

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



PROYECTO DE GRADO

**CONTROL Y SEGUIMIENTO FISICO FINANCIERO DE
EJECUCIÓN DE PROYECTOS GMLP CASO: MALLASA**

**PARA OPTAR EL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN INGENIERÍA DE SISTEMAS INFORMÁTICOS**

POSTULANTE: Wilma Alanoca Alanoca

TUTOR: Lic. Franz Cuevas Quiroz M. Sc.

REVISOR: Lic. Luisa Velázquez López M. Sc.

LA PAZ – BOLIVIA

2007

Dedicatoria

*A Dios por haberme permitido concluir
Mis estudios y por concederme la vida, A mis
Padres Julio y Cristina.*

Agradecimientos

A Lic. Franz Cuevas Quiroz, por su orientación y apoyo, sus correcciones y observaciones y apoyo moral durante el desarrollo del proyecto.

A Lic. Luisa Velásquez López, por su valiosa colaboración, por darme la confianza de terminar el proyecto, realizando las observaciones y correcciones correspondientes durante el desarrollo del proyecto.

A todas las autoridades de la Subalcaldía Mallasa, al Subalcalde Arq. Jaime Rivera Quiroz, Lic. Leonardo Yapu y a todo el personal por su colaboración y confianza.

Agradezco también a los Licenciados de la carrera de Informática, por los conocimientos transmitidos durante todo el tiempo de mi formación.

Resumen

La unidad de planificación y control tiene como funciones principales de coordinar la planificación de los recursos con la junta de vecinos, supervisar la elaboración de carpetas de inversión, supervisar el avance físico y económico de los proyectos, apoyando al desarrollo económico social.

En el presente proyecto se desarrolló un software denominado control y seguimiento físico financiero de ejecución de proyectos, para mejorar el manejo de la información y optimizando los procesos de seguimiento y ejecución de proyectos.

Para el análisis y diseño del sistema se utilizó la metodología RUP (Proceso Racional Unificado), que utiliza la notación UML (Unified Modeling Language).

Para la implementación del sistema se aplicó el lenguaje de programación Visual Basic.Net, el gestor de Base de datos SQL Server 2000 y para la elaboración de los informes Crystal Report, coadyuvando de esta manera a cumplir sus necesidades y requerimientos.

El control y seguimiento físico financiero de ejecución de proyectos minimiza el proceso de registro de datos en un tiempo aproximado de 2 minutos y la generación de informes en 3 minutos brindando una información fiable, desminuyendo el tiempo y costo en la elaboración de sus requerimientos.

CAPITULO I - INTRODUCCION Página

1.1	Antecedentes.....	2
1.2	Objeto de estudio.....	4
1.3	Planteamiento del problema.....	4
1.4	Objetivos.....	5
1.4.1	Objetivo general.....	5
1.4.2	Objetivo específico.....	5
1.5	Justificación.....	6
1.6	Alcance.....	6
1.7	Métodos y herramientas.....	7
1.8	Aportes.....	7

CAPITULO II - MARCO TEORICO

2.1	Planificación, control y seguimiento.....	8
2.1.1	Mecanismo de control y seguimiento.....	9
2.1.2	Modalidades de ejecución de proyectos.....	10
2.2	Redes Petri	11
2.2.1	Estructuras básicas de redes de petri	12
2.2.2	Reglas de evolución del marcado.....	13
2.2.3	Ventajas de redes de petri.....	14
2.2.4	Propiedades y validación.....	14
2.3	Metodologías de desarrollo orientado a objetos.....	15
2.3.1	El proceso unificado de desarrollo de software (RUP).....	15
2.3.1.1	Estructura del proceso unificado de desarrollo.....	17
2.3.1.2	Fases y ciclo de desarrollo.....	18
2.3.2	Lenguaje unificado de modelado (UML).....	18
2.3.2.1	Diagrama de casos de uso.....	19
2.3.2.2	Diagrama de transición de estados.....	19
2.3.2.3	Diagrama secuencia.....	20
2.4	Herramienta de desarrollo.....	21

2.4.1	Plataforma microsoft.Net.....	21
2.4.2	Arquitectura .Net.....	22
2.4.3	Crystal Report.....	24
2.4.4	Gestor de base de datos SQL Server.....	25
2.4.5	Los servidores.....	25
2.4.6	Plataforma.....	26
2.5	Diseño del interfaz del usuario	26
2.5.1	Reglas para el diseño de interfaz.....	26
2.5.2	Proceso de diseño de interfaz de usuario.....	27
2.6	Factores de calidad ISO 9126.....	30

CAPITULO III - MARCO APLICATIVO

3.1	Análisis del sistema actual.....	32
3.2	Fase de inicio.....	34
3.2.1	Requerimientos.....	34
3.2.2	Modelo de negocio.....	35
3.2.2.1	Identificación de actores y casos de uso.....	36
3.2.2.2	Modelo de caso de uso de negocio.....	37
3.2.3	Identificación de casos de uso.....	40
3.2.4	Modelado de redes de petri.....	41
3.3	Fase de elaboración.....	45
3.3.1	Análisis.....	45
3.3.1.1	Análisis de arquitectura.....	45
3.3.1.2	Análisis de casos de uso	46
3.3.2	Diseño.....	48
3.3.2.1	Diseño de arquitectura.....	48
3.3.2.2	Diseño de casos de uso.....	49
3.3.2.3	Diseño de clases.....	52
3.4	Fase de construcción.....	54
3.5	Fase de transición.....	58

CAPITULO IV - METRICA DE CALIDAD

4.1	Funcionalidad.....	59
4.2	Portabilidad.....	63
4.3	Usabilidad.....	64
4.4	Mantenibilidad.....	65
4.5	Análisis de Resultados.....	65

CAPITULO V - CONCLUSIONES Y RECOMENDACIONES

5.1	Conclusiones.....	67
5.2	Recomendaciones.....	69

BIBLIOGRAFIA.....	70
--------------------------	-----------

ANEXOS

ANEXO A - Organigrama de Sub Alcaldía de Mallasa

ANEXO B – Identificación de actores y casos de uso

ANEXO C - Identificación de casos de uso con el sistema

ANEXO D - Diagramas de análisis y diseño de casos de uso

LISTA DE FIGURAS

Figura 2.1	Estructura de Redes Petri.....	11
Figura 2.2	Evolución de marcado de redes de petri.....	13
Figura 2.3	Proceso de desarrollo iterativo incremental.....	16
Figura 2.4	Estructura del Proceso de desarrollo de software.....	17
Figura 2.5	Fases de ciclo de desarrollo de software.....	18
Figura 2.6	Diagrama de lenguaje unificado de modelado.....	19
Figura 2.7	Diagrama de Casos de Uso.....	19
Figura 2.8	Diagrama de Estados.....	20
Figura 2.9	Diagrama de Secuencia.....	20
Figura 2.10	Capas de Framework .Net.....	22
Figura 2.11	Crystal report Decisions.....	24
Figura 2.12	Estructura de base de datos.....	25
Figura 2.13	Proceso de diseño de interfaz de usuario.....	27
Figura 3.1	Proceso de control y seguimiento de proyectos.....	33
Figura 3.2	Modulo de organización de proyectos.....	37
Figura 3.3	Modulo de aprobación de proyectos.....	38
Figura 3.4	Modulo de Ejecución de proyectos.....	38
Figura 3.5	Modelo de negocio de casos de uso.....	39
Figura 3.6	Control de carpetas de los proyectos.....	41
Figura 3.7	Control de avance físico y financiero de proyectos.....	43
Figura 3.8	Identificación de paquetes de análisis recepción.....	45
Figura 3.9	Identificación de paquetes de análisis ejecución de proyectos.....	45
Figura 3.10	Diagrama de clase de una realización de Caso de uso de recepción y clasificación de requerimientos.....	46
Figura 3.11	Diagrama de clase de de una realización de Caso de uso Ejecución de proyectos.....	46
Figura 3.12	Diagrama de colaboración de una realización del Caso de uso recepción de requerimientos.....	47

Figura 3.13	Diagrama de colaboración de una realización del Caso de uso de Ejecución de proyectos.....	47
Figura 3.14	Diseño de control y seguimiento Físico financiero de proyectos.....	48
Figura 3.15	Diagrama de clases de diseño de caso de uso Recepción de requerimientos.....	49
Figura 3.16	Diagrama de clases de diseño de caso de uso de Aprobación de proyectos.....	49
Figura 3.17	Diagrama de Secuencia de diseño de caso de uso recepción de requerimientos.....	50
Figura 3.18	Diagrama de Secuencia de diseño de caso de uso de Aprobación de proyectos.....	50
Figura 3.19	Diagrama de Estados para la clase de recepción de requerimientos.....	51
Figura 3.20	Diagrama de Estados para la clase la Aprobación de proyectos.....	51
Figura 3.21	Diagrama de clases de Diseño de Requerimiento y clasificación de los proyectos.....	52
Figura 3.22	Diagrama de clases de Diseño de Licitación y Adjudicación de los proyectos.....	52
Figura 3.23	Diagrama de clases de Diseño de Validación de acceso al sistema.....	53
Figura 3.24	Diagrama de clases de Diseño de Aprobación de los proyectos.....	53
Figura 3.25	Diagrama de clases de Diseño de Avance físico y financiero...	53
Figura 3.26	Ventana de Autenticación.....	54
Figura 3.27	Ventana principal del sistema.....	55
Figura 3.28	Ventana de submenú de proyectos.....	55
Figura 3.29	Pantalla de registro de perfil de proyectos.....	56
Figura 3.30	Pantalla de registro de presupuesto del desembolso de proyecto.....	56

Figura 3.31	Pantalla de registro de avance físico.....	56
Figura 3.32	Pantalla de consultas de los proyectos.....	57
Figura 3.33	Reporte general de los proyectos.....	57
Figura 3.34	Reporte de avance financiero de proyectos.....	58
Figura 3.35	Reporte de avance físico de proyectos.....	58
Figura 4.1	Diagrama de análisis de valor de punto función.....	66

LISTA DE TABLAS

Tabla 3.1	Requerimientos funcionales y no funcionales.....	35
Tabla 3.2	Descripción de negocio. Unidad de planificación y control.....	36
Tabla 3.3	Casos de uso: Ejecución de proyectos.....	40
Tabla 4.1	Computación de métricas de Punto Función.....	62
Tabla 4.2	Ajuste de Complejidad.....	62
Tabla 4.3	Formulación de preguntas para calcular la usabilidad.....	64
Tabla 4.4	Análisis de valores de punto función.....	66

CAPITULO I

INTRODUCCION

La situación económica y política de Bolivia, durante los últimos años presenta signos de estabilidad, sin embargo la realidad es completamente distinta, debido al crecimiento considerable de la población con un entorno marcado por la pobreza y pocas oportunidades laborales.

El Gobierno Municipal de La Paz, no esta al margen de la situación general que vive la sociedad, lamentablemente no puede satisfacer las múltiples necesidades de cada municipio, en esa medida el Gobierno Municipal de Mallasa, pone a disposición sus servicios, contribuyendo con la realización de proyectos que mejoren la calidad y calidez de vida, además de promover el desarrollo económico y social del municipio.

Actualmente la información es una de las herramientas más importantes e indispensable para la oportuna toma de decisiones para las organizaciones públicas como para las privadas. Por otro lado con el avance de la tecnología muchas instituciones han optado el uso de sistemas informáticos, pero no llegando a satisfacerles totalmente sus necesidades, debido a la necesidad de contar con un sistema de información oportuno, confiable y seguro en el servicio.

Es así, que el presente trabajo pretende desarrollar un sistema de información de control y seguimiento físico financiero de ejecución de proyectos, para el Gobierno Municipal de Mallasa contribuyendo con la elaboración de planes que coadyuven al desarrollo Nacional.

En el desarrollo del proyecto se consideran los procesos más importantes de control y seguimiento físico financiero de ejecución de proyectos. Estos procesos permitirán realizar la verificación del presupuesto asignado a un determinado proyecto, además de controlar el avance físico financiero de ejecución de proyectos, considerando el control de materiales y la mano de obra que intervienen en dicho proyecto.

El sistema de control y seguimiento físico financiero de ejecución de proyectos, brindará información referente al avance físico financiero de proyectos, a través de reportes que ayudaran a los usuarios a la adecuada toma de decisiones.

En el capítulo II del Marco teórico, se describirá todas las definiciones y conceptos que se utilizará para el diseño y análisis del presente proyecto, en el capítulo III del Marco aplicativo, es donde se realizará el análisis de la situación actual de los procesos para luego realizar el diseño del sistema propuesto y en el capítulo IV la calidad del software, donde se demuestra el grado de utilidad y facilidad de uso.

1.1 ANTECEDENTES

De la institución

El Gobierno Municipal de La Paz, como parte de las políticas de descentralización, de brindar una mejor atención a la sociedad y mejorar su calidad de vida, crea la Subalcaldía de Mallasa la cual figura con presupuesto desde el año 1999 como inicio de sus actividades y funciones en las oficinas de la Alcaldía de La Paz. Posteriormente en el año 2000 comenzó a funcionar en el distrito 20 de Mallasa en la avenida florida, para que la atención sea eficiente y efectiva a la sociedad.

La Subalcaldía de Mallasa no solo se limita al cumplimiento de servicios básicos, mantenimiento de vías, áreas verdes, orden publico, sino también al desarrollo económico social, consolidando como un circuito Eco Turístico Cultural a las seis zonas dependientes a la Subalcaldía de Mallasa, cuenta con una población total de 5082 habitantes.

La Subalcaldía de Mallasa no cuenta con un sistema automatizado para el almacenamiento de información, lo que si existe personal que recauda toda la información sobre la ejecución de los proyectos, como el presupuesto asignado, el inicio

y el final de cada proyecto, estas son almacenadas en "SIGMA" que son hojas de calculo en Excel, el resto de la información detallada se almacena en grandes archivos manuales.

Del proyecto

A nivel mundial existen sistemas de información para la planificación y mantenimiento de infraestructuras, pero de diferentes características como ser:

- Lulo Win, es un sistema de control de obras de construcción, posee un conjunto de reportes generales que permite recuperar la información detallada de la obra.
- Lulo Gron, es un sistema que almacena cronograma de obra, cronograma físico, cronograma financiero, según las partidas de presupuesto y asignación de fechas.
- Menfis, software elaborado para la planificación y ejecución de proyectos.

A nivel Nacional, se tiene:

- Sistema de información geográfica (SIG), que ayuda a resolver infinidad de consultas y brindar información referente a los proyectos.
- Sistema Integrado de Gestión y Modernización Administrativo (SIGMA), para el manejo y asignación de los recursos, administrado por el Ministerio de Hacienda como un órgano ejecutivo del sistema.

El ministerio de hacienda, administra a los Gobiernos Municipales de cada departamento de manera eficiente y éstos a la vez poseen municipios que son encargados de:

- Lograr municipio productivo y habitable
- Consolidar municipio solvente con identidad
- Crear municipio con equidad de género integrador

Dentro de cada municipio se cuenta con:

Unidad de planificación y control, que tiene funciones principales de coordinar la planificación de los recursos con la junta de vecinos, supervisar la elaboración de carpetas de inversión, supervisar el avance físico y económico de los proyectos.

Unidad de mantenimiento Barrial, es una unidad ejecutora y operativa su función es la responsabilidad de carpetas de inversión en cuanto a la infraestructura, inicio de contratación e informes de conclusión de obras.

Para cumplir las funciones se requiere de soluciones inmediatos, ya que la información carece de claridad, veracidad y confiabilidad para la toma de decisiones, lo que ocasiona deficiencia en el desempeño de funciones.

En la actualidad en la carrera de informática de la Universidad Mayor de San Andrés, cuenta con trabajos de proyecto de grado relacionados con el presente tema, cito algunos de ellos:

- “Sistema de seguimiento y supervisión de obras de construcción civil” Edgar Revollo [Revollo,1997], empleando la metodología de análisis y diseño orientado a objetos.
- “Sistema de elaboración de presupuesto y control de ejecución de construcción civil” Siñani Davila [Siñani, 2004], empleando la metodología de Desarrollo de Software (RUP) y el Lenguaje Unificado de Modelado (UML).
- “Sistema de información para la administración de proyectos y control de almacén de la unidad de mantenimiento de servicios eléctricos” Velásquez Miranda Lourdes [Velásquez, 2003], empleando la metodología estructurado Moderado de Yourdon y el enfoque de pressman.

1.2 OBJETO DE ESTUDIO

Facilitar al usuario información coherente, confiable, integra y oportuna para el control y seguimiento físico financiero de ejecución de proyectos.

1.3 PLANTEAMIENTO DEL PROBLEMA

Toda unidad pública o privada de un tiempo a esta parte se vino utilizando la tecnología informática como una herramienta de oficina, más que una herramienta de ayuda a la toma de decisiones.

Después de haber realizado un estudio preliminar del problema se concluye lo siguiente:

- La elaboración de informes de la ejecución de cada proyecto se la realiza a través de hojas electrónicas” SIGMA”, lo que provoca que no se tenga reportes y costos del avance de los proyectos.
- Las carpetas elaboradas no poseen una organización específica de la información de los proyectos.
- Existe perdida de tiempo y recurso en la elaboración de reportes de los proyectos.
- Falta de control de los recursos que interviene en la realización de los proyectos.
- No permite registrar y disponer información oportuna.

- Falta de procesos de búsqueda de las carpetas solicitados por los usuarios.
- Existencia de duplicidad de información en la elaboración de informes.
- Falta de reportes mensuales y anuales del rendimiento del avance de los proyectos.

Para poder afrontar con éxito surge el planteamiento del siguiente problema:

¿El sistema de Información, permitirá registrar y disponer información oportuna, confiable y segura, en cuanto a la elaboración y clasificación de la información solicitados por el GMLP y los usuarios?

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

Diseñar e implementar un sistema de información para el control y seguimiento físico financiero de ejecución de proyectos para el Gobierno Municipal de Mollasa, que permitirá disponer información oportuna, para la adecuada toma de decisiones.

1.4.2 OBJETIVOS ESPECIFICOS

- Implementar una base de datos para el registro de todas las actividades involucradas en la ejecución del proyecto.
- Registro de carpetas que permitan la identificación de cada proyecto de manera automática para la realización de reportes.
- Registro de proyectos, accediendo al flujo de información detallada de cada proyecto.
- Elaborar un modelo de control de carpetas de los proyectos aplicando las redes de petri.
- Elaborar un proceso, que permita la búsqueda de carpetas vinculados al avance de cada proyecto.
- Elaborar un modelo de control de avance físico y financiero de proyectos mediante las redes de petri.
- Diseñar un interfaz de usuario para facilitar el manejo del sistema, reduciendo el tiempo en realizar los registros y los reportes requeridos.
- Evaluar la calidad del sistema, aplicando la ISO 9126.

1.5 JUSTIFICACIÓN

Se hace necesario implementar este proyecto, porque permitirá incrementar los beneficios y el desarrollo social, que reúnan con todas las condiciones de seguridad y de calidad de vida ofreciendo un control adecuado del rendimiento en la ejecución de proyectos.

La Subalcaldía de Mallasa se beneficiará con la información oportuna, es decir optimizar el uso de los recursos disminuyendo el costo, incluso las horas de trabajo y generación inmediata de los informes. Además, el hecho de contar con una herramienta informática, motivará al personal a presentar mayor interés, de esta manera aumentar así la confianza del entorno municipal.

Por otro lado se tiene equipos disponibles para implementar el sistema, además de contar con licencias para el uso lenguaje de programación de Visual Studio.Net y SQL Server 2000.

1.6 ALCANCE

En cuanto al alcance del proyecto, será el diseño y la implementación de un sistema de control y seguimiento físico financiero de ejecución de proyectos, brindando información actualizada de cada proyecto, emisión de reportes y la capacitación personal de la unidad en el manejo del sistema automatizado.

El presente proyecto, se limitará básicamente al desarrollo del objetivo principal y los objetivos específicos, no realizando el manejo de contabilidad por lo que se halla fuera de los objetivos.

1.7 METODOS Y HERRAMIENTAS

La metodología que se utilizaran en el desarrollo del proyecto son los siguientes:

- El método científico, es una sucesión de reglas que nos llevan al conocimiento científico, son pasos sistémicos que nos permiten generar y utilizar instrumentos desde el hecho que observamos, hasta la formulación de hipótesis, que implica búsqueda de soluciones.
- Métodos de obtención de información, se obtendrá mediante encuestas, entrevistas al personal e investigación en páginas Web.
- La Metodología que se utilizará es el proceso unificado de desarrollo de software (RUP), es un método incremental que comprende de cuatro fases; Inicio, elaboración, construcción y transición.
- Se utilizará la herramienta del Lenguaje unificado de modelado (UML), para modelar el análisis y diseño del sistema.
- Para la implementación del sistema, se utilizará la tecnología.Net, que cuenta con la facilidad de acceso a datos para la red y crystal Report para la elaboración de los diferentes reportes.
- Como motor de base de datos se utilizará el lenguaje estructurado de consultas SQLServer 2000, por el volumen de información y la velocidad.
- Método de verificación de calidad en base al estándar ISO 9126.

1.8 APORTES

El presente proyecto apoya al control y seguimiento físico financiero de ejecución de proyectos que podrá ser implementado en la Subalcaldía de Mallasa, proporcionando datos exactos, elaboración inmediata de informes, reportes y coadyuvando en la adecuada toma de decisiones.

También se aplicará las redes de petri para modelar el control del avance físico financiero y el control de carpetas de los proyectos, además de proporcionar al usuario final los datos estadísticos del avance físico de cada proyecto.

CAPITULO II

MARCO TEÓRICO

El marco teórico viene ha ser como una base principal para sustentar la investigación, para lo cual se debe hacer una selección de los aspectos más importantes de cada metodología.

Es decir, dando una pequeña descripción de los conceptos fundamentales de las herramientas y técnicas que contribuyen en el desarrollo e implementación del software.

2.1 CONTROL Y SEGUIMIENTO

La planificación se efectúa con un mayor nivel de incertidumbre y naturalmente, esta se refleja también en los parámetros de control.

En ese caso, el control debe ser altamente dinámico, de modo que acompañe a la etapa de ejecución, de manera permanente y en todas sus fases, proporcionando información constante de la situación real en las diversas variables, para permitir al agente evaluar y decidir en cuanto a la gravedad de los errores y tomar las decisiones necesarias.

El sistema de control se proyecta sobre la base de previsiones del futuro y debe ser suficientemente flexible para permitir adaptaciones y ajustes que se originan en discrepancias entre el resultado previsto y el ocurrido.

El seguimiento de proyecto consiste en la supervisión continua o periódica de ejecución física, en cuanto al avance, cumplimiento de cronogramas de actividades realizadas durante el periodo de ejecución y el cumplimiento de las metas, se utilizara como instrumento al Plan Operativo Anual.

2.1.1 MECANISMO DE CONTROL Y SEGUIMIENTO

Los mecanismos de control pueden clasificarse, dependiendo del momento en que se realice la acción de control, en la forma que se indica a continuación:

- *Control direccional:* El mecanismo de control actúa antes de que la actividad este totalmente concluida. En este caso el control se realiza de modo continuo y no en puntos determinados, de modo que cada elemento de la acción sea el resultado de la rectificación casi instantánea de la acción anterior.
- *Control aprobado – reprobado:* En este caso, el receptor del control se somete a un examen después de concluidas determinadas actividades. En caso de aprobación se permite la realización de la actividad siguiente.
- *Control post-operacional:* El mecanismo de control sólo se pone en funcionamiento después de concluida toda la operación. La información para la acción correctiva este tipo de control, solo se utilizará en un periodo futuro cuando se inicie la planificación para un nuevo ciclo de actividades.

Vale la pena mencionar que estos tres tipos de control no son mutuamente excluyentes, sino que más bien, deben ser complementarios. La decisión de emplear un tipo aislado de control o una combinación de los tipos antes mencionados, esta en función del carácter del sistema que se desea controlar y del nivel de complejidad que se intenta introducir en los mecanismos de control.

El mecanismo de control, propone el seguimiento de la ejecución del proyecto integral y la introducción de las correcciones que resultarán de la experiencia adquirida a lo largo del mismo, comprende de: Control físico, control Financiero y el equilibrio meta /costo o tiempo/costo.

- ***El control físico:*** El instrumento básico del control físico es la técnica de redes. Estas sólo son algunas de las herramientas utilizadas para el seguimiento en los proyectos como la técnica PERT/CPM/ROY integrada y cronograma de Gantt.
- ***El control Financiero:*** En él, se deben tener en cuenta aspectos tales como las inversiones, que es importante tener un seguimiento detallado de las finanzas del proyecto, la mayoría de los casos, el responsable es el ejecutor de la obra.
- ***El equilibrio meta /costo o tiempo/costo:*** Para desarrollar este control, se deben diseñar indicadores, que establezcan relaciones entre los tiempos empleados en la conclusión de una actividad y los gastos realmente efectuados.

El Status Index es uno de los más utilizados en este control, éste suministra información acerca de: relación tiempo/costo para una fecha determinada, tiempo y costo para la terminación del programa, áreas que presentan condiciones críticas, entre otras.

