

# UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

Facultad de Contaduría y Administración

Facultad de Ciencias Químicas e Ingeniería

Maestría en Tecnologías de la Información y la Comunicación



## **Implementación del Sistema de Gestión Académica en la Facultad de Contaduría y Administración (SIGAF), UABC.**

PROYECTO DE TESIS

PARA OBTENER EL GRADO DE  
MAESTRO EN TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN

*Presenta:*

**Raúl Súcilt Fonseca**

Bajo la dirección de:  
**Dra. María del Consuelo Salgado Soto**

Co-dirección:  
**Dra. Hilda Beatriz Ramírez Moreno**

# Mi dedicatoria

Con todo mi cariño y mi amor para las personas que hicieron todo en la vida para que yo pudiera lograr mis sueños, por creer en mí y darme la mano cuando sentía que el camino se terminaba, a ustedes por siempre mi corazón y mi agradecimiento.

Papá y mamá

# Agradecimientos

Me gustaría que estas líneas sirvieran para expresar mi agradecimiento a todas aquellas personas que con su ayuda han colaborado en la realización del presente trabajo, en especial a la Dra. María del Consuelo Salgado Soto, directora de este trabajo de tesis, por la orientación, el seguimiento y la supervisión continúa de la misma, por el apoyo recibido a lo largo de estos años.

A mi familia fuente de apoyo constante e incondicional en toda mi vida y más aún en mis duros años de estudio y en especial quiero expresar mi más grande agradecimiento a mis padres que sin su ayuda hubiera sido imposible culminar mi posgrado.

Quisiera hacer extensiva mi gratitud a mis compañeros y catedráticos de la maestría por su amistad y colaboración.

Un agradecimiento muy especial merece la comprensión, paciencia y el ánimo recibidos de mi familia y amigos.

A todos ellos, muchas gracias.

# Resumen

El proyecto de tesis fue desarrollado en base a un problema existente originado por la necesidad de optimizar y automatizar la creación de los horarios de clase, grupos y distribución de las materias de acuerdo a la disponibilidad y al rendimiento académico de los docentes en la Facultad de Contaduría y Administración de la Universidad Autónoma de Baja California.

Para efectos de esta tesis el enfoque será sobre la implementación del módulo de plan de estudio, carga académica, disponibilidad docente y usuarios del Sistema de Gestión Académica FCA (SIGAF) en la Facultad de Contaduría y Administración en la Universidad Autónoma de Baja California.

El proceso de implementación de SIGAF abarca las preparaciones necesarias para que el sistema funcione, dichas preparaciones consisten en instalar y configurar el software requerido para que el servidor funcione con una arquitectura Web. Otro aspecto de la implementación es la incorporación del sistema al entorno de operación, durante esta actividad se instala el sistema de gestión al servidor que ha sido previamente preparado para que el sistema funcione. De igual manera, las pruebas constituyen otra fase de la implementación ya que permiten verificar el correcto funcionamiento del sistema en el entorno de operación.

# Abstract

The thesis project was developed based on an existing problem caused by the need to optimize and automate the creation of class schedules, groups and distribution of subjects according to the availability and academic performance of teachers in the School of Accounting and Administration at the Autonomous University of Baja California.

For purposes of this thesis will focus on the implementation of the curriculum module, academic load, teaching users availability of Academic Management System FCA (SIGAF) in the School of Accounting and Administration at the Autonomous University of Baja California.

The implementation process includes the necessary preparations to make the system work, such preparations consist of installing and configuring the software required to operate the server with a Web architecture. Another aspect of implementation is the incorporation of the system to the operating environment, during the activity the management system is installed at the server that has been previously prepared for the system to work. Likewise, tests are another phase of implementation as possible to verify proper operation of the system in the operating environment.

# Índice general

	Página
<i>Capítulo I Introducción</i>	11
<b>1.1 Antecedentes</b>	13
<b>1.2 Casos relacionados</b>	15
1.2.1 Universidad de las Ciencias Informáticas	15
1.2.2 Escuela Politécnica Nacional	16
<b>1.3 Definición del problema</b>	17
1.3.1 Situación actual	17
1.3.2 Definición del problema	18
<b>1.4 Justificación</b>	19
<b>1.5 Objetivos</b>	21
1.5.1 General	21
1.5.2 Específicos	21
<b>1.6 Alcance y limitaciones de la Investigación</b>	22
1.6.1 Alcance	22
1.6.2 Limitaciones	22
<i>Capítulo II Marco Teórico</i>	23
<b>2.1 Teoría de sistemas</b>	24
2.1.1 Sujetos que intervienen en un sistema de información.	24
<b>2.2 Metodologías de desarrollo</b>	25
2.2.1 FAST	25
2.2.2 Cascada	26
2.3.3 Scrum	27
2.2.4 Métrica V3	28
<b>2.3 Arquitectura Software del sistema</b>	29
2.3.1 Sistema Operativo	29
2.3.2 Servidor Web Apache	32
2.3.3 PHP	33
2.3.4 MySQL	34
2.3.5 FTP	34
<b>2.4 Arquitectura Hardware del sistema</b>	35
2.4.1 Servidor	35

	<b>Página</b>
<i>Capítulo III Metodología</i>	37
<b>3. Metodología</b>	38
<b>3.1 Modelo de trabajo Scrum</b>	38
<b>3.2 Metodología MÉTRICA V3</b>	40
<b>3.3 Implementación</b>	40
3.3.1 Establecimiento del plan de implantación	41
3.3.2 Formación necesaria para la implantación	42
3.3.3 Incorporación del sistema al entorno de operación	42
3.3.3.1 Preparación de la instalación	42
3.3.3.2 Realización de la instalación	43
3.3.3.3 Pruebas unitarias	43
3.3.4 Carga de datos al entorno de operación	44
 <i>Capítulo IV Desarrollo</i>	 45
<b>4.1 Plan de implantación</b>	46
<b>4.2 Formación necesaria para la implantación</b>	46
<b>4.3 Incorporación del sistema al entorno de operación</b>	46
4.3.1 Preparación de la instalación	47
4.3.1.1 Instalación del sistema operativo	47
4.3.1.2 Instalación del servidor Web	49
4.3.1.3 Instalación del motor de base de datos MySQL	50
4.3.1.4 Instalación de PHP	50
4.3.2 Realización de la instalación.	52
4.3.2.1 Incorporación de la base de datos	52
4.3.2.2 Servidor HTTP	53
4.3.2.3 Implementación del sistema.	55
4.3.2.3.1 Módulo uno: plan de Estudios.	55
4.3.2.3.2 Módulo dos: carga académica.	57
4.3.3 Pruebas unitarias del sistema.	59
4.3.3.1 Plan de pruebas	61
4.3.3.2 Diseño de los casos de uso	63
4.3.3.3 Ejecución de las pruebas	92
4.3.4 Ajustes en el entorno de operación	92
<b>4.4 Carga de datos al entorno de operación</b>	94

	<b>Página</b>
<i>Capítulo V Resultados y Conclusiones</i>	98
<b>5.1 Resultados</b>	99
<b>5.2 Conclusiones</b>	107
<b>5.3 Recomendaciones</b>	110
<i>Anexos</i>	112
Anexo 1	113
Anexo 2	126
<i>Bibliografía</i>	130

# Índice de figuras

	<b>Página</b>
<b>Figura 3.1.</b> Modelo Scrum.	38
<b>Figura 3.2.</b> Proceso de implantación.	41
<b>Figura 4.1.</b> Menú de instalación.	48
<b>Figura 4.2.</b> Instalación completa.	48
<b>Figura 4.3.</b> CentOS Login.	48
<b>Figura 4.4.</b> Instalación del servidor HTTP.	49
<b>Figura 4.5.</b> Instalación de MySQL.	50
<b>Figura 4.6.</b> Instalación de PHP.	51
<b>Figura 4.7.</b> Verificar motor de base de datos.	52
<b>Figura 4.8.</b> Crear base de datos.	53
<b>Figura 4.9.</b> Cortafuego (HTTP).	54
<b>Figura 4.10.</b> Acceso al servidor.	55
<b>Figura 4.11.</b> Instalación de SIGAF.	55
<b>Figura 4.12.</b> Login.	56
<b>Figura 4.13.</b> Interfaz del sistema.	56
<b>Figura 4.14.</b> Ruta de los sistemas.	57
<b>Figura 4.15.</b> Fichero index.	58
<b>Figura 4.16.</b> Rutas.	64
<b>Figura 4.17.</b> Registra escenario 1.	65
<b>Figura 4.18.</b> Registra escenario 2.	66
<b>Figura 4.19.</b> Registra escenario 3.	67
<b>Figura 4.20.</b> Registra escenario 4.	68
<b>Figura 4.21.</b> Consultar unidad de aprendizaje escenario 1.	70
<b>Figura 4.22.</b> Consultar unidad de aprendizaje escenario 2.	71
<b>Figura 4.23.</b> Consultar unidad de aprendizaje escenario 3.	72
<b>Figura 4.24.</b> Consultar unidad de aprendizaje escenario 4.	73
<b>Figura 4.25.</b> Consultar unidad de aprendizaje escenario 5.	74
<b>Figura 4.26.</b> Consultar unidad de aprendizaje escenario 6.	75
<b>Figura 4.27.</b> Consultar unidad de aprendizaje escenario 7.	76
<b>Figura 4.28.</b> Consulta personalizada UA escenario 1.	78
<b>Figura 4.29.</b> Eliminar UA escenario 1.	80
<b>Figura 4.30.</b> Eliminar UA escenario 2.	81
<b>Figura 4.31.</b> Eliminar UA escenario 3.	82
<b>Figura 4.32.</b> Eliminar UA escenario 4.	83
<b>Figura 5.1.</b> Mi-ip.cl.	100
<b>Figura 5.2.</b> Conexión externa.	102
<b>Figura 5.3.</b> Conexión mediante SSH.	101
<b>Figura 5.4.</b> Acceso módulo log In	102
<b>Figura 5.5.</b> Acceso módulo plan de estudios	103
<b>Figura 5.6.</b> Acceso módulo plan de estudios II	103
<b>Figura 5.7.</b> Acceso módulo carga académica	104
<b>Figura 5.8.</b> Acceso módulo carga académica II	104
<b>Figura 5.9.</b> Acceso módulo disponibilidad docente	105
<b>Figura 5.10.</b> Acceso módulo disponibilidad docente II	106
<b>Figura 5.11.</b> Acceso módulo usuarios	106

# Índice de tablas

	<b>Página</b>
<b>Tabla 2.1.</b> Fases de la metodología FAST.	25
<b>Tabla 2.2.</b> Especificaciones del servidor FCA.	36
<b>Tabla 4.1.</b> Especificaciones técnicas del servidor.	47
<b>Tabla 4.2.</b> Plan de pruebas.	59
<b>Tabla 4.3.</b> Ejecución de pruebas.	60
<b>Tabla 4.4.</b> Clase de equivalencias.	62
<b>Tabla 4.5.</b> Registrar nueva unidad de aprendizaje.	63
<b>Tabla 4.6.</b> Registrar escenario 1.	65
<b>Tabla 4.7.</b> Registrar escenario 2.	66
<b>Tabla 4.8.</b> Registrar escenario 3.	67
<b>Tabla 4.9.</b> Registrar escenario 4.	68
<b>Tabla 4.10.</b> Consultar nueva unidad de aprendizaje.	69
<b>Tabla 4.11.</b> Consultar unidad de aprendizaje escenario 1.	70
<b>Tabla 4.12.</b> Consultar unidad de aprendizaje escenario 2.	71
<b>Tabla 4.13.</b> Consultar unidad de aprendizaje escenario 3.	72
<b>Tabla 4.14.</b> Consultar unidad de aprendizaje escenario 4.	73
<b>Tabla 4.15.</b> Consultar unidad de aprendizaje escenario 5.	74
<b>Tabla 4.16.</b> Consultar unidad de aprendizaje escenario 6.	75
<b>Tabla 4.17.</b> Consultar unidad de aprendizaje escenario 7.	76
<b>Tabla 4.18.</b> Consulta personalizada	77
<b>Tabla 4.19.</b> Consulta personalizada UA escenario 1.	78
<b>Tabla 4.20.</b> Eliminar unidad de aprendizaje.	79
<b>Tabla 4.21.</b> Eliminar UA escenario 1.	80
<b>Tabla 4.22.</b> Eliminar UA escenario 2.	81
<b>Tabla 4.23.</b> Eliminar UA escenario 3.	82
<b>Tabla 4.24.</b> Eliminar UA escenario 4.	83
<b>Tabla 4.25.</b> Modificar unidad de aprendizaje	84
<b>Tabla 4.26.</b> Registrar primera carga académica.	85
<b>Tabla 4.27.</b> Registrar segunda carga académica.	86
<b>Tabla 4.28.</b> Consultar carga académica.	87
<b>Tabla 4.29.</b> Modificar carga académica.	88
<b>Tabla 4.30.</b> Eliminar carga académica.	89
<b>Tabla 4.31.</b> Ejecución de pruebas.	90
<b>Tabla 4.32.</b> Ejecución de pruebas carga académica.	91

# *Capítulo I*

## *Introducción*

El desarrollo de este proyecto tiene como finalidad presentar el procedimiento de implantación del Sistema de Gestión de Académica FCA, sistema que facilita, optimiza y automatiza el proceso de desarrollo de tareas administrativas, contemplando entre ellas lo que es el generar los planes de estudio y la carga académica de acuerdo a las necesidades y requisitos de la Facultad de Contaduría y Administración, campus Tijuana.

El punto focal del desarrollo del proyecto es la implementación de un sistema capaz de generar los horarios, sin embargo, para llevar a cabo esta tarea se necesita conocer los requisitos que solicita el cliente y los recursos que se tienen a la mano, por otro lado, para la fase de la carga de datos al sistema se recaba información por medio de los usuarios que servirá como base para hacer cada horario.

A la vez, antes de llegar a la fase final de la implementación se debe comprobar que el sistema funciona conforme a los requisitos previamente establecidos, esto se hace mediante una serie de pruebas que permiten detectar errores en etapas tempranas del desarrollo de proyecto para ser corregidos y por analizar el funcionamiento del sistema en el entorno de operación.

La implementación del sistema supone grandes cambios en el proceso de generar y gestionar los horarios. En la medida que los cambios se establecen como predeterminados, la adaptabilidad de todos los usuarios se vuelve un factor determinante a la hora de implementar el sistema.

## **1.1 Antecedentes**

Un proceso común en todas las organizaciones educativas ha sido la elaboración de horarios, ya que permiten agrupar y distribuir la carga de trabajo de los docentes, al mismo tiempo el coordinador académico, el subdirector y el director puede tomar decisiones sobre la distribución de las materias con sus respectivas horas clase y si es necesario contratar nuevos maestros. Las personas que realizan este proceso se han valido de distintas herramientas para cumplir con el mismo. La elaboración de horarios puede ser una de las tareas que llega a ser de las más tardadas, sobre todo cuando la comunidad en la institución donde se realiza este proceso es extensa, tanto de alumnos como docentes.

Hoy en día las instituciones educativas dependen de esta tarea para definir la carga de trabajo de los docentes, tratando lo mejor posible, de que no existan horas libres entre cada clase y evitar los choques de horas entre otros maestros, basados a la vez, en la disponibilidad elaborada por el docente. Con lo anterior se puede apreciar que este proceso es complejo, sin embargo, otras instituciones han trabajado en situaciones similares, tal es la Universidad de las Ciencias Informáticas que lograron desarrollar un prototipo funcional llevado a cabo por Valdés Acosta y Guardia Macías (2007), y la Escuela Politécnica Nacional presentado por Alomoto Cuvi y Morales Hernández (2008) que lograron desarrollar e implementar otro sistema de esta naturaleza.

La Universidad de las Ciencias Informáticas realizó las siguientes tareas para desarrollar su proyecto:

1. Estudio del arte del desarrollo de aplicaciones Web, la inteligencia artificial, el diseño y desarrollo de bases de datos.
2. Confección de la documentación completa UML del sistema propuesto.
3. Desarrollo de la base de datos de los horarios docentes de la Facultad 9.
4. Desarrollo de un algoritmo basado en inteligencia artificial capaz de confeccionar los horarios docentes.
5. Desarrollo de un prototipo funcional del Sistema Automatizado para la Gestión de Horarios Docentes.

A lo largo del tiempo han sido numerosos los esfuerzos por dar solución a esta problemática derivada de la generación de horarios que a la vez necesita de la realización de otros procesos tales como la elaboración de una carga académica, plan de estudios y la necesidad de tener información a la mano sobre la disponibilidad del maestro y su capacidad de desempeño durante el ciclo escolar.

## **1.2 Casos relacionados**

### **1.2.1 Universidad de las Ciencias Informáticas**

En la Facultad nueve de la Universidad de las Ciencias Informáticas (UCI) en ciudad de la Habana, se presentó un sistema automatizado para la gestión de horarios docentes desarrollado por Valdés Acosta y Guardia Macías (2007). El trabajo consistió en la creación de un sistema automatizado para la gestión de los horarios docentes. El objeto de estudio de esta investigación fue el proceso de elaboración de los horarios docentes y el campo de acción fue la automatización de la gestión de los horarios docentes en la Facultad nueve de la Universidad de las Ciencias Informáticas. El objetivo general del trabajo consistió en proponer un prototipo funcional del Sistema Automatizado para la Gestión de Horarios Docentes.

En este proyecto se obtuvieron los siguientes resultados:

- Se modelaron los procesos que tienen lugar durante la confección de los horarios docentes.
- Se desarrolló un prototipo funcional del Sistema Automatizado para la Gestión de Horarios Docentes.
- Se desarrolló un algoritmo basado en inteligencia artificial capaz de generar automáticamente los horarios docentes.

### **1.2.2 Escuela Politécnica Nacional**

Se desarrolló otro proyecto en la escuela Politécnica Nacional consistente en un sistema de gestión de horarios en las aulas de la Facultad de Ingeniería de Sistemas Informáticos y de Computación (FIS). El trabajo fue presentado por Alomoto Cuvi y Morales Hernández (2008), cuyo objetivo consistió en entregar a dicha facultad un sistema que permita agilizar el proceso de asignación de horarios de acuerdo a las necesidades y especificaciones del decano. El sistema crea un nuevo plan de estudios de acuerdo a las necesidades vistas por las autoridades y la creación de un nuevo período cada inicio de un nuevo semestre, una característica de este sistema es que muestra la información de manera gráfica, amigable y permite almacenar la carga académica por aula, por profesor y por semestre.

Los resultados obtenidos de este proyecto fueron los siguientes:

- Se implantó el sistema en el período Septiembre 2008 – Febrero 2009, y se utilizó para la publicación de los horarios en las aulas de la FIS.
- Al finalizar el desarrollo de la solución planteada se encontraron con funcionalidades que en el sistema no fueron desarrolladas.
- La solución planteada disminuyó el tiempo y esfuerzo que eran dedicados al proceso de la gestión de horarios y de la asignación de aulas.
- La metodología utilizada permitió definir los documentos y realizar cambios durante el desarrollo del sistema.

## **1.3 Definición del problema**

### **1.3.1 Situación actual**

Actualmente la Facultad de Contaduría y Administración al presentar un crecimiento a lo largo de los años, dedica mucho tiempo para elaborar los horarios. Tres meses antes del inicio de cada ciclo escolar el analista del área de coordinación académica elabora la carga académica, el personal administrativo entrega el formato de disponibilidad a los docentes y los docentes posteriormente, una vez llenada su solicitud, entregan de regreso su disponibilidad para que la asignación de grupos, la distribución de salones y los horarios comiencen a ser generados, por lo anterior se entiende que es un proceso largo y por lo tanto tardado.

La Facultad cuenta con un software que se utiliza como herramienta para este proceso, es un software suplementario, ya que no realiza todo el proceso de forma automatizada. El proceso de registro de disponibilidad de horarios lo hace el maestro manualmente, mediante un formato que es proporcionado por las secretarías, una vez que el formato es llenado con la información necesaria, se lo entrega al personal administrativo. La entrega lo puede hacer físicamente o lo envía por correo. En el formato se indica las materias que desea impartir, el horario disponible, la fecha y otra información que sirve al momento de la toma de decisiones por parte del sub-director y el personal administrativo.

Para la asignación de salón de los grupos, el personal administrativo lo hace mediante una hoja electrónica, extrae los datos de otros formatos tales como grupo, semestre cursado, carrera, salón y número de alumnos.

### **1.3.2 Definición del problema**

En la Facultad de Contaduría y Administración existe un software para la elaboración de horarios que no es el óptimo para sistematizar o automatizar este proceso, al igual que la creación de grupos, asignación de salones y distribuir las materias a los docentes, debido a esto, no es posible almacenar de la información precisa del docente y el personal administrativo de la facultad no tiene la facilidad para generar informes y gráficos para ser entregados de forma oportuna al área de dirección para la toma de decisiones.

Como consecuencia el software que se utiliza actualmente en la Facultad de Contaduría y Administración, no cuenta con documentación alguna sobre métodos de pruebas ni implementación por lo que se tiene que generar una guía desde cero para implementar y realizar las pruebas del nuevo sistema para agilizar el proceso de generación de horarios.

## **1.4 Justificación**

El uso de tecnologías para llevar a cabo cualquier proceso es más común hoy en día, en los últimos años ha incrementado el número de software libre creado para múltiples tareas y con la expansión de Internet se ha vuelto cada vez es más accesible. Con la implementación adecuada, es posible la automatización de procesos, desde los más simples hasta el más robusto como lo es un sistema de creación de horarios.

El área de dirección de la Facultad de Contaduría y Administración (FCA) de la institución educativa UABC, desea tener un mejor control de las actividades que se realizan a la hora de generar los horarios de clase. Por otro lado, necesitan que la solicitud de los informes del proceso de creación de horarios, asignación de materias a los docentes o el progreso de la distribución de salones, tenga una respuesta inmediata, sin tener que esperar semanas para tener a la mano dicha información.

