

Modelo Integral para la Construcción de Aplicaciones de Comercio Electrónico – MICACE



**Muller Alirio Rosero Palacios
Holmes Giraldo Zambrano Melo**

Director: Ing. Mario Fernando Solarte Sarasty

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telemática
Línea de Investigación en Ingeniería de Sistemas Telemáticos
Popayán, Septiembre de 2003

A Dios por tener siempre presente mis plegarias y guiar mi camino.

A Myriam, mi madre, por ser la mano tierna que pule mis senderos y de la que siempre recibí la muestra más fiel de la comprensión; a Segundo, mi padre, por heredarme el carácter con el que conduzco mi vida, por su amistad sin límites y el sabio consejo. A los dos por brindarme lo mas puro de su amor y su confianza. A mi hermano Byron y su familia por su cariño y apoyo incondicional.

A Martha Isabel por llenar mi corazón de tanto amor y felicidad.

A mis amigos por todos los buenos momentos compartidos que llenan mi vida de alegría, y de los que siempre recibí una voz de aliento.

Holmes

A Bladimir,

A mi mamá por todo su amor,

A mi papá,

A mi hermano por su cariño,

A Victor Manuel por su apoyo,

A toda mi familia,

A los amigos que siempre estuvieron conmigo,

A Tatiana.

Muller

AGRADECIMIENTOS

Los autores manifiestan su agradecimiento:

A Dios por permitirnos este destino.

Al Ingeniero Mario Fernando Solarte por su amistad, consejo y guía incesante durante este proyecto.

Al Ingeniero Carlos Enrique Serrano por su colaboración y valiosos aportes.

A los Ingenieros Julio Ariel Hurtado y Siler Amador Donado por entregarnos su abierta colaboración.

A Ricardo Lombana por su amistad, su conocimiento y la asesoría brindada en el desarrollo de aplicación.

TABLA DE CONTENIDO

1. INTRODUCCIÓN.....	1
2. MARCO TEÓRICO.....	4
2.1 COMPONENTE METODOLÓGICO	4
2.1.1 Proceso Unificado de Desarrollo de Software – RUP	4
2.1.2 Método de Diseño Hipermedia Orientado a Objetos - OOHDM	5
2.1.3 Modelo para Construcción de Soluciones	6
2.2 ESTÁNDARES PARA EL ASEGURAMIENTO DE LA CALIDAD DE SOFTWARE.....	7
2.2.1 Estándares para producto.....	8
2.2.2 Estándares para proceso.....	9
2.3 COMERCIO ELECTRÓNICO	12
2.3.1 Sistemas de procesamiento de transacciones	12
2.3.2 Secure Socket Layer (SSL)	14
2.3.3 Servidores seguros	15
2.3.4 Agentes en comercio electrónico.....	15
3. DESCRIPCIÓN DEL MODELO	17
3.1 FUNDAMENTOS DEL MICACE	17
3.2 ESTRUCTURA PARA DESCRIPCIÓN DEL SISTEMA DE COMERCIO ELECTRÓNICO ...	17
3.3 MODELO DEL PROCESO DE DESARROLLO	18
3.3.1 Fases de referencia	19
3.3.2 Estructura de referencia para cada fase.....	20
3.4 MODELO DE REFERENCIA ORGANIZACIONAL	21
4. DESCRIPCIÓN DEL PROCESO DE DESARROLLO.....	23
4.1 ESTUDIO PRELIMINAR.....	23
4.1.1 Descripción general de la fase.....	23
4.1.2 Etapas de referencia.....	24
4.2 FORMULACIÓN DEL PROYECTO.....	46
4.2.1 Descripción general de la fase.....	47
4.2.2 Etapas de referencia.....	48
4.3 CONSTRUCCIÓN DEL SISTEMA	70
4.3.1 Descripción general de la fase.....	71
4.3.2 Etapas de referencia.....	72
4.4 VALIDACIÓN DEL SISTEMA	87
4.4.1 Descripción general de la fase.....	87

4.4.2 Etapas de referencia.....	88
5. DESCRIPCIÓN DE LA APLICACIÓN	95
6. CONCLUSIONES Y RECOMENDACIONES.....	99
BIBLIOGRAFIA.....	101

INDICE DE FIGURAS

Figura 2.1 Fases y objetivos de RUP	5
Figura 3.1. Estructura patrón para cada Fase de Referencia.....	21
Figura 4.1 Plantilla para plan de gestión de configuración	32
Figura 4.2 Plantilla para la descripción de escenarios para la Formulación del Problema	55
Figura 4.3 Capas propuestas por el Patrón Layers	57
Figura 5.1 Diagrama de Casos de Uso Esenciales del sistema.....	96
Figura 5.2 Diagrama de despliegue del sistema.....	97
Figura 5.3 Interfaz principal del sistema	98

INDICE DE TABLAS

Tabla 3.1 Estructura para Descripción del Sistema de Comercio Electrónico	18
Tabla 3.2 Descripción de las Fases de Referencia	20
Tabla 3.3 Estructura del Grupo de gestión	21
Tabla 3.4 Estructura del grupo de control.....	22
Tabla 3.5 Estructura del grupo de desarrollo.....	22
Tabla 4.1 Notación UML para actores y casos de uso de negocio	35
Tabla 4.2 Notación de modelado UML para trabajadores y entidades de negocio.....	36
Tabla 4.3 Plantilla para la descripción de escenarios a nivel del Estudio Preliminar	37
Tabla 4.4 Plantilla para la descripción de requisitos.....	39
Tabla 4.5 Plantilla para la descripción de riesgos	43
Tabla 4.6 Categorización de recursos computacionales	44
Tabla 4.7 Descripción de las clases de análisis propuestas por el patrón Boundary-Control-Entity.	59
Tabla 4.8 Plantilla para la descripción de clases de análisis.....	61

1. INTRODUCCIÓN

Pocas personas pueden poner en tela de juicio la importancia de Internet y la World-Wide Web en nuestras vidas. En la actualidad se ha presenciado un incremento gigantesco de aplicaciones sobre plataforma web que permiten a las personas solventar muchas de las necesidades básicas del sistema de vida tales como la información, la educación y el entretenimiento. El auge de esta tendencia y el desarrollo paralelo de las tecnologías de la información, han producido que los sectores productivos y de negocios de la sociedad mundial piensen seriamente en volcar rápidamente sus estrategias de negocio para poder promocionar, vender y distribuir productos y servicios a las personas a través de sistemas de comercio electrónico basados en web sobre Internet. No cabe duda que la velocidad es importante en el comercio electrónico, pues el cambio tecnológico hace que la competencia sea mucho más rápida en línea que fuera de línea. Las empresas no se pueden permitir permanecer inactivas mientras sus competidores en línea están aprendiendo con la práctica [ONU 01].

En este panorama, es muy importante para una empresa o grupo que pretende incursionar en el comercio electrónico, tener un modelo de negocio consistente y que además este se soporte sobre aplicaciones que permitan manejar eficientemente la complejidad inherente de los procesos de negocio y el crecimiento producto del cambio constante de los mercados, para así de sacar los mejores resultados. Sin embargo, el desarrollo apresurado de sistemas de comercio electrónico basado en métodos informales de desarrollo de software y sumado a esto la falta de planeación para la construcción de tales sistemas, ha terminado por inundar Internet de negocios en quiebra o sin perspectivas de evolución.

Es claro que en el ámbito local las tendencias mundiales sobre comercio electrónico no son indiferentes. Son muchas las pequeñas y medianas empresas (PYMES) de la región que tienen grandes expectativas sobre los nuevos canales que les ofrece Internet y la web. Como resultado de esto se ha presentado una considerable demanda de desarrollo de aplicaciones web para catapultar negocios e imágenes empresariales más allá de los nichos tradicionales de mercado, aprovechado además las ventajas que pueden ofrecer actualmente el uso generalizado de Internet y las políticas gubernamentales que promueven la expansión de las tecnologías de la información. Sin embargo muchos de los responsables en la construcción de sistemas de comercio electrónico, para este tipo de empresas, son desarrolladores web “tradicionales”, que si bien contribuyen con su experiencia en el manejo de tecnologías, muy pocas veces comprenden la complejidad y las responsabilidades tan críticas que estos sistemas comprometen.

La necesidad crear sistemas de comercio electrónico que se desempeñen eficientemente para lograr los objetivos de negocio de las organizaciones según sus condiciones específicas, y que además se adapten sin traumas al veloz mundo de Internet, llevan a pensar en definir estrategias formales que dirijan su desarrollo. Ante esta problemática, se propone al Modelo Integral para Construcción de Aplicaciones de Comercio Electrónico MICACE, como Modelo de Referencia útil en el desarrollo sistemas de comercio electrónico sobre plataforma web, con el fin de generar soluciones eficientes y bajo condiciones de calidad.

El propósito fundamental del MICACE es generar un Modelo Integral para la Construcción de Aplicaciones en el área del Comercio Electrónico, que propicie el desarrollo de soluciones de alta calidad, bajo una correcta administración del trabajo y el esfuerzo de equipo humano, y con conceptos de mejoramiento continuo del proceso y el producto de software.

El MICACE ejercita y adapta el Modelo para la Construcción de Soluciones (M.C.S) elaborado por el Grupo de Ingeniería Telemática (GIT), además de seguir las directrices planteadas por el Proceso Unificado para desarrollo de software (RUP). Complementariamente incorpora los conceptos más importantes de aseguramiento de la calidad del proceso de software propuestos por el CMM-SW (Capability Maturity Model for Software) versión 1.1, CMMI (Capability Maturity Model Integration) versión 1.0; y los correspondientes para el aseguramiento de la calidad del producto dados por las normas ISO/IEC 9126 (Draft) e IEEE 1061. Finalmente el MICACE incorpora algunos conceptos relevantes en el campo de aplicaciones web planteados por OOHDM (Método para Diseño Hipermedia Orientado a Objetos).

Paralelamente el MICACE contribuye significativamente a los objetivos establecidos para el Programa No. 3, Generación de Referencias Metodológicas para el Desarrollo de Soluciones, del Plan de Trabajo 1998-2003 del GIT, constituyendo también un aporte significativo para el nodo de Infraestructura Metodológica del Departamento de Telemática.

La monografía se encuentra distribuida de la siguiente forma:

El primer capítulo está constituido por la introducción que contiene la presentación de los aspectos más importantes como resultado del Trabajo de Grado.

En el segundo capítulo se describen aspectos tanto conceptuales como metodológicos que sirvieron como guía y soporte para la construcción del MICACE. Este se ha dividido de la siguiente forma: Componente Metodológico (M.C.S, RUP, OOHDM), Estándares para aseguramiento de la calidad del software (CMM-SW, CMMI, ISO 9000, ISO/IEC 9126, IEEE 1061), y Comercio

electrónico, que contiene algunos conceptos importantes considerados en aplicaciones de comercio electrónico sobre plataforma web de última generación.

El tercer capítulo presenta la Descripción General de Modelo. Este comprende los pilares conceptuales del Modelo, los componentes del Proceso de Desarrollo, y el Modelo de Referencia Organizacional.

El cuarto capítulo contiene la Descripción Detallada de Proceso del Proceso de Desarrollo. Esta plasma la arquitectura del Proceso de Desarrollo, constituido por la Fases de Referencia, la infraestructura de Actividades, y los Productos de Trabajo.

El capítulo quinto contiene una descripción general del sistema de comercio electrónico construido para un cliente externo con el fin de validar y ejercitar el MICACE.

Finalmente el capítulo sexto contiene las conclusiones y recomendaciones del trabajo.

Como anexos se tienen los siguientes documentos:

Anexo A – Marco Conceptual. Contiene una descripción más detallada de algunos aspectos conceptuales que sirvieron como guía y soporte para la construcción del MICACE y que por su extensión no fueron incluidos en detalle en la monografía.

Anexo B – Descripción del sistema de comercio electrónico desarrollado. Contiene los productos obtenidos en cada fase del desarrollo de la aplicación de comercio electrónico que se construyó para validar y ejercitar el MICACE.

2. MARCO TEÓRICO

En este capítulo se describe aspectos tanto metodológicos como conceptuales que sirvieron como guía y soporte para la construcción del Modelo Integral para la Construcción de Aplicaciones de Comercio Electrónico (MICACE). Los componentes en los cuales está dividido el Marco Teórico son los siguientes:

- **Componente Metodológico:** contiene específicamente las referencias metodológicas de soporte utilizadas para la construcción del modelo: RUP (Proceso Unificado para Desarrollo de Software), OOHDM (Método de Diseño Hipermedia Orientado a Objetos), y MCS (Modelo para Construcción de Soluciones).
- **Estándares para aseguramiento de la calidad de software:** este comprende una revisión de los modelos de calidad tanto para el producto de software como para el proceso. Aquí se encuentra consignada una breve descripción de las normas de organismos internacionales altamente aceptados como la ISO, ISO/IEC, IEEE y CMU/SEI.
- **Comercio Electrónico:** se compone de la definición de conceptos básicos en las aplicaciones de comercio electrónico de última generación tales como: los sistemas de procesamiento de transacciones, conceptos de seguridad, servidores seguros y SSL (Secure Socket Layer) además del concepto de agentes aplicado a este tipo de sistemas.

2.1 COMPONENTE METODOLÓGICO

2.1.1 Proceso Unificado de Desarrollo de Software – RUP

El Proceso Unificado de Rational propuesto por “Rational Software Corporation” es el resultado del esfuerzo de las tres últimas décadas en desarrollo de software y de la experiencia de sus creadores Ivar Jacobson, Grady Booch y James Rumbaugh [Jacobson 99].

Las características principales de RUP son:

- *Proceso iterativo.* Propone una comprensión del problema y un crecimiento de la solución incrementales a través de refinamientos sucesivos.
- *Dirigido por casos de uso.* Las nociones de casos de uso y escenarios se utilizan para guiar el flujo de procesos desde la captura de requisitos hasta las pruebas y para proporcionar caminos que se pueden reproducir durante el desarrollo del sistema.
- *Centrado en la arquitectura.* Utilizándola como producto primario de conceptualización, construcción, gestión y desarrollo del sistema bajo construcción.
- *Orientado a objetos.* Los modelos de RUP se basan en los conceptos de objetos y clase y las relaciones entre ellos, y utilizan UML como la notación común.

El proceso de desarrollo está dividido en cuatro fases, intervalos de tiempo entre dos hitos importantes, durante cada fase se cumple un conjunto de objetivos, se completa artefactos y se toma decisiones sobre cuando pasar a la siguiente fase. Las cuatro fases de RUP y sus principales objetivos se pueden ver en la figura 2.1:

Fases	Objetivos
Inicio (Inception)	<ul style="list-style-type: none"> • Justificar la puesta en marcha del proyecto a través del análisis del negocio • Esbozar una arquitectura candidata para el sistema.
Elaboración (Elaboration)	<ul style="list-style-type: none"> • Establecer una arquitectura de referencia para la construcción del sistema. • Definir una agenda estable para el desarrollo del proyecto
Construcción (Constuction)	<ul style="list-style-type: none"> • Desarrollar la capacidad operativa inicial del sistema
Transición (Transition)	<ul style="list-style-type: none"> • Validar la capacidad operativa inicial por parte de los usuarios finales

Figura 2.1 Fases y objetivos de RUP

2.1.2 Método de Diseño Hipermedia Orientado a Objetos - OOHDM

OOHDM [Rossi 96] propuesta por D. Schwabe y G.Rossi, surge de la evolución en el desarrollo de modelos hipermedia y de la necesidad de satisfacer el creciente avance de las aplicaciones en éste campo.

OOHDM, establece el desarrollo de un aplicación hipertexto como un proceso de cuatro fases, diseño conceptual, diseño navegacional, diseño de interfaz abstracta e implementación, en el que se combinan diferentes estilos de desarrollo: incremental, iterativo y prototipado. Las tres primeras fases son de diseño, en las que se obtiene un conjunto de modelos orientados a objeto que describen la aplicación o documento hipertexto que será construido en la última fase.

El modelado conceptual es la actividad donde se hace el análisis del dominio de la aplicación. Engloba todo el universo de informaciones relevantes para la aplicación siendo considerada, aunque apenas un subconjunto de esas informaciones sea considerado posteriormente en su implementación. El resultado del modelado conceptual de una aplicación consiste en un esquema conceptual con objetos del dominio de la aplicación (clases, relaciones y subsistemas).

El modelado navegacional define las informaciones que serán presentadas y la navegación entre ellas. Se puede definir el modelado navegacional a partir de un modelo conceptual ya definido o directamente sin que la actividad de modelado conceptual haya sido ejecutada. En el modelado navegacional se especifica qué contextos navegacionales y objetos navegacionales (nodos, ejes, clases en contexto, estructuras de datos, etc.) serán vistos por el usuario y se producen las siguientes salidas: un esquema navegacional con la definición de los objetos navegacionales; un esquema de contextos identificando los contextos navegacionales; fichas que especifican todos los objetos creados en el modelado navegacional (objetos navegacionales y contextos navegacionales) [Mercerat 02].

El modelado de interfaz abstracto es el resultado de la especificación de los objetos de interfaz que el usuario percibirá. OOHDM utiliza Abstract Data Views (ADVs) para modelar los aspectos estáticos de la interfaz del usuario mientras que los aspectos dinámicos de interfaz del usuario se modelan con una técnica basada en Diagramas de estado [Valderas 02].

La implementación constituye la última fase de la metodología OOHDM y es aquí donde los modelos navegacional y de interfaz se expresan en objetos particulares con sus correspondientes contenidos textuales, sonoros, visuales, etc. y sus posibilidades de navegación. Debido al carácter abstracto del diseño, no es necesario que en esta fase se trabaje con entornos de construcción orientados a objetos, a pesar de su uso durante las anteriores fases de desarrollo, y puede hacerse fácilmente en otros entornos que permitan trabajar con el lenguaje HTML. [Martínez 00]

2.1.3 Modelo para Construcción de Soluciones

El Modelo para Construcción de Soluciones se concibe como “una referencia metodológica esencial para cualquier proyecto cuyo propósito sea construir una solución de calidad, oportuna y con costos competitivos y sobretodo que pretenda contribuir a la creación y enriquecimiento de la base de conocimiento/experiencia institucional” [Serrano 01]. Su objetivo es el de mejorar la calidad de los procesos de desarrollo que se realizan en la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca buscando esencialmente incrementar la capacidad competitiva de los integrantes del equipo humano.

El modelo está constituido por tres componentes esenciales, la estructura para la descripción del sistema, constituida por tres modelos que describen los niveles de abstracción del sistema solución, el modelo del proceso de desarrollo, que define las fases de referencia y los criterios de transición entre las mismas y el modelo de organización del talento, que sirve como referencia para la organización del recurso humano del proyecto.

En la estructura para la descripción del sistema/solución, el modelo recomienda la adopción de cuatro técnicas básicas de modelamiento, la orientación a objetos, la estructuración por niveles, el modelamiento “outside-in”, y la participación para la minimización de interfaces. Este macrocomponente se divide en los modelos de establecimiento de responsabilidades, descripción del sistema e implementación del sistema, dentro de los cuales se definen componentes que se deben generar en el proceso de desarrollo de la solución.

El Modelo del proceso de desarrollo tiene como principal paradigma la técnica de desarrollo incremental que permite ofrecer al cliente/usuario un primer prototipo de la solución, a partir del cual se realizan incrementos y mejoras hasta llegar a una solución ideal que satisfaga a fondo las necesidades y requerimientos iniciales. Está dividido en cuatro fases: estudio de prefactibilidad, formulación del proyecto, ejecución del proyecto y validación de la solución.

Dentro del modelo de organización del talento se propone la conformación de dos equipos, uno de gestión y otro de desarrollo. El primero conformado por el director, asesor y consultor, siendo estos actores representados por uno o más miembros del equipo. Dentro del equipo de desarrollo se incluye en un primer nivel el analista del sistema, el arquitecto del sistema y el ingeniero de pruebas y en un segundo nivel el ingeniero de casos de uso y el ingeniero de subsistema.

2.2 ESTÁNDARES PARA EL ASEGURAMIENTO DE LA CALIDAD DE SOFTWARE

Los sistemas de software tradicionales que se utilizan en diferentes campos de trabajo del hombre en la actualidad han llegando a niveles de complejidad y dimensiones sorprendentes. Algunos de

estos incluyen sistemas de información, sistemas de tiempo real, y en general aplicaciones software telemáticas que involucran conceptos tan complejos como procesamiento distribuido, concurrencia, interoperabilidad, entre otros. Quizá algunas de las aproximaciones para manejar esta situación vayan desde las abstracciones generadas por el paradigma de orientación a objetos, la tecnología basada en componentes, hasta la definición de algunos métodos de desarrollo que permiten afrontar la construcción de sistemas complejos bajo lineamientos más o menos formales. A pesar de esto, en el contexto actual se viene presentado un interrogante sobre cómo garantizar y bajo qué condiciones evaluar que los sistemas software cumplan eficientemente con las altas responsabilidades para las que fueron concebidos, en esta coyuntura aparece el concepto de *calidad del software*.

En vista de que el concepto de *calidad* en el mundo del software puede ser visto e interpretado muy libremente, organismos internacionalmente reconocidos vienen prestando atención en generar marcos de referencia para tratar este aspecto. Algunos de estos esfuerzos se han traducido en la generación de *estándares de calidad* que afrontan diversos conceptos relacionados con la construcción y evaluación del software. A continuación se presentan las definiciones de los dos tipos de estándares más difundidos.

2.2.1 Estándares para producto

“Los estándares para producto definen modelos de calidad general para los productos de software o características específicas a ser consideradas mientras se desarrolla o evalúa el software” [ESSISCOPE 02].

A continuación se mencionan los estándares de referencia que se tuvieron en cuenta para la creación del MICACE. Básicamente se emplearon dos estándares pertenecientes a la ISO/IEC (International Standards Organization/International Electrical Engineering Commission) y a la IEEE (Institute of Electrical and Electronic Engineers).

2.2.1.1 ISO/IEC 9126

Su objetivo es proveer un esquema para la evaluación de la calidad de productos de software. El estándar define un modelo de calidad el cual es aplicable a cualquier clase de software. El estándar declara seis características de calidad del producto de software y en el anexo del mismo se sugiere una lista de subcaracterísticas. Las características que define el estándar son:

- *Funcionalidad*: que funciones requeridas están disponibles en el software
- *Confiabilidad*: que tan confiable es el software desarrollado
- *Usabilidad*: que tan fácil de usar es el software
- *Eficiencia*: como es de eficiente el software
- *Mantenibilidad*: que tan fácil es de modificar el software
- *Portabilidad*: que tan fácil es de transferir el software a otro ambiente

La anterior es la definición hecha por la primera versión de estándar [ISO/IEC 91], sin embargo en la actualidad se tiene una versión bajo revisión, que se compone de cuatro documentos: ISO/IEC 9126-1 que contienen las características y subcaracterísticas de calidad, ISO/IEC 9126- 2 que contiene la definición de las métricas externas, ISO/IEC 9126-3 que contiene la definición de las métricas internas, y el ISO/IEC 9126- 4 que contiene la agrupación de métricas de calidad en uso [Bevan 99].

2.2.1.2 IEEE 1061

El estándar IEEE 1061 [IEEE 92], es una metodología para el establecimiento de los requisitos de calidad y su identificación, implementación, análisis y validación del proceso y el producto de software, para lo cual se definen métricas. La metodología abarca todo el ciclo de vida de desarrollo, el estándar presenta ejemplos de métricas pero no prescribe métricas específicas. Esta filosofía que no impone métricas es importante ya que según el estándar las organizaciones son responsables de establecer las métricas que crean más convenientes para sus aplicaciones, mientras que la metodología sea seguida y las métricas sean validadas.

Al igual que el ISO 9126, el estándar IEEE 1061 propone una lista de características y subcaracterísticas que sirven como guía para la evaluación de la calidad del producto de software, estas están consignadas en el anexo A del estándar (informativo), los cuales se denominan *factores* y *subfactores* respectivamente, aunque los contenidos y las definiciones difieren algunas veces de forma considerable debido a la orientación de la norma.

Se puede encontrar una ampliación de estos estándares para producto en el Capítulo 2 del Anexo A – Marco Conceptual.

2.2.2 Estándares para proceso

“Los estándares para proceso de software definen los procesos y las actividades necesarias para producir software de calidad. Ellos son usados para administrar y controlar el proceso utilizando procedimientos y métodos estandarizados” [ESSISCOPE 02]. Los estándares que se usaron pertenecen conjuntamente a los siguientes organismos, SEI (Software Engineering Institute) y el CMU (Carnegie Mellon University).

2.2.2.1 CMM-SW (Capability Maturity Model for Software)

El modelo de capacidad de madurez para software [Paulk 93], provee a las organizaciones de software una guía de cómo aumentar el control de sus procesos de desarrollo y mantenimiento de software. El CMM - SW fue diseñado para guiar a las organizaciones de software en la selección de estrategias para el mejoramiento de procesos mediante la determinación de la madurez de los procesos actuales e identificando los elementos más críticos de la calidad de software y del mejoramiento de procesos.

