



**UNIVERSIDAD DEL AZUAY
FACULTAD DE CIENCIAS DE LA
ADMINISTRACION
ESCUELA DE INGENIERIA DE SISTEMAS**

TEMA

**Sistema WorkFlow para procesos manejados
dentro del Gobierno Provincial del Azuay**

**Tesis previa a la obtención del
Título en Ingeniería de Sistemas**

Autor

Michael Estuardo Astudillo Sánchez

Director

Ing. Pablo Esquivel

**CUENCA-ECUADOR
2007**

AUTORIA

Las Investigaciones, análisis, software y diversas ideas expresadas en el presente documento son de exclusiva responsabilidad de sus autores.

Michael Estuardo Astudillo Sánchez

Dedicatoria

Comienzo citando una frase “Cuando una luz se apaga inmediatamente otra se enciende” con una meta alcanzada, surgen muchas más, un objetivo que dedico a mi esposa por haber creído en mi, en mi alcance como persona y en mi interés como potenciador de una idea, el trabajo puesto en esta tesis lo he alcanzado gracias a su fe en mi, llenando de amor y comprensión esas hora intensas de trabajo, que ella supo comprender y que ella supo alivianar.

Agradezco también a mi padre y a mi madre ya que sin ellos no tuviera esa perseverancia y amor por mi trabajo, ya que desde el momento en que me integre a la universidad tuvieron la certeza de que alcanzaría mi meta.

Agradecimientos

En estos cinco cortos años puedo mencionar a muchas personas que formaron parte de esta etapa, sin embargo tengo un especial agradecimiento por aquellos profesores que nos pusieron tareas fuertes, esas que nos hicieron trabajar de sol a sol, que lograron unir a estudiantes en una sola fuerza. Finalmente agradezco a las personas e instituciones que de una u otra forma formaron parte del presente trabajo.

Índice de Contenidos

Dedicatoria.....	III
Agradecimientos.....	IV
Índice de Contenidos.....	V
Índice de Ilustraciones y Cuadros.....	IX
Índice de Anexos.....	X
Resumen.....	XI
Abstract.....	XII
Introducción General.....	1
CAPITULO I	2
INVESTIGACION DE CAMPO	2
1 INVESTIGACIÓN DE CAMPO	3
1.1 INTRODUCCIÓN.	3
1.2 LA INSTITUCIÓN	3
1.3 EL GOBIERNO PROVINCIAL DEL AZUAY	3
1.3.1 Especificación de Requisitos.....	3
1.3.2 Definición de Procesos.	4
1.3.3 Problemática de la Institución	5
1.4 CONCLUSIONES	5
CAPITULO II.....	6
INVESTIGACION DE TECNICA.....	6
2 INVESTIGACIÓN TÉCNICA	7
2.1 INTRODUCCIÓN	7
2.2 WORKFLOW.....	7
2.2.1 <i>Concepto</i>	8
2.2.1.1 Elementos	8
2.2.1.1.1 Procesamiento de Imágenes.....	8
2.2.1.1.2 Administración de Documentos.....	8
2.2.1.1.3 Correo Electrónico	8
2.2.1.1.4 Aplicaciones Groupware	9
2.2.1.1.5 Aplicaciones basadas en transacciones	9
2.2.1.1.6 Software de Soporte al Proyecto.....	9
2.2.1.1.7 BPR y herramientas de diseño de sistemas estructurados.....	9
2.2.1.1.8 Separación de la funcionalidad Workflow	10
2.2.2 <i>Elementos</i>	11
2.2.2.1 Infraestructura Necesaria.....	11
2.2.2.2 Tecnologías Necesarias	11
2.2.2.3 Diferencias S.I.	11
2.2.3 <i>Análisis</i>	12
2.2.4 <i>Diseño</i>	13
2.2.4.1 Diseño de Flujos de Trabajo.....	13
2.2.4.1.1 Elaboración de Fichas	13
2.3 PATRONES WORKFLOW	14

2.3.1	<i>Controles Básicos</i>	14
2.3.1.1	Secuencia.....	14
2.3.1.2	División en Paralelo.....	14
2.3.1.3	Sincronización.....	15
2.3.1.4	Elección Exclusiva.....	15
2.3.1.5	Simple Merge.....	16
2.3.2	<i>Bifurcación avanzada y Sincronización</i>	16
2.3.2.1	Elección Múltiple.....	16
2.3.2.2	Sincronizar Merge.....	16
2.3.2.3	Multi-Merge.....	17
2.3.2.4	Discriminador.....	17
2.3.3	<i>Patrones Estructurales</i>	18
2.3.3.1	Ciclos arbitrarios.....	18
2.3.3.2	Terminación Implícita.....	18
2.3.4	<i>Patrones Involucrando Múltiples Instancias</i>	18
2.3.4.1	Múltiples Instancias sin Sincronización.....	18
2.3.5	<i>Patrones basados en el estado</i>	19
2.3.5.1	Elección en diferido.....	19
2.3.5.2	Ruteo Interpaginado Paralelo.....	19
2.3.5.3	Hitos.....	20
2.3.6	<i>Patrones de Cancelación</i>	20
2.3.6.1	Cancelar un Caso.....	20
2.3.6.2	Cancelar Actividad.....	21
2.4	REGLAS DEL NEGOCIO.....	21
2.4.1	<i>Marco Estructural</i>	23
2.4.2	<i>Políticas</i>	23
2.4.3	<i>Estrategias</i>	24
2.4.4	<i>Operativa Institucional</i>	24
2.5	INTRANET.....	24
2.5.1	<i>Ambiente WEB</i>	24
2.5.2	<i>Servidor HTTP Apache</i>	25
2.5.3	<i>SQL Server</i>	25
2.6	CONCLUSIONES.....	25
CAPITULO III		26
ANALISIS		26
3	ANÁLISIS	27
3.1	INTRODUCCIÓN.....	27
3.2	DESCRIPCIÓN DE LA SITUACIÓN ACTUAL.....	27
3.3	UNIDADES DE NEGOCIO.....	27
3.3.1	<i>Invitaciones</i>	27
3.3.2	<i>Solicitud de Obras</i>	28
3.3.3	<i>Donaciones</i>	28
3.3.4	<i>Pagos</i>	28
3.3.5	<i>Mantenimientos o Eventos</i>	29
3.3.6	<i>Licitaciones</i>	29
3.3.7	<i>Actas Compromiso</i>	30
3.3.8	<i>Solicitudes Internas</i>	30
3.4	ROLES.....	30
3.5	PROCESOS DE NEGOCIO.....	32

3.6	ANÁLISIS DE PROCESOS DE NEGOCIO	32
3.6.1	<i>Invitaciones</i>	32
3.6.2	<i>Solicitud de Obras</i>	32
3.6.3	<i>Donaciones</i>	33
3.6.4	<i>Pagos</i>	34
3.6.5	<i>Mantenimientos o Eventos</i>	35
3.6.6	<i>Licitaciones</i>	35
3.6.7	<i>Actas Compromiso</i>	36
3.6.8	<i>Solicitudes Internas</i>	36
3.7	DIAGRAMA DE FLUJOS DE PROCESO	37
3.7.1	<i>Modelos Objeto-Relación</i>	46
3.7.2	<i>Modelos Objeto-Comportamiento</i>	46
3.8	ASOCIACIÓN DE ROLES.....	48
3.9	ASOCIACIÓN DE ATRIBUTOS CON LAS ACTIVIDADES.....	50
3.10	CONCLUSIONES	50
CAPITULO IV.....		51
DISEÑO.....		51
4	DISEÑO.....	52
4.1	INTRODUCCIÓN	52
4.2	DISEÑO DEL SISTEMA	52
4.3	ETAPAS GENÉRICAS	52
4.3.1	<i>Describir subsistemas</i>	52
4.4	DISEÑAR OBJETOS.....	55
4.5	DISEÑAR MENSAJES	61
4.6	CONCLUSIONES	64
CAPITULO V		65
DESARROLLO		65
5	DESARROLLO.....	66
5.1	INTRODUCCIÓN	66
5.2	APLICACIÓN DE MODELOS DE CALIDAD	66
5.3	CONFIGURACIONES ESENCIALES	67
5.3.1	<i>Instalación y Configuración de Apache</i>	67
5.3.2	<i>Instalación y Configuración de PHP</i>	68
5.4	DESARROLLO DE LA BD EN SQL SERVER.....	68
5.5	DESARROLLO DEL CÓDIGO PHP.....	77
5.6	MANUAL DE PROGRAMADOR	77
5.6.1	<i>Tipos de Archivo</i>	77
5.6.1.1	Archivos DLL.....	77
5.6.1.2	Archivos PHP	79
5.6.1.2.1	Motor.....	79
5.6.1.2.2	Admin.....	83
5.6.1.2.3	Usuarios.....	85
5.6.1.3	Archivos JSP	87
5.6.1.4	Archivos SWF	88
5.6.1.5	Archivos CSS	88
5.7	CONCLUSIONES	88

CAPITULO VI.....	89
PRUEBAS	89
6 PRUEBAS.....	90
6.1 INTRODUCCIÓN	90
6.2 PRUEBAS DE UNIDAD.....	90
6.3 PRUEBAS DE INTEGRACIÓN	92
6.4 PRUEBAS DE VALIDACIÓN	94
6.5 PRUEBAS DE DESEMPEÑO	95
6.6 PRUEBAS DE ACEPTACIÓN	96
6.7 PRUEBAS DE INSTALACIÓN	96
6.8 CONCLUSIONES	96
CAPITULO VII.....	97
INSTALACION Y MANTENIMIENTO.....	97
7 INSTALACIÓN Y MANTENIMIENTO	98
7.1 INTRODUCCIÓN	98
7.2 INSTALACIÓN EN SERVIDOR WEB.....	98
7.3 REVISIONES FINALES	98
7.4 CONCLUSIONES	99
CAPITULO VIII.....	100
CONCLUSIONES Y RECOMENDACIONES	100
8 CONCLUSIONES Y RECOMENDACIONES.....	101
8.1 CONCLUSIONES	101
8.2 RECOMENDACIONES	102

Índice de Ilustraciones y cuadros

ILUSTRACIÓN 2-1. MODELO DE INFRAESTRUCTURA GROUPWARE.....	10
ILUSTRACIÓN 2-2. MODELO DE REFERENCIA, COMPONENTES E INTERFACES.....	12
ILUSTRACIÓN 2-3. DIAGRAMA DE FLUJO DE PROCESOS.	12
ILUSTRACIÓN 2-4. PATRÓN DE SECUENCIA	14
ILUSTRACIÓN 2-5. PATRÓN DIVISIÓN EN PARALELO.....	15
ILUSTRACIÓN 2-6. PATRÓN SINCRONIZACIÓN	15
ILUSTRACIÓN 2-7. PATRÓN ELECCIÓN EXCLUSIVA.....	15
ILUSTRACIÓN 2-8. PATRÓN SIMPLE MERGE	16
ILUSTRACIÓN 2-9. PATRÓN DE ELECCIÓN MÚLTIPLE.....	16
ILUSTRACIÓN 2-10. PATRÓN SINCRONIZAR MERGE.....	17
ILUSTRACIÓN 2-11. PATRÓN MULTI MERGE	17
ILUSTRACIÓN 2-12. PATRÓN DISCRIMINADOR.....	17
ILUSTRACIÓN 2-13. PATRÓN CICLOS ARBITRARIOS.....	18
ILUSTRACIÓN 2-14. PATRÓN TERMINACIÓN IMPLÍCITA.....	18
ILUSTRACIÓN 2-15. PATRÓN MÚLTIPLES INSTANCIAS SIN SINCRONIZACIÓN	19
ILUSTRACIÓN 2-16. PATRÓN DE ELECCIÓN EN DIFERIDO.....	19
ILUSTRACIÓN 2-17. PATRÓN DE RUTEO INTERPAGINADO PARALELO	19
ILUSTRACIÓN 2-18. HITOS.....	20
ILUSTRACIÓN 2-19. PATRÓN CANCELAR UN CASO	20
ILUSTRACIÓN 2-20. PATRÓN CANCELAR ACTIVIDAD.....	21
ILUSTRACIÓN 3-1. DIAGRAMA DE PROCESO DE INVITACIÓN	38
ILUSTRACIÓN 3-2. DIAGRAMA DE PROCESO DE SOLICITUD DE OBRAS.....	39
ILUSTRACIÓN 3-3. DIAGRAMA DE PROCESO DE DONACIONES	40
ILUSTRACIÓN 3-4. DIAGRAMA DE PROCESO DE PAGOS	41
ILUSTRACIÓN 3-5. DIAGRAMA DE PROCESO DE MANTENIMIENTOS O EVENTOS	42
ILUSTRACIÓN 3-6. DIAGRAMA DE PROCESO DE LICITACIONES	43
ILUSTRACIÓN 3-7. DIAGRAMA DE PROCESO DE ACTAS COMPROMISO	44
ILUSTRACIÓN 3-8. DIAGRAMA DE PROCESO DE SOLICITUDES INTERNAS	45

Índice de Anexos

Anexo 1 Encuestas.....	104
Anexo 2 Ficha de Procesos.....	105
Anexo 3 Ficha de Actividades.....	105
Anexo 4 Diagrama de Flujo del Conjunto de Procesos.....	106
Anexo 5 Modelo Objeto Relación.....	107
Anexo 6 Modelo Objeto Comportamiento.....	108
Anexo 7 Pruebas de Desempeño.....	109
Anexo 8 Cuadro Estadístico WAPT.....	111

Resumen

El presente trabajo pretende explicar la utilidad de un sistema workflow en el “Gobierno Provincial del Azuay”, incluir el concepto de flujos de trabajo, para poder armar estructuras de manejo de procesos y modelación de los mismos, utilizando técnicas de investigación de campo con el objetivo de recopilar información de primera mano, de los funcionarios que tratan con la información.

Las distintas herramientas colaborativas mostraron la necesidad de construir una herramienta administrativa, basada en reglas de negocio, omitiendo la necesidad de las personas indispensables y mejorando el tiempo de inducción de personal.

El lenguaje utilizado para el desarrollo demostró un gran avance y utilidad con respecto al sin número de tecnologías que hoy se encuentran en el mercado, entregando una calidad de servicio (QoS) bueno para el usuario final.

Abstract

This work intends to explain the usefulness of a workflow system in the Provincial Government of Azuay in order to create structures for handling and modeling processes by using field investigation techniques aimed at collecting first-hand information from the employees who deal with the information.

The different collaborative tools showed the need to construct an administrative tool based on business rules, omitting the necessity of indispensable people and improving staff's time of induction.

The language used for development demonstrated great progress and usefulness regarding the countless number of technologies that are found nowadays in the market, offering a good quality of service (QoS) to the final user.

INTRODUCCION GENERAL

En toda empresa ya sea pública o privada, existen procesos que se llevan a cabo para lograr un objetivo, estos son repetitivos, y, la mayoría de instituciones no dan la importancia necesaria para poder definirlos y optimizar tiempos, ya sea en producción o administración; lograr este objetivo no es tarea fácil ya que podemos encontrar un sin número de inconvenientes, como el rechazo al cambio, o Leyes que se encuentran vigentes en la república.

Esta idea de flujos predefinidos se ha cultivado desde ya hace un tiempo, ahora con la ayuda de la tecnología, el modelamiento de redes Petri y apoyados con la ingeniería existente, podemos crear herramientas que brinden la ayuda necesaria para que el empleado tenga mayor desenvolvimiento y el empleador obtenga resultados óptimos con grandes beneficios. Una de esas herramientas es la que hoy en día se denomina WorkFlow, sistemas diseñados para el movimiento de tareas y documentos a través de un proceso de trabajo es decir; WorkFlow es el aspecto operacional de un procedimiento de trabajo, teniendo en cuenta: Como están estructuradas las tareas, quien las ejecuta, como están sincronizadas, como fluye la información para el soporte del proceso que esta siendo monitorizado. Como la dimensión de tiempo es considerada en WorkFlow, WorkFlow considera el “troughput” como una medida aparte.

Con esto, el Workflow registra y mantiene procedimientos estándares, además registra y controla las alternativas de decisión monitoreando las actividades del proceso y administrando el procesamiento de excepciones para que el proceso montado en el flujo sea eficiente y correcto.

El WorkFlow automatizado mejora el proceso de negocio, aumenta ahorros asociados con esta mejora, provee facilidades para procesamiento paralelo o serial, automatizando la toma de decisiones, por lo que se lo asocia con un sistema de toma de decisiones.

La evaluación de recursos tanto técnicos como humanos es esencial para evaluar la productividad y potencialidad para crear transiciones entre tareas.

En general, los procesos WorkFlow funcionan cuando asuntos como el personal están encaminados a una mejora continua para hallar la expresión de calidad de servicio que esta institución esta siguiendo para demostrar un desempeño eficiente.

Estos puntos se sumarán con muchos otros en el avance del presente documento que sirven de sustento para desarrollar un producto que demostrará ser una solución a muchos de los pesares diarios cuando se trabaja con metodologías que todavía no emplean la tecnología como una herramienta de trabajo.

CAPITULO I
INVESTIGACION DE CAMPO

1 Investigación de Campo

1.1 *Introducción.*

La Investigación de Campo proporciona una colección de información, base fundamental de todo el trabajo, descripción de la institución, identificación de requisitos, definición de procesos objetivo clave para el desarrollo, y una comprensión de la problemática de la institución.

1.2 *La Institución*

El Gobierno Provincial del Azuay es una institución de derecho público, goza de autonomía y representa a la provincia.

Su misión es la de impulsar el desarrollo de la provincia en aspectos humanos, bienestar poblacional y culturales, además colaborar con el Estado y las municipalidades dentro de su división política para el progreso como nación.

Sus objetivos son:

Asumir responsablemente funciones y competencias que contribuyan al desarrollo descentralizado de la provincia y apoyen la desconcentración del Estado.

Concertar la participación interactiva de los distintos actores sociales de la Provincia: gobiernos locales, instituciones públicas, privadas, universidades y demás actores del desarrollo, dando continuidad al proceso de planificación emprendido por el H. Consejo Provincial.

Generar políticas provinciales en: ordenamiento e integración social del territorio, mancomunidades y cuencas hidrográficas; promoción de "una vida saludable"; conectividad y gestión del conocimiento (comunicación e información) y el fortalecimiento de capacidades locales.

Fomentar el crecimiento económico, las pequeñas y medianas empresas y el desarrollo de micro finanzas.

1.3 *El Gobierno Provincial del Azuay*

El Gobierno Provincial del Azuay en la última administración es una entidad que viene experimentado cambios a mejor, convirtiéndose en pionera en muchos aspectos dentro de la provincia, tal es el caso del aspecto vial, tecnológico y humano.

1.3.1 *Especificación de Requisitos*

Desde el inicio del proyecto, lo más importante para alcanzar el éxito es la identificación de los requisitos del sistema. Esto se debe a que no es un proceso que pueda ser determinado matemáticamente. Es un proceso en el cual los datos son extraídos de las personas y estos datos pueden variar, dependiendo de la persona a la cual estemos consultando. Es por eso que la Ingeniería de Requisitos ha trabajado arduamente para tratar de desarrollar técnicas que permitan hacer este proceso de una forma más eficiente y segura.

El propósito de la Gestión de Requisitos es el de administrar los requerimientos del producto del proyecto y componentes e identificar inconsistencias entre requerimientos y planes del proyecto y productos de trabajo.

Para la gestión de requisitos debemos seguir los siguientes pasos

- Educción de Requisitos
- Análisis y Negociación de Requisitos
- Especificación de Requisitos
- Validación de Requisitos

Para la obtención de requisitos tomamos los siguientes métodos:

- Preliminares: Utilizar preguntas libres de contexto.
- Brainstorming: Seleccionar un grupo variado de participantes. Eliminar críticas, juicios y evaluaciones mientras los participantes sugieren ideas. Producir muchas ideas. Recogerlas todas por escrito. Otro día, en otra sesión, se evalúan las ideas (se puntúan).
- Entrevistas: debemos utilizar con otras técnicas como complemento.
 - Entrevistar a la(s) persona(s) adecuada(s).
 - Preparar las preguntas con antelación.
 - Utilizar diagramas y modelos.

Ver Anexo 1: Encuestas. Pág. 104

1.3.2 Definición de Procesos.

Contar con los procesos administrativos adecuados es un objetivo para lograr los resultados esperados. La definición de procesos es una de las partes clave para poder demostrar el alcance del proyecto ya que estas serán el ejemplo claro de cómo podemos hacer la orquestación dentro de la institución.

El tema principal de algunas empresas de nuestro medio es la incapacidad de gestión y en poner en manos del personal el poder comunicar entre una sección y otra el flujo normal de trabajo por lo que existen incrementos innecesarios en la ejecución de los mismos, minimizando el potencial de los ejecutores.

Sabiendo que un proceso es la agrupación de una serie de acciones dirigidas a conseguir un resultado particular. Un grupo de procesos relacionados entre sí puede ser visto como un sistema. Existen muchos y variados procesos en el Gobierno Provincial del Azuay.

Para lograr este paso fundamental, está en marcha el desarrollo de un Manual de Procesos del Gobierno Provincial del Azuay, que tiene las siguientes interrogantes para lograr el desempeño esperado:

¿En donde están los cuellos de botella? ¿Qué queremos obtener del proceso? ¿Qué estamos obteniendo en realidad? ¿Qué debemos hacer para llegar de un punto a otro? ¿Qué proceso es necesario para alcanzar nuestras metas? ¿Cuáles son los pasos innecesarios? ¿Dónde tropezamos con problemas?

Cada área administrativa tiene un proceso específico, pero en la interrelación entre las áreas, es necesario no duplicar actividades sino más bien complementarlas. Para una definición de procesos se realizó una revisión de los presentados en este manual encontrando ciertas incongruencias, como procesos repetitivos y un diagrama de flujo de funciones cruzadas mal empleado.

1.3.3 Problemática de la Institución

La institución ha manejado desde su creación una manipulación administrativa bastante decadente y conforme pasaron los años se quiso modernizar a la institución mirando a futuro desde una óptica hardware, olvidando completamente el hecho de que debe existir una evolución software a la par.

El Gobierno Provincial del Azuay esta trabajando ya con un sistema tramitológico que ayuda al ordenamiento y administración de: Solicitudes de Obra, Donaciones, Auspicios, Invitaciones entre otras, asumiendo ya este cambio tecnológico necesario para afrontar los retos de esta era. Debemos considerar que este cambio por si solo no es suficiente, por lo contrario es una parte y sin un sistema de procesos de flujo de trabajo el software depende 100% del personal para su avance.

Las actividades realizadas dentro del Gobierno Provincial del Azuay son llevadas o atribuidas a la persona y no al puesto o cargo que debe realizarlo, por lo tanto se vuelve una dependencia total con el empleado y los cambios que se llevan a cabo generan problemas ya que los procesos que realizaban no pueden tomarlos y realizarlos otras personas sin dejar de sentir el tiempo hasta empaparse de la situación.

La falta de control de las actividades de la organización como un todo genera una falta de conocimiento por parte del jefe del trabajo realizado y avances prioritarios de procesos. La necesidad de una estandarización que soporte una reingeniería de procesos y una flexibilidad operacional.

1.4 Conclusiones

El trabajo realizado en el Gobierno Provincial del Azuay esta basado en ideas no acordadas dentro de una institución pública, ya que los procesos no se encuentran correctamente definidos y las responsabilidades asignadas a un cargo.

CAPITULO II
INVESTIGACION DE TECNICA

2 Investigación Técnica

2.1 *Introducción*

Conocer la opción que ofrece la tendencia tecnológica es un tema implícito del presente trabajo, la Investigación Técnica muestra el avance del WorkFlow Management y de su entorno con herramientas Groupware, mediante el diagrama de esta infraestructura.

2.2 *WorkFlow*

WorkFlow Management es una tecnología de rápida evolución que está progresivamente siendo explotada por negocios en una variedad de industrias. Su característica principal es la automatización de procesos involucrando combinaciones de actividades humanas y basadas en máquinas, particularmente esa envolvente interacción con aplicaciones de IT (Tecnología de la Información) y herramientas. Aunque su uso más prevaeciente está dentro del ambiente de la oficina en operaciones intensivas fijas de planta como el seguro, la banca, la administración legal y pública, etc., es también aplicable a algunas clases de industria y manufactura.

Una definición de WorkFlow es la facilitación computarizada o automatización de un proceso de negocio, en todo o parte¹

Los Objetivos principales del sistema WorkFlow son

- Mejorar la calidad
- Aumentar la calidad del trabajo
- Reducir costos en el proceso

El Workflow a menudo es asociado con el BPR (Business Process Re-engineering) que concierne al asentamiento, análisis, modelamiento, definición y subsecuente implementación operacional de la base del proceso de negocio de una organización (o de otras entidades de negocio). Sin embargo no todas las actividades BPR resultan en implementaciones WorkFlow, la tecnología workflow es a menudo una apropiada solución ya que esta provee la separación de procesos logísticos de negocio y su soporte operacional IT, disponiendo subsecuentes cambios a ser incorporados en reglas procedimentales definiendo procesos de negocio. Inversamente, no todas las implementaciones workflow necesariamente forman parte de un ejercicio BPR, por ejemplo implementaciones para automatizar un procedimiento de negocio existente.

Un Sistema de Administración Workflow provee automatización procedimental de un proceso de negocio para la administración de una secuencia de actividades y la invocación de apropiados recursos humanos y/o tecnologías de la información asociados con varias actividades.

¹ Definición de Workflow por el Workflow Management Coalition.

2.2.1 Concepto

Una aplicación WorkFlow es un sistema que completamente define, administra y ejecuta “workflows” a través de la ejecución de software cuyo orden de ejecución está conducido por un computador que es la representación de la lógica workflow¹.

Un proceso individual de negocio puede tener un ciclo de vida extendiéndose desde minutos a días (o aun meses), dependiendo de su complejidad y la duración de las actividades compositivas diversas. Tales sistemas pueden ser implementados de diversos modos, pueden usar una gran variedad de IT y de infraestructura de comunicaciones y pueden funcionar en un ambiente extendiéndose desde pequeños workgroups locales hacia Inter-Empresa.

2.2.1.1 Elementos

Muchos productos en el mercado IT han soportado aspectos de funcionalidad workflow por algunos años, recientemente se le ha reconocido la importancia por si mismo. La evolución de workflow tiene que ver con muchas áreas tecnológicas IT teniendo su importancia en cada aspecto para una variedad de actividades.

2.2.1.1.1 *Procesamiento de Imágenes*

El Procesamiento de Imágenes mantiene una estrecha relación con workflow, muchos procedimientos de negocio involucran interacción con información en papel, que puede necesariamente ser capturada como un archivo de imagen, esto a menudo es requerido cuando es compartido por un número de participantes para diferentes propósitos, dentro del proceso, posiblemente involucrando una interacción con otras herramientas IT, por consiguiente creando un requisito funcional workflow.

2.2.1.1.2 *Administración de Documentos*

La Administración de Documentos es una tecnología que concierne al manejo del ciclo de vida de documentos electrónicos. Incrementar, esto es incluir facilidades para el manejo de un repositorio de documentos distribuidos dentro de una organización como un recurso compartido con facilidad de rastreo de documentos (o mejor aun de parte de documentos) hacia individuos para acceder o actualizar información de acuerdo a roles específicos relacionados con documentos específicos. El documento puede formar parte de un procedimiento particular que requiere acceso al documento por un staff individual que emprende actividades por separado, de acuerdo a una secuencia particular en conformidad con algunas reglas.

2.2.1.1.3 *Correo Electrónico*

El correo electrónico provee una gran facilidad de comunicación entre individuos dentro de una organización o entre organizaciones; el uso de directorios administra una forma de ubicar o identificar participantes dentro de un dominio, sus atributos y roles. Así los sistemas de correo electrónico han ido progresando hacia

¹ Definición de una aplicación WorkFlow por el Workflow Management Coalition.

funcionalidad workflow a través de la adición de comandos de rastreo para definir una secuencia de destinatarios de tipos de correo particulares.

2.2.1.1.4 Aplicaciones Groupware

La industria groupware ha introducido un amplio rango de aplicaciones software diseñados a dar soporte y mejorar la interacción entre un grupo de individuos. Inicialmente muchas de estas aplicaciones soportaban mejoramiento en grupos de trabajo vía procedimientos informales, accediendo a un boletín o agendas sobre una base ad-hoc. Como el alcance de tales aplicaciones se ha extendido hacia un enfoque formal de negocio de interacciones de grupo, esta ha ido incrementando requerimientos para proveer un mayor marco de trabajo formal procesal de soporte al uso de aplicaciones groupware. La tecnología Workflow provee una solución a este tipo de requerimientos.

2.2.1.1.5 Aplicaciones basadas en transacciones

Por muchos años las aplicaciones han soportado ciertas clases de procedimiento de negocio (“transacciones”) que han sido desarrollados usando un administrador transaccional como un software de administración de base de datos.

Del estilo de trabajo inicial centralizado, tales aplicaciones han incrementado la disponibilidad de distribución de transacciones a través de un número de plataformas. Las aplicaciones basadas en transacciones típicamente exhiben importantes características de robustez y el soporte para propiedades atómicas de las transacciones; sin embargo, ellos no exhiben una separación entre la lógica de negocio y la invocación de varias aplicaciones que pueden ser requeridas para el soporte de actividades individuales dentro de los procesos de negocios.

Con el paso del tiempo la capacidades workflow han conducido a que controle el procedimiento de negocio y que posea la habilidad de invocar a aplicaciones transaccionales para asignar partes del proceso de negocio, como también a otros tipos de aplicaciones a otras partes del proceso de negocio.

2.2.1.1.6 Software de Soporte al Proyecto

El Software para manejar proyectos de desarrollo de aplicaciones IT complejas a menudo ha provisto una forma de funcionalidad workflow dentro del ambiente del proyecto para “transferir” el desarrollo de tareas entre individuos, y el rastreo de información entre individuos soportados en estas tareas. En algunos casos este tipo de software ha sido generalizado para soportar una visión más amplia, orientada a procesos de negocio y un rango más amplio de herramientas de aplicación, ofreciendo una mayor capacidad de workflow.

