

UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA



PROYECTO DE GRADO

**“Sistema de Servicio de Archivo y Kardex
CASO: GOBIERNO AUTONOMO MUNICIPAL DE LA PAZ”
PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA**

POSTULANTE: Univ. JUSTA CONZUELO ILLANES CESPEDES
TUTOR METODOLOGICO: MG. SC. MIGUEL COTAÑA MIER
ASESOR: MG.SC. CARLOS MULLISACA CHOQUE

LA PAZ – BOLIVIA

2013



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



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria

A mi padre que ya se encuentra en presencia del Creador, dedicarle este presente documento quien permanentemente me apoyo con su espíritu alentador.

A mi madre, brindándome la fuerza necesaria para continuar y momentos de ánimo así mismo ayudándome en lo que fuera posible.

A mis hijos Gabriel y Marcelo que son motivo de mi inspiración para seguir adelante y cumplir mi objetivo.

Conzuelo

Agradecimiento

Mi gratitud, principalmente está dirigida a Dios por estar siempre a mi lado en el transcurso de mi vida.

También me gustaría agradecer a mis docentes durante toda mi carrera profesional porque todos han aportado con un granito de arena a mi formación, en especial a Mg.Sc. Carlos Mullisaca Choque por la disponibilidad y sugerencias en el proceso de revisión del presente proyecto.

A la Universidad Mayor de San Andrés, que abre sus puertas a todas las personas que quieren superarse.

A mis compañeros de RR.HH. de la Unidad de Administración de Personal del Gobierno Autónomo Municipal de La Paz por su colaboración y por el apoyo que me brindaron, especialmente a los señores Marco Antonio Saavedra M. Director de RR.HH. y a Guillermo Jimenez M. Jefe de Unidad de Administración de Personal.

Y a todas aquellos amig@s que de una y otra forma, colaboraron o participaron en la realización de este proyecto de grado, hago extensivo mi más sincero agradecimiento.

Muchas Gracias

RESUMEN

Hoy en día con las nuevas técnicas que se cuenta para poder desarrollar un sistema que pueda llevar un control riguroso de la documentación que se tiene, elaborar y mantener actualizados los expedientes del personal, ejecutivo, administrativo, técnico y obrero.

Garantizar el resguardo y custodia de los Expedientes del personal de la Institución, así como la confidencialidad de los documentos contenidos en él.

La metodología RUP (Rational Unified Progress) se emplea en el siguiente proyecto, para el análisis y diseño los cuales se desarrollan en las etapas de inicio, elaboración, construcción y transición esta metodología es aplicada en el desarrollo de Software, las cuales están apoyadas con la metodología UWE (Ingeniería Web basada en UML).

El sistema de Servicio de Archivo y kardex soluciona los problemas de sistematización de datos para un mejor control de toda la documentación personal que se genera de todos los empleados que trabajan o trabajaron en la institución. En la actualidad el Gobierno Autónomo Municipal de La Paz cuenta con aproximadamente 5000 empleados; de los cuales se tiene un archivo personal en la sección de kardex, en este archivo se tiene documentos de: hoja de vida, certificados de nacimiento, contratos, variedad de memorándums como se de llamadas de atención, de promoción, certificados de dependientes, accidentes, bajas medicas, permisos comisiones, vacaciones etc; esta voluminosa documentación que se genera nos lleva a la necesidad de contar con un sistema que ayude al personal administrativo a realizar un control de esta documentación generada y su respectiva ubicación y de esta forma puedan emitir certificaciones e informes que se les solicita de una forma oportuna y precisa.

INDICE

CAPITULO 1 INTRODUCCION	1
1.1. Introducción.	2
1.2. Antecedentes.	2
1.3. Planteamiento del Problema.....	3
1.4. Objetivos	3
1.4.1. Objetivo General	3
1.4.2. Objetivo Especifico.....	4
1.5. Justificación.	5
1.5.1. Justificación Técnica.....	5
1.5.2. Justificación Social.....	5
1.5.3. Justificación Económica.....	6
1.6. Alcances	6
CAPITULO 2 MARCO TEORICO	7
2.1. Servicio de Archivo.	8
2.1.1. Concepto de documento.....	9
2.1.2. Funciones del Archivo	9
2.2. Intranet	10
2.3. Metodología de Desarrollo	10
2.3.1. Metodología del Proceso Unificado de Rational	10
2.3.1.1. Fase de Inicio	13
2.3.1.2. Fase de Elaboración	15
2.3.1.3. Fase de Construcción	17
2.3.1.4. Fase de Transición.....	20

2.3.2.	UWE (UML – Based Web Engineering).....	21
2.3.3.	Metodología Scrum.....	27
2.4.	Programación y Arquitectura 3 capas.-	31
2.5.	Lenguaje Unificado de Modelado o UML.....	35
2.6.	Herramientas de Modelado.....	38
2.6.1	Rational Rose.....	38
2.6.2	Enterprise Architect.....	39
2.6.3	PowerDesigner	40
2.7.	Pruebas al Sistema.....	42
2.8.	Herramientas Informáticas	42
2.9	Métricas y Calidad de Software	44
CAPITULO 3 MARCO APLICATIVO.....		50
3.1.	Introducción	51
3.2.	Fase de Inicio.....	51
3.2.1.	Modelado del Negocio	51
3.2.2.	Modelado de Casos de uso del Negocio	53
3.2.3.	Requerimientos del Sistema.....	62
3.3.	Fase de Elaboración	63
3.3.1	Diagramas de Casos de uso	63
3.3.2.	Diagramas de Colaboración.....	73
3.3.3.	Diagramas de Secuencia.....	74
3.4.	Fase de Construcción	75
3.4.1	Requerimientos Finales.....	75
3.4.2.	Diagramas de Actividades	76

3.4.3. Diagrama de Paquetes	77
3.4.4. Diagrama de Componentes	78
3.4.5. Diagrama de Clases	79
3.4.6. Diagrama Navegacional.....	80
3.4.7. Prototipo de Interfaz de Usuario	81
3.4.8. Interfaces de Usuario	82
3.4.9. Pruebas del Sistema	84
3.5. Fase de Transición.....	87
CAPITULO 4 SEGURIDAD, CALIDAD Y COSTO BENEFICIO	88
4.1. Seguridad.....	89
4.1.1 Seguridad del Software	93
4.1.2 Seguridad Administrativa	94
4.2 Factores de Calidad ISO 9126.....	94
4.2.1 Confiabilidad.....	94
4.2.2 Funcionalidad.....	96
4.2.3 Facilidad de Mantenimiento.....	100
4.2.4 Portabilidad	101
4.3. Análisis de Costos	101
4.3.1. Costo de Desarrollo de Software.....	101
4.3.2. Costo de Capacitación del Usuario	102
4.3.3. Costo del Software de Construcción del Sistema	102
CAPITULO 5 CONCLUSIONES Y RECOMENDACIONES.....	103
5.1 Conclusiones.....	104
5.2 Recomendaciones.....	106

CAPITULO 6 BIBLIOGRAFIA	107
6.1. Bibliografía	108
6.2. WebGrafía	108
ANEXOS.....	109

INDICE DE FIGURAS

Figura 2. 1 Diagrama Metodológico RUP	11
Figura 2. 2 Ciclo de Vida RUP	12
Figura 2. 3 Casos de Uso	14
Figura 2. 4 Diagrama de Colaboración	15
Figura 2. 5 Diagrama de Secuencia.....	16
Figura 2. 6 Diagrama de Actividad.....	17
Figura 2. 7 Diagrama de Componentes.....	18
Figura 2. 8 Comparación entre Diagramas de Clase (a) RUP (b) UML.....	19
Figura 2. 9 Principio de separación avanzada de conceptos	21
Figura 2. 10 Elementos del Modelo de Navegación	23
Figura 2. 11 Ejemplo Modelo de Navegación	23
Figura 2. 12 Elementos del Modelo de Presentación.....	24
Figura 2. 13 Ejemplo del Modelo de Presentación	24
Figura 2. 14 Arquitectura tres capas	32
Figura 2. 15 Arquitectura para Aplicaciones.net	34
Figura 2. 16 Ejemplo de Estructura del Programa	35
Figura 3. 1 Diagrama del Modelado del Negocio.....	52
Figura 3. 2 Modelado de casos de uso de Negocio	53
Figura 3. 3 Registro de File.....	63
Figura 3. 4 Registro de Préstamos.....	64
Figura 3. 5 Registro de Files	73
Figura 3. 6 Diagrama de Colaboración de Registro de Préstamos.....	74
Figura 3. 7 Diagrama de Secuencia de Registro de File	74
Figura 3. 8 Diagrama de Secuencia de Registro de File	75
Figura 3. 9 Diagrama de Actividad de Registro de File.....	76
Figura 3. 10 Diagrama de Actividad de Registro de Prestamos	76
Figura 3. 11 Diagrama de Paquetes.....	77

Figura 3. 12 Diagrama de Componentes.....	78
Figura 3. 13 Diagrama de Componentes.....	79
Figura 3. 14 Diagrama Navegacional del Sistema.....	80
Figura 3. 15 Pantalla de Inicio	81
Figura 3. 16 Pantalla de Búsqueda.....	81
Figura 3. 17 Pantalla de Reportes	82
Figura 3. 18 Menú de Inicio.....	83
Figura 3. 19 Pantalla de Busqueda.....	83
Figura 3. 20 Pantalla de Reportes	84
Figura 3. 21 Grafo de flujo para el Registro de Files.....	84
Figura 3. 22 Grafo de flujo para el Registro de Prestamos	85
Figura 4. 1 Tabla tblUsuarioRol.....	89
Figura 4. 2 Tabla tblRol	90
Figura 4. 3 Tabla tblMenuRol.....	90
Figura 4. 4 Tabla tblMenu.....	91
Figura 4. 5 Tabla M_Usuarios	91
Figura 4. 6 Tabla Audit	92
Figura 4. 7 Diagrama de Transferencia.....	95



CAPITULO 1 INTRODUCCION

1.1.Introducción.

La evolución de nuevas tecnologías hace que la sociedad tenga mejores posibilidades de comunicación y por lo tanto las distancias y el tiempo se acortan.

El Gobierno Autónomo Municipal de La Paz (G.A.M.L.P.), está compuesta por varios niveles como ser: Legislativo y Fiscalizador, (Concejo Municipal) Direcciones, Direcciones Técnicas, Direcciones de Acciones estratégicas (Oficialías Mayores), Operación Desconcentrada (Sub alcaldías); para cada funcionario que trabaja en cada uno de estos niveles se tiene registrado un folder de documentos personales, cada persona puede generar a la vez otra documentación como ser vacaciones, permisos, bajas médicas, comisiones etc, esta nueva documentación debe ser archivada en los respectivos files como respaldo a la información ingresada al sistema.

El presente proyecto trata de tener un control adecuado y fiable de esta documentación cuya misión será la de recibir, clasificar custodiar y facilitar documentos de manera sistemática.

1.2.Antecedentes.

En esos últimos años el Gobierno Autónomo Municipal de La Paz – GMLP está encaminado en implementar herramientas TIC, es así que la Dirección de Gestión de Recursos Humanos – DGRH lleva adelante un proyecto informático denominado “*SISTEMA INTEGRADO DE GESTIÓN DE RECURSOS HUMANOS - SIGRH*”, con el objeto de contar y disponer de información útil, oportuna y confiable, en el marco de un enfoque sistémico de administración y control. Teniendo productos informáticos desarrollados, se tiene a bien dar curso a este emprendimiento de consolidación presentando una solución para esta nueva etapa con el objetivo de consolidar el Subsistema Servicio de Archivo y kardex, para el proceso de la información de manera integrada, lo cual permitirá una adecuada gestión de información del personal con el cual cuenta actualmente el Gobierno Municipal de La Paz y así sea convergente a los módulos ya desarrollados hasta la fecha como ser el Subsistema de Control de Personal, Subsistema de Bienestar Social, Sistema de Registro de Personal (Anexo).

Trabajos Similares

En la Biblioteca de la Carrera de Informática, en los proyectos de grado presentados por los estudiantes de la carrera, se puede encontrar los siguientes temas similares.

- Biblioteca Digital Académica (Aranda 2007), en este proyecto ya va encaminado a la digitalización de libros para así poder contribuir a la enseñanza de niños.

- Biblioteca Digital Programas de Ayuda Humanitaria Save the Children, cuyo objetivo principal es la administración de los recursos bibliográficos en formato digital tener acceso a estos, además de tener referencia de los programas de ayuda que ofrece la organización No gubernamental Save the Children. (Cerezo 2010).

Estos proyectos no enfocan el trabajo a través de un intranet, y la interacción con otros sistemas que provean de información al mismo; están más dirigidas a un enfoque social.

1.3.Planteamiento del Problema.

Para un mejor análisis y planteamiento se recurre al Árbol de problemas en el Anexo, de los cuales citamos los siguientes:

- Manejo no apropiado del expediente.
- Documentos Dispersos, deterioro de estos.
- Ubicación No Rápida del expediente.
- Información no exacta de tenencia del expediente.
- Desconocimiento de cantidad de folios de cada expediente
- Control no eficiente en la recepción de devoluciones de expedientes.
- Generación de documentación voluminosa para archivar.

Con la ayuda del Árbol de problemas se logra planear el siguiente problema principal:

¿El Sistema de Servicio de Archivo y Kardex mejorará los procesos de comunicación y transferencia de información a través del Intranet del G.A.M.L.P. centralizando y ajustando a la plataforma .net, tomando en cuenta los nuevos requerimientos de la entidad?

1.4.Objetivos

1.4.1. Objetivo General

Para hallar el objetivo principal se utiliza el análisis mediante el árbol de objetivos (Anexo).

Implantar el sistema de Servicio de Archivo y Kardex de los expedientes del personal cuya misión será la de recibir, clasificar custodiar y facilitar documentos de manera sistemática, y de esta manera disminuir el esfuerzo de trabajo y facilitar el acceso de la

información y el seguimiento, para cumplir de una forma rápida el control de toda la documentación y mantener actualizados los expedientes del personal, el cual ayudara en la toma de decisiones.

1.4.2. Objetivo Especifico

Los objetivos específicos son los siguientes:

- Generar una ficha de salida del expediente personal, con el registro de datos del funcionario a quien se presta esta documentación, en base al kardex de cada funcionario, cargo que ocupa, ubicación de puesto de trabajo, cantidad de files que se le presta, fecha de préstamo.
- Ubicar físicamente el expediente en los estantes respectivos.
- Llevar un control de cuantas veces es prestado un expediente y la devolución completa de las mismas.
- Elaborar, tramitar y enviar los récords de servicio y archivar copia en el expediente del empleado en cuestión.
- Elaborar y mantener actualizados los expedientes del personal, ejecutivo, administrativo, técnico y obrero.
- Garantizar la guarda y custodia de los Expedientes del personal de la Institución, así como la confidencialidad de los documentos contenidos en él.
- Garantizar que cada “Movimiento de Personal” se archive acompañado de los soportes legales correspondientes.
- Llevar control de los files del personal, tanto activo como egresado.
- Elaborar, tramitar, y entregar las constancias de trabajo así como cualquier otro documento que soliciten los empleados y los organismos oficiales.
- Mantener actualizada la relación de personal para el otorgamiento de los reconocimientos por tiempo de servicio en la Institución.
- Recibir y archivar la “Relación de Reposos Médicos” del Nivel Central y de los Centros Locales, seguidamente archivar cada reposo, debidamente sellado y firmado por el Servicio Médico, en el expediente correspondiente.
- Implantar y controlar el Sistema de Organización adecuado para manejar el archivo.

1.5. Justificación.

1.5.1. Justificación Técnica

La automatización de procesos operativos, proporciona información que sirva de apoyo para la toma de decisiones. A través de estos sistemas se lleva un control adecuado de toda la documentación que se obtiene, y el respectivo registro de este en el sistema.

Para el desarrollo de este sistema se cuenta con el hardware y el software necesario que ayudara en su implementación.

1.5.2. Justificación Social

La correcta evaluación del personal con respaldo de la documentación que tiene registrada genera un ambiente de trabajo donde todos los involucrados se desenvuelvan satisfactoriamente y puedan lograr alcanzar sus objetivos. También se debe tomar en cuenta los aspectos legales con los que está sujeto el G.A.M.L.P., ya que estas normas ligan de manera directa al personal con la entidad.

La administración de Recursos Humanos del G.A.M.L.P. se rige y fundamenta su accionar a las siguientes leyes:

- Ley General del Trabajo (LGT),
- Ley 2028 de Municipalidades
- Decreto Supremo 26115 que aprueba las Normas Básicas de Administración de Personal (NB-SAP),
- Reglamento Especifico del Sistema de Administración Municipal del G.A.M.L.P.,
- Manual de Organización y Funciones del G.A.M.L.P.,
- Manual de Puestos del G.A.M.L.P.,
- Reglamento Interno de Personal RIP,
- Ley 2027 del Estatuto del Funcionario Público,
- Ley 1178 SAFCO,
- Otras disposiciones Legales Vigentes.

1.5.3. Justificación Económica

Los costos económicos del sistema son racionales, además que el G.A.M.L.P. cuenta con equipos de computación servidores exclusivos para recursos humanos y software con licencia.

Otra de las justificaciones es la necesidad que tiene la unidad de kardex, de tener un control de sus files, en donde tienen archivados, las solicitudes de vacaciones, las licencias a cuenta de vacaciones, comisiones, etc. y además de toda la información que implicaría para procesos legales.

1.6. Alcances

El presente proyecto abarca el desarrollo de aplicaciones que sea viable y que tenga una aceptación y que cuente con las siguientes características:

- Este sistema debe estar integrado al Sistema de Gestión de Recursos Humanos del G.A.M.L.P.;
- Generación de boleta de registro de préstamo de expediente;
- Control de tenencia de los expedientes del personal;
- Control de registro de vacaciones en base a documentación;
- Control de ubicación física de los expedientes del personal;
- Generación de cuadros de control de mando.



CAPITULO 2 MARCO TEORICO

2.1. Servicio de Archivo.

La palabra archivo, del latín archivium, presenta tres acepciones distintas de acuerdo con el Diccionario de la R.A.L.E. (Real Academia de la Lengua Española):

- Local en el que se custodian documentos públicos o particulares (lo que significa la existencia de archivos no solo oficial sino también semipúblicos, comerciales o particulares); conjunto de estos documentos y el mueble/s que los contiene (carpetas, guías, etc., donde se colocan).
- Otras disposiciones hacen referencia al archivo como aquella institución cuya función primordial es la de reunir, organiza, conservar y difundir, utilizando las técnicas adecuadas, dichos conjuntos de documentos para el cumplimiento de los fines anteriormente mencionados.
- También se le da este nombre al servicio especializado (Servicio de archivo) cuya misión consiste en recibir, clasificar, custodiar y facilitar documentos.
- Archivar significa guardar de forma ordenada documentos útiles haciéndolo de un modo lógico y eficaz que permita su posterior localización de la forma más rápida posible cuando sea necesario.

El material de archivo más corriente lo constituyen: Carpetas, Archivadores. Guías, etc.

- Clase de Archivos:

Según el grado o frecuencia de utilización:

- Archivos activos o de gestión: recogen documentos recién entregados en la entidad o de consulta frecuente.
- Semiactivos: contienen documentos provenientes de archivos activos por haberse realizado sobre ellos el trabajo concreto.
- Inactivos: recoge los documentos que habiendo perdido todo su valor operativo y funcional, conservan valor histórico, político o documental.

Según el lugar de emplazamiento:

- Generales o centralizados: documentos que se conservan en el mismo lugar y es consultado por distintos departamentos.

- Descentralizados o parciales: se encuentran en las distintas dependencias de la entidad. Estos a su vez pueden ser:
- Departamentales o por secciones.- en el propio departamento por ser utilizado constantemente por este.
- Personales.- de cada puesto de trabajo para que pueda ser de acceso inmediato a la persona que trabaja con ellos.

Estas dos clasificaciones no son excluyentes, si no que se complementan entre sí.

2.1.1. Concepto de documento

Para la Real Academia Española documentar significa “probar documentos”, al mismo tiempo define como documento “un escrito o cualquier otra cosa que pruebe o acredita algo”. Este sería el concepto general de documento.

La Ley de Patrimonio Histórico define el documento como “toda expresión en lenguaje natural o convencional y cualquier otra expresión gráfica, sonora o en imagen”, recogidas en cualquier tipo de soporte material, incluidos los soportes informáticos.

Desde el punto de vista administrativo, documento sería toda información o hecho fijado o registrado en cualquier tipo de soporte material que sirvan para comprobar o acreditar algo. [MONOGRAFIAS, 2011]

2.1.2. Funciones del Archivo

La principal función del archivo consiste en la *conservación de documentos*, ya que la ley obliga a ello estableciendo unos plazos mínimos de conservación, durante los cuales puede ser requerida su presentación.

Estos documentos se conservan debidamente ordenados y clasificados. Esta conservación proporciona información sobre el funcionamiento y los asuntos tratados por la entidad. El archivo será eficaz cuando se encuentra rápidamente lo que se busca.

Otra función del archivo es la de ser un *centro activo de información* que permite relacionar los nuevos documentos con los ya archivados. Además sirve como *medio de consulta* cuando se pretende indagar en las actuaciones del pasado. También el archivo sirve como *elemento probatorio* cuando el organismo o entidad pretende demostrar la realización de un acto o la forma de hacerlo. [MONOGRAFIAS, 2011]

2.2. Intranet

La tecnología de Internet se adoptó a un nivel empresarial mediante la utilización de diversas herramientas informáticas colaborativas. De esta forma, surge la Intranet como respuesta a la necesidad de comunicarse e interactuar eficiente y efectivamente hacia el seno interno de las organizaciones manteniendo a los clientes y funcionarios cada vez más informados e interconectados entre sí, promoviendo mayor eficiencia, incrementando la productividad y reduciendo costos.

Definición de Intranet

Se podría definir a la Intranet como una herramienta similar o igual que la Internet, el cual se origina por la necesidad de comunicación interacción entre las diferentes áreas o miembros dentro de la organización; esta es la aplicación de la tecnología de Internet a una red privada, es decir en redes corporativas internas, cuyos usuarios son los miembros de la organización.

Su utilización brinda gran cantidad de beneficios, entre ellos: optimizar y compartir los recursos de la empresa, incrementar su productividad, minimizar el uso del papel, aumentar la prestación de servicios y la calidad de los mismos a los clientes; disponibilidad de información oportuna que apoye la búsqueda de soluciones y la toma de decisiones.

Esta tecnología no tiene restricción en cuanto a su tamaño o lugar físico, la única restricción es el acceso único a usuarios autorizados. Además, la Intranet podría estar conectada a Internet o funcionar como una red aislada.

2.3. Metodología de Desarrollo

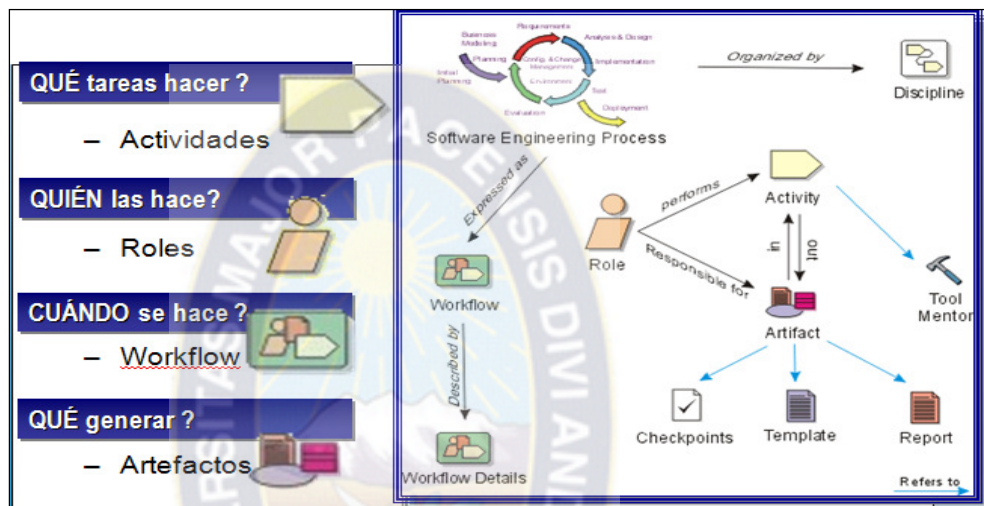
2.3.1. Metodología del Proceso Unificado de Rational

RUP es un proceso para el desarrollo de un proyecto de software que define claramente ¿qué tareas hacer?, ¿quién lo hace?, ¿cuándo se hace?, ¿qué generar en el proyecto? (Ver Figura 2.1).

Entre sus características esenciales se puede decir que está dirigida por los Casos de Uso, que orientan el proyecto a la importancia para el usuario y lo que este quiere, está centrado en la arquitectura, que relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden, es iterativo e incremental, donde divide el proyecto en mini proyectos.

Captura varias de las mejores prácticas en el desarrollo moderno de software que es aplicable para un amplio rango de proyectos y organizaciones. [E Villalba Domínguez, 2012]

Figura 2. 1 Diagrama Metodológico RUP



Fuente: [Cybertec, 2004]

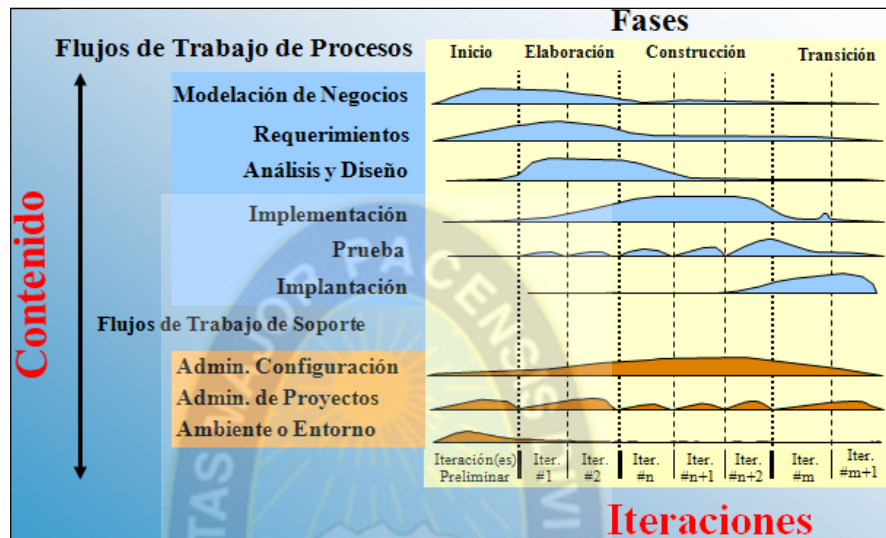
El ciclo de vida de RUP

RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades como muestra la Figura 2.2.

