

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR



FACULTAD DE INGENIERÍA

ESCUELA SISTEMAS

DISERTACION DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO EN SISTEMAS

TEMA:

**PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB**

AUTOR:

ANDRÉS HIDALGO

DIRECTOR DE TESIS:

ING. JAVIER CONDOR

QUITO, ABRIL DEL 2011

# PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

## DEDICATORIA

Dedico esta disertación y toda mi carrera universitaria a Dios por ser quien ha estado a mi lado en todo momento dándome las fuerzas necesarias para continuar luchando día tras día y seguir adelante rompiendo todas las barreras que se me presenten.

A mis padres ya que gracias a ellos soy quien soy hoy en día, fueron los que me dieron ese cariño, calor humano necesario y amor, son los que han velado por mi salud, mis estudios, mi educación, alimentación entre otros, son a ellos a quien les debo todo.

A mis hermanas las cuales han estado a mi lado, y que han estado siempre alerta ante cualquier problema que se me puedan presentar.

A mis amigos más cercanos, a esos amigos que siempre me han acompañado y con los cuales he contado desde que los conocí.

A todos los profesores que me han apoyado una y otra vez entre los cuales se encuentran el Ing. Javier Córdor, Ing. Fabián de la Cruz, Ing. Alfredo Calderón y todos aquellos a quien no menciono pero que han estado de una u otra manera ayudándome a lo largo de la carrera.

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

**AGRADECIMIENTOS**

Al finalizar esta disertación quiero agradecer primeramente A Dios por haberme dado la vida y salud para poder culminar con mi carrera.

A toda mi familia, que fueron participes y se hicieron presentes con sus consejos y apoyo a lo largo de la carrera.

A todas las personas e instituciones que hicieron posible la realización de esta disertación como es la empresa ACTIVELINE CIA. LTDA, con su representante quien es el Sr. Roberto Molina, quién fue la persona quien me dio la idea para hacer esta disertación

Respecto a la elaboración de esta disertación, quiero agradecer entonces al director de tesis quien es el Ing. Javier Córdor por sus sugerencias y correcciones como a los correctores quienes son el Ing. Alfredo Calderón y el Ing. Fabián de la Cruz.

Finalmente quiero dar un agradecimiento a mis compañeros por todos los momentos buenos y malos que como estudiantes pasamos.

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

**INDICE**

**Contenido**

CAPÍTULO 1 .....	7
MARCO TEÓRICO.....	7
1.1 Qué es un prototipo de software.....	7
1.2 Ventajas de desarrollar un prototipo de software en Web.....	7
1.3 Qué es software libre.....	8
1.4 Qué es software de código abierto [7] .....	9
1.4.1 Historia [7].....	9
1.4.2 Movimiento del "código abierto" [7] .....	10
1.5 Cuáles son las herramientas de software libre y código abierto.....	11
1.5.1 Herramientas de Software Libre [8].....	11
1.5.2 Herramientas de código abierto [9].....	16
1.6 Qué es PHP [10] .....	18
1.6.1 Visión general [10] .....	18
1.6.2 Historia[10].....	19
1.6.3 Características de PHP .....	21
1.7 Qué es MySQL [12] .....	22
1.7.1 Historia .....	23
1.7.2 Lenguajes de programación .....	23
1.7.3 Aplicaciones .....	24
1.8 Qué es el servidor apache HTTP [13].....	24
1.8.1 Ventajas.....	25
1.8.2 Uso .....	25
1.8.3 Configuración.....	26
1.9 Cuáles son las fases de desarrollo de software aceptadas por la Ingeniería de Software [14]...	26
1.10 Modelos de ciclo de vida.....	28
1.11 Metodología de Desarrollo de Software [16] .....	32
1.11.1 Valores .....	32
1.11.2 Características fundamentales .....	34

# PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

1.12 Procesos.....	36
1.12.1 Procesos gobernantes .....	36
1.12.2 Procesos habilitantes [19].....	36
1.12.3 Procesos agregadores de valor .....	37
CAPÍTULO 2 .....	38
ESTUDIO DE UN RESTAURANTE TIPO .....	38
2.1 Características del restaurante [A] .....	38
2.2 Procesos del restaurante [A].....	38
2.2.1 Procesos agregadores de valor .....	38
2.2.2 Procesos habilitantes .....	40
2.2.3 Procesos gobernantes .....	44
CAPÍTULO 3 .....	46
ANÁLISIS Y DISEÑO .....	46
3.1 Requerimientos funcionales .....	46
3.2 Requerimientos no funcionales .....	46
3.3 Metodología de Desarrollo de Software [A] .....	47
3.4 Análisis del Prototipo de Software para Restaurantes en Web [A].....	47
3.5 Diseño del Prototipo de Software para Restaurantes en Web .....	50
3.6 Diseño Entidad – Relación. Modelo Conceptual .....	52
3.7 Diseño de reportes.....	53
CAPÍTULO 4 .....	54
CODIFICACIÓN, PRUEBAS E IMPLEMENTACIÓN.....	54
4.1 Versión Alfa.....	54
4.1.1 Requerimientos funcionales del prototipo.....	54
4.1.2 Requerimientos del usuario .....	55
4.2 Versión Beta.....	57
4.2.1 Requerimientos funcionales del prototipo.....	57
4.2.2 Requerimientos del usuario .....	57
4.3 Versión Gamma.....	58
4.3.1 Requerimientos funcionales del prototipo.....	58
4.3.2 Requerimientos del usuario .....	59

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

CAPÍTULO 5 .....	60
CONCLUSIONES Y RECOMENDACIONES .....	60
Conclusiones .....	60
Recomendaciones.....	61
GLOSARIO DE TÉRMINOS .....	63
REFERENCIA BIBLIOGRÁFICA .....	68

## CAPÍTULO 1

### MARCO TEÓRICO

En este primer capítulo veremos toda la información documental que abarca el “PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB” para confeccionar el diseño metodológico de esta investigación, en este capítulo recogeremos toda la información teórica del tema dicho anteriormente, con un análisis de esta información.

Entonces este capítulo abarca conceptos y teorías que estoy utilizando para formular y desarrollar la disertación.

#### 1.1 Qué es un prototipo de software

“Un prototipo de software es una representación limitada del diseño del propio software que permite a las partes responsables de su creación experimentar, probarlo en situaciones reales y explorar su uso.”<sup>[1]</sup>

“El cual es evaluado por el cliente o el usuario para una retroalimentación; gracias a ésta se refinan los requisitos del software que se desarrollará. La interacción ocurre cuando el prototipo se ajusta para satisfacer las necesidades del cliente. Esto permite que al mismo tiempo el desarrollador entienda mejor lo que se debe hacer y el cliente vea resultados a corto plazo.”<sup>[2]</sup>

“Entonces un prototipo de software es un modelo del comportamiento del sistema que puede ser usado para entenderlo completamente o ciertos aspectos de él y así clarificar los requerimientos. Un prototipo es una representación de un sistema, aunque no es un sistema completo, posee las características del sistema final o parte de ellas.”<sup>[3]</sup>

#### 1.2 Ventajas de desarrollar un prototipo de software en Web

A continuación se van a enlistar algunas ventajas de desarrollar un prototipo de software en Web las cuales son las siguientes:

- “Permite al cliente identificar los requisitos detallados de entrada, procesamiento o salida del software en Web.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- Ofrece un mejor enfoque cuando el responsable del desarrollo del software está inseguro de la eficacia de un algoritmo que se esté utilizando en el software, de su adaptabilidad o de la forma que debería tomar la interacción humano -máquina.
- Ayuda al desarrollador del software y al cliente a entender de mejor manera cuál será el resultado de la construcción cuando los requisitos estén satisfechos.
- Permite la reducción de la incertidumbre y del riesgo, reducción de tiempo y de costos, incrementos en la aceptación del software, mejoras en la administración de proyectos, mejoras en la comunicación entre desarrolladores y clientes.
- Cuando se está trabajando con un nuevo software que es en Web con el prototipo de este, el desarrollador puede beneficiarse ampliamente sobre sus competidores siendo innovador si esta aplicación está entre la primera en aparecer en la Web.”<sup>[4]</sup>

### 1.3 Qué es software libre

“El software libre (en inglés free software) son programas desarrollados y distribuidos de acuerdo con la filosofía de dar al usuario la libertad de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar dichos programas.” <sup>[5]</sup>

El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar software libre a "software gratuito"<sup>1</sup> (denominado usualmente freeware), ya que, conservando su carácter de libre, puede ser distribuido comercialmente "software comercial"<sup>2</sup>.<sup>[6]</sup>

El software "libre" garantiza las siguientes libertades:

Libertad	Descripción
0	La libertad de usar el programa, con cualquier propósito.
1	La libertad de estudiar cómo funciona el programa y modificarlo,

<sup>1</sup>**Software gratuito:** Es un tipo de software de computadora que se distribuye sin costo

<sup>2</sup>**Software comercial:** Es el software, libre o no, que es comercializado, es decir, que las compañías que lo producen, cobran dinero por el producto

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

	adaptándolo a tus necesidades.
2	La libertad de distribuir copias del programa, con lo cual puedes ayudar a tu prójimo.
3	La libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie.

Tabla 1 -01 Libertades del software libre [6]

### 1.4 Qué es software de código abierto[7]

El término "código abierto", en inglés es open source.

El término "código abierto" fue acuñado por Christine Peterson del thinktankForesightInstitute, y se registró para actuar como marca registrada el término en inglés para los productos de software libre.

Código abierto es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre.

#### 1.4.1 Historia[7]

Su uso nació por primera vez en 1998 de la mano de algunos usuarios de la comunidad del software libre, tratando de usarlo como reemplazo al ambiguo nombre original en inglés del software libre (free software). Free en inglés significa dos cosas distintas dependiendo del contexto: gratuidad y libertad. Lo cual implica, para el caso que nos ocupa, "software que podemos leer, modificar y redistribuir gratuitamente" (software gratuito) y, además, software libre, según la acepción española de libertad.

El término para algunos no resultó apropiado como reemplazo para el ya tradicional free software, pues eliminaba la idea de libertad, confundida usualmente con la simple gratuidad. No obstante, el término código abierto continúa siendo ambivalente, puesto que se usa en la actualidad por parte de programadores que no ofrecen software libre pero, en cambio, sí ofrecen el código fuente de los programas para su revisión o modificación previamente autorizada por parte de sus pares académicos.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Dada la anterior ambivalencia, se prefiere el uso del término software libre para referirse a programas que se ofrecen con total libertad de modificación, uso y distribución bajo la regla implícita de no modificar dichas libertades hacia el futuro.

Desde el punto de vista de una "traducción estrictamente literal", el significado textual de "código abierto" es que "se puede examinar el código fuente", por lo que puede ser interpretado como un término más débil y flexible que el del software libre. Sin embargo, ambos movimientos reconocen el mismo conjunto de licencias y mantienen principios equivalentes.

Sin embargo, hay que diferenciar los programas de código abierto, que dan a los usuarios la libertad de mejorarlos, de los programas que simplemente tienen el código fuente disponible, previa restricciones sobre su uso o modificación.

Un término que pretende resolver posibles ambigüedades o confusiones que ambos términos generan es FOSS<sup>3</sup> (free and open source software).

En la actualidad el código abierto se utiliza para definir un movimiento nuevo de software, diferente al movimiento del software libre, incompatible con este último desde el punto de vista filosófico, y completamente equivalente desde el punto de vista práctico, de hecho, ambos movimientos trabajan juntos en el desarrollo práctico de proyectos.

La idea bajo el concepto de código abierto es sencilla: cuando los programadores (en Internet) pueden leer, modificar y redistribuir el código fuente de un programa, éste evoluciona, se desarrolla y mejora. Los usuarios lo adaptan a sus necesidades, corrigen sus errores a una velocidad impresionante, mayor a la aplicada en el desarrollo de software convencional o cerrado, dando como resultado la producción de un mejor software.

### **1.4.2 Movimiento del "código abierto"<sup>[7]</sup>**

La idea del código abierto se centra en la premisa de que al compartir el código, el programa resultante tiende a ser de calidad superior al software propietario, es una visión técnica. Por otro lado, el software libre tiene tendencias filosóficas e incluso morales: el software propietario, al no poder compartirse, es "anti ético" dado que prohibir compartir entre seres humanos va en contra del sentido común.

---

<sup>3</sup>FOSS: Es el software libre y de código abierto

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Al igual que el software libre, el código abierto u open source tiene una serie de requisitos necesarios para que un programa pueda considerarse dentro de este movimiento, éstos son:

- Libre redistribución: el software debe poder ser regalado o vendido libremente.
- Código fuente: el código fuente debe estar incluido u obtenerse libremente.
- Trabajos derivados: la redistribución de modificaciones debe estar permitida.
- Integridad del código fuente del autor: las licencias pueden requerir que las modificaciones sean redistribuidas sólo como parches.
- Sin discriminación de personas o grupos: nadie puede dejarse fuera.
- Sin discriminación de áreas de iniciativa: los usuarios comerciales no pueden ser excluidos.
- Distribución de la licencia: deben aplicarse los mismos derechos a todo el que reciba el programa
- La licencia no debe ser específica de un producto: el programa no puede licenciarse solo como parte de una distribución mayor.
- La licencia no debe restringir otro software: la licencia no puede obligar a que algún otro software que sea distribuido con el software abierto deba también ser de código abierto.
- La licencia debe ser tecnológicamente neutral: no debe requerirse la aceptación de la licencia por medio de un acceso por clic de ratón o de otra forma específica del medio de soporte del software.

### 1.5 Cuáles son las herramientas de software libre y código abierto

Existen varias herramientas que son software libre y otras herramientas que tienen código abierto.

Primero empezaremos a revisar algunas herramientas que son software libre.

#### 1.5.1 Herramientas de Software Libre<sup>[8]</sup>

Hay varias herramientas que son software libre. A continuación veremos algunas de ellas:

- **Para creación de documentos e informes**

**OpenOffice.org**

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Es una suite de oficina inspirada en la de Microsoft. Si estás totalmente acostumbrado a las herramientas de MS, ésta es tu opción. Además trae el Open OfficeDraw, que es un programa de dibujo vectorial similar al Corel Draw.

Hay muchos otros paquetes similares (KOffice<sup>4</sup>, SIAG Office<sup>5</sup>, GNOME Office<sup>6</sup>...) y también programas como Abiword y Gnumeric, todos con sus pros y sus contras.

- **Para Compiladores/Intérpretes**

### **gcc**

GNU CompilerCollection<sup>7</sup>, (y no GNU C Compiler). Es el compilador de GNU, que contiene front-ends para C, C++, Objective-C, Fortran, Java, y Ada.

Está incluido en cualquier distribución de GNU/Linux<sup>8</sup> (de hecho, casi todo el sistema operativo se compila en gcc).

### **Free Pascal**

Es un compilador Pascal semánticamente compatible con TP 7.0, así como la mayoría de las versiones de Delphi.

### **GNU Pascal**

El compilador Pascal de GNU, que utiliza a gcc como back-end. Es compatible con Borland Pascal, y soporta muchas de sus unidades (units).

### **Perl**

Perl es un lenguaje interpretado de alto nivel, optimizado para procesar archivos de texto arbitrarios, extraer datos de ellos, e imprimir informes basados en esa información.

### **Python**

Un lenguaje de programación interpretado, interactivo y orientado a objetos, que combina una gran potencia con una alta claridad en la sintaxis. Esta característica hace que

---

<sup>4</sup>**Koffice:** Es una suite ofimática multiplataforma, libre y de código abierto

<sup>5</sup>**Siag office:** Es una suite ofimática fuertemente integrada de software libre para Unix

<sup>6</sup>**Gnome Office:** Es una suite ofimática desarrollada por el Proyecto libre GNOME.

<sup>7</sup>**GNU Compiler Collection:** Es un conjunto de compiladores creados por el proyecto GNU.

<sup>8</sup>**GNU/Linux:** GNU/Linux es uno de los términos empleados para referirse a la combinación del núcleo o kernel libre similar a Unix denominado Linux

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Python sea ideal, tanto para el que está dando sus primeros pasos en programación, como para el programador avanzado.

### PHP

Lenguaje de programación interpretado, fuertemente orientado a desarrollo web.

### gprolog

Un intérprete/compilador Prolog, que cumple casi completamente con el estándar ISO Prolog, añadiendo algunas extensiones bastante útiles. Incluye resolución de condiciones sobre dominios finitos (FD), y una potente interfaz bidireccional entre Prolog y C.

