



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

**“DESARROLLO DE UNA APLICACIÓN WEB QUE
AUTOMATICICE EL PROCESO DE REGISTRO DE MATRÍCULAS
Y NOTAS EN LA ESCUELA FISCOMISIONAL MERCEDES
NAVARRETE UTILIZANDO LA TECNOLOGÍA SYMFONY”**

Trabajo de titulación presentado para optar al grado académico de:

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTOR: CARLOS HUMBERTO PATIÑO MORA

TUTOR: ING. EDUARDO VILLA

Macas - Ecuador

2016

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

El Tribunal de trabajo de Titulación certifica que: El trabajo de investigación: “DESARROLLO DE UNA APLICACIÓN WEB QUE AUTOMATICE EL PROCESO DE REGISTRO DE MATRÍCULAS Y NOTAS EN LA ESCUELA FISCOMISIONAL MERCEDES NAVARRETE UTILIZANDO LA TECNOLOGÍA SYMFONY”, de responsabilidad del señor Carlos Humberto Patiño Mora, ha sido minuciosamente revisado por los Miembros del Tribunal de trabajo de Titulación, quedando autorizada su presentación.

Ing. Washington Luna

**DECANO DE LA FACULTAD
DE INFORMÁTICA Y
ELECTRÓNICA**

Ing. Patricio Moreno

**DIRECTOR DE LA ESCUELA
DE INGENIERÍA EN
SISTEMAS**

Ing. Eduardo Villa

**DIRECTOR DEL TRABAJO
DE TITULACIÓN**

Ing. Raúl Rosero

**MIEMBRO DEL
TRIBUNAL**

Yo, Carlos Humberto Patiño Mora, soy responsable de las ideas, doctrinas y resultados expuestos en este trabajo de titulación y el patrimonio intelectual de la Tesis de Grado pertenece a la Escuela Superior Politécnica de Chimborazo.

Carlos Humberto Patiño Mora

DEDICATORIA

Este trabajo de titulación va dedicado a mis queridos padres Humberto Patiño y Julia Mora, a mi esposa Daysi y a mi hija Valeria que ha sido la mayor inspiración para seguir adelante día a día y a toda mi familia por todo el apoyo que me han brindado durante mi carrera profesional y por ser los pilares fundamentales de todas las metas alcanzadas en mi vida.

Carlos P

AGRADECIMIENTO

En primer lugar, agradezco a Dios por llenar mi vida de dicha y bendiciones

A mis queridos padres, familiares y amigos que han estado a mi lado en todo momento, brindándome su apoyo incondicional y por ser la base para poder alcanzar todas mis metas.

A mi universidad ESPOCH por darme la oportunidad de obtener una formación académica y a mis Profesores por todo el conocimiento que me han brindado durante mi formación profesional.

Carlos P

TABLA DE CONTENIDO

CONTENIDO	Paginas
RESUMEN	xv
SUMMARY	xvi
INTRODUCCIÓN	17
CAPÍTULO I.....	18
1. MARCO REFERENCIAL.....	18
1.1 Planteamiento del problema	18
1.1.1 Antecedentes	18
1.1.2 Formulación del problema	19
1.1.3 Sistematización del problema	19
1.2 Justificación	20
1.2.1 Justificación Teórica	20
1.2.2 Justificación Aplicativa	21
1.3 Objetivos	21
1.3.1 Objetivo general.....	21
1.3.2 Objetivos específicos	21
1.4 Métodos y Técnicas	22
1.4.1 Métodos.....	22
1.4.2. Técnicas	22
CAPITULO II	23
2. MARCO TEÓRICO.....	23
2.1 Herramientas utilizadas.....	23
2.2 Estudio del Framework Symfony	25
2.2.1 Características generales	26
2.2.2 Comunicación con la base de datos	26
2.2.3 Patrón de arquitectura Symfony	26
2.2.4 Estructura de un proyecto Symfony	27

2.2.5	<i>Ventajas y Desventajas de symfony</i>	29
CAPÍTULO III		31
3.	DESARROLLO DE SISTEMA “REGISTRO DE MATRÍCULAS Y NOTAS” ..	31
3.1	Método de Craig Larman	31
3.2	Aplicación del Método de Craig Larman para el desarrollo del sistema.	34
3.2.1	<i>Fase de planificación y especificación de requisitos</i>	34
3.2.1.1	<i>Ámbito</i>	34
3.2.1.2	<i>Referencias</i>	34
3.2.1.3	<i>Descripción General</i>	34
3.2.1.4	<i>Requerimientos Funcionales</i>	36
3.2.1.5	<i>Requisitos de comportamiento</i>	43
3.2.1.6	<i>Requerimientos de Interfaces Externas</i>	43
3.2.1.7	<i>Limitaciones de Diseño</i>	44
3.2.1.8	<i>Requerimientos No Funcionales</i>	45
3.2.1.9	<i>Estimación de Costos</i>	45
3.2.1.10	<i>Factibilidad</i>	46
3.2.1.11	<i>Definición de los Casos de Uso</i>	50
3.3	Craig Larman fase de construcción	58
3.3.1	<i>Análisis</i>	58
3.3.1.1	<i>Definición de Casos de Uso en Formato Expandido</i>	58
3.3.1.2	<i>Diagramas de Secuencia del Sistema</i>	66
3.3.1.3.	<i>Diagramas de Estado</i>	72
3.3.1.4.	<i>Modelo Conceptual</i>	74
3.3.1.5.	<i>Diagrama de Actividades</i>	74
3.3.2	<i>Diseño</i>	80
3.3.2.1	<i>Definición de la Arquitectura</i>	80
3.3.2.2	<i>Diagramas de Colaboración</i>	81
3.3.2.3	<i>Diagrama de Clases de Diseño</i>	86
3.3.2.5	<i>Diagramas de Despliegue</i>	88

3.3.3	<i>Implementación</i>	89
3.3.4	<i>Pruebas</i>	93
	Presentación de resultados	95
	CONCLUSIONES	100
	RECOMENDACIONES	101
	BIBLIOGRAFÍA	
	ANEXOS	

ÍNDICE DE TABLAS

Tabla 1-3: Requerimiento Funcional 1	36
Tabla 2-3: Requerimiento Funcional 2	37
Tabla 3-3: Requerimiento Funcional 3	38
Tabla 4-3: Requerimiento Funcional 4	38
Tabla 5-3: Requerimiento Funcional 5	39
Tabla 6-3: Requerimiento Funcional 6	40
Tabla 7-3: Requerimiento Funcional 7	41
Tabla 8-3: Requerimiento Funcional 8	41
Tabla 9-3: Requerimiento Funcional 9	42
Tabla 10-3: Costo Hardware	46
Tabla 11-3: Factibilidad Técnica - Hardware existente	47
Tabla 12-3: Factibilidad Técnica - Hardware requerido	47
Tabla 13 -3: Factibilidad Técnica - Software existente	47
Tabla 14-3: Factibilidad Técnica – Software requerido.....	47
Tabla 15-3: Factibilidad Técnica – Recurso humanos requeridos	48
Tabla 16-3: Factibilidad Económica	49
Tabla 17-3: Caso de uso - Gestión de cuentas de usuario.....	50
Tabla 18-3: Caso de uso - Gestión de periodos académicos	51
Tabla 19-3: Caso de uso - Gestión de cursos	52
Tabla 20-3: Caso de uso - Gestión de asignaturas	53
Tabla 21-3: Caso de uso - Asignar asignaturas	54
Tabla 22-3: Caso de uso - Registro de notas	55
Tabla 23-3: Caso de uso - Registro de matrículas	56
Tabla 24-3: Caso de uso - Emisión de reportes	57
Tabla 25-3: Caso de uso expandido - Gestión de cuentas de usuario	59
Tabla 26-3: Caso de uso expandido - Gestión de periodos académicos	60

Tabla 27-3: Caso de uso expandido - Gestión de cursos	61
Tabla 28-3: Caso de uso expandido - Gestión de asignaturas.....	62
Tabla 29-3: Caso de uso expandido - Asignar de asignaturas	63
Tabla 30-3: Caso de uso expandido - Registrar Notas.....	64
Tabla 31-3: Caso de uso expandido - Registro de matrículas.....	65
Tabla 32-3: Caso de uso expandido - Emisión de reportes.....	66
Tabla 33-3: Datos docentes.....	96
Tabla 34-3: Datos administrativos	96
Tabla 35-3: Presentación resultados - Tiempos en procesos	97
Tabla 36-3: Presentación de resultados – Suministros utilizados	99

ÍNDICE DE ILUSTRACIONES

Figura 1-2: Patrón de arquitectura symfony	27
Figura 1-3: Caso de uso - Gestión de cuentas de usuario	51
Figura 2-3: Caso de uso - Gestión de periodo académico	52
Figura 3-3: Caso de uso - Gestión de cursos	53
Figura 4-3: Caso de uso - Gestión de asignaturas	54
Figura 5-3: Caso de uso - Asignación de asignaturas.....	55
Figura 6-3: Caso de uso - Registro de notas.....	56
Figura 7-3: Caso de uso - Registro de matrículas.....	57
Figura 8-3: Caso de uso - Emisión de Reportes	58
Figura 9-3: Diagrama de secuencia - Autenticación de usuarios.....	67
Figura 10-3: Diagrama de secuencia - Gestión de cuentas de usuarios	68
Figura 11-3: Diagrama de secuencia - Gestión de periodos académicos.....	68
Figura 12-3: Diagrama de secuencia - Gestión de cursos.....	69
Figura 13-3: Diagrama de secuencia - Gestión de asignaturas	69
Figura 14-3: Diagrama de secuencia - Asignar asignaturas.....	70
Figura 15-3: Diagrama de secuencia - Registro de notas	70
Figura 16-3: Diagrama de secuencia - Registro de matrículas	71
Figura 17-3: Diagrama de secuencia - Emisión de reportes	71
Figura 18-3: Diagrama de estado - Gestión de usuarios.....	72
Figura 19-3: Diagrama de estado - Gestión de periodos académicos	73
Figura 20-3: Diagrama de estado - Gestión de asignaturas	73
Figura 21-3: Modelo conceptual de la base de datos.....	74
Figura 22-3: Diagrama de actividades - Inicio de sesión.....	75
Figura 23-3: Diagrama de actividades - Gestión de cuentas de usuario	76
Figura 24-3: Diagrama de actividades - Gestión de periodos académicos	76
Figura 25-3: Diagrama de actividades - Gestión de cursos	77

Figura 26-3: Diagrama de actividades - Gestión de asignaturas	77
Figura 27-3: Diagrama de actividades - Asignar Asignaturas	78
Figura 28-3: Diagrama de actividades - Registrar Notas.....	78
Figura 29-3: Diagrama de actividades - Registrar matrículas	79
Figura 30-3: Diagrama de actividades - Emisión de reportes.....	79
Figura 31-3: Arquitectura del sistema.....	81
Figura 32-3: Diagrama de colaboración - Autenticación de usuarios.....	82
Figura 33-3: Diagrama de colaboración - Gestión de cuentas de usuarios	82
Figura 34-3: Diagrama de colaboración - Gestión de periodos académicos.....	83
Figura 35-3: Diagrama de colaboración - Gestión de cursos.....	83
Figura 36-3: Diagrama de colaboración - Gestión de asignaturas	84
Figura 37-3: Diagrama de colaboración - Asignar asignaturas	84
Figura 38-3: Diagrama de colaboración - Registro de notas	85
Figura 39-3: Diagrama de colaboración - Registro de matrículas	85
Figura 40-3: Diagrama de colaboración - Emisión de reportes	86
Figura 41-3: Diagrama de clase de diseño	86
Figura 42-3: Diagrama de componentes	87
Figura 43-3: Diagrama de despliegue	88
Figura 44-3: Implementación - Cpanel de usuario	89
Figura 45-3: Implementación - Creación de base de datos.....	89
Figura 46-3: Implementación - Creación de usuario de bd	90
Figura 47-3: Implementación - Pantalla phpMyAdmin.....	90
Figura 48-3: Implementación - Restauración de base de datos	91
Figura 49-3: Implementación - Administrador de archivos.....	91
Figura 50-3: Implementación - Directorios del administrador de archivos	92
Figura 51-3: Implementación - Carga de archivos	92
Figura 52-3: Presentación resultados - Tiempo en matrículas.....	97
Figura 53-3: Presentación resultados - Tiempo en registro de notas	98

Figura 54-3: Presentación resultados - Tiempo en generación de reportes.....	98
Figura 55-3: Interfaz administrador - Pantalla principal.....	104
Figura 56-3: Interfaz administrador - Pantalla de asignación de permisos.....	105
Figura 57-3: Interfaz administrador - Pantalla de gestión de periodos académicos.....	105
Figura 58-3: Interfaz administrador - Pantalla de gestión de asignaturas.....	106
Figura 59-3: Interfaz administrador - Pantalla de gestión de paralelos.....	106
Figura 60-3: Interfaz administrador - Pantalla de gestión de niveles.....	107
Figura 61-3: Interfaz administrador - Pantalla de gestión de cursos.....	107
Figura 62-3: Interfaz administrador - Pantalla de gestión de asignaturas disponibles.....	108
Figura 63-3: Interfaz administrador - Pantalla de asignación de asignaturas.....	108
Figura 64-3: Interfaz administrador - Pantalla de registro de estudiantes.....	109
Figura 65-3: Interfaz administrador - Pantalla de registro de matrículas.....	109
Figura 66-3: Interfaz administrador - Pantalla de registro de notas.....	110
Figura 67-3: Interfaz administrador - Pantalla de gestión de usuarios.....	110
Figura 68-3: Interfaz administrador - Pantalla de emisión de reportes.....	111
Figura 69-3: Interfaz secretaria - Pantalla de registro de estudiantes.....	111
Figura 70-3: Interfaz secretaria - Pantalla de registro de matrículas.....	112
Figura 71-3: Interfaz secretaria - Pantalla de emisión de reportes.....	112
Figura 72-3: Interfaz docente- Pantalla de registro de notas.....	113
Figura 73-3: Interfaz docente - Pantalla de emisión de reportes.....	113

ÍNDICE DE ANEXOS

ANEXO A: Interfaces de usuario

ANEXO B: Evidencia de capacitación a usuarios.

ANEXO C: Encuestas realizadas.

ANEXO D: Certificado de capacitación a los usuarios.

ANEXO E: Certificado de desarrollo e implementación del sistema.

ANEXO F: Manual de usuario.

RESUMEN

El presente trabajo de titulación tuvo como objetivo el desarrollo de un sistema web de Registro de Matrículas y Notas (SIMANOT), el desarrollo fue debido al estudio y el análisis de los procesos de registro de matrículas y notas llevado a cabo manualmente en la Escuela Fiscomisional Mercedes Navarrete de la ciudad de Sucúa provincia de Morona Santiago, la aplicación tuvo como finalidad automatizar y agilizar el trabajo realizado por el personal administrativo y docente que labora en la institución. Para el desarrollo de la aplicación web propuesta se decidió utilizar el Framework Symfony, que simplifica y facilita el desarrollo de las aplicaciones, ya que automatiza muchos de los patrones utilizados para resolver las tareas comunes, además posee varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. También se planteó la utilización del método de Craig Larman el cual permite una buena gestión para el desarrollo de software. El sistema SIMANOT contiene los siguientes módulos: Registro de Matrículas, Registro de Notas, Registro de docente y Emisión de reportes que fueron automatizados con el desarrollo de la aplicación. Como resultado se obtuvo una disminución aproximada de 30 minutos en los tiempos de registro de información y sobre todo en la emisión de reportes. En conclusión se pudo mejorar el servicio que brinda la institución educativa a la sociedad al contar con un sistema automatizado, por lo que se recomienda el correcto uso del mismo para evitar la salida de información incorrecta.

Palabras Claves: <TECNOLOGÍAS DE LA INGENIERÍA> <INGENIERÍA DE SOFTWARE> < SISTEMA DE MATRÍCULAS Y NOTAS (SIMANOT)> <FRAMEWORK SYMFONY> <DESARROLLO DE SOFTWARE> <APLICACIÓN WEB>

SUMMARY

The present investigation research aimed the development of a web registration system of enrollment registration and grades (SIMANOT), the development was due to the study and analysis of the processes of registration and grades carried out manually at Escuela Fiscomisional Mercedes Navarrete of Sucua city Morona Santiago province, the application aimed to apply and facilitate work procedures of the administrative and teaching staff of the institution. For the development of the web application was decided to use the Framework Symfony, which simplifies and facilitates the development of applications by automating many of the patterns used to solve common tasks, also it has several tools and classes aimed to reduce the time development of a complex web application. Also Craig Larman method was used which allows a good management for software development. The SIMANOT system contains the following modules: enrollment registration, grades registration, teacher registration and issuance of reports that were automated with the development of the application. As result it was obtained a decrease of approximately 30 minutes recording time information and above all in the issuance of report. In conclusion it could improve the service provided by the institution to society by having an automated system, so it is recommended the correct use of it to avoid wrong information output.

Keywords: <ENGINEERING TECHNOLOGIES> <SOFTWARE ENGINEERING>
<SYSTEM ENROLLMENT AND GRADES (SIMANOT)> <SYMFONY FRAMEWORK>
<SOFTWARE DEVELOPMENT> <WEB APPLICATION>

INTRODUCCIÓN

La educación es uno de los factores que más influye en el avance y progreso de personas y sociedades. Además de proveer conocimientos, la educación enriquece la cultura, el espíritu, los valores y todo aquello que nos caracteriza como seres humanos.

Desde sus inicios las Instituciones Educativas llevan a cabo sus procesos escolares de forma manual debido a la falta de sistemas informáticos que ayuden a automatizar el trabajo al personal y agilizar la forma de llevar el control de la información.

Hoy en día mediante el creciente desarrollo de los sistemas informáticos se ha podido facilitar el trabajo en las Instituciones, ya que permiten tener un control efectivo de las actividades que en ellas se realizan y tener un mayor acceso a la información de una forma rápida y flexible, lo cual facilita el trabajo del personal de una manera eficiente.

Una aplicación web es una herramienta totalmente funcional en internet, es decir que los datos o los archivos en los que se trabaja son procesados y almacenados dentro de la web, mediante el uso de las aplicaciones web los usuarios pueden llegar a acceder a ellas desde cualquier parte del mundo y así poder realizar todas sus tareas.

Es por ello que este proyecto tiene como objetivo desarrollar e implementar una aplicación web, que permita optimizar el proceso de matrículas y registro de notas en la escuela fiscomisional Mercedes Navarrete, el sistema se encargará de guardar la información referente al registro de matrículas y notas obtenidas por los estudiantes para almacenarlos en la aplicación, optimizando así los tiempos en la generación de reportes.

Para el desarrollo del proyecto se utilizará la herramienta Symfony porque es un completo framework diseñado para optimizar el desarrollo de sistemas web ya proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja.

También se plantea la utilización de un método Craig Larman que es un método de desarrollo evolutivo, incremental y dirigido por casos de uso que permite desarrollar de manera ágil un sistema software.

CAPÍTULO I

1. MARCO REFERENCIAL

1.1 Planteamiento del problema

1.1.1 Antecedentes

La educación es uno de los factores que más influye en el avance y progreso de personas y sociedades. Además de proveer conocimientos, la educación enriquece la cultura, el espíritu, los valores y todo aquello que nos caracteriza como seres humanos.

Desde sus inicios las Instituciones Educativas llevan a cabo sus procesos escolares de forma manual debido a la falta de sistemas informáticos que ayuden a automatizar el trabajo al personal y agilizar la forma de llevar el control de la información.

Hoy en día mediante el creciente desarrollo de los sistemas informáticos se ha podido facilitar el trabajo en las Instituciones, ya que permiten tener un control efectivo de las actividades que en ellas se realizan y tener un mayor acceso a la información de una forma rápida y flexible, lo cual facilita el trabajo del personal de una manera eficiente.

Por su parte la escuela Fiscomisional Mercedes Navarrete al momento lleva el registro de matrículas y notas de forma manual mediante el llenado de hojas de registro que son almacenadas en las carpetas de cada estudiante lo que ocasiona tiempos elevados en acceder a los datos y sobre todo en la emisión de reportes.

Para el desarrollo de software existen varios frameworks que nos permiten agilizar y optimizar el proceso en el desarrollo de sistemas, por lo cual se decidió utilizar el framework Symfony que proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja y documentada con el método de Craig Larman.

De ahí la importancia de desarrollar un sistema web para la automatización de procesos escolares, que satisfaga las necesidades del cliente ayudando así a obtener un mejor control de datos y acceso a la información.

1.1.2 Formulación del problema

En la escuela de educación básica fiscomisional Mercedes Navarrete el proceso de registro de matrículas y notas de los estudiantes se los realiza de forma manual mediante el llenado de documentos físicos con los datos de los estudiantes. Todos estos documentos son almacenados en carpetas en los estantes por lo que no se puede tener un acceso rápido al registro de los datos, también se presentan un tiempo de espera demasiado alto al momento de realizar la emisión de un determinado reporte debido a que la persona encargada de dicho proceso debe buscar cada uno de los datos requeridos en la carpeta del estudiante.

Para determinar los principales problemas que se presentan al realizar el registro manual de matrículas y notas de los estudiantes se realizó una entrevista con el personal de la institución en la cual se identificaron los siguientes inconvenientes:

- Inconsistencia de datos.
- Pérdida de información.
- Desorganización.
- Lentitud en los procesos de matrícula.
- Información vulnerable en las notas de los estudiantes.
- Consumo de recursos innecesarios.

Con los inconvenientes mencionados anteriormente se vio la necesidad de desarrollar un sistema que automatice los procesos de registro de matrículas y notas de los estudiantes para mantener la información almacenada correctamente y acceder a los datos de una manera más rápida y eficiente.

1.1.3 Sistematización del problema

¿Cuáles son las causas que provocan la demora en la consulta de datos de los estudiantes?

¿Cuál es el proceso actual al momento de matricular y registrar las notas de los estudiantes?

¿Existe un sistema web que automatice el proceso de matriculación y registro de notas de los estudiantes?

1.2 Justificación

1.2.1 Justificación Teórica

En la actualidad el creciente desarrollo de sistemas web que se han generado a través de la informática y la tecnología están causando un gran impacto social en las instituciones educativas, es por ello que muchas de estas, se han adaptado a los cambios y optan por el desarrollo de dichos sistemas para mejorar la calidad de sus servicios.

El manejo de un sistema de automatización de procesos escolares es importante para los planes de cualquier unidad educativa, actualmente representan el medio eficaz para agilizar los procedimientos que en ella se desarrollan, generan mayor productividad en las instituciones que disponen de los beneficios de este tipo de sistemas.

Al momento de desarrollar un sistema es necesario utilizar las herramientas adecuadas que nos permitan facilitar y agilizar el desarrollo de los sistemas dependiendo del grado de complejidad que este represente.

Dentro del desarrollo de software existen varios frameworks que nos permiten agilizar y optimizar el proceso en el desarrollo, por lo cual se decidió utilizar el framework Symfony que es un completo framework diseñado para optimizar, gracias a sus características en el desarrollo de las aplicaciones web que separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web, además proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. (Potencier, Zaninotto, 2007, http://librosweb.es/libro/symfony_1_4/capitulo_1/symfony_en_pocas_palabras.html)

También se plantea la utilización de un método ágil para la gestión de proyectos de desarrollo de software como lo es Craig Larman, que es considerado un método de desarrollo evolutivo, incremental y dirigido por casos de uso que permite desarrollar completamente un sistema software partiendo de un prototipo funcional inicial cuyas funcionalidades se van extendiendo hasta culminar con el desarrollo del sistema. (Larman. C, 2004, http://productpatterns.wiki.sel.inf.uc3m.es/mediawiki/index.php/Metodo_de_Craig_Larman)

1.2.2 Justificación Aplicativa

Se desarrollará un sistema web para el registro de matrículas y notas en la escuela de educación básica fiscomisional Mercedes Navarrete, para automatizar el registro de datos de los estudiantes de la institución educativa.

Con la aplicación web, existirá un control de registro de los datos de los estudiantes tanto en el proceso de matrícula y registro de notas, además de contar con un módulo de emisión de reportes que disminuirá los tiempos en la búsqueda de datos de los estudiantes que actualmente se los realiza de forma manual.

A toda esta información se la puede acceder desde cualquier computador ya que se trata de una aplicación web, por lo que no es necesario estar en la institución para ingresar al sistema.

Es así como se mejorará el proceso de registro de matrículas y notas, desarrollando una aplicación de fácil uso y que permitirá llevar un mejor control en los procesos escolares.

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar una Aplicación Web que automatice el proceso “Registro de matrículas y notas” en la escuela de educación básica Fiscomisional Mercedes Navarrete mediante el uso de la tecnología Symfony.

1.3.2. Objetivos específicos

- Analizar la forma en la que se llevan los procesos referentes a “Registro de Matrículas y Notas” en la Escuela de Educación Básica Fiscomisional Mercedes Navarrete.
- Estudiar la herramienta Symfony para optimizar el proceso de Desarrollo de Software.
- Desarrollar los módulos de Administración, Matrículas, Registro, Docentes y Consultas de la aplicación web.
- Implementar la aplicación web para automatizar los procesos en la Escuela de educación básica Fiscomisional Mercedes Navarrete.

1.4 Métodos y Técnicas

1.4.1 Métodos

El método a utilizar para llevar a cabo esta investigación es el método científico, el cual contempla los siguientes puntos:

- Planteamiento del problema.
- Ideas a defender.
- Análisis e interpretación de resultados.

1.4.2. Técnicas

Se obtendrá información mediante las siguientes técnicas.

- Entrevistas y encuestas a usuarios que harán uso de la aplicación.

CAPITULO II

2. MARCO TEÓRICO

El presente capítulo se refiere al estudio del framework Symfony de manera general, identificando cuáles son sus características, funciones y las ventajas que brinda en el desarrollo de aplicaciones.

El desarrollo de la información se realiza para adquirir el conocimiento introductorio al framework Symfony, con el fin de obtener una base para futuras aplicaciones.

2.1 Herramientas utilizadas

Para el desarrollo del sistema web se utilizaron las siguientes herramientas:

- Framework Symfony
- MySQL
- JavaScript
- PHP
- Netbeans
- StarUML

Framework Symfony

Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web. (Potencier, 2007, http://librosweb.es/libro/symfony_1_4/capitulo_1/symfony_en_pocas_palabras.html)

MySQL

MySQL es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL).

MySQL se ejecuta en prácticamente todas las plataformas, incluyendo Linux, UNIX y Windows. A pesar de que se puede utilizar en una amplia gama de aplicaciones, MySQL se asocia más con las aplicaciones basadas en la web y la publicación en línea y es un componente importante de una pila empresarial de código abierto llamado LAMP. LAMP es una plataforma de desarrollo web que utiliza Linux como sistema operativo, Apache como servidor web, MySQL como sistema de gestión de base de datos relacional y PHP como lenguaje de programación orientado a objetos (a veces, Perl o Python se utiliza en lugar de PHP). (Searchdatacenter, <http://searchdatacenter.techtarget.com/es/definicion/MySQL>)

JavaScript

JavaScript es un lenguaje de programación interpretado y basado en objetos que permite crear documentos Web interactivos de forma sencilla. Una de las ventajas más importantes que ofrece es la disponibilidad de ejecutar determinadas acciones en la máquina del cliente con la consecuente descarga de trabajo del servidor Web. Este lenguaje tiene además la capacidad de capturar evento o sucesos generados por el cliente, de forma que ante una determinada acción del usuario se pueda ejecutar un trabajo en la máquina del cliente, como puede ser la validación de datos de un formulario. (Delgado, Tejedor, Gutiérrez y Bobadilla, 2001, p.39)

PHP

PHP (Hiptertext Presprocessor) [Pre-procesador Hipertexto], es uno de los lenguajes del lado servidor más extendidos en la web. Permite embeber tus pequeños fragmentos de código dentro de las paginas HTML y realizar determinadas acciones de una forma fácil y eficaz sin tener que generar programas en un lenguaje distinto al HTML. Ofrece varias funciones para la explotación de bases de datos de una manera sencilla. (Sabana, 2006, pp 17-18)

Características generales:

- Lenguaje exclusivo para el diseño de aplicaciones Web.
- Simplicidad en su sintaxis porque está basado en C y es fácil de comprender.
- Script de código abierto para servidores.
- Utiliza una amplia gama de servidores Web, como Apache, IIS, etc.

- Conexión a diversas bases de datos, como PostgreSQL, MySQL, Oracle, Sybase, Firebird, etc.
- Posee protocolos HTTP, POP3, SNMP, LDAP, IMAP.
- No requiere grandes cantidades de hardware

NetBeans

NetBeans IDE es un entorno de desarrollo integrado (IDE), modular, de base estándar (normalizado), escrito en el lenguaje de programación Java. El proyecto NetBeans consiste en un IDE de código abierto y una plataforma de aplicación, las cuales pueden ser usadas como una estructura de soporte general (framework) para compilar cualquier tipo de aplicación. (Oracle, https://netbeans.org/community/releases/61/index_es.html)

StarUML

StarUML es una herramienta para el modelamiento de software basado en los estándares UML (Unified Modeling Language) y MDA (Model Driven Architecture), para visualizar, especificar y documentar un sistema.