Iteración

Una iteración es un ciclo completo de desarrollo que produce una versión (interna o externa) de un producto ejecutable, que constituye un subconjunto del producto final en desarrollo, que luego se irá incrementando de iteración en iteración hasta convertirse en el sistema final; cada fase el Proceso Unificado de Rational puede descomponerse en iteraciones.

Figura 2. 2 Ciclo de Vida RUP



Fuente: [GUERRERO, 2004]

Flujos de Trabajo del Proceso

El proceso Unificado de Rational consta de nueve flujos de trabajo

- Modelado de Negocio**, describe la estructura y la dinámica de la organización;
- Requisitos**, describe el método basado en casos de uso para extraer los requisitos;
- Análisis y Diseño**, describe las diferencias arquitectónicas;
- Implementación**, tiene en cuenta el desarrollo de software, la prueba de unidades y la integración;
- Pruebas**; describe los casos de prueba, los procedimientos y las métricas para evaluación de defectos;
- Despliegue**, cubre la configuración del sistema entregable;
- Gestión de Configuraciones**, controla los cambios y mantiene la integridad de los artefactos de un proyecto;
- Gestión del Proyecto**, las diferencias vistas arquitectónicas;
- Análisis y Diseño**, Describe varias estrategias de trabajo en un proceso iterativo.

2.3.1.1. Fase de Inicio

Durante la fase de inicio las iteraciones ponen mayor énfasis en actividades del modelado del negocio y de requisitos.

- Modelado del negocio, en esta fase el equipo se familiarizará más al funcionamiento de la empresa, sobre conocer sus procesos.
 - Entender la estructura y la dinámica de la organización para la cual el sistema va a ser desarrollado;
 - Entender el problema actual en la organización objetivo e identificar potenciales mejoras;
 - Asegurar que clientes, usuarios finales y desarrolladores tengan un entendimiento común de la organización objetivo.
- Requisitos, en esta línea los requisitos son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que especifiquemos.
 - Establecer y mantener un acuerdo entre clientes y otros stakeholders sobre lo que el sistema podría hacer;
 - Proveer a los desarrolladores un mejor entendimiento de los requisitos del sistema;
 - Definir el ámbito del sistema;
 - Proveer una base para estimar costos y tiempo de desarrollo del sistema;
 - Definir una interfaz de usuarios para el sistema, enfocada a las necesidades y metas del usuario.

Para la elaboración de esta fase se utilizará uno de los diagramas perteneciente al Lenguaje Unificado de Modelado (UML) para representar lo detallado anteriormente.

Diagramas de Casos de Uso

El modelado de Casos de Uso es la técnica más efectiva y a la vez la más simple para modelar los requisitos del sistema desde la perspectiva del usuario. Los Casos de Uso se utilizan para modelar cómo un sistema o negocio funciona actualmente, o cómo los usuarios desean que funcione. No es realmente una aproximación a la orientación a

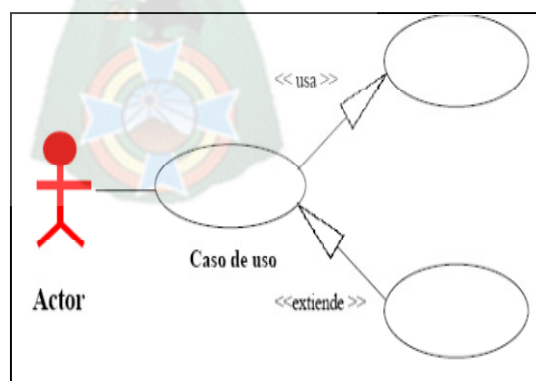
objetos; es realmente una forma de modelar procesos. Es, sin embargo, una manera muy buena de dirigirse hacia el análisis de sistemas orientado a objetos. Los casos de uso son generalmente el punto de partida del análisis orientado a objetos con UML (Ver Figura 2.3).

El modelo de casos de uso consiste en actores y casos de uso. Los actores representan usuarios y otros. Cada caso de uso se documenta por una descripción del escenario. La descripción puede ser escrita en modo de texto o en un formato paso a paso. Cada caso de uso puede ser también definido por otras propiedades, como las condiciones pre- y post- del escenario - condiciones que existen antes de que el escenario comience, y condiciones que existen después de que el escenario se completa.

Un caso de Uso está compuesto por los siguientes elementos:

- Un **actor** es el papel que el usuario juega con respecto al sistema. Un actor no tiene que ser un humano, puede ser por ejemplo otro sistema externo que pide información al sistema actual;
- La relación **<<extiende>>** se utiliza cuando un caso de uso es similar a otro caso de uso pero se le añade alguna característica nueva;
- La relación **<< usa >>** se utiliza cuando se tiene una parte del comportamiento común a más de un caso de uso, y no se desea almacenar una copia en cada caso de uso de la descripción de este comportamiento.

Figura 2. 3 Casos de Uso



Fuente: [Lovelley, 2001]

2.3.1.2. Fase de Elaboración

En la fase de elaboración, las iteraciones se orientan al desarrollo de la *baseline* de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la *baseline* de la arquitectura.

Análisis y Diseño, en esta actividad se especifican los requerimientos y se describen sobre cómo se van a implementar en el sistema.

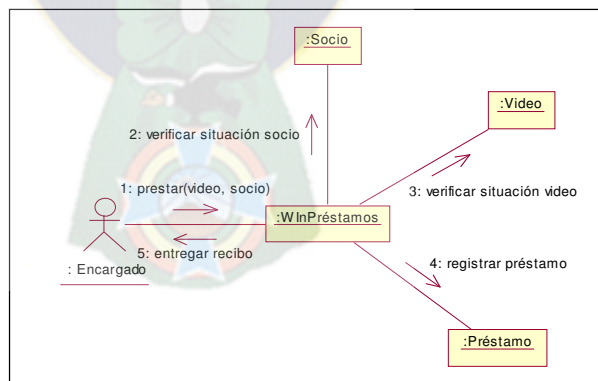
- Transformar los requisitos al diseño del sistema;
- Desarrollar una arquitectura para el sistema;
- Adaptar el diseño para que sea consistente con el entorno de implementación.

Para plasmar lo detallado se utilizara algunos diagramas pertenecientes al Lenguaje Unificado de Modelado (UML), los mismos serán detallados a continuación.

Diagramas de Colaboración

El Diagrama de Colaboración presenta una alternativa al diagrama de secuencia para modelar interacciones entre objetos en el sistema (Ver Figura 2.4).

Figura 2. 4 Diagrama de Colaboración



Fuente: [Letelier, 2003]

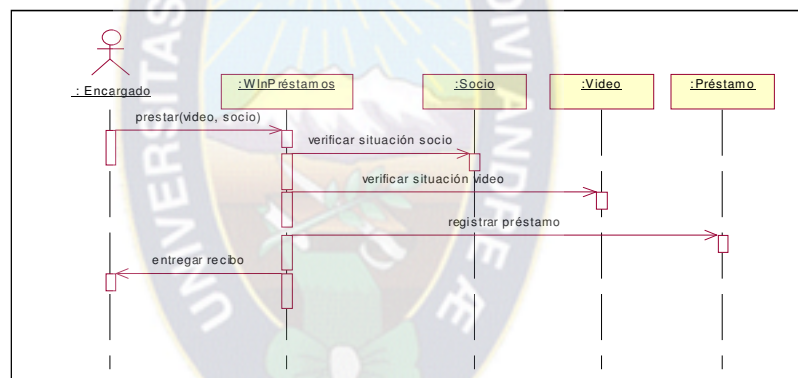
Mientras que el diagrama de secuencia se centra en la secuencia cronológica del escenario que estamos modelando, el diagrama de colaboración se centra en estudiar todos los efectos de un objeto dado durante un escenario. Los objetos se conectan por medio de enlaces, cada enlace representa una instancia de una asociación entre las clases

implicadas. El enlace muestra los mensajes enviados entre los objetos, el tipo de mensaje (sincrónico, asincrónico, simple, *blanking*, y *'time-out'*), y la visibilidad de un objeto con respecto a los otros.

Diagrama de Secuencia

El Diagrama de Secuencia es uno de los diagramas más efectivos para modelar interacción entre objetos en un sistema como muestra la Figura 2.5. Un diagrama de secuencia se modela para cada caso de uso. Mientras que el diagrama de caso de uso permite el modelado de una vista *'business'* del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes pasados entre los objetos.

Figura 2. 5 Diagrama de Secuencia



Fuente: [Letelier, 2003]

Típicamente uno examina la descripción de un caso de uso para determinar qué objetos son necesarios para la implementación del escenario. Si tienes modelada la descripción de cada caso de uso como una secuencia de varios pasos, entonces puedes "caminar sobre" esos pasos para descubrir qué objetos son necesarios para que se puedan seguir los pasos.

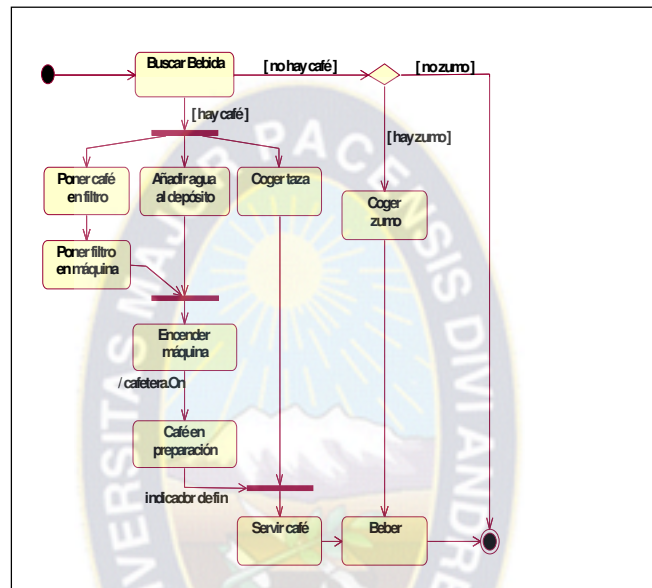
Diagramas de Actividad

El Diagrama de Actividad es un diagrama de flujo del proceso multipropósito que se usa para modelar el comportamiento del sistema. Los diagramas de actividad se pueden usar para modelar un Caso de Uso, o una clase, o un método complicado (Ver Figura 2.6).

Un diagrama de actividad es parecido a un diagrama de flujo; la diferencia clave es que los diagramas de actividad pueden mostrar procesado paralelo (*parallel processing*).

Esto es importante cuando se usan diagramas de actividad para modelar procesos 'business' algunos de los cuales pueden actuar en paralelo, y para modelar varios hilos en los programas concurrentes.

Figura 2. 6 Diagrama de Actividad



Fuente: [Letelier, 2003]

2.3.1.3. Fase de Construcción

En esta fase implementan las clases y objetos en ficheros fuente, binarios, ejecutables y demás. El resultado final es un sistema ejecutable.

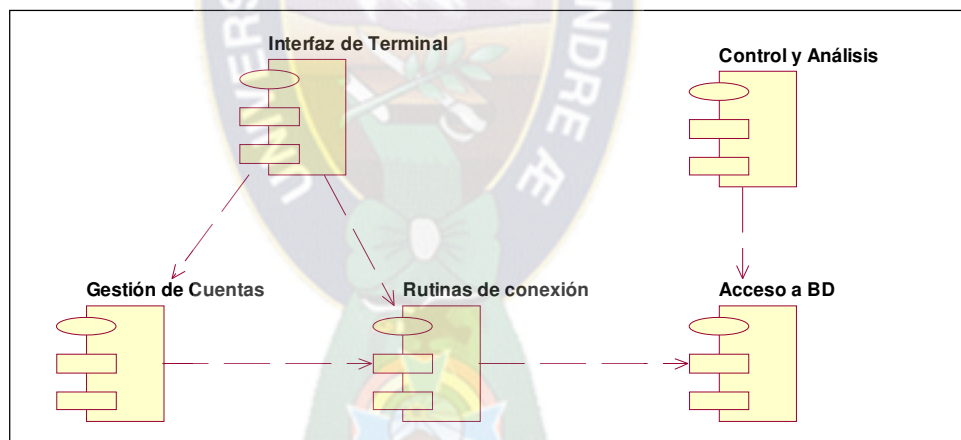
- Planificar qué subsistemas deben ser implementados y en qué orden deben ser integrados, formando el Plan de Integración;
- Cada implementador decide en qué orden implementa los elementos del subsistema;
- Si encuentra errores de diseño, los notifica;
- Se integra el sistema siguiendo el plan.
- Pruebas, este flujo de trabajo es el encargado de evaluar la calidad del producto que estamos desarrollando, pero no para aceptar o rechazar el producto al final del proceso de desarrollo, sino que debe ir integrado en todo el ciclo de vida.
 - Encontrar y documentar defectos en la calidad del software;

- Generalmente asesora sobre la calidad del software percibida;
- Provee la validación de los supuestos realizados en el diseño y especificación de requisitos por medio de demostraciones concretas;
- Verificar las funciones del producto de software según lo diseñado;
- Verificar que los requisitos tengan su apropiada implementación.

Diagrama de Componentes

El Diagrama de Componentes se usa para modelar la estructura del software, incluyendo las dependencias entre los componentes de software, los componentes de código binario, y los componentes ejecutables. Con el Diagrama de Componentes modelas componentes del sistema, a veces agrupados por paquetes, y las dependencias que existen entre componentes (paquetes de componentes) así como nos muestra la figura 2.7.

Figura 2. 7 Diagrama de Componentes



Fuente: [Letelier, 2003]

Diagrama de Clases

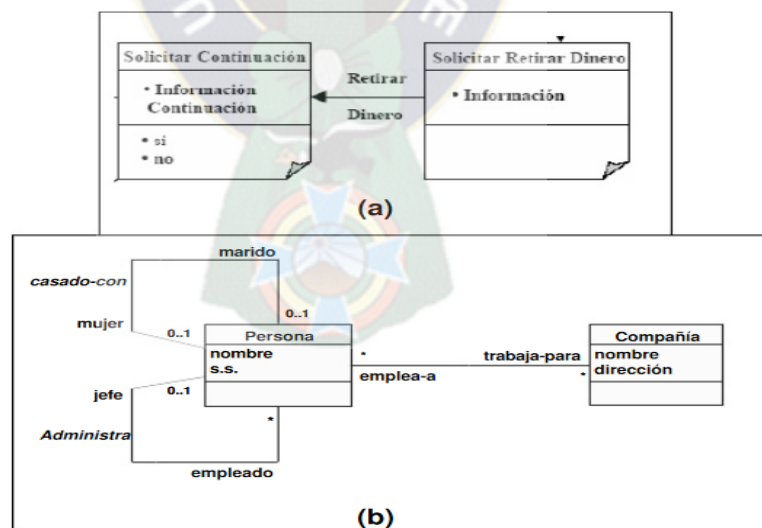
El Diagrama de Clase presenta un mecanismo de implementación neutral para modelar los aspectos de almacenamiento de datos del sistema. Las clases persistentes, sus atributos, y sus relaciones pueden ser implementados directamente en una base de datos orientada a objetos. Aun así, en el entorno de desarrollo actual, la base de datos relacional es el método más usado para el almacenamiento de datos.

Es en el modelado de esta área donde UML se queda corto. El diagrama de clase de UML se puede usar para modelar algunos aspectos del diseño de bases de datos relacionales, pero no cubre toda la semántica involucrada en el modelado relacional, mayoritariamente la noción de atributos clave que relacionan entre sí las tablas unas con otras.

El Diagrama de Clase se puede usar para modelar la estructura lógica de la base de datos, independientemente de si es orientada a objetos o relacional, con clases representando tablas, y atributos de clase representando columnas. Si una base de datos relacional es el método de implementación escogido, entonces el diagrama de clases puede ser referenciado a un diagrama de relación de entidad lógico. Las clases persistentes y sus atributos hacen referencia directamente a las entidades lógicas y a sus atributos; el modelador dispone de varias opciones sobre cómo inferir asociaciones en relaciones entre entidades.

Los diagramas de clase contienen lo siguiente: clases, interfaces, colaboraciones, y relaciones de dependencia, generalización y asociación. A través de esta representación se interpreta como base de datos por medio de objetos con sus respectivas relaciones (ver Figura 2.8).

Figura 2. 8 Comparación entre Diagramas de Clase (a) RUP (b) UML



Fuente: [Letelier, 2003]

Modelando la Distribución y la Implementación con UML

Los Diagramas de Implementación se usan para modelar la configuración de los elementos de procesamiento en tiempo de ejecución (*run-time processing elements*) y de los componentes, procesos y objetos de software que viven en ellos. En el diagrama *'deployment'*, empiezas modelando nodos físicos y las asociaciones de comunicación que existen entre ellos. Para cada nodo, puedes indicar qué instancias de componentes viven o corren (se ejecutan) en el nodo. También puedes modelar los objetos que contiene el componente.

Los Diagramas de Implementación se usan para modelar sólo componentes que existen como entidades en tiempo de ejecución; no se usan para modelar componentes solo de tiempo de compilación o de tiempo de enlazado. Puedes también modelar componentes que migran de nodo a nodo u objetos que migran de componente a componente usando una relación de dependencia con el estereotipo *'becomes'* (se transforma).

2.3.1.4. Fase de Transición

El objetivo es traspasar el software desarrollado a la comunidad de usuarios.

Una vez instalado surgirán nuevos elementos que implicarán nuevos desarrollos (ciclos).

Incluye:

- Pruebas Beta para validar el producto con las expectativas del cliente;
- Ejecución paralela con sistemas antiguos;
- Conversión de datos;
- Entrenamiento de usuarios;
- Distribuir el producto.

Obtener autosuficiencia de parte de los usuarios.

Concordancia en los logros del producto de parte de las personas involucradas.

Lograr el consenso cuanto antes para liberar el producto al mercado.
[EVillalbaDomínguez, 2012]

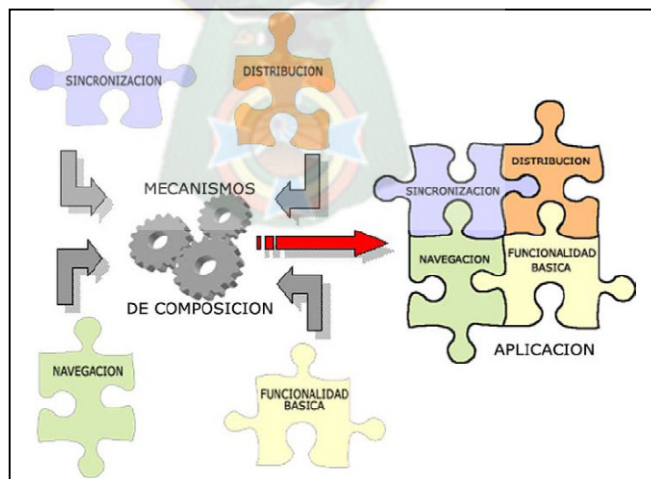
2.3.2. UWE (UML – Based Web Engineering)

El cada vez más diverso mundo de la ingeniería del software, la complejidad creciente de las aplicaciones y la gran cantidad de herramientas disponibles, hace que se convierta en fundamental, a la hora de embarcarse en un proyecto software, la reutilización de elementos. Desde los comienzos de la ingeniería del software se han buscado distintos mecanismos de encapsulación (funciones, plantillas, clases, componentes) para alcanzar este objetivo. El último pasó que han dado los investigadores a la hora de obtener un nuevo mecanismo de encapsulación, y por lo tanto de reutilización, se conoce como programación orientada a aspectos o separación avanzada de conceptos (Figura 2.9).

Esta nueva corriente se basa en la máxima de que es más fácil razonar sobre un sistema si se especifican cada uno de los conceptos que lo forman por separado, dejando a la separación avanzada de conceptos definir los mecanismos para componer estos aspectos. Esta especificación por separado tiene un claro impacto en la comprensión, la trazabilidad, la evolución y la reusabilidad del software.

Dados estos conceptos ahora la pregunta sería ¿Qué es UWE?, UWE es La 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. [O.Fernandez, 2011].

Figura 2. 9 Principio de separación avanzada de conceptos



Fuente:[Reina,2006]

UWE está especializada en la especificación de aplicaciones adaptativas, y por tanto hace especial hincapié en características de personalización, como es la definición de un modelo de usuario o una etapa de definición de características adaptativas de la navegación en función de las preferencias, conocimiento o tareas de usuario.

Otras características relevantes del proceso y método de autoría de UWE son el uso del paradigma orientado a objetos, su orientación al usuario, la definición de un meta-modelo (modelo de referencia) que da soporte al método y el grado de formalismo que alcanza debido al soporte que proporciona para la definición de restricciones sobre los modelos.

Por lo que respecta al proceso de autoría de la aplicación, UWE hace un uso exclusivo de estándares reconocidos como UML y el lenguaje de especificación de restricciones asociado OCL. Para simplificar la captura de las necesidades de las aplicaciones web, UWE propone una extensión que se utiliza a lo largo del proceso de autoría. Este proceso de autoría está dividido en cuatro pasos o actividades:

- Análisis de Requisitos: Fija los requisitos funcionales de la aplicación Web para reflejarlos en un modelo de casos de uso.
- Diseño Conceptual: Materializado en un modelo de dominio, considerando los requisitos reflejados en los casos de uso.
- Diseño Navegacional: Lo podemos subdividir en :
 - Modelo del Espacio de Navegacional.
 - Modelo de la Estructura de navegación: Muestra la forma de navegar ante el espacio de navegación.
- Diseño de Presentación: Representa las vistas del interfaz del usuario mediante modelos estándares de interacción UML. [PORTELLA, 2011]

La Ingeniería Web basada en UML, es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML.

Modelo de Navegación UWE

Este modelo indica como el sistema de páginas web del sitio está relacionado internamente. Es decir cómo se enlazan los elementos de navegación.

Para ellos se utilizan unidades de navegación llamadas “nodos” conectadas por enlaces de navegación. Estos nodos pueden ser mostrados en la misma página web, no tienen por qué estar en páginas diferentes.

Al mismo tiempo que explicamos este modelo con el ejemplo de la agenda de contactos, podemos ir viendo los distintos elementos que introduce la metodología UWE como la figura 2.10.

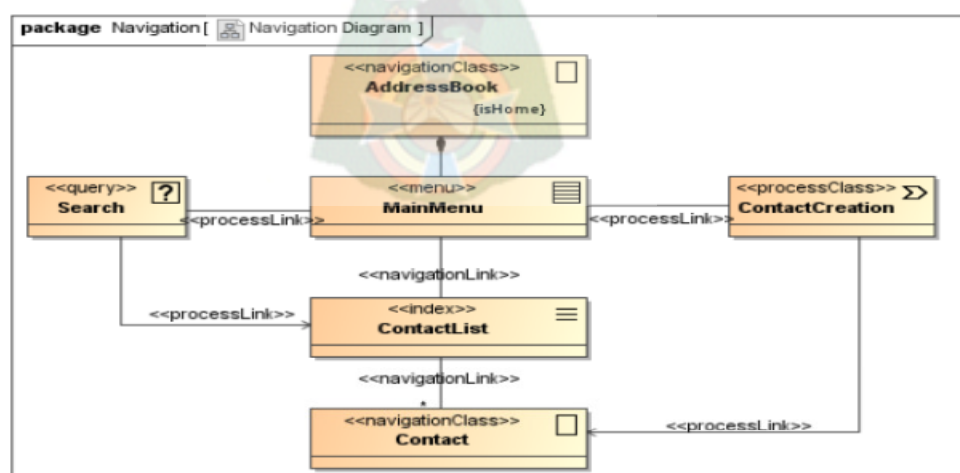
Figura 2. 10 Elementos del Modelo de Navegación



Fuente: [Koch, 2006]

Aquí tenemos el ejemplo de navegación del sitio web que representa una agenda de contactos como muestra la figura 2.11

Figura 2. 11 Ejemplo Modelo de Navegación

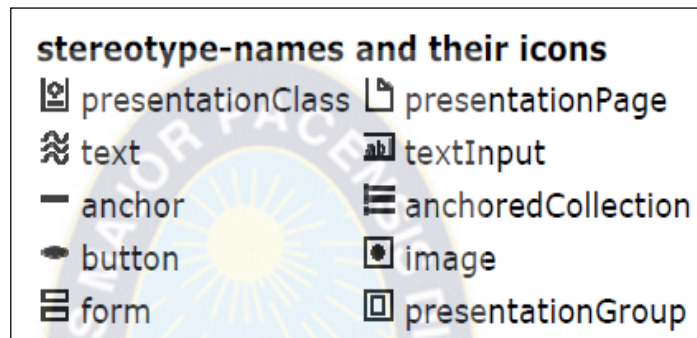


Fuente: [MagicDraw, 2010]

Modelo de Presentación

En este modelo se representan las clases de navegación y de procesos que pertenecen a cada página web. Los elementos que se introducen se muestran en la figura 2.12.