### SWI-Prolog

Otro intérprete/compilador Prolog con licencia libre. Incluye recolector de basura (garbagecollection) e interfaz C/C++, y ofrece una buena performance general.

- **Para entornos de programación**

### Anjuta

Es un IDE<sup>9</sup> para crear aplicaciones GNOME/Gtk+ con Glade<sup>10</sup>, o simplemente para crear aplicaciones genéricas, en C/C++, o cualquier otro lenguaje.

### KDevelop

Puede generar aplicaciones genéricas, o GUIs usando Qt<sup>11</sup>, KDE<sup>12</sup> o GNOME<sup>13</sup>, en C/C++.

### Kylix

Kylix := Delphi in [Linux]. Es software propietario, pero el Kylix Personal Edition es gratuito.

### ViM / Emacs

Son dos editores de texto plano tan flexibles que pueden utilizarse como entornos de programación para prácticamente cualquier lenguaje.

---

<sup>9</sup> **IDE:** Controlador Electrónico Integrado

<sup>10</sup> **Glade:** Es una herramienta de desarrollo visual de interfaces gráficas mediante GTK/GNOME.

<sup>11</sup> **Qt:** Es una biblioteca multiplataforma para desarrollar interfaces gráficas de usuario

<sup>12</sup> **KDE:** Es un proyecto de software libre para la creación de un entorno de escritorio

<sup>13</sup> **GNOME:** Es un entorno de escritorio e infraestructura de desarrollo para sistemas operativos Unix y derivados Unix

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

### Quanta Plus

Es un entorno de desarrollo para HTML<sup>14</sup> con soporte PHP. Está diseñado para el desarrollo rápido de sitios web.

### Eclipse

Un entorno de desarrollo integrado, especialmente orientado a proyectos en Java.

### Mono

Entorno de desarrollo .NET<sup>15</sup>.

- **Para gestión de contenidos**

Una revisión somera de las herramientas para gestión de contenidos que son distribuidas bajo alguna de las licencias consideradas para software libre muestra la existencia de un gran número que ofrecen variadas prestaciones y orientaciones. Brevemente, su arquitectura técnica se fundamenta en el tercer servidor web, intérprete de lenguaje de programación y gestor de base de datos. A este esquema responde el conocido acrónimo LAMP<sup>16</sup> (Linux, Apache, MySQL, PHP), o su versión Windows, WAMP<sup>17</sup>. Precisamente han sido PHP y MySQL las herramientas más extendidas entre los sistemas libres para gestión de contenidos, ya que la mayoría de las soluciones se basan en ellos.

**Plataformas para desarrollo de gestión de contenidos:** se trata de soluciones que ofrecen la plataforma necesaria para desarrollar e implementar aplicaciones que den solución a necesidades específicas. Ofrecen un entorno y unas herramientas de desarrollo.

Tenemos las siguientes:

- Zope, <http://www.zope.org/>
- Typo3, <http://www.typo3.org/>
- Midgard Project, <http://www.midgard.project.org/>
- OpenCMS, <http://www.opencms.org/>

---

<sup>14</sup> **HTML:** Es el lenguaje de marcado predominante para la elaboración de páginas web.

<sup>15</sup> **.NET:** es un framework de Microsoft que hace un énfasis en la transparencia de redes.

<sup>16</sup> **Lamp:** El acrónimo 'LAMP' se refiere a un conjunto de subsistemas de software necesarios para alcanzar una solución global

<sup>17</sup> **WAMP:** Es el acrónimo usado para describir un sistema de infraestructura de internet

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- Apache Lenya, <http://lenya.apache.org/>

**Portales:** la creación y mantenimiento de portales, como soporte y herramienta básica de los servicios de información web, es una de las funciones principales que desempeñan los sistemas de gestión de contenidos para portales.

Tenemos las siguientes:

- PHP Nuke, <http://phpnuke.org/>
- Drupal, <http://www.drupal.org/>
- Mambo, <http://www.mamboserver.com/>
- Plone (requiere Zope), <http://www.plone.org/>

**Aula virtual:** entornos que ofrecen las prestaciones necesarias para crear contenidos para aprendizaje en línea, y ciertos mecanismos de interacción, como foros, chats, evaluación interactiva, etc.

Las plataformas más conocidas son:

- Claroline, <http://www.claroline.net/>
- Moodle, <http://www.moodle.org/>

**Bibliotecas digitales:** las bibliotecas digitales, el paradigma para los servicios de información de finales del siglo XX y comienzos del siglo XXI, organizadas alrededor del terceto que forman los usuarios, las colecciones, y los servicios de valor añadido, se configuran como un espacio altamente especializado para la gestión de contenidos.

Tenemos las siguientes:

- Fedora, <http://www.fedora.info/>
- Dspace, <http://www.dspace.org/>
- Greenstone, <http://www.greenstone.org/>

**Publicaciones digitales:** son plataformas especialmente diseñadas teniendo en cuenta las necesidades de las publicaciones digitales, tales como periódicos, revistas, etc.

Tenemos las siguientes:

- Cofax, <http://www.cofax.org/>

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- Open Journal Systems, <http://www.pkp.ubc.ca/ojs/>
- ePrints, <http://www.eprints.org/>

**Blogs o bitácoras:** los blogs son el fenómeno, ya consolidado, que ha dado un verdadero potencial democrático a la web, a pesar de los inconvenientes que pueda conllevar.

- WordPress, <http://wordpress.org/>

### 1.5.2 Herramientas de código abierto [9]

#### **Herramientas imprescindibles para el diseño vectorial.**

Freehand – Illustrator –CorelDraw (programas NO open source)

Herramienta alternativa: InkScape

Información sobre InkScape

Inkscape es un editor de gráficos vectoriales de código abierto, con capacidades similares a Illustrator, Freehand, CorelDraw o Xara X, usando el estándar de la W3C<sup>18</sup>: el formato de archivo Scalable Vector Graphics (SVG<sup>19</sup>). Las características soportadas incluyen: formas, trazos, texto, marcadores, clones, mezclas de canales alfa, transformaciones, gradientes, patrones y agrupamientos. Inkscape también soporta meta-datos Creative Commons, edición de nodos, capas, operaciones complejas con trazos, vectorización de archivos gráficos, texto en trazos, alineación de textos, edición de XML directo y mucho más. Puede importar formatos como Postscript, EPS, JPEG, PNG, y TIFF y exporta PNG así como muchos formatos basados en vectores.

El objetivo principal de Inkscape es crear una herramienta de dibujo potente y cómoda, totalmente compatible con los estándares XML<sup>20</sup>, SVG y CSS<sup>21</sup>. También queremos mantener una próspera comunidad de usuarios y desarrolladores usando un sistema de

---

<sup>18</sup> **W3C:** El World Wide Web Consortium, abreviado W3C, es un consorcio internacional

<sup>19</sup> **Scalable Vector Graphics (SVG):** Es una especificación para describir gráficos vectoriales bidimensionales, tanto estáticos como animados (estos últimos con ayuda de SMIL), en formato XML.

<sup>20</sup> **XML:** eXtensible Markup Language. Lenguaje Extensible de Marcado. Lenguaje desarrollado por el W3 Consortium para permitir la descripción de información contenida en el WWW a través de estándares y formatos comunes

<sup>21</sup> **CSS:** Las hojas de estilo en cascada (en inglés Cascading Style Sheets), CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML).

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

desarrollo abierto y orientado a las comunidades, y estando seguros de que Inkscape sea fácil de aprender, de usar y de mejorar.

### **Editores de imágenes**

Adobe Photoshop (programas NO opensource)

Herramienta alternativa: GIMP

GIMP es el acrónimo para GNU Image Manipulation Program. Es un programa libre apropiado para tareas como retoque fotográfico, y composición y edición de imagen. Es especialmente útil para la creación de logotipos y otros gráficos para páginas web. Tiene muchas de las herramientas y filtros que se esperaría encontrar en programas comerciales similares, así como algunos interesantes extras.

### **Editor de HTML**

Adobe Dreamweaver (programas NO opensource)

Herramientas alternativas:

Kate<sup>22</sup>, Geany<sup>23</sup>, gPHPEdit, Eclipse<sup>24</sup>, Aptana.

Tienen IDEs bastante interesantes aunque la mayoría está orientada para la edición de PHP.

### **Editor CSS**

cssed-0.4.0-gtk2

### **Gestores de Base de Datos**

MySQL, SQLite, PostgreSQL

### **Lenguajes de programación para Desarrollos Web**

PHP, Ruby on Rails

### **Editores multimediales**

---

<sup>22</sup> **Kate:** Es un editor de textos para el entorno de escritorio KDE.

<sup>23</sup> **Geany:** Es un editor de texto ligero basado en Scintilla con características básicas de Entorno de desarrollo integrado (IDE).

<sup>24</sup> **Eclipse:** Es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido"

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Adobe Flash CS3 y ahora el nuevo CS4

Herramientas alternativas: OSFlash

Pencil - Animaciones en 2D

Pencil es una herramienta de código abierto y multiplataforma con la que es posible dibujar y generar animaciones en 2D.

Pencil no pretende competir con Flash, pese a que las animaciones están basadas en fotogramas clave y distintas capas. A diferencia de Flash, en Pencil es necesario dibujar a mano cada uno de los cuadros que componen la animación.

### Servidores Web

Apache, Nginx

#### 1.6 Qué es PHP<sub>[10]</sub>

**PHP** es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+<sup>25</sup>.

PHP es un acrónimo recursivo que significa *PHP Hypertext Pre-processor* (inicialmente PHP Tools, o, *Personal Home Page Tools*). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.

##### 1.6.1 Visión general<sub>[10]</sub>

PHP es un lenguaje interpretado de propósito general ampliamente usado, diseñado especialmente para desarrollo web y que puede ser incrustado dentro de código HTML.

---

<sup>25</sup> **GTK+ o The GIMP Toolkit:** Es un conjunto de bibliotecas multiplataforma para desarrollar interfaces gráficas de usuario (GUI)

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores, el número de sitios en PHP ha compartido algo de su preponderante sitio con otros nuevos lenguajes no tan poderosos desde agosto de 2005. Es también el módulo Apache más popular entre las computadoras que utilizan Apache como servidor web. La versión más reciente de PHP es la 5.3.2 (for Windows) del 04 de marzo de 2010.

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

Aunque todo en su diseño está orientado a facilitar la creación de página web, es posible crear aplicaciones con una interfaz gráfica para el usuario, utilizando la extensión PHP-Qt<sup>26</sup> o PHP-GTK<sup>27</sup>. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo, a esta versión de PHP se la llama PHP CLI (Command Line Interface).

Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF, Flash, así como imágenes en diferentes formatos.

### 1.6.2Historia<sub>[10]</sub>

Fue originalmente diseñado en Perl, con base en la escritura de un grupo de CGI binarios escritos en el lenguaje C por el programador danés-canadiense Rasmus Lerdorf en el año 1994 para mostrar su currículum vitae y guardar ciertos datos, como la cantidad de tráfico

---

<sup>26</sup> **PHP-Qt:** Es una extensión para PHP5 que permite escribir programas con el Toolkit de Qt.

<sup>27</sup> **PHP-GTK:** Es una extensión para el lenguaje de programación PHP que permite la utilización de GTK+.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

que su página web recibía. El 8 de junio de 1995 fue publicado "Personal Home Page Tools" después de que Lerdorf lo combinara con su propio Form Interpreter para crear PHP/FI.

### PHP 3

Dos programadores israelíes del Technion, Zeev Suraski y Andi Gutmans, reescribieron el analizador sintáctico (parser en inglés) en el año 1997 y crearon la base del PHP3, cambiando el nombre del lenguaje a la forma actual. Inmediatamente comenzaron experimentaciones públicas de PHP3 y fue publicado oficialmente en junio del 1998.

### PHP 4

En mayo de 2000 PHP 4 fue lanzado bajo el poder del motor Zend Engine 1.0. El día 13 de julio de 2007 se anunció la suspensión del soporte y desarrollo de la versión 4 de PHP, a pesar de lo anunciado se ha liberado una nueva versión con mejoras de seguridad, la 4.4.8 publicada el 13 de enero del 2008 y posteriormente la versión 4.4.9 publicada el 7 de agosto de 2008. Según esta noticia se dará soporte a fallos críticos hasta el 2008-08-09

### PHP 5

El 13 de julio de 2004, fue lanzado PHP 5, utilizando el motor Zend Engine 2.0 (o Zend Engine 2). La versión más reciente de PHP es la 5.3.2 (4 de marzo del 2010), que incluye todas las ventajas que provee el nuevo Zend Engine 2 como:

- Mejor soporte para la Programación Orientada a Objetos, que en versiones anteriores era extremadamente rudimentario, con PHP Data Objects.
- Mejoras de rendimiento.
- Mejor soporte para MySQL con extensión completamente reescrita.
- Mejor soporte a XML ( XPath<sup>28</sup>, DOM<sup>29</sup>, etc. ).
- Soporte nativo para SQLite<sup>30</sup>.
- Soporte integrado para SOAP<sup>31</sup>.

---

<sup>28</sup> **Xpath:** (XML Path Language) Es un lenguaje que permite construir expresiones que recorren y procesan un documento XML.

<sup>29</sup> **DOM:** Es esencialmente una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML

<sup>30</sup> **SQLite:** Es un sistema de gestión de bases de datos relacional compatible con ACID

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- Iteradores de datos.
- Manejo de excepciones.
- Mejoras con la implementación con Oracle.
- Aún se siguen publicando versiones de la rama 5.2.X, siendo publicada la versión 5.2.12 el 17 de diciembre de 2009, aunque la mayoría son actualizaciones de seguridad

### PHP 6

Está previsto el lanzamiento en breve de la rama 6 de PHP. Cuando se lance esta nueva versión quedarán solo dos ramas activas en desarrollo (PHP 5 y 6), pues se abandonó el desarrollo y soporte de PHP 4 el 13 de julio de 2007.

Las diferencias que encontraremos frente a PHP 5 son:

- Soportará Unicode;
- Limpieza de funcionalidades obsoletas como `register_globals`, `safe_mode`, etc;
- PECL y eliminación de soporte `ereg`;
- Mejoras en orientación a objetos;
- Inclusión en el núcleo de `xmlReader` y `xmlWriter` así como `Fileinfo`;
- Return por referencia devolverá un error;
- Se retira el soporte de las bibliotecas `FreeType1` y `GD21`;

### 1.6.3 Características de PHP

#### Ventajas

- “Es un lenguaje multiplataforma.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.

---

<sup>31</sup> **SOAP:** (siglas de Simple Object Access Protocol) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite aplicar técnicas de programación orientada a objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (desde PHP5).
- Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar (muchos otros lenguajes tampoco lo hacen), aun estando dirigido a alguna en particular, el programador puede aplicar en su trabajo cualquier técnica de programación y/o desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador (o MVC<sup>32</sup>), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes.”<sup>[10]</sup>

### Inconvenientes

- “PHP no es precisamente el lenguaje más rápido que existe, ya que se trata de un lenguaje interpretado.”<sup>[11]</sup>

### **1.7 Qué es MySQL<sup>[12]</sup>**

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB<sup>33</sup>—desde enero de 2008 una subsidiaria

---

<sup>32</sup> **MVC:** Es un estilo de arquitectura de software que separa los datos de una aplicación

<sup>33</sup> **MySQL AB:** es una compañía de software fundada en 1995

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL <sup>34</sup> para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

### 1.7.1 Historia

SQL (Lenguaje de Consulta Estructurado) fue comercializado por primera vez en 1981 por IBM, el cual fue presentado a ANSI<sup>35</sup> y desde entonces ha sido considerado como un estándar para las bases de datos relacionales. Desde 1986, el estándar SQL ha aparecido en diferentes versiones como por ejemplo: SQL:92, SQL:99, SQL:2003. MySQL es una idea originaria de la empresa opensource MySQL AB establecida inicialmente en Suecia en 1995 y cuyos fundadores son David Axmark, Allan Larsson, y Michael "Monty" Widenius. El objetivo que persigue esta empresa consiste en que MySQL cumpla el estándar SQL, pero sin sacrificar velocidad, fiabilidad o usabilidad.

Michael Widenius en la década de los 90 trató de usar mSQL para conectar las tablas usando rutinas de bajo nivel ISAM<sup>36</sup>, sin embargo, mSQL no era rápido y flexible para sus necesidades. Esto lo llevó a crear una API SQL denominada MySQL para bases de datos muy similar a la de mSQL pero más portable.

