

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



PROYECTO DE GRADO

**Sistema de Información de
Seguimiento de Trámites Vía Web
Gobierno Municipal de El Alto**

PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA

MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

AUTOR : RENAN CANCARI GUARACHI
TUTOR : Lic. MARIO LOAYZA MOLINA
REVISOR : Lic. JAVIER REYES PACHECO

LA PAZ – BOLIVIA

2009

Dedicatoria

*Agradezco a nuestro supremo creador por
acompañarme, iluminarme y abrirme las puertas
a nuevas oportunidades en todo momento de la vida.*

*A mis padres Dario y Rosa quienes siempre confiaron en mi
y me dieron la oportunidad de poder cumplir una de las metas
trazadas y por ser un pilar fundamental en mi vida.*

AGRADECIMIENTOS

Deseo Expresar mi profundo agradecimiento a las siguientes personas:

A mi Tutor Lic. Mario Loayza Molina, por el apoyo y orientación que me brindo en la elaboración y conclusión de mi proyecto de grado.

A mi Revisor Lic. Javier Reyes Pacheco, por su disponibilidad y sugerencias en el proceso de revisión del proyecto.

A la Universidad Mayor de San Andrés, que abre sus puertas a los estudiantes dispuestos a lograr sus aspiraciones profesionales.

Al Sr. Tec. Mario Quispe Jefe de la Unidad de Sistemas, a la Lic. Loyola N. Larico Patzi responsable de los Proyectos realizados en la Unidad de Sistemas, a la Srta. Stria. Soveida Valdez Bullain, y el Sr. Tec. Raúl Choque Paco encargado del Departamento Técnico de la Unidad de Sistemas quienes son funcionarios del Gobierno Municipal de El Alto , por brindarme su confianza , apoyo , sugerencias en el desarrollo del proyecto.

A mis padres Dario y Rosa y mis hermanos Eddy y Selena quienes me brindaron su apoyo, paciencia y comprensión en mi etapa de estudio.

A mi amiga Lic. Rosmery Salazar quien es una persona especial a quien siempre recordare por el apoyo moral que me dio a la distancia en cada momento desde el inicio de mi proyecto.

A todos mis grandes amigos Rolando Hinojosa, Martin Cruz y todo el club de SCHOOL FRIENDS quienes me brindaron su ayuda y apoyo moral durante mi etapa de estudio,

Muchas gracias.

RESUMEN

El Gobierno Municipal de El Alto es una entidad pública en competencia, para regular, controlar y supervisar todas las actividades de la Urbe Alteña. El presente proyecto beneficia a los funcionarios internos de la Institución y usuarios externos, así como los organismos estatales y privados, que requieran información del estado del trámite realizado.

En el presente proyecto se realizó un análisis y diseño de cada uno de los procesos, utilizando como metodología el Rational Unified Process (RUP) y parte de la metodología Object Oriented Hypermedia Design Method (OOHDM). Así mismo comprende de cinco capítulos los cuales son.

Capítulo I Introducción, donde se describen los antecedentes, la situación problemática que atraviesa las oficinas del Gobierno Municipal de El Alto en cuanto a la información requerida de cada trámite el cual constituye la base para formular el problema principal. Se presenta además los objetivos generales y específicos.

Capítulo II Marco teórico, donde se presentan y describen tanto las definiciones teóricas como conceptos que se utilizaron en el presente proyecto.

Capítulo III Análisis, Diseño e Implementación del nuevo sistema, donde se realiza un análisis de la situación actual de los procesos que se lleva a cabo en el Gobierno Municipal de El Alto, para luego aplicar las metodologías mencionadas y realizar el desarrollo del sistema propuesto para llegar al objetivo planteado.

Capítulo IV Calidad del software, donde se realiza una serie de medidas de calidad del sistema.

Y para terminar tenemos el capítulo V Conclusión y Recomendaciones, de todo el trabajo realizado en el presente proyecto.

INDICE DE CONTENIDO

1 INTRODUCCION

1.1 INTRODUCCION	1
1.1.1 ANTECEDENTES	2
1.1.2 ANTECEDENTES DE LA INSTITUCION.....	2
1.1.3 ANTECEDENTES DEL PROYECTO	3
1.2 PLANTEAMIENTO DEL PROBLEMA	4
1.3 FORMULACION DEL PROBLEMA.....	5
1.4 SISTEMATIZACION DEL PROBLEMA	5
1.5 OBJETIVOS DE LA INVESTIGACION	5
1.5.1 OBJETIVO GENERAL	5
1.5.2 OBJETIVOS ESPECIFICOS.....	5
1.5.3 LIMITES Y ALCANCES.....	6
1.5.4 APORTES	6
1.5.5 METODOLOGIA	6
1.6 JUSTIFICACION DE LA INVESTIGACION.....	7
1.6.1 JUSTIFICACION TEORICA.....	7
1.6.2 JUSTIFICACION METODOLOGICA.....	7
1.6.3 JUSTIFICACION PRACTICA	8
1.6.3.1 JUSTIFICACION ECONOMICA	8
1.6.3.2 JUSTIFICACION SOCIAL	8

2 MARCO TEORICO

2.1 INTRODUCCIÓN.....	9
2.2. RATIONAL UNIFIED PROCESS (RUP).....	9
2.2.1 LA VIDA DEL PROCESO UNIFICADO.....	10
2.2.1.1 FASE DE INICIO	11
2.2.1.2 FASE DE ELABORACIÓN.....	12
2.2.1.3 CONSTRUCCIÓN	12
2.2.1.4 TRANSICIÓN.....	13
2.2.2 ETAPAS DEL DESARROLLO DEL PRODUCTO.....	14
2.2.2.1 MODELO DEL NEGOCIO	14

2.2.2.2 ADMINISTRACIÓN DE REQUERIMIENTOS.....	15
2.2.2.4 ANÁLISIS.....	17
2.2.2.4.1 ANÁLISIS DE CLASES	17
2.2.2.4.2 DIAGRAMA DE CLASES.....	18
2.2.2.4.3 DIAGRAMA DE COLABORACIÓN.....	18
2.2.2.5 DISEÑO.....	19
2.2.2.5.1 CLASES DE DISEÑO.....	19
2.2.2.6 DIAGRAMA DE SECUENCIA.....	20
2.2.2.7 DIAGRAMA DE ESTADOS.....	20
2.2.2.8 ESTRUCTURA JERÁRQUICA.....	21
2.2.2.9 IMPLEMENTACIÓN.....	22
2.2.2.9.1 MODELO DE IMPLEMENTACIÓN.....	22
2.2.2.9.2 DIAGRAMA DE DESPLIEGUE.....	22
2.2.2.10 PRUEBA.....	23
2.3 OBJECT ORIENTED HYPERMEDIA DESIGN METHOD (OOHDM)	23
2.3.1 DESCRIPCIÓN DE LAS ETAPAS.....	23
2.3.1.1 DISEÑO NAVEGACIONAL.....	23
2.3.1.2 DISEÑO DE INTERFACES ABSTRACTAS.....	24
2.3.1.3 IMPLEMENTACIÓN.....	25
2.4 REDES DE PLANEAMIENTO Y CONTROL.....	25
2.4.1 REDES DE PERT	25
2.5 RECURSOS TECNOLÓGICOS.....	27
2.5.1 VISUAL STUDIO.NET.....	27
2.5.2 LENGUAJE DE CONSULTA ESTRUCTURADO SQL.....	28
2.6 CALIDAD DEL SOFTWARE.....	29
2.7 MÉTRICAS DE CALIDAD.....	31
2.7.1 MÉTRICA DE PUNTO DE FUNCIÓN (PF).....	32
3 MARCO APLICATIVO	
3.1 INTRODUCCION.....	33
3.2 ANALISIS Y DISEÑO PROPUESTO	34
3.2.1 FASE DE INICIO	36
3.2.1.1 MODELADO DE NEGOCIO.....	36

3.2.1.2 MODELOS DE CASOS DE USO DEL SISTEMA	38
3.2.1.3 ANALISIS DE REQUERIMIENTO.....	40
3.2.2 FASE DE ELABORACION	42
3.2.2.1 DIAGRAMA DE COLABORACION.....	46
3.2.2.2 DIAGRAMA DE SECUENCIA.....	47
3.2.2.3 DIAGRAMA DE ESTADOS.....	49
3.2.2.4 DIAGRAMA DE CLASES DE DISEÑO	51
3.2.2.5 ESTRUCTURA JERARQUICAS.....	52
3.2.2.6 DIAGRAMA DE COMPONENTES.....	52
3.2.2.7 DIAGRAMA DE DESPLIEGUE.....	53
3.2.2.8 DIAGRAMA NAVEGACIONAL.....	54
3.2.2.9 DIAGRAMA DE INTERZAS DE ABSTRACCION	55
3.2.2.10 INTERFAZ DE USUARIO.....	56
3.2.3 FASE DE CONSTRUCCION.....	57
3.2.4 FASE DE TRANSICION.....	60
4 CALIDAD DE SOFTWARE	
4.1 INTRODUCCION	64
4.2 FUNCIONALIDAD	64
4.3 CONFIABILIDAD	67
4.4 MANTENIMIENTO	68
4.5 USABILIDAD.....	69
5 CONCLUSIONES Y RECOMENDACIONES	
5.1 CONCLUSIONES	72
5.2 RECOMENDACIONES.....	73

BIBLIOGRAFIA..... 74

ANEXOS

ANEXO A: ARBOL DE PROBLEMAS

ANEXO B: ARBOL DE OBJETIVO

ANEXO C: MATRIZ DEL MARCO LOGICO

ANEXO D: ACTORES

ANEXO E: DESCRIPCION DE LOS CASOS DE USO

ANEXO F: DIAGRAMA DE COLABORACION

ANEXO G: DIAGRAMA DE ESTADOS



INDICE DE FIGURAS

DESCRIPCIÓN	PÁG.
FIGURA 2.1 UN CICLO CON SUS FASES ITERACIONES	10
FIGURA 2.2 FASES E ITERACIONES DE LA METODOLOGÍA RUP...	11
FIGURA 2.3 DEPENDENCIA ENTRE EL MODELO DE CASOS DE USO	14
FIGURA 2.4 REPRESENTACIÓN GRAFICA DE UN ACTOR.....	16
FIGURA 2.5 REPRESENTACIÓN GRAFICA DE CASOS DE USO ...	16
FIGURA 2.6 ESTEREOTIPOS DE UML	17
FIGURA 2.7 DIAGRAMA DE CLASES DE CASOS DE USO	18
FIGURA 2.8 UN DIAGRAMA DE COLABORACIÓN.....	19
FIGURA 2.9 LA CLASE DE DISEÑO EJEMPLO FACTURA	20
FIGURA 2.10 DIAGRAMA DE SECUENCIA CASO BANCO	20
FIGURA 2.11 DIAGRAMA DE ESTADOS	21
FIGURA 2.12 ESTRUCTURA JERÁRQUICA PARA WEB APP.....	21
FIGURA 2.13 DIAGRAMA DE COMPONENTES.....	22
FIGURA 2.14 DIAGRAMA DE DESPLIEGUE	22
FIGURA 2.15 AQUÍ MOSTRAMOS UN ESQUEMA NAVEGACIONAL	24
FIGURA 2.16 IMPLEMENTACIÓN DE UN PROTOTIPO.....	25
FIGURA 2.17 CALCULO DEL TIEMPO ESPERADO	26
FIGURA 2.18 DESVIACION ESTANDAR.....	26
FIGURA 2.19 VARIANZA	26
FIGURA 2.20 ESTIMADO PROBABILISTICO DEL TIEMPO TOTAL...	26
FIGURA 2.21 DESVIACION ESTANDAR DEL TIEMPO TOTAL.....	26
FIGURA 3.1 FLUJO DE INFORMACION DE LA INSTITUCION.....	33
FIGURA 3.2 REDUCCION DE TIEMO EN EL MANEJO DE INF.....	34
FIGURA 3.3 MODELADO DEL NEGOCIO	37
FIGURA 3.4 MODELADO DE CASOS DE USO DEL SISTEMA.....	38

FIGURA 3.5 CASOS DE USO: ACCESO AL SISTEMA.....	43
FIGURA 3.6 CASOS DE USO REGISTRO DEL PER. AUTORIZADO	44
FIGURA 3.7 CASOS DE USO: REGISTRO Y SEGUIMIENTO.....	45
FIGURA 3.8 CASOS DE USO: SOLICITUD DE FORMULARIO.....	45
FIGURA 3.9 CASOS DE USO: REPORTES DE SEG. DE TRÁMITES	46
FIGURA 3.10 DIAGRAMA DE COLABORACION: ACCESO AL SIST.	47
FIGURA 3.11 DIAGRAMA DE SECUENCIA: VERIFICAR EL ACCESO	47
FIGURA 3.12 DIAGRAMA DE SECUENCIA: REGISTRO DE FORM..	48
FIGURA 3.13 DIAGRAMA DE SECUENCIA REGISTRO Y PROCESOS	48
FIGURA 3.14 DIAGRAMA DE SECUENCIA REPORTES DE FORM..	49
FIGURA 3.15 DIAGRAMA DE ESTADOS REGISTRO Y SEGUIMIENTO	50
FIGURA 3.16 DIAGRAMA DE CLASES DE DISEÑO DEL SISTEMA...	51
FIGURA 3.17 ESTRUCTURA JERARQUICA DEL SITIO WEB.....	52
FIGURA 3.18 DIAGRAMA DE COMPONENTES DEL SITIO WEB	53
FIGURA 3.19 DIAGRAMA DE DESPLIEGUE DEL SITIO WEB.....	53
FIGURA 3.20 ESQUEMA DE CLASE NAVEGACIONAL DEL SITIO WEB	54
FIGURA 3.21 ADV ADMINISTRADOR DE SISTEMA DE INF.....	55
FIGURA 3.22 INTERFAZ DE USUARIO: INGRESO AL SISTEMA.....	56
FIGURA 3.23 INTERFAZ DE USUARIO: EXTERNO	56
FIGURA 3.24 INTERFAZ DE USUARIO: CONSULTA DE TRÁMITE ..	57
FIGURA 3.25 MODULO DE ADMINISTRADORES.....	57
FIGURA 3.26 MODULO DE FUNCIONARIOS	58
FIGURA 3.27 MODULO DE REGISTRO DE TRAMITE INICIAL.....	59
FIGURA 3.28 MODULO DE SEGUIMIENTO DE TRAMITE.....	59
FIGURA 3.29 MODULO DE REPORTES DE SEGUIMIENTO DE TRA.	60

INDICE DE TABLAS

TABLA 3.1 DEFINICION DEL PLAN DE DESARROLLO.....	35
TABLA 3.2 ATRIBUTOS DEL SISTEMA.....	42
TABLA 3.3 NIVELES DE USUARIO	62
TABLA 3.4 ROLES DE USUARIO	62
TABLA 4.1 CALCULOS DE PUNTO FUNCION.....	64
TABLA 4.2 VALORES DE AJUSTE DE COMPLEJIDAD.....	65
TABLA 4.3 DETALLES DE PARAMETROS	70
TABLA 4.4 EVALUACION DE FASES.....	71





CAPITULO 1

INTRODUCCION

1. INTRODUCCION

1.1 INTRODUCCION

Con la permanente expansión del Internet y la avanzada tecnología en esta era, se ha permitido que el World Wide Web (www) sea ampliamente utilizado como un medio que mejora la comunicación y difusión de la información a nivel mundial.

En el área de la Informática, las organizaciones e instituciones han optado por establecer una presencia en la red de redes (Internet), no solo con el propósito de presentación a la sociedad, sino brindando una plataforma acorde a las necesidades de las organizaciones, ofreciendo interfaces con usuarios potenciales, tomando en cuenta que es un medio de comunicación imprescindible para conformar grupos de difusión entre los actores, por medio de los servicios que brindan los municipios.

Por este motivo existe la inquietud en el Gobierno Municipal de El Alto en ingresar al mundo de las oportunidades, brindado por medio de la herramienta denominada Internet. Poder difundir la información necesaria y oportuna.

El Seguimiento de Trámites Vía Web para el Gobierno Municipal de El Alto surge de la necesidad de hacer conocer a los funcionarios del municipio y a la población el estado y la ubicación respectiva del trámite que ha sido solicitado.

Teniendo así toda la información oportuna de cada tramites cuyos destinos son las unidades, direcciones y oficialías del municipio [POA08]. De esta manera se pueda capturar, procesar y presentar la información en un tiempo mas reducido.

Siendo así el ciudadano Alteño el mas beneficiado donde reduce en gran manera los gastos de dinero y tiempo que invierte para hacer el seguimiento personal de su tramite mediante llenado de formularios alargándose mas la culminación de dicho tramite .

1.1.1 ANTECEDENTES

1.1.2 ANTECEDENTE DE LA INSTITUCION

La base legal que sostiene todas y cada una de las acciones del Gobierno Municipal de El Alto, desde su creación como una Sección de Provincia de la Provincia Murillo, con su capital El Alto, mediante Ley 728 de 6 de marzo de 1985, en la Ley de Municipalidades, la Ley de Descentralización y la Ley de Participación Popular entre las más importantes.

La ciudad de El Alto fue elevada a rango de ciudad, por la Ley 1014 de fecha 26 de septiembre de 1988, el mismo que dio lugar a la división de las ciudad de La Paz y el Alto, Asumiendo está los limites urbanos definidos por la Ley del 22 de diciembre de 1968.

La Constitución Política del Estado de 2 de febrero de 1967 y la Ley N° 1585 de Reforma de la C. P. E. de 12 de agosto de 1994, Arts. 200 al 206 y la Ley N° 1615 de 6 de febrero de 1995 de adecuaciones y concordancias de la C.P.E.

La Municipalidad se constituye en un gobierno local autónomo, en la entidad de derecho público con personalidad jurídica reconocida y patrimonio propio, que representa institucionalmente al municipio, forma parte del Estado y contribuye a la realización de sus fines, de conformidad a la Ley de Municipalidades del 28 de octubre de 1999.

El nuevo marco jurídico normativo definido a partir del año 1994 transforman y amplían las funciones y el rol de los Municipios.

A través de la Ley de Participación Popular, N° 1551 de 20 de abril de 1994, la Ley de Modificaciones y Ampliaciones a la Ley 1551 de Participación, Ley N° 1702 de 17 de julio de 1996, se establece que la jurisdicción territorial de los gobiernos municipales es la Sección de Provincia, de esta forma se amplían las competencias municipales al ámbito rural y se les transfiere la infraestructura física de los servicios de salud, educación, cultura, deportes, caminos vecinales y micro-riego que estuvieran bajo su jurisdicción encomendándoles su cuidado

y ampliación. Estas responsabilidades posibilitan ser a los municipios importantes agentes del Desarrollo local y los mismos que van acompañados de nuevas asignaciones.

La Ley de Administración y Control Gubernamental (SAFCO), Ley N°1178 de 20 de julio de 1990, permitió la aplicación de algunos Sistemas de Administración que se vienen desarrollando actualmente en el Municipio, tales como el SISPLAN, SPO, SNIP y los referidos al manejo administrativo financiero.

La ubicación geográfica

El Municipio de El Alto está ubicada en una meseta de superficie plana y ondulada, al pie de la Cordillera de La Paz (Meseta del Altiplano Norte) y de la Cordillera Oriental, al Noreste de Bolivia a 16°30' Sur y 68°12' Oeste. A una altura de 4.050 m.s.n.m.

Limites Territoriales

Limita al Norte con la Sección Capital de la Provincia Murillo, al Este con el Municipio de La Paz, al Sur con el Cantón Viacha, perteneciente a la Provincia Ingavi, al Oeste con el Cantón Laja, correspondiente a la Segunda Sección de la Provincia Los Andes.

Extensión

La extensión territorial del Municipio de El Alto, considerando el Distrito 103, es aproximadamente, de 387.56 Km². (38.756 Has), que representa el 7.58% de la superficie total de la Provincia Murillo: el 40.24% (15.596 Has.) corresponde al área urbana y el 59.76% (23.160 Has.) corresponde al área rural.

1.1.3 ANTECEDENTES DEL PROYECTO

En cuanto a proyectos de grado relacionados con el seguimiento de trámites en el Gobierno Municipal de El Alto, desarrollados en la carrera de informática no se encuentran sistemas similares, sin embargo se pueden citar trabajos relacionados:

- Sistemas de Trámites funcionando en la Honorable Alcaldía Municipal de La Paz conocido como el SISTRAM.
- Sistema de Control de Trámites Utilizando Maquinas de Turing Caso División de Gestiones Admisiones y Registro UMSA. Código T.1435 Autor Lic. Patricia Justina Nacho
- Sistema de Control y Seguimiento de Trámites para el Ministerio de Trabajo Código T 1518 Autor Lic. Miguel Angel Choque

1.2 PLANTEAMIENTO DEL PROBLEMA

El Gobierno Municipal de El Alto en el desenvolvimiento de sus funciones, se enfrenta con los siguientes problemas:

- Debido a que se recaba datos de distintas unidades surge la dificultad en la búsqueda y el seguimiento de trámites.
- Existen formularios con correcciones y tachaduras en el llenado del proveído causando confusión y/o duda en la recepción del trámite.
- La entrega de trámites son retrasados por la cola de trámites sin atender.
- La elaboración de informes y/o reportes de trámites se demora en un alto grado.
- Mal control y administración de trámites internos y externos

Finalmente en base a los problemas mencionados hace notar la carencia de un proceso centralizado y automatizado para la recolección y procesamiento de datos y manejo de información para el seguimiento de trámites. Ya que los procedimientos manuales generan una enorme pérdida de tiempo y elaboración de reportes tardíos. Se busca una alternativa de usar modelos y

metodologías para trámites, integrados en el Sistema de Información y Seguimiento de Trámites.

1.3 FORMULACION DEL PROBLEMA

¿El Desarrollo del Sistema de Información de Seguimiento Vía Web para el Gobierno Municipal de El Alto proporcionara información requerida del estado de un trámite realizado en la Institución, requeridos por los usuarios?

1.4 SISTEMATIZACION DEL PROBLEMA

¿De que forma se puede controlar adecuadamente el seguimiento de trámites?

¿Qué seguridad se dará para el acceso a la aplicación web?

¿Cuál será el diseño de base de datos para el seguimiento de trámites?

¿De que forma se puede controlar los indicadores de cada trámite?

¿Cómo diseñar un modulo de alerta de tramites sin atender?

¿Cuándo elaborar reportes de información diarios?

