

**COMPONENTES DE UNA METODOLOGÍA PARA LA GESTIÓN DE
PROYECTOS DE SOFTWARE EN UNA EMPRESA DEL SECTOR DE
ALIMENTOS.**

CAMILO ZULUAGA GALLEGO.

**UNIVERSIDAD EAFIT
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS
MEDELLIN
2006**

PROYECTO DE GRADO

TITULO

**COMPONENTES DE UNA METODOLOGÍA PARA LA GESTIÓN DE
PROYECTOS DE SOFTWARE EN UNA EMPRESA DEL SECTOR DE
ALIMENTOS.**

AUTOR: CAMILO ZULUAGA G.

CÓDIGO: 200110078010

ASESOR: RAFAEL DAVID RINCON B.

**PROFESOR DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS
UNIVERSIDAD EAFIT**

ESCUELA DE INGENIERÍA

INGENIERÍA DE SISTEMAS

UNIVERSIDAD EAFIT

MEDELLÍN

2006

NOTA DE ACEPTACIÓN

Jurado

Jurado

Lista de Figuras

- Figura 6.1 Esquema para la selección de un Proceso Software
- Figura 6.2 Metodología para desarrollo de aplicativos a nivel interno
- Figura 6.3 Definir metodología del aplicativo
- Figura 6.4 Ciclo de la metodología para el proceso de desarrollo de software
- Figura 6.5 Modelo de Desarrollo a nivel interno, vista por procesos
- Figura 6.6 Subproceso Definir Necesidades a Implementar
- Figura 6.7 Ciclo para Definir necesidades a Implementar
- Figura 6.8 Subproceso Analizar y Definir Prioridades
- Figura 6.9 Análisis de Viabilidad y Esfuerzo
- Figura 6.10 Priorización y definición de cronograma
- Figura 6.11 Diseño de Arquitectura
- Figura 6.12 Subproceso Desarrollar y Realizar Pruebas
- Figura 6.13 Subproceso Validar y Entrega
- Figura 6.14 Metodología para la Adquisición de Software
- Figura 6.15 Metodología para la Adquisición de Software Genérico
- Figura 6.16 Metodología para Modificaciones
- Figura 6.17 Metodología para Modificaciones Externas
- Figura 6.18 Metodología para Consultas
- Figura C.1 Ejemplo Vista 1. Tabla ADAT_PERSONA
- Figura C.2 Ejemplo Vista 2. Tabla ADAT_PERSONA1
- Figura F.1 Registro por Aplicativo
- Figura E.1 Inventario de Aplicativos
- Figura E.1 Inventario de Aplicativos

Lista de Tablas

Tabla 6.1 Descripción de Actividades para Identificación y Reconocimiento General

Tabla 6.2 Descripción de Actividades para inicio y definición de proyecto

Tabla 6.3 Descripción de Actividades para definir Necesidades a Implementar

Tabla 6.4 Descripción de Actividades de Análisis de Viabilidad y Esfuerzo

Tabla 6.5 Descripción de Actividades para Priorización y definición de cronograma

Tabla 6.6 Descripción de Actividades para Diseño de Arquitectura

Tabla 6.7 Descripción de Actividades para Desarrollar y Realizar Pruebas

Tabla 6.8 Descripción de Actividades para Validar y Entregar

Tabla 6.9 Descripción de Actividades para la Adquisición de Software

Tabla 6.10 Descripción de Actividades para la Adquisición de Software Genérico

Tabla 6.11 Descripción de Actividades para Modificaciones

Tabla 6.12 Descripción de Actividades para Modificaciones Externas

Tabla 6.13 Descripción de Actividades para Consultas

Tabla 7.1 Descripción de Actividades para el Proceso de Soporte

Tabla C.1 Abreviaciones para los Tipos de Datos Básicos

Tabla C.2 Abreviaciones para el manejo de Variables

Tabla C.3 Abreviaciones para los Tipos de Datos Básicos para manejo con Constantes.

Tabla C.4 Abreviaciones para Controles de WebForms .NET y de Formularios VFP

Tabla C.5 Abreviaciones para el Manejo de Datos y Conexiones

Tabla C.6 Abreviaciones Para los Objetos de las Formas en VFP

AGRADECIMIENTOS

El autor expresa sus agradecimientos a:

La Dirección de Informática de la Compañía Nacional de Chocolates, especialmente a Hernán Darío Osorio, Jefe de Proceso Desarrollo Industrial por sus enseñanzas, dedicación y por permitir la realización de este proyecto; al equipo de personas conformado por Gabriel Oscar Arango, Jazmín Múnera, Juan David Tangarife y Federico Guzmán, por su constante colaboración.

A Rafael David Rincón, asesor del proyecto de grado, gracias por toda su constancia contribución y conocimiento.

A mis padres por darme el estudio y por su apoyo en todo momento.

A Todos, Muchas Gracias.

TABLA DE CONTENIDO

1. ANTECEDENTES DEL PROBLEMA	11
2. DEFINICIÓN DEL PROBLEMA	12
3. ALCANCE Y PRODUCTOS	13
4. OBJETIVOS	14
5. RESPONSABILIDADES	15
6. PROCESO SOFTWARE	16
6.1 ESQUEMA GENERAL PARA LA SELECCIÓN DE PROCESOS SOFTWARE	16
6.2 METODOLOGÍA PARA APPLICATIVOS	20
6.2.1 METODOLOGÍA PARA EL PROCESO DE DESARROLLO A NIVEL INTERNO	30
6.2.1.1 DEFINIR NECESIDADES A IMPLEMENTAR	34
6.2.1.2 ANALIZAR Y DEFINIR PRIORIDADES	44
6.2.1.3 DESARROLLAR Y REALIZAR PRUEBAS	57
6.2.1.4 VALIDAR Y ENTREGAR	62
6.2.2 METODOLOGÍA PARA EL PROCESO DE DESARROLLO A NIVEL EXTERNO	67
6.2.2.1 METODOLOGÍA PARA LA ADQUISICIÓN DE SOFTWARE	68
6.2.2.2 METODOLOGÍA PARA LA ADQUISICIÓN DE SOFTWARE	74
6.2.2.3 ACTIVIDADES RECOMENDADAS PARA REALIZAR SEGUIMIENTO DESPUÉS DE LA CONTRATACIÓN	81
6.3 METODOLOGÍA PARA MODIFICACIONES	88
6.3.1 METODOLOGÍA MODIFICACIONES INTERNAS	89
6.3.2 METODOLOGÍA MODIFICACIONES EXTERNAS	96
6.4 METODOLOGÍA PARA CONSULTAS	101
7. PROCESO DE SOPORTE	106
8. EJEMPLO PRÁCTICO	110
9. CONCLUSIONES Y RECOMENDACIONES	118
10. BIBLIOGRAFÍA	121
APÉNDICE A	123
A.1 DEFINICIÓN DE REQUISITOS	123
A.1.1 REQUISITOS FUNCIONALES	123

A.1.2 REQUISITOS NO FUNCIONALES	124
A.1.3 REQUISITOS DE INFORMACIÓN	124
A.2 EJEMPLOS DE PREGUNTAS PARA RECONOCER LAS NECESIDADES DE LAS PERSONAS SEGÚN EL TIPO DE REQUISITO.	125
A.2.1 RECONOCIMIENTO GENERAL DEL SISTEMA	125
A.2.2 REQUISITOS DE INFORMACIÓN	125
A.2.3 REQUISITOS FUNCIONALES	126
A.2.4 REQUISITOS NO FUNCIONALES	127
A.3 PREGUNTAS PARA LA IDENTIFICACIÓN DE ACTORES	127

APÉNDICE B **128**

B.1 DIAGRAMAS DE CASOS DE USO O ESCENARIOS	128
B.1.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR UN DIAGRAMA DE CASO DE USO O ESCENARIO	128
B.1.2 CASOS DE USO	128
B.1.3 ESCENARIOS	129
B.1.4 ENTREGABLES	129
B.2 MODELOS DE DATOS	131
B.2.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR UN MODELO DE DATOS	131
B.2.2 ENTREGABLES	132
B.3 DOCUMENTACIÓN DE LA DISTRIBUCIÓN.	133
B.3.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR UN MODELO DE DATOS	134
B.3.2 ENTREGABLES	135
B.4 DIAGRAMA DE DISTRIBUCIÓN (A CRITERIO DEL ANALISTA)	136
B.4.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR UN DIAGRAMA DE DISTRIBUCIÓN	136
B.4.2 ENTREGABLES	137

APÉNDICE C **138**

C.1 ESTÁNDARES PARA CODIFICACIÓN Y BASES DE DATOS	138
C.2 ESTÁNDARES PARA BASES DE DATOS	139
C.2.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR LOS DIFERENTES NOMBRAMIENTOS	139
C.2.2 REGLAS PARA EL NOMBRAMIENTO DE LOS OBJETOS DE LA BASE DE DATOS	139
C.2.3 DEFINICIÓN DE TABLAS Y CAMPOS	140
C.2.4 DEFINICIÓN PARA LAS TABLAS	140
C.2.5 DEFINICIÓN PARA LOS CAMPOS	141
C.2.6 RESTRICCIONES	143
C.2.7 DEFINICIÓN DE PROCEDIMIENTOS, TRIGGERS, FUNCIONES Y VISTAS	144
C.2.8 DOCUMENTACIÓN EN LAS BASES DE DATOS	150
C.2.9 VENTAJAS	153
C.3 ESTÁNDARES PARA LA CODIFICACIÓN	154
C.3.1 INDENTACIÓN	155
C.3.2 LÍNEAS Y ESPACIOS EN BLANCO	155

C.3.3 ESTRUCTURAS DE CONTROL	156
C.3.4 CONVENCIONES PARA EL NOMBRAMIENTO DE VARIABLES, OBJETOS Y CONTROLES	158
C.3.5 DEFINICIÓN DE FUNCIONES	167
C.3.6. DEFINICIÓN DE CLASES	168
C.3.7 LLAMADO A FUNCIONES	169
C.3.8 DOCUMENTACIÓN DEL CÓDIGO (NECESARIO)	170
<u>APÉNDICE D</u>	<u>172</u>
<u>ÍNDICE</u>	<u>173</u>
<u>APÉNDICE E</u>	<u>192</u>
<u>APÉNDICE F</u>	<u>194</u>

1. ANTECEDENTES DEL PROBLEMA

Es importante para las compañías crear y definir metodologías de desarrollo de software en sus departamentos de informática, porque en algunos casos, los aplicativos que se realizan tiene algún procedimiento o serie de pasos definido, y mucho menos se crea la documentación necesaria, pero muchas de estas metodologías no pueden ser aplicadas completamente para todo tipo de proyectos debido a que no serían prácticas. Por esta razón y encontrando la oportunidad de complementar la metodología de desarrollo para el proceso software en una empresa del sector de alimentos, se propone desarrollar una metodología en la cual se documenten procesos y se creen elementos y componentes que permitan una mejor gestión de los proyectos.

Este proyecto complementa el proyecto de grado “Definición de una metodología de desarrollo para el proceso software en una empresa del sector de alimentos”, del Ingeniero de Sistemas, Federico Guzmán.

2. DEFINICIÓN DEL PROBLEMA

Ante la necesidad de las empresas de trabajar de forma ordenada y de estar preparadas para las nuevas tendencias administrativas de gestión por procesos, en donde todas las actividades deben estar definidas y ser medibles, es necesario desarrollar metodologías y estándares de trabajo que se adapten a las nuevas tendencias y en donde las personas puedan saber qué es lo que se debe realizar en cada una de las actividades de los diferentes procesos.

Las áreas de informática de las compañías se convierten en un apoyo fundamental para la ejecución de los procesos del negocio, sean estos estratégicos, clave o de soporte. Al no ser el desarrollo de software el fin último de estas áreas de informática pero sí una de sus funciones, y al no tener una metodología establecida para el cumplimiento de tales funciones, se requiere de una metodología sumamente ágil para el desarrollo de aplicaciones que les permita ser ordenados en esta labor y a la vez más eficientes.

Resulta también importante para las compañías que las personas trabajen bajo un mismo esquema, de modo que los proyectos se puedan adaptar y sea más fácil para las personas que ingresan e incluso para aquellas que ya trabajan allí, realizarlos; para esto es importante que la metodología que se desarrolle establezca etapas para la documentación de las actividades que se deben ejecutar en los proyectos.

En este proyecto de grado se pretende generar una metodología de trabajo que sea ágil y práctica para la gestión y el desarrollo de aplicaciones, ya sea a nivel interno o externo, en empresas para las cuales el desarrollo de software no sea su objetivo de negocio y que a la vez, las actividades que se realizan en los distintos procesos puedan ser documentadas para que puedan ser consultadas en cualquier momento.

3. ALCANCE Y PRODUCTOS

Al finalizar el proyecto se espera que todas las personas comprendan las ventajas de los nuevos esquemas de trabajo para la gestión de las aplicaciones del área, desde su inicio en la elaboración de los requisitos hasta el final en el nombramiento, documentación, ubicación y organización de las aplicaciones.

En lo referente al desarrollo de la metodología (producto), se espera darle una documentación apropiada a las aplicaciones, tanto a las desarrolladas a nivel interno como a las desarrolladas por los distintos proveedores, obteniendo métodos que puedan generar una serie de políticas de manejo y negociación con los usuarios y proveedores, para que así se garantice la generación de la información necesaria para el área de informática y se desarrollen aplicaciones que cumplan con las expectativas de los usuarios.

Los productos que se obtendrán serán, una metodología la cual va a ser de gran utilidad y practicidad para una empresa del sector de alimentos, con la cual se generará una serie de entregables en las actividades de los diferentes procesos, entre ellos tenemos:

- Plantillas
- Procedimientos
- Formatos
- Estándares

Además de documentos, también habrá mejoras en los siguientes aspectos

- Pautas para desarrollo
- Políticas para manejo de usuarios y proveedores

La metodología será probada con el desarrollo de una aplicación de la compañía.

4. OBJETIVOS

4.1 Objetivo General

Documentar e implementar procesos de calidad, los cuales abarquen la especificación de estándares para el área de desarrollo del departamento de informática de una compañía del sector de alimentos y que permitan la elaboración de forma ágil, de actividades como levantamiento de requisitos, análisis, codificación y documentación que se deban realizar.

4.2 Objetivos Específicos

- Estandarizar una metodología de desarrollo con el fin de que todas las personas puedan trabajar bajo un mismo proceso de forma ordenada.
- Documentar las diferentes etapas de las aplicaciones que se realizan, para que cuando se requiera información acerca de cualquier aplicación se pueda obtener fácilmente.
- Construir un proceso de requisitos en el que se pueda avanzar de manera rápida, pero que a la vez se pueda recolectar toda la información necesaria para la elaboración de las aplicaciones, permitiendo además una trazabilidad en todas las actividades que se realizan.
- Realizar un control de versiones de todas las aplicaciones de modo que se pueda conocer cuáles han sido los cambios que han sufrido y por qué se han realizado.
- Evaluar herramientas que puedan dar apoyo a los procesos que se realizarán en las diferentes etapas.
- Divulgar la metodología a través de una presentación y también generando un documento el cual pueda ser consultado por el grupo de informática de la Compañía.

5. RESPONSABILIDADES

5.1 Jefe De Desarrollo

- Verificar que el desarrollo de los nuevos aplicativos sean debidamente documentados.
- Estar al tanto de las conversaciones y adelantos que se realizan en los desarrollos de los aplicativos.

5.2 Analista

- Resolver peticiones de los usuarios.
- Analizar, evaluar y negociar las necesidades de los usuarios.

5.3 Usuario Líder

- Es la persona que realiza la petición del nuevo desarrollo a la dirección de informática.
- Permanecer al tanto de los desarrollos y hacer pruebas para validar que se está cumpliendo con sus necesidades.
- Está encargado de especificar las necesidades del aplicativo.

5.4 Usuario

- Participa en la elaboración del aplicativo y puede especificar las necesidades y funcionalidades del aplicativo.

5.5 Equipo de Trabajo

- Verificar que el proyecto cumpla con los todos requisitos de manera adecuada
- Determinar las condiciones contractuales
- Hacer seguimiento a los proveedores de software

6. PROCESO SOFTWARE

6.1 Esquema General Para la Selección de Procesos Software

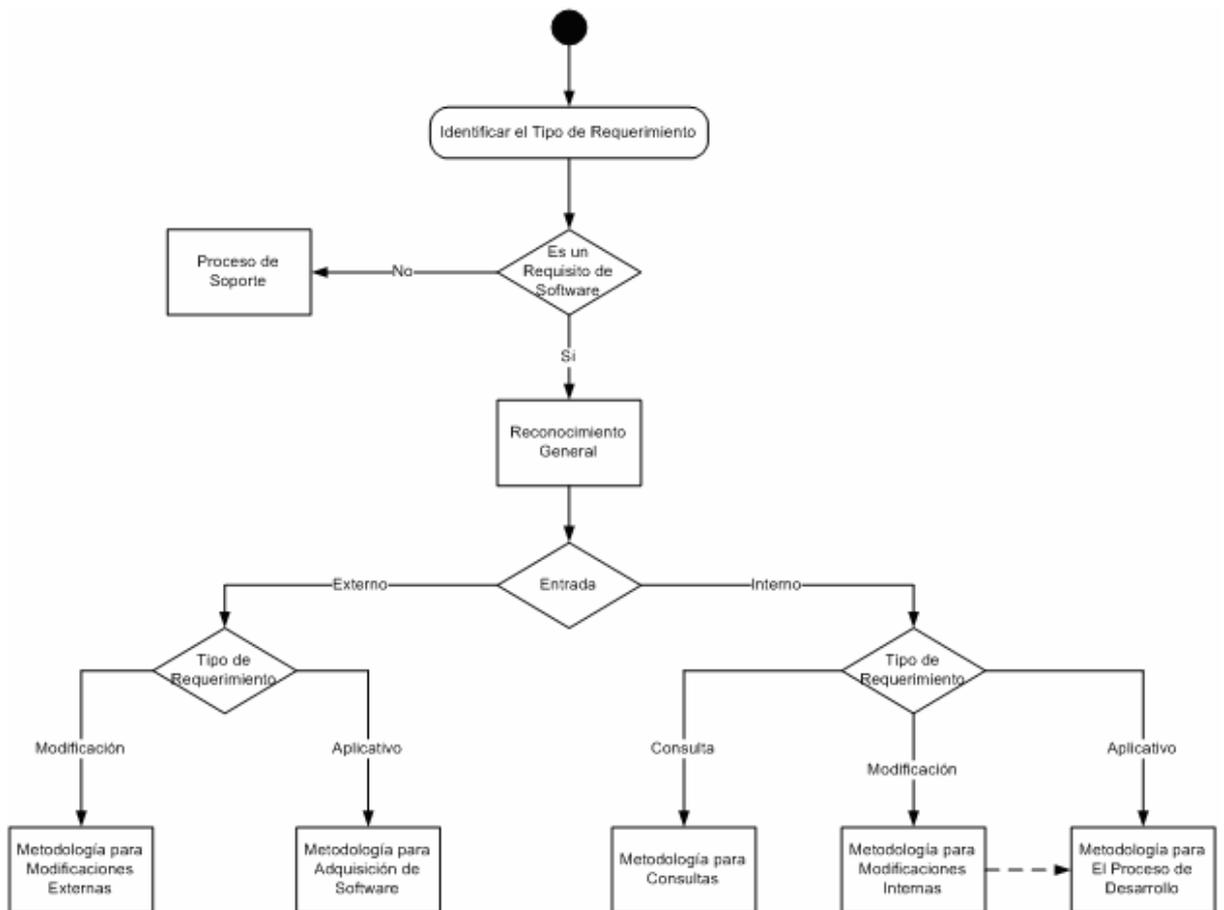


Figura 6.1 Esquema para la selección de un Proceso Software

Objetivos

- Identificar el requerimiento que ingresa a la Dirección de Informática.
- Obtener criterio para la selección de uno de los diferentes procesos de trabajo de software, dependiendo de las características del requerimiento que ingresó a la Dirección de Informática.

- Definir el proceso de software con el que se trabajará

Descripción

Después de realizar una serie de reuniones en 2 compañías del sector de alimentos en donde el objetivo era conocer la forma de trabajo y el estado de estandarización en los diferentes procesos de software que tenían, se obtuvo como resultado que no existía una metodología de trabajo definida, pero sí se encontraron similitudes en los tipos de problemas que deben resolver cotidianamente en las Direcciones de Informática de cada empresa.

La figura 6.1 muestra un esquema con los diferentes problemas que se deben resolver en una Dirección de Informática, los cuales se resumen de la siguiente manera:

A nivel Interno

- Aplicativos
- Modificaciones
- Consultas

A nivel Externo

- Aplicativos
- Modificaciones

En el momento en que se reconoce que es un requerimiento de software, éste pasa a “*Reconocimiento General*” para que se defina la metodología con la cual se va a trabajar.

De acuerdo con el esquema mostrado en la figura 6.1, se mostrará la forma en que se trabajan y se atienden los diferentes tipos de requerimientos que ingresan a una Dirección de Informática y a la vez cómo son documentados de forma apropiada.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Identificar el Tipo de Requerimiento		
Analista	Identificar el Tipo de Requerimiento	En el momento en que ingresa un requerimiento debe identificarse si es un requerimiento técnico o alguna otra actividad que no requiere desarrollo; en estos casos se trabaja con el Proceso de Soporte. Si hay que realizar algún tipo de desarrollo debe trabajarse con el Proceso de Software
Reconocimiento General		
Analista	Reconocer el requerimiento de Software	Con el fin de usar una metodología apropiada de trabajo, se debe verificar si el requerimiento es: A nivel Interno <ul style="list-style-type: none"> • Aplicativos • Modificaciones • Consultas

		A nivel Externo <ul style="list-style-type: none"> • Aplicativos • Modificaciones
--	--	---

Tabla 6.1 Descripción de Actividades para Identificación y Reconocimiento General

Entradas

- Requisito, Petición de trabajo para la Dirección de Informática.
- Información básica de la persona (Nombre, área a la cual pertenece) que realiza la petición.

Salidas

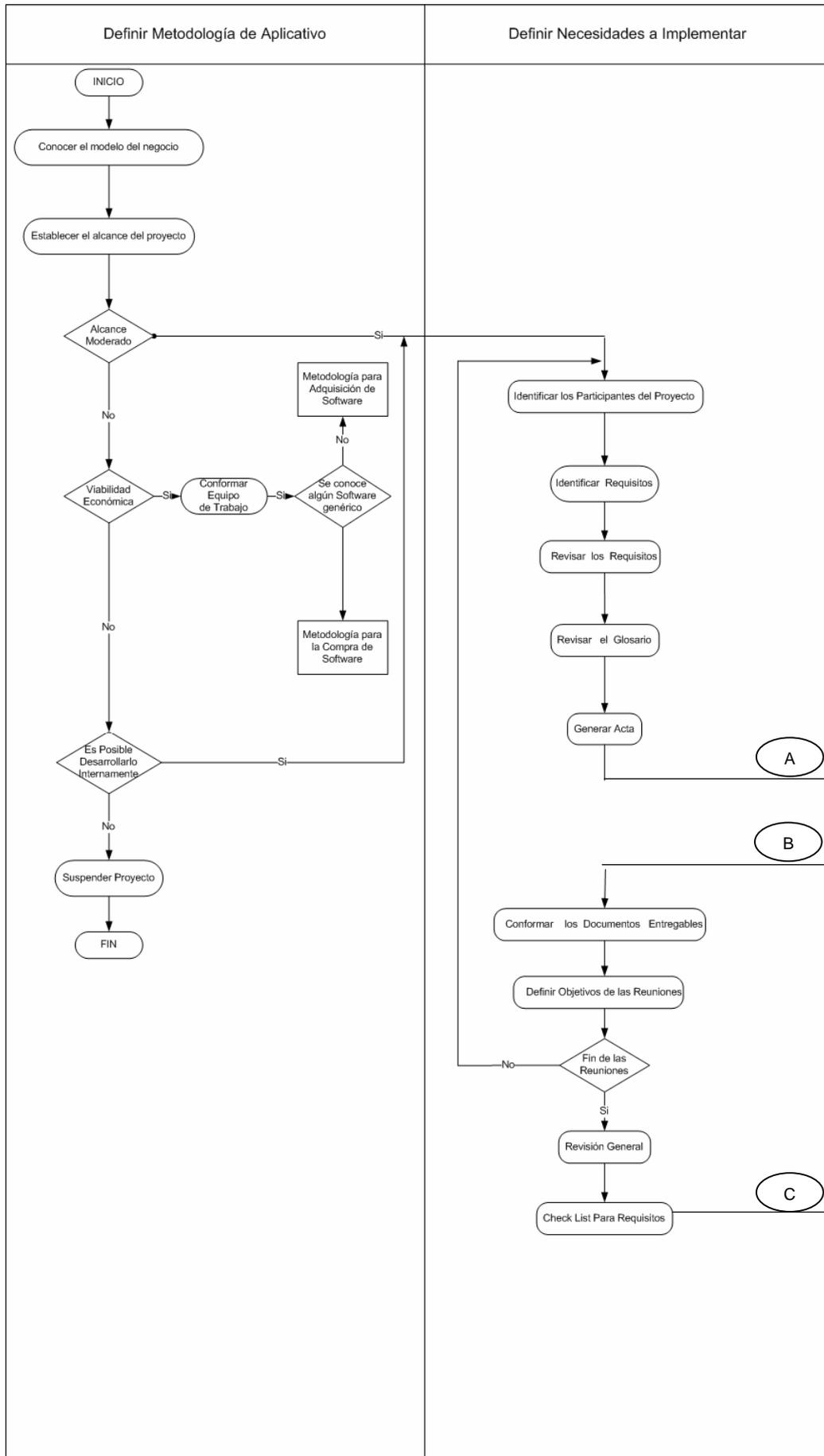
- Selección de Proceso con el que se trabajará
 - Proceso de Software
 - Proceso de Soporte
- Registro de las actividades realizadas en cualquiera de los diferentes procesos.

