



INSTITUTO SUPERIOR MINERO METALÚRGICO DE MOA  
"Dr. Antonio Núñez Jiménez"  
Facultad Geología - Minas

---

# Trabajo de Diploma

para optar por el título de Ingeniero Informático

*SISTEMA DE GESTIÓN DOCENTE PARA LA CARRERA  
DE INFORMATICA EN EL ISMM*

Autor: *Yadiris Martínez Cordero*

Tutor: *Ing. Marcos A. Martínez Rodríguez*

MOA, 2012

"Año 54 de la Revolución"

**DECLARACIÓN DE AUTORÍA**

Declaro que soy el único autor de este trabajo y autorizo al Departamento de Informática del Instituto Superior Minero Metalúrgico de Moa para que hagan el uso que estimen pertinente con el mismo.

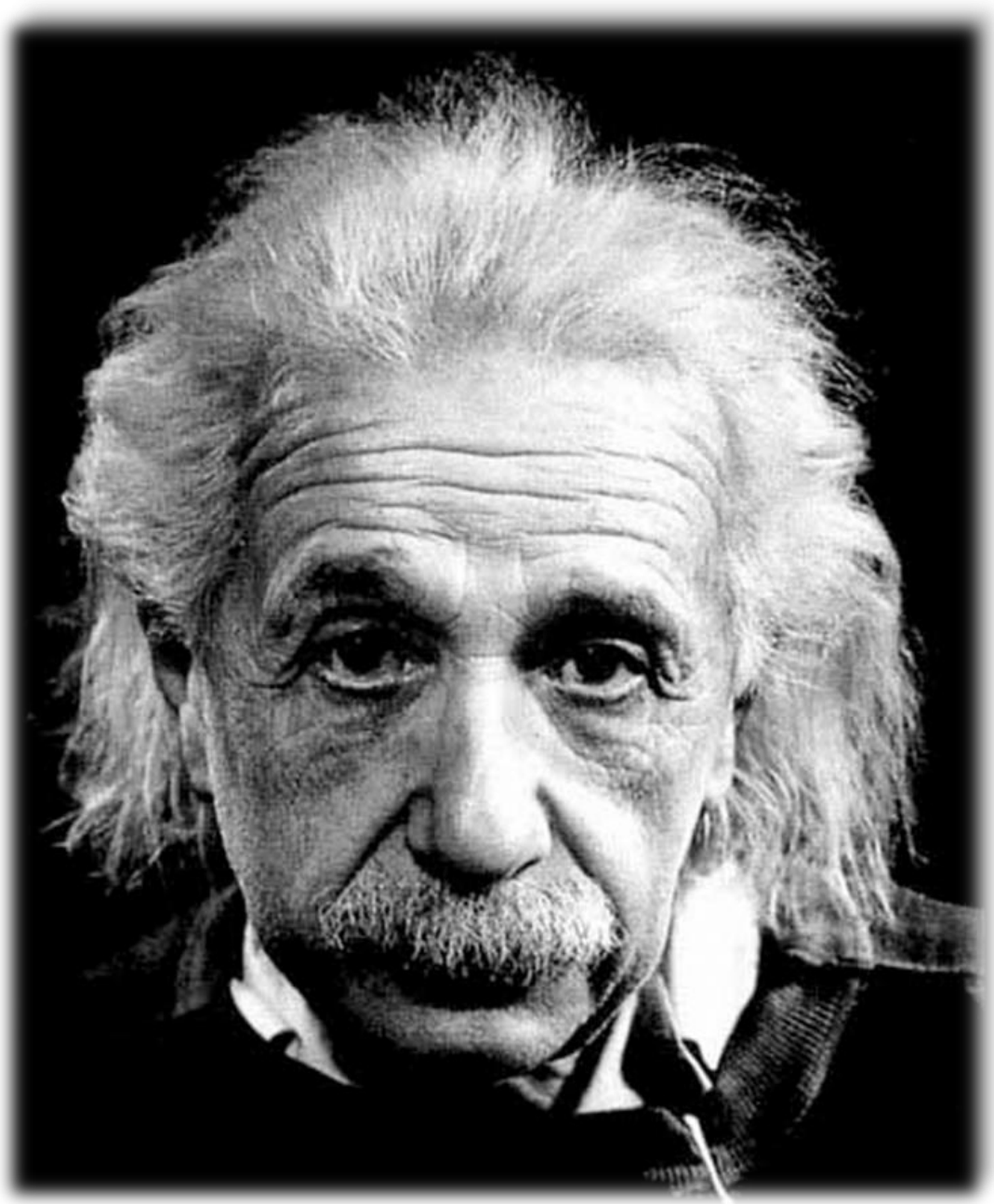
Para que así conste firmo la presente a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del 2012.

Yadiris Martínez Cordero

\_\_\_\_\_  
Firma del Autor

Ing. Marcos Martínez Rodríguez

\_\_\_\_\_  
Firma del Tutor



*“Hay una fuerza motriz más poderosa que el vapor,  
la electricidad y la energía atómica: la voluntad.”*

*Albert Einstein*

*Luego de un largo camino, al fin llego una gran meta que me he  
trazado,*

*Y por tal motivo mis logros se los quiero dedicar:*

*A Cristian Yuniel mi niño y mi vida,  
Quien me da fuerzas para enfrentar este mundo tan convulsionado.*

*A mis Padres Isnel y Nineyta, quienes siempre me han apoyado,  
Mucho más cuando atravesé momentos difíciles.*

*A Yuniel, mi esposo, compañero y amigo,  
Con quien he compartido gran parte de mi vida,  
En la tristeza y en la felicidad, en la salud y en la enfermedad,  
En la riqueza y en la pobreza.*

*A mi hermano, "mortificón" pero mi hermano,  
A mis abuelas y abuelos,  
Tíos y tías.*

*A Dios, que aunque no siga sus pasos desde una Iglesia lo llevo  
presente, y siempre he confiado en que con su mano creadora me  
guía por los buenos caminos de la vida.*

*A todos los que tuvieron fe en mí.*

*Primeramente quisiera agradecer a quienes me dieron la vida, mis padres, sin ellos nada de esto hubiera podido consumarse este trabajo.*

*A mi niño Cristian Yuniel, quien llegó a mi vida a darme alegría y amor, paz y felicidad, gracias a Dios que lo puso en mi camino.*

*A mi esposo Yuniel,  
Quien siempre me extendió su mano y con su ayuda logré realizar este sueño.*

*A toda mi familia...*

*A todo aquel que directa o indirectamente han influido en mi formación como persona y como profesional.*

*A mi tutor Marcos, quien estuvo presente en cada momento, tutor y amigo.*

*A mis profesores, donde unos más que otros han aportado su granito de arena.*

*A mis compañeros de "Info 07", tremendo piquete de años.*

*A todos, mis más grandes y sinceros agradecimientos.*

**Yadiris Martínez Cordero**

**RESUMEN**

Los sistemas gestores son de amplia variedad y por tanto se utilizan en disímiles procesos y funciones en el ámbito digital, permiten controlar con operaciones básicas de insertar, eliminar, modificar, etc., buscando siempre una mayor eficiencia y calidad en los servicios.

El departamento de Informática del ISMM se ha dado a la tarea de realizar su propio Sistema Gestor, el cual ayudaría y agilizaría los trabajos de rutinas diarias que se lleva a cabo con sus estudiantes, todo lo referente a notas, cortes evaluativos, así como su control, teléfonos, datos personales para su localización y otros.

El presente trabajo brinda a manera de propuesta una herramienta informática basada en la tecnología Web para un Sistema de Gestión Docente.

Es una herramienta concebida para los docentes con el objetivo de facilitar y complementar su labor diaria para con los alumnos y mejorar la comunicación directa y constante entre profesores, coordinadores de año y jefes de carrera.

**ABSTRACT**

The management systems are of great variety and are therefore used in dissimilar processes and functions in the digital environment, to control basic operations insert, delete, modify, etc., Always seeking greater efficiency and service quality.

The Informatics department of ISMM has been given the task of making their own Management System, which would help and speed up daily routines that work is carried out with students, all references to notes, evaluative courts and their control, telephones, personal data for location and other.

This paper provides a tool given way to computer-based Web technology for a Teaching Management System.

It is a tool designed for teachers in order to facilitate and supplement their daily work with students and improve the direct and constant communication between teachers, coordinators and heads of year career.

## INDICE DE CONTENIDO

ÍNDICE DE FIGURAS .....	XI
ÍNDICE DE TABLAS.....	XI
INTRODUCCIÓN .....	1
<b>CAPÍTULO 1: Fundamentación Teórica .....</b>	<b>5</b>
1.1 Introducción.....	5
1.2 Funcionamiento del Proceso Docente (PD) .....	5
1.3 Estrategia de la Gestión Docente.....	5
1.4 Estado del Arte.....	6
1.4.1 Antecedentes en el ISMM .....	7
1.5 Tendencias y Tecnologías actuales. ....	7
1.5.1 Tecnologías para el desarrollo de aplicaciones Web .....	7
1.5.2 Aplicaciones Web. Ventajas y Desventajas.....	8
1.5.2.1 ¿Qué denominamos como Páginas Web? .....	11
1.5.2.2 ¿Qué denominamos como Portal Web?.....	11
1.5.3 Lenguaje de Programación .....	12
1.5.3.1 ¿Por qué PHP v5.3.1?.....	16
1.5.4 Sistemas Gestores de Bases de Datos (SGBD) .....	17
1.5.4.1 ¿Por qué MySQL v5.1? .....	19
1.5.5 Herramientas CASE .....	20
1.5.6 Servidor para Aplicaciones Web: Apache v2.2.22.....	21
1.5.7 Entorno de Trabajo.....	22
1.5 Metodologías de desarrollo de Software .....	23
1.5.1 Metodologías tradicionales.....	24
1.5.2 Metodologías Ágiles .....	24
1.5.3 SCRUM .....	25
1.5.4 XP (Extreme Programming) .....	25
1.5.4.1 Valores de la metodología XP .....	26
1.5.4.2 Prácticas de la metodología XP.....	27
1.5.4.3 Fases de la metodología XP .....	31
1.5.5 ¿Por qué aplicar Scrum y XP? .....	33
1.5.5.1 Fases de la Metodología SXP .....	35
1.6 Arquitectura .....	36



1.6.1 Arquitectura en tres capas.....	36
CONCLUSIONES DEL CAPÍTULO 1 .....	38
2.2 Planificación y Definición del proyecto .....	39
2.2.1 Plantilla Concepción del Sistema .....	40
<b>CAPÍTULO 2: Planificación y Diseño .....</b>	<b>41</b>
2.1 Introducción.....	41
2.3 Lista de Reserva del Producto (LRP) .....	42
2.4 Plantilla Historias de Usuario.....	48
2.5 Planificación de entregas .....	49
2.5.1 Planificación de Iteraciones.....	49
2.5.3 Plan de duración de iteraciones .....	51
2.5.4 Valoración de Riesgos.....	52
2.6 Tarjetas CRC.....	55
CONCLUSIONES DEL CAPÍTULO 2.....	55
<b>CAPÍTULO 3: Implementación y Pruebas.....</b>	<b>56</b>
3.1 Introducción.....	56
3.2 Desarrollo del proyecto.....	56
3.3.2 Plantilla Tarea de ingeniería.....	56
3.3.3 Tareas de ingeniería por Historias de usuarios .....	56
3.3.3.1 Historias de Usuario abordadas en la primera iteración .....	60
3.3.3.2 Historias de Usuario abordadas en la segunda iteración .....	61
3.3.3.3 Historias de Usuario abordadas en la tercera iteración .....	61
3.3.3.4 Historias de Usuario abordadas en la cuarta iteración .....	62
3.3.3.5 Historias de Usuario abordadas en la quinta iteración .....	62
3.4 Pruebas.....	62
3.4.1 Desarrollo Dirigido por Pruebas .....	63
3.5 Objetivos de las pruebas .....	63
3.6 Plantilla Caso de prueba de aceptación .....	64
CONCLUSIONES DEL CAPÍTULO 3.....	64
<b>CAPÍTULO 4: Estudio de Factibilidad del Proyecto .....</b>	<b>65</b>
4.1 Introducción.....	65
4.2 Evaluación Costo-Beneficio.....	65
4.3 Efectos Económicos .....	66

4.3.1 Efectos directos .....	66
4.3.1.1 Positivos .....	66
4.3.1.2 Negativos .....	66
4.3.2 Efectos indirectos .....	67
4.3.3 Efectos externos.....	67
4.3.4 Intangibles .....	67
4.4 Beneficios y Costos Intangibles en el proyecto .....	68
4.4.1 Costos .....	68
4.4.2 Beneficios.....	68
4.5 Ficha de Costo .....	68
4.5.1 Costos en Moneda Librementemente Convertible .....	68
4.5.2 Costo en Moneda Nacional .....	69
CONCLUSIONES DEL CAPÍTULO 4 .....	71
<b>CONCLUSIONES GENERALES .....</b>	<b>72</b>
<b>RECOMENDACIONES .....</b>	<b>73</b>
<b>BIBLIOGRAFÍA .....</b>	<b>74</b>
<b>GLOSARIO DE TÉRMINOS .....</b>	<b>76</b>
<b>ANEXOS .....</b>	<b>77</b>
Anexo 1: Plantilla Concepción del Sistema .....	77
Anexo 2: Plantillas Historias de Usuario.....	79
Anexo 3: Tarjetas CRC.....	87
Anexo 4 Tarjetas de Tareas de Ingeniería o Programación .....	93
Anexo 5 Planillas Caso de Prueba de Aceptación .....	113
Anexo 6 Modelo Físico de Datos.....	134
Anexo 7 Principales Interfaces de la Aplicación .....	134

## **ÍNDICE DE FIGURAS**

Figura 1: Tecnologías para el desarrollo de aplicaciones Web. ....	8
Figura 2: Metodología SXP. ....	35
Figura 3: Arquitectura de tres niveles. ....	37
Figura 4: Gráfico relación solución manual y solución con el software. ....	71
Figura 5: Modelo Físico de Datos. ....	134
Figura 6: Portada de SIGEDO. ....	134
Figura 7: Login SIGEDO ....	135
Figura 8: Panel Administrador ....	135
Figura 9: Gestionar Facultad ....	136
Figura 10: Listado de Facultades ....	136
Figura 11: Modificar Facultades ....	136
Figura 12: Rol de Administrador ....	137
Figura 13: Establecer claustro. ....	137
Figura 14: Evaluación cuantitativa. ....	138

## **ÍNDICE DE TABLAS**

Tabla 1: Lista de Reserva del producto. ....	48
Tabla 2: Plan de duración de iteraciones. ....	52
Tabla 3: Lista de Riesgos ....	54
Tabla 4: Distribución de tareas por Historias de Usuario ....	60
Tabla 5: Planilla de Tareas de Ingeniería. ....	60
Tabla 6: HU de la Primera Iteración ....	61
Tabla 7: HU de la Segunda Iteración ....	61
Tabla 8: HU de la Tercera Iteración ....	61
Tabla 9: HU de la Cuarta Iteración ....	62

Tabla 10: HU de la Quinta Iteración .....	62
Tabla 11: Planilla Caso de Prueba de Aceptación. ....	64
Tabla 12: Costo en Moneda libremente convertible (CUC) .....	69
Tabla 13: Costo en Moneda Nacional .....	70
Tabla 14: Plantilla Concepción del Sistema. ....	79
Tabla 15: Modelo de Plantilla Historias de Usuario. ....	79
Tabla 16: HU No.1 Autenticar usuarios. ....	80
Tabla 17: HU No. 2 Gestionar Facultad. ....	80
Tabla 18: HU No. 3 Gestionar Carreras. ....	80
Tabla 19: HU No. 4 Gestionar Provincias.....	81
Tabla 20: HU No. 5 Gestionar Municipios. ....	81
Tabla 21: HU No. 6 Gestionar Profesores.....	82
Tabla 22: HU No. 7 Gestionar Roles. ....	82
Tabla 23: HU No. 8 Gestionar Estudiantes.....	82
Tabla 24: HU No. 9 Gestionar Asignaturas. ....	83
Tabla 25: HU No. 10 Gestionar Claustros. ....	83
Tabla 26: HU No. 11 Salvar Curso. ....	84
Tabla 27: HU No. 12 Cambiar Año.....	84
Tabla 28: HU No. 13 Gestionar Semestres. ....	84
Tabla 29: HU No. 14 Gestionar Cortes.....	85
Tabla 30: HU No. 15 Gestionar Evaluación.....	85
Tabla 31: HU No. 16 Exportar evaluaciones. ....	85
Tabla 32: HU No. 17 Exportar Listados.....	86
Tabla 33: HU No. 18 Gestionar Documentación. ....	86
Tabla 34: HU No. 19 Gestionar Informaciones.....	86
Tabla 35: HU No. 20 Realizar Búsquedas.....	87
Tabla 36: HU No. 21 Cambiar Contraseña.....	87

Tabla 37: Modelo tarjetas CRC. ....	87
Tabla 38: Tarjeta CRC No.1 Autenticar Usuarios. ....	87
Tabla 39: tarjeta CRC No.2 Cambiar Contraseña. ....	88
Tabla 40: Tarjeta CRC No. 3 Gestionar facultad. ....	88
Tabla 41: Tarjeta CRC No. 4 Gestionar carrera ....	88
Tabla 42: Tarjeta CRC No. 5 Gestionar Usuario. ....	88
Tabla 43: Tarjeta CRC No. 6 Gestionar Rol. ....	88
Tabla 44 Tarjeta CRC No. 7 Gestionar Estudiante.....	89
Tabla 45: Tarjeta CRC No. 8 Gestionar Asignatura. ....	89
Tabla 46: Tarjeta CRC No. 9 Gestionar Semestre. ....	89
Tabla 47: Tarjeta CRC No. 10 Gestionar cortes.....	90
Tabla 48: Tarjeta CRC No. 11 Gestionar Documentación.....	90
Tabla 49: Tarjeta CRC No. 12 Gestionar Evaluación. ....	90
Tabla 50: Tarjeta CRC No. 13 Reportar Evaluación.....	91
Tabla 51: Tarjeta CRC No.14 Exportar listado de estudiantes ....	91
Tabla 52: Tarjeta CRC No. 15 Buscar perfil de estudiante.....	91
Tabla 53: Tarjeta CRC No. 16 Gestionar Informaciones ....	91
Tabla 54: Tarjeta CRC No. 17 Salvar fin de Curso.....	92
Tabla 55: Tarjeta CRC No. 18 Cambiar de Año. ....	92
Tabla 56: Tarjeta CRC No. 19 Gestionar Provincia.....	92
Tabla 57: Tarjeta CRC No. 20 Gestionar Municipio. ....	92
Tabla 58: Tarjeta CRC No. 21 Gestionar Claustro ....	93
Tabla 59: Tarea de programación Autenticar usuarios.....	93
Tabla 60: Tarea de programación Insertar Facultad ....	93
Tabla 61: Tarea de programación Modificar facultad´ ....	94
Tabla 62: Tarea de programación Eliminar Facultad.....	94
Tabla 63: Tarea de programación listado de Facultades ....	94

Tabla 64: Tarea de programación Insertar Carrera .....	94
Tabla 65: Tarea de programación Modificar Carrera.....	95
Tabla 66: Tarea de programación Eliminar Carrera .....	95
Tabla 67: Tarea de programación Listado de carreras.....	95
Tabla 68: Tarea de programación Insertar Provincia .....	96
Tabla 69: Tarea de programación Modificar Provincia .....	96
Tabla 70: Tarea de programación Eliminar Provincia.....	96
Tabla 71: Tarea de programación Listado de Provincias .....	97
Tabla 72: Tarea de programación Insertar Municipio .....	97
Tabla 73: Tarea de programación Modificar Municipios.....	97
Tabla 74: Tarea de programación Eliminar Municipios .....	97
Tabla 75: Tarea de programación Listado de Municipios.....	98
Tabla 76: Tarea de programación Insertar Profesor.....	98
Tabla 77: Tarea de programación Modificar Profesor .....	98
Tabla 78: Tarea de programación Eliminar Profesor .....	99
Tabla 79: Tarea de programación Listado de Profesores .....	99
Tabla 80: Tarea de programación Insertar Coordinador .....	99
Tabla 81: Tarea de programación Modificar Coordinador .....	100
Tabla 82: Tarea de programación Eliminar Coordinador.....	100
Tabla 83: Tarea de programación Listado Coordinador .....	100
Tabla 84: Tarea de programación Insertar Administrador .....	100
Tabla 85: Tarea de programación Eliminar Administrador .....	101
Tabla 86: Tarea de programación Listado de Administradores .....	101
Tabla 87: Tarea de programación Insertar estudiantes.....	101
Tabla 88: Tarea de programación Modificar estudiantes .....	102
Tabla 89: Tarea de programación Eliminar estudiantes .....	102
Tabla 90: Tarea de programación Listado de estudiantes .....	102

Tabla 91: Tarea de programación Insertar asignatura .....	102
Tabla 92: Tarea de programación Modificar asignatura .....	103
Tabla 93: Tarea de programación Eliminar asignatura.....	103
Tabla 94: Tarea de programación Listado de asignaturas .....	103
Tabla 95: Tarea de programación Insertar claustro.....	104
Tabla 96: Tarea de programación Listado de claustros .....	104
Tabla 97: Tarea de programación Modificar claustro .....	104
Tabla 98: Tarea de programación Salvar fin de curso.....	104
Tabla 99: Tarea de programación Cambiar Año .....	105
Tabla 100: Tarea de programación Insertar semestre .....	105
Tabla 101: Tarea de programación Modificar semestre .....	105
Tabla 102: Tarea de programación Eliminar semestre.....	106
Tabla 103: Tarea de programación Listado de semestre .....	106
Tabla 104: Tarea de programación insertar fechas de cortes .....	106
Tabla 105: Tarea de programación Modificar fechas de cortes.....	106
Tabla 106: Tarea de programación Listado de fechas de cortes.....	107
Tabla 107: Tarea de programación Eliminar corte .....	107
Tabla 108: Tarea de programación Insertar evaluación .....	107
Tabla 109: Tarea de programación Modificar evaluación.....	108
Tabla 110: Tarea de programación Listado de evaluaciones .....	108
Tabla 111: Tarea de programación exportar evaluaciones .....	108
Tabla 112: Tarea de programación exportar listado de estudiantes.....	109
Tabla 113: Tarea de programación exportar listado de profesores .....	109
Tabla 114: Tarea de programación exportar listado de coordinadores .....	109
Tabla 115: Tarea de programación subir documentos .....	109
Tabla 116: Tarea de programación eliminar documentos .....	110
Tabla 117: Tarea de programación listado de documentos .....	110

Tabla 118: Tarea de programación insertar informaciones .....	110
Tabla 119: Tarea de programación modificar informaciones.....	111
Tabla 120: Tarea de programación eliminar informaciones .....	111
Tabla 121: Tarea de programación listado de informaciones.....	111
Tabla 122: Tarea de programación buscar estudiantes .....	111
Tabla 123: Tarea de programación buscar profesores.....	112
Tabla 124: Tarea de programación buscar coordinadores.....	112
Tabla 125: Tarea de programación buscar evaluaciones.....	112
Tabla 126: Tarea de programación cambiar contraseña .....	113
Tabla 127: PA Autenticar usuarios del sistema. ....	113
Tabla 128: PA cambiar contraseña .....	114
Tabla 129: PA Gestionar Facultad .....	115
Tabla 130: PA Gestionar Carrera .....	116
Tabla 131: PA Gestionar Provincia .....	117
Tabla 132: PA Gestionar Municipio .....	118
Tabla 133: PA Gestionar Profesor.....	119
Tabla 134: PA Gestionar Rol.....	120
Tabla 135: PA Gestionar Estudiante .....	121
Tabla 136: PA Gestionar Asignatura .....	122
Tabla 137: PA Gestionar Semestre.....	123
Tabla 138: PA Gestionar Corte .....	124
Tabla 139: PA Gestionar Documentación .....	125
Tabla 140: PA Gestionar Fechas de Cortes .....	126
Tabla 141: PA Gestionar Evaluación.....	127
Tabla 142: PA Exportar evaluación .....	127
Tabla 143: PA Exportar Listado de estudiantes .....	127
Tabla 144: PA Exportar Listado de profesores.....	128



Tabla 145: PA Exportar Listado de coordinadores .....	128
Tabla 146: PA Buscar Estudiantes .....	129
Tabla 147: PA Buscar Profesores .....	129
Tabla 148: PA Buscar Coordinadores .....	130
Tabla 149: PA Buscar Notas .....	131
Tabla 150: PA Gestionar Noticia .....	131
Tabla 151: PA Salvar Fin de Curso .....	132
Tabla 152: PA Cambiar Año .....	133
Tabla 153: PA Gestionar Claustro .....	134

## **INTRODUCCIÓN**

La Informática brinda un sin número de herramientas y tiene su uso y aprovechamiento en todos o casi todos los campos de acción imaginados, emergiendo como una de las nuevas tecnologías que más uso se le da en la actualidad. Su utilización, aprovechamiento y desarrollo es siempre bien recibida por todos. El mundo moderno tiene una gran dependencia de los sistemas informatizados, sistemas que controlan, gestionan, brindan información, etc.; pero que son de vital importancia para alcanzar el desarrollo científico-tecnológico, económico y hasta social en que se ve inmerso la sociedad.

En la actualidad el desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) han tomado un lugar cimero dentro del mundo desarrollado. No se puede hablar de progreso si no se tiene en cuenta el uso de la informática y las comunicaciones, por tanto, para lograr buenos resultados y eficiencia se hace necesario aprovechar las ventajas que estas nos brindan.

Cuba, país subdesarrollado y del tercer mundo, apuesta por automatizar procesos y darle un uso correcto a la informatización; es entonces que juegan un papel muy importante todo aquel que pueda aportar al desarrollo de las Tecnologías de la Informática y las Comunicaciones.

El municipio de Moa cuenta con un Instituto, centro universitario donde año tras año se promociona cierto número de Ingenieros Informáticos que realizan trabajos de diploma dándole solución, en la mayoría de los casos, a problemas reales del territorio y del propio centro docente.