¿Cómo se implementara una red de comunicación entre todas las estaciones de trabajo?

1.5 OBJETIVOS DE LA INVESTIGACION

1.5.1 OBJETIVO GENERAL

Diseñar, Desarrollar e Implementar un Sistema de Información de seguimiento de tramites vía web para proporcionar información requerida por los usuarios internos y externos del Gobierno Municipal de El Alto.

1.5.2 OBJETIVOS ESPECIFICOS

Desarrollar, implementar un sistema de información de seguimiento y control de trámites.

Establecer seguridad para el acceso a la aplicación web.

Diseñar una base de datos para trámites

Diseñar un modulo de control de indicadores

Diseñar un modulo de alerta de tramites sin atender.

Elaborar reportes de acuerdo a requerimientos.

Implementar una red de comunicación entre todas las estaciones de trabajo.

1.5.3 LIMITES Y ALCANCES

LIMITES

No se desarrolla un modulo para la elaboración de formularios que son necesarios en la realización de tramites por lo que se seguirán utilizando el formulario de Registro Único de Tramites (Hoja de Ruta).

ALCANCES

Los alcances del Sistema de Información de Seguimiento de trámites son:

- Módulos de adición, modificación y eliminación del registro único de trámites.
- Registro y control de asignación de tareas a cada usuario para la atención de tramites.
- Reportes de trámites atendidos por cada funcionario.
- Reporte del seguimiento de cada trámite.
- Reporte de trámites finalizados por unidades, direcciones y oficialías.

APORTES

El aporte del presente proyecto es dado por el Sistema de Información de Seguimiento de Tramites vía Web, que pretende llegar a ser una base de

información que permita regular y controlar el manejo de formularios (hoja de ruta) dentro de la Institución. Se utilizara un modelo estadístico consistente en las Redes de Pert, que estime el tiempo requerido para la atención de los trámites.

METODOLOGIA

La metodología a utilizar para el desarrollo del Sistema de Información es el RUP (Rational Unified Process), que es uno de los procesos mas generales de los existentes actualmente , ya que en realidad esta pensado para adaptarse a cualquier proyecto de desarrollo de software.

El software moderno es complejo y novedoso, así que no es realista usar un modelo lineal de desarrollo como el método de cascada. Un proceso iterativo, como el de RUP, permite una comprensión creciente de los requerimientos, a la vez que se va haciendo crecer el sistema. RUP sigue un modelo iterativo que aborda las tareas más riesgosas primero. Así se logra reducir los riesgos del proyecto y tener un subsistema ejecutable tempranamente. RUP divide el proceso de desarrollo en ciclos, donde se obtiene un producto al final de cada ciclo. Cada ciclo se divide en cuatro Fases: Concepción, Elaboración, Construcción y Transición, RUP tiene el propósito de:

- Asegurar la producción de software de calidad dentro de plazos y presupuestos predecibles.
- Obtener un sistema robusto.
- Reducir el Riesgo de tener un mal producto.
- Reducir el Riesgo de no obtener el producto en el tiempo previsto.
- Permitir atacar problemas con requisitos incompletos.

Podemos también destacar que se utiliza el U.M.L. (Lenguaje Unificado de Modelado). Además consultamos (muy superficialmente), para el diseño de hipermedias O.H.D.M. (Metodo de diseño de Hipermedias Orientado a Objetos) presentado por Schuable y Rossi . Que es utilizado como apoyo en el diseño hasta el paso previo de implementación.

1.6 JUSTIFICACION

1.6.1 JUSTIFICACION TEORICA

Se utilizara un modelo estadístico consistente en las Redes de Pert, que estime el tiempo requerido para la atención de los trámites, donde se consideran tres diferentes parámetros de tiempo; que es el tiempo optimista, el tiempo pesimista y el tiempo normal.

1.6.2 JUSTIFICACION METODOLOGICA

Para el análisis y desarrollo del sistema la metodología que se utilizara en el desarrollo del sistema será el Proceso Unificado de la Racional (RUP), porque brinda las siguientes características:

Es iterativo e incremental.

Es dirigido por los casos de uso.

Centrado en la arquitectura.

El ser iterativo e incremental da beneficio de: Obtener un sistema robusto, reducir el riesgo de tener un mal producto, reducir el riesgo de no obtener el producto en el tiempo previsto y permite atacar problemas con requisitos incompletos.

1.6.3 JUSTIFICACION PRÁCTICA

1.6.3.1 JUSTIFICACION ECONOMICA

La automatización de los procesos permitirá reducir costos pendientes a la actualización y mantenimiento dentro del seguimiento y control de trámites.

La atención al público será mas rápida, por lo que se reducirá el tiempo de atención al publico, también se reducirá el gasto de dinero y tiempo de las personas que deben apersonarse al municipio de El Alto para averiguar la situación de sus tramites, pues esta información ahora la tendrán disponible en Internet.

1.6.3.2 JUSTIFICACION SOCIAL

La implementación de un sistema de Información de Seguimiento de trámites se fundamentara en el hecho de poder brindar servicios a la población de la ciudad de El Alto.

Mejorar la atención al público y simplificar la burocracia que existe con los funcionarios de la alcaldía.

Evitar las largas filas de personas que se hacen para informarse de los trámites realizados.





CAPITULO 2

MARCO TEORICO

2. MARCO TEORICO

2.1 INTRODUCCION

En este capítulo se describe las diferentes definiciones empleadas en el sistema así como los métodos y herramientas a utilizar durante el desarrollo del sistema.

2.2 RATIONAL UNIFIED PROCESS (RUP)

El rational unified process [RUP05] es un proceso de desarrollo de Software planteado por Kruchten (1996) cuyo objetivo es producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos.

Sin embargo el Proceso Unificado es más que un simple proceso: es un marco de trabajo genérico que puede especificarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, para diferentes tipos de organizaciones, diferentes niveles de aptitudes y diferentes tamaños de proyecto.

Esta basado en componentes, lo cual quiere decir que el sistema software en construcción esta formado por componentes software interconectado a través de interfaces bien definidas.

Utiliza el lenguaje Unificado de Modelado (unified Modeling Lenguaje, UML) para preparar todos los esquemas de un sistema software.

Se define en tres aspectos claves; dirigidos por casos de uso, centrado en la arquitectura, iterativo e incremental. [BOO99]

Dirigido por casos de uso, quiere decir que el proceso de desarrollo sigue un hilo – avanza a través de una serie de flujos de trabajo que parten de los acaso de uso. Los casos de uso se especifican, se diseñan y los casos de uso finales son la fuente a partir de la cual se construyen los casos de prueba.

Centrado en la arquitectura, quiere decir que la arquitectura, en un sistema software se describe mediante vistas del sistema en construcción. Incluye también los aspectos estáticos y dinámicos más significativos del sistema, reflejados en los casos de uso y otros factores como la plataforma en que funcionara el software, sistema operativo, gestor de base de datos, etc.

Iterativo e Incremental, quiere decir que el desarrollo de un producto software comercial supone un gran esfuerzo que podría durar 6 meses o hasta un año o más. Es práctico dividir el trabajo en partes más pequeñas o mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en el flujo de trabajo y los incrementos al crecimiento del producto.

2.4.1 LA VIDA DEL PROCESO UNIFICADO

El Proceso Unificado se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema como se muestra en la figura 2.1

Figura 2.1 Un ciclo con sus fases iteraciones.

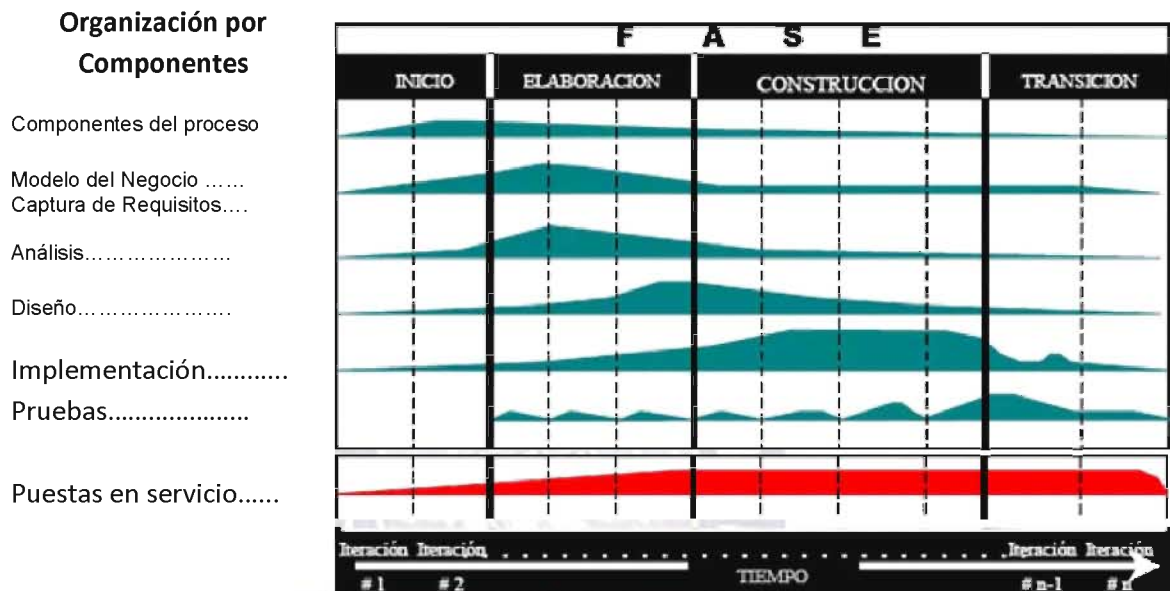
Fuente: [RUP00]



Cada ciclo consta de cuatro fases: inicio, elaboración, construcción y transición. Cada fase se subdivide a su vez en iteraciones como se puede observar en la siguiente figura 2.2

Figura 2.2 Fases e Iteraciones de la Metodología RUP

Fuente: [IAC00]



2.2.1.1 FASE DE INICIO

Es necesario en función del alcance tener una idea de la arquitectura.

- Una lista de características.
- Una primera versión del modelo de negocio (o del dominio) que describe el contexto del sistema.
- Un esbozo de los modelos que representan una primera versión del modelo de casos de uso, el modelo de análisis y el modelo de diseño.
- Un primer esquema de la descripción de una arquitectura candidata, que perfila las vistas de los modelos de casos de uso, análisis, diseño e implementación.
- Posiblemente, un prototipo exploratorio que muestra el uso del nuevo sistema
- Una lista inicial de riesgos y una clasificación de casos de uso.

- Los rudimentos de un plan para el proyecto en su totalidad, incluyendo el plan general de las fases.
- Un primer borrador del análisis de negocio.

2.2.1.2 FASE ELABORACIÓN

En la recopilación tenemos varias tareas:

- Recopilar la mayor parte de los requisitos pendientes
- Establecer una base de la arquitectura sólida (línea base) transición así como en posteriores generaciones.
- Continuar la observación y control de los riesgos críticos que aún quedan
- Completar los detalles del plan del proyecto.

Para continuar la elaboración se recaba información que se recibe de la fase de inicio:

- Un plan para la fase de elaboración
- Un modelo de casos de uso parcialmente completo
- Una descripción de la arquitectura candidata.
- Un prototipo que muestre el funcionamiento del sistema.

2.2.1.3 CONSTRUCCIÓN

Preparar un producto software en su versión operativa inicial (versión beta).

- Debe tener la calidad adecuada para su aplicación
- Debe cumplir los requisitos

Para cumplir el objetivo:

- Se parte de la línea base de la arquitectura ejecutable
- Se detallan los casos de uso y escenarios resultantes
- Se cierran los métodos de análisis-diseño-implementación
- Se integran los subsistemas y se prueban

- Se integra el sistema completo y se prueba
- El plan de proyecto para la fase de transición
- El sistema software ejecutable
- Todos los artefactos, incluyendo los modelos del sistema
- La descripción de la arquitectura, mínimamente modificada y actualizada
- Una versión preliminar del manual de usuario, lo suficientemente detallado como para guiar a los usuarios de la beta.

2.2.1.4 TRANSICIÓN

Cumplir los requisitos establecidos en las fases anteriores hasta la **satisfacción** de todos los usuarios, además de gestionar los aspectos relativos a la operación en el entorno del usuario.

Se recibe información de los usuarios para:

- Determinar si el sistema hace lo que debe hacer
- Descubrir riesgos inesperados
- Anotar problemas no resueltos
- Encontrar fallos
- Eliminar ambigüedades y lagunas en la documentación del usuario
- Centrarse en áreas en las que los usuarios muestren deficiencias y necesiten información o formación

La transición finaliza con el lanzamiento del Producto

- El propio software ejecutable, incluyendo el software de instalación
- Documentos legales como contratos, licencias, renunciaciones de derechos y garantías.
- La versión completa y corregida de línea base de la versión del producto, incluyendo todos los modelos del sistema
- La descripción completa y actualizada de la arquitectura
- Manuales y material de formación del usuario final, del operador y del administrador del sistema

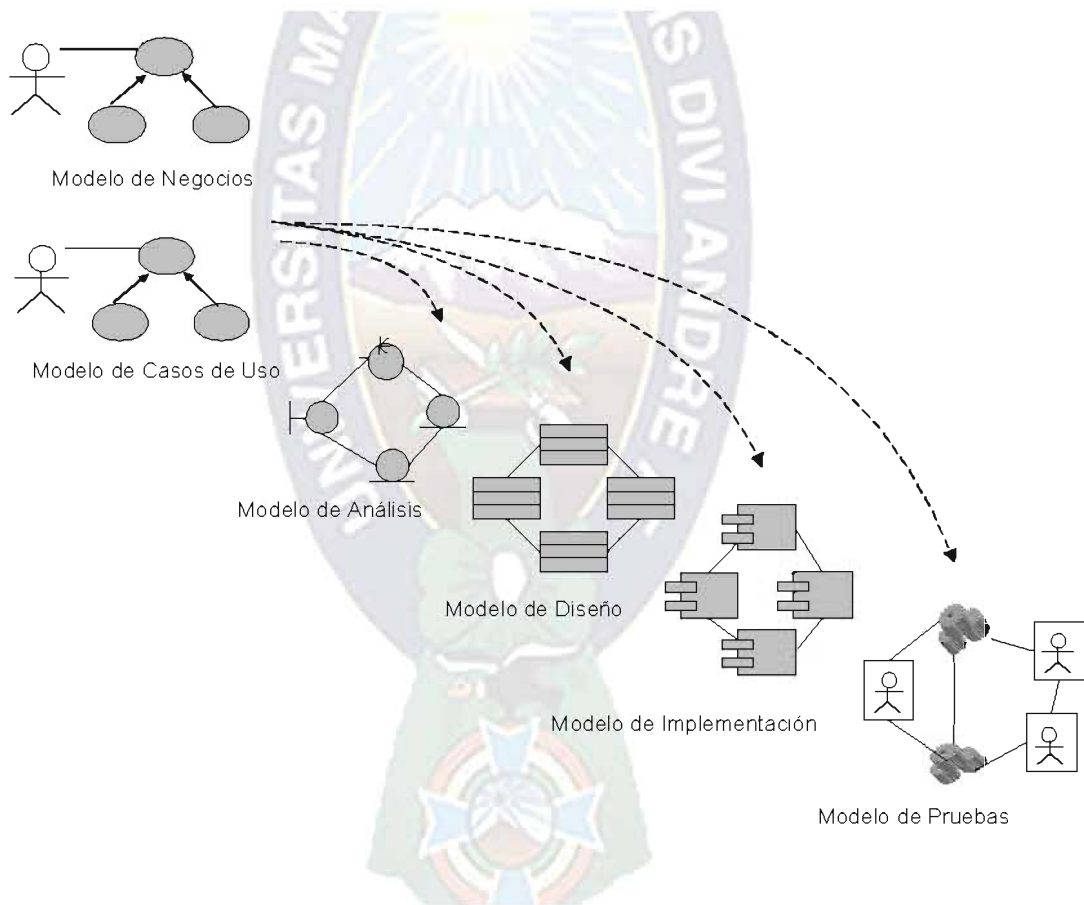
- Referencias para la ayuda del cliente, acerca de dónde encontrar más información, cómo informar de defectos o dónde encontrar información sobre defectos y actualizaciones.

2.2.2 ETAPAS DEL DESARROLLO DEL PRODUCTO

Mostraremos como se realiza el análisis de acuerdo a estos modelos, ver figura 2.3

Figura 2.3 Se indica la dependencia entre el modelo de casos de uso y los demás modelos.

Fuente: [RUP00]



2.2.2.1 MODELO DEL NEGOCIO

Un modelo del negocio representa a los procesos del negocio que son una rama de actividades dentro de la dinámica de una organización. Comprende los Diagramas de casos de Uso del Negocio, que muestran la funcionalidad

proporcionada en conjunto por una organización. Este diagrama describe las interacciones entre los casos de uso del negocio y los actores del negocio (cliente externo) que se usan para analizar el contexto del sistema. Los diagramas de casos de uso del negocio representan los procesos que un negocio realiza; mientras que los actores representan los roles o papeles que actúan precisamente con el negocio.

2.2.2.2 ADMINISTRACIÓN DE REQUERIMIENTOS

El propósito de la administración de requerimientos es establecer un entendimiento común entre el cliente y los requerimientos de este que serán abordados en el proyecto de software. Este entendimiento con el cliente conforma la base para la planificación y administración del proyecto de software.

A) REQUERIMIENTO FUNCIONALES

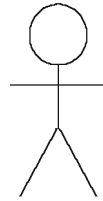
En el presente proyecto se aplica la captura de requerimiento funcional, ya que gracias a los requerimientos se utiliza los casos de uso que nos ayuda a representar el uso del sistema de forma mas sencilla y clara, también se esta especificando los modelos de interfaz para los usuarios.

Modelo de casos de uso; El modelo de casos de uso permite que los desarrolladores de software y los clientes lleguen a un acuerdo sobre los requisitos, condiciones y posibilidades que debe cumplir el sistema. Un modelo de casos de uso es un modelo del sistema que contiene **actores, cosas de uso y sus relaciones [BOO99]**

- **Actor;** Los actores representan terceros fuera del sistema que colaboran con el sistema. Un actor juega un papel por cada caso de uso con el colabora. Cada vez que un usuario en concreto (un humano u otro sistema) interactúa con el sistema, la instancia correspondiente del actor esta desarrollando ese papel. Una instancia de un actor es un usuario concreto que interactúa con el sistema. Figura 2.4

Figura 2.4 Representación grafica de un actor

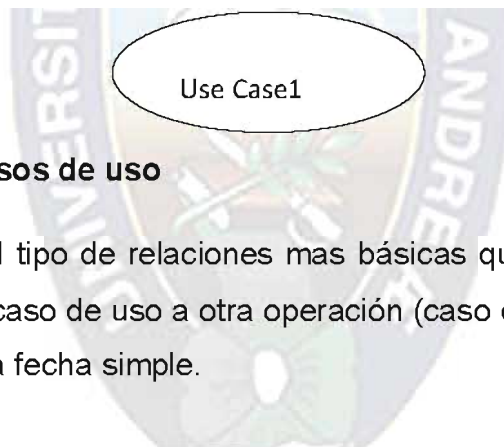
Fuente: [SCH00]



- **Casos de uso;** Cada forma en que los actores usan el sistema se representa con un caso de uso. Son fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. Un caso de uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores. Figura 2.5

Figura 2.5 Representación grafica de casos de uso

Fuente: [SCH00]



Relaciones entre casos de uso

- **Asociación,** es el tipo de relaciones mas básicas que indica la invocación desde un actor o caso de uso a otra operación (caso de uso). Dicha relación se denota con una línea simple.

Asociación



- **Generalización,** esta ultima es la mas usada y cumple dos funciones, la de uso cuando un actor utiliza el caso de uso y la de herencia cuando un caso de uso es similar a otro en sus características.

<< usa >>

<< extends >>



- **Descripción de casos de uso;** Es un detalle de cada caso de uso donde se describe su flujo de sucesos en detalle, incluyendo como comienza, termina e interactúa con los actores.

B) REQUERIMIENTOS NO FUNCIONALES

Los requerimientos no funcionales especifican propiedades del sistema, como restricciones del entorno o de la implementación, rendimiento, dependencia de la plataforma, en un mundo real, se encarga de verificar la velocidad, rendimiento, tiempo de respuesta y el uso de memoria del sistema.

2.2.2.4 ANALISIS

En el análisis podemos estructurar los requisitos de manera que nos facilite su comprensión, su preparación, su modificación, y en general, su mantenimiento. Esta estructura (basada en clases de análisis y paquetes) es independiente de la estructura que se dio a los requisitos (basado en casos de uso). Sin embargo se utiliza la estructura de clases y paquetes estereotipados ya que nos proporciona la estructura a la vista interna.

2.2.2.4.1 Análisis De Clases Para el desarrollo de las clases de análisis se hace uso de tres estereotipos que están estandarizados en UML, nos sirven para distinguir el ámbito de las diferentes clases, ver figura 2.6.

Figura 2.6 Estereotipos de UML

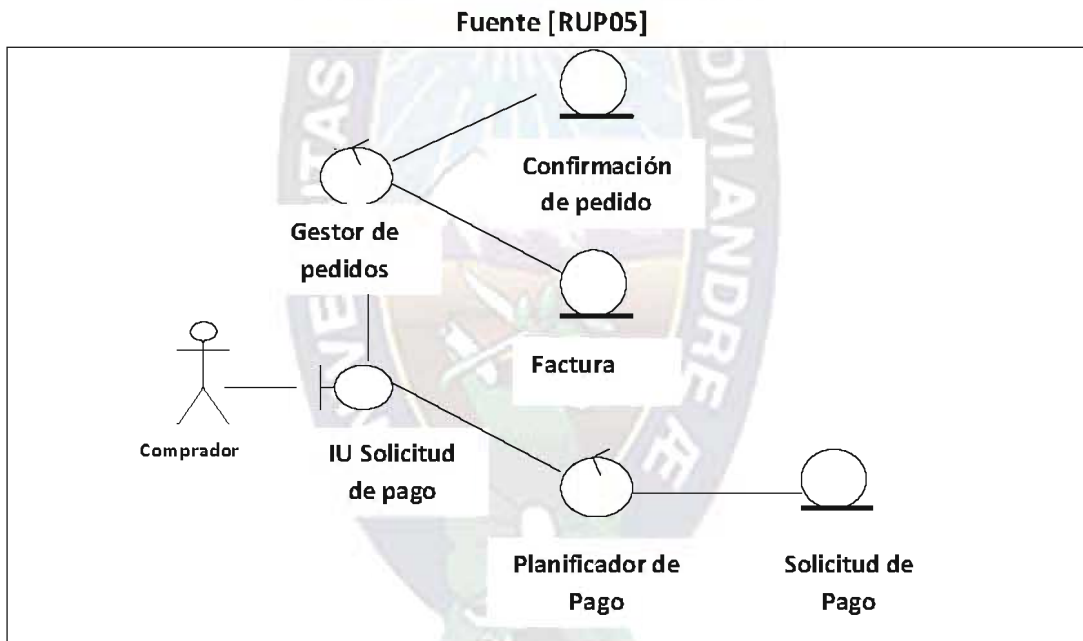
Fuente: [RUP00]



2.2.2.4.2 Diagrama De Clases

Una clase de análisis y sus objetos normalmente participan en varias realizaciones de casos de uso, y algunas de las responsabilidades, atributos y asociaciones de una clase concreta suelen ser sólo relevantes para una única realización de casos de uso. Por tanto, es importante durante el análisis coordinar todos los requisitos sobre una clase y sus objetos que pueden tener diferentes casos de uso. Para hacerlo, adjuntamos diagramas de clases a las realizaciones de casos de uso, mostrando sus clases participantes y sus realizaciones, ver figura 2.7.

