

# UNIVERSIDAD CATÓLICA DE SANTA MARÍA

FACULTAD DE CIENCIAS E INGENIERÍAS FÍSICAS Y FORMALES

CARRERA PROFESIONAL DE INGENIERÍA DE SISTEMAS



**TITULO:**

**“Desarrollo de Herramientas de E-marketing e  
Incremento de las Ventas en las Microempresas,  
Arequipa 2012, Caso: Panadería Carmencita”**

**PRESENTADO POR EL BACHILLER:**

**FREDY ANTONIO AREVALO DELGADO**

**AREQUIPA - PERÚ**

**2013**

## **AGRADECIMIENTO**

*En primer lugar a Dios por haberme guiado por el camino de la felicidad hasta ahora; en segundo lugar a cada uno de los que son parte de mi familia: a mi PADRE, mi MADRE y mi HERMANA, pilares fundamentales en mi vida, sin ellos, jamás hubiese podido conseguir lo que soy hasta ahora. Su tenacidad y lucha insaciable han hecho de ellos el gran ejemplo a seguir y destacar por siempre haberme dado su fuerza y apoyo incondicional*

*Por último a mis asesores de tesis porque son ellos quienes me apoyaron en todo momento para la culminación con éxito de este proyecto de Tesis.*

## ÍNDICE

### CAPITULO 1

#### PLANTEAMIENTO TEÓRICO

|                                                   |   |
|---------------------------------------------------|---|
| 1.1.INTRODUCCIÓN .....                            | 1 |
| 1.2. PLANTEAMIENTO DEL PROBLEMA .....             | 4 |
| 1.3. ANTECEDENTES .....                           | 5 |
| 1.4. LA EMPRESA.....                              | 6 |
| 1.5. PREGUNTA GENERADORA DE LA INVESTIGACIÓN..... | 6 |
| 1.6. JUSTIFICACIÓN .....                          | 7 |
| 1.7. OBJETIVOS                                    |   |
| 1.7.1. OBJETIVO GENERAL.....                      | 8 |
| 1.7.2. OBJETIVOS ESPECÍFICOS.....                 | 8 |
| 1.8. ALCANCE .....                                | 9 |

### CAPITULO II

#### MARCO TEÓRICO

|                                                                                       |    |
|---------------------------------------------------------------------------------------|----|
| 2.1. COMERCIO ELECTRÓNICO Y APLICACIONES WEB.....                                     | 10 |
| 2.1.1. COMERCIO ELECTRÓNICO.....                                                      | 10 |
| 2.1.2. APLICACIONES WEB.....                                                          | 16 |
| 2.2. CARACTERÍSTICAS VENTAJAS Y DESVENTAJAS DE APACHE,PHP, JAVA<br>SCRIPT Y XML ..... | 22 |
| 2.2.1. APACHE.....                                                                    | 22 |
| 2.2.2. PHP .....                                                                      | 28 |
| 2.2.3. JAVASCRIPT.....                                                                | 32 |
| 2.3. BASE DE DATOS CON MySQL .....                                                    | 38 |

|                                                               |    |
|---------------------------------------------------------------|----|
| 2.4. METODOLOGÍAS Y TECNOLOGÍAS PARA EL DESARROLLO DE         |    |
| APLICACIONES WEB (rias's) .....                               | 42 |
| 2.4.1. AJAX .....                                             | 42 |
| 2.4.2. MOZILLA PRISM.....                                     | 47 |
| 2.4.3. ADOBE AIR.....                                         | 48 |
| 2.4.4. JAVAFX .....                                           | 50 |
| 2.5. INDICADORES DE CALIDAD DEL SOFTWARE .....                | 52 |
| 2.6. EVALUACIÓN DE LOS FACTORES DE CALIDAD DEL SOFTWARE ..... | 58 |
| 2.6.1. MODELOS DE CALIDAD DEL SOFTWARE .....                  | 60 |
| 2.6.2. ESTÁNDARES DE CALIDAD DEL SOFTWARE A NIVEL PRODUCTO .  | 72 |

### **CAPITULO III**

### **INGENIERÍA DEL PROYECTO**

|                                                                 |     |
|-----------------------------------------------------------------|-----|
| 3.1. JUSTIFICACIÓN DE LA HERRAMIENTAS ELEGIDAS .....            | 82  |
| 3.1.1. HERRAMIENTAS PARA EL FRONT-END .....                     | 82  |
| 3.1.2. HERRAMIENTAS PARA EL BACK-END.....                       | 85  |
| 3.2. POSIBLES MÓDULOS DE APLICACIÓN.....                        | 92  |
| 3.3. REQUISITOS .....                                           | 93  |
| 3.4. INTERACTIVIDAD, CAPACIDAD DE RESPUESTA EN APLICACIONES WEB | 98  |
| 3.5. INCREMENTO DE VENTAS .....                                 | 99  |
| 3.6. DIAGRAMA DE CASO USO .....                                 | 101 |
| 3.7. DIAGRAMA DE SECUENCIA.....                                 | 102 |

## CAPITULO IV

### DISEÑO Y CONSTRUCCIÓN DE LA APLICACIÓN

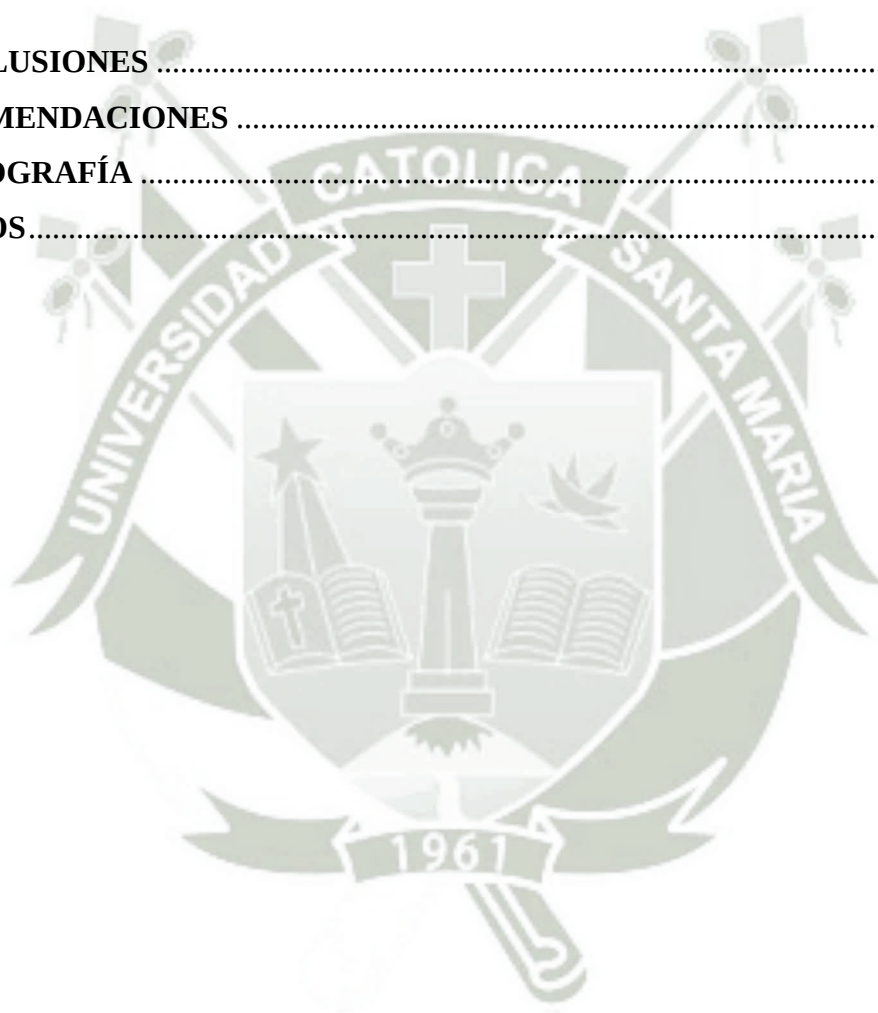
|                                                            |     |
|------------------------------------------------------------|-----|
| 4.1. OPERACIÓN DEL NEGOCIO DE PANIFICACIÓN .....           | 103 |
| 4.2. LOGÍSTICA DE LA EMPRESA .....                         | 106 |
| 4.3. SOFTWARE Y HARDWARE PARA SOPORTAR LA APLICACIÓN ..... | 107 |
| 4.4. METODOLOGIA DE IMPLEMENTACION .....                   | 113 |
| 4.5. MODELO DE E-MARKETING ELECTRONICO.....                | 117 |
| 4.6.DISEÑO Y CONSTRUCCION DEL BACK END.....                | 122 |
| 4.7. INSTALACION Y CONFIGURACION DEL SITIO WEB.....        | 143 |
| 4.8. DISEÑO Y CONSTRUCCION DEL FRONT-END .....             | 150 |
| 4.8.1. DISEÑO DEL FRONT-END .....                          | 150 |
| 4.8.2. CONSTRUCCION DEL FRONT-END .....                    | 155 |
| 4.8.3. TIENDA .....                                        | 156 |
| 4.8.4. GALERIA DE IMÁGENES .....                           | 167 |
| 4.8.5. DISEÑO Y COTIZACION EN LINEA .....                  | 168 |
| 4.8.6. INFORMACION DE LA PYME.....                         | 169 |
| 4.8.7. INTERFAZ DE LA INFORMACION .....                    | 171 |
| 4.9. PRUEBAS E INTEGRACION.....                            | 174 |

## CAPITULO V

### EVALUACIÓN DE FACTORES DE CALIDAD

|                                                                |     |
|----------------------------------------------------------------|-----|
| 5.1. ESTABLECIMIENTO DE LOS REQUERIMIENTOS DE EVALUACIÓN ..... | 177 |
| 5.1.1. ESTABLECIMIENTO DEL PROPÓSITO DE LA EVALUACIÓN .....    | 177 |
| 5.1.2. IDENTIFICACION DEL PRODUCTO A EVALUAR .....             | 178 |
| 5.1.3. ESPECIFICACION DEL MODELO DE CALIDAD .....              | 178 |
| 5.2. ESPECIFICACION DE LA EVALUACION .....                     | 182 |
| 5.2.1. METRICAS PARA LA EVALUACION .....                       | 182 |

|                                                              |         |
|--------------------------------------------------------------|---------|
| 5.2.2. ESTABLECIMIENTO DE CRITERIOS PARA LA VALORACION ..... | 183     |
| 5.3.DISEÑO Y PROCEDIMIENTO DEL PLAN DE EVALUACION .....      | 184     |
| 5.4. EJECUCION DE LA EVALUACION .....                        | 185     |
| 5.4.1. EJECUCIÓN DEL INSTRUMENTO .....                       | 185     |
| 5.4.2. RESULTADOS DE LA EVALUACIÓN .....                     | 188     |
| 5.4.3. INTERPRETACIÓN DE RESULTADOS .....                    | 191     |
| <br><b>CONCLUSIONES</b> .....                                | <br>193 |
| <b>RECOMENDACIONES</b> .....                                 | 194     |
| <b>BIBLIOGRAFÍA</b> .....                                    | 195     |
| <b>ANEXOS</b> .....                                          | 197     |



## RESUMEN

La presente investigación ubicada dentro de la modalidad de proyecto factible tiene como propósito el diseño y creación de un sitio Web que contenga las herramientas de E-Marketing y comercio electrónico para el incremento de las ventas en la microempresa del rubro de la panificación denominada Panadería “Carmencita” ubicado en el distrito de Miraflores de la ciudad de Arequipa. La página Web dinámica ha sido diseñada de manera interactiva, desarrollando el contenido, a través de hipervínculos, lo que permite desplazarse por el contenido del mismo. En su elaboración se utilizó el programa Macromedia Dream Weaver Css5, como editor para el diseño de la página Web, también se emplearon complementos jquery y scripts los cuales se utilizaron para realizar la animaciones del sitio, otras herramientas como Corel draw 5 photoshop y Paint para trabajar las imágenes y Java Álbum para la galería de imágenes. Como servidor web se eligió el APPSERV por ser una herramienta web que incluye Apache, PHP y MySQL. Finalmente este sitio Web será una herramienta importante para la empresa Panadería “Carmencita” dedicada al rubro alimenticio al emprender proyectos y estrategias de oportunidad que incorporen las tecnologías emergentes de tal manera que se optimicen los procesos y así poder potenciar su capacidad y promover aumentos porcentuales en las ventas que le ayuden a superar parte de la problemática que presenta.

## CAPITULO 1

### PLANTEAMIENTO TEÓRICO

#### 1.1. INTRODUCCIÓN

El incremento de la tecnología y la automatización ha tenido una incursión muy significativa en nuestro país en el último decenio, haciendo de las nuevas tecnologías un medio de ventas eficiente y masivo para las empresas. La llegada del Internet y la creciente evolución en cuanto a los procesos de marketing comercial, han creado nuevas necesidades y requerimientos en los procesos de negocio.

La informática y su revolución, el Internet y las telecomunicaciones han mostrado que las empresas pequeñas y medianas representaban enormes ventajas frente a las grandes, especialmente por procesos de constante crecimiento, flexibilidad y facilidad de adaptación a condiciones y entornos cambiantes, esto debido a la propia dinámica de la renovación tecnológica de la información en la sociedad actual.

Dentro de este escenario los gobiernos, tanto de economías desarrolladas como en desarrollo, han enfocado sus esfuerzos hacia la creación de los ambientes adecuados y a la remoción de obstáculos para el desarrollo de las pequeñas y medianas empresas.

Las Pequeñas y Micro Empresas producen el 49% del PBI nacional y emplean al 88% de la PEA del sector privado, generan el 80% del empleo del país y tienen presencia en los diferentes sectores productivos importantes datos que representan un núcleo de oportunidades de desarrollo.

La pequeñas y microempresas en el Perú tienen importantes repercusiones económicas y sociales para el proceso de crecimiento económico y desarrollo nacional del país, por ello hoy en día todas las PYMES competitivamente hablando cuidan aspectos como publicidad, marketing, calidad de servicio y costos accesibles; estos aspectos no pueden

ser descuidados en ninguna empresa ya que continuamente se enfrentan a un mundo más globalizado, donde la probabilidad de incremento de las ventas por consecuencia de las utilidades sean posibles para ellos que continuamente se mantengan actualizados en todo lo referente a las NNTT nuevas tecnologías de la información .

Las tecnologías de la información y comunicación TIC'S como muchas otras herramientas, son esenciales en el ámbito empresarial actual, el uso de las TIC's permite de manera eficiente el habilitar las estrategias del negocio y facilitar la evaluación del desempeño de cualquier organización, sin importar su tamaño. Este enfoque competitivo implica que las empresas lo requieran de manera prioritaria para estar al menos en igualdad de condiciones que sus competidores. Por eso, las áreas de las TIC's están siendo obligadas cada día más a transformarse, de tal manera que los servicios que brindan aporten valor al negocio, ofreciendo servicios de calidad a un bajo costo.

El empleo de las tecnologías de información por las pequeñas y medianas empresas ha sido ampliamente debatido bajo diversos enfoques, incluyendo la concepción de una brecha entre grandes y pequeñas empresas, producto de la capacidad (económica y técnica) de acceso a las tecnologías de información.

En el ámbito comercial Internet ha significado y ofrecido un aumento en la producción, las empresas se han dado cuenta que pueden llegar a conocer mejor los perfiles y necesidades de sus consumidores, reducir tiempo gastado en las transacciones, habiendo una mayor y más eficiente comunicación, logrando que los clientes actuales y potenciales puedan tener información de última hora de la empresa.

Es así como Internet se ha vuelto herramienta imprescindible para que las empresas tengan mayor presencia frente a sus competidores.

Por tanto hoy en día la tendencia en la comercialización de productos es mediante e-commerce, no solo existe la necesidad de tener que atraer a los clientes hasta los locales de venta para llegar a una mayor cantidad de clientes, permite también un ahorro de la mayor parte de los costos fijos, así como el alquiler de la tienda o el pago de la nómina

de empleados; además de que a través de las herramientas del Internet se puede recibir pedidos las 24 horas del día los 7 días a la semana. Con lo cual potenciales compradores pueden escoger los productos desde sus hogares, sin la asistencia o presión de un vendedor.

A través de los catálogos en línea y el empleo de las tarjetas de crédito la venta en línea tomó un mayor impulso, los catálogos en línea son uno de los mayores productos de las páginas Web dinámicas, ya que en una página estática puede haber contenido flash con muy buenos diseño pero eso no garantiza que tenga un flujo continuo de visitantes o clientes satisfechos que pueden dejar de visitar el sitio por ausencia de actualizaciones o contenidos personalizados, ésta es una de las mayores ventajas de las páginas Web dinámicas, que interactúan con el usuario, recibiendo la información que realmente busca y necesita.

Por lo dicho en la anteriormente este proyecto surge para la empresa de panadería “Carmencita” la necesidad de incursionar en un nuevo campo de negocio, por ello se pensó en el diseño y desarrollo de un sistema herramientas Web para potenciar la gestión de los procesos de ventas, marketing y posicionamiento.

Panadería “Carmencita” es una microempresa perteneciente al sector de panificación; se dedica a la elaboración de productos del rubro de la panificación. Cuenta con clientes en dentro de la ciudad, una organización con una visión de crecimiento tecnológico con el fin de prestar un mejor servicio a sus clientes considerados lo más importante.

Por esta razón en vista de ampliar las ventas, el posicionamiento, conocimiento de la marca y ofrecer un mejor servicio a los clientes, la empresa demanda un sistema que gestione, promueva y automatice de manera efectiva los procesos de mercadeo y comercio que se realizan dentro y fuera del área de ventas, también se requiere que los clientes tengan siempre a la mano la información del estado de sus pedidos, así como información pertinente sobre los productos que ofrece la empresa, con el fin de poder

generar incremento en las ventas y mantener un canal de comunicación con sus clientes y empleados.

De esta manera, Panadería “Carmencita” pretende mejorar sus procesos de negocio con el diseño y desarrollo de un sistema de información que emplee herramientas Web para la gestión de los procesos de venta, con el fin de prestar un mejor servicio a sus clientes, además de expandir sus servicios a clientes potenciales existentes en el internet y redes de comunicaciones.

## 1.2. PLANTEAMIENTO DEL PROBLEMA

En la actualidad todas las empresas, cualquiera que sea su rubro, tamaño y sector de actividad, deben mostrarse competitivas en un entorno global, disponer necesariamente de la tecnología y herramientas de Marketing y comercio de la información más moderna y saber aplicarla de forma eficaz y rentable.

Con el constante y acelerado avance tecnológico y las nuevas tendencias de compartir y utilizar eficientemente la información, el mundo global ha sentido la necesidad de comunicar todos sus recursos dentro y fuera de sus organizaciones, haciendo imperioso el uso de una gran herramienta tecnológica como es el internet, especialmente la llamadas redes de comunicaciones, las cuales traen consigo profundos cambios en el desarrollo de todos los ámbitos de la humanidad, haciendo que la distancia y el tiempo dejen de ser un obstáculo para realizar sus actividades.

Desde el punto de vista del comercio electrónico, la efectividad en el manejo de la información es sin lugar a duda un factor decisivo en el desempeño de las micro y medianas empresas, por lo tanto, debe contar con una plataforma tecnológica estratégica integrada que respalde la ejecución de su operación y procesos comerciales en forma metodizada y eficaz de manera que brinde la información necesaria para mejorar la productividad y rentabilidad para obtener así una ventaja competitiva.

Debido a la situación descrita se hace necesario para la empresa Panadería “Carmencita” dedicada al rubro alimenticio, emprender proyectos y estrategias de oportunidad que incorporen las tecnologías emergentes de tal manera que se optimicen

los procesos y así poder potenciar su capacidad promover aumentos porcentuales en las ventas que le ayuden a superar parte de la problemática que presentan. Entre las herramientas que debe utilizar se encuentran: El internet, comercio electrónico, E-commerce, E –marketing, Chats, foros y redes sociales, intranet e extranet disponible a través de un portal Web que permita su interacción con los usuarios, clientes, proveedores y público en general.

### **1.3. ANTECEDENTES**

Para efectos de apoyar la presente investigación, intercambiar ideas y lograr de esta manera base sólida sobre el Diseño e Implantación de Portales Web, se han consultado diferentes autores que realizaron sitios Web con diversos fines, que van desde brindar información hasta facilitar y aumentarla economía y el comercio a escala global , a continuación se mencionan los que se consideran de mayor trascendencia.

Brugnoli, (2003) presento un Trabajo Especial de Grado titulado “Desarrollo del Portal Electrónico de la Universidad del Valle”. Se propuso como objetivo implementar un portal Web para la Universidad del Valle. Para ello utilizó la tecnología Servidor de Paginas Activas (ASP). Para conseguir este producto, se utilizó una nueva metodología dentro de la Universidad, el Método de Diseño de Hipermedios de Comunicación Orientado (OOHDM), desarrollada en 1998 por Schwabe Daniel y Rossi Gustavo, la cual es ideal para la concepción de proyectos que interactúan en ambientes hipermedia (Ambientes que presentan diversos elementos de la información que se muestra en la pantalla y están vinculados con otras informaciones que pueden ser de otras fuentes, se acceden haciendo clic sobre ellos), como en este caso. Esta metodología guía al autor del proyecto a través de una serie de pasos por cada una de las tareas a realizar durante la realización del mismo.

### **1.4. LA EMPRESA**

En el presente trabajo de investigación se toma como objeto de estudio a la Empresa Panadería “Carmencita” dedicada al rubro de la panificación denominada así durante el

desarrollo del trabajo. El objeto social de la empresa es la producción de alimentos de panificación, para el público y cuenta con experiencia a nivel local.

Se analiza que existe una pérdida significativa de su participación en el mercado del rubro de la panificación ocasionada por un desarrollo débil en la especialidad de relaciones públicas y de explotar las ventajas competitivas en la producción y de los servicios al momento de presentarse a los consumidores, clientes y público en general. Dado el macro entorno y la evolución registrada por el micro entorno relacionado al comercio electrónico que han ido cambiado y avanzado muy rápidamente a la par de la tecnología, se ve necesaria la participación la empresa de Panadería “Carmencita” de forma que generen más facilidad de acceso para los clientes a los cuáles se les presta servicio y vende productos, se busca promover altos niveles de confiabilidad y mejoramientos en la cadena de valor por medio de herramientas tecnológicas ya utilizadas por las diferentes empresas a nivel global.

### **1.5. PREGUNTA GENERADORA DE LA INVESTIGACIÓN**

Tomando en cuenta lo visto anteriormente claramente se observa que la panadería “Carmencita” requiere la implementación de una estrategia que le permita establecer ventajas competitivas sostenibles de acuerdo a su entorno globalizado donde, como se observa el uso de las aplicaciones orientadas a tecnologías web, establecen un punto de quiebre desde que las empresas más conocidas del mercado utilizan estrategias de e-commerce basadas en Internet, que para el caso de nuestra empresa en cuestión, significa actualmente el desarrollo del concepto de e-marketing como herramienta que potencia la cadena de valor y establece claros puntos de diferenciación frente a los competidores inmediatos.

Se plantea una serie de interrogantes cuya respuesta permita implementar estrategias de acción y operación de la mejor forma posible:

¿Cuál es el empleo de herramientas de e-marketing y el incremento de las ventas en las microempresas, Arequipa 2012, Caso: Panadería “Carmencita”?

¿Cuál es el modelo de e-marketing a emplear para una microempresa, caso “Panadería Carmencita”?

¿De qué manera las herramientas de e-marketing, incrementan las ventas en la microempresa Panadería “Carmencita”?

¿Qué elementos deben considerarse para el desarrollo de aplicaciones basadas en técnicas de programación web para la empresa panadería “Carmencita”?

¿Esta nueva estrategia brinda a la empresa el acceso a nuevos nichos de mercado donde pueda crear incremento del valor agregado y su diferenciación?

¿Una estrategia de mercadeo fundamentada en el uso de herramientas de e-commerce incremento de ventas permite a la empresa de panadería “Carmencita” desarrollar una ventaja competitiva sostenible?

## 1.6. JUSTIFICACIÓN

Se considera que la presente investigación es válida pues resulta muy importante establecer el empleo de herramientas de e-marketing y el incremento de las ventas en las microempresas, Arequipa 2012, Caso: Panadería Carmencita. De allí también la conveniencia de la presente investigación y que ésta sirve, pues incrementa el conocimiento sobre este importante tema.

La relevancia social o humana se da porque el mayor conocimiento sobre esta materia, contribuirá a establecer soluciones para resolver las bajas ventas (aprovechamiento de la oportunidad) que en muchas oportunidades presentan las microempresas, lo que impide su crecimiento empresarial y/o determina su pronta desaparición.

En relación a la relevancia metodológica, la investigación aplicará instrumentos para la obtención de datos en el trabajo de campo. La investigación tiene también relevancia científica pues significa un aporte, es decir, un nuevo conocimiento, al establecer

metodológicamente el empleo de herramientas de e-marketing y el incremento de las ventas en las microempresas. Asimismo, tiene también relevancia contemporánea, pues el e-marketing es un tema de actualidad y de repercusión social.

Finalmente se dirá que el interés del investigador es genuino y la curiosidad por el problema elegido es real, lo que garantiza la culminación de la investigación; a lo que se agrega que ésta es factible realizarla y además es original.

## **1.7. OBJETIVOS**

### **1.7.1. OBJETIVO GENERAL**

- Desarrollar una herramienta de E-marketing para el incremento de las ventas en las microempresas, caso “Panadería Carmencita”. Arequipa 2012,

### **1.7.2. OBJETIVOS ESPECÍFICOS.**

- Construir una página web dinámica para la microempresa “Panadería Carmencita” de la ciudad de Arequipa.
- Desarrollar un modelo E-marketing electrónico de óptimo funcionamiento en el internet para la empresa arequipeña: Caso “Panadería Carmencita”.
- Utilizar las herramientas de ingeniería de software en la implementación del sitio web que permita incrementar las ventas en la microempresa de Panadería “Carmencita”.

## **1.8. ALCANCE**

Se pretende lograr un servicio adecuado para el cliente o usuario satisfaciendo sus necesidades incluyendo la difusión de la totalidad de los servicios y productos en la página web. Iniciando con un desarrollo adecuado para el proceso al que se pretende lograr en un determinado tiempo.

La creación de una página dará a conocer todos los productos que desea vender la Panadería “Carmencita”. Ofreciendo todos los niveles, una amplia gama de servicios para la página, desarrollando un mejor sistema para la promoción del negocio y para el cliente sea de su agrado tanto en el producto, negoció y página.

Dentro del radio de alcances de la creación y operación del sitio web este incluye aspectos del proyecto, como en encontrar oportunidades de promover el sitio web con las empresas afiliadas, las organizaciones profesionales, emisión o medios de impresos, visitantes y clientes locales, agencias de información y la reubicación de servicios, proveedores de acceso a Internet, y la ciudad local o directorio de sitios de la ciudad. identificar los requisitos posibles y equilibrar las demandas concurrentes de calidad, tiempo y costo para adaptar las especificaciones de los planes y del enfoque de diversas inquietudes expectativas.



## CAPITULO II

### MARCO TEÓRICO

#### 2.1 COMERCIO ELECTRÓNICO Y APLICACIONES WEB

##### 2.1.1 COMERCIO ELECTRÓNICO

El comercio, es la actividad ancestral desarrollada por el ser humano, ha evolucionado de muchas maneras, pero su significado y su fin siempre es el mismo.

El Comercio es "el proceso y los mecanismos utilizados, necesarios para colocar las mercancías, que son elaboradas en las unidades de producción, en los centros de consumo en donde se aprovisionan los consumidores, último eslabón de la cadena de comercialización. Es comunicación y trato".

El comercio electrónico se entiende como cualquier forma de transacción comercial en la cual las partes involucradas interactúan de manera electrónica y no de la manera tradicional por medio de intercambios físicos o trato físico directo.

En la actualidad la forma de comercio se caracteriza por el mejoramiento constante en los procesos de abastecimiento, y como respuesta a ello los negocios a nivel mundial están cambiando tanto su organización como sus operaciones. El comercio electrónico es el medio de llevar a cabo dichos cambios dentro de una escala global, permitiendo a las empresas ser más eficientes y flexibles en sus operaciones internas, para así trabajar de una manera más cercana con sus proveedores y estar más pendiente de las necesidades y expectativas de sus clientes. Además permiten seleccionar a los mejores proveedores sin importar su localización geográfica para que de esa forma se pueda vender a un mercado global.

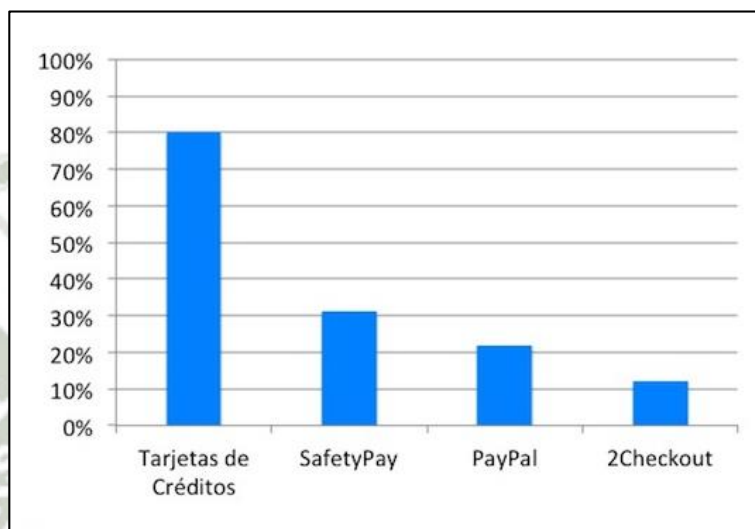
Jaime Neilson nos dice que: "El comercio electrónico es cualquier actividad de intercambio comercial en la que las órdenes de compra - venta y pagos se realizan a través de un medio electrónico, los cuales incluyen servicios financieros y bancarios suministrados por Internet. El comercio electrónico es la venta a distancia aprovechando las grandes ventajas que proporcionan las nuevas tecnologías de la información, como la ampliación de la oferta, la interactividad y la inmediatez de la compra, con la particularidad que se puede comprar y vender a quién se quiera, y, dónde y cuándo se quiera. Es toda forma de transacción comercial o intercambio de información, mediante el uso de Nueva Tecnología de Comunicación entre empresas, consumidores y administración corporativa."

En el Perú Futuro Labs presentó en junio del 2011 el I Estudio de Comercio Electrónico en el Perú con el objetivo de, por vez primera, evaluar los usos de los páginas peruanas que cuentan con tiendas electrónicas y medios de pago online. En detalle, este estudio refleja el panorama del Comercio Electrónico en el Perú, así como identifica los principales rubros y tipos de comercio del país. También se analizó el uso de las redes sociales como difusión de la oferta comercial y qué medios son los más usados por las tiendas electrónicas peruanas. El estudio de Comercio Electrónico comprendió el análisis de 400 sitios web ,dentro de 21 sectores empresariales. Como principales resultados resaltan:

- El 95% de las empresas de comercio electrónico son de venta directa al consumidor (B2C).
- El 57% de las empresas con tiendas electrónicas también tienen presencia física (Brick and Mortar).
- El 80% de los negocios acepta tarjeta de crédito, mientras que el 31% toma en cuenta los pagos mediante SafetyPay. Sólo el 22% usa PayPal y el 12% utiliza 2Checkout.

- El 64% de tiendas electrónicas tiene presencia en redes sociales, de las cuáles la más usada es Facebook con 63%.

**FIG. 2.1. EMPRESAS SEGÚN LOS PRINCIPALES MEDIOS DE PAGO DISPONIBLES**



Fuente: Estudio de Comercio Electrónico en Perú-2011

### **TIPOS DE TRANSACCIONES DE COMERCIO ELECTRÓNICO**

- "Business to business" (entre empresas): las empresas pueden intervenir como compradoras o vendedoras, o como proveedoras de herramientas o servicios de soporte para el comercio electrónico, instituciones financieras, proveedores de servicios de Internet, etc.
- "Business to consumers" (Entre empresa y consumidor): as empresas venden sus productos y prestan sus servicios a través de un sitio Web a clientes que los utilizarán para uso particular.
- "Consumers to consumers" (Entre consumidor y consumidor): es factible que los consumidores realicen operaciones entre sí, tal es el caso de los remates en línea.

- "Consumers to administrations" (Entre consumidor y administración): los ciudadanos pueden interactuar con las Administraciones Tributarias a efectos de realizar la presentación de las declaraciones juradas y/o el pago de los tributos, obtener asistencia informativa y otros servicios.
- "Business to administrations" (Entre empresa y administración): las administraciones públicas actúan como agentes reguladores y promotores del comercio electrónico y como usuarias del mismo.

