

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA



PROYECTO DE GRADO

**“SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE
PERSONAL
CASO: UNIVERSIDAD TECNOLÓGICA BOLIVIANA”**

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS

Postulante: María Juana Aguilar Torrez

Tutor: Lic. Fatima Consuelo Dolz de Moreno M. Sc.

Revisor: Lic. Aldo Ramiro Valdez Alvarado

LA PAZ – BOLIVIA
2011

Dedicatoria:

A mis padres Elizabeth y Florencio, quienes son la principal inspiración de superación y fortaleza en mi vida, a mis hermanos: Giovanni, María Antonieta, Milton y Beatriz por su apoyo constante y confianza, a mi pequeña sobrina Irasema por sus ánimos y ternura.

Con mucho cariño

María Juana

Agradecimientos

A nuestro Dios sobre todas las cosas por concederme la vida y la salud para culminar una etapa más en mi vida.

Agradecer a mi mamá por darme todo su apoyo y comprensión incondicional para que salga adelante, a mi papá que aunque no esté a mi lado sé que siempre me ha estado cuidando, a mi hermano Giovanni que es como mi papá por su comprensión y apoyo, a mi hermana María Antonieta que ha sido mi amiga y muchas veces mi consejera, a mis hermanos Milton y Beatriz por darme ánimos para continuar estudiando.

A la Lic. Fatima Dolz de Moreno M. Sc. docente tutor, por sus consejos y por guiarme a largo del desarrollo de este proyecto.

Al Lic. Aldo Ramiro Valdez Alvarado, docente revisor, por brindarme su apoyo incondicional, tiempo dedicado y por su paciencia, que me ayudaron a culminar este proyecto.

A los docentes de la carrera de Informática.

Agradecer también a la Dirección Administrativa y Dirección de Sistemas de la UTB, especialmente a la Lic. Stephany Díaz Flores por darme la oportunidad de realizar este proyecto.

Al Lic. Juan Rodrigo Aguilar Tenorio por su apoyo incondicional y alentarme en todo momento para no desmayar y seguir adelante.

A mis amigos que siempre me han estado apoyando y ayudando durante mi estancia en la ciudad La Paz.

De todo corazón muchas gracias a todos. . .

RESUMEN

El presente proyecto fue desarrollado en la Universidad Tecnológica Boliviana (UTB) en la Dirección Administrativa, que realiza los procesos de registro de personal, cálculo de planilla de sueldo, control de horarios de entrada y salida de los empleados, las actualizaciones de sus hojas de vida, registro de contratos, movimiento que realizan los empleados (Altas, bajas, permisos, vacación, liquidación)

La dirección Administrativa no cuenta con un sistema de información automático es decir todo lo realizan manualmente lo que ocasiona volúmenes de papelería con información.

Por lo mencionado anteriormente se desarrolló un Sistema Web de Control y Seguimiento de Personal, optimizando el trabajo en el tiempo de procesos y además llevar un control adecuado de la información.

El producto obtenido cuenta con todas las características requeridas por los usuarios, resultando una herramienta de ayuda para los procesos que se efectúan en la Dirección Administrativa de la UTB.

Este proyecto fue realizado con la metodología de desarrollo de software del Proceso Unificado Ágil denotado por el acrónimo AUP para el análisis y diseño del sistema, para el modelado del sistema se utilizó la propuesta de Ingeniería Web basado en UML UWE, utilizando la herramienta case Magic Draw para representar los diferentes objetos o diagramas necesarios para el modelado del sistema.

La Dirección de Sistemas de la UTB trabaja sobre la plataforma Windows, es por este motivo que el sistema fue desarrollado con ASP.NET 2008, como lenguaje de programación se utilizó C# y como gestor de base de datos se utilizó SQL server 2005.

La calidad del sistema fue medida con la norma ISO 9126 con todas las características necesarias, también se efectuó una descripción de las amenazas del sistema y la aplicación de medidas de seguridad para el funcionamiento adecuado del sistema.

INDICE GENERAL

	Pág.
CAPITULO I	1
MARCO INTRODUCTORIO	1
1.1 Introducción.....	1
1.2 Antecedentes.....	2
1.2.1 Antecedentes Institucionales	2
1.2.1.1 Misión.....	2
1.2.1.2 Visión.....	3
1.2.1.3 Dirección Administrativa.....	3
1.2.2 Trabajos Similares	3
1.3 Planteamiento de Problema	5
1.3.1 Problema Principal	5
1.3.2 Problemas Específicos.....	5
1.4 Objetivos.....	6
1.4.1 Objetivo Principal.....	6
1.4.2 Objetivos Específicos	6
1.5 Justificación	7
1.5.1 Económica	7
1.5.2 Social	7
1.5.3 Técnica.....	7
1.6 Límites y alcances	8
1.6.1 Alcances.....	8
1.6.2 Límites	8
1.7 Aportes	8
1.8 Metodología.....	9
CAPITULO II	11
MARCO TEÓRICO	11
2.1 Ingeniería de Software.....	11
2.2 Sistema	12
2.3 Metodología de desarrollo	12
2.3.1 Modelo en cascada.....	12
2.3.2 Método iterativo	14
2.3.3 Método evolutivo.....	14
2.3.4 Método ágil.....	15
2.3.5 Metodología ágil AUP	16
2.3.5.1 Estructura del proceso unificado Ágil (AUP).....	16
2.3.5.2 Disciplinas de AUP.....	17
2.3.5.3 Fases del AUP.....	19
2.3.5.4 Disciplina de Modelado.....	20

a)	Flujo de Trabajo	20
b)	Fase Por Fase	20
2.3.5.5	Disciplina de la implementación	23
a)	Flujo de trabajo	23
b)	Fase por fase	24
2.3.5.6	Disciplina de Pruebas	25
a)	Flujo de Trabajo	25
b)	Fase Por Fase	26
2.3.5.7	Disciplina de administración de la configuración	27
a)	Flujo de trabajo	27
b)	Fase por fase	28
2.3.5.8	Disciplina Administración de proyecto	29
a)	Flujo de trabajo	29
b)	Fase por Fase	29
2.3.5.9	Disciplina del entorno	32
a)	Flujo de trabajo	32
b)	Fase por fases	33
2.4	Ingeniería web	34
2.4.1	Metodología de modelado UWE	34
2.4.2	UWE y su relación con UML	35
2.4.2.1	Modelo de casos de Uso	35
2.4.2.2	Modelo de contenido	36
2.4.2.3	Modelo de Navegación	37
2.4.2.4	Modelo de Presentación	37
2.4.2.5	Modelo de proceso	38
a)	Modelo de Estructura de Procesos	38
2.4.2.6	Modelo de adaptación	39
2.5	Personal	39
2.6	Factores de Calidad ISO 9126	40
CAPITULO III		44
MARCO PRÁCTICO		44
3.1	Introducción	44
3.2	Fase de inicio	44
3.2.1	Modelado del negocio	45
3.2.1.1	Modelado de casos de uso del negocio	45
3.2.1.2	Descripción de actores del caso de uso de negocio	46
3.2.2	Modelado de requerimientos	46
3.2.2.1	Descripción de requerimientos a nivel de negocios	47
3.2.2.2	Descripción de requerimientos a nivel de Usuario	47
3.2.2.3	Descripción de requerimientos a nivel de sistema	47
3.2.2.4	Descripción de requerimientos a nivel técnicos	48

3.3	Fase de Elaboración.....	48
3.3.1	Modelado de Análisis	48
3.3.1.1	Modelo de casos de uso	48
3.3.1.2	Diagrama de casos de uso de alto nivel.....	48
3.3.1.3	Descripción de casos de uso	49
a)	Caso de uso: Administrar Empleado	49
c)	Caso de uso: Formalizar Liquidación.....	52
d)	Caso de uso: Realizar control y seguimiento de personal	54
e)	Caso de uso: Elaborar Reportes.....	55
f)	Caso de uso: Realizar Conceptos de Planilla.....	57
3.3.1.4	Diagrama de Paquetes	58
3.3.2	Modelo de diseño	59
3.3.2.1	Diagrama de clases	59
3.3.2.2	Modelo de navegación.....	61
3.3.2.3	Modelo de navegación de menús.....	62
3.3.2.4	Modelo de presentación.....	62
3.4	Fase de construcción.....	64
3.4.1	Diseño de interfaces.....	64
3.4.1.1	Autenticación	64
3.4.1.3	Registro de empleado	66
3.4.1.4	Modificar/eliminar empleado	67
3.4.1.5	Registrar usuario.....	69
3.4.1.6	Liquidación.....	71
3.4.1.7	Seguimiento de personal.....	74
3.4.1.8	Seguimiento de contrato	76
3.5	Fase de transición	77
3.6	Políticas de implementación.....	77
3.6.1	Corte y cambio	78
3.6.2	Operación paralela.....	78
3.6.3	Operación piloto	79
3.6.4	Operación de fase	79
3.6.5	Aplicación del método corte y cambio.....	80
	CAPITULO IV.....	82
	CALIDAD DE SOFTWARE Y SEGURIDAD.....	82
4.1	Introducción.....	82
4.2	Características Propuestas por ISO-9126	82
4.2.1	Confiabilidad	83
4.2.2	Funcionalidad	85
4.2.2.1	Número de entradas de usuarios.....	86
4.2.2.2	Número de salidas de usuario.....	87
4.2.2.3	Número de peticiones de usuario.....	87

4.2.2.4	Número de archivos.....	88
4.2.2.5	Número de interfaces externas	89
4.2.2.6	Ponderación	89
4.2.3	Mantenibilidad.....	91
4.2.4	Portabilidad.....	92
4.2.4.1	Nivel de aplicaciones.....	93
4.2.4.2	Nivel de sistema operativo	93
4.2.4.3	Nivel de hardware.....	93
4.2.5	Usabilidad.....	93
4.2.5.1	La complejidad de la descripción	94
4.2.5.2	Consistencia operacional	94
4.2.5.3	Consistencia operacional en el uso.....	95
4.3	Seguridad de Software.....	95
4.3.1	Amenazas.....	95
4.3.2	Guías de seguridad	96
4.3.3	Tipos de seguridad para sistemas web.....	96
4.3.3.1	Seguridad en el cliente.....	96
4.3.3.2	Seguridad en el servidor	97
4.3.3.3	Seguridad en la comunicación.....	98
4.3.3.4	Seguridad en la aplicación.....	98
CAPITULO V		101
ANÁLISIS DE COSTO/BENEFICIO		101
5.1	Introducción.....	101
5.2	Análisis de costos	101
5.2.1	Costo del software desarrollado	101
5.2.2	Costo de la implementación del proyecto.....	105
5.2.3	Costo de elaboración del proyecto.....	105
5.2.4	Costo total.....	105
5.3	Análisis de beneficios.....	106
5.4	Valor neto actual	106
5.5	Tasa interna de retorno	108
CAPITULO VI.....		111
CONCLUSIONES Y RECOMENDACIONES.....		111
6.1	Conclusiones.....	111
6.2	Recomendaciones	112
BIBLIOGRAFIA		114
ANEXOS		118

INDICE DE FIGURAS

	Pág.
Figura 2.1: Ciclo de vida del modelo en cascada.....	13
Figura 2.2: Ciclo de vida del método iterativo.....	14
Figura 2.3: Fases y Disciplinas del AUP	17
Figura 2.4: Flujo de trabajo de la disciplina del modelado	20
Figura 2.5: Flujo de trabajo de la Disciplina de la implementación	23
Figura 2.6: Flujo del trabajo de la disciplina de pruebas	25
Figura 2.7: Flujo de trabajo de la disciplina de administración de la configuración	27
Figura 2.8: Flujo de trabajo de la disciplina de Administración de proyecto	29
Figura 2.9: Flujo de Trabajo de la disciplina de entorno	32
Figura 2.10: Caso de uso UWE.....	36
Figura 2.11: Modelo de contenido UWE	36
Figura 2.12: Modelo de navegación del UWE.....	37
Figura 2.13: Modelo de presentación.....	37
Figura 2.14: Modelo de Proceso d UWE	38
Figura 3.1: Fases y Disciplinas del AUP con UWE.....	44
Figura 3.2: Diagrama de caso de uso del negocio.....	45
Figura 3.3: Diagrama de caso de uso del Sistema de Control y Seguimiento de Empleados ..	49
Figura 3.4: Diagrama de caso de uso: Administración de empleado	50
Figura 3.5: Diagrama de caso de uso: Administración de usuario.....	51
Figura 3.6: Diagrama de caso de uso: Formalizar Liquidación	53
Figura 3.7: Diagrama de caso de uso: Control y seguimiento de personal	54
Figura 3.8: Diagrama de caso de uso: Reportes	56
Figura 3.9: Diagrama de caso de uso: Concepto de planilla	57
Figura 3.10: Diagrama de paquetes: Sistema Web de Control y Seguimiento de Personal.....	59
Figura 3.11: Diagrama de clases: Sistema Web de control y seguimiento de Personal.....	60
Figura 3.12: Modelo de Espacio de Navegación	61
Figura 3.13: Modelo de Espacio de Navegación	62
Figura 3.14: Modelo de presentación: Pagina Maestra	63
Figura 3.15: Modelo de Presentación: Pagina Director Administrativo	64
Figura 3.16: Autenticación del sistema.....	65
Figura 3.17: Pantalla principal	65
Figura 3.18: Pantalla registro de empleado.....	66
Figura 3.19: Pantalla mensaje de registro	67
Figura 3.20: Pantalla modificar empleado	68
Figura 3.21: Pantalla 2 modificar empleado	69
Figura 3.22: Registrar usuario	70
Figura 3.23: Registro de usuario 2	71
Figura 3.24: Liquidación	72
Figura 3.25: Liquidación 2	73

Figura 3.26: Confirmación de Liquidación 2	73
Figura 3.27: Seguimiento de personal	74
Figura 3.28: Seguimiento de personal 2	75
Figura 3.29: Resultado del Seguimiento de personal	76
Figura 3.30: Resultado del Seguimiento de personal	77
Figura 3.31: Corte y Cambio	78
Figura 3.32: Operación Paralela	79
Figura 3.33: Operación piloto	79
Figura 3.34: Operación de fase	80
Figura 4.1: Modelo Del Sistema Web De Control Y Seguimiento De Personal	83

INDICE DE TABLAS

	Pág.
Tabla 2.1: Disciplinas del AUP.....	18
Tabla 2.2: Fases del AUP	19
Tabla 2.3: Fases de la disciplina del modelado	23
Tabla 2.4: Fases de la disciplina de la implementación	24
Tabla 2.5: Fases de la disciplina de pruebas.....	27
Tabla 2.6: Fases de la disciplina de administración de la configuración	28
Tabla 2.7: Fases de la disciplina de administración de proyectos	32
Tabla 2.8: Fases de la disciplina de entorno.....	34
Tabla 3.1: Descripción de actores de casos de uso del negocio	46
Tabla 3.2: Descripción de caso de uso: Administración de personal	51
Tabla 3.3: Descripción de caso de uso: Administración de usuario.....	52
Tabla 3.4: Descripción de caso de uso: Formalizar Liquidación	54
Tabla 3.5: Descripción de caso de uso: Control y seguimiento de personal	55
Tabla 3.6: Descripción de caso de uso: Reportes	57
Tabla 3.7: Descripción de caso de uso: Concepto de planilla	58
Tabla 4.1: Calculo de confiabilidad.....	84
Tabla 4.2: Entradas de Usuario	86
Tabla 4.3: Salidas de Usuario.....	87
Tabla 4.4: Peticiones de Usuario	88
Tabla 4.5: Calculo de archivos	88
Tabla 4.6: Interfaces externas.....	89
Tabla 4.7: Factores de ponderación.....	89
Tabla 4.8: Valores de ajustes de complejidad	90
Tabla 5.1: Calculo de punto función no ajustado	102
Tabla 5.2: Calculo de punto función ajustada	103
Tabla 5.3: Conversión de puntos Función a KLDC	103
Tabla 5.4: Coeficientes a_b y b_b	104
Tabla 5.5: Costo de elaboración del proyecto	105
Tabla 5.6: Costo total del proyecto	106

CAPITULO I

MARCO INTRODUCTORIO

La ciencia no nos ha enseñado aún si la locura es o no lo más sublime de la inteligencia. (Edgar Allan Poe)

MARCO INTRODUCTORIO

1.1 Introducción

Los sistemas de información se han establecido como factores cruciales para la toma de decisiones así como la aceleración de procesos.

La evolución de las tecnologías asociadas a la información, hace que nuestra sociedad este cada día más conectada electrónicamente. Las labores que tradicionalmente eran realizadas manualmente, ahora son realizadas por medio de sistemas de información.

En toda institución es importante tener un control fiable del Recurso Humano ya que este constituye el eje fundamental que mueve a todos los sistemas, además es esencial para el logro y alcance de los propósitos o metas trazadas por la institución.

Al referirnos a los Recursos Humanos podemos encontrar muchos desafíos, estos se deben afrontar de manera correcta al realizar el control y seguimiento del personal tomando en cuenta diversos factores.

El control y seguimiento, nos ayudara en la revisión de actividades, controlar la entrada y salida del personal, cumplimiento de horarios de trabajo, controlar horas extras, permisos, vacaciones, retrasos, licencias, bajas médicas, liquidaciones, presupuesto, cargos y además una herramienta de gestión que permite tomar decisiones estratégicas analizando el entorno, observando el avance o cumplimiento de los objetivos propuestos.

La Dirección Administrativa es el área encargada de los Recursos Humanos de las diversas áreas que forman parte de la Universidad Tecnológica Boliviana (UTB), además es la responsable que las actividades referentes al personal marchen de acuerdo con lo previsto.

El presente trabajo propone el desarrollo de un Sistema Web encargado del Seguimiento y Control del Personal de la Universidad Tecnológica Boliviana, que facilite el manejo de la información oportuna en la Dirección Administrativa para sus planes futuros y poder detectar a tiempo posibles dificultades o problemas.

1.2 Antecedentes

1.2.1 Antecedentes Institucionales

La Universidad Tecnológica Boliviana, es una institución que por su carácter universal y abierto, genera amplias oportunidades de profesionalización en todos los niveles y modalidades de la educación superior. Promoviendo la formación integral, que promueve la creación, el desarrollo y la adaptación del conocimiento científico, tecnológico, cultural, deportivo y humanístico, comprometiéndose en la búsqueda de soluciones a los problemas de la sociedad en beneficio del desarrollo humano y científico. Responsabilidad social, es la capacidad de respuesta que tiene la universidad frente a los efectos e implicaciones de sus acciones sobre los grupos con los que se relaciona, orientando todas sus actividades a la satisfacción de las necesidades y expectativas de sus miembros, de la sociedad y de quienes se benefician de su actividad de servicios, así como el cuidado y preservación del medio ambiente. Pensamiento crítico, es la reflexión constructiva y evaluativa de pensamientos, conocimientos, procesos, hechos y fenómenos del entorno. Desarrollo del espíritu emprendedor, La Universidad desarrolla la capacidad de las personas para crear, generar y concretar una idea o solución, asumiendo riesgos convirtiéndola en realidad. Finalmente el sentido ético, la universidad debe ser un centro de consolidación de los más altos valores humanos como el respeto, la tolerancia, justicia, equidad y garantizando la formación total del individuo en sus dimensiones intelectual, ética y moral.

1.2.1.1 Misión

La Universidad Tecnológica Boliviana es pionera en el desarrollo de programas de transformación científica-tecnológica, con responsabilidad social y en permanente búsqueda de la excelencia. Formando profesionales, líderes, emprendedores e idóneos que sean competitivos en el contexto nacional e internacional, comprometidos con la innovación tecnológica, el progreso, el bienestar social, capaces de establecer alternativas de solución a los problemas ambientales, económicos, jurídicos, científicos y técnicos basado en un modelo educativo, productivo abierto y flexible que promueve la Investigación, la Interacción social y Difusión cultural.

1.2.1.2 Visión

La Universidad Tecnológica Boliviana, será una institución reconocida internacionalmente por su calidad e innovación académica, alta competitividad de sus graduados y los servicios tecnológicos que ofrece; mediante programas académicos acreditados, claustros docentes consolidados y tecnologías educativas de vanguardia que cumplan las expectativas y necesidades de la sociedad y los sectores productivos para contribuir a su desarrollo integral.

1.2.1.3 Dirección Administrativa

La Dirección Administrativa tiene el objetivo principalmente de integrar a las personas nuevas o promovidas en el nuevo contexto de la organización; esto involucra la aclimatación y el acondicionamiento de las personas mediante un programa de iniciación e integración organizacional. La responsabilidad de esta unidad involucra al personal contratado o promovido en todas las áreas de la organización.

También se encarga del reclutamiento, selección, inducción, orientación, evaluación de desempeño, formación y desarrollo de los recursos humanos, de acuerdo a la normativa vigente; proponer un programa de compensaciones y administración de salarios donde se tome en cuenta las necesidades de la organización y de los empleados.

El responsable de la unidad de Recursos Humanos es el responsable de la gestión del programa de iniciación e integración organizacional al recurso humano de la institución. Dicha gestión involucra: diseño, seguimiento, control y retroalimentación del programa.

1.2.2 Trabajos Similares

Los trabajos citados a continuación fueron realizados en la carrera de Informática de la Universidad Mayor de San Andrés:

- Sistema de Administración de personal unidad de talento humano Hospital Agramont.¹

Este sistema realiza el control de horarios de entrada y salida de los empleados, la actualización de sus hojas de vida, el registro de contratos mixtos ya sea básico o porcentaje

¹ Nina, G. 2009

y el movimiento que tiene en la institución como ser altas, bajas, rotación, permiso y vacación. El desarrollo del sistema de administración de personal optimiza el tiempo en los procesos también se utilizará una base de datos centralizada en un servidor al cual pueden tener acceso los usuarios dados de alta desde cualquier punto de la institución.

Para el desarrollo de dicho proyecto se utilizó la metodología RUP, (proceso Unificado de Rational), metodología estándar para la construcción completa del ciclo de ingeniería de software, esta metodología es apoyado con UML (Lenguaje Unificado de Modelado).

- Sistema de control y seguimiento a programas operativos anuales (POA's) para el vice ministerio de biodiversidad, recursos forestales y medio ambiente.²

Este sistema se orienta a procesos realizados dentro del departamento de Monitoreo y Evaluación de Programas para el seguimiento, control y optimización de los diferentes procesos y proyectos en el Viceministerio de Biodiversidad, Forestal y Medio Ambiente.

El sistema tiene los siguientes procesos: Registro y ordenación de los planes de acción y componentes, registro y ordenación de las líneas de acción prioritarias con sus respectivas metas, generación de reportes de los diferentes estados, generación de reportes de datos históricos. Para el desarrollo del sistema se utiliza la metodología XP y UML como herramienta de modelado de diseño.

- Sistema de monitorio y control para el programa Tomando decisiones- Save the children.³

Este proyecto tiene como puerta de entrada a un conjunto de recursos existentes en Save de Children, con el objetivo de brindar un servicio de difusión de información que permita crear un nuevo canal de aprendizaje como ser proyectos sostenibles con participación de la comunidad también el control de un conjunto de principios y normas que garantían la adecuación entre datos, procesos e información, a fin de que los informes obtenidos reúnan las características demandadas por sus usuarios.

² Huanca, R. 2008

³ Gonzales, L. 2006

Este proyecto tiene la definición de un marco de trabajo basado en métodos y procesos recomendados (Craig, Larman), método de ingeniería de software orientado a objetos, herramienta UML (Lenguaje de Modelado Unificado).

- Sistema de control de personal biométrico Caso: Gobierno municipal de La Paz.⁴

El sistema de control de personal soluciona los problemas de integración de datos y de procesos, sin dejar de lado a los que inician los procesos y a los que finalizan, es decir que la información nazca en el sistema y administrativamente fenezca en el mismo de acuerdo a etapas de validación de datos sin dejar de lado las normas, reglamentos o leyes externas e internas vigentes. El desarrollo de dicho sistema permite al personal administrativo realizar el control correcto de todo personal, desde la ubicación física real hasta saber quién es la persona que se encarga de desempeñar las labores de control de personal. El sistema se encarga de las labores de control de personal, registro de licencias, horarios, feriados, ubicación física, registro de controles de personal, registro y emisión de tenores de memorándums, asistencias, registro de cursos de capacitación.

En el presente proyecto hace uso de la metodología RUP (Rational Unified Process), que se complementa con la metodología OOHDM (método de diseño de hipermedia orientada a objetos) también se utiliza la Herramienta UML (Lenguaje de Modelado Unificado).

1.3 Planteamiento de Problema

1.3.1 Problema Principal

¿De qué manera se podrá optimizar el Control y el Seguimiento de Personal para la Universidad Tecnológica Boliviana?

1.3.2 Problemas Específicos

- No se tiene el control adecuado de permisos, faltas, vacaciones, sanciones lo que ocasiona pérdida de tiempo y control de RRHH.

⁴ Silvestre, N. 2009

- Las liquidaciones se realizan en forma manual, lo que produce pérdida de tiempo y costos extras en algunos casos.
- No se hace el control de múltiples cargos de un empleado, lo que ocasiona conflictos a la hora de pagar sueldos.
- El incremento de nuevos sueldos y jornales no es automático, lo que ocasiona tiempo de espera para el pago de sueldos.
- No se hace el control de contrato de los empleados, lo que ocasiona pérdida de tiempo al realizar la liquidación.

1.4 Objetivos

1.4.1 Objetivo Principal

“Desarrollar un Sistema Web de Control y Seguimiento de Personal para la Universidad Tecnológica Boliviana”.

1.4.2 Objetivos Específicos

- Desarrollar el módulo de control y seguimiento de los procesos que realiza un empleado.
- Automatizar el proceso de liquidación tomando en cuenta todos los parámetros necesarios para dicho proceso.
- Realizar el módulo de administración de empleado para guardar sus datos personales y laborales.
- Realizar el módulo de administración de usuarios para que el usuario con rol de administrador realice modificaciones en los salarios de los empleados.
- Desarrollar los módulos control y seguimiento de personal y de contratos.
- Registrar los datos laborales del empleado desde el inicio del trabajo hasta su conclusión.
- Integrar el sistema biométrico con el Sistema Web de Control y Seguimiento de personal.

1.5 Justificación

1.5.1 Económica

Con la implementación de este sistema se evitará el gasto de dinero en impresiones, fotocopias de solicitudes y documentos. También se puede evitar gastos en llamadas telefónicas. Además que el sistema permitirá un control de la administración del personal con el objetivo de no realizar pagos en demasía, de esta manera evitar gastos y fraude a la UTB.

1.5.2 Social

El sistema proveerá de información a la Dirección Administrativa que comprende todo el personal con el que cuenta la UTB, además el sistema se constituirá en una herramienta útil en la administración de personal, se llevará un correcto control y seguimiento de personal, generando un ambiente de trabajo donde los empleadores, empleados, y el mismo control de personal se desenvuelvan de manera satisfactoria.

1.5.3 Técnica

El sistema será desarrollado con el Proceso Unificado Ágil o Agile Unified Process (AUP), es una versión simplificada del Procesos Unificado Racional (RUP) y es relativamente nueva. El AUP describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP.

El AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas, Modelado Ágil, Gestión de Cambios Ágil, y Refactorización de Base de Datos para mejorar la productividad.

La herramienta a utilizar para el modelado del diseño es UWE, propuesta de Ingeniería Web basada en UML (UWE (Koch,2000)) es una metodología detallada para el proceso de autoría de aplicaciones con una definición exhaustiva del proceso de diseño que debe ser utilizado. Este proceso, iterativo e incremental, incluye flujos de trabajo y puntos de control, y sus fases coinciden con las propuestas en el Proceso Unificado de Modelado.

El sistema se desarrollará bajo los estándares de la UTB teniendo como plataforma Visual Studio.Net 2008 y con el gestor de base de datos SQL Server 2005.

1.6 Límites y alcances

1.6.1 Alcances

El sistema tendrá los siguientes módulos:

- Administración de personal de la UTB.
- Administración de usuarios del sistema web.
- Formalización de liquidación para los empleados de la UTB.
- Control y seguimiento de personal de los empleados de la UTB.
- Reportes para la generación de planilla de empleados, boletas de pago y otros.
- Conceptos de planilla de los empleados de la UTB.

1.6.2 Límites

El sistema tiene los siguientes límites.

- El sistema no registrará a los estudiantes de la UTB.
- El sistema no controla el pago directo por entidades financieras.
- Los empleados habilitados por el director administrativo tendrán acceso al sistema, dependiendo el rol asignado.
- El sistema no controla la parte contable de toda la institución.
- El sistema no controla la asistencia de los estudiantes a la UTB.

1.7 Aportes

El sistema web de control y seguimiento de personal será un aporte para la Universidad Tecnológica Boliviana ya que el control de RRHH se debe realizar en toda institución.

Para el desarrollo del sistema se utilizará la metodología AUP, Proceso Unificado Ágil, es un enfoque al desarrollo de software basado en el Rational Unified Process (RUP) de IBM.

Para el modelado se utilizará UWE, es una herramienta para modelar aplicaciones web. UWE es una propuesta basada en el proceso unificado y UML pero adaptados a la web. En requisitos separa las fases de captura, definición y validación. También realiza una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.

1.8 Metodología

La metodología es la guía que nos va indicando qué hacer y cómo actuar cuando se quiere obtener una investigación. Es posible decir que la metodología es aquel enfoque que permite observar un problema de una forma total, sistemática, disciplinada y con cierta disciplina.

En cuanto a la obtención y desarrollo del sistema de información, se basará en una de las metodologías ágiles Proceso Unificado Ágil denotado por el acrónimo AUP que se adecua mejor en la consecución del producto final. En cuanto a la herramienta de diseño se utilizará la propuesta de Ingeniería Web basada en UML UWE es una herramienta detallada para el proceso de autoría de aplicaciones con una definición exhaustiva del proceso de diseño que debe ser utilizado

Método de investigación que realizaremos en este trabajo en el método científico, que es un proceso destinado a explicar fenómenos, establecer relaciones entre los hechos y enunciar leyes que expliquen los fenómenos físicos del mundo y permitan obtener, con estos conocimientos, aplicaciones útiles al hombre.

El tipo de investigación en principio será del tipo exploratorio, ya que se efectúa sobre un tema u objeto desconocido o poco estudiado, por lo que sus resultados constituyen una visión aproximada de dicho objeto, es decir, un nivel superficial de conocimiento. Después será una investigación descriptiva, ya que se conocerá las situaciones y actitudes predominantes a través de la descripción exacta de las actividades, objetos, procesos y personas. Se recogen los datos sobre la base de una hipótesis o teoría, luego se analiza minuciosamente los resultados, a fin de extraer generalizaciones significativas que contribuyan al conocimiento.



*El sabio no dice todo lo que piensa,
pero siempre piensa todo lo que dice.
(Aristóteles)*



MARCO TEÓRICO

En este capítulo daremos a conocer fundamentos teóricos para la realización del presente proyecto. Definiremos conceptos, estableceremos metodología, técnicas y herramientas adecuadas a usar en el desarrollo del Sistema.

2.1 Ingeniería de Software

Según la definición del Institute of Electrical and Electronics Engineers (IEEE), software es la suma total de los programas de computadora, procedimientos, reglas, la documentación asociada y los datos que pertenecen a un sistema de cómputo". Según el mismo autor, "un producto de software es un producto diseñado para un usuario". En este contexto, la Ingeniería de Software es un enfoque sistemático del desarrollo, operación, mantenimiento y retiro del software", que en palabras más llanas, se considera que "la Ingeniería de Software es la rama de la ingeniería que aplica los principios de la ciencia de la computación y las matemáticas para lograr soluciones costo-efectivas (eficaces en costo o económicas) a los problemas de desarrollo de software", es decir, "permite elaborar consistentemente productos correctos, utilizables y costo-efectivos".

El proceso de ingeniería de software se define como: un conjunto de etapas parcialmente ordenadas con la intención de logra un objetivo, en este caso, la obtención de un producto de software de calidad.

El proceso de desarrollo de software: es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo. Concretamente define quién está haciendo qué, cuándo hacerlo y cómo alcanzar un cierto objetivo.