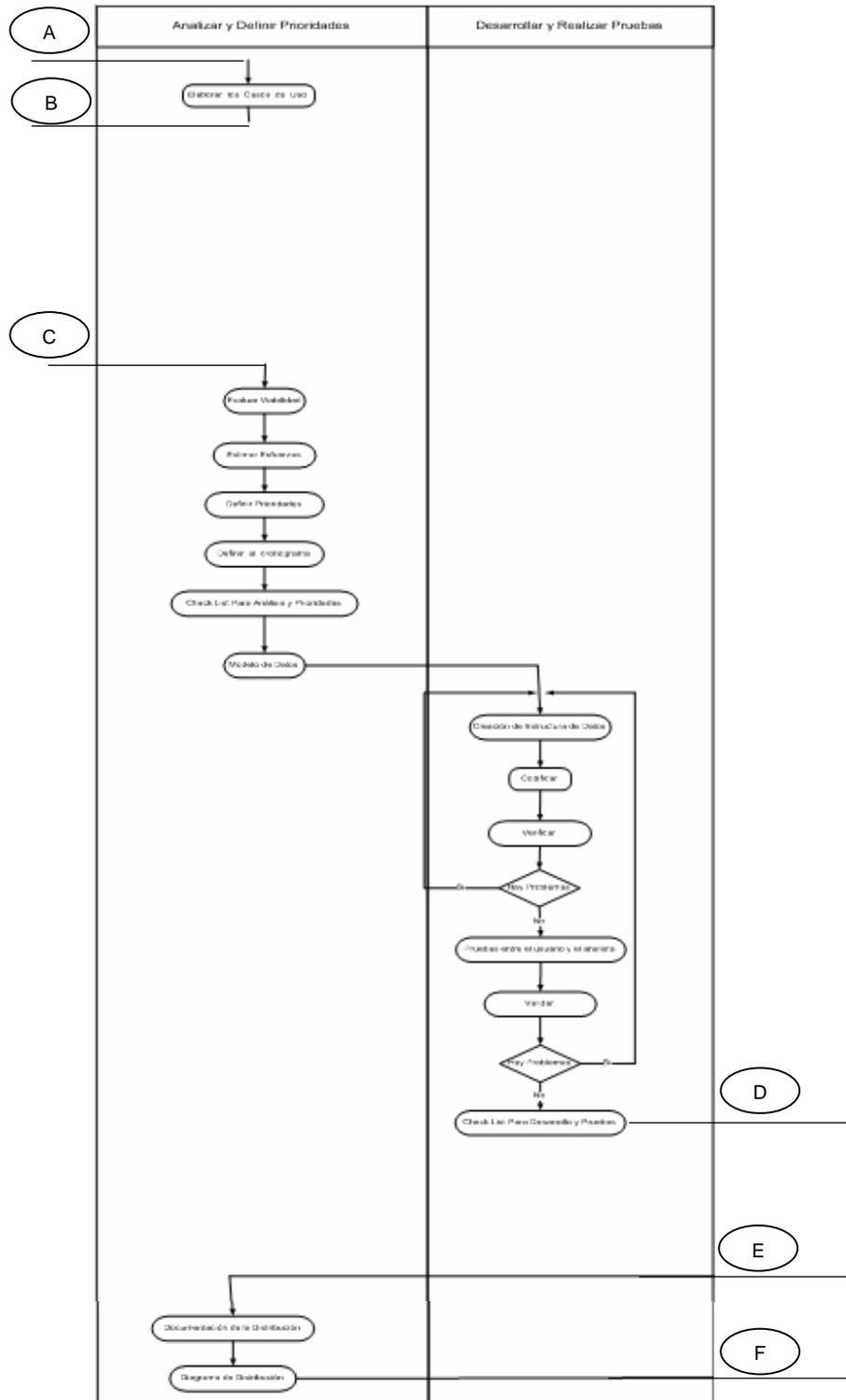
6.2 Metodología para Aplicativos

El desarrollo de nuevos aplicativos se divide en dos grandes metodologías, una para trabajo a nivel interno y la otra para trabajo a nivel externo, todo esto depende del alcance que tenga el proyecto. Si el alcance es alto (Requiere un alto uso de recursos, su ejecución implica demasiado tiempo y se requiere de personal y herramientas especializadas), se utilizará la metodología de desarrollo a nivel externo, pero si el alcance es moderado (Se requieren pocas funcionalidades y es un proyecto que se puede hacer en un tiempo no superior a cuatro meses) se usará la metodología para desarrollo a nivel interno.

Para la selección de alguna de las 2 metodologías existe un proceso en común que se muestra en la figura 6.3 el cual busca determinar el alcance del aplicativo y de esta forma conocer con cuál metodología se trabajará.

La metodología completa con la secuencia que se debe seguir para el desarrollo de nuevos aplicativos a nivel interno, se muestra en la figura 6.2; para realizar la explicación de la metodología, se explicará cada uno de los subprocesos que la componen de forma individual, teniendo en cuenta cada una de las etapas en las que se divide.





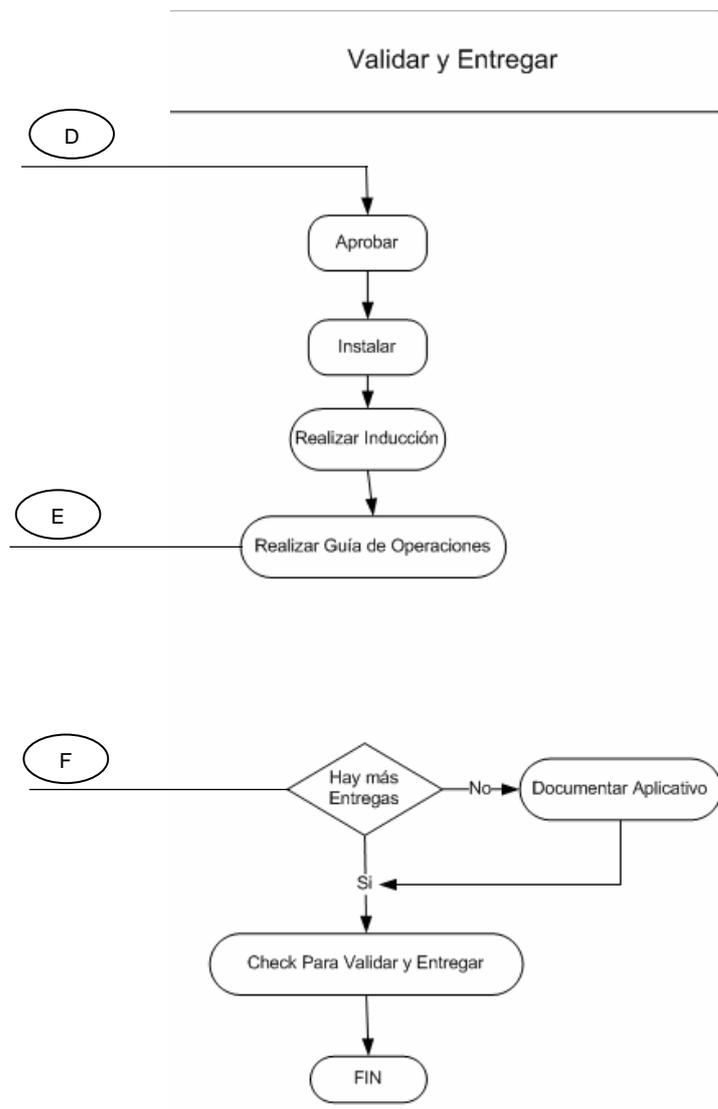


Figura 6.2 Metodología para desarrollo de aplicativos a nivel interno

Definir Metodología del Aplicativo

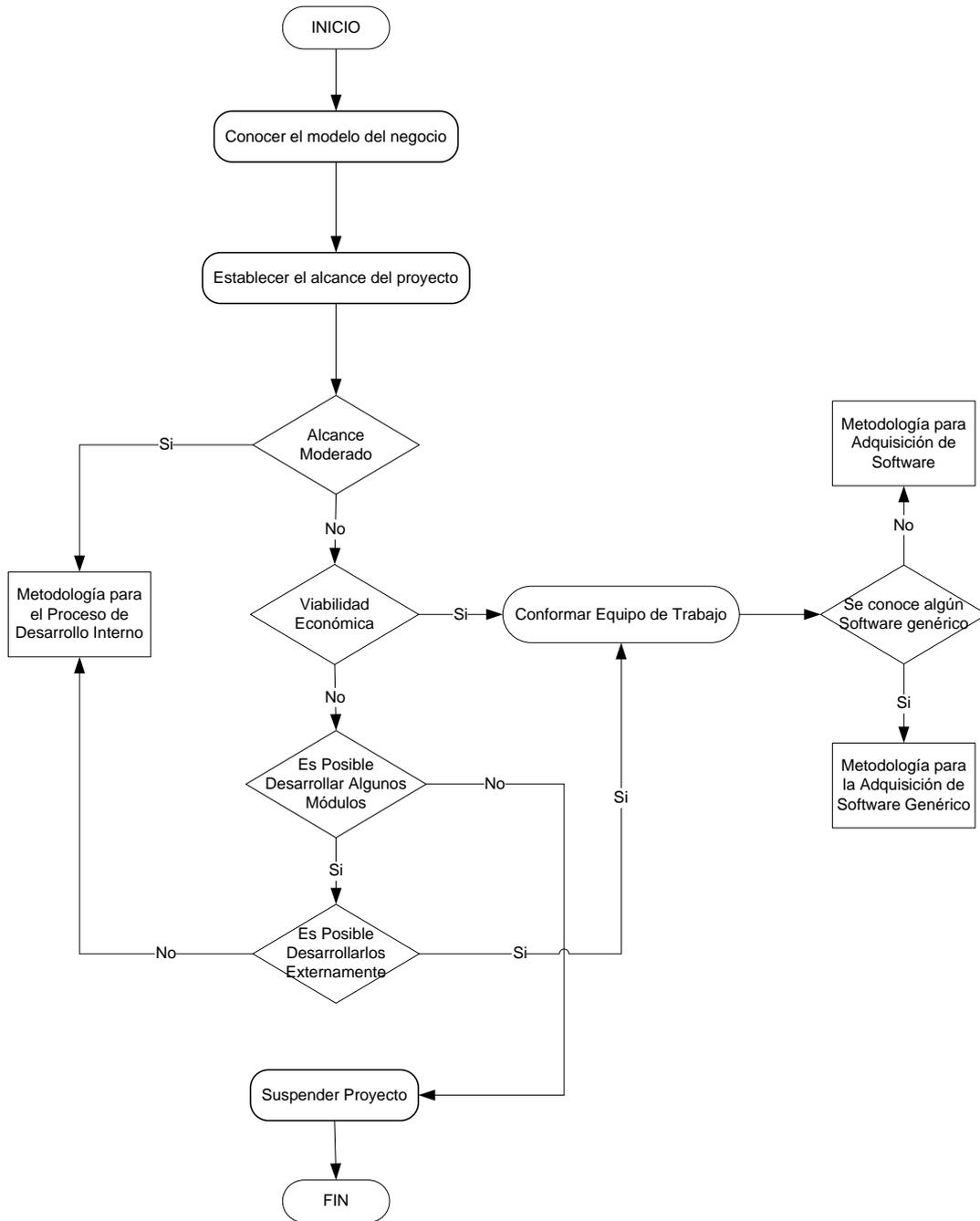


Figura 6.3 Definir metodología del aplicativo

Objetivos

- Conocer la forma como trabaja el área a la que se le realizará el proyecto.
- Conocer el alcance del proyecto.

Descripción

El esquema de la figura 6.3 busca conocer el modelo del negocio o los flujos de procesos y actividades que se realizan en el área a la cual se le va a realizar el desarrollo, además de que se obtiene el alcance del proyecto con el cual se puede establecer si el proyecto se puede realizar a nivel interno o si por el contrario, se debe contratar un proveedor para ejecutar el proyecto.

Al terminar este proceso se debe tener claro con qué metodología se va a trabajar o si el proyecto pasa a un estado de suspensión, debido a que su alcance es alto y a la vez por la falta de recursos económicos o de tiempo del área que quiere llevar a cabo el proyecto, lo cual no permite que se pueda realizar su contratación y ejecución con un tercero.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Conocer el Modelo del Negocio		
Analista	Introducción	Debe contener una descripción breve de las principales características del software que se va a desarrollar, la situación actual que genera la necesidad

		<p>del nuevo desarrollo, la problemática que se quiere resolver y cualquier otra consideración que sitúe a la persona que lea el documento acerca de lo que se va a resolver. La documentación se debe realizar en la plantilla de REM (Apéndice D).</p>
Analista	Descripción del Modelo del Negocio	<p>Se debe observar y documentar cuáles son las actividades que realizan las personas y cómo estas actividades influyen en el proceso.</p> <p>También se debe buscar la manera de mejorar los procesos, analizando sus fortalezas y debilidades.</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D).</p>
Establecer el Alcance del Proyecto		
Usuario Líder	Objetivo General	<p>Se debe especificar de forma clara y concisa qué es lo que se espera solucionar cuando la aplicación esté terminada.</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D)</p>

<p>Analista y Usuario Líder</p>	<p>Objetivos Específicos</p>	<p>Debe contener una lista con los objetivos que se espera alcanzar cuando la aplicación esté terminada.</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D)</p>
<p>Conformar Equipo de Trabajo</p>		
<p>Dirección de Informática y Área que pide el aplicativo</p>	<p>Conformación de Equipo de Trabajo</p>	<p>El área interesada en la adquisición del aplicativo y la Dirección de Informática deben conformar un equipo de trabajo para garantizar que el aplicativo cumpla con lo que se requiere.</p> <p>Por lo menos deben ser identificados los principales responsables de cada área de interés (Stakeholders), que son el usuario Líder (área que pide el aplicativo) y el analista (Dirección de Informática).</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D).</p>
<p>Equipo de Trabajo</p>	<p>Definir Roles</p>	<p>Deben ser definidos los roles de los integrantes del equipo de trabajo; algunos de los que se debe definir, son:</p> <p>Usuario Líder:</p>

		<p>Es la persona que realiza la petición de la aplicación y será la responsable de brindar la información, las necesidades y funcionalidades para el nuevo desarrollo.</p> <p>Analista:</p> <p>Es la persona que hace el levantamiento de los requisitos y/o asesora a los usuarios en la parte técnica.</p> <p>Rol Adicional:</p> <p>Cualquier otra persona que cumpla un papel distinto a los ya definidos en los roles anteriores.</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D)</p>
Suspender Proyecto		
Equipo de Trabajo	Suspender Proyecto	<p>Se llega a este estado por la imposibilidad económica del área interesada para que se lleve a cabo el proyecto o por la falta de los recursos necesarios por parte de la Dirección de Informática para realizar el proyecto.</p> <p>El proyecto luego puede ser retomado desde el punto en el</p>

		cual se suspendió.
--	--	--------------------

Tabla 6.2 Descripción de Actividades para inicio y definición de proyecto

Entrada

- Requisito o petición de Software. (Todavía no se ha definido cómo se trabajará o cuál será la metodología a seguir).

Salida

- Descripción del Modelo del Negocio del área a la que se le realiza el proyecto.
- Introducción → (Descripción breve de lo que se va a realizar y por qué se va a realizar)
- Alcance del Proyecto
 - Objetivo General
 - Objetivo Específico
- Decisión de cómo se trabajará en el proyecto
 - A nivel Interno
 - A nivel Externo
 - Software a la medida
 - Software Genérico
 - Equipo de Trabajo

6.2.1 Metodología para el Proceso de desarrollo A Nivel Interno

La definición de la metodología para el desarrollo de aplicativos a nivel interno se realizó con un grupo de analistas, con los cuales se pudo llegar a un acuerdo, estableciendo una forma común de trabajo para todos, que se rige por los parámetros y lineamientos de la Dirección de Informática de una Compañía del sector de alimentos y por las mejores prácticas de trabajo. La Dirección de Informática de la compañía tiene tres áreas que se ocupan de dar soporte a todos los aplicativos, ya sean desarrollados interna o externamente. De acuerdo con esta distribución, se seleccionó a un analista por cada área, de modo que la definición de la metodología pudiera enmarcarse de manera adecuada y fuera práctica y usable para cada una de las áreas de soporte; todo el trabajo del grupo fue liderado y coordinado por el jefe de una de las tres áreas.

De igual manera que para la definición de la metodología de trabajo, se definieron los formatos de documentación y las recomendaciones para la codificación y bases de datos.

El siguiente modelo muestra cada una de las diferentes etapas por iteración en las que se divide el proceso de desarrollo a Nivel Interno.

Vista del Modelo para el Proceso de Desarrollo A Nivel Interno

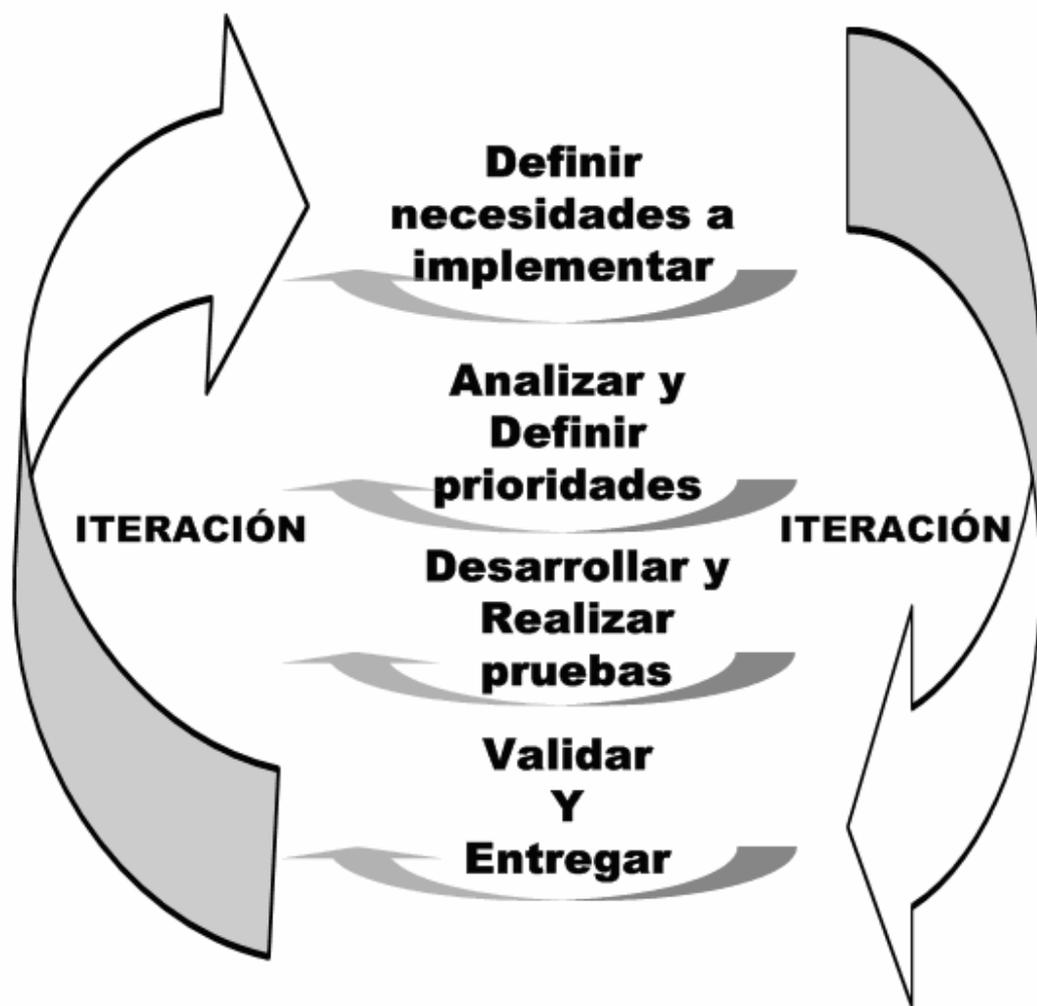


Figura 6.4 Ciclo de la metodología para el proceso de desarrollo de software¹

¹ Guzmán Vásquez, Federico. Definición de una metodología de desarrollo para el proceso software en una empresa del sector de alimentos. PROYECTO DE GRADO Universidad EAFIT, Junio 2006

Vista General del Modelo de Desarrollo a nivel Interno

Otra forma de observar el modelo para el proceso de desarrollo y que oriente de acuerdo con una vista por procesos, es la siguiente:

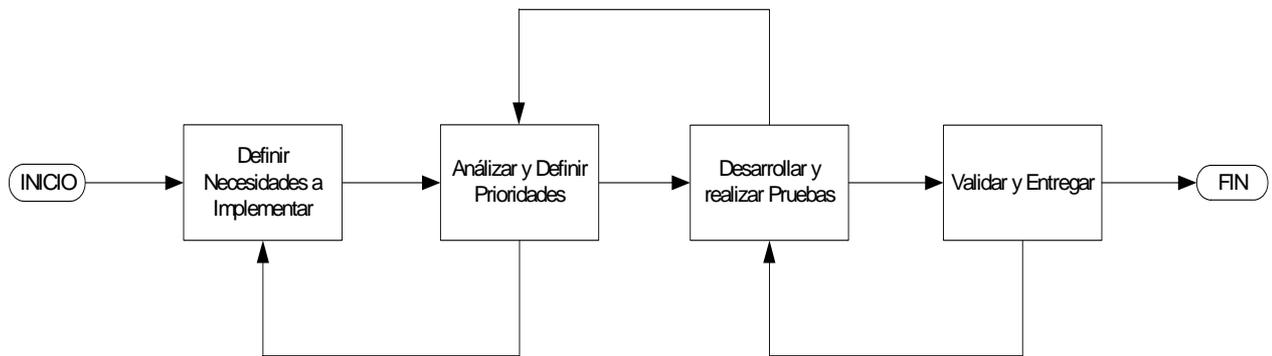


Figura 6.5 Modelo de Desarrollo a nivel interno, vista por procesos

Objetivos

- Realizar un proceso completo y ágil en el que se pueda alcanzar rápidamente las necesidades de los usuarios, pero que a la vez los aplicativos cumplan con los criterios de calidad y documentación que requiere la Dirección de Informática.

Descripción

De acuerdo con la forma como trabaja la Dirección de Informática, el modelo busca versatilidad y facilidad para moverse a través de los diferentes procesos en cualquier momento, además busca que los proyectos se dividan en ciclos, de modo que se puedan pactar por entregas, según la urgencia y necesidad de los usuarios y personas, logrando que se entreguen las necesidades más urgentes primero, pero garantizando por cada entrega, que se realice cada uno de los procesos del modelo de desarrollo interno.

La división del modelo en 4 subprocesos busca que se pueda segmentar el modelo en los principales frentes de trabajo que se tienen en los proyectos informáticos, de modo tal que se puedan trabajar los diferentes aspectos del proyecto de forma ordenada y orientada especialmente a cada uno de los temas que se trabajará.

Entrada

- Descripción del Modelo del Negocio del área a la que se le realiza el proyecto.
- Alcance del Proyecto
 - Objetivo General
 - Objetivo Específico

Salida

- Entrega de producto/servicio por iteración

A continuación se muestra y explica cada una de las etapas del modelo, de acuerdo con las actividades que se deben realizar.

6.2.1.1 Definir Necesidades a Implementar

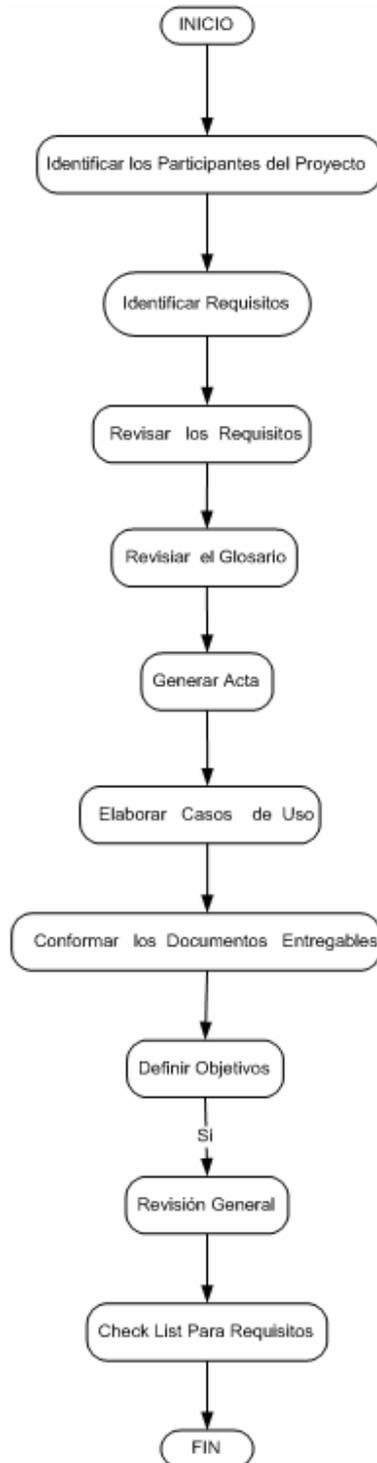


Figura 6.6 Subproceso Definir Necesidades a Implementar

Objetivos

- Obtener un documento de requisitos que contenga la toda la información necesaria para la entrega.
- Tener una visión clara del funcionamiento del sistema.