## **VENTAJAS DEL COMERCIO ELECTRÓNICO**

### **Para las Empresas**

- Reducción de costo real al hacer estudio de mercado.
- Desaparecen los límites geográficos y de tiempo.
- Disponibilidad las 24 horas del día, 7 días a la semana, todo el año.
- Reducción de un 50% en costos de la puesta en marcha del comercio electrónico, en comparación con el comercio tradicional.
- Hacer más sencilla la labor de los negocios con sus clientes.
- Reducción considerable de inventarios.
- Agilizar las operaciones del negocio.
- Proporcionar nuevos medios para encontrar y servir a clientes.
- Incorporar internacionalmente estrategias nuevas de relaciones entre clientes y proveedores.
- Reducir el tamaño del personal de la fuerza.
- Menos inversión en los presupuestos publicitarios.
- Reducción de precios por el bajo coste del uso de Internet en comparación con otros medios de promoción, lo cual implica mayor competitividad.

- Cercanía a los clientes y mayor interactividad y personalización de la oferta.
- Desarrollo de ventas electrónicas.
- Globalización y acceso a mercados potenciales de millones de clientes.
- Implantar tácticas en la venta de productos para crear fidelidad en los clientes.

#### **Para los clientes**

- Abarata costos y precios
- Da poder al consumidor de elegir en un mercado global acorde a sus necesidades
- Un medio que da poder al consumidor de elegir en un mercado global acorde a sus necesidades.
- Brinda información pre-venta y posible prueba del producto antes de la compra.
- Inmediatez al realizar los pedidos.
- Servicio pre y post-venta on-line.
- Reducción de la cadena de distribución, lo que le permite adquirir un producto a un mejor precio.
- Mayor interactividad y personalización de la demanda.
- Información inmediata sobre cualquier producto, y disponibilidad de acceder a la información en el momento que así lo requiera.
- Permite el acceso a más información.

#### **DESVENTAJAS DEL COMERCIO ELECTRÓNICO**

- **Desconocimiento de la empresa.** No conocer la empresa que vende es un riesgo del comercio electrónico, ya que ésta puede estar en otro país o en el mismo, pero en muchos casos las "empresas" o "personas-empresa" que ofrecen sus productos o servicios por Internet ni siquiera están constituidas legalmente en su país y no se trata más que de gente que esta "probando suerte en Internet".
- **Forma de Pago.** Aunque ha avanzado mucho el comercio electrónico, todavía no hay una transmisión de datos segura el 100%. Y esto es un problema pues nadie quiere dar sus datos de la Tarjeta de Crédito por Internet. De todos modos se ha de decir que ha mejorado mucho.
- **Intangibilidad.** Mirar, tocar, hurgar. Aunque esto no sea sinónimo de compra, siempre ayuda a realizar una compra.
- **El idioma.** A veces las páginas web que visitamos están en otro idioma distinto al nuestro; a veces, los avances tecnológicos permiten traducir una página a nuestra lengua materna. Con lo cual podríamos decir que éste es un factor "casi resuelto". (Hay que añadir que las traducciones que se obtienen no son excelentes ni mucho menos, pero por lo menos nos ayudan a entender de que nos están hablando o que nos pretenden vender).
- **Conocer quien vende.** Ya sea una persona o conocer de que empresa se trata. En definitiva saber quien es, como es, etc. Simplemente es una forma inconsciente de tener mas confianza hacia esa empresa o persona y los productos que vende.
- **Poder volver (post y pre-venta).** Con todo ello podemos reclamar en caso de ser necesario o pedir un servicio "post-venta". Al conocerlo sabemos donde poder ir. El cliente espera recibir una atención "pre-venta" o "post-venta".

- **Privacidad y seguridad.** La mayoría de los usuarios no confía en el Web como canal de pago. En la actualidad, las compras se realizan utilizando el número de la tarjeta de crédito, pero aún no es seguro introducirlo en Internet sin conocimiento alguno. Cualquiera que transfiera datos de una tarjeta de crédito mediante Internet, no puede estar seguro de la identidad del vendedor. Análogamente, éste no lo está sobre la del comprador. Quien paga no puede asegurarse de que su número de tarjeta de crédito no sea recogido y sea utilizado para algún propósito malicioso; por otra parte, el vendedor no puede asegurar que el dueño de la tarjeta de crédito rechace la adquisición. Resulta irónico que ya existan y funcionen correctamente los sistemas de pago electrónico para las grandes operaciones comerciales, mientras que los problemas se centren en las operaciones pequeñas, que son mucho más frecuentes.

## 2.1.2 APLICACIONES WEB

Se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador. Con el surgimiento de las aplicaciones Web el crecimiento de comercio electrónico ha superado las expectativas.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs, tiendas en línea y la propia Wikipedia que son ejemplos bien conocidos de aplicaciones web.

Es importante mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

## **ANTECEDENTES**

En los primeros tiempos de la computación cliente-servidor, cada aplicación tenía su propio programa cliente que servía como interfaz de usuario que tenía que ser instalado por separado en cada ordenador personal de cada usuario. El cliente realizaba peticiones a otro programa -el servidor- que le daba respuesta. Una mejora en el servidor, como parte de la aplicación, requería normalmente una mejora de los clientes instalados en cada ordenador personal, añadiendo un coste de soporte técnico y disminuyendo la productividad.

A diferencia de lo anterior, las aplicaciones web generan dinámicamente una serie de páginas en un formato estándar, como HTML o XHTML, soportados por los navegadores web comunes. Se utilizan lenguajes interpretados en el lado del cliente, directamente o a través de plugins tales como JavaScript, Java, Flash, etc., para añadir elementos dinámicos a la interfaz de usuario. Generalmente cada página web en particular se envía al cliente como un documento estático, pero la secuencia de páginas ofrece al usuario una experiencia interactiva. Durante la sesión, el navegador web interpreta y muestra en pantalla las páginas, actuando como cliente para cualquier aplicación web.

## **INTERFAZ**

Las interfaces web tienen ciertas limitaciones en las funcionalidades que se ofrecen al usuario. Hay funcionalidades comunes en las aplicaciones de escritorio como dibujar en la pantalla o arrastrar-y-soltar que no están soportadas por las tecnologías web estándar. Los desarrolladores web generalmente utilizan lenguajes interpretados(scripts) en el lado del cliente para añadir más funcionalidades, especialmente para ofrecer una experiencia interactiva que no requiera recargar la página cada vez (lo que suele resultar molesto a los usuarios). Recientemente se han desarrollado tecnologías para coordinar estos lenguajes con las tecnologías en el lado del servidor. Como ejemplo, AJAX es una técnica de desarrollo web que usa una combinación de varias tecnologías.

### **CONSIDERACIONES TÉCNICAS**

Una ventaja significativa es que las aplicaciones web deberían funcionar igual independientemente de la versión del sistema operativo instalado en el cliente. En vez de crear clientes para Windows, Mac OS X, GNU/Linux y otros sistemas operativos, la aplicación web se escribe una vez y se ejecuta igual en todas partes. Sin embargo, hay aplicaciones inconsistentes escritas con HTML, CSS, DOM y otras especificaciones estándar para navegadores web que pueden causar problemas en el desarrollo y soporte de estas aplicaciones, principalmente debido a la falta de adhesión de los navegadores a dichos estándares web (especialmente versiones de Internet Explorer anteriores a la 7.0). Adicionalmente, la posibilidad de los usuarios de personalizar muchas de las características de la interfaz (tamaño y color de fuentes, tipos de fuentes, inhabilitar Javascript) puede interferir con la consistencia de la aplicación web.

Otra aproximación es utilizar Adobe Flash Player o Java applets para desarrollar parte o toda la interfaz de usuario. Como casi todos los navegadores incluyen soporte para estas tecnologías (usualmente por medio de plug-ins), las aplicaciones basadas en Flash o Java pueden ser implementadas con aproximadamente la misma facilidad. Dado que ignoran las configuraciones de los navegadores, estas tecnologías permiten más control sobre la interfaz, aunque

las incompatibilidades entre implementaciones Flash o Java puedan crear nuevas complicaciones, debido a que no son estándares. Por las similitudes con una arquitectura cliente-servidor, con un cliente "no ligero", existen discrepancias sobre el hecho de llamar a estos sistemas “aplicaciones web”; un término alternativo es “Aplicación Enriquecida de Internet”.

### **Estructura de las aplicaciones web**

Aunque existen muchas variaciones posibles, una aplicación web está normalmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador web ofrece la primera capa, y un motor capaz de usar alguna tecnología web dinámica (ejemplo: PHP, Java Servlets o ASP,ASP.NET, CGI, ColdFusion, embPerl, Python (programming language) o Ruby on Rails) que constituye la capa intermedia. Por último, una base de datos constituye la tercera y última capa.

El navegador web manda peticiones a la capa intermedia que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario.

### **USO EMPRESARIAL**

Una estrategia que está emergiendo para las empresas proveedoras de software consiste en proveer acceso vía web al software. Para aplicaciones previamente distribuidas, como las aplicaciones de escritorio, se puede optar por desarrollar una aplicación totalmente nueva o simplemente por adaptar la aplicación para ser usada con una interfaz web. Estos últimos programas permiten al usuario pagar una cuota mensual o anual para usar la aplicación, sin necesidad de instalarla en el ordenador del usuario. A esta estrategia de uso se la denomina Software como servicio y a las compañías desarrolladoras se les denomina Proveedores de Aplicaciones de Servicio(ASP por sus siglas en inglés), un modelo de negocio que está atrayendo la atención de la industria del software.

### **Ventajas**

- Ahorra tiempo: Se pueden realizar tareas sencillas sin necesidad de descargar ni instalar ningún programa.
- No hay problemas de compatibilidad: Basta tener un navegador actualizado para poder utilizarlas.
- No ocupan espacio en nuestro disco duro.
- Actualizaciones inmediatas: Como el software lo gestiona el propio desarrollador, cuando nos conectamos estamos usando siempre la última versión que haya lanzado.
- Consumo de recursos bajo: Dado que toda (o gran parte) de la aplicación no se encuentra en nuestro ordenador, muchas de las tareas que realiza el software no consumen recursos nuestros porque se realizan desde otro ordenador.
- Multiplataforma: Se pueden usar desde cualquier sistema operativo porque sólo es necesario tener un navegador.
- Portables: Es independiente del ordenador donde se utilice (un PC de sobremesa, un portátil...) porque se accede a través de una página web (sólo es necesario disponer de acceso a Internet). La reciente tendencia al acceso a las aplicaciones web a través de teléfonos móviles requiere sin embargo un diseño específico de los ficheros CSS para no dificultar el acceso de estos usuarios.
- La disponibilidad suele ser alta porque el servicio se ofrece desde múltiples localizaciones para asegurar la continuidad del mismo.
- Los virus no dañan los datos porque éstos están guardados en el servidor de la aplicación.
- Colaboración: Gracias a que el acceso al servicio se realiza desde una única ubicación es sencillo el acceso y compartición de datos por parte de varios usuarios. Tiene mucho sentido, por ejemplo, en aplicaciones online de calendarios u oficina.
- Los navegadores ofrecen cada vez más y mejores funcionalidades para crear aplicaciones web ricas (RIAs).

### **Inconvenientes**

- Habitualmente ofrecen menos funcionalidades que las aplicaciones de escritorio. Se debe a que las funcionalidades que se pueden realizar desde un navegador son más limitadas que las que se pueden realizar desde el sistema operativo. Pero cada vez los navegadores están más preparados para mejorar en este aspecto. La aparición de HTML 5 representa un hito en este sentido. Es posible añadir funcionalidades a estas aplicaciones gracias al uso de Aplicaciones de Internet Enriquecidas.
- La disponibilidad depende de un tercero, el proveedor de la conexión a internet o el que provee el enlace entre el servidor de la aplicación y el cliente. Así que la disponibilidad del servicio está supeditada al proveedor. <sup>1</sup>

### **Diferencia entre aplicación web y aplicación de internet enriquecida (RIA)**

Las aplicaciones web se ejecutan nativamente desde el navegador. Pero existen algunas aplicaciones que funcionan desde el navegador pero además requieren la instalación de un software en el ordenador para poder utilizarse. Estas aplicaciones se denominan Aplicaciones de Internet Ricas. El motivo de usar este software adicional es que hay muchas funcionalidades que los navegadores no pueden ofrecer, y él enriquece a las aplicaciones web ofreciendo dichas funcionalidades. .

Ejemplos de funcionalidades que pueden ofrecer los programas online gracias al uso de software instalado:

- Procesamiento de imágenes
- Captura de imágenes
- Uso de webcam / Captura de video<sup>1</sup>

### **LENGUAJES DE PROGRAMACIÓN**

Existen numerosos lenguajes de programación empleados para el desarrollo de aplicaciones web en el servidor, entre los que destacan:

- PHP
- Java, con sus tecnologías Java Servlets y JavaServer Pages (JSP)

- Javascript
- Perl
- Ruby
- Python

También son muy utilizados otros lenguajes o arquitecturas que no son propiamente lenguajes de programación, como HTML, XML o ASP/ASP.NET. Se utilizan para servir los datos adecuados a las necesidades del usuario, en función de cómo hayan sido definidos por el dueño de la aplicación. Los datos se almacenan en alguna base de datos estándar.

## 2.2 CARACTERÍSTICAS VENTAJAS Y DESVENTAJAS DE APACHE,PHP, JAVA SCRIPT y HTML

### 2.2.1 APACHE

Una de las herramientas imprescindibles para el desarrollo de una aplicación Web es un servidor Web, antes es necesario comprender antes algunas cuestiones relacionadas.

**Servidor.** Un servidor técnicamente es un sistema que proporciona recursos, este sistema es conocido como una computadora que pertenece a una red y que suministra información a otras computadoras, las cuales ejecutan los programas clientes. Dependiendo el tipo de recursos que suministra es la clasificación del tipo de servidor, por ejemplo, servidor de archivos, de impresión, de correo, de telefonía, de acceso remoto, de bases de datos, o el que por ahora es de interés es el servidor Web.

**Servidor Web.** El servidor Web es un programa cuya función es atender y responder a las diferentes peticiones de los navegadores, proveyendo de los recursos que se soliciten usando el protocolo http (Hipertext Transfer Protocol, )

o el https (Hypertext Transfer Protocol Secure), que es la versión cifrada y autenticada. Este protocolo pertenece a la capa de aplicación del modelo OSI (Open System Interconnection, Modelo de referencia de interconexión de sistemas abiertos) (Figura 2.2.1.1) y que está diseñado para transferir lo que llamamos hipertextos, páginas Web o páginas HTML (Hypertext Markup Language): Textos complejos con enlaces, imágenes, formularios, botones y objetos incrustados como animaciones o reproductores de música.

Este programa llamado servidor generalmente se instala en una computadora conectada en forma permanente en Internet, de esta manera podrá atender las solicitudes de páginas Web y tareas de procesos de datos que los clientes pudieran solicitar en cualquier momento.

Con esto existe la ventaja de la disponibilidad de la información desde cualquier punto de la red y además de que es posible almacenar aplicaciones en el servidor, permitiendo que el cliente no requiera de elementos adicionales para ejecutar la aplicación.

### **Servidor web apache**

Apache es un servidor web cuya historia inicia con los orígenes del Internet. En el 2009 se consolida como el primer servidor web utilizado en más de 100 millones de sitios web. Es desarrollado y mantenido por toda una comunidad libre de desarrolladores que son auspiciados por la Fundación de Software Apache. Está disponible para un gran número de sistemas operativos, sin embargo los que le ofrecen un soporte más amplio son la familia de sistemas operativos Linux. El servidor Apache forma parte de la combinación APPSERV que incluye Apache + MySQL + PHP. Este combo de tecnologías es uno de los más utilizados en la actualidad para las aplicaciones web debido en parte a que el servidor Apache se encuentra disponible en un gran número de servicios de hosting. De esta forma Apache se convierte en una opción sumamente robusta y estable que presenta un gran nivel de integración con PHP.

En la actualidad es el principal servidor de web. Es el más rápido, eficiente y el que evoluciona a mayor velocidad. Y Apache, por su naturaleza de software abierto, es ideal para instalar en máquinas diversos sistemas operativos, que aseguran funcionamiento y operación excelentes. Apache + Mysql + php es una combinación que se está utilizando en el mundo empresarial, Apache ha ayudado a que el campo se amplíe de forma muy sólida en el mundo Internet, creando una Internet-box que difícilmente puede ser superada por otra plataforma en los sistemas actuales, tanto en coste como en potencia.

### **Características**

**Código abierto.** Todo el código fuente de Apache está escrito en C, Apache es una tecnología gratuita de código fuente abierto (*open source*) es el término con el que se conoce al software distribuido y desarrollado libremente, es decir, existe transparencia en la forma de su desarrollo. El API de Apache es de dominio público por lo que pueden realizarse mejoras al código además de agregarse nuevos módulos al servidor, así como una capacidad de adaptación a necesidades particulares. Existe una gran comunidad de desarrolladores a nivel mundial que desarrollan y lo mejoran gracias a esta realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos, logrando actualizaciones constantes sin necesidad de esperar a que un grupo reducido de desarrolladores resuelva un fallo sino que puede ser resuelta por un grupo más completo. Esto permite producir binarios para plataformas en las que no hay un binario oficial o compilarlo para un mejor rendimiento en una plataforma específica.

**Es un servidor basado en procesos**, utilizando la técnica pre-fork (consiste en la creación previa de un grupo de procesos y su mantenimiento hasta que sea necesaria su utilización).

**Arquitectura modular.** Arquitectura en la que existe una sección *core* o base y diversos módulos multiproceso y módulos adicionales que aportan mucha de la funcionalidad que podría considerarse básica para un servidor Web, es decir, gracias a esto se puede adaptar a diferentes entornos y necesidades con los diferentes módulos de apoyo que proporciona y con la API de programación de módulos para el desarrollo de módulos específicos. Cualquiera que posea una experiencia decente en la programación de C o Perl puede escribir un módulo para realizar una función determinada.

**Administración.** Es uno de los elementos principales de cualquier servidor. Apache cuenta con un conjunto de archivos de configuración ampliamente documentados, estos se encuentran en formato ASCII y pueden ser editados con cualquier editor de texto. Si se necesita realizar alguna actualización resulta sencillo respaldar la configuración del servidor. Se puede administrar de forma remota ya que puede operarse y configurarse desde la línea de comandos, así como por medio de interfaces gráficas y un conjunto de herramientas desarrolladas con este propósito.

**Extensible:** Gracias a que es modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.

**Personalizable,** la arquitectura modular de Apache permite construir un servidor hecho a la medida. Además permite la implementación de los últimos y más nuevos protocolos.

**Configuración de errores.** Cuenta con archivos de configuración específicos que hacen posible configurar el servidor para que ejecute un determinado script cuando ocurra un error en concreto, es decir, Apache permite la creación de *logs* de errores configurados por el administrador. Tiene una alta configuración en la creación y gestión de *logs* que contienen enormes cantidades de información, necesitando una herramienta que realice resúmenes de los datos para facilitar el

análisis del contenido, de este modo se puede tener un mayor control sobre lo que sucede en el servidor.

## VENTAJAS

***Eficiencia y estabilidad.*** La eficiencia de Apache es lograda gracias a que se encuentra optimizado para consumir la menor cantidad de recursos del sistema en comparación a otros servidores, gracias a esto puede lograr un gran desempeño convirtiéndolo en una aplicación estable y robusta, soportando un mayor número de transacciones.

***Multiplataforma.*** Apache es soportado por la gran mayoría de plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementan el protocolo HTTP/1.1 y la noción de sitio virtual, es por tanto una aplicación sumamente portable y configurable en diferentes ambientes lo que lo hacen prácticamente universal.

***Extensibilidad.*** Apache es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables a éste. Desde módulos que permiten generar contenido dinámico (con PHP, Java, Perl, Python), monitoreo del rendimiento, comunicaciones seguras por SSL, creación de servidores virtuales por IP. Otra cuestión importante es que cualquiera que posea una experiencia decente en la programación de C o Perl puede escribir un módulo para realizar una función determinada.

***Seguridad.*** La mayoría de las vulnerabilidades de la seguridad descubiertas y resueltas puede en la mayoría de los casos ser abusada solamente por los usuarios locales y no puede ser accionada remotamente. Cuenta con Módulos de autenticación: mod\_access, mod\_auth y mod\_digest, así como Soporte para SSL y TLS. También permite autenticación de base de datos basada en SGBD.

*Soporte para los lenguajes.* Apache trabaja con gran cantidad de lenguajes como Perl, Python, tcl, PHP y otros lenguajes de script. Perl destaca en el mundo del script y Apache utiliza su parte de Perl tanto con soporte CGI como con soporte mod perl. También trabaja con Java y páginas jsp. Teniendo todo el soporte que se necesita para tener páginas dinámicas.

*Personalización de los mensajes de error.* Apache permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto y negociación de contenido de esta manera se puede tener un mayor control sobre lo que sucede en el servidor.

### **Desventajas**

*Administración compleja.* Como la mayoría de los programas open-source, uno depende de configurar los archivos a mano o tener que instalarse herramientas adicionales para las tareas de administración. Fue criticado por la falta de una interfaz gráfica que ayude en su configuración ya que resultar difícil de configurar incluso para tareas sencillas y por ello algunos usuarios con poca experiencia tratan de evadir su uso ya que una configuración errónea puede hacer que no funcione adecuadamente el servidor, es necesario contar con la documentación necesaria respecto a los archivos de configuración, esto para poder ajustar correctamente algunos parámetros del servidor.

*Falta de integración.* Al ser un producto multiplataforma, el servidor no aprovecha al máximo las posibilidades que ofrece el sistema operativo.

*Formatos de configuración no estándar.* Esto dificulta un poco la automatización y el procesamiento de la configuración al no estar basada en formatos más soportados como el XML.

*Lentitud de arranque y ejecución.* El servidor es aproximadamente un 20% más lento en su arranque debido a la sobrecarga que la resolución representa para el

cargador (loader) y es aproximadamente un 5% más lento en su ejecución bajo algunas plataformas.

### 2.2.2 PHP

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas dinámicas. Se usa principalmente para la interpretación del lado del servidor 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.

Se ejecuta en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. El lenguaje PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores.

PHP puede ejecutar un conjunto de operaciones con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, operar con bases de datos, o mandar y recibir cookies.

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta.

Aunque todo en su diseño está orientado a facilitar la creación de sitios web, es posible crear aplicaciones con una interfaz gráfica para el usuario. 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. El resultado es enviado por el intérprete al servidor, quien a su vez lo envía al cliente.

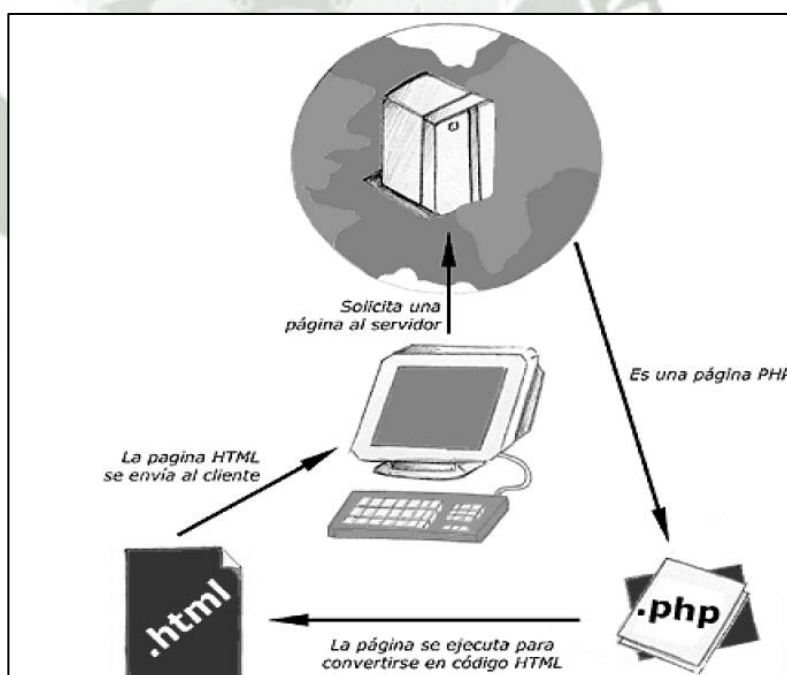
#### Características de PHP

Algunas de las características del lenguaje PHP son:

- Multiplataforma: PHP funciona tanto en sistemas Unix o Linux con servidor web Apache así como en sistemas Windows con Microsoft Internet

Information Server, de forma que el código generado por cualquiera de éstas plataformas no debe ser modificado al pasar a la otra plataforma. Sin embargo, las mejores prestaciones del lenguaje se obtienen trabajando en un entorno Unix o Linux con servidor web Apache. La principal ventaja que aporta el uso conjunto de PHP y Apache es que, de esta manera, se puede compilar el intérprete PHP como un módulo de Apache, consiguiendo así que la velocidad de ejecución de una página PHP sea elevada y que el consumo de recursos sea bajo ya que el intérprete PHP se carga una sola vez en memoria.

- **Ejecución en Servidor:** Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente. El cliente solamente recibe una página con el código HTML resultante de la ejecución de la página PHP.



Fuente: THOMSON –LAURA. 2009.

En la Figura 2.2. Se muestra el esquema del funcionamiento de PHP

Una página PHP no es más que un programa escrito en PHP que genera código HTML. Cuando un navegador solicita al servidor web una página PHP, antes de enviar dicha página al cliente, se la pasa al intérprete del mismo, éste la interpreta y es el resultado de esta interpretación del programa, contenido en la página PHP, lo que termina llegando al cliente.

- Es un lenguaje basado en herramientas con licencia de software libre.
- Sintaxis cómoda. Cuenta con una sintaxis similar a la de C, C++ o Perl. Lo más destacado ocurre a nivel semántico: el tipado es muy poco estricto. Es decir, cuando creamos una variable no tenemos que indicar de qué tipo es, pudiendo guardar en ella datos de cualquier tipo. Esto es muy flexible y cómodo para el desarrollador, aunque los errores que se cometen pueden ser graves y difíciles de corregir al reducirse las posibilidades del intérprete para detectar incompatibilidades entre variables.
- Soporta objetos y herencia. Tiene soporte para la programación orientada a objetos, es decir, es posible crear clases para la construcción de objetos, con sus constructores. Además soporta herencia, aunque no múltiple.
- Cuenta con una extensa librería de funciones que facilitan enormemente el trabajo de los desarrolladores.
- Compatibilidad con bases de datos
- Expansión. PHP Está alcanzando unos niveles de uso tan elevados que hacen que su conocimiento sea algo indispensable para los profesionales del desarrollo en Internet.

### **Ventajas de PHP**

Algunas de las ventajas por las que usamos PHP son las siguientes:

- Costo: Es un software de distribución libre, y con documentación muy amplia en internet, por lo general no tardamos en encontrar tutoriales y guías gratuitas acerca de cómo utilizar este lenguaje.

- Sencillez y Versatilidad:
- PHP es un lenguaje de una sintaxis muy simple, y fácil de aprender, además posee una gran variedad de funciones que pueden ser utilizadas para mejorar el rendimiento de nuestros programas.
- Seguridad: PHP es un lenguaje de uso muy común en la web, además de ser libre, esto significa que una inmensa comunidad de programadores que utilizan este lenguaje están cooperando para la mejora del motor de PHP, por lo cual es cada vez más seguro y estable a medida que pasa el tiempo y aumenta su versión, otra ventaja es que en internet se pueden encontrar muchos tips para evitar errores que puedan convertirse en bugs peligrosos en nuestros sitios web.
- Rapidez. PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz. Está completamente escrito en C, así que se ejecuta rápidamente utilizando poca memoria
- Soporte para bases de datos, flash, y dinamismo en páginas web, que a pesar de que también encontramos estas características en otros lenguajes, PHP hace más fácil la implementación de ellas, por su versatilidad y sencillez.

### **Desventajas de PHP**

Las desventajas observadas en PHP son:

- No posee una abstracción de base de datos estándar, sino bibliotecas especializadas para cada motor (a veces más de una para el mismo motor).
- No posee adecuado manejo de internacionalización. Por su diseño dinámico no puede ser compilado y es muy difícil de optimizar.
- Por sus características favorece la creación de código desordenado y complejo de mantener.
- PHP no obliga a quien lo usa a seguir una determinada metodología en el momento de programar (muchos otros lenguajes tampoco lo hacen), aún 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.

### 2.2.3 JAVASCRIPT

**JavaScript** es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador. Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML.

**Fig. 2.3. APLICACIÓN DE JAVASCRIPT**

|                            |
|----------------------------|
| <b>JavaScript</b>          |
| <b>Información general</b> |

|                             |                                                         |
|-----------------------------|---------------------------------------------------------|
| <b>JavaScript</b>           |                                                         |
| <b>Información general</b>  |                                                         |
| <b>Extensión de archivo</b> | .js                                                     |
| <b>Tipo de MIME</b>         | application/javascript,<br>text/javascript <sup>2</sup> |
| <b>Tipo de formato</b>      | scripting                                               |
| <b>Formato abierto</b>      | ?                                                       |

|                         |                                                                                                      |
|-------------------------|------------------------------------------------------------------------------------------------------|
| <b>Paradigma</b>        | Multi-paradigma, Programación funcional basada en prototipos, imperativo, Interpretado (Scripting) . |
| <b>Apareció en</b>      | 1995                                                                                                 |
| <b>Diseñado por</b>     | Netscape Communications Corp, Mozilla Foundation                                                     |
| <b>Tipo de dato</b>     | débil, dinámico, duck                                                                                |
| <b>Implementaciones</b> | SpiderMonkey, Rhino, KJS,JavaScriptCore, V8.                                                         |
| <b>Dialectos</b>        | ECMAScript                                                                                           |
| <b>Influido por</b>     | Java, Perl, Self, Python,C, Scheme                                                                   |
| <b>Ha influido a</b>    | ObjectiveJ, JScript, JScript .NET, TIScript                                                          |

**Fuente: COLOMES FORMOS -2009.**

## **CARACTERÍSTICAS**

- Los lenguajes Script sirven principalmente para añadir contenido “dinámico” a las páginas Web. El contenido dinámico es, básicamente, aquel contenido de una página Web (gráficos, textos y demás) que puede cambiar en el tiempo, según las iteraciones que realiza el usuario a lo largo de su ejecución.
- En una página Web HTML que no contiene código Script el contenido no puede cambiar a lo largo del tiempo, es por eso que el contenido que contiene se le llama “estático”.
- Javascript es un lenguaje de programación Script, creado por la empresa Netscape, y cuya principal función es incluir contenido “dinámico” a las páginas web.
- JavaScript es un lenguaje que lo mismo puede ser utilizado para pequeñas comprobaciones o alteraciones del contenido HTML, como también ser usado con funciones y demás estructuras complejas según requiera el sitio.
- Algunos consideran que JavaScript no es un lenguaje orientado a objetos por lo que no es necesario crear ni clases, ni métodos, etc., como en lenguajes de programación estructurada como C, Perl, etc.

- **Sintaxis.** Es similar a la usada en Java y C, al ser un lenguaje del lado del cliente este es interpretado por el navegador, no se necesita tener instalado ningún compilador adicional.
- **Lenguaje no extensible.** JavaScript en comparación con Java, no es totalmente extensible. El modelo de JavaScript está formado por un número limitado de objetos básicos, propiedades, métodos y tipos de datos, que son suficientes para desarrollar aplicaciones cliente-servidor.
- **Basado en acciones.** Es un lenguaje basado en acciones que posee menos restricciones. Lenguaje interpretado. Esto significa que los archivos con el código se ejecutan directamente. Ese tipo de lenguajes son más sencillos y fáciles de aprender.

Además, normalmente es más fácil desarrollar, modificar y depurar programas para lenguajes interpretados porque no es necesario recompilar el código cada vez que se realiza un cambio. Los programas para lenguajes interpretados suelen requerir un mayor tiempo de ejecución que los programas compilados.

En el caso de JavaScript esto no supone un problema, ya que, con los anchos de bandas actuales, la velocidad de ejecución es despreciable frente a la velocidad de transmisión. Los scripts escritos en JavaScript se compilan en formato Byte code (como los programas Java) y evalúan mientras se está transmitiendo el script.

Actualmente existen dos tipos de JavaScript:

El que se ejecuta en el cliente, este es el Javascript propiamente dicho, aunque técnicamente se denomina Navigator JavaScript. Javascript que se ejecuta en el servidor, es más reciente y se denomina LiveWire Javascript.

