



ESPE

**UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA**

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

**CARRERA DE INGENIERÍA DE SISTEMAS E
INFORMÁTICA**

**PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN
DEL TÍTULO DE INGENIERO DE SISTEMAS E INFORMÁTICA**

**AUTOR: MOROMENACHO CAIZA, DIEGO ARMANDO
SUNTAXI SUNTAXI, SUSANA LUCIA**

**TEMA: DESARROLLO DE UNA APLICACIÓN WEB DE
CONTROL DE ASISTENCIAS PARA PROGRAMAS DE
POSTGRADO, MEDIANTE LA UTILIZACIÓN DE LA
PLATAFORMA JAVA ENTERPRISE EDITION JEE6 WEB
APLICANDO LA METODOLOGÍA SCRUM.**

**DIRECTOR: ING. ARROYO, RUBÉN
CODIRECTOR: ING. PARRA, PABLO**

SANGOLQUÍ, MAYO 2014

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

CERTIFICADO

Ing. Rubén Arroyo (DIRECTOR DE TESIS)

Ing. Pablo Parra (CODIRECTOR DE TESIS)

CERTIFICAN

Que el presente trabajo titulado “DESARROLLO DE UNA APLICACIÓN WEB DE CONTROL DE ASISTENCIAS PARA PROGRAMAS DE POSTGRADO, MEDIANTE LA UTILIZACIÓN DE LA PLATAFORMA JAVA ENTERPRISE EDITION JEE6 WEB APLICANDO LA METODOLOGÍA SCRUM” fue realizado en su totalidad por la Srta. Susana Lucia Sntaxi Sntaxi y el Sr. Diego Armando Moromenacho Caiza, como requerimiento parcial a la obtención del título de INGENIERO DE SISTEMAS E INFORMÁTICA.

Sangolquí, Mayo de 2014

ING. RUBÉN ARROYO
DIRECTOR

ING. PABLO PARRA
CODIRECTOR

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

DECLARACIÓN DE RESPONSABILIDAD

Nosotros, Susana Lucia Sntaxi Sntaxi y Diego Armando Moromenacho Caiza

DECLARAMOS QUE:

El proyecto de grado denominado “DESARROLLO DE UNA APLICACIÓN WEB DE CONTROL DE ASISTENCIAS PARA PROGRAMAS DE POSTGRADO, MEDIANTE LA UTILIZACIÓN DE LA PLATAFORMA JAVA ENTERPRISE EDITION JEE6 WEB APLICANDO LA METODOLOGÍA SCRUM.”, ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan el pie de las páginas correspondiente, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, Mayo de 2014

Susana Lucia Sntaxi Sntaxi
C.C: 1719226209

Diego Armando Moromenacho Caiza
C.C: 1719996496

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE
CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

AUTORIZACIÓN DE PUBLICACIÓN

Nosotros, Susana Lucia Sntaxi Sntaxi y Diego Armando Moromenacho Caiza, autorizamos a la UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE, la publicación, en la biblioteca virtual de la Institución del proyecto de tesis “DESARROLLO DE UNA APLICACIÓN WEB DE CONTROL DE ASISTENCIAS PARA PROGRAMAS DE POSTGRADO, MEDIANTE LA UTILIZACIÓN DE LA PLATAFORMA JAVA ENTERPRISE EDITION JEE6 WEB APLICANDO LA METODOLOGÍA SCRUM.”, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y autoría.

Sangolquí, Mayo de 2014

Susana Lucia Sntaxi Sntaxi

C.C: 1719226209

Diego Armando Moromenacho Caiza

C.C:1719996496

DEDICATORIA

Dedico esta tesis primeramente a Jehová, por haber sido mi padre, mi amigo y mi Dios durante toda mi vida y especialmente por corregirme cuando ha sido necesario hacerlo. Gracias Jehová por haberme sujetado con tu diestra de justicia y haberme dado la fortaleza necesaria para afrontar los obstáculos que se me presentaron en la realización del presente proyecto de tesis.

A mi madre: Luciana Suntaxi, gracias madre por su infinito amor, por haber sido una mujer muy trabajadora y un ejemplo para nosotros sus hijos. Gracias mami por haberme enseñado que todas las metas se pueden cumplir siempre con perseverancia y sacrificio. A ella le dedico esta enorme felicidad, fruto de su esfuerzo constante.

A mi padre: José Suntaxi, por haber sido un pilar muy fundamental en mi vida, por motivarme y darme la mano cuanto sentía que el camino se terminaba.

A mi amiga y maestra espiritual, por haber llegado a mi vida en el momento preciso, para adquirir el conocimiento exacto de la verdad que procede del cielo. Gracias a tus sabios consejos he logrado levantarme en los momentos más difíciles de mi vida, Gracias Myriam y amigos de la congregación Testigos de Jehová.

A mis hermanos Marco, Patricio y Rocío, por sus acertados consejos y por haber sacrificado su tiempo en ayudarme a cumplir esta meta tan anhelada.

A mi sobrinito Dilan, que a tan corta edad, con sus ocurrencias y alegría me ha levantado los ánimos en muchos momentos.

A Janeth, mi cuñada, gracias por haber compartido conversaciones muy interesantes y edificadoras para cumplir esta meta.

Como no recordarte primo y amigo, aunque las distancias nos separen siempre te llevaré en mi corazón, gracias por estar siempre preocupado en mis problemas y por supuesto, gracias por ayudarme a solucionarlos.

SUSANA SUNTAXI

DEDICATORIA

El esfuerzo y sacrificio se lo dedico a toda mi familia, especialmente a mi madre y mis hermanos que a todo momento estuvieron conmigo apoyando y aconsejándome, para poder de esta manera seguir caminando hacia la meta y cumplir todos mis objetivos de la manera más correcta.

DIEGO MOROMENACHO

AGRADECIMIENTO

A Jehová por haberme guiado durante toda mi vida con infinito amor.

A mis padres, que con amor, esfuerzo y sacrificio supieron guiarme para alcanzar esta noble profesión y ser útil a la sociedad.

Mi eterna gratitud a quienes me apoyaron en todo momento, especialmente a mis amigos y hermanos, ya que fueron testigos de mis triunfos y fracasos.

Al Ing. Rubén Arroyo, por habernos dirigido correctamente en la realización del presente proyecto de tesis.

Al Ing. Pablo Parra, por sus consejos y guía para obtener un trabajo de calidad.

A la Ing. Mónica Gómez, que gracias a su experiencia y profesionalismo nos supo guiar y corregir en la etapa final de la tesis, gracias a ella se pudo obtener un trabajo excelente.

A la Universidad de las Fuerzas Armadas – ESPE, de la cual llevo las mejores enseñanzas.

A mi compañero de tesis, Diego gracias por su paciencia y apoyo incondicional.

SUSANA SUNTAXI

AGRADECIMIENTO

Agradezco en primer lugar a Dios por velar a cada momento de mi persona y de mi familia permitiéndonos estar unidos a cada momento para poder afrontar los problemas y resolverlos de la mejor manera.

Gracias a la mujer especial, mi madre Clemencia, que me dio la vida y que es la persona que me demostró a no rendirme para seguir luchando ante las adversidades demostrándome sus valores, amor, coraje y haberme inculcado su maravillosa humildad; por eso y muchas cosas más mi infinito agradecimiento a este ser tan maravilloso que siempre estará en mis pensamientos.

A mi hermano Nelson por estar siempre con nosotros apoyándonos y demostrándonos su esfuerzo para conseguir cosas importantes en la vida, sus buenos consejos me han guiado de forma positiva a seguir superándome profesionalmente y a seguir el ejemplo de ser un buen hermano lleno de pensamientos que van de la mano con el respeto a los demás.

A mi hermana Blanca por su apoyo moral en momentos de dificultad, los cuales han servido enormemente para conseguir este objetivo.

A mi hermano Ramiro, sus consejos de seguir luchando han sido también muy importantes en mi vida estudiantil.

Retribuyo también mis agradecimientos a todos los profesores que he tenido en mi vida estudiantil y que me han guiado educativamente aportándome conocimientos que me serán útiles para ponerlos en práctica en mi vida profesional.

DIEGO MOROMENACHO

CONTENIDO

RESUMEN.....	1
CAPÍTULO 1	1
1.1 INTRODUCCIÓN.....	1
1.2 PLANTEAMIENTO DEL PROBLEMA.....	2
1.3 JUSTIFICACIÓN	2
1.4 ALCANCE	3
1.4.1 <i>Proceso Administrativo - Recursos Humanos</i>	3
1.4.2 <i>Proceso Administrativo - Seguridad</i>	4
1.4.3 <i>Reportes Estadísticos</i>	4
1.5 OBJETIVOS.....	5
1.5.1 <i>General</i>	5
1.5.2 <i>Específicos</i>	5
1.6 METODOLOGÍA.....	5
1.7 RECURSOS	6
1.7.1 <i>Hardware</i>	6
1.7.2 <i>Software</i>	6
1.7.3 <i>Recursos Humanos</i>	7
1.8 FACTIBILIDAD	7
1.8.1 <i>Factibilidad Técnica</i>	7
1.8.2 <i>Factibilidad Operativa</i>	7
1.8.3 <i>Factibilidad Tecnológica</i>	7
CAPÍTULO 2: MARCO TEÓRICO	8
2.1 ARQUITECTURA WEB.....	8
2.1.1 <i>Ventajas y Desventajas:</i>	8
2.2 ARQUITECTURA FÍSICA DE TRES CAPAS	9
2.2.1 <i>Capas de la Arquitectura</i>	10

2.2.2	<i>Ventajas y Desventajas</i>	11
2.3	JAVA JEE	12
2.3.1	<i>Ventajas</i>	12
2.3.2	<i>Estructura de una Aplicación JEE</i>	12
2.4	ENTERPRISE JAVABEANS	13
2.4.1	<i>Características</i>	13
2.4.2	<i>Razones de la utilización de Java en el proyecto</i>	14
2.5	PRIMEFACES 3.5	14
2.5.1	<i>Características</i>	14
2.5.2	<i>Ventajas y Desventajas</i>	15
2.6	SERVIDOR DE APLICACIONES.....	15
2.6.1	<i>Glassfish 3.1.2</i>	15
2.7	NETBEANS	16
2.8	HERRAMIENTAS CASE	16
2.8.1	<i>StarUML</i>	16
2.8.2	<i>PowerDesigner 12.0</i>	16
2.9	JASPER REPORTS.....	17
2.10	ORACLE EXPRESS EDITION 11G FREE.....	17
2.11	SQL DEVELOPER	18
2.11.1	<i>Características</i>	19
2.12	METODOLOGÍA DE TRABAJO PARA EL DESARROLLO	19
2.12.1	<i>Scrum</i>	19
2.12.2	<i>Roles</i>	20
2.12.3	<i>Reuniones</i>	21
2.12.4	<i>Proceso</i>	21
2.12.5	<i>Documentos</i>	22
2.12.6	<i>Beneficios</i>	23
2.13	UML.....	23

CAPÍTULO 3: APLICACIÓN DE LA METODOLOGÍA	24
3.1	IMPORTANCIA DE UTILIZAR UNA METODOLOGÍA..... 24
3.2	DESCRIPCIÓN GENERAL DE LA METODOLOGÍA SCRUM..... 25
3.2.1	<i>Fundamentación</i> 25
3.2.2	<i>Desarrollo Iterativo e Incremental</i> 25
3.2.3	<i>Valores de trabajo.....</i> 27
3.2.4	<i>Personas y roles del proyecto</i> 27
3.3	CONSTRUCCIÓN DE HISTORIAS DE USUARIO
3.3.1	<i>Descripción formato Historia de usuario:.....</i> 29
3.4	PLANIFICACIÓN DEL PRODUCT BACKLOG
3.4.1	<i>Descripción formato Product Backlog</i> 30
3.5	PLANIFICACIÓN DE LOS SPRINTS
3.5.1	<i>Descripción del formato de sprints.....</i> 32
3.5.2	<i>Sprint 1: “Administración de usuarios, perfiles y permisos”.....</i> 33
3.5.3	<i>Sprint 2: “Administración de profesores”.....</i> 35
3.5.4	<i>Sprint 3: “Administración de Mallas curriculares”.....</i> 37
3.5.5	<i>Sprint 4: “Administración de alumnos”.....</i> 40
3.5.6	<i>Sprint 5: “Administración de aulas”.....</i> 42
3.5.7	<i>Sprint 6: “Administración de cronograma de clases”.....</i> 43
3.5.8	<i>Sprint 7: “Registro y modificación de asistencias de profesores y alumnos”.....</i> 45
3.5.9	<i>Sprint 8: “Administración de e-mails”</i> 47
CAPÍTULO 4: ANÁLISIS Y DISEÑO DEL SISTEMA	50
4.1	PRELIMINAR
4.2	PLANEACIÓN
4.2.1	<i>Toma de requerimientos.....</i> 52
4.3	DIAGRAMAS DE CASOS DE USO
4.3.1	<i>Diagrama de casos de uso del sistema AsistSoft</i> 54

4.4	DISEÑO DE INTERFACES	54
4.4.1	<i>Pantalla de Login</i>	56
4.4.2	<i>Pantallas: "Perfil administrador"</i>	56
4.4.3	<i>Pantallas: "Perfil alumno"</i>	59
4.4.4	<i>Pantallas "Perfil profesor"</i>	60
4.4.5	<i>Pantallas "Perfil coordinador"</i>	61
4.5	DIAGRAMA DE PAQUETES.....	61
4.5.1	<i>Paquete EJB</i>	64
4.5.2	<i>Paquete Web</i>	67
4.6	DIAGRAMA DE DISTRIBUCIÓN.....	71
4.7	DIAGRAMA DE CLASES	72
4.7.1	<i>Diagrama de Clases: Administración de aulas</i>	72
4.7.2	<i>Diagrama de Clases: Administración de cronogramas</i>	73
4.7.3	<i>Diagrama de Clases: Administración de mallas curriculares</i>	74
4.7.4	<i>Diagrama de Clases: Administración de profesores, coordinadores, alumnos y usuarios</i>	74
4.8	DIAGRAMA LÓGICO DE LA BASE DE DATOS.....	75
CAPÍTULO 5: PRUEBAS DEL SISTEMA.....		76
5.1	INTRODUCCIÓN.....	76
5.2	CASOS DE PRUEBA	76
5.2.1	<i>Prueba del caso de Uso: Ingresar al sistema</i>	76
5.2.2	<i>Prueba del caso de Uso: Actualizar Información Personal</i>	77
5.2.3	<i>Prueba del caso de Uso: Cambiar Contraseña</i>	78
5.2.4	<i>Prueba del caso de Uso: Consultar cronograma</i>	79
5.2.5	<i>Prueba del caso de Uso: Registrar Asistencia profesor y alumno</i>	80
5.3	PRUEBA DE CONCURRENCIA	81
5.4	EVALUACIÓN DE RESULTADOS	82

CAPITULO 6: CONCLUSIONES Y RECOMENDACIONES.....	83
6.1 CONCLUSIONES.	83
6.2 RECOMENDACIONES.	84
BIBLIOGRAFÍA.....	85
ANEXOS.....	88
ANEXO 1.- DOCUMENTO DE ESPECIFICACIÓN DE REQUERIMIENTOS.....	88
ANEXO 2.- DIAGRAMAS Y DESCRIPCIÓN DE CASOS DE USO DEL SISTEMA ASISTSOFT.....	103
ANEXO 3.- MANUAL DE USUARIO SISTEMA ASISTSOFT.....	157
ANEXO 4.- FORMATO DE ENCUESTAS A USUARIOS DEL SISTEMA ASISTSOFT.....	183

INDICE DE CUADROS

Cuadro 1: Descripción del Software a utilizar en el proyecto	7
Cuadro 2: Ventajas y Desventajas de utilizar una Arquitectura Web.....	9
Cuadro 3: Ventajas y Desventajas de utilizar una Arquitectura de 3 capas..	11
Cuadro 4: Estructura de una Aplicación JEE	12
Cuadro 5: Ventajas y Desventajas que posee Primefaces 3.5	15
Cuadro 6: Definición de roles del equipo de trabajo.....	28
Cuadro 7: Formato de Historia de usuario	28
Cuadro 8: Formato del Product Backlog	30
Cuadro 9: Product Backlog del proyecto AsistSoft.....	31
Cuadro 10: Formato para la especificación de sprints	32
Cuadro 11: Historia de usuario “Gestión de usuarios, perfiles y permisos” ..	34
Cuadro 12: Sprint N°. 1 “Administración de usuarios y asignación de perfiles.”	35
Cuadro 13: Historia de usuario “Administración de profesores”	36
Cuadro 14: Sprint N°. 2 “Administración de profesores”	37
Cuadro 15: Historia de usuario “Administración de Mallas curriculares.....	38
Cuadro 16: Sprint N° 3 “Administración de Mallas curriculares”	39
Cuadro 17: Historia de usuario “Administración de alumnos”	40
Cuadro 18: Sprint N°. 4 “Administración de alumnos”	41
Cuadro 19: Historia de usuario “Administración de aulas”	42
Cuadro 20: Sprint N°. 5 “Administración de aulas”	43
Cuadro 21: Historia de usuario “Administración de cronograma de clases” ..	43
Cuadro 22: Sprint N°.6 “Administración de cronograma de clases”	45

Cuadro 23: Historia de usuario “Administración de asistencias de profesores”	46
Cuadro 24: Sprint N°. 7 “Administración de asistencias”	47
Cuadro 25: Historia de usuario “Administración de e-mails”	48
Cuadro 26: Sprint N°. 8 “Administración de e-mails”	49
Cuadro 27: Principios de Diseño de las interfaces de usuario	55
Cuadro 28: Caso de prueba: Ingresar al sistema	77
Cuadro 29: Caso de Prueba: Registrar Información Personal profesor.....	78
Cuadro 30: Caso de Prueba: Cambiar Contraseña.....	79
Cuadro 31: Caso de Prueba: Consultar cronograma	79
Cuadro 32: Caso de Prueba: Registrar Asistencia profesor y alumno.....	81
Cuadro 33: Resultado Velocidad de Transacción	82

ÍNDICE DE FIGURAS

Figura 1: Arquitectura tres capas	10
Figura 2: Arquitectura Básica de EJB.	13
Figura 3: Pantalla inicial de Sql Developer.....	18
Figura 4: Metodología de trabajo de Scrum	20
Figura 5: Situación actual del proceso de control de asistencia	51
Figura 6: Situación propuesta toma de asistencia a los alumnos.	52
Figura 7: Diagrama de casos de uso del sistema AsistSoft.....	54
Figura 8: Pantalla de Login	56
Figura 9: Pantalla Gestión de profesores	56
Figura 10: Pantalla Gestión de alumnos	57
Figura 11: Pantalla Gestión de coordinadores	57
Figura 12: Pantalla Administración de módulos y asignaturas	58
Figura 13: Pantalla Gestión de Promociones	58
Figura 14: Pantalla Información Personal alumno.....	59
Figura 15: Pantalla Enviar Solicitudes.....	59
Figura 16: Pantalla Registrar Asistencia	60
Figura 17: Pantalla Información Personal profesor	60
Figura 18: Pantalla Asignar cronograma.....	61
Figura 19: Diagrama de Paquetes	64
Figura 20: Estructura Paquete EJB.....	65
Figura 21: Estructura Paquete Negocio	65
Figura 22: Estructura Paquete Interfaz	66
Figura 23: Estructura Paquete Sentencias.....	67
Figura 24: Estructura Paquete Web	67

Figura 25: Estructura Paquete Mensajes	68
Figura 26: Estructura Paquete Cálculo	68
Figura 27: Estructura Paquete Reportes	69
Figura 28: Estructura Paquete Vista	70
Figura 29: Diagrama de Distribución	71
Figura 30: Diagrama de Clases “Administración de aulas”	72
Figura 31: Diagrama de Clases “Administración de cronogramas”	73
Figura 32: Diagrama de Clases “Administración de mallas curriculares”	74
Figura 33: Diagrama de Clases “Administración de profesores, coordinadores, alumnos y usuarios”	74
Figura 34: Diagrama Lógico de la Base de Datos	75
Figura 35: Resultados de la encuesta aplicada a los usuarios	82

LISTADO DE ANEXOS

ANEXO 1

DOCUMENTO DE ESPECIFICACIÓN DE REQUERIMIENTOS.