2.2 Estudio del Framework Symfony

Symfony es un completo framework diseñado para optimizar, gracias a sus características en el desarrollo de sistemas web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web. (Potencier, 2007, http://librosweb.es/libro/symfony_1_4/capitulo_1/symfony_en_pocas_palabras.html)

Para el análisis del framework se tomaran en cuenta los siguientes aspectos:

- Cuáles son sus características generales
- Como es la forma de comunicación con la base de datos (ORM).
- Cuál es el patrón de arquitectura de Symfony
- Como es la estructura de un proyecto Symfony
- Cuales con sus ventajas y desventajas

2.2.1 Características generales

Symfony se diseñó para que se ajustara a los siguientes requisitos:

- Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de que funciona correctamente en los sistemas Windows y *nix estándares)
- Independiente del sistema gestor de bases de datos
- Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos
- Basado en la premisa de "convenir en vez de configurar", en la que el desarrollador solo debe configurar aquello que no es convencional
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web
- Preparado para aplicaciones empresariales y adaptable a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo
- Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros.
(Potencier, 2007, http://librosweb.es/libro/symfony_1_4/capitulo_1/symfony_en_pocas_palabras.html)

2.2.2 Comunicación con la base de datos

Las bases de datos siguen una estructura relacional, mientras que PHP y Symfony son orientados a objetos. Por este motivo, para acceder a la base de datos como si fuera orientada a objetos, es necesaria una interfaz que traduzca la lógica de los objetos a la lógica relacional. Esta interfaz se denomina "mapeo de objetos a bases de datos" (ORM), y está formada por objetos que permiten acceder a los datos y que contienen en sí mismos el código necesario para hacerlo. La ORM que utiliza Symfony para el accesos a la base de datos es la interfaz Doctrine.

La principal ventaja que aporta el ORM es la reutilización, permitiendo llamar a los métodos de un objeto de datos desde varias partes de la aplicación e incluso desde diferentes aplicaciones.
(Potencier, 2007, http://librosweb.es/libro/symfony_1_4/capitulo_1/conceptos_basicos.html)

2.2.3 Patrón de arquitectura Symfony

Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC (Modelo Vista Controlar), que está formado por tres niveles:

- El Modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La Vista transforma el modelo en una página web que permite al usuario interactuar con ella.
- El Controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

La arquitectura MVC (ver figura 1-2) separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. Si por ejemplo una misma aplicación debe ejecutarse tanto en un navegador estándar como un navegador de un dispositivo móvil, solamente es necesario crear una vista nueva para cada dispositivo; manteniendo el controlador y el modelo original. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, email, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación. (Potencier, 2007, http://librosweb.es/libro/symfony_1_4/capitulo_2/el_patron_mvc.html)

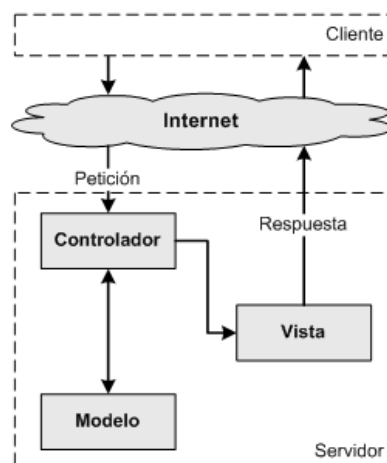


Figura 1-2: Patrón de arquitectura Symfony

Fuente: (Potencier, 2007)

2.2.4 Estructura de un proyecto Symfony

En esta sección, se determina dónde encontrar y colocar diferentes tipos de archivos de Symfony para obtener una comprensión completa del sistema.

Todas las aplicaciones Symfony tienen la misma estructura de directorios sencilla y recomendada, los directorios se detallan a continuación.

- `app/`: contiene la configuración de la aplicación.
- `src/`: aquí se encuentra todo el código PHP de la aplicación.
- `vendor/`: por convención aquí se guardan todas las librerías creadas por terceros.
- `web/`: este es el directorio web raíz y contiene todos los archivos que se pueden acceder públicamente.

El directorio de la aplicación (app)

Esta clase se debe implementar dos métodos que definen todo lo que Symfony necesita saber acerca de la aplicación. Ni siquiera se tiene que preocupar de estos métodos durante el arranque, Symfony los rellena con parámetros predeterminados.

- `registerBundles ()`: devuelve un array con todos los bundles necesarios para ejecutar la aplicación.
- `registerContainerConfiguration ()`: carga el archivo de configuración de recursos de la aplicación (consulta la sección [Configurando la aplicación](#)).

Durante el desarrollo de una aplicación, normalmente el directorio `app/` solo se utiliza para modificar la configuración y los archivos de enrutamiento en el directorio `app/config/`.

El directorio fuente (src)

El directorio `src` contiene todo el código real (código PHP, plantillas, archivos de configuración, estilos, etc.) que pertenece a la aplicación. De hecho, al programar una aplicación Symfony, la mayor parte del trabajo se llevará a cabo dentro de uno o más bundles creados en este directorio.

El directorio web

El directorio web raíz es el lugar donde se encuentran todos los archivos públicos y estáticos tales como imágenes, hojas de estilo y archivos JavaScript. También es el lugar donde se definen todos los controladores frontales, como por ejemplo el siguiente:

```
// web/app.php
require_once __DIR__.'../app/bootstrap.php.cache';
require_once __DIR__.'../app/AppKernel.php';

use Symfony\Component\HttpFoundation\Request;
```

```
$kernel = new AppKernel('prod', false);  
$kernel->loadClassCache();  
$kernel->handle(Request::createFromGlobals())->send();
```

El archivo del controlador frontal (app.php en este ejemplo) es el archivo PHP que realmente se ejecuta cuando utilizas una aplicación Symfony y su trabajo consiste en arrancar la aplicación utilizando una clase del núcleo (AppKernel). (Potencier, 2007, https://librosweb.es/libro/symfony_2_x/capitulo_4/la_estructura_de_directorios.html)

2.2.5 Ventajas y Desventajas de Symfony

Ventajas

Entre las múltiples ventajas técnicas de Symfony podríamos citar:

- Todo el código está orientado a objetos y completamente en PHP5.
- Implementación del patrón de diseño Modelo-Vista-Controlador (MVC) para una estructura clara y flexible.
- Abstracción de bases de datos vía Mapeo Relacional de Objetos (ORM): las tablas de la base de datos están disponibles como objetos en el código. La capa ORM está basada en Propel o Doctrine.
- Generación automática y configurable de “secciones de administración”.
- Integración de las librerías javascript más populares (jQuery, Prototype, Scriptaculous, YUI, entre otras), las cuales incluyen de serie funciones AJAX listas para usar en nuestra aplicación (autocompletados, drag&drop...).
- Documentación de gran calidad, así como una amplia (y activa) comunidad de desarrolladores.
- Symfony genera código orientado a objetos para las funcionalidades más comunes del manejo de bases de datos.
- Genera interfaces CRUD (Crear Leer Actualizar Eliminar) para las tablas de la base de datos.
- Permite trabajar en distintos ambientes; producción, desarrollo (en el que se activa una barra de herramientas para depuración), test, pero también es posible crear uno propio. (Tahira, 2008, p.1)

Desventajas

- Se necesita por lo menos un VPS (Servidor Virtual Privado) para poder contar con todas las características avanzadas y publicar las aplicaciones en la web.
- Otro problema de Symfony es el caché. Gran parte de la velocidad de Symfony se debe a un uso extensivo del caché, por lo que se debe limpiar el caché de vez en cuando.
- Los procesos utilizan demasiada memoria.

Luego del análisis del framework Symfony podemos concluir que las ventajas que ofrece este framework facilitan el desarrollo del sistema propuesto.

CAPÍTULO III

3. DESARROLLO DEL SISTEMA DE “REGISTRO DE MATRÍCULAS Y NOTAS”

En el presente capítulo se detalla el método de Craig Larman utilizado para el desarrollo del sistema web SIMANOT, utilizando la arquitectura del framework Symfony.

3.1. Método de Craig Larman

Craig Larman es considerado como un método y no una metodología, pues es una versión reducida del modelo de procesos descrito en Rational Unified Process (RUP). Es un método de desarrollo evolutivo, incremental y dirigido por casos de uso que permite desarrollar completamente un sistema software partiendo de un prototipo funcional inicial cuyas funcionalidades se van extendiendo hasta culminar con el desarrollo del sistema. (Larman. C, 2004, http://productpatterns.wiki.sel.inf.uc3m.es/mediawiki/index.php/Metodo_de_Craig_Larman)

El método de Craig Larman propone las siguientes fases:

Fase de planificación y especificación de requisitos

- Definir los requisitos
- Definir los Casos de Uso
- Definir la Arquitectura del Sistema

Fase de construcción

- Análisis
- Diseño
- Implementación
- Pruebas

Fase instalación

A continuación se realiza una breve descripción de cada uno de las fases con las que cuenta el método.

Fase de planificación y especificación de requisitos

La planificación y especificación de requisitos del sistema tiene como finalidad la recopilación de la información necesaria para analizar y entender todos los requerimientos que el cliente desea.

La fase de planificación y especificación de requisitos contiene las siguientes subfases:

- **Ámbito:** Se Identifica el producto de software a ser producido por su nombre y una descripción de lo que hará el software.
- **Referencias:** Se hace referencia a la especificación de las fuentes de donde se extrae la información.
- **Descripción General:** Describe los factores generales que afectan al producto y a sus requisitos
- **Requerimientos Funcionales:** Son los requerimientos propios con los que cuenta el sistema.
- **Requisitos de comportamiento:** Se definen como trabaja el sistema y el cumplimiento a limitaciones.
- **Requerimientos de Interfaces Externas:** Se definen todas las interfaces con las que contara el sistema.
- **Limitaciones de Diseño:** Descripción general de cualquier elemento que limitará el correcto funcionamiento del sistema.
- **Requerimientos No Funcionales:** Son requerimientos que no difieren directamente en las funciones del sistema.
- **Estimación de Costos:** Es la definición de todos los costos que conllevan a la realización del sistema.
- **Factibilidad:** Se define el estudio del proyecto para verificar si es viable realizarlo o no.
- **Definición de los Casos de Uso:** Se definen los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso.

Fase de construcción

Análisis

En esta etapa se aborda el problema viendo al sistema a construir como una caja negra, centrándose en la visión que desde el exterior tienen los actores, esto es, en los casos de uso.

Esta etapa contiene los siguientes aspectos:

Definición de Casos de Uso en Formato Expandido: Un caso de uso expandido describe un proceso de negocios más a fondo que el de alto nivel, durante la fase del análisis conviene escribir en este formato solo los casos más importantes

Diagramas de Secuencia del Sistema: Se utilizan para comenzar a modelar el comportamiento del sistema tal y como los actores lo necesitan, se parte de los casos de uso.

Diagramas de Estado: Este tipo de diagrama nos permite caracterizar un cambio en el sistema, se refiere a los objetos que lo conforman cambiaron su estado como respuesta a los sucesos y al tiempo.

Modelo Conceptual: Este modelo permite elaborar un esquema conceptual de la base de datos a partir de la lista de requerimientos que previamente fueron obtenidos.

Diagrama de Actividades: Este tipo de diagrama permite modelar el comportamiento dinámico de un procedimiento, transacción o caso de uso, haciendo énfasis en el proceso que se lleva a cabo.

Diseño

En esta etapa el software y que hardware se utilice es fundamental, ya que se define la arquitectura del sistema, las relaciones entre si y el ambiente en el que funcionará.

Definición de la Arquitectura: Es una representación porque la arquitectura es usada para transportar información abstracta sobre el sistema.

Diagramas de Colaboración: Este tipo de diagrama tiene como objetivo representar las interacciones que existen entre los objetos.

Diagrama de Clases de Diseño: Son diagramas estáticos por que muestran las clases del sistema, junto a sus métodos y atributos.

Diagrama de Componentes: Este tipo de diagrama representará cómo el sistema a desarrollarse, está dividido en componentes y mostrará las dependencias que existen entre estos

Diagramas de Despliegue: Este tipo de diagrama muestra las relaciones físicas de los distintos nodos que componen al sistema a desarrollarse.

Implementación: En esta etapa se pone en producción el sistema desarrollado.

Pruebas: En esta etapa se pone a prueba el sistema para verificar fallas o errores y verificar que cumple adecuadamente todos los requerimientos.

3.2 Aplicación del Método de Craig Larman para el desarrollo del sistema.

3.2.1 Fase de planificación y especificación de requisitos

En esta fase se definen los requisitos tradicionales ampliados con un borrador de modelo conceptual y con una definición de Casos de Uso de Alto Nivel. Las definiciones de los términos utilizados en esta fase están conforme a las definiciones proporcionadas en la IEEE STD 830.

3.2.1.1 Ámbito

A continuación se define el nombre del sistema y una descripción de lo que realiza.

Nombre del producto: SIMANOT

Significado: Sistema de registro de matrículas y notas.

El Sistema SIMANOT automatiza el proceso que se realiza manualmente en la unidad educativa al momento de llevar el registro de los procesos de matrícula y registro de notas de los estudiantes de los estudiantes, el software registra datos de matrícula, año lectivo, asignaturas, notas y emite reportes.

3.2.1.2 Referencias

Información recabada por parte del Desarrollador a los diferentes departamentos (Dirección, Sala de Docentes, Secretaria) y estatutos de la Institución.

3.2.1.3 Descripción General

Se describen los factores generales del sistema y sus principales funciones.

- **Perspectiva del Producto**

El sistema SIMANOT realiza las tareas de una manera más ágil y rápida además de tiene una interfaz intuitiva, facilidad de uso y posee diferentes módulos que facilitan la administración del sistema.

- **Funciones del Sistema SIMANOT**

A continuación se detallan las funciones del sistema.

- ✓ **Autenticación de Usuarios**

Para ingresar al sistema los usuarios deberán iniciar sesión, ingresando su Nick y Contraseña.

- ✓ **Gestión de Cuentas de Usuario**

El Sistema permite crear, modificar y eliminar cuentas de usuario (Administrador, Docente, Secretaria) del Sistema

- ✓ **Gestión de Periodos Académicos**

Al inicio del año escolar se debe ingresar la fecha de inicio y fin correspondiente al periódico académico.

- ✓ **Gestión de Cursos**

El Sistema permite Crear, Modificar y Eliminar los diferentes cursos con los que cuenta la unidad educativa.

- ✓ **Gestión de Asignaturas**

El Sistema permite Crear, Modificar y Eliminar las diferentes asignaturas impartidas en cada curso

- ✓ **Asignar Asignaturas**

El sistema permite asignar las diferentes asignaturas correspondientes a cada docente de la institución para proceder al registro de notas de las mismas.

- ✓ **Registrar Notas**

El sistema permitirá a cada docente registrar las notas de sus estudiantes en cada una de las materias asignadas.

- ✓ **Registrar Matrículas**

Al inicio de cada periodo académico el sistema registra los datos correspondientes a cada estudiante para luego proceder a su matrícula.

✓ **Emisión de Reportes**

El sistema emite reportes correspondientes a notas, cursos, asignaturas, estudiantes solicitados por los usuarios

3.2.1.4 Requerimientos Funcionales

En esta sección se describen todos los requerimientos obtenidos por los usuarios y que el sistema debe cumplir.

Requerimiento Funcional 1

Tabla 1-3: Requerimiento Funcional 1

Número de requisito	RF1
Nombre de requisito	Autenticación de usuarios
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

• **Introducción**

Para acceder al sistema los usuarios deben ingresar sus credenciales para el inicio de sesión

• **Entradas**

- ✓ Nick
- ✓ Contraseña

• **Proceso**

Validación de los datos ingresados.

• **Salidas**

Mensaje de confirmación de inicio de sesión.

Requerimiento Funcional 2

Tabla 2-3: Requerimiento Funcional 2

Número de requisito	RF2
Nombre de requisito	Gestión de Cuentas de Usuario
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

Introducción

El Sistema permite crear, modificar y eliminar cuentas de usuario (Administrador, Docente, Secretaria) del Sistema

- **Entradas**

- ✓ Nombres
- ✓ Apellidos
- ✓ Cedula
- ✓ Nick
- ✓ Contraseña
- ✓ Rol

- **Proceso**

El sistema verifica los datos ingresados y los almacenara en la base de datos.

- **Salidas**

Mensaje de error en caso de datos mal ingresados

Mensaje de confirmación de guardado de datos.

Requerimiento Funcional 3

Tabla 3-3: Requerimiento Funcional 3

Número de requisito	RF3
Nombre de requisito	Gestión de Periodos Académicos
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

- **Introducción**

El Sistema permite crear, modificar y eliminar periodos académicos, los periodos académicos deberán ser ingresados al inicio del año escolar.

- **Entradas**

- ✓ Descripción
- ✓ Fecha Inicio
- ✓ Fecha Fin

- **Proceso**

El sistema verifica los datos ingresados y los almacenara en la base de datos.

- **Salidas**

Mensaje de error en caso de datos mal ingresados

Mensaje de confirmación de guardado de datos.

Requerimiento Funcional 4

Tabla 4-3: Requerimiento Funcional 4

Número de requisito	RF4
Nombre de requisito	Gestión de Cursos
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

- **Introducción**

El Sistema permite crear, modificar y eliminar los diferentes cursos con los que cuenta la unidad educativa.

- **Entradas**

- ✓ Nivel
- ✓ Paralelo
- ✓ Descripción

- **Proceso**

El sistema verifica los datos ingresados y los almacenara en la base de datos.

- **Salidas**

Mensaje de error en caso de datos mal ingresados

Mensaje de confirmación de guardado de datos.

Requerimiento Funcional 5

Tabla 5-3: Requerimiento Funcional 5

Número de requisito	RF5
Nombre de requisito	Gestión de Asignaturas
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

- **Introducción**

El Sistema permite Crear, Modificar y Eliminar las diferentes asignaturas impartidas en cada curso. Las asignaturas pueden ser de 2 tipos: optativas u obligatorias

- **Entradas**

- ✓ Asignatura
- ✓ Tipo
- ✓ Descripción

- **Proceso**

El sistema verifica los datos ingresados y los almacenara en la base de datos.

- **Salidas**

Mensaje de error en caso de datos mal ingresados

Mensaje de confirmación de guardado de datos.

Requerimiento Funcional 6

Tabla 6-3: Requerimiento Funcional 6

Número de requisito	RF6
Nombre de requisito	Asignar Asignaturas
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

- **Introducción**

El sistema permite asignar las diferentes asignaturas correspondientes a cada docente de la institución para proceder al registro de notas de las mismas.

- **Entradas**

✓ Periodo Académico

✓ Docente

✓ Asignatura

✓ Curso

✓ Observación

- **Proceso**

El sistema verifica los datos ingresados y los almacenara en la base de datos.

- **Salidas**

Mensaje de error en caso de datos mal ingresados

Mensaje de confirmación de guardado de datos.

Requerimiento Funcional 7

Tabla 7-3: Requerimiento Funcional 7

Número de requisito	RF7
Nombre de requisito	Registrar Notas
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

- **Introducción**

El sistema permite a cada docente registrar las notas de sus estudiantes en cada una de las materias asignadas.

- **Entradas**

- ✓ Notas parciales
- ✓ Exámenes
- ✓ Días de asistencia

- **Proceso**

El sistema verifica los datos ingresados y los almacenara en la base de datos.

- **Salidas**

Mensaje de error en caso de datos mal ingresados

Mensaje de confirmación de guardado de datos.

Requerimiento Funcional 8

Tabla 8-3: Requerimiento Funcional 8

Número de requisito	RF8
Nombre de requisito	Registrar Matrículas
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

- **Introducción**

Al inicio de cada periodo académico el sistema registra los datos correspondientes a cada estudiante para luego proceder a su matrícula.

- **Entradas**

- ✓ Datos del padre
- ✓ Datos de la Madre
- ✓ Datos del Representante
- ✓ Datos del Alumno
- ✓ Periodo Académico
- ✓ Curso
- ✓ Estudiante
- ✓ Observaciones

- **Proceso**

El sistema verifica los datos ingresados y los almacenara en la base de datos.

- **Salidas**

Mensaje de error en caso de datos mal ingresados

Mensaje de confirmación de guardado de datos.

Requerimiento Funcional 9

Tabla 9-3: Requerimiento Funcional 9

Número de requisito	RF9
Nombre de requisito	Emisión de Reportes
Tipo	Requisito
Fuente del requisito	Primera entrevista con los Clientes
Prioridad del requisito	Alta/Esencial

Realizado por: Carlos Patiño, 2016

- **Introducción**

El sistema emite reportes correspondientes a notas, cursos, docentes, asignaturas, estudiantes solicitados por los usuarios.

- **Entradas**

- ✓ Tipo de Reporte
- ✓ Datos del Reporte

- **Proceso**

Verifica los datos del reporte.

- **Salidas**

Mensaje de error en caso de datos mal ingresados

Emisión del reporte solicitado.

3.2.1.5 Requisitos de comportamiento

Definimos como se desenvuelve el sistema tomando en cuenta al entorno en el que trabaja.

- **Número de terminales a manejar**

El sistema no tiene un número determinado de terminales ya que es una aplicación web y los usuarios podrán acceder desde cualquier parte.

- **Número de usuarios simultáneos**

Al momento el sistema se encuentra trabajando con 24 usuarios (22 usuarios docentes y 2 administrativos), por lo que es una cantidad mínima de usuarios para el sistema.

- **Número de transacciones a realizar dentro de ciertos periodos de tiempo**

El sistema maneja mínimo 52 transacciones diarias, por parte de los docentes y personal administrativo que utilizarán el sistema diariamente.

3.2.1.6 Requerimientos de Interfaces Externas

A continuación se describe las interfaces de usuario, de hardware/software y de comunicación.

- **Interfaces de Usuario**

El sistema SIMANOT ofrece una interfaz de usuario intuitiva y amigable, y fácil de utilizar.

Las interfaces deben estar bien diseñadas para tener compatibilidad en los diferentes navegadores web que existen en la actualidad.

Las pantallas de interfaces de usuario se detallan en el Anexo A.

- **Interfaces de hardware**

- ✓ Monitor: el software deberá mostrar información al usuario a través de la pantalla del monitor.
- ✓ Ratón: El software debe interactuar con el movimiento del ratón y los botones del ratón, siempre y cuando se necesite presionar un botón o seleccionar las opciones de los menús.
- ✓ Teclado: El software deberá interactuar con las pulsaciones del teclado, cuando se necesite ingresar datos.
- ✓ Impresora: El sistema imprimirá los reportes en la impresora instalada.

- **Interfaces de Software**

Para el uso del sistema es necesario tener instalado un navegador web de preferencia que utilice toda la compatibilidad de tecnología como lo es el navegador Opera o Google Chrome.

- **Interfaces de Comunicación**

Para la comunicación entre el servidor de base de datos MySQL y la aplicación desarrollada con el framework Symfony se lo realizo mediante el ORM.

3.2.1.7 Limitaciones de Diseño

Se detallan las limitaciones que pueden afectar el desempeño del sistema en cuanto a hardware y software.

- **Limitaciones de Software**

El sistema se limita con respecto al uso de herramientas software no adecuadas para su funcionamiento, las limitaciones de software se detallan a continuación.

- ✓ PHP inferior a la versión 4
- ✓ MySQL versión 3 o anteriores
- ✓ Incompatibilidad con el framework Symfony

- **Limitaciones de Hardware**

Los recursos hardware que pueden afectar el desempeño del sistema se detallan a continuación:

- ✓ Disco duro menor a 2GB
- ✓ Memoria inferior de 512MB
- ✓ Procesador inferior a Pentium 4

3.2.1.8 Requerimientos No Funcionales

De describen los requerimientos que no son directamente asociados al sistema.

- **Disponibilidad**

El sistema está disponible en todo momento ya que se encuentra implementado en un servidor web, lo que permite al acceso desde cualquier lugar.

- **Seguridad**

Cada usuario autorizado (Administrador, Docente, Secretaria) por seguridad utiliza un nombre de usuario y una contraseña para poder acceder al sistema, además el sistema cuenta con un mecanismo de encriptación en las contraseñas para aumentar su seguridad.

También se realiza un backup de la base de datos cada un determinado periodo de tiempo para evitar pérdidas de información.

- **Mantenibilidad**

SIMANOT se caracteriza por ser un sistema mantenible y permitir flexibilidad al realizar cambios en sus funciones, además de contar con una documentación la cual ayuda a realizar mejoras o aumentar sus funciones.

3.2.1.9 Estimación de Costos

En esta sección se determinan los costos que conlleva la realización del sistema en cuanto a hardware y software

- **Costo Hardware**

Entre los costos de la aplicación se procedió a contratar el servicio de Web Hosting a la Empresa Edamhost. Además se tuvieron los siguientes costos en equipo informático.

Tabla 10-3: Costo Hardware

Tipo	Descripción	Total
Laptop	Toshiba Satellite E305	600,00
Impresora	Epson	60,00
Memoria USB	SanDisk 4 GB	8,00
Servidor Web	PHP, Symfony	52,00
Total		720,00

Realizado por: Carlos Patiño, 2016

- **Costo de Software**

En cuanto a costos de software se utilizaron programas de código abierto por lo que no se generaron gastos en pago de licencias.

3.2.1.10 Factibilidad

El estudio de factibilidad permite recopilar información relevante sobre el desarrollo de un proyecto y poder tomar decisiones efectivas acerca de si se procede o no con el desarrollo del proyecto.

A continuación se definen los diferentes tipos de factibilidad.

- **Factibilidad Técnica**

En esta factibilidad se toman en cuenta los recursos de hardware, software y personal técnico.

✓ **Hardware Existente**

Tabla 11-3: Factibilidad Técnica - Hardware existente

CANTIDAD	DESCRIPCIÓN
1	Laptop Toshiba Satellite E305
1	Flash Memory 4GB
1	Equipos acceso a internet

Realizado por: Carlos Patiño, 2016

✓ **Hardware Requerido**

Tabla 12-3: Factibilidad Técnica - Hardware requerido

CANTIDAD	DESCRIPCIÓN
1	Impresora

Realizado por: Carlos Patiño, 2016

✓ **Software Existente**

Tabla 13 -3: Factibilidad Técnica - Software existente

NOMBRE	DESCRIPCIÓN	ESTADO	OBSERVACIÓN
Windows 7 64 bits	Sistema operativo	Legal	Ninguna
Microsoft Office 2010	Suite de oficina	Legal	Ninguna
Netbeans 8	Plataforma de desarrollo	Legal	Licencia Libre
MySQL	Gestor de Base de Datos	Legal	Licencia Libre
PHP 5	Lenguaje de Programación	Legal	Licencia Libre

Realizado por: Carlos Patiño, 2016

✓ **Software Requerido**

Tabla 14-3: Factibilidad Técnica – Software requerido

NOMBRE	DESCRIPCIÓN	ESTADO	OBSERVACIÓN
Servidor Apache	Servidor Web HTTP	Legal	Licencia Libre
Symfony	Framework para PHP	Legal	Licencia Libre

Realizado por: Carlos Patiño, 2016

✓ **Recurso Humano Requerido**

Tabla 15-3: Factibilidad Técnica – Recurso humanos requeridos

FUNCIÓN	FORMACIÓN	EXPERIENCIA
Desarrollador	Estudios de tercer nivel.	Desarrollador de software
Administrador DB	Estudios de tercer nivel.	MySQL

Realizado por: Carlos Patiño, 2016

Debido a que se cuenta con los recursos hardware, software y recursos humanos necesarios se procedió al desarrollo del sistema

• **Factibilidad Operativa**

La factibilidad operativa nos permite determinar la probabilidad de que el sistema se use de forma correcta por parte de los usuarios.

✓ **Recursos humanos**

Para un mejor uso el sistema cuenta con interfaces lo más amigable posible, de tal forma que sin mucha dificultad el usuario pueda adaptarse y aprovechar al máximo las funciones que este brinde.

Además de brindar la capacitación correspondiente al personal de la institución (administrativo, docente) para el correcto uso del sistema que fue implementado en la institución educativa.

Con esto se asegura el correcto uso del sistema desarrollado.