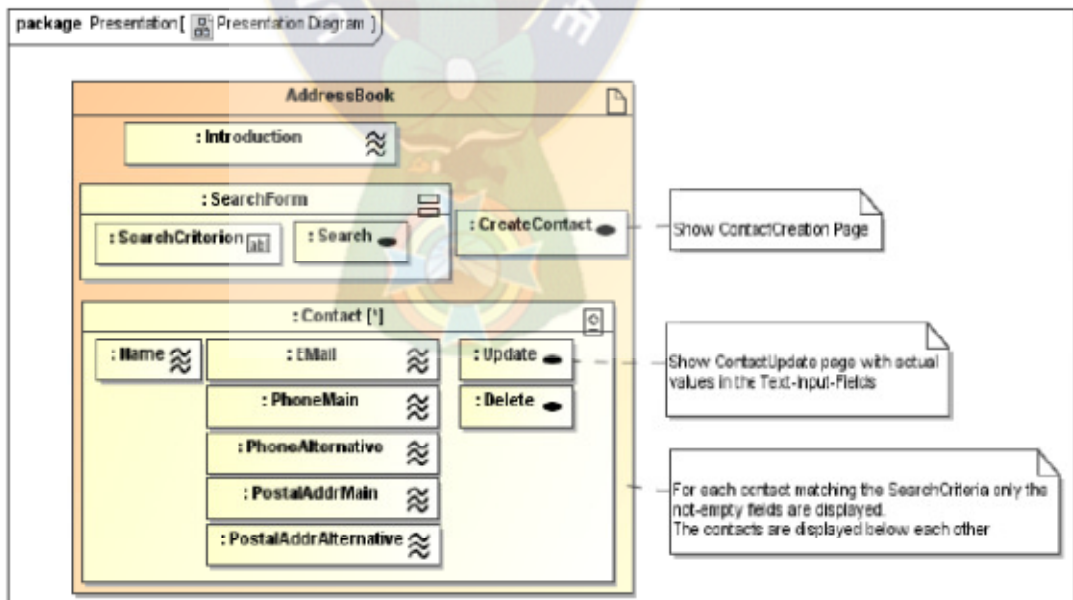
Figura 2. 12 Elementos del Modelo de Presentación



Fuente: [Koch, 2006]

A continuación se muestra el diagrama de presentación en la figura 2.13 del ejemplo de la agenda de contactos.

Figura 2. 13 Ejemplo del Modelo de Presentación



Fuente: [MagicDraw, 2010]

Consideraciones de diseño

Se deben considerar los siguientes aspectos en el diseño de componentes de una aplicación web: [SA5, 2008]

a) Textos:

- El uso de muchas fuentes en una página dificulta la lectura, es recomendable no utilizar más de dos fuentes distintas.
- Se debe utilizar normalmente una altura de 10 a 12 puntos para el texto.
- Se debe considerar que las líneas de texto cortas son más fáciles de leer que las largas.
- Es conveniente alinear los textos por la izquierda, la alineación derecha dificulta la lectura del mismo.

b) Hipervínculos:

- Es necesario que cada hipervínculo contenga una descripción coherente para que el usuario identifique su función.
- Se debe elegir una longitud adecuada para cada enlace.
- No se debe cambiar el color de los enlaces si no es necesario.
- No se debe colocar enlaces a páginas en construcción o que se encuentran fuera de servicio.

c) Longitud de una página

- Se debe diseñar páginas cortas y concisas, estas no deben exceder de una pantalla y media.
- Si una página tiende a ser bastante extensa, la mejor opción consiste en dividir en varias páginas el contenido de la Web.
- Las páginas no deben sobrepasar los 800 píxeles de longitud, pues gran parte de los monitores tienen esta medida.

d) Gráficos

- Se debe utilizar solo las imágenes necesarias, pues estas aumentan el tiempo de espera del navegador.
- Es conveniente utilizar iconos o símbolos que puedan sustituir a textos escritos, pues esto contribuirá a internacionalizar la aplicación Web.
- Se debe utilizar un mismo estilo para la creación de todos los iconos de la aplicación Web, con el mismo tamaño y estilo gráfico.
- No se debe utilizar fondos gráficos muy recargados, estos pueden desviar la atención del visitante.

e) Uso del color

- Utilizar colores contrastantes y seguros, para facilitar la lectura de un texto.
- Especificar el color de fondo, ya que este se mostrará en la pantalla inmediatamente.
- Los colores pastel o neutros para el fondo de la página aumentan considerablemente la legibilidad del texto.
- Utilizar colores y combinaciones iguales para todas las páginas de una misma aplicación Web.

f) Navegabilidad

- Es conveniente incluir un encabezado al inicio de cada página, que informe el nombre del sitio y la sección del mismo.
- Se debe elegir un buen título para el documento, que refleje el contenido general del documento.
- No dejar un camino cerrado, donde los usuarios tengan que presionar el botón back de su navegador para pasar a la página anterior.
- Proporcionar siempre un enlace a la página principal, esto será de gran ayuda en aplicaciones Web de muchas páginas.

g) Calidad del documento

- Probar regularmente cada enlace de la aplicación Web.
- Revisar la ortografía y la gramática de todas las páginas de la aplicación Web.

- No utilizar comandos específicos de ciertos navegadores.
- Se debe ofrecer a los visitantes la posibilidad de comunicarse con el administrador del sitio, pues su opinión es de gran ayuda para mejorar la calidad de la aplicación Web.
- Debe informarse a los usuarios la fecha de la última modificación de la aplicación Web. [[LMU – Ludwig-Maximilians-Universität München](#), 2012]

2.3.3. Metodología Scrum

¿Qué es?

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa (ROI). Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación.



¿Cuándo se utiliza?

Con la metodología Scrum el cliente se entusiasma y se compromete con el proyecto dado que lo ve crecer iteración a iteración. Asimismo le permite en cualquier momento realinear el software con los objetivos de negocio de su empresa, ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración sin ningún problema. Esta metódica de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades.

Beneficios

- **Cumplimiento de expectativas:** El cliente establece sus expectativas indicando el valor que le aporta cada requisito / **historia** del proyecto, el equipo los estima y con esta información el **Product Owner** establece su prioridad. De manera regular, en las demos de Sprint el **Product Owner** comprueba que efectivamente los requisitos se han cumplido y transmite se feedback al equipo.
- **Flexibilidad a cambios:** Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado. La metodología está diseñada para adaptarse a los cambios de requerimientos que conllevan los proyectos complejos.
- **Reducción del Time to Market:** El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo.
- **Mayor calidad del software:** La metódica de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.
- **Mayor productividad:** Se consigue entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
- **Maximiza el retorno de la inversión (ROI):** Producción de software únicamente con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión.
- **Predicciones de tiempos:** Mediante esta metodología se conoce la velocidad media del equipo por sprint (los llamados puntos historia), con lo que consecuentemente, es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está en el Backlog.
- **Reducción de riesgos:** El hecho de llevar a cabo las funcionalidades de más valor en primer lugar y de conocer la velocidad con que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

Procesos y Roles del Scrum

El proceso

El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada **Sprint**, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. En

cada nuevo **Sprint**, se va ajustando la funcionalidad ya construida y se añaden nuevas prestaciones priorizándose siempre aquellas que aporten mayor valor de negocio.



- **Product Backlog:** Conjunto de requisitos denominados historias descritos en un lenguaje no técnico y priorizados por valor de negocio, o lo que es lo mismo, por retorno de inversión considerando su beneficio y coste. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares.
- **Sprint Planning:** Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir.
- **Sprint:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las **historias** del **Product Backlog** a las que se ha comprometido, en una nueva versión del software totalmente operativo.
- **Sprint Backlog:** Lista de las tareas necesarias para llevar a cabo las **historias** del sprint.
- **Daily sprint meeting:** Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada. Cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos.
- **Demo y retrospectiva:** Reunión que se celebra al final del sprint y en la que el equipo presenta las historias conseguidas mediante una demostración del

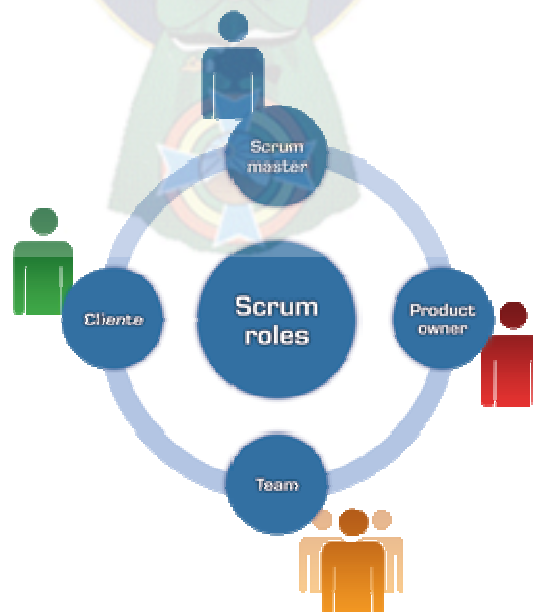
producto. Posteriormente, en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos.

Roles

En Scrum, el equipo se focaliza en construir software de calidad. La gestión de un proyecto Scrum se centra en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en vencer cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo.

El equipo Scrum está formado por los siguientes roles:

- **Scrum master:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el **Product Owner** para maximizar el ROI.
- **Product owner (PO):** Representante de los accionistas y clientes que usan el software. Se focaliza en la parte de negocio y es responsable del ROI del proyecto (entregar un valor superior al dinero invertido). Traslada la visión del proyecto al equipo, formaliza las prestaciones en **historias** a incorporar en el **Product Backlog** y las reprioriza de forma regular.
- **Team:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las **historias** a las que se comprometen al inicio de cada sprint. [SOFTENG, 2013]



2.4. Programación y Arquitectura 3 capas.-

La metodología utilizada actualmente en el desarrollo de soluciones de software empresarial depende en una gran medida de las características del proyecto, de los integrantes del equipo de desarrollo y de las tecnologías a utilizar. En realidad, no existe una metodología concreta que nos permita un desarrollo eficaz. El continuo proceso de cambio en el proyecto, los integrantes del equipo y la renovación y mejora de las tecnologías implica un continuo replanteamiento en los pasos a seguir. Esta falta de continuidad en la metodología conlleva graves inconvenientes para garantizar la obtención de una solución fiable, fácilmente mantenible y sobretodo una solución que pueda evolucionar en el tiempo.

A través del entorno de diseño en tres capas (3LD Figura 2.14) aporta una metodología de declaración de los conceptos de negocio, definición de la interacción con el sistema, procesos a realizar sobre los conceptos, y finalmente nos permite establecer restricciones a este modelo y validaciones.

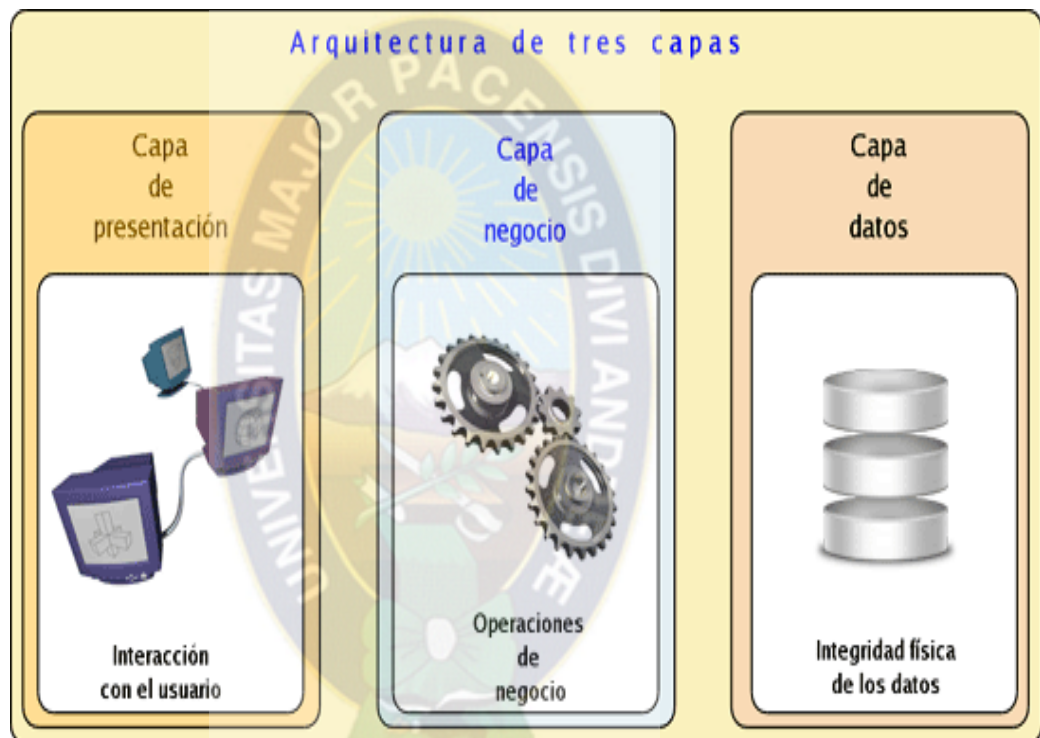
La idea del desarrollo en 3 capas consistía en utilizar un método de desarrollo para nuestros sistemas que nos permita separar esto en distintas capas. Las capas recomendadas eran: La interfaz de usuario, las reglas de negocio y la Base de Datos, la idea de esta arquitectura estaba basada principalmente en la capacidad de escalabilidad que esto nos ofrece, por ejemplo, si tenemos desarrollada una aplicación basada en un motor de datos de VFP, y si en el desarrollo de esta, aplicamos apropiadamente las reglas de diseño de tres capas, cuando quisiéramos llevar la misma a funcionar, por ejemplo, con SQL Server, no deberíamos tocar más que el motor de datos, la capa cliente quedaría intacta y, como mucho algún "toque" muy superficial pudiera sufrir la capa de reglas de negocios.

Si bien al momento del diseño podemos observar una mayor carga de complejidad, la utilización de esta arquitectura nos brinda un control más cercano de cada componente, así como también la posibilidad de una verdadera reutilización del código. Por ejemplo: Si tuviésemos en nuestras reglas de negocios una función que nos liste el resumen de cuenta de nuestros clientes cuando un usuario de nuestra aplicación Win32, desde su capa cliente ingrese a este módulo se ejecuta `MostrarListaCuenta()` para ver el detalle en su monitor; pero si el desarrollador del sitio web de la empresa, al momento de armar la página de consulta nos pide esto, le diremos que simplemente cargue el componente de reglas de negocios y ejecute `MostrarListaCuenta()`. Como podemos ver, este es un concreto ejemplo de reutilización de código y a su vez, una herramienta que nos ofrece seguridad, ya que mediante la utilización de esta arquitectura, el desarrollador de una

interfaz de usuario jamás llegará a manipular directamente un dato en nuestro servidor sino es a través de un componente.

Las ventajas de esta arquitectura son las siguientes: **escalabilidad, modularidad y flexibilidad.**

Figura 2. 14 Arquitectura tres capas



Fuente: [Callaos, 2004]

a) Capa de datos

Esta capa contiene clases que interactúan con la base de datos. Estas clases surgen como una necesidad de mantener la cohesión o clases altamente especializadas que ayuden a reducir la dependencia entre las clases y capas. Aquí podemos encontrar también una clase con métodos estáticos que permiten uniformizar las operaciones de acceso a datos a través de un único conjunto de métodos.

Esta Capa contiene los servidores de las bases de datos que se utilizan para almacenar la información de las aplicaciones con el motor de base de datos relacional, independiente de la aplicación y escalable en función de las necesidades de la empresa final.

b) Capa del Servidor de Aplicaciones o de Negocio

Contiene los procesos que se ejecutan en las aplicaciones. Esta capa tiene el acceso a los datos. La conexión con la base de datos, lo que asegura la posibilidad de intercambiar el servidor de base de datos sin afectar a la aplicación. Los servicios de esta capa son encapsulados en tres tipos de componentes.

Toda aplicación tiene código para implementar reglas de negocios, procesos relacionados a los datos o cálculos y otras actividades relativas a los negocios. Colectivamente este código es considerado para formar la capa de negocios. Otra vez, uno de los principios del diseño lógico cliente/servidor, la lógica de negocios debe mantenerse separada de la capa de presentación y de los servicios de datos. Esto no significa necesariamente que la lógica de negocios está en cualquier parte, por el contrario, esta separación es en un sentido lógico.

En esta Capa se debe lograr la separación de la Capa de presentación y la Capa de datos, el cual beneficiara en flexibilidad y ayudara a que se pueda almacenar la lógica de negocios.

Los objetos de negocios son diseñados para reflejar o representar los negocios. Ellos se convierten en un modelo de sus entidades de negocios e interrelaciones. Esto incluye tanto objetos físicos como conceptos abstractos.

No toda la lógica de negocio es la misma. Alguna lógica de negocio es un proceso intensivo de datos, requiriendo un eficiente y rápido acceso a la base de datos. Otras no requieren un frecuente acceso a los datos, pero es de uso frecuente por una interface de usuario robusta para la validación en la entrada de campos u otras interacciones de usuarios.

c) Capa de Presentación

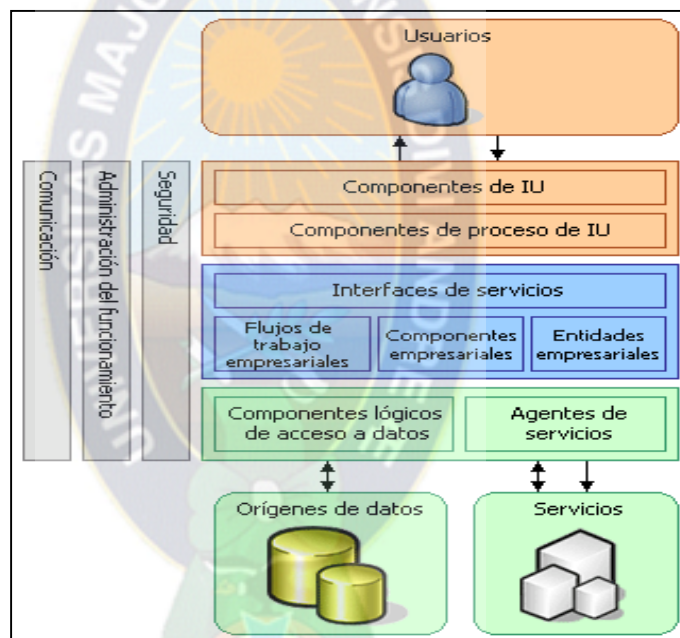
La capa de Presentación provee su aplicación con una interface de usuario (IU). Aquí es donde su aplicación presenta información a los usuarios y acepta entradas o respuestas del usuario para usar por su programa. Idealmente, la IU no desarrolla ningún procesamiento de negocios o reglas de validación de negocios. Por el contrario, la IU debería relegar sobre la capa de negocios para manipular estos asuntos. Esto es importante, especialmente hoy en día, debido a que es muy común para una aplicación tener múltiples IU, o para sus clientes o usuarios, que le solicitan que elimine una IU y la reemplace con otra. Una de las mayores dificultades y factores importantes cuando desarrollamos aplicaciones cliente/servidor es mantener una separación completa entre la presentación, la lógica de negocios y los servicios de datos. Es muy tentador para los

desarrolladores mezclar una o más capas; poniendo alguna validación u otro proceso de negocios dentro de la capa de presentación en vez de en la capa de negocios.

Esta Capa contiene los procesos que permiten la interacción de la aplicación con el mundo exterior, mediante aplicaciones de recogida y muestra de datos a usuarios mediante aplicaciones de intercambios de datos con otras aplicaciones.

Y en un marco aplicativo de la Arquitectura en tres capas dentro de lo que es la tecnología .net se puede visualizar en la Figura 2.15.

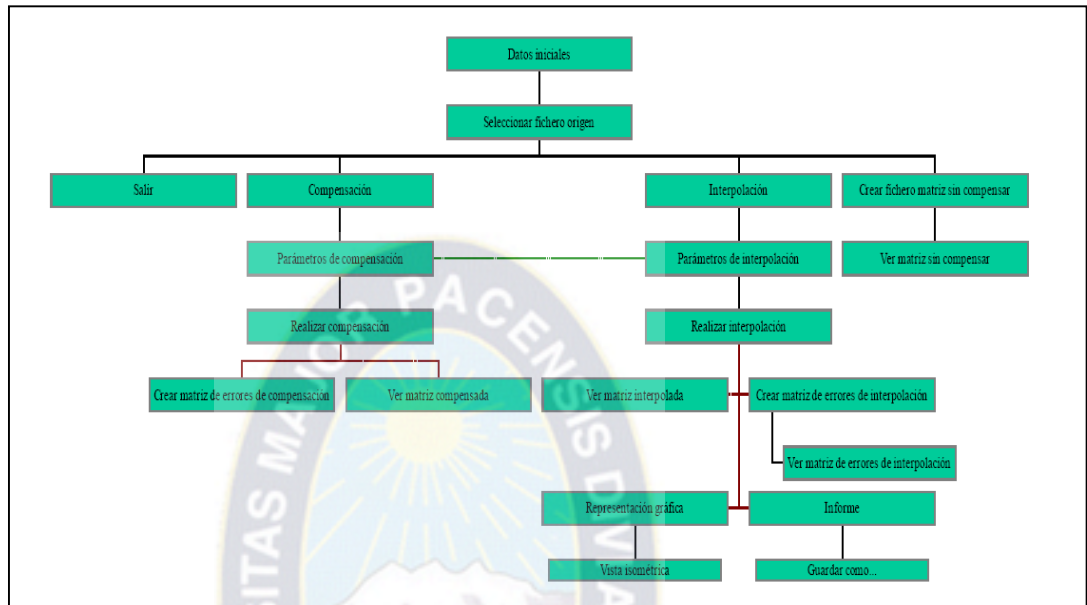
Figura 2. 15 Arquitectura para Aplicaciones.net



Estructura del Programa

Una vez definido el diseño a seguir para el desarrollo del programa, se presenta los resultados obtenidos que se consideran suficiente para satisfacer los requisitos buscados y se debe representas en un esquema así como la Figura 2.16.

Figura 2. 16 Ejemplo de Estructura del Programa



Fuente: [CAD/CAM, 2000]

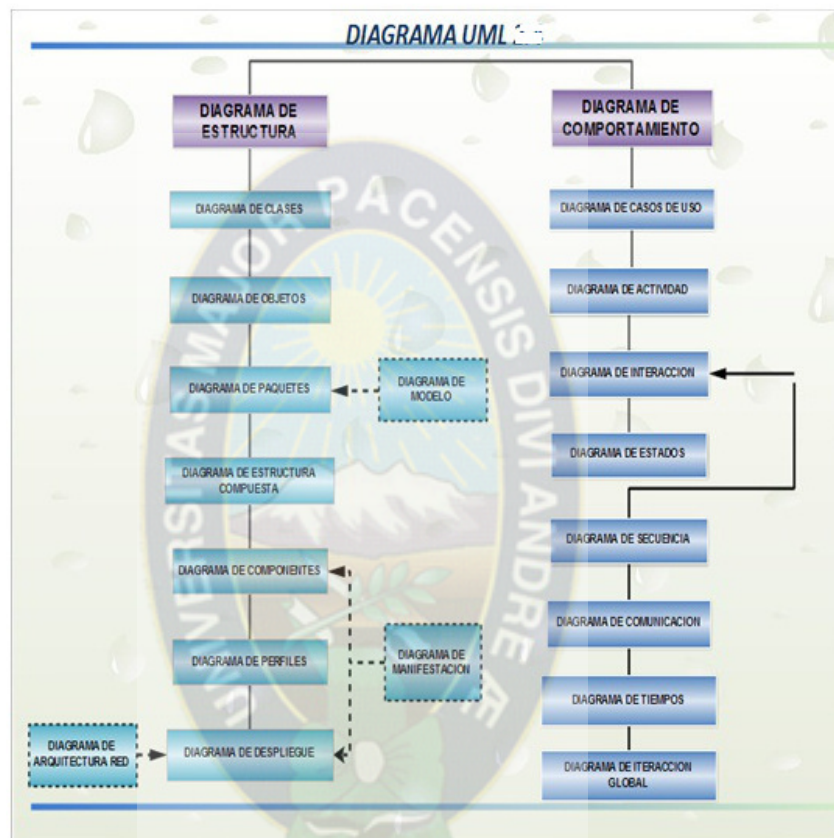
2.5. Lenguaje Unificado de Modelado o UML

El Lenguaje Unificado Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación.

Los objetivos de UML son muchos, pero se puede sintetizar sus funciones:

- Visualizar, UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender;
- Especificar, UML permite especificar cuáles son las características de un sistema antes de su construcción;
- Construir, a partir de los modelos especificados se pueden construir los sistemas diseñados;
- Documentar, los propios elementos gráficos sirven como documentación del Sistema desarrollado que pueden servir para su futura revisión.

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. UML ofrece nueve diagramas en los cuales modelar sistemas.



- **Diagramas de estructura** mostrar la **estructura estática** del sistema y sus partes en la abstracción diferentes y **niveles de** aplicación y cómo están relacionadas entre sí. Los elementos de un diagrama de estructura representan los conceptos significativos de un sistema, y pueden incluir mundo abstracto, real y conceptos de aplicación.
- **Diagramas de comportamiento** muestran el **comportamiento dinámico** de los objetos en un sistema, que puede ser descrita como una serie de cambios en el sistema con el **tiempo**.

Diagramas de Estructura.- Muestra la estructura estática del sistema y sus partes en la abstracción diferentes y niveles de aplicación. Los diagramas de estructura no están utilizando los conceptos del tiempo relacionados sin embargo se pueden clasificar:

- **Diagrama de clases.-** es un diagrama que describe la estructura de un sistema en el nivel de los clasificadores
- **Diagrama de objetos.-** se define en el obsoleto UML 1.4.2 se define como un grafico de los casos incluso los objetos y valores de datos de un diagrama de objetos.
- **Diagrama de Paquetes.-** muestra los paquetes y las relaciones entre los paquetes
- **Diagrama de modelo.-** Es una diagrama de la estructura auxiliar que muestra cierta abstracción o punto de vista específico de un sistema para describir aspectos lógicos del sistema.