Su expresión matemática se define:

$$SI = (DR / DP) * (P / GR)$$

Donde: SI : Status Index
 DR: Duración real
 DP: Duración programada
 P : Presupuesto
 GR : gasto real.

Si el resultado del índice es uno, se dice que el proyecto marcha de acuerdo a lo planeado y programado; mientras que, un resultado menor que uno representa un retraso y viceversa con un resultado mayor a uno.

2.1.2 MODALIDADES DE EJECUCIÓN DE PROYECTOS

Las modalidades de ejecución de proyectos son las siguientes:

Por Administración delegada: Proceso que se realiza mediante la contratación de empresas oferentes que están inscritas en el registro de beneficiarios y satisfagan la demanda de la ejecución de proyectos de inversión bajo la normatividad definida en las normas básicas del sistema administrativo de bienes y servicios.

Por Administración directa: El proceso de ejecución de los proyectos es realizado directamente por las dependencias municipales o por la unidad.

Por Administración Compartida: El proceso de ejecución de los proyectos, es realizado con la participación municipal y las organizaciones vecinales.

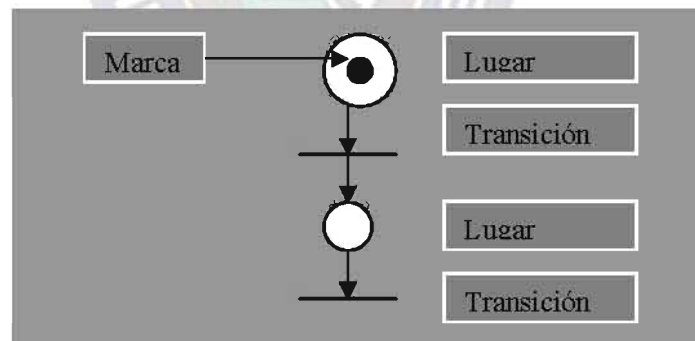
2.2 REDES PETRI

Las Redes de Petri representan una alternativa para modelar sistemas, sus características hacen que, para algunos problemas las redes de petri funcionen de una manera natural.

Las Petri Net como ahora conocemos a las Redes Petri, fueron desarrolladas por el alemán Karl Adam Petri en 1962, la Red Petri es una herramienta gráfica para modelar concurrencia y sincronización, es decir modelar el comportamiento y la estructura de un sistema, además de estudiar su comportamiento por separado y de está manera aislarlos, pero siempre teniendo en cuenta la interacción que guardan con los otros módulos.

Las Redes de Petri son muy similares al diagrama de Transición de estados, donde se debe reconocer los eventos y transiciones que se dan, de está manera conocer las condiciones que se necesitan para dar cierto evento, podemos diseñar los módulos y relacionarlos con otras condiciones, y para esto necesitamos saber la estructura de redes de petri. En la siguiente Figura 2.1 se muestra la estructura de Redes de Petri, la cual está formada por lugares o plazas, transiciones y arcos dirigidos. Los arcos conectan un lugar a una transición o una transición a un lugar.

Figura 2.1 Estructura de Redes Petri



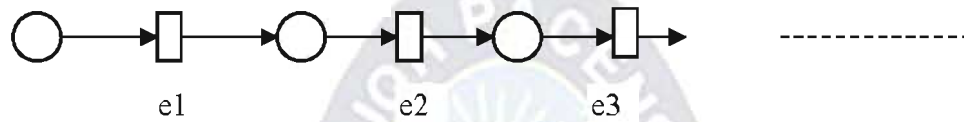
Fuente: [Redes de Petri, Murata]

Una red de petri es un conjunto formado por $R = \{P, T; Pre, Post\}$, donde P es un conjunto de fichas de cardinal n , T un conjunto de transiciones de cardinal m , Pre es la aplicación de incidencia previa que viene definida como $Pre = P \times T \rightarrow \text{Naturales}$ y la $Post$ viene definida como $Post = P \times T \rightarrow \text{Naturales}$.

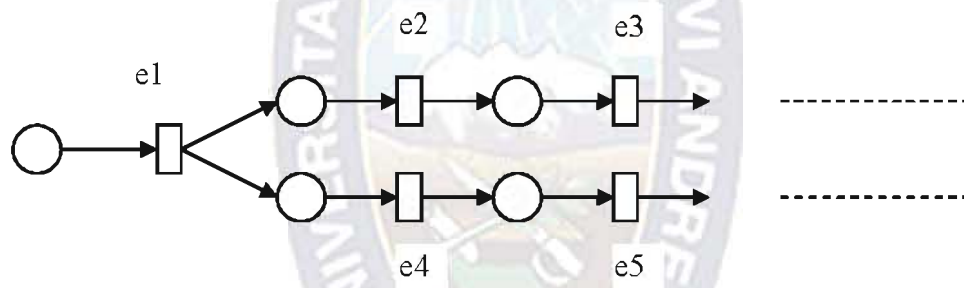
2.2.1 ESTRUCTURAS BÁSICAS DE REDES DE PETRI

A continuación se muestra las estructuras básicas de Redes Petri, [ingeniería de software de Fernando García].

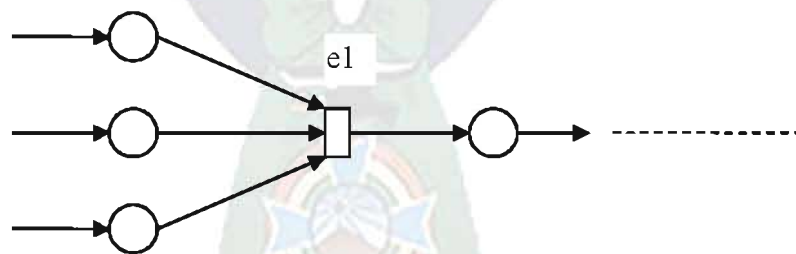
- Secuencia de eventos/acciones:



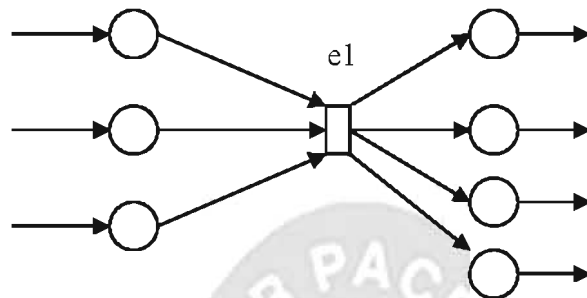
- Concurrencia



- Sincronización



➤ Sincronización y concurrencia

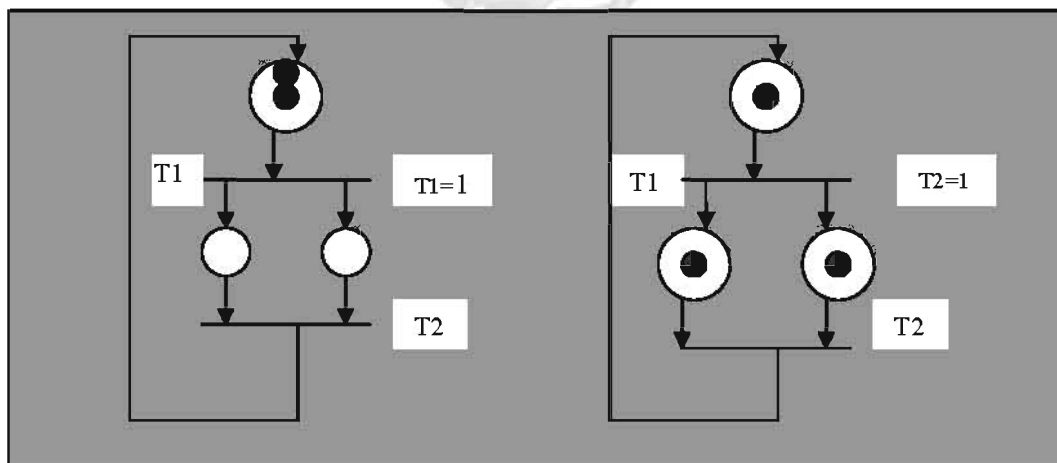


2.2.2 REGLAS DE EVOLUCION DE MARCADO

Los marcados presentes en los nodos controlan la ejecución de las transiciones de la red, el marcado cambia al franquear las transiciones, de manera que para franquear una transición ha de estar validada y sensibilizada.

Cuando una transición se franquea desaparecen las marcas de los lugares de origen y se añade una marca a cada uno de los lugares destino, pero cuando dos transiciones están sensibilizadas a la vez pueden entrar en conflicto. A continuación se muestra en la Figura 2.2 la evolución del marcado de redes de petri.

Figura 2.2 Evolución del marcado de Redes Petri



Fuente: [Redes de Petri, Murata]

2.2.3 VENTAJAS DE REDES DE PETRI

Una de las ventajas de redes petri, es modelar el comportamiento por separado, pero siempre tomando en cuenta la interacción que guardan con los otros módulos.

También permiten modelar sistemas donde un recurso es compartido por dos procesos de forma que el uso del recurso durante la ejecución de un proceso impide que dicho recurso sea utilizado por el otro proceso, a continuación se menciona las ventajas más importantes de redes de petri.

- Tratamiento individual de procesos independientes.
- Procesos paralelos o concurrentes.
- Recursos compartidos.

2.2.4 PROPIEDADES Y VALIDACIÓN

La validación en las redes de petri, consiste en comprobar que se cumplan las siguientes propiedades de: Vivacidad, limitación, reversibilidad.

- **Vivacidad:** Se trata de un concepto relacionado con la idea de “no bloqueo” es decir, una transición se dice que es viva si para un marcado inicial existe una secuencia de franqueos para la cual se puede franquear esa transición, si todas las transiciones de una red son vivas, entonces las redes de petri se llama viva y así la red nunca se bloquea. Se puede tener también una red Pseudos-vivas, donde existe algunas transiciones vivas y la red no se bloquea totalmente.
- **Limitación:** Se dice que la red está K-limitada si para todo marcado alcanzable tenemos que ningún lugar tiene un número de marcas mayor que k. Las redes 1-limitadas pueden implementarse mediante biestables, estas redes son conocidas como binarias.
- **Reversibilidad:** Una Red de Petri es reversible si para cualquier marcado alcanzable es posible volver al marcado inicial.

2.3 METODOLOGÍAS DE DESARROLLO ORIENTADAS A OBJETOS

En esta parte se describen las metodologías y herramientas que se utilizan para el desarrollo y la implementación del software.

Modelo, Un modelo es una abstracción del sistema, especificando el sistema modelado desde cierto punto de vista y en un determinado nivel de abstracción [LARMAN, 1999].

Un modelo siempre identifica el sistema que está modelando, es decir que un modelo debe describir las iteraciones entre el sistema y los que le rodean [JBR, 1999].

Método, Un método es un proceso disciplinado que permite generar un conjunto de modelos que describen varios aspectos de un sistema de software en desarrollo, utilizando alguna notación bien definida [Jacobson, Booch, R, 1999]

Los métodos abarcan una gama de tareas que incluyen análisis, requisitos, diseño, construcción de programas y mantenimiento [PRESSMAN, 1999].

Metodologías, Es una colección de métodos aplicados, a lo largo de ciclo de vida de desarrollo de software unificados por alguna aproximación general [JBR, 1999].

2.3.1 EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE (RUP)

La metodología RUP, llamada así por sus siglas en inglés Rational Unified Process, es la metodología estándar de la industria para la construcción completa del ciclo de ingeniería de software, es decir un conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software que está formado por componentes interconectados a través de interfaces bien definidas.

La metodología RUP, tiene características importantes que se resumen en las siguientes:

- a) Desarrollo iterativo de Software: Permiten tener un entendimiento del problema, mediante un refinamiento sucesivo del software. Habilita una fácil retroalimentación con los usuarios y la medición de progreso según vaya avanzando la implementación, en la Figura 2.3 se muestra el proceso de desarrollo iterativo incremental.

Figura 2.3 Proceso de desarrollo iterativo incremental



Fuente: [Jacobson, Booch, R., 1999]

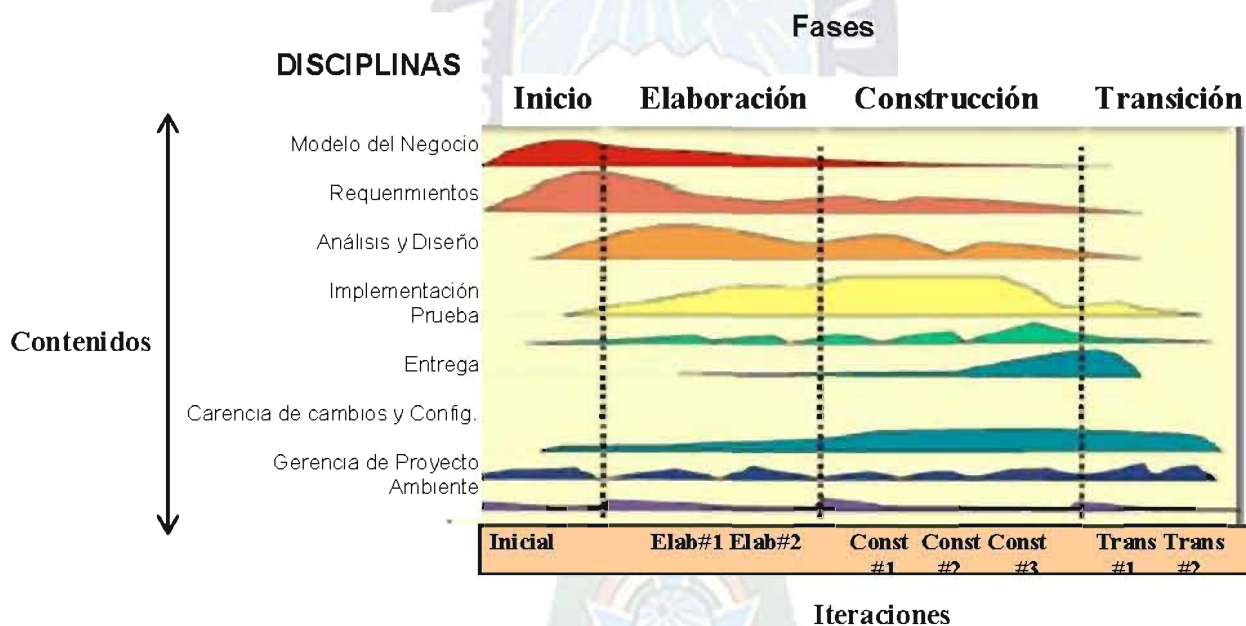
- b) Administración de requerimientos: Este proceso permite organizar, documentar la funcionalidad y las restricciones requeridas. Es decir que los requerimientos de negocio son fácilmente capturados.
- c) Arquitectura basada en componentes: El proceso de desarrollo de software se enfoca en el desarrollo de una arquitectura ejecutable, robusta y flexible basada en componentes.
- d) Mantenimiento visual del Software: Captura la estructura y el comportamiento de la arquitectura y componentes, los que ayudan a transmitir o comunicar distintos aspectos de software y ver si los elementos del software se adaptan, manteniendo la consistencia entre el diseño y la implementación.
- e) Verificación de calidad de software: El proceso de unificado de desarrollo de software crea pruebas para cada escenario para asignar que todos los requerimientos están propiamente implementados, verificar la calidad de software con respecto a los requerimientos en cuanto a la confiabilidad, funcionalidad y desempeño del sistema.
- f) Control de cambio de software: Se puede controlar, llevando un registro y monitorear cambios, de tal forma lograr un desarrollo iterativo, también establece espacios de trabajo seguros para cada desarrollador.

2.3.1.1 ESTRUCTURA DEL PROCESO UNIFICADO DE DESARROLLO

El Proceso unificado de desarrollo de software está compuesto por 4 fases, cada una de estas fases es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala.

El eje horizontal representa la parte dinámica del proceso, el tiempo, iteraciones y las metas, el eje vertical representa la parte estática del proceso donde se describe los flujos de trabajo, requisitos, análisis, diseño, implementación y prueba. Las curvas son una aproximación de hasta donde se llevan a cabo los flujos de trabajo en cada fase, como se puede ver en la Figura 2.4.

Figura 2.4 Estructura del Proceso de desarrollo de software



Fuente: [[Jacobson, Booch, R., 2000]

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.

2.3.1.2 FASES DEL CICLO DE DESARROLLO

El proceso de desarrollo de software tiene cuatro fases que están relacionadas con el tiempo de desarrollo.

Figura 2.5 Fases de ciclo de desarrollo de software



Fuente: [[Jacobson, Booch, R., 2000]

1) Fase de inicio: En esta fase se especifica el alcance del proyecto, es decir, se identifican las entidades externas que interactuarán con el sistema, priorizando los riesgos más importantes y se planifica en detalle la fase de elaboración.

2) Fase de elaboración: Durante esta fase, se analiza en detalle la mayoría de los casos de uso del producto, se diseña la arquitectura del sistema desarrollando un plan comprensivo de cómo el proyecto será completado con los recursos disponibles. Al final de la fase de elaboración, se empieza a planificar en forma más detallada la primera iteración de la fase de construcción y esbozar en términos más generales las iteraciones restantes.

3) Fase de Construcción: En esta fase, el producto se desarrolla a través de iteraciones donde cada iteración involucra tareas de análisis, diseño y implementación, el cual estará listo para ser transferido a los nuevos usuarios, incluyendo la documentación del sistema.

4) Fase de transición: En esta fase, se libera el producto de software a los usuarios para un uso real, realizando la instalación, configuración, entrenamiento, soporte y mantenimiento, además de la entrega de ejecutables del producto, la documentación de desarrollo actualizada y los materiales de formación del usuario final.

2.3.2 LENGUAJE UNIFICADO DE MODELADO (UML)

El lenguaje unificado de modelado (Unified modeling language), se define como un lenguaje que permite especificar, modelar, visualizar y construir los artefactos de los sistemas de software (BRJ 1999). Se puede decir, que es un sistema notacional destinado a los sistemas de modelado que utilizan conceptos orientados a objetos.

El lenguaje unificado de modelado permite que los sistemas sean vistos o representados por diferentes tipos de diagramas.

Figura 2.6 Diagramas del lenguaje unificado de modelado

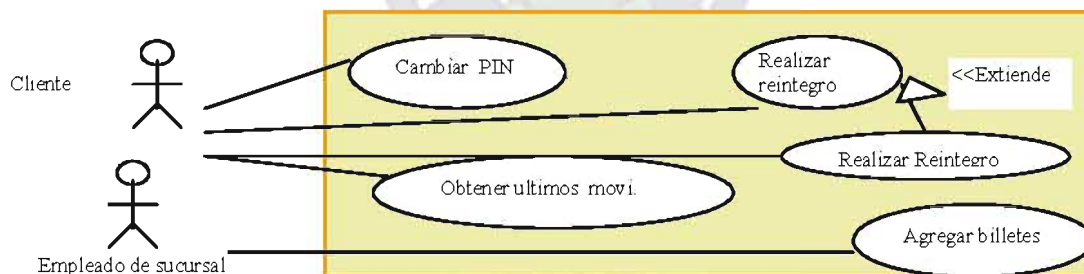


Fuente: [LARMAN, 1999]

2.3.2.1 DIAGRAMA DE CASOS DE USO

Un caso de uso es una descripción de la secuencia de iteraciones que se producen entre un usuario y un sistema de cómputo, es decir que es una secuencia de interacciones que son desarrolladas en un sistema en respuesta, a un evento que inicia un actor sobre el propio sistema. Se explica gráficamente en la Figura 2.7.

Figura 2.7 Diagrama de Casos de Uso



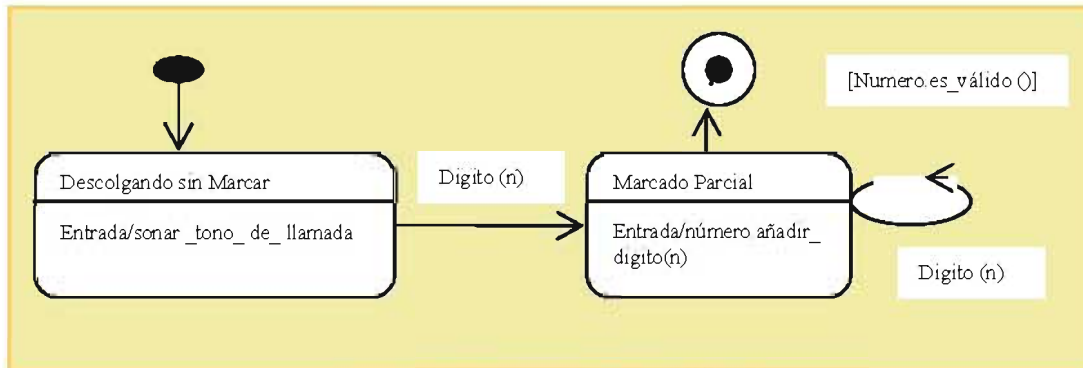
Fuente: [LARMAN, 1999]

2.3.2.2 DIAGRAMA DE TRANSICIÓN DE ESTADOS

La secuencia de estados, es cuando un objeto pasa durante su tiempo de vida en respuesta a estímulos recibidos, teniendo como característica los eventos que hacen posible la transición de un estado a otro.

En la Figura 2.8 muestra un objeto que pasa de un estado a otro por la ocurrencia de un evento y se dice que ha sufrido una transición.

Figura 2.8 Diagrama de Estados



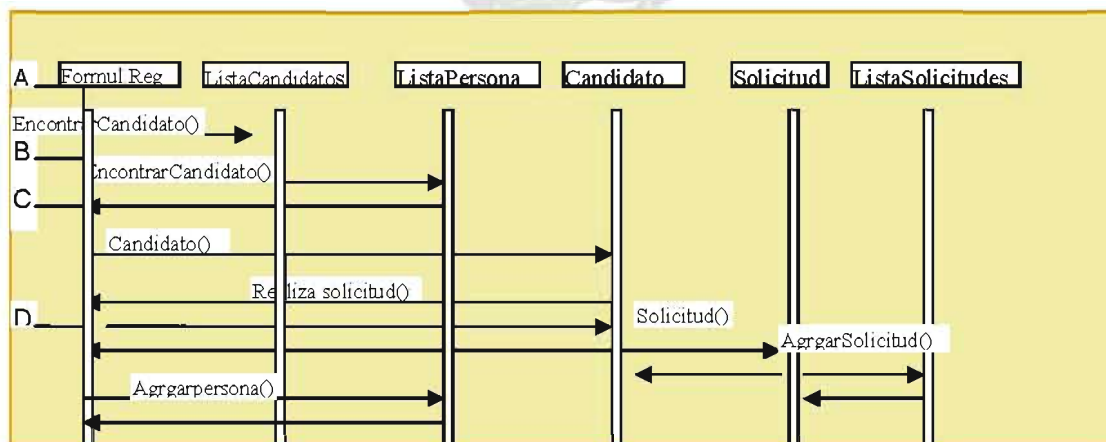
Fuente: [LARMAN, 1999]

2.3.2.3 DIAGRAMA DE SECUENCIA

Un diagrama de secuencia representa una forma de indicar el periodo de cuando el objeto esté desarrollando, una acción directamente o a través de un procedimiento.

En la Figura 2.9 se muestra un diagrama de secuencia donde el eje vertical representa el tiempo de vida del objeto, y el eje horizontal representa los objetos y actores participantes de la interacción, cada actor tiene una línea vertical y los mensajes se representan mediante flechas entre los distintos objetos.

Figura 2.9 Diagrama de Secuencia



Fuente: Visio Software

2.4 HERRAMIENTA DE DESARROLLO

2.4.1 PLATAFORMA MICROSOFT.NET

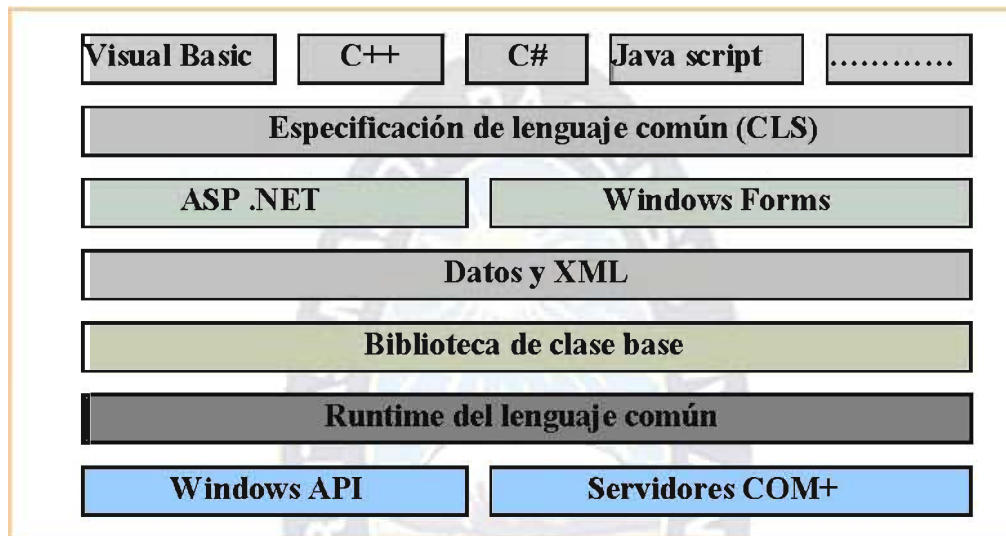
Microsoft.Net ofrece una visión orientada a objetos del sistema operativo de Windows, la mayoría de los desarrolladores serán capaces de generar software más potente y en menor tiempo, .NET es un cambio radical para los desarrolladores de Windows, la tecnología más importante de .NET son los servicios Web de XML.