El proceso de desarrollo de software requiere por un lado un conjunto de conceptos, una metodología y un lenguaje propio. A este proceso también se le llama el ciclo de vida del software que comprende cuatro grandes fases: concepción, elaboración, construcción y

transición. La concepción define el alcance del proyecto y desarrolla un caso de negocio. La elaboración define un plan del proyecto, especifica las características y fundamenta la arquitectura. La construcción crea el producto y la transición transfiere el producto a los usuarios.

2.2 Sistema

Es el conjunto de partes interrelacionadas, hardware, software y de recurso humano (humanware) que permite almacenar y procesar información. El hardware incluye computadoras o cualquier tipo de dispositivo electrónico inteligente, que consisten en procesadores, memoria, sistemas de almacenamiento externo, etc. El software incluye al sistema operativo, firmware y aplicaciones, siendo especialmente importante los sistemas de gestión de bases de datos. Por último el soporte humano incluye al personal técnico que crean y mantienen el sistema (analistas, programadores, operarios, etc.) y a los usuarios que lo utilizan.

2.3 Metodología de desarrollo

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Es como un libro de recetas de cocina, en el que se van indicando paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además qué personas deben participar en el desarrollo de las actividades y qué papel deben de tener. Además detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

En las metodologías de desarrollo de software se tiene los métodos: iterativos, evolutivos y ágiles.

2.3.1 Modelo en cascada

La versión original del modelo en cascada, fue presentada por Royce en 1970, aunque son más conocidos los refinamientos realizados por Boehm [1981], Sommerville [1985] y Sigwart y Col. [1990]. En este modelo, el producto evoluciona a través de una secuencia de fases ordenadas en forma lineal y permitiendo iteraciones al estado anterior.

El número de etapas suele variar, pero en general suelen ser:

- Análisis de requisitos del sistema
- Análisis de requisitos del software
- Diseño preliminar
- Diseño detallado
- Codificación y pruebas
- Explotación (u operación) y mantenimiento

Las características de este modelo son:

- Cada fase empieza cuando se ha terminado la anterior.
- Para pasar a la fase posterior es necesario haber logrado los objetivos de la previa.
- Es útil como control de fechas de entregas.
- Al final de cada fase el personal técnico y los usuarios tienen la oportunidad de revisar el progreso del proyecto.



Figura 2.1: Ciclo de vida del modelo en cascada
[Fuente: Lehman, M.M.]

2.3.2 Método iterativo

Las etapas son las mismas que en el ciclo de vida en cascada y su realización sigue el mismo orden, pero corrige la problemática de la linealidad del modelo en cascada. Este modelo incremental fue desarrollado por Lehman [1984].

En cada paso sucesivo se agregan al sistema nuevas funcionalidades o requisitos que permiten el refinado a partir de una versión previa.

Este modelo es útil cuando la definición de los requisitos es ambigua e imprecisa, porque permite el refinamiento, o sea se pueden ampliar los requisitos y las especificaciones derivadas de la etapa anterior.

Uno de los problemas que se puede presentar es la detección de requisitos tardíamente, siendo su corrección tan costosa como en el caso de la cascada. Un gráfico que muestra cual es el desarrollo de este proceso es el siguiente.

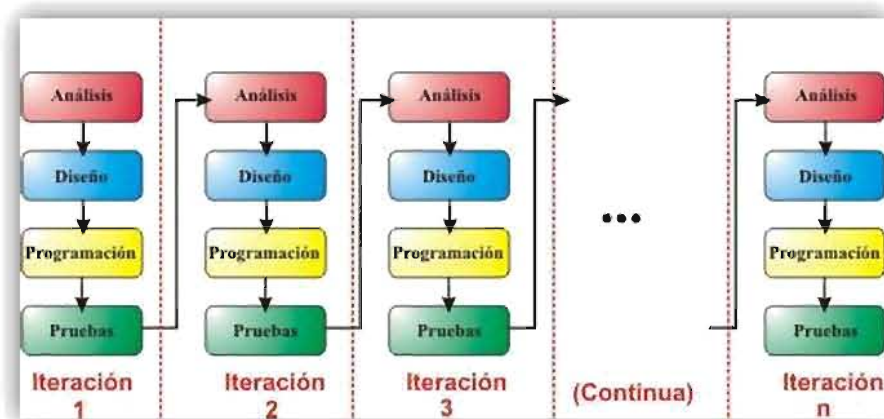


Figura 2.2: Ciclo de vida del método iterativo
[Fuente: Lehman, M.M.]

2.3.3 Método evolutivo

El uso de prototipos evolutivos se centra en la idea de ayudar a comprender los requisitos que plantea el usuario, sobre todo si este no tiene una idea muy acabada de lo que desea. También pueden utilizarse cuando el ingeniero de software tiene dudas acerca de la viabilidad de la solución pensada.

Esta versión temprana de lo que será el producto, con una funcionalidad reducida, en principio, podrá incrementarse paulatinamente a través de refinamientos sucesivos de las especificaciones del sistema, evolucionando hasta llegar al sistema final.

Al usar prototipos, las etapas del ciclo de vida clásico quedan modificadas de la siguiente manera:

- Análisis de requisitos del sistema
- Análisis de requisitos del software
- Diseño, desarrollo e implementación del prototipo
- Prueba del prototipo.
- Refinamiento iterativo del prototipo
- Refinamiento de las especificaciones del prototipo
- Diseño e implementación del sistema final
- Explotación (u operación) y mantenimiento

Si bien el modelo de prototipos evolutivos, fácilmente modificables y ampliables es muy usado, en muchos casos pueden usarse prototipos descartables para esclarecer aquellos aspectos del sistema que no se comprenden bien.

2.3.4 Método ágil

El desarrollo ágil de Software a un paradigma de Desarrollo de Software basado en procesos ágiles.

Los procesos ágiles de desarrollo de software, conocidos anteriormente como metodologías livianas, intentan evitar los tortuosos y burocráticos caminos de las metodologías tradicionales enfocándose en la gente y los resultados.

Según el manifiesto del método ágil se valora:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.
- Desarrollar software que funciona más que conseguir una buena documentación. La regla a seguir es .no producir documentos a menos que sean necesarios de forma inmediata para tomar un decisión importante.. Estos documentos deben ser cortos y centrarse en lo fundamental.
- La colaboración con el cliente más que la negociación de un contrato. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a los largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

2.3.5 Metodología ágil AUP

El Proceso Unificado Ágil (Agile UP) es un enfoque al desarrollo de software basado en el Rational Unified Process (RUP) de IBM. El ciclo de vida de Agile UP es serial en lo grande e iterativo en lo pequeño, liberando entregables incrementales en el tiempo.

2.3.5.1 Estructura del proceso unificado Ágil (AUP)

El Proceso Unificado Ágil está compuesto por cuatro fases, cada una de ellas se desarrolla mediante iteraciones, las cuales consisten en reproducir el ciclo de vida en cascada a menor escala.

El Eje Horizontal representa la parte dinámica del proceso en el tiempo, las iteraciones y las metas, el eje vertical representa la parte estática del proceso donde se describen los flujos de trabajo, requisitos, análisis, diseño, implementación y prueba. Las curvas son aproximaciones de hasta donde se llevan a cabo los flujos de trabajo en cada fase.

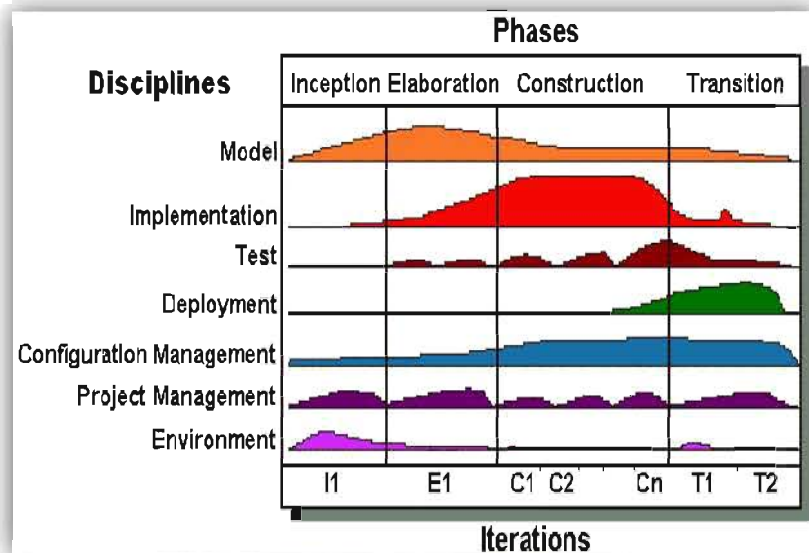


Figura 2.3: Fases y Disciplinas del AUP
[Fuente: Ambler, S.]

2.3.5.2 Disciplinas de AUP

Las disciplinas son ejecutadas de una forma iterativa, definiendo las actividades, las cuales, el equipo de desarrollo ejecuta para construir, validar y liberar software funcional, el cual cumple con las necesidades de los involucrados. En la siguiente tabla se puede ver las disciplinas del AUP.

Tabla 2.1

Disciplina	Descripción
Modelado	La meta de ésta disciplina es entender el negocio de la organización, el dominio del problema que el proyecto aborda e identificar una solución viable para abordar el dominio del problema.
Implementación	La meta de ésta disciplina es transformar su

	<p>modelo(s) en un código ejecutable y realizar una prueba de nivel básico en una unidad particular de prueba.</p>
<p>Pruebas</p>	<p>La meta de ésta disciplina es ejecutar una evaluación de los objetivos para asegurar la calidad. Esto incluye encontrar defectos, validar que el sistema función como fue diseñado y verificar que los requerimientos están completos.</p>
<p>Despliegue</p>	<p>La meta de ésta disciplina es planificar la entrega del sistema y ejecutar el plan para que el sistema esté disponible para los usuarios finales.</p>
<p>Administración de la Configuración</p>	<p>La meta de ésta disciplina es administrar el acceso a los entregables o productos del proyecto. Esto incluye no sólo el rastreo de versiones del producto en el tiempo, sino que también incluye controlar y administra los cambios que ocurran.</p>
<p>Administración de la Configuración</p>	<p>La meta de ésta disciplina es dirigir las actividades que se llevan a cabo en el proyecto. Esto incluye administración del riesgo, la dirección de personas (asignar tareas, seguimiento de los procesos, etc.), y coordinar con los sistemas y personas fuera del alcance del proyecto para que el este termine a tiempo y dentro del presupuesto.</p>
<p>Entorno</p>	<p>La meta de ésta disciplina es dirigir las actividades que se llevan a cabo en el proyecto. Esto incluye administración del riesgo, la dirección de personas (asignar tareas, seguimiento de los procesos, etc.), y coordinar con los sistemas y personas fuera del alcance del proyecto para que el este termine a tiempo y dentro del presupuesto.</p>

Disciplinas del AUP
[Fuente: Ambler, S. 2005]

2.3.5.3 Fases del AUP

Ágil UP está caracterizado por ser “serial en lo grande”, AUP contempla cuatro fases: inicio, elaboración, construcción y transición. Estas fases se muestran en la siguiente tabla y además podremos observar las metas e hitos de cada una de ellas, las fases del AUP se pueden mover en una forma serial.

Tabla 2.2

Fases	Metas	Hitos
1.Inicio	Identificar el alcance inicial de proyecto, una arquitectura inicial del sistema y obtener un presupuesto inicial del proyecto y una aceptación de los involucrados.	Objetivos de Ciclo de Vida(LCO, por su siglas en inglés)
2.Elaboración	Probar arquitectura del sistema.	Arquitectura del Ciclo de Vida (LCA, por su siglas en inglés)
3.Construcción	Construir un software funcional sobre una base regular e incremental, las cuales cumplan con las prioridades más importantes para los involucrados o usuarios del proyecto.	Capacidad Operacional Inicial (IOC, por su siglas en inglés)
4.Transición	Validar y desplegar el sistema en su ambiente de la producción.	Liberación del Producto (PR, por su siglas en inglés)

Fases del AUP
[Fuente: Ambler, S.]

La metodología ágil AUP está caracterizada por ser serial en lo grande, algo que se ve a través de estas cuatro fases.

2.3.5.4 Disciplina de Modelado

El objetivo de esta disciplina es entender el negocio de la organización, el problema de dominio que se abordan en el proyecto, y determinar una solución viable para resolver el problema de dominio.

a) Flujo de Trabajo

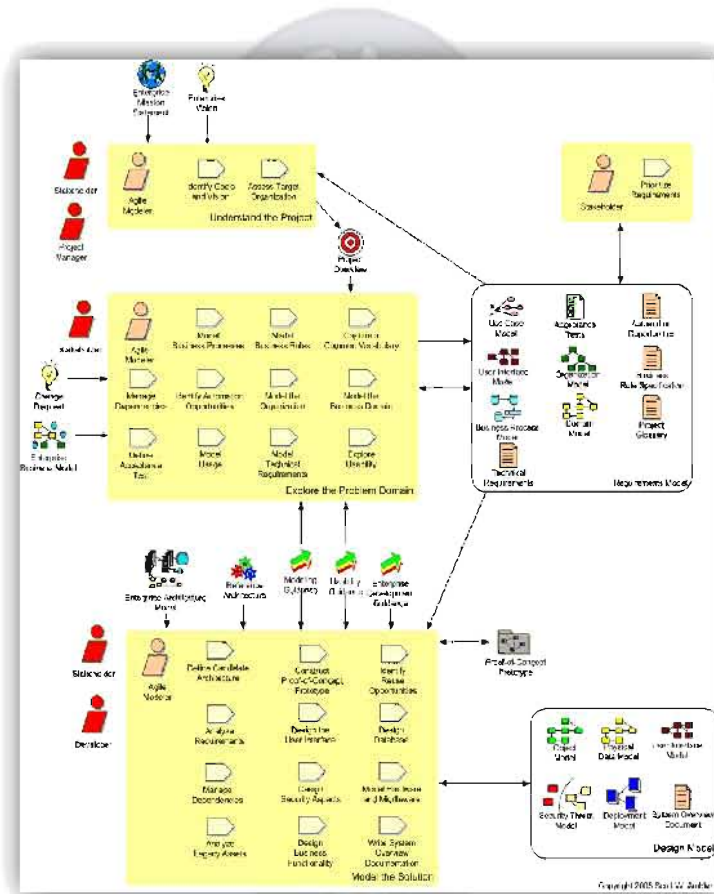


Figura 2.4: Flujo de trabajo de la disciplina del modelado [Fuente: Ambler, S.]

b) Fase Por Fase

Se observa las fases de la disciplina del modelado en la Tabla siguiente tabla.

Tabla 2.3

Fases	Actividades
Inicio	Modelado de requerimientos de alto nivel. Los interesados deben participar activamente en el modelo de requerimientos de alto nivel

	<p>el cual define el alcance inicial para el proyecto y proporciona suficiente información para una estimación aproximada. Debería considerar:</p> <ul style="list-style-type: none"> ✓ Explorar el uso de Casos de Uso. ✓ Identifique los Procesos de negocio, para la creación de diagramas de flujo de datos. ✓ Identifique las entidades principales del negocio y sus relaciones trabajando con modelos de dominio livianos. ✓ Identifique las principales reglas del negocio y requerimientos técnicos. Por ahora, el nombre de entidades, reglas y requerimientos técnicos del negocio que son suficientes. ✓ Inicie el desarrollo de un glosario que describa términos importantes técnicos y del negocio. ✓ La comprensión de la estructura política dentro de su comunidad de partes interesadas a través del modelado de la organización.
<p>Elaboración</p>	<p>Identificar riesgos técnicos, sus necesidades de trabajo, en particular, sus casos de uso y requerimientos técnicos, ponen de manifiesto los posibles riesgos técnicos a su proyecto. Estos riesgos pueden incluir la introducción de nuevas tecnologías a su organización, un nuevo uso de las tecnologías existentes, importante carga o estrés en su aplicación o sistemas actuales externos. La máxima prioridad debe ser abordar los riesgos por su esfuerzo de implementación en el desarrollo de un extremo a extremo del esqueleto del sistema.</p> <p>Modelado de la Arquitectura. Como usted construye el prototipo de la arquitectura necesitará modelar por lluvia de ideas algunos detalles para pensar pedazos de la arquitectura.</p>
<p>Construcción</p>	<p>Durante las iteraciones de la Construcción deberá trabajar cerca de los interesados del proyecto para entender sus necesidades. Aspectos importantes:</p> <ul style="list-style-type: none"> ✓ Participación activa de interesados y modelado inclusivo que usan técnicas y herramientas simples que son críticas para su negocio. ✓ Si lo desea, puede profundizar en los detalles de sus casos de uso, quizás visualmente utilizando diagramas de flujo o

	<p>diagramas de actividad UML en vez de descripciones de texto.</p> <ul style="list-style-type: none"> ✓ Explore las reglas del negocio y los requerimientos técnicos en la misma forma. ✓ Puede que necesite hacer interfaces de activos legados tales como sistemas actuales o una base de datos. El Análisis actual puede ser difícil y trabajo "políticamente cargado." ✓ En lugar de las descripciones de casos de uso, de reglas del negocio y de requerimientos técnicos, usted puede encontrar más efectivo simplemente escribir casos de prueba de aceptación. Esto le permite acercarse a una única fuente de información porque no es necesario para capturar el requisito tanto en un documento de requerimientos como en una descripción de prueba. <p>Diseño por modelo de lluvia de ideas. Durante las iteraciones de la Construcción su objetivo es hacer sólo lo suficiente para modelar para pensar en el diseño de un simple requerimiento, o sólo una parte, antes de la implementación del requerimiento. Modeladores ágiles modelan directo con sus usuarios, no simplemente pasar los modelos a los mismos, y, a menudo, asumen el papel de promotor. Es probable que desee crear:</p> <ul style="list-style-type: none"> ✓ Diagrama de secuencia de UML. Este diagrama representa la lógica dinámica dentro del código fuente. Son parte de su modelo de proyecto y usualmente se tiran lejos hasta que se tiene una buena herramienta CASE con compatibilidad para ingeniería inversa. Las pizarras son geniales herramientas para crear nuevos diagramas. ✓ Modelo de despliegue. Típicamente crear algún ordenamiento del diagrama de resumen representado en la arquitectura del sistema de despliegue/ red. ✓ Diagrama de Clases UML. Si va a hacer algún diagrama de clases use una herramienta de modelado que le permita generar código fuente. Su diagrama de clases debe estar basada en un modelo de dominio (si existe) ✓ Modelo físico de datos. Este es probablemente el modelo de diseño más importante, que debería considerar el uso de una herramienta CASE para desarrollar y mantener en el tiempo,
--	---

	sobre todo una herramienta que genera DDL código.
Transición	<p>Necesitará hacer algún modelado en el momento para tratar de entender las causas principales de un defecto.</p> <p>Finalice la documentación de resumen del sistema. El mejor momento para finalizar su documentación general del sistema es durante de la fase cuando el alcance de su sistema está realmente establecida. Realice su diseño crítico de decisiones, si lo documenta en la Construcción, como una base desde la cual se construye este documento. Cualquier otra información importante que quiera en este documento es un resumen del alcance del sistema y de los diagramas de arquitectura críticos (ahora es cuando se debería de colocar todos los diagramas de estilo libre y bocetos de pizarras en una buen diagrama usando una herramienta de dibujo).</p>

Fases de la disciplina del modelado
[Fuente: Ambler, S.]

2.3.5.5 Disciplina de la implementación

El objetivo de esta disciplina es transformar el modelo en código ejecutable y llevar a cabo un nivel básico de las pruebas, en particular, la unidad de prueba.

a) Flujo de trabajo

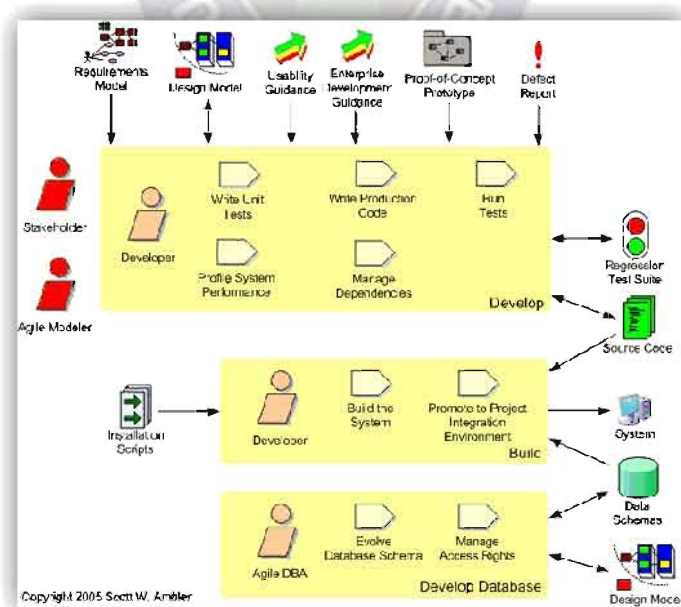


Figura 2.5: Flujo de trabajo de la Disciplina de la implementación
[Fuente: Ambler, S.]

b) Fase por fase

Se observa las fases de la disciplina de la implementación en la siguiente tabla.

Tabla 2.4

Fases	Actividades
Inicio	Prototipo técnico. Es posible que tenga que "picar" un pequeño aspecto de un requisito con el fin de entender lo suficiente, lo que le permite estimar el esfuerzo requerido. Estos prototipos son típicamente pequeños. "tire" las piezas de código.
Elaboración	Probar la arquitectura. Las actividades críticas dentro de la fase de Elaboración es identificar la arquitectura potencial y luego probar que esta arquitectura funcione a través del desarrollo de la arquitectura del prototipo extremo a extremo para su sistema, y a la vez mitigando gran parte de los riesgos técnicos en su proyecto. Los prototipos técnicos como son la calidad de producción de código que constituye el fundamento de su sistema.
Construcción	Primeras pruebas. Obtenga un acercamiento de la base del Desarrollo Dirigido por Pruebas para todos los aspectos de la aplicación. Construya constantemente. Creaciones diarias son un buen comienzo, pero idealmente usted quiera construir su sistema cada vez que el código fuente cambie. Evolución de la lógica de dominio. Implemente su lógica del negocio in sus clases de negocio/dominio. Evolucionar las interfaces de usuario. La interface de usuario es el sistema para la mayoría de usuarios. Esfuércese por hacer su software tan usable como sea posible siguiendo las estrategias de diseño de interfaces de usuario y usabilidad.
Transición	Corregir defectos. Concéntrese en la corrección de defectos encontrados como resultado de las pruebas.

Fases de la disciplina de la implementación

[Fuente: Ambler, S.]

2.3.5.6 Disciplina de Pruebas

El objetivo de esta disciplina es ejecutar una objetiva evaluación para asegurar la calidad. Esto incluye la detección de defectos, validaciones de que el sistema funciona como fue diseñado, y verificar que se cumplan los requerimientos.

a) Flujo de Trabajo

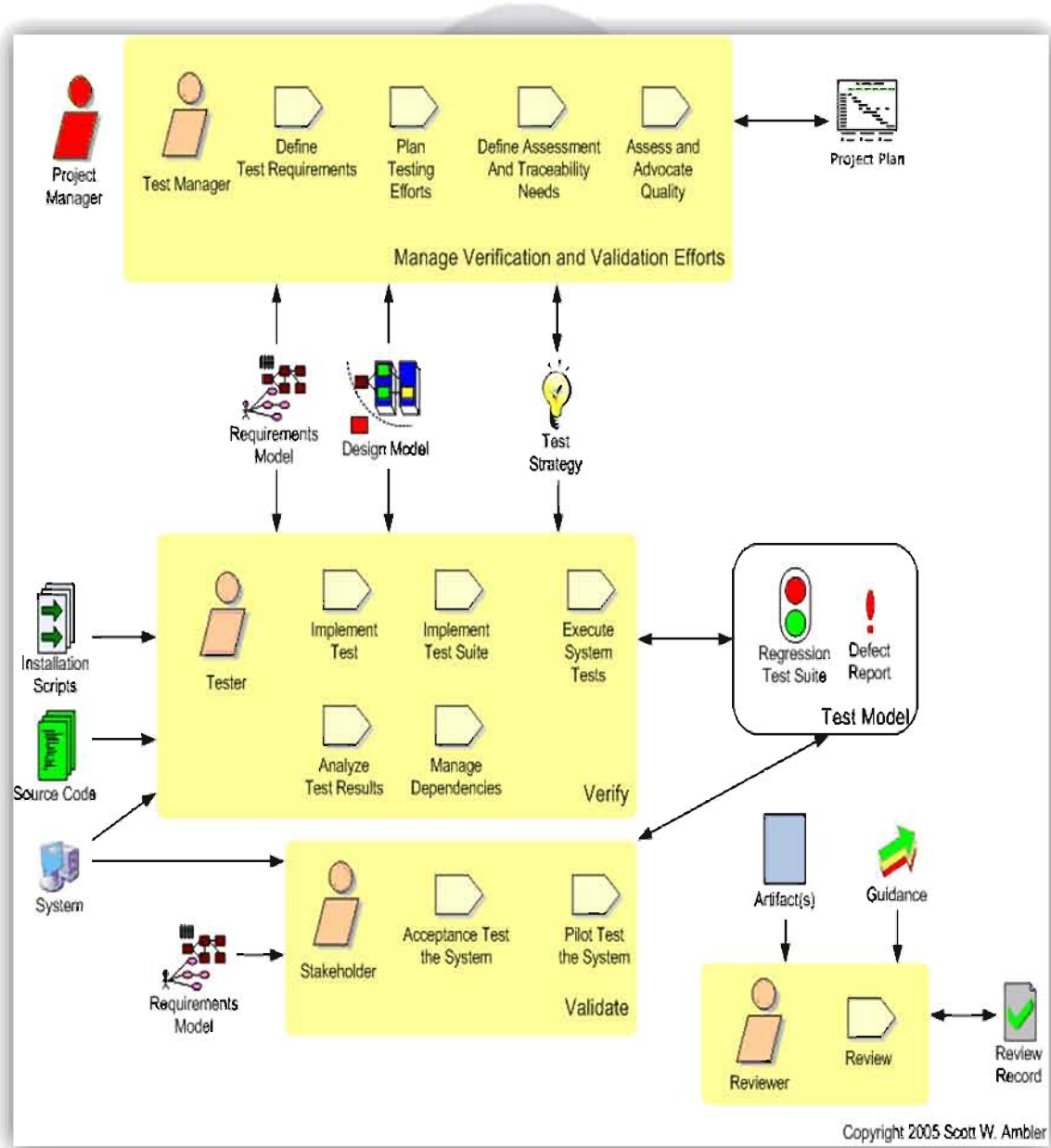


Figura 2.6: Flujo del trabajo de la disciplina de pruebas
[Fuente: Ambler, S.]

b) Fase Por Fase

Se observa las fases de la disciplina de pruebas en la siguiente tabla.

Tabla 2.5

Fases	Actividades
Inicio	Planificación inicial de pruebas. Deben ser a muy alto nivel al principio. El objetivo principal es identificar cuántas pruebas necesita hacer, quien será el responsable de hacerlas, el nivel de participación requerido por los usuarios, y los tipos de herramientas y los entornos necesarios. Exanimación inicial de los productos de trabajo del proyecto. Hacia el final de esta fase el plan de proyecto, la visión y mucho más deben estar disponibles. Este producto es a menudo examinado, típicamente como parte del hito de revisión, por los involucrados del proyecto.
Elaboración	Validación de la Arquitectura. Usted debe tomar un enfoque de desarrollo controlado por pruebas para construir su prototipo técnico el cual compruebe la arquitectura de su sistema. Un aspecto importante de hito de revisión es la validación de la arquitectura, que podría ser algo tan sencillo como presentar una visión general de la arquitectura y los resultados de sus esfuerzos de los prototipos para los interesados.
Construcción	Pruebas de software. Además de las unidades de prueba de los desarrolladores deberá hacer pruebas de instalación del script de despliegue o liberación, sistema de pruebas de esfuerzos tales como la carga de pruebas de tensión y las pruebas de función, y sus pruebas de aceptación de usuario.
Transición	Validación del sistema. Usted se concentrará en las "grandes pruebas" de actividades tales como las del sistema, Validación de la documentación. Su Documentación del sistema y los materiales de capacitación necesitarán ser validados. Todo esto puede ser hecho por medio de las revisiones. Analice su modelo de pruebas. Va a tener que seguir ejecutando el paquete de pruebas de regresión y actualizarlo

	tanto como
--	------------

Fases de la disciplina de pruebas
[Fuente: Ambler, S.]

2.3.5.7 Disciplina de administración de la configuración

La meta de esta disciplina es manejar el acceso a sus productos de trabajo de proyecto. Esta no sólo incluye el rastreo de versiones del trabajo del producto en el tiempo, sino que también el control y administración de los cambio estos productos.

a) Flujo de trabajo

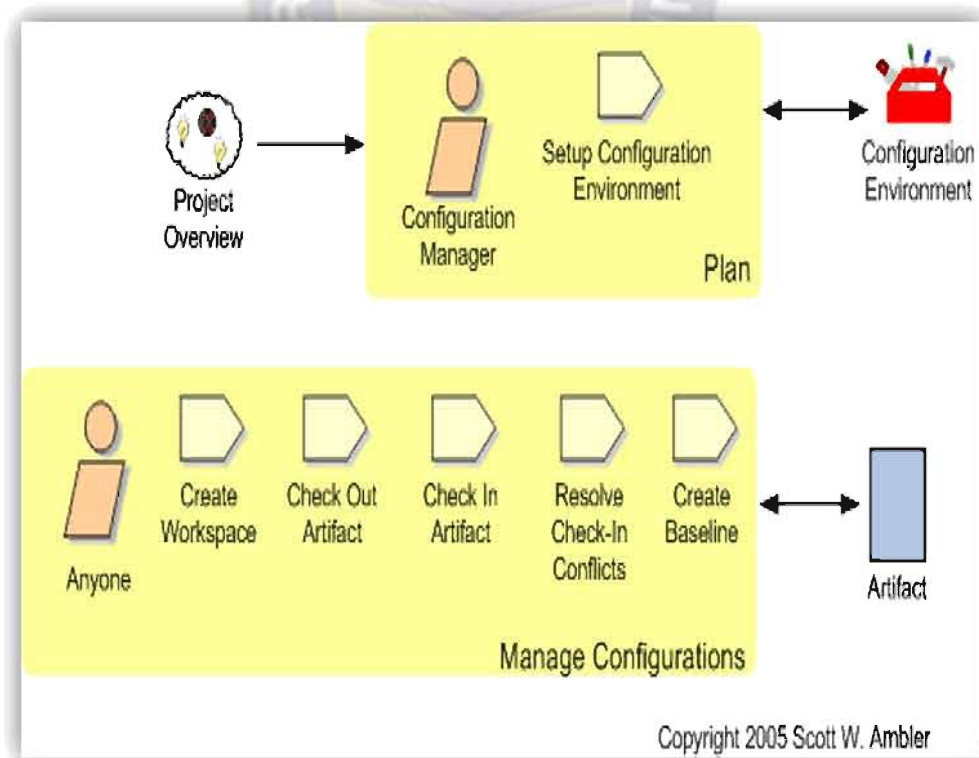


Figura 2.7: Flujo de trabajo de la disciplina de administración de la configuración
[Fuente: Ambler, S.]

b) Fase por fase

Se observa las fases de la disciplina de administración de la configuración en la siguiente tabla.

Tabla 2.6

Fases	Actividades
Inicio	<p>Usted tiene que hacer varias cosas:</p> <ul style="list-style-type: none">✓ La estructura de directorios apropiada, la cual debe seguir los lineamientos corporativos, necesita ser creada para el equipo del proyecto.✓ Los miembros del equipo del proyecto necesitan tener acceso al folder o directorios del proyecto. <p>Cualquier otro software cliente instalados en sus ordenadores</p> <ul style="list-style-type: none">✓ Los miembros del equipo del proyecto también necesitan ser entrenados con los conceptos básicos de CM. <p>Así como las herramientas necesarios.</p> <ul style="list-style-type: none">✓ Su repositorio de CM necesitará ser instalado si este aún no se ha instalado. <p>Ponga todos los productos del trabajo bajo el control de CM.</p> <p>Cada uno debe poner su trabajo bajo el control de CM en una base regular, verificar las entradas y salidas en cada caso.</p>
Elaboración	Poner todos productos del proyecto sobre el control de CM. Ver Arriba.
Construcción	Poner todos productos del proyecto sobre el control de CM. Ver Arriba.
Transición	Poner todos productos del proyecto sobre el control de CM. Ver Arriba.