Descripción

En el momento en que se comienza esta etapa ya se conoce el alcance y el modelo del negocio, por esta razón se comienza con la conformación e identificación de los participantes en el proyecto.

Para llevar a cabo la etapa se debe tener en cuenta que para lograrla hay que realizar varias reuniones en las cuales se va conformando el documento sobre el cual se trabajará en las etapas siguientes.

Los distintos tipos de requisitos que se deben documentar, son:

- Requisitos de Información
- Requisitos de Restricción
- Requisitos Funcionales
- Requisitos No Funcionales.

Para los pasos expresados anteriormente, la metodología seguirá la siguiente secuencia:

- Elicitación.
- Análisis
 - Se hallan los posibles conflictos
 - Se buscan nuevas propuestas

- Se organiza lo elicidado para documentarlo
- Documentación

Elicitación:

La elicitación se realiza en las reuniones y el entrevistador registra toda la información que describe el usuario sobre lo que debe hacer la aplicación; esta información debe hacer parte del documento entregable que se realiza en REM.

Análisis:

Después de realizar la entrevista se deben tener en cuenta los posibles conflictos y dificultades que existen, estos problemas se pueden dar porque entre los requisitos se pueden generar contradicciones.

También se puede dar la posibilidad que durante el análisis se generen nuevas ideas que simplifiquen los requisitos y desarrollar nuevas funcionalidades.

Documentación:

Luego de realizar el análisis, se describen los conflictos, dudas o los problemas que se encontraron con los requisitos para que en la próxima reunión se le haga saber a los usuarios. Luego se realizan los casos de uso de las funcionalidades que debe tener el sistema para que en la próxima reunión se validen.

De acuerdo con la documentación que se realizó, se pueden generar los objetivos para la siguiente entrevista.

Gráficamente la propuesta funciona como se muestra en la figura 6.7, de modo que cumpla un ciclo en el orden especificado.



Figura 6.7 Ciclo para Definir necesidades a Implementar

La figura 6.7 muestra la secuencia del ciclo que se debe seguir para la etapa “*Definir Necesidades a Implementar*”, en donde lo primero que se hace es obtener los requisitos a través de la elicitación, luego se realiza el análisis en el cual se buscan las inconsistencias y cualquier otro tipo de problemas que no se detectaron durante la elicitación, después se pasa a su documentación, en la cual se plasman los conflictos que se encontraron y los requisitos que se obtuvieron.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Identificar los participantes del Proyecto		
Analista	Documentar Usuario Líder	Es la persona que realiza la petición de la aplicación y será la responsable de brindar la información, las necesidades y funcionalidades del nuevo desarrollo. También puede convertirse en usuario Final. La documentación se debe realizar en la plantilla de REM (Apéndice D).
Analista	Documentar Ingeniero de Requisitos	Es la persona que hace el levantamiento de los requisitos. En la mayoría de los casos es el mismo analista. La documentación se debe realizar en la plantilla de REM (Apéndice D).
Analista	Documentar Usuario Final	Es la persona o personas que usará(n) la aplicación. La documentación se debe realizar en la plantilla de REM (Apéndice D).
Analista	Documentar Rol Adicional	Cualquier otra persona que cumpla un papel distinto a los ya definidos en los roles anteriores. La documentación se debe realizar

		en la plantilla de REM (Apéndice D).
Identificar requisitos		
Usuarios	Requisitos Funcionales	Se debe dar una descripción de los requisitos funcionales, que son los pasos que debe ejecutar la aplicación y deben estar asociados a alguno de los objetivos establecidos en el alcance. La documentación se debe realizar en la plantilla de REM (Apéndice D).
Usuarios	Requisitos de Información	Se debe dar la lista de los datos e información que almacenará el sistema. Estos requisitos son de gran ayuda para formar el modelo de datos. La documentación se debe realizar en la plantilla de REM (Apéndice D).
Analista y Usuarios	Requisitos No Funcionales	Debe contener la lista de los requisitos no funcionales del sistema que se hayan identificado, estos son los requisitos técnicos que se presentan en cada una de las etapas. La documentación se debe realizar en la plantilla de REM (Apéndice D).

Analista y Usuarios	Requisitos de Restricción	Se deben documentar las restricciones o reglas de negocio que debe cumplir la información. La documentación se debe realizar en la plantilla de REM (Apéndice D).
Revisar los requisitos		
Analista	Revisar los Requisitos	Al final de cada reunión se debe verificar que los requisitos escritos sean correctos y si hay algún problema o inconsistencia, deben corregirse.
Revisar el Glosario		
Analista y Usuarios	Revisar Glosario	Se debe verificar que los términos que generen ambigüedad o sean desconocidos para cualquiera de los participantes en el proyecto, se definan y tengan un significado para todas las personas.
Generar Acta		
Analista	Generar Acta de Reuniones	En la documentación se debe tener en cuenta: lo que se pactó, las personas que asistieron y otras consideraciones especiales que sean importantes, como cambios a los requisitos, riesgos e indicación de posibles problemas que se pueden presentar; esta documentación servirá como acta

		<p>del procesos y lo pactado en las reuniones.</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D).</p>
Elaborar Casos de Uso		
Analista y Usuario Líder	Definición de Actores	Debe definirse los actores que interactúan con el sistema y describir los roles que cumplen, de modo que puedan participar en los casos de uso.
Analista	Descripción de los requisitos Funcionales	Debe hacerse una descripción de los requisitos funcionales a través de los diagramas de casos de uso, casos de uso extendidos o escenarios. El uso de cualquiera de las técnicas se deja a criterio del analista.
Analista y Usuario Líder	Condiciones del Sistema	<p>La documentación debe contener la lista de precondiciones y poscondiciones respectivas para la ejecución de las funcionalidades del sistema.</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D).</p>
Conformar los Documentos Entregables		
Analista	Documentar Requisitos	Se deben documentar todos los requisitos anotados en el

		<p>transcurso del proceso.</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D).</p>
Analista	Documentar Glosario	<p>Se deben documentar todos los términos definidos en el transcurso del proceso.</p> <p>La documentación se debe realizar en la plantilla de REM (Apéndice D).</p>
Analista	Documentar Casos de Uso	<p>Se deben ubicar en la plantilla de REM los casos de uso que se realizaron en el paso de “<i>elaborar casos de uso</i>”.</p>
Definir Objetivos		
Analista	Definir los objetivos de las reuniones	<p>Deben prepararse las reuniones de acuerdo con lo que se documentó y analizó en las reuniones anteriores, para que no se pierda el rumbo de las entrevistas.</p>
Revisión General		
Analista y Usuario Líder	Revisión General del Documento	<p>Se debe realizar una revisión general al documento, en donde se verifique que todo lo anotado en él es correcto y no tiene inconsistencias.</p>

Tabla 6.3 Descripción de Actividades para definir Necesidades a Implementar

Entrada

- Alcance en el cual se especifican los objetivos
- Necesidades para el nuevo aplicativo
- Condiciones y restricciones funcionales y no funcionales del aplicativo

Salida

- Documento de Requisitos
 - Requisitos de Información
 - Requisitos Funcionales
 - Requisitos No Funcionales
- Actas de las reuniones en las cuales se pactan los acuerdos
- Glosario

Tarea	Actividad	Estado
1	Se establecieron los objetivos y el alcance del proyecto	
2	Fueron validados con el usuario los requisitos y casos de uso que se generaron	
3	Se generó y validó el glosario con todas las personas que participan en el proyecto	
4	Se generaron actas de las reuniones	
5	El usuario aprobó el documento final (Requisitos documentados, casos de uso y otro tipo de especificaciones y anotaciones)	

Check List 6.1 Verificación para Requisitos

6.2.1.2 Analizar y Definir Prioridades

Después de conocer el contexto y alcance de la aplicación en la etapa en la que se realiza el levantamiento de requisitos “*Definir Necesidades a Implementar*” y conociendo las necesidades del usuario, se debe comenzar con la concepción del sistema, para esto se buscará darle una mayor definición a la aplicación a realizar, debido a que se plantarán las bases técnicas para cuando se comience la etapa de desarrollo y pruebas.

El objetivo entonces será establecer el plan de proyecto y la arquitectura correcta del sistema tanto a nivel físico y lógico, como lo serán la creación de la estructura de datos y las funcionalidades de los aplicativos.

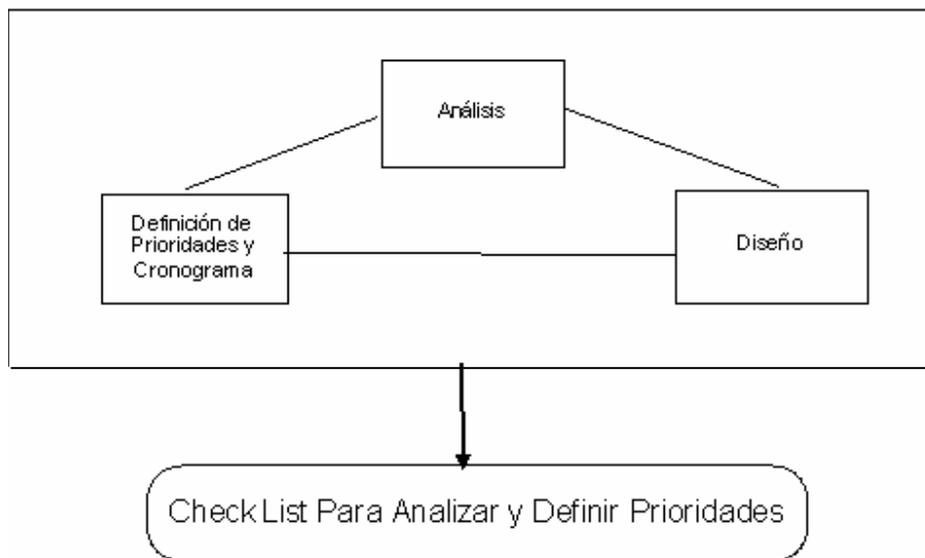


Figura 6.8 Subproceso Analizar y Definir Prioridades

La división de esta etapa en una serie de subprocesos obedece a que las tareas que se realizan pueden ser agrupadas de acuerdo con las funciones que cumplen en:

- Análisis
- Definición de Prioridades y cronograma
- Diseño

Las actividades que se desarrollan dentro de cada uno de los subprocesos de la figura anterior se trabajan a lo largo de todo el proceso de desarrollo y por eso no se muestra una secuencia a seguir, como se ve en la Figura 6.8.

Para conocer la secuencia como se trabajan estas actividades, debe mirarse la Figura 6.2, en la que se muestra la etapa y el flujo completo del proceso, es ahí donde se muestra con claridad la secuencia y el orden en que se ejecutan.

Cada uno de los subprocesos se explica a continuación, con sus respectivas actividades.

6.2.1.2.1 Análisis

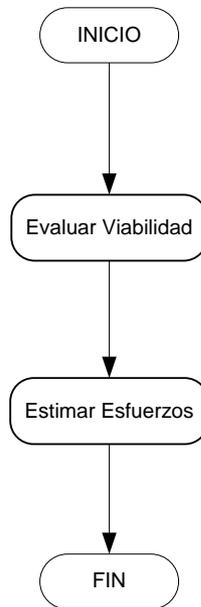


Figura 6.9 Análisis de Viabilidad y Esfuerzo

Objetivos

- Evaluar la capacidad del analista y de los recursos del área para determinar el tiempo de la elaboración del proyecto.

Descripción

En el proceso de análisis se busca medir la disponibilidad y capacidad de las personas y de los recursos del área, para que los usuarios más adelante puedan determinar, dependiendo del tiempo y de su urgencia, cuál será el orden, de acuerdo con sus prioridades, en que desean que se les hagan los entregables y se desarrolle el proyecto.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Evaluar Viabilidad		
Analista	Evaluar Viabilidad	<p>El analista debe evaluar la factibilidad de los requisitos y necesidades que expresa el usuario, además de los recursos del Departamento de Informática para determinar la posibilidad o no de realizar el proyecto.</p> <p>Esta actividad es importante como control de riesgos.</p>
Estimar Esfuerzos		
Analista	Estimar Esfuerzos	<p>El analista debe evaluar la capacidad suya y/o de los recursos del departamento de informática para estimar el tiempo que se puede tardar en la realización de las diferentes actividades del proyecto.</p>

Tabla 6.4 Descripción de Actividades de Análisis de Viabilidad y Esfuerzo

Entrada

- Documento de Requisitos.
- Condiciones y restricciones funcionales y no funcionales del aplicativo.

Salida

- Viabilidad o rechazo para el desarrollo del proyecto a nivel interno.
- Tiempo estimado para el cumplimiento de cada uno de los objetivos del proyecto.

6.2.1.2.2 Definición de Prioridades y Cronograma

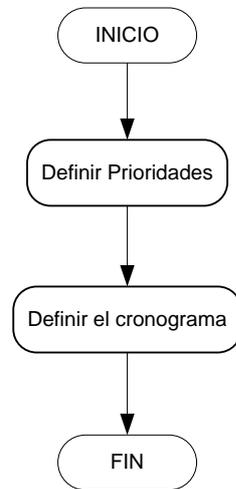


Figura 6.10 Priorización y definición de cronograma

Objetivos

- Definir las prioridades del proyecto para programar un cronograma general.

Descripción

Teniendo en cuenta sus necesidades y lo que el analista definió en el proceso de Análisis, entre el usuario y el analista se determina el orden en que se desarrollará el proyecto y de esta forma se pueda plantear el cronograma de actividades.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Definir Prioridades		
Analista	Tiempo por objetivo	El analista después de conocer la descripción de

		<p>los objetivos y de haber elaborado los casos de uso puede dar un tiempo estimado de lo que puede tardar en su elaboración.</p> <p>Esto servirá como base para que el usuario líder pueda definir mejor sus prioridades y así obtener las etapas en las que se dividirá el desarrollo.</p>
Usuario Líder	Definición de Prioridades	<p>La definición de prioridades se realiza para que las personas que participan en el desarrollo, pacten y definan, según la urgencia del usuario líder, el orden en que se llevará a cabo el desarrollo.</p> <p>Lo que se debe hacer es dividir los objetivos planteados durante el alcance, en etapas o iteraciones del proyecto de entrega.</p>
Definir el Cronograma		
Analista	Cronograma por Entrega	De acuerdo con los

		requisitos y funcionalidades que fueron definidas por el usuario líder y el analista; el analista debe determinar el tiempo que tardará en la realización de las actividades y procesos de Diseño, Desarrollo y Pruebas y Validación, que se realizan en cada entrega.
--	--	--

Tabla 6.5 Descripción de Actividades para Priorización y definición de cronograma

Entrada

- Documento de Requisitos.
- Definición de los problemas y riesgos que se pueden presentar durante el desarrollo del proyecto.
- Tiempo estimado para el cumplimiento de cada uno de los objetivos del proyecto.

Salida

- Cronograma de trabajo para el proyecto.

6.2.1.2.3 Diseño

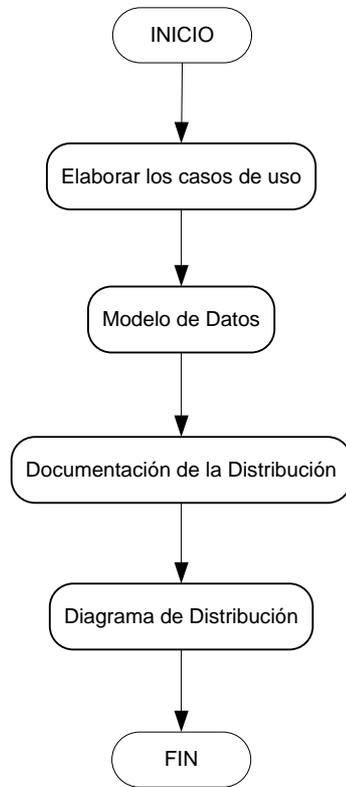


Figura 6.11 Diseño de Arquitectura

Objetivos

- Elaborar la arquitectura del sistema.
- Documentar las relaciones de información del aplicativo que se desarrolla.

Descripción

Para efectuar una definición correcta de la arquitectura del proyecto, lo mejor es realizar varios modelos con los que se pueda obtener una visión clara de los comportamientos que tendrá la aplicación durante la ejecución. Para obtener esta visión se propone que se realicen los siguientes diagramas:

- Casos de Uso
- Modelo de Datos
- Documentación de la Distribución (Ver apéndice B)
- Diagrama de Distribución (Ver apéndice B)

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Elaborar los Casos de Uso		
Analista y Usuario Líder	Definición de Actores	Debe definirse los actores que interactúan con el sistema y describir los roles que cumplen, de modo que puedan ser usados en los casos de uso
Analista	Descripción de los requisitos Funcionales	Debe hacerse una descripción de los requisitos funcionales a través de los diagramas de casos de uso, casos de uso extendidos o escenarios.

		El uso de cualquiera de las técnicas se deja a criterio del analista.
Analista y Usuario Líder	Condiciones del Sistema	La documentación debe contener la lista de precondiciones y poscondiciones respectivas para la ejecución de las funcionalidades del sistema.
Modelo de Datos		
Analista	Modelo de datos o Modelo Entidad Relación (MER), según sea el caso.	Se debe elaborar la estructura de las tablas y las relaciones que estas tienen (Si existen relaciones).
Documentación de la Distribución		
Analista	Documentar la Distribución del Aplicativo	Se debe documentar la distribución y relaciones del aplicativo en lo referente al manejo de los flujos de entradas y salidas de información que este posee.
Diagrama de Distribución		
Analista	Diagrama de Distribución (A criterio del Analista)	El diagrama deberá representar todos los nodos y componentes con

		<p>los que interactúa la aplicación.</p> <p>Hay que tener en cuenta que las relaciones estén bien definidas, de modo que se muestre la dirección en que se comunican los diferentes elementos y componentes.</p>
Analista	Descripción de las aplicaciones, bases de datos y tablas involucradas (A criterio del Analista)	Para cada componente se debe describir su funcionalidad, cómo se relacionan, los nombres de las bases de datos y tablas que interactúan y si estas reciben o entregan datos.

Tabla 6.6 Descripción de Actividades para Diseño de Arquitectura

Entrada

- Documento de Requisitos.
- Registros y trabajos generados en la etapa “*Desarrollar y Realizar Pruebas*”

Salida

- Casos de Uso.
- Registro con las relaciones de entrada y salida de información del Aplicativo.

Tarea	Actividad	Estado
1	Es viable desarrollar el aplicativo por el área de informática	
2	Se definió el cronograma de acuerdo con las prioridades definidas por el usuario	

Check List 6.2 Verificación para Análisis y Diseño

6.2.1.3 Desarrollar y Realizar Pruebas

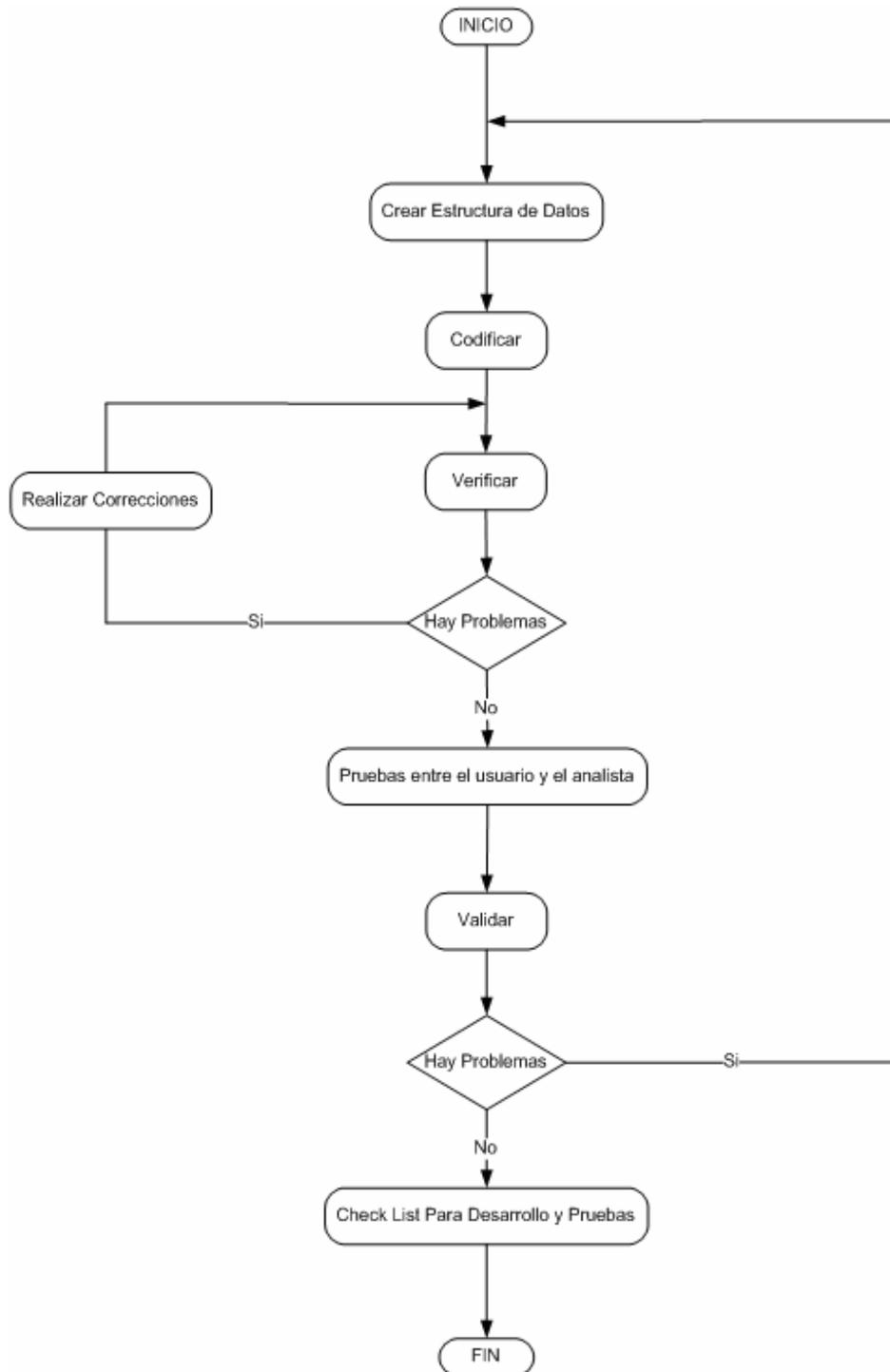


Figura 6.12 Subproceso Desarrollar y Realizar Pruebas

Objetivos

- Desarrollar las funcionalidades que se pactaron para la entrega.
- Probar las funcionalidades que se desarrollaron.
- Validar con los usuarios la pre-entrega de las funcionalidades.

Descripción

El proceso Desarrollar y Realizar Pruebas busca que se codifiquen todas las funcionalidades que se pactaron durante el proceso Definir Necesidades a implementar y que a la vez estas sean probadas, primero por el analista y luego por los usuarios, y durante un paso intermedio el analista pueda ir dando una inducción a los usuarios sobre el manejo del aplicativo, de modo que en el proceso siguiente se pueda realizar completamente la entrega más rápidamente.

También se busca que la codificación y la creación de estructura y objetos de las bases de datos cumplan con los estándares que se exponen en el Apéndice C.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Creación de Estructura de Datos		
Analista	Crear Modelo de Datos	Se debe crear el Modelo de Datos que se realizó en la actividad “ <i>Modelo de Datos</i> ” de la etapa de Análisis y Diseño. Se debe seguir el estándar para Nombramiento de Bases de datos

		que se encuentra en el Apéndice C.
Analista	Descripción de tablas	Todas las tablas que se creen y se elaboren deben ser descritas, identificando la información que almacenan. La descripción de la tabla se realiza en el motor de la base de datos.
Analista	Descripción de los campos de las tablas	Se debe dar una descripción de la información que contiene cada campo. Todos los campos de las tablas deben ser descritos. La descripción de la tabla se realiza en el motor de la base de datos.
Codificar		
Analista	Codificación	Deben ser codificadas las funcionalidades que se pactaron durante el proceso de levantamiento de requisitos. Se debe seguir el estándar para codificación que se encuentra en el Apéndice C.
Analista	Documentación	Se debe documentar el código realizado, de modo que explique y sea fácil de entender cómo realizan y ejecutan sus funcionalidades.