El modelo define al *Nivel de Madurez* como un nivel de capacidad de los procesos de software y procesos administrativos que posee una organización para crear software. El modelo define cinco Niveles de Madurez que en su orden de capacidad son: Nivel 1- Inicial (ad-hoc), Nivel 2 – Repetible, Nivel 3 – Definido, Nivel 4- Administrado, Nivel 5 – Optimizado.

Cada nivel, excepto por el primero, está compuesto por *Áreas Claves de Proceso* (KPAs), las cuales indican áreas en las que una organización debe enfocarse para mejorar su proceso de software. Cada KPA identifica un conjunto de actividades relacionadas que cuando son ejecutadas colectivamente logran unas metas consideradas importantes para incrementar la capacidad del proceso. Cada área clave de proceso está organizada en cinco *Características Comunes* los cuales son los atributos que indican cuando la implementación e institucionalización de la KPA es efectiva, repetible y duradera. Las Características Comunes especifican *Prácticas Claves* las cuales describen la infraestructura y actividades que contribuyen a una implementación e institucionalización más efectiva de la KPA.

2.2.2.2 CMMI (Capability Maturity Model Integration)

El propósito de CMMI es proveer un guía para la mejora de los procesos de la organización y su habilidad para administrar el desarrollo, adquisición, y mantenimiento de productos y servicios. CMMI provee prácticas dentro de una estructura que ayuda a su organización a determinar su madurez organizacional y su capacidad en el área de procesos, estableciendo prioridades para

mejora, y guía la implementación de esas mejoras. CCMI fue concebido como una iniciativa para agrupar varios CMMs dentro de un conjunto integrado de modelos. La fuente de modelos que sirvieron como base para el CMMI fueron: El CMM para software V2.0(Draft C), EIA-731 Systems Engineerig, y el IPD CMM (IPD) V0.98a.

El CMMI [CMMIPDT 00] tiene igual número de *Niveles de Madurez* que los definidos por el CMM – SW, los siguientes son sus nombres: Nivel 1- Inicial, Nivel 2 – Administrado, Nivel 3 – Definido, Nivel 4 – Cuantitativamente Administrado, y Nivel 5 – Optimizado. Cada nivel de madurez esta compuesto por *Áreas de Proceso*, para las cuales se definen *Metas Genéricas* y *Metas Específicas*. Las Metas Genéricas son aplicables a todas las Áreas de Proceso mientras que las Metas Específicas se aplican solo un Área de Proceso y direccionan características únicas que describen que debe ser implementado para satisfacer el propósito del Área de Proceso. Debido a esto, se definen en el modelo *Prácticas Genéricas* que se aplican a cualquier Área de Proceso y que se organizan en cuatro *Características Comunes*, así como *Prácticas Específicas* que especifican actividades importantes en el logro del las Metas Específicas.

2.2.2.3 ISO 9000 series

Las Normas internacionales de la familia ISO 9000, tiene como objetivo proporcionar un sistema de normas genéricas, las cuales son aplicables a toda una gama de industrias y actividades. Las normas ISO 9000 son independientes de cualquier industria o sector económico específico, ya que estas proporcionan orientación con relación a la administración de la calidad y los requisitos generales para el aseguramiento de la misma. La aplicación de estas normas se traduce a una empresa, en un *sistema de calidad*, el cual va a generar la certificación de la empresa ante el mercado que maneje, por medio de un estudio (serie de auditorias) que realiza la ISO a las empresas que implantan su sistema de calidad, con el fin de que se reconozca nacional e internacionalmente como empresas que generan productos de calidad certificados [Ledezma 00].

La serie de normas ISO 9000 se divide en Normas contractuales, las cuales están orientadas a tres tipos específicos de empresas y son impuestas por los clientes; las Normas no contractuales permiten ayudar a las empresas a establecer el sistema de calidad. La primera clasificación se compone de las normas ISO 9001 (empresas que diseñan o modifican el producto), ISO 9002 (empresas que solo producen), e ISO 9003 (para empresas que solo requieren control de calidad y no diseñan ni producen). La segunda clasificación se compone de la norma ISO 9000-1 (Guía de selección y uso), ISO 9000-2 (Guía genérica para la aplicación de las normas contractuales), e ISO 9000-3 (Guía para la aplicación de la ISO 9001 en el desarrollo suministro y mantenimiento de software).

NTC – ISO 9000-3: Norma para Administración y Aseguramiento de la Calidad. Parte 3: Directrices para la aplicación de ISO 9001 para el desarrollo suministro y mantenimiento de Software. Esta es la homologación (traducción al idioma del país, NTC – Norma Técnica Colombiana) de la norma ISO 9000-3.

La Norma esta compuesta por tres partes que son, Estructura del sistema de calidad, Ciclo de vida, y Procesos de soporte. Cada una de estas partes de la norma describe una infraestructura de actividades. En la Estructura del sistema de calidad se puede observar el manejo gerencial que se debe presentar en el aseguramiento de la calidad de los productos software que desarrollo la empresa; El Ciclo de vida describe las actividades relacionadas con la calidad que se deben implementar dentro del ciclo de vida (La Norma NTC-ISO 9000-3, esta diseñada para la aplicación en forma independiente del ciclo de vida que se utilice); Los Procesos de soporte implican procedimientos tales como la administración de configuración, aseguramiento de la calidad, entrenamiento, entre otros.

Se puede encontrar la estructura de estas actividades de aseguramiento de la calidad en el Capítulo 2 del Anexo A – Marco Conceptual.

2.3 COMERCIO ELECTRÓNICO

Existen diversas definiciones sobre el significado de comercio electrónico dependiendo de los organismos o individuos que las proponen y los dominios específicos en los cuales se mueven, tales como entes jurídicos, empresas, desarrolladores de software, entre otros. Debido a esta coyuntura los gestores del MICACE comparten la siguiente definición, ya que se acomoda eficientemente a los lineamientos del proyecto, “se considera al comercio electrónico como la utilización de la informática y las telecomunicaciones para canalizar los flujos de información y transacciones de negocio existentes entre una empresa y sus interlocutores de negocio habituales (consumidores, proveedores, entidades financieras, transportistas, etc.)” [Stephens 00].

2.3.1 Sistemas de procesamiento de transacciones

Uno de los factores más críticos para del comercio electrónico ha sido la seguridad y la flexibilidad en las transacciones. Los sistemas de procesamiento de transacciones han evolucionado desde hace casi una década en el afán de que el comprador, el vendedor y la entidad bancaria tengan plena confianza y satisfacción al momento de realizar transacciones a través de Internet.

En la actualidad estos sistemas están orientados al uso de la tarjeta de crédito como medio de autenticación en la red, pero sistemas como el del FSTC (Financial Services Technology Consortium), consorcio americano de bancos, organizaciones gubernamentales y empresas tecnológicas creado en 1995, se valía de una tarjeta inteligente a través de la cual el comprador generaba una firma digital que era enviada al vendedor y éste a su vez la firmaba nuevamente para enviarla al banco. First Virtual, en cambio ya trabajaba con los datos de la tarjeta de crédito, pero éstos eran enviados al sistema a través de medios off-line, sin pasar por la red.

La gran oferta de productos y de servicios en Internet exigió que los medios para realizar transacciones dieran al comprador una disponibilidad inmediata de medios de pago. Cybercash, que se encuentra en funcionamiento desde 1995, es el pionero en los sistemas para transacciones para Internet. Para su funcionamiento tanto el comprador como el vendedor deben instalar un software llamado "wallet". El software encripta todos los datos, realiza un registro de todas las transacciones y se encuentra protegido mediante contraseña. El vendedor envía una factura al comprador firmada digitalmente a través de una llave privada. El comprador autoriza la compra y devuelve al vendedor un mensaje de confirmación firmado digitalmente y encriptado para Cybercash. A este mensaje el vendedor añade los datos de la compra y lo envía a Cybercash quien lo desencripta, comprueba que coincidan los datos de ambos y lo envía a la red financiera para hacer la transferencia.

Cybercash dio origen a SET (Transacciones Electrónicas Seguras). En SET se definen tres entidades independientes: cliente, vendedor y la pasarela de pago, que se interconectan directamente por Internet, haciendo el vendedor de puente entre el cliente y la pasarela de pago. Previamente a cualquier comunicación entre ellos, todas las entidades deben haber obtenido un certificado digital válido a través de la Autoridad de Certificación (CA) adecuada. La pasarela de pago permite la conexión desde Internet con las redes bancarias. Entre las ventajas de SET están la confidencialidad que se logra al separar los datos financieros de la descripción de la compra ya que ni el vendedor ni el banco tienen acceso a datos que no le son imprescindibles, la integridad de todos los mensajes referentes a la descripción de la compra y a datos financieros transmitidos entre el comprador, el vendedor y la pasarela de pagos, ya que van firmados digitalmente incluso tras finalizar la conexión, además todos los participantes están certificados por una Autoridad Certificadora única, lo que imposibilita cualquier tipo de robo de identidad así como la utilización de números de tarjeta de crédito robados [Conesa 99].

Actualmente SSL responde eficazmente a las necesidades de pago a través de Internet, por su seguridad, estandarización y portabilidad, por lo que se ha convertido en la forma casi general para realizar transacciones en línea.

2.3.2 Secure Socket Layer (SSL)

El protocolo SSL fue desarrollado por Netscape para permitir confidencialidad y autenticación en Internet. SSL opera como una capa adicional entre Internet y las aplicaciones, esto permite que el protocolo sea independiente de la aplicación, siendo posible utilizar otras aplicaciones como FTP y Telnet, además de HTTP. Actualmente es el medio más utilizado para realizar transacciones seguras y viene incluido por defecto en la mayoría de navegadores, lo que permite una implementación muy sencilla del sistema de pago ya que el cliente puede comenzar a comprar sin tener que realizar ningún proceso de autenticación previa.

El proceso para establecer una comunicación segura utilizando SSL comienza con una solicitud hecha por un cliente a un servidor que soporte el protocolo, típicamente a través de una URL. Una vez hecha la solicitud, el cliente y el navegador comienzan a negociar la conexión SSL, proceso conocido como SSL Handshake. Durante éste, se cumplen varios propósitos, se hace autenticación del servidor y opcionalmente del cliente, se determina qué algoritmos de criptografía serán utilizados y se genera una llave secreta para ser utilizada durante el intercambio de mensajes subsiguientes durante la comunicación.

Una vez se ha establecido un canal de transmisión seguro SSL, es posible el intercambio de datos. Cuando el servidor o el cliente desea enviar un mensaje al otro, se genera un digest (utilizando un algoritmo de hash de una vía que previamente se ha acordado durante el handshake), se encripta el mensaje y el digest y se envían, cada mensaje es verificado utilizando el digest. Al terminar una sesión de SSL, generalmente la aplicación presenta un mensaje advirtiendo que la comunicación no es segura y confirma que el cliente efectivamente desea abandonar la sesión. Una función de hash genera a partir del mensaje que se desea enviar cierta información, de tamaño habitualmente mucho menor, que depende directamente de mensaje. Esa función de hash tiene que cumplir una serie de requisitos: debe minimizar la probabilidad de colisión, es decir, debe ser muy improbable que dos mensajes distintos den el mismo resultado al aplicar la función de hash y debe ser prácticamente imposible de invertir, es decir, de generar el mensaje original a partir del resultado de aplicar la función de hash.

Para crear un sistema de pago electrónico basado en SSL es necesario conseguir un certificado electrónico para el Vendedor, generalmente se obtiene de la empresa VeriSign (filial de RSA Data Security Inc. y principal Autoridad Certificadora mundial). VeriSign está considerada por Microsoft y Netscape como autoridad certificadora de confianza por lo que por defecto viene activada en sus respectivos navegadores.

Una vez realizado el pago, el vendedor obtiene el PIN de la tarjeta de crédito del cliente, por lo que debe estar provisto de algún método que permita enviar estos datos a una entidad financiera que sea capaz de realizar la transferencia bancaria. Generalmente estas transacciones entre el vendedor y el banco se hacen a través de un protocolo privado acordado entre los dos [Netscape 00].

2.3.3 Servidores seguros

Para hablar de servidores seguros primero es necesario hablar de los certificados de seguridad. Un Certificado de Seguridad es un conjunto de documentos electrónicos emitidos por una entidad certificadora, que permiten encriptar la información transmitida e identificar a la fuente de dicha información. Para que el certificado sea fiable, la entidad certificadora debe ser un organismo de confianza capaz de certificar la procedencia de la información, que es lo que en cierta medida proporciona la seguridad. Dos de las entidades certificadoras más conocidas son Thawte y Verisign. [Idómine 02]

Para el caso específico de Colombia, Certicámara [Certicámara 02] define a los certificados de seguridad como documentos digitales de identidad emitidos a un individuo. La emisión está bajo la responsabilidad por una entidad de certificación debidamente autorizada por la Superintendencia de Industria y Comercio (Ley 527 Art. 29-34). Esta entidad garantiza los datos contenidos en el citado documento relativos a una persona, ya sea natural o jurídica (empresas). Este documento digital vendría a ser el equivalente a un Documento de Identidad, Licencia, pasaporte o carné de empresa. Mediante un conjunto de claves asociadas a una identidad, un Certificado sirve para identificarse ante terceros, y previene suplantación de la identidad en Internet. Este hecho lo convierte en herramienta clave para la identificación de las partes contratantes en el comercio electrónico. Una vez obtenido el certificado de seguridad se procede a instalar un *servidor virtual seguro* en el dominio. Este servidor debe poseer una dirección IP propia lo cual garantiza la utilización del puerto 443 para las transacciones SSL que es el usado por defecto. Esto permite cargar sin inconvenientes las páginas contenidas en el servidor.

2.3.4 Agentes en comercio electrónico

Los agentes de software son programas que actúan con cierta flexibilidad y autonomía, normalmente en representación de alguien, y normalmente en un contexto digital más o menos estructurado [Iglesias 98].

El agente debe tener una cierta autonomía respecto al propio dueño o al entorno, por ejemplo, que entre o salga de un lugar virtual sin que haya intervención humana directa. El agente de software además, debe subsistir o permanecer después de la realización de esas acciones o debe poder realizarlas desplazándose sucesivamente a distintos lugares. Esas acciones se realizan en un contexto social en el que el agente es capaz de interactuar con otros agentes o con seres humanos. Para lograr sus propósitos, esas acciones que el agente realiza se deben adecuar a las circunstancias específicas y cambiantes en que se realicen ("situacionalidad").

La presencia virtual de los agentes permite concebir formas de interacción totalmente nuevas. Por ejemplo, se puede aprovechar que un agente de software está siempre alerta para establecer un monitoreo permanente de múltiples sitios de la red, o bases de datos o puntos de negocio de forma que al darse las condiciones propicias se tomen las acciones que deseamos. O también, por ejemplo, es posible "clonar" un agente para que participe de manera simultánea en varias negociaciones e, inclusive, que conforme algunas de esas negociaciones avancen, nuevos clones vayan apareciendo para reforzar o retomar con nuevos elementos las negociaciones en curso. También se puede delegar en uno o varios agentes algunas de las labores más rutinarias o incómodas de las transacciones habituales: la exploración sistemática de alternativas, el filtraje de un número excesivo de opciones, el contacto con los vendedores, y retomar personalmente el hilo de esa transacción sólo cuando resulte más provechoso o conveniente hacerlo.

La incorporación de agentes al comercio electrónico es cada vez mayor y más especializada, logrando mayor calidad y personalización de las ofertas a los consumidores. Se puede citar como ejemplo de esto a Amazon.com, uno de los sitios de comercio en Internet mejor estructurados tecnológicamente y gracias a esto uno de los más rentables y estables en el mercado de Internet. En este sitio es posible recibir de forma personalizada las novedades editoriales que se adecuan al perfil propio que cada usuario va construyendo en cada nueva visita gracias a "Eyes", el agente implementado que se encarga de "escoger" los nuevos títulos que le puedan interesar a cada cliente de acuerdo a las consultas, intereses y pedidos que haya hecho anteriormente. También se puede referir a la casa de subastas electrónica FM96.5, un proyecto del Laboratorio Nacional de Informática Avanzada (LANIA) de México, donde es posible, que tanto agentes de software como personas, participen como compradores o como vendedores en subastas en línea (via Internet o un Intranet); y que lo hagan bajo condiciones de equidad, transparencia y vivacidad comparables e incluso mejores que las de las casas de subastas tradicionales.

Se puede encontrar más acerca de agentes en sistemas de comercio electrónico en el Capítulo 4 del Anexo A – Marco Conceptual.

3. DESCRIPCIÓN DEL MODELO

Los pilares conceptuales arquitectónicos que soportan al MICACE se encuentran basados en Modelo para la Construcción de Soluciones (MCS) [Serrano 01], por lo tanto se heredan los macro-componentes constitutivos de este último. A continuación se describen los aspectos generales que constituyen el MICACE.

3.1 FUNDAMENTOS DEL MICACE

Los fundamentos que dan soporte al MICACE son los siguientes:

- *Orientación al diseño:* Este concepto hace referencia a que el desarrollo principalmente esta enfocado a especificar y diseñar la funcionalidad del sistema de comercio electrónico, más que en su implementación. Se debe definir claramente que es lo que el sistema hace independiente de consideraciones de implementación particulares.
- *Descripción formal mediante UML:* Se utiliza esta técnica de descripción formal ya que provee un lenguaje de modelamiento que favorece la comunicación de las abstracciones del sistema de comercio electrónico entre los integrantes del equipo humano del proyecto y los clientes/usuarios.
- *Orientación a Objeto:* Ya que los sistemas de comercio electrónico pueden llegar a automatizar procesos ejecutados generalmente por agentes humanos, las técnicas Orientación a Objeto cobran una importancia definitiva ya que permite generar abstracciones acordes al dominio.

3.2 ESTRUCTURA PARA DESCRIPCIÓN DEL SISTEMA DE COMERCIO ELECTRÓNICO

El conjunto de modelos que el MICACE propone para describir el sistema/solución de comercio electrónico se puede observar en la tabla 3.1. Los modelos se toman como abstracciones del sistema de comercio electrónico, especificando a dicho sistema modelado desde una cierta perspectiva y en un determinado nivel de abstracción. Los modelos se usan como herramientas de comunicación para definir y describir la solución.

MODELOS DE ALTO NIVEL	COMPONENTES PRINCIPALES
	1. Modelo de Negocio 2. Modelo de Casos de Uso

<p>MODELO PARA EL ESTABLECIMIENTO DE RESPONSABILIDADES</p>	<p>3. Arquitectura de Referencia del sistema de comercio electrónico 4. Lista de Riesgos y Estrategias 5. Plan de Gestión de Configuración 6. Plan del Proyecto 7. Caso del Negocio</p>
<p>MODELO PARA DESCRIPCIÓN DEL SISTEMA</p>	<p>8. Modelo de diseño 9. Modelo Navegacional 10. Modelo de Despliegue 11. Modelo de pruebas</p>
<p>MODELO DE IMPLEMENTACIÓN DEL SISTEMA</p>	<p>12. Prototipo Operacional 13. Manuales de Usuario</p>

Tabla 3.1 Estructura para Descripción del Sistema de Comercio Electrónico

3.3 MODELO DEL PROCESO DE DESARROLLO

La descripción detallada del proceso de desarrollo propuesto por el MICACE, sus fases de referencia, el propósito de las fases, las técnicas, y productos de trabajo se pueden encontrar en el capítulo 4 de la monografía. Sin embargo, a continuación se describen las características principales que presenta el proceso:

- *Dirigido por Objetivos:* Se busca que los implicados en el desarrollo del trabajo se concentren principalmente en los objetivos más que en las actividades.
- *Iterativo e Incremental:* Es útil para manejar la complejidad de un proyecto se propone utilizar aproximaciones iterativas, las cuales deben ser planeadas y seguidas como si se tratara de un proyecto a menor escala.
- *Permite Gestionar Riesgos:* Un riesgo es una variable que puede incidir en el éxito de un proyecto, y puede causar efectos adversos tales como contratiempos, sobrecostos, hasta la cancelación del mismo. La gente hereda riesgos cuando asume uno o más roles de un proyecto; Infortunadamente, la gente generalmente no hereda la capacidad necesaria para administrarlos [Hall 00]. El MICACE propone técnicas que permiten asegurar una gestión adecuada de los riesgos del proyecto.
- *Guiado por Casos de Uso:* Se considera al sistema de comercio electrónico según la orientación que da [Jacobson 99], el cual distingue que un sistema software da la luz para dar servicios a sus usuarios. Por lo tanto para construir un sistema de comercio electrónico eficiente, se debe conocer lo que usuarios necesitan y desean, entendiéndose por usuarios a personas u otros sistemas fuera del sistema de comercio electrónico en consideración, que

interactúan con él. Los casos de uso constituyen la guía fundamental para el desarrollo de las actividades durante el proceso.

- *Centrado en la Arquitectura:* La arquitectura de un sistema de comercio electrónico involucra sus elementos más significativos, y está influenciada por consideraciones de desarrollo como sistemas legados y requisitos no funcionales. Es importante aclarar que los casos de uso dirigen la arquitectura del sistema.
- *Permite Rehúso del Diseño:* El MICACE acepta que existen soluciones elaboradas para problemas comunes de diseño de sistemas de la misma naturaleza (*Patrones*). Por lo tanto, permite el uso de estas soluciones dadas por la experiencia en el dominio del problema.
- *Basado en Componentes:* En sistemas de comercio electrónico de gran escala, el uso de componente juega un papel importante, ya que permite dividir el sistema en componentes con interfaces bien definidas, que posteriormente se ensamblarán para generar el sistema objetivo.
- *Permite rastreo sistemático del progreso:* Se efectúa un seguimiento del proyecto basado más en la evaluación absoluta del esfuerzo restante que en porcentaje de terminación de actividades.
- *Permite el control de versiones:* El MICACE es consciente de que el manejo adecuado de las versiones de los productos de trabajo generados en el proyecto es un punto clave que influye en la calidad del producto y la capacidad del proceso. Por este hecho, el MICACE propone técnicas estándar que permiten alcanzar este objetivo.

3.3.1 Fases de referencia

El MICACE hace uso del paradigma principal la técnica de desarrollo incremental por medio de iteraciones, la cual busca manejar la complejidad de proyecto, y proveer al Cliente de una capacidad operacional inicial que pueda ser validada y mejorada a través su realimentación. En la tabla 3.2 se puede observar la descripción general de las fases.

FASES	OBJETIVOS GENERALES	MODELOS
ESTUDIO PRELIMINAR	Definir el alcance del proyecto y sus Objetivos principales.	Modelo para el Establecimiento de Responsabilidades (versión inicial)
FORMULACIÓN DEL PROYECTO	Definir una arquitectura de referencia para la construcción del sistema de comercio electrónico, y eliminar y mitigar los riesgos críticos.	Modelo para el Establecimiento de Responsabilidades (versión actualizada)
CONSTRUCCIÓN DEL SISTEMA	Construir la capacidad operativa inicial del sistema comercio electrónico.	Modelo para Descripción del Sistema y Modelo de

		Implementación del Sistema.
VALIDACIÓN DEL SISTEMA	Validar el sistema de comercio electrónico por parte del cliente, corregir errores detectados, y gestionar características nuevas que se puedan añadir.	

Tabla 3.2 Descripción de las Fases de Referencia

3.3.2 Estructura de referencia para cada fase

La estructura de referencia para todas las fases es la misma y se puede observar en la figura 3.1. Los aspectos más relevantes presentados en esta estructura son los siguientes:

- *Revisión y compromiso:* Es una actividad que cubre todos los productos de trabajo elaborados en la fase previa, incluyendo el plan de trabajo para la siguiente fase y los recursos requeridos para llevarlo a cabo.
- *Planeación de la fase:* En esta actividad se elabora un plan de trabajo para la siguiente fase que incluye estimaciones de los recursos requeridos, además se planean las políticas y los métodos para llevar el control de versiones de los productos de trabajo de la siguiente fase.
- *Flujo de trabajo:* Representa una iteración mediante la cual se integran dos o más de las siguientes actividades principales: Modelado de la Organización, Captura de Requisitos, Análisis, Diseño, Implementación, y Pruebas (O,R,A,D,I,P).
- *Evaluación de la fase:* Evalúa los productos de trabajo elaborados, con base en los compromisos adquiridos, y los objetivos establecidos para la fase, junto al estado de gestión de riesgos, con el fin de establecer la viabilidad de la transición hacia la siguiente fase.