2.2.1.1.7 BPR y herramientas de diseño de sistemas estructurados

Las herramientas BPR han provisto IT basada en el soporte para actividades de análisis, modelamiento, y (re) definición de los procesos fundamentales de negocio de una organización y los potenciales efectos de cambio en tales procesos o roles organizacionales y responsabilidades asociadas con tales procesos. Esto puede incluir el análisis de la estructura de procesos y la información de flujos

soportados, los roles de individuos o unidades organizacionales dentro del proceso, y acciones tomadas en respuesta a diferentes eventos, etc. Una extensión natural de tales herramientas es facilitar la implementación de los procesos con infraestructura de soporte IT, control de flujos de trabajo y actividades asociadas dentro del proceso de negocio.

2.2.1.1.8 Separación de la funcionalidad Workflow

El mercado workflow involucró un amplio espectro dentro de la industria IT y probablemente continúe así, ya que existen una variedad de productos enfocados en aspectos particulares de requisitos workflow. Algunos están por ejemplo diseñados para propósito de procesamiento de imágenes u otros de propósito general. Es por ello que no existe un estándar bien definido para este marco de trabajo por lo que conlleva a un futuro integrar todos estos productos en una arquitectura consistente.

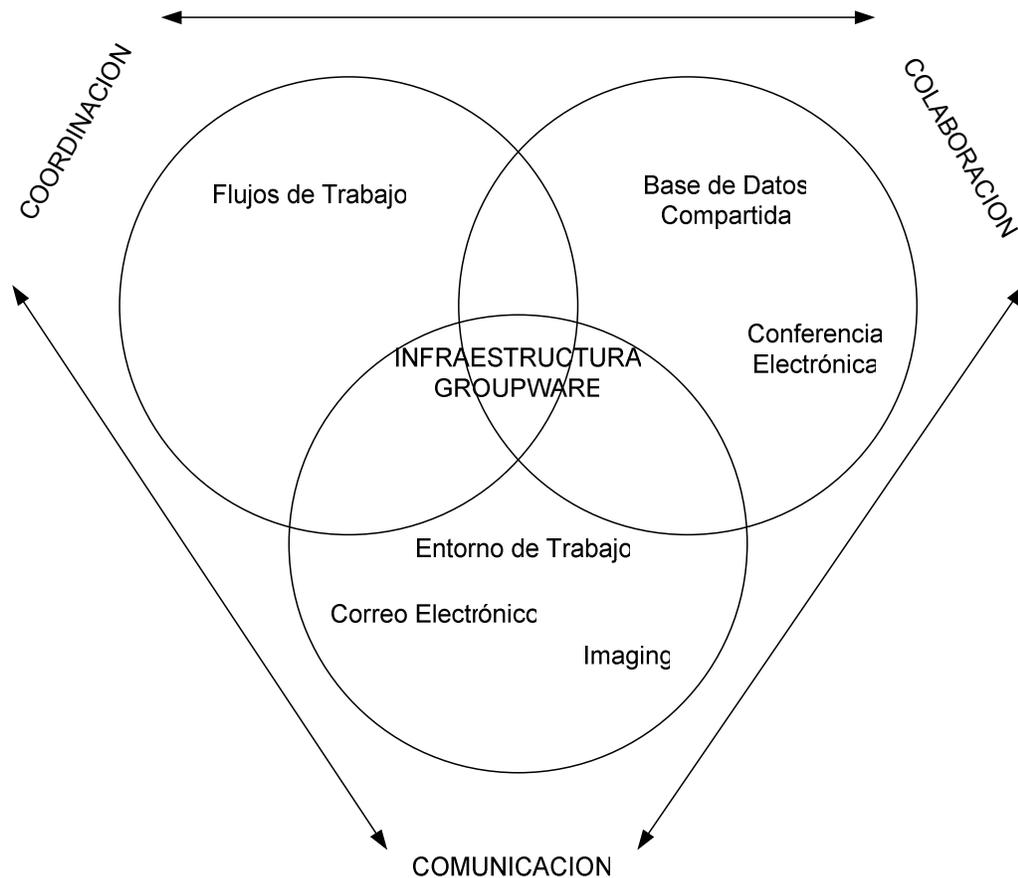


Ilustración 2-1. Modelo de Infraestructura Groupware

2.2.2 Elementos

El Workflow registra y mantiene procedimientos de operaciones estándares, además registra y controla las alternativas de decisión monitoreando las actividades del proceso y administrando el procesamiento de excepciones para que el proceso montado en el flujo sea eficiente y correcto.

Un modelo de referencia ha sido desarrollado de una estructura genérica para identificar interfaces dentro de una estructura que habilita productos para interoperar en una variedad de niveles. Todo sistema workflow contiene un número genérico de componentes dentro interactuando en un conjunto definido de formas; diferentes productos típicamente exhiben diferentes niveles de habilidad en cada uno de estos componentes genéricos. Conseguir interoperabilidad entre productos workflow un conjunto de interfaces estandarizados y formatos de intercambio de datos entre tales componentes es necesario.

Para lograr esta funcionalidad, un sistema Workflow debe poseer ciertos elementos que garanticen el refinamiento de los procesos de negocio.

2.2.2.1 Infraestructura Necesaria

Las Comunicaciones entre los individuos dentro del groupware son necesarias para ello están herramientas de mensajería (correo electrónico). Otra parte de la infraestructura es la Colaboración en entornos de trabajo con información y recursos compartidos y, por último la Coordinación de ambos entornos de trabajo en una herramienta de trabajo global.

2.2.2.2 Tecnologías Necesarias

Con respecto a la tecnología, Sistemas Workflow manejan contenedores dinámicos de objetos, modelos de acceso y distribución de la información y una infraestructura para el desarrollo de aplicaciones y procesos de negocio que comprende: Integración de recursos externos, No vinculación a una sola plataforma, Movilidad y Aplicaciones interempresariales.

2.2.2.3 Diferencias S.I.

Un Software Informático y uno WorkFlow plantean muchas diferencias al momento de su aplicación, es por ello que debemos diferenciar sus elementos base para continuar con el estudio de estas herramientas.

En un sistema tradicional existen elementos que definen el entorno en el que van a utilizarse como son: Entidades, Funciones, Flujos de Datos, Tablas Módulos, Columnas, etc. Sin embargo un sistema WorkFlow esta formado de otro tipo de elementos que son fundamentales y que sin ellos no podría constituirse como una herramienta de esta clase: Procesos, Caminos (Rutas), Roles, Reglas y Políticas.

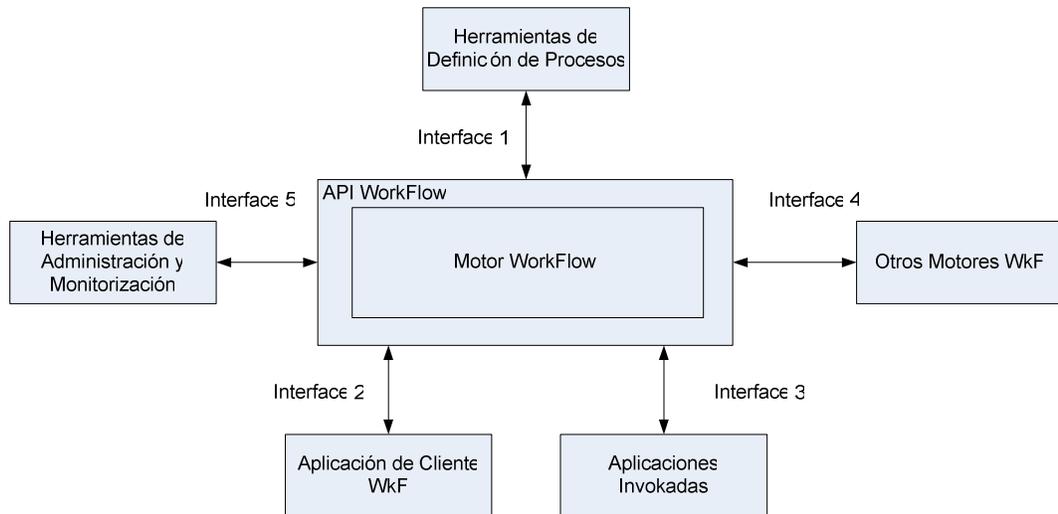


Ilustración 2-2. Modelo de Referencia, Componentes e Interfaces.

2.2.3 Análisis

En la fase de análisis establecemos una lista de los elementos antes mencionados para poder entender ampliamente el entorno. Debemos analizar en la institución lo siguiente: Lista de unidades de negocio, Lista de roles institucionales, Lista de procesos de negocio, y al final el Análisis de Procesos de Negocio y a su vez en estos intervienen: Rutas, Reglas Roles Implicados, Políticas y Eventos.

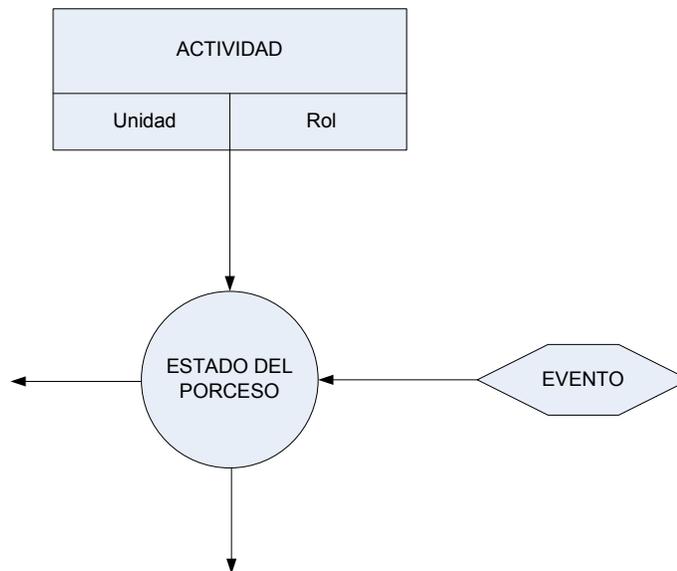


Ilustración 2-3. Diagrama de Flujo de Procesos.

Para la finalización de la fase de Análisis debemos realizar las siguientes actividades.

Asociar los roles con cada uno de los estados del proceso.

Relacionar las actividades de los proceso con cada una de las funciones de bajo nivel de los Diagramas de Flujos de Datos.

Relacionar la información tratada en los procesos de negocio con los conjuntos de datos de los Diagramas de Flujos de Datos.

2.2.4 Diseño

Como en todo desarrollo de software la fase de Diseño es una etapa importante para poder plasmar los subsistemas, el diseño parte de un nivel alto de abstracción donde se pueden seguir hasta requisitos específicos de datos, funcionales y de comportamiento, y a medida que ocurren iteraciones del diseño se da el refinamiento y se llega a una representación de mas bajo nivel.

Así, encontramos 3 etapas de diseño: de Datos, de Procesos, de Flujos de Trabajo. Las 2 primeras etapas entendemos como las que comúnmente se realizan y no son tema de este trabajo por lo cual profundizaremos en la tercera y ultima.

2.2.4.1 Diseño de Flujos de Trabajo

En un Sistema WorkFlow el diseño de flujos de trabajo comienza con la elección de la plataforma en la que podemos trabajar, elegir si es por desarrollo propio, la adquisición de entornos de desarrollo de aplicaciones WorkFlow y Elementos de Referencia: Número de aplicaciones a integrar y tecnología sobre las que se sustentan.

Identificación de Actividades.

- Módulos que se deben lanzar.
- Acciones automáticas a realizar.
- Información a presentar al usuario.
- Diseño de pantalla.

Identificación de Eventos.

- Módulos que deben originar eventos.
- Análisis de eventos automáticos

Aspectos de Seguridad

Herramientas de Control y umbral Escalado.

2.2.4.1.1 Elaboración de Fichas

Elaborar fichas es una actividad que ayudará a llevar un control sobre los procesos y actividades que se quieren incluir en el sistema para ellos elaboramos las siguientes reglas para su construcción.

Ficha de Proceso

- Identificación del proceso
- Objetivos
- Responsable
- Diagrama de flujo
- Información asociada a la instancia del proceso
 - Elemento de información

- Descripción
- Atributo, tabla y base de datos asociada
- Acciones asociadas a la instancia del proceso

Ver Anexo 2: Ficha de Procesos. Pág. 105

Ficha de Actividades

- Código de Actividad
- Descripción
- Periodicidad
- Roles que la llevan a cabo
- Información a visualizar
- Módulos a lanzar:
 - Aplicación
 - Identificación del módulo
 - Método de llamada
 - Parámetros a pasar
- Rutas
 - Identificador de ruta
 - Tipo de ruta: Automática, Manual o Escalado
 - Condición para enrutamiento automático
 - Condición necesaria para poder enrutar el proceso
 - Acciones a ejecutar al enrutar el proceso
 - Estado destinatario

Ver Anexo 3: Ficha de Actividades. Pág. 105

2.3 Patrones Workflow

2.3.1 Controles Básicos

Los controles básicos se interpretan como los caminos comunes en los que no implica alguna combinación o salto del flujo normal de los procesos.

2.3.1.1 Secuencia

Una actividad en un proceso workflow esta disponible solo después de completar la actividad anterior del proceso. También se le conoce como ruteo secuencial o ruteo serial.

El ruteo secuencial es usado para implementar modelos de pasos consecutivos y debe ser soportado por las herramientas administrativas workflow.



Ilustración 2-4. Patrón de Secuencia

2.3.1.2 División en Paralelo

Es algún punto del proceso workflow donde una rama se divide en múltiples ramas de control que pueden ser ejecutadas en paralelo, así permitiendo ejecutar

actividades simultáneamente en cualquier orden. Se lo conoce también como Divisores AND (AND-split).
Uno puede identificar dos avances básicos: AND-split explícitos e implícitos.

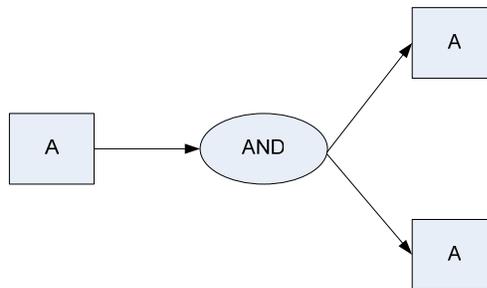


Ilustración 2-5. Patrón División en Paralelo

2.3.1.3 Sincronización

Es un punto en el proceso workflow donde múltiples subprocesos/actividades convergen en una sola rama. Es una suposición de este patrón que cada rama entrante de un sincronizador es ejecutada sólo una vez.

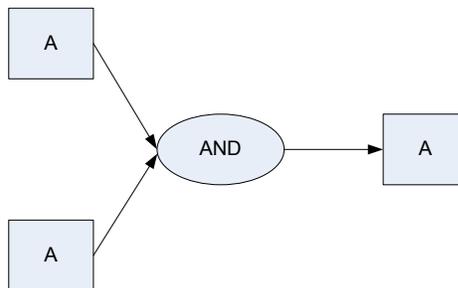


Ilustración 2-6. Patrón Sincronización

2.3.1.4 Elección Exclusiva

Es un punto en el proceso workflow donde, basados en una decisión o control de datos workflow, uno de las varias ramas es elegida. También se le conoce como XOR-split.

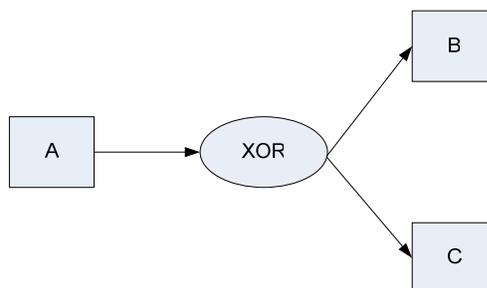


Ilustración 2-7. Patrón Elección Exclusiva

2.3.1.5 Simple Merge

Es un punto en el proceso workflow donde dos o mas ramas alternativas vienen sin una sincronización. Es una suposición de este patrón que ninguna de las ramas alternativas fueron ejecutadas en paralelo.

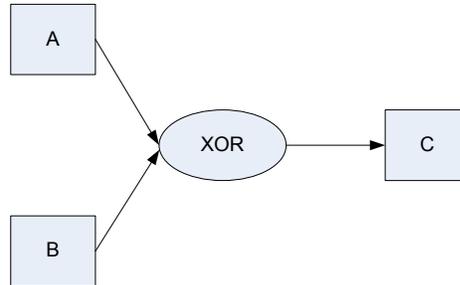


Ilustración 2-8. Patrón Simple Merge

2.3.2 Bifurcación avanzada y Sincronización

La bifurcación avanzada y sincronización trata de la unión de patrones básicos para resultados mas complejos como los que se detallan a continuación.

2.3.2.1 Elección Múltiple

Es un punto en el proceso workflow donde basados en una decisión o control de datos workflow un numero de ramas son elegidas. También se lo conoce como ruteo condicional, selector, OR-split.

Como se ha dicho, para los lenguajes del workflow que asignan condiciones de transición para cada transición la implementación de la múltiple elección es valida. El diseñador del workflow simplemente especifica condiciones deseadas para cada transición. Puede ser notable que el patrón de multielección generalice la División en Paralelo y la Elección Exclusiva.

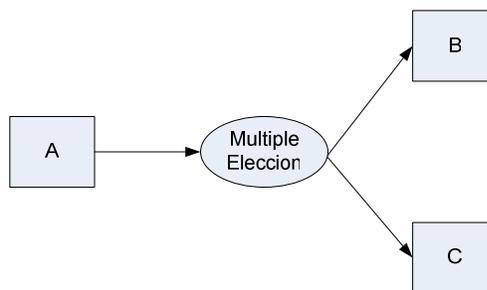


Ilustración 2-9. Patrón de Elección Múltiple

2.3.2.2 Sincronizar Merge

Es un punto en el proceso workflow donde múltiples caminos se unen en una simple rama. Si tenemos más de un camino se activa la sincronización de estas. Es una suposición de este patrón que una rama que ya ha sido activada, no puede ser activada otra vez mientras el merge está todavía esperando a que otras ramas se completen.

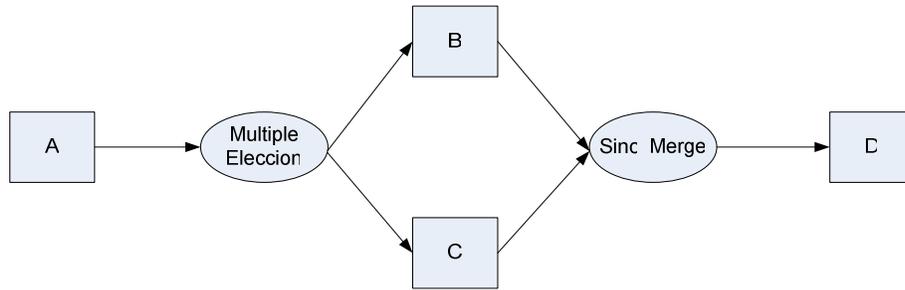


Ilustración 2-10. Patrón Sincronizar Merge

2.3.2.3 Multi-Merge

Es un punto en el proceso workflow donde dos o más ramas reconvergen en una sincronización. Si más de una fue activada, posiblemente concurrentemente, las actividades siguientes al merge comenzaron por cada activación de cada rama comenzada.

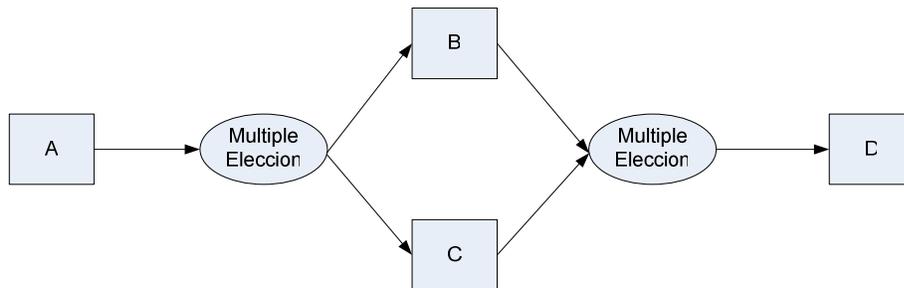


Ilustración 2-11. Patrón Multi Merge

2.3.2.4 Discriminador

El discriminador es un punto del proceso workflow que espera por una de las ramas entrantes para completar antes de activar las subsiguiente actividad. Para este momento el resto de ramas entrantes son ignoradas. Una vez todas las ramas entrantes han sido disparadas, se vuelve a arrancar a sí mismo a fin de que pueda desencadenarse otra vez (que es importante de otra manera realmente no podría ser usada en el contexto de un loop).

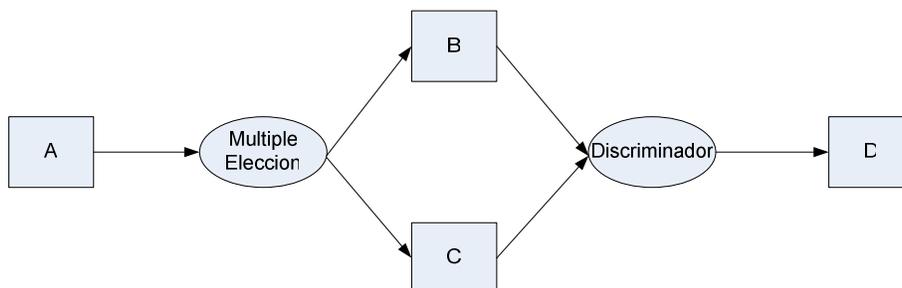


Ilustración 2-12. Patrón Discriminador

2.3.3 Patrones Estructurales

Como su nombre lo indica son patrones que conllevan a procesos en los cuales su acción tiene una razón implícita.

2.3.3.1 Ciclos arbitrarios

Es un punto en el proceso workflow donde una o mas actividades pueden realizarse en forma repetitiva. Los ciclos arbitrarios pueden convertirse en ciclos de estructura a menos que ellas contengan uno a más patrones avanzados tales como instancias múltiples.

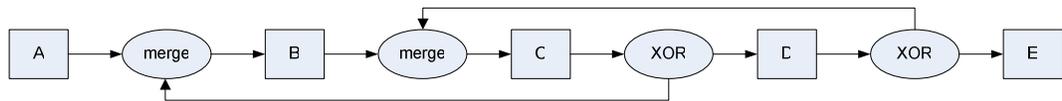


Ilustración 2-13. Patrón Ciclos Arbitrarios

2.3.3.2 Terminación Implícita

Un subproceso dado debería ser terminado cuando no hay nada mas que hacer. En otras palabras, si no hay actividades activas en el workflow y ninguna otra actividad puede ser activada (y al mismo tiempo el workflow no está en punto muerto).

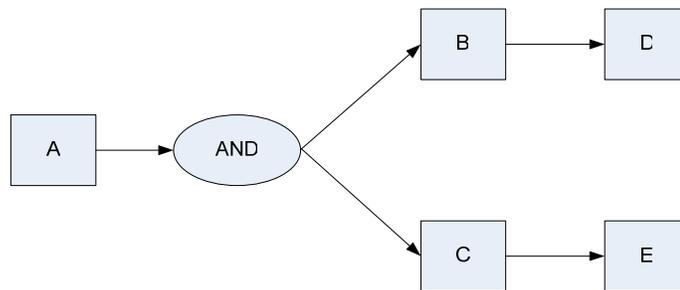


Ilustración 2-14. Patrón Terminación Implícita

2.3.4 Patrones Involucrando Múltiples Instancias

En estos patrones se involucra la creación de instancias en la ejecución del proceso workflow. A continuación se revisaran con más detalle cada uno de los procesos que forman parte de este.

2.3.4.1 Múltiples Instancias sin Sincronización

Dentro del contexto de un simple caso, las múltiples instancias pueden ser creadas, por ejemplo, hay la posibilidad de generar nuevas ramas de control. Cada una de estas ramas de control es independiente de las otras. Además no hay la necesidad de sincronizarlas.

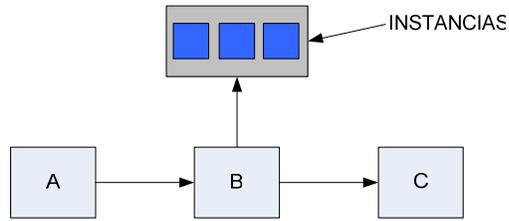


Ilustración 2-15. Patrón Múltiples Instancias sin Sincronización

2.3.5 Patrones basados en el estado

2.3.5.1 Elección en diferido

Es un punto en el proceso workflow donde varias ramas son seleccionadas. En contraste con el XOR-split, la elección no es hecha de manera explícita (basada en datos o decisiones), pero varias alternativas son ofrecidas al ambiente. Sin embargo, en contraste al AND-split, sólo una de las alternativas es ejecutada. Esto quiere decir que una vez que el ambiente activa una de las ramas las otras ramas alternativas son replegadas. Otros nombres para este patrón son: deferred XOR-split.

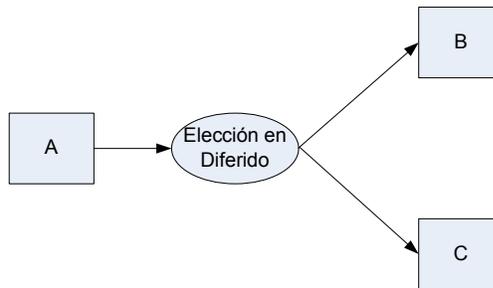


Ilustración 2-16. Patrón de Elección en diferido

2.3.5.2 Ruteo Interpaginado Paralelo

Un conjunto de actividades son ejecutadas en un orden arbitrario: cada actividad del conjunto es ejecutada, el orden se decide en tiempo de ejecución, y dos actividades no pueden ser ejecutadas al mismo tiempo. Este patrón no es posible de especificar desde que existe el AND-split y el AND-join.

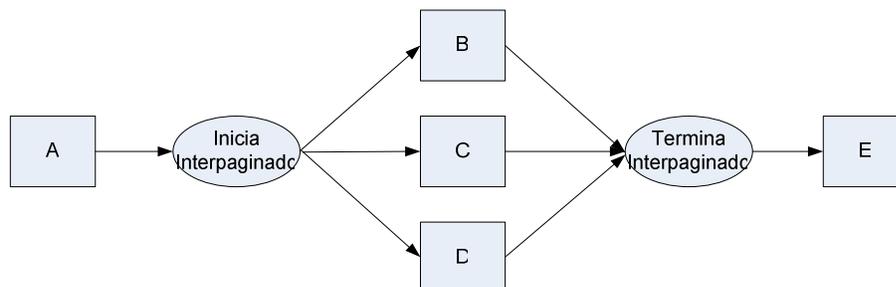


Ilustración 2-17. Patrón de Ruteo Interpaginado Paralelo

2.3.5.3 Hitos

La habilitación de una actividad depende en el caso de que se encuentre en un estado especificado, por ejemplo, la actividad es habilitada si un cierto hito ha sido cumplido y no expira todavía. Considerando tres actividades A, B y C. La actividad A es solo habilitada si la actividad B ha sido ejecutada y C no ha sido ejecutado todavía. Sinónimos de este patrón son: Test Arc, Línea Muerta.

Considerando tres actividades A, B y C. La actividad A puede ser ejecutada un arbitrario número de veces antes de la ejecución de C, y después de la ejecución de B. Tal hito puede ser llevado a cabo usando la Elección en Diferido. Después de ejecutar B hay un XOR-split implícito con dos posibles actividades siguientes: A y C. Si A es ejecutado el mismo XOR-split implícito es activado nuevamente. Si C es ejecutado entonces A es deshabilitado por la construcción del XOR-split implícito.

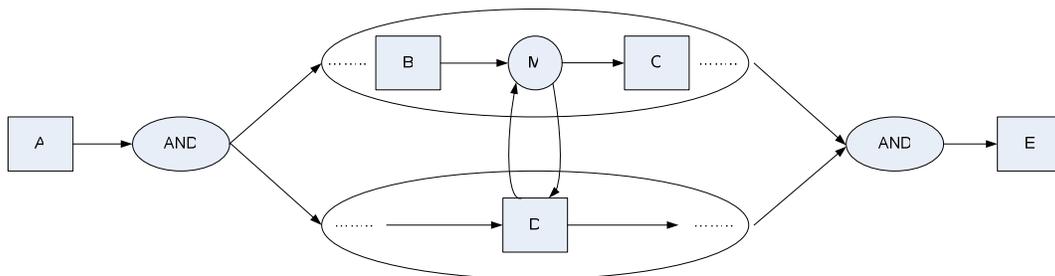


Ilustración 2-18. Hitos

2.3.6 Patrones de Cancelación

Los patrones de cancelación representan momentos en el proceso workflow en donde se deben hacer cancelaciones por decisiones o por actividades no completadas.

2.3.6.1 Cancelar un Caso

Un caso, por ejemplo una instancia workflow, es removida completamente (aun si parte de los procesos son instanciados varias veces, todo lo desencadenado es removido) Un sinónimo de este patrón en ingles es Withdraw case.

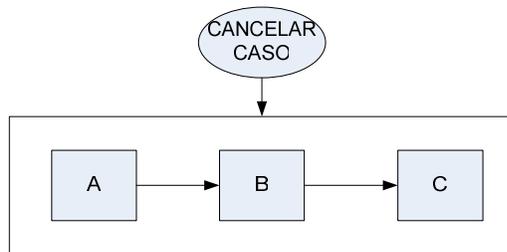


Ilustración 2-19. Patrón Cancelar un Caso

2.3.6.2 Cancelar Actividad

Una actividad habilitada es deshabilitada, por ejemplo, una rama esperando por la ejecución de una actividad es removida. Su sinónimo en inglés es *Withdraw activity*.

Si el workflow soporta la Elección en diferido, entonces es posible cancelar una actividad añadiendo una llamada a una “shadow activity”.

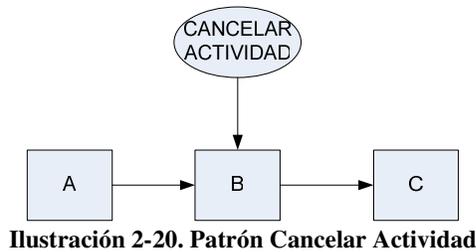


Ilustración 2-20. Patrón Cancelar Actividad

2.4 Reglas del Negocio

El evidente aumento de su importancia se ha puesto de manifiesto con la increíblemente rápida difusión de nuevos sistemas BMPS (Business Process Management Suite) que, siendo un nuevo paradigma de enfoque de la operativa institucional o empresarial, pueden ser notablemente mejorados si las reglas de negocio se mantienen fuera, independientes, claro en un íntimo contacto con los procesos.