• **Factibilidad Económica**

Se definen los costos de desarrollo, instalación y operación del sistema

Tabla 16-3: Factibilidad Económica

COSTOS	VALOR
Costos de Desarrollo	2.101,00
Costo de Personal	
Costo de Desarrollador	1.200,00
Costo de Administrado de Base de Datos	500,00
Costo de Hardware y Software	
Hardware: Existente en la Institución	00,00
Software: Herramientas libres	00,00
Costos Varios	
Suministros	50,00
Costo de Instalación	00,00
Web Hosting	51,00
Registro de dominio	50,00
Costo de Operación	00,00
Costos de personal de operación	00,00
Costo de Mantenimiento	00,00
Costo de suministros de operación	00,00
TOTAL	2.101,00

Realizado por: Carlos Patiño, 2016

La estimación de costos del personal ha sido evaluada debido al costo por hora en el desarrollo del sistema web, utilizando la herramienta PSP para obtener las horas de productividad, y el tiempo aproximado para el desarrollo del sistema se obtuvo en base a experiencias obtenidas en el desarrollo de sistemas similares.

Costo/Hora	Horas diarias	Total Días
2.50 \$	8	60

Calculo:

$$2.50 * 8 = 20 \text{ (día)}$$

$$20 * 60 = 1200 \$$$

Con el estudio de la factibilidad se determinó que existen todos recursos y herramientas necesarias por lo que se procedió al desarrollo del sistema.

3.2.1.11 Definición de los Casos de Uso

En esta actividad representamos los casos de uso a partir de la nomenclatura definida en UML, los aspectos que se tomamos en cuenta son los actores, casos de uso, interacciones y relaciones. Los casos de uso son definidos en base a los siguientes requerimientos

- **Requerimiento 1:** Gestión de Cuentas de Usuario
- **Requerimiento 2:** Gestión de Periodos Académicos
- **Requerimiento 3:** Gestión de Cursos
- **Requerimiento 4:** Gestión de Asignaturas
- **Requerimiento 5:** Asignar Asignaturas
- **Requerimiento 6:** Registrar Notas
- **Requerimiento 7:** Registrar Matrículas
- **Requerimiento 8:** Emisión de Reportes

Gestión de Cuentas de Usuario

Tabla 17-3: Caso de uso - Gestión de cuentas de usuario

Caso de Uso	Gestión de Cuentas de Usuario
Actores	Administrador
Tipo	Primario
Descripción	El usuario registrado como administrador es el encargado de gestionar (ingresar, modificar, eliminar) cuentas de usuario del sistema.

Realizado por: Carlos Patiño, 2016

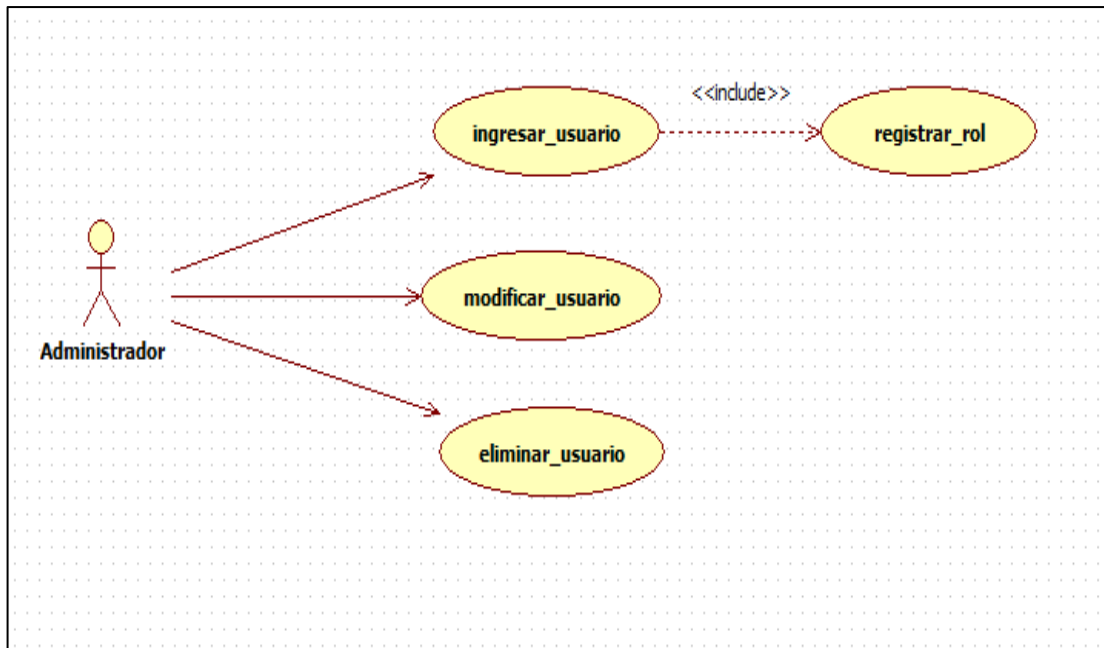


Figura 2-3: Caso de uso - Gestión de cuentas de usuario

Realizado por: Carlos Patiño, 2016

Gestión de Periodos Académicos

Tabla 18-3: Caso de uso - Gestión de periodos académicos

Caso de Uso	Gestión de Periodos Académicos
Actores	Administrador
Tipo	Primario
Descripción	El usuario registrado como administrador es el encargado de gestionar (ingresar, modificar, eliminar) los periodos académicos en el sistema.

Realizado por: Carlos Patiño, 2016

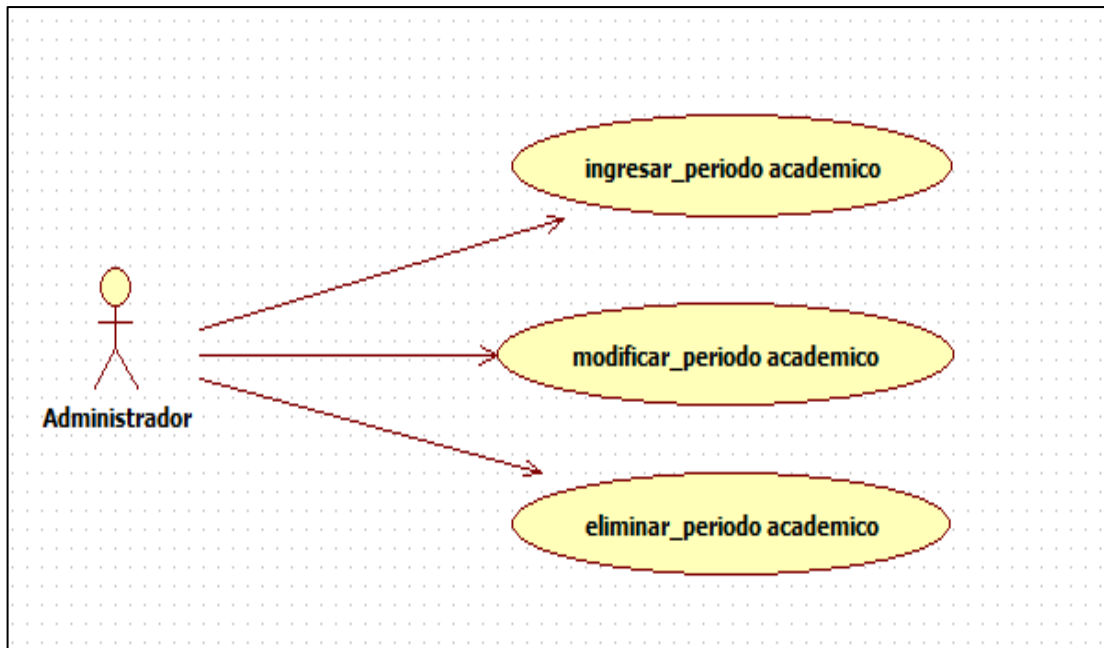


Figura 3-3: Caso de uso - Gestión de periodo académico

Realizado por: Carlos Patiño, 2016

Gestión de Cursos

Tabla 19-3: Caso de uso - Gestión de cursos

Caso de Uso	Gestión de Cursos
Actores	Administrador
Tipo	Primario
Descripción	El usuario registrado como administrador es el encargado de gestionar (ingresar, modificar, eliminar) los cursos en el sistema.

Realizado por: Carlos Patiño, 2016

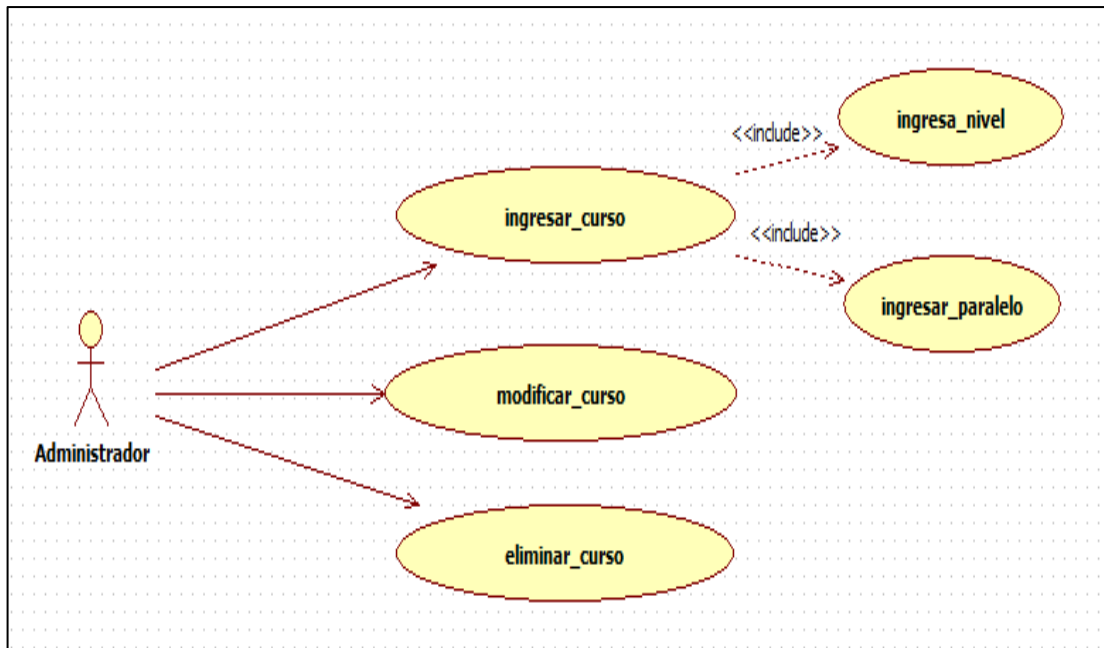


Figura 4-3: Caso de uso - Gestión de cursos

Realizado por: Carlos Patiño, 2016

Gestión de Asignaturas

Tabla 20-3: Caso de uso - Gestión de asignaturas

Caso de Uso	Gestión de Asignaturas
Actores	Administrador
Tipo	Primario
Descripción	El usuario registrado como administrador es el encargado de gestionar (ingresar, modificar, eliminar) las asignaturas correspondientes al nuevo periodo académico.

Realizado por: Carlos Patiño, 2016

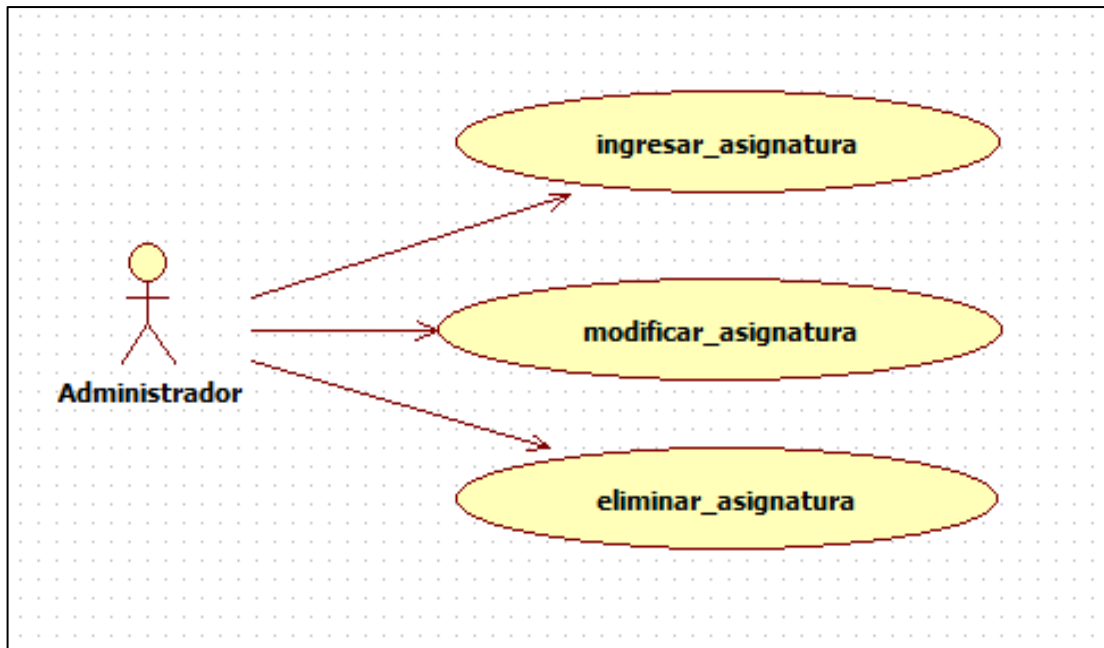


Figura 5-3: Caso de uso - Gestión de asignaturas

Realizado por: Carlos Patiño, 2016

Asignar Asignaturas

Tabla 21-3: Caso de uso - Asignar asignaturas

Caso de Uso	Asignar Asignaturas
Actores	Administrador
Tipo	Primario
Descripción	El usuario registrado como administrador es el encargado de asignar las asignaturas a cada docente de la institución educativa.

Realizado por: Carlos Patiño, 2016

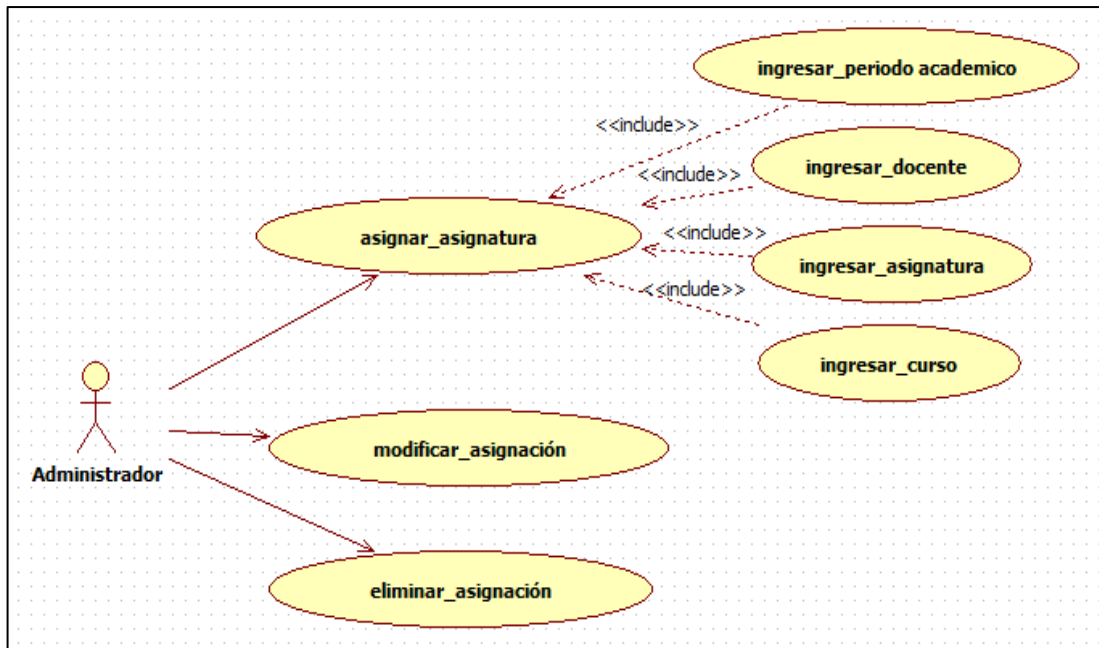


Figura 6-3: Caso de uso - Asignación de asignaturas

Realizado por: Carlos Patiño, 2016

Registrar Notas

Tabla 22-3: Caso de uso - Registro de notas

Caso de Uso	Registrar Notas
Actores	Administrador/Docente
Tipo	Primario
Descripción	El usuario registrado como administrador / docente son los encargados de registrar las notas de los estudiantes en las asignaturas correspondientes.

Realizado por: Carlos Patiño, 2016

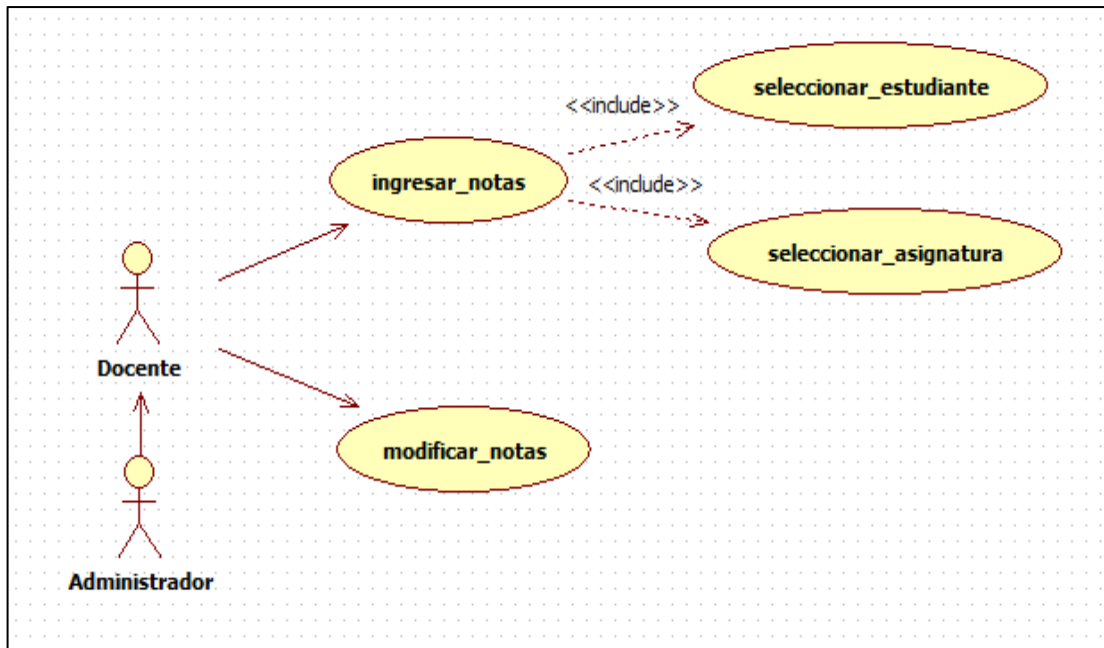


Figura 7-3: Caso de uso - Registro de notas

Realizado por: Carlos Patiño, 2016

Registrar Matrículas

Tabla 23-3: Caso de uso - Registro de matrículas

Caso de Uso	Registrar Matrículas
Actores	Administrador/Secretaria
Tipo	Primario
Descripción	El usuario registrado como administrador / secretaria son los encargados de registrar las matrículas de los estudiantes en el nuevo periodo académico.

Realizado por: Carlos Patiño, 2016

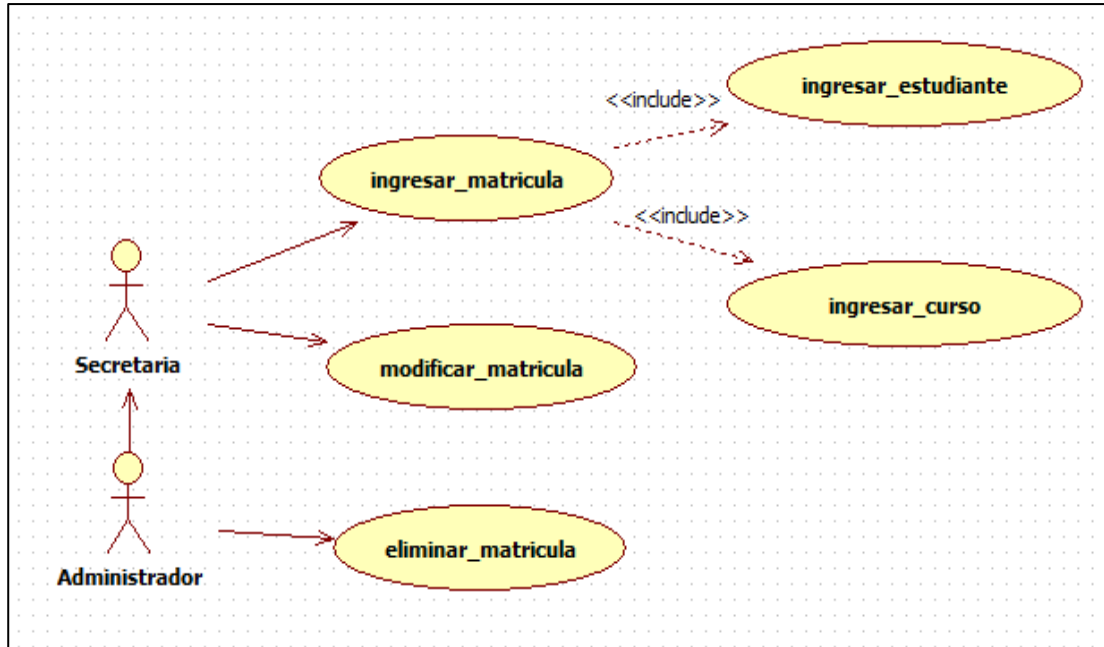


Figura 8-3: Caso de uso - Registro de matrículas

Realizado por: Carlos Patiño, 2016

Emisión de Reportes

Tabla 24-3: Caso de uso - Emisión de reportes

Caso de Uso	Emisión de Reportes
Actores	Administrador/Secretaria/Docente
Tipo	Primario
Descripción	Todos los usuarios registrados en el sistema pueden solicitar reportes según su rol en el sistema.

Realizado por: Carlos Patiño, 2016

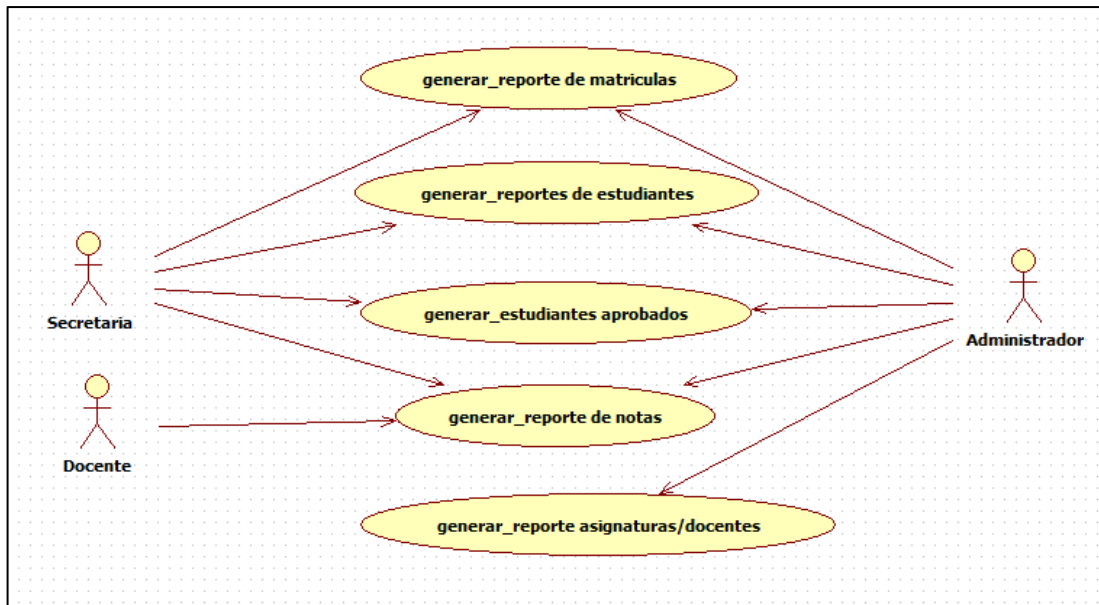


Figura 9-3: Caso de uso - Emisión de Reportes

Realizado por: Carlos Patiño, 2016

3.3 Craig Larman fase de construcción

3.3.1 Análisis

En esta etapa realizamos una visión externas de los actores con los requerimientos del sistema, representándolos en casos de uso.

3.3.1.1 Definición de Casos de Uso en Formato Expandido

Para la creación de los caso de uso expandido seleccionamos los casos de uso más importantes para tener una visión más detallada de los requerimientos. Los casos de uso expandidos son detallados en base a los siguientes requerimientos

- **Requerimiento 1:** Gestión de Cuentas de Usuario
- **Requerimiento 2:** Gestión de Periodos Académicos
- **Requerimiento 3:** Gestión de Cursos
- **Requerimiento 4:** Gestión de Asignaturas
- **Requerimiento 5:** Asignar Asignaturas
- **Requerimiento 6:** Registrar Notas
- **Requerimiento 7:** Registrar Matrículas
- **Requerimiento 8:** Emisión de Reportes

Gestión de cuentas de usuario

Tabla 25-3: Caso de uso expandido - Gestión de cuentas de usuario

Identificador caso de uso	CU01
Nombre del caso de uso	Gestión de Cuentas de Usuario
Actores	Administrador
Propósito	Gestión de las cuentas de usuarios que accederán al sistema
Visión General	Los usuarios registrados con el rol de administrador son los encargados de gestionar (ingresar, modificar, eliminar) las nuevas cuentas de usuarios que accederán al sistema.
Tipo	Primario
Referencia	
Curso típico de eventos	
Acciones del Usuario	Respuesta del Sistema
El usuario ingresa al sistema	Pide autenticación (usuario y contraseña)
Introduce su nombre y contraseña	Valida datos ingresados y proporciona acceso al sistema
Selecciona la opción requerida para la gestión de usuarios.	Muestra la pantalla de acuerdo a la opción seleccionada.
Ingresa todos los datos requeridos por el sistema.	Valida los datos y los almacena en la base de datos
Cursos Alternativos	
Si el usuario y contraseña son incorrectos. Se emite un mensaje de error de los datos de autenticación.	
Si los datos ingresados son incorrectos o están en blanco. Se emite un mensaje de alerta de acuerdo al dato para ser corregido.	

Realizado por: Carlos Patiño, 2016

Gestión de periodos académicos

Tabla 26-3: Caso de uso expandido - Gestión de periodos académicos

Identificador caso de uso	CU02
Nombre del caso de uso	Gestión de Periodos Académicos
Actores	Administrador
Propósito	Gestionar los datos ingresados parar el nuevo periodo académico
Visión General	Los usuarios registrados con el rol de administrador son los encargados de gestionar (ingresar, modificar, eliminar) los datos del nuevo periodo académico.
Tipo	Primario
Referencia	CU01
Curso típico de eventos	
Acciones del Usuario	Respuesta del Sistema
El usuario ingresa al sistema	Pide autenticación (usuario y contraseña)
Introduce su nombre y contraseña	Valida datos ingresados y proporciona acceso al sistema
Selecciona la opción requerida para la gestión de periodos académicos.	Muestra la pantalla de acuerdo a la opción seleccionada.
Ingresa todos los datos requeridos por el sistema.	Valida los datos y los almacena en la base de datos
Cursos Alternativos	
Si el usuario y contraseña son incorrectos. Se emite un mensaje de error de los datos de autenticación.	
Si los datos ingresados son incorrectos o están en blanco. Se emite un mensaje de alerta de acuerdo al dato para ser corregido.	

Realizado por: Carlos Patiño, 2016

Gestión de cursos

Tabla 27-3: Caso de uso expandido - Gestión de cursos

Identificador caso de uso	CU03
Nombre del caso de uso	Gestión de Cursos
Actores	Administrador
Propósito	Gestionar los cursos en el sistema
Visión General	Los usuarios registrados con el rol de administrador son los encargados de gestionar (ingresar, modificar, eliminar) los cursos en el sistema.
Tipo	Primario
Referencia	CU01
Curso típico de eventos	
Acciones del Usuario	Respuesta del Sistema
El usuario ingresa al sistema	Pide autenticación (usuario y contraseña)
Introduce su nombre y contraseña	Valida datos ingresados y proporciona acceso al sistema
Selecciona la opción requerida para la gestión de grados.	Muestra la pantalla de acuerdo a la opción seleccionada.
Ingresa todos los datos requeridos por el sistema.	Valida los datos y los almacena en la base de datos
Cursos Alternativos	
Si el usuario y contraseña son incorrectos. Se emite un mensaje de error de los datos de autenticación.	
Si los datos ingresados son incorrectos o están en blanco. Se emite un mensaje de alerta de acuerdo al dato para ser corregido.	