		También se debe documentar el encabezado, describiendo quién y en qué fecha se realizó la codificación.
Verificar		
Analista	Pruebas del Analista	El analista debe hacer pruebas funcionales a los desarrollos que realiza del aplicativo, para que después se lo pueda mostrar al usuario líder.
Realizar Correcciones		
Analista	Realizar Correcciones	Cuando el analista encuentra algún problema dentro de la verificación debe hacer las correcciones necesarias ya sean arreglos de codificación o de estructura de datos, para que luego pueda hacer las revisiones con el usuario.
Pruebas entre el Usuario y el analista		
Analista y Usuario Líder	Pruebas entre Usuario y Analista	Mientras el analista explica el manejo del aplicativo al usuario, a la vez va realizando pruebas a las funcionalidades de la aplicación.
Validar		
Usuario Líder	Pruebas del Usuario	El usuario debe probar y validar que el aplicativo cumpla con las especificaciones indicadas y que

		<p>los datos e información que arroja el aplicativo sean correctos.</p> <p>Mientras el usuario realiza las pruebas, el analista puede avanzar en el desarrollo de otras funcionalidades del aplicativo.</p>
--	--	---

Tabla 6.7 Descripción de Actividades para Desarrollar y Realizar Pruebas

Entrada

- Documento de Requisitos.
- Casos de Uso o escenarios.

Salida

- Aplicativo.
- Estructura de Datos.

Tarea	Actividad	Estado
1	Se ha documentado el código (Encabezado, descripción de funcionalidades).	
2	Fueron descritas las tablas y campos de la Base de Datos	
3	Se realizaron pruebas con el Usuario	

Check List 6.3 Verificación para Desarrollar y Realizar Pruebas

6.2.1.4 Validar y Entregar

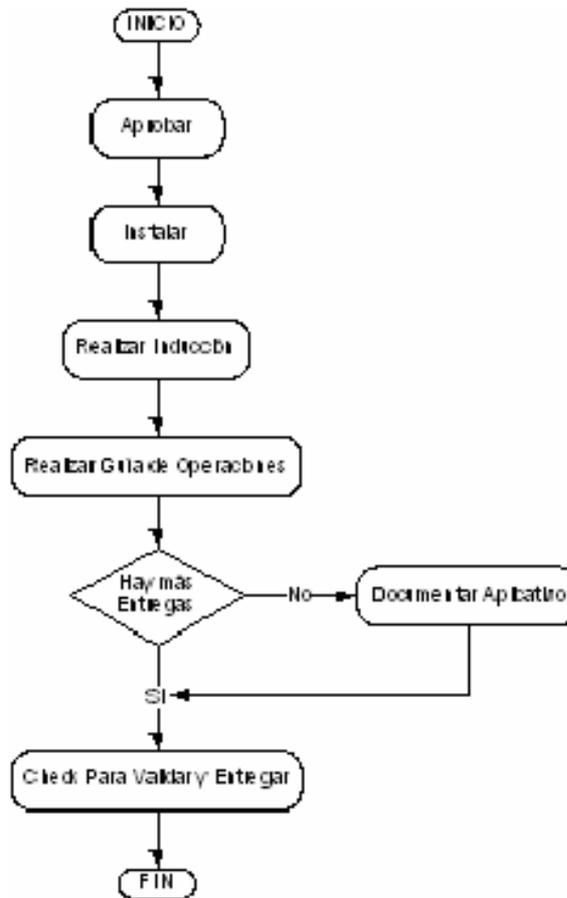


Figura 6.13 Subproceso Validar y Entregar

Objetivos

- Realizar entrega del aplicativo.
- Instalar el aplicativo.
- Realizar inducción y guía de operaciones del aplicativo.
- Actualizar el inventario de aplicativos.

Descripción

En este proceso se busca que el aplicativo se entregue a los usuarios, pero que también se documente y actualice la información que requiere el área.

También busca darle soporte a los usuarios a través de una inducción y una guía de operaciones en la cual puedan conocer el funcionamiento del aplicativo.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Aprobar		
Analista y Usuario Líder	Validación Funcional	Se debe realizar una validación de todos los requisitos y objetivos que se tienen pactados para la entrega, para así comprobar su cumplimiento y que están funcionando correctamente.
Usuario Líder	Confirmación de la Entrega	El usuario debe aprobar o solicitar cambios la entrega que se realiza del aplicativo, después de haber realizado la <i>“validación funcional”</i>
Instalar		
Analista	Crear Carpeta para el Aplicativo	Se debe crear una carpeta en donde se guarden todos

		los archivos y documentos relacionados con el aplicativo en el servidor que corresponda.
Analista	Guardar Código Fuente	Cuando la aplicación esté terminada, se debe guardar el código fuente en la carpeta del servidor correspondiente al aplicativo y en los ambientes de vivo y pruebas que correspondan al aplicativo.
Analista	Instalar la Aplicación	Dependiendo del tipo, las características y arquitectura del aplicativo, se debe realizar la instalación a la persona o personas que la requieran.
Realizar Inducción		
Analista	Inducción	El analista debe realizar la presentación del aplicativo, mostrando su funcionamiento y opciones de trabajo a las personas que lo usarán y/o que se pueden ver afectadas en el cambio de sus actividades debido a la

		implantación.
Realizar Guía de Operaciones		
Analista	Crear Guía de operaciones	El analista debe realizar la guía de operaciones del aplicativo. La guía debe contener los pasos y secuencias que se realizan para ejecutar las diferentes funcionalidades del aplicativo.
Analista	Entregar la guía de operaciones	El analista debe hacer entrega de la guía de operaciones a los usuarios del aplicativo.
Analista	Guardar la guía de operaciones	La guía de operaciones debe ser guardada en la carpeta en donde se encuentra el código fuente del aplicativo.
Documentar Aplicativo		
Analista	Documentar Aplicativo	Debe ser documentada la información del aplicativo, esto se hace llenando los campos que se piden en el archivo " <i>Inventario de Aplicativos</i> ". El formato con la información del inventario

		de aplicativos se encuentra en el Apéndice E.
--	--	---

Tabla 6.8 Descripción de Actividades para Validar y Entregar

Entrada

- Aplicativo.
- Actas de revisiones y pruebas.

Salida

- Acta de aprobación.
- Guía de Operaciones

Tarea	Actividad	Estado
1	Fueron aprobadas todas las entregas	
2	Se realizó la guía de operaciones	
3	Se realizó la inducción a los usuarios del aplicativo	
4	Se le instaló el aplicativo a los usuarios	
5	Se documentó la información que requiere el Área de informática para el inventario de aplicativos	

Check List 6.4 Verificación para Entregas

6.2.2 Metodología para el Proceso de desarrollo A Nivel Externo

Al realizar la adquisición de software se debe tener en cuenta que para muchas de las necesidades de las personas existe software genérico que puede realizar las tareas que los clientes requieren, pero también existe otro tipo de necesidades muy específicas para las cuales se debe desarrollar software a la medida, que debido a su nivel de complejidad, falta de recursos de personas o de tiempo no puede ser desarrollado internamente. Para ambos casos existen metodologías de trabajo que se explican a continuación.

Se debe tener en cuenta que para empezar con cualquiera de las 2 metodologías para la adquisición de software, se debió haber realizado un paso previo, que se explicó al inicio de la sección en la definición de aplicativos, específicamente en la Figura 6.3 Definir metodología del aplicativo.

6.2.2.1 Metodología para la Adquisición de Software

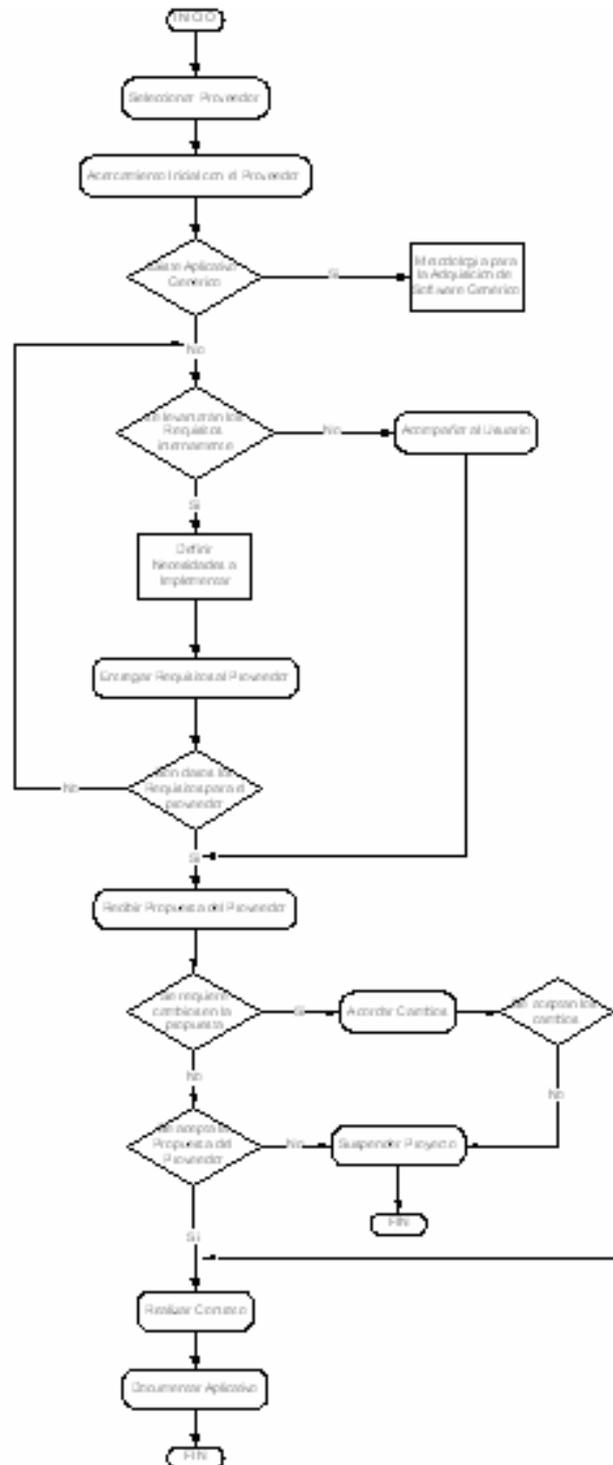


Figura 6.14 Metodología para la Adquisición de Software

Objetivos

- Garantizar el cumplimiento y logro de todos los requisitos y funcionalidades que se quieren con el software que se va a desarrollar.
- Realizar un contrato que garantice las exigencias de la dirección de informática.

Descripción

Para toda adquisición de software es importante conformar un equipo en el cual hayan personas del área que requiere el aplicativo y personas responsables del Departamento de Informática, las cuales garanticen que el software que se adquiriera cumpla realmente con lo que se requiere.

La mejor opción es que el levantamiento de requisitos lo realicen las personas de la Dirección de Informática de la compañía, ya que éstas son las que mejor conocen el modelo del negocio, pero si el levantamiento de requisitos lo realiza la compañía contratada para el desarrollo del aplicativo, en todo momento debe estar un representante de la Dirección de Informática, efectuando acompañamiento durante las reuniones.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Seleccionar Proveedor		
Equipo de Trabajo	Selección del Proveedor	El equipo de Trabajo conformado para la adquisición de software debe seleccionar uno o varios proveedores, de acuerdo

		<p>con el alcance y a lo que se requiera, de modo que después pueda seleccionar la mejor opción.</p> <p>Generalmente se buscan proveedores que hayan realizado trabajos en la compañía.</p>
Acercamiento inicial con el Proveedor		
Equipo de Trabajo	Presentación de Alcance	Se debe presentar el alcance al proveedor, en el cual se encuentran definidos los objetivos y las condiciones que tiene el equipo de trabajo para la adquisición del aplicativo
Proveedor	Análisis Inicial	<p>El proveedor analiza el alcance y las condiciones iniciales que propone el equipo de trabajo.</p> <p>De acuerdo con ese análisis, el proveedor determina si existe un aplicativo genérico en el mercado; si no existe, determina si lo puede realizar y estima el monto de lo que puede costar la realización del mismo.</p>
Acompañar al Usuario		
Analista	Acompañar al Usuario	Se debe acompañar al usuario en las reuniones que tenga con el Proveedor, de modo que pueda asesorarlo, apoyándolo en lo

		referente a la negociación y aspectos tecnológicos.
Analista	Generar Acta	Se deben establecer actas de las reuniones que se realizan con el proveedor, en donde se especifique los asistentes, la fecha y los acuerdos a los que se llegó.
DEFINIR NECESIDADES A IMPLEMENTAR (Ver Figura 6.6)		
Entregar Requisitos al Proveedor		
Equipo de Trabajo	Entrega del Documento de Requisitos	Se debe entregar al proveedor el documento de requisitos que se generó durante el proceso “ <i>Definir Necesidades a Implementar</i> ”.
Recibir Propuesta del Proveedor		
Proveedor	Presentar Propuesta	El proveedor debe presentar una propuesta económica en donde se incluya el cronograma y la descripción de las funcionalidades que tendrá el aplicativo.
Equipo de Trabajo	Analizar Propuesta	Se debe analizar la propuesta presentada por el proveedor, para determinar su viabilidad, condiciones de precio, entrega, responsabilidades, capacitación y así establecer si se acepta, se rechaza o si se debe realizar

		algún cambio.
Acordar Cambios		
Equipo de Trabajo y Proveedor	Acordar Cambios	El equipo de Trabajo debe pactar con el proveedor los cambios que se le desean hacer a la propuesta que éste último presentó.
Suspender Proyecto		
Equipo de Trabajo	Suspender Proyecto	Se llega a este estado por la imposibilidad económica del área interesada en que se lleve a cabo el proyecto o por la falta de los recursos necesarios por parte de la Dirección de Informática para realizar el proyecto. El proyecto luego puede ser retomado desde el punto en el cual se suspendió.
Realizar Contrato		
Equipo de Trabajo	Realizar de Contrato	El contrato es una herramienta que busca garantizar el cumplimiento de los términos establecidos por ambas partes. El contrato debe contener: <ul style="list-style-type: none"> - El documento de Requisitos - Propuesta presentada por el Proveedor - Formas de Pago Y debe garantizar además:

		<ul style="list-style-type: none"> - Servicio de Mantenimiento - Capacitación del personal - Pólizas de cumplimiento - Instalación - Ajustes y Pruebas
--	--	---

Tabla 6.9 Descripción de Actividades para la Adquisición de Software

Entrada

- Descripción básica de las necesidades de los usuarios.

Salida

- Contrato con las condiciones de compra.
- Compra del Aplicativo

6.2.2.2 Metodología para la Adquisición de Software Genérico

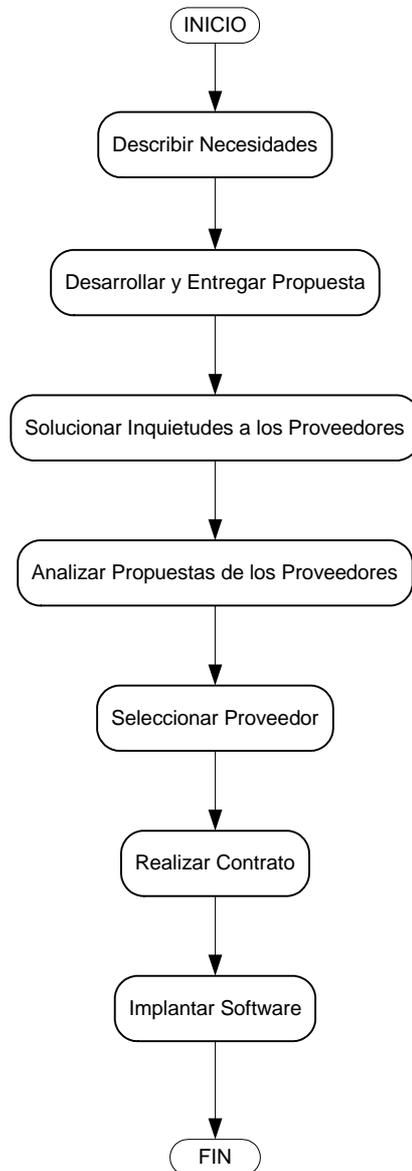


Figura 6.15 Metodología para la Adquisición de Software Genérico

Objetivos

- Seleccionar la mejor opción de software genérico, de acuerdo con lo que ofrecen los proveedores y las necesidades que se tienen.
- Realizar un contrato que garantice las exigencias de la Dirección de Informática.

Descripción

La adquisición de software es una tarea complicada, ya que para la realización de una serie de tareas determinadas pueden existir varias opciones, todas de diferentes proveedores que ofrecen distintas garantías y servicios.

Para seleccionar la mejor opción, la propuesta es conformar un grupo interdisciplinario en donde haya personas del área que requiere el aplicativo y personas de informática que garanticen el cumplimiento por parte del aplicativo, de una serie de condiciones técnicas.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Describir Necesidades		
Analista y usuario	Descripción de Necesidades	Se deben especificar claramente los requisitos y definir las necesidades a las cuales se quiere dar solución. Dependiendo del nivel de especificación que se requiera

		para la descripción de los requisitos, se recomienda trabajar con el proceso “ <i>Definir Necesidades a Implementar</i> ”.
Desarrollar y Entregar Propuesta		
Equipo de Trabajo	Especificación del Marco de Referencia para la Compra	Se debe especificar la información que requiere el proveedor como condiciones de compra, las condiciones de precio, términos y modalidades de pago, entrega, criterios para selección o descalificación de proponentes, métodos y estrategias de negociación y comunicación, requerimientos de confidencialidad, responsabilidades que asumirá, cronograma, etc.
Equipo de Trabajo	Creación de documento RFP (Request for Proposal)	Se debe crear el documento que contenga la descripción de los requisitos y el Marco de Referencia para la compra, para que sea entregado a los proveedores.
Equipo de Trabajo	Creación de Documento de Criterios de Evaluación	Luego de establecer el documento RFP y conocer qué es lo que se requiere y cómo se

		quiere, se debe generar un documento en el cual se especifiquen los criterios con los cuales se evaluarán las propuestas de los oferentes.
Equipo de Trabajo	Entrega de documento RFP a los Proveedores	El equipo del proyecto debe hacer llegar el documento en el cual se encuentran establecidos todos los requisitos a los diferentes proveedores que participarán en la licitación.
Solucionar Inquietudes a los Proveedores		
Equipo de Trabajo	Solución a Inquietudes	Lo más común en las negociaciones es que los proveedores generen preguntas e inquietudes. Estas inquietudes que genere cada proveedor deben ser resueltas y comunicadas a todos los demás proveedores involucrados en la licitación a través del canal de comunicación definido en el marco de referencia para las compras.
Analizar Propuestas de los Proveedores		
Equipo de Trabajo	Evaluar Propuesta	De acuerdo con los criterios de evaluación definidos, se deben evaluar las propuestas presentadas por los

		proveedores, dando puntaje a cada una de las respuestas que ellos presentan a las necesidades entregadas por el equipo de compra.
Equipo de Trabajo	Análisis de Variables con respecto al Proveedor	<p>Existen otros factores y variables diferentes a las necesidades funcionales especificadas a los proveedores que también entran a ser parte fundamental del proceso de decisión.</p> <p>Estos factores y variables que también se deben analizar, son:</p> <ul style="list-style-type: none"> - Tiempo de Entrega - Propuesta Económica - Experiencia del proveedor - Desempeño del aplicativo (Demostraciones) - Soporte del proveedor - Recursos adicionales como Hardware y cualquier otro tipo de recursos especiales. - Trabajos realizados anteriormente para la compañía.

Seleccionar Producto		
Equipo de Trabajo	Selección de Producto	De acuerdo con el puntaje obtenido en la evaluación realizada en el paso anterior y al análisis de los factores y variables especiales presentados por los proveedores, se debe escoger el producto que cumpla mejor con la solución de las necesidades que tiene la compañía.
Realizar Contrato		
Equipo de Trabajo	Realización de Contrato	<p>El contrato es un mecanismo que busca garantizar el cumplimiento de los términos establecidos por ambas partes.</p> <p>El contrato debe contener:</p> <ul style="list-style-type: none"> - Documento RFP - Propuesta presentada por el Proveedor - Formas de Pago <p>Y debe garantizar además:</p> <ul style="list-style-type: none"> - Servicio de Mantenimiento - Capacitación del personal - Pólizas de cumplimiento

		<ul style="list-style-type: none"> - Instalación - Ajustes y Pruebas.
Implantar Software		
Proveedor	Instalación	El proveedor debe encargarse de la instalación del software en todos los computadores y servidores que sea necesario.
Equipo de Trabajo y Proveedor	Ajustes y Pruebas	El proveedor y el equipo de compra deben garantizar que el aplicativo instalado funcione y se ejecute sin problemas.
Proveedor	Capacitación	El proveedor debe capacitar y enseñar al personal cómo manejar el nuevo aplicativo y explotar sus funcionalidades.

Tabla 6.10 Descripción de Actividades para la Adquisición de Software Genérico

Entrada

- Descripción básica de las necesidades de los usuarios.

Salida

- Contrato con las condiciones de compra.
- Compra del Aplicativo

6.2.2.3 Actividades recomendadas para realizar seguimiento después de la contratación²

A continuación se presenta una serie de actividades que buscan garantizar el cumplimiento del contrato después de que este se firma.

Objetivos

- Realizar seguimiento de la adquisición software.
- Realizar Proceso Legal de la adquisición de software.
- Realizar las actividades necesarias para llevar a feliz término el proyecto.

Descripción

Después de la realización de cualquier contrato se debe seguir una serie de actividades para hacer seguimiento y control y así verificar el cumplimiento a cabalidad de las condiciones estipuladas en el contrato que se cerró en las secciones anteriores.

Actividades a Tener en cuenta:

- Gestión de cambios
- Demoras y prórrogas en los plazos límites
- Perfeccionamiento del documento del contrato
- Seguimiento de Gastos

²Oyola Vivero, Agustín José. Jaramillo Gómez, Juan David. Desarrollo de Software Delegado Un Enfoque Estratégico. PROYECTO DE GRADO Universidad EAFIT, 2005

- Gestión de reclamos
- Derechos de Autor
- Aceptación final y cierre del contrato
- Fórmulas de reajuste
- Avance del proyecto de adquisición
- Consorcio de proveedores
- Seguridad industrial
- Preparación de especificaciones técnicas
- Evaluación del desempeño de los contratistas

Gestión de cambios

La gestión de cambios define los procedimientos formales para procesar, evaluar y controlar los cambios que surjan de parte del cliente durante el proyecto, dando así un orden y disminución del caos en el desarrollo de un sistema. La gestión de cambios debe incluirse como cláusula en el contrato para garantizar el cumplimiento de la gestión definida por las partes.

Cuando surja un cambio, el cliente debe diligenciar un documento haciendo referencia a la justificación del mismo, los módulos de software que modificará y su respectivo beneficio. Sí el cliente no tiene lo anterior, el proveedor deberá suministrárselo.

Cualquiera que sea la magnitud y tipo de modificación solicitada, debe seguir un proceso de control de cambios que se define dentro de esta cláusula. Para dicho proceso hay que tener en cuenta los siguientes pasos:

- Establecer quiénes serán los responsables de realizar el seguimiento de los cambios, éstos serán elegidos por parte del cliente y el proveedor.

- Tramitar un documento de petición de cambio, este debe tener: Número único de referencia, descripción del cambio, fecha, nombre de quien lo solicita, evaluación inicial sobre el impacto del cambio, evaluación final sobre impacto en costo y en tiempos, control de autorizaciones, firma y fecha de aprobación.
- Quién evalúa el cambio: Se establece un grupo de personas (pueden ser los responsables de la gestión de cambios) de las dos partes (Cliente y proveedor) quienes evalúan en primera instancia el cambio; de no llegar a un acuerdo, se organiza un segundo grupo el cual incluirá asesores externos.
- Cómo evaluar los cambios: Asignar consecutivos a las peticiones, identificación de áreas o personas claves que se necesitan en la evaluación, medición del impacto técnico, descripción sobre consecuencias de la implementación, aprobación o rechazo, comunicación sobre la decisión al responsable de gestión de los cambios, comunicación al usuario (responsable) solicitante.
- Si se aprueba el cambio, se debe nombrar a alguien responsable de su ejecución, quien mantendrá una comunicación permanente, tanto con el responsable del cliente como del proveedor.

Demoras y prórrogas en los plazos límites

Se definirán los plazos límites para la entrega de módulos por parte del proveedor para su posterior revisión por parte del cliente. En el caso que no se cumplan los plazos límites para la entrega, sólo se justificarán cuando la razón o la causa de la demora estuvieran fuera del alcance del contratista.

El contratista está en la obligación de notificar al cliente sobre las demoras. La causa o clase de la demora determina la acción a tomar. El cliente debe evaluar el tipo y

duración de las demoras para determinar si las prórrogas y los costos solicitados son apropiados y saber si las garantías toman efecto.

Perfeccionamiento del documento de contrato

Durante la ejecución del proyecto puede suceder alguna actividad (trabajos o servicios) no prevista en el contrato inicial, pero de mutuo acuerdo en su ejecución ó simplemente en la extensión del proyecto como tal, se puede acordar el cambio en el alcance. Se elaborará un documento donde señale la descripción de la actividad o la modificación del contrato con su respectivo ajuste y el objetivo del mismo.

Seguimiento de Gastos

Todo proyecto posee un presupuesto al cual hay que darle un buen uso, por ello es necesario hacerle seguimiento a los gastos.

Los indicadores en una empresa son las herramientas con los cuales se toman las decisiones de un negocio, con el seguimiento de los gastos se pueden realizar indicadores que permitan ver de primera mano cómo van las finanzas del proyecto.

Gestión de reclamos

Los reclamos se dan en el transcurso del proyecto o al final de éste por inconformidades de las partes; no quiere decir esto que no haya un buen contrato. Los reclamos pueden venir de ambos lados, tanto del proveedor como del cliente, bien sea por falta de cooperación por parte de la empresa o por costos adicionales no previstos por el cliente. Se hace necesario tener un medio disponible sólo para quejas, donde se le puedan dar una solución lo más rápido posible; cada una de estas quejas serán enumeradas y registradas con su respectiva solución o anotación.