ANEXO 2

DIAGRAMAS Y DESCRIPCIÓN DE CASOS DE USO DEL SISTEMA ASISTSOFT.

ANEXO 3

MANUAL DE USUARIO SISTEMA ASISTSOFT.

ANEXO 4

FORMATO DE ENCUESTAS A USUARIOS DEL SISTEMA ASISTSOFT.

GLOSARIO

Programa Postgrado: Comprende los estudios de segundo y tercer ciclo conducentes a la obtención de los títulos de Máster y Doctor.

Código SNNA: Código del programa, según el registro del Sistema Académico de la SENESCYT.

ID BANNER: Es un identificador único que el sistema escolástico BANNER le asigna a cada uno de los alumnos de la Universidad de la Fuerzas Armadas – ESPE.

Módulo: Un módulo es una propuesta organizada de los elementos o componentes instructivos para que el alumno de la Unidad de Gestión de Postgrados desarrolle unos aprendizajes específicos en torno a un determinado tema o tópico.

Asignatura: Materia que se enseña en un curso de postgrados y que forma parte de un módulo.

Cronograma de clases: Es una planificación en relación al tiempo que se debe realizar previo al registro de asistencia del profesor.

Ingeniería de Software: Disciplina formada por un conjunto de métodos, herramientas y técnicas que se utilizan en el desarrollo de los programas informáticos.

Servidor de Aplicaciones: Servidor en una red de computadores que ejecuta ciertas aplicaciones informáticas.

Open Office: Es una suite ofimática libre (código abierto y distribución gratuita) que incluye herramientas como procesador de textos, hoja de cálculo, presentaciones, herramientas para el dibujo vectorial y base de datos.

Router: Dispositivo que proporciona conectividad a nivel de red o nivel tres en el modelo OSI. Su función principal consiste en enviar o encaminar paquetes de datos de una red a otra.

Red LAN: (Local Area Network), Red de área local. Red que conecta los ordenadores en un área relativamente pequeña y predeterminada (como una habitación, un edificio, o un conjunto de edificios).

ACRÓNIMOS

SNNA: Secretaría Nacional de Nivelación y Admisión.

C.I: Cédula de Identidad

IEEE 830: Estándar para especificación de Requerimientos de Software de la IEEE, Institute of Electric and Electronic Engineers, en E.U.A.

RESUMEN

El presente proyecto tiene como objetivo el desarrollo e implementación de un sistema web que permite automatizar el Control de Asistencia de profesores y alumnos de la Unidad de Gestión de Postgrados de la Universidad de las Fuerzas Armadas – ESPE, el sistema web permite además generar los reportes más importantes que reflejan los procesos realizados dentro del sistema.

El análisis, diseño e implementación del sistema Web fue realizado siguiendo el proceso de la metodología ágil Scrum que permite realizar avances pequeños en funcionalidades de los módulos del sistema el cual está enfocado a resultados positivos que el usuario quiere visualizar en las acciones que realiza el software, el interrelacionar al usuario de forma directa con el Scrum Master y los desarrolladores garantizará la calidad del software.

El presente proyecto, el cual se denomina AsistSoft está desarrollado para la utilización en un ambiente web hecho en Java JEE6. En el desarrollo de la presente aplicación se utilizaron herramientas tecnológicas como un servidor de aplicaciones Glassfish 3.2, EJB 3.0, Primefaces 3.5 y un gestor de base de datos Oracle Express Edition 11g edición libre.

Con una adecuada utilización del presente sistema se logra agilizar los procesos de control de asistencias y se llega a obtener un control óptimo y dinámico de la información, además de ser un sistema cero papeles que ayuda a disminuir el consumo de recursos.

Palabras Clave: CONTROL DE ASISTENCIA, SCRUM, EJB, WEB, JAVA.

ABSTRACT

The present project has the purpose to develop and implement of a web system that allows automatizing the assistance control for the teacher and the student, made in the Postgraduate Management Unit at the University of the Armed Forces; the web system also allows generate the most important reports which reflect the processes carried out within the system.

The analysis, design and implementation of the web system was carried out by the process of the Scrum agile methodology that allows to carry on small advances in the system modules functionalities which is focused to obtain positive results that the user wants to visualize through actions the software executes, the interaction of user directly with the Scrum Master and the developers guarantee a high quality software.

The present project, called AsistSoft is developed for use in a web environment made in JEE6 Java. Technological tools; such are Glassfish 3.2 as application server, EJB 3.0, Primefaces 3.5 and an Oracle Express Edition 11g free edition as Database Manager in the development of the present application have been used.

The appropriate use of the present system is achieved to expedite the process of assistance control and so you obtain an optimal and dynamic control of the information, in addition to being a zero papers system that helps reduce resource consumption.

Key words: ASSISTANCE CONTROL, SCRUM, EJB, WEB, JAVA.

CAPÍTULO 1

DESARROLLO DE UNA APLICACIÓN WEB DE CONTROL DE ASISTENCIAS PARA PROGRAMAS DE POSTGRADO, MEDIANTE LA UTILIZACIÓN DE LA PLATAFORMA JAVA ENTERPRISE EDITION JEE6 WEB APLICANDO LA METODOLOGÍA SCRUM.

1.1 INTRODUCCIÓN

A través del tiempo, la tecnología ha evolucionado de una manera asombrosa, permitiendo automatizar procesos para un mejor control.

Los procesos rutinarios que son manejados en forma manual resuelven procesos de una manera lenta, es por ello que llevarla a un sistema automatizado garantiza un mejor trato de la información.

En gran parte la tecnología de la información, reúne, almacena y procesa, es por ello que se desea disminuir tiempo en la ejecución de las tareas realizadas mediante procesos tediosos para el ser humano.

Actualmente la Unidad de Postgrados no cuenta con la tecnología suficiente, sin embargo el alcance del tema de esta tesis tiene como objetivo la Gestión de Control de Asistencias ya que al momento el proceso se lo realiza de forma manual, por ello registran su asistencia en hojas de papel las cuales son archivadas y no brindan un control exacto de las horas laboradas, además de convertirse en una tarea tediosa y lenta a la hora de realizar consultas y obtener reportes de asistencias.

1.2 PLANTEAMIENTO DEL PROBLEMA

El proceso de control de asistencias en la Unidad de Postgrados se lleva a cabo de forma manual, recolectando los datos mediante un libro de asistencias en el cual cada persona debe registrar la hora de ingreso y salida. Entre las dificultades que se podría presentar al contar con un proceso manual se puede mencionar:

- Que las personas no tomen con responsabilidad la hora de ingreso y/o salida a la institución.
- Se puede alterar o manipular los datos que constan en el libro de ingreso.
- Que se pierdan los documentos y la manipulación de la información sea muy lento.

Con los antecedentes expuestos surge la necesidad de implementar un sistema que automatice dichos procesos, evite la pérdida de documentos y que agilice el proceso de control de tal manera que sea confiable, seguro y eficaz.

Es por ello que con el fin de solventar las necesidades del usuario, se desea desarrollar un sistema de software que permita gestionar y Controlar la Asistencia del personal involucrado en la Unidad de Gestión de Postgrados.

1.3 JUSTIFICACIÓN

El desarrollo del presente proyecto se lo debe realizar por las siguientes razones: es muy importante contar con un sistema web, ya que una aplicación orientada a la web es muy ventajosa, las bondades que se hacen

presentes son el ingreso desde cualquier ubicación, la independencia de plataformas por parte del cliente.

Con el afán de agilizar los procesos que conlleva el registro de asistencias de profesores y alumnos de la Unidad de Gestión de Postgrados de la Universidad de las Fuerzas Armadas - ESPE, se desarrollará e implantará un sistema que cumpla con los requisitos funcionales expuestos por el usuario a fin de que se dé una solución óptima que facilite el trabajo del personal involucrado en el manejo del sistema.

Los resultados producidos al utilizar esta herramienta web será prácticamente la reducción de tiempo y recursos.

1.4 ALCANCE

El alcance del proyecto viene dado por la creación de un sistema Informático para la Unidad de Postgrados que automatizará y atenderá los siguientes procesos:

1.4.1 Proceso Administrativo - Recursos Humanos

El sistema para el proceso de administración de recursos humanos permitirá el manejo de control de asistencias de profesores y alumnos de la Unidad de Postgrados. Luego gestionará y reportará la asistencia de todas las personas citadas anteriormente.

- Administración de profesores y alumnos.
- Administración de Asistencia.

1.4.2 Proceso Administrativo - Seguridad

Como apoyo para los procesos mencionados anteriormente el sistema incluirá el módulo de seguridad permitiendo el manejo de usuario, perfiles y permisos para garantizar el acceso, seguridad y veracidad de toda la información del sistema.

- Administración de usuarios
- Administración de permisos

1.4.3 Reportes Estadísticos

Los datos almacenados en la base de datos son mucho más importantes cuando el sistema es capaz de realizar reportes con estos datos, por tal motivo, para todos los procesos, el sistema permitirá generar los principales reportes necesitados por el usuario.

- Promedios del total de mujeres y hombres en una Promoción de Postgrados.
- Asistencias de profesores y alumnos.

El sistema constará de las etapas normales para el desarrollo como análisis, diseño, desarrollo, pruebas e instalación del sistema.

1.5 OBJETIVOS

1.5.1 General

Desarrollar un sistema Software de Gestión de Control de Asistencia para la Unidad de Postgrados de la Universidad de las Fuerzas Armadas - ESPE aplicando la Metodología Scrum que facilite el manejo eficiente de la información.

1.5.2 Específicos

- Realizar la Especificación de requerimientos del sistema.
- Diseñar el Plan de desarrollo con la metodología Scrum considerando la herramienta Ejb 3.0.
- Realizar el análisis y diseño en base de la metodología seleccionada.
- Realizar la construcción del sistema.
- Preparar y ejecutar las pruebas de la aplicación.
- Instalar el sistema y capacitar a los usuarios.

1.6 METODOLOGÍA

Para la construcción del presente proyecto se determinó utilizar una metodología ágil, la metodología Scrum es la metodología adecuada que mejor se acopla a las necesidades del proyecto, debido a que los requerimientos del proyecto son inestables, ya que el cliente solo comprende lo que necesita una vez que visualiza la primera versión del sistema, además al ser una metodología iterativa e incremental asegurará que siempre estará disponible una versión potencialmente útil y funcional del producto.

“SCRUM permite de forma rápida y repetidamente inspeccionar el software

que se está construyendo, permitiendo al negocio establecer las prioridades y ayudando a los equipos a focalizarse en desarrollar el más alto valor de negocio en el menor tiempo. Los equipos se gestionan a sí mismos para determinar la mejor manera de entregar las características de mayor prioridad. Cada dos semanas a un mes se puede ver software real y decidir si éste es el software que se desea en producción o continuar y mejorar el mismo en una próxima iteración” (Gioacchini, 2011).

Por los antecedentes expuestos, se ha determinado utilizar una metodología flexible y eficiente que se adapte a los cambios frecuentes del cliente.

1.7 RECURSOS

1.7.1 Hardware

- 2 Computadores personales

1.7.2 Software

Para desarrollar el software de control de asistencias para la Unidad de Gestión de Postgrados se requerirán las aplicaciones que se detallan en el Cuadro 1.

Cuadro 1: Descripción del Software a utilizar en el proyecto

Controlador	Aplicación a utilizar
IDE de Desarrollo	Netbeans 7.0.1
Servidor de Aplicaciones	Glasfish 3.2
Gestor de Base de Datos	Oracle 11 g Express Edition
Administración Base de Datos	Sql Developer Edición Libre
Modelado UML	Star UML v 5.02
Modelado Base de Datos	Power Designer 12.5
Browser	Mozilla Firefox v 14.0.1 o superior
Framework de Diseño	Primefaces 3.5

1.7.3 Recursos Humanos

- Susana Sntaxi
- Diego Moromenacho

1.8 FACTIBILIDAD

1.8.1 Factibilidad Técnica

Técnicamente el proyecto es viable, ya que se cuenta con las herramientas necesarias proporcionadas por la institución como son: información y recursos indispensables para su creación.

1.8.2 Factibilidad Operativa

La disponibilidad de la información involucrada en el proceso de levantamiento de requerimientos garantiza el perfecto establecimiento de las necesidades.

1.8.3 Factibilidad Tecnológica

Se trabajará sobre la plataforma Java Enterprise Edition JEE 6 Web.

CAPÍTULO 2: MARCO TEÓRICO

2.1 ARQUITECTURA WEB

Primero es necesario conocer que es una aplicación web, en lo que respecta a la Ingeniería de Software definir lo que significa una aplicación web se encarga de mencionar que:

Es aquel aplicativo que se encuentra en proceso de funcionamiento alojado en un servidor de aplicaciones, se podrá acceder desde cualquier parte del mundo mediante navegadores web los cuales harán una petición a este servidor el cual enviará una respuesta a lo solicitado, lo cual se conoce comúnmente como un proceso de peticiones cliente-servidor. Gracias a la explicación anterior se dirá lo siguiente:

“La arquitectura Web es aquel servicio que consiste en el uso del protocolo HTTP para que desde un cliente (navegador o browser) se solicite un documento dentro de la red, y un servidor http o servidor web le sirva una página en formato HTML.” (Areba, 2001).

2.1.1 Ventajas y Desventajas:

En el Cuadro 2 se especifican algunas ventajas y desventajas que se obtendrán al utilizar una Arquitectura Web en la construcción del sistema.

Cuadro 2: Ventajas y Desventajas de utilizar una Arquitectura Web

Ventajas	Desventajas
No es necesario la instalación del software en cada cliente: El aplicativo se tendrá que alojar en un servidor de aplicaciones y ponerlo en ejecución, así todos los clientes acceden al aplicativo.	Menos funcionalidad: La aplicación web tiene pocas funcionalidades que una de escritorio, por ejemplo en el escaso repertorio de controles disponibles para crear formularios (Mora, 2002).
Acceso rápido: El usuario solo debe tener acceso al internet y digitar el link correspondiente para redireccionar al aplicativo web.	Disponibilidad: Para que esté en funcionamiento depende de un proveedor de internet si este servicio deja de funcionar los usuarios no podrán acceder al aplicativo web.
“Fácil mantenimiento: El desarrollador puede realizar cambios en una sola computadora llamado servidor (en la que se encuentra el aplicativo ejecutándose) de manera remota.” (Cobo, 2005).	Mayor disponibilidad de recursos: se necesita invertir en ancho de banda y de implementación de una red muy robusta para los servidores.
Multiplataforma: Es posible usarlo desde cualquier sistema operativo cliente, basta con tener un navegador web.	Mayor coste en la implementación.
Se evitan inconsistencia de versionamiento: Solo existirá una actualización y esta se alojará en el servidor, los usuarios accederán a esa misma actualización en ejecución y no existirá usuarios con distintas versiones (Mora, 2002).	Las aplicaciones web requieren navegadores web totalmente compatibles para funcionar.

2.2 ARQUITECTURA FÍSICA DE TRES CAPAS

“En esta arquitectura la presentación, el procesamiento de la aplicación y la gestión de los datos son procesos lógicamente separados que se ejecutan sobre procesadores diferentes.” (GROUSSARD, 2010).

2.2.1 Capas de la Arquitectura

2.2.1.1 Capa visible para el usuario

Es prácticamente un navegador web que el usuario utiliza para acceder a una aplicación determinada desde su computador personal.

2.2.1.2 Capa intermedia

El navegador web se comunica con la lógica de negocio, la cual está en un servidor web, este servidor puede contener un servidor de aplicaciones en donde están las páginas alojadas.

2.2.1.3 Capa de base de datos

Generalmente éste tipo de aplicaciones maneja datos y para eso constará de un servidor en la cual estarán alojados y que enviará respuestas a una petición de la capa intermedia.

Conocida la definición de las capas que intervienen dentro de la Arquitectura, en la Figura 1 se visualiza un diagrama que muestra la relación y comunicación entre las capas.

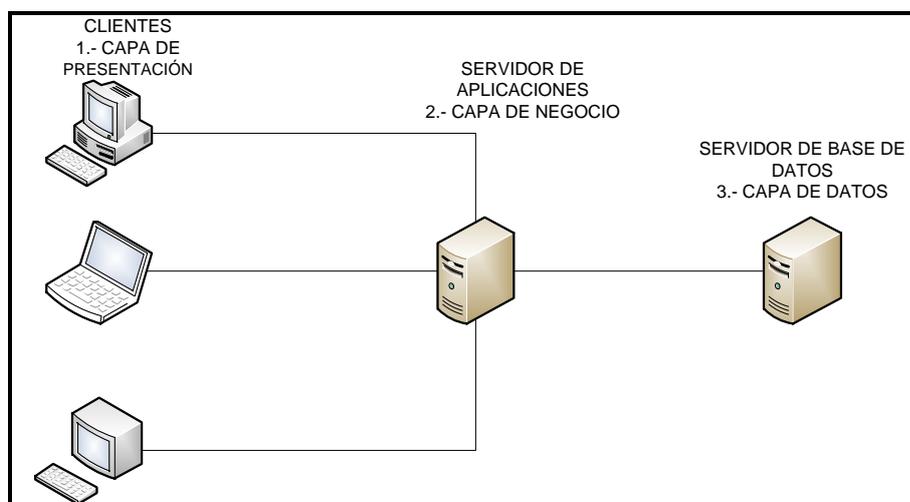


Figura 1: Arquitectura tres capas

2.2.2 Ventajas y Desventajas

Como ventajas y desventajas que ofrece una Arquitectura de 3 capas se puede mencionar algunas dentro del Cuadro 3.

Cuadro 3: Ventajas y Desventajas de utilizar una Arquitectura de 3 capas.

Ventajas	Desventajas
Escalabilidad de Hardware y Software.	Incrementan el tráfico de red.
Flexibilidad.	Mayor costo.
División de la carga de trabajo.	Dependencia de una de las tres capas ya que si una capa no funciona no estará en funcionamiento el aplicativo.
Mayor seguridad.	

El presente proyecto será desarrollado en el ambiente de tres capas orientado a la web, esto facilitará el análisis y desarrollo del software, al realizar este proyecto se analizará detalladamente que módulos se manejarán en la capa de negocio, métodos de conexión a la base de datos, validaciones, interacción con la capa de presentación y a su vez de la misma forma separarlos y facilitar el trabajo de programación y documentación. En lo que respecta al ambiente web es favorable ya que no será necesario realizar instalaciones en equipos cliente, estos solo accederán mediante un navegador web al servidor donde se encontrará el proyecto.

En lo que se refiere a la interfaz gráfica; es decir la capa de presentación para el usuario, se la manejará con la tecnología de Primefaces para facilitar la interacción con el usuario y por su presentación dinámica.

La base de datos que se utilizará será Oracle 11g Express Edition free, esta formará parte de la capa de datos.