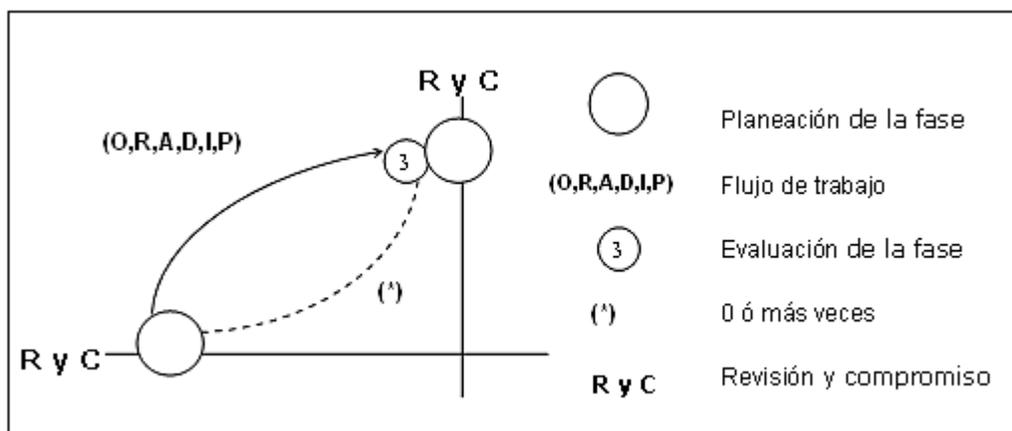


Figura 3.1. Estructura patrón para cada Fase de Referencia

3.4 MODELO DE REFERENCIA ORGANIZACIONAL

La clasificación del recurso humano que propone el MICACE permite cumplir eficientemente los objetivos de administración, desarrollo y control en un proyecto que pretende construir un sistema de comercio electrónico, este se presenta en las tablas 3.3, 3.4, 3.5. En estas se distinguen tres grupos en las cuales esta dividido el equipo del proyecto, estos grupos son Grupo de gestión, Grupo de Control, y Grupo de Desarrollo.

GRUPO DE GESTION		
CARGO	ÁREAS DE DESEMPEÑO	TAREA ESENCIAL
Director	1. Liderazgo del Proyecto 2. Integración del Producto 3. Calidad del Producto	1. Construir un ambiente de trabajo adecuado 2. Asegurar la coherencia del Producto 3. Garantizar la satisfacción del cliente
Asesor	1. Gestión de Reusabilidad 2. Calidad del Proceso	1. Enriquecer la base de conocimiento/experiencia 2. Mejorar sistemáticamente el proceso
Consultor	1. Producto 2. Proceso	1. Evaluar aspectos claves 2. Validar métodos y decisiones

Tabla 3.3 Estructura del Grupo de gestión

GRUPO DE CONTROL		
CARGO	AREA DE DESMPEÑO	TAREA ESENCIAL
Comité de Control de Cambios	1. Gestión de la configuración 2. Auditoría	1. Aprobar o denegar un cambio en alguna versión de la línea base del proyecto. 2. Aprobar o denegar la creación de nuevas componentes

		de configuración. 3. Revisar la línea base del proyecto y verificar la consistencia de los cambios según lo propuesto por los solicitantes y de acuerdo a los estándares y procedimientos aprobados
Administrador de Configuración	- Control de ejecución de cambios	1. Controlar los cambios que se realicen a la línea base del proyecto. 2. Administrar el repositorio de línea base del proyecto y generar reportes de la configuración.

Tabla 3.4 Estructura del grupo de control

GRUPO DE DESARROLLO	
CARGO/POSICIÓN	RESPONSABILIDADES BÁSICAS
Analista del sistema	1. Modelar la Organización 2. Elaborar el Modelo de casos de uso del sistema de comercio electrónico
Arquitecto del sistema	- Diseñar la arquitectura para el sistema de comercio electrónico
Ingeniero de casos de uso	1. Analizar un caso de uso 2. Diseñar un caso de uso
Diseñador de interfaces gráficas	- Prototipar una interfaz de usuario
Ingeniero de componentes	1. Analizar y diseñar clases 2. Analizar paquetes 3. Diseñar e implementar subsistemas 4. Implementar clases 5. Implementar pruebas 6. Realizar pruebas
Analista de seguridad	1. Verificar el ambiente de ejecución 2. Verificar la implantación del sistema de comercio electrónico
Ingeniero de pruebas	- Planear, diseñar, y evaluar las pruebas

Tabla 3.5 Estructura del grupo de desarrollo

4. DESCRIPCIÓN DEL PROCESO DE DESARROLLO

4.1 ESTUDIO PRELIMINAR

El presente apartado contiene la descripción del Estudio Preliminar como primera fase que esta aproximación de desarrollo define para iniciar un proyecto con el objetivo de crear una sistema de comercio electrónico sobre plataforma web.

4.1.1 Descripción general de la fase

4.1.1.1 Objetivos generales

Definición del alcance para el proyecto y sus objetivos principales.

4.1.1.2 Objetivos específicos

- Identificar el entorno y los procesos que soportará el sistema de comercio electrónico en la empresa u organización donde será implantada.
- Definir las responsabilidades iniciales que debe cumplir el sistema de comercio electrónico a desarrollar.
- Definir una arquitectura inicial que soporte la funcionalidad básica del sistema de comercio electrónico.
- Realizar una planeación inicial para el proyecto.
- Definir el caso de negocio.

4.1.1.3 Justificación

El Estudio Preliminar se considera de suma importancia para el inicio de un proyecto que pretenda crear un sistema de comercio electrónico sobre plataforma web. En esta primera fase se presta interés al estudio de la organización como un sistema compuesto por procesos internos los cuales deben ser descritos y analizados. Esta orientación es importante debido que las aplicaciones de comercio electrónico pueden llegar a automatizar procesos existentes y generar nuevas

dimensiones de mercado sobre los negocios de una organización, ya que por ejemplo pueden captar nuevos clientes con el consecuente aumento de labores logísticas y contables.

4.1.1.4 Etapas de referencia

1. Planeación del Estudio Preliminar
2. Modelado de la Organización, definición de los requisitos iniciales del sistema y selección inicial de la arquitectura.
3. Gestión Inicial de Riesgos para el Proyecto.
4. Planeación Inicial de Proyecto.
5. Determinación Preliminar de la Viabilidad del Proyecto.

4.1.1.5 Productos de trabajo del Estudio Preliminar

1. Declaración inicial del negocio
2. Plan de trabajo para el Estudio Preliminar
3. Lista de comprobaciones para el Estudio Preliminar
4. Plan de gestión de configuración
5. Modelo de negocio(primer versión)
6. Modelo de casos de uso(primer versión)
7. Lista de requisitos no funcionales
8. Documento de priorización de casos de uso
9. Modelo de despliegue
10. Lista de riesgos para el proyecto
11. Plan de trabajo para el proyecto
12. Lista de comprobaciones para el Estudio Preliminar
13. Caso del negocio

4.1.2 Etapas de referencia

4.1.2.1 Etapa I - Planeación del Estudio Preliminar

Propósito

Organizar y definir actividades, tiempos, objetivos y recursos para el desarrollo para el Estudio Preliminar, además de definir los mecanismos para asegurar el seguimiento de la fase.

Actividades

1. Declarar el negocio según la perspectiva del cliente

2. Elaborar el plan de trabajo
3. Establecer los criterios de evaluación
4. Definir de los mecanismos de seguimiento y control

Actividad 1. Declarar el negocio según la perspectiva del cliente

A. Descripción

El objetivo primordial de esta etapa es lograr una total comprensión de la idea que el cliente tiene sobre el sistema de comercio electrónico a desarrollar y obtener la mayor información posible sobre la organización. Esta actividad es llevada a cabo por un equipo compuesto por el director del proyecto, el Analista del sistema, el Arquitecto del sistema, y un asesor de la organización para una descripción adecuada de los productos/sevicios a ofrecer e información complementaria de procesos de interés que influyen en el sistema de comercio electrónico.

Como un resultado de esta actividad se tiene una lista inicial de requisitos la cual se ve desde dos puntos de vista, uno de parte del cliente y otro de parte del equipo del proyecto. Si bien los requisitos del cliente dan una idea aproximada del funcionamiento del sistema del comercio electrónico, es deber del equipo del proyecto complementar esta visión de sistema con características de *funcionalidad* que sean convenientes.

B. Técnicas

Entrevistas

Es una herramienta para reunir información de personas o grupos. Por lo general, los involucrados en las entrevistas son gerentes o empleados que proporcionan datos iniciales sobre el sistema de comercio electrónico que se propone construir y que en alguna forma los afecta. Se puede encontrar más información sobre las entrevistas en [Senn 98].

Para obtener una adecuada visión del negocio se recomienda tener en cuenta las siguientes directrices orientadas a capturar la información para iniciar la planeación del proyecto:

- Reunir toda la información de la estructura administrativa necesaria para tener una idea más clara del enfoque de la organización, por ejemplo, visión, misión, organización del personal, gestión de proveedores, entre otros.
- Si es necesario se debe profundizar sobre aspectos específicos de los productos a ofrecer a los clientes. Por ejemplo, si se desean vender productos que desconozca el equipo del proyecto se debe contactar con un asesor que sea de confianza del cliente para tener una idea clara de lo que se pretende vender.

- Es importante que el resultado de las reuniones esté documentado y sea aprobado por las dos partes.
- En muchos casos, el delegado de la organización para establecer reuniones con el grupo de desarrollo conoce a fondo los procesos de la organización pero no el ambiente de aplicaciones web. La comunicación, entonces, debe estar orientada en primer lugar al acuerdo del lenguaje que usarán entre ambas partes.
- Toda la información recolectada a partir de las reuniones con el cliente, más la información adicional necesaria reunida por el grupo del proyecto debe ser documentada (escrita) y además debe estar al alcance de todo el grupo de desarrollo y de gestión del proyecto ya que será la primera referencia al momento de construir el plan del proyecto.
- Si la arquitectura propuesta por el cliente requiere la interacción con sistemas ajenos a la organización se debe recopilar la información necesaria sobre el funcionamiento de estos sistemas y el estudio preliminar de la interacción con ellos.
- Es de resaltar que algunos requisitos pueden ser negociados más fácilmente si se cuenta con prototipos de interfaces gráficas para posibles aspectos del sistema de comercio electrónico, por lo tanto el uso de patrones de interfaz puede ayudar a conseguir acuerdos sobre funcionalidades comunes de este tipo de aplicaciones.

C. Productos de trabajo

Declaración inicial del negocio, documento que contiene como base los siguientes puntos:

1. Propósito del sistema de comercio electrónico que se desarrollará
2. Tipo de clientes a los cuales van dirigidos los productos y servicios ofrecidos por la organización
3. Características esenciales y opcionales del sistema de comercio electrónico. Descripción de las personas y sistemas que van a interactuar y todas sus relaciones
4. Beneficios que se buscan para la organización con el sistema de comercio electrónico a desarrollar.
5. Arquitectura inicial propuesta por el cliente
6. Lista inicial de requisitos

Actividad 2. Elaborar el plan de trabajo

A. Descripción

Esta actividad define el orden de las actividades requeridas para realizar el Estudio Preliminar del proyecto y establecer la organización del personal involucrado, determinar los costos y el esfuerzo requerido para el Estudio Preliminar. Esta actividad es responsabilidad del director del proyecto,

aunque se debe tener en cuenta que él puede apoyarse en el Analista del sistema y/o el Arquitecto del sistema para efectuar cálculos de tamaño o esfuerzo necesario para el proyecto.

B. Técnicas

Estimaciones de tiempo

Este tipo de estimaciones son aproximaciones de las horas, días o meses de esfuerzo necesario para el desarrollo de un sistema determinado. Se pueden mencionar los siguientes métodos comunes para estimar el tiempo de desarrollo de un proyecto:

- *Método Histórico:* Se basa en registros cuidadosos que se han mantenido con respecto a proyectos de desarrollo anteriores.
- *Método Intuitivo:* Se basa en la intuición para hacer cálculos de tiempo. Se debe tener en cuenta que es difícil estimar con un alto grado de precisión, sin embargo la popularidad de este método se debe a que es rápido para obtener una estimación.
- *Método Estándar:* en este se identifica y cuantifican (con pesos individuales) los factores que afectan más drásticamente al proyecto, tales como las características del personal, los detalles del sistema y la complejidad del proyecto. A este respecto se pueden mencionar por ejemplo la distinción de los tipos de proyecto que se hace para los modelos de COCOMO según el tamaño: *modo orgánico* para proyectos relativamente pequeños y con grupos con gran experiencia en sistema; *modo semiacoplado* para proyectos de software intermedios y con equipos de varios niveles de experiencia, y *modo empotrado* para proyectos de software desarrollados en conjunto con hardware y restricciones operativas muy restringidas.

Más información al respecto en [Senn 98]. Se pueden encontrar información sobre la orientación el modelo de COCOMO y la versión actualizada del mismo en [CSE 98].

Estimaciones de costo y esfuerzo

La estimación del costo y del esfuerzo para un proyecto de software nunca será una ciencia exacta. Son demasiadas las variables humanas, técnicas, de entorno, políticas que pueden afectar al costo final del software y al esfuerzo aplicado para desarrollarlo. Algunas de las técnicas y modelos utilizados para el cálculo de costo y el esfuerzo son las siguientes:

- *Método Histórico:* en este método se tiene en cuenta los cálculos obtenidos para proyectos similares anteriores. Los datos históricos se usan para identificar los factores de costo y determinar la importancia relativa de los diversos factores dentro de la organización.
- *Técnicas de Descomposición:* estas técnicas proponen que la estimación de costos puede llevarse en forma jerárquica hacia abajo o en forma jerárquica hacia arriba (Bottom Up). La primera se enfoca primero en los costos a nivel del sistema, así como los costos del manejo de

configuración, de control de calidad, la integración del sistema, de entrenamiento y de documentación. La segunda estima el costo de desarrollo de cada módulo o subsistema, tales costos se agrupan para obtener el costo total.

- *Modelos Empíricos*: estos modelos utilizan fórmulas derivadas empíricamente para predecir el esfuerzo como una función de Líneas de Código (LDC) y Puntos de Función (PF), es el caso los modelos de COCOMO. Sin embargo los datos empíricos que soportan la mayoría de los modelos de estimación se obtienen de una muestra limitada de proyectos y por tal razón este modelo de estimación no es adecuado para todos los tipos de software y entornos de desarrollo.

Más información al respecto en [ITCG 03].

Requisitos de tiempo calendario

La simple identificación de los requisitos de tiempo del proyecto no da como resultado el número de días, semanas o meses para la preparación del calendario del proyecto. Esto solo corresponde a un factor que influye sobre éste. Algunas actividades tales como reuniones con la gerencia, revisiones del proyecto, educación y capacitación, etc. consumen tiempo adicional del proyecto por lo tanto se necesitan técnicas para administrar dichos tiempos. Estos son algunos de los más usados:

- *Diagramas de Barras*: esta es una de las técnicas de planificación más simple en la cual se usan diagramas de barras que muestran cada actividad en un proyecto y la cantidad de tiempo que se tomará esa actividad. Este método fue desarrollado por Henry L. Gantt, con lo que comúnmente a sus diagramas se les llama diagramas de Gantt.
- *Diagramas de Eventos Críticos*: estos diagramas muestran los eventos significativos en la conclusión de un proyecto y la secuencia en la que deben llevarse a cabo. Difieren de los diagramas de barras en que representan puntos de conclusión, no tareas individuales de realizar.
- *Diagramas de PERT*: debido a que los proyectos constan de eventos y actividades, el diagrama de PERT usa nodos y rutas para representar la interrelación de las actividades del proyecto. Los nodos indican eventos y las rutas las actividades necesarias para moverse de un nodo a otro.

Más información al respecto en [Senn 98].

C. Productos de trabajo

Plan de trabajo para el Estudio Preliminar, documento que contiene:

1. Relación de actividades a realizar, duración estimada y esfuerzo

2. Descripción del equipo humano inicial y asignación de responsabilidades
3. Descripción de los recursos requeridos y las respectivas ventanas de disponibilidad
4. Estimación de los costos

Actividad 3. Establecer los criterios de evaluación

A. Descripción

En esta actividad se definen los aspectos fundamentales que deben obtener evaluación satisfactoria para poder dar por terminado el Estudio Preliminar del Proyecto, es actividad es responsabilidad del Director del proyecto. Los aspectos a evaluar pueden ser complementados o adaptados a las características particulares de cada proyecto.

B. Técnicas

La lista de comprobaciones del producto de trabajo sirve como base para implementar una lista propia para el equipo del proyecto de acuerdo a la experiencia en anteriores proyectos y las características particulares de cada uno de ellos.

Definición del alcance del proyecto

- ¿Se obtuvo una descripción suficientemente clara del sistema de comercio electrónico y una adecuada declaración de su propósito?
- ¿Se realizó una clara identificación de actores?
- ¿Se tiene un modelo inicial del negocio que representa realmente los procesos de negocio necesarios?
- ¿Se tiene un modelo de casos de uso del sistema de comercio electrónico adecuado?

Resolución de ambigüedades en los requisitos

- ¿Se ha logrado un reconocimiento satisfactorio de requisitos (funcionales y no funcionales) y en el nivel de detalle adecuado?
- ¿Se han llevado correctamente esos requisitos a casos de uso?
- ¿Existe coherencia entre los modelos elaborados y la Declaración Inicial del Negocio?
- ¿Se han priorizado convenientemente los casos de uso?

Viabilidad de la arquitectura inicial

- ¿Satisface la arquitectura propuesta las necesidades del Cliente y de los Usuarios?
- ¿Es factible realizar una implementación del sistema de comercio electrónico según la arquitectura propuesta?

- ¿Se han considerado alternativas a ésta?
- ¿La arquitectura propuesta utiliza apropiadamente la tecnología existente?
- ¿Se han evaluado sus criterios de eficiencia, tolerancia a fallas, adaptabilidad y robustez?
- ¿La arquitectura planteada posibilita el crecimiento o evolución del sistema de comercio electrónico?
- ¿Se han realizado todas las evaluaciones pertinentes indicadas en esta guía?

Identificación de riesgos críticos

- ¿Se han identificado todos los riesgos críticos?
- ¿Se evaluaron los elementos del entorno de desarrollo y del entorno de ejecución, y se han propuesto medidas adecuadas al respecto?
- ¿Se ha elaborado un presupuesto consistente y acorde a las necesidades?
- ¿Se ha elaborado un cronograma o un plan de trabajo a seguir realista?

Conveniencia del caso inicial del negocio

- ¿Es el Caso Inicial del Negocio lo suficientemente satisfactorio para justificar la continuación del proyecto?

C. Productos de trabajo

Lista de comprobaciones para el Estudio Preliminar, contiene una lista de comprobaciones para el estudio preliminar del proyecto similar a la presentada en las técnicas de la Actividad 3 con las modificaciones específicas que se puedan añadir según el criterio de los gestores del proyecto y sus características específicas.

Actividad 4. Definir los mecanismos de seguimiento y control

A. Descripción

Esta actividad es necesaria para asegurar un correcto seguimiento de los productos de trabajo generados en el transcurso del proyecto. Esto permite el establecimiento una línea base para la construcción de productos de trabajo en esta y posteriores etapas evolutivas del proyecto, y la posibilidad de controlar las versiones de todos los productos. Generalmente el proceso que implica esta actividad consiste en que todo producto generado por los integrantes del proyecto tales como documentos de planeación, documentos de especificación de requisitos, modelos, código fuente, ejecutables, interfaces de usuario, etc. se guardan en un repositorio que generalmente se denomina *librería de línea base de software* que sirve como almacén de todos los productos de trabajo. De esta forma, cuando un integrante del equipo del proyecto genera un producto de trabajo

este debe ser revisado con el fin de establecer su validez para que posteriormente sea integrado en el repositorio de productos de trabajo del proyecto como una nueva versión del mismo. Se debe tener en cuenta que para la creación de una nueva versión de un producto de trabajo se debe hacer a partir de las versiones que se tienen almacenadas en el repositorio.

Es de recalcar que la configuración de un repositorio de línea base consistente para un proyecto de comercio electrónico es de vital importancia puesto que puede servir de referencia para administrar posibles evoluciones del sistema (Escalabilidad), impuestas por condiciones del mercado y la demanda.

B. Técnicas

Los siguientes son los puntos a tener en cuenta cuando un equipo de desarrollo pretende llevar un proceso de seguimiento y control de cambios para un proyecto:

- Se deben definir responsabilidades para llevar a cabo las labores del seguimiento y control. Generalmente estas responsabilidades están distribuidas en dos grupos: El Comité de Control de Cambios quien es el encargado de procesar las peticiones de cambios por parte de alguno de los miembros del grupo del proyecto en algún producto de trabajo que se encuentra almacenado en el repositorio; y un Administrador de Configuración quien es el encargado de controlar los cambios al repositorio de líneas base del proyecto y también de controlar la introducción de nuevos productos de trabajo a la librería del proyecto.
- El comité de control de cambios es el encargado de auditar la librería de línea base del proyecto periódicamente para observar que los cambios se llevaron a cabo de acuerdo a los puntos definidos en las peticiones.
- Se deben seleccionar los productos de trabajo principales que van a ser incluidos para hacer seguimiento al proyecto además de un esquema de identificación para estos productos de trabajo guardados en la librería. Generalmente el esquema de identificación se establece de la forma 1.X , siendo X la versión del producto de trabajo.
- Toda petición de cambio hecha por parte de los implicados en el desarrollo de los productos de trabajo tiene que ser evaluada por el Comité de Control de Cambios y evaluar su implementación teniendo en cuenta algunos criterios de aprobación tales como tamaño, consecuencias, y necesidad de los cambios.
- La implementación de los cambios se hace por parte de los desarrolladores de los productos de trabajo con la coordinación del Administrador de Configuración.
- El Administrador de Configuración tiene la tarea de generar reportes periódicos del estado de la línea base del proyecto para los grupos que así lo soliciten.
- Se debe llevar una relación de los cambios realizados, autor de los cambios, fecha de realización.

Se puede encontrar una ampliación de los conceptos sobre el manejo de la configuración de un proyecto de software y algunas herramientas de soporte para estas actividades en [ASPL 03].

C. Productos de trabajo

Plan de gestión de configuración, que contiene:

1. Identificación de unidades de configuración
2. Asignación de responsabilidades
3. Políticas para la gestión y control de versiones

Es de indicar que el plan de gestión de configuración es otro documento más que se desarrolla en el transcurso del proyecto y por lo tanto también esta sujeto a cambios, o mejor dicho se puede colocar bajo Gestión de Configuración. La siguiente es una plantilla propuesta para documentar un Plan de Gestión de Configuración y llevar la relación de cambios del mismo:

[NOMBRE DEL PROYECTO]: PLAN DE GESTIÓN DE CONFIGURACIÓN	
[Versión del Documento]	
[Autor del Documento]	
[Fecha]	
I. Entidades Bajo Gestión de Configuración	
Productos de trabajo del proyecto para hacerles control de versiones	
II. Asignación de Responsabilidades	
Integrantes del equipo del proyecto encargados de las funciones de configuración	
III. Políticas para control de cambios y gestión de versiones.	
Qué criterios y controles se van a tener en cuenta para generar una nueva versión	
IV. Relación de cambios hechos al documento	
VERSIÓN	Versión del documento.
Responsable (s)	Nombres de los responsables del cambio
Fecha	Fecha de cambio
Descripción del cambio	Descripción de los cambios hechos al documento

Figura 4.1 Plantilla para plan de gestión de configuración

La forma de controlar la versión del documento de gestión de configuración también puede aplicarse a los diferentes productos de trabajo de interés para el proyecto.

4.1.2.2 Etapa II - Modelado de la organización, definición de requisitos iniciales del sistema y selección inicial de la arquitectura.

Propósito

Modelar los procesos de la organización y definir los requisitos iniciales para el sistema de comercio electrónico.

Actividades

1. Identificar los procesos de negocio.
2. Definir las responsabilidades iniciales del sistema.
3. Definir los requisitos no funcionales.
4. Priorizar casos de uso.
5. Describir la Arquitectura Inicial para el sistema.

Actividad 1. Identificar los procesos de negocio

A. Descripción

Esta actividad pretende hacer una revisión de los procesos de negocio que posee la organización con el fin de estudiar el impacto que el sistema de comercio electrónico pueda tener sobre esos procesos. El Analista del sistema es el encargado de recolectar la información de utilidad de parte del cliente con el objetivo de identificar aquellos procesos que pueden ser afectados por el sistema de comercio electrónico. Se debe tener en cuenta que un entendimiento completo de los procesos de la organización es un elemento clave para definir el ambiente donde se va aplicar el sistema de comercio electrónico e identificar los procesos con los cuales debe interactuar.

B. Técnicas

Técnicas para encontrar hechos

Debido a que el propósito de analizar los procesos de negocio de una organización es una de las primeras actividades que el Analista del sistema realiza con el fin de capturar requisitos para el sistema de comercio electrónico a construir, existen algunas técnicas específicas que son de utilidad para reunir datos y que pueden ser aplicadas con el fin de obtener información de la organización del cliente y sus procesos internos.

- *Entrevistas:* Esta es una técnica que se utiliza para reunir información de personas o grupos. Generalmente los entrevistados pueden ser usuarios de sistemas existentes o personas

involucradas con los procesos de negocio que se ejecutan en la organización y que tienen cargos que pueden describir comportamientos importantes para el sistema de comercio electrónico a construir. Es importante recordar que los entrevistados conversan durante la entrevista, es decir no se interroga a los primeros.

- *Cuestionarios:* Esto permite que el Analista del sistema reúna información relacionada con diversos aspectos de un sistema de un grupo grande de personas. Para este propósito generalmente se emplean formatos estandarizados. Sin embargo este método no permite observar las expresiones o reacciones de los encuestados.
- *Revisión de registros:* La revisión de registros y reportes proporciona al Analista del sistema información de interés respecto a la organización y a sus operaciones. Al revisar los registros el Analista del sistema puede examinar la información consignada en ellos relacionada con sistemas, procesos, y usuarios. Aunque esta información no puede indicar la forma en que se desarrollan las actividades dentro de la organización, si pueden servir de gran ayuda para familiarizarse con las operaciones que necesitan apoyo y las relaciones formales dentro de la organización.