Esto permite que los procesos puedan mantenerse inalterables, excepto los derivados de las mejoras introducidas en su diseño, ya que la mayor parte de los cambios se derivan del cambio del entorno empresarial (mercado, políticas, estrategia, etc.), que es justamente lo que queda definido en las Reglas de Negocio. Con este enfoque, los cambios se introducen en las Reglas de Negocio y los Procesos quedan automáticamente adaptados a las nuevas situaciones.

Para el establecimiento de procedimientos y estándares de creación y uso de las Reglas de Negocio, han aparecido y siguen apareciendo propuestas que se centran, bien en la semántica y sintaxis de las expresiones (prácticamente solo en idioma Inglés), bien en sistemas de definiciones de terminologías ligadas a repositorios de datos que puedan ser transformables a lenguaje XML y otros para la intercomunicación entre empresas.

Pero ¿Qué son realmente las Reglas de Negocio?

Para ello podemos citar a uno de los expertos mundialmente reconocidos en Reglas de Negocio:

“A business rule is simply a rule that is under business jurisdiction. Under business jurisdiction is taken to mean that the business can enact, revise, and discontinue their business rules as they see fit. If a rule is not under business

*jurisdiction in that sense, then it is not a business rule. For example, the 'law' of gravity is obviously not a business rule."*¹

Reglas de Negocio son los elementos individuales (atómicos) que permiten ser definidos, delimitados y expresados de forma inteligible y que en su conjunto componen el marco estructural, la política, la estrategia y la operativa de una empresa u organización.

Por su función, las Reglas de Negocio se dividen en Operativas, llamadas así porque su ejecución implica la vinculación personal y que por tanto pueden ser violadas (no cumplidas), y Estructurales, que son las que definen condiciones y tratamientos y que, al no depender de la acción de las personas, no pueden ser violadas. Un ejemplo de Regla Operativa es: Los obreros deben llevar casco. Evidentemente, si un obrero no se pone el casco, está violando esta regla. Un ejemplo de Regla Estructural es: Se considera Cliente VIP al que haya comprado más de 500 USD en el último año. Esta regla no puede ser violada. Podrá ser mal computada si el encargado de aplicarla se equivoca al consultar la ficha del cliente. Pero eso no es una violación sino un error.

Por su naturaleza las Reglas de Negocio pueden ser: Textuales (interpretables) y Mecánicas (automatizables). Las reglas Textuales muestran su contenido mediante expresiones de texto y normalmente aceptan algo de interpretación, aunque si están bien expresadas, su margen de desvío en la interpretación debe ser mínimo o nulo. Su aplicación siempre será realizada por una persona que es la que lee e interpreta el texto de la regla. Por el contrario, las reglas Mecánicas se expresan mediante fórmulas, tablas o expresiones matemáticas y por tanto pueden automatizarse. Su aplicación puede realizarse sin la intervención de personas si existe un dispositivo informático capaz de evaluarlas y ejecutarlas dentro del entorno operativo en el que actúan.

Como ejemplo de Regla Textual, tómese esta: *RN.101. Destino Inversión: Para la concesión del Préstamo es necesario, aunque no suficiente, que la inversión se pueda acoger a la legislación vigente sobre 'Desarrollos preferentes'*. Para valorar esta Regla, se necesita una persona que conozca la legislación vigente al respecto y pueda analizar si la inversión prevista la cumple o no. Por tanto, no puede automatizarse fácilmente sin la intervención humana.

Un ejemplo de Regla Mecánica es: *RN.236. Límite de Préstamo: El importe del Préstamo no puede ser superior a la Cantidad Límite. La Cantidad Límite se determina en la tabla adjunta mediante Criterios de Garantía y Disponibilidad de Fondos que se califican en las Reglas de Negocio indicadas*. Aquí sí es posible que de forma automática, sin intervención humana, y por tanto sin interpretaciones, se compare la cantidad solicitada (importe del Préstamo) con la Cantidad Límite expresada en la tabla adjunta. (Esta tabla contiene Criterios que se califican en otras Reglas de Negocio anidadas, que se indican en la misma).

Como se ha dicho, existen organizaciones dedicadas al estudio y establecimiento de estándares para la unificación de las Reglas de Negocio. En ese sentido,

¹ Ronald G. Ross. A definition of Business Rules

publicaciones como el SBVR (Semantics of Business Vocabulary & Business Rules) del OMG (Object Management Group) o el BRS RuleSpeak™ de Ron Ross son ‘de facto’ unas guías aceptadas por la comunidad interesada. Sin embargo, la documentación disponible al respecto adolece de dos importantes restricciones:

1. La semántica y la sintaxis de estas Reglas de Negocio se estudian a fondo y se proponen modos de expresión estructurados y literalmente correctos, pero no entran a proponer sistemas para la automatización de las mismas. Los sistemas para la automatización de la creación y aplicación de las Reglas de Negocio se dejan al criterio, conocimientos y posibilidades técnicas de las empresas que desarrollan software, con lo que cada entorno BPMS tiene su propio sistema (si es que lo tiene) de automatizar las Reglas de Negocio.
2. Solo se trabaja con el idioma Inglés. Los estudios y estándares se centran exclusivamente en ese idioma, y si bien la semántica y sintaxis de las Reglas de Negocio son traducibles a otros idiomas, pasará algún tiempo antes de que se disponga de bases aceptadas y extendidas para el tratamiento textual de aquellas en otros idiomas, de manera estandarizada.

2.4.1 Marco Estructural

Podría decirse que las Reglas de Negocio se encuentran siempre presentes en la actuación de una organización, bien de manera explícita (una política de salarios, el horario laboral, el descuento a aplicar en función de las condiciones de la venta, etc.) o de manera implícita no expresada (el trato cortés con los clientes, la responsabilidad del supervisor sobre sus supervisados, etc.) siempre implicando la participación directa o indirecta de personas.

Sin embargo, el término Reglas de Negocio queda reservado únicamente para aquellas reglas que revisten carácter explícito y que pueden ser y son, expresadas de manera entendible, registradas, localizables y modificables.

Las Reglas de Negocio deben definirse y mantenerse de manera independiente de los modelos y los procesos con los que la empresa funciona. No es que las Reglas de Negocio no tengan conexión con los procesos, mas bien al contrario, mantienen una relación íntima y constante. Sin embargo, su existencia y personalidad se deriva de la propia concepción de la empresa como ente económico-social y su misión es definir de manera granular sus políticas y modos operativos. Las Reglas de Negocio no están supeditadas a las definiciones o modelizaciones de los procesos ni a los cambios de éstos.

2.4.2 Políticas

Las políticas dentro de las Reglas de Negocio están definidas en dependencia de la institución a la que pertenezcan, es decir, si en una empresa de confección de joyas esta permitido el 10% de descuento a todos los clientes con más de 4 años, es una política que se encuentra como invariable y le ha dado una cierta posición en el mercado mayorista, pero no implica que un cambio en esta afecte el funcionamiento de la empresa.

Esta independencia frente a los procesos en cuanto a su definición y modificación es de vital importancia para los WorkFlow ya que permite que los procesos queden automáticamente actualizados con los cambios de políticas de la empresa sin necesidad de cambiar su modelo. Basta con cambiar las Reglas de Negocio a las que acceden.

2.4.3 Estrategias

Los términos Reglas de Negocios conllevan una conceptualización de importancia para toda empresa ya que llevarán las riendas en algunos aspectos importantes dentro de una institución. Es por ello que se deben establecer estrategias para la actualización de estas reglas y que estén acordes al mercado que se manejan, por ejemplo: establecer el cobro de interés en un 20 % anual representa una fuerte cantidad de dinero en USD, que en una moneda devaluada en una cantidad de 1 a 20000.

2.4.4 Operativa Institucional

Como técnica para una institución se deben poseer o generar manuales con las políticas que rigen a la empresa, aquí es de vital importancia el manejo de estos manuales y como son seguidos para la operatividad de la misma.

En la mayoría de los casos estos manuales contienen la construcción misma para el procesamiento workflow, ya que son realizados en base a estudios de la operatividad que manda en las actividades cotidianas y sus respectivas mejoras.

2.5 Intranet

La intranet es implantar la tecnología de Internet en una red interna de una organización, mas para una utilización interna que para una conexión externa, pero con posibilidad para ello. Eso resulta completamente transparente para el usuarios, pudiéndose acceder de forma individual, a todo el conjunto de recursos de la organización, con un coste mínimo tiempo y esfuerzo. Intranet e Internet no tienen diferencias mas que en su semántica.

Los factores que influyen en la utilización de la intranet son los siguientes:

- Coste asequible
- Fácil adaptación y configuración así como gestión y manipulación
- Sencilla integración de multimedia
- Disponible en todas las plataformas informáticas
- Posibilidad de integración con las bases de datos internas de la organización
- Utilización de estándares públicos y abiertos.

2.5.1 Ambiente WEB

Un ambiente WEB es un lugar sin barreras con un gran horizonte de información y técnicas que van mejorando conforme el tiempo transcurre, tenemos ya programación a nivel de capas, programación orientada a eventos como objetos, tenemos entornos completamente WEB que manejan tanto sistemas operativos con ejecuciones remotas, comercio electrónico, tanto como procesamiento de

información con ocultamiento. Las aplicaciones intranet tienen un rol inconfundible en estos momentos, especialmente en la operación empresarial interna, ahora todo esta a un clic.

2.5.2 Servidor HTTP Apache

El servidor apache es un software de desarrollo libre con un sin número de versiones, de código abierto para varias plataformas. Implementa el protocolo http y la noción de un sitio virtual.

Apache presenta características como mensajes de error configurables, varios sitios virtuales, dominios y subdominios, sin embargo no presenta un software de interfaz grafica para su configuración.

La arquitectura de apache se basa en módulos. Consta de una sección core, y de mucha funcionalidad que puede considerarse como base para un servidor Web.

2.5.3 SQL Server

La versión de SQL Server 2000 aporta todo lo necesario para facilitar una integración de sus datos en Internet, aparte de ser un servidor de datos, SQL Server ofrece, herramientas de análisis y gestión de almacén de datos.

La administración de este sistema RDBMS (Relational DataBase Management System) es sencilla y practica, no necesita de un conocimiento extra que en cualquier otro administrador de base de datos. Es altamente configurable siendo una de las bases de datos mas usada y confiable, a un precio mucho mas asequible que otras en el mercado.

2.6 Conclusiones

La tecnología WorkFlow es una herramienta que viene proporcionando beneficios a sus usuarios, ya que su evolución a hecho que pueda formar parte de procesos en la parte máquina y humana. El conjunto de herramientas que presta están acorde a los requisitos del trabajo por lo cual mediante una selección de patrones podremos armar la lógica de procesos que deberá ser sistematizada.

Por otro lado las reglas de negocio es un elemento vital y su definición es de vital importancia estas a su vez pueden se Operativas o Mecánicas, esto dependerá del contexto del proceso.

CAPITULO III
ANALISIS

3 Análisis

3.1 Introducción

El Análisis muestra la diagramación de cada uno de los procesos del Gobierno Provincial del Azuay, unidades de negocio, sus reglas, políticas, y su cooperativismo plasmado en los diagramas Objeto Relación, Objeto Comportamiento, y la definición de los roles con su respectiva asociación.

3.2 Descripción de la Situación Actual

El Gobierno Provincial del Azuay se encuentra en un periodo de cambio construyendo nuevas formas de hacer las cosas y tomando formas que se han comprobado ser efectivas, es por ello que en lo que respecta a tecnología y administración no pueden quedarse rezagados. La implementación de procesos mediante flujos de trabajo no es solo una opción sino una necesidad que nace de poder satisfacer a los clientes en sus actividades diarias.

Además el exagerado papeleo, procesos repetitivos y la innegable burocracia vuelven de trámites sencillos una verdadera odisea. Todos los directores concuerdan que la implementación de flujogramas dará una solución. De cualquier forma esta solución sería temporal ya que dejar ciertos procesos en manos de personas produce burocracia, entonces se expone el tema a una nueva solución, una herramienta que pondrá fin a estos problemas, que podrá responder al público sus inquietudes y que responderá de manera satisfactoria las expectativas de la gente.

Para poder obtener este resultado necesitamos un riguroso estudio de cada aspecto de la entidad y representar esto en cada sección de las que veremos a continuación.

3.3 Unidades de Negocio

Un workflow tiene una íntima relación, como mencionamos anteriormente, con las actividades de la empresa en la cual va a ser utilizado, esta íntima relación nos brinda las bases para poder realizar nuestro trabajo. Como en todo trabajo una de las mejores tácticas es divide y vencerás, este caso no es la excepción por lo que definiremos las unidades de negocio de cada actividad para en un siguiente tema armar a los mismos.

3.3.1 Invitaciones

Los procesos de Invitaciones nacen de comunidades, instituciones, o gente en particular que cree necesaria o por un gesto de amistad invitar a la máxima autoridad a ciertos eventos. Este es un proceso que trabaja con pocas unidades de negocio:

- Ingreso de Invitación

- Realizar Oficio de Delegación
- Adjuntar Informe
- Elaborar Excusa

Estas unidades de negocio son de tipo textuales, por lo que se registrarán a la voluntad de una persona.

3.3.2 Solicitud de Obras

Las comunidades dentro de la Provincia tienen una necesidad y esta debe ser atendida por el Prefecto, es aquí en donde el proceso comienza, teniendo las siguientes unidades de negocio:

- Ingreso de Solicitud
- Archivar Originales
- Coordinar Solicitudes
- Verificar POA (Planificación Operativa Anual)
- Designar Técnico
- Adjuntar Informe
- Analizar, Clasificar y Recomendar
- Realizar Términos de Referencia
- Legalizar Documentos
- Archivar Documentos
- Elaborar Respuesta

3.3.3 Donaciones

De la misma manera que en las solicitudes de obras las comunidades presentan necesidades que se rigen por montos de dinero. Sus unidades de negocio son:

- Ingreso de Solicitud
- Archivar Originales
- Coordinar Solicitudes
- Verificar POA
- Designar Técnico
- Adjuntar Informe
- Analizar, Clasificar y Recomendar
- Realizar Términos de Referencia
- Legalizar Documentos
- Archivar Documentos
- Elaborar Respuesta

3.3.4 Pagos

El proceso inicia cuando un producto no se tramita por solicitud de compras ya que en el medio no existe competencia de empresas para tales necesidades, este tipo de procesos no requiere de mucha intervención ya que debe ejecutarse una actividad tras otra.

Veamos sus unidades:

- Realizar Oficio
- Realizar Compromiso Presupuestario
- Elaborar Registros Contables
- Control Previo Contabilidad
- Legalizar Documentos
- Control Previo Tesorería
- Emitir Retenciones
- Existencia de Dinero
- Realizando Transferencia
- Legalizar el Pago

3.3.5 Mantenimientos o Eventos

Estos proceso surgen internamente de necesidades de cualquiera de las direcciones del Gobierno Provincial del Azuay, estas pueden ser por cambios estructurales físicos, necesidad de reconstrucción de muebles o necesidades de limpieza, además se tramitan peticiones de cafetería o buffet. Todo esto tiene que ser gestionada por Servicios Generales. Sus unidades son:

- Realizar Solicitud
- Registro de Caja Chica.
- Ingreso de Retención.
- Disponer a Servicios Generales
- Realizar Solicitud de Requerimientos.
- Registrar Ingreso del Bien
- Implementar el Bien o Servicio

3.3.6 Licitaciones

Inician en el momento de requerir compras en la que existen contratos de por medio para una competencia justa de las distintas empresas que puedan ofrecer el bien o servicio. Las unidades de negocio que conforman son las siguientes:

- Enviar Términos de Referencia
- Verificar Documentos
- Nombrar Fiscalizador
- Designar Fiscalizador
- Elaborar Borrador
- Designar Técnico (Comisión)
- Documentos para Publicación
- Elaborar Contrato
- Archivar Documentos

3.3.7 Actas Compromiso

Las Actas Compromiso se originan en el desarrollo del POA y presentan las siguientes unidades de negocio:

- Enunciado POA
- Registrar Datos
- Elaborar Acta Compromiso
- Realizar Oficio de Delegación
- Asistir
- Adjuntar Informe
- Archivar Documentos

3.3.8 Solicitudes Internas

Son procesos generados por peticiones de cualquier género y que no se enmarquen en procesos anteriores, están van dirigidas al prefecto. Sus unidades de negocio son:

- Ingresar Oficio
- Archivar Oficio
- Emitir Criterio

3.4 Roles

La definición de roles en un proceso ayuda a dejar a un lado la posesión de un cargo como propio, al contrario de este pensamiento, deja al cargo por si solo y se encomienda a este, las actividades a realizar. Para cada proceso tenemos unas actividades, que deberán ser ejecutadas por ciertos cargos algunas comunes para todos, en esta parte del capítulo listaremos los roles principales para los procesos antes mencionados.

Invitaciones

- Recepcionista
- Coordinador Administrativo de la Prefectura
- Prefecto
- Secretario General
- Jefe de Archivo

Solicitud de Obras

- Recepcionista
- Jefe de Archivo
- Coordinador Administrativo de la Prefectura
- Prefecto
- Director de Planificación
- Técnico
- Director de Asesoría Jurídica
- Secretario General

Donaciones

- Recepcionista
- Jefe de Archivo
- Coordinador Administrativo de la Prefectura
- Prefecto
- Secretario General
- Director de Planificación
- Técnicos

Pagos

- Directores o Jefes de Sección
- Director Administrativo
- Director Financiero
- Jefe de Presupuesto
- Contadores
- Contador General
- Tesorero

Mantenimientos o Eventos

- Funcionarios
- Encargado de Caja Chica
- Director Administrativo
- Jefe de Servicios Generales

Licitaciones

- Directores
- Jefe de Archivo
- Prefecto
- Director de Planificación
- Director de Asesoría Jurídica

Actas Compromiso

- Director de Planificación
- Director de Asesoría Jurídica y asesores
- Coordinador Administrativo
- Prefecto
- Secretario General
- Jefe de Archivo

Solicitudes Internas

- Directores o Jefes de Sección o Cámara
- Jefe de Archivo
- Secretario del Prefecto
- Prefecto

3.5 *Procesos de Negocio*

Los procesos de negocio son la parte fundamental y razón de existir del WorkFlow, ahora vamos a listar los procesos ya que los hemos mencionado en los puntos anteriores.

- Invitaciones
- Solicitud de Obras
- Donaciones
- Pagos
- Mantenimientos o Eventos
- Licitaciones
- Actas Compromiso
- Solicitudes Internas

Cada proceso fue analizado como prioridad para la entidad estos incluyen los roles mas importantes y mas impulsores para la provincia.

3.6 *Análisis de Procesos de Negocio*

En el análisis de procesos de negocio trataremos 5 temas principales Rutas, Reglas, Roles Implicados, Políticas, Eventos; tomando los procesos citados para el desarrollo del tema.

3.6.1 *Invitaciones*

Las invitaciones se manejaran por 3 rutas:

Primera Ruta: El documento tiene que pasar por la Recepción, luego se ingresa a la agenda para que el Prefecto Asista al evento.

Segunda Ruta: El documento ingresa por Recepción, luego se ingresa a la agenda y a continuación se Delega a un funcionario y este Adjunta un Informe.

Tercera Ruta: El documento ingresa por Recepción, se deja fuera de agenda y se elabora una excusa.

Las Reglas a aplicar a este proceso son textuales ya que se basan en decisiones del personal, la primera se define en base a si se ingresa o no en la agenda por lo que queda establecida así: “Se ingresa en la agenda siempre y cuando este acorde a la agenda del Prefecto” y la denominaremos Verificar Disponibilidad. La segunda regla es en base a una confirmación de asistencia: “El Prefecto decide asistir al evento o puede delegar a un funcionario como su representante” y la llamaremos Confirmar Asistencia.

Las políticas aplicadas en este proceso tienen que ver con la decisión del Prefecto de su asistencia y reza de la siguiente manera: “Se asistirá al evento siempre y cuando se requiera de su presencia y su prioridad sea alta”.

El único evento que genera este proceso es al momento de elaborar una excusa, se enviara un mail a la persona o entidad interesada.

3.6.2 *Solicitud de Obras*

Para este análisis existen 3 rutas para llegar al fin del proceso:

Primera Ruta: Se Ingresará la Solicitud por recepción para luego ser archivada en Archivo General, luego se procede a Coordinar las solicitudes para poder Priorizar las Obras a continuación se Verifica en el POA en la Dirección de Planificación para luego la Dirección de Infraestructura Vial Designar un Técnico que realice la inspección y Elabore un Informe, con este informe se procede a Analizar, Clasificar y Recomendar al Prefecto, en esta instancia el Prefecto lo toma como Prioritario siempre y cuando el monto sea mayor a USD 2500 y ordena al Jurídico que elabore en base a los Términos de Referencia, continua con la Legalización de Documentos y se Archiva los mismos.

Segunda Ruta: Se Ingresara la solicitud por recepción para luego ser archivada en Archivo General, luego se procede a Coordinar las solicitudes para poder Priorizar las Obras a continuación se Verifica en el POA en la Dirección de Planificación para luego Designar un técnico que realice la inspección y Elabore un Informe, con este informe se procede a Analizar, Clasificar y Recomendar al Prefecto, el decide no priorizar, y se procede a Elaborar un Oficio de Respuesta adjunto el informe técnico.

Tercera Ruta: Se Ingresara la solicitud por recepción para luego ser archivada en Archivo General, luego se procede a Coordinar las solicitudes para poder Priorizar las Obras, el Prefecto revisa y decide No Priorizar la solicitud por lo que se deja para el POA del próximo año y se procede a Elaborar Oficio.

Se necesita aplicar 3 reglas de negocio para poder tomar una ruta necesaria, la primera se basa en el monto de la obra “El monto de la solicitud deberá ser mayor a 2500” y la conoceremos como Monto de la Obra, la segunda se trata de la priorización “El Prefecto prioriza si y solo si la obra no esta en un lugar donde anteriormente ya se han realizado un número de obras de pendiendo de su criterio” la llamaremos Priorizar Obra, y la tercera regla trata de la priorización con un informe técnico “La Solicitud es aprobada siempre y cuando cuente con un informe positivo del técnico” y será mencionada como Priorizar con Informe.

Los eventos que generan este proceso son e-mail enviados a los responsables de la solicitud cuando ha sido aceptada o negada.

3.6.3 Donaciones

Primera Ruta: Se Ingresara la solicitud por recepción para luego ser archivada en Archivo General, luego se procede a Coordinar las solicitudes para poder Priorizar las Obras a continuación se Verifica en el POA en la Dirección de Planificación para luego Designar un técnico que realice la inspección y Elabore un Informe, con este informe se procede a Analizar, Clasificar y Recomendar al Prefecto, en esta instancia el Prefecto lo toma como Prioritario y ordena al Jurídico que elabore en base a los Términos de Referencia, continua con la Legalización de Documentos y se Archiva los mismos.

Segunda Ruta: Se Ingresara la solicitud por recepción para luego ser archivada en Archivo General, luego se procede a Coordinar las solicitudes para poder Priorizar las Obras a continuación se Verifica en el POA en la Dirección de Planificación para luego Designar un técnico que realice la inspección y Elabore un Informe, con este informe se procede a Analizar, Clasificar y Recomendar al Prefecto, el decide no priorizar, y se procede a Elaborar un Oficio de Respuesta.

Tercera Ruta: Se Ingresara la solicitud por recepción para luego ser archivada en Archivo General, luego se procede a Coordinar las solicitudes para poder Priorizar las Obras, el Prefecto revisa y decide No Priorizar la solicitud por lo que se deja para el POA del próximo año .

Se necesita aplicar 3 reglas de negocio para poder tomar una ruta necesaria, la primera se basa en el monto de la obra “El monto de la solicitud deberá ser menor a 2500 USD” y la conoceremos como Monto de la Obra, la segunda se trata de la priorización “El Prefecto prioriza si y solo si la obra no esta en un lugar donde anteriormente ya se han realizado un número de obras dependiendo de su criterio” la llamaremos Priorizar Obra, y la tercera regla trata de la priorización con un informe técnico “La Solicitud es aprobada siempre y cuando cuente con un informe positivo del técnico” y será mencionada como Priorizar con Informe.

Los eventos que generan este proceso son e-mail enviados a los responsables de la solicitud cuando ha sido aceptada o negada.

3.6.4 Pagos

Para los pagos existen tres rutas las que describiremos a continuación:

Primera Ruta: Se realiza un oficio dirigido al Director Administrativo solicitando el pago de un servicio, este ordena su ejecución, en la Dirección Financiera se Autoriza el Gasto y se envía a la sección de Presupuesto en donde se realiza el Compromiso Presupuestario, para luego enviar los documentos a la sección de Contabilidad y poder realizar el Control Previo y Registros Contables, una vez realizado estas actividades se procede Legalizar los Documentos, inmediatamente se envía a Tesorería para que ellos Emitan Retenciones y así puedan Efectuar la Transferencia y proceder finalmente a Legalizar el Pago.

Segunda Ruta: Se realiza un oficio dirigido al Director Administrativo solicitando el pago de un servicio, este Ordena el Gasto, en la Dirección Financiera se Niega el Gasto, Finaliza el Proceso.

Tercera Ruta: Se realiza un oficio dirigido al Director Administrativo solicitando el pago de un servicio, este Niega el Gasto, Finaliza el Proceso.

Las reglas de negocio que necesita este proceso son: “El Director Administrativo Ordenara el Gasto si considera que la petición puede entrar dentro de lo Presupuestado.” la denominaremos Ordenar el Gasto, “El Director Financiero Autorizará el Gasto si la cantidad presupuestada esta fuera del Plan de Participación y Presupuestación” la denominaremos Autorizar el Gasto, “El Tesorero verificará la disponibilidad de dinero”

Las políticas para los pagos son:

Para problemas con la documentación: “Se realizara un control sobre la documentación en las secciones de Contabilidad y Tesorería si existe alguna anomalía se encargaran de solicitar la corrección y los documentos permanecerán en su custodia”. Esta es usada por las actividades de Control Previo.

El evento que se genera en estos casos es al momento del Control Previo cuando se considera que el error es en una factura, se envía un mail al proveedor y al funcionario que generó el tramite.

3.6.5 Mantenimientos o Eventos

Contienen 3 rutas para el proceso y son:

Primera Ruta: Surge la necesidad por un mantenimiento o evento por lo que el funcionario decide Solicitar al Director Administrativo, Analiza la Solicitud y procede con la revisión del Monto y decide tramitarlo por Caja Chica, el funcionario recibe el dinero por lo que se procede a la Compra del bien o Servicio e Ingresas las Retenciones del caso.

Segunda Ruta: Surge la necesidad por un mantenimiento o evento por lo que el funcionario decide Solicitar al Director Administrativo, Analiza la Solicitud y procede con la revisión del Monto y Dispone a Servicio Generales, esta sección Realiza una Solicitud de Requerimientos del bien o servicio solicitado, una vez comprado Bodega General Registra el Bien Recibido y Servicios Generales procede a Implementar el Bien o Servicio.

Tercera Ruta: Surge la necesidad por un mantenimiento o evento por lo que el funcionario decide Solicitar al Director Administrativo, Analiza la Solicitud y rechaza el pedido.

Para el proceso según el estudio realizado vamos a necesitar las siguientes reglas de negocio: “El monto necesario deberá ser el establecido por la ley de lo contrario se procederá mediante una Solicitud de Requerimientos” al cual denominaremos Verificar Monto, “El Director Administrativo se guiará por su criterio al momento de recibir una solicitud” y lo llamaremos Analizar Solicitud.

No existen políticas con respecto a este proceso.

Se genera un evento al momento de rechazar el pedido del mantenimiento o evento por lo cual se enviará un mail al funcionario.

3.6.6 Licitaciones

Se generan dos rutas en este proceso:

Primera Ruta: Este proceso comienza con el envío de los Términos de Referencia a Archivo General, estos Verifican Documentos, archivan para luego enviarlos a la prefectura, El prefecto Autoriza e inmediatamente envía la documentación a la Dirección de Planificación realiza el estudio respectivo, Nombra un Fiscalizador y envían la información a Prefectura, en ese momento el prefecto Designa al Fiscalizador, se receipta los documentos en la Dirección de Asesoría Jurídica para elaborar el Borrador del Contrato y solicita el Compromiso Presupuestario, nuevamente los documentos llegan a Prefectura para que el prefecto Designe un Técnico para formar la comisión, la Dirección de Asesoría Jurídica prepara los documentos para la Publicación y finalmente Elabora el Contrato, Archivo General archiva los documentos.

Segunda Ruta: Este proceso comienza con el envío de los Términos de Referencia a Archivo General, estos Verifican Documentos, archivan para luego enviarlos a la prefectura, El prefecto rechaza y envía los documentos devuelta a Archivo General estos comunican la decisión.

Las reglas de negocio necesarias “El Prefecto autoriza una Licitación si considera conveniente” lo conoceremos como Autorizar Licitación.

No existen políticas en este proceso.

Se generan eventos de envío de un mail en las designaciones de Fiscalizador y Técnico, al momento de su elección. Se genera un evento cuando el prefecto rechaza esta licitación, en este momento se envía un mail al funcionario que comenzó el trámite.

3.6.7 Actas Compromiso

Las Actas compromiso presentan dos rutas:

Primera Ruta: Nacen de los Enunciados del POA, en la Dirección de Planificación Registran los Datos necesarios para ser enviados a la Dirección de Asesoría Jurídica para que Elaboren el Acta Compromiso, luego se ingresa a la agenda y a continuación se Delega a un funcionario y este Adjunta un Informe, se procede a Archivar Documentos.

Segunda Ruta: Nacen de los Enunciados del POA, en la Dirección de Planificación Registran los Datos necesarios para ser enviados a la Dirección de Asesoría Jurídica para que Elaboren el Acta Compromiso, se ingresa a la agenda y asiste el Prefecto, se procede a Archivar Documentos.

Las reglas de negocio para este proceso son: la primera se define en base a si se ingresa o no en la agenda por lo que queda establecida así: “Se ingresa en la agenda siempre y cuando este acorde a la agenda del Prefecto” y la denominaremos Verificar Disponibilidad. La segunda regla es en base a una confirmación de asistencia: “El Prefecto decide asistir al evento o puede delegar a un funcionario como su representante” y la llamaremos Confirmar Asistencia.

No existen políticas o eventos relacionados a este proceso.