2.3 JAVA JEE

“JEE es una plataforma de desarrollo Java Enterprise Edition y una herramienta muy popular para aplicaciones empresariales y web.” (GROUSSARD, 2010).

2.3.1 Ventajas

- Extensibilidad.
- Reestructuración.
- Flexibilidad.
- Mayor facilidad de uso.

2.3.2 Estructura de una Aplicación JEE

Existe una normativa de cómo esta tecnología está estructurada en un proyecto web, en el Cuadro 4 se puede observar esta estructura.

Cuadro 4: Estructura de una Aplicación JEE

Estructura	
Recursos Estáticos:	Paginas HTML, imágenes, sonidos, hojas de estilo.
Recursos dinámicos:	Servlets, JSP, Java Bean.
Librerías:	Se utilizan en los componentes dinámicos.
Descriptor de despliegue:	“Permite definir los parámetros de funcionamiento de la aplicación en el servidor” (Groussard, 2010).

El uso del lenguaje de programación Java realmente tiene muchas ventajas, el conocimiento extenso, facilidad de entendimiento del código, multiplataforma, la amplia documentación que existe en libros y en la web, por tal motivo se ha tomado la decisión de optar por este lenguaje de programación para realizar el proyecto.

Utilizar JEE6 beneficiará mucho al tener librerías actualizadas, ser compatible totalmente con Glasfish 3.0 o superior, ayudará a tener mejores resultados en el proyecto.

2.4 ENTERPRISE JAVABEANS

Los Enterprise JavaBeans (también conocidos por sus siglas EJB) son una de las API que forman parte del estándar de construcción de aplicaciones empresariales JEE. Su especificación detalla cómo los servidores de aplicaciones proveen objetos desde el lado del servidor, que son, precisamente los EJB (Cobo, 2005).

2.4.1 Características

- Comunicación remota.
- Transacciones.
- Control de la concurrencia.
- Seguridad.
- Ubicación de componentes en un servidor de aplicaciones.

En la Figura 2 se muestra la interacción de los elementos que conforman la Arquitectura EJB.

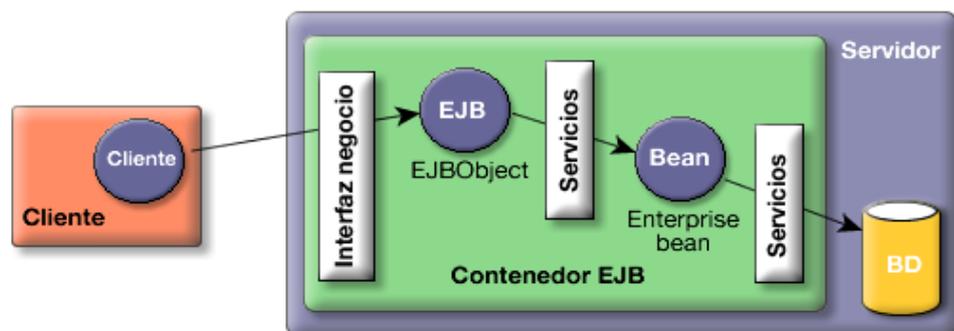


Figura 2: Arquitectura Básica de EJB.

Fuente: (Sierra, 2011)

2.4.2 Razones de la utilización de Java en el proyecto

La facilidad de crear programas que se vayan a manejar a través de un interfaz web, tener un programa distribuido en general, o un software que tengan que funcionar en distintos sistemas sin ningún cambio (programas "portables"), entre otros casos.

La facilidad de su sintaxis por la Familiarización de lenguaje C, ésta y muchas más razones que describen dentro de sus ventajas ha motivado desarrollar el proyecto en Java.

Para acceder al proyecto se deberá usar el navegador Mozilla, ya que brinda muchas ventajas en lo que se refiere a compatibilidad de librerías, y al ser un software de código libre y multiplataforma ayudará a no tener conflictos con algún plugin que use el proyecto.

2.5 PRIMEFACES 3.5

Es una librería de componentes de código abierto para JSF y que se encuentra en la capa del cliente, se lo utiliza para un toque de estilo vistoso a un proyecto web, posee varias librerías para facilitar el uso de la programación en lo que se refiere a estilos de una página web.

2.5.1 Características

- Un amplio conjunto de componentes de Interfaz de usuario.
- No se requiere extra configuración.
- Ajax incorporado.
- Conjunto de más de 25 temas integrados.
- Excelente documentación con ejemplos de código.

2.5.2 Ventajas y Desventajas

Varias son las ventajas que ofrece Primefaces 3.5, en lo que se refiere al diseño de una interfaz gráfica amigable para el usuario y se detalla en el Cuadro 5.

Cuadro 5: Ventajas y Desventajas que posee Primefaces 3.5

Ventajas	Desventajas
Una presentación muy animada de estilos ya que incorpora Ajax.	Demasiados componentes en un formulario podría aportar a la lentitud de la página web.
Posee varios componentes de validación de formularios.	Algunos componentes no son 100% compatibles con el conjunto de herramientas como jdk, se necesita siempre la actualización.
Fácil manejo de agregación de librerías.	Algunos navegadores web no logran cargar todos sus componentes, el que más se ajusta es Mozilla; es decir existe mucha compatibilidad con este navegador.

2.6 SERVIDOR DE APLICACIONES

El servidor de aplicaciones contiene el código de un proyecto y lo ejecuta, para que cada vez que un usuario envíe una solicitud éste emita una respuesta, para ello deberá estar en plena ejecución como un servicio normal dentro de un servidor físico.

Existen varios tipos de servidores de aplicaciones, cabe mencionar que están diseñados para cubrir una necesidad específica; pero los más comunes son sin duda Glassfish y Jboss.

2.6.1 Glassfish 3.1.2

Es un servidor de aplicaciones, tiene la característica de ser un software libre que permite ejecutar aplicaciones Java. La elección de un servidor de

aplicaciones depende de muchas variables: precio, desempeño, características, facilidad de uso, calidad de la documentación, apoyo, etc.

2.7 NETBEANS

“Es un entorno de desarrollo o una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación.” (Netbeans, 2013).

Esta herramienta es muy familiar y fácil de usar. En cuanto a la versión de Netbeans que se utilizará será a partir de la versión 7.0.

2.8 HERRAMIENTAS CASE

Las herramientas case son importantes para la investigación preliminar como son: análisis, diseño, implementación e instalación de un software, ya que ayudan mucho en el desarrollo de aplicaciones facilitando el trabajo del desarrollador.

2.8.1 StarUML

“Es una herramienta libre para el modelamiento de software basado en los estándares UML (Unified Modeling Language). Es una de las mejores alternativas gratis que hay en Internet para el modelamiento de software.” (Pérez, 2012).

2.8.2 PowerDesigner 12.0

“Es una herramienta para el análisis, diseño y construcción sólida de una base de datos y un desarrollo orientado a modelos de datos a nivel físico y conceptual.” (Salazar, 2012).

2.9 JASPER REPORTS

“Es un motor de informes de código abierto. Está escrito completamente en Java y es capaz de utilizar los datos procedentes de cualquier tipo de fuente de datos y producir documentos de píxeles perfectos que se pueden ver, imprimir o exportar en una variedad de formatos de documentos incluyendo HTML, PDF, Excel, OpenOffice y Word” (JASPERSOFT, 2013).

La utilización de Jasper Reports brindará muchas opciones de emitir un reporte en diferentes formatos y posteriormente permitir la impresión.

El manejo con Netbeans es muy simple, se debe tener el plugin de Ireports e instalar en las librerías de Netbeans.

2.10 ORACLE EXPRESS EDITION 11G FREE

Oracle es un potente gestor de base de datos, además es una herramienta cliente/servidor para la gestión de datos de alto nivel (Gabillaud, 2010).

Se considera a Oracle como uno de los sistemas de bases de datos más completos, destacando:

- Soporte de transacciones.
- Estabilidad.
- Escalabilidad.

Oracle Database 11g Express Edition free, ayudará en la construcción del proyecto y brindará agilidad, ya que no tiene muchos servicios que monitorear. Oracle tiene muchas librerías para la conexión con servidores de aplicación y fácilmente se puede trabajar en un ambiente Linux con la

conexión al proyecto.

Oracle sin duda es un gestor de base de datos que posee muchas características de gran utilidad que ayudan a administrar grandes cantidades de información.

2.11 SQL DEVELOPER

Es un entorno de desarrollo integrado gratuito que simplifica el desarrollo y gestión de base de datos. SQL Developer ofrece un desarrollo completo de las aplicaciones PL / SQL, una hoja de cálculo para ejecutar consultas y scripts, una consola de DBA para la gestión de la base de datos, una interfaz de informes, una solución completa de modelado de datos y una plataforma de migración (SQL DEVELOPER, 2013).

Esta herramienta ayudará a administrar de mejor manera la base de datos, posee muchas funcionalidades que facilita el manejo de un esquema y su interfaz se puede visualizar en la Figura 3.

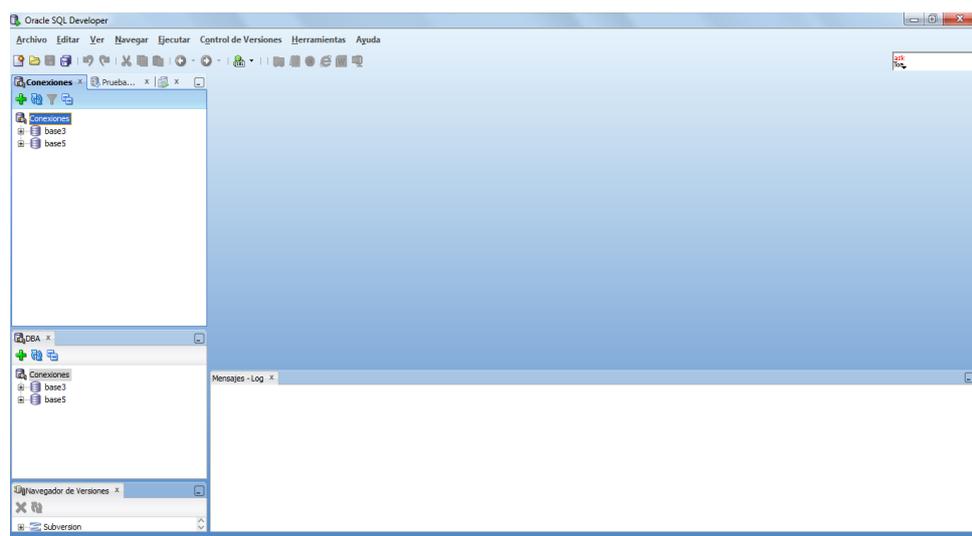


Figura 3: Pantalla inicial de Sql Developer

2.11.1 Características

- Conexiones múltiples.
- Modelamiento de base de datos.
- Consola Sql para scripts.
- Conexión remota.
- Reportes.
- Migración.
- Exportación e Importación de base de datos (SQL DEVELOPER, 2013).

2.12 METODOLOGÍA DE TRABAJO PARA EL DESARROLLO

2.12.1 Scrum

“Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para una empresa. Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación.” (SOFTENG, 2012), las fases de la Metodología Scrum se detallan en la Figura 4.

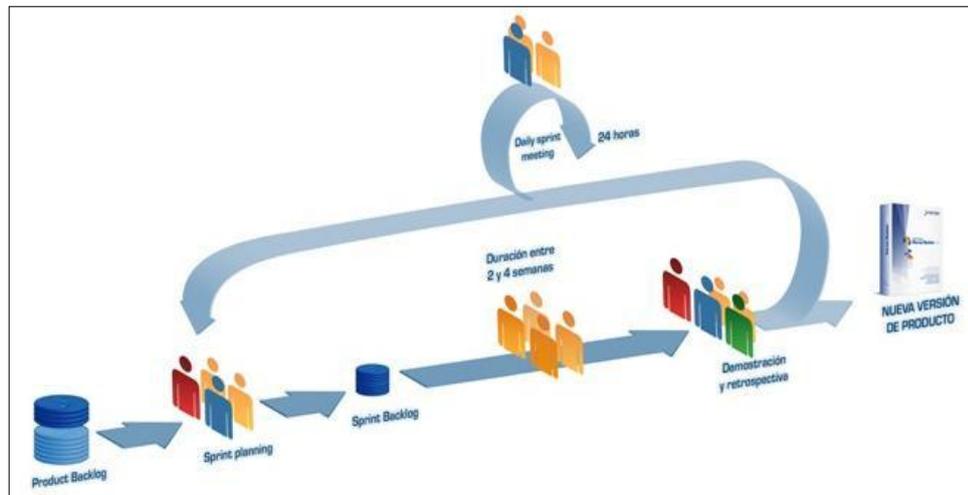


Figura 4: Metodología de trabajo de Scrum

Fuente: (SOFTENG, 2012)

2.12.2 Roles

2.12.2.1 The Scrum Team

Denominado como equipo de trabajo, son los desarrolladores del proyecto de software, ellos deciden como será realizado el trabajo y como distribuir las asignaciones o requerimientos y que tiempo tardarán en ello.

El equipo de desarrollo es auto organizado y tiene la responsabilidad de entregar el producto. Es un pequeño equipo de 3 a 9 personas con las habilidades transversales necesarias para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación, etc.) (Mann, 2008).

2.12.2.2 The Product Owner

Representa la voz del cliente, trabaja con el Scrum Team desde una perspectiva del negocio, administra un Product Backlog que es una lista de tareas con especificaciones de un producto y prioriza las funcionalidades posibles de construcción del sistema (IBM, 2010).

2.12.2.3 The Scrum Máster

Mantiene procesos y trabaja como el director del proyecto en las reuniones diarias que mantiene, su función es eliminar los impedimentos posibles para el cumplimiento de objetivos de cada sprint (Mann, 2008).

2.12.3 Reuniones

2.12.3.1 Sprint Planning Meeting

Se lleva a cabo al inicio del sprint y su duración es de ocho horas máxima (IBM, 2010).

2.12.4 Proceso

2.12.4.1 Creando Backlog

El Product Owner recoge todas las peticiones y especificaciones que son la base de los cambios del producto, pueden ser correcciones de errores y nuevas funcionalidades.

Al mismo tiempo se hace una lista de prioridades, el Product owner toma las decisiones personalmente acerca de prioridades en los cambios y entregas (IBM, 2010).

2.12.4.2 La fase sprint

Es un tiempo que lleva hacer un trabajo es recomendable que se estime 2 o tres semanas para terminarlo, después se hará una reunión para realizar las pruebas y entregar el producto final al cliente, durante los sprints el producto es diseñado, codificado y testeado (Schwaber, 2010).

2.12.4.3 El Scrum Diario

Es una reunión que se realiza cada tres días aconsejable para ver cómo van los avances y el tiempo estimado para la revisión de estas tareas, no van más allá de 30 minutos, son reuniones rápidas, esta metodología se enfoca más en el avance del desarrollo para ser entregado al cliente, todos los días el Scrum Team y el Scrum Máster deben tener una breve reunión tratando tres preguntas esenciales

¿Qué has hecho desde la última reunión?

¿Qué harás hoy hasta la próxima reunión?

¿Hay algo que impida lo que haces según lo tenías planeado?

Las dos primeras preguntas dan una visión a los participantes de cómo están avanzando en el proyecto y la tercera pregunta sirve de base para solucionar el problema (IBM, 2010).

2.12.5 Documentos

2.12.5.1 Product backlog

Es la lista de todo el trabajo que se desea realizar en el proyecto. Esta lista es priorizada por el Product Owner y puede ser actualizada y revisada cuando sea necesario (Gioacchini, 2011).

2.12.5.2 Sprint backlog

Es un subconjunto de requerimientos del product backlog que se implementarán durante un sprint. El equipo puede agregar nuevas tareas si el caso lo amerita para cumplir con el objetivo del sprint (IBM, 2010).

2.12.6 Beneficios

- **Cumplimiento de expectativas:** El líder del grupo de trabajo se encarga de dividir el trabajo en partes pequeñas y así poder cumplir con la finalización del desarrollo de cada módulo (SOFTENG, 2012).
- **Flexibilidad a cambios:** Cabe destacar que cuando se desarrolla de manera incremental pequeñas partes es más fácil entrar a cambios a etapas tempranas del proyecto (SOFTENG, 2012).
- **Producción de módulos más importantes:** Se puede realizar las partes más importantes y ponerlas en producción, ya que no es necesario terminar todo el proyecto para recién probarlo.
- **Mayor calidad del software:** Al ser una metodología expuesta a cambios tempranos se puede corregir cosas que no están claras y hacerlas de la mejor manera (SOFTENG, 2012).
- **Mayor productividad:** El equipo de trabajo tiene toda la visión del proyecto y no está a cargo de una burocracia que se encarga de todo (SOFTENG, 2012).
- **Predicciones de tiempos:** Es más fácil estimar un tiempo de desarrollo de un determinado módulo ya que se lo puede dividir y así facilitar las cosas para los programadores (IBM, 2010).

2.13 UML

Lenguaje Unificado de Modelado, es un lenguaje de modelado que ayuda a diseñar base de datos, diagramas de objetos y clases; es decir todo lo que intervendrá en el desarrollo de un proyecto (Weitzenfeld, 2007).

CAPÍTULO 3: APLICACIÓN DE LA METODOLOGÍA

3.1 IMPORTANCIA DE UTILIZAR UNA METODOLOGÍA

Antes de construir un producto software es indispensable elegir el mejor camino que conduzca a alcanzar un objetivo principal: el de haber llegado a satisfacer los requerimientos del cliente y que estarán presentes en las funciones que realice el sistema. Una metodología ayuda a tener una guía muy importante para conseguir un objetivo.

Las metodologías abarcan un amplio espectro de tareas que incluyen la comunicación, el análisis de requisitos, el modelo del diseño, la construcción de un software, la realización de pruebas y el soporte, por consiguiente se define que metodología de desarrollo de software en ingeniería de software es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información.

Es un marco de trabajo, donde intervienen subprocesos y estos son detallados en documentos que ayuda a llevar de forma ordenada un registro de cambios, acuerdos entre el cliente y la persona líder del proyecto. También se podrá mencionar que cualquier metodología que se aplique a un proyecto de desarrollo deberá ser ayudada de herramientas, métodos y modelos, así ayudará de mejor forma a que el producto sea de calidad para el cliente.

Otra de las funciones principales de una metodología es cumplir con los requisitos que demanda el cliente, en este punto cabe destacar la importancia que tiene el especificar con mucha claridad la toma de requerimientos, y así emprender primero un análisis exhaustivo y aclarar que

es lo que se podrá cumplir y que no, o en un caso de que realmente es posible desarrollar y que funcionalidades no son posibles construir en aspectos técnicos de programación.

3.2 DESCRIPCIÓN GENERAL DE LA METODOLOGÍA SCRUM

3.2.1 Fundamentación

Para el desarrollo del presente Proyecto se ha utilizado la Metodología Scrum, una metodología ágil que se enfoca en los resultados del aplicativo más allá de la documentación exhaustiva que aplican otras metodologías.

Las Metodologías Ágiles constituyen un nuevo enfoque en el desarrollo de software, mejor aceptado por los desarrolladores de proyectos que las metodologías convencionales debido a la simplicidad de sus reglas y prácticas, su orientación a equipos de desarrollo de pequeño tamaño, su flexibilidad ante los cambios y su ideología de colaboración (Pérez, 2012).

El estar orientado a presentar avances pequeños de forma rápida y bien definidos, este tipo de metodología se convierte en un potencial para ver resultados los cuales se pondrán en marcha lo más antes posible.