Figura 2.7 Un diagrama de clases de una realización de caso de uso, ejemplo Pagar Factura.

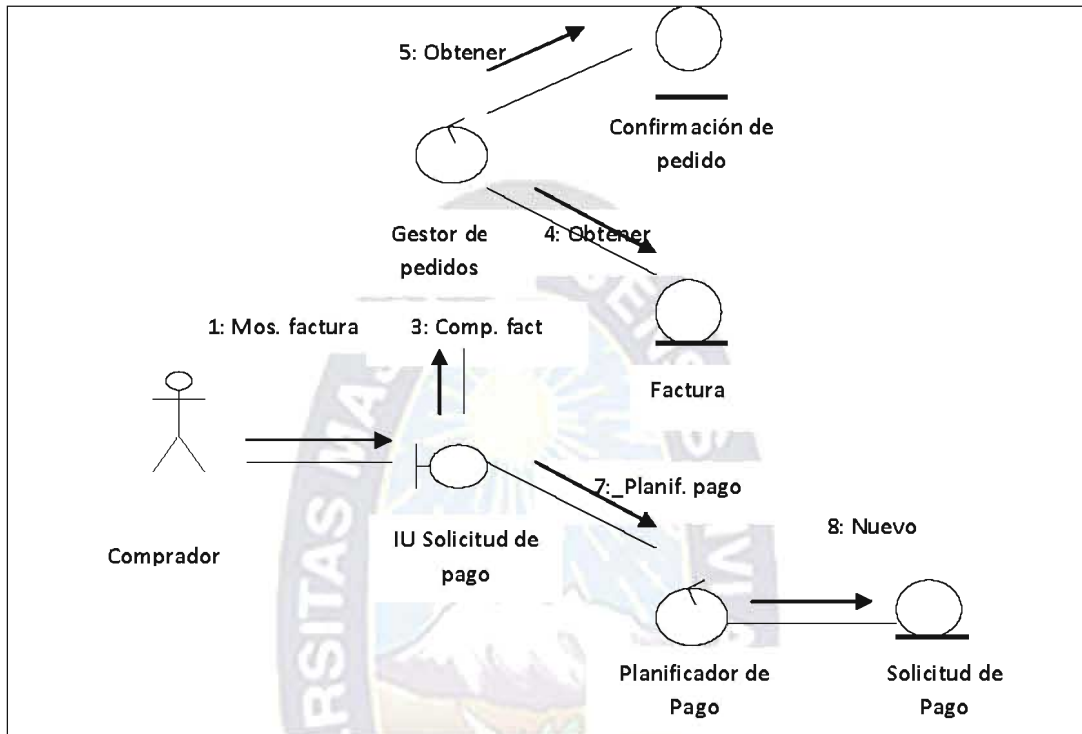


2.2.2.4.3 Diagrama De Colaboración

La secuencia de acciones en un caso de uso comienza cuando un actor invoca el caso de uso mediante el envío de algún tipo de mensaje al sistema. En el sistema Análisis preferimos mostrar esto con diagramas de colaboración ya que nuestro objetivo fundamental es identificar requisitos y responsabilidades sobre los objetos, ver figura 2.8.

Figura 2.8 Un diagrama de colaboración de una realización de caso de uso, ejemplo Pagar Factura.

Fuente [RUP05]



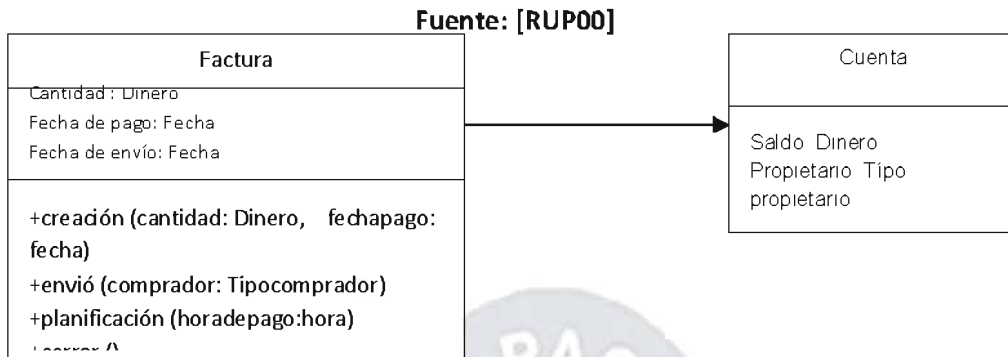
2.2.2.5 DISEÑO

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 y crear un plano del modelado de implementación.

2.2.2.5.1 Clase De Diseño

Una clase de diseño puede activarse, implicando que objetos de la clase mantengan su propio hito de control y se ejecuten concurrentemente con otros objetos activos. No obstante, las clases del diseño no están normalmente activas, lo que implica que sus objetos se ejecuten en el espacio de direcciones y bajo el control de otros objetos activos. En concreto, esto puede ser apropiado cuando hay muchas clases activas cuyos objetos tienen interacciones complejas, ver figura 2.9.

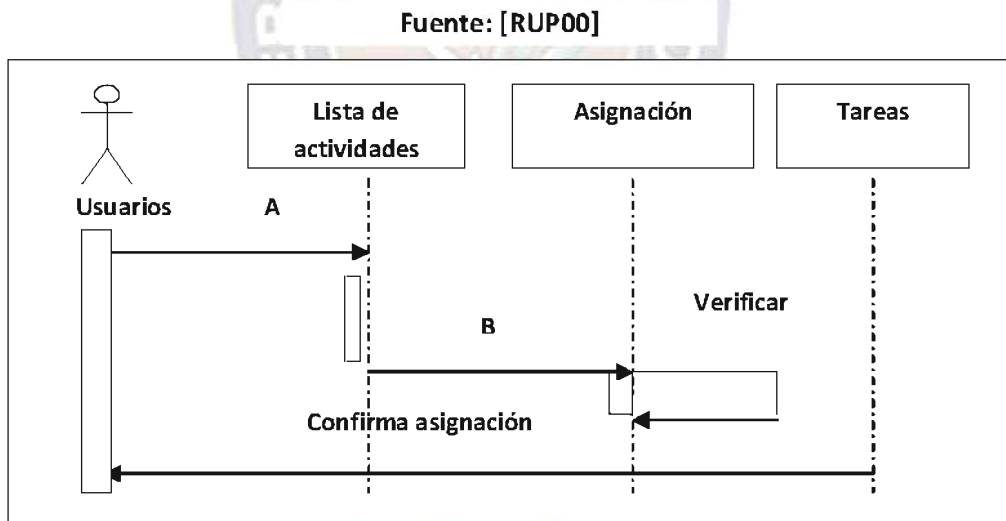
Figura 2.9 la clase de diseño ejemplo factura con sus atributos



2.2.2.6 DIAGRAMA DE SECUENCIA

Si los casos de uso tienen varios flujos o subflujos distintos, suelen ser útil en crear un diagrama de secuencia para cada uno de ellos. Esto puede hacer más clara la realización del caso de uso, y también permite extraer diagramas de secuencia que representen interacciones generales y reutilizables, ver figura 2.10.

Figura 2.10 Diagrama de Secuencia caso banco

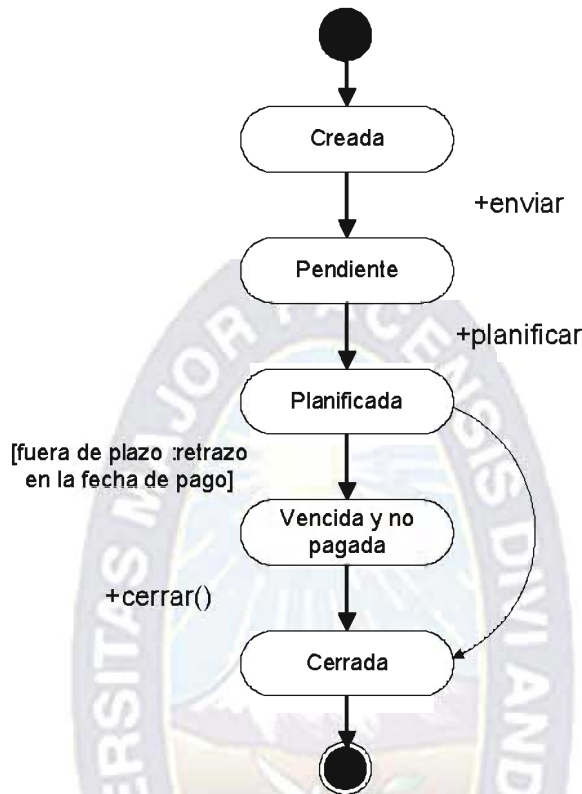


2.2.2.7 DIAGRAMA DE ESTADOS

El diagrama de estados determina el comportamiento de los mensajes recibidos. En este caso, es significativa la utilización de diagramas de estado para describir las diferentes transiciones de estados de un objeto del diseño. Cada diagrama de estados es entonces una entrada de valor para la implementación de la correspondiente clase del diseño, ver figura 2.11.

Figura 2.11 Diagrama de estados para la clase factura.

Fuente: [RUP00]

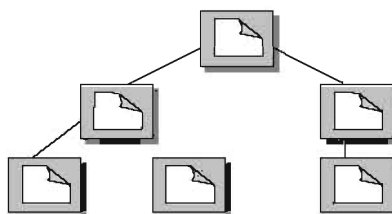


2.2.2.8 ESTRUCTURAS JERÁRQUICAS

Es un punto importante para el desarrollo de sitios o aplicaciones orientadas a la Web, la arquitectura de Aplicaciones Web (WebApp) más común, nos permite diseñar una estructura jerárquica de la WebApp que posibilita (por medio de la ramificación de hipertexto) el flujo de control en horizontal atravesando las ramas verticales de la estructura. Por tanto el contenido presentando en la rama de un extremo de la jerarquía puede tener enlaces de hipertexto que nos lleven al contenido que existe en otra rama, ver figura 2.12.

Figura 2.12 estructura jerárquica para WebApp

Fuente: [PRE02]



2.2.2.9 IMPLEMENTACIÓN

En esta etapa se debe, escribir en un lenguaje de programación concreto, el código de la aplicación. Dicho lenguaje debe ser escogido de manera que sea el que más se adecue a las necesidades de la aplicación y a los objetivos perseguidos. La implementación de un sistema software es algo más que únicamente escribir y compilar el código. Se debe considerar aspectos de las tecnologías utilizadas como la interacción con base de datos.

2.2.2.9.1 Modelo de implementación

Que comprende del diagrama de componentes ya que nos muestra la organización y dependencia de un conjunto de componentes, cubren la vista de implementación estático de un sistema, ver la figura 2.13

Figura 2.13 Diagrama de Componentes

Fuente: [JAC00]

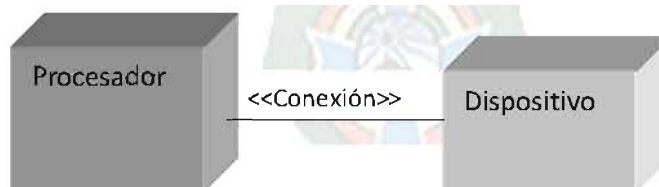


2.2.2.9.2 Diagrama de Despliegue

Los diagramas de despliegue servirán para modelar la configuración hardware pero mas la del software del sistema, mostrando que nodos lo componen y su apariencia, ver la figura 2.14.

Figura 2.14 Diagrama de Despliegue

Fuente: [JAC00]



2.2.2.10 PRUEBA

En esta última etapa se comprueba que el sistema cumpla los requisitos establecidos y que se comporta de acuerdo a las especificaciones indicadas. Para ello se utiliza los casos de uso del sistema, de tal manera que para cada uno de ellos se comprueba que el sistema se comporta de la forma esperada.

2.3 OBJECT ORIENTED HYPERMEDIA DESIGN METHOD (OOHDM)

El término HIPERMEDIA, es la combinación de los conceptos HIPERtexto y multiMEDIA, hace referencia a una tecnología de construcción de (hiper)documento que permite a los lectores encontrar fácilmente la información que realmente necesitan, de la manera que ellos decidan, a través de enlaces establecidos por el autor entre los diferentes elementos de información multimedia (texto, sonido, imagen, video, etc.) que conforman el documento. La metodología OOHDM (Object Oriented Hypermedia Design Method), propuesta por D. Schwabe y G. Rossi, establece que el desarrollo de un hiperdocumento (diseño de sitios web).

OOHDM propone el desarrollo de aplicaciones hipermedia a través de un proceso compuesto por cuatro etapas: diseño conceptual, diseño navegacional, diseño de interfaces abstractas e implementación [EAD05].

Esta metodología plantea la necesidad de capturar los requisitos de navegación. Para ello propone técnicas como los casos de uso o los escenarios para definir las necesidades del sistema.

2.3.1 DESCRIPCION DE LAS ETAPAS

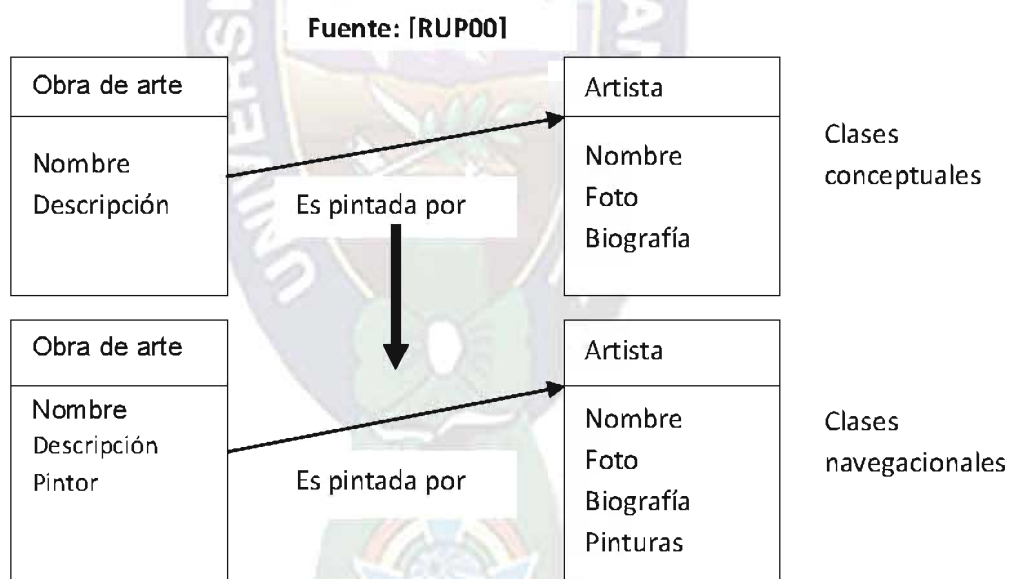
2.3.1.1 Diseño Navegacional

Una vez obteniendo el esquema conceptual, se establece una fase de diseño navegacional en la que se ha de definir la estructura de navegación a través del hiperdocumento mediante la realización de modelos navegacionales que representen diferentes vistas del esquema conceptual de la fase anterior.

El diseño de navegación es expresado en dos esquemas: el esquema de clases navegacionales y el esquema de contextos navegacionales. En OOHDM existe un conjunto de tipos predefinidos de clases navegacionales: nodos, enlaces y estructuras de acceso. La semántica de los nodos y los enlaces son las tradicionales de las aplicaciones hipertexto, y las estructuras de acceso, tales como índices o recorridos guiados, representan los posibles caminos de acceso a los nodos [MDA03].

Las estructuras de navegación por otro lado enriquecen este diagrama con patrones de navegación conocidos (menús, índices, etc.) que permiten navegar a través de las clases de contexto, ver figura 2.15

Figura 2.15 Aquí mostramos un esquema navegacional



2.3.1.2 Diseño de Interfaces Abstractas.- En esta fase se especifican, entre otros aspectos, la apariencia de los objetos de navegación, así como los objetos activadores de esa navegación y las distintas transformaciones que puede sufrir la interfaz.

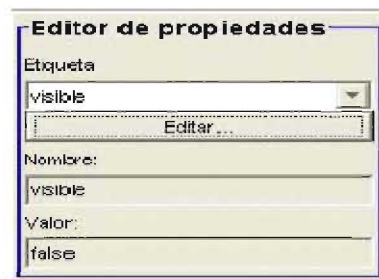
Para describir la interfaz de las hipermedias, OOHDM utiliza el modelo Vista de Datos Abstracto (ADV), que especifica la organización y el comportamiento de la interfaz.

2.3.1.3 IMPLEMENTACIÓN

En esta última etapa la metodología OOHDM incluye un entorno de desarrollo que permite el prototipado rápido de aplicaciones, y la generación de aplicaciones hipermedia basadas en módulos, que producen páginas dinámicas, ver figura 2.16 [OOH05].

Figura 2.16 implementación de un prototipo

Fuente: [RUP00]



2.4 REDES DE PLANEAMIENTO Y CONTROL

2.4.1 REDES DE PERT

PERT (Técnica de revisión y evaluación de proyectos), es una de las herramientas administrativas de planificación y control lo principal en una red PERT son los eventos y la interrelación entre eventos son flechas que representan actividades.

En las redes de PERT se tiene que estimar el tiempo requerido para cada una de las actividades. Se tienen tres diferentes estimados de tiempo:

Tiempo Optimista.- Cuando toda actividad funciona perfectamente sin ningún problema.

Tiempo Pesimista.- Es el tiempo estimado tomando en cuenta que existieran muchos problemas.

Tiempo Normal.- Es el tiempo en el cual la actividad debería siempre requerir el mismo tiempo suponiendo que el trabajo es repetido constantemente en condiciones idénticas.

Figura 2.17 Calculo del tiempo esperado

Fuente: [EST91]

$$T_e = \frac{a+4m+b}{6}$$

Donde

- a: tiempo optimista
- b: tiempo pesimista
- c: tiempo normal

Figura 2.18 Desviación Estándar

Fuente: [EST91]

$$G_{Te} = \frac{b-a}{6}$$

Figura 2.19 Varianza

Fuente: [EST91]

$$V_{Te} = \frac{(b-a)^2}{6^2}$$

Figura 2.20 Estimado Probabilístico del Tiempo Total

Fuente: [EST91]

$$T_{eT} = T_{e1} + T_{e2} + T_{e3} + \dots + T_{en}$$

Figura 2.21 Desviación Estándar del Tiempo Total

Fuente: [EST91]

$$G_{TeT} = \sqrt{T_{e1}^2 + T_{e2}^2 + T_{e3}^2 + \dots + T_{en}^2}$$

2.5 RECURSOS TECNOLÓGICOS

2.5.1 VISUAL STUDIO.NET

Visual Basic. NET (VB.NET), es una de las herramientas estrella para el desarrollo sobre esta nueva plataforma de trabajo, estamos asistiendo a una evolución/revolución sin precedentes en el mundo de la informática. VB.NET ayuda a producir aplicaciones de consola, servicios Windows, interfaces Web, componentes de servidores ASP.NET y servicios Web.

En el proyecto se utiliza la herramienta de desarrollo web Asp.Net. Por lo tanto al cargar una página ASP en nuestro navegador, en realidad no estamos cargando la página ASP como tal, sino el resultado de la ejecución de la página ASP, que tiene como salida mediante código HTML. Es decir, son páginas que se ejecutan en el servidor enviando como resultado al cliente código HTML.

Antes de seguir vamos a definir de forma sencilla lo que se considera un lenguaje de script o de secuencia de comandos. Un lenguaje de script es un subconjunto de otro lenguaje más general y que se utiliza para un entorno muy determinado, en este caso el entorno es la Web.

Una página ASP podrá contener los siguientes elementos: texto, componentes ActiveX, código HTML y comandos de script. Este script puede ser de dos tipos: script de cliente o script de servidor. El script de servidor es la nueva idea que introduce ASP, se debe tener en cuenta que en el script de servidor se tiene acceso a diferentes objetos y no está orientado a eventos.

El script de servidor utilizado en ASP utiliza la misma sintaxis que el script de cliente, la diferencia está en que con ASP el script de servidor es compilado y procesado por el servidor Web antes de que la página sea enviada al navegador.

ASP no es un lenguaje de script, ASP ofrece un entorno para procesar scripts que se incorporan dentro de páginas HTML, es decir, un entorno de procesamiento de scripts de servidor.

La propia Microsoft define ASP de la siguiente manera: "...es un entorno de secuencias de comandos en el lado del servidor que puede utilizar para crear y ejecutar aplicaciones de servidor Web dinámicas, interactivas y de alto rendimiento...".

Al usar aplicaciones en ASP es posible mantener un estado, es decir, se tiene la capacidad de mantener información. Dentro de una aplicación ASP se pueden mantener dos tipos de estado:

- Estado de la aplicación, en la que toda la información relativa a una aplicación está disponible para todos los usuarios de la misma.
- Estado de sesión, en la que la información sólo está disponible para un usuario o sesión específicos. Una sesión por lo tanto, pertenece a un solo usuario.