Derechos de Autor

En el desarrollo de Software debe quedar constancia de quién es el dueño de los derechos sobre el producto desarrollado. De acuerdo con la ley, el titular de derecho del Software es la persona natural o jurídica (cliente) que en virtud del contrato obtenga por su cuenta, y en las condiciones previstas, la producción de Software desarrollado por una o más personas; sin embargo, quienes participan en el desarrollo de Software (Proveedor), conservan por término indefinido los derechos morales de la obra. Por lo general en el desarrollo de aplicativos, el cliente es la persona que adquiere parte de los derechos patrimoniales del sistema, con lo cual puede instalarlo, modificarlo y hacerle posteriormente mantenimiento; por el otro lado, el proveedor queda dueño de la idea, lo que significa que el cliente no puede venderla ni cederla sin su consentimiento.

Aceptación final y cierre del contrato

El cierre del contrato se hará cuando se han cumplido las obligaciones pactadas por las partes. El contrato debe cerrarse dejando constancia del cumplimiento de los criterios de aceptación (descripciones indicando qué y cómo se va a probar el Software que ha de ser entregado) y las estipulaciones del contrato, también debe decir si hay algún asunto pendiente entre las partes.

Fórmulas de reajuste:

Si el proyecto que se está desarrollando se paga con diferentes rubros hay que tener en cuenta la variación de este pago frente a la moneda local, por ello hay que definir qué pasa si la moneda local se devalúa o por el contrario se aprecia, quién correrá con los gastos de diferencia; por ello se plantea hacer un ajuste entre los precios de

presentación de la oferta y terminación del proyecto, teniendo en cuenta la variación de la moneda.

Avance del proyecto de adquisición

Esta actividad se realizará durante todo el ciclo de vida del proyecto debido a su riesgo inherente, pero habrá un avance en una fecha próxima a la entrega del proyecto, podría ser una semana antes; se detallarán todas las actividades pendientes, las cuales servirán para determinar si es necesario acudir a un plazo y su respectiva negociación, o si por el contrario se encuentra alineado con el cronograma.

Consortio de proveedores

El proveedor se puede unir con otra empresa para desarrollar el proyecto, pero es a voluntad del cliente si acepta dicha unión, es decir, si hay un costo-beneficio que lo amerite.

Seguridad industrial y salud ocupacional

Dado que hay trabajadores (contratistas) dentro de la compañía, es muy importante definir su situación en caso de alguna eventualidad, asegurándose que todas las pólizas de riesgos profesionales estén al día, permitiendo de este modo salvaguardar a la empresa de cualquier imprevisto.

Sí estos documentos no se encuentran actualizados o no se poseen, es importante comunicarle al proveedor de su obligación de obtener los documentos como requisito para proseguir con el proyecto.

Preparación de especificaciones técnicas

Es esencial implantar los requerimientos antes de la instalación del desarrollo o el proyecto en la empresa, lo que implica tener todas las especificaciones y el ambiente adecuado para que la aplicación funcione sin ningún problema. Estas especificaciones deben ser dadas por el proveedor mucho antes de la instalación, para asegurar un ambiente de ejecución adecuado. Obviamente el cliente debió darle al proveedor las especificaciones técnicas que posee la empresa al comienzo del proyecto.

Evaluación del desempeño de los contratistas

En esta actividad se evaluará la empresa desarrolladora del proyecto; de este modo se podrá responder con confianza a la pregunta ¿se contratará a esta misma empresa en una próxima ocasión? Si la respuesta es afirmativa, es porque al proveedor le ha ido bien en la evaluación realizada por el cliente o en caso contrario, saldrá de la lista de proveedores confiables y por lo tanto no será tomado en cuenta para una futura negociación. Esta evaluación tendrá criterios ya establecidos; se archivará un ejemplar, para tenerla como fuente de primera mano a la hora de contratar.

6.3 Metodología para Modificaciones

Se debe identificar el tipo de modificación que ingresó, ya que éstas se dan porque se requieren actualizaciones o se debe dar solución a algún error de los aplicativos; la diferenciación del tipo de modificación servirá para llevar un control interno de los aplicativos.

Existen dos metodologías o procesos para realizar modificaciones a los aplicativos, que son:

- Metodología para Modificaciones a Nivel Interno
- Metodología para Modificaciones a Nivel Externo

Los criterios para la selección de cualquiera de las dos formas de trabajo se explica en la definición de cada una de las metodologías. También se debe tener en cuenta el nivel de complejidad y/o alcance de la modificación, ya que debido a estas decisiones, el trabajo con la Metodología para Modificaciones a Nivel Interno se puede abordar de diferentes maneras.

Entrada

- Descripción de la modificación requerida

Salida

- Definición de la forma en que se trabajará la modificación
 - Modificación a nivel Interno
 - Modificación a nivel Externo

6.3.1 Metodología Modificaciones Internas

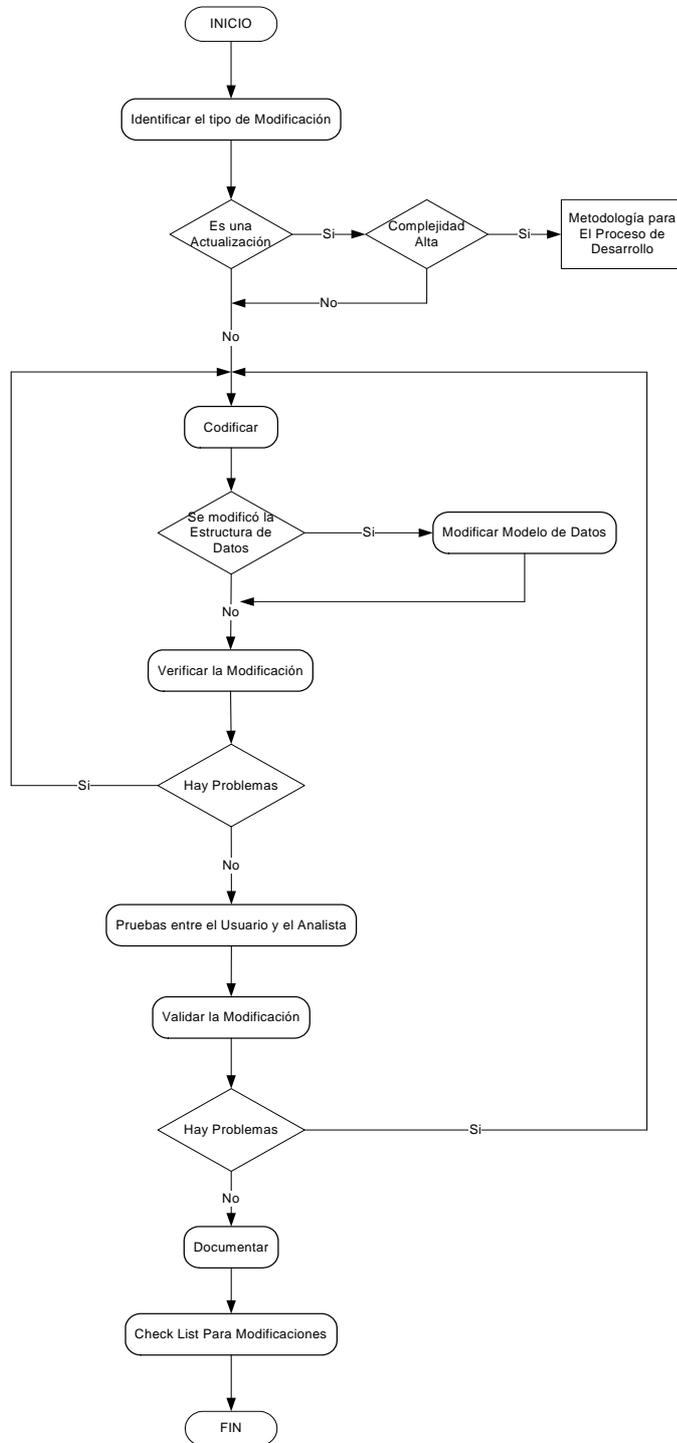


Figura 6.16 Metodología para Modificaciones

Objetivos

- Generar trazabilidad e información para las peticiones de modificación que llegan al área de informática.
- Documentar el trabajo que se realizó en la Dirección de Informática.
- Conocer cuáles son los aplicativos que más se modifican y dar razón de por qué se realiza la modificación.

Descripción

La metodología para modificaciones internas se realiza cuando los usuarios piden nuevas funcionalidades para los aplicativos o cuando debe darse solución a un error del aplicativo. En ambos casos el aplicativo debió haberse desarrollado en la Dirección de Informática

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Identificar el Tipo de Modificación		
Analista	Identificar el Tipo de Modificación	En el momento en que llegue el requerimiento de modificación se debe verificar su tipo, de esta manera se puede conocer cómo se trabajará y documentará la petición. Los tipos de modificación que existen son: Actualización (Adición de una o varias funcionalidades al aplicativo) y

		Solución a Error.
Analista	Identificar el Nivel de Complejidad	Cuando el tipo de modificación es una Actualización, se debe verificar su nivel de complejidad, ya que si su nivel es “Alto” debe trabajarse con la “ <i>metodología para el proceso de desarrollo</i> ”, si no, se continuará con el proceso para modificaciones.
Analista	Documentar	Al identificar el tipo de modificación y su nivel de complejidad, debe asegurarse de que esté documentado en la herramienta de Help Desk: -Quién pide la modificación -Descripción de lo que pide el usuario -Fecha de petición -Fecha estimada de solución -Nombre del responsable que atenderá el caso y -Cualquier tipo de información que sea relevante.

Codificar		
Analista	Codificación	Se debe codificar lo que se especifica en la descripción o las funcionalidades que se pactaron durante la etapa de requisitos;

		Se debe seguir el estándar para Codificación que se encuentra en el Apéndice C.
Analista	Documentación	Se debe documentar el código realizado, de modo que se explique y sea fácil de entender cómo realiza y ejecuta sus funcionalidades. También se debe documentar el encabezado, describiendo quién y en qué fecha se realizó la codificación.
Modificar Modelo de Datos		
Analista	Modificar Modelo de Datos	Se debe anexar o modificar estructuras de datos cuando sea necesario. Es normal que esto suceda, por los nuevos requerimientos de los usuarios. Se debe seguir el estándar para Nombramiento de Bases de datos que se encuentra en el Apéndice C.
Analista	Descripción de tablas	Todas las tablas que se creen y se elaboren deben ser descritas, identificando la información que almacenan. La descripción de la tabla se realiza en el motor de la base de

		datos en la cual se realiza.
Analista	Descripción de los campos de las tablas	Se debe dar una descripción de la información que contiene cada campo. Todos los campos de las tablas deben ser descritos. La descripción de la tabla se realiza en el motor de la base de datos.
Analista	Documentar las Modificaciones	Al realizarse un cambio al modelo de datos del aplicativo, el modelo original queda desactualizado, por esta razón debe actualizarse la información con sus nuevas relaciones.
Verificar la Modificación		
Analista	Pruebas del Analista	El analista debe hacer pruebas funcionales a los desarrollos que realiza del aplicativo, para que después se lo pueda mostrar al usuario líder.
Pruebas entre Usuario y Analista		
Analista y Usuario Líder	Pruebas entre Usuario y Analista	Mientras el analista explica el manejo del aplicativo al usuario, a la vez van realizando pruebas a las funcionalidades de la aplicación.
Validar la Modificación		
Usuario Líder	Pruebas del Usuario	El usuario debe probar y validar que el aplicativo cumpla con las

		<p>especificaciones indicadas y que los datos e información que arroja el aplicativo sean correctos.</p> <p>Mientras el usuario realiza las pruebas, el analista puede avanzar en el desarrollo de otras funcionalidades del aplicativo.</p>
Instalar		
Analista	Guardar Código Fuente	<p>Cuando la aplicación esté terminada, se debe guardar el código fuente en la carpeta del servidor correspondiente al aplicativo y en los ambientes de vivo y pruebas que correspondan al aplicativo.</p>
Analista	Instalar modificaciones al Aplicación	<p>Dependiendo del tipo, las características y arquitectura del aplicativo, se debe realizar la instalación a la persona o personas que la requieran.</p>
Documentar		
Analista	Documentar	<p>Se debe documentar en la herramienta de Help desk:</p> <ul style="list-style-type: none"> -Fecha de finalización -Tipo de modificación que se realizó -Descripción básica de lo que se

		realizó -Descripción técnica en donde se explique qué fue lo que se hizo y cómo se hizo
Analista	Actualizar Información (No siempre hay cambios ni adiciones a las modificaciones)	Al terminar la(s) modificación(es) del aplicativo se debe actualizar la información sobre los cambios experimentados en las relaciones del aplicativo modificado.

Tabla 6.11 Descripción de Actividades para Modificaciones

Entrada

- Descripción con información del usuario y fecha de petición
- Descripción de la modificación requerida.

Salida

- Solución a la modificación pedida
- Registro de las actividades que se realizaron durante la modificación

6.3.2 Metodología Modificaciones Externas

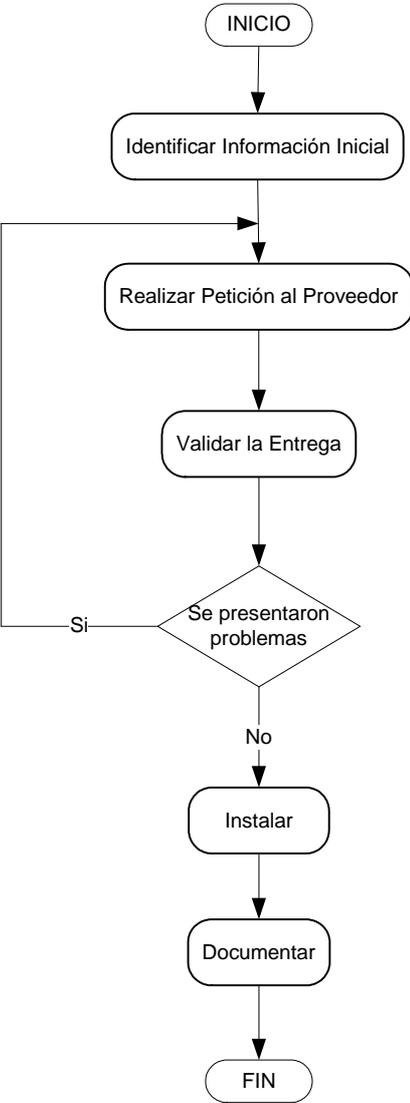


Figura 6.17 Metodología para Modificaciones Externas

Objetivos

- Generar trazabilidad e información para las peticiones de modificación que llegan a la Dirección de Informática.
- Documentar el trabajo realizado con los proveedores.
- Obtener métricas de los aplicativos conociendo cuáles son los que más se modifican y por qué.
- Actualizar la información de los aplicativos que se modifican.

Descripción

La metodología para modificaciones externas se realiza cuando los usuarios piden nuevas funcionalidades para los aplicativos o cuando debe darse solución a un error del aplicativo. En ambos casos el aplicativo debió haber sido desarrollado por un proveedor.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Identificar Información Inicial		
Analista	Identificar Petición	Se debe identificar la petición de modificación, de tal manera que sea claro para el proveedor qué es lo que se desea modificar.
Analista y Usuario Líder	Documentar Información	Debe asegurarse de que esté documentado en la herramienta de Help Desk: - Quién pide la consulta

		<ul style="list-style-type: none"> - Fecha de petición - Fecha estimada de solución - Descripción de lo que se quiere Modificar - Nombre del responsable que atenderá el caso y - cualquier tipo de información que sea relevante.
Realizar Petición al Proveedor		
Analista	Enviar Petición	Se debe hacer la petición de la modificación al proveedor, a través del canal de comunicación que se tenga definido.
Validar la Entrega		
Analista y Usuario Líder	Validar la Entrega	Se debe probar que las modificaciones que se realizaron al aplicativo cumplan a satisfacción con lo que se pidió.
Instalar		
Proveedor	Instalar modificaciones al aplicativo	Dependiendo del tipo, las características y arquitectura del aplicativo y de las modificaciones efectuadas, se debe realizar la instalación a la persona o personas que la requieran.
Documentar		

Analista	Documentar	Debe asegurarse que esté documentado - Fecha de finalización - Tipo de Modificación que se realizó - Descripción del proveedor sobre la modificación realizada
Analista	Actualizar Información (No siempre hay cambios ni adiciones a las modificaciones)	Al terminar la(s) modificación(es) del aplicativo se debe actualizar la información sobre los cambios experimentados en las relaciones del aplicativo modificado.

Tabla 6.12 Descripción de Actividades para Modificaciones Externas

Entrada

- Descripción con información del usuario y fecha de petición
- Descripción de la modificación requerida.

Salida

- Solución a la modificación pedida.
- Registro de las actividades que se realizaron durante la modificación por parte de la Dirección de informática con el proveedor.

Tarea	Actividad	Estado
1	Se tiene la información inicial (Descripción, datos del usuario y fechas de petición y estimada de solución) completa de lo que se pidió	
2	Se realizaron pruebas con el Usuario	
3	Si se crearon nuevas estructuras de datos, estas fueron descritas	
4	Si hubo algún cambio en las relaciones del aplicativo, se documentaron estas actualizaciones	
5	Se documentó la información básica y técnica de la solución de la consulta	

Check List 6.5 Verificación para Modificaciones

6.4 Metodología para Consultas

Las consultas son requerimientos en los que generalmente se piden informes. Las consultas deben trabajarse como se muestra a continuación:

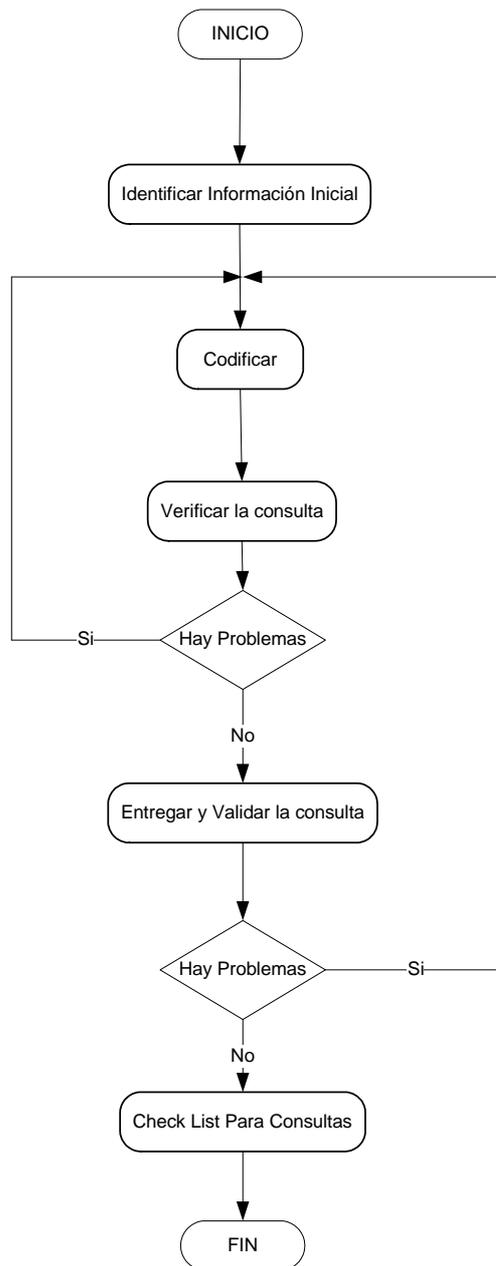


Figura 6.18 Metodología para Consultas

Objetivos

- Generar trazabilidad e información para las consultas que llegan a la Dirección de Informática.
- Documentar el trabajo que se realizó en la Dirección de Informática.

Descripción

En los requisitos para consultas se piden informes, ya sean por pantalla o a través de archivos, por tal razón se requiere de una metodología ágil y fácil de llevar a cabo, en donde se verifiquen los resultados que se obtienen para el informe y a la vez se generen registros del trabajo que se realiza.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Identificar Información Inicial		
Analista	Identificar y Documentar Información Inicial	Debe asegurarse de que esté documentado en la herramienta de Help Desk <ul style="list-style-type: none">- Quién pide la consulta- Descripción de lo que pide el usuario- Fecha de petición- Fecha estimada de solución- Nombre del responsable que atenderá el caso y- cualquier tipo de información que sea

		relevante.
Codificar		
Analista	Realizar Query	Debe codificarse el Query que realice la consulta que se describió en la actividad anterior.
Analista	Documentación	Se debe documentar el Query realizado en el mismo documento y/o archivo en el que se realizó, de modo que explique y sea fácil de entender cómo ejecuta la consulta.
Verificar la Consulta		
Analista	Verificación del analista	Se debe verificar que la información que se obtenga en la consulta realizada sea correcta.
Entregar y Validar la Consulta		
Analista	Entrega	Se debe entregar o enviar al usuario que realizó la petición el resultado, que puede ser un informe, una opción en el aplicativo o un archivo (Excel, plano, otros).
Usuario	Validación del Usuario	El usuario debe probar y validar que la consulta cumpla con las especificaciones indicadas y que los datos e información que arroja la consulta sean correctos.
Analista	Documentar	Debe asegurarse que se documente en la herramienta de Help desk: -Fecha de Finalización

		-Descripción básica de lo que se realizó y -Descripción técnica en donde se explique qué fue lo que se hizo y cómo se hizo.
--	--	--

Tabla 6.13 Descripción de Actividades para Consultas

Entrada

- Descripción con información del usuario y fecha de petición
- Descripción de la consulta requerida.

Salida

- Registro de las actividades realizadas.
- Informe o archivo de la solución, de acuerdo con la petición del usuario.

Tarea	Actividad	Estado
1	Se tiene la información inicial completa de lo que se pidió (Descripción, datos del usuario y fechas de petición y estimada de solución)	
2	Se realizó la documentación del Query	
3	Se verificó que la información obtenida sea correcta	
4	Se documentó la información básica y técnica de la solución de la consulta	

Check List 6.6 Verificación para Consultas

7. PROCESO DE SOPORTE

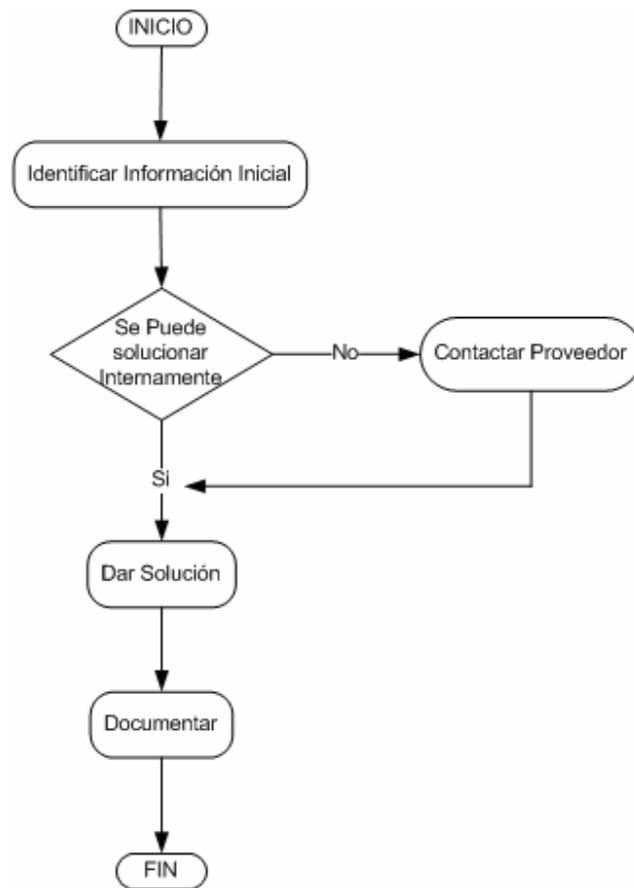


Figura 7.1 Proceso de Soporte

Objetivos

- Generar trazabilidad e información para los problemas que se generan con los equipos.
- Documentar el trabajo que se realizó en la Dirección de Informática.

Descripción

El proceso de soporte busca dar solución a los problemas técnicos a los equipos de las personas de la compañía, la solución a estos procesos ya están definidos en el área de redes y servidores, por esto aquí no se especifican.

Definición De Las Actividades Del Diagrama

Responsable	Actividad	Descripción
Identificar Información Inicial		
Analista	Identificar y documentar Información Inicial	Debe asegurarse que esté documentado en la herramienta de Help Desk: <ul style="list-style-type: none">- Quién pide la consulta- Descripción de lo que pide el usuario- Fecha de petición- Fecha estimada de solución- Nombre del responsable que atenderá el caso y- cualquier tipo de información que sea relevante.
Contactar Proveedor		
Analista	Seleccionar Proveedor	De acuerdo con el tipo de petición que ingrese se

		debe seleccionar el proveedor que pueda darle solución.
Analista	Contactar Proveedor	Se debe contactar al proveedor a través del canal de comunicación que se tiene definido.
Dar Solución		
Analista	Dar Solución	<p>Se debe dar solución al problema descrito en la actividad anterior.</p> <p>Los problemas son muy variados, tanto en lo que hay que hacer como en su grado de complejidad.</p> <p>Los procesos y pasos a seguir con los diferentes tipos de problemas que se presentan están definidos por el área de Redes y Servidores.</p>
Documentar		
Analista	Documentar	<p>Se debe documentar en la herramienta de Help desk:</p> <ul style="list-style-type: none"> -Fecha de finalización -Descripción básica de lo que se realizó -Sí se requiere,

		descripción técnica en donde se explique qué fue lo que se hizo y cómo se hizo
--	--	---

Tabla 7.1 Descripción de Actividades para el Proceso de Soporte

Entrada

- Requerimiento técnico.