El departamento de Informática del Instituto Superior Minero Metalúrgico de Moa, ISMM Dr. Antonio Núñez Jiménez en este momento no cuenta con un sistema que gestione la información docente de sus estudiantes, por lo que su trabajo no es del todo eficiente, por tal motivo dicho departamento se vio en la necesidad de informatizar el control y gestión docente a través de una herramienta informática que informatice el proceso.

Debido a lo antes planteado se tiene la siguiente **situación problemática**: La gestión docente de la carrera se hace prácticamente de forma manual ya que no se cuenta con una herramienta que facilite este servicio, los coordinadores de año no entregan y/o reciben informaciones en tiempo. Existe el Sistema de Gestión de la Nueva Universidad (**SIGENU**) sistema solo para la secretaría docente por lo que no permite el acceso de otro personal que lo necesite.

Debido a esta situación se define como **problema científico**: ¿Cómo favorecer la gestión docente de la carrera de informática del ISMM de Moa?, para darle solución al problema el **objeto de estudio** se define como los Sistema de Gestión Docente. Este objeto nos delimita como **campo de acción**: El Sistemas de Gestión Docente de la carrera de Informática del ISMM.

De manera que el **objetivo general** sería: Desarrollar una aplicación Web que permita realizar la gestión docente de la carrera de Informática en el ISMM.

Como **idea a defender** se tiene que: Si se desarrolla una aplicación Web que brinde a sus datos, confiabilidad, integridad y se caracterice por acceder de manera óptima a la información en una interfaz amigable para el usuario que la consulte, entonces favorecerá una adecuada gestión del proceso docente de los estudiantes de la carrera de Informática en el ISMM.

Como **objetivos específicos**:

1. Analizar el proceso docente actual.
2. Diseñar el sistema gestor del proceso docente.
3. Implementar el sistema gestor del proceso docente.
4. Analizar la factibilidad del proyecto.
5. Documentar el Software.

Para darle cumplimiento al objetivo general, se propusieron las siguientes **tareas**:

1.1 Entrevistar a profesores y coordinadores de la carrera de Informática para una mejor comprensión del proceso docente actual.

1.2 Analizar procesos actuales.

1.3 Elaborar el marco teórico metodológico que sustenta a la investigación.

2.1 Seleccionar la metodología de desarrollo de software, lenguajes de programación y herramientas que se utilizarán.

2.2 Diseñar la base de Datos.

3.1 Implementar el software.

4.1 Estudiar la factibilidad.

5.1 Realizar pruebas y documentación del sistema.

5.2 Desarrollar el manual de usuario.

Para cumplimentar estos objetivos se emplearon los **métodos teóricos y empíricos** de la investigación científica.

Entre los **métodos empíricos** usados se puede citar la **entrevista y el análisis de documentos** para la recopilación de la información. Mediante el análisis de la documentación se supo cuál era la funcionalidad de los mecanismos actuales del proceso de gestión docente, la entrevista nos permitió conocer en detalles como era el trabajo docente en el departamento y además para determinar los requerimientos que debe de cumplir el sistema a desarrollar.

Los métodos **teóricos** proporcionan calidad en la investigación. En el desarrollo del proceso de investigación se usaron el **análisis y síntesis** para la recopilación y el procesamiento de la información obtenida en los métodos empíricos y arribar a las conclusiones de la investigación. El **hipotético-deductivo** se utilizó en la elaboración de la idea a defender. **Modelación**, pues se crean abstracciones que explican la realidad, por ejemplo todos los modelos y diagramas presentados y el método **Histórico-Lógico**: este método es usado para comprender el antecedente del objeto de estudio y obtener su esencia, así

como la necesidad de su desarrollo en una forma superior, así como en el análisis de la forma en que se llevaba el control y la gestión docente en el departamento.

El presente trabajo consta de introducción, 4 capítulos, conclusiones, recomendaciones, bibliografía, glosario de términos y anexos.

**Capítulo 1 “Fundamentación Teórica”:** Se analizan aspectos relacionados con la Gestión del Proceso Docente, conceptualización y características, se realiza un estudio acerca de los diferentes sistemas existentes vinculados al campo de acción, además se presenta la metodología y las herramientas a utilizar en el desarrollo del sistema propuesto.

**Capítulo 2 “Planeación y Diseño”:** Se hace uso de la metodología expuesta en el capítulo inicial para el desarrollo del proyecto, abordando en detalles cada una de sus fases.

**Capítulo 3 “Desarrollo y Pruebas”:** Se presentan los principales métodos y definiciones dentro de la implementación de los flujos de trabajo. Se escriben además las pruebas realizadas y sus resultados.

**Capítulo 4 “Estudio de Factibilidad y Sostenibilidad”:** Se hace referencia a los beneficios tangibles e intangibles y se analizan los costos de desarrollo de la aplicación contra los beneficio para definir si es factible o no la aplicación y se valora la sostenibilidad del producto.

## **CAPÍTULO 1: Fundamentación Teórica**

### **1.1 Introducción**

No puede existir desarrollo en un mundo tan evolucionado sin la aplicación y utilización de las Tecnologías, la Informática y las Telecomunicaciones, y junto a ellas la automatización de sistemas. En este capítulo se abordarán conceptos asociados al problema y que son necesarios para un mejor entendimiento y darle solución al mismo. Además se realiza un estudio de procesos de gestión docente y la estructura que debe tener con sus funciones asociadas, algunos sistemas automatizados existentes, así como las definiciones básicas relacionadas con las aplicaciones Web, las tecnologías para su desarrollo, herramientas y lenguaje de programación a utilizar, aparte de la metodología a seguir en el desarrollo de la aplicación.

### **1.2 Funcionamiento del Proceso Docente (PD)**

Al tomar como punto de partida el enfoque holístico de la enseñanza resulta válido referirnos al Proceso Docente Educativo (PDE) como algo más que la integración de la enseñanza y el aprendizaje, más que cada uno de sus componentes, es la integración holística y sistémica de todos ellos junto con las cualidades, niveles de asimilación, de profundidad y estructural, en sus tres dimensiones: educativa, instructiva y desarrolladora.

El PDE es la integración, la sistematización de todos los aspectos en una unidad teórica totalizadora, se desarrolla en un movimiento propio en que se manifiestan todos los componentes, sus relaciones o leyes, sus cualidades y resultados. Este tiene su esencia con las leyes estudiadas e implica que en la didáctica ley y contradicción son una misma cosa y son la causa y la fuente del desarrollo del PDE, siendo la contradicción fundamental, la relación que se establece entre el objetivo y el método. [1]

### **1.3 Estrategia de la Gestión Docente**

La Gestión Docente conlleva a una eficaz optimización de las tareas y labores diarias de cualquier proceso educativo.

La **misión** de esta actividad es poder contar automáticamente y de forma digital con las últimas actualizaciones en las distintas materias docentes, cortes, evaluaciones, etc.

La **visión** consolida un buen desarrollo del proceso educativo, agilidad para el mismo además de integrar al claustro de profesores.

## **1.4 Estado del Arte**

Con el agilizad avance de la Informática se ha venido automatizando todos los procesos que una vez se hacían de forma manual, por tal motivo el departamento de informática no se ha quedado atrás y ha decidido implementar un sistema que permita la gestión de los procesos docentes en la carrera. En el mundo, incluyendo nuestro país y nuestro centro se pueden apreciar sistemas de esta índole tales como la versión WEB de la aplicación del SGD (Sistema de Gestión Docente) de los centro educativos de España, que permite acceder a los horarios, faltas y notas de cualquiera de ellos, además de permitir la comunicación entre padres, profesores y alumnos desde casa, sin necesidad de tener que ir al centro, no lo utilizamos ya que está diseñado con las especificaciones y necesidades de los centros educativos para los cuales fue concebido, así como para un país donde existe una gran conexión a Internet para la interacción padres-profesores. El Sistema de Gestión de la Nueva Universidad (SIGENU) es un ejemplo de sistemas nacionales, solo que está disponible única y exclusivamente para la secretaría del centro, así como en la Universidad de las Ciencias Informática el AKADEMOS, software diseñado para un centro muy peculiar como lo es la UCI, se manejan distintos datos, planes de estudio, etc. El sistema que se pretende realizar está enfocado entre otras cosas a los cortes docentes, uno de los controles que los coordinadores de año tienen que llevar y estos sistemas no lo toman en cuenta, por otro lado se quiere uno propio del departamento de Informática del ISMM, SIGEDO, Sistema de Gestión Docente debe cubrir todas las necesidades y expectativas que se esperan de él.

### **1.4.1 Antecedentes en el ISMM**

En la actualidad existen sistemas con similares propósitos, e incluso contamos en el ISMM con un Software destinado a la gestión docente, **SIGENU**, solo que no está a disposición de la carrera, es única y exclusivamente utilizado por la secretaría docente.

El proyecto SIGENU surge en Junio de 2004 a solicitud de la dirección del Ministerio de Educación Superior de Cuba (MES), como requisito a las necesidades de informatización de los procesos fundamentales de la gestión académica de una Institución de Educación Superior (IES) en todas sus modalidades de estudio. El sistema está compuesto por varios módulos que gestionan la información de un estudiante desde que se matricula hasta que se gradúa o causa baja definitiva.

## **1.5 Tendencias y Tecnologías actuales.**

A través de la historia, las tecnologías han sido portadoras de muchos de los grandes avances del hombre, avances que nunca serán suficientes, pues ante un nuevo descubrimiento surgen infinidades de interrogantes, por tal motivo las tecnologías brindan cada día nuevas herramientas y funcionalidades y por ende se le exige a los desarrolladores de software nuevos retos y más difícil se vuelve satisfacer las ambiciones de los clientes. Por tal motivo no les queda otra alternativa que ser innovadores y aplicar nuevas ideas capaces de revolucionar el complejo mundo en que se desenvuelven, el mundo de los ceros y los unos.

### **1.5.1 Tecnologías para el desarrollo de aplicaciones Web**

La característica dinámica de una aplicación Web se la proporciona la tecnología sobre la que está soportada. La mayoría de las tecnologías utilizadas para el desarrollo poseen características comunes:

- Permiten, a través de un lenguaje de programación, la creación de funcionalidades lógicas de aplicación, en el servidor.



- El código asociado a una página, o servicio Web, es ejecutado en el servidor (PHP, ASP, Java, etc.) o en el cliente (JavaScript, VBScript), permitiendo el envío/recibo de información entre el servidor y la máquina cliente a través de un documento con formato estándar (HTML, XML, etc.).

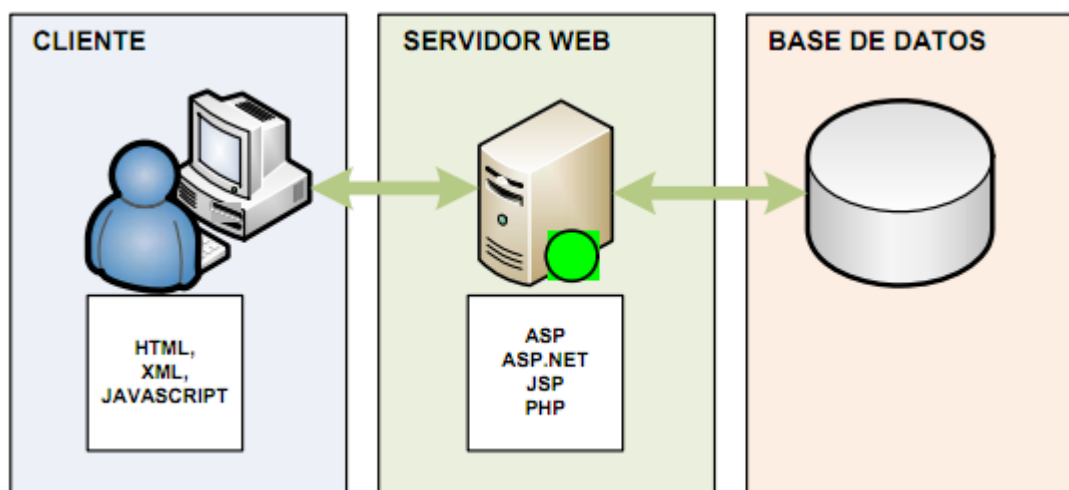


Figura 1: Tecnologías para el desarrollo de aplicaciones Web.

### 1.5.2 Aplicaciones Web. Ventajas y Desventajas.

Las aplicaciones Web son sistemas informáticos, usados por los usuarios para acceder a un servidor Web, a través de Internet o Intranet, y obtener así la información que se encuentra almacenada en él. Las aplicaciones Web deben su popularidad a la practicidad del navegador Web como cliente ligero, además a la facilidad que poseen para actualizar y mantener aplicaciones Web sin distribuir e instalar software en miles de potenciales clientes.

Las aplicaciones Web son muy utilizadas hoy en día debido a las **ventajas** que las mismas ofrecen. [4]

Algunas de las cuales se ven reflejadas a continuación:

**Compatibilidad multiplataforma:** Tienen un camino mucho más sencillo para la compatibilidad con múltiples plataformas que las aplicaciones de escritorio. Varias tecnologías incluyendo Java, Flash, ASP y Ajax permiten un desarrollo efectivo de programas soportando todos los sistemas operativos principales.

**Actualización:** Las aplicaciones basadas en Web están siempre actualizadas con la última versión, sin necesidad de llamar la atención del usuario o interferir con sus hábitos de trabajo, pues no se hace necesario iniciar nuevas descargas y/o procedimientos de instalación.

**Inmediatez de acceso:** Las aplicaciones basadas en Web no necesitan ser descargadas, instaladas y configuradas. Usted accede a su cuenta online y están listas para trabajar sin importar cuál es su configuración o su hardware.

**Menos requerimientos de memoria:** Las aplicaciones basadas en Web tienen menos demandas de memoria RAM de parte del usuario final que los programas instalados localmente. Al residir y correr en los servidores del proveedor, esas aplicaciones basadas en Web usan en muchos casos la memoria de las computadoras donde ellas corren, dejando más espacio para correr múltiples aplicaciones sin incurrir en frustrantes deterioros en el rendimiento.

**Menos Bugs:** Las aplicaciones basadas en Web deberían ser menos propensas a colgarse y crear problemas técnicos debido a software o conflictos de hardware con otras aplicaciones existentes, protocolos o software personal interno. Con aplicaciones basadas en Web, todos utilizan la misma versión, y todos los bugs pueden ser corregidos tan pronto como son descubiertos.

**Precio:** Las aplicaciones basadas en Web no requieren la infraestructura de distribución, soporte técnico y marketing requerido por el software descargable tradicional. Esto permite que las aplicaciones online cuesten una fracción de sus contrapartes descargables y no totalmente gratuitas, mientras que ofrecen componentes adicionales y servicios Premium como una opción.

**Los datos también van online:** El hecho de que el manejo de los datos sea realizado de forma remota libra al usuario de la responsabilidad en la protección de los mismos, y al mismo tiempo logra que los recursos sean accesibles en cualquier momento.

**Múltiples usuarios concurrentes:** Las aplicaciones basadas en Web pueden ser utilizadas por múltiples usuarios al mismo tiempo. No hay más necesidad

de compartir pantallas o enviar instantáneas cuando múltiples usuarios pueden ver e incluso editar el mismo documento de manera conjunta.

**Los datos son más seguros:** Si bien la ruptura de discos no va a desaparecer, es probable que los usuarios escuchen mucho menos del tema. A medida que las compañías se hagan cargo del almacenamiento de los datos del usuario, granjas de almacenamiento de datos redundantes, altamente fiables, los usuarios van a tener mucho menos riesgo de perder sus datos debido a una ruptura de disco impredecible o a un virus de la computadora. Las compañías que proveen aplicaciones basadas en Web van a brindar amplios servicios de resguardo de datos ya sea como una parte integral del servicio básico o como una opción paga.

**Desarrollar aplicaciones en el lenguaje que usted quiera:** Una vez que las aplicaciones han sido separadas de computadoras locales y sistemas operativos específicos, pueden también ser escritas en prácticamente cualquier lenguaje de programación. Debido a que las aplicaciones Web son esencialmente una colección de programas más que un simple programa, ellas podrían ser escritas en cualquier lenguaje de programación existente.

### **Desventajas Web [5]**

- Acceso limitado, la necesidad de conexión permanente y rápida a Internet hacen que el acceso a estas aplicaciones no esté al alcance de todos.
- La interactividad no se produce en tiempo real, en las aplicaciones Web cada acción del usuario conlleva un tiempo de espera hasta que se obtiene la reacción del sistema.
- Elementos de interacción muy limitados. En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones Web (mediante formularios principalmente) son muy escasas.

- Diferencias de presentación entre plataformas y navegadores. La falta de estándares ampliamente soportados dificulta el desarrollo de las aplicaciones.

### **1.5.2.1 ¿Qué denominamos como Páginas Web?**

Una página Web es un documento que contiene información sobre un tema específico y la cual es almacenada en un servidor remoto.

Una página Web es una fuente de información adaptada para la World Wide Web (WWW) y accesible mediante un navegador de Internet. Esta información se presenta generalmente en formato HTML y puede contener hiperenlaces a otras páginas Web, constituyendo la red enlazada de la World Wide Web.

Las páginas Web pueden ser cargadas de un ordenador o computador local o remoto, llamado Servidor Web, el cual servirá de HOST. El servidor Web puede restringir las páginas a una red privada, por ejemplo, una Intranet, o puede publicar las páginas en el World Wide Web. Las páginas Web son solicitadas y transferidas de los servidores usando el Protocolo de Transferencia de Hipertexto (HTTP - Hypertext Transfer Protocol). La acción del Servidor HOST de guardar la página Web, se denomina "HOSTING".

Las páginas Web pueden consistir en archivos de texto estático, o se pueden leer una serie de archivos con código que instruya al servidor como construir el HTML para cada página que es solicitada, a esto se le conoce como Página Web Dinámica.

### **1.5.2.2 ¿Qué denominamos como Portal Web?**

Un portal Web, es un conjunto de páginas Web que ofrecen información, herramientas y/o servicios a sus usuarios, de esta manera se le brinda la facilidad de poder encontrar en dicho sitio todas sus necesidades sin salir de dicho portal.

A las páginas de un portal Web se accede desde una URL raíz común llamada portada, que normalmente reside en el mismo servidor físico. Las URLs organizan las páginas en una jerarquía, aunque los hiperenlaces entre ellas

controlan como el lector percibe la estructura general y como el tráfico Web fluye entre las diferentes partes de los sitios.

Algunos sitios Web requieren una suscripción para acceder a algunos o todos sus contenidos. Ejemplos de sitios con suscripción incluyen muchos sitios de noticias, sitios de juegos, foros, servicios de correo electrónico basados en Web y sitios que proporcionan datos de bolsa en tiempo real.

### **1.5.3 Lenguaje de Programación**

Un lenguaje de programación es un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. [2]

#### **Lenguajes del Lado del Servidor**

##### **PHP**

PHP (Personal Home Page) es el acrónimo de Hypertext Preprocessor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación.

Es también un lenguaje interpretado y embebido en el HTML. Fue creado originalmente en 1994 por Rasmus Lerdorf, pero como PHP está desarrollado en política de código abierto, a lo largo de su historia ha tenido muchas contribuciones de otros desarrolladores. Actualmente PHP se encuentra en su versión 5, que utiliza el motor Zend, desarrollado con mayor meditación para cubrir las necesidades de las aplicaciones Web actuales. PHP es un lenguaje de programación de estilo clásico, esto quiere decir que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones, etc. No es un lenguaje de marcas como podría ser HTML, XML o WML.

A diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso permite acceder a los recursos que tenga el servidor, como por ejemplo podría ser, una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página WML.

PHP es la gran tendencia en el mundo de Internet. Últimamente se puede observar un ascenso imparable, puesto que cada día son muchísimas más las páginas Web que lo utilizan para su funcionamiento, según las estadísticas, PHP se utiliza en más de 10 millones de páginas, y cada mes realiza un aumento del 15%.

Como síntesis, PHP corre en 7 plataformas, funciona en 11 tipos de servidores, ofrece soporte sobre unas 20 Bases de Datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de Aplicaciones Web muy robustas, y contiene unas 40 extensiones estables sin contar las que se están experimentando, también tiene soporte para comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (en Windows) y muchos otros.

Además de que:

- Es software libre, lo que implica menos costes y servidores más baratos que otras alternativas.
- Es muy rápido. Su integración con la base de datos MySQL y el servidor Apache, le permite constituirse como una de las alternativas más atractivas del mercado.
- Su sintaxis está inspirada en C, ligeramente modificada para adaptarlo al entorno en el que trabaja, de modo que si se está familiarizado con esta sintaxis, resultara muy fácil aprender PHP.
- Su librería estándar es realmente amplia, lo que permite reducir los llamados "costes ocultos", uno de los principales defectos de ASP.
- PHP tiene una de las comunidades más grandes en Internet, esto permite encontrar fácilmente ayuda, documentación, artículos, noticias, y otros recursos.
- Permite las técnicas de Programación Orientada a Objetos.

- Posibilita crear los formularios para la Web.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

## **ASP**

ASP (Active Server Pages) es la tecnología desarrollada por Microsoft para la creación de páginas dinámicas del servidor ASP, se escribe en la misma página Web utilizando el lenguaje Visual Basic Script o Jscript (Javascript de Microsoft). Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente. El cliente solamente recibe una página con el código HTML resultante de la ejecución de la página ASP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores.

**[6]**

## **Perl**

Es un lenguaje de programación muy utilizado para construir aplicaciones CGI para el Web. Perl es un acrónimo de Practical Extracting and Reporting Language, que viene a indicar que se trata de un lenguaje de programación muy práctico para extraer información de archivos de texto y generar informes a partir del contenido de los ficheros. Es un lenguaje libre de uso, eso quiere decir que es gratuito. Antes estaba muy asociado a la plataforma Unix, pero en la actualidad está disponible en otros sistemas operativos como Windows. Perl es un lenguaje de programación interpretado, al igual que muchos otros lenguajes de Internet como JavaScript o ASP. **[2]**

## **JSP**

Es un acrónimo de Java Server Pages, que en castellano vendría a decir algo como Páginas de Servidor Java. Es pues, una tecnología orientada a crear páginas Web con programación en Java. Con JSP podemos crear aplicaciones Web que se ejecuten en variados servidores Web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para

programar scripts de servidor en sintaxis Java. Por tanto, las JSP podremos escribirlas con nuestro editor HTML/XML habitual. [6]

## **Lenguajes del lado del Cliente**

### **Javascript**

Es un lenguaje de programación interpretado que permite a los desarrolladores crear acciones en sus páginas Web. Es utilizado para crear pequeños programas que luego son insertados en una página Web y en programas más grandes, orientados a objetos pero mucho más complejos. (HyperText Markup Language). [6]

### **HTML (HyperText Markup Languaje)**

El Lenguaje de Marcas de Hipertexto es el lenguaje de marcado predominante para la construcción de páginas Web, usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. PHTML es una extensión para un tipo de páginas Web que llevan código PHP y HTML para ser generadas. Cuando una página está escrita en PHP podemos encontrarla con varios tipos de extensiones como por ejemplo .php, .php4, .php5. [6]

### **Ventajas**

- Sencillo que permite describir hipertexto.
- Texto presentado de forma estructurada y agradable.
- No necesita de grandes conocimientos cuando se cuenta con un editor de páginas Web o WYSIWYG.
- Archivos pequeños.
- Despliegue rápido.
- Lenguaje de fácil aprendizaje.
- Lo admiten todos los exploradores.



## **Desventajas**

- Lenguaje estático.
- La interpretación de cada navegador puede ser diferente.
- Guarda muchas etiquetas que pueden convertirse en “basura” y
- dificultan la corrección.
- El diseño es más lento.
- Las etiquetas son muy limitadas. [3]

### **1.5.3.1 ¿Por qué PHP v5.3.1?**

Luego de hacer el análisis se decide utilizar el PHP embebido en el código HTML, PHP por el lado del servidor y HTML y JavaScript por el lado del cliente debido que:

- Está soportado en la mayoría de las plataformas de Sistemas operativos.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos llamados **ext's** o extensiones.
- Posee una amplia documentación en su página oficial.
- Permite las técnicas de programación orientada a objetos.
- El PHP no tiene costo oculto, o sea que cuando se adquiere incluye un sinnúmero de bibliotecas que proporcionan el soporte para la mayoría de las aplicaciones Web, por ejemplo e-mail, generación de ficheros PDF y otros.

- En caso de que no se tengan las bibliotecas estas se pueden encontrar gratis en Internet.
- PHP es más rápido.

#### **1.5.4 Sistemas Gestores de Bases de Datos (SGBD)**

Un Sistema de Gestión de Bases de Datos (SGBD) se puede considerar como la interfaz entre el usuario y los datos físicos. Se compone de un lenguaje de definición de datos, un lenguaje de manipulación de datos y de un lenguaje de consulta.

Algunos ejemplos de SGBD son PostgreSQL, MySQL, SQL Server, etc.

##### **PostgreSQL**

PostgreSQL es un sistema de gestión de base de datos relacional orientada a objetos de software libre, comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California. Ingres fue más tarde desarrollado comercialmente por la Relational Technologies/Ingres Corporation. En 1986 otro equipo dirigido por Michael Stonebraker de Berkeley continuó el desarrollo del código de Ingres para crear un sistema de bases de datos objeto-relacionales llamado Postgres. En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgres fue renombrado a PostgreSQL, tras un breve periplo como Postgres95. Se pueden crear nuevos tipos de datos, hacer herencias entre objetos. PostgreSQL es una herramienta muy potente para los desarrolladores de sistemas de bases de datos, tiene todo aquello de lo que carece MySQL. Al igual que otros proyectos, PostgreSQL no está controlado por una compañía específica, sino que cuenta con una comunidad global de desarrolladores y compañías para su desarrollo, esto posibilita que cada vez aumente sus funcionalidades. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto. Está ampliamente considerado como el sistema de bases de datos de código abierto

más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre. [8]

## **MySQL**

Es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. Software libre que se desarrolla en un esquema de licenciamiento dual. Está desarrollado en su mayor parte en ANSI C. Al contrario de proyectos como el Apache, donde el software es desarrollado por una comunidad pública, y el copyright del código está en poder del autor individual, MySQL está poseído y patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet.