Las aplicaciones ASP no son aplicaciones al uso, ya que en realidad no se dispone de un ejecutable sino de un conjunto de páginas, imágenes y recursos, por lo tanto se trata de aplicaciones muy particulares que requieren para su ejecución de un servidor Web que soporte las páginas ASP.

2.5.2 LENGUAJE DE CONSULTA ESTRUCTURADO SQL

SQL Server es un sistema administrador para Bases de Datos relacionales basadas en la arquitectura Cliente / Servidor una red de ordenadores. El ordenador cliente es el que inicia la consulta; el ordenador servidor es que atiende esa consulta.

El SQL permite básicamente:

- Definir una base de datos mediante tablas
- Almacenar información en tablas
- Seleccionar la información que sea necesaria de la base de datos

- Combinar y calcular datos para conseguir la información necesaria

2.6 CALIDAD DEL SOFTWARE

Es importante examinar las características del software en cuanto a la calidad, para Pressman [PRE02] la calidad de concordancia es el grado de cumplimiento de las especificaciones de diseño durante su realización. También podemos decir que la calidad de concordancia es un aspecto centrado principalmente en la implementación.

Para aplicar este método debemos medir la calidad del software, primeramente identificando los aspectos particulares del sistema que contribuyan a la calidad global, con el objetivo de relacionar la visión externa del usuario con la visión interna del desarrollador; uno de los modelos que ayuda a determinar estos aspectos es el ISO 9126 que considera seis atributos:

- **Funcionalidad**, evalúa el grado en que el software satisface las necesidades indicadas por los siguientes subatributos: corrección interoperatividad conformidad y seguridad.
- **Confiabilidad**, mide la cantidad de tiempo que el software esta disponible para su uso. Para medir la confiabilidad del sistema se tiene que conocer madurez, tolerancia a fallos y facilidad de recuperación. Para medir podemos aplicar la siguiente función:

$$R_i(t) = e^{-rP(t)}$$

Donde:

R(t): Confiabilidad del subsistema.

r: Probabilidad que se presente una perturbación durante un intervalo de tiempo.

P(t): Probabilidad de fallo del sistema cuando ocurra una perturbación.

La aplicación de la confiabilidad del sistema esta dado por los siguientes teoremas.

Teorema 1. Si n componentes, que funcionan independientemente, están conectados en serie, y si el i-esimo componente tiene confiabilidad $R_i(t)$, entonces la confiabilidad del sistema completo, $R(t)$ esta dada por::

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

Teorema 2. Si n componentes que funcionan independientemente, actúan en paralelo y la i-esima componente tiene confiabilidad $R_i(t)$, entonces la confiabilidad es la siguiente:

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

El nivel de confiabilidad del sistema esta estrictamente relacionado con la cantidad de errores que arroje el sistema de ejecución en una relación inversamente proporcional, es decir, que a menos errores más confiabilidad y viceversa.

- **Facilidad de Uso** Es el grado en que el software es fácil de usar, de acuerdo a la valoración individual por parte de un conjunto de usuarios.
- **Eficiencia** Establece la relación entre el desempeño del software y la cantidad de recursos utilizados bajo condiciones establecidas.
- **Facilidad de Mantenimiento** Se centra en el cambio que va asociado a la corrección de errores (mantenimiento correctivo) durante la creación del sistema; a las adaptaciones requeridas a medida que evoluciona el

- entorno del software (mantenimiento adaptativo), como cambios en las reglas o políticas de la empresa y a la adaptación de nuevas versiones del sistema operativo; así también ha descubrir funciones adicionales que van ha producir beneficios, mas allá de sus requisitos funcionales originales (mantenimiento de mejora o perfectivo). Y finalmente prevee los cambios en programas de computadora a fin de que se puedan corregir, adaptar y mejorar más fácilmente (mantenimiento preventivo o reingeniería del software).
- **Portabilidad** Es el esfuerzo necesario para transferir un programa de un entorno de sistema hardware y/o software a otro. El grado de portabilidad del sistema esta dado por la siguiente ecuación:

$$GP = 1 - (\text{Costo de Transportar} / \text{Costo de Re-desarrollo})$$

Donde:

GP: Grado de Portabilidad

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

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

2.7 MÉTRICAS DE CALIDAD

Para el desarrollo del sistema informático resulta importar utilizar las métricas de calidad ya que las métricas son medidas cuantitativas del grado en que un sistema, posee un atributo dado, de alguna forma permite medir algunos aspectos individuales del proyecto, es decir la métrica proporciona una visión mas profunda. Estas mediciones pueden servir también para identificar los problemas que tiene un sistema informático con el objetivo de solucionarlo [PRE02].

2.7.1 MÉTRICA DE PUNTO DE FUNCIÓN (PF)

La métrica de punto de función (PF) se utiliza para medir el tamaño de un sistema que se obtiene de un modelo de análisis, en la cual se definen las siguientes características:

- Número de Entradas de Usuario
- Número de Salidas de Usuario
- Número de Peticiones de Usuario
- Número de Archivos
- Número de Interfases Externas

$$PF = \text{Cuenta total} * [R(t) + 0.01 * \sum F_i] \quad (4.8)$$

Donde:

Cuenta total: Es la suma de todas las entradas obtenidas.

R(t): Es el porcentaje de confiabilidad.

F_i (i = 1-14): Son valores de ajuste de la complejidad.



CAPITULO 3

MARCO APLICATIVO

3. MARCO APLICATIVO

3.1 INTRODUCCION

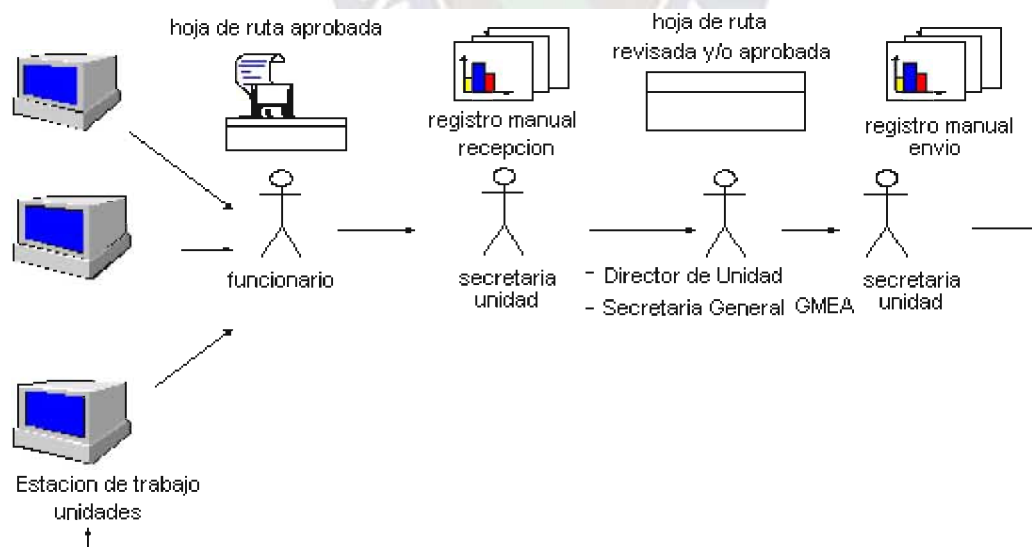
En este capítulo se pretende poner en práctica todo lo mencionado en el anterior capítulo, entendiéndose que se explicara de forma clara, concisa los aspectos relacionados y los diferentes procesos que existen en el sistema propuesto.

Para comenzar, se muestra la figura 3.1, el cual describe los procesos actuales de la institución, posteriormente se realiza el modelado de los mismos con la metodología RUP, como se puede observar en la figura 3.2 y finalmente comienza el desarrollo del sistema...

A continuación identificaremos los procesos que se aplicara, en función a los requerimientos, los cuales están sujetos a posteriores cambios, en algunos casos trascendentales o en otros superficiales. También mostraremos el desenvolvimiento de la recolección de datos de los formularios en el Gobierno Municipal de El Alto, ver Figura 3.1.

Figura 3.1 Flujo de Información En la Institución

Fuente: [Elaboración propia]



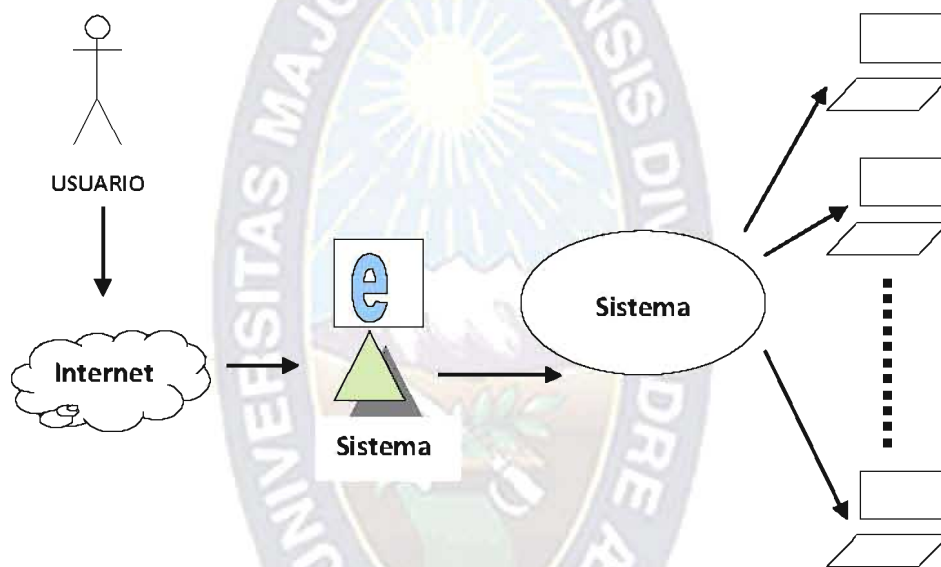
En la figura 3.1 nos muestra como es el desenvolvimiento del llenado de formularios, esta información tiene como recorrido desde una estación de trabajo de una Unidad (origen) a otra Unidad (destino) donde es registrado por secretaria, pasa por la dirección donde se realiza el desglose de la

información. La dirección emite resultados que se envía por secretaria a otras Unidades correspondientes, mostrando así un ciclo de flujo de información hasta culminar su recorrido.

Mostramos como es la reducción del tiempo gracias al sistema realizado, ver figura 3.2.

Figura 3.2 Reducción de tiempo en el manejo de la información

Fuente: [Elaboración propia]



3.2 ANALISIS Y DISEÑO PROPUESTO

Se da comienzo al proceso de planificación inicial a un nivel macro para todo el proyecto, determinando la duración de cada fase en semanas.

Tabla 3.1: Definición del plan de Desarrollo de Software

Fuente: [Elaboración propia]

Nº	ACTIVIDADES	DURACIÓN
	FASE DE INICIO	4 Semanas
1	Desarrollo de actividades	
2	Identificación de actores	
3	Modelado del negocio	
4	Modelado de Casos de Uso	
5	Análisis de requerimientos	
	FASE DE ELABORACION	6 Semanas
6	Definición de Diagrama de Casos de Uso	
7	Definición de Diagrama de colaboración	
8	Definición de Diagrama de secuencia	
9	Definición de Diagrama de estados	
10	Definición de Diagrama de clases de diseño	
11	Definición de la estructura jerárquica del sitio Web	
12	Definición de diagrama de componentes	
13	Definición del diagrama de despliegue	
14	Definición del diagrama navegacional	
15	Definición del diagrama de interfaz abstracta	
16	Interfaz de usuario	
	FESE DE CONSTRUCCION Y PRUEBA	6 Semanas
17	Implementación de esquema de base de datos	
18	Implementación de interfaz de usuario	
19	Implementación de ventanas	
20	Implantación y puestas en marcha el sistema	
21	Planificación de pruebas	
22	Realización de pruebas	

3.2.1 FASE DE INICIO

Se da comienzo al proceso de planificación inicial a un nivel macro para todo el proyecto, a fin de delimitar el ámbito del sistema haciendo un análisis del ciclo de desarrollo de acorde con las actividades que se describen en el Proceso Unificado Racional.

En la primera iteración se destaca a toda la información requerida y obtenida por un trámite y representarlos como requerimientos candidatos a satisfacer la construcción del sistema.

Se muestra en la primera parte de introducción, el desarrollo de actividades de tramites, también se podrá explicar con más detalle sobre los actores que interactúan en el sistema, con mas detalle Ver [Anexos D], a continuación partimos con el desarrollo del modelado del negocio.

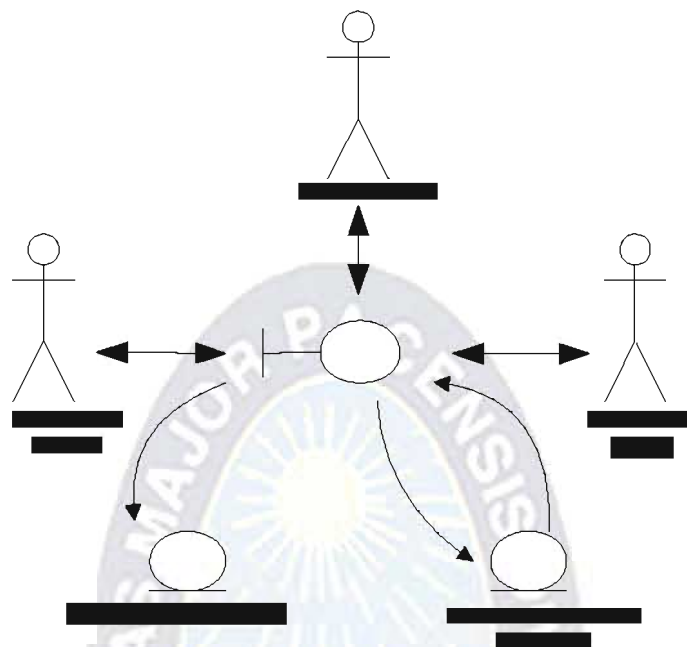
3.2.1.1 MODELADO DEL NEGOCIO

El modelo de negocio es una técnica para comprender los procesos de la organización. El modelo del negocio describe los procesos en términos de casos de uso y actores del contexto que se corresponden con los procesos del sistema y los usuarios. [JAC00]

El modelado del negocio detallado a continuación esta enmarcado dentro de las delimitaciones del contexto del sistema y describe los procesos exactos relacionados con los actores y casos de uso encontrados. Ver Figura 3.3

Figura 3.3 Modelado del Negocio. De Trámites

Fuente: [Elaboración propia]



a. Descripción del modelo del Negocio Servicio de Telefonía

El Funcionario Destino es el que recibe el formulario de la hoja de ruta verificando el contenido del tipo de trámite con el llenado del proveído. Esto ayuda para la habilitación del nuevo llenado y reenvió del formulario a otras instancias. Dicho funcionario esta autorizado para poder sacar reportes del estado de trámite.

La Administración conformados por Directores de cada Dirección, Jefes de cada Unidad y Secretaria General encargados de recolectar todos los reportes finales del estado de tramites realizados en sus respectivas direcciones y unidades . El sistema esta enfocado para proporcionar informes de estado de cada Trámite.

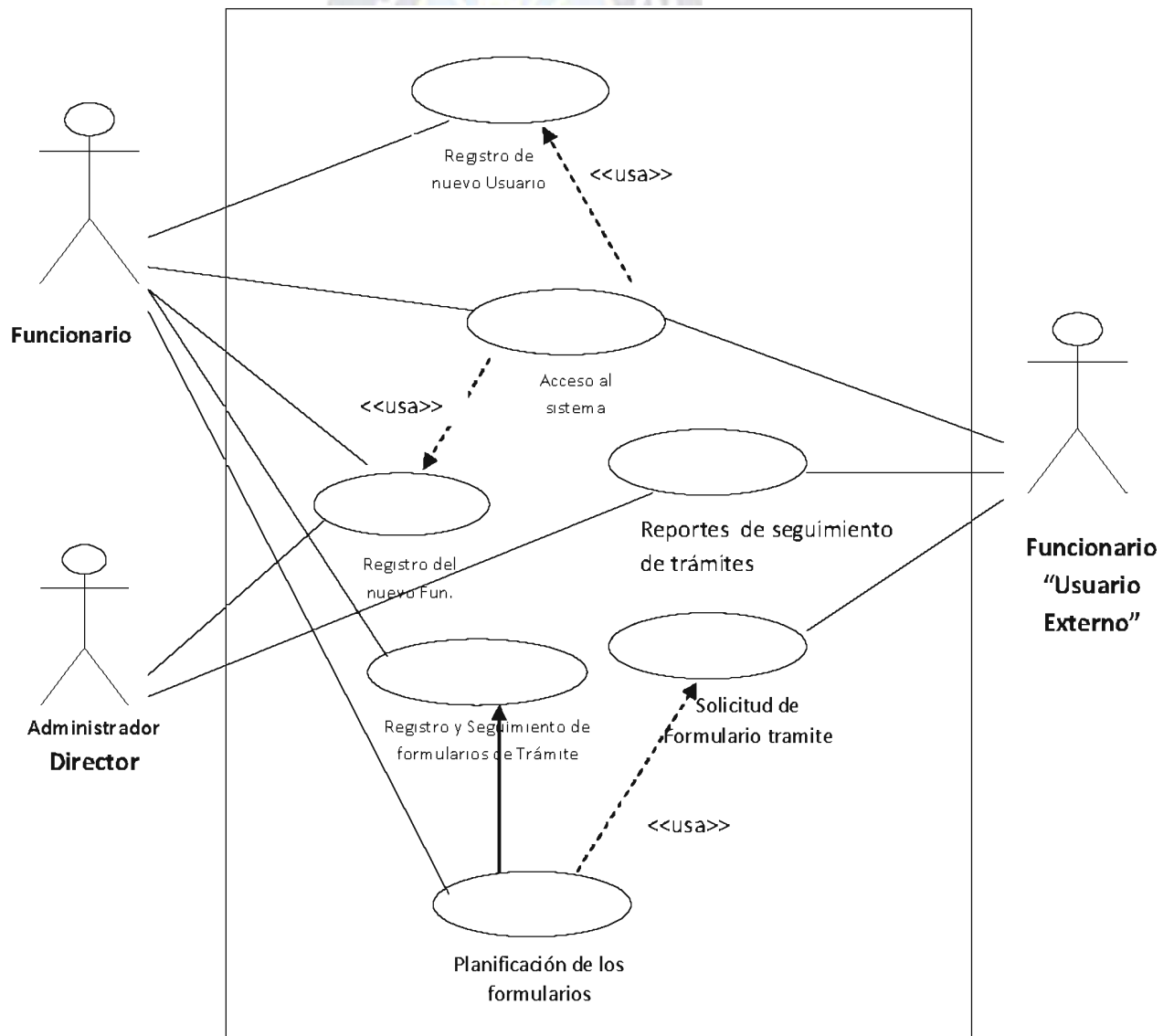
3.2.1.2 MODELO DE CASOS DE USO DEL SISTEMA

Identificamos los actores y casos de uso que intervienen en el sistema, ahora se puede esbozar el modelo de casos de uso para el sistema en cuestión. Ver Figura 3.4

Para delimitar el sistema de su entorno se identifica a los actores externos que interactúan con el sistema de los casos de uso.

Figura 3.4 Modelo de Casos de Uso del Sistema

Fuente: [Elaboración propia]



a. Descripción del modelo de casos de uso

Todos los actores identificados se registran para el acceso al sistema y así poder identificarse con sus funciones de acuerdo a los cargos asignados. El Administrador tiene acceso a realizar consultas de todos los formularios de trámites realizados en su determinada unidad.

El funcionario esta encargado de la recepción, verificación del llenado de los formularios por los funcionarios origen o usuarios externos. Explicaremos un poquito más sobre los casos de uso.

Registro y seguimiento de formularios de trámite: El registro y seguimiento de formularios ayudara a dirigir el proceso que se tiene en la unidad de controlar y registrar todos los formularios que solicitan los funcionarios o usuarios externos para su respectivo llenado, el cual mostrara el seguimiento de todos los tramites realizados en determinadas fechas.

Solicitud de formulario: Los formularios se habilitaran en cuanto se llegue a iniciar un tramite o correspondencia, esto se realiza con una autorización de la Administración o Dirección de cada Unidad.

Registro del nuevo Usuario: El registro del usuario esta respaldado por la unidad de personal en la cual se recaba datos importantes de la persona quien llena el formulario, como ser a que dirección, unidad , a la que pertenece, etc.

Registro del nuevo administrador: El administrador podrá registrar a otro funcionario dentro de su área que no haya sido registrado, para que pueda manejar el sistema.

Reportes de seguimiento: La elaboración de reportes se realiza bajo el proceso que se tiene en la Secretaria General, que muestra seguimientos parciales y finales de los formularios de trámite, donde llega a centralizarse toda la información, en la base de datos.

Planificación de los formularios: En la planificación de formularios esta respaldado con el proceso que se tiene en el Gobierno Municipal de El Alto de acuerdo a reglamentos ya establecido, todo esto para el permiso del llenado de formularios, en fechas específicas bajo requisitos preestablecidos.

Para poder describir los casos de uso, podremos Ver en [Anexo G], la descripción de cada uno.

3.2.1.3 ANALISIS DE REQUERIMIENTOS

Detallaremos los requerimientos de la institución para poder especificar y desarrollar los casos de uso.

En el desarrollo del sistema se han identificado dos tipos de requerimientos que son funcionales y no funcionales.

A) REQUERIMIENTOS FUNCIONALES

La descripción de los requerimientos funcionales del sistema se representa mediante casos de uso y actores.

Casos de Uso: Los casos que se han identificado son los siguientes.

Casos de Uso operaciones de registros: es necesario realizar operaciones sobre los registros de la base de datos del sistema como ser: adicionar un nuevo registro, eliminar un registro existente y modificar un registro existente. Utiliza los casos de uso consultar datos y verificar datos.

Casos de Uso consultar datos: para realizar operaciones de adición, modificación, eliminación y consulta sobre los registros en la base de datos del sistema, primeramente se debe verificar la existencia de los datos, realizando consultas en la base de datos.

Casos de uso verificar datos: cuando se introduce los datos al sistema este deberá verificar la exactitud de los mismos, para que no se almacene información errónea en la base de datos.

Casos de uso elaborar reportes: una vez registrado los datos, es necesario elaborar reportes para realizar el seguimiento correspondiente a los diferentes trámites. Utilizando el caso de uso consultar datos.

Casos de uso asignar formulario: es necesario recabar información de los formularios cuya información nos ayudara a mostrar el desenvolvimiento de la hoja de ruta en la institución.