Salida

- Registro de las actividades realizadas.
- Solución

8. EJEMPLO PRÁCTICO

A continuación se presentarán fragmentos de un ejemplo real de una compañía del sector de alimentos para un proyecto realizado para el área internacional donde se buscan sistematizar los memos de exportación que se generan para el envío de productos de la compañía a otros países. En el ejemplo se aplica el subproceso “*Definir Necesidades a Implementar*”; Aquí se puede observar cómo se realiza la documentación de un proyecto real, siguiendo el formato y plantillas especificados en el Apéndice D.

Proyecto DATEX

Documento de Requisitos del Sistema

Versión 1.0

Preparado para:

[Área Internacional - Soporte](#)

Preparado por:

[Informática - Desarrollo Comercial](#)

1 Introducción

Descripción general del problema:

En el área internacional de la Compañía se tiene actualmente un proceso de Ventas Internacionales, que es poco ágil y donde no se conserva la integridad de los datos que se manejan.

Dentro de este proceso está el de generación de memos de exportación, el cual involucra la creación o actualización del cliente, la creación o actualización de referencias, el diligenciamiento del [memo de exportación](#) y su respectivo envío a la planta de producción.

4 Alcance de la Aplicación

4.1 Objetivo General

OBJ-0001	Objetivo principal
Versión	1.0 (30/03/2006)
Autores	
Fuentes	
Descripción	El sistema deberá <i>administrar los datos de los clientes, referencias y memos de exportación que son manejados por el área Internacional de la Compañía Nacional de Chocolates.</i>
Subobjetivos	Ninguno
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	baja
Comentarios	Ninguno

4.2 Objetivos Específicos

OBJ-0002	Depuración de la información
Versión	1.0 (28/06/2006)
Autores	
Fuentes	
Descripción	El sistema deberá <i>permitir depurar la información de las referencias, clientes y memos de exportación.</i>
Subobjetivos	Ninguno
Importancia	importante
Urgencia	puede esperar
Estado	en construcción
Estabilidad	baja
Comentarios	Ninguno

OBJ-0006	Estructuración de los procesos
Versión	1.0 (28/06/2006)
Autores	
Fuentes	
Descripción	El sistema deberá <i>permitir estructurar apropiadamente los procesos de ventas internacionales.</i>
Subobjetivos	Ninguno
Importancia	importante
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	baja
Comentarios	Ninguno

5 Cronograma

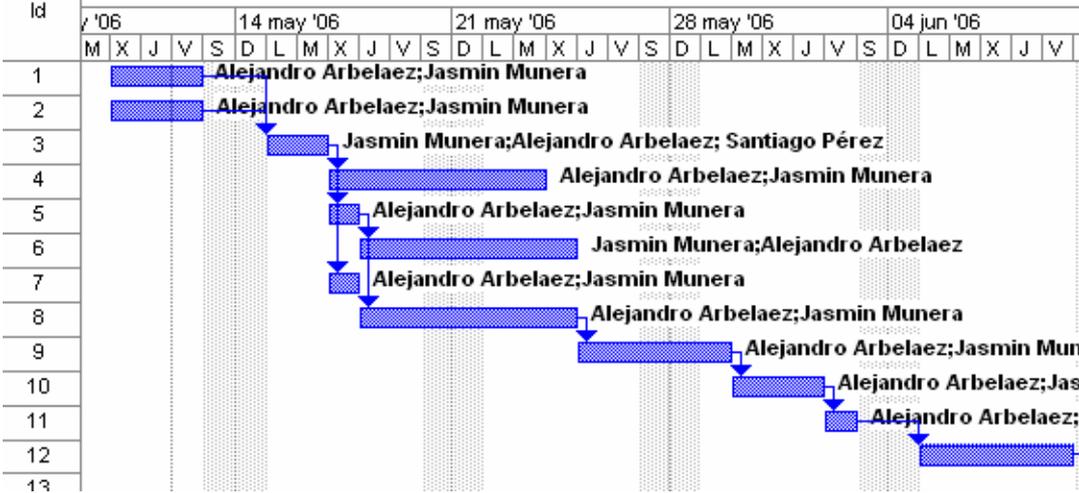


Figura 1: Diagrama de Gantt

7 Requisitos

7.1 Requisitos de información

7.1.1 Requisitos de almacenamiento

IRQ-0003	Usuarios del sistema	
Versión	1.0 (10/04/2006)	
Autores		
Fuentes	?	
Dependencias	Ninguno	
Descripción	El sistema deberá almacenar la información correspondiente a <i>los diferentes usuarios que pueden ingresar al sistema y su perfil</i> . En concreto:	
Datos específicos	Tipo de usuario Nombre del usuario Clave de acceso Permisos Email	
Tiempo de vida	Medio	Máximo
	PD	PD
Ocurrencias simultáneas	Medio	Máximo
	PD	PD
Importancia	PD	
Urgencia	PD	
Estado	PD	
Estabilidad	PD	
Comentarios	Ninguno	

7.2 Requisitos Funcionales

FRQ-0002	Administración de clientes
Versión	1.0 (10/04/2006)
Autores	
Fuentes	?
Dependencias	Ninguno
Descripción	El sistema deberá <i>administrar los datos de los clientes de exportación, es decir, debe permitir crear un nuevo cliente, eliminarlo o actualizarlo.</i>
Importancia	PD
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

7.3 Requisitos No Funcionales

NFR-0002	Base de Datos
Versión	1.0 (18/05/2006)
Autores	
Fuentes	
Dependencias	Ninguno
Descripción	El sistema deberá <i>almacenar la información en SQL server 2000 y posiblemente en System 21.</i>
Importancia	vital
Urgencia	inmediatamente
Estado	en construcción
Estabilidad	baja
Comentarios	Ninguno

8 Reuniones

Reunión	Revisión interfaces de clientes y referencias ACTA 7
Fecha	22/06/2006
Hora	14:10
Lugar	Salita de Sistemas
Asistentes	
Resultados	<p>OBJETIVO:</p> <p>Revisar el formato, datos presentados y funcionalidad de las interfaces de clientes y referencias.</p> <p>ORDEN DEL DÍA:</p> <ol style="list-style-type: none">1. Nombre del sistema2. Revisión de la interfaz de clientes3. Revisión de la interfaz de referencias <p>DESARROLLO:</p> <ol style="list-style-type: none">1. NOMBRE DEL SISTEMA <ul style="list-style-type: none">- Se propusieron varios nombres para el sistema y se preguntó a los ejecutivos del área internacional cual de ellos le gustaba más.- Se definió que el nombre sea DATEX. <ol style="list-style-type: none">2. REVISIÓN DE LA INTERFAZ DE CLIENTES <p>Se mostró la interfaz de creación de cliente, actualización de cliente y consulta de cliente.</p> <p>Observaciones:</p> <ul style="list-style-type: none">- Estudiar la posibilidad de que mediante el sistema se cree la referencia en tutor. <ol style="list-style-type: none">3. REVISIÓN DE LA INTERFAZ DE REFERENCIAS <p>Se mostró la interfaz de creación de referencia. Se hicieron las siguientes observaciones:</p> <ul style="list-style-type: none">- Poner ayuda en línea en los campos del formulario, de tal manera que el ejecutivo sepa como debe llenar un campo o que datos debe poner en el campo.- Al proponer un número de referencia, sólo proponer la de longitud 6.

	<p>- Lista de empaques: el ejecutivo debe saber los códigos de los empaques para la nueva referencia y si quiere adicionarlo, debe digitar el código, pero el sistema no le debe permitir que ingrese empaques con código mayor a E79999.</p> <p>- Estudiar la posibilidad de que mediante el sistema se cree la referencia en tutor.</p> <p>- Registro sanitario: Crear una tabla con el registro sanitario, la cobertura y el país, con el fin de que al crear una referencia y seleccionar la cobertura y los países para los que se va a exportar la referencia, el sistema coloque los números de registro sanitario que le corresponden por país a la referencia de acuerdo al tipo de cobertura que tiene.</p> <p>- Averiguar en la fábrica con Carlos Toro Puerta y José Luis Cardona, que datos de una referencia necesitan consultar mediante el sistema.</p> <p>RESPONSABILIDADES:</p> <ol style="list-style-type: none"> 1. Jasmin Munera: Enviarle a SP archivo Excel con denominación y país y el archivo de coberturas. 2. Santiago Pérez: Completar el archivo en Excel enviado por JM con el tipo de cobertura, según archivo de coberturas. 3. Pedro Soto: Definir las fórmulas para calcular el número de cajas por reef, etc...
Comentarios	Ninguno

9. CONCLUSIONES Y RECOMENDACIONES

Es importante para las Direcciones de Informática de las compañías tener un proceso estructurado que puedan utilizar para ofrecer un servicio de calidad y atender de forma eficaz y rápida las necesidades y nuevos requerimientos de software que se generan constantemente en todas las áreas de las compañías, para que se pueda brindar un apoyo adecuado a cada una de las áreas del negocio.

A la vez, la Dirección de Informática debe soportar su proceso interno obteniendo registros de los trabajos que se realizan, de modo que en un momento determinado puedan hacer una revisión sólida de lo que se hace y cómo se hace.

La documentación debe comenzar desde el momento en que se plantea el proyecto, definiendo los participantes y responsables en él, luego pasando a la definición de las necesidades para que se pueda comenzar la ejecución del desarrollo, y finalmente, en la etapa de cierre se documenta la información básica del software y los aplicativos con los cuales se relaciona; de esta manera se puede obtener una documentación básica y una trazabilidad para el desarrollo de cualquier proyecto.

En lo referente a la adquisición de software, las compañías deben tener una serie de consideraciones especiales, que van desde criterios de aceptación y definición de un producto, en donde se puede determinar si existe algún software en el mercado que cumpla con lo que se requiere o si debe ser un desarrollo a la medida, y que va hasta situaciones tan complejas como el manejo y selección de proveedores y sistemas contractuales, para que después se pueda realizar un seguimiento y así poder efectuar reclamaciones, si se requiere.

En la construcción de la metodología para adquisición de software se consideraron las mejores prácticas para el trabajo con proveedores de software y a la vez se determinó cómo realizar un seguimiento adecuado a estos; todo esto se hizo plasmando estas practicas en una secuencia de pasos que permitirán lograr los objetivos de la contratación. Esta metodología aun no se ha puesto a prueba en la empresa.

Debido a la participación de un grupo tan amplio de personas, el trabajo tuvo aspectos altamente positivos ya que se encontraron y unificaron prácticas y experiencias de cada uno de los participantes de las diferentes áreas de soporte y desarrollo en lo referente a las metodologías, los estándares para programación y definición de objetos de las bases de datos, con lo cual se obtuvo retroalimentación y sinergias que no habían sido identificadas debido al trabajo cotidiano del día a día y que no permitía que las personas se concentraran en esta labor.

La coordinación del grupo de trabajo no fue fácil, ya que todas las personas tenían que seguir realizando sus trabajos y en muchas ocasiones tenían compromisos ineludibles y no podían dedicar mucho tiempo a este trabajo, por tal razón, había que aprovechar al máximo restricciones críticas como el tiempo en las reuniones que se realizaban con cada persona o en las reuniones generales en las que se hacia seguimiento y se revisaban los avances que se lograban. También fue difícil unificar las diferentes ideas que proporcionaban las personas, ya que en muchas ocasiones la forma de realizar la misma actividad por diferentes personas era totalmente distinta, con lo cual, al tener que decidir por una forma de trabajo se hacia muy difícil.

El ejemplo práctico que se muestra en la sección 8, demuestra que actualmente la metodología está siendo usada y probada; como se puede ver en el ejemplo, los formatos para los distintos registros que apoyan la metodología han sido de gran ayuda para la definición de los objetivos del proyecto, la documentación de los

requisitos y para la concepción de las actas de las reuniones, lo cual no se estaba haciendo anteriormente en su totalidad o de forma adecuada.

Además de la aplicación práctica que se muestra en el ejemplo de la sección 8, la metodología también se ha empezado a probar con otro proyecto y los resultados para ambos casos han sido positivos, ya que se está haciendo una documentación adecuada que puede ser revisada fácilmente, debido a que se tiene un formato estándar; en lo que tiene que ver con la codificación, los estándares y las recomendaciones también han sido usados y recibidos con satisfacción, ya que son sencillos de usar y fácilmente recordables por sus nombres nemotécnicos con respecto a cada uno de los nombres de los controles que tienen las herramientas de desarrollo que se usan en la compañía

Es claro que a través del tiempo y de su uso, la metodología y la generación de registros van a tener una serie de modificaciones con las cuales se moldearán y refinarán hasta llegar a un punto óptimo de trabajo; para esto es importante que cada una de las personas a través del uso y la experiencia determinen los aspectos positivos y negativos que han encontrado y los comuniquen a los demás para que se hagan las correcciones correspondientes.

El aprendizaje en este proyecto ha sido amplio y enriquecedor en gran cantidad y variedad de aspectos que van desde lo técnico que fue la consulta e investigación de una serie de metodologías, estándares para el desarrollo y creación de objetos de bases de datos, buenas prácticas en proyectos de software y contratación con terceros, hasta la gestión como lo fue la coordinación, recolección de información y trabajo con un equipo de analistas.

10. BIBLIOGRAFÍA

1. Durán Toro, Amador (2002) Metodología para la Elicitación de Requisitos de Sistemas Software (Versión 2.3)
2. Pressman, Roger S. (2005). Ingeniería del Software: Un enfoque práctico; Quinta edición. McGraw-Hill, Madrid.
3. Introduction to Process Assessment, José Luis Iparragurri, European Software Institute. Madrid SEPG, Tutorial Materials.
4. Harrington, H. James (1993). Mejoramiento de los procesos de la empresa. Editorial McGraw-Hill, Bogotá.
5. Schmuller, Joseph. Aprendiendo UML en 24 horas. Prentice Hall
6. Restrepo Velásquez, Alberto A. Diapositivas de Clase semestre 2004
7. Oyola Vivero, Agustín José. Jaramillo Gómez, Juan David. Desarrollo de Software Delegado Un Enfoque Estratégico. PROYECTO DE GRADO Universidad EAFIT, 2005
8. Guzmán Vásquez, Federico. Definición de una metodología de desarrollo para el proceso software en una empresa del sector de alimentos. PROYECTO DE GRADO Universidad EAFIT, Junio 2006
9. Malan, Ruth. Bredemeyer, Dana. PAPER Functional Requirements and Use Cases

ENLACES

10. Software Engineering Institute
<http://www.sei.cmu.edu/cmmi/>
11. Gestión de riesgos en ingeniería del software,
[http://www.wikilearning.com/gestion de riesgos en ingenieria del software-wkc-3620.htm](http://www.wikilearning.com/gestion-de-riesgos-en-ingenieria-del-software-wkc-3620.htm)
12. SPICE Software Process Improvement and Capability dEtermination,
<http://www.sqi.gu.edu.au/spice/>
13. Software Development,
www.sdmagazine.com
14. developerWorks Rational Rational Edge Seine,
<http://www-128.ibm.com/developerworks/rational/rationaledge/>
15. Alfa – Redi
<http://www.alfa-redi.org/>
16. Modelos aplicados al desarrollo de Software, disponible en Internet
www.software.org/quagmire
17. The Java Language Specification
http://java.sun.com/docs/books/jls/third_edition/html/j3TOC.html

Apéndice A

En este apéndice se hará explicación de los temas relevantes para el manejo de la metodología, en donde se expondrán conceptos claves para el levantamiento de requisitos.

A.1 DEFINICIÓN DE REQUISITOS

La ingeniería de requisitos es el proceso de estudiar las necesidades del usuario para llegar a una definición de requisitos de sistema, hardware o software³.

A continuación se dará una definición y explicación de los requisitos funcionales, no funcionales y de información, con breves ejemplos que ayuden a aclarar su definición.

A.1.1 REQUISITOS FUNCIONALES

Los requisitos funcionales capturan el comportamiento previsto del sistema. Este comportamiento se puede expresar como servicios, tareas o funciones que el sistema se requiere para ejecutarse de forma adecuada⁴.

Todas las operaciones, procedimientos, cálculos, que tenga que realizar el sistema deben estar incluidos en esta categoría; También pertenecen en esta categoría cuál es la información que se va a mostrar y cómo la va a mostrar.

³ Restrepo Velásquez, Alberto A. Introducción a la Ingeniería de Requisitos.

⁴ Malan, Ruth. Bredemeyer, Dana. Functional Requirements and Use Cases

Además, los requisitos funcionales también pueden ser especificados a través de casos de uso o escenarios, con los cuales se les puede hacer una explicación más intuitiva a las personas que no tengan conocimientos tan técnicos como el personal de sistemas.

A.1.2 REQUISITOS NO FUNCIONALES

Son todas las especificaciones técnicas que se deben tener en cuenta para la realización de la aplicación, sus ejemplos más comunes son los siguientes:

- Lenguaje de programación.
- Interfaz de usuario.
- Portabilidad
- Colores
- Entre otros

Cualquier tipo de especificación técnica que no tenga que ver con el funcionamiento ni con los procesos de la aplicación a nivel de software (es decir cualquier tipo de diseño o adición especial), hace parte de esta categoría.

A.1.3 REQUISITOS DE INFORMACIÓN

Esta categoría se usa para saber qué información se va a almacenar en el sistema y es importante para que se desarrollen y organicen las bases de datos.

Con los requisitos de información se busca conocer cuáles son los datos que se van a almacenar en las bases de datos:

- Datos de los clientes
- Datos de los productos
- Datos históricos

- Datos de las facturas
- Datos administrativos

A.2 EJEMPLOS DE PREGUNTAS PARA RECONOCER LAS NECESIDADES DE LAS PERSONAS SEGÚN EL TIPO DE REQUISITO.

A.2.1 Reconocimiento General del sistema

Funcionalidad del sistema:

¿En qué consiste, qué se espera del sistema a desarrollar y qué datos se le proporcionarán?

¿Cuáles departamentos estarán a cargo de las decisiones del proyecto?

¿Quiénes serán responsables por el proyecto?

¿Qué problemas están teniendo las personas?

¿Cuál es el objetivo del Proyecto?

¿Cuáles serían las secciones que conformarían la aplicación?

¿Cuáles de las secciones anteriores requieren ser administradas?

A.2.2 Requisitos de Información

¿Qué datos va a almacenar el sistema?

A continuación se presenta un ejemplo para mostrar las posibles preguntas en un pequeño sistema de facturación

Facturación

- Número de factura.
- IVA (Ó va a estar incluido en los precios de los productos).
- Cantidad de productos vendidos.
- Todos los productos (Uno por uno).
- Precio de los productos.

A.2.3 Requisitos Funcionales

¿Qué funciones quiere que tenga el sistema?

¿El sistema necesita notificar a una persona acerca de los cambios en su estado interno?

¿Hay algún evento externo que el sistema deba conocer? (Entrada de información de otro sistema)

¿Cómo quiere que funcione?

- ¿Cómo se desarrolla el proceso?
- ¿Qué tipo de procedimientos realizan?
- ¿Qué pasos, y si existe algún orden, cuál sería ese orden?
- ¿Qué políticas piensa usar?
- ¿Quiere que la persona ingrese al sistema con su nombre de usuario o que el sistema ingrese con la clave de la intranet?.

¿Qué información quiere que aparezca?

¿Cuáles son las consultas más frecuentes?

- ¿Quiere que se le genere algún informe?

A.2.4 Requisitos No Funcionales

¿En qué herramienta se desarrollará la aplicación?

¿A qué servidor o servidores se conectará la aplicación?

¿De cuáles bases de datos se obtendrá la información?

A.3 PREGUNTAS PARA LA IDENTIFICACIÓN DE ACTORES

Estas preguntas son útiles a la hora de identificar cuáles son las personas que podrán usar el sistema, además sirven como base para la identificación de nuevos requisitos; también son importantes ya que al identificar al usuario final del sistema se le pueden hacer las preguntas apropiadas:

- ¿Quién usa el sistema?
- ¿Cómo inicia el sistema?
- ¿Quién mantiene el sistema?
- ¿Quién apaga el sistema?
- ¿Qué otros sistemas se integraran al aplicativo?
- ¿Quién obtiene la información para el sistema?
- ¿Quién provee información al sistema?⁵

⁵ Restrepo Velásquez, Alberto A. Elicitación, 2004.

Apéndice B

En este apéndice se busca dar una explicación breve de los diferentes elementos que se usarán para el diseño de la arquitectura de la aplicación, especificando cuáles son los documentos y artefactos que se deben elaborar.

B.1 Diagramas de casos de uso o escenarios

Los casos de uso representan las descripciones de las acciones que se realizan desde el punto de vista del usuario. Esta técnica es de gran utilidad a la hora de modelar los requisitos de los usuarios y se vuelve importante ya que genera un lenguaje sencillo y común tanto para las personas encargadas de desarrollar las aplicaciones como para los usuarios.

B.1.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR UN DIAGRAMA DE CASO DE USO O ESCENARIO

Se puede hacer una combinación entre casos de uso y escenarios para explicar las funcionalidades de los sistemas; la selección de cualquiera de estos quedará a criterio de la persona encargada de realizar la modelación y dependerá de la forma en que se pueda expresar más fácilmente con el usuario. Los casos de uso y escenarios presentan características similares y son una representación del comportamiento del sistema.

B.1.2 CASOS DE USO

La modelación de los casos de uso parte de los objetivos que se definieron y que fueron documentados en las diferentes reuniones; en algunas ocasiones se plantean

funcionalidades que quedan en un entorno muy general, son complejas o difíciles de entender, por tal razón se debe recurrir a explicaciones más detalladas en donde entran los casos de uso extendidos; estos deben realizarse para dar una mayor claridad a la hora de elaborar requisitos.

B.1.3 ESCENARIOS

Los escenarios permiten entender la aplicación y su funcionalidad: cada escenario describe una situación específica de la aplicación, centrando la atención en su comportamiento.⁶

Los escenarios se logran realizando una descripción de las secuencias y las interacciones que realizan las personas con el sistema, esta descripción se hace usando un lenguaje natural y fluido, asemejándose al relato de una historia.

Al realizar los escenarios se debe tener en cuenta la identificación de las secuencias y de que sean claras las funcionalidades y las secuencias e interacciones que se ejecutarán cuando se use el programa.

B.1.4 ENTREGABLES

Los artefactos o documentos que se generarán son:

- Diagramas de casos de uso y/o escenarios

⁶ Bertolami, Mabel. Oliveros, Alejandro. Análisis de Puntos Función en la elicitación de requerimientos.

La modelación de los requisitos del sistema es flexible y puede realizarse a través de casos de uso o escenarios.

- Definición de actores

Se debe definir el rol del actor(es) o usuario(s) que interactúa(n) con el sistema.

- Condiciones del sistema

La documentación debe contener la lista de precondiciones y poscondiciones respectivas para la ejecución de las funcionalidades del sistema.

B.2 Modelos de Datos

Los modelos de datos se usan como punto intermedio entre los diagramas de clases y los Modelos Entidad Relación (MER).

En muchas ocasiones no es necesaria la elaboración de diagramas de clases ni la creación de Modelos Entidad Relación (MER), debido a que la complejidad de las aplicaciones no es muy alta o porque el uso de las herramientas y lenguajes no se prestan para la realización de estos tipos de modelos y análisis. Por esta razón el Modelo de Datos se presenta como una opción óptima y ágil.

Para dar integridad a la información y que ésta pueda ser consultada fácilmente, lo mejor es la elaboración de los Modelos de Datos, de acuerdo con las recomendaciones para la creación y nombramiento de objetos de Bases de Datos (Apéndice C), con los cuales se garantice unicidad y uniformidad en los Modelos de Datos.

B.2.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR UN MODELO DE DATOS

No todos los proyectos de la compañía requieren de la construcción de grandes y elaborados modelos de bases de datos; incluso, en muchas ocasiones sólo será necesario crear unas cuantas estructuras o tablas en las bases de datos, que pueden no estar relacionadas.

Sí el proyecto implica el diseño de un nuevo modelo de datos, se debe tener en cuenta que para la creación de todas las tablas y bases de datos se tienen establecidos estándares que cumplir y deben ser descritas todas las tablas y campos que se creen; esto se hará en las opciones que ofrece cada motor en el que se cree el modelo de datos.

También se deben documentar los aplicativos con los cuales interactúa y realiza intercambio de información, ya que esto simplificará la realización de los diagramas de distribución al final del proyecto. La documentación de las relaciones de los aplicativos se realizará en el archivo “*Inventario de Aplicativos*” (apéndice E), el cual contiene el formato apropiado para realizar esta tarea.

B.2.2 ENTREGABLES

Los artefactos o documentos que se generarán son:

- Modelo de datos o Modelo Entidad Relación (MER), según sea el caso.

Se debe elaborar la estructura de las tablas y las relaciones que estas tienen.

- Descripción de tablas.

Todas las tablas que se creen y se elaboren deben ser descritas, identificando la información que almacenan.

La descripción de la tabla se realiza en el motor de la base de datos en la cual se realiza.