### **Ventajas**

- El código Javascript se ejecuta en el cliente por lo que el servidor no es solicitado más de lo debido.
- El lenguaje de scripting es seguro y fiable porque está en claro y hay que interpretarlo, por lo que puede ser filtrado; para el mismo Javascript, la seguridad es casi total y sólo en su primera versión el CIAC (Computer Incident Advisor y Committee) señaló problemas de leve entidad, entre ellos la lectura de la caché y de los sitios visitados, de la dirección e-mail y de los archivos presentes en el disco. Sin embargo, estos fallos se corrigieron ya en las versiones de Netscape sucesivas a la 2.0.
- No requiere tiempo de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos.
- Javascript tiene la ventaja de ser incorporado en cualquier página Web, puede ser ejecutado sin la necesidad de instalar otro programa para ser visualizado.
- Es un lenguaje independiente de la plataforma.
- Es simple, no hace falta tener conocimientos de programación para poder hacer un programa en JavaScript y para su desarrollo se necesita básicamente un editor de textos y un navegador Web compatible con JavaScript para sus pruebas.
- Maneja objetos dentro de nuestra página Web y sobre ese objeto se pueden definir diferentes eventos. Dichos objetos facilitan la programación de páginas interactivas, a la vez que se evita la posibilidad de ejecutar comandos que puedan ser peligrosos para la máquina del usuario, tales como formateo de unidades, modificar archivos, etc.

- Es dinámico, responde a eventos en tiempo real. Eventos como presionar un botón, pasar el puntero del ratón sobre un determinado texto o el simple hecho de cargar la página o caducar un tiempo. Con esto podemos cambiar totalmente el aspecto de nuestra página al gusto del usuario, evitándonos tener en el servidor una página para cada gusto, hacer cálculos en base a variables cuyo valor es determinado por el usuario, etc.
- Los programas JavaScript tienden a ser pequeños y compactos no requieren mucha memoria ni tiempo adicional de transmisión.

#### **Desventajas**

- Los script tienen capacidades limitadas, por razones de seguridad, por lo cual no es posible hacer todo con Javascript, sino que es necesario usarlo conjuntamente con otros lenguajes evolucionados, posiblemente más seguros, como Java. Se puede poner en riesgo la seguridad del sitio, con el actual problema llamado XSS (significa en inglés *Cross Site Scripting* renombrado a XSS por su similitud con las hojas de estilo CSS).
- Una desventaja considerable también es que el código es visible y puede ser leído por cualquiera, incluso si está protegido con las leyes del copyright.
- El código del script debe descargarse completamente antes de poderse ejecutar, si los datos que un script utiliza son muchos, el tiempo que tardará en descargarse será muy largo.

### **2.3 BASES DE DATOS CON MYSQL**

MySQL es un sistema administrador de bases de datos relacionales (RDBMS, Relational Data Base Management System) de código abierto. En sus inicios MySQL no contaba con soporte para algunas construcciones de SQL importantes como son las

llaves foráneas, esto se reflejó en la oposición de los usuarios a su uso. Sin embargo, últimamente, MySQL cuenta con una gran cantidad de usuarios preocupados por mejorar este sistema en términos de licencia, rendimiento y facilidad de uso. Su aceptación se ha visto ayudada en parte por una amplia variedad de tecnologías como PHP, Perl, Python y similares que ha fomentado su uso por medio de módulos y extensiones estables y bien documentadas.

MySQL tiene un diseño multihilo que le permite soportar una gran carga de forma muy eficiente. MySQL fue creado por una empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca. La parte "AB" del nombre de la compañía es el acrónimo del sueco "aktiebolag", o "stock company", o "sociedad anónima".

Las bases de datos son por lo general de utilidad, quizá la familia más útil de productos de software en informática moderna. Como muchos productos competidores, tanto gratuitos como comerciales, MySQL no es una base de datos hasta que se le asigna una estructura y forma. Podría pensar en esto como la diferencia entre una base de datos y un RDBMS (es decir, RDBMS mas requisitos de usuario es igual a una base de datos).

### **Características de MySQL.**

La siguiente lista describe algunas de las características del software de base de datos MySQL.

- Proporciona sistemas de almacenamiento transaccional y no transaccional.
- Un sistema de reserva de memoria muy rápido basado en threads (hilo de ejecución o subproceso).
- Joins muy rápidos usando un multi-join de un paso optimizado.
- Tablas hash en memoria, que son usadas como tablas temporales.
- Diversos tipos de columnas como son: enteros con/sin signo, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM, y tipos especiales.

- Registros de longitud fija y longitud variable.
- Soporte completo para operadores y funciones en las cláusulas de consultas SELECT y WHERE.
- Soporte completo para las cláusulas SQL GROUP BY y ORDER BY.
- Soporte de funciones de agrupamiento (COUNT(), COUNT(DISTINCT ...),AVG(), STD(), SUM(), MAX(), MIN(), y GROUP\_CONCAT()).
- Soporte para LEFT OUTER JOIN y RIGHT OUTER JOIN cumpliendo estándares de sintaxis SQL y ODBC.
- Soporte para alias en tablas y columnas como lo requiere el estándar SQL.
- Comandos para obtener información acerca de la base de datos, el motor de base de datos, tablas e índices, así como comandos para determinar cómo el optimizador resuelve una consulta.
- Los nombres de funciones no colisionan con los nombres de tabla o columna. La única restricción es que para una llamada a una función, no se permiten espacios entre el nombre de función y el '(' a continuación.
- Proporciona soporte a grandes bases de datos.
- Se permiten hasta 64 índices por tabla. Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas.
- MySQL server tiene soporte para comandos SQL para checar, optimizar, y reparar tablas. Estos comandos están disponibles a través de la línea de comandos y el cliente MySQLcheck. MySQL también incluye myisamchk, una utilidad de línea de comandos muy rápida para efectuar estas operaciones en tablas MyISAM.

### **Ventajas de MySQL**

MySQL ofrece varias ventajas respecto a otros sistemas gestores de base de datos:

- Tiene licencia pública, permitiendo no solo la utilización del programa sino también la consulta y modificación de su código fuente. Resulta por tanto fácil de personalizar y adaptar a las necesidades concretas.

- El programa está desarrollado en C y C++, lo que facilita su integración en otras aplicaciones desarrolladas igualmente en esos lenguajes.
- Puede ser descargado gratuitamente de Internet (<http://www.MySQL.com>) haciendo uso de su licencia GLP (GENERAL PUBLIC LICENSE).
- Aunque MySQL es software libre, MySQL AB distribuye una versión comercial llamada “MySQL Enterprise Server”, que se diferencia de la versión libre en el soporte técnico que ofrece, en la posibilidad de integrar este gestor en un software propietario y en algunas herramientas de interfaz gráfica de usuario.
- MySQL utiliza el lenguaje SQL (Lenguaje de Consulta Estructurado) que es el lenguaje de consulta más usado y estandarizado para acceder a bases de datos relacionales. Soporta la sintaxis estándar del lenguaje SQL para la realización de consultas de manipulación, creación y de selección de datos.
- Es un sistema cliente/servidor, permitiendo trabajar como servidor multiusuario y de subprocesamiento múltiple, es decir, cada vez que se establece una conexión con el servidor, el programa servidor crea un subproceso para manejar la solicitud del cliente, controlando el acceso simultáneo de un gran número de usuarios a los datos y asegurando el acceso solo a usuarios autorizados.
- MySQL dispone de un sistema sencillo de ayuda en línea, y de un monitor que permite realizar todas las operaciones desde la línea de comandos del sistema, sin necesitar ningún tipo de interfaz de usuario gráfica. Esto facilita la administración remota del sistema utilizando una herramienta como telnet.
- Es portable, es decir, puede ser llevado a cualquier plataforma informática. MySQL está disponible en más de veinte plataformas diferentes incluyendo las distribuciones más usadas de Linux, sistema operativo MacOS, Unix y MicrosoftWindows.
- Es posible encontrar gran cantidad de software desarrollado sobre MySQL o que lo soporte. En concreto son de destacar diferentes aplicaciones *open source* para la administración de las bases de datos a través de un servidor web.

- Es estable, lo que significa que no es propenso a estropearse incluso con mucha carga. El código MySQL ha funcionado en proyectos desde mediados de 1996 sin ningún problema. Cuando el software de base de datos MySQL fue distribuido entre un público más amplio, los nuevos usuarios rápidamente encontraron trozos de código no probados. Cada nueva versión desde entonces ha tenido pocos problemas de portabilidad incluso considerando que cada nueva versión ha tenido muchas nuevas funcionalidades.
- Es ligero, lo que significa que no requiere muchos recursos para instalarse o bien ejecutarse.
- Gestión de usuarios, manteniendo un muy buen nivel de seguridad en los datos. Las contraseñas son seguras porque todo el tráfico de contraseña está cifrado cuando se conecta con un servidor.

Todas estas características han hecho de MySQL uno de los sistemas gestores de bases de datos más utilizados en la actualidad, no solo por pequeñas empresas sino también por algunas grandes corporaciones, como pueden ser: Yahoo! Finance, Google, CISCO, MP3.com, Motorola, NASA, Silicon Graphics, Texas Instruments, A mediados del 2004 se estimaba que existían más de 5 millones de instalaciones activas del programa.

#### **Desventajas de MySQL.**

- Una de las desventajas de MySQL es que es gratis para aplicaciones de código abierto, de lo contrario hay que pagar licencia comercial.
- La característica “select into table” no está implementada. El soporte para transacciones no es por defecto (se ha de activar un modo especial).
- Integridad referencial: aunque sí se admite la declaración de llaves foráneas en la creación de tablas, internamente no las trata de forma diferente al resto de campos.
- Mantenimiento y soporte: Debido a que MySQL es "open-source", no es posible contratar algún tipo de mantenimiento o soporte. Si se da el caso de que el manejador de base de datos falle, el usuario deberá resolver el problema sin poder contar con ayuda técnica de los desarrolladores de software. Como se comentó con anterioridad esta desventaja se elimina si se contrata la licencia comercial.

Dar un juicio absoluto acerca de si un motor de base de datos es mejor que otro no es viable. Realmente, cada herramienta tiene fortalezas y debilidades que deben analizarse en el momento de decidir que RDBMS se va a escoger, partiendo, eso sí, del criterio de potencializar las fortalezas que puede tener cada motor de acuerdo al tipo de proyecto que se esté desarrollando; verificando, al mismo tiempo, que las posibles debilidades no sean elementos críticos en la aplicación que se va a desarrollar.

Desarrollar un estudio de este tipo a través de software libre trae ventajas no solo académicas, sino económicas, ya que este tipo de software está a disposición del público en general.

## **2.4 METODOLOGÍAS Y TECNOLOGÍAS PARA EL DESARROLLO DE APLICACIONES WEB (rias's)**

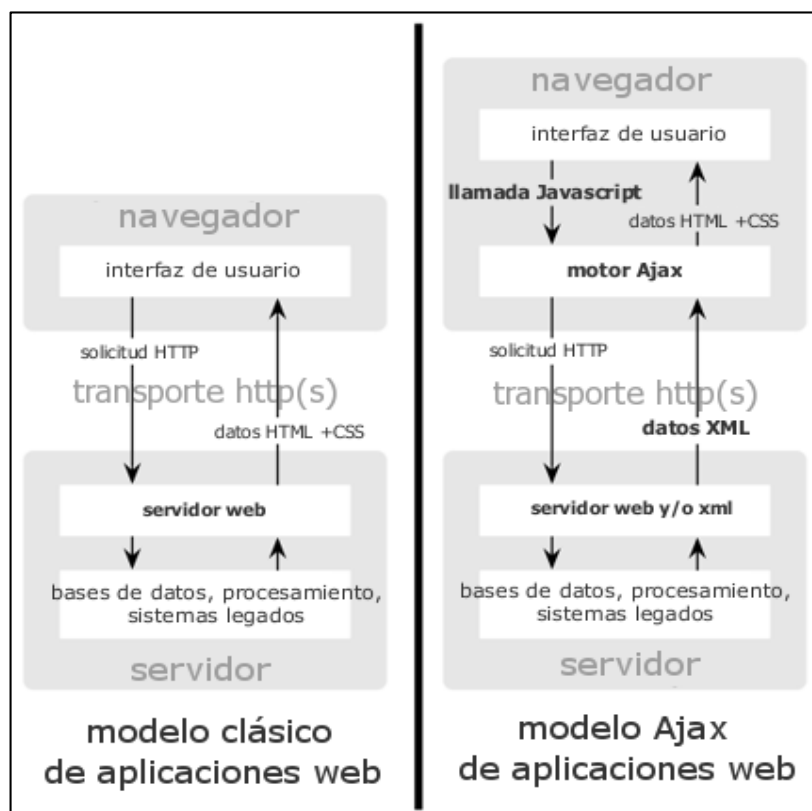
### **2.4.1 AJAX**

Se define AJAX como un acrónimo de Asynchronous JavaScript + XML, que se puede traducir como "JavaScript asíncrono + XML". AJAX no es una tecnología en sí, sino que se trata de varias tecnologías independientes que se unen. Es realmente muchas tecnologías, cada una floreciendo por su propio mérito, uniéndose en poderosas nuevas formas. AJAX incorpora:

- Presentación basada en estándares usando XHTML y CSS;
- Exhibición e interacción dinámicas usando el Document Object Model;
- Intercambio y manipulación de datos usando XML and XSLT;
- Recuperación de datos asíncrona usando XML HttpRequest; y JavaScript poniendo todo junto.

El modelo clásico de aplicaciones Web funciona de esta forma: La mayoría de las acciones del usuario en la interfaz disparan un requerimiento HTTP al servidor web. El servidor efectúa un proceso (recopila información, procesa números, hablando con varios sistemas propietarios), y le devuelve una página

HTLM al cliente. Este es un modelo adaptado del uso original de la Web como un medio hipertextual, pero como fans de acce The Elements of User Experience • sabemos, lo que hace a la Web buena para el hipertexto, no la hace necesariamente buena para las aplicaciones de software.



Fuente: COLOMES FORMOS, XAVIER - 2009.

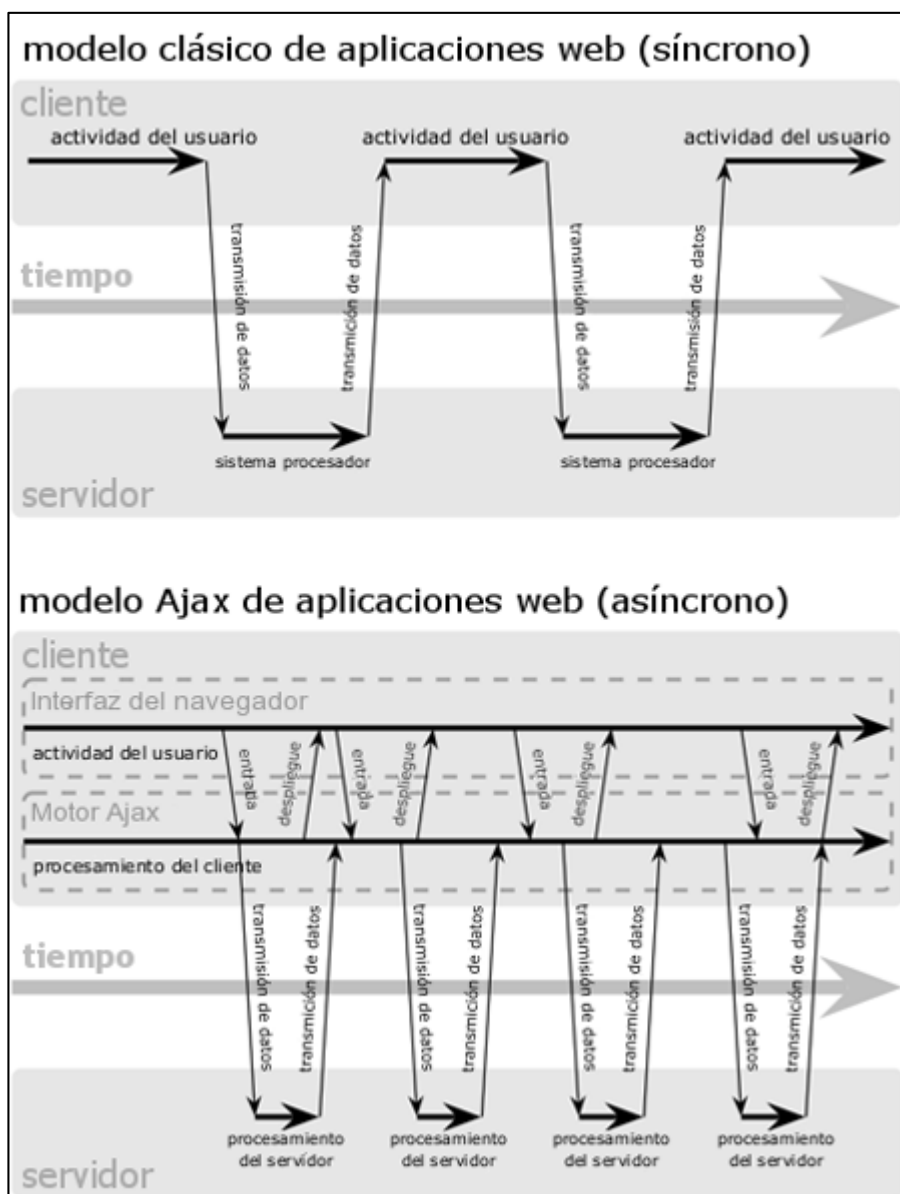
**Figura 2.4: El modelo tradicional para las aplicaciones Web (izq.) comparado con el modelo de AJAX.**

Este acercamiento tiene mucho sentido a nivel técnico, pero no lo tiene para una gran experiencia de usuario. Mientras que el servidor está haciendo lo suyo, el usuario está esperando. Y, en cada paso de la tarea, el usuario espera por más. Si estuviéramos diseñando la Web desde cero para aplicaciones, no querríamos hacer esperar a los usuarios. Una vez que la interfaz está cargada, la interacción del usuario no debería detenerse cada vez que la aplicación necesita algo del servidor.

Una aplicación AJAX elimina la naturaleza de arrancar-frenar- arrancar-frenar • de la interacción en la Web introduciendo un intermediario -un motor AJAX- entre el usuario y el servidor. Puede parecer que sumar una capa a la aplicación la haría menos reactiva, pero la verdad es lo contrario.

En vez de cargar un página Web, al inicio de la sesión, el navegador carga al motor AJAX (escrito en JavaScript y usualmente en un frame oculto). Este motor es el responsable por renderizar la interfaz que el usuario ve y por comunicarse con el servidor en nombre del usuario. El motor AJAX permite que la interacción del usuario con la aplicación suceda asincrónicamente (independientemente de la comunicación con el servidor). Así el usuario nunca estará mirando una ventana en blanco del navegador y un icono de reloj de arena esperando a que el servidor haga algo.





Fuente: COLOMES FORMOS, XAVIER. (2009).

**Fig.2.5: El patrón de interacción sincrónica de una aplicación Web tradicional (arriba) comparada con el patrón asincrónico de una aplicación**

## AJAX..

Cada acción de un usuario que normalmente genera un requerimiento HTTP toma la forma de un llamado JavaScript al motor AJAX en vez de ese

requerimiento. Cualquier respuesta a una acción del usuario que no requiera una viaje de vuelta al servidor (como una simple validación de datos, edición de datos en memoria, incluso algo de navegación) es manejado por su cuenta. Si el motor necesita algo del servidor para responder (sea enviando datos para procesar, cargar código adicional, o recuperando nuevos datos) hace esos pedidos asincrónicamente, usualmente usando XML, sin frenar la interacción del usuario con la aplicación.

Como se mencionó AJAX no es una nueva tecnología sino la utilización de varias tecnologías ya existentes, de una manera coordinada. Se ejecuta en el cliente, es decir, en el navegador del usuario mientras este mantiene una comunicación asíncrona con el servidor en segundo plano.

Las tecnologías que forman AJAX son la mayoría de las tecnologías disponibles para páginas Web:

- XHTML y CSS, para crear una presentación basada en estándares de todos los elementos de la aplicación y la información recibida por el servidor.
- DOM, para la interacción y manipulación dinámica de la presentación.
- XML, XSLT y JSON, para el intercambio y la manipulación de información, específicamente XML es el formato de los datos transmitidos del servidor al cliente (navegador) y que posteriormente serán mostrados.
- XML HttpRequest, para el intercambio asíncrono de información.
- JavaScript, Para manejar el objeto XMLHttpRequest y para unir todas las demás tecnologías.
- Lenguaje de servidor, Genera la información útil en XML y la envía al navegador, puede ser PHP o ASP.

## Programación AJAX frente a los programas nativos

### Ventajas

- No se necesita instalar una aplicación AJAX, solo es necesario introducir la URL en el navegador y puede comenzarse a trabajar.
- Puede accederse a la última versión de la aplicación con solo actualizarla en el servidor Web; ya no es necesario actualizaciones individuales en los clientes.
- Es independiente del sistema operativo, solo es necesario un navegador.
- La curva de aprendizaje del uso de una aplicación AJAX es bastante rápida.
- Una aplicación AJAX utiliza menos recursos, espacio en disco y memoria que un programa nativo.
- La migración de una aplicación AJAX a otras arquitecturas de hardware y software es más sencillo que el caso contrario.

### Desventajas

- Las aplicaciones AJAX, en comparación con los programas nativos, tiene acceso limitado a los recursos locales de la computadora.
- Las aplicaciones AJAX no pueden acceder a bases de datos locales como lo hacen los programas nativos sino que debe transferir la información a través de redes generalmente más lentas.

### 2.4.2 MOZILLA PRISM



**Mozilla Prism**, anteriormente conocido como **WebRunner**, es un producto de Mozilla actualmente fuera de desarrollo. Su objetivo principal era el de integrar aplicaciones web con el entorno de escritorio, lo que permitía a las aplicaciones web ejecutarse desde el escritorio y configurarse independientemente del navegador web por defecto. Era comúnmente usado

junto con aplicaciones Google AJAX , como Gmail y Google Docs. La publicación de Prism se hizo en octubre de 2007. El software que mantuvo en fase beta fue reemplazado en el 2011 cuyo proyecto se centraba definitivamente como aplicación operativa.

Mozilla Prism fue la primera de estas aplicaciones en llegar al gran público, es la apuesta de Mozilla. Estas son las principales cosas a tener en cuenta:

- Es multiplataforma: Windows, Linux y Mac funcionan perfectamente.
- Su motor interno es el de Firefox, por tanto la compatibilidad con casi todas las páginas está garantizada. Por lo contrario, las extensiones que se tengan que usar no funcionarán de manera directa, y aunque es posible hacerlas funcionar, es algo un tanto laborioso.
- Existe en dos versiones: como extensión para Firefox o como aplicación independiente. Las dos maneras funcionan de forma similar, con las mismas opciones.
- Posibilidad de ocultar o no la barra de navegación.
- Permite elegir cualquier imagen como icono de la aplicación que se creará.
- Tiene soporte para notificaciones, como por ejemplo cuando llegue un email; también nos puede avisar en la barra de tareas/Dock/bandeja de notificación cambiando el icono para mostrar el número de elementos sin leer.
- Integración con la bandeja de notificación, a la que podemos minimizar la aplicación.

### 2.4.3 ADOBE AIR

Adobe AIR es una tecnología que permite la creación de aplicaciones de escritorio (de propósito general) a partir de tecnologías de desarrollo de páginas web, como pueden ser HTML, Ajax o Flash. Con Adobe AIR los desarrolladores pueden aprovechar sus conocimientos a la hora de crear páginas web para hacer aplicaciones multimedia para el escritorio. Es decir, a partir de un desarrollo de una aplicación web, crear una aplicación general con los mismos contenidos o

utilidades del sitio web. Permite desarrollar RIAs (Rich Internet Application) para los diferentes sistemas operativos (siguiendo el mantra “write once run everywhere”), podemos utilizar Flash, Flex, Javascript, HTML y Ajax para desarrollar sobre AIR. Existe una gran cantidad de aplicaciones en AIR para diseñadores web.

Adobe AIR se distribuye gratuitamente para cualquier persona, tanto los usuarios que deseen instalar y usar programas creados con AIR, como para los desarrolladores que quieran crear aplicaciones de escritorio a partir de sus proyectos web. Por tanto, no es necesario disponer de nada especial que haya que pagar para poder beneficiarse de las capacidades de Adobe AIR. Según el equipo de Adobe, están seguros que esta situación de gratuidad permanecerá por siempre. Eso sí, usando las herramientas de Adobe, como Flash, Dreamweaver o Flex Builder, se simplifica bastante la generación de programas AIR.

Una de las ventajas, adicionales a las que ya he apuntado, sobre AIR es que las aplicaciones generadas son multiplataforma. Es decir, los programas de escritorio de Adobe AIR se podrán utilizar en cualquier ordenador, independientemente del sistema operativo que tenga. Esto se consigue porque cuando descargamos una aplicación con Adobe Air, se comprueba si está instalado el entorno AIR en nuestro ordenador. Si no estaba instalado se descarga el entorno de ejecución (ocupa actualmente unos 10 megas) necesario para que las aplicaciones funcionen en nuestro sistema operativo concreto. Una vez ya el entorno se ha instalado en nuestro sistema no se tiene que volver a descargar ni instalar de nuevo. Con ello, Adobe AIR consigue acercarse al paradigma, "programa una vez y ejecuta donde quieras", a la vez que se hace universal y al alcance de todos los usuarios y plataformas. Queda saber hasta que punto han conseguido solucionar todos los problemas de incompatibilidades que en principio pueden existir de ejecutar un mismo software en distintos sistemas.

AIR, de Adobe, es un producto novedoso y con grandes posibilidades para toda la comunidad de desarrolladores. Adobe ha irrumpido con un producto singular que da respuesta a la necesidad de crear aplicaciones de escritorio a partir de los conocimientos de millones de personas en la creación de páginas web.

#### 2.4.4. JAVA FX

**JavaFX** es una familia de productos y tecnologías de Sun Microsystems, adquirida por Oracle Corporation, para la creación de Rich Internet Applications (RIAs), esto es, aplicaciones web que tienen las características y capacidades de aplicaciones de escritorio, incluyendo aplicaciones multimedia interactivas. Las tecnologías incluidas bajo la denominación JavaFX son JavaFX Script y JavaFX Mobile, aunque hay más productos JavaFX planeados.

Las aplicaciones JavaFX pueden ser ejecutadas en una amplia variedad de dispositivos. En su versión (JavaFX 1.3, abril 2010) permite crear aplicaciones de escritorio, para celulares, la Web, TV, consolas de videojuegos, reproductores Blu-ray, entre otras plataformas planeadas. En octubre de 2011 fue lanzada la versión 2.0. Para el desarrollo de aplicaciones JavaFX un lenguaje declarativo, tipado llamado JavaFX Script, además puede integrarse código Java en programas JavaFX. JavaFX es compilado a código Java, por lo que las aplicaciones JavaFX pueden ser ejecutadas en computadores con la máquina virtual de Java instalada (JRE), o celulares corriendo Java ME.

JavaFX fue anunciado en la conferencia de desarrolladores JavaOne en mayo de 2007 y liberado en diciembre de 2008.

La intención de Sun Microsystems respecto de JavaFX es competir en el espacio que ya ocupan Flash de Adobe, y Silverlight de Microsoft.

En palabras de James Gosling "La mayoría de los lenguajes de script están orientados a las páginas web; éste está orientado a las interfaces que son altamente animadas".

### JavaFX Mobile: Java+Linux en teléfonos móviles

Se trata de JavaFX Mobile, un sistema software completo (SO incluido) para teléfonos móviles, el sistema está construido en torno al kernel Linux y a tecnologías Java (todo bajo licencia GPL), y pretende proporcionar a los fabricantes de teléfonos móviles un entorno software completo (con sus herramientas de desarrollo incluidas) en el que basar sus productos, de forma que estos se puedan centrar en el desarrollo de servicios que los diferencien de sus competidores y no en reinventar la rueda una y otra vez].

### Ventajas

- **JavaFX** es una extensión a la plataforma Java que ofrece interactividad, animación y programación compatible con AJAX, Flash de Adobe y Silverlight de Microsoft. A diferencia de éstas anteriores, utiliza los runtimes Java ya instalados en el cliente local en lugar de JavaScript.
- **JavaFX** permite a los creadores de contenido crear contenido rico en medios, lo que incluye el arrastrar y soltar contenido de equipos de escritorio o móviles hacia el escritorio, **algo que no es posible en ninguna otra RIA** .
- **JavaFX** se trata de un sistema de desarrollo para la web que pretende competir con el Flash de Adobe y el Silverlight de Microsoft, con el añadido de que es Open Source .
- Millones de desarrolladores, muchos equipos con Java instalado y a eso hay que sumar que Java es libre bajo GPL.
- Multiplataforma, desde el escritorio (corriendo sobre una máquina virtual Java SE) hasta los dispositivos móviles para los que habrá un JavaFX Mobile.
- Utiliza el mismo lenguaje para la web, para el escritorio y para la telefonía móvil.

- **JavaFX Mobile** utiliza tecnologías estándar de la industria, lo que hace posible que las aplicaciones construidas con JavaFX pueden ejecutarse en una amplia gama de dispositivos habilitados para Java, y los proveedores de contenido pueden agregarlos rápidamente a sus dispositivos.

### **Desventajas**

- Con JavaFx, se agrega un lenguaje de script complejo, que necesita mantener en memoria su propia estructura semántica y sus propios objetos de plataforma, simultáneamente con los objetos gráficos.
- Al ser un lenguaje de script agrega un factor de ralentización importantísimo, por eso el desempeño no es óptimo.
- Por otro lado, las APIs de tratamiento de XML ya vienen incorporadas al JDK, generalmente están implementadas a bajo nivel, por lo que un sistema de templates es mucho mas liviano que soportar JavaFX.

## **2.5 INDICADORES DE CALIDAD DEL SOFTWARE**

Existen varios indicadores de calidad medibles para conocer si se alcanzan las metas trazadas, por tanto los indicadores son metas para satisfacer a los clientes, metas que son trazadas por cada empresa o equipo de desarrollo respectivamente y que nadie ajeno a la empresa lo debe saber. A continuación se muestra la descripción detallada de un conjunto de indicadores, destacando las principales categorías de la calidad, presentes en software de simulación, que las mismas son: fiabilidad, eficiencia y mantenibilidad. Después de realizar un estudio de los indicadores propuestos por diferentes investigadores que trabajan el área de calidad, los autores del presente artículo han realizado un estudio minucioso mediante el cual se seleccionaron un conjunto de indicadores que serán explicados posteriormente.

### **Indicadores propuestos para software de simulación**

Los indicadores de calidad propuestos forman parte de una colección de indicadores tomados del libro de Roger Pressman (2002) y de la autora Mariela la Torre Hernández (2008), en el presente artículo se describirán los mismos, pues son de gran utilidad para evaluar el software de simulación.

### **Indicadores para evaluar la fiabilidad**

**El tiempo medio entre fallos:** consiste en detectar una tasa de fallos, es decir, el tiempo medio que está entre el instante de tiempo en el que el sistema o un integrante del equipo de desarrollo detecta un fallo en el software y el tiempo que demoran en repararlo, este indicador se utilizará fundamentalmente en la etapa de diseño. Con esto se puede medir el impacto sobre la fiabilidad del software de que sea escaso o no.

Para evaluar este indicador se utiliza la siguiente ecuación:

$$TMEF = TMDF + TMDR$$

Donde:

TMDF: Tiempo Medio de Fallo.

TMDR: Tiempo Medio de Reparación.

### **Análisis del procedimiento:**

El objetivo del indicador es lograr disminuir el tiempo que el equipo de desarrollo emplea para reparar el fallo detectado, mientras menor sea el tiempo empleado, más eficaz será el producto y el impacto de la fiabilidad no será escaso. Para esto se establece un rango de comparación que va a depender de la magnitud del fallo encontrado, si es detectado un error sencillo para el tiempo sea bueno, deben demorarse como máximo 2 días, si por el contrario, el error es complejo como máximo 7 días.

**Proporción de la eliminación de defectos (P):** Es la capacidad para medir el rendimiento que tiene la eliminación de defectos durante cualquier etapa del proceso de desarrollo de software. En muchas ocasiones se introducen defectos, que son fácilmente

identificados y eliminados. Se considera defectos evadidos a todos los defectos que se insertaron antes o durante una etapa determinada, pero que no fueron encontrados en esa etapa, sino en otra posterior, estos traen consigo una costosa inversión de tiempo y esfuerzos para su eliminación.

