseñó el sistema siguiendo el paradigma de programación orientada a objetos, auxiliándose con el lenguaje unificado de modelado<sup>4</sup>. Este proyecto sugiere las tecnologías: *Java Applet*<sup>5</sup>, lenguaje de marcado de hipertexto con *CGI*<sup>6</sup> y conexión a una base de datos mediante *DBSP*<sup>7</sup>, para implementar el sistema en web [1].

En nuestro proyecto se diseñarán e implementarán los módulos descritos en los objetivos particulares, mediante el uso del proceso unificado<sup>8</sup> y el lenguaje unificado de modelado. Su implementación se realizará en el lenguaje *PHP*<sup>9</sup> orientado a objetos. La base de datos se desarrollará en *MySQL*<sup>10</sup>.

---

4 Lenguaje unificado de modelado: Lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

5 *Java Applet*: Componente de una aplicación que se ejecuta en el contexto de otro programa escrito en lenguaje *Java*.

6 *CGI (Common Gateway Interface)*: Tecnología que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web.

7 *DBSP (Data Base Scripting Pages)*: Tecnología que se ejecuta en el servidor diseñada para acelerar el proceso de construcción de sitios *B2B (Business to Business)*, *B2C (Business to Client)*, cualquier sistema *e-business* o de información que requiera la explotación de datos a través de Internet.

8 Proceso unificado: Marco de desarrollo de software que se caracteriza por estar dirigido por casos de uso, centrado en la arquitectura y por ser iterativo e incremental.

9 *PHP (PHP Hypertext Pre-processor)*: Es un lenguaje de programación interpretado diseñado para la creación de páginas web dinámicas.

10 *MySQL*: Sistema de gestión de bases de datos relacional, multihilo y multiusuario.

Sistema de información de la producción académica de los profesores de la División de Ciencias Básicas e Ingeniería. Este proyecto terminal gestiona la información académica, estudios realizados, especialidades, artículos, libros y congresos de los profesores adscritos a la División de CBI de la UAM Iztapalapa. Este proyecto fue implementado en *Clipper*<sup>11</sup> con una base de datos elaborada en *DbaseIII+*<sup>12</sup>. El proyecto fue diseñado para instalarse en una o varias computadoras pero sin la posibilidad de sincronizar la información en una base de datos única [2].

Análisis y diseño de un sistema de información para el Departamento de Economía. Este proyecto terminal realiza la gestión de información académica de profesores adscritos al Departamento de Economía de la UAM Iztapalapa. La gestión incluye publicaciones, proyectos de investigación, cursos impartidos, inventario de equipo de cómputo y convenios con otras instituciones. El sistema fue diseñado siguiendo el paradigma orientado a objetos, auxiliándose del lenguaje unificado de modelado. Las interfaces gráficas de la aplicación fueron desarrolladas en *Visual Basic*<sup>13</sup> y únicamente propone el modelo relacional de la base de datos (no realizó su implementación) [3].

En los dos proyectos anteriores se considera únicamente la gestión de información de profesores. Nuestro proyecto considera la gestión de información de profesores, alumnos, planes de estudio de posgrado y reportes. Además, se usará una base de datos relacional y centralizada en la cual se almacenará la información.

### **Referencias externas**

*Dropbox*. Es un servicio de alojamiento de archivos basado en web. Utiliza el modelo de red *cloud storage*<sup>14</sup> para que sus usuarios almacenen y compartan archivos, y/o directorios, a través de Internet. El servicio es gratuito hasta los 2GB de almacenamiento; si se requiere de mayor capacidad debe pagarse una cuota mensual. Los usuarios registrados en el servicio pueden gestionar sus archivos desde el sitio web de la compañía; o bien, descargar una aplicación cliente en su computadora local. Está disponible para distintos sistemas operativos (*Windows*, *MacOS*, *Linux* y teléfonos móviles) [4].

*Owl*. Software para crear un repositorio de documentos basado en web que permite a múltiples usuarios guardar documentos y compartirlos con otros usuarios inscritos a *Owl*. Su licencia es de tipo *GNU/GPL*<sup>15</sup> [5].

*OpenKM*. Software para gestión documental orientado a la pequeña y mediana empresa. Posee una interfaz de usuario web para realizar operaciones como: compartir, establecer la seguridad, auditar y localizar documentos. Este software se encuentra disponible para los sistemas operativos *Windows*, *Linux*, *MacOS* y *Unix*. El software es de código abierto<sup>16</sup> y se distribuye mediante dos licencias: de paga y licencia *GNU/GPL* [6].

Nuestro proyecto no será un servicio para alojar y compartir archivos en red, como los tres mencionados anteriormente. Nuestro sistema se centrará en la gestión de un archivo digital, por medio un sistema web. La información de los estudiantes, profesores y personal administrativo serán alojados en una base de datos y los documentos como: tesis, actas, solicitudes de becas, *kárdex*, y artículos de

---

11 *Clipper*: Lenguaje de programación procedural e imperativo creado en 1985 por *Nantucket Corporation*.

12 *DbaseIII+*: Sistema de gestión de bases de datos desarrollado en 1960.

13 *Visual Basic*: Lenguaje de programación dirigido por eventos, desarrollado por el alemán Alan Cooper para Microsoft.

14 *Cloud storage*: Modelo de red de almacenamiento en línea donde los datos se almacenan en repositorios virtualizados que son generalmente organizados por terceros.

15 *GNU/GPL (GNU General Public License)*: Licencia creada por la Fundación de Software Libre orientada a proteger la libre distribución, modificación y uso del software.

16 Código abierto: Término con el que se conoce al software distribuido y desarrollado libremente.

investigación, entre otros serán almacenados en forma de archivos. Esta información podrá ser accedida únicamente por los roles que tengan derecho sobre ella. Por lo tanto, no se considera la posibilidad de compartir la información, tal y como lo hacen los sistemas tipo *Dropbox*, *Owl* y *OpenKM*.

### Descripción técnica

Este sistema de información automatizará la gestión del archivo digital de información de los docentes y alumnos de la Coordinación de Posgrados de CBI. El usuario deberá contar con un equipo de cómputo o dispositivo con acceso a Internet y un navegador web (*Mozilla Firefox*, *Internet Explorer*, *Google Chrome*, etc.) con el cuál podrá interactuar con el sistema.

La Figura 1 representa los casos de uso generales a partir de los cuales se desarrollarán los módulos del sistema.

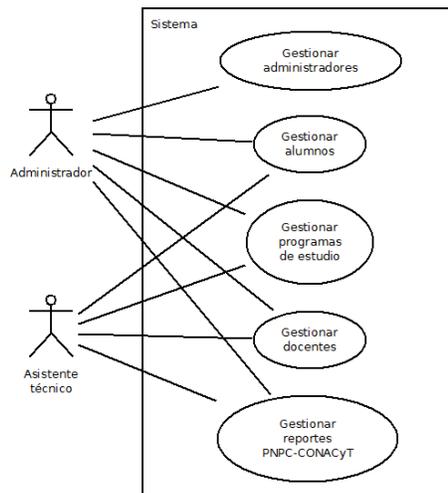


Figura 1: Casos de uso generales del sistema

Los módulos que formarán el sistema son:

#### **Módulo de gestión de administradores**

Permitirá crear, eliminar, modificar e inhabilitar dos tipos de usuario (administradores y asistentes). Los usuarios administradores tendrán acceso total a cualquiera de los módulos del sistema, mientras que los asistentes sólo podrán gestionar la información de los módulos de gestión de alumnos, docentes, planes de estudio del posgrado e informes PNPC de CONACyT.

#### **Módulo de gestión de alumnos**

El caso de uso para alumnos se subdivide en siete casos de uso (Figura 2) que incluyen el manipular: datos personales, movilidad, antecedentes académicos, trayectoria académica, tutorías, tesis y artículos de investigación. En cada uno de estos casos de uso se consideran las operaciones: altas, bajas, cambios e inhabilitación de la información de los alumnos.

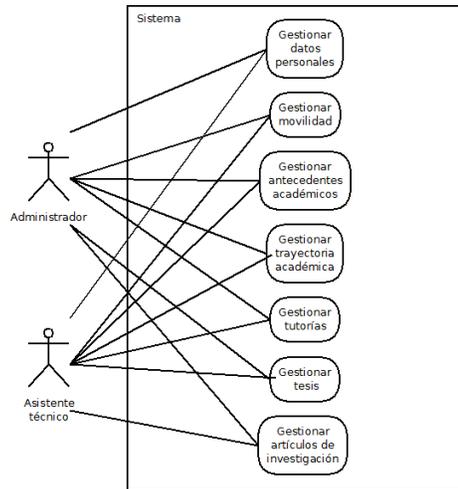


Figura 2: Subcasos de uso para el caso de uso gestionar alumnos

### **Módulo de gestión de planes de estudio**

El caso de uso para los planes de estudio se subdivide en diez casos de uso que incluyen manipular: plan de estudios vigente, justificación del programa, objetivos y metas, perfil de ingreso, perfil de egreso, congruencia del plan de estudios, mapa curricular, actualización del plan de estudios, opciones de graduación y conocimiento de un idioma extranjero. Para cada caso de uso se considera las operaciones de altas, bajas, cambios e inhabilitación.

### **Módulo de gestión de docentes**

El caso de uso para docentes se subdivide en cinco casos de uso (Figura 3) que incluyen manipular: datos personales, antecedentes académicos, núcleo básico, artículos de investigación y proyectos de investigación. Para cada caso de uso se consideran las operaciones de altas, bajas, cambios e inhabilitación de la información de los docentes.

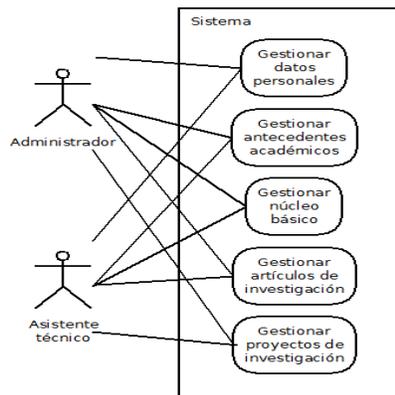


Figura 3: Subcasos de uso para el caso de uso gestionar docentes

### **Módulo gestión de informes de acuerdo a los requerimientos establecidos por PNPC de CONACyT**

Permitirá seleccionar información de la base de datos, implementada para el sistema, de acuerdo a los requerimientos del PNPC de CONACyT, para posteriormente generar una imagen ISO<sup>17</sup> que podrá ser descargada por el usuario solicitante.

### **Interfaz gráfica de usuario del sistema web**

Proporcionará una interfaz gráfica para que el usuario pueda interactuar con el sistema. Con ayuda de

<sup>17</sup> Imagen ISO: Es un archivo donde se almacena una copia o imagen exacta de un sistema de archivos.

esta interfaz podrá consultar la información de los alumnos, docentes y planes de estudio, e identificarse para acceder al sistema (mediante un nombre de usuario y contraseña).

Otra vista del sistema se muestra en el diagrama de comunicación entre bloques. En éste se observa la interacción del usuario con el sistema para realizar operaciones (Figura 4). Los puntos relevantes que se pueden observar son:

- El usuario ingresa al sitio web.
- El usuario se identifica y accede al sistema.
- El usuario selecciona el módulo a gestionar.
- El usuario realiza la transacción que ha seleccionado.

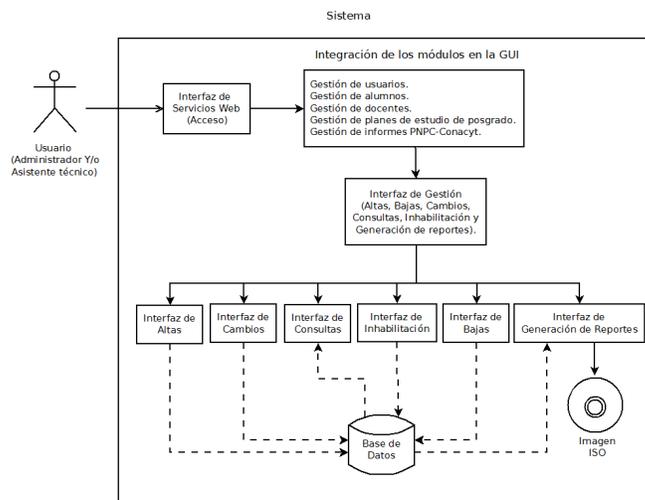


Figura 4: Diagrama de comunicación entre bloques.

La aplicación cliente y el sistema de información web interactuarán como se muestra en la Figura 5.

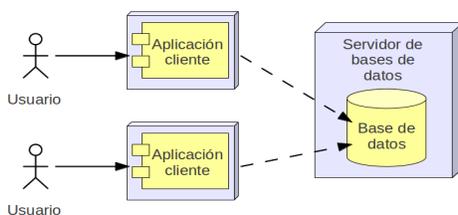


Figura 5: Interacción entre los componentes necesarios para la gestión

## Especificación técnica

### Valores de entrada y salida

Las variables de entrada serán cadenas de caracteres en modo texto, imágenes (*JPEG*<sup>18</sup> y *PNG*<sup>19</sup>) y archivos electrónicos en formato *PDF*<sup>20</sup> para los módulos de gestión de alumnos, programas de estudio, docentes y reportes.

18 *JPEG (Joint Photographic Experts Group)*: Algoritmo de compresión con pérdida para reducir el tamaño de los archivos de imágenes.

19 *PNG (Portable Network Graphics)*: Formato gráfico basado en un algoritmo de compresión sin pérdida para *bitmaps* no sujeto a patentes.

20 *PDF (Portable Document Format)*: Formato de almacenamiento de documentos.

Las variables de salida serán cadenas de caracteres en modo texto y documentos en formatos *PDF* para los módulos de gestión de alumnos, programas de estudio, docentes y reportes; además, este último generará una imagen *ISO*.

### **Protocolos**

*HTTP* es un protocolo de transferencia de hipertexto, que será usado para cada una de las transacciones que ocurran en el sistema, a saber: el inicio de sesión, las modificaciones y el fin de una sesión.

*TCP/IP* es un modelo de red que especifica cómo un equipo se conectará a una red. El nombre del modelo se origina por los dos principales protocolos de la pila llamados: a) *Transmission Control Protocol (TCP)* orientado a conexión, que garantiza que los datos se entreguen sin errores a su destino, e b) *Internet Protocol (IP)* no orientado a conexión, que permite el envío de datos en forma de paquetes a través de la red y proporciona una dirección *IP* para la identificación de los dispositivos.

### **Entorno de desarrollo**

Para diseñar el sistema web se utilizará la metodología orientada a objetos llamada proceso unificado y el lenguaje unificado de modelado para representar los sistemas de información [7]. Una de nuestras inquietudes es reforzar nuestros conocimientos en el paradigma orientado a objetos, ya que en el transcurso de nuestra formación profesional el tipo de paradigma que más hemos desarrollado y trabajado ha sido el paradigma estructurado.

*PHP* orientado a objetos será la plataforma de desarrollo, dado que posee una amplia documentación, es un lenguaje orientado a objetos y se encuentra disponible en la distribución *Debian*.

*PHP* será usado para implementar los módulos y la interfaz gráfica de usuario del sistema. Para implementar la interfaz gráfica de usuario, se utilizarán los lenguajes *JavaScript* y *CSS*<sup>21</sup>. Se utilizará *MySQL* para implementar la base de datos. La base de datos incluirá niveles de aislamiento para evitar que más de un usuario realice transacciones simultáneas sobre la misma información. El sistema usará el servidor web *HTTP Apache*<sup>22</sup> sobre una capa de seguridad *SSL*<sup>23</sup> en sus transacciones Cliente-Servidor, así como cifrado de contraseñas en *MD5*<sup>24</sup>. Lo anterior se realizará sobre el sistema operativo *Debian*.

El sistema se instalará en un servidor *HP ProLiant G6 Series* con sistema operativo *Debian v6*, *HTTP Apache server v2.2.16-6*, *MySQL server v5.1.49-3*, *PHP server v5.3* y *OpenSSL*<sup>25</sup> *v0.9.8*.

### **Pruebas**

Las funcionalidad de cada uno de los módulos a implementar será comprobada mediante pruebas de unidad. Los módulos serán probados localmente de manera independiente para asegurarse de que cada uno cumple con la funcionalidad esperada (se verificará que las entradas y salidas de las funciones proporcionadas por un módulo sean procesadas correctamente). Finalizadas las pruebas de unidad, se realizarán pruebas de integración. En estas pruebas se comprobará la correcta interacción entre los módulos implementados.

### **Características mínimas**

El proyecto se dará por terminado una vez que:

21 *CSS (Cascading Style Sheets)*: lenguaje usado para definir la presentación de un documento estructurado.

22 *HTTP Apache*: Servidor web desarrollado por la *Apache Software Foundation*.

23 *SSL (Secure Sockets Layer)*: Protocolo criptográfico que proporciona comunicaciones seguras en una red.

24 *MD5 (Message-Digest Algorithm 5)*: Algoritmo de reducción criptográfico de 128 bits.

25 *OpenSSL*: Implementación en código abierto de los protocolos *SSL* y *TLS*.

1. El sistema permita dar de alta un o más usuarios, tipo administrador o asistente técnico;
2. El sistema realice las operaciones de altas, bajas, cambios, consultas e inhabilitaciones en los distintos módulos del sistema;
3. En el caso particular del módulo de gestión de reportes PNPC de CONACyT, el sistema permita generar un reporte en imagen ISO.

### **Possible continuación del proyecto**

Una posible continuación consistiría en diseñar módulos para gestionar información sobre la infraestructura, servicios y seguimiento de egresados del posgrado. Otra posibilidad consistiría en implementar la gestión de recursos en una arquitectura distribuida para manejar grandes cantidades de información.

### **Entregables**

Los productos de trabajo que se entregarán grabados en un CD son:

- Reporte de proyecto terminal en formato *PDF*.
- Documentación del marco de desarrollo del sistema de información.
- Código fuente del sistema de información web implementado.
- Manuales de instalación y usuario.

### **Calendario de trabajo**

El calendario de trabajo está definido en términos de los 6 casos de uso generales referidos en la "Descripción técnica".

Calendario de Trabajo del alumno Hernández Báez Antonio para la UEA Proyecto Terminal de Ingeniería en Computación I del trimestre 12-P con 9 horas por semana (Tabla 1).

HERNÁNDEZ BÁEZ ANTONIO											
PROYECTO TERMINAL 1											
ACTIVIDAD	SEMANA										
	1	2	3	4	5	6	7	8	9	10	11
Modelo de negocio	■	■	■	■	■	■					
Análisis de requerimientos	■	■	■	■	■	■	■	■			
Diseño del módulo usuarios		■	■	■	■	■					
Implementación del módulo usuarios				■	■	■	■	■			
Pruebas del módulo usuarios						■	■	■			
Diseño del módulo docentes		■	■	■	■	■	■	■	■		
Implementación del módulo docentes				■	■	■	■	■	■	■	
Pruebas del módulo docentes						■	■	■	■	■	
Diseño de la base de datos		■	■	■							
Implementación de la base de datos			■	■	■						
Pruebas de la base de datos				■	■	■	■				
Gestión de versiones del software							■	■	■	■	
Reporte parcial	■	■	■	■	■	■	■	■	■	■	■

Tabla 1: Calendario de actividades para el Proyecto Terminal I asignadas al alumno Hernández Báez Antonio

Para la UEA Proyecto Terminal de Ingeniería en Computación II del trimestre 12-O con 18 horas por semana (Tabla 2).

HERNÁNDEZ BÁEZ ANTONIO											
PROYECTO TERMINAL 2											
ACTIVIDAD	SEMANA										
	1	2	3	4	5	6	7	8	9	10	11
Diseño de la interfaz gráfica	■	■	■	■	■						
Implementación de la interfaz gráfica			■	■	■	■	■				
Pruebas de la interfaz gráfica					■	■	■				
Integración de los módulos y la interfaz gráfica						■	■	■	■		
Gestión de versiones del software					■	■	■	■	■	■	
Reporte final	■	■	■	■	■	■	■	■	■	■	■

Tabla 2: Calendario de actividades para el Proyecto Terminal II asignadas al alumno Hernández Báez Antonio

Calendario de Trabajo del alumno Neria Martínez Francisco Javier para la UEA Proyecto Terminal de Ingeniería en Computación I del trimestre 12-P con 9 horas por semana (Tabla 3).

NERIA MARTÍNEZ FRANCISCO JAVIER											
PROYECTO TERMINAL 1											
ACTIVIDAD	SEMANA										
	1	2	3	4	5	6	7	8	9	10	11
Modelo de negocio	■	■	■	■	■	■					
Análisis de requerimientos	■	■	■	■	■	■					
Diseño del módulo alumnos		■	■	■	■	■	■	■			
Implementación del módulo alumnos				■	■	■	■	■	■	■	■
Pruebas del módulo alumnos						■	■	■	■	■	■
Diseño del módulo gestión de informes PNP		■	■	■	■	■					
Implementación del módulo gestión de informes PNP				■	■	■	■	■			
Pruebas del módulo gestión de informes PNP						■	■	■			
Diseño de la base de datos		■	■	■	■						
Implementación de la base de datos			■	■	■						
Pruebas de la base de datos				■	■						
Gestión de versiones del software							■	■	■	■	■
Reporte parcial	■	■	■	■	■	■	■	■	■	■	■

Tabla 3: Calendario de actividades para el Proyecto Terminal I asignadas al alumno Neria Martínez Francisco Javier.

Para la UEA Proyecto Terminal de Ingeniería en Computación II del trimestre 12-O con 18 horas por semana (Tabla 4).

NERIA MARTÍNEZ FRANCISCO JAVIER											
PROYECTO TERMINAL 2											
ACTIVIDAD	SEMANA										
	1	2	3	4	5	6	7	8	9	10	11
Diseño del módulo plan de estudios											
Implementación del módulo plan de estudios											
pruebas del módulo plan de estudios											
Integración de los módulos y la interfaz gráfica											
Gestión de versiones del software											
Reporte final											

Tabla 4: Calendario de actividades para el Proyecto Terminal II asignadas al alumno Neria Martínez Francisco Javier.

## Recursos

Se cuenta con todos los recursos de software y hardware necesarios para realizar el proyecto, los cuales se mencionan a continuación.

### Disponibles

Hardware:

- Equipo 1: *Laptop Compaq-Presario CQ43*, procesador *Intel® Celeron® Dual-Core T3500 2.10 GHz*, 2 GB en RAM, disco duro de 320 GB
- Equipo 2: *Netbook Benq Joybook Lite U105*, procesador *Intel® Atom™ N450 1.66 Ghz*, 2 GB en RAM, disco duro de 240 GB
- *Gateway 3Com OfficeConnect 3C8657*
- Servidor *HP ProLiant G6 Series*, procesador *Xeon 3.2 GHz*, 2 GB en RAM, disco duro de 300 GB (El asesor de proyecto proporcionará este recurso).

Software:

- Los equipos portátiles cuentan con el siguiente software:
  - Sistema operativo *Microsoft® Windows® 7 Starter* (equipo 1) y *Microsoft® Windows® 7 Professional* (equipo 2). En ambos casos se cuenta con las licencias originales.
  - Sistema operativo *Debian* versión 6 con *kernel 2.6.32-5-486*, licencia *GNU/GPL* versión 3.
  - *HTTP Apache server* versión 2.2.16-6, licencia *Apache License 2.0*<sup>26</sup>.
  - *MySQL server* versión 5.1.49-3, licencia *GNU/GPL* versión 2.
  - *PHP server* versión 5.3.3-7+squeeze1, licencia *PHP License* versión 3.01<sup>27</sup>.
  - *OpenSSL* versión 0.9.8o4squeeze7, licencia *Apache License 1.0*.
- El equipo *HP ProLiant G6 Series* tiene:
  - Sistema operativo *OpenSolaris 2009.06*.
  - *VirtualBox 4.1.8* para *hosts Solaris*, licencia *GNU/GPL* versión 2.

## Bibliografía

- [1] J. A. Morales Orea, “Sistema automatizado de información para las oficinas de la Coordinación Divisional de Docencia y Atención a Alumnos de CBI para el Área de Posgrado,” proyecto

<sup>26</sup> *Apache License 2.0*: Licencia de software libre creada por la *Apache Software Foundation* compatible con la *GNU General Public License*.

<sup>27</sup> *PHP License*: Licencia de código abierto creada por el *PHP Group*.

terminal, Licenciatura en Computación, División de CBI, Universidad Autónoma Metropolitana Iztapalapa, México, 2002.

- [2] J. Rivera Campuzano, “Sistema de información de la producción académica de los profesores de la División de Ciencias Básicas e Ingeniería,” proyecto terminal, Licenciatura en Computación, División de CBI, Universidad Autónoma Metropolitana Iztapalapa, México, 1992.
- [3] G. P. Hernández Trejo y L. Pérez Peña, “Análisis y diseño de un sistema de información para el Departamento de Economía,” proyecto terminal, Licenciatura en Computación, División de CBI, Universidad Autónoma Metropolitana Iztapalapa, México, 2001.
- [4] Dropbox. (2012, febrero 24). *Recorrido*. [En línea]. Disponible en : <http://www.dropbox.com/tour>
- [5] Owl. (2012, marzo 28). *Inicio*. [En línea]. Disponible en: <http://www.doxbox.ca/>
- [6] OpenMK. (2012, febrero 25). *Inicio*. [En línea]. Disponible en: <http://www.openkm.com/>
- [7] S. R. Schach, “The Object-Oriented Paradigm, UML, and the Unified Process,” en *Introduction to object oriented analysis and design with UML and the unified process*, P. Ducham, Ed. New York: McGraw-Hill, 2004, pp. 47-48.

Universidad Autónoma Metropolitana Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería

Licenciatura en Ingeniería en Computación

Diseño del sistema

Gestión del archivo digital de la Coordinación de Posgrados de CBI mediante un sistema de información web

Hernández Báez Antonio

202314716

Neria Martínez Francisco Javier

206304789

Trimestre 13 - I

Abril de 2013

Asesora: Dra. María Lizbeth Gallardo López

Profesor-investigador

Departamento de Sistemas

---

# Índice de contenido

---

<b>1 Visión.....</b>	<b>1</b>
1.1 Introducción.....	1
1.2 Orientación.....	1
1.2.1 Oportunidad del negocio.....	1
1.2.2 Enunciado del problema.....	2
1.2.3 Enunciado de la posición.....	2
1.3 Personal involucrado.....	2
1.3.1 Personal por parte del cliente.....	2
1.3.2 Personal por parte del equipo de desarrollo.....	2
1.3.3 Objetivos de alto nivel y soluciones actuales por parte del cliente.....	3
1.3.4 Objetivos a nivel de usuario.....	3
1.4 Visión general del sistema computacional.....	3
1.4.1 Perspectiva del sistema.....	3
1.4.2 Resumen de los beneficios del sistema.....	4
1.5 Resumen de las características del sistema.....	4
<b>2 Especificación complementaria.....</b>	<b>5</b>
2.1 Introducción.....	5
2.2 Funcionalidad.....	5
2.2.1 Registro de usuarios.....	5
2.2.2 Disponibilidad de información.....	5
2.2.3 Reportes.....	5
2.2.4 Seguridad.....	5
2.3 Facilidad de uso.....	6
2.4 Fiabilidad.....	8
2.5 Rendimiento.....	8
2.6 Soporte.....	8
2.7 Implementación.....	8
2.8 Reglas de dominio (negocio).....	9
2.9 Información en dominios de interés.....	9
<b>3 Lógica de negocio.....</b>	<b>10</b>
3.1 Lógica de negocio actual.....	10
3.2 Lógica de negocio propuesta.....	11
<b>4 Glosario.....</b>	<b>12</b>
<b>5 Diagramas de casos de uso.....</b>	<b>13</b>
5.1 Casos de uso generales.....	13
5.2 Caso de uso gestionar autenticación de usuarios.....	14
5.3 Caso de uso gestionar usuarios.....	14
5.4 Caso de uso gestionar alumnos.....	15
5.5 Caso de uso gestionar planes de estudios.....	15
5.6 Caso de uso gestionar docentes.....	16
<b>6 Escenarios.....</b>	<b>17</b>
6.1 GAU: Gestionar autenticación de usuarios.....	18
6.1.1 GAU1: Iniciar sesión de usuario.....	18
6.1.2 GAU2: Finalizar sesión de usuario.....	19
6.2 GU: Gestionar usuarios.....	20
6.2.1 GU1: Realizar alta de usuario.....	20

6.2.2 GU2: Realizar baja de usuario.....	21
6.2.3 GU3: Realizar cambios de información de usuario.....	22
6.2.4 GU4: Consultar información de usuario.....	24
6.3 GA: Gestionar alumnos.....	25
6.3.1 GA1: Realizar alta de alumno.....	25
6.3.2 GA2: Realizar baja de alumno.....	26
6.3.3 GA3: Realizar cambios de información de alumno.....	27
6.3.4 GA4: Consultar información de alumno.....	28
6.4 GPE: Gestionar planes de estudio.....	29
6.4.1 GPE1: Realizar alta de un plan de estudios.....	29
6.4.2 GPE2: Realizar baja de un plan de estudios.....	30
6.4.3 GPE3: Realizar cambio de información de un plan de estudios.....	31
6.4.4 GPE4: Consultar información de un plan de estudios.....	33
6.5 GD: Gestionar docentes.....	34
6.5.1 GD1: Realizar alta de docente.....	34
6.5.2 GD2: Realizar baja de docente.....	35
6.5.3 GD3: Realizar cambios en información de docente.....	36
6.5.4 GD4: Consultar información de docente.....	37
<b>7 Tarjetas CRC.....</b>	<b>39</b>
<b>8 Modelo de dominio.....</b>	<b>45</b>
8.1 Modelo de dominio general del SIP-CBI.....	45
8.2 Modelo de dominio para la entidad Docentes.....	46
8.3 Modelo de dominio para la entidad Alumnos.....	46
8.4 Modelo de dominio para la entidad Posgrados.....	47
<b>9 Diagramas de secuencias.....</b>	<b>48</b>
<b>10 Diagrama de clases.....</b>	<b>51</b>
<b>11 Esquema de la base de datos.....</b>	<b>54</b>
<b>12 Arquitectura.....</b>	<b>59</b>
<b>13 Anexos.....</b>	<b>60</b>
13.1 Formato de control de medios de verificación del PNPC.....	60
13.2 Claves y nomenclatura de archivos.....	61
13.3 Nomenclatura de archivos.....	63
<b>14 Bibliografía.....</b>	<b>65</b>

---

## Índice de figuras

---

Figura 1.4.1.1: Diagrama de contexto del sistema.....	4
Figura 2.3.1: Prototipo de interfaz gráfica.....	6
Figura 2.3.2: Propuesta de navegabilidad para el proceso de búsqueda y cambio de información.....	7
Figura 2.3.3: Propuesta de navegabilidad para el proceso de alta.....	7
Figura 5.1.1: Casos de uso generales del sistema.....	13
Figura 5.2.1: Caso de uso gestionar autenticación de usuarios.....	14
Figura 5.3.1: Caso de uso gestionar usuarios.....	14
Figura 5.4.1: Caso de uso gestionar alumnos.....	15
Figura 5.5.1: Caso de uso gestionar planes de estudio.....	15
Figura 5.6.1: Caso de uso gestionar docentes.....	16
Figura 7.1: Tarjeta CRC Autenticación.....	39
Figura 7.2: Tarjeta CRC Usuarios.....	40
Figura 7.3: Tarjeta CRC Trimestres.....	40
Figura 7.4: Tarjeta CRC Alumnos.....	40
Figura 7.5: Tarjeta CRC Datos personales.....	41
Figura 7.6: Tarjeta CRC Antecedentes académicos.....	41
Figura 7.7: Tarjeta CRC Tesis.....	41
Figura 7.8: Tarjeta CRC Reportes técnicos.....	42
Figura 7.9: Tarjeta CRC Libros.....	42
Figura 7.10: Tarjeta CRC Artículos.....	42
Figura 7.11: Tarjeta CRC Posgrados.....	43
Figura 7.12: Tarjeta CRC Movilidad de alumnos.....	43
Figura 7.13: Tarjeta CRC Planes de estudios.....	43
Figura 7.14: Tarjeta CRC Docentes.....	43
Figura 7.15: Tarjeta CRC Proyectos de investigación.....	44
Figura 8.1.1: Entidades primarias del modelo de dominio.....	45
Figura 8.2.1: Modelo de dominio para las entidades secundarias de la entidad docentes.....	46
Figura 8.3.1: Modelo de dominio para las entidades secundarias de la entidad alumnos.....	46
Figura 8.4.1: Modelo de dominio para las entidades secundarias de la entidad posgrados.....	47
Figura 9.1: Diagrama de secuencias general del SIP-CBI.....	48
Figura 9.2: Diagrama de secuencias para la operación de alta.....	49
Figura 9.3: Diagrama de secuencias para la operación de baja.....	49
Figura 9.4: Diagrama de secuencias para la operación de cambios.....	50
Figura 9.5: Diagrama de secuencias para la operación de consultas.....	50
Figura 10.1: Diagrama de clases del SIP-CBI.....	53
Figura 11.1: Modelo de base de datos del SIP-CBI.....	58
Figura 12.1: Funcionamiento del patrón MVC.....	59
Figura 13.1.1: Formato de control de medios de verificación PNPC.....	60
Figura 13.3.1: Propuesta de nomenclatura para el archivo de fotografía de usuario.....	63
Figura 13.3.2: Propuesta de nomenclatura de archivos asociados a la entidad alumnos.....	64

---

## Índice de tablas

---

Tabla 1.1: Historial de revisiones del documento de visión.....	1
Tabla 1.3.3.1: Objetivos de alto nivel y soluciones actuales por parte del cliente.....	3
Tabla 1.4.2.1: Beneficios del sistema.....	4
Tabla 2.1: Historial de revisiones del documento de especificación complementaria.....	5
Tabla 2.8.1: Reglas de dominio.....	9
Tabla 3.1: Historial de revisiones del documento de lógica de negocio.....	10
Tabla 4.1: Historial de revisiones del documento glosario.....	12
Tabla 4.2: Tabla de definiciones del glosario.....	12
Tabla 5.1: Historial de revisiones del documento diagramas de casos de uso.....	13
Tabla 6.1: Historial de revisiones del documento de escenarios.....	17
Tabla 7.1: Historial de revisiones del documento tarjetas CRC.....	39
Tabla 8.1: Historial de revisiones del documento modelo de dominio.....	45
Tabla 9.1: Historial de revisiones del documento diagramas de secuencias.....	48
Tabla 10.1: Historial de revisiones del documento diagramas de clases de diseño.....	51
Tabla 11.1: Historial de revisiones del documento del esquema de la base de datos.....	54
Tabla 12.1: Historial de revisiones del documento del arquitectura.....	59
Tabla 13.2.1 Claves para posgrados.....	61
Tabla 13.2.2 Tipos de estatus de las entidades trimestresalumnos, tesis, docentes, posgrados, planes de estudios y usuarios.....	61
Tabla 13.2.3 Claves para estatus de alumnos por trimestre.....	62
Tabla 13.2.4 Claves para estatus de docentes.....	62
Tabla 13.2.5 Claves para estatus de tesis.....	62
Tabla 13.2.6 Claves para estatus de posgrados.....	62
Tabla 13.2.7 Claves para estatus de planes de estudios.....	62
Tabla 13.2.8 Claves para estatus de usuarios.....	63
Tabla 13.2.9 Claves para documentos personales.....	63
Tabla 13.2.10 Claves para documentos asociados al alumno.....	63
Tabla 13.2.11 Clave para proyectos de investigación dirigidos por docentes.....	63
Tabla 13.2.12 Clave para planes de estudios.....	63

---

# 1 Visión

Versión	Fecha	Descripción	Autores
borrador	09/V/12	Primer borrador.	Hernández Báez Antonio Neria Martínez Francisco Javier
1	20/VI/12	Definición de los objetivos de alto nivel.	Hernández Báez Antonio Neria Martínez Francisco Javier
2	16/VII/12	Resumen de las características del sistema.	Hernández Báez Antonio Neria Martínez Francisco Javier
3	9/VIII/2012	Se revisaron los objetivos de alto nivel	Neria Martínez Francisco Javier
4	25/IX/2012	Se revisó la introducción, oportunidad de negocio, enunciado del problema, enunciado de la posición.	Hernández Báez Antonio
5	10/X/2012	Se agregó el nombre del sistema a desarrollar.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 1.1: Historial de revisiones del documento de visión

## 1.1 Introducción

El proyecto tiene por proyecto crear un sistema de información con el cual se facilitará la gestión de la información de los docentes, alumnos y planes de estudios pertenecientes a la Coordinación de Estudios de Posgrados de Ciencias Básicas e Ingeniería. El sistema también deberá ser capaz de generar reportes de acuerdo a los parámetros establecidos por el PNPC<sup>1</sup> del Consejo Nacional de Ciencia y Tecnología.