Tras la compra de MySQL por parte de **Sun Microsystems**, el gestor de bases de datos de código abierto pasó a manos de Sun. Siete meses después **Oracle Corporation**, propietarios del gestor de bases de datos relacional Oracle, ha comprado Sun por 7.400 millones de dólares. El control de MySQL por parte de Oracle puede ser muy perjudicial para el futuro desarrollo de MySQL, que está en competencia directa con Oracle en aplicaciones de tamaño medio. No obstante las versiones libres (anterior a v5.1) han demostrado ser eficientes y preferidas por delante de muchos SGBD. [9]

### **Principales Características:**

- El principal objetivo de MySQL es velocidad y robustez.
- Escrito en C y C++.
- Clientes C, C++, JAVA, Perl, TCL.
- Multiproceso, es decir puede usar varias CPU si éstas están disponibles.
- Puede trabajar en distintas plataformas y S.O. distintos.

- Sistema de contraseñas y privilegios muy flexibles y seguros.
- Todos los datos están grabados en formato ISO8859\_1.
- Los clientes usan TCP o UNIX Socket para conectarse al servidor.
- El servidor soporta mensajes de error en distintas lenguas.
- Todos los comandos tienen -help o - Para las ayudas. [2]

### **Oracle Database**

Es un SGBD objeto relacional, fabricado por Oracle Corporation. Se considera a Oracle como uno de los sistemas de bases de datos más completos, destacando su: soporte de transacciones, estabilidad, escalabilidad. Es orientado a objetos y también es multiplataforma, estas características lo convierten en uno de los SGBD más caros del mercado con licencia de software de tipo propietario.

### **SQL**

El Lenguaje de consulta estructurado (SQL Structured Query Language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo lanzar consultas con el fin de recuperar, de una forma sencilla, información de interés de una base de datos, así como también hacer cambios sobre la misma. Es un lenguaje de cuarta generación (4GL).

Es un lenguaje declarativo de alto nivel o de no procedimiento, que gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros, y no a registros individuales, permite una alta productividad en codificación.

#### **1.5.4.1 ¿Por qué MySQL v5.1?**

A pesar de caer en manos privadas se decidió optar MySQL v5.0, (última versión libre) ya que MySQL es muy rápido, fiable y fácil de usar, surge para manipular bases de datos muy grandes. Es un sistema multiplataforma de base

de datos relacionales, lo que da velocidad y flexibilidad, cuenta con un sistema de privilegios contraseñas muy seguro que permite la autenticación básica para el acceso al servidor. MySQL es un sistema de administración de base de datos. Opera en una arquitectura cliente/servidor. Ha sido el sistema gestor de bases de datos "Open-Source" más popular, además que cualquiera puede estudiar su código y comprenderlo fácilmente. Luego de analizadas las características y facilidades del SGBD presentado, y la de la herramienta a desarrollar se decide usar el MySQL como SGBD, por las siguientes razones:

- El principal objetivo de MySQL es velocidad y robustez.
- Escrito en C y C++.
- Clientes C, C++, JAVA, Perl, TCL.
- Multiproceso, es decir puede usar varias CPU si éstas están disponibles.
- Puede trabajar en distintas plataformas y Sistemas Operativos distintos.
- Sistema de contraseñas y privilegios muy flexibles y seguros.
- Utilidad (Isamchk) para chequear, optimizar y reparar tablas.
- Todos los datos están grabados en formato ISO8859\_1.
- Los clientes usan TCP o UNIX Socket para conectarse al servidor.
- El servidor soporta mensajes de error en distintas lenguas.
- Todos los comandos tienen -help o - Para las ayudas.
- El PHP maneja más fácil al MySQL, debido a la gran cantidad de funciones que tiene explícitas.

### **1.5.5 Herramientas CASE**

- **Embarcadero ERStudio 8.0:** Es una herramienta profesional para diseñar bases de datos, facilitándote herramientas para la creación de diagramas de relación, modelado de datos y gestión de estructuras (Es

compatible con ambos SGBD, MySQL y PostgreSQL), fue la seleccionada para el diseño de la base de datos de SIGEDO.

- **MYSQL Workbench 5.1 OSS:** Es una herramienta de modelado de datos, se usa para el diseño y la construcción lógica y física de bases de datos. Su ambiente es de gran alcance y multinivel. Se diseña para hacer más fácil de entender el estado actual de los datos. Simple y fácil al usuario, ayuda a organizaciones para tomar decisiones en cómo resolver embotellamientos de los datos, elimina redundancia.

Existen otras como es el caso del CaseStudio y Rational Rose, el primero solo para relacionar entidades para la base de datos y el segundo además de esto es una herramienta que se basa especialmente en la realización de diagramas.

### **1.5.6 Servidor para Aplicaciones Web: Apache v2.2.22**

Un servidor Web es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. El servidor Web se encarga de contestar a estas peticiones de forma adecuada, entregando como resultado una página Web o información de todo tipo de acuerdo a los comandos solicitados. De cualquier modo, siempre devolverá algún tipo de resultado HTML (hypertext markup language) al cliente o navegador que realizó la petición.

Algunos servidores, no sólo atienden y sirven peticiones HTTP y HTTPS, sino que pueden servir también peticiones basadas en otros protocolos o en protocolos implementados sobre HTTP.

**Apache**, es un servidor de protocolo para la transferencia de hipertextos (Hypertext Transfer Protocol, HTTP por sus siglas en inglés) de software libre para plataformas Unix, Windows, y Macintosh, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual.

Las características que lo definen son las siguientes:

- Multiplataforma.

- Es un servidor de Web conforme al protocolo HTTP/1.1.
- Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.
- Basado en hebras en la versión 2.0.
- Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos.
- Se desarrolla de forma abierta. [2]

### **1.5.7 Entorno de Trabajo**

#### **Macromedia Dreamweaver 8**

Es una aplicación en forma de estudio (basada en la forma de estudio de Adobe Flash) enfocada en la construcción y edición de sitios y aplicaciones Web basadas en estándares. Creado inicialmente por Macromedia (actualmente producido por Adobe Systems). Es el programa de este tipo más utilizado en el sector del diseño y la programación Web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y, recientemente, por su soporte de los estándares del World Wide Web Consortium.

Sin lugar a dudas, una de las herramientas más utilizadas por los webmasters que trabajan con aplicaciones visuales es Dreamweaver. El programa se adapta increíblemente a las necesidades de todo tipo de profesional de diseño Web, tanto para los que prefieren programar el código directamente en el editor de texto como para los que les gusta una metodología de trabajo totalmente visual.

#### **PhpDesigner 2008 v6.0.0**

Es un completo entorno de desarrollo y programación especialmente diseñado para los gurús de PHP, aunque también permite trabajar con comodidad en otros lenguajes de programación como HTML, XHTML, CSS y SQL.

Ofrece toda una serie de asistentes y diálogos integrados que facilitan en todo momento las tareas, además de acceso directo a librerías de código o scripts de uso habitual, utilidades diversas y toda suerte de herramientas, todo ello en una interfaz de diseño sencillo y elegante que se puede personalizar con nada menos que dieciocho temas distintos.

Cuenta con cliente de FTP y navegador de ficheros integrado, utilidades de corrección y autocompletado, búsqueda integrada en Google y soporte para proyectos, además de usar un práctico esquema de color para la sintaxis del código fuente que facilita enormemente la programación.

PhpDesigner soporta: PHP, HTML, XHTML, CSS, Java, Perl, JavaScript, VB, C# y SQL. [11]

## **1.5 Metodologías de desarrollo de Software**

Modelar es diseñar aplicaciones de software antes de codificarlas y es esencial tanto para los proyectos grandes como para los pequeños. Usando un modelo, los responsables del éxito de un proyecto de desarrollo de software pueden asegurar que la funcionalidad está corregida y completa; que se satisfacen las necesidades del usuario final y que el plan de la programación apoya los requisitos de escalabilidad, robustez, seguridad, expansión, además de otras características necesarias; antes de que su equipo empiece a codificar, ya que cuando la programación está hecha realizar cambios es más caro y difícil.

Una metodología de desarrollo de software es un conjunto de técnicas, herramientas, procedimientos y soporte documental que permite a los desarrolladores definir los elementos necesarios para la construcción de un nuevo producto de software. Es la que durante el proceso de desarrollo del software define “quién está haciendo qué, cuándo y cómo para alcanzar un determinado objetivo.” Mediante la metodología de desarrollo de software se van indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además qué personas deben participar en el desarrollo de las actividades y qué papel deben de tener. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.



### **1.5.1 Metodologías tradicionales**

#### **RUP**

El Proceso Unificado es una tecnología ampliamente utilizada. RUP es una metodología iterativa, lo que hace posible que se minimice el riesgo de obtención de un mal producto (o un producto no deseado) porque el sistema puede validarse con el cliente en cada iteración. De esta forma se potencia la robustez del producto y se incluye un marco en el que el problema de tener que gestionar requisitos incompletos, que es bastante frecuente, sea fácil de llevar.

Sus características principales son:

- Guiado/Manejado por casos de uso.
- Centrado en arquitectura.
- Iterativo e Incremental.

### **1.5.2 Metodologías Ágiles**

Actualmente para desarrollar un proyecto con éxito, debe estar regido por una metodología de desarrollo, la cual puede seguir uno o varios modelos de ciclo de vida, o sea, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto. Las Metodologías Ágiles se basan en los siguientes principios:

- Realizar entregas cortas en el tiempo y continuas.
- Dar la bienvenida a los cambios.
- Entregas periódicas y frecuentes que funcionen.
- La comunicación directa es el método más eficiente y efectivo para comunicar información. Intenta evitar el teléfono, correos electrónicos, fax, etc.
- La medida principal de progreso es el software que funciona.

- Buen diseño y calidad técnica.
- La simplicidad es algo básico.

Entre las más conocidas están: **[10]**

- **Metodología Crystal**
- **Dynamic Systems Development Method (DSDM)**
- **Adaptive Software Development (ASD)**
- **Feature - Driven Development (FDD)**
- **Lean Development (LD)**
- **SCRUM**
- **XP (Extreme Programming)**

### **1.5.3 SCRUM**

Esta metodología se basa en una filosofía del desarrollo ágil, creado por Hirotaka Takeuchi e Ikujiro Nonaka ahí por el año 1986, para desarrollo de Software. SCRUM aunque puede ser usado para otro tipo de proyectos y tiene demostrada efectividad en otras áreas, aunque generalmente es funcional solo para desarrollos de software porque para eso fue diseñado. La idea es desarrollar aplicaciones mucho más rápido y eficazmente.

### **1.5.4 XP (Extreme Programming)**

XP es una de las metodologías ágiles de desarrollo de software más exitosas en la actualidad, utilizadas para proyectos de corto plazo y corto equipo. Centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software. XP se basa en realimentación continua entre el cliente y el desarrollador, comunicación fluida, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como

especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. [20]

#### **1.5.4.1 Valores de la metodología XP**

##### **Simplicidad**

XP propone el principio de hacer la cosa más simple que pueda funcionar, en relación al proceso y la codificación. Es mejor hacer hoy algo simple, que hacerlo complicado y probablemente nunca usarlo mañana. El programador tiene que estar en constante comunicación con el cliente para satisfacer sus requisitos y responder rápidamente a los cambios de los mismos. Muchos problemas que surgen en los proyectos se deben a que después de concretar los requisitos que debe cumplir el programa no hay una revisión de los mismos, pudiendo dejar olvidados puntos importantes.

##### **Comunicación**

El Extreme Programming se nutre del ancho de banda más grande que se puede obtener cuando existe algún tipo de comunicación: la comunicación directa entre personas. Es muy importante entender cuáles son las ventajas de este medio. Cuando dos (o más) personas se comunican directamente pueden no solo consumir las palabras formuladas por la otra persona, sino que también aprecian los gestos, miradas, etc. que hace su compañero. Sin embargo, en una conversación mediante el correo electrónico, hay muchos factores que hacen de esta una comunicación, por así decirlo, mucho menos efectiva. Muchos problemas que surgen en los proyectos se deben a que después de concretar los requisitos que debe cumplir el programa no hay una revisión de los mismos, pudiendo dejar olvidados puntos importantes.

##### **Retroalimentación (Feedback)**

Retroalimentación concreta y frecuente del cliente, del equipo y de los usuarios finales da una mayor oportunidad de dirigir el esfuerzo eficientemente. Mediante la retroalimentación se ofrece al cliente la posibilidad de conseguir un sistema adecuado a sus necesidades. Se le va mostrando el proyecto a tiempo

para sugerir cambios y poder retroceder a una fase anterior para rediseñarlo a su gusto.

### **Coraje**

El coraje es un valor muy importante dentro de la programación extrema. Un miembro de un equipo de desarrollo extremo debe de tener el coraje de exponer sus dudas, miedos, experiencias sin “embellecer” éstas de ninguna de las maneras. Esto es muy importante ya que un equipo de desarrollo extremo se basa en la confianza para con sus miembros. Faltar a esta confianza es una falta más que grave. El coraje existe en el contexto de los otros tres valores.

Cada uno de ellos se apoya en los demás:

- Se requiere valor para comunicarse con los demás cuando eso podría exponer la propia ignorancia.
- Se requiere valor para mantener el sistema simple, dejando para mañana las decisiones de mañana.
- Se requiere coraje para confiar en que la retroalimentación durante el camino es mejor que tratar de adivinar todo con anticipación.
- Y, sin un sistema simple, comunicación constante y retroalimentación, es difícil mantenerse valiente.

#### **1.5.4.2 Prácticas de la metodología XP**

Algunas de ellas son:

##### **Planificación incremental**

La Programación Extrema asume que la planificación nunca será perfecta, y que variará en función de cómo varíen las necesidades del negocio. Por tanto, el valor real reside en obtener rápidamente un plan inicial, y contar con mecanismos de feedback que permitan conocer con precisión dónde estamos. Como es lógico, la planificación es iterativa: un representante del negocio decide al comienzo de cada iteración qué características concretas se van a

implementar. El objetivo de la XP es generar versiones de la aplicación tan pequeñas como sea posible, pero que proporcionen un valor adicional claro, desde el punto de vista del negocio. A estas versiones se las denomina *release*s. Una *release* cuenta con un cierto número de historias. La historia es la unidad de funcionalidad en un proyecto XP, y corresponde a la mínima funcionalidad posible que tiene valor desde el punto de vista del negocio. Durante cada iteración se cierran varias historias, lo que hace que toda iteración añada un valor tangible para el cliente. Es fundamental en toda esta planificación la presencia de un representante del cliente, que forma parte del equipo y que decide cuáles son las historias más valiosas. Estas historias son las que se desarrollarán en la iteración actual. Como se puede ver, y como siempre ocurre con la Programación Extrema, el enfoque utilizado para llevar a cabo la planificación es eminentemente pragmático. Gran parte de la eficacia de este modelo de planificación deriva de una división clara de responsabilidades, que tiene en cuenta las necesidades del negocio en todo momento. Dentro de esta división, el representante del cliente tiene las siguientes responsabilidades:

- Decidir qué se implementa en cada *release* o iteración.
- Fijar las fechas de fin de la *release*, recortando unas características o añadiendo otras.
- Priorizar el orden de implementación, en función del valor de negocio.

## **Testing**

La ejecución automatizada de test es un elemento clave de la XP. Existen tanto tests internos (o test de unidad), para garantizar que el mismo es correcto, como tests de aceptación, para garantizar que el código hace lo que debe hacer. El cliente es el responsable de definir los tests de aceptación, no necesariamente de implementarlos. Él es la persona mejor cualificada para decidir cuál es la funcionalidad más valiosa. El hecho de que los tests sean automatizados es el único modo de garantizar que todo funciona: desde el punto de vista de la XP, si no hay tests, las cosas sólo funcionan en apariencia. Aún más, si un test no está automatizado, no se le puede considerar como tal.

El objetivo de los tests no es corregir errores, sino prevenirlos. Por ejemplo, los tests siempre se escriben antes que el código a testear, no después: esto aporta un gran valor adicional, pues fuerza a los desarrolladores a pensar cómo se va a usar el código que escriben, poniéndolos en la posición de consumidores del software. Elaborar los tests exige pensar por adelantado cuáles son los problemas más graves que se pueden presentar, y cuáles son los puntos dudosos. Esto evita muchos problemas y dudas, en lugar de dejar que aparezcan “sobre la marcha”. Un efecto lateral importante de los tests es que dan una gran seguridad a los desarrolladores: es posible llegar a hacer cambios más o menos importantes sin miedo a problemas inesperados, dado que proporcionan una red de seguridad. La existencia de tests hace el código muy maleable.

### **Refactorización continúa**

Es el proceso de modificar el código de un sistema de software de modo que no se altere su comportamiento externo pero se mejore su estructura interna. Se refactoriza el sistema eliminando duplicación, mejorando la comunicación y agregando flexibilidad sin cambiar la funcionalidad. El proceso consiste en una serie de pequeñas transformaciones que modifican la estructura interna preservando su conducta aparente. La práctica también se conoce como Mejora Continua de Código o Refactorización implacable. Se lo ha parafraseado diciendo: “Si funciona bien, arréglole de todos modos”.

### **Diseño simple**

Otra práctica fundamental de la Programación Extrema es utilizar diseños tan simples como sea posible. El principio es “utilizar el diseño más sencillo que consiga que todo funcione”. Se evita diseñar características extra porque a la hora de la verdad la experiencia indica que raramente se puede anticipar qué necesidades se convertirán en reales y cuáles no.

La XP nos pide que no vivamos bajo la ilusión de que un diseño puede resolver todas o gran parte de las situaciones futuras: lo que parece necesario cambia con frecuencia, es difícil acertar a priori. Es obvio que, si no vamos a anticipar futuras necesidades, debemos poder modificar el diseño si alguna de estas se

materializa. La XP soporta estas modificaciones gracias a los tests automatizados. Estos permiten hacer cambios importantes gracias a la red de protección que proporcionan. La refactorización, que hace que el código existente sea claro y sencillo, también ayuda a hacer factibles las modificaciones.

La XP define un “diseño tan simple como sea posible” como aquél que:

- Pasa todos los tests.
- No contiene código duplicado.
- Deja clara la intención de los programadores (enfatisa el qué, no el cómo) en cada línea de código.
- Contiene el menor número posible de clases y métodos.

### **Propiedad colectiva del código**

La XP aboga por la propiedad colectiva del código. En otras palabras, todo el mundo tiene autoridad para hacer cambios a cualquier código, y es responsable de ellos. Esto permite no tener que estar esperando a otros cuando todo lo que hace falta es algún pequeño cambio. Por supuesto, cada cuál es responsable de las modificaciones que haga. El principio básico es “tú lo rompes, tú lo arreglas, no importa si está en el código propio o en el de otros”. Por último, vale la pena tener en cuenta que la existencia de tests automatizados impide que se produzca un desarrollo anárquico, al ser cada persona responsable de que todos los tests se ejecuten con éxito al incorporar los cambios que ha introducido al programa.

### **Integración continúa**

En muchos casos la integración de código produce efectos laterales imprevistos, y en ocasiones la integración puede llegar a ser realmente traumática, cuando dejan de funcionar cosas por motivos desconocidos. La Programación Extrema hace que la integración sea permanente, con lo que todos los problemas se manifiestan de forma inmediata, en lugar de durante

una fase de integración más o menos remota. La existencia de una fase de integración separada tiene dos efectos laterales indeseables: se empieza a hacer codificación “yo-yo”, en la que todo el mundo modifica código “sólo para que funcione, ya lo ajustaremos”, y hace que se acumulen defectos. Evitar que se acumulen defectos es muy importante para la XP, como lo es el conseguir que los defectos que cada programador inyecta los elimine él mismo. Entregas pequeñas Siguiendo la política de la XP de dar el máximo valor posible en cada momento, se intenta liberar nuevas versiones de las aplicaciones con frecuencia. Éstas deben ser tan pequeñas como sea posible, aunque deben añadir suficiente valor como para que resulten valiosas para el cliente. Una entrega no debería tardar más tres meses.

### **Entregas pequeñas**

Siguiendo la política de la XP de dar el máximo valor posible en cada momento, se intenta liberar nuevas versiones de las aplicaciones con frecuencia. Éstas deben ser tan pequeñas como sea posible, aunque deben añadir suficiente valor como para que resulten valiosas para el cliente. Una entrega no debería tardar más tres meses.

### **Estándares de programación**

XP enfatiza la comunicación de los programadores a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación (del equipo, de la organización u otros estándares reconocidos para los lenguajes de programación utilizados). Los estándares de programación mantienen el código legible, facilitando los cambios.

#### **1.5.4.3 Fases de la metodología XP**

Hay diversas prácticas inherentes a la Programación Extrema, en cada uno de los ciclos de desarrollo del proyecto.

### **Planificación**

Historias de Usuario: El primer paso de cualquier proyecto que siga la metodología XP es definir las Historias de Usuario con el cliente. Las Historias



de Usuario tienen la misma finalidad que los casos de uso pero con algunas diferencias: Constan de tres o cuatro líneas escritas por el cliente en un lenguaje no técnico sin hacer mucho hincapié en los detalles; no se debe hablar ni de posibles algoritmos para su implementación ni de diseños de base datos adecuados.

**Release planning:** Tras definir las Historias de Usuario es necesario crear un plan de publicaciones, donde se indiquen las Historias de Usuario que se implementarán para cada versión de la aplicación y las fechas en las que se publicarán dichas versiones. **Iteraciones.** Todo proyecto que siga la metodología XP se ha de dividir en iteraciones de aproximadamente 3 semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las Historias de Usuario definidas en el “Release planning” que serán implementadas.

## **Diseño**

- **Diseños simples:** La metodología XP sugiere que hay que conseguir diseños simples y sencillos. Hay que procurar hacerlo todo lo menos complicado posible para conseguir un diseño fácil de entender e implementar que a la larga costará menos tiempo y esfuerzo desarrollar.
- **Glosarios de términos:** Usar una correcta especificación de los nombres de clases, métodos y propiedades ayudará a comprender el diseño y facilitará futuras ampliaciones y la reutilización del código.
- **Tarjetas CRC:** El uso de las tarjetas CRC (Class, Responsibilities and Collaboration) permiten al programador centrarse y apreciar el desarrollo orientado a objetos olvidándose de los malos hábitos de la programación procedural clásica.

## **Codificación**

El cliente es una parte más del equipo de desarrollo; su presencia es indispensable en las distintas fases de XP. A la hora de codificar una historia

de usuario su presencia es aún más necesaria. No olvidemos que los clientes son los que crean las Historias de Usuario y negocian los tiempos en los que serán implementadas. Antes del desarrollo de cada historia de usuario el cliente debe especificar detalladamente lo que ésta hará y también tendrá que estar presente cuando se realicen los tests que verifiquen que la historia implementada cumple la funcionalidad especificada. La codificación debe hacerse atendiendo a estándares y patrones de codificación ya creados. Programar bajo estándares mantiene el código consistente y facilita su comprensión y la escalabilidad. Crear test que prueben el funcionamiento de los distintos códigos implementados nos ayudará a desarrollar dicho código. Crear estos test antes nos ayuda a saber qué es exactamente lo que tiene que hacer el código a implementar y sabremos que una vez implementado pasará dichos test sin problemas ya que dicho código ha sido diseñado para ese fin.

## **Pruebas**

Un punto importante es crear test que no tengan ninguna dependencia del código que en un futuro evaluará. Hay que crear los tests abstrayéndose del futuro código, de esta forma aseguraremos la independencia del test respecto al código que evalúa. El uso de los test es adecuado para observar la refactorización. Los test permiten verificar que un cambio en la estructura de un código no tiene por qué cambiar su funcionamiento. Para asegurar el funcionamiento final de una determinada historia de usuario se deben crear “Test de aceptación”; estos tests son creados y usados por los clientes para comprobar que las distintas Historias de Usuario cumplen su cometido.

### **1.5.5 ¿Por qué aplicar Scrum y XP?**

Luego del estudio realizado se arriba a la conclusión de que el Sistema estará guiado por la metodología de desarrollo SXP por sus facilidades, documentación y flexibilidad, además es la que mejor se adapta a las condiciones de desarrollo del sistema propuesto.

Antes de plantearse un cambio tan importante como el que puede suponer usar una metodología ágil es necesario tener claro hacia dónde se quiere ir. Lo primero es identificar que problemas existen y evaluar si una metodología ágil

te ayuda a resolver estos problemas; si esta metodología encaja en el proyecto y es posible aplicarla, si quien la va a aplicar cuenta con los conocimientos y los medios necesarios, ya que lanzarse a lo loco a usar esto o aquello simplemente porque está de moda es como algo “arriesgado” y podemos terminar consiguiendo que el remedio sea peor que la enfermedad.

Para el desarrollo de este trabajo se propone el uso de dos de las muchas metodologías ágiles enunciadas anteriormente, Scrum para la parte de planificación, y XP para la parte de desarrollo.

- SCRUM (Jeff Sutherland, Ken Schwaber).
- XP (Extreme Programming – Kent Beck).

De ahí SXP

XP es una de las metodologías ágiles de desarrollo de software más exitosas en la actualidad, utilizadas para proyectos de corto plazo y corto equipo. Centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software. XP se basa en retroalimentación continua entre el cliente y el desarrollador, comunicación fluida, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Scrum es un modelo de referencia que define un conjunto de prácticas y roles, y que puede tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto. Los roles principales en Scrum son el ScrumMaster, que mantiene los procesos y trabaja de forma similar al director de proyecto, el ProductOwner, que representa a los stakeholders (clientes externos o internos), y el Team que incluye a los desarrolladores.