Realizado por: Carlos Patiño, 2016

Gestión de asignaturas

Tabla 28-3: Caso de uso expandido - Gestión de asignaturas

Identificador caso de uso	CU04
Nombre del caso de uso	Gestión de Asignaturas
Actores	Administrador
Propósito	Gestionar las asignaturas que estarán disponibles en el periodo académico.
Visión General	Los usuarios registrados con el rol de administrador son los encargados de gestionar (ingresar, modificar, eliminar) las asignaturas disponibles en el periodo académico.
Tipo	Primario
Referencia	CU01
Curso típico de eventos	
Acciones del Usuario	Respuesta del Sistema
El usuario ingresa al sistema	Pide autenticación (usuario y contraseña)
Introduce su nombre y contraseña	Valida datos ingresados y proporciona acceso al sistema
Selecciona la opción requerida para la gestión de asignaturas.	Muestra la pantalla de acuerdo a la opción seleccionada.
Ingresa todos los datos requeridos por el sistema.	Valida los datos y los almacena en la base de datos
Cursos Alternativos	
Si el usuario y contraseña son incorrectos. Se emite un mensaje de error de los datos de autenticación.	
Si los datos ingresados son incorrectos o están en blanco. Se emite un mensaje de alerta de acuerdo al dato para ser corregido.	

Realizado por: Carlos Patiño, 2016

Asignar asignaturas

Tabla 29-3: Caso de uso expandido - Asignar asignaturas

Identificador caso de uso	CU05
Nombre del caso de uso	Asignar Asignaturas
Actores	Administrador
Propósito	Asignar las asignaturas correspondientes a cada docente.
Visión General	Los usuarios registrados con el rol de administrador son los encargados de asignar las materias correspondientes a cada docente de la unidad educativa.
Tipo	Primario
Referencia	CU01, CU02, CU03
Curso típico de eventos	
Acciones del Usuario	Respuesta del Sistema
El usuario ingresa al sistema	Pide autenticación (usuario y contraseña)
Introduce su nombre y contraseña	Valida datos ingresados y proporciona acceso al sistema
Selecciona la opción requerida para asignar asignaturas.	Muestra la pantalla de asignación de asignaturas.
Ingresa todos los datos requeridos por el sistema.	Valida los datos y los almacena en la base de datos
Cursos Alternativos	
Si el usuario y contraseña son incorrectos. Se emite un mensaje de error de los datos de autenticación.	
Si los datos ingresados son incorrectos o están en blanco. Se emite un mensaje de alerta de acuerdo al dato para ser corregido.	

Realizado por: Carlos Patiño, 2016

Registrar Notas

Tabla 30-3: Caso de uso expandido - Registrar Notas

Identificador caso de uso	CU06
Nombre del caso de uso	Registrar Notas
Actores	Administrador / Docente
Propósito	Registrar las notas de todos los estudiantes en cada una de las asignaturas.
Visión General	Los usuarios registrados con el rol de administrador o docente son los encargados de ingresar las notas de los estudiantes en cada una de sus asignaturas.
Tipo	Primario
Referencia	CU01, CU02, CU03
Curso típico de eventos	
Acciones del Usuario	Respuesta del Sistema
El usuario ingresa al sistema	Pide autenticación (usuario y contraseña)
Introduce su nombre y contraseña	Valida datos ingresados y proporciona acceso al sistema
Selecciona la opción de registro de notas.	Muestra la pantalla de registro de notas.
Ingresa todos los datos requeridos por el sistema.	Valida los datos y los almacena en la base de datos
Cursos Alternativos	
Si el usuario y contraseña son incorrectos. Se emite un mensaje de error de los datos de autenticación.	
Si los datos ingresados son incorrectos o están en blanco. Se emite un mensaje de alerta de acuerdo al dato para ser corregido.	

Realizado por: Carlos Patiño, 2016

Registrar matrículas

Tabla 31-3: Caso de uso expandido - Registro de matrículas

Identificador caso de uso	CU07
Nombre del caso de uso	Registrar Matrículas
Actores	Administrador / Secretaria
Propósito	Ingresar los datos de los estudiantes para proceder a registrar su matrícula.
Visión General	Los usuarios registrados con el rol de administrador o secretaria son los encargados de ingresar los datos necesarios para proceder a registrar las matrículas.
Tipo	Primario
Referencia	CU01, CU02, CU03
Curso típico de eventos	
Acciones del Usuario	Respuesta del Sistema
El usuario ingresa al sistema	Pide autenticación (usuario y contraseña)
Introduce su nombre y contraseña	Valida datos ingresados y proporciona acceso al sistema
Selecciona la opción de registro de estudiantes y matrículas.	Muestra la pantalla de registro de notas.
Ingresar todos los datos requeridos por el sistema.	Valida los datos y los almacena en la base de datos
Cursos Alternativos	
Si el usuario y contraseña son incorrectos. Se emite un mensaje de error de los datos de autenticación.	
Si los datos ingresados son incorrectos o están en blanco. Se emite un mensaje de alerta de acuerdo al dato para ser corregido.	

Realizado por: Carlos Patiño, 2016

Emisión de reportes

Tabla 32-3: Caso de uso expandido - Emisión de reportes

Identificador caso de uso	CU08
Nombre del caso de uso	Emisión de Reportes
Actores	Administrador, Secretaria, Docente
Propósito	Generar reportes de las diferentes actividades realizadas.
Visión General	Los usuarios dependiendo del su rol pueden generar reportes de sus diferentes actividades realizadas en el sistema.
Tipo	Primario
Curso típico de eventos	
Acciones del Usuario	Respuesta del Sistema
El usuario ingresa al sistema	Pide autenticación (usuario y contraseña)
Introduce su nombre y contraseña	Valida datos ingresados y proporciona acceso al sistema
Selecciona la opción de generar reportes	Muestra los diferentes tipos de reportes y solicita elección del reporte requerido.
Selecciona el reporte requerido.	Valida los datos y genera el reporte en formato PDF.
Cursos Alternativos	
Si el usuario y contraseña son incorrectos. Se emite un mensaje de error de los datos de autenticación.	

Realizado por: Carlos Patiño, 2016

3.3.1.2 Diagramas de Secuencia del Sistema

Para cada caso de uso construimos una serie de diagramas que muestren la secuencia que sigue un determinado requerimiento del sistema. Los diagramas de secuencia se definen en base a los siguientes requerimientos:

- **Requerimiento 1:** Autenticación de Usuarios
- **Requerimiento 2:** Gestión de Cuentas de Usuario
- **Requerimiento 3:** Gestión de Periodos Académicos
- **Requerimiento 4:** Gestión de Cursos

- **Requerimiento 5:** Gestión de Asignaturas
- **Requerimiento 6:** Asignar Asignaturas
- **Requerimiento 7:** Registrar Notas
- **Requerimiento 8:** Registrar Matrículas
- **Requerimiento 9:** Emisión de Reportes

Autenticación de Usuarios

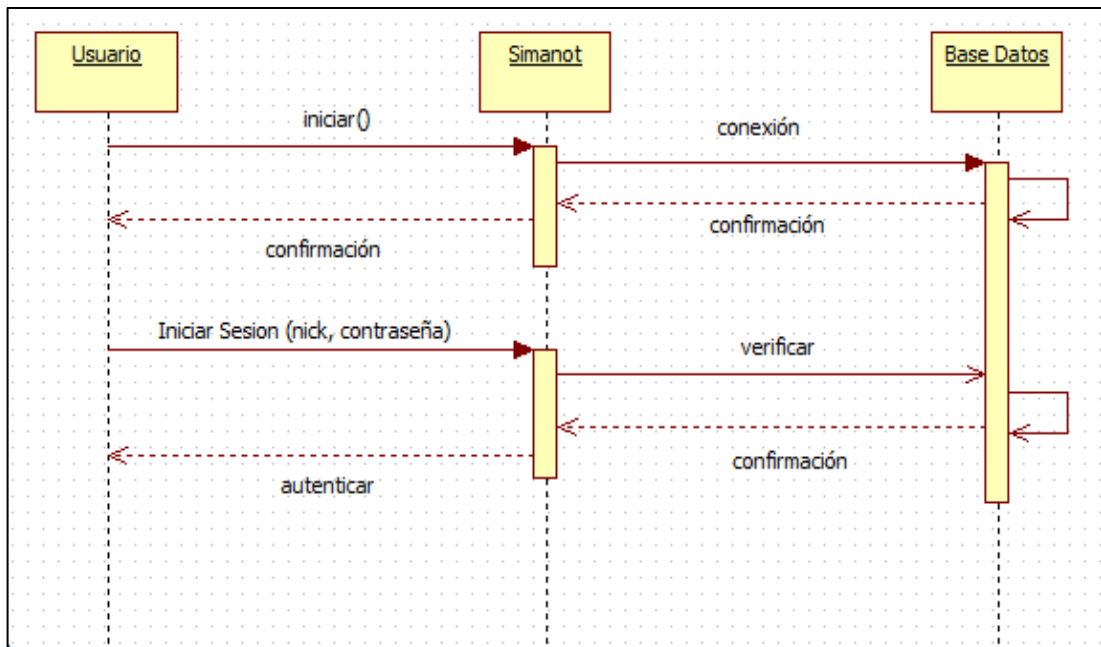


Figura 10-3: Diagrama de secuencia - Autenticación de usuarios

Realizado por: Carlos Patiño, 2016

Gestión de Cuentas de Usuario

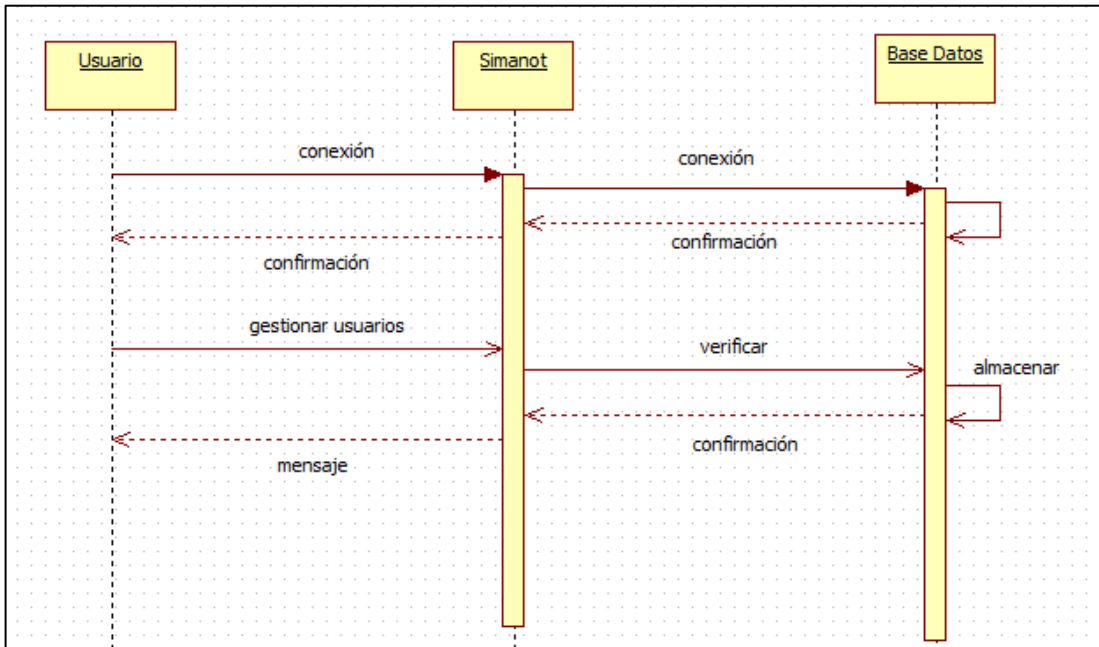


Figura 11-3: Diagrama de secuencia - Gestión de cuentas de usuarios

Realizado por: Carlos Patiño, 2016

Gestión de Periodos Académicos

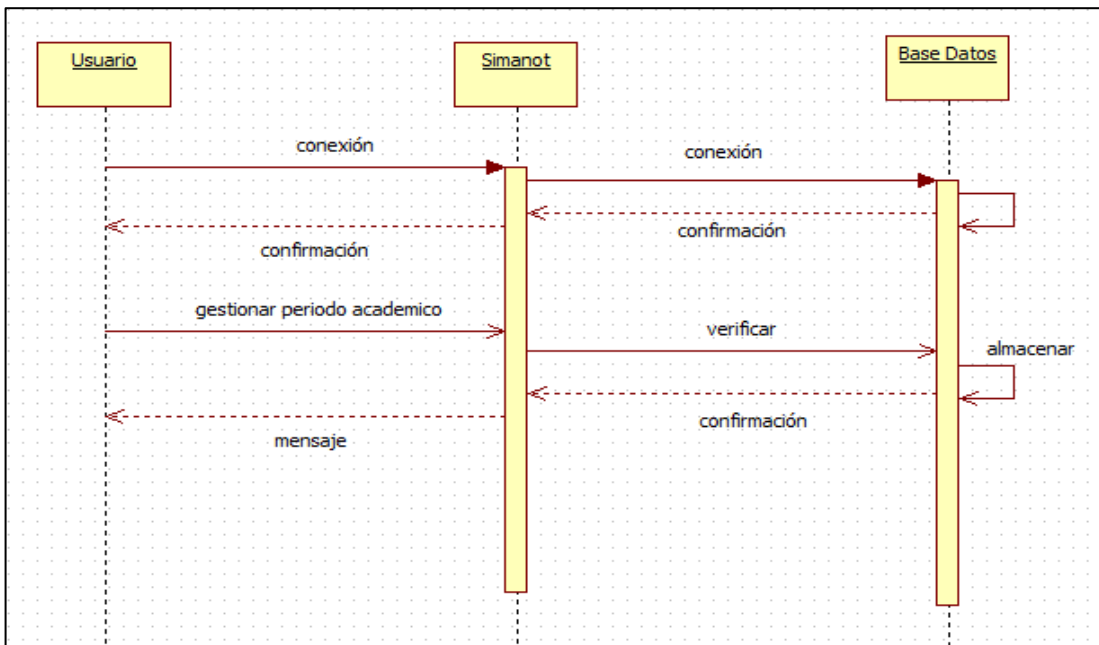


Figura 12-3: Diagrama de secuencia - Gestión de periodos académicos

Realizado por: Carlos Patiño, 2016

Gestión de Cursos

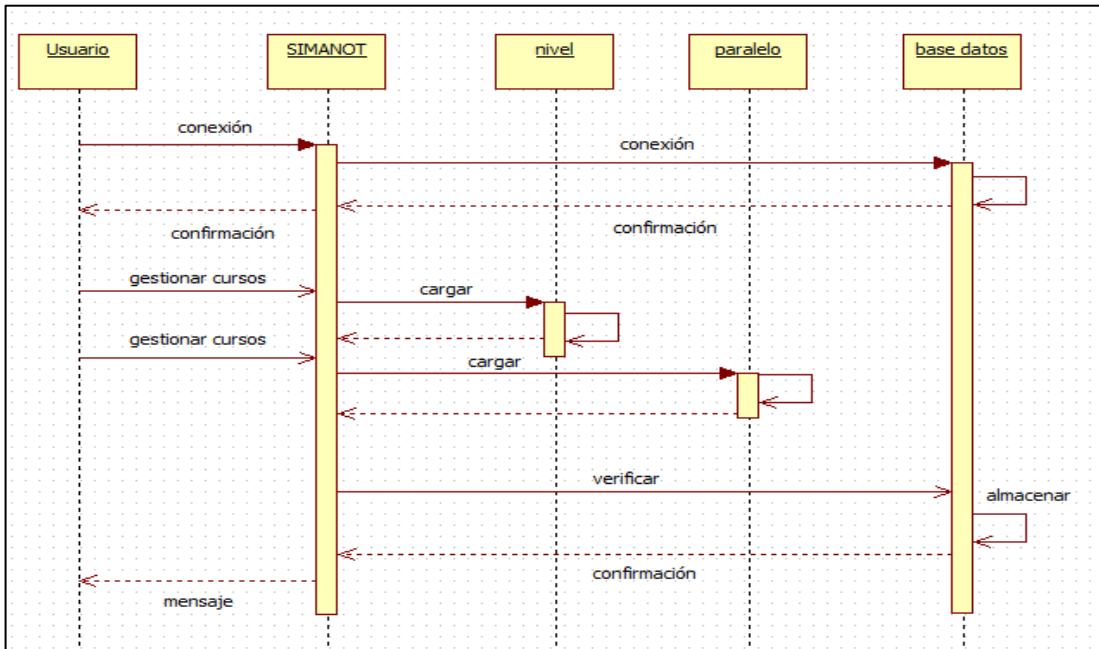


Figura 13-3: Diagrama de secuencia - Gestión de cursos

Realizado por: Carlos Patiño, 2016

Gestión de Asignaturas

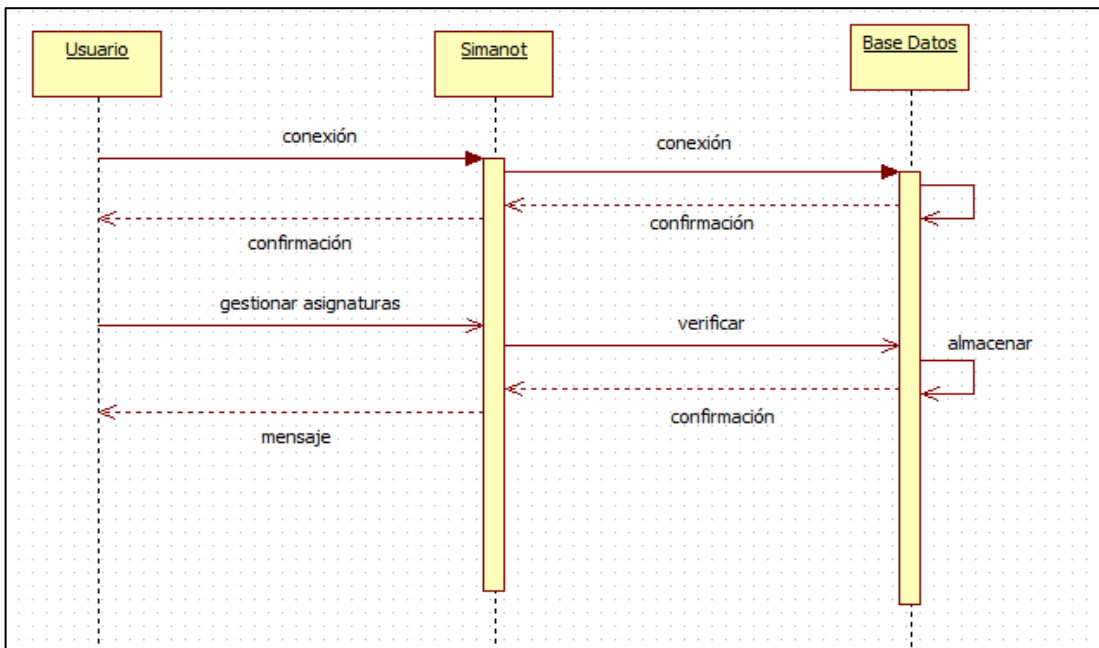


Figura 14-3: Diagrama de secuencia - Gestión de asignaturas

Realizado por: Carlos Patiño, 2016

Asignar Asignaturas

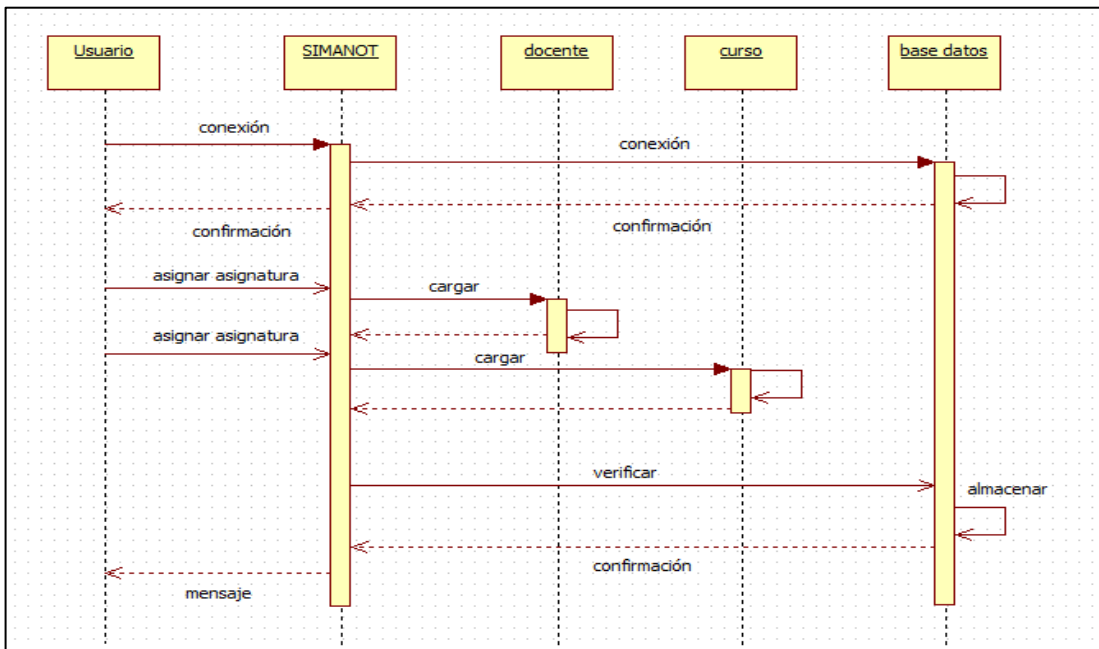


Figura 15-3: Diagrama de secuencia - Asignar asignaturas

Realizado por: Carlos Patiño, 2016

Registrar Notas

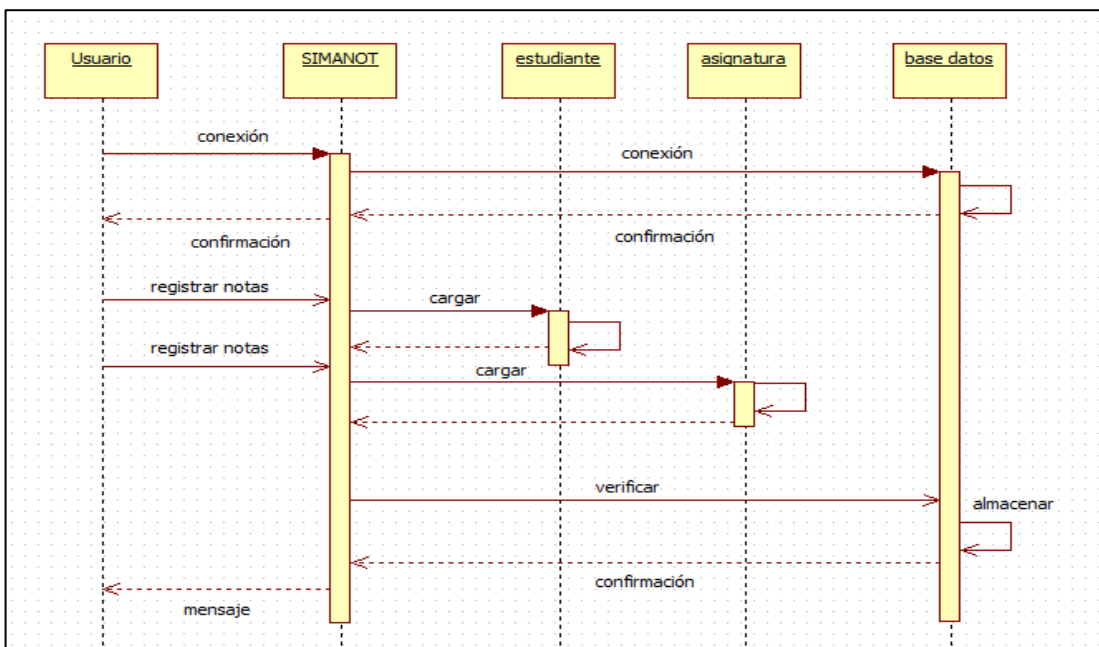


Figura 16-3: Diagrama de secuencia - Registro de notas

Realizado por: Carlos Patiño, 2016

Registrar Matrículas

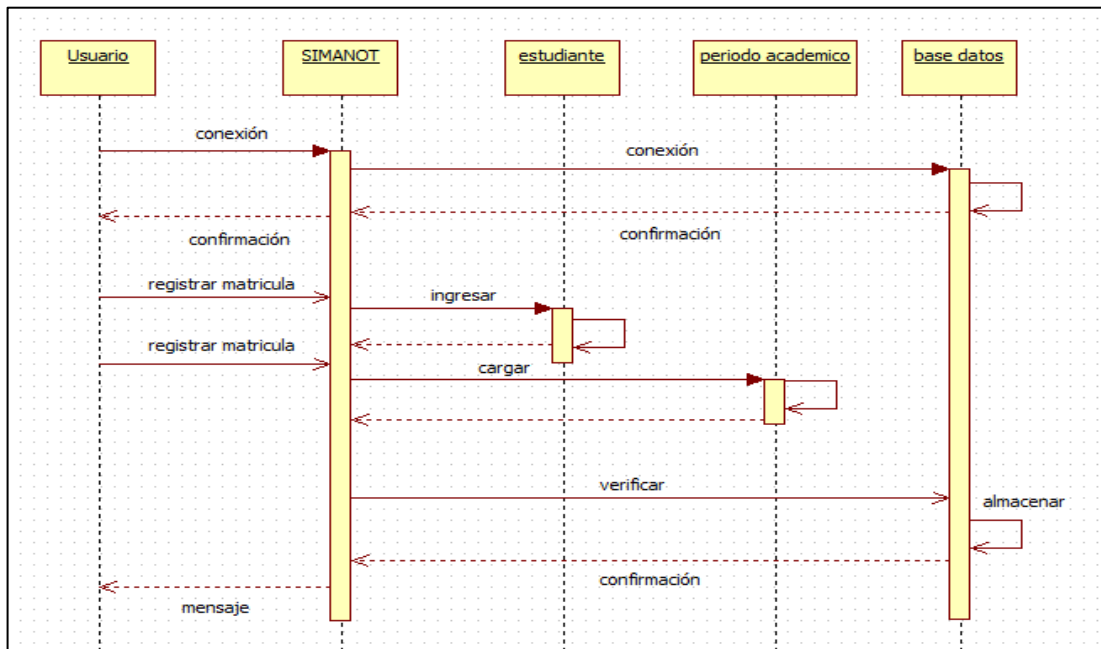


Figura 17-3: Diagrama de secuencia - Registro de matrículas

Realizado por: Carlos Patiño, 2016

Emisión de Reportes

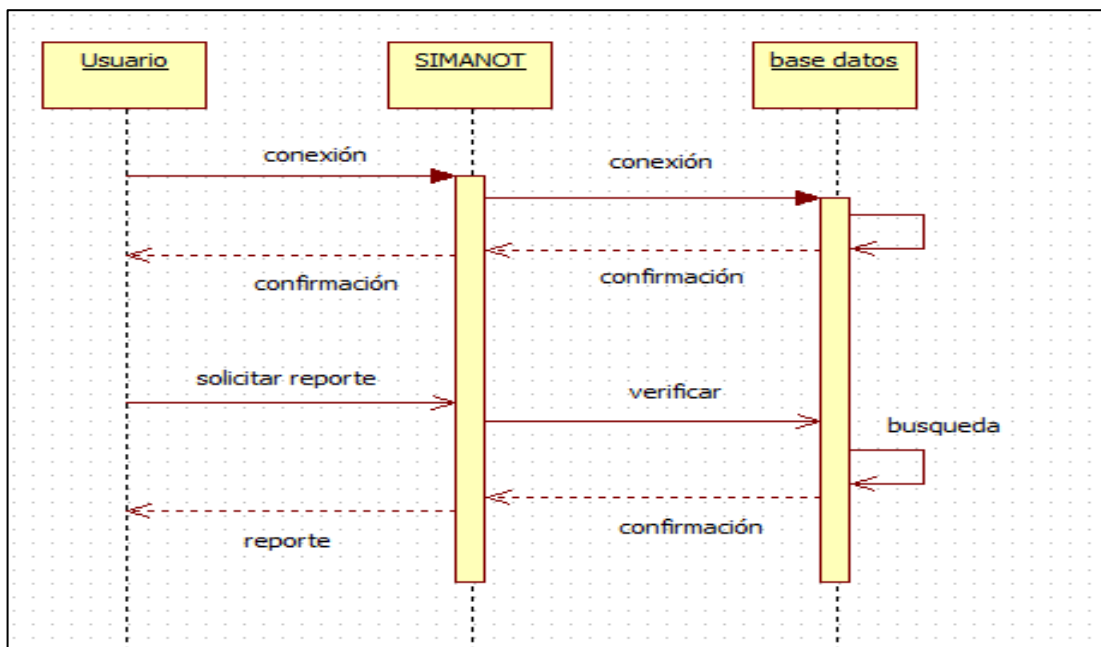


Figura 18-3: Diagrama de secuencia - Emisión de reportes

Realizado por: Carlos Patiño, 2016

3.3.1.3. Diagramas de Estado

En estos diagramas identificamos cuales los casos de uso que pueden caracterizar un posible cambio en el sistema, en relación a los sucesos y al tiempo. . Los diagramas de estado se definen en base a los siguientes requerimientos:

Los requerimientos establecidos son los siguientes:

- **Requerimiento 1:** Gestión de Cuentas de Usuario
- **Requerimiento 2:** Gestión de Periodo Académico
- **Requerimiento 3:** Gestión de Asignaturas

Gestión de Cuentas de Usuarios

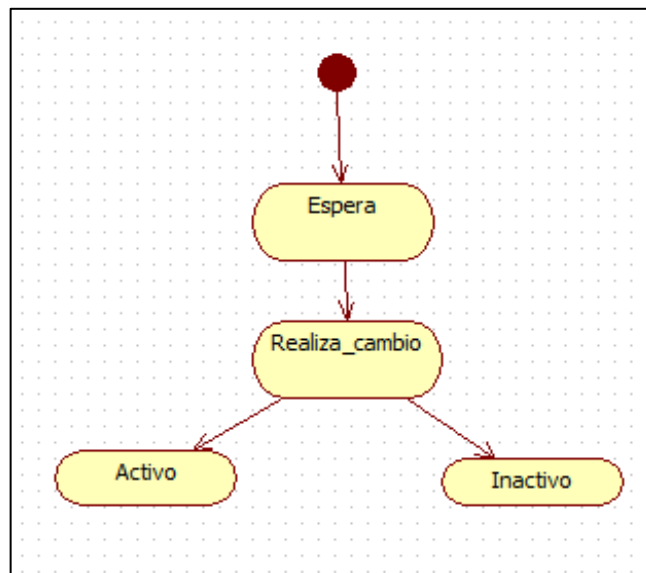


Figura 19-3: Diagrama de estado - Gestión de usuarios

Realizado por: Carlos Patiño, 2016

Gestión de Periodo académico

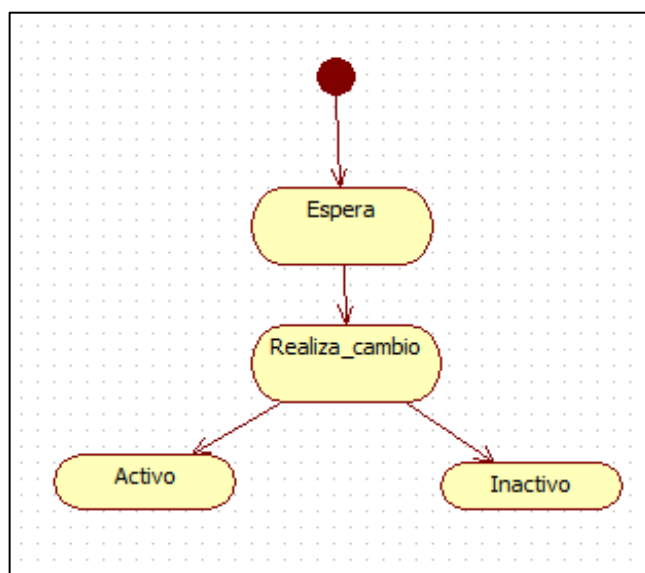


Figura 20-3: Diagrama de estado - Gestión de periodos académicos

Realizado por: Carlos Patiño, 2016

Gestión de Asignatura

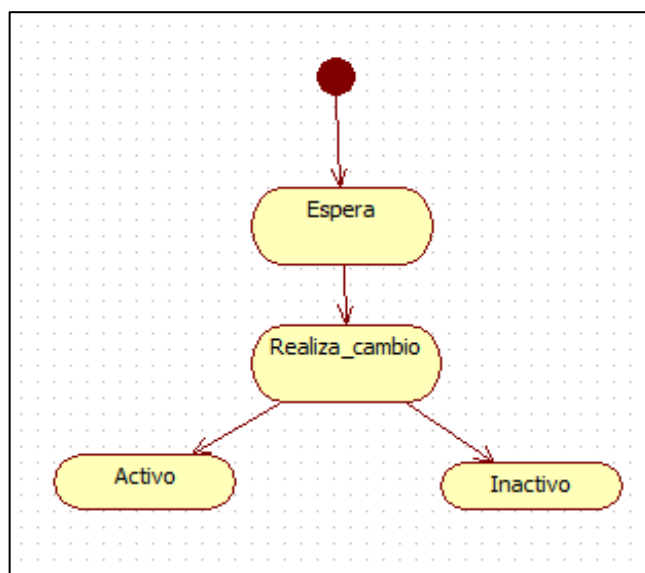


Figura 21-3: Diagrama de estado - Gestión de asignaturas

Realizado por: Carlos Patiño, 2016

3.3.1.4. Modelo Conceptual

Este modelo identificamos el esquema de la base de datos con sus respectivas tablas y relaciones, realizado a partir de los requerimientos.

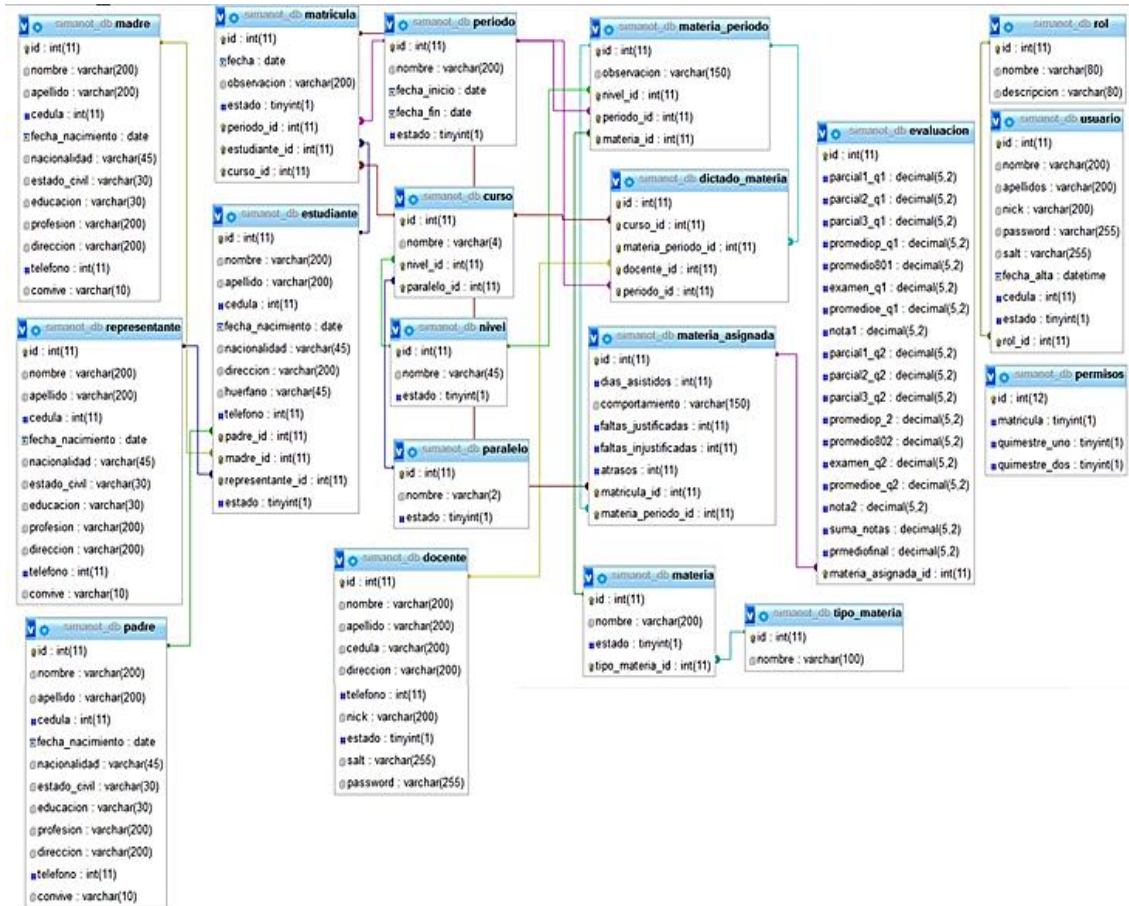


Figura 22-3: Modelo conceptual de la base de datos

Realizado por: Carlos Patiño, 2016

3.3.1.5. Diagrama de Actividades

En este diagrama modelamos a partir del comportamiento dinámico de un caso de uso, para mostrar la secuencia de la actividad que se lleva a cabo. Los diagramas de actividades se definen en base a los siguientes requerimientos:

- **Requerimiento 1:** Autenticación de Usuarios
- **Requerimiento 2:** Gestión de Cuentas de Usuario
- **Requerimiento 3:** Gestión de Periodos Académicos

- **Requerimiento 4:** Gestión de Cursos
- **Requerimiento 5:** Gestión de Asignaturas
- **Requerimiento 6:** Asignar Asignaturas
- **Requerimiento 7:** Registrar Notas
- **Requerimiento 8:** Registrar Matrículas
- **Requerimiento 9:** Emisión de Reportes