Casos de uso elaborar cronograma: el cronograma tiene que estar registrado, para poder habilitar el formulario para el llenado de los funcionarios y usuarios. Utiliza los casos de uso consultar datos, asignar formularios.

Caso de uso inscribir a nuevos funcionarios usuarios: para tener un encargado de los formularios es necesario registrar a los usuarios con datos necesarios que reflejen su identificación actualizada indicando la dirección y/o unidad a la que representan y registrar la información que se genera durante su ejecución.

Caso de uso habilitar usuarios para administración: es necesario registrar a usuarios que tengan acceso a información y a procesos de más alto nivel.

Caso de uso habilitar usuarios de consulta: se registra a usuarios que tienen acceso a la información ya procesada.

Caso de uso habilitar funcionario: es importante registrar a los funcionarios ya que son los que brindan información de estado de los trámites.

B) REQUERIMIENTO NO FUNCIONAL

Los atributos o cualidades que se requieren del sistema son: facilidad de uso, tolerancia de las fallas, tiempo de respuesta, metáforas de interfaz y plataformas. Estos atributos pueden tener un conjunto de detalles y restricciones de fronteras. Ver la tabla 3.2

TABLA 3.2: Atributos del Sistema

Fuente: [Elaboración propia]

Atributo	Detalles y restricciones de fronteras
Tiempo de respuesta	Restricciones de frontera: cuando se realice alguna operación sobre los registros del sistema, la respuesta será de 1 segundo.
	Detalle: grafico, colorido, basa en formularios y cuadros de diálogo.
Tolerancia a fallas	Restricción de frontera: debe registrarse toda la información en un plazo de 12 horas, aun cuando se produzca fallas de energía o del equipo.
Plataforma del Sistema Operativo	Detalle: Windows XP.

3.2.2 FASE DE ELABORACION

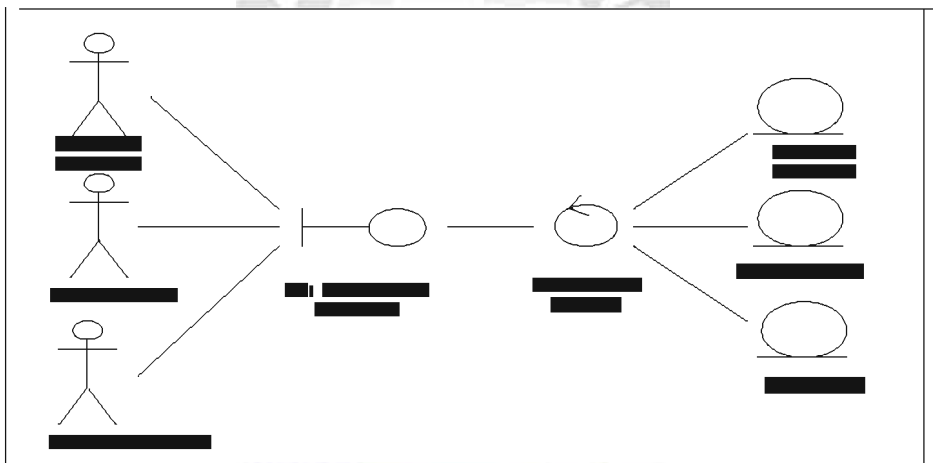
A continuación se detalla en diagramas de casos de uso, colaboración, secuencias y estados los procesos más representativos. El resto de los casos de uso se documentara en diferentes Anexos.

Casos de uso: Acceso al sistema

El acceso al sistema tiene como primer paso el interfaz de acceso al sistema y se conecta con el objeto de control del gestor de tareas, el objeto de tareas ayuda a la clase Usuario Externo a observar el seguimiento del trámite solicitado, de igual manera ayudara a la clase Funcionarios a recabar datos de los formularios, también ayuda a la clase administrativo a poder desempeñar tareas dentro del sistema, la clase reportes especifica informes finales y/o parciales del seguimiento de tramites. Ver figura 3.5

Figura 3.5.- Casos de Uso: Acceso al Sistema

Fuente: [Elaboración Propia]

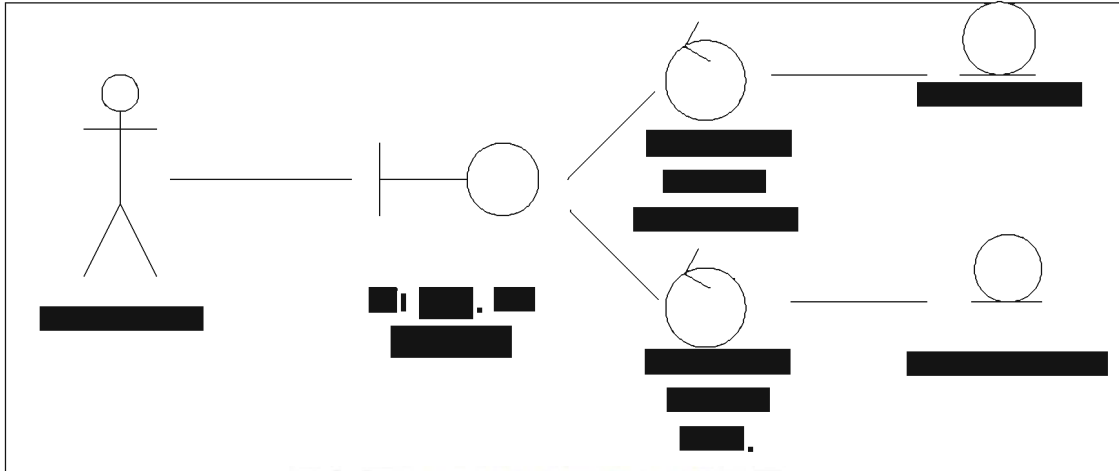


Casos de uso: Registro del Personal Autorizado

La realización de los dos casos de uso, registro de nuevo usuario, registro de nuevo administrador tiene como interfaz al registro del personal, el objeto gestor del usuario Funcionario y gestor del usuario Administrativo pide datos para registrar al Funcionario y al administrador mediante el interfaz del registro del personal y así poder almacenar estos datos a la clase Funcionarios y a la clase administrador como al nuevo usuario administrativo ya que puede habilitarle para consultas al sistema, ver figura 3.6.

Figura 3.6.- Casos de Uso: Registro del personal Autorizado.

Fuente: [Elaboración Propia]

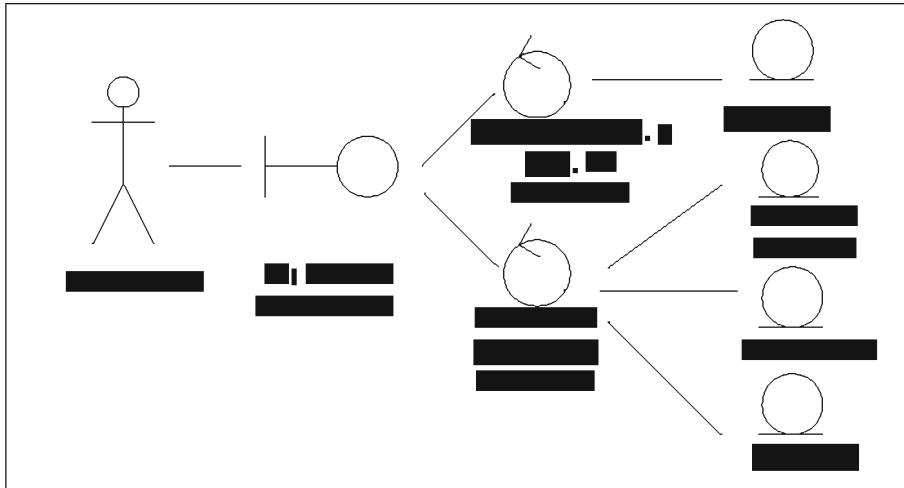


Casos de uso: Registro y Seguimiento, Planificación de Periodo de los formularios

La realización de los casos de uso, registro y seguimiento, planificación de periodos tiene como interfaz al Funcionario, el objeto gestor de registro y seguimiento de formulario, pide a la clase reportes registrar a todos los formularios de tramite parciales o ya terminados que pasan por las unidades respectivas, gestor del periodo de formulario, pide a la clase alerta de tramites a especificar el tiempo de permanencia de los formularios de trámite en una unidad respectiva , la clase de indicadores con los datos recabados por los formularios, la clase reportes mostrara los informes de los formularios registrados, ver figura 3.7.

Figura 3.7.- Casos de Uso registro y seguimiento, periodo

Fuente: [Elaboración Propia]

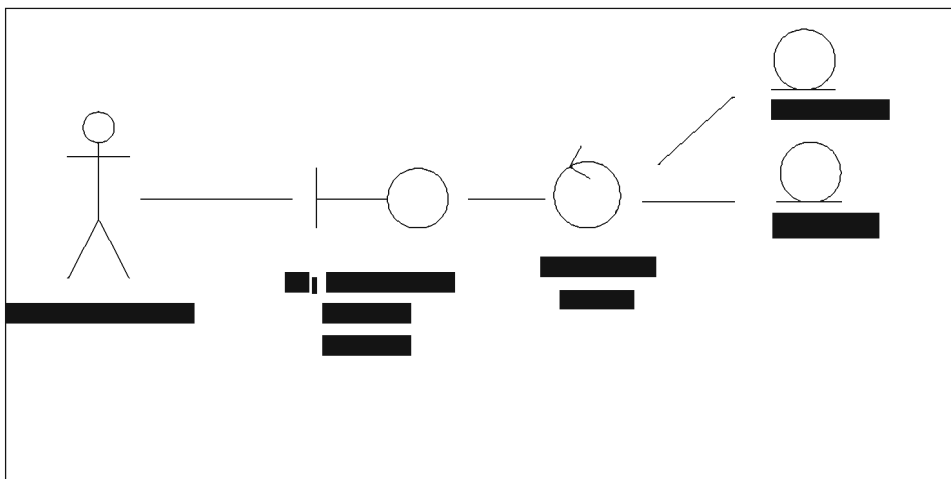


Casos de uso: Solicitud de formulario por el usuario externo

La solicitud de los formularios al sistema tiene como primer paso el interfaz de acceso del usuario externo y se conecta con el objeto de control gestor de tareas, la clase formulario pide al gestor de tareas consultas y reportes que ayuda a detallar el estado del tramite, ver figura 3.8.

Figura 3.8.- Casos de Uso: Solicitud de Formulario

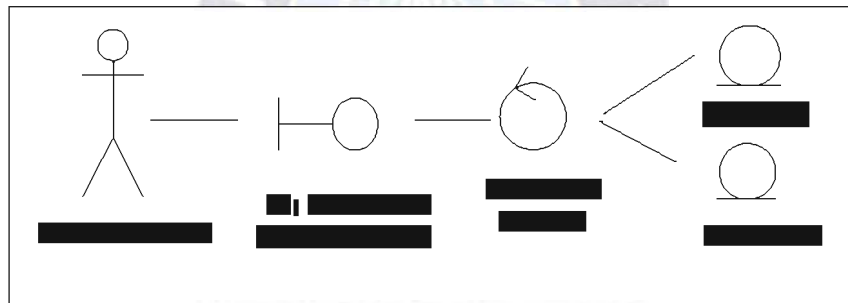
Fuente: [Elaboración Propia]



Casos de uso: Reportes de Seguimiento de Tramites

Los reportes de Seguimiento de Tramites tiene como primer paso el interfaz de acceso Administración, que se conecta con el objeto de control gestor de acceso, la clase Consultas de Seguimiento de cada tramite , la clase Reportes ayudara a detallar el estado de Seguimiento de cada tramite revisado, ver figura 3.9

Figura 3.9.- Casos de Uso: Reportes de Seguimiento de Tramites

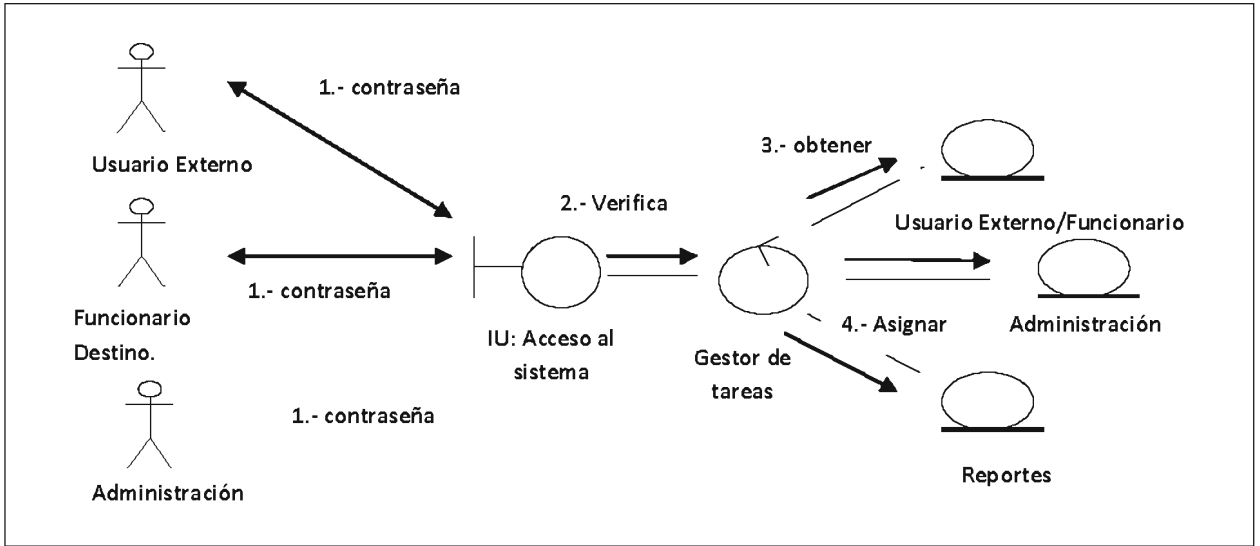


Fuente: [Elaboración Propia]

3.2.2.1 DIAGRAMA DE COLABORACIÓN

Hacemos uso de los diagramas de colaboración para describir las realizaciones de cada Casos de Uso ya que el objetivo fundamental es identificar los requisitos y responsabilidades sobre los objetos encontrados. El nombre de un mensaje debería denotar el propósito del objeto invocante en la interacción con el objeto invocado. A continuación presentamos los diagramas de colaboración para el sistema Ver figura 3.10.

Figura 3.10.- Diagrama de Colaboración: Acceso al sistema
Fuente: [Elaboración Propia]



3.2.2.2 DIAGRAMA DE SECUENCIA

Figura 3.11.- Diagrama de secuencia: verificar el acceso del personal autorizado al sistema
Fuente: [Elaboración Propia]

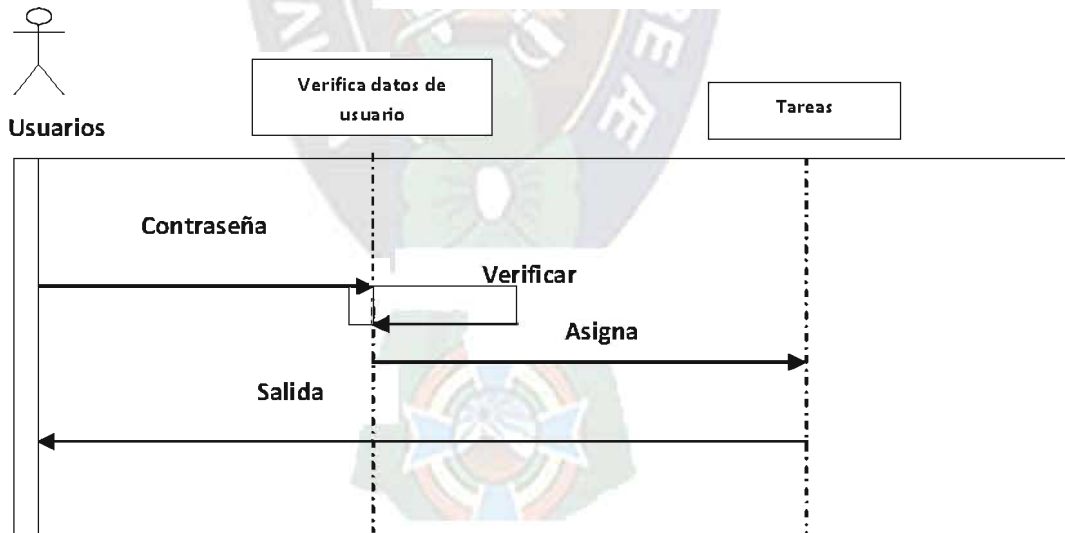


Figura 3.12: Diagrama de secuencia: registro de formularios del funcionario

Fuente: [Elaboración Propia]

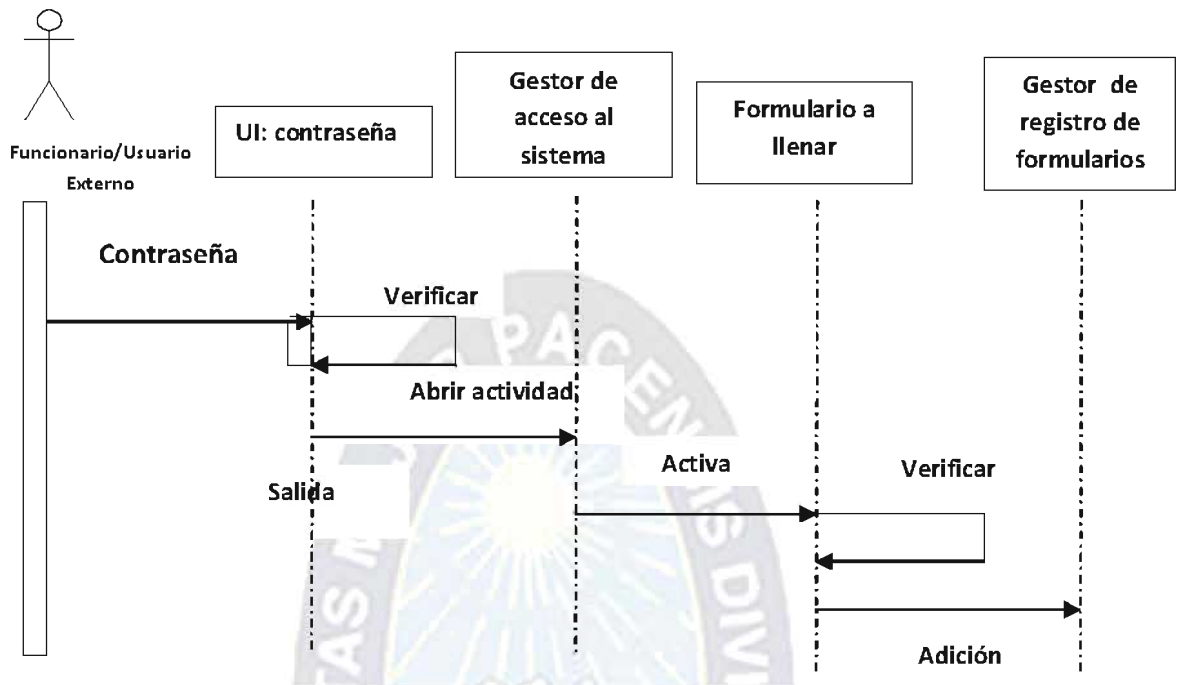


Figura 3.13 Diagrama de secuencia: Registro y Procesos de los datos del formulario

Fuente: [Elaboración Propia]