3.6.8 Solicitudes Internas

Tenemos dos rutas en este proceso:

Primera Ruta: Comienza en cualquiera de las direcciones o jefaturas al Elaborar un Oficio, se envía a Archivo General para que se archive, y a continuación en Prefectura, el prefecto emite su criterio y se envía al Archivo, archivo da a Conocer su Respuesta al involucrado.

Segunda Ruta: Comienza en cualquiera de las direcciones o jefaturas al Elaborar un Oficio, se envía a Archivo General para que se archive, y a continuación en Prefectura, el prefecto emite su criterio y se envía al Archivo, y se da a conocer la respuesta al involucrado, el cual reinicia el proceso.

Las reglas de negocio para este proceso son sencillas: “Se reiniciará el proceso cuando el criterio del Prefecto ha sido negativo a la solicitud”. Esta regla la llamaremos Conocer Criterio.

No existen políticas o eventos relacionados al proceso.

3.7 Diagrama de Flujos de Proceso

Los diagramas de flujo muestran las relaciones entre las distintas actividades de cada uno de los procesos y principalmente los caminos posibles hacia una situación final, del mismo modo cada uno de los procesos tienen una relación entre estos para poder finalizar y dar un resultado efectivo a la población.

Ver Anexo 4: Diagrama de Flujo del Conjunto de Procesos. Pág. 106

En este punto elaboraremos un diagrama de flujo de cada uno de los procesos, estos podrán ser reflejados en el software y alterados por el mismo, es decir los diagramas pueden ser realizados únicamente con la información de cada proceso.

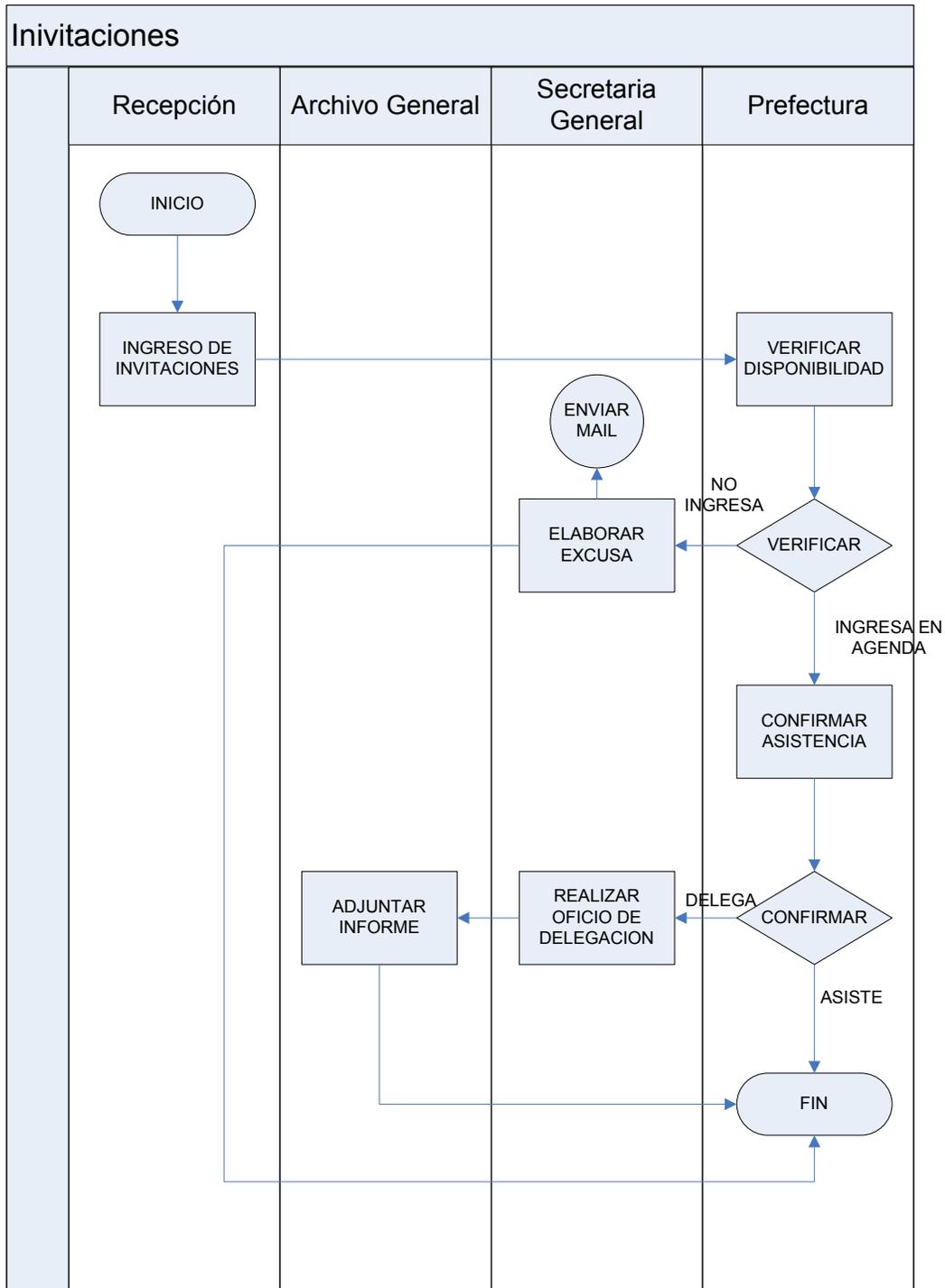


Ilustración 3-1. Diagrama de Proceso de Invitación

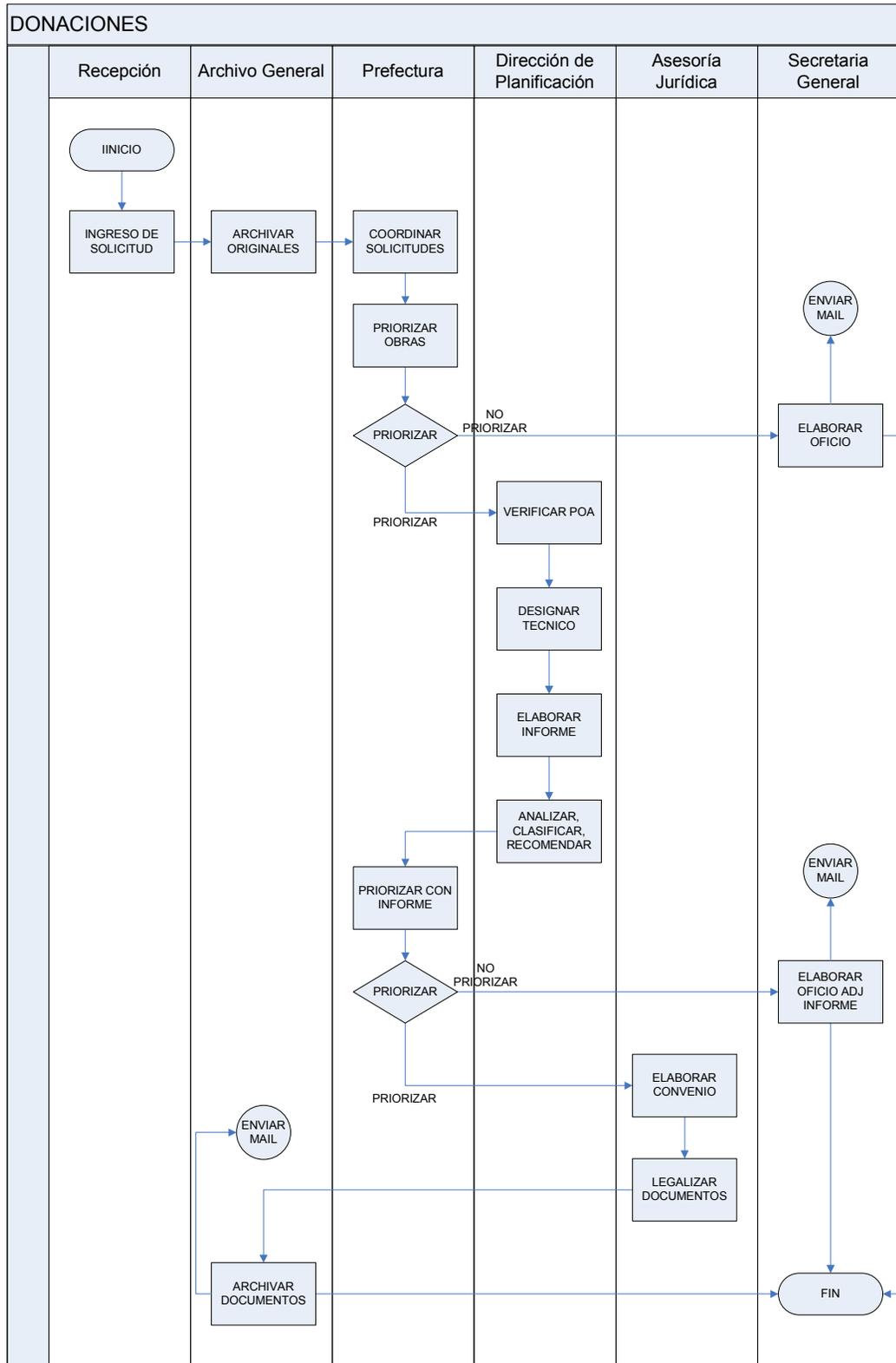


Ilustración 3-3. Diagrama de Proceso de Donaciones

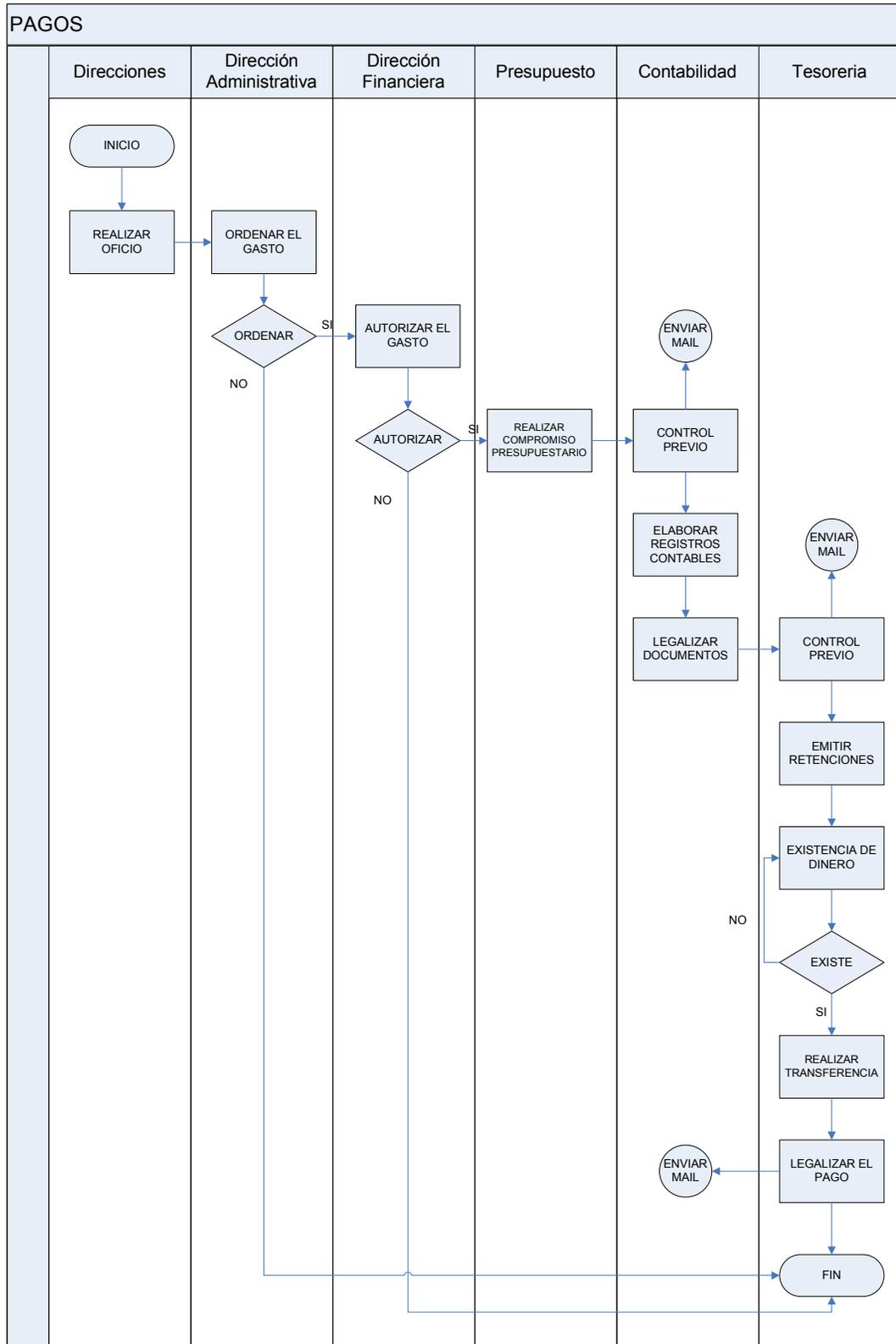


Ilustración 3-4. Diagrama de Proceso de Pagos

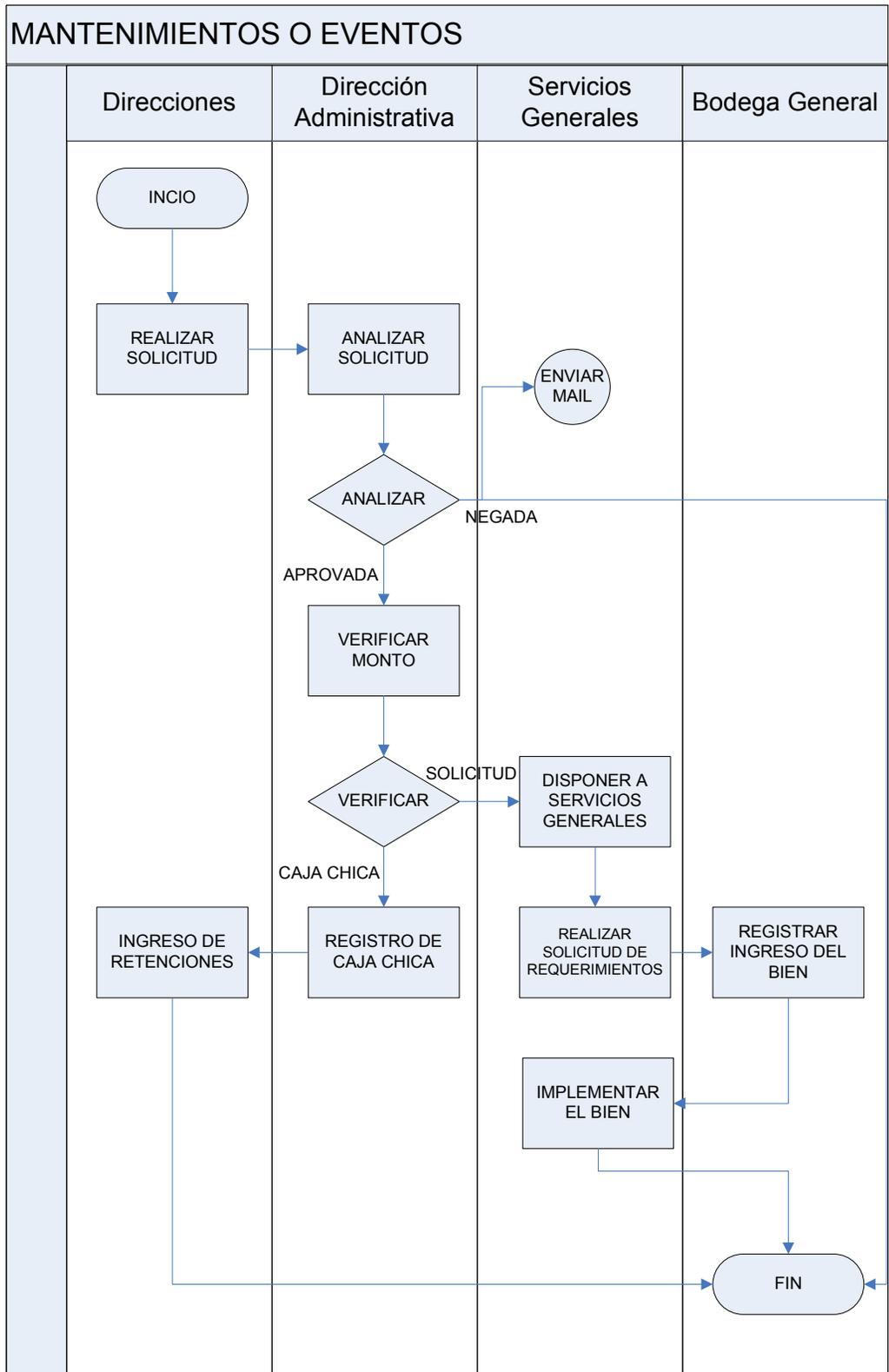


Ilustración 3-5. Diagrama de Proceso de Mantenimientos o Eventos

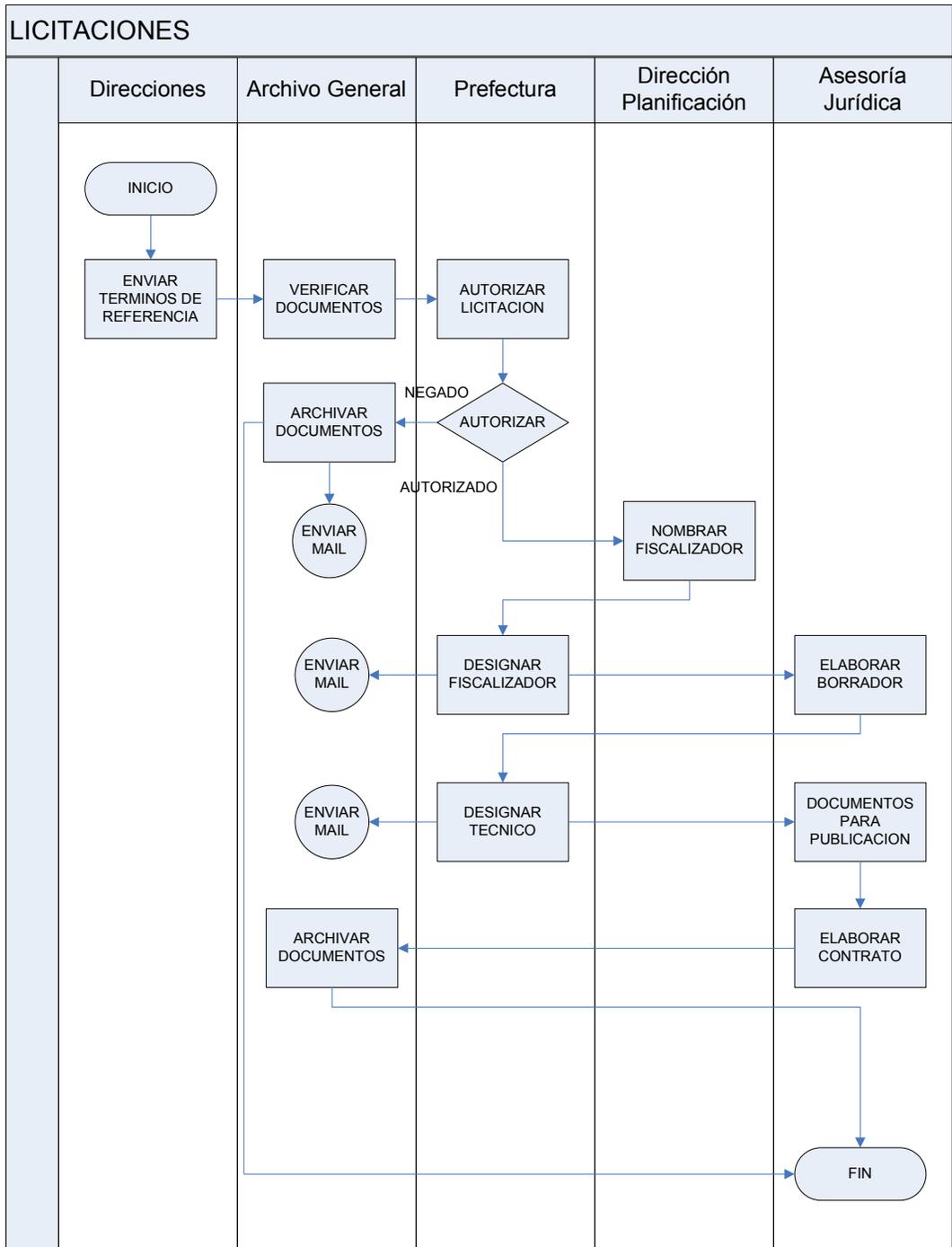


Ilustración 3-6. Diagrama de Proceso de Licitaciones

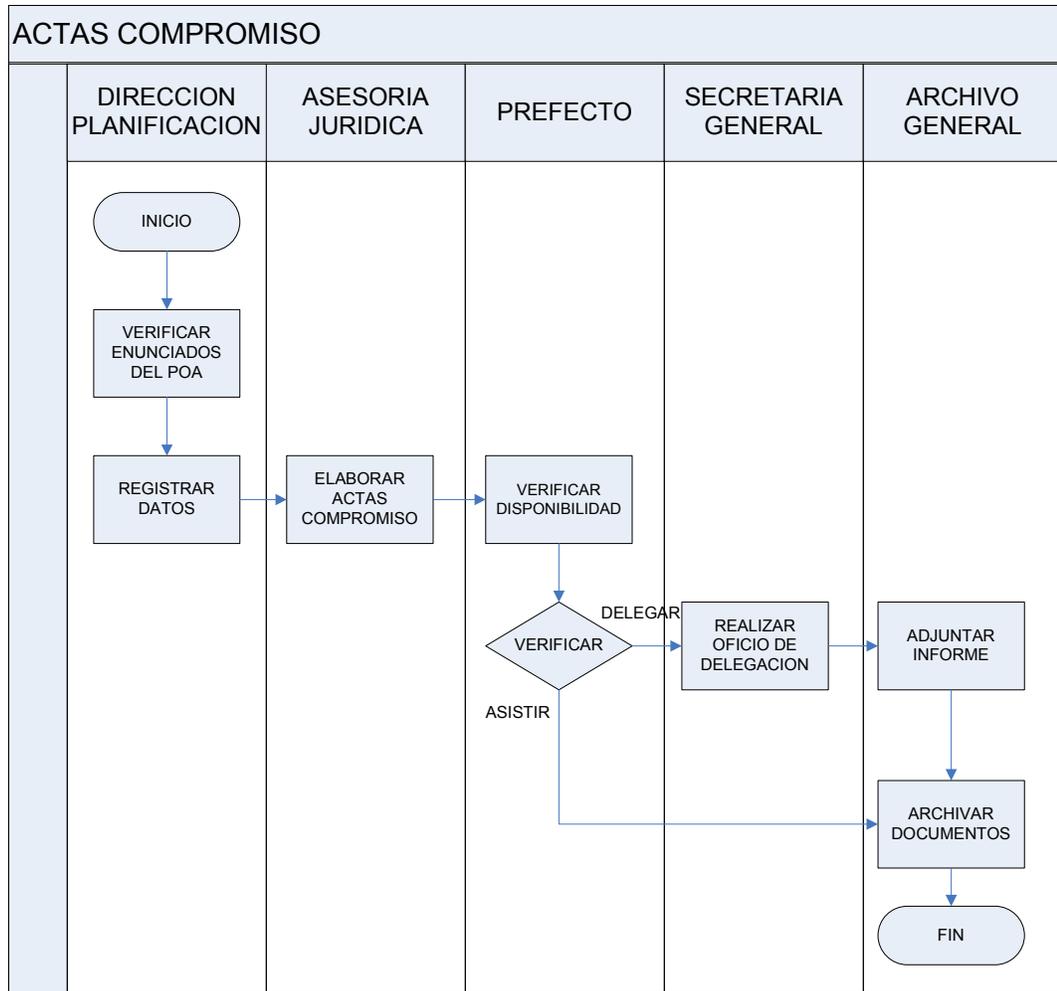


Ilustración 3-7. Diagrama de Proceso de Actas Compromiso

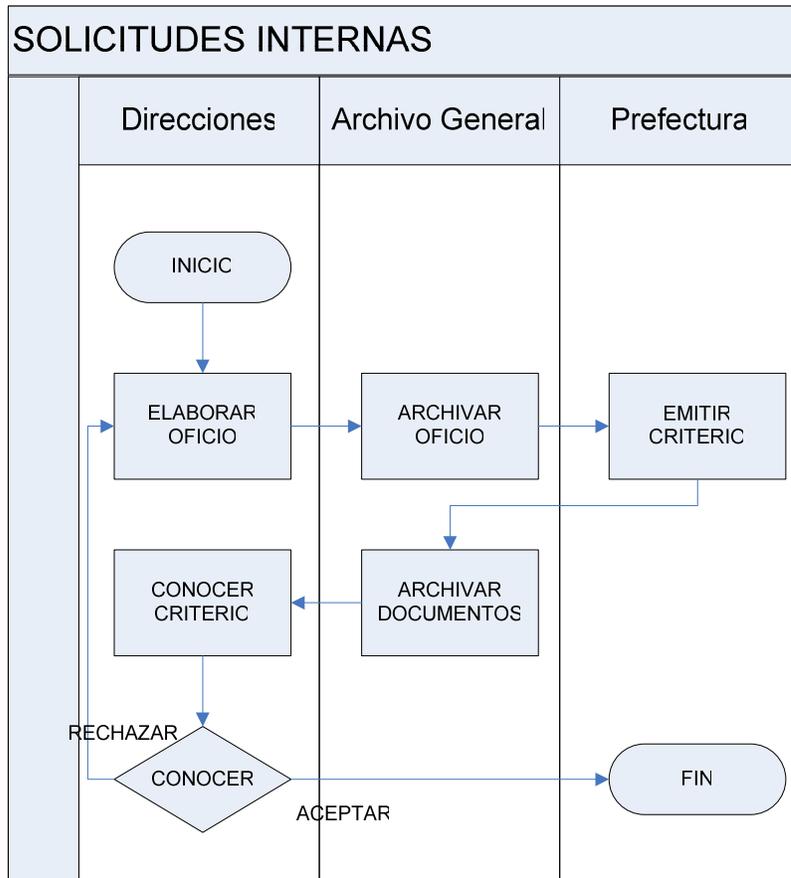


Ilustración 3-8. Diagrama de Proceso de Solicitudes Internas

3.7.1 Modelos Objeto-Relación

Los procesos han sido levantados junto con las especificaciones del software, es hora de iniciar con la construcción del mismo, para ello tomaremos todos los datos que hemos obtenido, se identificaron clases y las relaciones entre ellas, objetos, atributos y operaciones.

Ver Anexo 5: Modelo Objeto Relación. Pág. 107

3.7.2 Modelos Objeto-Comportamiento

El modelo Objeto-Relación es una representación estática del sistema, ahora veremos la representación del comportamiento del sistema. Para ello procedimos a evaluar todos los casos de uso e identificar los siguientes eventos:

Escenario Normal del WorkFlow

- El WorkFlow pide que ingrese una identificación al usuario.
- El Usuario ingresa la identificación “mastudillo” y la contraseña “123456”
- El WorkFlow verifica la identificación con la información del Servidor y notifica la autenticidad al WorkFlow
- El Usuario elige la nueva tarea que desea realizar (Solicitud de Servicios, Invitaciones, etc.) y el nombre que se le va a dar.
- El WorkFlow genera el formulario necesario
- El Workflow solicita se ingresen los datos necesarios, el usuario ingresa la descripción y datos varios
- El Workflow verifica los datos ingresados según su control dependiendo del contenido y confirma el éxito de la misma
- El WorkFlow pide que ingrese los documentos de respaldo
- El Usuario adjunta el número de documentos necesarios
- El WorkFlow confirma el éxito de la misma
- El WorkFlow pregunta la decisión al usuario, este toma su decisión
- El WorkFlow envía la tarea a un usuario para que ejecute la siguiente actividad que competa
- El WorkFlow muestra la lista de tareas pendientes al usuario actual
- El WorkFlow pide que ejecute una tarea pendiente al usuario
- El Usuario elige la tarea y comienza la ejecución
- El WorkFlow muestra la tarea y pide al usuario adjuntar los documentos pertinentes al usuario
- El Usuario adjunta los documentos necesarios
- El WorkFlow pregunta la decisión al usuario, este toma una decisión
- El Workflow genera un evento en relación a la actividad actual y envía la tarea a un usuario para que ejecute la siguiente actividad que competa
- El WorkFlow muestra la lista de tareas pendientes al usuario actual
- El WorkFlow pide que ejecute una tarea pendiente al usuario
- El Usuario decide cerrar la sesión, El Workflow pregunta si esta Seguro
- El Usuario responde que si y El WorkFlow cierra la sesión
- El WorkFlow pide que ingrese una identificación al usuario.

- El Usuario ingresa la identificación “mastudillo” y la contraseña “123456”
- El WorkFlow verifica la identificación con la información del Servidor y notifica la autenticidad al WorkFlow
- El WorkFlow muestra la lista de tareas pendientes al usuario actual
- El WorkFlow pide que ejecute una tarea pendiente al usuario
- El Usuario elige la tarea y comienza la ejecución
- El WorkFlow muestra la tarea y pide al usuario adjuntar los documentos pertinentes al usuario
- El Usuario adjunta los documentos necesarios
- El WorkFlow pregunta la decisión al usuario, este toma una decisión
- El Workflow genera un evento en relación a la actividad actual y envía la tarea a un usuario para que ejecute la siguiente actividad que competa
- El WorkFlow muestra la lista de tareas pendientes al usuario actual
- El WorkFlow pide que ejecute una tarea pendiente al usuario
- El Usuario decide cerrar la sesión, El Workflow pregunta si esta Seguro
- El Usuario responde que si y El WorkFlow cierra la sesión

Escenario con excepciones del WorkFlow

- El WorkFlow pide que ingrese una identificación al usuario.
 - El Usuario ingresa la identificación “mastudillo” y la contraseña “123456”
 - El WorkFlow verifica la identificación con la información del Servidor y notifica la no autenticidad al WorkFlow
 - El Workflow indica que la contraseña es incorrecta y pide nuevamente la contraseña
 - El Usuario ingresa la identificación “mastudillo” y la contraseña “123456”
 - El WorkFlow verifica la identificación con la información del Servidor y notifica la autenticidad al WorkFlow
 - El Usuario decide cerrar la sesión, El Workflow pregunta si esta Seguro
 - El Usuario responde que si y El WorkFlow cierra la sesión
-
- El WorkFlow pide que ingrese una identificación al usuario.
 - El Usuario ingresa la identificación “mastudillo” y la contraseña “123456”
 - El WorkFlow verifica la identificación con la información del Servidor y notifica la autenticidad al WorkFlow
 - El Usuario elige la nueva tarea que desea realizar (Solicitud de Servicios, Invitaciones, etc.) y el nombre que se le va a dar.
 - El WorkFlow genera el formulario necesario
 - El Workflow solicita se ingresen los datos necesarios, el usuario decide regresar a la pantalla lista de tareas
 - El WorkFlow muestra la lista de tareas pendientes al usuario actual
 - El Usuario decide cerrar la sesión, El Workflow pregunta si esta Seguro
 - El Usuario responde que si y El WorkFlow cierra la sesión

Ver Anexo 6: Modelo Objeto Comportamiento. Pág. 108

3.8 Asociación de Roles

Invitaciones

Las actividades y reglas tienen que ser realizadas por los roles que especificamos anteriormente.