## 1.2 Orientación

### 1.2.1 Oportunidad del negocio

Se han realizado esfuerzos por crear sistemas de información que ayuden a gestionar el archivo del posgrado y el proceso de preregistro de los aspirantes. En el periodo de 2008 a 2010 se implantó un sistema, el cual fue desarrollado por personal de la División de Ciencias Sociales y Humanidades de la UAM Azcapotzalco. Este sistema permite actualizar los datos personales de docentes, el contenido del sitio web y el preregistro de aspirantes a los posgrados. De acuerdo a los comentarios expresados por la Coordinación de Posgrados, este sistema carece de manuales de usuario, manuales de instalación y documentación sobre su diseño. El mismo sistema fue desarrollado en el lenguaje ASP<sup>2</sup> e implantado en un equipo con sistema operativo *Windows 2003 Server* y servidor web IIS<sup>3</sup>, y no se especificó si se contaban con las licencias originales. El sistema tampoco cuenta con cifrado de contraseñas, ni está soportado sobre una capa segura que proteja la integridad de las transacciones entre el cliente y el servidor. Por lo anterior, este sistema fue retirado en el trimestre 10-P.

---

1 PNPC: El Programa Nacional de Posgrados de Calidad de CONACyT solicita a las instituciones registradas en él un CD con un reporte que contenga datos académicos de alumnos, docentes y planes de posgrados, entre otros.

2 ASP (*Active Server Pages*): Tecnología de *Microsoft* para desarrollar páginas web dinámicas.

3 IIS (*Internet Information Services*): Servidor web desarrollado por *Microsoft* para el sistema operativo *Windows*.

## 1.2.2 Enunciado del problema

Los Coordinadores de los posgrados de la División de Ciencias Básicas e Ingeniería tienen la necesidad de llevar el control de la información de los planes de estudio, alumnos y docentes. Esta información se encuentra distribuida en documentación impresa y en archivos en formato electrónico; a los cuales nos referiremos como archivo del posgrado. Actualmente estos archivos se encuentran almacenados en diferentes computadoras lo que dificulta su rápida localización y administración, además, no se lleva un control de las personas que acceden a esta información.

## 1.2.3 Enunciado de la posición

En este proyecto terminal proponemos crear un nuevo sistema de gestión del archivo del posgrado que permita incluir una base de datos para almacenar los datos de los usuarios, y una base de documentos para toda la información: tesis, actas, solicitudes de becas, *kárdex*, y artículos de investigación, entre otros. Para garantizar la seguridad de las transacciones entre el cliente y el servidor, el sistema se implementará sobre una capa de conexión segura, además se usarán contraseñas con cifrado. Se podrá tener acceso al sistema a través de una aplicación cliente (navegador) en diferentes sistemas operativos. Se diseñará una interfaz gráfica de usuario intuitiva, que permita su fácil uso. Además, se podrán generar reportes para el PNPC de CONACyT.

El sistema, el cual denominaremos Sistema de Información de los Posgrados de CBI o SIP-CBI, ofrecerá los siguientes beneficios: se automatizarán los procesos de búsqueda de información, generación de reportes y trámites, se tendrá la información de los alumnos y docentes centralizada, el acceso y transferencias de información se efectuarán de forma segura, contará con una interfaz intuitiva y fácil de utilizar. Además, permitirá una administración adecuada de la información de los alumnos y docentes del posgrado. Este sistema contará con la documentación del análisis, diseño e implementación, lo que permitirá que el sistema sea escalable y adaptable a nuevos requerimientos. También, se contará con un manual de usuario e instalación. El sistema será implementado en su totalidad en software libre.

## 1.3 Personal involucrado

### 1.3.1 Personal por parte del cliente

Actores primarios:

- Coordinadores y personal administrativo encargados de gestionar la información de los alumnos, docentes y planes de estudios; y la generación de los reportes.

Actores secundarios:

- Alumnos y docentes que soliciten actualización de su información o registrar nueva información correspondiente a sus datos personales o académicos.

### 1.3.2 Personal por parte del equipo de desarrollo

Analista/ Diseñador/Programador:

- Hernández Báez Antonio
- Neria Martínez Francisco Javier

### 1.3.3 Objetivos de alto nivel y soluciones actuales por parte del cliente

Objetivos	Prioridad	Solución actual	Problemas e inquietudes
Controlar el acceso a la información del posgrado.	Alta	No se cuenta con un sistema que registre el acceso a la información del posgrado.	Se requiere tener un control del personal que accede a la información del posgrado.
Gestionar información personal y académica de los alumnos del posgrado.	Alta	Las tesis, artículos, comprobantes de movilidad o documentos de antecedentes académicos se encuentran impresos en papel o en formato <i>PDF</i> <sup>4</sup> . Los datos personales se obtienen de documentos impresos en papel o del Sistema Integral de Información UAM (SIUAM). Cuando se requiere un documento se realiza su búsqueda en los archiveros, en las computadoras destinadas a ser repositorios de documentos en <i>PDF</i> y en el SIUAM.	No se tiene orden en el almacenamiento de la información en papel y en formato <i>PDF</i> . Se pierde tiempo en la búsqueda de los documentos.
Gestionar la información de los planes de estudios del posgrado.	Alta	Los planes de estudios se encuentran impresos en papel y en formato <i>PDF</i> .	Cuando se reúne la información para los reportes PNP-C-CONACyT es necesario dividir en secciones el plan de estudios.
Gestionar la información personal y académica de los docentes.	Alta	Los datos personales, antecedentes académicos, información del núcleo básico, proyectos de investigación se solicitan a la División de CBI. La información referente a los artículos de investigación se solicita los coordinadores de posgrado o a los autores del artículo. En algunos casos se cuenta con una copia de los artículos en el archivo impreso del posgrado (biblioteca).	No se tiene orden en el almacenamiento de la información en papel y en formato <i>PDF</i> . Se pierde tiempo en la búsqueda de los documentos.
Reunir la documentación solicitada por el PNP-C-CONACyT.	Alta	La información requerida se busca en los documentos en formato <i>PDF</i> o en los documentos impresos en papel. Si el documento se encuentra aún impreso en papel se escanea para almacenarlo en formato <i>PDF</i> . Los archivos requeridos se organizan en directorios y posteriormente se graban en disco.	No se tiene orden en el almacenamiento de la información en formato <i>PDF</i> . Se pierde tiempo en la búsqueda de los documentos en formato <i>PDF</i> .

Tabla 1.3.3.1: Objetivos de alto nivel y soluciones actuales por parte del cliente

### 1.3.4 Objetivos a nivel de usuario

- Director del proyecto: Supervisar que las tareas se hagan correctamente.
- Personal Administrativo: Administrar y gestionar la información de los alumnos, docentes y planes de estudio.

## 1.4 Visión general del sistema computacional

### 1.4.1 Perspectiva del sistema

El SIP-CBI se instalará en un servidor perteneciente a la Dra. María Lizbeth Gallardo, asesora de este

<sup>4</sup> *PDF* (*Portable Document Format*): Formato de almacenamiento de documentos.

proyecto. En una posible continuación de este proyecto el sistema podría interactuar con otros sistemas como se muestra en la Figura 1.4.1.1.

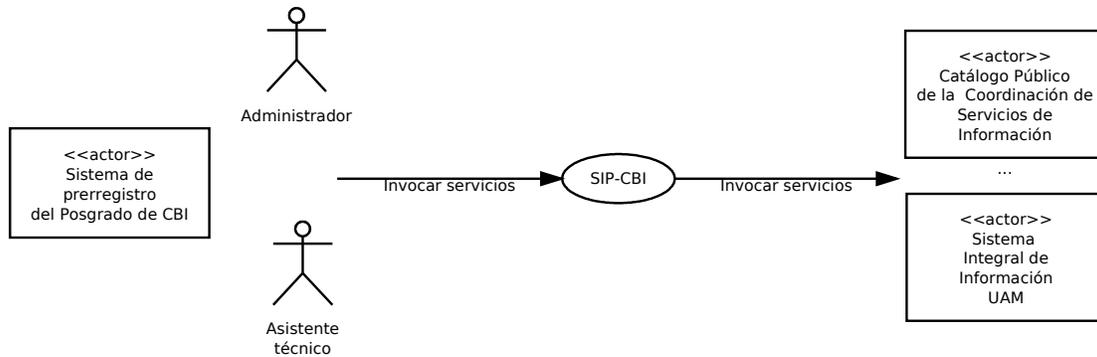


Figura 1.4.1.1: Diagrama de contexto del sistema.

### 1.4.2 Resumen de los beneficios del sistema

Características a Soportar	Beneficio del personal por parte del cliente
El sistema de información que administrará la información de los alumnos, docentes y planes de estudios del posgrado con una página web amigable.	Mediante el uso de una página web se controlará el acceso a la información, así como la captura de la misma, evitando errores por documentos en mal estado.
Gestión de la información de alumnos, docentes y planes de estudios del posgrado.	Mediante la página web se actualizará y/o cambiará la información de los alumnos, profesores y planes de estudio, evitando la generación de un archivo nuevo, si no, que trabajando en el mismo. Se llevará un registro sobre la última fecha de actualización y el tiempo en realizar dicha operación será de 5 minutos en promedio.
Generación de reportes e imagen ISO.	Mediante la página web se seleccionara la información que se desea y permitirá grabar un conjunto de carpetas con la información correspondiente en un CD, reduciendo el tiempo a utilizar en cada operación.

Tabla 1.4.2.1: Beneficios del sistema

### 1.5 Resumen de las características del sistema

- El sistema debe contar con una base de datos que pueda contener la información de alumnos, docentes y planes estudios del posgrado.
- El sistema debe autenticar usuarios
- El sistema debe gestionar la información de los alumnos, docentes y planes estudios del posgrado.
- El sistema deberá gestionar reportes sobre la información de los alumnos, docentes y planes estudios del posgrado.
- El sistema debe de generar una imagen ISO con los parámetros establecidos de la PNPC de CONACyT.

---

## 2 Especificación complementaria

---

Versión	Fecha	Descripción	Autores
borrador	09/V/12	Primer borrador	Hernández Báez Antonio Neria Martínez Francisco Javier
1	13/VII/12	Se agregó reglas de negocio conectables.	Hernández Báez Antonio Neria Martínez Francisco Javier
2	16/VII/12	Se agregó capacidad de recuperación, rendimiento, adaptabilidad, restricciones implementación, interfaz, legales.	Neria Martínez Francisco Javier
3	10/X/12	Se realizaron correcciones en los apartados: funcionalidad, reglas de dominio e información de los dominios de interés.	Hernández Báez Antonio

Tabla 2.1: Historial de revisiones del documento de especificación complementaria

### 2.1 Introducción

El objetivo del proyecto en general es crear un sistema de información con el cual se llevará el control de la información de los docentes, alumnos y planes de estudios pertenecientes a la Coordinación de Estudios de Posgrados de Ciencias Básicas e Ingeniería. Este documento contiene los requisitos del sistema de información a elaborar que no se consideran en los casos de uso.

### 2.2 Funcionalidad

#### 2.2.1 Registro de usuarios

El SIP-CBI dará soporte para registrar vía web a administradores, alumnos, docentes y planes de estudios.

#### 2.2.2 Disponibilidad de información

Una vez registrado el administrador, alumno, docentes o plan de estudios el SIP-CBI tendrá disponible esta información para ser gestionada.

#### 2.2.3 Reportes

El SIP-CBI proporcionará reportes de alumnos, docentes y planes de estudios cuando el administrador lo desee.

#### 2.2.4 Seguridad

El SIP-CBI usará el servidor web *HTTP Apache*<sup>5</sup> sobre una capa de seguridad *SSL*<sup>6</sup> en sus transacciones Cliente-Servidor, así como cifrado de contraseñas en *MD5*<sup>7</sup>. Además, solo dará acceso a administradores registrados en el sistema.

---

5 *HTTP Apache*: Servidor web desarrollado por la *Apache Software Foundation*.

6 *SSL (Secure Sockets Layer)*: Protocolo criptográfico que proporciona comunicaciones seguras en una red.

7 *MD5 (Message-Digest Algorithm 5)*: Algoritmo de reducción criptográfico de 128 bits.

## 2.3 Facilidad de uso

### Factores humanos

- Los administradores (coordinadores y asistentes técnicos) podrán observar todas las opciones disponibles en la interfaz gráfica.
- El SIP-CBI contará con colores, tamaño de fuentes e imágenes que deberán verse claramente en pantallas de equipos de escritorio y portátiles, como laptops.
- El diseño gráfico del sistema de información a desarrollar se basará en el diseño del sitio web de la Unidad, no obstante, se evitará utilizar colores asociados con formas comunes de daltonismo.
- Este sistema contará con la documentación del análisis, diseño e implementación, lo que permitirá que el sistema sea escalable y adaptable a nuevos requerimientos. También, se contará con un manual de usuario e instalación.

Se podrá tener acceso al SIP-CBI, mediante un nombre de usuario y contraseña, a través de una aplicación cliente (navegador) en modo gráfico sin importar el sistema operativo instalado en el equipo cliente. Se diseñará una interfaz gráfica de usuario intuitiva, que permita su fácil uso para consultar la información de los alumnos, docentes y planes de estudio. La interfaz gráfica será dividida en cuatro secciones, lo cual permitirá organizar la información. Tres secciones se mantendrán fijas: *encabezado*, *menú* y *pie de página*. La sección *contenido* permitirá visualizar la información solicitada al SIP-CBI. La sección *menú* será desplegable con el fin de no saturar visualmente la interfaz. En la Figura 2.3.1 se observa el prototipo de interfaz gráfica.



Figura 2.3.1: Prototipo de interfaz gráfica.

Se implementará un diseño con navegabilidad de tres niveles. El menú principal permitirá cargar dos opciones en la zona de *contenido*:

## 2 Especificación complementaria

- Alta. Generará un formulario para operaciones de: alta de usuarios, alumnos, docentes o posgrados. Después de procesar el formulario se mostrará un mensaje de confirmación así como un menú de opciones. En la Figura 2.3.2 se muestra la propuesta de navegabilidad para el proceso de alta.
- Búsqueda. Generará una lista de coincidencias las cuales pueden ser: datos de usuarios, alumnos, docentes o posgrados. Estos datos podrán editarse, para lo cual, se generará en otro nivel un formulario. Después de procesar el formulario se mostrará un mensaje de confirmación así como un menú de opciones. En la Figura 2.3.3 se muestra la propuesta de navegabilidad para el proceso de alta de un alumno.

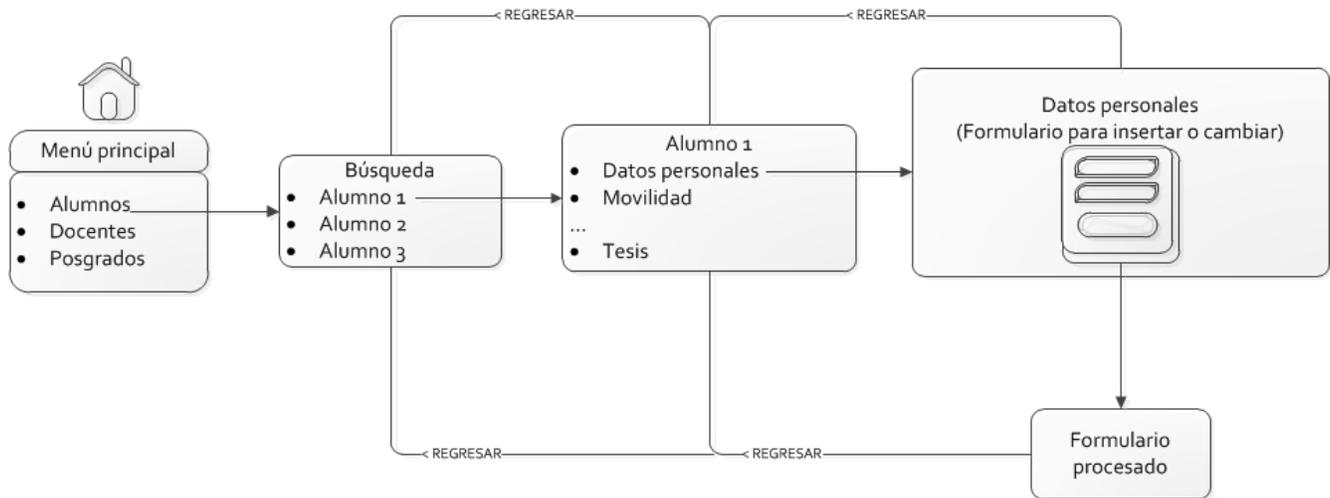


Figura 2.3.2: Propuesta de navegabilidad para el proceso de búsqueda y cambio de información.

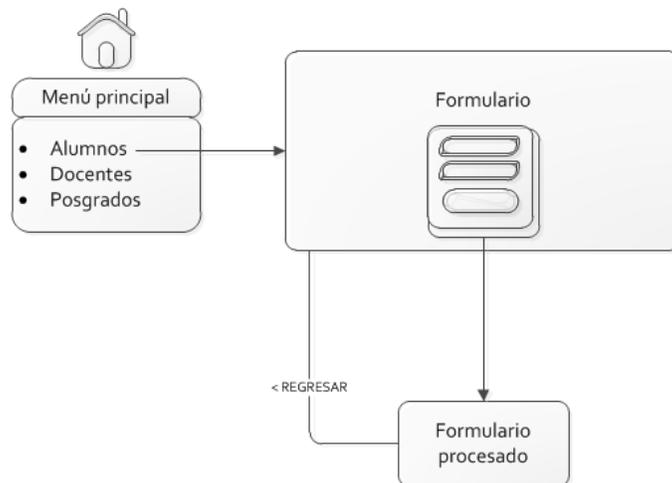


Figura 2.3.3: Propuesta de navegabilidad para el proceso de alta.

### 2.4 Fiabilidad

#### Niveles de aislamiento

La base de datos incluirá niveles de aislamiento para evitar que más de un usuario realice transacciones simultáneas sobre la misma información.

### 2.5 Rendimiento

El SIP-CBI estará en funcionamiento durante las 24 horas del día de tal forma que se garantice el acceso a la información en cualquier momento.

### 2.6 Soporte

#### Adaptabilidad

El sistema de información al ser web, podrá ser utilizado en cualquier computadora que cuente con ambiente gráfico, acceso a Internet y navegador web que soporte *Javascript*, *Ajax* y *Cookies*. Esto permitirá el uso adecuado de la interfaz gráfica del SIP-CBI.

### 2.7 Implementación

Para diseñar el SIP-CBI se utilizará la metodología orientada a objetos llamada proceso unificado y el lenguaje unificado de modelado para representar los sistemas de información [1]. Una de nuestras inquietudes es reforzar nuestros conocimientos en el paradigma orientado a objetos, ya que en el transcurso de nuestra formación profesional el tipo de paradigma que más hemos desarrollado y trabajado ha sido el paradigma estructurado.

*PHP* orientado a objetos será la plataforma de desarrollo, dado que posee una amplia documentación, es un lenguaje orientado a objetos y se encuentra disponible en la distribución *Debian*.

Se utilizará el patrón de desarrollo de software Modelo-Vista-Controlador (MVC) para separar los datos de la aplicación, la interfaz de usuario y la lógica de negocio.

*PHP* será usado para implementar los módulos y la interfaz gráfica de usuario del sistema. Para implementar la interfaz gráfica de usuario, se utilizarán los lenguajes *JavaScript* y *CSS*<sup>8</sup>. Se utilizará *MySQL* para implementar la base de datos. La base de datos incluirá niveles de aislamiento para evitar que más de un usuario realice transacciones simultáneas sobre la misma información. El sistema usará el servidor web *HTTP Apache* sobre una capa de seguridad *SSL* en sus transacciones Cliente-Servidor, así como cifrado de contraseñas en *MD5*. Lo anterior se realizará sobre el sistema operativo *Debian*.

El sistema se instalará en un servidor *HP ProLiant G6 Series* con sistema operativo *Debian v6*, *HTTP Apache server v2.2.16-6*, *MySQL server v5.1.49-3*, *PHP server v5.3* y *OpenSSL*<sup>9</sup> v0.9.8.

---

8 *CSS (Cascading Style Sheets)*: lenguaje usado para definir la presentación de un documento estructurado.

9 *OpenSSL*: Implementación en código abierto de los protocolos *SSL* y *TLS*.

## 2.8 Reglas de dominio (negocio)

ID	Regla	Dirigido a	Fuente
1	Estar o haber estado inscrito en algún posgrado de CBI	Alumno	Reglamento de la UAM
2	Estar o haber estado adscrito a un posgrado de la división de CBI	Docente	Reglamento de la UAM
3	Estar o haber estado vigente dentro de los planes de estudios de la UAM	Planes de estudios	Reglamento de la UAM

Tabla 2.8.1: Reglas de dominio

## 2.9 Información en dominios de interés

- **Alta de usuarios**

Con el SIP-CBI podrán darse de alta usuarios con atributos de administrador A, capaz de dar de alta a otros usuarios y gestionar la información del sistema y administrador tipo B únicamente autorizado para gestionar la información del sistema.

- **Archivo digital**

El SIP-CBI podrá recibir archivos en formato *PDF* vía web los cuales se almacenarán en el servidor donde el sistema resida. Los usuarios del SIP-CBI podrán consultar estos archivos directamente en el servidor o con ayuda del sistema.

## 3 Lógica de negocio

Versión	Fecha	Descripción	Autores
Borrador	13/VI/2012	Borrador de inicio	Hernández Báez Antonio Neria Martínez Francisco Javier
1	10/X/2012	Se revisaron la lógica de negocio propuesta	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 3.1: Historial de revisiones del documento de lógica de negocio

### 3.1 Lógica de negocio actual

- **Alumnos**

Durante el proceso de preregistro se obtienen datos personales, académicos y laborales del aspirante a posgrado. Los datos personales solicitados son: apellidos y nombre(s), correo electrónico, sexo, estado civil, fecha de nacimiento, lugar de nacimiento, nacionalidad, registro federal de contribuyentes, clave única del registro de población, domicilio (calle, número, colonia, ciudad o estado, delegación o municipio, país y código postal), teléfono particular. Los datos académicos solicitados son: posgrado que desea cursar, último grado de estudios, grado obtenido, institución de procedencia, país donde realizó sus estudios, año de finalización de sus estudios, promedio de calificaciones y conocimiento de algún idioma extranjero (inglés, alemán o francés). La información laboral solicitada es: nombre de la institución o empresa, sector al que pertenece la empresa o institución, puesto, tipo de contrato, teléfono/fax de la oficina.

Esta información es proporcionada por el alumno mediante un sistema de preregistro vía web.

También se solicita al aspirante copia en papel o en formato *PDF* de los siguientes documentos: título de licenciatura, certificado de estudios, CURP, certificado de habilidad del idioma inglés, curriculum vitae (CVU de CONACyT) y formato FM3 (en el caso de los extranjeros).

Cuando un aspirante es aceptado cambia su estado a alumno y se genera un expediente para dar seguimiento a su trayectoria académica. A este expediente se anexa información sobre movilidad o intercambio estudiantil, tutorías, tesis y artículos de investigación. Esta información se presenta impresa en papel y en formato electrónico (hojas electrónicas, procesador de textos o formato *PDF*).

- **Planes de estudios**

Se cuenta con copias en papel y en formato *PDF* de los planes de estudios. Los planes de estudios de cada posgrado se encuentran disponibles vía web o se pueden solicitar impresos en la Coordinación de posgrados.

- **Docentes**

Los datos personales y académicos, núcleo básico, tesis dirigidas y proyectos de investigación de los docentes se encuentran en expedientes de papel almacenados en archiveros, archivos de hojas de cálculo y procesadores de textos. Cuando se desea realizar una actualización se solicita al docente o al coordinador correspondiente, en persona o por correo electrónico, envíe un

listado con los datos actualizados.

- **Informe en CD al PNPC-CONACyT**

Algunos posgrados se encuentran adscritos al Programa Nacional de Posgrados de Calidad de CONACyT. Durante el proceso de evaluación que realiza CONACyT, cada 2 años, se solicita un *CD* con información del posgrado a evaluar de acuerdo al documento denominado Control de medios de verificación PNPC<sup>10</sup>.

Para elaborar el *CD* se escanean los documentos indicados en el Control de medios de verificación PNPC o en caso de estar ya escaneados y guardados en archivos en formato *PDF* se buscan en las computadoras que han sido destinadas como repositorios de documentos. Se crea la estructura de directorios solicitada, se colocan los archivos en formato *PDF* en los directorios correspondientes y se procede a su grabado en un medio físico (*CD* o *DVD*).

## 3.2 Lógica de negocio propuesta

- **Alumnos**

Los datos personales, académicos y laborales de los alumnos podrán solicitarse al sistema de prerregistro cambiando el estado de aspirante a alumno. En el caso de alumnos o exalumnos que hayan sido matriculados antes de la creación del sistema de prerregistro podrán ser agregados al sistema. Una vez realizada el alta del alumno podrá hacerse cambios, consultas y reportes de la información. En los casos donde se requiera anexar documentación en formato *PDF* el sistema permitirá anexar estos documentos para su resguardo en el servidor.

- **Planes de estudios**

El sistema permitirá guardar los documentos relacionados con los planes de estudios. En los casos donde se requiera anexar documentación en formato *PDF* el sistema permitirá anexar estos documentos para su resguardo en el servidor.

- **Docentes**

Podrán realizarse altas, bajas, cambios y consulta de información de docentes. En los casos donde se requiera anexar documentación en formato *PDF* el sistema permitirá anexar estos documentos para su resguardo en el servidor.

- **Informes para el PNPC-CONACyT**

Al tener la información sobre alumnos, docentes y planes de estudios en una base de datos se facilitará la localización de los archivos solicitados por el PNPC de CONACyT.

---

10 Control de medios de verificación PNPC: documento que indica los nombres de los directorios y el contenido que estos deberán incluir en el *CD*. Este documento considera plan de estudios, proceso de enseñanza-aprendizaje, ingreso de estudiantes, trayectoria escolar, movilidad e intercambio de estudiantes, tutorías y asesorías, núcleo académico básico, entre otros.

---

## 4 Glosario

---

<b>Versión</b>	<b>Fecha</b>	<b>Descripción</b>	<b>Autores</b>
borrador	10/06/12	Primer borrador del documento	Hernández Báez Antonio Neria Martínez Francisco Javier
1	27/06/12	Se agregó el término usuario.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 4.1: Historial de revisiones del documento glosario.

<b>Término</b>	<b>Definición e información</b>	<b>Alias</b>
Baja	Representa la acción de inhabilitar. Los usuarios, alumnos, docentes y planes de estudios no se eliminarán, únicamente se inhabilitará su acceso.	Eliminar
Usuario	Actor primario que gestiona la información del sistema.	Administrador tipo A o B
ACL	Lista de control de acceso	Lista

Tabla 4.2: Tabla de definiciones del glosario.

## 5 Diagramas de casos de uso

Versión	Fecha	Descripción	Autores
borrador	10/VI/2012	Primer borrador del documento	Hernández Báez Antonio Neria Martínez Francisco Javier
1	27/VI/2012	Se agregaron diagramas del caso de uso Docentes	Hernández Báez Antonio Neria Martínez Francisco Javier
2	10/IX/2012	Se agregaron los diagramas del caso de uso planes de estudio	Hernández Báez Antonio
3	13/IX/2012	Se corrigieron los diagramas del caso de uso planes de estudio	Hernández Báez Antonio
4	16/IX/2012	Se agregó el diagrama del caso de uso autenticar usuarios. Se agregó éste caso de uso al diagrama de casos de uso generales.	Hernández Báez Antonio
5	12/X/2012	Se agregó el nombre del sistema (SIP-CBI) a cada diagrama.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 5.1: Historial de revisiones del documento diagramas de casos de uso

### 5.1 Casos de uso generales

Como resultado del análisis de requerimientos se detectaron cuatro grandes casos de uso: gestionar usuarios, gestionar alumnos, gestionar planes de estudio y gestionar docentes. Los casos de uso incluyen al caso “gestionar autenticación de usuarios” ya que es necesario estar autenticado dentro del sistema, con un nombre de usuario y contraseña, para poder utilizar los demás casos de uso. También se identificaron dos tipos de actores primarios que interactúan con el sistema: administrador tipo A y tipo B. Ambos administradores podrán realizar operaciones de alta, baja, cambios y consultas en los cuatro casos de uso principales a excepción del administrador tipo B quién no podrá gestionar usuarios. El SIP-CBI, actores primarios y casos de uso se representan gráficamente en la Figura 5.1.1.

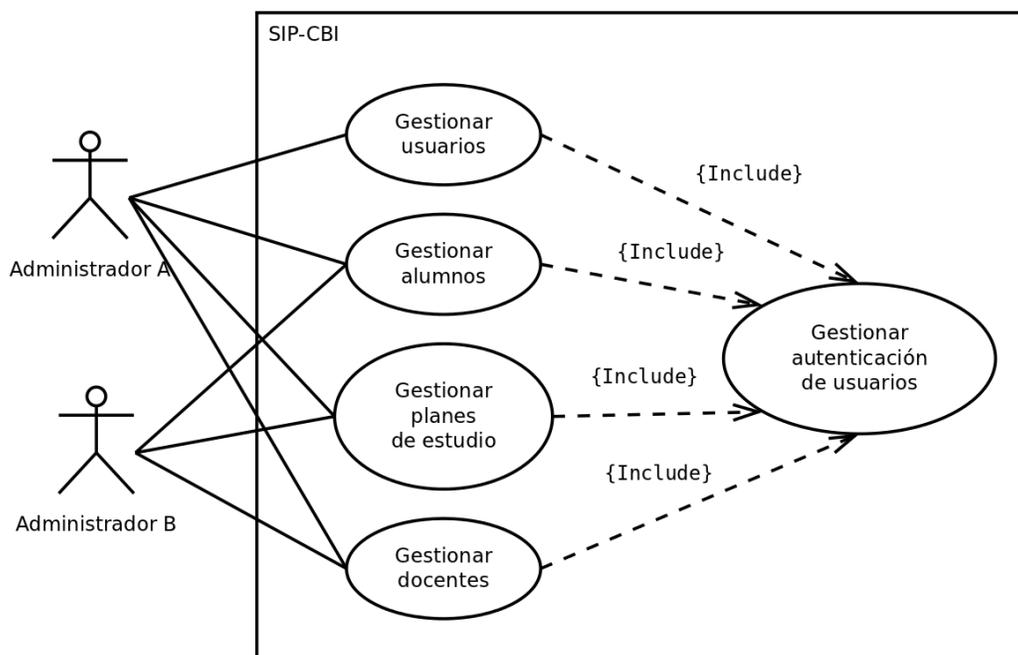


Figura 5.1.1: Casos de uso generales del sistema.

