

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



PROYECTO DE GRADO

**SISTEMA DE INFORMACIÓN PARA EL SEGUIMIENTO Y CONTROL DE LOS
PRÉSTAMOS DEL ARCHIVO DE LA UNIDAD CATASTRO
“SISCPAUC”**

**PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA
MENCION: INGENIERIA DE SISTEMAS INFORMATICOS**

**POSTULANTE: EDSON RODRIGO FERNANDEZ APAZA
TUTOR: MG. SC. FRANZ CUEVAS QUIROZ
REVISOR: LIC. VICTORIA HURTADO SERRUTO**

**LA PAZ – BOLIVIA
2006**

***D**edico este proyecto a mis Padres quienes con su amor inquebrantable, sus consejos, apoyo y comprensión me otorgaron la oportunidad de crecer profesionalmente, estando siempre presentes en todos mis logros y metas alcanzadas; a mis hermanas que con su cariño, paciencia y comprensión siempre me dieron su apoyo; por último a la persona que es muy especial e importante en mi vida a Lizeth con quien pasamos muchos momentos buenos y malos decirte mi niña ¡Por fin lo logre! gracias por todo tu apoyo y amor.....Este éxito es de y para todos ellos.*

AGRADECIMIENTOS

Agradezco a:

A todos los docentes que transmitieron todos sus conocimientos forjando así mi formación profesional.

A la Licenciada Victoria Hurtado Serruto por brindarme su tiempo, sus consejos comprensión, asesoramiento y apoyo incondicional.

Al Gobierno Municipal de La Paz y a la Unidad de Catastro por el tiempo otorgado para el desarrollo y conclusión del presente proyecto; en especial Jefe de Unidad Arq. Salim Yapur por la comprensión y el apoyo brindado para la culminación del presente trabajo.

A los Arquitectos Gustavo Siles, Angela Alabi y Licenciado Carlos Juanes por el tiempo otorgado y brindarme la información requerida para la culminación de este trabajo.

Al Licenciado Franz Cuevas Quiroz que con su conocimiento profesional y asesoramiento apporto a la culminación de este proyecto.

RESUMEN

El Archivo de la Unidad de Catastro de la Oficialía Mayor de Gestión Territorial perteneciente al Gobierno Municipal de La Paz, tiene por objeto el registro, clasificación y custodia de la documentación predial de la ciudad de La Paz. Unidad donde se detectó varios problemas: insuficiente control y seguimiento de muchos procesos importantes, manejo de la información de manera no organizada, muchos procesos manuales que ocasionaban la carencia de información oportuna e inmediata sobre préstamos, devoluciones de carpetas ,planimetrías, registro de trámites y el inventario de carpetas.

El objetivo principal del actual proyecto es la implementación de un sistema de seguimiento y control de los préstamos y devoluciones de carpetas y planimetrías, registro de salida e ingreso de trámites, inventario de carpetas (Vigentes, Propiedad Horizontal PH, ILIS, PRAC I, PRACII), registro de carpetas de planimetrías y planimetrías individuales, así como el inventariado tanto manual como automático de las carpetas, lo que ha permitido el mejoramiento de la administración de la información en respuesta a los requerimientos del archivo y de la unidad. En si el sistema actual se desarrolló con la metodología orientada a objetos RUP (Proceso Unificado de Desarrollo).

La implementación del “Sistema de Información para el seguimiento y control de los préstamos del Archivo de la Unidad de Catastro” (SISCPAUC) coadyuva de gran manera el manejo de la información, ya que posee una base de datos centralizada en un servidor al cual se puede acceder desde el archivo y las demás áreas, además de tener incorporado un agente de interfaz que hace que el sistema sea mas dinámico y agradable para el usuario, así mismo el entorno del sistema se lo realizó de tal manera que el aprendizaje por parte del usuario sea fácil y rápido.

INDICE

	Pag.
CAPITULO I	
MARCO INTRODUCTORIO	1
1.1. Introduccion	1
1.2. Antecedentes	2
1.3. Análisis y planteamiento del problema	4
1.3.1. Descripción de la problemática general	5
1.3.2. Definición del problema	6
1. 4. Objetivo general	6
1.4.1. Objetivos específicos	7
1.5. Justificación	7
1.5.1. Justificacion tecnica	8
1.5.2 Justificación social	8
1.6. Métodos y técnicas	8
1.7. Alcances y aportes	9
1.7.1. Alcances	9
1.7.2. Aportes	9
CAPITULO II	
FUNDAMENTO TEORICO	10
2.1. Introducción	9
2.2. Archivos	10
2.3. Metodología para el desarrollo del sistema	11
2.3.1. Lenguaje de modelado unificado UML	11
2.3.2. Proceso Unificado de Modelado (RUP)	13
2.4. Herramientas de pruebas y calidad de software	19
2.4.1. Pruebas	19
2.4.2. Calidad de software	20
2.4.2.1. Que es calidad de software?	20
2.4.2.2. Funcionalidad	21
2.4.2.3. Mantenibilidad	24
2.4.2.4. Confiabilidad	25
2.4.2.5. Portabilidad	26

2.5. Tecnologías.....	27
2.5.1. Visual Basic .NET 2003 (VB .NET).....	27
2.5.2. NET FRAMEWORK.....	27
2.6. Agente de interfaz	29
CAPITULO III	
PROCESO DE INVESTIGACIÓN O METODOLOGIA.....	30
3.1 Introducción	30
3.2 Fase de inicio.....	30
3.2.1 Listado de requerimientos del sistema.....	31
3.2.1.1. Requerimientos tecnológicos.....	32
3.2.2 Modelo del negocio	33
3.2.2.1 Modelo de casos de uso del negocio.....	33
3.2.2.2 Modelo de objetos	36
3.3 Fase de elaboración.....	38
3.3.1 Modelo de casos de uso.....	38
3.3.1.1 Definición de diagrama de casos esenciales	39
3.3.1.2 Descripción de casos de uso	44
3.3.2 Modelo de análisis.....	50
3.3.3 Modelo de diseño	56
3.3.3.1 Definición de diagramas de secuencia.....	56
3.3.3.2 Definición de diagramas de estado.....	61
3.3.3.3 Definición del modelado de la bases de datos	64
3.3.3.4 Diseño de la arquitectura del sistema	66
3.4 Fase de construcción	67
3.4.1 Modelo de despliegue del sistema.....	67
3.4.1.1 Definición del diagrama de componentes	67
3.4.1.2 Definición del diagrama de despliegue	70
3.4.2 Implementación	71
3.4.2.1. Interfaces de usuario	71
3.5 Fase de transición.....	82
3.5.1. Modelo de pruebas.....	82
3.5.1.1 Prueba de interfaces gráficas de usuario.....	83
3.5.1.2. Prueba de caja negra	87

CAPITULO IV	89
CALIDAD DE SOFTWARE	89
4.1. Introducción	89
4.2. Confiabilidad del software	89
4.3. Funcionalidad del software	91
4.4. Mantenibilidad de software	94
4.5. Portabilidad del software	95
4.5.1. Facilidad de instalación	95
CAPITULO V	97
CONCLUSIONES Y RECOMENDACIONES	97
5.1. Conclusiones.....	97
5.2. Recomendaciones	99
BIBLIOGRAFÍA	101
Referencias bibliograficas	101
Referencias de Internet	102

ANEXO A	ORGANIGRAMA DE LA EMPRESA
ANEXO B	ARBOL DE PROBLEMAS
ANEXO C	ARBOL DE OBJETIVOS

INDICE DE FIGURAS

	Pag.
Figura 2.1. Relación de flujo de trabajos y fases en RUP.....	15
Figura 2.2. Proceso Unificado de desarrollo.....	19
Figura 2.3. Factor de ponderación.....	23
Figura 2.4. Interacción del usuario con la aplicación y el agente.....	29
Figura 3.1. Diagrama de casos de uso del negocio.....	34
Figura 3.2. Diagrama de casos de uso de solicitud de carpetas y/o planimetrías.....	34
Figura 3.3. Diagrama de casos de uso de préstamo de carpetas y/o planimetrías....	35
Figura 3.4. Diagrama de casos de uso de devolución de carpetas y/o planimetrías...	35
Figura 3.5. Diagrama de casos de uso de registro de trámites.....	36
Figura 3.6. Modelo de objetos. Préstamo de carpetas y/o planim (Funcionario int.)...	36
Figura 3.7. Modelo de objetos. Devolución de carpetas y/o planimetrías.....	37
Figura 3.8. Modelo de objetos. Préstamos de carpetas y /o planim (Fun. externos)...	37
Figura 3.9. Modelo de objetos. Registro de trámites nuevos.....	38
Figura 3.10. Casos de uso esenciales del sistema SISCPAUC.....	39
Figura 3.11. Diagrama de casos de uso: Solicitudes de carpetas y/o planimetrías.....	41
Figura 3.12. Diagrama de casos de uso: Recepción de solicitudes de carp y/o Planim.....	42
Figura 3.13. Diagrama de casos de uso: Devolución de carpetas y/o planimetrías....	42
Figura 3.14. Diagrama de casos de uso: Administración de seguridad.....	42
Figura 3.15. Diagrama de casos de uso: Generación de Reportes.....	43
Figura 3.16. Diagrama de casos de uso: Mantenimiento.....	43
Figura 3.17. Diagrama de casos de uso: Registro de ingresos y salidas de trámites..	43
Figura 3.18. Diagrama de casos de uso: Opciones de Agente.....	44
Figura 3.19. Realización de casos de uso-análisis: Solicitud de carpetas y/o planim..	51
Figura 3.20. Realización de casos de uso-análisis: Recepción de solicitudes de carp.....	51
Figura 3.21. Realización de casos de uso - análisis: Devolución de carp. y/o planim...	52
Figura 3.22. Realización de casos de uso - análisis: Administración de seguridad.....	52
Figura 3.23. Realización de casos de uso - análisis: Generación de reportes.....	53
Figura 3.24. Realización de casos de uso - análisis: Mantenimiento.....	54

Figura 3.25. Realización de casos de uso-análisis: Registro de ingr. y salidas de tram	55
Figura 3.26. Realización de casos de uso - análisis: Opciones de agente.....	55
Figura 3.27. Diagrama de secuencia: Solicitudes de carpetas y/o planimetrías.....	57
Figura 3.28. Diagrama de secuencia: Recepción de solicitudes de carpetas y/o planimetrías.....	57
Figura 3.29. Diagrama de secuencia: Devolución de solicitudes de carp. y/o planim...	58
Figura 3.30. Diagrama de secuencia: Administración de seguridad.....	58
Figura 3.31. Diagrama de secuencia: Generación de reportes.....	59
Figura 3.32: Diagrama de secuencia: Mantenimiento.....	59
Figura 3.33. Diagrama de secuencia: Registro de ingreso y salidas de trámites.....	60
Figura 3.34. Diagrama de secuencia: Opciones de agente.....	60
Figura 3.35. Diagrama de estado: Solicitudes de carpetas y/o planimetrías.....	61
Figura 3.36. Diagrama de estado: Recepción de solicitudes de carpetas y/o planim	61
Figura 3.37. Diagrama de estado: Devolución de solicitudes de carpetas y/o planim..	62
Figura 3.38. Diagrama de estado: Administración de seguridad.....	62
Figura 3.39. Diagrama de estado: Generación de reportes.....	63
Figura 3.40. Diagrama de estado: Mantenimiento.....	63
Figura 3.41. Diagrama de estado: Registro de ingreso y salidas de trámites nuevos..	64
Figura 3.42. Diagrama de estado: Opciones de agente.....	64
Figura 3.43. Modelado de la base de datos de SISCPAUC.....	65
Figura 3.44. Subsistemas y algunas de las dependencias iniciales del sistema.....	66
Figura 3.45. Diagrama de componentes del sistema SISCPAUC.....	67
Figura 3.46. Diagrama de componentes: Componentes comunes.....	68
Figura 3.47. Diagrama de componentes: Préstamo.....	68
Figura 3.48. Diagrama de componentes: Administración de seguridad.....	69
Figura 3.49. Diagrama de componentes: Mantenimiento.....	69
Figura 3.50. Diagrama de componentes: Registro ingresos y salidas de trámites.....	70
Figura 3.51. Diagrama de componentes: Opciones de agente.....	70
Figura 3.52. Diagrama de despliegue del SISCPAUC.....	71
Figura 3.53. Splash de inicio de SISCPAUC.....	71
Figura 3.54. Formulario principal SISCPAUC.....	72
Figura 3.55. Formulario de autenticación.....	73
Figura 3.56. Formulario de Información del servidor.....	73

Figura 3.57. Formulario de Solicitudes.....	74
Figura 3.58. Formulario de Recepción de Solicitudes.....	75
Figura 3.59. Formulario de Boleta de préstamo.....	76
Figura 3.60. Formulario de Devoluciones.....	77
Figura 3.61. Boleta de devolución.....	77
Figura 3.62. Formulario de Registro de Trámites.....	78
Figura 3.63. Formulario de Inventariado de carpetas.....	79
Figura 3.64. Formulario de Carpeta de planimetrías.....	80
Figura 3.65. Formulario de Carpeta de planimetrías.....	81
Figura 3.66. Formulario de Registro de usuarios – Datos personales.....	81
Figura 3.67. Formulario de Registro de usuarios – Datos de cuentas y asignación de permisos.....	82
Figura 3.68. Prueba de la caja negra para registro de trámites.....	88

INDICE DE TABLAS

	Pag.
Tabla 1.1. Tipos de carpetas.....	3
Tabla 2.1. Diferencias entre: Almacén de papel y Archivo de documentos.....	11
Tabla 3.1. Listado de requerimientos del sistema.....	32
Tabla 3.2. Especificación de actores: Administrador del sistema.....	40
Tabla 3.3. Especificación de actores: Archivista.....	40
Tabla 3.4. Especificación de actores: Funcionario Interno.....	40
Tabla 3.5. Especificación de actores: Funcionario Externo.....	41
Tabla 3.6. Descripción de caso de uso: Solicitud de carpetas y/o planimetrías.....	45
Tabla 3.7. Descripción de caso de uso: Recepción solicitudes de carpetas y/o planimetrías.....	46
Tabla 3.8. Descripción de caso de uso: Devolución de carpetas y/o planimetrías....	46
Tabla 3.9. Descripción de caso de uso: Administración de seguridad.....	47
Tabla 3.10. Descripción de caso de uso: Generación de reportes.....	48
Tabla 3.11. Descripción de caso de uso: Mantenimiento.....	49
Tabla 3.12. Descripción de caso de uso: Registro de ingresos y salidas de trámites...	49
Tabla 3.13. Descripción de caso de uso: Opciones de Agente.....	50
Tabla 3.14. Listado de preguntas y respuestas para las pruebas IGU.....	86
Tabla 4.1. Fallas por cada mil líneas de código.....	90
Tabla 4.2. Cálculo de puntos de función.....	91
Tabla 4.3. Escala de evaluación para los factores.....	92
Tabla 4.4. Valores de ajuste de complejidad.....	93
Tabla 4.5. Información requerida por el IMS.....	95

CAPITULO I

MARCO INTRODUCTORIO

1.1. INTRODUCCION

En la actualidad se exige que las organizaciones e instituciones mejoren el uso de tecnología y sus directivos incrementen sus conocimientos a la par del desarrollo tecnológico, donde cada tarea manual es automatizada.

La Unidad de Catastro de la Oficialía Mayor de Gestión Territorial que pertenece al Gobierno Municipal de La Paz cuenta con un archivo (ver organigrama en el Anexo A), cuyo objetivo primordial es el registro, clasificación, custodia de documentación predial y así también la emisión de Certificación Catastral, duplicado de la certificación catastral y cambio de nombre [GMLP, 2005]; para realizar dichas tareas se realizan préstamos de los documentos existentes en el archivo sin un óptimo control y seguimiento del mismo, un registro manual de trámites entrantes a la unidad.

Luego de un diagnóstico preliminar se pudo determinar que los procesos manuales generan: gran pérdida de tiempo, bajo rendimiento del control y

seguimiento de la información es debido a ello que se ha propuesto desarrollar el actual proyecto donde:

- ❖ El Capítulo I, proporciona un estudio preliminar de la problemática del Archivo de la Unidad de Catastro, lo que ha permitido definir los objetivos en el presente proyecto.
- ❖ El Capítulo II, provee los fundamentos teóricos acerca de que herramientas, teorías y métodos fueron empleados para el desarrollo del presente proyecto de grado.
- ❖ El Capítulo III muestra el desarrollo de todo modelado del sistema mediante el Proceso Unificado de Modelado (RUP) que es una metodología orientada a objetos la cual se sirve del Lenguaje de Modelado Unificado (UML).
- ❖ En el Capítulo IV se realiza el respectivo control de calidad

1.2. ANTECEDENTES

En la actualidad, en el ámbito de los Archivos se produjo una evolución hacia la aplicación de técnicas de almacenamiento y tratamiento de datos referidos a los documentos, ya sea para realizar un estudio de sus características o para tener un acceso a copias de dichos documentos sin poner en peligro la conservación física de los originales.

Uno de los primeros pasos, para la conservación de documentos en Archivos es el que impulsó en España el rey Carlos III que fue el fundador del Archivo General de Indias en el último tercio del siglo XVIII. Siguiendo esta línea

se crearon otros archivos en diferentes países asumiendo la importancia de tener un lugar donde preservar y organizar la información acerca de los documentos.

En nuestro medio el catastro urbano ha sido administrado por diferentes organismos desde aproximadamente 1930, en la actualidad es parte del Gobierno Municipal de La Paz donde el 75% de la información catastral esta registrada en forma manual y tan solo un 25% digitalizada.

Entre los trámites que realiza la unidad de Catastro tenemos la certificación catastral, cambio de nombre y el duplicado de la certificación catastral, a medida que ha ido avanzando el tiempo el volumen de la documentación llegó a ser muy grande; además dicha documentación está organizada en carpetas con códigos catastrales y clasificándose en distintos tipos de carpetas (véase la Tabla 1.1).

CARPETA	DESCRIPCION
MAU	<i>Ministerio de asuntos urbanos</i> , son carpetas que tienen registros de información predial, siendo estas uno de los primero levantamientos de información territorial, por esa razón es considerada información histórica.
PRAC CIM y PRAC II	<i>Proyecto de Recatastro del Anillo Cartográfico</i> , dichas carpetas contienen información de un relevamiento de datos de un gran numero de predios que aun faltan por concluir.
ILIS	Estas siglas representan a una empresa Holandesa que realizó un proyecto con el propósito de mejorar el rendimiento de servicio que realiza la unidad
VIGENTES	Las cuales contienen datos de información actual de todos los predios que tengan su certificado catastral.
PH	Contienen documentación también actual de Propiedades Horizontales que contemplan edificios que tienen departamentos, jauleras y otros llamados subpredios.
PLANIMETRIAS	Contienen planos prediales de gran parte de la ciudad.

Tabla 1.1. Tipos de carpetas

Por otro lado, luego de una revisión bibliográfica en nuestro medio se han identificado algunos proyectos relacionados al presente trabajo:

- **“Sistema de Información de Archivo para la Dirección Distrital de Educación El Alto – Distrito Norte”**; Su objetivo principal fue automatizar la administración, control y consulta de la información académica generada en los 330 colegios del Distrito Norte, mediante la aplicación de reingeniería de procesos utilizando la Metodología Rápida Re de Manganelli y el paradigma de análisis y diseño orientado a objetos con la Metodología de Modelado y Diseño Orientado a Objetos de Rumbaugh, y la implantación se la realiza en un manejador de base de datos SQL Server y en el software de desarrollo Visual Basic [MENDOZA,2001].
- **“Sistema de Información del Archivo Central de la Corte de Justicia del Distrito de La Paz”**; Este sistema fue desarrollado para la Dirección de Archivo Central el cual está basado en la utilización del paradigma del ciclo de vida clásico, pues permite un desarrollo secuencial y sistemático, en las etapas de Análisis y Diseño hace uso de la Metodología Estructurada, aplicando el Modelo de comportamiento, la implantación utilizó como lenguaje Visual Basic y como gestor de base de datos SQL Server 7.0 para aplicar tecnología cliente-servidor [MAMANI,2002].

A diferencia de los proyectos anteriores el actual proyecto de grado con todos sus capítulos tiene como propósito desarrollar un Sistema de Información que será elaborado mediante un análisis y diseño orientado a objetos, el cual estará implementado en el lenguaje de programación Visual Basic .NET 2003, con el fin de automatizar el control de préstamos, registro de los trámites entrantes y brindar un inventariado actualizado de las carpetas existentes en el archivo

además de ofrecer al usuario informes o reportes que contengan información apropiada para la toma de decisiones y apoyo al trabajo realizado por el personal de la unidad.

1.3. ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA

1.3.1. DESCRIPCIÓN DE LA PROBLEMÁTICA GENERAL

El Archivo de la Unidad de Catastro actualmente es un elemento indispensable, por la función principal que tiene, que es la custodia de los registros catastrales que están contenidos en los distintos tipos de carpetas (ver la Tabla 1.1), con fines de consulta para la realización de trámites para certificado de registro catastral, cambio de nombre y duplicado de certificado de registro catastral que son realizados en la Unidad de Catastro, así también otros trámites que pertenecen a diferentes Direcciones y áreas del GMLP, como también en instituciones públicas del gobierno central y entidades privadas.

Con las consideraciones anteriormente realizadas a continuación se describen los problemas que se identificaron:

- El archivo de la Unidad de Catastro cuenta con procedimientos manuales que no siempre son confiables, esto acrecienta aun mas la necesidad y posibilidad de resolver el problema mediante un sistema computarizado.
- Debido al constante manipuleo de la documentación, estos están sujetos a cambios, alteraciones, extravíos y sustracciones.
- No existe un óptimo control y seguimiento de los préstamos y devoluciones de dichos documentos.

- Los procesos de registro para préstamos son realizados manualmente por el personal interno y externo a la unidad.
 - Existe rezago y pérdida de tiempo por funcionarios internos, por la realización de procesos manuales de registro.
 - El proceso de registro de trámites (desgloses) es realizado también manualmente.
 - No se cuenta con un inventario actualizado de las carpetas existentes en el archivo.
 - Existe una sobrecarga de trabajo en el Archivo de Catastro debido a los procesos manuales.
- Ver el árbol de problemas en ANEXO B.

1.3.2. DEFINICIÓN DEL PROBLEMA

¿El Sistema de Información a implantar mejorará la administración de la información generada por los procesos de préstamos de carpetas en la Unidad de Catastro de manera que la información proporcionada sea confiable y oportuna a los usuarios?

1. 4. OBJETIVO GENERAL

Implantar un sistema de información que permita el seguimiento y control de los préstamos de documentos del Archivo de la Unidad de Catastro así como el registro de trámites entrantes a la unidad, logrando el mejoramiento en la administración de la información generada en dicha unidad.

1.4.1. OBJETIVOS ESPECÍFICOS

- Automatizar el proceso de envío de solicitudes de carpetas y/o planimetrías para el préstamo respectivo.
- Automatizar el proceso de recepcionar las solicitudes de carpetas y/o planimetrías que sean enviadas por otros usuarios.
- Automatizar el proceso de devoluciones de carpetas y/o planimetrías.
- Facilitar el registro de trámites entrantes a la unidad.
- Realizar un inventario tanto automático y manual de los distintos tipos de carpetas (véase la Tabla 1.1).
- Implantar un agente de interfaz que brinde información al usuario de manera dinámica y que de alertas a cerca de tareas pendientes que tenga el usuario
- Diseñar entornos para el óptimo mantenimiento del sistema.
- Realizar la óptima administración de seguridad del sistema.
- Emitir reportes con información fiable que ayude a la toma de decisiones.

Ver árbol de objetivos en ANEXO C.

1.5. JUSTIFICACIÓN

Las Justificaciones de un proyecto establecen la importancia y el porque de

su creación en relación a los beneficios que cause a la sociedad y la institución.

1.5.1. JUSTIFICACION TECNICA

Ajustar a instituciones y entidades hacia las nuevas tendencias tecnológicas en el área de informática es de gran importancia para un óptimo desempeño y manejo de la información, en tal caso el Archivo de la Unidad de Catastro sigue este camino en su afán de brindar un mejor servicio. El presente proyecto de grado se justifica técnicamente por el uso de la tecnología .NET con el lenguaje de programación Visual Basic .NET que permite la creación de soluciones efectivas, así mismo la utilización del sistema gestor de base de datos SQL Server 2000 que coadyuvó cuantiosamente en el acceso y manipulación de la información.

1.5.2 JUSTIFICACIÓN SOCIAL

El nuevo Sistema de Información del Archivo de la Unidad de Catastro se justifica socialmente, por la ayuda que brinda indirectamente a las personas que realizan sus trámites en la Unidad de Catastro, especialmente disminuyendo el tiempo de espera en la entrega de estos trámites.

1.6. MÉTODOS Y TÉCNICAS

Las metodologías que se consideraron son las siguientes:

- El método científico como procedimiento o conjunto de procedimientos que se utiliza para obtener conocimientos científicos, el modelo de trabajo o pauta general que orienta la investigación, utilizando: marco teórico como parte referencial y didáctica, árbol de problemas y objetivos.

- Para el modelo de desarrollo se utiliza el Proceso Racional Unificado (RUP), permitiendo este un proceso iterativo incremental basado en el análisis y diseño orientado a objetos, utiliza la notación del Modelado de Lenguaje Unificado (UML).

1.7. ALCANCES Y APORTES

1.7.1. ALCANCES

El presente proyecto de grado se limita a la administración automatizada de la información del Archivo de la Unidad de Catastro.

Mediante el sistema el Archivo de la Unidad de Catastro se realiza el control y seguimiento de los préstamos de documentos, se efectúa un adecuado registro de los nuevos trámites ingresados a la unidad, se realiza el inventario tanto manual como automático de las carpetas existentes en el archivo, además incorpora un agente de interfaz para mayor dinamicidad del sistema, también se generaran reportes para los distintos tipos de procesos.

1.7.2. APORTES

El aporte más sobresaliente del presente proyecto de grado es brindar un óptimo control de los préstamos de las carpetas del archivo, además incorporar un agente que pueda informar al usuario de algunas tareas pendientes; también se llegó a descentralizar de gran manera el trabajo realizado por el Archivo de la Unidad de Catastro, efectuando el desarrollo mediante la aplicación de métodos y utilización de tecnologías para su automatización.

CAPITULO II

FUNDAMENTO TEORICO

2.1. INTRODUCCION

Este capítulo proporcionará los fundamentos teóricos acerca de las herramientas, teorías y métodos empleados para el desarrollo del presente proyecto de grado.

2.2. ARCHIVOS

En nuestro medio existen muchos tipos de archivos los cuales almacenan distintos tipos de información, dependiendo del tipo de información así como de la forma en la que se manipula la información se identificaron el almacén de papel y archivo de documentos cuyas características de describen en la Tabla 2.1.

ALMACEN DE PAPEL	ARCHIVO DE DOCUMENTOS
Papeles apilados, diarios, boletines, Fotocopias.	Documentos importantes que hay que conservar clasificados y ordenados.