Fases de la disciplina de administración de la configuración
[Fuente: Ambler, S.]

2.3.5.8 Disciplina Administración de proyecto

El objetivo de esta disciplina es dirigir las actividades a lo largo del proyecto. Esto incluye la administración del riesgo, administración del personal (asignación de tareas, rastreo del progreso, etc.), y coordinación con personas y sistemas fuera del alcance del proyecto para asegurar su liberación a tiempo y dentro del presupuesto.

a) Flujo de trabajo

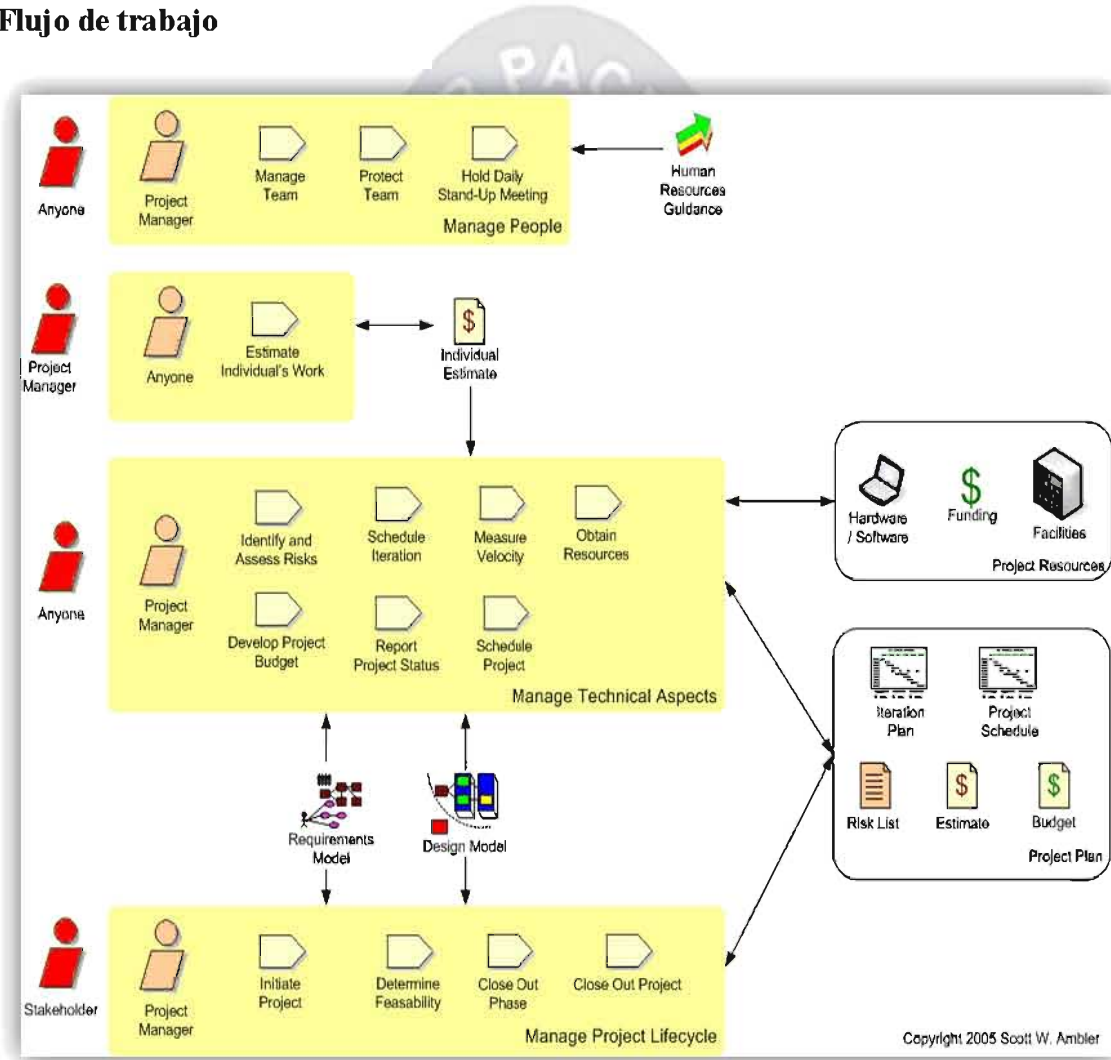


Figura 2.8: Flujo de trabajo de la disciplina de Administración de proyecto

Fuente: Ambler, S. 2005

b) Fase por Fase

Se observa las fases de la disciplina de administración de proyecto en la siguiente tabla.

Tabla 2.7

Fases	Actividades
Inicio	<p>Inicie conformando el equipo. En este punto necesitará alguien con habilidades de modelado para trabajar con los usuarios para identificar los requerimientos iniciales del sistema y con las personas técnicas para identificar una arquitectura potencial.</p> <p>Crear relaciones con sus involucrados del proyecto. El soporte a los usuarios y la participación es crítica para su éxito.</p> <p>Desarrollar un cronograma de alto nivel para todo el proyecto. El cronograma del proyecto debe mostrar su proyecto organizado en iteraciones, indicar descripciones de los principales hitos, tareas con dependencias críticas, tanto para su equipo de trabajo como para otros equipos, y su fecha de finalización prevista.</p> <p>Desarrollar un plan detallado de iteración para la siguiente iteración. La planificación detallada se realiza basada en el principio justo a tiempo. Cuando hay eventos importantes, tales como una revisión o una tarea crítica de alguna dependencia del proyecto, en una próxima iteración el director del proyecto debería trabajar con las personas afectadas antes de tiempo para planificar las actividades adecuadas. Con respecto a la planificación detallada para una iteración, la mejor forma de hacerlo es simplemente agrupar el equipo en el inicio de una iteración y trabajar con ellos para planificar su trabajo: la persona que va a hacer el trabajo es a menudo la mejor adaptada para planearlo.</p>
Elaboración	<p>Construya el equipo. Conforme su proyecto tome forma y crezca, necesitará agregar miembros al equipo. Durante esta fase necesitará personas con habilidades de análisis, desarrollo e implementación. Seguramente tendrá que capacitar a su equipo en las nuevas habilidades de desarrollo.</p> <p>Proteger el equipo. Las políticas de empresa son una realidad y un buen administrador de proyectos</p>

	<p>protegen a sus equipos lo mejor posible.</p> <p>Obtener recursos. Su equipo necesita financiación, instalaciones (por ejemplo salas y cubículos), hardware, software, y así sucesivamente para hacer su trabajo.</p> <p>Manejo del riesgo. Continúe los esfuerzos de administración del riesgo.</p> <p>Actualice el plan del proyecto. Continúe las actividades planeadas tal y como las describió.</p> <p>Cerrar esta fase. Tendrá que ejecutar la revisión del ciclo de vida de la arquitectura, cuya principal finalidad es demostrar que su arquitectura funciona y que se está enfrentando correctamente los principales riesgos del proyecto.</p>
Construcción	<p>Administre el equipo. Continúe desarrollando el equipo, manténgase protegiéndolos y proveyéndoles los recursos que necesitan.</p> <p>Manejo del riesgo. Continúe los esfuerzos de administración del riesgo.</p> <p>Actualizar su plan de proyecto. Durante la fase de construcción necesitará asegurar que tiene identificadas las principales dependencias involucradas en el desarrollo exitoso de su sistema. Debe considerar las necesidades de sus equipos de operación y soporte, capacitación del usuario final, y el plan de pruebas al sistema piloto.</p> <p>Cerrar esta fase. Tendrá que ejecutar la revisión de la capacidad operativa inicial, cuya principal finalidad es demostrar que su equipo ha desarrollado un sistema que está potencialmente listo para implementarse en producción.</p>
Transición	<p>Administrar el equipo, incluye el equipo de desarrolladores, de pruebas e implementadores.</p> <p>Cerrar esta fase. Tendrá que ejecutar la revisión de los productos entregables, cuya principal finalidad es demostrar que su sistema ha pasado las pruebas y es</p>

	<p>aceptable para los involucrados.</p> <p>Iniciar el próximo ciclo del proyecto. Los sistemas se desarrollan y se ponen en producción de manera incremental. Durante la fase de transición del entregable N, deberá comenzar los primeros esfuerzos del entregable N+1.</p>
--	--

Fases de la disciplina de administración de proyectos
[Fuente: Ambler, S.]

2.3.5.9 Disciplina del entorno

El objetivo de esta disciplina es soportar el resto del esfuerzo asegurando que el proceso apropiado, las guías (normas y directrices), y herramientas (hardware y software) estén disponibles para cuando el equipo las necesite.

a) Flujo de trabajo

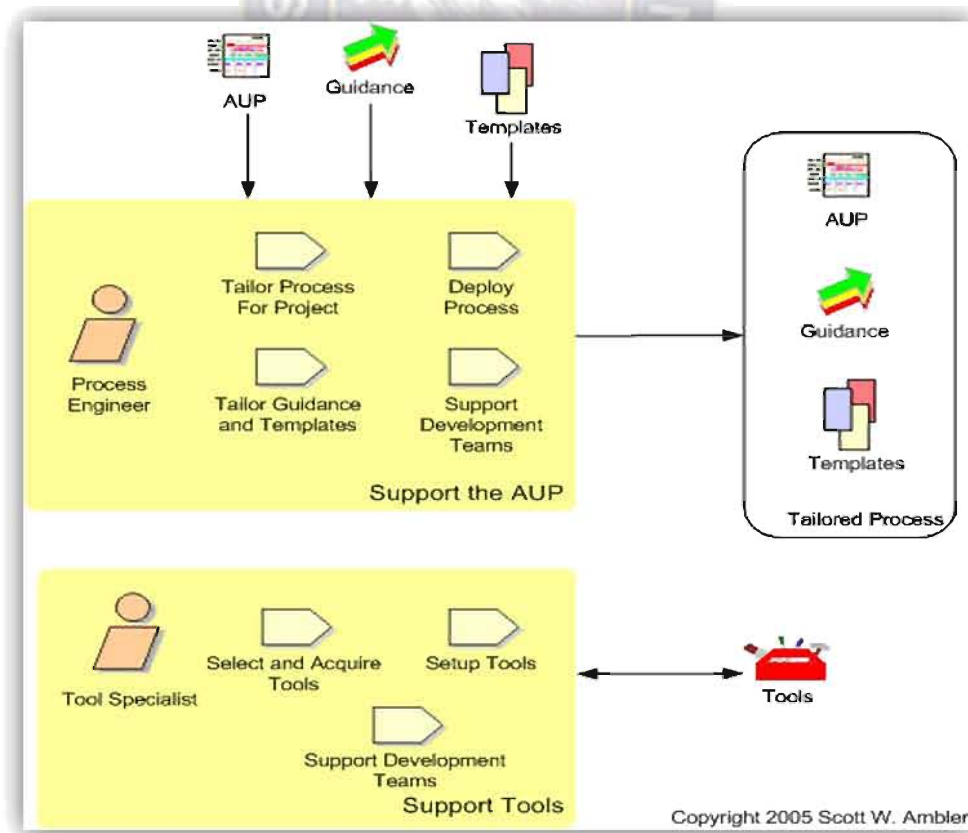


Figura 2.9: Flujo de Trabajo de la disciplina de entorno
[Fuente: Ambler, S.]

b) Fase por fases

Se observa las fases de la disciplina de entorno en la siguiente tabla.

Tabla 2.8

Fases	Actividades
Inicio	<p>Configure el entorno de trabajo. Esta será una tarea permanente ya que hay gente que se añade al equipo en el tiempo.</p> <p>Identifique la categoría del proyecto. Muchas organizaciones desarrollan varias versiones de sus procesos de software, por ejemplo uno para equipos pequeños, uno para remplazar sistemas legales, otro para sistemas de plataforma comercial, etc. Esto brinda un punto de partida para ajustar la AUP a cumplir con las necesidades para cada proyecto porque han ocurrido muchos ajustes comunes.</p>
Elaboración	<p>Evolucionar el entorno de trabajo. Su proyecto progresa a medida que se entiende la evolución de los requisitos, la estrategia de la arquitectura, y su enfoque general. El resultado es que necesitará evolucionar su entorno instalando nuevas herramientas, o remover las herramientas que ya no necesita.</p> <p>Ajuste de los procesos de materiales. Se debe ajustar AUP para cumplir con las necesidades del equipo. Esto puede incluir materiales del proceso de AUP (por ejemplo esta página), se debe elegir entre escribir un documento corto que no se quiere hacer, o se debe simplificar eligiendo la cosa correcta en el momento correcto.</p>
Construcción	<p>Apoyar al equipo. Miembros del equipo del proyecto necesita ayuda para utilizar y / o la configuración de diversas herramientas para satisfacer sus necesidades. También se necesita ayuda para elegir las plantillas de la documentación adecuada y seguir la guía de su empresa.</p> <p>Establecer el ambiente de capacitaciones. A medida que progrese en el plan de despliegue o liberación se debe descubrir que se necesita entrenar al usuario, personal de soporte y el personal de operación. Este esfuerzo de capacitación debe requerir espacios de entrenamiento y versiones de entrenamiento del sistema disponibles, frecuentemente en la fase de Transición. Se puede iniciar estableciendo estos pilares del entorno y finalizar en la fase de Construcción.</p>
Transición	Configuración de las operaciones y soporte de los entornos. Personal de

	<p>sopORTE, y algunas veces personal de operación, frecuentemente se necesita una versión del sistema configurada que se use para simular reportes de defectos en una forma segura.</p> <p>Recobrar licencias de software. A medida que su proyecto llega a la conclusión puede ser necesario desinstalar las licencias de software los equipos que ya no necesitan el software para que las licencias puedan estar disponibles a los demás dentro de su organización.</p>
--	--

Fases de la disciplina de entorno
[Fuente: Ambler, S.]

2.4 Ingeniería web

La ingeniería web es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones de alta calidad en la World Wide Web.

La ingeniería web se debe al crecimiento desenfrenado que está teniendo la Web está ocasionando un impacto en la sociedad y el nuevo manejo que se le está dando a la información en las diferentes áreas en que se presenta ha hecho que las personas tiendan a realizar todas sus actividades por esta vía.

Desde que esto empezó a suceder el Internet se volvió más que una diversión y empezó a ser tomado más en serio, ya que el aumento de publicaciones y de informaciones hizo que la Web se volviera como un desafío para los (Ingeniería del software) ingenieros del software, a raíz de esto se crearon enfoques disciplinados, sistemáticos y metodologías donde tuvieron en cuenta aspectos específicos de este nuevo medio

2.4.1 Metodología de modelado UWE

UWE (UML-Based Web Engineering) es una propuesta basada en UML y en el proceso unificado para modelar aplicaciones web.

Esta propuesta está formada por una notación para especificar el dominio (basada en UML) y un modelo para llevar a cabo el desarrollo del proceso de modelado. Los sistemas adaptativos y la sistematización son dos aspectos sobre los que se enfoca UWE.

Además de estar considerado como una extensión del estándar UML, también se basa en otros estándares como por ejemplo: XMI como modelo de intercambio de formato, MOF para el meta modelado, los principios de modelado de MDA, el modelo de transformación del lenguaje XML.

2.4.2 UWE y su relación con UML

UWE define una extensión del Lenguaje Unificado de Modelado (UML). Ésta, es considerada como una extensión ligera de peso e incluye en su definición tipos, etiquetas de valores y restricciones para las características específicas del diseño Web, las cuales, unidas a las definiciones de UML forman el conjuntos de objetos de modelado que se usarán para el desarrollo del modelo utilizado en UWE.

Las funcionalidades que cubren UWE abarcan áreas relacionadas con la Web como la navegación, presentación, los procesos de negocio y los aspectos de adaptación.

Una de las ventajas de que UWE extienda el estándar UML es la flexibilidad de éste para la definición de un lenguaje de modelado específico para el dominio web y sobretodo la aceptación universal de dicho estándar en el campo de la ingeniería del software.

Otra gran ventaja es que actualmente existen múltiples de herramientas CASE basadas en UML, con lo cual es relativamente sencillo su utilización y ampliación para utilizar los objetos de modelado definidos en UWE.

El modelo que propone UWE está compuesto por 6 etapas o sub-modelos:

2.4.2.1 Modelo de casos de Uso

Un Caso de Uso es una representación de una unidad discreta de trabajo realizada por un usuario (u otro sistema) usando el sistema en operación.

Se ejecuta en su totalidad o no se ejecuta nada, devolviendo algo de valor al usuario. Algunos ejemplos de casos de uso son Agregar_Pedido, Eliminar_Pedido, Modificar_Pedido, Registro_Usuario, Reportes, Dirección_libros entre otros. A continuación se puede observar un ejemplo de caso de uso.

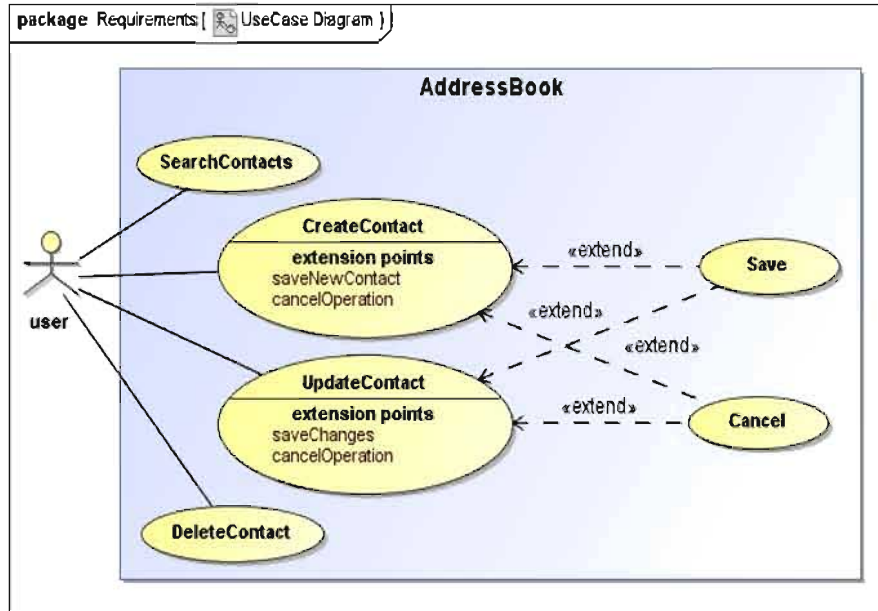


Figura 2.10: Caso de uso UWE
 [Fuente: Koch N., Kraus A.]

2.4.2.2 Modelo de contenido

Este modelo especifica cómo se encuentran relacionados los contenidos del sistema, es decir, define la estructura de los datos que se encuentran alojados en el sitio web. A continuación se muestra un ejemplo de este modelo contenido en la página web de UWE.

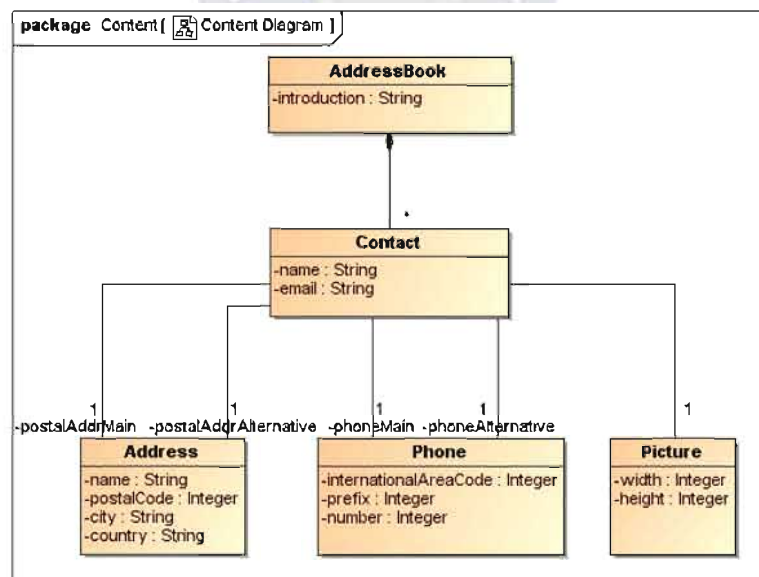


Figura 2.11: Modelo de contenido UWE
 [Fuente: Koch N., Kraus A.]

2.4.2.3 Modelo de Navegación

Este modelo especifica la relación interna del sitio web, es decir cómo se relaciona cada página web con las demás, con lo cual, en definitiva es como se navega por el sitio web.

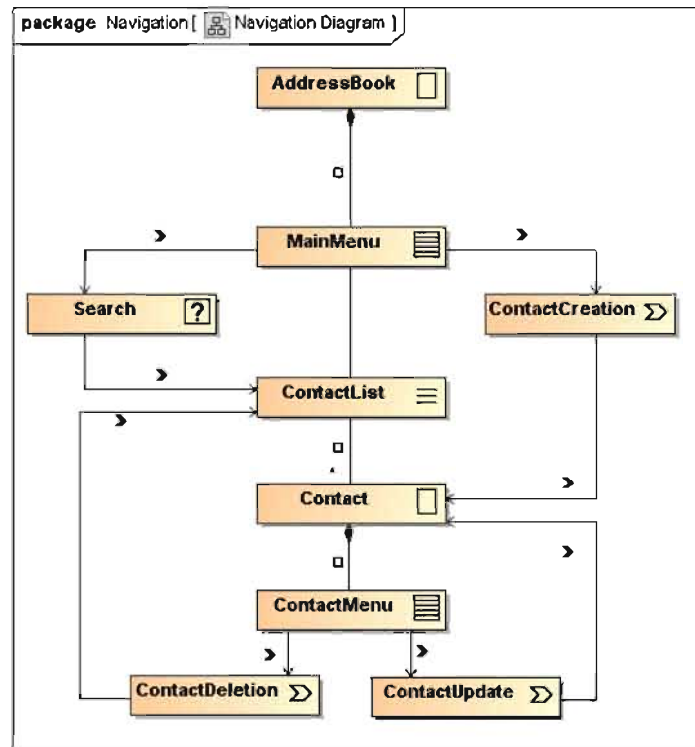


Figura 2.12: Modelo de navegación del UWE
[Fuente: Koch N., Kraus A.]

2.4.2.4 Modelo de Presentación

En este modelo se representan las clases de navegación y de procesos que pertenecen a cada página web. Estos son los elementos que introduce la metodología UWE en este modelo:

stereotype-names and their icons	
presentationClass	presentationPage
text	textInput
anchor	anchoredCollection
button	image
form	presentationGroup

Figura 2.13: Modelo de presentación
[Fuente: Koch N., Kraus A.]

2.4.2.5 Modelo de proceso

Este modelo especifica las acciones que realiza cada clase de proceso, en este modelo se incluye:

a) Modelo de Estructura de Procesos

Que define las relaciones entre las diferentes clases proceso. Un ejemplo de diagrama de clases de este modelo siguiendo el caso de la Agenda de contactos sería:

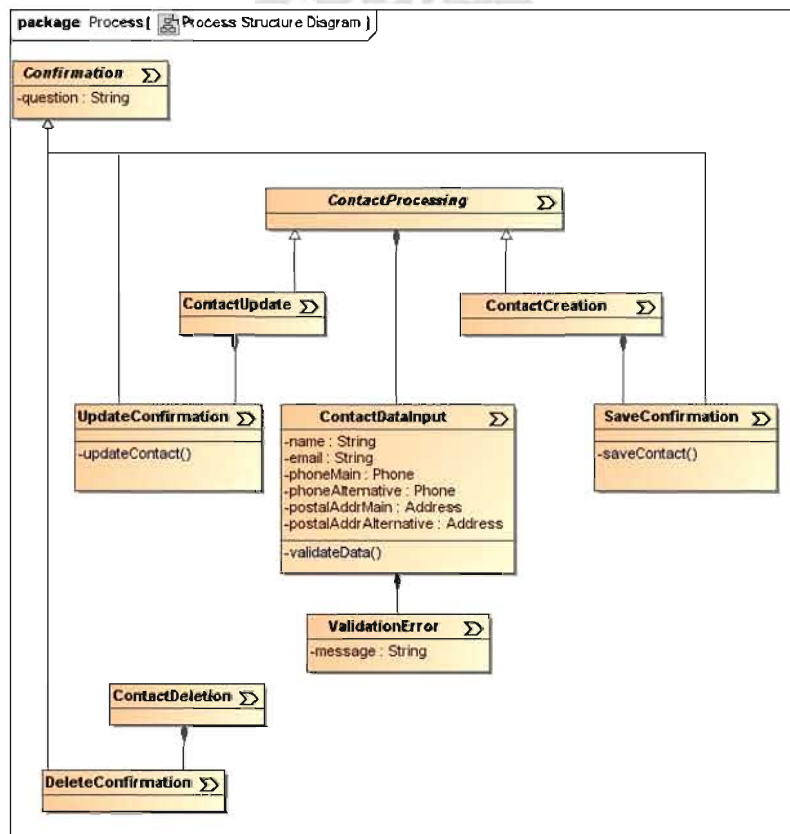


Figura 2.14: Modelo de Proceso d UWE
[Fuente: Koch N., Kraus A.]

En este diagrama se puede ver que hay clases para definir 3 operaciones que necesita una confirmación.

Modelo de Flujo de Procesos: que especifica las actividades conectadas con cada proceso. Describe los comportamientos de una clase proceso. Lo que ocurre en detalle dentro de cada una.

2.4.2.6 Modelo de adaptación

En cuanto a los requisitos, UWE los clasifica dependiendo del carácter de cada uno. Además distingue entre las fases de captura, definición y validación de requisitos.

2.5 Personal

Para la Ciencia de la administración de empresas, se denomina recursos humanos (**RRHH**) al trabajo que aporta el conjunto de los empleados o colaboradores de esa organización. Pero lo más frecuente es llamar así a la función que se ocupa de seleccionar, contratar, formar, emplear y retener a los colaboradores de la organización. Estas tareas las puede desempeñar una persona o departamento en concreto (los profesionales en Recursos Humanos) junto a los directivos de la organización.

El objetivo básico que persigue la función de Recursos Humanos con estas tareas es alinear las políticas de RRHH con la estrategia de la organización, lo que permitirá implantar la estrategia a través de las personas, quienes son consideradas como los únicos recursos vivos e inteligentes capaces de llevar al éxito organizacional y enfrentar los desafíos que hoy en día se percibe en la fuerte competencia mundial. Es imprescindible resaltar que no se administran personas ni recursos humanos, sino que se administra *con* las personas viéndolas como agentes activos y proactivos dotados de inteligencia, creatividad y habilidades intelectuales.

Generalmente la función de Recursos Humanos está compuesta por áreas tales como reclutamiento y selección, contratación, capacitación, inducción de personal y su permanencia en la empresa. Dependiendo de la empresa o institución donde la función de Recursos Humanos opere, pueden existir otros grupos que desempeñen distintas responsabilidades que pueden tener que ver con aspectos tales como la administración de la nómina de los empleados o el manejo de las relaciones con sindicatos, entre otros. Para poder ejecutar la estrategia de la organización es fundamental la administración de los Recursos humanos, para lo cual se deben considerar conceptos tales como la comunicación organizacional, el liderazgo, el trabajo en equipo, la negociación y la cultura organizacional.

Este apartado del Plan de Empresa tiene como objetivo garantizar el correcto funcionamiento de los diferentes departamentos de la empresa, así como una apropiada coordinación de los mismos.

Debe incluir una descripción detallada de las funciones de todos los puestos de la empresa, directivos y de línea. Para cada persona que ocupe un puesto de responsabilidad es necesario describir en detalle su experiencia profesional y su especialización en una determinada área funcional. Estas referencias hacen que aumente la confianza de los posibles inversores en el equipo gestor.

También conviene describir las responsabilidades y tareas concretas que cada miembro de la organización ejercerá en el proyecto.

En cuanto al personal de línea, es necesario recoger las categorías laborales que existirán en la empresa, las tareas a desempeñar por cada una de esas categorías, el convenio laboral al que se acogen los trabajadores, las fórmulas de contratación y el número de trabajadores por categoría y puesto estableciendo turnos de trabajo que garanticen el correcto funcionamiento de la empresa, y finalmente la remuneración correspondiente a cada grupo de trabajadores.

También es importante hacer referencia a la política global de la empresa en el área de recursos humanos, donde queden establecidos los planes de formación y especialización para todo el personal, las fórmulas de promoción y ascensos, y todo aquello que tenga referencia con temas sociales en la empresa.

Igualmente es conveniente confeccionar un organigrama de la empresa por áreas de actividad y que recoja las personas específicas que ocupen cada puesto de trabajo. Si esto no fuera posible, se debería hacer, al menos, un cuadro semejante al que se indica, donde se pueda ver el número de trabajadores y el departamento al que pertenecen.

2.6 Factores de Calidad ISO 9126

La calidad del software se define como el cumplimiento de los requisitos funcionales y del rendimiento establecido, en relación de los estándares de desarrollo documentados y características implícitas esperadas.

A continuación la Tabla 2.10 muestra las características que establece el estándar de la Norma ISO 9126 que constituye una guía para la evaluación de la calidad del software, determinando que cualquier componente de la calidad del software puede ser descrito en términos de las siguientes características:

Tabla 2.9

	Descripción	Sub Atributos
FUNCIONALIDAD	Nuestro sistema debe ser capaz de proveer las funciones que cumplen con las necesidades explícitas e implícitas cuando es utilizado en las condiciones especificadas por el cliente.	<ul style="list-style-type: none"> • Adecuación: es el conjunto apropiado de funciones para tareas u objetivos especificados. • Precisión: referido a resultados o efectos correctos, acordados con el cliente con el grado necesario de precisión. • Interoperabilidad: se refiere a la capacidad de nuestro sistema para interactuar con más sistemas especificados. • Conformidad: evalúa si el software se adhiere a estándares, convenciones o regulaciones en leyes. • Seguridad: Proteger la información de forma tal, que las personas o sistemas no autorizados no puedan leerla o modificarla y si pueden hacerlo sólo las que están autorizadas.
CONFIABILIDAD	Grado en que el sistema responde bajo las condiciones definidas durante un intervalo de tiempo dado	<ul style="list-style-type: none"> • Nivel de madurez: no fallar a causa de faltas en el software. • Tolerancia a fallos: indica el grado en que el sistema mantiene un nivel de respuesta ante fallos del sistema o interfaces. • Recuperabilidad: indica la capacidad que tiene el sistema para restablecer su nivel de respuesta después de un fallo crítico o de un error de hardware.
USABILIDAD	La capacidad del sistema de ser entendido, aprendido, usado y de resultado atractivo para el usuario final.	<ul style="list-style-type: none"> • Comprensión: el uso del sistema debe ser fácilmente comprensible para el usuario final. • Facilidad de aprendizaje: indica las características del software que influyen en el esfuerzo del usuario para aprender su aplicación. • Operatividad: es la dificultad que presenta el software para ser controlado por el usuario final. • Atractivo: indica las características del software que influyen en la satisfacción de los deseos del usuario y en las preferencias a través de servicios, comportamiento y presentación.