### 1.7.2 Lenguajes de programación

Existen varias APIs que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Pascal, Delphi (via dbExpress), Eiffel, Smalltalk, Java (con una implementación nativa del driver de Java), Lisp, Perl, PHP, Python, Ruby, Gambas, REALbasic (Mac y Linux), (x)Harbour (Eagle1),

---

<sup>34</sup> **GNU GPL:** Es una licencia creada por la Free Software Foundation en 1989

<sup>35</sup> **ANSI:** por sus siglas en inglés: American National Standards Institute, es una organización sin ánimo de lucro que supervisa el desarrollo de estándares para productos, servicios, procesos y sistemas en los Estados Unidos.

<sup>36</sup> **ISAM:** método para almacenar información a la que se pueda acceder rápidamente.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

FreeBASIC, y Tcl; cada uno de estos utiliza una API<sup>37</sup> específica. También existe un interfaz ODBC, llamado MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL. También se puede acceder desde el sistema SAP, lenguaje ABAP.

### 1.7.3 Aplicaciones

MySQL es muy utilizado en aplicaciones web, como Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM<sup>38</sup>, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante adelantar monitoreos sobre el desempeño para detectar y corregir errores tanto de SQL como de programación.

### 1.8 Qué es el servidor apache HTTP<sub>[13]</sub>

El servidor HTTP Apache es un servidor web HTTP de código abierto para plataformas Unix (BSD<sup>39</sup>, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que Behelendorf quería que tuviese la connotación de algo que es firme y enérgico pero no agresivo, y la tribu Apache fue la última en rendirse al que pronto se convertiría en gobierno de EEUU, y en esos momentos la preocupación de su grupo era que llegasen las empresas y "civilizasen" el paisaje que habían creado los primeros ingenieros de internet. Además Apache consistía solamente en un conjunto de

---

<sup>37</sup> **API:** Es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

<sup>38</sup> **MyISAM:** Es la tecnología de almacenamiento de datos usada por defecto por el sistema administrador de bases de datos relacionales MySQL.

<sup>39</sup> **BSD:** Son las iniciales de Berkeley Software Distribution (en español, Distribución de Software Berkeley) y se utiliza para identificar un sistema operativo derivado del sistema Unix

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

parches a aplicar al servidor de NCSA. Era, en inglés, a patchy server (un servidor "parcheado").

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: desde 1996, Apache, es el servidor HTTP<sup>40</sup> más usado. Alcanzó su máxima cuota de mercado en 2005 siendo el servidor empleado en el 70% de los sitios web en el mundo, sin embargo ha sufrido un descenso en su cuota de mercado en los últimos años. (Estadísticas históricas y de uso diario proporcionadas por Netcraft).

La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas tan sólo pueden ser aprovechadas por usuarios locales y no remotamente. Sin embargo, algunas se pueden accionar remotamente en ciertas situaciones, o explotar por los usuarios locales malévolos en las disposiciones de recibimiento compartidas que utilizan PHP como módulo de Apache.

### **1.8.1 Ventajas**

Las ventajas son las siguientes: es Modular, de Código abierto, es Multi-plataforma, es Extensible, es Popular (fácil conseguir ayuda/sopORTE)

### **1.8.2 Uso**

Apache es usado primariamente para enviar páginas web estáticas y dinámicas en la World Wide Web. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web.

Apache es el componente de servidor web en la popular plataforma de aplicaciones LAMP, junto a MySQL y los lenguajes de programación PHP/Perl/Python (y ahora también Ruby).

---

<sup>40</sup> **HTTP**: Es el protocolo usado en cada transacción de la World Wide Web.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Este servidor web es redistribuido como parte de varios paquetes propietarios de software, incluyendo la base de datos Oracle y el IBM WebSphere application server. Mac OS X<sup>41</sup> integra apache como parte de su propio servidor web y como soporte de su servidor de aplicaciones WebObjects. Es soportado de alguna manera por Borland en las herramientas de desarrollo Kylix y Delphi. Apache es incluido con Novell NetWare 6.5, donde es el servidor web por defecto, y en muchas distribuciones Linux.

Apache es usado para muchas otras tareas donde el contenido necesita ser puesto a disposición en una forma segura y confiable. Un ejemplo es al momento de compartir archivos desde una computadora personal hacia Internet. Un usuario que tiene Apache instalado en su escritorio puede colocar arbitrariamente archivos en la raíz de documentos de Apache, desde donde pueden ser compartidos.

Los programadores de aplicaciones web a veces utilizan una versión local de Apache con el fin de previsualizar y probar código mientras éste es desarrollado.

Microsoft Internet Information Services (IIS) es el principal competidor de Apache, así como Sun Java System Web Server de Sun Microsystems y un anfitrión de otras aplicaciones como Zeus Web Server. Algunos de los más grandes sitios web del mundo están ejecutándose sobre Apache. La capa frontal (front end) del motor de búsqueda Google está basada en una versión modificada de Apache, denominada Google Web Server (GWS). Muchos proyectos de Wiki media también se ejecutan sobre servidores web Apache.

### **1.8.3 Configuración**

La mayor parte de la configuración se realiza en el fichero apache2.conf o httpd.conf, según el sistema donde esté corriendo. Cualquier cambio en este archivo requiere reiniciar el servidor, o forzar la lectura de los archivos de configuración nuevamente.

### **1.9 Cuáles son las fases de desarrollo de software aceptadas por la Ingeniería de Software<sup>[14]</sup>**

Las fases de desarrollo de software aceptadas por la Ingeniería de Software son las siguientes:

---

<sup>41</sup> **Mac OS X:** Es un sistema operativo desarrollado y comercializado por Apple Inc.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Definición de objetivos:** definir el resultado del proyecto y su papel en la estrategia global.
- **Análisis de los requisitos y su viabilidad:** recopilar, examinar y formular los requisitos del cliente y examinar cualquier restricción que se pueda aplicar.
- **Diseño general:** requisitos generales de la arquitectura de la aplicación.
- **Diseño en detalle:** definición precisa de cada subconjunto de la aplicación.
- **Programación** (programación e implementación): es la implementación de un lenguaje de programación para crear las funciones definidas durante la etapa de diseño.
- **Prueba de unidad:** prueba individual de cada subconjunto de la aplicación para garantizar que se implementaron de acuerdo con las especificaciones.
- **Integración:** para garantizar que los diferentes módulos se integren con la aplicación. Éste es el propósito de la *prueba de integración* que está cuidadosamente documentada.
- **Prueba beta** (o *validación*), para garantizar que el software cumple con las especificaciones originales.
- **Documentación:** sirve para documentar información necesaria para los usuarios del software y para desarrollos futuros.
- **Implementación:** Es la etapa dónde se implementa el software en términos de componentes, es decir, ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares. El objetivo principal de la etapa de implementación es desarrollar la arquitectura y el software como un todo.
- **Mantenimiento:** para todos los procedimientos correctivos (mantenimiento correctivo) y las actualizaciones secundarias del software (mantenimiento continuo).

El orden y la presencia de cada uno de estos procedimientos en el ciclo de vida de una aplicación dependen del tipo de modelo de ciclo de vida acordado entre el cliente y el equipo de desarrolladores.

# PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

## 1.10 Modelos de ciclo de vida

“Para facilitar una metodología común entre el cliente y la compañía de software, los modelos de ciclo de vida se han actualizado para reflejar las etapas de desarrollo involucradas y la documentación requerida, de manera que cada etapa se valide antes de continuar con la siguiente etapa.” [14]

- **Modelo en cascada**

“El modelo de ciclo de vida en cascada comenzó a diseñarse en 1966 y se terminó alrededor de 1970. Se define como una secuencia de fases en la que al final de cada una de ellas se reúne la documentación para garantizar que cumple las especificaciones y los requisitos antes de pasar a la fase siguiente:” [14]

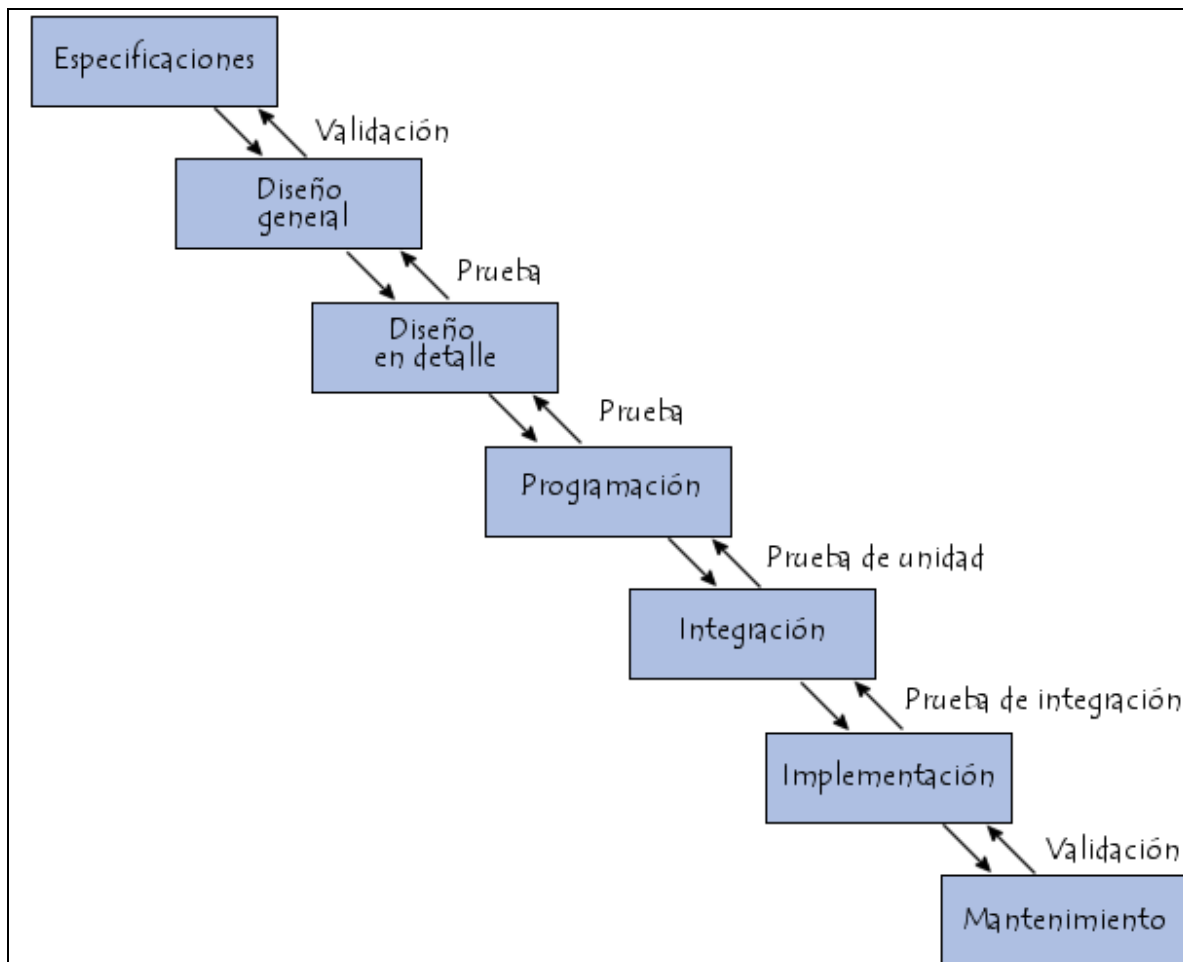


Figura 1 – 01 Modelo en cascada [14]

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Modelo V**

“El modelo de ciclo de vida V proviene del principio que establece que los procedimientos utilizados para probar si la aplicación cumple las especificaciones ya deben haberse creado en la fase de diseño.” [14]

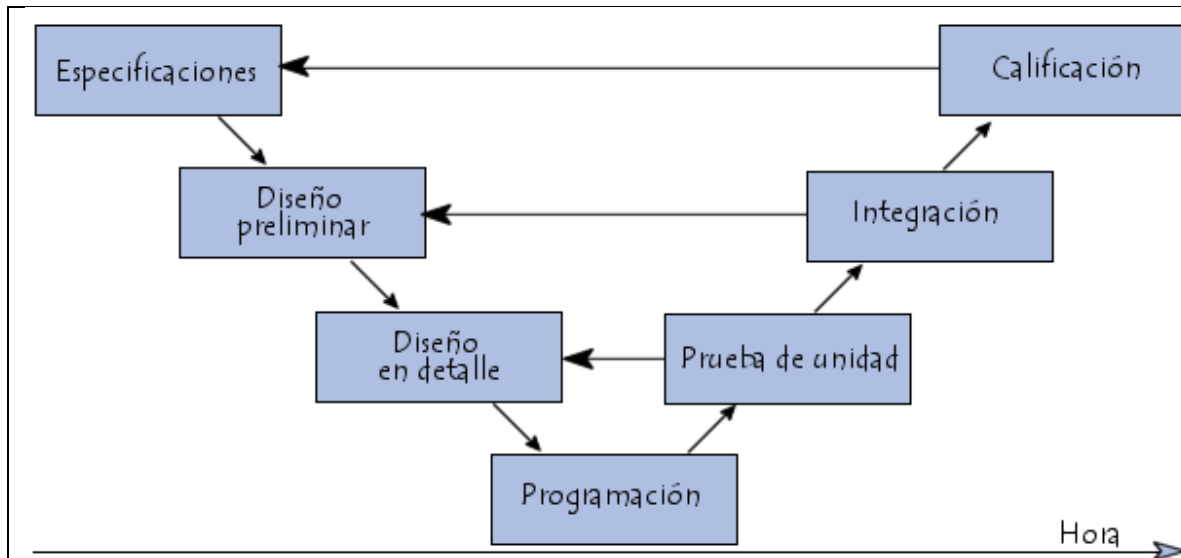


Figura 1 – 02 Modelo V [14]

- **Modelo iterativo incremental**

“En la Figura 1 - 03 se muestra el esquema del Modelo de ciclo de vida Iterativo Incremental, con sus actividades genéricas asociadas. Aquí se observa claramente cada ciclo cascada que es aplicado para la obtención de un incremento; estos últimos se van integrando para obtener el producto final completo. Cada incremento es un ciclo Cascada Realimentado, aunque, por simplicidad, en la Figura 1 - 03 se muestra como secuencial puro.

En la Figura 1 - 03 se observa que existen actividades de desarrollo (para cada incremento) que son realizadas en paralelo o concurrentemente, así por ejemplo, en la figura, mientras se realiza el diseño detalle del primer incremento ya se está realizando en análisis del segundo. La figura 1 -03 es sólo esquemática, un incremento no necesariamente se iniciará durante la fase de diseño del anterior, puede ser posterior (incluso antes), en cualquier tiempo de la etapa previa. Cada incremento concluye con la actividad de “Operación y

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Mantenimiento” (indicada "Operación" en la figura), que es donde se produce la entrega del producto parcial al cliente. El momento de inicio de cada incremento es dependiente de varios factores: tipo de sistema; independencia o dependencia entre incrementos (dos de ellos totalmente independientes pueden ser fácilmente iniciados al mismo tiempo si se dispone de personal suficiente); capacidad y cantidad de profesionales involucrados en el desarrollo; etc.

Bajo este modelo se entrega software “por partes funcionales más pequeñas”, pero reutilizables, llamadas incrementos. En general cada incremento se construye sobre aquel que ya fue entregado.” [15]

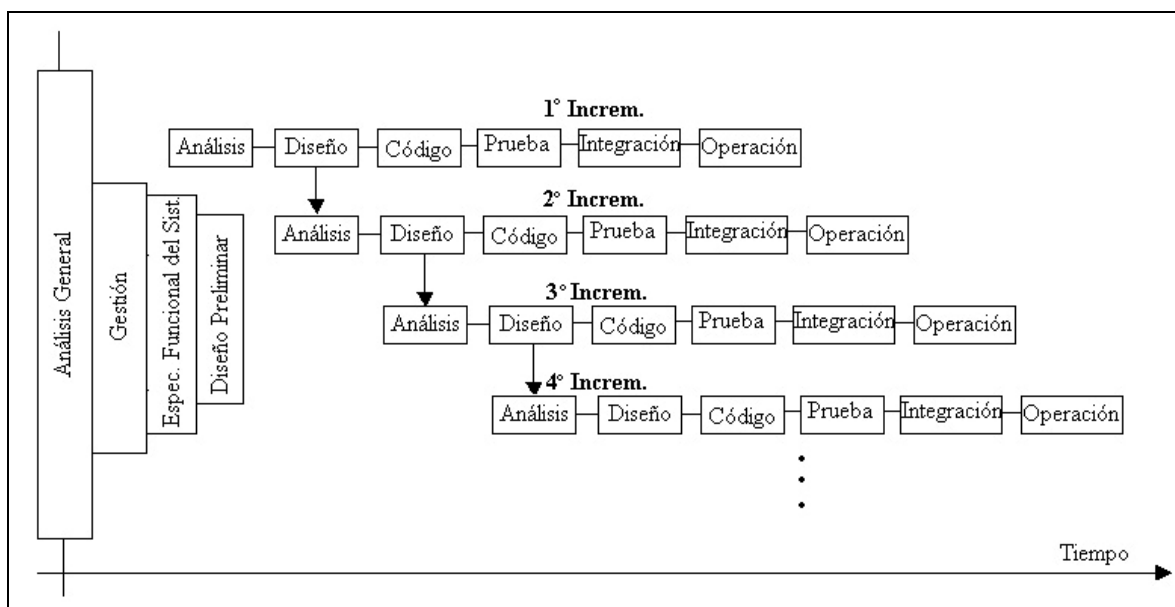


Figura 1 – 03 Modelo iterativo incremental para el ciclo de vida del software [15]

- **Modelo espiral**

“Es un modelo evolutivo que conjuga la naturaleza iterativa del modelo MCP con los aspectos controlados y sistemáticos del Modelo Cascada. Proporciona potencial para desarrollo rápido de versiones incrementales.