Pocas personas saben lo que hay.	Inventario de los documentos con la descripción correspondiente.
Los documentos llegan como pueden.	Traslado organizado.
Cada persona conserva o elimina lo que le parece oportuno.	Criterios únicos de conservación y eliminación.
Es difícil encontrar y consultar un documento.	Consulta y préstamo de la documentación de forma ágil.
Todo el mundo puede acceder.	Acceso para el personal autorizado.
No hay normas de archivo ni de clasificación.	Manual de gestión de documentación y archivos y cuadro de clasificación corporativo.

Tabla 2.1. Diferencias entre: Almacén de papel y Archivo de documentos

Los archivos también llegan a utilizar libros de registro que son un instrumento jurídico, cuya finalidad es conseguir un sistema de control y de garantía externa e interna de los documentos que se presentan en la Administración y de los documentos oficiales que se envían a otros órganos o particulares. El registro permite certificar la existencia de un documento aunque éste no se haya conservado.

Por otro lado la clasificación de la documentación, consiste en agrupar la documentación por conceptos o categorías dentro de una estructura jerárquica y lógica.

2.3. METODOLOGÍA PARA EL DESARROLLO DEL SISTEMA

2.3.1. LENGUAJE DE MODELADO UNIFICADO UML

UML es ante todo un lenguaje que proporciona un vocabulario y reglas para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema. Este lenguaje nos indica cómo crear y leer

los modelos, pero no dice cómo crearlos, siendo esto el objetivo de las metodologías de desarrollo.

Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones como sigue:

- Permite expresar de forma gráfica un sistema de manera que sea fácil de entender.
- Permite especificar cuáles son las características de un sistema antes de su construcción.
- A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Aunque UML está pensado para modelar sistemas complejos con gran cantidad de software, el lenguaje es suficientemente expresivo como para modelar sistemas que no son informáticos, como flujos de trabajo en una empresa, diseño de la estructura de una organización y por supuesto, en el diseño de hardware.

Un modelo UML esta compuesto por tres clases de bloques de construcción:

- Elementos: Los elementos son abstracciones de cosas reales o ficticias (objetos, acciones, etc.)
- Relaciones: relacionan los elementos entre sí.

- Diagramas: Son colecciones de elementos con sus relaciones [HERNANDEZ,2001].

2.3.2. PROCESO UNIFICADO DE MODELADO (RUP).

El proceso unificado de modelado actúa como un modelo que puede adaptarse a cualquier tipo de proyecto y empresa (grandes y pequeñas), en si RUP es una guía de cómo usar UML de la forma más efectiva. Las características del proceso unificado de modelado son:

- Centrado en los Modelos: Los diagramas son un vehículo de comunicación más expresivo que las descripciones en lenguaje natural. Se trata de minimizar el uso de descripciones y especificaciones textuales del sistema.
- Guiado por los casos de uso: Los casos de uso son el instrumento para validar la arquitectura del software y extraer los casos de prueba.
- Centrado en la arquitectura: Los modelos son proyecciones del análisis y el diseño constituye la arquitectura del producto a desarrollar.
- Iterativo e incremental: Durante todo el proceso de desarrollo se producen versiones incrementales (que se acercan al producto terminado) del producto en desarrollo.

El RUP describe cómo obtener los requerimientos, organizarlos, documentar requerimientos de funcionalidad y restricciones, rastrear y documentar decisiones; captar y comunicar requerimientos del negocio.

Los casos de uso y los escenarios indicados por el proceso han probado ser una buena forma de captar requerimientos y guiar el diseño, la implementación y las pruebas [JACOBSON, BOOCH, RUMBAUGH, 2000].

RUP divide el proceso de desarrollo en ciclos, teniendo un producto al final de cada ciclo y cada ciclo se divide en cuatro Fases:

a) Fase de inicio, esta fase establece la oportunidad y alcance el proyecto, también se identifican todas las entidades externas con las que se trata (actores) y se define la interacción a un alto nivel de abstracción [JACOBSON, BOOCH, RUMBAUGH, 2000]: identificar todos los casos de uso, describir algunos en detalle.

La oportunidad del negocio incluye:

- Criterios de éxito
- Identificación de riesgos
- Estimación de recursos necesarios
- Plan de las fases incluyendo hitos

b) Fase de elaboración, esta fase tiene objetivos: Establecer una arquitectura base sólida, desarrollar un plan de proyecto, analizar el dominio del problema, eliminar los elementos de mayor riesgo para el desarrollo exitoso del proyecto [JACOBSON, BOOCH, RUMBAUGH, 2000].

c) Fase de construcción, en esta fase todas las componentes restantes se desarrollan e incorporan al producto. Todo es probado en profundidad, se da énfasis en la producción eficiente y no ya en la creación intelectual.

Puede hacerse construcción en paralelo, pero esto exige una planificación detallada y una arquitectura muy estable [JACOBSON, BOOCH, RUMBAUGH, 2000].

d) Fase de transición, El objetivo de esta fase es traspasar el software desarrollado a la comunidad de usuarios, una vez instalado surgirán nuevos elementos que implicarán nuevos desarrollos (ciclos) [JACOBSON, BOOCH, RUMBAUGH, 2000].

En cada fase se ejecutarán una o varias iteraciones (de tamaño variable según el proyecto), cada fase concluye con un hito bien definido donde deben tomarse ciertas decisiones y dentro de cada una de ellas seguirá un modelo de cascada para los flujos de trabajo que son (ver Figura 2.1):

- Requisitos.
- Análisis.
- Diseño.
- Implementación y prueba.

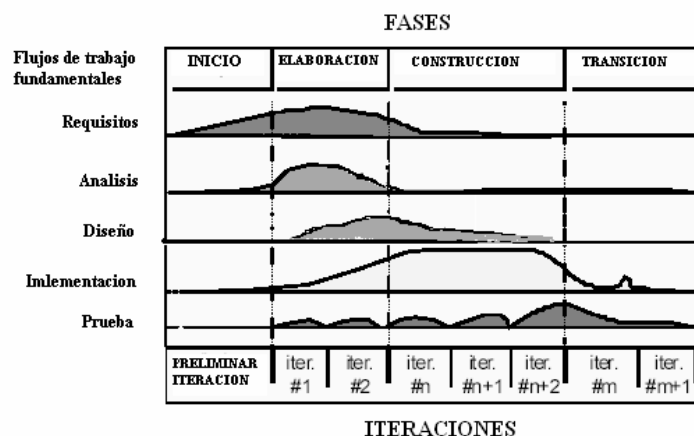


Figura 2.1. Relación de flujo de trabajos y fases en

a) Requisitos, los desarrolladores y clientes deben acordar qué es lo que el sistema debe hacer: Relevar requerimientos, documentar funcionalidad y restricciones, documentar decisiones, identificar actores, identificar casos de uso. Los casos de uso describen la funcionalidad y los requerimientos no funcionales se incluyen en una especificación complementaria [JACOBSON, BOOCH, RUMBAUGH, 2000].

El papel de los requisitos depende de la fase de desarrollo del software en la que nos encontremos.

Durante la fase de *inicio*, los analistas identifican la mayoría de los casos de uso para delimitar el sistema, alcance del proyecto y para detallar los más importantes (menos del 10%). Durante la fase de *elaboración*, los analistas capturan la mayoría de los requisitos restantes para que los desarrolladores puedan estimar el tamaño del esfuerzo de desarrollo que se requerirá. El objetivo es haber capturado un 80% de los requisitos y haber descrito la mayoría de los casos de uso; los requisitos restantes se capturan e implementan durante la fase de *construcción*; casi así no hay captura de requisitos en la fase de *transición*, a menos que haya requisitos que cambien [NAVARRO,2002].

b) Análisis, analizar los requisitos en forma de un modelo de análisis es importante, ya que, un modelo de análisis ofrece una especificación más precisa de los requisitos que el modelo de casos de uso; un modelo de análisis se describe utilizando el lenguaje de los desarrolladores, y puede por tanto introducir un mayor formalismo y ser utilizado para razonar sobre el funcionamiento interno del sistema; un modelo de análisis estructura los requisitos de un modo que facilita su comprensión, su preparación, su modificación, y en general, su mantenimiento. Un modelo de análisis puede

considerarse como una primera aproximación al modelo de diseño, aunque es un modelo por sí mismo.

El papel del análisis depende de la fase de desarrollo del software en la que nos encontremos. Las iteraciones iniciales de *elaboración* centran en el análisis, más adelante, al término de la fase de *elaboración* y durante la *construcción* el énfasis pasa a diseño e implementación, además, hay tres formas diferentes de considerar al modelo de análisis:

- Para describir los resultados del análisis y mantener la consistencia de este modelo durante todo el ciclo de vida del software.
- Para describir los resultados del análisis, considerando a este modelo como una herramienta transitoria e intermedia hasta el diseño.

El proyecto concibe los requisitos como parte integrada de la captura de requisitos o en el diseño. Suele aplicarse en el caso de tratarse de sistemas simples [NAVARRO,2002].

c) Diseño, en este flujo se modela el sistema y se le proporciona su forma para que soporte todos los requisitos. Una entrada fundamental del diseño es el modelo de análisis, que proporciona una comprensión detallada de los requisitos e impone una estructura al sistema. Los objetivos del diseño son:

- Adquirir una comprensión en profundidad de los aspectos relacionados con los requisitos no funcionales y restricciones técnicas.

- Crear una entrada apropiada y un punto de partida para actividades de implementación subsiguientes.
- Ser capaces de descomponer los trabajos de implementación en partes más manejables que puedan ser llevadas a cabo por diferentes equipos de desarrollo, teniendo en cuenta la posible concurrencia.

El diseño es el centro de atención al final de la fase de elaboración y el comienzo de las iteraciones de construcción, esto contribuye a una arquitectura estable y sólida, para crear un plano del modelo de implementación. Más tarde, durante la fase de construcción cuando la arquitectura es estable y los requisitos están bien entendidos, el centro de atención se desplaza a la implementación. [NAVARRO,2002].

d) Implementación, en este flujo empezamos con el resultado del diseño e implementamos el sistema en términos de componentes, es decir, ficheros de código fuente, *scripts*, ficheros de código binario, ejecutables y similares. Los propósitos de la implementación son:

- Planificar las integraciones de sistema necesarias en cada iteración.
- Distribuir el sistema asignando componentes ejecutables a nodos en el diagrama de despliegue.
- Implementar las clases y subsistemas encontrados durante el diseño.
- Probar los componentes individualmente e integrarlos.

La implementación es el centro durante las iteraciones de construcción, también se lleva a cabo trabajo de implementación durante la

fase de elaboración, para crear la línea base ejecutable de la arquitectura. Durante la fase de transición se puede tratar defectos tardíos, ya que el modelo de implementación denota la implementación actual del sistema en términos de componentes y subsistemas, es natural mantener el modelo de implementación a lo largo de todo el ciclo de vida del software.

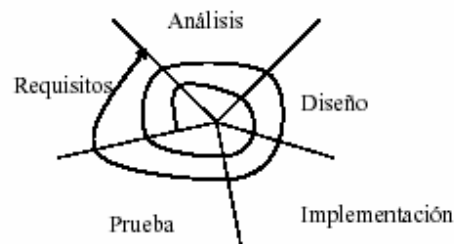


Figura 2.2. Proceso Unificado de desarrollo

Explicando un poco la Figura 2.2. tendríamos que cada vuelta en la espiral se denomina iteración, y la agrupación de iteraciones se las denomina fases. La agrupación de fases se las denomina ciclos donde cada ciclo concluye con una versión del producto [NAVARRO,2002].

2.4. HERRAMIENTAS DE PRUEBAS Y CALIDAD DE SOFTWARE

2.4.1. PRUEBAS

El proceso de prueba es la ejecución de un programa con el fin de encontrar errores.

Representan una revisión final de las especificaciones del diseño y de la implementación o codificación, Un caso de prueba es bueno cuando su ejecución conlleva una probabilidad elevada de encontrar un error y tiene éxito cuando se descubre un error no detectado hasta entonces.

Para cualquier producto de ingeniería existen dos enfoques a la hora de probar un producto:

Prueba de interfaces gráficas de usuarios (IGU), realiza pruebas para los entornos, arquitecturas y aplicaciones especializadas [PRESSMAN, 1998].

Prueba de Caja Negra, analiza principalmente la compatibilidad entre sí, en cuanto a las interfaces de cada uno de los componentes del software (no toma en cuenta la lógica del sistema). Combinando ambas estrategias se obtiene una completa validación del software de modo que se pueden detectar errores en las siguientes categorías [PRESSMAN, 1998]:

- Funciones incorrectas o ausentes
- Errores de interfaz.
- Errores en estructuras de datos o en acceso a base de datos externas.
- Errores de rendimiento, errores de inicialización y de terminación.

2.4.2. CALIDAD DE SOFTWARE

2.4.2.1. QUE ES CALIDAD DE SOFTWARE?

Definiciones de calidad del software:

- “Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente” [PRESSMAN, 1998].
- “El conjunto de características de una entidad que le confieren su aptitud para satisfacer las necesidades expresadas y las implícitas” ISO 8402 (UNE 66-001-92).

Además de las anteriores definiciones se puede decir que la calidad de software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. Donde la calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, mantenibilidad, portabilidad, usabilidad, seguridad e integridad [PRESSMAN,2001].

La calidad de software es medible y puede variar de un sistema a otro o de un programa a otro. Un producto de software para ser explotado durante un largo periodo (10 años o mas), necesita ser confiable, mantenible y flexible para disminuir los costos de mantenimiento y perfeccionamiento durante el tiempo de explotación.

De acuerdo al estándar ISO 9126 identifica seis atributos clave de calidad: funcionalidad, confiabilidad, usabilidad, eficiencia, facilidad de mantenimiento, portabilidad, subatributos. Estos factores no necesariamente son utilizados como medidas directas, facilitan una valiosa base para las medidas indirectas y una excelente lista para determinar la calidad de software.

2.4.2.2. FUNCIONALIDAD

Las métricas orientadas a la función son medidas indirectas del software y del proceso por el cual se desarrolla. En lugar de calcular las LDC (líneas de código), las métricas orientadas a la función se centran en la funcionalidad o utilidad del software, midiendo la aplicación desde la perspectiva del usuario dejando de lado los detalles de codificación. La medida de punto de función se propuso en 1979 por Albercht quien sugirió un acercamiento a la medida de la productividad; siendo ahora un estándar de ISO: ISO/IEC 20926:2003 [PRESSMAN,2001].

Los puntos de función se obtienen utilizando una función empírica basada en medidas cuantitativas del dominio de información del software y valoraciones subjetivos de la complejidad del software. Se determinan 5 características del ámbito de la información y los cálculos aparecen en la posición apropiada de la tabla. Los valores del ámbito de información están definidos de la siguiente manera; los puntos de función se calculan rellenando la tabla como se muestra en la Figura 2.3.

- **Número de entradas de usuario:** se cuenta cada entrada del usuario que proporcione al software diferentes datos orientados a la aplicación. Las entradas deben ser distinguidas de las peticiones que se contabilizan por separado.
- **Número de salidas del usuario:** se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto las salidas se refieren a informes, pantalla, mensajes de error. Los elementos de datos individuales dentro de un informe se cuentan por separado.
- **Número de peticiones al usuario:** una petición esta definida como una entrada interactiva que resulta de la generación de algún tipo de respuesta en forma de salida interactiva. Se cuenta cada petición por separado.
- **Número de archivos:** se cuenta cada archivo maestro lógico, o sea una agrupación lógica de datos que puede ser una parte en una gran base de datos o un archivo independiente.
- **Número de interfaces externas:** se cuentan todas las interfaces legibles por la maquina por ejemplo: archivos de datos, en cinta o discos que son utilizados para transmitir información a otro sistema.

Factor de ponderación						
Parámetro de medición	Cuenta		Simple	Medio	Complejo	
Número de entradas de usuario	<input type="text"/>	*	3	4	6	= <input type="text"/>
Número de salidas de usuario	<input type="text"/>	*	4	5	7	= <input type="text"/>
Número de peticiones de usuario	<input type="text"/>	*	3	4	6	= <input type="text"/>
Número de archivos	<input type="text"/>	*	7	10	15	= <input type="text"/>
Número de interfaces externas	<input type="text"/>	*	5	7	10	= <input type="text"/>
Cuenta total	—————→					<input type="text"/>

Figura 2.3 Factor de ponderación [PRESSMAN,2003]

Para calcular los puntos de función se utiliza la siguiente relación.

$$PF = Cuenta\ Total * [0.65 + 0.01 * \sum Fi]$$

Donde: *Cuenta Total* es la suma de todas las entradas de PF obtenidas de la tabla anterior.

Fi donde i puede ser de uno hasta 14 los valores de ajuste de complejidad basados en las respuestas a un listado de cuestiones, donde cada respuesta tomará un valor o factor en la escala de 0 a 5 tomando en cuenta lo siguiente:

0	1	2	3	4	5
Sin influencia	Incidental	Moderado	Medio	Significativo	Esencial

Los valores constantes de la ecuación anterior y los factores de peso aplicados en las encuestas de los ámbitos de información han sido determinados empíricamente, seguidamente se describen las preguntas:

1. ¿Requiere el sistema copias de seguridad y de recuperación fiables?
2. ¿Se requieren comunicaciones de datos?

3. **¿Existen funciones de procesamiento distribuido?**
4. **¿Es crítico el rendimiento?**
5. **¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?**
6. **¿Requiere el sistema entrada de datos interactiva?**
7. **¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples?**
8. **¿Se actualizan los archivos maestros de forma interactiva?**
9. **¿Son complejas las entradas, las salidas, los archivos o las peticiones?**
10. **¿Es complejo el procesamiento interno?**
11. **¿Se ha diseñado el código para ser reutilizable?**
12. **¿Están incluidas 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?**

Una vez evaluada la ecuación el valor que toma el PF indica si la funcionalidad del sistema es óptimo o no.

2.4.2.3. MANTENIBILIDAD

El estándar IEEE 982.1-1988 sugiere el índice de madurez del software (IMS) que proporciona una indicación de la estabilidad de un producto software basada en los cambios que ocurren con cada versión del producto. Con el IMS se determina la siguiente información:

- MT = Número de módulos en la versión actual.
- Fc = Número de módulos en la versión actual que se han cambiado.

- Fa = Número de módulos en la versión actual que se han añadido.
- Fe = Número de módulos en la versión actual que se han eliminado.

El índice de madurez del software se calcula de la siguiente manera:

$$\text{IMS} = [\text{MT} - (\text{Fc} + \text{Fa} + \text{Fe})] / \text{MT}$$

A medida que el IMS se aproxima a 1 el producto se empieza a estabilizar [PRESSMAN,2001].

2.4.2.4. CONFIABILIDAD

Se pretende que el software funcione de forma consistente y correcta durante largos períodos de tiempo que esté disponible cuando se lo necesita, y que sea reparado rápida y fácilmente si acaso falla. Formalmente se dice que la *confiabilidad del software* es la probabilidad de que un sistema opere sin fallas bajo determinadas condiciones para un intervalo de tiempo dado.

Los usuarios quieren que el software que vayan a utilizar tenga un funcionamiento correcto y los desarrolladores esperan que el número de defectos que existan puedan ser solucionados sin causar más problemas a medida que se lo va probando, esto hace que la confiabilidad en el software crezca y los tiempos entre fallas se hacen cada vez más grandes [PRESSMAN, 2001].

En esta sección, se examina la confiabilidad en relación a la incertidumbre, considerando una incertidumbre de tipo- 1, que refleja la incertidumbre sobre la utilización del sistema. Por lo tanto, en cualquier instante, el tiempo hasta la próxima falla es incierto y se lo puede considerar como una variable aleatoria .

Para estudiar esta variable aleatoria utilizaremos una distribución exponencial encausada en sus Pruebas con cero fallas, que deriva de una función de cifras de falla, se supone que el número de fallas en el instante t es igual a:

$$a e^{-b(t)}$$

Donde:

a, b son constantes.

Este modelo se puede utilizar para determinar por cuántas horas se debe probar un sistema a fin de satisfacer una meta de confiabilidad. Por lo tanto el modelo requiere tres entradas; el número proyectado promedio de fallas como objetivo (*fallas*), el número total de fallas observada en las pruebas (*fallas probadas*) y el número total de horas de ejecución de pruebas hasta la última falla (*horas hasta la última falla*). El cálculo de las horas necesarias de prueba para cero fallas es:

$$\frac{\text{Ln}[(\text{fallas})/(0,5 + \text{fallas})] * (\text{horas hasta última falla})}{\text{Ln}[(0,5 + \text{fallas}) / (\text{fallas probadas} + \text{fallas})]}$$

2.4.2.5. PORTABILIDAD

La portabilidad es la facilidad con la que el software puede ser llevado de un entorno a otro. Este criterio se subdivide: Facilidad de instalación, facilidad de ajuste, facilidad de adaptación al cambio [NUÑEZ,2005].

La portabilidad viene dado por la medida de la subcaracterística de Facilidad de instalación, que puede ser medido respondiendo a la pregunta: ¿El usuario o quien mantiene el software puede fácilmente instalar el software en un ambiente operacional?, teniendo la siguiente relación:

$$X = A / B$$

Donde:

A: es el número de instalaciones exitosas que el usuario realizó.

B: es el número total de instalaciones que realizó el usuario.

2.5. TECNOLOGIAS

2.5.1. VISUAL BASIC .NET 2003 (VB .NET)

El lanzamiento de la plataforma .NET de Microsoft viene acompañado de una nueva versión de Visual Basic que difiere sustancialmente de versiones anteriores. Además de mejorar muchas características (como el soporte para herencia), VB .NET se ha simplificado y modificado para funcionar con el .NET Framework. Son tantos los cambios que algunos programadores de VB consideran a VB.NET como un lenguaje enteramente nuevo.

En el lenguaje VB .NET encontrará toda la documentación sobre dicho lenguaje, incluyendo la totalidad de sus elementos lingüísticos.

2.5.2. .NET FRAMEWORK

El .NET Framework es una nueva plataforma informática que simplifica el desarrollo de aplicaciones en un entorno altamente distribuido como es Internet. El diseño de .NET Framework está enfocado a cumplir los objetivos siguientes:

- Proporcionar un entorno coherente de programación orientada a objetos, en el que el código de los objetos se pueda almacenar y ejecutar de forma local, ejecutar de forma local pero distribuida en Internet o ejecutar de forma remota.
- Proporcionar un entorno de ejecución de código que reduzca lo máximo posible la implementación de software y los conflictos de versiones.

- Ofrecer un entorno de ejecución de código que garantice la ejecución segura del mismo, incluso del creado por terceras personas desconocidas o que no son de plena confianza.
- Proporcionar un entorno de ejecución de código que elimine los problemas de rendimiento de los entornos en los que se utilizan secuencias de comandos o intérpretes de comandos.
- Ofrecer al programador una experiencia coherente entre tipos de aplicaciones muy diferentes, como las basadas en Windows o en el Web.
- Basar toda la comunicación en estándares del sector para asegurar que el código de .NET Framework se puede integrar con otros tipos de código.

El .NET Framework contiene dos componentes principales: Common Language Runtime y la biblioteca de clases de .NET Framework. Common Language Runtime es el fundamento de la tecnología. El motor de tiempo de ejecución se puede considerar como un agente que administra el código en tiempo de ejecución y proporciona servicios centrales, como la administración de memoria, la administración de subprocesos y la interacción remota, al tiempo que aplica una seguridad estricta a los tipos y otras formas de especificación del código que garantizan su seguridad y solidez. De hecho, el concepto de administración de código es un principio básico del motor de tiempo de ejecución. El código destinado al motor de tiempo de ejecución se denomina *código administrado*, a diferencia del resto de código, que se conoce como *código no administrado*. La biblioteca de clases, el otro componente principal de .NET Framework, es una completa colección orientada a objetos de tipos reutilizables que se pueden emplear para desarrollar aplicaciones que abarcan desde las tradicionales herramientas de interfaz gráfica de usuario (GUI) o de línea de comandos hasta las aplicaciones basadas en las innovaciones más recientes proporcionadas por ASP.NET, como los formularios Web Forms y los servicios Web XML, y también la creación de aplicaciones de escritorio Windows Forms [MICROSOFT,2003] .

2.6. AGENTE DE INTERFAZ

Un agente de interfase es un software cuasi-inteligente que asiste a un usuario cuando interactúa con una o más aplicaciones. La motivación es que se les pueda delegar tareas aburridas y laboriosas, esta categoría de agentes apoyan y proveen asistencia a su usuario.

Los agentes de interfaz poseen características distintivas al mantener una comunicación con usuarios humanos, el agente de interfase llegaría a ser un asistente personal que colabora con el usuario en su mismo entorno de trabajo, un vista de cómo es la interacción del usuario con la aplicación y el agente (ver la Figura 2.4).

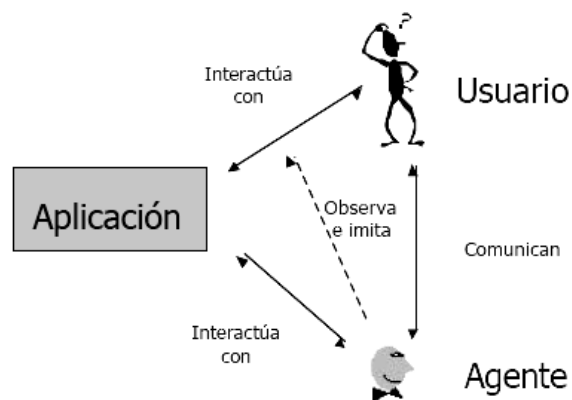


Figura 2.4. Interacción del usuario con la aplicación y el agente

CAPITULO III

PROCESO DE INVESTIGACIÓN O METODOLOGIA

3.1 INTRODUCCION

Este capítulo describe el sistema obtenido a la conclusión de la aplicación del proceso unificado del desarrollo del software que toma forma durante cada ciclo cumpliendo con las cuatro fases, como se muestra en la Figura 2.1, a través de una secuencia de modelos permitiendo visualizar lo que está sucediendo en esas fases pero se debe advertir que fueron logrados incrementalmente a través de iteraciones que dieron como resultado la obtención de un conjunto de artefactos: es decir modelos o documentos, que son versiones mejoradas del producto.

3.2 FASE DE INICIO

En esta fase se establece el contexto del sistema mediante el modelo del negocio a través de los casos de uso del Archivo de la Unidad de Catastro, logrando definir el alcance del proyecto. Para lograr tal efecto se debe identificar

las entidades externas con la que el sistema interactuará, es decir los actores, y definimos la interacción a un nivel alto; esto involucra realizar un análisis de requerimientos del usuario.

3.2.1 LISTADO DE REQUERIMIENTOS DEL SISTEMA

Después de realizar el análisis de requerimientos se logró obtener un listado de los requisitos más sobresalientes e importantes del Archivo de la Unidad de Catastro (ver Tabla 3.1.)