Se puede encontrar más información acerca de estas técnicas en [Senn 98].

Modelado del negocio

El modelado del negocio es una estrategia para modelar los procesos de negocio que posee una determinada organización o empresa. Los modelos de negocio proveen un camino para expresar los procesos de negocio en términos de actividades y conductas colaborativas de negocio.

Esta orientación del análisis del negocio es fundamental cuando se construyen aplicaciones para negocios electrónicos (*e-business*) y por ende para aplicaciones de comercio electrónico como sub dominio de las anteriores. Este tipo de aplicaciones se caracterizan por emular con exactitud los procesos del mundo real y apoyarse en nuevas tecnologías y nuevos canales. Los beneficios que acarrea el modelado del negocio son producir mejores requisitos del sistema de comercio electrónico que conducirán la creación de aplicaciones que realmente se acomodan a la organización y que serán usados por los usuarios finales. Si bien en Internet se pueden encontrar dos tipos bien definidos de empresas, las que nacen enteramente en Internet y que no tienen unas instalaciones físicas para entregar sus productos o servicios directamente a los clientes (“Click”), y quienes ya las tienen y desean migrar algunos de sus procesos de negocio hacia las nuevos canales gracias a las nuevas tecnologías (“Brick & Click”); Indistintamente sigue siendo de vital importancia el modelado del negocio para todos los tipos de aproximaciones empresariales hacia el comercio electrónico sobre Internet, ya que esto permite visualizar el impacto que la solución tiene sobre los procesos internos además de la visualización del panorama comercial que se desprende de la utilización de este tipo de tecnologías, como por ejemplo el aumento del tamaño

del mercado puede producir un incremento del volumen de ventas y por lo tanto un incremento de las operaciones y alianzas logísticas de la organización así como nuevos retos y compromisos financieros para la empresa, etc.

Modelo de casos de uso de negocio. Este observa el negocio desde una perspectiva externa. El modelo de casos de uso de negocio incluye una descripción de los actores de negocio y de los casos de uso de negocio y las interacciones entre los actores de negocio y los casos de uso de negocio (Diagramas de casos de uso de negocio).

Diagramas de Casos de Uso del Negocio. Se considera como el primer paso en el modelado del negocio, este define la interacción entre las entidades fuera del negocio y los procesos de negocio. Un diagrama de Casos de Uso de negocio representa visualmente la interacción entre los servicios primarios (casos de uso de negocio) que el negocio provee y para quien son proveídos dichos servicios (actores de negocio).

Estereotipo UML	Nombre	Definición UML
	Actor de negocio	Alguien o algo fuera del negocio que interactúa con él.
	Caso de uso de negocio	Una secuencia de acciones que ejecuta el negocio y que produce un resultado observable para un actor de negocio particular.

Tabla 4.1 Notación UML para actores y casos de uso de negocio

Modelo de objetos de negocio. Este modelo detalla cómo los procesos de negocio son implementados internamente. El modelo de objetos de negocio incluye una descripción de los empleados del negocio (trabajadores de negocio), las cosas (entidades de negocio) que el negocio manipula y como los trabajadores de negocio manipulan las entidades de negocio para ejecutar los procesos de negocio (Diagramas de clases).

Estereotipo UML	Nombre	Definición UML
	Trabajador del negocio	Rol o conjunto de roles dentro del negocio. Un trabajador de negocio interactúa con otro y manipula entidades de negocio.

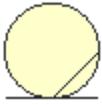
	Entidad del negocio	Son cosas manejadas o usadas por los trabajadores de negocio.
---	---------------------	---

Tabla 4.2 Notación de modelado UML para trabajadores y entidades de negocio

C. Productos de trabajo

Modelo de negocio (primera versión), compuesto por:

1. Modelo de casos de uso de negocio
2. Modelo de objetos de negocio
3. Glosario de términos

Se puede encontrar más al respecto de esta técnica en [RSC 01].

Actividad 2. Definir las responsabilidades iniciales del sistema

A. Descripción

Con esta actividad se propone definir la funcionalidad inicial para el sistema de comercio electrónico con el fin de capturar los elementos de comportamiento más importantes. Para esta finalidad se utiliza el modelado del sistema de comercio electrónico a través de casos de uso con el objetivo de recoger los requisitos funcionales. El responsable de la identificación de casos de uso es el Analista del sistema.

B. Técnicas

Casos de uso

El modelo de casos de uso es un modelo del sistema que contiene actores, casos de uso y sus relaciones (Diagrama de casos de uso). En general este modelo describe lo que hace el sistema para cada tipo de usuario. Cada usuario del sistema se representa mediante uno o más actores. También se representan de la misma forma cada sistema externo con que interactúa con sistema. Por lo tanto se puede decir que los actores son terceros fuera del sistema que colaboran con él. Se dice que cuando se han identificado todos los actores del sistema se ha identificado su entorno externo.

Un caso de uso delimita la forma en que un actor usa el sistema. Representan trozos de funcionalidad que el sistema ofrece con el fin de aportar un resultado de valor para los actores. Los Casos de Uso pueden verse como acciones secuenciales y sus alternativas cuando un actor interactúa con el sistema.

Debido a que primero se ha desarrollado un modelo de negocio para el sistema de comercio electrónico, este se utiliza para encontrar los casos de uso y actores. Estos son algunos de los pasos a seguir cuando se busca casos de uso y actores a partir del modelo de negocio:

- Identificar cada trabajador o actor de negocio que vaya a ser uso del sistema de comercio electrónico requiere funcionalidades de soporte por parte del mismo.
- Se debe identificar para cada trabajador todas las realizaciones de casos de uso de negocio en las que participa.
- Para encontrar los casos de uso para los actores del sistema de comercio electrónico se debe encontrar todos los roles de un trabajador o de un actor de negocio. Cada trabajador y cada actor de negocio se corresponden con al menos un actor del sistema de comercio electrónico. Se debe tener en cuenta que una vez realizado el modelo de negocio previamente, se han identificado los procesos a los cuales va a brindar soporte el sistema de comercio electrónico, por lo tanto se ha identificado tanto los trabajadores de negocio y actores de negocio.
- Se necesita que por cada papel de un trabajador o actor de negocio exista al menos un caso de uso para el actor del sistema de comercio electrónico correspondiente.

Escenarios. La descripción del escenario permite tener una visión textual del caso de uso con el fin de mejorar su comprensión. Para la descripción de los escenarios a nivel del Estudio Preliminar se propone la siguiente plantilla:

Nombre del Caso de Uso	Es un nombre descriptivo dado al caso de uso acompañado de un número de orden.
Iniciador	Se identifica el actor iniciador del Caso de Uso.
Propósito	Se describe el propósito para el cual fue creado el caso de uso.
Resumen	Una breve descripción de la funcionalidad del caso de uso.

Tabla 4.3 Plantilla para la descripción de escenarios a nivel del Estudio Preliminar

Más información al respecto de esta técnica en [Jacobson 99].

C. Productos de trabajo

Modelo de casos de uso (primera versión), el cual contiene:

1. Diagrama de casos de uso del sistema de comercio electrónico.
2. Descripción de escenarios.
3. Descripción de actores.

Actividad 3. Definir los requisitos no funcionales

A. Descripción

Con esta actividad se permite tener una visión inicial de los requisitos no funcionales para el sistema de comercio electrónico. Los requisitos no funcionales son requisitos que no pueden asociarse a ningún caso de uso en concreto. Este tipo de requisitos son frecuentes en aplicaciones sobre plataforma web y en general para sistemas de comercio electrónico. La captura de los requisitos no funcionales la realizan el Analista del sistema y el Arquitecto del sistema, este último identifica los requisitos que pueden tener una influencia importante sobre la arquitectura.

B. Técnicas

Generalmente la forma en que se pueden capturar los requisitos adicionales es parecida a la que se utiliza en la captura de requisitos tradicional, en otras palabras, a través de un listado de requisitos. Este tipo de requisitos se emplea en conjunto con el Modelo de Casos de Uso para hacer análisis y diseño al sistema de comercio electrónico. Entre algunos ejemplos de requisitos no funcionales frecuentemente encontrados están:

- Seguridad: quizá la más común en cuanto a requisitos adicionales para comercio electrónico debido a su naturaleza. Es indispensable para evitar que terceras partes maliciosas puedan acceder a la información almacenada tales como bienes comerciales intangibles (software, documentos, etc.) o números de tarjetas de crédito e información personal de clientes almacenada, así como para proteger la información que viaja a través de Internet tales como números de tarjetas de crédito, contraseñas de acceso, y catálogos de productos de proveedores a intermediarios. Se puede consultar a [Hauswirth 00] para ampliar conceptos al respecto.
- Disponibilidad: es también un requisito común en cuanto a sitios comerciales en Internet ya que la no-disponibilidad de un sistema de este tipo por un intervalo considerable de tiempo puede causar pérdidas económicas importantes o producir desconfianza en los clientes.
- Restricciones de diseño: este tipo de restricciones de alguna forma limita el diseño del sistema de comercio electrónico, es el caso del uso de sistemas existentes en la organización o empresa donde se vaya a implantar, tal es el caso de sistemas de facturación o sistemas contables.
- Restricciones de implementación: muchas veces limita la codificación y construcción del sistema de comercio electrónico. Es el caso de sistemas que ofrecen Hosting, los cuales ya

ofrecen un soporte para ciertos lenguajes de programación y manejadores de bases de datos predeterminados.

- **Requisitos de interfaz:** se refiere a especificaciones de interfaz con elementos externos con el cual debe interactuar el sistema de comercio electrónico, o que establece restricciones en formatos, tiempos, u otros factores importantes en la interacción, como por ejemplo con pasarelas de pago. En sistemas de comercio electrónico de última generación los requisitos de interfaz son un punto clave, sobre todo en ambientes donde se intercambia información entre proveedores, intermediarios, y distribuidores finales de productos y servicios. Para este tipo de ambientes generalmente se definen formatos estándar para el intercambio de información comercial tales como catálogos de productos.

C. Productos de trabajo

Lista de requisitos no funcionales

Para la descripción de estos requisitos se plantea el siguiente formato:

Número de orden. Nombre del requisito no funcional	
Tipo	Identifica la clase de requisito (de diseño, de interfaz, de implementación, otros, etc),según lo descrito en las técnicas de la actividad 3 de la Etapa II de esta fase.
Descripción	Breve descripción del requisito

Tabla 4.4 Plantilla para la descripción de requisitos

Actividad 4. Priorizar casos de uso

A. Descripción

Con esta actividad se propone distinguir los casos de uso más importantes para el desarrollo posterior del sistema de comercio electrónico, además de la obtención de un refinamiento de estos mediante la descripción detallada de sus escenarios, si fuese necesario. Esta actividad es responsabilidad del Arquitecto del sistema, parcialmente intervendrá el Diseñador de interfaces gráficas cuando se quiera comprender en detalle una interacción entre un usuario y el sistema de comercio electrónico.

B. Técnicas

- Determinar qué casos de uso son necesarios para prestar la funcionalidad básica del sistema de comercio electrónico
- Determinar qué casos de uso constituyen la esencia del sistema de comercio electrónico, o que son fundamentales para definir una arquitectura previa. Estos deben ser implementados en los primeros ciclos de desarrollo.
- Identificar qué casos de uso conllevan un nivel de riesgo mayor. Estos deben ser tenidos en cuenta en las primeras iteraciones con el fin de mitigar posibles riesgos.
- Una forma de obtener una descripción detallada de un escenario de un caso de uso puede hacerse a través del uso de patrones de interfaces, esto permite visualizar de una forma clara como un actor lleva a cabo el caso de uso que necesita de interfaz gráfica, y por lo tanto descubrir los flujos que lo componen.

C. Productos de trabajo

Documento de priorización de casos de uso, que contiene:

1. Lista priorizada de casos de uso.
2. Refinamiento de los escenarios para los casos de uso críticos (opcional).

Actividad 5. Describir la arquitectura inicial

A. Descripción

Esta actividad pretende describir los recursos técnicos mínimos que debe tener el cliente para soportar la funcionalidad básica del sistema de comercio electrónico. Es de resaltar que se pueden dar muchos casos dependiendo del tipo de cliente que se pueda presentar. Estos casos pueden variar desde clientes que no tienen ninguna infraestructura tanto técnica como logística para afrontar el soporte del sistema de comercio electrónico, en cuyo caso es muy importante desarrollar esta actividad, hasta los que poseen infraestructuras para soporte de servicios web funcionando (sea propias o contratadas) y quieren implementar este tipo de aplicaciones con el fin de comercializar productos o servicios. Esta actividad es responsabilidad del Arquitecto del sistema.

B. Técnicas

A continuación se listan una serie de aspectos a tener en cuenta para la obtención de los componentes principales de la arquitectura básica de soporte:

- ¿Qué nodos se necesitan y cual debe ser su capacidad de procesamiento y almacenamiento?

- ¿Qué tipo de protocolos de comunicación y protocolos adicionales deben ejecutarse para la conexión de los nodos?
- ¿Qué características deben tener las conexiones entre los nodos tales como disponibilidad y ancho de banda?
- ¿Existen nodos que requieran redundancia o sistemas de respaldo (Backup)?

Patrones de arquitectura

Existen algunos patrones que pueden servir de guía al momento de definir una arquitectura de soporte al sistema de comercio electrónico, ellos pueden ayudar a comprender el hardware que contienen algunos sistemas y como se distribuye su lógica. Algunos de los ejemplos de estos patrones son: Client/Server, Three-Tier, Peer-to-Peer, entre otros.

C. Productos de trabajo

Modelo de despliegue, que identifica:

1. Nodos y sus características.
2. Protocolos (Comunicación, Seguridad, etc.)

4.1.2.3 Etapa III - Gestión inicial de riesgos para el proyecto

Propósito

El propósito de esta actividad es identificar preliminarmente los riesgos genéricos y específicos del proyecto con el fin de analizar su impacto y definir estrategias para afrontarlos. Esta etapa es de gran importancia ya que puede influir en la viabilidad del proyecto y en la definición temporal de las actividades de planeación.

Actividades

1. Identificar riesgos

Actividad 1. Identificar riesgos

A. Descripción

Con esta actividad se definen los factores que incidirán en el éxito del proyecto e identificar las posibles salidas para los mismos. El director del proyecto es el encargado de definir los riesgos más importantes que pueden afectar al proyecto, el debe definir una categorización de los riesgos de acuerdo al agente que los produce. Para esto el director del proyecto puede apoyarse en las

evaluaciones que puedan dar el Analista del sistema y el Arquitecto del sistema con el fin de identificar plenamente los riesgos en todos los sectores.

Los diferentes tipos de riesgos se deben analizar y priorizar teniendo en cuenta el posible impacto sobre el proyecto y definir los posibles planes de contingencia para los riesgos verdaderamente importantes.

B. Técnicas

Para identificar y gestionar un grupo inicial de riesgos se plantean las siguientes recomendaciones:

- Categorizar los riesgos para identificar claramente el agente que los produce, como por ejemplo definir riesgos:
 - *Relacionados con el tamaño del sistema:* tamaño estimado para el proyecto (LDC, PF, otras), confianza en la estimación, tamaño relativo con relación a otros proyectos, cantidad de cambios a los requisitos proyectados antes y después de la entrega, etc.
 - *Relacionados con el impacto en la organización:* como impacta el sistema de comercio electrónico a las ventas, fecha de entrega, número de sistemas con los que deberá interactuar, etc.
 - *Relacionados con el cliente:* existen experiencias anteriores con el cliente, el cliente está dispuesto a destinar tiempo a la especificación de requisitos, etc.
 - *Relacionados con el entorno de desarrollo:* existen herramientas para gestión de proyectos, hay herramientas para soportar el análisis y diseño del sistema de comercio electrónico apropiadas, existen expertos a los cuales se le pedirá soporte sobre las herramientas, etc.
 - *Relacionados con la tecnología:* existen nuevas tecnologías en el proyecto que la organización no haya manejado, se deben utilizar métodos de desarrollo no formales tales como la Inteligencia Artificial, existen requisitos de rendimiento muy estrictos, etc.
 - *Relacionados con la experiencia del equipo de desarrollo:* existe suficiente gente disponible, está comprometido el equipo para toda la duración del proyecto, el personal está adecuadamente formado, etc.
- Definir para cada riesgo identificado un impacto con el fin de hacer una distinción de los riesgos verdaderamente importantes.

Se puede encontrar más al respecto en [DMI 99].

C. Productos de trabajo

Lista de riesgos para el proyecto

Para la descripción de los riesgos contenidos en el lista se propone la siguiente plantilla:

No de Orden. Nombre del riesgo	
Descripción	Una breve descripción del riesgo.
Categoría	Por ejemplo: técnico, del cliente, etc.
Impacto	Componente del riesgo que se afecta: Rendimiento, Costos, Mantenibilidad, Planificación.
Efecto	Nivel en el cual afecta al proyecto: Despreciable, marginal, crítica, catastrófica.
Estrategia	Definición de las estrategias iniciales para afrontar el riesgo.

Tabla 4.5 Plantilla para la descripción de riesgos

4.1.2.4 Etapa IV - Planeación inicial del proyecto

Propósito

El propósito de la planeación inicial del proyecto es establecer planes razonables para administrar el proyecto. Esto involucra el desarrollo de estimaciones para el trabajo a ser ejecutado, estableciendo los compromisos necesarios, y definiendo el plan para ejecutar el trabajo.

Actividades

1. Elaborar el plan de trabajo

Actividad 1. Elaborar el plan de trabajo

A. Descripción

Esta actividad tiene como fin definir una estimación inicial costos para el desarrollo de las actividades futuras del proyecto, tiempo, y además estimar el esfuerzo y los recursos para la realización de las actividades por parte de los responsables. La realización de estos estimativos es responsabilidad del Director del proyecto apoyándose en los criterios que puedan suministrarle los encargados del análisis y diseño del sistema de comercio electrónico.

B. Técnicas

Estimación de recursos computacionales críticos

Para establecer un listado de recursos computacionales críticos se plantea hacer una definición de la categoría de los recursos requeridos para el proyecto y que pueden incidir en forma definitiva en la construcción del sistema de comercio electrónico. La siguiente es una posible categorización de recursos computacionales respecto a su tipo:

Categoría	Tipo	Ejemplos
Hardware	Plataformas	- Estaciones de Trabajo o PC y el número requerido. - Servidores, especificación de memoria, capacidad de procesamiento y canales de comunicación.
	Conexiones de Internet	MODEM, ADSL, etc.
Software	Sistemas Operativos	Windows family, Linux, Unix, Mac.
	Herramientas de desarrollo	Compiladores, Entornos de desarrollo, Generadores de código, herramientas de diseño gráfico, herramientas para modelado de software, motores de bases de datos, etc.
	Herramientas de Gestión	Herramientas de gestión de requisitos, herramientas para automatización de pruebas, software para monitoreo de tareas, software para gestión de versiones, hojas de cálculo, Procesadores de texto, etc.
	Software para seguridad	Sistemas para detección de intrusiones, software para encriptación, Antivirus, etc.
	Otros	-Software cliente o de servidor para transacciones (SET, Monederos electrónicos). - Certificados
Otros	-	Pasarelas de pago, Sistemas para procesamiento de transacciones, etc.

Tabla 4.6 Categorización de recursos computacionales

Para efectos de estimaciones generales de planeación pueden utilizarse las técnicas mencionadas en la actividad 2 de la etapa I de esta fase.

C. Productos de trabajo

Plan de trabajo para el proyecto, que contiene:

1. Relación de actividades a realizar, duración estimada y esfuerzo
2. Lista de recursos computacionales críticos para el proyecto
3. Descripción del equipo humano y asignación de responsabilidades
4. Estimación de los costos.

4.1.2.5 Etapa 5 - Determinación preliminar de viabilidad del proyecto

Propósito

Evaluar todo el trabajo realizado y elaborar el Caso Inicial del Negocio concluyendo con una definición de si es viable continuar con el proyecto.

Actividades

1. Evaluar los productos de trabajo
2. Elaborar el caso inicial de negocio

Actividad 1. Evaluar los productos de trabajo

A. Descripción

El desarrollo de esta actividad permite comprobar a través de los criterios de evaluación establecidos en la primera etapa, si al finalizar la fase se realizaron correctamente todas las actividades propuestas y si los resultados obtenidos con la elaboración de los productos satisfacen los objetivos.

B. Productos de Trabajo

Lista de comprobaciones para el Estudio Preliminar, con indicación de cumplimiento con cada uno de los criterios de evaluación establecidos.

Actividad 2. Elaborar el caso inicial de negocio

A. Descripción

El Director del proyecto realiza una primera aproximación de la apuesta económica y los beneficios que se van a conseguir con la ejecución del proyecto. La declaración de una posición clara sobre

costos y agenda se deja para la fase de Formulación del Proyecto donde se entiende el sistema de comercio electrónico en más detalle, por el momento el Director del proyecto solo necesita justificar la continuación del proyecto

B. Técnicas

Método histórico

La elaboración del caso del negocio es una tarea compleja y lleva consigo gran cantidad de trabajo investigativo que involucra aspectos fundamentales tales como la apuesta económica y la recuperación de la inversión. Si bien se cuenta con una primera aproximación de costos en esta primera fase, la validez de este cálculo depende mucho de la experiencia del equipo de desarrollo con proyectos similares y el conocimiento del dominio.

Plan de negocios

Lo relacionado con la justificación económica del proyecto es una tarea compleja ya que los sistemas de comercio electrónico conjugan aspectos relacionados con la promoción, venta y distribución de productos en un determinado mercado. Por lo tanto es fundamental cuando se emprende un proyecto con esta finalidad hacer un análisis serio del mercado a través de un *plan de negocios*, el cual permitirá vislumbrar las oportunidades, ventajas y riesgos que la nueva situación de negocio genera para la empresa.

C. Productos de trabajo

Caso del negocio, que establece claramente:

1. Beneficios que se quieren lograr.
2. Costo estimado para construcción del sistema de comercio electrónico.
3. Duración estimada para el desarrollo del proyecto
4. Apreciación sobre la conveniencia del proyecto.

4.2 FORMULACIÓN DEL PROYECTO

En este apartado se presentan los lineamientos a seguir que esta aproximación de desarrollo propone para la segunda fase de la construcción de una aplicación de comercio electrónico sobre plataforma web.

4.2.1 Descripción general de la fase

4.2.1.1 Objetivos generales

Configuración de una arquitectura de referencia que guiará el desarrollo del sistema de comercio electrónico y la eliminación y mitigación de riesgos críticos.

4.2.1.2 Objetivos específicos

- Capturar y estructurar los requisitos esenciales del sistema
- Definición de la arquitectura para la construcción del sistema
- Gestionar los riesgos del proyecto
- Completar el plan para el desarrollo del proyecto
- Completar el caso de Negocio.

4.2.1.3 Justificación

La fase de la Formulación del Proyecto es importante ya que en ella se completa la captura de la mayor parte de los requisitos para el desarrollo del sistema de comercio electrónico, los cuales servirán para tener una visión clara de sus responsabilidades fundamentales y por lo tanto especificar una arquitectura que soporte el desarrollo del mismo. Una vez cumplido este objetivo se tendrán criterios suficientes para determinar la viabilidad del proyecto.

4.1.1.4 Etapas de referencia

1. Planeación de la Formulación del Proyecto
2. Refinamiento de Requisitos
3. Definición de un Modelo de Objetos Conceptual para el Sistema
4. Descripción de la Arquitectura de Referencia para el Sistema
5. Gestión de Riesgos
6. Planeación del Proyecto
7. Actualización del caso del negocio

4.1.1.5 Productos de trabajo de la Formulación del Proyecto

1. Plan de trabajo para la Formulación del Proyecto
2. Plan de gestión de configuración
3. Modelo de negocio (Actualizado)
4. Modelo de casos de uso (Actualizado)

5. Documento de prioridad de casos de uso
6. Documento de descripción de paquetes del sistema
7. Documento de análisis para casos de uso esenciales
8. Lista de requisitos no funcionales(Actualizada)
9. Modelo de despliegue (Actualizado)
10. Documento de descripción de subsistemas
11. Descripción inicial de clase de diseño
12. Lista de riesgos y estrategias para el proyecto (Actualizada)
13. Plan de trabajo para el proyecto (Actualizado)
14. Lista de Comprobaciones para la Formulación del Proyecto
15. Caso del negocio (Actualizado)

4.2.2 Etapas de referencia

4.2.2.1 Etapa I - Planeación de la formulación del proyecto

Propósito

Organizar y definir actividades, tiempos, objetivos y recursos para el desarrollo para la Formulación del Proyecto, además de definir los mecanismos para asegurar el seguimiento de la fase.

Actividades

1. Elaborar el plan de trabajo
2. Establecer los criterios de evaluación
3. Definir los mecanismos de seguimiento y control.