- | | |
|---------------------------------|-------------------------------|
| • Ingreso de Invitaciones | Recepcionista |
| • Verificar Disponibilidad | Coordinador de Administrativo |
| • Confirmar Asistencia | Prefecto |
| • Realizar Oficio de Delegación | Secretario General |
| • Adjuntar Informe | Funcionario |
| • Elaborar Excusa | Secretario General |

Solicitud de Obras

Ahora asociemos las actividades y reglas con sus roles:

- | | |
|-------------------------------------|----------------------------------|
| • Ingreso de Solicitud | Recepcionista |
| • Archivar Originales | Jefe de Archivo |
| • Coordinar Solicitudes | Coordinador Administrativo |
| • Verificar POA | Director de Planificación |
| • Designar Técnico | Director de Infraestructura Vial |
| • Adjuntar Informe | Infraestructura Vial |
| • Analizar, Clasificar y Recomendar | Director de Planificación |
| • Realizar Términos de Referencia | Director Asesoría Jurídica |
| • Legalizar Documentos | Asesor Jurídico |
| • Archivar Documentos | Jefe de Archivo |
| • Elaborar Respuesta | Secretario General |
| • Monto de la Obra | Coordinador Administrativo |
| • Priorizar Obra | Prefecto |
| • Priorizar con Informe | Prefecto |

Donaciones

Las actividades y reglas son parecidas a la Solicitud de Compras por lo que sus roles lo también los son:

- | | |
|-------------------------------------|----------------------------|
| • Ingreso de Solicitud | Recepcionista |
| • Archivar Originales | Jefe de Archivo |
| • Coordinar Solicitudes | Coordinador Administrativo |
| • Verificar POA | Director de Planificación |
| • Designar Técnico | Director de Planificación |
| • Adjuntar Informe | Jefe de Archivo |
| • Analizar, Clasificar y Recomendar | Director de Planificación |
| • Realizar Términos de Referencia | Director Asesoría Jurídica |
| • Legalizar Documentos | Asesor Jurídico |
| • Archivar Documentos | Jefe de Archivo |
| • Elaborar Respuesta | Secretario General |
| • Monto de la Obra | Coordinador Administrativo |
| • Priorizar Obra | Prefecto |
| • Priorizar con Informe | Prefecto |

Pagos

Para los pagos existen las siguientes asociaciones:

- | | |
|--------------------------------------|-------------------------|
| • Realizar Oficio | Funcionarios |
| • Ordenar el Gasto | Director Administrativo |
| • Autorizar el Gasto | Director Financiero |
| • Realizar Compromiso Presupuestario | Jefe de Presupuesto |
| • Control Previo Contabilidad | Contadores |
| • Elaborar Registros Contables | Contadores |
| • Legalizar Documentos | Contador General |
| • Control Previo Tesorería | Tesorero |
| • Emitir Retenciones | Tesorero |
| • Existencia de Dinero | Tesorero |
| • Realizando Transferencia | Tesorero |
| • Legalizar el Pago | Tesorero |

Mantenimientos o Eventos

Con esta información asignemos roles a las actividades y reglas del negocio:

- | | |
|---|-----------------------------|
| • Realizar Solicitud | Funcionarios |
| • Registro de Caja Chica. | Encargado Caja Chica |
| • Ingreso de Retención. | Encargado Caja Chica |
| • Disponer a Servicios Generales | Director Administrativo |
| • Realizar Solicitud de Requerimientos. | Jefe de Servicios Generales |
| • Registrar Ingreso del Bien | Jefe de Bodega |
| • Implementar el Bien o Servicio | Jefe de Servicios Generales |
| • Verificar Monto | Director Administrativo |
| • Analizar Solicitud | Director Administrativo |

Licitaciones

El rol respectivo para las actividades y reglas son:

- | | |
|---------------------------------|-------------------------------|
| • Enviar Términos de Referencia | Directores |
| • Verificar Documentos | Jefe de Archivo General |
| • Nombrar Fiscalizador | Directora de Planificación |
| • Designar Fiscalizador | Prefecto |
| • Elaborar Borrador | Director de Asesoría Jurídica |
| • Designar Técnico (Comisión) | Prefecto |
| • Documentos para Publicación | Director de Asesoría Jurídica |
| • Elaborar Contrato | Director de Asesoría Jurídica |
| • Archivar Documentos | Jefe de Archivo General |
| • Autorizar Licitación | Prefecto |

Actas Compromiso

Los roles sus actividades y reglas de negocio son:

- | | |
|---------------------------------|-------------------------------|
| • Enunciados del POA | Director de Planificación |
| • Registrar Datos | Técnicos de Planificación |
| • Elaborar Acta Compromiso | Director de Asesoría Jurídica |
| • Verificar Disponibilidad | Coordinador de Administrativo |
| • Asistir | Prefecto |
| • Realizar Oficio de Delegación | Secretario General |
| • Adjuntar Informe | Funcionario |
| • Archivar Documentos | Jefe de Archivo |

Solicitudes Internas

La asociación de roles es la siguiente:

- | | |
|----------------------|-----------------|
| • Ingresar Oficio | Funcionarios |
| • Archivar Oficio | Jefe de Archivo |
| • Emitir Criterio | Prefecto |
| • Archivar Respuesta | Jefe de Archivo |
| • Conocer Criterio | Funcionarios |

3.9 Asociación de Atributos con las Actividades

Los atributos sobre cada actividad serán dictados por las restricciones que puedan fijar la ejecución de las mismas tal es el caso del tiempo necesario para realizarlas. Otro dato importante son las prioridades que tienen las actividades ya que estas pueden ser ejecutadas dependiendo del criterio del personal encargado de armar los flujos de trabajo. De la misma manera que damos importancia a este atributo de la actividad lo dejamos sin restricción ya que no supieron definir por la forma interna de trabajo.

3.10 Conclusiones

Cada proceso pudo ser definido mediante sus diagramas y de un diagrama general que reúne a los mismos, se pudieron identificar claramente sus actividades y las reglas y políticas que rigen a los mismos. Se reasignaron roles a actividades realizadas por personas que no tenían relación con esta, resultando en un refinamiento de la especificación de requisitos.

CAPITULO IV
DISEÑO

4 Diseño

4.1 *Introducción*

Las técnicas utilizadas junto al conjunto de procedimientos forman parte de este capítulo, se realizó la división en subsistemas, diseño de objetos con iteraciones en íntima relación con la etapa de análisis, y finalmente el diseño de mensajes para establecer la comunicación.

4.2 *Diseño del Sistema*

Para el diseño del sistema se utilizarán técnicas orientadas a objetos con el fin de lograr un nivel de modularidad. En la etapa de Análisis se estableció los modelos de Objeto – Relación y Objeto – Comportamiento, estos son la base para poder identificar los subsistemas y llegar a un alto nivel de refinamiento. Debido a que en esta etapa existe varias iteraciones se explicará lo íntimamente necesario.

4.3 *Etapas Genéricas*

Similar a los métodos de AOO existen varios métodos (nombres del autor), dentro de los cuales tenemos los más importantes: Booch, Coad-Yourdon, Rumbaugh, Wirfs-Brock, estos difieren en la terminología y sus pasos, pero los procesos generales son muy similares

4.3.1 **Describir subsistemas**

El Subsistema de Control de Usuarios: este se compone de las clases Usuarios, Roles las mismas que contienen las relaciones:

- Los Usuarios cumplen Roles de un Departamento
- Un Departamento definen Roles y estos participan en Actividades

Las clases Roles y Usuarios generan los comportamientos de identificación y de ejecución correspondiente. La clase Usuarios se comunica con la clase Roles Cliente Servidor, es decir en una sola dirección, y ambas generan clases colaborativas con el resto de subsistemas.

La interfaz Hombre-Maquina queda establecida de la siguiente manera:

- Usuarios
 - Listado
 - Ingreso(Asignación de Roles), Modificación y Eliminación
- Roles
 - Listado
 - Ingreso, Modificación y Eliminación.
- Orgánico
 - Listado
 - Ingreso, Modificación y Eliminación.

Tabla de Colaboraciones del Subsistema

Tipo: Punto a Punto

Colaboradores: Subsistema Tareas, Subsistema Flujos.

Clase: Usuarios, Roles, Organico.

Operación: Los nombres de las Operaciones son Ejecuta y Participa

Formato del Mensaje: Comenzar con un Verbo que explique la operación y luego lo que afecta.

El Subsistema de Tareas: está compuesto de las clases Tareas, Expedientes, Histórico y Documentos. Las relaciones existentes son:

- Las Tareas generan Expedientes
- Los Expedientes generan Históricos
- Un Documentos pertenece a un Expediente

Estas clases generan los comportamientos de elegir tareas, procesar datos, adjuntar documentos, introducir datos. Los enlaces que existen en este subsistema se definen de la siguiente forma, la clase Tareas es Cliente Servidor con la clase Expedientes, la clase Expedientes es Cliente Servidor con la clase Documentos, la clase expedientes es Cliente Servidor con la clase Históricos.

La interfaz Hombre Maquina se establece como:

- Documentos
 - Listado
 - Adjuntar, Eliminar
- Tareas
 - Ingresar
 - Cancelar
- Expedientes
 - Ingresar, Modificar
 - Cancelar
- Histórico
 - Listar

Tabla de Colaboración del Subsistema

Tipo: Punto a Punto

Colaboradores: Subsistema Flujos.

Clase: Tareas, Expedientes, Documentos, Históricos.

Operación: Los nombres de las Operaciones son Realizan y Agregados

Formato del Mensaje: Comenzar con un Verbo que explique la operación y luego lo que afecta.

El Subsistema de Flujos: esta compuesto por las clases que conforman la estructura del WorkFlow: Actividades, Procesos, Páginas, Atributos, Decisiones, Rutas, RegyPol (Reglas y Politicas), Rutas y Eventos. Las relaciones existentes en este modulo son:

- Los Procesos esta formados por Actividades
- Las Actividades se interfasan en Paginas.
- Las Actividades poseen Atributos.

- Las Actividades posibilitan Decisiones.
- Las Actividades tienen Rutas
- Las Actividades necesitan Reglas y Políticas
- Las Actividades accionan Eventos.

Las clases generan los comportamientos de introducir decisiones, ejecutar eventos. Todas las clases en este subsistema son Cliente Servidor excepto la de Rutas. La Interfaz Hombre Maquina queda definida de la siguiente forma:

- Procesos
 - Listado.
 - Ingreso, Modificación, Eliminación.
- Actividades
 - Listado.
 - Ingreso, Modificación, Eliminación.
 - Asociación con Proceso.
 - Asociación con Roles.
 - Asociación con Decisiones
- Páginas
 - Listado.
 - Ingreso, Modificación, Eliminación.
 - Propiedades de Pagina.
 - Código Generado.
- Atributos
 - Listado.
 - Ingreso, Modificación, Eliminación.
 - Asociación con Páginas.
- Decisiones
 - Listado.
 - Ingreso, Modificación, Eliminación.
 - Asociación con Actividades.
- Rutas
 - Generación de Rutas.
 - Eliminación de Rutas.
- Reglas y Políticas
 - Listado.
 - Ingreso, Modificación, Eliminación.
 - Asociación con Actividades
- Eventos
 - Listado.
 - Ingreso, Modificación, Eliminación.
 - Asociación con Actividades.

Tabla de Colaboración del Subsistema

Tipo: Punto a Punto

Colaboradores: Subsistema Usuarios, Subsistema Tareas.

Clase: Actividades, Procesos, Páginas, Atributos, Decisiones, Rutas, Reg y Pol (Reglas y Políticas), Rutas y Eventos

Operación: Los nombres de las Operaciones son Participan y Realizan

Formato del Mensaje: Comenzar con un Verbo que explique la operación y luego lo que afecta.

4.4 Diseñar Objetos

El diseño de objetos es una parte fundamental y fue realizada a la par con el análisis para trabajar con iteraciones, vamos a definir las clases internas, atributos y sus operaciones con un detalle más a fondo.

Descripción de Clases

Procesos. Necesarios para recuperar información acerca de los procesos que se utilizaran en el workflow.

Se podrá ingresar datos del nombre del conjunto para las actividades, se podrá modificar y eliminar.

Se envía un mensaje solicitando un código de actividad y se recibe una identificación, esta identificación se la relaciona con la identificación de la proceso.

Paginas. Necesarios para mostrar la interfaz de cada una de las actividades.

Se ingresará un nombre de la página y se colocará el código necesario para dicha interfaz.

Se envía un mensaje solicitando un código de actividad y se recibe una identificación, esta identificación se la relaciona con la identificación de la página.

Atributos. Información de las actividades que pueden contener más de un datos para la misma relación actividad atributo.

Se envía un mensaje solicitando un código de actividad y se recibe una identificación, esta identificación nos sirve para relacionar un atributo con una actividad

Decisiones. Información que ayudará a tomar una decisión cuando las actividades sean de este tipo.

Se ingresa una descripción de la decisión, esta se puede modificar o eliminar.

Se envía un mensaje solicitando un código de actividad y se recibe la identificación, esta se ingresa con la decisión que se registra.

Rutas. Información sobre los caminos que podrá tomar un proceso dependiendo de la decisión del usuario.

Las actividades generadas deben entrelazarse, se envía un mensaje pidiendo un código para la actividad anterior, se lee ese código; se envía un mensaje pidiendo un código para la actividad actual, se lee ese código; se envía un mensaje pidiendo un código para la actividad siguiente, se lee ese código; se registra la ruta junto al valor que el usuario deberá elegir.

Registro y Políticas. Información teórica sobre los datos que norman el workflow.

Las actividades de tipo decisión tendrán una relación con un registro y política, se envía un mensaje solicitando un código de identificación, se lee ese código y se registra junto con los datos descriptivos de la clase.

Eventos. Situaciones que deberán accionarse en ciertos casos del workflow.

Se ingresa la descripción del evento y junto a ello el código necesario para disparar el evento. Se envía un mensaje solicitando un código de actividad, se lee el código y se registra el evento.

Actividades. Información sobre cada paso que debe ejecutarse en un proceso dentro del workflow.

Se ingresa los datos relacionados con la actividad, estos se pueden modificar o eliminar. Se envía un mensaje solicitando un código de proceso, se lee el código y se registra la actividad. Se envía un mensaje solicitando un código de rol, se lee el código y se registra el responsable.

Roles. Información sobre los roles que tiene la empresa, cada cargo del orgánico estructural de la misma.

Se ingresa la descripción y nombre del rol, se envía un mensaje solicitando el código del departamento al cual pertenece el rol, se registra el rol.

Usuarios. Información sobre los empleados operativos de la institución.

Se ingresa los datos de los empleados, se pueden modificar y eliminar.

Tareas. Información sobre el trabajo diario que debe realizar la institución.

Se solicita un código de proceso, se lee el código y se registra la tarea.

Expedientes. Información sobre los datos que generan cada proceso.

Se envía un mensaje solicitando un código de expediente, se lee el código; se envía un mensaje solicitando código de página, se lee el código y se aplica la interfaz.

Documentos. Información relacionada a los documentos digitalizados generados en cada actividad dentro de un proceso.

Se envía un mensaje solicitando código de tarea, se lee el código; se envía un mensaje solicitando el código del usuario, se lee el código del usuario; se registra el documento y se registra la relación.

Históricos. Información generada sobre los expedientes y documentos ingresados, modificados y eliminados.

Se envía información solicitando el código de expediente o documentos, se lee el código y se registra junto con la acción realizada.

Estructura de Datos

La estructura de datos serán normada colocando el nombre de cada atributo de la tabla con las tres primeras letras del nombre de la tabla y en el caso de que el nombre de la tabla sea compuesto por dos palabras, los atributos se formaran con la primera letra de la primera palabra y las dos primeras letras de la segunda palabra.

Proceso.

```
[PRO_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,  
[PRO_NOMBRE] [varchar] (50) NULL ,  
[PRO_DESCRIPCION] [text] NULL ,
```

[PRO_PARSER] [int] NULL ,
[PRO_UNIDAD] [int] NULL ,
[PRO_PREVISTA] [int] NULL ,
[PRO_MINIMA] [int] NULL ,
[PRO_MAXIMA] [int] NULL ,
[PRO_VALOR] [varchar] (50) NULL ,
[EXP_TIPO] [int] NULL ,
[TIP_ESTADO] [int] NOT NULL.

Actividades

[ACT_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[PRO_CODIGO] [int] NULL ,
[ACT_NOMBRE] [varchar] (50) NULL ,
[ACT_DESCRIPCION] [text] NULL ,
[ACT_PLAZO] [int] NULL ,
[PAG_CODIGO] [int] NULL ,
[ACT_TIPO] [int] NULL ,
[ACT_LOGICO] [varchar] (50) NULL ,
[ACT_VALOR] [varchar] (50) NULL ,
[ACT_PARTICULAR] [int] NULL ,
[ACT_ELECCION] [int] NULL ,
[ACT_ASIGNACION] [int] NULL ,
[TIP_ESTADO] [int] NULL ,

Páginas

[PAG_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[PAG_DESCRIPCION] [text] NULL ,
[PAG_ENLACE] [varchar] (50) NULL ,
[PAG_INTERFAZ] [text][PAG_ESTADO] [int] NULL

Atributos

[ATR_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[ATR_NOMBRE] [varchar] (50) NULL ,
[ATR_DESCRIPCION] [varchar] (50) NULL ,
[ATR_VALOR] [varchar] (50) NULL.

Decisiones

[DES_CODIGO] [int] NOT NULL ,
[DES_DESCRIPCION] [varchar] (50) NULL ,
[DES_VALOR] [varchar] (50) NULL ,
[DES_ESTADO] [int] NULL.

Rutas

[RUT_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[RUT_ACT_PADRE] [int] NULL ,
[RUT_ACT_ACTUAL] [int] NULL ,
[RUT_ACT_VALOR] [varchar] (50) NULL ,
[RUT_ACT_HIJO] [int] NULL ,
[RUT_ACT_HIJO_POS] [int] NULL ,

[RUT_GRUPO] [int] NULL ,
[RUT_ORDEN] [int] NULL.

Reg_Pol

[RPO_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[RPO_NOMBRE] [varchar] (50) NULL ,
[RPO_DESCRIPCION] [text] NULL ,
[RPO_TIPO] [int] NULL ,
[RPO_VALOR] [varchar] (50) NULL ,
[TIP_ESTADO] [int] NOT NULL.

Eventos

[EVE_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[EVE_NOMBRE] [varchar] (50) NULL ,
[EVE_DESCRIPCION] [varchar] (50) NULL ,
[EVE_ACCION] [text] NULL.

Posiciones

[POS_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[PRO_CODIGO] [int] NULL ,
[ACT_CODIGO] [int] NULL ,
[POS_X1] [float] NULL ,
[POS_Y1] [float] NULL ,
[POS_X2] [float] NULL ,
[POS_Y2] [float] NULL ,
[POS_X3] [float] NULL ,
[POS_Y3] [float] NULL ,
[POS_X4] [float] NULL ,
[POS_Y4] [float] NULL ,

Roles

[ROL_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[ROL_NOMBRE] [varchar] (50) NULL ,
[ROL_DESCRIPCION] [varchar] (50) NULL.

Usuarios

[USU_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[USU_APELLIDO] [nvarchar] (50) NULL ,
[USU_SEGUNDOAPELLIDO] [nvarchar] (50) NULL ,
[USU_NOMBRE] [nvarchar] (50) NULL ,
[USU_SEGUNDONOMBRE] [nvarchar] (50) NULL ,
[USU_CONTRASENIA] [nvarchar] (50) NULL ,
[USU_FECHAN] [datetime] NULL ,
[TIP_CONTRATACION] [int] NULL , (Atributo de sistema Intranet)
[TIP_USUARIO] [int] NULL , (Atributo de sistema Intranet)
[TIP_PERFIL] [int] NULL , (Atributo de sistema Intranet)
[USU_TRATAMIENTO] [varchar] (50) NULL ,
[TIP_ESTADO] [int] NULL ,

[USU_ID_SESION] [nvarchar] (25) NULL ,
[USU_MAIL] [nvarchar] (200) NULL ,
[CAMBIARCLAVE] [int] NULL.

Tareas

[TAR_CODIGO] [int] NOT NULL ,
[TAR_FECHA] [datetime] NULL ,
[TAR_ANIO] [int] NULL ,
[TAR_NOMBRE] [varchar] (50) NOT NULL ,
[USU_ID_SESION] [varchar] (50) NULL ,
[PRO_CODIGO] [int] NULL ,
[TAR_PRIORIDAD] [int] NULL ,
[EXP_NUMERO] [int] NULL ,
[EXP_ANIO] [int] NULL ,
[TIP_ESTADO] [int] NOT NULL.

Expedientes

[EXP_TIPO] [int] NOT NULL ,
[EXP_NUMERO] [int] NOT NULL ,
[EXP_ANIO] [int] NOT NULL ,
[ARC_NUMERO] [int] NULL ,
[USU_CODIGO] [varchar] (50) NULL ,
[TIP_UBICACIONACTUAL] [varchar] (255) NULL ,
[EXP_FUN_INS] [varchar] (255) NULL ,
[EXP_TELEFONO] [varchar] (50) NULL ,
[EXP_RESUMEN] [text] NULL ,
[EXP_FIRMA] [varchar] (200) NULL ,
[EXP_CEDULARUC] [varchar] (50) NULL ,
[EXP_NOTIFICACIONES] [text] NULL ,
[EXP_CREADO] [datetime] NULL ,
[TIP_PROCESO] [int] NULL ,
[TIP_ACTIVIDAD] [int] NULL ,
[TIP_CLASE] [int] NULL ,
[TIP_PRIORIDAD] [int] NULL ,
[TIP_ESTADO] [int] NULL ,
[EXP_LUGAR] [varchar] (50) NULL ,
[EXP_FECHA] [datetime] NULL ,
[EXP_HORA] [varchar] (5) NULL ,
[PRV_CODIGO] [int] NULL ,
[CAN_CODIGO] [int] NULL ,
[PAR_CODIGO] [int] NULL.

Documentos

[DOC_CODIGO] [int] NOT NULL ,
[EXP_TIPO] [int] NULL ,
[EXP_NUMERO] [int] NULL ,
[EXP_ANIO] [int] NULL ,
[DOC_REGISTRO] [datetime] NULL ,
[DOC_HOST] [char] (100) NULL ,
[DOC_NOMBRE] [char] (255) NULL ,

[DOC_TAMANIO] [char] (10) NULL ,
[DOC_NOTA] [char] (255) NULL ,
[USU_CODIGO] [char] (40) NULL ,
[DOC_ESTADO] [int] NOT NULL.

Histórico

[HIS_NUMERO] [decimal](18, 0) NOT NULL ,
[HIS_FECHA] [datetime] NULL ,
[HIS_ACCION] [varchar] (200) NULL ,
[EXP_TIPO] [int] NULL ,
[EXP_NUMERO] [int] NULL ,
[EXP_ANIO] [int] NULL ,
[USU_CODIGO] [varchar] (40) NULL ,
[EXP_CREADO] [datetime] NULL ,
[TIP_PROCESO] [varchar] (25) NULL ,
[TIP_ACTIVIDAD] [varchar] (50) NULL ,
[TIP_UBICACIONACTUAL] [varchar] (100) NULL ,
[EXP_FUN_INS] [varchar] (100) NULL ,
[EXP_TELEFONO] [varchar] (50) NULL ,
[EXP_RESUMEN] [varchar] (500) NULL ,
[EXP_FIRMA] [varchar] (100) NULL ,
[EXP_CEDULARUC] [varchar] (200) NULL ,
[EXP_NOTIFICACIONES] [varchar] (500) NULL ,
[TIP_CLASE] [varchar] (100) NULL ,
[TIP_PRIORIDAD] [varchar] (15) NULL ,
[TIP_ESTADO] [int] NOT NULL ,
[HOST] [varchar] (100) NULL .

Adjuntos

[ADJ_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[TAR_CODIGO] [int] NOT NULL ,
[ACT_CODIGO] [int] NOT NULL ,
[DOC_CODIGO] [int] NOT NULL ,

Ejecutadas

[EJE_CODIGO] [int] NOT NULL ,
[TAR_CODIGO] [int] NOT NULL ,
[ACT_CODIGO] [int] NOT NULL ,
[ORG_CODIGO] [int] NOT NULL ,
[TIP_CARGO] [int] NULL ,
[USU_CODIGO] [varchar] (50) NULL ,
[RUT_ACT_VALOR] [varchar] (50) NULL ,
[EJE_FECHA] [datetime] NULL.

Organico.

```
[ORG_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[NIV_CLAVE] [int] NULL ,
[DIR_CLAVE] [int] NULL ,
[SEC_CLAVE] [int] NULL ,
[DESCRIPCION] [varchar] (255) NULL ,
[CODIGO] [int] NULL ,
[EST_CLAVE] [int] NULL ,
[USU_CODIGO] [int] NULL ,
[TIPO] [int] NULL ,
[USU_ENCARGADO] [varchar] (50) NULL
```

4.5 Diseñar Mensajes

El diseño de mensajes nos indicará la forma por la cual los objetos se comunicaran con sus colaboradores, para ello utilizaremos el modelo objeto-relación que fue desarrollado en el capítulo anterior.

Comunicación Usuarios-Tareas

Para ello definimos la función autenticar esta función nos permite establecer el alcance del usuario durante su sesión. Al ingresar en cada sección del sistema se utilizará esta función y se controlará a que partes del sistema puede acceder ya que compara con la base de datos si la pagina esta abierta a su uso.

```
function autenticar($USER,$PW,$PAGINA=0,$mensaje='Autenticación Usuarios')
{
    $conn = baseconecta();
    mssql_select_db("workflow", $conn);
    if ($PAGINA==1){
        authenticate("Reingrese la clave");
        exit;
    }
    if ($PAGINA==0)
        $strSelectUSUARIOS="SELECT * FROM USUARIOS WHERE
USU_ID_SESION=$USER' and CONTRASENIA= ". $PW .""";
    else
        $strSelectUSUARIOS="SELECT * FROM USUARIOS_PAGINAS WHERE USU_CLAVE
IN (SELECT USU_CLAVE FROM USUARIOS WHERE USU_ID_SESION=$USER' and
CONTRASENIA= ". $PW .") AND PAG_CLAVE in ( $PAGINA)";
    $OK=0;
    if(!$rs = mssql_query($strSelectUSUARIOS, $conn)){
        $GLOBALS["ERROR__"]="No se puedo Ejecutar el Comando inténtelo mas tarde";
    }

    while ($rsUSUARIOS__=mssql_fetch_array ($rs )) {
        if(!empty($rsUSUARIOS__[0]))
            $OK=1;
    }

    if ($OK==0){
        authenticate("$mensaje");
    }

    return $OK;
}
```

Comunicación Usuarios-Flujos

La función que utilizaremos será EscogeEjecutor(tipo) esta procesa la orden de listar los roles existentes en la institución recibe todas las identificaciones de roles para definir el usuario según el rol que ejecute.

function EscogeEjecutor(tipo)

```
{
    var strFeatures = "width=765, height=400, toolbar=no, menubar=no," + "location=no,
directories=no, scrollbars=yes";
    var enlace="listusu.php?fieldTIPO=";
    enlace=enlace.concat(tipo);
    vlista=window.open(enlace,"Usuarios",strFeatures);
}
```

En el instante que el administrador digita una letra el sistema provee de un listado según las siguientes instrucciones:

```
if($Accion=="Buscar")
{
    if($fieldNOMBRE!="" && $fieldCARGO=="")
    {
        $sql="SELECT [USU_ID_SESION], [APELLIDO]+' '+[SEGUNDOAPELLIDO]+' '+
[NOMBRE]+' '+[SEGUNDONOMBRE] as COMPLETO, [TIP_DEPARTAMENTO],
[TIP_CARGO]
                FROM [LIBERTIS].[dbo].[USUARIOS]
                WHERE [APELLIDO] LIKE '$fieldNOMBRE%' AND
TIP_ESTADO <> 142 ORDER BY COMPLETO";
    }
    if($fieldNOMBRE=="" && $fieldCARGO!="")
    {
        $sql="SELECT [USU_ID_SESION], [APELLIDO]+' '+[SEGUNDOAPELLIDO]+' '+
[NOMBRE]+' '+[SEGUNDONOMBRE] as COMPLETO, [TIP_DEPARTAMENTO],
[TIP_CARGO]
                FROM [LIBERTIS].[dbo].[USUARIOS]
                WHERE TIP_CARGO IN (SELECT [SUB_CLAVE] FROM
[LIBERTIS].[dbo].[SUBTIPO] WHERE SUB_NOMBRE LIKE '$fieldCARGO%') AND
                TIP_ESTADO <>142 ORDER BY
COMPLETO";
    }
    if($fieldNOMBRE!="" && $fieldCARGO!="")
    {
        $sql="SELECT [USU_ID_SESION], [APELLIDO]+' '+[SEGUNDOAPELLIDO]+' '+
[NOMBRE]+' '+[SEGUNDONOMBRE] as COMPLETO, [TIP_DEPARTAMENTO],
[TIP_CARGO]
                FROM [LIBERTIS].[dbo].[USUARIOS]
                WHERE [APELLIDO] LIKE '$fieldNOMBRE%' AND
TIP_CARGO IN (SELECT [SUB_CLAVE] FROM [LIBERTIS].[dbo].[SUBTIPO] WHERE
SUB_NOMBRE LIKE '$fieldCARGO%') AND
                TIP_ESTADO <> 142 ORDER BY COMPLETO";
    }
}
```

Así este sql es armado y ejecutado una vez se ha cargado la página. Por ultimo se utiliza la función PasarDatos para que se regrese el dato necesario para su grabación.

```

function PasarDatos(dep,cargo,des)
{
  <? if($fieldTIPO==0){?>
  window.opener.document.actividades.fieldDESCRIPCION0.value= des;
  window.opener.document.actividades.fieldORG_CLAVE0.value= dep;
  window.opener.document.actividades.fieldTIP_CARGO0.value= cargo;
  <? }?>
  <? if($fieldTIPO==1){?>
  window.opener.document.actividades.fieldDESCRIPCION1.value= des;
  window.opener.document.actividades.fieldORG_CLAVE1.value= dep;
  window.opener.document.actividades.fieldTIP_CARGO1.value= cargo;
  <? }?>
  <? if($fieldTIPO==2){?>
  window.opener.document.actividades.fieldDESCRIPCION2.value= des;
  window.opener.document.actividades.fieldORG_CLAVE2.value= dep;
  window.opener.document.actividades.fieldTIP_CARGO2.value= cargo;
  <? }?>
  <? if($fieldTIPO==3){?>
  window.opener.document.actividades.fieldDESCRIPCION3.value= des;
  window.opener.document.actividades.fieldORG_CLAVE3.value= dep;
  window.opener.document.actividades.fieldTIP_CARGO3.value= cargo;
  <? }?>
  window.close();
}

```

Finalmente estos datos serán guardados en la tabla Actividades.