Características	Descripción	Estado	Prioridad	Nivel de riesgos
Préstamos físicos	Es un proceso por el cual funcionarios internos y externos pueden prestarse documentos físicos.	Incluido, Aprobado	Importante	Significativo
Registro de trámites nuevos	Registro de información nueva acerca de la salida e ingreso de trámites para la obtención de documentos generados por la unidad.	Incluido, Aprobado	Importante	Significativo
Mantenimiento de información.	Registro de carpetas automatizado y planimetrías (Inventario), configurar los criterios con los cuales se realizarán los préstamos	Incluido, Aprobado	Crítico	Crítico
Administrar usuarios	Registro de usuarios, los cuales accederán al sistema y también podrán realizar el préstamo de alguna carpeta, realizando así un control y seguimiento del proceso	Incluido, Aprobado	Importante	Significativo

	de préstamo.			
--	--------------	--	--	--

**Tabla 3.1. Listado de requerimientos del sistema
[Elaboración propia]**

3.2.1.1. REQUERIMIENTOS TECNOLÓGICOS

a) REQUERIMIENTO DE SOFTWARE

- Para la implementación del sistema SISCPAUC se utilizará el lenguaje de programación VISUAL BASIC .NET 2003, por las siguientes razones:
- El Gobierno Municipal de La Paz cuenta con la licencia respectiva de este lenguaje.
- Este lenguaje de programación no es muy complejo en su sintaxis, el interfaz de desarrollo es muy amigable ya que tiene muchos componentes que facilitan el entendimiento y el uso de sus propiedades, además que esta versión es totalmente orientado a objetos.
- La arquitectura que se utilizará es Cliente/Servidor ya que es el mas aconsejable por el tipo de sistema de que desarrollará; siendo muy beneficiosa porque utilizará el motor de base de datos SQL Server 2000, la cual cuenta también con la licencia respectiva, además de tener múltiples opciones y cuenta con una seguridad de manipulación

b) REQUERIMIENTO DE HARDWARE

El hardware que llegará a requerir el presente sistema es:
Servidor Central con la que cuenta la Unidad de Catastro.
Pentium IV, con procesador 1.20 GHz.

Memoria RAM de 256 Mb.

Disco duro de 40 GB.

Así mismo se conoce que los funcionarios de la Unidad de Catastro, cuentan con computadoras que tienen las siguientes características:

Pentium IV, con procesador de 1.20 GHz como mínimo

Memoria RAM de 128 MGb como mínimo.

Disco duro de 40 Gb como mínimo.

Como plataforma de software el Sistema Operativo Windows XP, que cuenta con una gran variedad de características y posibilidades que optimizan el rendimiento y proporciona una alta disponibilidad de recuperación de fallos.

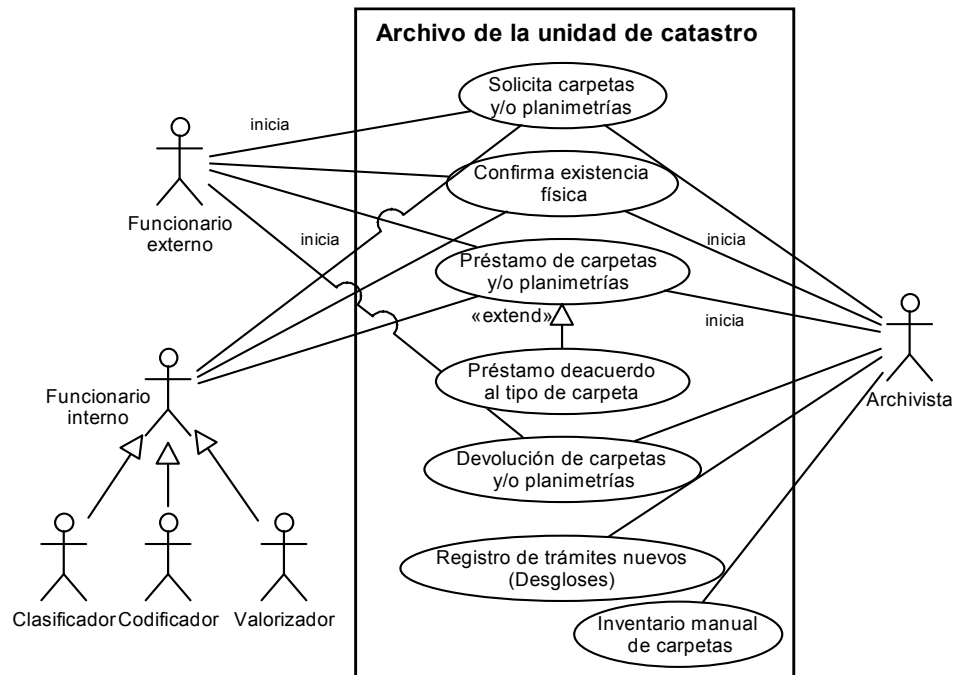
La red Intranet con la que cuenta la Unidad de Catastro se utilizará como entorno de red para el sistema.

3.2.2 MODELO DEL NEGOCIO

El modelado del negocio tiene como propósito primordial lograr comprender los objetivos de los procesos existentes u observados en el Archivo de la unidad de catastro, mediante el modelo de casos de uso del negocio y el modelo de objetos, que se describe con el diagrama de casos de uso del negocio y con el modelo de objetos que es un modelo interno del negocio.

3.2.2.1 MODELO DE CASOS DE USO DEL NEGOCIO

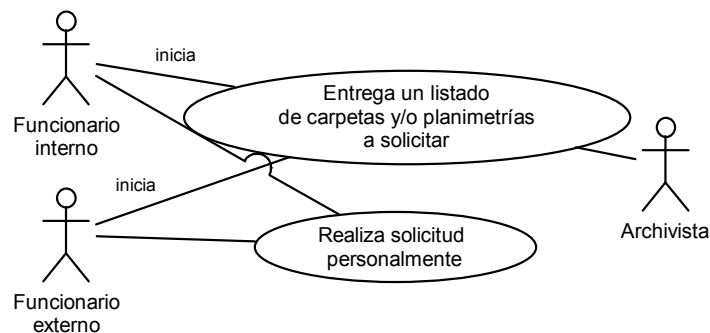
Para efectuar este modelado a continuación se representarán los procesos y los actores del negocio en un diagrama de casos de uso, ver figura 3.1



**Figura 3.1 Diagrama de casos de uso del negocio
[Elaboración propia]**

a) DESARROLLO DE LOS CASOS DE USO DEL NEGOCIO

Identificados los procesos del negocio se desarrollan y se especifican cada uno de ellos a través de diagramas de casos de uso del negocio, descritas en la Figura 3.2, Figura 3.3, Figura 3.4 y Figura 3.5.



**Figura 3.2 Diagrama de casos de uso de solicitud de carpetas y/o planimetrías
[Elaboración propia]**

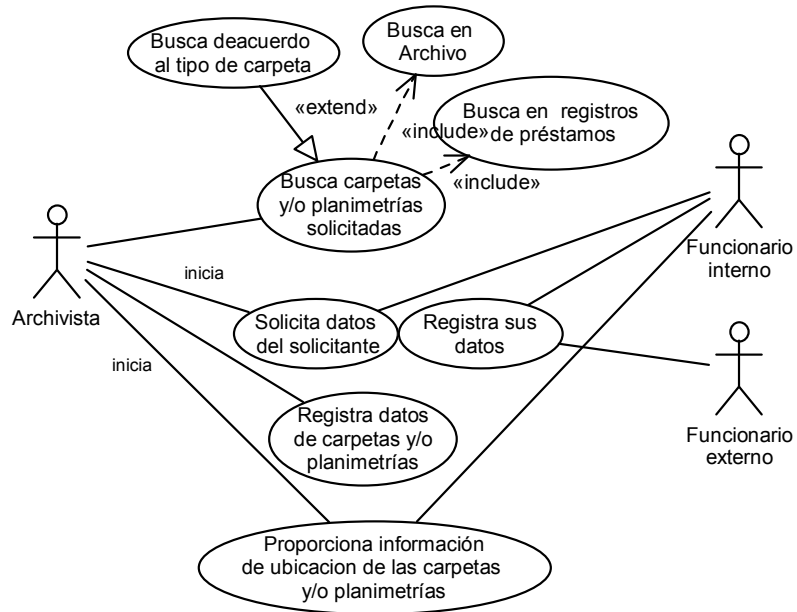


Figura 3.3 Diagrama de casos de uso de préstamo de carpetas y/o planimetrías [Elaboración propia]

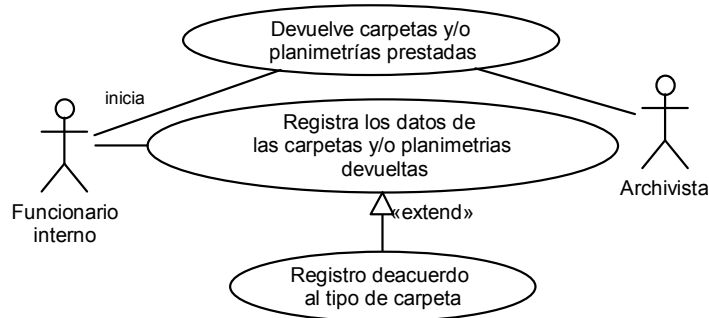


Figura 3.4 Diagrama de casos de uso de devolución de carpetas y/o planimetrías [Elaboración propia]

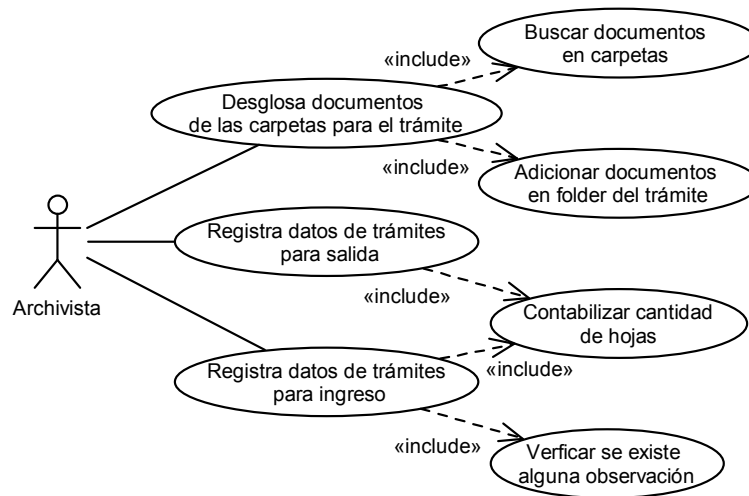


Figura 3.5 Diagrama de casos de uso de registro de trámites [Elaboración propia]

3.2.2.2 MODELO DE OBJETOS

En el modelo de objetos se describirá internamente cada caso de uso del negocio, los cuales son llevados a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y de unidades de trabajo, como se puede observar en la Figura 3.6, Figura 3.7, Figura 3.8 y Figura 3.9.

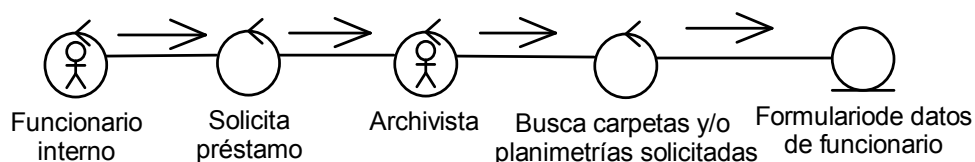
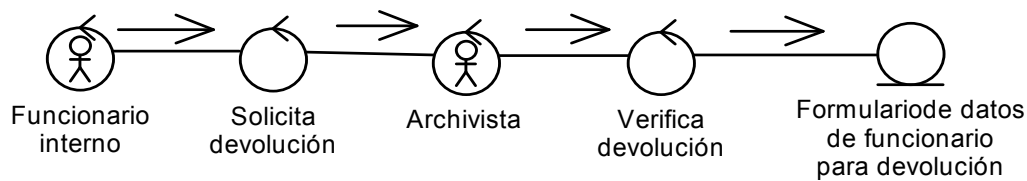


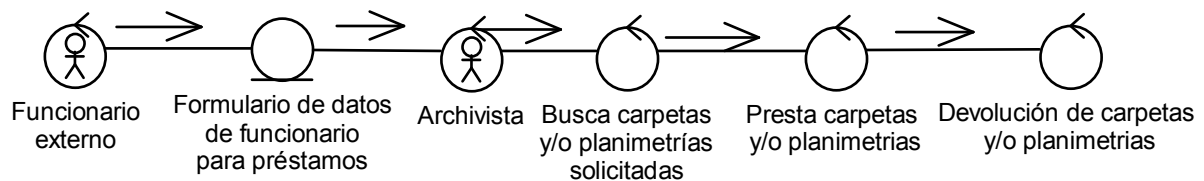
Figura 3.6 Modelo de objetos. Préstamo de carpetas y/o planimetrías (Funcionario interno) [Elaboración propia]

La Figura 3.6 describe el proceso del préstamo el cual se realizaba sin el sistema, donde un funcionario interno realiza una solicitud al archivista y este efectúa la tarea de buscar las solicitudes de las carpetas o planimetrías, posteriormente se llena los datos del funcionario en un formulario de préstamos.



**Figura 3.7 Modelo de objetos. Devolución de carpetas y/o planimetrías
[Elaboración propia]**

La Figura 3.7 describe el proceso de devolución el cual se realizaba sin el sistema, donde un funcionario interno realiza una solicitud de devolver las carpetas o planimetrías que tenía en su poder al archivista y este efectúa la verificación respectiva, posteriormente se llenan los datos del funcionario en un formulario de préstamos.



**Figura 3.8 Modelo de objetos. Préstamos de carpetas y/o planimetrías
(Funcionarios externos)
[Elaboración propia]**

La Figura 3.8 describe el proceso que debe seguir un funcionario externo para un préstamo para consultas el cual se realizaba sin el sistema, donde un funcionario externo solicitaba consultar carpetas o planimetrías al archivista, mediante un formulario que debía ser llenado, el archivista busca dichas carpetas o planimetrías, luego se realizaba el préstamo, posteriormente después de haber sido consultadas por el funcionario externo son devueltas.

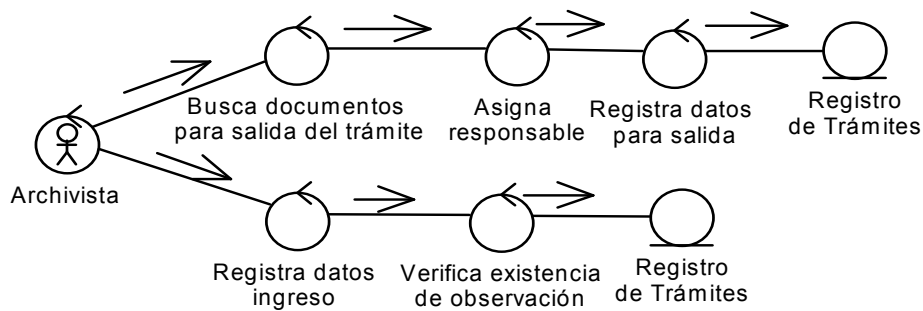


Figura 3.9 Modelo de objetos. Registro de trámites nuevos [Elaboración propia]

La Figura 3.9 describe el proceso por el cual se realizaba el registro de trámites nuevos existen dos casos, primero se realiza la búsqueda de documentos para sacar el desgloses correspondiente del trámite para la salida , se le asigna un responsable de dicho trámite, luego se registran los datos propios de la salida; una vez terminado el trámite estos documentos vuelven a ingresar al archivo (son devueltos al archivo), se registra los datos del ingreso o devolución, posteriormente se verifica si existió una observación.

3.3 FASE DE ELABORACION

Al comienzo de esta fase se tiene conocimiento del negocio y su funcionamiento, información significativa para el desarrollo de la fase de elaboración, seguidamente puntualizará en la mayoría de los casos de uso y se diseñará la arquitectura del sistema. Por tanto la arquitectura se expresará en vistas arquitectónicas del modelo de casos de uso, del modelo de análisis, del modelo de diseño.

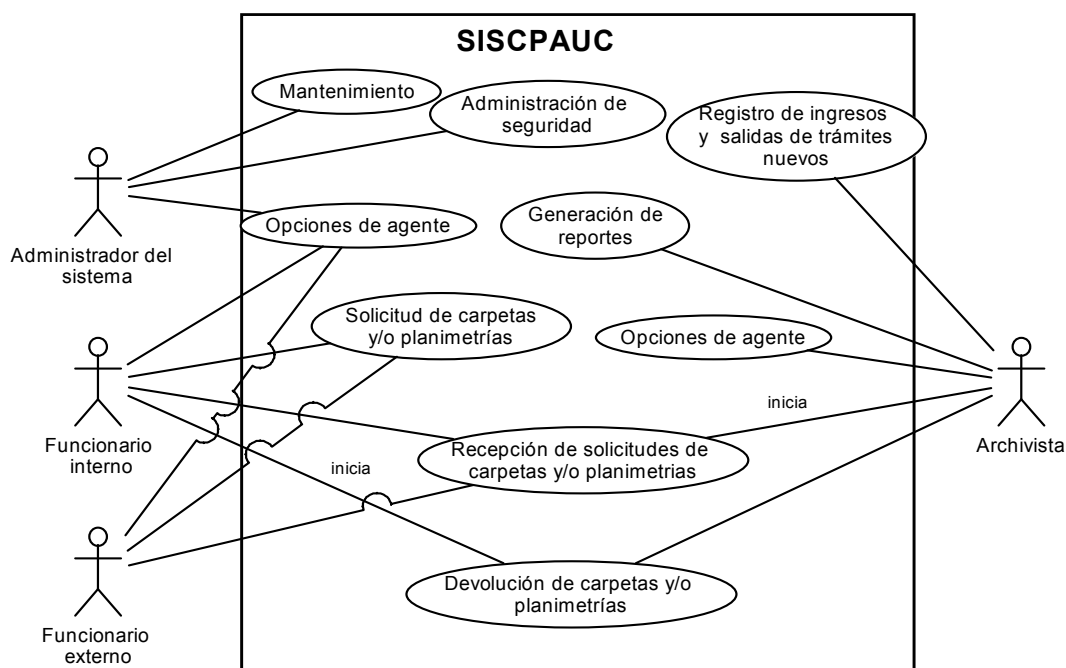
3.3.1 MODELO DE CASOS DE USO

Los procesos del sistema o fragmentos de funcionalidad que reflejarán los requisitos, se describen mediante los casos de uso esenciales del sistema, actores

y la descripción de los mismos.

3.3.1.1 DEFINICION DE DIAGRAMA DE CASOS ESENCIALES

La Figura 3.10 define los casos de uso esenciales del Sistema SISCPAUC correspondiente a la forma en la que los actores usan el sistema.



**Figura 3.10 Casos de uso esenciales del sistema SISCPAUC
[Elaboración propia]**

La identificación de los actores con los que el sistema interactuará se realiza con una *Especificación de actores*, indicando con que casos de usos se relacionan los actores, como se muestra a continuación en las siguientes tablas.

Actor	Administrador de sistema
Casos de uso	Mantenimiento, Administración de seguridad, Registro de ingreso y salidas de trámites nuevos, Solicitud de carpetas y/o planimetrías,

	Recepción de solicitudes de carpetas y/o planimetrías, Devolución de carpetas y/o planimetrías, Reportes de ingresos y salidas de trámites nuevos, Generación de reportes, Opciones de Agente
Tipo	Principal
Descripción	Es el actor principal y representa a cualquier persona que realiza la tarea de mantener y administrar el sistema.

**Tabla 3.2 Especificación de actores: Administrador del sistema
[Elaboración propia]**

Actor	Archivista
Casos de uso	Registro de ingreso y salidas de trámites nuevos, Solicitud de carpetas y/o planimetrías, Recepción de solicitudes de carpetas y/o planimetrías, Devolución de carpetas y/o planimetrías, Reportes de ingresos y salidas de trámites nuevos, Generación de reportes, Opciones de Agente
Tipo	Secundario
Descripción	Es un actor secundario y representa a cualquier persona que tenga el cargo de archivista en la Unidad de catastro.

**Tabla 3.3 Especificación de actores: Archivista
[Elaboración propia]**

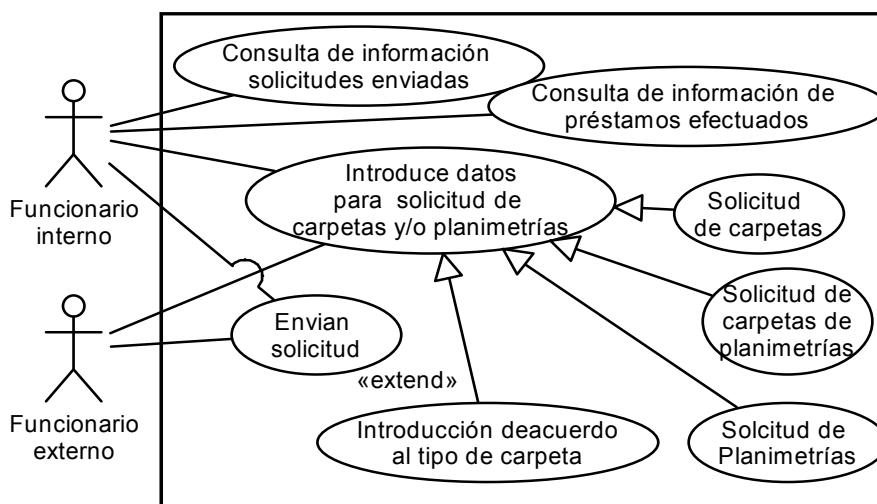
Actor	Funcionario Interno
Casos de uso	Solicitud de carpetas y/o planimetría, Recepción de solicitudes de carpetas y/o planimetrías, Devolución de carpetas y/o planimetrías, Opciones de Agente.
Tipo	Secundario
Descripción	Es un actor secundario y representa a todas las personas que son funcionarios del Gobierno Municipal de La Paz y que pertenecen a la Unidad de Catastro, se puede generalizarlos en: cartógrafo, clasificador y valorizador

**Tabla 3.4 Especificación de actores: Funcionario Interno
[Elaboración propia]**

Actor	Funcionario Externo
Casos de uso	Solicitud de carpetas y/o planimetría, Recepción de solicitudes de carpetas y/o planimetrías, Opciones de Agente
Tipo	Secundario
Descripción	Es un actor secundario y representa a todas las personas que no pertenecen a la unidad de Catastro, pero que si son funcionarios pertenecientes al Gobierno Municipal de La Paz.

**Tabla 3.5 Especificación de actores: Funcionario Externo
[Elaboración propia]**

La Figura 3.11 a la Figura 3.18 son diagramas de casos más internos y específicos a los Casos de uso Esenciales, es decir, se detalla por separado cada caso de uso que se ve en la Figura 3.10, cada caso muestra la funcionalidad del sistema y proporciona una guía para el diseño, implementación y prueba, hasta la conclusión del sistema.



**Figura 3.11 Diagrama de casos de uso: Solicitudes de carpetas y/o planimetrías
[Elaboración propia]**

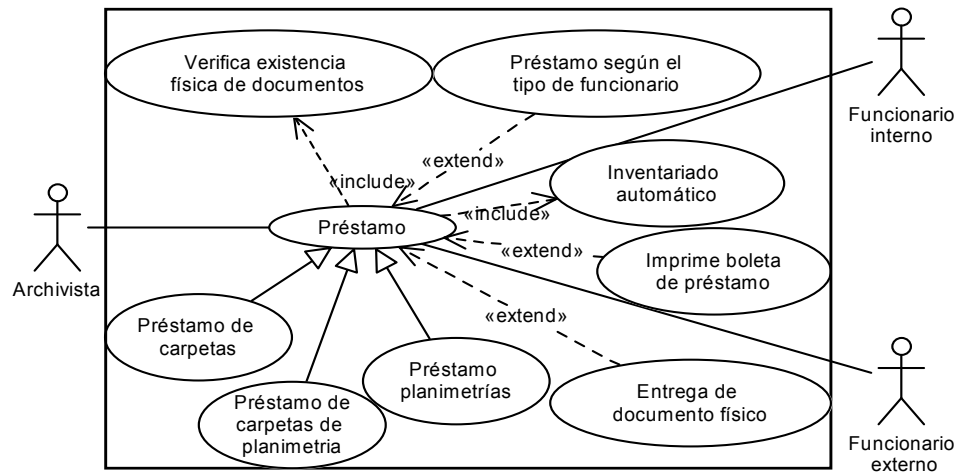


Figura 3.12 Diagrama de casos de uso: Recepción de solicitudes de carpetas y/o Planimetrías [Elaboración propia]

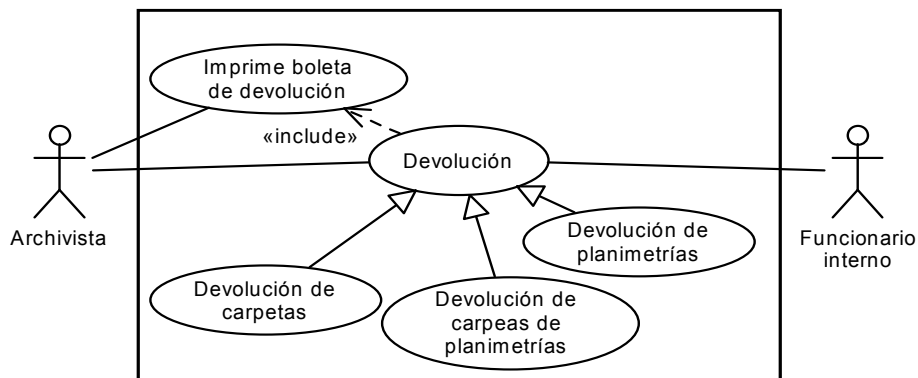


Figura 3.13 Diagrama de casos de uso: Devolución de carpetas y/o planimetrías [Elaboración propia]

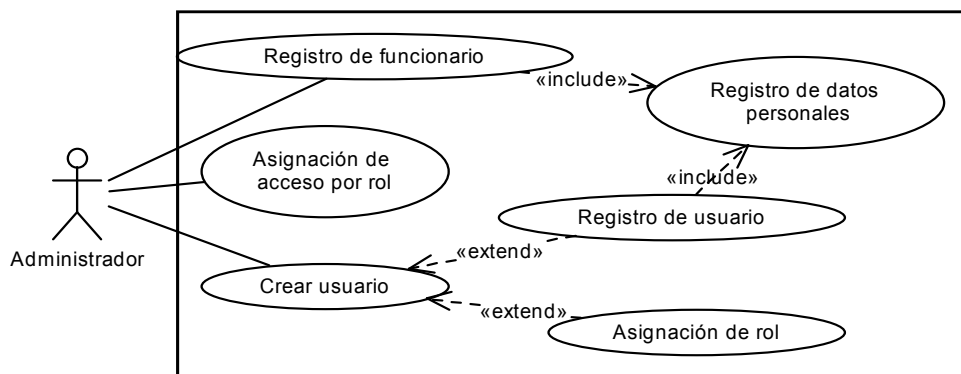


Figura 3.14 Diagrama de casos de uso: Administración de seguridad [Elaboración propia]

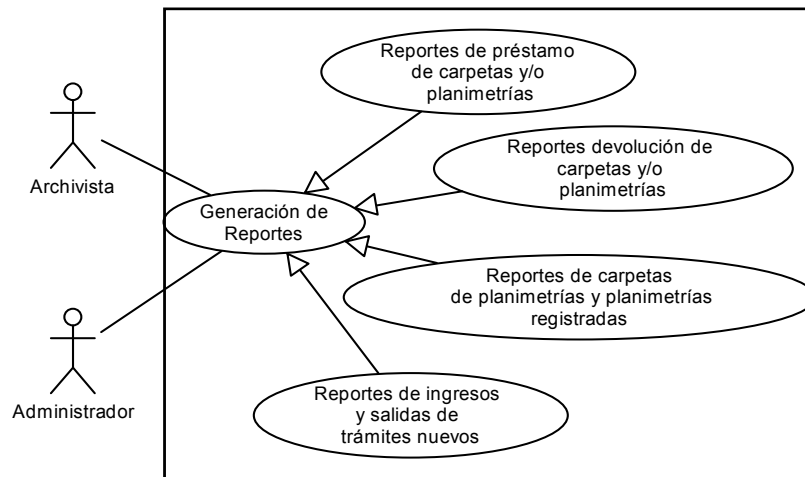


Figura 3.15 Diagrama de casos de uso: Generación de Reportes [Elaboración propia]