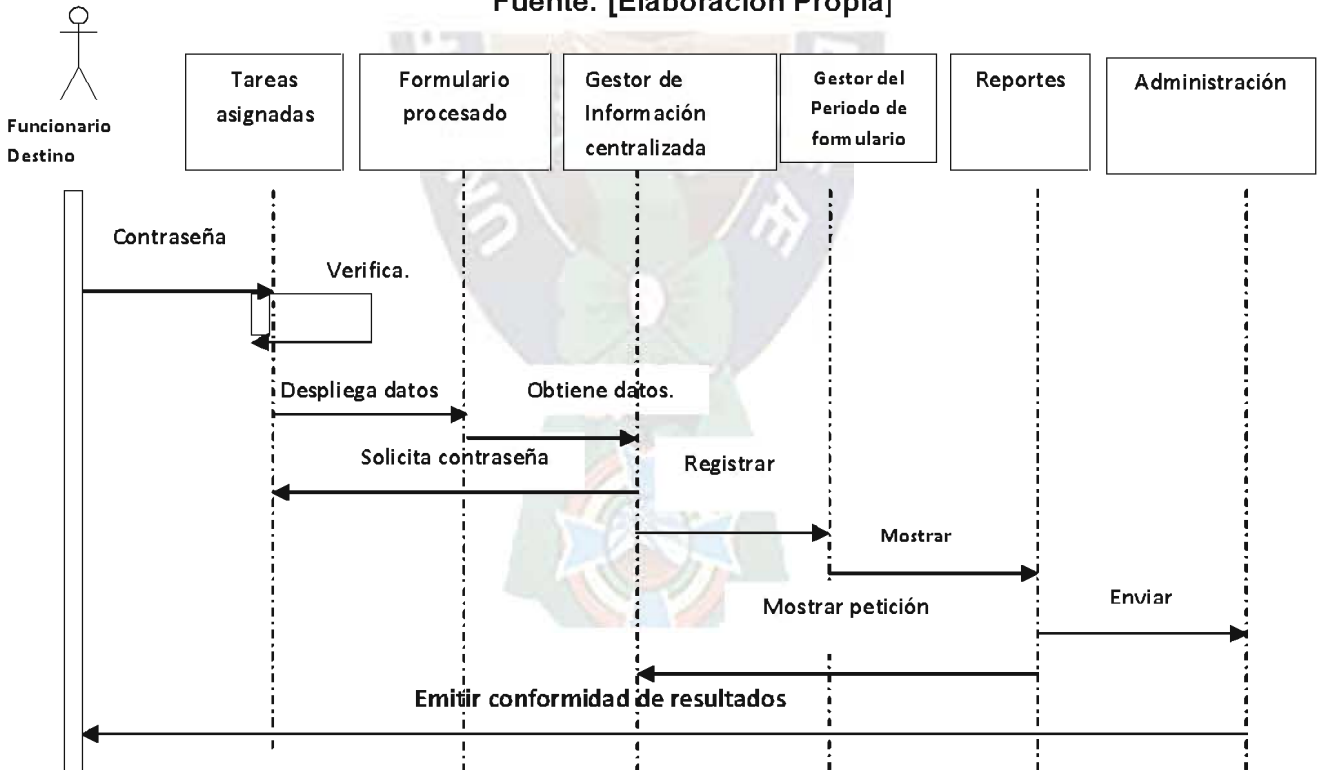
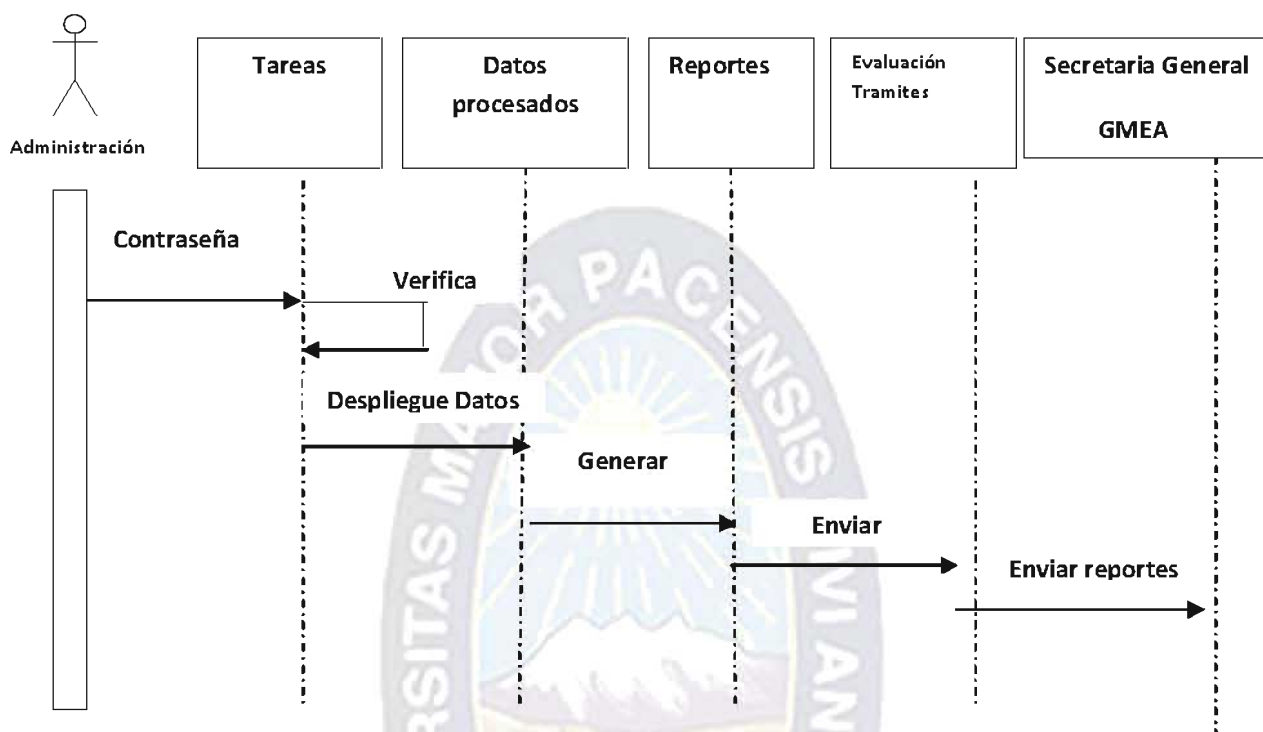


Figura 3.14 Diagrama de secuencia: Reportes de formularios finales

Fuente: [Elaboración Propia]

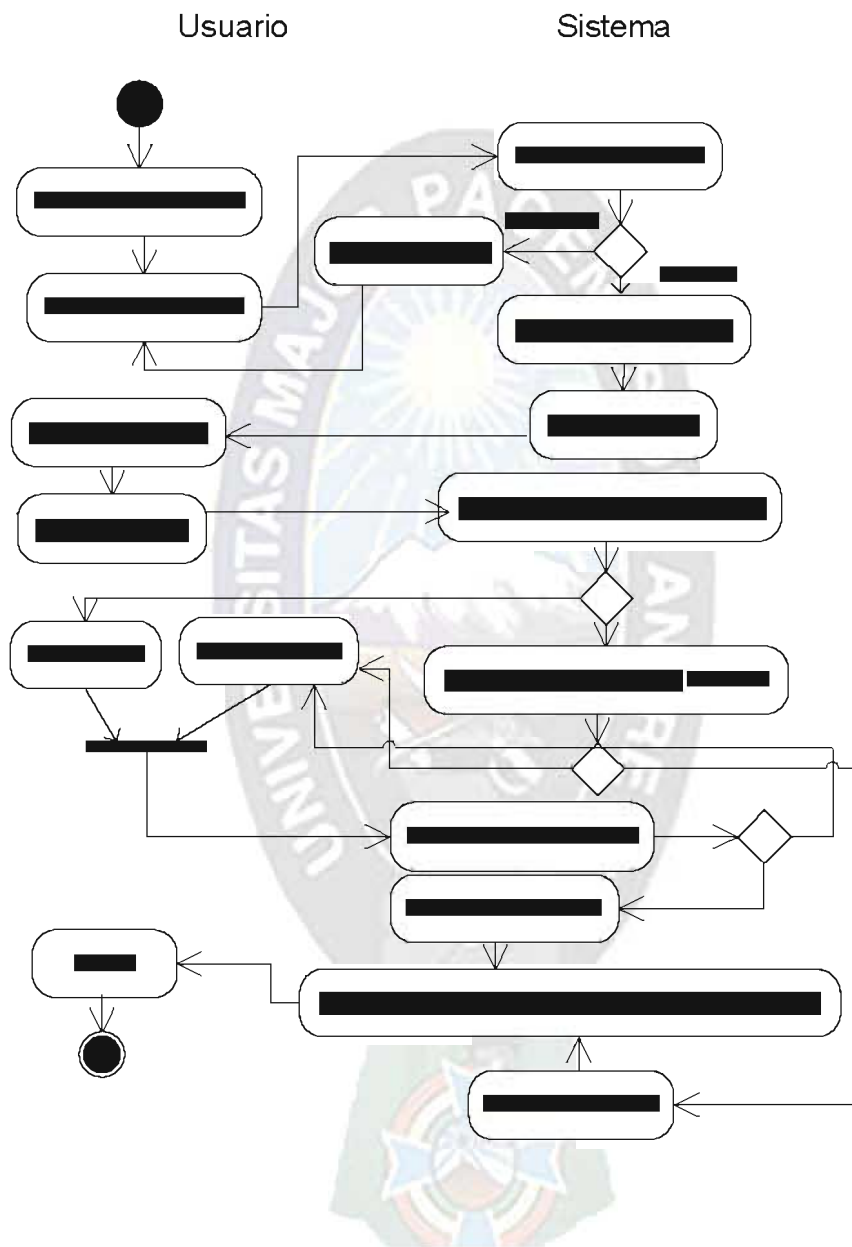


3.2.2.3 DIAGRAMA DE ESTADOS

Algunos objetos del diseño son estados controlados, lo que significa que sus estados determinan su comportamiento cuando reciben un mensaje. En estos casos es significativa la utilización de diagramas de estados para describir las diferentes transiciones de estados de un objeto de diseño. En nuestro caso describiremos cada diagrama de estado en función de una realización de caso de uso y la correspondiente clase mas critica en tal realización, ver figura 3.15, con más detalle podemos Ver en [Anexo G].

Figura 3.15.- Diagrama de estados: Registro y seguimiento de formulario

Fuente [Elaboración propia]



3.2.2.4 DIAGRAMA DE CLASES DE DISEÑO

Las clases identificadas por los casos de uso de análisis se muestran en la figura 3.16, este describe el conjunto de objetos con atributos, operaciones, métodos, relaciones y comportamientos.

Figura 3.16.- Diagrama de Clases de Diseño del sistema

Fuente [Elaboración propia]

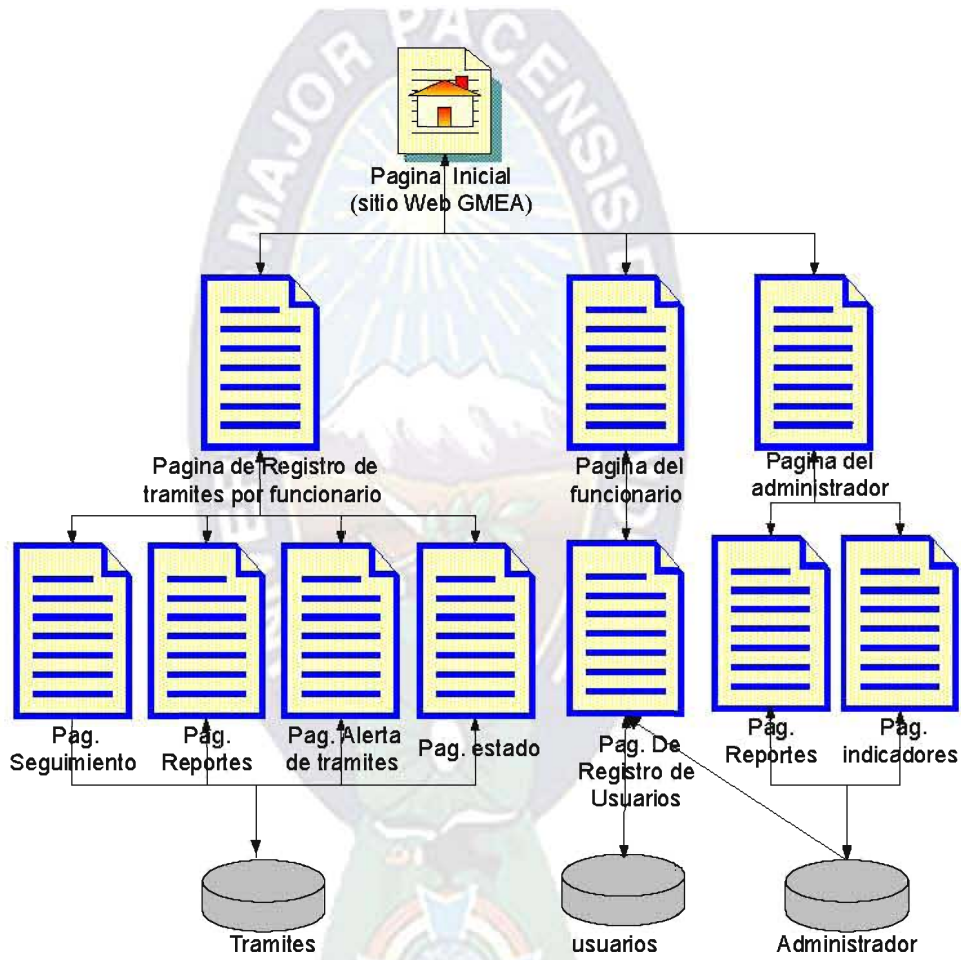


3.2.2.5 ESTRUCTURA JERARQUICAS

A continuación se muestra un diseño de la posible estructura jerárquica de las aplicaciones que se construirán mas adelante, ver figura 3.17, (Se muestra la ramificación o conexión que existe en el hipertexto).

Figura 3.17: Estructura Jerárquica del Sitio Web

Fuente: [Elaboración propia]

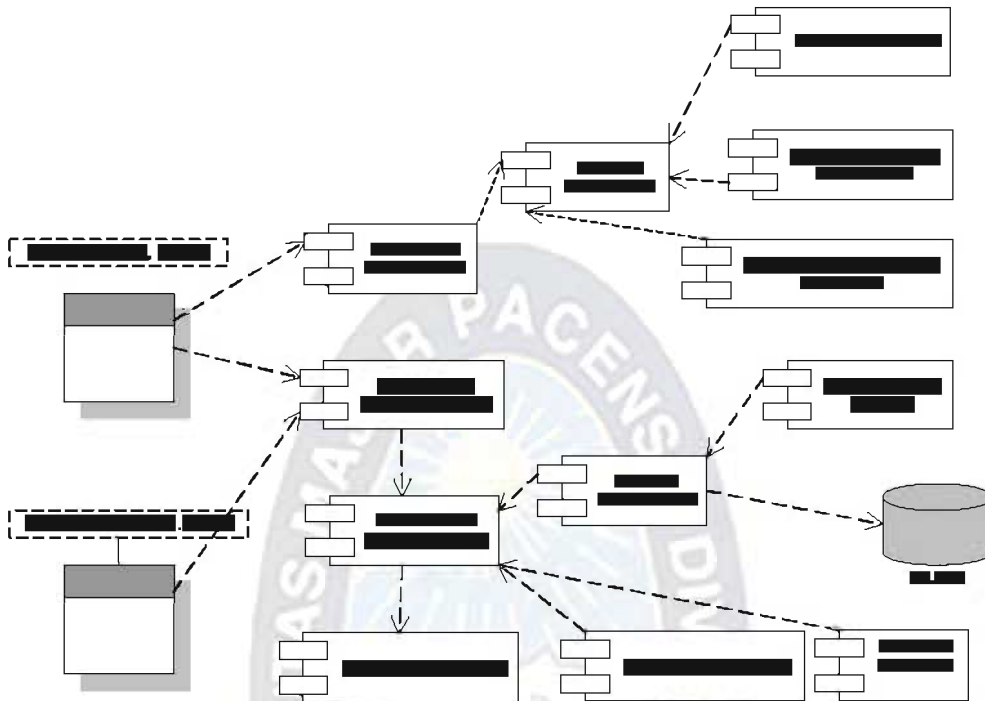


3.2.2.6 DIAGRAMA DE COMPONENTES

Teniendo en cuenta sus características, en la siguiente figura 3.18, se muestra el diagrama de componentes de nuestro sitio Web.

Figura 3.18: Diagrama de Componentes del Sitio Web

Fuente: [Elaboración propia]

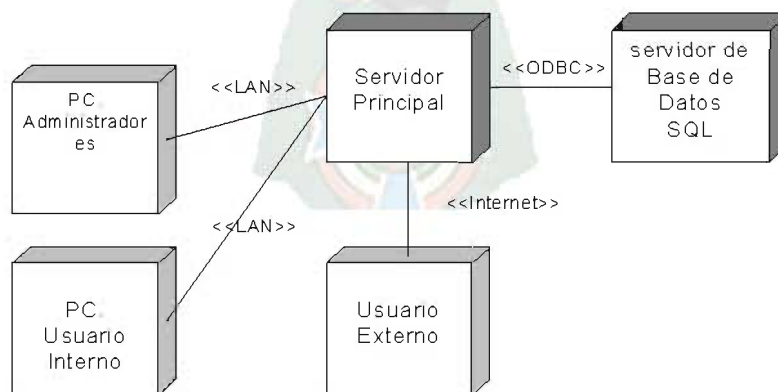


3.2.2.7 DIAGRAMA DE DESPLIEGUE

En este caso los nodos están representando a computadoras, que pueden ser además del servidor principal, administradores y de usuarios que pueden ser cercanos (LAN) o remotos (de Internet), ver figura 3.19.

Figura 3.19: Diagrama de Despliegue del Sitio Web

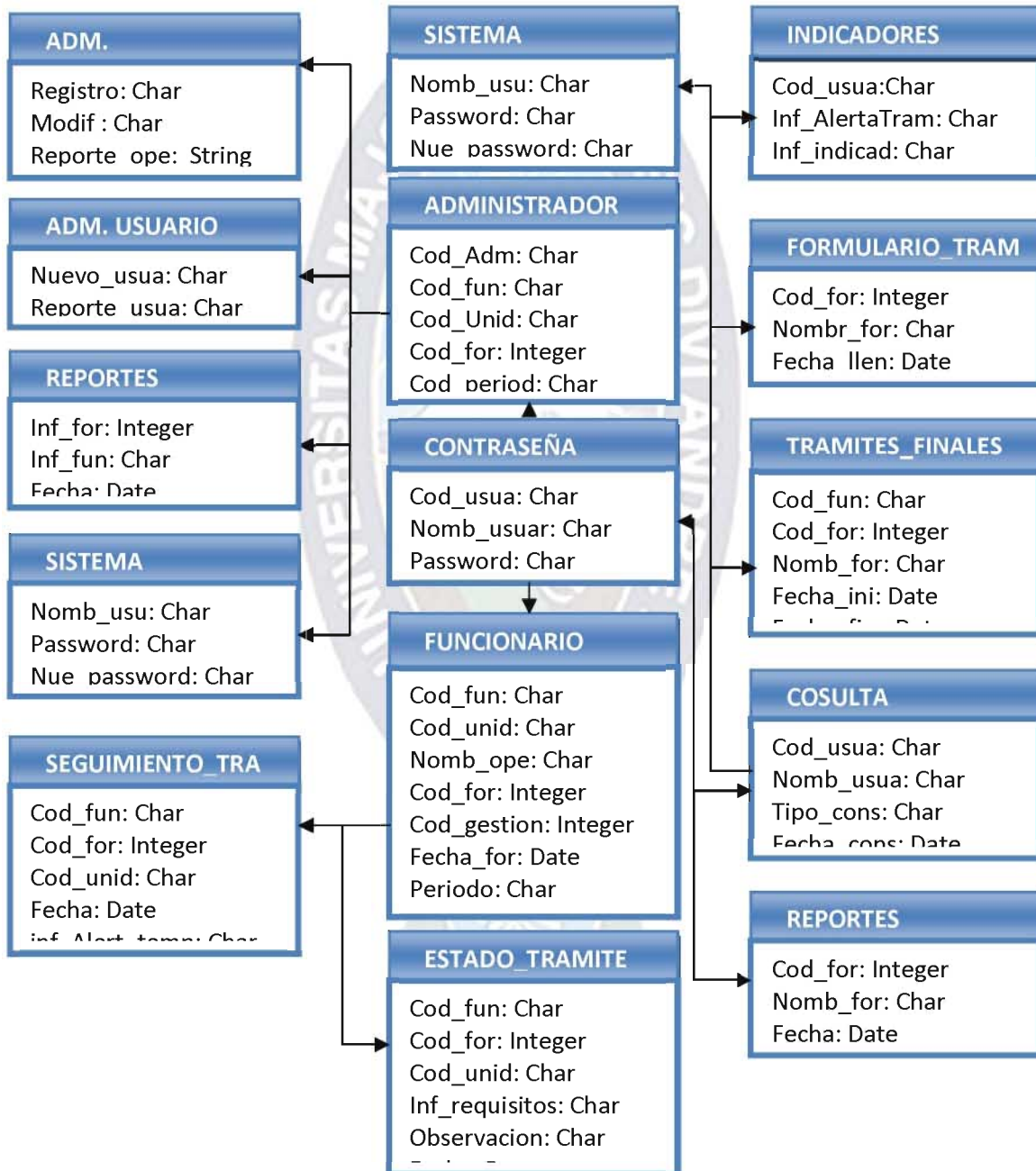
Fuente: [Elaboración propia]



3.2.2.8 DIAGRAMA NAVEGACIONAL

Aquí mostraremos la aplicación del modelo navegacional de la metodología OOHDM donde se reorganizo la información para adaptar a las necesidades de los usuarios del sistema, ver figura 3.20.

Figura 3.20: Esquema de clase Navegacional del Sitio Web
Fuente: [Elaboración propia]

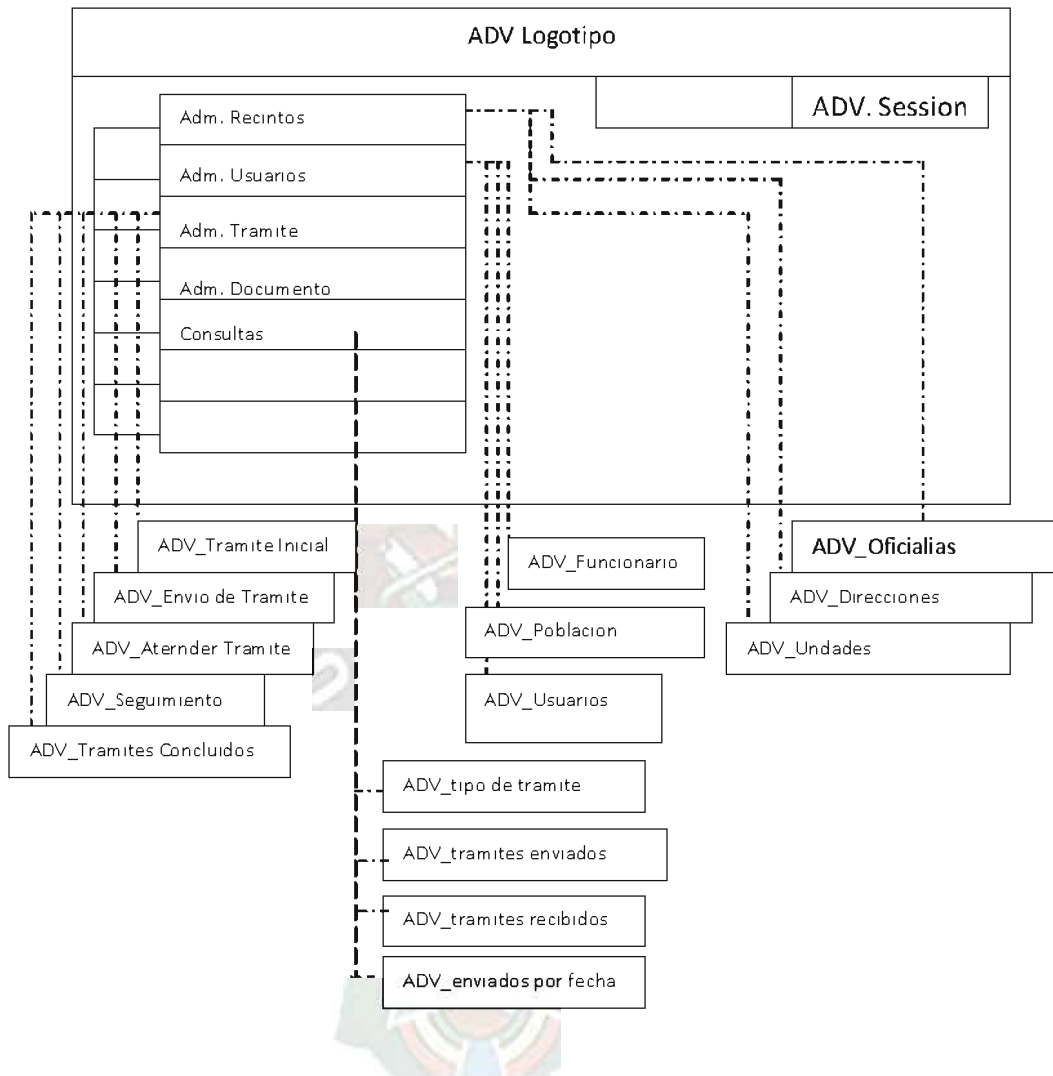


3.2.2.9 DIAGRAMA DE INTERFAZ ABSTRACTA

Para describir la interfaz del usuario de la aplicación en OOHDM se utiliza el modelo Vista de Datos Abstractos (ADV) que especifica la organización y el comportamiento de la interfaz. Los ADVs se desarrollan para cada uno de los usuarios que utilizará el sistema, ver figura 3.21.