3.2.2 Desarrollo Iterativo e Incremental

Este tipo de desarrollo iterativo e incremental en un proyecto de software hace que cada requerimiento se divida en diversos bloques temporales, se desarrolle y se entregue el avance al cliente si ocurre un cambio se volverá a revisar y se hará cualquier modificación si así lo requiere el cliente.

Scrum divide un requerimiento en bloques con una asignación de tiempo que no va más allá de 4 semanas, estos bloques comúnmente llamados

sprint los asigna a cada desarrollador. Las personas inmersas en el proyecto se reúnen cada dos días como recomendable para ver cómo va el avance de la funcionalidad que lleva 2 o 4 semanas.

La entrega de pequeñas partes a los clientes hace que se cumpla los requerimientos y estos deberán ser revisados por las dos partes tanto el líder del proyecto como el cliente al finalizar la entrega, cabe destacar que cada requisito se debe cumplir en toda su totalidad.

El objetivo de utilizar esta metodología es justamente recoger los requerimientos, dividirlos en bloques y asignarles un tiempo estimado las cuales serán desarrolladas por el grupo de trabajo, para ver el avance de cada tarea se procede a una reunión no más de dos días, se realizan pruebas y se las entregará al cliente, el cual tendrá la potestad de aprobar o no un bloque de desarrollo con sus resultados. La facilidad que tiene Scrum para adaptarse a las personas de desarrollo es una de los principales motivos que ayudará a realizar un buen trabajo en este proyecto (IBM, 2010).

Las principales razones del uso de un ciclo de desarrollo iterativo e incremental de tipo Scrum para la ejecución de este proyecto son:

- **Sistema modular:** Permiten desarrollar una base funcional mínima y sobre ella ir incrementando las funcionalidades o modificando el sistema (Scrum Manager, 2014).
- **Entregas frecuentes y continuas:** La entrega de módulos terminados al cliente en un tiempo mínimo (IBM, 2010).
- **Previsible inestabilidad de requisitos.**

- Es posible que el sistema incorpore más funcionalidades de las inicialmente identificadas.
- Es posible que durante la ejecución del proyecto se altere el orden en el que se desean recibir los módulos o historias de usuario terminadas.
- Para el cliente resulta difícil precisar cuál será la dimensión completa del sistema, y su crecimiento puede continuarse en el tiempo suspenderse o detenerse (IBM, 2010).

3.2.3 Valores de trabajo

Los valores que deben ser practicados por todos los miembros involucrados en el desarrollo del presente proyecto y que harán posible que la metodología Scrum tenga éxito son:

- Autonomía del equipo
- Respeto en el equipo
- Responsabilidad y auto-disciplina
- Foco en la tarea
- Reconocer y estimular el trabajo de los demás (IBM, 2010).

3.2.4 Personas y roles del proyecto

En base a los roles utilizados en la metodología Scrum, se definieron las personas que participarán en el desarrollo del proyecto, las cuales se detallan en el Cuadro 6.

Cuadro 6: Definición de roles del equipo de trabajo

Persona	Rol
- Susana Sntaxi - Diego Moromenacho	Scrum Team
Ing. Mario Ron (Coordinador del Programa de Maestría de Evaluación y Auditoría de Sistemas)	Product Owner

3.3 CONSTRUCCIÓN DE HISTORIAS DE USUARIO

Para empezar con el proceso de desarrollo del sistema AsistSoft es importante primero construir el Product Backlog, que será elaborado en base a las historias de usuario que fueron determinadas conjuntamente con el cliente en la reunión denominada “Sprint Planning Meeting”, a dicha reunión asistieron tanto el cliente como el equipo de desarrollo. Las historias de usuario son concretas, ya que son el resultado de escuchar atentamente al cliente, las historias de usuario contienen información con un vocabulario propio del negocio del cliente; es decir en las historias de usuario no se utilizan palabras técnicas, su estructura es simple y el formato que se va a utilizar se presenta en el Cuadro 7.

Cuadro 7: Formato de Historia de usuario

HISTORIA DE USUARIO	
Número:	Usuario:
Nombre Historia:	
Prioridad en Negocio:	Riesgo en desarrollo:
Iteración:	
Programador Responsable:	
Descripción:	
Como probarlo:	
Observaciones del equipo de desarrollo:	

3.3.1 Descripción formato Historia de usuario:

Número: Es el número de Historia de usuario.

Usuario: Define el usuario de la historia.

Nombre Historia: Es una breve descripción de la historia que se logró extraer luego del dialogo con el cliente.

Prioridad en Negocio: Determina cuán importante es la historia de usuario para el negocio y se evalúa de la siguiente manera:

- Prioridad alta
- Prioridad media
- Prioridad baja

Riesgo en desarrollo: Define el nivel de riesgo que existirá en el proceso de desarrollo y se lo evalúa de la siguiente manera:

- Alta
- Medio
- Bajo

Iteración: Determina el número de iteración que se le asigna a la historia.

Programador responsable: Señala quien será el responsable de desarrollar e implementar la historia de usuario.

Descripción: Indica las actividades y datos que se deben ingresar para cumplir con la historia de usuario.

Como probarlo: Determina los pasos a seguir para comprobar si las actividades de la historia fueron desarrolladas correctamente por el programador responsable.

Observaciones del equipo de desarrollo: Determina si existe alguna limitación o condición en la historia de usuario.

3.4 PLANIFICACIÓN DEL PRODUCT BACKLOG

Scrum facilita los artefactos necesarios para el cumplimiento de las tareas programadas en el tiempo estimado y con los recursos establecidos, para ello se deberá crear el primer artefacto que es el Product Backlog, este artefacto contiene un detalle de las tareas que se tendrán que realizar, dichas tareas se las categorizará por prioridades y el formato que se va a utilizar se detalla en el Cuadro 8.

Cuadro 8: Formato del Product Backlog

Prioridad	Ítem #	Descripción	Responsable(s)
Alta			
Media			
Baja			

3.4.1 Descripción formato Product Backlog

Prioridad: Señala el nivel de prioridad

Ítem #: Un número secuencial que identifica al módulo y empieza en 1.

Descripción: Se detalla los módulos de las historias de usuario definidas previamente.

Responsable: Señala que persona del equipo de desarrollo es el responsable del módulo.

Una vez detalladas las historias de usuario, el equipo de desarrollo

procede a redactar el Product Backlog que especifican los módulos del sistema AsistSoft, el nivel de prioridad que se les ha asignado y los responsables de cada módulo, el Product Backlog del sistema se encuentra detallado en el Cuadro 9.

Cuadro 9: Product Backlog del proyecto AsistSoft

Prioridad	Ítem #	Descripción	Responsable(s)
Alta			
	1	Administración de usuarios	Susana-Diego
	2	Administración de profesores	Susana-Diego
	3	Administración de Mallas curriculares	Susana-Diego
Media			
	4	Administración de alumnos	Susana-Diego
	5	Administración de aulas.	Susana-Diego
	6	Administración de cronogramas	Susana-Diego
Baja			
	7	Registro y modificación de asistencia de profesores y alumnos.	Susana-Diego
	8	Administración de e-mails.	Susana-Diego

3.5 PLANIFICACIÓN DE LOS SPRINTS

Para la planificación de cada uno de los sprints se elaboró un formato, donde se detallarán las tareas que deben realizar cada uno de los integrantes del equipo, la fecha de inicio y la fecha final de cada sprint, y el estado en que se va encontrando mientras avanza la iteración.

En el Cuadro10 se muestra el formato en donde se redactará cada uno de los sprints con sus respectivas tareas.

Cuadro 10: Formato para la especificación de sprints

Sprint N°:				
Referencia Historial:				
Fecha de Inicio:			Tareas Pendientes:	
Fecha culminación:			Días pendientes:	
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado

3.5.1 Descripción del formato de sprints

Sprint N°: Detalla el número de iteración.

Referencia Historial: Señala a que historial corresponde el sprint.

Fecha de inicio: Especifica la fecha inicial de la iteración.

Fecha de culminación: Especifica la fecha en la cual termina la iteración.

Orden de priorización: Determina el orden en el cual se desarrollarán cada una de las tareas.

Descripción de la tarea: Es una breve y concreta descripción de las actividades a desarrollarse.

Responsable: Es la persona encargada de desarrollar la tarea.

Duración días: Muestra el número de días que durará la iteración.

Estado: Señala en qué estado se va encontrando la iteración durante el trascurso del tiempo.

Tareas pendientes: Especifica el número de tareas que faltan por completar.

Días pendientes: Número de días restantes para completar la iteración.

Una vez definido el formato que se va a utilizar para redactar cada uno de los sprints, se lleva a cabo una reunión con el equipo de desarrollo y el cliente para definir el primer sprint que contiene las actividades con un valor alto de prioridad para el cliente y se ha definido que cada iteración tiene que proporcionar un resultado completo es decir que añada funcionalidad de calidad al producto final y que sea susceptible de ser entregado al cliente en el tiempo acordado; es decir se utilizará 4 días como máximo para la entrega de cada tarea, en este tiempo se incluye cualquier modificación que se tenga que realizar según el punto de vista del cliente. Una vez empezado el sprint se llevará a cabo reuniones diarias para conocer las inquietudes e inconvenientes que se han presentado durante la ejecución de una tarea.

Conforme avanza el desarrollo del sprint es necesario que el equipo de desarrollo vaya actualizando el estado del sprint, y determinando cuantas tareas tienen pendientes y los días que faltan para que se complete el sprint de acuerdo a las fechas definidas previamente.

Una vez definidas los aspectos relevantes para el correcto desarrollo del sistema AsistSoft siguiendo el marco de trabajo de la metodología Scrum, se procede a definir el primer sprint.

3.5.2 Sprint 1: “Administración de usuarios, perfiles y permisos”.

Antes de detallar el sprint 1 se procede a revisar las Historias de usuario que fueron hechas en base a los requerimientos expuestos por el cliente, para esto se ha seleccionado la historia de usuario con el requerimiento de mayor prioridad y se lo detalla en el Cuadro 11.

Cuadro 11: Historia de usuario “Gestión de usuarios, perfiles y permisos”

HISTORIA DE USUARIO	
Número: 1	Usuario: Súper usuario
Nombre Historia: Gestión de usuarios, perfiles y permisos.	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Baja
Iteración: 1	
Programador Responsable: Susana	
Descripción: El sistema permita guardar, eliminar y modificar la información de los usuarios que podrán acceder al sistema y además deberá permitir asignarles sus respectivos perfiles y conceder permisos para que puedan acceder a un menú de opciones.	
Como probarlo: Ingresar al sistema con el rol de Súper usuario. Guardar la información de los usuarios, asignar un perfil a cada usuario y luego conceder permisos para las diferentes páginas del sistema de acuerdo a su perfil. Luego permitir que los diferentes usuarios ingresen al sistema mediante una autenticación y ver que cada uno de ellos puede acceder a las diferentes opciones de acuerdo al permiso que se les ha concedido.	
Observaciones del equipo de desarrollo: No se puede registrar un usuario con el mismo número de cédula de identidad (CI).	

3.5.2.1 Planificación Sprint 1

En sprint número uno redacta las tareas que corresponden al módulo de gestión de usuarios, asignación de perfiles y permisos. Con el cumplimiento del primer sprint se podrá controlar el acceso al sistema por los diferentes usuarios de acuerdo a los permisos concedidos. Este sprint contiene 4 tareas, las cuales se han detallado en orden para su ejecución, cada una de las tareas de este sprint tomará 4 días, cabe mencionar que las reuniones de los desarrolladores se realizarán cada dos días y al finalizar se harán pruebas de cada sección y se las entregará al cliente, si existen funciones que modificar se realizará en un plazo no mayor a dos días, en el Cuadro 12 se detalla el primer Sprint.

Cuadro 12: Sprint N°. 1 “Administración de usuarios y asignación de perfiles.”

Sprint N°:	1			
Referencia Historial:	Administración de usuarios y asignación de perfiles.			
Fecha de Inicio:	28/06/2013		Tareas Pendientes: 0	
Fecha de culminación:	05/07/2013		Días pendientes: 0	
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado
1	Inserción, modificación, eliminación de usuarios y asignación de perfiles.	Susana	4	Terminado
2	Inserción, modificación y eliminación de pantallas del sistema	Susana	4	Terminado
3	Inserción, modificación y eliminación de asignación de pantallas por perfiles de usuario	Diego	4	Terminado
4	Autenticación de usuarios en el sistema.	Diego	4	Terminado

3.5.3 Sprint 2: “Administración de profesores”.

Para la construcción del segundo sprint se procedió a revisar detenidamente la Historia de usuario que se detalla en el Cuadro 13, teniendo en consideración el orden de prioridad que definió el cliente.

Cuadro 13: Historia de usuario “Administración de profesores”

HISTORIA DE USUARIO	
Número: 2	Usuario: administrador.
Nombre Historia: Administración de profesores	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Baja
Iteración: 1	
Programador Responsable: Susana	
Descripción: El sistema permita insertar, modificar, visualizar y eliminar la información de los profesores de la Unidad de Postgrados. Registrar la cédula de identidad (CI), apellidos ,nombres, correo electrónico, el tipo de profesor (profesor interno o externo), contrato(tiempo completo o tiempo parcial), teléfono convencional, celular, especialidad, dirección de la casa, la dirección del trabajo y la nacionalidad,	
Como probarlo: El administrador ingresa al sistema. Guardar la información del profesor. Una vez guardada la información, se puede proceder a modificar los datos o eliminarlos.	
Observaciones del equipo de desarrollo: No se puede guardar los datos de un profesor con la misma cédula de identidad (CI).	

3.5.3.1 Planificación Sprint 2

En el sprint dos se van a realizar las tareas referentes a la Administración de profesores, este sprint define 4 tareas a las cuales se las colocó en el orden que van a ser elaboradas, cada una de las tareas de este sprint tomará 4 días, tomando en consideración que las reuniones de los desarrolladores se realizarán cada dos días y al finalizar se harán pruebas de cada sección y se las entregará al cliente, si existen funciones que modificar se realizará en un plazo no mayor a dos días, en el Cuadro 14 se puede visualizar detalladamente el segundo sprint.

Cuadro 14: Sprint N°. 2 “Administración de profesores”

Sprint N°:	2			
Referencia Historial:	Administración de profesores.			
Fecha de Inicio:	06/07/2013		Tareas Pendientes: 0	
Fecha de culminación:	14/07/2013		Días pendientes: 0	
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado
1	Inserción, modificación y eliminación de profesores.	Susana	4	Terminado
2	Modificación de datos personales por parte del profesor.	Susana	4	Terminado
3	Registrar y eliminar asignación de cargo de coordinador a un profesor de tiempo completo.	Diego	4	Terminado
4	Resetear la clave de ingreso al sistema.	Diego	4	Terminado

3.5.4 Sprint 3: “Administración de Mallas curriculares”.

Para establecer el sprint 3, como primer paso se procede a revisar La Historia de usuario que contiene los requisitos funcionales y el nivel de prioridad que el usuario determinó en base a las exigencias del negocio, en el Cuadro 15 se muestra a detalle la Historia de usuario correspondiente a este sprint.

Cuadro 15: Historia de usuario "Administración de Mallas curriculares"

HISTORIA DE USUARIO	
Número: 3	Usuario: coordinador
Nombre Historia: Administración de Mallas curriculares	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Baja
Iteración: 1	
Programador Responsable: Susana	
Descripción: El sistema permita guardar, modificar, visualizar y eliminar los módulos y asignaturas que pertenecen a cada uno de los Programas de Postgrado.	
Como probarlo: El coordinador ingresa al sistema. Guardar la información del módulo y asignatura. Después de registrar los datos del módulo y asignatura se podrá visualizar, modificar y eliminar la información de los módulos y asignaturas si lo creyere conveniente.	
Observaciones del equipo de desarrollo: Antes de registrar un módulo y asignatura de debe previamente crear los Programas de Postgrado. Un módulo puede o no tener asignaturas.	

3.5.4.1 Planificación Sprint 3

El sprint número tres define la tareas sobre la administración de Mallas curriculares, en este sprint se realizará la creación de módulos y asignaturas que pertenecen a los diferentes Programas de Postgrado, este sprint contiene tareas priorizadas, el sprint contiene 5 tareas que serán ejecutadas cada una en 4 días, tomando en consideración que las reuniones de los desarrolladores se realizarán cada dos días y al finalizar se harán pruebas de cada sección y se las entregará al cliente, si existen funciones que modificar se realizará en un plazo no mayor a dos días, a continuación en el Cuadro 16 se detalla el sprint 3.

Cuadro 16: Sprint N° 3 “Administración de Mallas curriculares”

Sprint N°:	3			
Referencia Historial:	Administración de Mallas curriculares.			
Fecha de Inicio:	15/07/2013	Tareas Pendientes: 0		
Fecha de culminación:	27/07/2013	Días pendientes: 0		
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado
1	Inserción, modificación y eliminación de un Programa de Postgrado.	Susana	4	Terminado
2	Inserción, modificación y eliminación de promociones para un Programa de Postgrado.	Susana	4	Terminado
3	Inserción, modificación y eliminación de tipos de información de un módulo o asignatura.	Diego	4	Terminado
4	Inserción, modificación y eliminación de los módulos que pertenecen a cada uno de los Programas de Postgrado.	Diego	4	Terminado
5	Inserción, modificación y eliminación de asignaturas que pertenecen a cada uno de los módulos.	Diego	4	Terminado

3.5.5 Sprint 4: “Administración de alumnos”.

Como primer paso para determinar el sprint cuatro, se procedió a revisar minuciosamente la Historia de usuario que contiene los requisitos funcionales del sistema y el nivel de prioridad, información dada por el cliente en base a las exigencias del negocio y se la detalla en el Cuadro 17.

Cuadro 17: Historia de usuario “Administración de alumnos”

HISTORIA DE USUARIO	
Número: 4	Usuario: administrador, coordinador
Nombre Historia: Administración de Alumnos.	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Baja
Iteración: 1	
Programador Responsable: Susana	
Descripción: El sistema permita guardar, modificar, visualizar y eliminar los alumnos que pertenecen a cada una de las promociones de la Unidad de Postgrados. Registrar la cédula de identidad (CI), apellidos, nombres, correo electrónico, género, ID BANNER, teléfono convencional, celular, fecha nacimiento, fecha graduación pregrado, dirección del trabajo, dirección de casa, especialidad, cursos adicionales de estudio, nacionalidad,	
Como probarlo: El administrador y coordinador Ingresan al sistema. Guardar la información del alumno. Después de registrar los datos del alumno, se podrá visualizar, modificar y eliminar si lo creyere conveniente.	
Observaciones del equipo de desarrollo: No se puede guardar los datos del alumno con una cédula de identidad (CI) repetida.	

3.5.5.1 Planificación Sprint 4

El sprint número cuatro detalla las tareas que son referentes al módulo de administración de alumnos, dichas tareas han sido priorizadas de acuerdo al orden que se van a ir elaborando, al finalizar este sprint se podrá empezar a desarrollar el módulo de asistencias de alumnos que se definirá más adelante en otro sprint, el sprint contiene 6 tareas que serán ejecutadas cada una en 4 días, tomando en consideración que las reuniones de los desarrolladores se realizarán cada dos días y al finalizar se harán pruebas

de cada sección y se las entregará al cliente, si existen funciones que modificar se realizará en un plazo no mayor a dos días, se procede a definir el sprint 4 en el Cuadro 18.

