



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**DISEÑO DE LA INVESTIGACIÓN DE LA ARQUITECTURA WEB DEL
SISTEMA DE COTIZACIONES PERSONALIZADAS EN LÍNEA MAKE IT
YOURSELF DE PRODUCTOS ALIMENTICIOS**

Carlos Fernando Sermeño Vásquez

Asesorado por el Msc. Ing. Pablo Augusto Rosales Tejada

Guatemala, febrero de 2013

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO DE LA INVESTIGACIÓN DE LA ARQUITECTURA WEB DEL
SISTEMA DE COTIZACIONES PERSONALIZADAS EN LÍNEA MAKE IT
YOURSELF DE PRODUCTOS ALIMENTICIOS**

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

CARLOS FERNANDO SERMEÑO VÁSQUEZ

ASESORADO POR EL MSC. ING. PABLO AUGUSTO ROSALES TEJADA

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, FEBRERO DE 2013

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympto Paiz Recinos
VOCAL I	Ing. Alfredo Enrique Beber Aceituno
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Walter Rafael Véliz Muñoz
VOCAL V	Br. Sergio Alejandro Donis Soto
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympto Paiz Recinos
EXAMINADOR	Ing. Marlon Antonio Pérez Türk
EXAMINADORA	Inga. Floriza Felipa Ávila Pesquera de Medinilla
EXAMINADORA	Inga. Sonia Yolanda Castañeda Ramírez
SECRETARIO	Ing. Hugo Humberto Rivera Pérez.

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

DISEÑO DE LA INVESTIGACIÓN DE LA ARQUITECTURA WEB DEL SISTEMA DE COTIZACIONES PERSONALIZADAS EN LÍNEA MAKE IT YOURSELF DE PRODUCTOS ALIMENTICIOS

Tema que me fuera asignado por la Dirección de la Escuela de Estudios de Posgrado, con fecha 28 de enero de 2013.



Carlos Fernando Sermeño Vásquez

Universidad de San Carlos
de Guatemala



Escuela de Estudios de Postgrado
Facultad de Ingeniería
Teléfono 2418-9142

AATT-MTIPP-0015-2013

Guatemala, 12 de febrero de 2013

Director:
Marlon Antonio Pérez Turk
Escuela de Ingeniería en Ciencias y Sistemas
Presente.

Estimado Director:

Reciba un atento y cordial saludo de la Escuela de Estudios de Postgrado. El propósito de la presente es para informarle que se ha revisado los cursos aprobados del primer año y el Diseño de Investigación del estudiante **Carlos Fernando Sermeño Vasquez** con carné número **2002-12218**, quien opto la modalidad del **"PROCESO DE GRADUACIÓN DE LOS ESTUDIANTES DE LA FACULTAD DE INGENIERÍA OPCIÓN ESTUDIOS DE POSTGRADO"**.

Y si habiendo cumplido y aprobado con los requisitos establecidos en el normativo de este Proceso de Graduación en el Punto 6.2, aprobado por la Junta Directiva de la Facultad de Ingeniería en el Punto Decimo, Inciso 10.2, del Acta 28-2011 de fecha 19 de septiembre de 2011, firmo y sello la presente para el trámite correspondiente de graduación de Pregrado.

Sin otro particular, atentamente,

PABLO ROSALES TEJADA
INGENIERO EN SISTEMAS
COLEGIADO 8824

Msc. Ing. Pablo Augusto Rosales Tejado
Asesor (a)

"Id y enseñad a todos"

Msc. Ing. Marlon Antonio Pérez Turk
Coordinador de Área
Aplicación y transferencia tecnológica

César Akú Castillo MSc.
INGENIERO INDUSTRIAL
COLEGIADO No. 4,073

Dra. Mayra Virginia Castillo Montes
Directora
Escuela de Estudios de Postgrado

Cc: archivo
/la

E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“DISEÑO DE LA INVESTIGACIÓN DE LA ARQUITECTURA WEB DEL SISTEMA DE COTIZACIONES PERSONALIZADAS EN LÍNEA MAKE IT YOURSELF DE PRODUCTOS ALIMENTICIOS”**, realizado por el estudiante CARLOS FERNANDO SERMEÑO VÁSQUEZ, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

Ing. Marlon Antonio Pérez Turk
Director, Escuela de Ingeniería en Ciencias y Sistemas



Guatemala, 28 de febrero 2013

Universidad de San Carlos
de Guatemala



Facultad de Ingeniería
Decanato

DTG. 151.2013

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al Trabajo de Graduación titulado: **DISEÑO DE LA INVESTIGACIÓN DE LA ARQUITECTURA WEB DEL SISTEMA DE COTIZACIONES PERSONALIZADAS EN LÍNEA MAKE IT YOURSELF DE PRODUCTOS ALIMENTICIOS**, presentado por el estudiante universitario: **Carlos Fernando Sermeño Vásquez**, autoriza la impresión del mismo.

IMPRÍMASE:



Ing. Murphy Olympo Paiz Recinos
Decano

Guatemala, 28 de febrero de 2013

/gdech



ACTO QUE DEDICO A:

Dios	Por darme la vida y permitirme superarme.
Mis padres	Por haberme dado la vida, por el apoyo incondicional que siempre me brindaron tanto en prósperos días, como en días adversos y por proveerme de los recursos necesarios para desarrollar mis habilidades educativas.
Mis hermanas	Por ser mis primeras compañeras de vida y estar conmigo todo el tiempo.
Mis abuelos (q.e.p.d.)	Por cuidarme y criarme desde pequeño, por brindarme su amor incondicional y mis primeros pasos para mi educación.
Mi tía	Por estar conmigo en mis primeros años de vida y guiarme por un buen camino.
Mis compañeros de estudio	Por ser de gran apoyo tanto en el transcurso de la vida universitaria como laboral.
Mis demás compañeros	Por creer en mí, darme su apoyo y en muchas ocasiones su ayuda.

AGRADECIMIENTOS A:

**La Universidad de San
Carlos de Guatemala**

Alma máter que me abrió las puertas para realizar mis estudios.