La facultad necesita implementar un sistema con una única base de datos que alimente con información a todos los usuarios que lo requieren, de esta manera se mejora el rendimiento del uso de los datos ya que para acceder a ellos se deja de usar una serie de programas aislados y se utiliza sólo un sistema robusto centralizado.

Existen sistemas de este tipo, sin embargo, sus funciones son limitadas de acuerdo a las necesidades de la FCA. Los sistemas actuales no cumplen con las expectativas de la facultad, además de no estar de acuerdo con la idea de utilizar sistemas que tengan más o menos funciones de las que necesitan o que sea de

difícil manejo, por lo cual se propone la implementación del Sistema de Gestión Académica FCA, sistema desarrollado por la misma facultad a la medida, fácil e intuitivo a la hora de utilizar.

La facultad necesita el procedimiento de implementación, el documento que se genere de esta actividad contendrá las configuraciones necesarias para que el sistema funcione y pueda ser incorporado en otros entornos de operación, además, dentro de esta fase se comprueba que el sistema funcione de acuerdo a las especificaciones que la institución solicitó.

## **1.5 Objetivos**

### **1.5.1 General**

Implementación y pruebas del módulo de plan de estudio, carga académica, disponibilidad docente y usuarios del Sistema de Gestión Académica FCA (SIGAF) en la Facultad de Contaduría y Administración UABC, campus Tijuana.

### **1.5.2 Específicos**

- Elaborar un procedimiento de implementación para el SIGAF.
- Realizar las pruebas de los módulos de SIGAF.
- Implementar los módulos de SIGAF basándose en los requerimientos de la organización educativa.

## **1.6 Alcance y limitaciones de la Investigación**

### **1.6.1 Alcance**

En este trabajo se presentará el procedimiento que permite la implementación de los módulos del Sistema de Gestión Académica FCA, un sistema que utiliza un interfaz Web y el motor de bases de datos MySQL.

- Para la verificación del funcionamiento del sistema se realizan pruebas que permitan comprobar que cumple con los requerimientos establecidos.
- El procedimiento de implementación que se propone en el presente trabajo es aplicable únicamente a esta facultad.
- Los usuarios finales del sistema tendrán acceso al sistema desde computadoras conectadas a la red local de la facultad.

### **1.6.2 Limitaciones**

- Este procedimiento parte del hecho de que la Facultad de Contaduría y Administración haya aprobado el desarrollo de Sistema de Gestión Académica y que este cumpla con la funcionalidad y requerimientos de la misma.
- El equipo desconoce la cifra exacta de recursos con los que cuenta por lo que se debe analizar la infraestructura actual de la facultad.
- Para la instalación del servidor se utilizara únicamente software libre.

# *Capítulo II*

## *Marco Teórico*

## **2.1 Teoría de sistemas**

Un sistema es “un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar la toma de decisiones y el control de una organización”. (Laudon y Laudon, 2004).

Los sistemas de información permiten a las organizaciones capturar y administrar datos con el fin de obtener información útil.

### **2.1.1 Sujetos que intervienen en un sistema de información.**

Durante el desarrollo de un sistema de información intervienen varios sujetos que realizan tareas distintas para dar por cumplido el mismo, los participantes en este proceso son los siguientes: el propietario de sistemas que atiende a cuestiones sobre el costo que tendrá el sistema o cuales son los beneficios de incorporar un sistema al negocio; los usuarios del sistema interesados en que el software funcione, les facilite su trabajo y que sea fácil de manejar; los diseñadores de sistemas que se encargan de elegir la tecnología adecuado para construir sistemas más actuales y funcionales; los constructores de sistemas que desarrollan los sistemas de acuerdo a las especificaciones establecidas por los diseñadores; los analistas de sistemas que son los que entienden el negocio buscan los problemas y oportunidades para transformarlos en requerimientos; el administrador del proyecto que se encarga de que las tareas se ejecuten para que el proyecto se desarrolle a tiempo.

## 2.2 Metodologías de desarrollo

De acuerdo a Whitten, Bentley y Dittman (2004), las fases del desarrollo de software son las siguientes: inicio del sistema, análisis, diseño e implantación. En la fase de inicio del sistema los propietarios del sistema aportan los recursos para que se desarrolle el sistema y lo pongan en funcionamiento. Durante el análisis se definen los requerimientos en base a las necesidades y funcionalidades solicitadas por los usuarios del sistema. El diseño del sistema consiste en la transformación de los requerimientos en una solución técnica que sea posible de realizar. En la última fase se desarrolla, se implanta y se mantiene el sistema, se toma en cuenta tanto el software como el hardware durante la implantación. Existen diversas metodologías que se emplean en el desarrollo de software, entre ellas se encuentra la metodología fast, cascada, scrum y métrica 3.

### 2.2.1 FAST (Framework for the Application of Systems Thinking)

FAST es una metodología compuesta por ocho fases que permiten definir con mayor claridad los hitos periódicos y los productos. En la tabla 2.1 se muestra la comparación entre las fases de la metodología FAST y la clásica de acuerdo a Whitten, Bentley y Dittman (2004).

**Tabla 2.1.** Fases de la metodología FAST.  
Fuente: Whitten, Bentley y Dittman (2004)

Fases FAST	Fases Clásicas
Definición del alcance	Inicio de proyecto
Análisis del problema	Análisis del sistema
Análisis de requerimientos	
Diseño lógico	
Análisis de decisión	Diseño del sistema
Diseño físico e integración	
Construcción y pruebas	Implantación del sistema
Instalación y entrega	

### 2.2.2 Cascada

En ingeniería de software el desarrollo en cascada, denominado así por la posición de las fases en el desarrollo de esta, que parecen caer en cascada “por gravedad” hacia las siguientes fases, es el enfoque metodológico que ordena rigurosamente las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior (Pressman, Roger S. 1997).

Las fases en las que se divide este modelo de desarrollo son las siguientes:

- Análisis de requisitos.
- Diseño del Sistema.
- Diseño del Programa.
- Codificación.
- Pruebas.
- Implantación.
- Mantenimiento.

De acuerdo al modelo, la etapa de pruebas procede una vez que se ha generado el sistema y existe una versión funcional del mismo. Durante las pruebas se verifica la lógica interna del software, y las funciones externas, que aseguren que la entrada definida produce los resultados que realmente se requieren.

Los errores en el sistema pueden ser generados por distintas causas ya sea porque el software no hace algo que está definido en las especificaciones o porque es difícil de entender, utilizar o lento (Ron Patton, 2006).

Así mismo, para llevar a cabo las pruebas se debe elaborar un plan basado en los casos de uso.

Un caso de uso se trata de una secuencia de acciones realizadas por el sistema, que producen un resultado para un usuario en particular, es decir, representa el comportamiento del sistema, mostrando resultados a los usuarios (Booch, 1992).

Para verificar que un sistema cumpla con los requisitos establecidos en los casos de uso se recomienda la utilización de pruebas unitarias (Jarrin Ortiz, M.y Valarezo Vargas E. (2010).

### **2.3.3 Scrum**

Otra metodología que se utiliza en el desarrollo de software es Scrum, sin embargo, a diferencia de la metodología FAST y en cascada, Scrum se encuentra dentro de las metodologías ágiles. Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones de un mes natural y hasta de dos semanas, si así se necesita). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite (proyectosagiles, 2014).

### 2.2.4 Métrica V3

MÉTRICA versión 3 se concibe como una metodología de planificación, desarrollo y mantenimiento de Sistemas de Información. Puede ser utilizada libremente con la única restricción de citar la fuente de su propiedad intelectual: el Ministerio de Administraciones Públicas del Gobierno de España.

Este Ministerio, desde el Consejo Superior de Informática, ofrece así a las organizaciones un instrumento para la sistematización de las actividades que dan soporte al ciclo de vida del software en el desarrollo de sistemas de información, y un marco de gestión para asegurar que los proyectos cumplen sus objetivos en términos de calidad, coste y plazos ([www.administracionelectronica.gob.es](http://www.administracionelectronica.gob.es)).

Dentro de Métrica se manejan principalmente los siguientes procesos para el desarrollo de sistemas.

- Planificación de sistemas de información (psi).
- Estudio de viabilidad del sistema (evs).
- Análisis del sistema de información (asi).
- Diseño del sistema de información (dsi).
- Construcción del sistema de información (csi).
- Implantación y aceptación del sistema (ias).
- Mantenimiento de sistemas de información (msi).

## **2.3 Arquitectura Software del sistema**

### **2.3.1 Sistema Operativo**

Una de las primeras cosas que hay que definir al momento de comenzar desde cero cuando se trabaja con aplicaciones Web que estarán en ejecución dentro de un servidor de la misma institución es el sistema operativo. “Un sistema operativo (SO, frecuentemente OS, del inglés Operating System) es un programa o conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación, ejecutándose en modo privilegiado respecto de los restantes y anteriores próximos y viceversa.” (Tanenbaum, 1992).

Dentro de los distintos sistemas operativos existentes en la actualidad, principalmente Mac Os, Windows, Linux y Unix, se ha optado por instalar Linux en el servidor Dell PowerEdge 850, servidor con el que se realizarán las pruebas conforme va progresando el proyecto. Linux es un Sistema Operativo basado en Unix, un sistema pensado originalmente para trabajar en redes, y que cuenta con un núcleo o Kernel libre. “GNU/Linux es uno de los términos empleados para referirse a la combinación del núcleo o kernel libre similar a Unix denominado Linux con el sistema GNU. Su desarrollo es uno de los ejemplos más prominentes de software libre; todo su código fuente puede ser utilizado, modificado y redistribuido libremente por cualquiera bajo los términos de la GPL (Licencia Pública General de GNU, en inglés: General Public License) y otra serie de licencias libres.” (Glyn Moody, 2001).

Existen distintas distribuciones en cuanto a Linux se refiere, ya que el código fuente de este Sistema Operativo es libre. Diversos grupos han adquirido este software libre y han trabajado con él, agregando módulos y haciendo sus propias modificaciones, dando origen a distintas distribuciones con funcionalidades distintas, algunas más que otras. De las diferentes variantes que existen del Sistema Operativo Linux se realiza el análisis específicamente de dos, Linux Mandriva y Linux CentOS.

Mandriva, originalmente llamado Mandrake, es un sistema operativo destinado tanto a usuarios principiantes como avanzados, además, está orientado tanto para computadoras personales como servidores. Mandriva ofrece la posibilidad de configurar y manejar tareas de forma fácil e intuitiva, además, es un sistema operativo estable y cuenta con un centro de control mejorado que nos brinda la posibilidad de configurar varios aspectos del hardware. El Hardware recomendado para operar es contar con memoria RAM: 256 MB (Mínimo), espacio en Disco Duro de 3 GB (Mínimo) - 4 GB (Recomendado) y un Procesador Intel Pentium I/II/III/IV/Celeron, AMD K6/II/III, AMD Duron, AMD Athlon/XP/MP (Ortiz Giménez, 2009).

CentOS por sus siglas (**C**ommunity **ENTER**prise **O**perating **S**ystem) es un proyecto con una dirección distinta derivado de la distribución Linux Red Hat Enterprise Linux (RHEL), compilado por voluntarios a partir del código fuente liberado por Red Hat. CentOS es un O.S. de tipo empresarial pero con licencia libre, es un O.S que dispone de aproximadamente 5 años de actualizaciones y debido a que es una variación parecida, por no decir una copia exacta pero sin los logos de Red Hat, es una versión estable, muchas de las aplicaciones están certificadas para

correr sobre Red Hat por lo que deben de hacerlo de la misma forma en CentOS, aunque la compatibilidad no se limita sólo a software sino que abarca también a lo que es hardware (centos, 2014). El Hardware recomendado para operar es: Memoria RAM de 64 MB (mínimo), espacio en Disco Duro de 1024 MB (mínimo) - 2 GB (recomendado).

No hay documentos donde se describa específicamente que distribución de Linux es la mejor, sin embargo, en información expuesta en distintos foros de discusión por usuarios tanto principiantes como experimentados se describe con más precisión el comportamiento de las distintas distribuciones al instalarlos en distintos equipos de cómputo y servidores, claro está, que cada uno defiende la distribución que utiliza, afirmando que es la mejor, por otro lado, me llamo la atención un comentario que decía “Ninguna distribución de Linux es la mejor, es sólo que existen necesidades diferentes”, y es cierto, todas las distribuciones de este Sistema Operativo tienen el mismo núcleo o Kernel para funcionar, la diferencia entre cada una consiste en el toque personal que le hace cada equipo de trabajo a su distribución añadiéndole funciones de acuerdo a las necesidades que hayan previsto, partiendo de ahí se puede empezar por definir cuáles son las necesidades del Sistema que se va a desarrollar en la Facultad de Contaduría y Administración, y en base a eso deducir que distribución de Linux es la que necesita el proyecto.

El Sistema en desarrollo necesita para funcionar un servidor y no una computadora de escritorio, para dar acceso a los distintos usuarios distribuyendo permisos de manera jerárquica, además el hardware tiene que responder de forma adecuada a las peticiones hechas por los distintos usuarios autenticados en el sistema.

CentOS es un Sistema Operativo orientado al uso específico y exclusivo de servidor, además de ser una distribución Enterprise es completamente libre, lo que supone una reducción en los gastos que la Facultad de Contaduría y Administración, gastos que estará exenta de hacerlos y poder destinarlos a otra actividad, CentOS contiene un excelente interface, es una distribución estable, se actualiza regularmente, y tienen un contacto cercano con Red Hat y la gente de FedoraProject. Al otro lado de la balanza se encuentra Mandriva que de igual manera cuenta con una versión para servidor sin embargo las versiones más nuevas incluyendo la versión Enterprise de esta distribución no son libres, se puede optar por una versión desktop completamente funcional y adaptarla para que funcione de forma adecuada para el servidor, sin embargo, se considera que no tiene sentido elegir una distribución que se tiene que adaptar para que cumpla las necesidades para funcionar adecuadamente en el hardware con el que se cuenta, si existe una distribución que está diseñada para ese tipo de funciones.

### **2.3.2 Servidor Web Apache**

Además del sistema operativo es necesario que el servidor pueda brindar servicio orientado a Web, es por eso que se necesita software adicional para poder proveerlo, uno de ellos es el HTTPD (HTTP daemon). “Programa que corre de fondo en un servidor Web y espera peticiones de entrada para responderles.” (Guzmán Rodríguez, C., Acevedo Rueda, W. y Duque, Y. (2011).

El protocolo HTTP generalmente utiliza el puerto 80. El HTTP está basado en el modelo cliente-servidor, en donde un cliente HTTP (un navegador por ejemplo) abre una conexión y realizar una solicitud al servidor. Este responde a la petición

con un recurso (texto, gráficos, etc) o un mensaje de error, y finalmente se cierra la conexión. Uno de los más famosos mensajes de error HTTP es el 404 Not found (alegsa, 2014).

Dentro de los HTTP existentes se encuentran los siguientes:

- Servidor Apache HTTP
- Servidor Lighttpd HTTP
- Servidor Nginx HTTP
- Servidor NCSA HTTPd HTTP
- Servidor Nullhttpd HTTP
- Servidor Thttpd HTTP

### **2.3.3 PHP**

Una vez que se ha configurado un servicio orientado a web es necesario incorporar las herramientas que permitan interpretar el código de la aplicación desarrollada, SIGAF es un sistema basado en arquitectura Web escrito en PHP. **PHP (PHP Hypertext Pre-processor)** es un lenguaje de programación desarrollado inicialmente por Rasmus Leodorf en 1995 el cual está orientado al desarrollo Web de contenido dinámico. PHP es un “lenguaje de programación usado generalmente en la creación de contenidos para sitios Web. Es un lenguaje interpretado especialmente usado para crear contenido dinámico Web y aplicaciones para servidores ya que suele utilizarse en interpretación del lado del servidor aunque también puede usarse desde una interfaz de línea de comandos y para la creación de otros tipos de programa.” (Guzmán Rodríguez, C., Acevedo Rueda, W. y Duque, Y., 2011).

### **2.3.4 MySQL**

Debido a que el sistema requiere almacenar datos para posteriormente utilizarlos en otros módulos del sistema, es necesario utilizar un motor de base de datos. “MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.”(Guzmán Rodríguez, C., Acevedo Rueda, W. y Duque, Y., 2011). Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero las empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

### **2.3.5 FTP**

Otro aspecto que se debe contemplar es la forma en que se harán las actualizaciones del sistema, hacerlo de manera remota resulta práctico y para ello se ha optado por utilizar FTP. El protocolo FTP (Protocolo de transferencia de archivos) es un protocolo para transferir archivos. (Alcantud, Marin, 1999).

La implementación del FTP se remonta a 1971 cuando se desarrolló un sistema de transferencia de archivos (descrito en RFC141) entre equipos del Instituto Tecnológico de Massachusetts (MIT, Massachusetts Institute of Technology). Desde entonces, diversos documentos de RFC (petición de comentarios) han mejorado el protocolo básico, pero las innovaciones más importantes se llevaron a cabo en julio de 1973.

Actualmente, el protocolo FTP está definido por RFC 959 (ccm, 2014).

## **2.4 Arquitectura Hardware del sistema**

Un sistema informático funcional además del software y el personal que lo maneja está formado por hardware. “Un sistema informático un conjunto de personas, datos, procesos y tecnología de la información que interactúan para recoger, procesar, almacenar y proveer la información necesaria para el correcto funcionamiento de la organización.”(Whitten, Bentley y Dittman, 2004).

Es importante conocer el recurso material con el que se cuenta actualmente en el lugar donde se pretende implementar un sistema, de esa manera, se puede determinar lo que se necesita y solicitar los elementos faltantes para que el proyecto siga avanzando. La Facultad de Contaduría y Administración tiene un servidor disponible, hardware necesario para el funcionamiento del Sistema en desarrollo, por el momento no se utiliza para brindar algún tipo de servicio a la facultad, por lo que en su momento se pudo proceder a realizar el trámite necesario con las respectivas formalidades para su pronta adquisición.

### **2.4.1 Servidor**

“Un servidor es una computadora en la que se ejecuta un programa que realiza alguna tarea en beneficio de otras aplicaciones llamadas clientes, tanto si se trata de un ordenador central (mainframe), un miniordenador, una computadora personal, una PDA o un sistema embebido; sin embargo, hay computadoras destinadas únicamente a proveer los servicios de estos programas: estos son los servidores por antonomasia.” (Microsoft Official Academic Course, 2011).

En cuanto al servidor se pretende utilizar el localizado en el centro de cómputo de la FCA, ya que se encuentra en condiciones para disponer de él, para posteriormente poder realizar las respectivas tareas. El servidor se encuentra funcional, y se pretende instalar en el servidor un sistema operativo gratuita llamado CentOS (CommunityENTerpriseOperatingSystem), este sistema operativo es una bifurcación a nivel binario de la distribución Linux, Red Hat Enterprise Linux. CentOS es una de las distribuciones de Linux más estables. En la tabla 2.2 se observan las especificaciones del servidor con el que cuenta la FCA.

**Tabla 2.2.** Especificaciones del servidor FCA. Fuente: dell, 2014.

<b>Especificaciones</b>	
<b><i>Servidor</i></b>	DELL PowerEdge 850
<i>Procesador</i>	Pentium 4 Xeon de 3.8 Ghz.
<i>RAM</i>	8 Gb
<i>HD</i>	SCSI 140
<b><i>TARJETA DE RED</i></b>	NIC PCI Express PCI-e con 2 Puertos Gigabit Ethernet RJ45

# *Capítulo III*

## *Metodología*

### 3. Metodología

El desarrollo del proyecto se llevará a cabo por un equipo de trabajo, por lo que se requerirá una forma de sincronizar las tareas que realizarán cada uno de los miembros del equipo. Para el desarrollo de este trabajo se utilizarán principalmente dos metodologías, la metodología Scrum que permitirá la sincronización de actividades del desarrollo, a la vez, definirá los roles y el tiempo de desarrollo de los productos, y la metodología Métrica V3 para realizar la implementación y las pruebas del sistema en el área de operación.

#### 3.1 Modelo de trabajo Scrum

La forma como se trabaja con Scrum se basa en unas series de prácticas orientadas al trabajo en equipo con el objetivo de obtener los mejores resultados posibles en cualquier proyecto en el que se trabaje; En la figura 3.1 se logra apreciar de forma gráfica el modelo de trabajo Scrum.

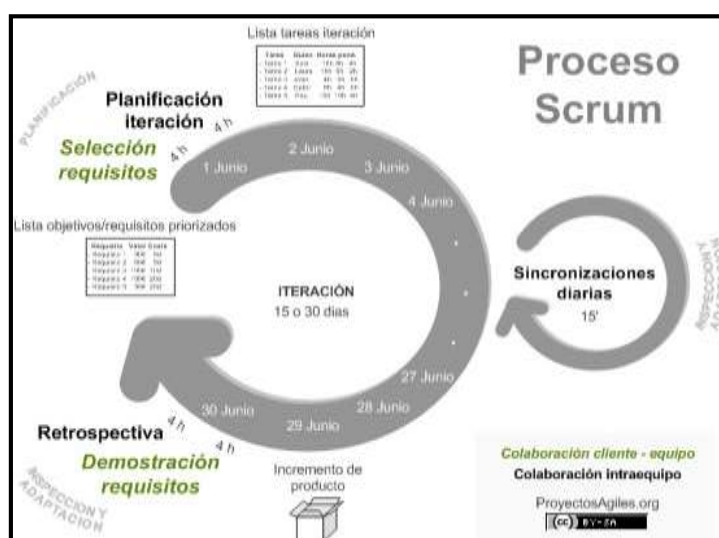


Figura 3.1. Modelo Scrum. Fuente: [www.proyectosagiles.org](http://www.proyectosagiles.org)

Utilizar Scrum conlleva el obtener resultados de forma rápida ya sea porque se trata de un proyecto complejo o los requisitos del mismo varían fácilmente o inclusive son poco definidos.