El modelo se divide en un número de Actividades de marco de trabajo, llamadas "regiones de tareas".

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Las regiones definidas en el modelo de la Figura 1 - 04 son:

- Región 1 - Tareas requeridas para establecer la comunicación entre el cliente y el desarrollador.
- Región 2 - Tareas inherentes a la definición de los recursos, tiempo y otra información relacionada con el proyecto.
- Región 3 - Tareas necesarias para evaluar los riesgos técnicos y de gestión del proyecto.
- Región 4 - Tareas para construir una o más *representaciones* de la aplicación software.
- Región 5 - Tareas para construir la aplicación, instalarla, probarla y proporcionar soporte al usuario o cliente (Ej. documentación y práctica).
- Región 6 - Tareas para obtener la reacción del cliente, según la evaluación de lo creado e instalado en los ciclos anteriores.” [15]

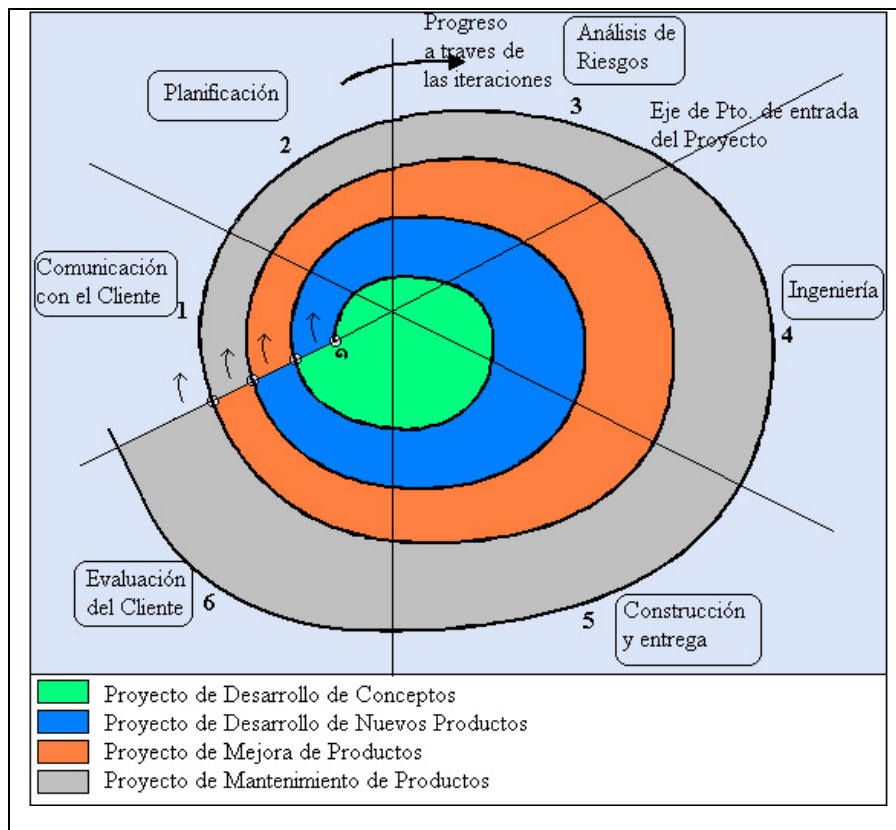


Figura 1 – 04 Modelo espiral para el ciclo de vida del software [15]

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

### 1.11 Metodología de Desarrollo de Software [16]

La programación extrema o eXtreme Programming (XP) es un enfoque de la ingeniería de software<sup>42</sup> formulado por Kent Beck<sup>43</sup>, autor del primer libro sobre la materia, *Extreme Programming Explained: Embrace Change* (1999). Es el más destacado de los procesos ágiles<sup>44</sup> de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

Se puede considerar la programación extrema como la adopción de las mejores metodologías de desarrollo de acuerdo a lo que se pretende llevar a cabo con el proyecto, y aplicarlo de manera dinámica durante el ciclo de vida del software.

#### 1.11.1 Valores

Los Valores originales de la programación extrema son: simplicidad, comunicación, retroalimentación (feedback) y coraje. Un quinto valor, respeto, fue añadido en la segunda edición de *Extreme Programming Explained*. Los cinco valores se detallan a continuación:

- Simplicidad:

La simplicidad es la base de la programación extrema. Se simplifica el diseño para agilizar el desarrollo y facilitar el mantenimiento. Un diseño complejo del código junto a sucesivas modificaciones por parte de diferentes desarrolladores hace que la complejidad aumente exponencialmente. Para mantener la simplicidad es necesaria la refactorización del código, ésta es la manera de mantener el código simple a medida que crece. También se aplica la

---

<sup>42</sup>**Ingeniería de software:** Es la disciplina o área de la informática que ofrece métodos y técnicas para desarrollar y mantener software de calidad.

<sup>43</sup>**Kent Beck:** Es uno de los creadores de la metodología ágil para el desarrollo de software conocida como programación extrema.

<sup>44</sup>**Procesos ágiles:** Los procesos ágiles de desarrollo de software, conocidos anteriormente como metodologías livianas, intentan evitar los tortuosos y burocráticos caminos de las metodologías tradicionales enfocándose en la gente y los resultados.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

simplicidad en la documentación, de esta manera el código debe comentarse en su justa medida, intentando eso sí que el código esté autodocumentado. Para ello se deben elegir adecuadamente los nombres de las variables, métodos y clases. Los nombres largos no decrementan la eficiencia del código ni el tiempo de desarrollo gracias a las herramientas de autocompletado y refactorización que existen actualmente. Aplicando la simplicidad junto con la autoría colectiva del código y la programación por parejas se asegura que cuanto más grande se haga el proyecto, todo el equipo conocerá más y mejor el sistema completo.

- **Comunicación:**

La comunicación se realiza de diferentes formas. Para los programadores el código comunica mejor cuanto más simple sea. Si el código es complejo hay que esforzarse para hacerlo inteligible. El código autodocumentado es más fiable que los comentarios ya que éstos últimos pronto quedan desfasados con el código a medida que es modificado. Debe comentarse sólo aquello que no va a variar, por ejemplo el objetivo de una clase o la funcionalidad de un método. Las pruebas unitarias son otra forma de comunicación ya que describen el diseño de las clases y los métodos al mostrar ejemplos concretos de como utilizar su funcionalidad. Los programadores se comunican constantemente gracias a la programación por parejas. La comunicación con el cliente es fluida ya que el cliente forma parte del equipo de desarrollo. El cliente decide que características tienen prioridad y siempre debe estar disponible para solucionar dudas.

- **Retroalimentación (feedback):**

Al estar el cliente integrado en el proyecto, su opinión sobre el estado del proyecto se conoce en tiempo real. Al realizarse ciclos muy cortos tras los cuales se muestran resultados, se minimiza el tener que rehacer partes que no cumplen con los requisitos y ayuda a los programadores a centrarse en lo que es más importante. Considérense los problemas que derivan de tener ciclos muy largos. Meses de trabajo pueden tirarse por la borda debido a cambios en los criterios del cliente o malentendidos por parte del equipo de desarrollo. El código también es una fuente de retroalimentación gracias a las herramientas de desarrollo. Por ejemplo, las pruebas unitarias informan sobre el estado de salud del

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

código. Ejecutar las pruebas unitarias frecuentemente permite descubrir fallos debidos a cambios recientes en el código.

- Coraje o valentía:

Los puntos anteriores parecen tener sentido común, entonces, ¿por qué coraje? Para los gerentes la programación en parejas puede ser difícil de aceptar, porque les parece como si la productividad se fuese a reducir a la mitad ya que solo la mitad de los programadores está escribiendo código. Hay que ser valiente para confiar en que la programación por parejas beneficia la calidad del código sin repercutir negativamente en la productividad. La simplicidad es uno de los principios más difíciles de adoptar. Se requiere coraje para implementar las características que el cliente quiere ahora sin caer en la tentación de optar por un enfoque más flexible que permita futuras modificaciones. No se debe emprender el desarrollo de grandes marcos de trabajo (frameworks) mientras el cliente espera. En ese tiempo el cliente no recibe noticias sobre los avances del proyecto y el equipo de desarrollo no recibe retroalimentación para saber si va en la dirección correcta. La forma de construir marcos de trabajo es mediante la refactorización del código en sucesivas aproximaciones.

- Respeto:

El respeto se manifiesta de varias formas. Los miembros del equipo se respetan los unos a otros, porque los programadores no pueden realizar cambios que hacen que las pruebas existentes fallen o que demore el trabajo de sus compañeros. Los miembros respetan su trabajo porque siempre están luchando por la alta calidad en el producto y buscando el diseño óptimo o más eficiente para la solución a través de la refactorización del código. Los miembros del equipo respetan el trabajo del resto no haciendo menos a otros, si no orientandolos a realizarlo mejor, obteniendo como resultado una mejor autoestima en el equipo y elevando el ritmo de producción en el equipo.

### **1.11.2 Características fundamentales**

Las características fundamentales del método son:

- Desarrollo iterativo e incremental: pequeñas mejoras, unas tras otras.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- Pruebas unitarias continuas, frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión<sup>45</sup>. Se aconseja escribir el código de la prueba antes de la codificación. Véase, por ejemplo, las herramientas de prueba JUnit<sup>46</sup> orientada a Java, DUnit orientada a Delphi y NUnit para la plataforma.NET. Estas dos últimas inspiradas en JUnit.
- Programación en parejas: se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código escrito de esta manera -el código es revisado y discutido mientras se escribe- es más importante que la posible pérdida de productividad inmediata.
- Frecuente integración del equipo de programación con el cliente o usuario. Se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.
- Corrección de todos los errores antes de añadir nueva funcionalidad. Hacer entregas frecuentes.
- Refactorización del código, es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad pero sin modificar su comportamiento. Las pruebas han de garantizar que en la refactorización no se ha introducido ningún fallo.
- Propiedad del código compartida: en vez de dividir la responsabilidad en el desarrollo de cada módulo en grupos de trabajo distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.
- Simplicidad en el código: es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrá añadir funcionalidad si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo.

La simplicidad y la comunicación son extraordinariamente complementarias. Con más comunicación resulta más fácil identificar qué se debe y qué no se debe hacer. Cuanto más

---

<sup>45</sup>**Pruebas de regresión:** Es a cualquier tipo de pruebas de software que intentan descubrir las causas de nuevos errores (bugs).

<sup>46</sup>**JUnit:** Es un conjunto de bibliotecas creadas por Erich Gamma y Kent Beck que son utilizadas en programación para hacer pruebas unitarias de aplicaciones Java.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

simple es el sistema, menos tendrá que comunicar sobre éste, lo que lleva a una comunicación más completa, especialmente si se puede reducir el equipo de programadores.

### 1.12 Procesos

“La palabra proceso tiene su origen en el término latino processus.”<sup>[17]</sup>

“Se denomina proceso al conjunto de acciones o actividades sistematizadas<sup>47</sup> que se realizan o tienen lugar con un fin.

Un proceso quiere decir distintas combinaciones operativas que ocurren simultáneamente para alcanzar un resultado o un producto, como la instalación de un nuevo software, o la consecución de un análisis antivirus.”<sup>[18]</sup>

#### 1.12.1 Procesos gobernantes

“Los procesos gobernantes orientan la gestión institucional a través de la formulación de políticas, directivas, normas e instrumentos para poner en funcionamiento a la organización.”<sup>[19]</sup>

“Están orientados a definir la planificación y dirección de la organización.”<sup>[20]</sup>

Ejemplo:



Figura 1 – 05 Procesos gobernantes <sup>[19]</sup>

#### 1.12.2 Procesos habilitantes<sup>[19]</sup>

Los procesos habilitantes están encaminados a generar productos y servicios para los procesos gobernantes, agregadores de valor y para sí mismos, viabilizando la gestión institucional.

<sup>47</sup> **Actividades sistematizadas:** Acciones interrelacionadas entre sí que, a partir de una o varias entradas de información, materiales o de salidas de otros procesos, dan lugar a una o varias salidas también de materiales (productos) o información con un valor añadido

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

Ejemplo:



Figura 1 – 06 Procesos habilitantes<sup>[19]</sup>

### 1.12.3 Procesos agregadores de valor

“Los procesos agregadores de valor generan, administran y controlan los productos y servicios destinados a usuarios externos y permiten cumplir con la misión institucional.” <sup>[19]</sup>

“Son la razón de ser de la organización, generan productos para sus clientes externos, y tienen relación directa con la misión institucional.” <sup>[20]</sup>

Ejemplo:



Figura 1 – 07 Procesos agregadores de valor<sup>[19]</sup>

## CAPÍTULO 2

### ESTUDIO DE UN RESTAURANTE TIPO

En este capítulo haremos un detalle de un restaurante tipo que pudiera utilizar este software; por lo tanto veremos un ejemplo de este tipo de restaurantes describiendo sus características, sus procesos, su estructura, su organización y las ventas

#### 2.1 Características del restaurante<sub>[A]</sub>

Este software va dirigido a restaurantes pequeños o medianos dónde se caracterizan por tener la capacidad de máximo de 7 mesas.

El tipo de gente que trabaja son personas que tiene conocimiento solo de cocina, pero son amables, responsables, puntuales y creativos para preparar la comida.

La comida es sana como la comida que se prepara en casa.

El Horario de Atención es de 7H30 a 17H00 de Lunes a Sábado

#### 2.2 Procesos del restaurante<sub>[A]</sub>

Los procesos típicos de un restaurante son: Atención al público, Caja, Reservaciones, Menús, Abastecimiento, Recursos Humanos y el Servicio a la oficina los cuales van a ser detallados a continuación

##### 2.2.1 Procesos agregadores de valor

En la Atención al público hay varios subprocessos y además son los procesos agregadores de valor de este tipo de restaurantes los cuáles son los siguientes:

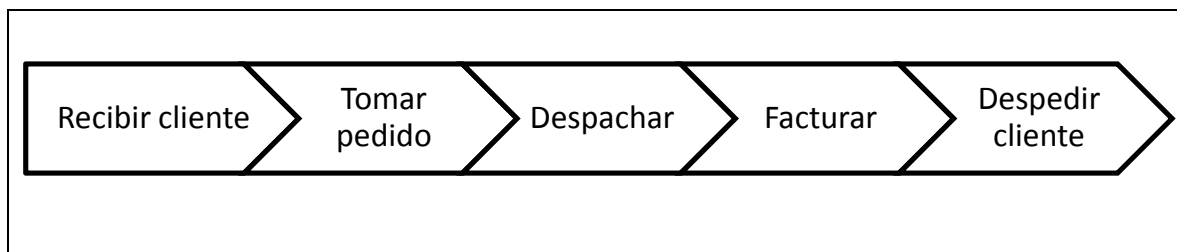


Figura 2 – 01 Procesos agregadores de valor de este tipo de restaurantes<sub>[A]</sub>

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Ingreso del cliente**

El cliente ingresa al restaurante, inmediatamente se sienta en la mesa que él desee.

Generalmente en el horario del almuerzo las mesas están llenas, entonces lo que se hace es si son personas que están solas se les da sillas desocupadas para que esperen hasta que desocupe alguna persona la silla de alguna mesa, y si es un grupo de personas se les pide que esperen para ubicarlos en alguna mesa próxima a salir.