EFICIENCIA	Suministrar un desempeño correcto en relación a los recursos utilizados, bajo condiciones establecidas.	<ul style="list-style-type: none"> • Comportamiento temporal: características que influyen en el tiempo de respuesta, procesado y productividad cuando se ejecuta su función. • Utilización de recursos: usar cantidades y tipos de recursos apropiados, en condiciones especificadas (no incluye a los recursos humanos).
MANTENIMIENTO	Indica el esfuerzo requerido para implementar cambios	<ul style="list-style-type: none"> • Capacidad para ser analizado: indica la cantidad de esfuerzo requerido para diagnosticar la causa de un fallo o un nuevo requerimiento. • Capacidad de modificación: se describe como la cantidad de esfuerzo requerido para la modificación o eliminación de un defecto. • Estabilidad: indica el volumen de riesgo y de efectos inesperados luego de realizar una modificación. • Facilidad para ser probado: capacidad del software para permitir que sea validado después de ser modificado.
PORTABILIDAD	Capacidad del software para ser transferido de un ambiente de operaciones a otro.	<ul style="list-style-type: none"> • Adaptabilidad: característica del software que influye en las posibilidades de adaptación a diferentes entornos especificados, sin realizar otras acciones que las indicadas para este propósito. • Facilidad de instalación: es el esfuerzo requerido para instalar el software en un entorno especificado. • Coexistencia: capacidad del software de coexistir con otro, independiente, en un entorno común en el que se comparten recursos. • Reemplazo: característica del software que incluye en la posibilidad y en el esfuerzo requerido para usarlo en vez de otro en el mismo entorno.

Factores de calidad de software
Fuente: [CODE25, 08 – REV, 04]

CAPITULO III
MARCO PRÁCTICO

Todo lo que somos es el resultado de lo que hemos pensado; está fundado en nuestros pensamientos y está hecho de nuestros pensamientos. (Buda)

MARCO PRÁCTICO

3.1 Introducción

El objetivo del presente capítulo es formalizar el análisis y diseño del “Sistema Web De Control y Seguimiento de Personal ” caso Universidad Tecnológica Boliviana, utilizando para este fin la metodología de desarrollo AUP y siguiendo las fases del modelado UWE.

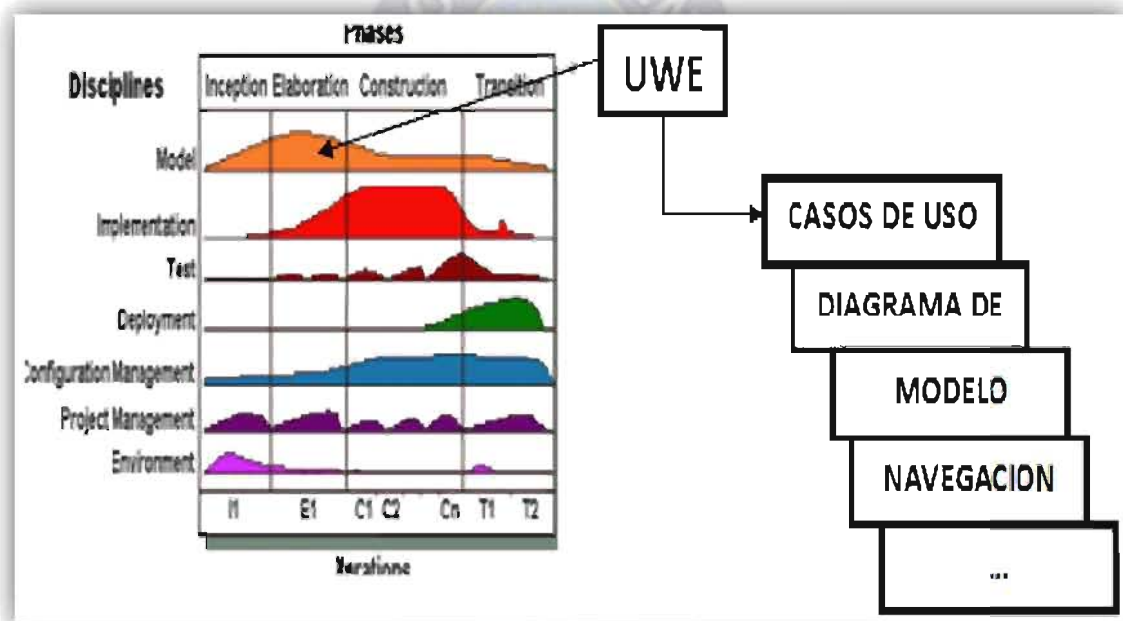


Figura 3.1: Fases y Disciplinas del AUP con UWE
[Fuente: Elaboración Propia]

Este proceso, iterativo e incremental, incluye flujos de trabajo y puntos de control, y sus fases coinciden con las propuestas en el Proceso Unificado de Modelado.

3.2 Fase de inicio

Esta fase tiene como propósito definir y acordar el alcance del proyecto, identificar los riesgos asociados al proyecto, proponer una visión muy general de la arquitectura de software. En esta fase el objetivo principal es el modelado de requerimientos de alto nivel. La fase Inicio podría ser muy breve si se trata de un problema conocido, o se ha decidido realizar el proyecto de todas formas.

3.2.1 Modelado del negocio

El modelado de negocios permitirá comprender mejor los procesos de funcionamiento de los recursos humanos de la Universidad Tecnológica Boliviana.

3.2.1.1 Modelado de casos de uso del negocio

En el diagrama de caso de uso de la Figura 3.1 se observa las operaciones que se efectúan en la UTB para el personal, se realiza el proceso de selección de personal donde interviene directamente el director administrativo que por medio de una entrevista selecciona a la persona para el puesto, al contratar a una persona para el puesto se firma un documento y se entrega los documentos pertinentes al encargado de personal para que el registre su nombre completo y su CI, además el contador es el encargado de realizar los ajustes de ingresos y descuentos que tiene un empleado posteriormente se genera la planilla de sueldo pero no antes de que el director administrativo realice los descuentos y finalmente efectúe el pago de sueldos.

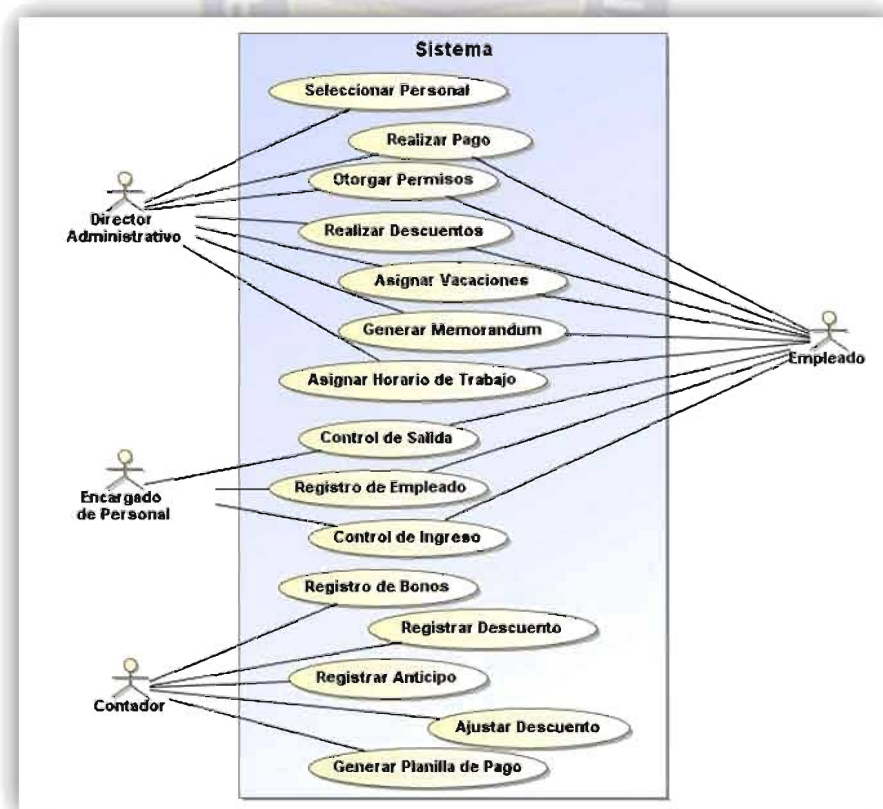


Figura 3.2: Diagrama de caso de uso del negocio
[Fuente: Elaboración Propia]

3.2.1.2 Descripción de actores del caso de uso de negocio

Descripción de los actores del diagrama de casos de uso del negocio.

Tabla 3.1

Actor	Descripción
Director administrativo	Persona encargada de la gestión del sistema, crea y administra las cuentas de usuarios, realiza el control y seguimiento, genera reportes, realiza las liquidaciones del personal y puede acceder a todas las opciones del sistema.
Encargado de Personal	Encargado del registro de datos personales, modificación y dar de baja a los empleados.
Contador	Persona encargada del ajuste de descuentos por retrasos, permisos, adelantos y otros, también realiza los ajustes de ingreso como ser bonos, horas extras, feriados y domingos. Otra de sus responsabilidades es efectuar reportes operativos de los empleados.
Empleado	Persona que tiene que proporcionar sus documentos para poder registrar sus datos personales y obtener información para el contrato.

Descripción de actores de casos de uso del negocio

[Fuente: Elaboración Propia]

3.2.2 Modelado de requerimientos

Es el conjunto de técnicas y procedimientos que nos permiten conocer los elementos necesarios para definir un proyecto de software.

Es la etapa más crucial del desarrollo de un proyecto de software. Describe los requisitos que el sistema debe cumplir.

Consta de una variedad de productos de trabajo, incluyendo: requerimientos técnicos, requerimientos de usuario, requerimiento a nivel de negocio. Los requerimientos son la parte final de la fase de inicio.

3.2.2.1 Descripción de requerimientos a nivel de negocios

Desarrollar un sistema web de seguimiento y control para el personal de la Universidad Tecnológica Boliviana.

3.2.2.2 Descripción de requerimientos a nivel de Usuario

U1.- Registrar los datos personales de un empleado.

U2.- Administrar los cargos de un empleado, es decir que se pueda adicionar, eliminar y modificar un cargo.

U3.- Registrar los contratos de un empleado.

U4.- Controlar la fecha de vencimiento de los contrato de trabajo.

U5.- Registrar todos los descuentos e ingresos de un empleado.

U6.- Generar boleta de pago con sus respectivos ingresos y descuentos cada fin de mes.

U7.- Realizar liquidación de un empleado.

U8.- Realizar la modificación de los datos de un empleado

3.2.2.3 Descripción de requerimientos a nivel de sistema

S1.- El sistema debe tener una interfaz para el registro de los datos personales del empleado.

S2.- El sistema tendrá una interfaz que nos permita elegir entre modificación y eliminación de un cargo de empleado.

S3.- El sistema debe tener una interfaz que permita registrar todos los datos relacionados con el contrato laboral.

S4.- El sistema debe dar una alerta cuando el contrato haya expirado.

S5.- El sistema tendrá una interfaz para registrar los descuentos del empleado.

S5.- El sistema tendrá una interfaz para registrar los ingresos del empleado.

S6.- El sistema tendrá la opción para imprimir boletas de pago para los empleados.

S7.- El sistema debe tener la opción para realizar la liquidación de un empleado.

S8.- El sistema tendrá la opción para buscar un empleado.

3.2.2.4 Descripción de requerimientos a nivel técnicos

Se tiene los requerimientos a nivel de hardware y software.

T1.- La codificación será realizada sobre la plataforma Microsoft, utilizando Visual Studio 2008 y SQL SERVER 2005.

T2.- Para realizar los reportes se utilizara Crystal Reports.

T3.- Servidor de aplicación y Servidor de base de datos.

3.3 Fase de Elaboración

En esta fase se determinaran las soluciones técnicas del proyecto. Durante la cual se elaboraran los requisitos al nivel del diseño y por tanto, se realizan los modelos de casos de uso, casos de uso extendidos, diagramas de secuencia, estado, clases y diagramas de navegación.

3.3.1 Modelado de Análisis

3.3.1.1 Modelo de casos de uso

Los Casos de uso representan la interacción entre los usuarios y el sistema. Este modelado de casos de uso se desarrolla a lo largo de varias iteraciones añadiendo nuevos casos de usos y mejorando la descripción de los casos de uso que ya se crearon anteriormente.

3.3.1.2 Diagrama de casos de uso de alto nivel

Se describe clara y concisamente los procesos.

Después de realizar un análisis de los requerimientos, se han identificado los siguientes casos de uso y actores, que se puede observar en la Figura 3.3.

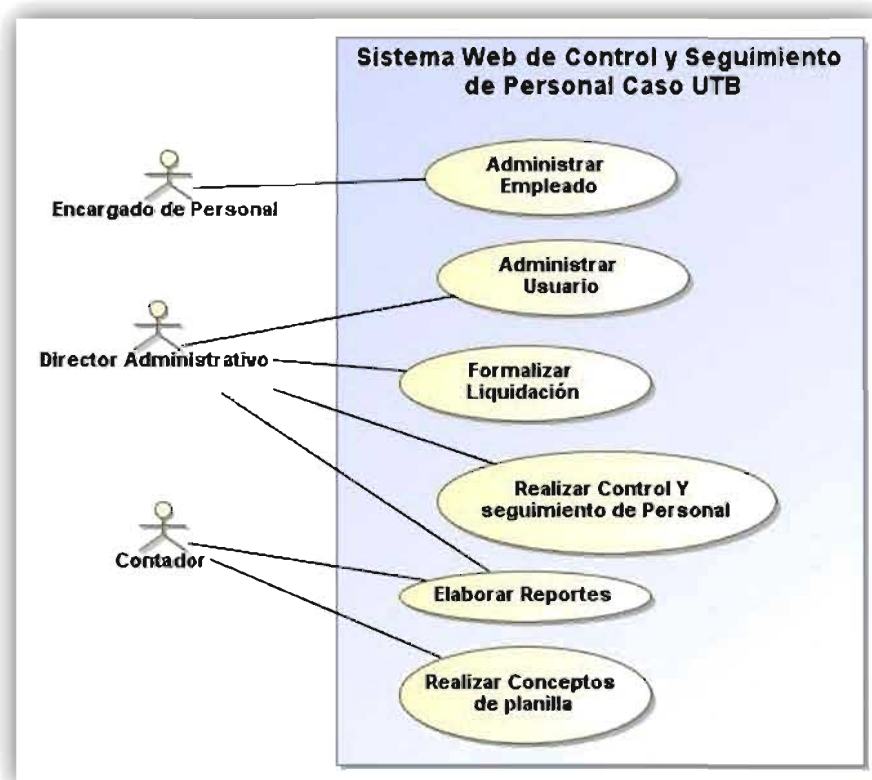


Figura 3.3: Diagrama de caso de uso del Sistema de Control y Seguimiento de Empleados [Fuente: Elaboración propia]

3.3.1.3 Descripción de casos de uso

A continuación, se describen los diagramas de casos de uso de la Figura 3.2, estos diagramas proporcionan una guía para los siguientes flujos de trabajo y su descripción, desde el diseño hasta las pruebas.

a) Caso de uso: Administrar Empleado

La administración de empleado consiste en el registro de empleado que no es más que introducir todos los datos personales del empleado, la modificación de empleado que se realiza en caso de tener algún cambio de información después de haber realizado el registro y finalmente la eliminación de un empleado que se puede realizar cuando se despide a un empleado. La persona a cargo de todos estos procesos es el encargado de personal que su labor

consistirá en el manejo de la administración de personal. A continuación se puede ver lo mencionado en la figura 3.4.

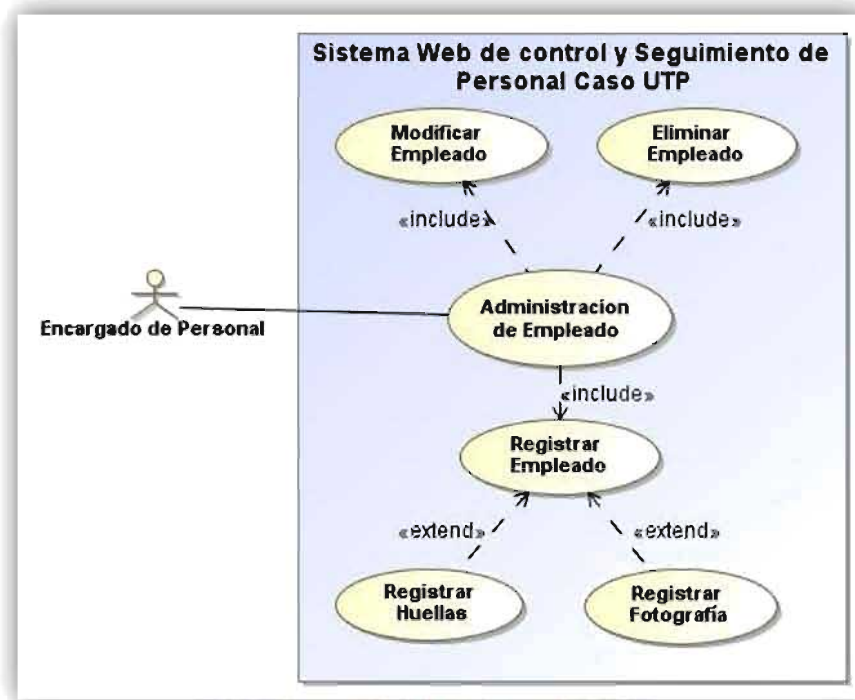


Figura 3.4: Diagrama de caso de uso: Administración de empleado
[Fuente: Elaboración Propia]

Tabla 3.2

Descripción del diagrama de Caso de Uso: Administración de personal	
Nombre:	Administración de Personal
Autor:	María Juana Aguilar Torrez
Descripción:	Permite al encargado de personal registrar, modificar y eliminar empleado.
Actores:	Encargado de Personal
Precondiciones:	El actor debe ser usuario del sistema y tener el rol correspondiente.
Flujo Normal:	

<ol style="list-style-type: none"> 1. Autenticación del actor en el sistema. 2. Elegir la opción de adicionar, eliminar o modificar empleado y cambio de contraseña. 3. Salir del sistema.
<p>Flujo alternativo:</p> <p>En caso de cometer algún error en la adición, eliminación o modificación el sistema informará dicho error.</p>
<p>Pos condiciones:</p> <p>El sistema guarda los cambios en la administración de personal.</p>

Descripción de caso de uso: Administración de personal
[Fuente: Elaboración Propia]

b) Caso de uso: Administrar Usuario

La administración de usuario consiste en la asignación de usuarios que es la dar un rol a un empleado en el sistema, la modificación de usuario en caso de haber cometido algún error al habilitar a algún empleado y finalmente la eliminación de usuario que se puede realizar cuando se despide a un empleado. La persona en cargo de todos estos procesos es el director administrativo. A continuación se puede ver lo mencionado en la figura 3.5.

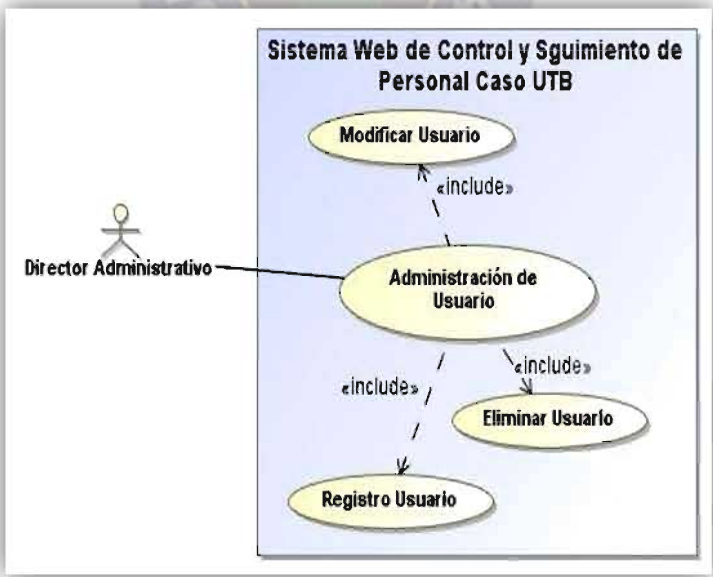


Figura 3.5: Diagrama de caso de uso: Administración de usuario
[Fuente: Elaboración Propia]

Tabla 3.3

Descripción del diagrama de Caso de Uso: Administración de usuario	
Nombre:	Administración de Usuario
Autor:	María Juana Aguilar Torrez
Descripción:	Permite al director administrativo registrar, modificar y eliminar usuario.
Actores:	Director Administrativo.
Precondiciones:	El actor debe ser usuario del sistema y tener el rol de administrador.
Flujo Normal:	<ol style="list-style-type: none"> 1. Autenticación del actor en el sistema. 2. Elegir la opción de administración de usuario. 3. Elegir la opción de adicionar, eliminar o modificar usuario. 4. Salir del sistema.
Flujo alternativo:	En caso de cometer algún error en la administración de usuario el sistema informará dicho error.
Pos condiciones:	El sistema guarda los cambios en la administración de usuario.

Descripción de caso de uso: Administración de usuario
 [Fuente: Elaboración Propia]

c) Caso de uso: Formalizar Liquidación

La formalización de liquidación, consiste en la selección de un empleado que ya no va a trabajar en la UTB, entonces se debe hacer un finiquito o liquidación del empleado basado en la ley del trabajo realizando las operaciones de promediar los últimos tres meses trabajados, obtener datos de contrato, generar importe líquido, cálculo de deducciones, cálculo de beneficios sociales y finalmente imprimir la liquidación. A continuación se puede ver lo mencionado en la figura 3.6.

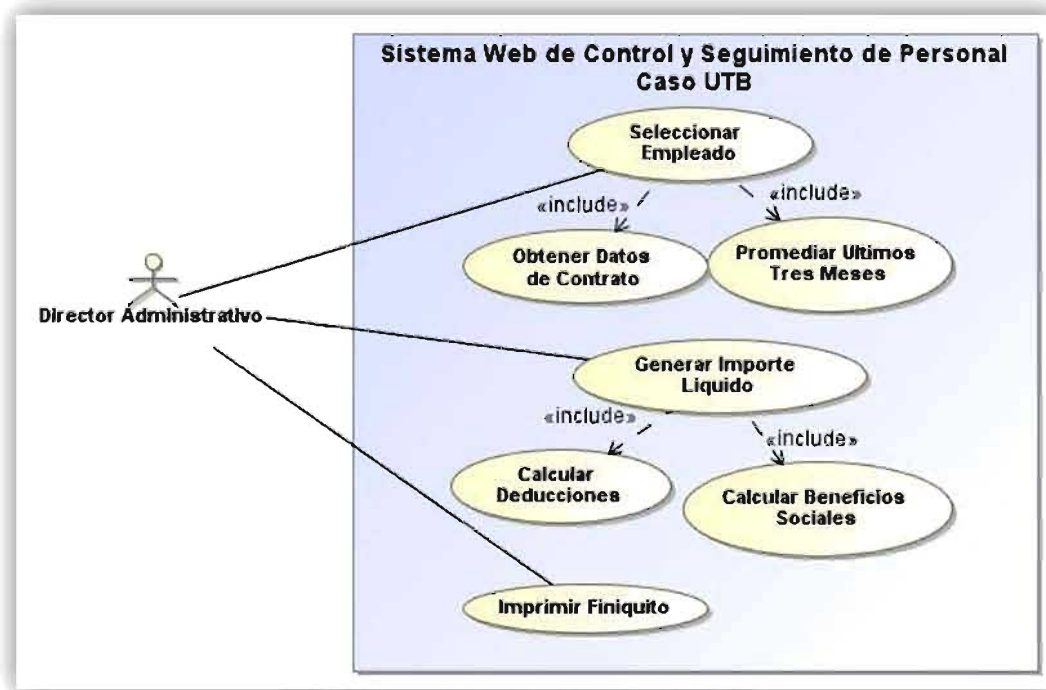


Figura 3.6: Diagrama de caso de uso: Formalizar Liquidación
[Fuente: Elaboración Propia]

Tabla 3.4

Descripción del diagrama de Caso de Uso: Liquidación	
Nombre:	Liquidación
Autor:	María Juana Aguilar Torrez
Descripción:	Permite al director administrativo realizar finiquito.
Actores:	Director Administrativo.
Precondiciones:	El actor debe ser usuario del sistema y tener el rol de administrador.
Flujo Normal:	<ol style="list-style-type: none"> 1. Autenticación del actor en el sistema. 2. Elegir la opción de liquidación. 3. Seleccionar a un empleado. 4. Seleccionar realizar finiquito

<p>5. Imprimir finiquito.</p> <p>6. Salir del sistema.</p>
<p>Flujo alternativo:</p> <p>En caso de cometer algún error en la liquidación de empleado el sistema informará dicho error.</p>
<p>Pos condiciones:</p> <p>El sistema guarda los cambios en la liquidación.</p>

Descripción de caso de uso: Formalizar Liquidación
[Fuente: Elaboración Propia]

d) Caso de uso: Realizar control y seguimiento de personal

El control y seguimiento de personal, se realiza a los procesos que realiza un empleado como asignación de vacación, generar memorándums, asignación de horarios de trabajo, gestionar contrato, asignar sucursal, asignar departamento, asignar cargo, todos estos procesos son realizados por el administrador de personal. Lo que se puede observar en la figura 3.7.

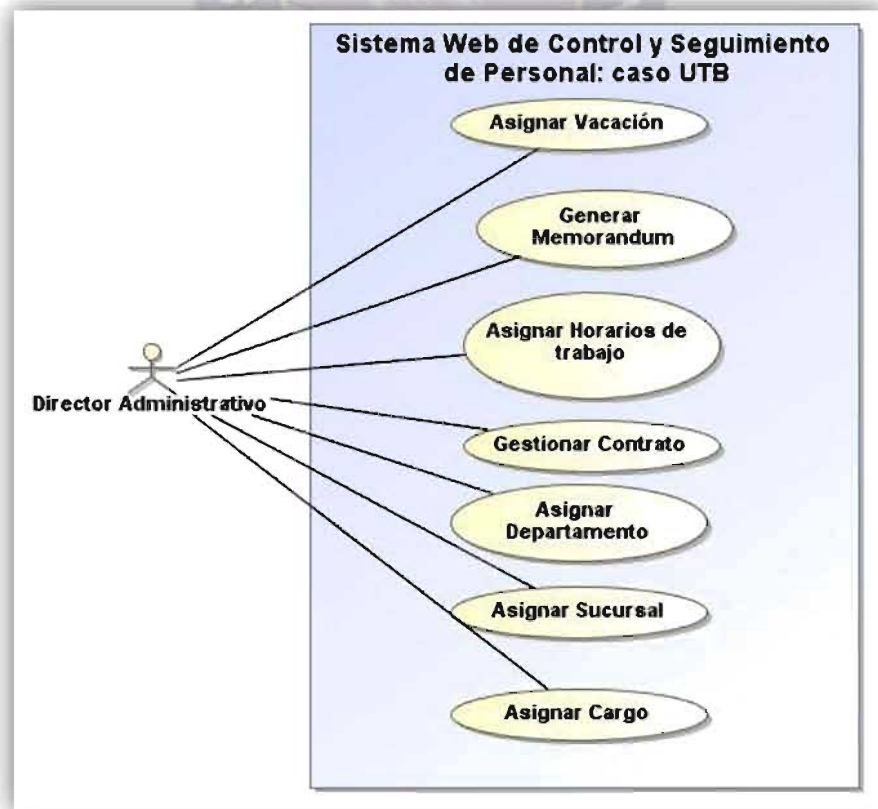


Figura 3.7: Diagrama de caso de uso: Control y seguimiento de personal
[Fuente: Elaboración Propia]

Tabla 3.5

Descripción del diagrama de Caso de Uso: Control y seguimiento de personal	
Nombre:	Control y seguimiento de personal
Autor:	María Juana Aguilar Torrez
Descripción:	Permite al director administrativo realizar el control y seguimiento de personal.
Actores:	Director Administrativo.
Precondiciones:	El actor debe ser usuario del sistema y tener el rol de administrador.
Flujo Normal:	<ol style="list-style-type: none"> 1. Autenticación del actor en el sistema. 2. Elegir la opción de control y seguimiento. 3. Seleccionar a un empleado. 4. Seleccionar una opción para realizar un control. 5. Salir del sistema.
Flujo alternativo:	En caso de cometer algún error en el control y seguimiento de empleado el sistema informará dicho error.
Pos condiciones:	El sistema genera reportes de control y seguimiento de personal.

Descripción de caso de uso: Control y seguimiento de personal
 [Fuente: Elaboración Propia]

e) Caso de uso: Elaborar Reportes

Laborar reporte, consiste en la culminación de algún proceso como realizar pagos a los empleados, donde se necesita una boleta de pago, en el caso de dar adelantos se necesita imprimir un comprobante de adelanto, luego la impresión de planillas de pago mensuales para todos los empleados, también es necesario generar historiales laborales y reportes operativos. Estos procesos son realizados por el contador y también por el administrador personal, todos estos procesos se pueden ver en la figura 3.8.

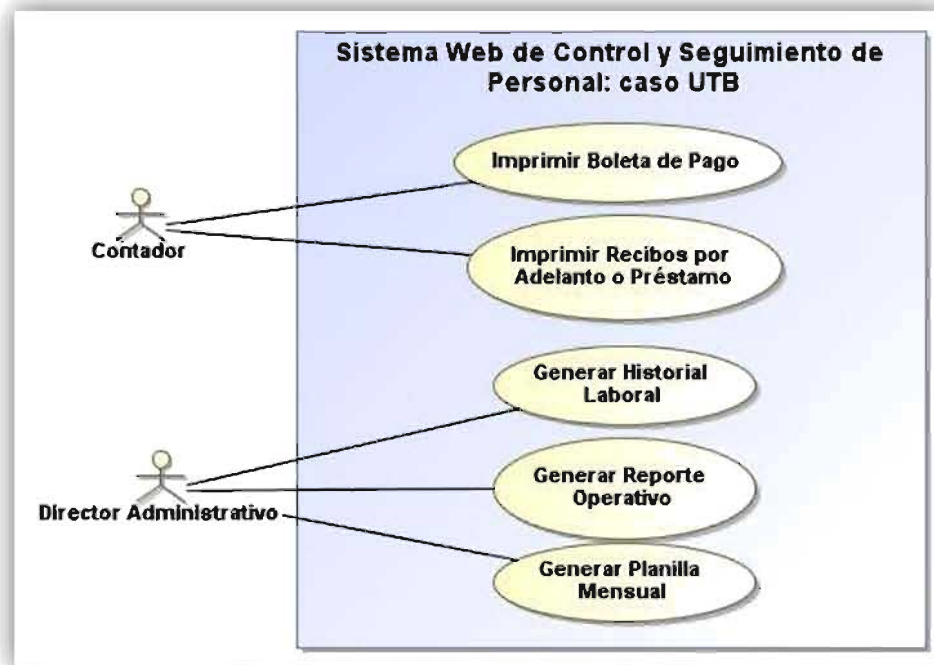


Figura 3.8: Diagrama de caso de uso: Reportes
[Fuente: Elaboración Propia]

Tabla 3.6

Descripción del diagrama de Caso de Uso: Reportes	
Nombre:	Reportes
Autor:	María Juana Aguilar Torrez
Descripción: Permite al director administrativo y contador realizar reportes.	
Actores: Director Administrativo, contador.	
Precondiciones: Los actores deben ser usuario del sistema y tener el rol respectivo.	
Flujo Normal: <ol style="list-style-type: none"> 1. Autenticación de actor en el sistema. 2. Ingresar a reportes. 3. Seleccionar reporte. 4. Imprimir reporte. 	

5. Salir del sistema.
Flujo alternativo: En caso de cometer algún error en los reportes el sistema informará dicho error.
Pos condiciones: El sistema genera o imprime reportes.

Descripción de caso de uso: Reportes
Fuente: Elaboración Propia

f) Caso de uso: Realizar Conceptos de Planilla

Conceptos de planilla, es un proceso que se realiza cada mes para realizar el pago de sueldos, se deben registrar descuentos, ajustar descuentos, ajustar ingresos, registrar ingresos y luego generar la planilla de pago, también contempla la autorización de conceptos y autorización de permisos ya que son aspectos que se debe considerar para generar la planilla, estos procesos son realizados por el contador y también por el director administrativo. Se puede ver estos procesos en la figura 3.9.