Cada caso de uso representado en la Figura 5.1.1 contiene subcasos de uso que representan las operaciones alta, baja, cambios y consultas. A continuación se muestran los diagramas de estos subcasos.

### 5.2 Caso de uso gestionar autenticación de usuarios

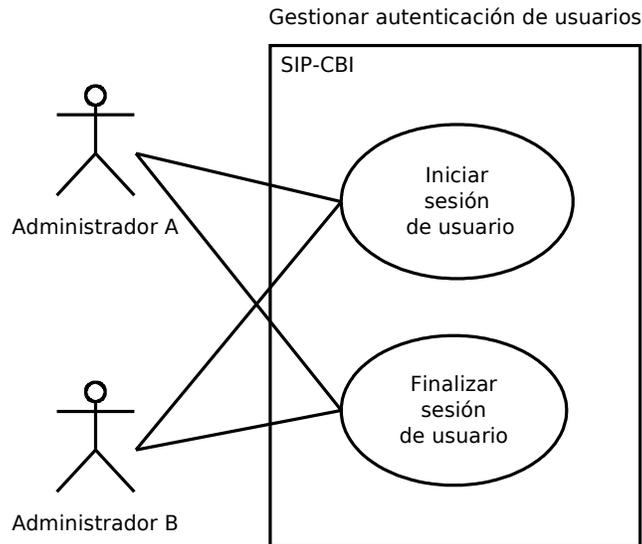


Figura 5.2.1: Caso de uso gestionar autenticación de usuarios.

### 5.3 Caso de uso gestionar usuarios

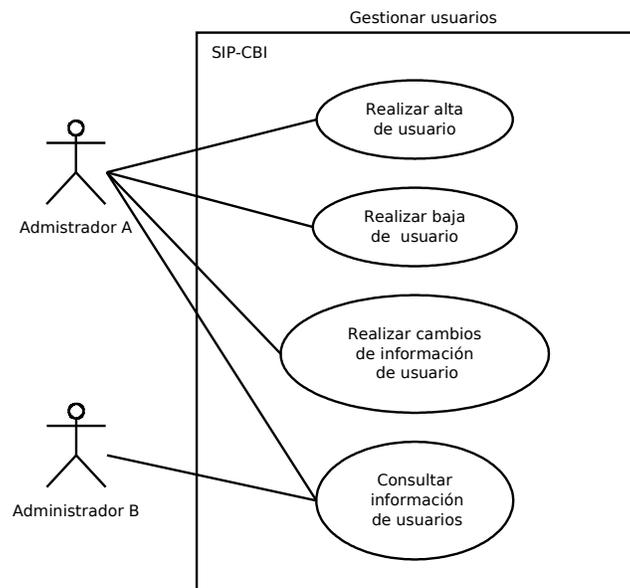


Figura 5.3.1: Caso de uso gestionar usuarios.

### 5.4 Caso de uso gestionar alumnos

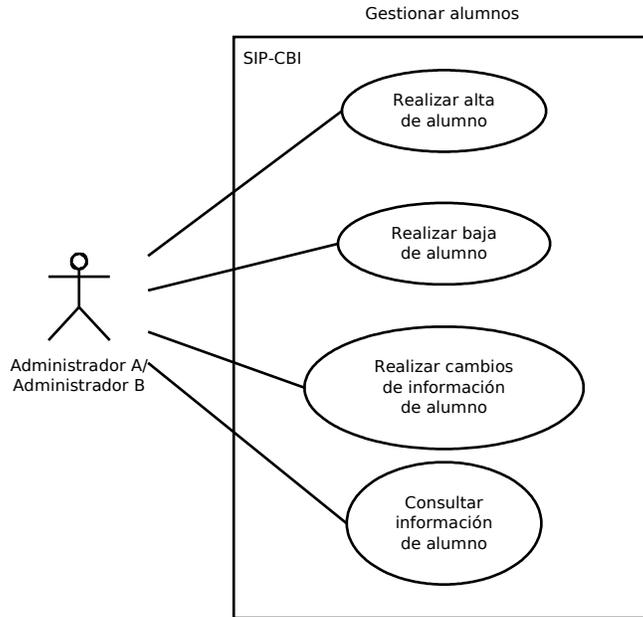


Figura 5.4.1: Caso de uso gestionar alumnos.

### 5.5 Caso de uso gestionar planes de estudios

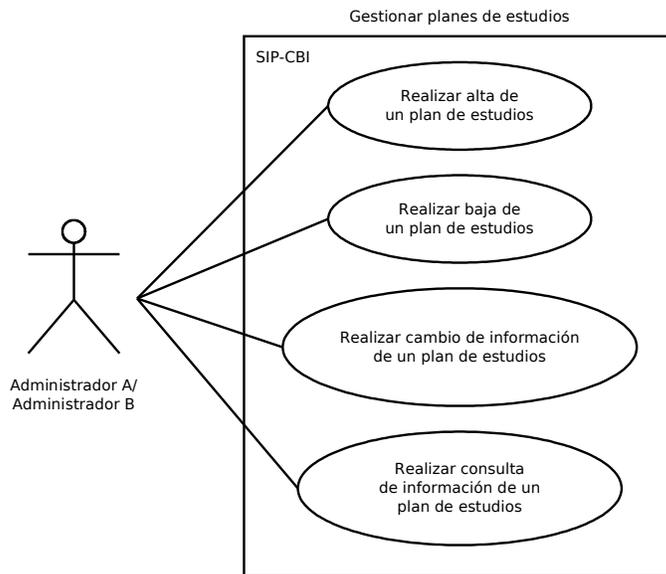


Figura 5.5.1: Caso de uso gestionar planes de estudio.

## 5.6 Caso de uso gestionar docentes

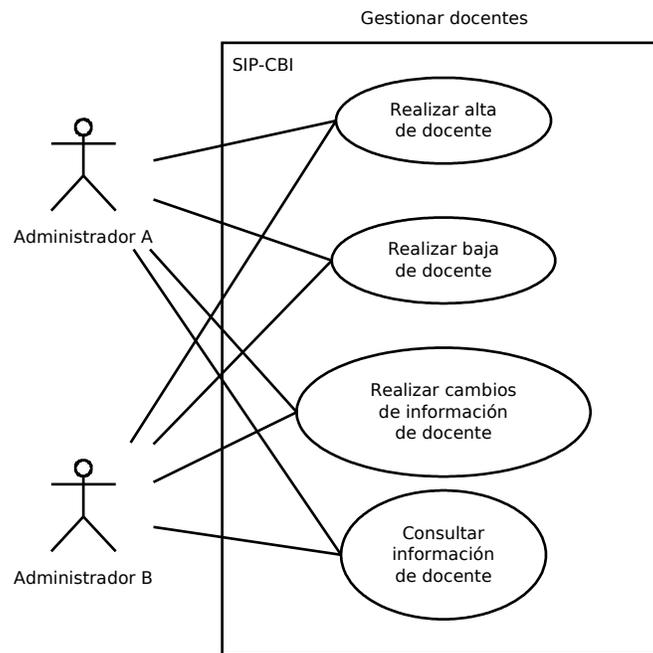


Figura 5.6.1: Caso de uso gestionar docentes.

---

## 6 Escenarios

---

Versión	Fecha	Descripción	Autores
Borrador	13/VI/2012	Borrador de inicio.	Hernández Báez Antonio Neria Martínez Francisco Javier
1	21/VI/2012	Se asignaron claves a los casos de uso.	Hernández Báez Antonio Neria Martínez Francisco Javier
2	28/VI/2012	Se agregó el caso de uso gestionar usuarios.	Hernández Báez Antonio Neria Martínez Francisco Javier
3	8/VII/2012	Se organizó el caso de uso gestionar docentes por operación.	Hernández Báez Antonio Neria Martínez Francisco Javier
4	2/VIII/2012	Se agregaron los escenarios para el caso de uso gestionar alumnos. Se agregaron los escenarios para el caso de uso gestionar planes de estudios.	Hernández Báez Antonio Neria Martínez Francisco Javier
5	17/IX/2012	Se agregó el caso de uso gestionar autenticación de usuarios.	Hernández Báez Antonio
6	21/IX/2012	Se modificó el escenario para el caso de uso gestionar planes de estudios.	Neria Martínez Francisco Javier

Tabla 6.1: Historial de revisiones del documento de escenarios

El siguiente paso en nuestro diseño fue determinar los escenarios en base a los casos de uso planteados anteriormente. A cada escenario se le asignó una clave y un número. Los escenarios del SIP-CBI son:

- GAU: Gestionar autenticación de usuarios.
  - GAU1: Iniciar sesión de usuario.
  - GAU2: Finalizar sesión de usuario.
- GU: Gestionar usuarios.
  - GU1: Realizar alta de usuario.
  - GU2: Realizar baja de usuario.
  - GU3: Realizar cambios de información de usuario.
  - GU4: Consultar información de usuario.
- GA: Gestionar alumnos.
  - GA1: Realizar alta de alumno.
  - GA2: Realizar baja de alumno.
  - GA3: Realizar cambios de información de alumno.
  - GA4: Consultar información de alumno.
- GPE: Gestionar planes de estudios.
  - GPE1: realizar alta de un plan de estudios.
  - GPE2: realizar baja de un plan de estudios.
  - GPE3: realizar cambios de información un plan de estudios.

- GPE4: Consultar información de un plan de estudios.
- GD: Gestionar docentes.
  - GD1: Realizar alta de docente.
  - GD1: Realizar baja de docente.
  - GD1: Realizar cambios en información de docente.
  - GD1: Consultar información de docente.

A continuación se describen los escenarios del SIP-CBI.

## **6.1 GAU: Gestionar autenticación de usuarios**

### **6.1.1 GAU1: Iniciar sesión de usuario**

#### **6.1.1.1 Descripción**

Caso de uso que autentica a un actor primario y crea una sesión asociada a él.

#### **6.1.1.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

#### **6.1.1.3 Disparador**

El actor primario ingresa al sistema mediante un nombre de usuario y contraseña.

#### **6.1.1.4 Precondiciones**

No existe una sesión asociada al usuario que pretende autenticarse.

#### **6.1.1.5 Postcondiciones**

El usuario fue autenticado en el sistema y se genera una sesión asociada a él.

#### **6.1.1.6 Flujo principal**

1. El actor primario ingresa vía web al sistema que se encuentra bajo una capa SSL.
2. El sistema verifica si existe una sesión asociada al actor primario.
3. El sistema muestra la pantalla de acceso al sistema.
4. El actor primario ingresa su nombre de usuario y contraseña.
5. El sistema busca en su base de datos si existe un usuario con el nombre y contraseña

suministrados.

6. El sistema autentica al usuario y crea una sesión asociada a él.
7. El sistema muestra el menú principal de módulos.
8. El actor primario selecciona un módulo (El sistema a partir de este momento verificará constantemente si la sesión se encuentra activa).

#### **6.1.1.7 Flujos alternativos**

6.1 El sistema no encuentra un usuario con el nombre suministrado o la contraseña no coincide.

6.1.1 El sistema envía un mensaje de error al actor primario.

6.1.2 Regresa la punto 3 del flujo principal.

#### **6.1.1.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

### **6.1.2 GAU2: Finalizar sesión de usuario**

#### **6.1.2.1 Descripción**

Caso de uso que permite finalizar la sesión asociada a un actor primario autenticado en el sistema.

#### **6.1.2.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

#### **6.1.2.3 Disparador**

El actor primario solicita al sistema finalizar la sesión asociada a él.

#### **6.1.2.4 Precondiciones**

Existe una sesión asociada al actor primario.

#### **6.1.2.5 Postcondiciones**

La sesión fue finalizada.

#### **6.1.2.6 Flujo principal**

1. El actor primario solicita al sistema finalizar la sesión.

2. El sistema pregunta al usuario si desea o no finalizar la sesión.
3. El actor primario confirma la finalizar la sesión.
4. El sistema finaliza la sesión.
5. El sistema muestra la pantalla de acceso al sistema

#### **6.1.2.7 Flujos alternativos**

- 1.1 El actor primario cierra el navegador.
  - 1.1.1 El sistema finaliza la sesión asociada al usuario.
- 2.1 El actor primario no finaliza la sesión.

#### **6.1.2.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

### **6.2 GU: Gestionar usuarios**

#### **6.2.1 GU1: Realizar alta de usuario**

##### **6.2.1.1 Descripción**

Caso de uso que permite realizar el alta a un usuario.

##### **6.2.1.2 Actores**

Primarios: administrador tipo A.

##### **6.2.1.3 Disparador**

El actor primario solicita al sistema realizar el dar de alta a un usuario.

##### **6.2.1.4 Precondiciones**

El usuario no se encuentra registrado en el sistema.

##### **6.2.1.5 Postcondiciones**

Se registra un usuario en el sistema.

##### **6.2.1.6 Flujo principal**

1. El actor primario ingresa al módulo para realizar la alta de un usuario.

2. El actor primario selecciona la opción “alta de un usuario”.
3. El sistema muestra un formulario para ingresar los datos del usuario.
4. El actor primario ingresa los datos solicitados en el formulario.
5. El actor primario selecciona la opción “agregar nuevo registro”.
6. El sistema solicita confirmación.
7. El actor primario confirma la solicitud.
8. El sistema realiza el registro.
9. El sistema indica al actor primario que su solicitud a tenido éxito.
10. El sistema indica al actor primario si desea realizar otro registro.
11. El actor primario sale del módulo.

#### **6.2.1.7 Flujos alternativos**

- 3.1. El actor primario cancela la operación.
  - 3.1.1. Regresa al punto 2 del flujo principal.
- 4.1. El actor primario cancela la operación.
  - 4.1.1. Regresa al punto 3 del flujo principal.
- 7.1. El sistema indica que ya existe un registro igual.
  - 7.1.1. El sistema muestra el formulario con los datos suministrados.
  - 7.1.2. El actor primario suministra un registro con datos distintos.
  - 7.1.3. Regresa al punto 5 del flujo principal.

#### **6.2.1.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

### **6.2.2 GU2: Realizar baja de usuario**

#### **6.2.2.1 Descripción**

Caso de uso que permite dar de baja a un usuario.

#### **6.2.2.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

### **6.2.2.3 Disparador**

El actor primario solicita al sistema dar de baja a un usuario.

### **6.2.2.4 Precondiciones**

El usuario se encuentra registrado en el sistema.

### **6.2.2.5 Postcondiciones**

Un usuario es dado de baja en el sistema.

### **6.2.2.6 Flujo principal**

1. El actor primario ingresa al módulo para realizar la baja de un usuario.
2. El sistema muestra en pantalla la lista de usuarios registrados.
3. El actor primario selecciona un usuario.
4. El actor primario selecciona la opción “dar de baja”.
5. El sistema muestra una notificación en pantalla para confirmar la baja del usuario.
6. El actor primario confirma la baja del usuario.
7. El sistema realiza la baja del usuario.
8. El sistema muestra una pantalla indicando que el usuario ha sido dado de baja.
9. El sistema muestra un mensaje indicando si se desea realizar otra operación.
10. El actor primario sale del módulo.

### **6.2.2.7 Flujos alternativos**

6.1 El actor primario cancela la baja de usuario.

6.1.1. Regresa al punto 2 del flujo principal.

### **6.2.2.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

## **6.2.3 GU3: Realizar cambios de información de usuario**

### **6.2.3.1 Descripción**

Caso de uso que permite realizar cambios de información de un usuario.

### 6.2.3.2 Actores

Primarios: administrador tipo A, administrador tipo B.

### 6.2.3.3 Disparador

El actor primario solicita al sistema modificar la información de un usuario.

### 6.2.3.4 Precondiciones

El usuario se encuentra registrado en el sistema.

### 6.2.3.5 Postcondiciones

Se modificó la información del usuario.

### 6.2.3.6 Flujo principal

1. El actor primario ingresa al módulo para realizar cambios de información de usuarios.
2. El sistema muestra en pantalla la lista de usuarios registrados.
3. El actor primario selecciona un usuario.
4. El sistema busca la información del usuario seleccionado.
5. El sistema muestra en pantalla la información del usuario seleccionado.
6. El actor primario realiza los cambios en la información.
7. El actor primario acepta los cambios.
8. El sistema solicita confirmación por parte del actor primario.
9. El sistema modifica la información del usuario.
10. El sistema muestra un mensaje indicando si se desea realizar otra operación.
11. El actor primario sale del módulo.

### 6.2.3.7 Flujos alternativos

8.1 El actor primario cancela los cambios de información.

8.1.1. Regresa al punto 2 del flujo principal.

### 6.2.3.8 Requerimientos no funcionales

Este proceso no debe tardar más de 2 minutos.

## **6.2.4 GU4: Consultar información de usuario**

### **6.2.4.1 Descripción**

Caso de uso que permite consultar la información de un usuario.

### **6.2.4.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

### **6.2.4.3 Disparador**

El actor primario solicita al sistema consultar la información de un usuario.

### **6.2.4.4 Precondiciones**

El usuario se encuentra registrado en el sistema.

### **6.2.4.5 Postcondiciones**

Se consultó la información del usuario.

### **6.2.4.6 Flujo principal**

1. El actor primario ingresa al módulo para consultar la información de usuario.
2. El sistema muestra en pantalla la lista de usuarios registrados.
3. El actor primario selecciona un usuario.
4. El sistema busca la información del usuario seleccionado.
5. El sistema muestra en la pantalla la información del usuario seleccionado.
6. El actor primario sale del módulo.

### **6.2.4.7 Flujos alternativos**

- 5.1 El actor primario regresa a 2.

### **6.2.4.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

## **6.3 GA: Gestionar alumnos**

### **6.3.1 GA1: Realizar alta de alumno**

#### **6.3.1.1 Descripción**

Caso de uso que permite realizar alta de un alumno en el sistema.

#### **6.3.1.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

#### **6.3.1.3 Disparador**

El actor primario solicita realizar el alta de un alumno

#### **6.3.1.4 Precondiciones**

El alumno no se encuentra registrado en el sistema.

#### **6.3.1.5 Postcondiciones**

El alumno se encuentra registrado en el sistema.

#### **6.3.1.6 Flujo principal**

12. El actor primario ingresa al módulo para realizar la alta de un alumno.
13. El actor primario selecciona la opción “alta de un alumno”.
14. El sistema muestra un formulario para ingresar los datos del alumno.
15. El actor primario ingresa los datos solicitados en el formulario.
16. El actor primario selecciona la opción “agregar nuevo registro”.
17. El sistema solicita confirmación.
18. El actor primario confirma la solicitud.
19. El sistema realiza el registro.
20. El sistema indica al actor primario que su solicitud a tenido éxito.
21. El sistema indica al actor primario si desea realizar otro registro.
22. El actor primario sale del módulo.

#### **6.3.1.7 Flujos alternativos**

- 3.1. El actor primario cancela la operación.

3.1.1. Regresa al punto 2 del flujo principal.

4.1. El actor primario cancela la operación.

4.1.1. Regresa al punto 3 del flujo principal.

7.1. El sistema indica que ya existe un registro igual.

7.1.4. El sistema muestra el formulario con los datos suministrados.

7.1.5. El actor primario suministra un registro con datos distintos.

7.1.6. Regresa al punto 5 del flujo principal.

### **6.3.1.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

## **6.3.2 GA2: Realizar baja de alumno**

### **6.3.2.1 Descripción**

Caso de uso que permite dar de baja a un usuario.

### **6.3.2.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

### **6.3.2.3 Disparador**

El actor primario solicita al sistema dar de baja a un alumno.

### **6.3.2.4 Precondiciones**

El alumno se encuentra registrado en el sistema.

### **6.3.2.5 Postcondiciones**

Un alumno es dado de baja en el sistema.

### **6.3.2.6 Flujo principal**

1. El actor primario ingresa al módulo para realizar la baja de un alumno.
2. El sistema muestra en pantalla la lista de alumnos registrados.
3. El actor primario selecciona un alumno.
4. El actor primario selecciona la opción “dar de baja”.

5. El sistema muestra una notificación en pantalla para confirmar la baja del alumno.
6. El actor primario confirma la baja del alumno.
7. El sistema realiza la baja del alumno.
8. El sistema muestra una pantalla indicando que el alumno ha sido dado de baja.
9. El sistema muestra un mensaje indicando si se desea realizar otra operación.
10. El actor primario sale del módulo.

### **6.3.2.7 Flujos alternativos**

6.1 El actor primario cancela la baja del alumno.

- 6.1.1. Regresa al punto 2 del flujo principal.

### **6.3.2.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

## **6.3.3 GA3: Realizar cambios de información de alumno**

### **6.3.3.1 Descripción**

Caso de uso que permite realizar cambios de información de un alumno.

### **6.3.3.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

### **6.3.3.3 Disparador**

El actor primario solicita al sistema modificar la información de un alumno.

### **6.3.3.4 Precondiciones**

El alumno se encuentra registrado en el sistema.

### **6.3.3.5 Postcondiciones**

Se modificó la información del alumno.

### **6.3.3.6 Flujo principal**

1. El actor primario ingresa al módulo para realizar cambios de información de alumnos.

2. El sistema muestra en pantalla la lista de alumnos registrados.
3. El actor primario selecciona un alumno.
4. El sistema busca la información del alumno seleccionado.
5. El sistema muestra en pantalla la información del alumno seleccionado.
6. El actor primario realiza los cambios en la información.
7. El actor primario acepta los cambios.
8. El sistema solicita confirmación por parte del actor primario.
9. El sistema modifica la información del alumno.
10. El sistema muestra un mensaje indicando si se desea realizar otra operación.
11. El actor primario sale del módulo.

#### **6.3.3.7 Flujos alternativos**

- 8.1 El actor primario cancela los cambios de información.
  - 8.1.1. Regresa al punto 2 del flujo principal.

#### **6.3.3.8 Requerimientos no funcionales**

### **6.3.4 GA4: Consultar información de alumno**

#### **6.3.4.1 Descripción**

Caso de uso que permite consultar la información de un alumno.

#### **6.3.4.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

#### **6.3.4.3 Disparador**

El actor primario solicita al sistema consultar la información de un alumno.

#### **6.3.4.4 Precondiciones**

El alumno se encuentra registrado en el sistema.

#### **6.3.4.5 Postcondiciones**

Se consultó la información del alumno.

#### **6.3.4.6 Flujo principal**

1. El actor primario ingresa al módulo para consultar la información de alumno.
2. El sistema muestra en pantalla la lista de alumnos registrados.
3. El actor primario selecciona un alumno.
4. El sistema busca la información del alumno seleccionado.
5. El sistema muestra en la pantalla la información del alumno seleccionado.
6. El actor primario sale del módulo.

#### **6.3.4.7 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

### **6.4 GPE: Gestionar planes de estudio**

#### **6.4.1 GPE1: Realizar alta de un plan de estudios**

##### **6.4.1.1 Descripción**

Caso de uso que permite realizar alta de un plan de estudios en el sistema.

##### **6.4.1.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

##### **6.4.1.3 Disparador**

El actor primario solicita realizar alta de un plan de estudios

##### **6.4.1.4 Precondiciones**

El plan de estudios no se encuentra registrado en el sistema.

##### **6.4.1.5 Postcondiciones**

El plan de estudios se encuentra registrado en el sistema.

##### **6.4.1.6 Flujo principal**

1. El actor primario ingresa al módulo para realizar la alta de un plan de estudios
2. El actor primario selecciona la opción “alta de un plan de estudios”
3. El sistema muestra un formulario para ingresar el plan de estudios

4. El actor primario ingresa los datos solicitados en el formulario.
5. El actor primario selecciona la opción “agregar nuevo registro”.
6. El sistema solicita confirmación.
7. El actor primario confirma la solicitud.
8. El sistema realiza el registro.
9. El sistema indica al actor primario que su solicitud a tenido éxito.
10. El sistema indica al actor primario si desea realizar otro registro.
11. El actor primario sale del módulo.

#### **6.4.1.7 Flujos alternativos**

- 3.1. El actor primario cancela la operación.
  - 3.1.1. Regresa al punto 2 del flujo principal.
- 4.1. El actor primario cancela la operación.
  - 4.1.1. Regresa al punto 3 del flujo principal.
- 7.1. El sistema indica que ya existe un registro igual.
  - 7.1.1. El sistema muestra el formulario con los datos suministrados.
  - 7.1.2. El actor primario suministra un registro con datos distintos.
  - 7.1.3. Regresa al punto 5 del flujo principal.

#### **6.4.1.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

### **6.4.2 GPE2: Realizar baja de un plan de estudios**

#### **6.4.2.1 Descripción**

Caso de uso que permirte realizar baja de un plan de estudios

#### **6.4.2.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

#### **6.4.2.3 Disparador**

El actor primario solicita realizar baja de un plan de estudios

#### **6.4.2.4 Precondiciones**

El plan de estudios se encuentra registrado en el sistema.

#### **6.4.2.5 Postcondiciones**

El plan de estudios fue dado de baja del sistema.

#### **6.4.2.6 Flujo principal**

1. El actor primario ingresa al módulo realizar baja de plan de estudios
2. El sistema muestra una lista de los planes de estudios
3. El actor primario selecciona un registro.
4. El actor primario selecciona la opción “realizar baja de registro”.
5. El sistema solicita confirmación.
7. El actor primario confirma la solicitud.
8. El sistema realiza la baja del registro.
9. El sistema indica al actor primario que su solicitud a tenido éxito.
10. El sistema indica al actor primario si desea realizar la baja de otro registro.
11. El actor primario sale del módulo.

#### **6.4.2.7 Flujos alternativos**

- 3.1. El actor primario cancela la operación.
  - 3.1.1. Regresa al punto 2 del flujo principal.
- 4.1 El actor primario cancela la operación.
  - 4.1.1. Regresa al punto 3 del flujo principal.

#### **6.4.2.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

### **6.4.3 GPE3: Realizar cambio de información de un plan de estudios**

#### **6.4.3.1 Descripción**

Caso de uso que permite realizar cambio de información de un plan de estudios.

### 6.4.3.2 Actores

Primarios: administrador tipo A, administrador tipo B.

### 6.4.3.3 Disparador

El actor primario solicita realizar cambio de información de un plan de estudios

### 6.4.3.4 Precondiciones

El plan de estudios se encuentra registrado en el sistema.

### 6.4.3.5 Postcondiciones

Se realizaron cambios de información en el plan de estudios vigente.

### 6.4.3.6 Flujo principal

1. El actor primario ingresa al módulo realizar cambio de información de un plan de estudios
2. El sistema muestra una lista de los planes de estudios
3. El actor primario selecciona un registro.
4. El sistema realiza la búsqueda de información del registro seleccionado.
5. El sistema muestra la información del registro en un formulario.
6. El actor primario realiza cambios en la información del registro.
7. El actor primario selecciona la opción “realizar cambios en el registro”.
8. El sistema solicita confirmación.
9. El actor primario confirma la solicitud.
10. El sistema realiza los cambios en el registro.
11. El sistema indica al actor primario que su solicitud a tenido éxito.
12. El sistema indica al actor primario si desea realizar cambios en otro registro.
13. El actor primario sale del módulo.

### 6.4.3.7 Flujos alternativos

- 3.1. El actor principal cancela la operación
    - 3.1.1. Regresa al punto 2 del flujo principal.
- De 4 a 7: El actor primario cancela la operación.
- Regresa al punto 3 del flujo principal

### **6.4.3.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

## **6.4.4 GPE4: Consultar información de un plan de estudios**

### **6.4.4.1 Descripción**

Caso de uso que permite realizar la consulta de información de un plan de estudios.

### **6.4.4.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

### **6.4.4.3 Disparador**

El actor primario solicita realizar consulta de información de un plan de estudios.

### **6.4.4.4 Precondiciones**

El plan de estudios se encuentra registrado en el sistema.

### **6.4.4.5 Postcondiciones**

Se realizó una consulta de información en el plan de estudios vigente.

### **6.4.4.6 Flujo principal**

1. El actor primario ingresa al módulo realizar consulta de información de un plan de estudios
2. El sistema muestra una lista de los planes de estudios
3. El actor primario selecciona un registro.
4. El sistema realiza la búsqueda de información del registro seleccionado.
5. El sistema muestra la información del registro en pantalla.
6. El actor primario sale del módulo.

### **6.4.4.7 Flujos alternativos**

- 3.1. El actor primario cancela la operación.
    - 3.1.1. Regresa al punto 2 del flujo principal.
- De 4 y 5: El actor primario cancela la operación
- Regresa al punto 3 del flujo principal.

#### **6.4.4.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

### **6.5 GD: Gestionar docentes**

#### **6.5.1 GD1: Realizar alta de docente**

##### **6.5.1.1 Descripción**

Caso de uso que permite realizar alta de un docente en el sistema.

##### **6.5.1.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

##### **6.5.1.3 Disparador**

El actor primario solicita realizar el alta de un docente.

##### **6.5.1.4 Precondiciones**

El docente no se encuentra registrado en el sistema.

##### **6.5.1.5 Postcondiciones**

El docente se encuentra registrado en el sistema.

##### **6.5.1.6 Flujo principal**

1. El actor primario ingresa al módulo para realizar la alta de un docente.
2. El actor primario selecciona la opción “alta de un docente”.
3. El sistema muestra un formulario para ingresar los datos del docente.
4. El actor primario ingresa los datos solicitados en el formulario.
5. El actor primario selecciona la opción “agregar nuevo registro”.
6. El sistema solicita confirmación.
7. El actor primario confirma la solicitud.
8. El sistema realiza el registro.
9. El sistema indica al actor primario que su solicitud a tenido éxito.
10. El sistema indica al actor primario si desea realizar otro registro.
11. El actor primario sale del módulo.

### **6.5.1.7 Flujos alternativos**

- 3.1. El actor primario cancela la operación.
  - 3.1.1. Regresa al punto 2 del flujo principal.
- 4.1. El actor primario cancela la operación.
  - 4.1.1. Regresa al punto 3 del flujo principal.
- 7.1. El sistema indica que ya existe un registro igual.
  - 7.1.1. El sistema muestra el formulario con los datos suministrados.
  - 7.1.2. El actor primario suministra un registro con datos distintos.
  - 7.1.3. Regresa al punto 5 del flujo principal.

### **6.5.1.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

## **6.5.2 GD2: Realizar baja de docente**

### **6.5.2.1 Descripción**

Caso de uso que permite dar de baja a un docente.

### **6.5.2.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

### **6.5.2.3 Disparador**

El actor primario solicita al sistema dar de baja a un docente.

### **6.5.2.4 Precondiciones**

El docente se encuentra registrado en el sistema.

### **6.5.2.5 Postcondiciones**

Un docente es dado de baja en el sistema.

### **6.5.2.6 Flujo principal**

1. El actor primario ingresa al módulo para realizar la baja de un docente.
2. El sistema muestra en pantalla la lista de docentes registrados.

3. El actor primario selecciona un docente.
4. El actor primario selecciona la opción “dar de baja”.
5. El sistema muestra una notificación en pantalla para confirmar la baja del docente.
6. El actor primario confirma la baja del docente.
7. El sistema realiza la baja del docente.
8. El sistema muestra una pantalla indicando que el docente ha sido dado de baja.
9. El sistema muestra un mensaje indicando si se desea realizar otra operación.
10. El actor primario sale del módulo.

#### **6.5.2.7 Flujos alternativos**

6.1 El actor primario cancela la baja del docente.

6.1.1. Regresa al punto 2 del flujo principal.

#### **6.5.2.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

### **6.5.3 GD3: Realizar cambios en información de docente**

#### **6.5.3.1 Descripción**

Caso de uso que permite realizar cambios de información de un docente.

#### **6.5.3.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

#### **6.5.3.3 Disparador**

El actor primario solicita al sistema modificar la información de un docente.

#### **6.5.3.4 Precondiciones**

El docente se encuentra registrado en el sistema.

#### **6.5.3.5 Postcondiciones**

Se modificó la información del docente.

### **6.5.3.6 Flujo principal**

1. El actor primario ingresa al módulo para realizar cambios de información de docentes.
2. El sistema muestra en pantalla la lista de docentes registrados.
3. El actor primario selecciona un docente.
4. El sistema busca la información del docente seleccionado.
5. El sistema muestra en pantalla la información del docente seleccionado.
6. El actor primario realiza los cambios en la información.
7. El actor primario acepta los cambios.
8. El sistema solicita confirmación por parte del actor primario.
9. El sistema modifica la información del docente.
10. El sistema muestra un mensaje indicando si se desea realizar otra operación.
11. El actor primario sale del módulo.

### **6.5.3.7 Flujos alternativos**

- 8.1 El actor primario cancela los cambios de información.
  - 8.1.1. Regresa al punto 2 del flujo principal.

### **6.5.3.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

## **6.5.4 GD4: Consultar información de docente**

### **6.5.4.1 Descripción**

Caso de uso que permite realizar la consulta de información de un docente.

### **6.5.4.2 Actores**

Primarios: administrador tipo A, administrador tipo B.

### **6.5.4.3 Disparador**

El actor primario solicita realizar consulta de información de un docente.

### **6.5.4.4 Precondiciones**

El docente se encuentra registrado en el sistema.

#### **6.5.4.5 Postcondiciones**

Se consultó la información de un docente.

#### **6.5.4.6 Flujo principal**

1. El actor primario ingresa al módulo realizar consulta de información de un docente.
2. El sistema muestra una lista de los docentes.
3. El actor primario selecciona un registro.
4. El sistema realiza la búsqueda de información del registro seleccionado.
5. El sistema muestra la información del registro en pantalla.
6. El actor primario sale del módulo.

#### **6.5.4.7 Flujos alternativos**

- 3.1. El actor primario cancela la operación.
    - 3.1.1. Regresa al punto 2 del flujo principal.
- De 4 y 5: El actor primario cancela la operación
- Regresa al punto 3 del flujo principal.