- Descripción de los campos de las tablas.

Se debe dar una descripción de la información que contiene cada uno de los campos. Todos los campos de las tablas deben ser descritos.

La descripción de los campos se realiza en el motor de la Base de Datos en el cual se creó.

B.3 Documentación de la Distribución.

La documentación de la distribución busca mostrar textualmente las relaciones de información de los aplicativos, tanto de la información que obtiene de otros aplicativos, archivos, áreas o personas, como la información que genera a otros aplicativos, archivos, áreas o personas.

La información se almacenará en el archivo “*Inventario de Aplicativos*” (apéndice E), de la siguiente manera:

Información de entrada:

La información de entrada se obtiene con los siguientes campos:

- Entrada → Descripción de la Información que ingresan al aplicativo.
- Fuente → Indica de dónde se obtiene la información; las opciones pueden ser:
 - Aplicativo
 - Rol de la persona
 - Área
 - Ente externo
- Tipo de Entrada → Describe el tipo de entrada de los datos al sistema; las opciones pueden ser:
 - Enlace Directo
 - Manual
 - Archivo

Información de Salida:

La información de salida se obtiene con los siguientes campos:

- Salida → Descripción de la Información que genera el aplicativo.
- Destino → Indica dónde se enlaza la información; las opciones pueden ser:
 - Aplicativo
 - Rol de la persona
 - Área
 - Ente externo
- Tipo de Salida → Describe el tipo de salida de los datos que genera el aplicativo; las opciones pueden ser:
 - Enlace Directo
 - Informe
 - Archivo

B.3.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR UN MODELO DE DATOS

Cuando se realice la documentación de las relaciones de los aplicativos se debe verificar que la información sea consistente y que realmente la descripción de la misma que ingresa al aplicativo coincida con el tipo de entrada y la fuente de la cual se obtiene la información, lo mismo debe pasar con la información que genera el aplicativo.

B.3.2 ENTREGABLES

Los artefactos o documentos que se generarán son:

- Descripción de las relaciones de información entre los aplicativos.

Para cada aplicativo deben ser descritas sus entradas y salidas y cómo se relacionan con las otras áreas, personas o aplicativos.

La información quedará contenida en el “*Inventario de Aplicativos*” (apéndice E), en el archivo de Excel especificado para realizar esta tarea.

B.4 Diagrama de Distribución (A Criterio Del Analista)

Los Diagramas de Distribución se usan para modelar la configuración y arquitectura física de los sistemas. Con ellos se pueden representar equipos y dispositivos, cómo éstos se interconectan y también las aplicaciones que están dentro de esos equipos.

B.4.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR UN DIAGRAMA DE DISTRIBUCIÓN

Los diagramas de Distribución comienzan modelando los nodos, que son los elementos de hardware del sistema y las asociaciones de comunicación que existen entre ellos. Dentro de cada nodo es posible ubicar el software que contiene cada uno, con esto se podrá tener una visión global de las relaciones que existen entre los elementos de hardware y los componentes de software que se ejecutan con la aplicación.

- Nodos

Pueden ser computadores de escritorio (PC), servidores, elementos de comunicación como routers, switches y antenas, entre otros.

- Componentes

Son todas las aplicaciones, bases de datos, vistas, interfaces lógicas, con las que interactúe el sistema; los componentes se encuentran dentro de los nodos.

B.4.2 ENTREGABLES

Los artefactos o documentos que se generarán son:

- Diagrama de Distribución.

El diagrama deberá representar todos los nodos y componentes con los que interactúa la aplicación.

- Relaciones bien definidas que muestren la dirección en que se comunican los diferentes elementos y componentes.

Todas las relaciones que tenga la aplicación deben ser legibles y estar correctamente direccionadas.

Apéndice C

C.1 ESTÁNDARES PARA CODIFICACIÓN Y BASES DE DATOS

En lo referente a los estándares de desarrollo se reunió una serie de sugerencias relacionadas con el nombramiento de los objetos que son más comunes en el trabajo con las bases de datos, como los son las tablas y sus campos, procedimientos, índices, triggers, funciones, entre otros.

En lo referente a la codificación, también se hacen sugerencias, en las que no se pretende cambiar la forma de codificar de las personas, pero sí se busca darle orden, legibilidad y uniformidad al código.

A continuación se muestran los estándares que se desarrollaron para las bases de datos y la codificación.

C.2 ESTÁNDARES PARA BASES DE DATOS

Al crear las bases de datos es importante definir desde el comienzo un nombre que sea de fácil recordación y entendimiento, que indique la aplicación y función para la cual fue creada, lo mismo pasa con las tablas que contiene y con los campos que están dentro de esas tablas.

C.2.1 QUÉ SE DEBE TENER EN CUENTA AL REALIZAR LOS DIFERENTES NOMBRAMIENTOS

Las consideraciones que se tienen para la creación de las tablas de las bases de datos son las siguientes:

1. Cualquier nombramiento de tablas y campos se hace con letra mayúscula totalmente, sin marcación de tildes ni letras con otro tipo de marcaciones especiales como la “Ñ” (eñe).
2. La separación de las palabras y/o letras que indican alguna aplicación o tipo de objeto de base de datos deben ser seguidas del carácter “_”.

C.2.2 REGLAS PARA EL NOMBRAMIENTO DE LOS OBJETOS DE LA BASE DE DATOS

Conservar un lenguaje común para el nombramiento de las bases de datos es de suma importancia, ya que facilita su entendimiento a todas las personas y da orden a la hora de crear los modelos de datos.

Las pautas que se proponen a continuación pretenden dar unicidad y uniformidad en la creación de las bases de datos y de sus objetos, también buscan mejorar la identificación de todos sus elementos y evitar los problemas que se pueden presentar por la aparición de sinónimos.

Las abreviaciones que se usarán según el tipo de objeto de la base de datos, son las siguientes:

- Tablas: T
- Índices: I
- Vistas: V
- Procedimientos almacenados: P
- Funciones: F
- Triggers: TR
- Paquetes: PQ

C.2.3 Definición de tablas y campos

En la creación de las bases de datos que almacenarán la información de una o varias compañías es conveniente dar nombres nemotécnicos a las tablas, campos, vistas, procedimientos, entre otros, con los cuales se puedan identificar claramente las aplicaciones a las que pertenecen y los datos que contienen; para esto se proponen las siguientes estructuras que permitirán la estandarización de los modelos de datos:

C.2.4 Definición para las Tablas

Los nombres de las tablas deben seguir la siguiente estructura:

XXXT_YYYYYYYYYYYYYY

Donde:

XXX	Alias definido para la aplicación
T	Letra que indica que es una tabla
YYYYYYYYYYYYYYYYYY	Nombre dado a la tabla (Máximo 15 caracteres).

C.2.5 Definición para los Campos

1. Los nombres de los campos de cada tabla tienen las siguientes consideraciones:

- 1.1. Los nombres de los campos deben ser unificados para todos los campos de las tablas cuando estos almacenan la misma información, sin importar la base de datos o aplicación que los genere, esto es importante en los casos en que se relacionan y se hacen cruces de tablas, permitiendo que sean fácilmente entendidos.
- 1.2. El tipo de dato del campo debe especificarse al inicio del nombramiento, según los criterios que se muestran a continuación.

Tipos básicos:

Tipo de Dato	Tipo Básico	Abreviación
Numérico	Int Long Double Decimal Money Real	N
Cadena	String Char	C
Fecha	Date DateTime SmallDateTime	D

Lógico	Bit Bool Boolean	B
--------	------------------------	---

Tabla C.1 Abreviaciones para los Tipos de Datos Básicos

- 1.3. Cuando los nombres de los campos son compuestos (dos o más palabras), las palabras deben estar separadas por el carácter “_”.

Con las especificaciones anteriores y dando nombres nemotécnicos, se busca que los campos puedan conservar el mismo nombre, sin importar la tabla o base de datos en que fueron generados.

Campos de una palabra

ZYYYYYYYYYYYYYYYY

Donde:

Z	Abreviación del tipo de dato del Campo
YYYYYYYYYYYYYYYY	Nombre nemotécnico del campo que identifica el atributo que se almacena bajo esa variable.

Campos de dos Palabras

ZXXXXXXXXXXXXXXXXX_YYYYYYYYYYYYYYYY

Donde:

Z	Abreviación del tipo de dato del Campo
XXXXXXXXXXXXXXXX	Nombre nemotécnico del campo que identifica la primera palabra del atributo que se almacena bajo esa variable
YYYYYYYYYYYYYYYY	Nombre nemotécnico del campo que identifica el atributo que se almacena bajo esa variable.

Ejemplo:

Variable = string

Información = Nombre de cliente

Nombre de variable que se debería generar = CNOMBRE_CLIENTE

Ejemplo:**Generación de una Tabla y de un Campo**

Compañía	Nacional de Chocolates
Aplicación	ADAM
Sigla de la aplicación	ADA
Tabla de personas	ADAT_PERSONA
Nombre del campo	CTELEFONO

C.2.6 Restricciones

Las restricciones no serán diferenciadas con letras o palabras clave, debido a que muchas veces las condiciones de los campos y de las tablas cambian, lo que haría muy difícil su administración.

La propuesta para reconocer cuál es la Primary Key y/o la Foreign Key es ubicarlos en los primeros campos de las tablas de la base de datos.

En muchos casos la estructura de las tablas tendrá un campo llamado "CCOMPANÍA", esto se debe a que la información que se almacenará, normalmente comprenderá la de varias compañías, luego de ese campo es donde se ubicará en primer lugar la Primary Key y después la Foreign Key.

Vista de la Tabla con sus Campos

Dependiendo del motor de la base de datos y cómo éste represente las tablas, se podrá ver de la siguiente manera.

1. <table border="1" style="margin-left: auto; margin-right: auto;"><thead><tr><th colspan="2">ADAT_PERSONA</th></tr></thead><tbody><tr><td style="text-align: center;">PK</td><td style="text-align: center;"><u>CCEDULA</u></td></tr><tr><td></td><td>CCOMPANIA CNOMBRE CAPELLIDOS CTELEFONO DNACIMIENTO</td></tr></tbody></table> <p style="text-align: center;"><i>Figura C.1 Ejemplo Vista 1. Tabla ADAT_PERSONA</i></p>	ADAT_PERSONA		PK	<u>CCEDULA</u>		CCOMPANIA CNOMBRE CAPELLIDOS CTELEFONO DNACIMIENTO	2. <table border="1" style="margin-left: auto; margin-right: auto;"><thead><tr><th colspan="2">ADAT_PERSONA1</th></tr></thead><tbody><tr><td></td><td>CCOMPANIA CCEDULA CNOMBRE CAPELLIDOS CTELEFONO DNACIMIENTO</td></tr></tbody></table> <p style="text-align: center;"><i>Figura C.2 Ejemplo Vista 2. Tabla ADAT_PERSONA1</i></p>	ADAT_PERSONA1			CCOMPANIA CCEDULA CNOMBRE CAPELLIDOS CTELEFONO DNACIMIENTO
ADAT_PERSONA											
PK	<u>CCEDULA</u>										
	CCOMPANIA CNOMBRE CAPELLIDOS CTELEFONO DNACIMIENTO										
ADAT_PERSONA1											
	CCOMPANIA CCEDULA CNOMBRE CAPELLIDOS CTELEFONO DNACIMIENTO										

La *Figura C.1* muestra explícitamente cuál es la “*Primary key*”, pero en la *Figura C.2* se puede ver que la “*Primary key*” es “*CCEDULA*”, ya que está después del campo “*CCOMPANIA*”.

C.2.7 Definición de Procedimientos, Triggers, Funciones y Vistas

Cuando se creen Procedimientos, Triggers, Funciones, Índices, Paquetes y Vistas, se les debe dar un nombre que haga referencia a la tarea que ejecuten.

C.2.7.1 Definición de Procedimientos

El nombre de los procedimientos deberá estar acorde con la función que ejecutan, su estructura será la siguiente:

XXXXP_YYYYYYYYYYYYY

Donde:

XXX	Alias definido para la aplicación.
P	Letra que indica que es un Procedimiento de bases de datos.
YYYYYYYYYYYYYY YY	Nombre descriptivo del procedimiento.

Existen momentos en los que los “*Procedimientos Almacenados*” son independientes de la aplicación, para estos casos se propone la siguiente estructura de nombramiento:

DBP_YYYYYYYYYYYYYY

Donde:

DB	Especifica que el procedimiento es genérico.
P	Letra que indica que es un Procedimiento de bases de datos.
YYYYYYYYYYYYYY YY	Nombre descriptivo del procedimiento.

C.2.7.2 Definición de vistas

Cuando la vista sea de la tabla original completa, lleva el mismo nombre de la tabla y se sigue la siguiente estructura:

XXXV_YYYYYYYYYYYYYY

Donde:

XXX	Alias definido para la tabla.
V	Letras que indican que es una vista.
YYYYYYYYYYYYYYYY	Nombre que describa la vista.

Para los casos en los que las “*Vistas*” se realizan para más de una tabla, se debe seguir el procedimiento de nombramiento que se muestra a continuación:

DBV_YYYYYYYYYYYY

Donde:

DB	Especifica que la vista es genérica y que se realiza para más de una tabla.
V	Letra que indica que es una vista.
YYYYYYYYYYYYYYYY	Nombre que describa la vista.

C.2.7.3 Definición de Triggers

Los triggers se ejecutan en el momento en que ocurre un disparador o un evento determinado y que cumplen una condición establecida; su estructura es la siguiente:

XXXTR_ CONSECUTIVO

Donde:

XXX	Alias definido para la tabla.
TR	Letras que indican que es un Trigger de base de datos.
CONSECUTIVO	Consecutivo que se le da al Trigger, pero se debe documentar la tarea que realiza.

Nota: En la descripción del Trigger se debe especificar la acción que lo ejecuta y la tarea que realiza cuando es activado.

C.2.7.4 Definición de Funciones

Son métodos que se definen para darle funcionalidades y operaciones especiales a las bases de datos.

El nombre de las funciones deberá estar acorde con la tarea que realizan, su estructura será la siguiente:

XXXF_YYYYYYYYYYYYYY

Donde:

XXX	Alias definido para la aplicación.
F	Letra que indica que es una función.
YYYYYYYYYYYYYYYYYY	Nombre que defina la función.

C.2.7.5 Definición de Índices

La definición de un índice debe identificar la tabla sobre la cual se hace el índice; la estructura será la siguiente

XXXI_YYYYYYYYYYYYYY

Donde:

XXX	Alias definido para la tabla.
I	Letra que indica que es un índice de una tabla.
YYYYYYYYYYYYYYYYYY	Descripción que indique el objetivo del índice.

Nota: En la descripción del índice se debe incluir el campo o campos sobre los cuales se definió el índice.

C.2.7.6 Definición de Paquetes de Procedimientos

Los paquetes de procedimientos se crean cuando existe un grupo de procedimientos que comparten una serie de tareas en común. La estructura que se seguirá para el nombramiento de los paquetes será:

XXXPQ_YYYYYYYYYYYYYY

Donde:

XXX	Alias definido para la aplicación.
PQ	Letras que indican que es un paquete de procedimientos.
YYYYYYYYYYYYYYYYYY	Nombre o descripción del paquete.

C.2.8 DOCUMENTACIÓN EN LAS BASES DE DATOS

La documentación de las bases de datos debe ser una tarea a realizar siempre; es importante para los momentos en que se revisa su estructura, en que se quieran realizar modificaciones o se le quiera agregar nuevas estructuras y funcionalidades, y también para las personas que no estuvieron durante el desarrollo, puedan aprenderlas y conocerlas más fácil y rápidamente.

La documentación de las bases de datos y de sus objetos se debe realizar dentro de la misma base de datos, aprovechando las opciones que cada uno de los ambientes de trabajo brinde, de modo que se simplifique esta tarea.

Todos los procedimientos, triggers, funciones, vistas que se realicen deben seguir las sugerencias de codificación y deben ser documentados como se describe en el documento de “Codificación”.

Ejemplo para la documentación de las tablas y sus campos:

En el siguiente ejemplo se muestra de manera resumida la documentación de una tabla de la base de datos

La tabla PERSONA de la aplicación ADAM.

ADAT_PERSONA1	
	CCOMPANIA CCEDULA CNOMBRE CAPELLIDOS CTELEFONO DNACIMIENTO

Figura C.2 Ejemplo Vista 1. Tabla ADAT_PERSONA1

TABLA

Descripción: Tabla que contiene la información de los empleados de la compañía.

CAMPOS

Todos los campos deben ser descritos sin importar lo simples o explícitos que parezcan.

En el ejemplo el campo CTELEFONO, es un campo en el cual su nombre lo dice todo, pero en este caso el teléfono presenta una ambigüedad, la cual es solucionada por la descripción que se le da al campo, en esta caso particular es el teléfono de la oficina y no el de la casa de la persona.

NOMBRE DEL CAMPO	TIPO	DESCRIPCIÓN
CCEDULA	Varchar2	Cédula del empleado
CNOMBRE	Varchar2	Primer y segundo nombre
CAPELLIDOS	Varchar2	Primer y segundo apellido
CTELEFONO	Varchar2	Teléfono de la oficina
DNACIMIENTO	Date	Campos tipo fecha, formato (AAAAMMDD)
CSEXO	Varchar2	Sexo del empleado

Tabla C.2 Descripción de los campos de la Tabla ADAT_PERSONA1 de la Figura

C.2

C.2.9 VENTAJAS

Lo que se busca con las recomendaciones anteriores es ofrecer versatilidad y facilitar el entendimiento y reconocimiento de los objetos de las bases de datos, dando las siguientes ventajas:

- Permite unicidad y uniformidad a la hora de la creación de las bases de datos y de sus objetos.
- Cuando las personas vean un objeto de las bases de datos, con su nombre podrán identificar la aplicación a la que pertenece, el tipo de objeto y la información que contiene o el tipo de tarea que ejecuta.

- C.3 ESTÁNDARES PARA LA CODIFICACIÓN

Al crear los estándares de programación se pretende dar orden y uniformidad al código que se genera en los distintos desarrollos que se presentan en el área de informática.

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Si bien se debe implementar un estándar de forma prudente, éste debe tender siempre a lo práctico.

Los que se propone a continuación son recomendaciones para mejorar la presentación y entendimiento del código de los aplicativos. Los estándares para la codificación en ningún momento pretenden tocar ni involucrarse en la forma y estilo de programación de las personas.

C.3.1 INDENTACIÓN

Todos los códigos deben ser indentados, de modo que puedan ser fáciles de leer y entender para todas las personas; la mejor opción para indentar el código es usar la tecla TAB.

C.3.2 LÍNEAS Y ESPACIOS EN BLANCO

Las líneas y espacios en blanco mejoran la lectura separando secciones de código que están relacionadas lógicamente.

Siempre se deberían dejar por lo menos 2 líneas en blanco entre las definiciones de clases y métodos.

C.3.3 ESTRUCTURAS DE CONTROL

Las estructuras de control incluyen los condicionales como: **If, For, While, Switch**, entre otros.

Dos buenos ejemplos en los cuales se muestra el uso de la indentación y los espacios en blanco en el uso de las estructuras de control, son los siguientes:

Ejemplo

```
switch (condicion) {  
  case 1:  
    accion1;  
    break;  
  case 2:  
    accion2;  
    break;  
  default:  
    accionpordefecto;  
    break;  
}
```

Ejemplo

```
if ((condicion1) || (condicion2)) {  
    accion1;  
} elseif ((condicion3) && (condicion4)) {  
    accion2;  
} else {  
    accionpordefecto;  
}
```

Los estamentos de control deben estar separados por un espacio entre la estructura de control y el paréntesis, de modo que se diferencien de las funciones.

Este tipo de indentación sirve como apoyo en momentos en los que se presentan errores de tipo lógico o de sintaxis.

C.3.4 CONVENCIONES PARA EL NOMBRAMIENTO DE VARIABLES, OBJETOS Y CONTROLES

C.3.4.1 Nombramiento de Variables

Para el nombramiento de variables se tendrá en cuenta tanto el alcance como el tipo de dato, para facilitar el entendimiento y dar orden a la generación del código. Los nombres de variables no deben empezar con los caracteres subrayado “_” o dólar “\$”, incluso aunque estén permitidos.

Los nombres de variables deben ser cortos y con un significado acorde con lo que representan. La elección de una variable debería ser nemotécnica.

Se debe evitar que los nombres de variable sean de un sólo carácter, excepto para variables temporales. El esquema de nombramiento para las variables es el siguiente:

{Alcance variable}{Tipo básico}{Nombre Variable}

Alcance variable:

Alcance	Abreviación	Descripción
Local	l	Variable definida Dentro de la Función
Módulo de Clase	m	Variable definida en la clase, pero no dentro de la función
Argumento de Función	a	Parámetro de la función
Públicas	g	Variable definida globalmente

Tabla C.2 Abreviaciones para el manejo de Variables

Tipo básico:

Tipo de Dato	Tipo Básico	Abreviación
Numérico	Int Long Double Decimal Money Real	n
Cadena	String Char	c
Fecha	Date DateTime SmallDateTime	d
Lógico	Bit Bool Boolean	b
Objeto	Tipos Complejos	o

Tabla C.1 Abreviaciones para los Tipos de Datos Básicos

Ejemplo

Alcance = Local (Definida dentro de la Función)

Variable = string

Información = Nombre

Nombre de variable que se debería generar = lcNombre

C.3.4.2 Nombramiento de Constantes (k)

Las constantes son datos numéricos o alfanuméricos que no cambian durante la ejecución del programa.

{letra k}{tipo básico}{NombreConstante}

Tipos básicos:

Tipo de Dato	Tipo Básico	Abreviación
Numérico	Int	n
	Long	
	Double	
	Decimal	
	Money	
Cadena	Real	c
	String	
	Char	
Fecha	Date	d
	DateTime	
	SmallDateTime	
Lógico	Bit	b
	Bool	
	Boolean	

Tabla C.3 Abreviaciones para los Tipos de Datos Básicos para manejo con Constantes.

Ejemplo

Alcance = Constante

Variable = string

Información = Nombre

Nombre de variable que se debería generar = kcNombre

C.3.4.3 Nombramiento de Controles y Objetos

Cuando se declaren controles especiales se deben ubicar las letras que lo distinguan al inicio, seguidas de su nombre y este debe ser nemotécnico, de modo que cuando se usen más adelante dentro del código y cualquier persona lo lea, pueda entender y conocer el tipo de control y la información que contiene.

La forma para nombrar los controles es la siguiente:

- Cuando el nombre del control sea de dos palabras o más, se escribe la primera letra de cada una de las palabras del control.
- Cuando el nombre del control sea de una palabra, se debe usar sus consonantes.
- Algunas veces para evitar confusiones al momento de abreviar el tipo de control se aprovechará que los nombres de algunos controles tienen dos palabras y se usarán las consonantes de la primera palabra y la letra inicial de la segunda; para evitar que algunas abreviaciones se repitan.

Las convenciones que se mencionaron anteriormente se aplicarán especialmente para los controles de las WebForms de Visual Studio .Net.

Ejemplo

Control = TextBox

Información = Nombre

Nombre de control que se debería generar = tbNombre

Ejemplo

Control = Label

Información = Nombre

Nombre de control que se debería generar = lblNombre

C.3.4.4 Lista de Abreviaturas Para los Controles

A continuación se describe una lista con los controles que se trabajan en Visual Studio .Net 2003 y en Visual FoxPro con su respectiva abreviación.

Controles de WebForms .NET y de Formularios VFP

VFP	WebForm	Abreviación
Label	Label	lbl
CommandButton	Button	btn
CommandGroup		cg
TextBox	TextBox	tb
EditBox		eb
	LinkButton	lnkb
	ImageButton	ib
HyperLink	HyperLink	hl
Pageframe		pf
ComboBox	DropDownList	ddl
ListBox	ListBox	lb
Gris	DataGrid	dg
	DataList	dl
	Repeater	rpt
CheckBox	CheckBox	cb
	CheckBoxList	cbl
OptionGroup	RadioButtonList	rbl
	RadioButton	rb
Image	Image	img
	Panel	pnl
	PlaceHolder	ph

Timer		tmr
	Calendar	cal
OleControl		oc
	AdRotator	ar
	Table	tbl
	RequiresFieldValidator	rfv
	CompareValidator	cmpv
	RegularExpressionValidator	rev
	RangeValidator	rgv
	CustomValidator	ctmv
	ValidationSummary	vs
	Xml	xml
	Literal	ltr
	CrystalReportViewer	crv

Tabla C.4 Abreviaciones para Controles de WebForms .NET y de Formularios VFP

Datos

Los controles y conexiones para el manejo de datos se distribuyen de acuerdo con su tipo, por lo general cuando se crea una conexión se sigue trabajando con ella durante el desarrollo de todo el proyecto, por esta razón en las siguientes recomendaciones no se hace diferencia de su tipo, pero sí de la función que realizan; las abreviaciones para el manejo de datos y conexiones son las siguientes:

Conexión	Abreviación
DataSet	Ds
DataTable	Dt
DataView	Dv
OleDbDataAdapter	Da
OracleDataAdapter	
SqlDataAdapter	
OleDbConnection	Conn
SqlConnection	
OracleConnection	
OdbcConnection	
SqlCommand	
OdbcCommand	Cmd
OracleCommand	

Tabla C.5 Abreviaciones para el Manejo de Datos y Conexiones

C.3.4.5 Lista de Abreviaturas Para los Objetos de las Formas en VFP

En Visual FoxPro existen objetos (documentos en Visual FoxPro) que se trabajan dentro de las formas y que no se encuentran en las WebForms de Visual Studio .Net. Estos objetos son de uso frecuente y es conveniente identificarlos.