Facultad de Ingeniería

Por darme la oportunidad de superarme mediante los conocimientos que adquirí, habilidades y experiencias que hoy en día me sirven para desenvolverme profesionalmente.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
GLOSARIO	VII
RESUMEN.....	IX
INTRODUCCIÓN	XI
ANTECEDENTES.....	XV
OBJETIVOS.....	XVII
ALCANCE DEL TEMA	XIX
NECESIDADES A CUBRIR	XXI
1. CONCEPTO DE SISTEMAS INFORMÁTICOS Y TECNOLOGÍA.....	1
1.1. Hardware	1
1.2. Software	1
1.3. Sistema Informático	1
1.4. Ciclo de Vida del Software.....	2
1.5. Componentes del Sistema Make It Yourself.....	3
1.5.1. Modulo web	3
1.5.2. Módulo de ventas y pedidos	3
1.5.3. Módulo de inventarios.....	3
2. ARQUITECTURA WEB.....	5
2.1. Componentes Semánticos de la web	5
2.2. Arquitectura de las aplicaciones web.....	5
2.2.1. Componentes del software de la web.....	6
2.2.2. Arquitectura cliente/servidor	6
2.2.3. Aplicaciones web	7

2.2.4.	Componentes de una aplicación web.....	7
2.2.5.	Modelo de capas.....	8
2.2.6.	Patrón de diseño de arquitectura	9
2.2.7.	Requerimientos No Funcionales	9
2.2.7.1.	Escalabilidad	10
2.2.7.2.	Portabilidad.....	11
2.2.7.3.	Seguridad	11
2.2.7.4.	Integración.....	11
2.2.7.5.	Disponibilidad	11
2.2.7.6.	Confiabilidad.....	12
3.	PROCESO DE ELABORACIÓN DEL SOFTWARE	13
3.1.	Análisis y diseño.....	13
3.2.	Casos de uso	13
3.3.	Componentes de la interfaz gráfica.....	13
3.4.	Arquitectura de la Solución.....	14
3.5.	Diagrama de clases.....	14
3.6.	Diagrama de secuencia.....	14
3.7.	Diagrama de estados	14
3.8.	Diagrama entidad-relación	15
4.	DOCUMENTACIÓN	17
4.1.	Manual de Usuario	17
4.2.	Manual Técnico	17
4.3.	Diccionario de Datos	17
4.4.	Diagrama de Procesos.....	17
4.5.	Scripts del Modelado de la Base de Datos.....	18

5.	METODOLOGÍA A UTILIZAR Y TÉCNICA CORRESPONDIENTE	19
5.1.	Fase 1 - Configuración de la Red	19
5.2.	Fase 2 - Modelo de Capas	20
5.3.	Fase 3 - Patrón de Arquitectura.....	20
5.4.	Fase 4 - Requerimientos No Funcionales.....	21
5.5.	Fase 5 - Lenguaje de programación.....	22
5.6.	Fase 6 - Análisis, Diseño e Implementación de la Aplicación..	22
6.	RECURSO FÍSICOS Y FINANCIEROS	25
7.	CRONOGRAMA.....	27
8.	BIBLIOGRAFÍA	29

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Elaboración de hamburguesa.....	XVI
2.	Recursos físicos del proyecto.....	25
3.	Proyección de los activos y pasivos.....	26
4.	Diagrama de actividades de Gantt.....	28

GLOSARIO

Aplicación Web	Conjunto de recursos que participan en el funcionamiento de la propia aplicación web.
Arquitectura Web	Define una tarea que requiere conocimientos técnicos de construcción, de funcionalidades y de diseño para sitios o páginas web.
Hardware	Es la parte tangible de una computadora.
Internet	Es una red de redes que permite la interconexión descentralizada de computadoras a través de un conjunto de protocolos denominado TCP/IP.
Sistema	Conjunto de partes que funcionan relacionándose entre sí para conseguir un objetivo preciso.

RESUMEN

El sistema Make It Yourself permitirá al usuario ingresar una cantidad determinada de ingredientes alimenticios por unidad, los cuales serán utilizados por el usuario mediante una aplicación web que permitirá, calcular la cantidad exacta de ingredientes a utilizar en la elaboración de un determinado producto alimenticio, que luego conjuntamente con otros productos previamente elaborados permitirán al usuario crear un menú o un combo, con su respectivo cálculo de precios y de ingredientes utilizados.

La aplicación en su primera versión permitirá elaborar productos alimenticios sencillos y solo se encontrará disponible en idioma español. Es una plataforma de pedidos alimenticios en línea en donde se calculará la cotización inmediatamente del pedido a solicitar según los ingredientes a utilizar para la elaboración de los productos. A la vez llevará un inventario de productos que indicará la cantidad de ingredientes en existencia.

El sistema será accedido mediante algún explorador de internet como lo es Opera, Mozilla Firefox, Google Chrome, Safari o Internet Explorer y estará disponible en la web sin importar la ubicación geográfica.

INTRODUCCIÓN

Elaborar productos alimenticios ha sido una tarea que diariamente se realiza en todas partes del mundo, con la finalidad de satisfacer las necesidades del ser humano ya que los alimentos le dan vitaminas al organismo y permiten vivir diariamente. En muchas ocasiones elaborar recetas utilizando una cantidad determinada de ingredientes es una tarea fácil, pero en ocasiones no se sabe la cantidad exacta de ingredientes a utilizar y tampoco es fácil determinar el costo, no se sabe si la cantidad de ingredientes en existencia serán suficientes, por lo tanto es importante buscar una alternativa que permita elaborar productos alimenticios con su respectivo cálculo exacto de ingredientes a utilizar y el costo que representa la elaboración del producto.

El presente estudio especial de graduación pretende mostrar el diseño de la arquitectura web a ser utilizada en la aplicación, la cual permitirá elaborar productos alimenticios, en donde el usuario ingresará un inventario de los ingredientes almacenados y posteriormente los utilizará para calcular una determinada cantidad de productos, con su respectivo precio por unidad. Así mismo, la aplicación permitirá a los usuarios elaborar un combo o algún menú y calcular el costo exacto de producción para determinar el valor de venta. Cada producto elaborado se asociará a un sistema de ventas y de inventarios en donde el usuario final o administrador tendrá la capacidad de observar una serie de reportes que le indicarán los movimientos de costos y ventas de su empresa.

El proyecto se enfoca directamente en la arquitectura física y web de la aplicación que se utilizará para la elaboración de los productos alimenticios, en donde se definirán las tecnologías a utilizarse internamente en la aplicación y los estándares a definirse para la funcionalidad del sistema sea la correcta.

Se pretende de esta forma que el documento brinde una visión global y comprensible del diseño arquitectónico interno general de la aplicación a ser desarrollada, así mismo se pretende capturar y transmitir las decisiones arquitectónicas más importantes realizadas en el sistema.