Figura 3.21: ADV Administrador del sistema de información.

Fuente: [Elaboración propia]



3.2.2.10 INTERFAZ DE USUARIO

El Sistema de Información y Seguimiento de Trámites vía Web está disponible para ser accedido desde internet, en este sentido, también se puede ingresar desde cualquier equipo conectado en la intranet de la institución, ver figura 3.22

Figura 3.22 Interfaz de usuario: Ingreso al Sistema



Figura 3.23 Interfaz de usuario: Registro del Usuario Externo

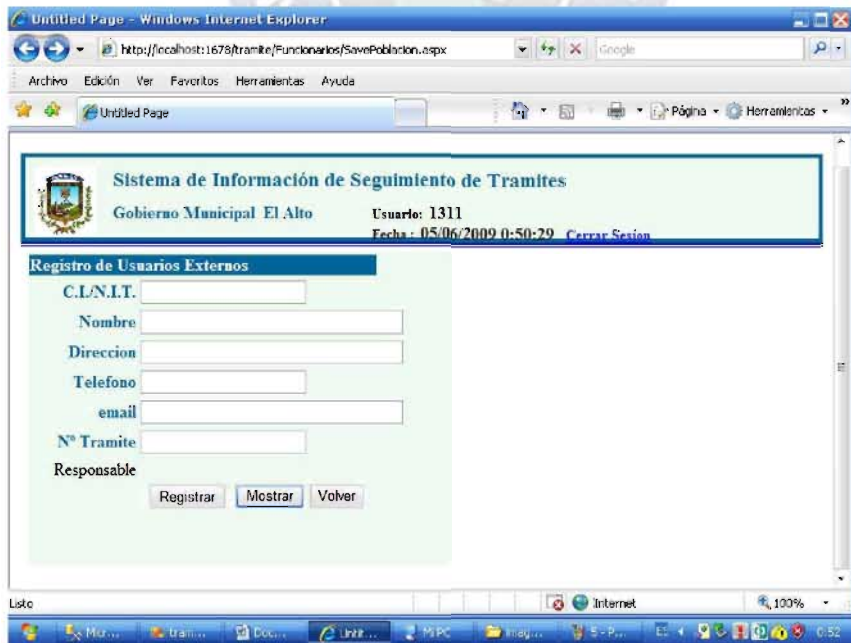
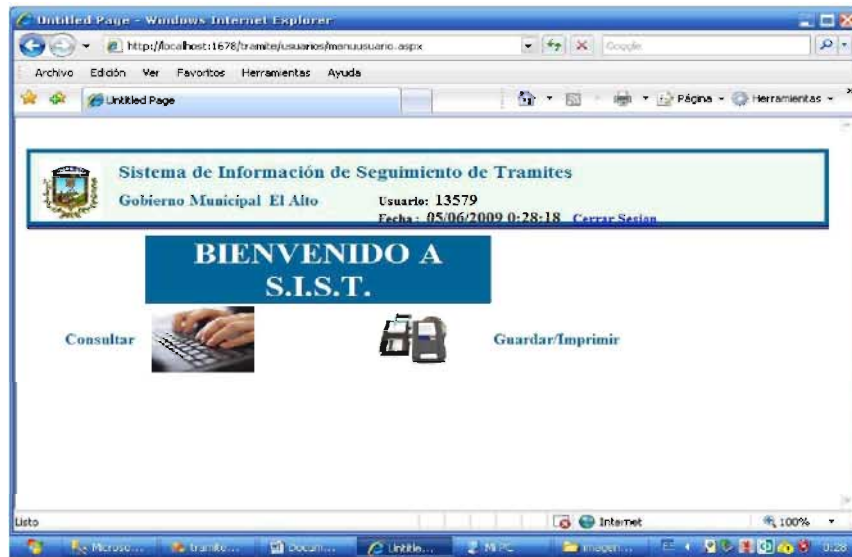


Figura 3.24 Interfaz de usuario: Consulta de trámite por el Usuario

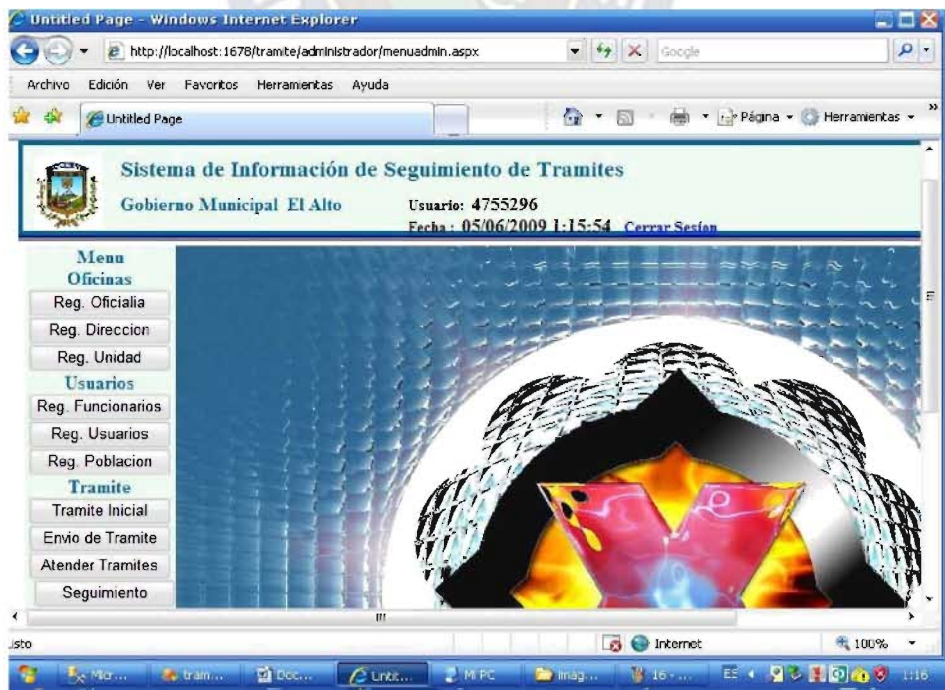


3.2.3 FASE DE CONSTRUCCION

MODULO DE ADMINISTRACION DEL SISTEMA

El ingreso a este modulo se lo hace vía intranet, y esta destinado a usuarios de Administración y jefes de cada Dirección o Unidades.

Figura 3.25 Modulo de Administradores



MODULO DEL FUNCIONARIO

Este modulo esta disponible desde internet , y esta destinado solo para funcionarios “Secretarias”.

a) Pantalla Principal

Es la primera pantalla que aparece una vez autenticado el usuario , y en ella se despliega los diferentes tipos de módulos , ver figura 3.26.

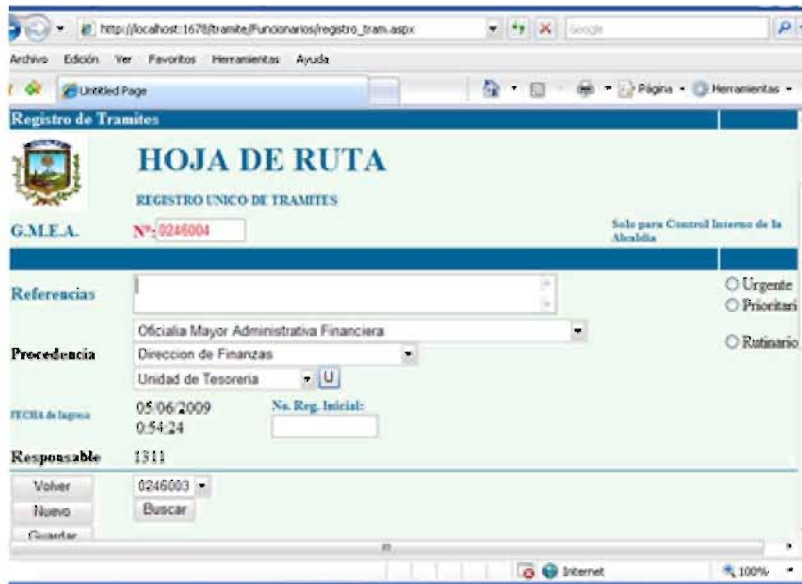
Figura 3.26 Modulo de Funcionarios



b) Modulo Tramite Inicial

Aquí se procede al llenado de la cabecera del formulario de la hoja de Ruta por primera vez, ver figura 3.27

Figura 3.27 Modulo Registro de Tramite Inicial



c) Modulo de Seguimiento

Muestra el seguimiento de cada trámite la ubicación exacta y el detalle de cada trámite. Ver figura 3.28

Figura 3.28 Modulo Seguimiento de Tramite



Sistema de Información de Seguimiento de Trámites

Gobierno Municipal El Alto

Usuario: 1311

Fecha: 05/06/2009 1:04:40 [Cerrar Sesión](#)

SEGUIMIENTO DE TRAMITE

NºTramite	Descripción	Procedencia	Fecha_Salida	Destino	Proveido
0246003	Datos de la balanza mal estado	Unidad de Tesoreria	02/06/2009 1:05:43	Unidad de Tesoreria	XXXXXXXXXXXXXXXXXX
0246003	Datos de la balanza mal estado	Unidad de Tesoreria	02/06/2009 1:08:42	Unidad de Credito Publico	ZZZZZZZZZZZZZZZZZZ

Modulo de Reportes

Aquí se muestra reportes obtenidos por el seguimiento de cada trámite.
Ver figura 3.29

Figura 3.29 Modulo Reportes de Seguimiento de Tramite



NroTramite	Descripción	Unidad de Origen	Unidad Destino	Observaciones
0246003	Datos de la balanza mal estado Unidad de Tesorería archivador 02/05/2009 1:05:43	Selena Cancari Guarachi Unidad de Tesorería	Renan Cancari Guarachi Unidad de Tesorería	yyyyyyyyy
0246003	Datos de la balanza mal estado Unidad de Tesorería carta 02/05/2009 1:08:42	Selena Cancari Guarachi Unidad de Tesorería	Dario Cancari Quispe Unidad de Credito Publico	ooooooooo

3.2.4 FASE DE TRANSICION

En la fase de transición llegan a cumplir los requisitos establecidos en las anteriores hasta la satisfacción de todos los usuarios. La actividad del Seguimiento de Tramites, comprende la descripción de los requerimientos técnicos mínimos para el funcionamiento del sistema, así como la modalidad de implementación que involucra la programación de los módulos que se ha organizado y especificado.

La implementación y puesta en marcha el Sistema de Información y Seguimiento de Tramites Vía Web, dependerá del cumplimiento de los siguientes procesos:

REQUERIMIENTOS DE HARDWARE

Se detalla los requerimientos mínimos de hardware, necesarios para el funcionamiento del sistema, estos se describen a continuación:

Servidor de aplicaciones y base de datos:

- HP ProLiant ML350 G4p, Dual Option SCSI - Tower Server
- Intel® Xeon™ 3.40GHz/800MHz, 2M Processor (up to 2 supported)
- Hard Drive 72.8-GB Pluggable Ultra320 SCSI 15,000 rpm
- 512MB REG PC2-3200 (1x512) memory

REQUERIMIENTOS DE SOFTWARE

En cuanto a software se describen los siguientes requerimientos:

- Sistema operativo Windows 2000 Server.
- Servidor de paginas Internet Information Server (IIS)
- Gestor de base de datos SQL Server 2005
- Lenguaje de desarrollo de aplicaciones ASP.NET con Visual Studio.NET 2003.

SEGURIDAD DE LA INFORMACIÓN

Los riesgos fundamentales asociados con la incorrecta protección de la información son:

Revelación a personas no autorizadas, inexactitud de los datos y la inaccesibilidad de la información cuando se necesita.

A su vez, estos aspectos se relacionan con las tres características que debe cubrir un sistema de información seguro: confidencialidad, integridad y disponibilidad.

Confidencialidad

Para ello se definen niveles de usuario, quienes son lo únicos usuarios autorizados para el acceso a la información, teniendo cada nivel de usuario diferentes roles de acuerdo a su grado o nivel de autoridad, ver tabla 3.3.

Tabla 3.3. Niveles de usuario

Fuente: Elaboración propia

1	Funcionario
2	Jefe u Administrador
3	Usuario Externo

Integridad

En este sentido, se han definido roles específicos para cada nivel de usuario, lo cual garantiza que solo el personal autorizado a su nivel o grado de autoridad puedan alterar o no la información, ver tabla 3.4.

Tabla 3.4. Roles de usuario

Fuente: Elaboración propia

Usuarios	Roles
Usuario Externo	<ul style="list-style-type: none">• Consulta de Seguimiento de su Tramite• Reportes referente a su estado de su tramite
Funcionario	<ul style="list-style-type: none">• Registro de Tramites• Administración de usuario externos• Informes• Reportes• Consultas• Indicadores
Jefe ,Dirección, Unidad	<ul style="list-style-type: none">• Informes• Reportes• Administración de Funcionarios• Administración y Seguimientos de Tramites• Indicadores

Disponibilidad

Para esto se implanta el uso de variables de sesión, los cuales permiten que después de la autenticación del usuario y una vez autorizado el ingreso al sistema, este tiene acceso a la información sin restricciones, sin embargo, una vez que no se realiza ninguna transacción durante un periodo de tiempo determinado, automáticamente se niega la estadía en el sistema.

Existen otros aspectos o características de la seguridad que pueden, en su mayor parte, incluirse o similares ha uno de los aspectos fundamentales:

Autenticación

Esta propiedad permita asegurar el origen de la información, es decir, la identidad del emisor puede ser validada.

La identidad del usuario es verificada mediante la validación de un nombre de usuario y una contraseña ingresada al momento del login. El sistema verifica la identidad de usuario y la utiliza para determinar qué operaciones puede realizar, enviándolo a la interfaz respectiva, ver figura 3.30.

Fig. 3.30 Control de acceso al sistema

Fuente: Elaboración propia



The image shows a login form with a blue header bar containing the text "Control de acceso". Below the header, there are two input fields: "Nombre de Usuario:" and "Contraseña:", each preceded by a blue circular icon with a white key symbol. Below these fields is a yellow button with the text "Ingresar".

Auditoria

Es la capacidad de determinar que acciones o procesos se han llevado en el sistema, y quien y cuando las han llevado a cabo. Para esto se mantiene un registro de las actividades del sistema de manera parcial, creando estructuras específicas para esta tarea.



CAPITULO 4

CALIDAD DE SOFTWARE

4. CALIDAD DEL SOFTWARE

4.1 INTRODUCCIÓN

La evaluación de productos Web no es una tarea sencilla. Es difícil considerar todas las características y atributos deseables y obligatorios de una aplicación o sitio Web si no se cuenta con un modelo de calidad que permita especificar ordenadamente dichas características y atributos. Se pueden destacar varios trabajos que tratan de definir modelos y criterios de calidad para productos de software. Algunos de ellos podemos destacar como el ISO 9126, IEEE1061.

4.2 FUNCIONALIDAD

Se cuantifica el tamaño y la complejidad del sistema en términos de las funciones de usuario, puede ser valorado mediante una medida denominada punto función. Determinaremos las cinco características del dominio de información, tomando en cuenta su cantidad.

Tabla de 4.1 cálculos del punto de función

Fuente: [Elaboración propia]

Parámetro de medición	Factor de ponderación			
	Cuenta	*	Simple	Total
Numero de entradas de usuarios	7	*	3	= 21
Numero de salidas de usuarios	3	*	4	= 12
Numero de consultas de usuarios	2	*	3	= 6
Numero de archivos	1	*	7	= 7
Numero de interfaces externas	3	*	5	= 15
Cuenta total sin ajustar				61

Donde el punto de función esta dado por la formula.

$$PF = \text{cuenta total} * (0.65 + 0.01 * \text{SUM}(Fi)) \quad (1)$$

Donde $(0.65 + 0.001 * \sum F_i)$ es el ajuste de la complejidad respecto a la cuenta total y F_i ($i = 1$ a 14) son ajustes de complejidad según el factor cuyo valor puede ser de 0 a 5.

Además que 0.65 es el valor mínimo de ajuste respecto a la cuenta total y 0.01 es el factor de conversión:

La tabla de valores de ajuste de complejidad es mostrada en la tabla 4.2

Tabla 4.2 Valores de ajuste de complejidad

Fuente: [PRE02]

1	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	5
2	¿Se ha diseñado el código para ser utilizable?	5
3	¿Existen funciones de procesamiento distribuido?	1
4	¿Se ejecutara el sistema en un entorno operativo existente o fuertemente utilizado?	4
5	¿Requiere el sistema entrada de datos iterativos?	5
6	¿Requiere la entrada de datos interactivo que las transacciones de entradas de datos se lleven acabo sobre múltiples pantallas y operaciones?	3
7	¿Es crítico el rendimiento?	3
8	¿Se utilizan los archivos maestros de forma interactiva?	4
9	¿Son complejos las entradas, salidas, los archivos o las peticiones?	1
10	¿Es complejo el procesamiento interno?	2
11	¿Se requiere comunicación de datos?	2
12	¿Están incluidas en el diseño la conversión y la instalación?	3
13	¿Requiere el sistema copias de seguridad?	5
14	¿Se ha diseñado las aplicaciones para facilitar los cambios y para ser fácilmente utilizada por el usuario?	5
	SUM F_i	48

Pf: medida de funciones concedida a la aplicaron como valor de Normalización

Cuenta_total: Suma de las entradas obtenidas, de acuerdo al nivel de complejidad del sistema respecto a los usuarios.

$(0.65 + 0.01 * \text{SUM}(F_i))$: Ajuste de complejidad

0.65: Valor mínimo de ajuste de complejidad

0.01: factor de conversión, asumiendo un error de 1%

F_i : Valores de ajuste de complejidad

Entonces, reemplazando en (1) se tiene:

$$PF = 61 * (0.65 + (0.01 * 48))$$

$$PF = 68.93pf$$

Para poder encontrar las líneas de código se aplica la equivalencia entre líneas de código y punto de función según Jones un valor de 20 [Ldc/pf]

$$LDC = PF * \text{equivalencia}$$

$$LDC = 68.93 [pf] * 20 [Ldc/pf]$$

$$LDC = 1378.6 [Ldc]$$

Considerando como máximo valor de ajuste de complejidad $\text{SUM}(F_i) = 70$, tiene

$$PF \text{ máximo} = 61 * (0.65 + (0.01 * 70))$$

$$PF \text{ máximo} = 82.35$$

Por tanto la funcionalidad esta dada por:

$$PF = \frac{68.93}{82.35} = 0.84$$

$$PF \text{ máximo} = 82.35$$

Entonces la funcionalidad del sistema es de un 84%, esto quiere decir que el sistema tiene un 84% que funcione sin riesgos a fallos y operatividad constante y un 16% de colapso del sistema.

4.3 CONFIABILIDAD

La confiabilidad nos ayuda a medir la cantidad de tiempo que el software esta disponible para su uso, para medir la madurez de los fallos.

Para calcular la confiabilidad del sistema, se toma en cuenta el periodo de tiempo en el cual se ejecuta, y se obtiene muestras.

$$F(t) = e^{-x P(t)} \quad (1)$$

- El inicio de ejecución del sistema se define en el instante $t_0=0$, lo que significa el tiempo inicial en el cual dará inicio el funcionamiento del sistema.

$$F(t_0) = e^{-x t_0}$$

$$F(0) = 1$$

- Se observa el trabajo del sistema hasta que se produce una falla en el instante T, el cual va aproximado a una variable aleatoria continua.

Como se aproxima a variables aleatorias continuas, la confiabilidad será obtenida en términos probabilísticas.

Entonces, el termino en el cual el sistema trabaja sin fallas esta dado por la ecuación (2) y el tiempo en el cual no falla el sistema esta dado por (3).

$$P(T \leq t) = F(t) \quad (2) ; F(t) \text{ probabilidad de fallas}$$

$$P(T \geq t) = 1 - F(t) \quad (3) ; 1 - F(t) \text{ probabilidad de trabajo sin fallas}$$

Dado en el tiempo inicial y otro final en el cual ocurre una falla, se utiliza la distribución exponencial.

$$F(t) = \begin{cases} 0.9e^{-x/10} & ; x > 0 \\ 0 & ; \text{e.o.c} \end{cases} \quad (4)$$

- En un periodo de 12 meses como tiempo de prueba se define de cada 10 ejecuciones una falla.
- Teniendo la funcionalidad del 84% del sistema calculamos para el periodo establecido.

Probabilidad de fallas $F(t) = P(T \leq t)$

$$P(T \leq t)$$

$$F(t) = 0.84 * e^{(-1/10)*t}$$

$$F(t) = 0.84 * e^{(-1/10)*12}$$

$$P(t) = 0.91 \text{ sin fallas}$$

Se tiene que para el periodo establecido de 12 meses de prueba se tendrá la seguridad de un 91% de periodos sin fallas.

4.4 MANTENIMIENTO

Cuando se aplica este tipo de métrica, se muestra la cantidad de esfuerzo, requerida para conservar el funcionamiento normal o para restablecerlo, una vez se representa un fallo. A continuación se establece el índice de madurez del software (IMS), que proporciona una indicación de la estabilidad de un producto de software basada en los cambios que ocurren con cada versión. Esta dado por:

M_t = Número de módulos en la versión actual

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

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

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

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

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

$$\text{IMS} = [69 - (1+9+0)]/69$$

$$\text{IMS} = 0.85$$

A medida que el IMS se aproxima a 1.0 el producto se empieza a estabilizar. Esto quiere decir que 0.85 es próximo a uno, se está mejorando.

4.5 USABILIDAD

Es muy difícil conocer con exactitud una medida cuantitativa acerca del grado de usabilidad, por ser este un atributo muy subjetivo.

Una de las alternativas para conocer el grado de usabilidad en el sistema, es la aplicación de cuestionarios especializados para este fin. En este sentido se toma en cuenta la métrica denominada "Esfuerzo de usabilidad" [Granollers], que indica en que medida se han utilizado recursos y se han desarrollado actividades con la finalidad de alcanzar un grado de usabilidad.