Para evaluar este indicador, se consulta la métrica de rendimiento de la eliminación de defectos que se basa en la relación entre los defectos eliminados y todos los existentes en una etapa determinada. Este indicador se calcula con la siguiente ecuación:

$$P = (DE * 100) / (DE + DV)$$

Donde:

DE: Defectos Eliminados

DV: Defectos Evadidos

#### **Análisis del procedimiento:**

El objetivo o meta es lograr un rendimiento de 100%, o sea que todos los defectos sean encontrados y eliminados en la etapa que se esté analizando y se reduzcan los defectos evadidos. Es importante tener identificados los posibles defectos que ocurran frecuentemente en una etapa o fase determinada dentro del proceso de desarrollo, de manera que se pueda evitar o solucionar a tiempo. Suponiendo que se logre un rendimiento entre el 95% y 100% se puede considerar que el equipo de desarrollo está en buen estado, de lo contrario no cumple con están cumpliendo con su trabajo en la eliminación de errores y aumentaría los defectos evadidos, por lo tanto estarían en mal estado.

**Madurez de las pruebas (MP):** es un indicador que se desarrolla en el proceso de pruebas, no solo se preocupa de la completitud de los casos de prueba según los definidos para cumplir los requisitos, sino que también se interesa por cuales han obtenido resultados satisfactorios, para ello es necesario llevar un control de los casos de prueba que arrojaron resultados satisfactorios y el total de los casos de prueba definidos para el cumplimiento de los requisitos (Torre, 2008).

La madurez de las pruebas está dada por la siguiente ecuación:

$$MP = CPS / CPR$$

Donde:

CPS: Número de casos de pruebas que han obtenido un resultado satisfactorio.

CPR: Número de casos de pruebas a ejecutar requeridos para cubrir los requisitos.

### **Análisis del procedimiento:**

Mientras mayor sea el número de casos de pruebas que han obtenido un resultado satisfactorio de los casos realmente ejecutados y que se requieren para cubrir los requisitos, mayor será la madurez alcanzada por el equipo de probadores del software, lo ideal es un valor cercano a 1. Cuando el valor se encuentre entre 0.97 y 1 será mayor la madurez alcanza, cuando se encuentre entre 0.50 y 0.96 tendrá una trivial madurez, de lo contrario si el valor es menor que 0.50 la madurez será muy mínima, se han obtenidos muy poco casos de pruebas con resultados satisfactorios del total de casos de pruebas diseñados.

### **Indicadores para evaluar la Eficiencia en el Proceso Productivo y del Producto Software**

**El porcentaje de la eficiencia en la eliminación de errores en un tiempo determinado (PE):** es un indicador que permite evaluar la eficiencia del equipo de desarrollo, tanto individual como general en la eliminación de los problemas y errores detectados en el simulador.

Para poder evaluar este indicador, las autoras se apoyan de la métrica de la Eficiencia en la Eliminación de Problemas de donde se puede decir que el indicador se calcula de la siguiente forma:

$$PE = (total (DD) / total (DE)) * 100$$

Donde:

DD: número de defectos detectados en el proceso de desarrollo del simulador.

DE: número de defectos eliminados en el proceso de desarrollo del simulador.

**Análisis del procedimiento:**

Un valor de eficiencia alto es mucho mejor que un valor bajo, el cual es un resultado pobre. El objetivo o meta es una eficiencia de 100%, o sea, que se traten de eliminar los defectos o errores encontrados en el menor tiempo posible. El rango establecido es, en dependencia del número de errores detectados, se suponen que encuentran 3 fallos en un día, para que sea eficiente el equipo de desarrollo, deben demorarse como máximo 2 días, de lo contrario esto significa que el equipo de desarrollo no es muy bueno en la eliminación de errores.

**Número máximo de defectos eliminados por unidad de tiempo (MDET):** se propone para tener un control de la efectividad del tiempo dedicado a resolver los defectos encontrados durante cada una de las etapas del proceso de desarrollo de software, para ser corregidos en el menor tiempo posible.

Para la evaluación de este indicador, donde se recurrió a la métrica de Defectos eliminados por unidad de tiempo, se utilizó la siguiente ecuación:

$$MDET = n * DT / TiD$$

Donde:

TiD: Tiempo invertido en la eliminación de defectos en una etapa, dado en la unidad de tiempo n.

DT: Defectos Totales encontrados en una etapa de desarrollo.

n: Unidad de tiempo.

**Análisis del procedimiento:**

Los defectos por unidad de tiempo indican la efectividad del tiempo dedicado a las revisiones así como la eficiencia relativa de los múltiples métodos de eliminación de defectos que se estén usando. Si los defectos encontrados por unidad de tiempo han disminuido y el rendimiento no se ha incrementado, puede significar que se está dedicando mucho tiempo a revisiones improductivas. En la medida que el rendimiento se incrementa, una disminución de los defectos por unidad de tiempo es natural. Para

que el tiempo sea efectivo deben eliminar de 10 a 12 errores encontrado en 2 días como máximo, de lo contrario el tiempo estaría en mal estado.

**Efectividad en el proceso productivo (EPP):** Este indicador centra su atención, como bien su nombre lo indica, en la efectividad del proceso se desarrollo. Que se puede calcular de la siguiente manera:

$$EPP = PD/PP$$

Donde

PD: número de productos software defectuosos

PP: número total de productos software producidos

#### **Análisis del procedimiento:**

Este indicador establece el rendimiento de los productos defectuosos sobre el total de la producción, esto se puede aplicar en un período de tiempo específico o una orden específica. Para esto se establece un criterio de comparación, entre los valores de 5 a 15 productos defectuosos, la producción está bien, entre 16 a 30 la producción es regular, mayor que esta cifra rendimiento de la producción estaría muy mal.

#### **Indicadores para evaluar la mantenibilidad del software**

**Indicador de Madurez de software (IMS):** Este indicador proporciona una indicación de la estabilidad del producto, basado en los cambios que ocurren en cada versión del producto, es decir, permite evaluar el rendimiento del producto software.

Se determina de la siguiente forma:

$$IMS = [Mt - (Fa + Fc + Fd)] / Mt$$

Donde:

Mt: Números de módulos en la versión actual.

Fa: Número de módulos en la versión actual que se han añadido.

Fc: Número de módulos en la versión actual que se han cambiado

Fd: Números de módulos de la versión anterior que se han borrado en la versión actual.

### **Análisis del procedimiento:**

A medida que el IMS se aproxima a 1,0 el producto se empieza a estabilizar. Para esto se establece que cuando el valor se encuentre entre 0.97 y 1 ya se considera que el producto está estabilizado, cuando se encuentre entre 0.50 y 0.96 está muy cerca de llegar a la estabilización, de lo contrario si el valor es menor que 0.50 la estabilización es mínima, lo que significa que le queda mucho por trabajar en las versiones de productos hasta llegar la última versión.

**Indicador de la disponibilidad (D):** permite determinar la probabilidad de que un programa funcione de acuerdo con los requisitos en un momento dado. Este indicador se determina con la siguiente ecuación:

$$D = TMDF / (TMDF + TMDR) \times 100$$

Donde:

TMDF: Tiempo medio de fallos

TMDR: Tiempo medio de reparación

### **Análisis del procedimiento:**

La medida de disponibilidad es algo más sensible al TMDR, es una medida indirecta de la facilidad de mantenimiento del software. En dependencia de los requisitos establecidos, si el valor está entre 0.97 y 1 se considera que el programa puede funcionar, cuando se encuentre entre 0.50 y 0.96 está muy cerca de llegar a funcionar correctamente, de lo contrario si el valor es menor que 0.50 la probabilidad de que el simulador funcione es muy remota.

## **2.6 EVALUACIÓN DE FACTORES DE CALIDAD DEL SOFTWARE**

La calidad es importante en todos los aspectos, pero especialmente en el desarrollo de software. El interés por la calidad crece de forma continua, a medida que los clientes se vuelven más selectivos y comienzan a rechazar productos poco fiables o que realmente no dan respuesta a sus necesidades. Las metas que se establezcan para la calidad del

producto van a determinar las metas a establecer para la calidad del proceso de desarrollo, ya que la calidad del producto va a estar en función de la calidad del proceso de desarrollo. Sin un buen proceso de desarrollo es casi imposible obtener un buen producto.

La calidad del producto de software se diferencia de la calidad de otros productos de fabricación industrial, ya que el software tiene ciertas características especiales:

- 1- El software es un producto mental, no restringido por las leyes de la Física o por los límites de los procesos de fabricación. Es algo abstracto, y su calidad también lo es.
- 2- Se desarrolla, no se fabrica. El coste está fundamentalmente en el proceso de diseño, no en la producción. Y los errores se introducen también en el diseño, no en la producción.
- 3- El software no se deteriora con el tiempo. No es susceptible a los efectos del entorno, y su curva de fallos es muy diferente a la del hardware. Todos los problemas que surjan durante el mantenimiento estaban desde el principio, y afectan a todas las copias del mismo; no se generan nuevos errores.
- 4- Es artesanal en gran medida. El software, en su mayoría, se construye a medida, en vez de ser construido ensamblando componentes existentes y ya probados, lo que dificulta aún más el control de su calidad. Aunque se ha escrito mucho sobre la reutilización del software, hasta ahora se han conseguido pocos éxitos tangibles.
- 5- El mantenimiento del software es mucho más complejo que el mantenimiento del hardware. Cuando un componente de hardware se deteriora se sustituye por una pieza de repuesto, pero cada fallo en el software implica un error en el diseño o en el proceso mediante el cual se tradujo el diseño en código de máquina ejecutable.
- 6- Es engañosamente fácil realizar cambios sobre un software, pero los efectos de estos cambios se pueden propagar de forma explosiva e incontrolada.

7- Como disciplina, el desarrollo de software es aún muy joven, por lo que las técnicas de las que disponemos aún no son totalmente efectivas o no están totalmente calibradas.

8- El software con errores no se rechaza. Se asume que es inevitable que el software represente errores.

Es importante destacar que la calidad del software debe ser considerada en todos sus estados de evolución (especificaciones, diseño, código, etc). No basta con tener en cuenta la calidad del producto una vez finalizado, cuando los problemas de mala calidad ya no tienen solución o la solución es muy costosa.

El software juega un papel muy importante para el desarrollo de las organizaciones. Día tras día son liberados para su uso distintos tipos de programas para diferentes clases de clientes, los hay para cada necesidad de tal manera que resulta difícil imaginar alguna situación en la que el software no estuviera presente, dado que es uno de los componentes básicos de la tecnología que se involucra en las empresas, no sólo como soporte a los procesos de negocio, productivos y administrativos, sino como parte integral de las estrategias corporativas para la generación de ventajas competitivas.

## 2.6.1 MODELOS DE CALIDAD DEL SOFTWARE

### **Modelo de Gilb**

El modelo de Gilb plantea la creación de una especificación de requisitos de calidad para cada proyecto que deben escribir conjuntamente el usuario y el analista. Es un modelo que permite determinar una lista de características que definen la calidad de la aplicación.

Puede ser de 2 tipos: (1) Originales y (2) de modelos tradicionales.

Las características se pueden medir mediante varias subcaracterísticas o métricas detalladas. Para cada una de ellas, se deben especificar los siguientes conceptos: (1) nombre y definición de la característica, (2) Escala o unidades de medición,

(3) recopilación de datos o prueba, (4) valor previsto, (5) valor óptimo, (6) valor en el sistema actual y (7) comentarios.

### **Modelo GQM (Goal – Question - Metric)**

El modelo GQM (objetivo-pregunta-métrica /goal – question - metric) de Basili y Rombach (1998) es una propuesta de objetivos / metas orientado a la definición de modelos de calidad. Se propone el paradigma GQM para evaluar la calidad de cada proyecto. Este modelo utiliza una propuesta para definir un modelo de calidad hasta obtener las métricas respectivas con el análisis e interpretación de los datos de las mediciones respectivas. Plantea el enfoque de medición para evaluar la calidad del software basado en la identificación de objetivos a lograr. El enfoque de GQM basa la mejora en la definición clara de procesos y productos. Proporciona la estructura para obtener los objetivos cruciales del proyecto. Consta de 3 etapas

- 1- Listar los objetivos principales del desarrollo y mantenimiento del proyecto
  - 2- Para cada objetivo, se deben obtener las preguntas que deben contestarse para saber si se están cumpliendo los objetivos
  - 3- Decidir qué medir para poder contestar las preguntas de manera adecuada, es decir, desarrollar un conjunto de métricas que ayuden a responder la pregunta.
- El modelo GQM es un enfoque útil para decidir qué medir. Es un enfoque orientado a metas, por lo tanto, permite a los tomadores de decisión, elegir aquellas métricas que se relacionen a las metas más importantes de los problemas más urgentes.

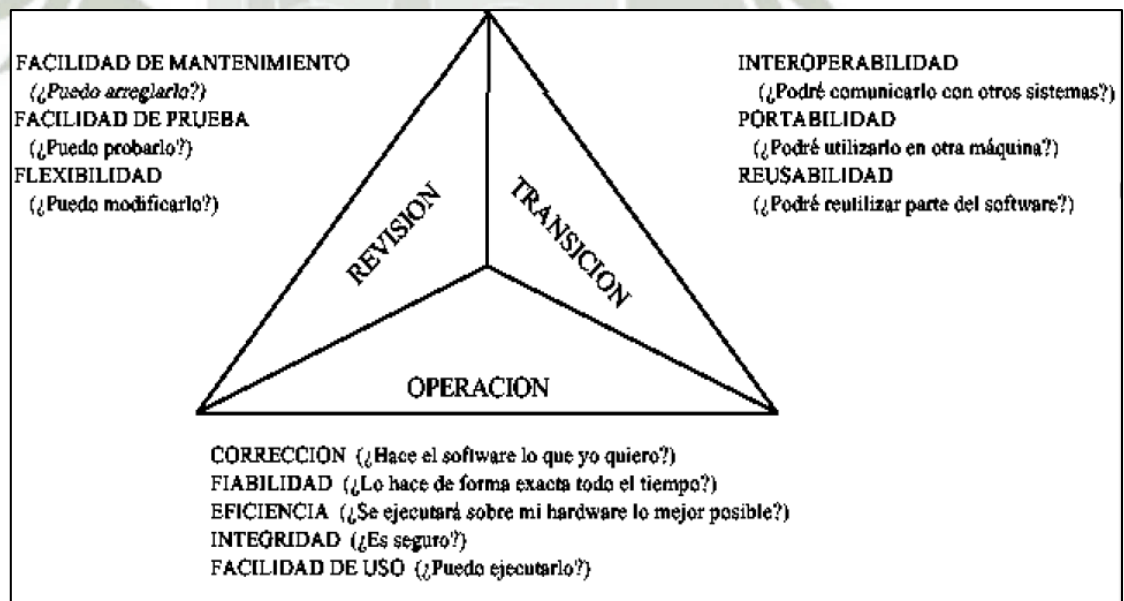
### **Modelo de McCall**

La calidad de un sistema, aplicación o producto es tan buena como los requisitos que describen el problema, el diseño que modela la solución, el código que

conduce a un programa ejecutable y las pruebas que ejercitan el software para detectar errores. Un buen Ingeniero del Software utiliza mediciones que evalúan la calidad del análisis y los modelos de diseño, el código fuente y los casos de prueba que se han creado al aplicar la Ingeniería de Software. Para lograr estas evaluaciones de la calidad en tiempo real, el Ingeniero debe utilizar medidas técnicas que evalúan la calidad con objetividad, no con subjetividad. Hace 25 años se definieron factores de calidad como los primeros pasos hacia el desarrollo de métricas de calidad del software.

El modelo de McCall organiza los factores en tres ejes o puntos de vista desde los cuales el usuario puede contemplar la calidad de un producto (1) Operación del producto, (2)

Revisión del producto y (3) Transición del producto (Figura 21). Cada punto de vista se descompone en una serie de factores que determinan la calidad de cada una de ellos. Cada factor determinante de la calidad, se descompone, a su vez, en una serie de criterios o propiedades que determinan su calidad. Los criterios pueden ser evaluados mediante un conjunto de métricas. Para cada criterio deben fijarse unos valores máximo y mínimo aceptables para cada criterio.



Fuente: Métricas de control de calidad-2012

Fig. 2.6. Factores de calidad del modelo McCall

Antes de comenzar a utilizar el modelo de McCall hay que seguir las siguientes pautas: (1)

Se aceptan los factores, criterios y métricas que propone el modelo, (2) Se aceptan las relaciones entre factores y criterios, y entre criterios y métricas; y (3) Se selecciona un subconjunto de factores de calidad sobre los que se aplican los requisitos de calidad establecidos para el proyecto.

Al comienzo del proyecto habrá que especificar los requisitos de calidad del software, para lo cual se seleccionarán los aspectos inherentes a la calidad deseada del producto, teniendo en cuenta lo siguiente:

- Las características particulares del propio producto que se está diseñando: por ejemplo, su ciclo de vida, es decir, si se espera que sea largo implicará un mayor énfasis en la facilidad de mantenimiento y la flexibilidad, o bien si el sistema en desarrollo está destinado a un entorno donde el hardware evoluciona rápidamente implicará como requisito su portabilidad.
- La relación calidad-precio, que puede evaluarse a través del coste de cada factor de calidad frente al beneficio que proporciona. La siguiente tabla muestra la relación calidad-precio para cada factor considerado.
- La determinación de las etapas del ciclo de vida, ya que es necesario evaluar cada factor de calidad para conocer en cuales se dejan sentir más los efectos de una calidad pobre con respecto a cada uno de los factores.

Las propias interrelaciones entre los factores debido a que algunos factores pueden entrar en conflicto entre sí: por ejemplo, la eficiencia plantea conflictos prácticamente con todos los demás factores de calidad.

También habrá que establecer valores deseables para los criterios, para lo cual se emplearán datos históricos, el promedio en la industria, y con ellos se concretarán los valores finales y otros intermedios o predictivos en cada período de medición durante el desarrollo, así como unos valores mínimos aceptables. La explicación para cualquier selección o decisión deberá ser adecuadamente documentada.

En la fase de desarrollo será necesario implementar las métricas elegidas, analizar sus resultados y tomar medidas correctivas cuando los valores obtenidos estén por debajo de los mínimos aceptables. Una vez finalizado el proyecto será necesario contrastar las medidas predictivas utilizadas y comprobar si, en efecto, se pueden tomar como indicadores de los valores finales.

### **Modelo FURPS**

El modelo FURPS propuesto por Robert Grady y Hewlett Packard Co (HP) cuenta con 5 características de calidad del software: (1) Funcionalidad, (2) Facilidad de uso, (3) Confiabilidad, (4) Performance y (5) Facilidad de soporte. Además plantea 2 categorías de requerimientos, las cuales son:

1- requerimientos funcionales (F): especifican funciones que el sistema debe ser capaz de realizar, sin tomar restricciones físicas a consideración, y se definen a través de las entradas y salidas esperadas.

2- requerimientos no funcionales (URPS): Usability (Facilidad de uso), Reliability (Confiabilidad), Performance y Supportability (Facilidad de soporte). describen atributos del sistema o atributos del ambiente del sistema.

Funcionalidad - Los requisitos de funcionalidad deben incluir : (1) Conjunto de Características, (2) Capacidades y (3) Seguridad. Facilidad de Uso - Deben incluir subcategorías tales como: (1) Factores humanos, (2) Estéticos, (3) Consistencia en la Interfaz de Usuario, (4) Ayuda en línea, (5) Asistentes, (6) Documentación del usuario y (7) Material de capacitación.

Confiabilidad - Se considera requisitos de confiabilidad: (1) Frecuencia y severidad de fallas, (2) Recuperación a fallos y (3) Tiempo entre fallos.

Performance (Rendimiento) - Un requisito de rendimiento impone condiciones a los requisitos funcionales. Por ejemplo a una acción dada, se pueden especificar los siguientes parámetros de rendimiento: (1) Velocidad, (2) Eficiencia, (3) Disponibilidad, (4) Tiempo de Respuesta, (5) Tiempo de Recuperación y (6) Utilización de Recursos.

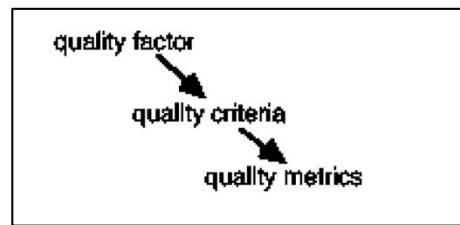
Soporte - Los requisitos de soporte pueden incluir: (1) Requisitos de instalación, (2) Requisitos de Configuración, (3) Requisitos de Adaptabilidad y (4) Requisitos de Compatibilidad.

FURPS se aplica realizando los siguientes pasos: (1) asignación de prioridades y (2) definición de los atributos de calidad que pueden ser medidos

Para controlar la calidad en el proceso de fabricación de su hardware, HP contempla un concepto global bajo las siglas FURPS: Funcionalidad, Facilidad de uso, Confiabilidad (Reliability), Prestaciones (Performance) y Servicio. Se trata de variables sobre las que se incide durante todo el ciclo de vida del producto compuesto por diversas fases. La primera es la concepción del producto a partir de análisis del mercado, después se marcan los objetivos y se define el proyecto, apartado que se mezcla inevitablemente con la fase de diseño real del producto e investigación. Posteriormente, se procede a traducir los diseños en piezas o prototipos reales en el laboratorio. El Manufacturing Release es la fabricación real a partir de los prototipos. Una vez fabricado, la siguiente fase en la vida del producto es su comercialización hasta que se convierta en algo obsoleto y se deje de fabricar como última etapa del ciclo.

### **Modelo de BOEHM**

El modelo de Boehm (1978) agrega algunas características a las existentes en el modelo de McCall y representa una estructura jerárquica de características, cada una de las cuales contribuye a la calidad total. Consiste en un modelo de descomposición de características de calidad del software en 3 niveles (usos principales, componentes intermedios y componentes primitivos) previos a la aplicación de métricas. Este modelo plantea factores de calidad formados por criterios de calidad y métricas respectivas (Figura 23).



**Fig. 2.7: Modelo de Boehm**

El modelo de Boehm tiene como finalidad que a través de la calidad del software, el software: (1) realice lo que desea el usuario, (2) utilice recursos informáticos de manera correcta y eficiente, (3) sea fácil de utilizar y aprender; y (4) sea bien diseñado, codificado, probado y mantenido. Este modelo es similar al de McCall ya que presenta una jerarquía de características, está basado en un amplio rango de características e incorpora 19 criterios que incluyen características de performance del hardware.

### **Modelo SATC (Software Assurance Technology Center)**

SATC desarrolló un modelo dinámico que permite la producción de varios proyectos en desarrollo. Los datos del proyecto son usados para realizar proyecciones acerca de los riesgos y puntos de control del proyecto. Este modelo utiliza un amplio rango de medidas o métricas y; tiene objetivos, atributos y métricas asociadas a los procesos de desarrollo y al software propiamente dicho. Este modelo define un conjunto de metas u objetivos relacionados al producto de software y atributos del proceso que permiten realizar indicaciones de la probabilidad de éxito de los objetivos. Un conjunto de métricas es seleccionado o desarrollado, el cual medirá los atributos seleccionados.

A partir del concepto de calidad del software, se deducen 4 metas u objetivos.

1- Calidad de los Requerimientos: el objetivo de esta meta es que los documentos de requerimientos estén completos, no ambiguos y entendibles.

Esta meta tiene los siguientes atributos: (1) Ambigüedad: requerimientos con múltiples significados, (2) Integridad: ítems a ser especificados, (3) Facilidad de entender : documento legible y (4) Trazabilidad: trazabilidad de los requerimientos generales respecto del código y de las pruebas.

2- Calidad del Producto: un objetivo importante de un proyecto de desarrollo de software es desarrollar código y documentación que se correspondan con los requerimientos del proyecto. Esta meta u objetivo tiene los siguientes atributos:

- Estructura / Arquitectura: la evaluación de un módulo para identificar posibles errores e indicar problemas potenciales en la facilidad de uso y facilidad de mantenimiento
- Reutilización: utilizar el software en diferentes contextos o aplicaciones
- Facilidad de mantenimiento: es el esfuerzo requerido para localizar y corregir un error en un programa
- Documentación: tener la adecuada documentación del código interno y la documentación externa.

3- Efectividad de la implementación: el objetivo de la efectividad de la implementación es maximizar la efectividad de los recursos dentro de las actividades programadas en el proyecto. Los atributos de este objetivo son:

- Uso del recurso: el uso del recurso relacionado a la etapa apropiada del proyecto
- Cumplimiento de los porcentajes: avances realizados en los ítems

4- Efectividad de la prueba: los objetivos de la prueba de efectividad es ubicar y reparar las fallas del software. El atributo es la corrección. Una vez generado el código, se realizan pruebas de unidades, pruebas finales y pruebas de aceptación.

### **Modelo de Dromey**

El modelo de Dromey tiene el propósito de trabajar con una estructura que permite construir y utilizar un modelo de calidad práctico para evaluar las etapas de Determinación de los requerimientos, Diseño e Implementación. Esta información puede ser usada para elaborar, comparar y evaluar la calidad de los productos de software. Este modelo plantea la calidad del producto por medio de

la definición de subcaracterísticas que pueden ser medidas y evaluadas como características. También, permite aumentar el entendimiento respecto de la relación entre los atributos (características) y los subatributos (subcaracterísticas) de calidad. Dromey propone 3 modelos para cada etapa del proceso de desarrollo: (1) modelo de requerimientos, (2) modelo de diseño y (3) modelo de calidad de la implementación.

Las características de calidad planteadas en este modelo son: Eficiencia, Confiabilidad, Facilidad de mantenimiento, Portabilidad, Facilidad de uso y Funcionalidad.

Estas características pueden ser agrupadas de acuerdo a diversos aspectos a tener en cuenta en la implementación: (1) corrección, (2) aspectos internos, (3) aspectos del contexto y (4) aspectos descriptivos. (Tabla 20)

|                |                             |                                                           |
|----------------|-----------------------------|-----------------------------------------------------------|
| Implementation | Correctness                 | Functionality, Reliability                                |
|                | Internal                    | Maintainability, Efficiency, Reliability                  |
|                | Contextual                  | Maintainability, Reliability,<br>Portability, Reliability |
|                | Description                 | Maintainability, Reusability,<br>Portability, Usability   |
| Etapas         | Propiedades del<br>Producto | Características de Calidad                                |

Fuente: DOLADO, José Javier, 2000

**Fig. 2.8. Agrupamiento de las características de calidad**

Los pasos para la aplicación del modelo de Dromey son: (1) Seleccionar el conjunto de atributos que se necesitan evaluar, (2) Realizar una lista de todos los componentes o módulos del sistema, (3) Identificar las propiedades de calidad de cada componente, (4)

Determinar cómo afecta cada propiedad en los atributos de calidad y (5) Evaluar el modelo de calidad.

### WebQEM (Web Quality Evaluation Method)

Los desarrollos centrados en la Web, en los más diversos dominios de aplicación como comercio electrónico, sistemas académicos, financieros, entre otros, se están tornando cada vez más en sistemas complejos. La complejidad en la evaluación es producto de la gran cantidad de características y atributos que pueden intervenir en los requerimientos de calidad y en las varias relaciones existentes entre los atributos, subcaracterísticas y características, entre otros aspectos.

WebQEM puede ser usada para evaluar diversos dominios de aplicación de acuerdo a los distintos puntos de vista y objetivos de evaluación. La definición y la especificación de los requerimientos de calidad son actividades esenciales en el proceso de evaluación.

Una de las metas principales de la evaluación y comparación de calidad de una Web, radica en comprender el grado de cumplimiento de un conjunto de características y subcaracterísticas con respecto a los requerimientos de calidad establecidos. Luis Olsina desarrolló la metodología WebQM, la cual plantea 4 características de calidad con sus respectivas subcaracterísticas y atributos. Las características de calidad planteadas son: (1) Facilidad de Uso, (2) Funcionalidad, (3) Confiabilidad y (4) Eficiencia.

## **1. Facilidad de Uso**

### **1.1 Comprensibilidad Global del Sitio**

#### **1.1.1 Esquema de Organización Global**

#### **1.1.2 Calidad en el Sistema de Etiquetado**

#### **1.1.3 Visita Guiada Orientada al Estudiante**

#### **1.1.4 Mapa de Imagen (Campus/Edificio)**

### **1.2 Mecanismos de Ayuda y Retroalimentación en línea**

#### **1.2.1 Calidad de la Ayuda**

#### **1.2.2 Indicador de Última Actualización**

##### **1.2.2.1 Global (de todo el sitio Web)**

#### **1.2.3 Directorio de Direcciones**

1.2.4 Facilidad FAQ

1.2.5 Retroalimentación

1.3 Aspectos de Interfaces y Estéticos

1.3.1 Cohesividad al Agrupar los Objetos de Control Principales

1.3.2 Permanencia y Estabilidad en la Presentación de los Controles Principales

1.3.3 Aspectos de Estilo.

1.3.4 Preferencia Estética

1.4 Misceláneas

1.4.1 Soporte a Lenguaje Extranjero

1.4.2 Atributo “Qué es lo Nuevo”

1.4.3 Indicador de Resolución de Pantalla

## **2. Funcionalidad**

2.1 Aspectos de Búsqueda y Recuperación

2.1.1 Mecanismo de Búsqueda en el Sitio Web

2.1.2 Mecanismos de Recuperación

2.2 Aspectos de Navegación y Exploración

2.2.1 Navegabilidad

2.2.2 Objetos de Control Navegacional

2.2.3 Predicción Navegacional

2.3 Aspectos del Dominio orientados al Estudiante

2.3.1 Relevancia de Contenido

2.3.2 Servicios On- line

## **3. Confiabilidad**

3.1 No Deficiencia

3.1.1 Errores de Enlaces

3.1.2 Errores o Deficiencias Varias

## **4. Eficiencia**

#### 4.1 Performance

##### 4.1.1 Páginas de Acceso Rápido

#### 4.2 Accesibilidad

##### 4.2.1 Accesibilidad de Información

##### 4.2.2 Accesibilidad de Ventanas

En el siguiente cuadro (Figura 28) se muestran tres tipos de plantillas: para el componente característica, para el componente subcaracterística, y para el atributo (elemento de más bajo nivel en la jerarquía).

|                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                              |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Título:</b> Código: Tipo:<br>Característica Sub-característica/s<br>(Código/s): Definición / Comentarios:<br>Modelo para determinar el Cómputo Global                                                                                                                                                                         | <b>Título:</b> Código: Tipo:<br>Atributo Característica de más Alto Nivel<br>(Código): Super-característica (Código):<br>Definición / Comentarios: Tipo de Criterio                                          |
| Herramienta Empleada: Peso: Operador<br>Aritmético / Lógico: Ejemplo/s: Valor/es de<br>Preferencia/s Computado/s:                                                                                                                                                                                                                | Elemental: Plantilla de Referencia de<br>Variables y Parámetros: Escala de<br>Preferencia: Tipo de Recolección de Datos:<br>Herramienta Empleada: Ejemplo/s: Peso:<br>Valor/es de Preferencia/s Computado/s: |
| <b>Título:</b> Código: Tipo:<br>Subcaracterística Super-característica (Código): Sub-característica/s (Código/s): Atributo/s<br>(Código/s): Definición / Comentarios: Modelo para determinar el Cómputo Parcial:<br>Herramienta Empleada: Peso: Operador Aritmético/Lógico: Ejemplo/s: Valor/es de<br>Preferencia/s Computado/s: |                                                                                                                                                                                                              |

Fuente: DOLADO, José Javier, 2000

**Fig.2.9. Plantillas para el componente características, subcaracterística y atributo.**

Dado un proceso de evaluación y en consideración de un dominio Web, podríamos ver fácilmente que no necesariamente todos los atributos deben existir simultáneamente; podría ser necesario un Mapa del Sitio, o una Tabla de Contenidos, o un Índice.