El primer capítulo consta de la descripción de la aplicación, en donde se plantea el problema, las necesidades a cubrirse, las limitaciones y objetivos por los cuales es importante desarrollar la aplicación. El segundo capítulo comprende los componentes por los cuales el proyecto será dividido, ya que se trabajarán por medio de módulos, los cuales cada uno tendrá una función principal dentro del sistema. El tercer capítulo denominado Arquitectura del Software comprende estudios técnicos que se llevaron a cabo para la elaboración del sistema. En el cuarto capítulo denominado Elaboración del Software se detallaran los casos de uso, así como el proceso de desarrollo, implementación y elaboración de pruebas del sistema. Finalmente, el quinto capítulo comprenderá la documentación final del sistema.

El producto está clasificado como un emprendimiento. Es emprendimiento debido a que existen aplicaciones sencillas que permiten elaborar productos mediante un catálogo de ingredientes pero estos solo se encuentran en juegos y en páginas web de empresas privadas que únicamente permiten al usuario personalizar los productos ya existentes, sin llevar un inventario de disponibilidad, y modificar los combos que están a la venta.

El proyecto es un emprendimiento porque se ha identificado una oportunidad de negocio en donde se organizarán los recursos necesarios para, poner en marcha una aplicación que permitirá al público en general elaborar cualquier producto y estar disponible en la web ya que estará orientado a los activos y pasivos del negocio, determinando precios y cantidades de materiales a utilizarse. El emprendimiento busca involucrar la tecnología en un canal de un negocio ya existente.

El proyecto responde a las siguientes interrogantes:

¿Qué arquitectura web se utilizará para alojar el sistema Make It Yourself que permita brindarle un funcionamiento óptimo?

¿Qué configuración de red es la necesaria para conectar todos los componentes físicos del sistema?

¿Cuál será el modelo de capas ideal que permitirá separar los diferentes ambientes del desarrollo de la aplicación?

¿Qué patrón de diseño de arquitectura brindara una solución óptima para el rendimiento de la aplicación?

¿Cómo se manejarán los requerimientos no funcionales que satisfacerán las necesidades del sistema, tales como la seguridad, confiabilidad, escalabilidad, portabilidad y disponibilidad?

¿Cuál es el lenguaje de programación que se utilizará para desarrollar la aplicación? y ¿qué servicios web se necesitan para poner en marcha el sistema?

ANTECEDENTES

Al navegar sobre la web se puede encontrar una gran variedad de aplicaciones que permiten tanto a personas individuales, como a empresas, crecer y mejorar en sus negocios. Todas las industrias trabajan hoy en día con plataformas informáticas que permiten facilitar los procesos de los negocios y las franquicias alimentarias no son la excepción.

En internet se encuentran varias páginas privadas en las cuales un usuario o posible consumidor accede y elabora su propio menú para posteriormente ordenarlo vía web, tal es el caso de la página <http://www.mcdonalds.com.gt/>, en donde se pueden personalizar los menús y posteriormente ser cotizados.

A la vez existe una gran cantidad de aplicaciones en las cuales el usuario accede y puede elaborar diferentes productos alimenticios pero solamente se tratan de juegos como los que se pueden encontrar en el siguiente listado de páginas web:

- <http://www.juegoscocinapasteleria.com/category/juegos-de-hacer-comida/>
- <http://es.nourishinteractive.com/nutrition-tools-healthy-family/parents-interactive-meal-planner>
- <http://www.juegosdehacercomida.com/>

Figura 1. **Elaboración de hamburguesa**



Ingredientes para la elaboración de una hamburguesa.

Fuente: www.mcdonalds.com.gt/. Consulta: 25 de mayo de 2012.

Como se pudo observar, en internet hay una gran cantidad de aplicaciones que permiten a usuarios crear o modificar combos de comidas, pero lamentablemente no existe una aplicación que permita llevar una cotización, un inventario, un manual que indique el proceso de elaboración con cantidades exactas de ingredientes a utilizar para crear un producto, que luego conjuntamente con otros productos crearan un combo. Si se realizará una combinación de todas las aplicaciones existentes, ya sea de juegos o de páginas privadas, posiblemente se elaboraría una aplicación que funcionará con todos los componentes mencionados y aun así, no satisfecerá del todo a los usuarios, por lo que desarrollar una aplicación de este tipo es importante para satisfacer tanto las necesidades de los usuarios como de los clientes al buscar una cotización en la vida real.

OBJETIVOS

General

Establecer e implementar una arquitectura web que permita alojar el sistema Make It Yourself, brindándole un óptimo funcionamiento.

Específicos

1. Determinar una configuración de red que permita conectar todos los componentes físicos del sistema.
2. Establecer un modelo de capas ideal para separar los diferentes ambientes del desarrollo de la aplicación.
3. Definir un patrón de diseño de arquitectura que brinde una solución óptima para el rendimiento de la aplicación.
4. Desarrollar un sistema web que cumpla con los requerimientos no funcionales, que sea seguro, confiable, escalable, portable y este siempre disponible.
5. Establecer el lenguaje de programación que se utilizará para el desarrollo de la aplicación y determinar los servicios web necesarios para poner en marcha el sistema.

ALCANCE DEL TEMA

La aplicación permitirá al usuario elaborar un inventario con el ingreso de información al sistema, determinando la cantidad de ingredientes a almacenarse que previamente estarán definidos con su respectiva imagen y con sus detalles.

El sistema permitirá ingresar una cantidad limitada de ingredientes y conforme vaya creciendo y expandiéndose se agregarán más ingredientes. Posteriormente en base a la cantidad de ingredientes en existencia permitirá a los usuarios crear productos y almacenar una cantidad determinada de productos, para luego ser utilizados en los combos. Cada combo será la unión de una cantidad determinada de productos.

Únicamente se permitirá la elaboración de productos que no sean bebidas, ya que estas entrarán en una segunda fase a desarrollarse una vez finalizado el proyecto actual que se encuentra enfocado en los productos alimenticios sólidos.

Se lleva un control de los usuarios que ingresan al sistema y los accesos mediante los roles administrativos y de usuarios finales.

El inventario proporcionará reportería de todos los movimientos generados según las fechas, ingredientes, productos y cantidades en existencias.