Autenticación de Usuarios

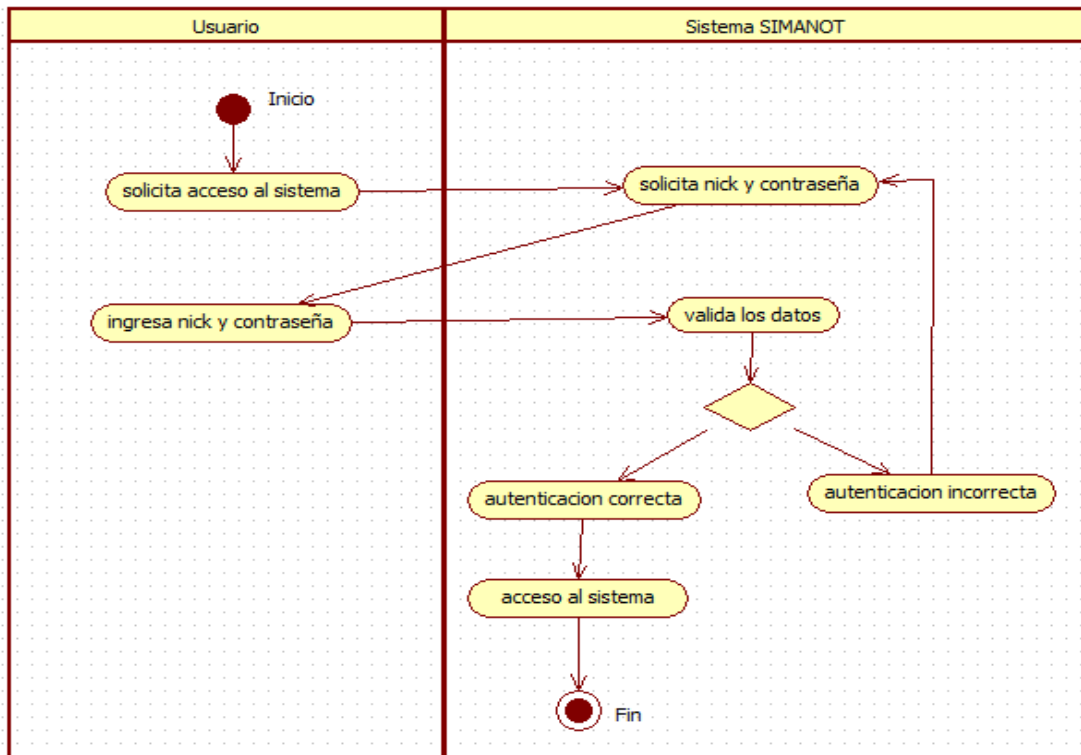


Figura 23-3: Diagrama de actividades - Inicio de sesión

Realizado por: Carlos Patiño, 2016

Gestión de Cuentas de Usuario

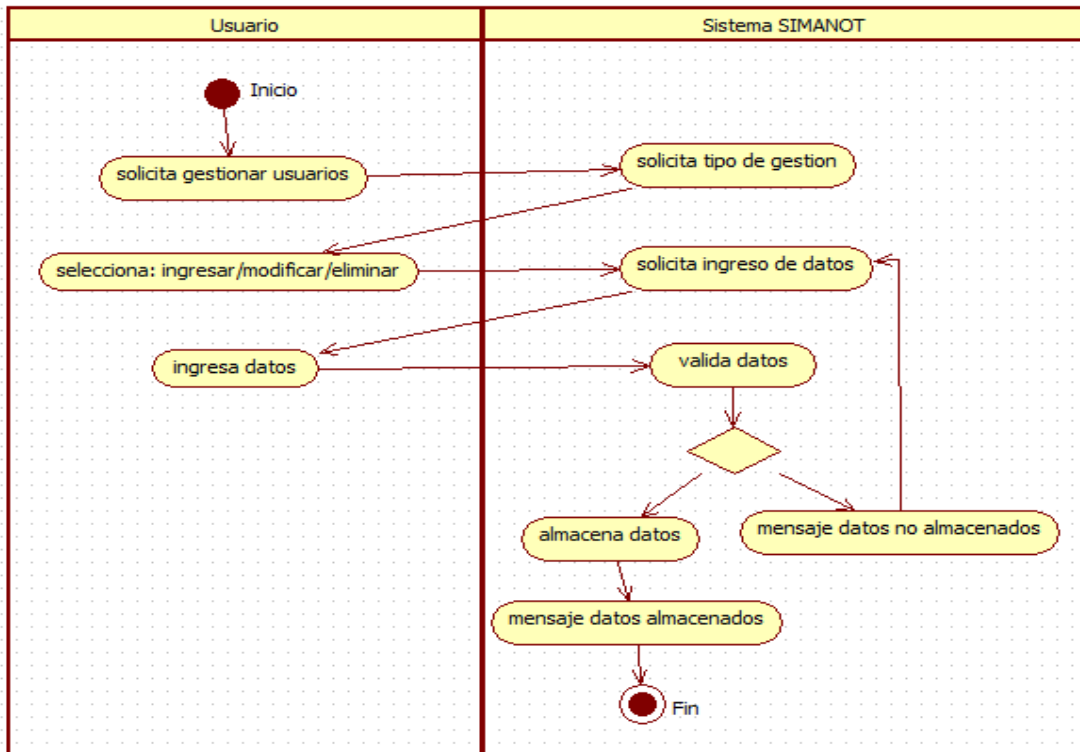


Figura 24-3: Diagrama de actividades - Gestión de cuentas de usuario

Realizado por: Carlos Patiño, 2016

Gestión de Periodos Académicos

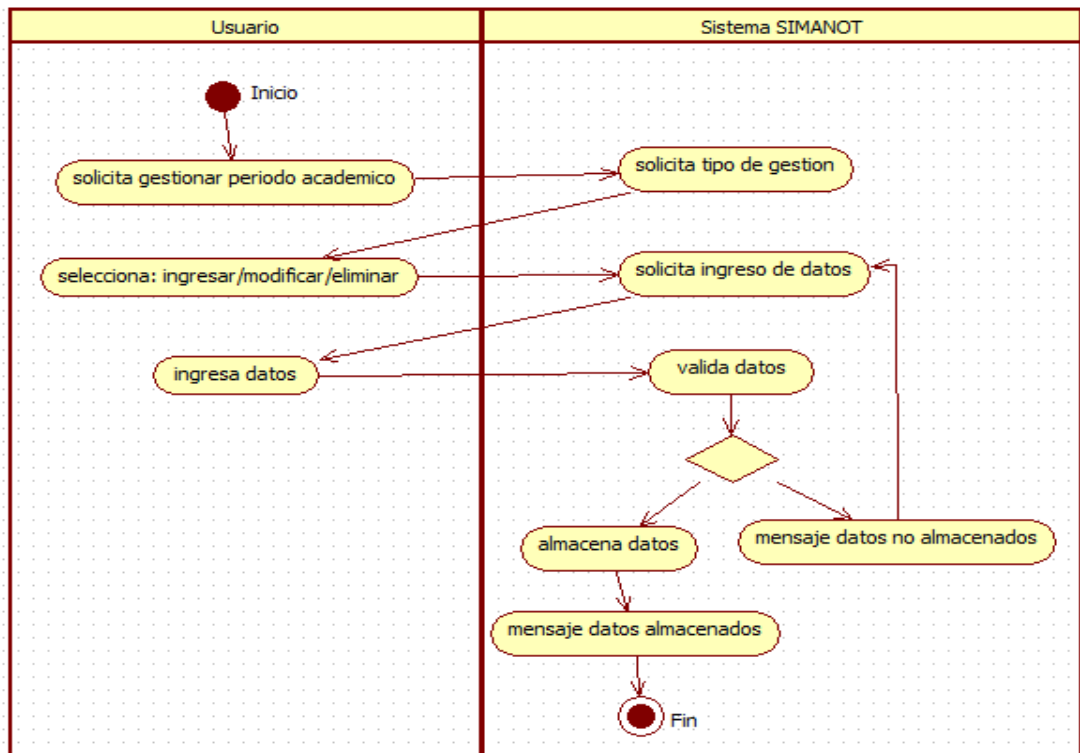


Figura 25-3: Diagrama de actividades - Gestión de periodos académicos

Realizado por: Carlos Patiño, 2016

Gestión de Cursos

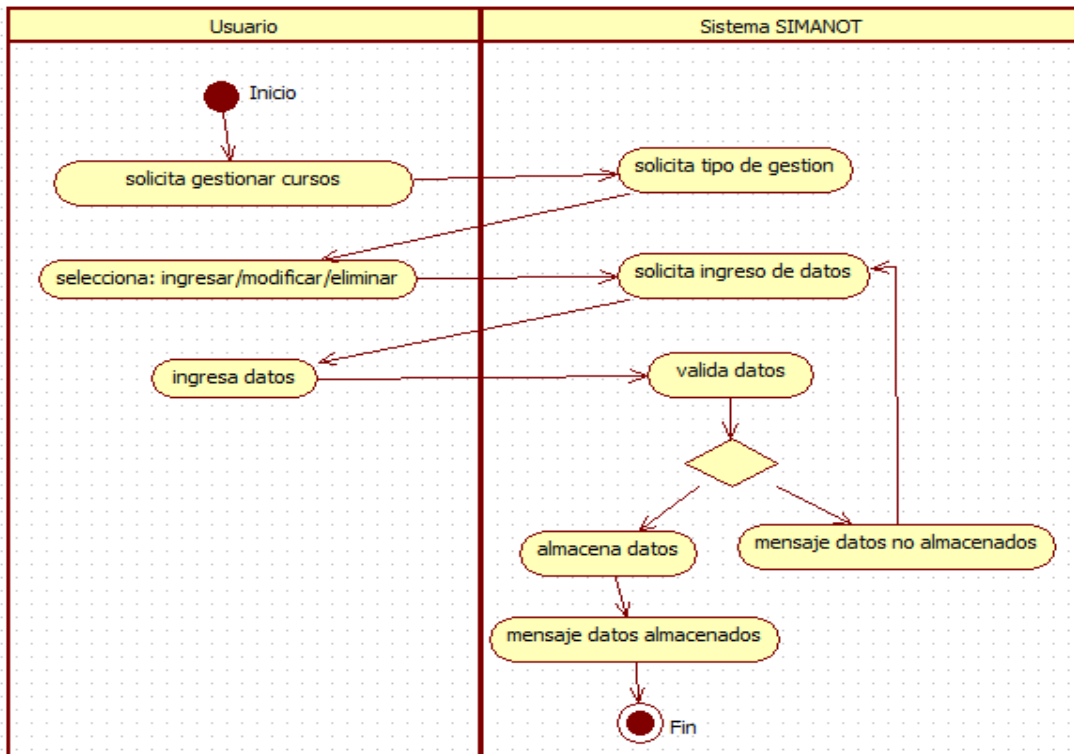


Figura 26-3: Diagrama de actividades - Gestión de cursos

Realizado por: Carlos Patiño, 2016

Gestión de Asignaturas

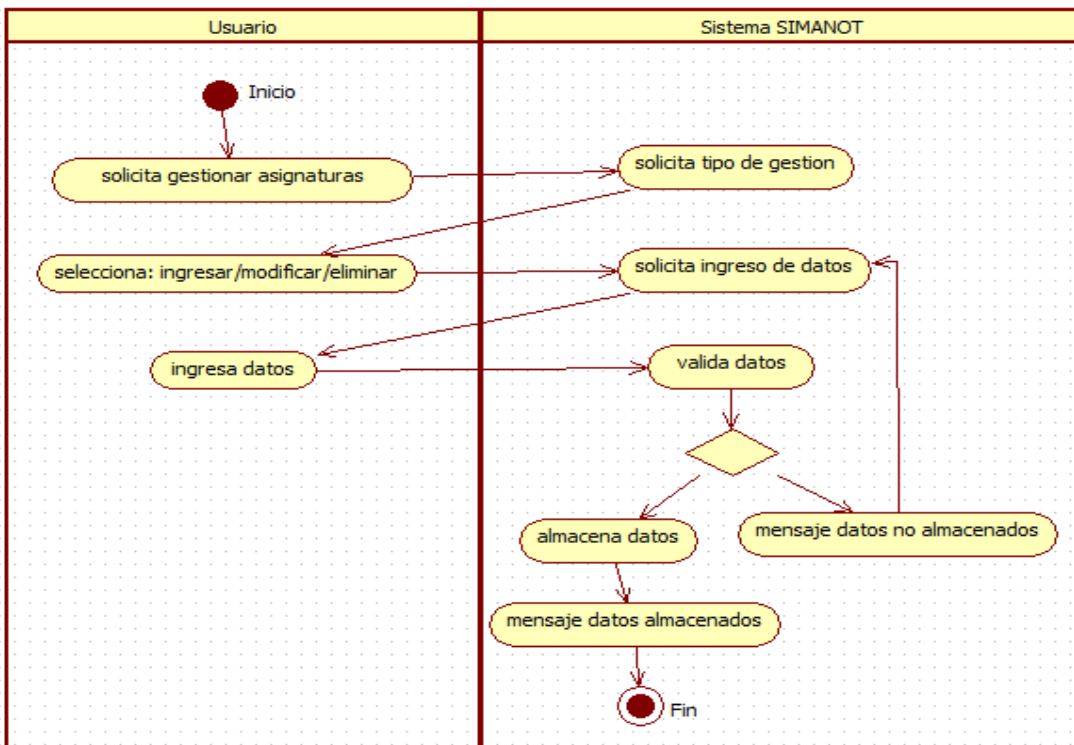


Figura 27-3: Diagrama de actividades - Gestión de asignaturas

Realizado por: Carlos Patiño, 2016

Asignar Asignaturas

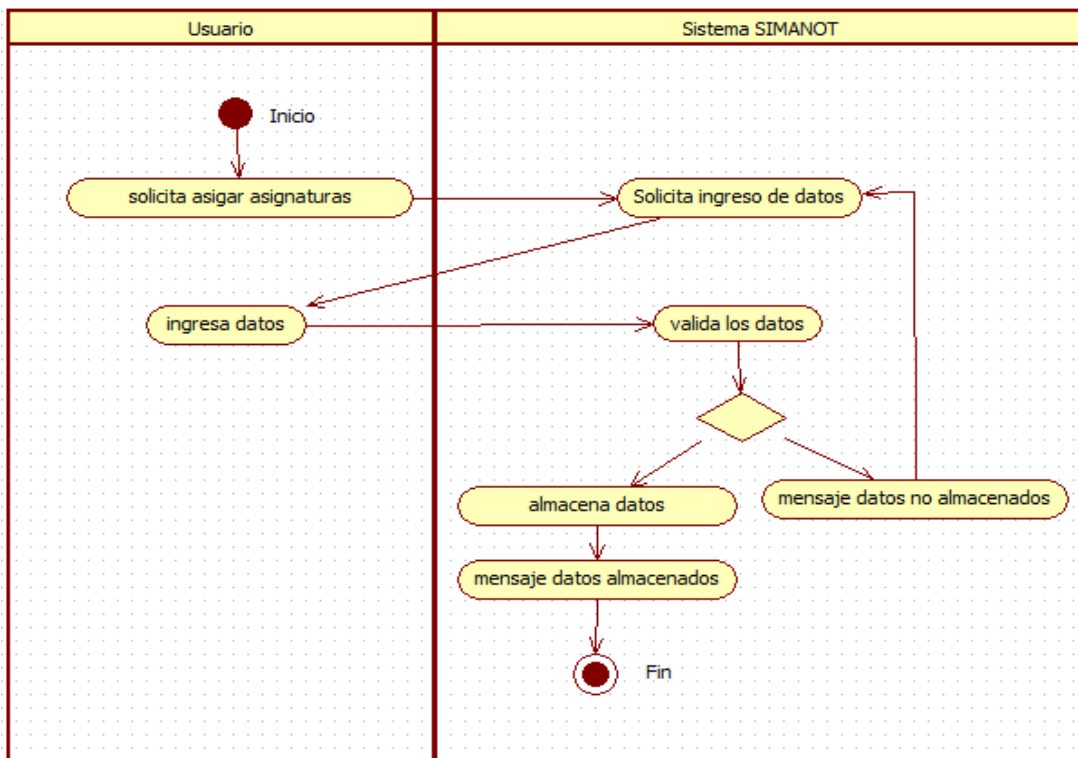


Figura 28-3: Diagrama de actividades - Asignar Asignaturas

Realizado por: Carlos Patiño, 2016

Registrar Notas

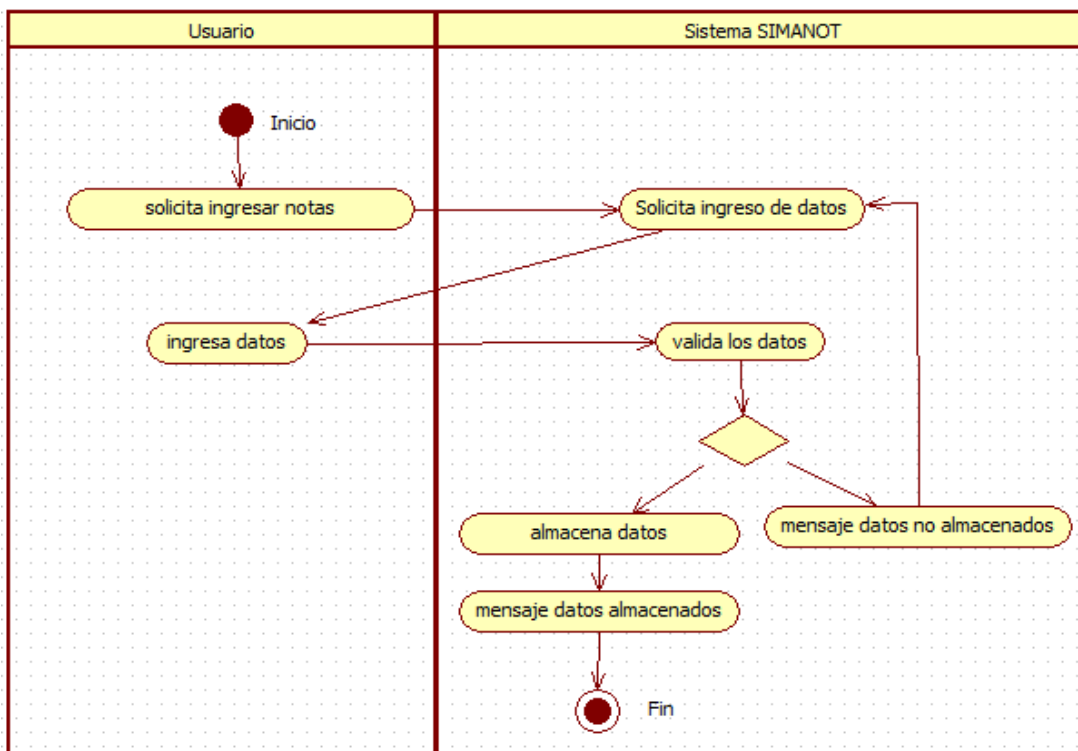


Figura 29-3: Diagrama de actividades - Registrar Notas

Realizado por: Carlos Patiño, 2016

Registrar Matrículas

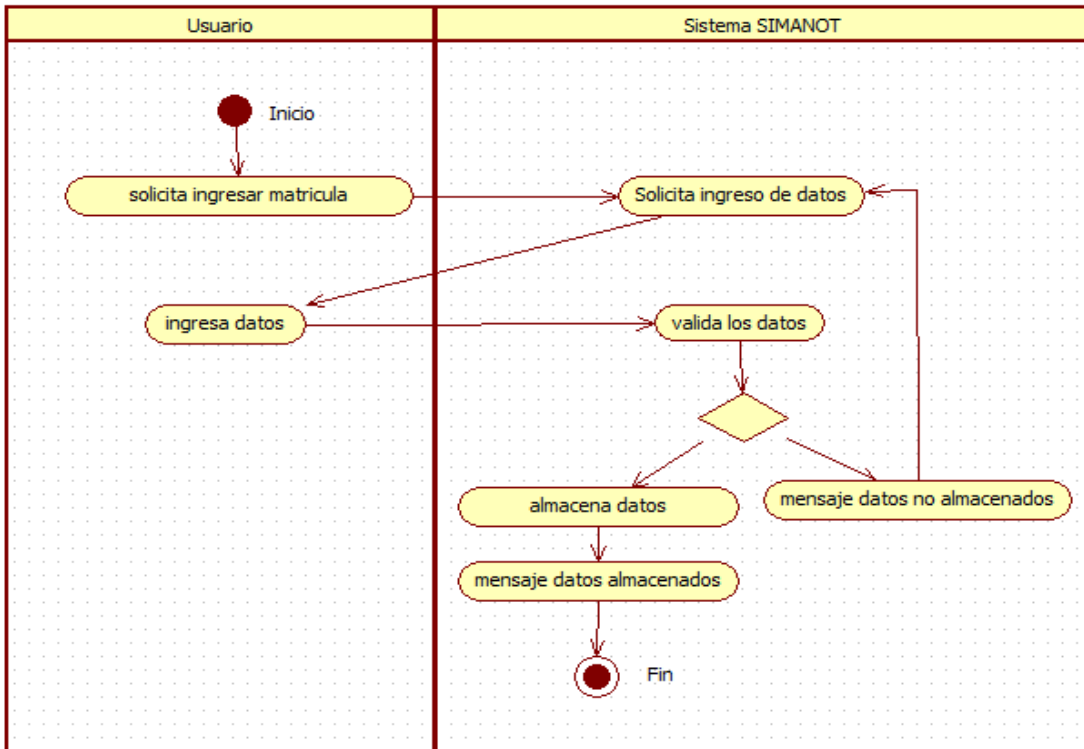


Figura 30-3: Diagrama de actividades - Registrar matrículas

Realizado por: Carlos Patiño, 2016

Emisión de Reportes

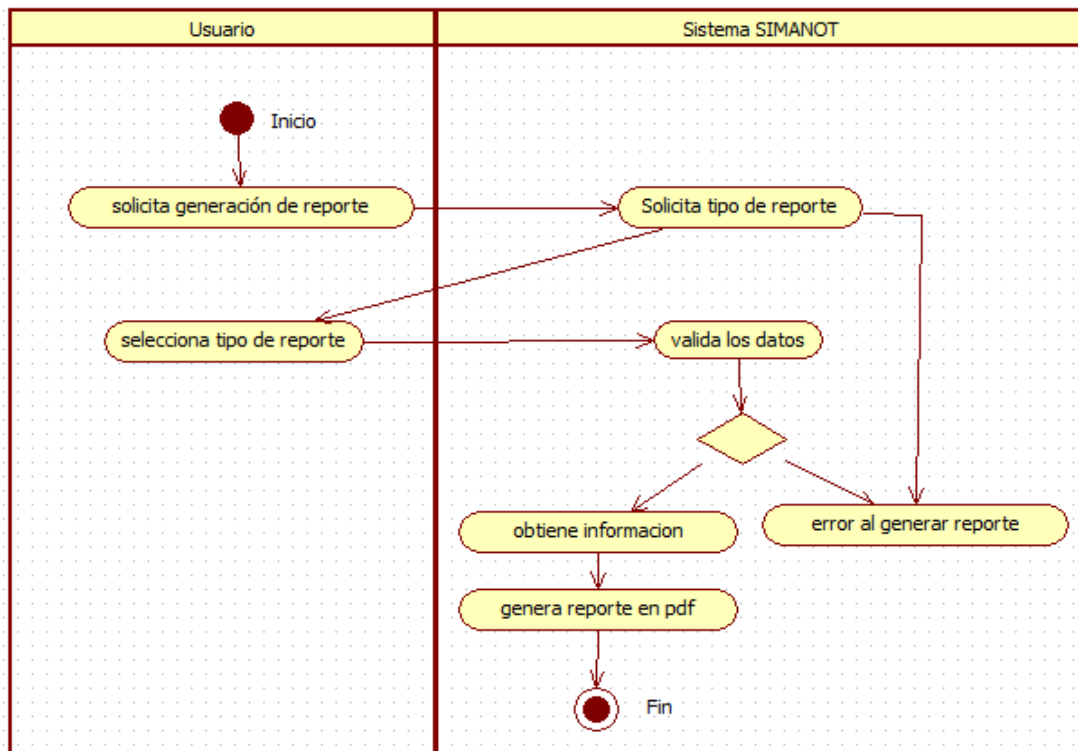


Figura 31-3: Diagrama de actividades - Emisión de reportes

Realizado por: Carlos Patiño, 2016, 2016

3.3.2 Diseño

En esta etapa tomamos en cuenta todos los componentes del sistema para definir su arquitectura y tener una visión global del mismo, las relaciones entre si y el ambiente en el que interactúa.

3.3.2.1 Definición de la Arquitectura

La arquitectura que se utilizó para el desarrollo del sistema fue la arquitectura MVC (Modelo, Vista, Controlador) utilizada por el framework Symfony, está formado por tres niveles:

- El Modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La Vista transforma el modelo en una página web que permite al usuario interactuar con ella.
- El Controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. Si por ejemplo una misma aplicación debe ejecutarse tanto en un navegador estándar como un navegador de un dispositivo móvil, solamente es necesario crear una vista nueva para cada dispositivo; manteniendo el controlador y el modelo original. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, email, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación. (Potencier, 2007, http://librosweb.es/libro/symfony_1_4/capitulo_2/el_patron_mvc.html)

A continuación se muestra la imagen de la arquitectura del sistema

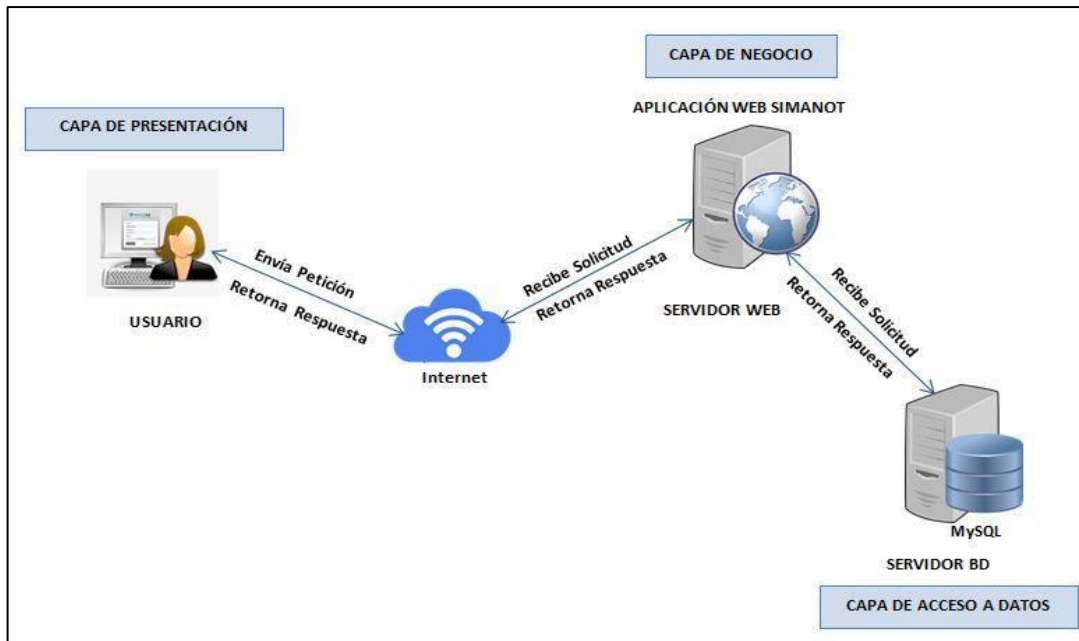


Figura 32-3: Arquitectura del sistema

Realizado por: Carlos Patiño

3.3.2.2 Diagramas de Colaboración

Para realizar este diagrama identificamos las interacciones que existen entre los objetos del sistema, con la finalidad de representar un objetivo un común. Los diagramas de colaboración se definen en base a los siguientes requerimientos:

- **Requerimiento 1:** Autenticación de Usuarios
- **Requerimiento 2:** Gestión de Cuentas de Usuario
- **Requerimiento 3:** Gestión de Periodos Académicos
- **Requerimiento 4:** Gestión de Cursos
- **Requerimiento 5:** Gestión de Asignaturas
- **Requerimiento 6:** Asignar Asignaturas
- **Requerimiento 7:** Registrar Notas
- **Requerimiento 8:** Registrar Matrículas
- **Requerimiento 9:** Emisión de Reportes

Autenticación de Usuarios

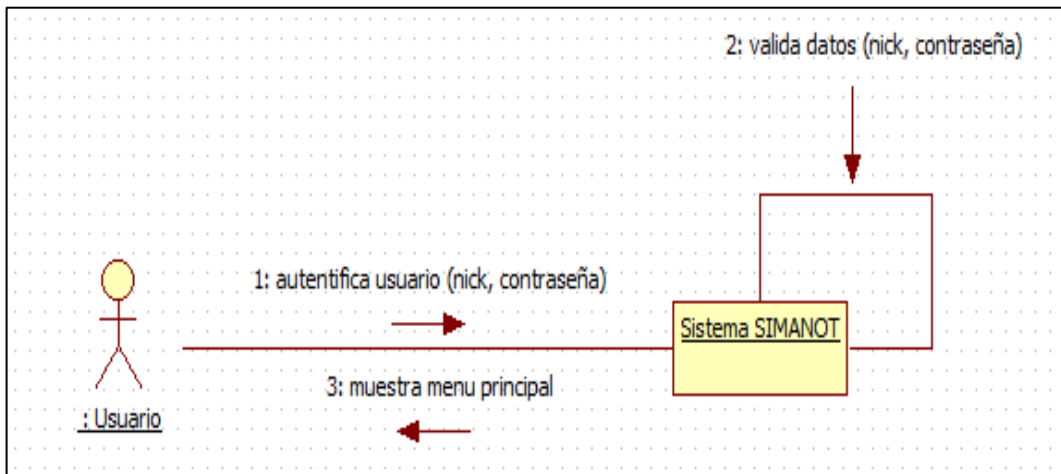


Figura 33-3: Diagrama de colaboración - Autenticación de usuarios

Realizado por: Carlos Patiño, 2016

Gestión de Cuentas de usuario

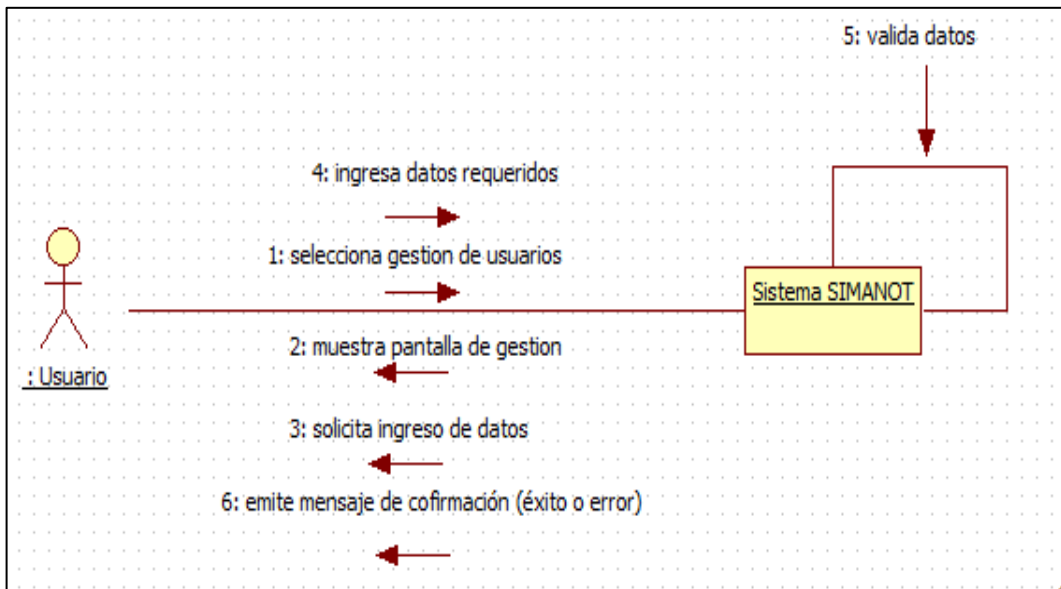


Figura 34-3: Diagrama de colaboración - Gestión de cuentas de usuarios

Realizado por: Carlos Patiño, 2016

Gestión de Periodos Académicos

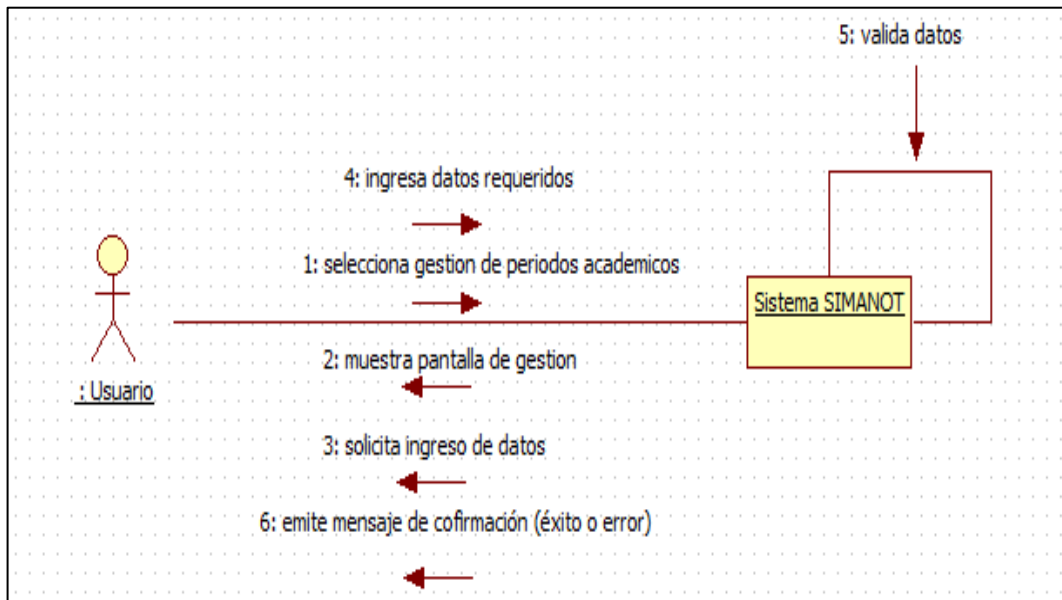


Figura 35-3: Diagrama de colaboración - Gestión de periodos académicos

Realizado por: Carlos Patiño, 2016

Gestión de Cursos

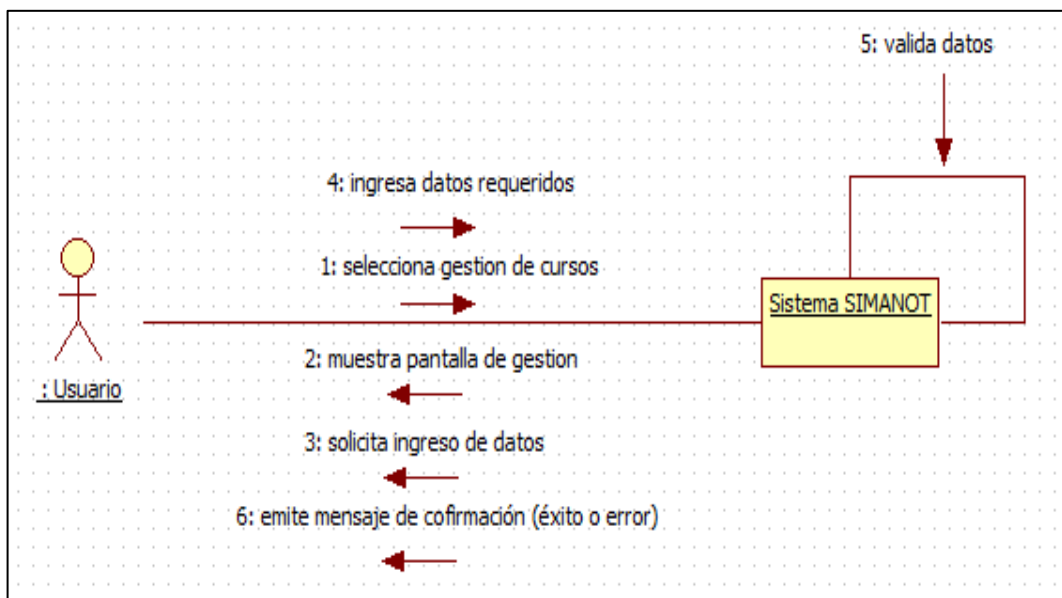


Figura 36-3: Diagrama de colaboración - Gestión de cursos

Realizado por: Carlos Patiño, 2016

Gestión de Asignaturas

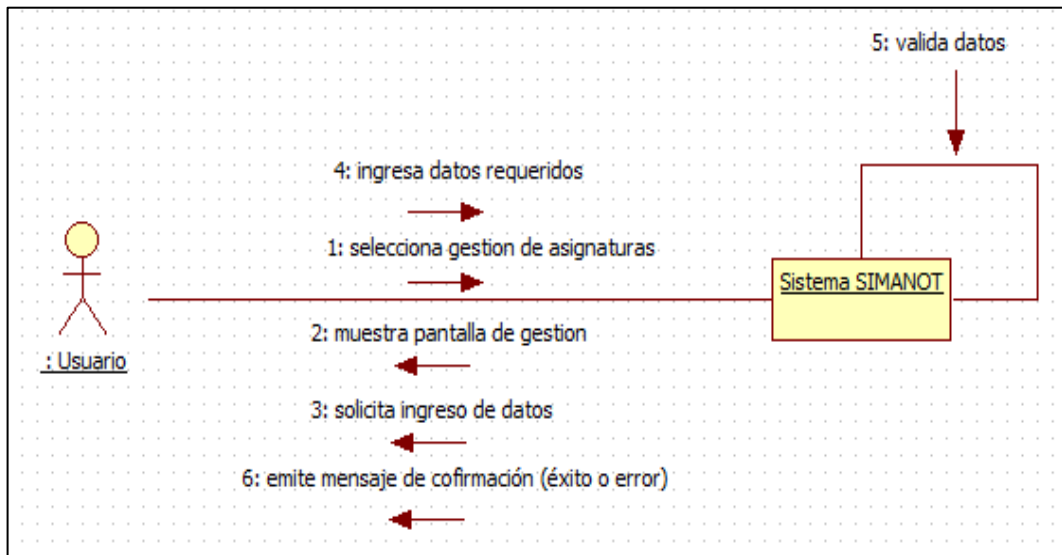


Figura 37-3: Diagrama de colaboración - Gestión de asignaturas

Realizado por: Carlos Patiño, 2016

Asignar Asignaturas

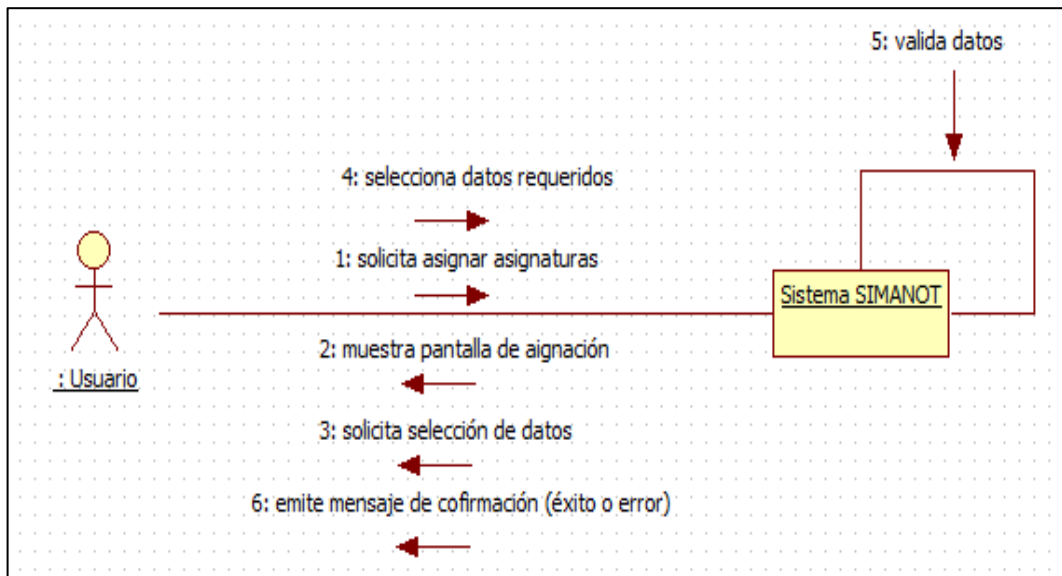


Figura 38-3: Diagrama de colaboración - Asignar asignaturas

Realizado por: Carlos Patiño, 2016

Registrar Notas

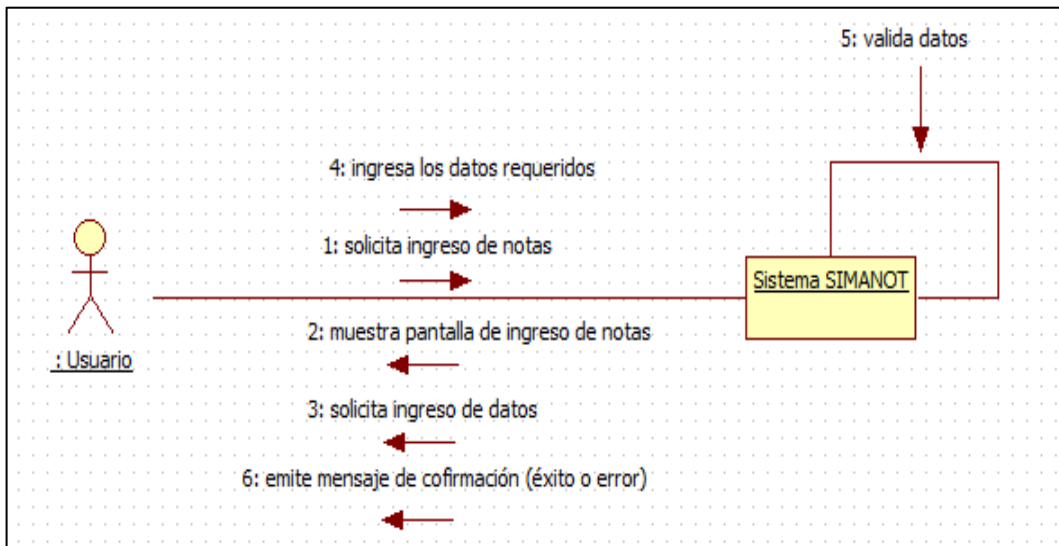


Figura 39-3: Diagrama de colaboración - Registro de notas

Realizado por: Carlos Patiño, 2016

Registrar Matrículas

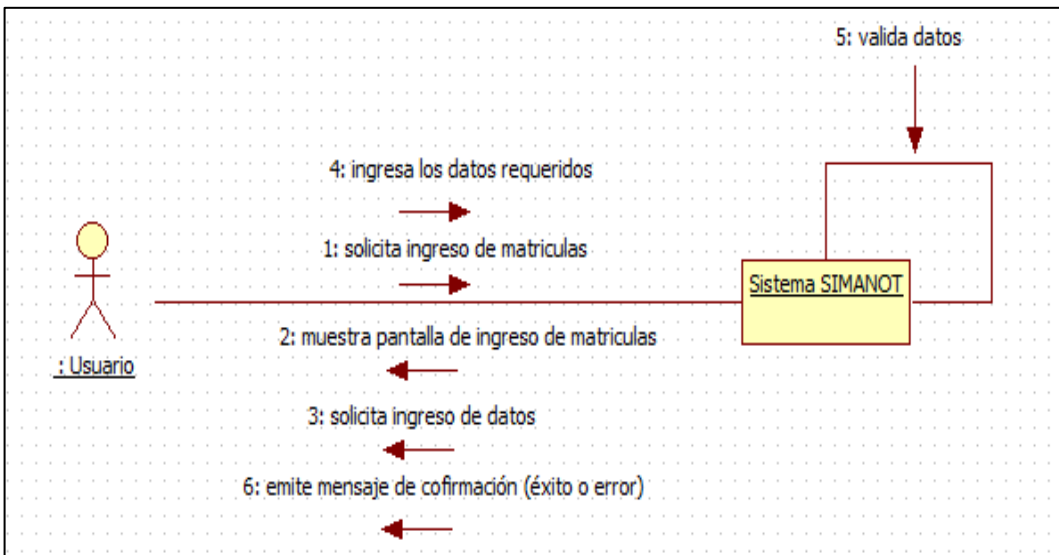


Figura 40-3: Diagrama de colaboración - Registro de matrículas

Realizado por: Carlos Patiño, 2016

Emisión de Reportes

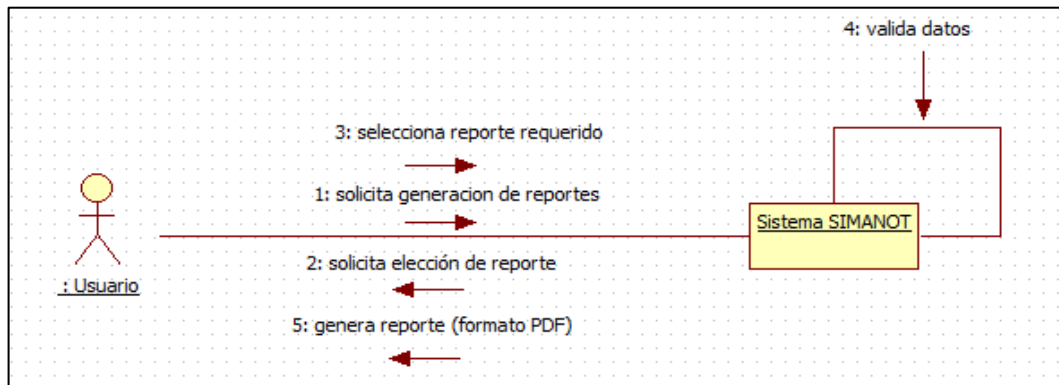


Figura 41-3: Diagrama de colaboración - Emisión de reportes

Realizado por: Carlos Patiño, 2016

3.3.2.3 Diagrama de Clases de Diseño

En estos diagramas identificamos todas las clases del sistema, junto a sus métodos, atributos y las relaciones que existen entre ellas.

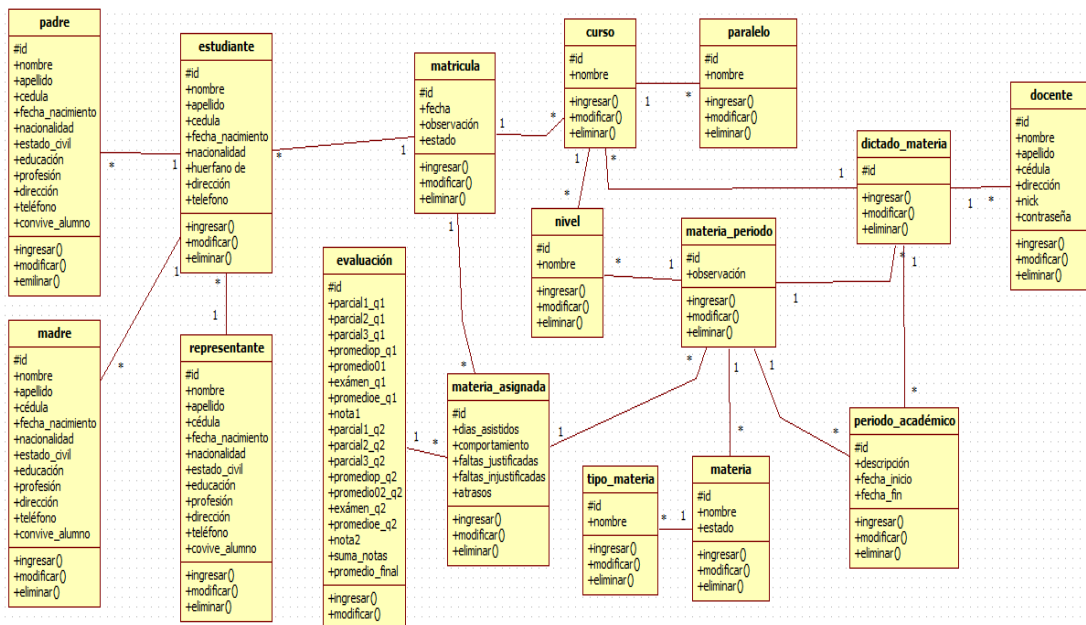


Figura 42-3: Diagrama de clase de diseño

Realizado por: Carlos Patiño, 2016

3.3.2.4 Diagrama de Componentes

En este diagrama identificamos todos los componentes en los que se divide el sistema, y la relación que existe entre ellos.