El conjunto de características que forma parte de cada sprint viene del Product

Backlog, que es un conjunto de requisitos de alto nivel priorizados que definen el trabajo a realizar. Los elementos del Product Backlog que forman parte del

sprint se determinan durante la reunión de Sprint Planning. Durante esta reunión, el Product Owner identifica los elementos del Product Backlog que quiere ver completados. Entonces, se determina la cantidad de ese trabajo que puede comprometerse a completar durante el siguiente sprint. Durante el sprint, nadie puede cambiar el Sprint Backlog, lo que significa que los requisitos están congelados durante el sprint. Un principio clave de Scrum es el reconocimiento de que durante un proyecto los clientes pueden cambiar de idea sobre lo que quieren y necesitan (a menudo llamado requirements churn), y que los desafíos impredecibles no pueden ser fácilmente enfrentados de una forma predictiva y planificada. Por lo tanto, Scrum adopta una aproximación pragmática, aceptando que el problema no puede ser completamente entendido o definido, y centrándose en maximizar la capacidad del equipo de entregar rápidamente y responder a requisitos emergentes.



Figura 2: Metodología SXP.

#### 1.5.5.1 Fases de la Metodología SXP

- ✚ Planificación-Definición: Donde se establece la visión, se fijan las expectativas y se realiza el aseguramiento del financiamiento del proyecto. Se generan los documentos que se encuentran relacionados con la concepción inicial del sistema, concepción de sistema, usuarios del negocio, lista de reserva del producto, historias de usuarios, lista de riesgos y modelo de diseño.

- Desarrollo: Es donde se realiza la implementación del sistema hasta que esté listo para ser entregado y se realizan las pruebas.
- Entrega: Puesta en marcha.
- Mantenimiento: Donde se realiza el soporte para el cliente.

De cada una de ellas se despliegan 7 flujos de trabajo: concepción inicial, captura de requisitos, diseño con metáforas, implantación, prueba, entrega de la documentación, soporte e investigación, el cual se utiliza por el equipo de desarrollo cuando sea necesario, es decir, es un flujo que se puede mover y utilizarlo en cualquier parte del ciclo de vida del proyecto. [12]

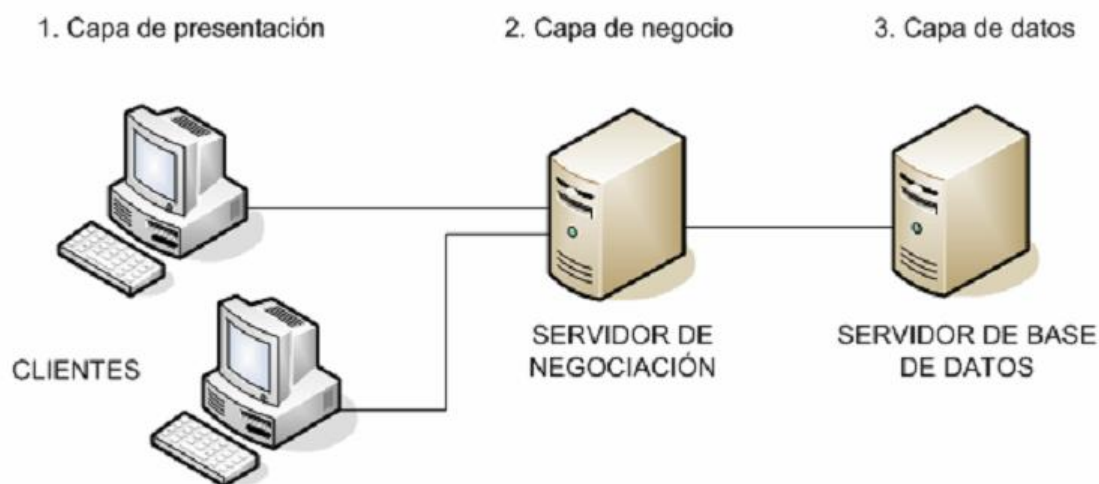
## **1.6 Arquitectura**

La Arquitectura es el esqueleto o base de una aplicación. Representa la organización fundamental de un sistema. Desde los pequeños programas hasta los sistemas más grande poseen una estructura y un comportamiento que los hace clasificables según su "arquitectura". En la Web es muy común la utilización de la arquitectura "3-capas", "n-capas" "MVC", entre otras.

### **1.6.1 Arquitectura en tres capas**

Arquitectura en capas: es donde se define como organizar el modelo de diseño a través de capas, que pueden estar físicamente distribuidas, lo que quiere decir que los componentes de una capa solo pueden hacer referencia a componentes en capas inmediatamente inferiores. Este patrón es importante porque simplifica la comprensión y la organización del desarrollo de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no son consistentes de ningún detalle o interfaz de las superiores.

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario.



**Figura 3: Arquitectura de tres niveles.**

La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio solo se ataca al nivel requerido sin tener que revisar entre código mezclado. Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles; simplemente es necesario conocer la API que existe entre niveles. En el diseño de sistemas informáticos actual se suele usar las arquitecturas multi-nivel o Programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

### Capas o niveles

**Capa de presentación:** es la que ve el usuario (hay quien la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

**Capa de negocio:** es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se

comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él.

**Capa de datos:** es donde residen los datos y es la encargada de acceder ellos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Todas estas capas pueden residir en un único ordenador (no es lo típico). Si bien lo más usual es que haya una multitud de ordenadores en donde reside la capa de presentación (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en el mismo ordenador, y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o más ordenadores. Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varios ordenadores los cuales recibirán las peticiones del ordenador en que resida la capa de negocio. [3]

## **CONCLUSIONES DEL CAPÍTULO 1**

En este capítulo profundizamos en el conocimiento de algunos conceptos necesarios para el entendimiento de la gestión docente. Además de un análisis previo de las tecnologías que serán empleadas a lo largo del desarrollo del sistema propuesto, se dieron a conocer aspectos como el lenguaje de programación, el sistema gestor de bases de datos, la metodología a utilizar, etc. Una vez conocidas las herramientas utilizadas, y los conceptos pertinentes podemos empezar a desarrollar la propuesta de sistema. Fueron abordadas además las ventajas y desventajas de las herramientas de desarrollo permitiendo fundamentar el uso de cada una de ellas. Se hizo referencia también a la arquitectura en capas ya que será fundamental en la implementación del sistema para una mejor organización del mismo.

## 2.2 Planificación y Definición del proyecto

La planificación-definición constituye la primera fase que define la metodología Scrum-XP, en esta se definen todos los procesos a automatizar, así como el tiempo que se determinará para realizar cada una de estas aplicaciones. Por tal motivo es la encargada de generar toda la documentación correspondiente a la concepción inicial del sistema, su definición, además incluye algunos vinculados a la primera fase de los procesos de ingeniería de Software, tales como los relacionados con la estimación inicial de esfuerzos y la valoración de los riesgos. Las planillas surgen a partir de cada actividad en específico y constituyen un baluarte para la documentación del Software, estas cobran suma importancia ya que en ellas se detalla la razón de su existencia, la descripción de los objetivos a tener en cuenta.

El desarrollo de software con metodologías ágiles exige de la creación de pequeños grupos de trabajo, donde los roles son pocos, pero están bien definidas sus actividades. El principal aspecto antes de comenzar el proceso de documentación es distribuir las tareas por cada uno de los roles existentes, lo que garantiza un trabajo organizado, de ahí la necesidad de tenerlos bien definidos.

**Gerente (Management):** Es el responsable de tomar las decisiones finales, acerca de estándares y acuerdos a seguir durante el proyecto. Participa en la definición de objetivos y requerimientos. Tiene la responsabilidad de controlar el avance del software.

**Cliente (Customer):** El cliente contribuye a definir las Historias de Usuario y los casos de prueba de aceptación, para validar su implementación. Además, asigna la prioridad a las Historias de Usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio y participa en la concepción inicial del sistema. El cliente es sólo uno dentro del proyecto pero puede corresponder a un interlocutor que está representando a varias personas que se verán afectadas por el sistema. Contribuye a definir las tareas que involucra la lista de reserva del producto.



**Programador (Programmer):** El programador define las tareas de ingeniería y produce el código del sistema. Además selecciona el estándar de programación a utilizar, controlando incluso la gestión recambios. Además dedica parte de su tiempo a la confección de los Manuales de usuario y de desarrollo.

**Analista (Analyst):** Escribe la concepción del sistema y las Historias de Usuario. Crea el Modelo de historia de usuario del negocio y la LRP. Además, asigna la prioridad a las Historias de Usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio, todo esto lo realiza junto con el cliente.

**Diseñador (Designer):** Son los encargados del diseño del sistema, así como el de los prototipos de interfaces, máximos responsables de la realización del diseño de las metáforas y supervisan el proceso de construcción.

**Encargado de Pruebas (Tester):** Escribe los casos de prueba de aceptación. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas. [2]

### **2.2.1 Plantilla Concepción del Sistema**

Primer documento generado en la fase de Planificación-Definición, este recoge los detalles relacionados con las actividades entrevistas y/o encuestas al cliente, momento en el que se define la concepción inicial del sistema. Este documento además de reflejar la visión general del producto a implementar, también recoge los diferentes roles que intervendrán en el desarrollo del software. Se documenta el tipo de proyecto al que pertenece así como la especificación del Polo Productivo y su clasificación. Se recoge además cuales herramientas serán utilizadas para el desarrollo de la aplicación, el alcance que va a tener, una descripción de los involucrados en el negocio, cuales son los motivos de la necesidad del desarrollo del software y la propuesta de solución. Esta es la plantilla principal dentro de la documentación por ser la guía para los demás documentos que se generan durante el ciclo de desarrollo de software. (Ver [Anexo 1](#))

Para el caso del presente Sistema se realizó una encuesta para conocer cómo funciona la interrelación profesores-departamento, con el fin de conocer la situación actual de los estudiantes, esto dio la posibilidad de saber que se quiere y que hace falta para el desarrollo del Sistema de Gestión Docente de la carrera, estos serán aún más detallados con posterioridad en las demás plantillas que se generan a partir de las metodologías ágiles en cuestión. A partir de los resultados arrojados por la encuesta realizada se determinó que el producto a desarrollar debe estar acorde con las necesidades imperantes para los usuarios finales. Además de integrar otros sistemas para darle al mismo. Los colores del sitio deben estar en correspondencia con las características del territorio donde se encuentra enmarcada la universidad, la forma de visualizar el contenido debe estar de manera que el acceso a ellos sea rápido y comprensible, para esto se llevó a cabo una correcta AI (arquitectura de la información) para que todo el contenido estuviese al alcance en tan solo tres niveles de navegación para más información.

**Roles:** Analista y Cliente.

## **CAPÍTULO 2: Planificación y Diseño**

### **2.1 Introducción**

En este capítulo veremos las cuestiones referentes a los parámetros principales del Sistema de Gestión Docente, se presenta parte de la propuesta de solución de la investigación, haciendo uso adecuado de las metodologías de desarrollo SXP en la que se aborda la fase de planificación y definición, explicándose toda la dinámica del proyecto basándose en el expediente del proyecto.

La metodología está dividida en cuatro fases importante para el desarrollo del proyecto las cuales son las siguientes:

- Planificación - Definición
- Desarrollo

- Entrega
- Mantenimiento

Cada una de estas fases está compuesta por una serie de actividades que son las que generan los artefactos que quedan incluidos en el expediente de proyecto.

### 2.3 Lista de Reserva del Producto (LRP)

La plantilla de Lista de Reserva del Producto, es el primer artefacto generado en la etapa de captura de requisitos, está conformada por una lista priorizada que define el trabajo que se va a realizar en el proyecto. Cuando un proyecto comienza es muy difícil tener claro todos los requerimientos sobre el producto. Sin embargo, suelen surgir los más importantes que casi siempre son más que suficientes para una iteración.

Esta lista puede crecer y modificarse a medida que se obtienen más conocimientos acerca del producto y del cliente. Con la restricción de que sólo puede cambiarse entre iteraciones. El objetivo es asegurar que el producto definido al terminar la lista es el más correcto, útil y competitivo posible y para esto la lista debe acompañar los cambios en el entorno y el producto. Esta lista puede estar conformada por requerimientos técnicos y del negocio, funciones, errores a reparar, defectos, mejoras y actualizaciones tecnológicas requeridas.

**Roles:** Analista y Cliente

Asignado a	Ítem *	Descripción	Estimación
<b><i>Prioridad Muy Alta</i></b>			
Yadiris Martínez Cordero	1.	Autenticar Usuarios	2 Semanas
	2.	Insertar Facultades	
	3.	Listar Facultades	

	4.	Modificar Facultades	
	5.	Eliminar Facultades	
	6.	Insertar Carreras	
	7.	Listar Carreras	
	8.	Modificar Carreras	
	9.	Eliminar Carreras	
	10.	Insertar Provincias	1 Semana
	11.	Listar Provincias	
	12.	Modificar Provincias	
	13.	Eliminar Provincias	
	14.	Insertar Municipios	
	15.	Listar Municipios	
	16.	Modificar Municipios	
	17.	Eliminar Municipios	
	18.	Insertar Profesores	
	19.	Listar Profesores	

	20.	Modificar Profesores	
	21.	Eliminar Profesores	
	<b>Prioridad Alta</b>		
	22.	Asignar Coordinadores	1 Semana
	23.	Listar Coordinadores	
	24.	Modificar Coordinadores	
	25.	Eliminar Coordinadores	
	26.	Asignar Administradores	
	27.	Listar Administradores	
	28.	Eliminar Administradores	
	29.	Insertar Estudiantes	
	30.	Listar Estudiantes	
	31.	Modificar Estudiantes	
	32.	Eliminar Estudiantes	
	33.	Insertar Asignaturas	
	34.	Listar Asignaturas	

	35.	Modificar Asignaturas	
	36.	Eliminar Asignaturas	
	37.	Conformar Claustros	
	38.	Listar Claustros	
	39.	Modificar Claustros	
	40.	Salvar fin de Curso.	2 Semanas
	41.	Cambiar de Año	
<b><i>Prioridad Media</i></b>			
Yadiris Martínez Cordero	42.	Insertar Semestres	1 Semana
	43.	Listar Semestres	
	44.	Modificar Semestres	
	45.	Eliminar Semestres	
	46.	Insertar fecha de Cortes	
	47.	Modificar fecha de Cortes	
	48.	Listar Cortes	

	49.	Eliminar Cortes	1 Semana
	50.	Insertar Evaluaciones	
	51.	Listar Evaluaciones	
	52.	Modificar Evaluaciones	
	53.	Exportar reporte de evaluaciones.	1 Semana
	54.	Exportar listado de estudiantes.	
	55.	Exportar listado de profesores.	
	56.	Exportar listado de coordinadores.	
Prioridad Baja			
Yadiris Martínez Cordero	57.	Subir Documentos	1 Semana
	58.	Listar Documentos	
	59.	Eliminar Documentos	
	60.	Insertar Informaciones	
	61.	Listar Informaciones	

	62.	Modificar Informaciones	
	63.	Eliminar Informaciones	
	64.	Buscar Estudiantes	2 Semanas
	65.	Buscar Profesores	
	66.	Buscar Coordinadores	
	67.	Buscar Evaluaciones	
	68.	Cambiar Contraseña	
<b>Requisitos No Funcionales (RNF)</b>			
<b>Apariencia o Interfaz Externa:</b>  Diseño sencillo permitiendo que se pueda interactuar de manera fácil con el sistema. Su interfaz agradable logra captar la confianza de los usuarios así como su identificación con los colores y formatos en que realizan sus acciones.			
<b>Rendimiento:</b>  Como la aplicación está concebida para un ambiente cliente/servidor los tiempos de respuestas deben ser generalmente rápidos al igual que la velocidad de procesamiento de la información. Estará implementado sobre una tecnología Web, facilitando su uso a través de la red.			



**Soporte:**

La base de datos que utiliza el sistema como medio de almacenamiento de la información está soportada sobre un gestor de bases de datos MYSQL.

**Usabilidad:**

El sistema en general se desarrolla con el objetivo de facilitar el trabajo manual realizado hasta el momento por lo que se espera cuente con un alto nivel de usabilidad por los usuarios debido a que constituye una forma más fácil de realizar el trabajo. El sistema garantizará un acceso fácil y rápido por los usuarios.

**Portabilidad:**

El sistema trabaja sobre Windows y Linux.

**Tabla 1: Lista de Reserva del producto**

## 2.4 Plantilla Historias de Usuario

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software, lo que equivaldría a los casos de uso en el proceso unificado. Las mismas son escritas por los clientes como las tareas que el sistema debe hacer y su construcción depende principalmente de la habilidad que tenga el cliente para definirlas.

Estas se encargan de dirigir la construcción de las pruebas de aceptación, elemento clave en XP (deben generarse una o más pruebas para verificar que la historia ha sido correctamente implementada) y son utilizadas para estimar tiempos de desarrollo. En este sentido, sólo proveen detalles suficientes para hacer una estimación razonable del tiempo que llevará implementarlas. En el momento de implementar una historia de usuario, se debe detallar a través de la comunicación con el cliente. En esta plantilla los campos de puntos estimados y puntos reales se llenan, luego del desarrollo de la actividad de estimación de esfuerzo, donde se decide qué tiempo se le dedicará a cada historia de usuario. (Ver [Anexo 2](#))

Las historias de usuario proporcionan ventajas, tales como:

- ✓ Están escritas en lenguaje del cliente, por lo que es muy fácil su comprensión.
- ✓ Especifican cada uno de los requisitos del sistema, sin necesidad de documentaciones extensas.
- ✓ Reflejan todas las características del sistema.
- ✓ Si se definen correctamente, guían el proceso de implementación.

**Roles:** Analista y Cliente

## **2.5 Planificación de entregas**

En esta fase se establece la prioridad de cada HU, y a continuación, se realiza una estimación del esfuerzo necesario de cada una de ellas por parte de los programadores. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente.

Las estimaciones asociadas a la implementación de las historias se establecen empleando como medida el punto de estimación. Un punto de estimación equivale a una semana ideal de programación, donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción, este punto de estimación que se utiliza para representar la semana ideal, es de 5 días.

Las historias generalmente tienen un valor de 1 a 3 puntos. Además, se mantiene un registro de la velocidad de desarrollo, establecida por puntos de iteración, basado fundamentalmente en la suma de puntos de estimación correspondientes a las HU, que fueron terminadas en la última iteración.

### **2.5.1 Planificación de Iteraciones**

A partir de las HU antes expuestas y la estimación del esfuerzo propuesto para la realización de las mismas, se procede a realizar la planificación de la etapa de implementación del sistema, apoyándose en el tiempo e intentando

concentrar las funcionalidades relacionadas en una misma iteración. En este plan se establece cuántas iteraciones serán necesarias realizar sobre el sistema para su terminación. El plan de iteraciones puede contener indicaciones sobre las HU. En relación con lo antes tratado se decide realizar el sistema en 5 iteraciones, las cuales se explican de forma detalla a continuación:

### **Primera Iteración:**

En esta primera iteración tiene como objetivo darle cumplimiento a las Historias de Usuario que se consideraron de importancia inicial para el desarrollo del sistema. Al concluir dicha iteración se contará con todas las funcionalidades descritas en las HU 1 hasta la 6.

Además se tendrá la primera versión de prueba, que contará con una vista inicial del sistema esta primera versión se le presentarán al cliente con el objetivo de obtener una retroalimentación del mismo para posteriores iteraciones del producto.

### **Segunda Iteración:**

Esta iteración tiene como finalidad desarrollar las HU 7, 8, 9 y 10. Las mismas son las que brindan las funcionalidades de Gestionar roles, ya sea administradores o coordinadores de año, gestionar estudiantes, asignaturas, así como conformar los claustros de profesores que tendrá cada grupo.

El resultado de esta iteración integrada a la iteración anterior dará como resultado la segunda versión del sistema, además de la segunda versión de prueba del mismo. Esta versión será entregada al cliente para verificar si cumple con los requisitos acordados.

### **Tercera Iteración:**

En esta iteración de la aplicación se tiene como propósito llevar a cabo el desarrollo de la HU 11, 12, 13 y 14 las cuáles proporcionan las funcionalidades de salvar el curso, realizar el cambio de año, gestionar semestres y cortes evaluativos. Estas historias de usuario serán integradas con el resultado de las

iteraciones anteriores y será entregada al cliente para verificar si cumple los requisitos acordados.

#### **Cuarta Iteración:**

Esta iteración de la aplicación tiene como propósito llevar a cabo el desarrollo de la HU 15 a la 17 las cuáles proporcionan las funcionalidades de gestionar evaluaciones, ya sean cualitativas (cortes evaluativos B, R y M) o cuantitativas (notas de trabajos de control, seminarios, etc.), además de exportar dichas evaluaciones y los listados de estudiantes, profesores y coordinadores.

#### **Quinta Iteración:**

Esta última iteración de la aplicación tiene como propósito llevar a cabo el desarrollo de la HU 18 a la 21 las cuáles proporcionan las funcionalidades de gestionar documentos, carga y descarga de planillas, P4, etc., gestionar información, realizar búsquedas y cambiar contraseña.

Estas historias de usuario serán integradas con el resultado de las iteraciones anteriores, y como fruto de esta integración se obtendrá la versión 1.0 del producto final. A partir de este momento la aplicación será puesta a un proceso de prueba para evaluar el desempeño de la misma.

### **2.5.3 Plan de duración de iteraciones**

Iteración	Historias de Usuario	Duración Total
1	1. Autenticar Usuario	4/5
	2. Gestionar Facultad	3/5
	3. Gestionar Carrera	
	4. Gestionar Provincia	3/5
	5. Gestionar Municipio	1/5
	6. Gestionar Profesores	
3 Semanas		

<b>2</b>	<b>7. Gestionar Roles</b>	<b>1</b>
	<b>8. Gestionar Estudiante</b>	<b>2/5</b>
	<b>9. Gestionar Asignatura</b>	
	<b>10. Gestionar Claustro</b>	<b>3/5</b>
<b>3 Semanas</b>		
<b>3</b>	<b>11. Salvar Curso</b>	<b>1</b>
	<b>12. Cambiar de Año</b>	<b>1</b>
	<b>13. Gestionar Semestre</b>	
	<b>14. Gestionar Corte</b>	<b>2/5</b>
<b>3 Semanas</b>		
<b>4</b>	<b>15. Gestionar Evaluaciones</b>	<b>1</b>
	<b>16. Exportar Evaluaciones</b>	<b>2/5</b>
	<b>17. Exportar Listados</b>	
<b>3 Semanas</b>		
<b>5</b>	<b>18. Gestionar Documentos</b>	<b>3/5</b>
	<b>19. Gestionar Informaciones</b>	<b>2/5</b>
	<b>20. Realizar Búsquedas</b>	
	<b>21. Cambiar Contraseña</b>	<b>1</b>
<b>3 Semanas</b>		

Tabla 2: Plan de duración de iteraciones.

#### 2.5.4 Valoración de Riesgos.

La planilla de lista de riesgos, es el documento que se genera de la actividad de valoración de riesgos. En ella se definen los posibles riesgos que actuarán sobre el proceso de desarrollo de software, así como la estrategia trazada para

mitigarlos, además de un plan de contingencia que describe que curso seguirán las acciones si el riesgo se materializa.

### **Plan de Contingencia.**

#### **Riesgo # 1**

En caso de que no se logre mitigar este riesgo se debe tomar como una solución alternativa utilizar herramientas y tecnologías que se conozcan.

#### **Riesgo # 2**

Si no se logra el contacto con los clientes se deberá considerar estudiar el negocio por otras vías.

Para comprender cuáles son los datos que se recogerán en esta planilla, se muestra la misma gráficamente.

Riesgos	Tipos de Riesgos	Impacto	Descripción	Probabilidad	Efectos	Mitigación de Riesgos
<b>Escaso Dominio de las Herramientas a utilizar.</b>	Tecnológico	Puede atrasar el plazo de entrega del software	Estudiando las herramientas y tecnologías a utilizar en el desarrollo del software se puede llegar a la conclusión que estas son las más factibles a emplear por sus características.	Medias.	Serias.	Intenso estudio de las bibliografías que tratan estas tecnologías, además de entrevistar al personal capacitado en el tema.
<b>Falta de comunicación con los clientes.</b>	Organización.	Pobre captura de requisitos.	Los clientes tienen mucha responsabilidad por lo que no disponen de mucho tiempo para realizar reuniones y así determinar los requisitos del sistema.	Medias.	Serias.	Planificación de reuniones con el cliente.

Tabla 3: Lista de Riesgos

## 2.6 Tarjetas CRC

En este epígrafe tiene lugar la realización de las tarjetas de clases, responsabilidades y colaboración, conocidas tradicionalmente como tarjetas CRC, las cuáles se realizan con el objetivo de facilitar la comunicación y documentar los resultados. Además, las mismas permiten la total participación y contribución del equipo de desarrollo en la tarea de diseño. Una tarjeta CRC representa un objeto, por tanto es una clase, cuyo nombre se ubica en forma de título en la parte superior de la tarjeta, los atributos y las responsabilidades más significativas se colocan a la izquierda y las clases implicadas con cada responsabilidad a la derecha, en la misma línea de su requerimiento correspondiente. Para mejor comprensión de las mismas, se determina agruparlas por HU. (Ver [Anexo 3](#))

## CONCLUSIONES DEL CAPÍTULO 2

En este capítulo se han desarrollado las bases que sustentarán al sistema. Profundizamos en la fase de planificación y definición en la cual se explica parte de la dinámica del proyecto basándose en el expediente del proyecto, se planificaron las Historias de Usuario que se deben tener en cuenta posteriormente para comprender mejor el problema que la aplicación pretende solucionar y enfocándose en la programación orientada a objetos dentro de la fase de diseño de la metodología XP, se elaboraron las tarjetas CRC. Culminando así esta fase por lo que el equipo de trabajo está listo para pasar a la siguiente etapa.



## **CAPÍTULO 3: Implementación y Pruebas**

### **3.1 Introducción**

En este capítulo se aborda lo perteneciente a la fase de desarrollo de acuerdo a la metodología de desarrollo, aparecen las tareas de ingeniería para llevar a cabo el desarrollo de las historias de usuario, así como las pruebas de aceptación que fueron realizadas por el cliente para demostrar que el sistema funcione adecuadamente. Estas pruebas fueron llevadas a cabo antes de cada entrega que se realizó durante todo el desarrollo del proyecto.

### **3.2 Desarrollo del proyecto**

En XP generalmente las historias de usuarios se dividen en tareas de ingenierías (TI). Estas se crean para lograr una mejor planificación de las historias; con ellas se pretende cumplir con las funcionalidades generales de cada historia. Las tareas no tienen que obligatoriamente ser entendidas por el cliente, pueden ser escritas en lenguajes técnicos, puesto que estas son usadas únicamente por los programadores.