#### **6.5.4.8 Requerimientos no funcionales**

Este proceso no debe de tardar más de 2 minutos.

## 7 Tarjetas CRC

Versión	Fecha	Descripción	Autores
Borrador	14/VI/2012	Borrador de inicio	Neria Martínez Francisco Javier
1	3/X/2012	Correcciones en las entidades.	Hernández Báez Antonio Neria Martínez Francisco Javier
2	25/X/2012	Se agregaron entidades faltantes.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 7.1: Historial de revisiones del documento tarjetas CRC

Formalmente las tarjetas CRC (*Class-Responsibility-Collaboration*) no forman parte del lenguaje unificado de modelado<sup>11</sup>, pero son un mecanismo útil para distinguir responsabilidades, atributos y colaboraciones [2].

Una vez definidos los escenarios se procedió a determinar los prototipos de entidades que potencialmente se convertirán en clases de software. Con ayuda de las tarjetas CRC se determinaron las entidades que se listan a continuación:

- Autenticación de usuarios.
- Usuarios.
- Trimestres.
- Alumnos.
- Datos personales.
- Antecedentes académicos.
- Tesis.
- Reportes técnicos.
- Libros.
- Artículos.
- Posgrados.
- Movilidad.
- Planes de estudios.
- Docentes.
- Proyectos de investigación.

Para cada entidad se determinaron atributos, responsabilidades y colaboraciones como se muestra a continuación:

Autenticación	
<b>Responsabilidades</b> Iniciar sesión Cerrar sesión Comprobar sesión Expirar sesión	<b>Colaboraciones</b> Usuarios
<b>Atributos</b> Usuario solicitante Fecha y hora de inicio de sesión IP de la computadora del usuario Navegador web del usuario Tipo de acceso (exitoso o fallo)	

Figura 7.1: Tarjeta CRC Autenticación.

11 Lenguaje unificado de modelado: Lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

Usuarios	
<b>Responsabilidades</b> Realizar alta de un usuario Realizar baja de un usuario Realizar consulta de un usuario Realizar cambios de un usuario	<b>Colaboraciones</b> Autenticación.
<b>Atributos</b> Tipo de usuario Estatus Nombre Apellido Paterno Apellido Materno Correo Electrónico Nombre de Usuario (Login) Contraseña (password) Foto	

Figura 7.2: Tarjeta CRC Usuarios.

Trimestres	
<b>Responsabilidades</b> Realizar alta de trimestres Realizar la baja de trimestres Realizar la consulta de trimestres Realizar cambios de trimestres	<b>Colaboraciones</b> Alumnos
<b>Atributos</b> Trimestre Alias Fecha de inicio Fecha de termino Comentario	

Figura 7.3: Tarjeta CRC Trimestres.

Alumnos	
<b>Responsabilidades</b> Realizar alta de alumnos Realizar la baja de alumnos Realizar la consulta de alumnos Realizar cambios de alumnos	<b>Colaboraciones</b> Docentes Posgrados Artículos Movilidad Datos personales Antecedentes académicos Trimestres
<b>Atributos</b> Estatus Matricula Nombre Apellido paterno Apellido materno Foto	

Figura 7.4: Tarjeta CRC Alumnos.

Datos personales	
<b>Responsabilidades</b> Realizar alta de datos personales Realizar la baja de datos personales Realizar la consulta de datos personales Realizar cambios de datos personales	<b>Colaboraciones</b> Docentes Alumnos
<b>Atributos</b> Dirección Teléfono Correo electrónico institucional Correo electrónico externo Nacionalidad Fecha de nacimiento País de nacimiento Lugar de nacimiento CURP CVU en <i>PDF</i> Comprobante de idioma en <i>PDF</i>	

Figura 7.5: Tarjeta CRC Datos personales.

Antecedentes académicos	
<b>Responsabilidades</b> Realizar alta de antecedentes académicos Realizar la baja de antecedentes académicos Realizar la consulta de antecedentes académicos Realizar cambios de antecedentes académicos	<b>Colaboraciones</b> Docentes Alumnos
<b>Atributos</b> Nombre de la institución Grado País Año de egreso Promedio Institución Titulo en <i>PDF</i>	

Figura 7.6: Tarjeta CRC Antecedentes académicos.

Tesis	
<b>Responsabilidades</b> Realizar alta de tesis Realizar la baja de tesis Realizar la consulta de tesis Realizar cambios de tesis	<b>Colaboraciones</b> Docentes Alumnos
<b>Atributos</b> Titulo Coasesores Fecha Documento <i>PDF</i>	

Figura 7.7: Tarjeta CRC Tesis.

## 7 Tarjetas CRC

Reportes técnicos	
<b>Responsabilidades</b> Realizar alta de reportes técnicos Realizar la baja de reportes técnicos Realizar la consulta de reportes técnicos Realizar cambios de reportes técnicos	<b>Colaboraciones</b> Alumnos
<b>Atributos</b> Título Instancia a la que presenta el reporte Descripción Número de páginas Fecha Objetivo del reporte Coautores Documento <i>PDF</i>	

Figura 7.8: Tarjeta CRC Reportes técnicos.

Libros	
<b>Responsabilidades</b> Realizar alta de libros Realizar la baja de libros Realizar la consulta de libros Realizar cambios de libros	<b>Colaboraciones</b> Alumnos
<b>Atributos</b> ISBN                                  Volumen Título                                      Páginas Editorial                                  Tiraje Edición                                      Idioma Año de publicación                      Palabras clave Documento <i>PDF</i>	

Figura 7.9: Tarjeta CRC Libros.

Artículos	
<b>Responsabilidades</b> Realizar alta de artículos Realizar la baja de artículos Realizar la consulta de artículos Realizar cambios de artículos	<b>Colaboraciones</b> Alumnos
<b>Atributos</b> Título                                      Revista Año de publicación                      Páginas Volumen                                      Coautores Número de volumen                      Posición autor Documento <i>PDF</i>	

Figura 7.10: Tarjeta CRC Artículos.

## 7 Tarjetas CRC

Posgrados	
<b>Responsabilidades</b> Realizar alta de posgrados Realizar la baja de posgrados Realizar la consulta de posgrados Realizar cambios de posgrados	<b>Colaboraciones</b> Planes de estudios Alumnos Docentes
<b>Atributos</b> Estatus Nombre Clave	

Figura 7.11: Tarjeta CRC Posgrados.

Movilidad	
<b>Responsabilidades</b> Realizar alta de movilidad Realizar la baja de movilidad Realizar la consulta de movilidad Realizar cambios de movilidad	<b>Colaboraciones</b> Alumnos
<b>Atributos</b> Institución Dirección Fecha de inicio Fecha de finalización Documento en <i>PDF</i>	

Figura 7.12: Tarjeta CRC Movilidad de alumnos.

Planes de estudios	
<b>Responsabilidades</b> Realizar alta de un plan de estudios Realizar la baja de un plan de estudios Realizar la consulta de un plan de estudios Realizar cambios de un plan de estudios	<b>Colaboraciones</b> Posgrados
<b>Atributos</b> Estatus Versión Documento de aprobación en <i>PDF</i> Documento del plan en <i>PDF</i> Documento del plan en <i>RTF</i>	

Figura 7.13: Tarjeta CRC Planes de estudios.

Docentes	
<b>Responsabilidades</b> Realizar alta de un docente Realizar baja de un docente Realizar consulta de un docente Realizar cambios de un docente	<b>Colaboraciones</b> Alumnos Posgrados Proyectos Datos personales Antecedentes académicos
<b>Atributos</b> Estatus Número económico Nombre Apellido paterno Apellido materno Foto	

Figura 7.14: Tarjeta CRC Docentes.

## 7 Tarjetas CRC

<b>Proyectos de investigación</b>	
<b>Responsabilidades</b> Realizar alta de proyectos de investigación Realizar la baja de proyectos de investigación Realizar la consulta de proyectos de investigación Realizar cambios de proyectos de investigación	<b>Colaboraciones</b> Docentes
<b>Atributos</b> Titulo del proyecto Fecha de inicio Fecha de finalización Alumnos participantes Documento <i>PDF</i>	

Figura 7.15: Tarjeta *CRC* Proyectos de investigación.

## 8 Modelo de dominio

Versión	Fecha	Descripción	Autores
Borrador	14/VI/2012	Borrador de inicio	Neria Martínez Francisco Javier
1	3/X/2012	Se agregó el modelo de dominio del sistema. Se agregó navegabilidad al diagrama. Se corrigió la multiplicidad.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 8.1: Historial de revisiones del documento modelo de dominio

A continuación se elaboró un modelo de clases común para todas las entidades involucradas. Este modelo considera las correspondencias directas entre entidades.

### 8.1 Modelo de dominio general del SIP-CBI

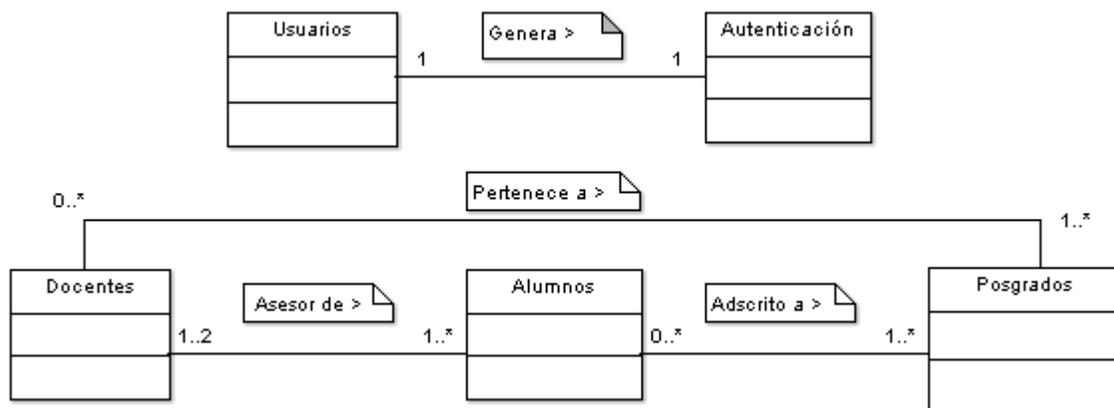


Figura 8.1.1: Entidades primarias del modelo de dominio.

### 8.2 Modelo de dominio para la entidad Docentes

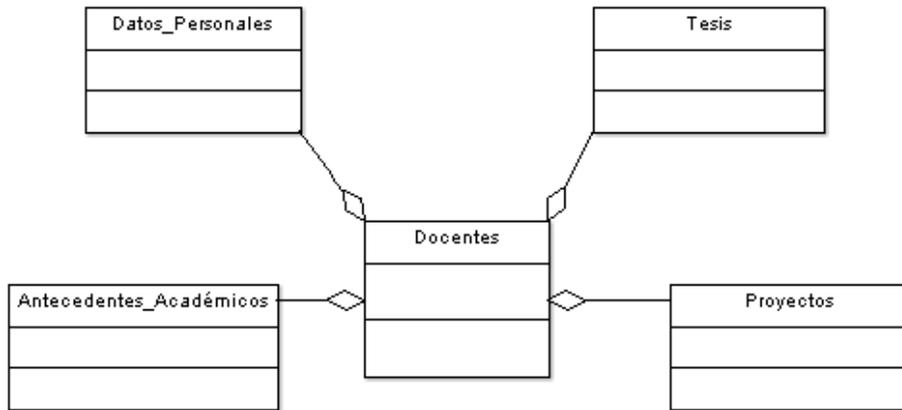


Figura 8.2.1: Modelo de dominio para las entidades secundarias de la entidad docentes.

### 8.3 Modelo de dominio para la entidad Alumnos

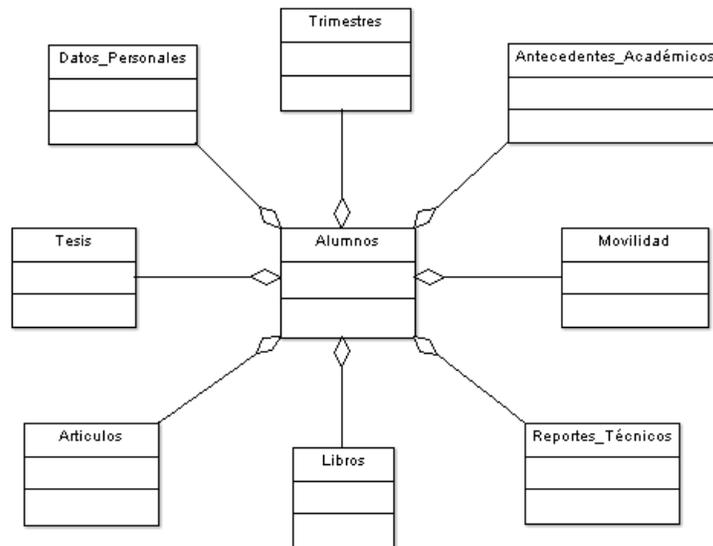


Figura 8.3.1: Modelo de dominio para las entidades secundarias de la entidad alumnos.

### 8.4 Modelo de dominio para la entidad Posgrados

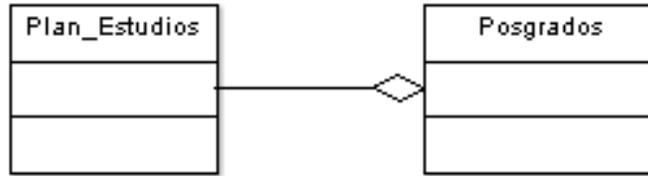


Figura 8.4.1: Modelo de dominio para las entidades secundarias de la entidad posgrados.

## 9 Diagramas de secuencias

Versión	Fecha	Descripción	Autores
Borrador	14/VII/2012	Borrador de inicio	Neria Martínez Francisco Javier
1	1/XI/2012	Se agregaron los diagrama de secuencia.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 9.1: Historial de revisiones del documento diagramas de secuencias

A continuación se elaboraron diagramas de secuencias que muestran la interacción entre los objetos del modelo de análisis. En estos diagramas se representa la comunicación entre objetos a través del tiempo. Los elementos mostrados en estos diagramas provienen de los escenarios de alta, baja, cambios y consultas, y el modelo de dominio. A continuación se muestran los diagramas de secuencias del SIP-CBI.

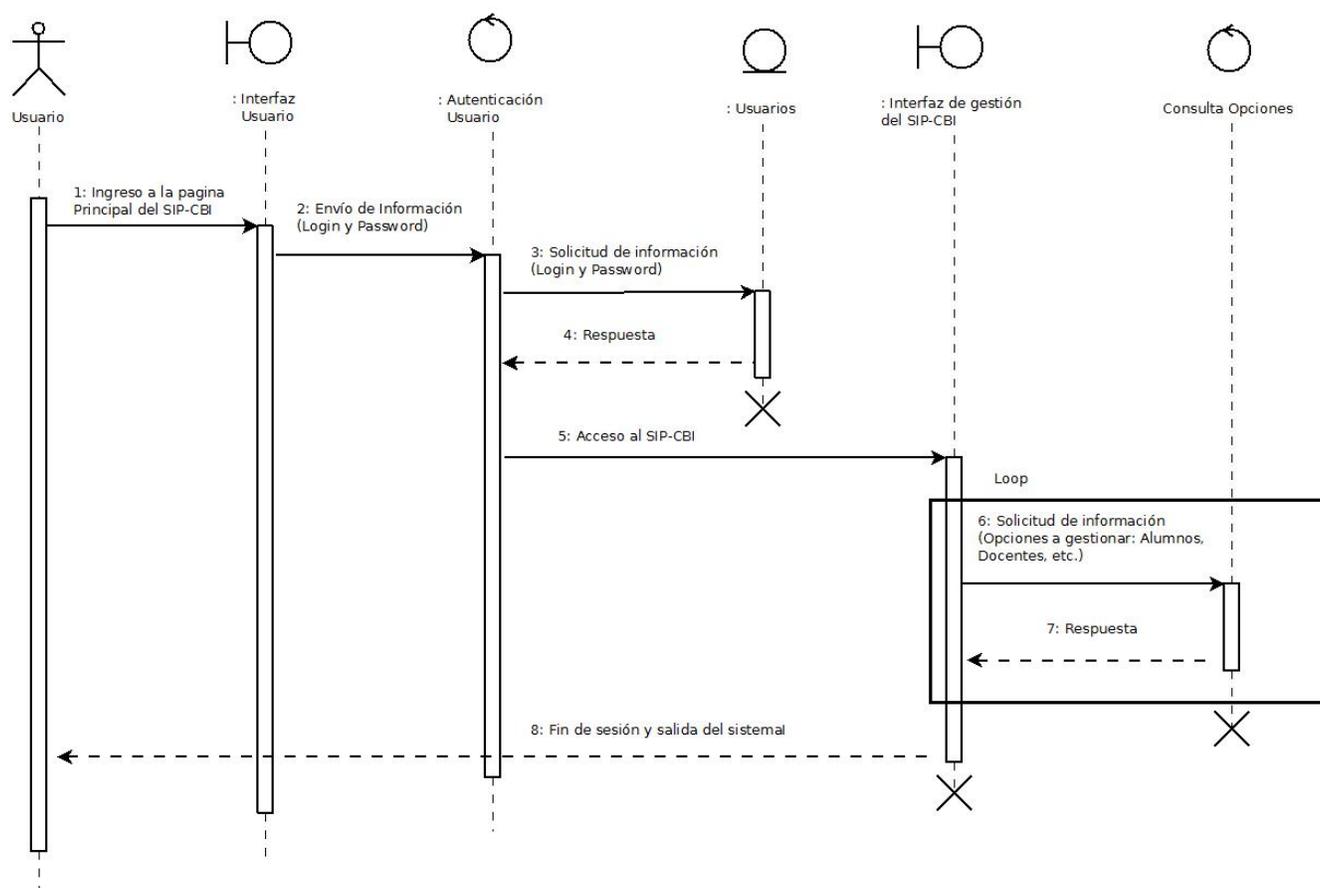


Figura 9.1: Diagrama de secuencias general del SIP-CBI.

## 9 Diagramas de secuencias

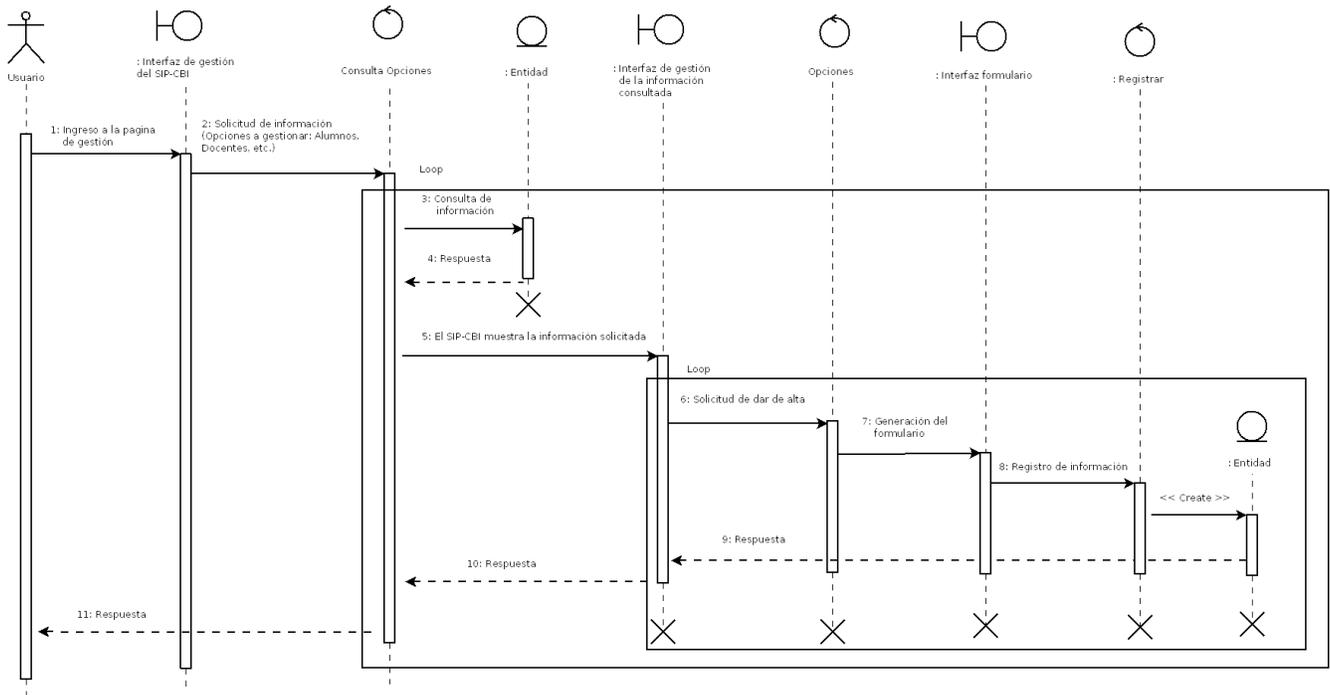


Figura 9.2: Diagrama de secuencias para la operación de alta.

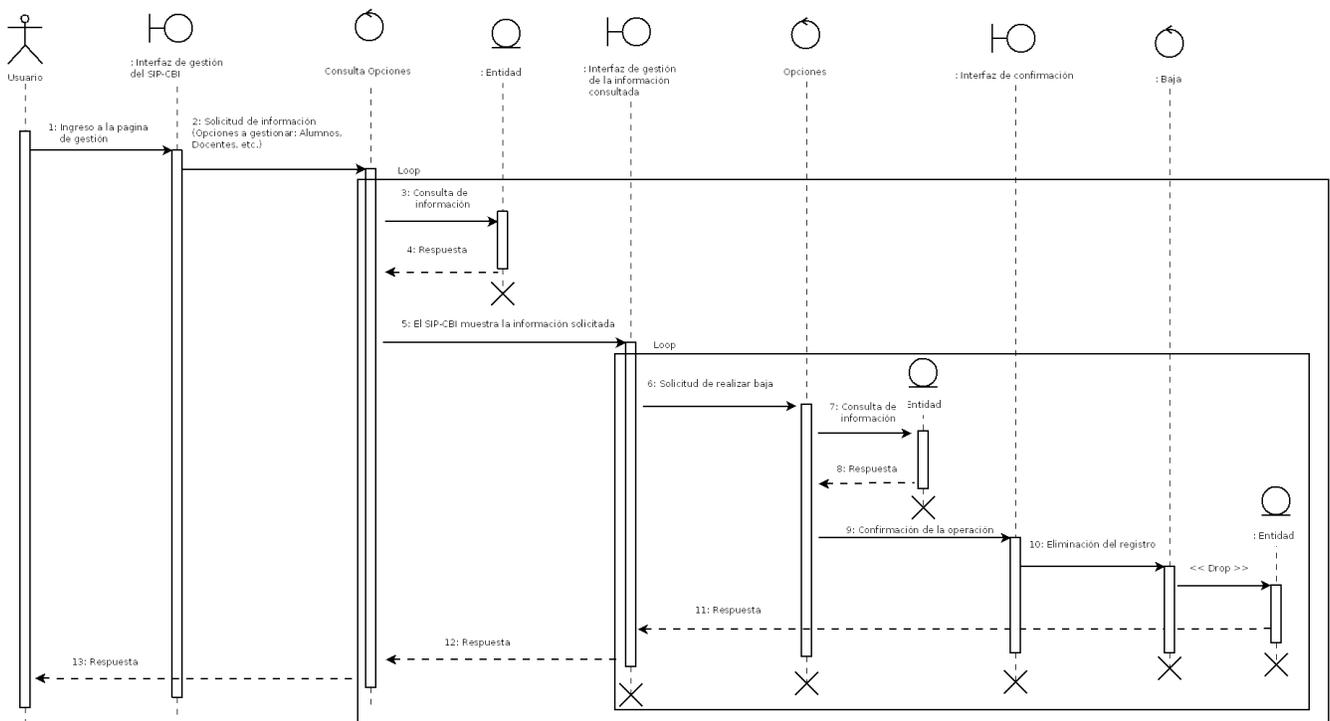


Figura 9.3: Diagrama de secuencias para la operación de baja.

## 9 Diagramas de secuencias

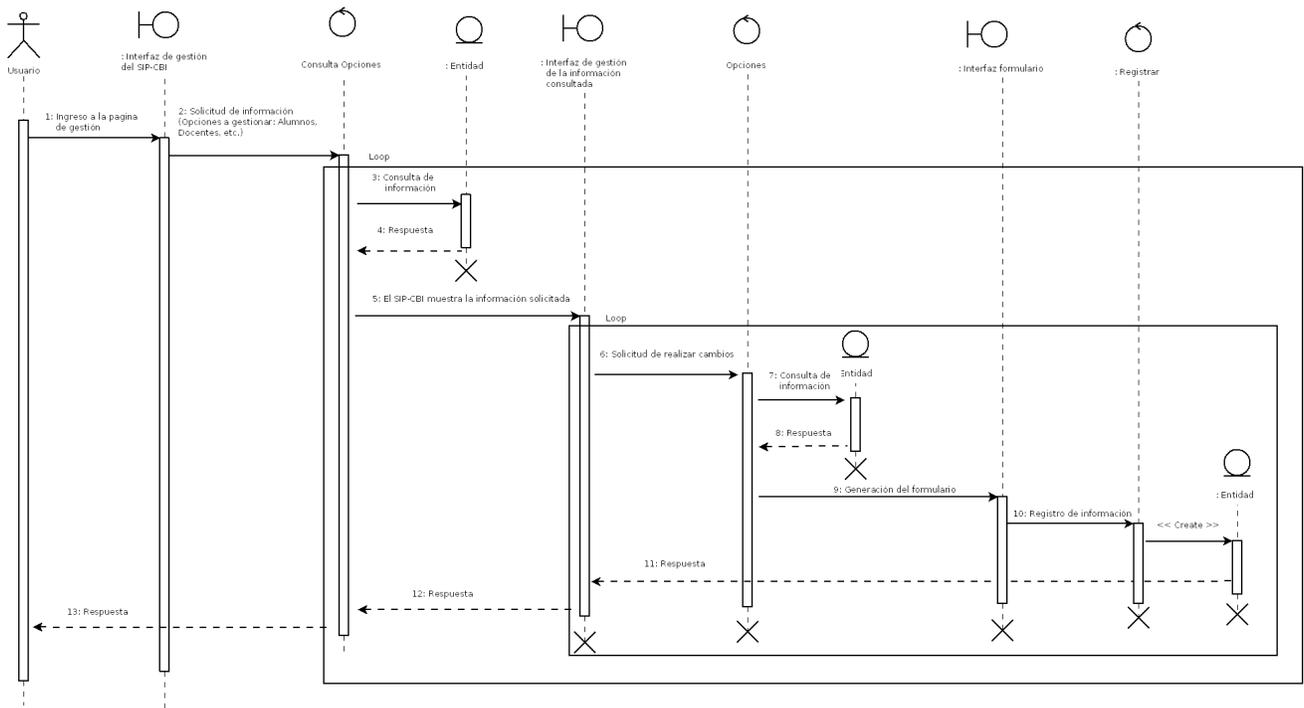


Figura 9.4: Diagrama de secuencias para la operación de cambios.

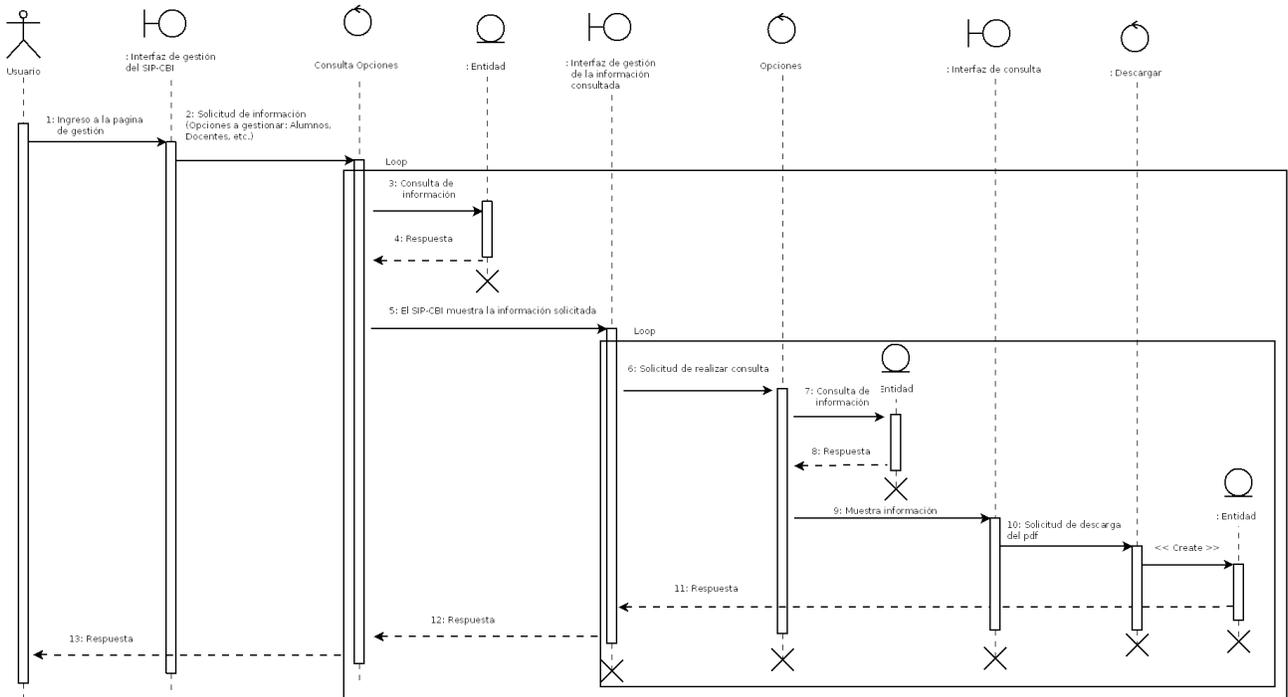


Figura 9.5: Diagrama de secuencias para la operación de consultas.

## 10 Diagrama de clases

Versión	Fecha	Descripción	Autores
Borrador	14/VI/2012	Borrador de inicio	Hernández Báez Antonio Neria Martínez Francisco Javier
1	16/XI/2012	Correcciones en las clases determinadas.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 10.1: Historial de revisiones del documento diagramas de clases de diseño.

Con ayuda del modelo de dominio y los diagramas de secuencias se construyó el diagrama de clases del SIP-CBI. En este diagrama se muestran las clases, relaciones, métodos, atributos, y navegabilidad entre todas las entidades del sistema. Las clases definidas para el SIP-CBI son: *Autenticacion*, *Usuarios*, *Trimestres*, *Alumnos*, *DatosPersonales*, *AntecedentesAcad*, *Tesis*, *Libros*, *Articulos*, *Posgrados*, *Movilidad*, *PlanEstudios*, *Docentes* y *Proyectos*.

A continuación se describe cada una de las clases que forman el SIP-CBI:

- **Autenticacion:** Es una clase principal o primaria del SIP-CBI. Permite crear o finalizar una sesión para autenticarse dentro del SIP-CBI. Esta formada por los atributos: *usuario*, *fecha*, *ip*, *browser*, *acceso*.
- **Usuarios:** Es una clase principal o primaria del SIP-CBI. Permite realizar operaciones de alta, baja, cambios y consultas de usuarios en la base de datos. Esta formada por los atributos: *estatus*, *nombre*, *apellidop*, *apellidom*, *correoe*, *login*, *password* y *foto*.
- **Trimestres:** Clase secundaria que gestiona los trimestres. Tiene una relación de agregación con la clase *Alumnos*. Esta formada por los atributos: *trimestre*, *alias*, *fechainicio*, *fechafin* y *comentario*.
- **Alumnos:** Es una clase principal o primaria del SIP-CBI. Nos permite realizar operaciones de alta, baja, cambios y consultas de alumnos en la base de datos. Contiene los atributos: *matricula*, *trimestreingreso*, *nombre*, *apellidop*, *apellidom* y *foto*. *Alumnos* contiene a las clases *DatosPersonales*, *Trimestres*, *AntecedentesAcad*, *Movilidad*, *ReporteTecnico*, *Libros*, *Articulos* y *Tesis* mediante relaciones de agregación.
- **DatosPersonales:** Esta clase permite realizar operaciones de alta, baja, cambios y consultas en los datos personales de los alumnos que serán almacenados en la base de datos. Tiene una relación de agregación con las clases *Alumnos* y *Docentes*. Los atributos asociados esta clase son: *direccion*, *telefono*, *celular*, *correoinst*, *correooext*, *nacionalidad*, *fechanacimiento*, *paisnacimiento*, *lugarnacimiento*, *curp*, *cvupdf* e *idiomapdf*.
- **AntecedentesAcad:** Permite las operaciones de alta, baja, cambios y consultas en los antecedentes académicos de los alumnos. Tiene una relación de agregación con las clases *Alumnos* y *Docentes*. Los atributos que la forman son: *grado*, *pais*, *AAegreso*, *promedio*, *institucion* y *titulopdf*.
- **Tesis:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de tesis del alumno. Tiene una relación de agregación con la clase *Alumnos* y *Docentes*. Contiene a los atributos: *titulo*, *coasesores*, *fecha* y *tesispdf*.