En cuanto a la arquitectura se definirán los patrones a utilizarse en cada una de las capas del modelo seleccionado, se establecerán métricas y se definirá el nivel de seguridad en la web. Asimismo, las tecnologías a utilizarse en el desarrollo de la aplicación, como lo son Php, Ajax, MySql entre otras, y se definirán los detalles de cada uno de los servidores que se utilizarán.

El estudio de graduación se enfoca directamente en todo lo que es la arquitectura del sistema por completo, tomando en cuenta cada uno de los tres módulos anteriormente descritos en la Arquitectura General de la Solución.

NECESIDADES A CUBRIR

Como producto está la aplicación web en donde el usuario ingresará a un inventario de los ingredientes almacenados en su bodega o dispensador y posteriormente, los utilizará para calcular una determinada cantidad de productos con su respectivo precio por unidad. Asimismo, la aplicación permitirá a los usuarios elaborar combos calculados con precios exactos de venta. Estos productos estarían asociados a un sistema de inventario y ventas. Permitiendo hacer la compra más ágil, directa y clara con respecto a lo que se ofrece.

Se prestarán los servicios de configuración y mantenimiento de la aplicación, la cual primero se estaría configurando de acuerdo al negocio de cada uno de los clientes, es decir, habilitar los productos y restricciones según las necesidades requeridas.

Asimismo, se dará el servicio de tutorías como el de ayuda de escritorio en un determinado horario, una vez la aplicación este ya en la web y se tengan usuarios, esto para controlar los módulos de inventario y ventas.

Los clientes que utilizarán la aplicación web son vendedores individuales o personas particulares que necesiten elaborar productos como lo son pasteles y necesiten calcular la cantidad de porciones a consumir, cantidad de ingredientes y el precio, así mismo el mercado está destinado a pequeñas y medianas empresas que se dediquen a la venta de productos alimenticios. En sí, el mercado objetivo son las pequeñas empresas.

Actualmente, es una aplicación en emprendimiento y la misma en el mercado no posee competencia alguna directamente, existen aplicaciones como la de McDonald's en la cual se logra personalizar el combo, u otras que se basan en juegos, pero ninguna a nivel corporativo que sea utilizada en la industria alimenticia en general.

La aplicación estará desarrollada en su totalidad por tecnologías Open Source (sin costo o gratuitas) como lo es Php y MySQL, dependiendo de qué tan rápido vaya creciendo la aplicación podrá ser necesario adquirir la licencia de MySQL para obtener soporte de la base de datos.

1. CONCEPTO DE SISTEMAS INFORMÁTICOS Y TECNOLOGÍA

Es de suma importancia definir los conceptos de sistemas informáticos y de tecnología, con el fin de especificar los términos que se utilizarán durante el desarrollo de este documento.

1.1. Hardware

Se denomina hardware (Carrillo, 2006) es la parte física de un sistema, es decir, el conjunto de dispositivos, cables, transistores, etc. que lo conforman.

1.2. Software

Se denomina software (Carrillo, 2006) a la parte lógica, es decir, al conjunto de programas, y por tanto intangible del sistema.

1.3. Sistema Informático

Un sistema informático (Gallego, 2010), es un conjunto de partes que funcionan relacionándose entre sí para conseguir un objetivo preciso. Las partes fundamentales de un sistema informático se detallan a continuación:

- Hardware: está formado por los dispositivos electrónicos y mecánicos que realizan cálculos y el manejo de la información.
- Software: se trata de las aplicaciones y los datos que explotan los recursos del hardware.

- Personal: está compuesto tanto por los usuarios que interactúan con los equipos como por aquellos que desarrollan el software para que esa interacción sea posible.

El concepto de sistema informático más simple sería el formado por un equipo con su usuario y el manual de instrucciones. No obstante, un sistema informático puede indefinidamente e incluso abarcar o interactuar con otros sistemas informáticos.

Para el desarrollo de un sistema informático es necesario llevar a cabo el ciclo de vida del software.

1.4. Ciclo de Vida del Software

El desarrollo de un sistema de información contempla una serie de etapas bien diferenciadas tanto en características como en el volumen de recursos a utilizar. El conjunto ordenado de estas etapas es conocido generalmente como ciclo de vida del sistema (Areba, 2001). La forma en que son denominadas las distintas etapas presentan grandes variaciones según los autores, pero en su conjunto estas etapas son las mismas, con un grado mayor o menor de detalle. Así, globalizando todas las metodologías de desarrollo de sistemas podemos decir que el ciclo de vida de un sistema se compone fundamentalmente de las etapas de especificación (donde se estudia y define el nuevo sistema a crear), diseño (donde se detallan y modelan los componentes del nuevo sistema) e implantación (donde se prueba y se pone en marcha el nuevo sistema).

Una vez definido el ciclo de vida del software es necesario centrar la atención en la arquitectura de la aplicación web.

1.5. Componentes del Sistema Make It Yourself

El sistema se encontrara dividido por tres componentes o módulos importantes que conjuntamente realizaran las funciones principales del sistema.

1.5.1. Modulo web

Modulo en el cual se alojara la aplicación del sistema y es en donde el usuario accederá al sistema según los privilegios que le sean otorgados.

1.5.2. Módulo de ventas y pedidos

Módulo en el cual se realizarán las operaciones de cotización de precios por productos, elaboración de productos según una determinada cantidad de ingredientes y la creación de menús.

1.5.3. Módulo de inventarios

Módulo encargado de gestionar la cantidad de productos disponibles en el sistema y por lo tanto en vida real, permitirá administrar de mejor forma los recursos disponibles dentro del mismo, con el fin de calcular mejor las cotizaciones y evitar la falta de productos al realizar pedidos.

2. ARQUITECTURA WEB

La arquitectura web se basa en todos los componentes externos, no funcionales, que utiliza una página web para su funcionamiento correcto sobre alguna plataforma.

2.1. Componentes Semánticos de la web

La web semántica está apoyada en la incorporación de los metadatos semánticos que circula por internet, lo que hace a los datos más accesibles.

Los principales componentes de la web semántica (Busón, 2007) son los metalenguajes y estándares de representación XML, XML Schema, RDF, RDF Schema y OWL. La OWL (Web Ontology Language Overview) describe la función y relación de cada uno de los componentes de la web semántica, cuyo desarrollo está marcado un antes y un después en la forma de trabajar en internet.