#### **3.3.2 Plantilla Tarea de ingeniería**

La plantilla Tarea de ingeniería, tiene gran importancia, pues permite definir cada una de las actividades asociadas a las Historias de Usuario. Así se estimará el tiempo que llevará cada historia de usuario en implementarse de acuerdo a su complejidad. Esta plantilla proporciona ventajas tales como: Permite organizar el proceso de implementación, pues las tareas se van implementando de acuerdo a su prioridad, posibilita conocer el grado de complejidad de cada historia de usuario, teniendo en cuenta la cantidad de tareas asociadas.

**Roles:** Programador

#### **3.3.3 Tareas de ingeniería por Historias de usuarios**

Dentro del contenido de este plan, las HU se descomponen en tareas de programación o ingeniería, y a su vez, estas son asignadas al equipo de

desarrollo para su implementación. Las tareas no tienen que ser entendidas necesariamente por el cliente, pues las mismas, sólo son utilizadas por los miembros del equipo de desarrollo, por lo que pueden ser escritas en lenguaje técnico. Las mismas se representan mediante las tarjetas de tareas.

<b>Historias de usuarios</b>	<b>Tareas de ingenierías</b>
<b>1. Autenticar Usuario</b>	Insertar los datos para entrar al Sistema (nombre de usuario y contraseña).
<b>2. Gestionar Facultad</b>	Insertar Facultad.  Modificar Facultad.  Eliminar Facultad.  Mostrar listado de Facultades.
<b>3. Gestionar Carrera</b>	Insertar Carrera.  Modificar Carrera.  Eliminar Carrera.  Mostrar listado de Carreras.
<b>4. Gestionar Provincia</b>	Insertar Provincia.  Modificar Provincia.  Eliminar Provincia.  Mostrar listado de Provincias.
<b>5. Gestionar Municipio</b>	Insertar Municipio.  Modificar Municipio.

	<b>Eliminar Municipio.</b>  <b>Mostrar listado de Municipios.</b>
<b>6. Gestionar Profesor</b>	<b>Insertar Profesor.</b>  <b>Modificar Profesor.</b>  <b>Eliminar Profesor.</b>  <b>Mostrar listado de Profesores.</b>
<b>7. Gestionar Rol</b>	<b>Insertar Coordinador.</b>  <b>Modificar Coordinador.</b>  <b>Eliminar Coordinador.</b>  <b>Mostrar listado de Coordinadores.</b>  <b>Insertar Coordinador.</b>  <b>Eliminar Coordinador.</b>  <b>Mostrar listado de Coordinador.</b>
<b>8. Gestionar Estudiante</b>	<b>Insertar Estudiante.</b>  <b>Modificar Estudiante.</b>  <b>Eliminar Estudiante.</b>  <b>Mostrar listado de Estudiantes.</b>
<b>9. Gestionar Asignatura</b>	<b>Insertar Asignatura.</b>  <b>Modificar Asignatura.</b>  <b>Eliminar Asignatura.</b>

	<b>Mostrar listado de Asignaturas.</b>
<b>10.Gestionar Claustro</b>	<b>Conformar Claustros.</b>  <b>Listar Claustros.</b>  <b>Modificar Claustros.</b>
<b>11.Salvar Fin de Curso</b>	<b>Salvar Fin de Curso</b>
<b>12.Cambiar Año</b>	<b>Cambiar Año</b>
<b>13.Gestionar Semestre</b>	<b>Insertar Semestre.</b>  <b>Eliminar Semestre.</b>  <b>Modificar Semestre.</b>
<b>14.Gestionar Corte</b>	<b>Insertar fecha de corte</b>  <b>Modificar fecha de corte</b>  <b>Listar Corte.</b>  <b>Eliminar Corte.</b>
<b>15.Gestionar Evaluación</b>	<b>Insertar Evaluación.</b>  <b>Modificar Evaluación.</b>  <b>Eliminar Evaluación.</b>  <b>Mostrar listado de Evaluación.</b>
<b>16.Exportar reporte de evaluaciones</b>	<b>Exportar reporte de evaluaciones</b>
<b>17.Exportar listado de estudiantes.</b>	<b>Exportar listado de estudiantes.</b>  <b>Exportar listado de profesores.</b>

	Exportar listado de coordinadores.
18.Gestionar Documentos	Subir Documentos.  Listar Documentos.  Eliminar Documentos.
19.Gestionar Informaciones	Insertar Informaciones  Modificar Informaciones.  Eliminar Informaciones.  Mostar listado de Informaciones.
20.Buscar Estudiantes	Buscar Estudiantes.
21. Cambiar Contraseña	Cambiar Contraseña.

Tabla 4: Distribución de tareas por Historias de Usuario

<i>Tarea de Programación</i>	
<i>Número tarea:</i>	<i>Número historia:</i>
<i>Nombre de la tarea:</i>	
<i>Tipo de tarea:</i>	<i>Puntos estimados:</i>
<i>Fecha de inicio:</i>	<i>Fecha de fin:</i>
<i>Programador responsable:</i>	
<i>Descripción:</i>	

Tabla 5: Planilla de Tareas de Ingeniería.

Para consultar las tarjetas de tareas de ingeniería ver: [Anexo 4](#)

### 3.3.3.1 Historias de Usuario abordadas en la primera iteración

Iteración	Orden de implementación por Historias de Usuario	Semanas
1	Autenticar Usuario	4/5
	Gestionar Facultad	3/5
	Gestionar Carrera	3/5
	Gestionar Provincia	1/5
	Gestionar Municipio	1/5
	Gestionar Profesores	3/5

Tabla 6: HU de la Primera Iteración

### 3.3.3.2 Historias de Usuario abordadas en la segunda iteración

Iteración	Orden de implementación por Historias de Usuario	Semanas
2	Gestionar Roles	1
	Gestionar Estudiante	2/5
	Gestionar Asignatura	3/5
	Gestionar Claustro	1

Tabla 7: HU de la Segunda Iteración

### 3.3.3.3 Historias de Usuario abordadas en la tercera iteración

Iteración	Orden de implementación por Historias de Usuario	Semanas
3	Salvar Curso	1
	Cambiar de Año	1
	Gestionar Semestre	2/5
	Gestionar Corte	3/5

Tabla 8: HU de la Tercera Iteración

#### 3.3.3.4 Historias de Usuario abordadas en la cuarta iteración

Iteración	Orden de implementación por Historias de Usuario	Semanas
4	Gestionar Evaluaciones	1
	Exportar Evaluaciones	2/5
	Exportar Listados	3/5

Tabla 9: HU de la Cuarta Iteración

#### 3.3.3.5 Historias de Usuario abordadas en la quinta iteración

Iteración	Orden de implementación por Historias de Usuario	Semanas
5	Gestionar Documentos	3/5
	Gestionar Informaciones	2/5
	Realizar Búsquedas	1
	Cambiar Contraseña	1/5

Tabla 10: HU de la Quinta Iteración

### 3.4 Pruebas

Las pruebas de software son elemento crítico para la garantía de calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación. Una vez generado el código fuente, el software debe ser probado para descubrir y corregir el máximo de errores posibles antes de su entrega al cliente. El objetivo es diseñar una serie de casos de prueba que tengan una alta probabilidad de encontrar errores. Aquí es donde se aplican las técnicas de pruebas del software. En la metodología XP es importante el desarrollo de las pruebas. Cada vez que se quiere implementar las funcionalidades que tendrá el software, XP propone una redacción sencilla de prueba, para ser pasada por el código posteriormente. XP

posee dos tipos de pruebas; las unitarias o TDD (desarrollo dirigido por pruebas, del inglés Test Driven Development), desarrolladas por los

programadores verificando su código de forma automática y las pruebas de aceptación que son las que se utilizarán en este capítulo, dichas pruebas serán evaluadas luego de culminar una iteración verificando así si se cumplió la funcionalidad requerida.

### **3.4.1 Desarrollo Dirigido por Pruebas**

El desarrollo dirigido por pruebas (TDD), se enfoca en la implementación orientada a pruebas. El código debe ser probado paso a paso y obtener un resultado, aunque aún no con lógica para el negocio, pero sí funcional. El TDD se aplica antes de comenzar a implementar cada paso de la tarea en desarrollo asumiendo que la prueba es insatisfactoria desde un inicio. Solo una vez que se haya cumplido de la forma más sencilla posible la lógica del código a probar se asume como cumplida. Luego se realiza un proceso conocido informalmente como "refactorización" de código, el cual consiste en limpiarlo, organizarlo y adaptarlo a los patrones.

## **3.5 Objetivos de las pruebas**

El objetivo de las pruebas, es encontrar el mayor número posible de errores con una cantidad razonable de esfuerzo, aplicado sobre una cantidad de tiempo realista. Se debe ejecutar antes de que llegue al cliente, con el propósito de descubrir todos los errores, de manera que el cliente obtenga el producto de calidad. Otro de sus objetivos son que, un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces, una prueba tiene éxito si descubre un error no detectado hasta entonces, reducir costos de mantenimiento, obtener información concreta acerca de fallas, que pueda usarse como apoyo en la mejora de procesos, y en la de los desarrolladores. En fin nuestro objetivo es diseñar pruebas que sistemáticamente saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y de esfuerzo. [12]



### 3.6 Plantilla Caso de prueba de aceptación

El objetivo de las pruebas de aceptación es validar que un sistema cumple con el funcionamiento esperado y permitir al usuario de dicho sistema que determine su aceptación, desde el punto de vista de su funcionalidad y rendimiento.

**Roles:** Cliente y Tester.

Prueba de Aceptación
HU: Nombre de la HU que va a comprobar su funcionamiento.
Nombre: Nombre del caso de prueba.
<b>Descripción: Descripción del propósito de la prueba.</b>
<b>Condiciones de ejecución: Precondiciones para la prueba se pueda realizar.</b>
<b>Entrada/Pasos de ejecución: Pasos para probar la funcionalidad.</b>
<b>Resultado esperado: Resultado que se desea de la prueba.</b>
<b>Evaluación de la prueba: Aceptada o Denegada.</b>

Tabla 11: Planilla Caso de Prueba de Aceptación.

Para consultar las planillas de Caso de prueba de aceptación ver: [Anexo 5](#)

### CONCLUSIONES DEL CAPÍTULO 3

Con la realización de este capítulo hemos abordado la fase de desarrollo planteada por la metodología SXP en el cual se hace referencia a las tareas de ingeniería correspondientes a cada HU, así como la fase de pruebas planteada por la metodología SXP, con la realización de las pruebas de aceptación en las que el cliente se asegura de que las funciones implementadas cumplan su objetivo satisfactoriamente, probando individualmente cada HU y asignándole la evaluación correspondiente. Todas las pruebas que se realizaron fueron efectivas y el cliente estuvo satisfecho.

## **CAPÍTULO 4: Estudio de Factibilidad del Proyecto**

### **4.1 Introducción**

En la actualidad para un satisfactorio desarrollo de cualquier proyecto se hace imprescindible el estudio de factibilidad para tener en cuenta una estimación de los costos a incurrir logrando así definir si será factible o no desarrollar el mismo.

La probabilidad de restricciones, insuficiencia de recursos y presiones en las fechas de entregas implícitas en el desarrollo de un producto, es real, por lo que es necesario y sensato evaluar la viabilidad del mismo. De identificarse en la fase de definición, posibles riesgos en la concepción del proyecto, se necesita minimizar de forma razonable recursos humanos, materiales y financieros. En general los productos informáticos no están excepto de estas dificultades, por lo que es de vital importancia estimar la relación costo-beneficio.

En este capítulo se expone el estudio de factibilidad del proyecto, centrado en estimaciones de esfuerzo humano, tiempo de desarrollo para su ejecución y costo. Se estiman los beneficios tangibles e intangibles que representan para el sistema propuesto, un análisis de costos y beneficios.

### **4.2 Evaluación Costo-Beneficio**

La mayoría, por no decir todos los proyectos de informática, son evaluados según el criterio de Costo-Beneficio. Esta Metodología, plantea que la conveniencia de la ejecución de un proyecto se determina por la observación conjunta de dos factores:

- El costo, que involucra la implementación de la solución informática, adquisición y puesta en marcha del sistema hardware/software y los costos de operación asociados.
- La efectividad, que se entiende como la capacidad del proyecto para satisfacerla necesidad, solucionar el problema o lograr el objetivo para el

cual se ideó, es decir, un proyecto será más o menos efectivo con relación al mayor o menor cumplimiento que alcance en la finalidad para la cual fue ideado (costo por unidad de cumplimiento del objetivo). El desarrollo de un producto informático, siempre tiene un costo.

Este puede estar justificado por los beneficios tanto tangibles como intangibles que origina el mismo. En este proceso, se necesita de una selección adecuada de los elementos más convenientes para su evaluación.

### **4.3 Efectos Económicos**

- Efectos directos.
- Efectos indirectos.
- Efectos externos.
- Intangibles.

#### **4.3.1 Efectos directos**

##### **4.3.1.1 Positivos**

Se cuenta con una herramienta capaz de mantener la seguridad e integridad de los datos que se procesan.

Facilita la visibilidad de los estudiantes, profesores y coordinadores de año, así como una actualización de las evaluaciones, tanto cualitativas como cuantitativas.

Brinda una forma rápida de acceso a datos del proceso docente (claustrós y evaluaciones) desde cualquier computadora conectada a la red del ISMMM.

##### **4.3.1.2 Negativos**

Para el uso de esta aplicación implementada en plataforma Web se utilizan los navegadores preferiblemente Mozilla Firefox, porque se adapta a cualquier tipo de diseño.

### **4.3.2 Efectos indirectos**

- Los efectos económicos observados que pudiera repercutir sobre otros mercados no son perceptibles, aunque este proyecto no está construido con la finalidad de comercializarse.

### **4.3.3 Efectos externos**

- Se obtendrá un producto disponible que le facilitará la gestión del proceso docente actual en el ISMMM a los profesores, así como estudiantes.

### **4.3.4 Intangibles**

- En la valoración económica siempre hay elementos perceptibles por una comunidad como perjuicio o beneficio, pero al momento de ponderar en unidades monetarias esto resulta difícil o prácticamente imposible.

A fin de medir con precisión los efectos, deberán considerarse dos situaciones:

#### **SITUACIÓN SIN PROYECTO**

Para llevar a cabo el flujo de información actualmente, referente a las evaluaciones tanto cualitativas (cortes docentes) como cuantitativas (trabajos de controles, seminarios, etc.) se hace muy lenta, existen pérdidas de documentos, así como retrasos en las entregas.

Lo anteriormente mencionado se hace de forma manual por lo que trae consigo gastos excesivos de recursos de oficinas, dígase hojas y otros no menos importantes, además de no tener una rápida vía de difusión.

#### **SITUACIÓN CON PROYECTO**

Mediante este sistema se agiliza el proceso de gestionar docente, ya que existe un mejor flujo de información confiable y actualizada y queda garantizada la seguridad de los datos accediendo a ellos con facilidad y rapidez.

## 4.4 Beneficios y Costos Intangibles en el proyecto

### 4.4.1 Costos

- Resistencia al cambio, es decir que puede variar el costo.

### 4.4.2 Beneficios

- Mejor comodidad para los usuarios.
- Mejora la calidad de información por la integridad, oportunidad de la información y confiabilidad.
- Menor tiempo empleado en el proceso de analizar las evaluaciones de los estudiantes.
- Conectividad desde cualquier PC que esté conectada a la red.
- Facilidad a la hora de evaluar situaciones personales como colectivas con respecto a la situación docente.

## 4.5 Ficha de Costo

Para determinar el costo económico del proyecto se utilizará el procedimiento para elaborar una Ficha de Costo de un Producto Informático. Para la elaboración de la ficha se consideran los siguientes elementos de costo, desglosados en moneda libremente convertible y moneda nacional. [13]

### 4.5.1 Costos en Moneda Libremente Convertible

Ficha de Costo	
	Precios(\$)
<b>Costos Moneda Libremente Convertible</b>	
<b>Costos Directos</b>	
Compra de equipos de cómputo	0,00
Alquiler de equipos de cómputo	0,00
Compra de licencia de Software	0,00

Depreciación de equipos		0,00
Materiales directos		0,00
Subtotal		0,00
Costos Indirectos		
Formación del personal que elabora el proyecto		0,00
Gastos en llamadas telefónicas		0,00
Gastos para el mantenimiento del centro		0,00
Know How		0,00
Gastos en representación		0,00
Subtotal		0,00
Gastos de Distribución y Venta		
Participación en ferias o exposiciones		0,00
Gastos en transportación		0,00
Compra de materiales de propagandas		0,00
Subtotal		0,00
Total		0,00

Tabla 12: Costo en Moneda libremente convertible (CUC)

#### 4.5.2 Costo en Moneda Nacional

Ficha de Costo		
		Precios(\$)
Costos Moneda Nacional		
Costos Directos		
Salario del personal que laborará en el proyecto		100,00
12,5% del total de gastos por salarios se dedica a la seguridad		0,00
9.09% de salario total, por concepto de vacaciones a acumular		0,00
Gasto por consumo de energía eléctrica		0,00
Gastos en llamadas telefónicas		0,00
Gastos administrativos		0,00
Subtotal		0,00
Costos Indirectos		
Know How		0,00

<b>Subtotal</b>		<b>0.00</b>
	<b>Total</b>	<b>100,00</b>

**Tabla 13: Costo en Moneda Nacional**

La evaluación económica se efectúa conjuntamente con la que se puede llamar evaluación técnica del proyecto, que consiste en cerciorarse de la factibilidad técnica del mismo. En el análisis de la Factibilidad Técnica del proyecto, se pudo apreciar que:

- Se cuenta con hardware mayor a los requisitos mínimos.
- Se cuenta con la disponibilidad de software.

Por lo que se puede inferir que el proyecto es factible técnicamente.

Normalmente no se encuentran problemas en relación con el mercado o la tecnología disponible que se empleará en la fabricación del producto; por tanto la decisión de inversión casi siempre recae en la evaluación económica. Los costos de los proyectos de informática son relativamente simples de cuantificar, no así los beneficios, que se presentan como ahorro de costos con respecto a la situación base.

**Valores de la variable (Solución manual):**

- Buscar Modelo para cortes y notas. (30 min.)
- Establecer notas de cortes evaluativos y evaluaciones. (30 min.)
- Imprimir cortes y evaluaciones. (3 min.)
- Entregar los cortes y evaluaciones a los coordinadores. (60 min.)

**Valores de la variable (Solución con el software):**

- Cargar cada formulario con los datos necesarios para registrar cortes y evaluaciones. (0.033 min.)
- Establecer notas de cortes evaluativos y evaluaciones. (15 min.)

- Exportar resumen de evaluaciones. (0.16 min.)
- Entrega de cortes y evaluaciones a los coordinadores. (0.033 min.)

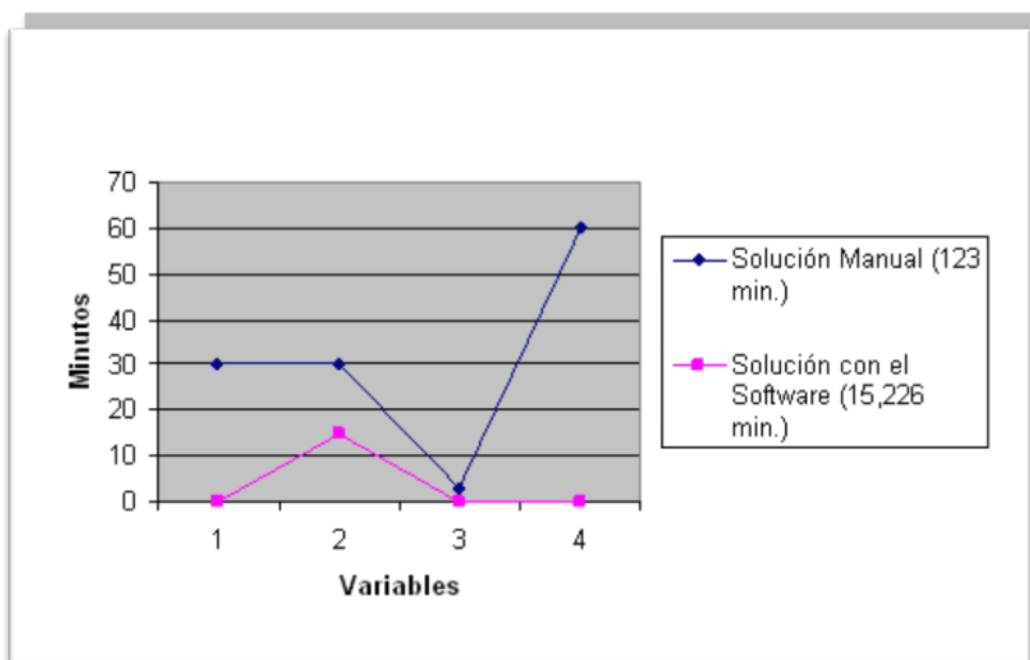


Figura 4: Gráfico relación solución manual y solución con el software.

Teniendo en cuenta los resultados reflejados en la gráfica en cuanto al Punto de Equilibrio queda demostrada la factibilidad del sistema evidenciado por la relación entre la complejidad del problema (cantidad de variables) y el tiempo que demora la solución del mismo de forma manual y con el software.

## CONCLUSIONES DEL CAPÍTULO 4

En este capítulo se realizó el estudio de factibilidad del proyecto mediante la Metodología Costo-Efectividad (Beneficio). En este se analizaron los efectos económicos, los beneficios y costos intangibles, y además se calculó el costo de ejecución del proyecto mediante la ficha de costo, la cual arrojó como resultado \$ 0.00 CUC y \$ 100.00 MN demostrándose la factibilidad del proyecto.



## **CONCLUSIONES GENERALES**

Con el desarrollo de este proyecto se logró obtener un sistema informático seguro, confiable y de fácil utilización pues trata de no distanciarse mucho de la forma de realización actual de los procesos en cuanto a la estructura y organización del trabajo. Una vez concluida la investigación e implementación del sistema, se dieron cumplimiento a los objetivos planteados, obteniendo las siguientes conclusiones:

- Se obtuvo un modelo de cómo se lleva a cabo el proceso actual de gestión docente en el departamento de Informática del ISMM.
- Se utilizaron las herramientas PHP, SGBD MySQL, servidor web Apache entre otras con las que se le realizó el software.
- Luego del análisis de las distintas metodologías se aplicó SXP para el desarrollo del trabajo obteniéndose el expediente de proyecto.
- SIGEDO no sólo está listo para utilizarse en cualquier facultad del ISMM, sino que se puede emplear en cualquier universidad.

## **RECOMENDACIONES**

Poner en explotación el software no solo para la carrera de Informática sino para todas las facultades del ISMM.

Redefinir y agregar funcionalidades acorde a nuevos requisitos que pudiesen surgir, para aumentar el rendimiento del sistema.

Realizar un estudio más profundo de este sistema en vista a perfeccionarlo en versiones futuras.

Ejecutar preferiblemente la aplicación en navegadores Open-Source ya que traducen los estilos de diseño con mayor claridad.

Con la puesta en práctica de las recomendaciones que se expusieron anteriormente se logrará una mejor capacidad de respuesta. Además se incentiva para que se creen o mejoren las funcionalidades de la aplicación propuesta, para que conlleven a un mejor rendimiento de las mismas.

**BIBLIOGRAFÍA**

1. Funcionamiento del Proceso Docente. [Online] [Cited: Enero 16, 2012.]  
<http://es.shvoong.com/social-sciences/sociology/1849819-funciones-del-docente-en-el/>
2. **SANTANA DÍAZ, Osmany.** Sistema de Gestión y Control de las Prácticas Laborales. Trabajo de Diploma. Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”, 2010.
3. **Díaz Sánchez, Iliana y Martínez Rodríguez, Marcos A.** Herramienta para el control de los medios informáticos de la Empresa Cdte. “Ernesto Che Guevara”. Trabajo de Diploma. Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”, 2008.
4. **GRAHAM, Paul.** The Other Road Ahead [Libro]. - 2001.
5. **Masternewmedia.org** [Online] // Beneficios de las aplicaciones basadas en Web y el anuncio de Microsoft de la era “En Vivo”. [Citad: Enero 15, 2012].  
[http://www.masternewmedia.org/es/aplicaciones\\_web/temas\\_de\\_aplicaciones\\_web/Beneficios\\_De\\_Las\\_Aplicaciones\\_Basadas\\_En\\_Web\\_Y\\_El\\_Auncio\\_De\\_Microsoft\\_De\\_La\\_Era\\_En\\_Vivo.htm](http://www.masternewmedia.org/es/aplicaciones_web/temas_de_aplicaciones_web/Beneficios_De_Las_Aplicaciones_Basadas_En_Web_Y_El_Auncio_De_Microsoft_De_La_Era_En_Vivo.htm)
6. **HERNÁN RUIZ, Marcelo.** Programación Web Avanzada. La Habana, Cuba. Editorial Félix Varela, 2006.
7. **Eaprende.com** [Online] // Gestor de Base de Datos: MySQL, PostgreSQL, SQLite. [Cited: Marzo 7, 2012].  
<http://www.eaprende.com/gestor-de-basededatos-mysql-postresql-sqlite.html>
8. **DÍAZ COBAS, Walberto.** Implementación del módulo Caja del Sistema Integral de Gestión CEDRUX, Trabajo de Diploma, UCI, 2009.

9. **Programación Web.** [Citad: Enero 22, 2012]  
<http://www.programacionweb.net/noticias/mysql-pasa-a-manos-de-oracle/>
10. **Metodologías ágiles.** [Cited: Diciembre 12, 2011].  
<http://es.wikipedia.org/w/metodologias-agiles>.
11. **Intercambiosvirtuales.org** [Online] PHP Designer 2008 v6.0.2.0 Professional. - 2009. - enero de 2012. -  
<http://www.intercambiosvirtuales.org/software/php-designer-2008-v6020-professional>.
12. **RIVERA LEGRÁ, Joandris.** Sistema de gestión de los cortes evaluativos. Trabajo de Diploma. Instituto Superior Minero Metalúrgico de Moa “Dr. Antonio Núñez Jiménez”, 2011.
13. GARCÍA PERÉZ, A M. Procedimiento para la elaboración de la ficha de costo de un producto informático. Facultad MFC UCLV. Villa Clara.