Diagramas de Comportamiento.- Muestran el comportamiento dinámico de los objetos en un sistema que puede ser descrita como una serie de cambios en el sistema con el tiempo.

- **Casos de uso.-** Son esquemas de comportamiento utilizados para describir un conjunto de acciones de algún sistema, que puede realizarse con mas objetos del sistema.
- **Diagrama de actividad.-** Muestra la secuencia y las condiciones para la coordinación de las conductas de menor nivel

Los diagramas de interacción incluyen varios tipos de diagramas:

- **Diagrama de secuencia** es el tipo más común de los diagramas de interacción, que se centra en el intercambio de mensajes entre líneas de vida (los objetos).
- **Diagrama de la Comunicación** (anteriormente conocido como Diagrama de colaboración) es una especie de diagrama de interacción, que se centra en la interacción entre líneas de vida, donde la arquitectura de la estructura interna y cómo esto se corresponde con el paso de mensajes es fundamental. La secuencia de mensajes se da a través de una secuencia de numeración de esquema.
- **Diagrama de interacción** resumen define las interacciones a través de una variante de los diagramas de actividad de una manera que promueve la visión general del flujo de control. Los diagramas de interacción visión general se centran en la visión general del flujo de control, donde los nodos son las interacciones o usos de interacción. Las líneas de vida y los mensajes no aparecen en este nivel general.

- **Diagramas de tiempo** se utilizan para mostrar las interacciones que un objetivo principal del diagrama es la razón sobre el tiempo. Diagramas de tiempo se centran en las condiciones cambiantes dentro y entre las líneas de vida a lo largo de un eje de tiempo lineal.

UML es una consolidación de muchas de las notaciones y conceptos más usados orientados a objetos, el mismo que será útil a lo largo del desarrollo del proyecto.
[UMSA 2012]

2.6. Herramientas de Modelado

Rational Software es actualmente conocida como una familia de software de IBM para el despliegue, diseño, construcción, pruebas y administración de proyectos en el proceso desarrollo de software.

Sus productos están centrados en la metodología del Proceso Racional Unificado o [RUP](#) (Rational Unified Process). Entre los productos más conocidos tenemos a:

- Rational Application Developer
- Rational Software Architect (Posee la funcionalidad del producto antes conocido como Rational Rose)
- Rational Portafolio Manager
- Rational Requisite Pro
- Rational Rose

Modelado visual para maximizar el desarrollo de arquitecturas de software

2.6.1 Rational Rose proporciona un conjunto de prestaciones controladas por modelo para el desarrollo de aplicaciones de software. Permite a los clientes y a los diseñadores generar modelos UML (Unified Modeling Language) de arquitecturas de software, necesidades empresariales, activos reutilizables y comunicación de nivel de gestión.

Rational Rose Modeler ofrece una herramienta y un lenguaje de modelado común para simplificar el entorno de trabajo y permitir una creación más rápida de software de calidad.

- Desarrollo más rápido de las aplicaciones: contiene un entorno de modelado visual que permite agilizar el desarrollo de aplicaciones.

- Integración del diseño de aplicaciones con el desarrollo: unifica el equipo del proyecto proporcionando una ejecución y una notación de modelos UML comunes.

Desarrollo más rápido de aplicaciones

- Proporciona un lenguaje y un entorno de modelado visual común para una creación más rápida de software.
- Admite patrones de análisis basados en software reutilizable orientado a objetos.
- Permite la gestión granular y el uso de modelos a través de una característica de componentes de modelo que se puede controlar por separado.

Integración del diseño de aplicaciones con el desarrollo

- Permite utilizar un lenguaje para el ciclo de vida de análisis y desarrollo.
- Se integra con otras herramientas de desarrollo de ciclo de vida de IBM Rational y con cualquier sistema de control de versiones compatible con SCC, incluido IBM Rational ClearCase
- Proporciona modelos e informes de publicación web para mejorar la comunicación en el equipo ampliado.

2.6.2 Enterprise Architect es una herramienta comprensible de diseño y análisis UML, cubriendo el desarrollo de software desde el paso de los requerimientos a través de las etapas del análisis, modelos de diseño, pruebas y mantenimiento. EA es una herramienta multi-usuario, basada en Windows, diseñada para ayudar a construir software robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad.

Algunas de las características de Enterprise Architect:

Crear elementos del modelo UML para un amplio alcance de objetivos. Ubicar esos elementos en diagramas y paquetes. Crear conectores entre elementos. Documentar los elementos que ha creado. Generar código para el software que está construyendo. Realizar ingeniería reversa del código existente en varios lenguajes. Importación/Exportación XMI 2.1. Nuevo motor de Reporte HTML. Perfiles y soporte de Tecnologías.

Velocidad, estabilidad y buen rendimiento

El Lenguaje Unificado de Modelado provee beneficios significativos para ayudar a construir modelos de sistemas de software rigurosos y donde es posible mantener la

trazabilidad de manera consistente. Enterprise Architect soporta este proceso en un ambiente fácil de usar, rápido y flexible.

Trazabilidad de extremo a extremo

Enterprise Architect provee trazabilidad completa desde el análisis de requerimientos hasta los artefactos de análisis y diseño, a través de la implementación y el despliegue. Combinados con la ubicación de recursos y tareas incorporados, los equipos de Administradores de Proyectos y Calidad están equipados con la información que ellos necesitan para ayudarles a entregar proyectos en tiempo. [ECURED, ARCHITECT]

2.6.3 PowerDesigner, herramienta para el análisis, diseño inteligente y construcción sólida de una base de datos y un desarrollo orientado a modelos de datos a nivel físico y conceptual, que da a los desarrolladores Cliente/Servidor la más firme base para aplicaciones de alto rendimiento.

Características de PowerDesigner

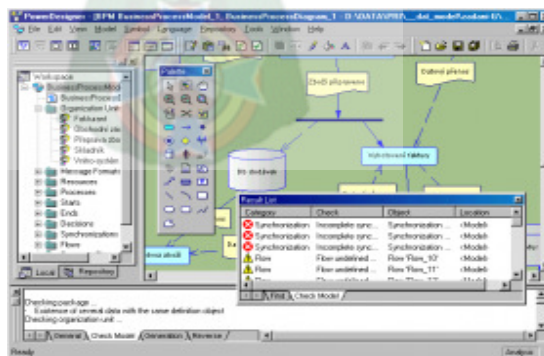
- Es nombrada **La Herramienta Líder en Modelamiento Empresarial**
- Permite a las empresas, de manera más fácil, visualizar, analizar y manipular metadatos, logrando un efectiva arquitectura empresarial de información.
- Brinda un enfoque basado en modelos, el cual permite alinear al negocio con la tecnología de información, facilitando la implementación de arquitecturas efectivas de información empresarial.
- Brinda potentes técnicas de análisis, diseño y gestión de metadatos a la empresa.
- Combina varias técnicas estándar de modelamiento con herramientas líder de desarrollo, como .NET, [Sybase WorkSpace](#), [Sybase Powerbuilder](#), [Java](#) y [Eclipse](#), para darle a las empresas soluciones de análisis de negocio y de diseño formal de base de datos.
- Trabaja con más de 60 bases de datos relacionales.

Beneficios de PowerDesigner

1. **Constituye una Elección Segura:** El compromiso de PowerDesigner con el modelamiento de datos, UML y de negocio, además de estar comprobado en el mercado, lo hace la elección segura para todos los requerimientos de modelamiento. PowerDesigner es el estándar en muchas organizaciones a nivel mundial.
2. **Mejora la Productividad Individual:** El enfoque orientado a modelos de PowerDesigner incorpora una serie de generadores DDL y de código personalizables, y capacidades de ingeniería reversa y sincronización de código,

que reduce significativamente los esfuerzos de creación, mantenimiento y reingeniería manual de código.

3. **Brinda Facilidad de Uso Gráfica:** La interfaz gráfica es altamente personalizable, hace que las tareas comunes sean muy fáciles y le da el poder a los usuarios avanzados de tener acceso rápido a todas las funciones.
4. **Alinea el Negocio con el Área de Tecnología:** Facilita el alineamiento del negocio con el área de tecnología a través de técnicas de colaboración en grupo.
5. **Mejora la Productividad en Grupo:** Brinda a todos los modeladores un ambiente ideal para compartir recursos a través de un repositorio de meta-datos único, completo y seguro para todos los tipos de modelos.
6. **Documenta los Sistemas Existentes:** Adopta una mayor colaboración a nivel empresarial a través de generación de reportes flexible y basada en asistentes, o RTF / HTML multi-modelo.
7. **Brinda Soporte Abierto:** Permite el entendimiento de sistemas heterogéneos con el soporte a los principales estándares de lenguajes de desarrollo, XML, base de datos y procesos, con una sola herramienta e infraestructura.
8. **Es Altamente Personalizable:** Puede ser fácilmente "programado" para asegurar los estándares y prácticas corporativas o legales, a través del soporte a "scripts" VB, una interfaz COM completamente programable, un meta-modelo personalizable y un API totalmente documentado.
9. **Reduce el Impacto del Cambio:** Reduce significativamente el costo y tiempo al implementar cualquier cambio a través de una vista exacta, bi-direccional y multi-modelo para análisis de impacto que integra todos los modelos de requerimientos, análisis, base de datos y aplicación.



Interfaz Gráfica de PowerDesigner
[ECURED,2012]

2.7. Pruebas al Sistema

El objetivo de la fase de pruebas de un sistema es el de detectar todo posible malfuncionamiento antes de que entre en producción. Un error detectado en el laboratorio puede ser costoso de reparar; pero siempre es peor que el error le aparezca al usuario final.

En esta idea, una batería de pruebas será de mayor calidad cuantos menos errores queden por descubrir tras haberla pasado. Y, viceversa, si un programa aún tiene muchos fallos tras haber superado una batería de pruebas, diremos que esta batería es de poca calidad.

Si pudiéramos probar un programa con todos los posibles datos de entrada, tendríamos una batería de pruebas perfecta, pues no hay lugar para las sorpresas. Lamentablemente, casi nunca es posible probar con todos los casos. En consecuencia necesitamos un criterio para elegir qué casos probamos.

Para la realización de pruebas se recomienda lo siguiente:

- En primer lugar realizar pruebas de caja negra;
- Superadas las pruebas anteriores, realizar pruebas de caja blanca complementarias;
- Y finalmente terminar con pruebas adicionales de condiciones.

a) Pruebas de caja negra

Las **pruebas de caja negra** se limitan a que el tester pruebe con “*datos*” de entrada y estudie como salen, **sin preocuparse de lo que ocurre en el interior.**

b) Pruebas de caja blanca

Pruebas se conducen a nivel de interfaz (análisis de fronteras, clases de equivalencia).

c) Prueba de condiciones

Este es un complemento de las anteriores pruebas, en donde se puede identificar la cobertura elevada de sentencias ya que se puede tener confusiones con las ramas condicionales.

2.8. Herramientas Informáticas

Para el desarrollo y construcción del sistema se tomara en cuenta el software necesario para el desarrollo del Sistema, tecnologías de desarrollo para el aspecto del sistema, y herramientas para conceptualizar el Desarrollo del Proyecto.

El software o lenguaje de programación que se utilizara en el presente proyecto son los siguientes:

- Microsoft Visual C# 2008 (lenguaje de programación), es una herramienta de desarrollo de software sencilla de utilizar e ideal para la construcción de Sistemas.

del proceso de desarrollo, ya que la calidad del primero va a depender, entre otros aspectos, de ésta. Sin un buen proceso de desarrollo es casi imposible obtener un buen producto.

- Pero la calidad del producto software se ASP.net, es un framework para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y servicios web XML.
- AJAX 2.0, acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.
- Framework 3.5 un **framework**, en el desarrollo de software, es una estructura de soporte definida, mediante la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto, representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.
- SQL Server 2008 (Gestor de Base de Datos), es un herramienta que ayuda a gestionar, diseñar, administrar base de datos a través de los componentes que tiene.
- Reporting Service 2008, es una plataforma de reportes basada en servidores, la misma que puede ser empleada para crear y administrar reportes tabulares, de matrices, gráficos y de libre formato, la información de estos reportes pueden provenir de diferentes orígenes de datos. Los reportes que se definen pueden ser administrados a través de una conexión basada en Web.
- Windows 2008 Server R2 (Servidor), Windows Server 2003 R2 es un sistema operativo desarrollado por Microsoft, este facilita la extensión de conectividad y control a identidades, ubicaciones, datos y aplicaciones en toda su empresa

- Visio 2010 (Para el desarrollo de Diagramas), es un software desarrollado por la empresa Microsoft, el mismo que servirá al momento de construir y elaborar los diagramas correspondientes al Sistema;

2.9 Métricas y Calidad de Software

A la hora de definir la calidad del software se debe diferenciar entre la calidad del producto software y la calidad del proceso de desarrollo de éste (calidad de diseño y fabricación). No obstante, las metas que se establezcan para la calidad del producto van a determinar los objetivos a establecer de calidad diferencia de la calidad de otros productos de fabricación industrial, ya que el software tiene sus propias características específicas:

- El software es un producto mental, no restringido por las leyes de la Física o por los límites de los procesos de fabricación. Es algo abstracto, un intangible;
- Se desarrolla, no se fabrica. El coste está fundamentalmente en el proceso de diseño, no en la posterior producción en serie, y los errores se introducen también en el diseño, no en la producción;
- Los costes del desarrollo de software se concentran en las tareas de Ingeniería, mientras que en la fabricación clásica los costes se acentúan más en las tareas de producción;
- El software no se deteriora con el tiempo. No es susceptible de los efectos del entorno y su curva de fallos es muy diferente de la del hardware. Todos los problemas que surjan durante el mantenimiento estaban allí desde el principio y afectan a todas las copias del mismo; no se generan nuevos errores;
- Es artesanal en gran medida. El software, en su mayoría, se construye a medida, en vez de ser construido ensamblando componentes existentes y ya probados, lo que dificulta aún más el control de su calidad;
- El mantenimiento del software es mucho más complejo que el mantenimiento del hardware. Cuando un componente del hardware se deteriora se sustituye por una pieza de repuesto, pero cada fallo en el software implica un error en el diseño o en el proceso mediante el cual se tradujo el diseño en código máquina ejecutable;
- Es engañosamente fácil realizar cambios sobre un producto software, pero los efectos de estos cambios se pueden propagar de forma explosiva e incontrolada;

- Como disciplina, el desarrollo de software es aún muy joven, por lo que las técnicas de las que dispone aún no están perfeccionadas;
- El software con errores no se rechaza. Se asume que es inevitable que el software presente algunos errores de poca importancia.

Se medirán los siguientes puntos de calidad de acuerdo a los factores de calidad de la **ISO 9126**, entre los factores señalados son los siguientes:

a) Confiabilidad, es el grado con el que responde el sistema bajo las condiciones definidas durante un intervalo de tiempo. Para la medición de la misma se toma la siguiente fórmula:

$$R_i(t) = e^{-\lambda_i * t}$$

Donde:

$R(t)$, es la función de confiabilidad de un componente en el tiempo t .

λ , tasa constante de fallo.

t , Periodo de operación de tiempo.

Teorema 1, si n componente, funcionan independientemente conectados en serie y el i -ésimo componente tiene confiabilidad $R_i(t)$, entonces la confiabilidad total esta dado por:

$$R(t) = R_1(t) * R_2(t) * R_3(t) * \dots * R_n(t)$$

Teorema 2, si n componentes funcionar independientemente y actúan en paralelo y el i -ésimo componente tiene confiabilidad $R_i(t)$, entonces la confiabilidad total esta dad por:

$$R(t) = 1 - [1 - R_1(t)] * [1 - R_2(t)] * [1 - R_3(t)] * \dots * [1 - R_n(t)]$$

b) Funcionalidad, es el conjunto de atributos que permiten que un sistema provea ciertas propiedades y funciones, satisfaciendo las necesidades del usuario final.

Para el cálculo de la funcionalidad se utilizara la ecuación del punto función que es la siguiente:

$$PF = \text{Cuenta Total} * (A + (E * \text{Sum}(F_i)))$$

Donde:

PF, funcionalidad estimada del sistema.

Cuenta Total, representa la suma de todas las entradas.

A, representa la portabilidad subjetiva estimada de la confiabilidad

E, error estimado del sistema

Sum(F_i), valores de ajuste de complejidad

Para la obtención de la cuenta total se hará de uso de la Tabla 2.1.

Tabla 2.1 Calculo de Cuenta Total

Parámetro de medida	Cuenta	Ponderación	Total
Número de entradas de Usuario	#	#	#
Número de Salidas de Usuario	#	#	#
Número de Peticiones de usuario	#	#	#
Numero de Archivos	#	#	#
Número de Interfaces externas	#	#	#
Cuenta Total			#

Los ajustes de complejidad se realizan de acuerdo a las siguientes respuestas que se encuentran en la Tabla 2.2 donde cada campo tendrá una calificación, dicho resultado serán los F_i de la ecuación de punto función.

Tabla 2.2 Ajustes de Complejidad

Importancia	0%	20%	40%	60%	80%	100%
Escala	No Influyente	Incidental	Moderado	Medio	Significativo	Esencial
Factor	0.	1	2	3	4	5

1. ¿Requiere el sistema copias de seguridad y de recuperación fiables?						
2. ¿Se requiere comunicación de datos?						
3. ¿Existen funciones de procesamiento distribuido?						
4. ¿Es crítico el rendimiento?						
5. ¿Se ejecutara el sistema en un entorno operativo existente y fuente utilizado?						
6. ¿Requiere el sistema entrada de datos interactivo?						
7. ¿Requiere la entrada de datos interactivos que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variables utilizadas?						
8. ¿Se actualizan los archivos maestros de forma interactiva?						
9. ¿Son complejas las entradas, las salidas o las operaciones?						
10. ¿Es complejo el procesamiento Interno?						
11. ¿Se ha diseñado el código para ser reutilizable?						
12. ¿Están incluidos en el diseño la conversión y la instalación?						
13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?						
14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?						

c) **Facilidad de Uso**, este se determina midiendo el grado de esfuerzo para que un usuario aprenda y use cierto software, además de determinar que productivos son los usuarios con ese sistema y cuanta ayuda necesitaran.

Para la obtención de la misma se calculara mediante la Tabla 2.3, analizando y obteniendo un porcentaje por cada casilla y para finalizar un porcentaje promedio, el mismo que indicara que grado de facilidad de uso tiene el sistema.

Tabla 2.3 Tabla del cálculo para la Facilidad de Uso

Usuarios	Facilidad de Comprensión	Facilidad de Aprendizaje	Facilidad de Operación
Usuario_1			
Usuario_2			
.....			
Usuario_n			
Promedio			

d) **Facilidad de Mantenimiento**, se representa en el esfuerzo requerido para implementar cambios, y para la medición se aplicara la métrica de tiempo medio TMC el mismo que consiste en el cálculo mediante la siguiente ecuación.

$$TMC = TA + TD + TI + TP$$

Dónde:

TA, es el tiempo en el que se tarda analizar la petición de cambio.

TD, es el tiempo empleado en diseñar una modificación adecuada

TI, es el tiempo necesario para implementar el cambio

TP, es el tiempo empleado en probar y distribuir el cambio a todos los usuarios

e) **Portabilidad**, es el conjunto de características que determinan la capacidad del software para ser transferido de un entorno de operación a otra.

El grado de portabilidad esta dado por la siguiente ecuación:

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

Dónde:

GP, es el grado de portabilidad

CT, costo de transportar

CRD, costo de re desarrollo

Se debe tomar en cuenta lo siguiente:

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

Si $GP = 0$, la portabilidad es perfecta.

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





CAPITULO 3 MARCO APLICATIVO

3.1. Introducción

El presente capítulo describe el desarrollo del Sistema de Servicio de Archivo y Kardex mediante la metodología del Proceso Unificado de desarrollo de software.

Durante el desglose del mismo se tomara a consideración lo siguiente: el Modelado de Negocio del Gobierno Autónomo Municipal de La Paz para conocer los requerimientos y necesidades de las diferentes instancias que la componen ante la labor que desempeña el resguardo de la documentación, específicamente la unidad de Kardex de Personal para la cual se desea desarrollar un sistema que cumpla las funciones forma sistémica y sistematizada. Se tomara en cuenta para definir los alcances del sistema un análisis de los requerimientos que se manifiestan por la unidad, tendremos que definir posteriormente un diseño que contemple la arquitectura del sistema, y así finalmente mediante pruebas lleguemos al producto deseado para su puesta en producción.

3.2. Fase de Inicio

3.2.1. Modelado del Negocio

El modelado del negocio nos permitirá comprender los procesos de la Unidad de Kardex de Personal del Gobierno Autónomo Municipal de La Paz, los mismos serán identificados mediante la identificación de los casos de uso como también de los actores más representativos para el sistema.

Kardex de personal del Gobierno Autónomo Municipal de La Paz de acuerdo a su estructura organizacional, por el número de funcionarios que cuenta y por el movimiento de estos, hace que se tenga muchos expedientes que se debe administrar.

Los actores principales que cuenta Kardex de Personal es la siguiente.

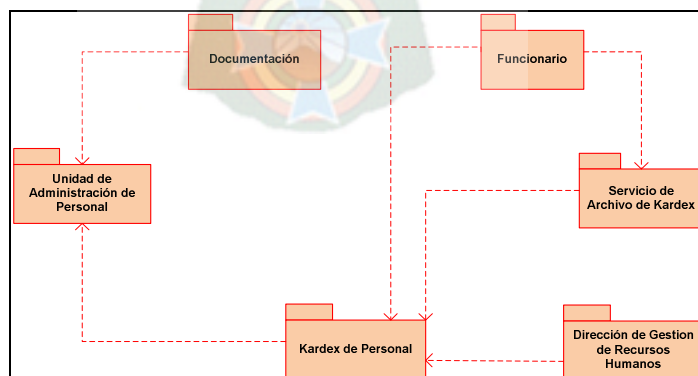
- Funcionarios públicos que tienen derechos y obligaciones normados por las diferentes leyes vigentes;
- La Jefatura de Kardex de Personal se encarga de las siguientes tareas:
 - Controlar el sistema de registro de las filiaciones de los funcionarios de la institución.
 - Realizar el control de las papeletas de vacaciones
 - Verificar los informes y el cálculo de beneficios sociales
 - Custodiar los file personal de los funcionarios

- Elevar informes cotidianos sobre los antecedentes del personal
 - Emisión de certificados de trabajo de personal activo y pasivo
 - Legalización de documentos del personal a solicitud de órdenes judiciales
 - Recepción de documentación de funcionarios
 - Registro en sistema de Estudios realizados beneficiarios
- La Jefatura de Remuneraciones se encarga de hacer las siguientes tareas:
- Registro de contratos en base a solicitud de la unidad ejecutora;
 - Emisión y registro de comisiones a tiempo completo, a medio tiempo, a tiempo indefinido, por viajes, por representación o por capacitación y memorándums de sanción.
- La Unidad de Centro de Capacitación de administración de Personal quienes elaboran sus propios memorándums de capacitación.

Toda esta documentación es la que se va almacenando en cada file de cada funcionario municipal.

Para un mejor análisis es mejor conocer el proceso y procedimientos que se efectúan dentro de la institución y para esto se realizó los diagramas de casos de uso iniciales y el modelo del negocio (Ver Figura 3.1).

Figura 3. 1 Diagrama del Modelado del Negocio

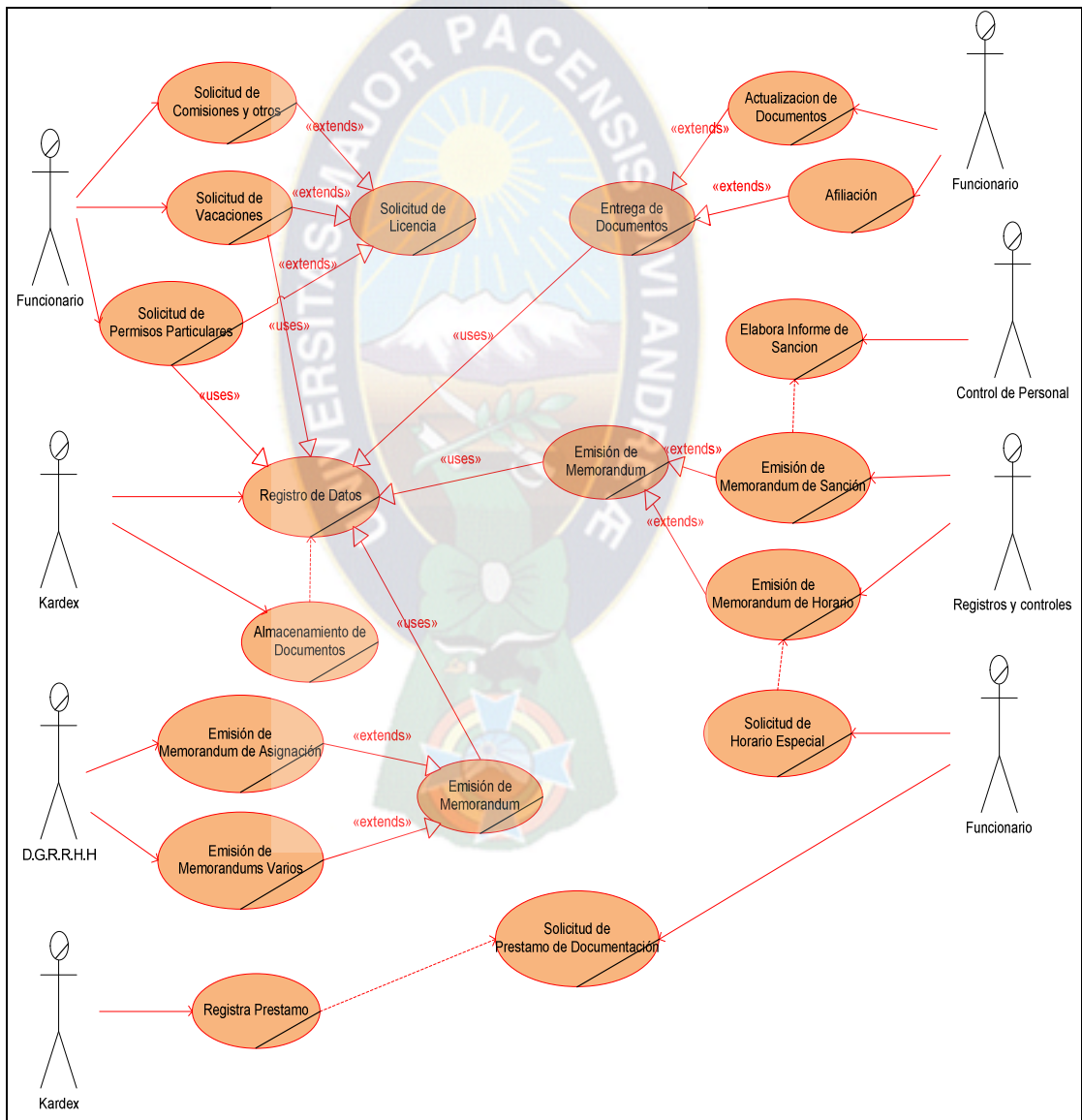


Fuente [Elaboración Propia]

3.2.2. Modelado de Casos de uso del Negocio

Describe los procesos de negocio del Gobierno Autónomo Municipal de La Paz en términos de casos de uso y actores del negocio que corresponde desde los procesos del negocio al cliente (Ver figura 3.2).