Algunas veces ingresan clientes pasajeros que son impacientes si reinciden en su impaciencia se les pide que cancelen lo que han consumido o sino que hagan el favor de retirarse.

Algunas veces ya se acaba la comida eso se lo dice al cliente y este inmediatamente se retira.

- **Toma del pedido**

Se acerca el personal de servicio y le informa el menú del día para que elija las 2 opciones del primer plato que es (sopa o fruta picada), en el segundo plato que es el plato fuerte escoge entre las opciones de carne como (pescado, pollo, res o cerdo), el jugo elige entre (agua aromática o jugo) y el postre que es fijo.

Verbalmente el personal de servicio le comunica a cocina el pedido que ha realizado el cliente para luego despachar

- **Despacho**

Una vez reciba la orden en la cocina se prepara la orden que el cliente ha pedido y luego se monta el plato de cliente para que el personal de servicio se lo lleve hacia donde está el cliente.

- **Facturación**

Una vez que el cliente termina de servirse su comida se acerca a caja y el administrador del restaurante elabora la factura que luego es extendida hacia el cliente para que éste a su vez pueda cancelar su consumo

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Despedida**

Luego de que el cliente cancela su factura sólo en efectivo o cheque personal al administrador del restaurante, éste a su vez amablemente le agradece por su visita.

### 2.2.2 Procesos habilitantes

Estos son los procesos habilitantes de este tipo de restaurantes los cuáles son los siguientes:



Figura 2 – 02 Procesos habilitantes [A]

- **Caja**

Este tipo de restaurantes por lo general manejan caja chica en dónde se encuentra sólo el dinero de la venta del día.

En la caja solo hay dinero de caja chica y dinero extra.

La caja chica solo hay dinero en efectivo y es la venta constante.

El dinero extra también es solo en efectivo y son las ventas extras lo que se vende de desayunos, postres almuerzos, en otras palabras es lo que sobra de la venta constante.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Reservaciones**

Las reservaciones se las hace con clientes frecuentes o particulares que deseen hacer la reservación de salón completo previo el pago del 50% del total de la factura, y el otro 50% se paga una vez finalizado el evento.

Todas las reservaciones para eventos se las hacen con 2 días de anticipación para hacer las requisiciones del menú

El horario es abierto para el cliente entendiéndose que el cliente puede usar el salón en el momento que él lo desee y luego de 4 horas a partir del despacho de comida se le cobrará al cliente horas extras por el servicio.

El cliente puede hacer la reservación durante todo el día desde las 7:30 am hasta las 17:00 pm.

- **Menús o Platos**

En este tipo de restaurantes por lo general el menú se los divide en desayunos y almuerzos

Los desayunos son los siguientes:

- ❖ **Desayuno continental:**

- Dos huevos revueltos
- Salchicha
- Sanduche de queso
- Jugo
- Un jarro de café: que puede ser en leche o en agua o también agua aromática

- ❖ **Desayuno típico:**

- Dos huevos revueltos
- Salchicha
- Empanada de verde
- Jugo
- Un jarro de café: que puede ser en leche o en agua o también agua aromática

Los almuerzos son de la siguiente manera:

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

### ❖ **Primer plato:**

Sopa: Las sopas más habituales son: Sopa de espinaca, sopa de sancocho, sopa de locro de queso.

Fruta picada: Es un mix de frutas las cuales contiene: papaya, sandía, melón y piña.

### ❖ **Plato fuerte:**

- Pescado: que puede ser Encocado de pescado
- Pollo: que puede ser Pollo salteado con vegetales
- Res: que puede ser Filete al grill
- Cerdo: que puede ser Seco de choncho

### ❖ **Jugo:**

Jugo: que puede ser de Mora, piña, guayaba

Agua aromática que puede ser de horchata, cedrón, toronjil

### ❖ **Postre:**

Una sola opción las cuales pueden ser: higos con queso, torta de chocolate, musse de mora

#### • **Abastecimiento**

El abastecimiento de los productos al restaurante es diario, ya que se hace compras diarias en el Supermercado como en el Santa María y en el mercado como el de Iñaquito.

La persona encargada va con el menú del día a estos lugares y realiza las compras.

El pago de estas compras se lo realiza solo en efectivo.

En el Mercado generalmente se hace las compras de las carnes que son las de res, choncho, pescado y pollo

En el Supermercado generalmente se hace las compras de harinas y de los ingredientes para hacer los postres y el azúcar.

Las frutas y los vegetales se los trae un proveedor.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Recursos humanos**

Como se los hace en la mayoría de las empresas, en este tipo de restaurantes todas las personas que trabajan en estos restaurantes pasaron por un proceso de selección.

La recepción de carpetas se lo hizo publicitando en los medios de comunicación escritos en este caso fue El Comercio.

Los requerimientos fueron personas con conocimiento de cocina y que tengan edad de 18 a 40 años de ambos sexos.

- **Servicio a la oficina**

Se lo hace dentro del perímetro del restaurante, el cliente llama al restaurante, elige las opciones del menú diario, elige el número de almuerzos que desea, en el restaurante se le empaca el pedido del cliente y se le lleva la comida.

- **Organización del restaurante**

Aquí se muestra los ingresos y egresos que tienen estos tipos de restaurantes recalando que aún son restaurantes que no tienen todavía utilidades:

- ❖ **Ingresos:**

- Las ventas de comida diaria a las empresas y personas naturales
- Atención de coffee breaks
- Ventas de tarjetas de comida
- Arriendo del local para reuniones sociales y de trabajo.

- ❖ **Egresos:**

- Gastos de compras diarias de vegetales, frutas y carnes
- Pagos mensuales de arriendo del local
- Gastos básicos de luz, agua y teléfono.
- Sueldos a los empleados
- Pagos al SRI: (son restaurantes que en el SRI tienen categoría medio – alto)
- Pagos a los proveedores de materiales plásticos que son tarinas, vasos desechables, viandas y contenedores.
- Gastos en desinfectantes como cloro, detergente y lava vajillas.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Finanzas del restaurante**

Son restaurantes que en ventas están en un punto de equilibrio con tendencia a la alta.

Esta tendencia de estos restaurantes lo han logrado por el tipo de comida que preparan que es variada, sana, y también por el buen servicio.

Esperan llegar a un punto para registrar utilidades y captar más clientes corporativos porque la mayoría son clientes personales.

Al momento son restaurantes sin pérdidas.

### 2.2.3 Procesos gobernantes

Estos son los procesos gobernantes de este tipo de restaurantes los cuáles son los siguientes:

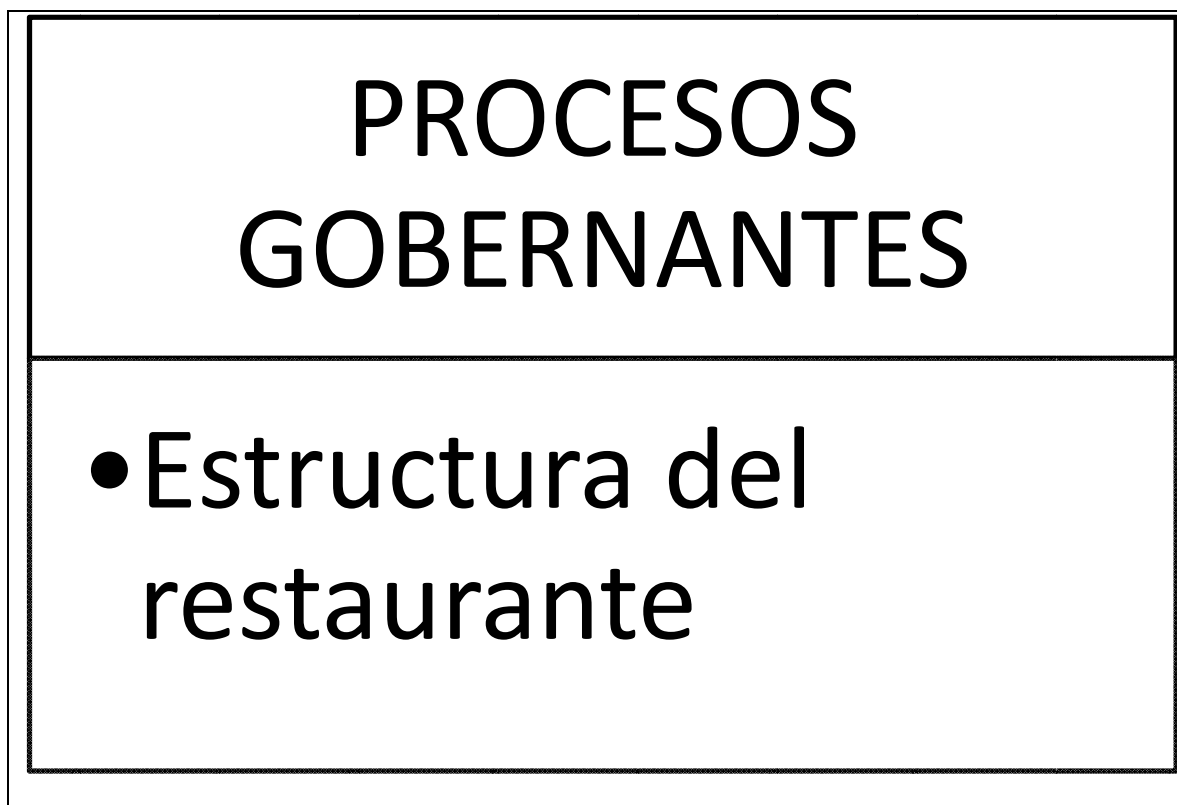


Figura 2 – 03 Procesos gobernantes [A]

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Estructura del restaurante**

Por lo general en este tipo de restaurantes hay el siguiente personal de servicio:

- Un Administrador que igual hace de todo es poli funcional por lo tanto hace de cajero, mesero y cocinero
- Un jefe de cocina
- Dos ayudantes del jefe de cocina
- Un posillero

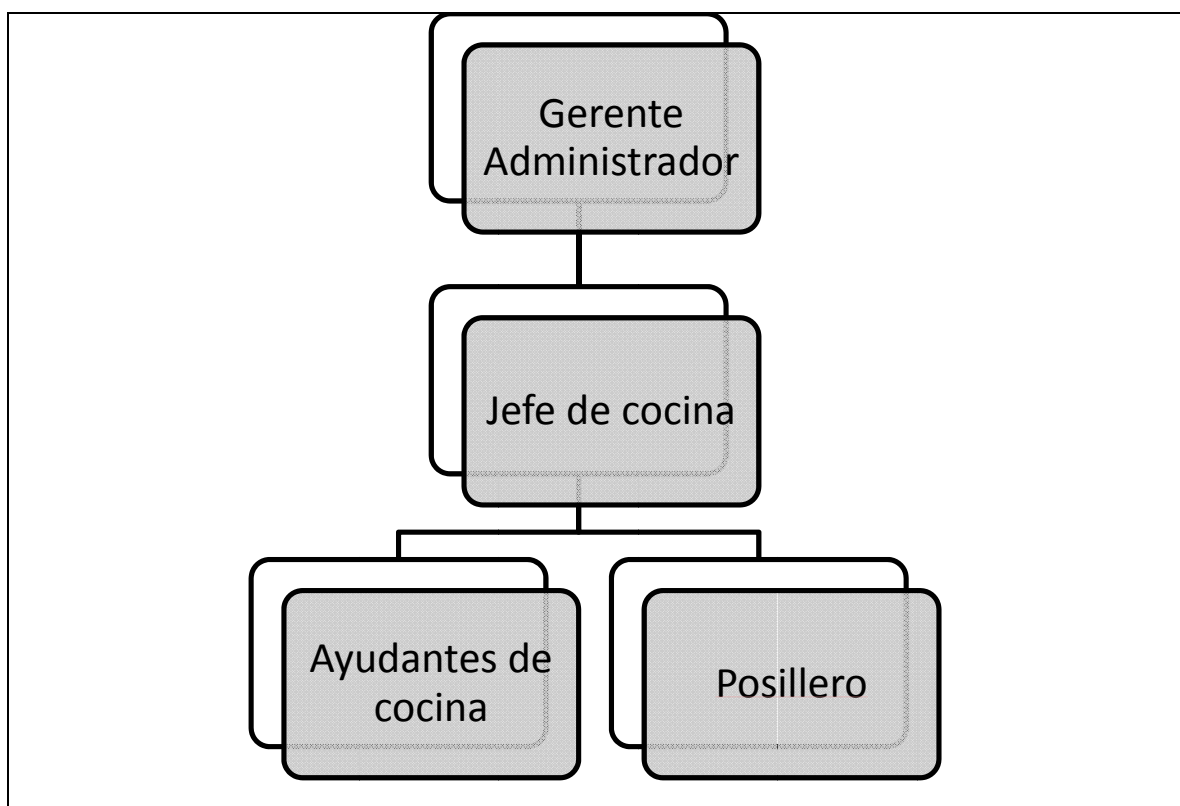


Figura 2 – 04 Organigrama del personal del restaurante [A]

### CAPÍTULO 3

#### ANÁLISIS Y DISEÑO

En este capítulo se hará un análisis de los datos de las diferentes interfaces y del diseño que encierran el Prototipo de Software para Restaurantes en Web.

##### **3.1 Requerimientos funcionales**

Como se ha visto en el capítulo anterior dentro de los procesos que tiene un restaurante mediano o pequeño este software tiene los siguientes requerimientos funcionales:

- El módulo de la administración de las órdenes
- El módulo de la facturación
- El módulo de la administración del menú o platos
- El módulo de la administración de clientes
- El módulo de la administración de los pedidos para la cocina
- El módulo de la administración de los empleados

##### **3.2 Requerimientos no funcionales**

Básicamente estos requerimientos están en el diseño que tiene el prototipo los cuáles son los siguientes:

- El logo que tiene la plantilla del prototipo de software debe ser más pequeño en todas las ventanas de consulta de los módulos de Imágenes, Categorías, Productos, Mesa, Nueva Orden, Clientes, Cocina y Empleados excepto la ventana de Acceso al Sistema.
- El título y la parte de abajo del título que es una descripción que tiene la plantilla del prototipo cambiarle por el título y la descripción del restaurante tipo, del cual se está desarrollando el prototipo.
- Las tablas que tiene las ventanas de consulta de todos los módulos del prototipo debe de tener diferente color en la fila del encabezado con el resto de filas del contenido de las tablas y el resto de filas de las tablas debe de tener un tramado de dos colores diferentes.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- El Copyright que tiene la plantilla del diseño del prototipo debe tener el nombre de la persona quien está desarrollando el prototipo
- Reestructurar el orden del menú del prototipo quedando así: Productos, Mesa, Nueva Orden, Clientes, Cocina, Empleados, Logout
- Ampliar el espacio de la celda donde va estar el contenido que son las tablas de consulta de todos los módulos excepto del módulo de Acceso al Sistema.
- En la factura salga el logo del prototipo del software.

### 3.3 Metodología de Desarrollo de Software<sub>[A]</sub>

La forma en que se va a desarrollar el PROTOTIPO FUNCIONAL ADMINISTRATIVO DEL SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB será utilizando la Metodología de Programación Extrema, en el cuál yo expongo al director de la disertación el prototipo y él me dirá los cambios sobre la marcha que debo realizar al prototipo, para lograr mejoras de éste.

A los cambios que se vayan realizando al prototipo se le hará pruebas unitarias continuas para verificar que los cambios estén correctos y así poder corregir los errores que tenga el prototipo al momento que se le está desarrollando.

### 3.4 Análisis del Prototipo de Software para Restaurantes en Web<sub>[A]</sub>

En esta sección haremos el análisis del prototipo del software mediante sus interfaces en este caso se mostrará las interfaces de un módulo que será el módulo de la Administración de las Nuevas Órdenes conjuntamente con el módulo de la Facturación y del Acceso al Sistema.

Las demás interfaces del resto de módulos irán en el ANEXO A

- **Diseño del prototipo del software**

Esta es una plantilla de restaurantes para web gratis que me descargué para utilizar su diseño en el desarrollo del prototipo del software.

# PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

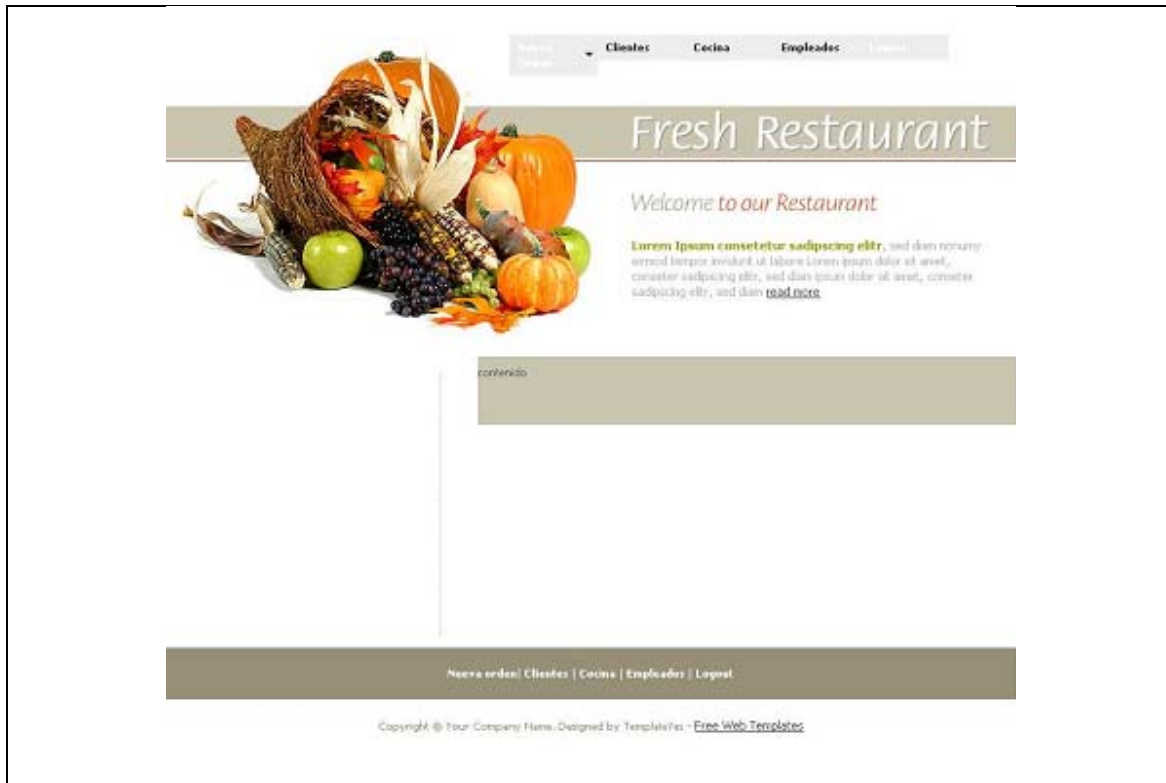


Figura 3 – 01 Diseño del prototipo del software [A]

- **Interface del Acceso al Sistema**

Usuario	<input type="text"/>
Clave	<input type="text"/>
	<input type="button" value="Login"/>

Figura 3 – 02 Interface de Acceso al Sistema

- **Interfaces del Módulo de Nuevas Órdenes**

Escriba el número de la factura

NUEVAORDENID	No FACTURA	No MESA	No PERSONAS	FECHA	NOMBRE DEL CLIENTE	APELLIDO DEL CLIENTE	DESCRIPCION	CANTIDAD	PRECIO	NOMBRE DEL EMPLEADO		

Insertar Nueva Orden

Figura 3 – 03 Interface de la visualización de datos del módulo de las nuevas órdenes

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

No de factura:	<input type="text"/>
Fecha:	<input type="text"/>
Cliente:	<input type="text" value="v"/> <input type="button" value="Buscar"/>
Mesa:	<input type="text" value="v"/>
Número de personas:	<input type="text"/>
	<input type="button" value="Insertar nueva orden"/> <input type="button" value="Cancelar"/>

Figura 3 – 04 Interface de la ventana de insertar nueva orden

No de factura:	<input type="text"/>
Fecha:	<input type="text"/>
Identificador de cliente:	<input type="text" value="v"/> <input type="button" value="Buscar"/>
Identificador de la mesa:	<input type="text" value="v"/>
Identificador del empleado:	<input type="text"/>
Número de personas:	<input type="text"/>
	<input type="button" value="Insertar productos"/> <input type="button" value="Cerrar factura"/>

Figura 3 – 05 Interface de la ventana de visualización la nueva orden

No de factura:	<input type="text"/>
Descuento:	<input type="text"/>
Producto:	<input type="text" value="v"/>
Cantidad:	<input type="text"/>
	<input type="button" value="Insertar"/> <input type="button" value="Cancelar"/>

Figura 3 – 06 Interface de la ventana de insertar el producto, la cantidad y el descuento en el módulo de la nueva orden

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

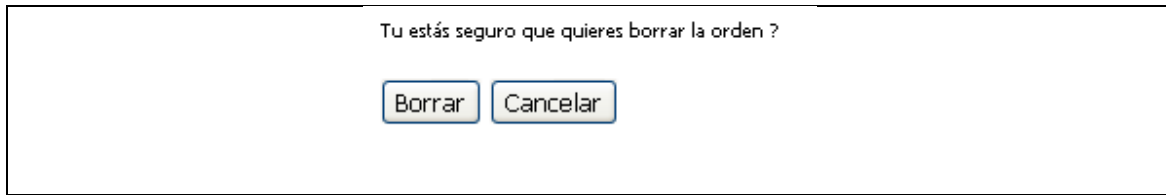


Figura 3 – 07 Interface de la ventana de eliminar nuevas órdenes

### 3.5 Diseño del Prototipo de Software para Restaurantes en Web

En esta sección mostraremos el diseño del prototipo del software usando los diagramas de casos de uso; en esta sección se mostrará el diagrama de caso de uso del prototipo a nivel general como también el diagrama de caso de uso del módulo de la Administración de las Nuevas Órdenes conjuntamente con el módulo de la Facturación y del Acceso al Sistema.

Los demás diagramas de casos de uso del resto de módulos irán en el ANEXO B.

- **Casos de Uso del prototipo del software**

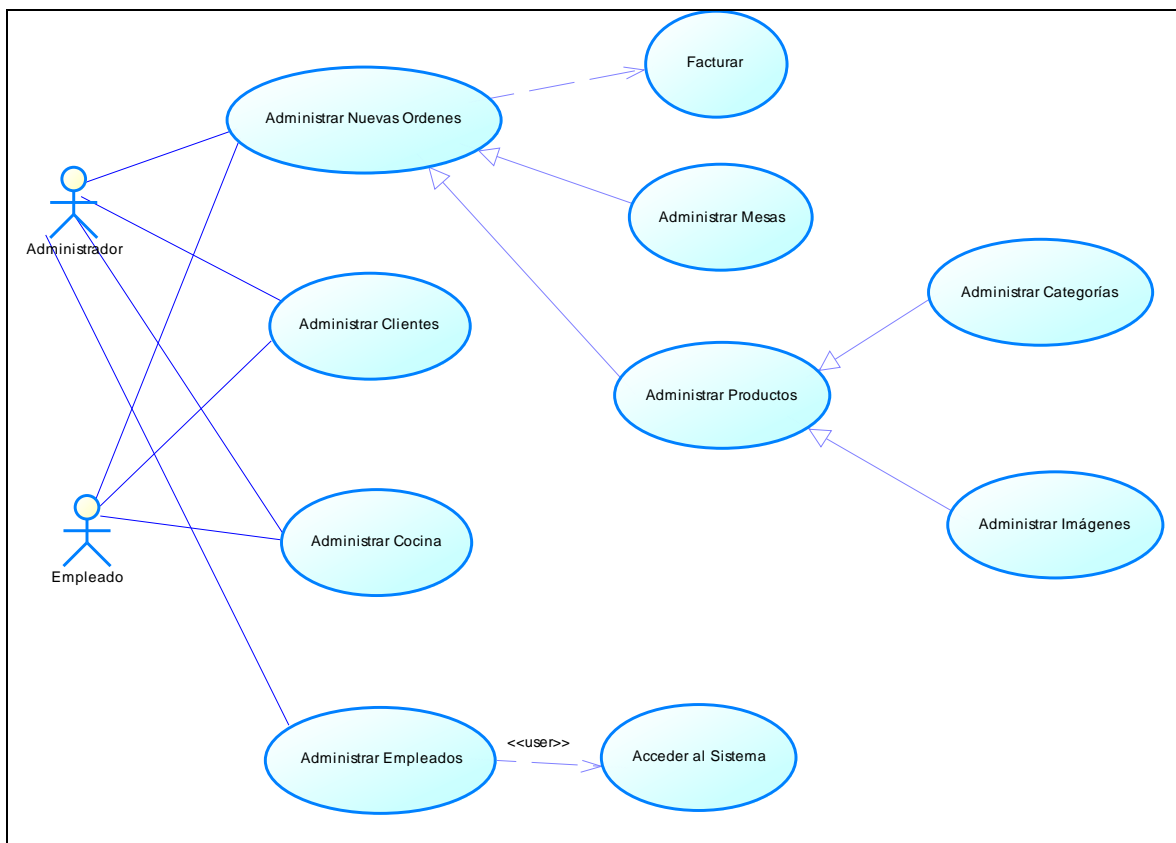


Figura 3 – 08 Casos de uso del prototipo del software

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

- **Casos de Usodel Acceso al Sistema**

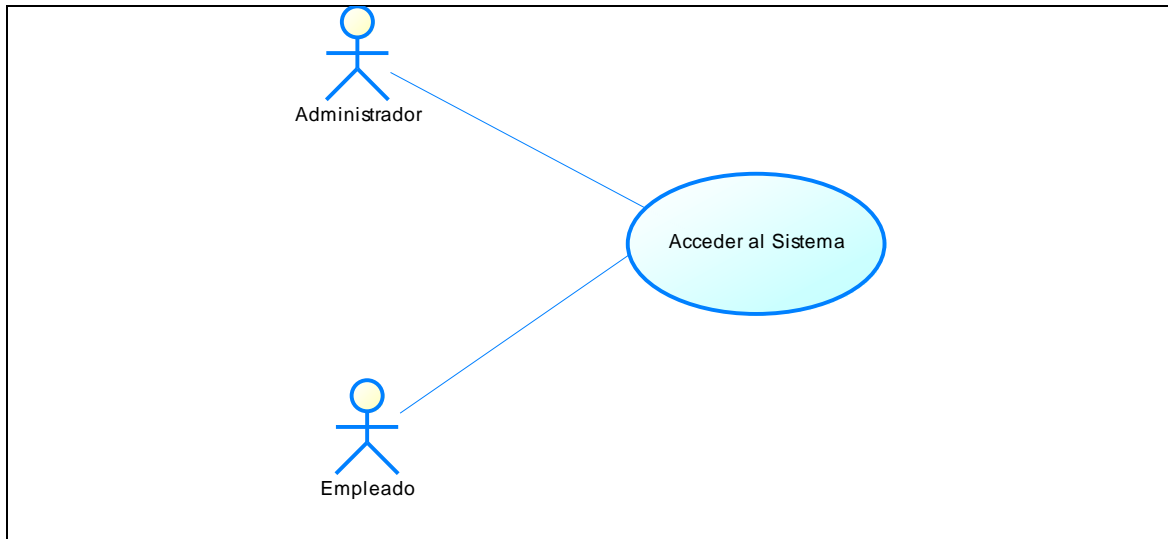


Figura 3 – 09 Casos de uso del Acceso al Sistema

- **Casos de Uso del Módulo de Nuevas Órdenes**

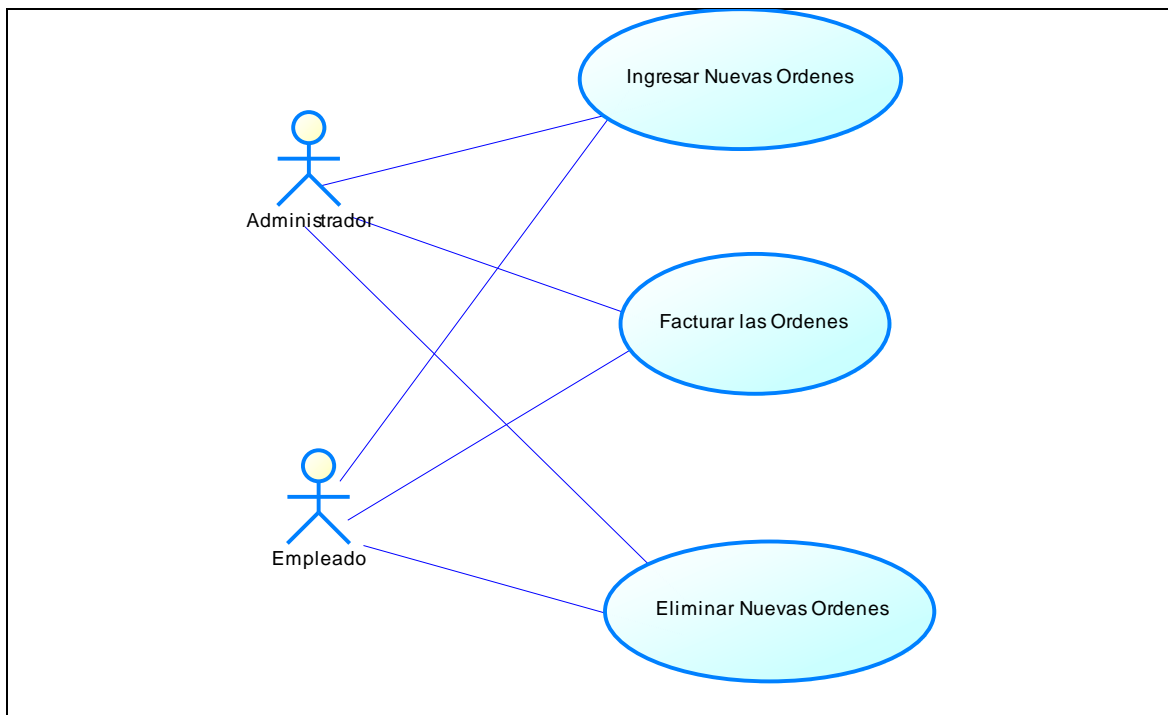


Figura 3 – 10 Casos de uso Módulo de Nuevas Órdenes

# PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

- **Casos de Uso de la Factura**

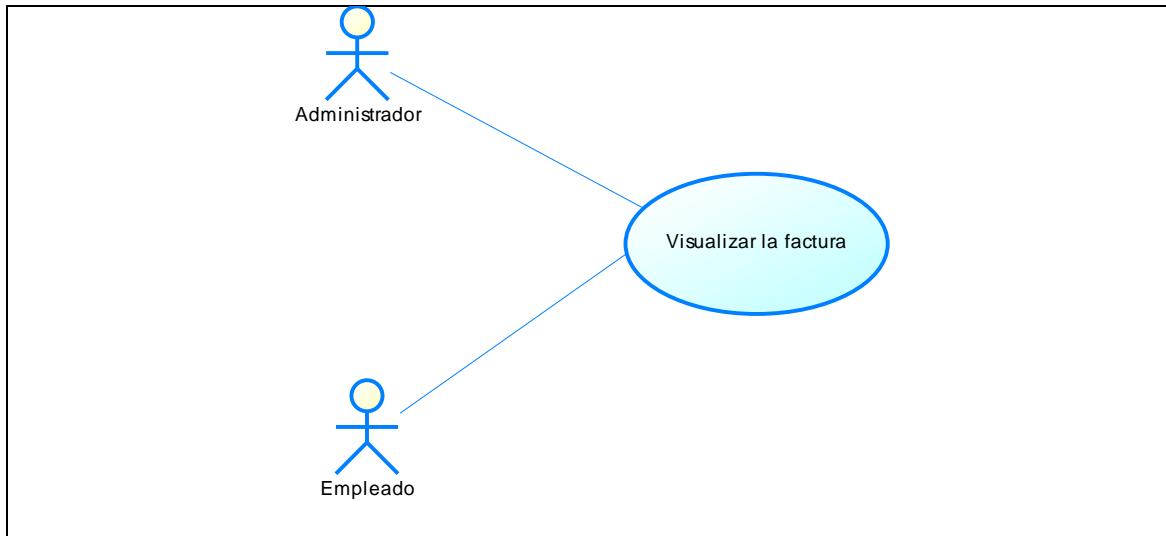


Figura 3 – 11 Casos de uso de la Factura

### 3.6 Diseño Entidad – Relación. Modelo Conceptual

En esta sección mostraremos el modelo conceptual de la base de datos del Prototipo del Software.