Actividad 1. Elaborar el plan de trabajo

A. Descripción

Esta actividad define el orden de las actividades requeridas para realizar la etapa de Formulación del Proyecto y establecer la organización del personal involucrado, determinar los costos y el esfuerzo requerido. Es verdad que ya en la fase anterior (Estudio Preliminar) se realizó una aproximación de los puntos en consideración, esta fase se considera fundamental para definir, por parte del Director del proyecto, completamente dichos puntos.

B. Técnicas

Para una descripción más detallada de estas técnicas y sus respectivas referencias se pueden consultar las técnicas de la actividad 2 de la etapa I del Estudio Preliminar.

C. Productos de trabajo

Plan de trabajo para la formulación del proyecto, documento que contiene los puntos definidos para la actividad 2 de la etapa I del Estudio Preliminar.

Actividad 2. Establecer los criterios de evaluación

A. Descripción

En esta actividad se definen los aspectos fundamentales que deben obtener evaluación satisfactoria para poder dar por terminado la fase de Formulación del Proyecto. Esta actividad es responsabilidad del Director del proyecto apoyándose en los conceptos sobre el sistema de comercio electrónico propuestos por el Analista y Arquitecto del sistema

B. Técnicas

Para definir los criterios de evaluación de la fase, o si se han considerado iteraciones para el propósito de hacer aproximaciones incrementales de desarrollo, no se tiene una forma única, esto depende de las características particulares del proyecto, sin embargo se pueden definir algunos criterios en función de los objetivos de la fase.

Especificación de requisitos

- ¿Se han identificado los requisitos, actores, y casos de uso necesarios para diseñar una línea base de la arquitectura del sistema, o para poder proponer una arquitectura conocida que soporte los requisitos?
- ¿Se han detallado lo suficiente los requisitos como para satisfacer los objetivos de esta fase?

Definición de la arquitectura

- ¿Satisface la línea base para la arquitectura, o la arquitectura existente, los requisitos determinados para el sistema y sus usuarios?
- ¿Se han adaptado suficientemente los casos de uso a la arquitectura ya existente, en el caso de que el cliente la posea?
- ¿Se ha ajustado la arquitectura para encajar con sistemas heredados existentes en la organización tales como sistemas contables o sistemas de bases de datos?

- ¿Se ha investigado lo suficiente con el fin de garantizar que los sistemas heredados suministran funcionalidades que se puedan reutilizar?
- ¿Es la línea base para la arquitectura del sistema lo suficientemente robusta para soportar las demandas de crecimiento del sistema con el tiempo y el crecimiento del número de usuarios del mismo?
- ¿Es la línea base de la arquitectura coherente con los requisitos no funcionales tales como disponibilidad, seguridad, carga, tiempo de respuesta, etc.?

Gestión de riesgos

- ¿Se han identificado los riesgos críticos para el proyecto?
- ¿Se han eliminado convenientemente los críticos de acuerdo a las estrategias planteadas o los planes propuestos para este fin?
- ¿Se consideran manejables los riesgos que aún subsisten?
- ¿Se ha creado un plan de contingencia para afrontar los riesgos que no se pudieron eliminar?

Validez del caso de negocio

- ¿Está el proyecto suficientemente bien definido en cuanto a precio del contrato, cronograma del proyecto y calidad del sistema a construir?
- ¿Indica el caso del negocio un retorno de inversión satisfactorio para el Cliente?
- ¿Estamos listos para comprometernos con un contrato de precio fijo?

C. Productos de trabajo

Lista de comprobaciones para la formulación del proyecto₁ contiene una lista de comprobaciones para la Formulación del Proyecto similar a la presentada en las técnicas de la actividad 2 de esta fase con las modificaciones específicas que se puedan añadir según el criterio de los gestores del proyecto y sus características específicas

Actividad 3. Definir los mecanismos de seguimiento y control

A. Descripción

Esta actividad es necesaria para asegurar un correcto seguimiento de los productos de trabajo generados en el transcurso de la Formulación del Proyecto. Se debe tener en cuenta que en el Estudio Preliminar se desarrolló una primera versión del repositorio de líneas base para los productos de trabajo del proyecto. Por lo tanto se debe partir de las versiones estables de estos productos de trabajo para proceder a la construcción de nuevas versiones o diferentes componentes o líneas base del repositorio.

Los integrantes del equipo del proyecto involucrados en las actividades deben identificar los productos de trabajo que se van a incluir como nuevas líneas base en esta fase, para que posteriormente con supervisión del Administrador de Configuración sean incluidas en el repositorio.

B. Técnicas

Se pueden encontrar técnicas y referencias en la actividad 4 de la etapa I de la fase correspondiente al Estudio Preliminar.

C. Productos de trabajo

Plan de gestión de configuración, que contiene los puntos definidos para la actividad 4 de la etapa I del Estudio Preliminar.

4.2.2.2 Etapa II - Refinamiento de requisitos

Propósito

Completar la captura de los requisitos esenciales para la definición del sistema con el objetivo de que estos sirvan para definir sus responsabilidades principales.

Actividades

1. Completar el estudio de la organización
2. Recopilar requisitos para el sistema
3. Determinar la prioridad de los casos de uso

Actividad 1. Completar el estudio de la organización

A. Descripción

Mediante esta actividad se busca tener una visión completa de los procesos de negocio existentes en la organización. El Analista del sistema procura capturar los detalles adicionales importantes de los procesos de negocio que va a soportar el sistema y analiza el impacto que puede tener en el negocio o en sus procesos la automatización o soporte de tareas. Se debe tener en cuenta que en la etapa anterior (Estudio Preliminar) ya se realizó una primera aproximación del Modelo de Negocio de organización, por lo tanto el Analista del sistema debe tomarlo como punto de partida para obtener las nuevas versiones que puede generar en esta etapa de desarrollo.

B. Técnicas

Revisar la actividad 1 de la etapa 2 del Estudio Preliminar para ver detalles de las técnicas útiles en esta actividad.

C. Productos de trabajo

Modelo de negocio (Actualizado), compuesto por los puntos definidos en la actividad 1 de la etapa 2 del Estudio Preliminar.

Actividad 2. Recopilar requisitos para el sistema

A. Descripción

En esta actividad se pretende recopilar los requisitos funcionales que no se capturaron en el Estudio Preliminar y verificar la consistencia de los ya capturados hasta el momento de acuerdo a la nueva información recolectada. El Analista del sistema pretende complementar y detallar los requisitos del sistema de forma tal que no existan inconsistencias con lo planteado inicialmente por el cliente.

B. Técnicas

Casos de uso

Ya de la fase anterior se produjo un primer modelo de casos de uso, en esta etapa se pretende generar versiones estructuradas y detalladas de este modelo. Para esto el Analista del sistema identifica los casos de uso y los actores adicionales. Referente al número de casos de uso a tener en cuenta se puede decir que se necesita identificar alrededor de 80 por ciento, detallar una fracción de casos de uso interesantes desde el punto de vista de riesgo o complejidad que presenten y por lo tanto hacer un análisis de estos casos significativos. Para lograr un modelo de casos de uso estructurado, modificable y mantenible el Analista del sistema puede utilizar mecanismos como la extensión y la generalización. Se puede encontrar más información sobre estos mecanismos en [Dimu 01].

Una descripción más detallada de la técnica en el apartado 2.2.2 del Estudio Preliminar.

Prototipado de interfaces

La realización de prototipos exploratorios se hace con el fin de poder visualizar la manera en que los actores del sistema llevan a cabo los casos de uso que necesitan de ellas, como resultado se pueden tener una comprensión más amplia de los casos de uso, y por lo tanto de los requisitos,

con lo cual eventualmente se pueden descubrir nuevos comportamientos y requisitos para el sistema. Una vez diseñados los prototipos de interfaces éstos pueden ser entregados a los clientes y usuarios finales para ser validados con el fin de tener una realimentación por parte de ellos o definir nuevos requisitos. El Diseñador de interfaces gráficas diseña los prototipos de las interfaces si son necesarias para observar su funcionamiento e interacción con los usuarios del sistema.

Si bien se pueden diseñar interfaces propias para la aplicación teniendo en cuenta los requisitos capturados a través de casos de uso y requisitos adicionales para las interfaces de usuario, en este campo es muy importante la parte de rehusos mediante el uso de patrones de interfaz. Un patrón es una solución a un problema común encontrado en un dominio específico. Para el caso de las interfaces de usuario típicas de aplicaciones de comercio electrónico se pueden encontrar muchos de estos patrones examinando sitios comerciales tales como tiendas virtuales, sitios de subasta, o cualquier otro tipo de aplicación comercial en Internet que ofrezca productos o servicios. El uso de patrones entonces cobra importancia a la hora de diseñar la interfaz ya que permite un rápido y eficiente prototipado, además de permitir visualizar posibles funcionalidades tales como características de personalización de la interfaz de usuario y formas de estructurar los procesos de compra de productos, estos sin duda aspectos críticos a tener en cuenta cuando se diseñan sistemas de comercio electrónico actualmente.

Algunos de los patrones de interfaces para sitios de comercio electrónico cubren aspectos tales como formas de anunciar los productos, presentación de productos, comparación de productos, registro de clientes, etc. Se pueden encontrar algunos patrones de interfaces en el Anexo A de la monografía del MICACE.

C. Productos de trabajo

Modelo de casos de uso (Actualizado), el cual contiene:

1. Diagrama de casos de uso del sistema
2. Descripción de actores.
3. Prototipos de Interfaces de usuario (Opcionales).

Actividad 3. Determinar la prioridad de los casos de uso

A. Descripción

Si bien se ha construido en la presente fase un modelo de casos de uso aun más completo que el conseguido en el Estudio Preliminar, sin embargo hay que destacar que las decisiones asociadas al proyecto se encuentran influenciadas por las prioridades asociadas a los riesgos que se hayan

percibido y por la forma que se pretende enfrentar el desarrollo del proyecto (que casos de uso se implementarán en las primeras iteraciones y cuales se implementarán en las posteriores). El Arquitecto del sistema es el encargado de organizar los casos de uso de acuerdo a estos criterios.

B. Técnicas

Para asignar una correcta prioridad a los casos de uso en esta Fase se debe tener en cuenta que los casos de uso con mayor relevancia (esenciales) para la Formulación del Proyecto son los siguientes:

- Los casos de uso que ayudan a mitigar riesgos importantes
- Los casos de uso que son fundamentales para los usuarios del sistema
- Los casos de uso que ayudan a cubrir todas las funcionalidades significativas del sistema de forma que ningún aspecto importante quede sin estudio.

Una vez identificados los casos de uso esenciales, estos se describen detalladamente con el fin de tener una visión amplia de sus funcionamientos e interacciones con los actores, esto se realiza en esta fase del desarrollo a través de la descripción de los escenarios. Para esto se tomó como guía la plantilla de descripción de escenarios propuesta en [Melo 01] con algunas modificaciones, esta se presenta a continuación:

<p>Número de orden del caso de uso.</p> <p>Nombre del caso de uso</p> <p>Iniciador: [persona que dispara el caso de uso]</p> <p>Precondición: [condiciones necesarias para disparar el caso de uso]</p> <p>Flujo de eventos</p> <p>Flujo principal : [secuencia de actividades]</p> <p>1. _____</p> <p>2. _____</p> <p>3. _____</p> <p>Postcondiciones: [condiciones del sistema después de disparar el caso de uso]</p> <p>_____</p> <p>_____</p> <p>Flujos alternativos: [secuencias alternas al flujo principal]</p> <p>_____</p> <p>Excepciones : [posibles condiciones que alteran el curso normal]</p> <p>_____</p> <p>Recursos especiales utilizados: [componentes relevantes involucrados]</p> <p>_____</p> <p>_____</p> <p>Interfaces de Usuario relacionadas : [Lista de Interfaces de Usuario relacionadas]</p> <p>_____</p> <p>_____</p>

Figura 4.2 Plantilla para la descripción de escenarios para la Formulación del Problema

Quizá sea conveniente en esta fase hacer una *descripción detallada* de los escenarios con el fin de comprender mejor su dinámica. Para este fin se pueden utilizar diagramas tales como los de Actividad, Secuencia, y Colaboración. Se puede encontrar más información sobre estos diagramas en [Booch 99].

C. Productos de trabajo

Documento de prioridad de casos de uso, que contiene:

1. Lista de casos de uso esenciales para el sistema
2. Descripción de escenarios para los casos de uso esenciales

4.2.2.3 Etapa III - Definición de un modelo de objetos conceptual para el sistema

Propósito

Refinar y estructurar los requisitos esenciales del sistema mediante abstracciones conceptuales que permitan obtener una estructura centrada en el mantenimiento, en aspectos tales como flexibilidad ante los cambios y la reutilización. Esta orientación permitirá obtener mejores criterios en el momento de definir los factores económicos en el análisis del Negocio para el proyecto, ya que los requisitos esenciales terminan siendo mejor entendidos. Como producto de la ejecución de las actividades de la presente etapa se obtiene una representación conceptual del sistema que se denomina *Modelo de Análisis*.

Actividades

1. Analizar la arquitectura
2. Analizar casos de uso
3. Capturar requisitos no funcionales

Actividad 1. Analizar la arquitectura

A. Descripción

Como objetivo del estudio de los requisitos esenciales a este nivel, se pretende extender el análisis de la arquitectura del sistema abordado en el Estudio Preliminar con el fin que pueda garantizar para servir como línea base de la arquitectura ejecutable. En esta actividad el Arquitecto del Sistema realiza una partición de alto nivel del sistema teniendo en cuenta los requisitos esenciales para el sistema (casos de uso esenciales), el glosario de términos, y el modelo de negocio obtenidos en la etapa "Refinamiento de Requisitos" de la presente fase. Además de la identificación de los paquetes de análisis el Arquitecto del Sistema configura inicialmente una distribución coherente de las partes que van a componer el sistema.

B. Técnicas

Paquetes de análisis

Debido a que ya se tiene una visión detallada de los casos de uso, se puede hacer una agrupación de ellos clasificándolos por medio de paquetes de análisis. La utilización de paquetes en este caso se hace con el fin de organizar el Modelo de Análisis del sistema en piezas pequeñas y manejables. El paquete representa una porción de este modelo suficientemente manejable, entendible, y significativa. Un paquete de análisis puede ser un agregado de clases de análisis,

realizaciones de casos de uso, y paquetes de análisis en forma recursiva. Debido a que los requisitos funcionales del sistema se han capturado en forma de casos de uso y que además de ellos ya se han identificado aquellos que son fundamentales para el sistema, la forma de identificar los paquetes de análisis es asignar de forma adecuada los casos de uso a un paquete concreto. Algunos criterios para realizar asignaciones apropiadas podrían ser los siguientes:

- Los casos de uso para dar soporte a un determinado proceso de negocio
- Los casos de uso para dar soporte a un determinado actor del sistema
- Los casos de uso que están relacionados mediante relaciones de generalización y extensión
- Los casos de uso que pertenezcan a funcionalidades similares

De esta forma los paquetes conformados de esta forma localizan los cambios tanto en un proceso de negocio, en el comportamiento de los actores y en un conjunto de casos de uso estrechamente relacionados [Jacobson 99].

Patrones de arquitectura

La utilización de patrones de arquitectura puede ayudar a configurar una distribución de paquetes para el sistema. Un patrón frecuentemente utilizado en diversidad de sistemas es el patrón Layers. Este patrón define como organizar el sistema en capas, lo que significa que los componentes de una capa solo pueden hacer referencia a los componentes de las capas inmediatamente inferiores. La importancia del patrón radica en que simplifica la comprensión y la organización del desarrollo de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no son concientes de ningún detalle de las superiores, un diagrama explicativo de este patrón se puede observar en la Figura 4.3.

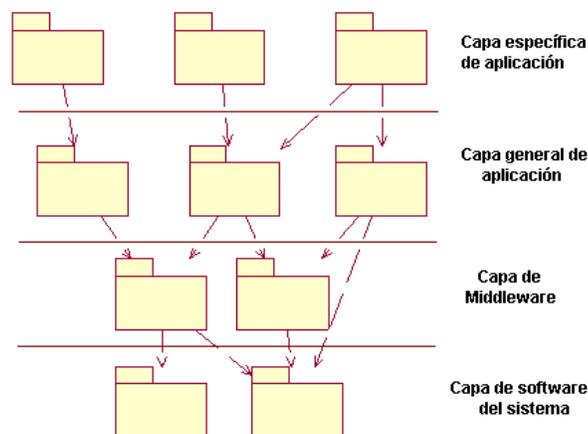


Figura 4.3 Capas propuestas por el Patrón Layers

La utilización de este patrón puede ayudar al Arquitecto del sistema a identificar paquetes que se encuentren distribuidos en las diferentes capas propuestas.

Respecto a este tipo de patrones se puede anotar que si bien se pueden encontrar patrones que describan el sistema (patrones predominantes), estos muchas veces pueden combinarse con otros tipos de patrones tales como los *patrones de diseño* cuando se especifica el sistema. Por ejemplo en una capa específica del patrón Layers puede utilizarse patrones de diseño que se consideren convenientes para reutilizar experiencias de diseño. Se puede encontrar información acerca de esta técnica en [Jacobson 99].

Paquetes de servicio

Aparte de proporcionar casos de uso a sus actores, todo sistema de software proporciona un conjunto de servicios a sus clientes. Los servicios representan un conjunto coherente de acciones relacionadas funcionalmente que se utiliza en varios casos de uso. Este concepto de servicio esta soportado a través de los *paquetes de servicio* [Jacobson 99], la cual permite estructurar el sistema en función de los servicios que proporciona. Los paquetes de servicio son paquetes de análisis en un nivel más bajo de la jerarquía de agregación. Sobre los paquetes de servicio se puede decir:

- Contienen un conjunto de clases relacionadas funcionalmente
- Son indivisibles. Cada cliente obtiene o bien todas las clases o ninguna del paquete.
- Cuando se lleva a cabo un caso de uso, es posible que sean participantes uno o más de un paquete de servicio.
- Estos pueden ser dependientes unos de otros.

El Arquitecto del sistema al trabajar a este nivel lo esta haciendo con las necesidades colectivas de los casos de uso, esto le permite distinguir paquetes genéricos tales como los relacionados con persistencia, interfaces de usuario o distribución de objetos. Una descripción más detallada de este tema se puede encontrar en [Jacobson 99].

C. Productos de trabajo

Documento de descripción de paquetes del sistema, que contiene:

1. Diagrama de paquetes del servicio
2. Relación de las clases de análisis contenidas en los paquetes

Actividad 2. Analizar casos de uso

A. Descripción

Muchos de los casos de uso pueden ser comprensibles de la forma como se describen en el modelo de casos de uso para el sistema de comercio electrónico y en la descripción de escenarios, sin embargo es necesario refinar los casos de uso esenciales para el sistema de comercio electrónico, a través de clases conceptuales o de análisis que solamente existen en el ámbito de los requisitos y que generalmente no se implementan de forma directa en el dominio de la aplicación. Para realizar la actividad el Ingeniero de casos de uso identifica las clases de análisis cuyos objetos son necesarios para llevar el flujo de sucesos de los casos de uso, teniendo en cuenta aquellos que dependen unos de otros.

B. Técnicas

Patrones de análisis

Un patrón de análisis que se puede utilizar para identificar las clases de análisis necesarias para realizar los casos de uso es el patrón *Boundary – Control - Entity*, el cual facilita la distinción del comportamiento específico de los casos de uso, las clases de interfaz, y las clases genéricas. El patrón define tipos de clases como se indica en la Tabla 4.9:

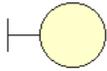
Nombre de la clase	Estereotipo UML	Descripción
Interfaz		Se utilizan para modelar la interacción entre el sistema de comercio electrónico y sus actores. Esta interacción frecuentemente implica recibir información y peticiones de los usuarios y los sistemas externos
Entidad		Modelan información que poseen un existencia prolongada y que frecuentemente es persistente.
Control		Representan coordinación, secuencia, transacciones, y control de otros objetos. Se usan para encapsular el control del caso de uso en concreto.

Tabla 4.7 Descripción de las clases de análisis propuestas por el patrón Boundary-Control-Entity.

Se propone tener presente las siguientes recomendaciones con el fin de identificar las clases mediante el uso de este patrón de análisis:

- Identificar las clases de entidad mediante el estudio en detalle de la descripción del caso de uso (escenarios) y del modelo de negocio actualizado que se tiene, para después considerar qué información debe utilizarse y manipularse en la realización del caso de uso¹.
- Se puede identificar una clase interfaz central por cada actor humano que interactúe con el sistema de comercio electrónico, esta representa la interfaz de usuario central con la que interactúa el actor. Sin embargo, se puede presentar que muchas clases interfaz para atender al mismo actor.
- Identificar una clase de interfaz central para cada actor que sea un sistema externo, y dejar que esta represente la interfaz de comunicación. Si las comunicaciones del sistema de comercio electrónico se dividen en varios niveles de protocolo, se puede especificar en el modelo de análisis algunos de esos niveles (una clase interfaz para cada nivel de interés).
- Identificar una clase de control responsable del tratamiento del control y de la coordinación de la realización del caso de uso, para luego refinarla de acuerdo a los requisitos del caso de uso.

Para cada clase de análisis identificada debe asignársele un nombre, indicar su tipo (control, entidad o frontera) y establecer sus responsabilidades.

Se puede encontrar más acerca de esta técnica en [Jacobson 99].

Diagramas de colaboración

Los diagramas de colaboración muestran las interacciones entre objetos de análisis. Estos diagramas destacan la organización de los objetos que participan en la interacción. El diagrama se construye colocando los objetos que participan en la realización del caso de uso como los nodos de un grafo. Después se representan los enlaces que conectan los objetos como arcos de un grafo. Los enlaces se pueden adornar con los mensajes que envían y reciben los objetos. Estos diagramas son importantes debido a que permiten identificar requisitos y responsabilidades sobre los objetos. Para una descripción más detallada de la técnica se puede consultar [Booch 99].

Es de resaltar que como ya se han descrito en detalle los escenarios de los casos de uso esenciales para el sistema de comercio electrónico, se puede utilizar los diagramas de colaboración para analizar los flujos y subflujos de los casos de uso.

C. Productos de trabajo

¹ Una **realización del caso de uso** es una colaboración que describe como se lleva a cabo y se ejecuta un caso de uso en función de las clases de análisis y sus objetos.

Documento de análisis para casos de uso esenciales, que contiene:

1. Descripción de las clases de análisis identificadas
2. Diagramas de colaboración para casos de uso esenciales

Para hacer la descripción de las clases de análisis fundamentales se propone utilizar la siguiente plantilla:

Nombre de la clase	Es un nombre descriptivo para la clase
Tipo	El tipo de clase: entidad, control, interfaz.
Responsabilidades	Las responsabilidades de las clases pueden obtenerse observando el papel de la clase en las realizaciones de casos de uso que participa.

Tabla 4.8 Plantilla para la descripción de clases de análisis

Actividad 3. Capturar requisitos no funcionales

A. Descripción

Con esta actividad se pretende completar en mayor parte los requisitos no funcionales para el sistema de comercio electrónico. El Arquitecto del sistema es el encargado de recopilar los requisitos no funcionales importantes restantes puesto que ya se tiene una descripción más detallada del sistema de comercio electrónico.

B. Técnicas

Ya que se tiene una descripción detallada de los aspectos fundamentales del sistema de comercio electrónico, los requisitos no funcionales pueden capturarse estudiando las realizaciones de casos de uso y las responsabilidades de las clases de análisis. Se pueden tener por ejemplo identificación de requisitos relacionados con:

- Características de persistencia
- Distribución y concurrencia
- Características de seguridad
- Aspectos de *confiabilidad* del sistema tales como Tolerancia a fallos – características del sistema que permitan mantener un nivel de funcionamiento cuando se presentan fallos, y recuperabilidad - características definidas en el sistema que permitan reestablecer el nivel de funcionamiento y datos cuando se presentan fallos.

- Disponibilidad
- Gestión de transacciones

Cada requisito debe ser descrito en forma detallada con el fin de identificar todas sus características, esto con el fin de tener una entrada consistente para la construcción del sistema de comercio electrónico.

C. Productos de trabajo

Lista de requisitos no funcionales (actualizada).

Para la descripción de los requisitos no funcionales se propone usar la plantilla de la Tabla 4.6.

4.2.2.4 Etapa IV - Descripción de la arquitectura de referencia del sistema

Propósito

Establecer la línea base de la arquitectura del sistema de comercio electrónico para todo su ciclo de vida, a través de la definición de subsistemas y las interfaces entre estos, la actualización del modelo de despliegue respecto a la primera versión desarrollada en el Estudio Preliminar y la identificación inicial de las clases de diseño.

Actividades

1. Identificar nodos y configuraciones de red.
2. Definir subsistemas e interfaces
3. Identificar clases de diseño

Actividad 1. Identificar nodos y configuración de red

A. Descripción

Debido a que los sistemas de comercio electrónico constituyen generalmente un sistema distribuido, el Arquitecto del sistema necesita identificar los nodos y las configuraciones de red para modelar la vista de despliegue estática del sistema. Esta cubre principalmente la distribución, entrega e instalación de las partes que configuran el sistema físico.