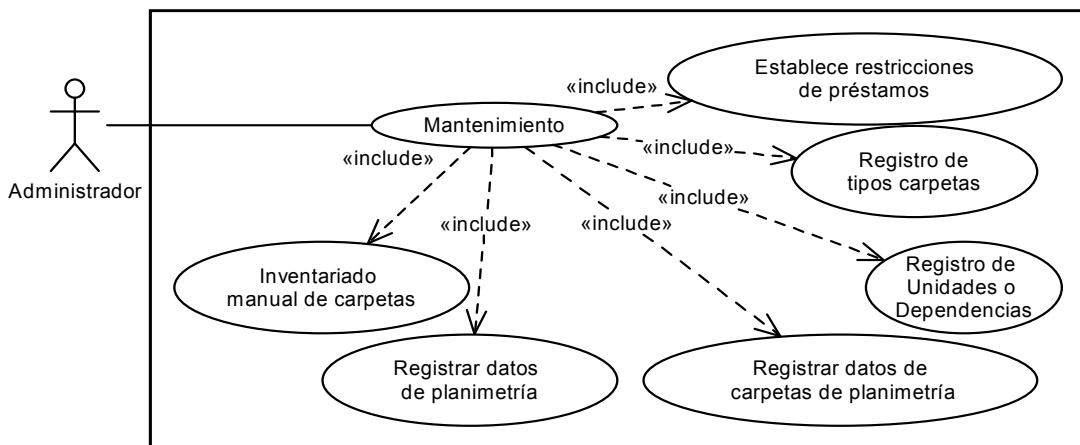


Figura 3.16 Diagrama de casos de uso: Mantenimiento [Elaboración propia]

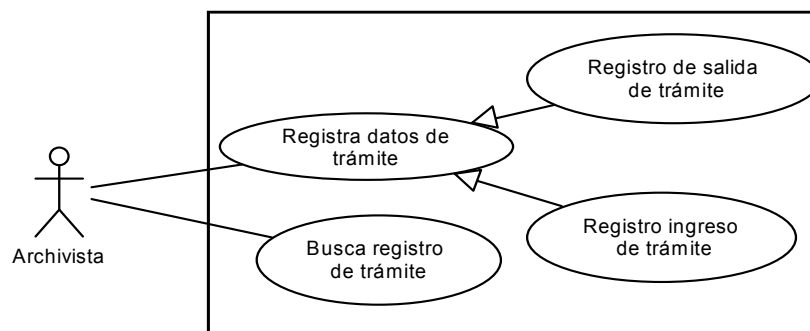


Figura 3.17 Diagrama de casos de uso: Registro de ingresos y salidas de trámites nuevos [Elaboración propia]

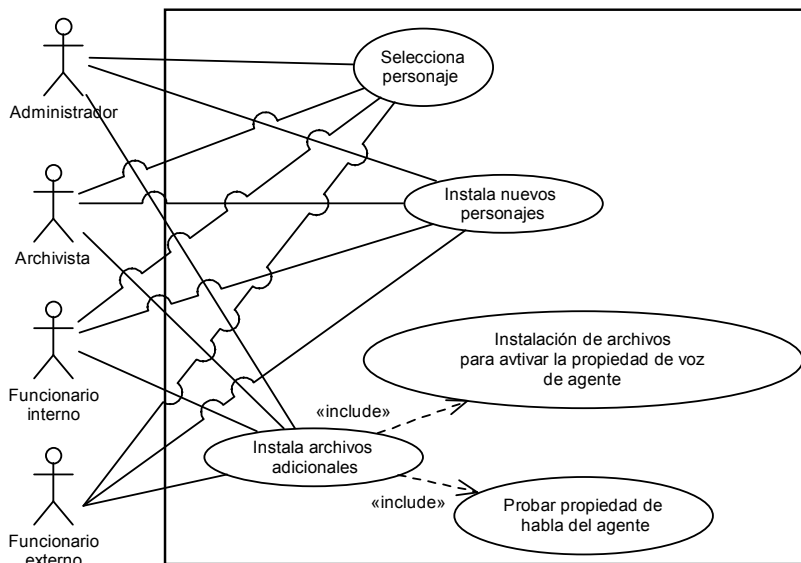


Figura 3.18 Diagrama de casos de uso: Opciones de Agente [Elaboración propia]

3.3.1.2 DESCRIPCIÓN DE CASOS DE USO

Cada uno de los casos de uso de la sección 3.3.1.1 son descritos a partir de la Tabla 3.6 a la Tabla 3.13 en estas se muestran las posibles interacciones de los actores con el sistema para los eventos enviados o recibidos por los actores.

DOCUMENTACION DE DESCRIPCION DE CASOS DE USO	
Nombre	Solicitud de carpetas y/o planimetrías.
Actor	Funcionario interno, Funcionario externo
Descripción	Se inicia cuando el funcionario interno o externo hace una solicitud de préstamo de carpetas y/o planimetrías.
Flujo Principal	Eventos: Funcionario interno o Funcionario externo
	Eventos: Sistema
	1. Selecciona el tipo de solicitud (solo Funcionario interno)
	2. Habilita opciones según elección
	3. Selecciona ventanilla cual será la solicitud de carpetas, carpetas de planimetría o planimetrías.
	4. Genera la interfaz para la tipo de elección
	5. Registra los datos de solicitud.

		6. Graba los datos de solicitud.
		7. Genera solicitud.
	8. Tiene la opción de ver listado de las solicitudes realizadas	
	9. Tiene la opción de ver listado de los préstamos realizados.	
		10. Informa si la solicitud ya fue recepcionada
Precondición	Contar con una cuenta de usuario y haber ingresado al sistema correctamente.	
Poscondición	El envío de la solicitud para su posterior préstamo.	

**Tabla 3.6. Descripción de caso de uso: Solicitud de carpetas y/o planimetrías.
[Elaboración propia]**

La Tabla 3.6 describe las posibles interacciones existentes, cuando los actores Funcionario interno y Funcionario externo realizan ciertos eventos para realizar una solicitud de carpetas y/o planimetrías, haciendo que el sistema realice también sus propios eventos.

DOCUMENTACION DE DESCRIPCION DE CASOS DE USO		
Nombre	Recepción de solicitudes de carpetas y/o planimetrías	
Actor	Archivista, Funcionario interno, Funcionario externo	
Descripción	Este proceso se realizará cuando ha sido aceptada la solicitud del funcionario ya sea interno o externo	
Flujo Principal	Eventos: Archivista, Funcionario interno, Funcionario externo	Eventos: Sistema
		1. Carga en datos de las personas que realizaron su solicitud
	2. El archivista selecciona una opción del tipo préstamo	
		3. Carga y actualiza datos de solicitante.
	4. El archivista selecciona a un solicitante	
		5. Carga solicitudes
	6. El archivista alista carpetas o planimetría solicitadas.	
	7. Envía aviso a funcionario interno o funcionario externo.	
		8. Sistema informa al solicitante
9. El archivista tiene la opción imprimir boleta de préstamo para constancia.		

	10. El archivista acepta cada solicitud.	
		11. Graba los datos de préstamo
		12. Graba los datos de carpetas (Vigentes, MAU, PH, PRACC I, PRAC II, ILIS) prestadas para el inventario.
	13. El archivista entrega físicamente las carpetas o planimetrías a los Funcionarios internos o externos solicitantes.	
Precondición	Existencia de solicitudes que son enviadas por los funcionarios internos y externos	
Poscondición	Registro de los datos del préstamo realizado.	

Tabla 3.7 Descripción de caso de uso: Recepción de solicitudes de carpetas y/o planimetrías. [Elaboración propia]

La Tabla 3.7 también describe las posibles interacciones que existe cuando los actores Funcionario interno, Funcionario externo y Archivista realizan ciertos eventos para realizar la Recepción de solicitudes de carpetas y/o planimetrías, haciendo que el sistema realice también sus propios eventos.

DOCUMENTACION DE DESCRIPCION DE CASOS DE USO		
Nombre	Devolución de carpetas y/o planimetrías.	
Actor	Archivista y funcionario interno.	
Descripción	Este proceso se lleva a cabo cuando un funcionario interno hace una devolución de carpetas y/o planimetrías.	
Flujo Principal	Eventos: Funcionario interno, Archivista	Eventos: Sistema
	1. Archivista selecciona la opción de devoluciones según tipo de carpeta y/o planimetrías.	
		2. Muestra la interfaz de usuario de devoluciones.
	3. Archivista verifica las carpetas que están siendo devueltas.	
	4. Se registra observación si existiera	
	5. Archivista tiene la opción de imprimir boleta de devolución.	
		6. Genera y imprime la boleta de devolución.
	7. Se acepta la devolución.	
	7. El funcionario interno recibe su boleta.	7. Se graban los datos de devolución.
	9. Graba los datos.	

Precondición	Haberse realizado préstamos de carpetas y/o planimetrías.
Poscondición	Registro de los datos de devoluciones.

**Tabla 3.8. Descripción de caso de uso: Devolución de carpetas y/o planimetrías.
[Elaboración propia]**

La Tabla 3.8 describe las posibles interacciones que existen entre los actores Funcionario interno, Archivista y el sistema, bajo ciertos eventos realizados por los actores y el propio sistema, todo esto para realizar la devolución de carpetas y/o planimetría.

DOCUMENTACION DE DESCRIPCION DE CASOS DE USO		
Nombre	Administración de seguridad	
Actor	Administrador	
Descripción	Este proceso se realizará, cuando se efectuó la habilitación de usuarios para el sistema.	
Flujo Principal	Eventos: Administrador	Eventos: Sistema
	1. Selecciona opción de nuevo usuario.	
		2. Carga pantalla de registro de datos personales
	3. Ingresar los datos personales o los modifica	
	4. Ingresar datos de funcionario o los modifica	
	5. Ingresar los datos de cuenta de usuario o los modifica.	
	6. Selecciona el tipo de rol para el usuario.	
	7. Selección la opción grabar	
		8. Se graban los datos
9. Tiene la opción de búsquedas (filtrados de información)		
Precondición	Deberá existir una autorización para registrar a nuevos usuarios o quitar privilegios a los usuarios.	
Poscondición	Los usuarios del sistema tendrán distintos tipos de roles y privilegios para que puedan desenvolverse en el sistema	

**Tabla 3.9 Descripción de caso de uso: Administración de seguridad.
[Elaboración propia]**

La Tabla 3.9 describe las posibles interacciones que existen entre el actor Administrador y sistema, donde ambos crean una comunicación mediante ciertos eventos efectuados, para realizar la administración de la seguridad

DOCUMENTACION DE DESCRIPCION DE CASOS DE USO		
Nombre	Generación de reportes	
Actor	Archivista, Administrador	
Descripción	Este proceso se realizará cuando el archivista o administrador desee obtener el algún tipo de reporte.	
Flujo Principal	Eventos: Archivista, Administrador	Eventos: Sistema
	1. Selecciona que tipo de reporte desea obtener.	
		2. Carga el reporte de acuerdo a la selección.
	2. Tiene la opción de imprimir o no el reporte.	
Precondición	Ninguna	
Poscondición	Los reportes impresos no tendrán validez si cuentan con respaldo de las autoridades competentes.	

**Tabla 3.10 Descripción de caso de uso: Generación de reportes.
[Elaboración propia]**

La Tabla 3.10 describe las posibles interacciones que existen entre los actores Administrador o Archivista y sistema, donde todos ellos crean una comunicación mediante ciertos eventos efectuados, para realizar la generación de algún tipo de reporte.

DOCUMENTACION DE DESCRIPCION DE CASOS DE USO		
Nombre	Mantenimiento	
Actor	Administrador	
Descripción	Este proceso se realizará cuando se desee actualizar datos para el optimo funcionamiento de todo el sistema.	
Flujo Principal	Eventos: Administrador	Eventos: Sistema
	1. Selecciona que información desea actualizar.	
		2. Carga interfaz según la selección, pudiendo ser interfaz de: Carpetas de planimetría, Planimetrías, Tipos de carpetas, Unidades o dependencias, Restricciones de préstamos, Inventariado manual de carpetas (Vigentes, MAU, PH, PRACC I, PRACII ILISS).
		3. Visualiza los datos de carpetas y documentos ingresados al sistema

Precondición	Los datos visualizados son los ya digitalizados y también utilizados en préstamos.
Poscondición	Los datos ya digitalizados y utilizados para el préstamo se graban automáticamente

**Tabla 3.11 Descripción de caso de uso: Mantenimiento.
[Elaboración propia]**

La Tabla 3.11 describe las posibles interacciones que existen entre el actor Administrador y sistema, donde ambos crean una comunicación mediante ciertos eventos efectuados, para realizar el mantenimiento del sistema.

DOCUMENTACION DE DESCRIPCION DE CASOS DE USO																	
Nombre	Registro de ingresos y salidas de trámites nuevos.																
Actor	Archivista																
Descripción	Este proceso se lleva a cabo cuando llegan trámites nuevos a la Unidad de Catastro.																
Flujo Principal	<table border="1"> <thead> <tr> <th>Eventos: Archivista</th> <th>Eventos: Sistema</th> </tr> </thead> <tbody> <tr> <td>1. Introduce datos de salida de trámites</td> <td></td> </tr> <tr> <td>2. Graba los datos introducidos</td> <td></td> </tr> <tr> <td></td> <td>3. Se graban los datos.</td> </tr> <tr> <td>4. Introduce datos de ingresos de trámites.</td> <td></td> </tr> <tr> <td>5. Graba los datos introducidos</td> <td></td> </tr> <tr> <td></td> <td>6. Se graban los datos.</td> </tr> <tr> <td>7. Tiene la opción de búsqueda (filtrado de información)</td> <td></td> </tr> </tbody> </table>	Eventos: Archivista	Eventos: Sistema	1. Introduce datos de salida de trámites		2. Graba los datos introducidos			3. Se graban los datos.	4. Introduce datos de ingresos de trámites.		5. Graba los datos introducidos			6. Se graban los datos.	7. Tiene la opción de búsqueda (filtrado de información)	
Eventos: Archivista	Eventos: Sistema																
1. Introduce datos de salida de trámites																	
2. Graba los datos introducidos																	
	3. Se graban los datos.																
4. Introduce datos de ingresos de trámites.																	
5. Graba los datos introducidos																	
	6. Se graban los datos.																
7. Tiene la opción de búsqueda (filtrado de información)																	
Precondición	Debe existir trámites nuevos para realizar su registro.																
Poscondición	Los trámites nuevos terminan cuando se registra la salida y el ingreso de los mismos.																

Tabla 3.12. Descripción de caso de uso: Registro de ingresos y salidas de trámites nuevos. [Elaboración propia]

La Tabla 3.12 describe las posibles interacciones que existen entre el actor Archivista y sistema, donde ambos crean una comunicación mediante ciertos eventos efectuados, para realizar el registro de trámites nuevos.

DOCUMENTACION DE DESCRIPCION DE CASOS DE USO	
Nombre	Opciones de Agente.

Actor	Administrador, Archivista, Funcionario interno, Funcionario externo	
Descripción	Este proceso se realiza cuando el usuario desea configurar al agente incorporado.	
Flujo Principal	Eventos: Administrador, Archivista, Funcionario interno, Funcionario externo	Eventos: Sistema
	1. Selecciona al de personaje.	
		2. Cambia el agente al personaje seleccionado.
	3. Instala nuevos personajes.	
		4. Carga los datos del nuevo personaje instalado.
	4. Instala archivos para activar la propiedad de habla del agente	
		6. Se activa la propiedad.
	7. Se realiza la prueba de esta propiedad.	
Precondición	Debe existir por lo menos un agente en la maquina del usuario	
Poscondición	Se tiene mas opciones para la elección libre del personaje del agente.	

**Tabla 3.13. Descripción de caso de uso: Opciones de Agente.
[Elaboración propia]**

La Tabla 3.12 describe las posibles interacciones que existen entre los actores Administrador, Funcionario interno, Funcionario externo o Archivista y sistema, donde todos ellos crean una comunicación mediante ciertos eventos efectuados, para realizar el algún tipo de configuración del agente.

3.3.2 MODELO DE ANALISIS

En el modelo de análisis veremos una vista interna del sistema, especificando de forma más precisa a los requisitos. Esto contribuye a obtener una arquitectura estable y sólida para el momento de la implementación; en si las vistas estarán conformadas por diagramas de realización de casos de uso – análisis, siendo estas una colaboración dentro del modelo de análisis que describe como se lleva a cabo y se ejecuta un caso de uso determinado en términos de las clases de análisis y sus objetos de análisis de interacción; donde una clase de análisis es una abstracción de una o varias clases y/o subsistemas del diseño del

sistema; a continuación a partir de la Figura 3.19 a la Figura 3.26 se puede apreciar los respectivos diagramas para cada caso de uso identificado en el modelo de casos de uso de la sección 3.3.1

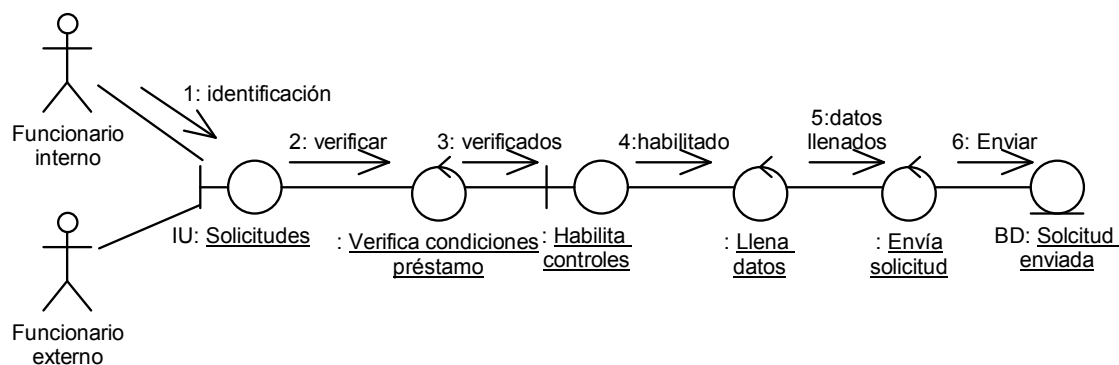


Figura 3.19 Realización de casos de uso - análisis: Solicitud de carpetas y/o planimetrías. [Elaboración propia]

La Figura 3.19 representa un flujo de sucesos para el caso de uso solicitud de carpetas y/o planimetrías, cuando un Funcionario Interno o Funcionario externo realiza una solicitud primero debe ser identificado como usuario, luego en la interfaz respectiva se verifica las condiciones de préstamo, de acuerdo a esto se habilita los controles de la interfaz, posteriormente si no quebranta alguna de las condiciones puede llenar los datos de carpetas y/o planimetrías, se envía la solicitud y esta es grabada en la base de datos.

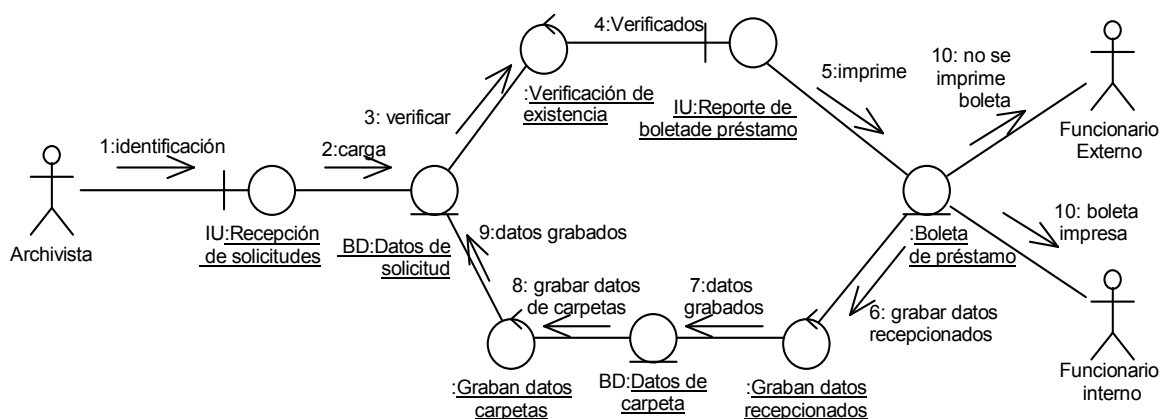


Figura 3.20 Realización de casos de uso - análisis: Recepción de solicitudes de carpetas y/o planimetrías. [Elaboración propia]

La Figura 3.20 representa el siguiente flujo de sucesos para la recepción de solicitudes de carpetas y/o planimetrías: el archivista debe ser identificado como usuario del sistema, se carga la interfaz respectiva, se leen los datos de solicitud conectándose con la base de datos, luego se verifica existencia (si esta o no prestado), seguidamente se visualiza boleta de préstamo y se imprime, posteriormente se graba los datos recepcionados, se graba los datos de carpetas (este ciclo se cumple hasta terminar de recepcionar las solicitudes), después se realiza la entrega de la boleta a los funcionarios internos, para los funcionarios externos no se imprime ninguna boleta.

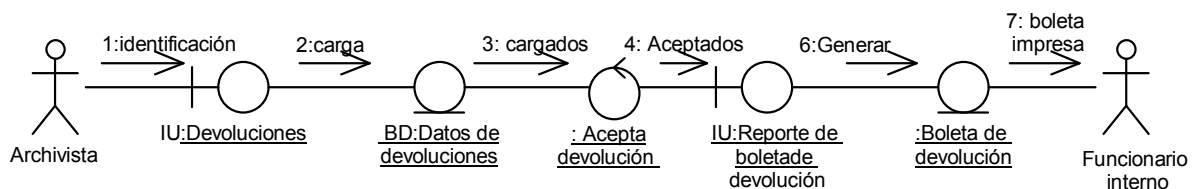


Figura 3.21 Realización de casos de uso - análisis: Devolución de carpetas y/o planimetrías. [Elaboración propia]

La Figura 3.21 muestra el siguiente flujo de sucesos: primero se identifica al archivista como usuario del sistema, se cargan los datos de devolución de la base de datos, posteriormente se acepta la devolución, seguidamente se visualiza la boleta de devolución y se imprime, luego se hace la entrega de la boleta al Funcionario Interno.

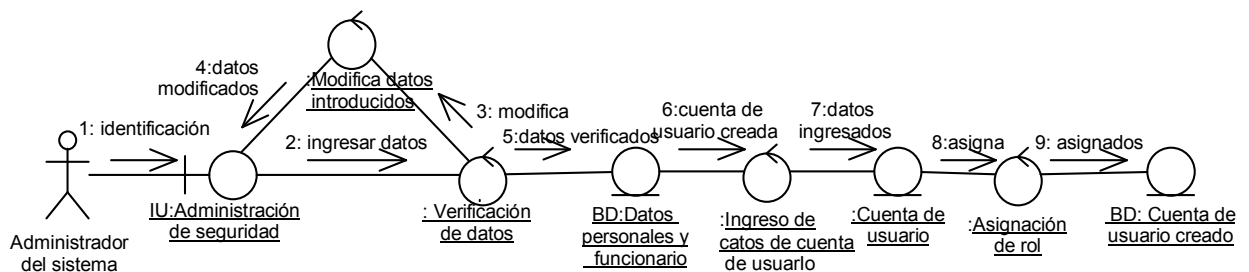


Figura 3.22 Realización de casos de uso - análisis: Administración de seguridad [Elaboración propia]

La Figura 3.22 representa el flujo de sucesos siguiente: se identifica al Administrador como usuario del sistema, se carga la interfaz correspondiente a la administración de seguridad, se ingresan los datos personales y de funcionario de la persona a registrar, en caso de que exista la persona se modifica los datos si no se prosigue al grabado de los datos, luego se ingresa los datos de la cuenta de usuario (login, password y otros) y por último se le asigna un rol a la cuenta de usuario.

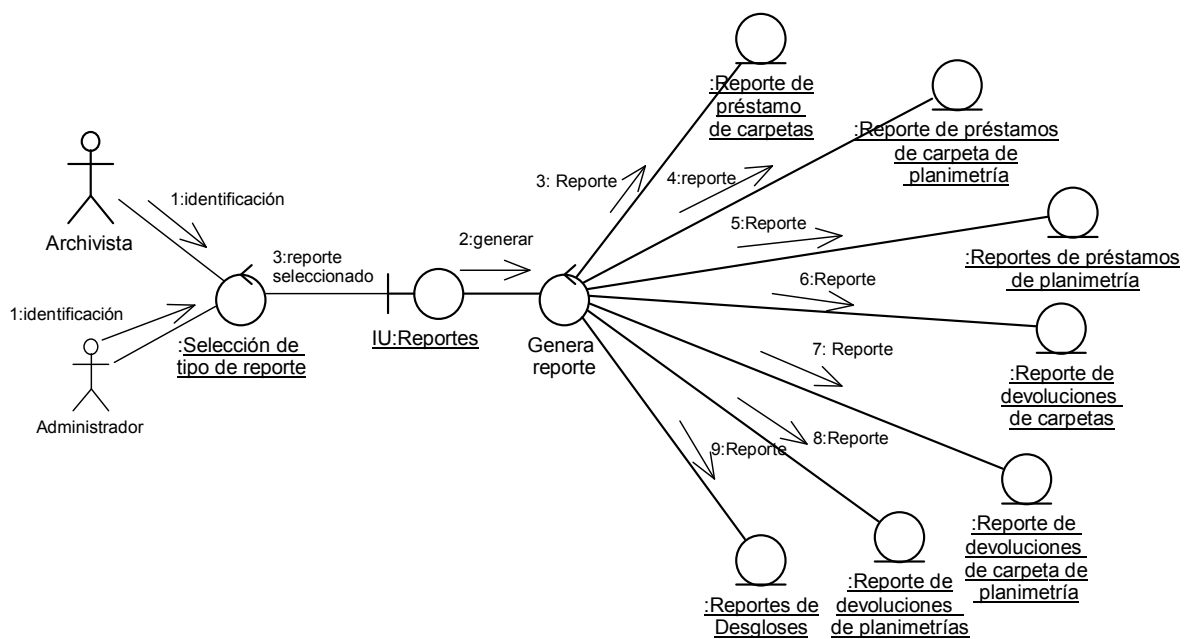


Figura 3.23 Realización de casos de uso - análisis: Generación de reportes. [Elaboración propia]

La Figura 3.23 muestra el siguiente flujo de sucesos: primero se identifica al Archivista o Administrador como usuarios del sistema, estos seleccionan algún tipo de reporte, luego se visualiza la interfaz del reporte, luego se imprime o se genera el reporte.