## 2.6.2 ESTÁNDARES DE CALIDAD DEL SOFTWARE A NIVEL PRODUCTO.

### ISO/IEC 9126-1:2001 – Quality Model

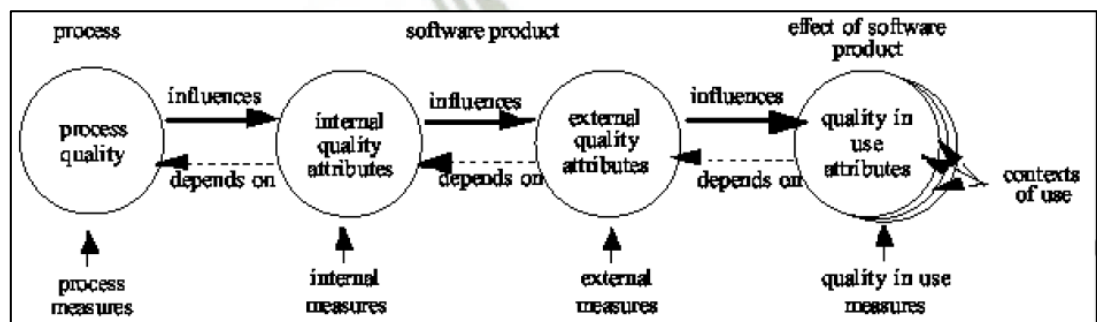
Esta parte de la ISO 9126 describe el modelo de calidad del producto de software. La primera parte del modelo especifica 6 características de calidad interna y externa, las cuales están divididas en subcaracterísticas, son manifestadas externamente cuando el software es utilizado como parte de un sistema, y son un resultado de atributos internos del software.

La calidad externa evalúa que el software satisfaga las necesidades del usuario teniendo en cuenta las condiciones especificadas.<sup>29</sup> Esta calidad es medible en el comportamiento del producto. La calidad interna evalúa el total de atributos que un software debe satisfacer teniendo en cuenta condiciones especificadas.<sup>30</sup> Esta calidad es medible a partir de las características intrínsecas.

Las características definidas son aplicables a todo tipo de software. Las características y subcaracterísticas proveen una terminología consistente respecto de la calidad del producto del software.

Esta Norma permite especificar y evaluar la calidad del software desde distintas perspectivas, las cuales están asociadas a la adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad, y auditoria del software.

Puede ser usada por desarrolladores, evaluadores independientes y grupos de aseguramiento de la calidad, responsables de especificar y evaluar la calidad del software.



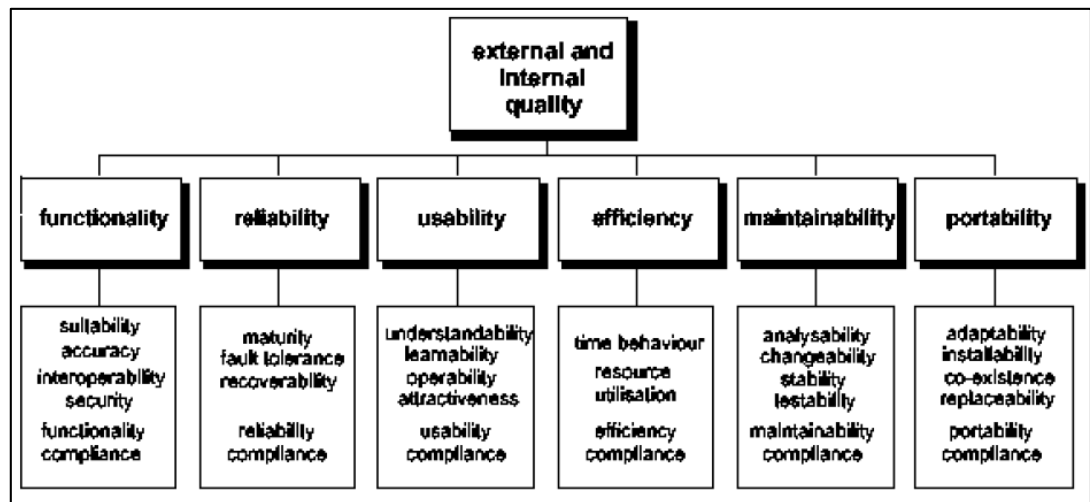
Fuente: DOLADO, José Javier, 2000

**Fig. 2.10. Calidad en el ciclo de vida según ISO/IEC 9126-1**

La evaluación de los productos de software que satisfacen las necesidades de calidad del software es uno de los procesos del ciclo de vida de desarrollo del software. La calidad del producto de software puede ser evaluada por medio de la medición de atributos internos, externos o a través de la calidad en uso. El objetivo es que el producto tenga el efecto requerido en un contexto particular de uso.

La calidad del proceso contribuye a mejorar la calidad del producto, y la calidad del producto contribuye a mejorar la calidad en uso. Evaluar y mejorar la calidad de un proceso contribuye a mejorar la calidad del producto; y esto contribuye a mejorar la calidad en uso. De manera similar, evaluar la calidad en uso puede mejorar la calidad del producto, y evaluar un producto puede mejorar un proceso. Las necesidades de calidad del usuario contribuyen a especificar los “requisitos de calidad externa”, los cuales contribuyen a especificar los “requisitos de calidad interna”. La “calidad interna” indica la existencia de “calidad externa” y ésta indica la existencia de “calidad en uso”. El modelo de calidad de ISO 9126-1 establece 3 niveles: (1) Característica, (2) Subcaracterística y (3) Métricas.

Existen métricas internas y externas. Las métricas internas pueden ser aplicadas a un software no ejecutable durante el diseño y la codificación. Las métricas externas se utilizan en el software ejecutable. El modelo de calidad interna y externa está formado por las siguientes características: (1) Funcionalidad, (2) Confiabilidad, (3) Facilidad de uso, (4) Eficiencia, (5) Facilidad de mantenimiento y (6) Portabilidad.



Fuente: DOLADO, José Javier, 2000

**Fig. 2.11. Modelo de calidad interna y externa de ISO/IEC 9126-1**

### Partes de ISO 9126

ISO/IEC 9126 está formada por las siguientes partes:

- Parte 1 – Modelo de Calidad
- Parte 2 – Métricas Externas
- Parte 3 – Métricas Internas
- Parte 4 – Calidad en Uso

### ISO/IEC TR 9126-2:2003 – Métricas Externas

ISO/IEC TR 9126-2:2003 provee métricas externas para la medición de atributos a través de 6 características de calidad externa definidas en ISO/IEC 9126-1.

ISO/IEC TR 9126-2:2003 define métricas externas., ISO/IEC TR 9126-3 define métricas internas e ISO/IEC 9126-4 define métricas en calidad en uso para la medición de características o subcaracterísticas.

Las métricas internas miden el software en sí mismo, las métricas externas miden el comportamiento del sistema basado en computadora que incluye el software, y las métricas de calidad en uso miden los efectos de uso del software en un contexto de uso específico.

Las métricas externas usan medidas de un software, derivadas del comportamiento del mismo, a través de la prueba, operación y observación del software. Antes de adquirir o usar un software, éste debe ser evaluado usando las métricas basadas en los objetivos del área usuaria de la institución relacionados al uso, explotación y dirección del producto, considerando la organización y el ambiente técnico. La métrica externa proporciona a los usuarios, evaluadores, verificadores y desarrolladores, el beneficio que puedan evaluar la calidad del software durante las pruebas o el funcionamiento.

Las métricas listadas en ISO/IEC TR 9126-2:2003 no están destinadas a ser un conjunto exhaustivo. Los desarrolladores, evaluadores, gerentes de calidad y compradores pueden seleccionar métricas de ISO/IEC TR 9126-2:2003 para definir requerimientos, evaluar productos de software, medir aspectos de calidad y otros propósitos.

Los usuarios de ISO/IEC TR 9126-2:2003 pueden seleccionar o modificar y aplicar métricas y mediciones a partir de ISO/IEC TR 9126-2:2003 o pueden definir métricas

específicas de la aplicación para su dominio de aplicación en particular.

ISO/IEC TR 9126-2:2003 es usada junto con ISO/IEC 9126-1.

ISO/IEC TR 9126-2:2003 contiene una explicación de cómo aplicar las métricas de calidad del software, un conjunto básico de métricas para cada Subcaracterística y un ejemplo de cómo aplicar las métricas durante el ciclo de vida del producto de software. ISO/IEC TR 9126-2:2003 no asigna un rango de valores para estas métricas en niveles de porcentajes o grados de conformidad, debido a que estos valores son definidos en cada producto de software o en una parte del producto de software, dependiendo de factores tales como la categoría del software, nivel de integridad y necesidades del usuario.

Algunos atributos pueden tener un rango de valores deseable, el cual no depende de las necesidades específicas del usuario pero si depende de factores genéricos; por ejemplo factores humanos.

### **ISO/IEC TR 9126-3:2003 – Métricas Internas**

ISO/IEC TR 9126-3:2003 provee métricas internas para la medición de atributos a través de 6 características de calidad interna definidas en ISO/IEC 9126-1.

La métrica interna mide el software en sí mismo y puede ser aplicada a un software no ejecutable (como una especificación o código fuente) durante el diseño y la codificación.

En el desarrollo de un software, los productos intermedios deben ser evaluados usando métricas internas que permitan medir las propiedades intrínsecas, incluyendo aquellas que pueden derivarse de comportamientos simulados. El propósito principal de esta métrica interna es asegurar que se logre la calidad externa y la calidad de uso requerida. La métrica interna proporciona a los usuarios, evaluadores, verificadores y desarrolladores el beneficio que puedan evaluar la calidad del software y lo referido a problemas de calidad antes que el software sea puesto en ejecución.

Las métricas internas miden atributos internos o indican los atributos externos, a través del análisis de las propiedades estáticas de productos intermedios o entregables del software.

Las medidas de las métricas internas usan números o frecuencias de elementos de composición de software, los cuales aparecen, por ejemplo, en las sentencias de código de fuente, control de gráficos, flujo de datos y estados de representación de procesos. Las métricas listadas en ISO/IEC TR 9126-3:2003 no están destinadas a ser un conjunto exhaustivo. Los desarrolladores, evaluadores, gerentes de calidad y compradores pueden seleccionar métricas de ISO/IEC TR 9126-2:2003 para definir requerimientos, evaluar productos de software, medir aspectos de calidad y otros propósitos.

ISO/IEC TR 9126-3:2003 es usada junto con ISO/IEC 9126-1.

ISO/IEC TR 9126-3:2003 contiene una explicación de cómo aplicar las métricas de calidad del software, un conjunto básico de métricas para cada Subcaracterística y un ejemplo de cómo aplicar las métricas durante el ciclo de vida del producto de software.

ISO/IEC TR 9126-3:2003 no asigna un rango de valores para estas métricas en niveles de porcentajes o grados de conformidad, debido a que estos valores son definidos en cada producto de software o en una parte del producto de software, dependiendo de factores tales como la categoría del software, nivel de integridad y necesidades del usuario.

Algunos atributos pueden tener un rango de valores deseable, el cual no depende de las necesidades específicas del usuario pero si depende de factores genéricos; por ejemplo factores humanos.

### **Relación entre las métricas internas y externas**

Cuando los requisitos de calidad del software son definidos, se listan las características o subcaracterísticas de calidad del software que contribuyen a dichos requisitos. Entonces, las métricas externas apropiadas y los rangos aceptables son especificados para cuantificar el criterio de calidad que valida que el software satisface las necesidades del usuario.

Luego, los atributos de calidad interna del software se definen y especifican para planear y finalmente lograr la calidad externa y calidad en el uso requeridas, para construirlos durante el desarrollo del producto.

Apropiadas métricas internas y rangos aceptables son especificados para cuantificar los atributos de calidad interna, así ellos pueden usarse para verificar que el software intermedio reúne las especificaciones de calidad interna durante el desarrollo. Se recomienda que las métricas internas que se usen tengan en lo posible una fuerte relación con la métrica externa diseñada, para que ellas puedan ser usadas para predecir los valores de las métricas externas. Sin embargo, es generalmente difícil diseñar un modelo teórico riguroso que proporcione una relación fuerte entre la métrica interna y la externa.

### **ISO/IEC TR 9126-4:2004 – Calidad en Uso**

ISO/IEC TR 9126-4:2004 provee métricas para la calidad en uso para la medición de los atributos definidos en ISO/IEC 9126-1.

Las métricas de calidad en uso miden los efectos de uso del software en un contexto específico de uso. Estas métricas miden si el producto se corresponde con las necesidades específicas de los usuarios para así obtener los objetivos específicos con eficiencia, productividad, seguridad y satisfacción en un contexto de uso específico. Esto solo es llevado a cabo en un ambiente de sistema realista. Las métricas listadas en ISO/IEC TR 9126-4:2004 no están destinadas a ser un conjunto exhaustivo. Los desarrolladores, evaluadores, gerentes de calidad y compradores pueden seleccionar métricas de ISO/IEC TR 9126-4:2004 para definir requerimientos, evaluar productos de software, medir aspectos de calidad y otros propósitos.

ISO/IEC TR 9126-4:2004 es usada junto con ISO/IEC 9126-1.

ISO/IEC TR 9126-4:2004 contiene una explicación de cómo aplicar las métricas de calidad del software, un conjunto básico de métricas para cada característica y un ejemplo de cómo aplicar las métricas durante el ciclo de vida del producto de software.

### **ISO/IEC 25000:2005 - SQuaRE**

SQuaRE (Software Quality Requirements and Evaluation) es una nueva serie de normas que se basa en ISO 9126 y en ISO 14598 (Evaluación del software). Uno de los principales objetivos de la serie SQuaRE es la coordinación y armonización del contenido de ISO 9126 y de ISO 15939:2002 (Measurement Information Model). ISO 15939 tiene un modelo de información que ayuda a determinar que se debe especificar durante la planificación, performance y evaluación de la medición, Para su aplicación, cuenta con los siguientes pasos: (1) Recopilar los datos, (2) Preparación de los datos y (3) Análisis de los datos.

La integración de ISO 9126 e ISO 15939 permiten plantear un proceso de 4 pasos: (1) Identificación de los requerimientos relacionados a la calidad del producto, es decir, seleccionar la parte del modelo de calidad (ISO/IEC 9126-n)

que resulta relevante para la evaluación de calidad. (2) Identificación del contexto de interpretación. Es decir, selección de los valores de referencia y determinación de los target especificados en un contexto determinado (3) Uso de las medidas derivadas de la etapa de preparación de los datos (4) Comparación y análisis de los resultados obtenidos respecto de un conjunto de valores de referencia.

SQuaRE incluye un estándar de requerimientos de calidad. Está compuesto por 14 documentos agrupados en 5 tópicos: (1) Administración de la Calidad – 2500n, (2) Modelo de Calidad – 2501n, (3) Medidas de Calidad – 2502n, (4) Requerimientos de Calidad – 2503n y (5) Evaluación de la Calidad – 2504n.

**Administración de la Calidad (1):** abarca

- a) Guía para SquaRE – Overview de la estructura y terminología
- b) Planificación y Administración – Provee una guía para planificar y administrar las evaluaciones del software

**Modelo de Calidad (2):** describe el modelo de calidad interno / externo y la calidad en uso. Presenta características y subcaracterísticas.

**Medidas de Calidad (3):** Medición de primitivas, Medidas para la calidad interna,

Medidas para la calidad externa y Medidas para la calidad en uso.

**Requerimientos de Calidad (4):** permite habilitar la calidad del software a ser especificado en términos de requerimientos de calidad durante todo el ciclo de vida de un proyecto de software o adquisición, mantenimiento y operación.

**Evaluación de la Calidad (5):** Evaluación de la Calidad, Proceso para desarrolladores, Proceso para compradores, Proceso para evaluadores y Documentación del módulo de evaluación.

Estos 5 tópicos conforman la Arquitectura de SQuaRE (Figura 38).

Los beneficios de utilizar SQuare son: (1) El modelo representa la calidad esperada del producto de software, (2) Planteo del desdoblamiento de las necesidades o expectativas en calidad en uso, calidad externa y calidad interna, (3) Permite una mayor eficacia en la definición del software, (4) Plantea la evaluación de productos intermedios, (5) Propone una calidad final a través de las evaluaciones intermedias, (6) Permite efectuar un rastreo entre las expectativas, requisitos y medidas de evaluación; y (7) Mejora la calidad del producto.



## CAPITULO III

### INGENIERÍA DEL PROYECTO

#### 3.1 JUSTIFICACION DE LA HERRAMIENTAS ELEGIDAS

Puesto que la aplicación está basada en Web Se hace necesario justificar las herramientas elegidas para cada elemento de la aplicación como son:

- El servidor Web.
- El DBMS.
- Los lenguajes de programación tanto del lado del servidor como del lado del cliente.
- Los navegadores para soportar la aplicación.

Es importante establecer que para el desarrollo de una aplicación es necesario hacerlo desde dos perspectivas diferentes: del lado del Front-End y del lado del Back-End; como un tipo de abstracciones que ayudarán a mantener las diferentes partes del sistema en forma separada.

El *Front-End* es la visualización del usuario navegante, o sea, la parte del sitio que interactúa directamente con el usuario. Y el *Back-End* el área de administración en donde se ingresa, edita y organiza el contenido, es la parte del administrador del sitio con sus respectivos sistemas.

##### 3.1.1 HERRAMIENTAS PARA EL FRONT-END

Las principales herramientas a utilizar para el Front-End son los lenguajes JavaScript y PHP. El lenguaje que se ejecuta en el lado del cliente es JavaScript. JavaScript no es la única alternativa para obtener las mismas funcionalidades requeridas.

**JavaScript.** Porque permite insertar código especial dentro del HTML de una página, su función es ampliar las posibilidades de HTML, JavaScript no crea programas independientes, dependen por completo del código HTML de la página.

Por ser un lenguaje basado en objetos sin tipo y liviano utilizado para acceder a objetos en aplicaciones. Principalmente, integrado en un navegador Web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas Web dinámicas, caracterizado por ser un lenguaje basado en prototipos.

Además una de las ventajas fundamentales de JavaScript es que su aprendizaje y uso son muy sencillos además de que permite realizar labores complejas en una página.

Acerca de los lenguajes de programación del lado del servidor existe una amplia gama de posibilidades: ASP.Net, ASP clásico y PHP. En la tabla 3.1.1 se muestra una tabla comparativa de las principales características de los dos lenguajes más usados en el mercado.

Es importante resaltar que ASP es considerado como una tecnología de aplicación Web que funciona solamente para una infraestructura Microsoft y no es considerado meramente como un lenguaje de programación.

| CARACTERISTICAS | PHP                                                                                                                                                                                                                         | ASP                                                                                                                                                                            |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Seguridad       | Instalado sobre servidores Unix o Linux, es más veloz y seguro. PHP permite configurar el servidor de modo que se permita o rechacen diferentes usos, lo que puede hacer al lenguaje seguro dependiendo de las necesidades. | Debido a que solo funciona bajo Windows, es más vulnerable a sufrir caídas e infecciones de virus                                                                              |
| Rendimiento     | En el caso de estar montado sobre un servidor Unix o Linux, es más rápido dado que se ejecuta en un único espacio de memoria.                                                                                               | Son demasiadas las comunicaciones entre componentes COM que se realizan entre todas las tecnologías implicadas en una página ASP. Consumiendo así más recursos.                |
| Plataforma      | Independiente de la plataforma y servidor (Unix, Windows, Linux)                                                                                                                                                            | Se trata de un sistema propietario que es usado nativamente sólo por Microsoft Internet Information Server (IIS). Esto limita su disponibilidad a servidores basados en Win32. |
| Soporte         | Recibe contribuciones de diversos desarrolladores                                                                                                                                                                           | ASP es específica de Microsoft que desarrolla sus procesos internamente.                                                                                                       |

Fuente: WELLING THOMSON –LAURA. 2009.

**Tabla 3.1.1 Comparación entre PHP vs ASP**

### 3.1.2 HERRAMIENTAS PARA EL BACK-END

Las herramientas utilizadas para el Back-End son las que tienen que ver con las bases de datos y los SGBD, en el presente trabajo se utiliza un SGBD o DBMS llamado MySQL.

Existen varias alternativas DBMS que pueden emplearse para la administración de las bases de datos. Pero se eligió MySQL porque es considerada una BD muy depurada, es plataforma propietaria, es decir, puede usarse obligatoriamente sobre Windows, se puede descargar una versión de prueba por un periodo muy corto y todo se ejecuta en modo gráfico, es decir, solo se tiene que aprender a utilizar su entorno gráfico para usarlo por lo que no es la forma correcta de aprender. Es una de las más usadas en sitios Web (más que todo) porque es muy liviana y rápida en sus accesos. MySQL: ofrece velocidad consume muy pocos recursos, tanto de CPU como de memoria.

#### **Funcionamiento en diferentes plataformas.**

Se compone de un servidor que es multihilo y multiusuario, lo cual quiere decir que nos permite llevar a cabo varios trabajos en un solo momento.

| VENTAJAS                                                                                                                                                                                                                      |                                                                                                                                                                                 |                                                                                                                                                                                           |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PostgreSQL                                                                                                                                                                                                                    | MySQL                                                                                                                                                                           | FireBird                                                                                                                                                                                  |
| Gran escalabilidad. Capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correct         | Lo mejor de MySQL es su velocidad a la hora de realizar las operaciones, lo que le hace uno de los gestores que ofrecen mayor rendimiento                                       | Ofrece una gran variedad de medios para acceder a los datos (nativo/API, driver dbExpress, ODBC, OLEDB, .NET provider, driver JDBC nativo de tipo 4, módulo para Python, PHP, Perl, etc.) |
| Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle. | Las utilidades de administración de este gestor son envidiables para muchos de los gestores comerciales existentes, debido a su gran facilidad de configuración e instalación.  | Firebird administra la concurrencia en las actualizaciones en los datos, así como el manejo de las transacciones.                                                                         |
|                                                                                                                                                                                                                               | Tiene una probabilidad muy reducida de corromper los datos, incluso en los casos en los que los errores no se produzcan en el propio gestor, sino en el sistema en el que está. | Firebird ofrece una concurrencia excelente, alto rendimiento y un poderoso lenguaje de procedimientos                                                                                     |
|                                                                                                                                                                                                                               | El conjunto de aplicaciones Apache-PHP-MySQL es uno de los más utilizados en Internet en servicios de foro (Barrapunto.com) y de buscadores de aplicaciones (Freshmeat.net).    |                                                                                                                                                                                           |

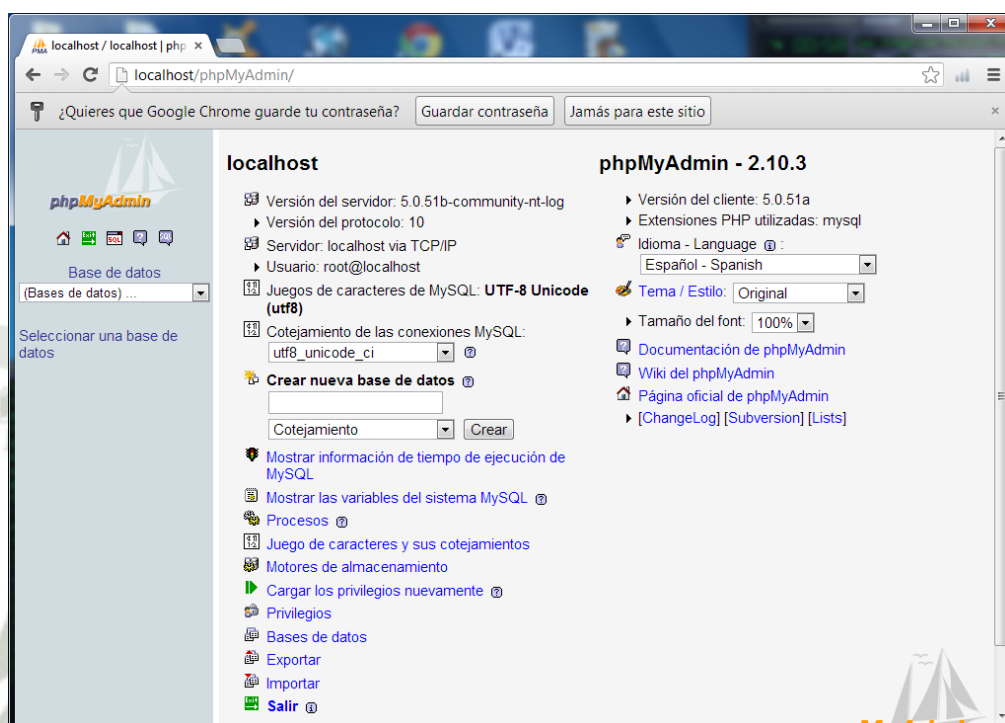
**Fuente: WELLING THOMSON –LAURA. 2009.**

**Tabla.** Comparativa de las desventajas entre PostgreSQL, MySQL y FireBird

Para una elección adecuada del SGBD es importante tener en cuenta la compatibilidad y rendimiento. Como una herramienta adicional para el manejo de MySQL se usa phpMyadmin

La herramienta phpMyAdmin es una herramienta escrita en PHP cuyo objetivo es manejar la administración de MySQL a través de páginas Web, utilizando Internet. Actualmente phpMyAdmin puede crear y eliminar Bases de Datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios, exportar

datos en varios formatos y está disponible en 50 idiomas. Se encuentra disponible bajo la licencia GPL. Una herramienta similar a phpMyAdmin es phpPgAdmin, la cual ofrece funcionalidades similares pero está diseñado para PostgreSQL.



Fuente: Elaboración Propia

**Figura 3.1.1** Interfaz de la herramienta phpMyadmin para el manejo de la administración de MySQL.

En cuanto al servidor Web, los más utilizados mundialmente son: el servidor Web Apache, IIS de Microsoft y el relativamente reciente servidor Web Cherokee.

**Servidor http Apache.** Tiene amplia aceptación en la red, ya que es el más utilizado para la creación de sitios Web, debido a que brinda ventajas como: velocidad, seguridad, simplicidad y facilidad de desarrollo distribuido, además de ser rápido, estable y fácilmente adaptable a nuevas tecnologías y protocolos. Apache es un servidor altamente configurable ya que su arquitectura está

formada por un diseño modular, además puede trabajar con la mayoría de lenguajes de programación como, PHP, JSP, etc. y otros lenguajes script. Apache soporta HTTPS, haciendo de esta manera más seguro el servidor Web, ya que permite mantener un canal de comunicación encriptado utilizando SSL (Secure Socket Layers) con lo cual, no se permite el robo de datos ó información que se está transfiriendo entre los usuarios, puesto que, permite identificar verdaderamente al servidor y, en algunos casos, al cliente. Lo que lo hace mucho más seguro.

A continuación se muestra una tabla comparativa de las principales características de los servidores Web mencionados.

|                                                | Cherokee                                                                      | Apache                                                                        | IIS                                                 |
|------------------------------------------------|-------------------------------------------------------------------------------|-------------------------------------------------------------------------------|-----------------------------------------------------|
| Desarrollador                                  | Cherokee Project                                                              | Apache Software Foundation                                                    | Microsoft                                           |
| Sitio Web oficial                              | <a href="http://www.cherokeeproject.com/">http://www.cherokeeproject.com/</a> | <a href="http://www.cherokeeproject.com/">http://www.cherokeeproject.com/</a> | <a href="http://www.iis.net">http://www.iis.net</a> |
| Sistema operativo                              | Multiplataforma                                                               | Multiplataforma                                                               | Microsoft Windows                                   |
| Soporte de múltiples lenguajes de programación | SI                                                                            | SI                                                                            | Pobre                                               |
| Conexiones seguras                             | SI                                                                            | SI                                                                            | SI                                                  |
| Mayor velocidad                                | SI                                                                            | SI                                                                            | No                                                  |
| Mayor seguridad                                | Pobre                                                                         | SI                                                                            | Pobre                                               |

Fuente: WELLING THOMSON –LAURA. 2009.

**Tabla** Comparativa de los principales servidores Web

Dentro del conjunto de herramientas que se utilizan para una aplicación basada en AJAX existe también un competidor que ha alcanzado un gran mercado, Flash. Pero por ser un software propietario y por las características mostradas en la tabla siguiente, se opta por una programación basada en AJAX.

| Función                                 | Flash                                                                                                                                         | Ajax                                                                                                                                                                |
|-----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Audio                                   | Dinámicamente la carga de audio. Soporte flash integrado de audio.                                                                            | Sólo se admite a través de plug-ins externos (como el Media Player).                                                                                                |
| Navegador de Integración                | Flash Player plug-in. Flash se limita a un área rectangular predefinidos del navegador.                                                       | JavaScript de forma nativa con el apoyo de los navegadores modernos. Simple interacción con cualquier parte del navegador.                                          |
| Problemas de Compatibilidad             | Puede haber variaciones menores entre las versiones de Flash.                                                                                 | Principales diferencias de compatibilidad entre versiones de navegadores.                                                                                           |
| CSS                                     | Soporte limitado.                                                                                                                             | Soporte completo (según el navegador).                                                                                                                              |
| Dinámica generación de contenidos       | Difícil. SWF es un formato de pre-compilados cerrado. (Comparación de SVG, el cual está basado en XML.)                                       | HTML puede ser escrito usando casi cualquier tipo de tecnología de servidor. Es más fácil la generación de imágenes dinámicas.                                      |
| Modelo de Programación                  | Ofrece ActionScript 2.0, en Java, como marco sólido.                                                                                          | JavaScript 2.0 pero no con el apoyo De <i>cualquier</i> navegador principal. JS 1.5 no se recomienda para grandes aplicaciones de programación orientada a objetos. |
| De mapa de bits (mapa de bits) Gráficos | Carga estática de las imágenes de forma dinámica. Apoyo a JPG, GIF y PNG. La manipulación de mapa de bits.                                    | Carga estática de las imágenes de forma dinámica.                                                                                                                   |
| Las expresiones Regulares               | No se admite de forma nativa por AS 2.0, pero de código abierto de soluciones están disponibles.                                              | Soporte completo.                                                                                                                                                   |
| Integration Server                      | Muchas de las soluciones disponibles. Flash puede comunicarse con ASP, PHP, ASPX y muchos otros tipos de secuencias de comandos del servidor. | Limitada. Se pueden comunicar con el servidor de forma dinámica con truco IFRAME o Objeto XML HttpRequest.                                                          |
| Video                                   | Carga dinámica de archivos de vídeo FLV o reproducción de videos incrustados.                                                                 | Sólo se admite a través de plug-ins externos (como el Media Player). Múltiples Formatos de vídeo se pueden cargar.                                                  |

Fuente: COLOMES FORMOS, XAVIER. (2009).

Tabla Comparativa AJAX vs FLASH

Cada opción tiene sus ventajas y desventajas, no puede decirse que una es mejor que la otra ya que depende de la solución que está en desarrollo, pero hay un área donde AJAX es superior en cuanto a aplicaciones pesadas de texto generadas dinámicamente y en el momento, estas dominan la Web. Por tanto si se desea desarrollar un programa de correo electrónico, o un organizador personal en línea, o una tienda online, flash no tiene mucho sentido, no da la misma flexibilidad para el análisis y la visualización de grandes cantidades de datos de texto. Sin necesidad de escribir demasiado código, simplemente no hay suficiente flexibilidad para reposicionar o cambiar el tamaño de los campos de texto.

Una vez que se analizaron las opciones tecnológicas para implementar el sistema se determinó que MySQL junto con Apache y PHP forman un buen equipo. Respecto de los manejadores de base de datos, MySQL resulta la mejor opción ya que no consume demasiados recursos, es más seguro y tiene un mejor rendimiento.

En cuanto a los lenguajes de programación PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix, PHP soporta la mayoría de servidores Web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, y muchos otros. PHP tiene módulos disponibles para la mayoría de los servidores.

Con PHP no se encuentra limitado a resultados HTML. Entre las habilidades de PHP se incluyen: creación de imágenes, archivos PDF y películas Flash (usando libswf y Ming) sobre la marcha. También puede presentar otros resultados, como XHTML y archivos XML. PHP puede autogenerar estos archivos y almacenarlos en el sistema de archivos en vez de presentarlos en la pantalla. La característica más potente y destacable de PHP es su soporte para una gran cantidad de bases

de datos. Escribir la interfaz vía Web para una base de datos es una tarea simple con PHP. PHP también cuenta con soporte para comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (en Windows) y muchos otros. También se pueden crear sockets puros. PHP soporta WDDX para el intercambio de datos entre lenguajes de programación en Web, puede utilizar objetos Java de forma transparente como objetos PHP y la extensión de CORBA puede ser utilizada para acceder a objetos remotos.

En cuanto al lenguaje que se ejecuta del lado del cliente debemos considerar cual es más compatible con los navegadores de Internet más utilizados. Debido a que Javascript es compatible con más navegadores, como son Internet Explorer, Netscape, Opera, entre otros, es por ello que se utiliza JavaScript como lenguaje para ejecutarse del lado del cliente Aunque muchos de los lenguajes pudieran funcionar bien con AJAX, PHP tiene muchas similitudes con JavaScript (que es el lenguaje en el que se basa AJAX) tanto de funcionalidad como de disposición del código y de concepto. Esto retribuye en un mejor funcionamiento con JavaScript, que lo hace más fiable.