Para el desarrollo del proyecto se utilizará esta metodología para sincronizar las tareas del equipo de trabajo que elaboran el proyecto. La Metodología Scrum maneja los SPRINT que básicamente son lapsos de tiempo donde se realiza todo el proceso de desarrollo, desde la determinación de los requerimientos de una parte del sistema que se va a crear hasta su desarrollo, para entregar una parte funcional del sistema al final de dicho lapso. Al inicio del desarrollo del proyecto se establecerá para los SPRINT un lapso de cuatro semanas para hacer los entregables pertinentes del avance del proyecto.

Se utilizará Scrum ya que permite que los entregables sean presentados al cliente en lapsos de tiempo relativamente cortos, además que los productos generados logran cumplir con la calidad mínima requerida.

### **3.2 Metodología MÉTRICA V3**

“La metodología MÉTRICA Versión 3 proporciona un conjunto de métodos y técnicas que guía a los distintos profesionales de Sistemas y Tecnologías de la Información y Comunicaciones (STIC) en la obtención de los diversos productos de los procesos del ciclo de vida de un proyecto informático. Con el fin de mejorar la productividad de los distintos participantes y asegurar la calidad de los productos resultantes, la mayoría de las técnicas propuestas están soportadas por herramientas disponibles en el mercado que automatizan en mayor o menor grado su utilización.” (Métrica V3, 2001).

Dado que el alcance del proyecto no es el desarrollo del sistema sino la implementación del mismo, únicamente se siguió como guía el proceso de implantación y aceptación del sistema establecido en esta metodología.

### **3.3 Implementación**

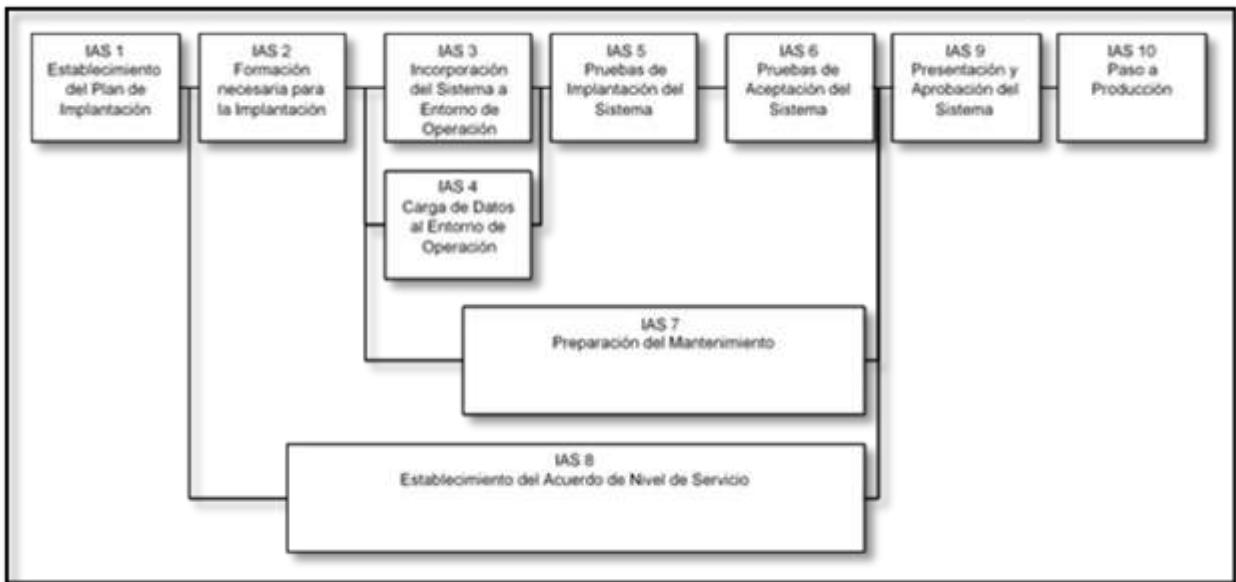
Durante la implementación se realizarán todas aquellas actividades necesarias para que el Sistema de Gestión Académica se incorpore al área de operación una vez que este se encuentre en la fase funcional.

Antes de comenzar con este proceso se verificarán algunos aspectos, entre ellos se encuentran los siguientes:

- Estudio de viabilidad del sistema: con este documento se pretende determinar su alcance y en base a esto elaborar el plan de implantación, junto al equipo que llevará a cabo esta tarea.

- Productos software probado: se toma como punto inicial para preparar la infraestructura necesaria y configurar el entorno.

Para dar seguimiento al procedimiento de implantación se llevarán a cabo la realización de las acciones necesarias para dar inicio a su operación. En la figura 3.2 se muestra la relación de actividades de este proceso.



**Figura 3.2.** Proceso de implantación. Fuente: Métrica V3, 2001.

### 3.3.1 Establecimiento del plan de implantación

Para la elaboración del plan de implantación se verificará el análisis del sistema, esto permitirá conocer el alcance de la implementación. Se analizarán los requerimientos para la implantación, el procedimiento de implantación y la especificación de requisitos de operación y seguridad. Básicamente lo que se buscará en esta etapa de la implantación es elaborar los siguientes elementos:

- Definición del plan de implantación.
- Especificación del equipo de implantación

### **3.3.2 Formación necesaria para la implantación**

Durante esta etapa se formará el equipo de trabajo que llevará a cabo la implantación del sistema. En caso de ser necesario se definirán los recursos humanos técnicos y materiales que se utilizarán para la formación del equipo de implantación.

### **3.3.3 Incorporación del sistema al entorno de operación**

En esta etapa se realizarán las actividades necesarias para incorporar el sistema al entorno de operación para que los usuarios finales tengan acceso al mismo.

Esta etapa se dividió en dos fases:

- Preparación de la instalación.
- Realización de la instalación.
- Pruebas unitarias.

#### **3.3.3.1 Preparación de la instalación**

En esta etapa se verificará que la infraestructura necesaria está disponible para poder hacer la configuración del entorno.

Una vez que se ha comprobado la existencia de la infraestructura se realizará la instalación del software de base necesario para la incorporación de los módulos el sistema objeto de la implementación.

### **3.3.3.2 Realización de la instalación**

En esta parte se realizará la instalación de todos los módulos del Sistema de Gestión Académica. Además se preparará el entorno de datos y se identificarán los sistemas de información que formarán parte del Sistema de Gestión.

Para el funcionamiento del sistema se hará lo siguiente:

- Se incorporará la base de datos que necesitará el Sistema de Gestión Académica.
- Se establecerá el protocolo para la realización de copias de seguridad de la base de datos y la restauración de las mismas.
- Se prepararán las autorizaciones de acceso de los datos para los distintos perfiles de usuario.
- Se incorporará el sistema al entorno de operación.

### **3.3.3.3 Pruebas unitarias**

Con el fin de verificar la funcionalidad del Sistema de Gestión Académica FCA, se llevaron a cabo una serie de pruebas, que se realizaron de acuerdo al avance que el sistema tuvo en distintas etapas del desarrollo, las cuales fueron las siguientes: pruebas unitarias, pruebas de integración y pruebas de implantación.

Debido a que el sistema se desarrollará en módulos, subsistemas que formarán parte de un sistema completo, la verificación de las funcionalidades del mismo se llevarán a cabo en fases, las actividades que se realizarán en cada una de las fases del desarrollo de las pruebas básicamente se repetirán debido a que el

proceso se realizará de forma iterativa y cada uno de los ciclos se abarcará más módulos de manera progresivamente en lo que el sistema se iba desarrollando.

### **3.3.4 Carga de datos al entorno de operación**

Teniendo en cuenta que los sistemas de información que forman parte del sistema a implantar pueden mejorar, ampliar o sustituir a otros ya existentes en la facultad, puede ser necesaria una carga inicial y/o una migración de datos cuyo alcance dependerá de las características y cobertura del sistema de información implicado.

# *Capítulo IV*

## *Desarrollo*

## **4.1 Plan de implantación**

Durante esta parte se revisaron los documentos que contenían el análisis de los módulos del sistema y se determinaron los requisitos necesarios para la implantación. Dentro de estos requisitos se contempló la instalación, infraestructura y formación necesaria del equipo para llevar a cabo esta tarea.

Además, se elaboró un documento donde se definió el plan de implantación y la especificación del equipo de implantación (ver anexo 1).

## **4.2 Formación necesaria para la implantación**

En esta sección se definió a una sola persona a cargo de la implantación, sin embargo, se determinó que los demás miembros del equipo apoyarían para dar por finalizado ciertas tareas, por ejemplo, la carga de datos al sistema desarrollado para poder realizar las pruebas necesarias al mismo.

## **4.3 Incorporación del sistema al entorno de operación**

En esta etapa se realizaron una serie de actividades requeridas para incorporar el sistema al entorno de operación donde los usuarios finales tuvieran acceso al mismo.

Esta etapa se dividió en tres fases:

- Preparación de la instalación.
- Realización de la instalación.
- Pruebas unitarias.

### 4.3.1 Preparación de la instalación

Antes de iniciar con la incorporación del sistema al entorno de operación se verificó que la Facultad tuviera la infraestructura necesaria para operar. Estos datos se obtuvieron mediante la observación minuciosa del área de operación. La Facultad contaba con una red que tenía acceso a Internet, por otro lado, tenía en disposición un servidor el cual podría ser configurado para la implementación del sistema, las especificaciones técnicas del servidor se encuentran en la tabla 4.1.

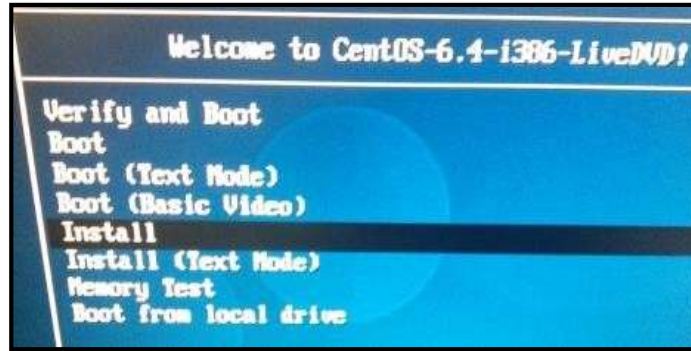
**Tabla 4.1.** Especificaciones técnicas del servidor. Fuente: dell, 2014.

Procesador	RAM	Disco Duro	Tarjeta de red
Pentium 4 Xeon de 3.8 Ghz.	8 Gb	SCSI 140 GB	NIC PCI Express PCI-e con 2 Puertos Gigabit Ethernet RJ45

Después de verificar que la Facultad contaba con la infraestructura necesaria para la implementación, se prosiguió a instalar los componentes que permitieron el funcionamiento del sistema.

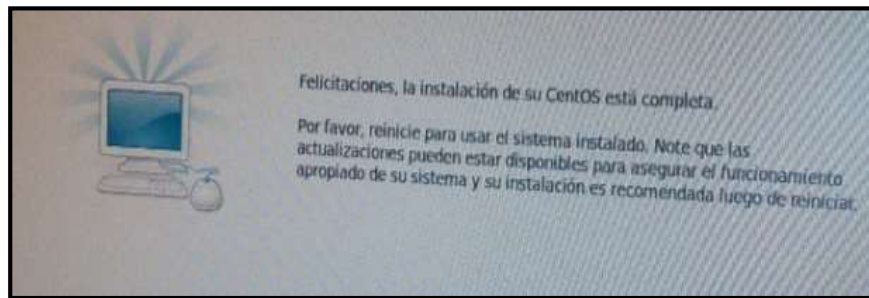
#### 4.3.1.1 Instalación del sistema operativo

Se instaló el sistema operativo definido en el plan de implantación, durante este proceso se trabajó con “CentOS release 6.4 Final” ya que era la versión más reciente en ese momento. Durante la secuencia de arranque del DVD de instalación de CentOS se mostró la primera pantalla con diversas opciones. Para continuar se seleccionó la opción “INSTALL” (ver figura 4.1).



**Figura 4.1.** Menú de instalación. Fuente: SIGAF.

Después de haber seleccionado la configuración necesaria para realizar este proceso, inicio la instalación del sistema operativo. Una vez terminada la instalación de manera satisfactoria, el asistente mostró una pantalla donde se indicaba que la instalación de CentOS había terminado (ver figura 4.2).



**Figura 4.2.** Instalación completa. Fuente: SIGAF.

Después de que la instalación finalizó se comprobó que el sistema operativo iniciara correctamente y que permitiera acceder al mismo con la contraseña asignada durante la instalación (ver figura 4.3).



**Figura 4.3.** CentOS Login. Fuente: SIGAF.

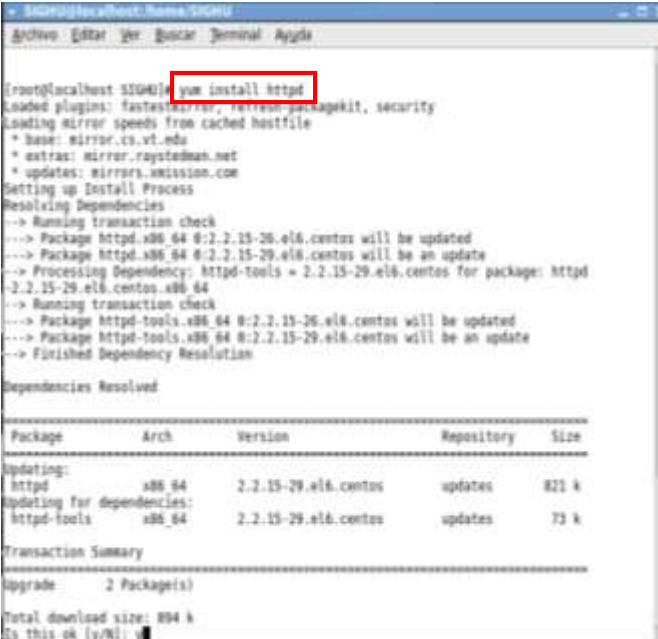
### 4.3.1.2 Instalación del servidor Web

Al finalizar la instalación del sistema operativo se continuó con la configuración del servidor HTTP Apache.

CentOS es un sistema operativo dirigido especialmente para servidores por lo que los paquetes necesarios para la instalación del software para que funcione como tal se encuentran dentro del repositorio del mismo sistema operativo, para instalar el servidor HTTP se abrió una terminal, se elevó a nivel superusuario y se escribió la siguiente línea de comando en la terminal.

```
$ su  
# yum install httpd
```

Una vez ejecutado el comando, el sistema operativo comprobó que los paquetes necesarios para instalar el software estuvieran disponibles en el repositorio para comenzar a descargarlos.



```
root@localhost:~# yum install httpd  
Loaded plugins: fastestmirror, refresh-packagekit, security  
Loading mirror speeds from cached hostfile  
* base: mirror.cs.vt.edu  
* extras: mirror.rostvedman.net  
* updates: mirrors.emission.com  
Setting up Install Process  
Resolving Dependencies  
--> Running transaction check  
--> Package httpd.x86_64 0:2.2.15-26.el6.centos will be updated  
--> Package httpd.x86_64 0:2.2.15-29.el6.centos will be an update  
--> Processing Dependency: httpd-tools = 2.2.15-29.el6.centos for package: httpd-2.2.15-29.el6.centos.x86_64  
--> Running transaction check  
--> Package httpd-tools.x86_64 0:2.2.15-26.el6.centos will be updated  
--> Package httpd-tools.x86_64 0:2.2.15-29.el6.centos will be an update  
--> Finished Dependency Resolution  
  
Dependencies Resolved  
  
-----  
Package Arch Version Repository Size  
-----  
Updating:  
httpd x86_64 2.2.15-29.el6.centos updates 821 k  
Updating for dependencies:  
httpd-tools x86_64 2.2.15-29.el6.centos updates 73 k  
-----  
Transaction Summary  
-----  
Upgrade 2 Package(s)  
Total download size: 894 k  
Is this ok [y/N]:
```

**Figura 4.4.** Instalación del servidor HTTP.  
Fuente: SIGAF.

Después de que se han encontrado los paquetes y se han transferido al servidor, se procede con la instalación del software (ver figura 4.4).

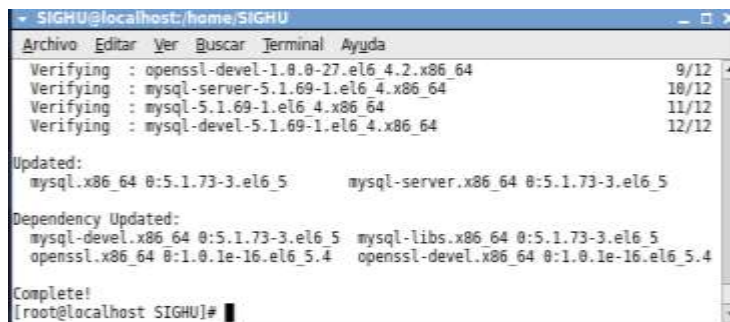
#### 4.3.1.3 Instalación del motor de base de datos MySQL

Para la instalación del motor de base de datos se ejecutó la siguiente línea en la terminal como usuario root:

```
$ su
# yum install -y mysql mysql-server
```

Durante cada instalación que se realizó desde la terminal, el sistema comprobó la existencia de los paquetes necesarios dentro del repositorio y posteriormente instaló el software.

El sistema no encontró ningún error en este paso, por lo que se completó la instalación del motor de base de datos MySQL (ver figura 4.5).



```
SIGHU@localhost/home/SIGHU
Archivo Editar Ver Buscar Terminal Ayuda
Verifying : openssl-devel-1.0.0-27.el6_4.2.x86_64 9/12
Verifying : mysql-server-5.1.69-1.el6_4.x86_64 10/12
Verifying : mysql-5.1.69-1.el6_4.x86_64 11/12
Verifying : mysql-devel-5.1.69-1.el6_4.x86_64 12/12

Updated:
mysql.x86_64 0:5.1.73-3.el6_5 mysql-server.x86_64 0:5.1.73-3.el6_5

Dependency Updated:
mysql-devel.x86_64 0:5.1.73-3.el6_5 mysql-libs.x86_64 0:5.1.73-3.el6_5
openssl.x86_64 0:1.0.1e-16.el6_5.4 openssl-devel.x86_64 0:1.0.1e-16.el6_5.4

Complete!
[root@localhost SIGHU]#
```

**Figura 4.5.** Instalación de MySQL. Fuente: SIGAF.

#### 4.3.1.4 Instalación de PHP

Para el desarrollo del sistema que se pensaba implementar se utilizó el lenguaje PHP, por lo que para que pudiera funcionar en el servidor se necesitaba instalar dentro del mismo. Para su instalación se ejecutó lo siguiente en la terminal del sistema operativo:

```

$su

#yum install php php-devel php-gd php-imap

#yum install php-ldap php-mysql php-odbc

#yum install php-pear php-xml php-xmlrpc

#yum install php-pecl-apc php-mbstring

#yum install php-mcrypt php-mssql php-

#yum install snmp php-soap php-tidy curl

#yum install curl-devel perl-libwww-perl

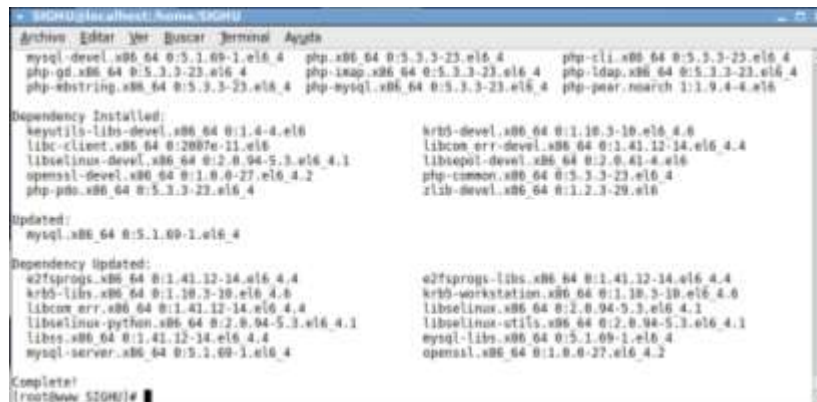
#yum install ImageMagick libxml2

#yum install libxml2-devel mod_fcgid

#yum install php-cli httpd-devel

```

Una vez que se instalaron los componentes de PHP necesarios para poder ejecutar el sistema (Figura 4.6) se procedió a hacer las configuraciones para el correcto funcionamiento.



**Figura 4.6.** Instalación de PHP. Fuente: SIGAF.

### 4.3.2 Realización de la instalación.

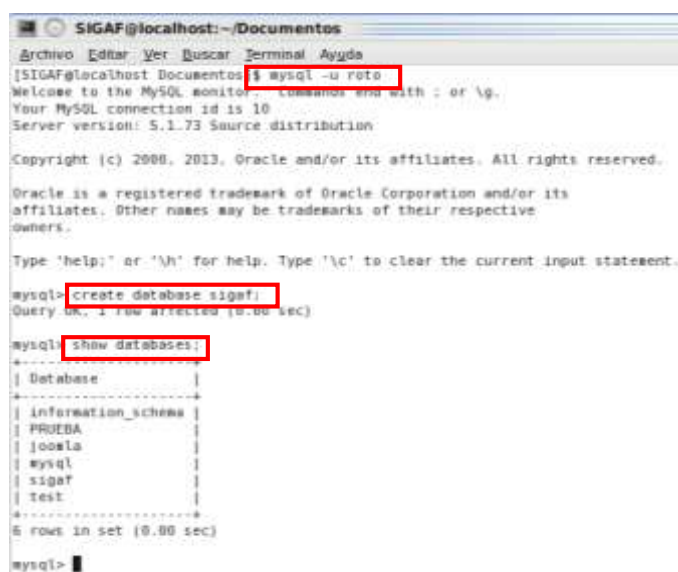
Después de instalar los componentes que permitirían el funcionamiento del sistema y haber realizado las configuraciones necesarias, se comienza con la incorporación de los componentes del Sistema de Gestión Académica.