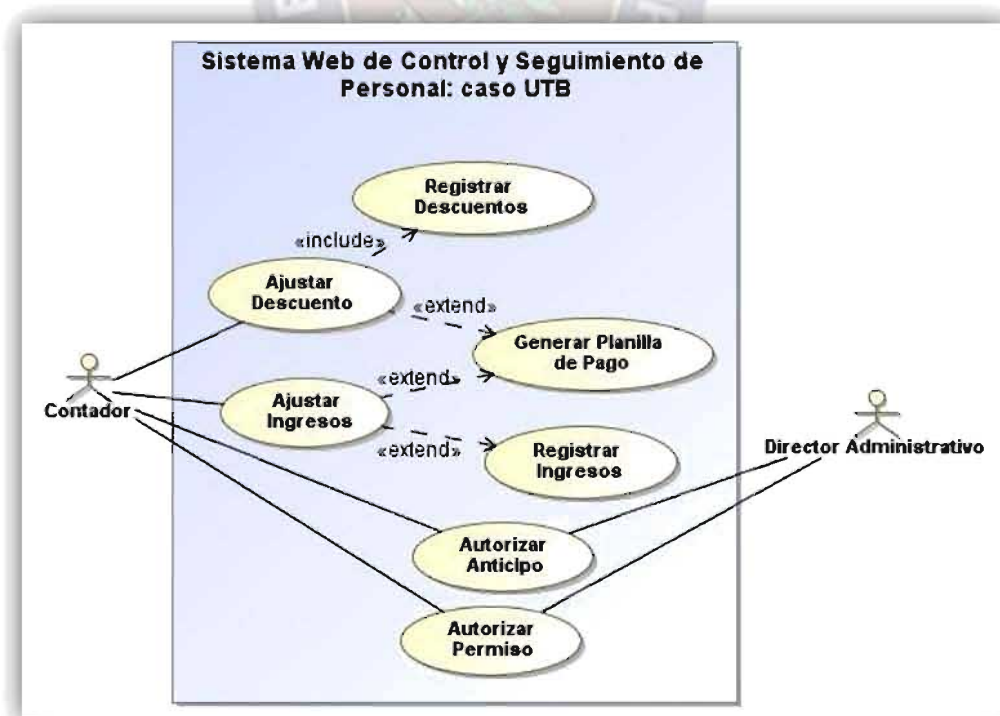


Figura 3.9: Diagrama de caso de uso: Concepto de planilla
[Fuente: Elaboración Propia]

Tabla 3.7

Descripción del diagrama de Caso de Uso: Concepto de planilla	
Nombre:	Concepto de planilla
Autor:	María Juana Aguilar Torrez
Descripción:	Permite generar planilla de pago con todas sus características.
Actores:	Director Administrativo, contador.
Precondiciones:	Los actores deben ser usuario del sistema y tener el rol respectivo.
Flujo Normal:	<ol style="list-style-type: none"> 1. Autenticación de actor en el sistema. 2. Ingresar a conceptos de planilla. 3. Registro de ingresos y descuentos 4. Seleccionar cálculo de conceptos para la planilla. 5. Salir del sistema.
Flujo alternativo:	En caso de cometer algún error en los conceptos de planilla el sistema informará dicho error.
Pos condiciones:	El sistema genera o imprime reporte de planilla de empleado.

Descripción de caso de uso: Concepto de planilla
 [Fuente: Elaboración Propia]

3.3.1.4 Diagrama de Paquetes

El diagrama de paquetes muestra la forma en la que el Sistema Web de Control y Seguimiento está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones, los diagramas de paquetes muestran la descomposición jerárquica lógica de un sistema, es decir se muestra un esquema de los módulos que comprende nuestro sistema. A Continuación se puede observar el diagrama de paquetes en la Figura 3.10 para comprender la organización del sistema.

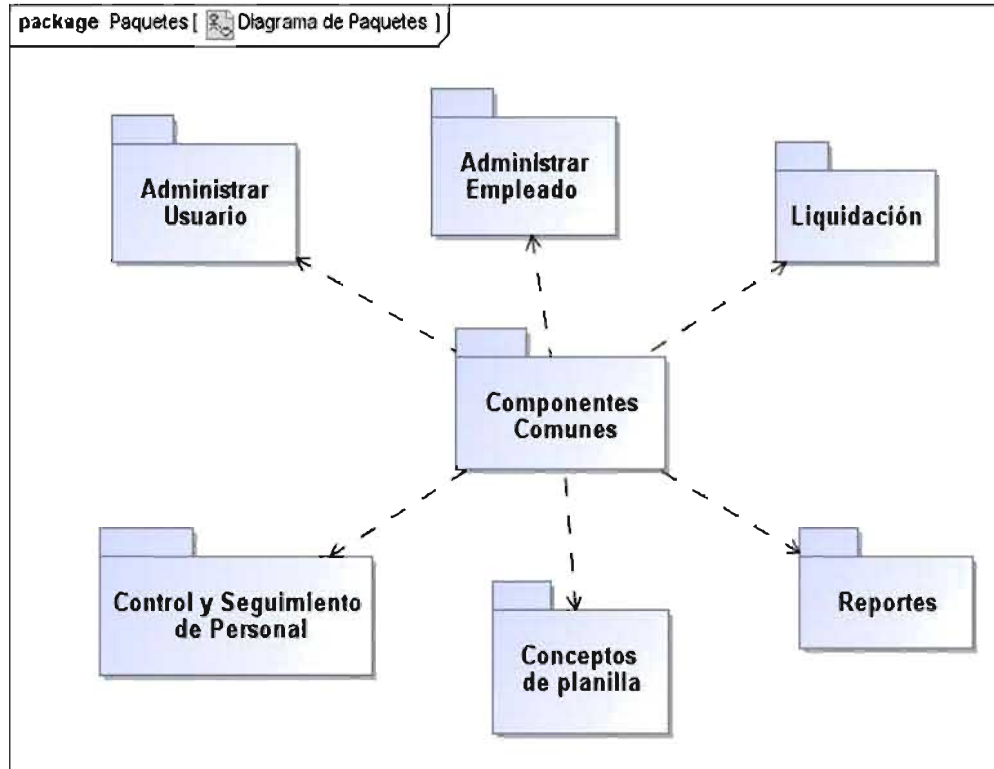


Figura 3.10: Diagrama de paquetes: Sistema Web de Control y Seguimiento de Personal
[Fuente: Elaboración propia]

3.3.2 Modelo de diseño

Diagramas descriptivos del diseño lógico, sin referencias al modo de implementación. Comprende diagramas de clases del software, diagramas de navegación y otros.

3.3.2.1 Diagrama de clases

Los diagramas se utilizan generalmente para facilitar el entendimiento de largas cantidades de datos y la relación entre diferentes partes de los datos también para realizar cálculos electrónicos.

Un diagrama de Clases representa las clases que serán utilizadas dentro del sistema y las relaciones que existen entre ellas. Nos sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de convencimiento. Un diagrama de clases está compuesto por los siguientes elementos: Clase: atributos, métodos y visibilidad. Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

Este es el diagrama principal para el análisis y diseño, en la Figura 3.11 se representan las relaciones entre las clases, atributos y sus operaciones para representar la información del sistema. Después de haber realizado el análisis para la base de datos e identificar todas las entidades que intervienen en el sistema, se elabora el diagrama de clases.

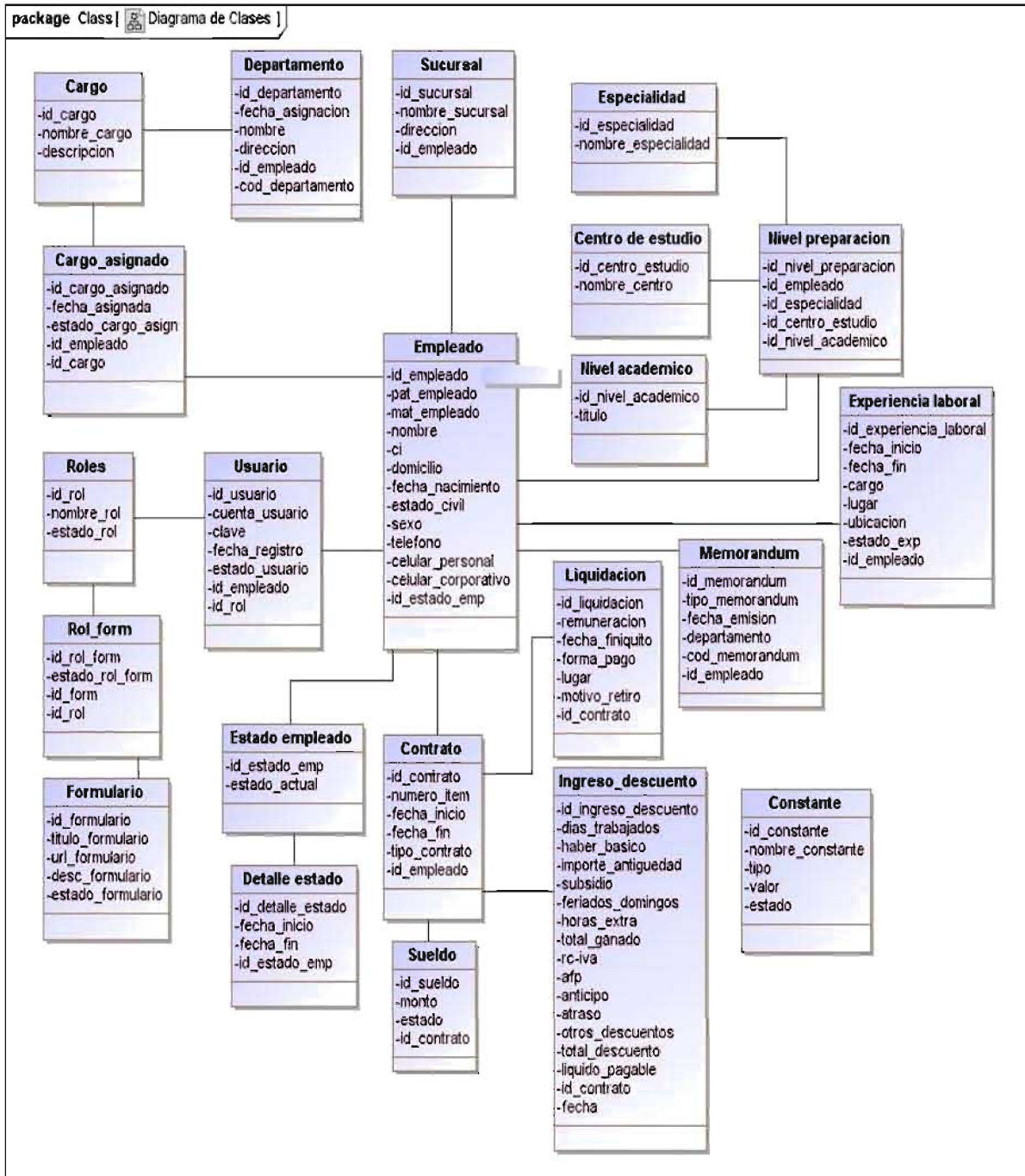


Figura 3.11: Diagrama de clases: Sistema Web de control y seguimiento de Personal
[Fuente: Elaboración Propia]

3.3.2.2 Modelo de navegación

En la Figura 3.12 se observa el espacio de navegación que especifica los objetos que pueden ser visitados mediante la navegación de los usuarios, clasificados por el rol que se les asigne. En el diagrama de navegación se tienen tres usuarios: el director administrativo, el contador, encargado de personal y en cada uno de los usuarios se puede observar su navegación.

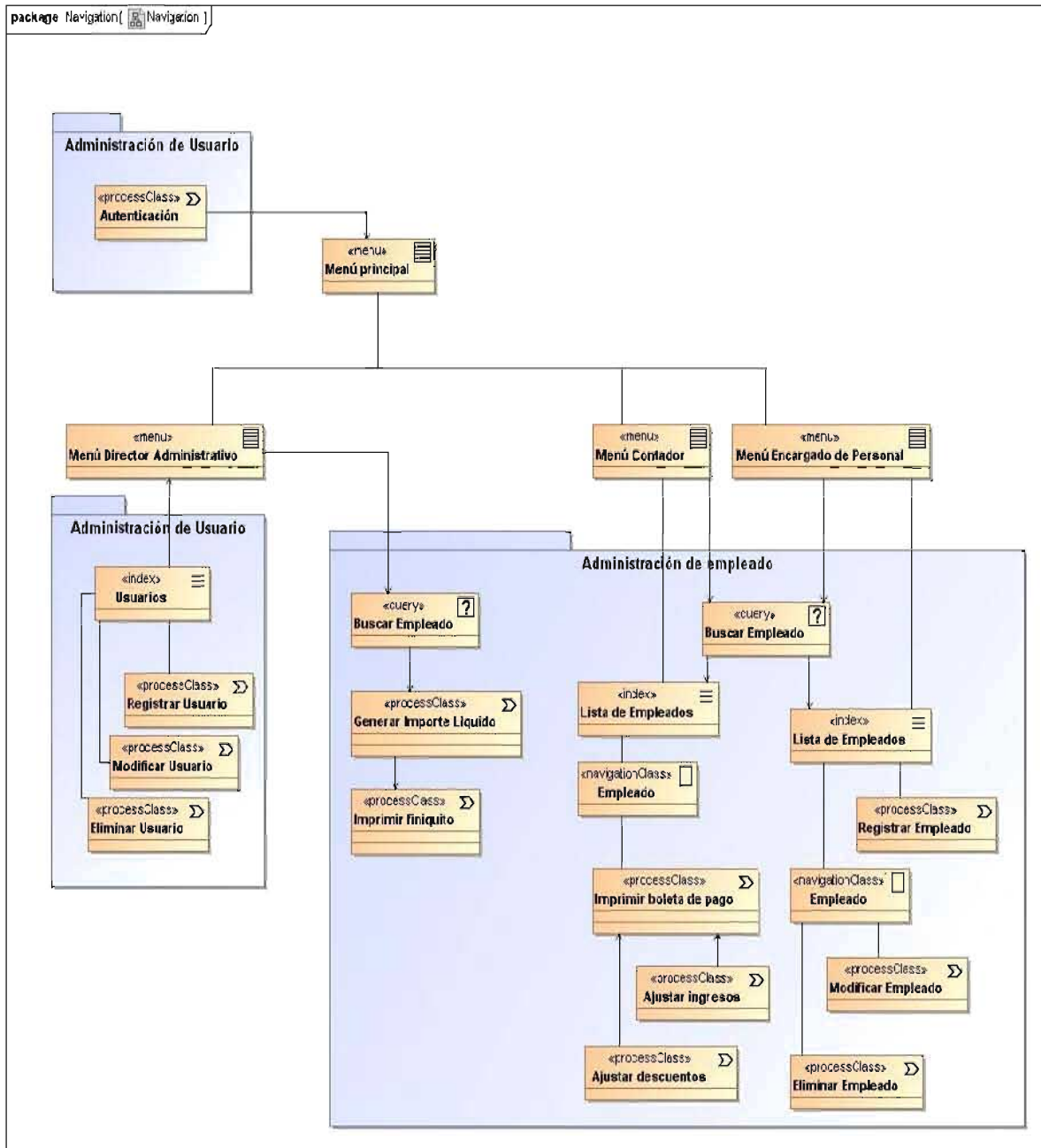


Figura 3.12: Modelo de Espacio de Navegación
[Fuente: Elaboración Propia]

3.3.2.3 Modelo de navegación de menús

La navegación de menús es un diseño de cómo se presentará nuestro sistema dependiendo el tipo de usuario que se tenga, es decir dependiendo el rol que tenga nuestro usuario.

En la figura 3.13 se puede observar las opciones del menú que tiene cada usuario, todos los usuarios deben autenticarse antes de ingresar al sistema, en el caso del usuario contador tiene las opciones de realizar ajuste de descuentos, ajuste de ingresos y también está encargado de generar las boletas de pago, en el caso del director administrativo tiene más privilegios que el contador o el encargado de personal, el director administrativo tiene las opciones de administrar usuarios, realizar las liquidaciones, realizar el seguimiento de los procesos del personal.

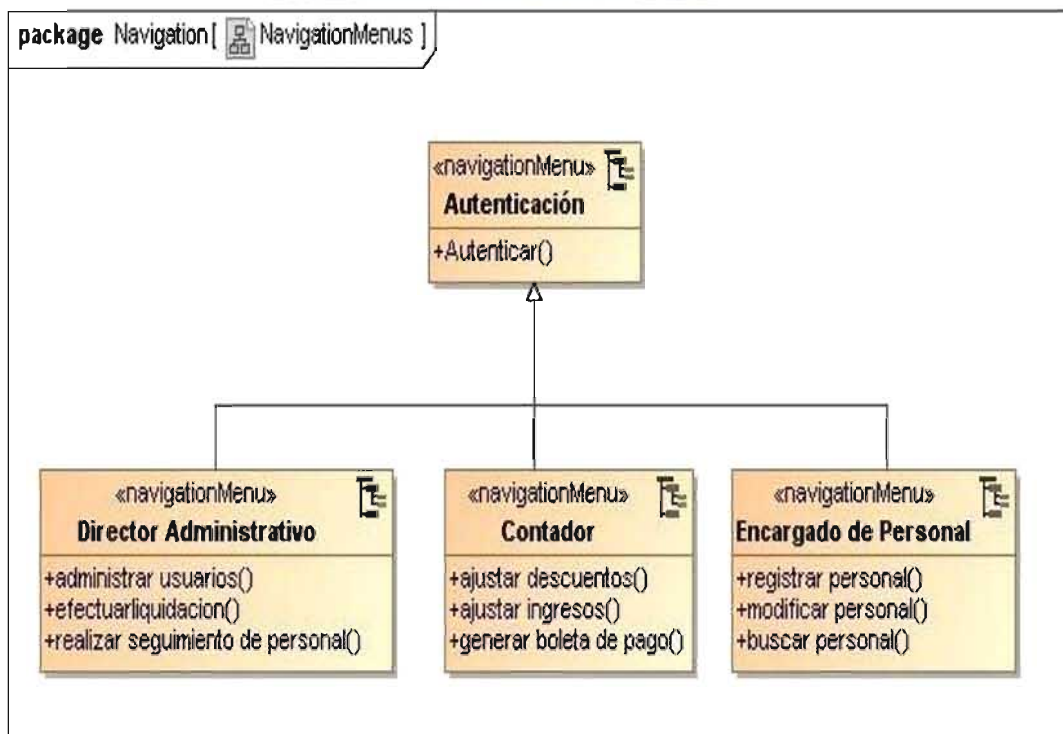


Figura 3.13: Modelo de Espacio de Navegación
[Fuente: Elaboración Propia]

3.3.2.4 Modelo de presentación

El modelo de presentación nos va mostrando cómo se verá el sistema, nosotros lo mostraremos secuencialmente

En la figura 3.14 se puede observar la página principal después de haberse autenticado en el sistema. En la página principal tenemos un esquema del diseño de cómo se verá. Se tiene el encabezado del sistema que no es otra cosa que la presentación de nuestro sistema, en encabezado se pondrá el nombre del sistema, el menú de navegación observaremos todas las opciones que tenga cada usuario, en el área de contenido se observará el resultado de la elección que se realice en el menú

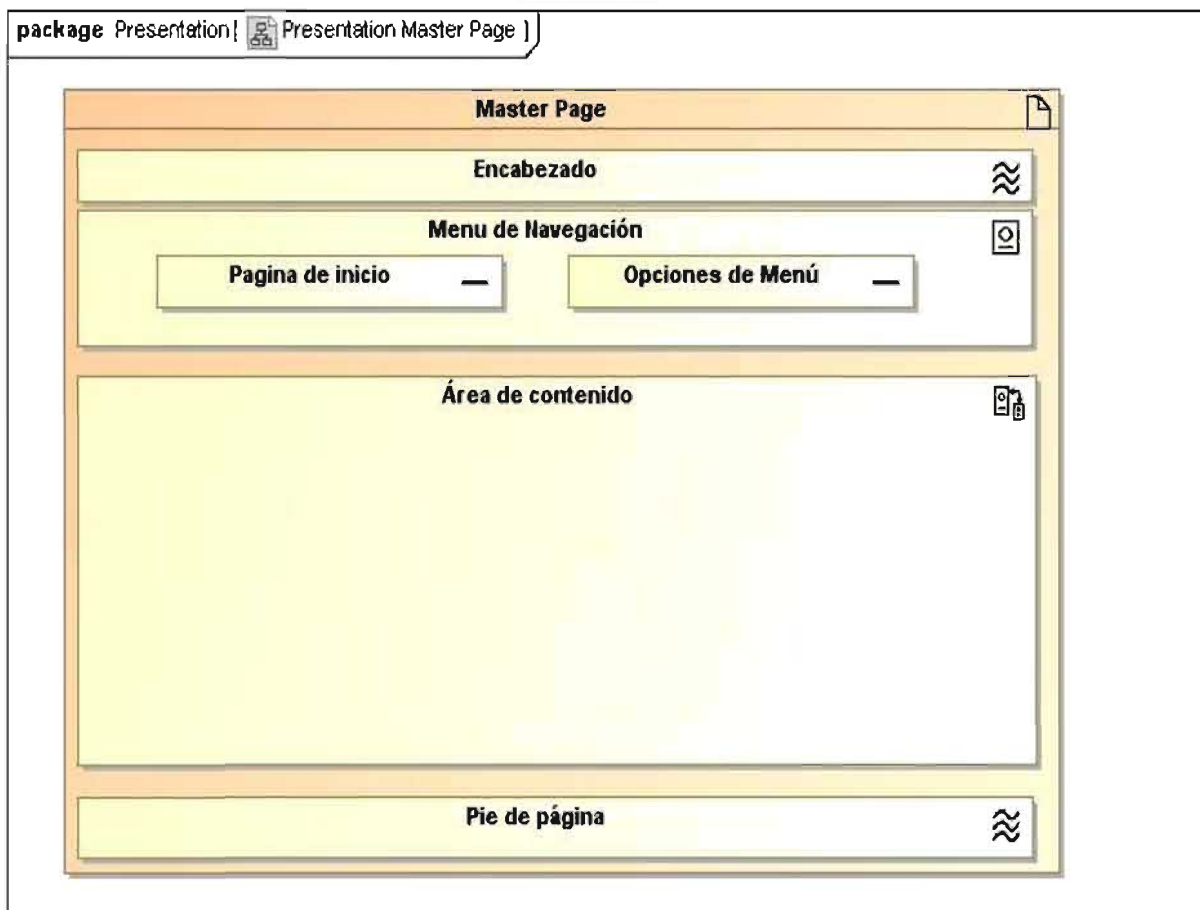


Figura 3.14: Modelo de presentación: Pagina Maestra
[Fuente: Elaboración Propia]

En la figura 3.15 se puede observar el menú del director administrativo, como este usuario tiene privilegios de administrador, podrá observar todas las opciones sin restricciones, aparte de tener sus propias funciones específicas como ser administración de usuarios, la liquidación, la asignación de cargo, departamento, horario, permisos entre otros, también registrará los contratos de los empleados, reportes y otros.

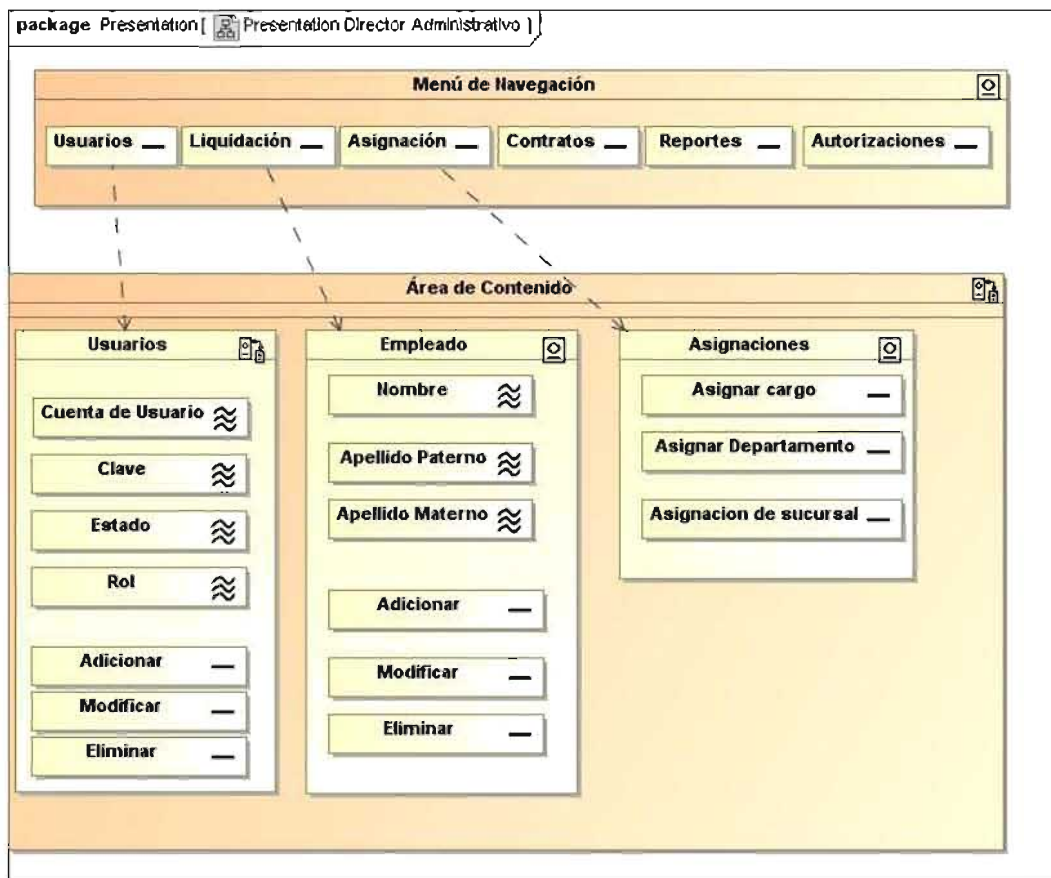


Figura 3.15: Modelo de Presentación: Pagina Director Administrativo
[Fuente: Elaboración Propia]

3.4 Fase de construcción

El objetivo de esta fase consiste en desarrollar el sistema hasta el punto en que esté listo para pre producción de pruebas.

3.4.1 Diseño de interfaces

El diseño de la interfaz del sistema se desarrolló siguiendo el modelo de requerimientos y el modelo de diseño.

3.4.1.1 Autenticación

El empleado que quiera acceder al sistema debe tener su cuenta de usuario y contraseña, que son asignados por el administrador del sistema en este caso por el director administrativo, que se muestra en la figura 3.16.

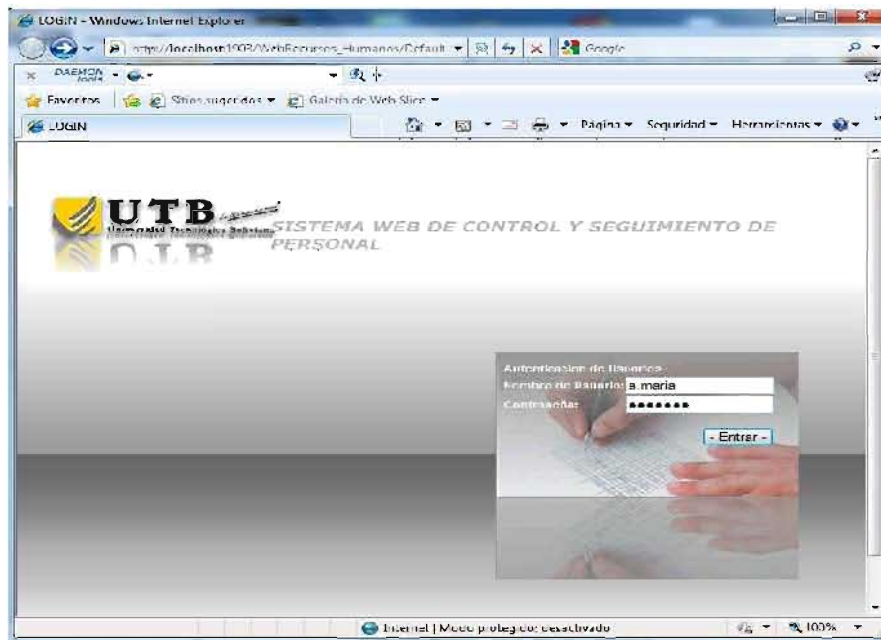


Figura 3.16: Autenticación del sistema
[Fuente: Elaboración Propia]

3.4.1.2 Pantalla principal

Una vez que el usuario ha realizado el ingreso de su cuenta de usuario y contraseña, el sistema le permitirá el ingreso a la página principal, donde dependiendo del rol asignado al usuario pueda acceder a las diferentes opciones del menú del sistema, que se muestra en la figura 3.17.



Figura 3.17: Pantalla principal
[Fuente: Elaboración Propia]

3.4.1.3 Registro de empleado

Previamente se tiene que elegir la opción de registrar en el menú empleado. En el registro de empleado se adicionan los datos personales, después de llenar toda la información se tiene la opción de finalizar para terminar el registro, continuar para registrar los datos académicos y cancelar en caso de cometer algún error cancelar el registro.

Como se puede observar en la figura 3.18.

The screenshot shows a web browser window with the URL `http://localhost:1903/WebRecursos_Humanos/Emplead`. The page title is "Página sin título - Windows Internet Explorer". The browser's address bar shows the URL. The page content includes a logo for "UTB" and the title "SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE PERSONAL". A navigation menu contains the following items: EMPLEADO, USUARIO, LIQUIDACION, SEGUIMIENTO, REPORTES, and CAMBIO CONTRASEÑA. The main content area is titled "REGISTRO DE EMPLEADO" and contains a form for entering personal data. The form fields are as follows:

DATOS PERSONALES	
Nombre Empleado	Marcelo
Apellido Paterno	Salinas
Apellido Materno	Fernandez
C.I.	3265432
Fecha de Nacimiento	23 Abril 1981
Estado Civil	Soltero(a)
Sexo	M
Dirección	Av. 4, N° 80 Villa Bolivar D
Telefono	2816590
Celular Personal	
Celular Corporativo	
Codigo Empleado	
Estado Empleado	Activo

At the bottom of the form, there are three buttons: "FINALIZAR", "CONTINUAR", and "CANCELAR". The footer of the page reads "SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE PERSONAL © 2011". The browser's status bar shows "RegistrarEmpleado.aspx" and "Internet | Modo protegido: desactivado".

Figura 3.18: Pantalla registro de empleado
[Fuente: Elaboración Propia]

Después de realizar el registro del empleado el sistema nos muestra una pantalla de mensaje, que no es más que una confirmación del registro del empleado.

Como se puede observar en la figura 3.19.