- **ReporteTecnico:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de reportes técnicos del alumno. Tiene una relación de agregación con la clase *Alumnos*. Esta formada por los atributos: *titulo*, *instanciapresentarreporte*, *descripcion*, *numeropaginas*, *fecha*, *objetivosreporte*, *coautores* y *reportepdf*.
- **Libros:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de libros del alumno. Tiene una relación de agregación con la clase *Alumnos*. Esta formada por los atributos: *ISBN*, *titulo*, *editorial*, *edicion*, *AApublicacion*, *volumen*, *paginas*, *tiraje*, *idioma*, *palabrasclave* y *libropdf*.
- **Artículos:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de artículos del alumno. Tiene una relación de agregación con la clase *Alumnos*. Contiene los atributos: *tipoparticulos*, *titulo*, *AApublicacion*, *volumen*, *numerovolumen*, *revistapublicacion*, *delapag*, *alapag*, *coautores*, *posicionautor* y *articulopdf*.
- **Posgrados:** Clase primaria que permite realizar operaciones de alta, baja, cambios y consultas en la información de posgrados. Tiene una relación de agregación con la clase *PlanEstudios*. Esta formada por los atributos: *estatus*, *nombre* y *clave*.
- **Movilidad:** Esta clase permite realizar operaciones del alta, baja, cambios y consultas en la información de movilidad de un alumno. Tiene una relación de agregación con la clase *Alumno*. Esta formada por los atributos: *institucion*, *direccion*, *fechainicio*, *fechafin*, y *movilidadpdf*.
- **PlanEstudios:** Es una clase principal o primaria del SIP-CBI. Permite gestionar la información básica de un plan de estudios en la base de datos. Tiene una relación de agregación con la clase *Posgrados*. Esta formada por los atributos: *estatus*, *version*, *aprobacionpdf*, *planpdf* y *planrtf*.
- **Docentes:** Es una clase principal o primaria del SIP-CBI. Permite la gestión de la información de los docentes en la base de datos. Tiene una relación de agregación con las clases *DatosPersonales*, *Tesis*, *AntecedentesAcad* y *Proyectos*. Esta formada por los atributos: *estatus*, *eco*, *nombre*, *apellidop*, *apellidom* y *foto*.
- **Proyectos:** Clase que gestionar los proyectos de investigación de un docente. Tiene una relación de agregación con la clase *Docentes*. Esta formada por los atributos: *titulo*, *fechainicio*, *fechafin*, *alumnosparticipantes* y *proyectopdf*.

En la Figura 10.1 podemos observar las relaciones entre todas las clases que forman el SIP-CBI.

## 10 Diagrama de clases

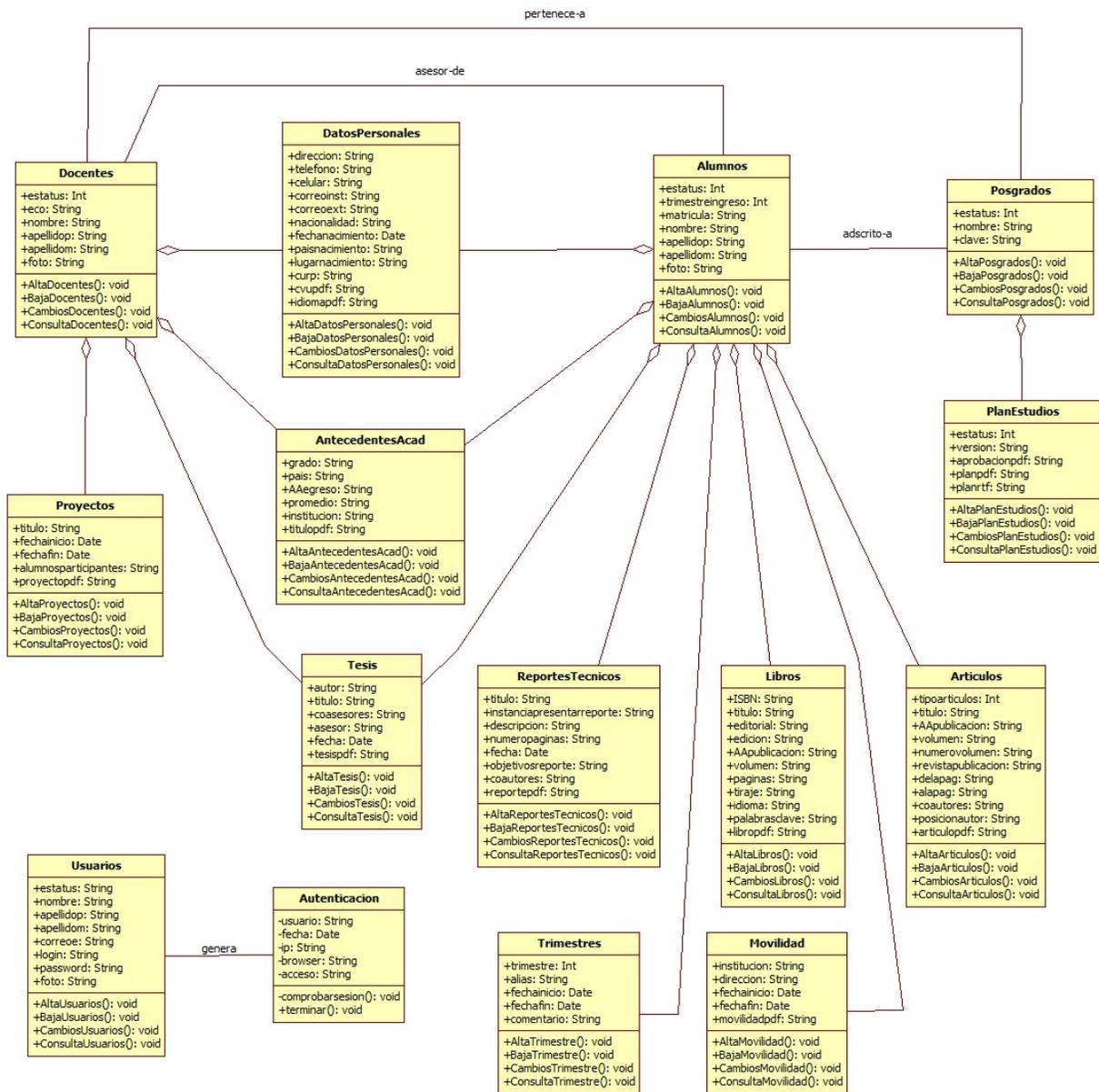


Figura 10.1: Diagrama de clases del SIP-CBI.

# 11 Esquema de la base de datos

Versión	Fecha	Descripción	Autores
Borrador	3/XI/2012	Borrador de inicio	Hernández Báez Antonio Neria Martínez Francisco Javier
1	19/XI/2012	Corrección en atributos y tipos de datos.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 11.1: Historial de revisiones del documento del esquema de la base de datos.

Para construir el modelo de la base de datos a partir del modelo de dominio y el diagrama de clases se aplicó la siguiente estrategia:

1. Convertir cada entidad del modelo de dominio en una tabla.
2. Convertir los atributos de las clases en “renglones” de la tabla.
3. Especificar llaves primarias e índices en cada tabla.
4. Relacionar las tablas de manera adecuada para evitar relaciones del tipo M:N. En las relaciones 1:N la relación se representa con una llave foránea que se inserta en la tabla que tiene la cardinalidad N. En caso de tener cardinalidades múltiples en ambos lados, M:N, debe introducirse una tabla adicional que contenga las llaves primarias de ambas entidades.

A continuación se describe cada una de la entidades de bases de datos que forman el SIP-CBI:

- **autenticacion:** Tabla en la cual se registra si la autenticación de usuarios es exitosa o no. Esta formada por los campos: *usuario*, *fecha*, *ip*, *browser*, *acceso*.
- **usuarios:** Tabla que almacena los datos de los usuarios administradores del SIP-CBI. Esta formada por los campos: *id\_usuarios*, *id\_tipousuarios\_usuarios*, *id\_estatus\_usuarios*, *nombre\_usuarios*, *apellidop\_usuarios*, *apellidom\_usuarios*, *correoe\_usuarios*, *login\_usuarios*, *password\_usuarios* y *foto\_usuarios*. Los campos *id\_tipousuarios\_usuarios*, *id\_estatus\_usuarios* son llaves foráneas de los *id* de las tablas *tipousuarios* y *estatus*, respectivamente.
- **tipousuarios:** Tabla que permite asignar varios tipos de usuarios. Esta formada por los campos: *id\_tipousuarios*, *tipo\_tipousuarios*, *clave\_tipousuarios* y *nombre\_tipousuarios*.
- **trimestres:** Tabla que almacena la información de los trimestres. Esta formada por los campos: *id\_trimestres*, *trimestre\_trimestres*, *alias\_trimestres*, *fechainicio\_trimestres*, *fechafin\_trimestres* y *comentario\_trimestres*.
- **trimestresalumnos:** Contiene la información de la actividad trimestral del alumno. La tabla esta formada por los campos: *id\_trimestresalumnos*, *id\_trimestres\_trimestresalumnos*, *id\_alumnos\_trimestresalumnos* e *id\_estatus\_trimestresalumnos*. Los campos *id\_trimestres\_trimestresalumnos*, *id\_alumnos\_trimestresalumnos* e *id\_estatus\_trimestresalumnos* son llaves foráneas que direccionan hacia el *id* de las tablas *trimestres*, *alumnos* y *estatus*.
- **alumnos:** Permite almacenar la información de los alumnos. Contiene los campos: *id\_alumnos*,

*matricula\_alumnos*, *id\_trimestreingreso\_alumnos*, *nombre\_alumnos*, *apellidop\_alumnos*, *apellidom\_alumnos* y *foto\_alumnos*. El campo *id\_trimestreingreso\_alumnos* es una llave foránea que apunta hacia el campo *id\_trimestres* de la tabla *trimestres*.

- **alumnoadscrito:** Almacena la información que relaciona a un alumno con un posgrado. La tabla esta formada por los campos: *id\_alumnoadscrito*, *id\_alumnos\_alumnoadscrito* e *id\_posgrados\_alumnoadscrito*. Los campos *id\_alumnos\_alumnoadscrito* e *id\_posgrados\_alumnoadscrito* son llaves foráneas de los *id* de las tablas *alumnos* y *posgrados*.
- **alumnoplanestudios:** Contiene la información que relaciona a un alumno con un plan de estudios de algún posgrado. Esta tabla se forma con los campos: *id\_alumnoplanestudios*, *id\_alumnoadscrito\_alumnoplanestudios* e *id\_planestudios\_alumnoplanestudios*. Los campos *id\_alumnoadscrito\_alumnoplanestudios* e *id\_planestudios\_alumnoplanestudios* funcionan como llaves foráneas que direccionan al *id* de las tablas *alumnoadscrito* y *planestudios*.
- **datospersonales:** Esta tabla almacena datos personales adicionales de alumnos y docentes. Los campos asociados esta tabla son: *id\_datospersonales*, *direccion\_datospersonales*, *telefono\_datospersonales*, *celular\_datospersonales*, *correoinst\_datospersonales*, *correoext\_datospersonales*, *nacionalidad\_datospersonales*, *fechanacimiento\_datospersonales*, *paisnacimiento\_datospersonales*, *lugarnacimiento\_datospersonales*, *curp\_datospersonales*, *cvpdf\_datospersonales* e *idiomapdf\_datospersonales*.
- **alumnospersonales:** Esta tabla permite relacionar la tabla *alumnos* con la tabla *datospersonales*. Esta formada por los campos: *id\_alumnospersonales*, *id\_alumnos\_alumnospersonales* e *id\_datospersonales\_alumnospersonales*. Los campos *id\_alumnos\_alumnospersonales* e *id\_datospersonales\_alumnospersonales* son llaves foráneas provenientes de las *id* de las tablas *alumnos* y *datospersonales*, respectivamente.
- **docentespersonales:** Relaciona la tabla *docentes* con la tabla *datospersonales*. Esta formada por los campos: *id\_docentespersonales*, *id\_docentes\_docentespersonales* e *id\_datospersonales\_docentespersonales*. Los campos *id\_docentes\_docentespersonales* e *id\_datospersonales\_docentespersonales* son llaves foráneas provenientes de las *id* de las tablas *docentes* y *datospersonales*, respectivamente.
- **antecedentesacad:** Tabla que almacena los antecedentes académicos de *alumnos* y *docentes*. Esta formada por los campos: *id\_antecedentesacad*, *grado\_antecedentesacad*, *pais\_antecedentesacad*, *AAegreso\_antecedentesacad*, *promedio\_antecedentesacad*, *institucion\_antecedentesacad* y *titulopdf\_antecedentesacad*.
- **alumnosantecedentes:** Relaciona las tablas *alumnos* y *antecedentesacad*. Contiene los campos: *id\_alumnosantecedentes*, *id\_alumnos\_alumnosantecedentes* e *id\_antecedentesacad\_alumnosantecedentes*. Los campos *id\_alumnos\_alumnosantecedentes* e *id\_antecedentesacad\_alumnosantecedentes* son llaves foráneas provenientes de las *id* de las tablas *alumnos* y *antecedentesacad*, respectivamente.
- **docentesantecedentes:** Relaciona las tablas *docentes* y *antecedentesacad*. Contiene los campos: *id\_docentes\_antecedentes*, *id\_docentes\_docentes\_antecedentes* e *id\_antecedentesacad\_docentes\_antecedentes*. Los campos *id\_docentes\_docentes\_antecedentes* e *id\_antecedentesacad\_docentes\_antecedentes* son llaves foráneas provenientes de las *id* de las tablas *docentes* y *antecedentesacad*, respectivamente.

*antecedentes* son llaves foráneas de las *id* de las tablas *docentes* y *antecedentesacad*, respectivamente.

- **tesis:** Almacena información relacionada con las tesis de alumnos. Contiene los campos: *id\_tesis*, *id\_estatus\_tesis*, *id\_alumnos\_tesis*, *id\_docentes\_tesis*, *titulo\_tesis*, *coasesores\_tesis*, *fecha\_tesis* y *tesispdf\_tesis*. Los campos *id\_estatus\_tesis*, *id\_alumnos\_tesis* e *id\_docentes\_tesis* son llaves foráneas del *id* las tablas *estatus*, *alumnos* y *docentes*, respectivamente.
- **reportetecnico:** Tabla que almacena información sobre los reportes técnicos del alumno. Esta formada por los campos: *id\_reportetecnico*, *id\_alumnos\_reportetecnico*, *titulo\_reportetecnico*, *instanciapresentarreporte\_reportetecnico*, *descripcion\_reportetecnico*, *numeropaginas\_reportetecnico*, *fecha\_reportetecnico*, *objetivosreporte\_reportetecnico*, *coasesores\_reportetecnico* y *reportepdf\_reportetecnico*. El campo *id\_alumnos\_reportetecnico* es una llave foránea que apunta hacia el *id* de la tabla *alumnos*.
- **libros:** Almacena la información de los libros del alumno. Esta formada por los campos: *id\_libros*, *id\_alumnos\_libros*, *ISBN\_libros*, *titulo\_libros*, *editorial\_libros*, *edicion\_libros*, *AApublicacion\_libros*, *volumen\_libros*, *paginas\_libros*, *tiraje\_libros*, *idioma\_libros*, *palabrasclave\_libros* y *libropdf\_libros*. El campo *id\_alumnos\_libros* es una llave foránea que apunta hacia el *id* de la tabla *alumnos*.
- **articulos:** Clase que permite realizar operaciones de alta, baja, cambios y consultas en la información de artículos del alumno. Tiene una relación de agregación con la clase *Alumnos*. Contiene los atributos: *id\_articulos*, *id\_tipoarticulos\_articulos*, *id\_alumnos\_articulos*, *titulo\_articulos*, *AApublicacion\_articulos*, *volumen\_articulos*, *numerovolumen\_articulos*, *revistapublicacion\_articulos*, *delapag\_articulos*, *alapag\_articulos*, *coautores\_articulos*, *posicionautor\_articulos* y *articulopdf\_articulos*. Los campos *id\_articulos* e *id\_tipoarticulos\_articulos* son llaves foráneas de los *id* de las tablas *alumnos* y *tipoarticulos*, respectivamente.
- **tipoarticulos:** Tabla que permite asignar un tipo de artículo. Esta formada por los campos *id\_tipoarticulo*, *tipo\_tipoarticulo*, *clave\_tipoarticulo* y *nombre\_tipoarticulo*.
- **posgrados:** Tabla que contiene el nombre y clave de los posgrados. Esta formada por los campos: *id\_posgrados*, *id\_estatus\_posgrados*, *nombre\_posgrados* y *clave\_posgrados*. El campo *id\_estatus\_posgrados* es una llave foránea que apunta hacia el *id* de la tabla *estatus*.
- **movilidad:** Contiene la información sobre las movilidades del alumno. Esta formada por los campos: *id\_movilidad*, *id\_alumnos\_movilidad*, *institucion\_movilidad*, *direccion\_movilidad*, *fechainicio\_movilidad*, *fechafin\_movilidad*, y *movilidadpdf\_movilidad*. El campo *id\_alumnos\_movilidad* funciona como llave foránea hacia el *id* de la tabla *estatus*.
- **planestudios:** Esta tabla contiene la información básica de un plan de estudios. Esta formada por los campos: *id\_planestudios*, *id\_posgrados\_planestudios*, *id\_estatus\_planestudios*, *version\_planestudios*, *aprobacionpdf\_planestudios*, *planpdf\_planestudios* y *planrtf\_planestudios*. Los campos *id\_posgrados\_planestudios* e *id\_estatus\_planestudios* son llaves foráneas que apuntan hacia los *id* de las tablas *posgrados* y *estatus*, respectivamente.

## 11 Esquema de la base de datos

- **docentes:** Contiene la información básica de los docentes. Esta formada por los campos: *id\_docentes*, *id\_estatus\_docentes*, *eco\_docentes*, *nombre\_docentes*, *apellidop\_docentes*, *apellidom\_docentes* y *foto\_docentes*. El campo *id\_estatus\_docentes* funciona como llave foránea que apunta hacia el *id* de la tabla *estatus*.
- **nucleobasico:** esta tabla contiene la relación entre docentes y posgrados. Esta formada por los campos: *id\_nucleobasico*, *id\_posgrados* e *id\_docentes*. Los campos *id\_posgrados* e *id\_docentes* son llaves foráneas que apuntan hacia el *id* de las tablas *posgrados* y *docentes*, respectivamente.
- **proyectos:** La tabla contiene la información de los proyectos de investigación de un docente. Esta formada por los campos: *id\_proyectos*, *id\_docentes\_proyectos*, *titulo\_proyectos*, *fechainicio\_proyectos*, *fechafin\_proyectos*, *alumnosparticipantes\_proyectos* y *proyectopdf\_proyectos*. El campo *id\_docentes\_proyectos* funciona como llave foránea y apunta hacia el *id* de la tabla *docentes*.
- **estatus:** Esta tabla contiene los estatus del sistema. Esta formada por los campos: *id\_estatus*, *clave\_estatus*, *tipo\_estatus* y *nombre\_estatus*.

En todas las entidades los *id* son llaves primarias (*primary key*), no nulas (*not null*), únicas (*unique index*) y autoincrementables (*autoincremental*).

La Figura 11.1 muestra el modelo de base de datos obtenido a partir del modelo de clases. En esta figura se incluye el tipo de datos para cada campo y la cardinalidad.



## 12 Arquitectura

Versión	Fecha	Descripción	Autores
Borrador	3/XI/2012	Borrador de inicio	Hernández Báez Antonio Neria Martínez Francisco Javier
1	10/II/2013	Se agregó información sobre el patrón MVC.	Hernández Báez Antonio Neria Martínez Francisco Javier

Tabla 12.1: Historial de revisiones del documento del arquitectura.

La implementación del SIP-CBI se llevará a cabo mediante el uso del patrón MVC. El MVC es un patrón de arquitectura de software que permite separar la lógica de negocio de la interfaz gráfica de usuario. Agiliza la funcionalidad, mantenibilidad y escalabilidad de un sistema.

El MVC se divide en tres niveles de abstracción:

- **Modelo:** representa la lógica del negocio. Actúa como un intermediario para acceder de forma directa a la información almacenada en la base de datos.
- **Vista:** elemento encargado de mostrar la información al usuario de forma gráfica.
- **Controlador:** intermediario entre la vista y el modelo. Gestiona las solicitudes del usuario hacia el modelo y regresa datos a la vista para presentarlos al usuario.

El funcionamiento del patrón MVC puede describirse de la siguiente forma:

- El usuario realiza una petición.
- El controlador recibe esta petición.
- El controlador realiza una llamada al modelo.
- El modelo interactúa con la base de datos directamente o mediante servicios web y regresa datos al controlador.
- El controlador recibe los datos y los envía a la vista.
- La vista recibe los datos, los procesa y crea una capa de abstracción para el proceso de datos y otra para generar la interfaz gráfica de usuario.
- La vista mostrará los datos procesados al usuario.

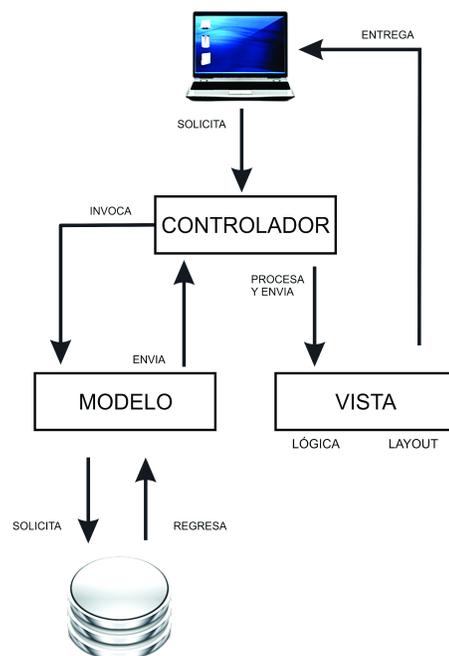


Figura 12.1: Funcionamiento del patrón MVC.

En la Figura 12.1 se muestra el flujo de datos a través del patrón MVC.

## 13.1 Formato de control de medios de verificación del PNPC

RECTORÍA GENERAL APOYO INSTITUCIONAL AL POSGRADO CONTROL DE MEDIOS DE VERIFICACIÓN PNPC			
NOMBRE DEL PROGRAMA	UNIDAD	CONTENIDO	REFERENCIA
			ORIENTACIÓN
No	CARPETA		1 2 3
<b>1 ESTRUCTURA DEL PROGRAMA</b>			
<b>1. Plan de estudios</b>			
		1.1 Plan de estudios completo y sellado	<input type="checkbox"/>
		1.2 Justificación del programa	<input type="checkbox"/>
		1.3 Objetivos y metas	<input type="checkbox"/>
		1.4 Perfil de ingreso	<input type="checkbox"/>
		1.5 Perfil de egreso	<input type="checkbox"/>
		1.6 Congruencia del plan de estudios	<input type="checkbox"/>
		1.7 Mapa curricular	<input type="checkbox"/>
		1.8 Actualización del plan de estudios	<input type="checkbox"/>
		1.9 Opciones de graduación	<input type="checkbox"/>
		1.10 Idioma	<input type="checkbox"/>
<b>2. Proceso de enseñanza-aprendizaje</b>			
		2.1 Relación de medios de evaluación utilizados	<input type="checkbox"/>
		2.2 Evaluación de los instrumentos de evaluación	<input type="checkbox"/>
		2.3 Participación de estudiantes en eventos académicos internos	<input type="checkbox"/>
<b>2 ESTUDIANTES</b>			
<b>3. Ingreso de estudiantes</b>			
		3.1 Selección de estudiantes (requisitos de ingreso)	<input type="checkbox"/>
<b>4. Trayectoria escolar</b>			
		4.1 Estadísticas de la Trayectoria Escolar	<input type="checkbox"/>
		4.2 Opción y mecanismos para la obtención de grado	<input type="checkbox"/>
		4.3 Actas de examen de grado	<input type="checkbox"/>
<b>5. Movilidad e intercambio de estudiantes</b>			
		5.1 Resultados del programa de movilidad (nacional y en el extranjero)	<input type="checkbox"/>
		5.2 Ejemplos de los mecanismos usados para fomentarla	<input type="checkbox"/>
		5.3 Proyectos terminales o tesis codirigidas	<input type="checkbox"/>
<b>6. Tutorías y asesorías</b>			
		6.1 Relación de estudiantes y tutores	<input type="checkbox"/>
		6.2 Relación de estudiantes por director de tesis	<input type="checkbox"/>
<b>7. Dedicación exclusiva</b>			
		7.1 Relación de estudiantes de TP vs TC	<input type="checkbox"/>
<b>3 PERSONAL ACADÉMICO</b>			
<b>8. Núcleo académico básico</b>			
		8.1 Perfil del núcleo académico	<input type="checkbox"/>
		8.2 Tiempos de dedicación	<input type="checkbox"/>
		8.3 Distinciones académicas	<input type="checkbox"/>
		8.4 Organización académica	<input type="checkbox"/>
		8.5 Programa de supervisión	<input type="checkbox"/>
		8.6 Evaluación académica	<input type="checkbox"/>
<b>8 PLAN DE MEJORAS Y ÚLTIMA EVALUACIÓN (reportes electrónicos)</b>			
		Estructura del programa + FODA	<input type="checkbox"/>
		Estudiantes + FODA	<input type="checkbox"/>
		Personal Académico + FODA	<input type="checkbox"/>
		Infraestructura y Servicios + FODA	<input type="checkbox"/>
		Resultados + FODA	<input type="checkbox"/>
		Cooperación con otros actores de la sociedad + FODA	<input type="checkbox"/>
		Mejor FODA General	<input type="checkbox"/>
		Estructura del programa	<input type="checkbox"/>
		Estudiantes	<input type="checkbox"/>
		Personal Académico	<input type="checkbox"/>
		Infraestructura y Servicios	<input type="checkbox"/>
		Resultados	<input type="checkbox"/>
		Cooperación con otros actores de la sociedad	<input type="checkbox"/>
		Acciones basadas en las recomendaciones	<input type="checkbox"/>
		20.2 Plan de Mejora Formato Libre de texto (máx. 10 cuartillas)	<input type="checkbox"/>
<b>9 INFORMACIÓN ESTADÍSTICA DEL PROGRAMA (REPORTES GENERADOS POR LA APLICACIÓN)</b>			
		Datos del programa Reporte de Datos Generales	<input type="checkbox"/>
		Última evaluación Reporte de última evaluación	<input type="checkbox"/>
		Plan de estudios Reporte de plan de estudios	<input type="checkbox"/>
		Personal Académico Reporte de académicos	<input type="checkbox"/>
		Reporte de LGAC	<input type="checkbox"/>
		Reporte de movilidad	<input type="checkbox"/>
		Estudiantes Reporte de estudiantes	<input type="checkbox"/>
		Reporte de movilidad	<input type="checkbox"/>
		Reporte de tutorías	<input type="checkbox"/>
		Reporte de LGAC de estudiantes	<input type="checkbox"/>
		Vinculos Reporte de vinculación	<input type="checkbox"/>
		Productividad académica Reportes de productividad académica	<input type="checkbox"/>
		Distinciones y premios	<input type="checkbox"/>
		Experiencia Profesional	<input type="checkbox"/>
		Grados Académicos	<input type="checkbox"/>
<b>10 PÁGINA WEB</b>			
		Reser un documento breve donde se describa el contenido de la página WEB del programa.	<input type="checkbox"/>
		a) Perfil de ingreso	<input type="checkbox"/>
		b) Objetivos generales y particulares del programa posgrado.	<input type="checkbox"/>
<b>9. Líneas de generación y/o aplicación del conocimiento</b>			
		9.1 Congruencia de las LGAC y aplicación del conocimiento	<input type="checkbox"/>
		9.2 Participación de los estudiantes en proyectos de investigación	<input type="checkbox"/>
<b>4 INFRAESTRUCTURA Y SERVICIOS</b>			
<b>10. Espacios y equipamiento</b>			
		10.1 Disponibilidad de aulas (fotos y descripción)	<input type="checkbox"/>
		10.2 Espacios para profesores y estudiantes	<input type="checkbox"/>
<b>11. Laboratorios y talleres</b>			
		11.1 Espacios, equipos y servicios	<input type="checkbox"/>
		11.2 Materiales y suministros	<input type="checkbox"/>
		11.3 Programación y utilización	<input type="checkbox"/>
<b>12. Información y documentación</b>			
		12.1 Biblioteca e instalaciones	<input type="checkbox"/>
		12.2 Acercos y servicios	<input type="checkbox"/>
<b>13. Tecnologías de información y comunicación</b>			
		13.1 Equipo e instalaciones	<input type="checkbox"/>
		13.2 Redes	<input type="checkbox"/>
		13.3 Atención y servicios	<input type="checkbox"/>
<b>5 RESULTADOS</b>			
<b>14. Trascendencia, cobertura y evolución del programa</b>			
		14.1 Alcance y tendencia de los resultados del programa	<input type="checkbox"/>
		14.2 Cobertura del programa	<input type="checkbox"/>
		14.3 Evolución y pertinencia del programa	<input type="checkbox"/>
<b>15. Pertinencia del programa (Seguimiento de egresados)</b>			
		15.1 Satisfacción de los egresados	<input type="checkbox"/>
		15.2 Proyección del programa	<input type="checkbox"/>
<b>16. Efectividad del posgrado</b>			
		16.1 Análisis de la eficiencia terminal	<input type="checkbox"/>
		16.2 Estrategias para incrementar graduación	<input type="checkbox"/>
<b>17. Contribución al conocimiento</b>			
		17.1 Investigación y Desarrollo (Listado de productividad)	<input type="checkbox"/>
		17.2 Listado de profesores en las LGAC	<input type="checkbox"/>
		17.3 Dirección de tesis (participación de estudiantes)	<input type="checkbox"/>
		17.4 Publicación de resultados de tesis de doctorado	<input type="checkbox"/>
		17.5 Participación de estudiantes y profesores en encuentros académicos	<input type="checkbox"/>
		17.6 Reorientación de la investigación y del trabajo profesional	<input type="checkbox"/>
<b>6 COOPERACIÓN CON OTROS ACTORES DE LA SOCIEDAD</b>			
<b>18. Vinculación</b>			
		18.1 Descripción de mecanismos de vinculación	<input type="checkbox"/>
		18.2 Copia de convenios o contratos vigentes	<input type="checkbox"/>
		18.3 Estudios o informes relacionados con la vinculación	<input type="checkbox"/>
<b>19. Financiamiento</b>			
		19.1 Recursos aplicados	<input type="checkbox"/>
		19.2 Ingresos extraordinarios	<input type="checkbox"/>
<b>7 PLANEACIÓN INSTITUCIONAL (REPORTES GENERADOS POR LA APLICACIÓN)</b>			
		1. Planeación institucional del posgrado	<input type="checkbox"/>
		2. Autoevaluación del programa de posgrado	<input type="checkbox"/>
		c) Estructura del plan de estudios	<input type="checkbox"/>
		d) Número de alumnos matriculados por cohorte generacional.	<input type="checkbox"/>
		e) Núcleo académico básico (desable con una breve reseña curricular de los participantes).	<input type="checkbox"/>
		f) Líneas generación y/o aplicación del conocimiento del programa.	<input type="checkbox"/>
		g) Tutorías	<input type="checkbox"/>
		h) Productividad académica relevante del programa de posgrado.	<input type="checkbox"/>
		i) Vinculación con otros sectores de la sociedad.	<input type="checkbox"/>
		j) Procesos administrativos	<input type="checkbox"/>
		k) En los casos de programas con participación de varias universidades, la información deberá figurar en la pag. De cada una de ellas	<input type="checkbox"/>

Figura 13.1.1: Formato de control de medios de verificación PNPC.

## 13.2 Claves y nomenclatura de archivos

MCIA	Maestría en Ciencias e Ingeniería Ambientales.
MCC	Maestría en Ciencias de la Computación.
MIE	Maestría en Ingeniería Estructural.
MCIM	Maestría en Ciencias e Ingeniería de Materiales.
MO	Maestría en Optimización.
MIP	Maestría en Ingeniería de Procesos.
DCIAM	Doctorado en Ciencias e Ingeniería (Ambiental, de Materiales).
DIE	Doctorado en Ingeniería Estructural.
DO	Doctorado en Optimización.
DIP	Doctorado en Ingeniería de Procesos.

Tabla 13.2.1 Claves para posgrados.

trimes	Trimestres
tesis	Tesis
docent	Docentes
alumno	Alumnos
posgra	Posgrados
planes	Planes de estudios
usuari	Usuarios

Tabla 13.2.2 Tipos de estatus de las entidades trimestres, alumnos, tesis, docentes, posgrados, planes de estudios y usuarios.

insact	Inscrito o Activo	Alumno con carga académica (UEA).
norein	No reinscrito	Alumno que no se inscribió a ninguna UEA, ni tampoco se inscribió "en blanco".
suspen	Suspendido	Alumno sancionado temporalmente por cometer una falta a la Universidad. Lo anterior, con base a un dictamen de órgano colegiado.
bajade	Baja definitiva	Recurso que tiene el alumno para solicitar su baja voluntaria en la Universidad.
egresa	Egresado	Alumno que concluyó los créditos del plan de estudios, cubrió actividades académicas aunque no tengan valor en créditos y solicitó su certificado de estudios de especialización, maestría o doctorado.
bajare	Baja reglamentaria	Es la pérdida de la calidad de alumno. Entre algunas de las causas está el no haber concluido sus estudios en el plazo máximo establecido en el plan de estudios. Así también en algunos planes de estudio se establece que el alumno sólo tiene dos oportunidades para aprobar una UEA.
egreaa	Egresado "a"	Egresado potencial ; es decir, es el alumno activo que está por concluir sus créditos del plan de estudios. Se le pone este estado para no afectar su historial académico, cuando su plan de estudios se encuentra en un proceso de conversión.
bajadi	Baja por dictamen de órgano colegiado	Expulsión definitiva del alumno dictada por órgano colegiado
insbla	Inscrito en blanco	Alumno que hizo el trámite de reinscripción sin "carga" de UEA. Lo anterior, para tener derecho a utilizar los servicios de biblioteca, actividades deportivas y salas de cómputo, etc.
acenuc	Aceptado nuevo ingreso	Estado académico temporal que se utiliza para identificar a los aspirantes seleccionados de primer ingreso. Sólo se tiene unos cuantos días ya que después cambia conforme el admitido tiene o no actividad escolar.
credcu	Créditos cubiertos	Alumno que ya concluyó los créditos del plan de estudios y actividades académicas requeridas aunque éstas no tengan valor en créditos. No ha solicitado su certificado de estudios de especialización, maestría o doctorado en la UAM, ni ha obtenido diploma de especialización ni grado académico de maestría o doctorado.
nueinp	Nuevo ingreso no presentado	Aspirante seleccionado de primer ingreso que no hizo ningún trámite de inscripción. Este estado empezó a utilizarse a partir del año de 1996.
abseis	Abandono de más de seis trimestres consecutivos	No aplica para posgrado. Se incluyó en este documento por compatibilidad con el catálogo de licenciatura.
alumdi	Alumno con diploma	Egresado de una especialización que obtuvo el grado académico correspondiente.
alugra	Alumno con grado	Egresado de una maestría o doctorado que obtuvo el grado académico correspondiente.