JavaScript, es una tecnología abierta, especialmente porque es visible para el usuario. Existe una gran comunidad de desarrolladores de JavaScript que poseen objetivos y formas de pensar muy parecidas a los de PHP. JavaScript es también razonablemente fiable y también comparte características con los lenguajes orientados a objetos.

Entonces al combinar estos dos lenguajes, que se encuentran en una etapa de madurez considerable, obtenemos lo mejor de dos grandes herramientas.

### 3.2 POSIBLES MÓDULOS DE APLICACIÓN

A continuación se presenta una descripción de los posibles módulos a desarrollar con base en los requerimientos.

El sitio Web cuenta con varias secciones y cada una de ellas posee sus propios módulos.

**Primer** módulo: Tienda en línea

**Segundo** módulo: Cotización de productos.

**Tercer** módulo: Imágenes de los productos y servicios de la PYME que pueden venderse.

**Cuarto** módulo: Presentación de la empresa Panadería “Carmencita”. Información de contacto.

Se desarrolla cada módulo de la siguiente manera:

**Primer módulo: Tienda en línea**

Este primer módulo contiene tres submódulos principales (Ver figura ).

|          |                                             |                |
|----------|---------------------------------------------|----------------|
| Catalogo | Carro de compras y procesamiento de pedidos | Administración |
|----------|---------------------------------------------|----------------|

**Figura 3.2.1** Módulos principales para la Tienda en línea

Además el sistema de venta en línea contará con dos tipos de vistas básicas: la vista de usuario y la vista del administrador.

Analizando la vista básica de usuario, en primer lugar el cliente accede a la página principal en la que se enumeran todas las categorías de los productos del sitio, desde aquí, puede acceder a una categoría determinada y después a los detalles del producto. Con una opción visible para añadir un producto en concreto al carro.

Vista básica de administrador. En el diseño del flujo del sistema para la vista de administrador, se permitirá al administrador registrar y añadir categorías y productos. (Ver figura)

### **Segundo módulo: Cotización de productos.**

Es un módulo de prueba para la empresa en la que el usuario podrá obtener una cotización en tiempo real de productos en stock.

Primer submódulo: Redimensionamiento visual

Segundo submódulo: Calculo de la cotización según la redimensión de la imagen (guía de cálculo proporcionada por la empresa)

### **Tercer módulo: Galería de productos**

Es necesario contar con un mecanismo apropiado para la promoción de estos productos. Se hace con una galería de imágenes.

Primer submódulo: Obtención de las categorías de los productos de una base de datos.

Segundo submódulo: Presentación de las imágenes categorizadas.

### **Cuarto módulo: Presentación de la empresa Panadería “Carmencita”**

En esta sección se reúne y se pone a disposición toda la información pertinente y necesaria para la presentación de la empresa, así como sus datos y formularios de contacto para una posible consultoría o cotización.

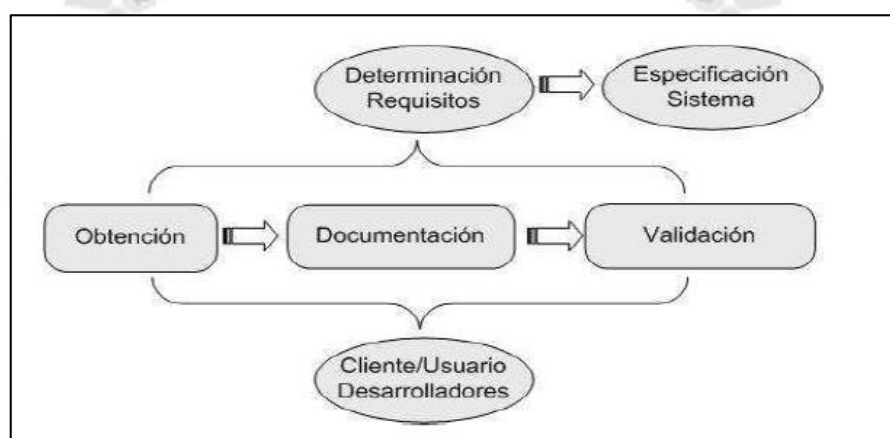
## **3.3 REQUISITOS**

### **Determinación de los Requisitos**

Para poder determinar los requisitos es necesario seguir una serie de pasos para una correcta elaboración de estos. Primeramente es necesaria la obtención de estos, la forma habitual es mediante el encuentro de interlocución con los usuarios o clientes mediante entrevistas, intercambio de opiniones, lluvia de ideas, algún cuestionario, etc., que puedan ser útiles en la redacción de un documento llamado “especificación de requisitos”. En este documento se integran además una definición de términos descriptivos para quienes se encargarán de la programación así como un esbozo de especificación previo al diseño completo. De esta manera quienes participan en la

elaboración del documento son los representantes del cliente, representantes del desarrollador, diseñadores quienes usarán los requisitos como base del desarrollo, así como el equipo de pruebas quienes asegurarán que el sistema cumpla con los requisitos. Para el presente trabajo se pudieron de igual manera obtener los requisitos mediante la interlocución con el representante directo de la Panadería “Carmencita”. Logrado en varias sesiones.

Después de la obtención de los requisitos se documenta para finalmente hacer una validación de estos (ver figura 3.3.1).



Fuente: DOLADO, JOSÉ JAVIER, 2000.

**Figura 3.3.1** Ciclo de la determinación de Requisitos

## **Tipos de requisitos**

### **Requisitos funcionales:**

Este tipo de requisitos presenta lo siguiente: Un listado completo de las cosas que el cliente espera que haga el sistema propuesto para todas las categorías de los usuarios, redactado de forma que el cliente pueda entenderlo (con claridad, conciso y congruente), generado entre cliente y desarrollador como documento de compromiso. Contiene el propósito general del sistema con referencias a otros sistemas relacionados y un diccionario de términos y abreviaturas que puedan ser útiles. Tiene una descripción de antecedentes y objetivos del desarrollo. Requerimientos sobre soporte, seguridad y privacidad y cualquier restricción.

Estos requisitos se obtuvieron mediante el planteamiento de preguntas que ayuden a determinar cuales son las características y funciones que debe cumplir el objeto para satisfacer las necesidades y las expectativas del usuario para que el producto sea utilizado, sirva y guste a quienes está dirigido, en el sentido de acotar con precisión las especificaciones del objeto por diseñar.

Dos variables de estos requisitos que resultan importantes se consideraron para el presente trabajo de aplicación Web:

*Diseño:* Para conseguir un diseño atractivo en la página Web del presente trabajo se realiza mediante programas específicos como Dreamweaver.

*Usabilidad:* Uno de los módulos a desarrollar es una plataforma de comercio electrónico por tanto es fundamental que los usuarios de la aplicación no se pierdan entre contenidos y productos y en consecuencia decidan no volver a la página Web. Por tanto se procuró la sencillez y la intuición en la página, para que el usuario sepa en todo momento donde se encuentra y pueda tener el control sobre la navegación que realiza.

Para los administradores del sitio el sistema deberá ser capaz de: Contar con claves de acceso únicas para cada usuario autorizado que opere en el área administrativa del sistema.

El usuario administrador podrá acceder sin problemas desde cualquier navegador, para realizar cada una de las actividades que el sistema propone. El administrador podrá tener acceso a una lista de módulos con sus propiedades.

Para cualquier tipo de usuario:

- Cualquier tipo de usuario podrá acceder al sitio desde cualquier navegador de Internet comercial.
- El usuario podrá tener acceso fácilmente a todas las áreas permitidas.

- Se requiere una forma de conectar la base de datos de los productos y sus categorías al explorador del usuario para que los usuarios puedan examinarlos artículos por categoría.
- El usuario podrá ver una galería de imágenes de los productos en venta, así como los detalles de estos.
- La página deberá contar con un apartado para la presentación formal de la PYME.
- Se requerirá una sección de contacto.
- El sistema será capaz de presentar al usuario un apartado para la solicitud de una cotización especial.
- 

### **Requisitos no funcionales**

Para el presente trabajo los requisitos no funcionales que se determinaron son los siguientes:

- Como regla comercial, todas las cantidades de los productos deberán ser siempre positivas. El sistema deberá ser capaz de evitar que el usuario intente establecer una cantidad negativa en la base de datos de información de productos, siendo rechazada esta opción por la lógica de control.
- La interfaz del usuario deberá realizarse en un ambiente gráfico e intuitivo para evitar inconvenientes en el uso por parte de los usuarios y para que personas con una mínima capacitación sean capaces de utilizarlo.
- El sistema debe ser capaz de manejar variables de sesión para que el administrador pueda acceder a las aplicaciones que este requiera dentro del sitio sin perder su autenticación.
- La base de datos se ubicará en un servidor central.
- El sistema deberá garantizar los servicios de seguridad de la información como son la confidencialidad, integridad y disponibilidad.
- Mediante sentencias SQL se podrá consultar la información contenida en la base de datos.

- Debido a que el sistema posee interfaz gráfica es necesario utilizarlo en su mayor parte mediante dispositivos de entrada como son el ratón y el teclado.
- Si algún usuario intenta entrar en algún área no permitida, el sistema le anunciará el acceso denegado.
- Debe el sistema soportar y preveer medios de envío de registros y archivos(fotos).Respaldo de datos.

### **Requisitos de la empresa**

Para el presente trabajo el requisito de empresa es el siguiente:

- Para aminorar costos el sistema tendrá que ser desarrollado con herramientas de distribución gratuita.

La determinación de requisitos tiene 2 propósitos:

- El acuerdo entre clientes y desarrolladores sobre qué debe ser el sistema.
- Proporcionar a los diseñadores pautas para el desarrollo

## **3.4 INTERACTIVIDAD Y CAPACIDAD DE RESPUESTA EN APLICACIONES WEB**

El desarrollo Web basado en una programación múltiple han demostrado obtener mejores rendimientos contra las aplicaciones Web de programación tradicional. Se han hecho algunas pruebas para medir la eficiencia de dicha programación, tomando en cuenta el número total de Bytes transferidos y el tiempo de espera. Haciendo consultas a una base de datos considerando ambas arquitecturas. Demostrando así que es superior en mucho, implicando una mayor posibilidad de brindar una mejor experiencia al usuario. Ya que reduce tiempos de espera llevando al usuario a una mayor interactividad. CSS se utiliza principalmente para estilizar las páginas Web y hacerlas más fácilmente comprensibles para los usuarios. Javascript permite la accesibilidad al DOM (Document Object Model) para permitir una presentación de información dinámica y una mayor interactividad. El DOM es un modelo de objetos estándar que hace HTML, XML y otros formatos relacionados. XMLHttpRequest es importante

en las técnicas AJAX de desarrollo Web para aumentar la capacidad de respuesta de páginas Web y la interactividad. Por ejemplo, se utiliza actualmente en los servicios de Gmail de Google y MapQuest entre otros sitios altamente interactivos.

Los principales beneficios de las aplicaciones multiples permiten a los usuarios acceder y crear aplicaciones Web más rápido con un mejor nivel de respuesta de lo que se suele atribuir a las aplicaciones de escritorio. Esta mayor capacidad de respuesta va acompañada de una mayor interactividad que hacen innecesario que la página Web sea vuelta a cargar cada vez que un usuario realiza una solicitud nueva de la aplicación Web. La información aquí presentada será reafirmada en el momento de las pruebas e integración del presente trabajo.

### **3.5 INCREMENTO DE VENTAS**

El E-commerce "hacer negocios electrónicamente", permite a la empresa de panadería “Carmencita” adquirir una fuerte posición estratégica dentro del mercado, una manera de incrementar sus ventas, alcanzar nuevos mercados y servir mejor a sus clientes. Esta estrategia de incremento de ventas ya empieza a palpase en otras industrias de muchos rubros de productos y servicios, donde las operaciones y producción, basándose en las diferencias reales manifestadas directamente en las compras de los consumidores obtenidos en tiempo real vía Internet, demuestran que estos negocios tienden a aumentar sus ventas prácticamente sin inventarios y los intermediarios cada vez son menos requeridos, lo que significa una disminución de los costos y generar formas novedosas de añadir valor a los servicios ofrecidos.

En los últimos años, la economía mundial se ha desarrollado más allá de las expectativas, y la dirección del sector privado de forma eficaz jugando un papel importante en dicho proceso económico.

De acuerdo al volumen de negocios que efectúan las empresas, la perspectiva cambia: es posible pensar cómo una oportunidad de negocio puede llegar a representar mayores beneficios económicos a todos los sectores.

En nuestro país, el número de operaciones comerciales o financieras que pueden realizarse va aumentando progresivamente. Es alentador observar como las grandes tiendas y algunas entidades financieras ofrecen la posibilidad de comprar o realizar transacciones económicas.

El Comercio Electrónico mejora visiblemente las ventas de los productos servicios que ofrecen las empresas y abre el camino para entablar relaciones comerciales más provechosas y eficientes.

Es posible realizar compras en línea configurando una extranet o red externa de computadoras que este directamente conectada a los proveedores, disminuyendo los costos de inventario y aumentando la disponibilidad para sus clientes agilizando las relaciones financieras con clientes y proveedores habilitando sistemas de facturación y pago a través de la Web; logrando que la empresa reduzca el costo de procesamiento de los pedidos en forma electrónica, recortar gastos derivados del inventario y minimizar los costos de la venta, que se conservando la lealtad de los clientes.

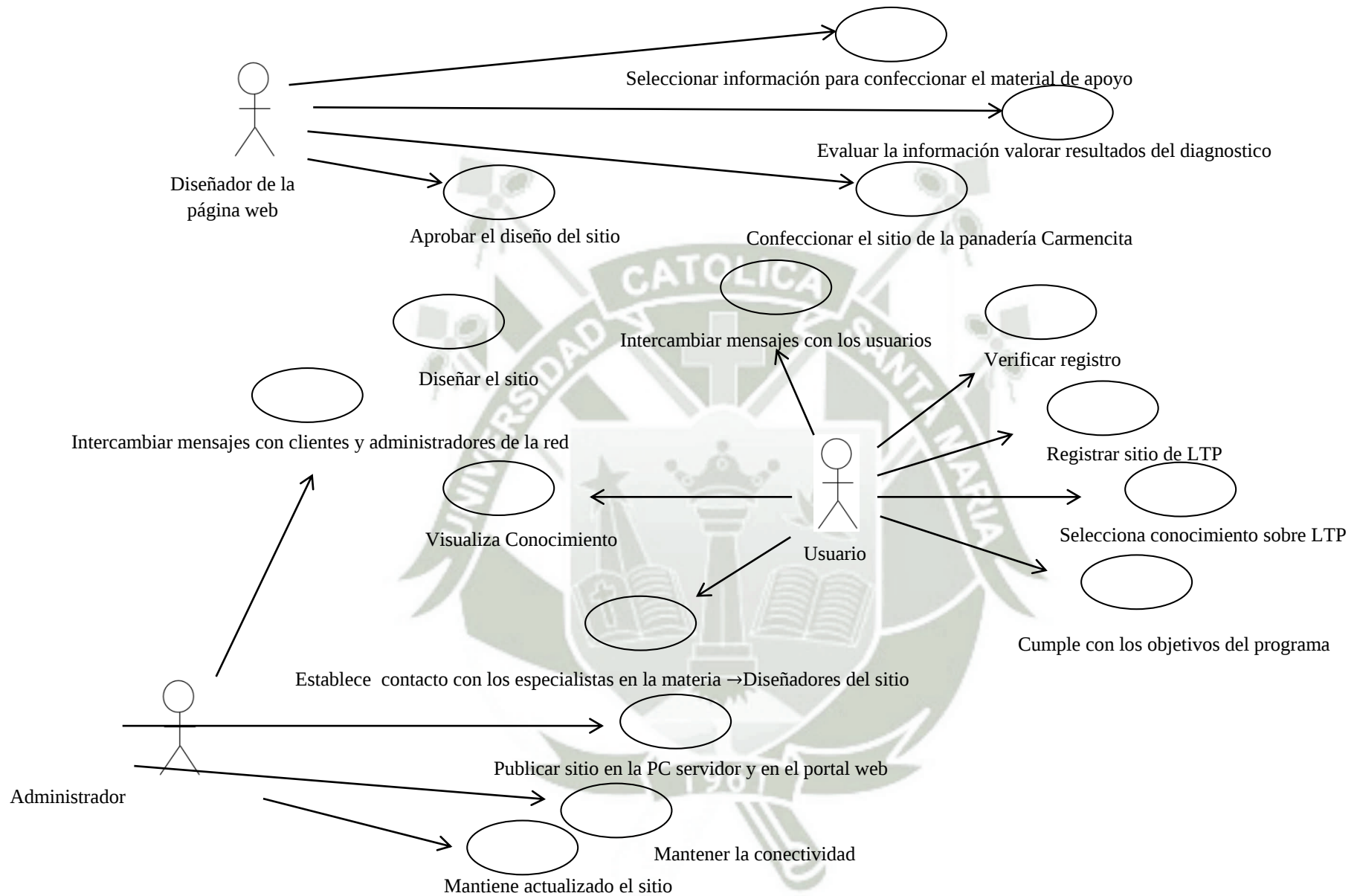
En la ciudad de Arequipa un alto porcentaje de su población activa se ve limitado a realizar compras tradicionales para satisfacer demandas en sus hogares ya sea por falta de tiempo, trabajo, estrés, u otra situación interna o externa a su dinámica habitual.

La idea de ofrecer un Portal Web para incrementar las ventas de Productos de panificación para la empresa de panadería “Carmencita”, es abordar las debilidades e incrementar su plataforma comercial y tecnológica, ofreciendo un modelo de comercialización de productos de panificación que permita a sus clientes realizar compras y ventas permanentes desde cualquier sitio donde se tenga acceso a Internet y entre otras empresas en las cuales ya se haya implementado este modo de hacer negocios. Dicha plataforma permite fortalecer la producción y difusión lo que impactará en la obtención de beneficios económicos, competitividad empresarial, y financiamiento.

La empresa de panadería “Carmencita” a través de esta estrategia modelo incremento de ventas y comercio electrónico propuesto, garantiza la reducción de costos de producción así como los de inventarios, además de mantener un servicio constante y directo las 24 horas del día con el cliente.

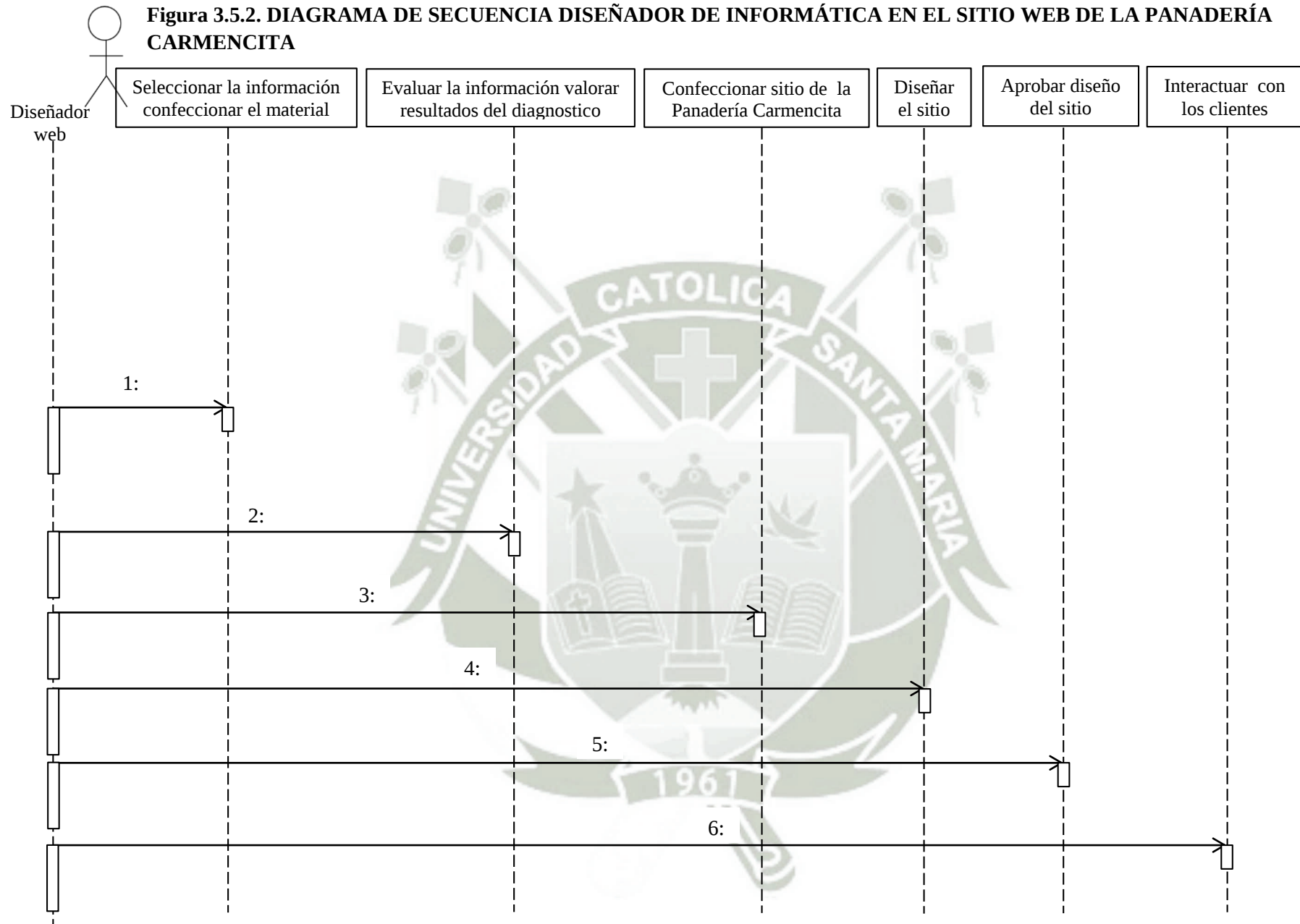


**Figura 3.5.1. CASOS DE USO DEL NEGOCIO DE SITIO WEB PARA LA PANADERIA**



**Fuente: Elaboración propia**

**Figura 3.5.2. DIAGRAMA DE SECUENCIA DISEÑADOR DE INFORMÁTICA EN EL SITIO WEB DE LA PANADERÍA CARMENCITA**



## CAPITULO IV

### DISEÑO Y CONSTRUCCIÓN DE LA APLICACIÓN

#### 4.1. OPERACIÓN DEL NEGOCIO DE PANIFICACIÓN

De acuerdo al título y los objetivos planteados el presente trabajo pretende desarrollar un sistema e-commerce (comercio electrónico) donde se utilicen herramientas de código abierto así ofrecer una plataforma de información en línea para la empresa Panadería “Carmencita” dedicada a la elaboración de productos de panadería, los productos que ofrece esta pequeña empresa son el resultado de un proceso tecnológico con equipamiento e infraestructura optimizada industrialmente puesto que la elaboración del producto final se realiza empleando técnicas tanto tradicionales como emergentes, posee una administración en su mayor parte empírica. Su posición esta en continuo crecimiento.

De manera que el negocio sobre el cual se realiza esta tesis está clasificado como una Pequeña y Mediana Empresa (PYME), este tipo de clasificación está determinada por la Ley N° 28015. Ley de promoción y formalización de la micro y pequeña empresa en nuestro país, de acuerdo a la legislación peruana que categoriza básicamente como una empresa que reúne todos los aspectos de una empresa tradicional pero que naturalmente cuenta con características distintivas como son el número de empleados, que para nuestro país corresponde de 1 a 15 trabajadores para microempresa y para pequeña empresa entre 50 y 100 en el sector comercio.

Para mostrar un panorama global de la manera como está actualmente operando el negocio, se expone el funcionamiento de cada área y lineamiento que están actualmente involucrados en el proceso de producción de la pequeña empresa. Se cuenta con los siguientes mecanismos para la producción y movimiento del negocio, los cuales clasificamos en cuatro principales áreas:

**Administración y operaciones.** Esta área se ocupa de la transacción operativa relacionada con la empresa, Desde la contratación del personal hasta la compra de la materia prima, el pago del personal, firma de los cheques, horarios del personal, pago de proveedores, realizar gestiones empresariales como facturas y cuentas simples, control de inventarios tanto de los materiales como de los productos finales, limpieza del local.

### **Producción.**

*Producción y distribución.* Área donde se encargan de mezclar los ingredientes según el requerimiento, después de este proceso se procede a pasar esta mezcla a la batidora donde es amasado, de ahí a la mesa donde se da forma al producto, es decir, se hacen productos elaborados de diferentes tamaños. Después de este proceso pasan los productos a la fase de crecimiento, luego al horno y por último se empacan. En el área administrativa se establecen las formas de pago considerando los costos de envío e instalación y finalmente el tiempo de entrega del producto terminado. El producto se realiza en serie ya que se produce aproximadamente el mismo volumen y cantidad todos los días.

En el proceso de distribución el personal encargado se toma la tarea de almacenar el producto en recipientes adecuados los cuales se transfieren a mostradores para expenderlos luego al público, o pueden ser llevados en una movilidad que sirve para llegar a los diferentes destinos (tiendas) si en el itinerario del empresario lo establece.

*Condiciones ambientales de la planta.* La planta está ubicada en el primer piso. El área de producción es de 6m con 30 cm x 3m con 25cm. Contiguos al área de producción están el almacén y el depósito. Los dos ambientes poseen compartimentos donde actualmente están una cámara de fermentación y una mesa de trabajo. Al lado del área de producción están también la tienda y la trastienda, todos estos ambientes están adecuadamente acondicionados.

*Ventilación.* La ventilación de la planta de producción es óptima pues las máquinas están situadas a distancia de las paredes y la ventilación es natural que entra a los ambientes de producción por puertas, ventanas y espacios acondicionados en el techo además cuenta con ventiladores.

*Iluminación.* La iluminación es buena los ambientes son amplios. Esto se ve reflejado en el área de producción que consta de buena iluminación natural y artificial.

*Limpieza.* El área de trabajo permanece limpia y los insumos están apilados en zona adecuada a distancia del piso cerca del área de producción, y se encuentran almacenados de manera organizada de manera que no pueda afectar el ritmo producción, tráfico de empleados e higiene.

*Ruido.* El ruido de las maquinas no es alto gracias y se genera escaso ruido.

*Accidentes laborales:* no se ha tenidos accidentes en el área de producción, ni distribuyendo el producto a los diferentes clientes.

*Políticas de compra de materias primas:* Es política de la empresa comprar al contado y a crédito a su único proveedor, establece plazos máximos para pagar la deuda que haya adquirido el empresario. Algunas materia primas perecibles son las únicas por la cuales se debe pagar al contado.

**Contabilidad y finanzas.** Es el área encargada del registro de todos los movimientos de ingresos monetarios a través del tiempo, en esta área se llevan a cabo las declaraciones e impuestos de acuerdo a los resultados de los libros de contabilidad que lleva la empresa como emisión de facturas, proyección del negocio por ventas y costos asociados al desarrollo de todo el negocio.

**Mercadeo y ventas.** En esta área se hace el primer contacto con el cliente, estableciendo su contacto con el personal de la empresa del área administrativa, solicitando una cotización de algún producto de su necesidad, si el producto solicitado se encuentra en posibilidad de elaboración y satisface las necesidades del cliente se procede a presentar una cotización del producto indicándole los beneficios y características.

En esta sección también se incluye el personal responsable de las estrategias de mercado del negocio, tales como la publicidad en forma impresa pero no en medios electrónicos ni Internet. Está también el personal que atiende en el mostrador a las personas que acuden al local para solicitar una compra de algún producto ofreciéndoles los diferentes productos y sus cotizaciones ya elaborados o los que puedan ser elaborados de acuerdo a las necesidades particulares del cliente y posteriormente haciéndolos llegar al área de producción.

#### 4.2. LOGÍSTICA DE LA EMPRESA

**Razón social** : Panadería Carmencita  
**Actividad económica** : Panadería  
**Dirección comercial** : Esquina Calle Piura –Mariscal catilla Cuadra 10 distrito de Miraflores  
**Representante legal** : Fredy Arevalo  
**Antigüedad de la empresa** : 5 años.

#### Gastos Elaboración de pan.

| Harina      | Mejorador | Sal       | Azúcar    | Leche en polvo | Total      |
|-------------|-----------|-----------|-----------|----------------|------------|
| 1,924 Kg    | 5,426gr   | 32,110 gr | 86,060gr  | 5,200gr        |            |
| S/ 3 386,24 | S/ 35,26  | S/ 38,53  | S/ 172,12 | S/ 62,40       | S/ 3694,31 |

**Fuente: Elaboración Propia**

#### Gastos elaboración de Pasteles

| Mantequilla | Harina   | Crema<br>pastelera | Huevos   | Chantilly | Manjar      | pasas      | Azúcar  | Total        |
|-------------|----------|--------------------|----------|-----------|-------------|------------|---------|--------------|
| 13,400 gr   | 38,400gr | 4,800 gr           | 200 unid | 3,000gr   | 6,000gr     | 800gr      | 3,500gr |              |
| S/ 172,57   | S/67,58  | S/ 36,00           | S/ 57,00 | S/ 33,00  | S/<br>36,00 | S/<br>9,50 | S/ 6,86 | S/<br>418,51 |

Fuente: Elaboración Propia

### Gastos Servicios

| Agua     | Luz       | Teléfono | Gas        | Empleados  | Total      |
|----------|-----------|----------|------------|------------|------------|
| S/ 45,50 | S/ 387,50 | S/ 33,00 | S/ 1200,00 | S/ 2700,40 | S/ 4366,00 |

Fuente: Elaboración Propia

### Gastos totales

| Panadería   | Pastelería | Servicios  | total       |
|-------------|------------|------------|-------------|
| S/ 3 694,31 | S/ 418,51  | S/4,366,00 | S/ 8,475,00 |

Fuente: Elaboración Propia

## 4.3. SOFTWARE Y HARDWARE PARA SOPORTAR LA APLICACIÓN

Luego de analizar las áreas involucradas en los procesos de venta y producción se comenzará por revisar los requisitos técnicos tanto de software como hardware para que la aplicación tenga el desempeño deseado.

Para los efectos de implementación del sitio Web a construir, vamos a tomar como referencia técnicas de programación Web que lideran la Web 2.0 como Asynchronous Javascript And XML, JavaScript asíncrono y XML, que requieren de ciertas capacidades tanto de software y de hardware.

### Herramientas para el Desarrollo de la aplicación:

**Adobe Dreamweaver CS5.** El cual nos permite diseñar, desarrollar, implementar y mantener de forma eficiente el sitio y las aplicaciones Web, ya que el propio programa va guiando al usuario en el manejo y utilización.

#### **Características:**

- Entorno de desarrollo compatible con PHP, J2EE y Microsoft.NET, Windows xp, Windows 7.
- Diseño de páginas Web con interfaz gráfica.
- Barra de herramientas de programación, que facilita las operaciones de programación más usadas.
- Contracción del código, para centrarse sólo en el que se está utilizando.
- Barra de herramientas de reproducción de estilos, con la que se puede visualizar el resultado final.

Tomando en cuenta las especificaciones y estándares de desarrollo es indispensable contar con los siguientes requisitos del sistema:

#### **Para Windows**

- Procesador Intel® Pentium® 4, Intel Xeon® o Intel Core™ Duo (o compatible), core 3, core 7.
- Microsoft® Windows® XP con Service Pack 2 ó Windows Vista™ Home Premium, Business, Ultimate o Enterprise (certificada para ediciones de 32 bits)
- 512 MB de RAM como mínimo,
- 40 MB de espacio disponible en el disco duro (se necesita espacio libre adicional durante la instalación)
- Resolución de pantalla de 1.024 x 768
- Unidad de DVD-ROM
- Software necesario para funciones multimedia.

- Conexión a Internet alámbrica o inalámbrica para activar el producto

#### **Para Macintosh**

- Procesador PowerPC® G4, G5 ó Intel® multinúcleo
- Mac OS X v10.4.8-10.5 (Leopard)
- 512 MB de RAM
- 40 MB de espacio disponible en el disco duro (se necesita espacio libre adicional durante la instalación)
- Resolución de pantalla de 1.024 x 768
- Unidad de DVD-ROM
- Software necesario para funciones multimedia
- Es necesaria una conexión a Internet o telefónica para activar el producto.
- Algunas extensiones *Toolbox (Caja de herramientas de desarrollo)* que permitirá desarrollar la aplicación en menor tiempo.

De igual manera es necesario también contar con un mínimo de requisitos para el uso de la extensión, estos también son señalados por el fabricante:

- Adobe Dreamweaver CS5.
- Adobe Extension Manager CS5

#### **Servidores compatibles.**

- ASP - Microsoft IIS 5.0 ó superior.
- PHP - Apache 1.3.x or 2.x / Appserv 2.5.10.
- CF - Built in ColdFusion 6 ó 7.