#### 4.3.2.1 Incorporación de la base de datos

El script que permitió incorporar la base de datos al servidor que proveería los datos necesarios para funcionar se encontraba dentro del archivo “sigaf.sql”, sin embargo, antes de ejecutar el script se inició el servicio de MySQL.

```
$su  
#service mysqld start
```

Una vez iniciado el servicio se verificó que el motor de base de datos trabajara de manera correcta y que estuviera habilitado para realizar operaciones en las bases de datos almacenadas (ver figura 4.7).



```
SIGAF@localhost:~/Documentos  
[Archivo Editor Ver Buscar Terminal Ayuda  
[SIGAF@localhost Documentos] $ mysql -u root  
Welcome to the MySQL monitor.  Use lowercase and with ; or \g.  
Your MySQL connection id is 10  
Server version: 5.1.73 Source distribution  
  
Copyright (c) 2008, 2013, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
mysql> create database sigaf;  
Query OK, 1 row affected (0.00 sec)  
  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| PRUEBA |  
| Joomla |  
| mysql |  
| sigaf |  
| test |  
+-----+  
5 rows in set (0.00 sec)  
  
mysql>
```

**Figura 4.7.** Verificar motor de base de datos.  
Fuente: SIGAF.

Al observar que el motor trabajaba de manera correcta se continuó con la incorporación de la base de datos al servidor, en la figura 4.8 se observa el script que se ejecutó para cumplir dicha tarea.

```

SIGAF@localhost:~/Documentos
Archivo Editar Ver Buscar Terminal Ayuda
[SIGAF@localhost Documentos]$ mysql -u root sigaf -e "/sigaf.sql
[SIGAF@localhost Documentos]$
[SIGAF@localhost Documentos]$ mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 5.1.73 Source distribution

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use sigaf;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_sigaf |
+-----+
| campus          |
| caracter        |
| ciudad          |
| coordinacion    |
| detallamiento  |
| directivos      |
| directivosua    |
| esp_nivel_prog  |
| estado          |
| etapas          |
| noprograma      |
| p_usa           |
| pais            |
| planestudio     |
| reoperacion     |
| tipoprograma    |
| academica       |
| usrendizaje     |
| users           |
+-----+
19 rows in set (0.00 sec)

mysql>

```

**Figura 4.8.** Crear base de datos.  
Fuente: SIGAF.

#### 4.3.2.2 Servidor HTTP

Para poder acceder al sistema desde una computadora externa, lo primero que se hizo fue iniciar el servicio HTTP:

```
$su
#service httpd start
```

Luego de tener el servicio disponible se habilitó dentro del firewall el puerto 80 para que el servidor HTTP Apache tuviera acceso a las peticiones de los clientes, como se puede observar en la figura 4.9.

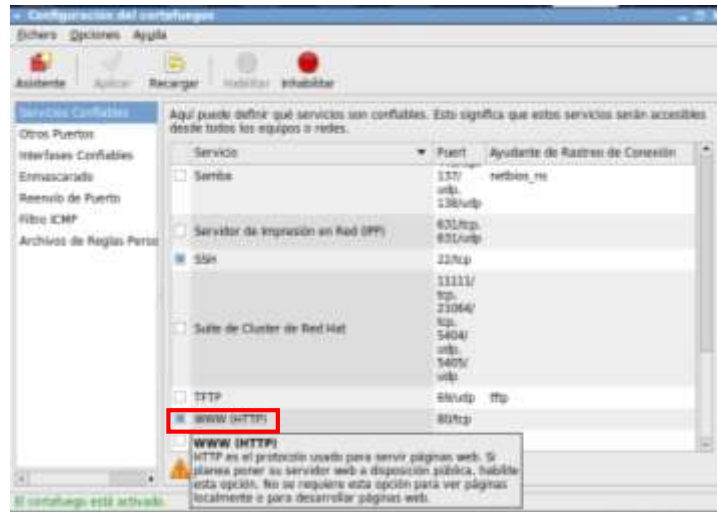


Figura 4.9. Cortafuego (HTTP). Fuente: SIGAF.

Se comprobó que el servidor estuviera habilitado para recibir peticiones, para ello, se instaló una página HTML de prueba dentro del mismo, como se observa en la figura 4.10, se intentó acceder desde una computadora conectada a la misma red mediante su dirección IP.

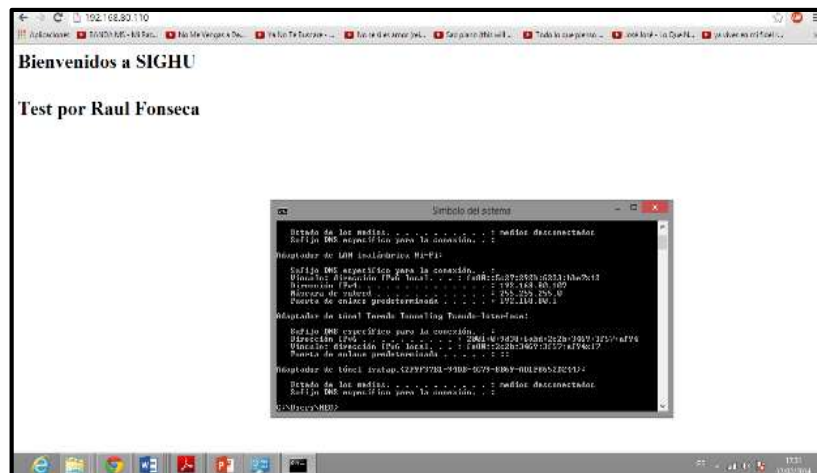


Figura 4.10. Acceso al servidor. Fuente: SIGAF.

Una vez comprobado el funcionamiento del servidor HTTP Apache se continuó con la instalación de SIGAF.

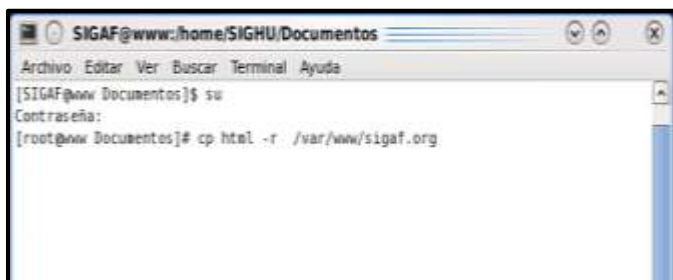
### 4.3.2.3 Implementación del sistema.

El Sistema de Gestión Académica es un sistema que contempla varias funciones por lo que para su desarrollo se dividió en varios módulos. La división en partes del sistema permitió incorporar varias versiones del sistema y comprobar el funcionamiento del mismo.

#### 4.3.2.3.1 Módulo uno: plan de Estudios.

Durante la primera etapa de la implementación del sistema se trabajó directamente con el servidor por lo que se incorporó los archivos necesarios del sistema en el directorio establecido en el archivo de configuración del servidor HTTP (ver figura 4.11).

Para poder ingresar a la página de acceso del sistema se necesitó introducir los datos de los usuarios almacenados en la base de datos, sin embargo, las contraseñas se encontraban encriptadas, por lo que era necesario que el servidor



**Figura 4.11.** Instalación de SIGAF. Fuente: SIGAF.

tuviera instalada la criptografía y habilitar al sistema para utilizar esta función. Para instalar la criptografía MCRYPT desde el repositorio se ejecutó la siguiente instrucción:

```
$ su
# yum update
# yum install php-mcrypt
```

Ya instalada la criptografía, el sistema estaba habilitado para encriptar y desencriptar los datos necesarios para su funcionamiento, en la figura 4.12 se aprecia la pantalla de acceso al sistema instalado en el servidor.



**Figura 4.12.** Login. Fuente: SIGAF.

Después se validó el acceso al sistema mediante el uso de usuarios previamente registrados y se comprobó que el sistema permitiera acceder a las herramientas administrativas como se ve en la figura 4.13.



**Figura 4.13.** Interfaz del sistema. Fuente: SIGAF.

#### 4.3.2.3.2 Módulo dos: carga académica.

En la segunda fase de la implementación del sistema se trabajó con el módulo carga académica. Durante esta parte se incorporó una nueva versión, tanto del sistema como de la base de datos.

La configuración del servidor sufrió cambios, además, se optó por tener una copia del sistema en un servidor externo para comprobar su comportamiento, tanto en un ambiente local como global, y digo global porque se puede acceder al sistema desde cualquier parte siempre y cuando el usuario tenga conexión a Internet.

Durante esta parte se habilitó el servidor para que pudiera trabajar con diferentes sistemas a la vez, por lo que se tuvo que realizar configuraciones adicionales para que se pueda acceder a ellos.

En la figura 4.14 se aprecia la nueva ruta en donde se incorporaron los dos sistemas, uno con el nombre de SIGAF, sistema desarrollado por el equipo de desarrollo, y otro con el nombre de SIEG, sistema de prueba.



**Figura 4.14.** Ruta de los sistemas. Fuente: SIGAF.

En ese momento la IP local del servidor era 192.168.10.100 y para acceder al sistema se deseaba que se hiciera ingresando a la dirección 192.168.10.100/sigaf. Para poder realizar lo anterior, como se puede observar en la figura 4.15, se agregó un archivo llamado index.php a la raíz del directorio del sistema.



**Figura 4.15.** Fichero index. Fuente: SIGAF.

El archivo index.php contenía la siguiente línea de código:

```
<?php  
header("Location:
```

Lo que hace este archivo es direccionar al usuario a otro archivo con el mismo nombre ubicado en una carpeta llamado public. En esta carpeta se encuentra el archivo que permite dar acceso al sistema.

De esta manera se pudo lograr acceder al sistema ingresando a la dirección 192.168.10.100/sigaf, lo cual le agrega un poco de facilidad de acceso al usuario en cuestión.

### 4.3.3 Pruebas unitarias del sistema.

Con el fin de comprobar el funcionamiento del sistema se elaboró un plan de pruebas para cada uno de los módulos que lo componen basado en los requerimientos establecidos en la fase de análisis y las funcionalidades que el sistema debe cumplir. En la tabla 4.2 se muestra el plan de pruebas del primer módulo.

**Tabla 4.2. Plan de pruebas. Fuente: Elaboración propia.**

UABC	Nombre del Proyecto:	SIGAF	Diseñado por:	Raúl Fonseca		
	Nombre del Módulo:	Plan de Estudio	Fecha de Diseño:	21/04/2014		
	Release Version:	3,0				
Pre-condición:						
Dependencias:						
Prioridad:						
Caso de Prueba#	Código#	Título	Resumen	Post-condición	Resultado	Estado
UC01	ES01	Cancelar registro.	El usuario cancela la operación y el caso de uso termina.			PROCESO
UC01	ES02	Nuevo registro.	El sistema almacena el nuevo registro.		El sistema permite ingresar información pero aun no permite almacenar el registro.	PROCESO
UC01	ES03	Cancelar registro (iteración).	Después de una iteración el usuario cancela la operación y el caso de uso termina.			PROCESO
UC01	ES04	Nuevo registro (iteración).	Después de una iteración el sistema almacena el nuevo registro.			PROCESO
UC02	ES01	Error en consulta.	Si no hay ninguna Unidad de Aprendizaje o el sistema no puede recuperar las unidades, se muestra un error y el caso se termina.			PROCESO
UC02	ES02	Cancelar consulta.	El usuario cancela la operación y este caso de uso termina.			PROCESO
UC02	ES03	Consulta personalizada.	El usuario desea una consulta personalizada y se abre la ventana nueva.			PROCESO
UC02	ES04	Consulta.	El sistema muestra la Consulta.			PROCESO
UC02	ES05	Cancelar consulta (iteración).	Después de un error el usuario cancela la consulta y este caso de uso termina.			PROCESO
UC02	ES06	Consulta personalizada (iteración).	Después de un error el usuario desea una consulta personalizada y se abre la ventana nueva.			PROCESO
UC02	ES07	Consulta (iteración).	Después de un error el usuario intenta la operación de nuevo y el sistema muestra la Consulta.			PROCESO
UC03	ES01	Error en consulta personalizada.	Si no hay ningún campo o el sistema no puede recuperar los campos, se muestra un error y el caso se termina.			PROCESO
UC03	ES02	Cancelar consulta personalizada.	El usuario cancela la operación y este caso de uso termina.			PROCESO
UC04	ES01	Error al eliminar unidad académica.	Si no hay ninguna Unidad de Aprendizaje o el sistema no puede recuperar Unidades de Aprendizaje, se muestra un error.			PROCESO
UC04	ES02	Cancelar la eliminación de unidad académica.	El usuario cancela la operación			PROCESO
UC04	ES03	Eliminar unidad académica.	El sistema borra Unidad de Aprendizaje.			PROCESO
UC04	ES04	Error al eliminar unidad académica. (Iteración)	Después de un error el usuario intenta de nuevo la operación y si no hay ninguna Unidad de Aprendizaje o el sistema no puede recuperar Unidades de Aprendizaje, se muestra un error			PROCESO

En la tabla 4.3 se muestra la plantilla que permitió registrar los resultados de las pruebas realizadas en el primer módulo del sistema.

**Tabla 4.3. Ejecución de pruebas. Fuente: Elaboración propia.**

Sección: Plan de Estudio - Registro							
Caso de prueba	cancelar registro		ID: UC010301	Severidad	Fecha: FEBRERO 2014		
Propósito de la prueba:	Dependencia de red: No						
Criterios de prueba	Test Step		Stage Result	Test Steps			
	No.	Descriptive	Descriptor	Pass	Fail	NA	Execution
	1	El usuario solicita Registrar Unidad de Aprendizaje		X			
	2	El sistema solicita la información de la Unidad de Aprendizaje.		X			
	3	El usuario introduce la información de la Unidad de Aprendizaje.				X	
4	El usuario cancela la operación y este caso de uso termina.				X		
Overall Result				2	0	2	0
Resultados observados en caso de Fallo.							
Descripción del impacto en el usuario final.							
Comentarios							

Con el fin de verificar la funcionalidad de SIGAF, se llevó a cabo una serie de pruebas, que se realizaron de acuerdo al avance que el sistema tuvo en distintas etapas del desarrollo, tales como las pruebas unitarias, las pruebas de integración y las pruebas de implantación.

#### 4.3.3.1 Plan de pruebas

Debido a que el sistema se desarrolló en módulos, la verificación de las funcionalidades del mismo se llevó en fases. Las actividades realizadas en cada una de las fases del desarrollo de las pruebas, se repitieron debido a que el proceso se realizó de forma iterativa y durante cada uno de los ciclos se agregaron funcionalidades al sistema.

Con el fin de conocer lo que se va a probar y determinar el tiempo requerido para realizar esta actividad se desarrolló un plan de pruebas en cada fase. En este plan se definieron las funcionalidades que se iban a verificar, a la vez, se le asignó un código a cada funcionalidad para identificarla durante su ejecución y mostrar los resultados a los miembros del equipo de desarrollo.

Para el diseño del plan de prueba es necesario que en la etapa de análisis se especifiquen los siguientes elementos:

- Catálogo de Requisitos.
- Modelo de Casos de Uso.
- Especificación de Casos de Uso.

Durante esta fase se verificó el módulo Plan de Estudios. En el diseño del plan de pruebas también se analizaron los datos válidos e inválidos soportados en este módulo del sistema, de esta forma se determinó que la funcionalidad de cada elemento estuviera alineadas con los requerimientos establecidos en la etapa de análisis. En la tabla 4.4 se puede observar la clase de equivalencia donde se encuentran estos datos.

**Tabla 4.4 Clase de Equivalencias. Fuente: Elaboración propia.**

Sec.	Condición de		Clases Válidas		Clases no Válidas	
	Entrada	Tipo	Entrada	Código	Entrada	Código
1	Unidad Académica	ComboBox	FCA	CEV<01>		
2	Campus	ComboBox	TIJUANA	CEV<02>		
3	Nivel Programa	ComboBox	Orden = "TRONCO COMUN"	CEV<03>		
			Orden = "LICENCIATURA"	CEV<04>		
			Orden = "POSTGRADO"	CEV<05>		
4	Carrera	ComboBox	Orden = "TECNICO"	CEV<06>		
			Orden = "CP"	CEV<07>		
			Orden = "LAE"	CEV<08>		
			Orden = "U"	CEV<09>		
			Orden = "LNI"	CEV<10>		
			Orden = "Especialidad,"	CEV<11>		
			Orden = "Maestría"	CEV<12>		
Orden = "Doctorado"	CEV<13>					
5	No. de Plan	Valor	¿??????	CEV<14>		
6	Materia	Valor	Cualquier cadena de caracteres alfanumericos hasta de 15 posiciones.	CEV<15>		
7	Clave	Valor	Cualquier numero.	CEV<16>		
8	HC: Horas Clase	Valor	Cualquier numero.	CEV<17>		
9	HE: Horas Extra clase	Valor	Cualquier numero.	CEV<18>		
10	HL: Horas Laboratorio	Valor	Cualquier numero.	CEV<19>		
11	HT: Horas Taller	Valor	Cualquier numero.	CEV<20>		
12	HPC: Horas Prácticasde Campo	Valor	Cualquier numero.	CEV<21>		
13	HCL: Horas Clínicas	Valor	Cualquier numero.	CEV<22>		
15	Créditos	Valor	Cualquier numero.	CEV<23>		
16	Etapa	ComboBox	Orden = "Básica"	CEV<24>		
			Orden = "Disciplinaria"	CEV<25>		
			Orden = "Terminal"	CEV<26>		
17	Semestre	Valor	Cualquier numero.	CEV<27>		
18	Carácter	ComboBox	Orden = "Optativo"	CEV<28>		
			Orden = "Obligatoria"	CEV<29>		

### 4.3.3.2 Diseño de los casos de uso

Una vez establecidos las clases de equivalencia se diseñaron los casos de uso con base en las funcionalidades descritas en el plan de pruebas.

Los casos de uso permitieron definir las funcionalidades que se iban probar de esta parte del sistema.

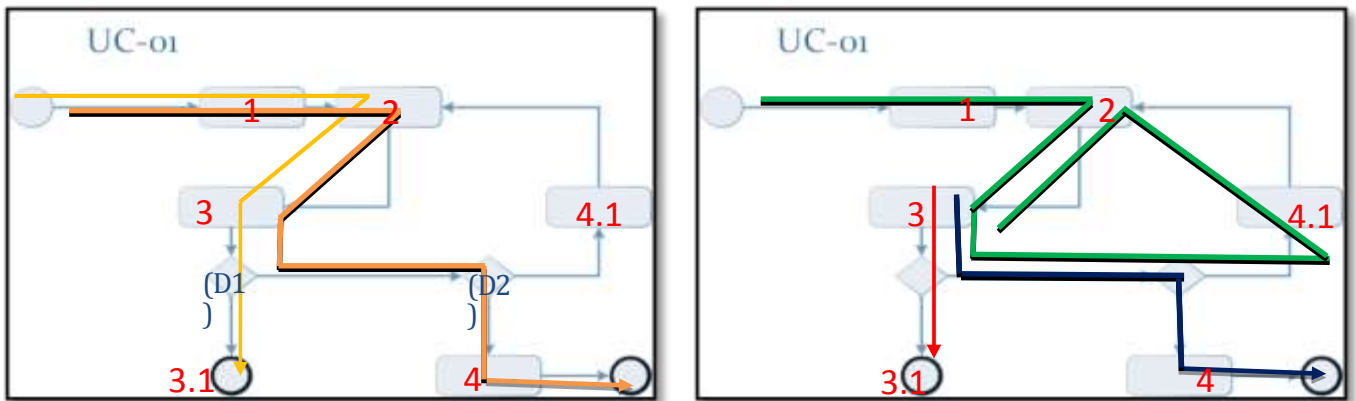
- *Registra unidad de aprendizaje*

En la tabla 4.5 se muestra el caso de uso para registrar una nueva unidad de aprendizaje.

Tabla 4.5. Registrar nueva unidad de aprendizaje. Fuente: Elaboración propia.

Nombre		UC-01. Registrar nueva unidad de aprendizaje
Precondición	Ingresar al sistema.	
Secuencia principal	<ol style="list-style-type: none"> <li>1. El usuario solicita registrar unidad de aprendizaje.</li> <li>2. El sistema solicita la información de la unidad de aprendizaje.</li> <li>3. El usuario introduce la información de la unidad de aprendizaje.</li> <li>4. El sistema almacena el nuevo registro.</li> </ol>	
Error/secuencias alternativas	<ol style="list-style-type: none"> <li>3.1. El usuario cancela la operación y este caso de uso termina.</li> <li>4.1. Si algún campo está vacío, el sistema muestra un mensaje de error y solicita de nuevo los datos de la unidad de aprendizaje.</li> </ol>	
Post-condición	Nueva unidad de aprendizaje registrada.	
Notas	<ul style="list-style-type: none"> <li>• En caso que el usuario capture en minúsculas o tipo oración, el sistema debe convertir los caracteres en mayúsculas.</li> <li>• Que en nivel licenciatura una unidad de aprendizaje se asigne a uno o varios niveles de programa ejemplo contabilidad (11237) esté disponible para las 4 carreras sin necesidad de realizar 4 veces la captura de la misma unidad de aprendizaje.</li> <li>• La clave de una materia puede ser la misma de plan a plan. La clave de una materia puede variar de plan a plan en ambos casos y es importante identificar a que plan corresponde.</li> <li>• No hay unidad de aprendizaje sin clave.</li> <li>• Si ya existe enviar notificación al usuario y dar la opción de mostrar la coincidencia existente.</li> </ul>	

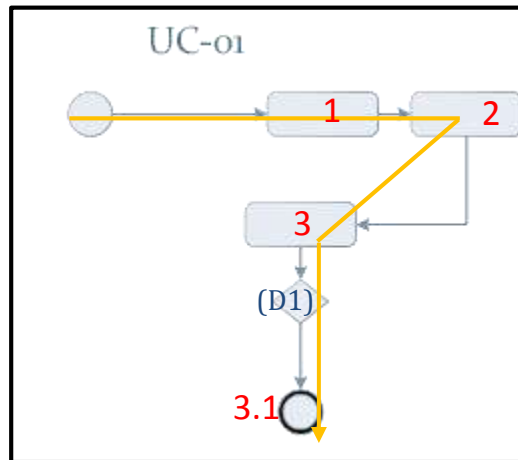
Para determinar las rutas principales por las que un usuario accede al sistema con el fin de verificar el correcto funcionamiento de este, se diseñaron varios escenarios expuestos en la figura 4.16, los escenarios están representados de forma gráfica basados en los casos de uso mostrados en la tabla 4.5.