Tabla 13.2.3 Claves para estatus de alumnos por trimestre.

activo	Activo.
sabati	En período sabático.

Tabla 13.2.4 Claves para estatus de docentes.

proces	En proceso.
finali	Finalizada.

Tabla 13.2.5 Claves para estatus de tesis.

activo	Activo.
inacti	Inactivo.
proact	En proceso de aprobación.

Tabla 13.2.6 Claves para estatus de posgrados.

activo	Activo.
inacti	Inactivo.
proact	En proceso de aprobación.

Tabla 13.2.7 Claves para estatus de planes de estudios.

activo	Activo.
inacti	Inactivo.
inhabi	Inhabilitado.

Tabla 13.2.8 Claves para estatus de usuarios.

FOTO	Fotografía.
TITULO	Título de licenciatura o grado.
CVU	Currículum Vitae Único de Conacyt.

Tabla 13.2.9 Claves para documentos personales

TESIS	Tesis de maestría o doctorado.
LIBRO	Libros.
ARTICULO	Artículos de investigación.
REPORTE	Reportes técnicos.
MOVILIDAD	Movilidad de alumnos.

Tabla 13.2.10 Claves para documentos asociados al alumno.

PROYECTOS	Proyectos de investigación
-----------	----------------------------

Tabla 13.2.11 Clave para proyectos de investigación dirigidos por docentes.

PLAN	Planes de estudios
------	--------------------

Tabla 13.2.12 Clave para planes de estudios

### 13.3 Nomenclatura de archivos

La Figura 13.3.1 muestra la nomenclatura para el archivo de fotografía de la entidad usuarios.



Figura 13.3.1: Propuesta de nomenclatura para el archivo de fotografía de usuario.

La Figura 13.3.2 muestra la nomenclatura para los archivos correspondientes a las claves: CVU, TITULO, FOTO, PLAN, REPORTE, MOVILIDAD, LIBRO, TESIS, ARTICULO asociadas a la entidad alumnos.

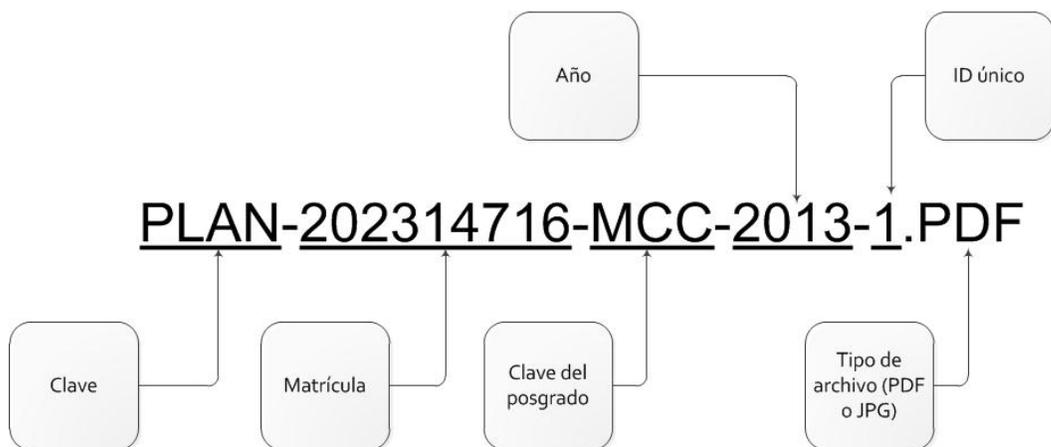


Figura 13.3.2: Propuesta de nomenclatura de archivos asociados a la entidad alumnos.

---

## 14 Bibliografía

---

- [1] S. R. Schach, “The Object-Oriented Paradigm, UML, and the Unified Process,” en *Introduction to object oriented analysis and design with UML and the unified process*, P. Ducham, Ed. New York: McGraw-Hill, 2004, pp. 47-48.
- [2] B. Holmes y D.T. Joyce, “Objects working together” en *Object-oriented programming with java*, J. H. Hauck, Ed., Sudbury, MA: Jones and Bartlett Publishers, Inc., 2001, pp. 582-586.

Universidad Autónoma Metropolitana Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería

Licenciatura en Ingeniería en Computación

Manual técnico

Gestión del archivo digital de la Coordinación de Posgrados de CBI mediante un sistema de información web

Hernández Báez Antonio

202314716

Neria Martínez Francisco Javier

206304789

Trimestre 13 - I

Abril de 2013

Asesora: Dra. María Lizbeth Gallardo López  
Profesor-investigador  
Departamento de Sistemas

---

# Índice de contenido

---

1	Introducción.....	IV
2	Hardware.....	IV
3	Software.....	IV
3.1	Servidor LAMP.....	6
3.2	Capa SSL.....	6
3.3	MVC.....	12
3.3.1	Modelo.....	12
3.3.2	Vista.....	14
3.3.3	Controlador.....	17
4	Red.....	17
5	Directorio de la aplicación.....	18
6	Instalación.....	19
7	Pruebas.....	21
8	Cuestiones sin resolver.....	24

---

## Índice de figuras

Figura 3.2.1: Mensajes del sistema operativo al habilitar la capa SSL mediante los comandos a2enmod y a2ensite.....	7
Figura 3.2.2: Instalación del software para generar certificados SSL.....	8
Figura 3.2.3: Generación del certificado para el SIP-CBI.....	9
Figura 3.2.4: Generación de clave privada insegura.....	9
Figura 3.2.5: Solicitud de certificación del certificado generado para el SIP-CBI.....	10
Figura 3.2.6: Autofirmado del certificado.....	10
Figura 3.2.7: Archivo default-ssl.....	11
Figura 3.2.8: Por medio del navegador se verifica que la conexión se encuentra en una capa cifrada..	12
Figura 3.2.9: Datos del certificado mostrados en el visor del navegador Chromium.....	12
Figura 3.3.1.1: Interfaz gráfica para el proceso de alta de un alumno.....	13
Figura 3.3.1.2: La interfaz muestra un mensaje que indica el procesamiento del formulario de manera exitosa.....	13
Figura 3.3.2.1: Interfaz gráfica del proceso de alta de un alumno.....	15
Figura 5.1: Directorio del SIP-CBI.....	19
Figura 6.1: Vista del SIP-CBI desde el navegador.....	21
Figura 7.1: Prueba de integridad referencial a la tabla alumnos.....	22
Figura 7.2: Prueba de integridad referencial a la tabla posgrados.....	22
Figura 7.3: Pruebas realizadas al proceso de autenticación de usuarios.....	23
Figura 7.4: Validación de los campos matrícula, nombre y apellido paterno.....	23
Figura 7.5: Validación de datos en el formulario para cambiar datos de un alumno.....	24

---

# 1 Introducción

---

El proyecto tiene por objetivo crear un sistema de información con el cual se facilitará la gestión de la información de los docentes, alumnos y planes de estudios pertenecientes a la Coordinación de Estudios de Posgrados de Ciencias Básicas e Ingeniería. El sistema también deberá ser capaz de generar reportes y almacenarlos en una imagen *ISO* de acuerdo a los parámetros establecidos por el PNPC<sup>1</sup> del Consejo Nacional de Ciencia y Tecnología.

Este documento incluye:

- Descripción de las características del hardware donde se instaló la aplicación.
- Características de la red para la instalación de la aplicación.
  - Software utilizado para la implementación:
  - Manejador de base de datos
  - Lenguaje de programación
  - Manejador de páginas dinámicas
  - Sistema operativo
- Estructura del directorio de la aplicación.
- Proceso de Instalación de la aplicación.
- Conjunto de pruebas de validación y de integridad referencial.
- Cuestiones sin resolver.

---

## 2 Hardware

---

Para implementar el SIP-CBI se utilizó el siguiente hardware:

- Equipo 1: *Laptop Compaq-Presario CQ43*, procesador *Intel® Celeron® Dual-Core T3500 2.10 GHz*, 6 GB en RAM, disco duro de 320 GB, sistema operativo *Windows 7*.
- Equipo 2: Computadora de escritorio con procesador *Intel® Pentium® Dual Core E5200 2.50 GHz*, 2 GB en RAM, disco duro de 400 GB, sistema operativo *Windows XP SP2*.

---

## 3 Software

---

EL SIP-CBI fue implementado en software libre. A continuación se menciona el software utilizado en la implementación y los criterios de selección:

- Infraestructura *LAMP (Linux, Apache<sup>2</sup>, MySQL, PHP)*. La infraestructura *LAMP* para el SIP-

---

1 PNPC: El Programa Nacional de Posgrados de Calidad de CONACyT solicita a las instituciones registradas en él un *CD* con un reporte que contenga datos académicos de alumnos, docentes y planes de posgrados, entre otros.

2 *HTTP Apache*: Servidor web desarrollado por la *Apache Software Foundation*.

CBI esta formada por:

- Sistema operativo *Debian 6* con *kernel 2.6.32.-5-486*, licencia *GNU/GPL* versión 3. Este sistema operativo ofrece corregir los errores de seguridad en sus paquetes mediante su equipo de auditoria de seguridad<sup>3</sup> y el denominado *Debian Security Tracker*<sup>4</sup>. Ofrece un servicio de listas de correo en donde informa sobre los últimos avisos de seguridad y comandos mínimos para obtener las últimas actualizaciones de sus paquetes<sup>5</sup>. La versión utilizada fue la más reciente al momento de iniciar la implementación.
- Servidor web *HTTP Apache server 2.2.16-6*, licencia *Apache License 2.0*<sup>6</sup>. Uno de los servidores más utilizados a nivel mundial. De acuerdo al sitio web *news.netcraft.com* el 56% de 625,329,303 sitios web encuestados, en noviembre de 2012, utiliza *Apache* como servidor web<sup>7</sup>.
- Gestor de bases de datos *MySQL server 5.1.49-3*, licencia *GNU/GPL* versión 2. Posee un amplio subconjunto de sentencias del lenguaje *SQL*, soporta transacciones y llaves foráneas, conectividad segura y la búsqueda e indexación de campos de texto.
- Lenguaje de programación *PHP server 5.3.3-7+squeeze1*, licencia *PHP License* versión 3.0<sup>8</sup>. Fue seleccionado para desarrollar el SIP-CBI por ser un lenguaje orientado al desarrollo de aplicaciones web dinámicas, tiene capacidad de conexión con el gestor de bases de datos *MySQL*, permite utilizar técnicas de programación orientada a objetos y permite escribir código ordenado, estructurado y manejable (ej. patrón MVC).
- Seguridad
  - Cifrado de contraseñas mediante el algoritmo *sha512*<sup>9</sup>. El algoritmo *MD5*<sup>10</sup> es utilizado en gran variedad de aplicaciones de seguridad y para verificar la integridad de archivos. En marzo de 2005 se demostró una colisión de *hash* provocada por la generación de dos llaves públicas distintas con el mismo *MD5 hash*. Por tal motivo se decidió cambiar el algoritmo de cifrado de contraseñas del SIP-CBI por *sha512*, el cuál, hasta el momento de desarrollar este proyecto terminal no ha presentado colisiones de *hash* y está soportado por *PHP*.
  - Capa segura *OpenSSL 0.9.8o4squeeze7*, licencia *Apache License 1.0*. Este software permite operar el servidor web *HTTP Apache* sobre una capa de seguridad cifrada.

La infraestructura *LAMP*, la capa de seguridad y la plataforma de desarrollo se instalaron en una máquina virtual en el software *VirtualBox 4.1.8* (licencia *GNU/GPL* versión 2) con el objetivo de implementar en el sistema operativo invitado (*Debian*) sin necesidad de modificar las particiones de los discos duros del hardware seleccionado para el desarrollo.

Los diagramas *UML* se elaboraron con software con licencia *GNU/GPL*: casos de uso y secuencias con *Dia 0.97.2*, diagrama de clases con *StarUML 5.0.2.1470* y el modelo lógico de la base de datos con *MySQL Workbench 5.2.40 CE*.

Las licencias *GNU/GPL*, *Apache License 2.0* permiten usar los programas para cualquier propósito.

---

3 <http://www.debian.org/security/audit/>

4 <http://security-tracker.debian.org/>

5 <http://www.debian.org/security/index.es.html#keeping-secure>

6 *Apache License 2.0*: Licencia de software libre creada por la *Apache Software Foundation* compatible con la *GNU General Public License*.

7 <http://news.netcraft.com/archives/2012/11/01/november-2012-web-server-survey.html>

8 *PHP License*: Licencia de código abierto creada por el *PHP Group*.

9 *SHA*: Es un sistema de funciones *hash* criptográficas.

10 *MD5 (Message-Digest Algorithm 5)*: Algoritmo de reducción criptográfico de 128 bits.

### 3.1 Servidor *LAMP*

El servidor web utilizado fue un *LAMP* (*Linux*, *Apache*, *MySQL* y *PHP*). Se instaló en un servidor *Debian* como se muestra en el Listado 1.

```
apt-get install apache2 libapache2-mod-php5 mysql-server php5-mysql  
/etc/init.d/apache2 restart
```

Listado 1

### 3.2 Capa *SSL*

En el sistema operativo *Debian* se instalaron el servidor web *Apache* y la capa de cifrado *SSL*. Una vez instalados fue necesario activar la capa *SSL* ejecutando los comandos mostrados en el Listado 2.

```
a2enmod ssl  
/etc/init.d/apache2 restart  
a2ensite default-ssl  
/etc/init.d/apache2 restart
```

Listado 2

En la Figura 3.2.1 se muestra la habilitación del módulo *SSL* para el servidor web *Apache*.

```

root@debian6pt:~# a2enmod ssl
Enabling module ssl.
See /usr/share/doc/apache2.2-common/README.Debian.gz on how to configure SSL and create self-signed certificates.
Run '/etc/init.d/apache2 restart' to activate new configuration!

root@debian6pt:~# /etc/init.d/apache2 restart
Restarting web server: apache2
apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName
... waiting ..apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName
.

root@debian6pt:~# a2ensite default-ssl
Enabling site default-ssl.
Run '/etc/init.d/apache2 reload' to activate new configuration!

root@debian6pt:~# /etc/init.d/apache2 restart
Restarting web server: apache2
apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName
... waiting .apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName

root@debian6pt:~#
root@debian6pt:~#
root@debian6pt:~#
root@debian6pt:~#
root@debian6pt:~#
root@debian6pt:~#
root@debian6pt:~#

```

Figura 3.2.1: Mensajes del sistema operativo al habilitar la capa SSL mediante los comandos *a2enmod* y *a2ensite*.

A continuación se instaló el software necesario para generar los certificados utilizando el comando mostrado en el Listado 3.

```
apt-get install openssl ca-certificates
```

Listado 3

La Figura 3.2.2 muestra el resultado de ejecutar el Listado 3 para instalar el software de generación de certificados.

```

root@debian6pt: /home/baez
Archivo Editar Ver Buscar Terminal Ayuda
root@debian6pt:/home/baez# apt-get install openssl ca-certificates
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
openssl ya está en su versión más reciente.
fijado openssl como instalado manualmente.
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
 libopenal1 chromium-bsu libalut0 ttf-uralic chromium-bsu-data libglc0
 libSDL-image1.2
Utilice «apt-get autoremove» para eliminarlos.
Se actualizarán los siguientes paquetes:
 ca-certificates
1 actualizados, 0 se instalarán, 0 para eliminar y 233 no actualizados.
Necesito descargar 146 kB de archivos.
Se liberarán 90.1 kB después de esta operación.
¿Desea continuar [S/n]? s
Des:1 http://http.us.debian.org/debian/ squeeze/main ca-certificates all 20090814+nmu3squeeze1 [146 kB]
Descargados 146 kB en 0seg. (154 kB/s)
Leyendo lista de cambios... Hecho.
Preconfigurando paquetes ...
(Leyendo la base de datos ... 123201 ficheros o directorios instalados actualmente.)
Preparando para reemplazar ca-certificates 20090814+nmu2 (usando ../ca-certificates_20090814+nmu3squeeze1_all.deb) ...
Desempaquetando el reemplazo de ca-certificates ...
Procesando disparadores para man-db ...
Configurando ca-certificates (20090814+nmu3squeeze1) ...
Updating certificates in /etc/ssl/certs... WARNING: Skipping duplicate certificate brasil.gov.br.pem
0 added, 1 removed; done.
Running hooks in /etc/ca-certificates/update.d...done.
:
```

Figura 3.2.2: Instalación del software para generar certificados SSL.

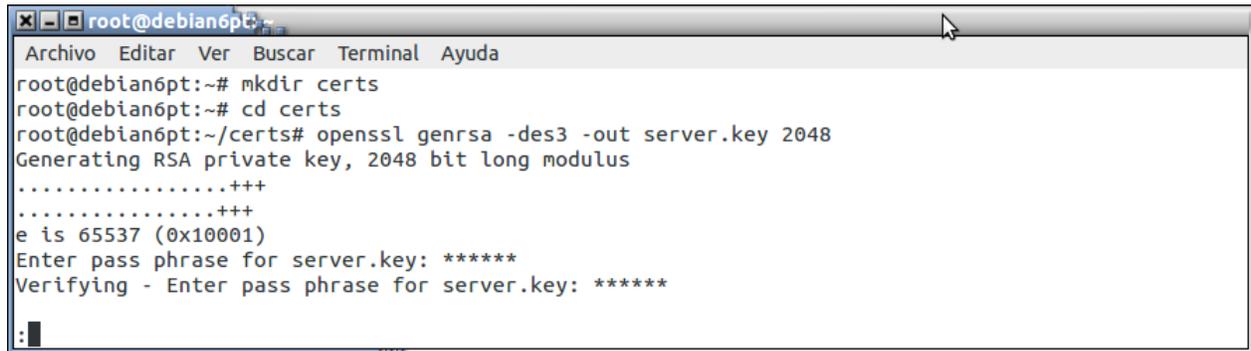
A continuación se creó un directorio para crear en él los certificados. Una vez en él, se generó la llave privada del certificado. Los comandos utilizados se observan en el Listado 4.

```

mkdir certs
cd certs
openssl genrsa -des3 -out server.key 2048
```

Listado 4

La generación de la llave privada solicita al usuario una contraseña como puede verse en la Figura 3.2.3.



```

root@debian6pt:~# mkdir certs
root@debian6pt:~# cd certs
root@debian6pt:~/certs# openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
Enter pass phrase for server.key: *****
Verifying - Enter pass phrase for server.key: *****
:

```

Figura 3.2.3: Generación del certificado para el SIP-CBI.

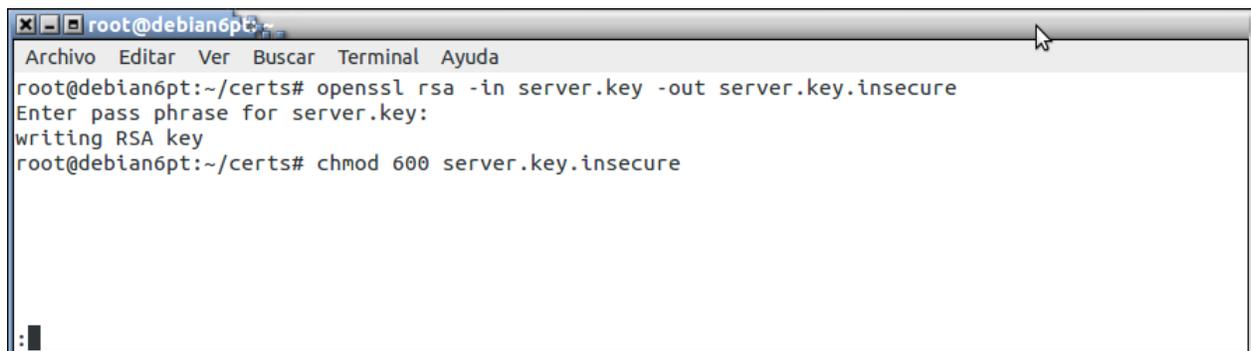
Se optó por utilizar una llave privada insegura para el caso donde el servidor se reiniciara sin la presencia del administrador del sistema o el dueño del certificado. Esta llave privada insegura se le otorgaron permisos de lectura y escritura para el superusuario (*root*). En el Listado 5 se observan los comandos utilizados. En la Figura 3.2.4 se observa la creación de la llave privada insegura. *OpenSSL* solicita al usuario la contraseña suministrada durante la creación de la llave privada.

```

openssl rsa -in server.key -out server.key.insecure
chmod 600 server.key.insecure

```

Listado 5



```

root@debian6pt:~/certs# openssl rsa -in server.key -out server.key.insecure
Enter pass phrase for server.key:
writing RSA key
root@debian6pt:~/certs# chmod 600 server.key.insecure
:

```

Figura 3.2.4: Generación de clave privada insegura.

A continuación se realizó la generación del certificado utilizando la llave privada mediante el comando mostrado en el Listado 6.

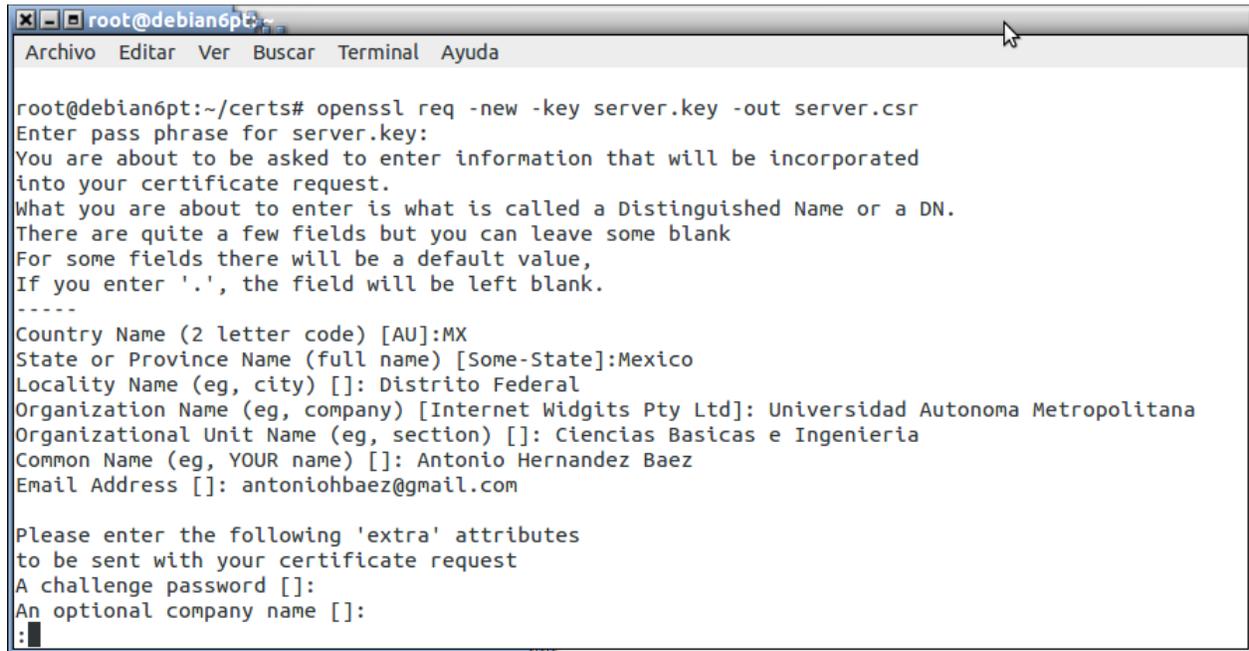
```

openssl req -new -key server.key -out server.csr

```

Listado 6

Durante la generación del certificado el sistema solicitó la contraseña asignada a la llave privada, clave del país (código *ISO 484*), Estado o provincia, localidad, organización, unidad en la organización, nombre y correo electrónico (ver Figura 3.2.5)



```

root@debian6pt:~/certs# openssl req -new -key server.key -out server.csr
Enter pass phrase for server.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:MX
State or Province Name (full name) [Some-State]:Mexico
Locality Name (eg, city) []: Distrito Federal
Organization Name (eg, company) [Internet Widgits Pty Ltd]: Universidad Autonoma Metropolitana
Organizational Unit Name (eg, section) []: Ciencias Basicas e Ingenieria
Common Name (eg, YOUR name) []: Antonio Hernandez Baez
Email Address []: antoniohbaez@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
:

```

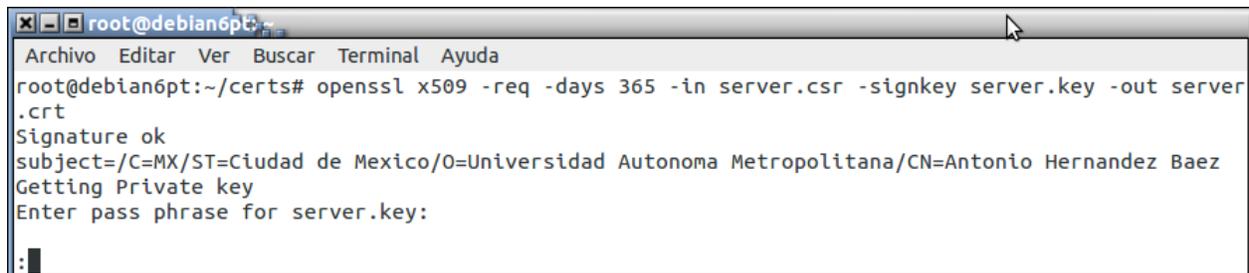
Figura 3.2.5: Solicitud de certificación del certificado generado para el SIP-CBI.

Finalmente, se realizó el autofirmado del certificado con el comando mostrado en el Listado 7.

```
openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
```

Listado 7

En el proceso de autofirmado se asigna la caducidad del certificado. En nuestro caso 365 días. Para realizar el autofirmado el sistema solicita nuevamente la contraseña de la llave privada (ver Figura 3.2.6).



```

root@debian6pt:~/certs# openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
Signature ok
subject=/C=MX/ST=Ciudad de Mexico/O=Universidad Autonoma Metropolitana/CN=Antonio Hernandez Baez
Getting Private key
Enter pass phrase for server.key:
:

```

Figura 3.2.6: Autofirmado del certificado.

Los archivos generados durante este proceso se colocaron en la ruta `/etc/ssl/`. El certificado `server.crt` se colocó en el subdirectorio `certs/` y la llave privada en el subdirectorio `private/`. Posteriormente se editó el archivo de configuración del servidor SSL por defecto `default-ssl` para indicar al servidor web Apache el lugar y nombre del certificado y la llave (ver Listado 8).

```
cp /etc/ssl/certs/server.crt
cp /etc/ssl/private/server.key.insecure
vim /etc/apache2/sites-enabled/default-ssl
```

Listado 8

Dentro del archivo *default-ssl* se incluyeron las directivas que se muestran en el Listado 9.

```
SSLCertificateFile /etc/ssl/certs/server.crt
SSLCertificateKeyFile /etc/ssl/private/server.key.insecure
```

Listado 9

La Figura 3.2.7 muestra parcialmente el archivo *default-ssl* y las directivas para leer el certificado y la llave privada.

```

CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined

#  SSL Engine Switch:
#  Enable/Disable SSL for this virtual host.
SSLEngine on

#  A self-signed (snakeoil) certificate can be created by installing
#  the ssl-cert package. See
#  /usr/share/doc/apache2.2-common/README.Debian.gz for more info.
#  If both key and certificate are stored in the same file, only the
#  SSLCertificateFile directive is needed.

#      LINEAS POR DEFAULT
#      SSLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem
#      SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key

#      CERTIFICADOS GENERADOS PARA EL SIP-CBI

SSLCertificateFile /etc/ssl/certs/server.crt
SSLCertificateKeyFile /etc/ssl/private/server.key.insecure

#  Server Certificate Chain:
#  Point SSLCertificateChainFile at a file containing the
#  concatenation of PEM encoded CA certificates which form the
#  certificate chain for the server certificate. Alternatively
#  the referenced file can be the same as SSLCertificateFile
#  when the CA certificates are directly appended to the server
#  certificate for convenience.
#SSLCertificateChainFile /etc/apache2/ssl.crt/server-ca.crt

-- INSERTAR --
50,60-67 20%
```

Figura 3.2.7: Archivo *default-ssl*

Una vez instalado el certificado y la llave privada se reinició el servidor web *Apache*. Para comprobar que el servidor web se encuentra ya sobre la capa de conexión segura, se solicitó a través de un navegador (en este caso particular *Chromium*) la URL con el protocolo *HTTPS*. Por tratarse de un certificado autofirmado el navegador enviará un mensaje de advertencia indicando que “el certificado del servidor no es de confianza” (Ver Figura 3.2.8).

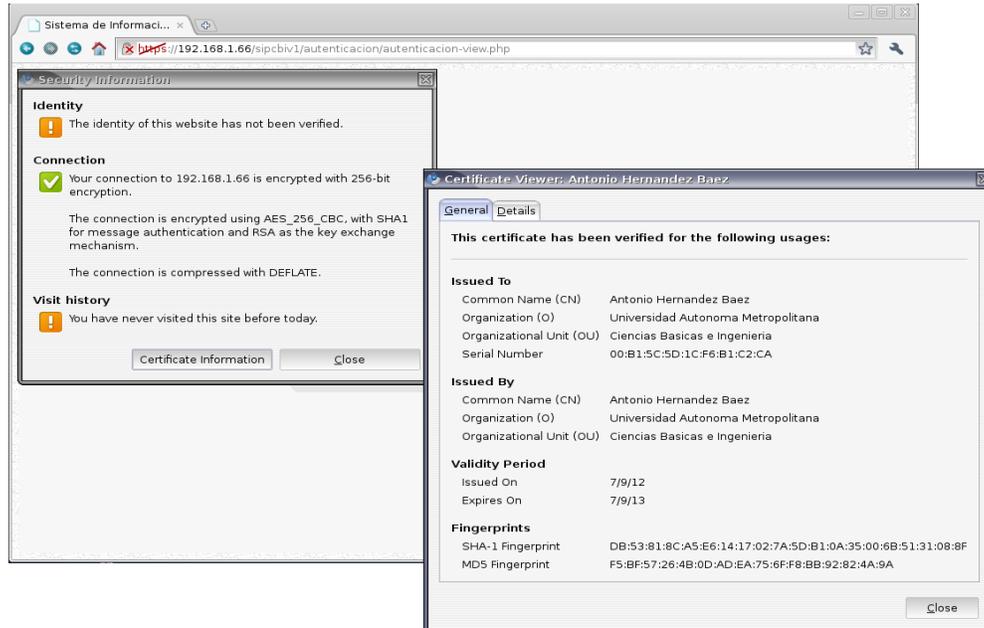


Figura 3.2.8: Por medio del navegador se verifica que la conexión se encuentra en una capa cifrada.

Al solicitar los datos del certificado nos muestra los datos suministrados durante la generación del certificado, vigencia y las huellas digitales cifradas con el algoritmo *sha* (ver Figura 3.2.9).



Figura 3.2.9: Datos del certificado mostrados en el visor del navegador *Chromium*.

## 3.3 MVC

### 3.3.1 Modelo

Se implementó un modelo para cada módulo del sistema. Cada modelo incluye las operaciones *select*, *update*, *insert* y *delete* realizadas a la base de datos. La Figura 3.3.1.1 muestra la interfaz gráfica para la operación de alta de un alumno. Esta interfaz gráfica presenta el formulario con los datos solicitados

por el SIP-CBI. Al ser procesado el formulario se muestra un mensaje como se observa en la Figura 3.3.1.2. En el Listado 10 se muestra parcialmente del código perteneciente a la clase *Alumnos()* contenida en el archivo *alumnos-model.php* que corresponde a la operación de alta de un alumno.

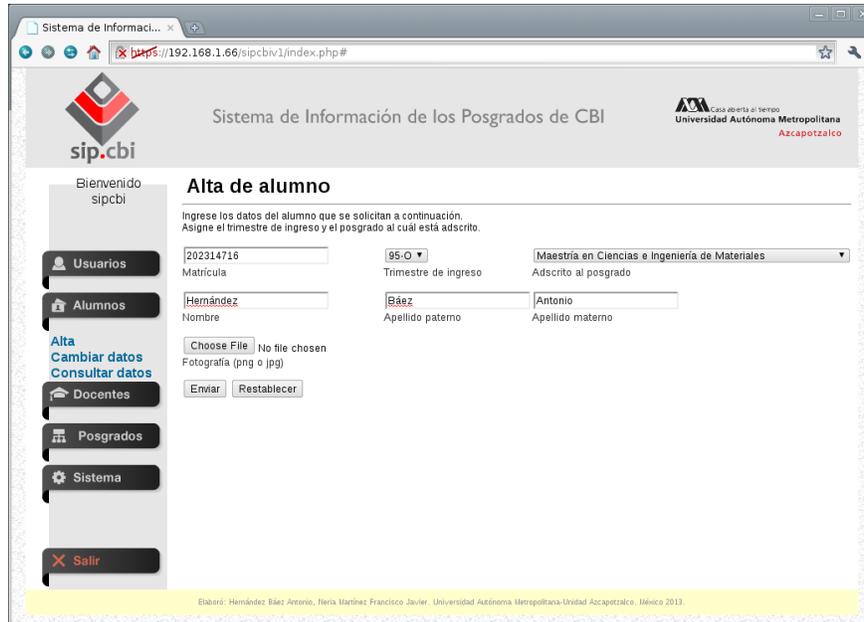


Figura 3.3.1.1: Interfaz gráfica para el proceso de alta de un alumno.