#### **Base de datos compatibles**

- MySQL 4.0 ó superior
- Microsoft MSSQL Server 2008
- Microsoft Access 2007 y 2008

### **Navegadores compatibles**

- Microsoft Internet Explorer 6.0 ó superior
- Mozilla Firefox 1.5 ó superior
- Safari 2.0 ó superior.
- Google Chrome
- Opera

También es necesario contar con una herramienta que permita la facilidad de uso del servidor y los lenguajes involucrados, llamada APPSERV; que es de uso gratuito. y funciona con los sistemas operativos:

- Linux
- Windows
- Mac OS X
- Solaris

Además de proporcionar Apache, PHP y MySql, ofrece otras utilidades de interés, tales como Webalizer, Filezilla FTP Server, Mercury Mail Transport o Zend Optimizer, siendo éste el motivo por el que se ha decantado por su uso frente al paquete Wampp.

Se instaló la versión APPSERV 2.5.10 para Windows que contiene las siguientes herramientas:

Apache 2.2.14 (IPV6 enabled), MySQL 5.1.41 (Community Server) con PBXT engine 1.0.09-rc, PHP 5.3.1 (PEAR), Miniperl 5.10.1, ApacheFriends Edition, SQLite 3.6.20, OpenSSL 0.9.8l, phpMyAdmin 3.2.4, msmtmp 1.4.19 y Webalizer 2.21-02.

### Requisitos de sistema para la operación de APPSERV:

- 128 MB RAM
- 192 MB libres de espacio en Disco duro.
- Windows 2000, XP (Server 2003), Vista (Server 2008), 7
- Todos los sistemas de 32 bit (64 también debería trabajar)

Finalmente para que la aplicación funcione adecuadamente al momento de su implementación, es necesario contar con un mínimo de requerimientos basados en los datos del fabricante de las herramientas elegidas, en los siguientes aspectos:

### Equipo de cómputo

#### ❖ En el cliente

##### • Hardware:

- Memoria RAM: 128 MB RAM (Recomendado: 256 MB RAM o más).
- Disco Duro (HD): 100 MB de espacio en disco.
- Procesador: Pentium 233 MHz (Recomendado: Pentium 500MHz o superior).

##### • Software:

- Sistema Operativo: Windows, Linux, Mac OS
- Navegadores: Que permitirán usar el servicio HTTP (protocolo de transferencia de hipertextos), es decir, poder visualizar las páginas Web.

**Navegadores Web:** Navegadores basados en Gecko como Mozilla, Mozilla Firefox, SeaMonkey, y Netscape versión 7.1 y superiores, Google Chrome, Microsoft Internet Explorer para Windows versión 5.0 y superiores, y los navegadores basados en él. Apple Safari

versión 1.2 y superiores, y el Web. Opera versión 8.0 y superiores, incluyendo Opera Mobile Browser versión 8.0 y superiores.

Navegadores que no permiten desarrollo optimo: Opera 7 y anteriores, Microsoft Internet Explorer para Windows versión 4.0 y anteriores, Safari 1.2 y anteriores, Dillo, Navegadores basados en texto como Lynx y Links, Navegadores para personas con capacidades especiales visuales (Braille), Algunos navegadores de teléfonos móviles, Navegador de la PSP.

- Lenguajes de Cliente: JavaScript.

❖ En el Servidor:

- Hardware:
  - Memoria RAM: 128 MB RAM (Recomendado: 256 MB RAM o más).
  - Disco Duro (HD): 100 MB de espacio en disco.
  - Procesador: Pentium 233 MHz (Recomendado: Pentium 500MHz o superior).
- Software:
  - Sistema Operativo: Windows, Linux, Mac OS.
  - Navegadores Web. (Las mismas consideraciones que para el cliente)
  - Lenguajes de Servidor como son PHP, ASP, JSP, etc.
  - Servidor Web: Apache o en la versión comercial de Windows llamada IIS.

### **Conexión a Internet**

Las pruebas se hacen en un servidor local en el que no es imprescindible una conexión a Internet, cuando el proyecto esté funcionando en línea es necesario contar

obviamente con una conexión a Internet a través de un proveedor del servicio. Es necesario que el cliente Web esté conectado a un módem con suficiente ancho de banda, mínimo 56kb.

#### **Unidades**

Unidad de CD-ROM (si la instalación del software se hace desde un CD-ROM).

#### **Pantalla**

Monitor LG StudioWorks 700E (800 x 600) o de mayor resolución con 256 colores.

#### **Periféricos**

Mouse, teclado.

### **4.4. METODOLOGÍA DE IMPLEMENTACIÓN.**

El presente trabajo de desarrollo de un modelo e-commerce para una microempresa panadera emplea el modelo Top-down, el cual establece un resumen del sistema, tomando en cuenta todos los detalles. En cada parte del sistema se hace el rediseño con la mayor aproximación hasta que la especificación completa sea lo suficientemente detallada para validar el modelo. El modelo se diseñó con frecuencia haciendo uso de la ayuda de "cajas negras" que hicieron más sencillo cumplir con los requerimientos. La utilización de la técnica de diseño empleada tiene los siguientes objetivos básicos:

- Simplificación del problema y de los subprogramas de cada descomposición.
- Las diferentes partes del problema pueden ser programadas de modo independiente e incluso por diferentes personas.
- El programa final queda estructurado en forma de bloque o módulos lo que hace más sencilla su lectura implementación y mantenimiento.

### **Planificación de sitio Web.**

Considerando algunos aspectos útiles para la creación del sitio Web o página Web. Primeramente se tomó en cuenta la elección del tipo de Web a crear de acuerdo a las necesidades ya mencionadas. Por las características de nuestra Web el sitio será comercial, atractivo y dinámico ya que ofrecemos al usuario la posibilidad de realizar pedidos y compras en línea de algún producto en existencia.

Posteriormente se definió la temática del sitio. Se definieron los temas materia de contenido del sitio Web, permitiendo seleccionar términos claves de búsquedas para posteriormente realizar una metodología de posicionamiento en los buscadores más utilizados, la temática del sitio Web a desarrollar es la promoción, venta y distribución de productos y servicios de panificación en general.

Teniendo en cuenta los objetivos que se pretenden alcanzar con la creación del sitio Web, los objetivos son dar a conocer la empresa, el negocio, los servicios y productos que ofrece, captar clientes, generar nuevos pedidos de una forma más eficiente y amplia.

La forma y el diseño de la construcción ayuda a poder ampliar el sitio Web con futuros cambios para su mejora agregando nuevas aplicaciones, secciones, páginas, actualizaciones constantes, etc.

Después de plantear los objetivos y la escalabilidad de la Web tanto a corto como a largo plazo, se procedió a definir sobre papel el diseño de la Web, incluyendo fondos, tipos de letras, botones, formularios, enlaces, plantillas aplicaciones, de tal manera que se obtengan de manera global todos los elementos y diseños que se desean implementar para generar posicionamiento efectivo en la Web.

### **Interfaz visual y elaboración de la información a implementar**

Aquí se ha definido el sitio con un máximo nivel de detalle, se establecieron modelos de aspecto gráfico, del contenido y del funcionamiento. Estos modelos son prototipos del sitio y lo reflejan con exactitud, se trabajaron con el cliente evolucionándolos hasta

quedar satisfecho. Fue fundamental la aprobación del cliente, pues los cambios en etapas posteriores son más costosos.

Luego conocer al detalle el sitio se corrigen los plazos y costos de las siguientes etapas. Creación del esqueleto de la Web, tablas, encabezados, espacio para imágenes, texto, botones, etc. Elección de las imágenes que acompañara a nuestro sitio Web, logos, cabeceras, fotografías, además del proceso concepción y materialización de la información que se ofrece.

### **Aplicaciones Web**

Incluyó la creación de las aplicaciones elegidas para nuestro sitio Web, como el mecanismo de venta en línea de los productos, la personalización de productos, etc., mediante programación específica y creación de bases de datos.

Se hizo realidad el modelo de diseño, es decir que se elaboraron y adaptaron los elementos gráficos y multimediales, se codificaron las páginas, los programas y scripts, se definieron y prepararon las bases de datos para que el sitio quede en funcionamiento.

En esta etapa se realizaron pruebas exhaustivas para asegurar el perfecto funcionamiento del mismo, se probaron la integración con los sistemas internos del cliente, todo esto primero en el ambiente de desarrollo y luego en el servidor de Internet en el que realmente funciona. Se separó lógica y contenido para que pueda resultar más sencillo de utilizar y de mantener a largo plazo. Esta separación de lógica y contenido se llevó a cabo mediante:

- Archivos y almacenamiento de distintas partes del contenido.
- Utilización de un API de funciones o de clases con un conjunto de funciones miembro para conectar contenido dinámico o plantillas de páginas estáticas.
- Utilización de un sistema de plantillas, lo cual permite a otra persona diseñar sobre nuestras plantillas.

- Posicionamiento.

Una vez terminado el sitio Web se procedió a que sea localizable en cualquier buscador, especialmente Google, y pueda aparecer en las primeras posiciones para recibir mas visitas en la red.

En este punto es donde se ha de implementado una metodología específica para alcanzar un posicionamiento optimo, definiendo las palabras claves de búsqueda.

### **Pruebas**

Se realizaron pruebas para comprobar la utilización y correcto funcionamiento del Sitio y para evitar mostrar páginas con un formato diferente que pueden aportar una imagen negativa.

Puesta en Funcionamiento: Habilitación del sitio, en Internet, para que los verdaderos usuarios comiencen a servirse del mismo.

Fue necesario difundirlo, mediante publicidad, volantes organización, por banners en sitios específicos, inclusión en buscadores, redes sociales, etc. Esto se hizo en función de la estrategia y de los objetivos.

El sitio se puso en pleno funcionamiento con usuarios verdaderos accediendo y con los administradores realizando las tareas periódicas de mantenimiento como son: atención al cliente, actualización de contenidos y administración del servidor.

### **Implementación de sistemas de pago**

Se sumó el pedido del usuario y se registraron los detalles de entrega. No se procesaron los pagos. Hay una gran variedad de sistemas de pago disponibles, cada uno con una implementación diferente. Como mínimo, el sistema de pagos selecciona información del usuario (por ejemplo, un número de tarjeta de crédito), información que se identifica (para especificar en qué cuenta se va a realizar el cargo) y la suma total de la transacción.

## **4.5. MODELO DE E- MARKETING ELECTRONICO**

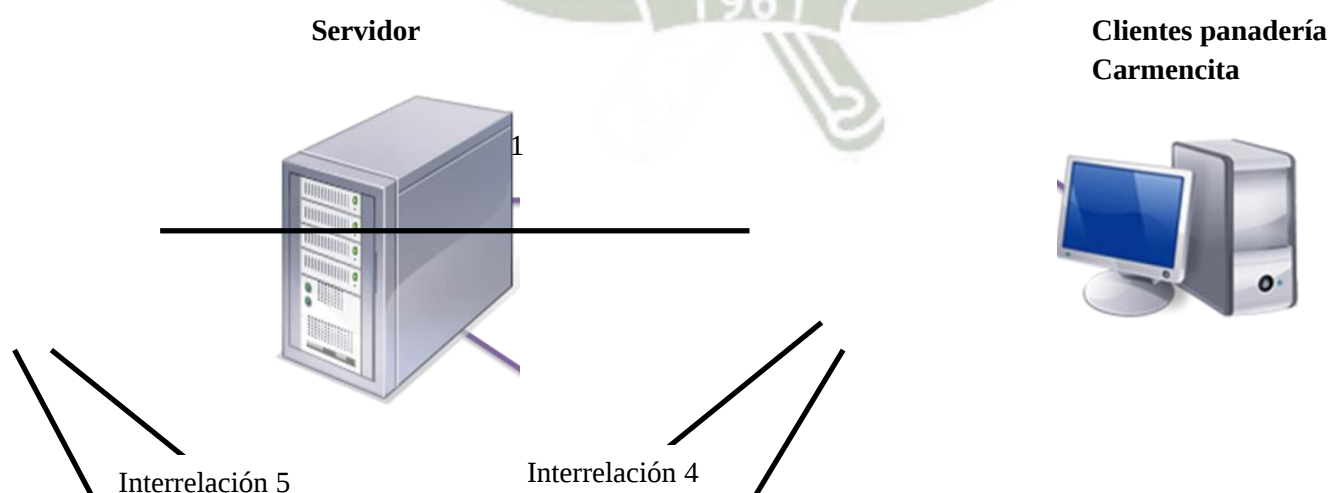
Para el desarrollo de nuestra página web hemos utilizado la estructura de comunicación existente en Internet, que permite el acceso universal a los servicios de información presentes en la Web mediante la conexión remota entre una red de ordenadores o máquinas llamadas servidores, y los ordenadores clientes. Así pues, en nuestro caso el modelo que nos ha permitido establecer relaciones comerciales y marketing electrónico vía web ha sido el modelo cliente-servidor.

### **Modelo de marketing página web “Panadería Carmencita”: cliente –servidor**

El modelo de la página web panadería Carmencita es un modelo que está basado en dos parte conocidas como Cliente o Front-End y un servidor o Back- End, el Fron -end se encuentra del lado del usuario que es el que consume el producto o servicio y el Back-end provee los productos que requieren los clientes de la panadería Carmencita.

El modelo de E-Marketing de la panadería Carmencita está basado en los siguientes elementos:

#### **ESQUEMA GENERAL DE INTERRELACIÓN ENTRE LOS ELEMENTOS DEL MODELO DE E-MARKETING DE LA PANADERÍA CARMENCITA**

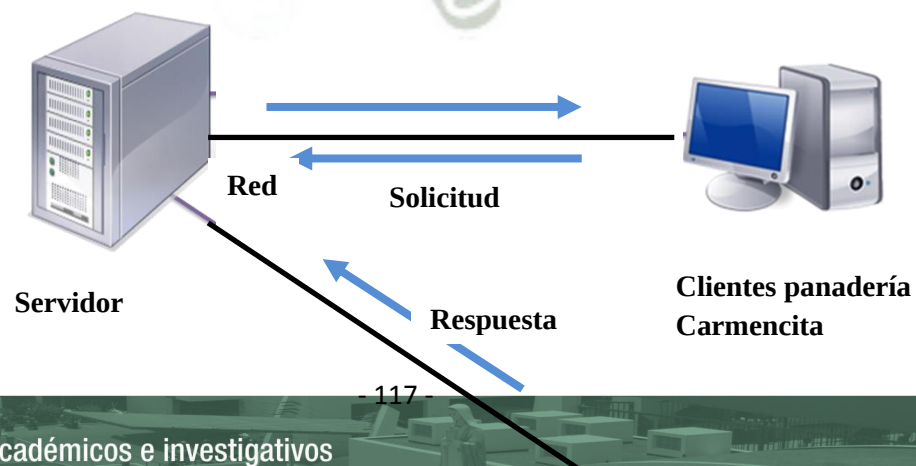




Paquetes de datos de la página web

- **Cliente o Front end:** Dentro del modelo cliente/servidor de la página web panadería Carmencita, esta es una persona, usuario o público en general que emplea los recursos que proporciona la interfaz de un servidor que le brinda los servicios que está buscando.

**Objetivo:** Iniciar algún requerimiento de servicio o producto de la panadería Carmencita.



Solicitud



- **Servidor o Back end:** Este es un nodo de red que proporciona el servicio a las PCs clientes de la panadería Carmencita; por ejemplo, acceso a la página web panadería Carmencita, formación de trabajos de cotización o realización de pedidos.

**Objetivo:** El objetivo del servidor donde se aloja la página web de la panadería Carmencita es almacenarla como un dato que luego podrá ser compartido desde ahí por los clientes de la panadería.

Cliente panadería  
Carmencita



Servidor



Cliente panadería  
Carmencita



- **Nodos:** Están dados por cada uno de los ordenadores individuales a través de las cuales se puede tener acceso a la página web de la panadería Carmencita, no solamente podrán ser computadoras sino iPhones y smartphones u otros dispositivos conectados a la red.

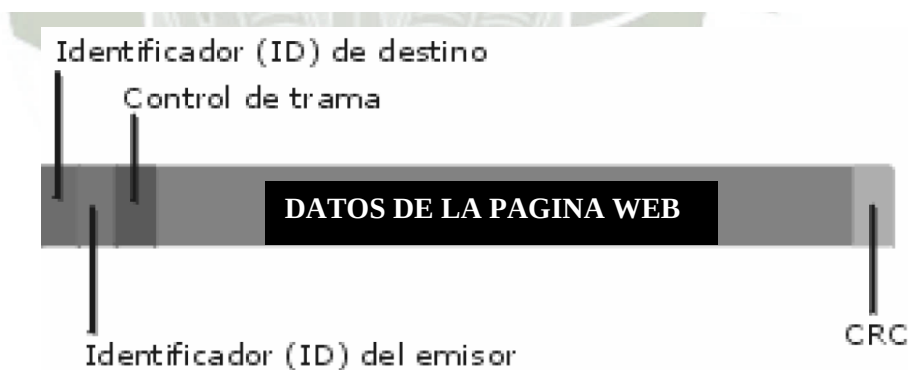
**Objetivo:** organizar y presentar los datos de la forma adecuada, ser unidad de información básica en los que la página web de la panadería Carmencita guarda sus contenidos. Así el contenido del sitio se guarda en un “depósito de nodos” que va creciendo conforme el administrador, si así lo desea vaya generando nuevos contenidos.

INTERNET



- **Paquetes de datos:** Es cada uno de los bloques en que se divide, en el nivel de Red, la información de la página web de la panadería a enviar. Por esa razón es que nuestra página requirió dividir la información para enviar en bloques de un tamaño máximo conocido, lo cual simplifica el control de la comunicación, las comprobaciones de errores, la gestión de los equipos de encaminamiento (*routers*), etc.

**Objetivo.** Contener la información necesaria para trasladar el los archivo de nuestro sitio web desde el emisor hasta el receptor, el *área de datos* (*payload* en inglés) que contiene los datos que se desean trasladar, y la *cola* (*trailer* en inglés), que comúnmente incluye código de detección.

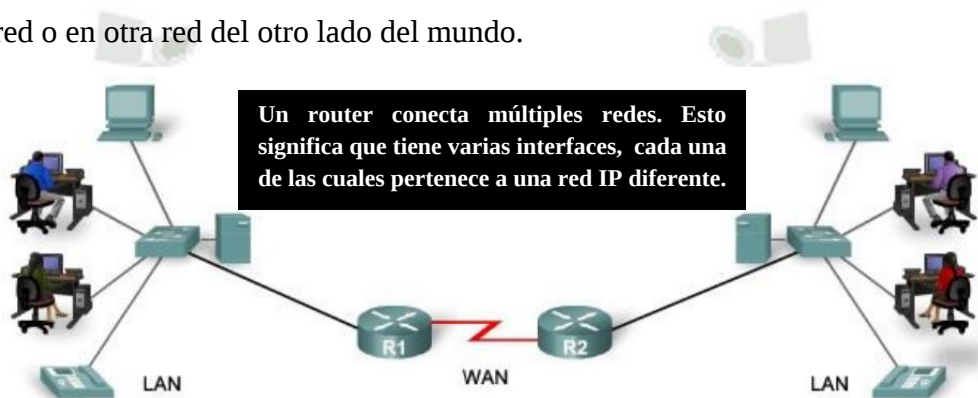


Se puede observar que las partes principales del paquete son el identificador o dirección destino, el identificador o dirección origen, el control de trama el cual “indica si en el paquete enviado o recibido hay datos o solo información para fines de control”

- **Ruteador:** dispositivo que conecta dos redes y que opera como un puente, pero que también puede elegir rutas a través de una red.

**Objetivo:** Los router utilizan la tabla de enrutamiento como un mapa para descubrir el mejor camino para una dirección determinada.

Los usuarios esperan poder acceder a las páginas Web de la panadería, enviar mensajes, o recibirlos ya sea si el servidor al que están accediendo está en su propia red o en otra red del otro lado del mundo.



Cuando un router recibe un paquete IP en un interfaz, determina que interfaz usar para reenviar el paquete hacia su destino.

#### 4.6. DISEÑO Y CONSTRUCCIÓN DEL BACK END

Se diseñó primero la base de datos a utilizar en todo el sitio Web. En la tabla 4.6.1 se resumen todas las tablas propuestas de la Base de Datos que fueron necesarias para la sección de la tienda en línea y la galería de imágenes así como una breve descripción de cada tabla.

| Nombre de la tabla | Descripción                                                               |
|--------------------|---------------------------------------------------------------------------|
| Admin              | Guarda las credenciales del los usuarios con privilegios de administrador |

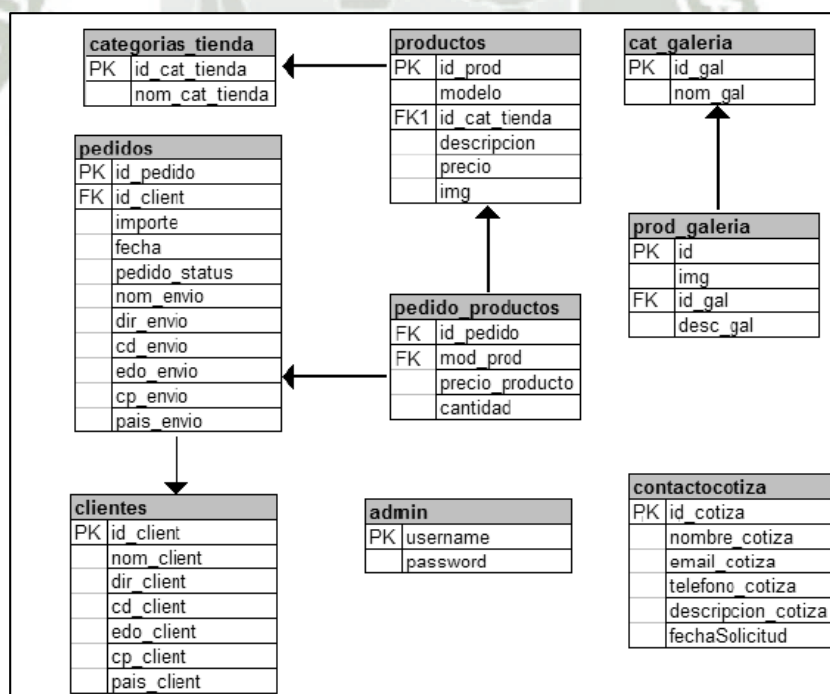
|                   |                                                                       |
|-------------------|-----------------------------------------------------------------------|
| categorias_tienda | Almacena todas las categorías de los productos de la tienda en línea. |
| Contacto_cotiza   | Guarda los datos de los usuarios que solicitan una cotización.        |
| Productos         | Almacena todos los productos que se venderán en línea.                |

Fuente: Elaboración propia

Tabla 4.6.1 Resumen de las tablas propuestas para el sitio.

A continuación se explica mejor cada tabla:

- **categorias\_tienda.** Esta tabla almacena las categorías de los productos. Desde el DER se puede ver que la llave primaria de esta categoría es la llave foránea de la categoría productos, podría decirse que es una tabla hija.
- **cat\_galeria.** De forma análoga a la tabla anterior en esta tabla se guardan los nombres de las categorías de los productos que no se venderán en línea pero que se mostrarán en una galería de imágenes. De igual manera la llave primaria de esta tabla es la llave foránea de la tabla prod\_galeria. Reafirmando la integridad referencial.



Fuente: WELLING THOMSON –LAURA. 2009.

**Figura. 4.6.1.** Diagrama Entidad Relación aplicaciones.

- **productos.** En esta tabla se guardan el modelo de los productos que se venderán en línea, la categoría a la que pertenecen, una descripción del producto, su precio asociado y el nombre del archivo de imagen del producto. El sistema será capaz de generar automáticamente las imágenes en miniatura de cualquier tamaño de imagen que el administrador guarde para ser mostradas en línea, por tanto no será necesario guardar el nombre de un archivo adicional de la imagen (thumbnail) en esta tabla.
- **prod\_galeria.** Aquí es donde se almacenan todos los productos que se mostrarán en la galería de imágenes, de acuerdo a su categoría. Para los fines de la galería solo se requerirá guardar el nombre del archivo de imagen asociado al producto, la categoría a la que pertenece y una breve descripción.
- **admin.** Esta tabla contiene las credenciales de los usuarios con derechos de administrador, como la manera de autenticar a este tipo de usuario será por nombre de usuario y contraseña, solo esos dos parámetros serán guardados.
- **contactocotiza.** Aquí irán los datos de los usuarios finales que deseen obtener una cotización de algún producto por elaborar o solicitar más información de algún producto en existencia.

Las siguientes tablas se plantean para una posterior mejora del proyecto.

- **pedidos.** Tabla que almacena los pedidos que se han hecho así como datos de envío. Esta tabla requerirá del id del cliente que ha hecho el pedido por lo que esta tabla contendrá una llave foránea que será la llave primaria de la tabla clientes. El ingreso de datos a esta tabla tendrá que ser, por norma de la empresa, posterior al proceso de pago.
- **clientes.** Datos de los usuarios que han hecho compras en el sitio.
- **pedido\_productos.** Esta tabla temporal es para eliminar la relación M:M entre la tabla productos y pedidos ya que varios productos pueden estar relacionados con

varios pedidos, con esta tabla se logra que solamente un producto pueda estar asociado a un pedido en particular.

### **Instalación de las herramientas necesarias**

Para el desarrollo de la aplicación se emplea la herramienta AppServ. A continuación se describe el proceso de instalación: Comenzamos la instalación de AppServ respondiendo algunas preguntas del asistente:

- 1.- Welcome to the AppServ Setup Wizard: Clic en -Next-
- 2.- License Agreement: Clic en -I Agree-
- 3.- Destination Folder: Dejar C:AppServ > Clic en -Next-
- 4.- Select Components: Seleccionamos los cuatro componentes: Apache, MySQL, PHP y Phpmyadmin > Clic en -Next-
- 5.-Apache HTTP Server Information: Escribimos el nombre del servidor, ej: miservidor > Escribimos nuestro email > Clic en -Next-
- 6.-MySQL Server Configuration: Elegimos la contraseña del administrador de MySQL > Clic en -Install-
- 7.- Alerta de seguridad de Windows (cortafuegos): Clic en -Desbloquear- para que el cortafuegos permita las comunicaciones del servidor web.
- 8.- Completing the AppServ Setup Wizard: Podemos activar la casilla -Launch WampServer 2 now- para arrancar el servicio > Clic en -Finish-. Aparecerá el AppServer funcionando.

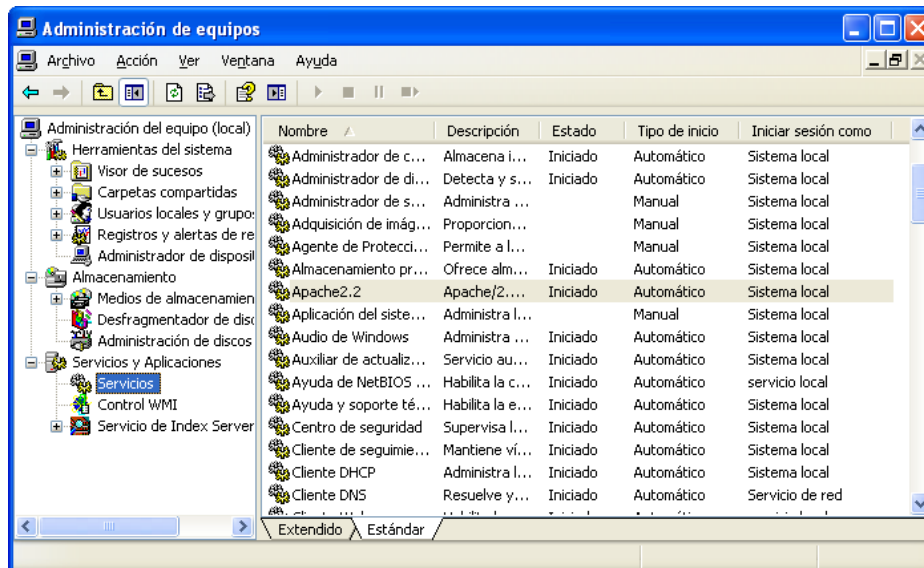
A continuación veremos una captura de la secuencia de instalación de AppServ:

### **Demostración de la instalación de AppServ**

Una vez instalado AppServ se habrá creado la carpeta C:AppServ con un tamaño de 54 MB. El DocumentRoot, es decir, la carpeta en la cual podemos copiar contenidos o aplicaciones para ser accesibles vía web, la carpeta C:AppServ/www.

### **Configuración general de AppServ**

AppServ instala de forma automática y como servicios del sistema, el servidor Apache y el servidor MySQL. No se dispone de ningún asistente ni accesos directos para administrar AppServ, detener o iniciar los servicios, editar las configuraciones, etc. Para iniciar o detener los servicios hacemos clic derecho en Mi PC > Administrar > Servicios y Aplicaciones > Servicios > Clic derecho en Apache2.2 > Iniciar / Detener.



Fuente: Elaboración Propia

Figura 4.6.2. Configuración del Appserv

Desde la ventana de Administración de equipos podemos configurar el inicio automático de los servicios

## Página Web de prueba

Para comprobar el correcto funcionamiento de Apache y de PHP, hemos creado una sencilla página web de prueba dentro del DocumentRoot de Apache y acceder desde el navegador para visualizar el resultado.

```
// Probando Apache y PHP. Crear archivo c:/AppServ/www/prueba.php
```

```
<HTML>
<H1>Probando Apache y PHP</H1>
```

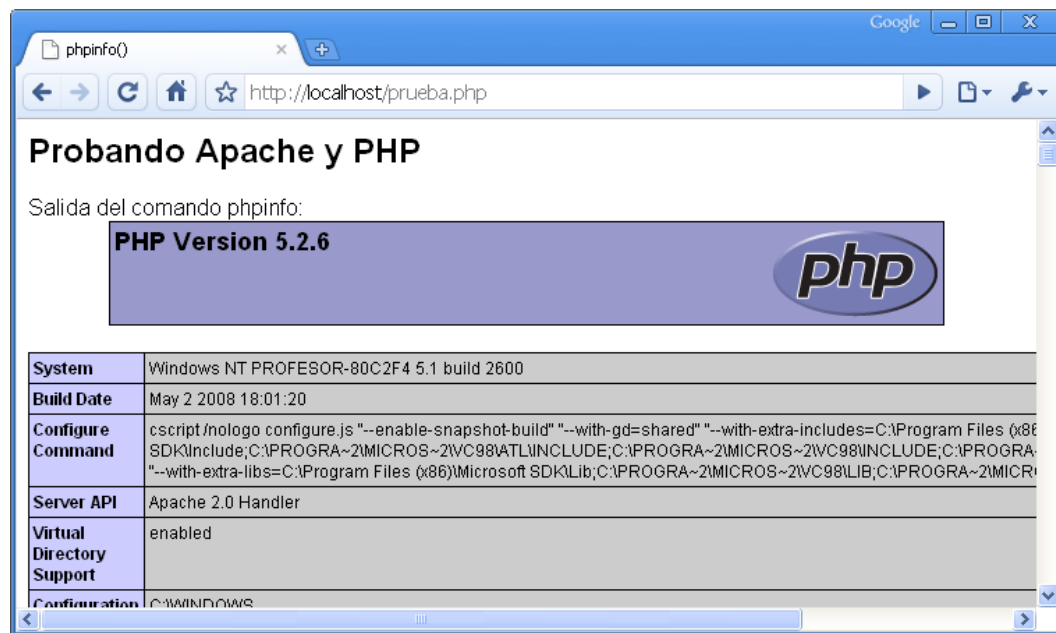
Salida del comando phpinfo:

```
phpinfo();
```

```
?>

</HTML>
```

Después debemos abrir el navegador y acceder a la dirección por ejm:  
http://127.0.0.1/prueba.php para comprobar el resultado:



**Fuente: Elaboración Propia**

### Figura 4.6.3. Probando Apache y PHP

## Configuración de Apache con AppServ

Para realizar cambios en la configuración de Apache con AppServ, fue necesario editar manualmente el archivo `C:\AppServ\Apache2.2\conf\httpd.conf` con el bloc de notas y reiniciar el servicio de Apache desde el apartado –Servicios y Aplicaciones- del sistema. El archivo `httpd.conf` permite configurar cientos de parámetros, pero para un uso normal del servidor, no es necesario modificar el archivo `httpd.conf`, aunque los usuarios avanzados suelen editarlo principalmente para:

- Modificar el raíz de documentos (DocumentRoot)
- Permitir la visualización de carpetas
- Crear alias de carpetas
- Crear espacio web para los usuarios

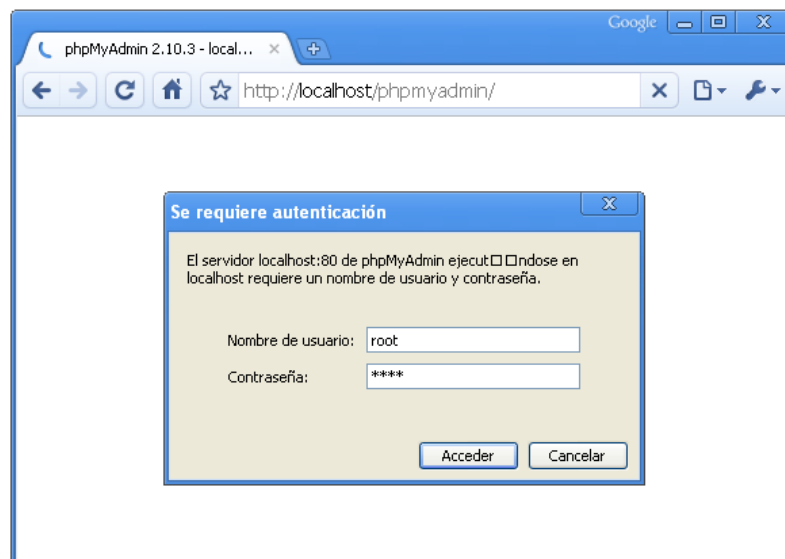
### Configuración de MySQL con AppServ

Para realizar cambios en la configuración de MySQL con AppServ, fue necesario editar manualmente el archivo C:\AppServ\MySQL\my.ini con el bloc de notas y reiniciar el servicio de MySQL desde el apartado –Servicios y Aplicaciones- del sistema. Para un uso normal, no es necesario modificar el archivo my.ini.