El modelo físico y el script de la base de datos se encuentran en el ANEXO C

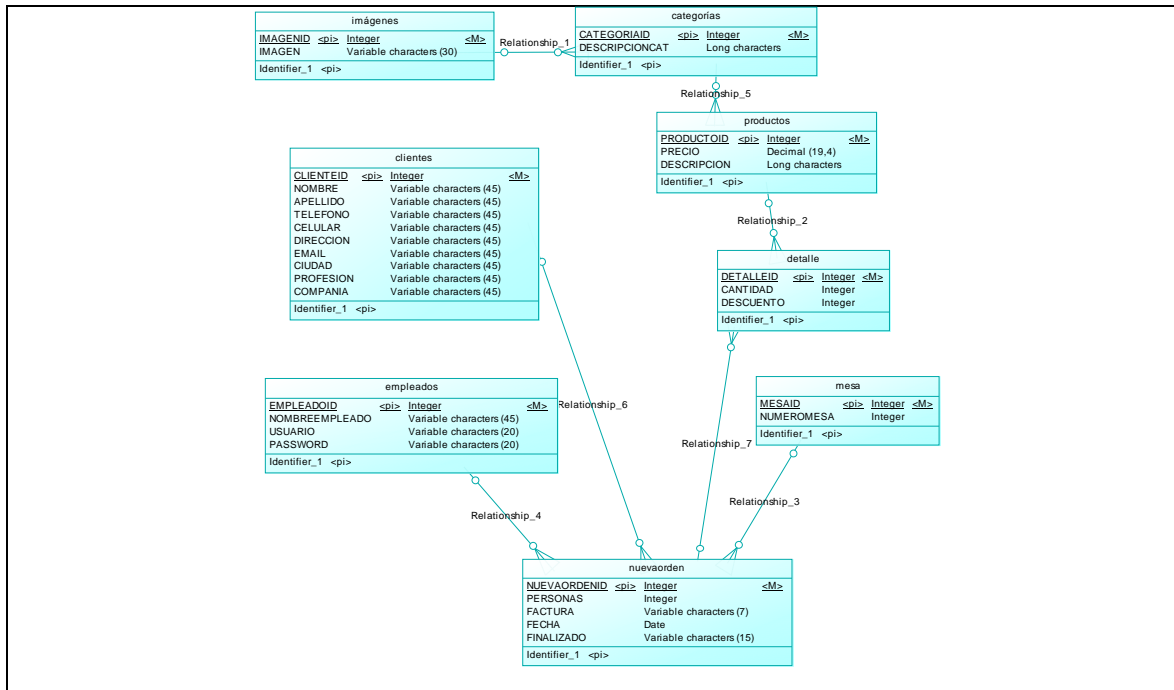


Figura 3 – 12 Modelo Conceptual de la base de datos

# PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

## 3.7 Diseño de reportes

Aquí se muestra el único reporte que tiene este prototipo que es el reporte de la factura.

En el ANEXO D se vuelve a mostrar este mismo reporte con los cambios que el usuario pidió

Factura No :				
NOMBRE :		APELLIDO :		
TELEFONO :				
CELULAR :				
DIRECCION :				
NUMERO DE MESA :		PERSONAS :		
FECHA :				
DESCRIPCION	CANTIDAD	PRECIO	DESCUENTO	TOTAL
SUBTOTAL :				
DESCUENTO :				
IVA :				
TOTAL :				

Figura 3 – 13Diseño del reporte de la Factura

## CAPÍTULO 4

### CODIFICACIÓN, PRUEBAS E IMPLEMENTACIÓN

“En esta sección se detallará los requerimientos funcionales del prototipo al usuario, y este a la vez dice los cambios sobre la marcha que se debe realizar al prototipo quedando los requerimientos del prototipo en varias versiones.” [A]

#### 4.1 Versión Alfa

“Esta es la primera vez que el desarrollador le muestra al usuario el prototipo y éste lo revisa y también son los primeros cambios que el usuario dice al desarrollador para que los haga al prototipo.

Todas las figuras de la versión Alfa, se encuentran en la parte de ANEXOS” [A]

##### 4.1.1 Requerimientos funcionales del prototipo

El prototipo de software al ser presentado por primera vez al usuario cumple con los siguientes requerimientos:

- El Acceso al Sistema está correcto si en la base de datos no está el nombre de Usuario y la Contraseña al poner un Usuario y Contraseña no válidos no se podrá acceder al sistema, y una vez que se accedió al sistema funciona correctamente el Logout que tiene el sistema
- Las operaciones de Insertar, Editar y Eliminar de los módulos de Imágenes, Categorías, Productos, Mesas, Clientes, y Empleados funcionan correctamente.
- Las operaciones de Insertar y Eliminar del módulo de Nueva Orden funciona correctamente.
- La operación de Editar del módulo de Cocina funciona correctamente.
- Particularmente en el módulo de Empleados solo el administrador es el que puede Insertar, Editar y Eliminar Empleados, ningún empleado puede hacer eso solo puede acceder al sistema con el Nombre de Usuario y Contraseña que el administrador le crea.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

### 4.1.2 Requerimientos del usuario

Luego de que el usuario revisa el prototipo por primera vez, éste me dice los cambios que yo debo realizarle al prototipo, los cuáles son los siguientes:

- En el Acceso al Sistema si el usuario pone un Nombre de Usuario y Contraseña que no son válidos entonces debe de salir un mensaje diciendo que ese Nombre de Usuario y Contraseña no es válido.
- En el módulo de Categorías en la ventana de Insertar categorías y Editar validar que el campo de texto Categoría solo se puede ingresar texto.
- En el módulo de Productos en la ventana de Insertar productos y Editar validar que el campo de texto Descripción del producto solo se pueda ingresar texto y en el campo Precio validar que solo se pueda ingresar números positivos.
- En el módulo de Mesas en la ventana de Insertar numero de las mesas y Editar validar que el campo de texto Número de la mesa solo se pueda ingresar números enteros positivos.
- En el módulo de Nueva Orden en la ventana de Insertar Nueva Orden quitar el botón Buscar que está al lado de la lista de Cliente, que en el campo de texto Numero de personas validar que solo se pueda ingresar números enteros positivos, que el botón Cancelar funcione que cuando se dé clic en ese botón nos lleve al archivo nuevaorden.php, cuando se dé clic en el botón Insertar nueva orden nos lleva al archivo factura.php, que este archivo se pase automáticamente y que nos lleve al archivo detalle.php directamente y que no se tenga que dar clic en el botón Insertar productos del archivo factura.php para pasar al archivo detalle.php. En la ventana del archivo detalle.php que el campo de texto No de factura no sea editable, que el campo de texto Descuento validar que solo se pueda ingresar números enteros positivos y que en el campo de texto Cantidad de igual manera validar que solo se pueda ingresar números enteros positivos.

También que al momento de Eliminar una orden no se eliminen por el número de factura sino por cada detalle que tiene la factura.

- En el módulo de Clientes en la ventana de Insertar Clientes y Editar validar que los campos de texto Nombre, Apellido, Ciudad, Profesión y Compañía solo se pueda

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

ingresar texto, que los campos de texto Nombre y Apellido sean obligatorios y que los campos Teléfono y Celular solo se pueda ingresar números enteros positivos.

- En el módulo de Cocina en la ventana de Editar que se cambie de no procesado a despachado por cada detalle no por el número de la factura.
- En el módulo de Empleados desarrollar en el prototipo que los empleados puedan cambiar su Clave no así su Usuario y que no se espere que el administrador sea el único que cambie la Clave de todos los empleados, y que en la ventana de Insertar nuevo empleado y Editar que el campo de texto Clave sea de tipo Contraseña y que el campo de texto Nombre validar que solo se pueda ingresar texto.

A continuación se mostrará el cambio más significativo de los requerimientos del usuario en esta versión:

De la:

Este prototipo muestra una interfaz de usuario para actualizar un empleado. Está organizada en una tabla con cuatro filas y dos columnas. Las primeras tres filas contienen los campos de texto 'Nombre:', 'Usuario:' y 'Clave:'. La cuarta fila contiene dos botones: 'Actualizar empleado' y 'Cancelar'.

Nombre:	<input type="text"/>
Usuario:	<input type="text"/>
Clave:	<input type="text"/>
	<input type="button" value="Actualizar empleado"/> <input type="button" value="Cancelar"/>

Figura 4.1.2 – 1 Interface de la ventana de actualizar empleados por parte del administrador

Se añadió el:

Este prototipo muestra un ajuste de la interfaz de usuario para actualizar un empleado, diseñado para ser utilizado por los empleados. Está organizada en una tabla con tres filas y dos columnas. Las primeras dos filas contienen los campos de texto 'Usuario:' y 'Clave:'. La tercera fila contiene dos botones: 'Actualizar empleado' y 'Cancelar'. El área de entrada de datos está rodeada por una línea punteada.

EMPLEADO:	
Usuario:	<input type="text"/>
Clave:	<input type="text"/>
	<input type="button" value="Actualizar empleado"/> <input type="button" value="Cancelar"/>

Figura 4.1.2 – 2 Ajuste de la Interface de la ventana de actualizar empleados por parte de los empleados

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

### **4.2 Versión Beta**

“Esta es la segunda vez que el desarrollador le muestra al usuario el prototipo y éste lo revisa y también son los cambios que el usuario le dice al desarrollador para que los haga al prototipo una vez que ha revisado nuevamente el prototipo.

Todas las figuras de la versión Beta, se encuentran en la parte de ANEXOS” [A]

#### **4.2.1 Requerimientos funcionales del prototipo**

Aquí se le presenta al usuario los cambios que él mismo pidió que se le haga al prototipo en la Versión Alfa, mediante pruebas a todos los cambios éste verifica que todos los requerimientos que él pidió funcionen.

#### **4.2.2 Requerimientos del usuario**

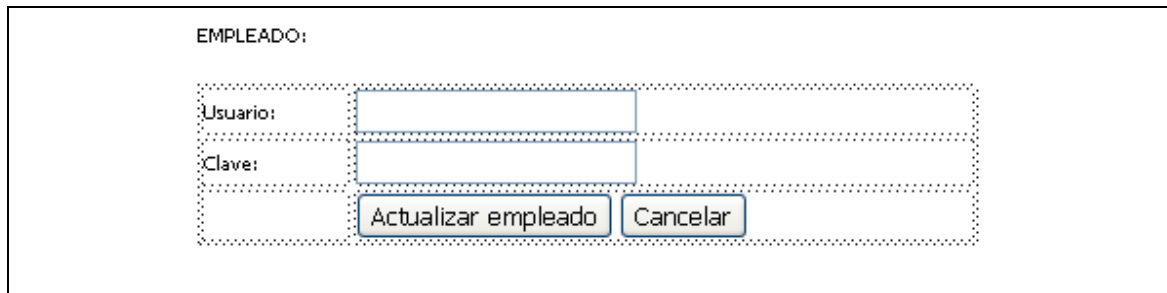
Luego de que el usuario revisa que todos los cambios que el pidió en la Versión Alfa del prototipo funcionen, éste me dice algunos otros cambios que yo debo realizar al prototipo los cuáles son los siguientes:

- En la ventana de actualizar empleado se agregue los campos de Nueva clave y Confirmar clave, esto quiere decir que tanto el administrador o cada usuario al querer cambiar su clave tenga que poner su nueva clave y confirmar su nueva clave.
- En la factura del prototipo antes del número de factura agregar la serie y sub serie del restaurante tipo.
- En el archivo detalle.php agregar la plantilla que está siendo utilizada en todos los demás archivos del prototipo, para mantener un estándar en todos los archivos del prototipo.
- Hacer pruebas de impresión de la factura del prototipo de software.

A continuación se mostrará el cambio más significativo de los requerimientos del usuario en esta versión:

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

De la:

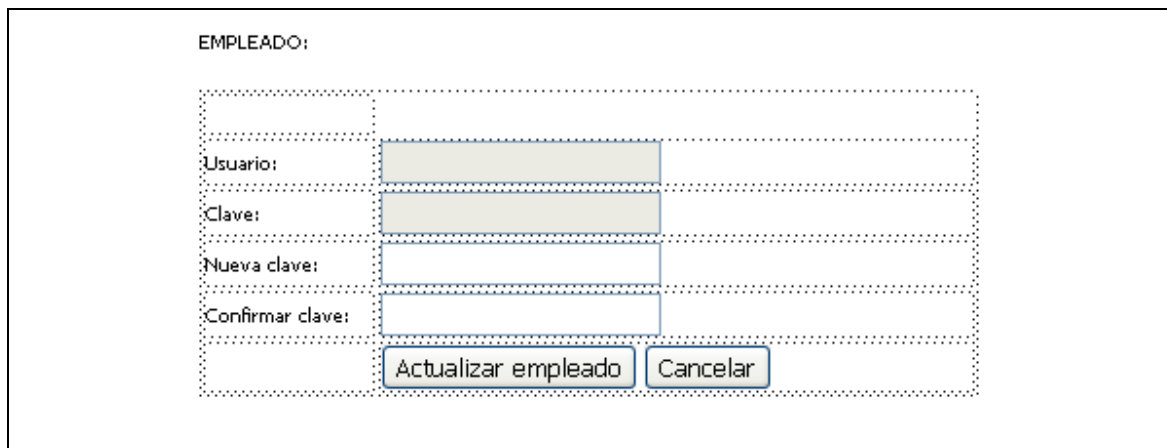


EMPLEADO:

Usuario:	<input type="text"/>
Clave:	<input type="text"/>
	<input type="button" value="Actualizar empleado"/> <input type="button" value="Cancelar"/>

Figura 4.2.2 – 1 Interface de la ventana de actualizar empleados

Se añadió el:



EMPLEADO:

	<input type="text"/>
Usuario:	<input type="text"/>
Clave:	<input type="text"/>
Nueva clave:	<input type="text"/>
Confirmar clave:	<input type="text"/>
	<input type="button" value="Actualizar empleado"/> <input type="button" value="Cancelar"/>

Figura 4.2.2 – 2 Ajuste de la Interface de la ventana de actualizar empleados

### 4.3 Versión Gamma

“Esta es la tercera y última vez que el desarrollador le muestra al usuario el prototipo y éste lo revisa y también son los últimos cambios que el usuario le dice al desarrollador para que los haga al prototipo una vez que ha revisado nuevamente el prototipo.

Todas las figuras de la versión Gamma, se encuentran en la parte de ANEXOS” [A]

#### 4.3.1 Requerimientos funcionales del prototipo

Aquí se le presenta al usuario los cambios que él mismo pidió que se le haga al prototipo en la Versión Beta, mediante pruebas a todos los cambios éste verifica que todos los requerimientos que él pidió funcionen.

# PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

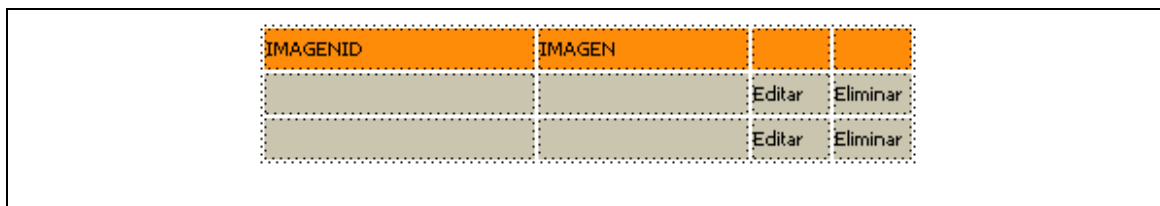
## 4.3.2 Requerimientos del usuario

Luego de que el usuario revisa que todos los cambios que el pidió en la Versión Beta del prototipo funcionen, éste me dice los últimos cambios que yo debo realizar al prototipo, quedando así satisfecho de todos los cambios que se le ha hecho al prototipo, los últimos cambios son los siguientes:

- Que todas las tablas de consultas del prototipo tenga tramas, esto quiere decir que cada fila de la tabla tenga diferente color, utilizando solamente dos colores.
- Poner formato a la factura y si se puede incluir en la factura el logo que tiene la plantilla del prototipo.