- 1) .NET Framework incluye el CLR (common lenguaje runtime), y la biblioteca de clases, el CLR es una base estándar para generar diversas aplicaciones y en las tecnologías de la biblioteca están ASP .NET.
- 2) Servicios .Net BUILDING BLOCKS, son servicios programables distribuida que están disponibles tanto en modo conectado como desconectado, un servicio puede invocarse desde un ordenador aislado no conectado a Internet.
- 3) Visual Studio.Net, ofrece un entorno de desarrollo de alto nivel para desarrollar aplicaciones que se ejecutan sobre el .Net framework , proporciona las tecnologías fundamentales para simplificar la creación, implantación y evolución constante de aplicaciones y servidores Web seguros. Visual Studio .NET admite varios lenguajes de programación que pueden utilizarse con el .NET Framework, entre ellos se encuentran Visual Basic .NET, ASP.NET y C #.
- 4) .NET My Services, son un grupo de servicios que permiten a los usuarios acceder a información personal o acceder a servicios disponibles en Internet.
- 5) Servidores .Net, los servidores Enterprise servers proporcionan escalabilidad, fiabilidad, administración e integridad dentro de las organizaciones, las más principales son: Microsoft Windows 2000 server, Microsoft Sql Server 2000, Microsoft Host Integration, Microsoft application center 2000, Microsoft internet security and acceleration server 2000 y Microsoft commence server 2000.

2.4.2 ARQUITECTURA .NET

Para comprender la función que cumple .Net es necesario analizar las numerosas capas en la que se divide .Net Framework tal y como se puede ver en la Figura 2.10.

Figura 2.10 Capas de .Net Framework



Fuente: [Balena Francesco, 2002]

A continuación se describe cada uno de los niveles.

Windows API: .Net proporciona una visión orientada a objeto de las funciones del sistema operativo pero no las sustituye, por lo que la mayor parte de .Net Framework se transforman finalmente en una llamada a una DLL del núcleo de Windows.

Los servicios COM+, en esta parte de la jerarquía podrá crear componentes .Net que saquen partido a los servicios de transacción, sincronización y seguridad, incluso aunque esta aproximación agregue cierta sobrecarga porque la ejecución deberá fluir desde .Net a COM.

El runtime del lenguaje común, es la primera capa que pertenece a .Net Framework, esta capa es la responsable de los servicios básicos de .Net, tales como la administración de memoria, la recolección de los elementos no utilizados, el control estructurado de excepciones y del sub procesamiento múltiple.

La Biblioteca de clase base (BCL), es la parte de .Net Framework que define todos los tipos de datos básicos, tales como System Object (la raíz de la jerarquía de objetos de .Net), tipos numéricos y de fechas, el tipo de string, matrices y colecciones. La BCL contiene también clases de administración las características centrales de .Net, tales como la E/S de archivos, subprocesamiento, serialización y seguridad.

Datos y XML, contiene las clases de .Net que trabajan con bases de datos y con XML, de aquí se comprueba la compatibilidad con XML está basada directamente en el propio .Net Framework en lugar de estarlo sobre los componentes externos, como es el caso de los lenguaje previos a .Net.

ASP .Net y Windows Forms, que se encuentran al mismo nivel en el diagrama, estas porciones del marco de trabajo contienen todas las clases que la interfaz de usuario podrá generar. ASP .NET comprende tanto los Web Forms como los servicios Web XML.

Los Web Forms se utilizan para desarrollar aplicaciones de Internet que cuenten con una interfaz de usuario y su objetivo principal es sustituir a las aplicaciones ASP, aunque podrá seguir ejecutando ASP y ASP.NET en el mismo equipo.

ASP.NET tiene una característica clave, un amplio apoyo para generar aplicaciones que utilizan y exponen servicios Web XML, están desarrolladas en lenguajes compilados y plenamente funcionales tales como el visual Basic.NET y C# , por lo que cabe esperar que el código desarrollado en ASP.NET se ejecutará con mayor rapidez.

Especificación de lenguaje común (CLS), es un conjunto de especificaciones que Microsoft ha suministrado para ayudar a los fabricantes de compiladores. Estas especificaciones fijan el número mínimo de características que un lenguaje .Net debe tener, tales como la compatibilidad con enteros con signos de 16, 32 o 64 bits, matrices con índice inferior cero y control estructurado de excepciones.

Visual Basic.Net y C#, la estructura de estos compiladores son similares, la única diferencia se encuentra en el bloque de espacio de nombres y el bloque de clase en lugar del bloque de modulo. C# es un lenguaje lacónico, que utiliza llaves para marcar el principio y el final de cada uno de los bloques sintácticos.

2.4.3 CRYSTAL REPORT

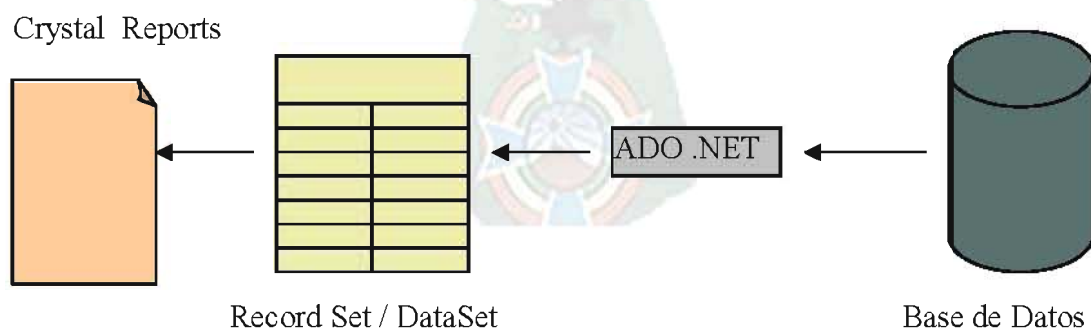
Para la creación de informes en la mayoría de las aplicaciones necesitan evaluar información obtenida de una o varios orígenes de datos mediante procesos realmente complejos con el fin de brindar un resultado capaz de ser comprendido por el usuario final.

El generador de informes Crystal Report Decisions está incluido en la infraestructura de .Net, la misma no debe ser tomada como una simple herramienta para procesar datos y generar documentos ya que el alcance de sus funcionalidades es realmente sorprendente, se cuenta con tres medios para que sean publicados los informes.

- Aplicaciones para Windows
- Paginas de servicios activo ASP. Net
- Servicios Web ASP.Net

Crystal provee un conjunto de clases y un control Web para visualizar un informe, con la ayuda de ADO .Net y su objeto Data set, el cual permite mediante un adaptador especificar el lugar de donde deberían ser obtenidas las filas, a continuación en la Figura 2.11 se muestra la interfaz de la base de datos y Cristal Report.

Figura 2.11 Crystal Report Decisions



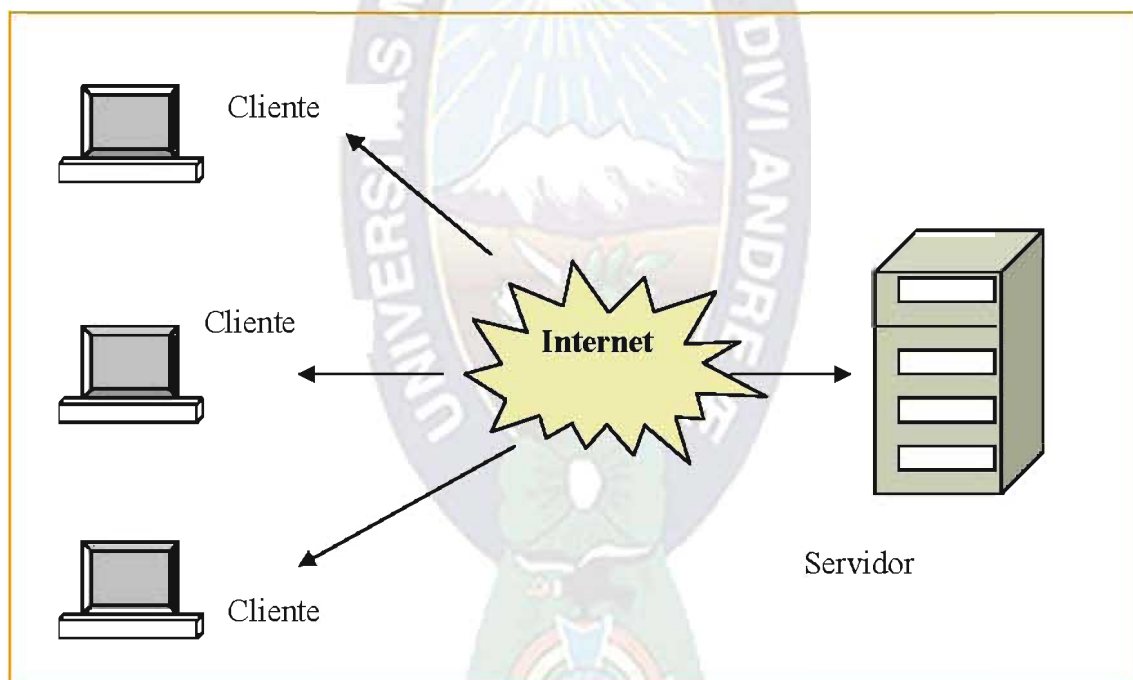
Fuente: [Balena Francesco, 2002]

2.4.4 GESTOR DE BASE DE DATOS SQL SERVER

SQL Server 2000, este producto es una versión del bastión de Microsoft para el manejo de base de datos, es decir un gestor de base de datos preparado a fondo para Internet. SQL Server 2000, lenguaje de base de datos normalizados, es un servidor de base de datos multiusuario Cliente/ Servidor, en la Figura 2.12 se muestra la estructura de gestor de base de datos.

Entre las mejoras de esta versión está su integración con XML, lo que permite que un cliente ejecute consultas SQL que devuelvan datos definidos por XML y pase documentos directamente al DBMS.

Figura 2.12 Estructura de gestor de base de datos



Fuente: Estructura de Cliente/servidor que utiliza el presente proyecto

2.4.5 LOS SERVIDORES

Los servidores que se encuentran disponibles son: Siz Talk Server 2000, Aplicacion Center 2000, Comerse Server 2000, Host Integration Server 2000, Sql Sever 2000 y Mobile Information Server 2000, e Internet Security and Acceleration Server 2000.

2.4.6 PLATAFORMA

El Visual Studio se puede ejecutar en cualquier plataforma Windows que sea compatible con .NET framework utilizando un equipo basado en Windows.

Todas las versiones de a partir de Windows 2000, Windows XP Profesional estándar y posteriores, está se basa en la robustez de los fundamentos de la tecnología probada de Microsoft, para poder sacar partido de todas las capacidades presentes en .NET.

2.5 DISEÑO DEL INTERFAZ DEL USUARIO

El objetivo del diseño del interfaz del usuario, es definir un conjunto de objetos y acciones del interfaz que ayudaran a los usuarios a realizar sus tareas definidas de forma que cumplan con todos los objetivos de usabilidad definidas por el sistema, una vez identificadas las tareas, se crean y se analizan los escenarios del usuario.

2.5.1 REGLAS PARA EL DISEÑO DE LA INTERFAZ

Theo Mantel [Mantel, 97] define tres reglas de oro para el diseño de la interfaz:

1. Dar el control al usuario: Tomando en cuenta los siguientes principios de diseño que permite dar el control al usuario. Definir los modos de interacción de manera que no obligue a que el usuario realice acciones innecesarias y no deseadas, Tener en consideración una interacción flexible, permitir que la interacción del usuario se pueda interrumpir y deshacer, aligerar la interacción a medida que avanza el nivel de conocimiento y permitir personalizar la interacción y diseñar la interacción directa con los objetos que aparecen en la pantalla.
2. Reducir la carga de memoria del usuario: Tomando en cuenta los siguientes principios de diseño reducir la demanda de memoria a corto plazo, establecer valores por defecto útiles y definir las deficiencias que sean intuitivas.
3. Construir de una interfaz consistente: La interfaz deberá adquirir y presentar la información de forma consecuente, tomando en cuenta los siguientes principios

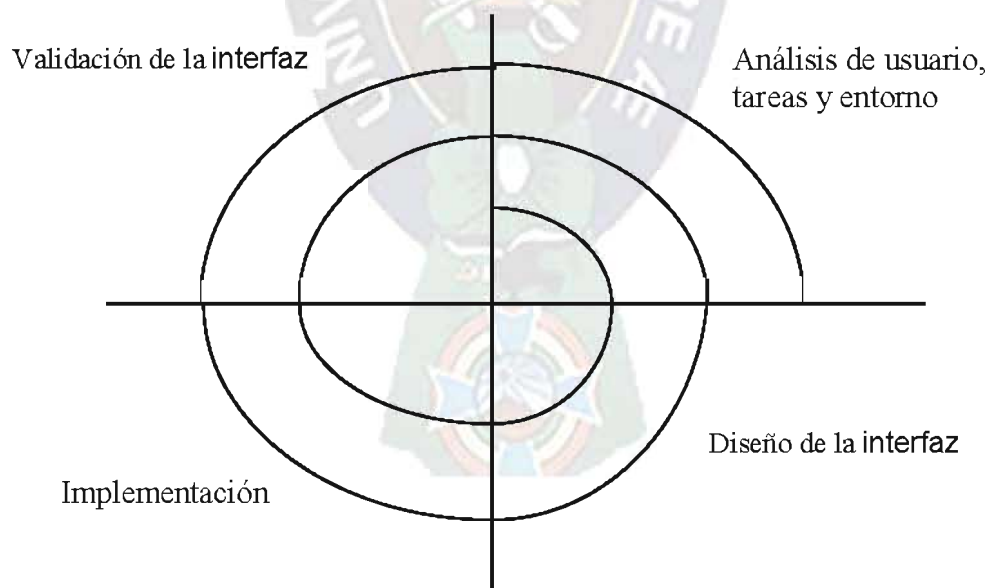
permitir que el usuario realice una tarea en el contexto adecuado y mantener la consistencia en toda la familia de aplicaciones.

Esto es lo que forma la base para la creación del formato de la pantalla que representa el formato del diseño gráfico y la colocación de iconos, la definición del texto descriptivo en pantalla, la especificación y secundarios del menú. Cuando la imagen y la percepción del sistema coinciden, los usuarios generalmente sienten el gusto con el software y con su funcionamiento.

2.5.2 PROCESO DE DISEÑO DE LA INTERFAZ DE USUARIO

El proceso de diseño de la interfaz de usuario es iterativo y se puede representar mediante un modelo espiral, a continuación en la Figura 2.13 se puede observar que el proceso de diseño de la interfaz de usuario acompaña 4 actividades distintas de marco de trabajo, donde cada una de las tareas anteriores aparecerán más de una vez, en donde que a medida que se avanza por el espiral se representara la elaboración adicional de los requisitos y el diseño resultante.

Figura 2.13 Proceso de diseño de la interfaz de usuario



Fuente: [Roger S. Pressman, 1993]

1. **Análisis de usuario, tareas y entorno:** Se conecta en el análisis del perfil de los usuarios que van a interactuar con el sistema, se registran el nivel de conocimiento, la comprensión del negocio y la receptividad general del nuevo sistema, y se definen diferentes categorías de usuarios.

El análisis de entorno de usuario se centra en el entorno del trabajo físico, entre las preguntas que se formulan son:

¿Dónde se ubicara el físicamente la interfaz?

¿Dónde se situara el usuario?

¿Llevara a cabo tareas no relacionadas a la interfaz?

¿Se adapta bien al hardware a las limitaciones de luz, espacio y ruido?

2. **Diseño de la interfaz:** El objetivo del diseño de la interfaz es definir un conjunto de objetos y acciones de interfaz, que posibiliten al usuario llevar a cabo todas las tareas definidos de forma que cumplan los objetivos de usabilidad definidos por el sistema.

3. **Implementación de la interfaz:** Comienza normalmente con la creación de un prototipo que permita evaluar los escenarios de utilización. El prototipo se basa en la elaboración de modelos o prototipos de interfaz del sistema a medida que avanza el proceso de diseño iterativo y para completar la construcción de la interfaz, se puede utilizar un kit de herramienta de usuario.

4. **La validación de la interfaz:**

La validación se centra en lo siguiente:

- La habilidad de la interfaz, para implementar correctamente todas las tareas de usuario, acoplar todas las variaciones de tareas, y para archivar todos los requisitos generales del usuario.
- El grado de facilidad de utilización de la interfaz y de aprendizaje.
- La aceptación de la interfaz por parte del usuario, como una herramienta útil en su trabajo.

Con todo lo mencionado anteriormente, el diseño de interfaz de usuario para el presente proyecto se realizara de la siguiente manera.

1. Redacción: Utilizar un lenguaje de usuario evitando el lenguaje técnico, para referirse a los conceptos que contiene el formulario.
2. Texto: Colocar en mayúscula la primera letra de la palabra, está ayuda a identificar el inicio de una palabra o etiqueta. Evitar abreviaciones de textos para que el usuario no se confunda.
3. Ubicación: La posición de los textos se ubicaran a la izquierda del campo o del área de texto, debe situarse preferiblemente a la izquierda y no a la derecha ni debajo y los botones de opción.
4. Alineación: Las etiquetas se alinearan a la izquierda, la alineación a la izquierda ayuda al escaneo del visual por lo cual facilita su localización.
5. Los textos de las etiquetas y de los campos estarán alineados horizontalmente.
6. Separación con el campo de control: Se adopta la alineación a la derecha todas las etiquetas de verificación y tendrán la misma separación de su campo o control.
7. Estado: La etiqueta debe reflejar el estado del campo o control al que está asociado.
8. Interacciones: En las etiquetas de las casillas de verificación y de los botones de opción, el texto será duable al activar a la misma acción de su control.
9. Vínculos: Se utiliza para la conexión de páginas y hacen posible que los usuarios accedan a otras páginas, existen tipos de vínculos como ser: El vinculo de navegación estructural, se refiere a los botones de inicio y botones de paginas subordinados a otras paginas, Los vínculos asociados a la pagina, suelen ser palabras subrayadas que señalan a otras paginas.
10. Marcos: El uso de los marcos permitirá dar la información a los usuarios si el URL sigue funcionando, viene determinada con una secuencia de acciones de desplazamiento.
11. Gráficos: Los gráficos se ubicaran en la página de manera adecuada posible, para tener una óptima resolución.

2.6 FACTORES DE CALIDAD ISO 9126

La calidad del software se define como la concordancia de los requisitos funcionales y de rendimiento explícitamente establecido, con los estándares de desarrollo, por lo cual estas métricas se basan en el borrador de estándar ISO/IEC 9126.

El ISO/IEC 9126 ha sido desarrollado en un intento de identificar los atributos clave de calidad para el software, la característica del ISO 9126 son:

- Funcionalidad
- Facilidad de mantenimiento
- Portabilidad
- Confiabilidad
- Usabilidad

FUNCIONALIDAD

La métrica de Función definida por Alan Albert, de IBM el 1979, es un método para medir el tamaño del software. Pretende medir la funcionalidad entregado al usuario final independiente de la tecnología utilizada para la construcción y explotación del software, y también ser útil en cualquiera de las fases de vida del software, desde el diseño inicial hasta la explotación y mantenimiento.

La métrica punto función (PFs), es una métrica orientada a la función y sugiere un acercamiento a la medida de productividad. Los puntos de función se obtienen utilizando una relación empírica basada en medidas cuantitativas del dominio de información de software y valoraciones subjetivas de la complejidad del software.

Para determinar la funcionalidad del sistema primero se debe determinar cinco características del dominio de información.

- Numero de Entradas de Usuario
- Numero de Salidas de Usuario
- Numero de Peticiones de Usuario (Consultas)
- Número de Archivos
- Numero de Interfaces externas

La Formula para calcular la funcionalidad está dada de la siguiente forma:

$$PF = \text{Cuenta Total} * (X + Y * \text{SUM Fi})$$

Donde:

CT: Suma de las entradas obtenidas de la tabla

X= Nivel de confiabilidad del sistema

Y= Nivel de error igual a 0.01

Fi: Valores de ajuste de Complejidad

Para obtener el cálculo del factor de ajuste que está dado por: Fi, Grado de influencia se evalúa en un rango de 0 a 5.

0= La característica no está presente o no influye si está presente

1= La característica tiene una influencia incidental

2= La característica tiene una influencia moderada

3= La característica tiene una influencia promedio

4= La característica tiene una influencia significativa

5= La característica tiene una influencia fuerte, es esencial

PORTABILIDAD

El esfuerzo requerido para transferir el programa desde un hardware y/o un entorno de sistema del software en un ambiente operacional. Es decir la fiabilidad con que el software puede ser llevado de un entorno a otro, facilidad de ajuste y la facilidad de adaptación al cambio.

MANTENIBILIDAD

La facilidad de mantenimiento, es óptimo es decir que se puede realizar correcciones en el programa sin alterar los demás programas. El mantenimiento correctivo es donde se puede corregir las observaciones, fallas cuando el usuario ha detectado errores en el sistema durante su funcionamiento, para dar soluciones futuras.

CAPITULO III

MARCO APLICATIVO

Este capítulo, tiene el propósito de demostrar el desarrollo del sistema, con el objeto de identificar y resolver los problemas que aquejan a dicha institución, en cuanto al control y seguimiento físico financiero de ejecución de proyectos.

También se detallará las fases del proceso unificado de desarrollo (RUP), el mismo responde a los requerimientos del lenguaje unificado de modelado UML.

El proceso unificado de desarrollo comprende de 4 fases:

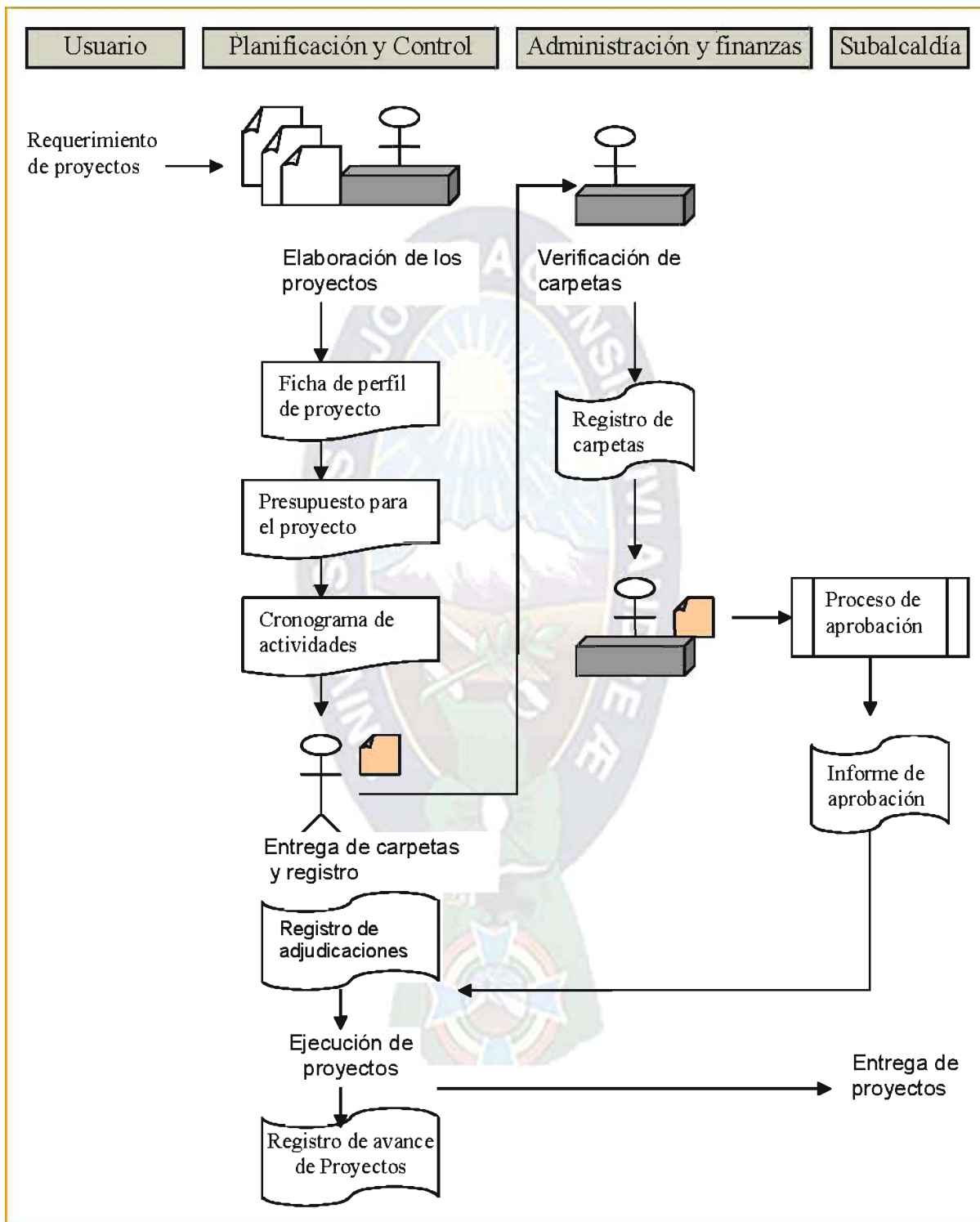
- Fase de Inicio
- Fase de elaboración
- Fase de construcción
- Fase de transición

Primeramente se explicará de forma clara los aspectos relacionados con las características de la organización, descripción de funciones y los diferentes procesos que existen en dicha unidad.