Figura 3. 2 Modelado de casos de uso de Negocio



Fuente: [Elaboración Propia]

Descripción de los Casos de Uso del Modelo del Negocio

Tabla 3.4 Solicitud de Comisiones y Otros

CU-01: Solicitud de Comisiones y Otros	
Frecuencia de Uso:	Siempre
Actores:	Funcionario
Propósito:	Solicitud de Baja Médica, Justificativo Médico, Solicitud de permiso por cumpleaños, natalidad y defunción.
Descripción:	El funcionario podrá hacer la solicitud de acuerdo a la necesidad o derecho.
Pre-requisitos:	El funcionario debe contar con una asignación vigente.
Escenario Principal de Éxito:	a. Documentación firmada.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Solicitud aceptada

Tabla 3.2 Solicitud de Vacaciones

CU-02: Solicitud de Vacaciones	
Frecuencia de Uso:	Siempre
Actores:	Funcionario
Propósito:	Solicitud de Vacaciones
Descripción:	El funcionario podrá programar sus vacaciones y hacer uso de las mismas con las autorizaciones respectivas.
Pre-requisitos:	El funcionario debe contar con saldo de vacaciones (días de vacación).
Escenario Principal de Éxito:	a. Contar con días de vacación b. Documentación firmada.

Escenario de extensiones alternativas:	En caso de que el funcionario se esté retirando se hace un cálculo de duodécimas de vacación.
Post Condiciones:	Solicitud aceptada

Tabla 3.3 Solicitud de Permisos Particulares

CU-03: Solicitud de Permisos Particulares	
Frecuencia de Uso:	Siempre
Actores:	Funcionario
Propósito:	Solicitud de Permiso Particular
Descripción:	El funcionario podrá hacer uso cualquier momento de este tipo de permisos.
Pre-requisitos:	El funcionario debe contar con saldo de vacaciones (días de vacación).
Escenario Principal de Éxito:	<ul style="list-style-type: none"> a. Contar con días de vacación b. Documentación firmada.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Solicitud aceptada

Tabla 3.4 Actualización de Documentos

CU-04: Actualización de Documentos	
Frecuencia de Uso:	Siempre
Actores:	Funcionario
Propósito:	Actualizar documentación de acuerdo al incremento en la hoja de vida del funcionario.
Descripción:	El funcionario podrá actualizar cualquier momento sus documentos correspondientes a su hoja de vida.
Pre-requisitos:	El funcionario debe contar con una asignación vigente.

Escenario Principal de Éxito:	a. Contar con el original y fotocopia de la documentación que será actualizada
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Actualización correcta

Tabla 3.5 Afiliación

CU-05: Afiliación	
Frecuencia de Uso:	Se la realiza cuando el funcionario es nuevo
Actores:	Funcionario
Propósito:	Afiliación a Kardex.
Descripción:	El funcionario debe afiliarse con toda la documentación que se cuenta.
Pre-requisitos:	El funcionario debe contar con una asignación vigente.
Escenario Principal de Éxito:	a. Contar con el original y fotocopia de la documentación.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Afiliación correcta

Tabla 3.6 Elabora informe de Sanción

CU-06: Elabora Informe de Sanción	
Frecuencia de Uso:	Siempre
Actores:	Control de Personal
Propósito:	Informar sobre faltas, abandonos, y otras sanciones de acuerdo a reglamento.
Descripción:	El control de personal de acuerdo a la información de los relojes

	biométricos, información de los jefes de unidad podrá realizar el informe de sanción correspondiente.
Pre-requisitos:	El control de personal debe contar con la información actualizada de los relojes biométricos.
Escenario Principal de Éxito:	a. Contar con la información correcta y oportuna.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Elaboración de Informe

Tabla 3.7 Emisión de Memorándum de Sanción

CU-07: Emisión de Memorándum de Sanción	
Frecuencia de Uso:	Siempre
Actores:	Unidad de Registros y Controles
Propósito:	Sancionar de acuerdo a reglamento
Descripción:	Se emitirá el memorándum de acuerdo al informe de control de personal.
Pre-requisitos:	Se debe contar con el informe del control de personal correspondiente.
Escenario Principal de Éxito:	a. Contar con la información correcta y oportuna.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Emisión de Memorándum

Tabla 3.8 Emisión de Memorándum de Horario

CU-08: Emisión de Memorándum de Horario	
Frecuencia de Uso:	Siempre

Actores:	Unidad de Registros y Controles
Propósito:	Emisión de Memorándum de Horario especial.
Descripción:	Se emitirá el memorándum de acuerdo a la solicitud y la documentación presentada por el funcionario.
Pre-requisitos:	El funcionario debe contar con la documentación respaldatoria para la solicitud de horario especial
Escenario Principal de Éxito:	a. Contar con los respaldos correspondientes.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Emisión de Memorándum

Tabla 3.9 Solicitud de Horario Especial

CU-09: Solicitud de Horario Especial	
Frecuencia de Uso:	A veces
Actores:	Unidad de Registros y Controles
Propósito:	Emisión de Memorándum de Horario especial.
Descripción:	<p>El funcionario solicita cambio de horario o un horario especial de acuerdo a ley y reglamentos internos de la institución. Los horarios que se pueden solicitar son los siguientes:</p> <ul style="list-style-type: none"> • Estudiantil o Universitario • Lactancia • Docencia <p>Postgrado</p>
Pre-requisitos:	El funcionario debe tener el VoBo y la documentación respectiva para la solicitud de cambio de horario
Escenario Principal de Éxito:	a. Contar con los respaldos correspondientes.

Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Solicitud aceptada

Tabla 3.10 Registro de Datos

CU-10: Registro de Datos	
Frecuencia de Uso:	Siempre
Actores:	Unidad de Kardex de Personal
Propósito:	Registrar la documentación ingresada
Descripción:	De acuerdo a la llegada de documentación se debe registrar los datos referentes a las mismas.
Pre-requisitos:	Debe existir el file del funcionario en cuestión.
Escenario Principal de Éxito:	a. Funcionario afiliado.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Registro correcto de datos

Tabla 3.11 Almacenamiento de Documentación

CU-11: Almacenamiento de Documentación	
Frecuencia de Uso:	Siempre
Actores:	Unidad de Kardex de Personal
Propósito:	Almacenar y guardar la documentación en los files correspondientes.
Descripción:	De acuerdo a la llegada de documentación se debe almacenar la documentación en los files y en la sección correspondiente de acuerdo a

	los documentos.
Pre-requisitos:	Debe existir el file del funcionario en cuestión.
Escenario Principal de Éxito:	a. Funcionario afiliado.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Almacenamiento de Documentación

Tabla 3.12 Emisión de Memorándum de Asignación

CU-12: Emisión de Memorándum de Asignación	
Frecuencia de Uso:	A veces
Actores:	Dirección de Gestión de Recursos Humanos
Propósito:	Emitir Memorándums de asignación o toma de cargo
Descripción:	Se emite memorándums de asignación al inicio de cada gestión o movimiento de personal
Pre-requisitos:	El funcionario debe contar con todos los requisitos para gozar de una asignación
Escenario Principal de Éxito:	a. Debe existir un ítem acéfalo.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Almacenamiento de Documentación

Tabla 3.13 Emisión de Memorándums Varios

CU-13: Emisión de Memorándums Varios

Frecuencia de Uso:	A veces
Actores:	Dirección de Gestión de Recursos Humanos
Propósito:	Emitir Memorándums de asignación comisión, viaje, sanciones, etc.
Descripción:	De acuerdo a solicitudes o resoluciones legales se emite el memorándum de acuerdo al tenor del caso.
Pre-requisitos:	El funcionario debe contar con asignación vigente.
Escenario Principal de Éxito:	a. El funcionario debe estar vigente.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Memorándum emitido correctamente.

Tabla 3.14 Solicitud de Préstamo de Documentación

CU-14: Solicitud de Préstamo de Documentación	
Frecuencia de Uso:	A veces
Actores:	Funcionario
Propósito:	Solicitud de files
Descripción:	Este caso de préstamo preferentemente se da cuando existe casos en proceso jurídico o de análisis legal.
Pre-requisitos:	El funcionario solicitante debe contar con los respaldos correspondientes para la solicitud del file.
Escenario Principal de Éxito:	a. El funcionario solicitante debe estar vigente.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Solicitud aceptada

Tabla 3.15 Registro de Préstamo

CU-15: Registro de Préstamo	
Frecuencia de Uso:	A veces
Actores:	Unidad de Kardex de Personal
Propósito:	Registrar el préstamo
Descripción:	De acuerdo a la salida de file o asignación para revisión se debe registrar los datos del file y la persona solicitante
Pre-requisitos:	El funcionario solicitante debe contar con los respaldos correspondientes para la solicitud del file.
Escenario Principal de Éxito:	a. El funcionario solicitante debe estar vigente.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Préstamo registrado

3.2.3. Requerimientos del Sistema

En la Actualidad en el G.A.M.L.P. cuenta con bastante personal los cuales están clasificados de la siguiente manera: Concejales, Personal del Concejo, Personal del Ejecutivo, para quienes se genera constantemente, documentación que tiene que ser archivada en su respectivo folder como ser contratos, asignaciones de horarios especiales, comisiones, permisos personales, memorándums de llamada de atención, memorándums de capacitación etc., como se detalló en el modelado de negocio.

Con la constante documentación que se obtiene de cada funcionario es que se necesita tener un servicio de control de kardex, que ayude a organizar de una manera eficiente toda la documentación que se genera de cada empleado.

Viendo las necesidades se determinó poder contar con un sistema que pueda ayudarnos a:

- Llevar una descripción, ubicación y capacidad de muebles de almacenamiento;

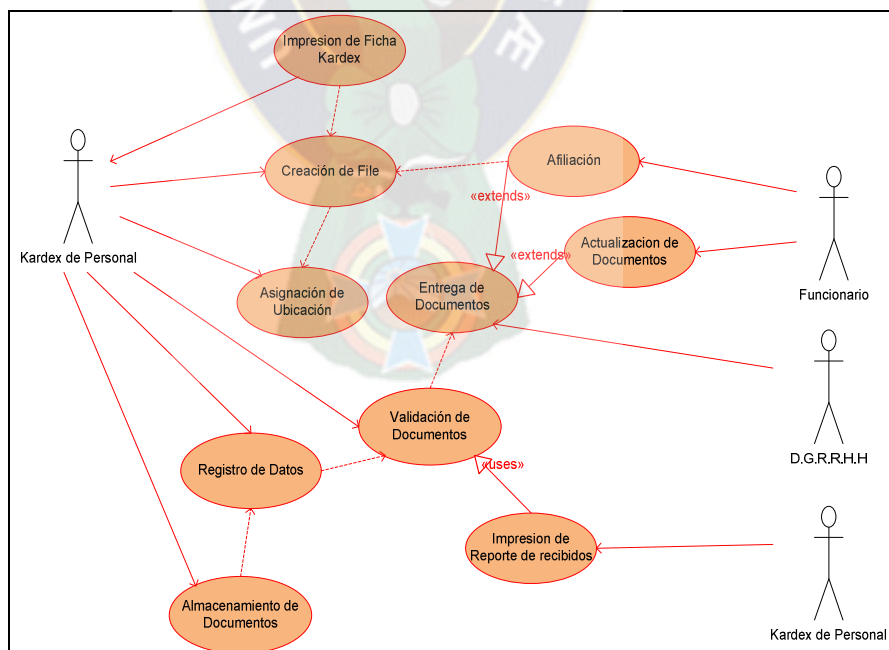
- Identificar la ubicación física el file de cada funcionario en sus respectivo mueble de almacenamiento, tomando en cuenta el tipo de funcionarios (pasivo o activo);
- Llevar un control en los préstamos de file de personal que se realiza a diferentes unidades como ser directores o asesores jurídicos, quien es el solicitante del préstamo, el responsable del préstamo, la devolución de estos;
- Entrega de file solicitados y emisión de boleta de préstamo;
- Almacenar de forma rápida toda documentación que llegue a kardex, y contar con un registro de toda la documentación.
- Contar con un módulo que nos permita hacer el seguimiento a toda la documentación emitida por la Dirección de Gestión der Recursos Humanos y que estos deben ser custodiados por la unidad de Kardex de personal.

3.3. Fase de Elaboración

3.3.1 Diagramas de Casos de uso

Caso de Uso: Registro de File, Ver Figura 3.3

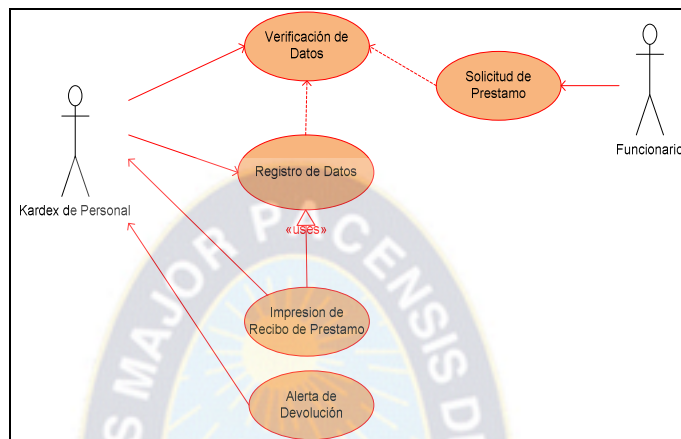
Figura 3. 3 Registro de File



Fuente: [Elaboración Propia]

Caso de Uso: Registro de Préstamos, Ver Figura 3.4

Figura 3. 4 Registro de Préstamos



Fuente: [Elaboración Propia]

Caso de Uso: Registro de File

Tabla 3.16 Afiliación

CU-01: Afiliación	
Frecuencia de Uso:	Se la realiza cuando el funcionario es nuevo
Actores:	Funcionario
Propósito:	Afiliación a Kardex.
Descripción:	El funcionario debe afiliarse con toda la documentación que se cuenta.
Pre-requisitos:	El funcionario debe contar con una asignación vigente.
Escenario Principal de Éxito:	a. Contar con el original y fotocopia de la documentación.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Afiliación correcta

Tabla 3.17 Actualización de Documentos

CU-02: Actualización de Documentos	
Frecuencia de Uso:	Siempre
Actores:	Funcionario
Propósito:	Actualizar documentación de acuerdo al incremento en la hoja de vida del funcionario.
Descripción:	El funcionario podrá actualizar cualquier momento sus documentos correspondientes a su hoja de vida.
Pre-requisitos:	El funcionario debe contar con una asignación vigente.
Escenario Principal de Éxito:	a. Contar con el original y fotocopia de la documentación que será actualizada
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Actualización correcta

Tabla 3.18 Entrega de Documentos

CU-03: Entrega de Documentos	
Frecuencia de Uso:	A veces
Actores:	Dirección de Gestión de Recursos Humanos
Propósito:	Entregar toda documentación generada en la dirección.
Descripción:	Entregar de toda la documentación que se genera en la dirección, es decir todo documento que corresponda a la labor, actividades y desempeño de los funcionarios.
Pre-requisitos:	Toda documentación debe estar debidamente firmada por todos los involucrados.
Escenario Principal de Éxito:	a. Documentación firmada como corresponde.

Éxito:	
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Entrega de documentos

Tabla 3.19 Creación de File

CU-03: Entrega de Documentos	
Frecuencia de Uso:	A veces
Actores:	Kardex de Personal
Propósito:	Crear File tanto físico como virtual.
Descripción:	Una vez revisado los documentos se debe crear el file virtual con toda la documentación presentada, como también se debe crear una carpeta física
Pre-requisitos:	Toda documentación debe ser presentada en fotocopia y original
Escenario Principal de Éxito:	a. Datos correctamente registrados
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Afiliación correcta

Tabla 3.20 Asignación de Ubicación

CU-04: Asignación de Ubicación	
Frecuencia de Uso:	A veces
Actores:	Kardex de Personal

Propósito:	Asignar una ubicación física en el Sistema
Descripción:	Se asignara un lugar físico para el file, y el mismo debe ser registrado en el sistema
Pre-requisitos:	Ubicación debe estar registrada en el sistema para poder ser asignado.
Escenario Principal de Éxito:	a. Ubicación existente
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Asignación correcta

Tabla 3.21 Impresión de Ficha Kardex

CU-05: Impresión de Ficha Kardex	
Frecuencia de Uso:	A veces
Actores:	Kardex de Personal
Propósito:	Impresión de constancia de afiliación
Descripción:	Se asignara un lugar físico para el file, y el mismo debe ser registrado en el sistema
Pre-requisitos:	Ubicación debe estar registrada en el sistema para poder ser asignado.
Escenario Principal de Éxito:	a. Ubicación existente
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Asignación correcta

Tabla 3.22 Validación de Documentos

CU-06: Validación de Documentos	
Frecuencia de Uso:	Siempre
Actores:	Kardex de Personal
Propósito:	Validación de documentos recepcionados y recibidos de la Dirección de Gestión de Recursos Humanos.
Descripción:	Se valida la información recepcionada, preferentemente la información generada por la Dirección de Gestión de Recursos Humanos.
Pre-requisitos:	<ul style="list-style-type: none"> ➤ La información debe ser generada a través del sistema. ➤ Los documentos deben estar debidamente firmados.
Escenario Principal de Éxito:	a. Información correcta y debidamente firmado.
Escenario de extensiones alternativas:	En caso que la documentación no sea emitida por el sistema, existirá otro modulo que permita realizar esa tarea.
Post Condiciones:	Validación correcta

Tabla 3.23 Impresión de Reporte de Recibidos

CU-07: Impresión de reporte de Recibidos	
Frecuencia de Uso:	Siempre
Actores:	Kardex de Personal
Propósito:	Imprimir un reporte de respaldo de toda la documentación recibida.
Descripción:	Una vez recepcionado la documentación, se debe imprimir un reporte de respaldo de lo recibido
Pre-requisitos:	<ul style="list-style-type: none"> ➤ La información debe ser generada a través del sistema. ➤ Los documentos deben estar debidamente firmados.
Escenario Principal de Éxito:	a. Información correcta y cruzada con el reporte.

Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Impresión de reporte

Tabla 3.24 Registro de Datos

CU-08: Registro de Datos	
Frecuencia de Uso:	Siempre
Actores:	Unidad de Kardex de Personal
Propósito:	Registrar la documentación ingresada al Sistema
Descripción:	De acuerdo a la llegada de documentación se debe registrar los datos referentes a las mismas en los files virtuales correspondientes.
Pre-requisitos:	Debe existir el file del funcionario en cuestión.
Escenario Principal de Éxito:	a. Funcionario afiliado.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Registro correcto de datos

Tabla 3.25 Almacenamiento de Documentación

CU-09: Almacenamiento de Documentación	
Frecuencia de Uso:	Siempre
Actores:	Unidad de Kardex de Personal
Propósito:	Almacenar y guardar la documentación en los files correspondientes.
Descripción:	De acuerdo a la llegada de documentación se debe almacenar documentación en los files y en la sección correspondiente de acuerdo a

	los documentos.
Pre-requisitos:	Debe existir el file del funcionario en cuestión.
Escenario Principal de Éxito:	a. Funcionario afiliado.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Almacenamiento de Documentación

Caso de Uso: Registro de Préstamos

Tabla 3.26 Solicitud de Préstamo

CU-01: Solicitud de Préstamo	
Frecuencia de Uso:	A veces
Actores:	Funcionario
Propósito:	Solicitud de préstamo de file
Descripción:	Este caso de préstamo de files se da cuando existe algún proceso o análisis a un funcionario.
Pre-requisitos:	El funcionario solicitante debe contar con los respaldos correspondientes para la solicitud del file.
Escenario Principal de Éxito:	a. El funcionario solicitante debe estar vigente.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Solicitud aceptada

Tabla 3.27 Verificación de Datos

CU-02: Verificación de Datos	
Frecuencia de Uso:	Siempre
Actores:	Kardex de Personal
Propósito:	Verificar la documentación respaldatoria y la información en el sistema
Descripción:	Se debe verificar los datos respaldatorios del funcionario solicitante como también su relación contractual con la institución
Pre-requisitos:	El funcionario solicitante debe contar con los respaldos correspondientes para la solicitud del file.
Escenario Principal de Éxito:	a. El funcionario solicitante debe estar vigente.
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Verificación correcta

Tabla 3.28 Registro de Datos

CU-03: Registro de Datos	
Frecuencia de Uso:	Siempre
Actores:	Unidad de Kardex de Personal
Propósito:	Registrar los datos del préstamo
Descripción:	De acuerdo a la salida de file o asignación para revisión se debe registrar los datos del file y la persona solicitante en el sistema
Pre-requisitos:	El funcionario solicitante debe contar con los respaldos correspondientes para la solicitud del file.
Escenario Principal de Éxito:	b. El funcionario solicitante debe estar vigente.

Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Préstamo registrado

Tabla 3.29 Impresión de recibo de préstamo

CU-04: Impresión de recibo de préstamo	
Frecuencia de Uso:	Siempre
Actores:	Unidad de Kardex de Personal
Propósito:	Imprimir un recibo de préstamo.
Descripción:	Una vez registrado todos los datos correspondientes al préstamo se debe imprimir un comprobante de préstamo tanto para el solicitante como el que realiza el préstamo.
Pre-requisitos:	Información correctamente ingresada.
Escenario Principal de Éxito:	a. Información correcta
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Impresión de comprobante

Tabla 3.30 Alerta de devolución

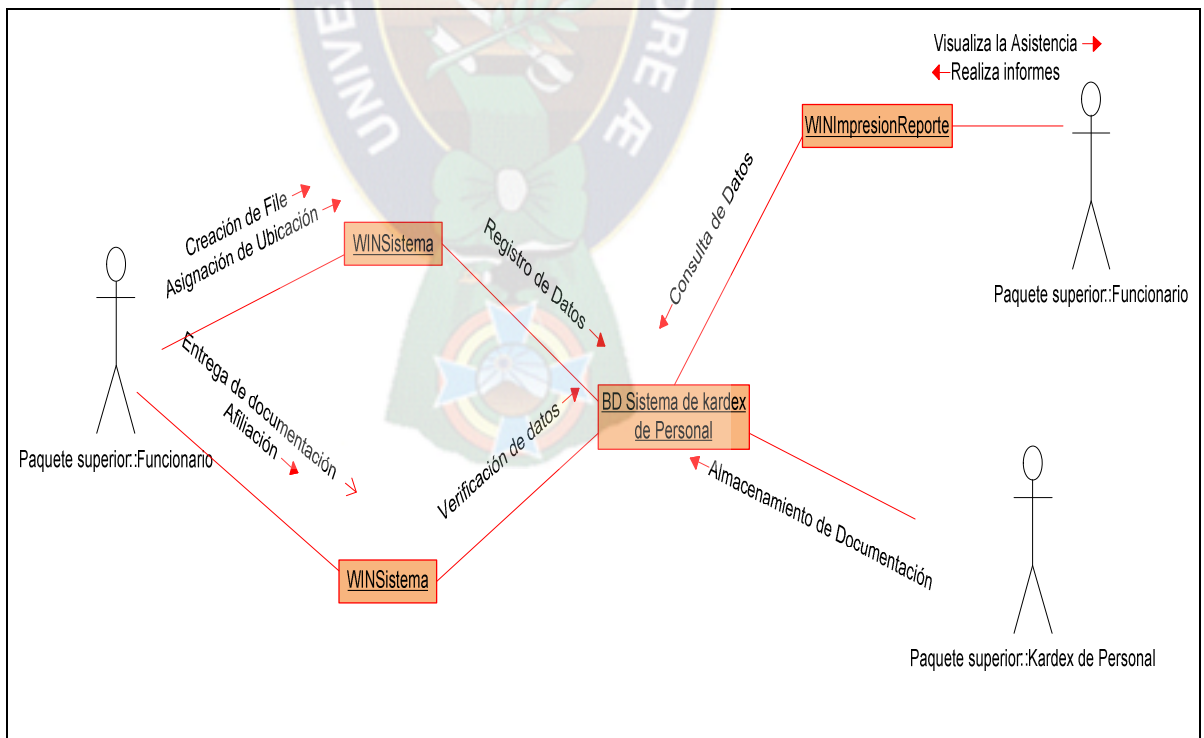
CU-05: Alerta de devolución	
Frecuencia de Uso:	Siempre
Actores:	Sistema de kardex de Personal
Propósito:	Alertar sobre vencimiento de prestamos.
Descripción:	El sistema alertara sobre files prestados que fenecieron las fechas de la

	misma.
Pre-requisitos:	Información ingresada.
Escenario Principal de Éxito:	a. Información correcta
Escenario de extensiones alternativas:	Ninguna
Post Condiciones:	Alerta correcta y temprana

3.3.2. Diagramas de Colaboración

Diagrama de Colaboración: Registro de Files, Ver Figura 3.5

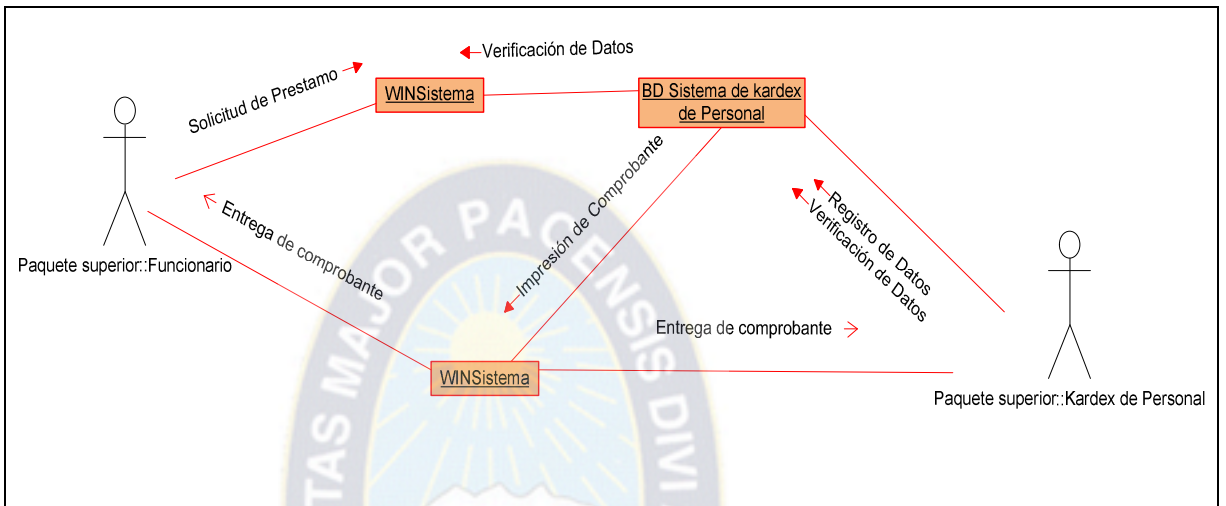
Figura 3. 5 Registro de Files



Fuente: [Elaboración Propia]

Diagrama de Colaboración: Registro de Préstamos, Ver Figura 3.6

Figura 3. 6 Diagrama de Colaboración de Registro de Préstamos



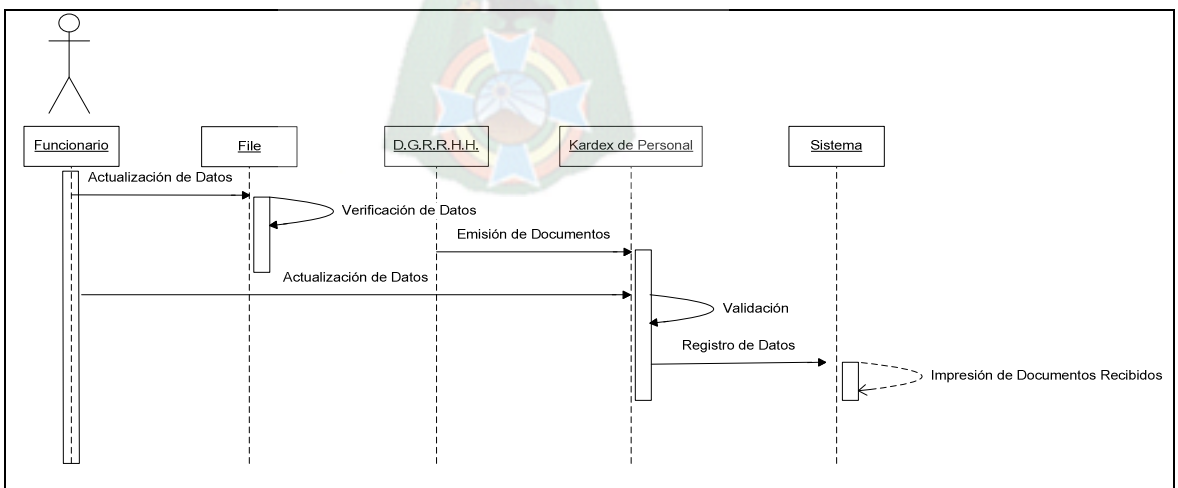
Fuente: [Elaboración Propia]

3.3.3. Diagramas de Secuencia

De acuerdo a los diagramas de secuencia se explicara la interacción y el seguimiento de cada uno de los procesos descritos en los casos de uso.

Diagrama de Secuencia: Registro de File, Ver figura 3.7

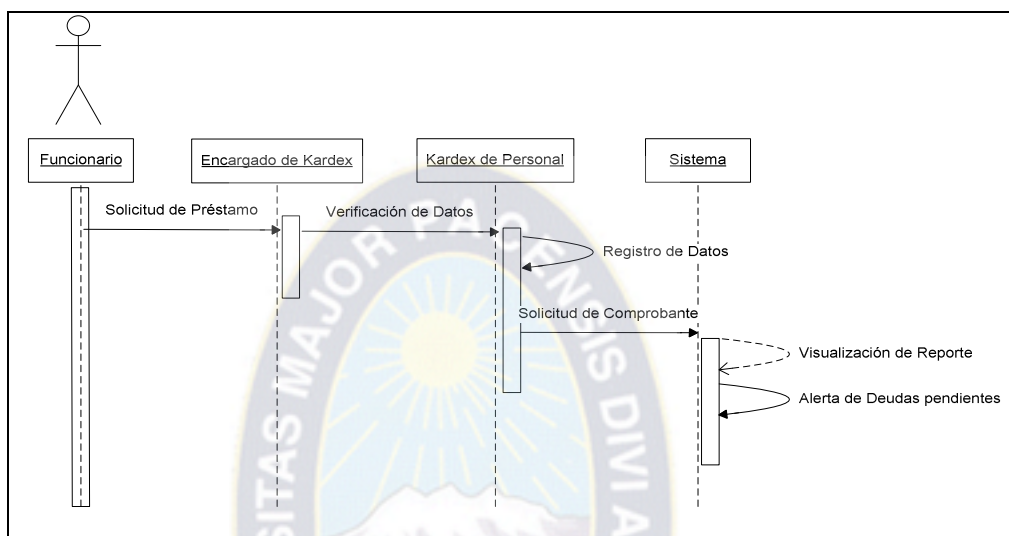
Figura 3. 7 Diagrama de Secuencia de Registro de File



Fuente: [Elaboración Propia]

Diagrama de Secuencia: Registro de Préstamos, Ver Figura 3.8

Figura 3. 8 Diagrama de Secuencia de Registro de File



Fuente: [Elaboración Propia]

3.4. Fase de Construcción

Durante esta fase se preparara el material adicional, es decir las herramientas para el usuario final cumpliendo los requerimientos solicitados y con esto cubrir las nesidades de la Institución.

3.4.1 Requerimientos Finales

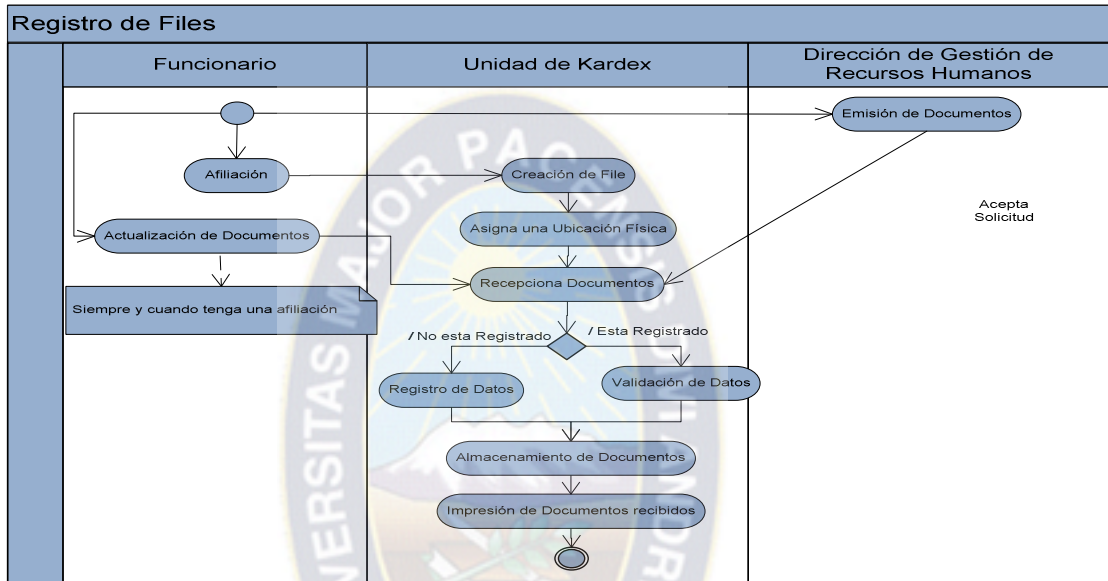
Requerimientos solicitados

- Formas de seguimiento al registro y afiliación de funcionarios;
- Visualización de Files de forma Virtual;
- Registro de préstamos de files;
- Reportes de Prestamos;
- Validación de datos realizados en otras instancias de registro;
- Estadísticas sobre los registros realizados;
- Creación y control de Memorándums utilizando tecnología Wysiwyg para el tema a de afiliaciones.

3.4.2. Diagramas de Actividades

Diagrama de Actividad: Registro de File, Ver figura 3.9

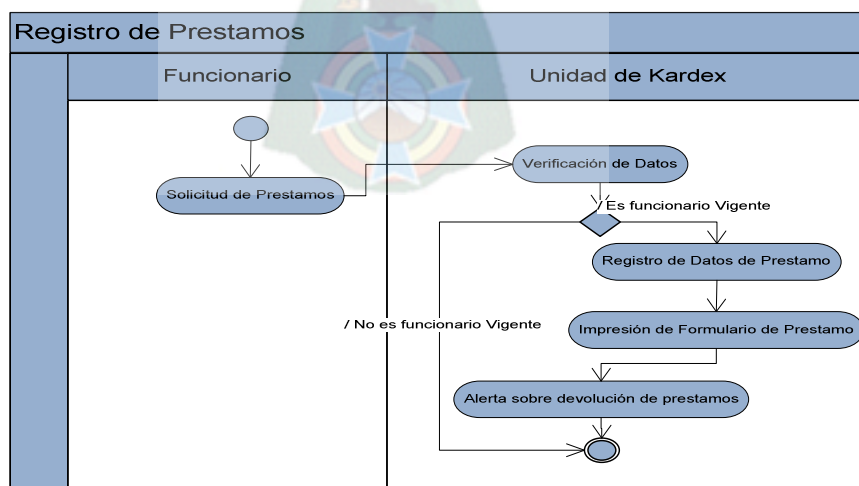
Figura 3. 9 Diagrama de Actividad de Registro de File



Fuente: [Elaboración Propia]

Diagrama de Actividad: Registro de Préstamos, Ver Figura 3.10

Figura 3. 10 Diagrama de Actividad de Registro de Prestamos



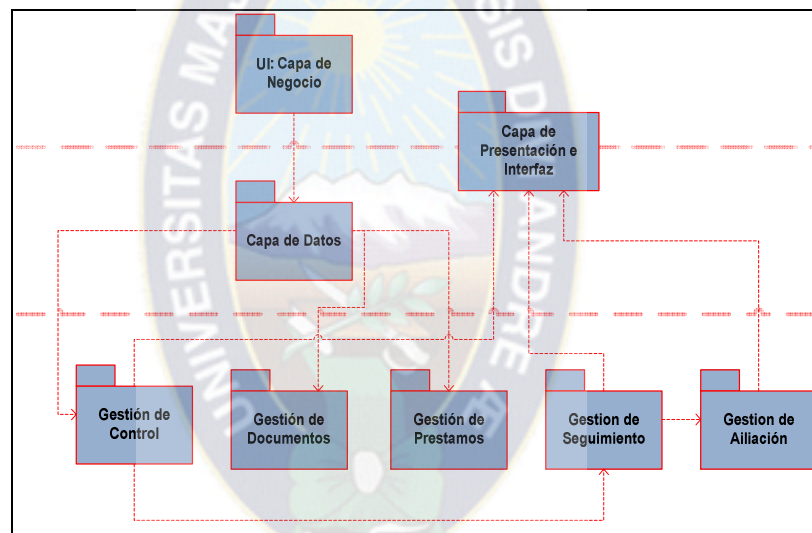
Fuente: [Elaboración Propia]

Durante esta fase se preparara el material adicional, es decir material para el usuario final y la definición final de los requerimientos los mismos que estarán reflejados en el Sistema de Archivo y Kardex y con esto cubrir las necesidades de la Institución.

3.4.3. Diagrama de Paquetes

Dentro del análisis del sistema es la categorización. Esto es, los diferentes elementos de los modelos de análisis se clasifican de una manera que se empaquetan como una agrupación, añadiendo a esto la arquitectura que se decidió utilizar: es la Arquitectura en tres capas al diseño e implementación del Producto como indica la Figura 3.11.

Figura 3. 11 Diagrama de Paquetes

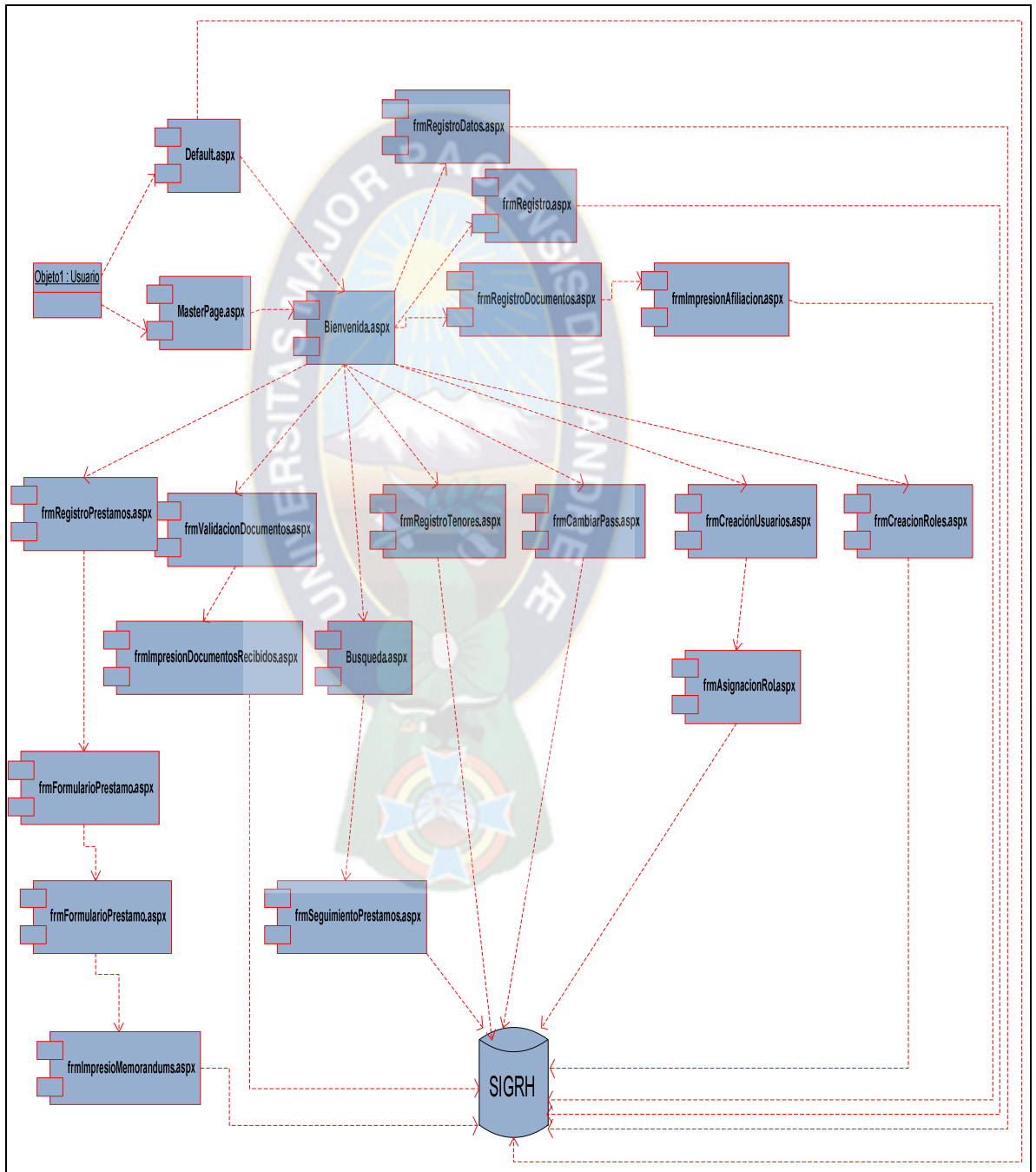


Fuente: [Elaboración Propia]

3.4.4. Diagrama de Componentes

Ver Figura 3.12

Figura 3. 12 Diagrama de Componentes



Fuente: [Elaboración Propia]

3.4.5. Diagrama de Clases

La Figura 3.13 muestra la representación de las clases encontradas y sus relaciones, así mismo tomar en cuenta que cada clase tiene respectivamente y por defecto las operaciones de Alta,Baja y Modificar.

Figura 3. 13 Diagrama de Componentes

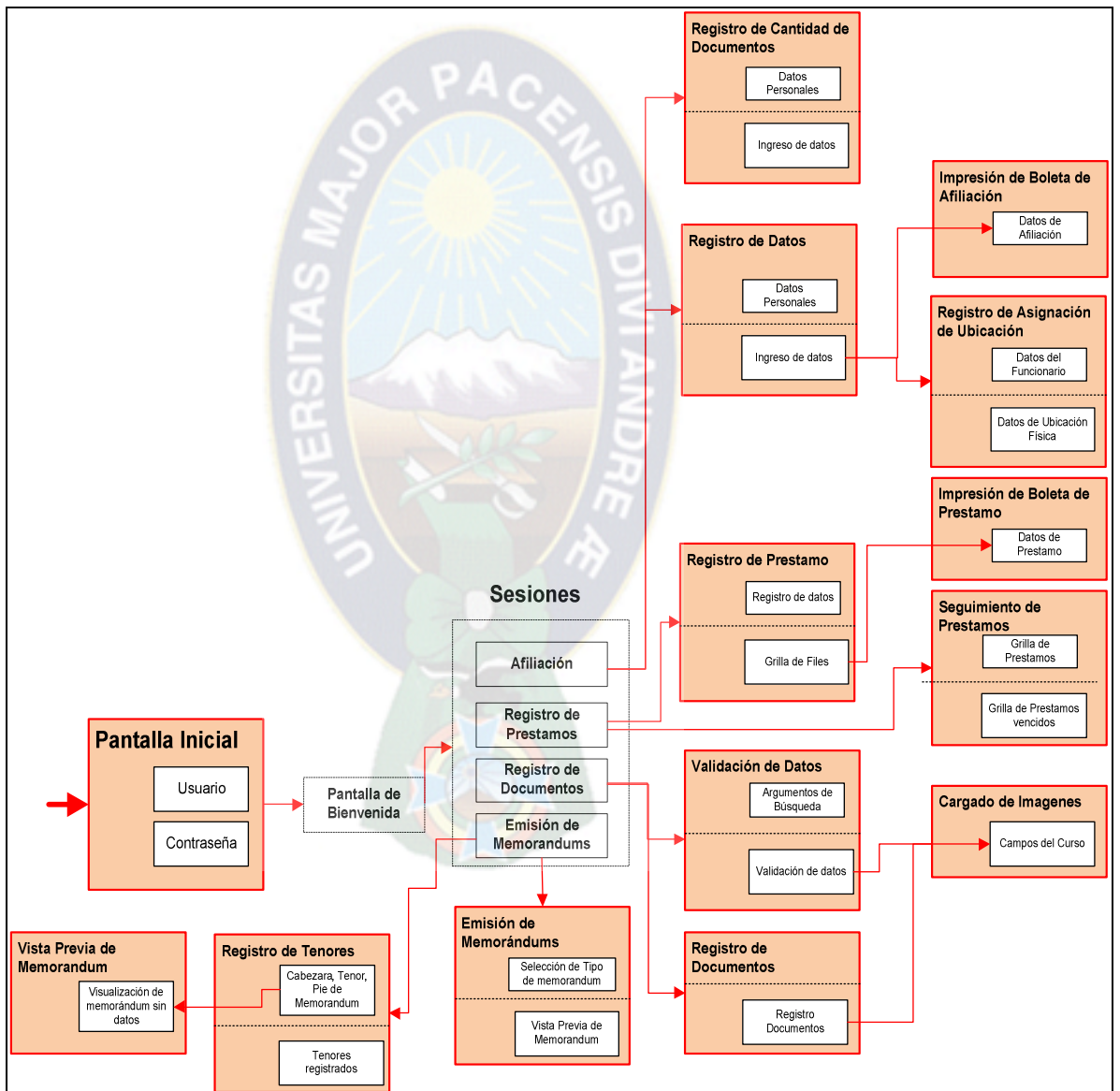


Fuente: [Elaboración Propia]

3.4.6. Diagrama Navegacional

Ver Figura 3.14

Figura 3. 14 Diagrama Navegacional del Sistema

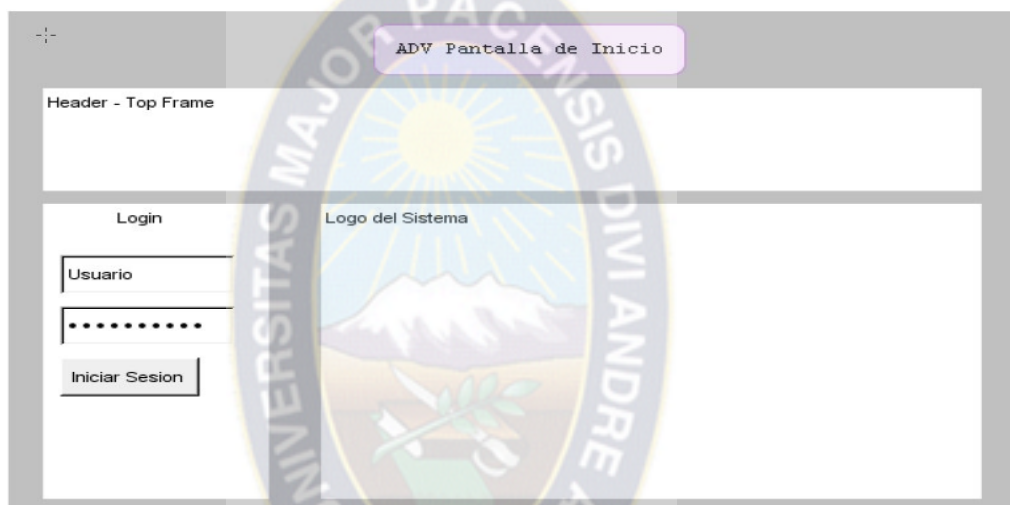


Fuente: [Elaboración Propia]

3.4.7. Prototipo de Interfaz de Usuario

Veamos los diferentes prototipos los cuales nos permitirán tener una visión de cómo se define el diseño final para poder realizar los formularios que utilizemos en cada uno de los procesos identificados. Como los formularios y la lógica del negocio presentan una misma funcionalidad la mayoría de los procesos tendrán la estructura de una pantalla de inicio, pantalla de búsqueda, pantalla de registro de datos y pantalla de reportes los cuales mostraremos a continuación.