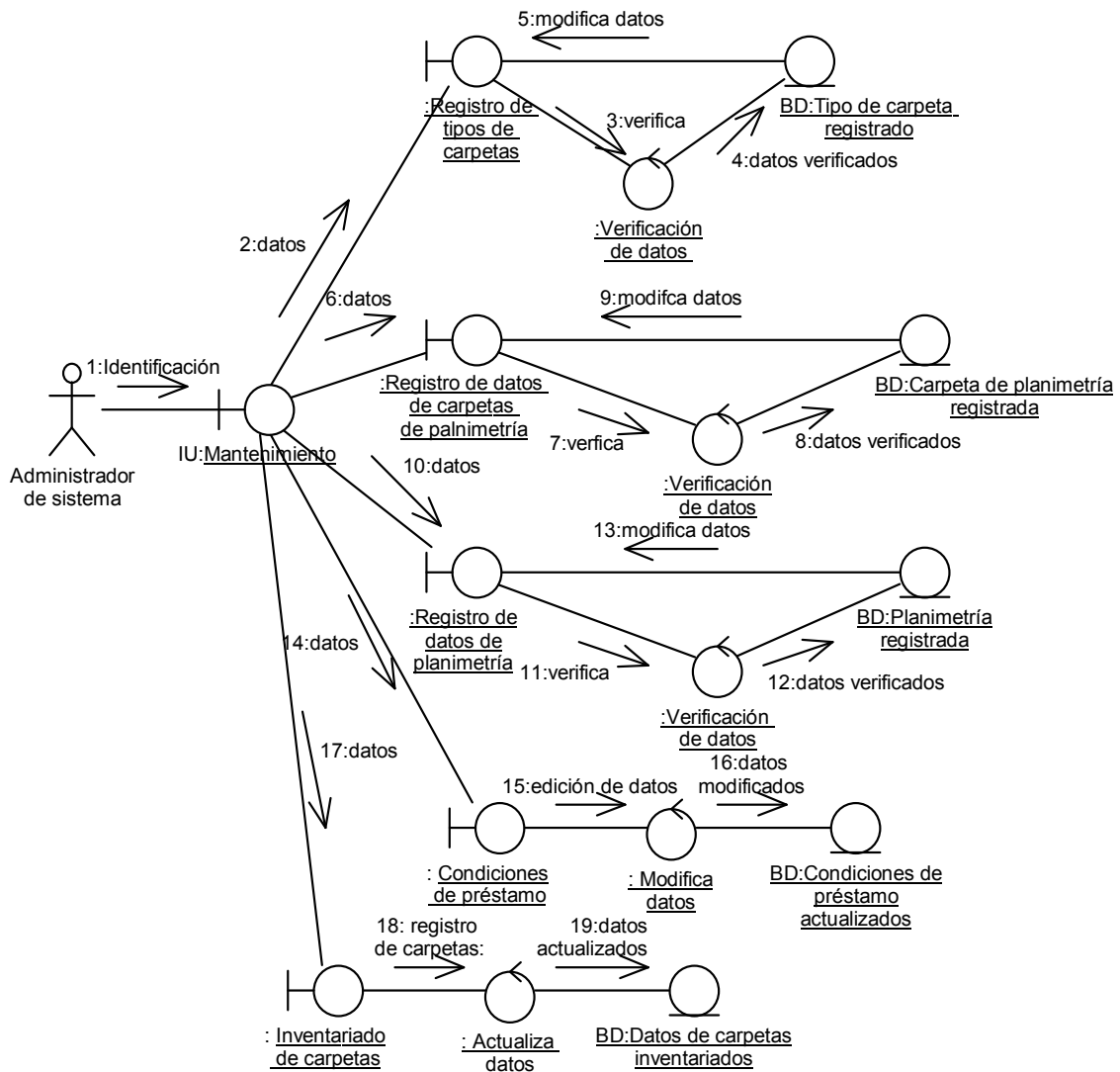


Figura 3.24 Realización de casos de uso: Mantenimiento [Elaboración propia]

La Figura 3.24 describe el siguiente flujo de procesos: se identifica al usuario Administrador, luego selecciona el tipo una opción, luego se carga la interfaz correspondiente; existen cinco opciones que son el registro de tipos de carpetas, registro de datos de carpetas de planimetría, registro de planimetrías, condiciones de préstamo y el inventariado de carpetas todas estas opciones son actualizados y grabados en la base de datos.

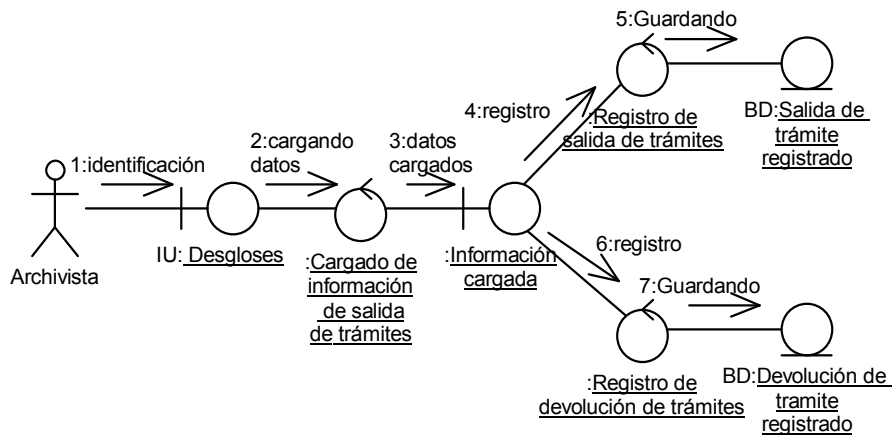


Figura 3.25 Realización de casos de uso - análisis: Registro de ingreso y salidas de trámites nuevos

La figura 3.25 muestra el siguiente flujo de sucesos: se identifica al Archivista, se abre la interfaz correspondiente, luego se cargan datos de salidas de trámites, posteriormente se puede hacer el registro de salidas trámites o la devolución de trámites, estos registros son grabados en la base de datos.

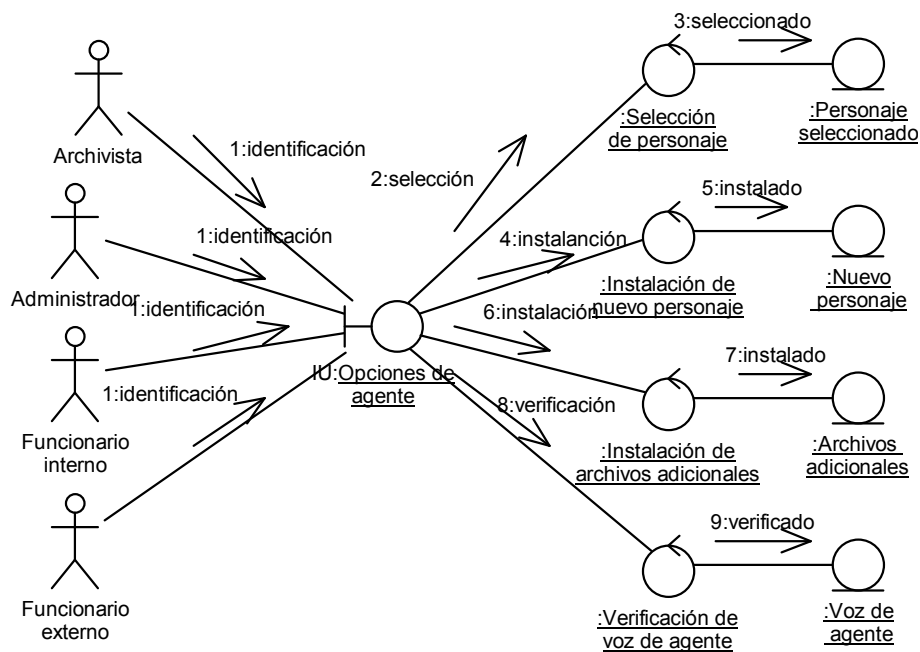


Figura 3.26 Realización de casos de uso - análisis: Opciones de agente [Elaboración propia]

La Figura 3.26 describe el siguiente flujo de sucesos: se identifica al usuario, se carga la interfaz de opciones de agente, se selecciona alguna opción que podría ser la selección de personaje, instalación de nuevo personaje, instalación de archivos adicionales (esta opción permite instalar archivos que permitan brindar la propiedad de habla del agente) o verificación de voz de agente, cualquiera de estas opciones activa alguna propiedad del agente.

3.3.3 MODELO DE DISEÑO

El modelo de diseño ayudará a modelar al sistema de manera tal que soporte los requisitos, haciendo que sea un punto de partida apropiado para las actividades de implementación.

El modelo de diseño se crea tomando como entrada al modelo de análisis, pero este se adapta al entorno de implementación y para eso se realizarán las trazas de diagramas de secuencia y diagramas de estados como se verá seguidamente en las secciones consecuentes.

3.3.3.1 DEFINICIÓN DE DIAGRAMAS DE SECUENCIA

Los diagramas de secuencia ayudarán a plasmar la forma en que los objetos encontrados en las anteriores secciones se comunican entre sí al transcurrir un determinado tiempo, es decir, los diagramas de secuencias del UML agregan dimensiones de tiempo a las interactividades de los objetos; en las Figura 3.27, Figura 3.28, Figura 3.29, Figura 3.30, Figura 3.31 a la Figura 3.32, Figura 3.33 y Figura 3.34 se muestran los diagramas de secuencias de los casos de uso encontrados en la sección 3.3.1.

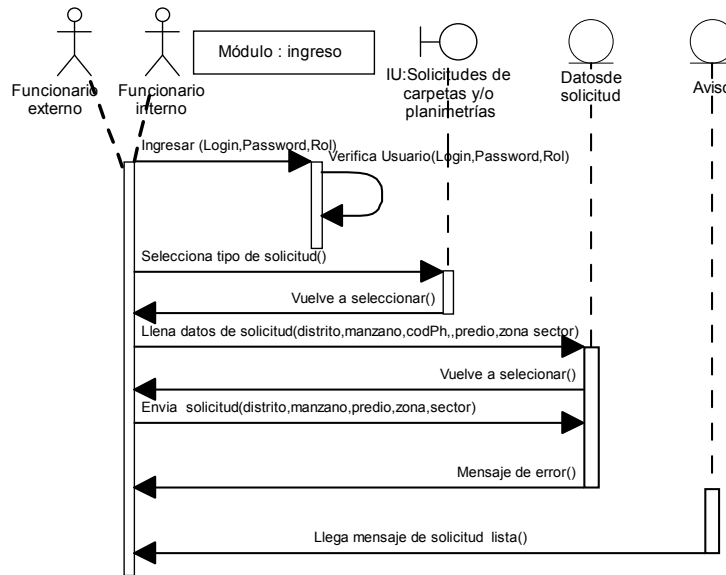


Figura 3.27 Diagrama de secuencia: Solicitudes de carpetas y/o planimetrías. [Elaboración propia]

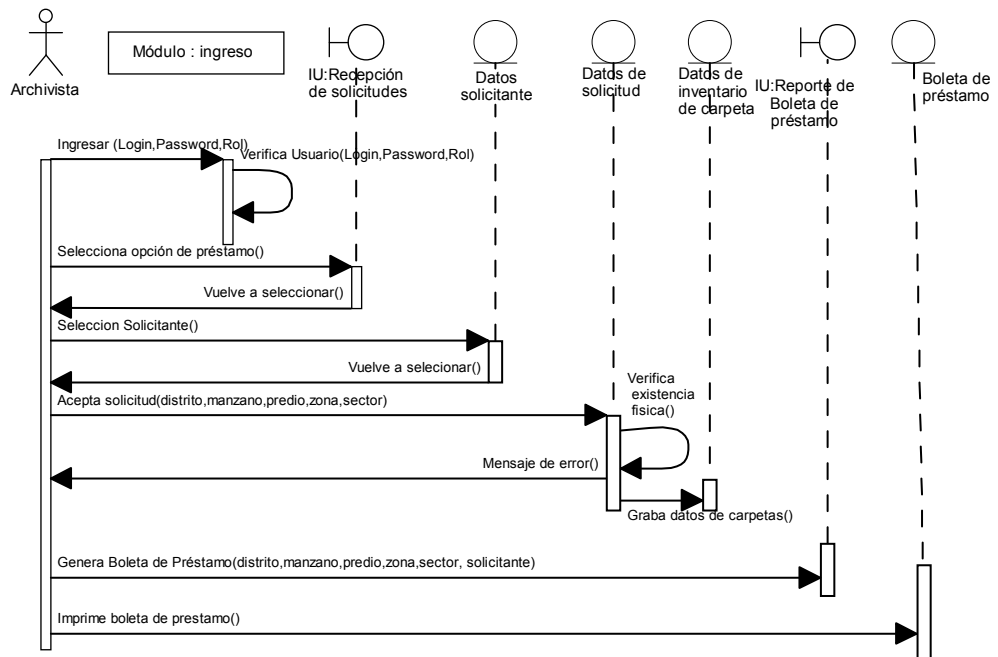


Figura 3.28 Diagrama de secuencia: Recepción de solicitudes de carpetas y/o planimetrías. [Elaboración propia]

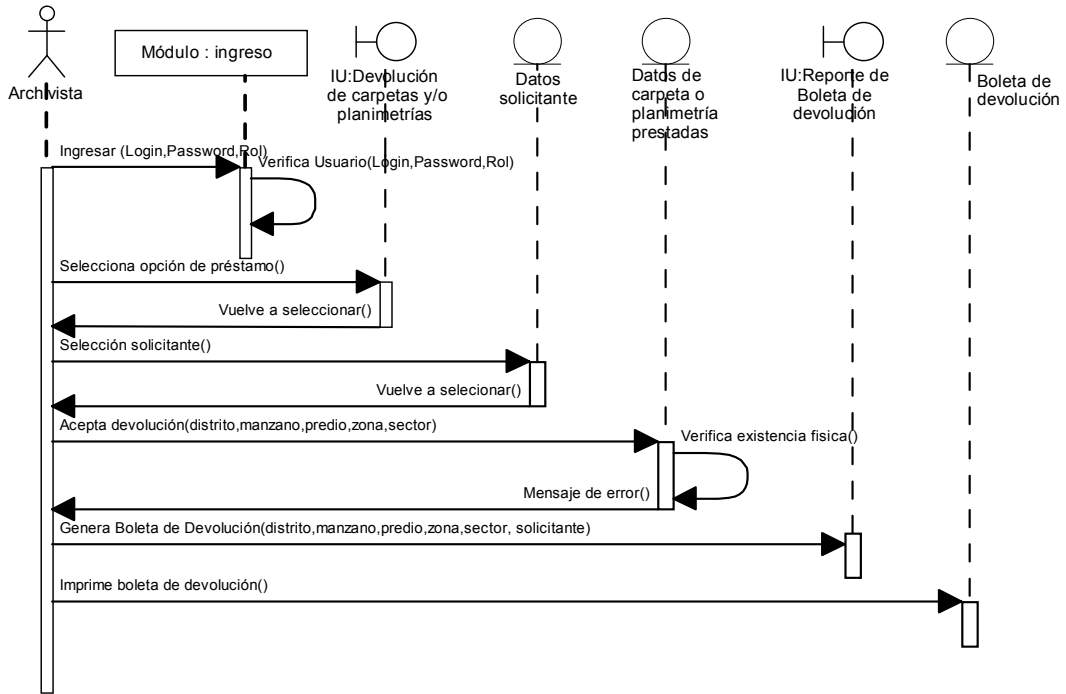


Figura 3.29 Diagrama de secuencia: Devolución de solicitudes de carpetas y/o planimetrías. [Elaboración propia]

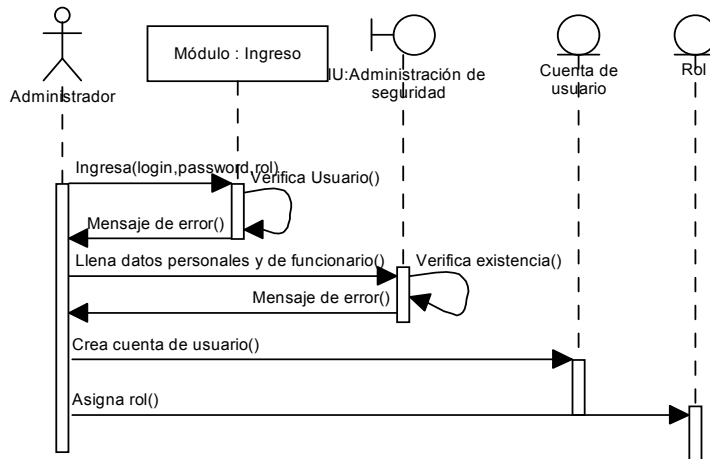


Figura 3.30 Diagrama de secuencia: Administración de seguridad [Elaboración propia]

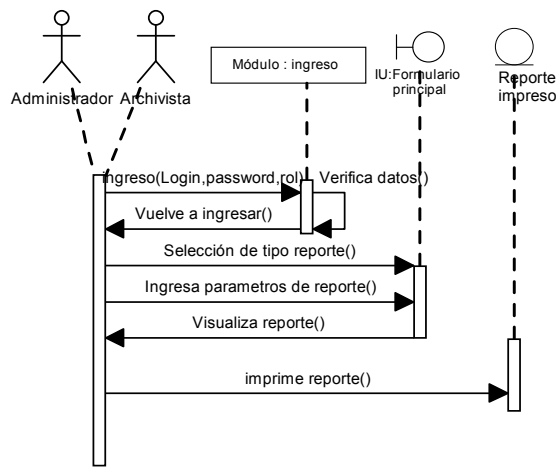


Figura 3.31 Diagrama de secuencia: Generación de reportes [Elaboración propia]

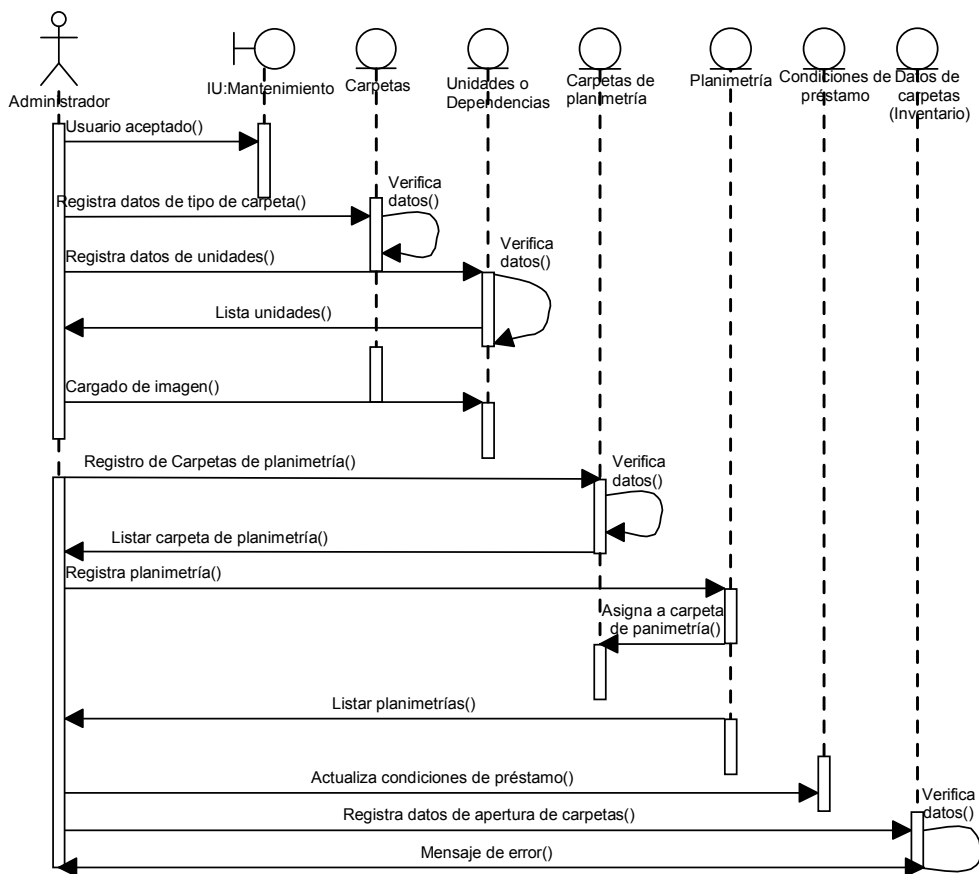


Figura 3.32 Diagrama de secuencia: Mantenimiento [Elaboración propia]

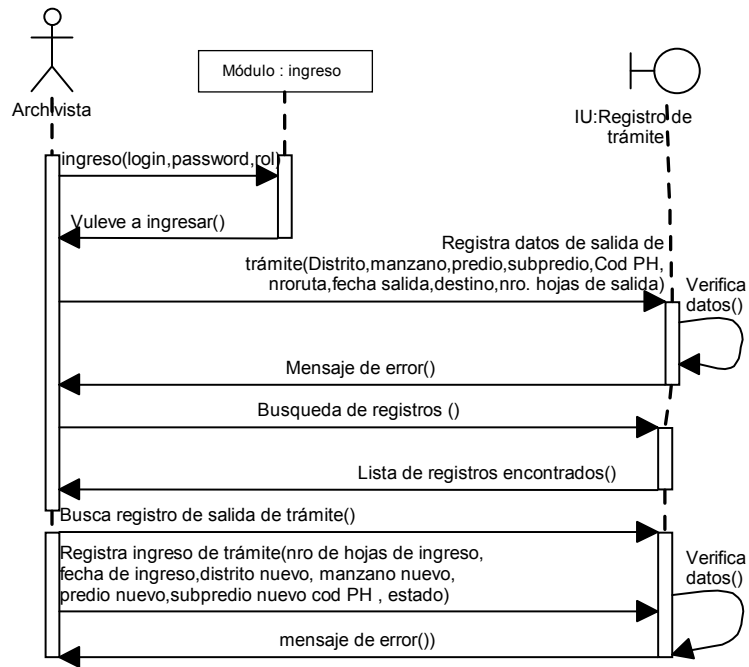


Figura 3.33 Diagrama de secuencia: Registro de ingreso y salidas de trámites nuevos. [Elaboración propia]

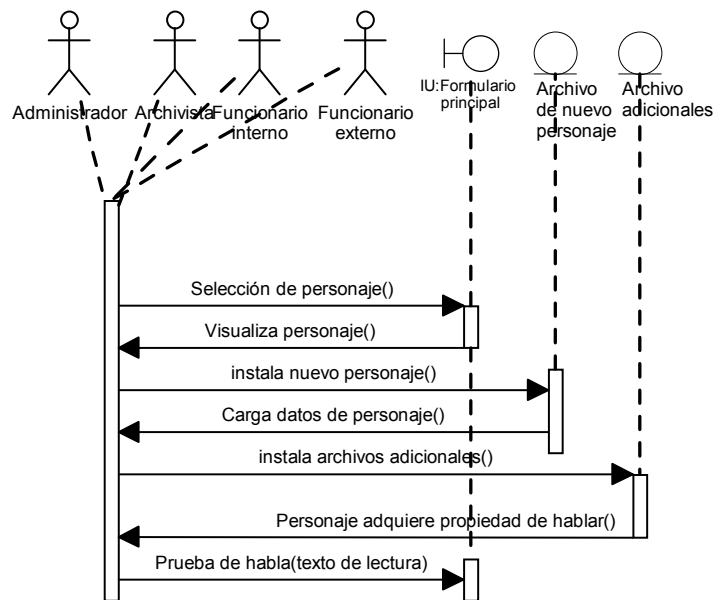


Figura 3.34 Diagrama de secuencia: Opciones de agente [Elaboración propia]

3.3.3.2 DEFINICIÓN DE DIAGRAMAS DE ESTADO

En esta sección se empleará los diagramas de estados para poder caracterizar los cambios para cada caso de uso encontrado en la sección 3.3.1, es decir, se presentará los estados en los que puede encontrarse los objetos junto con las transiciones entre los estados, mostrando los puntos iniciales y finales de una secuencia de cambios de estados; en la Figura 3.35, Figura 3.36, Figura 3.37, Figura 3.38, Figura 3.39 a la Figura 3.40, Figura 3.41 y Figura 3.42 se muestran los diagramas de estados, del sistema SISCPAUC.