3.1 ANALISIS DEL SISTEMA ACTUAL

En esta etapa se analiza la profundidad del flujo de trabajo para determinar los requerimientos, describiendo la estructura orgánica y física dentro del organigrama de la sub alcaldía, como las actividades de ejecución de proyectos, ver la organigrama de la institución en el **[Anexo A]**.

Figura 3.1 Proceso de control y seguimiento de proyectos



Fuente: Elaboración propia

DESCRIPCION DE FUNCIONES

A continuación se describe las funciones más importantes:

- Coordinar la planificación de los recursos con la junta de vecinos.
- Supervisar la elaboración de carpetas de inversión.
- Coordinar la elaboración del plan operativo anual.
- Supervisar el avance de los proyectos.
- Coordinar la elaboración de los informes oficiales.
- Elaboración de los informes anuales de actividades.
- Elaborar, actualizar y evaluar los avances de los proyectos.
- Realizar el seguimiento de la ejecución física y financiera.

Ver la descripción de procesos y funciones con más detalle en el **[Anexo B]**.

3.2 FASE DE INICIO

En esta fase, se realiza un estudio de los requerimientos desde el punto de vista del usuario, identificando todas las entidades externas para determinar el modelo de negocio.

3.2.1 REQUERIMIENTOS

El propósito de este punto es identificar y definir las necesidades y las características del sistema actual, en la Tabla 3.1 muestra los requerimientos funcionales y no funcionales de los usuarios.

Es de esa forma que el presente proyecto permitirá la captura de toda la información, la cual dará versatilidad al registro de cada uno de los informes emitidos que ingresa a dicha institución, es decir el usuario final podrá ingresar, actualizar los datos y obtener informes en un tiempo mínimo, además de visualizar el porcentaje de avance de los proyectos.

Tabla 3.1 Requerimientos funcionales y no funcionales

Requerimientos	Descripción
Requerimientos Funcionales	
R1.1	Registro de los requerimientos y aprobación de proyectos con todas sus observaciones.
R1.2	Registro de la nota de adjudicaciones para su posterior impresión.
R1.3	Registro del seguimiento de ejecución de los proyectos de acuerdo al presupuesto asignado.
R1.4	Registro de control del porcentaje de avance físico y financiero de los proyectos, además de sus respectivas cronogramas durante un tiempo determinado.
R1.5	Reporte general del seguimiento y control de ejecución de los proyectos, incluyendo los reportes de certificación, para que sirvan de respaldo de la existencia de un proyecto.
Requerimientos no funcionales	
R 1.7	Software: ASP.NET para generar aplicaciones que utilizan y exponen servicios Web XML.
R 1.8	Gestor de Base de Datos: SQL Server 2000
R 1.9	Hardware: Equipos Windows 2000 para adelante, con microprocesador de 700 Mhz, un servidor central, memoria RAM 128 Magabytes.
R 1.10	Plataforma: La plataforma de software puede ser ejecutado en cualquier plataforma Windows 2000 a Windows XP.

3.2.2 MODELO DE NEGOCIO

Un modelo de negocio nos permite capturar los requisitos correctos y construir el sistema, de acuerdo a las necesidades de los usuarios.

Un modelo de negocio se desarrolla en los siguientes pasos:

- Modelo de casos de uso de negocio, se identifican los actores y los casos de uso, mediante los diagramas de casos de uso.

- Modelo de actividades del negocio, representado mediante diagramas de iteración o por diagramas de actividades.

3.2.2.1 IDENTIFICACIÓN DE ACTORES Y CASOS DE USO

En esta parte se describen los actores y los casos de uso más esenciales, además de definir los roles que cumplen cada actor o usuario.

En la siguiente Tabla 3.2 se muestra el comportamiento y la función que cumple un actor.

Tabla 3.2 Descripción de negocio. Unidad de planificación y control

ACTOR	UNIDAD DE PLANIFICACIÓN Y CONTROL
Descripción	Es el encargado de llevar la actividad de control y planificación, su función consiste de controlar la ejecución.
Características	<ul style="list-style-type: none"> - Este actor se encarga de controlar los proyectos. - Realiza solicitudes de avance tanto físico como financiero. - Este puede asumir la responsabilidad de supervisor. - Supervisar la elaboración de carpetas de inversión - Coordinar la elaboración del plan operativo anual.
Relaciones	<p>Casos de uso: Ejecución de proyecto, avance físico financiero de proyectos y Consultas</p> <p>Actores: Unidad de Planificación y Control, Unidad de administración y finanzas</p>

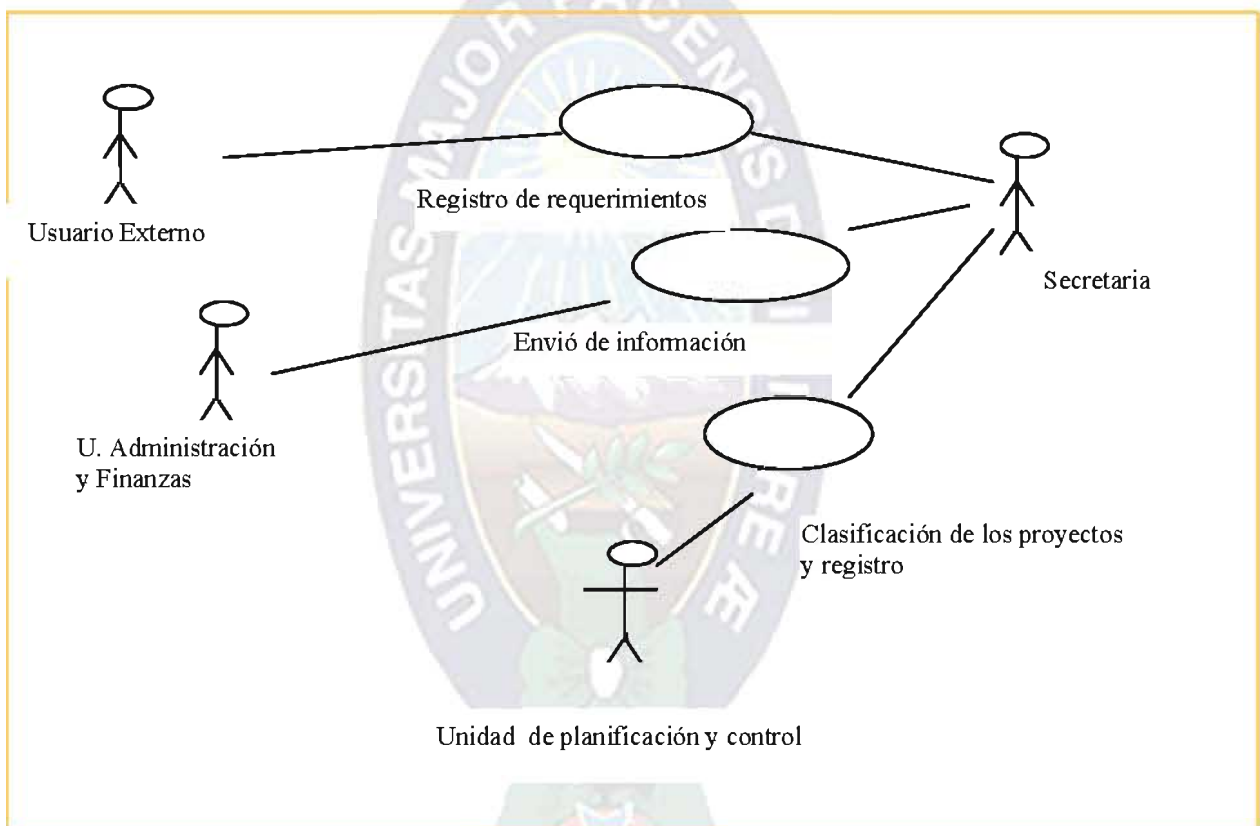
Fuente: Elaboración Propia

Se describe con más detalle los demás actores en el **[Anexo B]**.

3.2.2.2 MODELO DE CASOS DE USO DE NEGOCIO

En la Figura 3.2 se describe el modelo de casos de uso de negocio del modulo de clasificación de proyectos donde la secretaria general cumple la función de registrar los requerimientos y el envío respectivo de la información a las unidades.

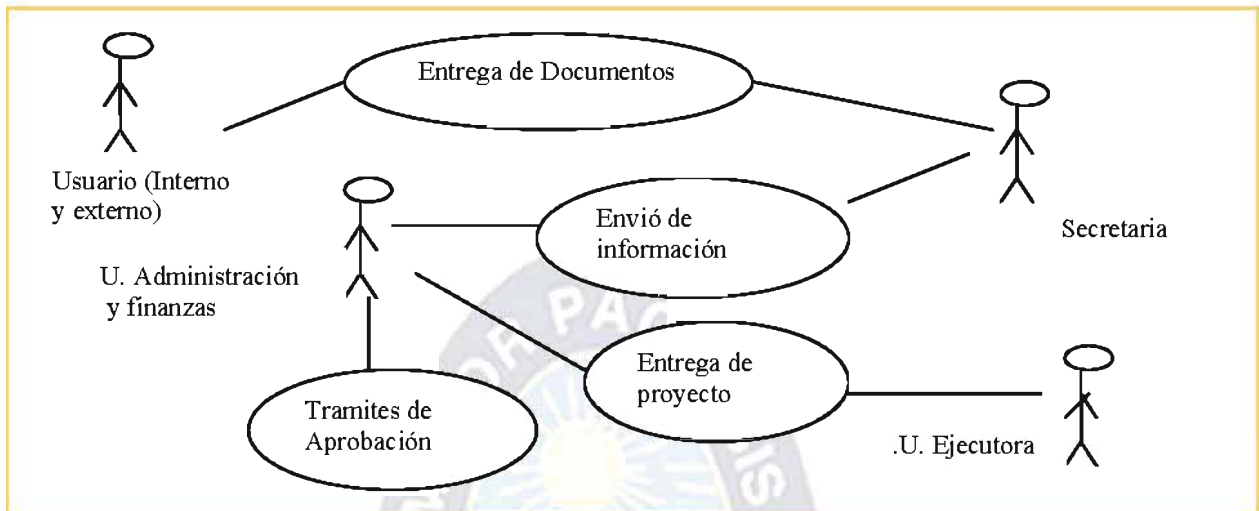
Figura 3.2 Modulo de organización de proyectos



Fuente: Elaboración propia

En la Figura 3.3 muestra el modelo de casos de uso del modulo de aprobación de proyectos, donde se realiza la verificación de los documentos para dicha aprobación.

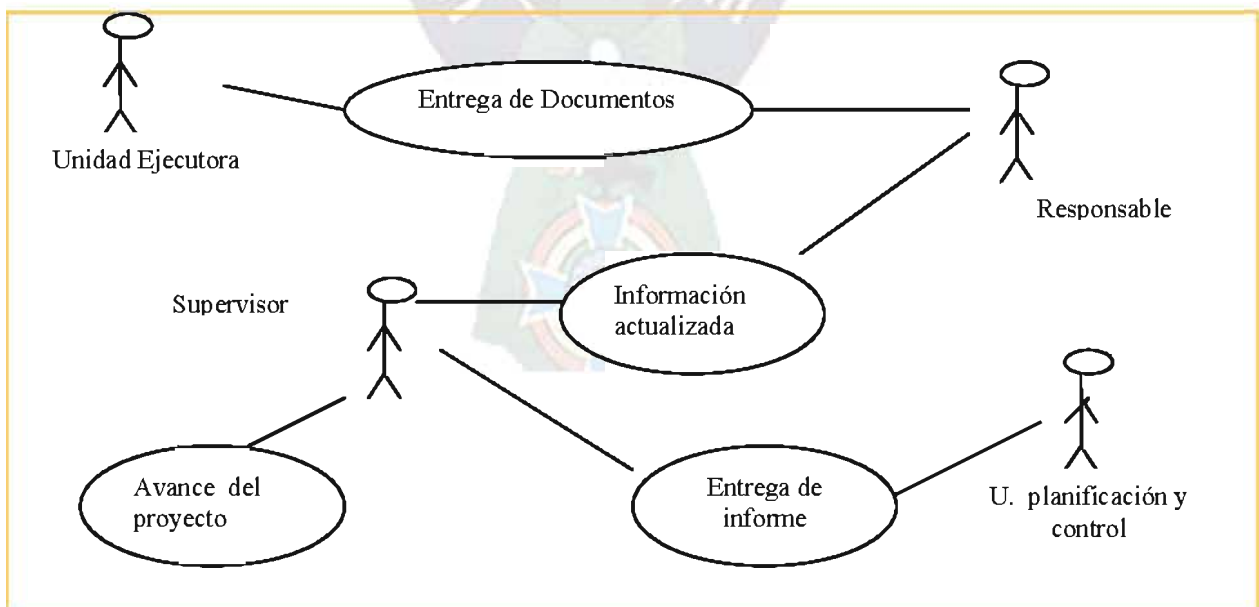
Figura 3.3 Modulo de aprobación de proyectos



Fuente: Elaboración propia

En la Figura 3.4 se observa el modelo de caso de uso del modulo de ejecución de proyectos, donde la unidad ejecutora entrega los documentos al responsable y el responsable al supervisor quien es el encargado del registro de los avances y entrega de los informes de la ejecución de los proyectos.

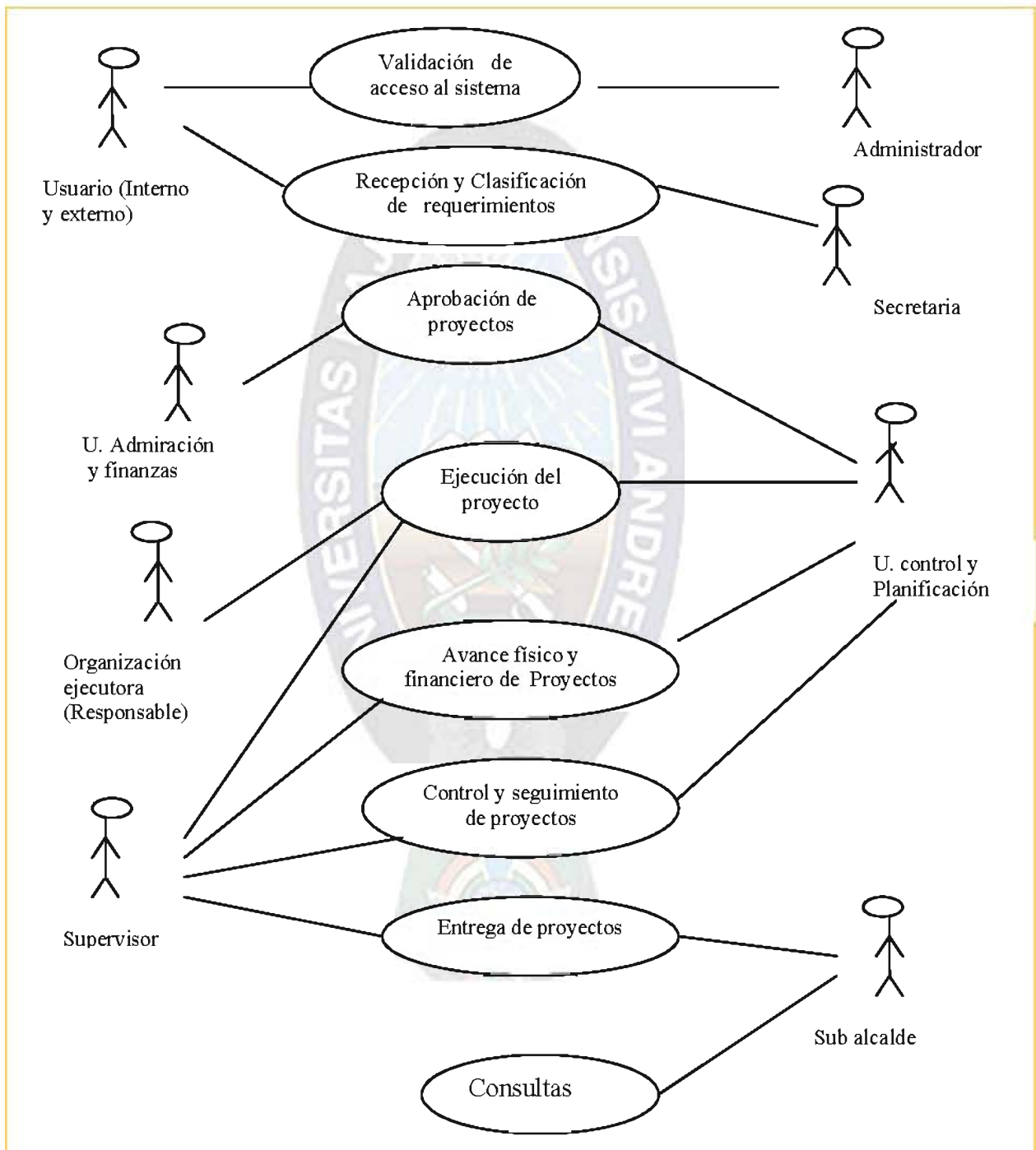
Figura 3.4 Ejecución de proyectos



Fuente: Elaboración propia

MODELO DE CASO DE USO

Figura 3.5 Modelo de negocio de casos de uso



Fuente: Elaboración propia

3.2.3 IDENTIFICACION DE CASOS DE USO

La identificación de caso de uso es la interacción que tiene el actor con el sistema, como se muestra en la siguiente Tabla 3.3

Tabla 3.3 Casos de uso: Ejecución de proyectos

Caso de uso	Ejecución de proyectos	
Actor	Responsable , supervisor , Unidad de planificación y control	
Propósito	Realizar el registro de los proyectos en ejecución y el cronograma de actividades.	
Resumen	Genera informes de actividades para dicho proyecto.	
Tipo	Primario	
Flujo principal	Acción del actor	Respuesta del sistema
	1. El responsable activa el formulario de proyectos. 3. Registra el nuevo proyecto que se está ejecutando. 5. Posteriormente revisa si los datos están correctamente y luego hace clic en confirmar. 7. Realiza una petición de la revisión de cronograma de actividades. 9. Procede a reprogramar el cronograma de actividades no cumplidos hasta la fecha. 11. Para salir de la página, clic en el botón salir.	2. El sistema despliega el formulario. 4. El sistema verifica los valores si están correctos. 6. El sistema almacena los datos correspondientes, en la base de datos. 8. El sistema visualiza el cronograma de actividades. 10. Almacena en la base de datos, con la nueva reprogramación de actividades. 13. El sistema cierra el menú, y vuelve a la página principal
Precondición	El responsable debe pertenecer a la unidad de planificación y control y tener permiso de realizar el registro correspondiente.	
Poscondición	Registrar el proyecto que se está ejecutando	

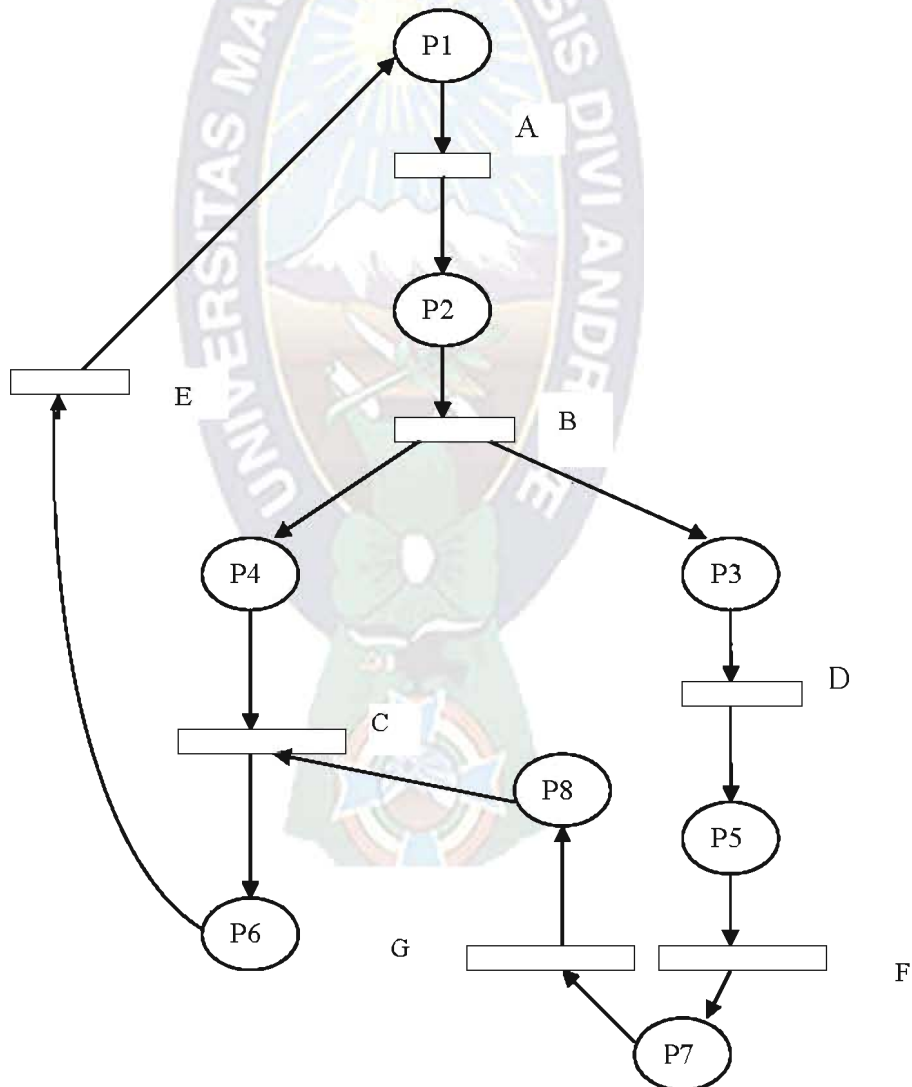
En el **[Anexo C]** se describen los casos de uso extendidos, que detallan las interacciones de los casos de uso con el sistema.

3.2.4 MODELADO DE REDES PETRI

A continuación se modelará dos casos de uso aplicando redes petri, que es una herramienta que nos permitirá identificar de manera adecuada las plazas y transiciones, para definir de manera precisa y operativa.

En la siguiente Figura 3.6 se observa el modelo de control de carpetas de los proyectos, aplicando las redes de petri, identificando las precondiciones y poscondiciones para elaborar la secuencia de marcados.