Figura 3. 15 Pantalla de Inicio



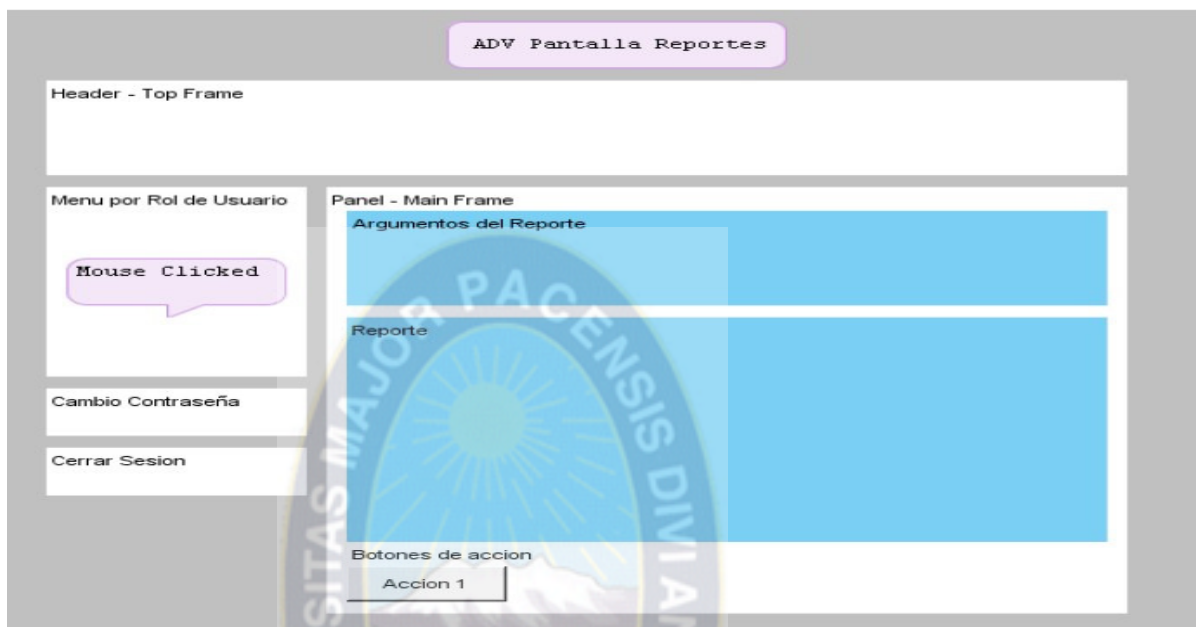
Fuente: [Elaboración Propia]

Figura 3. 16 Pantalla de Búsqueda



Fuente: [Elaboración Propia]

Figura 3. 17 Pantalla de Reportes



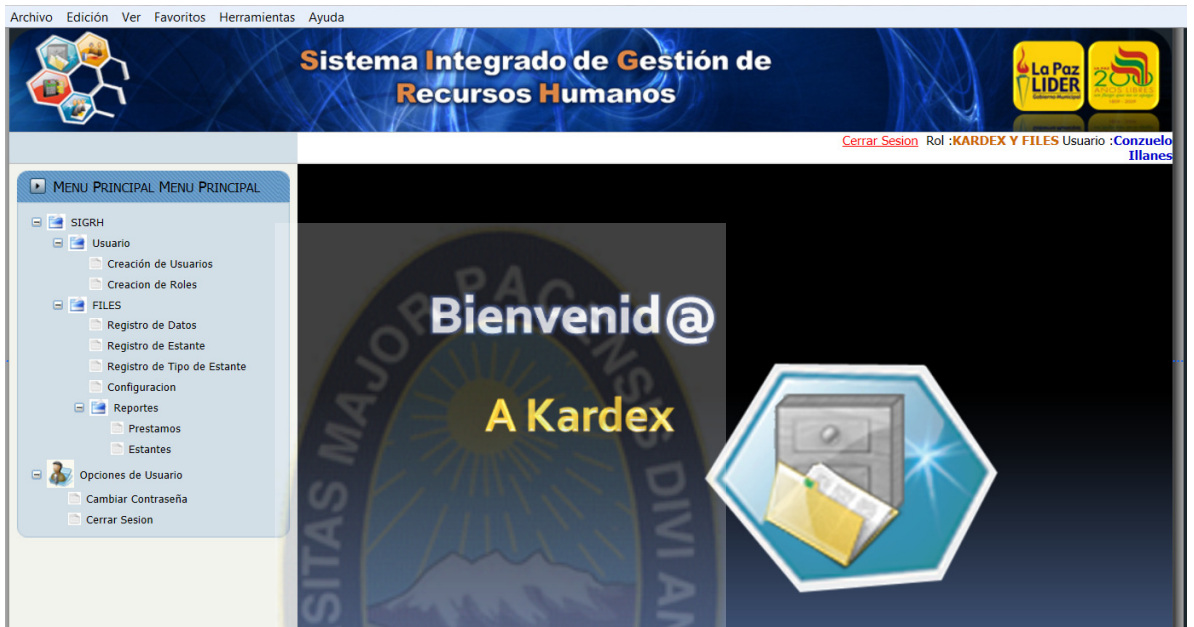
Fuente: [Elaboración Propia]

3.4.8. Interfaces de Usuario



Fuente: [Elaboración Personal de Desarrollo de Software]

Figura 3. 18 Menú de Inicio



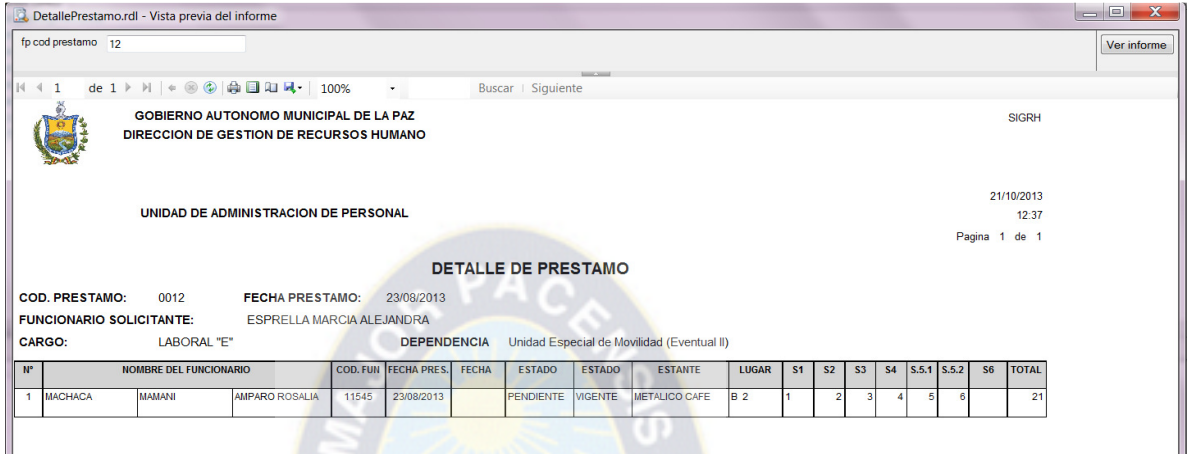
Fuente: [Elaboración Personal de Desarrollo de Software]

Figura 3. 19 Pantalla de Búsqueda



Fuente: [Elaboración Propia]

Figura 3. 20 Pantalla de Reportes



Fuente: [Elaboración Propia]

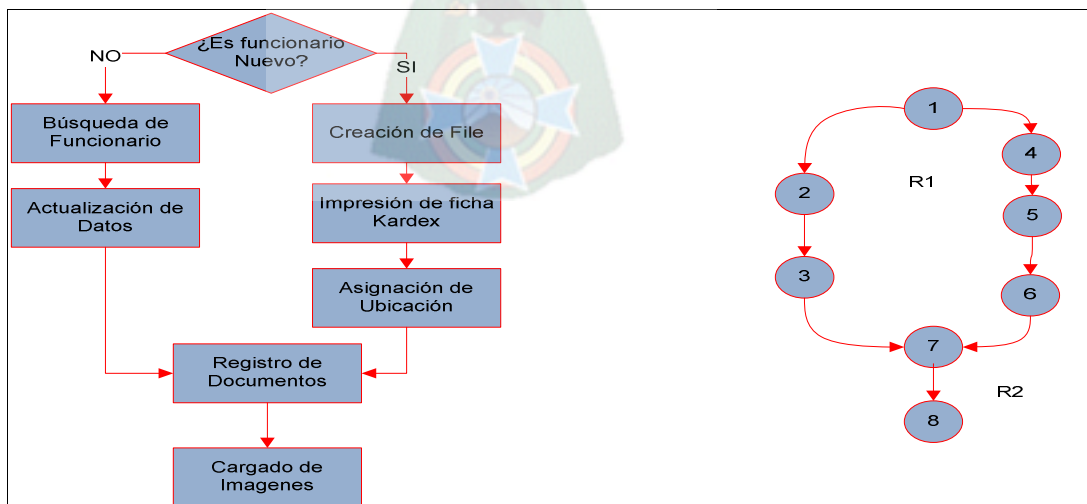
3.4.9. Pruebas del Sistema

La prueba es una serie de pasos planificados para contar con una construcción de software de forma correcta, se realizaran dos tipos de pruebas: pruebas de caja blanca y pruebas de caja negra.

a) Prueba de Caja Blanca

Registro de Files, Ver Figura 3.21

Figura 3. 21 Grafo de flujo para el Registro de Files



Fuente: [Elaboración Propia]

Número de Nodos = 8

Número de Aristas = 8

Número de Regiones = 2

Número de Nodos Predicado = 1

La complejidad Ciclomática será la siguiente:

$$V(G) = \text{N}^\circ \text{ de regiones} \rightarrow V(G) = 2$$

$$V(G) = \text{Número de Aristas} - \text{Número de Nodos} + 2$$

$$V(G) = 8 - 8 + 2 = 2$$

$$V(G) = \text{Número de Nodos Predicado} + 1$$

$$V(G) = 1 + 1 = 2$$

Caminos independientes:

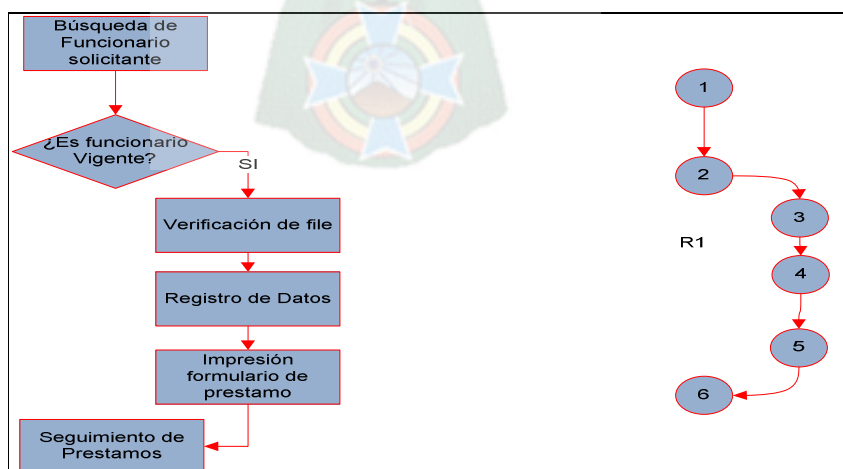
➤ Camino 1: 1-2-3-7-8

➤ Camino 2: 1-4-5-6-7-8

La complejidad ciclomática para el Registro de Files es 2, y con lo comprobamos que existe 2 caminos independientes

Registro de Prestamos, Ver Figura 3.22

Figura 3. 22 Grafo de flujo para el Registro de Prestamos



Fuente: [Elaboración Propia]

Número de Nodos = 6

Número de Aristas = 5

Número de Regiones = 1

Número de Nodos Predicado = 0

La complejidad Ciclomática será la siguiente:

$$V(G) = \text{N}^\circ \text{ de regiones} \rightarrow V(G) = 1$$

$$V(G) = \text{Número de Aristas} - \text{Número de Nodos}$$

$$V(G) = 5 - 6 + 2 = 1$$

$$V(G) = \text{Número de Nodos Predicado} + 1$$

$$V(G) = 0 + 1 = 1$$

Caminos independientes:

- Camino 1: 1-2-3-4-5-6

La complejidad ciclomática para el Registro Prestamos es 1, y con lo comprobamos que existe solo un camino.

b) Prueba de Caja Negra

Este tipo de pruebas denominadas también como pruebas de comportamiento, se centran en los requisitos funcionales del software, es decir que se la realiza con la ejecución del sistema y tomando en cuenta los siguientes puntos:

- Prueba de Unidad, se realizó a cada uno de los módulos verificando que contengan el menor número de errores;
- Prueba de Integridad, se realizó para verificar que los módulos funcionen de la manera esperada;
- Prueba de validación, este se hizo con la ayuda del usuario final y comprobar la satisfacción de los mismos;
- Prueba del Sistema, realizada con los módulos implantados y su ejecución con datos reales de la institución.

3.5. Fase de Transición

Esta fase es llevada a cabo con la implementación del Sistema, donde se realizarán las siguientes tareas:

- Subir la aplicación software denominado Sistema de Servicio de Archivo y Kardex en un servidor de Pruebas;
- Capacitación a todos los usuarios finales.
- Asistencia y ayuda a los usuarios en la operación con el Sistema.





CAPITULO 4 SEGURIDAD, CALIDAD Y COSTO BENEFICIO

4.1. Seguridad

Podemos entender como seguridad un estado de cualquier tipo de información (informático o no) que nos indica que ese sistema está libre de peligro, daño o riesgo. Se entiende como peligro o daño todo aquello que pueda afectar su funcionamiento directo o los resultados que se obtienen del mismo. Para la mayoría de los expertos el concepto de seguridad en la informática es utópico porque no existe un sistema 100% seguro. Para que un sistema se pueda definir como seguro debe tener estas cuatro características:

- Integridad, la información sólo puede ser modificada por quien está autorizado y de manera controlada;
- Confidencialidad, la información sólo debe ser legible para los autorizados;
- Disponibilidad, Debe estar disponible cuando se necesita;
- Irrefutabilidad (No repudio), el uso y/o modificación de la información por parte de un usuario debe ser irrefutable, es decir, que el usuario no puede negar dicha acción.

a) Integridad

La creación de usuarios con roles identificados y/o permisos ayudaran a clasificar a los mismos y designar tareas a realizar. La buena administración de los roles de usuario debe estar reflejada de acuerdo a la estructura organizativa de la Institución de forma natural (Ver Figura 4.1), y de acuerdo a las funciones que realiza los funcionarios que serán usuarios del Sistema.

Figura 4. 1 Tabla tblUsuarioRol

	IdUsuarioRol	IdRol	IdUsuario	IdEstado	dtmFechaAsign...	dtmFecha_Baja
▶	135	40	39	True	2013-03-12 15:...	NULL
	136	27	10062	True	2013-03-13 10:...	NULL
	140	20	10072	True	2013-03-18 11:...	NULL
	141	30	39	True	2013-03-18 16:...	NULL
	147	44	10063	True	2013-03-19 11:...	NULL
	148	12	39	True	2013-03-20 08:...	NULL
	149	45	10000	True	2013-03-21 17:...	NULL
	150	18	10073	True	2013-03-27 15:...	NULL
	152	16	10075	True	2013-04-05 16:...	NULL
	153	26	10074	True	2013-04-05 17:...	NULL
	155	40	10030	True	2013-04-05 18:...	NULL
	156	46	10036	True	2013-04-09 09:...	NULL
	157	47	10038	True	2013-04-10 09:...	NULL
	158	19	10076	True	2013-04-10 14:...	NULL
	159	37	277	True	2013-04-12 00:...	NULL
	160	2	10076	True	2013-04-12 17:...	NULL
	161	49	10077	True	2013-04-16 09:...	NULL
	162	39	10078	True	2013-04-19 15:...	NULL
	163	43	428	True	2013-04-16 18:...	NULL

Fuente: [Base de Datos]

Figura 4. 2 Tabla tblRol

IdRol	DescripcionRol	IdModulo	IdEstado
1	Administrador	0	True
2	Operador	0	True
3	Auxiliar	0	True
4	Consulta	0	True
5	Auxiliar Tratamiento Individualizado - AFP	0	True
6	Auxiliar Proyectos Sociales	0	True
7	Auxiliar Filiaciones E.G.S.	0	True
8	Area Tratamiento Individualizado	0	True
9	Auxiliar Proyectos Sociales Ex. Preoc.	0	True
10	Salarios Auxiliar 4	0	True
11	Reportes SIGRH-SISPER	0	True
12	Remuneraciones	0	True
13	Remuneraciones Validacion	0	True
14	Remuneraciones Reg. Cta. Bancaria	0	True
15	Emission de Credenciales	0	True

Fuente: [Base de Datos]

Figura 4. 3 Tabla tblMenuRol

IdMenuRol	IdMenu	IdRol	IdEstado	IdFechaRegistro
3	1195	10	True	2013-01-01 00:...
4	1196	10	True	2013-01-01 00:...
5	1197	10	True	2013-01-01 00:...
6	1198	10	True	2013-01-01 00:...
7	1199	10	True	2013-01-01 00:...
8	1193	35	True	2013-01-01 00:...
9	1194	35	True	2013-01-01 00:...
10	1195	35	True	2013-01-01 00:...
11	1196	35	True	2013-01-01 00:...
12	1197	35	True	2013-01-01 00:...
13	1198	35	True	2013-01-01 00:...
14	1199	35	True	2013-01-01 00:...
15	1193	43	True	2013-01-01 00:...
16	1194	43	True	2013-01-01 00:...
17	1195	43	True	2013-01-01 00:...
18	1196	43	True	2013-01-01 00:...

Fuente: [Base de Datos]

Figura 4. 4 Tabla tblMenu

IntIdMenu	NvcrDescripcio...	NvcrUrl	NvcrTooltip	NvcrEstado	IntIdMenuSupe...	IdMenuMaster
4076	Creacion de Roles	~/Usuario/tblRol...	Permite crear rol...	V	4069	2
4077	Registro Bono A...	~/Retribuciones...	Registrar Bono d...	V	4065	3
4078	Busqueda	~/ConsultaRem/...	Consulta Inform...	V	1127	5
4079	Funcionarios Dis...	~/Programacion ...	Registro de caso...	V	1000	3
4080	Registro Casos c...	~/Programacion ...	Registro de caso...	V	4079	3
4050	Asistencia Estru...	~/ControlPerson...	NULL	V	4005	4
1184	Incompatibilidad ...	NULL	Módulo de Inco...	V	1100	3
1185	Registrar Incom...	~/Programacion ...	Para Añadir Inco...	V	1184	3
1186	Modificar Incom...	~/Programacion ...	Para Modificar I...	V	1184	3
1187	Incompatibilidad ...	NULL	Módulo de Inco...	V	1150	3
1188	Registrar Incom...	~/Programacion ...	Para Añadir Inco...	V	1187	3
1189	Modificar Incom...	~/Programacion ...	Para Modificar I...	V	1187	3
3017	Consulta Funcio...	~/Programacion ...	Datos del Funcio...	V	3001	3
1190	Reportes	~/Programacion ...	Listados de Inco...	V	1184	3
1191	Reportes	~/Programacion ...	Listados de Inco...	V	1187	3
4051	Presupuestos	NULL	NULL	V	0	8
4053	Reporte Individual	~/Presupuestos...	Reporte Individu...	V	4051	8

Fuente: [Base de Datos]

Como se podrá observar en los gráficos anteriores de las tablas que tienen como objetivo el manejo de roles y al tipo de acceso del usuario.

Figura 4. 5 Tabla M_Usuarios

IdUsuario	IdRoles	Usuario	Contraseña	NombreCompleto	Apellidos	Estado	EstadoPassword	CodigoUsuario	PathBienvenida	ProveedorMap...	CorreoInterno	EstadoSession	Fecha_Creacion
272	20	kd_eapaza	...	Eida	APZA	V	True	11807	NULL	NULL	NULL	False	2012-01-01 00:...
301	15	kd_amayta_reg	...	Abel	Mayta	V	True	1976	NULL	NULL	NULL	False	2012-01-01 00:...
284	8	reloj_hampaturi	...	Sub-Alcaldia	Hampaturi	C	True	22	NULL	NULL	NULL	False	2012-01-01 00:...
300	58	UE164	...	ADMINISTRACI...	Ministerio de Ha...	C	True	164	NULL	NULL	NULL	False	2012-01-01 00:...
208	59	cp_aajpl	...	Antonio	Ajpa	C	False	5918	NULL	NULL	NULL	False	2012-01-01 00:...
302	0	cp_elopez	...	Emilio	Lopez	V	True	1150	NULL	NULL	NULL	False	2012-01-01 00:...
10004	0	cp_mzurita	...	Maria Antonieta	Zurita	V	True	461	NULL	NULL	NULL	False	2012-01-01 00:...
350	20	kd_pacheco	...	JOVANA	PACHECO	C	True	13182	NULL	NULL	NULL	False	2012-01-01 00:...
284	8	reloj_mallasa	...	Sub-Alcaldia	Mallasa	V	True	18	NULL	NULL	NULL	False	2012-01-01 00:...
10006	0	des_sambientales	...	Servicios Ambien...	Servicios Ambien...	V	False	10006	NULL	NULL	NULL	False	2012-01-01 00:...
160	58	UE174	...	PROGRAMA DE ...	Ministerio de Ha...	C	True	174	NULL	NULL	NULL	False	2012-01-01 00:...
10001	0	cp_pchura	...	Patricia	Chura	V	True	2457	NULL	NULL	NULL	False	2012-01-01 00:...
128	9	nelcam	...	Julio	Leon Quispe	C	True	2254	NULL	NULL	NULL	False	2012-01-01 00:...
267	17	ccam	...	CCAM	CCAM	V	True	0	NULL	NULL	NULL	False	2012-01-01 00:...
666	98	sal_yo	...	yo	yo	C	True	11545	bsstos/retribu...	NULL	NULL	True	2012-01-01 00:...
310	59	raul	...	Raul	Reque	C	True	7078	NULL	NULL	NULL	False	2012-01-01 00:...
353	59	licencias	...	Licencias	Licencias	C	True	123456789	NULL	NULL	NULL	False	2012-01-01 00:...
311	59	maira	...	Maira	Cuentas	C	True	7533	NULL	NULL	NULL	True	2012-01-01 00:...

Fuente: [Base de Datos]

b) Confidencialidad

De acuerdo a lo mencionado en los anteriores puntos la definición de roles de usuarios, permitirá toda acción realizada será registrado con el Id de usuario que es único y a la vez tendrá acceso a la información que el mismo usuario registro y no así a la

información que otros usuarios introdujeron al sistema; con esta acción se evitara la pérdida o cambio de información.

También cabe recalcar que el campo de Contraseña de la tabla M_Usuarios se encuentra encriptado, esto para brindar más seguridad en el sistema.

c) Disponibilidad

La disponibilidad del sistema, como también el acceso a los datos deben realizarse en el momento preciso en que el usuario lo solicite, la tecnología con la que cuenta el sistema permitirá la accesibilidad en cualquier momento y desde cualquier terminal perteneciente a la Red del Gobierno Autónomo Municipal de La Paz sin necesidad de que el Sistema sea instalado en el equipo ya que el Sistema está desarrollado bajo tecnología Web.

Además se cuenta con una réplica de la Base de Datos y del Servidor para que en el caso de que el servidor tenga conflictos se direcciona y se levante el otro servidor.

d) Irrefutabilidad

Para la realización de esta tarea se maneja una tabla llamada Audit como muestra la Figura 4.5, el mismo que tiene la función de registrar todos los movimientos (Alta, Baja, Modificación y Consulta) realizados en el Sistema, a su vez almacenar la fecha y la en la que se realizó la operación, quien la realizo, que funcionario fue afectado con el cambio, la tabla que fue afectada y el IP del equipo donde se realizo dicha operación.

Figura 4. 6 Tabla Audit

	AuditID	Type	TableName	PrimaryKeyField	PrimaryKeyValue	FieldName	OldValue	NewValue	UpdateDate	UserName
	1	I	SI_CP_Licencia...	lj_id	488821	lj_id	NULL	488821	2012-10-23 23:...	sigrh/GMLPSR0057
	2	I	SI_CP_Licencia...	lj_id	488821	lj_cod_fun	NULL	10662	2012-10-23 23:...	sigrh/GMLPSR0057
	3	I	SI_CP_Licencia...	lj_id	488821	lj_tipo_licencia	NULL	1	2012-10-23 23:...	sigrh/GMLPSR0057
	4	I	SI_CP_Licencia...	lj_id	488821	lj_fecha_emision	NULL	23/10/2012 00:...	2012-10-23 23:...	sigrh/GMLPSR0057
	5	I	SI_CP_Licencia...	lj_id	488821	lj_fecha_inicial	NULL	23/10/2012 00:...	2012-10-23 23:...	sigrh/GMLPSR0057
	6	I	SI_CP_Licencia...	lj_id	488821	lj_fecha_final	NULL	23/10/2012 00:...	2012-10-23 23:...	sigrh/GMLPSR0057
	7	I	SI_CP_Licencia...	lj_id	488821	lj_hora_salida	NULL	23/10/2012 14:...	2012-10-23 23:...	sigrh/GMLPSR0057
	8	I	SI_CP_Licencia...	lj_id	488821	lj_hora_retorno	NULL	23/10/2012 20:...	2012-10-23 23:...	sigrh/GMLPSR0057
	9	I	SI_CP_Licencia...	lj_id	488821	lj_motivo	NULL	MUESTREO Y PR...	2012-10-23 23:...	sigrh/GMLPSR0057
	10	I	SI_CP_Licencia...	lj_id	488821	lj_lugar	NULL	CUENCA ORKOJ...	2012-10-23 23:...	sigrh/GMLPSR0057
	11	I	SI_CP_Licencia...	lj_id	488821	lj_nom_inim_sup...	NULL	Ing. Efrain Fern...	2012-10-23 23:...	sigrh/GMLPSR0057
	12	I	SI_CP_Licencia...	lj_id	488821	lj_nom_director	NULL		2012-10-23 23:...	sigrh/GMLPSR0057
	13	I	SI_CP_Licencia...	lj_id	488821	lj_cod_usuario	NULL	10662	2012-10-23 23:...	sigrh/GMLPSR0057
	14	I	SI_CP_Licencia...	lj_id	488821	lj_cod_inm_supe...	NULL	4824	2012-10-23 23:...	sigrh/GMLPSR0057
	15	I	SI_CP_Licencia...	lj_id	488821	lj_cod_unidad_tr...	NULL	5155	2012-10-23 23:...	sigrh/GMLPSR0057
	16	I	SI_CP_Licencia...	lj_id	488821	lj_estado_recep...	NULL	0	2012-10-23 23:...	sigrh/GMLPSR0057
	17	I	SI_CP_Licencia...	lj_id	488821	lj_estado_anulado	NULL	0	2012-10-23 23:...	sigrh/GMLPSR0057
	15395	I	SI_CP_Licencia...	lj_id	488925	lj_motivo	NULL	REUNION	2012-10-24 09:...	sigrh/GMLPSR0057
	15727	U	SI_CP_Licencia...	lj_id	483428	lj_CorrespondeA...	NULL	0	2012-10-24 10:...	sigrh/GMLPSR0057
	16172	I	SI_CP_Licencia...	lj_id	488939	lj_estado_lic_just	NULL	V	2012-10-24 10:...	sigrh/GMLPSR0057

Fuente: [Base de Datos]

4.1.1 Seguridad del Software

La seguridad del software es un tema de suma importancia y se debe aplicar durante todo el proceso de desarrollo del mismo, tomando en cuenta en que es lo que va hacer el Sistema y como será utilizado. Ya que el sistema está desarrollado con tecnología web se debe tener un modelador para la identificación de posibles amenazas al sistema y para esto se utilizara el modelo DREAD (Daño potencial. Facilidad de reproducción, Capacidad de Explotación, usuarios afectados y la dificultad para su descubrimiento).