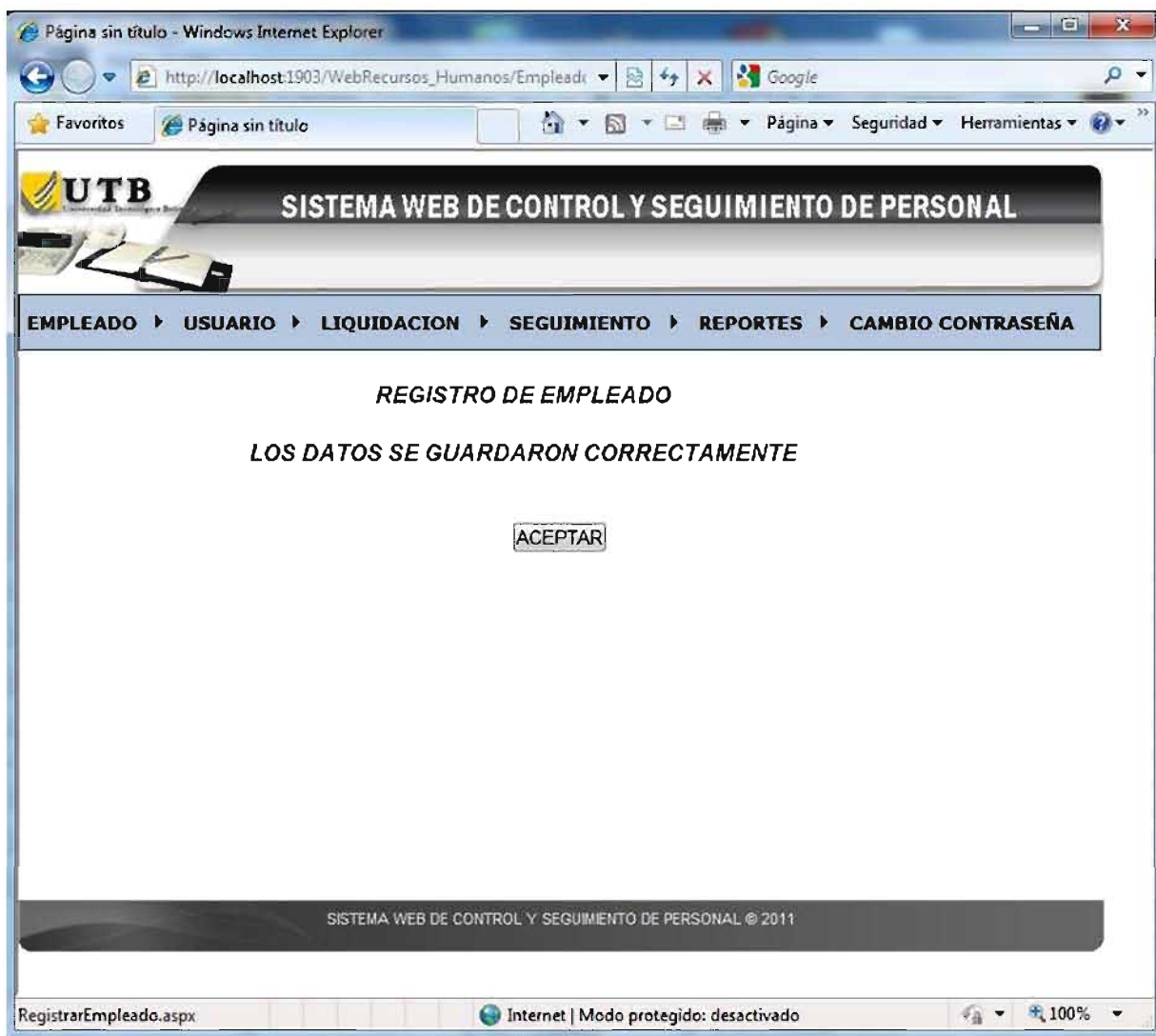


Figura 3.19: Pantalla mensaje de registro
[Fuente: Elaboración Propia]

3.4.1.4 Modificar/eliminar empleado

Al elegir la opción de modificar en el menú empleado, se puede cambiar o eliminar la información registrada anteriormente de un empleado, después de elegir la opción de modificar se ve la pantalla de modificar empleado en la que se observan un listado de empleados con las siguientes características: id de empleado, nombre y apellido, luego se tiene al final de cada columna un icono de lápiz; al elegir a un empleado de la lista también podemos dar un clic sobre el lápiz para realizar la operación de modificación.

Además el listado que se observa tiene una paginación de ocho ítems, es decir cada trece ítems podemos ir viendo el listado de empleados.

Como se puede observar en la figura 3.20.

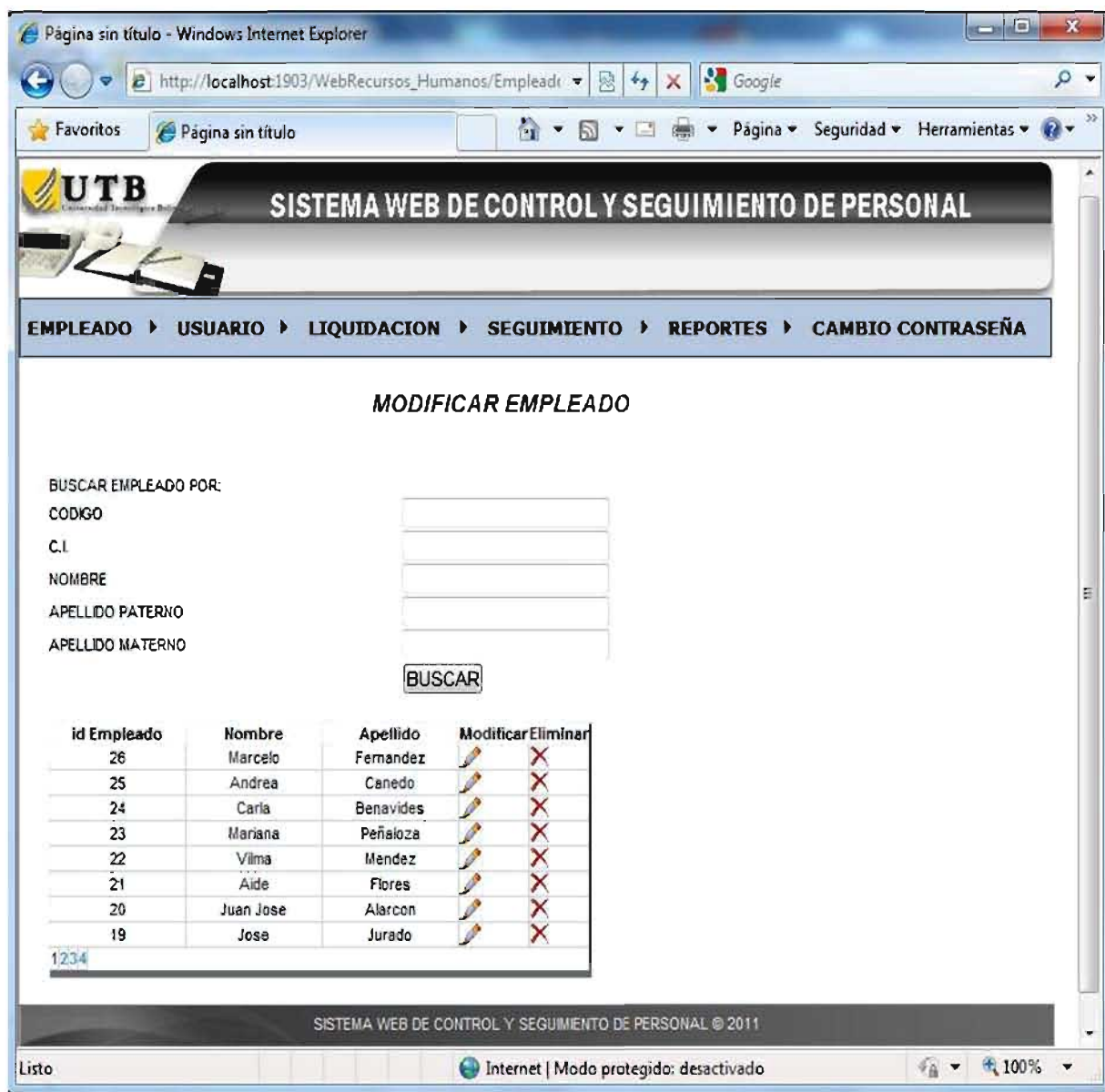


Figura 3.20: Pantalla modificar empleado
[Fuente: Elaboración Propia]

Después de elegir a un empleado se carga la siguiente pantalla que nos muestra los datos ya cargados del empleado escogido, para luego realizar algún cambio.

Como se puede observar en la figura 3.21.

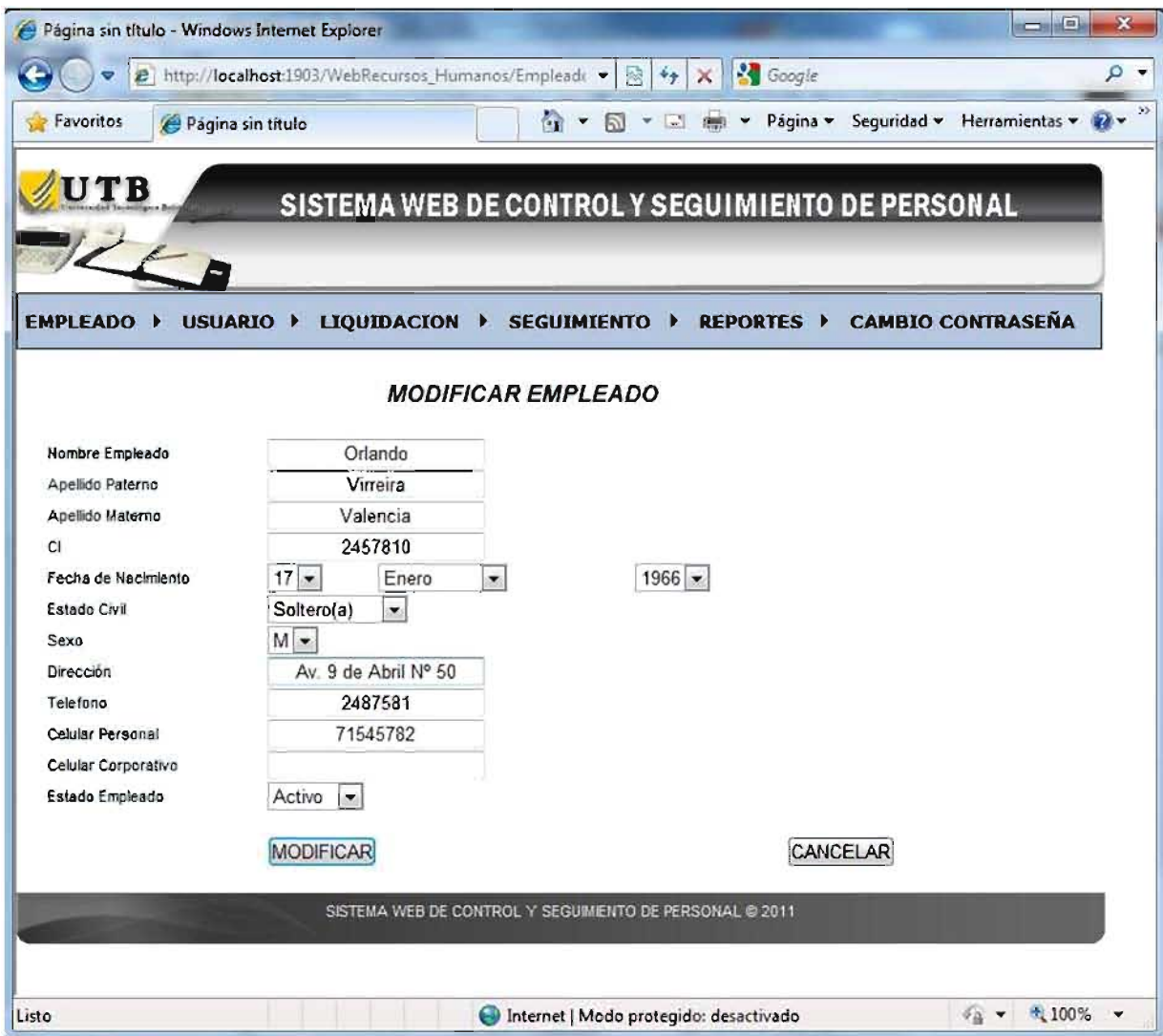


Figura 3.21: Pantalla 2 modificar empleado
[Fuente: Elaboración Propia]

3.4.1.5 Registrar usuario

En el menú principal en la pestaña usuario se tiene la opción de registro, se puede adicionar a un empleado.

Después de elegir la opción se observa la pantalla de registro de usuario en la que inicialmente se tiene un espacio de búsqueda de empleado, en la que se puede buscar por id de empleado, ci, nombre, apellido y posteriormente se debe clicar sobre el botón buscar empleado. O se tiene una opción más que es la de elegir de un listado al empleado que se desea registrar como usuario, dicho listado esta paginado cada ocho ítems.

Como se puede observar en la figura 3.22.



Figura 3.22: Registrar usuario
[Fuente: Elaboración Propia]

Después de elegir a un empleado del listado o haber realizado la búsqueda, se puede observar la siguiente pantalla en la que ya nos muestra el nombre completo del empleado luego se tiene que introducir una cuenta de usuario, clave y elegir un estado de usuario. Después de ingresar esta información se tienen dos opciones que son registrar y en caso de no efectuar la operación se tiene la opción de cancelar.

Como se puede observar en la figura 3.23.

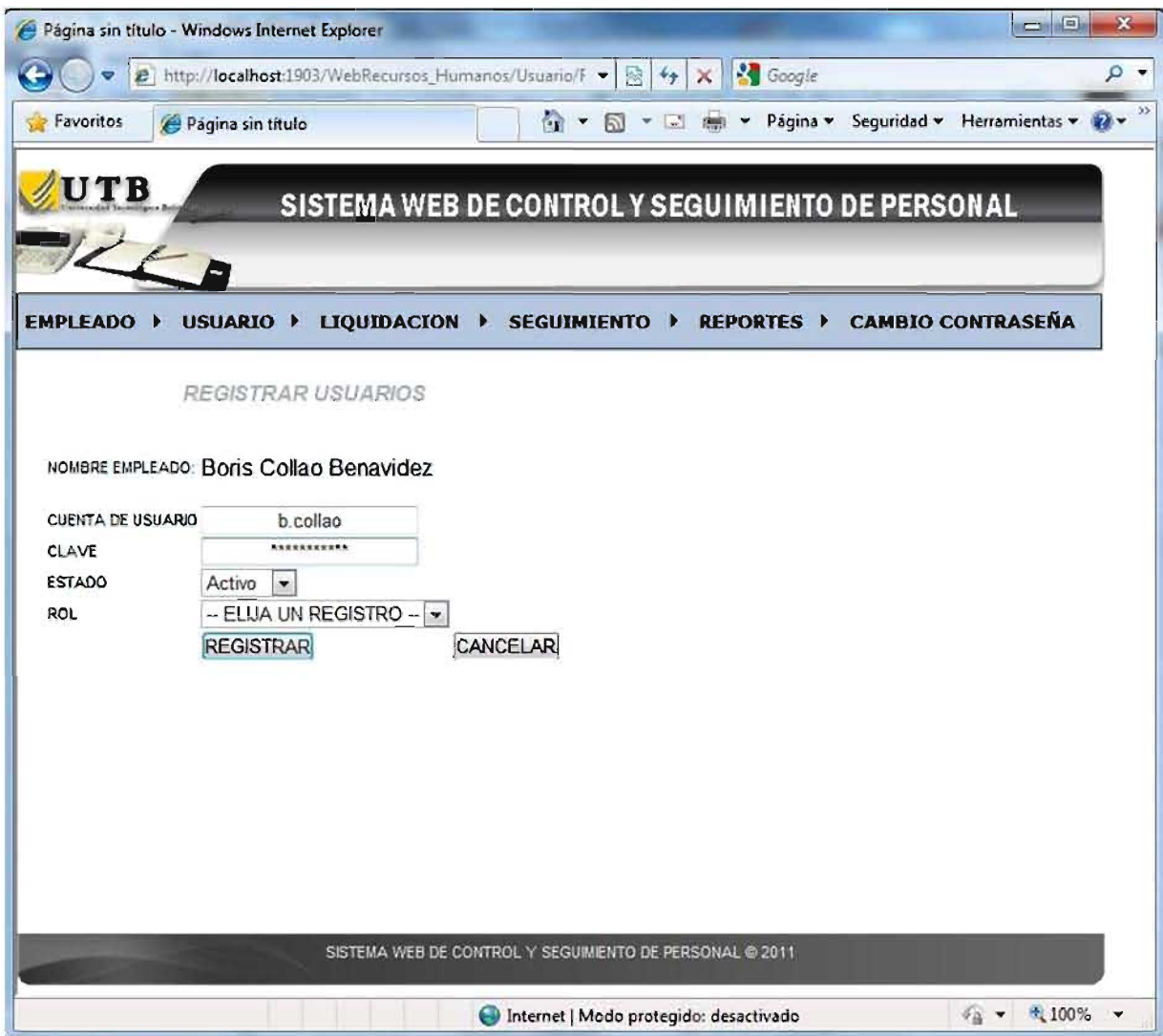


Figura 3.23: Registro de usuario 2
[Fuente: Elaboración Propia]

3.4.1.6 Liquidación

En el menú principal elegimos la opción de liquidación. Después de elegir la opción se busca a un empleado o se lo selecciona de la lista, se puede observar en la siguiente figura las mencionadas opciones.

En el listado se puede observar los siguientes datos: el nombre completo del empleado, id y finalmente la opción de liquidación para efectuar la operación.

Como se puede observar en la figura 3.24.



Figura 3.24: Liquidación
[Fuente: Elaboración Propia]

Después de haber clickeado en liquidación tenemos la siguiente figura con la operación de liquidación con todos los datos necesarios y los cálculos realizados.

Como se puede observar en la figura 3.25.

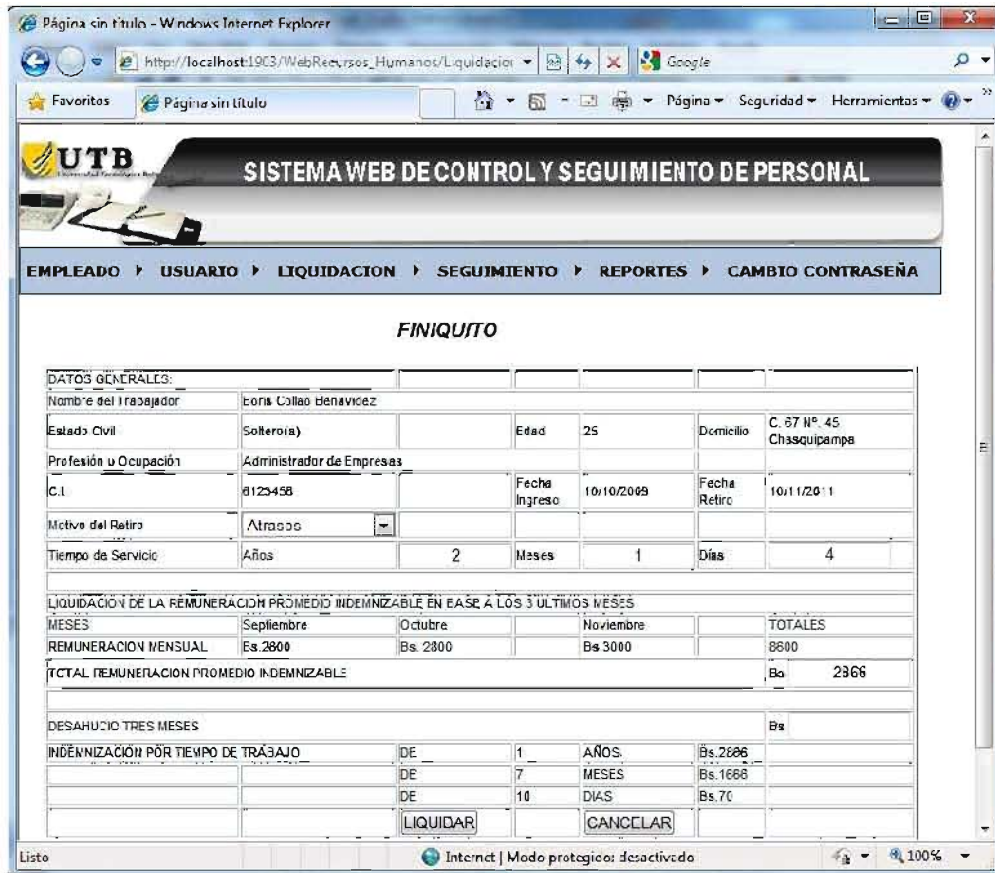


Figura 3.25: Liquidación 2
[Fuente: Elaboración Propia]

Finalmente la pantalla de confirmación de liquidación. Como se puede observar en la figura 3.26.



Figura 3.26: Confirmación de Liquidación 2
[Fuente: Elaboración Propia]

3.4.1.7 Seguimiento de personal

En el menú principal se elige la opción de seguimiento, en este caso se realizará el seguimiento de un empleado específico. Se tiene un listado para elegir al que se va a hacer el seguimiento y se presiona la opción de retraso.

Como se puede observar en la figura 3.27.

id Empleado	Codigo Empleado	Nombre	Paterno	Materno	Retrasos
26	SFM3265432	Marcelo	Salinas	Fernandez	▶
25	LCA3651243	Andrea	Lenz	Canedo	▶
24	ABC4263403	Carla	Antela	Benavides	▶
23	OPM5124326	Mariana	Ortega	Peñaloza	▶
22	PMV3652312	Vilma	Paredes	Mendez	▶
21	PFA3534231	Aide	Pereira	Flores	▶
20	CAJ4523412	Juan Jose	Chambi	Alarcon	▶
19	HJJ3451265	Jose	Hernandez	Jurado	▶

Figura 3.27: Seguimiento de personal
[Fuente: Elaboración Propia]

Después de haber cliqueado en la opción de retraso nos muestra la siguiente figura, en la que se escoge la gestión, el intervalo de fechas para realizar el seguimiento.

Como se puede observar en la figura 3.28.

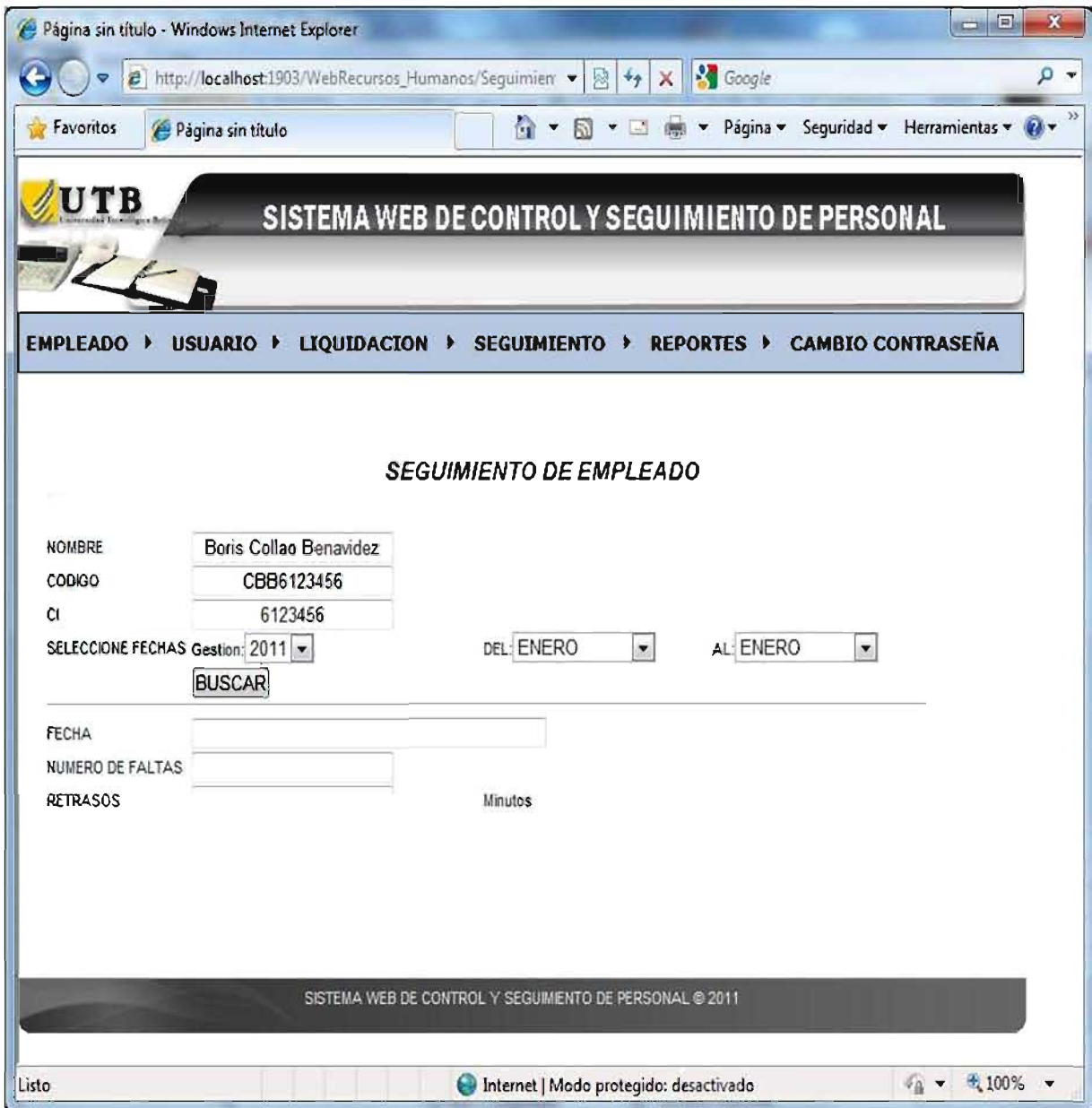


Figura 3.28: Seguimiento de personal 2
[Fuente: Elaboración Propia]

Después de clicar sobre buscar nos mostrará la fecha, la cantidad de faltas que tiene un empleado y los minutos de retraso que tiene el empleado. Además de poder observar todos sus datos para identificarlo.

Como se puede observar en la figura 3.29.

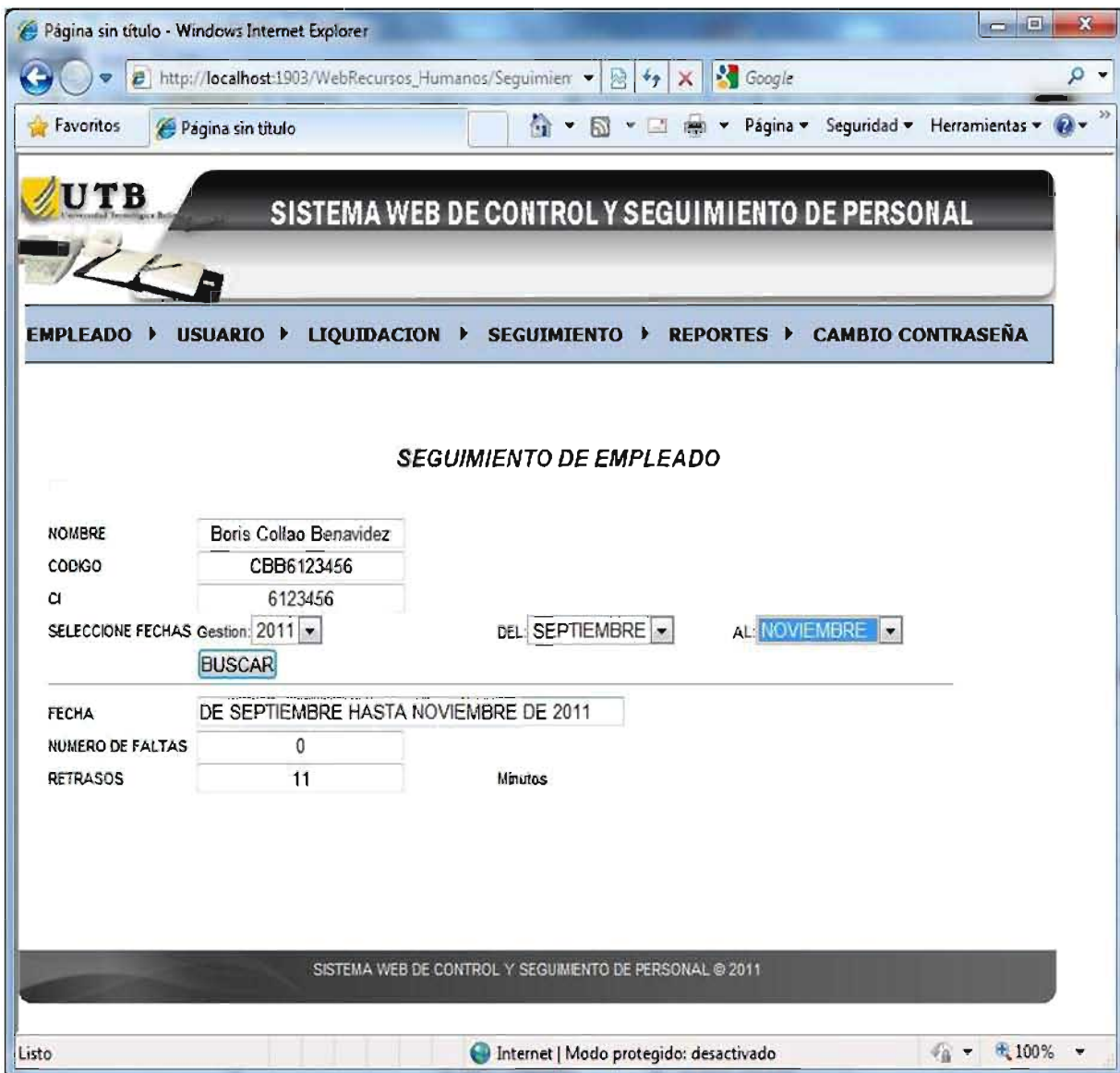


Figura 3.29: Resultado del Seguimiento de personal
[Fuente: Elaboración Propia]

3.4.1.8 Seguimiento de contrato

En esta parte también se ingresa al menú principal y escogemos la opción de seguimiento para realizar el control y seguimiento de contratos de los empleados. Es decir que empleados tienen contrato, ítem o son pasantes.

En este caso se mostrara los resultados gráficamente ya que se verán de todos los empleados en general.

Como se puede observar en la figura 3.30.

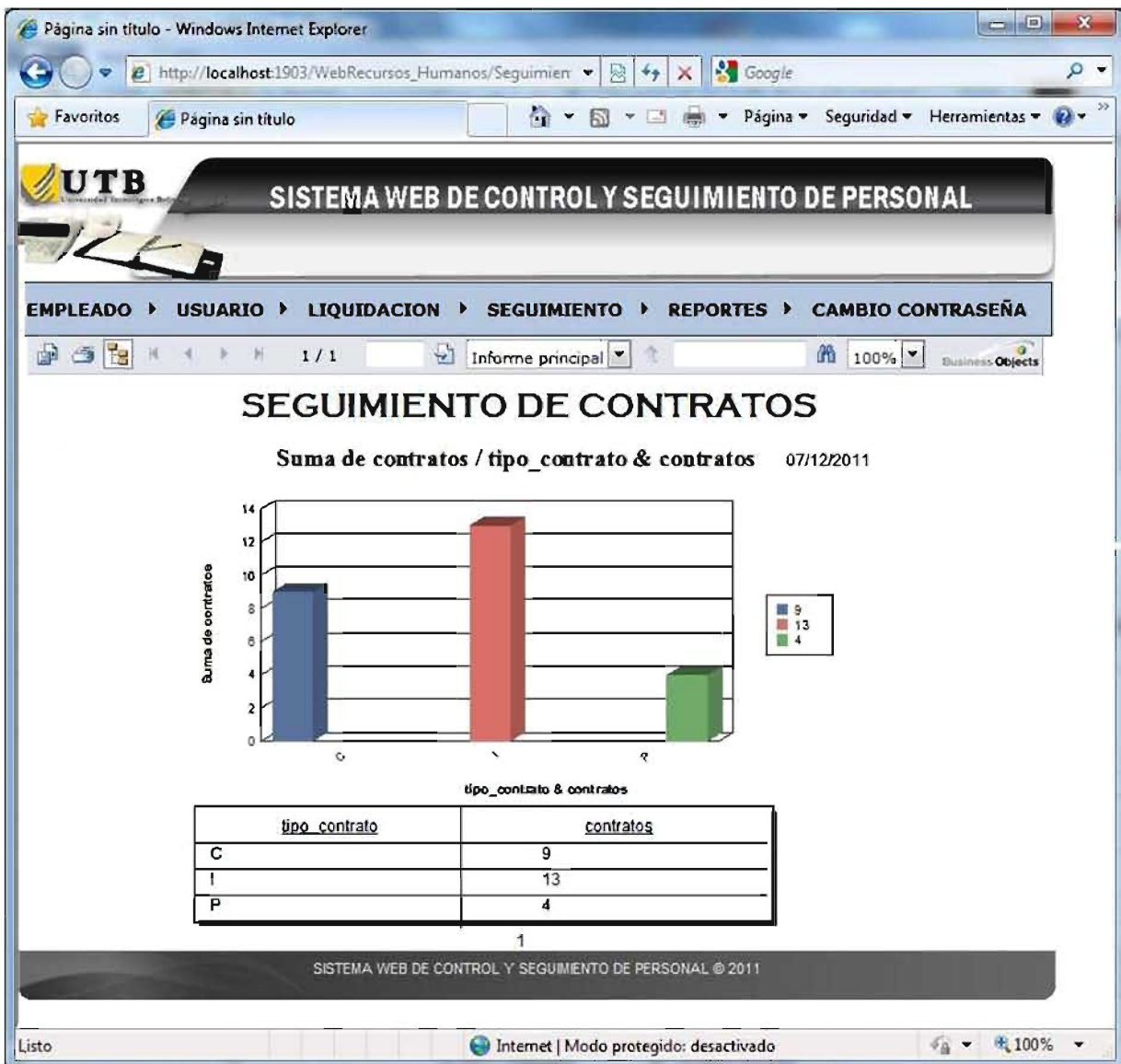


Figura 3.30: Resultado del Seguimiento de personal
[Fuente: Elaboración Propia]

3.5 Fase de transición

La fase de transición se enfoca en liberar el sistema a producción. Para finalizar esta fase se completa con la aprobación y visto bueno del diseño e implementación del sistema, por parte de los usuarios.