Una sistema de comercio electrónico deberá interactuar a menudo con sistemas ya creados (legados) y en funcionamiento, cuando se trata de una organización que migra sus procesos de negocio a Internet. Si por el contrario se trata de una organización que se constituye directamente basada en la red, sus sistemas con más razón deben ser orientados a un modelo distribuido que

soporte características técnicas como la redundancia de información y los sistemas de protección de datos, o que se adecuen, si los sistemas son contratados, a las ofertas que ofrece el mercado. En ambos casos, la identificación clara de la configuración de su arquitectura permitirá una idea lo suficientemente clara para su posterior implementación. Se puede consultar en [Guías 02] aspectos referentes a características de distribución en sistemas de comercio electrónico.

B. Técnicas

El *Modelo de Despliegue* se constituye en la base para la identificación correcta de la configuración de la red donde se soporte el sistema de comercio electrónico. Se deben detallar en este modelo los aspectos claves para definir la configuración de red:

- Nodos que están involucrados y sus características que pueden influir sobre la *eficiencia* del sistema sobre todo en la tales como capacidades en términos de poder de procesamiento y tamaño de memoria.
- Tipo de conexiones que se existen entre los nodos, y protocolos de comunicación que se utilizan entre ellos.
- Características de las conexiones y protocolos de comunicación, tales como ancho de banda, disponibilidad y calidad.
- Necesidades en cuanto a: Capacidad de procesamiento redundante, migración de procesos, respaldo de datos, etc.

Sin embargo la especificación en algunos casos no puede ser descrita totalmente debido a que los sistemas son contratados, por ejemplo el servidor web, las relaciones entre nodos y las características propias de estos deben ser lo suficientemente detalladas para que no se constituyan en un riesgo futuro.

C. Productos de trabajo

Modelo de despliegue (actualizado)

Actividad 2. Definir subsistemas e interfaces

A. Descripción

Los *subsistemas* constituyen un medio de organización para aplicaciones de gran tamaño o que requieran la descomposición del diseño, además de que pueden dividir el trabajo de diseño en

partes del sistema que pueden ser contratadas o trabajadas independientemente estructurando adecuadamente sus interfaces.

En sistemas de comercio electrónico es muy frecuente tener un sistema de administración de la aplicación o subsistemas contratados, como por ejemplo, la interfaz de gestión de transacciones bancarias, sistemas de búsqueda, etc. Una temprana identificación de estos subsistemas por parte del Arquitecto del sistema ayuda a organizar el trabajo de diseño para la siguiente fase de desarrollo.

B. Técnicas

Una técnica efectiva para la definición de Subsistemas es la utilización de los paquetes de análisis encontrados anteriormente, haciendo uso del patrón Layers o Capas de la arquitectura del sistema. Esta identificación se realiza en dos etapas, identificando primero los subsistemas correspondientes a las capas de aplicación y luego a las correspondientes a las capas middleware y software del sistema [Jacobson 99].

Una vez se encuentran los Subsistemas correspondientes a cada capa de la arquitectura se deben buscar sus dependencias para analizar si ésta se puede traducir en una interfaz². Así si un subsistema tiene una dependencia que apunta hacia él, es probable que deba proporcionar una interfaz que puede ser provista por una clase o bien por otro subsistema. En las capas inferiores de la arquitectura la identificación de interfaces es más sencilla, ya que los subsistemas de estas capas encapsulan productos software, y esos productos suelen tener algún tipo de interfaces definidas

C. Productos de trabajo

Documento de descripción de subsistemas, que contiene:

1. Diagrama de paquetes de diseño del sistema
2. Definición de las interfaces de los subsistemas

Actividad 3. Identificar clases de diseño

A. Descripción

² Una interfaz agrupa operaciones que son utilizadas para definir un servicio de una clase o un componente.

El Arquitecto del sistema Identifica las clases de diseño que sean relevantes para la arquitectura dentro del ciclo de vida del sistema de comercio electrónico, evitando identificar demasiadas clases en esta etapa o adentrándose mucho en detalles.

B. Técnicas

Se pueden identificar las clases de diseño primero a partir de las clases de análisis encontradas anteriormente, utilizando además las relaciones entre estas para identificar un conjunto de relaciones tentativas entre las clases de diseño. En este punto se puede identificar algunas clases activas (clases cuyas instancias son objetos activos tales como procesos o hilos) necesarias para el sistema de comercio electrónico, considerando los requisitos de concurrencia del mismo. Por ejemplo, los requisitos de rendimiento del sistema, como tiempo de respuesta y disponibilidad o de persistencia de conexión a una base de datos; la distribución del sistema sobre los nodos, que puede requerir la existencia de un objeto activo por cada nodo y otros aparte para gestionar la interconexión entre los nodos.

A este nivel se debe describir las clases identificadas a través de sus *atributos* y *operaciones*. Estos son algunos lineamientos a seguir para obtener las operaciones de las clases de diseño:

- Se pueden obtener operaciones a partir de las responsabilidades de las clases de análisis que tengan una *traza* con las clases de diseño. A menudo una responsabilidad puede implicar más de una operación.
- Se pueden identificar operaciones a través de los requisitos especiales que posea una clase de análisis que tenga una traza hacia una clase de diseño.
- A partir de las interfaces que la clase de diseño necesita proporcionar

Para el caso de los atributos pueden ser de ayuda estos lineamientos:

- Considerar los atributos de las clases de análisis que tienen una traza hasta clases de diseño. Es posible que los atributos de las clases de análisis puedan ser descompuestos en más de un atributo en el diseño.
- Manejar la complejidad de los atributos como sigue: Las instancias de los atributos no pueden ser compartidos por varios objetos de diseño; la comprensión de una clase no se debe opacada por los atributos complejos. En ambos casos puede ser de utilidad definir los atributos en clases independientes.

Se debe considerar que las características de las operaciones y atributos encontrados para el diseño deben estar acordes con las capacidades del lenguaje de programación que va a soportar

la implementación de las clases. Esto puede limitar características como *visibilidad* de las clases y *tipos* de atributos.

C. Productos de trabajo

Descripción inicial de las clases de diseño

4.2.2.5 Etapa V - Gestión de riesgos

Propósito

El propósito de esta etapa es complementar la identificación de los riesgos importantes del proyecto con el fin de determinar estrategias para su administración y eliminación.

Actividades

1. Actualizar los riesgos del proyecto
2. Administrar los riesgos

Actividad 1. Actualizar los riesgos del proyecto

A. Descripción

Con la realización de esta actividad se pretende identificar los riesgos que no fueron identificados en el Estudio Preliminar. El Director del proyecto a esta altura del desarrollo tiene una visión más amplia de las necesidades del proyecto ya que se tiene una mejor información de los requisitos del sistema de comercio electrónico que se pretende construir. Con ayuda de los criterios sobre el sistema que el Arquitecto o el Analista le suministran, el Director puede Identificar los diferentes tipos de riesgos con detalle.

B. Técnicas

Con el fin de identificar riesgos adicionales para el proyecto se pueden utilizar las estrategias definidas para la actividad 1 de la etapa III del Estudio Preliminar. De acuerdo a la complejidad que se visualice en la identificación de riesgos se pueden utilizar mecanismos tales como:

- Listas de chequeo
- Entrevistas

- Revisiones formales
- Inspección
- Utilización de la experiencia

Más información al respecto se puede encontrar en [Hall 98]

C. Productos de trabajo

Lista de riesgos para el proyecto (actualizada)

Para la descripción de los riesgos se utiliza la plantilla consignada en la Tabla 4.7.

Actividad 2. Administrar los riesgos

A. Descripción

Una vez que se hayan identificado los riesgos que afectan al proyecto, el Director del proyecto los analiza y los refina con el fin de definir la estrategia para administrarlos según la importancia o impacto definido para cada uno de ellos.

B. Técnicas

Agrupación de riesgos

Si bien se han identificado los riesgos críticos para el proyecto, es necesario hacer un refinamiento de los mismos. Para proyectos en los cuales se tiene identificado gran cantidad de riesgos, se pueden reunir los riesgos de una misma categoría o un cierto grado de similitud con el fin de determinar redundancias en las identificaciones y proceder a eliminarlas.

Árboles de decisión

Con esta técnica se pretende tener una mejor comprensión entre las causas del riesgo y las acciones que permitirán manejar el riesgo en el futuro. “Los árboles de decisión son diagramas que representan en forma secuencial condiciones y acciones; muestra que condiciones se consideran en primer lugar, cuales en segundo lugar y así sucesivamente. Este también permite ver la relación que existe entre cada condición y el grupo de acciones permisibles asociadas con ella” [Senn 98].

Definición de estrategias

Las estrategias para administrar el riesgo dependen de la posición o políticas de la Dirección del proyecto. Estas son algunas posturas sobre los riesgos que se puede encontrar para afrontar la planeación de estos:

- *Convivir con el riesgo:* En esta estrategia se convive con el riesgo en el transcurso del proyecto. Esta estrategia es definida usualmente para riesgos con bajo impacto.
- *Eliminación:* Esta estrategia busca la solución del riesgo mediante la eliminación del mismo. Esta estrategia se define para riesgos que tienen gran impacto en el proyecto y cuyas alternativas para afrontarlo no son muy eficaces para su fin.
- *Mitigación:* esta estrategia busca reducir el impacto de la ocurrencia del riesgo, mediante el estudio de sus causas o estudiando estrategias para anticiparse a su ocurrencia.
- *Contingencias:* Esta estrategia busca definir salidas alternativas ante la ocurrencia del riesgo cuando las estrategias de mitigación no tienen efecto.

Se pueden encontrar algunas estrategias adicionales y más sobre estas técnicas en [Hall 98].

C. Productos de trabajo

Lista de estrategias para riesgos del proyecto

4.2.2.6 Etapa VI - Planeación del proyecto

Propósito

El propósito es realizar una actualización del plan inicial del proyecto realizado en Estudio Preliminar, para tener en cuenta las modificaciones que podrían haberse presentado en cuanto a estimaciones de tiempo, esfuerzo o de recursos necesarios de acuerdo a los avances en el proyecto. Además se tiene ahora una idea más clara del proyecto y las estimaciones antes muy tentativas pueden ahora ser más precisas y de mayor confiabilidad.

Actividades

1. Actualizar el plan de trabajo

Actividad 1. Actualizar el plan de trabajo

A. Descripción

A este nivel de desarrollo del proyecto ya se tiene una visión más clara de los requisitos esenciales del sistema de comercio electrónico, además de una comprensión clara de los riesgos asociados. Con todo esto el Director del proyecto tiene mayores recursos para actualizar las estimaciones hechas para la agenda del proyecto en el Estudio Preliminar y generar proyecciones acertadas que sirvan para soportar la fase de Construcción del sistema.

B. Técnicas

Para hacer las estimaciones del proyecto se puede hacer uso de las técnicas descritas para la actividad 2 de la etapa I del Estudio Preliminar. Se debe prestar especial atención al método histórico debido a que ya se realizaron estimaciones previas para el proyecto. El Director del proyecto debe guardar una relación de las estrategias que usó para generar las estimaciones actuales así como las previas con el fin de que sirvan de guía para futuras y efectivas estimaciones con el uso del método histórico. Esto es importante para la organización ya que permite conocerse más así misma y por lo tanto ayuda a realimentar estrategias para fomentar el crecimiento organizacional.

C. Productos de trabajo

Plan de trabajo para el proyecto (actualizado), que contiene los puntos definidos en la actividad 2 de la fase I del Estudio Preliminar.

4.2.2.7 Etapa VII - Actualización del caso del negocio

Propósito

Determinar la viabilidad del proyecto para la continuación de su desarrollo a través de la revisión del trabajo realizado y de la actualización del caso Inicial del negocio.

Actividades

1. Evaluar los productos de trabajo.
2. Actualizar el caso del negocio.

Actividad 1. Evaluar los productos de trabajo

A. Descripción

El desarrollo de esta actividad permite comprobar a través de los criterios de evaluación establecidos en la primera etapa, si al finalizar la presente fase se realizaron correctamente todas las actividades propuestas y si los resultados obtenidos con la elaboración de los productos satisfacen los objetivos.

B. Productos de trabajo

Lista de comprobaciones para la Formulación del Proyecto, con indicación de cumplimiento con cada uno de los criterios de evaluación establecidos.

Actividad 2. Actualizar el caso del negocio

A. Descripción

La actualización del Caso Inicial del Negocio tiene por finalidad decidir si es viable la continuación del proyecto. El Equipo de Desarrollo debe tener plena certeza de que los riesgos han sido mitigados y que se ha estructurado una línea base para la arquitectura del sistema de comercio electrónico, para iniciar la construcción del producto en los términos que han sido especificados en el contrato con el cliente.

Una vez se han administrado los riesgos de forma conveniente, se tiene una línea base para la arquitectura, se determina la rentabilidad del proyecto, y se han establecido los recursos y necesidades, se evalúa junto con el cliente la viabilidad del proyecto. Esta evaluación debe tener en cuenta los plazos para la entrega del producto, los costos que deben ser asumidos por el cliente para la consecución de los recursos necesarios, los aspectos legales que el cliente debe tener en cuenta para la implementación del sistema de comercio electrónico y la capacidad del Equipo de Desarrollo para llevar a cabo el proyecto. Si el balance es satisfactorio o manejable y si se cumplen satisfactoriamente todos los criterios de evaluación establecidos en la primera etapa de esta fase, se decide declarar junto con el cliente la viabilidad del proyecto y proceder a concertar los términos legales de ser necesario.

B. Productos de trabajo

Caso del negocio (actualizado), que contiene los puntos definidos para la actividad 2 de la fase V del Estudio Preliminar.

4.3 CONSTRUCCIÓN DEL SISTEMA

En este apartado se presentan los lineamientos a seguir propuestos por este modelo de desarrollo para la tercera fase de la construcción de sistemas de comercio electrónico sobre plataforma web.

4.3.1 Descripción general de la fase

4.3.1.1 Objetivo general

Construir la primera versión operacional del sistema de comercio electrónico que cumpla con todos los requisitos acordados.

4.3.1.2 Objetivos específicos

- Terminar el refinamiento de requisitos del sistema mediante refinamiento de los casos de uso y los escenarios restantes.
- Completar el modelo de análisis del sistema
- Completar el diseño del sistema en su estructura funcional y no funcional
- Diseñar la estructura de navegación para el espacio de información en el que se van a mover los usuarios del sistema.
- Implantar estrategias para efectuar el seguimiento de los riesgos del proyecto
- Implementar, integrar y probar los componentes diseñados para el sistema.

4.3.1.3 Justificación

Esta fase es fundamental ya que en ella se materializan todos los requisitos planteados por los entes involucrados en el proyecto, generando un sistema de comercio electrónico completa que cubre las expectativas planteadas.

4.3.1.4 Etapas de referencia

1. Planeación de la Construcción del sistema
2. Especificación final de los requisitos del sistema
3. Construcción del prototipo operacional del sistema
4. Planeación del proyecto

4.3.1.5 Productos de trabajo para la Construcción del Sistema

1. Plan de Trabajo para la Construcción del Sistema
2. Lista de comprobaciones para el estudio preliminar
3. Plan de gestión de configuración
4. Modelo de casos de uso
5. Modelo de análisis
6. Modelo de diseño

7. Modelo de despliegue
8. Modelo Navegacional
9. Modelo de implementación
10. Modelo de pruebas
11. Lista de riesgos y estrategias (actualizada)
12. Caso del Negocio (actualizado)
13. Manuales para el sistema
14. Plan de trabajo para la Validación del sistema

4.3.2 Etapas de referencia

4.3.2.1 Etapa I - Planeación de la construcción del sistema

Propósito

Organizar y definir actividades, tiempos, objetivos y recursos para el desarrollo para la Construcción del Sistema, además de definir los mecanismos para asegurar el seguimiento de la fase.

Actividades

1. Elaborar el plan de trabajo
2. Establecer los criterios de evaluación
3. Definir los mecanismos de seguimiento y control.

Actividad 1. Elaborar un plan de trabajo

A. Descripción

En esta actividad se definen el orden de las actividades necesarias para efectuar la Construcción del sistema, la organización del personal, determinar el costo y el esfuerzo requerido. Si bien se cuenta con un esbozo temporal para las tareas que se van a realizar a partir de las consideraciones sobre planeación que se hicieron en la fase de Formulación del Proyecto, es probable que el Director del Proyecto tenga que modificar algunas de las estimaciones debido a cambios en el alcance del sistema causados por ejemplo a limitación de fondos requeridos para la inversión en el proyecto por parte del cliente o incluso a limitaciones temporales exigidas por estrategias de mercado o marketing.

Se debe tener en cuenta que en la fase de Construcción el personal involucrado en el proyecto se incrementa ya que el sistema crece hasta su primera versión entregable al cliente. En esta fase se pueden requerir además de la participación de algunos de los integrantes del equipo ya incluidos, otros como diseñadores y expertos en interfaces gráficas, expertos en seguridad para el caso de transacciones comerciales e información sensible a ser alterada.

B. Técnicas

Para la realización de las estimaciones correspondientes a esta actividad se pueden utilizar las técnicas definidas para la actividad 2 de la etapa I del Estudio Preliminar.

C. Productos de trabajo

Plan de trabajo para la Construcción del sistema, documento que contiene los puntos descritos para la actividad 2 de la etapa I del Estudio Preliminar.

Actividad 2. Establecer los criterios de evaluación

A. Descripción

En esta actividad se definen los aspectos fundamentales que deben obtener evaluación satisfactoria para poder dar por terminado la Construcción del sistema. Se debe tener en cuenta que en la fase de Construcción se implementa el conjunto de casos de uso del sistema en su totalidad, por lo tanto el Director del proyecto además de haber definido los subconjuntos de casos de uso a implementar en las iteraciones de esta fase, debe acordar los criterios de evaluación al final de las iteraciones.

B. Técnicas

Si bien la asignación de criterios de evaluación para llevar control de las iteraciones son propios de las condiciones específicas del proyecto, en general se pueden distinguir los siguientes puntos a tener en cuenta como criterios de evaluación de esta fase:

Conformidad entre implementación y requisitos

- ¿Satisface la implementación los requisitos funcionales y no funcionales especificados en los casos de uso?

Material de apoyo para usuarios finales

- ¿Se tienen preparadas ayudas, manuales o documentos de apoyo para los usuarios finales (Administrador de contenidos, Administrador de Inventarios, Administrador de Pedidos, etc.) del sistema?
- ¿Son suficientes las ayudas para soportar la validación de estas funcionalidades del sistema por parte de dichos usuarios?

Referentes a la primera versión del sistema

- ¿Se tiene preparado un plan para la etapa de validación del sistema?
- ¿Se tienen completos los componentes finales (scripts, ejecutables, binarios, base de datos, etc.) que conforman la primera versión del sistema?
- ¿Se tiene los modelos del sistema completos?
- ¿Se tiene la descripción de arquitectura del sistema actualizada?

C. Productos de trabajo

Lista de comprobaciones para la Construcción del sistema, contiene una lista de comprobaciones similar a la presentada en las técnicas para esta actividad con las modificaciones específicas que se puedan añadir según el criterio de los gestores del proyecto y sus características.

Actividad 3. Definir los mecanismos de seguimiento y control

A. Descripción

Esta actividad es necesaria para asegurar un correcto seguimiento de los productos de trabajo generados en el transcurso de la fase de Construcción del sistema. A este nivel del desarrollo quizá se tienen gran cantidad de documentación del proyecto y modelos de software que hacen parte de la librería de línea base del proyecto. En la fase de construcción entraran a formar parte de la librería nuevos componentes o nuevas líneas bases correspondientes por ejemplo a modelos de diseño del sistema, interfaces gráficas de usuario, código fuente, etc. Es labor del Gestor de Configuración crear estas nuevas ramas del repositorio con el fin de incluir en ellas las versiones de las líneas base que se vayan generando en el transcurso de la fase.

En la fase de Construcción se utilizan algunos de los productos de trabajo generados en la fase anterior (Formulación del Proyecto), como por ejemplo modelos de software, con el fin de refinarlos o adaptarlos a nuevas condiciones. Por lo tanto, la creación de las nuevas versiones en el

repositorio de líneas base se deben realizar a partir de las versiones “estables” anteriores con ayuda del Gestor de Configuración y con previa autorización del Comité de Control de Cambios.

B. Técnicas

Se pueden encontrar técnicas y referencias en la actividad 4 de la etapa I del Estudio Preliminar.

C. Productos de trabajo

Plan de gestión de configuración, que contiene los puntos descritos para la actividad 4 de la etapa I del Estudio Preliminar.

4.3.2.2 Etapa II - Ampliación y análisis de los requisitos del sistema

Propósito

Completar el refinamiento de los requisitos del sistema, detallando, estructurando y analizando la totalidad de los casos de uso y verificar la consistencia de la arquitectura del sistema.

Actividades

1. Describir los requisitos del sistema
2. Ampliar el Modelo de objetos conceptual para el sistema

Actividad 1. Describir los requisitos del sistema

A. Descripción

En esta actividad se pretende culminar la especificación de requisitos del sistema. En la fase anterior se identificaron y entendieron alrededor del 80% de todos los casos de uso, pero solo una parte se describió en detalle. En esta actividad, el Analista del sistema se encarga de detallar y analizar la totalidad de los casos de uso, con el fin de tener una visión conceptual definitiva del sistema.

B. Técnicas

Para la realización de esta actividad se emplea las técnicas descritas en la etapa II de la Formulación del Proyecto.

Casos de uso

Aunque casi la totalidad de los casos de uso han sido identificados en las fases anteriores, el refinamiento del modelo de casos de uso puede arrojar nuevas funcionalidades que no se habían tenido en cuenta o que no se habían especificado en detalle. En la fase anterior se detallaron los casos de uso esenciales para el sistema, sin embargo, la descripción detallada de los restantes casos de uso da la base para el inicio de la implementación del sistema.

Prototipado de interfaces

Una de las diferencias de aplicaciones sobre plataformas web con respecto a otras aplicaciones que requieren interfaces estándar propias de cada sistema operativo radica en que en gran parte la aceptación de los usuarios radica en la interfaz que la aplicación le ofrezca. De esta manera la interfaz se convierte en el primer punto de referencia para el usuario, por lo que su desarrollo es pieza clave en la construcción final del sistema de comercio electrónico. El Especialista en Interfaces Gráficas debe trabajar de la mano del Analista de sistemas para obtener la mayor ventaja que las interfaces puedan ofrecer a la funcionalidad del sistema. Para su desarrollo se deben tener en cuenta aspectos como la imagen corporativa del cliente (si ya existe), y el tipo de usuarios a los que van dirigidos los contenidos del sistema de comercio electrónico y el tipo de tecnologías que se utilicen para su construcción.

C. Productos de trabajo

Modelo de casos de uso, que contiene

1. Diagrama de casos de uso
2. Descripción de los escenarios de todos los casos de uso
3. Prototipos de interfaces

Actividad 2. Ampliar el modelo de objetos conceptual para el sistema

A. Descripción

En esta actividad, el Analista del sistema, tomando como base los nuevos casos de uso identificados, define el conjunto de clases arquitecturalmente significativas y la forma como ellas participan en la realización de estos casos de uso. Además recoge los nuevos requisitos no funcionales identificados en el transcurso del desarrollo del sistema y los que puedan surgir como requisitos propios del cliente, o que se identifiquen a través de las primeras fases de implementación del sistema.

B. Técnicas

Para la realización de esta actividad se puede continuar empleando las técnicas descritas en la Etapa III de la Fase de Formulación del Proyecto.

C. Productos de trabajo

Modelo de análisis (actualizado), que contiene:

1. Descripción de clases de análisis
2. Diagramas de colaboración para todos los casos de uso
3. Descripción de paquetes de análisis
4. Lista de requisitos no funcionales

4.3.2.3 Etapa III - Construcción del prototipo operacional del sistema

Propósito

Obtener la capacidad operacional del sistema inicialmente proyectada y verificar que satisface los requisitos establecidos. Se debe tener en cuenta que la construcción del primer prototipo es un proceso incremental dirigido por iteraciones. El número de iteraciones que se pueden definir depende de las condiciones propias del proyecto y de los criterios tenidos en cuenta para la planificación temporal. Un aspecto importante mientras se construye el sistema es la verificación de la consistencia de la arquitectura de referencia definida en la Formulación del Proyecto. Esta labor es responsabilidad del Arquitecto del sistema, quien verifica que la construcción se ajusta a la arquitectura, y si se necesita, la modifica para incorporar los cambios que pueden ocurrir en la construcción del sistema.

Actividades

1. Ampliar el modelo de diseño
2. Definir la estructura de navegación
3. Implementar el sistema
4. Planear las pruebas para el sistema
5. Actualizar los compromisos del proyecto
6. Elaborar manuales de usuario y material de apoyo

Actividad 1. Ampliar el modelo de diseño

A. Descripción

En esta actividad se deben complementar los modelos mediante los cuales se describe el sistema a través del diseño de la totalidad de los casos de uso, que para esta fase deben ser alrededor del 90% del total. En esta actividad el Ingeniero de casos de uso debe identificar las clases de diseño necesarias para llevar a cabo un caso de uso. Por otro lado, el Ingeniero de componentes diseña las clases identificadas anteriormente.

B. Técnicas

Para la realización de esta actividad se puede continuar empleando las técnicas descritas en la Etapa III de la Fase de Formulación del Proyecto, sobre todo para el diseño de las clases.