En la tabla 4.1 se identifica los posibles riesgos que pueden afectar al sistema

Tabla 4.1 Evaluación de vulnerabilidades según DREAD

Nro	Amenaza	D	R	E	A	D	Σ
1	Inyección de SQL	1	1	1	3	3	9
2	Script entre Sitios	1	1	2	1	1	6
3	Modificación del Ingreso	2	2	2	1	2	9
4	Robo de Claves	1	1	2	2	1	7
5	Reemplazo de Sesión	1	1	1	2	2	7
6	Reemplazo de Identidad	2	1	2	1	2	8
7	Revelado de Información	3	2	2	1	1	9

Fuente: [Elaboración Propia]

A través del framework utilizado y la arquitectura en tres capas, los mismos que ayudan a que el sistema proporcione seguridad y se debiliten las posibles amenazas como ser la Inyección de SQL y Script entre sitios.

El Robo de claves no se dará ya que se cuenta con modelos de encriptación tanto en el campo usuario y password a la hora del Ingreso al Sistema.

Para evitar el reemplazo de sesión se realiza un corte automático a las sesiones que dejan de ser utilizados más de 15 minutos, esto para evitar la infiltración de individuos ajenos al sistema.

4.1.2 Seguridad Administrativa

De acuerdo a las normas vigentes de la institución y el proceso de designación de tareas se realizara la emisión de Memorándums que especifiquen las tareas a realizar para los diferentes usuarios y la responsabilidad de la información.

4.2 Factores de Calidad ISO 9126

4.2.1 Confiabilidad

Para la medición de la misma se toma la siguiente fórmula:

$$R_i(t) = e^{-\lambda * t}$$

Donde:

R(t), es la función de confiabilidad de un componente en el tiempo t.

λ , tasa constante de fallo.

t, Periodo de operación de tiempo.

Teorema 1, si n componente, funcionan independientemente conectados en serie y el i-esimo componente tiene confiabilidad $R_i(t)$, entonces la confiabilidad total esta dado por:

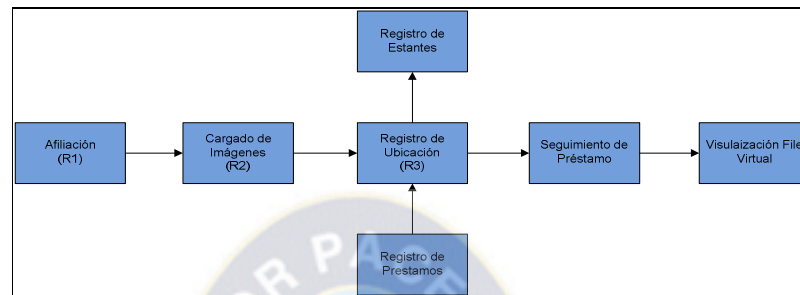
$$R(t) = R_1(t) * R_2(t) * R_3(t) * \dots * R_n(t)$$

Teorema 2, si n componentes funcionar independientemente y actúan en paralelo y el i-esimo componente tiene confiabilidad $R_i(t)$, entonces la confiabilidad total esta dada por:

$$R(t) = 1 - [1 - R_1(t)] * [1 - R_2(t)] * [1 - R_3(t)] * \dots * [1 - R_n(t)]$$

Para el cálculo y la obtención del nivel de confiabilidad, el mismo que será calculado por medio del diagrama de transferencia (Ver Figura 4.7)

Figura 4. 7 Diagrama de Transferencia



Fuente: [Elaboración Propia]

Luego obtenemos la siguiente tabla:

Tabla 4.2 Calculo de Confiabilidad

$R_i(t)$	Λ	$t[\text{Hrs}]$	$e^{-\lambda*t}$
$R_1(t)$	0.01	10	0.90
$R_2(t)$	0.01	10	0.90
$R_3(t)$	0.005	10	0.95
$R_4(t)$	0.005	10	0.95
$R_5(t)$	0.005	10	0.95
$R_6(t)$	0.01	10	0.90
$R_7(t)$	0.01	10	0.90

Fuente: [Elaboración Propia]

Realizando los cálculos por medio de los teoremas 1 y 2 tenemos:

$$R_T(t) = \{1 - [1 - R_3(t)] * [1 - R_4(t)] * [1 - R_5(t)] * [1 - R_6(t)] * [R_1(t)] * [R_2(t)] * [R_7(t)]\}$$

$$R_T(t) = \{1 - [1 - 0.95] * [1 - 0.95] * [1 - 0.95] * [1 - 0.90] * [0.90] * [0.90] * [0.90]\}$$

$$R_T(t) = 0.85$$

$$\text{Confiabilidad} = R_T(t) * 100\% = 85\%$$

4.2.2 Funcionalidad

La ecuación de punto función permite su cálculo:

$$PF = \text{Cuenta Total} * (A + (E * \text{Sum}(F_i)))$$

Donde:

PF, funcionalidad estimada del sistema.

Cuenta Total, representa la suma de todas las entradas.

A, representa la portabilidad subjetiva estimada de la confiabilidad

E, error estimado del sistema

Sum(F_i), valores de ajuste de complejidad

Primero identificamos los parámetros de medición

Tabla 4.3 Parámetros de Medición

Número de entradas de Usuario	<ul style="list-style-type: none">-Registro de Datos de Funcionario-Registro de Datos de Estantes-Registro de Columnas-Registro de Documentos-Registro de Prestamos-Registro de Imágenes-Consultas
Número de Salidas de Usuario	<ul style="list-style-type: none">-Reporte de Prestamos-Formulario de Afiliación-Visualización de File Virtual-Formulario de Prestamos-Reporte de Estantes-Reporte Ubicación Física de File

	-Emisión de Memorándums. -Análisis de Finiquitos -Grafico Estadístico
Número de Peticiones de usuario	-Menú principal del usuario de acuerdo al rol -Búsqueda de Funcionarios, documentos -Historial de asignaciones y visualización de documentos escaneados
Numero de Archivos	-Funcionarios, documentos, estantes, files, prestamos.
Número de Interfaces externas	-Información de respaldo, Exportación de Informes

Fuente: [Elaboración Propia]

Tabla 4.4 Calculo de Cuenta Total

Parámetro de medida	Cuenta	Ponderación	Total
Número de entradas de Usuario	7	4	28
Número de Salidas de Usuario	9	5	45
Número de Peticiones de usuario	3	6	18
Numero de Archivos	5	5	25
Número de Interfaces externas	2	5	10
Cuenta Total			126

Fuente: [Elaboración Propia]

Los ajustes de complejidad se realizan de acuerdo a las siguientes respuestas:

Tabla 4.5 Ajustes de Complejidad

Importancia	0%	20%	40%	60%	80%	100%
Escala	No Influyente	Incidental	Moderado	Medio	Significativo	Esencial
Factor	0.	1	2	3	4	5
1. ¿Requiere el sistema copias de seguridad y de recuperación fiables?	X					
2. ¿Se requiere comunicación de datos?					X	
3. ¿Existen funciones de procesamiento distribuido?			X			
4. ¿Es crítico el rendimiento?		X				
5. ¿Se ejecutara el sistema en un entorno operativo existente y fuente utilizado?					X	
6. ¿Requiere el sistema entrada de datos interactivo?			X			
7. ¿Requiere la entrada de datos interactivos que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variables utilizadas?				X		
8. ¿Se actualizan los archivos maestros de forma interactiva?			X			
9. ¿Son complejas las entradas, las salidas o las operaciones?		X				
10. ¿Es complejo el procesamiento Interno?				X		

11. ¿Se ha diseñado el código para ser reutilizable?					X	
12. ¿Están incluidas en el diseño la conversión y la instalación?				X		
13. ¿Se ha diseñado el Sistema para soportar múltiples instalaciones en diferentes organizaciones?			X			
14. ¿Se ha diseñado la aplicación para facilitar los cambios para ser fácilmente utilizado por el usuario?						X

Fuente: [Elaboración Propia]

$$\sum F_i = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 + F_9 + F_{10} + F_{11} + F_{12} + F_{13} + F_{14}$$

$$\sum F_i = 5 + 4 + 2 + 1 + 4 + 2 + 3 + 2 + 1 + 3 + 4 + 3 + 2 + 0$$

$$\sum F_i = 36$$

Reemplazando en la ecuación de punto funciona para un nivel de confianza del 65%

$$PF_{\text{real}} = \text{cuenta total} * (A + (E * \text{Sum}(F_i)))$$

$$PF_{\text{real}} = 214 * (0.65 + 0.01 * 36)$$

$$PF_{\text{real}} = 216.14$$

Ahora calculamos para un nivel de confianza del 100%

$$PF_{\text{esperado}} = \text{cuenta total} * (A + (E * \text{Sum}(F_i)))$$

$$PF_{\text{esperado}} = 214 * (1 + 0.01 * 36)$$

$$PF_{\text{esperado}} = 291.04$$

El porcentaje de funcionalidad será como sigue:

$$\% PF = PF_{\text{real}} / PF_{\text{esperado}}$$

$$\% PF = 216.14 / 291.04 = 0.74$$

Por lo tanto se tiene una funcionalidad de 74%

4.2.3 Facilidad de Mantenimiento

Para obtener y medir la facilidad de mantenimiento aplicaremos la métrica del tiempo medio TMC.

$$\text{TMC} = \text{TA} + \text{TD} + \text{TI} + \text{TP}$$

Donde:

TA, es el tiempo en el que se tarda analizar la petición de cambio.

TD, es el tiempo empleado en diseñar una modificación adecuada

TI, es el tiempo necesario para implementar el cambio

TP, es el tiempo empleado en probar y distribuir el cambio a todos los usuarios

El mantenimiento de un sistema se puede dividir en dos partes que son: el mantenimiento correctivo y el mantenimiento de mejora; los mismos que deben ser calculables de acuerdo a la fórmula del TMC.

a) Mantenimiento Correctivo

Tabla 4.6 Mantenimiento Correctivo

En el mejor de los casos [min]	En el peor de los casos [hrs]
$\text{TMC} = 20 + 30 + 60 + 80$	$\text{TMC} = 24 + 12 + 48 + 24$
$\text{TMC} = 190[\text{min}] = 3.2[\text{hrs}]$	$\text{TMC} = 108[\text{hrs}]$

Fuente: [Elaboración Propia]

Calculando el promedio de facilidad de mantenimiento correctivo se tiene que el trabajo para desarrollar esta tarea es de 56 hrs.

b) Mantenimiento de Mejora

Tabla 4.7 Mantenimiento de Mejora

En el mejor de los casos [hrs]	En el peor de los casos [días]
$\text{TMC} = 5 + 5 + 10 + 6$	$\text{TMC} = 10 + 10 + 20 + 10$

TMC = 26[hrs] = 1.1[días]	TMC = 50[días]=1200
----------------------------------	----------------------------

Fuente: [Elaboración Propia]

Calculando el promedio de facilidad de mantenimiento de mejora se tiene que el trabajo para desarrollar esta tarea es de 613 hrs.

4.2.4 Portabilidad

Para el cálculo del grado de portabilidad se hará uso de la siguiente ecuación:

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

Donde:

GP, es el grado de portabilidad

CT, costo de transportar

CRD, costo de re desarrollo

Se debe tomar en cuenta lo siguiente:

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

Si $GP = 0$, la portabilidad es perfecta.

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

Realizando los cálculos tendremos:

$$GP = 1 - [200\$/1000\$]$$

$$GP = 0.80$$

Por lo tanto es portable en un 80%, esto se debe a que si hacemos uso de otros sistemas operativos como ser una Macintosh o Linux el sistema presentaría a nivel de servidor algunos problemas o demora en el levantamiento del sistema, pero cabe mencionar que si se tratara de sistemas operativos de Microsoft no se tendría ningún problema.

4.3. Análisis de Costos

Los costos que incidieron en la construcción y finalización del Sistema de Información se especifican de la siguiente manera.

Costo de desarrollo del software, costo de capacitación del usuario y costo del software de construcción del sistema.

4.3.1. Costo de Desarrollo de Software

La obtención de este precio se realizó sobre la base del método de la estimación empírica COCOMO básica, que estima el esfuerzo E y el tiempo de duración del desarrollo del software T, mediante las siguientes ecuaciones:

$$E = a*(KLDC)^b$$

$$T = c * E^d$$

Dadas las características del software, se pueden afirmar que el mismo está comprendido entre el grupo de sistemas semiacoplados.

Por otra parte, para el cálculo del esfuerzo se tomó en cuenta el número de puntos de función calculado en el anterior acápite y la conversión a número de líneas de código. Como la herramienta utilizada en la construcción del software es de cuarta generación, entonces el número estimado de líneas de código se encuentra entre 4.186 y 7440, según la tabla de conversión de KLDC por puntos función, en este caso se toma en cuenta el mínimo (por el uso principal de SQL) donde un 60% es código reutilizable, se tendrán 1.934LDC \approx 1.9KLDC.

Entonces los cálculos del esfuerzo y de la duración se realizaran así:

$$E = 3 * 1.9^{1.12} = 6.16 \approx 6 \text{ personas}$$

$$T = 2.5 * (E)^{0.35}$$

$$T = 2.5 * 6^{0.35}$$

$$T = 1.245$$

$$T = 1$$

Finalmente si un analista gana aproximadamente un salario mensual de 400\$ (2855 Bs.), entonces el software tendrá un costo de 2.400 \$.

4.3.2. Costo de Capacitación del Usuario

Este costo se calcula de acuerdo a la magnitud del sistema y a sus características, el tiempo de la capacitación será una hora y media diaria por cinco días, los mismos que tendrán un costo diario de 15\$ (10\$ por hora), lo que hace un total de 75\$.

4.3.3. Costo del Software de Construcción del Sistema

Los costos de licencia del software a utilizar en el presente sistema ya se encuentran cubiertos por la institución por ser un estándar de herramientas que se utilizan en el Gobierno Autónomo Municipal de La Paz



CAPITULO 5 CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

En la culminación del presente proyecto y el empleo de nuevas tecnologías como las apoyadas sobre interfaces web, permiten modernizar sistemas orientados a cliente servidor, y expandir el acceso a los mismos desde otros nodos de una red, sin requerir la instalación de software adicional, a excepción de un browser.

Por otro lado, esta misma tecnología, al permitir la integración por medio de una interfaz única, el browser, se adapta perfectamente para hacer posible la implementación Sistema de Servicio de Archivo y Kardex, de una forma rápida, ágil y adecuada. Esto es fundamental para complementar el tipo de herramientas informáticas que está buscando una Institución como es el Gobierno Autónomo Municipal de La Paz.

Cabe mencionar que con la implementación del Sistema de Servicio de Archivo y Kardex no se pretende reemplazar al personal que se encarga del proceso de administración y resguardo de files, más aun el sistema desarrollado será un apoyo para el desempeño de sus funciones.

Evaluando las distintas experiencias obtenidas y las tareas desarrolladas durante el Proyecto de Grado, se pueden concluir que el objetivo principal y los objetivos secundarios fueron cubiertos, los mismos que fueron planteados en el capítulo 1.

Objetivo Principal

“Implantar el sistema de Servicio de Archivo y Kardex de los expedientes del personal y disminuir el esfuerzo de trabajo y facilitar el acceso de la información y el seguimiento, para cumplir de una forma rápida el control de toda la documentación y mantener actualizados los expedientes del personal, el cual ayudara en la toma de decisiones”.

Este objetivo fue alcanzado con el desarrollo del Servicio de Archivo y Kardex siguiendo la metodología RUP como principal herramienta, el mismo permite el desarrollo y la construcción de un modelo de solución de manera ágil y práctica, permitiendo visualizar todos los conceptos técnicos del método en cuestión.

Objetivos Secundarios alcanzados

- “Generar una ficha de salida del expediente personal, con el registro de datos del funcionario a quien se presta esta documentación, en base al kardex de cada funcionario, cargo que ocupa, ubicación de puesto de trabajo etc.”

Por medio del módulo de registro de préstamos, se lograra obtener un formulario de constancia de préstamo y el almacenamiento de toda la información necesaria.

- “Ubicar físicamente el expediente en los estantes respectivos”.

Gracias al módulo de Asignación de Ubicación, se pudo cubrir este objetivo, ya que esta operación permite guardar información referente a la ubicación física del file.

- “Llevar un control de cuantas veces es prestado un expediente y la devolución completa de las mismas”.

Gracias al módulo de Registro de Prestamos podremos realizar el seguimiento de préstamos de files a través de alertas.

- “Elaborar, tramitar y enviar los récords de servicio y archivar copia en el expediente del empleado en cuestión”.
- “Elaborar, tramitar, y entregar las constancias de trabajo así como cualquier otro documento que soliciten los empleados y los organismos oficiales”.
- “Mantener actualizada la relación de personal para el otorgamiento de los reconocimientos por tiempo de servicio en la Institución”

A través de la información registrada se podrá elaborar informes sobre años de servicio, certificados de trabajo, haciendo uso de la tecnología Wysiwyg y de esta manera cumplir los tres objetivos anteriores.

- “Elaborar y mantener actualizados los expedientes del personal, ejecutivo, administrativo, técnico y obrero, etc”.
- “Garantizar la guarda y custodia de los Expedientes del personal de la Institución, así como la confidencialidad de los documentos contenidos en él”.
- “Garantizar que cada “Movimiento de Personal” se archive acompañado de los soportes legales correspondientes”

Los tres anteriores objetivos son cubiertos por el módulo de registro de documentación, el mismo que permitirá registrar, actualizar y para luego almacenar físicamente toda la documentación ingresada y emitida en todas las reparticiones pertenecientes a la institución,

Como conclusión final sobre el trabajo efectuado, se destaca que se alcanzaron exitosamente todos los objetivos como se describió anteriormente. El producto obtenido cuenta con todas las características que se relevaron como deseables al inicio del proyecto de grado, resultando efectivamente en una herramienta de ayuda para los procesos que se efectúan en la Unidad de Administración de Personal y el Servicio de Files y Kardex del Gobierno Autónomo Municipal de La Paz.

5.2 Recomendaciones

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

- Administrar la gestión de cambios sobre los procesos evaluados, conservando sus versiones a lo largo de las diferentes evaluaciones;
- Migrar la capa de aplicación de la herramienta de un entorno Intranet al entorno de Internet;
- A efectos de incorporar esta ampliación será necesario tener en cuenta la integración de todos los sistemas transaccionales que de una u otra manera interactúan con el sistema, para determinar la información requerida por parte de los procesos del sistema de información;
- Ampliación de las facilidades del sistema con ayudas en línea y tutores que brinden explicaciones y permitan conocer y comprender el manejo del sistema;
- Contemplar la aplicabilidad en otras instituciones, y de esta manera contar con una herramienta más general adecuada al contexto nacional



CAPITULO 6 BIBLIOGRAFIA

6.1. Bibliografía

- [JACOBSON, 1999] Jacobson, Rumbaugh, J., Jacobson, 1999: "Proceso Unificado de desarrollo de Software", Ed: Addison Wesley, España, 438 pags.
- [BOOCH, 1999] Booch, G., Rumbaugh, J., Jacobson, 1999: "El Lenguaje Unificado de Modelado", Ed: Addison Wesley, España, 525 pags.
- [REINA, 2006] Reina A. M., Torres J., Toro M., Escalona M. J. 2006: "Caracterizando el aspecto de navegación", España, 10 pags.
- [GUERRERO, 2004] Guerrero A. Luis 2004: "CC61J - Taller de UML", Chile, 40 pags.
- [LETELIER, 2003] Letelier, T. Patricio 2004: "Introducción a Rational Unified Process (RUP)".
- [KOCH, 2000] The Authoring Process of the UML-based Web Engineering Approach (Nora Koch 1,2, Andreas Kraus1, Rolf Hennicker1)

6.2. WebGrafía

- [MONOGRAFIAS, 2011] Monografias.com. Importancia y manejo de gestión y organización de archivos y documentación en la empresa Recuperado de <http://www.monografias.com/trabajos59/gestion-organizacion-archivos/gestion-organizacion-archivos2.shtml>
- [EVillalbaDomínguez, 2012] Proceso Unificado Rational Aplicado <http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/175/A8%20Cap%C3%ADtulo%205.pdf?sequence=8>
- [O.Fernandez, 2011]. <http://es.scribd.com/doc/50987400/Que-es-UWE>
- [LMU – Ludwig-Maximilians-Universität München, 2012] UWE – UML-based Web Engineering - <http://uwe.pst.ifi.lmu.de/teachingTutorialPresentationSpanish.html>
- [SOFTENG, 2013] Metodología Scrum para desarrollo de software - aplicaciones complejas <http://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>
- [UMSA 2012] ANALISIS Y DISEÑO DE INFORMACION Sistema de Seguimiento Académico Escolar UML Abril 2012 <<http://grupo13sistemaacademicoescolar.wordpress.com/2-5-uml/>> [consulta: 21 octubre 2013].
- [PORTELLA, 2011] Metodologías para el desarrollo de aplicaciones web: UWE <http://jorgeportella.files.wordpress.com/2011/11/analisis-diseo-y-desarrollodeaplicacionesweb.pdf>
- [ECURED, POWER] Power Designer <http://www.ecured.cu/index.php/PowerDesigner>
- [ECURED, ARCHITECT] Architect http://www.ecured.cu/index.php/Enterprise_Architect



ANEXOS

Datos de la Institución

El Gobierno Autónomo Municipal de La Paz es uno de los más importantes en el territorio nacional, lo que hace que su influencia geopolítica alcance al sur de la República del Perú y al Norte de la República de Chile.

A nivel del departamento de La Paz es el primer municipio de los 80 municipios que se tiene.

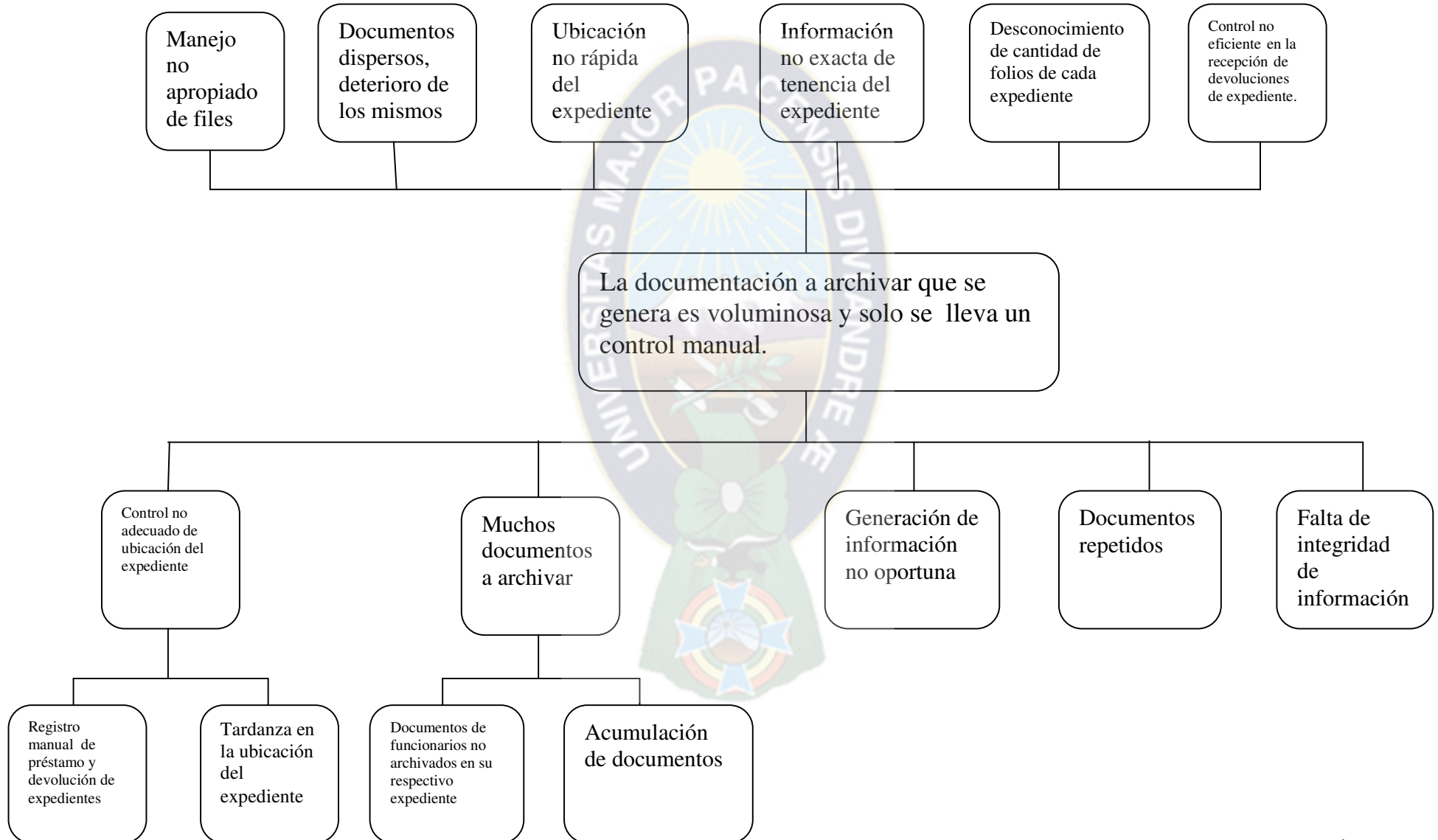
Está compuesto por 11 concejales que son elegidos proporcionalmente de acuerdo al resultado de las elecciones que se realizan cada 5 años.

El municipio está organizado en 23 Distritos y 8 Macro distritos, cuenta con 9 sub alcaldías.

Tiene oficinas que se encuentran ubicadas en diferentes lugares de la ciudad de La Paz, siendo su principal dirección el Palacio Consistorial que se encuentra en la calla Mercado, entre calle Ayacucho y Calle Colón.

El organigrama nos muestra como está estructurado el G.A.M.L.P., en sus diferentes áreas.

ARBOL DE PROBLEMAS



ARBOL DE OBJETIVOS