2.2. Arquitectura de las aplicaciones web

Las aplicaciones web se basan en una arquitectura cliente/servidor: por un lado está el cliente (el navegador o visualizador) y por el otro lado el servidor (el servidor web). Existen diversas variantes de la arquitectura básica según como se implementen las diferentes funcionalidades de la parte del servidor. Las arquitecturas más comunes son:

- Todo en un servidor

- Servidor de datos separado
- Todo en un servidor con servicio de aplicaciones
- Servidor de datos separado
- Todo separado, (Mora, 2002).

Una vez definida la arquitectura web que se utilizará para montar la aplicación, se es necesario definir cual se utilizará.

2.2.1. Componentes del software de la web

Una aplicación web contiene componentes diferentes. Cuando se crea una aplicación web, debe comprender cómo implementar y utilizar todos los componentes.

2.2.2. Arquitectura cliente/servidor

Actualmente, el entorno distribuido que se utiliza más es el denominado cliente/servidor.

La idea básica de la arquitectura cliente/servidor (Falgueras, 2003) es que un programa, el servidor, gestiona un recurso compartido concreto y hace determinadas funciones sólo cuando las pide otro, el cliente, que es quien interactúa con el usuario. Normalmente, estos dos programas, el servidor y el cliente, están en ordenadores distintos.

Los requerimientos de los ordenadores clientes en lo que respecta a velocidad, memoria y capacidad de disco son muy diferentes de los de los servidores; unos y otros pueden ser ordenadores de modelo y marca diferentes y, además, con frecuencia utilizan un sistema operativo diferente.

Las aplicaciones web estarán montadas sobre la arquitectura web ya definida.

2.2.3. Aplicaciones web

Una aplicación web es un conjunto de recursos que participan en el funcionamiento de la propia aplicación web.

Una aplicación web está compuesta de 1) componentes de servidor dinámicos como Servlets, JSPs; 2) bibliotecas de clases Java utilitarias; 3) elementos web estáticos como paginas HTML, imágenes, sonidos; 4) componentes de cliente dinámicos como lo son los Applets, JavaBeans y clases; 5) un descriptor de desarrollo y de configuración de la aplicación web, en forma de uno o de múltiples archivos en formato XML (fundamentalmente web.xml). Este archivo contiene información que permite definir el entorno de ejecución de la aplicación web así como relacionar entre sí los componentes: relación entre las URL y los Servlets/JSP, comportamiento de la aplicación por defecto (página de inicio, página de error), directivas de seguridad, definiciones de los recursos de acceso a los datos, definición de los recursos EJB, (Aumaille, 2002).

2.2.4. Componentes de una aplicación web

Una aplicación web (Guerrero, 2012) contiene componentes diferentes. Cuando se crea una aplicación web, debe comprender cómo implementar y utilizar todos los componentes. Los componentes de una aplicación web (Itescam, Itescam, 2008) son:

- Lógica de negocio

- Parte más importante de la aplicación
- Define los procesos que involucran a la aplicación
- Conjunto de operaciones requeridas para proveer el servicio

- Administración de los datos
 - Manipulación de BD y archivos

- Interfaz
 - Los usuarios acceden a través de navegadores, móviles, PDAs, etc.
 - Funcionalidad accesible a través del navegador
 - Limitada y dirigida por la aplicación

Las aplicaciones web se modelan mediante lo que se conoce como modelo de capas. Una capa representa un elemento que procesa o trata información.

2.2.5. Modelo de capas

El modelo de capas (Maria Isabel Alfonso Galipienso, El Modelo de Capas, 2005) de una arquitectura (algunas veces denominada modelo de máquina abstracta) organiza el sistema en capas, cada una de las cuales proporciona un conjunto de servicios. Cada capa puede pensarse como una máquina abstracta cuyo lenguaje máquina se define por los servicios proporcionados por la capa. Este lenguaje se usa para implementar el siguiente nivel de la máquina abstracta.

Por ejemplo, una forma usual de implementar un lenguaje es definir un lenguaje máquina ideal y compilar el lenguaje para convertirlo en código para esta máquina.

Una vez definido el modelo de capas a utilizarse es necesario definir el patrón de diseño.

2.2.6. Patrón de diseño de arquitectura

La noción de utilizar patrones de diseño para crear sistemas de software se originó en el campo de la arquitectura. Al diseñar edificios, los arquitectos utilizan un conjunto de elementos de diseño arquitectónico establecidos, como arcos y columnas. Diseñar con arcos y columnas es una estrategia comprobada para construir edificios sólidos; estos elementos pueden verse como patrones de diseño arquitectónico.

En el software, los patrones de diseño no son clases ni objetos. Los diseñadores utilizan patrones de diseño para construir conjuntos de clases y objetos. Para utilizar los patrones de diseño con efectividad, los diseñadores deben familiarizarse a sí mismos con los patrones más populares y efectivos que se utilicen en la industria de la ingeniería de software, (Paul J. Deitel, Descubrimiento de patrones de diseño: introducción, 2004).

Otro de los factores a tomar en cuenta en la arquitectura del software son los Requerimientos No Funcionales.

2.2.7. Requerimientos No Funcionales

Los Requerimientos No Funcionales, como su nombre sugiere, son aquellos requerimientos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de este como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento.

De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema.

Los Requerimientos No Funcionales rara vez se asocian con características particulares del sistema. Más bien, estos requerimientos especifican o restringen las propiedades emergentes del sistema.

Estos requerimientos no solo se refieren al sistema software a desarrollar. Algunos de estos requerimientos pueden restringir el proceso que se debe utilizar para desarrollar el sistema.

A la vez estos surgen de las necesidades del usuario, debido a las restricciones en el presupuesto, a las políticas de la organización, a la necesidad de interoperabilidad con otros sistemas software o hardware, o a factores externos como regulaciones de seguridad o legislaciones sobre privacidad, (Maria Isabel Alfonso Galipienso, Requerimientos No Funcionales, 2005).

2.2.7.1. Escalabilidad

A medida que un sistema crece se hace necesario disponer de más recursos para seguir ofreciendo el mismo nivel de servicio. En función de la capacidad de un sistema para crecer acorde a sus necesidades, se establece su nivel o grado de escalabilidad. La escalabilidad se ve desde diferentes puntos de vista:

- El número de usuarios que accede al sistema de forma concurrente
- El volumen de datos que tiene que ser almacenado de forma persistente

- El número de transacciones que deben ser ejecutadas en paralelo, (Santi Caballé, 2008).

2.2.7.2. Portabilidad

La capacidad de que una aplicación sea ejecutada fácilmente sobre diferentes plataformas de software/hardware diferentes de la plataforma para la que fue diseñada, (Gorton, 2006).