**Figura 4.16.** Rutas. Fuente: Elaboración propia.

- *Escenario uno del caso de uso registrar unidad de aprendizaje*

En la imagen 4.17 se muestra de forma gráfica la ruta que sigue el escenario uno del caso de uso para registrar unidad de aprendizaje.



**Figura 4.17.** Registra Escenario 1.  
Fuente: Elaboración propia.

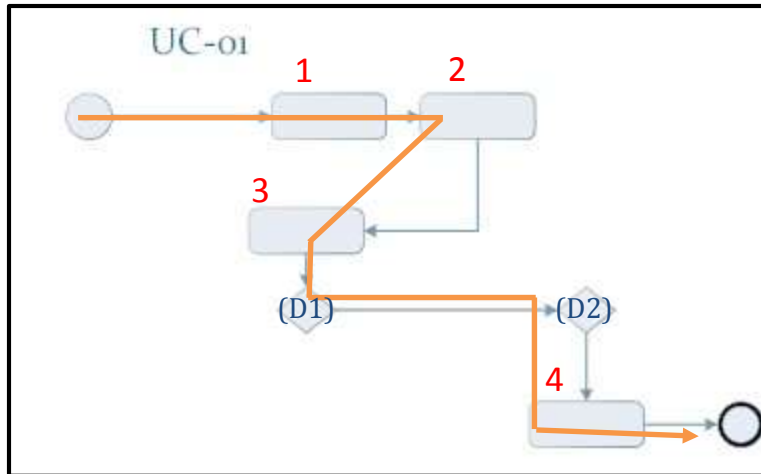
En la tabla 4.6 se puede apreciar los pasos que contempla el escenario uno del caso de uso registrar unidad de aprendizaje.

**Tabla 4.6.** Registrar escenario 1. Fuente: Elaboración propia.

Escenario 1
<b>01, 02, 03, D1 (CANCELAR), 3.1</b>
<b>Descripción:</b>
1. El usuario solicita registrar unidad de aprendizaje. 2. El sistema solicita la información de la unidad de aprendizaje. 3. El usuario introduce la información de la unidad de aprendizaje. <b>(D1) Introducir.</b> <b>3.1 El usuario cancela la operación y este caso de uso termina.</b>

- *Escenario dos del caso de uso registrar unidad de aprendizaje*

En la figura 4.18 se muestra de forma gráfica la ruta que sigue el escenario dos del caso de uso para registrar unidad de aprendizaje.



**Figura 4.18.** Registra Escenario 2. Fuente: Elaboración propia.

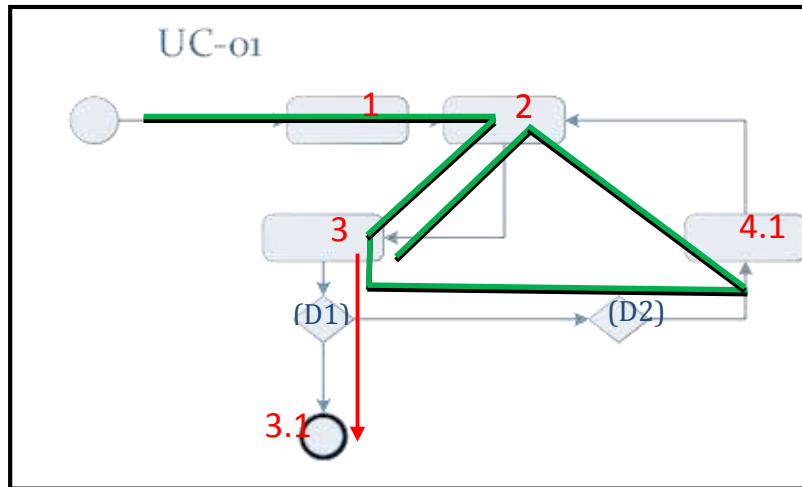
En la tabla 4.7 se puede apreciar los pasos que contempla el escenario dos del caso de uso registrar unidad de aprendizaje.

**Tabla 4.7.** Registrar escenario 2. Fuente: Elaboración propia.

Escenario 2
<b>01, 02, 03, D1 (INTRODUCIR), D2 (VÁLIDO), 04 (NUEVO REGISTRO)</b>
<b>Descripción:</b>
1. El usuario solicita registrar unidad de aprendizaje.
2. El sistema solicita la información de la unidad de aprendizaje.
3. El usuario introduce la información de la unidad de aprendizaje.
<b>(D1). Introducir</b>
<b>(D2).Válido</b>
4. El sistema almacena el nuevo registro.

- *Escenario tres del caso de uso registrar unidad de aprendizaje*

En la figura 4.19 se muestra de forma gráfica la ruta que sigue el escenario tres del caso de uso para registrar unidad de aprendizaje.



**Figura 4.19.** Registra Escenario 3. Fuente: Elaboración propia.

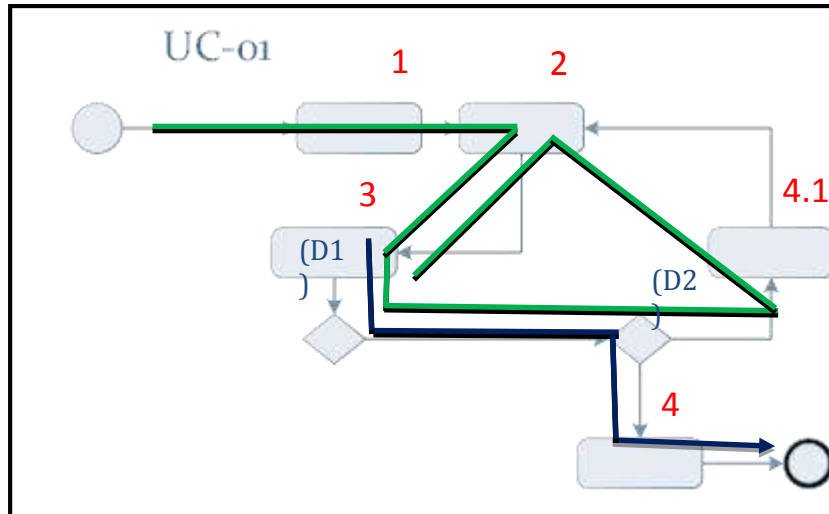
En la tabla 4.8 se puede apreciar los pasos que contempla el escenario tres del caso de uso registrar unidad de aprendizaje.

**Tabla 4.8.** Registrar escenario 3. Fuente: Elaboración propia.

Escenario 3
01 ,02, 03, D1 (INTRODUCIR), D2 (INVÁLIDO) 4.1, 02, 03, D1 (INTRODUCIR), 3.1 (CANCELAR)
<b>Descripción:</b>
1. El usuario solicita registrar unidad de aprendizaje. 2. El sistema solicita la información de la unidad de aprendizaje. 3. El usuario introduce la información de la unidad de aprendizaje. <b>(D1) Introducir.</b> <b>(D2) Inválido</b> 4.1. Mensaje de error 2. El sistema solicita la información de la unidad de aprendizaje. 3. El usuario introduce la información de la unidad de aprendizaje. <b>(D1) Introducir</b> 3.1 El usuario cancela la operación y este caso de uso termina.

- *Escenario cuatro del caso de uso registrar unidad de aprendizaje*

En la figura 4.20 se muestra de forma gráfica la ruta que sigue el escenario cuatro del caso de uso para registrar unidad de aprendizaje.



**Figura 4.20.** Registra Escenario 4. Fuente: Elaboración propia.

En la tabla 4.9 se puede apreciar los pasos que contempla el escenario tres del caso de uso registrar unidad de aprendizaje.

**Tabla 4.9.** Registrar escenario 4. Fuente: Elaboración propia.

Escenario 4
<b>01, 02, 03, D1 (INTRODUCIR), D2 (INVÁLIDO), 04, 02, 03, D1 (INTRODUCIR), D2 (VÁLIDO), 05</b>
<b>Descripción:</b>
1. El usuario solicita registrar unidad de aprendizaje. 2. El sistema solicita la información de la unidad de aprendizaje. 3. El usuario introduce la información de la unidad de aprendizaje. <b>(D1) Introducir.</b> <b>(D2) Inválido</b> 4.1. Mensaje de error 2. El sistema solicita la información de la unidad de aprendizaje. 3. El usuario introduce la información de la unidad de aprendizaje. <b>(D1). Introducir</b> <b>(D2). Válido</b> 4. El sistema almacena el nuevo registro.

- *Consulta unidad de aprendizaje.*

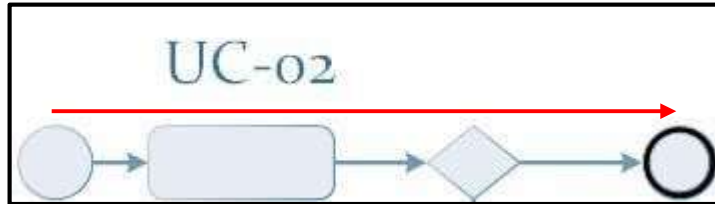
En la tabla 4.10 se aprecia el caso de uso para consultar una unidad de aprendizaje.

**Tabla 4.10.** Consultar nueva unidad de aprendizaje. Fuente: Elaboración propia.

NOMBRE	UC-02. Consultar unidad de aprendizaje
Precondición	Ingresar al sistema.
Secuencia principal	<ol style="list-style-type: none"> <li>1. El usuario solicita consulta de nueva unidad de aprendizaje.</li> <li>2. El sistema muestra mapa curricular.</li> <li>3. El usuario selecciona unidad de aprendizaje.</li> <li>4. El sistema muestra la consulta.</li> </ol>
Error/secuencias alternativas	<p>2.1. Si no existen unidades de Aprendizaje o el sistema no puede recuperar las unidades, se muestra un error y el caso se termina.</p> <p>3.1 El usuario cancela la operación y este caso de uso termina.</p> <p>3,2 El usuario desea una consulta personalizada se ejecuta el caso de uso <b>“consulta personalizada”</b> y continua el caso de uso.</p> <p>4.1 Si la consulta no tiene opciones seleccionadas el sistema muestra un mensaje de error y solicita de nuevo la información de la consulta.</p>
Post-condición	No.
Notas	<p>Por defecto el sistema selecciona unidad Académica “Facultad de Contaduría y Administración”.</p> <p>Por defecto el sistema selecciona Campus “Tijuana”.</p>

- *Escenario uno del caso de uso consultar unidad de aprendizaje*

En la figura 4.21 se muestra de forma gráfica la ruta que sigue el escenario uno del caso de uso para consultar unidad de aprendizaje.



**Figura 4.21.** Consultar unidad de aprendizaje escenario 1.

*Fuente: Elaboración propia.*

En la tabla 4.11 se puede apreciar los pasos que contempla el escenario uno del caso de uso consultar unidad de aprendizaje.

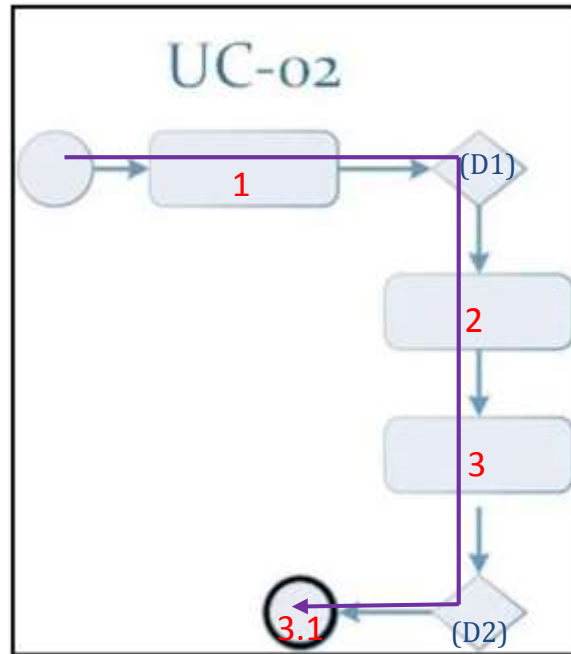
**Tabla 4.11.** Consultar unidad de aprendizaje escenario 1.

*Fuente: Elaboración propia.*

<b>Escenario 1</b>
<b>01, D1, 2.1</b>
<b>Descripción:</b>
1. El usuario solicita consulta de nueva unidad de aprendizaje.
<b>(D1) ERROR</b>
2.1. Si no existen unidades de Aprendizaje o el sistema no puede recuperar las unidades, se muestra un error y el caso se termina.

- *Escenario dos del caso de uso consultar unidad de aprendizaje*

En la figura 4.22 se muestra de forma gráfica la ruta que sigue el escenario dos del caso de uso para consultar unidad de aprendizaje.



**Figura 4.22.** Consultar unidad de aprendizaje escenario 2.  
Fuente: Elaboración propia.

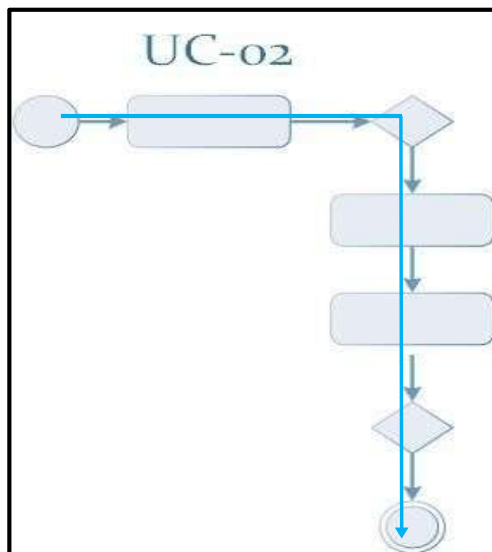
En la tabla 4.12 se puede apreciar los pasos que contempla el escenario dos del caso de uso consultar unidad de aprendizaje.

**Tabla 4.12.** Consultar unidad de aprendizaje escenario 2.  
Fuente: Elaboración propia.

Escenario 2
<b>01,02, 03, D1, 3.1</b>
<b>Descripción:</b>
1. El usuario solicita consulta de nueva unidad de aprendizaje.
<b>(D1)</b>
2. El sistema muestra mapa curricular.
3. El usuario selecciona unidad de aprendizaje.
<b>(D2) CANCELAR</b>
3.1. El usuario cancela la operación y este caso de uso termina.

- Escenario tres del caso de uso consultar unidad de aprendizaje

En la imagen 4.23 se muestra de forma gráfica la ruta que sigue el escenario tres del caso de uso para consultar unidad de aprendizaje.



**Figura 4.23.** Consultar unidad de aprendizaje escenario 3.  
Fuente: Elaboración propia.

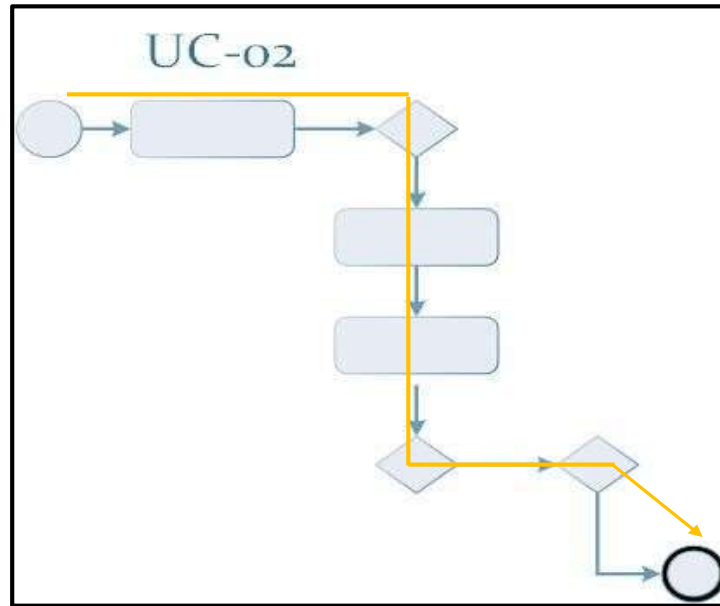
En la tabla 4.13 se puede apreciar los pasos que contempla el escenario tres del caso de uso consultar unidad de aprendizaje.

**Tabla 4.13.** Consultar unidad de aprendizaje escenario 3.  
Fuente: Elaboración propia.

Escenario 3
<b>01, 02, 03, D1, 3.2</b>
<b>Descripción:</b>
1. El usuario solicita consulta de nueva unidad de aprendizaje. <b>(D1)</b>
2. El sistema muestra mapa curricular.
3. El usuario selecciona unidad de aprendizaje. <b>(D2) PERSONALIZAR</b>
3.2. El usuario desea una consulta personalizada se ejecuta el caso de uso “consulta personalizada” y continua el caso de uso.

- *Escenario cuatro del caso de uso consultar unidad de aprendizaje*

En la figura 4.24 se muestra de forma gráfica la ruta que sigue el escenario cuatro del caso de uso para consultar unidad de aprendizaje.



**Figura 4.24.** Consultar unidad de aprendizaje escenario 4.  
Fuente: Elaboración propia.

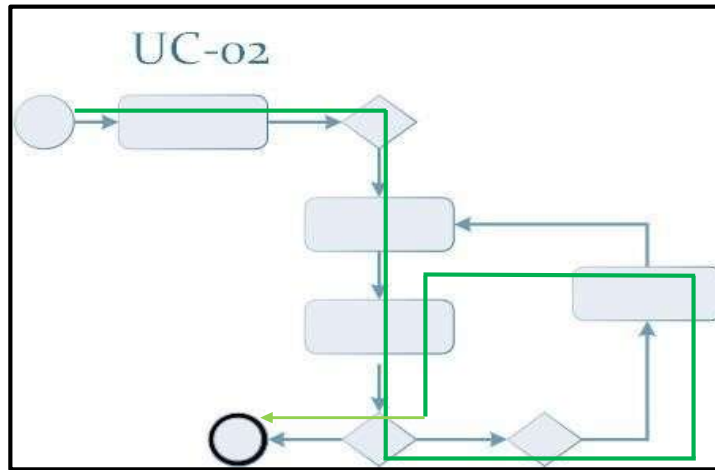
En la tabla 4.14 se puede apreciar los pasos que contempla el escenario cuatro del caso de uso consultar unidad de aprendizaje.

**Tabla 4.14.** Consultar unidad de aprendizaje escenario 4.  
Fuente: Elaboración propia.

Escenario 4
<b>01,02, 03, D1, 4</b>
<b>Descripción:</b>
1. El usuario solicita consulta de nueva unidad de aprendizaje. <b>(D1)</b>
2. El sistema muestra mapa curricular.
3. El usuario selecciona unidad de aprendizaje. <b>(D2) SELECCIÓN</b>
<b>(D3) CONSULTA</b>
El sistema muestra la consulta.

- *Escenario cinco del caso de uso consultar unidad de aprendizaje*

En la figura 4.25 se muestra de forma gráfica la ruta que sigue el escenario cinco del caso de uso para consultar unidad de aprendizaje.



**Figura 4.25.** Consultar unidad de aprendizaje escenario 5.  
Fuente: Elaboración propia.

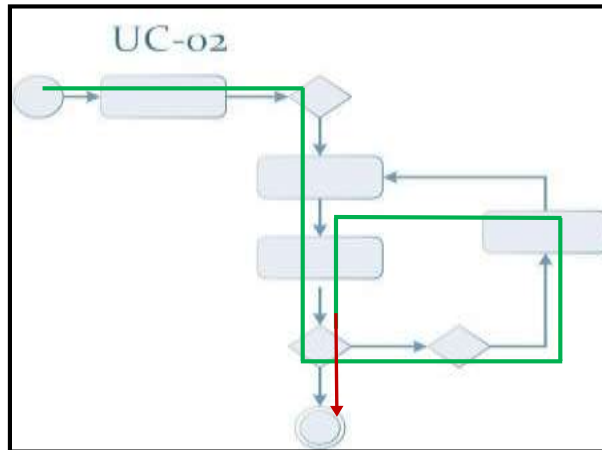
En la tabla 4.15 se puede apreciar los pasos que contempla el escenario cinco del caso de uso consultar unidad de aprendizaje.

**Tabla 4.15.** Consultar unidad de aprendizaje escenario 5.  
Fuente: Elaboración propia.

Escenario 5
<b>01, D1, 02, 03, D2, D3, 4.1 , 02, 03, D2, 3.1</b>
<b>Descripción:</b>
1. El usuario solicita consulta de nueva unidad de aprendizaje. <b>(D1)</b>
2. El sistema muestra mapa curricular.
3. El usuario selecciona unidad de aprendizaje. <b>(D2) SELECCIÓN</b>
<b>(D3) ERROR</b>
4.1 Si la consulta no tiene opciones seleccionadas el sistema muestra un mensaje de error y solicita de nuevo la información de la consulta.
2. El sistema muestra mapa curricular.
3. El usuario selecciona unidad de aprendizaje. <b>(D2) CANCELAR</b>
3.1 El usuario cancela la operación y este caso de uso termina.

- *Escenario seis del caso de uso consultar unidad de aprendizaje*

En la figura 4.26 se muestra de forma gráfica la ruta que sigue el escenario seis del caso de uso para consultar unidad de aprendizaje.