Estas abreviaciones serán de utilidad en el código, ya que cuando se hagan llamados de estos objetos, se puede saber fácilmente cuál es su tipo.

Documento	Abreviación
Formas	frm
Menús	Mnu
Informes	inf
Programas	prg
Librería	lbr

Tabla C.6 Abreviaciones Para los Objetos de las Formas en VFP

Algunas de las abreviaciones para los objetos de las formas de VFP no cumplen con las reglas establecidas para el nombramiento de controles que se especificó anteriormente, pero esto se hace con el fin de que sea más fácil recordar la abreviación.

Ejemplo

Documento = Forma

Información = Ingreso de Empleados

Nombre de control que se debería generar = frmIngresoEmpleados

C.3.5 DEFINICIÓN DE FUNCIONES

Las funciones deberían ser verbos, para la declaración de funciones se debe nombrar completamente en minúsculas, pero si tiene 2 palabras o más, la primera letra de cada palabra adicional debe escribirse con mayúscula.

Ejemplo

```
bool comparar(Parámetro1, Parámetro2)
{
    if (Parámetro1 == Parámetro2) {
        return true;
    } else {
        return false;
    }
}
```

Ejemplo

```
int sumarValores(Parámetro1, Parámetro2)
{
    int resultado = Parámetro1 + Parámetro2;
    return resultado;
}
```

C.3.6. DEFINICIÓN DE CLASES

Los nombres de las clases deben ser mezclas de mayúsculas y minúsculas, con la primera letra de cada palabra en mayúsculas. No se debe escribir ninguna de las palabras con acento o con caracteres especiales.

Ejemplo

```
class Persona
{
    string mcNombre;
    int mnEdad;

    persona(string acNombre, int anEdad)
    {
        mcNombre = acNombre;
        mnEdad = anEdad;
    }

    void cumpleaños()
    {
        mnEdad++;
    }
}
```

C.3.7 LLAMADO A FUNCIONES

Las funciones se deben crear sin espacios entre el nombre de la función y el paréntesis inicial, seguido inmediatamente del primer parámetro.

Después de las comas debería contener un espacio para escribir cada parámetro. A continuación se muestra un ejemplo:

Ejemplo

```
int = sumar(Parámetro1, Parámetro2);
```

Según lo visto en el ejemplo anterior, debería existir un espacio a ambos lados del igual que es usado para asignar el valor de vuelta de una función a una variable. En el caso de un bloque de asignaciones relacionadas, es bueno ampliar el espacio para mejorar la legibilidad.

Ejemplo

```
int           = sumar(Parámetro1, Parámetro2);  
long_int     = sumar(Parámetro1, Parámetro2);
```

C.3.8 DOCUMENTACIÓN DEL CÓDIGO (NECESARIO)

Para la documentación del código lo primero que se debe hacer es crear un encabezado, el cual contenga la siguiente información:

Ejemplo

```
/**  
• Autor del código  
  *  
• Fecha de modificación o finalización  
  *  
• Descripción de las Modificaciones  
**/
```

Además se debe documentar cada una de las funciones que se creen, dando una explicación clara de qué es lo que realiza y cómo lo hace.

En los casos en que las funciones sean muy grandes o que la programación sea estructurada y no presente el uso de funciones, se debe realizar la documentación por bloques de código, en los cuales se explique claramente que es lo que realiza y cómo lo hace.

Ejemplo

```
/**
```

- *Ésta función realiza.*
-

```
**/
```

```
void miMetodo() {
```

```
    int lnContador = 0;
```

```
    // Comienzo del bloque del método
```

-
-
-

```
/**
```

- *Éste bloque de código realiza una comparación de...*

```
**/
```

```
    if (condicion) {           //Comienzo del bloque if
```

```
        int lnInicio = 0;
```

-
-
-

```
    }
```

```
}
```

Apéndice D

Documento para el Proceso de Desarrollo en CNCH

Versión 1.0

Fecha 22/03/2006

Preparado para:

Nombre Centro de Costos o área a la que se le desarrolla

Preparado por:

Dirección de Informática

Índice

- 1 [Introducción](#)
- 2 [Participantes](#)
 - 2.1 [Áreas](#)
 - 2.2 [Personas](#)
- 3 [Descripción del Modelo del Negocio](#)
- 4 [Alcance de la aplicación](#)
 - 4.1 [Objetivo General](#)
 - 4.2 [Objetivos Específicos](#)
 - 4.3 [Definición de Prioridades](#)
- 5 [Cronograma](#)
- 6 [Control de Entregas del desarrollo de la aplicación](#)
 - 6.1 [Entregas Por Iteración](#)
- 7 [Requisitos](#)
 - 7.1 [Requisitos de Información](#)
 - 7.1.1 [Requisitos de Restricción](#)
 - 7.2 [Requisitos Funcionales](#)
 - 7.2.1 [Definición de actores](#)
 - 7.2.2 [Casos de Uso](#)
 - 7.3 [Requisitos No Funcionales](#)
- 8 [Reuniones](#)
- 9 [Matriz de rastreabilidad objetivos/requisitos](#)
- 10 [Glosario](#)
- 11 [Imágenes](#)

1 Introducción

Esta sección debe contener una descripción breve de las principales características del sistema software que se va a desarrollar, la situación actual que genera la necesidad del nuevo desarrollo, la problemática que se quiere resolver y cualquier otra consideración que sitúe a cualquier persona que lea el documento acerca de qué es lo que se desea resolver.

2 Participantes

Al ser un documento para el desarrollo de aplicaciones a nivel interno para la Compañía Nacional de Chocolates, las áreas y centros de costos se tomarán como si fueran las organizaciones, y si es conveniente en la plantilla de participantes en el campo de FAX, se puede colocar el correo de la persona.

2.1 Áreas

Organización	Nombre Centro de Costos o área a la que se le desarrolla
Dirección	Carrera 43A # 1 A Sur - 143
Teléfono	PD
Fax	PD
Comentarios	Ninguno

Organización	Dirección de Informática
Dirección	Carrera 43A # 1 A Sur - 143
Teléfono	3258501
Fax	PD
Comentarios	Ninguno

2.2 Personas

Participante	Nombre Responsable de la Dirección de Informática
Organización	Dirección de Informática
Rol	Desarrollador de la aplicación e Ingeniero de requisitos
Es desarrollador	Sí

Es cliente	No
Es usuario	No
Comentarios	Ninguno
Participante	Nombre Usuario Líder (Cliente)
Organización	Nombre Centro de Costos o área a la que se le desarrolla
Rol	Responsable de dar la información acerca del sistema
Es desarrollador	No
Es cliente	Sí
Es usuario	No
Comentarios	El usuario Líder también puede convertirse en usuario final de la aplicación
Participante	Nombre Usuario Final
Organización	Nombre Centro de Costos o área a la que se le desarrolla
Rol	Será la persona que le dará uso a la aplicación
Es desarrollador	No
Es cliente	No
Es usuario	Sí
Comentarios	Ninguno

3 Descripción del Modelo del Negocio

Se observa y documenta cuáles son las actividades que realizan las personas y cómo estas actividades influyen en el proceso. Es bueno realizar un análisis del comportamiento de los procesos, reconociendo también cuáles son sus entradas y salidas.

También se debe buscar la manera de que dicho proceso pueda ser mejorado analizando sus fortalezas y debilidades

4 Alcance de la aplicación

4.1 Objetivo General

Debe especificar de forma clara y concisa qué es lo que se espera solucionar cuando la aplicación esté siendo utilizada

OBJ-0003	Objetivo General
Versión	1.0 (22/03/2006)
Autores	?
Fuentes	?
Descripción	El sistema deberá PD
Subobjetivos	Ninguno
Importancia	PD
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

4.2 Objetivos Específicos

Esta sección debe contener una lista con los objetivos que se esperan alcanzar cuando la aplicación esté siendo utilizada, especificados mediante la plantilla para objetivos.

OBJ-0001	Objetivo Específico 1
Versión	1.0 (22/03/2006)
Autores	?
Fuentes	?
Descripción	El sistema deberá PD
Subobjetivos	Ninguno
Importancia	PD
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno
OBJ-0002	Objetivo Específico 2
Versión	1.0 (22/03/2006)
Autores	?
Fuentes	?
Descripción	El sistema deberá PD
Subobjetivos	Ninguno
Importancia	PD
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

4.3 Definición de Prioridades

La definición de prioridades se realiza para que las personas que participan en el desarrollo, pacten y definan, según la urgencia del usuario líder, el orden en que se llevará a cabo el desarrollo.

Reunión	Planeación de las etapas del desarrollo
Fecha	28/03/2006
Hora	11:03
Lugar	PD
Asistentes	?
Resultados	Según lo pactado con el Usuario Líder (Nombre), el desarrollo se dividirá en ## entregas. Entrega 1 Los objetivos y módulos que se trabajarán son: Objetivo (#) (Nombre) Objetivo (#) (Nombre) Entrega N Los objetivos y módulos que se trabajarán son: Objetivo (#) (Nombre) Objetivo (#) (Nombre)
Comentarios	Ninguno

5 Cronograma

Debe especificarse el cronograma que se seguirá durante el tiempo que tarde el desarrollo.

Es común que el proyecto se divida por entregas; cuando esto ocurra se debe realizar un cronograma específico por cada una de las entregas que se pactaron.

6 Control de Entregas del desarrollo de la aplicación

Debido a las características especiales del desarrollo de aplicaciones y a la metodología que se tiene, los proyectos se dividen por etapas de entregas para ir haciendo avances de las funcionalidades más críticas. Esto se realiza después de conocer los objetivos de la aplicación a desarrollar y el usuario líder prioriza los objetivos que se desarrollarán en cada una de las etapas, de acuerdo con sus principales necesidades.

6.1 Entregas Por Iteración

En esta sección se ubicarán las actas que se generen en los momentos en que se realice una entrega, en la que se debe exponer qué fue lo que se entregó y cuáles serán los pasos a seguir, de acuerdo con los resultados que se obtuvieron. Para complementar esta tarea, se debe crear una actividad en la herramienta de help desk y asignarla al usuario líder.

Reunión	Entrega ##
Fecha	12/07/2006
Hora	09:47
Lugar	PD
Asistentes	?

Resultados	<p>Se entregaron los siguientes requisitos relacionados con el objetivo ##</p> <p>Requisito ###</p> <p>Requisito ###</p> <p>Requisito ###</p> <p>Quedó pendiente.....</p> <p>En la próxima entrega se realizará.....</p>
Comentarios	Ninguno

7 Requisitos

7.1 Requisitos de Información

Debe contener la lista de requisitos de almacenamiento y de restricciones de información que se hayan identificado.

IRQ-0002	Requisito de Información ##
Versión	1.0 (31/03/2006)
Autores	?
Fuentes	?
Dependencias	Ninguno
Descripción	El sistema deberá almacenar la información correspondiente a PD . En concreto:
Datos específicos	Ninguno

Tiempo de vida	Medio	Máximo
	PD	PD
Ocurrencias simultáneas	Medio	Máximo
	PD	PD
Importancia	PD	
Urgencia	PD	
Estado	PD	
Estabilidad	PD	
Comentarios	Ninguno	

7.1.1 Requisitos de Restricción

CRQ-0001	Restricción
Versión	1.0 (10/04/2006)
Autores	?
Fuentes	?
Dependencias	Ninguno
Descripción	La información almacenada por el sistema deberá satisfacer la siguiente restricción: PD
Importancia	PD
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

7.2 Requisitos Funcionales

Son los pasos y/o procedimientos que tienen que ver con el funcionamiento del sistema y que se realizan para alcanzar los distintos tipos de objetivos de la aplicación.

FRQ-0001	Requisito ##
Versión	1.0 (31/03/2006)
Autores	?
Fuentes	?
Dependencias	Ninguno
Descripción	El sistema deberá PD
Importancia	PD
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

7.2.1 Definición de actores

Se debe definir el rol de actor que ejecutará el sistema.

Es el participante de los casos de uso que se realizaron.

ACT-0001	Nombre Actor 1
Versión	1.0 (22/03/2006)
Autores	?
Fuentes	?

Descripción	Este actor representa <i>PD</i>
Comentarios	Ninguno

7.2.2 Casos de Uso

Aquí deben ubicarse los diagramas de casos de uso, casos de uso extendidos o escenarios que se elaboren después de cada una de las reuniones. El uso de cualquiera de las técnicas se deja a criterio del analista.

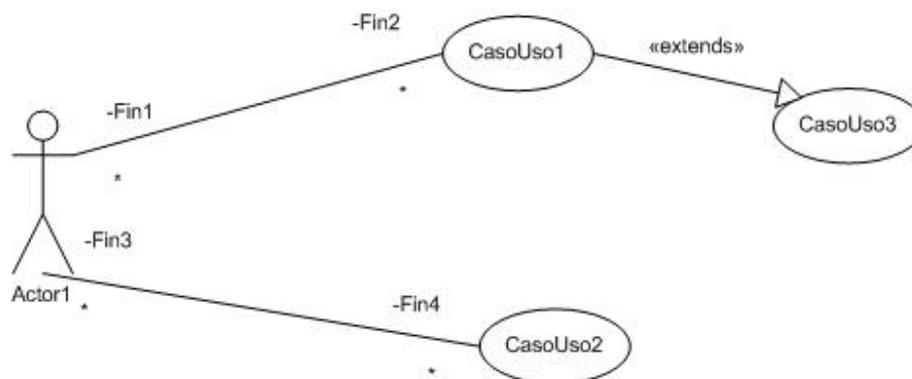


Figura 1: Caso de Uso 01

7.3 Requisitos No Funcionales

Después de conocer el alcance de la aplicación, se puede determinar qué herramienta se va a usar en el desarrollo y en qué base de datos se va almacenar la aplicación.

Lo que se quiere al definir los requisitos no funcionales es conocer las condiciones iniciales con las que se abordará el desarrollo de la aplicación.

Debe contener la lista los requisitos no funcionales del sistema que se hayan identificado, estos son los requisitos técnicos que se presentan en cada una de las etapas.

NFR-0001	Herramienta de Desarrollo
Versión	1.0 (22/03/2006)
Autores	Nombre Responsable de la Dirección de Informática
Fuentes	?
Dependencias	Ninguno
Descripción	El sistema deberá <i>ser desarrollado en la herramienta PD</i>
Importancia	PD
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno
NFR-0004	Requisito No Funcional ##
Versión	1.0 (31/03/2006)
Autores	?
Fuentes	?
Dependencias	Ninguno
Descripción	El sistema deberá PD
Importancia	PD
Urgencia	PD
Estado	PD
Estabilidad	PD
Comentarios	Ninguno

8 Reuniones

Se debe documentar cada una de las reuniones que se elaboren en la etapa.

En la documentación se debe tener en cuenta: lo que se pactó, las personas que asistieron y otras consideraciones especiales que sean importantes como cambios a los requisitos, riesgos e indicación de posibles problemas que se pueden presentar. Esta documentación servirá como acta del proyecto y lo pactado en las reuniones.

Reunión	Reunión para aclarar.....
Fecha	22/03/2006
Hora	09:36
Lugar	PD
Asistentes	?
Resultados	PD
Comentarios	Ninguno

9 Matriz de rastreabilidad objetivos/requisitos

Debe contener una matriz objetivo - requisito, de forma que para cada objetivo se pueda conocer los requisitos con que está asociado.

Opcionalmente y de acuerdo con las necesidades y gustos de las personas, también puede elaborarse otro tipo de matrices que pueda facilitar el entendimiento a todos los involucrados.

10 Glosario

Esta sección deberá contener la definición de los términos abreviaturas, palabras ambiguas o desconocidas que aparezcan en las reuniones y que se considere que su significado deba ser aclarado.

11 Imágenes

Se deben ubicar todas las imágenes que sean útiles para el desarrollo de la aplicación.

Plantillas de Apoyo

A continuación se presentan las plantillas, con la información necesaria para que la Dirección de Informática pueda conocer cuál es la necesidad del usuario y a la vez generar un historial que le permita obtener una trazabilidad de los trabajos que se realizan en la Dirección.

Plantilla para Consultas

Para realizar las consultas, la Dirección de Informática requiere 2 plantillas, la primera la llena el Usuario con la información básica de él y con las necesidades que deben ser resueltas; la segunda es una plantilla interna que será para uso y control interno de la Dirección de Informática.

La información que se tiene en estas plantillas puede ser tomada de una manera más simple y rápida a través de una herramienta de Help Desk, lo importante es que se asegure que se documente toda la información básica que se muestra en las plantillas.

PLANTILLA PARA CONSULTAS DEL USUARIO

Nombre del usuario:	Nombre del Ingeniero:
Centro de costos:	Área Interna:
Fecha de Inicio:	Fecha de culminación:

Prioridad:

Requiere un programa para realizar esta tarea periódicamente:

SI_____

NO_____

Desea que se le entregue un Informe con los resultados de la consulta:

SI_____

NO_____

Descripción:

Especifique la información que quiere que se le muestre:

- 1.
- 2.
- 3.
- 4.

Aprobación: SI___

NO___

Plantilla 1: Plantilla Para Consultas Del Usuario

Como se puede observar en la Plantilla 1 “PLANTILLA PARA CONSULTAS DEL USUARIO”, dependiendo de la periodicidad con que el usuario desea realizar la consulta, tiene la posibilidad de pedir que se le desarrolle una pequeña aplicación que ejecute la consulta en el momento que lo desee.

PLANTILLA PARA SOLUCIÓN CONSULTAS (PARA USO INTERNO)	
Nombre del usuario:	Nombre del Ingeniero:
Centro de costos:	Área Interna:
Fecha de Inicio:	Fecha de culminación:
<p>DESCRIPCIÓN DE LA SOLUCIÓN:</p> <p>DETALLE DE LA SOLUCIÓN:</p> <p>(Query o Código del programa que realizó)</p> <p>-Explique específicamente qué fue lo que hizo.</p> <p>-Qué programas y herramientas usó.</p>	

Plantilla 2: Plantilla Para Solución Consultas

Plantilla Para Modificaciones.

Para realizar las Modificaciones, la Dirección de Informática requiere 1 plantilla en la que el Usuario especifica las necesidades de modificación del aplicativo, ya sea porque requiere una actualización o porque encontró algún error y requiere que sea corregido.

PLANTILLA PARA MODIFICACIONES DEL USUARIO	
Nombre del líder:	Nombre del Ingeniero:
Centro de costos:	Área Interna:
Fecha de Inicio:	Fecha de culminación:
Tipo de petición:	
Cambio_____	Error_____
Aplicación a la que se le hará el cambio:	
Justificación del cambio que se realiza:	
Requerimientos	
Aprobación: SI____ NO____	

Plantilla 3: Plantilla Para Modificaciones Del Usuario

Para completar la información y generar un historial de los cambios y trabajos que se le han hecho a los diferentes aplicativos, se debe llenar el formato de Excel que se explica en el Apéndice F.

Apéndice E

Se definió un formato en Excel, el cual contiene todos los campos que se requieren por cada uno de los aplicativos, de modo que la información allí contenida sea consistente y genérica para los diferentes aplicativos.

	A	B	C	D	E	F	G	H	I	
1	Datos Básicos									
2										
3										
4	Compañía	Categoría	C.Costos (Área)	Aplicativo	Descripción	Tipo Licencia	Version	Estado	Tipo de Softwar	Respo
5	CNCH	Tercero	Fonal	Visual Mutuo	Sistema para la administración del Fondo Mutuo de Inversion de los empleados de Inversiones Nacional de Chocolates, Compañía Nacional de Chocolates, Industria Colombiana de Café y Fábrica de Café La Bastilla (Fonal).	Nombrada	2.1	Activo	Aplicativo	Hernán Osorio
6	Fonal	Tercero	Fonal	Acceso informacion BVC (Pantalla Pasiva)	Permite realizar consulta del movimiento en línea del mercado de acciones de la bolsa de valores de Colombia con el fin de tomar oportunamente decisiones de inversión.	Nombrada		Activo	Aplicativo	Hernán Osorio
7	Fonal	Propio	Fonal	Asamblea Fonal	Sistema para el registro de asistentes a la Asamblea de Afiliados a Fonal, así mismo para la generación del Quorum.	Propio	1.07	Activo	Aplicativo	Hernán Osorio
8	Fonal	Propio	Fonal	Comprobantes de Pago Fonal	Permite elaborar recibos de caja y cheques para pago, los cuales pueden ser impresos y además consultados durante el mes en curso.	Propio	1	Activo	Aplicativo	Gabriel Arangc

Figura E.1 Inventario de Aplicativos

Además de la información que se documentó de cada uno de los aplicativos, se creó una tabla dinámica que se actualiza cada que se abre el archivo, logrando que la información esté resumida, actualizada y sea fácil de consultar.

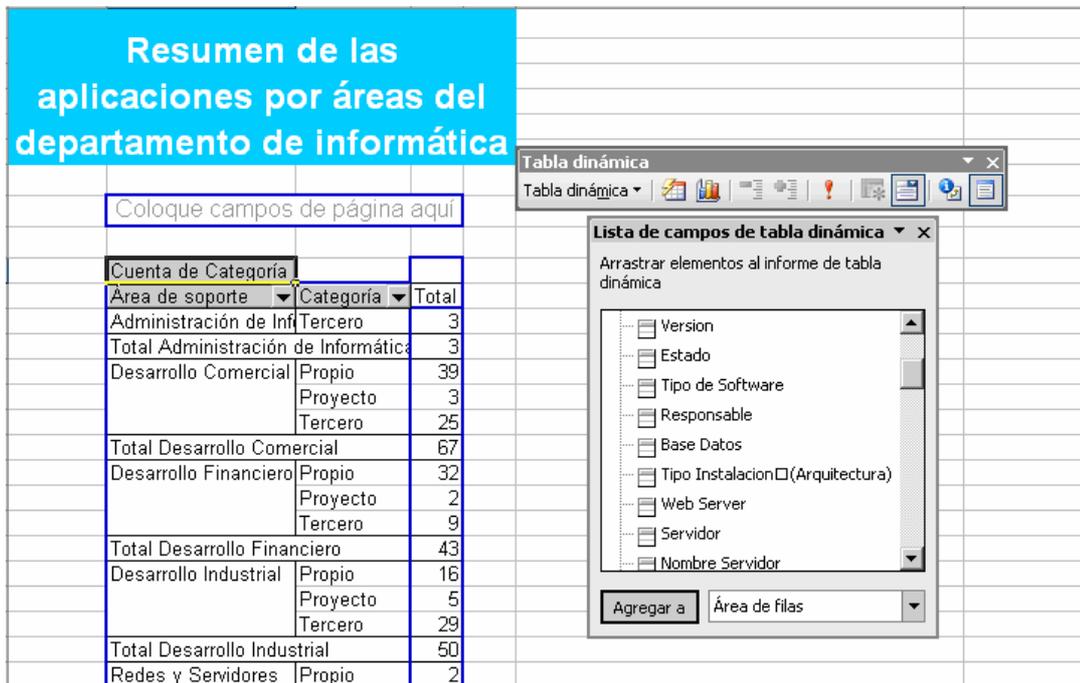


Figura E.1 Inventario de Aplicativos

La gráfica muestra un ejemplo de la tabla dinámica en la cual se pueden generar diferentes resúmenes del inventario de aplicativos de la compañía.

Apéndice F

Con la intención de generar trazabilidad e historial de los diferentes trabajos que se realizan con los aplicativos, se desarrolló el siguiente formato en Excel, el cual contiene toda la información que se requiere para obtener registros apropiados de las modificaciones que los aplicativos han tenido, justificando el por qué se hizo el trabajo en ellos, quién los solicitó y en qué fecha.

Consecutivo	Tipo E=Error R=Requerimiento C=Consulta	Solicitado por	Descripción	Estado	Fecha Solicitud	Fecha Solución	Fecha Aplicación	Responsable
3	R		Activar el sistema de Alarmas	Finalizado		15/01/2003	07/02/2003	
4	R		Agregar mas criterios de búsqueda dentro de un expediente	Finalizado		16/01/2003	07/02/2003	
5	R		Permitir realizar búsquedas entre expedientes	Finalizado		16/01/2003	07/02/2003	
6	R		Carga automática de Proveedores, acreedores y clientes	Finalizado		21/01/2003	07/02/2003	
7	R		Manejo de Multicompañía	Cancelado		SD		
8	R		Interfase con System 21	Finalizado		30/07/2003	08/08/2003	
9	C		Explicar el funcionamiento de la Paleta de Controles (Pos_it, Sellos, Firmas, Anotaciones etc.)	Finalizado	28/01/2003	20/02/2003	20/02/2003	
10	R		Selección Masiva de documentos en las Bandejas Personales, para facilitar el paso de dichos documentos al siguiente paso de la ruta	Finalizado	02/04/2003	20/06/2003	24/06/2003	
11	R		Posibilidad de Organizar los documentos en las Bandejas Personales, por los iconos que muestran el numero del paso en la ruta	Finalizado	29/01/2003	20/02/2003	20/02/2003	
			Cuando un documento es enviado a una área, a todas las personas de dicha área les llega copia del documento a la bandeja de entrada					

Figura F.1 Registro por Aplicativo