## **GLOSARIO DE TÉRMINOS**

**Cliente:** Persona, organización o grupo de personas que solicita la construcción de un sistema, ya sea empezando desde cero, o mediante el refinamiento de versiones sucesivas.

**Glosario de términos:** Términos comunes que se utilizan para describir el sistema.

**HTML:** Hyper Text Markup Language, o simplemente HTML, es un lenguaje de etiquetas de programación.

**Interfaz:** Conjunto de representaciones de operaciones públicas.

**Open-Source:** Estamos en presencia de un software libre el cual permite estudiar su código y adecuarlo a las necesidades que se necesite, o sea es un código abierto.

**Metodología ágil:** Nuevo enfoque metodológico orientado a la gente y los resultados.

**Metodologías de desarrollo de software:** Son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software.

**Metodologías tradicionales:** Metodologías basadas en procesos.

**Proyecto:** Esfuerzo de desarrollo para llevar un sistema a lo largo de un ciclo de vida.

**Requisitos:** Son las funciones, servicios y restricciones operativas del sistema.

**Sistema:** Delimita el mundo sobre el cual se está construyendo el modelo.

**Software:** Se refiere a los programas y datos almacenados en un ordenador.

**Usuario:** Persona que interactúa con el sistema.

## ANEXOS

### Anexo 1: Plantilla Concepción del Sistema

<b>Clasificación del Proyecto</b>		
Desarrollo Web		
<b>Tipo de proyecto.</b>		
Nacional		
<b>Resumen:</b>		
Es una herramienta concebida para los docentes con el objetivo de facilitar y complementar su labor diaria para con los alumnos y mejorar la comunicación directa y constante entre profesores, coordinadores de año y jefes de carrera.		
<b>Palabras claves:</b> Gestión Docente, Aplicación Web		
<b>Surgimiento</b>		
Este producto surge por la no existencia de una herramienta en la carrera de Informática que gestione el desempeño de los estudiantes desde que entran al centro hasta que se gradúan.		
<b>¿Qué es?</b>		
Sistema de Gestión Docente		
<b>Metodología a utilizar</b>		
Como tecnología a utilizar SXP por sus facilidades, documentación y flexibilidad, además es la que mejor se adapta a las condiciones de desarrollo del sistema propuesto.		
<b>Roles</b>	<b>Responsabilidades</b>	<b>Nombre y Apellidos</b>
Gerente	Toma las decisiones finales, acerca de estándares y convenciones a seguir durante el proyecto.	Marcos Marínez Rodríguez
Cliente	Participa en las tareas que involucran la lista de reserva del	Profesores del ISMMM

	producto.	
Programador	Es el encargado de producir el código y escribir las pruebas unitarias.	Yadiris Marínez Cordero
Analista	Es el encargado de escribir las historias de usuario y las pruebas funcionales para validar su implementación.	Yadiris Marínez Cordero
Diseñador	Encargado del diseño del sistema, así como el de los prototipos de interfaces, máximos responsables de la realización del diseño de las metáforas y supervisan el proceso de construcción.	Yadiris Marínez Cordero
Encargado de Pruebas	Es el encargado de ayudar al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente y es responsable de las herramientas de soporte para pruebas.	Yadiris Marínez Cordero
<b>Misión</b>		
Llevar el control de los estudiantes, poder contar automáticamente y de forma digital con las últimas actualizaciones en las distintas materias docentes, cortes, evaluaciones, etc.		
<b>Visión</b>		
Consolidar un buen desarrollo del proceso educativo, agilidad para el mismo además de integrar al claustro de profesores.		
<b>Herramientas utilizadas</b>		
<ul style="list-style-type: none"> <li>• Como herramientas a utilizar se tiene SGBD MySQL y servidor Web</li> </ul>		

Apache, entornos de trabajo Dreamweaver y PHPDesigner, además de contar con herramientas CASE ERStudio.
<b>Solución Propuesta</b>
Se tiene como propuesta de solución el desarrollo de una aplicación Web que permita la gestión, difusión y seguimiento de las actividades docentes en ISMMM.

Tabla 14: Plantilla Concepción del Sistema.

## Anexo 2: Plantillas Historias de Usuario

Historia de Usuario	
<b>Número:</b> No. HU	<b>Usuario:</b> Usuario entrevistado para obtener información.
<b>Nombre:</b> Nombre de la historia de usuario para identificarla.	
<b>Prioridad en el negocio:</b>  Importancia: Alta / Media / Baja	<b>Riesgo en desarrollo:</b>  Dificultad: Alta / Media / Baja
<b>Puntos estimados:</b>  Estimación: de 1 a 3 puntos	<b>Iteración asignada:</b>  Iteración a la que corresponde
<b>Programador responsable:</b> Nombre de encargado de programación.	
<b>Descripción:</b> Una breve descripción de lo que lo que realizará la HU.	
<b>Observaciones:</b> Algunas observaciones de interés.	

Tabla 15: Modelo de Plantilla Historias de Usuario.

Historia de Usuario	
<b>Número:</b> 1	<b>Usuario:</b> Usuarios del Sistema (admin., coordinador, profesor).
<b>Nombre:</b> Autenticar usuario	
<b>Prioridad en el negocio:</b> Muy Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 4/5	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> Los usuarios deben insertar los datos de entrada al sistema	



**Observaciones:** Confirmado.

Tabla 16: HU No.1 Autenticar usuarios.

Historia de Usuario	
<b>Número:</b> 2	<b>Usuario:</b> Administrador
<b>Nombre:</b> Gestionar Facultad	
<b>Prioridad en el negocio:</b> Muy Alta	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 3/5	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador debe crear, eliminar, modificar y mostrar un listado de las facultades. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado.	

Tabla 17: HU No. 2 Gestionar Facultad.

Historia de Usuario	
<b>Número:</b> 3	<b>Usuario:</b> Administrador.
<b>Nombre:</b> Gestionar Carreras.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador debe crear, eliminar, modificar y mostrar un listado de las carreras. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado.	

Tabla 18: HU No. 3 Gestionar Carreras.

Historia de Usuario	
<b>Número:</b> 4	<b>Usuario:</b> Administrador.
<b>Nombre:</b> Gestionar Provincia.	

<b>Prioridad en el negocio:</b> Muy Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 1/5	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador debe crear, eliminar, modificar y mostrar un listado de las provincias. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado.	

Tabla 19: HU No. 4 Gestionar Provincias.

Historia de Usuario	
<b>Número:</b> 5	<b>Usuario:</b> Administrador.
<b>Nombre:</b> Gestionar Municipio.	
<b>Prioridad en el negocio:</b> Muy Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 1/5	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador debe crear, eliminar, modificar y mostrar un listado de los municipios. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado.	

Tabla 20: HU No. 5 Gestionar Municipios.

Historia de Usuario	
<b>Número:</b> 6	<b>Usuario:</b> Administrador.
<b>Nombre:</b> Gestionar Profesores.	
<b>Prioridad en el negocio:</b> Muy Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 3/5	<b>Iteración asignada:</b> 1

<b>Programador responsable:</b> Yadiris Martínez Cordero.
<b>Descripción:</b> El Administrador debe de insertar los usuarios que van a trabajar con el Sistema. Además eliminarlos, modificarlos, cambiarles la contraseña y mostrar un listado de los mismos. El sistema guarda los datos.
<b>Observaciones:</b> Confirmado.

Tabla 21: HU No. 6 Gestionar Profesores.

Historia de Usuario	
<b>Número:</b> 7	<b>Usuario:</b> Administrador.
<b>Nombre:</b> Gestionar roles.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador debe de asignar privilegios de administración, así como designar los profesores que serán coordinadores de año. Además eliminarlos, modificarlos y mostrar un listado de los mismos. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado.	

Tabla 22: HU No. 7 Gestionar Roles.

Historia de Usuario	
<b>Número:</b> 8	<b>Usuario:</b> Administrador, Coordinador.
<b>Nombre:</b> Gestionar estudiante.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 2/5	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador y el Coordinador deben de insertar los estudiantes de su año. Además eliminarlos, modificarlos y mostrar un listado de los mismos. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado	

Tabla 23: HU No. 8 Gestionar Estudiantes.

Historia de Usuario	
<b>Número:</b> 9	<b>Usuario:</b> Administrador, Coordinador.
<b>Nombre:</b> Gestionar asignatura.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3/5	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador y el Coordinador deben de insertar las asignaturas de su año. Además eliminarlas, modificarlas y mostrar un listado de las mismas. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado	

Tabla 24: HU No. 9 Gestionar Asignaturas.

Historia de Usuario	
<b>Número:</b> 10	<b>Usuario:</b> Administrador, Coordinador.
<b>Nombre:</b> Gestionar claustro.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador y el Coordinador deben de seleccionar el claustro de profesores que impartirán clases a los distintos grupos. Además modificarlos y mostrar un listado de los mismos. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado	

Tabla 25: HU No. 10 Gestionar Claustros.

Historia de Usuario	
<b>Número:</b> 11	<b>Usuario:</b> Administrador.
<b>Nombre:</b> Salvar Fin de Curso.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El administrador del sistema puede hacer una salva de todos los	

estudiantes al finalizar el curso. El sistema guarda los datos.

**Observaciones:** Confirmado

Tabla 26: HU No. 11 Salvar Curso.

Historia de Usuario	
<b>Número:</b> 12	<b>Usuario:</b> Administrador.
<b>Nombre:</b> Cambiar Año.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El administrador del sistema puede cambiar de año a los estudiantes que lo hayan aprobado.	
<b>Observaciones:</b> Confirmado	

Tabla 27: HU No. 12 Cambiar Año.

Historia de Usuario	
<b>Número:</b> 13	<b>Usuario:</b> Administrador, Coordinador.
<b>Nombre:</b> Gestionar semestres.	
<b>Prioridad en el negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 2/5	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador y el Coordinador deben de insertar el semestre. Además puede eliminarlos y modificarlos. El sistema guarda los datos.	
<b>Observaciones:</b> Confirmado	

Tabla 28: HU No. 13 Gestionar Semestres.

Historia de Usuario	
<b>Número:</b> 14	<b>Usuario:</b> Administrador, Coordinador.
<b>Nombre:</b> Gestionar cortes.	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3/5	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Yadiris Martínez Cordero.	

**Descripción:** El Administrador y el Coordinador deben de insertar los cortes, así como establecer sus fechas. Además puede eliminarlos y modificarlos, así como listarlos. El sistema guarda los datos.

**Observaciones:** Confirmado

Tabla 29: HU No. 14 Gestionar Cortes.

Historia de Usuario	
<b>Número:</b> 15	<b>Usuario:</b> Administrador, Coordinador y Profesor.
<b>Nombre:</b> Gestionar Evaluación.	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> El Administrador, Coordinador y los profesores deben insertar las evaluaciones, ya sea cualitativa o cuantitativa. Además pueden eliminarlas, modificarlas y listarlas. El sistema guarda los apuntes.	
<b>Observaciones:</b> Confirmado	

Tabla 30: HU No. 15 Gestionar Evaluación.

Historia de Usuario	
<b>Número:</b> 16	<b>Usuario:</b> Todos los usuarios.
<b>Nombre:</b> Exportar reporte de evaluaciones	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 2/5	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> Todos los usuarios pueden hacer reportes de evaluaciones.	
<b>Observaciones:</b> Confirmado	

Tabla 31: HU No. 16 Exportar evaluaciones.

Historia de Usuario	
<b>Número:</b> 17	<b>Usuario:</b> Todos los usuarios.
<b>Nombre:</b> Exportar listados.	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 3/5	<b>Iteración asignada:</b> 4

<b>Programador responsable:</b> Yadiris Martínez Cordero.
<b>Descripción:</b> Todos los usuarios pueden hacer reportes de estudiantes, profesores y coordinadores.
<b>Observaciones:</b> Confirmado

Tabla 32: HU No. 17 Exportar Listados.

Historia de Usuario	
<b>Número:</b> 18	<b>Usuario:</b> Administrador, Coordinador, Profesores.
<b>Nombre:</b> Gestionar documentación.	
<b>Prioridad en el negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3/5	<b>Iteración asignada:</b> 5
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> Los usuarios con acceso al sitio pueden insertar documentación. Además eliminarlas, modificarlas y listarlas. El sistema guarda los documentos.	
<b>Observaciones:</b> Confirmado	

Tabla 33: HU No. 18 Gestionar Documentación.

Historia de Usuario	
<b>Número:</b> 19	<b>Usuario:</b> Administrador, Coordinador, Profesores.
<b>Nombre:</b> Gestionar informaciones.	
<b>Prioridad en el negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 2/5	<b>Iteración asignada:</b> 5
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> Los usuarios con acceso al sitio pueden insertar informaciones que se muestran en la página inicio del sitio. Además eliminarlas, modificarlas y listarlas. El sistema guarda los documentos.	
<b>Observaciones:</b> Confirmado	

Tabla 34: HU No. 19 Gestionar Informaciones.

Historia de Usuario	
<b>Número:</b> 20	<b>Usuario:</b> Todos los usuarios.
<b>Nombre:</b> Realizar Búsquedas.	

<b>Prioridad en el negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 5
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> Todos los usuarios pueden buscar perfiles de estudiantes, profesores, coordinadores, buscar documentos, P4, etc.	
<b>Observaciones:</b> Confirmado	

Tabla 35: HU No. 20 Realizar Búsquedas.

Historia de Usuario	
<b>Número:</b> 2	<b>Usuario:</b> Usuarios del Sistema (admin., coordinador, profesor).
<b>Nombre:</b> Cambiar Contraseña	
<b>Prioridad en el negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Yadiris Martínez Cordero.	
<b>Descripción:</b> Los usuarios deben insertar los datos de entrada al sistema correctamente para poder acceder al mismo.	
<b>Observaciones:</b> Confirmado.	

Tabla 36: HU No. 21 Cambiar Contraseña.

### Anexo 3: Tarjetas CRC

Clase	
<b>Responsabilidades</b>	<b>Colaboraciones</b>
Clase A	Clase B

Tabla 37: Modelo tarjetas CRC.

Clase Autenticar usuarios	
<b>Responsabilidades</b>	<b>Colaboraciones</b>
Autenticar_usuarios	conexion

Tabla 38: Tarjeta CRC No.1 Autenticar Usuarios.

Clase Cambiar Contraseña	
<b>Responsabilidades</b>	<b>Colaboraciones</b>



cambiar_clave	conexion
---------------	----------

Tabla 39: tarjeta CRC No.2 Cambiar Contraseña.

Clase Gestionar facultad	
Responsabilidades	Colaboraciones
insertar_facultad	conexion
eliminar_facultad	
modificar_facultad	
listar_facultad	

Tabla 40: Tarjeta CRC No. 3 Gestionar facultad.

Clase Gestionar carrera	
Responsabilidades	Colaboraciones
insertar_carrera	conexion
eliminar_carrera	
modificar_carrera	
listar_carrera	

Tabla 41: Tarjeta CRC No. 4 Gestionar carrera

Clase Gestionar profesor	
Responsabilidades	Colaboraciones
insertar_profesor	conexion

Tabla 42: Tarjeta CRC No. 5 Gestionar Usuario.

Clase Gestionar rol	
Responsabilidades	Colaboraciones
insertar_coord	conexion

Tabla 43: Tarjeta CRC No. 6 Gestionar Rol.

Clase Gestionar estudiante	
Responsabilidades	Colaboraciones

insertar_estud	conexion
eliminar_estud	
modificar_estud	
listar_estud	

Tabla 44 Tarjeta CRC No. 7 Gestionar Estudiante.

Clase Gestionar asignatura	
Responsabilidades	Colaboraciones
insertar_asig	conexion
eliminar_asig	
modificar_asig	
listar_asig	

Tabla 45: Tarjeta CRC No. 8 Gestionar Asignatura.

Clase Gestionar semestre	
Responsabilidades	Colaboraciones
insertar_sem	conexion
listar _ sem	
eliminar_sem	
modificar_sem	

Tabla 46: Tarjeta CRC No. 9 Gestionar Semestre.

Clase Gestionar cortes	
Responsabilidades	Colaboraciones

insertar_fecha_cortes	conexion
modificar_fecha_cortes	
listar_cortes	
eliminar_cortes	

Tabla 47: Tarjeta CRC No. 10 Gestionar cortes.

Clase Gestionar documentación	
Responsabilidades	Colaboraciones
subir_doc	conexion
descargar_doc	
listar_doc	
eliminar_doc	
modificar_doc	

Tabla 48: Tarjeta CRC No. 11 Gestionar Documentación.

Clase Gestionar evaluación	
Responsabilidades	Colaboraciones
insertar_eval	conexión
eliminar_eval	
modificar_eval	
listar_eval	

Tabla 49: Tarjeta CRC No. 12 Gestionar Evaluación.

Clase Exportar evaluación	
Responsabilidades	Colaboraciones

exportar_eval	conexión  class_ezpdf
---------------	-----------------------------

Tabla 50: Tarjeta CRC No. 13 Reportar Evaluación.

Clase Exportar listados	
Responsabilidades	Colaboraciones
exportar_ estudiantes	conexión
exportar_ profesores	class_ ezpdf
exportar_ coordinadores	

Tabla 51: Tarjeta CRC No.14 Exportar listado de estudiantes

Clase Realizar Búsquedas	
Responsabilidades	Colaboraciones
buscar_estudiantes	conexión
buscar_ profesores	
buscar_coordinadores	
buscar_evaluaciones	

Tabla 52: Tarjeta CRC No. 15 Buscar perfil de estudiante.

Clase Gestionar informaciones	
Responsabilidades	Colaboraciones
insertar _ info	conexión
eliminar_ info	
modificar_ info	
listar_ info	

Tabla 53: Tarjeta CRC No. 16 Gestionar Informaciones

Clase Salvar fin de curso	
Responsabilidades	Colaboraciones
salvar_datos_fin_curso	conexión

Tabla 54: Tarjeta CRC No. 17 Salvar fin de Curso.

Clase Cambiar año	
Responsabilidades	Colaboraciones
cambiar_año	conexion

Tabla 55: Tarjeta CRC No. 18 Cambiar de Año.

Clase Gestionar Provincia	
Responsabilidades	Colaboraciones
insertar_provincia	conexion
listar_provincia	
modificar_provincia	
eliminar_provincia	

Tabla 56: Tarjeta CRC No. 19 Gestionar Provincia.

Clase Gestionar Municipio	
Responsabilidades	Colaboraciones
insertar_municipio	conexion
listar_municipio	
modificar_municipio	
eliminar_municipio	

Tabla 57: Tarjeta CRC No. 20 Gestionar Municipio.

Clase Gestionar Claustro	
Responsabilidades	Colaboraciones

insertar_ claustro	conexion
listar_ claustro	
modificar_ claustro	

Tabla 58: Tarjeta CRC No. 21 Gestionar Claustro

## Anexo 4 Tarjetas de Tareas de Ingeniería o Programación

Tareas de Ingeniería agrupadas por las respectivas historias de usuario a las que pertenecen.

### Primera Iteración

<i>Tarea de Programación</i>	
<b>Número tarea: 1</b>	<b>Número historia:1</b>
<b>Nombre de la tarea: Autenticar Usuario</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 4/5</b>
<b>Fecha de inicio: 23/01/2012</b>	<b>Fecha de fin: 30/01/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción: Autenticación de los usuarios del sistema.</b>	

Tabla 59: Tarea de programación Autenticar usuarios.

<i>Tarea de Programación</i>	
<b>Número tarea: 2</b>	<b>Número historia:2</b>
<b>Nombre de la tarea: Insertar Facultad</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 3/5</b>
<b>Fecha de inicio: 30/01/2012</b>	<b>Fecha de fin: 05/02/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción: Esta tarea es realizada por el Administrador y la misma</b>	

Tabla 60: Tarea de programación Insertar Facultad

<i>Tarea de Programación</i>	
<b>Número tarea: 3</b>	<b>Número historia:3/5</b>
<b>Nombre de la tarea: Modificar Facultad</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 1</b>
<b>Fecha de inicio: 30/01/2012</b>	<b>Fecha de fin: 05/02/2012</b>

<b>Programador responsable:</b> Yadiris Martínez Cordero
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Modificar facultades.

Tabla 61: Tarea de programación Modificar facultad

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 4	<b>Número historia:</b> 2
<b>Nombre de la tarea:</b> Eliminar Facultad	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 30/01/2012	<b>Fecha de fin:</b> 05/02/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Facultad	

Tabla 62: Tarea de programación Eliminar Facultad

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 5	<b>Número historia:</b> 2
<b>Nombre de la tarea:</b> Mostrar Listado de Facultades	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 30/01/2012	<b>Fecha de fin:</b> 05/02/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de facultades	

Tabla 63: Tarea de programación listado de Facultades

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 6	<b>Número historia:</b> 3
<b>Nombre de la tarea:</b> Insertar Carrera.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 06/02/2012	<b>Fecha de fin:</b> 11/01/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita insertar Carrera	

Tabla 64: Tarea de programación Insertar Carrera

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 7	<b>Número historia:</b> 3
<b>Nombre de la tarea:</b> Modificar Carrera	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 06/02/2012	<b>Fecha de fin:</b> 11/01/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Modificar Carrera	

Tabla 65: Tarea de programación Modificar Carrera

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 8	<b>Número historia:</b> 3
<b>Nombre de la tarea:</b> Eliminar Carrera	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 06/02/2012	<b>Fecha de fin:</b> 11/01/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Carrera	

Tabla 66: Tarea de programación Eliminar Carrera

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 9	<b>Número historia:</b> 3
<b>Nombre de la tarea:</b> Mostrar Listado de Carrera	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 06/02/2012	<b>Fecha de fin:</b> 11/01/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Carrera	

Tabla 67: Tarea de programación Listado de carreras

<b>Número tarea:</b> 10	<b>Número historia:</b> 4
<b>Nombre de la tarea:</b> Insertar Provincia	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1/5



<b>Fecha de inicio:</b> 12/02/2012	<b>Fecha de fin:</b> 18/02/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita insertar Provincias.	

Tabla 68: Tarea de programación Insertar Provincia

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 11	<b>Número historia:</b> 4
<b>Nombre de la tarea:</b> Modificar Provincia	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1/5
<b>Fecha de inicio:</b> 12/02/2012	<b>Fecha de fin:</b> 18/02/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Modificar Provincias.	

Tabla 69: Tarea de programación Modificar Provincia

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 12	<b>Número historia:</b> 4
<b>Nombre de la tarea:</b> Eliminar Provincia	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1/5
<b>Fecha de inicio:</b> 12/02/2012	<b>Fecha de fin:</b> 18/02/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Provincias.	

Tabla 70: Tarea de programación Eliminar Provincia

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 13	<b>Número historia:</b> 4
<b>Nombre de la tarea:</b> Mostrar Listado de Provincia	
<b>Fecha de inicio:</b> 12/02/2012	<b>Fecha de fin:</b> 18/02/2012
<b>Fecha de inicio:</b> 23/01/2012	<b>Fecha de fin:</b> 30/01/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	

**Descripción:** Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Provincias

Tabla 71: Tarea de programación Listado de Provincias

<b>Tarea de Programación</b>	
<b>Número tarea: 14</b>	<b>Número historia:5</b>
<b>Nombre de la tarea:</b> Insertar Municipios.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 1/5</b>
<b>Fecha de inicio: 19/02/2012</b>	<b>Fecha de fin: 25/02/2012</b>
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita insertar Municipios	

Tabla 72: Tarea de programación Insertar Municipio

<b>Tarea de Programación</b>	
<b>Número tarea:15</b>	<b>Número historia:5</b>
<b>Nombre de la tarea:</b> Modificar Municipios	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 1/5</b>
<b>Fecha de inicio: 19/02/2012</b>	<b>Fecha de fin: 25/02/2012</b>
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Modificar Municipios	

Tabla 73: Tarea de programación Modificar Municipios

<b>Tarea de Programación</b>	
<b>Número tarea: 16</b>	<b>Número historia:5</b>
<b>Nombre de la tarea:</b> Eliminar Municipios	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 1/5</b>
<b>Fecha de inicio: 19/02/2012</b>	<b>Fecha de fin: 25/02/2012</b>
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Municipios	

Tabla 74: Tarea de programación Eliminar Municipios

<b>Tarea de Programación</b>	
<b>Número tarea: 17</b>	<b>Número historia:5</b>
<b>Nombre de la tarea: Mostrar Listado de Municipios</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 1/5</b>
<b>Fecha de inicio: 19/02/2012</b>	<b>Fecha de fin: 25/02/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción: Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Municipios</b>	

Tabla 75: Tarea de programación Listado de Municipios

<b>Tarea de Programación</b>	
<b>Número tarea: 18</b>	<b>Número historia:6</b>
<b>Nombre de la tarea: Insertar Profesores</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 3/5</b>
<b>Fecha de inicio: 26/02/2012</b>	<b>Fecha de fin: 03/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción: Esta tarea es realizada por el Administrador y la misma facilita Insertar Profesores</b>	

Tabla 76: Tarea de programación Insertar Profesor

<b>Tarea de Programación</b>	
<b>Número tarea: 19</b>	<b>Número historia:6</b>
<b>Nombre de la tarea: Modificar Profesores</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 3/5</b>
<b>Fecha de inicio: 26/02/2012</b>	<b>Fecha de fin: 03/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción: Esta tarea es realizada por el Administrador y la misma facilita Modificar Profesores</b>	

Tabla 77: Tarea de programación Modificar Profesor

<b>Tarea de Programación</b>	
<b>Número tarea: 20</b>	<b>Número historia:6</b>
<b>Nombre de la tarea: Eliminar Profesores</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 3/5</b>

<b>Fecha de inicio: 26/02/2012</b>	<b>Fecha de fin: 03/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Profesores	

Tabla 78: Tarea de programación Eliminar Profesor

<b>Tarea de Programación</b>	
<b>Número tarea: 21</b>	<b>Número historia:6</b>
<b>Nombre de la tarea:</b> Mostrar Listado de Profesores	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 3/5</b>
<b>Fecha de inicio: 26/02/2012</b>	<b>Fecha de fin: 03/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Profesores	

Tabla 79: Tarea de programación Listado de Profesores

## Segunda Iteración

<b>Tarea de Programación</b>	
<b>Número tarea: 22</b>	<b>Número historia:7</b>
<b>Nombre de la tarea:</b> Insertar Coordinador	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha de inicio: 04/03/2012</b>	<b>Fecha de fin: 15/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Insertar Coordinadores	

Tabla 80: Tarea de programación Insertar Coordinador

<b>Tarea de Programación</b>	
<b>Número tarea: 23</b>	<b>Número historia:7</b>
<b>Nombre de la tarea:</b> Modificar Coordinador	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha de inicio: 04/03/2012</b>	<b>Fecha de fin: 15/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	

**Descripción:** Esta tarea es realizada por el Administrador y la misma facilita Modificar Coordinadores.

Tabla 81: Tarea de programación Modificar Coordinador

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 24	<b>Número historia:</b> 7
<b>Nombre de la tarea:</b> Eliminar Coordinador	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 04/03/2012	<b>Fecha de fin:</b> 15/03/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Coordinadores	

Tabla 82: Tarea de programación Eliminar Coordinador

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 25	<b>Número historia:</b> 7
<b>Nombre de la tarea:</b> Mostrar Listado de Coordinadores	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 04/03/2012	<b>Fecha de fin:</b> 15/03/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Coordinadores	

Tabla 83: Tarea de programación Listado Coordinador

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 26	<b>Número historia:</b> 7
<b>Nombre de la tarea:</b> Insertar Administrador	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 04/03/2012	<b>Fecha de fin:</b> 15/03/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Insertar Administrador.	