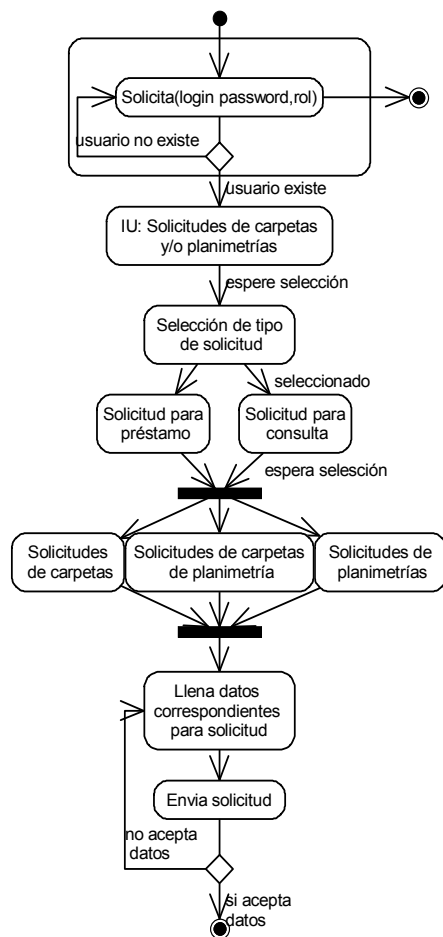


Figura 3.35 Diagrama de estado: Solicitudes de carpetas y/o planimetrías [Elaboración propia]

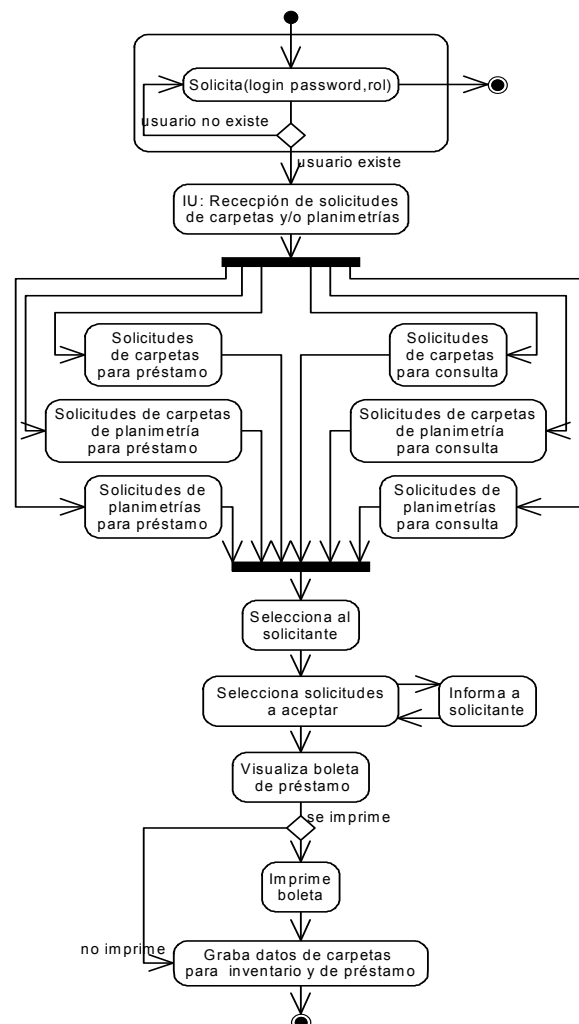
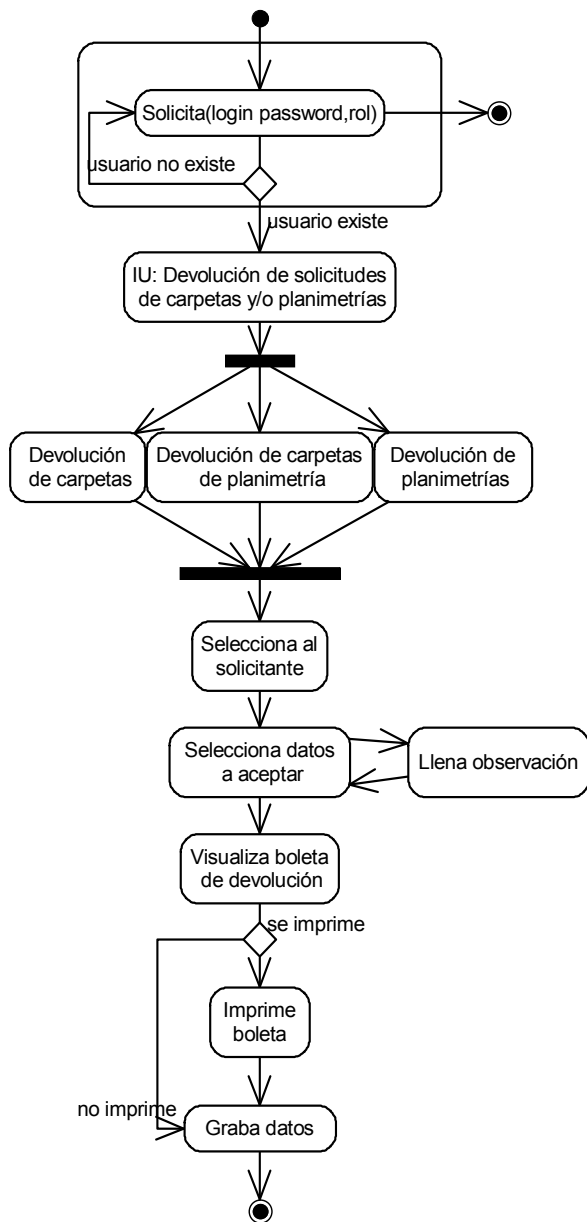
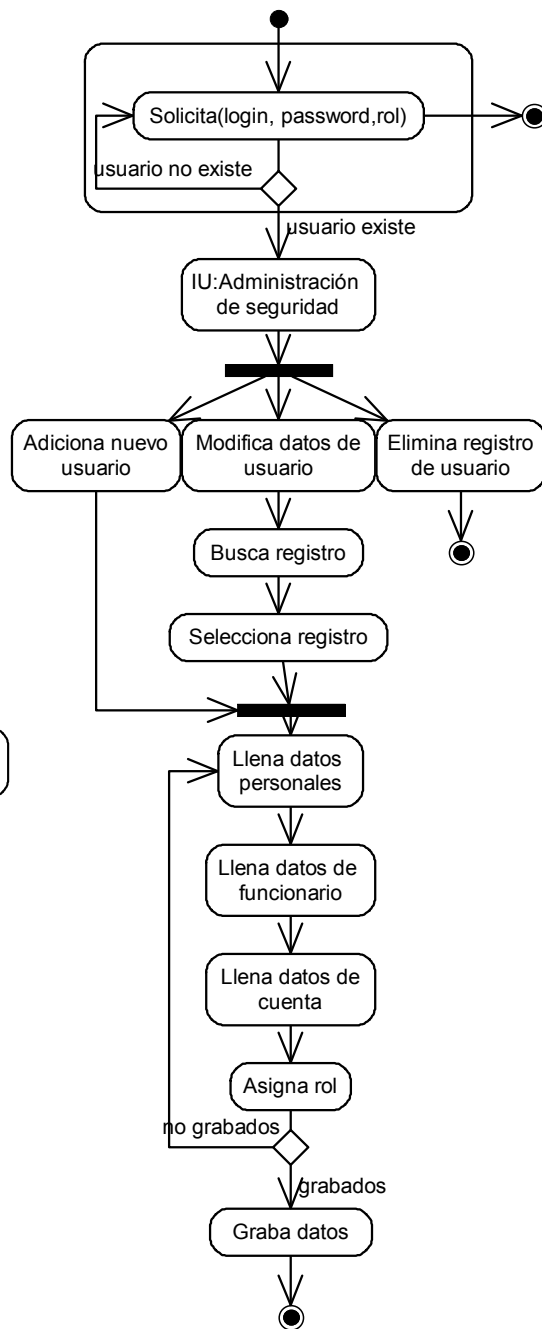


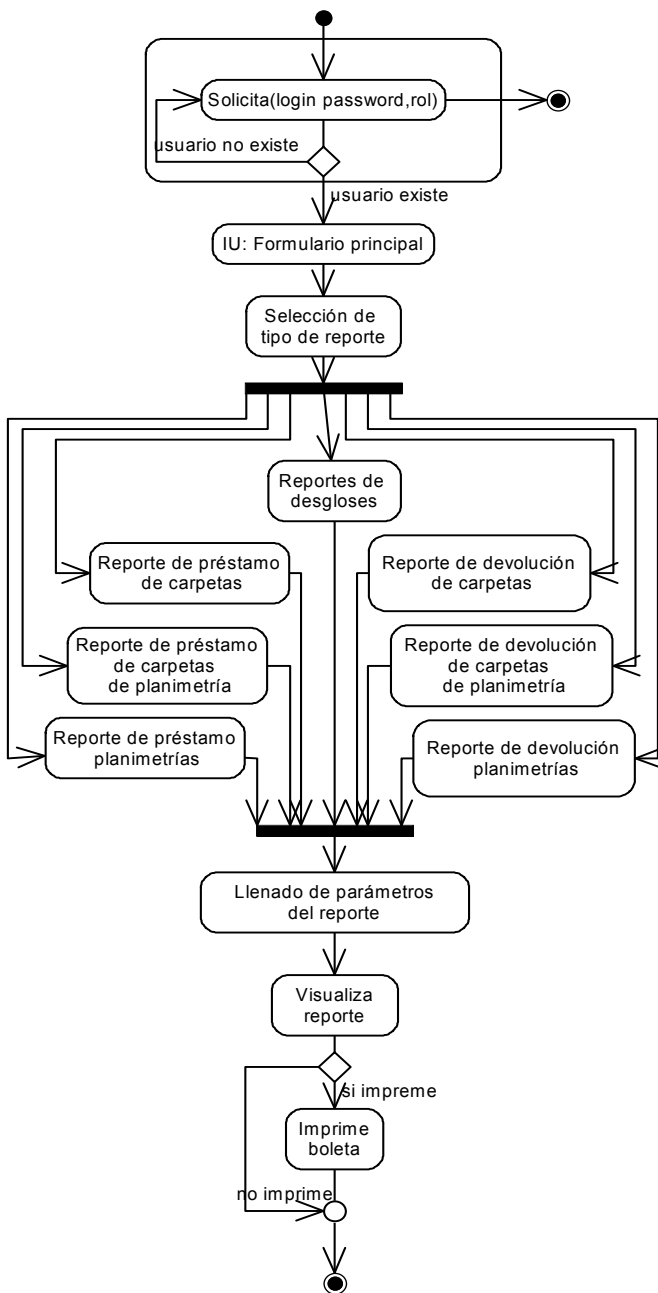
Figura 3.36 Diagrama de estado: Recepción de solicitudes de carpetas y/o planimetrías [Elaboración propia]



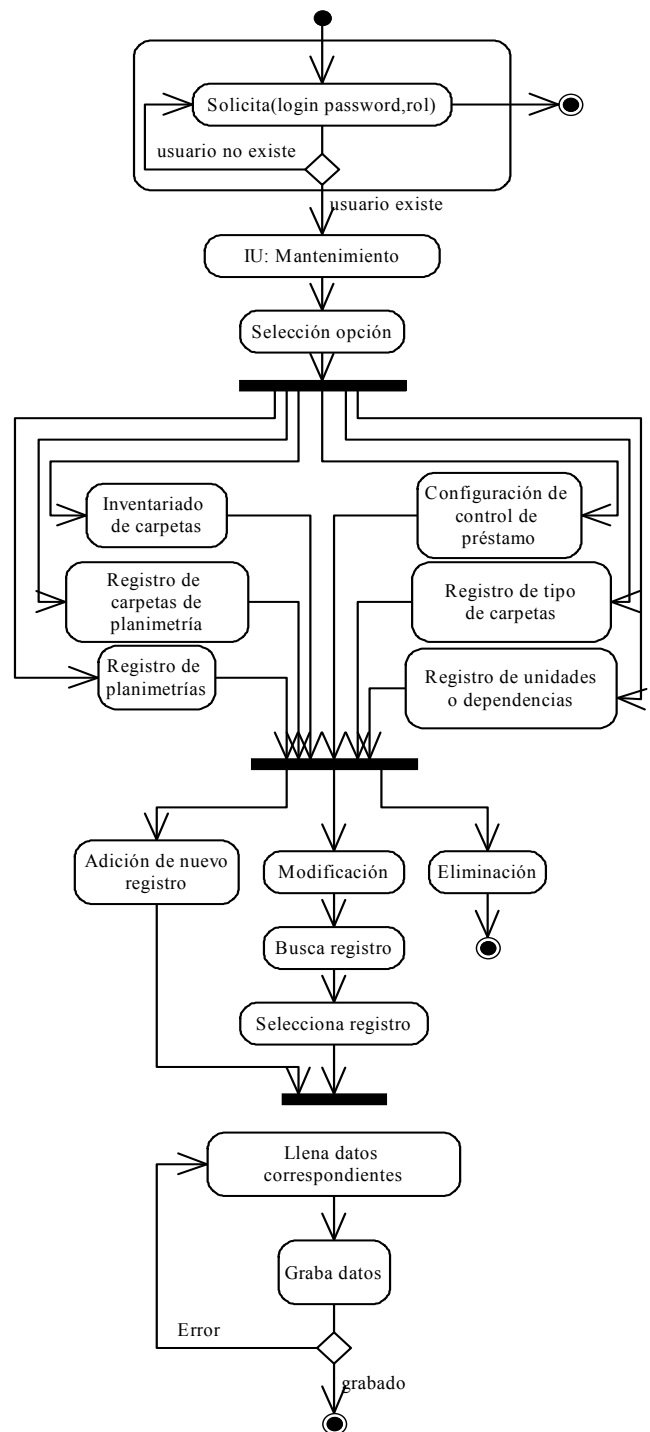
**Figura 3.37 Diagrama de estado:
Devolución de solicitudes
de carpetas y/o planimetrías
[Elaboración propia]**



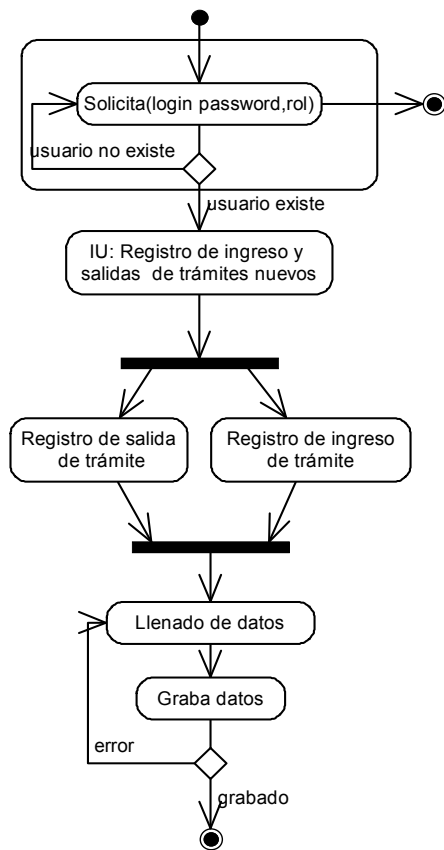
**Figura 3.38 Diagrama de estado:
Administración de seguridad
[Elaboración propia]**



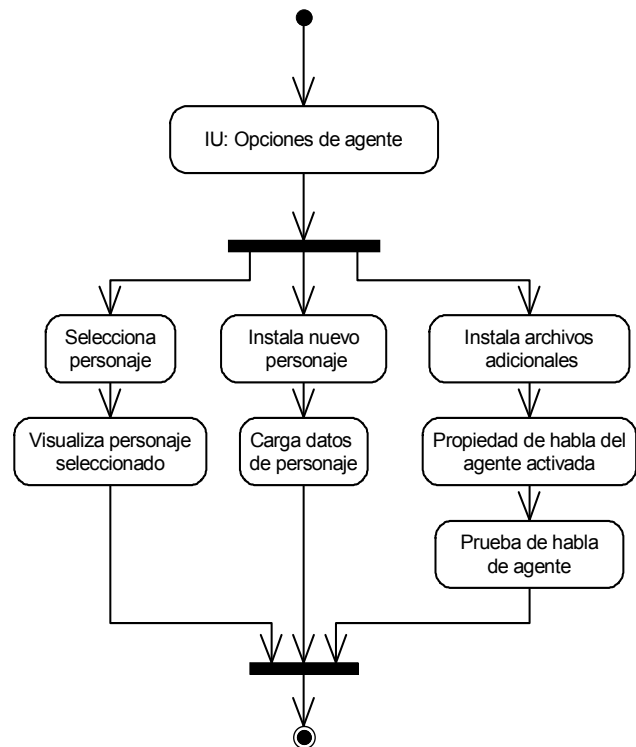
**Figura 3.39 Diagrama de estado:
Generación de reportes
[Elaboración propia]**



**Figura 3.40 Diagrama de estado:
Mantenimiento
[Elaboración propia]**



**Figura 3.41 Diagrama de estado:
Registro de ingreso y salidas de
trámites nuevos
[Elaboración propia]**



**Figura 3.42 Diagrama de estado: Opciones
de agente
[Elaboración propia]**

3.3.3.3 DEFINICIÓN DEL MODELADO DE LA BASE DE DATOS

En la Figura 3.43 se puede observar el modelado de la base de datos del sistema SISCPAUC, este diagrama muestra todas las tablas y sus atributos, también muestra las relaciones que existe entre las tablas con su respectiva cardinalidad.

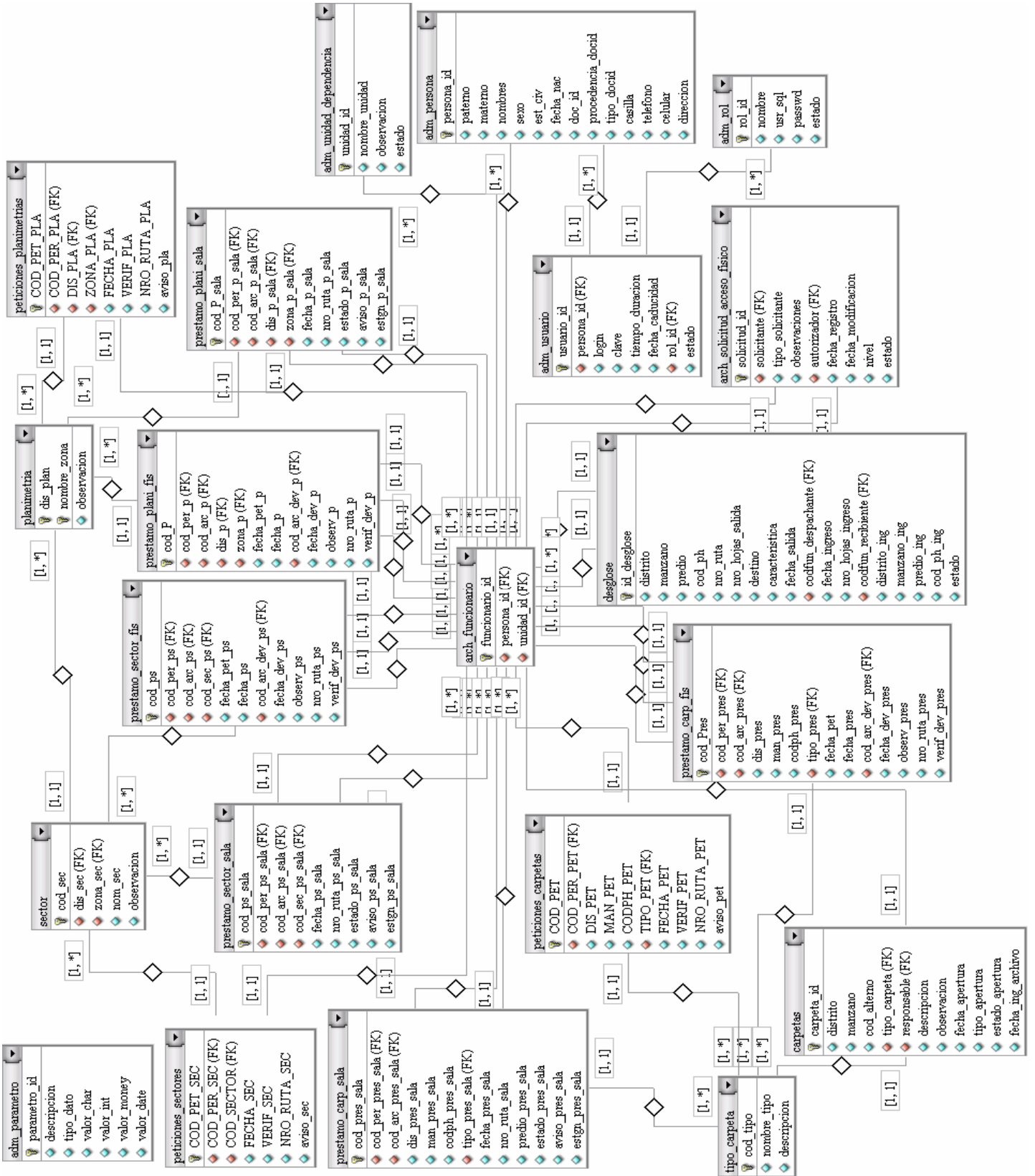


Figura 3.43 Modelado de la base de datos de SISCPAUC [Elaboración propia]

3.3.3.4 DISEÑO DE LA ARQUITECTURA DEL SISTEMA

Con el diseño de la arquitectura del sistema esbozaremos los modelos de diseño y despliegue, pero ahora la tarea fundamental es la identificación de subsistemas y de sus interfaces para las capas de aplicación, el software del sistema y la capa intermedia que constituyen los cimientos del sistema.

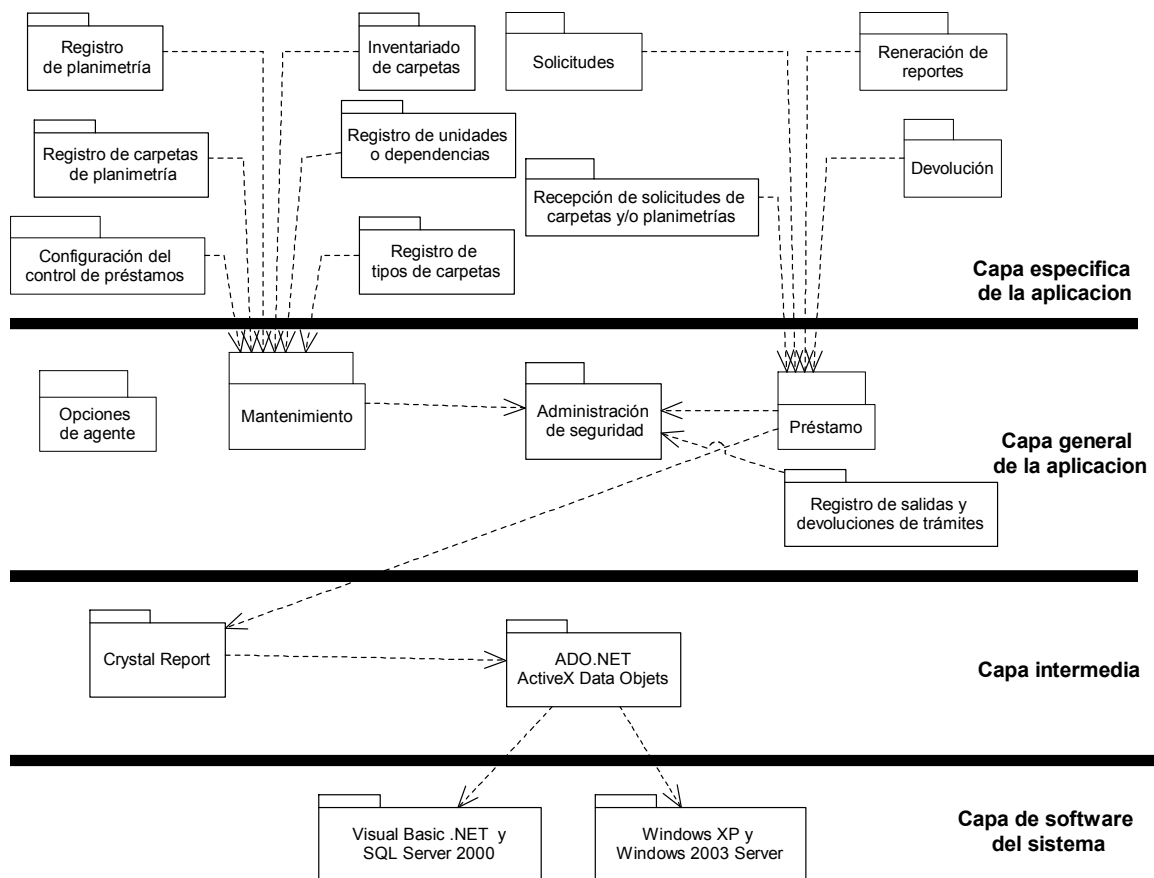


Figura 3.44 Subsistemas y algunas de las dependencias iniciales del sistema. [Elaboración propia]

En la Figura 3.44 se puede observar distintas capas en la Capa específica de la aplicación se muestra todos los subsistemas existentes en el sistema; en la capa general se muestra el empaquetamiento de los subsistemas, es decir, los subsistemas se relacionan con un paquete por la dependencia que

existe entre el subsistema y el paquete, así también puede existir dependencias entre paquetes; en la capa intermedia se encuentran las rutinas u objetos que permiten el acceso a la información; en la capa de software del sistema se encuentra el software que permite la manipulación de la información siendo esta la aplicación elaborada en Visual Basic .Net, SQL Server 2000 y las plataformas para que estas funcionen como ser Windows XP o Windows 2003

3.4 FASE DE CONSTRUCCION

3.4.1 MODELO DE DESPLIEGUE DEL SISTEMA

A continuación se presentan los *diagramas de componentes* y el *diagrama de despliegue* del presente proyecto.

3.4.1.1 DEFINICION DEL DIAGRAMA DE COMPONENTES

El diagrama de componentes de la Figura 3.45 describe como los elementos del modelo de diseño son empaquetados, mostrándonos los componentes de todo el sistema.

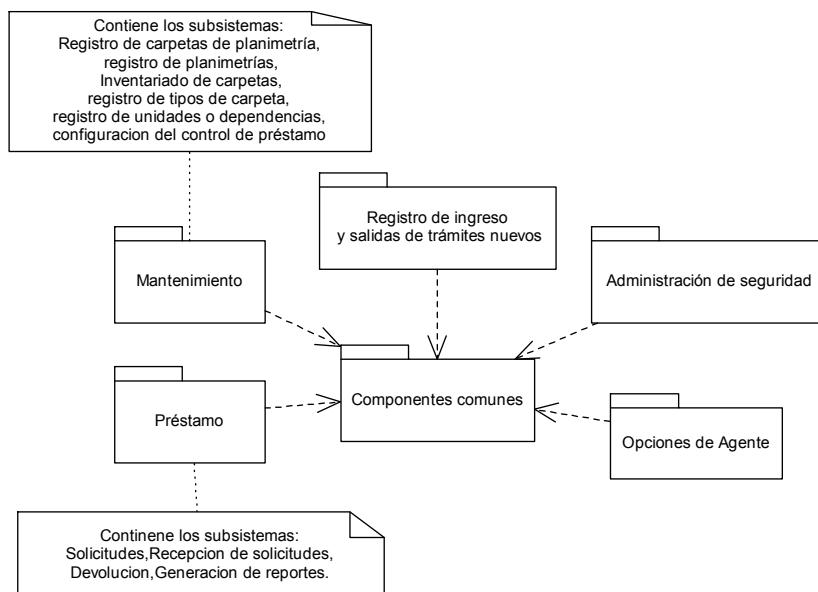


Figura 3.45 Diagrama de componentes del sistema SISCPAUC
[Elaboración propia]

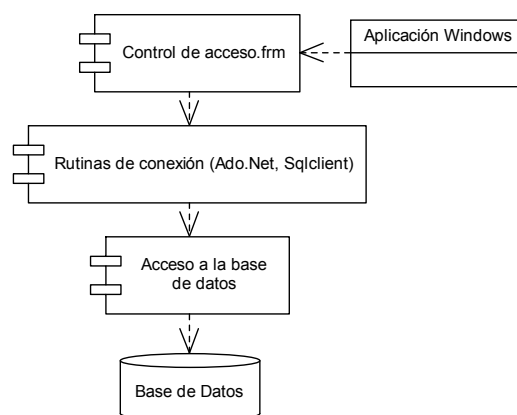


Figura 3.46 Diagrama de componentes: Componentes comunes [Elaboración propia]

El diagrama de componentes comunes de la Figura 3.46, nos muestra los componentes que son utilizados por todos los componentes del sistema para el acceso a la información.

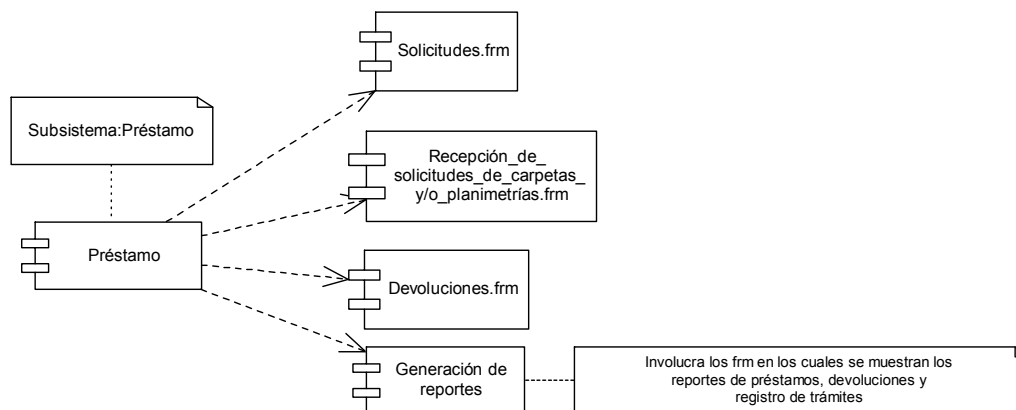


Figura 3.47 Diagrama de componentes: Préstamo [Elaboración propia]

La Figura 3.47 muestra al subsistema Préstamo (módulo de préstamo) con sus respectivos componentes (cada componente lleva la extensión frm ya que en programación Visual Basic .Net una ventana creada es guardada con esa extensión) que son: Solicitudes, Recepción de solicitudes de carpetas y/o

planimetrías, Devolución y la Generación de reportes, todos estos conforman el módulo de préstamo.