**Figura 4.26.** Consultar unidad de aprendizaje escenario 6.  
Fuente: Elaboración propia.

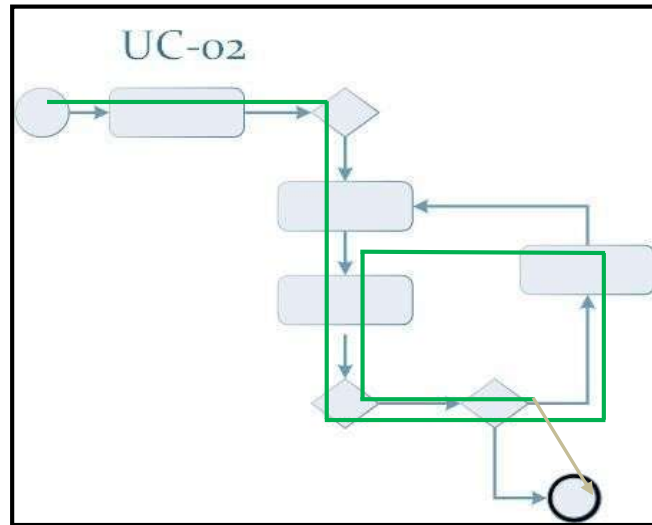
En la tabla 4.16 se puede apreciar los pasos que contempla el escenario seis del caso de uso consultar unidad de aprendizaje.

**Tabla 4.16.** Consultar unidad de aprendizaje escenario 6.  
Fuente: Elaboración propia.

Escenario 6
<b>01, D1, 02, 03, D2, D3, 4.1 , 02, 03, D2, 3.2</b>
<b>Descripción:</b>
1. El usuario solicita consulta de nueva unidad de aprendizaje. <b>(D1)</b> 2. El sistema muestra mapa curricular. 3. El usuario selecciona unidad de aprendizaje. <b>(D2) SELECCIÓN</b> <b>(D3) ERROR</b> 4.1. Si la consulta no tiene opciones seleccionadas el sistema muestra un mensaje de error y solicita de nuevo la información de la consulta. 2. El sistema muestra mapa curricular. 3. El usuario selecciona unidad de aprendizaje. <b>(D2) PERSONALIZAR</b> 3.2. El usuario desea una consulta personalizada se ejecuta el caso de uso “consulta personalizada” y continua el caso de uso.

- *Escenario siete del caso de uso consultar unidad de aprendizaje*

En la figura 4.27 se muestra de forma gráfica la ruta que sigue el escenario siete del caso de uso para consultar unidad de aprendizaje.



**Figura 4.27.** Consultar unidad de aprendizaje escenario 7.  
Fuente: Elaboración propia.

En la tabla 4.17 se puede apreciar los pasos que contempla el escenario siete del caso de uso consultar unidad de aprendizaje.

**Tabla 4.17.** Consultar unidad de aprendizaje escenario 7. Fuente: Elaboración propia.

Escenario 7
<b>01, D1, 02, 03, D2, D3, 4.1 , 02, 03, D2, D3, 04</b>
<b>Descripción:</b>
1. El usuario solicita consulta de nueva unidad de aprendizaje. <b>(D1)</b> 2. El sistema muestra mapa curricular. 3. El usuario selecciona unidad de aprendizaje. <b>(D2) SELECCIÓN</b> <b>(D3) ERROR</b> 4.2 Si la consulta no tiene opciones seleccionadas el sistema muestra un mensaje de error y solicita de nuevo la información de la consulta. 2. El sistema muestra mapa curricular. 3. El usuario selecciona unidad de aprendizaje. <b>(D2) SELECCIÓN</b> <b>(D3) CONSULTA</b> 4. El sistema muestra la consulta.

- *Consulta personalizada de unidad de aprendizaje.*

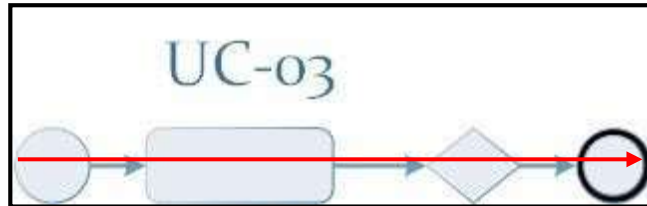
En la tabla 4.18 se muestra el caso de uso consulta personalizada.

**Tabla 4.18.** Consulta personalizada. Fuente: *Elaboración propia.*

NOMBRE	UC-03. Consulta personalizada
<b>Precondición</b>	El usuario está realiza una consulta.
<b>Secuencia principal</b>	<ol style="list-style-type: none"> <li>1. El usuario solicita consulta personalizada.</li> <li>2. El sistema muestra todos los campos disponibles.</li> <li>3. El usuario selecciona los campos.</li> <li>4. El sistema muestra la consulta.</li> </ol>
<b>Error/secuencias alternativas</b>	<ol style="list-style-type: none"> <li>2.1. Si no hay ningún campo o el sistema no puede recuperar los campos, se muestra un error y el caso se termina.</li> <li>3.1. El usuario cancela la operación y este caso de uso termina.</li> </ol>
<b>Post-condición</b>	No.
<b>Notas</b>	

- *Escenario uno de caso de uso consulta personalizada.*

En la figura 4.28 se muestra de forma gráfica la ruta que sigue el escenario uno del caso de uso para consulta personalizada de unidad de aprendizaje.



**Figura 4.28.** Consulta Personalizada UA Escenario 1.  
Fuente: Elaboración propia.

En la tabla 4.19 se puede apreciar los pasos que contempla el escenario uno del caso de uso consulta personalizada de unidad de aprendizaje.

**Tabla 4.19.** Consulta personalizada UA escenario 1.  
Fuente: Elaboración propia.

<b>Escenario 1</b>
<b>01, D1, 2.1</b>
<b>Descripción:</b>
1. El usuario solicita consulta personalizada.
<b>(D1) Error</b>
2.1. Si no hay ningún campo o el sistema no puede recuperar los campos, se muestra un error y el caso se termina.

- *Eliminar unidad de aprendizaje.*

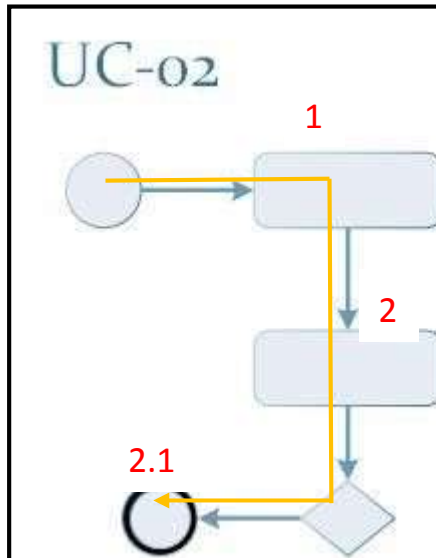
En la tabla 4.20 se aprecia el caso de uso cuatro que contiene la secuencia para eliminar una unidad de aprendizaje.

**Tabla 4.20.** *Eliminar unidad de aprendizaje. Fuente: Elaboración propia.*

NOMBRE	UC-04. Eliminar unidad de aprendizaje
<b>Precondición</b>	Ninguna
<b>Secuencia principal</b>	<ol style="list-style-type: none"> <li>1. El usuario solicita Eliminar unidad de aprendizaje.</li> <li>2. El sistema muestra mapa curricular.</li> <li>3. El usuario selecciona unidad de aprendizaje.</li> <li>4. El sistema borra unidad de aprendizaje.</li> </ol>
<b>Error/secuencias alternativas</b>	<ol style="list-style-type: none"> <li>2.1. Si no existen unidades de Aprendizaje o el sistema no puede recuperar unidades de Aprendizaje, se muestra un error y el caso se termina.</li> <li>3.1. El usuario cancela la operación y este caso de uso termina.</li> <li>4.1 Si no existen unidades de Aprendizaje seleccionada el sistema muestra un mensaje de error y solicita de nuevo la selección de la unidad de aprendizaje.</li> </ol>
<b>Post-condición</b>	unidad de aprendizaje Eliminada.
<b>Notas</b>	

- *Escenario uno del caso de uso eliminar unidad de aprendizaje.*

En la figura 4.29 se muestra de forma gráfica la ruta que sigue el escenario uno del caso de uso para eliminar unidad de aprendizaje.



**Figura 4.29.** Eliminar UA Escenario 1.  
Fuente: Elaboración propia.

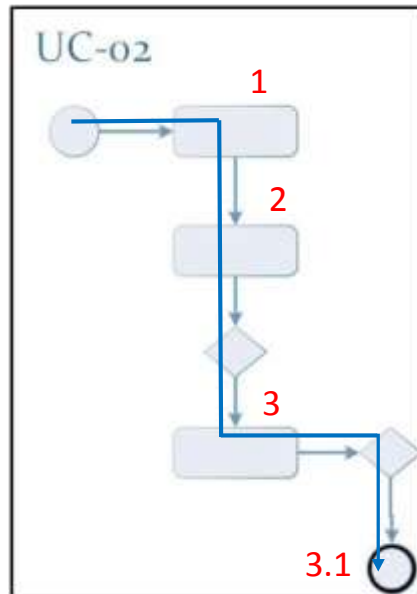
En la tabla 4.21 se puede apreciar los pasos que contempla el escenario uno del caso de uso eliminar unidad de aprendizaje.

**Tabla 4.21.** Eliminar UA escenario 1. Fuente: Elaboración propia.

<b>Escenario 1</b>
<b>01,02, D1, 2.1</b>
<b>Descripción:</b>
1. El usuario solicita Eliminar unidad de aprendizaje.
2. El sistema muestra mapa curricular.
<b>(D1) ERROR</b>
2.1 Si no existen unidades de Aprendizaje o el sistema no puede recuperar unidades de Aprendizaje, se muestra un error y el caso se termina.

- *Escenario dos del caso de uso eliminar unidad de aprendizaje.*

En la figura 4.30 se muestra de forma gráfica la ruta que sigue el escenario dos del caso de uso para eliminar unidad de aprendizaje.



**Figura 4.30.** Eliminar UA Escenario 2.  
Fuente: Elaboración propia.

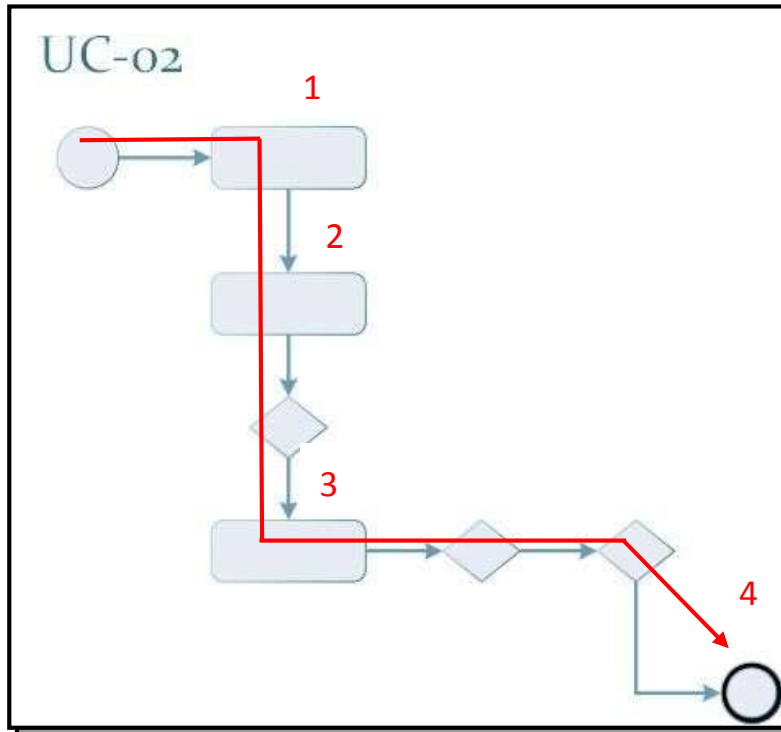
En la tabla 4.22 se puede apreciar los pasos que contempla el escenario dos del caso de uso eliminar unidad de aprendizaje.

**Tabla 4.22.** Eliminar UA escenario 2. Fuente: Elaboración propia.

Escenario 2
<b>01,02, D1, 03, D2, 3.1</b>
<b>Descripción:</b>
1. El usuario solicita Eliminar unidad de aprendizaje.
2. El sistema muestra mapa curricular.
<b>(D1) SELECCIONAR</b>
3. El usuario selecciona unidad de aprendizaje.
<b>(D2) CANCELAR</b>
3.1. El usuario cancela la operación y este caso de uso termina.

- *Escenario tres de caso de uso eliminar unidad de aprendizaje.*

En la figura 4.31 se muestra de forma gráfica la ruta que sigue el escenario tres del caso de uso para eliminar unidad de aprendizaje.



**Figura 4.31.** Eliminar UA Escenario 3. Fuente: Elaboración propia.

En la tabla 4.23 se puede apreciar los pasos que contempla el escenario tres del caso de uso eliminar unidad de aprendizaje.

**Tabla 4.23.** Eliminar UA escenario 3. Fuente: Elaboración propia.

Escenario 3
<b>01, 02, 03, D1, D2, 04</b>
<b>Descripción:</b>
1. El usuario solicita Eliminar unidad de aprendizaje.
2. El sistema muestra mapa curricular.
3. El usuario selecciona unidad de aprendizaje.
<b>(D1) SELECCIONAR</b>
<b>(D2) BORRAR</b>
4. El sistema borra unidad de aprendizaje.



- *Modificar unidad de aprendizaje.*

En la tabla 4.25 se muestra el caso de uso para modificar unidad de aprendizaje.

**Tabla 4.25.** *Modificar unidad de aprendizaje. Fuente: Elaboración propia.*

NOMBRE	UC-06. Modificar unidad de aprendizaje
<b>Precondición</b>	Ninguna
<b>Secuencia principal</b>	<ol style="list-style-type: none"> <li>1. El usuario solicita Modificar unidad de aprendizaje.</li> <li>2. El sistema muestra mapa curricular.</li> <li>3. El usuario selecciona unidad de aprendizaje.</li> <li>4. El sistema solicita la información de la unidad de aprendizaje.</li> <li>5. El usuario introduce la nueva información de la unidad de aprendizaje.</li> <li>6. El sistema muestra los datos capturados.</li> <li>7. El usuario selecciona la opción guardar.</li> <li>8. El sistema Guarda unidad de aprendizaje.</li> </ol>
<b>Error/secuencias alternativas</b>	<ol style="list-style-type: none"> <li>2.1. Si no existen Unidades de Aprendizaje o el sistema no puede recuperar unidades de aprendizaje, se muestra un error y el caso se termina.</li> <li>3.1. El usuario cancela la operación y este caso de uso termina.</li> <li>7.1 El usuario selecciona la opción de modificación y solicita introducir de nuevo la información de la unidad de aprendizaje.</li> </ol>
<b>Post-condición</b>	El sistema muestra un mensaje diciendo “Alta exitosa”.
<b>Notas</b>	La clave de la unidad de aprendizaje no se debe permitir modificarse.

- *Registra primera carga académica.*

En la tabla 4.26 se muestra el caso de uso para ingresar primera carga académica.

**Tabla 4.26.** Registrar primera carga académica. Fuente: Elaboración propia.

NOMBRE	UC-01. Ingresar primera carga académica
Precondición	Ingresar al sistema.
Secuencia principal	<ol style="list-style-type: none"> <li>1. El usuario solicita Ingresar a Registro de carga académica.</li> <li>2. El sistema muestra Plan de estudio vigente.</li> <li>3. El usuario Ingresa período.</li> <li>4. El sistema activa y desactiva unidades de aprendizaje.</li> <li>5. El usuario registra clave, turno y semestre.</li> <li>6. El usuario selecciona grupos y pulsa guardar.</li> <li>7. El sistema almacena la nueva carga académica.</li> </ol>
Error/secuencias alternativas	
Post-condición	No.
Notas	

- *Registra segunda carga académica.*

En la tabla 4.27 se observa el caso de uso para ingresar primera carga académica.

**Tabla 4.27.** Registrar segunda carga académica. Fuente: Elaboración propia.

NOMBRE	UC-02. Ingresar segunda carga académica
Precondición	Ingresar al sistema.
Secuencia principal	<ol style="list-style-type: none"> <li>1. El usuario solicita ingresar a registro de carga académica.</li> <li>2. El sistema muestra plan de estudio vigente.</li> <li>3. El usuario ingresa período.</li> <li>4. El sistema activa y desactiva unidades de aprendizaje.</li> <li>5. El usuario registra clave, turno y semestre.</li> <li>6. El usuario selecciona grupos y pulsa guardar.</li> <li>7. El sistema almacena la nueva carga académica.</li> </ol>
Error/secuencias alternativas	3.1 El sistema no puede continuar porque no existe grupo. Caso de uso UC-01.
Post-condición	No.
Notas	

- *Consulta carga académica*

En la tabla 4.28 se muestra el caso de uso para consultar carga académica.

**Tabla 4.28.** Consultar carga académica. Fuente: Elaboración propia.

NOMBRE	UC-03. Consultar carga académica
<b>Precondición</b>	Ingresar al sistema.
<b>Secuencia principal</b>	<ol style="list-style-type: none"> <li>1. El usuario solicita Ingresar a consulta de carga académica.</li> <li>2. El sistema muestra la última carga académica registrada.</li> <li>3. El usuario selecciona consulta por oferta de UA's.</li> <li>4. El sistema muestra consulta seleccionada.</li> </ol>
<b>Error/secuencias alternativas</b>	<p>3.1 No existe la oferta por UA. El usuario selecciona consulta por oferta de materias por período.</p> <p>3.2 No existe el período. El usuario selecciona consulta por oferta de materias por carrera.</p> <p>3.3 No existe la carrera. El usuario selecciona consulta por oferta de materias por semestre.</p> <p>3.4 No existe el semestre. El usuario selecciona consulta por oferta de materias por grupo.</p> <p>3.5 No existe el grupo. El usuario selecciona consulta por oferta de materias por turno.</p> <p>3.5 No existe el turno. El usuario selecciona consulta por combinación de campos.</p>
<b>Post-condición</b>	No.
<b>Notas</b>	

- *Modificar carga académica*

En la tabla 4.29 se muestra el caso de uso para Modificar carga académica.

**Tabla 4.29.** *Modificar carga académica. Fuente: Elaboración propia.*

NOMBRE	UC-04. Modificar carga académica
<b>Precondición</b>	Tener una consulta de carga académica.
<b>Secuencia principal</b>	<ol style="list-style-type: none"> <li>1. El usuario solicita la modificación de la carga académica.</li> <li>2. El sistema comprueba si la carga está autorizada.</li> <li>3. El sistema muestra la pantalla de modificación.</li> <li>4. El usuario modifica los atributos y pulsa guardar.</li> <li>5. El sistema guarda los cambios</li> <li>6. El sistema muestra mensaje de registro actualizado y pregunta modificación de catálogo de períodos.</li> <li>7. EL usuario modifica atributos y pulsa guardar.</li> <li>8. El sistema muestra mensaje de registro actualizado y pregunta modificación de catálogo de grupos.</li> <li>9. El usuario modifica atributos y pulsa guardar.</li> <li>10. El sistema muestra mensaje de registro actualizado</li> </ol>
<b>Error/secuencias alternativas</b>	<p>2.1 La carga académica no está autorizada y el caso de uso termina.</p> <p>6.1 El usuario no desea modificar catálogo de períodos y el caso de uso termina.</p> <p>8.1 el usuario no desea cambiar catálogo de grupos y el caso de uso termina.</p>
<b>Post-condición</b>	No.
<b>Notas</b>	

- *Eliminar carga académica*

En la tabla 4.30 se muestra el caso de uso para Eliminar carga académica.

**Tabla 4.30.** *Eliminar carga académica. Fuente: Elaboración propia.*

NOMBRE	UC-05. Eliminar carga académica
<b>Precondición</b>	Tener una consulta de carga académica.
<b>Secuencia principal</b>	<ol style="list-style-type: none"> <li>1. El usuario solicita la modificación de la carga académica.</li> <li>2. El sistema muestra consulta solicitada.</li> <li>3. El usuario desactiva UA a eliminar y pulsa el botón guardar.</li> <li>4. El sistema muestra UA desactivada.</li> </ol>
<b>Error/secuencias alternativas</b>	
<b>Post-condición</b>	No.
<b>Notas</b>	

### 4.3.3.3 Ejecución de las pruebas.

Después de elaborar los caso de uso se procedió con la ejecución de las pruebas basado en los casos de uso diseñados, para ello se utilizó la matriz que se muestra en la tabla 4.31 con los campos necesario para verificar si el sistema cumplía los requerimientos basado en las funcionalidades del mismo.

**Tabla 4.31.** Ejecución de pruebas. Fuente: Elaboración propia.

Sección: Plan de Estudio - Registro							
Caso de prueba	Cancelar registro.		ID: UC01ES01	Severidad	1a - Fatal	FECHA: 21-04-2014	
			Dependencia de red: Yes				
Propósito de la prueba.							
Criterios de prueba	Test Step		Step Result	Test Steps			
	No.	Description	Description	Pass	Fail	N/A	Exención
	1	El usuario solicita Registrar Unidad de Aprendizaje.		X			
	2	El sistema solicita la información de la Unidad de Aprendizaje.		X			
	3	El usuario introduce la información de la Unidad de Aprendizaje.				X	
4	El usuario cancela la operación y este caso de uso termina.				X		
Overall Result				2	0	2	0
Resultados observados en caso de Fallo.							
Descripción del impacto en el usuario final							
Comentarios							

Al finalizar cada rutina de pruebas se le entrego al equipo los resultados para validar que el sistema funcionaba de forma correcta, o en su defecto, corregir las partes que no lo hicieran.

Después de que se tuvo una versión funcional del segundo módulo del sistema se procedió a realizar las rutinas de prueba en la tabla 4.32 se aprecia la matriz que contiene la información de una funcionalidad el sistema para evaluar su

desempeño. La rutina de pruebas es iterativa por lo que se realizaban cada vez que se liberaba una nueva versión, en el anexo 2 se muestran varias matrices sobre esta etapa.