Figura 3.3.1.2: La interfaz muestra un mensaje que indica el procesamiento del formulario de manera exitosa.

```

function altaalumnos($trimestre, $matricula, $nombre, $ap, $am, $posgrado,$foto)
{
    if($_FILES["foto"]["error"] != UPLOAD_ERR_OK)
    {
        $sql="INSERT INTO sipcbi.alumnos (id_alumnos, id_trimestreingreso_alumnos, matricula_alumnos, nombre_alumnos,
apellidop_alumnos, apellidom_alumnos) VALUES (NULL, ".$trimestre.", ".$matricula.", ".$nombre.", ".$ap.", ".$am.");";
        mysql_query($sql,Conectar::con());

        $sqlultimoid="SELECT MAX(id_alumnos) FROM sipcbi.alumnos";
        $arregloultimoid=mysql_query($sqlultimoid,Conectar::con());
        $elementoultimoid = mysql_fetch_array($arregloultimoid);
        $ultimoid=$elementoultimoid[0];
        $sql2="INSERT INTO sipcbi.alumnoadscrito (id_alumnoadscrito, id_alumnos_alumnoadscrito, id_posgrados_alumnoadscrito) VALUES
(NULL, ".$ultimoid.", ".$posgrado.");";
        mysql_query($sql2, Conectar::con());
        mysql_close();
    }
    else
    {
        $sql="INSERT INTO sipcbi.alumnos (id_alumnos, id_trimestreingreso_alumnos, matricula_alumnos, nombre_alumnos,
apellidop_alumnos, apellidom_alumnos) VALUES (NULL, ".$trimestre.", ".$matricula.", ".$nombre.", ".$ap.", ".$am.");";
        mysql_query($sql,Conectar::con());
        $sqlultimoid="SELECT MAX(id_alumnos) FROM sipcbi.alumnos";
        $arregloultimoid=mysql_query($sqlultimoid,Conectar::con());
        $elementoultimoid = mysql_fetch_array($arregloultimoid);
        $ultimoid=$elementoultimoid[0];
        $sql2="INSERT INTO sipcbi.alumnoadscrito (id_alumnoadscrito, id_alumnos_alumnoadscrito, id_posgrados_alumnoadscrito) VALUES
(NULL, ".$ultimoid.", ".$posgrado.");";
        mysql_query($sql2, Conectar::con());

        $tipomime=$_FILES['foto']['tmp_name'];
        $tipomime2=mime_content_type($tipomime);
        if ($tipomime2=="image/jpeg")
        {
            $extension=".jpg";
        }
        else if ($tipomime2=="image/png")
        {
            $extension=".png";
        }
        else if ($tipomime2=="image/gif")
        {
            $extension=".gif";
        }
        else
        {
            $extension=".txt";
        }
        $sqlclaveposgrado="SELECT clave_posgrados FROM sipcbi.posgrados WHERE id_posgrados=".$posgrado.";";
        $arregloclaveposgrado=mysql_query($sqlclaveposgrado,Conectar::con());
        $elementoclaveposgrado = mysql_fetch_array($arregloclaveposgrado);
        $claveposgrado=$elementoclaveposgrado[0];

        $AA=date ("Y");
        if(is_uploaded_file($_FILES['foto']['tmp_name']))
        {
            $nombreDirectorio = "../archivo/fotos/";

            $nombreArchivo = "FOTO-".$matricula."-".$claveposgrado."-".$AA."-".$ultimoid."".$extension;
            if(move_uploaded_file($_FILES['foto']['tmp_name'], $nombreDirectorio.$nombreArchivo))
            {
                echo "Archivo subido correctamente";
            }
            else
            {
                echo "No puede mover el archivo a su ubicación final";
            }
        }
        else
        {
            echo "No se ha podido subir el archivo";
        }
        $sqlnombearchivo="UPDATE sipcbi.alumnos SET foto_alumnos=".$nombreArchivo." WHERE id_alumnos=".$ultimoid.";";
        mysql_query($sqlnombearchivo,Conectar::con());
        mysql_close();
    }
}

```

Listado 10: vista parcial de la clase *Alumnos* contenida en el archivo *alumnos-model.php*

### 3.3.2 Vista

Se implementaron vistas para cada módulo del sistema. Las vistas incluyen: *HTML*, *JavaScript*,

*Jquery*, *AJAX* y controladores propios en *PHP*. La Figura 3.3.2.1 muestra la interfaz gráfica para el proceso de alta de un alumno.

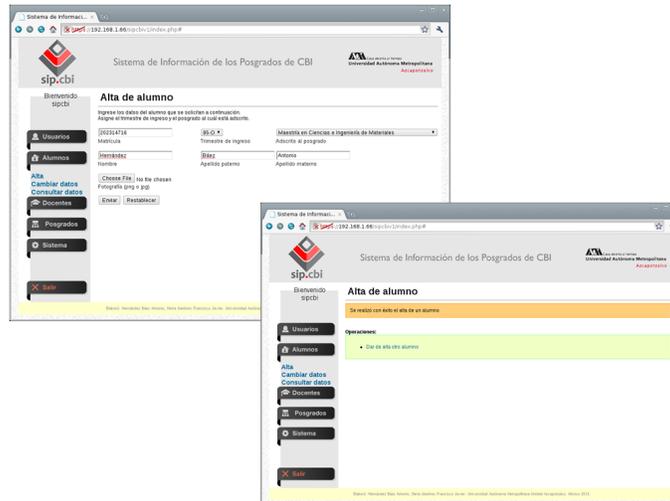


Figura 3.3.2.1: Interfaz gráfica del proceso de alta de un alumno.

Por medio del controlador se asignan nombres a las secciones que la vista deberá mostrar al usuario. En el caso específico del alta de alumno se muestran dos secciones: la primera denominada *alta*, que presenta el formulario para dar de alta al alumno, y la segunda, llamada *mensajealta*, que muestra un mensaje que informa del registro exitoso del alumno en el sistema.

En el Listado 11 se observa parcialmente el código de del archivo *alumnos-alta-view.php*.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Vista alta</title>
...
</head>
<body>
<?php
if ($seccion=="alta")
{
...
}
else if ($seccion=="mensajealta")
{
...
}
?>
</body>
</html>
```

Listado 11: Secciones principales de la vista (archivo *alumnos-alta-view.php*).

En el Listado 12 se muestra parcialmente el código fuente que genera la vista del formulario.

```

<h1>Alta de alumno</h1><hr /><div>Ingrese los datos del alumno que se solicitan a continuación.</div><div>Asigne el trimestre de
ingreso y el posgrado al cuál está adscrito.</div><br />
<form action="alumnos-alta-controller.php?operacion=1" method="post" enctype="multipart/form-data" name="form1" id="form1"
class="formular">
<table width="780" border="0" cellspacing="0" cellpadding="0">
<tr>
<td width="33%"><input class="validate[required,custom[integer]] text-input" name="matricula" type="text" id="matricula"
placeholder="Matricula" size="20" maxlength="9" /></td>
<td width="33%">
<select name="trimestre" id="trimestre">
<option value="">---</option>
<?php foreach($arreglo as $lista): ?>
<option value="<?php echo $lista['id_trimestres']; ?>">
<?php
echo $lista['alias_trimestres'];
?></option>
<?php
endforeach;
?>
</select>
</td>
<td width="33%">
<select name="posgrado" id="posgrado">
<option value="">---</option>
<?php foreach($arreglo2 as $lista2): ?>
<option value="<?php echo $lista2['id_posgrados']; ?>">
<?php
echo $lista2['nombre_posgrados'];
?></option>
<?php
endforeach;
?>
</select>
</td>
</tr>
<tr>
<td width="33%"><label for="matricula" class="rotulosformularios">Matricula</label></td>
<td width="33%"><label for="trimestreingreso" class="rotulosformularios">Trimestre de ingreso</label></td>
<td width="33%"><label for="posgrado" class="rotulosformularios">Adscrito al posgrado</label></td>
</tr>
<tr>
<td width="33%">&nbsp;</td><td width="33%">&nbsp;</td><td width="33%">&nbsp;</td>
</tr>
<tr>
<td width="33%"><input class="validate[required] text-input" type="text" name="nombre" type="text" id="nombre" size="20"
placeholder="Nombre" maxlength="50" /></td>
<td width="33%"><input class="validate[required] text-input" type="text" name="ap" type="text" id="apellido paterno" size="20"
placeholder="Apellido paterno" maxlength="50" /></td>
<td width="33%"><input name="am" type="text" id="apellido materno" size="20" placeholder="Apellido materno" maxlength="50"
/></td>
</tr>
<tr>
<td width="33%"><label for="nombre" class="rotulosformularios">Nombre</label></td><td width="33%"><label for="apellido
paterno"
class="rotulosformularios">Apellido paterno</label></td><td width="33%"><label for="apellido materno"
class="rotulosformularios">Apellido materno</label></td></tr>
<tr>
<td width="33%">&nbsp;</td><td width="33%">&nbsp;</td><td width="33%">&nbsp;</td></tr>
<tr>
<td width="33%"><input name="foto" type="file" id="foto" size="15" name="foto" placeholder="Fotografia" /></td><td
width="33%">&nbsp;</td><td width="33%">&nbsp;</td></tr>
<tr>
<td width="33%">&nbsp;</td><td width="33%"><label class="rotulosformularios">Fotografia (png o jpg)</label></td><td
width="33%">&nbsp;</td></tr>
</table>
<?php if ($cantidad==0)
{
?>
<input type="submit" name="enviar" id="enviar" value="Enviar" disabled="" />
<input type="reset" name="cancelar" id="cancelar" value="Restablecer" disabled="" />
 No se pueden agregar alumnos.<br />Es necesario dar de alta trimestres
<?php
}
else
{
?>
<input type="submit" name="enviar" id="enviar" value="Enviar" />
<input type="reset" name="cancelar" id="cancelar" value="Restablecer" />
<?php
}
?>
</form>

```

Listado 12: Sección *alta* contenida en el archivo *alumnos-alta-view.php*

En el Listado 13 se muestra el código que genera la sección *mensajealta*.

```

<h1>Alta de alumno</h1>
<hr />
<div class="aviso">
Se realizó con éxito el alta de un alumno
</div>
<br>
<br>
<strong>Operaciones:</strong>
<div class="agregar">
<ul>
<li><a href="alumnos-alta-controller.php?operacion=0">Dar de alta otro alumno</a></li>
</ul>
</div>

```

Listado 13: Sección *mensajealta* contenida en el archivo *alumnos-alta-view.php*.

### 3.3.3 Controlador

Se implementaron controladores para cada módulo. Estos controladores tienen las siguientes funciones:

- Crean los objetos a partir de las clases contenidas en el modelo.
- Invocan a los métodos de los objetos creados.
- Controlan las secciones que el usuario podrá observar en la interfaz gráfica.
- Reciben información proveniente de la interfaz gráfica.
- Reciben información proveniente del modelo.

En el Listado 14 se observa parcialmente el código fuente del controlador contenido en el archivo *alumnos-alta-controller.php*. Se observa: el control de las secciones con el uso de sentencia *case*, la inclusión del modelo y la vista, la creación del objeto a partir de una clase, la llamada a métodos de la clase y la destrucción del objeto.

```

<?php
$operacion=$_GET['operacion'];
switch ($operacion)
{
    case 0:
        include("alumnos-model.php");
        $seccion="alta";
        $trim = new Alumnos();
        $arreglo=$trim -> listatrimestres();
        $arreglo2=$trim -> listaposgrados();
        include("alumnos-alta-view.php");
        unset($trim);
        break;
    case 1:
        include("alumnos-model.php");
        $seccion="mensajealta";
        $alta= new Alumnos();
        $alta -> altaalumnos($_POST["trimestre"], $_POST["matricula"], $_POST["nombre"], $_POST["ap"], $_POST["am"], $_POST["posgrado"],
$_FILES['foto']);
        include("alumnos-alta-view.php");
        unset($alta);
        break;
    ...
    ...
}

```

Listado 14: *alumnos-alta-controller.php*


---

## 4 Red

*HTTP* es un protocolo de transferencia de hipertexto, que será usado para cada una de las transacciones que ocurran en el sistema, a saber: el inicio de sesión, las modificaciones y el fin de una sesión.

*TCP/IP* es un modelo de red que especifica cómo un equipo se conectará a una red. El nombre del modelo se origina por los dos principales protocolos de la pila llamados: a) *Transmission Control Protocol (TCP)* orientado a conexión, que garantiza que los datos se entreguen sin errores a su destino, e b) *Internet Protocol (IP)* no orientado a conexión, que permite el envío de datos en forma de paquetes a través de la red y proporciona una dirección *IP* para la identificación de los dispositivos.

El SIP-CBI está diseñado como un sistema de información web, por tanto, puede funcionar y ser accedido desde Internet o una intranet con protocolos *HTTP* y *TCP/IP*.

---

## 5 Directorio de la aplicación

---

El SIP-CBI se organizó de la siguiente manera:

- **media:** Directorio exclusivo para los diseñadores gráficos. Aquí podrán trabajar libremente creando los archivos de imagen que la aplicación requiera.
- **usuarios:** Directorio del módulo de usuarios. Aquí se almacenan los archivos correspondientes al modelo, la lógica de la vista y el controlador, exclusivos del módulo de usuarios.
- **alumnos:** Directorio del módulo de usuarios. Aquí se almacenan los archivos correspondientes al modelo, la lógica de la vista y el controlador, exclusivos del módulo de alumnos.
- **Docentes:** Directorio del módulo de usuarios. Aquí se almacenan los archivos correspondientes al modelo, la lógica de la vista y el controlador, exclusivos del módulo de docentes.
- **posgrados:** Directorio del módulo de usuarios. Aquí se almacenan los archivos correspondientes al modelo, la lógica de la vista y el controlador, exclusivos del módulo de posgrados.
- **Trimestres:** Directorio del módulo de usuarios. Aquí se almacenan los archivos correspondientes al modelo, la lógica de la vista y el controlador, exclusivos del módulo de trimestres.
- **Estatus:** Directorio del módulo de estatus. Aquí se almacenan los archivos correspondientes al modelo, la lógica de la vista y el controlador, exclusivos del módulo de estatus.
- **Archivo:** En este directorio se almacenan los archivos en formato *PDF*. Contiene a los directorios:
  - antecedentes
  - articulos
  - fotos
  - libros
  - movilidades
  - posgrados
  - reportes
  - tesis
  - usuarios

En la Figura 5.1 se observa el directorio del SIP-CBI.

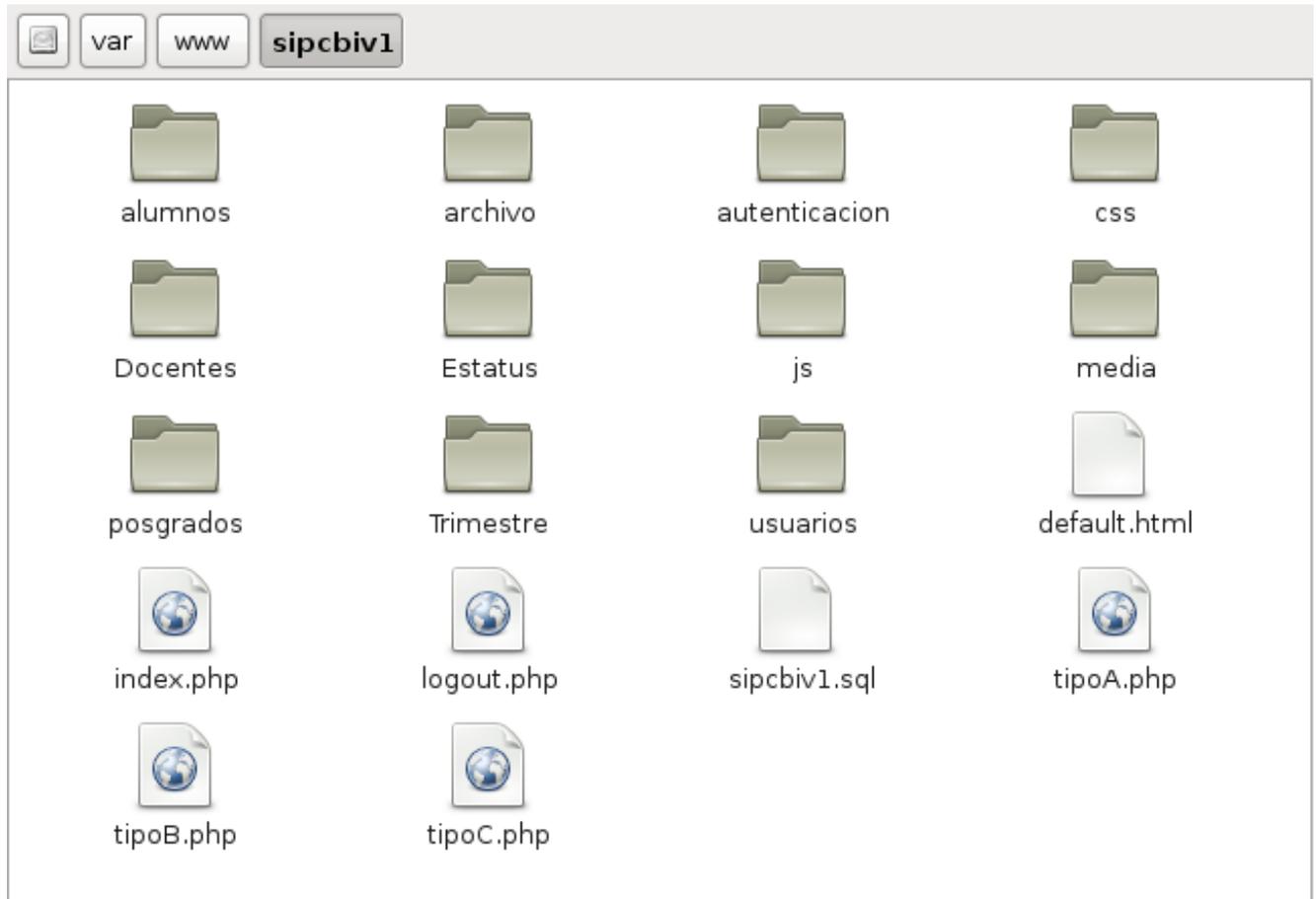


Figura 5.1: Directorio del SIP-CBI.

## 6 Instalación

Crear una base de datos y usuario asociado a ella, llamados *sipcbi*, como se muestra en el Listado 15. En caso de contar con una base de datos hospedada en servidores de terceros, deberá verificarse que se cuente con los permisos *CREATE* y *DROP*, para tablas, y *DELETE*, *INSERT*, *SELECT* y *UPDATE*, para campos.

```
mysql> CREATE DATABASE sipcbi;
mysql> GRANT ALL PRIVILEGES ON sipcbi.* TO 'sipcbi'@'localhost' IDENTIFIED BY 'qazwsx';
```

Listado 15

Descomprimir en el directorio usado por *Apache*, o el asignado por el servicio de hospedaje web de terceros, el archivo *sipcbiv1-2abr2013.tar.gz* como se muestra en el Listado 16.

```
cd /var/www/
tar xvzf sipcbiv1-2abr2013.tar.gz
```

Listado 16

Configurar en el archivo *config.cnf* los datos de conexión a la base de datos como se muestra en el Listado 17. El archivo se localiza en el directorio *autenticacion/*. Esta configuración puede variar en caso de contar con hospedaje web de terceros.

```
<?php
$servidor="localhost";
$usuariodb="sipcbi";
$passwordbd="qazwsx";
$db="sipcbi";
?>
```

Listado 17

Cargar la estructura de la base de datos contenida en el archivo *sipcbiv1.sql* que se encuentra en el directorio *sipcbiv1/*. La base puede cargarse desde el *shell* en modo texto de *linux* con ayuda de *MySQL* (Listado 18) o por medio de un gestor gráfico como *phpMyAdmin*.

```
mysql --user=sipcbi -password=qazwsx sipcbi < sipcbiv1.sql
```

Listado 18

Dentro del directorio *sipcbiv1/* deberá asignarse permisos de lectura, escritura y ejecución al subdirectorio *archivo/*. A su vez, *archivo/* contiene a los directorios: *antecedentes/*, *articulos/*, *datospersonales/*, *fotos/*, *libros/*, *movilidades/*, *posgrados/*, *reportes/*, *tesis/* y *usuarios/* a los cuales también se les deberá asignar permisos de lectura, escritura y ejecución. Con esta asignación de permisos el SIP-CBI podrá gestionar el archivo digital. Si se cuenta con acceso al *shell* en modo texto de *linux* podrán ejecutarse los comandos que se muestran en el Listado 19.

```
cd sipcbiv1/
chmod 777 archivo
cd archivos/
chmod 777 antecedentes articulos datospersonales fotos libros movilidades posgrados
reportes tesis usuarios
```

Listado 19

Escribir en la barra del *URL* del navegador la *IP* o nombre del servidor, y a continuación la ruta donde fue instalado el SIP-CBI. En la Figura 6.1 puede observarse que el SIP-CBI se instaló en un servidor con *IP* 192.168.1.66 en el directorio */var/www/* por lo que deberá teclearse en la *URL* de navegador: *http://192.168.1.66/sipcbiv1/*. En caso de usar un servidor web con capa *SSL* deberá sustituirse *http* por *https*.



Figura 6.1: Vista del SIP-CBI desde el navegador.

Para autenticarse por primera vez, el SIP-CBI cuenta con el usuario *sipcbi* con contraseña *qazwsx*. Este usuario le permitirá acceder al sistema y dar de alta a otros usuarios. Se recomienda cambiar la contraseña de este usuario y eliminar el archivo *sipcbiv1.sql* una vez instalado el sistema.

El sistema se instalará con una lista de trimestres, estatus y posgrados definidos en la etapa de diseño.

---

## 7 Pruebas

Se realizaron pruebas de integridad referencial al modelo de bases de datos implementado. Al realizar el alta de un alumno no solo se realiza la inserción del registro en la tabla *alumnos*, también se inserta el posgrado al cual se encuentra adscrito el alumno en la tabla *alumnoadscrito*. Esto crea una relación entre las tablas *alumnos* y *posgrados* a través de la tabla *alumnoadscrito*.

Para comprobar la integridad referencial entre las tablas se realizó lo siguiente:

1. Mostrar la lista de alumnos del SIP-CBI
2. Seleccionar el *id* de un alumno.
3. Mostrar la lista de alumnos adscritos.
4. Identificar el *id* del alumno y el *id* del posgrado al cual está adscrito
5. Borrar de la tabla de *alumnos* al alumno con ayuda del *id*.

Al intentar borrar el registro del alumno seleccionado el sistema envió un mensaje de error indicando que el registro no puede ser borrado ya que forma parte de una llave foránea en la tabla *alumnoadscrito*. En la Figura 7.1 se muestra el procedimiento antes descrito y el mensaje de error al intentar borrar el registro.

```

root@debian6pt: ~
mysql>
mysql>
mysql> SELECT * FROM alumnos;
+-----+-----+-----+-----+-----+-----+-----+
| id_alumnos | id_trimestreingreso_alumnos | matricula_alumnos | nombre_alumnos | apellidop_alumnos | apellidom_alumnos | foto_alumnos |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 9 | 202314716 | Antonio | Hernández | Báez | NULL |
| 2 | 66 | 95303250 | José | Hernández | Báez | NULL |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> SELECT * FROM alumnoadscrito;
+-----+-----+-----+
| id_alumnoadscrito | id_alumnos_alumnoadscrito | id_posgrados_alumnoadscrito |
+-----+-----+-----+
| 1 | 1 | 2 |
| 2 | 2 | 4 |
+-----+-----+-----+
2 rows in set (0.01 sec)

mysql> DELETE FROM posgrados WHERE id_posgrados='2';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('sipcbi`.`alumnoadscrito', CONSTRAINT `fk_alumnoadscrito_posgrados1` FOREIGN KEY (`id_posgrados_alumnoadscrito`) REFERENCES `posgrados` (`id_posgrados`) ON DELETE NO ACTION ON UPDATE NO ACTION)
mysql>

```

Figura 7.1: Prueba de integridad referencial a la tabla *alumnos*.

Del mismo modo se realizó una prueba con la tabla de posgrados. Al intentar borrar un posgrado que se encuentra relacionado con un alumno en la tabla *alumnoadscrito* el sistema enviará un mensaje de error (ver Figura 7.2).

```

root@debian6pt: ~
+-----+-----+-----+-----+
| id_posgrados | id_estatus_posgrados | nombre_posgrados | clave_posgrados |
+-----+-----+-----+-----+
| 1 | 3 | Maestría en Ciencias e Ingeniería Ambientales | MCIA |
| 2 | 3 | Maestría en Ciencias de la Computación | MCC |
| 3 | 3 | Maestría en Ingeniería Estructural | MIE |
| 4 | 3 | Maestría en Ciencias e Ingeniería de Materiales | MCIM |
| 5 | 3 | Maestría en Optimización | MO |
| 6 | 3 | Maestría en Ingeniería de Procesos | MIP |
| 7 | 3 | Doctorado en Ciencias e Ingeniería (Ambiental, de Materiales) | DCIAM |
| 8 | 3 | Doctorado en Ingeniería Estructural | DIE |
| 9 | 3 | Doctorado en Optimización | DO |
| 10 | 3 | Doctorado en Ingeniería de Procesos | DIP |
+-----+-----+-----+-----+
10 rows in set (0.08 sec)