Con el objetivo de identificar las clases de diseño que participan en un caso de uso se puede tener en cuenta lo siguiente:

- Se debe estudiar las clases de análisis que participan en las realizaciones de casos de uso.
- Identificar las clases de diseño creadas por el Ingeniero de componentes o el Arquitecto del sistema que poseen una traza hacia las clases de diseño.
- Se debe estudiar los requisitos no funcionales que se observan en las realizaciones de casos de uso del análisis ya que pueden generar clases de diseño que los realizan.

Patrones de diseño

Se puede pensar en un patrón como “...una solución de diseño de software a un problema, aceptada como correcta, a la que se ha dado un nombre y que puede ser aplicada en otros contextos” [Molpeceres 02].

Los patrones de diseño proponen soluciones a teorías concretas basados en programación orientada a objetos, en ocasiones tienen más utilidad con algunos lenguajes de programación y en otras son aplicables a cualquier lenguaje. Ya que se generan a través de la experiencia acumulada al resolver problemas reiterativos, favorecen la reutilización de código aunque el uso de un patrón no se refleja en el código, al aplicar un patrón, el código resultante no tiene por que delatar el patrón o patrones que lo inspiró [Tejada 02].

De acuerdo al propósito hacia el cual está orientado el patrón se pueden clasificar en: patrones de creación, si tratan la creación de instancias, patrones estructurales, si se ocupan de la combinación de objetos para crear estructuras complejas, y patrones de comportamientos, si tratan la interacción y cooperación entre clases.

Una selección de algunos patrones de diseño aplicados a sistemas de comercio electrónico y su descripción se puede encontrar en el Anexo A de la monografía del MICACE. Sin embargo para una referencia completa sobre patrones de diseño se puede consultar [Gamma 95].

Técnicas para modelar arquitecturas web

Una característica implícita para las aplicaciones sobre plataforma web es la interacción entre el usuario, mediante una herramienta de navegación (browser), y el servidor web quien es el encargado de suministrar los contenidos que los usuarios desean consultar. Esta clara división del comportamiento ha llevado a proponer mecanismos para modelar aplicaciones web en las que se incluyen los elementos más significativos que incluyen el comportamiento de objetos en el lado del cliente y de los objetos que distribuyen la lógica del negocio del lado del servidor.

Las dos abstracciones fundamentales que constituyen esta técnica son las *páginas clientes* y las *páginas servidoras*. Las primeras representan la información con la que el usuario va a interactuar y definen un conjunto de comportamientos que se van a ejecutar del lado del cliente. Las páginas servidoras son objetos que se ejecutan del lado del servidor y que participan en la lógica del negocio del sistema. Generalmente estas sirven de controladores que dirigen la actividad de objetos de negocio necesaria para lograr las metas de negocio iniciadas por las páginas cliente. De esta forma se puede decir que las páginas servidoras reciben solicitudes de las páginas clientes y responden haciendo ejecutar la lógica del negocio y construyendo nuevas páginas clientes necesarias. Una discusión detallada acerca de esta técnica se puede hallar en [Conallen 99].

C. Productos de trabajo

Modelo de diseño, que debe contener:

1. Descripción de subsistemas y clases de diseño
2. Diagramas de colaboración y/o secuencia para los casos de uso
3. Diagrama de clases de diseño

Actividad 2. Definir la estructura de navegación

A. Descripción

Esta aproximación de desarrollo propone configurar la navegación del sistema teniendo en cuenta las abstracciones utilizadas en el dominio de las aplicaciones hipermedia. Una aplicación hipermedia sobre plataforma web puede definirse como la capaz de estructurarse mediante contenidos de múltiples medios (audio, imágenes, secuencias de video, animaciones) además de

poseer contenidos de hipertexto tradicionales. Se puede encontrar una descripción completa sobre conceptos de aplicaciones hipermedia en [Rossi 96]. En esta actividad el Ingeniero de casos de uso identifica los objetos que serán visibles por los usuarios y sus relaciones, la estructura del espacio que va a navegar el usuario, y los mecanismos de acceso que permiten alcanzar los contenidos.

B. Técnicas

Mapeo de clases de diseño

Los *nodos de navegación* son unos de los componentes principales de las aplicaciones hipermedia, ellos agrupan la información que será visible por los usuarios. Los nodos pueden considerarse para el diseño de la navegación como clases. Un nodo de navegación es un tipo de clase navegacional que muestra la agrupación de información desde un determinado criterio, estos son particulares de cada aplicación. Se debe tener en cuenta que un nodo generalmente no está aislado ya que se puede relacionar con otros objetos navegacionales (otros nodos de navegación) mediante relaciones de navegación. Estas relaciones navegacionales se denominan enlaces o "links". Debido a la definición de un nodo como un tipo de clase, este puede estar compuesto por atributos y operaciones. Otro componente importante de los nodos son las *anclas*, estos son un tipo de atributo del nodo que permiten activar la navegación y que tienen asociado un enlace. Las anclas pueden tener una representación en la interfaz de usuario como una imagen, un texto, un botón e incluso una secuencia de audio o video.

Una forma de obtener los nodos y sus atributos es mediante *vistas (orientadas a objetos)* de las clases de diseño del sistema que modelan la información de interés para los nodos, de esta forma un nodo puede estar compuesto por una combinación de atributos de diferentes clases de este tipo. Debido a que los enlaces reflejan relaciones deseadas para ser exploradas por el usuario del sistema también pueden definirse como vistas de las relaciones existentes entre las clases antes mencionadas. Se puede encontrar más acerca de estas técnicas en [Schwabe 98] y en [Rossi 96].

Patrones de diseño navegacional

Existen algunos patrones para navegación sobre sistemas hipermedia que pueden ser útiles para definir algunos aspectos de la navegación sobre el espacio de información ofrecido al usuario.

Contextos Navegacionales: Este patrón concibe al proceso de navegación como una actividad que se realiza bajo un contexto. Este patrón se propone para manejar apropiadamente colecciones de nodos de navegación que pueden ser explorados en diferentes caminos de acuerdo a la tarea que el usuario está desempeñando. Para ilustrar el concepto de contexto se puede considerar el caso de una tienda virtual que ofrezca productos electrónicos. En este caso se puede definir por ejemplo

un Nodo de navegación llamado Computadores, dependiendo de la cantidad de computadores se podría explorar este nodo en los contextos Computadores por marca, Computadores por promoción, y así sucesivamente.

Lo que pretende este patrón es ofrecerle al usuario subespacios de navegación cerrados que contienen relaciones y guías relacionadas con el contexto, así la información puede ser mejor presentada y comprendida. Además del patrón mencionado se pueden encontrar algunos más en [Rossi 97].

Estructuras de acceso

Si bien se han definido nodos y enlaces entre los nodos, estos componentes no son suficientes para permitir un acceso completo a la información ofrecida al usuario. Las estructuras de acceso son aquellos elementos que sirven para acceder a un determinado nodo de navegación y a contextos navegacionales. Básicamente en la terminología de la hipermedia se definen las siguientes estructuras de acceso:

- *Índices*: Permiten acceso directo a cada elemento dentro de un contexto navegacional. Por lo general los índices contienen una lista de descripción de los nodos del contexto, desde donde el usuario selecciona uno. Las descripciones son puntos de partida para la navegación.
- *Vistas guiadas*: Dan acceso al primer objeto del contexto de navegación. Para este caso los objetos son navegados en forma secuencial. Las visitas guiadas pueden ser controladas por el usuario o por el sistema.
- *Consulta*: Se compone de un formulario de entrada que permite filtrar un contexto de navegación. Generalmente son representadas por herramientas de búsqueda.
- *Menú*: Es un índice sobre un contexto navegacional de un conjunto de nodos navegacionales.

Se puede encontrar más información sobre contextos y estructuras de acceso en [Koch 00].

C. Productos de trabajo

Modelo navegacional, compuesto por:

1. Diagrama de clases navegacionales, que contiene nodos, enlaces y anclas.
2. Diagrama de contextos navegacionales, que contiene contextos y estructuras de acceso

Actividad 3. Implementar el sistema

A. Descripción

Esta actividad el Ingeniero de componentes y los desarrolladores (que se comportan como Ingenieros de componentes a más bajo nivel) implementa los *componentes* constitutivos del sistema de comercio electrónico que constituyen la realización de las clases de diseño, representados usualmente por código fuente, ejecutables y binarios, Scripts, archivos, entre otros. Como resultado de esta actividad se tiene una representación del sistema dado por componentes, los que están organizados de acuerdo a los mecanismos de reestructuración y modularización disponibles en el entorno de implementación y en el lenguaje de programación.

Por otro lado el Arquitecto del sistema realiza actualizaciones a la arquitectura si esto se puede presentar.

B. Técnicas

Estos son algunos puntos que se pueden tener en cuenta para implementar una funcionalidad dada del sistema de comercio electrónico:

- La implementación de una funcionalidad debe añadir nuevas características a una construcción anterior, generalmente implementando casos de uso completos.
- Una implementación de una funcionalidad adicional no debería no debería incluir demasiados componentes nuevos o refinados ya que podría dificultar la integración del sistema y las pruebas de integración.
- Una implementación de una funcionalidad nueva debería estar basada en una implementación anterior. Par esto se podría seguir hacia arriba y en forma horizontal la jerarquía de subsistemas definida, para tener una orientación adecuada se hace uso de la arquitectura en capas definida para el sistema.
- Para identificar los componentes ejecutables que pueden ser implantados en nodos, se consideran las clases encontradas durante el diseño y se les asigna un componente ejecutable por clase. Esto también puede incluir identificar otros archivos y/o componentes binarios que son requeridos para crear los componentes ejecutables que hacen parte significativa de la arquitectura.

C. Productos de trabajo

Modelo de implementación, que contiene:

1. Subsistemas de implementación y sus dependencias, interfaces y contenidos.
2. Componentes y sus dependencias
3. Modelo de despliegue (actualizado)

Actividad 4. Planear las pruebas para el sistema

A. Descripción

En esta actividad el Ingeniero de pruebas planifica los esfuerzos de prueba teniendo en cuenta las estrategias seguidas, los recursos necesarios para ejecutarlas, y la distribución temporal para las pruebas. Se debe tener en cuenta que el desarrollo de las pruebas se hace a través de las diferentes iteraciones planteadas para la construcción del primer prototipo del sistema, por lo tanto el Ingeniero de pruebas debe planificar los aspectos antes mencionados por cada iteración. Además de esto, el Ingeniero de pruebas define los contenidos de pruebas según los tipos definidos para estas. Se debe tener en cuenta que ningún sistema puede probarse completamente, por lo tanto el Ingeniero de pruebas da prioridad a aquellas partes críticas del sistema cuya funcionalidad está relacionada con riesgos importantes o que puede influir en la calidad en la prestación de los servicios.

B. Técnicas

El Ingeniero de Pruebas puede basarse en el modelo de casos de uso, el modelo de diseño, y en los requisitos no funcionales para decidir el tipo adecuado de pruebas y los esfuerzos necesarios para llevarlas a cabo. Entre algunos tipos de pruebas que se pueden hacer están las *pruebas de integración* (Definidas para observar como interaccionan los componentes después de un ciclo de construcción, iteración, del sistema), *pruebas de sistema* (Definidas para observar como funciona el sistema como un todo) y *pruebas de regresión*, se puede encontrar una información detallada al respecto en [Jacobson 99].

Para definir algunas estrategias de prueba se puede tener en cuenta los siguientes puntos:

- El porcentaje de pruebas que tienen que automatizarse
- El porcentaje de pruebas que deben hacerse manualmente
- La cantidad de flujos alternativos para los cuales puede probarse un caso de uso
- Definir los criterios de éxito a partir de un número de funcionalidades o casos de uso superados exitosamente

El contenido de las pruebas puede estructurarse mediante *casos de prueba* y *procedimientos de prueba*. Los casos de prueba van a especificar la forma en que se prueba el sistema, teniendo en cuenta la entrada o resultado con que se ha de probar y las condiciones bajo la que ha de probarse. Los procedimientos de prueba especifican como se realizan uno o varios casos de prueba, por ejemplo un procedimiento de prueba puede referirse a un conjunto de instrucciones dadas a un individuo sobre como realizar un caso de prueba. Una discusión más detallada sobre este tema se puede encontrar en [Jacobson 99].

C. Productos de trabajo

Modelo de pruebas, el cual esta compuesto por:

1. Plan de pruebas, el cual contiene
 - Tipos de pruebas a realizar
 - Definición de estrategias
 - Definición de los recursos necesarios
 - Cronograma
2. Documento de especificación de pruebas, el cual contiene:
 - Relación de casos de prueba
 - Procedimientos de prueba utilizados

Actividad 5. Actualizar los compromisos del proyecto

A. Descripción

En esta actividad el Director del proyecto pone al día los compromisos del proyecto que se ven afectados en el transcurso de la construcción del prototipo operacional inicial. Se debe tener en cuenta que el Director del proyecto cada vez que se ha terminado una iteración de la construcción del sistema, verifica si se están cumpliendo los márgenes de productividad que se han marcado para cada incremento del prototipo, si existen desviaciones al respecto se tiene que actualizar el caso del negocio definido en la fase de Formulación del Proyecto. De igual manera se debe actualizar los riesgos identificados si existen condiciones que los afectan mientras se está construyendo el sistema.

B. Productos de trabajo

Lista de riesgos y estrategias (actualizada)

Caso del negocio (actualizado)

Actividad 6. Elaborar manuales de usuario y el material de apoyo

A. Descripción

El objetivo de esta actividad es elaborar la documentación que servirá de referencia a los usuarios finales del sistema. El equipo de desarrollo deberá elegir junto con el Director del proyecto la forma cómo se presente esta documentación de tal forma que sea lo suficientemente clara y de utilidad para el usuario final.

B. Técnicas

Los manuales de usuario y el material complementario de apoyo se elaboran con el objeto de lograr que el usuario final del sistema entienda y comprenda su funcionamiento y que le sirva de referencia en el momento de realizar sus actividades específicas en el sistema.

Para sistemas de comercio electrónico dirigidos a consumidores, el manual de usuario para el navegante no se hace necesario debido a que el sistema debe permitir la navegación intuitiva, sin embargo, se puede tener un enlace a una guía para funciones específicas como comprar, registrarse, etc. o una sección de preguntas frecuentes (FAQs), aclarando dudas que puedan presentarse.

Los siguientes son ejemplos de manuales que se pueden desarrollar para el cliente final:

- *Manual del administrador.* Contiene la explicación de las funciones del sistema administrativo basadas en los casos de uso que involucren al administrador. Puede darse el caso de existir múltiples administradores con distintas funciones para lo cual se hace necesario elaborar un manual para cada uno de acuerdo a sus necesidades específicas.
- *Manual de implantación.* Contiene la información necesaria para implantar el sistema, por ejemplo requisitos hardware y software, librerías específicas, configuración de variables de entorno, y configuración del software necesario para que el sistema funcione correctamente.

Algunos aspectos críticos en sistemas de comercio electrónico tienen relación con la *seguridad*. Si bien algunos puntos sobre seguridad se pueden tratar en el desarrollo del sistema mediante requisitos no funcionales, tales como tipos de conexiones seguras entre los nodos que interactúan para comunicación de datos confidenciales, es de recalcar que los sistemas de comercio electrónico sobre web están disponibles del lado del cliente para cualquier persona que se encuentre conectado a Internet. Este carácter abierto de Internet hace posible que terceras partes *maliciosas* tengan la posibilidad de intervenir en el sistema de comercio electrónico a través de “puertas” que los sistemas operativos, sistemas gestores de bases de datos, y otros sistemas del lado del servidor, dejan abiertas debido a problemas de configuración, errores del funcionamiento del sistema, o deficiencias propias del software. Este tipo de debilidades tecnológicas y de diseño pueden comprometer seriamente la integridad y confidencialidad de los datos que se manejan. Debido a todo lo anterior cuando se implanta un sistema de comercio electrónico es fundamental el apoyarse en el Analista de Seguridad, el cual es una persona idónea en el campo de la *seguridad computacional*, que puede definir los puntos a tener en cuenta para configurar un ambiente aceptablemente seguro para el entorno de ejecución, de acuerdo a las tecnologías vigentes.

- *Carpeta de imagen corporativa.* Constituye un material de apoyo para la empresa dueña del sistema. Este documento lo realiza el Diseñador de interfaces gráficas, que por lo general es una persona idónea en el campo del diseño gráfico. Especifica las características del diseño gráfico de la interfaz de usuario del sistema que definen la imagen corporativa de la empresa, por ejemplo, colores, tipos de fuentes, logotipos, dimensiones de las imágenes, etc.

C. Productos de trabajo

Manuales para el sistema, entre los que deben estar:

1. Manual del administrador del sistema
2. Manual de implantación del sistema
3. Material de referencia y apoyo complementario

4.3.2.4 Etapa IV - Planeación del proyecto

Propósito

Esta etapa pretende definir la agenda de actividades a seguir para la Validación del sistema y los recursos tanto humanos y técnicos necesarios para soportar la validación del sistema por parte de los representantes del cliente.

Actividades

1. Actualizar el Plan de trabajo

Actividad 1. Actualizar el plan de trabajo

A. Descripción

En este punto del desarrollo del proyecto se actualiza la organización de actividades necesarias para llevarse a cabo la validación del sistema por parte del cliente. Es de tener en cuenta que no se puede tener una visión completa de la distribución temporal de algunas actividades de la Validación del sistema ya que algunas implican posibles realimentaciones por parte del cliente sobre el sistema las cuales son difíciles de predecir tales como reportes de fallos o problemas, sugerencias, riesgos generados, cantidad de personal necesario para solucionar los problemas, entre otros. Esta incertidumbre solo se vería reducida si el equipo del proyecto implicado ha acumulado experiencia sobre validación de sistemas por parte de clientes en proyectos pasados. Sin embargo, el Director del proyecto realiza una primera aproximación sobre la relación de actividades para la fase de Validación del sistema, centrándose en aspectos predecibles tales

como la selección del personal para realizar las pruebas, configurar las versiones de prueba del sistema, y general el manual de instrucción de para dicho personal.

B. Técnicas

Se pueden usar las técnicas mencionadas en la actividad 2 de la etapa I del Estudio Preliminar.

C. Productos de trabajo

Plan de trabajo para la Validación del sistema, que contiene los puntos definidos en la actividad 2 de la etapa IV del Estudio Preliminar.

4.4 VALIDACIÓN DEL SISTEMA

4.4.1 Descripción general de la fase

4.4.1.1 Objetivo general

Implantar la versión operativa inicial de sistema de comercio electrónico en su entorno de ejecución.

4.4.1.2 Objetivos específicos

- Obtener la validación del sistema por parte del cliente o su(s) representante(s) y los usuarios.
- Configurar el entorno de ejecución para lograr un adecuado y seguro funcionamiento del sistema de comercio electrónico.
- Corrección de defectos localizados por las pruebas de aceptación
- Incorporar nuevas características permisibles que puedan ser solicitadas
- Mitigar y eliminar los riesgos inesperados que se deriven de las pruebas de aceptación.
- Preparar manuales y documentación que se entregarán con la versión operativa final del sistema de comercio electrónico.

4.4.1.3 Justificación

La fase de validación del sistema es necesaria puesto que permite verificar aspectos críticos del ambiente de ejecución para el sistema de comercio electrónico que pueden incidir en su correcto funcionamiento e integridad, además de permitir la realimentación de errores y características por

parte de los usuarios finales para ajustar la versión operativa final y por lo tanto ser adecuadamente entregada.

4.4.1.4 Etapas de referencia

1. Planeación de la Validación del sistema
2. Migración de la versión operativa inicial al ambiente de ejecución
3. Evaluación la Validación del sistema

4.4.1.5 Productos de trabajo para la Validación del sistema

1. Plan de trabajo para la Validación del sistema
2. Lista de comprobaciones para la Validación del sistema
3. Reporte de pruebas de validación del sistema
4. Versión final del sistema
5. Caso del negocio
6. Documentación legal

4.4.2 Etapas de referencia

4.4.2.1 Etapa I – Planeación de la Validación del sistema

Propósito

Organizar y definir actividades, tiempos, objetivos y recursos para el desarrollo para la Validación del Sistema, además de definir los mecanismos para asegurar el seguimiento de la fase.

Actividades

1. Elaborar el plan de trabajo
2. Establecer los criterios de evaluación
3. Elaborar el plan de trabajo

Actividad 1. Elaborar el plan de trabajo

A. Descripción

En esta actividad el Director del proyecto define el orden de las actividades necesarias para efectuar la Validación del sistema, la organización del personal, determina el costo y el esfuerzo requerido.

B. Técnicas

Si bien se pueden utilizar las técnicas descritas para la actividad 2 de la etapa I del Estudio Preliminar, es de tener en cuenta que generalmente no se puede tener una planeación temporal y de recursos detallada para la fase de Validación del sistema. Algunos aspectos en los cuales se pueden tener una mayor certeza para la planeación pueden ser los siguientes:

- Cantidad de trabajo para la producción de la versión operativa final del sistema de comercio electrónico.
- Preparación de la documentación de las pruebas
- Selección de usuarios de prueba

Sin embargo, existen aspectos acerca de los cuales se tiene puede tenerse muy poca certeza, tales como los relacionados con la cantidad de trabajo que pueden derivar los errores detectados en las pruebas o problemas con el ambiente de ejecución. A pesar de esto, el Director del proyecto teniendo en cuenta el nivel de rigurosidad aplicado al proceso, puede arrancar con una planeación inicial asumiendo que se va a necesitar poca reelaboración como resultado de la validación de los usuarios. Considerando lo anterior se puede tener la asesoría del Ingeniero de Componentes y un grupo de desarrolladores (quienes se comportan como Ingenieros de componentes a más bajo nivel) con el fin de corregir problemas de diseño funcional. También se puede contar con la Asesoría del Diseñador de Interfaces gráficas, el cual puede ayudar a administrar algunas características de interfaz de usuario que propongan a última hora los responsables de las pruebas de aceptación y los usuarios de prueba.

C. Productos de trabajo

Plan de trabajo para la Validación del sistema, documento que contiene:

1. Relación de actividades a realizar, duración estimada y esfuerzo
2. Descripción del equipo humano inicial y asignación de responsabilidades
3. Lista de usuarios de prueba
4. Descripción de los recursos requeridos y las respectivas ventanas de disponibilidad
5. Estimación de los costos

Actividad 2. Establecer los criterios de evaluación

A. Descripción

En esta actividad El Director del proyecto define los aspectos fundamentales que deben obtener evaluación satisfactoria para poder dar por terminado la Validación del sistema.

B. Técnicas

Se pueden considerar como guía algunos de los siguientes criterios generales de evaluación:

- ¿Han cubierto los usuarios de prueba todas las funcionalidades claves en el dominio de la aplicación?
- ¿Se han superado las pruebas de aceptación realizadas por el cliente?
- ¿Se han corregido eficientemente los errores encontrados?
- ¿Se han actualizado los modelos del sistema como resultado de los cambios?
- ¿Se ha configurado adecuadamente el ambiente de ejecución con el objetivo de implantar la versión operativa final del sistema de comercio electrónico sin comprometer la integridad y privacidad de sus datos?
- ¿Se tiene el material de soporte apropiado para la implantación de la versión operativa final del sistema?
- ¿Parecen el cliente y los usuarios satisfechos con la versión de sistema?

C. Productos de trabajo

Lista de comprobaciones para la Validación del sistema, contiene una lista de comprobaciones similar a la presentada para esta actividad con las modificaciones específicas que se puedan añadir según el criterio de los gestores del proyecto y sus características.

4.4.2.2 Etapa II - Migración de la versión operativa inicial al ambiente de ejecución

Propósito

Implantar la versión operativa inicial y gestionar los errores y peticiones del cliente y usuarios con el fin de generar la versión final entregable.

Actividades

1. Implantar la versión operativa inicial y consignar anomalías

2. Corregir errores e incluir nuevas características

Actividad 1. Implantar la versión operativa inicial y consignar anomalías

A. Descripción

En esta actividad el equipo destinado para la Validación del sistema pone en orden la documentación para pruebas preparada en la Construcción del sistema, implanta la primera versión operativa del sistema de comercio electrónico en el ambiente de ejecución y procede a ejecutar las pruebas con los usuarios de prueba de acuerdo a los manuales.

B. Técnicas

Estos son algunos aspectos a tener en cuenta cuando se realiza la implantación y validación del sistema:

- Se debe verificar que la documentación que se tiene para los usuarios de prueba este completa, en caso de que necesite ajustes se debe complementar con el fin de que los usuarios de prueba entiendan específicamente las instrucciones a seguir.
- Si el lanzamiento es de una actualización o reemplazo, previamente se deben proveer instrucciones sobre migración de datos o conversión de bases de datos.
- En el caso de que se vayan a realizar pruebas en diferentes lugares, se debe verificar la consistencia de los Manuales de implantación con el fin de que el sistema de comercio electrónico sea correctamente configurado.
- En el caso de que las pruebas se realicen en un solo lugar, el equipo de Validación del sistema puede apoyarse en el Analista de seguridad con el fin de configurar un ambiente de ejecución aceptablemente seguro (esto puede incidir en las pruebas de aceptación por parte del cliente).
- Se debe tener un documento formal de pruebas de aceptación por parte del cliente.
- Se debe establecer un mecanismo claro para reporte de fallos, problemas encontrados, y nuevas características opcionales por parte de los clientes y los usuarios de prueba.