**Tabla 4.32. Ejecución de pruebas carga académica. Fuente: Elaboración propia.**

Sección: <b>Carga Académica - Consulta</b>								
Caso de prueba	Consultar carga		ID:	UC02ES01	Severidad	1a - Fatal	FECHA:	21-08-2014
			Dependencia de red:	Yes				
Propósito de la prueba.								
Test Criteria	Test Step		Step Result		Test Steps			
	No.	Description	Description		Pass	Fail	N/A	Exención
	1	1. El usuario solicita Ingresar a Consulta de carga académica.			X			
	2	2. El sistema muestra la última carga académica registrada.			X			
	3	3. El usuario selecciona consulta por oferta de UA's.			X			
	4	4. El sistema muestra consulta seleccionada.			X			
Overall Result								
Resultados observados en caso de Fallo.	No es posible realizar una consulta.							
End User Impact / Waiver description								
Comments								

#### 4.3.4 Ajustes en el entorno de operación

En el desarrollo de esta fase se hicieron algunas modificaciones en el servicio vsftpd. Este servicio permite gestionar los archivos del servidor de forma remota. Las configuraciones de ftp fueron necesarias hasta este punto porque las versiones liberadas del sistema eran más frecuentes, debido a esto se crea la necesidad de utilizar algún medio para llevar las actualizaciones de forma rápida y sencilla.

Las modificaciones al servicio de ftp se hicieron con el fin de poder actualizar SIGAF sin tener que acceder físicamente al servidor. De forma predeterminada el servicio vsftpd no permite el acceso al servidor como root. Root es un usuario que tiene permisos para hacer modificaciones a cualquier archivo o servicio que se encuentre en el servidor, como la ruta donde se encuentra el sistema está fuera del acceso de un usuario normal, es necesario que se ingrese con permisos de súper usuario, es decir, con todos los permisos para gestionar archivos ubicados en cualquier ruta.

- Acceso root

Para permitir el acceso al servidor con el usuario root se hicieron modificaciones en los archivos ftpusers y user\_list ubicados en la ruta /etc/vsftpd.

En cada uno de los archivos mencionados anteriormente se eliminó el usuario root, ya que estos archivos contienen la lista de los usuarios a los que no se les permite el acceso via ftp.

- SELinux

En este punto se puede reiniciar el servicio vsftpd para comprobar el acceso como root, si no, se puede hacer más adelante. Hay ocasiones en las que al acceder al servidor vía ftp marque un error que imposibilita cambiar al directorio: /root/. Lo que se hizo en el servidor fue deshabilitar SELinux. “SELinux es un módulo de seguridad para el kernel Linux que proporciona el mecanismo para soportar políticas de seguridad para el control de acceso...” (nsa, 2014).

Para deshabilitar este módulo de seguridad se accedió a la terminal y se ejecutó como usuario root el siguiente comando: `setenforce 0`.

## 4.4 Carga de datos al entorno de operación

En este apartado se muestran las distintas versiones del sistema que se cargaron al servidor para comprobar las funcionalidades del mismo. Los datos de la carga se obtuvieron al extraer la última versión de la plataforma GITHUB. En cada versión se muestran las partes del sistema que se modificaron y que posteriormente fueron implementadas. Cada que se liberaba una versión del sistema esta se cargaba al entorno de operación, la clave que tiene cada versión sirve para llevar un control sobre los cambios de cada una y su comportamiento en el servidor.

### Versión 669be4e..1eb656e

En esta versión de incorpora el módulo de plan de estudios, el interfaz del módulo de carga académica y de autenticación.

```
remote: Counting objects: 69, done.
remote: Compressing objects: 100% (54/54), done.
remote: Total 69 (delta 33), reused 22 (delta 14)
Unpacking objects: 100% (69/69), done.
From https://github.com/wolfogan/sigaf
   669be4e..1eb656e  master    -> origin/master
* [new ref]         refs/pull/23/head -> origin/pr/23
* [new ref]         refs/pull/24/head -> origin/pr/24
Updating 669be4e..1eb656e
Fast-forward
 app/controllers/CargaAcademicaController.php      | 10 +-
 .../DisponibilidadDocenteController.php          |  8 +
 app/views/ca/registro.blade.php                  | 621 ++++++++
 app/views/dd/consulta.blade.php                  | 780 ++++++++
 app/views/dd/eliminar.blade.php                  | 250 ++++++
 app/views/dd/registro.blade.php                  | 698 ++++++++
```

```

app/views/dd/registrodisponibilidad.blade.php      | 71 --
app/views/dd/registroestudios.blade.php           | 165 -----
app/views/includes/menu.php                       | 8 +-
app/views/main/index.blade.php                    | 6 +-
app/views/pe/catalogosAdmin.blade.php             | 15 +-
app/views/pe/consulta.blade.php                   | 6 +-
create mode 100644 app/views/dd/consulta.blade.php
create mode 100644 app/views/dd/eliminar.blade.php
delete mode 100644 app/views/dd/registrodisponibilidad.blade.php
delete mode 100644 app/views/dd/registroestudios.blade.php

```

### **Versión 10af0a0..3e887de**

Se incorporan nuevas funcionalidades en el apartado de registro y consulta del módulo de plan de estudios. Se integra el modulo carga académica.

```

remote: Counting objects: 81, done.
remote: Compressing objects: 100% (66/66), done.
remote: Total 81 (delta 51), reused 23 (delta 15)
Unpacking objects: 100% (81/81), done.
From https://github.com/wolfogan/sigaf
   10af0a0..3e887de  master    -> origin/master
* [new ref]         refs/pull/27/head -> origin/pr/27
Updating 10af0a0..3e887de
Fast-forward
 CODIGOS ANTERIORES.txt          | 58 ++
 PENDIENTES.txt                  | 12 +-
 app/controllers/CargaAcademicaController.php | 20 +-
 app/controllers/PlanEstudioController.php     | 28 +-
 app/routes.php                  | 9 +-
 app/views/ca/consulta.blade.php | 22 +-
 app/views/ca/registro.blade.php | 103 ++-
 app/views/ca/registro2.blade.php | 895 ++++++-----
 app/views/dd/registro.blade.php | 75 ++-

```

```
app/views/pe/registro.blade.php          | 73 +--
public/css/estiloPrincipal.css           | 93 ++-
11 files changed, 788 insertions(+), 600 deletions(-)
```

## Versión 10af0a0..3e887de

Existen modificaciones en el diseño del módulo plan de estudio y carga académica. Se agrega el interfaz del módulo disponibilidad docente.

```
remote: Counting objects: 81, done.
remote: Compressing objects: 100% (66/66), done.
remote: Total 81 (delta 51), reused 23 (delta 15)
Unpacking objects: 100% (81/81), done.
From https://github.com/wolfogan/sigaf
   10af0a0..3e887de  master    -> origin/master
* [new ref]         refs/pull/27/head -> origin/pr/27
Updating 10af0a0..3e887de
Fast-forward
 CODIGOS ANTERIORES.txt          | 58 ++
 PENDIENTES.txt                 | 12 +-
 app/controllers/CargaAcademicaController.php | 20 +-
 app/controllers/PlanEstudioController.php   | 28 +-
 app/routes.php                 | 9 +-
 app/views/ca/consulta.blade.php | 22 +-
 app/views/ca/registro.blade.php | 103 ++-
 app/views/ca/registro2.blade.php | 895 ++++++-----
 app/views/dd/registro.blade.php | 75 ++-
 app/views/pe/registro.blade.php | 73 +--
 public/css/estiloPrincipal.css   | 93 ++-
11 files changed, 788 insertions(+), 600 deletions(-)
```

## Versión d8a0233..de5bfe0

Existen modificaciones en el módulo de plan de estudios y carga académica. Se incorpora el módulo de disponibilidad docente el registro y modificación de usuarios dentro del sistema.

```
remote: Counting objects: 40, done.
remote: Compressing objects: 100% (39/39), done.
remote: Total 40 (delta 34), reused 7 (delta 1)
Unpacking objects: 100% (40/40), done.
From https://github.com/wolfogan/sigaf
   d8a0233..de5bfe0 master   -> origin/master
Updating d8a0233..de5bfe0
Fast-forward
 LINKS.txt | 8 +
 app/config/database.php | 2 +-
 app/controllers/PlanEstudioController.php | 16 +-
 app/controllers/UserLoginController.php | 6 +
 app/routes.php | 1 +
 app/views/dd/consulta.blade.php | 608 ++-
 app/views/dd/eliminar.blade.php | 36 +-
 app/views/dd/registro.blade.php | 242 +-
 app/views/includes/menu.php | 4 +-
 app/views/main/index.blade.php | 2 +-
 app/views/pe/registro.blade.php | 241 +-
 app/views/pe/usuariosRegistro.blade.php | 128 -
 app/views/usuarios/usuariosRegistro.blade.php | 334 ++
 public/css/estiloPrincipal.css | 497 +-
 (+), 794 deletions(-)
create mode 100644 LINKS.txt
delete mode 100644 app/views/pe/usuariosRegistro.blade.php
create mode 100644 app/views/usuarios/usuariosRegistro.blade.php
create mode 100644 sigaf_last.sql
```

*Capítulo V*  
*Resultados y Conclusiones*

## 5.1 Resultados

Durante el desarrollo de este proyecto, se analizaron distintos modelos de desarrollo de software. Además se observó la técnica para llevar a cabo la rutina de pruebas sobre el software. La revisión de estos temas permitió establecer bases sólidas para generar un plan de implementación y ejecución de pruebas para una aplicación con arquitectura web llamada SIGAF, herramienta útil para la facultad de contaduría y administración que facilita las tareas de índole académico realizadas en este lugar.

La dirección del proyecto permitió obtener un producto con las siguientes características:

- Servidor independiente, ya que se optó por configurar uno de los servidores de la facultad específicamente para el funcionamiento de SIGAF mediante la instalación de herramientas de uso libre, tales como, sistema operativo centOS, lenguaje de programación php, motor de base de datos mysql, servidor HTTP apache, ftp y ssh, tecnologías que permiten construir un servidor robusto y seguro.
- La arquitectura del servidor permite la escalabilidad, ya que se puede incorporar nuevos módulos a SIGAF que permiten incrementar su funcionalidad o incorporar otros sistemas ajenos a este.

Para verificar que SIGAF fuera funcional en el entorno de operación, que consiste en ejecutar el sistema en el servidor dedicado de la facultad, se logró mediante la validación de los requerimientos establecidos en la etapa de análisis.

De acuerdo a los resultados obtenidos la implementación de SIGAF presenta un nivel de robustez aceptable, las funciones habilitadas en el servidor son las siguientes:

- Acceso al servidor desde la red interna de la facultad y desde fuera mediante conexión a Internet.
  - El sistema cuenta con un módulo llamado disponibilidad docente el cual permite a los maestros de la institución enviar distintos tipos de información académica, de acuerdo a los requerimientos esta tarea la puede hacer directamente en la facultad o desde su casa, lo cual hace necesario habilitar el sistema para que las conexiones tanto de entrada y salida sean permitidas. Para comprobar el acceso del sistema desde una red distinta a la de la facultad se utilizó una computadora fuera de ella, la ip del dispositivo se puede apreciar en la figura 5.1 la cual es proporcionada por una herramienta que se encuentra en el enlace [www.mi-ip.cl](http://www.mi-ip.cl).



**Figura 5.1.** mi-ip.cl. Fuente: mi-ip.cl.

- A la vez, en la figura 5.2 se puede observar que el acceso al servidor está permitido desde cualquier dispositivo, incluso, fuera de las instalaciones donde se encuentra físicamente el servidor.



**Figura 5.2.** Conexión externa. Fuente: SIGAF.

- Es importante mencionar que el acceso al servidor no está limitado solamente por medio del protocolo http, sino que también se puede hacer, como se aprecia en la figura 5.3, mediante el protocolo ssh o ftp, el cual permite hacer actualizaciones al sistema o a la base de datos sin tener que estar frente al servidor.



**Figura 5.3.** Conexión mediante SSH. Fuente: SIGAF.

- Acceso al módulo de autenticación.
  - Debido a que SIGAF era una sistema en desarrollo, el proceso de implementación se llevó a cabo en módulos, esto es, el sistema se dividió en partes más pequeñas de acuerdo a su funcionalidad y al final del desarrollo de cada una de ellas se fue incorporando al servidor para verificar su funcionalidad, en la figura 5.4 podemos ver el módulo de autenticación de SIGAF incorporado en el servidor de la facultad, en este punto el resultado de la conexión de la base de datos con este módulo es como se esperaba.



**Figura 5.4.** Acceso módulo Log In. Fuente: SIGAF.

- Acceso al módulo para crear, modificar y consultar plan de estudios.
  - El segundo módulo incorporado al servidor fue el módulo de plan de estudios, en la figura 5.5 se puede observar el acceso a este apartado junto con la realización de la consulta de un plan anterior 2009-1.



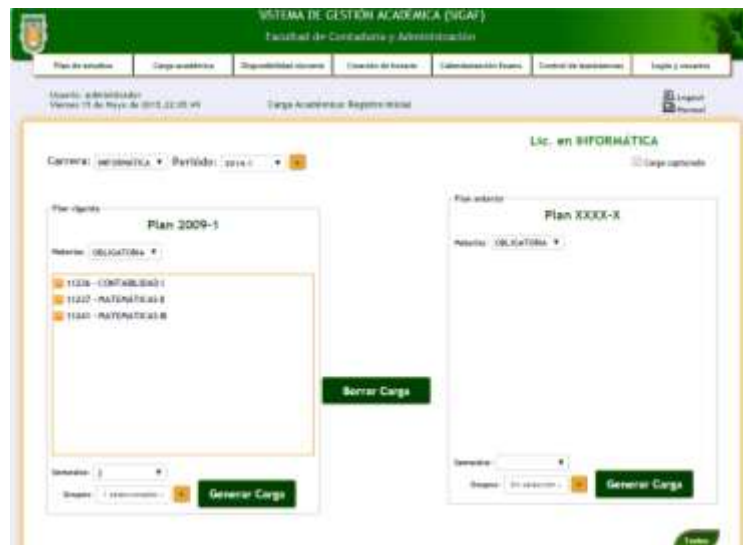
Figura 5.5. Acceso módulo plan de estudios. Fuente: SIGAF.

- La incorporación de este módulo abarca otro apartado donde se puede realizar la consulta o modificación de un plan de estudios o una unidad académica tal como se observa en la figura 5.6.



Figura 5.6. Acceso módulo plan de estudios II. Fuente: SIGAF.

- Acceso a las funciones para registrar, consultar carga académica.
  - Después de la implementación del módulo de plan de estudios se incorpora el de carga académica, en la figura 5.7 se puede observar que los datos incorporados en el módulo anterior pueden ser accedidos desde aquí, aunque los datos son almacenados en la base de datos se tiene que comprobar que el interfaz es capaz de comunicarse con dicha base para obtenerlos.



**Figura 5.7.** Acceso módulo carga académica. Fuente: SIGAF.

- A la vez, como se puede observar en la figura 5.8, los registros incorporados en el bloque de registro de este módulo son consultados satisfactoriamente en el interfaz de consulta.



**Figura 5.8.** Acceso módulo carga académica II. Fuente: SIGAF.

- Acceso al módulo de disponibilidad docente.
  - La implementación de este módulo al sistema se lleva a cabo de acuerdo a lo establecido en los requerimientos del sistema, el cual contempla, como se puede apreciar en la figura 5.9, un apartado para la incorporación de datos personales, estudios y cursos, y finalmente la disponibilidad del docente que sirve como base para el último módulo del sistema.

**Figura 5.9.** Acceso módulo disponibilidad docente. Fuente: SIGAF.

- Acceso al módulo para generar horarios.
  - El módulo de generar horarios que se muestra en la figura 5.10 se implementó en el servidor de la facultad para verificar su funcionamiento, dicho apartado toma los datos ingresados en los módulos anteriores y genera los horarios.
  - De igual forma cuenta con un interfaz que permite consultar las horas frente al grupo basados en los horarios generados con anterioridad.



Figura 5.10. Acceso módulo disponibilidad docente II. Fuente: SIGAF.

- Registro y modificación de usuarios.
  - El último módulo incorporado al servidor fue el de gestión de usuarios, en la figura 5.11 se aprecia el interfaz que permite registrar y modificar usuarios, la función de este módulo se ve reflejada al momento de iniciar sesión al sistema.

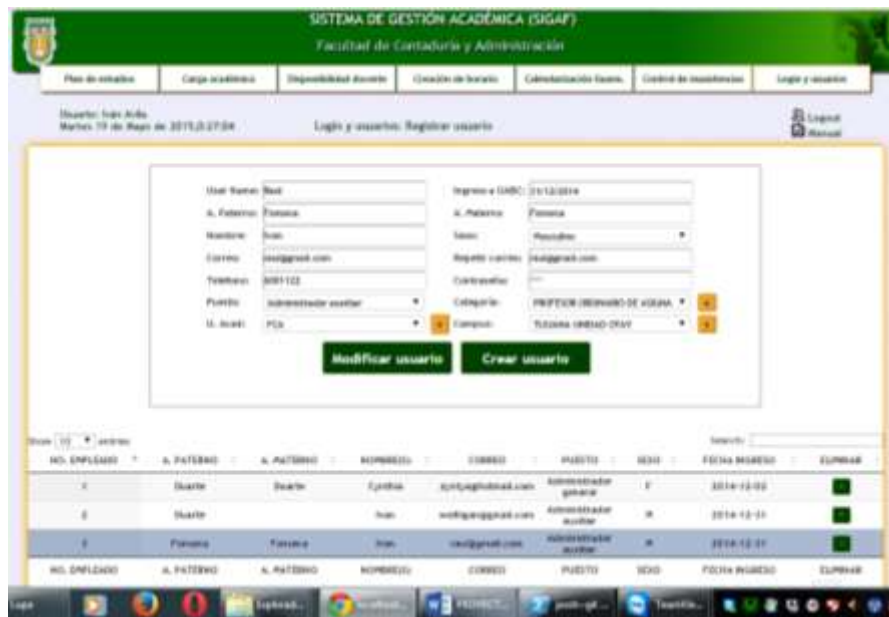


Figura 5.11. Acceso módulo usuarios. Fuente: SIGAF.

## 5.2 Conclusiones

La elaboración de horarios ha sido una actividad común en todas las organizaciones educativas, ya que a partir de este documento se distribuye la carga de trabajo de los docentes, y a la vez, facilita que el coordinador académico, el subdirector y el director tomen decisiones sobre la distribución de las materias con sus respectivas horas clase, la implementación de un sistema de gestión académica con arquitectura Web que consiste en la instalación y configuración de una serie de herramientas para realizar las tareas administrativas de esta naturaleza, permiten mayor facilidad en su elaboración.

Actualmente la facultad cuenta con un software que no es el óptimo para sistematizar y automatizar el desarrollo las tareas relacionadas con la generación de horarios ya que no permite generar informes ni gráficos y tampoco cuenta con documentación alguna para incorporarlo a otros entornos de operación, lo que hace difícil unificar el trabajo de los involucrados en el proceso, siendo nuestro objetivo principal implementar el Sistema de Gestión Académica FCA (SIGAF) en la Facultad de Contaduría y Administración UABC, ya que la institución necesita un procedimiento de implementación y el documento generado en este proyecto contiene las configuraciones necesarias para que el sistema funcione y pueda ser incorporado a otros entornos de operación, con la implementación de este sistema se optimiza el tiempo en la realización de las tareas y se unifican para su desempeño en una sola herramienta.

Para el desarrollo de este proyecto se analizaron dos metodologías para la implementación del sistema de gestión académica, estas metodologías son SCRUM

que permite la sincronización de actividades del desarrollo, a la vez, define los roles y el tiempo de los productos, y la metodología Métrica V3 que define el procedimiento para la implementación y las pruebas del sistema en el área de operación.

La ventaja de trabajar con SCRUM es que se puede trabajar en bloques temporales cortos y fijos, además, cada iteración proporciona un incremento de producto final. Aplicar esta metodología al proyecto se logra avanzar de forma progresiva, lo que le da flexibilidad a la forma de trabajar, además, se obtienen resultados completos, esto es, al final de cada iteración se crea una versión funcional del sistema que puede ser probada e implementada para observar su funcionamiento en el entorno de operación.

La aplicación de MÉTRICA versión 3 es ideal en proyectos de desarrollo de software ya que define la metodología y las técnicas necesarias para llevar a cabo cada una de las etapas de desarrollo, la implementación de SIGAF tiene como base esta metodología ya que a partir de ella se genera una serie de documentos que permiten darle seguimiento a este tipo de proyectos, desde la generación de un plan de implantación hasta la verificación del funcionamiento del sistema mediante la realización de las pruebas en el entorno de operación.

La arquitectura empleada para la implementación del sistema SIGAF y la incorporación de un servidor dedicado en la facultad para su funcionamiento, permite que el usuario pueda acceder con mayor rapidez sin limitación en cuando al área geográfica en donde se encuentre.

El sistema operativo utilizado CentOS, permite que el servidor funcione de forma ininterrumpida debido a su robustez, con esto se logra estabilidad y los usuarios finales no tendrán problemas relacionados con bloqueos inesperados o errores por fallas en el sistema operativo.

De acuerdo a los resultados se concluye que SIGAF funciona de forma adecuada en el servidor de la facultad. Aunque el sistema está conformado por módulos, unidades separadas de acuerdo a su funcionalidad, estas trabajan en forma conjunta como una sola unidad, ya que los datos introducidos en cualquiera de las interfaces son introducidos a la base de datos para su uso en otros procesos.

Respecto al módulo de autenticación, este requiere de la base de datos para permitir el acceso al usuario, sin embargo, los datos ingresados en la pantalla de autenticación no sólo son comparados con la base de datos, también, son encriptados mediante mcrypt, logrando tener un sistema funcional y seguro a la vez.

Los datos ingresados en el interfaz de unidad académica pueden ser accedidos por el de carga académica, a la vez, el módulo de generar horarios accede a todos los datos junto con los capturados en la pantalla de disponibilidad académica, con esto, la implementación del sistema en la primera fase finaliza de forma adecuada en el entorno de operación.

Con lo anterior, se obtuvo un producto con arquitectura escalable, robusta y funcional que nos ayuda incorporar sistemas como SIGAF, por último, este proyecto ha sido, para el autor, una gran experiencia en el área de desarrollo de software debido a que se obtuvo como resultado del proyecto, un producto que puede ser aplicado a diferentes áreas e incluso puede ser incorporado a otras facultades del campus.