Existen parámetros que se adecuan a cualquier tipo de software de aplicación, la tabla 4.3 muestra estos parámetros.

Tabla 4.3 Detalle de parámetros

Fuente: [USE02]

Nombre del parámetro	Valor del parámetro	Resultado función
Fase del proyecto: A1	R: Análisis	80
	D: Diseño	80
	I: Implantación	80
	L: Despliegue	100
Numero de usuarios: A2	1 a 5	80
	6 a 9	100
	Mayor a 9	80
Numero de implicados: A3	1 a 3	40
	4 a 6	65
	7 a 10	70
	Mayor a 10	70
Numero de evaluadores: A4	1 a 3	40
	4 a 6	65
	7 a 10	70
	Mayor a 10	10
Compleitud: C	%	
Prototipo: A4	Papel	75
	Pizarra	50
	Maqueta	75
	Escenario	75
	Video	100
	Pantallas	100

Se ha realizado una evaluación a cada una de las fases importantes de desarrollo, para conocer el grado en que se ha dedicado esfuerzos para lograr el objetivo. Ver tabla 4.4

Tabla 4.4 Evaluación de fases

Fuente: [Elaboración propia]

Nombre del parámetro	Fase de análisis	Fase de diseño	Fase de implantación	Despliegue
Fase del proyecto: A1	80	80	80	100
Numero de usuarios: A2	80	80	80	80
Numero de implicados: A3	65	65	70	70
Numero de evaluadores: A4	65	40	65	65
Compleitud: C	80%	80%	90%	100%
Prototipo: A4	100	75	100	100
Total	312	272	373,5	415

Resultado: 1375.5 = EU

Según la metodología aplicada, el rango aceptable esta comprendido entre 1200 y 1600, por lo tanto se puede concluir que la usabilidad del sistema es aceptable, esto quiere decir que es fácil de usar el sistema.

Para calcular la efectividad en el aprendizaje de los usuarios para manipular el sistema, se tiene la siguiente relación.

$$\%Efectividad = (\text{cantidad de tareas aprendidas}) / \text{cantidad de tareas}$$

Donde Cantidad de tareas por usuario es de: U1=12, U2=6, U3=9

Las principales tareas efectuadas en la unidad consisten en una serie de llenado de formularios, así como la generación de consultas y reportes por medio de parámetros de entrada.

$$U1: 11/12 = 0.91, \quad U2: 6/6 = 1, \quad U3: 7/9 = 0.78$$

$$Efectividad\ total = (0.91+1+0.78)/3 = 0.89 = 89\% \text{ de efectividad}$$

Por lo que el sistema tiene una usabilidad que es efectiva de un 89%. Que tiene una fácil comprensión y una facilidad de aprendizaje para los usuarios.



CAPITULO 5
CONCLUSIONES Y
RECOMENDACIONES

5. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- Se ha logrado diseñar y elaborar el Sitio Web requerido por el Gobierno Municipal El Alto aplicando el uso de herramientas existentes para su desarrollo y que fueron mencionados en la parte teórica.
- Se obtiene indicadores que reflejan la cantidad de tráfico de trámites realizados en diferentes unidades, permitiendo hacer un seguimiento de los mismos.
- La calidad del producto queda garantizada, cumpliendo con los principales aspectos de calidad de software.
- La información sobre los formularios de hoja de ruta quedan centralizadas, permitiendo su consolidación y difusión a distintos niveles de información.
- Se presenta reportes para un seguimiento garantizado de cada formulario, de trámite como también los registros de los funcionarios quienes son responsables en el llenado.
- Se reduce en gran medida el tiempo de recolección de información que es recabado por los formularios de trámites vía Web.
- Se logro una aplicación robusta, flexible, gracias al uso de la metodología RUP como metodología central y con ayuda de UML y OOHDM. Se observo que esta combinación puede obtener resultados muy buenos a la hora de pensar en el mantenimiento y ampliación de un proyecto de este tipo además se logro obtener una estructura Web que no presenta dificultad de navegación que, combinada con la teoría de usabilidad, el producto Web esta muy orientado al usuario.

5.2 Recomendaciones

- En la asignación de un responsable de sistemas, debe ser de preferencia un informático o técnico del área de sistemas, para que se encargue de la administración y mantenimiento del sitio Web.
- Realizar copias de seguridad diariamente para evitar la pérdida de la información.
- Se debe elaborar un Sistema de Alerta Temprana e Información Gerencial exclusivo para la Secretaría General para tener un control de la atención que realiza el funcionario de cada unidad.





BIBLIOGRAFÍA

BIBLIOGRAFIA

LEYES DECRETOS Y RESOLUCIONES

[POA08] Dr. FANOR NAVA SANTIESTEBAN, G (Agosto 2008)

Programa de Operación Anual 2008. Bolivia : **Honorable Concejo Municipal**.

[LPP94] SANCHEZ DE LOZADA . G (20 de Abril de 1994): *Ley de Participación Popular*. Bolivia: **Honorable Concejo**

Municipal

LIBROS

[PRE02] PRESSMAN, R. (2002): *Ingeniería del Software*. España: **Quinta Edición en español: editorial Fareso.S.A.**

[RUP05] GARCIA, F (15 de febrero de 2005): *Rational Unified Process(RUP)*. E.E.U.U: Quinta Edición.

[RUP04] IBARRA, A (1 de enero de 2004): *Rational Unified Process*. Chile: Cuarta Edición.

[RUP00] GRADY BOOCH, I (2000): *El Proceso Unificado de Desarrollo Software*. España: primera edición Pearson Educación.

[EAD05] VERNAZZA, L (1996): *Estado del Arte del desarrollo sistemático De Interfaces Gráficas de usuario*: México.

[MDA03] PEREZ SERRANO, C (DASGBD-2003/2004) : *Metodología de desarrollo de Aplicaciones Web*: España

[AAI05] FERNÁNDEZ CAMACHO DAVID : *Aplicaciones de Agentes Inteligentes Web*: España

[LAR99] LARMAN Craig. (1999): *UML y patrones*. Prentice Hall Hispanoamericana. S.A.

[JAC00] Jacobson Ivan. (2000): *Proceso Unificado de Desarrollo*. Edición. Wesley de Pearson Educación, S.A.

REFERENCIAS WEB

[OOH05] OOHDM (Object Oriented Hypermedia Design Method): Chile

<http://www.geocities.com/cocheli99/hipermedi/htp>.

[RUP06] Conceptos de diagramas de estados, secuencia, colaboración, etc. Bolivia

http://www.cs.ualberta.ca/~figuero/soo/uml/casos_uso01.html



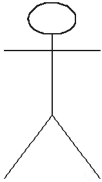

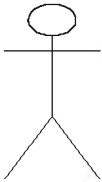


ANEXOS

ANEXO D

En esta fase inicial del proyecto, en lo referente a su desarrollo, centramos nuestra actividad en la creación y generación de los casos de uso con la colaboración de los futuros usuarios (actores), a continuación se describe el detalle de la identificación de los actores, ver tabla 1.

Tabla 1.- Actores relacionados con el sistema

ACTORES	DESCRIPCION
 USUARIO	Ingresa al sistema por medio de un nombre de usuario y contraseña asignada, el cual le permitirá acceder al formulario de tramites para su posterior llenado.
 ADMINISTRADOR	Usuario que verifica los informes finales de seguimiento de tramite. Tiene acceso a Reportes e Informes de Indicadores de tramite.
 FUNCIONARIO	Encargado de hacer el seguimiento de tramites y llenado de formularios. Manejo y verificación de datos a procesar

ANEXO E

DESCRIPCION DE LOS CASOS DE USO

En las siguientes tablas se describen cada uno de los casos de uso generados por el diagrama de casos de uso del negocio, ver tabla 2.

Tabla 2: Registro del Usuario	Elaborado por: Renan Cancari Guarachi
Nombre:	Registro del Usuario
Actor:	Usuario, Funcionario
Descripción:	El usuario tiene que registrarse en la institución con sus documentos, donde detalla a que unidad representa, todo esto se realiza con la revisión del funcionario, ya que esta persona da un informe respectivo a sus superiores.
Flujo Principal:	Eventos
	<ul style="list-style-type: none"> • Revisión de la documentación prestada por los usuarios a así la unidad de la institución. • Periodo de tiempo para la revisión • Dara el visto bueno o no, el funcionario al usuario
Precondición:	La documentación prestada por el Usuario tiene que estar bajo los reglamentos de la Institución GMEA.
Pos condición:	El funcionario entregara la respuesta en el tiempo mas corto posible.
Presunción:	Usuario tiene que esperar.

Tabla 3: Registro y Seguimiento de formularios de tramite	Elaborado por: Renan Cancari Guarachi
Nombre:	Registro y Seguimiento de formularios
Actor:	Usuario, Funcionario
Descripción:	Llenado de formulario vías Web.
Flujo Principal:	Eventos
	<ul style="list-style-type: none"> • El Usuario busca en la pagina web del GMEA el formulario de tramites que se llenara . • La pagina Web expone información , donde le pedirá el código y contraseña para la obtención del formulario requerido. • El usuario llenara el formulario después de haber sido aceptado por el sistema,

	este formulario ya estará registrado, el funcionario realizara un seguimiento de todos los que llenen los formularios de acuerdo a normativa.
Precondición:	Que el usuario este registrado por la unidad respectiva de el GMEA.
Pos condición:	Si el llenado de formulario esta realizado de forma correcta pasara al registro de reportes.
Presunción:	Introducir código y contraseña del usuario de forma correcta.

Tabla 4: Verificación de datos del formulario.	Elaborado por: Renan Cancari Guarachi
Nombre:	Verificación de datos del formulario
Actor:	Funcionario, Administración
Descripción:	Se realiza la verificación de los datos llenados en el formulario por los operadores.
Flujo Principal:	Eventos
	<ul style="list-style-type: none"> • La información recabado de los formularios será analizado por el Funcionario quien realizara el seguimiento de los datos de la documentación. • La entrega de informes de funcionarios, referentes de los formularios será revisado por la Administración.
Precondición:	El informe tiene que ser lo mas claro posible.
Pos condición:	Todo esto dará paso a reportes finales.
Presunción:	Los datos recabados de los formularios tienen que estar aceptables.

Tabla 5: Elaboración de Reportes de seguimiento de tramites	Elaborado por: Renan Cancari Guarachi
Nombre:	Elaboración de Reportes de seguimiento de tramites
Actor:	Funcionario, Usuario, Administración
Descripción:	Generación de reportes e indicadores gracias a los formularios llenos que son obtenidos por los Usuarios
Flujo Principal:	Eventos
	<ul style="list-style-type: none"> • El funcionario realiza reportes e indicadores solicitados por la Administración. • Todo reporte puede ser obtenido mediante

	<p>la solicitud y permiso, sobre la información de los diferentes Usuarios.</p> <ul style="list-style-type: none"> • La información llegara a Administración de GMEA, a funcionarios y a Usuarios Externos que participen en el sistema de trámites.
Precondición:	Formularios Ordenados
Pos condición:	Si los reportes están realizados de forma correcta los resultados serán mostrados al publico de forma general, mediante indicadores.
Presunción:	Los resultados finales tendrán la autorización de las unidades respectivas y la secretaria general del GMEA.

Tabla 6: Planificación de formularios	Elaborado por: Renan Cancari Guarachi
Nombre:	Planificación de formularios
Actor:	Usuario, Funcionario
Descripción:	Los funcionarios tienen un cronograma de entrega de los formularios a los usuarios, estos formularios serán registrados en la pagina web para su llenado correspondiente.
Flujo Principal:	Eventos
	<ul style="list-style-type: none"> • Se realiza un control para la habilitación de acceso a formularios. • Modificación de fechas con visto bueno del Funcionario. • Clasificación de formularios de acuerdo al tipo de trámites y periodos.
Precondición:	Formularios aprobados
Pos condición:	La habilitación programada para los formularios via Web, brindara un buen servicio a los Usuarios
Presunción:	Los usuarios tendrán un buen servicio siempre y cuando cumplan con sus fechas especificas de entrega.

ANEXO F

DIAGRAMA DE COLABORACION

Figura 3.10.- Diagrama de Colaboración: registro y seguimiento de formularios de Trámite por el Usuario

Fuente: [Elaboración Propia]

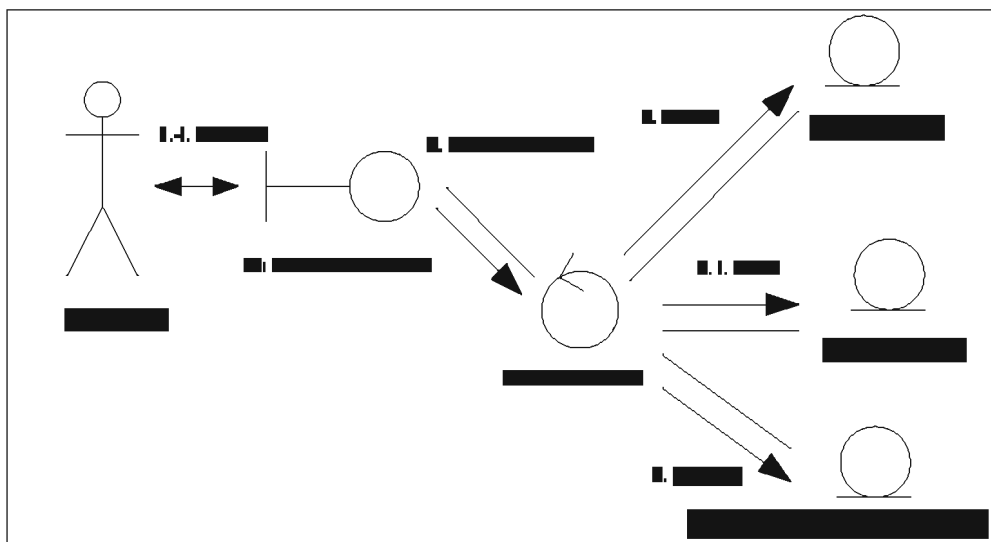


Figura 3.11.- Diagrama de Colaboración: verificación de datos de los formularios de Trámite

Fuente: [Elaboración Propia]

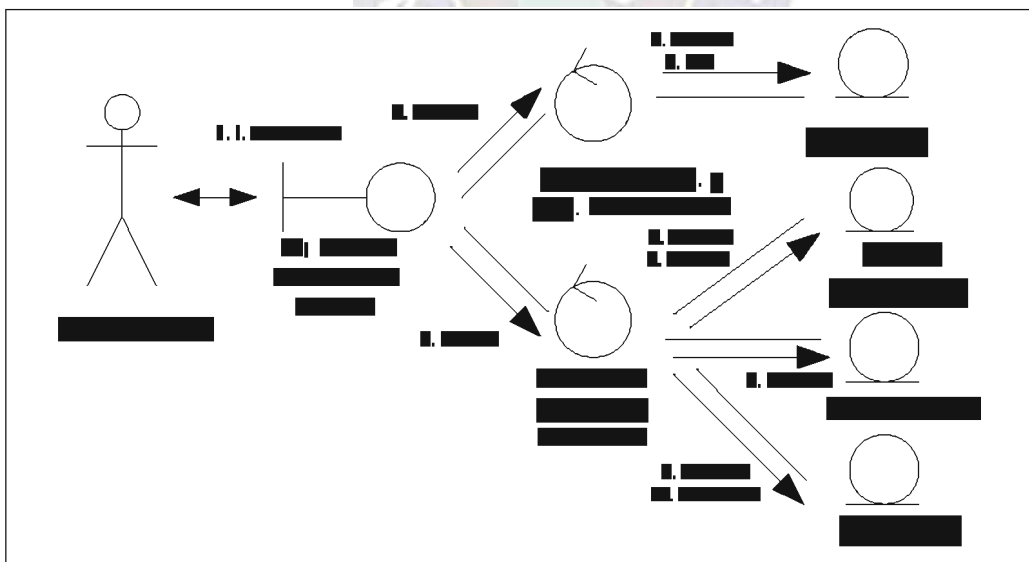


Figura 3.12.- Diagrama de Colaboración: Informe e Indicadores formularios de Trámite

Fuente: [Elaboración Propia]

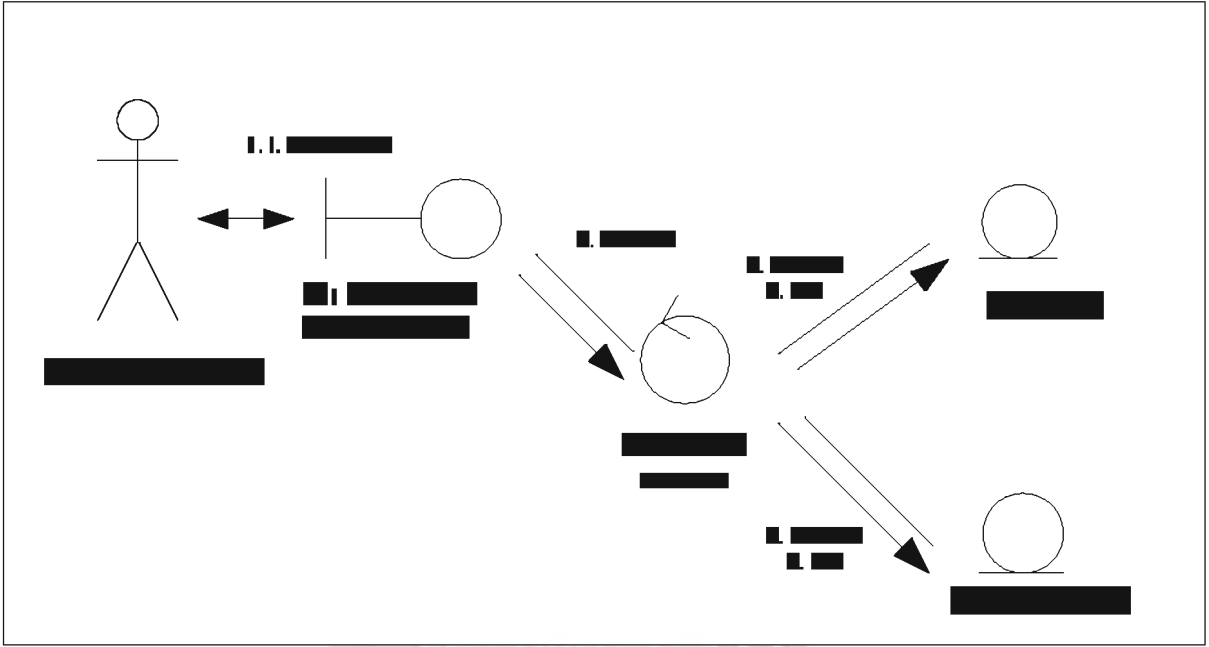


Figura 3.22.- Diagrama de estados, casos de uso: Registro de Usuario

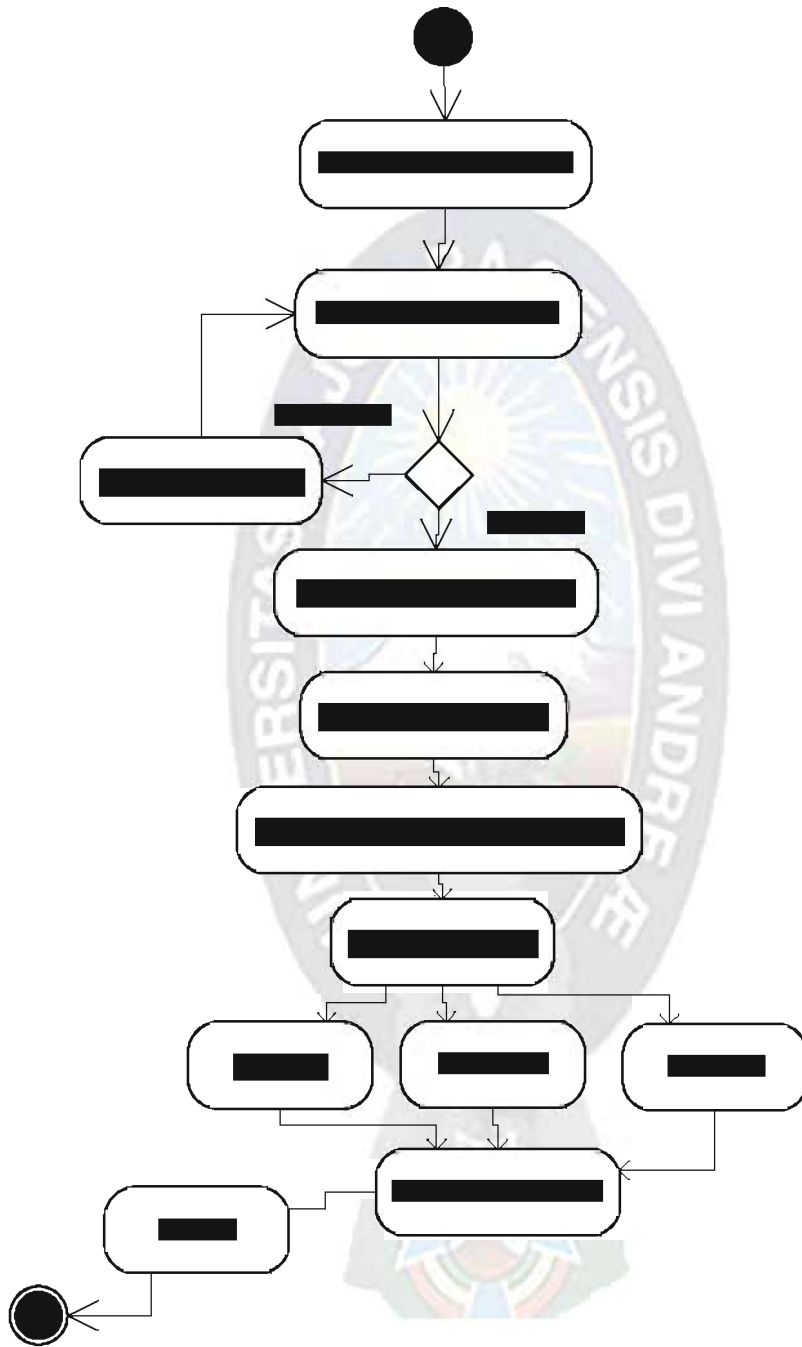


Figura 3.23.- Diagrama de estados, casos de uso: Elaboración de reportes e indicadores