2.2.7.3. Seguridad

La solución debe reflejar patrones de seguridad teniendo en cuenta la sensibilidad de la información personal de los usuarios que utilizarán la aplicación. La seguridad se basa en proteger la información y evitar que sea manipulada o accedida, (Gorton, 2006).

2.2.7.4. Integración

Es la facilidad con que una aplicación puede ser incorporada de manera útil en el contexto de una aplicación más amplia. La integración se puede dar de dos formas: permitiendo que otros accedan los datos de la aplicación o creando programas de interfaz (APIs) que controlen el acceso a los datos para terceros. (Gorton, 2006).

2.2.7.5. Disponibilidad

Se relaciona con la confiabilidad de una aplicación. Si una aplicación no está disponible para su uso cuando se necesita, entonces es improbable que cumpla sus requerimientos no funcionales. Es fácil de especificar y medir.

Muchas aplicaciones deben estar disponibles durante las horas laborales. La mayoría de sitios de Internet necesitan estar disponibles el 100% del tiempo porque allí no existen horarios. (Gorton, 2006)

2.2.7.6. Confiabilidad

Termino que se refiere a la madurez del sistema, a tolerar defectos y tener la facilidad de recuperarse de errores.

3. PROCESO DE ELABORACIÓN DEL SOFTWARE

En esta sección se determina el proceso de elaboración del software por el cual será sometido el proyecto, con la finalidad de cumplir con la funcionalidad del sistema.

3.1. Análisis y diseño

El análisis y diseño según (Alarcón, 2006) son dos etapas que habitualmente aparecen unidas en la resolución de problemas. El análisis de sistemas se centra en qué se tiene que hacer, mientras que el diseño de sistemas se centra en cómo se tiene que realizar.

3.2. Casos de uso

Un caso de uso según (Areba J. B., 2001) es una manera de utilizar el sistema o de interactuar con él. Los casos de uso proporcionan una definición de las necesidades a cubrir por un proyecto desde el punto de vista del usuario.

3.3. Componentes de la interfaz gráfica

Una interfaz gráfica del usuario (GUI, graphical user interface) utiliza imágenes (llamadas iconos) y menús que aparecen en la pantalla para enviar comandos al sistema de la computadora.

A muchas personas les resulta más fácil usar el GUI porque los usuarios comprenden las funciones en forma intuitiva, (Stair, 1997).

3.4. Arquitectura de la Solución

Documento en el cual se detallan las características de la arquitectura de la aplicación a ser desarrollada y los componentes que se utilizarán.

3.5. Diagrama de clases

Un diagrama de clases permite modelar el esquema de una base de datos. A la vez permite visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de contenido.

3.6. Diagrama de secuencia

Un diagrama de secuencia muestra la interacción que representa la secuencia de mensajes entre las instancias de clases, componentes, subsistemas o actores. El tiempo fluye hacia abajo en el diagrama y muestra el flujo de control de un participante a otro.

3.7. Diagrama de estados

Un diagrama de estados es un diagrama utilizado para identificar cada una de las rutas o caminos que puede tomar un flujo de información luego de ejecutarse cada proceso.

3.8. Diagrama entidad-relación

Un diagrama o modelo entidad-relación es una herramienta para el modelado de datos, que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades.

4. DOCUMENTACIÓN

En esta sección se incluyen lo que es la documentación final de la elaboración del proyecto.

4.1. Manual de Usuario

Documento que contiene las instrucciones a seguir para utilizar el sistema de forma adecuada.

4.2. Manual Técnico

Documento técnico que contiene información de las operaciones que realizan internamente los sistemas detallándose cada uno de sus componentes internos y funcionalidad.

4.3. Diccionario de Datos

Documento técnico a nivel de base de datos que contiene información de cada una de las tablas del sistema, así como de cada uno de sus atributos, relaciones con las demás tablas e información propia de cada tabla.

4.4. Diagrama de Procesos

Diagrama de flujo que contiene la representación gráfica del algoritmo o proceso del sistema.

4.5. Scripts del Modelado de la Base de Datos

Son documentos que dentro de ellos se contiene la información detallada de la elaboración de todas las tablas pertenecientes al sistema, así como sus relaciones entre sí de cada una de las tablas.

5. METODOLOGÍA A UTILIZAR Y TÉCNICA CORRESPONDIENTE

Para la implementación de la arquitectura web del sistema, previo su análisis, desarrollo, implementación y elaboración de pruebas, fue necesario que en este estudio de investigación se realizaran seis fases.

5.1. Fase 1 - Configuración de la Red

El proyecto previamente fue definido, por lo que es necesario comenzar con una ardua investigación en la cual se deberá de seleccionar el tipo de red que se utilizará internamente.

Una vez seleccionado, se necesita cotizar el valor de todo el equipo físico para posteriormente comprarlo.

Cuando se obtengan todos los componentes de la red, según el tipo de red que se utilizará, se deberán conectar todos los componentes.

Posteriormente se necesita configurar todos los componentes y realizar pruebas de comunicación entre cada punto de la red.

Se espera elaborar una red interna que permita alojar al sistema y que cumpla con las necesidades básicas.

5.2. Fase 2 - Modelo de Capas

Como primer paso de la fase dos, se debe de definir el modelo de capas a utilizarse. Existe una gran diversidad de modelos de capas que se adaptan al proyecto, pero primero se debe evaluar cual se adapta mejor al sistema que se va a desarrollar y que cumpla con los requisitos del sistema.

Una vez establecido, se necesita separar cada uno de los ambientes e instalar los componentes necesarios en cada uno de los servidores de cada capa.

Se espera tener mejor control y seguridad en el manejo de la información, así como llevar un orden interno en el sistema.

5.3. Fase 3 - Patrón de Arquitectura

Seleccionar el patrón de arquitectura a utilizarse en la elaboración del sistema, el cual dará soluciones a los problemas de arquitectura de software. De todos los patrones existentes se deberá de seleccionar el que mejor se adapte a los requerimientos funcionales del sistema a elaborar.

Una vez seleccionado, se deberá de aplicar a la elaboración del diseño del sistema.

Se espera con esto mejorar el rendimiento de toda la aplicación y que su funcionamiento sea óptimo.

5.4. Fase 4 - Requerimientos No Funcionales

Identificar los puntos que pueden ser vulnerables en el sistema, así como en la infraestructura y red.