### 5.3 Recomendaciones

Una vez concluido el proyecto, se considera importante continuar con la segunda fase del proyecto consistente en el desarrollo e implementación de otros módulos con el fin de complementar el sistema.

Es necesario que se realicen pruebas de integración sobre los módulos desarrollados con el fin de verificar la efectiva conexión con los módulos ya existente.

La realización de las pruebas abarca también las hechas sobre el servidor, es de vital importancia comprobar que las nuevas versiones del sistema sean funcionales en el entorno de operación, ya que hay configuraciones en el sistema de gestión que impiden que se ejecute correctamente en el servidor.

Se recomienda seguir las configuraciones descritas en este documento, sobre todo, en las configuraciones hechas en el fichero `.htaccess` del sistema y tener habilitado las extensiones necesarias en php.

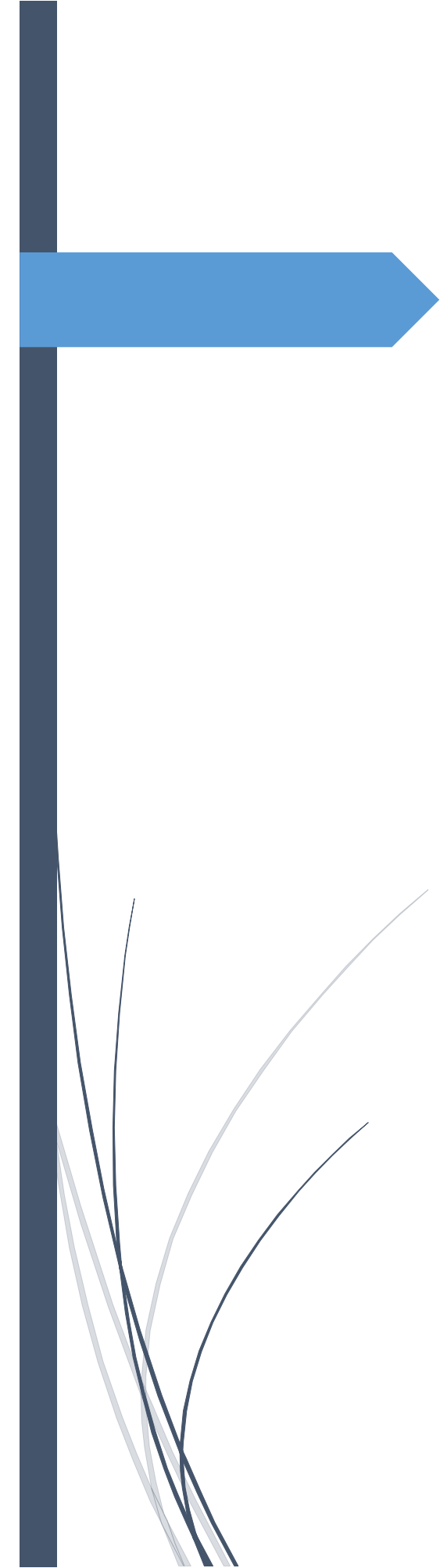
De acuerdo a la distribución de Linux utilizada, puede o no haber problemas en la ejecución del sistema, el origen es la carpeta vendor del framework utilizado para el desarrollo del sistema, para ello, hay que generar otra vez esta carpeta mediante el fichero `composer.json` que se encuentra en el mismo directorio del sistema.

Por último, para el desarrollo de la segunda fase del sistema o para futuros trabajos similares, en el apartado de acceso del sistema mediante conexión externa,

es importante tener en cuenta que la mayoría de puertos están bloqueados, tanto como para conexiones entrantes como de salida al servidor, por lo que el uso de la mayoría de aplicaciones necesarias para trabajar en la implementación no funcionarían, así que es necesario dirigirse al área correspondiente para solicitar el permiso y que estos puertos sean habilitados.

# *Anexos*

# *Anexo 1*



Sistema de Gestión Académica FCA  
UABC, campus Tijuana  
**Plan de Implantación**

Versión 1.0

Raul Fonseca  
UABC

### Historial de Revisión.

Fecha	Versión	Descripción	Autor
25/02/2014	1.0	Documento de Plan de Implantación	RAUL FONSECA

# TABLA DE CONTENIDOS

<b>1. Introducción</b>	117
1.1 Propósito	117
1.2 Alcance	117
1.3 Definiciones, acrónimos, y abreviaciones	119
1.4 Referencias	120
1.5 Vista global	121
<b>2. Planificación de la Entrega</b>	122
2.1 Responsabilidades	122
2.2 Cronograma	122
<b>3. Recursos</b>	124
3.1 Facilidades	124
3.2 Hardware	124
3.3 Unidad de implantación	125
3.3.1 Software de soporte	125
3.3.2 Documentación de soporte	125
3.3.3 Personal de soporte (personas, perfiles)	125
<b>4. Entrenamiento</b>	126

# PLAN DE IMPLANTACIÓN

## Introducción

### 1.1 Propósito

El presente documento describe el conjunto de tareas necesarias para instalar y probar el sistema de software desarrollado, para que así la transición a los usuarios se realice efectiva y eficientemente. Provee una lista detallada de eventos o cronograma de actividades y dependencias requeridas y formula la estrategia de implantación para el sistema basada en el alcance y limitaciones.

### 1.2 Alcance

Este documento se inicia en la fase de Elaboración y se concreta en la fase de Construcción, buscando definir la lista de tareas o agenda de implantación que incluye la preparación del ambiente para la transición y ejecución del sistema, el entrenamiento de los usuarios y detalles de los futuros procedimientos de soporte.

También se van a definir las dependencias y los recursos necesarios tanto de software como de hardware para poder instalar el sistema, es decir, los condicionantes de la implantación. En base a esto se determina la factibilidad del proceso de implantación (si se puede llevar a cabo).

### 1.3 Definiciones, Acrónimos, y Abreviaciones

*SIGAF:*

Sistema de Gestión Académica Facultad de Contaduría y Administración.

*FCA:*

Facultad de Contaduría y Administración.

*Caso de Uso:*

Formato que muestra una funcionalidad del sistema a través de su curso normal y cursos alternos.

*Diagrama de Implementación*

Diagrama para modelar la distribución del sistema

*Decanato de Extensión*

Es la unidad encargada de formular el plan general de actividades de extensión que desarrollará la Universidad, de acuerdo con las políticas aprobadas por el Consejo Directivo.

Además de promover, planificar, coordinar y evaluar las políticas universitarias en el área de extensión.

*Licencia:*

Un contrato mediante el cual una persona recibe de otra el derecho de uso de varios de sus bienes, normalmente de carácter no tangible o intelectual, a cambio del pago de un monto determinado por el uso de los mismos

*PHP:*

Acrónimo de PHP Hypertext Pre-processor, usado normalmente para la creación de contenido para sitios Web.

*MySQL:*

Manejador de base de datos de código libre.

*Requerimientos:*

Funcionalidades que debe tener el sistema desde el punto de vista del usuario, los cuales son expuestos más claramente en los casos de uso.

*Restricción:*

Limitaciones del sistema desde el punto de vista del usuario. Pueden ser restricciones de plataforma, costos, herramientas, entre otros.

*Stakeholders:*

Ente o individuo que afecta o puede ser afectado por el sistema

**1.4 Referencias**

- 1.0\_plan\_estudios\_requerimientos070214.doc
- 2.0\_carga\_academica\_requerimientos07022014.doc
- 3.0\_captura\_disponibilidad\_requerimientos.doc
- 4.0\_creacion\_horarios.doc

### **1.5 Vista global**

Se crea el equipo de implantación determinando los recursos humanos necesarios para la instalación del sistema de software desarrollado y para realizar las pruebas luego de implantado y la aceptación de los usuarios.

En este documento se van a presentar las descripciones de todos aquellos recursos necesarios para llevar a cabo la implantación del sistema SIGAF dentro de las instalaciones de la Facultad de Contaduría y Administración, UABC, Campus Tijuana.

También veremos cómo se van a planificar las actividades de instalación del sistema, donde cada uno de los involucrados del equipo de desarrollo del sistema tendrá un conjunto de responsabilidades durante la ejecución de ésta actividad.

## 2 Planificación de la entrega

### 2.1 Responsabilidades

Para poder llevar a cabo la implantación del sistema desarrollado, es necesario establecer un compromiso por parte de los clientes o stakeholders y el grupo de desarrolladores.

Las tareas o responsabilidades que deberá llevar a cabo el equipo de desarrolladores de SIGAF, se enfocan en la realización de pruebas del sistema (Casos de Prueba) en un servidor de la FCA, para asegurarnos de que el sistema puede ser llevado a otro ambiente de trabajo diferente a las computadoras personales del grupo de desarrolladores.

Cuando se hayan realizado estas pruebas el grupo de desarrolladores tiene el compromiso de instalar el sistema totalmente operativo en el Departamento de sistemas de la FCA.

### 2.2 Cronograma

La implantación del sistema en el Departamento de sistemas es responsabilidad de Raúl Fonseca. A continuación se propone un cronograma aproximado de las actividades.

- ✚ Primero que nada el equipo de desarrollo terminará todo lo relacionado con la documentación del sistema SIGAF.
- ✚ Posteriormente, se visualizará todos los recursos necesarios de software y los recursos de hardware mínimos necesarios para que el sistema pueda implantarse de manera correcta en el Departamento de Sistemas de la FCA. El tiempo estimado para ésta actividad sería de tres semanas.

- ✚ Una vez que se tienen los requisitos tanto de software como de hardware. Se notifica al encargado del Departamento de Sistemas (Prof. María del Consuelo Salgado), los espacios que se van a necesitar para la instalación del servidor, las máquinas y la red en general (en caso de que hagan falta). Esto se realiza en conjunto entre el encargado del Departamento de Sistemas y el equipo de desarrollo de SIGAF.
  
- ✚ Posteriormente de tener a la disposición el espacio geográfico necesario para toda la implantación del sistema se instalarán los equipos (si se requieren más de los que ya están) y a su vez el software dentro de ellos, y se realizarán una serie de pruebas para probar que el sistema está totalmente operativo en ese ambiente. Esta actividad tiene un tiempo estimado de dos semanas.
  
- ✚ Luego de tener el sistema totalmente operativo en el Departamento de Sistemas de la FCA se realizaría el proceso de entrenamiento del personal, es decir, los usuarios reales del sistema desarrollado. Se tiene un tiempo estimado de 2 semanas.
  
- ✚ Finalmente tenemos el soporte, donde el grupo de desarrolladores del sistema SIGAF pueden asistir a los usuarios del sistema en el uso correcto del mismo, y de corregir las fallas que se presenten durante el tiempo de vida de uso del sistema.

## **Recursos**

### **3.1 Facilidades**

La implantación cuenta con una serie de ventajas por parte del equipo de desarrollo.

Se tiene la facilidad de que el sistema fue implementado con el uso de software libre, lo que facilita la implantación y entendimiento del sistema SIGAF.

Otras de las facilidades con las que se va a contar a la hora de instalar el sistema, probarlo y darle el soporte, es que va a tener la documentación de todo el proceso de desarrollo del sistema, más un manual de usuario.

El coordinador de la MTIC tendrá la posibilidad de contactar al grupo de desarrolladores de SIGAF para que ante cualquier duda, se pueda resolver de la manera más rápida posible.

### **3.2 Hardware**

En cuanto a los requisitos de hardware necesarios para la implantación del sistema desarrollado, tenemos que se necesita una buena tecnología pero tampoco tecnología tan nueva y avanzada. Esto es debido a que el sistema va a ser altamente frecuentado, sin embargo, su nivel de stress no va a ser tan alto como el sistema de inscripción, por ejemplo.

La base de datos necesita cierta capacidad de disco duro ya que, aunque al comienzo no se tendrán muchos registros, luego de un tiempo el volumen que ocuparan los datos se incrementara.




La implantación de prueba que se hizo en el servidor de la FCA por parte del grupo de desarrolladores del sistema SIGAF, tiene aproximadamente lo mismo que se necesita en el ambiente real de uso. En otras palabras, es un servidor que tiene un Pentium 4 Xeon de 3.8 GHz, con 4 MB de memoria Caché, 8GB de memoria RAM y un disco duro de 140 GB.

### **3.3 Unidad de implantación**

#### **3.3.1 Software de Soporte**

El equipo de desarrolladores de SIGAF considera que para un funcionamiento correcto del sistema, es necesario que la se instale en el servidor que dispongan para el Departamento de Sistemas el software que se usó en el desarrollo del sistema.

El software que se necesita instalar es el siguiente:

-  El sistema operativo necesario es Linux CentOS.
-  El manejador de base de datos es MySQL (última versión).
-  El software que utilizamos para el desarrollo fue Laravel PHP.

Los usuarios del sistema, que va a ser el personal de la FCA y docentes, tendrán que contar con un navegador Web, para poder acceder al sistema de SIGAF.

#### **3.3.2 Documentación de Soporte**

Toda la documentación para el soporte del sistema se podrá encontrar en el manual del usuario, que va a ser el documento utilizado por los usuarios del sistema.

#### **3.3.4 Personal de Soporte**

El personal de soporte estará conformado por el grupo de desarrolladores del sistema SIGAF. Los encargados de realizar la implantación del sistema en el Departamento de Sistemas de la Facultad de Contaduría y Administración van ser las personas que van a tener el papel de dar soporte directamente durante las primeras semanas. En el sistema va a existir un perfil de usuario llamado Administrador, que va a tener la responsabilidad de asignar privilegios y de crear los respectivos usuarios del sistema.

## **4 Entrenamiento**

Realmente es importante resaltar que el entrenamiento del personal está en manos de los desarrolladores del sistema SIGAF.

Se tiene pensado que el entrenamiento de los usuarios se haga durante unos cinco días, y que posteriormente ellos por su parte sigan practicando y puedan así conocer el sistema.

En realidad el período de adaptación y conocimiento de los usuarios al sistema no debe ser largo, ya que varios de los que trabajan en la FCA han tenido contacto visual con el sistema y han estado en presencia de demostraciones funcionales del mismo cuando se han hecho las diferentes entregas de casos de uso.

Es importante que durante el entrenamiento y soporte de los usuarios del sistema, tanto el equipo de desarrolladores de SIGAF estén presentes para que así juntos puedan brindar mayor ayuda.

## *Anexo 2*

Fuente: Elaboración propia.

Sección: Plan de Estudio - Registro							
Caso de prueba	Cancelar registro.		ID: UC01ES01	Severidad	1a - Fatal	FECHA:	21-04-2014
			Dependencia de red: Yes				
Proposito de la prueba.							
Criterios de prueba	Test Step		Step Result	Test Steps			
	No.	Description	Description	Pass	Fail	N/A	Exención
	1	El usuario solicita Registrar Unidad de Aprendizaje.		X			
	2	El sistema solicita la información de la Unidad de Aprendizaje.		X			
	3	El usuario introduce la información de la Unidad de Aprendizaje.				X	
4	El usuario cancela la operación y este caso de uso termina.				X		
Overall Result				2	0	2	0
Resultados observados en caso de Fallo.							
Descripcion del impacto en el usuario final							
Comentarios							

Fuente: Elaboración propia.

Caso de prueba							
	Nuevo registro.		ID: UC01ES02	Severidad	1a - Fatal	FECHA:	21-04-2014
			Dependencia de red: Yes				
Proposito de la prueba.							
Test Criteria	Test Step		Step Result	Test Steps			
	No.	Description	Description	Pass	Fail	N/A	Exención
	1	El usuario solicita Registrar Unidad de Aprendizaje.		X			
	2	El sistema solicita la información de la Unidad de Aprendizaje.		X			
	3	El usuario introduce la información de la Unidad de Aprendizaje.				X	
4	4. El sistema almacena el nuevo registro.				X		
Overall Result				2	0	2	0
Resultados observados en caso de Fallo.							
End User Impact / Waiver description							
Comments							

Fuente: Elaboración propia.

Caso de prueba	Cancelar registro (iteración)	ID: UC01ES03	Severidad	1a - Fatal	FECHA: 21-04-2014		
	Dependencia de red: Yes						
Proposito de la prueba.							
Test Criteria	Test Step		Step Result	Test Steps			
	No.	Description	Description	Pass	Fail	N/A	Exención
	1	El usuario solicita Registrar Unidad de Aprendizaje.		X			
	2	El sistema solicita la información de la Unidad de Aprendizaje.		X			
	3	El usuario introduce la información de la Unidad de Aprendizaje.				X	
		El sistema muestra un Mensaje de error.				X	
		El sistema solicita la información de la Unidad de Aprendizaje.				X	
		El usuario introduce la información de la Unidad de Aprendizaje.				X	
	El usuario cancela la operación y este caso de uso termina.				X		
Resultado Global.				2	0	5	0
Resultados Observados en Caso de Fallo.							
End User Impact / Waiver description							
Comments							

Fuente: Elaboración propia.

Caso de prueba	Cancelar registro (iteración).	ID: UC01ES03	Severidad	1a - Fatal	FECHA: 21-04-2014	
		Dependencia de red: Yes				
<b>Proposito de la prueba.</b>						
<b>Test Criteria</b>	Test Step		Step Result		Test Steps	
	No.	Description	Pass	Fail	N/A	Exención
	1	El usuario solicita Registrar Unidad de Aprendizaje.	X			
	2	El sistema solicita la información de la Unidad de Aprendizaje.	X			
	3	El usuario introduce la información de la Unidad de Aprendizaje.			X	
		El sistema muestra un Mensaje de error.			X	
		El sistema solicita la información de la Unidad de Aprendizaje.			X	
		El usuario introduce la información de la Unidad de Aprendizaje.			X	
	El usuario cancela la operación y este caso de uso termina.			X		
<b>Resultado Global.</b>			<b>2</b>	<b>0</b>	<b>5</b>	<b>0</b>
<b>Resultados Observados en Caso de Fallo.</b>						
<b>End User Impact / Waiver description</b>						
<b>Comments</b>						

# Bibliografía

- Alomoto Cuvi, L. y Hernández Edwin, E. (2008). "Sistema de gestión para asignación de horarios en las aulas de la facultad de ingeniería en sistemas informáticos y de computación". Proyecto Previo a la Obtención del Título de Ingeniero en Sistemas Informáticos y Computación, Escuela Politécnica Nacional.
- Alcantud, Marin (1999). "Teleformación: Diseño Para Todos" Universitat de València.
- Booch, G. J. Rumbaugh y I. (1999) Jacobson, "El Lenguaje Unificado de Modelado". Addison Wesley.
- Glyn Moody, (2001). "Rebel Code: Linux and the Open Source Revolution (en inglés)".
- Guzmán Rodríguez, C., Acevedo Rueda, W. y Duque, Y., (2011). "Trabajo de monitoreo". Colombia.
- Jarrin Ortiz, M. y Valarezo Vargas E. (2010). "Desarrollo e implantación del sistema de gestión académica y administrativa vía Web para el colegio modelo politécnico". Proyecto previo de tesis.
- Laudon, K, J.P. Laudon, (2004). "Sistemas de información gerencial".
- MÉTRICA Versión 3, (2001). "Metodología de Planificación y Desarrollo de Sistemas de Información. Guías de Referencia, de Técnicas y del Usuario. Ministerio para las Administraciones Públicas". Editorial TECNOS, Madrid,
- Microsoft Official Academic Course, (2011). "Windows Server Administration Fundamentals". John Wiley & Sons.
- Ortiz Giménez, Luis F., (2009). "Mandriva 2009 Spring. Para Usuarios".
- Pressman, Roger S., (1997). "Ingeniería Del Software, Un Enfoque Práctico", Cuarta Edición. Ed. Mc. Graw Hill.
- Ron Patton, (2006). "Software Testing". (Segunda edición). Ed. Sams.

- Tanenbaum, A. (1992). "*Modern Operating Systems*, Englewood Cliffs: Prentice-Hall".
- Valdés Acosta, A. y Guardia Macías, M. (2007). "Sistema automatizado para la gestión de horarios docente". Trabajo de Diploma para Optar por el Título de Ingeniero en Ciencias Informáticas, Universidad de las Ciencias Informáticas, la Habana.
- Whitten, J, Bentley. L.D y K.C. Dittman, (2004). "Desarrollo de sistemas de información: una metodología basada en el modelado". UPC.

# Referencias Web

- [www.alegsa.com.ar/Dic/http.php](http://www.alegsa.com.ar/Dic/http.php). Sitio recuperado: 08 junio 2014. Leandro Alegsa.
- [www.centos.org/forums/viewtopic.php?t=33693](http://www.centos.org/forums/viewtopic.php?t=33693). Sitio recuperado: 15 marzo 2014. General Support CentOS.
- [es.ccm.net/contents/263-protocolo-ftp-protocolo-de-transferencia-de-archivos](http://es.ccm.net/contents/263-protocolo-ftp-protocolo-de-transferencia-de-archivos). Sitio recuperado: 05 junio 2014. ccm.
- [www.administracionelectronica.gob.es/pae/Home/pae/Documentacion/pae/Metodolog/pae/Metrica\\_v3.html#.VcLujCZ\\_NBc](http://www.administracionelectronica.gob.es/pae/Home/pae/Documentacion/pae/Metodolog/pae/Metrica_v3.html#.VcLujCZ_NBc). Sitio recuperado: 15 febrero 2014. Ministerio de Hacienda y Administraciones Públicas del Gobierno de España.
- [www.nsa.gov/research/selinux/faqs.shtml](http://www.nsa.gov/research/selinux/faqs.shtml). Sitio recuperado: 10 agosto 2014. National Security Agency.
- [http://www.dell.com/downloads/ap/products/servers/en/850\\_spec\\_sheet.pdf](http://www.dell.com/downloads/ap/products/servers/en/850_spec_sheet.pdf). Sitio recuperado: 30 julio 2014. Dell
- [www.proyectosagiles.org](http://www.proyectosagiles.org). Sitio recuperado: 10 febrero 2014. Xavier Albaladejo.