Cuadro 18: Sprint N°. 4 “Administración de alumnos”

Sprint N°:	4			
Referencia Historial:	Administración de alumnos.			
Fecha de Inicio:	28/07/2013	Tareas Pendientes: 0		
Fecha de culminación:	08/08/2013	Días pendientes: 0		
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado
1	Inserción, modificación y eliminación de alumnos por parte del administrador y coordinador.	Diego	4	Terminado
2	Alta de alumnos desde un archivo Excel por parte del administrador y coordinador.	Diego	4	Terminado
3	Modificación de datos personales por parte del alumno.	Diego	4	Terminado
4	Cambio de clave de alumno.	Susana	4	Terminado
5	Validación de datos de entrada en los formularios web.	Susana	4	Terminado
6	Generar reportes de alumnos por promociones.	Susana	4	Terminado

3.5.6 Sprint 5: “Administración de aulas”.

Para la definición del sprint 5 se determinó revisar la Historia de usuario siguiendo el orden de prioridad dada por el cliente, dicha Historia de usuario se detalla en el Cuadro 19.

Cuadro 19: Historia de usuario “Administración de aulas”

HISTORIA DE USUARIO	
Número: 5	Usuario: administrador
Nombre Historia: Administración de aulas.	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Baja
Iteración: 1	
Programador Responsable: Diego	
Descripción: El sistema permita guardar, modificar, visualizar y eliminar la información de las aulas que van a ser asignadas dentro de los cronogramas. La información de las aulas que se debe registrar es la siguiente: tipo de aula, número de aula, la capacidad, características, bloque en donde se encuentra ubicada y el piso.	
Como probarlo: Ingresar al sistema con el rol de administrador. Guardar la información de las aulas. Después de registrar los datos del aula si el administrador desea puede proceder a modificar los datos o eliminarlos según crea conveniente.	
Observaciones del equipo de desarrollo: El tipo de aula puede ser aula normal, audiovisuales o laboratorio.	

3.5.6.1 Planificación Sprint 5

Para este sprint, las tareas se las dividirá en dos secciones las cuales tomarán 4 días cada uno, cabe mencionar que las reuniones de los desarrolladores se realizarán cada dos días y al finalizar se harán pruebas de cada sección y se las entregará al cliente, si existen funciones que modificar se realizará en un plazo no mayor a dos días, a continuación en el Cuadro 20 se puede ver a detalle el sprint 5.

Cuadro 20: Sprint N°. 5 “Administración de aulas”

Sprint N°:	5			
Referencia Historial:	Administración de aulas			
Fecha de Inicio:	09/08/2013		Tareas Pendientes: 0	
Fecha de culminación:	13/08/2013		Días pendientes: 0	
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado
1	Inserción, modificación y eliminación de Tipos de aula	Susana	4	Terminado
2	Inserción, modificación y eliminación de aulas	Diego	4	Terminado

3.5.7 Sprint 6: “Administración de cronograma de clases”.

Para definir el sprint 6 se procedió a revisar la Historia de usuario detallada en el Cuadro 21, que contiene los requisitos funcionales del sistema con su nivel de prioridad definido por el cliente.

Cuadro 21: Historia de usuario “Administración de cronograma de clases”

HISTORIA DE USUARIO	
Número: 6	Usuario: coordinador
Nombre Historia: Administración de cronograma de clases.	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Baja
Iteración: 1	
Programador Responsable: Diego	
Descripción: El sistema permita registrar, modificar, visualizar y eliminar la información de los cronogramas de clases tales como: promoción, módulos, asignaturas, aula, profesor, hora de inicio de clases, hora de finalización de clases y fecha.	
Como probarlo: Ingresar al sistema con el perfil de coordinador. Guardar la información del cronograma de clases.	
Observaciones del equipo de desarrollo: Si la promoción ha completado su vigencia; es decir si la fecha actual es mayor a la fecha de finalización de la promoción, el sistema ya no permitirá crear cronogramas.	

3.5.7.1 Planificación Sprint 6

Para este sprint, las tareas se las dividirá en cuatro secciones las cuales tomarán dos días cada uno, cabe mencionar que las reuniones de los desarrolladores se realizarán cada dos días y al finalizar se harán pruebas de cada sección y se las entregará al cliente, si existen funciones que modificar se realizará en un plazo no mayor a dos días, a continuación en el Cuadro 22 se puede visualizar el sprint 6.

Cuadro 22: Sprint N°.6 “Administración de cronograma de clases”

Sprint N°:	6			
Referencia Historial:	Administración de cronograma de clases			
Fecha de Inicio:	14/08/2013		Tareas Pendientes: 0	
Fecha de culminación:	18/08/2013		Días pendientes: 0	
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado
1	Inserción, modificación y eliminación de un cronograma de clases.	Susana	2	Terminado
2	Realizar listas desplegables dinámicas, si se escoge un módulo la lista desplegable de asignaturas se actualizará de acuerdo al módulo de la primera lista desplegable	Susana	2	Terminado
3	Emitir reportes de cronogramas.	Diego	2	Terminado
4	Realizar una utilidad para consultar el horario de clases para el profesor y alumno.	Diego	2	Terminado

3.5.8 Sprint 7: “Registro y modificación de asistencias de profesores y alumnos”.

Como primer paso se procedió a revisar minuciosamente la Historia de usuario que se encuentra en el Cuadro 23, la cual define los requisitos

funcionales para el módulo de Administración de asistencias de profesores.

Cuadro 23: Historia de usuario “Administración de asistencias de profesores”

HISTORIA DE USUARIO	
Número: 7	Usuario: profesor
Nombre Historia: Registro y modificación de asistencias de profesores y alumnos.	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Baja
Iteración: 1	
Programador Responsable: Diego	
Descripción: El sistema permita timbrar la asistencia de entrada y de salida e inmediatamente acceda a la lista de alumnos para registrar la asistencia de cada uno de ellos.	
Como probarlo: Ingresar al sistema con el perfil de profesor. Seleccionar la opción entrada o salida e inmediatamente registrar la asistencia de los alumnos.	
Observaciones del equipo de desarrollo:	

3.5.8.1 Planificación sprint 7

Para este sprint, las tareas se las dividirá en cinco secciones, las cuales tomaran 2 días cada una, cabe mencionar que las reuniones de los desarrolladores se realizarán cada dos días y al finalizar se harán pruebas de cada sección y se las entregará al cliente, si existen funciones que modificar se realizará en un plazo no mayor a dos días, a continuación en el Cuadro 24 se puede visualizar a detalle el sprint 7.

Cuadro 24: Sprint N°. 7 “Administración de asistencias”

Sprint N°:	7			
Referencia Historial:	Registro y modificación de asistencias de profesores y alumnos.			
Fecha de Inicio:	19/08/2013	Tareas Pendientes: 0		
Fecha de culminación:	25/08/2013			
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado
1	Realizar la inserción de timbre de entrada y salida	Susana	2	Terminado
2	Validar que pueda registrar su entrada los días que tiene un horario de clases	Susana	2	Terminado
3	Realizar un reporte de asistencias de profesores.	Diego	2	Terminado
4	Inserción y modificación de asistencias de alumnos por parte del profesor y coordinador.	Diego	2	Terminado
5	Realizar un reporte de asistencias de alumnos.	Diego	2	Terminado

3.5.9 Sprint 8: “Administración de e-mails”

Como primer paso para la definición del sprint 8 se procedió a revisar la Historia de usuario correspondiente, la cual se encuentra en el Cuadro 25.

Cuadro 25: Historia de usuario "Administración de e-mails"

HISTORIA DE USUARIO	
Número: 8	Usuario: coordinador
Nombre Historia: Administración de e-mails.	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Baja
Iteración: 1	
Programador Responsable: Diego	
Descripción: El coordinador permita responder a los mensajes enviados por los alumnos.	
Como probarlo: Ingresar al sistema con el perfil de coordinador. Seleccionar el mensaje del alumno y contestar el mismo.	
Observaciones del equipo de desarrollo: Es un mini buzón de mensajes que son enviados por el alumno y recibidos por el coordinador del Programa de Postgrado.	

3.5.9.1 Planificación sprint 8

El sprint número ocho detalla dos tareas correspondientes al módulo de administración de emails, el tiempo de desarrollo será de 4 días cada tarea, tomando en consideración que las reuniones de los desarrolladores se realizarán cada dos días y al finalizar se harán pruebas de cada sección y se las entregarán al cliente, si existen funciones que modificar se realizarán en un plazo no mayor a dos días, a continuación en el Cuadro 26 se define el sprint 8.

Cuadro 26: Sprint N°. 8 “Administración de e-mails”

Sprint N°:	8			
Referencia Historial:	Administración de e-mails			
Fecha de Inicio:	26/08/2013		Tareas Pendientes: 0	
Fecha de culminación:	30/08/2013		Días pendientes: 0	
Orden de priorización	Descripción de la tarea	Responsable	Duración días	Estado
1	Envío de solicitudes para realizar algún trámite personal por parte del alumno al coordinador del Programa de Postgrado al cual pertenece.	Susana	4	Terminado
2	Recepción de solicitudes por parte del coordinador.	Diego	4	Terminado

CAPÍTULO 4: ANÁLISIS Y DISEÑO DEL SISTEMA

4.1 PRELIMINAR

En este capítulo se detallará una planeación de cómo se procedió a la toma de especificación de requerimientos considerando el estándar IEEE 830, como estas fueron analizadas y puestas en marcha para el desarrollo del software.

Es importante mencionar el proceso de control de asistencia de profesores y alumnos que actualmente se lleva a cabo en la Unidad de Gestión de Postgrados.

- El coordinador crea los cronogramas, estos se los envía a los alumnos por correo electrónico.
- El coordinador del Programa de Postgrado asigna un horario de clases y le entrega una lista de alumnos al profesor.
- El profesor toma lista a sus alumnos, anota su hora de llegada en una hoja impresa y lo devuelve al coordinador para que éste lo revise.

El coordinador revisa la lista y verifica que alumnos tienen problemas en faltas, dicha revisión le ayudará a determinar si un alumno aprueba o no el módulo o asignatura con respecto al cumplimiento de asistencias, dado que en el Art 153 del Reglamento de estudiantes de la Universidad de las Fuerzas Armadas - ESPE establece que para obtener un título de magíster es necesario aprobar mínimo 960 horas equivalente a 60 créditos y para alcanzar un título de Doctor 1440 horas equivalente a 90 créditos.

En la Figura 5 se visualiza la situación actual.

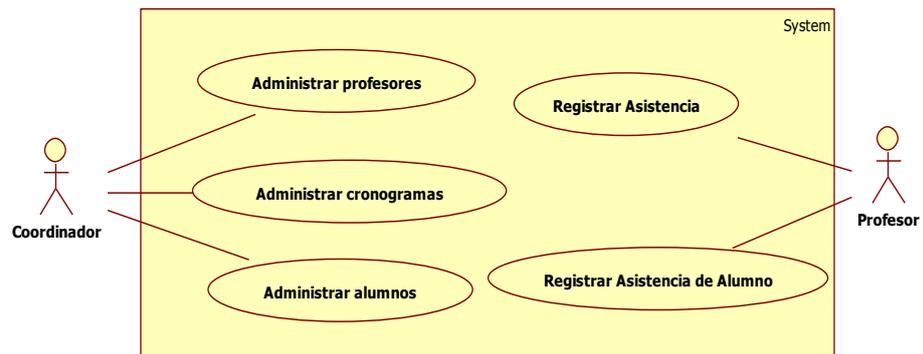


Figura 5: Situación actual del proceso de control de asistencia

La propuesta que se pone en marcha es tener una base de datos de alumnos, profesores, manejados por un software que brinde resultados de las asistencias a sus horas de clases.

El coordinador crea un cronograma y automáticamente se envía al correo electrónico de cada alumno informando la asignación.

El profesor y el alumno pueden revisar todos los cronogramas en el sistema, la asistencia se registrará dentro del sistema y éste emitirá un reporte al coordinador determinando las horas asistidas del alumno y profesor. En la Figura 6 se visualiza la situación propuesta.

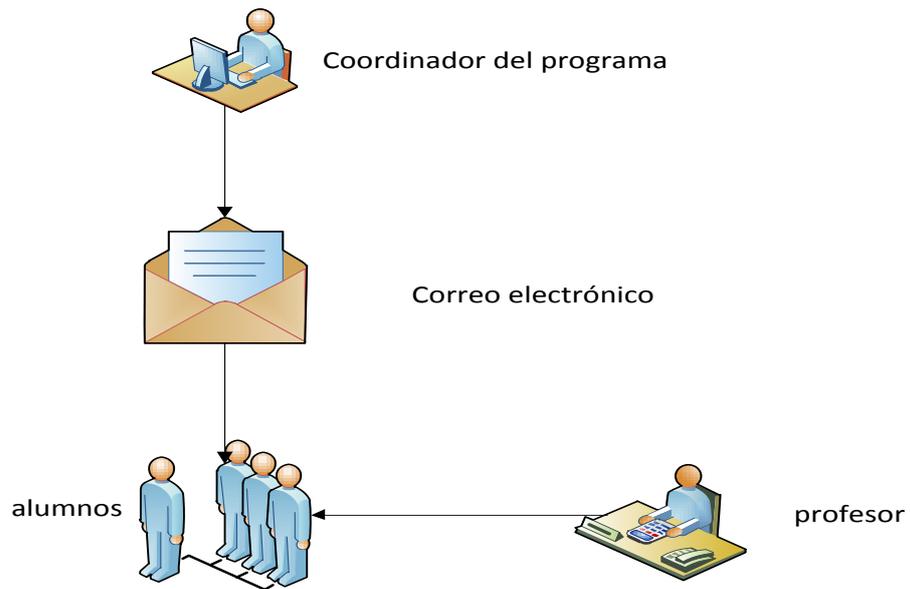


Figura 6: Situación propuesta toma de asistencia a los alumnos.

También se detallarán los diagramas de Casos de Uso con su respectiva descripción, Diagramas de paquetes, Diagramas de Distribución, Diagrama lógico de la base de datos, Diagrama de clases, notaciones que se utilizaron en cuanto a la programación y la estructura de la pantalla (interfaz de usuario) genérica que se utilizó para la construcción del Software.

4.2 PLANEACIÓN

4.2.1 Toma de requerimientos

Para realizar este proyecto se empezó con la toma de requerimientos del cliente. Estos requerimientos se encuentran en el documento de Especificación de Requerimientos del Software AsistSoft, la cual consta en el **ANEXO 1**.

4.3 DIAGRAMAS DE CASOS DE USO

Un caso de uso es una descripción de las acciones de un sistema desde el punto de vista del usuario.

Todo aplicativo necesita ser detallado de forma precisa y puntual lo que cada actor realizará en el sistema, el objetivo de un diagrama de casos de uso es detallar el comportamiento del sistema.

Para tener una especificación de forma ordenada se procede a realizar una división lógica del proyecto de la siguiente manera:

- Módulo de usuarios, asignación de permisos.
- Módulo de profesores.
- Módulo de Mallas curriculares.
- Módulo de alumnos.
- Módulo de aulas.
- Módulo de cronogramas.
- Módulo de asistencias de profesores y alumnos.
- Módulo de e-mails.
- Módulo de emisión de reportes.

A continuación en la Figura 7 se visualiza el diagrama de casos de uso general del software.

4.3.1 Diagrama de casos de uso del sistema AsistSoft

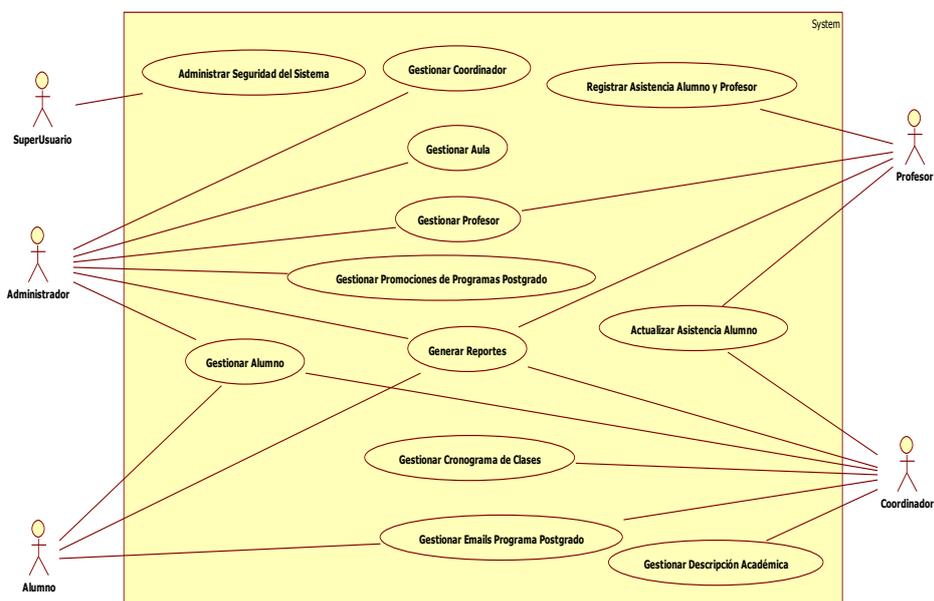


Figura 7: Diagrama de casos de uso del sistema AsistSoft

Para mayor comprensión del diseño general de casos de uso del sistema AsistSoft, los diagramas y las descripciones correspondientes a cada caso de uso se encuentra en el **ANEXO 2**.

4.4 DISEÑO DE INTERFACES

Es la parte de una aplicación que el usuario observa y con la cual interactúa, la interfaz incluye las pantallas, ventanas, controles, menús, la ayuda en línea, la documentación y el entrenamiento (ESCOBAR, 2008).

A continuación en el Cuadro 27 se da a conocer algunos principios básicos que se deben tomar en cuenta al momento de diseñar una interfaz de usuario.

Cuadro 27: Principios de Diseño de las interfaces de usuario

Principio	Descripción
Familiaridad del usuario	La interfaz debe utilizar términos y conceptos obtenidos de la experiencia de las personas que más utilizan el sistema.
Uniformidad	Siempre que sea posible, la interfaz debe ser uniforme en el sentido de que las operaciones comparables se activen de la misma forma.
Mínima sorpresa	El comportamiento del sistema no debe provocar sorpresa a los usuarios.
Recuperabilidad	La interfaz debe incluir mecanismos para permitir a los usuarios recuperarse de los errores.
Guía de usuario	Cuando ocurran errores, la interfaz debe proporcionar retroalimentación significativa y características de ayuda sensible al contexto.
Diversidad de usuarios	La interfaz debe proporcionar características de interacción apropiadas para los diferentes tipos de usuarios del sistema.

Fuente: (Sommerville, 2005)

A continuación se presentan las pantallas más representativas del sistema AsistSoft.

Son muy importantes especificarlas ya que el uso correcto dará el menor porcentaje de errores cuando se las utilice y bajo el cuadro detallado de los principios básicos de construcción de interfaces web, el sistema AsistSoft cuenta con las siguientes pantallas:

4.4.1 Pantalla de Login

La Figura 8 muestra la pantalla inicial de acceso al sistema, en la cual los usuarios del sistema tendrán que autenticarse mediante el ingreso del número de cédula de identidad (CI) y la clave.

Figura 8: Pantalla de Login

4.4.2 Pantallas: “Perfil administrador”

4.4.2.1 Pantalla Gestión de profesores

La Figura 9 muestra un formulario en el cual se debe ingresar la información de los profesores de la Unidad de Gestión de Postgrados.

Figura 9: Pantalla Gestión de profesores

4.4.2.2 Pantalla Gestión de alumnos

El administrador del sistema podrá ingresar la información de los alumnos de cada una de las promociones de la Unidad de Gestión de Postgrados, en la Figura 10 se muestra el formulario en donde se ingresará dicha información.