3.6 Políticas de implementación

Las especificaciones del diseño del sistema sirven como base para la construcción del nuevo sistema. El proceso de activar el nuevo sistema de información y retirar el viejo, el cambio

puede ser rápido o lento, dependiendo del método a usarse. Los cuatro métodos son: Corte y cambio o direct cutover, operación paralela o parallel operation, operación piloto o pilot operation y operación de fase o phased changeover.

3.6.1 Corte y cambio

El cambio del sistema viejo al nuevo ocurre tan pronto el nuevo sistema es operacional. Usualmente es el método menos costoso, puesto que se debe operar y mantener un sistema a la vez. Por otro lado, es el método con mayor riesgo, pues no se puede usar el sistema viejo para ayudar a resolver los problemas que ocurran, o para verificar que todos los detalles de los datos sean correctos. Este método se puede usar al implantar programas comprados, para sistemas no críticos o cuando los dos sistemas (el viejo y el nuevo) no pueden co-existir.

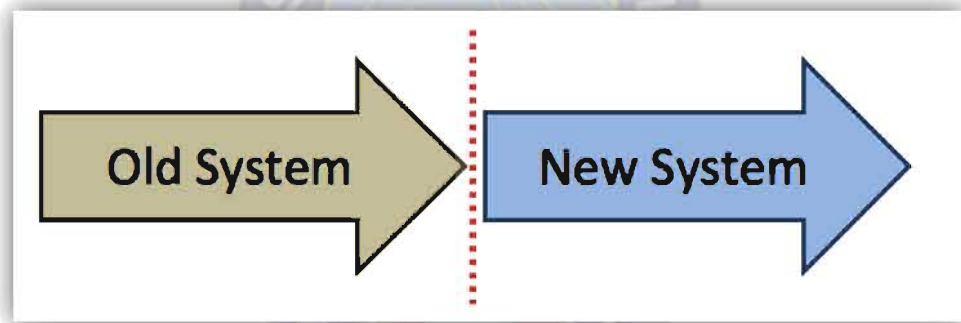


Figura 3.31: Corte y Cambio
[Fuente: Copley, S.]

3.6.2 Operación paralela

Ambos sistemas operan totalmente por un tiempo específico. Los datos son entrados a ambos sistemas, y el output (salida) generado por el sistema nuevo es comparado con su equivalente en el sistema viejo. Cuando el nuevo sistema opera correctamente, se deja de usar el sistema viejo. La ventaja mayor es su bajo riesgo, si el nuevo sistema no trabaja bien, se puede usar el sistema viejo hasta que se realicen los cambios apropiados. Sin embargo, este método es el más costoso, pues hay que darle servicios a ambos sistemas. Los usuarios deben trabajar en los dos sistemas y a veces hace falta ayuda temporaria. Este método no es práctico si los dos sistemas son incompatibles o realizan funciones diferentes.

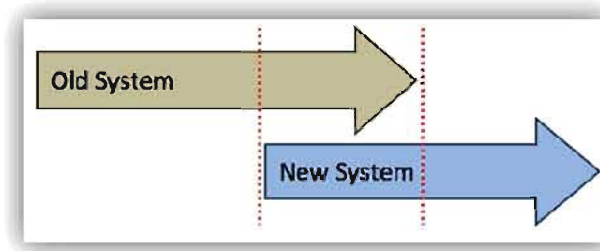


Figura 3.32: Operación Paralela
[Fuente: Copley, S.]

3.6.3 Operación piloto

Se implanta todo el nuevo sistema en un área de la compañía. El grupo que usa el nuevo sistema se conoce como área o grupo piloto. El sistema viejo continúa operando en toda la organización, incluyendo el área piloto. Después de probar que el sistema trabaja correctamente en el área piloto, se implanta en toda la organización, normalmente usando el método corte y cambio. La operación piloto reduce el riesgo de fallas del sistema comparado con corte y cambio; y es menos costoso que la operación paralela.

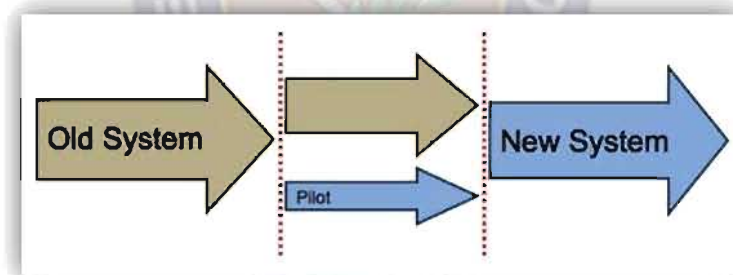


Figura 3.33: Operación piloto
[Fuente: Copley, S.]

3.6.4 Operación de fase

Se implanta el nuevo sistema en fases, o módulos, en vez de implantar todo el sistema a la vez. Cada subsistema se puede implantar usando uno de los otros métodos.

Una ventaja es que el riesgo se limita solo al módulo implantado. Este método no es posible si no se puede separar el sistema en módulos. No se debe confundir el método operación de fase con la operación piloto. En la operación de fase se da parte del sistema a todos los usuarios, mientras que la operación piloto provee el sistema completo, pero solo a un grupo de usuarios.

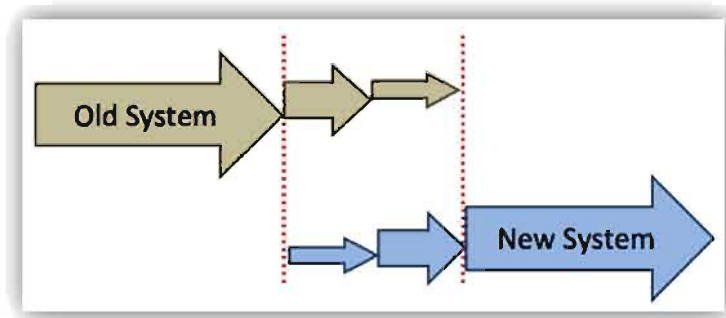


Figura 3.34: Operación de fase
[Fuente: Copley, S.]

3.6.5 Aplicación del método corte y cambio

Para el presente trabajo se aplicara el método corte cambio por las siguientes razones:


- Toma menos tiempo y esfuerzo
- El nuevo sistema está funcionando inmediatamente.



CAPITULO IV

CALIDAD DE SOFTWARE Y SEGURIDAD

La llave que se usa constantemente reluce como plata; no usándola se llena de herrumbre. Lo mismo pasa con el entendimiento. (Benjamín Franklin)



CALIDAD DE SOFTWARE Y SEGURIDAD

4.1 Introducción

La calidad del software, es el desarrollo de software basado en estándares con la funcionalidad y rendimiento total que satisfacen los requerimientos del cliente. A continuación se describen los factores de calidad con el objeto de evaluar la calidad del software.

A la conclusión del Sistema Web de Control y Seguimiento de Personal se debe medir la calidad del producto, hay dos tipos de medición: directa e indirecta. En el presente proyecto se aplicará las medidas indirectas planteadas por la norma ISO 9126

4.2 Características Propuestas por ISO-9126

La norma ISO – 9126 plantea un modelo normalizado que permite evaluar y comparar productos sobre la misma base.

Aquí la calidad se define por seis características:

Funcionalidad: En este grupo se conjunta una serie de atributos que permiten calificar si un producto de software maneja en forma adecuada el conjunto de funciones que satisfagan las necesidades para las cuales fue diseñado.

Confiabilidad: Capacidad del software de mantener su nivel de ejecución bajo condiciones normales en un periodo de tiempo establecido.

Usabilidad: Consiste de un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema.

Portabilidad: Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.

Mantenibilidad: Es el esfuerzo necesario para diagnosticar las deficiencias o causas de fallas, o para identificar las partes que deberán ser modificadas.

Eficiencia: Esta característica permite evaluar la relación entre el nivel de funcionamiento del software y la cantidad de recursos usados

4.2.1 Confiabilidad

La confiabilidad de un sistema es un elemento importante en su calidad general. Para determinar la confiabilidad se toma en cuenta las fallas que se producen en el sistema en un tiempo determinado, también es el grado en que el sistema responde bajo las condiciones definidas durante un intervalo de tiempo dado. Primeramente se considera la confiabilidad de cada módulo independientemente. Para ello se requiere el modelo de sistema, que se puede observar en la Figura 4.1.

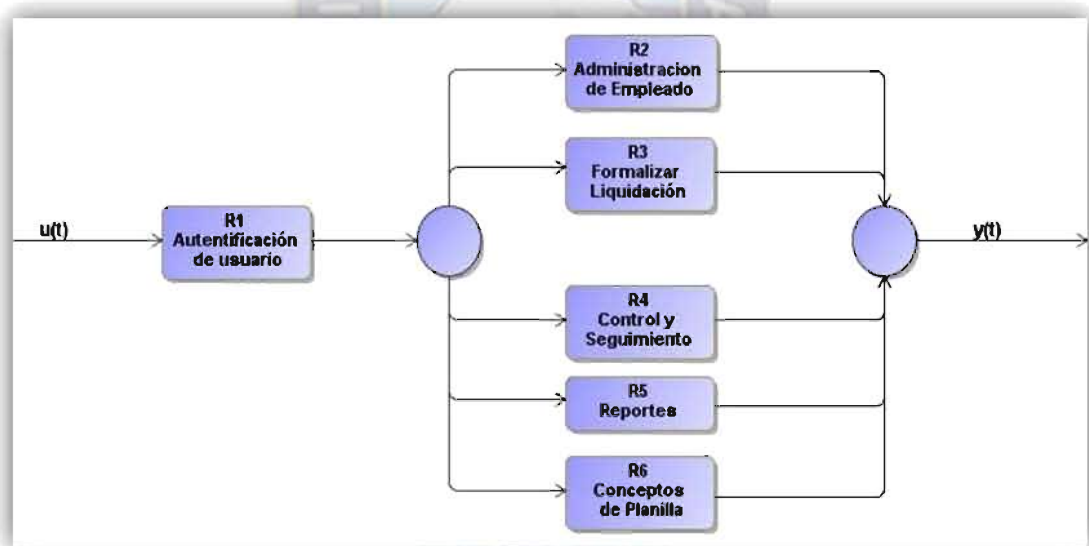


Figura 4.1: Modelo Del Sistema Web De Control Y Seguimiento De Personal
[Fuente: elaboración propia]

Tomando en cuenta la relación de: **(1)** $R(t) = e^{-\lambda t}$

Donde:

$R(t)$ = confiabilidad de un componente o subsistema t .

$e^{-\lambda t}$ = Probabilidad de falla de un componente o subsistema en el tiempo t.

T = Tiempo de trabajo sin falla.

λ = Tasa constante de fallos.

t = Periodo de operación de tiempo

Se realiza el cálculo de la confiabilidad de cada módulo del sistema con (1).

Tabla 4.1

	Módulo	λ	T	R(t)
1	Autenticación de usuario	0.02	2 Hrs.	0.96
2	Administración de empleado	0.03	3 Hrs.	0.91
3	Formalizar liquidación	0.04	3 Hrs.	0.89
4	Control y Seguimiento	0.03	3 Hrs.	0.91
5	Reportes	0.04	3 Hrs.	0.89
6	Conceptos de Planilla	0.03	3 Hrs.	0.92

Calculo de confiabilidad
[Fuente: Elaboración propia]

El modelo de sistema en la Figura 4.1, nos muestra una conexión compuesta donde se tiene inicialmente una conexión en serie y posteriormente una conexión en paralelo. Por lo tanto realizando los cálculos correspondientes tenemos:

$$(2) R = R_i * R_s$$

Donde: $R_i = R_1 = 0.96$

$$(3) R_s(t) = 1 - \{[1 - R_2(t)] * [1 - R_3(t)] * \dots * [1 - R_{n-1}(t)] * [1 - R_n(t)]\}$$

Reemplazando en (3) se tiene:

$$R_s(t) = 1 - [(0.09) * (0.11) * (0.09) * (0.11) * (0.08)]$$

$$R_s(t) = 0.9999922$$

Reemplazando en (2) se tiene

$$R = 0.96 * 0.9999922$$

$$\%R = 96\%$$

Con el resultado obtenido se puede decir que el “Sistema Web de Control y Seguimiento de Personal Caso: Universidad Tecnológica Boliviana”, presenta una confiabilidad de 96%, entonces se afirma que es un sistema confiable.

4.2.2 Funcionalidad

La funcionalidad no se mide directamente, por tanto es necesario evaluar un conjunto de características y capacidades del sistema. Nuestro sistema debe ser capaz de proveer las funciones que cumplen con las necesidades explícitas e implícitas cuando es utilizado en las condiciones especificadas por el cliente.

Para el cálculo de la funcionalidad utilizaremos la métrica de punto función, para esto se debe determinar cinco características de dominios de información y se proporciona las cuentas en la posición apropiada a la tabla. Los valores de los dominios de información se definen de la siguiente forma:

- Número de entradas de usuario.
- Número de salidas de usuario.
- Número de peticiones de usuario.
- Número de archivos.
- Número de interfaces externas.

Para calcular puntos función (PF), se utiliza la siguiente relación:

$$(1) PF = \text{cuenta total} * (0.65 + 0.01 * \sum F_i)$$

Dónde:

PF = Medida de funcionalidad.

Cuenta total = Es la suma de todas las entradas obtenidas en: N° de entradas, N° de salidas, N° de peticiones, N° de archivos y N° de interfaces externas.

$\sum F_i$ = Son los valores de ajuste de complejidad según las respuestas a preguntas destacadas en la siguiente tabla.

Tomando en cuenta estos pasos ahora encontraremos el punto función.

4.2.2.1 Número de entradas de usuarios

Es la información que llega desde el exterior, tiene una sola dirección del exterior al interior. Se puede observar en la Tabla 4.2, las entradas de usuario que tiene el sistema.

Tabla 4.2

N°	Entradas de Usuario
1	Pantalla de ingreso al sistema
2	Registro de empleados
3	Registro de contrato
4	Registro de sueldos
5	Registro de Entradas
6	Registro de descuentos
7	Registro de permisos
8	Registro de memorándums
9	Registro de puesto de trabajo
10	Registro de sucursales
11	Registro de liquidación

Entradas de Usuario
[Fuente: Elaboración propia]

4.2.2.2 Número de salidas de usuario

Es la información elaborada por el sistema que son transmitidas al usuario, también actualizan algunos archivos, tiene una sola dirección del interior al exterior. Se puede observar en la Tabla 4.3 las salidas de usuario.

Tabla 4.3

N°	Salida de Usuario
1	Detalle del empleado
2	Detalle de contrato
3	Detalle de Memorándum
4	Detalle de permisos
5	Detalle de historial laboral
6	Boleta de pago
7	Planilla mensual
8	Reporte operativo
9	Detalle asignación de cargo
10	Detalle asignación de sucursal
11	Reporte de empleado por sucursales
12	Reporte de recibo de adelanto
13	Reporte liquidación

Salidas de Usuario

[Fuente: Elaboración propia]

4.2.2.3 Número de peticiones de usuario

Es una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se puede observar las peticiones de usuario en la Figura 4.4.

Tabla 4.4

N°	Peticiones de Usuario
1	Autenticación usuario

2	Modificación de usuario
3	Modificación de empleado
4	Modificación de Salarios
5	Listado de empleados
6	Listado de sucursales
7	Listado de liquidaciones
8	Listado de ingresos
9	Listado de descuentos
10	Listado de planilla de sueldo

Peticiones de Usuario

[Fuente: Elaboración propia]

4.2.2.4 Número de archivos

Es cada archivo maestro lógico, es decir es un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente. Se puede observar el número de archivos en la Tabla 4.5.

Tabla 4.5

N°	Archivos
1	Empleado
2	Experiencia laboral
3	Cargo
4	Contrato
5	Departamento
6	Memorándum
7	Liquidación
8	Ingreso
9	Descuento
10	Sucursal
11	Centro de estudio

Calculo de archivos

[Fuente: Elaboración propia]

4.2.2.5 Número de interfaces externas

Son todas las interfaces legibles por la máquina que se utilizan para transmitir información a otro sistema. Una vez que se han recopilado los datos anteriores, a la cuenta se asocia un valor de complejidad. Se puede observar las interfaces externas en la Figura 4.6.

Tabla 4.6

N°	Interfaces externas
1	Internet
2	Intranet

Interfaces externas
[Fuente: Elaboración propia]

4.2.2.6 Ponderación

Ahora reuniremos todos los datos encontrados, para realizar una ponderación. Para utilizar puntos de función es necesario elegir un criterio de ponderación, que en este caso utilizaremos el factor medio. Como se puede observar en la Tabla 4.7.

Tabla 4.7

Parámetro de medición	Cuenta	*	Factores de ponderación				Total
			Simple	Medio	Complejo	=	
N° de entradas de usuario	11	*	3	4	6	=	44
N° de salidas de usuario	13	*	4	5	7	=	65
N° de peticiones de usuario	10	*	3	4	6	=	40
N° de archivos	11	*	7	10	15	=	110
N° de interfaces externas	2	*	5	7	10	=	14
Cuenta total							273

Factores de ponderación
[Fuente: Elaboración propia]

La cuenta total de los puntos de función se debe ajustar en función a las características ambientales del sistema, en la Tabla 4.8 se obtienen los valores de ajustes de la complejidad, según las respuestas a las siguientes preguntas, evaluados entre 0 y 5.

Tabla 4.8

N°	Factores de complejidad	Sin Influencia	Incidental	Moderado	Medio	Significativo	Esencial	F _i
		0	1	2	3	4	5	
1	Requiere el sistema copias de seguridad y de recuperación fiables.						X	5
2	Se requiere comunicación de datos.						X	5
3	Existen funciones de procesamiento distribuido.			X				2
4	Es crítico el rendimiento.			X				2
5	Se ejecutará el sistema con un entorno operativo existente y fuertemente utilizado.					X		4
6	Requiere el sistema entrada de datos interactiva.				X			3
7	Facilidad operativa.					X		4
8	Se actualiza los archivos maestros de forma interactiva.			X				2
9	Son complejas las entradas, las salidas, los archivos o las peticiones.				X			3
10	Procesamiento interno complejo.					X		4
11	Diseño del código reutilizable.				X			3
12	Facilidad de instalación						X	5
13	Soporta múltiples instalaciones en diferentes sitios						X	5
14	Facilidad de cambios					X		4
Factor de ajuste de complejidad								51

Valores de ajustes de complejidad
[Fuente: Elaboración propia]

Reemplazando los datos en la fórmula de punto función, es decir en (1) se tiene:

$$PF = 273 * (0.65 + 0.01 * 51)$$

El punto función del sistema de Control y Seguimiento de Personal Caso: UTB es:

$$PF = 316.68$$

Luego, comparando los valores de funcionalidad del sistema con el Punto Función máximo que se puede alcanzar es:

$$PF = 273 * (0.65 + 0.01 * 70)$$

$$PF = 368.55$$

Por tanto la funcionalidad del Sistema Web de Control y Seguimiento de Personal Caso: UTB será:

$$\text{Funcionalidad} = \left(\frac{316.68}{368.55} \right) * 100$$

$$\text{Funcionalidad} = 85.92\%$$

4.2.3 Mantenibilidad

Para calcular la Mantenibilidad del sistema utilizaremos las medidas directas proporcionadas por la IEEE 982.1 – 1998 el cual sugiere un índice de madurez del sistema, que consiste en los cambios que producen en cada versión del producto para lo cual se tiene la siguiente relación.

$$(1)IMS = [M_T - (F_a + F_b + F_c)]/M_T$$

Dónde:

M_T = Número de módulos en la versión actual.

F_a = Numero de módulos en la versión actual que se han cambiado.

F_b = Número de módulos en la versión actual que han añadido.

F_c = Número de módulos en la versión anterior que se han borrado en la versión actual.

Si el valor del IMS se aproxima a 1, el sistema empieza a estabilizarse.

Entonces los valores encontrados son:

$$M_T = 6$$

$$F_a = 1$$

$$F_b = 0$$

$$F_c = 0$$

Reemplazando estos valores en (1):

$$IMS = [6 - (1 + 0 + 0)]/6$$

$$IMS = 0.83 * 100$$

$$IMS = 83\%$$

Por tanto puede indicarse que el sistema tiende a estabilizarse en un 83% y el 17% indica que aún no es estable debido a los cambios

4.2.4 Portabilidad

Es la capacidad del software para ser trasferido de un ambiente de operaciones a otro. La portabilidad del software se enfoca en tres aspectos: a nivel de aplicaciones, a nivel de sistema operativo y a nivel de hardware.

A demás de ver estos tres aspectos, también podemos considerar la facilidad de instalación, ajuste y adaptación al cambio. Dado por la siguiente formula:

$$(1) GP = 1 - [CT/CRD]$$

Dónde:

GP = Grado de portabilidad.

CT = Costo de Transportar.

CRD = Costo de Re- Desarrollo.

Si $GP > 0$, la portabilidad es más rentable que el re-desarrollo.

Si $GP = 1$, la portabilidad es perfecta.

Si $GP < 0$, el re-desarrollo es más rentable que portabilidad

Reemplazando en **(1)** tenemos:

$$GP = 1 - [75/2000]$$

$$GP = 0.9625$$

Por lo tanto, el sistema puede transportarse de un entorno a otro en un 96% más rentable que el re-desarrollo.

4.2.4.1 Nivel de aplicaciones

El software es desarrollado en el Framework Visual Studio .Net 2008 con su gestor de base de datos SQL Server 2005, en cuanto a la aplicación es portable ya que el sistema puede ser distribuido en CDs.

4.2.4.2 Nivel de sistema operativo

Para los sistemas operativos de Windows 2000, XP, Vista, 7 es portable nuestro sistema.

4.2.4.3 Nivel de hardware

A nivel de hardware es portable para todas las computadoras mayores o iguales a la tecnología Pentium IV.

4.2.5 Usabilidad

Grado en el que el software es fácil de usar, y viene reflejado por la: facilidad de comprensión, facilidad de aprendizaje y operatividad. Para medir la usabilidad del sistema se usaran las siguientes tres métricas.

4.2.5.1 La complejidad de la descripción

Dada por la fórmula: (1) $X = A/B$

Dónde:

A = Número de funciones (casos de uso) o tipos de funciones descritas en la descripción del producto.

B = Número total de funciones (casos de uso).

Reemplazando en (1) tenemos:

$$X = 8/10$$

$$X = 0.8$$

Por lo tanto, el sistema presenta un 80% de entendimiento por parte de los usuarios finales respecto a la capacidad del producto.

4.2.5.2 Consistencia operacional

Dada por la fórmula: (2) $X = 1 - A/B$

Donde:

A = Número de instancias de operaciones con comportamiento inconsistente.

B = Número total de operaciones.

Reemplazando en (2) tenemos:

$$X = 1 - 3/11$$

$$X = 0.72$$

Por lo tanto, el sistema presenta un 72% de no instancias de operaciones con comportamiento inconsistente.

4.2.5.3 Consistencia operacional en el uso

Dada por la fórmula: $(3) X = 1 - A/B$

Donde:

A = Número de funciones que el usuario encontró inaceptablemente inconsistentes según son expectativas.

B = Número de funciones usadas por el usuario durante el periodo de prueba.

Reemplazando en (3) tenemos:

$$X = 1 - 2/18$$

$$X = 0.89$$

Por lo tanto el usuario encuentra un 10% del sistema inaceptable en el periodo de prueba. De acuerdo a los resultados obtenidos anteriormente el usuario se encuentra satisfecho con la consistencia operacional del uso del sistema.

4.3 Seguridad de Software

Los problemas de seguridad de un sistema web pueden venir de la configuración de las herramientas que se utilizaron para su desarrollo o pueden ser producto de una falla en el diseño lógico, a menudo es la segunda falla la que ocasiona problemas en el funcionamiento del sistema.

4.3.1 Amenazas

Existen diversas amenazas, a continuación se mencionan las más comunes:

- Ingreso de usuario no válido.
- Control de acceso roto.
- Administración de sesión y autenticación rota.
- Desbordamiento de buffer.

- Inyección de código.
- Manejo de errores inadecuado.
- Almacenamiento inseguro.
- Administración de configuración insegura.

4.3.2 Guías de seguridad

Estos son algunos principios de seguridad para el diseño de aplicaciones web.

- Validar todas las entradas y salidas.
- Mantener un esquema de seguridad simple.
- Manejar las fallas y errores de forma adecuada.
- Utilizar sólo componentes de confianza.
- Controlar las excepciones.

4.3.3 Tipos de seguridad para sistemas web

Hay cuatro tipos de seguridad en los sistemas web:

- Seguridad en el cliente.
- Seguridad en el servidor.
- Seguridad en las comunicaciones.
- Seguridad en la aplicación.

4.3.3.1 Seguridad en el cliente

Uno de los mecanismos de seguridad que se implementan son las validaciones por el lado del cliente.

Existen mecanismos de validación provistas por las herramientas que utilizamos para hacer la aplicación, en el caso de asp.net se tienen los controles de validación para la información introducida por el cliente, estas validaciones son realizadas antes de que la información introducida llegue al servidor, esto evita que se envíen datos incorrectos al servidor, además se ahorra tiempo, ya que si la información es incorrecta simplemente no se envía al servidor.

4.3.3.2 Seguridad en el servidor

La validación del lado del cliente no es suficiente, también deben realizarse otro tipo de controles por el lado del servidor, ya sea del servidor de aplicaciones o del servidor de base de datos.

- Seguridad en el servidor de aplicaciones.

Un servidor de aplicaciones proporciona muchos servicios y no todos son necesarios para el funcionamiento de la aplicación web.

Es conveniente deshabilitar lo que no se necesite en el servidor de aplicaciones, para ello debe configurarse adecuadamente el servidor de aplicaciones.

- Seguridad en el servidor de base de datos.

Existen muchos problemas a nivel de base de datos, uno de ellos es la Inyección SQL (lenguaje de consulta estructurada), son los ataques realizados contra bases de datos. En este caso, un usuario utiliza debilidades en el diseño de la base de datos o de la página web para extraer información o más aún, para manipular información dentro de la base de datos. Una forma de evitar este tipo de debilidades es restringiendo los caracteres que el usuarios introduce, por ejemplo las comillas, los punto y coma y otros.

También está el acceso no autorizado a la información. Esto podría solucionarse asignando correctamente los roles que tiene cada usuario en el sistema. El acceso de los usuarios a los formularios web es cargado desde la base de datos, cuando un usuario se autentica el sistema le da acceso sólo a los formularios asignados para él.

4.3.3.3 Seguridad en la comunicación

SSL es un protocolo diseñado y propuesto por Netscape Communications Corporation. Se encuentra en la pila OSI entre los niveles de TCP/IP y de los protocolos HTTP, FTP, SMTP, etc. Proporciona sus servicios de seguridad cifrando los datos intercambiados entre el servidor y el cliente con un algoritmo de cifrado simétrico, típicamente el RC4 o IDEA, y cifrando la clave de sesión de RC4 o IDEA mediante un algoritmo de cifrado de clave pública, típicamente el RSA. La clave de sesión es la que se utiliza para cifrar los datos que vienen del y van al servidor seguro. Se genera una clave de sesión distinta para cada transacción, lo cual permite que aunque sea reventada por un atacante en una transacción dada, no sirva para descifrar futuras transacciones. MD5 se usa como algoritmo de hash.

Proporciona cifrado de datos, autenticación de servidores, integridad de mensajes y, opcionalmente, autenticación de cliente para conexiones TCP/IP.

Cuando el cliente pide al servidor seguro una comunicación segura, el servidor abre un puerto cifrado, gestionado por un software llamado Protocolo SSL Record, situado encima de TCP. Será el software de alto nivel, Protocolo SSL Handshake, quien utilice el Protocolo SSL Record y el puerto abierto para comunicarse de forma segura con el cliente.

4.3.3.4 Seguridad en la aplicación

El control de acceso de los usuarios es una parte fundamental para una aplicación web.

- Autenticación, determina si un usuario es quien dice ser.

Autenticación HTTP básica, cuando se quiere ingresar a un formulario protegido el servidor devuelve un código “HTTP/1.1 401 Authorization required”. El cliente debe enviar sus datos al servidor.

Esta autenticación es fácil de implementar, pero los datos viajan por la red sin encriptar, no se puede “cerrar sesión”, la única forma es cerrar el navegador.

- Autenticación basada en la aplicación, en este caso la aplicación implementa su propio mecanismo de autenticación. Es más costosa pero es más flexible porque permite establecer diferentes permisos y niveles de acceso asignados al usuario.
- Passwords, se recomienda restringir los valores para los nombre de los usuarios. Almacenar los passwords de forma segura protegiendo el acceso a la base de datos. Bloquear una cuenta cuando se detecta un número determinado de intentos de acceso incorrectos. Tener una política de recuperación de passwords en caso de olvido por parte del usuario.
- Sesiones, después de que el usuario se ha autenticado se debe mantener esta autenticación en cada conexión subsiguiente. Para esto se utilizan las variables de sesión, que permiten mantener el estado entre las diferentes peticiones HTTP. El procedimiento es el siguiente: Después de autenticarse el usuario recibe un identificador de sesión, este identificador es invisible y lo acompañará en cada petición. Este identificador se almacena en la máquina del cliente, mediante una cookie.
- La gestión de sesiones es responsabilidad del programador. Un sistema de gestión de sesiones debe: Establecer un tiempo límite de vida para la sesión, pedir una nueva autenticación cuando realizase una operación importante, proteger los identificadores de sesión durante su transición y destruir la cookie cuando finalice la sesión para evitar el acceso de otro usuario.



CAPITULO V

ANÁLISIS DE COSTO/BENEFICIO

Una colección de pensamientos debe ser una farmacia donde se encuentra remedio a todos los males. (Voltaire)



ANÁLISIS DE COSTO/BENEFICIO

5.1 Introducción

Son una parte importante de todo análisis de sistemas. El propósito, es mostrar a los usuarios del nuevo sistema, al igual que a otros grupos de administradores de la organización, que los beneficios que se espera obtener con el nuevo sistema superan a los costos esperados.

En las secciones siguientes examinaremos diversos aspectos de los cálculos de costo/beneficio:

- Análisis de costos.
- Análisis de beneficios.
- Cómo expresar los ahorros.
- Análisis de riesgo.

5.2 Análisis de costos

Se deben calcular todos los costos anticipados asociados con el sistema. Para determinar el costo total del proyecto se tomará en cuenta los siguientes costos:

- Costo del Software desarrollado.
- Costo de la implementación del sistema.
- Costo de la elaboración del proyecto.

5.2.1 Costo del software desarrollado

Para determinar el costo del software desarrollado, se utilizará el modelo constructivo COCOMO II, orientado a los puntos de función.