A continuación se mostrará el cambio más significativo de los requerimientos del usuario en esta versión:

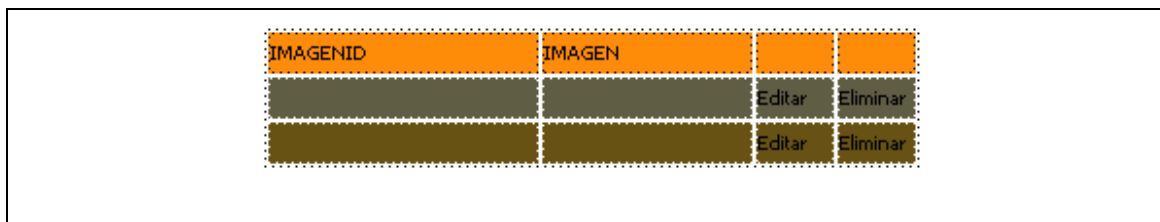
Del:



IMAGENID	IMAGEN		
		Editar	Eliminar
		Editar	Eliminar

Figura 4.3.2 – 1 Tramado de la tabla de consulta del módulo de imágenes

Se realizó un:



IMAGENID	IMAGEN		
		Editar	Eliminar
		Editar	Eliminar

Figura 4.3.2 – 2 Ajuste del Tramado de la tabla de consulta del módulo de imágenes

*Nota: Este tipo de tramado se realizó en todas las tablas de consultas del prototipo*

## CAPÍTULO 5

### CONCLUSIONES Y RECOMENDACIONES

En este capítulo se hará conclusiones acerca de la metodología que se utilizó para desarrollar el prototipo del software, de las interacciones con el usuario, del desarrollar software para web, de hacer prototipos de software y de utilizar herramientas de software libre.

También se hará recomendaciones a la universidad, a la facultad de sistemas, sobre la profesión, de los estudiantes y de desarrollar proyectos en bases de datos.

#### Conclusiones

- La programación extrema es una metodología de desarrollo ligera basada en una serie de valores y una docena de prácticas como entregas continuas, juego de planificación, retroalimentación con el cliente, que propician un aumento en la productividad a la hora de generar software. Se basa en una serie de reglas y principios que se han ido gestando a lo largo de toda la historia de la ingeniería del software, siendo uno de los métodos más representativos dentro de las metodologías llamadas ligeras que lleva al extremo las prácticas para la consecución de sistemas funcionales y que cumplan las características del usuario cliente. Es una nueva forma de ver el desarrollo de software está basada en la simplicidad, la comunicación, la retroalimentación y la refactorización de código.
- Las interacciones con el usuario ocurren en las interfaces de usuario, que tiene el prototipo del software, para su acto comunicativo dónde el usuario le dice al desarrollador si sus necesidades están satisfechas respecto al software o no, el objetivo de las futuras interacciones es mejorar el prototipo del software haciéndole más usable el software y adaptarle a las necesidades de los usuarios. También el usuario puede decir observaciones negativas del software para que el desarrollador las pueda corregir.
- El desarrollo de software para web cada vez es más utilizado porque es más sencillo y rápido crearlo, además que también este software puede ser visto como un sitio web dinámico que contiene varias páginas web, y si no sabemos hacer páginas

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

web o software para web podemos encontrar muchos programas educativos que nos permitirán aprender fácilmente y finalmente, queda la posibilidad de construir el software para web o páginas web en línea mediante el servidor de alojamiento web.

- Desarrollar prototipos de software tienen la ventaja de que son construidos en poco tiempo, usando los programas adecuados y no se debe utilizar mucho dinero pues a partir de que este sea aprobado se puede iniciar el verdadero desarrollo del software, todo prototipo es evaluado por el cliente para una retroalimentación; gracias a ésta se refinan los requisitos del software que se desarrollará, el prototipo se ajusta para satisfacer las necesidades del cliente.
- Las herramientas de software libre en general, nos da la libertad de elección al diseñador, yo decido qué herramientas usar y qué plataformas, tengo la libertad para ejecutar el programa en cualquier sitio, con cualquier propósito y para siempre, libertad para estudiarlo y adaptarlo a mis necesidades, libertad de redistribución, libertad para mejorar el programa y publicar las mejoras y todas estas libertades nos conlleva a tener ventajas técnicas y económicas.
- Los procesos tienen el potencial para mejorar la posición de una compañía en el mercado ya que dan una solución sencilla y rentable en términos económicos para obtener una mejora significativa al sistema como conjunto.

### Recomendaciones

- Para la Universidad pienso que debería haber la opción de clases en línea para que el estudiante si faltó alguna clase pueda igualarse siguiendo la clase en línea, asimismo que el estudiante pueda seguir algún curso que esté ofreciendo alguna facultad, por medio de cursos en línea, pero que estas clases en línea o los cursos en línea estén grabados en la plataforma de la PUCE, para evitar que gente que está ajena a la universidad no pueda acceder a esa información, en otras palabras que solo sirva para los estudiantes y los docentes de la universidad, y finalmente hablando sobre el mismo tema cuento con una biblioteca virtual donde el estudiante pueda realizar solicitudes de artículos en bibliotecas de todo el mundo.
- Para la facultad de Ingeniería en Sistemas, sugiero que las materias de: cálculo, matemáticas, métodos numéricos, ecuaciones diferenciales, física se dicten tan

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

detalladamente y que en cambio otras materias que vamos a utilizar en nuestra vida profesional se dicten más detalladamente cómo son: diseño de computadores, redes, diseño de páginas web, desarrollo de sistemas en web.

- Para la vida profesional de Ingeniería en Sistemas, el estudiante debe estar preparado de manera multi disciplinaria y estar en renovación constante.
- Para los estudiantes de Ingeniería en Sistemas, sugiero investigar, auto aprender, mantenerse al tanto de la tecnología por nuestra propia cuenta. Para crecer en conocimientos y en madurez profesional.
- Al desarrollar proyectos en bases de datos tener en cuenta siempre las relaciones o jerarquías de las tablas de las bases de datos porque si no el proyecto se puede volver en algo muy complejo, además como actualmente las bases de datos se han convertido en una herramienta muy sofisticada para las empresas es necesario conocer muy bien el diseño y arquitectura de éstas, para que en la administración de éstas no tener problemas de ningún tipo y sobre todo con los usuarios.
- Qua al prototipo del software se evalúe constantemente su tiempo de vida o el tiempo de vida de cada elemento, y también evaluar los requerimientos que necesita este prototipo para su instalación y correcto funcionamiento como son: memoria, intercambio y procesador de las computadoras.

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

## GLOSARIO DE TÉRMINOS

### A

**ACTIVIDADES SISTEMATIZADAS:** Acciones interrelacionadas entre sí que, a partir de una o varias entradas de información, materiales o de salidas de otros procesos, dan lugar a una o varias salidas también de materiales (productos) o información con un valor añadido

**ANSI:** por sus siglas en inglés: American National Standards Institute, es una organización sin ánimo de lucro que supervisa el desarrollo de estándares para productos, servicios, procesos y sistemas en los Estados Unidos.

**API:** (del inglés application programming interface) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

### B

**BSD:** Son las iniciales de *Berkeley Software Distribution* (en español, Distribución de Software Berkeley) y se utiliza para identificar un [sistema operativo](#) derivado del sistema [Unix](#) nacido a partir de los aportes realizados a ese sistema por la [Universidad de California](#) en [Berkeley](#).

### C

**CGI:** Es una importante tecnología de la World Wide Web que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web.

**CSS:** Las hojas de estilo en cascada (en inglés Cascading Style Sheets), CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML).

### D

**DOM:** Es esencialmente una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos.

### E

**Eclipse:** Es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

### F

**FOSS:** Es el software libre y de código abierto. Es el software que está licenciado de tal manera que los usuarios pueden estudiar, modificar y mejorar su diseño mediante la disponibilidad de su código fuente.

### G

**Geany:** Es un editor de texto ligero basado en Scintilla con características básicas de Entorno de desarrollo integrado (IDE).

**Glade:** Es una herramienta de desarrollo visual de interfaces gráficas mediante GTK/GNOME.

**GNU:** Es un acrónimo recursivo que significa GNU No es Unix (GNU is Not Unix).

**Gnome:** Es un entorno de escritorio e infraestructura de desarrollo para sistemas operativos Unix y derivados Unix como GNU/Linux, BSD o Solaris; compuesto enteramente de software libre.

**Gnome Office:** Es una suite ofimática desarrollada por el Proyecto libre GNOME.

**GNU Compiler Collection:** Es un conjunto de compiladores creados por el proyecto GNU.

**GNU GPL:** Es una licencia creada por la Free Software Foundation en 1989 (la primera versión), y está orientada principalmente a proteger la libre distribución, modificación y uso de software.

**GNU/Linux:** GNU/Linux es uno de los términos empleados para referirse a la combinación del núcleo o kernel libre similar a Unix denominado Linux, que es usado con herramientas de sistema GNU.

**GTK+ o The GIMP Toolkit:** Es un conjunto de bibliotecas multiplataforma para desarrollar interfaces gráficas de usuario (GUI), principalmente para los entornos gráficos GNOME, XFCE y ROX aunque también se puede usar en el escritorio de Windows, MacOS y otros.

### H

**HTML:** siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web.

**HTTP:** (en español protocolo de transferencia de hipertexto) es el protocolo usado en cada transacción de la World Wide Web.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

### I

**IDE:**Controlador Electrónico Integrado

**ISAM:**son siglas de Indexed Sequential Access Method (Método de Acceso Secuencial Indexado), se trata de un método para almacenar información a la que se pueda acceder rápidamente.

**Ingeniería de software:** Es la disciplina o área de la informática que ofrece métodos y técnicas para desarrollar y mantener software de calidad.

### J

**JUnit:** Es un conjunto de bibliotecas creadas por Erich Gamma y Kent Beck que son utilizadas en programación para hacer pruebas unitarias de aplicaciones Java.

### K

**Kate:** Es un editor de textos para el entorno de escritorio KDE.

**KDE:** Es un proyecto de software libre para la creación de un entorno de escritorio e infraestructura de desarrollo para diversos sistemas operativos como GNU/Linux, Mac OS X, Windows, etc.

**Kent Beck:** Es uno de los creadores de la metodología ágil para el desarrollo de software conocida como programación extrema. También creó, junto con Erich Gamma el framework de pruebas unitarias para Java, JUnit.

**Koffice:** Es una suite ofimática multiplataforma, libre y de código abierto para el proyecto KDE, aunque es independiente de este.

### L

**Lamp:** El acrónimo 'LAMP' se refiere a un conjunto de subsistemas de software necesarios para alcanzar una solución global, en este caso configurar sitios web o Servidores dinámicos con un esfuerzo reducido.

### M

**Mac OS X:**Es un sistema operativo desarrollado y comercializado por Apple Inc. que ha sido incluido en su gama de computadoras Macintosh desde 2002

**MVC:** Es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

**MyISAM:** Es la tecnología de almacenamiento de datos usada por defecto por el sistema administrador de bases de datos relacionales MySQL.

**MySQL AB:** es una compañía de software fundada en 1995, creadora del sistema administrador de bases de datos relacionales MySQL, y una de las más grandes empresas de software libre del mundo.

### P

**.NET:** Es un framework de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones.

**PHP-GTK:** Es una extensión para el lenguaje de programación PHP que permite la utilización de GTK+.

**PHP-Qt:** Es una extensión para PHP5 que permite escribir programas con el Toolkit de Qt.

**Procesos ágiles:** Los procesos ágiles de desarrollo de software, conocidos anteriormente como metodologías livianas, intentan evitar los tortuosos y burocráticos caminos de las metodologías tradicionales enfocándose en la gente y los resultados.

**Pruebas de regresión:** Es a cualquier tipo de pruebas de software que intentan descubrir las causas de nuevos errores (bugs), carencias de funcionalidad, o divergencias funcionales con respecto al comportamiento esperado del software, inducidos por cambios recientemente realizados en partes de la aplicación que anteriormente al citado cambio no eran propensas a este tipo de error.

### Q

**Qt:** Es una biblioteca multiplataforma para desarrollar interfaces gráficas de usuario y también para el desarrollo de programas sin interfaz gráfica como herramientas de la consola y servidores.

### S

**Scalable Vector Graphics (SVG):** Es una especificación para describir gráficos vectoriales bidimensionales, tanto estáticos como animados (estos últimos con ayuda de SMIL), en formato XML.

**Siag office:** Es una suite ofimática fuertemente integrada de software libre para Unix, consiste en al hoja de cálculo SIAG, el procesador de texto PW (Pathetic Writer), el programa de animación Egon, el editor de texto XedPlus, el gestor de archivos Xfiler y el previzualizador Gvu.

## PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB

**SOAP:**(siglas de Simple Object Access Protocol) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.

**Software comercial:** Es el software, libre o no, que es comercializado, es decir, que las compañías que lo producen, cobran dinero por el producto, su distribución o soporte (este dinero casi siempre se emplea para pagar licencias de ciertos programas no libres o para darle un salario a las personas que contribuyeron a crear ese software).

**Software gratuito:** Es un tipo de software de computadora que se distribuye sin costo, disponible para su uso y por tiempo ilimitado.

**SQLite:** Es un sistema de gestión de bases de datos relacional compatible con ACID, y que está contenida en una relativamente pequeña (~275 kiB) biblioteca en C.

### W

**WAMP:** Es el acrónimo usado para describir un sistema de infraestructura de internet que usa las siguientes herramientas:

- Windows, como sistema operativo;
- Apache, como servidor web;
- MySQL, como gestor de bases de datos;
- PHP (generalmente), Perl, o Python, como lenguajes de programación.

**W3C:**El World Wide Web Consortium, abreviado W3C, es un consorcio internacional que produce recomendaciones para la World Wide Web.

### X

**XML:** eXtensible Markup Language. Lenguaje Extensible de Marcado. Lenguaje desarrollado por el W3 Consortium para permitir la descripción de información contenida en el WWW a través de estándares y formatos comunes, de manera que tanto los usuarios de Internet como programas específicos (agentes) puedan buscar, comparar y compartir información en la red. El formato de XML es muy parecido al del HTML aunque no es una extensión ni un componente de éste.

**Xpath:** (XML Path Language) Es un lenguaje que permite construir expresiones que recorren y procesan un documento XML.

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

**REFERENCIA BIBLIOGRÁFICA**

[A]Disertación: PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR PARA RESTAURANTES EN WEB, 210/05, Andrés Hidalgo, PUCE, Quito-Ecuador

[1][http://albertolacalle.com/hci\\_prototipos.htm](http://albertolacalle.com/hci_prototipos.htm) | 2010/03

[2][http://es.wikipedia.org/wiki/Modelo\\_de\\_prototipos](http://es.wikipedia.org/wiki/Modelo_de_prototipos) | 2010/04

[3]<http://html.rincondelvago.com/desarrollo-orientado-a-prototipos.html> | 2010/05

[4]<http://www.monografias.com/trabajos59/elaboracion-prototipos/elaboracion-prototipos2.shtml> | 2010/01

[5][http://www.google.com.ec/search?hl=es&defl=es&q=define:Software+libre&ei=JWTrS-fKAoOKlwfg29nsCg&sa=X&oi=glossary\\_definition&ct=title&ved=0CByQkAE](http://www.google.com.ec/search?hl=es&defl=es&q=define:Software+libre&ei=JWTrS-fKAoOKlwfg29nsCg&sa=X&oi=glossary_definition&ct=title&ved=0CByQkAE) | 2010/05

[6][http://es.wikipedia.org/wiki/Software\\_libre](http://es.wikipedia.org/wiki/Software_libre) | 2010/05

[7][http://es.wikipedia.org/wiki/C%C3%B3digo\\_abierto](http://es.wikipedia.org/wiki/C%C3%B3digo_abierto) | 2010/05

[8]<http://www.hipertext.net/web/pag258.htm> | 2010/05

[9]<http://www.eaprende.com/plantilla.php?id=28> | 2010/05

[10]<http://es.wikipedia.org/wiki/PHP> | 2010/05

[11]<http://www.dipler.org/tag/php/> | 210/06

[12]<http://es.wikipedia.org/wiki/MySQL> | 2010/05

[13][http://es.wikipedia.org/wiki/Servidor\\_HTTP\\_Apache](http://es.wikipedia.org/wiki/Servidor_HTTP_Apache) | 2010/05

[14]<http://es.kioskea.net/contents/genie-logiciel/cycle-de-vie.php3> | 2010/05

[15]<http://es.wikipedia.org/wiki/Software> | 2010/06

[16][http://es.wikipedia.org/wiki/Programaci%C3%B3n\\_extrema](http://es.wikipedia.org/wiki/Programaci%C3%B3n_extrema) | 2010/07

[17] <http://definicion.de/proceso/> | 2011/01

[18] <http://www.definicionabc.com/general/proceso.php> | 2011/01

[19] [http://www.galapagospark.org/institucion\\_estructura\\_directorio.html](http://www.galapagospark.org/institucion_estructura_directorio.html) | 2009/06

PROTOTIPO FUNCIONAL ADMINISTRATIVO DE UN SOFTWARE ESTÁNDAR  
PARA RESTAURANTES EN WEB

[20] [http://imaginar.org/iicd/index\\_archivos/TUS3/tramitologia.pdf](http://imaginar.org/iicd/index_archivos/TUS3/tramitologia.pdf) | 2005/03