The screenshot shows the 'ADMINISTRACIÓN ALUMNOS' interface. On the left is a sidebar with a tree view under 'UNIDAD DE POSTGRADOS' containing 'ADMINISTRACIÓN' and various management options. The main area is a form with the following fields:

- Promoción: * (Dropdown menu: Seleccione una Promoción)
- Cédula: * (Text input)
- Apellido(s): * (Text input)
- Nombre(s): * (Text input)
- Correo Electrónico: * (Text input)
- Género: * (Radio buttons: MASCULINO, FEMENINO)
- ID BANNER: (Text input)
- Teléfono Convencional: (Text input)
- Celular: (Text input)
- Fecha de Nacimiento: (Text input)

ASISTSOFT

Figura 10: Pantalla Gestión de alumnos

4.4.2.3 Pantalla Gestión de coordinadores

En la Figura 11 se puede observar la interfaz, donde el administrador del sistema podrá asignar a un profesor el cargo de coordinador de un Programa de Postgrado.

The screenshot shows the 'ADMINISTRACIÓN COORDINADORES' interface. On the left is the same sidebar as in Figure 10. The main area is a form with the following elements:

- Código: (Text input)
- Profesor Candidato: * (Dropdown menu: Seleccione el Profesor)
- Buttons: Asignar, Cancelar
- Table with columns: Profesor, Cédula

Profesor	Cédula
H	
HOMERO GERMÁN TINOCO MATAMOROS	1800385997

ASISTSOFT

Figura 11: Pantalla Gestión de coordinadores

4.4.2.4 Pantalla Gestión de módulos y asignaturas

La Figura 12 muestra la Pantalla, donde se ingresarán los módulos y asignaturas que pertenecen a cada uno de los Programas de Postgrado.

Nombre Módulo	Número Créditos:	Tipo:	
AUDITORÍA EN TECNOLOGÍA I (GESTIÓN, METODOLOGÍA Y SEGURIDAD)	2	Técnico	
AUDITORÍA EN TECNOLOGÍA II (SISTEMAS Y SOLUCIONES)	2	Técnico	
AUDITORÍA EN TECNOLOGÍA III (SISTEMAS DE COMUNICACIÓN)	2	Técnico	

ASISTISOFT

Figura 12: Pantalla Administración de módulos y asignaturas

4.4.2.5 Pantalla Gestión de Promociones

El administrador del sistema podrá registrar, modificar y eliminar una promoción perteneciente a cada uno de los Programas de Postgrado, su interfaz gráfica se muestra en la Figura 13.

MAESTRÍA EN AUDITORÍA Y EVALUACIÓN DE SISTEMAS						
Código:	Nombre Promoción	Paralelo:	N. - Estudiantes Estimado:	Número Real de estudiantes:	Fecha Inicio:	Fecha Fin:

Figura 13: Pantalla Gestión de Promociones

4.4.3 Pantallas: “Perfil alumno”

4.4.3.1 Pantalla Información Personal

En la Pantalla que se puede observar en la Figura 14, los alumnos de cada una de las promociones de la Unidad de Gestión de Postgrados podrán modificar su información personal.

Figura 14: Pantalla Información Personal alumno

4.4.3.2 Pantalla Enviar Solicitudes

El alumno podrá enviar solicitudes al coordinador del Programa de Postgrado al cual pertenece, el formulario donde deberá ingresar la información se muestra en la Figura 15.

Fecha	Asunto	Redacción	Respuesta
2014-02-22	Calificaciones	notas	En Proceso...
2014-02-22	diego	diego hola	En Proceso...
2014-02-21	hoy	ssssedggggggg	En Proceso...

Figura 15: Pantalla Enviar Solicitudes

4.4.4 Pantallas “Perfil profesor”

4.4.4.1 Registrar Asistencia

En la Figura 16 se muestra la pantalla donde el profesor de la Unidad de Gestión de Postgrados podrá registrar la asistencia de entrada o salida, en la parte superior del formulario se puede observar un reloj que indica la hora actual, de esta manera el profesor podrá conocer la hora de registro de su asistencia.

BIENVENIDO: VICTOR PALIZ OSORIO [Cerrar Sesión](#)

UNIDAD DE POSTGRADOS 00:05:35 12.11.2013

REGISTRE SU ASISTENCIA

Cédula: 1708034622

Opciones:

Entrada Salida

ASISTSOFT

Figura 16: Pantalla Registrar Asistencia

4.4.4.2 Información Personal profesor

En el formulario que se muestra en la Figura 17, el profesor de la Unidad de Gestión de Postgrados podrá modificar su información personal.

UNIDAD DE POSTGRADOS

Información Personal

CAMPOS OBLIGATORIOS (*)

Cédula:* 1709434243

Nombre(s):* JAIME

Apellido(s):* AYALA TACO

Correo Electrónico:*

Titulo Pregrado:*

Teléfono Convencional:*

Celular:*

Direccion Casa:*

Dirección Trabajo:*

Figura 17: Pantalla Información Personal profesor

4.4.5 Pantallas “Perfil coordinador”

4.4.5.1 Pantalla Asignar cronograma

En el formulario que se muestra en la Figura 18, el coordinador del Programa de Postgrado podrá asignar un cronograma.

Figura 18: Pantalla Asignar cronograma

Para el manejo de esta herramienta web se detalla la descripción del funcionamiento del sistema AsistSoft en el **ANEXO 3**.

4.5 DIAGRAMA DE PAQUETES

Los diagramas de paquetes se usan para reflejar la organización de las clases del sistema.

Es importante visualizar la agrupación de las clases de un proyecto de software, las cuales están agrupadas o empaquetadas para ayudar a entender mejor la estructura lógica del Software. Las dependencias de estos paquetes son vitales para el paso de información de cada clase.

En este apartado se detallará el diagrama de paquetes que se estableció para la construcción de software y una breve explicación de que función realiza cada una.

Antes de la explicación de cada una de ellas se realizará una descripción de cómo está siendo usada la terminología de cada objeto dentro del proyecto.

El proyecto está dividido en dos partes principales:

- Proyecto EJB
- Proyecto WEB

Proyecto EJB

Cada paquete dentro del Proyecto EJB está denominado de la siguiente manera: paq_nombre del paquete.

Cada clase dentro de un paquete está denominada de la siguiente manera:

- Para el paquete “paq_clase interfaz”:
 - interfaz_nombre de la clase.java
- Para el paquete “paq_entidades”:
 - enti_nombre de la clase.java
- Para el paquete “paq_negocio”:
 - neg_nombre de la clase.java
- Para el paquete “paq_sentencias”:
 - Ejecutarsql.java

Proyecto Web

Cada paquete dentro del Proyecto Web está denominado de la siguiente manera: paq_nombre del paquete.

Cada clase dentro de un paquete está denominada de la siguiente manera:

- Para el paquete “paq_cálculo”:
 - Nombre de la clase.java
- Para el paquete “paq_mensajes”:
 - mensajes.properties
- Para el paquete “paq_entidades”:
 - entiv_ nombre de la clase.java
- Para el paquete “paq_reportes”:
 - Vista_nombre de la clase.java
 - Reporte_nombre del reporte.jasper
- Para el paquete “paqv_vista”:
 - Vista_nombre de la clase.java

Para cada página web (interfaz de usuario) lleva un nombre siguiendo el estándar de la siguiente manera:

Nombre_pantalla.xhtml

El diagrama de paquetes del software se describe en la Figura 19.

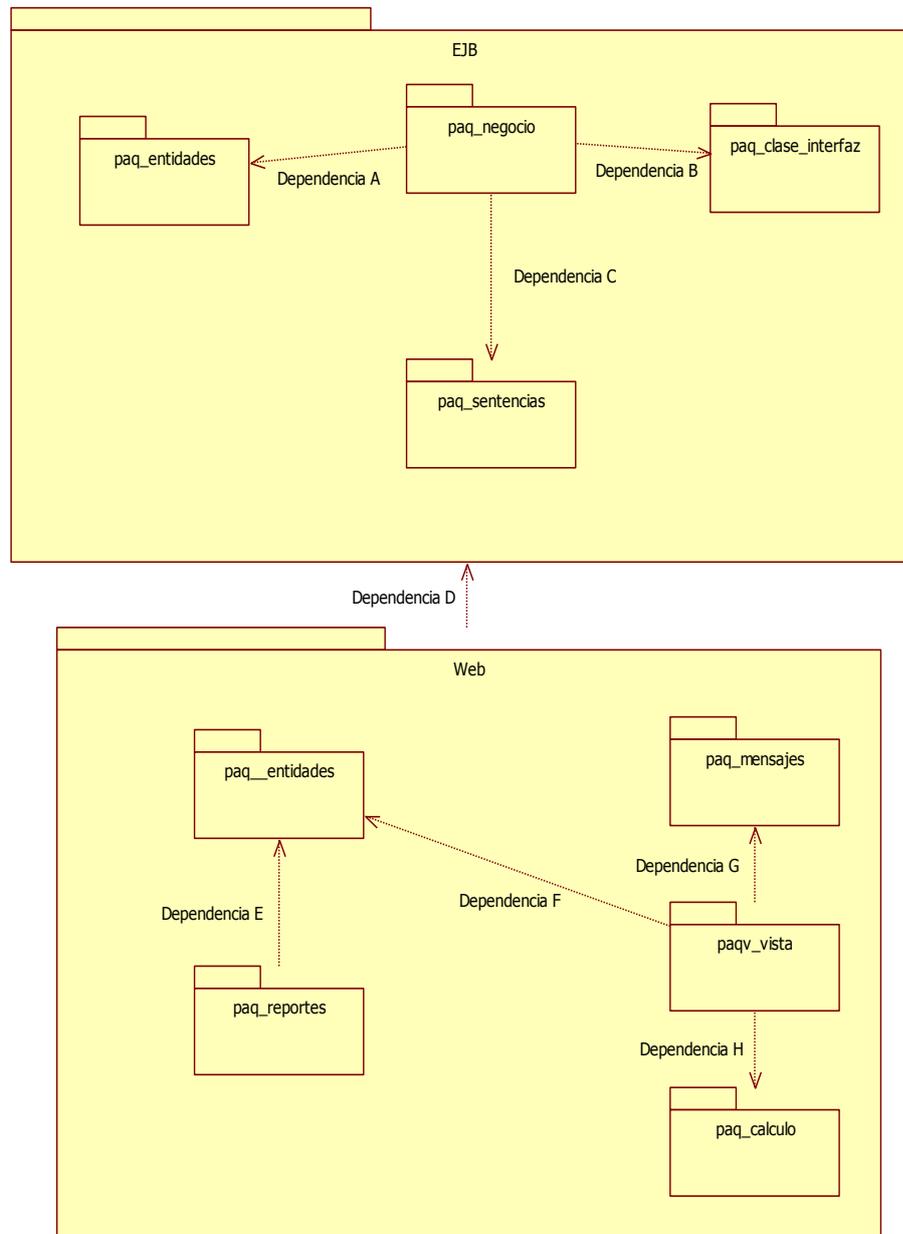


Figura 19: Diagrama de Paquetes

4.5.1 Paquete EJB

Este paquete abarca todo lo que respecta al negocio del sistema de forma general, la cual consta de cuatro paquetes como se explican a continuación:

`paq_entidades`.- Reúne las clases entidades de cada objeto.

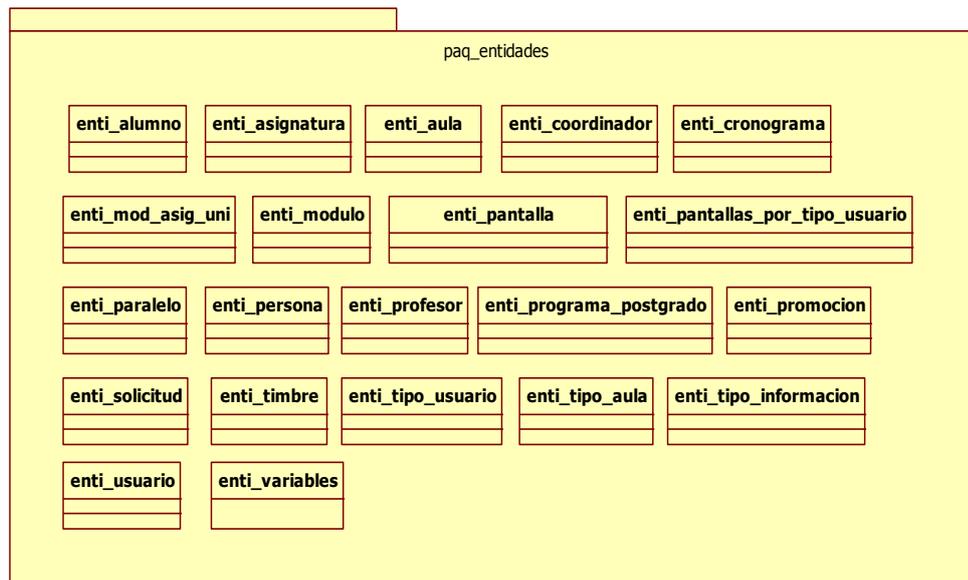


Figura 20: Estructura Paquete EJB

paq_negocio.- Reúne las clases que ejecutan las sentencias tipo SQL a la base de datos del sistema.

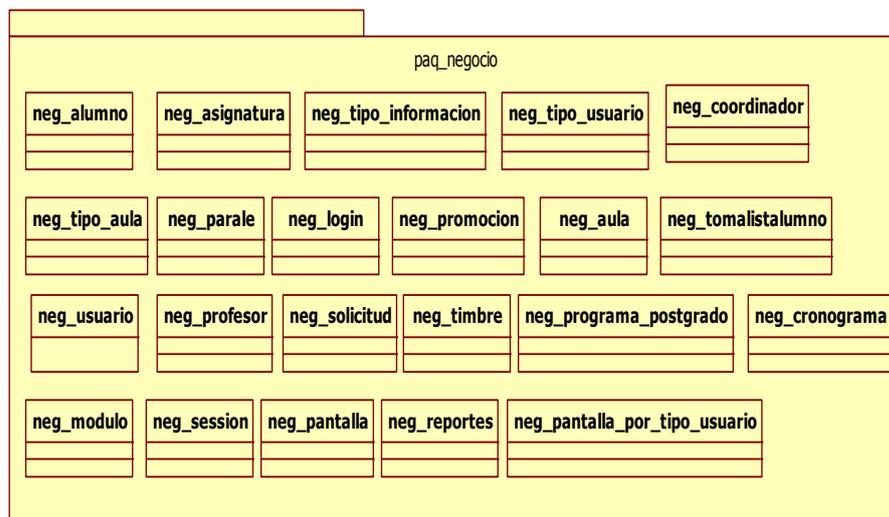


Figura 21: Estructura Paquete Negocio

Cada clase de este paquete depende del funcionamiento que tengan las clases de los paquetes “paq_clase_interfaz”, “paq_entidades” y

“paq_sentencias”, las dependencias se detallan a continuación:

Dependencia A.- El paquete paq_negocio utilizará las propiedades de las clases del paquete “paq_entidades”.

Dependencia B.- El paquete “paq_negocio” utilizará los métodos de las clases del paquete “paq_clase_interfaz”.

Dependencia C.- El paquete “paq_negocio” utilizará los métodos de conexión a la base de datos de las clases del paquete “paq_sentencias”.

paq_clase_interfaz.- Reúne las clases que poseen todos los métodos o funciones. En la Figura 22 se puede visualizar este paquete.

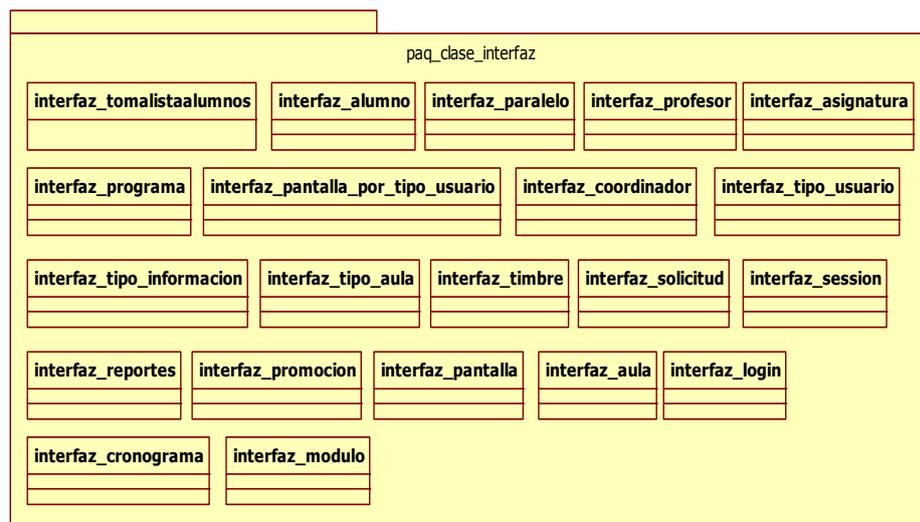


Figura 22: Estructura Paquete Interfaz

paq_sentencias.- Tiene una clase que realiza la conexión a la base de datos del sistema. En la Figura 23 se puede visualizar este paquete.

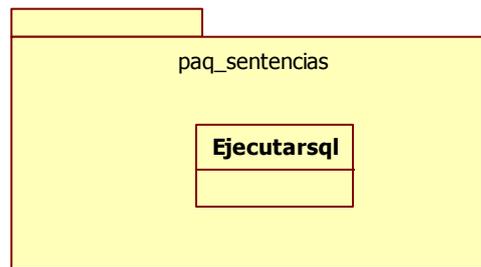


Figura 23: Estructura Paquete Sentencias

4.5.2 Paquete Web

Este paquete abarca todo lo que respecta a la interfaz gráfica del sistema de forma general la cual consta de cinco paquetes:

paq_entidades.- Reúne las clases entidades de cada objeto tal como se visualiza en la Figura 24.

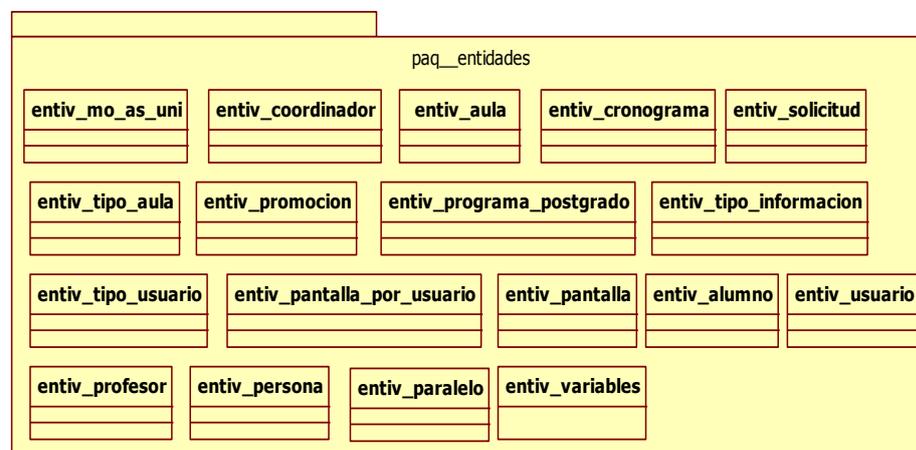


Figura 24: Estructura Paquete Web

Paq_mensajes.- Aquí se encuentran las etiquetas de botones, variables, campos de texto, etc. de las páginas web. En la Figura 25 se puede visualizar este paquete.