Comunicación Tareas-Flujos

La comunicación Realizan se hace a través de la función Href propia de html aquí enviaremos los datos de identificación de tareas. Esta función es parte html y php.

```

href="contenido.php?<?
print"fieldTAR_ACT_USU_CODIGO=$fieldTAR_ACT_USU_CODIGO&fieldTAR_CODIGO=
$fieldTAR_CODIGO&fieldACT_CODIGO=$fieldACT_CODIGO&Accion=Siguiente&fieldEXP
_NUMERO=$fieldEXP_NUMERO&fieldPRO_CODIGO=$fieldPRO_CODIGO&Tarea=$Tarea&
fieldCAM_ACT_VALOR=$fieldCAM_ACT_VALOR";?>"

```

Por otra parte la comunicación Agregados se realiza mediante la función anterior una vez comience a cargar la página con las siguientes instrucciones:

```

if($Accion=="Siguiente" && $Tarea==0)
{
  EjecutaSQL("INSERT INTO [SMGA].[dbo].[TAR_ACT_USU]([TAR_CODIGO],
[ACT_CODIGO], [ORG_CLAVE], [TIP_CARGO], [USU_ID_SESION],
[TAR_ACT_USU_FECHA])
  VALUES('$fieldTAR_CODIGO', '$fieldACT_CODIGO', '$sent_TIP_DEPARTAMENTO',
'$sent_TIP_CARGO', '$sent_USU_ID_SESION', getdate())");
  BuscaSiguiente($fieldTAR_CODIGO,$fieldACT_CODIGO,$fieldCAM_ACT_VALOR);
// header("Location : contenido.php");
}
if($Accion=="Siguiente"&& $Tarea <> 0)
{
  if($fieldCAM_ACT_VALOR!="")
  {
    EjecutaSQL("UPDATE [SMGA].[dbo].[TAR_ACT_USU]
SET [TIP_CARGO]='$sent_TIP_CARGO',
[USU_ID_SESION]='$sent_USU_ID_SESION',
[CAM_ACT_VALOR]=$fieldCAM_ACT_VALOR, [TAR_ACT_USU_FECHA]=getdate()

```

```

WHERE TAR_ACT_USU_CODIGO=$fieldTAR_ACT_USU_CODIGO");
}else{
    EjecutaSQL("UPDATE [SMGA].[dbo],[TAR_ACT_USU]
SET [TIP_CARGO]=$ent_TIP_CARGO',
[USU_ID_SESION]=$ent_USU_ID_SESION', [TAR_ACT_USU_FECHA]=getdate()
WHERE TAR_ACT_USU_CODIGO=$fieldTAR_ACT_USU_CODIGO");
}
BuscaSiguiete($fieldTAR_CODIGO,$fieldACT_CODIGO,$fieldCAM_ACT_VALOR);
}

```

4.6 Conclusiones

Las etapas establecidas nos proporcionaron como resultado un conjunto de subsistemas y los comportamientos, se definió las clases internas, sus atributos y operaciones, se genero las estructuras necesarias, con esto diseñamos mensajes mediante los cuales cada objeto se comunicara con sus colaboradores.

CAPITULO V
DESARROLLO

5 Desarrollo

5.1 *Introducción*

La aplicación de técnicas adecuadas para el desarrollo permite construir un sistema fiable, verificaremos mediante métricas, realizaremos configuraciones del servidor HTTP, PHP, construiremos la base de datos en SQL SERVER, y finalmente se detallará el funcionamiento de cada archivo que comprende el sistema.

5.2 *Aplicación de Modelos de Calidad*

Para la calidad en sistemas Web manejaremos la medición de: Actualidad, Disponibilidad, Tasa de información, Autoridad, Facilidad de utilización, Confiabilidad, Corrección (actualización de especificaciones), Robustez (no ruptura para situaciones no esperadas), Escalabilidad, Rendimiento, Seguridad.

Actualidad: Manejamos la última versión de PHP junto con la nueva herramienta AJAX que permite una comunicación interna con el servidor sin que muestre un intercambio de información.

Disponibilidad: el sistema permite mantenerse 24x7 sin ningún problema ya que el software que maneja es probado por más de 1 millón de usuarios que respaldan lo anteriormente citado.

Tasa de información: la tasa de información no representa ya que en hardware el sistema cuenta con un procesador Xeon Intel de 3,4 Ghz de última generación a la fecha junto con un ancho de banda respetable de 1024 Kbps tanto de download como de upload.

Autoridad: El sistema cuenta con un posicionamiento que puede estar a la altura de cualquier sistema desarrollado a igual condiciones.

Facilidad de utilización: el sistema fue concebido en una plataforma que actualmente es la más usada en el mundo como es el ambiente WEB.

Confiabilidad: el sistema responde ante fallos únicamente con una actualización o refrescamiento de una página, además el sistema está diseñado con una expiración de cada página que tenga a su responsabilidad.

Corrección: el sistema trabaja con un servidor de aplicaciones por lo tanto las correcciones que se realicen no necesitan ser cambiadas en cada máquina que utilice el software.

Robustez: el sistema cuenta con un histórico que puede mostrar las situaciones que se ejecutaron antes de un fallo.

Escalabilidad: Sin ninguna mención sabemos que la plataforma web es el siguiente camino para toda aplicación, además la base de datos está abierta a cualquier enlace ya que su codificación consta de id que permiten esta funcionalidad.

Rendimiento: El sistema cuenta con un alto rendimiento en el momento de ejecución de código y acceso a la información.

Seguridad: Se realizó un esfuerzo valioso con el tema de seguridad ya que contamos con una autenticación de usuarios que nos asegura mostrar solo lo necesario, además la información de cada usuario se registra codificada.

Finalmente explicaremos los métodos para evaluar el rendimiento
Finalmente explicar el marco de trabajo para calidad en sistemas Web: Rendimiento (métodos de evaluación de rendimiento), Confiabilidad, Seguridad, Abstracción (calidad de datos que consume y de la información que produce: imágenes, datos, documentos, etc.), Expresividad (que las herramientas utilizadas ayuden a expresar de mejor manera los problemas de un determinado contexto), Automatización (automatización de tareas porque el mercado electrónico es extremadamente dinámico), Temporalidad (manejo de periodo de vigencia de productos y/o servicios ofertados vía Web), Negociación (expresar reglas de negociación entre cliente y proveedor).

5.3 Configuraciones Esenciales

Finalmente entraremos en la etapa de configuraciones esenciales, comenzaremos diciendo que el software que vamos a utilizar es el Servidor html Apache 2 y PHP 5 como su complemento. Estos elementos los podemos descargar de los sitios web www.apache.org y www.php.net respectivamente.

5.3.1 Instalación y Configuración de Apache

Ejecutamos el archivo descargado este muestra una ventana de bienvenida hacemos clic en siguiente, a continuación se acepta el acuerdo de licencia. Por ultimo en “Network Domain” y “Server Name” se escribe la dirección IP de nuestro computador (127.0.0.1) y en Administrator’s Email Address una dirección de email. En cuanto al tipo de instalación se elije la Típica (Typical). A partir de aquí solo se pulsa Next , Install y Finish en este orden sin modificar nada.

Una vez instalado, se procede a configurar el archivo httpd.conf este archivo está ubicado en el path C:\Archivos de programa\Apache Group\Apache2\conf se abre el archivo con un editor de texto y se busca lo siguiente “Dynamic Shared Object (DSO) Support” debajo de esto aparece lo siguiente:

```
# Example:  
# LoadModule foo_module modules/mod_foo.so  
#
```

En la siguiente línea se escribe

```
LoadModule php5_module C:/PHP5/php5apache2.dll
```

Esto sirve para que apache cargue el modulo de php cuando arranque el servicio, sin esto PHP no funcionará por lo que el lenguaje en nuestras páginas no será comprensible.

Luego se procede a cambiar la carpeta en donde se localizará la aplicación, para esto se busca la línea en donde se encuentra “DocumentRoot” a continuación se escribe "C:/PHP/WEB/".

Por ultimo se configura las extensiones que podrán ser el index de la aplicación. Para ello se busca la línea que contenga “DirectoryIndex” y se sustituye por esta línea

```
DirectoryIndex index.html index.htm index.php index.php3 index.phtml  
index.html.var
```

5.3.2 Instalación y Configuración de PHP

Con el archivo descargado en zip se extrae su contenido en el Path "C:/PHP5/". Ahora dentro de la carpeta se encuentra el archivo php5ts.dll el cual copiamos a la carpeta "C:/WINDOWS/System32/". Además de este archivo se copia los siguientes archivos:

- fdfdk.dll
- fribidi.dll
- gds32.dll
- libeay32.dll
- libmhash.dll
- libmysql.dll
- libmysqli.dll
- msql.dll
- ntwdblib.dll
- ssleay32.dll
- yaz.dll

A continuación se copia el archivo php.ini en el path “C:/WINDOWS/” y se procede con su configuración. Se buscaran las siguientes líneas:

register_globals = Off y se cambia la palabra Off por On.

extension_dir = “/” se sustituye “/” por C: /PHP5/ext

Windows Extensions y se elimina el punto y coma de las líneas:

- extension=php_gd2.dll
- extension=php_mime_magic.dll
- extension=php_mssql.dll

5.4 Desarrollo de la BD en SQL Server

Para el desarrollo de la base de datos se generó un script que se encarga de crear las tablas y sus relaciones, este se formula a continuación:

Creación de la Base de Datos Libertis

```
CREATE DATABASE [LIBERTIS] ON (NAME = N'LIBERTIS_Data', FILENAME = N'C:\Archivos de  
programa\Microsoft SQL Server\MSSQL\data\LIBERTIS_Data.MDF', SIZE = 31, FILEGROWTH = 10%) LOG ON  
(NAME = N'LIBERTIS_Log', FILENAME = N'C:\Archivos de programa\Microsoft SQL  
Server\MSSQL\data\LIBERTIS_Log.LDF', SIZE = 19, FILEGROWTH = 10%)  
COLLATE Modern_Spanish_CI_AS  
GO
```

```
exec sp_dboption N'LIBERTIS', N'autoclose', N'true'  
GO
```

```
exec sp_dboption N'LIBERTIS', N'bulkcopy', N'false'
```

```

GO

exec sp_dboption N'LIBERTIS', N'trunc. log', N'true'
GO

exec sp_dboption N'LIBERTIS', N'torn page detection', N'true'
GO

exec sp_dboption N'LIBERTIS', N'read only', N'false'
GO

exec sp_dboption N'LIBERTIS', N'dbo use', N'false'
GO

exec sp_dboption N'LIBERTIS', N'single', N'false'
GO

exec sp_dboption N'LIBERTIS', N'autoshrink', N'true'
GO

exec sp_dboption N'LIBERTIS', N'ANSI null default', N'false'
GO

exec sp_dboption N'LIBERTIS', N'recursive triggers', N'false'
GO

exec sp_dboption N'LIBERTIS', N'ANSI nulls', N'false'
GO

exec sp_dboption N'LIBERTIS', N'concat null yields null', N'false'
GO

exec sp_dboption N'LIBERTIS', N'cursor close on commit', N'false'
GO

exec sp_dboption N'LIBERTIS', N'default to local cursor', N'false'
GO

exec sp_dboption N'LIBERTIS', N'quoted identifier', N'false'
GO

exec sp_dboption N'LIBERTIS', N'ANSI warnings', N'false'
GO

exec sp_dboption N'LIBERTIS', N'auto create statistics', N'true'
GO

exec sp_dboption N'LIBERTIS', N'auto update statistics', N'true'
GO

if( @@microsoftversion / power(2, 24) = 8) and ( @@microsoftversion & 0xffff >= 724 )
exec sp_dboption N'LIBERTIS', N'db chaining', N'false'

```

Creación de las Tablas

```

CREATE TABLE [PROCESOS] (
    [PRO_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [PRO_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [PRO_DESCRIPCION] [text] COLLATE Modern_Spanish_CI_AS NULL ,
    [PRO_PARSER] [int] NULL ,
    [PRO_UNIDAD] [int] NULL ,
    [PRO_PREVISTA] [int] NULL ,
    [PRO_MINIMA] [int] NULL ,
    [PRO_MAXIMA] [int] NULL ,
    [PRO_VALOR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_TIPO] [int] NULL ,
    [TIP_ESTADO] [int] NOT NULL ,
    CONSTRAINT [PK_PROCESOS] PRIMARY KEY CLUSTERED
    (
        [PRO_CODIGO]
    ) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO

CREATE TABLE [PAGINAS] (

```

```

[PAG_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
[PAG_DESCRIPCION] [text] COLLATE Modern_Spanish_CI_AS NULL ,
[PAG_ENLACE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[PAG_INTERFAZ] [text] COLLATE Modern_Spanish_CI_AS NULL ,
[PAG_ESTADO] [int] NULL ,
CONSTRAINT [PK_PAGINAS] PRIMARY KEY CLUSTERED
(
    [PAG_CODIGO]
) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
CREATE TABLE [ACTIVIDADES] (
    [ACT_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [PRO_CODIGO] [int] NULL ,
    [PAG_CODIGO] [int] NULL ,
    [ACT_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [ACT_DESCRIPCION] [text] COLLATE Modern_Spanish_CI_AS NULL ,
    [ACT_PLAZO] [int] NULL ,
    [ACT_TIPO] [int] NULL ,
    [ACT_LOGICO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [ACT_VALOR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [ACT_PARTICULAR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [ACT_ELECCION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [ACT_ASIGNACION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [TIP_ESTADO] [int] NULL ,
CONSTRAINT [PK_ACTIVIDADES] PRIMARY KEY CLUSTERED
(
    [ACT_CODIGO]
) ON [PRIMARY] ,
CONSTRAINT [FK_ACTIVIDADES_PAGINAS] FOREIGN KEY
(
    [PAG_CODIGO]
) REFERENCES [PAGINAS] (
    [PAG_CODIGO]
),
CONSTRAINT [FK_ACTIVIDADES_PROCESOS] FOREIGN KEY
(
    [PRO_CODIGO]
) REFERENCES [PROCESOS] (
    [PRO_CODIGO]
)
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
CREATE TABLE [ATRIBUTOS] (
    [ATR_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [ATR_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [ATR_DESCRIPCION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [ATR_VALOR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
CONSTRAINT [PK_ATRIBUTOS] PRIMARY KEY CLUSTERED
(
    [ATR_CODIGO]
) ON [PRIMARY]
) ON [PRIMARY]
GO
CREATE TABLE [DECISIONES] (
    [DES_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [DES_DESCRIPCION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [DES_VALOR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [DES_ESTADO] [int] NULL ,
CONSTRAINT [PK_DECISIONES] PRIMARY KEY CLUSTERED
(
    [DES_CODIGO]
) ON [PRIMARY]
) ON [PRIMARY]
GO
CREATE TABLE [ACT_DES] (
    [ADE_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [ACT_CODIGO] [int] NULL ,
    [DES_CODIGO] [int] NULL ,
    [ADE_VALOR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
CONSTRAINT [PK_ACT_DES] PRIMARY KEY CLUSTERED
(
    [ADE_CODIGO]

```

```

) ON [PRIMARY],
CONSTRAINT [FK_ACT_DES_ACTIVIDADES] FOREIGN KEY
(
    [ACT_CODIGO]
) REFERENCES [ACTIVIDADES] (
    [ACT_CODIGO]
),
CONSTRAINT [FK_ACT_DES_DECISIONES] FOREIGN KEY
(
    [DES_CODIGO]
) REFERENCES [DECISIONES] (
    [DES_CODIGO]
)
) ON [PRIMARY]
GO

CREATE TABLE [RUTAS] (
    [RUT_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [RUT_ACT_PADRE] [int] NULL ,
    [RUT_ACT_ACTUAL] [int] NULL ,
    [RUT_ACT_VALOR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [RUT_ACT_HIJO] [int] NULL ,
    [RUT_ACT_HIJO_POS] [int] NULL ,
    [RUT_GRUPO] [int] NULL ,
    [RUT_ORDEN] [int] NULL ,
    CONSTRAINT [PK_RUTAS] PRIMARY KEY CLUSTERED
    (
        [RUT_CODIGO]
    ) ON [PRIMARY],
    CONSTRAINT [FK_RUTAS_ACTIVIDADES] FOREIGN KEY
    (
        [RUT_ACT_PADRE]
    ) REFERENCES [ACTIVIDADES] (
        [ACT_CODIGO]
    ),
    CONSTRAINT [FK_RUTAS_ACTIVIDADES1] FOREIGN KEY
    (
        [RUT_ACT_ACTUAL]
    ) REFERENCES [ACTIVIDADES] (
        [ACT_CODIGO]
    ),
    CONSTRAINT [FK_RUTAS_ACTIVIDADES2] FOREIGN KEY
    (
        [RUT_ACT_HIJO]
    ) REFERENCES [ACTIVIDADES] (
        [ACT_CODIGO]
    )
) ON [PRIMARY]
GO

CREATE TABLE [POSICIONES] (
    [POS_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [PRO_CODIGO] [int] NULL ,
    [ACT_CODIGO] [int] NULL ,
    [POS_X1] [float] NULL ,
    [POS_Y1] [float] NULL ,
    [POS_X2] [float] NULL ,
    [POS_Y2] [float] NULL ,
    [POS_X3] [float] NULL ,
    [POS_Y3] [float] NULL ,
    [POS_X4] [float] NULL ,
    [POS_Y4] [float] NULL ,
    CONSTRAINT [PK_POSICIONES] PRIMARY KEY CLUSTERED
    (
        [POS_CODIGO]
    ) ON [PRIMARY],
    CONSTRAINT [FK_POSICIONES_ACTIVIDADES] FOREIGN KEY
    (
        [ACT_CODIGO]
    ) REFERENCES [ACTIVIDADES] (
        [ACT_CODIGO]
    ),
    CONSTRAINT [FK_POSICIONES_PROCESOS] FOREIGN KEY
    (
        [PRO_CODIGO]
    ) REFERENCES [PROCESOS] (

```

```

        [PRO_CODIGO]
    )
) ON [PRIMARY]
GO

CREATE TABLE [REG_POL] (
    [RPO_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [ACT_CODIGO] [int] NULL ,
    [RPO_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [RPO_DESCRIPCION] [text] COLLATE Modern_Spanish_CI_AS NULL ,
    [RPO_TIPO] [int] NULL ,
    [RPO_VALOR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [TIP_ESTADO] [int] NOT NULL ,
    CONSTRAINT [PK_REG_POL] PRIMARY KEY CLUSTERED
    (
        [RPO_CODIGO]
    ) ON [PRIMARY] ,
    CONSTRAINT [FK_REG_POL_ACTIVIDADES] FOREIGN KEY
    (
        [ACT_CODIGO]
    ) REFERENCES [ACTIVIDADES] (
        [ACT_CODIGO]
    )
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO

CREATE TABLE [EVENTOS] (
    [EVE_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [EVE_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [EVE_DESCRIPCION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [EVE_ACCION] [text] COLLATE Modern_Spanish_CI_AS NULL ,
    CONSTRAINT [PK_EVENTOS] PRIMARY KEY CLUSTERED
    (
        [EVE_CODIGO]
    ) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO

CREATE TABLE [ACT_EVE] (
    [AEV_CODIGO] [int] NOT NULL ,
    [ACT_CODIGO] [int] NULL ,
    [EVE_CODIGO] [int] NULL ,
    CONSTRAINT [PK_ACT_EVE] PRIMARY KEY CLUSTERED
    (
        [AEV_CODIGO]
    ) ON [PRIMARY] ,
    CONSTRAINT [FK_ACT_EVE_ACTIVIDADES] FOREIGN KEY
    (
        [ACT_CODIGO]
    ) REFERENCES [ACTIVIDADES] (
        [ACT_CODIGO]
    ) ,
    CONSTRAINT [FK_ACT_EVE_EVENTOS] FOREIGN KEY
    (
        [EVE_CODIGO]
    ) REFERENCES [EVENTOS] (
        [EVE_CODIGO]
    )
) ON [PRIMARY]
GO

CREATE TABLE [ROLES] (
    [ROL_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [ROL_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [ROL_DESCRIPCION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    CONSTRAINT [PK_ROLES] PRIMARY KEY CLUSTERED
    (
        [ROL_CODIGO]
    ) ON [PRIMARY]
) ON [PRIMARY]
GO

CREATE TABLE [USUARIOS] (
    [USU_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [USU_APELLIDO] [nvarchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [USU_SEGUNDOAPELLIDO] [nvarchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,

```

```

[USU_NOMBRE] [nvarchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[USU_SEGUNDONOMBRE] [nvarchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[USU_CONTRASENIA] [nvarchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[USU_FECHAN] [datetime] NULL ,
[TIP_CONTRATACION] [int] NULL ,
[TIP_USUARIO] [int] NULL ,
[TIP_PERFIL] [int] NULL ,
[USU_TRATAMIENTO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[TIP_ESTADO] [int] NULL ,
[USU_ID_SESION] [nvarchar] (25) COLLATE Modern_Spanish_CI_AS NULL ,
[USU_MAIL] [nvarchar] (200) COLLATE Modern_Spanish_CI_AS NULL ,
[CAMBIARCLAVE] [int] NULL ,
CONSTRAINT [PK_USUARIOS] PRIMARY KEY CLUSTERED
(
    [USU_CODIGO]
) ON [PRIMARY]
) ON [PRIMARY]
GO

```

```

CREATE TABLE [ORGANICO] (
    [ORG_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [NIV_CLAVE] [int] NULL ,
    [DIR_CLAVE] [int] NULL ,
    [SEC_CLAVE] [int] NULL ,
    [DESCRIPCION] [varchar] (255) COLLATE Modern_Spanish_CI_AS NULL ,
    [CODIGO] [int] NULL ,
    [EST_CLAVE] [int] NULL ,
    [USU_CODIGO] [int] NULL ,
    [TIPO] [int] NULL ,
    [USU_ENCARGADO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    CONSTRAINT [PK_ORGANICO] PRIMARY KEY CLUSTERED
(
    [ORG_CODIGO]
) ON [PRIMARY]
) ON [PRIMARY]
GO

```

```

CREATE TABLE [ORG_ROL] (
    [ORO_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [ORG_CODIGO] [int] NULL ,
    [ROL_CODIGO] [int] NULL ,
    CONSTRAINT [PK_ORG_ROL] PRIMARY KEY CLUSTERED
(
    [ORO_CODIGO]
) ON [PRIMARY] ,
CONSTRAINT [FK_ORG_ROL_ORGANICO] FOREIGN KEY
(
    [ORG_CODIGO]
) REFERENCES [ORGANICO] (
    [ORG_CODIGO]
),
CONSTRAINT [FK_ORG_ROL_ROLES] FOREIGN KEY
(
    [ROL_CODIGO]
) REFERENCES [ROLES] (
    [ROL_CODIGO]
)
) ON [PRIMARY]
GO

```

```

CREATE TABLE [ORG_ROL_ACT] (
    [ORA_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [ORO_CODIGO] [int] NULL ,
    [ACT_CODIGO] [int] NULL ,
    CONSTRAINT [PK_ORG_ROL_ACT] PRIMARY KEY CLUSTERED
(
    [ORA_CODIGO]
) ON [PRIMARY] ,
CONSTRAINT [FK_ORG_ROL_ACT_ORG_ROL] FOREIGN KEY
(
    [ORO_CODIGO]
) REFERENCES [ORG_ROL] (
    [ORO_CODIGO]
)
) ON [PRIMARY]
GO

```

```

CREATE TABLE [ORG_ROL_USU] (
    [UOR_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [ORO_CODIGO] [int] NULL ,
    [USU_CODIGO] [int] NULL ,
    CONSTRAINT [PK_USU_ORG_ROL] PRIMARY KEY CLUSTERED
    (
        [UOR_CODIGO]
    ) ON [PRIMARY] ,
    CONSTRAINT [FK_ORG_ROL_USU_ORG_ROL] FOREIGN KEY
    (
        [ORO_CODIGO]
    ) REFERENCES [ORG_ROL] (
        [ORO_CODIGO]
    )
) ON [PRIMARY]
GO

CREATE TABLE [ORG_ROL_PAG] (
    [ORP_CODIGO] [int] IDENTITY (1, 1) NOT NULL ,
    [ORO_CODIGO] [int] NULL ,
    [PAG_CODIGO] [int] NULL ,
    CONSTRAINT [PK_USU_PAG] PRIMARY KEY CLUSTERED
    (
        [ORP_CODIGO]
    ) ON [PRIMARY] ,
    CONSTRAINT [FK_ORG_ROL_PAG_ORG_ROL] FOREIGN KEY
    (
        [ORO_CODIGO]
    ) REFERENCES [ORG_ROL] (
        [ORO_CODIGO]
    ) ,
    CONSTRAINT [FK_ORG_ROL_PAG_PAGINAS] FOREIGN KEY
    (
        [PAG_CODIGO]
    ) REFERENCES [PAGINAS] (
        [PAG_CODIGO]
    ) ,
    CONSTRAINT [FK_USU_PAG_USUARIOS] FOREIGN KEY
    (
        [ORO_CODIGO]
    ) REFERENCES [USUARIOS] (
        [USU_CODIGO]
    )
) ON [PRIMARY]
GO

CREATE TABLE [EXPEDIENTES] (
    [EXP_TIPO] [int] NOT NULL ,
    [EXP_NUMERO] [int] NOT NULL ,
    [EXP_ANIO] [int] NOT NULL ,
    [ARC_NUMERO] [int] NULL ,
    [USU_CODIGO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [TIP_UBICACIONACTUAL] [varchar] (255) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_FUN_INS] [varchar] (255) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_TELEFONO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_RESUMEN] [text] COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_FIRMA] [varchar] (200) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_CEDULARUC] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_NOTIFICACIONES] [text] COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_CREADO] [datetime] NULL ,
    [TIP_PROCESO] [int] NULL ,
    [TIP_ACTIVIDAD] [int] NULL ,
    [TIP_CLASE] [int] NULL ,
    [TIP_PRIORIDAD] [int] NULL ,
    [TIP_ESTADO] [int] NULL ,
    [EXP_LUGAR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_FECHA] [datetime] NULL ,
    [EXP_HORA] [varchar] (5) COLLATE Modern_Spanish_CI_AS NULL ,
    [PRV_CODIGO] [int] NULL ,
    [CAN_CODIGO] [int] NULL ,
    [PAR_CODIGO] [int] NULL ,
    CONSTRAINT [PK_EXPEDIENTES] PRIMARY KEY CLUSTERED
    (
        [EXP_TIPO],
        [EXP_NUMERO],
    )
)

```

```

        [EXP_ANIO]
    ) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO

```

```

CREATE TABLE [TAREAS] (
    [TAR_CODIGO] [int] NOT NULL ,
    [TAR_FECHA] [datetime] NULL ,
    [TAR_ANIO] [int] NULL ,
    [TAR_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NOT NULL ,
    [USU_ID_SESION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [PRO_CODIGO] [int] NULL ,
    [TAR_PRIORIDAD] [int] NULL ,
    [EXP_NUMERO] [int] NULL ,
    [EXP_ANIO] [int] NULL ,
    [TIP_ESTADO] [int] NOT NULL ,
    CONSTRAINT [PK_TAREAS] PRIMARY KEY CLUSTERED
(
    [TAR_CODIGO]
) ON [PRIMARY]
) ON [PRIMARY]
GO

```

```

CREATE TABLE [DOCUMENTOS] (
    [DOC_CODIGO] [int] NOT NULL ,
    [EXP_TIPO] [int] NULL ,
    [EXP_NUMERO] [int] NULL ,
    [EXP_ANIO] [int] NULL ,
    [DOC_REGISTRO] [datetime] NULL ,
    [DOC_HOST] [char] (100) COLLATE Modern_Spanish_CI_AS NULL ,
    [DOC_NOMBRE] [char] (255) COLLATE Modern_Spanish_CI_AS NULL ,
    [DOC_TAMANIO] [char] (10) COLLATE Modern_Spanish_CI_AS NULL ,
    [DOC_NOTA] [char] (255) COLLATE Modern_Spanish_CI_AS NULL ,
    [USU_CODIGO] [char] (40) COLLATE Modern_Spanish_CI_AS NULL ,
    [DOC_ESTADO] [int] NOT NULL ,
    CONSTRAINT [PK_DOCUMENTOS] PRIMARY KEY CLUSTERED
(
    [DOC_CODIGO]
) ON [PRIMARY] ,
    CONSTRAINT [FK_DOCUMENTOS_EXPEDIENTES] FOREIGN KEY
(
    [EXP_TIPO],
    [EXP_NUMERO],
    [EXP_ANIO]
) REFERENCES [EXPEDIENTES] (
    [EXP_TIPO],
    [EXP_NUMERO],
    [EXP_ANIO]
)
) ON [PRIMARY]
GO