Tabla 84: Tarea de programación Insertar Administrador

<b>Tarea de Programación</b>
------------------------------

<b>Número tarea: 27</b>	<b>Número historia:7</b>
<b>Nombre de la tarea: Eliminar Administrador</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 1</b>
<b>Fecha de inicio: 04/03/2012</b>	<b>Fecha de fin: 15/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Administradores.	

Tabla 85: Tarea de programación Eliminar Administrador

<b>Tarea de Programación</b>	
<b>Número tarea: 28</b>	<b>Número historia:7</b>
<b>Nombre de la tarea: Mostrar Listado de Administradores</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 1</b>
<b>Fecha de inicio: 04/03/2012</b>	<b>Fecha de fin: 15/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Administradores.	

Tabla 86: Tarea de programación Listado de Administradores

<b>Tarea de Programación</b>	
<b>Número tarea: 29</b>	<b>Número historia:8</b>
<b>Nombre de la tarea: Insertar Estudiantes</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 2/5</b>
<b>Fecha de inicio: 16/03/2012</b>	<b>Fecha de fin: 25/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Insertar Estudiantes	

Tabla 87: Tarea de programación Insertar estudiantes

<b>Tarea de Programación</b>	
<b>Número tarea: 30</b>	<b>Número historia:8</b>
<b>Nombre de la tarea: Modificar estudiantes</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 2/5</b>
<b>Fecha de inicio: 16/03/2012</b>	<b>Fecha de fin: 25/03/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	

**Descripción:** Esta tarea es realizada por el Administrador y la misma facilita Modificar estudiantes

Tabla 88: Tarea de programación Modificar estudiantes

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 31	<b>Número historia:</b> 8
<b>Nombre de la tarea:</b> Eliminar Estudiantes	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 16/03/2012	<b>Fecha de fin:</b> 25/03/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Estudiantes	

Tabla 89: Tarea de programación Eliminar estudiantes

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 32	<b>Número historia:</b> 8
<b>Nombre de la tarea:</b> Mostrar Listado de Estudiantes	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 16/03/2012	<b>Fecha de fin:</b> 25/03/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Estudiantes	

Tabla 90: Tarea de programación Listado de estudiantes

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 33	<b>Número historia:</b> 9
<b>Nombre de la tarea:</b> Insertar Asignaturas	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 25/03/2012	<b>Fecha de fin:</b> 02/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Insertar Asignaturas	

Tabla 91: Tarea de programación Insertar asignatura

<b>Tarea de Programación</b>
------------------------------

<b>Número tarea:</b> 34	<b>Número historia:</b> 9
<b>Nombre de la tarea:</b> Modificar Asignaturas	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 25/03/2012	<b>Fecha de fin:</b> 02/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Modificar Asignaturas	

Tabla 92: Tarea de programación Modificar asignatura

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 35	<b>Número historia:</b> 9
<b>Nombre de la tarea:</b> Eliminar Asignaturas	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 25/03/2012	<b>Fecha de fin:</b> 02/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Asignaturas	

Tabla 93: Tarea de programación Eliminar asignatura

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 36	<b>Número historia:</b> 9
<b>Nombre de la tarea:</b> Mostrar Listado de Asignaturas	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 25/03/2012	<b>Fecha de fin:</b> 02/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Asignaturas	

Tabla 94: Tarea de programación Listado de asignaturas

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 37	<b>Número historia:</b> 10
<b>Nombre de la tarea:</b> Insertar Claustro	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 02/04/2012	<b>Fecha de fin:</b> 10/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	



**Descripción:** Esta tarea es realizada por el Administrador y la misma facilita Insertar Claustros

Tabla 95: Tarea de programación Insertar claustro

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 38	<b>Número historia:</b> 10
<b>Nombre de la tarea:</b> Mostrar Listado de Claustros	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 02/04/2012	<b>Fecha de fin:</b> 10/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Claustros	

Tabla 96: Tarea de programación Listado de claustros

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 39	<b>Número historia:</b> 10
<b>Nombre de la tarea:</b> Modificar Claustro	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 02/04/2012	<b>Fecha de fin:</b> 10/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Modificar Claustros.	

Tabla 97: Tarea de programación Modificar claustro

### Tercera Iteración

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 33	<b>Número historia:</b> 11
<b>Nombre de la tarea:</b> Salvar Fin de Curso	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 10/04/2012	<b>Fecha de fin:</b> 17/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Salvar Fin de Curso.	

Tabla 98: Tarea de programación Salvar fin de curso

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 34	<b>Número historia:</b> 12
<b>Nombre de la tarea:</b> Cambiar Año	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 17/04/2012	<b>Fecha de fin:</b> 24/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Cambiar Año.	

Tabla 99: Tarea de programación Cambiar Año

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 35	<b>Número historia:</b> 13
<b>Nombre de la tarea:</b> Insertar Semestre	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 24/04/2012	<b>Fecha de fin:</b> 27/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Insertar Semestre	

Tabla 100: Tarea de programación Insertar semestre

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 36	<b>Número historia:</b> 13
<b>Nombre de la tarea:</b> Modificar Semestre	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 24/04/2012	<b>Fecha de fin:</b> 27/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Modificar Semestre	

Tabla 101: Tarea de programación Modificar semestre

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 37	<b>Número historia:</b> 13
<b>Nombre de la tarea:</b> Eliminar Semestre	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 24/04/2012	<b>Fecha de fin:</b> 27/04/2012

<b>Programador responsable:</b> Yadiris Martínez Cordero
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Semestre

Tabla 102: Tarea de programación Eliminar semestre

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 38	<b>Número historia:</b> 13
<b>Nombre de la tarea:</b> Mostrar Listado de Semestres	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 24/04/2012	<b>Fecha de fin:</b> 27/04/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Semestres	

Tabla 103: Tarea de programación Listado de semestre

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 39	<b>Número historia:</b> 14
<b>Nombre de la tarea:</b> Insertar Fecha de Cortes	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 27/04/2012	<b>Fecha de fin:</b> 02/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Insertar Fecha de Cortes	

Tabla 104: Tarea de programación insertar fechas de cortes

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 40	<b>Número historia:</b> 14
<b>Nombre de la tarea:</b> Modificar Fecha de Cortes	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 27/04/2012	<b>Fecha de fin:</b> 02/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita modificar Fecha de Cortes	

Tabla 105: Tarea de programación Modificar fechas de cortes

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 41	<b>Número historia:</b> 14
<b>Nombre de la tarea:</b> Mostrar Listado de Cortes	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 27/04/2012	<b>Fecha de fin:</b> 02/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Cortes.	

Tabla 106: Tarea de programación Listado de fechas de cortes

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 42	<b>Número historia:</b> 14
<b>Nombre de la tarea:</b> Eliminar Corte	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 27/04/2012	<b>Fecha de fin:</b> 02/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Eliminar Corte	

Tabla 107: Tarea de programación Eliminar corte

## Cuarta Iteración

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 43	<b>Número historia:</b> 15
<b>Nombre de la tarea:</b> Insertar Evaluación	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 02/05/2012	<b>Fecha de fin:</b> 09/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Insertar Evaluación	

Tabla 108: Tarea de programación Insertar evaluación

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 44	<b>Número historia:</b> 15
<b>Nombre de la tarea:</b> Modificar Evaluación	

<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 02/05/2012	<b>Fecha de fin:</b> 09/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita modificar Evaluación	

Tabla 109: Tarea de programación Modificar evaluación

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 45	<b>Número historia:</b> 15
<b>Nombre de la tarea:</b> Mostrar listado de Evaluaciones	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 02/05/2012	<b>Fecha de fin:</b> 09/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Evaluaciones.	

Tabla 110: Tarea de programación Listado de evaluaciones

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 46	<b>Número historia:</b> 16
<b>Nombre de la tarea:</b> Exportar reporte de Evaluaciones	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 09/05/2012	<b>Fecha de fin:</b> 12/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita hacer reporte de evaluaciones.	

Tabla 111: Tarea de programación exportar evaluaciones

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 47	<b>Número historia:</b> 17
<b>Nombre de la tarea:</b> Exportar listado de Estudiantes	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 12/05/2012	<b>Fecha de fin:</b> 16/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	

**Descripción:** Esta tarea es realizada por el Administrador y la misma facilita hacer reporte del listado estudiantes.

Tabla 112: Tarea de programación exportar listado de estudiantes

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 48	<b>Número historia:</b> 17
<b>Nombre de la tarea:</b> Exportar listado de profesores	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 12/05/2012	<b>Fecha de fin:</b> 16/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita hacer reporte del listado de profesores.	

Tabla 113: Tarea de programación exportar listado de profesores

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 49	<b>Número historia:</b> 17
<b>Nombre de la tarea:</b> Exportar listado de coordinadores	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 12/05/2012	<b>Fecha de fin:</b> 16/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita hacer reporte del listado de coordinadores.	

Tabla 114: Tarea de programación exportar listado de coordinadores

### Quinta Iteración

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 50	<b>Número historia:</b> 18
<b>Nombre de la tarea:</b> Subir Documentación	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3/5
<b>Fecha de inicio:</b> 17/05/2012	<b>Fecha de fin:</b> 20/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Subir Documentación	

Tabla 115: Tarea de programación subir documentos

<b>Tarea de Programación</b>	
<b>Número tarea: 51</b>	<b>Número historia:18</b>
<b>Nombre de la tarea: Eliminar Documentación</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 3/5</b>
<b>Fecha de inicio: 17/05/2012</b>	<b>Fecha de fin: 20/05/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita eliminar Documentación.	

Tabla 116: Tarea de programación eliminar documentos

<b>Tarea de Programación</b>	
<b>Número tarea: 52</b>	<b>Número historia:18</b>
<b>Nombre de la tarea: Mostrar Listado de Documentación</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 3/5</b>
<b>Fecha de inicio: 17/05/2012</b>	<b>Fecha de fin: 20/05/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Documentación	

Tabla 117: Tarea de programación listado de documentos

<b>Tarea de Programación</b>	
<b>Número tarea: 53</b>	<b>Número historia:19</b>
<b>Nombre de la tarea: Insertar Informaciones</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 2/5</b>
<b>Fecha de inicio: 21/05/2012</b>	<b>Fecha de fin: 25/05/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Insertar Informaciones	

Tabla 118: Tarea de programación insertar informaciones

<b>Tarea de Programación</b>	
<b>Número tarea: 54</b>	<b>Número historia:19</b>
<b>Nombre de la tarea: Modificar Informaciones</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 2/5</b>
<b>Fecha de inicio: 21/05/2012</b>	<b>Fecha de fin: 25/05/2012</b>

<b>Programador responsable:</b> Yadiris Martínez Cordero
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Modificar Informaciones.

Tabla 119: Tarea de programación modificar informaciones

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 55	<b>Número historia:</b> 19
<b>Nombre de la tarea:</b> Eliminar Informaciones	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 21/05/2012	<b>Fecha de fin:</b> 25/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita eliminar Informaciones.	

Tabla 120: Tarea de programación eliminar informaciones

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 56	<b>Número historia:</b> 19
<b>Nombre de la tarea:</b> Mostrar Listado de Informaciones	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2/5
<b>Fecha de inicio:</b> 21/05/2012	<b>Fecha de fin:</b> 25/05/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Mostrar Listado de Informaciones	

Tabla 121: Tarea de programación listado de informaciones

<b>Tarea de Programación</b>	
<b>Número tarea:</b> 57	<b>Número historia:</b> 20
<b>Nombre de la tarea:</b> Buscar Estudiantes	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha de inicio:</b> 25/05/2012	<b>Fecha de fin:</b> 02/06/2012
<b>Programador responsable:</b> Yadiris Martínez Cordero	
<b>Descripción:</b> Esta tarea es realizada por el Administrador y la misma facilita Buscar Estudiantes.	

Tabla 122: Tarea de programación buscar estudiantes



<b>Tarea de Programación</b>	
<b>Número tarea: 58</b>	<b>Número historia:20</b>
<b>Nombre de la tarea: Buscar Profesores.</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 1</b>
<b>Fecha de inicio: 25/05/2012</b>	<b>Fecha de fin: 02/06/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción: Esta tarea es realizada por el Administrador y la misma facilita Buscar Profesores.</b>	

Tabla 123: Tarea de programación buscar profesores

<b>Tarea de Programación</b>	
<b>Número tarea: 59</b>	<b>Número historia:20</b>
<b>Nombre de la tarea: Buscar Coordinadores</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 1</b>
<b>Fecha de inicio: 25/05/2012</b>	<b>Fecha de fin: 02/06/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción: Esta tarea es realizada por el Administrador y la misma facilita Buscar Estudiantes.</b>	

Tabla 124: Tarea de programación buscar coordinadores

<b>Tarea de Programación</b>	
<b>Número tarea: 60</b>	<b>Número historia:20</b>
<b>Nombre de la tarea: Buscar Evaluaciones</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 1</b>
<b>Fecha de inicio: 25/05/2012</b>	<b>Fecha de fin: 02/06/2012</b>
<b>Programador responsable: Yadiris Martínez Cordero</b>	
<b>Descripción: Esta tarea es realizada por el Administrador y la misma facilita Buscar Evaluaciones.</b>	

Tabla 125: Tarea de programación buscar evaluaciones

<b>Tarea de Programación</b>	
<b>Número tarea: 61</b>	<b>Número historia:21</b>
<b>Nombre de la tarea: Cambiar Contraseña</b>	
<b>Tipo de tarea: Desarrollo</b>	<b>Puntos estimados: 1/5</b>
<b>Fecha de inicio: 03/06/2012</b>	<b>Fecha de fin: 10/06/2012</b>

**Programador responsable:** Yadiris Martínez Cordero

**Descripción:** Cambiar clave de acceso al sitio.

Tabla 126: Tarea de programación cambiar contraseña

## Anexo 5 Planillas Caso de Prueba de Aceptación

Prueba de aceptación
<b>HU:</b> Autenticar usuarios.
<b>Nombre:</b> Autenticación de usuarios.
<b>Descripción:</b> Validación de entrada de los datos de los usuarios.
<b>Condiciones de ejecución:</b> El usuario debe introducir su nombre de usuario y contraseña.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe su nombre de usuario y contraseña y luego da clic en el botón Entrar.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si el usuario tiene acceso para entrar a la aplicación e inserta sus datos correctamente entrará sin problemas al Sistema.</li> </ul> <p>Se emite un mensaje de error en caso de que:</p> <ul style="list-style-type: none"> <li>• Se inserte los datos de un usuario no válido para el Sistema o incorrectos. (Ya sea su nombre de usuario o su contraseña).</li> <li>• Se dé clic en el botón Entrar sin insertar nada en los campos de texto.</li> </ul>
<b>Evaluación de la prueba:</b> Aceptada

Tabla 127: PA Autenticar usuarios del sistema.

Prueba de aceptación
<b>HU:</b> Cambiar Contraseña.
<b>Nombre:</b> Cambio de Clave.
<b>Descripción:</b> Validación del cambio de contraseña de los usuarios del sistema.

<b>Condiciones de ejecución:</b> El usuario debe introducir su nombre de usuario y contraseña vieja, así como la nueva dos veces, las cuales deben coincidir.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe su nombre de usuario y contraseña vieja y la vieja dos veces y luego da clic en el botón Entrar.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si el usuario entra los datos correctamente se cambia su contraseña.</li> </ul> <p>Se emite un mensaje de error en caso de que:</p> <ul style="list-style-type: none"> <li>• Se inserte los datos de un usuario no válido para el Sistema o incorrectos. (Ya sea su nombre de usuario o su contraseña).</li> <li>• Se dé clic en el botón Entrar sin insertar nada en los campos de texto.</li> <li>• Las contraseñas nuevas no coincidan.</li> </ul>
<b>Evaluación de la prueba:</b> Aceptada

Tabla 128: PA cambiar contraseña

Prueba de aceptación
<b>HU:</b> Gestionar Facultad.
<b>Nombre:</b> Gestión de facultades.
<b>Descripción:</b> Validación de la gestión de facultades.
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevas facultades, modificarlas, eliminarlas, así como listarlas.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar una nueva facultad, luego presiona el botón Insertar. Después que la información esté insertada es que se puede

eliminar, modificar y acceder a ella en el listado de facultades.

**Resultado esperado:**

- Si se insertó la información correctamente muestra un mensaje de confirmación, Facultad insertada correctamente.
- Cuando se elimina se emite otro mensaje de confirmación

**"Se ha eliminado la facultad".**

- Cuando se modifica una facultad, se emite un mensaje de confirmación.

**Se emite un mensaje de error en caso de que:**

- Falten datos de la facultad a la hora de insertarla o modificarla.
- Se inserte una facultad que ya exista.
- Se quiera modificar una facultad por otra, y esta última ya exista.

**Evaluación de la prueba: Aceptada**

Tabla 129: PA Gestionar Facultad

Prueba de aceptación
<b>HU: Gestionar Carrera.</b>
<b>Nombre: Gestión de carreras.</b>
<b>Descripción: Validación de la gestión de carreras.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevas carreras, modificarlas, eliminarlas, así como listarlas.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar una carrera, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de carreras.

**Resultado esperado:**

- Si se insertó la información correctamente muestra un mensaje de confirmación, Carrera insertada correctamente.
- Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado la carrera”.
- Cuando se modifica una carrera, se emite un mensaje de confirmación.

**Se emite un mensaje de error en caso de que:**

- Falten datos de la carrera a la hora de insertarla o modificarla.
- Se inserte una carrera que ya exista.
- Se quiera modificar una carrera por otra, y esta última ya exista.

**Evaluación de la prueba: Aceptada**

Tabla 130: PA Gestionar Carrera

Prueba de aceptación
<b>HU: Gestionar Provincia.</b>
<b>Nombre: Gestión de provincias.</b>
<b>Descripción: Validación de la gestión de provincias.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevas provincias, modificarlas, eliminarlas, así como listarlas.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar una nueva provincia, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de provincias.
<b>Resultado esperado:</b>

- Si se insertó la información correctamente muestra un mensaje de confirmación, Provincia insertada correctamente.
- Cuando se elimina se emite otro mensaje de confirmación

”Se ha eliminado la provincia”.

- Cuando se modifica una provincia, se emite un mensaje de confirmación.

Se emite un mensaje de error en caso de que:

- Falten datos de la provincia a la hora de insertarla o modificarla.
- Se inserte una provincia que ya exista.
- Se quiera modificar una provincia por otra, y esta última ya exista.

**Evaluación de la prueba: Aceptada**

Tabla 131: PA Gestionar Provincia

Prueba de aceptación
<b>HU: Gestionar Municipio.</b>
<b>Nombre: Gestión de municipios.</b>
<b>Descripción: Validación de la gestión de municipios.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevos municipios, modificarlos, eliminarlos, así como listarlos.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se piden para poder insertar un municipio, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a él en el listado de municipios.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de</li> </ul>

<p>confirmación, municipio insertado correctamente.</p> <ul style="list-style-type: none"> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado el municipio”.</li> <li>• Cuando se modifica un municipio, se emite un mensaje de confirmación.</li> </ul> <p>Se emite un mensaje de error en caso de que:</p> <ul style="list-style-type: none"> <li>• Falten datos del municipio a la hora de insertarlo o modificarlo.</li> <li>• Se inserte un municipio que ya exista.</li> <li>• Se quiera modificar un municipio por otro, y este última ya exista.</li> </ul> <p>Evaluación de la prueba: Aceptada</p>
--

Tabla 132: PA Gestionar Municipio

Prueba de aceptación
<b>HU: Gestionar Profesor.</b>
<b>Nombre: Gestión de Profesores.</b>
<b>Descripción: Validación de la gestión de profesores.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevos profesores, modificarlos, eliminarlos, así como listarlos.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar un nuevo profesor, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de profesores.
<p><b>Resultado esperado:</b></p> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, Profesor insertado correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha</li> </ul>

<p>eliminado el profesor”.</p> <ul style="list-style-type: none"> <li>• Cuando se modifica un profesor, se emite un mensaje de confirmación.</li> </ul> <p>Se emite un mensaje de error en caso de que:</p> <ul style="list-style-type: none"> <li>• Falten datos del profesor a la hora de insertarlo o modificarlo.</li> <li>• Se inserte un profesor que ya exista.</li> <li>• Se quiera modificar un profesor por otro, y este último ya exista.</li> </ul>
Evaluación de la prueba: Aceptada

Tabla 133: PA Gestionar Profesor

Prueba de aceptación
<b>HU: Gestionar Rol.</b>
<b>Nombre: Gestión de roles.</b>
<b>Descripción: Validación de la gestión de roles.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevos roles (coordinadores o administradores), modificarlos, eliminarlos, así como listarlos.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar un nuevo coordinador, luego presiona el botón Insertar o selecciona el profesor que será administrador del sitio. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, coordinador o administrador insertado correctamente.</li> </ul>



- Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado correctamente”.
- Cuando se modifica un coordinador, se emite un mensaje de confirmación.

Se emite un mensaje de error en caso de que:

- Falten datos del coordinador a la hora de insertarlo o modificarlo.
- Se inserte un coordinador que ya exista.
- Se quiera modificar un coordinador por otro, y este último ya exista.

Evaluación de la prueba: Aceptada

Tabla 134: PA Gestionar Rol

<b>Prueba de aceptación</b>	
<b>HU: Gestionar Estudiante.</b>	
<b>Nombre: Gestión de estudiantes.</b>	
<b>Descripción: Validación de la gestión de estudiantes.</b>	
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevos estudiantes, modificarlos, eliminarlos, así como listarlos.	
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar un nuevo estudiante, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de estudiantes.	
<b>Resultado esperado:</b>	
<ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, estudiante insertado correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación</li> </ul>	

**“Se ha eliminado el estudiante”.**

- Cuando se modifica un estudiante, se emite un mensaje de confirmación.

**Se emite un mensaje de error en caso de que:**

- Falten datos del estudiante a la hora de insertarlo o modificarlo.
- Se inserte un estudiante que ya exista.
- Se quiera modificar un estudiante por otro, y este último ya exista.

**Evaluación de la prueba: Aceptada**

Tabla 135: PA Gestionar Estudiante

Prueba de aceptación
<b>HU: Gestionar Asignatura.</b>
<b>Nombre: Gestión de asignaturas.</b>
<b>Descripción: Validación de la gestión de asignaturas.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador o Coordinador para poder insertar nuevas asignaturas, modificarlas, eliminarlas, así como listarlas.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar una nueva asignatura, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de asignaturas.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, asignatura insertada correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado la asignatura”.</li> </ul>

- Cuando se modifica una asignatura, se emite un mensaje de confirmación.

Se emite un mensaje de error en caso de que:

- Falten datos de la asignatura a la hora de insertarla o modificarla.
- Se inserte una asignatura que ya exista.
- Se quiera modificar una asignatura por otro, y esta última ya exista.

**Evaluación de la prueba: Aceptada**

Tabla 136: PA Gestionar Asignatura

Prueba de aceptación
<b>HU: Gestionar Semestre.</b>
<b>Nombre: Gestión de semestres.</b>
<b>Descripción: Validación de la gestión de semestres.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador o Coordinador para poder insertar nuevos semestres, modificarlos, eliminarlos, así como listarlos.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se piden para poder insertar un nuevo semestre, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar y modificar.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, semestre insertado correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado el semestre”.</li> <li>• Cuando se modifica un semestre, se emite un mensaje de confirmación.</li> </ul>

Se emite un mensaje de error en caso de que:

- Falten datos del semestre a la hora de insertarlo o modificarlo.
- Se inserte un semestre que ya exista.
- Se quiera modificar un semestre por otro, y este último ya exista.

Evaluación de la prueba: Aceptada

Tabla 137: PA Gestionar Semestre

Prueba de aceptación
<b>HU: Gestionar Corte.</b>
<b>Nombre: Gestión de cortes.</b>
<b>Descripción: Validación de la gestión de cortes.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador o Coordinador para poder insertar nuevos cortes, modificarlos, eliminarlos, así como listarlos.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se piden para poder insertar un nuevo cortes, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar y modificar.
<p><b>Resultado esperado:</b></p> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, corte insertado correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado el corte”.</li> <li>• Cuando se modifica un corte, se emite un mensaje de confirmación.</li> </ul> <p>Se emite un mensaje de error en caso de que:</p>

- Falten datos del corte a la hora de insertarlo o modificarlo.
- Se inserte un corte que ya exista.
- Se quiera modificar un corte por otro, y este último ya exista.