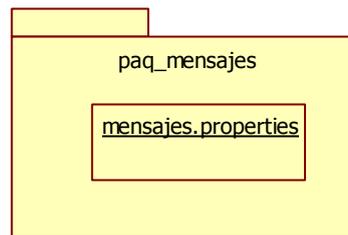


Figura 25: Estructura Paquete Mensajes

Paq_cálculo.- Se encuentran las clases que contienen funciones para validar campos de texto y las que realizan algún cálculo matemático. En la Figura 26 se puede visualizar este paquete.

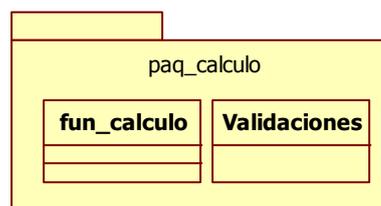


Figura 26: Estructura Paquete Cálculo

Paq_reportes.- Se encuentran las clases que contienen los reportes y métodos para enviarlos a la web. En la Figura 27 se puede visualizar este paquete.

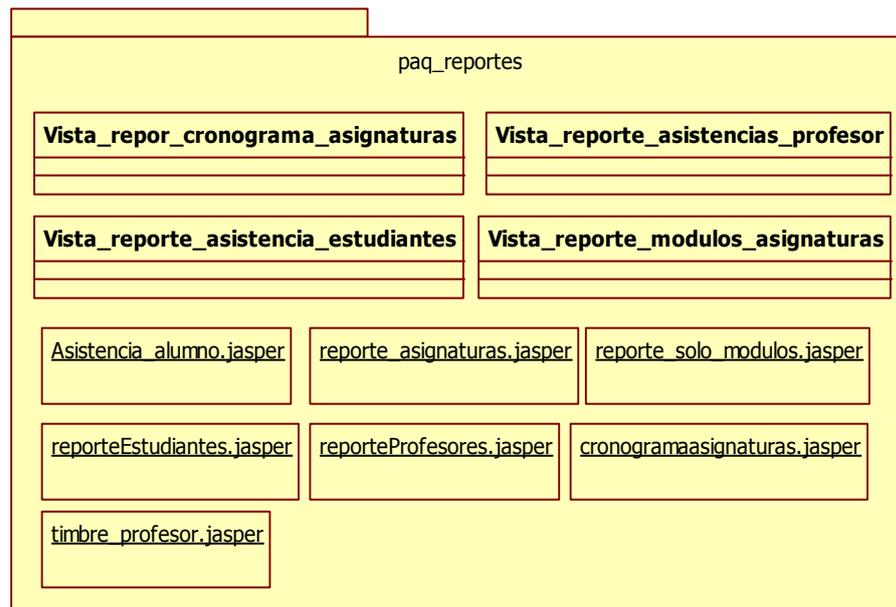


Figura 27: Estructura Paquete Reportes

Cada clase de este paquete depende del funcionamiento que tengan las clases de los paquetes “paq_entidades”, las dependencias se detallan a continuación:

Dependencia E.- El paquete “paq_reportes” utilizará las propiedades de las clases del paquete “paq_entidades”.

paqv_vista.- Reúne las clases que son la interfaz gráfica del usuario y que son llamadas por las páginas web. En la Figura 28 se puede visualizar este paquete.

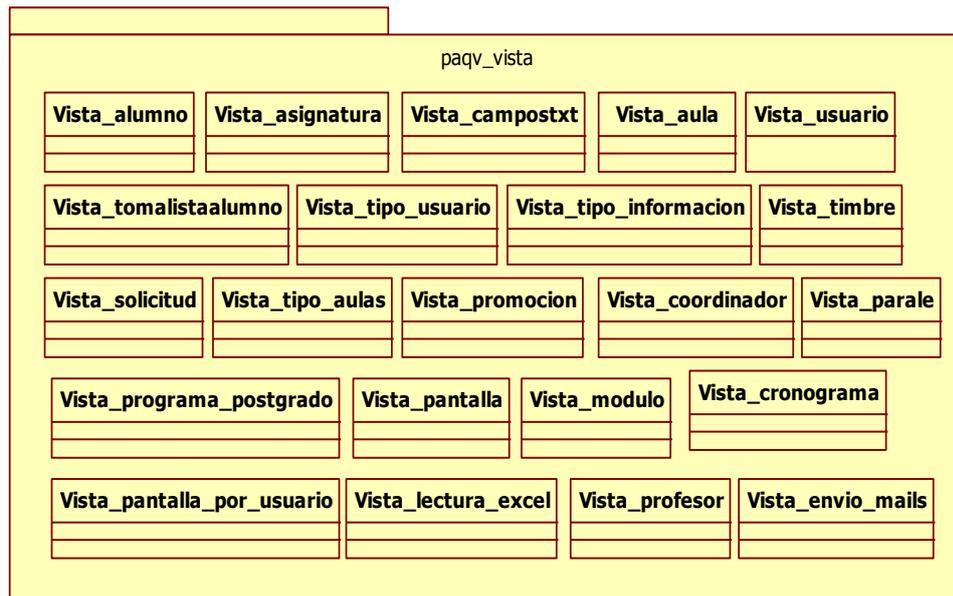


Figura 28: Estructura Paquete Vista

Cada clase de este paquete depende del funcionamiento que tengan las clases de los paquetes “paq_entidades”, “paq_mensajes” y “paq_cálculo”, las dependencias se detallan a continuación:

Dependencia F.- El paquete “paqv_vista” utilizará las propiedades de las clases del paquete paq_entidades.

Dependencia G.- El paquete “paqv_vista” utilizará los mensajes del paquete “paq_mensajes” y estas principalmente se encuentran en las páginas web.

Dependencia H.- El paquete “paqv_vista” utilizará las funciones de validación de las clases del paquete “paq_cálculo”.

4.6 DIAGRAMA DE DISTRIBUCIÓN

El diagrama de distribución UML muestra la arquitectura física de un sistema informático. Puede representar los equipos y dispositivos, mostrar sus interconexiones y el software que se encontrará en cada máquina. (Sommerville, 2005)

El hardware es un punto muy importante de una aplicación, representar como estará estructurada es muy primordial para entender mejor el funcionamiento a nivel de nodos de conexión. En esta parte del documento se detallará como es la estructura para la comunicación de computadores clientes con el servidor donde estará alojado el aplicativo y la base de datos. En la Figura 29 se puede visualizar el Diagrama de Distribución.

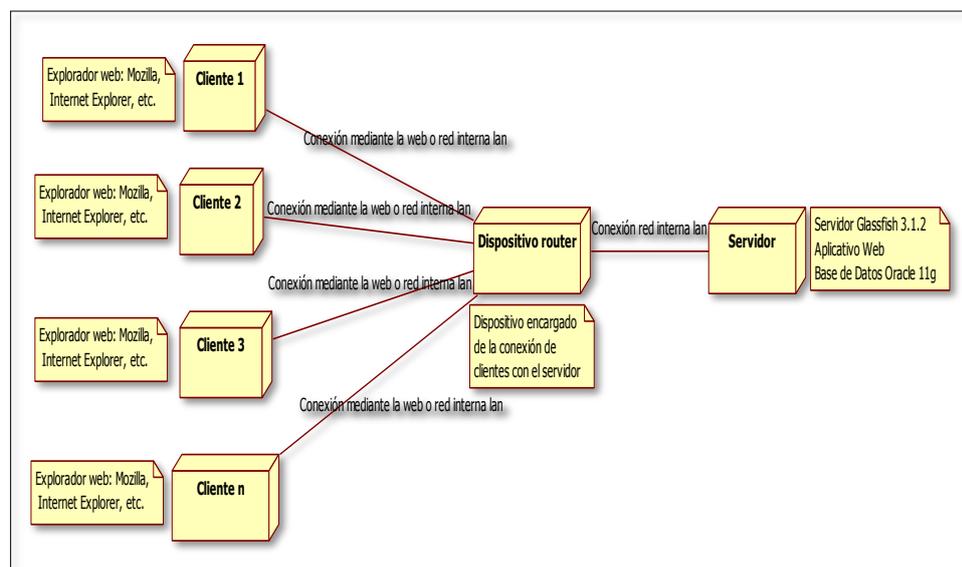


Figura 29: Diagrama de Distribución

Nodo Servidor.- Aquí se encuentra el servicio Glassfish, el aplicativo web y la base de datos Oracle 11g edición libre.

Nodo Dispositivo router.- Aquí se encuentra los dispositivos que están encargados de la conexión del servidor con los clientes.

Nodo Cliente.- Clientes que se conectarán vía web o red LAN interna al router de conexión y este a su vez al servidor.

4.7 DIAGRAMA DE CLASES

Este diagrama representa los objetos fundamentales del sistema a construirse. La clase define el ámbito de definición de un conjunto de objetos por lo tanto cada objeto pertenece a una clase.

La clase se representa con una figura rectangular dividida en 3 partes que contiene el nombre de la clase, los atributos y las operaciones de la clase respectivamente. A continuación se presenta el Diagrama de Clases del sistema AsistSoft correspondiente a cada módulo:

4.7.1 Diagrama de Clases: Administración de aulas

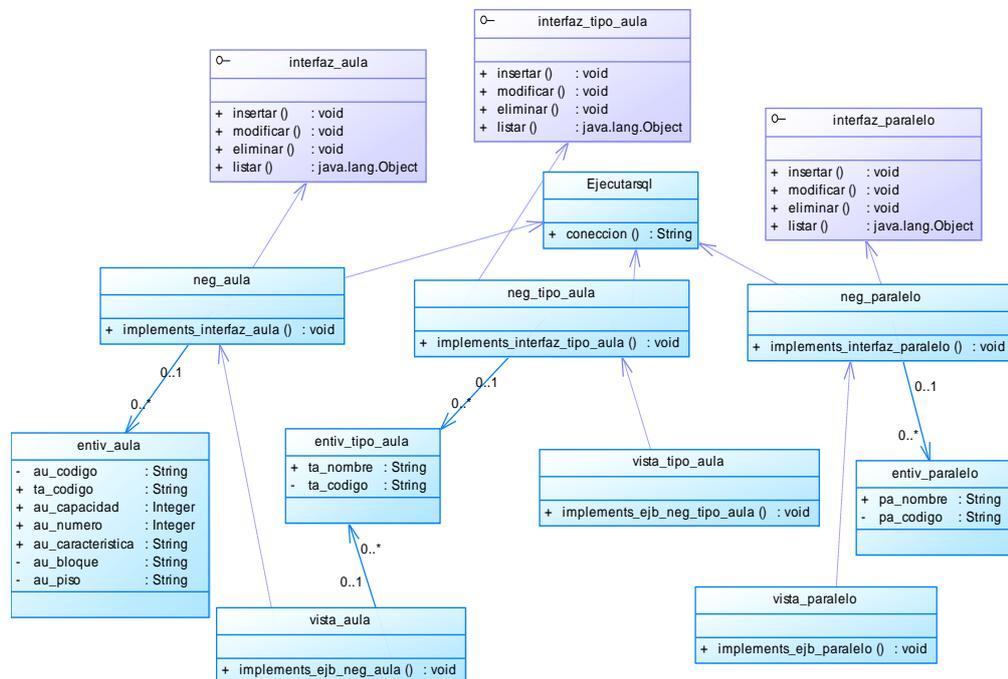


Figura 30: Diagrama de Clases “Administración de aulas”

4.7.2 Diagrama de Clases: Administración de cronogramas

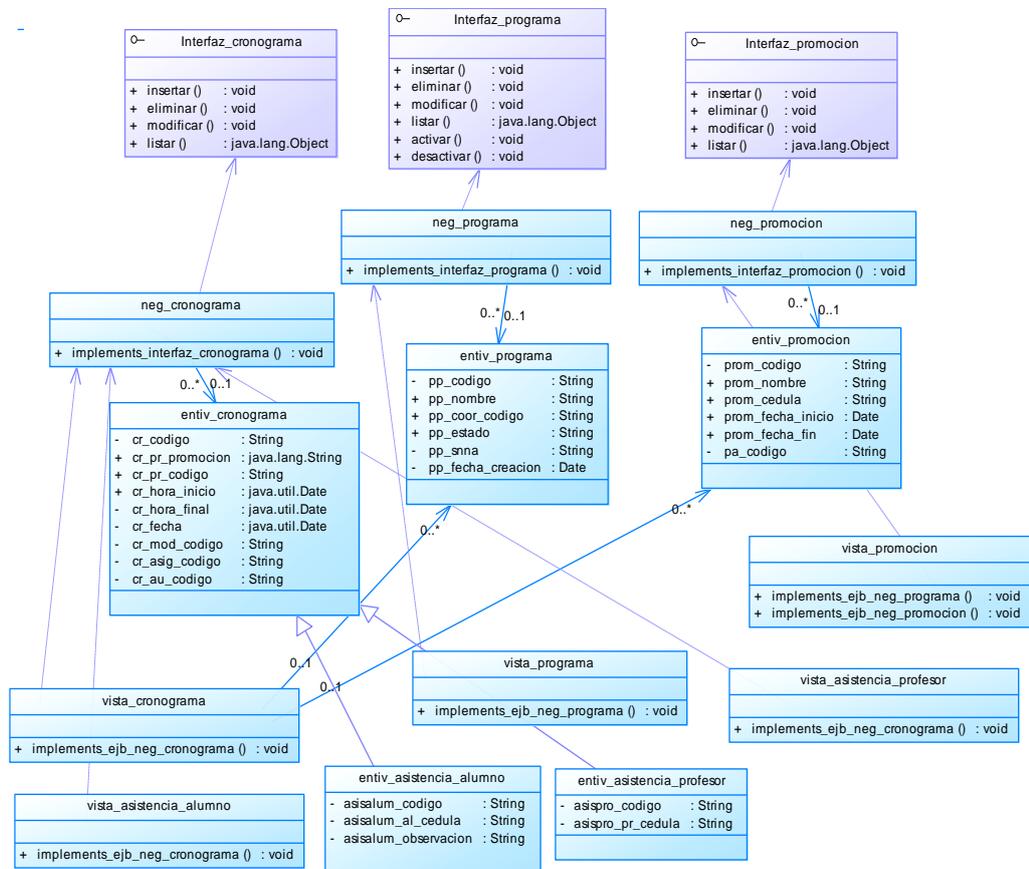


Figura 31: Diagrama de Clases “Administración de cronogramas”

4.7.3 Diagrama de Clases: Administración de mallas curriculares

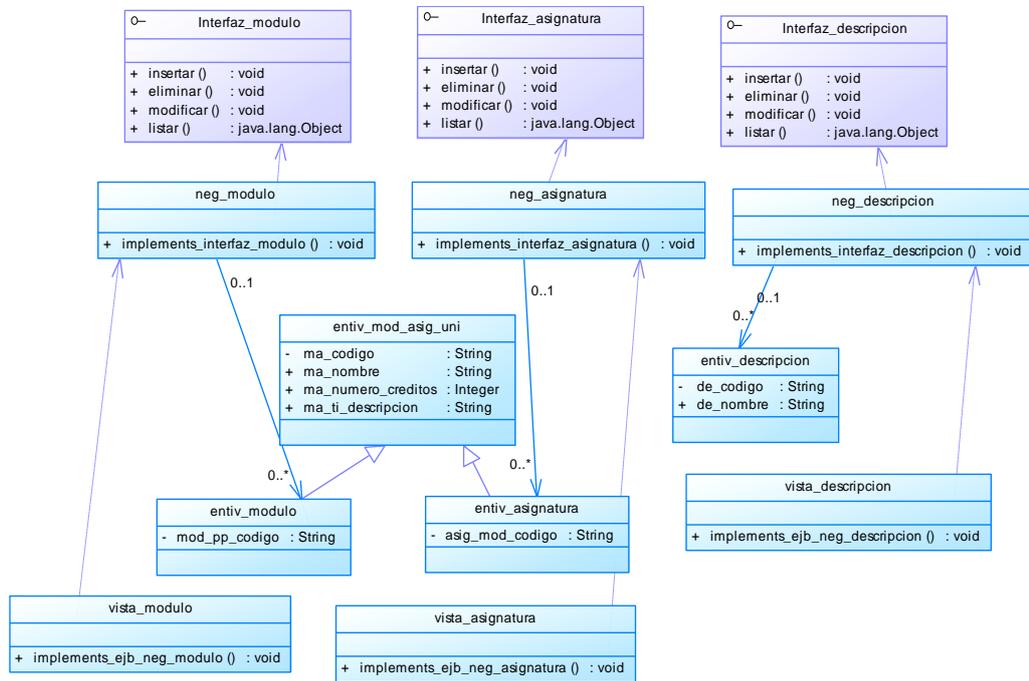


Figura 32: Diagrama de Clases “Administración de mallas curriculares”

4.7.4 Diagrama de Clases: Administración de profesores, coordinadores, alumnos y usuarios

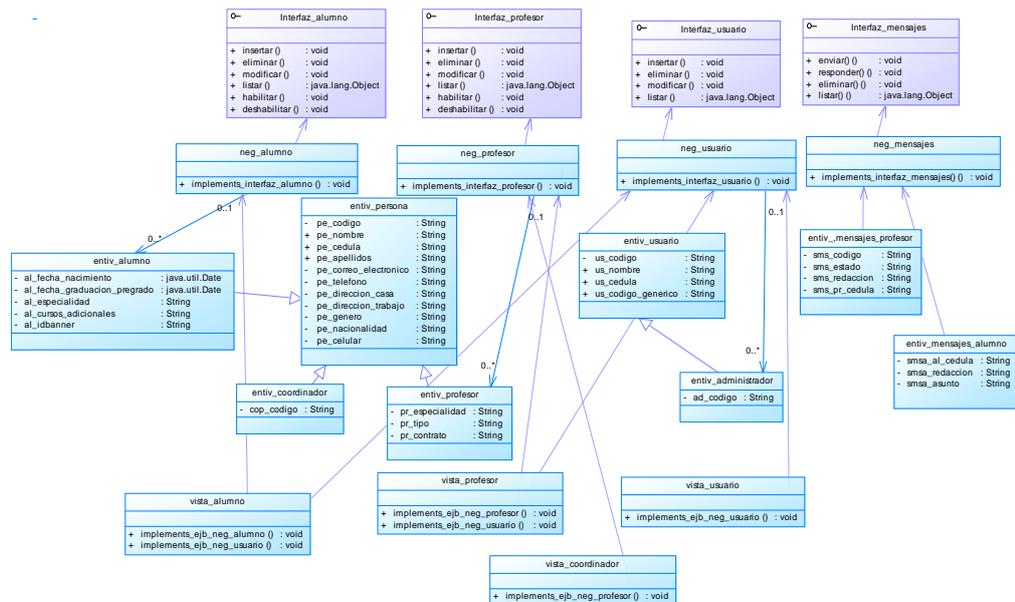


Figura 33: Diagrama de Clases “Administración de profesores, coordinadores, alumnos y usuarios”