mysql> DELETE FROM posgrados WHERE id_posgrados='2';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('sipcbi`.`alumnoadscrito', CONSTRAINT `fk_alumnoadscrito_posgrados1` FOREIGN KEY (`id_posgrados_alumnoadscrito`) REFERENCES `posgrados` (`id_posgrados`) ON DELETE NO ACTION ON UPDATE NO ACTION)
mysql>

```

Figura 7.2: Prueba de integridad referencial a la tabla *posgrados*.

También se realizaron pruebas de validación al SIP-CBI. La primer prueba se realizó en el formulario de autenticación de usuarios. Se intentó acceder con los campos de usuario y contraseña vacíos pero el sistema envió un mensaje advirtiéndolo que faltaba completar ambos campos. En otro caso se intentó

acceder con un usuario y contraseña no existentes en la tabla *usuarios*. Para este caso, el sistema envió un mensaje de error en la autenticación. En la Figura 7.3 se muestra el resultado de la prueba.

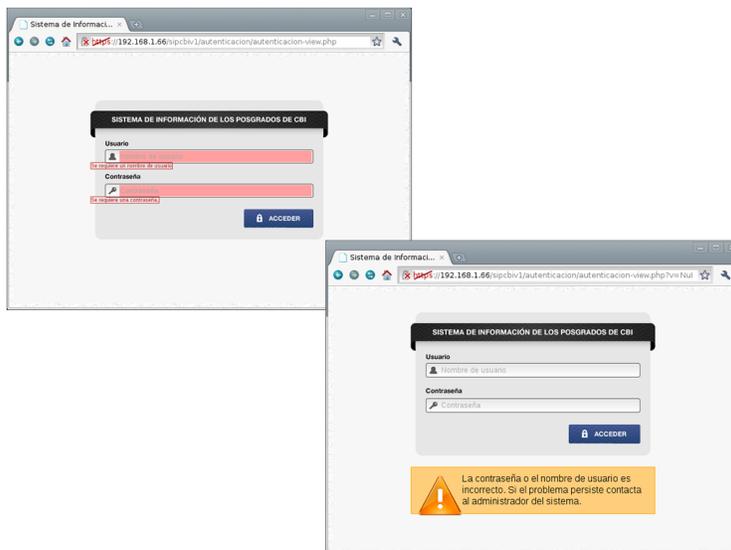


Figura 7.3: Pruebas realizadas al proceso de autenticación de usuarios.

Para el módulo de alumnos se realizó una prueba de validación en los campos. Se dejaron vacíos intencionalmente los campos matrícula, nombre y apellido paterno. Al procesar el formulario el sistema envía una serie de mensajes que indican los campos que deben llenarse (ver Figura 7.4).



Figura 7.4: Validación de los campos matrícula, nombre y apellido paterno.

Para el formulario que actualiza los datos del alumno se realizó una prueba similar. En este caso se eliminó el nombre del alumno y al intentar procesar el formulario se mostró un mensaje que indicó el campo necesario a llenar para procesar el formulario.

Figura 7.5: Validación de datos en el formulario para cambiar datos de un alumno.

## 8 Cuestiones sin resolver

Inicialmente se eligió el *framework* *CakePHP* 2.3 como entorno de desarrollo pero su uso fue descartado posteriormente debido a su poca flexibilidad en el manejo de los datos, lo cual corrompía el diseño original; por lo tanto, la implementación de las clases del sistema no extienden ni heredan de las clases de alguna *framework*

Los tres aspectos que se mencionan a continuación fueron determinantes para detener la implementación por medio del *framework*:

- Crear nuevamente las relaciones entre tablas en las clases generadas por el *framework* mediante asociaciones *hasOne*, *hasMany*, *belongsTo* y *hasAndBelongsToMany* (el modelo relacional elaborado en *MySQL* ya contiene estas relaciones)
- El uso de listas de control de acceso en archivos de texto para controlar la autenticación de usuarios (las listas contienen el nombre de usuario y no están cifradas), y
- La falta de documentación por parte de los desarrolladores del *framework* para modificar la vista y adaptarla al diseño gráfico personalizado

Por otra parte, algunas validaciones del sistema quedaron pendientes. Al utilizar tecnologías como *JavaScript*, *AJAX* y *JQuery*, y mezclar sus *script* de validaciones provocaron la inutilización del sistema o el incorrecto funcionamiento de las funciones validadoras. Por lo que se optó por utilizar únicamente *JQuery* para validar. Se recomienda utilizar un solo tipo de tecnología para realizar las validaciones.

Universidad Autónoma Metropolitana Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería

Licenciatura en Ingeniería en Computación

Manual de usuario

Gestión del archivo digital de la Coordinación de Posgrados de CBI mediante un sistema de información web

Hernández Báez Antonio

202314716

Neria Martínez Francisco Javier

206304789

Trimestre 13 - I

Abril de 2013

Asesora: Dra. María Lizbeth Gallardo López  
Profesor-investigador  
Departamento de Sistemas

---

## Índice de contenido

---

1	Introducción.....	1
2	Ingreso al SIP-CBI.....	2
3	Módulos que integran el SIP-CBI.....	4
4	Módulo de usuarios.....	9
5	Módulo de alumnos.....	15
6	Finalizar sesión.....	22

---

# 1 Introducción

---

El SIP-CBI surgió debido a que los Coordinadores de los posgrados de la División de Ciencias Básicas e Ingeniería tienen la necesidad de llevar el control de la información de los planes de estudio, alumnos y docentes. Esta información se encuentra distribuida en documentación impresa y en archivos en formato electrónico; a los cuales nos referiremos como archivo del posgrado. Actualmente estos archivos se encuentran almacenados en diferentes computadoras lo que dificulta su rápida localización y administración, además, no se lleva un control de las personas que acceden a esta información.

El presente manual pretende transmitir la estructura, conceptos e información básica de la operación del SIP-CBI, con la finalidad de que conociendo su funcionamiento los usuarios lo puedan utilizar de manera efectiva. Por lo mencionado, es de suma importancia, leer el manual que se detalla a continuación antes y durante el manejo del sistema.

El manual explica: la forma de ingresar al SIP-CBI, el acceso a los módulos, las operaciones a realizar en cada uno de ellos y como salir del sistema.

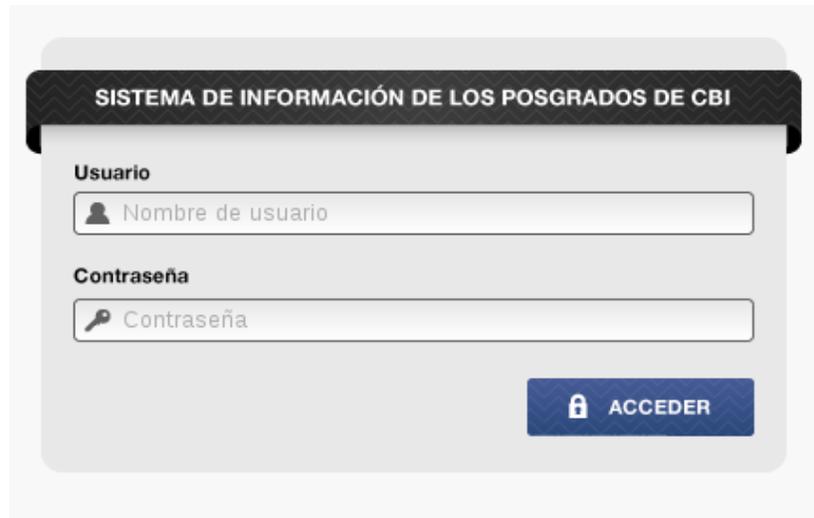
---

## 2 Ingreso al SIP-CBI

---

Una vez instalado el SIP-CBI en un servidor web<sup>1</sup>, abrimos el navegador y escribimos la URL<sup>2</sup> donde se encuentra el sistema. Una vez hecho esto podremos ver la pantalla de inicio de sesión.

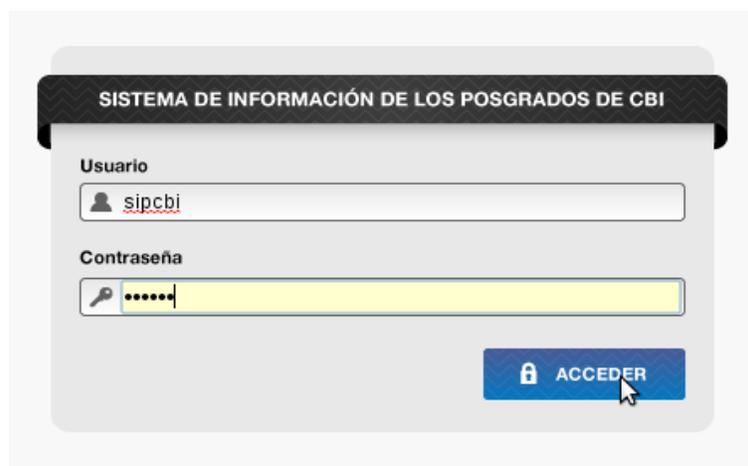
Es una pantalla de inicio de sesión (ver Imagen 1), donde se requiere un nombre de usuario y contraseña.



La imagen muestra la interfaz de inicio de sesión del sistema. El título es "SISTEMA DE INFORMACIÓN DE LOS POSGRADOS DE CBI". Hay dos campos de entrada: "Usuario" con el texto "Nombre de usuario" y "Contraseña" con el texto "Contraseña". Un botón azul con un icono de candado y el texto "ACCEDER" está ubicado en la parte inferior derecha.

Imagen 1. Pantalla de inicio de sesión.

El sistema cuenta con un usuario y contraseña precargados: *sipcbi* y *qazwsx*, respectivamente. Al introducir los datos anteriores y presionar el botón "Acceder" (ver Imagen2), podremos ingresar al sistema, ver Imagen 3.



La imagen muestra la interfaz de inicio de sesión con los datos pre-cargados. El campo "Usuario" contiene el texto "sipcbi" y el campo "Contraseña" contiene "\*\*\*\*\*". El botón "ACCEDER" está resaltado en azul y tiene un cursor de mouse encima.

---

1 Para la instalación del sistema revisar el Manual Técnico.

2 La url se la proporcionara el personal encargado de la instalación del sistema.

Imagen 2. Introducción de nombre de usuario y contraseña.



Imagen 3. Acceso al SIP-CBI.

---

### 3 Módulos que integran el SIP-CBI

---

Los módulos que integran al sistema son: usuarios, alumnos, docentes, posgrados y sistema.

- Módulo de usuarios: En este módulo se puede dar de alta y cambiar datos de usuarios (ver Imagen 4) para llevar a cabo la gestión del sistema. Los tipos de usuarios que se pueden dar de alta son:
  - Administrador tipo A, es aquel que tiene acceso total a los módulos y sus operaciones.
  - Administrador tipo B, es aquel que tiene acceso a los módulos alumnos, docentes y posgrados con todas sus operaciones.
  - Administrador tipo C, es aquel que tiene acceso a los módulos alumnos, docentes y posgrados, pero solo puede ver la información, no puede editar, dar de alta o baja de información.



Imagen 4. Módulo de usuarios.

- Módulo de alumnos: En este módulo se puede dar de alta, cambiar datos y consultar datos de alumnos (ver Imagen 5).



Imagen 5. Módulo de alumnos.

- Módulo de docentes: En este módulo se puede dar de alta, cambiar datos y consultar datos de docentes (ver Imagen 6).



Imagen 6. Módulo de docentes.

- Módulo de posgrados: En este módulo se puede dar de alta, cambiar datos y consultar datos de posgrados (ver Imagen 7).



Imagen 7. Módulo de posgrados.

- Módulo del sistema: En este módulo se divide en dos submódulos, estatus y trimestres (ver Imagen 8).



Imagen 8. Módulo del sistema.

- Submódulo de trimestres: En este submódulo se puede agregar, editar y consultar los trimestres (ver Imagen 9).

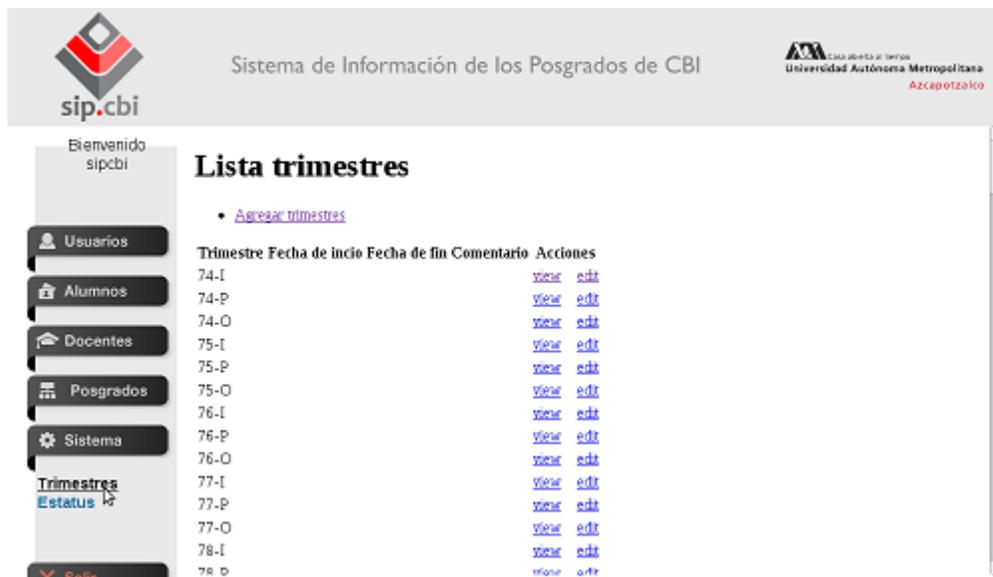


Imagen 9. Submódulo de trimestres.

- Submódulo de estatus: En este submódulo se puede agregar, editar, consultar y eliminar los estatus de los alumnos, docentes, posgrados, planes de estudios, usuario y trimestres (ver Imagen 10).

Bienvenido  
sipcbi

## Lista de Estatus

• [Agregar un estatus](#)

Tipo de Estatus	Clave	Estado	Acciones
usuari	inacti	Inactivo	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
usuari	activo	Activo	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
posgra	activo	Posgrado activo	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
posgra	inacti	Posgrado no activo	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
tests	proces	En proceso	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
tests	finali	Finalizada	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
posgra	proact	En proceso de aprobación	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
planes	activo	Activo	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
planes	inactivo	Inactivo	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
planes	proact	En proceso de aprobación	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
alumno	insact	Inscrito o activo	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
alumno	noein	No reinscrito	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
alumno	suspen	Suspendido	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>
alumno	haisda	Data definitiva	<a href="#">view</a> <a href="#">edit</a> <a href="#">delete</a>

Imagen 10. Submódulo de estatus.

---

## 4 Módulo de usuarios

---

En este capítulo nos enfocaremos en las operaciones dar de alta y cambiar datos de un usuario. Como antes se menciona existen tres tipo de usuarios, administrador A, B y C.

### 1. Alta de un usuario

- Primero seleccionamos la opción dar de alta y aparecerá el formulario (ver Imagen 11) para realizar la operación.



The image shows a web interface for the 'Sistema de Información de los Posgrados de CBI'. The page title is 'Alta de usuarios'. On the left, there is a navigation menu with options: 'Usuarios', 'Alta' (highlighted), 'Cambiar datos', 'Alumnos', 'Docentes', 'Posgrados', 'Sistema', and 'Salir'. The main content area contains a form with the following fields and controls:

- Nombre:** Input field for 'Nombre'.
- Estatus de usuario:** Dropdown menu with 'Activo' selected.
- Apellido paterno:** Input field.
- Apellido materno:** Input field.
- Correo electrónico:** Input field.
- Tipo de usuario:** Dropdown menu with '...' selected.
- Nombre de usuario:** Input field.
- Contraseña:** Input field.
- Repetir contraseña:** Input field.
- Fotografía:** File upload area with a 'Choose File' button and the text 'No file chosen'.
- Buttons:** 'Enviar' and 'Restablecer' buttons at the bottom of the form.

Imagen 11. Formulario para dar de alta un usuario.

- Ahora procederemos a rellenar el formulario (ver Imagen 12).

The screenshot shows the 'Alta de usuarios' (User Registration) form in the sip.cbi system. The form is filled with the following data:

- Nombre: Luis
- Estatus de usuario: Inactivo
- Apellido materno: Martínez
- Tipo de usuario: Administrador C
- Correo electrónico: Luis@gmail.com
- Nombre de usuario: Luis
- Contraseña: \*\*\*\*\*
- Repetir contraseña: \*\*\*\*\*
- Fotografía: No file chosen

The 'Enviar' button is visible at the bottom of the form.

Imagen 12. Formulario rellenado con los datos del usuario.

- Después de ver que los datos son válidos y presionar el botón "Enviar", el sistema nos dirá que el usuario fue dado de alta y nos dará la opción para dar de alta otro usuario (ver Imagen 13).

The screenshot shows the 'Alta de usuarios' page after successful registration. A yellow message bar at the top says "Se agregó un usuario nuevo". Below it, a green box labeled "Opciones:" contains a link "Agregar otro usuario".

Imagen 13. El usuario ha sido dado de alta correctamente.

## 2. Cambiar datos de un usuario

- Primero seleccionamos la opción cambiar datos (ver Imagen 14).



Imagen 14. Operación cambiar datos de usuarios.

- Después escribimos el nombre del usuario y presionamos el botón "enviar" (ver Imagen 15) . El sistema nos mostrará el resultado de la búsqueda (ver Imagen 16).



Imagen 15. Búsqueda de un usuario.

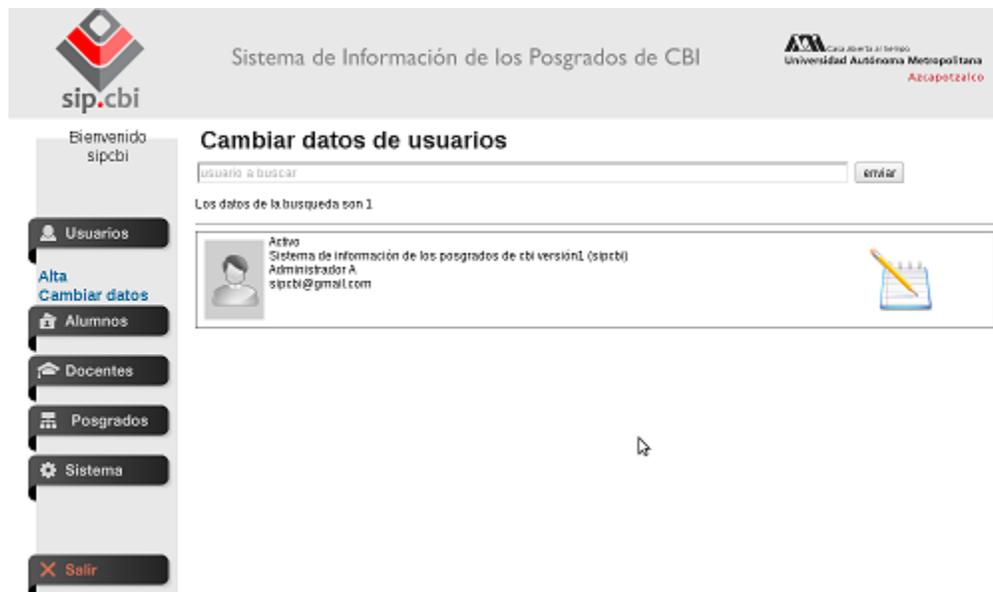


Imagen 16. Resultados de la búsqueda del usuario.

- Ahora procederemos presionar el ícono de editar y el sistema nos mostrará el formulario para editar la información del usuario (ver Imagen 17).

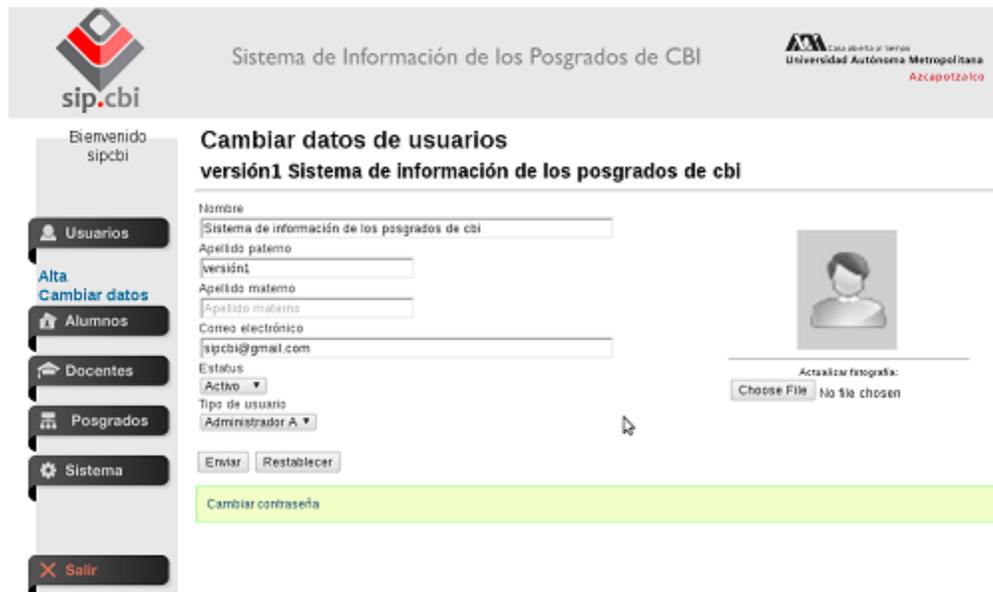


Imagen 17. Formulario para editar información de un usuario.

- También se puede editar la contraseña del usuario al presionar en "Cambiar contraseña" (ver Imagen 18) y el sistema nos mostrará el formulario para el cambio de contraseña (ver Imagen 19).

The screenshot shows the 'Cambiar datos de usuarios' page. The header includes the sip.cbi logo and the text 'Sistema de Información de los Posgrados de CBI' and 'Universidad Autónoma Metropolitana Azcapotzalco'. The main content area is titled 'Cambiar datos de usuarios versión1 Sistema de información de los posgrados de cbi'. It contains a form with the following fields: 'Nombre' (filled with 'Sistema de información de los posgrados de cbi'), 'Apellido paterno' (filled with 'versión1'), 'Apellido materno' (empty), 'Correo electrónico' (filled with 'sipcbi@gmail.com'), 'Estatus' (dropdown menu with 'Activo' selected), and 'Tipo de usuario' (dropdown menu with 'Administrador A.' selected). There are 'Enviar' and 'Restablecer' buttons. A 'Cambiar contraseña' button is highlighted in green at the bottom of the form. On the right, there is a placeholder for a profile picture with the text 'Actualizar fotografía: Choose File No file chosen'.

Imagen 18. Selección de la operación cambiar de contraseña.

The screenshot shows the 'Cambiar datos de usuarios' page with the 'Cambio de contraseña' form. The header is the same as in Imagen 18. The main content area is titled 'Cambiar datos de usuarios versión1 Sistema de información de los posgrados de cbi'. The form is titled 'Cambio de contraseña' and contains three input fields: 'Contraseña anterior', 'Contraseña nueva', and 'Vuelva a teclear su contraseña nueva'. There are 'Enviar' and 'Restablecer' buttons. The 'Cambiar contraseña' button from the previous image is no longer visible.

Imagen 19. Formulario para cambiar contraseña.

- Una vez realizados los cambios de información en el usuario y presionar el botón "Enviar", el sistema nos mostrará que los cambios se han realizado. También nos mostrará mas opciones

(ver Imagen 20).

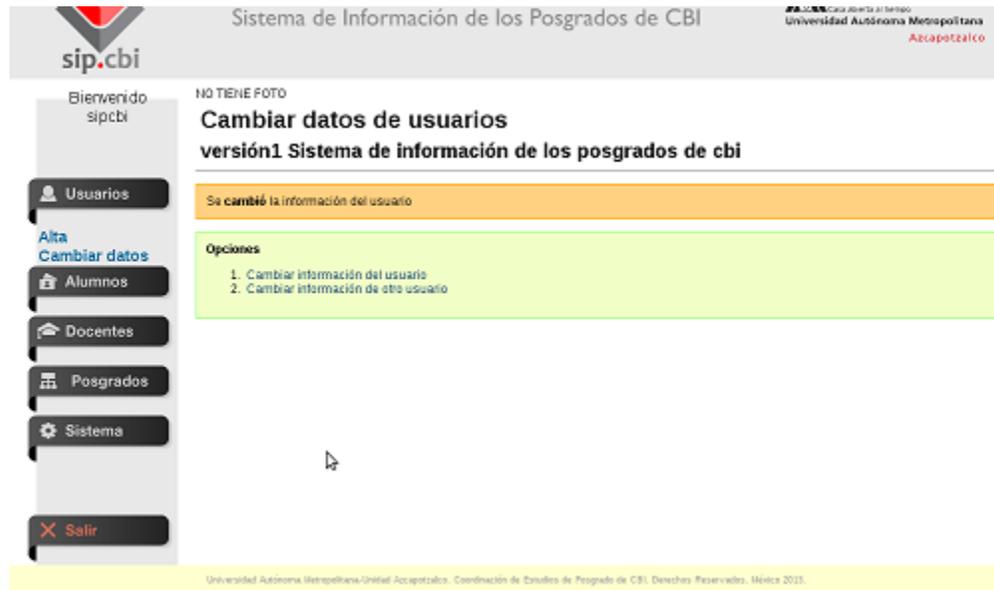


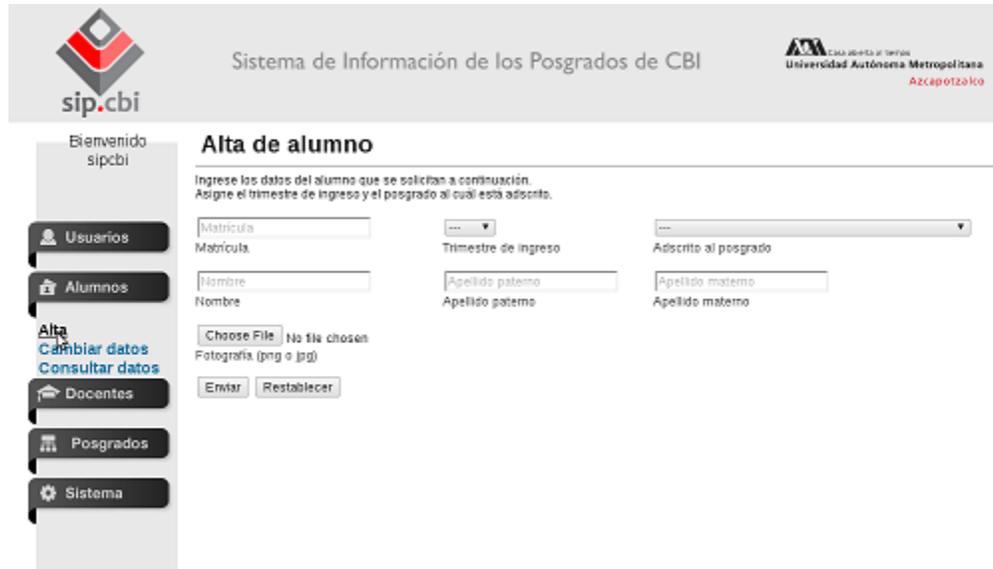
Imagen 20. Los datos han sido cambiados correctamente.

## 5 Módulo de alumnos

En este capítulo nos enfocaremos en las operaciones dar de alta, cambiar datos y consultar datos de un alumno.

### 1. Alta de un alumno

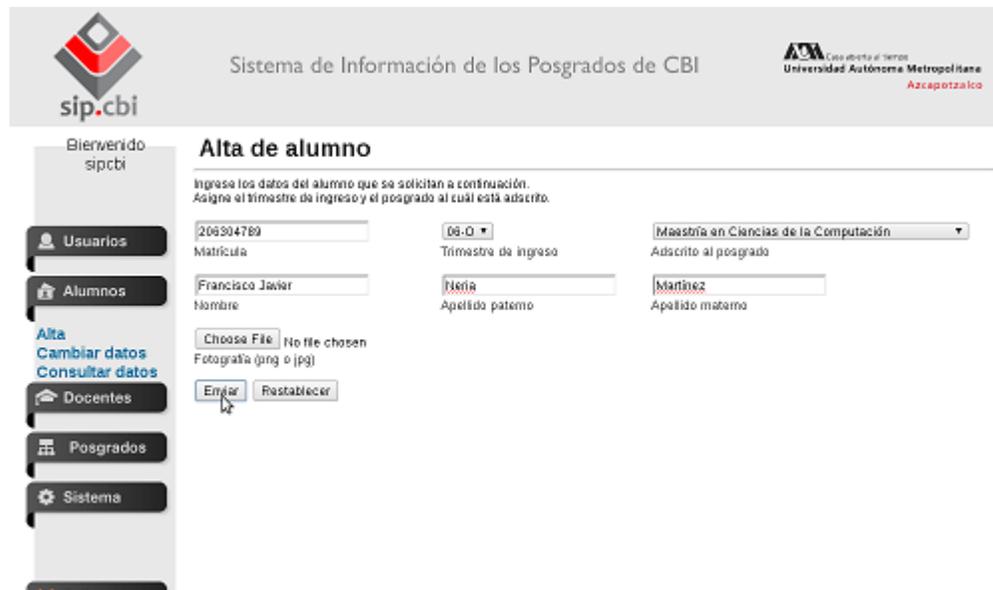
- Primero seleccionamos la opción dar de alta y aparecerá el formulario (ver Imagen 21) para realizar la operación.



The screenshot shows the 'Alta de alumno' form in the sip.cbi system. The form is empty, showing input fields for Matricula, Trimestre de ingreso, Adscrito al posgrado, Nombre, and Apellido paterno. A 'Enviar' button is visible at the bottom.

Imagen 21. Formulario para dar de alta un alumno.

- Ahora procederemos a rellenar el formulario (ver Imagen 22).



The screenshot shows the 'Alta de alumno' form in the sip.cbi system, now filled with data. The fields contain: Matricula: 204304789, Trimestre de ingreso: 06-0, Adscrito al posgrado: Maestría en Ciencias de la Computación, Nombre: Francisco Javier, Apellido paterno: Heria, Apellido materno: Martínez. A mouse cursor is over the 'Enviar' button.

Imagen 22. Formulario rellenado con los datos del alumno.

- Después de ver que los datos son válidos y presionar el botón "Enviar", el sistema nos dirá que el alumno fue dado de alta y nos dará la opción para dar de alta otro alumno (ver Imagen 23).

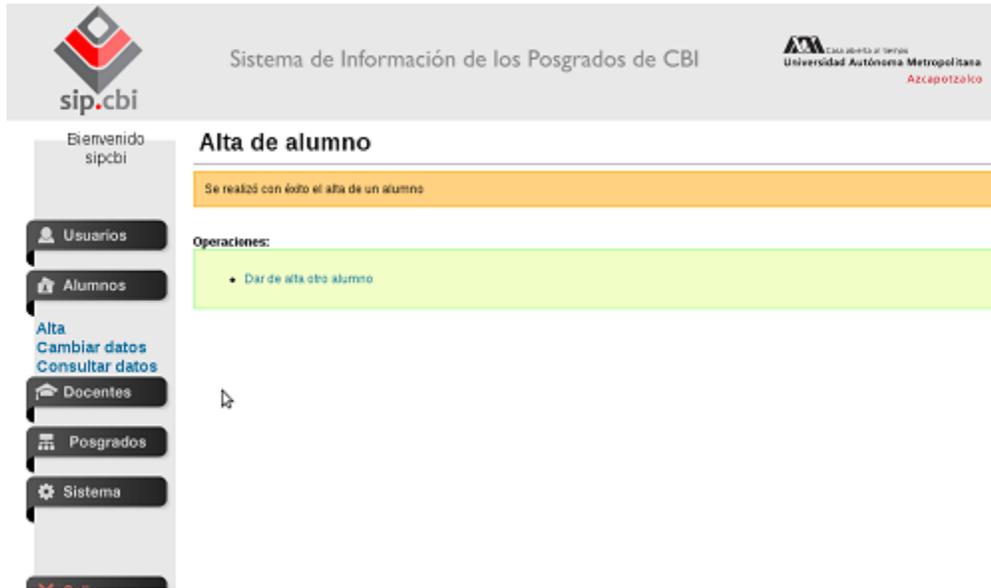


Imagen 23. El usuario ha sido dado de alta correctamente.

## 2. Cambiar datos de un alumno

- Primero seleccionamos la opción cambiar datos (ver Imagen 24).



Imagen 24. Operación cambiar datos de usuarios.

- Después comenzamos a escribir el nombre o matrícula del alumno e inmediatamente el sistema dinámicamente nos mostrará una lista de alumnos que contengan dicho nombre o matrícula (ver Imagen 25).



Imagen 25. Búsqueda dinámica de un alumno.

- Ahora procederemos a presionar el ícono de la libreta con el lápiz (ver Imagen 26) y el sistema nos mostrará un conjunto de opciones para editar los datos del alumno (ver Imagen 27).

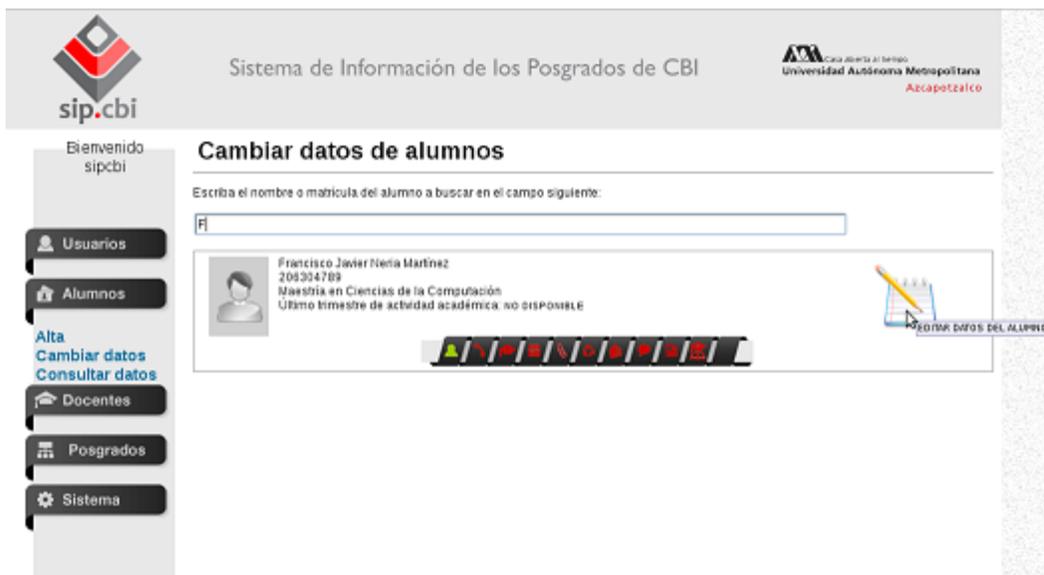


Imagen 26. Selección de la opción editar datos del alumno.



Imagen 27. Listado de opciones a editar del alumno (datos personales, datos adicionales, etc).

- Seleccionamos la opción datos personales y el sistema nos mostrará un formulario para editar la información del alumno (ver Imagen 28).

Bienvenido sipcbi

ALUMNOS • MODIFICAR OTROS DATOS DEL ALUMNO

**Cambiar datos** **Neria Martínez Francisco Javier**  
**Datos personales** **Maestría en Ciencias de la Computación**

matrícula: 206304789 | Timestre de ingreso: 06 | Maestría en Ciencias de la Computación

nombre: Francisco | apellido paterno: Neria | apellido materno: Martínez

Enviar Restablecer

Actualizar fotografía  
 Choose File No file chosen  
 Enviar

Imagen 28. Formulario para editar los datos personales del alumno.

- Una vez realizados los cambios de información en el alumno y presionar en el botón "Enviar", el sistema nos mostrará que los cambios se han realizado. También nos mostrará mas opciones (ver Imagen 29).

Bienvenido sipcbi

ALUMNOS • MODIFICAR OTROS DATOS DEL ALUMNO

**Cambiar datos** **Neria Martínez Francisco Javier**  
**Datos personales** **Maestría en Ciencias de la Computación**

Los datos se cambiaron correctamente.

**Opciones**

- Modificar datos personales
- Buscar alumno

Imagen 29. Los datos han sido cambiados correctamente.

### 3. Consultar datos de un alumno

- Seleccionamos la opción consultar datos (ver Imagen 30).



Imagen 30. Operación consultar datos de un alumno.

- Después comenzamos a escribir el nombre o matrícula del alumno e inmediatamente el sistema dinámicamente nos mostrará una lista de alumnos que contengan dicho nombre o matrícula (ver Imagen 31).

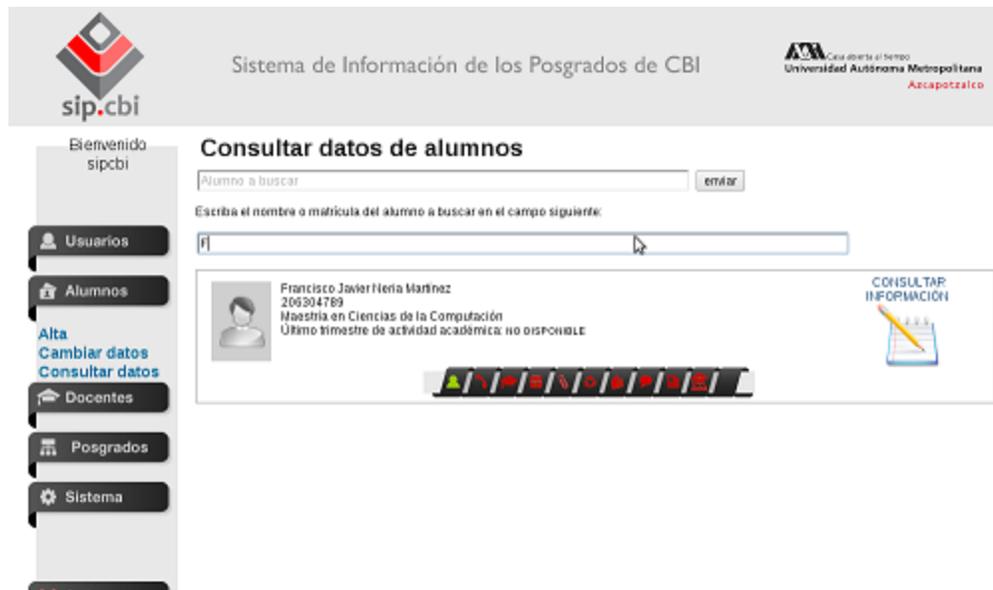


Imagen 31. Búsqueda dinámica de un alumno.

- Ahora procederemos a presionar el ícono de consultar información (ver Imagen 32) y el sistema nos mostrará toda la información del alumno (datos personales, datos adicionales, etc.)

contenida dentro del sistema (ver Imagen 33).

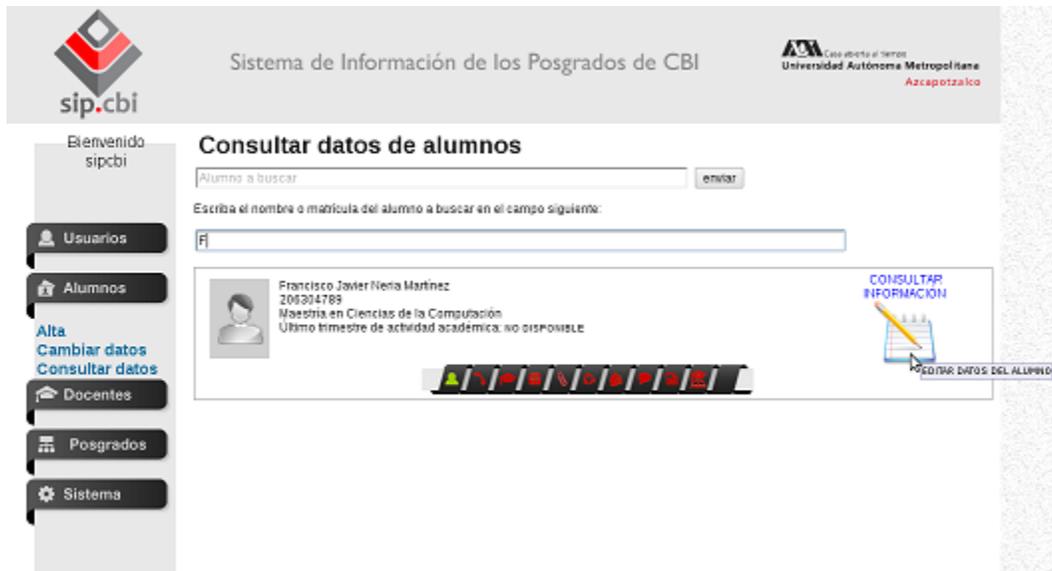


Imagen 32. Selección de la opción editar datos del alumno.



Imagen 33. Consulta de información del alumno.

---

## 6 Finalizar sesión

---

Para cerrar la sesión una vez que se haya terminado la gestión de usuarios, alumnos, docentes, trimestres, estatus, posgrados y/o planes de estudios, según corresponda el tipo de usuario (administrador A, B o C), se debe presionar el botón "Salir" (ver Imagen 34).



Imagen 34. Fin de sesión del usuario.