Figura 3.6 Control de carpetas de los proyectos



Fuente: Elaboración propia

A continuación se identificara las plazas y las transiciones:

Plazas:

- P1 = Responsable
- P2 = Verificación de la información de los proyectos
- P3 = Proyecto en licitación
- P4 = Pedido en espera, no aprobado
- P5 = Proyecto adjudicado
- P6 = Informe elaborado
- P7 = Proyecto en ejecución
- P8 = Proyecto finalizado

Transiciones:

- A = Pedido de información de los proyectos
- B = Pedido de aprobación de los proyectos
- C = Pedido de registro y elaboración del informe
- D = Pedido de adjudicación del proyecto
- E = Emite informe de control
- F = Realización de ejecución de proyecto
- G = Pedido de registro de ejecución

Procedemos a hallar la poscondición de la siguiente manera:

Precondición:

	A	B	C	D	E	F	G
P1	1	0	0	0	0	0	0
P2	0	1	0	0	0	0	0
P3	0	0	0	1	0	0	0
P4	0	0	1	0	0	0	0
P5	0	0	0	0	0	1	0
P6	0	0	0	0	1	0	0
P7	0	0	0	0	0	0	1
P8	0	0	0	0	0	1	0
P9	0	0	1	0	0	0	0

Poscondición:

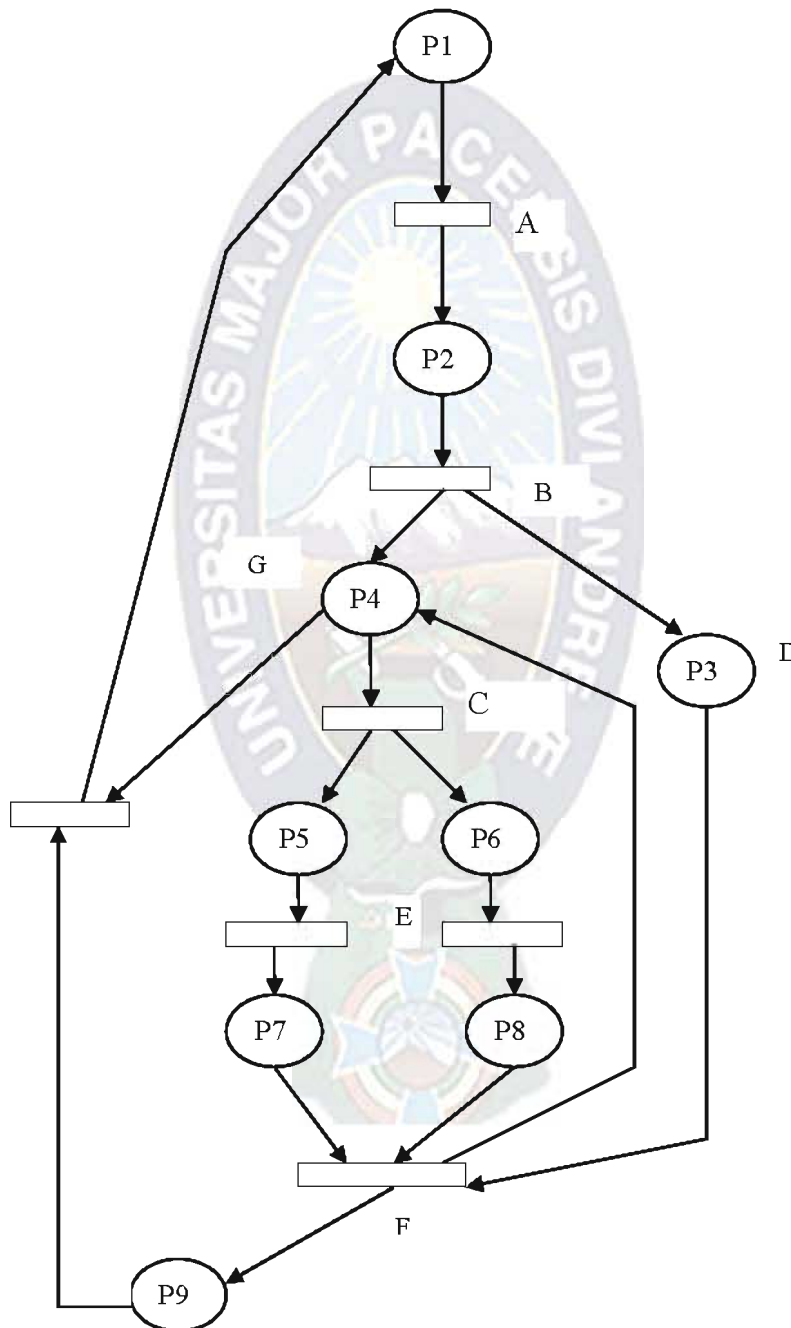
	A	B	C	D	E	F	G
P1	0	0	0	0	1	0	0
P2	1	0	1	0	0	0	0
P3	0	1	0	0	0	0	0
P4	0	1	0	0	0	0	0
P5	0	0	0	1	0	0	0
P6	0	0	1	0	0	0	0
P7	0	0	0	1	0	0	0
P8	0	0	0	0	0	1	0
P9	0	0	0	0	0	0	1

Por tanto el resultado de marcado será lo siguiente:

M0 = 10000000, M1 = 01000000, M2 = 00110000, M3 = 00011000, M4 = 00010010
 M5 = 00010001 y M6 = 00000100.

A continuación se observa el modelo de control de avance físico y financiero de los proyectos, aplicando redes petri, donde las transiciones son condiciones que nos permitirá pasar de una plaza a otra, el la Figura 3.7 se muestra el modelado de control físico y financiero de los proyectos.

Figura 3.7 Control de avance físico y financiero de los proyectos



Fuente: Elaboración propia

A continuación se identificara las plazas y las transiciones:

Plazas:

- P1 = Responsable
- P2 = Recibe la lista de los proyectos
- P3 = Pedido en espera o proyecto no existente
- P4 = Verificación de datos realizados
- P5 = Pedido de control financiero realizado
- P6 = Elabora el control físico
- P7 = Control de avance físico registrado
- P8 = Elabora el control físico
- P9 = Pedido de elaboración de informe realizado

Transiciones:

- A = Pedido de información de los proyectos
- B = Realizar la verificación de los datos
- C = Pedido de registro de control
- D = Pedido de actualización de avance financiero
- E = Pedido de actualización de avance físico
- F = Pedido de elaboración de informe de avance
- G = Emite el informe

Procedemos a hallar la poscondición de la siguiente manera:

Precondición:

	A	B	C	D	E	F	G
P1	1	0	0	0	0	0	0
P2	0	1	0	0	0	0	0
P3	0	0	0	0	0	1	0
P4	0	0	1	0	0	0	1
P5	0	0	0	1	0	0	0
P6	0	0	0	0	1	0	0
P7	0	0	0	0	0	1	0
P8	0	0	0	0	0	1	0
P9	0	0	0	0	0	0	1

Poscondición:

	A	B	C	D	E	F	G
P1	0	0	0	0	0	0	1
P2	1	0	0	0	0	0	0
P3	0	1	0	0	0	0	0
P4	0	1	0	0	0	0	0
P5	0	0	1	0	0	0	0
P6	0	0	1	0	0	0	0
P7	0	0	0	1	0	0	0
P8	0	0	0	0	1	0	0
P9	0	0	0	0	0	1	0

Por tanto el resultado de marcado será lo siguiente:

M0 = 100000000, M1 = 010000000, M2 = 001100000, M3 = 001011000, M4 = 001010010
 M5 = 001001100, M6 = 001000110 y M7 = 000000001.

3.3 FASE DE ELABORACIÓN

3.3.1 ANÁLISIS

Terminado el proceso de identificación de requerimientos del sistema y el estudio de los casos de uso, se procederá a la identificación de clases de objetos que intervienen en el sistema.

3.3.1.1 ANÁLISIS DE ARQUITECTURA

El propósito de análisis de la arquitectura es esbozar el modelo de análisis y la arquitectura mediante la identificación de paquetes del análisis, clases de análisis evidentes y requisitos especiales.

IDENTIFICACIÓN DE PAQUETES DE ANÁLISIS A PARTIR DE CAOS DE USO:

Figura 3.8 Identificación de paquetes de análisis Recepción y clasificación de requerimientos

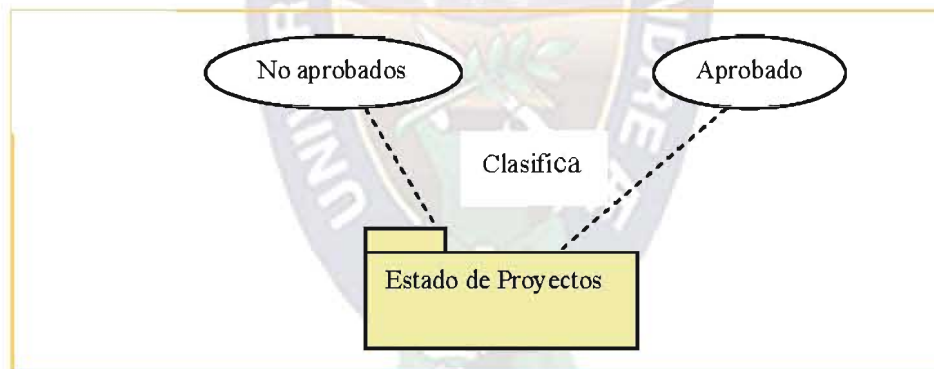
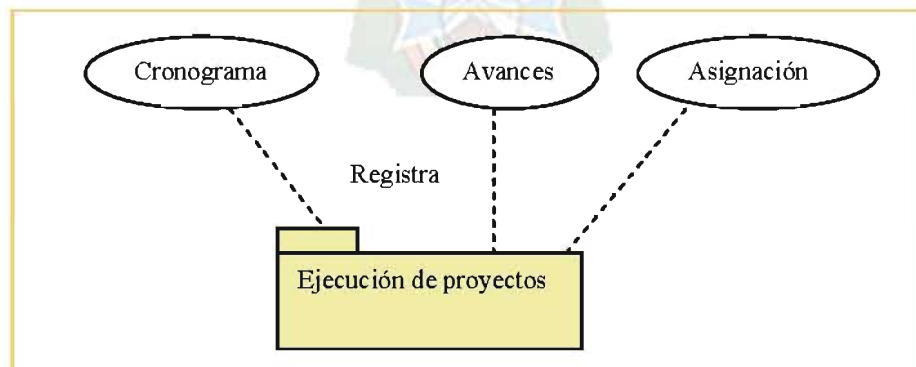


Figura 3.9 Identificación de paquetes de análisis Ejecución de proyectos



3.3.1.2 ANÁLISIS DE CASOS DE USO

Los casos de uso, permiten identificar las clases de análisis cuyos objetos son necesarios para llevar a cabo el flujo de sucesos de los casos de uso. La clase de análisis se centra en el tratamiento de los requisitos funcionales y no funcionales de los casos de uso, además de la identificación de clases de control, entidad e interfaces necesarias, para ello utilizaremos diagramas de clase y diagramas de colaboración.

A. DIAGRAMAS DE CLASE

En la Figura 3.10 del diagrama de clase de una realización de caso de uso permitirá facilitar el registro de requerimientos y su respectiva clasificación almacenada en carpetas y a los formularios para su posterior impresión.

Figura 3.10 Diagrama de clase de una realización de Caso de uso
Recepción y clasificación de requerimientos

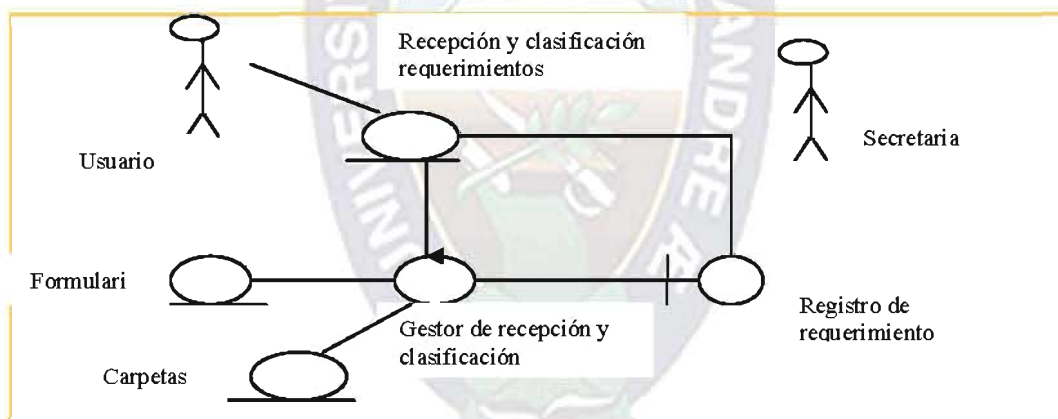
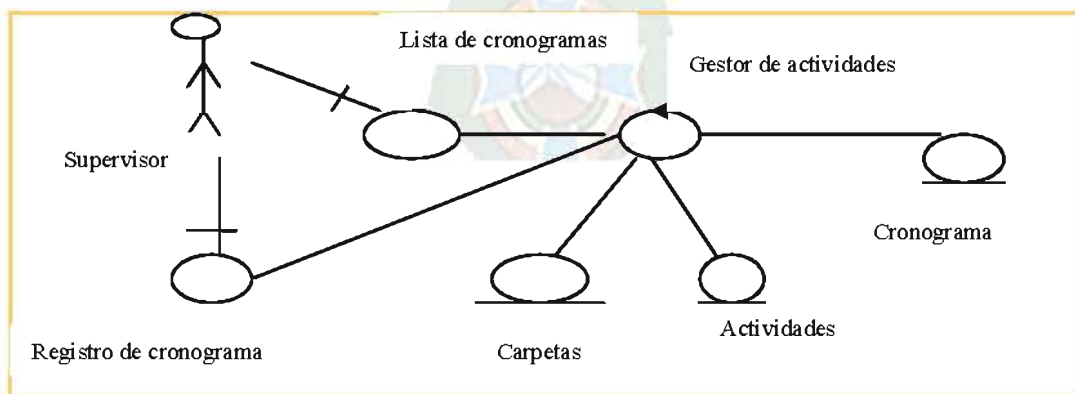


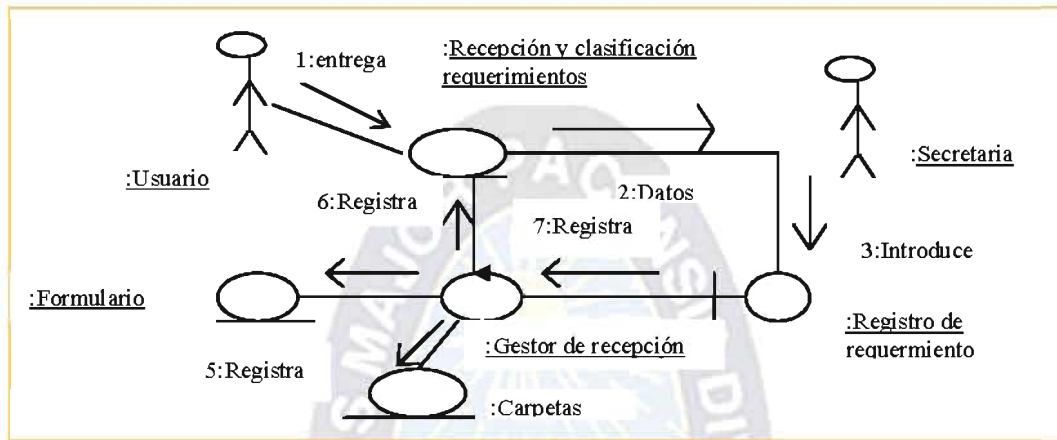
Figura 3.11 Diagrama de clase de Caso de uso de ejecución de proyectos



Ver en el [Anexo D], los diagramas de clases de casos de uso.

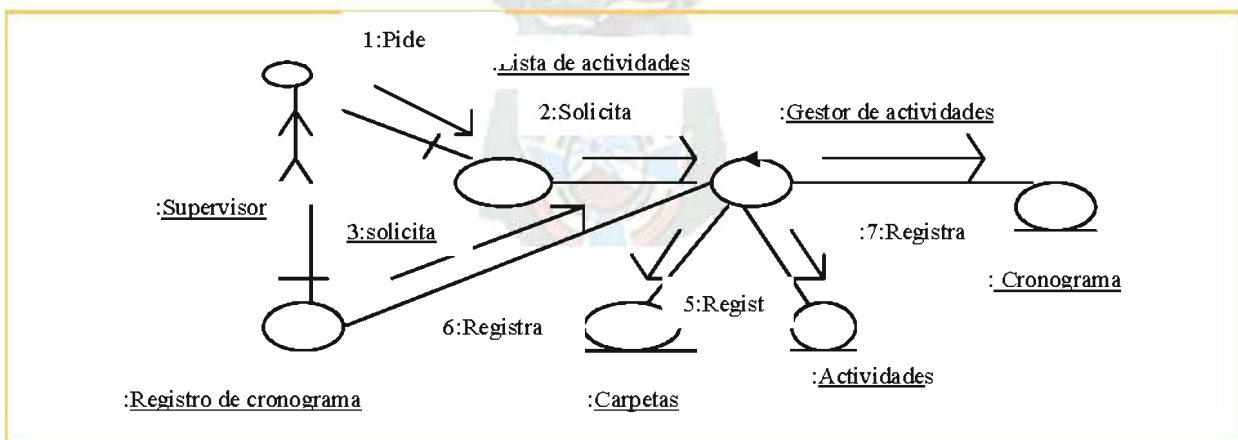
B. DIAGRAMA DE COLABORACIÓN

Figura 3.12 Diagrama de colaboración de una realización del Caso de uso
Recepción y clasificación de requerimientos



En la Figura 3.13 facilita el registro de la ejecución de proyectos donde permite minimizar el tiempo de búsqueda de los proyectos, las carpetas y es almacenada la información de ejecución de proyecto, cronograma de actividades desde el inicio hasta la finalización de actividades, en caso contrario su reprogramación.

Figura 3.13 Diagrama de colaboración de una realización del Caso de uso
Ejecución de proyectos



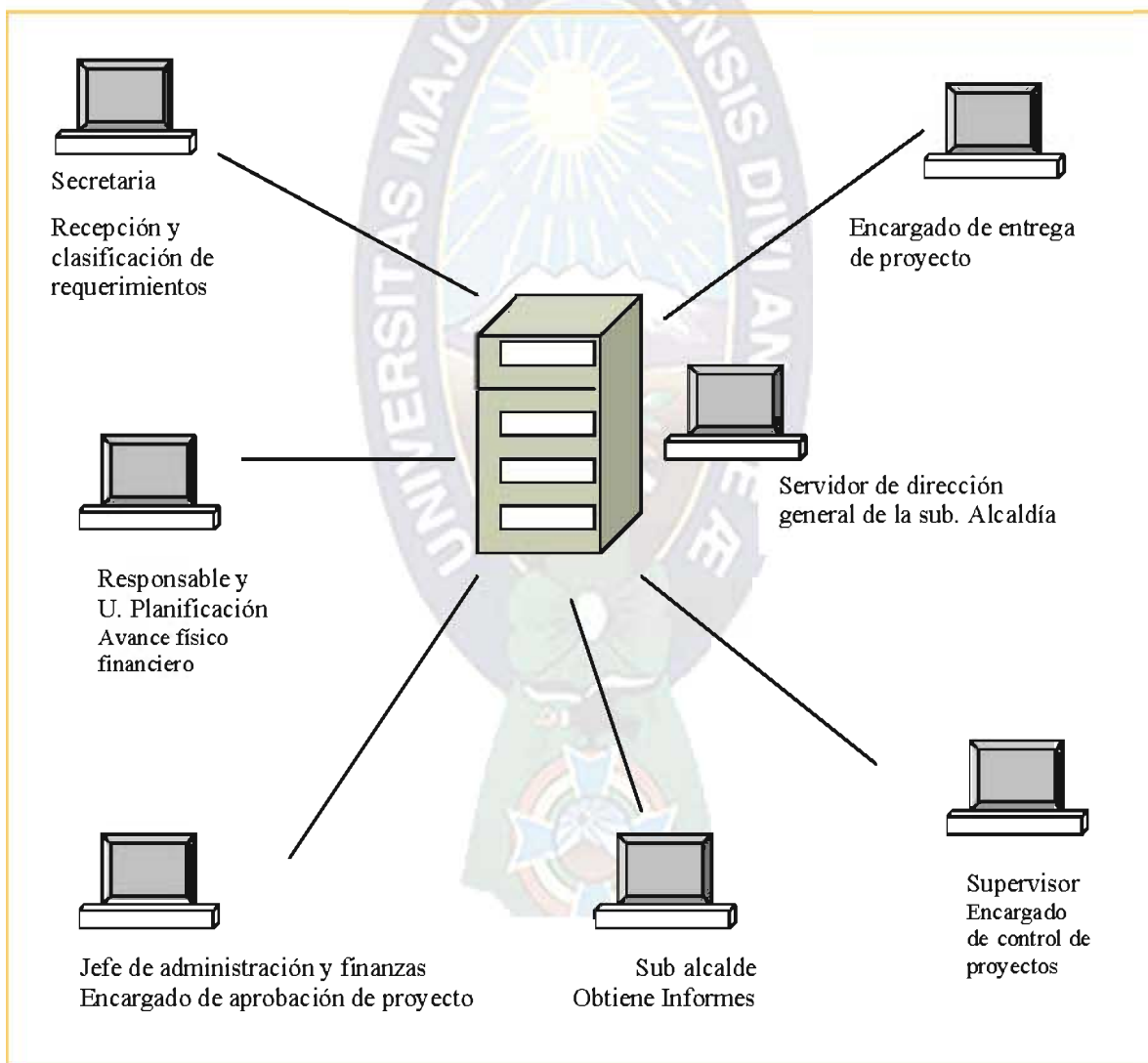
Ver en el [anexo D] los demás diagramas de colaboración de los casos de uso.

3.3.2 DISEÑO

3.3.2.1 DISEÑO DE ARQUITECTURA

El objetivo del diseño de la arquitectura es esbozar los modelos de diseño y despliegue y su arquitectura mediante la identificación de elementos, utilizando la identificación de nodos y la configuración de la red.

Figura 3.14 Diseño de control y seguimiento físico financiero de ejecución de proyectos



Fuente: Elaboración propia

3.3.2.2 DISEÑO DE CASOS DE USO

El diseño de casos de uso del sistema describe como se realiza y se ejecuta un caso de uso, a partir de identificar las clases de diseño, distribuir el comportamiento de los casos de uso entre los objetos de diseño y capturar los requisitos de la implementación de los casos de uso. A continuación se presentan los diagramas relacionados con casos de uso que permiten visualizar las interacciones.

A. DIAGRAMA DE CLASES DE DISEÑO DE CASO DE USO

Figura 3.15 Diagrama de clases de diseño de caso de uso
Recepción y clasificación de requerimientos

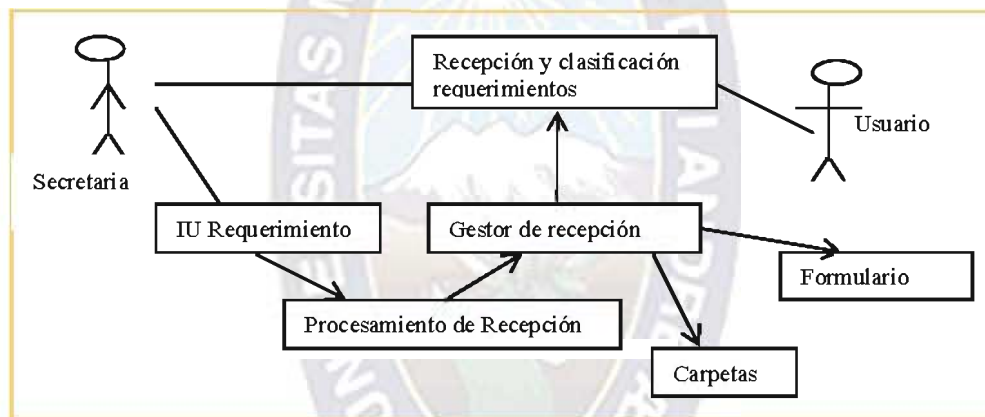
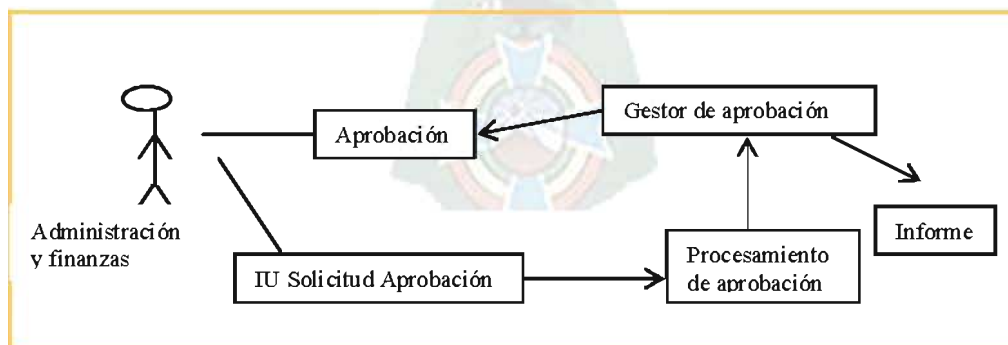


Figura 3.16 Diagrama de clases de diseño de caso de uso
Aprobación de proyectos



Ver en el [anexo D] los demás Diagrama de clases de diseño de caso de uso

B. DIAGRAMA DE SECUENCIA

En la Figura 3.17 se observa el proceso de los diagramas de Secuencia de la recepción y clasificación de requerimientos, es decir el usuario activa el interfaz del sistema autenticándose, luego se realiza verificación en el gestor de recepción y posteriormente almacenando en las carpetas respectivas.