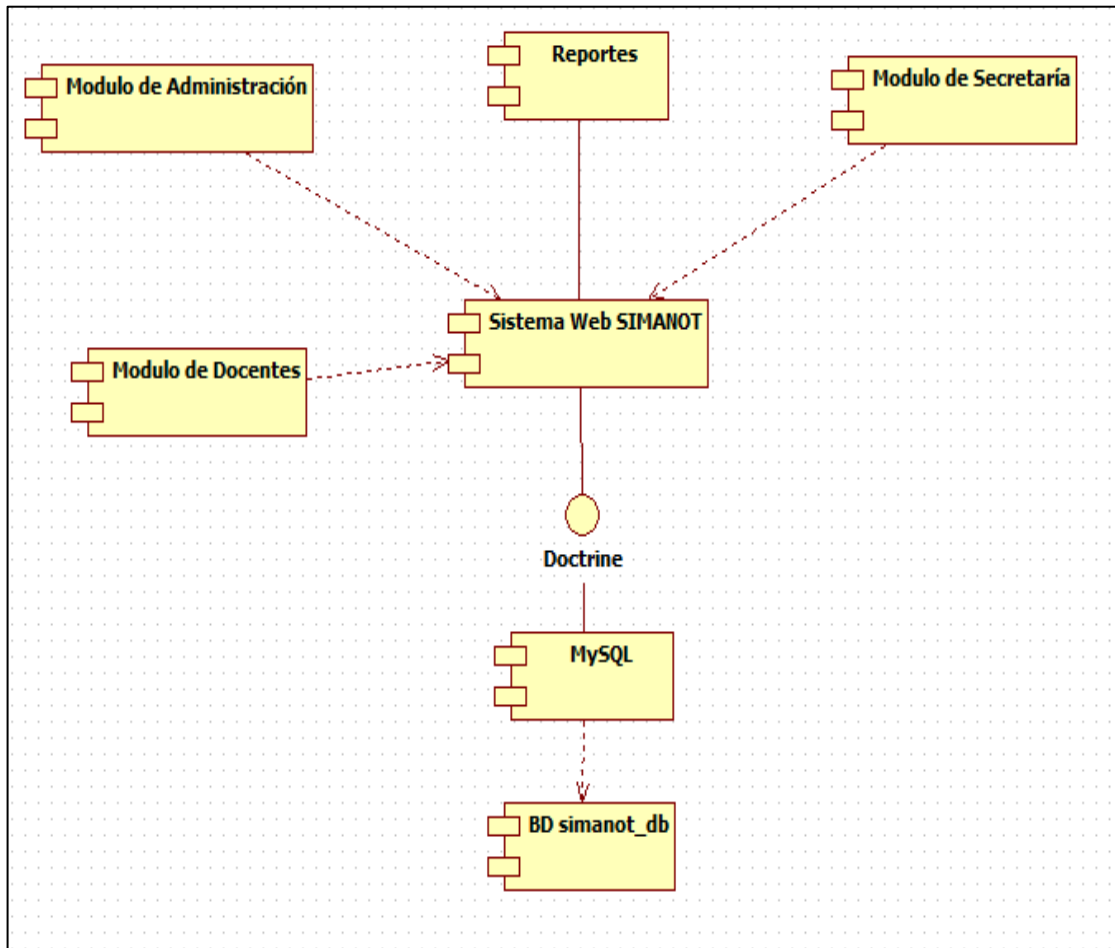


Figura 43-3: Diagrama de componentes

Realizado por: Carlos Patiño, 2016

3.3.2.5 Diagramas de Despliegue

En este diagrama lo realizamos tomamos en cuenta las relaciones físicas de los distintos nodos que componen al sistema.

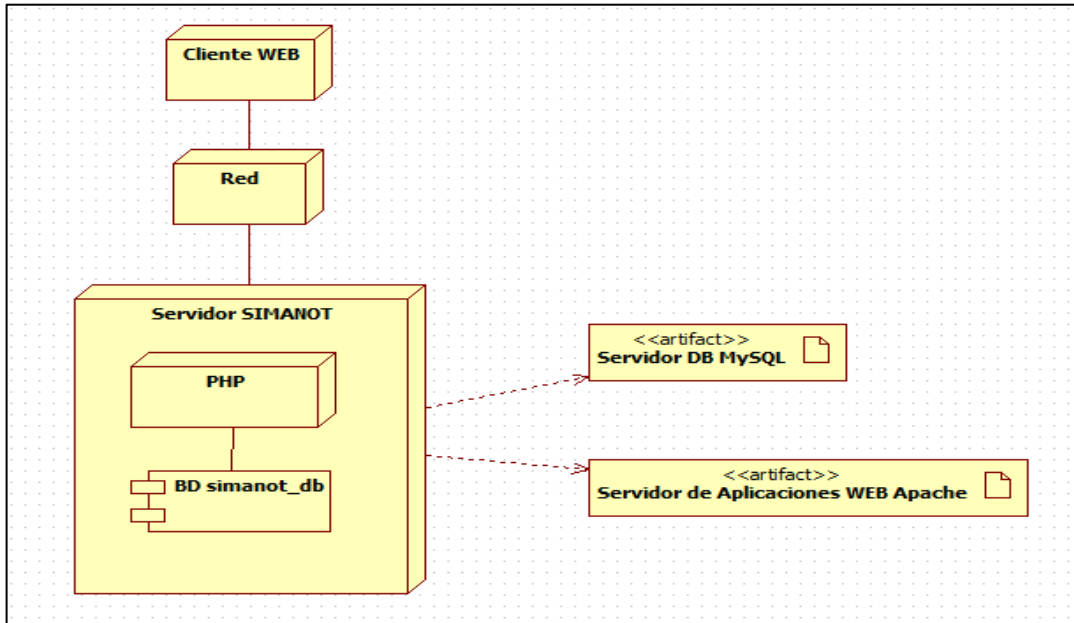


Figura 44-3: Diagrama de despliegue

Realizado por: Carlos Patiño, 2016, 2016

3.3.3 Implementación

Una vez concluida la etapa de desarrollo se procedió a la implementación del sistema SIMANOT en el Web Hosting contratado **EdamHost**.

Como primer paso se debe crear la base de datos del sistema, para ello debemos seguir los siguientes pasos.

Primero ingresamos a la cuenta de hosting y digitamos nuestros datos de usuario y contraseña.

Luego debemos crear nuestra base de datos, para ello seleccionamos la opción de **Base de Datos MYSQL**.

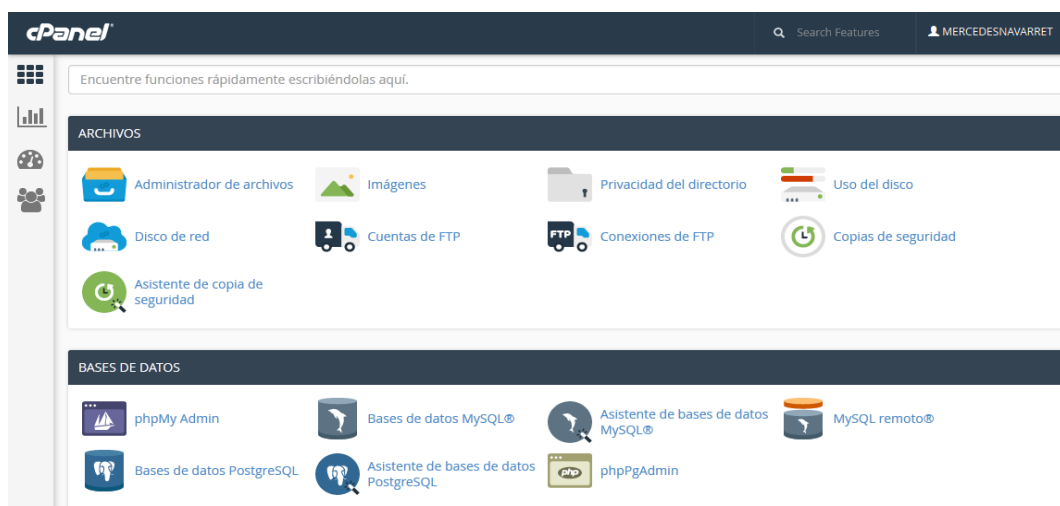


Figura 45-3: Implementación - Cpanel de usuario

Realizado por: Carlos Patiño, 2016

Escribimos el nombre de la base de datos que deseamos crear y pulsamos el botón **Crear una base de datos**.



Figura 46-3: Implementación - Creación de base de datos

Realizado por: Carlos Patiño, 2016

Seguidamente creamos un usuario para nuestra base de datos



Figura 47-3: Implementación - Creación de usuario de bd

Realizado por: Carlos Patiño, 2016

Una vez creada nuestra base de datos y nuestro usuario procedemos a restaurar nuestra base de datos para ello elegimos la opción **phpMyAdmin**

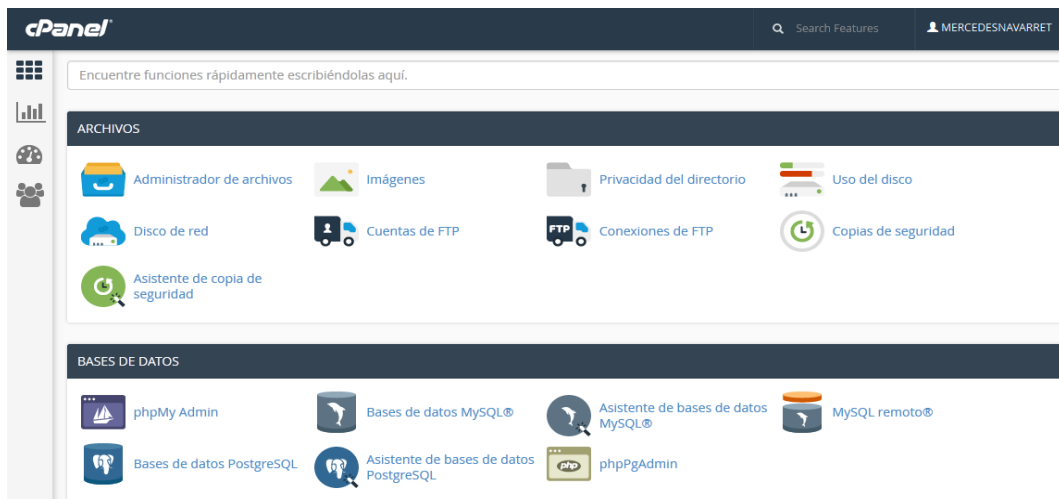


Figura 48-3: Implementación - Pantalla phpMyAdmin

Realizado por: Carlos Patiño, 2016

Seleccionamos nuestra base de datos y seleccionamos la opción Importar en donde debemos seleccionar el backup de la base de datos.

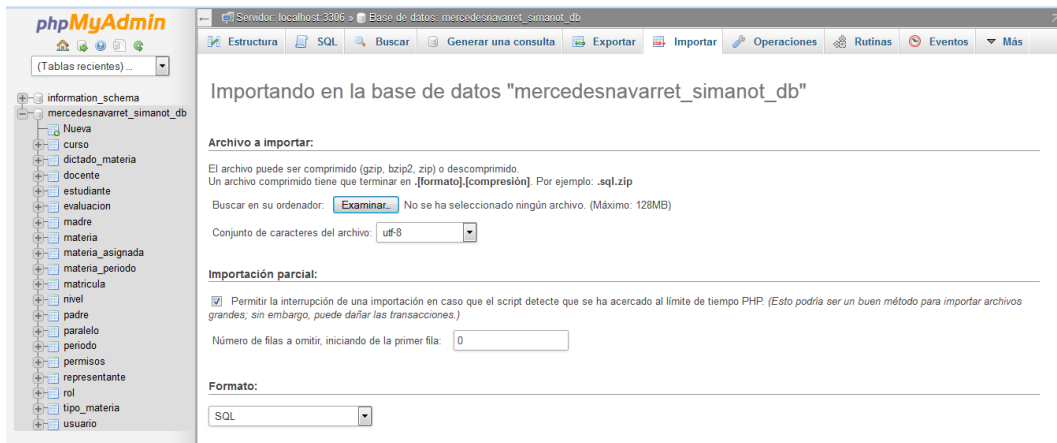


Figura 49-3: Implementación - Restauración de base de datos

Realizado por: Carlos Patiño, 2016

Como segundo paso debemos subir nuestro proyecto al hosting, para ello seleccionamos la opción **Administrador de archivos**

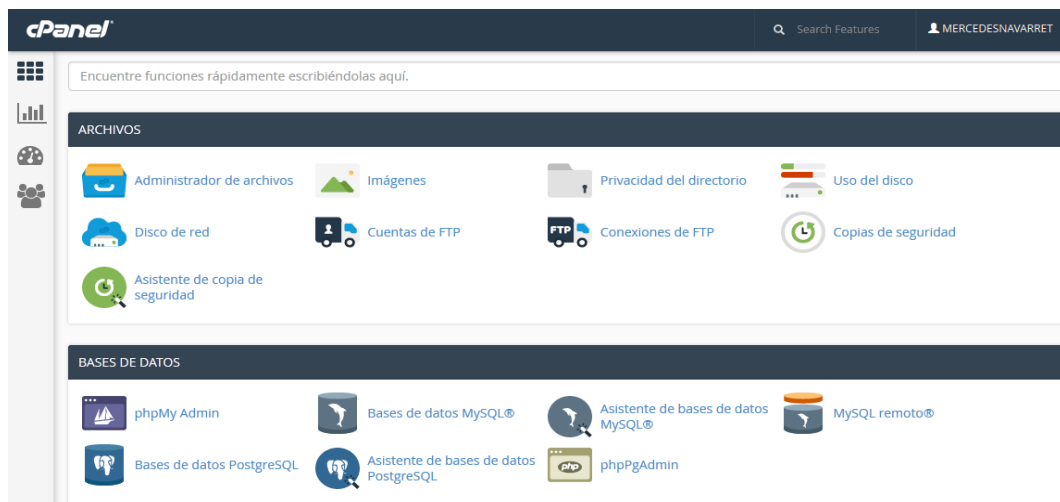


Figura 50-3: Implementación - Administrador de archivos

Realizado por: Carlos Patiño, 2016

En la ventana de administración de archivos seleccionamos la carpeta **Public_html** y presionamos la opción **Cargar** y subimos nuestras carpetas de la aplicación web desarrollada.

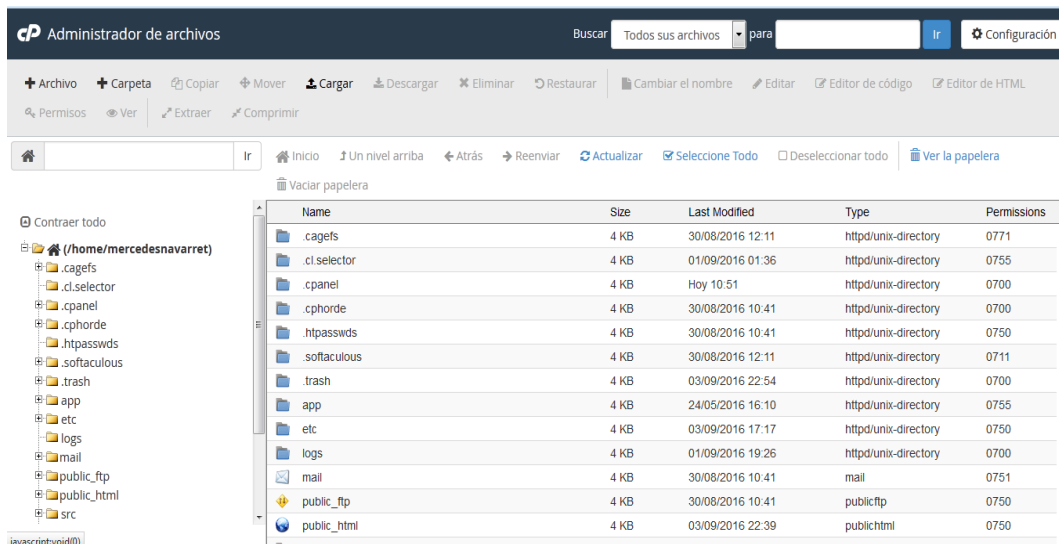


Figura 51-3: Implementación - Directorios del administrador de archivos

Realizado por: Carlos Patiño, 2016

En esta ventana seleccionamos las carpetas que vamos a subir para el correcto funcionamiento del sistema web.

Las carpetas que se deben subir son las carpetas: SRC, WEB, APP Y VENDOR.

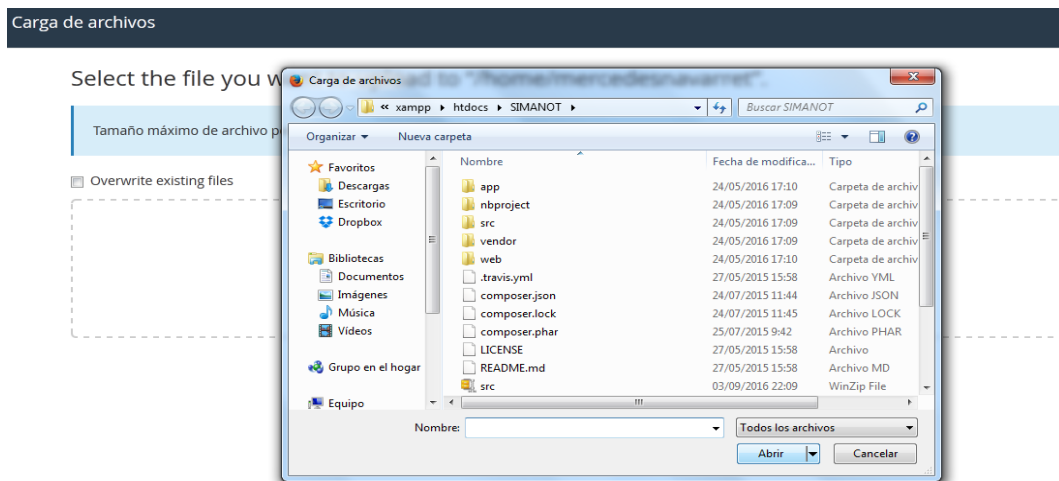


Figura 52-3: Implementación - Carga de archivos

Realizado por: Carlos Patiño, 2016

Una vez subido la aplicación al hosting podemos ingresar a ella mediante la dirección del dominio contratado.

3.3.4 Pruebas


El día 23 de agosto del 2016 se implementó el sistema de registro de Matrículas y Notas (SIMANOT) en el web hosting de los servidores de EDAMHOST, en el cual comenzó la etapa de pruebas del sistema.


Durante la etapa de pruebas se presentaron los siguientes inconvenientes.

- No se pudieron generar reportes en el sistema esto fue debido a que no se encontraba instalado el aplicativo wkhtmltopdf el cual nos permite visualizar los reportes en formato PDF. Por lo cual se optó por cambiar de bundle para la generación de reportes.
- No se guardaban los registros de asignación de materias a los docentes debido a la falta de configuración en los parámetros del sistema con la base de datos. El inconveniente fue solucionado volviendo a subir el archivo config.yml corregido de la aplicación al servidor.
- Se tuvo la necesidad rediseñar el formato de los reportes de las matrículas de los estudiantes debido a la nueva disposición de la Institución Educativa. En la que se procedió a cambiar el formato del reporte.

A continuación se presenta el plan de pruebas realizado a la directora y un docente de la Institución educativa

Plan de Pruebas

Fecha y Hora: 23/08/2016, 15:00		Evaluador: Docente: Sandra Calle	
Descripción: Evaluar las funciones del sistema SIMANOT			
Nombre del Caso de Prueba	Prerrequisitos	Resultados Esperados	Resultados Obtenidos
Registro de Notas	Ingreso de Datos: <ul style="list-style-type: none"> • Notas Parciales • Notas Exámenes 	Ingreso satisfactorio de notas en el sistema	Mensaje de confirmación de registro.
Generación de reportes	Selección de Reporte: <ul style="list-style-type: none"> • Notas Parciales • Promedios • Notas Curso 	Generación de reportes adecuados	Emisión de reportes satisfactorio
Firma evaluador:  Sandra Calle			

Fecha y Hora: 23/08/2016, 14:00		Evaluador: Directora: Sor María Heras	
Descripción: Evaluar las funciones del sistema SIMANOT			
Nombre del Caso de Prueba	Prerrequisitos	Resultados Esperados	Resultados Obtenidos
Registro de matriculas	Ingreso de Datos: <ul style="list-style-type: none"> • Nombre del Alumno • Año Lectivo • Curso • Observaciones 	Ingreso satisfactorio de datos de matrícula en el sistema	Mensaje de confirmación de registro.
Generación de reportes	Selección de Reporte: <ul style="list-style-type: none"> • Matriculas • Hoja de Vida 	Generación de reportes adecuados	Emisión de reportes satisfactorio
Firma evaluador:  Sor María Heras			

Presentación de resultados

Para la obtención de resultados se realizó una encuesta con el personal docente y administrativo de la Institución Educativa, la misma que tuvo como finalidad comparar los resultados obtenidos por el sistema en comparación con los con los procesos realizados manualmente en el proceso de registro de matrículas y notas.

Para obtener el número de docentes a aplicar la encuesta se realizó el cálculo estadístico para determinar el tamaño de la muestra de la población, para la parte administrativa la encuesta se realizó a todo el personal ya que la población es de 2 personas (directora y secretaria).

Fórmula estadística para determinar el tamaño de la muestra de la población:

$$n = \frac{N * Z\sigma^2 * p * q}{d^2(N - 1) + Z\sigma^2 * p * q}$$

Dónde:

n = Tamaño de la muestra.

N = Total de la población.

$Z\sigma$ = Valor obtenido mediante niveles de confianza. Es un valor constante que, si no se tiene su valor, se lo toma en relación al 95% de confianza equivale a 1,96 (como más usual).

p = proporción esperada, suele utilizarse un valor constante de 0,05.

q = 1 – p (en este caso 1-0.05 = 0.95)

d = precisión (normalmente se usa el 5%).

Datos:

N= 22

$\sigma = 0,5$.

Z = 95% = 1,96

e = 0,05

$$n = \frac{22 * 1,96^2 * 0.05 * 0.95}{0,05^2(22 - 1) + 1,96^2 * 0.05 * 0.95} = 16$$

Por lo que se toma una muestra de 16 personas para realizar la encuesta.

La siguiente tabla muestra los resultados obtenidos de la encuesta realizada al personal docente de la Institución en el proceso de registro de notas:

Datos obtenidos por los docentes:

Tabla 33-3: Datos docentes

D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	D16
25 min	35 min	35 min	25 min	25 min	25 min	35 min	25 min	25 min	35 min	35 min	25 min	25 min	35 min	25 min	25 min

Realizado por: Carlos Patiño, 2016

Para obtener el promedio del tiempo de los docentes en el registro de notas se empleó la fórmula estadística de la media aritmética.

Calculo Media Aritmética:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \dots + x_n}{n}$$

$$x = 25+35+35+25+25+25+35+25+25+35+35+25+25+35+25+25/16 = 29 \text{ minutos}$$

Datos obtenidos por el personal administrativo:

Tabla 34-3: Datos administrativos

Proceso	Directora	Secretaria
Registro de Matrículas	30 min	30 min
Generación de Reportes	30 min	30 min

Realizado por: Carlos Patiño, 2016

$$x_1 = 30+30/2 = 30 \text{ minutos (matrículas)}$$

$$x_2 = 30+30/2 = 30 \text{ minutos (notas)}$$

Tabla 35-3: Presentación resultados - Tiempos en procesos

Procesos	Proceso Manual	Sistema SIMANOT
Matrícula de estudiantes	30 minutos	5 minutos
Registro de notas	29 minutos	4 minutos
Generación de reportes	30 minutos	1 minutos

Realizado por: Carlos Patiño, 2016

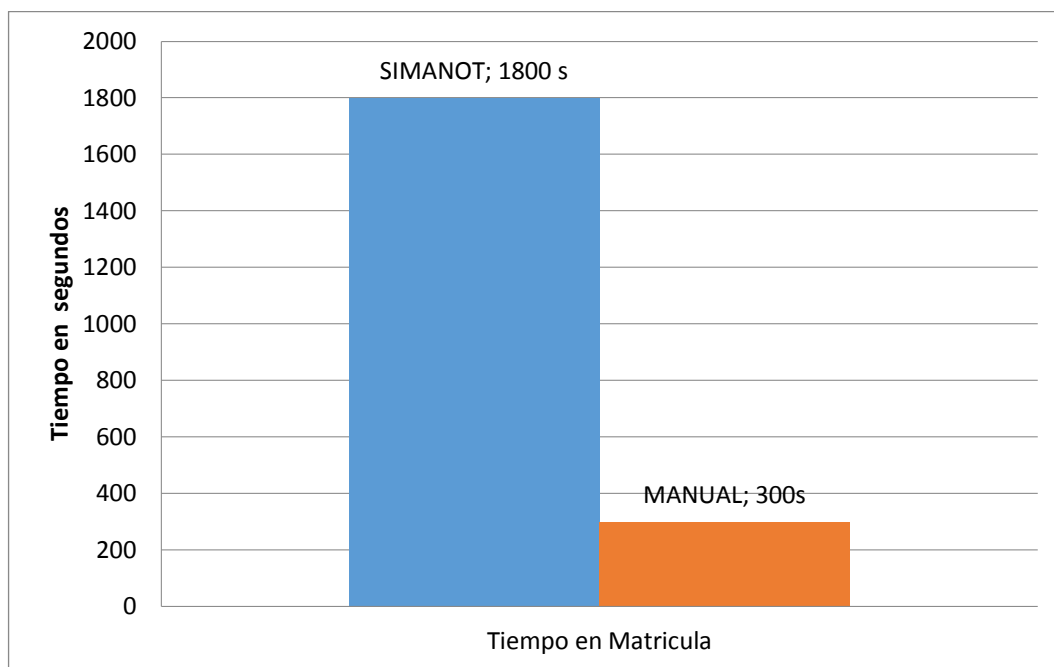


Figura 53-3: Presentación resultados - Tiempo en matrículas

Realizado por: Carlos Patiño, 2016

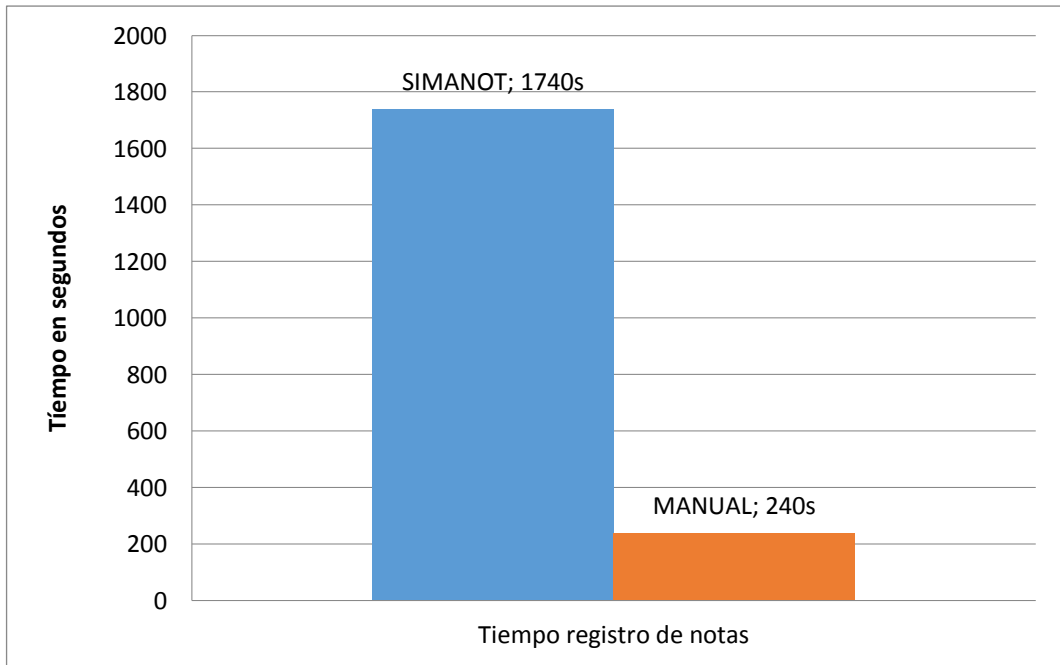


Figura 54-3: Presentación resultados - Tiempo en registro de notas

Realizado por: Carlos Patiño, 2016

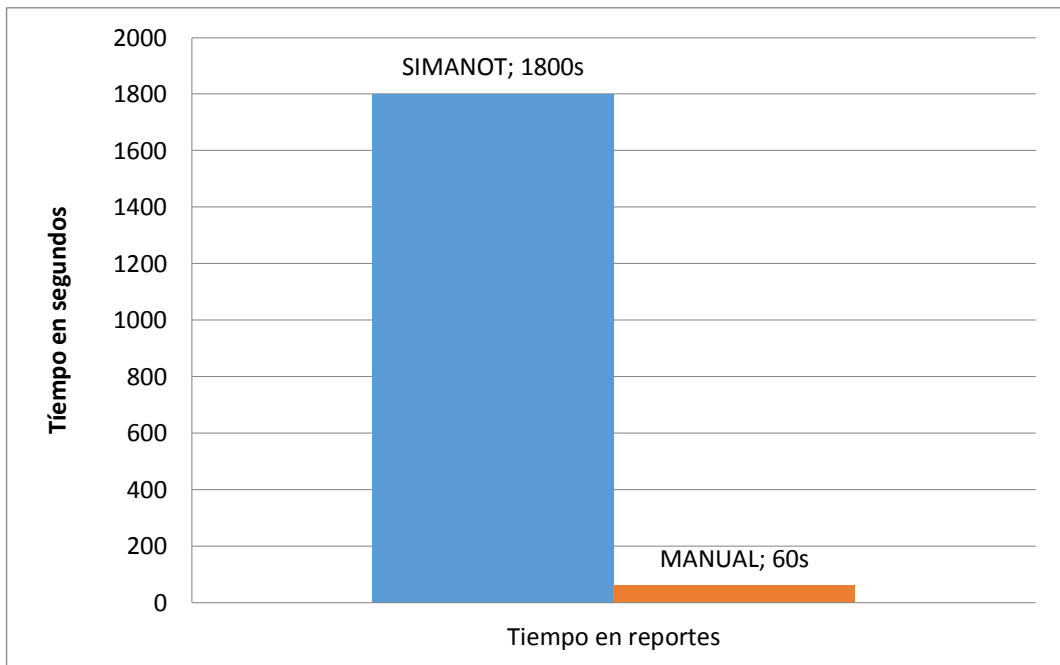


Figura 55-3: Presentación resultados - Tiempo en generación de reportes

Realizado por: Carlos Patiño, 2016

Tabla de suministros utilizados

Tabla 36-3: Presentación de resultados – Suministros utilizados

Actividades	Proceso Manual	Sistema SIMANOT
Matrícula de estudiantes	Suministro de oficina (esferos, lápiz, borrador)	Computador
Ingreso de notas	Suministro de oficina (esferos, lápiz, borrador)	Computador
Generación de reportes	Documentos/Computador/Impresora	Computador/Impresora

Realizado por: Carlos Patiño, 2016

Como podemos observar en la tabla anterior, con la implementación del sistema se ha logrado reducir el consumo de suministros de oficina y disminuyendo así gastos en la institución

Además se realizó la capacitación correspondiente tanto al personal docente como administrativo para el correcto manejo del sistema, la capacitación tuvo una duración de 4 horas y fue dictada en el laboratorio de computación de la institución (ANEXO A), también se realizó una encuesta a la Directora de la escuela (ANEXO B), para verificar que el sistema cumple con todos los requerimientos planteados, en el cual con mucha satisfacción dijo que el sistema cumple con todos los objetivos especificados, es fácil de utilizar y ayuda a optimizar los procesos realizados en la institución educativa..