Estimación de puntos de función:

Tabla 5.1

Parámetro de medición	Cuenta	Factor de ponderación	Total
N° de entradas de usuario	11	4	44
N° de salidas de usuario	13	5	65
N° de peticiones de usuario	10	4	40
N° de archivos	11	10	110
N° de interfaces externas	2	7	14
Cuenta total			273

Calculo de punto función no ajustado
[Fuente: Elaboración propia]

Calculo de valores de ajuste de la complejidad tomando los valores de la Tabla 5.2, en la cual se determina la complejidad.

$$\text{Factor de Ajuste} = (0.65 + 0.01 * 51)$$

$$\text{Factor de Ajuste} = 1.16$$

Tabla 5.2

Factores de complejidad	Valor
Requiere el sistema copias de seguridad y de recuperación fiables.	5
Se requiere comunicación de datos.	5
Existen funciones de procesamiento distribuido.	2
Es crítico el rendimiento.	2
Se ejecutará el sistema con un entorno operativo existente y fuertemente utilizado.	4
Requiere el sistema entrada de datos interactiva.	3
Facilidad operativa.	4
Se actualiza los archivos maestros de forma interactiva.	2
Son complejas las entradas, las salidas, los archivos o las peticiones.	3
Procesamiento interno complejo.	4

Diseño del código reutilizable.	3
Facilidad de instalación	5
Soporta múltiples instalaciones en diferentes sitios	5
Facilidad de cambios	4
Total	51

Calculo de punto función ajustada
[Fuente: Elaboración propia]

El cálculo de los puntos de función se basa en la fórmula:

$PF = \text{Cuenta total} * \text{Factor de Ajuste}$

$PF = 273 * 1.16$

$PF = 316.68$

Conversión de los puntos de función a KDLC.

Ahora convertiremos los PF a miles de líneas de código. Para ello veremos la Tabla 5.3.

Tabla 5.3

Lenguaje	Nivel	Factor LDC / PF
C	2.5	128
Ansi Basic	5	64
Java	6	53
Ansi Cobol	3	107
Visual Basic	7	46
ASP	9	36
PHP	11	29
Visual C++	9.5	34

Conversión de puntos Función a KLDC
[Fuente: Elaboración propia]

$LDC = PF * \text{Factor LDC} / PF$

$LDC = 316.68 * 36$

$LDC = 11400,48$

$$KLDC=11.40$$

Aplicando las fórmulas básicas del esfuerzo, tiempo calendario y personal requerido.

Las ecuaciones del COCOMO tienen la siguiente forma:

$$E = a_b (KLDC) b_b$$

$$D = c_b (E) d_b$$

Donde:

E: Esfuerzo aplicado en personas por mes.

D: Tiempo de desarrollo en meses.

KLDC: Número estimado de líneas de código distribuidas (en miles)

Tabla 5.4

Proyecto de software	a_b	b_b	c_b	d_b
Orgánico	2.4	1.05	2.5	0.38
Semi-acoplado	3	1.12	2.5	0.35
Empotrado	3.6	1.2	2.5	0.32

Coefficientes a_b y b_b

[Fuente: Elaboración propia]

En la Tabla 5.4 se muestra los tipos de proyecto de software. Como este es un proyecto intermedio, en tamaño y complejidad, se elige Semi-acoplado.

$$E = 3 * (11.40)^{1.12}$$

$$E = 45.79$$

$$D = 2.5 * (45.79)^{0.35}$$

$$D = 9.53$$

El personal requerido se obtiene con la siguiente fórmula:

$$\text{Numero Programadores} = E/D$$

Numero Programadores=45.79/9.53

Numero Programadores=4.80=5

EL salario de un programador aproximadamente es de 300 \$us., cifra que se tomará en cuenta para la estimación siguiente:

Costo Software= Numero Programadores * Salario Programador

Costo Software= 5*300

Costo Software= 1500 \$us

5.2.2 Costo de la implementación del proyecto

Como la Universidad Tecnológica Boliviana ya cuenta con sus servidores de aplicaciones y base de datos completamente instalados y configurados, por lo que el costo de implementación es cero para el sistema.

5.2.3 Costo de elaboración del proyecto

Se refieren a los costos de estudio del sistema, en la etapa de análisis estos costos se representan en la siguiente tabla:

Tabla 5.5

Descripción	Costo total (\$us)
Análisis y diseño del proyecto	250
Bibliografía	40
Material de escritorio	30
Otros	30
Total	350

Costo de elaboración del proyecto
[Fuente: Elaboración propia]

5.2.4 Costo total

El costo total es la suma del costo de software de desarrollo y el costo de elaboración del proyecto, que se puede observar en la siguiente tabla:

Tabla 5.6

Descripción	Costo total (\$us)
Costo de software de desarrollo	1500
Costo de implementación	0
Costo de elaboración del proyecto	350
Total	1850

Costo total del proyecto
[Fuente: Elaboración propia]

5.3 Análisis de beneficios

Es mucho más difícil calcular los beneficios de un nuevo sistema de información que calcular su costo. La labor más grande al llevar a cabo un cálculo de costo-beneficio será acorralar a los usuarios y hacer que identifiquen beneficios tangibles que pueden medirse y calcularse de manera cuantitativa.

Los beneficios para el presente proyecto son de tipo intangible. Para realizar el análisis de beneficios se utilizarán cinco criterios de evaluación. Los cuales permiten evaluar el sistema diseñado respecto al sistema anterior estos son:

- Incremento de velocidad en los procesos.
- Capacidad en el volumen de información.
- Control de procesos.
- Integración de la información.
- Información para la toma de decisiones.

5.4 Valor neto actual

El Valor neto Actual (VAN) es un indicador financiero que mide los flujos de los futuros ingresos y egresos que tendrá un proyecto, para determinar, si luego de descontar la inversión inicial, nos quedaría alguna ganancia. Si el resultado es positivo, el proyecto es viable.

Basta con hallar VAN de un proyecto de inversión para saber si dicho proyecto es viable o no. El VAN también nos permite determinar cuál proyecto es el más rentable entre varias opciones

de inversión. Incluso, si alguien nos ofrece comprar nuestro negocio, con este indicador podemos determinar si el precio ofrecido está por encima o por debajo de lo que ganaríamos de no venderlo.

La fórmula del VAN es:

$$\text{VAN} = \text{BNA} - \text{Inversión}$$

Donde el beneficio neto actualizado (BNA) es el valor actual del flujo de caja o beneficio neto proyectado, el cual ha sido actualizado a través de una tasa de descuento.

La tasa de descuento (TD) con la que se descuenta el flujo neto proyectado, es el la tasa de oportunidad, rendimiento o rentabilidad mínima, que se espera ganar; por lo tanto, cuando la inversión resulta mayor que el BNA (VAN negativo o menor que 0) es porque no se ha satisfecho dicha tasa. Cuando el BNA es igual a la inversión (VAN igual a 0) es porque se ha cumplido con dicha tasa. Y cuando el BNA es mayor que la inversión es porque se ha cumplido con dicha tasa y además, se ha generado una ganancia o beneficio adicional.

$\text{VAN} > 0 \rightarrow$ el proyecto es rentable.

$\text{VAN} = 0 \rightarrow$ el proyecto es rentable también, porque ya está incorporado ganancia de la TD.

$\text{VAN} < 0 \rightarrow$ el proyecto no es rentable.

Entonces para hallar el VAN se necesitan:

- Tamaño de la inversión.
- Flujo de caja neto proyectado.
- Tasa de descuento.

Veamos un ejemplo:

El proyecto tiene una inversión de 1200 \$us. y una tasa de descuento (TD) de 10%, por defecto.

Hallando el VAN:

$$\text{VAN} = \text{BNA} - \text{Inversión}$$

$$\text{VAN} = 1450 / (1 + 0.14)^1 + 1450 / (1 + 0.14)^2 + 1450 / (1 + 0.14)^3 + 1450 / (1 + 0.14)^4 + 1450 / (1 + 0.14)^5 - 1450$$

$$\text{VAN} = 1425.69 - 1450$$

$$\text{VAN} = 151.69$$

5.5 Tasa interna de retorno

La TIR es la tasa de descuento (TD) de un proyecto de inversión que permite que el BNA sea igual a la inversión (VAN igual a 0). La TIR es la máxima TD que puede tener un proyecto para que sea rentable, pues una mayor tasa ocasionaría que el BNA sea menor que la inversión (VAN menor que 0).

Entonces para hallar la TIR se necesitan:

- Tamaño de inversión.
- Flujo de caja neto proyectado.

Veamos un ejemplo:

El proyecto tiene una inversión de 1200.

Para hallar la TIR hacemos uso de la fórmula del VAN, sólo que en vez de hallar el VAN (el cual reemplazamos por 0), estaríamos hallando la tasa de descuento:

$$\text{VAN} = \text{BNA} - \text{Inversión}$$

$$0 = 1200 / (1 + i)^1 + 1200 / (1 + i)^2 + 1200 / (1 + i)^3 + 1200 / (1 + i)^4 + 1200 / (1 + i)^5 - 1200$$

$$i = 10\%$$

$$\text{TIR} = 10\%$$

Si esta tasa fuera mayor, el proyecto empezaría a no ser rentable, pues el BNA empezaría a ser menor que la inversión. Y si la tasa fuera menor (como en el caso del ejemplo del VAN donde la tasa es de 10%), a menor tasa, el proyecto sería cada vez más rentable, pues el BNA sería cada vez mayor que la inversión.





CAPITULO VI

CONCLUSIONES Y RECOMENDACIONES

*Considero más valiente al que conquista sus deseos que al que conquista a sus enemigos, ya que la victoria más dura es la victoria sobre uno mismo.
(Aristóteles)*

CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

A la culminación del presente proyecto y conforme a las actividades definidas para el análisis e implementación del Sistema Web de Control y Seguimiento de Personal caso: Universidad Tecnológica Boliviana se concluye que:

- El producto obtenido cuenta con todas las características requeridas por los usuarios, resultando una herramienta de ayuda para los procesos que se efectúan en la Dirección Administrativa de la UTB.
- Con la implementación del sistema, no se pretende reemplazar a los empleados a cargo del proceso de Control y Seguimiento de personal, más aun el sistema desarrollado será un apoyo para el desempeño de sus funciones.
- Se logró centralizar y almacenar la información laboral, académica necesaria de todos los empleados para tener acceso inmediato.
- Las liquidaciones se realizan automáticamente en el sistema, permitiendo ahorro de tiempo al momento de hacer una liquidación a un empleado, además se genera un informe de liquidación inmediatamente.
- El sistema lleva un control de todos los movimientos y cambios de usuario, así de esta manera poder hacer un seguimiento de todos los procesos y lo más importante tener la información de quien es el responsable de dichos cambios.
- Con el módulo de control y seguimiento del sistema, se obtendrá la información de los empleados que se encuentran trabajando, o el estado del empleado, que nos indica si esta con permiso, vacación o retirado, también se realiza un control de la fecha de vencimiento de contratos de empleado, a la vez de poder tener dicha información de forma rápida y precisa.

6.2 Recomendaciones

Para ampliar el presente proyecto de grado, se hacen las siguientes recomendaciones:

- Se deberá incorporar normas y políticas de uso del sistema.
- Se deberá cambiar la contraseña semanalmente o mensualmente para dar mayor seguridad al sistema.
- Se recomienda integrar el Sistema Web de Control y Seguimiento de Personal con el Sistema de contabilidad para llevar un mejor control de ambas direcciones ya que están estrechamente relacionados.





*Estudia el pasado si quieres pronosticar
el futuro. (Confucio)*



BIBLIOGRAFIA

BIBLIOGRAFÍA

TEXTOS

- [Booch & Jacobson, 1999] Booch, G., J., R., & Jacobson. (1999). Proceso Unificado de Desarrollo de Software.
- [Caro, 1996] Caro, T. (1996). Ingeniería de Requisitos de Software. Madrid: Servicio de Publicaciones de la Facultad de Informática, Universidad Politécnica.
- [Fowler & Kendall, 1997] Fowler, M., & Kendall, S. (1997). UML Gota a Gota . Mexico DF: Addison Wesley Longman .
- [Idaberto, 2000] Idaberto, C. (2000). Administración de recursos humanos. Bogotá: ME Graw HILL.
- [Kendall, 1997] Kendall, K. &. (1997). Análisis y Diseño de Sistemas. New York: Services of New England Inc.
- [Mendoza, 2008] Mendoza, M. S. (2008). Analiysis Services. Perú: Megabyte.
- [Minguez & Garcia, 2000] Minguez, D., & Garcia, E. (2000). Metodologia para el Desarrollo de Aplicaciones Web: UWE.
- [Piatini, 1999] Piatini, M. (1999). Fundamentos y Modelos de Bases de Datos (2da ed.). Mexico DF: Alfaomega.
- [Prentice, 1999] Prentice, L. (1999). UML y Patrones Introduccion al Analisis y Diseño Orientado a Objetos .

[Pressman, 2006]

Pressman, R. (2006). Ingeniería de Software un Enfoque Practico (sexta edicion ed.). Mexico DF.

[Rumbaugh, Jacobson & Booch, 2000]

Rumbaugh, J., Jacobson, I., & Booch, G. (2000). El Lenguaje Unificado de Modelo.Madrid: Pearson .

SITIOS DE INTERNET

[AISI, 2007]

Asociación de investigación en software inteligente AISI. (2007). Ingeniería Web. Recuperado el 4 de Agosto de 2011, de <http://www.AISI.html>.

[Ambler, 2000]

Ambler, S. (12 de Marzo de 2010). Metodologias Proceso Agil . Recuperado el 15 de Junio de 2011, de Metodologias del Proceso Agil : www.agilemodeling.com/essays/agileModelingRUP.htm.

[ICT, 2011]

Academia Latinoameericana de Seguridad Informática. (28 de Enero de 2011). Implementación y plan de seguridad. Recuperado el 1 de Noviembre de 2011, de <http://www.piramidedigital.com/Documentos/ICT/pdictseguridadinformaticaimplementacion>.

[Copley, 2010]

Copley, S. (25 de Enero de 2010). Implementación del Nuevo sistema. Recuperado el 5 de Noviembre de 2011, de <http://www.igcseict.info/theory/8/implem/>

[IEEE, 2011]

IEEE, Norma IEEE 9000 estándar. Recuperado el 15 Noviembre del 2011, de <http://www.ieee.org>.

[Koch, 2011]

Unit of programming and software engineering. (11 de Junio de 2011). UWE-UML- Based Web Engineering. Recuperado el 10 de Agosto de 2011, de <http://uwe.pst.ifi.lmu.de/examples.html>.

[Ministerio de Trabajo, 2011]

Ministerio de Trabajo del Estado Plurinacional de Bolivia. (2011). Ministerio de Trabajo, empleo y previsión social . Recuperado el 5 de Diciembre de 2011, de <http://www.mintrabajo.gob.bo/Principal.asp>.





*Si tu intención es describir la verdad,
hazlo con sencillez y la elegancia
déjasela al sastre. (Albert Einstein)*

Anexo A

Análisis de Situación

Introducción.

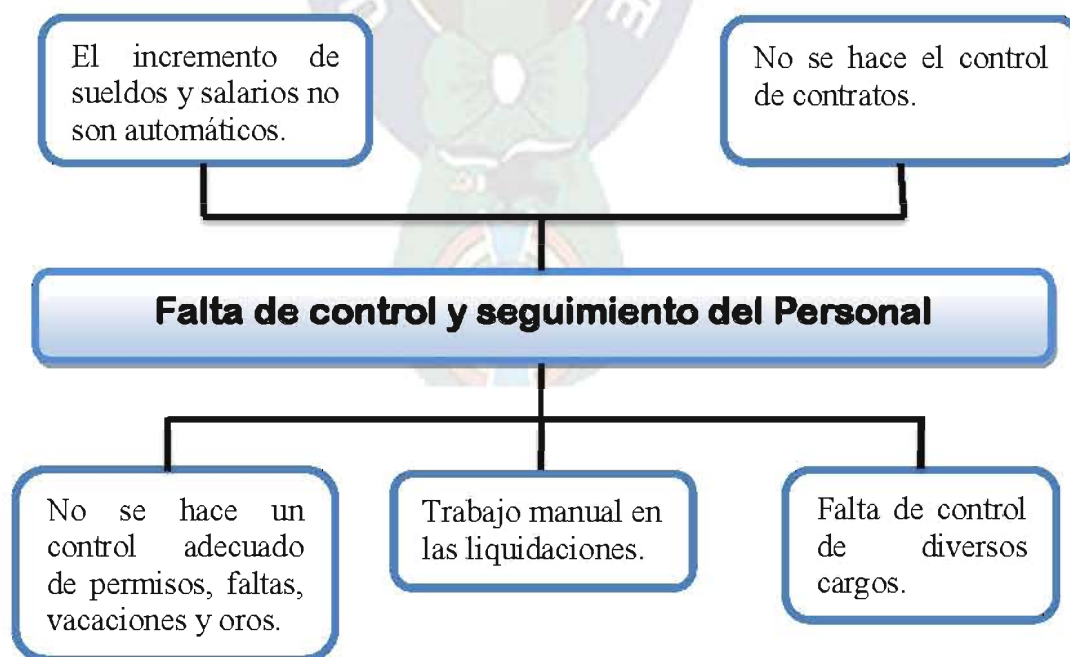
El tema a desarrollar es acerca de la administración de personal en la Dirección Administrativa de la UTB (Universidad Tecnológica Boliviana), para poder llevar un control y seguimiento del mencionado, además poder llevar un control de todos los empleados y sus respectivas funciones y roles que desempeñan en la institución. Con el control y seguimiento del mismo se espera hacer llegar resultados fidedignos para la toma de decisiones de la UTB.

1. Título

“Sistema Web de Control y Seguimiento de Personal Caso: Universidad Tecnológica Boliviana”.

2. Problema

2.1 Árbol de problemas



2.2 Problema Principal

Realizar el control y el seguimiento del personal.

2.3 Causas

- No se hace un control adecuado de permisos, faltas, vacaciones y otros.
- Trabajo manual en las liquidaciones.
- Falta de control de diversos cargos.

2.4 Efectos

- El incremento de sueldos y salarios no son automáticos.
- Mala toma de decisiones.
- No se tiene integración entre las áreas de liquidación y personal.

3. Objetivo

3.1 Árbol de Objetivos



3.2 Medios y Fines

Realizar un control y seguimiento del personal.

3.3 Condiciones Positivas

- Desarrollar el módulo de control y de procesos.
- Automatizar el proceso de liquidación.
- Realizar el módulo de registro de empleado.
- Automatizar el incremento de sueldos y jornales.

4. Matriz de involucrados

Grupos	Intereses	Problemas Percibidos	Recursos y Mandatos
Beneficiarios: Dirección Administrativa	Realizar un control y seguimiento minucioso del personal.	No se puede realizar el seguimiento de todos los empleados.	Ley del trabajo.
Implementadores: María J. Aguilar	Dar la solución al problema de la manera más óptima.	Se debe realizar el control y seguimiento.	Ley del trabajo.
Decisores: Director Administrativo	Poder rendir informes a sus superiores en el momento que sea necesario.	El sistema actual no es amigable para los usuarios.	Ley del trabajo.
Financiadores: Área financiera	Poder controlar los sueldos y salarios del personal.	El sistema no hace un seguimiento de los sueldos y salarios.	Ley del trabajo.

5. Importancia del Tema

Lo que se destaca en la realización de este proyecto será el registro de todos los empleados en el sistema, en la parte de control se podrá llevar a cabo un seguimiento minucioso del personal, además se podrá pasar informes en fecha actual para que instancias superiores se encuentre al tanto del estado de algún empleado o actividad que se encuentre realizando, la parte encargada de todos estos procesos es la dirección administrativa que se encarga de los recursos humanos de la UTB.

6. Metodología

Para el desarrollo de este sistema se utilizará la metodología ágil AUP y en cuanto al diseño la herramienta será UWE.

7. Matriz de Marco Lógico

Descripción	Indicadores	Verificadores	Supuestos
Desarrollar un Sistema Web de Control y Seguimiento de Personal		En el portal de la UTB se Podrá ver el sistema propuesto.	La dirección administrativa obtiene los datos de todo el personal de la UTB.
Automatizar el control, seguimiento del personal.	Desarrollar un sistema web para el control y seguimiento del personal para el mes de Noviembre.	El sistema se encontrará en el portal de la UTB y se podrá realizar el control y seguimiento del Personal.	Se debería haber firmado un contrato para tener un empleado registrado en el sistema.
C-1 Registro del personal. C-2 Registro de permisos o vacaciones. C-3 Seguimiento del personal. C-4 Generar informes para toma de decisiones.	1. Realizar una interfaz para autenticación de empleado en el mes de Septiembre. 2. Realizar la interfaz de Registro de permisos o vacaciones en el mes de Octubre. 3. Realizar la interfaz para hacer un seguimiento en el mes de Octubre. 4. Realizar la interfaz de Reportes en el mes Noviembre.	1. Listado de todo los empleados de la UTB. 2. Con documentación de respaldo o verificación que presentan los Empleados. 3. Con los informes que se realicen después del registro de empleados. 4. Con los informes impresos que se realizan después de cada registro.	El director administrativo tiene que hacer contratos previos al registro.

<p>A-1</p> <ul style="list-style-type: none"> • Crear cuenta en el sistema. • Registrarse en el sistema. • Revisar los datos del Personal. <p>A-2</p> <ul style="list-style-type: none"> • Actualizar el estado del empleado. <p>A-3</p> <ul style="list-style-type: none"> • Confirmar datos registrados. • Realizar el seguimiento al empleado escogido. • Obtener resultados para informes. <p>A-4</p> <ul style="list-style-type: none"> • Comparar los resultados con los documentos. • Realizar informes • Enviar informe al encargado de la toma de decisiones. 	<p>1.1 El costo de crear cuenta es aproximadamente Bs 100.</p> <p>1.2 El costo de registro es aproximadamente Bs 100.</p> <p>1.3 El costo de revisar los datos es aproximadamente Bs 500.</p> <p>2.1 El costo de actualizar es aproximadamente Bs 700.</p> <p>3.1 El costo de confirmación es aproximadamente Bs 700.</p> <p>3.2 El costo de seguimiento es aproximadamente Bs 200.</p> <p>3.3 El costo de obtención de resultados es aproximadamente Bs 1000.</p> <p>4.1 El costo de ver resultados es aproximadamente Bs 500.</p> <p>4.2 El costo de reportes es Bs 1000.</p> <p>4.3 El costo de enviar información es Bs 500.</p>	<p>Los indicadores de las actividades se verificarán viendo el sistema funcionando en el portal de la UTB en la etapa de implementación del sistema.</p>	<p>Todos</p>
--	--	--	--------------

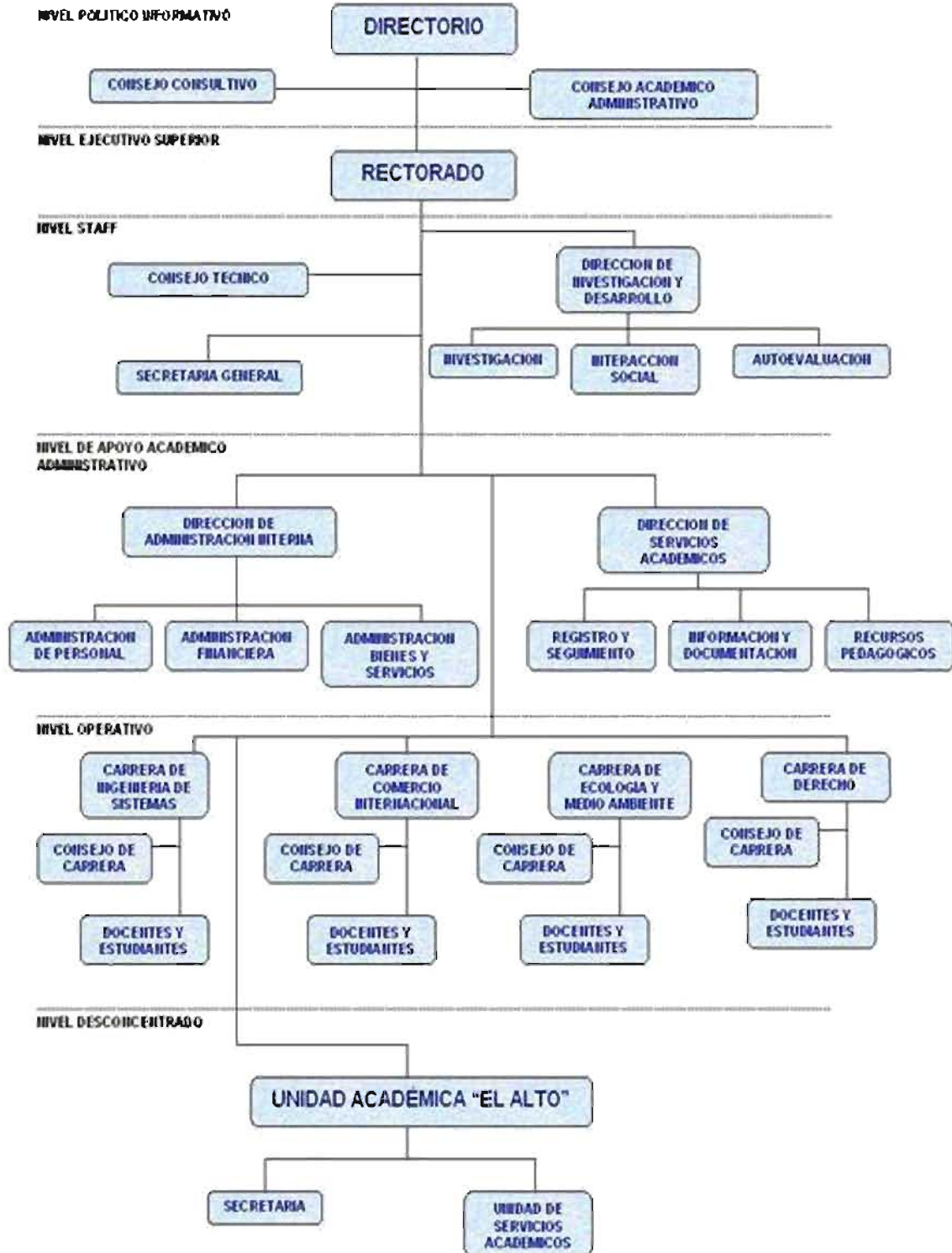
8. Cronograma

Actividad	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre
Realizar el análisis y el perfil.						
Marco Teórico						
Realizar una interfaz para autenticación de usuarios.						
Marco Aplicativo						
Realizar la interfaz de Registro de empleado.						
Realizar la interfaz de modificación del empleado.						
Realizar la interfaz para el registro de avance del empleado.						
Realizar la interfaz para hacer el seguimiento.						
Métricas de Calidad						
Realizar la interfaz para consultas.						
Realizar la interfaz de Reportes.						
Realizar pruebas en el sistema.						
Conclusiones y recomendaciones						





ANEXO B

ORGANIGRAMA FUNCIONAL UNIVERSIDAD TECNOLÓGICA BOLIVIANA



ANEXO D

	<h1>FINIQUITO</h1>		
ESTADO PLURENACIONAL DE BOLIVIA		MINISTERIO DE TRABAJO, EMPLEO Y PREVISIÓN SOCIAL	
I.- DATOS GENERALES			
RAZÓN SOCIAL O NOMBRE DE LA EMPRESA		1 2	
RAMA DE ACTIVIDAD ECONÓMICA	DOMICILIO		
NOMBRE DEL TRABAJADOR		1 2	
ESTADO CIVIL	EDAD	DOMICILIO	
PROFESION U OCUPACIÓN			
CI	FECHA DE INGRESO	FECHA DE RETIRO	
MOTIVO DEL RETIRO	REMUNERACION MENSUAL Bs		
TIEMPO DE SERVICIO	AÑOS	MESES DIAS	
II - LIQUIDACIÓN DE LA REMUNERACIÓN PROMEDIO INDEMNIZABLE EN BASE A LOS 3 ÚLTIMOS MESES			
A) MESES			TOTALES
REMUNERACIÓN MENSUAL	Bs	Bs	Bs
B) OTROS CONCEPTOS PERCIBIDOS EN EL MES			
	Bs	Bs	Bs
	Bs	Bs	Bs
	Bs	Bs	Bs
TOTAL			
III.- TOTAL REMUNERACIÓN PROMEDIO INDEMNIZABLE (A + B) DIVIDIDO ENTRE 3:			Bs
C) DESAHUCIO TRES MESES (EN CASO DE RETIRO FORZOSO)			Bs
D) INDEMNIZACIÓN POR TIEMPO DE TRABAJO:			
DE	AÑOS	Bs	
DE	MESES	Bs	
DE	DIAS	Bs	
AGUINALDO DE NAVIDAD	DE	MESES Y	DIAS Bs
VACACION	DE	MESES Y	DIAS Bs
PRIMA LEGAL (SI CORRESPONDE)	DE	MESES Y	DIAS Bs
OTROS			Bs
	GESTION	DE	DIAS Bs
IV.- TOTAL BENEFICIOS SOCIALES: C + D			Bs
E) DEDUCCIONES:			
		Bs	
		Bs	
		Bs	
		Bs	TOTAL Bs
V. IMPORTE LÍQUIDO A PAGAR C + D - E =			Bs



*No vayas fuera, vuelve a tí mismo. En
el hombre interior habita la verdad.
(San Agustín)*

La Paz, Diciembre del 2011

Señora:

Lic. Menfy Morales Ríos
**DIRECTORA DE LA CARRERA DE INFORMÁTICA
PRESIDENTA DEL HONORABLE CONSEJO DE LA CARRERA DE INFORMÁTICA
UNIVERSIDAD MAYOR DE SAN ANDRÉS**

Presente.-

Ref.: AVAL DE CONCLUSIÓN DE PROYECTO DE GRADO

De mi consideración.

Mediante la presente, tengo a bien dirigirme a su autoridad a objeto de informar la conclusión del Proyecto de Grado titulado "SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE PERSONAL CASO: UNIVERSIDAD TECNOLÓGICA BOLIVIANA", desarrollado por la Univ. **MARÍA JUANA AGUILAR TORREZ** con C.I. 5010742 TJA, que después de realizar el seguimiento y supervisión del mencionado proyecto, doy mi conformidad para que se fije fecha y hora para la defensa del Proyecto de grado.

Esperando una respuesta favorable me despido.

Atentamente



Lic. Fatima Consuelo Dolz de Moreno M. Sc.
**DOCENTE TUTOR
CARRERA DE INFORMÁTICA
U.M.S.A.**

La paz, Diciembre del 2011

Señora:

Lic. Fatima Consuelo Dolz de Moreno M. Sc.
**DOCENTE TUTOR TALLER DE LICENCIATURA II
CARRERA DE INFORMÁTICA**


Presente.-

Ref: **AVAL DE CONCLUSIÓN DE PROYECTO DE GRADO**

Mediante la presente, tengo a bien dirigirme a usted con el objeto de comunicarle que habiendo realizado el seguimiento y revisión del Proyecto de grado titulado "**SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE PERSONAL CASO: UNIVERSIDAD TECNOLÓGICA BOLIVIANA**", realizado por la Univ. María Juana Aguilar Torrez con C.I. 5010742 TJA. En mi calidad de docente revisor doy a conocer mi conformidad y aval del contenido de forma y fondo del trabajo presentado, para que la postulante pueda realizar la defensa pública del mencionado Proyecto de Grado, para optar el título de Licenciatura en Informática con mención en Ingeniería de Sistemas Informáticos, de acuerdo a normas y reglamentos vigentes.

Con este motivo saludo a usted con mis consideraciones más distinguidas.

Atentamente:


Lic. Aldo Ramiro Vazquez Alvarado
**DOCENTE REVISOR
CARRERA DE INFORMÁTICA
U.M.S.A.**

La Paz, Diciembre del 2011

Señora:


Lic. Menfy Morales Ríos
DIRECTORA DE LA CARRERA DE INFORMÁTICA
PRESIDENTA DEL HONORABLE CONSEJO DE LA CARRERA DE INFORMÁTICA
UNIVERSIDAD MAYOR DE SAN ANDRÉS

Presente.-

De mi consideración:

Por medio de la presente me dirijo a su respetable autoridad con el fin de hacerle conocer que la universitaria **MARÍA JUANA AGUILAR TORREZ**, con C.I. 5010742 TJA., de la carrera de Informática, implementó el proyecto de Grado denominado **"SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE PERSONAL CASO: UNIVERSIDAD TECNOLÓGICA BOLIVIANA"** en nuestra institución presentando su propuesta respaldada con la demostración final y la puesta en producción del sistema informático. Por consiguiente doy mi conformidad y aval de conclusión de proyecto de grado.

Con este motivo saludo a usted muy atentamente.



Lic. Miguel O. Medrano Caredo
DIRECTOR ADM. FINANCIERO
Universidad Tecnológica Boliviana S.A.