```

```

CREATE TABLE [HISTORICO] (
    [HIS_NUMERO] [decimal](18, 0) NOT NULL ,
    [HIS_FECHA] [datetime] NULL ,
    [HIS_ACCION] [varchar] (200) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_TIPO] [int] NULL ,
    [EXP_NUMERO] [int] NULL ,
    [EXP_ANIO] [int] NULL ,
    [USU_CODIGO] [varchar] (40) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_CREADO] [datetime] NULL ,
    [TIP_PROCESO] [varchar] (25) COLLATE Modern_Spanish_CI_AS NULL ,
    [TIP_ACTIVIDAD] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [TIP_UBICACIONACTUAL] [varchar] (100) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_FUN_INS] [varchar] (100) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_TELEFONO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_RESUMEN] [varchar] (500) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_FIRMA] [varchar] (100) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_CEDULARUC] [varchar] (200) COLLATE Modern_Spanish_CI_AS NULL ,
    [EXP_NOTIFICACIONES] [varchar] (500) COLLATE Modern_Spanish_CI_AS NULL ,
    [TIP_CLASE] [varchar] (100) COLLATE Modern_Spanish_CI_AS NULL ,
    [TIP_PRIORIDAD] [varchar] (15) COLLATE Modern_Spanish_CI_AS NULL ,
    [TIP_ESTADO] [int] NOT NULL ,
    [HOST] [varchar] (100) COLLATE Modern_Spanish_CI_AS NULL ,
    CONSTRAINT [PK_HISTORICO] PRIMARY KEY CLUSTERED
(

```

```

        [HIS_NUMERO]
    ) ON [PRIMARY],
    CONSTRAINT [FK_HISTORICO_EXPEDIENTES] FOREIGN KEY
    (
        [EXP_TIPO],
        [EXP_NUMERO],
        [EXP_ANIO]
    ) REFERENCES [EXPEDIENTES] (
        [EXP_TIPO],
        [EXP_NUMERO],
        [EXP_ANIO]
    )
) ON [PRIMARY]
GO
CREATE TABLE [ADJUNTOS] (
    [ADJ_CODIGO] [int] IDENTITY (1, 1) NOT NULL,
    [TAR_CODIGO] [int] NOT NULL,
    [ACT_CODIGO] [int] NOT NULL,
    [DOC_CODIGO] [int] NULL,
    CONSTRAINT [PK_ADJUNTOS] PRIMARY KEY CLUSTERED
    (
        [ADJ_CODIGO]
    ) ON [PRIMARY],
    CONSTRAINT [FK_ADJUNTOS_ACTIVIDADES] FOREIGN KEY
    (
        [ACT_CODIGO]
    ) REFERENCES [ACTIVIDADES] (
        [ACT_CODIGO]
    ),
    CONSTRAINT [FK_ADJUNTOS_DOCUMENTOS] FOREIGN KEY
    (
        [DOC_CODIGO]
    ) REFERENCES [DOCUMENTOS] (
        [DOC_CODIGO]
    ),
    CONSTRAINT [FK_ADJUNTOS_TAREAS] FOREIGN KEY
    (
        [TAR_CODIGO]
    ) REFERENCES [TAREAS] (
        [TAR_CODIGO]
    )
) ON [PRIMARY]
GO
CREATE TABLE [EJECUTADAS] (
    [EJE_CODIGO] [int] NOT NULL,
    [TAR_CODIGO] [int] NOT NULL,
    [ACT_CODIGO] [int] NOT NULL,
    [ORG_CODIGO] [int] NOT NULL,
    [TIP_CARGO] [int] NULL,
    [USU_CODIGO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL,
    [RUT_ACT_VALOR] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL,
    [EJE_FECHA] [datetime] NULL CONSTRAINT [DF_EJECUTADAS_EJE_FECHA] DEFAULT (getdate()),
    CONSTRAINT [PK_EJECUTADAS] PRIMARY KEY CLUSTERED
    (
        [EJE_CODIGO]
    ) ON [PRIMARY],
    CONSTRAINT [FK_EJECUTADAS_ACTIVIDADES] FOREIGN KEY
    (
        [ACT_CODIGO]
    ) REFERENCES [ACTIVIDADES] (
        [ACT_CODIGO]
    ),
    CONSTRAINT [FK_EJECUTADAS_ORGANICO] FOREIGN KEY
    (
        [ORG_CODIGO]
    ) REFERENCES [ORGANICO] (
        [ORG_CODIGO]
    ),
    CONSTRAINT [FK_EJECUTADAS_TAREAS] FOREIGN KEY
    (
        [TAR_CODIGO]
    ) REFERENCES [TAREAS] (
        [TAR_CODIGO]
    )
) ON [PRIMARY]
GO

```

5.5 *Desarrollo del Código PHP*

El código PHP se encuentra plasmado en el aplicativo por lo cual pido dirigirse al CD adjunto del aplicativo el cual contiene toda la información del código fuente del software.

5.6 *Manual de Programador*

El desarrollo del software generó varios archivos que deben ser detallados junto con su funcionamiento, con esto se comprenderá la ejecución de cada uno de los archivos y se podrá realizar cambios si es tomado el presente trabajo como tema de estudio para futuros proyectos.

5.6.1 **Tipos de Archivo**

Se emplearon varios tipos de archivos comunes en la programación web, script DLL, PHP, JSP, SWF y por último CSS.

5.6.1.1 **Archivos DLL**

La extensión de estos archivos no tiene relación alguna con las librerías dll de Windows, se utilizó simplemente como seguridad para que se oculte el tipo de archivo real ya que tienen script en php.

Comenzamos por este tipo de archivos ya que mantienen estrecha relación con cada uno de los scripts en PHP.

FuncionesAuth.dll

El Archivo funcionesauth.dll, contiene las funciones de autenticación del sistema

```
function authenticate($mensaje){}
```

Función que utiliza el método de autenticación de PHP, el parámetro \$mensaje es el mensaje utilizado por el programador para que se visualice por el usuario.

```
function autenticar($USER,$PW,$PAGINA=0,$mensaje='Autenticación  
Usuarios'){}
```

Función que utiliza las variables \$PHP_AUTH_USER,\$PHP_AUTH_PW del array \$_SERVER para autenticar contra los datos de la Base de Datos.

```
function baseconecta(){}
```

Función que genera la cadena de conexión con la base de datos y coloca el resultado en la variable \$conn.

```
function basedesconecta($conn){}
```

Función que con la variable \$conn cierra la conexión con la base de datos.

```
LLenarSelect($sql,$activo=0,$ini,$m=1,$l=80){}
```

Función que se utiliza para llenar una etiqueta <select> con información de la base de datos, sus parámetros de entrada son:

- \$sql la sentencia para buscar en la base de datos
- \$activo el código activado inicialmente
- \$x número de campos a mostrar
- \$ini El mensaje al valor seleccionado inicialmente
- \$m Mostrar solo en mayúsculas
- \$l Longitud máxima del select

DevuelveCadena(\$sql,\$x=1, \$m=1){}

Función que devuelve un valor solicitado mediante un SELECT a la base de datos, sus parámetros son:

- \$sql la sentencia para buscar en la base de datos
- \$x número de campos a mostrar
- \$m Mostrar solo en mayúsculas = 1

EjecutaSQL(\$sql,\$x=1){}

Función que ejecuta una sentencia sql, sus parámetros son:

- \$sql la sentencia que se va a ejecutar
- \$i si desea imprimir la sentencia SQL

function

BuscaSiguiente(\$fieldTAR_CODIGO,\$fieldACT_CODIGO,\$fieldRUT_ACT_VALOR){}

Función destinada a buscar la actividad siguiente dependiendo de los valores que fueron enviados como parámetros, los mismos que son:

- \$fieldTAR_CODIGO Código de la Tarea
- \$fieldACT_CODIGO Actividad en la que se encuentra la Tarea
- \$fieldRUT_ACT_VALOR Valor que tomo la actividad en una decisión.

El resultado a generarse en esta función es registrado en la base de datos en la tabla Ejecutadas.

function BuscaAsignaciones(\$fieldRUT_ACT_HIJO){}

Función destinada a buscar las asignaciones diarias de cada usuario dependiendo de la actividad, esta función es llamada en el caso de que la actividad tiene que ser asignada entre los usuarios de forma equivalente, su parámetro es:

- \$fieldRUT_ACT_HIJO Código de la Actividad siguiente.

Funciones.dll

Archivo con funciones limitadas que se encuentra incluido en script PHP de consulta a usuarios invitados.

function baseconecta(){}

Función que genera la cadena de conexión con la base de datos y coloca el resultado en la variable \$conn.

function basedesconecta(\$conn){}

Función que con la variable \$conn cierra la conexión con la base de datos.

LLenarSelect(\$sql,\$activo=0,\$x,\$ini,\$m=1,\$l=80){}

Función que se utiliza para llenar una etiqueta <select> con información de la base de datos, sus parámetros de entrada son:

- \$sql la sentencia para buscar en la base de datos
- \$activo el código activado inicialmente
- \$x número de campos a mostrar
- \$ini El mensaje al valor seleccionado inicialmente
- \$m Mostrar solo en mayúsculas
- \$l Longitud máxima del select

DevuelveCadena(\$sql,\$x=1, \$m=1){}

Función que devuelve un valor solicitado mediante un SELECT a la base de datos, sus parámetros son:

- \$sql la sentencia para buscar en la base de datos
- \$x número de campos a mostrar
- \$m Mostrar solo en mayúsculas = 1

EjecutaSQL(\$sql,\$x=1){}

Función que ejecuta una sentencia sql, sus parámetros son:

- \$sql la sentencia que se va a ejecutar
- \$i i si desea imprimir la sentencia SQL

5.6.1.2 Archivos PHP

Son los archivos base del aplicativo, aquí observaremos tres grupos El primer grupo lo denominaremos MOTOR, el segundo grupo ADMIN, y el tercero USUARIO.

5.6.1.2.1 Motor

En este grupo se encuentran todos los archivos esenciales para que el workflow se distinga del resto de aplicaciones, aquí se generan los procesos, actividades y rutas con el conjunto de reglas y políticas que los preceden.

Para todos los script se incluyen los siguientes archivos:

- FuncionesAuth.dll
- Forms.css

Actividades.php

Descripción

El script manipula las actividades de la tabla Actividades pertenecientes a cada uno de los procesos, realiza procedimientos de inserción, modificación, eliminación y consulta, toma información de la tabla Paginas para asignar la pagina que corresponde a dicha actividad, de la misma forma ingresa los distintos tipos de roles que pueden ser asignados a cada una de las actividades mediante el

archivo *listusu.php*, y asigna reglas y políticas a estas actividades con el archivo *rpo_act.php*.

Funciones

Utiliza procedimientos AJAX y estos lo enlazan con el archivo *procedimientos_ajax.php*

function eliminaPagina(tipo){}

Función encargada de revisar que el parámetro actual no sea 1033 que significa actividad de tipo decisión y por lo tanto deshabilita la elección de Página caso contrario envía al archivo *procedimientos.php* los parámetros de archivo con el valor “actividades” y acción con el valor “paginas”, sus parámetros son:

- Tipo parámetro que contiene el valor del tipo de actividad

function AbrePagina(pagina,feature){}

Función encargada de abrir una pagina nuevo, sus parámetros son:

- Pagina Contiene el nombre de la pagina que va a ser abierta.
- Feature Contiene las características de la pagina a ser abierta.

function EscojeEjecutor(tipo){}

Función encargada de abrir la pagina *listusu.php*, y sus parámetros son:

- Tipo Contiene el tipo de ejecutor al que va a ser asignado el valor que devolverá la pagina *listusu.php*

Act_dec.php

Descripción

El script genera un listado de las decisiones posibles para las actividades de tipo decisión, el listado nos sirve para realizar inserciones de las asociaciones en la tabla ACT_DEC con la actividad. Se manejan procedimientos de inserción, consulta y eliminación.

Contenido.php

Descripción

El script genera una pantalla que visualiza el administrador al momento de ingresar por primera vez. No realiza una función sobre ningún otro archivo, simplemente tiene incluido los archivos que todo script php utiliza, en este aplicativo.

Decisiones.php

Descripción

El script manipula la tabla Decisiones que aparecerán en el listado generado en el script *act_dec.php*, realiza procedimientos de inserción, modificación, eliminación y consulta.

Diagramaspreorden.php

Descripción

Script especial que se encarga de dibujar el diagrama perteneciente a cada proceso, sus correspondientes características y su simbología.

Archivos Inclusión

Para el desarrollo de esta etapa se utilizo una librería especial llamada jpgraph.

- jpgraph.php
- jpgraph_canvas.php
- jpgraph_canvtools.php

Funciones

```
function GrabaPosicion($fieldPRO_CODIGO , $fieldACT_CODIGO, $posx, $posy){}
```

Función encargada de realizar una inserción en la tabla Posiciones de la base de datos, sus parámetros son:

- \$fieldPRO_CODIGO Código del Proceso
- \$fieldACT_CODIGO Código de la Actividad
- \$posx Posición de la Actividad en X
- \$posy Posición de la Actividad en Y

```
function BuscarHijos($fieldPRO_CODIGO, $actividad, $posx, $posy, $imagen){}
```

Función encargada de devolver y dibujar la siguiente actividad en relación a la actividad dibujada (padre), sus parámetros son:

- \$fieldPRO_CODIGO Código del Proceso
- \$actividad Actividad Padre
- \$posx Posición en X de la Actividad padre
- \$posy Posición en Y de la Actividad padre
- \$imagen Parámetro que contiene a la imagen antes de ser dibujada.

```
function Simbologia($imagen, $anchoimagen, $saltoimagen){}
```

Función que se encarga de dibujar el significado de cada símbolo en la grafica, sus parámetros son:

- \$imagen Parámetro que contiene a la imagen antes de ser dibujada.
- \$anchoimagen Parámetro que contiene el valor del ancho de la imagen.
- \$saltoimagen Parámetro que contiene el valor del ancho de la imagen.

function

```
Caracteristicas($imagen, $anchoimagen, $saltoimagen, $fieldPRO_CODIGO){}
```

Función que dibujo las características del proceso, sus parámetros son:

- \$imagen Parámetro que contiene a la imagen antes de ser dibujada.
- \$anchoimagen Parámetro que contiene el valor del ancho de la

- `$saltoimagen` imagen.
Parámetro que contiene el valor del ancho de la imagen.
- `$fieldPRO_CODIGO` Código del Proceso

Index.php

Descripción

Script que contiene los marcos (frames) que dividen a la pantalla.

Listusu.php

Descripción

Script que manipula la lista de usuarios y la relación que existe con los roles y el departamento en el que trabajen.

Funciones

Las funciones aquí utilizadas fueron realizadas en código javascript, utiliza procedimientos AJAX y estos lo enlazan con el archivo *proceso_general.php*

```
function cargaLista(Dep,Rol,Usu){}
```

Función encargada de enviar los parámetros de departamento (Dep), rol (Rol) y usuario (Usu), estos tres parámetros pueden contener o no valores.

```
function PasarDatos(oro_codigo,des){}
```

Función encargada de colocar los datos del formulario que se encontraban en campos ocultos, a la ventana de la cual fue llamado (*actividades.php*)

Menu.php

Descripción

Script encargado de armar el menú con las opciones que se encuentran almacenadas en la base de datos en relación al motor del WorkFlow.

Funciones

```
function BuscarHijos($fieldMEN_CODIGO,$num,$numh){}
```

Función encargada de encontrar todos los hijos pertenecientes al nodo padre o menú de nivel superior, sus parámetros son:

- `$fieldMEN_CODIGO` Código del menú padre
- `$num` Código perteneciente al nodo padre (js)
- `$numh` Código que pertenecerá al nodo hijo (js)

Paginas.php

Descripción

Script que manipula a la tabla Paginas de la base de datos, realiza procedimientos de inserción, modificación, eliminación y consulta, estos datos serán utilizados en la *actividades.php* para asignar la pagina correspondiente a la actividad.

Proceso_general.php

Descripción

Script que se encarga de devolver el listado mediante petición por parte de la función *cargaLista(Dep,Rol,Usu)* perteneciente al archivo *listusu.php*, este listado se armara dependiendo de la existencia o no de los valores enviados como parámetros al archivo *proceso_general.php*.

Procesos.php

Script que se encarga de manipular la información de la tabla Procesos de la base de datos, realiza procedimientos de inserción, modificación, eliminación y consulta.

Proceso_ajax.php

Script que arma un select con las páginas disponibles para dicha actividad. Obtiene la información de la tabla Paginas.

Reglas.php

Descripción

Script encargado de manipular la tabla REG_POL de la base de datos, realiza procedimientos de inserción, modificación, eliminación y consulta, estos datos corresponden a las reglas y políticas correspondientes a cada una de las actividades de tipo Común.

Rpo_act.php

Descripción

Script encargado de manipular la asociación de las actividades con las reglas y políticas que correspondan dentro de la tabla REG_POL, realiza procedimientos de modificación y consulta.

Rutas.php

Descripción

Script encargado de manipular la tabla Rutas de la base de datos, realiza procedimientos de inserción, modificación, eliminación, y consulta, Existe una relación con el archivo *Diagramapreorden.php*.

5.6.1.2.2 Admin

En este grupo se encuentran todos los archivos de administración de usuarios su lugar dentro del aplicativo.

Para todos los a script se incluyen los siguientes archivos:

- FuncionesAuth.dll
- Forms.css

Contenido.php

Descripción

El script genera una pantalla que visualiza el administrador al momento de ingresar a esta parte de la aplicación. No realiza una función sobre ningún otro archivo, simplemente tiene incluido los archivos que todo script php utiliza.

Index.php

Descripción

Script que contiene los marcos (frames) que dividen a la pantalla.

Menu.php

Descripción

Script encargado de armar el menú con las opciones que se encuentran almacenadas en la base de datos en relación al motor del WorkFlow.

Funciones

```
function BuscarHijos($fieldMEN_CODIGO,$num,$numh){}
```

Función encargada de encontrar todos los hijos pertenecientes al nodo padre o menú de nivel superior, sus parámetros son:

- *\$fieldMEN_CODIGO* Código del menú padre
- *\$num* Código perteneciente al nodo padre (js)
- *\$numh* Código que pertenecerá al nodo hijo (js)

Org_rol.php

Descripción

Script encargado de manipular la tabla ORG_ROL de la base de datos, la asociación de roles con el orgánico funcional, realiza procedimientos de inserción, modificación, eliminación y consulta.

Org_rol_usu.php

Descripción

Script encargado de manipular la tabla ORG_ROL_USU de la base de datos, su cometido es insertar la asociación de un usuario con el organico-rol correspondiente, realiza procedimientos de inserción, modificación, eliminación y consulta.

Organico.php

Descripción

Script que manipula la tabla Organico de la base de datos, realiza procedimientos de inserción, modificación, eliminación y consulta.

Roles.php

Descripción

Script que manipula la tabla Roles de la base de datos, realiza procedimientos de inserción, modificación, eliminación y consulta.

Usuarios.php

Descripción

Script que manipula la tabla Usuarios de la base de datos, realiza procedimientos de inserción, modificación, eliminación y consulta.

5.6.1.2.3 Usuarios

En este grupo se encuentran todos los archivos de ejecución del aplicativo.

Para todos los script se incluyen los siguientes archivos:

- FuncionesAuth.dll
- Forms.css

Activarcookies.php

Descripción

Script encargado de mostrar la información necesaria para activar las cookies dependiendo del explorador que manejen.

Borde.php

Descripción

Script que se encarga de mostrar el borde superior y que forma parte de uno de los archivos marco de la parte derecha del marco principal *contenido.php*. Otra funcionalidad es el llamado que realiza dentro de su formulario para búsquedas al archivo *Busqueda.php*.

Busqueda.php

Descripción

Script que se encarga de generar el marco dentro del marco contenido, haciendo una llamada a *borde.php* y *busquedaejecucion.php*.

Busquedaejecucion.php

Descripción

Script que se encarga de realizar la búsqueda de un proceso y de armar el respectivo resultado.

Contenido.php

Descripción

Script que muestra la información de los procesos pendientes que pertenecen al rol de usuario autenticado, además es el encargado de llamar a la función *BuscaSiguiente(\$fieldTAR_CODIGO,\$fieldACT_CODIGO,\$fieldRUT_ACT_V ALOR)*, detallada en el archivo *FuncionesAuth.dll*.

Cookies.php

Descripción

Script que contiene el marco de división cuando muestran explicación de cómo activar los cookies del explorador. Los archivos son *Borde.php* y *activarcookies.php*

Decisiones.php

Descripción

Script que se encarga de mostrar las posibles acciones a tomar por parte del usuario autenticado dentro de la actividad actual. El valor se almacena en una cookie denominada *fieldRUT_ACT_VALOR*.

Documentos.php

Descripción

Script encargado de adjuntar los documentos, subirlos al servidor y registrarlos en las Tablas Documentos y Adjuntos de la base de datos, realiza procedimientos de inserción, eliminación y consulta.

Estadisticas.php

Descripción

Script encargado de dibujar estadísticas del proceso activo, utilizando la tabla Ejecutadas de la base de datos.

Para el desarrollo de esta etapa se utilizó una librería especial llamada jgraph.

- jgraph.php
- jgraph_bar.php
- jgraph_line.php

Formularios.php

Descripción

Script que contiene los enlaces mediante etiquetas html <iframe> de la página correspondiente a la actividad, *Documentos.php* y *Decisiones.php*.

Historico.php

Descripción

Script encargado de armar un listado de los procesos pertenecientes al usuario autenticado dependiendo de la variable tipo enviada mediante parámetro a la misma.

Index.php

Descripción

Script que contiene los marcos (frames) que dividen a la pantalla.

Menu.php

Descripción

Script encargado de armar el menú con las opciones que se encuentran almacenadas en la base de datos en relación al motor del WorkFlow.

Funciones

```
function BuscarHijos($fieldMEN_CODIGO,$num,$numh){}
```

Función encargada de encontrar todos los hijos pertenecientes al nodo padre o menú de nivel superior, sus parámetros son:

- *\$fieldMEN_CODIGO* Código del menú padre
- *\$num* Código perteneciente al nodo padre (js)
- *\$numh* Código que pertenecerá al nodo hijo (js)

L_estadisticas.php

Descripción

Script encargado de listar las estadísticas de cada uno de los procesos activos del aplicativo, para ello utiliza el archivo *estadisticas.php*.

Redirección.php

Descripción

Script que se encarga de registrar en la tabla Tareas de la base de datos la información correspondiente a la nueva tarea y a continuación redirecciona dependiendo de la actividad relacionada al proceso escogido por el usuario autenticado.

Verdocumentos.php

Descripción

Script que permite visualizar los documentos adjuntos que se encuentran de lado del servidor.

5.6.1.3 Archivos JSP

Son los archivos de ayuda del aplicativo, como herramienta para el funcionamiento mediante el código javascript.

Cookies.js

Descripción

Script que contiene un conjunto de funciones que trata con cookies del explorador.

Funciones

function CojerCookie(nombre) {}

Función encargada de tomar el valor de la cookie, sus parámetros son:

- Nombre Parámetro que contiene el nombre de la cookie

function Caduca(dias) {}

Función que devuelve el valor de la fecha actual + un número de días, sus parámetros son:

- dias Parámetro que contiene el valor en días.

function GuardarCookie (nombre, valor) {}

Función que guarda el valor de una cookie, sus parámetros son:

- nombre Nombre de la cookie
- valor Valor de la cookie a ser guardada

function BorrarCookie(nombre) {}

Borra una cookie haciéndola expirar, sus parámetros son:

- nombre Nombre de la cookie

function test() {}

Función que revisa si la cookie “prueba” pudo ser creada si el valor es falso referencia a la pagina *cookies.php* dentro del marco contenido.

function ValidarTarea(cookie, desvio){}

Función encargada de validar si la cookie que se crea cuando una actividad es completa fue creada, si es verdadera referencia a la página indicada por desvio, caso contrario no realiza ninguna acción, sus parámetros son:

- cookie Contiene el nombre de la cookie
- desvio Contiene una dirección de la pagina para desvió.

5.6.1.4 Archivos SWF

Archivos de tipo flash, que dan otro tipo de interactividad al aplicativo.

Documentos.swf

Archivo que permite que se pueda subir al servidor más de un archivo a la vez y que nos muestre información sobre el tiempo y bits subidos por segundo transcurrido.

5.6.1.5 Archivos CSS

Son archivos que definen el estilo de una interfaz sobre aspectos de la etiqueta html.

Forms.css

Descripción

Archivo que diseña la forma como serán visualizadas las etiquetas <input>, <textarea>, <form>, <select> y algunos estilos como tipos de letras, tamaños y bordes de una tabla.

5.7 Conclusiones

Los modelos de calidad nos dan cierta seguridad de la calidad de la aplicación, pudimos realizar las configuraciones esenciales tanto de Servidor HTTP y PHP como de la creación de la base de datos con su respectiva integridad, por ultimo se escribió a detalle el Manual de Programador con todos los archivos que forman parte del sistema.

CAPITULO VI
PRUEBAS

6 Pruebas

6.1 *Introducción*

En cualquier sistema complejo, es de vital importancia probarlo para saber si funcionará una vez este instalado. Una de las principales tareas en el hecho de pruebas es evitar que el programador sea el único que realice estas pruebas, ya que sabe como funciona y exactamente lo que se debe hacer, es por ello que en el presente trabajo se solicitó la ayuda de una persona sin conocimientos de ingeniería de sistemas para algunas de las pruebas que se realizaron.

6.2 *Pruebas de Unidad*

Hay ciertas condiciones que hacen que sea necesario realizar pruebas de unidad, una es que el programador, si este está aislado del sistema no se pueden realizar pruebas extremas y por el contrario si existe un aislamiento cometeremos los mismos errores. Otro problema es que si colocamos a un usuario no permite que nos muestre problemas de diseño, y por último el problema de no utilizar las mismas funciones para métodos similares.

El software cuenta con varios programas que interactúan con las librerías Funciones.dll, FuncionesAuth.dll, cookies.js, son estas las que contienen las funciones que vamos a probar.

Archivos DLL

Función Autenticar (Usuario, Password, Pagina, mensaje='Autenticación Usuarios')

La función Autenticar es fundamental ya que proporciona el acceso dependiendo de si la información proporcionada se encuentra en la tabla de usuarios.

Se verificó el caso de que no exista el usuario por lo que el sistema generó el mensaje de que no tiene acceso al sistema, también se comprobó que el usuario coloque un password erróneo produciendo el mismo resultado, otro caso es cuando el usuario ingresa con información existente en la tabla de usuarios.

Función baseconecta()

Esta función se encuentra establecida en los archivos dll y nos facilita la conexión al servidor .En todos los casos se conectó apropiadamente si por algún motivo no están levantados los servicios muestra un mensaje de error de no estar conectado.

Función LLenarSelect()

Esta función fue establecida para llenar la etiqueta select como su nombre lo indica. Las pruebas se realizaron llenando con código valido(SQL con resultados) y códigos erróneos, en el segundo caso no existió problemas ya que al tener problemas se evita que muestre resultados de mensajes de error.

Función DevuelveCadena()

Es una de las funciones mas utilizadas, al realizar pruebas se obtuvo resultados de error cuando la consulta no producía resultados y cuando la coincidencia era nula, por lo cual se mejoró con la restricción de devolución de resultados cuando ocurría los casos antes mencionados.

Función EjecutaSQL()

La función Ejecuta SQL se encarga de ejecutar el sql indicado cuando no se desea obtener un resultado tal es el caso del insert update delete. Se realizó las pruebas de forma satisfactoria y se insertó una actualización cuando la base de datos no esté levantada u ocurra un problema de cualquier tipo con la conexión.

Función FormatoFecha()

Esta función es bastante sencilla, nos proporciona la opción de obtener una cadena de caracteres con una fecha obtenida en formato completo aaaa:mm:dd hh:mm:ss y devuelve únicamente lo que necesitamos. Se realizaron las pruebas con todas las posibles devoluciones y no presento inconvenientes.

Función BuscaSiguiete(Tarea, Actividad, Valor)

Una de las funciones primordiales para que fluya el sistema, está encargada de verificar los valores actuales de la actividad y elegir la acción siguiente que deberá ejecutar el usuario. Para esta prueba se envió todas las opciones posibles de las actividades con los respectivos valores y se comprobaron que los resultados añadidos en la base de datos fueran los correctos.

Archivo Cookies.js

Función CojerValorCookie(indice)

Esta función nos permite obtener el valor de una cookie. Las pruebas se realizaron en dos ambientes con cookies habilitadas y deshabilitadas, en la opción de cookies habilitadas se pide el valor de una cookie cualquiera y nos devuelve el contenido sin ningún problema, en el caso de la deshabilitación de la cookie se encuentra el problema de la expiración de las cookies por lo que se actualiza el código realizando antes un refrescamiento y luego pidiendo el valor de la cookie. Una vez realizado estas correcciones las pruebas fueron satisfactorias.

Función CojerCookie(nombre)

Esta función busca entre las cookies el nombre de nuestra cookie, se verificó en dos casos, cuando la cookie existe y cuando no existe, en el primero nos ubica la cookie el resultado de éxito, en el otro caso nos indica que no existe la cookie.

Función Caduca(dias)

La función nos ayuda a obtener una fecha dependiendo del valor pasado como parámetro, se paso valores en números y letras en el caso de números los resultados fueron correctos, no en el caso de letras ya que produjo errores, ya que internamente se controla como valor fijo no es necesario tomar esta respuesta como problema.

Función GuardarCookie(nombre, valor)

Esta función se encarga de registrar la cookie con su valor, Se ingresaron valores cuando estaban habilitadas las cookies como cuando no, en ambas opciones la función respondió como se esperaba.

Función BorrarCookie(nombre)

Esta función permite expirar una cookie, se probó con varias cookies y el resultado fue exitoso.

Función test()

La función encargada de verificar si las cookies están habilitadas es una función sencilla que muestra un mensaje que pide habilitar las cookies y redirecciona a una página de ejemplo. Funciona exitosamente.