**Evaluación de la prueba: Aceptada**

Tabla 138: PA Gestionar Corte

Prueba de aceptación
<b>HU: Gestionar Documentación.</b>
<b>Nombre: Gestión de documentos.</b>
<b>Descripción: Validación de la gestión de documentos.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser usuario del sistema para poder insertar nuevos documentos, modificarlos, eliminarlos, así como listarlos.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar un nuevo documento, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en una lista de documentación.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, documento insertado correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado el documento”.</li> <li>• Cuando se modifica un documento, se emite un mensaje de confirmación.</li> </ul> <p><b>Se emite un mensaje de error en caso de que:</b></p> <ul style="list-style-type: none"> <li>• Falten datos del documento a la hora de insertarlo o modificarlo.</li> </ul>

- Se inserte un documento que ya exista.
- Se quiera modificar un documento por otro, y este último ya exista.

**Evaluación de la prueba: Aceptada**

Tabla 139: PA Gestionar Documentación

Prueba de aceptación
<b>HU: Gestionar Fecha de cortes.</b>
<b>Nombre: Gestión de fechas de cortes.</b>
<b>Descripción: Validación de la gestión de fechas de cortes.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador o Coordinador para poder insertar fechas de cortes, modificarlas, eliminarlos, así como listarlas.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se piden para poder insertar la fecha de cortes, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de fechas de cortes.
<p><b>Resultado esperado:</b></p> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, fecha de cortes insertado correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado la fecha de cortes”.</li> <li>• Cuando se modifica una fecha de cortes, se emite un mensaje de confirmación.</li> </ul> <p><b>Se emite un mensaje de error en caso de que:</b></p> <ul style="list-style-type: none"> <li>• Falte la fecha de cortes a la hora de insertarla o modificarla.</li> <li>• Se inserte una fecha de cortes que ya exista.</li> </ul>

- Se quiera modificar una fecha de cortes por otra, y esta última ya exista.

**Evaluación de la prueba: Aceptada**

Tabla 140: PA Gestionar Fechas de Cortes

Prueba de aceptación
<b>HU: Gestionar Evaluación.</b>
<b>Nombre: Gestión de evaluaciones.</b>
<b>Descripción: Validación de la gestión de evaluaciones.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación para poder insertar evaluaciones, modificarlas, eliminarlas, así como listarlas.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se piden para poder insertar evaluaciones, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de evaluaciones.
<p><b>Resultado esperado:</b></p> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, evaluaciones insertadas correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado las evaluaciones”.</li> <li>• Cuando se modifica una evaluación, se emite un mensaje de confirmación.</li> </ul> <p><b>Se emite un mensaje de error en caso de que:</b></p> <ul style="list-style-type: none"> <li>• Falten datos de las evaluaciones a la hora de insertarla o modificarla.</li> <li>• Se inserte una evaluación que ya exista.</li> </ul>

Evaluación de la prueba: Aceptada
-----------------------------------

Tabla 141: PA Gestionar Evaluación

Prueba de aceptación
<b>HU:</b> Exportar evaluaciones.
<b>Nombre:</b> Exportar reporte de evaluaciones.
<b>Descripción:</b> Validación de exportar reporte de evaluaciones.
<b>Condiciones de ejecución:</b> Que exista un listado de evaluaciones
<b>Entrada/Pasos de ejecución:</b> El usuario selecciona la lista de evaluaciones y hace el reporte de evaluaciones.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Un documento en formato .xsl con las evaluaciones</li> </ul>
Evaluación de la prueba: Aceptada

Tabla 142: PA Exportar evaluación

Prueba de aceptación
<b>HU:</b> Exportar listados.
<b>Nombre:</b> Exportar listado de estudiantes.
<b>Descripción:</b> Validación de exportar listado de estudiantes.
<b>Condiciones de ejecución:</b> Que exista un listado de estudiantes.
<b>Entrada/Pasos de ejecución:</b> El usuario selecciona la lista de estudiantes y hace el reporte.
<b>Resultado esperado:</b> <p>Un documento en formato .xsl con el listado de estudiantes.</p>
Evaluación de la prueba: Aceptada

Tabla 143: PA Exportar Listado de estudiantes

Prueba de aceptación
<b>HU:</b> Exportar listados.
<b>Nombre:</b> Exportar listado de profesores.
<b>Descripción:</b> Validación de exportar listado de profesores.



<b>Condiciones de ejecución:</b> Que exista un listado de profesores
<b>Entrada/Pasos de ejecución:</b> El usuario selecciona la lista de profesores y hace el reporte.
<b>Resultado esperado:</b>  Un documento en formato .xsl con el listado de profesores.
<b>Evaluación de la prueba:</b> Aceptada

Tabla 144: PA Exportar Listado de profesores

Prueba de aceptación
<b>HU:</b> Exportar listados.
<b>Nombre:</b> Exportar listado de coordinadores.
<b>Descripción:</b> Validación de exportar listado de coordinadores.
<b>Condiciones de ejecución:</b> Que exista un listado de coordinadores.
<b>Entrada/Pasos de ejecución:</b> El usuario selecciona la lista de coordinadores y hace el reporte.
<b>Resultado esperado:</b>  Un documento en formato .xsl con el listado de coordinadores.
<b>Evaluación de la prueba:</b> Aceptada

Tabla 145: PA Exportar Listado de coordinadores

Prueba de aceptación
<b>HU:</b> Buscar estudiantes.
<b>Nombre:</b> Buscar estudiantes.
<b>Descripción:</b> Validación de buscar estudiantes.
<b>Condiciones de ejecución:</b> El usuario busca estudiantes dado su carnet de identidad o carrera-año.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe el CI del estudiante o selecciona carrera-año.

**Resultado esperado:**

- Se muestra el o los estudiantes que coincidan con los parámetros de búsqueda.

**Se emite un mensaje de error en caso de que:**

- No se seleccionen todos los parámetros de búsqueda.

**Evaluación de la prueba: Aceptada**

Tabla 146: PA Buscar Estudiantes

Prueba de aceptación
<b>HU: Realizar Búsquedas.</b>
<b>Nombre: Buscar profesores.</b>
<b>Descripción: Validación de buscar profesores.</b>
<b>Condiciones de ejecución: El usuario busca profesores dado su carnet de identidad o carrera-año-asignatura.</b>
<b>Entrada/Pasos de ejecución: El usuario escribe el CI del profesor o selecciona carrera-año-asignatura.</b>
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Se muestra el o los profesores que coincidan con los parámetros de búsqueda.</li> </ul> <b>Se emite un mensaje de error en caso de que:</b> <ul style="list-style-type: none"> <li>• No se seleccionen todos los parámetros de búsqueda.</li> </ul>
<b>Evaluación de la prueba: Aceptada</b>

Tabla 147: PA Buscar Profesores

Prueba de aceptación
<b>HU: Realizar Búsquedas.</b>
<b>Nombre: Buscar coordinadores.</b>
<b>Descripción: Validación de buscar coordinadores.</b>

<b>Condiciones de ejecución:</b> El usuario busca coordinadores dado su carnet de identidad o carrera-año.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe el CI del coordinador o selecciona carrera-año.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Se muestra el o los coordinadores que coincidan con los parámetros de búsqueda.</li> </ul> <p>Se emite un mensaje de error en caso de que:</p> <ul style="list-style-type: none"> <li>• No se seleccionen todos los parámetros de búsqueda.</li> </ul>
<b>Evaluación de la prueba:</b> Aceptada

Tabla 148: PA Buscar Coordinadores

Prueba de aceptación
<b>HU:</b> Realizar Búsquedas.
<b>Nombre:</b> Buscar Notas.
<b>Descripción:</b> Validación de buscar notas.
<b>Condiciones de ejecución:</b> El usuario busca las notas dado un estudiante específico o un grupo especificando carrera-año.
<b>Entrada/Pasos de ejecución:</b> El usuario selecciona el CI de un estudiante específico o un grupo especificando carrera-año.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Se muestra el o los estudiantes que coincidan con los parámetros de búsqueda.</li> </ul> <p>Se emite un mensaje de error en caso de que:</p> <ul style="list-style-type: none"> <li>• No se seleccionen todos los parámetros de búsqueda.</li> </ul>

**Evaluación de la prueba: Aceptada**

Tabla 149: PA Buscar Notas

Prueba de aceptación
<b>HU: Gestionar Informaciones.</b>
<b>Nombre: Gestión de Noticias.</b>
<b>Descripción: Validación de la gestión de noticias.</b>
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevas noticias, modificarlas, eliminarlas, así como listarlas.
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar una noticia, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de noticias.
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, noticia insertada correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado la noticias”.</li> <li>• Cuando se modifica una noticia, se emite un mensaje de confirmación.</li> </ul> <p><b>Se emite un mensaje de error en caso de que:</b></p> <ul style="list-style-type: none"> <li>• Falten datos de la noticia a la hora de insertarla o modificarla.</li> <li>• Se inserte una noticia que ya exista.</li> <li>• Se quiera modificar una noticia por otra, y esta última ya exista.</li> </ul>
<b>Evaluación de la prueba: Aceptada</b>

Tabla 150: PA Gestionar Noticia

Prueba de aceptación
<b>HU: Salvar Fin de Curso.</b>
<b>Nombre: Salvar Fin de Curso.</b>
<b>Descripción: Validación de salvar fin de curso, guardando los listados de estudiantes con sus notas de cada año.</b>
<b>Condiciones de ejecución: El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder hacer una salva de fin de curso.</b>
<b>Entrada/Pasos de ejecución: El usuario hace la salva de fin de curso guardando las notas de los estudiantes.</b>
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Mensaje de confirmación de que la salva fue hecha correctamente.</li> </ul>
<b>Evaluación de la prueba: Aceptada</b>

Tabla 151: PA Salvar Fin de Curso

Prueba de aceptación
<b>HU: Cambiar Año.</b>
<b>Nombre: Cambiar año.</b>
<b>Descripción: Validación de cambiar de año a los estudiantes que hayan aprobado.</b>
<b>Condiciones de ejecución: El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder hacer el cambio de año.</b>
<b>Entrada/Pasos de ejecución: El usuario hace el cambio de año.</b>
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Mensaje de confirmación de que el cambio de año fue hecho correctamente.</li> </ul>

Evaluación de la prueba: Aceptada

Tabla 152: PA Cambiar Año

<b>Prueba de aceptación</b>	
<b>HU: Gestionar Claustros.</b>	
<b>Nombre: Gestión de Claustros.</b>	
<b>Descripción: Validación de la gestión de Claustros.</b>	
<b>Condiciones de ejecución:</b> El usuario debe de entrar a la Aplicación y el mismo tiene que ser Administrador para poder insertar nuevas Claustros, modificarlos, eliminarlos, así como listarlos.	
<b>Entrada/Pasos de ejecución:</b> El usuario escribe todos los datos que se le piden para poder insertar un Claustro, luego presiona el botón Insertar. Después que la información esté insertada es que se puede eliminar, modificar y acceder a ella en el listado de Claustros.	
<b>Resultado esperado:</b> <ul style="list-style-type: none"> <li>• Si se insertó la información correctamente muestra un mensaje de confirmación, Claustro insertado correctamente.</li> <li>• Cuando se elimina se emite otro mensaje de confirmación “Se ha eliminado el Claustro”.</li> <li>• Cuando se modifica un Claustro, se emite un mensaje de confirmación.</li> </ul>	
<b>Se emite un mensaje de error en caso de que:</b> <ul style="list-style-type: none"> <li>• Falten datos del Claustro a la hora de insertarlo o modificarlo.</li> <li>• Se inserte un Claustros que ya exista.</li> <li>• Se quiera modificar un Claustro por otro, y este último ya exista.</li> </ul>	

Evaluación de la prueba: Aceptada

Tabla 153: PA Gestionar Claustro

## Anexo 6 Modelo Físico de Datos

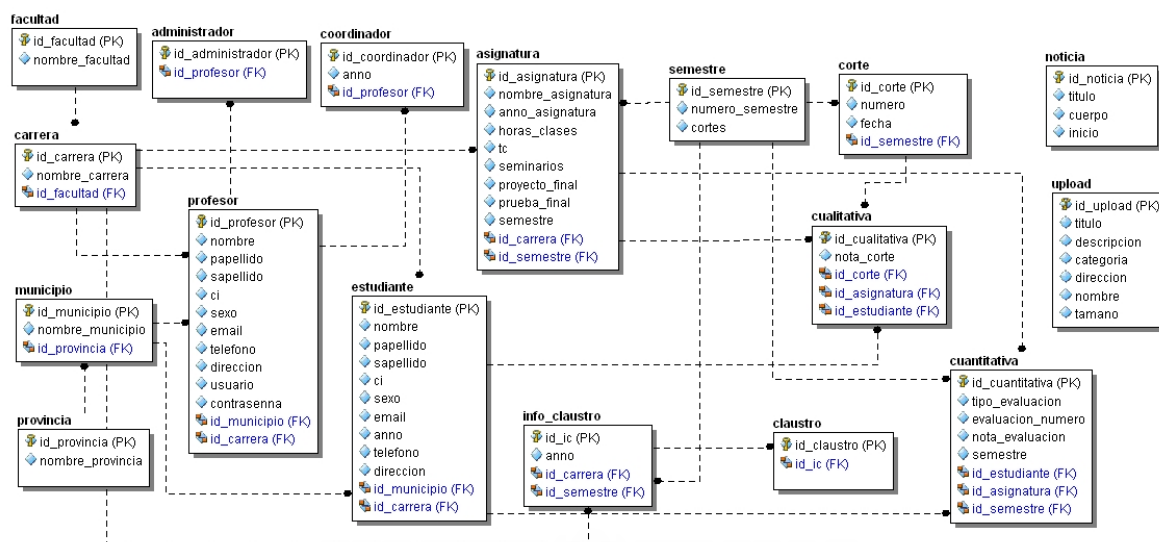


Figura 5: Modelo Físico de Datos

## Anexo 7 Principales Interfaces de la Aplicación

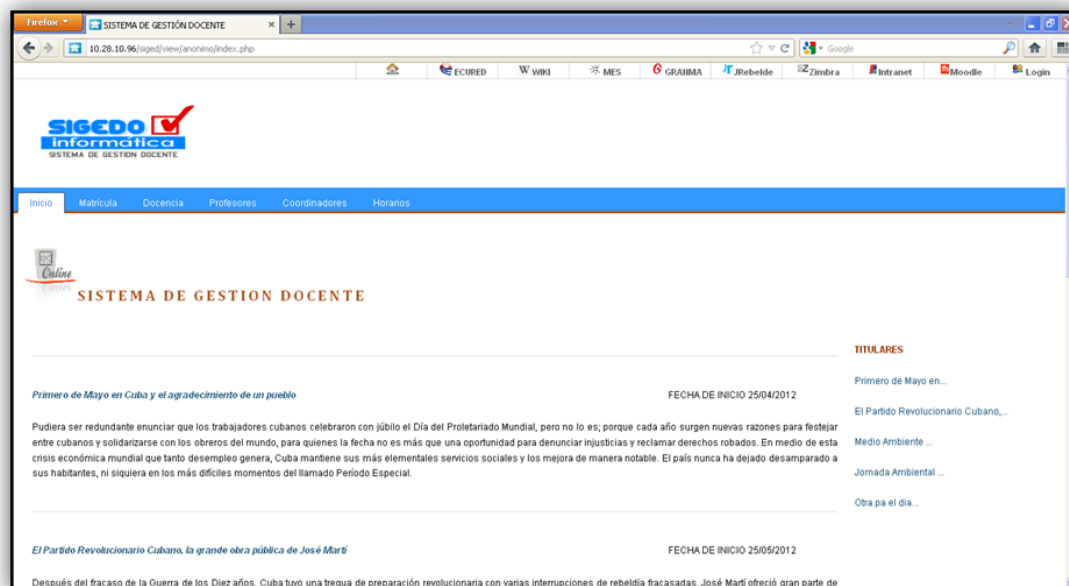


Figura 6: Portada de SIGEDO

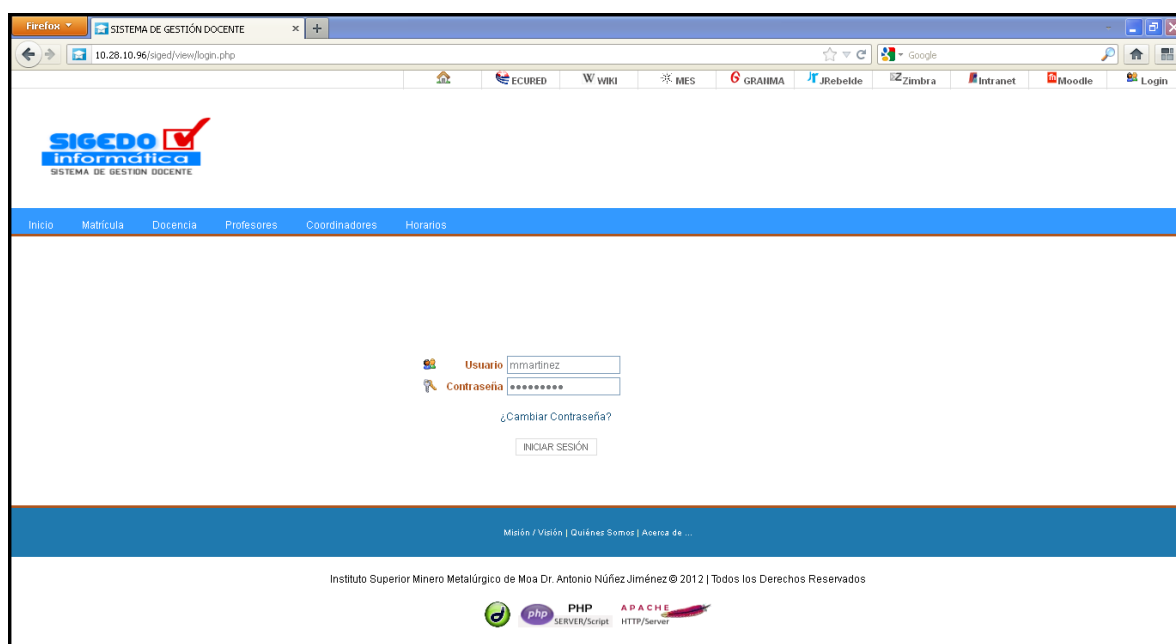


Figura 7: Login SIGEDO

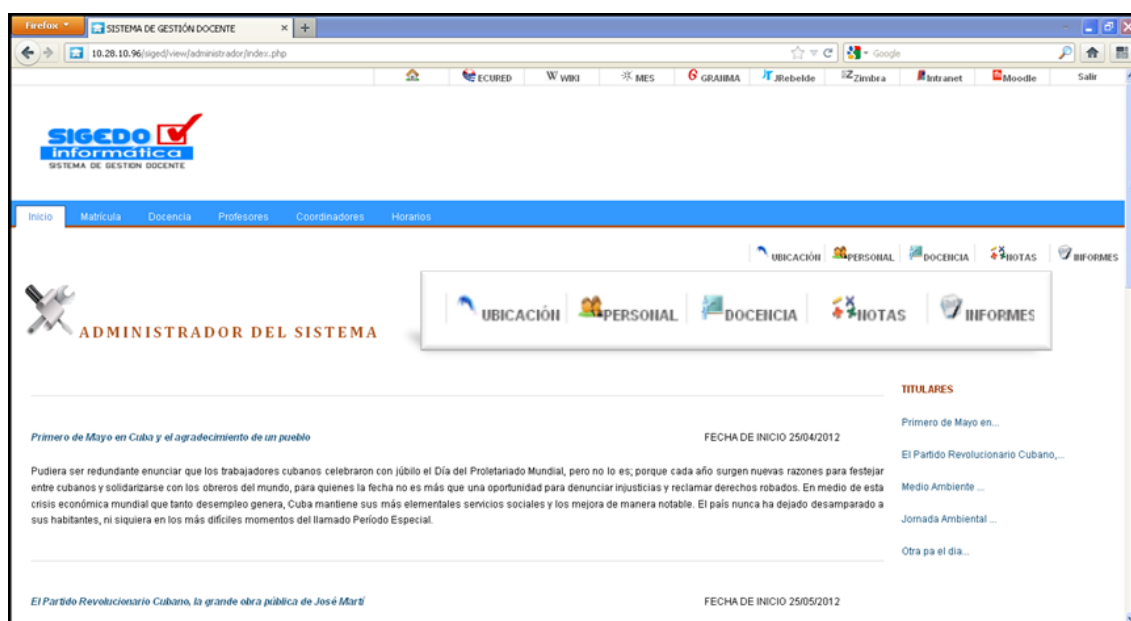


Figura 8: Panel Administrador





## GESTIONAR FACULTAD

*Inserte el Nombre de la Facultad y modifique o elimine en la lista de facultades.*

Nombre de la nueva Facultad

Figura 9: Gestionar Facultad

FACULTADES DEL ISMM

NOMBRE DE LA FACULTAD

Geología - Minas

✖

✎

Existe(n) 1 Facultad(es) en la BD

Figura 10: Listado de Facultades




## MODIFICAR FACULTAD

*Inserte el Nombre de la Facultad y modifique o elimine en la lista de facultades.*

Nombre de la Facultad

Figura 11: Modificar Facultades




## GESTIONAR ROL ADMINISTRADOR / COORDINADOR

Seleccione el o los profesores que formaran parte del grupo de administradores del sitio.

<input type="checkbox"/>	NOMBRE Y APELLIDOS	CI	SEXO	EMAIL	USUARIO	FACULTAD	CARRERA
<input type="checkbox"/>	Roberto San Miguel	5555555555	M	rsanm@ismm.edu.cu	rsanm	Geología - Minas	Informática
<input type="checkbox"/>	Roiky Rodríguez Noa	8888888888	M	rrnoa@ismm.edu.cu	rrnoa	Geología - Minas	Informática
<input type="checkbox"/>	Ariel Montes de Oca Rodríguez	7777777777	M	amontes@ismm.edu.cu	amontes	Geología - Minas	Informática
<input type="checkbox"/>	Djaofera Ambendahy Roja	12312312312	F	djaofera@ismm.edu.cu	djaofera	Geología - Minas	Informática
<input type="checkbox"/>	Eglys Martín Astorga	3333333333	F	emartin@ismm.edu.cu	emartin	Geología - Minas	Informática
<input type="checkbox"/>	Juan Carlos Figueroa Sosa	98798798798	M	jcfigueroa@ismm.edu.cu	jcfigueroa	Geología - Minas	Informática
<input checked="" type="checkbox"/>	Marcos Martínez Rodríguez	86051512105	M	mmartinez@ismm.edu.cu	mmartinez	Geología - Minas	Informática
<input type="checkbox"/>	Mariela Cuba Tamayo	1111111111	F	mcuba@ismm.edu.cu	mcuba	Geología - Minas	Informática
<input type="checkbox"/>	Yelins de la Cruz Pérez	9999999999	F	ycruz@ismm.edu.cu	ycruz	Geología - Minas	Informática

Existe(n) 1 Administrador(es) en la BD

Figura 12: Rol de Administrador



## ESTABLECER CLAUSTRO DE PROFESORES

Conforme el Claustro de Profesores y modifique o elimine en la [lista de claustros](#).

Informática 2		SEMESTRE I
PROFESOR	ASIGNATURA	
Yelins de la Cruz Pérez	Debate Histórico y Contemporáneo	
Eglys Martín Astorga	Economía Política	
Juan Carlos Figueroa Sosa	Educación Física III	
Roiky Rodríguez Noa	Estructura de Datos I	
Yelins de la Cruz Pérez	Ética de la Profesión	
Djaofera Ambendahy Roja	Idioma Inglés	
Ariel Montes de Oca Rodríguez	Introducción a la Inteligencia Artificial	
Mariela Cuba Tamayo	Matemática III	
Roberto San Miguel	Seguridad Nacional	

Figura 13: Establecer claustro

NOMBRE Y APELLIDOS	CI	SEXO	EMAIL	EVALUACIÓN
William Cantillo Terrero	87451245124	M	wcantillo@info.ismm.edu.cu	4 ▼
Carlos M. Fuentes Suárez	8912333654	M	cfuentes@info.ismm.edu.cu	4 ▼
Jesús Espinosa Sánchez	89543014225	M	jespinosa@info.ismm.edu.cu	3 ▼
José Andrés Castillo Díaz	89789462010	M	jcastillo@info.ismm.edu.cu	5 ▼
Marbin Aldana Matos	88211054879	M	maldana@info.ismm.edu.cu	3 ▼
Abel Nicó Quintero	52222222222	M	anico@info.ismm.edu.cu	3 ▼
Alfredo Barallobre Mateo	89512434047	M	amateo@info.ismm.edu.cu	5 ▼
Dayron A. Recio Tamayo	32222222222	M	drecio@info.ismm.edu.cu	4 ▼
Erick Lobaina López	36936936936	M	elobaina@info.ismm.edu.cu	3 ▼
Frank A. González Rodríguez	74174174174	M	fgonzalez@info.ismm.edu.cu	5 ▼
José A. Mendoza Mantilla	92222222222	M	jmendoza@info.ismm.edu.cu	3 ▼
Lisbeth Segura Driggs	21222222222	F	lsegura@info.ismm.edu.cu	4 ▼
Pedro C. Oquendo Calderín	42222222222	M	poquendo@info.ismm.edu.cu	3 ▼
Rafael de J. Millet Leyva	72222222222	M	rmillet@info.ismm.edu.cu	3 ▼
Raidel R. Molina Oro	62222222222	M	rmolina@info.ismm.edu.cu	5 ▼
Rolbis Matos Matos	25825825825	M	rmmatos@info.ismm.edu.cu	4 ▼
Viviana Molina Milán	62222222221	F	vmolina@info.ismm.edu.cu	3 ▼
Yalimilka Taró Ballester	12222222222	F	ytballester@info.ismm.edu.cu	3 ▼
Yerlin Matos Matos	14714714714	F	ymatos@info.ismm.edu.cu	3 ▼
Registrar				
Existen 19 estudiante(s) en este grupo.				

Figura 14: Evaluación cuantitativa