Figura 3.17 Diagrama de Secuencia de diseño de caso de uso
Recepción y clasificación de requerimientos

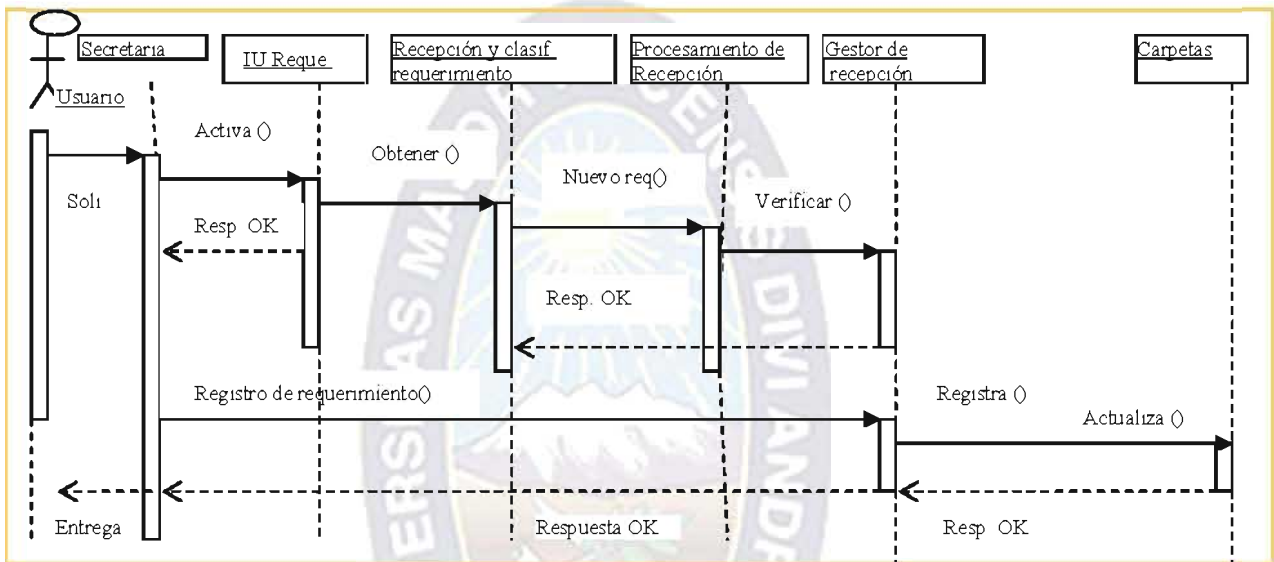
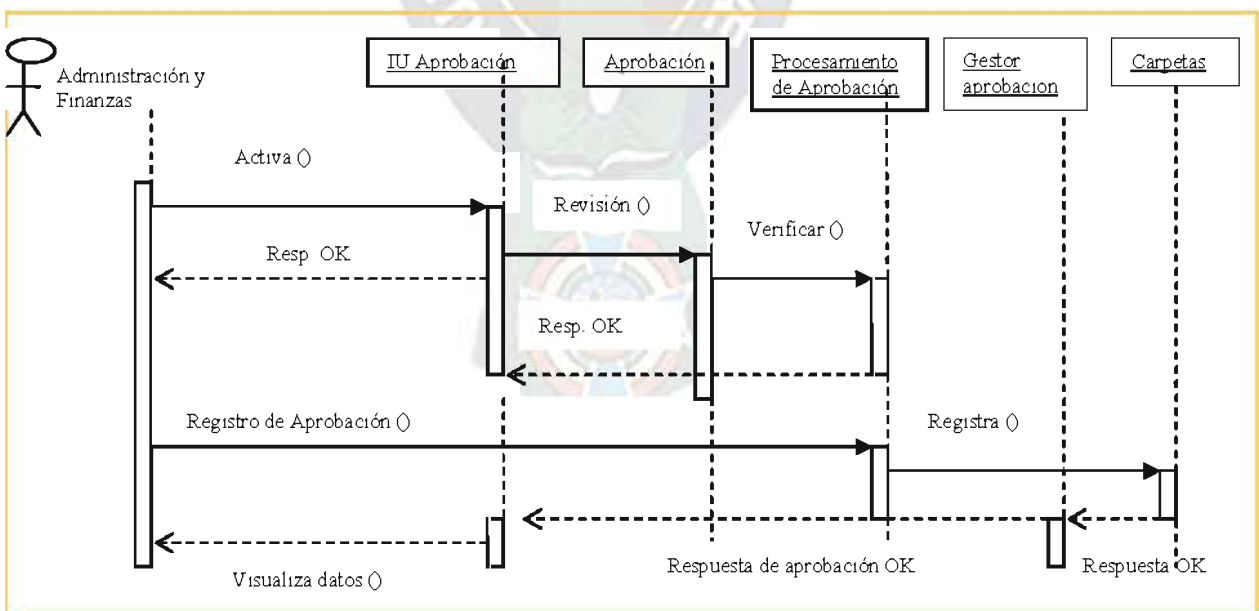


Figura 3.18 Diagrama de Secuencia de diseño de caso de uso Aprobación de proyectos



Los demás Diagrama de secuencia de diseño de caso de uso, ver en el [anexo D].

C. DIAGRAMA DE ESTADOS

Los diagramas de estado representan la secuencia por la que un objeto pasa durante su tiempo de vida, teniendo como característica los eventos que hacen posible la transición de un estado a otro, en la Figura 3.19 se muestra el registro de los requerimientos.

Figura 3.19 Diagrama de Estados para la clase Registro de requerimientos

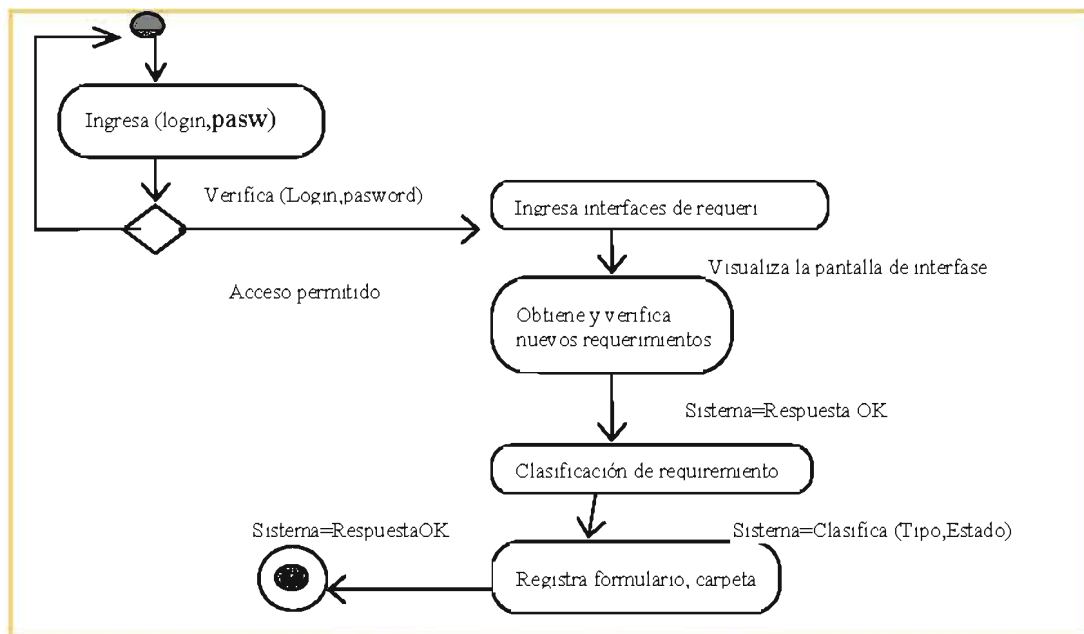
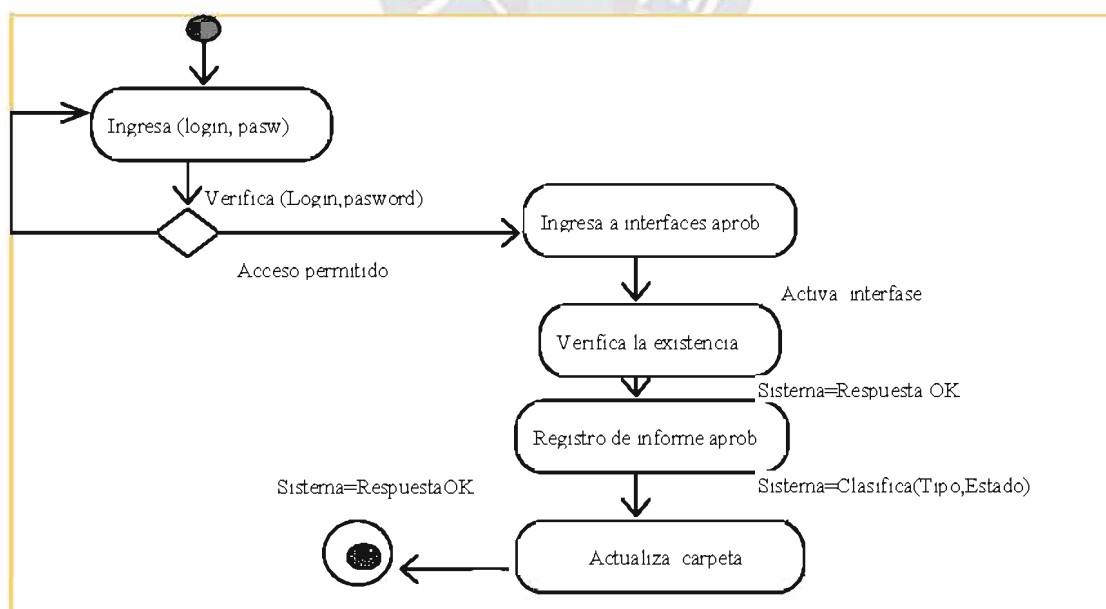


Figura 3.20 Diagrama de Estados para la clase Aprobación de proyecto



Ver en el [anexo D] los demás diagramas de estados.

3.3.2.3 DISEÑO DE CLASES

En el diagrama de clases de diseño para el sistema, se identifica la estructura de clases, relaciones, atributos y métodos, como muestra en la siguiente Figura 3.21.

Figura 3.21 Diagrama de clases de Diseño de Requerimiento y clasificación de los proyectos

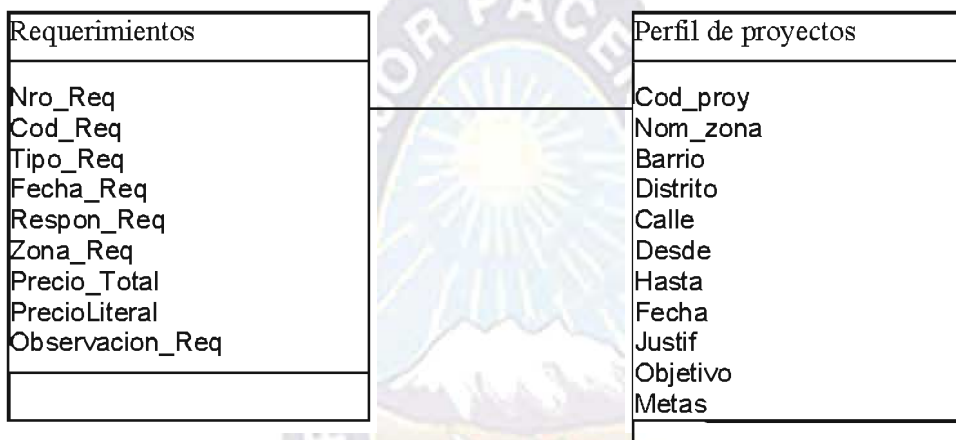


Figura 3.22 Diagrama de clases de Diseño de Licitación y Adjudicación de los proyectos

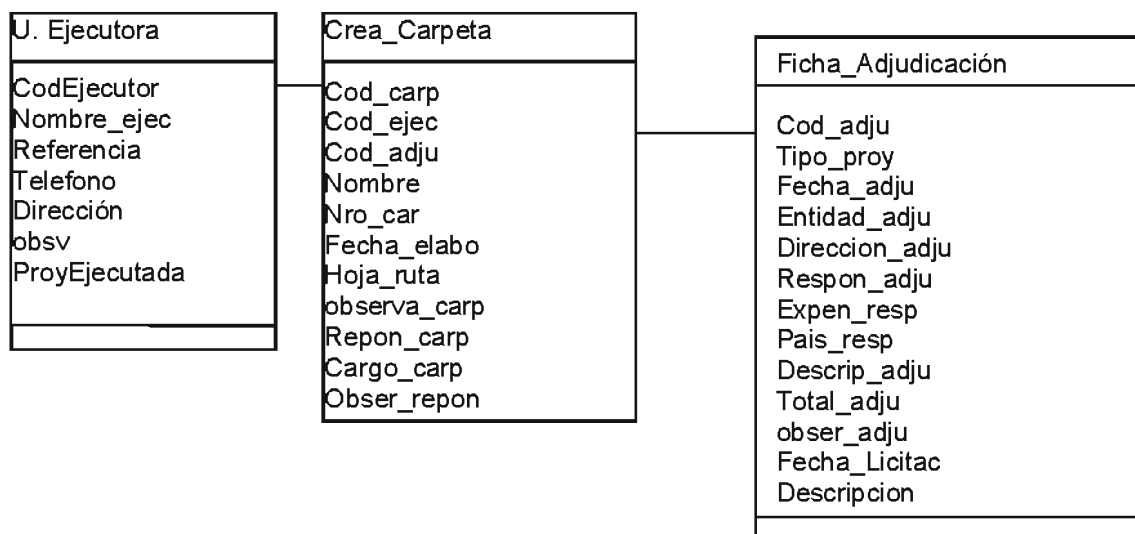


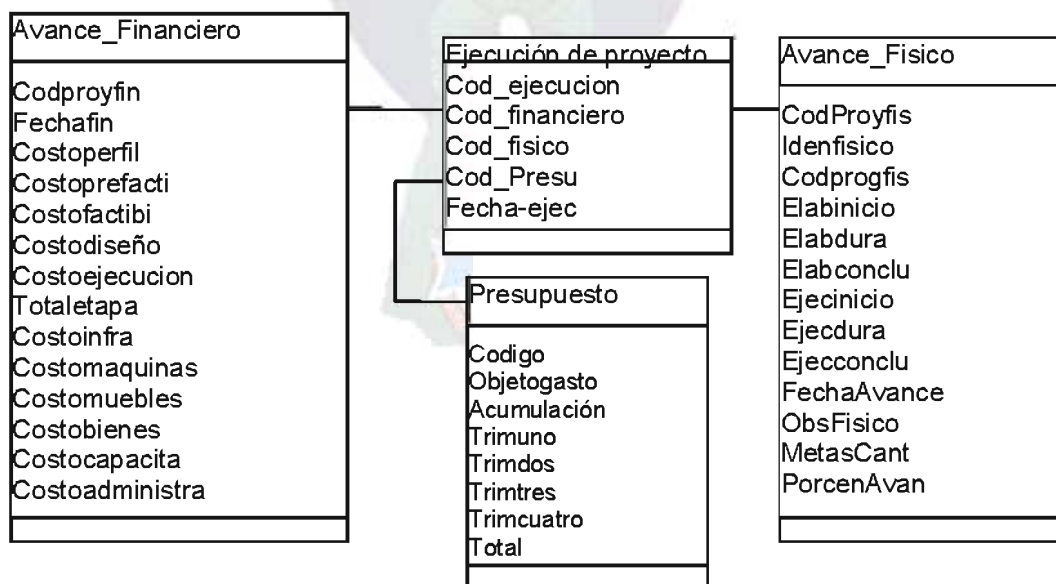
Figura 3.23 Diagrama de clases de Diseño de Validación de acceso al sistema



Figura 3.24 Diagrama de clases de Diseño de Aprobación de los proyectos



Figura 3.25 Diagrama de clases de Diseño de Avance físico y financiero



3.4 FASE DE CONSTRUCCIÓN

En está fase del proyecto se considera los aspectos de construcción y/o implementación del software, así mismo la interfaces para la creación de los prototipos, es decir el diseño de interfaz que se realizo íntegramente de los casos de uso identificados.

DISEÑO DE INTERFACES

El diseño y la implementación de la interfaz del nuevo sistema, se realizo de acuerdo al método espiral y a las reglas que propone Theo Mantel [MAN97], mencionados en el capítulo II del marco teórico, mediante lo cual se pueda satisfacer la necesidad de los usuarios y facilitar el manejo del sistema.

Restricciones de seguridad

Cuando se implementa un software, es necesario que se establezcan medidas de seguridad, para mejorar el manejo de las informaciones, es por eso que el sistema cuenta con un modulo de autenticación como se observa en la Figura 3.26, que permitirá al usuario ingresar a las informaciones que le sean permitidas, introduciendo el código del usuario y su contraseña respectiva, cada usuario tiene su nivel de permiso de registro y elaboración de informes, si la contraseña no es valida no podrá ingresar al sistema.

Figura 3.26 Ventana de Autenticación

Fuente: Elaboración propia

Pantalla principal

Una vez que el usuario a realizado el ingreso de su código y la contraseña, es decir logrado autenticarse correctamente en el sistema, permitirá el ingreso a la pagina principal donde podrá acceder a diferentes opciones del menú del sistema, con el fin de elaborar los nuevos registros de proyectos y realizar las respectivas actualizaciones.

Figura 3. 27 Ventana principal del sistema



Fuente: Elaboración propia

Submenú del registro de proyectos

La siguiente pantalla visualiza el submenú de registro del perfil de proyecto, elaboración de carpetas, avance físico, fichas técnicas entre otros como se muestran en las siguientes figuras.

Figura 3.28 Pantalla Submenú de proyectos

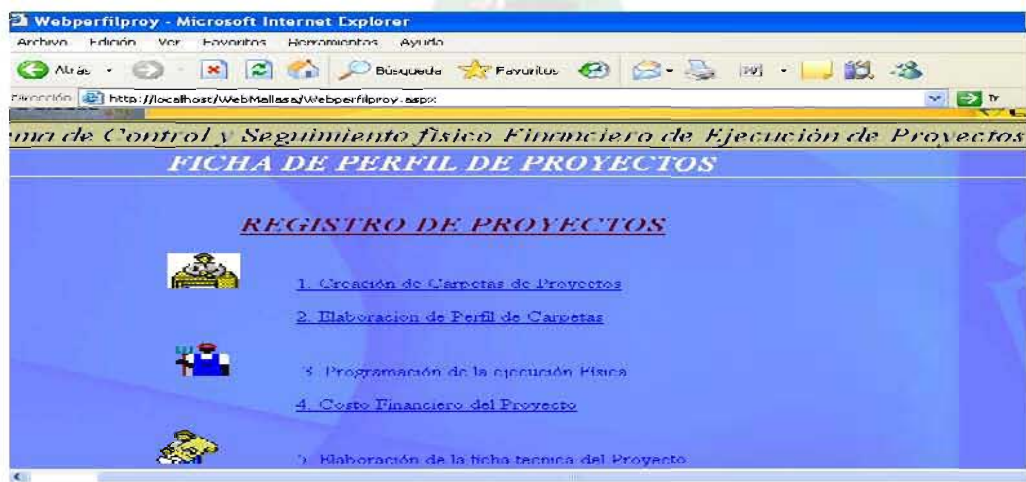


Figura 3.29 Pantalla de registro de perfil de proyectos

Webproyecto - Microsoft Internet Explorer

Responsable de información

Nombre: Magdalena Fecha: 13/12/2006 Cargo: Administradora

G. Identificación de impactos Ambientales

Se encuentra desarrollando en un ambiente más o menos perfecto

H. Descripción del proyecto

El proyecto se realiza de acuerdo a los beneficios

Código	Nombre proyecto	Zona	Responsable	Cargo Responsable	Fecha de s
P1	mejoramiento	Araucuez	Rosa flores	responsable	18/12/200
P2	g/e	Mallasilla	Rosalba riano	supervisor	18/12/200
P3	mantenimiento	Mallasilla	Mario	administrador	19/12/200
P4	retención	Chiriqué	maria teresa	supervisora	26/12/200
P5	mantenimiento	Jupapina	juan carlos	responsable	26/12/200
P6	retención	Mallas	tomados	planificación	26/12/200

Volver a la Principal

Fuente: Elaboración propia

Figura 3.30 Pantalla de registro de presupuesto de desembolso de proyecto

WebFinanciero - Microsoft Internet Explorer

REGISTRO DEL AVANCE FINANCIERO

Código: 0001 Fecha del proyecto: 25/12/2006

FOR ETAPA 1	COSTO(Bs)	FOR COMPONENTES	COSTO(Bs)
<input checked="" type="checkbox"/> Perfil	000,00	<input type="checkbox"/> Infraestructura	
<input type="checkbox"/> Prefeasibilidad		<input type="checkbox"/> Maquinas,Equipo y vehiculos	
<input checked="" type="checkbox"/> Feasibilidad	000,00	<input checked="" type="checkbox"/> Muebles y equipos de oficina	400,00
<input type="checkbox"/> Diseño final		<input type="checkbox"/> Bienes capitalizables	
<input checked="" type="checkbox"/> Ejecución	200,00	<input type="checkbox"/> Capacitación	
TOTALES	600,00	<input type="checkbox"/> Administración	
VERIFICACIÓN ADMINISTRATIVA		<input type="checkbox"/> Supervisión	
Unidad: U. Planificación	Cantidad: 5	<input type="checkbox"/> Personal	
Costo: U. Planificación		<input checked="" type="checkbox"/> Otros	500,00
U. Administración		TOTALES	900,00
Mejoramiento Uan			

Figura 3.31 Pantalla de registro de avance físico

REGISTRO DE AVANCE FISICO DE LOS PROYECTOS

Ingrese los datos de los avances

Código de Proyecto: DMA10
Nombre de Proyecto: construccion de colegio
Zona: Jupapina

Ingrese los datos de los avances

Estado Físico: Ejecucion
Descripción de estado Físico: hghghg
Porcentaje de Avance este mes: 40 %
Porcentaje Total de Avance: 100%
Fecha de actualización de avances: 18/07/2007

Julio de 2007

D	L	M	M	J	V	S
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

BUSCAR
ACTUALIZAR
ADICIONAR

DISEÑO DE REPORTES Y CONSULTAS

Para la realización de los reportes se utilizó Cristal Report, Como se muestra en las siguientes pantallas, el usuario podrá realizar consultas sobre los proyectos que se realizan en diferentes zonas y obtener reportes de los mismos, el porcentaje de avance físico, como financiero.

Figura 3.32 Pantalla de consultas de los proyectos

Fecha	Zona	Codigo	Responsable	Nombre de proyecto
21/05/2007 0:00:00	Mallasilla	2	carlos	mantenimiento
30/05/2007 0:00:00	Chiaraque	3	marcos	refaccioa
29/05/2007 0:00:00	Jupapina	4	maria	numm
29/05/2007 0:00:00	Mallasilla	5	jorge	numm
28/05/2007 0:00:00	Aranjuez	6	carmen	num

Figura 3.33 Reporte general de los proyectos

Codigo	Codigo carp	Nombre proyecto	Nom zona	Desde	Hasta	Responsable	Fecha
1	P1	refaccionamiento	Mallasilla	calle 1	calle 2	cristina	10/07/2007
2	P2	refaccionamiento	Mallasilla	calle 2	calle 10	Jose	17/07/2007
3	P3	Implementación de distribución vehicular	Aranjuez	mansano a	mansano a	Fernando	24/07/2007
4	P4	Electrificación rural chiarache	Chiaraque	mansano c	mansano c	Roberto	17/07/2007
5	P5	Electrificación rural	Florida	calle 7	calle 8	Rosmery	24/07/2007
6	P6	Mantenimiento	Chiaraque	calle 14	calle 15	jorge	10/07/2007

Fuente: Elaboración propia

Figura 3.34 Reporte de avance financiero de proyectos

Informe prior 1 / 1 100% powered by crystal

Fecha actual de emisión del informe 27/07/2007

Subalcaldía de Mallasa

REPORTE DE AVANCE FISICO DE LOS PROYECTOS

<u>Codigo_proy</u>	<u>Nombre_proy</u>	<u>Zona</u>	<u>Est_fisico</u>	<u>% Mes</u>	<u>% Total</u>	<u>Responsable</u>	<u>Fecha_avance</u>
P1	refaccionamiento	Mallasilla	Ejecucion	50,00	50,00	Jorge	25/07/2007 0:1
P2	refaccionamiento	Mallasilla	Concluido	100,00	100,00	Carlos	24/07/2007 0:1
P3	refaccionamiento	Mallasilla	Ejecucion	60,00	60,00	Carlos mmm	24/07/2007 0:1
P6	mantenimiento	Mallasilla	Ejecucion	40,00	80,00	Maria tope	24/07/2007 0:1

Figura 3.35 Reporte de avance Físico de proyectos

Subalcaldía Mallasa

REPORTE DE AVANCE FINANCIERO DE PROYECTOS

Codigo de proyecto : P1

Nombre del proyecto : refaccionamiento

Costo del proyecto : 40 000,00

<u>ActObj</u>	<u>Provigente</u>	<u>Preventivo</u>	<u>Comprometido</u>	<u>Devengado</u>	<u>Saldo preventivo</u>	<u>Saldo comprometido</u>	<u>Saldo devengar</u>	<u>Saldo paga</u>
112	30 000,00	2 000,00	900,00	100,00	28 000,00	1 100,00	800,00	100,00
114	100,00	30,00	21,00	12,00	70,00	9,00	9,00	12,00
115	400,00	12,00	10,00	9,00	388,00	2,00	1,00	9,00
Total Trans.	30.500,00	2.042,00	931,00	121,00	28.458,00	1.111,00	810,00	121,00

3.5 FASE DE TRANSICIÓN

El objetivo de esta fase es transferir el nuevo producto a los usuarios finales, pues son ellos quienes darán la verdadera funcionalidad y valor del mismo, de manera que les permita facilitar la manipulación del sistema con datos reales y darle un intervalo de tiempo para que el usuario no tenga ningún inconveniente en el manejo del sistema, o para detectar los errores y los requerimientos que en la fase de elaboración se han podido obviar.