4.8 DIAGRAMA LÓGICO DE LA BASE DE DATOS

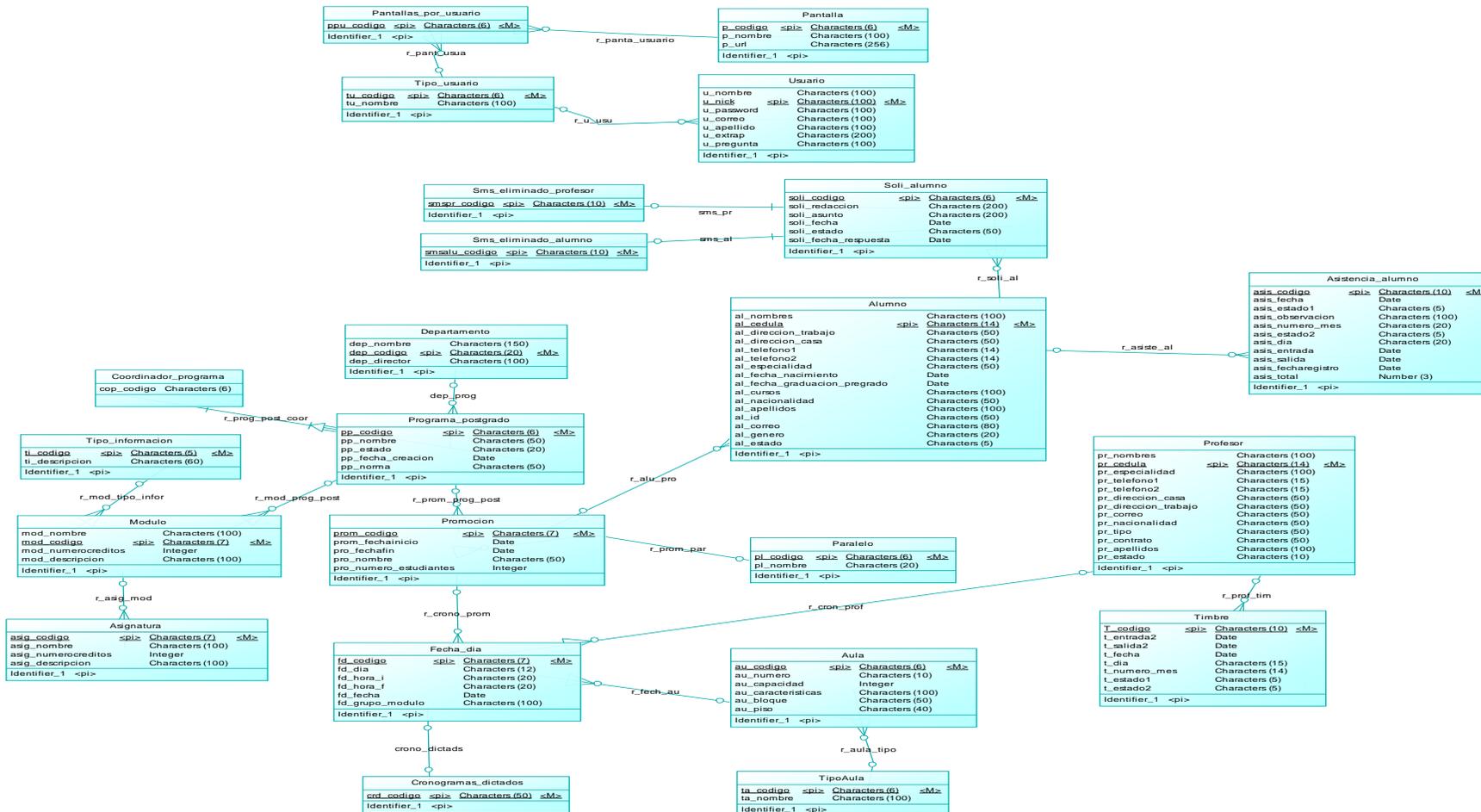


Figura 34: Diagrama Lógico de la Base de Datos

CAPÍTULO 5: PRUEBAS DEL SISTEMA

5.1 INTRODUCCIÓN

En esta etapa se ha examinado la calidad del producto que se desarrolló, teniendo como objetivo principal encontrar la mayor cantidad de errores y defectos del sistema, para que los mismos luego de ser detectados se corrijan de forma minuciosa y tener como resultado un sistema de alta calidad que cumpla con los requisitos funcionales detallados inicialmente en el documento de especificación de requerimientos.

Siguiendo la metodología Scrum en la fase de desarrollo se dio mayor énfasis al proceso de pruebas ya que durante cada uno de los sprints el producto era diseñado, codificado y testeado.

Para que las pruebas sean más eficaces se la realizará con la participación directa de los usuarios del sistema.

5.2 CASOS DE PRUEBA

5.2.1 Prueba del caso de Uso: Ingresar al sistema

Aquí se detalla las acciones del usuario para ejecutar esta prueba, esta prueba se visualiza en el Cuadro 28.

Cuadro 28: Caso de prueba: Ingresar al sistema

Identificador:	1
Referencia de Caso de Uso:	Ingresar al sistema
Usuario:	Profesor
Descripción:	El usuario registrado previamente en la base de datos ingresa al sistema.
Variables de entrada:	Datos de entrada: Cédula de identidad (CI): Clave Generica:Espe.2013
Flujo normal del evento:	En el navegador ir a la dirección web de la aplicación. En la página de ingreso digitar la cédula de identidad (CI) y clave. Presionar sobre el botón Entrar, para ingresar al sistema.
Resultado esperado:	El sistema valida la información ingresada por el usuario y presenta la ventana principal con el menú de opciones que le corresponde de acuerdo al perfil de usuario.
Evaluación de prueba	
Ejecutado por:	Susana – Diego
Lugar de ejecución:	Laboratorios de Computación ESPE.
Resultados Obtenidos:	El usuario ingresa al sistema sin ningún problema.
Observaciones:	El control de validación para la autenticación del usuario es correcto.
Notas del Programador	
Estado:	Terminado
Acciones de Corrección:	Ninguna
Corregido por:	-

5.2.2 Prueba del caso de Uso: Actualizar Información Personal

Aquí se detalla las acciones del usuario para ejecutar esta prueba la cual se visualiza en el Cuadro 29.

Cuadro 29: Caso de Prueba: Registrar Información Personal profesor

Identificador:	2
Referencia de Caso de Uso:	Registrar Información Personal profesor.
Usuario:	Profesor
Descripción:	Proceso en el cual los profesores de la Unidad de Gestión de Postgrados actualizarán sus datos personales dentro del sistema.
Variables de entrada:	Datos de entrada: Correo electrónico, Título Pregrado, Teléfono Convencional, Celular, Dirección Casa, Nacionalidad:
Flujo normal del evento:	Ingresar al sistema con el perfil de profesor. Seleccionar del menú la opción Información Personal. Llenar todos los campos requeridos en el formulario. Hacer clic sobre el botón Actualizar.
Resultado esperado:	El sistema valida los datos ingresados y muestra el mensaje: “Transacción Exitosa”.
Evaluación de prueba	
Ejecutado por:	Susana – Diego
Lugar de ejecución:	Laboratorios de Computación ESPE.
Resultados Obtenidos:	El sistema muestra un mensaje de error de la base de datos con respecto al extenso tamaño de texto ingresado por el usuario en los siguientes campos: Dirección Casa, Dirección Trabajo.
Observaciones:	El control que valida que los campos requeridos estén llenos funciona correctamente.
Notas del Programador	
Estado:	Terminado.
Acciones de Corrección:	Aumentar tamaño de los campos: Dirección Trabajo y Dirección Casa en la base de datos. Definir un número máximo de caracteres de entrada en los campos del formulario.
Corregido por:	Susana

5.2.3 Prueba del caso de Uso: Cambiar Contraseña

Aquí se detalla las acciones del usuario para ejecutar esta prueba, la cual se visualiza en el Cuadro 30.

Cuadro 30: Caso de Prueba: Cambiar Contraseña

Identificador:	3
Referencia de Caso de Uso:	Cambiar Contraseña.
Usuario:	Profesor
Descripción:	Proceso en el cual el profesor cambiará su clave de acceso al sistema.
Variables de entrada:	Datos de entrada: Contraseña actual, Contraseña nueva, Confirmar contraseña, Pregunta y respuesta secreta:
Flujo normal del evento:	Ingresar al sistema con el perfil de profesor. Seleccionar del menú la opción Cambiar Clave. Ingresar los datos de entrada requeridos para el cambio de contraseña. Hacer clic sobre el botón Guardar Cambios.
Resultado esperado:	El sistema valida los datos ingresados y muestra el mensaje: "Transacción Exitosa".
Evaluación de prueba	
Ejecutado por:	Susana – Diego
Lugar de ejecución:	Laboratorios de Computación ESPE.
Resultados Obtenidos:	Los usuarios logran modificar su contraseña correctamente.
Observaciones:	La validación que controla que las entradas contraseña nueva y confirmar contraseña sean idénticas y su tamaño sea mayor a 6 caracteres funciona correctamente. La validación que controla que todos los campos estén llenos funciona correctamente.
Notas del Programador	
Estado:	Terminado.
Acciones de Corrección:	Ninguna.
Corregido por:	-

5.2.4 Prueba del caso de Uso: Consultar cronograma

Aquí se detalla las acciones del usuario para ejecutar esta prueba, la cual se visualiza en el Cuadro 31.

Cuadro 31: Caso de Prueba: Consultar cronograma

Identificador:	4
Referencia de Caso de Uso:	Consultar cronograma.
Usuario:	Profesor
Descripción:	Proceso en el cual los profesores de la Unidad de Gestión de Postgrados consultan su cronograma de clases.
Variables de entrada:	Ninguno.
Flujo normal del evento:	Ingresar al sistema con el perfil de profesor. Hacer clic sobre la opción Consultar Cronograma.
Resultado esperado:	El muestra la información detallada del cronograma de clases correspondiente al usuario en sesión.
Evaluación de prueba	
Ejecutado por:	Susana – Diego
Lugar de ejecución:	Laboratorios de Computación ESPE.
Resultados Obtenidos:	La consulta del cronograma de clases por parte del usuario se realizó correctamente
Observaciones:	Ninguna.
Notas del Programador	
Estado:	Terminado
Acciones de Corrección:	Ninguna.
Corregido por:	-

5.2.5 Prueba del caso de Uso: Registrar Asistencia profesor y alumno

Aquí se detalla las acciones del usuario para ejecutar esta prueba, la cual se visualiza en el Cuadro 32.

Cuadro 32: Caso de Prueba: Registrar Asistencia profesor y alumno

Código de Identificación:	5
Referencia de Caso de Uso:	Registrar Asistencia profesor y alumno.
Usuario:	Profesor.
Descripción:	Proceso en el cual los profesores de la Unidad de Gestión de Postgrados registrarán su asistencia y la de sus alumnos.
Variables de entrada:	Datos de entrada: Entrada o Salida.
Flujo normal del evento:	Ingresar al sistema con el perfil de profesor. Seleccionar del menú la opción Registrar Asistencia. Timbrar la entrada o salida. Hacer clic sobre el botón Registrar.
Resultado esperado:	El sistema valida la información ingresada y muestra la lista de alumnos para el registro de su asistencia.
Evaluación de prueba	
Ejecutado por:	Susana – Diego
Lugar de ejecución:	Laboratorios de Computación ESPE.
Resultados Obtenidos:	El registro de asistencia de los profesores y alumnos fue realizado con éxito.
Observaciones:	Ninguna.
Notas del Programador	
Estado:	Terminado
Acciones de Corrección:	Ninguna
Corregido por:	-

Las encuestas aplicadas para evaluar la aceptación de los usuarios se encuentran en el **ANEXO 4**.

5.3 PRUEBA DE CONCURRENCIA

En base a una prueba realizada en varios laboratorios se procedieron a conectar al sistema varios usuarios, después de tomar el tiempo que demoraba en una transacción y visualizar si los registros se insertaban en la base de datos se llega a la conclusión que la velocidad con la que trabaja el aplicativo fue aceptable según las exigencias del cliente, a continuación en el

Cuadro 33 se puede observar una descripción.

Cuadro 33: Resultado Velocidad de Transacción

Acciones	Velocidad de transacción en acciones por usuario en segundos de tiempo	Cantidad de Usuarios
Insertar	3	120
Modificar	1	5
Eliminar	1	5
Consulta	3	120

Se estima de acuerdo a la información del cliente que el número de personas conectadas concurrentemente al sistema sea de hasta 50.

5.4 EVALUACIÓN DE RESULTADOS

Una vez realizadas las pruebas del sistema y aplicadas las encuestas a los usuarios reales se obtuvo los resultados que se muestran en la Figura 35.

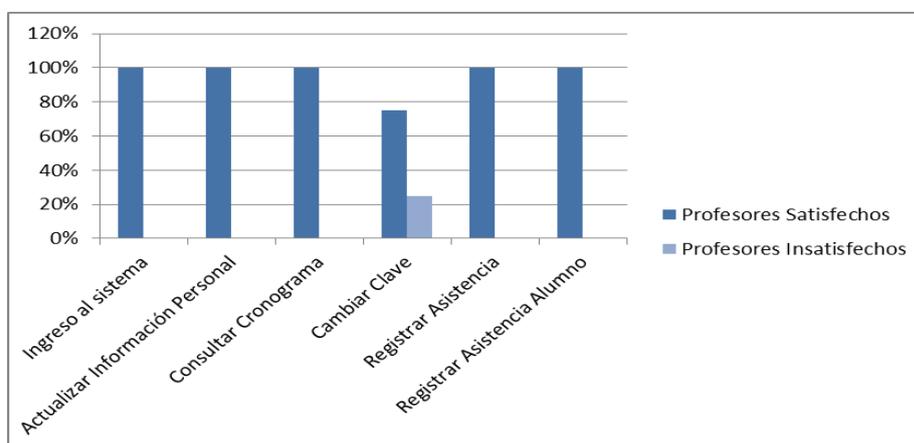


Figura 35: Resultados de la encuesta aplicada a los usuarios

CAPITULO 6: CONCLUSIONES Y RECOMENDACIONES

6.1 CONCLUSIONES.

Los requerimientos extraídos de forma concreta y que no quede la mínima especificación sin haberse entendido de forma correcta, clara y precisa, ayudó mucho al desenvolvimiento de la construcción del sistema.

La forma de organización para la construcción del proyecto software se siguió de acuerdo a un cronograma establecido gracias a la metodología Scrum, que ésta aportó mucho en la construcción del producto software que es lo que más le interesaba al usuario.

En lo que se refiere a la construcción del sistema con un previo análisis se optó por utilizar la tecnología EJB 3.0, que aportó mucho a la programación aportando con comunicación sencilla con entidades de objetos y funciones que comunican con la base de datos.

Las pruebas que se realizaron en el proyecto con datos reales fueron satisfactorias en temas como recurrencia de usuarios, transaccionalidad, validación de datos y navegación entre pantallas web, esto aportó a tener una visión amplia en lo que respecta a la actuación del sistema a futuro.

La capacitación en los módulos del sistema que se hizo a los usuarios fue de forma sencilla, dinámica y gráfica, lo cual ayudó al entendimiento sobre el uso del sistema.

6.2 RECOMENDACIONES.

Cuando se empieza a construir un sistema Software cabe destacar que existen muchas guías que ayudan de forma considerable a realizar la especificación de requerimientos y es importante optar por seguir sus normas de forma disciplinada ya que esto aportará de forma profesional y normativa el éxito de un proyecto.

Realizar un estudio previo de la situación actual en la Institución es importante ya que permite elegir un tipo de metodología ágil que le otorgue al cliente una primera versión del producto.

Utilizar la tecnología EJB 3.0 cuando se desee mejorar la versión actual del sistema, ya que será más fácil añadir funcionalidades y reutilizar funciones que se han puesto como genéricas, ayudará a no consumir mucho tiempo al momento de realizar cambios en el sistema.

Realizar pruebas de todos los módulos durante el proceso de construcción de un sistema es importante ya que así se logrará verificar tiempos de respuesta, validación y consistencia de datos ingresados al sistema, esto es una práctica que se deberá tomar en cuenta como una fase de desarrollo.

Siempre se considera la parte de capacitación al usuario, por tal motivo preparar un ambiente adecuado, los procesos a seguir y si es posible con diagramas, son elementos importantes que las personas inmersas en el proyecto deben considerarlo en la práctica, esto ayuda a que el usuario entienda de la mejor forma las explicaciones para el uso de un sistema.

Bibliografía

- Areba, J. B. (2001). *Metodología del análisis estructurado de sistemas*. Madrid: Belen Rocio.
- Cobo, Á. (2005). *Tecnología para el desarrollo de aplicaciones web*. España: Diaz de Santos.
- ESCOBAR, G. J. (10 de Noviembre de 2008). *DISEÑO DE INTERFAZ DE USUARIO*. Recuperado el 15 de Abril de 2013, de DISEÑO DE INTERFAZ DE USUARIO: <http://diseodeinterfazdeusuario.blogspot.com/>
- Gabillaud, J. (2010). *Recursos Informáticos Oracle 11g - SQL, PL/SQL, SQL*Plus*. Barcelona: ENI.
- Gioacchini, R. (Septiembre de 2011). *Test Lead - SOGETI*. Recuperado el 15 de Enero de 2013, de Test Lead - SOGETI: <http://www.es.sogeti.com/PageFiles/173/Introducci%C3%B3n%20a%20SCRUM.pdf>
- GROUSSARD, T. (2010). *Desarrollo de aplicaciones web con JEE 6*. Barcelona: Editions ENI.
- Groussard, T. (2010). *Recursos Informáticos Java Enterprise Edition - Desarrollo de aplicaciones web con JEE 6*. Barcelona: Ediciones ENI.
- IBM. (22 de Noviembre de 2010). Recuperado el 12 de Mayo de 2013, de SCRUM como metodología: <https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/Rational+Team+Concert+for+Scrum+Projects/page/SCRUM+como+metodolog%C3%ADa>
- JASPERSOFT. (07 de 11 de 2013). *JASPERSOFT COMMUNITY*.

- Recuperado el 06 de 09 de 2013, de JASPERSOFT COMMUNITY:
<http://community.jaspersoft.com/project/jasperreports-library>
- Mann, M. (2008). *Ingeniería del Software*. [s.n]: [s. n.].
- Mora, S. L. (2002). *Programación de aplicaciones web: historia, principios básicos y clientes web*. España: Editorial Club Universitario.
- Netbeans. (01 de 01 de 2013). *Netbeans*. Recuperado el 08 de 09 de 2013, de Netbeans: http://netbeans.org/index_es.html
- Pérez, J. (9 de 02 de 2012). *Software Libre para UML* . Recuperado el 07 de 09 de 2013, de Software Libre para UML : <http://fundamentos-ing-requisitos.blogspot.com/2012/02/software-libre-para-uml.html>
- Salazar, D. (05 de 08 de 2012). *Galeon.com*. Recuperado el 05 de 08 de 2013, de Galeon.com: <http://salazardaniela.galeon.com/>
- Schwaber, K. (2010). *Agile Project Management with Scrum*. New York: Kansas.
- Scrum Manager*. (1 de Abril de 2014). Recuperado el 5 de Abril de 2014, de Gestión de Proyectos Scrum Manager: http://www.scrummanager.net/files/sm_proyecto.pdf
- Sierra, K. (02 de Enero de 2011). *Introducción a la tecnología EJB*. Recuperado el 04 de Enero de 2012, de Introducción a la tecnología EJB: <http://www.jtech.ua.es/j2ee/2003-2004/abierto-j2ee-2003-2004/ejb/sesion01-apuntes.htm>
- SOFTENG. (13 de 10 de 2012). *SOFTENG SOFTWARE ENGINEERS*. Recuperado el 04 de 08 de 2013, de SOFTENG SOFTWARE ENGINEERS: <http://www.softeng.es/es-es/empresa/metodologias-de->

trabajo/metodologia-scrum.html

Sommerville, I. (2005). *Ingeniería del software*. Madrid: PEARSON

EDUCACIÓN S.A.

SQL DEVELOPER. (03 de 04 de 2013). ORACLE. Recuperado el 02 de 04

de 2013, de ORACLE: <http://www.oracle.com/us/sqldeveloper>

Weitzenfeld, A. (2007). *Ingeniería de software orientada a objetos con UML,*

Java e Internet. Madrid: Ediciones Cantara.