Trabajar y sellar las vulnerabilidades del sistema según sea cada uno de los casos que se identifiquen.

Realizar pruebas para verificar que el sistema sea capaz de funcionar en cualquier ambiente, es decir, en diferentes sistemas operativos y diversidad de navegadores web.

Una vez se identifiquen los problemas de portabilidad, es necesario establecer parámetros que permitan identificar las causas que impiden el funcionamiento de la aplicación. Utilizar técnicas de compatibilidad para verificar que el sistema funcione en cualquier plataforma sin importar el sistema operativo que se utilice.

Establecer parámetros que permitan que el sistema sea escalable, es decir, que pueda crecer tanto horizontal como vertical.

Realizar pruebas de estrés con la finalidad de establecer la cantidad de conexiones simultáneas que permitirá el sistema y la carga de información, con la finalidad de obtener disponibilidad.

Se espera con esto darle seguridad al sistema, confiabilidad, portabilidad, escalabilidad y disponibilidad, que son los requerimientos no funcionales del sistema.

5.5. Fase 5 - Lenguaje de programación

Primeramente es necesario investigar los lenguajes de programación más comunes y determinar cuál es el que mejor se adapta a la solución del problema.

Es necesario determinar que el lenguaje seleccionado sea capaz de adaptarse a la arquitectura del sistema.

Establecer reglas de programación con la finalidad de estandarizar el desarrollo del sistema.

Una vez seleccionado el lenguaje y establecidas las reglas, se necesita instalar en los servidores los servicios que utilizarán para que funcione el sistema.

Se espera como resultado final, obtener un lenguaje de programación que se adapte de mejor forma a la arquitectura del sistema.

5.6. Fase 6 - Análisis, Diseño e Implementación de la Aplicación

Una vez se haya establecido la arquitectura web, es necesario elaborar los artefactos que permitan el desarrollo de la aplicación web.

Se necesita recopilar información de varios sitios webs de aplicaciones alimenticias. Se identificaron las aplicaciones que realizan funciones en las cuales permiten al usuario elaborar productos alimenticios pero ninguna de ellas satisfacía las necesidades básicas y la mayoría consistían en juegos o en páginas web privadas de empresas distribuidoras de alimentos.

Elaborar el documento de visión, misión, FODA, análisis, diseño y casos de uso, con la finalidad de especificar los requerimientos y el proceso a llevar a cabo para el desarrollo de la aplicación.

Determinar el diagrama entidad relación en base a la información que se almacenará sobre la base de datos.

Establecer un diagrama de clases en donde se encuentren todos los métodos y atributos a utilizarse en la aplicación.

Desarrollar la aplicación y definir el tipo de pruebas por las cuales la aplicación será sometida y ejecutarlas.

Documentar toda la aplicación. Realizar el manual técnico, manual de usuario, diccionario de datos, modelado de procesos y diccionario de datos, que se utilizarán de soporte de la aplicación. Con esta última fase se espera completar el desarrollo del sistema.

6. RECURSO FÍSICOS Y FINANCIEROS

El presupuesto al ser de índole confidencial, será solamente manejado por medio de un contador contratado para tal objetivo, y siendo sometido a decisión.

Figura 2. Recursos físicos del proyecto

Descripción			
Caja y bancos	Unidades	Precio	Total
Caja	1	Q. 10,000.00	Q. 10,000.00
Bancos	1	Q. 15,000.00	Q. 15,000.00
			Q. 25,000.00
Mobiliario y Equipo	Unidades	Precio	Total
Escritorios	2	Q. 500.00	Q. 1,000.00
Sillas	2	Q. 200.00	Q. 400.00
Computadoras	2	Q. 6,500.00	Q. 13,000.00
Equipo de Oficina		Q. 1,500.00	Q. 1,500.00
Impresora	1	Q. 700.00	Q. 700.00
			Q. 16,600.00
Sueldos y Servicios	Unidades	Precio	Total
Socios	2	Q. 5,000.00 / mensual	Q. 10,000.00
Servicios básicos		Q. 400.00 / mensual	Q. 400.00
Servicios Web	1	Q. 1,500.00 / anual	Q. 1,500.00
			Q. 126,300.00

Fuente: elaboración propia.

Lo cual permite identificar una inversión inicial de aproximadamente Q25,000.00 quetzales, adquiriendo un préstamo bancario para acoplar y generar colchones en los gastos proyectados de manera anual.

Proyectando una proyección de presupuesto anual según depreciaciones y pagos de aproximadamente Q. 143,000.00

Figura 3. Proyección de los activos y pasivos

Activo	
Circulante	
Caja y bancos	Q25,000.00
Total	Q25,000.00
Fijo	
Mobiliario y equipo	Q16,600.00
Depreciación del Mobiliario y equipo	(Q800.00)
Total	Q15,800.00
Pasivo	
Circulante	
Proveedores	Q1,000.00
Alquileres y gastos	Q3,500.00
Gastos Constitución de Sociedad	Q5,000.00
Salarios y Prestaciones	Q10,000.00
Total Pasivo Circulante	Q195,000.00