Función ValidarTarea(cookie, desvio)

Esta función es útil cuando queremos verificar si la tarea se completó. Verificamos cuando el programa está activo como cuando no, y se presentó un error ya que la cookie no se la expiraba por lo cual se corrigió el error y se pudo continuar tranquilamente.

Las Pruebas de Unidad no servirían de nada si no se realiza un análisis de cobertura, El análisis de cobertura nos ayuda a verificar que porcentaje de código fue revisado. Pudimos comprobar que el porcentaje revisado fue de un 93% del código útil, el resto son funciones que a lo largo de la carrera fueron reutilizadas y que de alguna manera pueden ser útiles en casos de actualizaciones del sistema.

6.3 Pruebas de Integración

Las pruebas de integración se realizan en realidad durante la construcción del sistema, ya que se basa en la interacción entre programas (módulos) del sistema, y es aquí donde la semejanza con los requerimientos de usuario comienzan a visualizarse.

Para el presente trabajo hemos utilizado las pruebas de integridad funcionales, encargadas de encontrar fallos en la respuesta de un programa cuando su trabajo depende de los servicios prestados por otros.

Interfaz de Usuario

Menú-Redirección

Comenzamos por el programa de menú y su relación con el programa de redirección, el programa de menú nos ofrece la opción de crear un nuevo proceso y se necesita que envíe un formulario con el tipo de proceso el nombre de la Tarea y su prioridad, se envió esa información al programa redirección este se encarga de registrar la tarea y de conectarse con el programa formulario, estas acciones se realizaron de forma correcta verificándose que la información guardada en el servidor era exacta a la ingresada.

Menú-Contenido

Existen varias opciones de espera de información, la primera es Mis Actividades, el menú envía como dato el tipo de listado y el programa de contenido responde bien presentando los valores que corresponden a todo el trabajo pendiente por mi. El segundo son Tareas Personales, el menú envía información que pide que el programa contenido ejecute la consulta para que devuelva el listado correspondiente, se comprueba que el listado emitido corresponde a la consulta hecha directamente al servidor. El Tercero son Tareas Iniciadas, en este espacio el menú envía el dato del tipo de listado y efectivamente se comprueba que el programa contenido ejecuta esta acción y el resultado es el requisito especificado por el usuario.

Redirección- Formulario

La integridad funcional entre estos dos programas se comprueba registrando Tareas. El programa redirección realiza el registro y envía los datos de Tarea, Proceso, Actividad y Pagina, con ello el programa de Formulario construye la interfaz para el usuario. Se realiza la prueba con distintos procesos y se observa el innecesario envío de Página y Proceso; se corrige y se realizan nuevamente pruebas que demuestran que la integridad y el resultado esperado es el correcto.

Formulario-Documentos

La integridad entre estos dos programas es de suma importancia, gracias a esto podemos ingresar nuestros documentos en la Tarea y Actividad correspondiente. El programa de formulario envía el código de identificación del programa relacionado con la actividad a realizar además envía, Tarea, Actividad, se realizan las pruebas mostrando un resultado favorable al momento de enviar todo tipo de actividad, y código de identificación del programa correspondiente.

Formulario-Decisiones

La integridad en la relación de estos dos programas se puede probar únicamente con actividades anteriores a una decisión. Se realiza las pruebas en la que el programa formulario envía la información de la Actividad, el proceso fue ejecutado mostrándonos las posibilidades o caminos que puede tomar este proceso, no existen inconvenientes.

Contenido-Formulario

El programa contenido tiene opciones de listado variadas, y en estos listados es que inicia la relación con el programa formularios. Contenido envía información de código de Ejecutadas, Tarea, Actividad y Página; en la pruebas se produjo un error ya que se realizaron correcciones en la pagina de formularios para que encuentre la pagina correspondiente a la actividad y los valores de construcción de interfaz fueron erróneos. Se realizaron las correcciones suprimiendo el envío del dato Página. Se testeó nuevamente y el resultado no produjo errores.

Formularios-Contenido

La integridad es la siguiente, formularios debe enviar información a ser guardada en la base de datos, para ello envía la información de la Tarea, Actividad y Valor de la Actividad en el caso de que sea decisión. Contenido verifica que toda la información sea correcta y registra automáticamente en la base de datos. Se realizan las pruebas enviando datos solo de Tareas, solo de Actividad, y solo

Valor, en los 3 casos el programa contenido al no tener la información completa muestra únicamente el listado con un mensaje de que el proceso no fue guardado.

Formularios-Cookies

El programa formularios se encarga de enviar una solicitud de información de actividad de cookies ValidarTarea, se realiza las pruebas ingresando un proceso y sus diferentes actividades. El programa Cookies responde de manera satisfactoria en todos los casos.

Contenido-Cookies

El programa contenido envía una solicitud de inclusión de script de cookies, inmediatamente al programa cookies se ejecuta y se verifica si las cookies están habilitadas. La situación planteada se realiza en cada acceso al software quedando demostrado con éxito el funcionamiento.

Motor

Actividades-ReglasPolíticas

El programa Actividades envía la Actividad y el programa Reglas y Políticas recibe para relacionar la actividad con una regla o política, se registra. Se realiza las pruebas en cada actividad, se verifica en la base de datos para buscar coherencia de la información. Las pruebas son satisfactorias.

Actividades-Decisiones

El programa Actividades envía la Actividad y el programa Decisiones lista la Información de decisiones. Se verifica con las actividades decisión y no se encuentran errores.

Rutas-Diagramas

El Programa Rutas envía información sobre el proceso a ser graficado al programa Diagramas. La comprobación la realizamos en el momento de ingresar a rutas, se realiza cada una de las rutas para cada proceso y se observa que el grafico se muestra en pantalla de forma satisfactoria.

6.4 Pruebas de Validación

Las pruebas de validación se centran en las acciones visibles del los usuarios (interacción) con el sistema. Tomaremos como referencia el diagrama Objeto Comportamiento para ahondar en las partes que se consideren críticas en el sistema.

Ingreso al Sistema

El usuario conoce la dirección del software entra en el Browser y automáticamente pide un nombre de usuario y una contraseña. En esta interacción el sistema requiere su autenticación y ofrece 3 opciones de ingreso. Se le proporcionó al usuario 5 nombres de usuario y 5 contraseñas, pudo ingresar sin ningún problema.

Elección de Nueva Tarea

Se pide al usuario que ingrese una nueva Tarea, el usuario busca el tipo de tarea pero no sabe que ingresar en la caja de texto NOMBRE, se indica que es información de reconocimiento sobre como desea identificar la Tarea, comprende y continua con la tarea. Elige la prioridad de la tarea y a continuación hace clic sobre la palabra Ingresar.

La confusión por el usuario se resuelve renombrando a la etiqueta de la caja de texto como Nombre de Tarea y además se incluye tooltips con frases explicativas.

Introducir Datos

El usuario observa que el sistema lo envía a una nueva pantalla en la que consta la información necesaria sobre la tarea que desea realizar. Se ingresa información, y la registra, intenta hacer clic sobre TAREA COMPLETA, el sistema responde que los documentos necesarios no fueron adjuntados. El usuario adjunta los documentos y procede a registrar la información nuevamente haciendo clic sobre TAREA COMPLETA. El sistema registra e informa al usuario que su trámite fue registrado.

Adjuntar Documentos

El usuario a elegido la tarea, introdujo la información necesario e intenta registrar haciendo clic en TAREA COMPLETA, el sistema exhibe un mensaje de alerta informando que debe adjuntar los documentos necesarios, el usuario se percata y se adjunta los documentos necesarios, el sistema le informa del avance. El usuario ha adjuntado los documentos y se dirige nuevamente a TAREA COMPLETA, el sistema muestra un mensaje de afirmación.

Introducción de Decisión

El usuario ingresa los datos necesarios en el sistema, adjunta los documentos, intenta registrar TAREA COMPLETA, el sistema muestra un mensaje indicando que la decisión no fue efectuada. El usuario revisa y elige la opción, nuevamente se dirige a hacer clic en TAREA COMPLETA el sistema muestra un mensaje que afirma la acción realizada por el usuario.

Elegir Tarea

El usuario observa la lista de Tareas que le corresponde ejecutar, elige una haciendo clic sobre cualquier parte de la fila de referencia. El sistema muestra la pantalla de introducción de información.

Las pruebas se realizan y en su mayoría el usuario comprueba que son funcionales y de fácil acceso, su comprensión esta en niveles aceptables pero se modifica y en algunos casos corrige los problemas mencionados en cada prueba, determinando con éxito la validación.

6.5 Pruebas de Desempeño

Las pruebas de desempeño sirven para llevar al máximo la carga en todo sentido sobre el sistema. Se empleó el software de la casa SoftLogica llamado WAPT (Web Application Testing), el cual emuló a 20 usuarios concurrentes en proceso normal de trabajo.

El sistema soportó trabajar con 20 usuarios concurrentes mostrando tener la capacidad de soportar mucho más ya que en ningún momento se cortó la conexión o dio mensajes de error, todo esto se encuentra en la sección de Anexos.

Ver Anexos 7: Prueba de Desempeño. Pág. 109

Ver Anexo 8: Cuadro Estadístico WAPT. Pág. 111

El siguiente paso fue probar los errores del sistema, tal es el caso de no digitar algún tipo de dato o tratar de registrar sin la información necesaria, o simplemente intentar ingresar en una tarea que se ejecuta en serie. En cada uno de los situaciones antes mencionadas el sistema pudo responder al error y retornar a la pantalla principal sin conflictos.

Continuamos con defectos que sabemos no se manifestarían por si solos, tal es el caso de pulsar un botón de función o botones con funciones específicas fuera de contexto. Como sabemos el sistema trabaja sobre Web es decir por medio de un explorador por lo tanto habrán ciertas opciones que ya tienen botones del teclado asignados a funciones específicas por lo cual se realizó la verificación al momento de pulsarlos. El usuario ingresa al sistema y comienza a pulsar los botones del teclado, el sistema no responde y continúa en su trabajo normal, exceptuado la tecla de función F5 para ello se colocó en los lugares críticos que si actualiza la página no será registrado o actualizada la información.

6.6 Pruebas de Aceptación

Después de realizar las pruebas con el usuario común expresaron que el sistema, de una u otra manera es muy sencillo de utilizar y de comprender (cabe recalcar que las sugerencias y problemas encontrados fueron corregidos antes de pedir una opinión), expresaron además que el requerimiento de información es muy aplicable a lo que necesitan y que de esta manera ningún empleado que sea removido de un puesto para enviarlo a otro, estaría con el problema de desconocer de su labor.

6.7 Pruebas de Instalación

Se instala en el servidor y se pide a un departamento que pruebe el sistema por un mes para de esta forma descartar cualquier problema. Las personas del departamento de Desarrollo Organizacional quedaron complacidas dando el visto bueno del sistema.

6.8 Conclusiones

Realizadas las pruebas correspondientes, comprobamos que el sistema pasó a satisfacción la mayoría de estas, y que los errores detectados, fueron corregidos, sin mayor complicación, además el uso de personas sin relación al presente trabajo, para que utilicen el sistema mostró que no hay mayor dificultad para su operabilidad, además el uso de un software de stress reafirmó, el soporte de este a un número de 20 usuarios concurrentes.

CAPITULO VII
INSTALACION Y MANTENIMIENTO

7 Instalación y Mantenimiento

7.1 Introducción

La instalación y mantenimiento, seguridad, y revisiones generales comprenden esta parte del trabajo, revisiones que tienen y pondrán en óptimas condiciones el ambiente de la aplicación.

7.2 Instalación en Servidor WEB

La instalación en el servidor Web es clave para que el sistema WorkFlow trabaje de forma eficiente, como es lógico el sistema se desarrolló en un equipo de prueba para luego proceder con este paso, una vez realizadas las pruebas necesarias, se realizaron las siguientes acciones.

Seguridad

Creación de una carpeta diferente a la default designada por apache, esto ayuda a que personas malintencionadas no conozcan de su ubicación y por lo tanto no puedan acceder al sistema.

Puertos

Se procedió a cerrar puertos que no necesita el sistema ni otros productos instalados en el mismo tal es el caso del puerto 25 que suele utilizarse para correo electrónico.

Router

Se procedió a crear una DMZ en el router con el fin de que el resto de servidores ocupen esta sección y no llegue al aplicativo.

Firewall.

Se crearon reglas para la restricción de usuarios común al servidor Web.

Subdominio

Se creó el subdominio intranet.azuay.gov.ec para el acceso al servidor desde la Internet.

7.3 Revisiones Finales

Una vez terminado el sistema se procedió junto con el jefe de sistemas encargado a las revisiones de todos los procesos mediante el uso del modelo objeto comportamiento.

Se procedió a revisar el servidor Web con el fin de que no interviniese o provocara conflicto con el resto de herramientas que se encuentran instaladas en la institución y cerciorándonos además que la carga en recursos de red no sea un conflicto por el hecho de alojar la página Web de la institución dentro del mismo hardware.

7.4 Conclusiones

Se logró una instalación limpia sin inconvenientes, y se mejoró sustancialmente la seguridad del servidor, mediante el uso de varias restricciones como cierre de puertos, reglas en el firewall y división de redes.

CAPITULO VIII
CONCLUSIONES Y RECOMENDACIONES

8 Conclusiones y Recomendaciones

A lo largo de los capítulos anteriores se mostró el sistema WorkFlow, que implementa procesos que intentan facilitar la vida con la generación de rutas preestablecidas para que el usuario disminuya los tiempos de ejecución e inducción en la entidad. En este capítulo se presentan los resultados obtenidos, las limitaciones que presenta el sistema y luego las recomendaciones, las cuales pueden ser tomadas como futuras líneas de investigación o trabajos de graduación.

8.1 Conclusiones

Basándonos en las definiciones conceptuales que se dieron en los primeros capítulos sobre tecnología workflow y lo que es groupware se consiguió crear una herramienta que en base a la definición de procesos y sus actividades da una flexibilidad y además permite una verificación que asegura que el flujo de información dentro de la entidad sea continuo.

El Software WorkFlow tiene unas ventajas sorprendentes al momento de su implementación, ya que cada director o jefe de sección puede establecer un nuevo proceso únicamente de una hoja de papel al sistema y esta listo.

Se estudiaron las características de los procesos que forman parte del Gobierno Provincial del Azuay, mediante este estudio se identificaron una serie de reglas y políticas necesarias, que de una u otra manera viabilizan a dichos procesos con la finalidad de satisfacerlas por medio de WorkFlow.

El sistema WorkFlow es una de la partes básicas para una arquitectura groupware o de trabajo colaborativo es por ello que debe trabajar de forma incondicional con un sistema de correo electrónico para abordar mas casos en los que existen personas que no se encuentran en la institución o que estando en la institución trabajan mas con este tipo de sistemas.

El mantenimiento es de vital importancia es por ello que la entidad que vaya utilizar WorkFlow deberá contar con personal calificado para poder desarrollar en conjunto con equipos de cada área su proceso o procesos necesarios para el trabajo diario.

Se establecieron diagramas de flujo de información para que estos puedan ser sometidos a una aprobación, estos diagramas han contemplado adaptabilidad en cuanto a criterios y sugerencias del usuario mismo, a su vez se utilizaron reglas de graficas en horizontal los que fueron repartidos a ciertos usuarios para verificar su comprensión.

En cuanto a limitaciones el sistema WorkFlow presenta lo siguiente:

- Al momento de querer integrarse con herramientas que no sean Web deberán crearse programas que puedan obtener la información de su base de datos para manipular dicha información.
- El sistema no acepta operaciones de múltiples instancias sin sincronización.
- El sistema no acepta operaciones de ciclos arbitrarios y elección en diferido.
- El sistema no muestra en tiempo real si una actividad es ingresada y su ejecución es para el usuario actual.

8.2 Recomendaciones

Si queremos introducir un sistema workflow debemos recordar antes realizar una reingeniería de procesos tanto operativos como administrativos. Además la implementación deberá ser realizada mediante apuntes que registren a los administradores y el tiempo que brindarán este servicio en la institución.

Ya que el sistema WorkFlow desarrollado en el presente trabajo en su primera validación ha conseguido un resultado impactante sobre los usuarios, se debe mejorar el prototipo, ya que no presenta todas las habilidades que un sistema en este campo posee.

A continuación presento puntos importantes que puedes ser futuros casos de estudio a resolver:

- La integración con herramientas que no sean Web
- Implementación de las operaciones de múltiples instancias sin sincronización
- Implementación de las operaciones de ciclos arbitrarios y de elección en diferido.
- El trabajo en tiempo real sobre actualizaciones en pantalla.
- Utilizar modelos de calidad para su perfeccionamiento.

BIBLIOGRAFIA

<http://www.wfmc.org>
WorkFlow Management Coalition
Jon Pyke, WfMC Chair
29/09/2006

Workflow Handbook 2006
BMP Business Process Management
Published in association with the Workflow Management Coalition (WfMC)
Edited by Layna Fischer

NEW TOOLS FOR NEW TIMES: THE WORKFLOW PARADIGM,
SECOND EDITION.
The Impact of Information Technology on Business Process Reengineering
Layna Fischer

Workflow Patterns
<http://is.tm.tue.nl/research/patterns/index.htm>
29/09/2006

Desarrollo WEB
<http://www.desarrolloweb.com/articulos/1380.php>
26/10/2006

ANEXOS

Anexo 1: Encuestas

ENCUESTA PERTENECIENTE AL ANALISIS, ESTUDIO Y DESARROLLO PARA UN "SISTEMA WORKFLOW PARA PROCESOS MANEJADOS DENTRO DEL GOBIERNO PROVINCIAL DEL AZUAY"
NOMBRE: DEPARTAMENTO: CARGO FECHA:
<p>Por Favor, Defina cuales son las funciones de su cargo:</p> <p>En que procesos interviene usted?</p> <p>Que función realiza en dichos procesos?</p> <p>Para la realización de procesos, Tiene usted algún tipo de regla, política o manual a seguir para realizar dichas funciones?</p> <p>Para las funciones que realice, Describa de quien depende para realizarlas.</p>

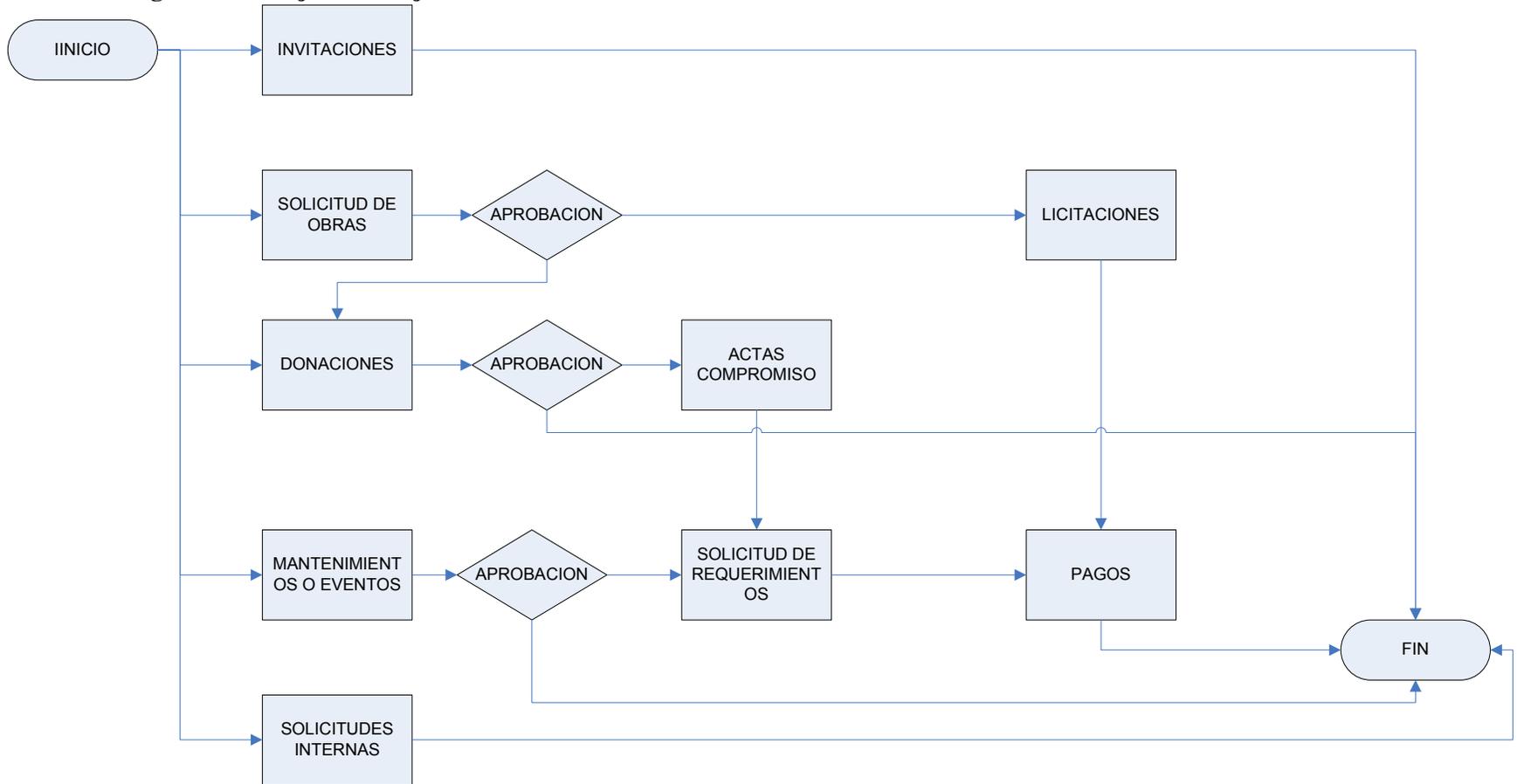
Anexo 2: Ficha de Procesos

Identificación:			
Objetivos:			
Responsable:			
Diagrama de Flujo			
Información Asociada			
Elemento:			
Descripción:			
Atributo	Tabla	Base	

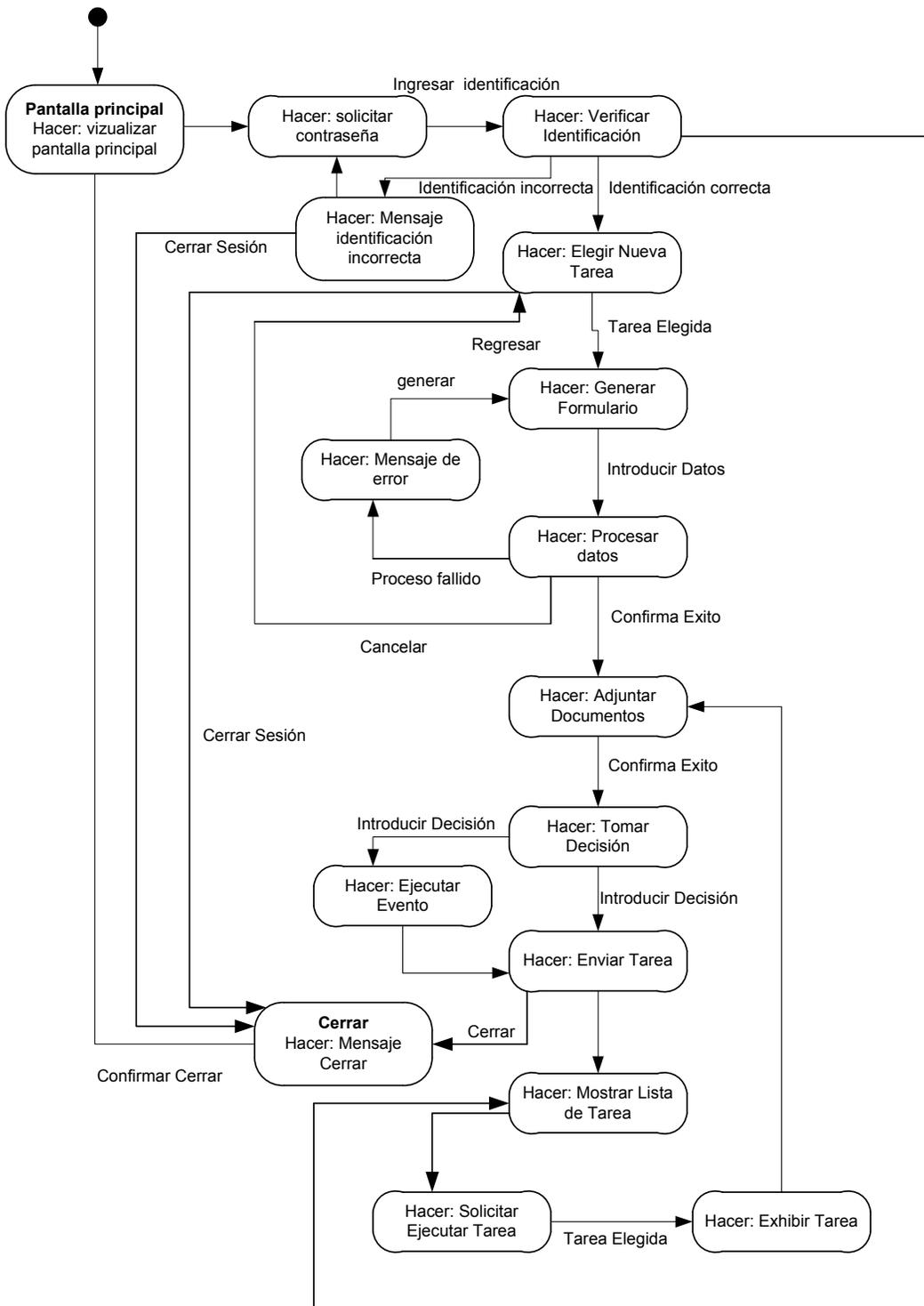
Anexo 3: Ficha de Actividades

Código:	
Descripción:	
Periodicidad:	
Roles:	
Información a Visualizar	
Módulos a Lanzar	
Aplicación:	
Identificación:	
Método de Llamada	
Parámetros	
Rutas	
Identificador	
Tipo	
Condición para enrutamiento automático	
Condición necesaria para poder enrutar	
Acciones a ejecutar al enrutar el proceso	
Estado Destinatario	

Anexo 4: Diagrama de Flujo del Conjunto de Procesos



Anexo 6: Modelo Objeto Comportamiento



Anexo 7: Pruebas de Desempeño

Resultados del Análisis con la herramienta WAPT 5.0

Test execution parameters:

Test status: finished

Test started at: 10/09/2007 14:42:32

Test finished at: 10/09/2007 14:44:07

Scenario name:

Test run comment:

Test executed by: Michael

Test executed on: LTMASTUDILLO

Test duration: 0:01:00

Virtual users: 20

Sumario

Profile	Sessions performed	Sessions with errors	Pages performed	Pages with errors	Hits performed	Hits with errors	Total KBytes sent	Total KBytes received
Perfill	1	0	553	0	963	0	405	3.245
Total	1	0	553	0	963	0	405	3.245

Número de Usuarios activos

Profile	0:00:00- 0:00:10	0:00:10- 0:00:20	0:00:20- 0:00:30	0:00:30- 0:00:40	0:00:40- 0:00:50	0:00:50- 0:01:00
Perfill	20	20	20	20	20	20
Total	20	20	20	20	20	20

Sesiones

Profile	0:00:00- 0:00:10	0:00:10- 0:00:20	0:00:20- 0:00:30	0:00:30- 0:00:40	0:00:40- 0:00:50	0:00:50- 0:01:00	Total
Perfill	0	0	0	0	0	1	1
Total	0	0	0	0	0	1	1

Sesiones por segundo

Profile	0:00:00- 0:00:10	0:00:10- 0:00:20	0:00:20- 0:00:30	0:00:30- 0:00:40	0:00:40- 0:00:50	0:00:50- 0:01:00	Total
Perfill	0	0	0	0	0	0,1	0,02
Total	0	0	0	0	0	0,1	0,02

Paginas

Profile	0:00:00- 0:00:10	0:00:10- 0:00:20	0:00:20- 0:00:30	0:00:30- 0:00:40	0:00:40- 0:00:50	0:00:50- 0:01:00	Total
Perfill	125	115	25	10	17	16	308
Total	125	115	25	10	17	16	308

Hits

Profile	0:00:00-0:00:10	0:00:10-0:00:20	0:00:20-0:00:30	0:00:30-0:00:40	0:00:40-0:00:50	0:00:50-0:01:00	Total
Perfil	321	259	37	31	38	32	718
Total	321	259	37	31	38	32	718

Páginas por segundo

Profile	0:00:00-0:00:10	0:00:10-0:00:20	0:00:20-0:00:30	0:00:30-0:00:40	0:00:40-0:00:50	0:00:50-0:01:00	Total
Perfil	12,5	11,5	2,5	1	1,7	1,6	5,13
Total	12,5	11,5	2,5	1	1,7	1,6	5,13

Hits por segundo

Profile	0:00:00-0:00:10	0:00:10-0:00:20	0:00:20-0:00:30	0:00:30-0:00:40	0:00:40-0:00:50	0:00:50-0:01:00	Total
Perfil	32,1	25,9	3,7	3,1	3,8	3,2	12,0
Total	32,1	25,9	3,7	3,1	3,8	3,2	12,0

Páginas con errores

Profile	0:00:00-0:00:10	0:00:10-0:00:20	0:00:20-0:00:30	0:00:30-0:00:40	0:00:40-0:00:50	0:00:50-0:01:00	Total
Perfil	0	0	0	0	0	0	0
Total	0	0	0	0	0	0	0

Hits con errores

Profile	0:00:00-0:00:10	0:00:10-0:00:20	0:00:20-0:00:30	0:00:30-0:00:40	0:00:40-0:00:50	0:00:50-0:01:00	Total
Perfil	0	0	0	0	0	0	0
Total	0	0	0	0	0	0	0

KBytes enviados

Profile	0:00:00-0:00:10	0:00:10-0:00:20	0:00:20-0:00:30	0:00:30-0:00:40	0:00:40-0:00:50	0:00:50-0:01:00	Total
Perfil	83,7	106	11,0	7,65	13,0	10,7	232
Total	83,7	106	11,0	7,65	13,0	10,7	232

KBytes recibidos

Profile	0:00:00-0:00:10	0:00:10-0:00:20	0:00:20-0:00:30	0:00:30-0:00:40	0:00:40-0:00:50	0:00:50-0:01:00	Total
Perfil	997	1.012	96,1	154	331	223	2.814
Total	997	1.012	96,1	154	331	223	2.814

Anexo 8: Cuadro Estadístico WAPT