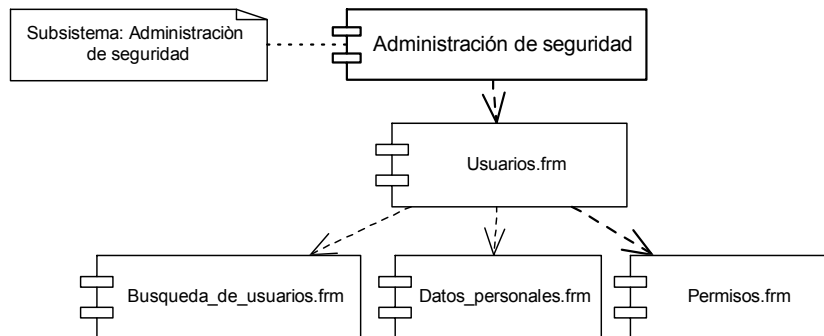


Figura 3.48 Diagrama de componentes: Administración de seguridad. [Elaboración propia]

En el diagrama de componentes de la Figura 3.48 se puede apreciar al subsistema de seguridad con sus respectivos componentes (formularios) que son: usuarios, búsqueda de usuario, datos personales y permisos.

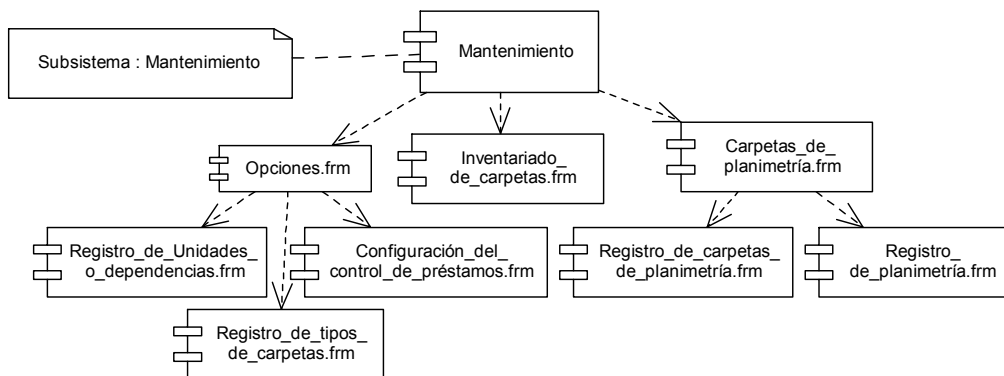


Figura 3.49 Diagrama de componentes: Mantenimiento [Elaboración propia]

El diagrama de componentes de la Figura 3.49 se puede apreciar al subsistema de mantenimiento con sus respectivos componentes (formularios), indicando también las dependencias que existen entre componentes, donde Registro de unidades o dependencias, Registro de tipos de carpets y

Configuración del control de préstamos dependen del componente Opciones; los componentes Registro de carpetas de planimetrías y Registro de planimetrías dependen del componente Carpetas de Planimetría; y por último los componentes Opciones, Inventariado de carpetas y Carpetas de planimetrías dependen del subsistema de Mantenimiento o bien pertenecen a este subsistema.

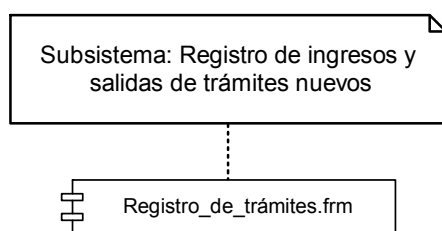


Figura 3.50 Diagrama de componentes: Registro ingresos y salidas nuevos de trámites [Elaboración propia]

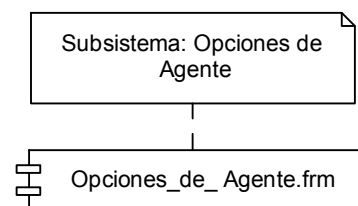
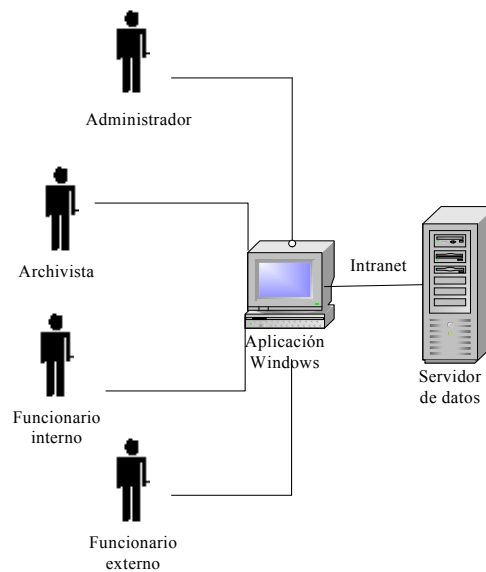


Figura 3.51 Diagrama de componentes: Opciones de agente

Los diagramas mostrados en la Figura 3.50 y Figura 3.51 muestran los subsistemas o módulos de Registro de trámites y Opciones de agente.

3.4.1.2 DEFINICION DEL DIAGRAMA DE DESPLIEGUE

El sistema SISCPAUC se implementó como una aplicación Windows, desarrollándose en la red Intranet con la que cuenta la Unidad de Catastro; esto se representa de manera gráfica en la Figura 3.52.



**Figura 3.52 Diagrama de despliegue del SISCPAUC
[Elaboración propia]**

3.4.2 IMPLEMENTACION

A continuación se muestran las interfaces gráficas principales de la aplicación final de SISCPAUC v1.0. Cabe resaltar que los datos que se presentan son ficticios.

3.4.2.1. INTERFACES DE USUARIO

a) MENU PRINCIPAL DE SISCPAUC

Al inicio del sistema se muestra un splash en el cual se muestra información correspondiente al sistema SISCPAUC, este puede verse en la Figura 3.53.



**Figura 3.53 Splash de inicio de SISCPAUC
[Elaboración propia]**

En la Figura 3.54 se presenta la pantalla inicial principal del sistema SISCPAUC, en cual se muestra el menú general para todos los tipos de usuarios y también se muestra el formulario de autenticación donde el usuario debe ingresar el *nombre de usuario*, la *contraseña* y el *tipo de rol* al cual pertenece; donde el nombre de usuario y la contraseña son únicos para cada usuario, así también se muestra el agente incorporado al sistema.

Si la información introducida en la ventana de autenticación es correcta muestra mensajes de aceptación o error y posterior mente el sistema habilita los menús correspondientes, dependiendo el rol del funcionario esto se muestra en la Figura 3.55.

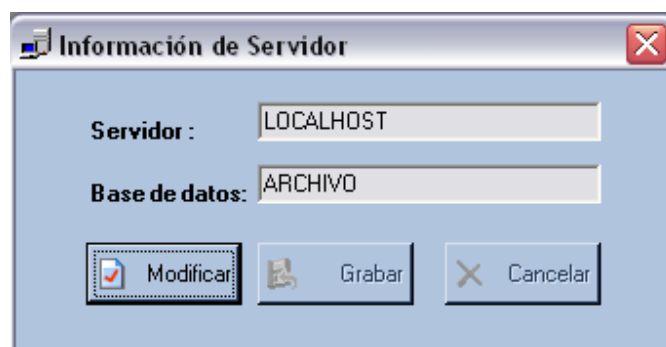


Figura 3.54 Formulario principal SISCPAUC
[Elaboración propia]



**Figura 3.55 Formulario de autenticación
[Elaboración propia]**

En la ventana de autenticación también se muestra algunos datos adicionales los cuales son *Servidor* y *Base de Datos* para mayor flexibilidad del sistema, los cuales pueden ser modificados en otra ventana, esto se muestra en la Figura 3.56.



**Figura 3.56 Formulario de Información del servidor
[Elaboración propia]**

b) ENVIO DE SOLICITUD PARA PRESTAMO O CONSULTA

La Figura 3.57 muestra el formulario para el envío de *Solicitudes*, esta será habilitada para todos los funcionarios internos, y donde se debe llenar los

datos respectivos dependiendo el tipo de préstamo que se desea realizar, estos tipos de préstamo son *Solicitudes para carpetas* (VIGENTE, MAU, PH, ILIS, PRAC I, PRAC II), *Solicitudes para carpetas de planimetría* y *Solicitudes para planimetrías*, en esta ventana también, existen opciones donde el usuario puede realizar un control de los préstamos que realizo, así también de las solicitudes que envió; una de las funciones del agente es informar al usuario que su solicitud ya fue recepcionada en el archivo. Los opciones son mas reducidas para los funcionarios externos, donde estos sólo pueden enviar solicitudes sólo para consultas, mientras que los funcionarios internos pueden realizar solicitudes para préstamos y para consultas.

SISCPAUC - [Solicitudes]

Archivo Procesos Ver

Préstamos

Solicitudes

Cerrar

Listar Carpetas prestadas

Listar Carp. de Plan. prestadas

Listar planimetrías prestadas

Solicitudes de carpetas enviadas

Solicitudes de carp. plan. enviadas

Solicitudes de planimetría enviadas

Usuario: **GUSTAVO SILES**

Fecha: **miércoles, jun 28 2006** Hora: **04:57:41**

La Paz LIDER Gobierno Municipal

Seleccione una de estas dos tareas: Préstamos Consultas

Solicitud carpetas Solicitud Carpetas de Planimetría Solicitud Planimetrías

Datos de carpeta

Nro de ruta: 65481

Tipo de Carpeta: PH

Distrito Manzano Codigo PH

5 2 1725

Enviar Limpiar

GUSTAVO SILES miércoles, jun 28 2006

Figura 3.57 Formulario de Solicitudes
[Elaboración propia]

c) RECEPCIÓN DE SOLICITUDES

En el formulario *Recepción de solicitudes* (véase la Figura 3.58) el archivista recibe todas las solicitudes enviadas por los funcionarios internos y funcionarios externos, siendo estas para carpetas y para planimetrías; lo que el archivista realiza es seleccionar a un funcionario del listado, posteriormente selecciona los registros (que contienen datos de las carpetas o planimetrías a prestar) a continuación informa al funcionario solicitante que ya fue recepcionada su solicitud presionando en el botón informar a usuario, seguidamente si se desea puede imprimirse la boleta de préstamo (véase la figura 3.59) como una constancia del préstamo efectuado.

Entrada de Solicitudes	Cant.
1. GUSTAVO SILES	4
2. DANIELA RAMOS	1
3. OSCAR SANCHEZ	1

Verificación	Distrito	Manzano	Codigo_PH	Tipo	nro_ruta_pet
<input checked="" type="checkbox"/>	12	8	0	VIGENTE	32156
<input type="checkbox"/>	6	1	0	MAU	32154
<input checked="" type="checkbox"/>	6	3	0	VIGENTE	3214
<input type="checkbox"/>	12	6	0	PRAC I	4247

Figura 3.58 Formulario de Recepción de Solicitudes
[Elaboración propia]

The screenshot shows a web application window titled "SISCOAUC - [Devoluciones]". The main heading is "DEVOLUCION DE CARPETAS". Below this, there are tabs for "Devolucion de carpetas", "Devolucion de carpetas de planimetrías", and "Devolucion de Planimetrías". A "Salir" button is in the top right. The main content area is titled "DEVOLUCION DE CARPETAS" and contains a table of devolutions and a form for adding observations.

Funcionarios	Cant.	1. GUSTAVO SILES						
Codigo	Verificacion	Nro_de_Ruta	Distrito	Manzano	Codigo_PH	Tipo		
1	<input checked="" type="checkbox"/>	32156	12	6	0	VIGENTE		
2	<input type="checkbox"/>	32154	6	1	0	MAU		
3	<input checked="" type="checkbox"/>	3214	6	3	0	VIGENTE		
4	<input checked="" type="checkbox"/>	4247	12	6	0	PRAC I		
7	<input type="checkbox"/>	2245	11	5	1234	PH		

Detalle de observacion
Codigo : 4
Observacion:

Buttons: Guardar, Cancelar, Aceptar, Boleta, Observacion

Footer: CARLOS JUANES, miércoles, jun 28 2006

Figura 3.60 Formulario de Devoluciones
[Elaboración propia]

The form is titled "Boleta de Devolucion de Carpetas". It includes the logo of "La Paz LIDER Gobierno Municipal". The responsible person is "CARLOS JUANES". The date of devolution is "28/06/2006 19:10:02". The applicant is "SILES MONRROY GUSTAVO". The form is from the "Oficialía Mayor de Gestión Territorial, Dirección de Información Territorial, Unidad de Catastro".

Nro	Distrito-Manzano	Tipo de Carpeta	Codigo PH	Nro de Ruta	Fecha de Prestamo
1	12 - 6	VIGENTE	0	32156	28/06/2006 18:45:20
2	6 - 1	MAU	0	32154	28/06/2006 18:45:20
3	6 - 3	VIGENTE	0	3214	28/06/2006 18:45:20
4	12 - 6	PRAC I	0	4247	28/06/2006 18:45:19
5	11 - 5	PH	1.234	2245	28/06/2006 18:53:18

Firma de responsable Firma de solicitante

Figura 3.61 Boleta de devolución
[Elaboración propia]

e) REGISTRO DE TRÁMITES

En el formulario de *Registro de Trámites* (véase Figura 3.62) se realiza el proceso de registro de los trámites entrantes a la Unidad de Catastro, en esta el archivista introduce los datos correspondientes a los trámites dependiendo si es un ingreso o salida del mismo, tiene la opción de filtrar la información (es decir realizar una búsqueda avanzada) dependiendo del criterio que elija el usuario, una vez realizado el filtrado existe la opción de imprimir dicho filtrado de información.

REGISTRO DE TRAMITES

Salida de trámites

Distrito	Manzano	Predio	Subpredio	Codigo PH
12	11	1	1	0

Nro ruta : 4756
 Nro fojas de salida : 2
 Destino : 2
 Característica : Copia de Certificado
 Fecha salida : martes, 18 de julio de 2006

Devolución de trámites

Fecha devolución : miércoles, 19 de julio de 2006
 Nro de fojas de devolución : 10
 Nuevo código catastral :

Distrito	Manzano	Predio	Subpredio	Codigo PH
12	11	11	0	0

Observado Finalizado

Desgloses

Codigo	Distrito	Manzano	Predio	SubPred	PH	Nro ruta	Fojas Sa	Destino	Característica	Fecha de sali
1	19	1	31	0	0	3248	1	2	Copia de Certifica	18/07/2006 2
2	12	11	1	0	0	4756	2	2	Copia de Certifica	18/07/2006 2
3	1	22	13	0	0	789	2	2	Registro nuevo	01/07/2006 2
4	11	6	12	0	0	1124	1	2	Registro nuevo	19/07/2006 2
5	4	14	5	0	0	4441	4	S. CANAVIRI	Cambio de nombr	19/07/2006 2
6	5	4	15	0	0	5145	2	2	Cambio de nombr	19/07/2006 2
12	47	31	44	0	45	6548	0	V. SIÑANI	Copia de Certifica	20/07/2006 1
15	41	11	36	0	0	9874	0	V. SIÑANI	Registro nuevo	22/08/2006 1

OSCAR AGUILAR sábado, oct 22 2005

**Figura 3.62 Formulario de Registro de Trámites
[Elaboración propia]**

f) INVENTARIADO DE CARPETAS

En el formulario de *Inventariado de carpetas* (véase Figura 3.63) se muestra los datos correspondientes a las carpetas inventariadas automáticamente por el sistema (mediante el proceso del préstamo de carpetas), así también de las carpetas registradas manualmente por el usuario, actualizar la información ya existente; esta ventana también brinda la opción de filtrar la información bajo cierto criterios de búsqueda que el usuario vea conveniente, así también este filtrado puede ser impreso.

Código	Disco	Módulo	Código	Tipo carpeta	Responsable	Tipo apertura	Estado
15	13	1	668	PH		(null)	A
12	21	14	495	PH	DANIELA RAMOS	(null)	A
31	22	95	0	VIGENTE	MANUEL PEREZ	(null)	A
21	31	4	0	VIGENTE	DANIELA RAMOS	N	A
9	41	12	7132	PH	OSCAR SANCHE	N	A
39	41	21	1114	VIGENTE	GUSTAVO SILES	D	A
26	44	24	524	VIGENTE	CARLOS JUANES	F	A
30	45	14	554	VIGENTE	CARLOS JUANES	N	A

Figura 3.63 Formulario de Inventariado de carpetas
[Elaboración propia]

g) CARPETAS DE PLANIMETRIA

En el formulario de *Carpetas de Planimetría* este ofrece la opción de registrar los datos de carpetas de planimetría (véase Figura 3.64) y planimetrías individuales (véase Figura 3.65), el usuario también tiene la opción de filtrar la información de acuerdo a criterios que elija, pudiendo también imprimir este filtrado, así también permite actualizar la información ya existente.

h) REGISTRO DE USUARIOS

El formulario de *Registro de usuarios* ofrece la capacidad de administrar las cuentas de usuarios del sistema, realizando tareas de adición, eliminación y actualización de la información de datos personales (véase figura 3.66) y los datos de cuentas de usuarios donde también se realiza la asignación de permisos (véase figura 3.67), también para una buena administración de esta información existe la opción del filtrado de información o búsqueda avanzada.

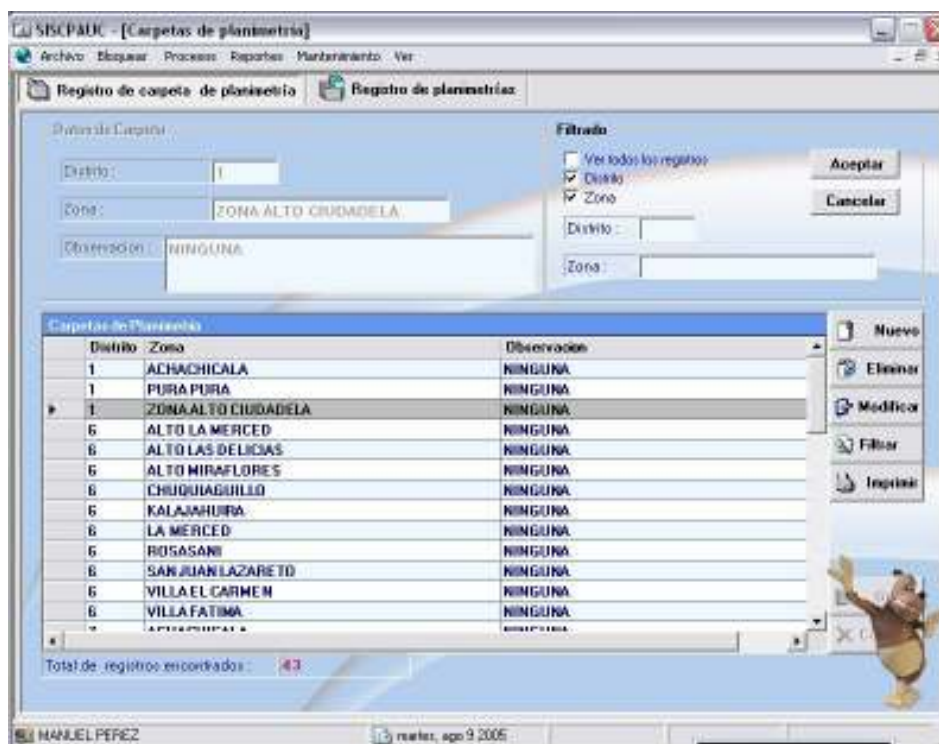


Figura 3.64 Formulario de Carpeta de planimetrías
[Elaboración propia]

The screenshot shows the 'Registro de carpeta de planimetría' window. It features a search section on the right with filters for 'Distrito', 'Zona', and 'Sector'. Below the search is a table of records with columns for 'Distrito', 'Zona', 'Sector y/o urbanización', and 'Observación'. The table lists various urbanization projects in the ACHACHICALA and PURA PURA zones. A status bar at the bottom indicates 'Total de registros encontrados: 141'.

Distrito	Zona	Sector y/o urbanización	Observación
1	ZONA ALTO CIUDADELA	PURA PURA	NINGUNA
1	ACHACHICALA	PLANO GENERAL	NINGUNA
1	ACHACHICALA	SECTOR PLAN AUTOPISTA	NINGUNA
1	ACHACHICALA	TEXTIL FORNO (VINO TINTO)	NINGUNA
1	ACHACHICALA	REMEDIACION 19 DE MAYO	NINGUNA
1	ACHACHICALA	VILLA VINO TINTO ALTO	NINGUNA
1	ACHACHICALA	URB. 18 DE MAYO (VINO TINTO)	NINGUNA
1	PURA PURA	PLANO GENERAL DISTRITOS 1 Y 18	NINGUNA
1	PURA PURA	LOTEAMIENTO ACHACHICALA PROP. SR. SIMON BEDOÑA	NINGUNA
6	ALTO MIRAFLORES	27 DE MAYO (DIVISION Y PARTICION)	NINGUNA
6	ALTO MIRAFLORES	AVENIDA PERIFERICA TRAMO: RIO POQUENI-RIO MINASA	NINGUNA
6	VILLA FATIMA	SEÑOR EXALTACION	NINGUNA

Figura 3.65 Formulario de Planimetrías
[Elaboración propia]

The screenshot shows the 'REGISTRO DE USUARIOS' window. It contains a form for entering personal data, including fields for 'Nombres', 'Apellido Paterno', 'Apellido Materno', 'Sexo', 'Fecha de Nacimiento', 'Estado civil', 'Documento de identidad', 'Procedencia (id)', 'Tipo documento (id)', 'Telefono', 'Celular', 'Correo', 'Direccion', and 'Dependencia'. A blue cartoon character is visible on the right side of the form.

Figura 3.66 Formulario de Registro de usuarios – Datos personales
[Elaboración propia]

Figura 3.67 Formulario de Registro de usuarios – Datos de cuentas y asignación de permisos[Elaboración propia]

3.5 FASE DE TRANSICION

3.5.1. MODELO DE PRUEBAS

La prueba de software es un elemento crítico para la garantía de la calidad de software y representa una revisión final de las especificaciones del diseño y de la codificación [PRESSMAN, 2001].

Cualquier producto puede comprobarse de una de estas formas:

- Prueba de interfaces gráficas de usuario (IGU): Conociendo la función específica para la que fue diseñado el producto, se puede llevar a cabo pruebas que demuestran que el entorno está creado de acuerdo a las necesidades del usuario.

- Prueba de la caja negra: Conociendo la función específica para la que fue diseñado el producto se puede llevar a cabo pruebas que demuestren que cada función es completamente operativa y al mismo tiempo buscando errores en cada función.

3.5.1.1 PRUEBA DE INTERFACES GRAFICAS DE USUARIO

Las interfaces gráficas de usuario (IGU) presentan interesantes desafíos, debido a los componentes reutilizables provistos como partes como parte de los entornos de desarrollo de las IGU, la complejidad de las IGU ha aumentado, originando mas dificultad en el diseño y ejecución de los casos de prueba. Dado que las IGU modernas tienen la misma apariencia y filosofía se puede obtener una serie de pruebas estándar, en la Tabla 3.14 se muestra una serie de cuestiones que servirán como directrices para realizar la prueba, también para dar respuesta a estas preguntas se da una escala de evaluación.

Escala Listado de preguntas	Sin Influencia	Incidencia	Moderado	Medio	Significativo	Esencial
	0	1	2	3	4	5
1. ¿Se abrirán las ventanas basándose en órdenes basadas en el teclado o en un menú?						<input checked="" type="checkbox"/>
2. ¿Se puede ajustar el tamaño, mover y desplegar la ventana?					<input checked="" type="checkbox"/>	
3. ¿Está todo el contenido de la información dentro de la ventana accesible adecuadamente con el ratón, teclas de función, flechas de dirección y teclado?						<input checked="" type="checkbox"/>
4. ¿Se regenera adecuadamente cuando se sobrescribe y se vuelve abrir?		<input checked="" type="checkbox"/>				

5. ¿Están operativas todas las funciones relacionadas con la ventana?					<input checked="" type="checkbox"/>
6. ¿Están disponibles y desplegados apropiadamente en la ventana todos los menús emergentes, barras de herramientas, barras deslizantes, cuadros de diálogo, botones, iconos y controles importantes?					<input checked="" type="checkbox"/>
7. Cuando se despliegan varias ventanas, ¿se representa adecuadamente el nombre de cada ventana?					<input checked="" type="checkbox"/>
8. ¿Esta resaltada adecuadamente la ventana activa?				<input checked="" type="checkbox"/>	
9. Si se utiliza multitarea, ¿están actualizadas todas las ventanas en los momentos adecuados?					<input checked="" type="checkbox"/>
10. ¿Causan las selecciones múltiples o incorrectas del ratón dentro de la ventana efectos secundarios inesperados?	<input checked="" type="checkbox"/>				
11. ¿Están de acuerdo con las especificaciones los indicadores de audio y/o de color de la ventana o como consecuencia de operaciones de la ventana?				<input checked="" type="checkbox"/>	
12. ¿Se cierra adecuadamente la ventana?					<input checked="" type="checkbox"/>
13. ¿Se muestra la barra de menú apropiada en el contexto apropiado?					<input checked="" type="checkbox"/>
14. ¿Despliega la barra de menú de la aplicación características readicionadas con el sistema?		<input checked="" type="checkbox"/>			
15. ¿Funcionan adecuadamente todas las funciones del menú y subfunciones emergentes?					<input checked="" type="checkbox"/>

16. ¿Funcionan adecuadamente las operaciones de despliegue?						<input checked="" type="checkbox"/>
17. ¿Están listadas adecuadamente todas las funciones del menú y subfunciones emergente?						<input checked="" type="checkbox"/>
18. ¿Son todas las funciones del menú accesibles con el ratón?						<input checked="" type="checkbox"/>
19. ¿Es correcto el tipo, tamaño y formato del texto?					<input checked="" type="checkbox"/>	
20. ¿Es posible invocar todas las funciones del menú usando su orden alternativa de texto?		<input checked="" type="checkbox"/>				
21. ¿Están resaltadas todas las funciones del menú (o difuminadas) dependiendo del contexto de las operaciones actuales de la ventana?					<input checked="" type="checkbox"/>	
22. ¿Se ejecutan todas las funciones de cada menú como se anunciaba?						<input checked="" type="checkbox"/>
23. ¿Son suficientemente claros los nombres de las funciones del menú?						<input checked="" type="checkbox"/>
24. ¿Hay ayuda disponible para cada elemento del menú y es sensible al contexto?			<input checked="" type="checkbox"/>			
25. ¿Se reconocen apropiadamente las operaciones del ratón a lo largo de todo el contexto interactivo?						<input checked="" type="checkbox"/>
26. ¿Si se necesitan múltiples clics, están apropiadamente reconocidos en el contexto?						<input checked="" type="checkbox"/>
27. ¿Si el ratón tiene varios botones, son reconocidos apropiadamente en el contexto?						<input checked="" type="checkbox"/>
28. ¿Cambian adecuadamente el cursor, el indicador de procesamiento y puntero al invocar diferentes operaciones?						<input checked="" type="checkbox"/>

29. ¿Se repiten y son introducidos adecuadamente los datos alfanuméricos en el sistema?						<input checked="" type="checkbox"/>
30. ¿Funciona adecuadamente los modos gráficos de entrada de datos?						<input checked="" type="checkbox"/>
31. ¿Se reconocen adecuadamente los datos no válidos?						<input checked="" type="checkbox"/>
32. ¿Son inteligibles los mensajes de entrada de datos?						<input checked="" type="checkbox"/>

**Tabla 3.14. Listado de preguntas y respuestas para las pruebas IGU
[Elaboración propia]**

El valor óptimo máximo se lo puede obtener dando un valor de 5 a todas las respuestas de las preguntas anteriormente mencionadas, este valor encontrado llegaría a ser el 100%, este valor podría ser encontrado de la siguiente manera:

$$\text{Total óptimo} = 32 * 5 = 160$$

Ahora realizando una regla de tres, para encontrar el valor porcentual VP que nos interesa del sistema se tendría:

$$\text{VP} = (100\% / \text{Total óptimo}) * \sum R_i$$

Donde: R_i es el valor dado o asignado a las respuestas a las preguntas realizadas.