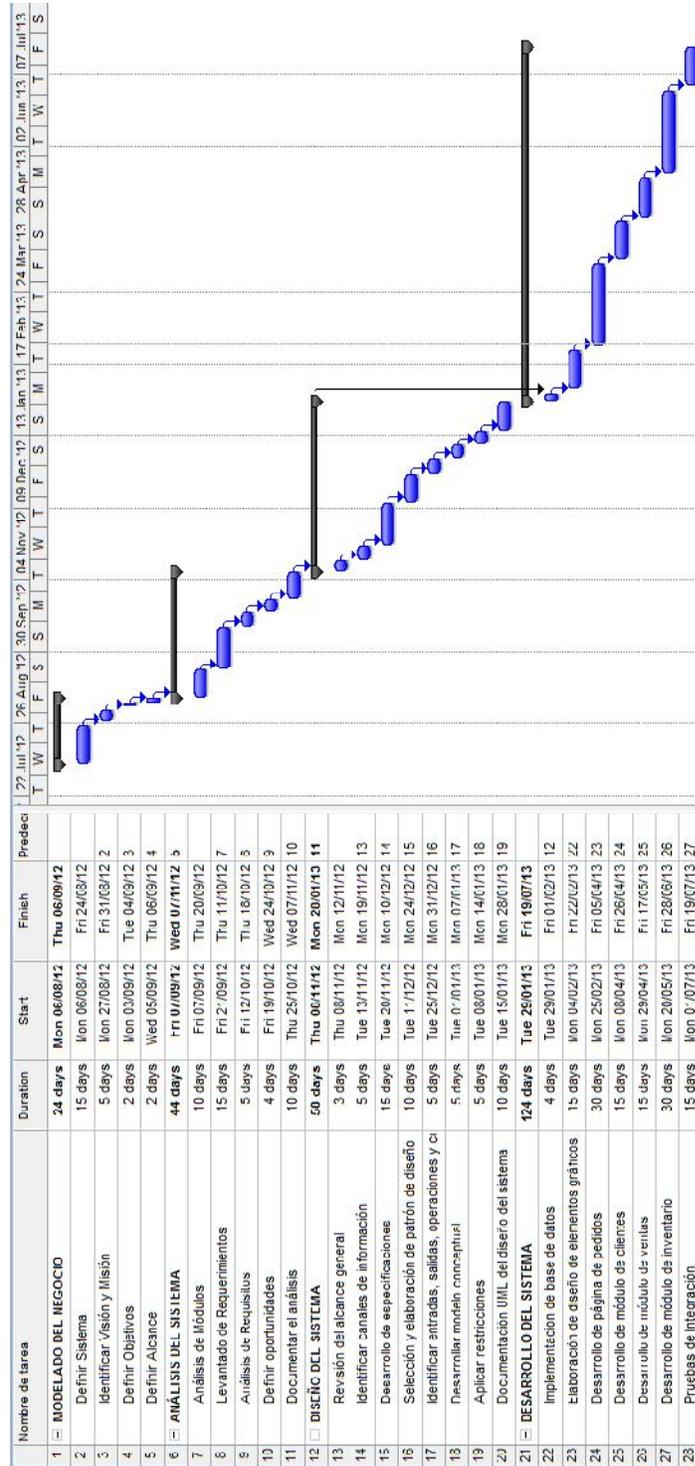
Fuente: elaboración propia.

7. CRONOGRAMA

A continuación se muestra el diagrama de Gantt con el cronograma de actividades para la elaboración del estudio de graduación.

La figura 4, muestra las actividades, sub-actividades, duración en días y relación entre actividades.

Figura 4. Diagrama de actividades Gantt



Fuente: elaboración propia.

8. BIBLIOGRAFÍA

1. Alarcón, V. F. (2006). Analisis de Sistemas de Información. En *Desarrollo de Sistemas de Información* (pág. 69). Catalunya: Edicions UPC.
2. Areba, J. B. (2001). Analisis y Diseño Orientado a Objetos. En J. B. Areba, *Metodología del Analisis Estructurado de Sistemas* (pág. 501). Universidad Pontificia Comillas.
3. Areba, J. B. (2001). Ciclo de Vida del Software. En J. B. Areba, *Ingeniería del Software: Modelización y Técnicas de Desarrollo* (pág. 40). España: Universidad Pontificia Comillas.
4. Aumaille, B. (2002). Aplicaciones Web. En B. Aumaille, *J2EE Desarrollo de aplicaciones Web* (pág. 26). París, Francia: ENI.
5. Busón, S. O. (2007). De la web 2.0 a la web 3.0. En S. O. Busón, *Convergencia de Medios* (págs. 102-103). Barcelona: Icaria Editorial.
6. Carrillo, A. G. (2006). Introducción. En A. G. Carrillo, *Fundamentos de Programación en C++* (pág. 2). Madrid: Delta Publicaciones.
7. catarina.udlap.mx. (2002). *catarina.udlap.mx*. Obtenido de catarina.udlap.mx:
http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/garcia_a_jl/capitulo3.pdf
8. Falgueras, B. C. (2003). Entornos cliente/servidores clásicos. En B. C. Falgueras, *Ingeniería del Software* (págs. 264,265). Catalunya, España: Editorial UOC (February 28, 2003).

9. Gallego, J. C. (2010). Concepto de Sistema Informático. En J. C. Gallego, *Mantenimiento de Sistemas Microinformáticos* (pág. 5). España: Editex.
10. Gorton. (2006). Ian. *Essential Software Architecture*. En Gorton, *Ian. Essential Software Architecture*. Springer Verlag Berlin Heidelberg.
11. Guerrero, V. (13 de Diciembre de 2012). *Componentes de una aplicacion Web Asp.Net*. Recuperado el 09 de 02 de 2013, de Mexico Fisrt Balancan: <http://mfbalancan.blogspot.com/2012/12/componentes-de-una-aplicacion-web-aspnet.html>
12. IST. (2006). <http://informatica.uv.es/>. Obtenido de <http://informatica.uv.es/>: <http://informatica.uv.es/iiguia/IST/Tema1.pdf>
13. Itescam. (2008). *Itescam*. Recuperado el 09 de 02 de 2013, de <http://www.itescam.edu.mx>: <http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r83902.PDF>
14. Maria Isabel Alfonso Galipienso, A. B. (2005). El Modelo de Capas. En I. Sommerville, *Ingenieria del Software* (págs. 227,228). Madrid (España): Pearson Educacion (January 2005).
15. Maria Isabel Alfonso Galipienso, A. B. (2005). Requerimientos No Funcionales. En I. Sommerville, *Ingeniería del software* (págs. 111, 112). Pearson Educacion (Enero 2005).
16. Mora, S. L. (2002). Arquitecturas de las aplicaciones web. En S. L. Mora, *Programación de aplicaciones web* (págs. 54-56). España: Editorial Club Universitario (Noviembre 15, 2002).
17. Paul J. Deitel, H. M. (2004). Descubrimiento de patrones de diseño: Introducción. En H. M. Paul J. Deitel, *Java: Como Programar* (págs. 19, 20). Mexico: Prentice Hall Mexico, 2004.

18. Santi Caballé, F. X. (2008). Beneficios de la computación distribuida. En F. X. Santi Caballé, *Aplicaciones distribuidas en Java con tecnología RMI* (pág. 24). Madrid.
19. Stair, R. M. (1997). Software: Software de Sistemas y de Aplicaciones. En R. M. Stair, *Principios de sistemas de información: Enfoque Administrativo* (pág. 143). The University of Cincinnati.
20. Uniovi. (2009). <http://di002.edv.uniovi.es/>. Obtenido de <http://di002.edv.uniovi.es/>: <http://di002.edv.uniovi.es/~dflanvin/docencia/dasdi/teoria/Transparencias/06.%20Arquitectura%20Web.pdf>