CAPITULO IV

METRICA DE CALIDAD

Todo proyecto de la ingeniería de software debe partir con una buena planificación, pero lamentablemente, la planificación es una tarea nada trivial. Una de las dificultades de labor de los administradores y jefes de proyecto en torno a la planificación es la difícil tarea de realizar una estimación de costos y plazos realistas.

Por esta razón, que es necesario aplicar un método de medición de calidad para que un producto de software pueda ser óptimo.

El ISO/IEC 9126 ha sido desarrollado en un intento de identificar los atributos clave de calidad para el software, la característica del ISO/IEC 9126 son:

- Funcionalidad
- Portabilidad
- Usabilidad
- Mantenimiento

4.1 FUNCIONALIDAD

Una vez desarrollado el modelo de análisis, que es la fase de obtención de los requisitos que nos ayuda establecer el funcionamiento para el diseño, se aplicara la métrica punto función.

Punto Función

Métrica que examina el modelo de análisis, para predecir el tamaño del sistema resultante ya que el tamaño y la complejidad del diseño están directamente relacionados.

Primeramente aplicaremos las siguientes características:

Numero de entrada del Usuario: Representan cada entrada de control del usuario que ingresa datos a un archivo del sistema, además el software proporciona diferentes datos orientados a la aplicación. Las entradas deben ser distinguidas de las peticiones, que se contabilizan por separado, cada archivo lógico requerirá de tres tipos de entradas, agregar, cambiar y borrar.

1. Pantalla de Registro de los Requerimientos
2. Pantalla de Registro perfil de proyectos
3. Pantalla de Registro de Adjudicación
4. Pantalla de Registro de ejecución de avance físico
5. Pantalla de Registro de ejecución de avance financiero
6. Pantalla de Registro de costo de presupuestos de proyectos
7. Pantalla de Registro de Carpetas de proyectos
8. Pantalla de Registro de ficha técnica de proyectos
9. Pantalla de autenticación
10. Pantalla de Registro de los usuarios
11. Pantalla de Registro Organización ejecutora
12. Pantalla de menú principal
13. Pantalla de creación de carpetas
14. Pantalla de los reportes de los proyectos
15. Pantalla de submenú de consultas

Numero de Salidas de Usuario: Representa cada salida de control de usuario que realiza la exportación de datos de un archivo del sistema. La salida se refiere a reportes, datos a pantalla y mensajes de error.

1. Informe de Administrativo
2. Reporte del avance físico de proyectos

3. Reporte del avance financiero de proyectos
4. Informe de los costos de proyectos
5. Informe del cronograma de ejecución de los proyectos
6. Reportes generales de la ejecución de los proyectos
7. Informe de programación física de proyectos
8. Informe de la unidad ejecutora
9. Reporte de proyectos por carpetas

Numero de Peticiones de Usuario (Consultas): Representa cada combinación entrada/salida única, donde una entrada causa y genera una salida inmediata, como una consulta externa, incluir las consultas externas que entran directamente desde el usuario y aquellas que entran desde otra aplicación.

1. Consulta General de los Proyectos
2. Consulta de los proyectos en ejecución, y concluidos
3. Consulta de los proyectos pertenecientes en una determinada zona
4. Consulta del avance físico y financiero

Número de Archivos: Representa cada archivo maestro, también se define como datos relacionados lógicamente o información de control que se encuentra dentro de los límites de la aplicación, es decir datos que puede ser parte de una gran base de datos o también un archivo independiente.

Total de numero de archivos = 10

Numero de Interfaces externas: Representa cada agrupación importante de datos traspasados o compartidos entre aplicaciones. Todas las interfaces legibles por la maquina que son utilizados para transmitir información a otro sistema.

1. Disco
2. Backup

El resumen de las operaciones anteriormente descritas se las presenta en la siguiente Tabla 4.1.

Tabla 4.1 Computación de métricas de Punto Función

Parámetros de Medición	Cuenta	Factor de Ponderación	Total
Numero de Entradas de Usuario	15	4	60
Numero de Salidas de Usuario	9	5	45
Numero de peticiones de Usuario	4	5	16
Numero de Archivos	10	10	100
Numero de Interfaces Externos	2	7	14
Cuenta Total			235

Fuente: [Pressman, 1999]

Tabla 4.2 Ajuste de Complejidad

Valores de Ajuste de Complejidad	Valor
1. ¿Requiere el sistema copias de Seguridad y de recuperación fiable?	5
2. ¿Se requieren comunicaciones de datos?	4
3. ¿Existen funciones de procesamiento distribuido?	0
4. ¿Es Critico el rendimiento?	2
5. ¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?	5
6. ¿Requiere el sistema entrada de datos interactivo?	4
7. ¿Requiere la entrada de datos interactivo que las transacciones de entrada se lleven a cabo sobre múltiples pantallas y variadas operaciones?	2
8. ¿Se actualizan los archivos maestros de forma interactiva?	4
9. ¿Son Complejas las entradas, las salidas, los archivos y peticiones?	2
10. ¿Es complejo el procesamiento interno?	3
11. ¿Se ha diseñado el código para ser reutilizable?	5
12. ¿Están incluidas en el diseño la conversión y la instalación?	4
13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?	3
14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	5
TOTAL	48

Con los datos obtenidos en la anterior tabla se puede calcular el Factor de Ajuste de Complejidad del sistema cuya ecuación es la siguiente:

$$PF = \text{Cuenta_Total} * [0,65 + 0,01 * \text{SUM} (Fi)]$$

Donde:

Cuenta_total = Es la suma de todas las entradas obtenidas en la tabla anterior.

0.01 = Error de Confiabilidad del sistema

Sum (Fi) = Es la sumatoria de los Factores de Complejidad

Por tanto:

$$PF = 235 * (0.65 + 0,01 * 48)$$

$$PF = 235 * 1.15$$

$$PF = 265.55$$

Ahora calculamos el ajuste:

$$\text{Ajuste} = \text{Cuenta_total} * \text{Valor máximo de complejidad}$$

$$\text{Ajuste} = 235 * 1.35$$

$$\text{Ajuste} = 317.25$$

Por tanto calculamos la funcionalidad

$$\begin{aligned} \text{Funcionalidad} &= PF / \text{Ajuste} \\ &= 265.55 / 317.25 \\ &= 0.85 \end{aligned}$$

Entonces se puede decir que se tiene 85 % de funcionalidad.

4.2 PORTABILIDAD

La portabilidad es la facilidad con que el software puede ser llevado de un entorno a otro, cumpliendo lo siguiente: facilidad de instalación, facilidad de ajuste y facilidad de adaptación al cambio.

Por lo tanto debe tener las siguientes características:

- El sistema puede ser portable a cualquier lenguaje de programación, porque ha sido diseñado en VB.NET.
- Plataforma Windows 2000, Windows XP para adelante,

- Requiere como mínimo un equipo Pentium III de 250 Mhz de velocidad y una memoria de 128 Mb.

4.3 USABILIDAD

Es lo mismo que la facilidad de uso, esta métrica nos muestra el coste de aprender a manejar el producto, se lo calcula de la siguiente manera.

$$FU = [(Sum xi / n) * 100] / n$$

La evaluación está en una escala de 1 a 5

Evaluación	Escala
Pésimo	1
Malo	2
Regular	3
Bueno	4
Muy Bueno	5

Se debe formular las siguientes preguntas:

Tabla 4.3 Formulación de preguntas para calcular la usabilidad

Nro	Preguntas	Evaluación
1	¿El sistema satisface los requerimientos de manejo de información?	4
2	¿Las salidas del sistema estan de acuerdo a sus requerimientos?	4
3	¿Cómo considera el ingreso de datos al sistema?	4
4	¿Cómo considera los formularios que elabora el sistema?	4
5	¿El sistema facilita el trabajo que realiza?	5
Total		21

Luego de obtener los datos, procedemos a calcular, tomando en número de veces que se realizaron las pruebas.

$$FU = [(21/5) * 100] / 5$$

$$FU = 84\%$$

Por tanto la facilidad de uso es 84 %.

4.4 MANTENIBILIDAD

La facilidad de mantenimiento, es óptimo es decir que se puede realizar correcciones en el programa sin alterar los demás programas.

El estándar IEE94 sugiere un índice de madurez de software (IMS) que proporciona un indicador de la estabilidad de un producto, tomando en cuenta de los cambios que ocurre en cada versión del producto, calculando de la siguiente manera:

$$IMS = [Mt - (Fa + Fc + Fd)] / Mt$$

Donde:

Mt = Numero de módulos de la versión actual

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

Fa = Numero de módulos de la versión actual que se han añadido.

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

Por lo tanto se tiene:

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

$$IMS = 0.8$$

A medida que IMS se acerca a uno el producto va estabilizarse.

4.5 ANALISIS DE RESULTADOS

Una vez aplicadas las métricas proseguimos a realizar un análisis de resultados para determinar la calidad del sistema.

Aplicaremos el análisis de la métrica de punto función, donde se toma en cuenta el parámetro obtenido por la métrica punto función que es de un valor de 265.55 que equivale a un 85%, a partir de este valor realizaremos el análisis haciendo variar el nivel de confiabilidad en un rango de 0 a 1, a continuación se muestra en la Tabla 4.4 el cálculos de los datos.

$$PF = 235 * [(Confiabilidad + (0,01 * 48))]$$

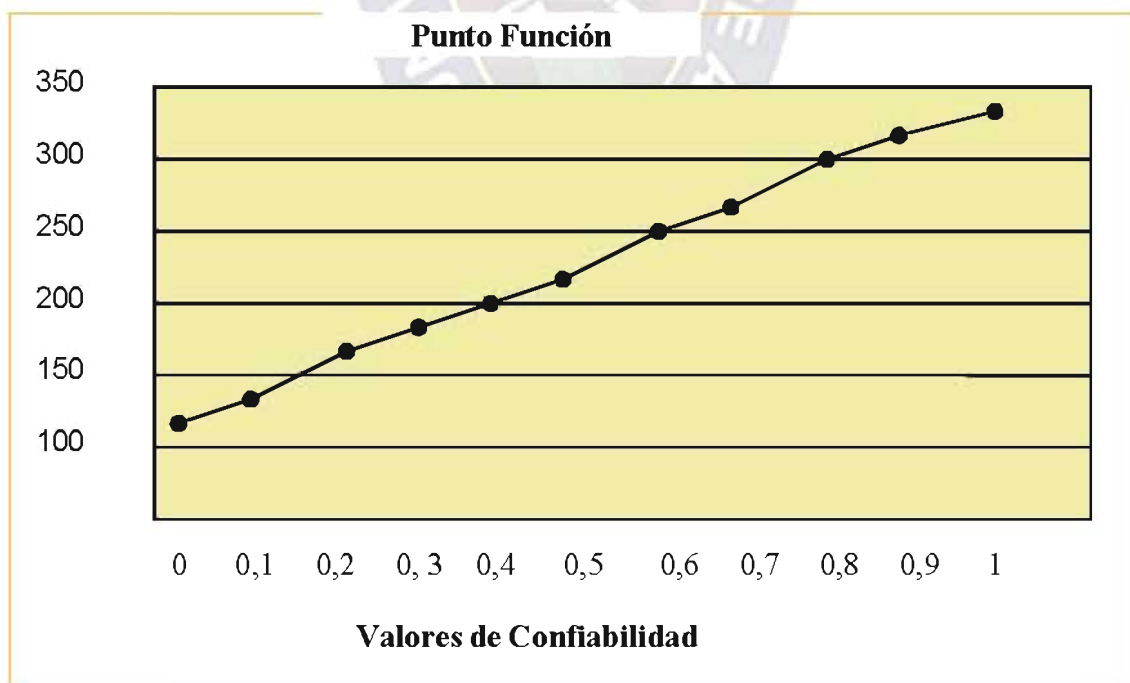
Tabla 4.4 Análisis de valores de Punto Función

Confiabilidad	Punto Función
0	112,8
0,1	136,3
0,2	159,8
0,3	183,3
0,4	206,8
0,5	230,3
0,6	253,8
0,7	277,3
0,8	300,8
0,9	324,3
1	347,8

Fuente: Elaboración propia

Por lo tanto se tiene se tiene el siguiente diagrama de análisis de valor de Punto Función que se obtuvo en la Tabla 4.4.

Figura 4.1 Diagrama de análisis de valor de Punto Función



Fuente: Elaboración Propia

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

El presente proyecto se llevo a las siguientes conclusiones

- El presente sistema de control y seguimiento físico de ejecución de proyectos, ha sido concluido de manera satisfactoriamente.
- El objetivo principal del proyecto se logro aplicando la metodología RUP que permitió optimizar el tiempo de desarrollo y UML que permitió facilitar la realización del análisis y diseño del sistema.
- Se ha logrado implementar el modelo de control de carpetas de los proyectos y los respectivos avances mediante redes de petri, que nos permite modelar los eventos o acciones que se presenten en el seguimiento de proyectos.
- Se ha logrado implementar el registro y la clasificación de los recursos de cada proyecto, disminuyendo el tiempo de registro en un 85%.
- Se ha logrado implementar el modelo de control de avance físico y financiero de proyectos mediante redes de petri, que disminuye el trabajo en un 79%.

- Se ha logrado optimizar la elaboración de las carpetas en un 85%, de manera que las carpetas se muestran en un tiempo óptimo.
- El sistema se encarga de generar reportes e informes de forma automática permitiendo optimizar el tiempo de respuesta de la solicitud de los usuarios, incluso disminuyendo las horas de trabajo.
- La utilización del sistema está considerado entre los siguientes actores; Unidad de Planificación y Control, Unidad de Administración, Finanzas y técnicos.



5.2 RECOMENDACIONES

Después de realizar el proyecto se recomienda lo siguiente:

- Se puede ampliar el estudio de este proyecto, en la parte de control físico de proyectos aplicando la técnica PERT/CPM/ROY la cual brinda grandes beneficios para el seguimiento de proyectos.
- Se sugiere incorporar una página Web, que se integre con el sistema para que las personas fuera de la institución puedan tener la información respecto al avance de los proyectos u obras y gocen de los beneficios que brinda el sistema.
- Para futuros trabajos se recomienda que se estudie la técnica de cronograma de Gantt que integra todas las actividades y duraciones de un proyecto, tomando en cuenta las holguras totales y libres correspondientes a cada una de ellas.
- Se recomienda que el administrador realice mantenimiento de los datos para que la información tenga valor y uso, de manera que no exista algún tipo de errores para emitir informes, además debe bajar el Backup de la base de datos y verificar el acceso de los usuarios para tener una mayor seguridad física.

BIBLIOGRAFÍA

[Date C.I., 1994] Introducción a los sistemas se Base de Datos, Volumen I, 5ta Edición Addison Wesley Iberoamerica.

[Kendall, kendall., 1997] Análisis y Diseño de sistemas, Prentice Hall Hispanoamericana.

[Pressman Roger, 1993] Ingeniería de Software un enfoque Practico, Volumen1, 4ta, Edicion, España

[Jacobson, Booch, R., 2000] El Proceso Unificado de Desarrollo de Software, Addison Wesley Iberoamerica.

[Balena Francesco , 2002] Programación de Visual Basic, Addison Wesley Iberoamerica.

[Revollo Paton Edgar 1997] Sistema de Seguimiento Y supervisión de obras de construccion civil, Carrera de Informática.

[Rojas Rojas Abdón, 2004] Sistema de Calculo de costos para los Proyectos de "AADDPAL".

[Velásquez Miranda Lourdes, 2004] Sistema de información para la Administración de proyectos y control de almacén de mantenimiento de Servicios Eléctricos.

REFERENCIAS DE PAGINAS

[Métrica de calidad, 10] Métrica de medición de calidad

<http://www.SQA.com/tallerdegestiondesoftware.ppt>

"Modelado de Software y tiempo real", 2002, Edición en línea.

http://www.therationalledge.com/content/jan_01/t_rup_ge.html

"Murata, T(1989, April)Petri Nets, Basic Models Springer-Verlag the world of petri nets "

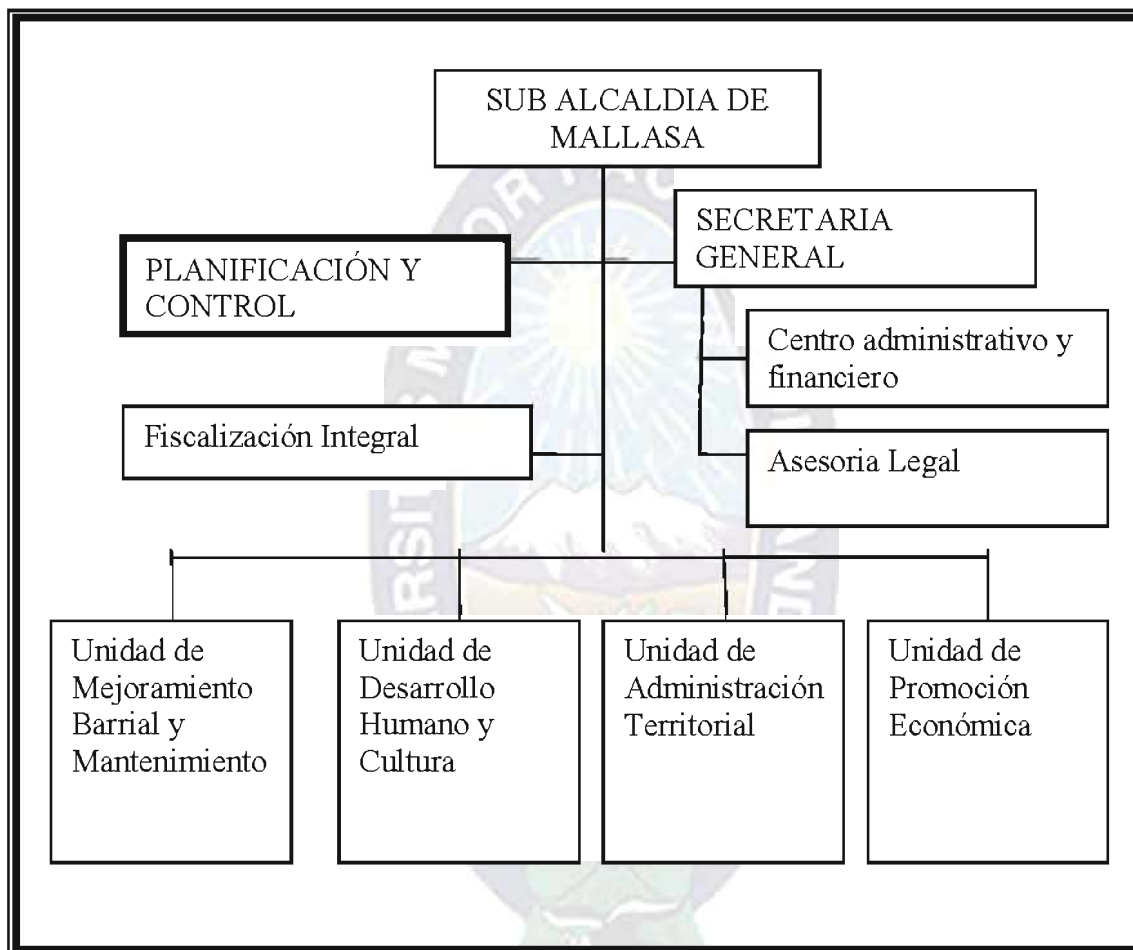
<http://www.damimi.au.dk/PetriNets/>

<http://www.lulo.com.ve>

<http://www.es.Geocities.com/webdetesis>

ANEXO A

Organigrama de la Sub Alcaldía de Mallasa



Fuente Sub Alcaldía de Mallasa

ANEXO B

IDENTIFICACIÓN DE ACTORES Y CASOS DE USO

Tabla B.1 Descripción de negocio: Secretaria

ACTOR	RESPONSABLE DE RECEPCIÓN (Secretaria)
Descripción	Es la encargada de llevar la actividad de recepción, su función consiste de capturar toda la información de solicitudes
Características	<ul style="list-style-type: none"> - Este actor se encarga de clasificar las solicitudes. - Es encargado de llevar los formularios a su destino. - Actualizar cada carpeta.
Relaciones	Casos de uso: Recepción y Clasificación de requerimientos

Tabla B.2 Descripción de negocio: Administración y finanzas

ACTOR	UNIDAD DE ADMINISTRACIÓN Y FINANZAS
Descripción	Es el encargado de llevar la actividad de aprobación, su función consiste de revisar detalladamente los presupuestos asignados y mandar a las respectivas instancias para su aprobación.
Características	<ul style="list-style-type: none"> - Este actor se encargo de revisión de presupuesto de obras. - Es encargado de la aprobación del presupuesto o el rechazo. - Emitir los respectivos informes de aprobación a su destino.
Relaciones	Casos de uso: Aprobación de los proyectos Actores. Unidad Administración y finanzas

Tabla B.3 Caso de uso negocio. Clasificación de requerimientos

CASO DE USO	RECEPCIÓN Y CLASIFICACIÓN DE REQUERIMIENTOS
ACTORES	Responsable de recepción, usuarios
Descripción	<ul style="list-style-type: none"> - Registra los requerimientos nuevos. - Es el encargado de entrada y salida de los formularios. - Actualiza los historiales como las carpetas. - Realiza solicitudes de materiales, de los responsables de los proyectos a la unidad de almacenes.

Tabla B.4 Descripción de negocio: Responsable de proyecto

ACTOR	RESPONSABLE DE PROYECTO, SUPERVISOR
Descripción	Es el encargado de llevar la ejecución del proyecto.
Características	<ul style="list-style-type: none"> - Este actor se encargo de ejecutar el proyecto asignado. - Es encargado de controlar el avance físico de los proyectos. - Emitir los respectivos informes del avance físico y financiero - Responsable de la entrega de los proyectos ya finalizados.
Relaciones	<p>Casos de uso: Ejecución de proyecto, avance físico y financiero de los proyectos.</p> <p>Actores. Responsable, Unidad de planificación y control.</p>

Tabla B.5 Caso de uso negocio. Ejecución de proyectos

CASO DE USO	EJECUCIÓN DE PROYECTO
ACTORES	Responsable , supervisor, Unidad de planificación y control
Descripción	<ul style="list-style-type: none"> - Realiza cronograma de actividades - El responsable brinda informes de ejecución del proyecto. - El responsable registra las los datos correspondientes y se elaboran los respectivos informes. - El responsable realiza una lista de pedidos.

Tabla B.6 Caso de uso negocio. Registro de avance físico y financiero de proyectos

CASO DE USO	ENTREGA DE PROYECTOS
ACTORES	Sub alcalde
Descripción	<ul style="list-style-type: none"> - Este actor se encarga de la entrega de los proyectos. - Realiza cronograma de entrega de proyectos finalizados. - Realizar la certificación de los informes de proyectos terminados.

Tabla B.7 Caso de uso de negocio. Entrega de proyectos

CASO DE USO	AVANCE FISICO y FINANCIERO DE LOS PROYECTOS
ACTORES	Responsable, supervisor, Unidad de planificación y control
Descripción	<ul style="list-style-type: none"> - Responsable registra los avances físicos como financiero. - Emite reportes del avance del proyecto. - El supervisor solicita reportes del avance. - Actualiza las carpetas de cada proyecto. - Controla las entradas y salidas de los materiales.

ANEXO C

IDENTIFICACIÓN DE CASOS DE USO CON EL SISTEMA

Tabla C.1 Validación de acceso al sistema

Caso de uso	Validación de acceso al sistema	
Actor	Administrador, Usuario	
Propósito	Realizar el registro y control de usuarios	
Resumen	El administrador registra los datos del usuario, asignando niveles de privilegios. Los usuarios solo podrán acceder si están autorizados.	
Tipo	Primario	
Flujo principal	Acción del actor	Respuesta del sistema
	<p>1. El Administrador activa el formulario.</p> <p>3. Registra los datos personales de los usuarios como el login y el password, luego presiona grabar.</p> <p>5. El usuario activa el formulario de identificación del formulario.</p> <p>7. El usuario ingresa su nombre y password.</p> <p>9. El usuario puede ingresar a las páginas que solamente le están permitidas.</p>	<p>2. El sistema despliega el formulario de nuevo usuario.</p> <p>4. El sistema verifica los valores si están correctos y la no existencia de duplicidad almacena.</p> <p>6. El sistema despliega el formulario de identificación.</p> <p>8. El sistema verifica los datos y permite el acceso al sistema.</p>
Precondición	Registra solamente el administrador.	
Poscondición	Registro de los usuarios autorizados.	