Entonces se tendría:

$$\sum R_i = 22*5 + 5*4 + 2*2 + 3*1 = 137$$

$$\text{VP} = (100\% / 160) * 137 = 85\%$$

Por lo tanto como el valor calculado llega al 85%, se podría decir que al realizar la prueba de interfaz gráfica de usuario, se encontró que el sistema es óptimo, tomando en cuenta la siguiente escala:

$75\% \leq \mathbf{VP} \leq 100\% \Rightarrow \textit{Optima}$

$50\% \leq \mathbf{VP} < 75\% \Rightarrow \textit{Buena}$

$25\% \leq \mathbf{VP} < 50\% \Rightarrow \textit{Suficiente}$

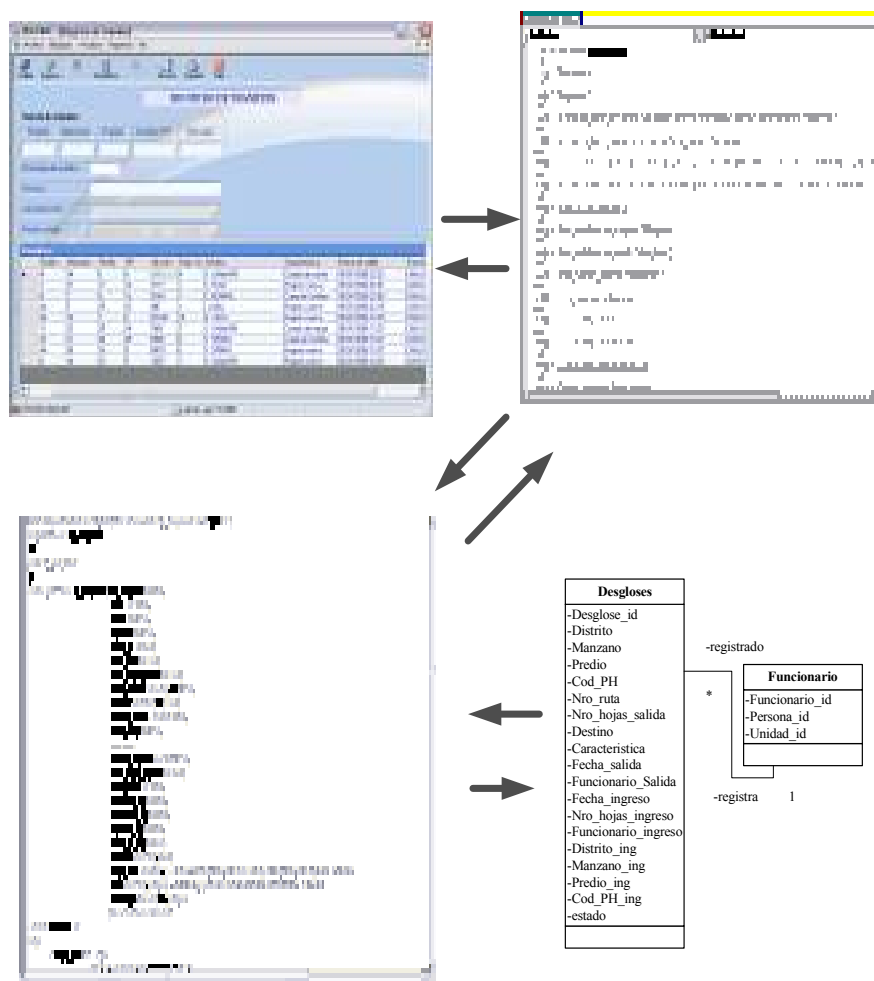
$0\% \leq \mathbf{VP} < 25\% \Rightarrow \textit{Deficiente}$

3.5.1.2. PRUEBA DE CAJA NEGRA

Las pruebas de la caja negra se centran en los requisitos funcionales del software, además que no toman en cuenta la estructura lógica, examinan si las entradas se aceptan de forma adecuada y si se produce un resultado correcto e intenta encontrar errores de los siguientes tipos:

- Errores relativos a las interfaces.
- Errores debidos al rendimiento.
- Errores de inicialización o terminación.
- Errores en estructuras de datos o en accesos a base de datos.
- Funciones incorrectas o inexistentes.

En la Figura 3.77 se muestra las funciones correctas para el registro de trámites nuevos (desgloses), no existe error en la interfaz ya que satisface al usuario en el ingreso de datos del registro de trámites nuevos, no existe error en la estructura de base de datos, pues hace referencia a las tablas correspondientes al mismo.



**Figura 3.68 Prueba de la caja negra para registro de trámites
[Elaboración propia]**

En general todos los módulos del sistema tienen el mismo funcionamiento en el acceso a la información, se procede primero accediendo a la aplicación luego a las clases (propias de la aplicación), posteriormente a los procedimientos almacenados y por último a las tablas de la base de datos como se ve en la Figura 3.68 generando un tipo de resultado, esto ocurre siempre en todo proceso ejecutado por el software por lo tanto se puede concluir que existe un funcionamiento y rendimiento que son correctos.

CAPITULO IV

CALIDAD DE SOFTWARE

4.1. INTRODUCCION

Cuando se habla de calidad de software esto quiere decir que miramos el valor técnico del producto, medido normalmente con expresiones matemáticas que incluyen defectos, las fallas y el tiempo, los mismos que permiten evaluar la calidad del producto en este caso software. En este capítulo desarrollaremos algunas métricas para tener una visión de la calidad del sistema obtenido en el presente proyecto de grado.

4.2. CONFIABILIDAD DEL SOFTWARE

En el sistema se probaron 29000 líneas de código del programa aproximadamente y han ocurrido 25 fallas sobre un total de tiempo de prueba de 280 horas (8 horas * 35 días). La última falla ocurrió en la hora 224. Durante las últimas 26 horas no han ocurrido fallas.

La Tabla 4.1 nos presenta las fallas ocurridas por cada mil líneas de código:

Fallas	Fallas por cada mil líneas de código
9	$9 / 1000 = 0,009$
8	$8 / 1000 = 0,008$
5	$5 / 1000 = 0,005$
3	$3 / 1000 = 0,003$

Tabla 4.1 Fallas por cada mil líneas de código [Elaboración propia]

Basados en la información de la Tabla 4.1 el *promedio máximo de fallas* por cada mil líneas de código es:

$$\frac{0,009 + 0,008 + 0,005 + 0,003}{4} = 0,00625$$

El número proyectado promedio de fallas es:

$$\frac{\text{Promedio máximo de fallas}}{1000} * \text{líneas de código} = \frac{0,00625}{1000} * 29000 = 0.181$$

El número de horas de prueba necesaria para alcanzar la confiabilidad del sistema será:

$$\frac{\text{Ln}[(0,181)/(0,5 + 0,181)] * (224)}{\text{Ln}[(0,5 + 0,181) / (25 + 0,181)]} = 39,086$$

Por lo tanto, se puede alcanzar el nivel deseado de confiabilidad si se prueba adicionalmente 39.086 horas después de la última falla detectada, sin que se hayan producido nuevas fallas. Dado que ya han transcurrido 64 horas de

prueba sin fallas, solo resta probar adicionalmente por 39.086 horas, las mismas que han dado como resultado cero fallas haciendo que el sistema es muy confiable.

4.3. FUNCIONALIDAD DEL SOFTWARE

Los puntos de función son determinados de acuerdo a las cinco características del dominio de la información las que se detallan en la Tabla 4.2.

Parámetro de medición	Cuenta	Factor de ponderación			Total
		Simple	Medio	Complejo	
Número de entradas de usuario	10	3	4	6	30
Número de salidas de usuario	135	4	5	7	540
Número de peticiones de usuario	49	3	4	6	147
Número de archivos	16	7	10	15	112
Número de interfaces externas	1	5	7	10	5
Cuenta total					834

Tabla 4.2 Cálculo de puntos de función

Para calcular los puntos de función se utiliza la siguiente relación.

$$PF = \text{Cuenta Total} * [0.65 + 0.01 * \Sigma Fi]$$

Donde:

- *Cuenta Total*, es la suma de todas las entradas de PF obtenidas de la Tabla 4.2.
- *Fi* puede ser de uno hasta 14 los valores de ajuste de complejidad, basados en las respuestas a las 14 preguntas que aparecen en la Tabla 4.4, evaluando las respuestas en una escala de 0 a 5, tomando en cuenta la Tabla 4.3.

0	1	2	3	4	5
Sin influencia	Incidental	Moderado	Medio	Significativo	Esencial

Tabla 4.3. Escala de evaluación para los factores [PRESSMAN,2001]

Los valores constantes de la ecuación anterior y los factores de peso aplicados en las encuestas de los ámbitos de información han sido determinados empíricamente.

En la Tabla 4.4 se expresa los valores que toman las respuestas dependiendo cual valor se ajusta al sistema y a la explicación que se dio a cada pregunta.

Factores	Sin Influencia	Incidencia	Moderado	Medio	Significativo	Esencial
	0	1	2	3	4	5
1. ¿Requiere el sistema copias de seguridad y de recuperación fiables?						<input checked="" type="checkbox"/>
2. ¿Se requieren comunicaciones de datos?						<input checked="" type="checkbox"/>
3. ¿Existen funciones de procesamiento distribuido?						<input checked="" type="checkbox"/>
4. ¿Es crítico el rendimiento?	<input checked="" type="checkbox"/>					
5. ¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?						<input checked="" type="checkbox"/>
6. ¿Requiere el sistema entrada de datos interactiva?						<input checked="" type="checkbox"/>

7. ¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variadas operaciones?					<input checked="" type="checkbox"/>	
8. ¿Se actualizan los archivos maestros de forma interactiva?			<input checked="" type="checkbox"/>			
9. ¿Son complejas las entradas, las salidas, los archivos o las peticiones?					<input checked="" type="checkbox"/>	
10. ¿Es complejo el procesamiento interno?		<input checked="" type="checkbox"/>				
11. ¿Se ha diseñado el código para ser reutilizable?					<input checked="" type="checkbox"/>	
12. ¿Están incluidas en el diseño la conversión y la instalación?		<input checked="" type="checkbox"/>				
13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	<input checked="" type="checkbox"/>					
14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?				<input checked="" type="checkbox"/>		

Tabla 4.4. Valores de ajuste de complejidad

Mediante los siguientes valores de ajuste de complejidad obtenidos en la Tabla 4.4 calculamos la sumatoria de los F_i :

$$\sum F_i = 5 * 5 + 4 * 3 + 3 * 1 + 2 * 1 + 1 * 2 + 0 * 2 = 43$$

Finalmente calculamos los puntos de función del sistema:

$$PF = \text{Cuenta Total} * [0.65 + 0.01 * \sum F_i]$$

$$PF = 834 * [0,65 + 0,01 * 43] = 900,72$$

Por lo tanto el PF es de 900,72, de donde se concluye que el sistema tiene una funcionalidad óptima, tomando en cuenta la siguiente escala:

<p>PF > 300 ⇒ <i>Óptimo</i></p> <p>200 >= PF >= 300 ⇒ <i>Buena</i></p> <p>100 >= PF > 200 ⇒ <i>Suficiente</i></p> <p>PF < 100 ⇒ <i>Deficiente</i></p>

4.4. MANTENIBILIDAD DE SOFTWARE

Para calcular la estabilidad de del sistema, es decir, índice de madurez del software (IMS), se establecerá los cambios que ocurrieron con cada versión del producto. Para esto el IMS se determina con la siguiente información:

- MT = Número de módulos en la versión actual.
- Fc = Número de módulos en la versión actual que se han cambiado.
- Fa = Número de módulos en la versión actual que se han añadido.
- Fe = Número de módulos en la versión actual que se han eliminado.

El índice de madurez del software se calcula de la siguiente manera:

$$IMS = [MT - (Fc + Fa + Fe)] / MT$$

En el sistema se obtuvieron los siguientes valores como muestra la Tabla 4.5 para la información requerida para el IMS:

Información	Valores obtenidos
MT	8
Fc	1
Fa	1
Fe	0

Tabla 4.5 Información requerida por el IMS [Elaboración propia]

Ahora calculamos el índice de madurez del software sustituyendo los valores de la Tabla 4.5 los cuales son resultados obtenidos del sistema:

$$\text{IMS} = 8 - (1 + 1 + 0) / 8 = 0,75$$

Por tanto tomando en cuenta la escala siguiente se concluye que el IMS obtenido tiene una estabilidad alta al final de la evolución en las versiones logradas.

75% <= IMS <=100% ⇒ <i>Optima</i>
50% <= IMS <75% ⇒ <i>Buena</i>
25% <= IMS <50% ⇒ <i>Suficiente</i>
0% <= IMS < 25% ⇒ <i>Deficiente</i>

4.5. PORTABILIDAD DEL SOFTWARE

Mediante la métrica de facilidad de instalación se calcular el factor de portabilidad, mediante este se obtiene el porcentaje de éxitos de instalación de los usuarios responsables de realizar la instalación del sistema.

4.5.1. FACILIDAD DE INSTALACION

La facilidad de instalación viene dado por la siguiente relación:

$$X = A / B$$

A = Número de casos de éxitos de la operación innatación por parte del usuario.

B = Número total de operaciones de instalación que realizó el usuario.

Entonces se tiene que:

$$X = 8/9 \Rightarrow X = 0.89$$

Por tanto existe un 89 % de probabilidad que el usuario instale exitosamente el sistema y de acuerdo a la escala que se muestra en la parte inferior se concluye que el sistema es portable.

$75\% \leq X \leq 100\% \Rightarrow$ *Optima*

$50\% \leq X < 75\% \Rightarrow$ *Buena*

$25\% \leq X < 50\% \Rightarrow$ *Suficiente*

$0\% \leq X < 25\% \Rightarrow$ *Deficiente*

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

El Sistema de Información para el seguimiento y control de los préstamos del archivo de la unidad catastro SISCPAUC fue puesto en ejecución realizando la implantación correspondiente en la Unidad de Catastro y sin mayores observaciones por parte de los usuarios, por tanto se puede apreciar claramente los beneficios que aportó el sistema SISCPAUC, esto se logró cumpliendo los objetivos planteados en el Capítulo I:

- ✓ Se automatizó el proceso de solicitudes de carpetas para sus préstamos, de manera que cada usuario que desee realizar esta tarea, lo pueda efectuar desde su computador además de ser anoticiados por el mismo sistema cuando ya fue atendida su solicitud; logrando así eliminar el llenado manual de boletas.

- ✓ El proceso de recepción de las solicitudes también fue automatizada de tal forma que el archivista puede realizar esta tarea desde la ventana

correspondiente, generando boletas de préstamo más ordenadas y legibles; con esto se eliminó la planilla de registro de préstamos que era llenado por los funcionarios internos a la Unidad de Catastro, realizando así un óptimo control y seguimiento de este proceso.

- ✓ El proceso de devolución ahora es realizada por parte del archivista más organizada y rápidamente, eliminando también el llenado manual de boletas correspondiente a la devolución creando así un óptimo control y seguimiento de este proceso.
- ✓ El registro de los trámites entrantes a la unidad ahora se lo realiza de una manera fácil, organizada y segura, brindando al usuario (archivista) distintas opciones de búsquedas y reportes.
- ✓ El sistema mediante el módulo de recepción de solicitudes realiza el inventariado automático de las carpetas (los tipos de carpetas que son inventariados automáticamente puede verse en la Tabla 1.1), esta tarea también puede ser realizada de manera manual por el usuario.
- ✓ El mantenimiento de la información se la realiza de manera fácil y entendible para el usuario, realizando un buen registro de las carpetas de planimetrías, datos para la configuración y desenvolvimiento del sistema.
- ✓ La seguridad del sistema puede ser administrada por el usuario de manera segura, contando con un control de accesos basada en roles.
- ✓ El sistema ofrece al usuario final una gama de reportes para los módulos de recepción de solicitudes, devoluciones, registro de trámites, registro carpetas de planimetrías, registro de planimetrías, inventariado de carpetas.

- ✓ El agente de interfaz fue incorporado para informar al usuario de las tareas pendientes que tenga, como por ejemplo informar a los funcionarios internos y externos que sus solicitudes ya fueron atendidas en el archivo, informa al archivista que llegaron solicitudes las cuales no fueron atendidas; también brinda información de cómo debe realizar algunas tareas en el sistema.

Por lo tanto el software construido, que se encuentra en la etapa de utilización, se convirtió para los usuarios (funcionarios internos y externos a la Unidad de Catastro) una herramienta que brinda un óptimo control y seguimiento de todos los procesos de los cuales se hablaron en este proyecto, brindando también una información íntegra, precisa y oportuna.

El empleo de la metodología RUP para la construcción del software fue muy beneficioso por la característica de retroalimentación del sistema, logrando obtener una buena arquitectura del software.

5.2. RECOMENDACIONES

Las recomendaciones que se puede dar tras un análisis de la implantación y el funcionamiento del sistema, son las siguientes:

- A los administradores del sistema, dar pleno cumplimiento a las políticas de seguridad de la información adoptadas en el proyecto.
- La Jefatura de la Unidad de Catastro, debe promover e instruir el registro de la información en todos los procesos ya automatizados, ya que, el sistema constituye un aporte valioso para control y seguimiento de estos procesos, para tal efecto se recomienda equipar por lo menos con una máquina

adicional para el archivo y otra para uso exclusivo de funcionarios externos.

- Es recomendable que se realice copias de seguridad cada mes, para evitar la pérdida de información debe a algún tipo de desastre.
- La información nueva introducida al sistema deberá ser previamente revisados, para que los reportes emitidos por el sistema sean fiables para los usuarios que los requieran.
- Se recomienda que todos los usuarios del sistema cambien su contraseña semanalmente o mensualmente para dar mayor seguridad al sistema.
- Es necesario que la Unidad de Catastro realice una reingeniería de procesos para que conjuntamente con los procesos ya automatizados, se llegue a optimizar los tiempos de trabajo.

BIBLIOGRAFÍA

REFERENCIAS BIBLIOGRAFICAS

[MAMANI, 2002] MAMANI GOMEZ, H. D., (2002): *Sistema de Información del Archivo Central Corte Suprema de Justicia de La Paz.*

[MENDOZA, 2001] MENDOZA, M., DELGADO, F., (2001): *Sistema de Información del Archivo par ala Dirección Distrital de Educación El Alto –Distrito Norte.*

[HERNANDEZ, 2001] HERNÁNDEZ, E., (2001): *El Lenguaje Unificado de Modelado (UML).*

[MICROSOFT, 2003] MICROSOFT, (2003):*Documentación MSDN de Visual Studio .Net*

[JACOBSON,BOOCH,RUMBAUGH, 2000] JACOBSON, I., BOOCH, G., RUMBAUGH, J., (2000): *El proceso unificado de desarrollo de software*

[PRESSMAN, 2001] PRESSMAN, R. S., (2001): *Ingeniería de software enfoque práctico*

[PRESSMAN, 1998] PRESSMAN, R. S.,(1998): *Ingeniería de software enfoque práctico*

REFERENCIAS DE INTERNET

[NAVARRO, 2002] NAVARRO, A.,(2002): *Proceso Unificado de Desarrollo*
<http://www.fdi.ucm.es>

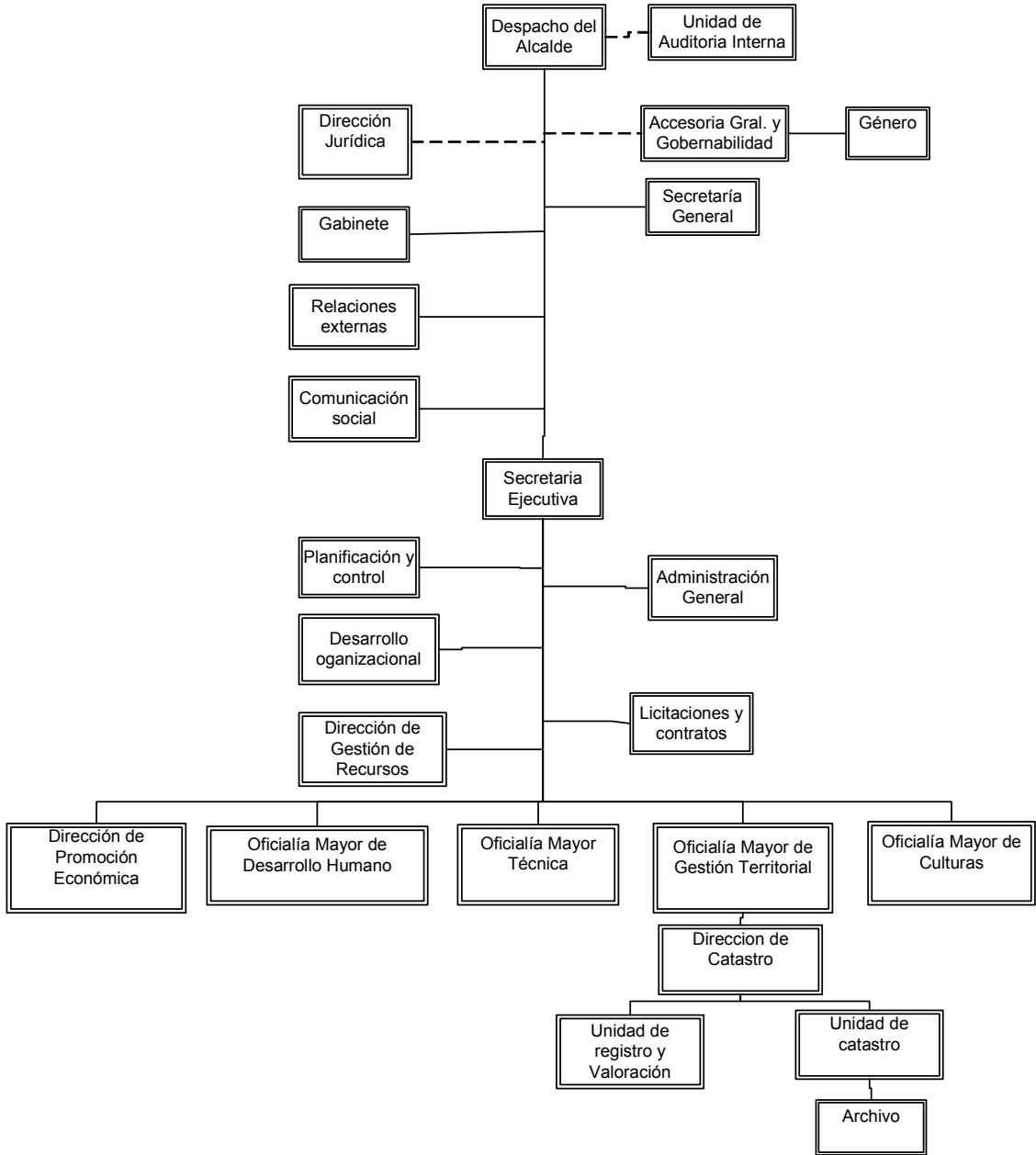
[MICROSOFT, 2005] MICROSOFT MSDN, (2005): *Información general acerca de Visual Studio .NET 2003*, <http://www.microsoft.com>

[GMLP, 2005] GMLP, (2005): *Portal del Gobierno municipal de La Paz*,
<http://www.ci-lapaz.gov.bo>

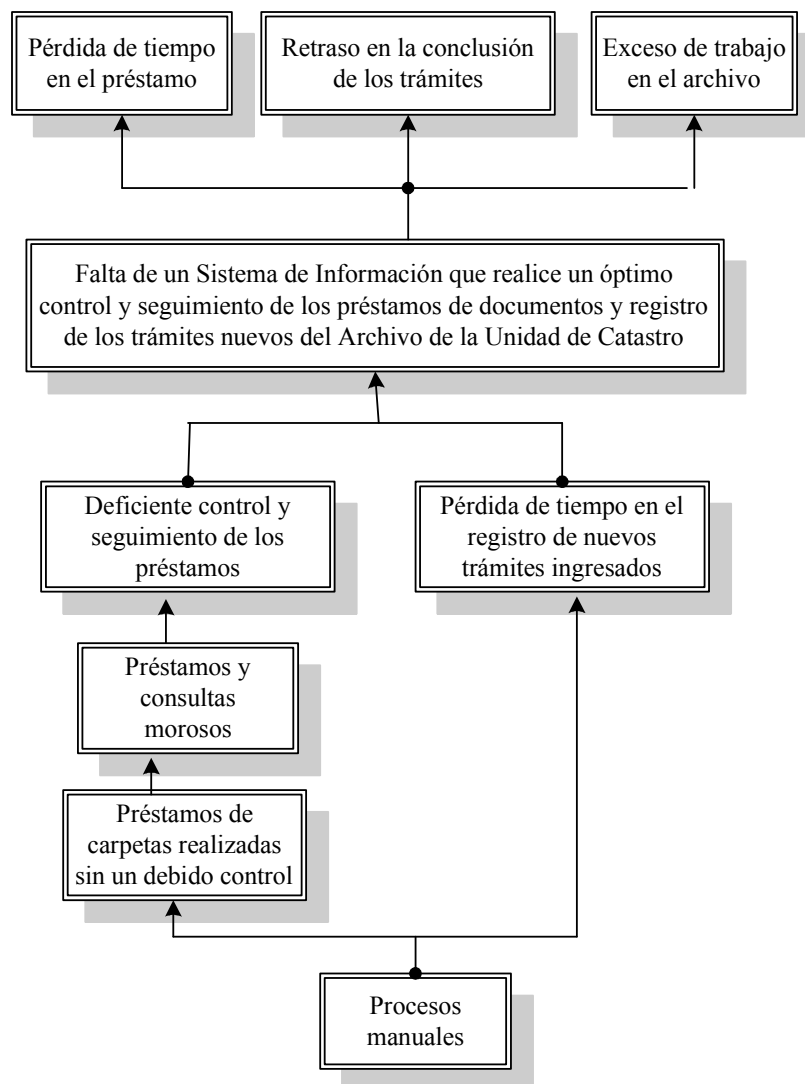
[NUÑEZ, 2005] NUÑEZ, P., (2005): *Aplicabilidad de la gestión de calidad de software* <http://www.issi.uned.es>

ANEXO A

ORGANIGRAMA DE LA EMPRESA



ANEXO B
ARBOL DE PROBLEMAS



ANEXO C
ARBOL DE OBJETIVOS