C. Productos de trabajo

Reporte de pruebas de validación del sistema, que contiene:

1. Lista de errores y problemas encontrados
2. Lista de características opcionales planteadas

Actividad 2. Corregir errores e incluir nuevas características

A. Descripción

Se procede a corregir los errores y anomalías encontradas por parte del los Integrantes del equipo de Validación del sistema que tengan competencia en estos aspectos. Se implementan las características factibles adicionales propuestas por el cliente y los usuarios finales.

B. Técnicas

Estos son algunos aspectos a tener en cuenta cuando se realiza la corrección errores y la implantación de nuevas características en el sistema de comercio electrónico:

- La corrección de errores debe significar una verificación de posibles deficiencias en los modelos construidos a lo largo del desarrollo del sistema de comercio electrónico.
- Se debe observar que los errores que se van a corregir no tienen relación con otros que no se han descubierto hasta el momento.
- Se debe verificar que las correcciones hechas no causen la aparición de nuevos defectos.
- Al identificar una posible forma de corregir el error, se debe garantizar que una vez ejecutada esta alternativa no afecta la arquitectura ni el diseño del sistema.
- Cuando se propongan nuevas características a ser implementadas al sistema de comercio electrónico, se debe verificar que se pueden implementar dentro de los límites del cronograma y que además no afecten significativamente la arquitectura. Si esto no ocurre el Director del proyecto planteará estas nuevas características al cliente, para ser implementadas en una nueva versión del sistema.
- El Director del proyecto debe verificar la agenda del proyecto y realizar los ajustes necesarios si es que la implementación de cambios y características nuevas la modifican.
- La corrección de errores puede generar una actualización de los modelos del sistema de comercio electrónico. Debido a que se ha construido un repositorio de línea base del proyecto, la actualización debe hacerse sobre los componentes definidos en el repositorio. Los cambios deben hacerse con asesoría del Administrador de configuración.

B. Productos de trabajo

Versión final de sistema, que contiene:

1. Versión operativa final del sistema de comercio electrónico
2. Manuales para el sistema actualizados
3. Línea base del proyecto actualizada

4.4.2.3 Etapa III – Evaluación de la Validación del sistema

Propósito

Evaluar los objetivos de la fase y actualizar el caso del negocio.

Actividades

1. Evaluar los productos de trabajo
2. Actualizar el caso del negocio

Actividad 1. Evaluar los productos de trabajo

A. Descripción

El desarrollo de esta actividad permite comprobar a través de los criterios de evaluación establecidos en la primera etapa, si al finalizar la fase se realizaron correctamente todas las actividades propuestas y si los resultados obtenidos con la elaboración de los productos satisfacen los objetivos.

B. Productos de trabajo

Lista de comprobaciones para la Validación del sistema, con indicación de cumplimiento con cada uno de los criterios de evaluación establecidos

Actividad 2. Actualizar el caso del negocio

A. Descripción

En esta actividad se pretende actualizar el caso del negocio con el fin de observar si se tuvo éxito con la ejecución del proyecto. Además de definir los términos legales para la entrega de la versión operativa final.

B. Técnicas

Para hacer la revisión del negocio del proyecto se puede tener en cuenta las siguientes ideas:

- Para el caso de la apuesta económica se debe evaluar si efectivamente el precio contratado ha cubierto los costos del proyecto sobre los márgenes de utilidad económica.
- El caso de la recuperación de la inversión se debe analizar desde el punto de vista del plan de negocios. El éxito se mide respecto a sí se ha generado correctamente las estimaciones sobre los márgenes de beneficios sobre el capital invertido. Esto es, si el sistema de comercio electrónico arroja los beneficios suficientes que justifiquen la inversión realizada. Es claro que para eso se debe esperar el tiempo que el plan de negocios estimó.

C. Productos de trabajo

Caso del negocio (actualizado), que contiene los puntos definidos para la actividad 2 de la etapa V del Estudio Preliminar.

Documentación legal

5. DESCRIPCIÓN DE LA APLICACIÓN

Con el propósito de validar y ejercitar el Modelo Integral para la Construcción de Aplicaciones de Comercio Electrónico, se desarrolló un sistema de comercio electrónico teniendo como cliente a *S&S Communications Corporation*, una empresa con sede en Miami, dedicada a la venta y comercialización de productos y servicios de telecomunicaciones.

El sistema cumple con la función primordial de dar a conocer los productos y servicios ofrecidos por la empresa además de vender en línea tarjetas telefónicas internacionales prepagadas, con la posibilidad de extender a futuro los productos comercializados on-line.

Otro de los productos ofrecidos por la empresa y soportado por el sistema es el PIN, producto con el cual el comprador obtiene una determinada carga de consumo telefónico internacional prepagado a través de un número de identificación, el cual es posible recargar. Para este producto, el sistema utiliza el sistema de gestión telefónica de la empresa D-TAC (www.d-tac.com), que trabaja en alianza con la empresa para la cual se desarrolló el sistema.

La siguiente es una descripción breve de los actores que interactúan con el sistema:

- *Usuario*. Es la persona que utiliza la aplicación para navegar a través de sus secciones, pedir información y realizar compras de productos. Hereda sus características al Visitante y Cliente.
- *Visitante*. Es un usuario que utiliza la aplicación pero sin haberse registrado en el sistema. Este actor no es capaz de gestionar el consumo de un PIN y sus datos no son guardados para agilizar su proceso de compra y personalizar su navegación en el sistema.
- *Cliente*. Es un usuario que tiene almacenados en el sistema sus datos personales y de contacto. Mediante el registro en el sistema este usuario puede personalizar su navegación en el sistema, gestionar su cuenta de PIN y agilizar su proceso de compra.
- *Administrador*. Es la persona que tiene los privilegios para acceder al sistema administrativo del sitio desde donde se pueden configurar los contenidos y funcionalidad del sistema. Hereda sus características al Administrador Comercial y al Administrador Web.
- *Administrador Web*. Es el administrador que tiene los privilegios para administrar los contenidos del sistema, como por ejemplo, secciones, textos, imágenes, etc.
- *Administrador Comercial*. Es el administrador que tiene los privilegios para administrar la funcionalidad del sistema. Este administrador es el encargado de gestionar las peticiones de compra, gestionar ofertas, productos y perfil de los clientes registrados.

- *Sistema de Procesamiento de transacción:* Es el gateway desde donde se realiza la transacción entre el cliente, el banco y el vendedor para la compra de productos y servicios. Este sistema maneja la autorización de tarjetas de crédito internamente, siendo esta parte aislada del sitio de la empresa.
- *Sistema de Procesamiento de Consumo Telefónico:* Es el sistema que gestiona el consumo de llamadas telefónicas hechas a través de un PIN.

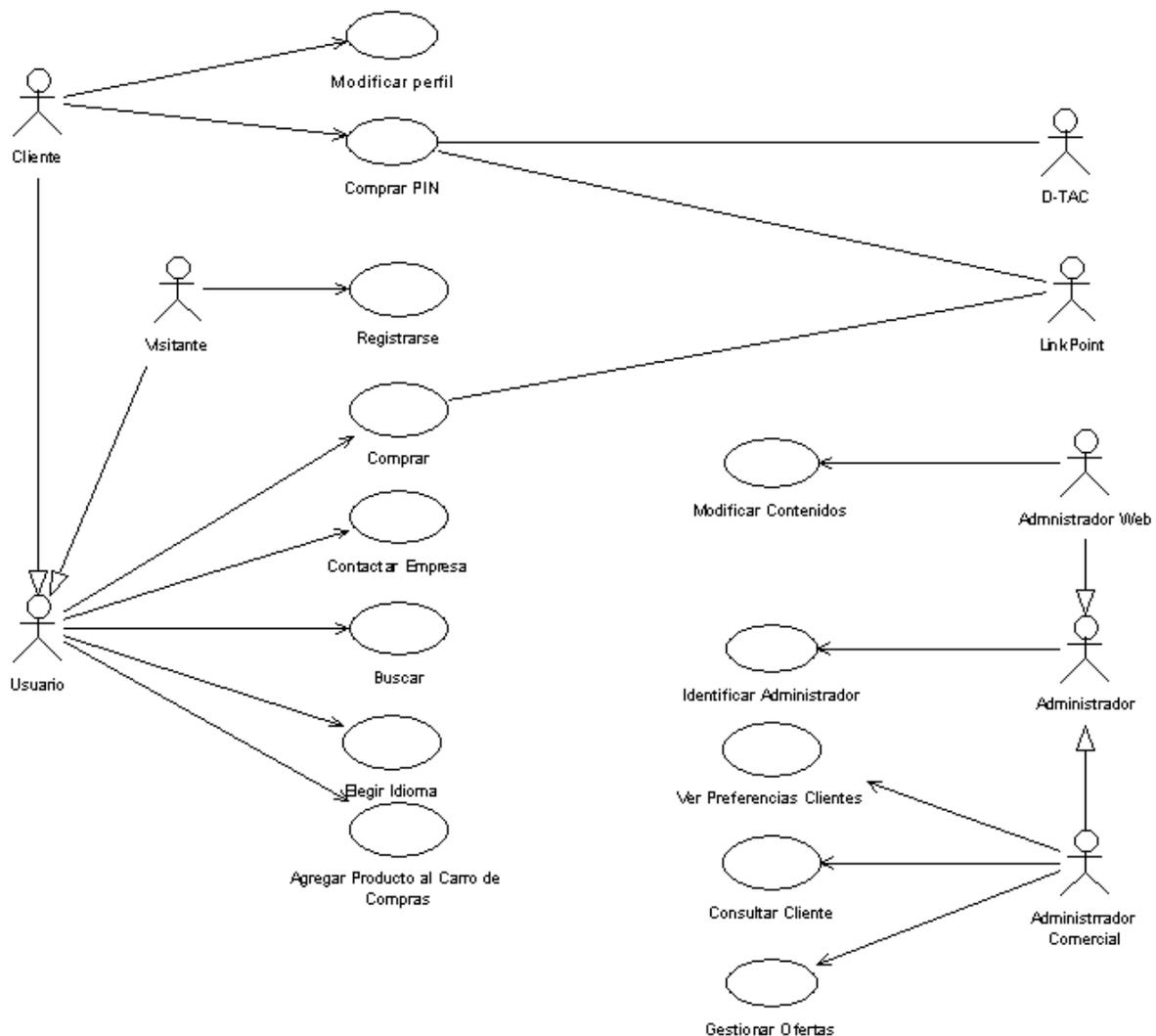


Figura 5.1 Diagrama de Casos de Uso Esenciales del sistema

En la figura 5.1 se presenta el Diagrama de Casos de Uso Esenciales del sistema desarrollado, pero para un mayor detalle de la descripción de sistema construido se puede consultar el Anexo B

- Descripción del sistema de comercio electrónico desarrollado, donde se presentan los productos generados para su desarrollo.

En la figura 5.2 se muestra el diagrama de despliegue del sistema, con los siguientes nodos:

- *Servidor web.* Esta montado sobre un servidor contratado por la empresa con sistema operativo Linux Red Hat 9.0, el servidor web utilizado es Apache Web Server y el lenguaje embebido en HTML es PHP que se impuso como requisito no funcional al inicio del proyecto debido a que está instalado en el servidor donde se instaló el sistema.
- *Servidor de base de datos.* El motor de base de datos utilizado es MySQL, impuesto también como un requisito inicial ya que es el que proporciona el servidor contratado.
- *Cliente.* El sistema está desarrollado para que sea compatible con navegadores superiores a Internet Explorer 4.0 y de otros fabricantes con características similares.
- *Gateway de pagos.* Para el establecimiento de las transacciones financieras, el sistema se soporta en el sistema de transacciones monetarias de *LinkPoint International* (www.linkpoint.com), sistema que a través de una manera segura permite la comunicación con las entidades financieras proveedoras de tarjetas de crédito, entidades bancarias y el sistema desarrollado.
- *Sistema de consumo telefónico.* Este sistema es contratado con la empresa D-TAC se encarga de procesar el consumo de las cuentas de PIN compradas a través del sistema desarrollado.

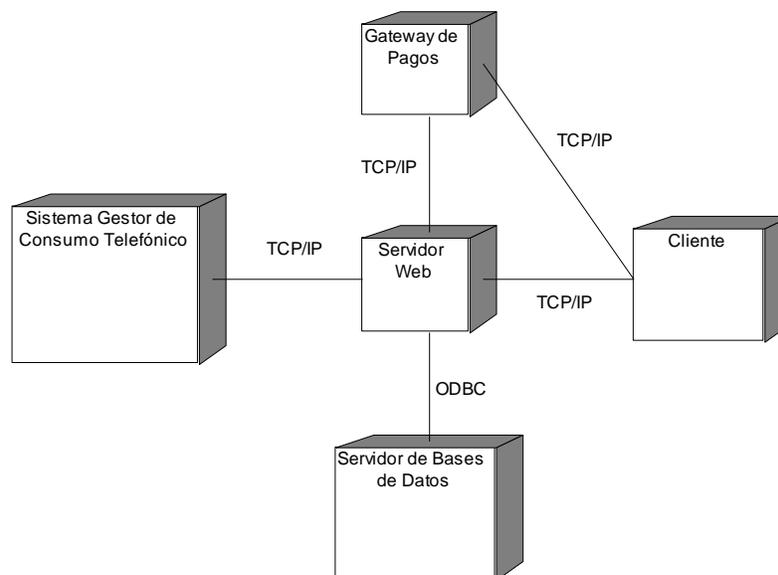


Figura 5.2 Diagrama de despliegue del sistema

En la figura 5.3 se puede ver la interfaz principal del sistema construido. Para su desarrollo se tuvieron en cuenta requisitos impuestos por la empresa, como colores corporativos, logotipos y distribución de la información. La interfaz del sistema está diseñada para una resolución de 800 x 600 pixels, que garantiza una correcta distribución sobre el navegador para casi la totalidad de usuarios de Internet.

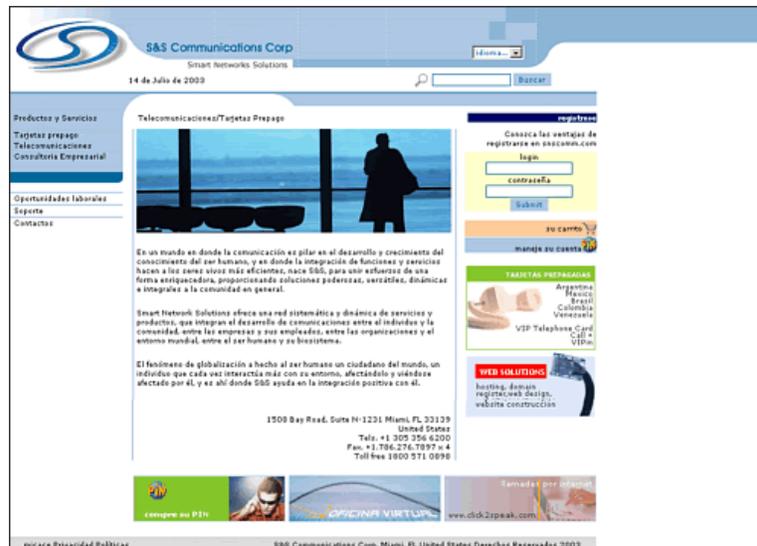


Figura 5.3 Interfaz principal del sistema

6. CONCLUSIONES Y RECOMENDACIONES

- El carácter universal y abierto de Internet y de las tecnologías basadas en web ha llevado a que estas se conviertan en plataformas ideales para la construcción de aplicaciones que pretenden difundir contenidos más allá de las fronteras tradicionales. El dinamismo de los mercados, la globalización y la necesidad de las empresas por conquistar nuevos nichos de mercado y difundir su imagen corporativa, han llevado a ver a estas plataformas como herramientas factibles para configurar un nuevo horizonte en la promoción, venta y distribución de productos-servicios a través de sistemas de comercio electrónico que convivan en este ambiente. Sin embargo para un correcto aprovechamiento de esta situación, los sistemas deben estar contruidos teniendo en cuenta condiciones de calidad bien definidas y basándose en técnicas formales de desarrollo de software que permitan un adecuado manejo de la complejidad inherente de este tipo de sistemas, mientras se aprovecha eficientemente los recursos ofrecidos por las tecnologías disponibles.
- Un aspecto importante cuando se emprende un proyecto de construcción de un sistema de comercio electrónico es el análisis del negocio. Es importante que además de preocuparse por el correcto funcionamiento del sistema, las empresas, sobre todo las PYMES, consideren el costo que implica conseguir la modificación de las conductas de los consumidores, de construir una imagen corporativa desde cero, y administrar la compleja logística de distribución de productos. La experiencia muestra que no es suficiente con tener buenas ideas sobre productos o servicios que se puedan comercializar sobre Internet a través de web, las empresas de comercio electrónico no pueden iniciar su trabajo sin tener un conocimiento claro de sus clientes y de lo que la empresa les puede ofrecer.
- Los sistemas de comercio electrónico pueden llegar a convivir íntimamente con los procesos definidos por la organización, e incluso llegar a formar parte de ella como procesos automatizados. Por lo tanto es vital entender primero qué procesos maneja la empresa u organización destino con el fin de definir los puntos claves en los que un sistema de comercio electrónico puede interactuar apoyando o reemplazando conjuntos de funcionalidades.

- Una característica distintiva en la construcción de sistemas de comercio electrónico son los tiempos de desarrollo y ciclos de vida cortos para el producto, generalmente influenciados por estrategias de marketing y competencia definidas por las empresas. Esto es muy importante ya que compromete características importantes del proceso de desarrollo que permita manejar aspectos tales como administración de la escalabilidad del sistema. Quizá un área importante que colabora en este objetivo es el manejo adecuado de la configuración del sistema (Control de Configuración), identificando los componentes más importantes que constituyen la configuración y los mecanismos de control. Este reto de evolución constante y veloz de los sistemas de comercio electrónico también hace que las organizaciones que los desarrollan deseen poseer estructuras y políticas que les permitan manejar eficientemente equipos multidisciplinarios como los comúnmente encontrados en este dominio y tener un proceso que les permita obtener resultados planeados de una forma predecible (capacidad del proceso).
- La seguridad dentro de los sistemas de comercio electrónico constituye uno de los factores críticos que determinan el éxito o fracaso de estos sistemas y de la organización. Los aspectos de seguridad incluyen desde la adecuada arquitectura física del sistema como la protección contra virus, filtrado de la información, copias de respaldo de la información, etc., hasta las políticas de seguridad dentro de la organización y para con los usuarios finales. Es importante que al interior de la organización se implementen políticas de seguridad informática que garanticen que la información no se desvíe hacia el exterior, permitiendo la violación de los datos del sistema. Las políticas con respecto a los usuarios finales incluye la orientación adecuada en los procesos de compra advirtiendo que la responsabilidad en el manejo de contraseñas y de datos de tarjetas de crédito recae sobre el propietario y es el quien en primera instancia debe garantizar la protección de sus datos.

BIBLIOGRAFIA

[ONU 01] Organización de Naciones Unidas. Conferencia de las Naciones Unidas sobre Comercio y Desarrollo. 2001.

[Jacobson 99] Ivar Jacobson, Grady Booch, James Rumbaugh. El proceso unificado de desarrollo de software. Addison Wesley. 1999.

[Rossi 96] G. Rossi. "An Object-Oriented Method for Designing Hypermedia Applications". PHD Tesis, Departamento de Informática, PUC-Rio, Brazil, Julio 1996.

[Mercerat 02] Bárbara Mercerat, Darío Andrés Silva. "Construyendo aplicaciones web con una metodología de diseño orientada a objetos". Enero de 2002.

[Valderas 02] Pedro J. Valderas Aranda. " Análisis, Diseño e Implementación de un Portal Web para un Departamento Universitario". Universidad Politécnica de Valencia, Valencia, Junio de 2002.

[Martínez 00] José M. Martínez, José R. Hilera, Javier Martínez, José A. Gutiérrez. Orientación a Objetos en la Documentación Hipermedia. Universidad Alcalá de Henares. 2000.

[Serrano 01] Carlos Enrique Serrano Castaño. Modelo para construcción de soluciones v1.0. Popayán, 9 de Mayo de 2001

[Conesa 99] O. Conesa. Seguridad en el Comercio Electrónico: Protocolo SET. ETSETB, Abril 1999.

[Netscape 00] Netscape Communications. <http://wp.netscape.com/security/techbriefs/ssl.html>. 2000.

[Idómine 02] Idómine, Diseño y Desarrollo Web. <http://www.idomine.com/ServidorSeguro.htm>. 2002

[Iglesias 98] Carlos Ángel Iglesias Fernández. "Definición de una Metodología para un Sistema Multiagente ". Universidad Politécnica de Madrid, Madrid, Enero de 1998.

[ISO/IEC 91] ISO/IEC 9126 - International Standard. "Information technology – Software product evaluation – Quality characteristics and guidelines for their use". 1991.

[IEEE 92] IEEE Std 1061, "IEEE Standard for a Software Quality Metrics Methodology", IEEE Computer Society Press. 1992.

[Bevan 99] N. Bevan. "Quality in Use: Meeting User Needs for Quality". Serco Usability Services. 1999.

[ESSISCOPE 02] Proyecto ESSI-SCOPE. http://www.cse.dcu.ie/essiscope/es_home.html. 2002

[Paulk 93] M.Paulk, B. Curtis, M. Chrissis, Ch. Weber. Capability Maturity Model for Software, v 1.1. CMU/SEI. 1993

[CMMIPDT 00] CMMI Product Development Team. CMMI for Systems Engineering/Software Engineering, v 1.02. CMU/SEI. 2000.

[Ledezma 00] A. Ledezma. "Aseguramiento de la calidad del software". Colegio Mayor del Cauca. 2000.

[Stephens 00] E. Stephens. Cybertrust. Seminario Internacional de Comercio Electrónico en Colombia. Cartagena. 2000.

[Certicamara 02] Sociedad Cameral de Certificación Digital, Certicámara S.A. http://www.certicamara.com/certificados/que_es.html. Colombia. 2002.

[Hall 00] E. Hall, Ph. D. – Level 6 Software. "Risk Management Map". Software Tech News. 2000.

[Senn 98] James A. Senn. "Análisis y Diseño de Sistemas de Información". Segunda Edición. McGRAW-HILL, 1998.

[CSE 98] Center for Software Engineering. "COCOMO II Model Definition Manual". USC. 1998.

[RSC 01] Rational Software Corporation. "Business Modeling with the UML and RAS". Rational Software White Paper. Pp. 1-9, 2001.

[ITCG 03] Instituto Tecnológico Ciudad de Guzmán. <http://www.itcg.edu.mx/ingsoft/modelote.htm>. Mexico. 2001.

[ASPL 03] Advanced Software Production Line,S.L.http://fact.aspl.es/doc/doc_html/gcs-node1.html. 2003.

[Hauswirth 00] M. Hauswirth, M. Jazayeri, M. Schneider. "A phase model for e-commerce business models and its application to security assessment". Technical University of Viena.2000.

[DMI 99] Departamento de Ciencias Matemáticas e Informática. Universidad de les Illes Balears. <http://dmi.uib.es/~bbuades/riesgos/>. 2003.

[Dimu 01] N. Dimu, J. Kereki, A. Vignaga. "Detección de Relaciones entre Casos de Uso". CLEI 2001. Mérida, Venezuela. 2001.

[Melo 01] D. Melo, G. Ramírez. Metodología Integral para la Construcción de Servicios. Grupo de Ingeniería Telemática. Universidad del Cauca. Colombia. 2001.

[Booch 99] G. Booch, J. Rumbaugh, I. Jacobson. El Lenguaje unificado de modelado. Addison Wesley. 1999.

[Gulías 02] J. Gulías, V. Gulías, A. Valderruten. "Un sistema de comercio electrónico sobre un cluster de computadores". LFCIA. Universidad de A Coruña. 2002.

[Hall 98] E. Hall. Managing Risk : Methods for Software Systems Development. Adison Wesley. 1998.

[Schwabe 98] D. Schwabe, G. Rossi. "An Object Oriented Approach to Web-Based Application Desing ". LIFIA, UNLP. 1998.

[Rossi 97] G. Rossi, D. Schwabe. A. Garrido. "Design Reuse in Hypermedia Applications Development". UNLP, PUC-RIO. 1997.

[Koch 00] N. Koch, L. Mandel. "Using UML to Design Hypermedia Applications". Universität München. 2000.

[Molpeceres 02] A. Molpeceres. "Diseño de software con patrones". <http://www.javahispano.com>. 2002.

[Tejada 02] D. Tejada. "Guía de patrones de diseño". <http://www.teleprogramadores.com>. 2002.

[Gamma 95] E. Gamma, R. Helm, R. Johnson, J. Vlissides. Design Patterns : Elements of Reusable Object-Oriented Software. Addison-Wesley Professional Computing Series. 1995.

[Conallen 99] J. Conallen. "Modeling Web Applications Architectures with UML". Communications of the ACM. Vol 42, No 10. Octubre de 1999.