Fuente: Elaboración propia

Tabla C.2 Registro del avance físico y financiero de proyectos

Caso de uso	Registro de avance físico financiero de proyectos	
Actor	Responsable , Supervisor , U. de planificación y control	
Propósito	Realizar el registro avance del proyecto físico como financiero.	
Resumen	Genera informes de los avance de los proyectos	
Tipo	Primario	
Flujo principal	Acción del actor	Respuesta del sistema
	<p>1. El responsable activa el formulario avance físico.</p> <p>3. Registra el avance físico del proyecto con los datos correspondientes.</p> <p>5. El responsable activa el formulario del avance financiero.</p> <p>7. Procede al cálculo del presupuesto empleado.</p> <p>9. Verifica si está correcto y hace clic en aceptar.</p> <p>11. Procede al registro del avance financiero del proyecto.</p> <p>13. Para salir de la página, clic en el botón salir.</p>	<p>2. El sistema despliega el formulario del avance físico.</p> <p>4. El sistema verifica los valores si están correctos y almacena en la base de datos.</p> <p>6. El sistema despliega el formulario del avance financiero.</p> <p>8. El sistema calcula el presupuesto empleado y visualiza.</p> <p>10. El sistema almacena la información en la base de datos.</p> <p>12. Almacena en la base de datos los datos introducidos.</p> <p>14. El sistema cierra el menú y vuelve a la página principal, en caso de salir.</p>
Precondición	El responsable del registro debe pertenecer a la unidad.	
Poscondición	Registro de avance físico y financiero	

ANEXO D

DIAGRAMAS DE ANALISIS Y DISEÑO DE CASOS DE USO

1. DIAGRAMA DE CLASES DE UNA REALIZACIÓN DE CASOS DE USO

Figura D. 1.1 Diagrama de clase de Licitación de los proyectos

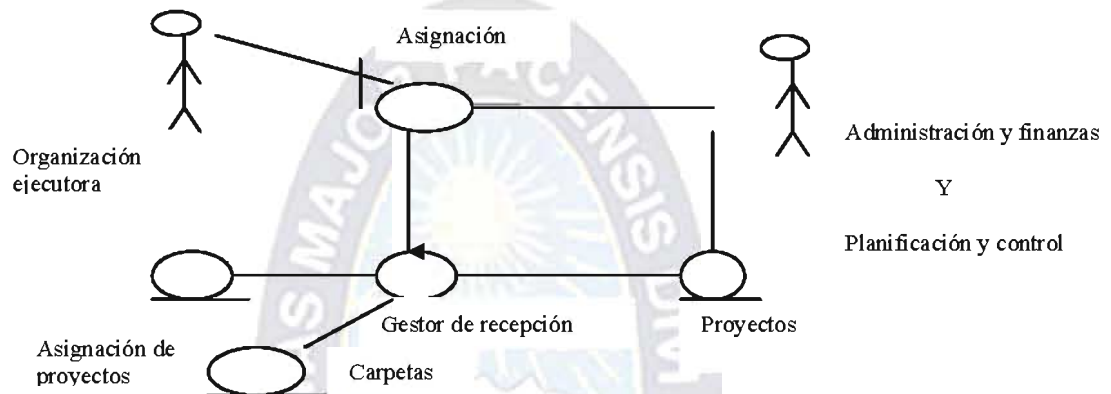
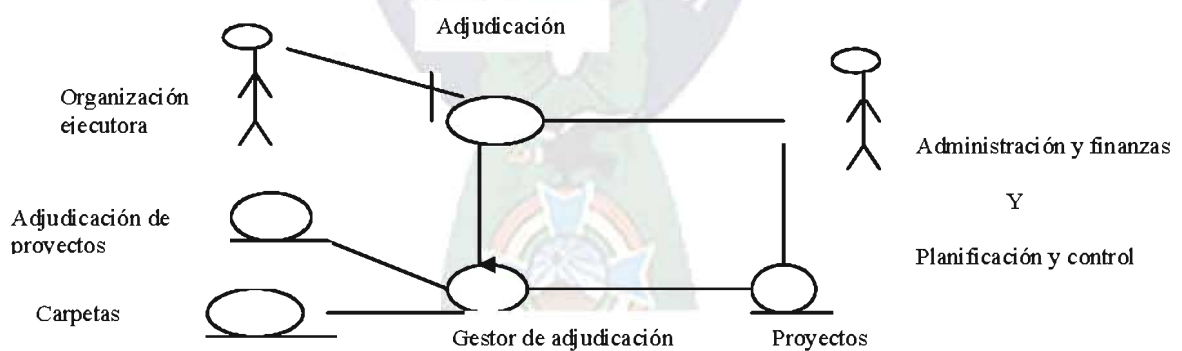


Figura D.1.2 Diagrama de clase de Adjudicación de los proyectos



2. DIAGRAMA DE COLABORACION DE UNA REALIZACIÓN DEL CASO DE USO

Figura D.2.1 Diagrama de colaboración de Licitación de proyectos

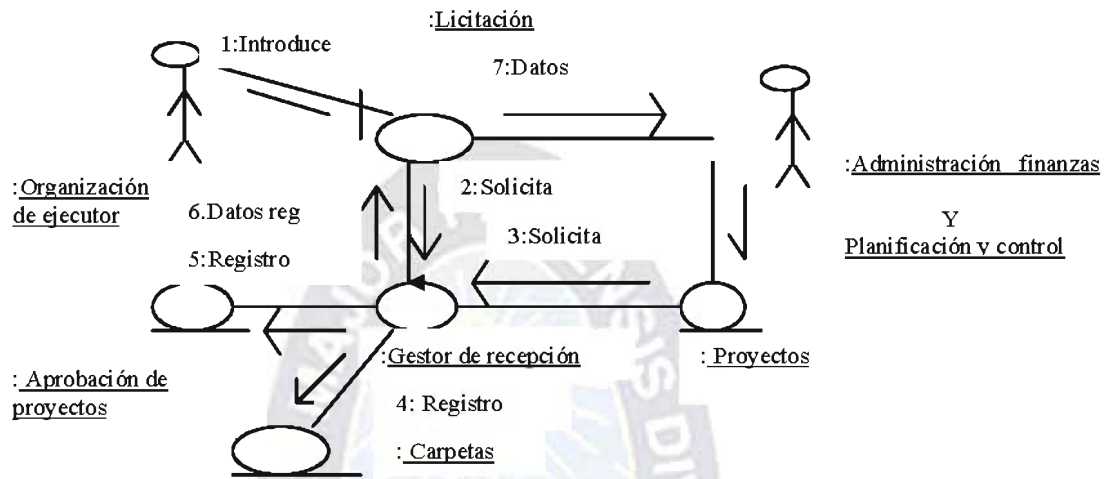
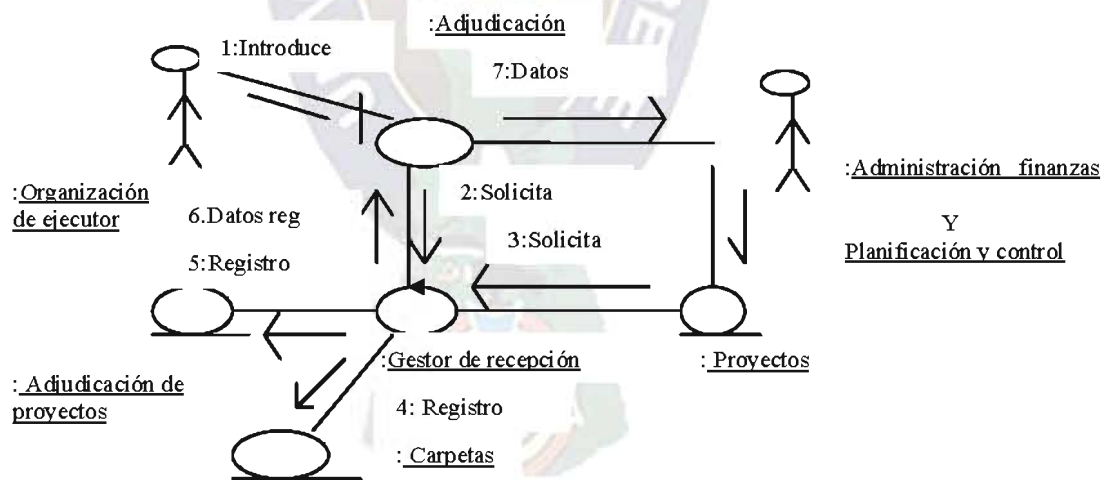


Figura D.2.2 Diagrama de colaboración de Adjudicación de los proyectos



3. DIAGRAMA DE CLASES DE DISEÑO DE CASOS DE USO

Figura D.3.1 Diagrama de clases de Eiecución de proyectos

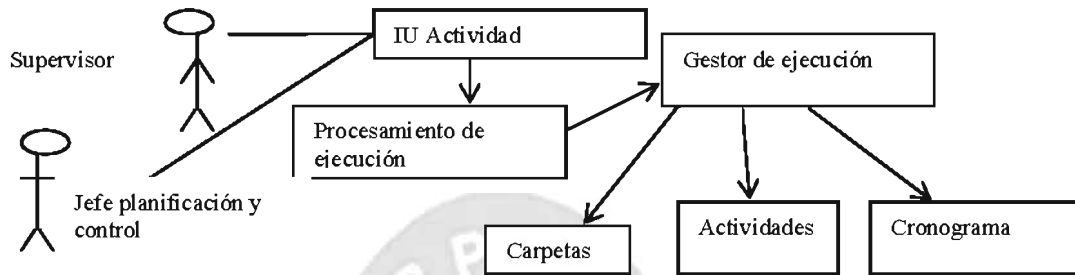


Figura D.3.2 Diagrama de clases de Validación del acceso al sistema



4. DIAGRAMA DE SECUENCIA DE DISEÑO DE CASO DE USO

Figura D.4.1 Diagrama de Secuencia de Ejecución de proyectos

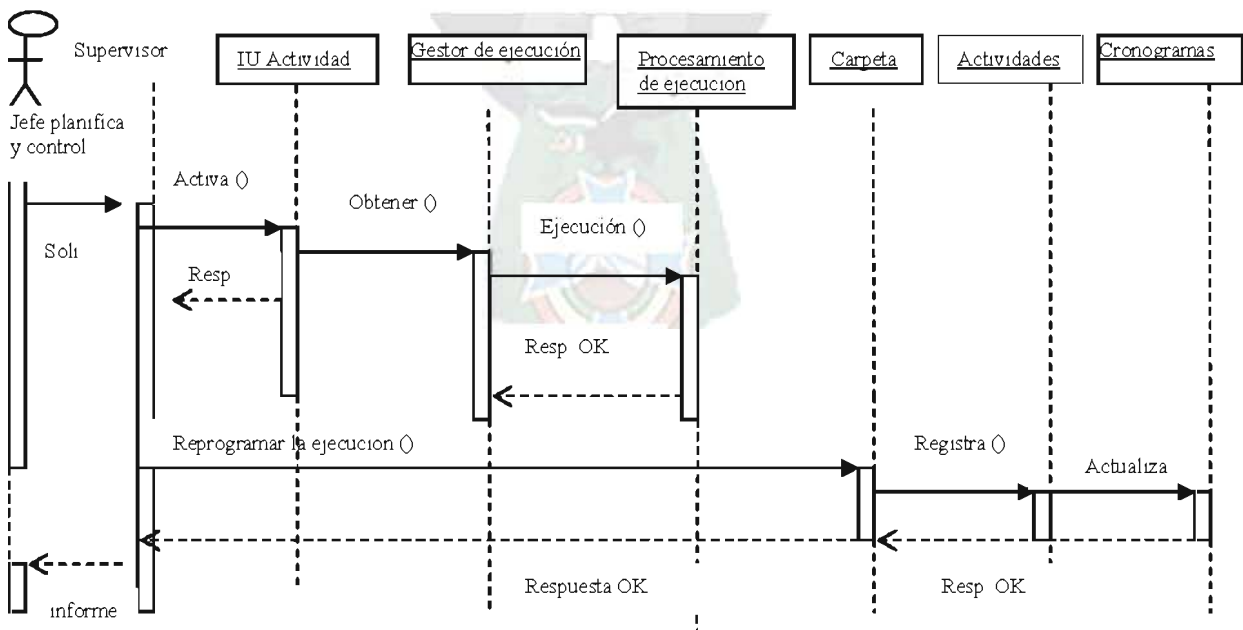


Figura D.4.2 Diagrama de Secuencia de diseño de caso de uso
Licitación de proyecto

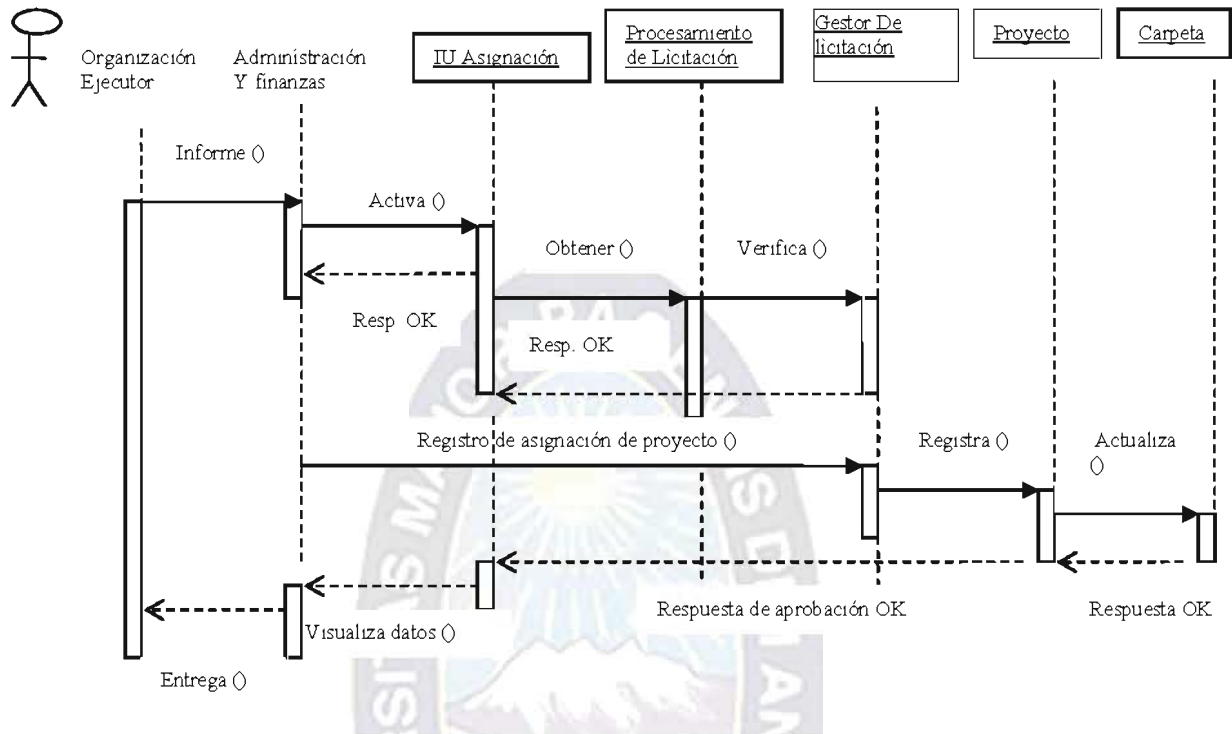


Figura D.4.3 Diagrama de Secuencia de diseño de caso de uso
Avance físico y financiero de proyectos

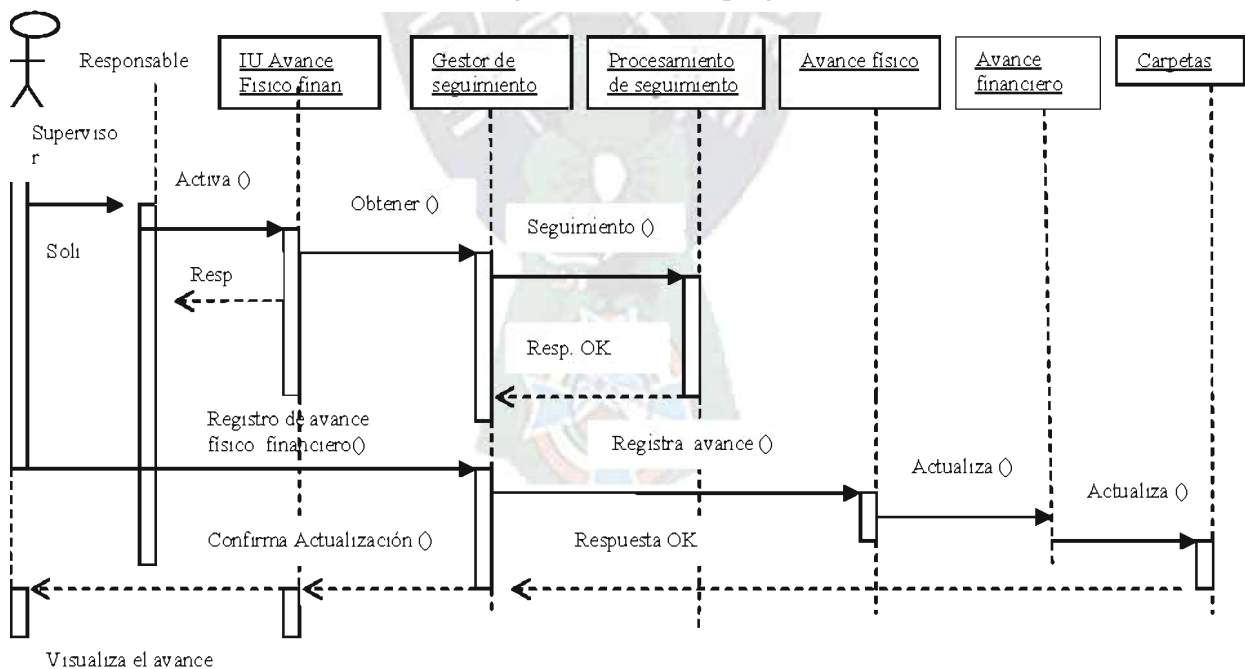
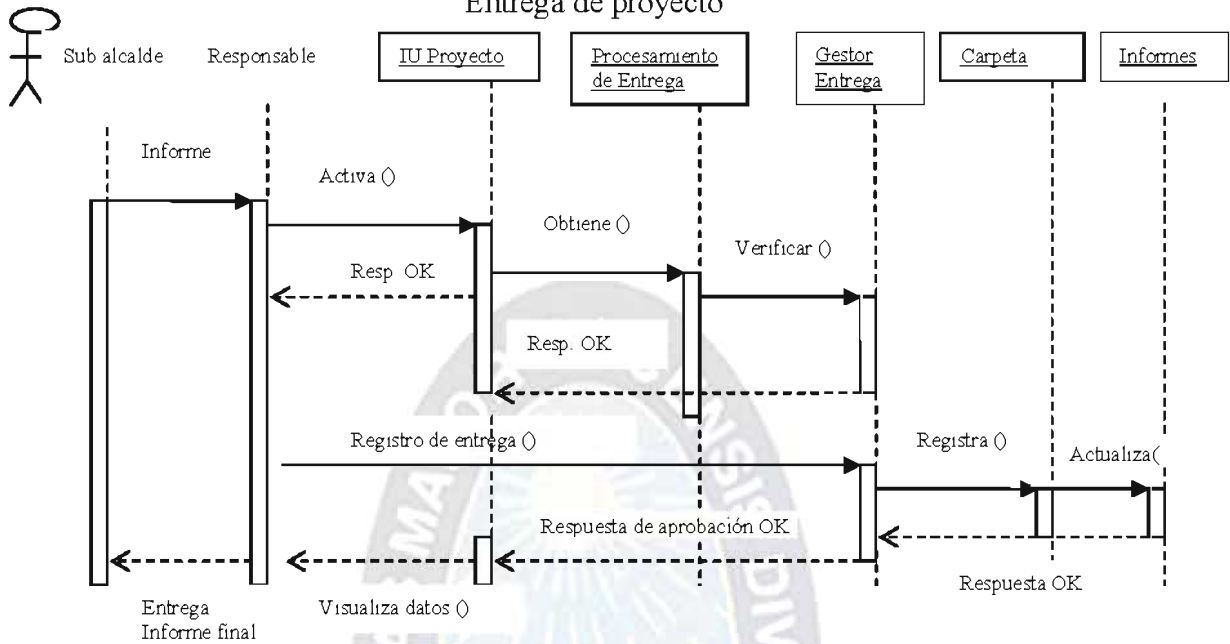


Figura D.4.4 Diagrama de Secuencia de diseño de caso de uso
Entrega de proyecto



5. DIAGRAMA DE ESTADOS PARA LAS CLASES

Figura D.5.1 Diagrama de Estados para la clase Ejecución de proyecto

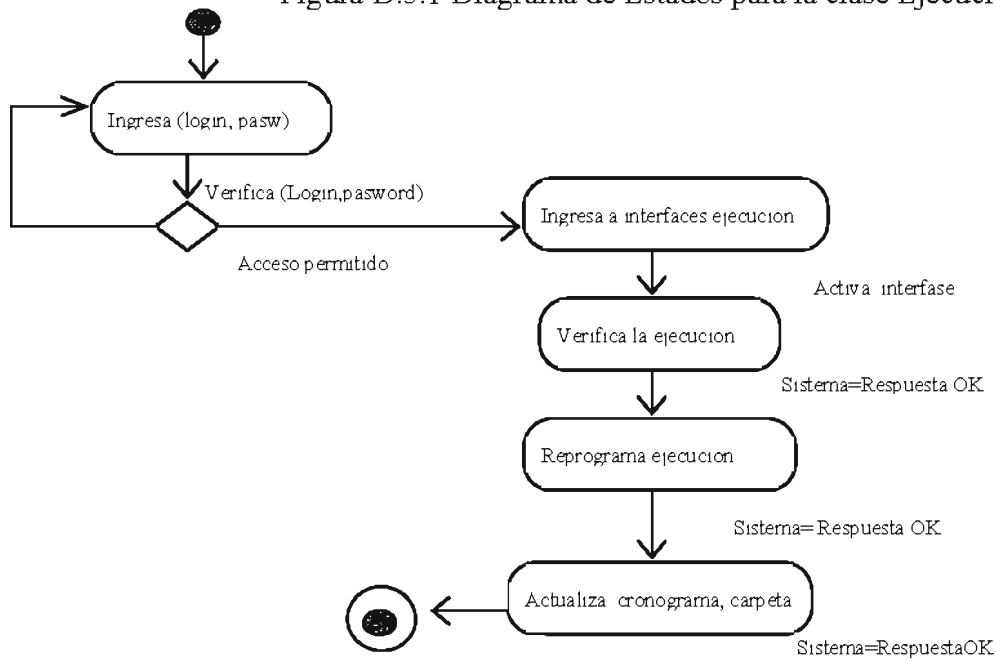


Figura D.5.2 Diagrama de Estados para la clase Avance físico y financiero de proyecto

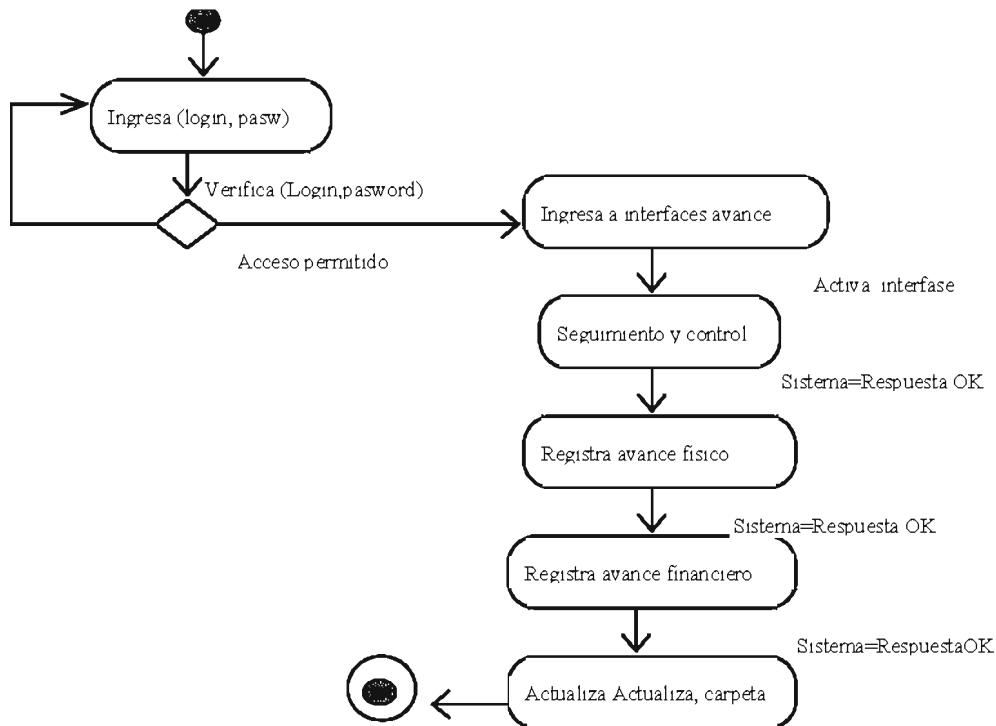


Figura D.5.3 Diagrama de Estados para la clase Adjudicación de proyectos