CONCLUSIONES

- Mediante el análisis de los procesos de registro de matrículas y notas llevados a cabo manualmente en la Institución Educativa se determinó la necesidad de desarrollar un sistema informático que permita automatizar estos procesos y llevar un mejor control de estos.
- El método de Craig Larman facilitó el análisis de los requerimientos y el diseño de creación de la aplicación web
- La utilización del Framework Symfony facilitó el desarrollo de la aplicación, sus herramientas aceleran el proceso de desarrollo y creación de aplicaciones web rápidas y seguras.
- Con la implementación del sistema web SIMANOT se logró optimizar considerablemente los tiempos en los procesos de matriculación, registro de notas y emisión de reportes.
- En la fase de pruebas se verificó el cumplimiento de los requisitos del sistema con las necesidades de la institución.
- Al utilizar el estándar IEEE 830 para la especificación de requerimientos se obtuvo una mejor interacción con el personal de la institución educativa para la obtención de requisitos del sistema.
- Al trabajar con bases de datos, se garantiza la integridad de los datos permitiendo mayor confianza al momento de generar reportes.
- Para el desarrollo del sistema web se utilizaron herramientas open source para disminuir costos en el desarrollo del sistema.

RECOMENDACIONES

- La utilización de Frameworks para desarrollar aplicaciones es una excelente solución de programación ya que permiten agilizar y reducir tiempos en desarrollo, mantener una organización y reutilización de código de las aplicaciones.
- Se debe tomar en cuenta los complementos necesarios para la implementación del sistema y obtener así un correcto funcionamiento de la aplicación y evitar que se generen fallos en la aplicación.
- Se recomienda contratar un servicio de hosting que ofrezca la mayor disponibilidad del servicio para evitar inconveniente en la utilización del sistema.
- Se recomienda desarrollar un módulo de seguimiento académico, que puedan acceder y obtener información en cuanto a tareas, notas y su rendimiento estudiantil.

BIBLIOGRAFÍA

DELGADO, S; ET AL. BOBADILLA: *JavaScript*. Madrid-España:McGRAW-HILL.2001.pp. 39

GÓMEZ JIMÉNEZ, E. *Desarrollo de software con Netbeans 7.1*. España: S.A. Marcombo, 2012, pp. 1-4.

LARMAN, C. *UML y Patrones*. Mexico:PEARSON.2001

ORACLE. *Netbeans*. [en línea].[Consultado el 22 de julio de 2016].disponible en: https://netbeans.org/community/releases/61/index_es.html

POTENCIER, F; & ZANINOTTO, F. *Symfony 1.4, la guia definitiva*. [en línea]. [Consultado el 17 de julio de 2016]. disponible en http://librosweb.es/libro/symfony_1_4/capitulo_1/symfony_en_pocas_palabras.html

POTENCIER, Fabien; & WEAVER Ryan. *Symfony 2 - libro oficial*. [en línea]. [Consulta: 18 julio de 2016]. Disponible en: https://librosweb.es/libro/symfony_2_x/

QUIJADO, J. *Domine PHP y MySQL*.Madrid-España: Alfaomega.2007.pp.13

SÁNCHEZ, M. *Método de craig Larman*. [en línea]. [Consulta: 23 de Julio de 2016]. Disponible en: <http://ocw.uc3m.es/ingenieria-informatica/metodologia-de-desarrollo-visual/course-files/material-del-tema-9>

SEARCHDATACENTER. *Que es MySQL*. [en línea]. [Consultado el 23 de julio de 2016]. disponible en: <http://searchdatacenter.techtarget.com/es/definicion/MySQL>

SABANA, M. *PHP 5 con PostgreSQL 8*. Lima-Peru: Megabyte. 2006. pp.17-18

THAIRA. *Symfony beneficios e información*. [en línea]. [Consultado el 18 de julio de 2016].
disponible en: <http://www.thaira.net/blog/desarrolloweb/symfony-beneficios-e-informacion/>

VÁZQUEZ MARIÑO C. *Programación en PHP5*. Nivel Básico. España: 2008, pp. 9.

WIKIBOOKS. *Programación en PHP*. [en línea]. [Consultado el 18 de julio de 2016]
disponible en: https://es.wikibooks.org/wiki/Programaci%C3%B3n_en_PHP

ANEXOS

ANEXO A: Interfaces de usuario.

A continuación se muestran todas las interfaces de usuario del sistema SIMANOT

Interfaz del Administrador

Pantalla Principal

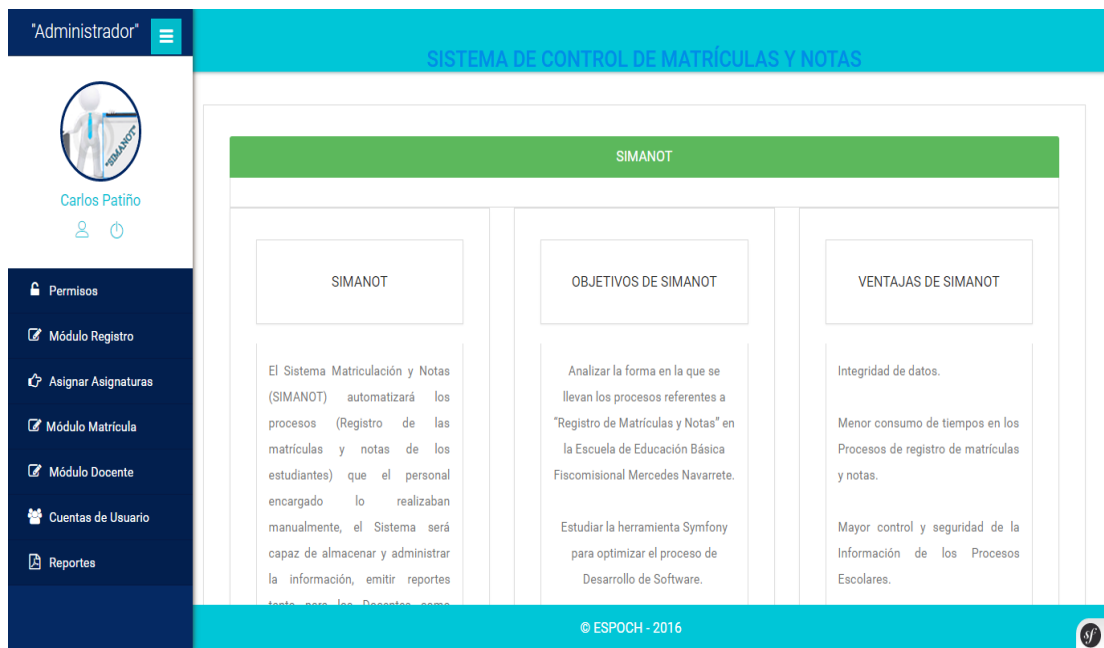


Figura 5657-3: Interfaz administrador - Pantalla principal

Realizado por: Carlos Patiño, 2016

Pantalla de asignación de permisos

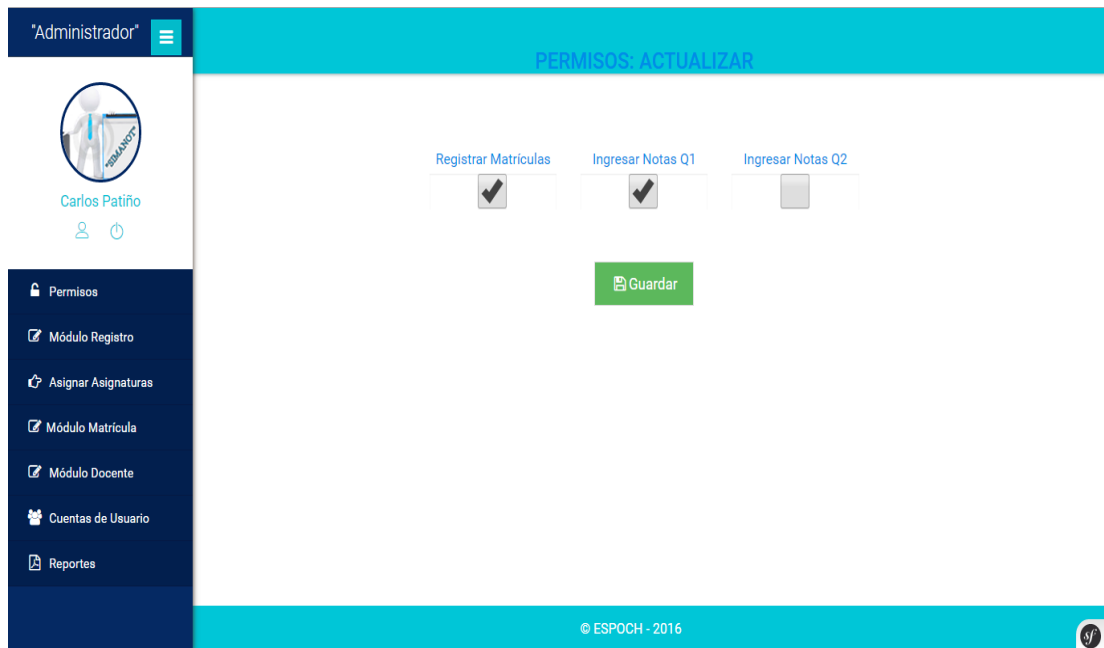


Figura 58-3: Interfaz administrador - Pantalla de asignación de permisos

Realizado por: Carlos Patiño, 2016

Pantalla de Gestión de Periodos Académicos

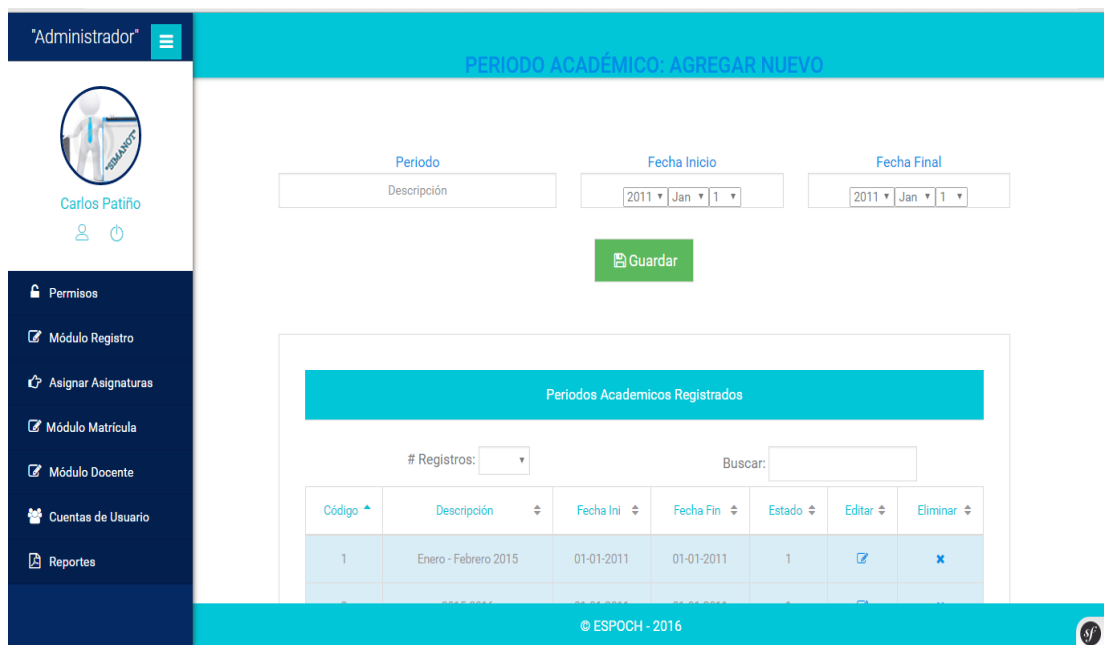


Figura 59-3: Interfaz administrador - Pantalla de gestión de periodos académicos

Realizado por: Carlos Patiño, 2016

Pantalla de Gestión de Asignaturas

Administrador

ASIGNATURA: AGREGAR NUEVA

Asignatura Tipo

Dato

Guardar

Asignaturas Registradas

Registros: Buscar:

Código	Descripción	Tipo	Estado	Editar	Eliminar
1	Matematicas	Obligatoria			
2	Fisica	Obligatoria			
3	Quimica	Obligatoria			

© ESPOCH - 2016

Figura 60-3: Interfaz administrador - Pantalla de gestión de asignaturas

Realizado por: Carlos Patiño, 2016

Pantalla de Gestión de Paralelos

Administrador

ASIGNATURA: AGREGAR NUEVA

Asignatura Tipo

Dato

Guardar

Asignaturas Registradas

Registros: Buscar:

Código	Descripción	Tipo	Estado	Editar	Eliminar
1	Matematicas	Obligatoria			
2	Fisica	Obligatoria			
3	Quimica	Obligatoria			

© ESPOCH - 2016

Figura 61-3: Interfaz administrador - Pantalla de gestión de paralelos

Realizado por: Carlos Patiño, 2016

Pantalla de Gestión de Niveles

Nivel

Dato

Guardar

Niveles Registrados

Registros: Buscar:

Código	Descripción	Editar	Eliminar
1	Primer		

Mostrando Registros: 1 al 1 del total de 1

Anterior 1 Siguiete

© ESPOCH - 2016

Figura 62-3: Interfaz administrador - Pantalla de gestión de niveles

Realizado por: Carlos Patiño, 2016

Pantalla de Gestión de Cursos

Nivel

Dato

Guardar

Niveles Registrados

Registros: Buscar:

Código	Descripción	Editar	Eliminar
1	Primer		

Mostrando Registros: 1 al 1 del total de 1

Anterior 1 Siguiete

© ESPOCH - 2016

Figura 63-3: Interfaz administrador - Pantalla de gestión de cursos

Realizado por: Carlos Patiño, 2016

Pantalla de Gestión de Asignaturas Disponibles

Administrador

NIVEL: AGREGAR NUEVO

Nivel

Dato

Guardar

Niveles Registrados

Registros: Buscar:

Código	Descripción	Editar	Eliminar
1	Primer		

Mostrando Registros: 1 al 1 del total de 1

Anterior 1 Siguiente

© ESPOCH - 2016

Figura 64-3: Interfaz administrador - Pantalla de gestión de asignaturas disponibles

Realizado por: Carlos Patiño, 2016

Pantalla de Asignación de Asignaturas

Administrador

MATERIA - DOCENTE: ASIGNAR

Periodo Académico

Docente

Materia

Curso

2015-2016

Jose Lopez

Primer A

Observación

Concuido / Si - No

Guardar

Asignaturas Asignadas al Docente

Registros: Buscar:

Periodo Académico	Docente	Asignatura	Curso	Editar	Eliminar
-------------------	---------	------------	-------	--------	----------

© ESPOCH - 2016

Figura 65-3: Interfaz administrador - Pantalla de asignación de asignaturas

Realizado por: Carlos Patiño, 2016

Pantalla de Registro de Estudiantes

"Administrador" ☰

DATOS DEL PADRE: AGREGAR

Nombres
Nombres

Apellidos
Apellidos

Cédula
140000000

Estado Civil
Descripción

Fecha Nacimiento
Año Mes Día

Nacionalidad
Descripción

Nivel Educación
Primaria.. Bachiller...

Profesión
Descripción

Dirección
Descripción

Teléfono
072741253

Vive con el Estudiante
Si o No

Representante
Si o No

Guardar

Datos Estudiante > **Madre**
Matricula **Padre**
Representante
Estudiante

Lista de Padres Registrados

© ESPOCH - 2016

Figura 66-3: Interfaz administrador - Pantalla de registro de estudiantes

Realizado por: Carlos Patiño, 2016

Pantalla de Registro de Matrícula

"Administrador" ☰

MATRÍCULA: AGREGAR NUEVA

Periodo Académico
2015-2016

Estudiante
pepe pepito

Curso
Primer A

Observaciones
Descripción

Guardar

Estudiantes Matriculados

Registros: ▾ Buscar:

Nº	Periodo Académico	Fecha Matricula	Estudiante	Curso	Editar	Eliminar
No existe información para mostrar						

Showing 0 to 0 of 0 entries

Anterior Siguiete

Registro Eliminado Correctamente

© ESPOCH - 2016

Figura 67-3: Interfaz administrador - Pantalla de registro de matrículas

Realizado por: Carlos Patiño, 2016

Pantalla de Registro de Notas

The screenshot shows the 'ASIGNATURAS - DOCENTES' page. The top navigation bar is blue with the text 'ASIGNATURAS - DOCENTES'. The sidebar menu on the left includes 'Permisos', 'Módulo Registro', 'Asignar Asignaturas', 'Módulo Matricula', 'Módulo Docente', 'Cuentas de Usuario', and 'Reportes'. The main content area features a 'Selecionar Asignatura' section with a '# Registros:' dropdown and a 'Buscar:' input field. Below this is a table with 5 rows and 3 columns: 'Nº', 'Docente', and 'Asignatura'. The table data is as follows:

Nº	Docente	Asignatura
1	Jose Lopez	Ciencias
2	Jose Lopez	Computacion
3	Jose Lopez	Fisica
4	Jose Lopez	Ingles
5	Jose Lopez	Matematicas

Below the table, it says 'Mostrando Registros: 1 al 5 del total de 8'. The footer of the page includes '© ESPOCH - 2016' and a URL: 'localhost/SIMANOT/web/app_dev.php/admin/lista/materias'.

Figura 68-3: Interfaz administrador - Pantalla de registro de notas

Realizado por: Carlos Patiño, 2016

Pantalla de Gestión de Usuarios

The screenshot shows the 'CUENTA DE USUARIO: AGREGAR NUEVO' page. The top navigation bar is blue with the text 'CUENTA DE USUARIO: AGREGAR NUEVO'. The sidebar menu on the left is the same as in the previous screenshot. The main content area features a form for adding a new user. The form fields are:

- Nombres:** Ingrese los Nombre
- Apellidos:** Ingrese los Apellidos
- Cédula:** Ingrese la Cédula
- Nick:** Ingrese el Nick
- Contraseña Nueva:** Contraseña
- Repetir contraseña:** Contraseña
- Rol:** ADMIN

Below the form is a 'Registrar' button. At the bottom of the page, there is a section titled 'Listado de Usuarios'. The footer of the page includes '© ESPOCH - 2016' and a URL: 'localhost/SIMANOT/web/app_dev.php/admin/usuario/registro'.

Figura 69-3: Interfaz administrador - Pantalla de gestión de usuarios

Realizado por: Carlos Patiño, 2016

Pantalla de Emisión de Reportes

Administrador

SELECCIONAR TIPO DE REPORTE

Carlos Patiño

Permisos

Módulo Registro

Asignar Asignaturas

Módulo Matricula

Módulo Docente

Cuentas de Usuario

Reportes

Lista de Reportes

Registros: [dropdown] Buscar: [input]

Reporte 1	Reporte 2	Reporte 3	Reporte 4	Reporte 5	Reporte 6	Reporte 7
Asignaturas - Docente	Estudiantes	Matriculas	Notas - Estudiante	Notas - Curso	Notas - Curso 2	Aprobados - Reprobados

Mostrando Registros: 1 al 1 del total de 1

Anterior 1 Siguiete

© ESPOCH - 2016

Figura 70-3: Interfaz administrador - Pantalla de emisión de reportes

Realizado por: Carlos Patiño, 2016

Interfaz de Usuario Secretaria

Pantalla de Registro de Estudiantes

SECRETARI@

DATOS DE LA MADRE: AGREGAR

secretaria secretaria

OPCIONES

Datos Estudiante

Matricula

Reportes

Madre

Padre

Representante

Estudiante

Nombres

Apellidos

Cédula

Estado Civil

Fecha Nacimiento

Nacionalidad

Nivel Educación

Profesión

Dirección

Teléfono

Vive con el Estudiante

Representante

Guardar

Lista de Madres Registradas

© ESPOCH - 2016

Figura 71-3: Interfaz secretaria - Pantalla de registro de estudiantes

Realizado por: Carlos Patiño, 2016

Pantalla de Registro de Matrícula

SECRETARI@

MATRÍCULA: AGREGAR NUEVA

Periodo Académico: 2015-2016

Estudiante: pepe pepito

Curso: Primer A

Observaciones: Descripción

Guardar

Estudiantes Matriculados

Registros: [dropdown] Buscar: [input]

N°	Periodo Académico	Fecha Matriculación	Estudiante	Curso	Observación	Editar
No existe información para mostrar						

Showing 0 to 0 of 0 entries

Anterior Siguiete

© ESPOCH - 2016

Figura 72-3: Interfaz secretaria - Pantalla de registro de matrículas

Realizado por: Carlos Patiño, 2016

Pantalla de Emisión de Reportes

SECRETARI@

PERIODO ACADÉMICO: 2015-2016

Lista de Estudiantes

Registros: [dropdown] Buscar: [input]

Código	Estudiante	Curso	Reporte
No existe información para mostrar			

Showing 0 to 0 of 0 entries

Anterior Siguiete

© ESPOCH - 2016

- OPCIONES
- Datos Estudiante
- Matrícula
- Reportes
- Estudiantes
- Matriculas
- Notas
- Estudiantes Aprobados

Figura 73-3: Interfaz secretaria - Pantalla de emisión de reportes

Realizado por: Carlos Patiño, 2016

Interfaz del Usuario Docente

Pantalla de Registro de Notas

"DOCENTE" [Menu Icon]

ASIGNATURA: Ciencias

Lista de Estudiantes [Refresh Icon]

Registros: [Dropdown] Buscar: [Input]

ESTUDIANTE	QUIMESTRE 1				QUIMESTRE 2				NOTAS	DATOS
	P1	P2	P3	EX	P1	P2	P3	EX		
No existe información para mostrar										
Showing 0 to 0 of 0 entries										

Anterior Siguiente

© ESPOCH - 2016

Figura 74-3: Interfaz docente- Pantalla de registro de notas

Realizado por: Carlos Patiño, 2016

Pantalla de Emisión de Reportes

"DOCENTE" [Menu Icon]

ASIGNATURA: Ciencias

Lista de Estudiantes [Refresh Icon]

Registros: [Dropdown] Buscar: [Input]

ESTUDIANTE	QUIMESTRE 1				QUIMESTRE 2				NOTAS	DATOS
	P1	P2	P3	EX	P1	P2	P3	EX		
No existe información para mostrar										
Showing 0 to 0 of 0 entries										

Anterior Siguiente

© ESPOCH - 2016

Figura 75-3: Interfaz docente - Pantalla de emisión de reportes

Realizado por: Carlos Patiño, 2016

ANEXO B: Evidencia de capacitación a usuarios.

Para el correcto uso del sistema se brindó la respectiva capacitación a los usuarios que laboran en la Institución Educativa



ANEXO C: Encuestas realizadas al personal administrativo y docente.

ENCUESTA SISTEMA DE REGISTRO DE MATRICULAS Y NOTAS

1. ¿El sistema de registro de matriculas y notas "SIMANOT" es de fácil utilización?
Sí No
2. ¿El sistema SIMANOT tiene interfaces intuitivas y amigables para los usuarios?
Sí No
3. ¿Cree que el sistema SIMANOT es de gran ayuda para realizar los procesos de matriculas y notas?
Sí No
4. ¿De forma manual que tiempo se tarda en registrar una matrícula?
10 minuto 20 minutos 30 minutos
5. ¿Con el sistema SIMANOT que tiempo se tarda en registrar una matrícula?
Menos de 1 minuto Más de 1 minuto
6. ¿De forma manual que tiempo se tarda en emitir un reporte?
10 minuto 20 minutos 30 minutos
7. ¿Con el sistema SIMANOT que tiempo se tarda en generar un reporte?
Menos de 1 minutos Más de 1 minutos
8. ¿Cree Ud. que con la implementación del sistema se tendrá un mejor control en el registro de matriculas y notas?
Sí No
9. ¿Qué le parece el sistema SIMANOT?
Malo Regular Bueno Excelente


Sor María Heras
DIRECTORA

ESCUELA E.B.F.
MERCEDES NAVARRETE
SUCÚA

ANEXO D: Certificado de capacitación a los usuarios.



REPÚBLICA DEL ECUADOR
DISTRITO 14D03 LOGROÑO – SUCUA EDUCACIÓN

**ESCUELA DE EDUCACIÓN BÁSICA FISCOMISIONAL MERCEDES NAVARRETE
SUCÚA**

Sucúa, 20 de julio del 2016

La suscrita Directora de la Escuela De Educación Básica Fiscomisional MERCEDES NAVARRETE, del cantón Sucúa, provincia Morona Santiago a petición verbal de la parte interesada.

CERTIFICA:

Que el Sr. Carlos Humberto Patiño Mora, con CI: 140046705-4 estudiante de la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO EXTENSIÓN MORONA SANTIAGO, ha brindado la capacitación correspondiente el día 17 de julio del presente año, sobre el correcto uso del sistema de registro de matriculación que será implementado en nuestra institución educativa.

Es cuanto puedo certificar en honor a la verdad, para fines legales pertinentes.


Sor María Heras
DIRECTORA

**ESCUELA E.B.F.
MERCEDES NAVARRETE
SUCÚA**

Dirección: Carlos Olson y María Troncatti
mnavarretesucua12@yahoo.com

Tel. 2740-156

ANEXO E: Certificado de implementación del sistema.



REPÚBLICA DEL ECUADOR
DISTRITO 14D03 LOGROÑO – SUCUA EDUCACIÓN

**ESCUELA DE EDUCACIÓN BÁSICA FISCOMISIONAL MERCEDES NAVARRETE
SUCÚA**


Sucúa, 25 de agosto del 2016

Yo Sor María Heras, Dir. De LA ESCUELA DE EDUCACIÓN BÁSICA FISCOMISIONAL MERCEDES NAVARRETE a petición verbal del interesado.

CERTIFICO

Que el Sr. Carlos Humberto Patiño Mora, con CI: 140046705-4 estudiante de la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO EXTENSIÓN MORONA SANTIAGO, ha desarrollado e implementado el sistema para registro de matrículas y notas para nuestra institución educativa, el cual cumple con todos los requisitos establecidos.

Es cuanto puedo certificar en honor a la verdad, para fines legales pertinentes.


Sor María Heras
DIRECTORA

**ESCUELA E.B.F.
MERCEDES NAVARRETE
SUCÚA**