### Administración de MySQL con phpmyadmin

La administración de MySQL comprende las tareas de creación y mantenimiento de bases de datos y la gestión de usuarios y permisos. Con nuestros conocimientos de SQL, es posible administrar la base de datos con el cliente de mysql.

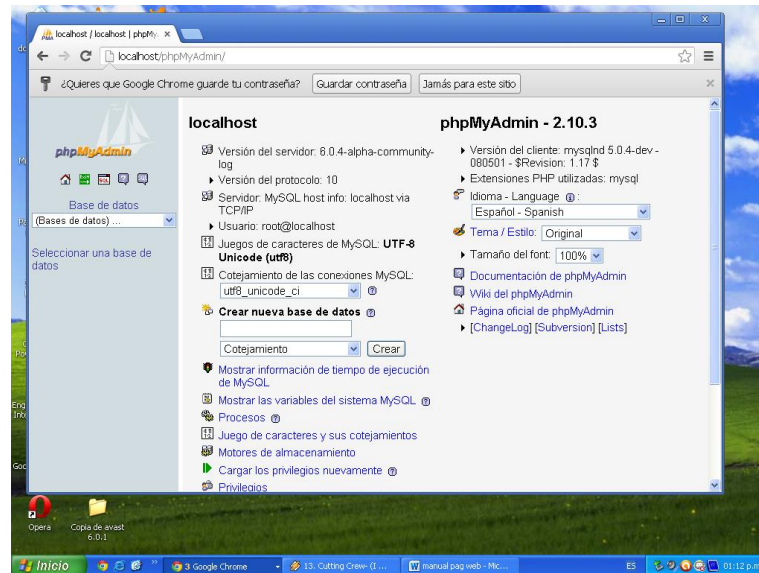
Para acceder a phpmyadmin, tan solo tenemos que abrir el navegador en el servidor e ir a la siguiente URL: <http://localhost/phpmyadmin/>. Nos solicitará un nombre de usuario y una contraseña. En el nombre de usuario debemos poner -root- y en la contraseña debemos poner la contraseña del administrador de MySQL que pusimos durante la instalación.



Fuente: Elaboración Propia

**Figura 4.6.4.**Indicacion de contraseña para acceder a phpmyadmin

También podemos utilizar phpmyadmin desde cualquier PC de la red dirigiéndonos a la URL: `http://IP-del-servidor/phpmyadmin/`, sustituyendo IP-del-servidor por la IP del PC que hace de servidor, ejemplo: 192.168.1.10 o la que sea.



**Fuente: Elaboración Propia**

**Figura 4.6.5.** Administración de MySQL vía web con phpmyadmin

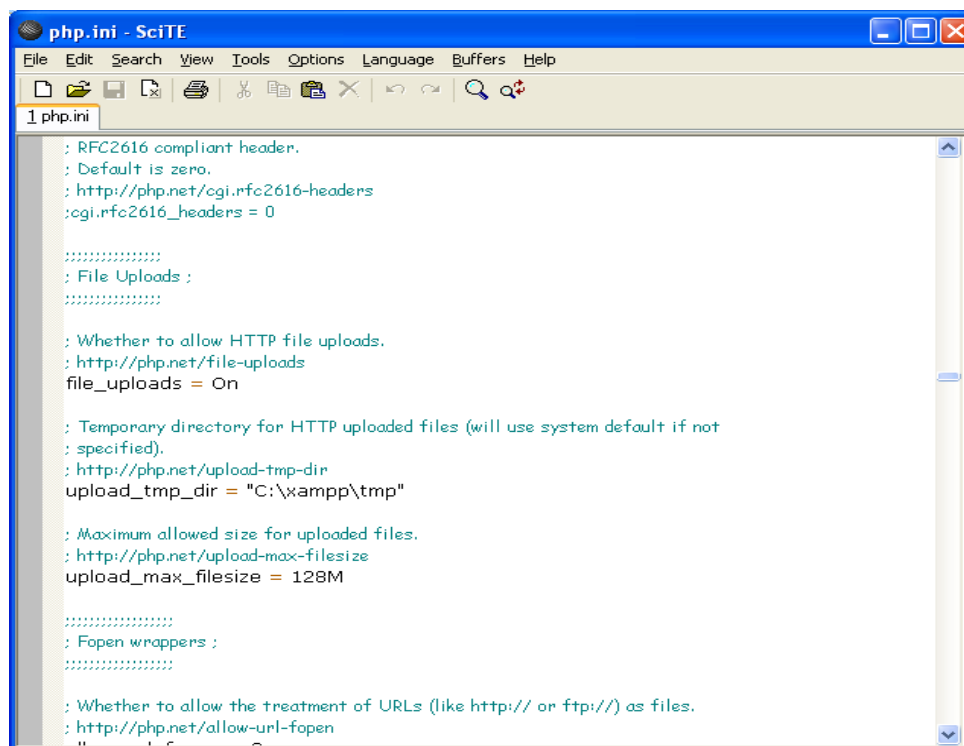
Con la aplicación web phpmyadmin, podemos realizar todas las funciones que permite mysql:

- Crear bases de datos y tablas
- Modificar bases de datos y tablas
- Insertar, actualizar y eliminar registros
- Realizar consultas
- Crear usuarios y establecer permisos

### Configuración de PHP con AppServ

Para realizar cambios en la configuración de PHP con AppServ, es necesario editar manualmente el archivo `C:WINDOWSphp.ini` con un editor de textos como el bloc de

notas o el editor -scite- y reiniciar el servicio de Apache desde el apartado –Servicios y Aplicaciones- del sistema.



Fuente: Elaboración Propia

**Figura 4.6.6.** Edición del archivo php.ini con el editor de textos scite

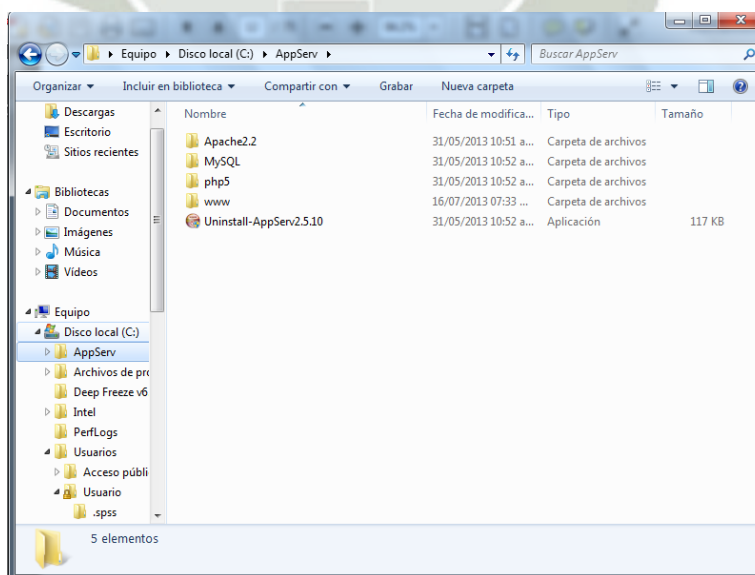
Para un uso normal del servidor, no es necesario realizar ninguna modificación en los cientos de parámetros que permite reconfigurar el archivo php.ini. Si algún usuario desea realizar algún cambio en la configuración, a continuación ponemos una lista de los parámetros más utilizados:

- **Safe Mode = Off** (Modo Seguro. Si el Modo seguro está desactivado, se habilitan todas las funciones del php. Para un uso educativo es mejor ser funcional y no activar el modo seguro).
- **Display errors = on** (Mostrar Errores. Muestra los errores en las mismas páginas, cuando les haya. Cuando hay errores en los scripts, es más fácil encontrarlos si se muestran en las páginas)

- **max\_execution\_time=30** (Tiempo máximo en segundos, de ejecución de un script)
- **memory\_limit = 128M** (Memoria máxima que puede utilizar un script durante su ejecución)
- **error\_reporting = E\_ALL & ~E\_NOTICE & ~E\_DEPRECATED** (Nivel de errores que se muestran durante la ejecución)
- **display\_errors = on** (Mostrar o no mostrar errores)
- **file\_uploads = on** (Permite subir archivos por http)
- **post\_max\_size=8M** (Tamaño máximo de datos que se pueden enviar al servidor mediante POST)
- **upload\_max\_filesize = 8M** (Tamaño máximo de archivo que se puede subir al servidor)
- **SMTP=localhost** (servidor de correo saliente para enviar emails desde aplicaciones PHP)

Para comprobar que todo funciona perfectamente se confirmó la marcha del módulo Apache e, inmediatamente, aviso mediante la operatividad signo de que el módulo se encuentra activo.

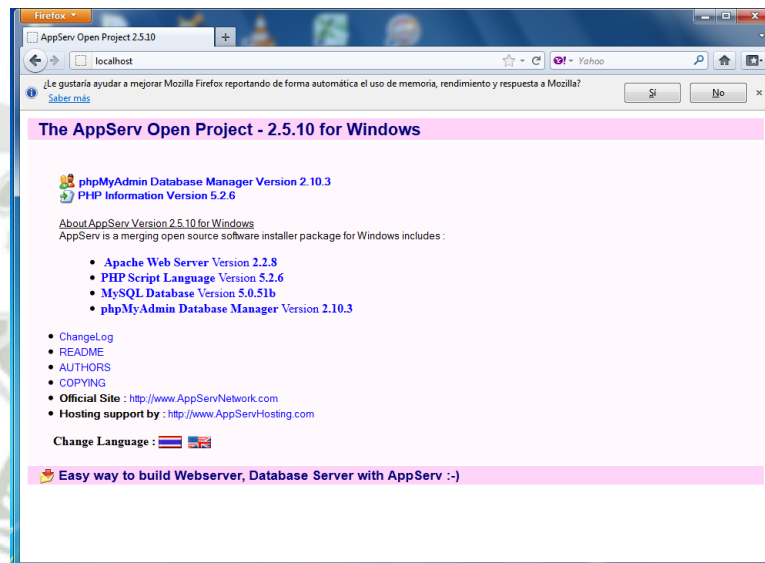
Lo mismo se hizo con el módulo de MySQL que muestra el mismo mensaje que en el caso anterior (Ver figura 4.6.7).



**Fuente: Elaboración Propia**

**Figura 4.6.7** Panel de operación de APPSERV

Después de lo anterior se procedió a abrir una ventana del algún navegador para acceder a la consola principal, en la barra de direcciones se introduce la dirección `http://localhost`, e inmediatamente se accede a la consola de administración. (Ver figura 4.6.8)

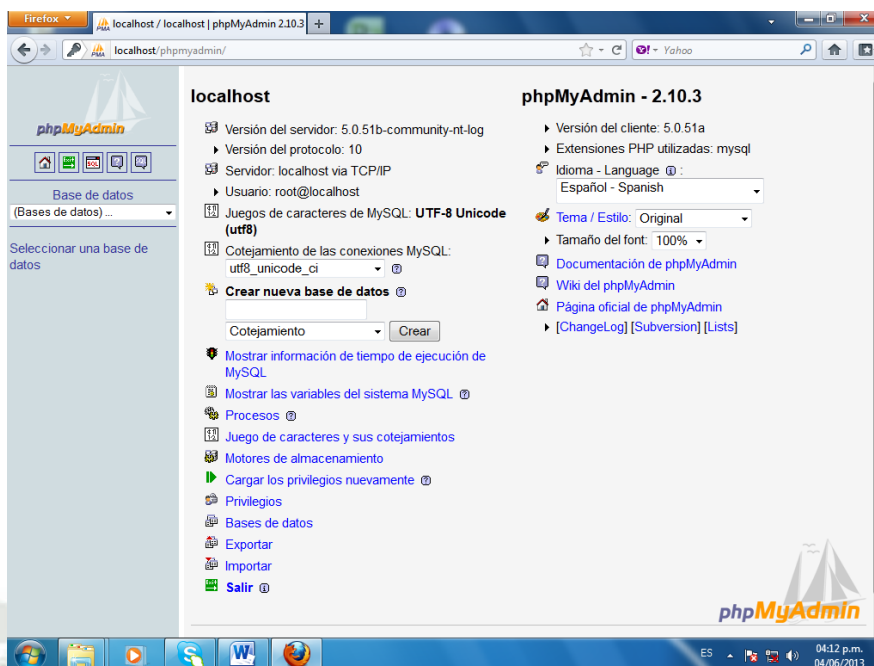


**Fuente: Elaboración Propia**

**Figura. 4.6.8.** Consola de administración de APPSERVER

La raíz del servidor Apache se encuentra en el directorio `C:\appserver\htdocs`. Todos los archivos que se encuentren dentro de este directorio son procesados por Apache cuando se accede mediante la URL especial `http://localhost`.

Desde la consola anterior, en el menú principal, se puede acceder a la aplicación phpMyAdmin que será la que nos facilitará el uso de MySQL. Esta herramienta es de gran ayuda en el momento de realizar las tablas de la base de datos, ya que ofrece la posibilidad de usar tanto el modo comando como el modo gráfico, agilizando el proceso de las operaciones esenciales de las bases de datos como actualizar, agregar o borrar datos de las tablas de forma rápida y confiable. En la figura 4.6.9. se muestra la página principal de la herramienta phpMyAdmin.



Fuente: Elaboración Propia

Fig.4.6.9. Página principal de phpmyadmin

Con la ayuda de esta herramienta se pueden crear las tablas necesarias para cada una de las secciones del sitio, el phpMyadmin por ser aplicación Web tiene la ventaja de poder ejecutarse en diferentes plataformas.

En la configuración por defecto se accede con el usuario *root* con una contraseña, obviamente, por motivos de seguridad, esta configuración se cambiará, posteriormente se agregará una contraseña renovándola cada cierto tiempo. Como la importancia de realizar páginas dinámicas radica en las base de datos, esa es la primera tarea que se llevó a cabo.

### Creación de la base de datos

Implica una tarea esencial en ella es donde se almacena todas las tablas que utiliza el sitio, necesarias tanto para la aplicación en línea como para las categorías y los productos respectivos. El nombre que tendrá la base de datos será “Panadería

Carmencita” y el comando SQL que permite su creación es `CREATE DATABASE` seguido del nombre de la base de datos:

```
CREATE DATABASE `Panaderia`;
```

Posteriormente se hizo la elaboración de cada tabla respectiva.

### **Elaboración de tablas en MySQL**

Estas se crearon con el comando `CREATE TABLE` seguido del nombre de la tabla, en la sentencia se describen los tipos de datos en cada campo, la llave primaria y Foránea si es el caso, así como si el campo soportará valores nulos.

#### **Tablas para la aplicación en línea**

Tabla para las categorías. Como uno de los propósitos del sitio Web es la venta de productos de panadería clasificados por categoría fue necesario elaborar primeramente la tabla que guarda el nombre de las categorías. Esto se logra con el comando ya mencionado. El nombre de la tabla para las categorías de la tienda es: `'categorias_tienda'`.

Si se requiere posteriormente cambiar el proveedor del hospedaje del sitio y migrar la base de datos es posible utilizar el condicional `'IF NOT EXISTS'` para que se cree la tabla si es que no se ha creado previamente, quedando así:

```
CREATE TABLE IF NOT EXISTS `categorias_tienda` (
  `id_cat_tienda` int(11) NOT NULL AUTO_INCREMENT,
  `nom_cat_tienda` varchar(30) CHARACTER SET latin1 NOT NULL,
  PRIMARY KEY (`id_cat_tienda`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=10 ;
```

Para verificar que se haya creado correctamente puede emplearse el comando `DESCRIBE categorias_tienda` que mostrará la siguiente información. (Ver figura 4.6.10).

Mostrando registros 0 - 1 (2 total, La consulta tardó 0.0020 seg)

consulta SQL: `DESCRIBE categorias_tienda`

[ Editar ] [ Crear código PHP ]

Operaciones sobre los resultados de la consulta

 Vista de impresión  Previsualización para imprimir (documento completo)

Mostrar : 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

|                          | Field          | Type        | Null | Key | Default | Extra          |
|--------------------------|----------------|-------------|------|-----|---------|----------------|
| <input type="checkbox"/> | id_cat_tienda  | int(11)     | NO   | PRI | NULL    | auto_increment |
| <input type="checkbox"/> | nom_cat_tienda | varchar(30) | NO   |     | NULL    |                |

☐ Marcar todos/as / ☐ Desmarcar todos Para los elementos que están marcados:  

Mostrar : 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

Fuente: Elaboración Propia

**Figura 4.6.10.** Resultado del comando DESCRIBE para las tabla categorias\_tienda

### Tabla para los productos.

Análogamente al procedimiento anterior se creó la tabla de los productos que contiene el modelo del producto, un campo para almacenar su clasificación, dicho campo es una llave foránea que apunta a la tabla de las categorías, esto con el fin de evitar que el usuario administrador pueda cometer errores al registrar la categoría asociada a un nuevos producto, también la tabla contiene un campo para la descripción del producto, otro para el precio, uno más para guardar el nombre del archivo de la imagen asociada. El código para la creación de dicha tabla es el que se muestra a continuación:

```
CREATE TABLE IF NOT EXISTS `productos` (
  `id_prod` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `modelo` varchar(20) NOT NULL,
  `id_cat_tienda` int(11) NOT NULL,
  `descripcion` text NOT NULL,
  `precio` float(10,2) NOT NULL,
  `img` varchar(20) NOT NULL,
  PRIMARY KEY (`id_prod`),
  KEY `id_cat_tienda` (`id_cat_tienda`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=36 ;
```

De igual manera, para verificar que la tabla esta creada correctamente se usa el comando DESCRIBE productos (Ver Figura 4.6.11).

Mostrando registros 0 - 5 (6 total, La consulta tardó 0.0021 seg)

consulta SQL: `DESCRIBE productos`

[ Editar ] [ Crear código PHP ]

Operaciones sobre los resultados de la consulta

Vista de impresión Previsualización para imprimir (documento completo)

Mostrar: 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

|                          | Field         | Type             | Null | Key | Default | Extra          |
|--------------------------|---------------|------------------|------|-----|---------|----------------|
| <input type="checkbox"/> | id_prod       | int(10) unsigned | NO   | PRI | NULL    | auto_increment |
| <input type="checkbox"/> | modelo        | varchar(20)      | NO   |     | NULL    |                |
| <input type="checkbox"/> | id_cat_tienda | int(11)          | NO   | MUL | NULL    |                |
| <input type="checkbox"/> | descripcion   | text             | NO   |     | NULL    |                |
| <input type="checkbox"/> | precio        | float(10,2)      | NO   |     | NULL    |                |
| <input type="checkbox"/> | img           | varchar(20)      | NO   |     | NULL    |                |

Marcar todos/as / Desmarcar todos Para los elementos que están marcados:

Mostrar: 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

Fuente: Elaboración Propia

**Figura 4.6.11.** Resultado del comando DESCRIBE para las tabla productos

Donde el campo "id\_cat\_tienda" por ser una llave o clave foránea se muestra como índice (Ver figura 4.6.12).

| Índices: ?                                |         |              |        |  |               | Espacio utilizado |       |       |
|-------------------------------------------|---------|--------------|--------|--|---------------|-------------------|-------|-------|
| Nombre de la clave                        | Tipo    | Cardinalidad | Acción |  | Campo         | Tipo              | Uso   |       |
| PRIMARY                                   | PRIMARY | 0            |        |  | id_prod       | Datos             | 0     | Bytes |
| id_cat_tienda                             | INDEX   | Ninguna      |        |  | id_cat_tienda | Índice            | 1,024 | Bytes |
| Crear un índice en 1 columna(s) Continuar |         |              |        |  |               | Total             | 1,024 | Bytes |

Fuente: Elaboración Propia

**Figura 4.6.12.** Muestra de claves de la tabla productos.

## Tablas para la galería

También es necesario tener un registro de los productos que se mostrarán en la galería de imágenes.

*Tabla para las categorías de la galería.* Las imágenes de los productos también se encuentran organizadas por categorías, por tanto para mostrarlas se requirió previamente construir en la base de datos una tabla para guardar los nombres de las categorías de la galería. Usando el procedimiento ya descrito el código queda como sigue:

```
CREATE TABLE IF NOT EXISTS `cat_galeria` (  
  `id_gal` int(11) NOT NULL AUTO_INCREMENT,  
  `nom_gal` varchar(30) NOT NULL,  
  PRIMARY KEY (`id_gal`)  
  ) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=8 ;
```

Verificando su correcta creación con `DESCRIBE cat_galeria` (Ver Figura 4.6.13)

Mostrando registros 0 - 1 (2 total, La consulta tardó 0.0021 seg)

consulta SQL: [DESCRIBE cat\\_galeria](#) [\[ Editar \]](#) [\[ Crear código PHP \]](#)

Operaciones sobre los resultados de la consulta

[Vista de impresión](#) [Previsualización para imprimir \(documento completo\)](#)

Mostrar: 30 filas empezando de 0 y repetir los encabezados cada 100 celdas

|                          | Field   | Type        | Null | Key | Default | Extra          |
|--------------------------|---------|-------------|------|-----|---------|----------------|
| <input type="checkbox"/> | id_gal  | int(11)     | NO   | PRI | NULL    | auto_increment |
| <input type="checkbox"/> | nom_gal | varchar(30) | NO   |     | NULL    |                |

[Marcar todos/as](#) / [Desmarcar todos](#) Para los elementos que están marcados: [✎](#) [✖](#)

Mostrar: 30 filas empezando de 0 y repetir los encabezados cada 100 celdas

Fuente: Elaboración Propia

**Figura 4.6.13.** Tabla para las categorías de la galería

*Tabla para los productos de la galería.* Del mismo modo fue necesario crear una tabla de los productos que se exhiben en la galería, dicha tabla, a diferencia de la tienda, solamente almacena el nombre del archivo de la imagen del producto, su categoría a la que pertenece y la descripción. Por tanto se creó dicha tabla de la siguiente manera:

```
CREATE TABLE IF NOT EXISTS `prod_galeria` (
  `id` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `img` varchar(20) NOT NULL,
  `id_gal` tinyint(4) NOT NULL,
  `desc_gal` text NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=3
```

Verificamos con DESCRIBE prod\_galeria (ver figura 4. 6.14)

Mostrando registros 0 - 3 (4 total, La consulta tardó 0.0023 seg)

consulta SQL: `DESCRIBE prod_galeria`

[ Editar ] [ Crear código PHP ]

Operaciones sobre los resultados de la consulta

Vista de impresión Previsualización para imprimir (documento completo)

Mostrar: 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

|                          | Field    | Type             | Null | Key | Default | Extra          |
|--------------------------|----------|------------------|------|-----|---------|----------------|
| <input type="checkbox"/> | id       | int(11) unsigned | NO   | PRI | NULL    | auto_increment |
| <input type="checkbox"/> | img      | varchar(20)      | NO   |     | NULL    |                |
| <input type="checkbox"/> | id_gal   | tinyint(4)       | NO   |     | NULL    |                |
| <input type="checkbox"/> | desc_gal | text             | NO   |     | NULL    |                |

Marcar todos/as / Desmarcar todos Para los elementos que están marcados:

Mostrar: 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

Fuente: Elaboración Propia

**Figura 4. 6.14.** Resultado del comando DESCRIBE para la tabla “prod\_galeria”.

### Área administrativa

En esta sección también fue necesario contar con un registro de los usuarios con privilegios de administrador, por estar basado el proyecto en una autenticación del tipo username password solamente es necesario contar con una tabla que contenga un campo para el nombre de usuario y otro para el password. Usando el mismo procedimiento de las tablas anteriores queda el código SQL como sigue:

```
CREATE TABLE IF NOT EXISTS `admin` (
  `username` char(16) NOT NULL,
  `password` char(40) NOT NULL,
  PRIMARY KEY (`username`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Verificando con DESCRIBE admin (Ver figura 4. 6.15)

Mostrando registros 0 - 1 (2 total, La consulta tardó 0.0027 seg)

consulta SQL: `DESCRIBE admin`

[ Editar ] [ Crear código PHP ]

Operaciones sobre los resultados de la consulta

Vista de impresión Previsualización para imprimir (documento completo)

Mostrar: 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

|                          | Field    | Type     | Null | Key | Default | Extra |
|--------------------------|----------|----------|------|-----|---------|-------|
| <input type="checkbox"/> | username | char(16) | NO   | PRI | NULL    |       |
| <input type="checkbox"/> | password | char(40) | NO   |     | NULL    |       |

Marcar todos/as / Desmarcar todos Para los elementos que están marcados:

Mostrar: 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

Fuente: Elaboración Propia

Figura 4. 6.15. Resultado del comando DESCRIBE para la tabla “admin”

### Eliminación de tablas

Si se desea borrar una tabla existente, así como sus datos, introducir el comando: DROP TABLE <tabla>;

### Consulta de la base de datos

La manipulación de datos del lenguaje de consultas estructurado, se resume en lo siguientes cuatro comandos: SELECT, INSERT, UPDATE y DELETE. Existe una serie de reglas comunes de sintaxis para todas las instrucciones. A continuación se muestran algunas de estas:

Si se utilizara más de una tabla en la cláusula FROM, todos los campos con el mismo nombre, deben ir precedidos por el nombre de la tabla en donde se vayan a utilizar.

Las instrucciones múltiples de una cláusula WHERE deben conectarse por medio de alguna palabra clave: AND u OR. Todas las instrucciones SQL deben tener un verbo, una cláusula FROM o INTO y usualmente una lista de parámetros de nombres de columna.

**SELECT.** Esta instrucción SELECT es probablemente la más utilizada de las instrucciones SQL. Ésta devuelve únicamente los datos contenidos en la base de datos.

MySQL probablemente procesa este tipo de consulta más rápido que cualquier otra base de datos. A continuación se muestra su sintaxis.

```
SELECT <columna> FROM <tabla> WHERE <condición>;
```

El nombre de columna puede ser de una o más columnas que pertenezcan a una tabla en la cláusula FROM. El orden en que se especifiquen los nombres de las columnas es con el que se mostrarán. Si se utiliza el comodín \* se seleccionarán todas las columnas.

**INSERT INTO.** Este comando añade nuevos registros a una tabla existente. El formato es básicamente el mismo que el de la instrucción SELECT.

Ejemplo:

```
INSERT INTO `panaderia`.`admin` (`username`, `password` )  
VALUES ('administrator', 'contraseña');
```

El nombre de la tabla en la que se realizará la inserción se pone después de la palabra reservada INTO. Después de la palabra clave VALUES deberá ir la lista de valores que se quieren insertar. Si se inserta un registro de esta forma, debe tener un valor para cada columna y el orden de los datos deberá concordar con el orden en que se declararon las columnas en la base de datos.

**UPDATE.** El comando UPDATE permite editar valores de datos existentes. Este comando es muy similar a SELECT, pero permite manipular los valores seleccionados. La sintaxis es la siguiente:

```
UPDATE <tabla> SET <columna> = <valor> WHERE <condición>;
```

Así como todas las instrucciones SQL, el verbo reservado va primero. A éste le sigue el nombre de la tabla que se manipulará. A continuación se emplea la palabra reservada

SET y, por último, se coloca la lista de nombres de columna y valores que se desean establecer. Pueden existir valores múltiples después del comando SET. Cada par columna/valor debe estar separado por una coma. La cláusula WHERE designa qué registros serán actualizados.

**DELETE.** La instrucción DELETE es muy similar a SELECT, la única diferencia es que en lugar de seleccionar registros para verlos, borra esos registros. Debe tenerse precaución al usar esta instrucción y evitar accidentalmente borrar registros. Puede usarse primero la instrucción SELECT y así asegurarse de que solamente se borrará lo deseado.

El verbo reservado se escribe primero, enseguida la palabra reservada FROM, seguida del nombre de la tabla y después la cláusula WHERE donde se establece la condición que deben cumplir los registros a borrarse. Si no se especifica la condición se borrarán todos los registros de la tabla:

```
DELETE FROM <tabla> WHERE <condición>;
```

### Respaldo de la base de datos

Consideramos que respaldar la base de datos es parte fundamental de la administración de nuestro sistema de bases de datos. Los motivos van desde cualquier incidente físico del equipo, hasta algún daño accidental por parte de algún usuario. Tan importante como el respaldo es la restauración de los datos.

En phpMyadmin fue posible manejar el proceso de respaldo SQL mediante la opción “Exportar”, que se encuentra en el menú general en el área checkbox de “Enviar” para generar un archivo descargable (ver figura 4. 6.16).

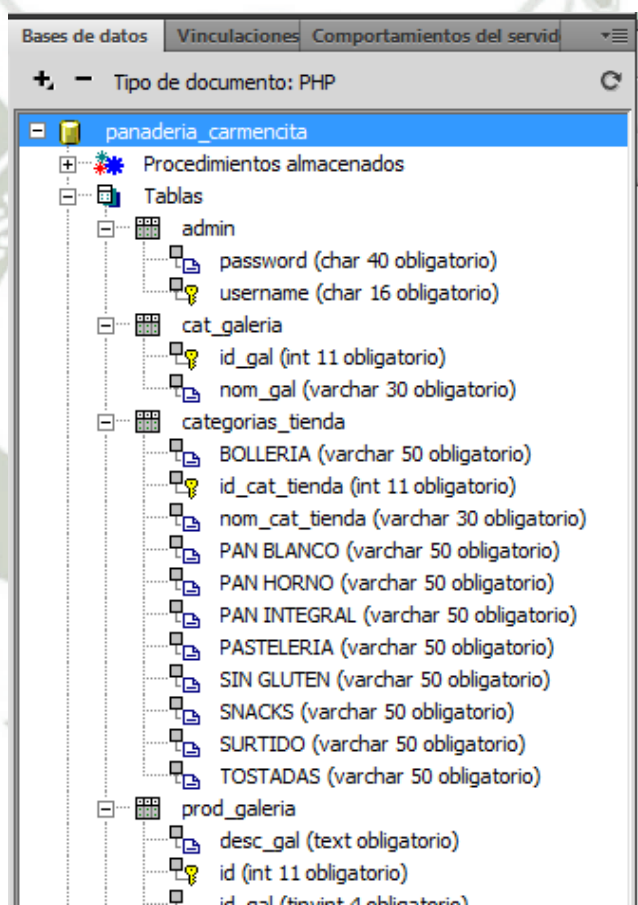


Fuente: Elaboración Propia

**Figura 4. 6.16.** Menú de phpMyadmin para exportar la BD

Con esta opción el servidor exporta a un archivo de texto con extensión ".sql" que el navegador recibirá para dar la opción de "Grabar como" o "Save as" en la PC, una vez que se descargue el archivo con terminación ".sql" se podrá ver su contenido en cualquier editor de texto en caso de querer revisar el contenido de la descarga.

Finalmente después de haber creado todas las tablas hemos realizado una conexión a la base de datos “panadería carmencita” con DREAWEAVER y de esta manera se pudo tener una vista general de todas las tablas involucradas, en forma de árbol el cual se muestra en la figura 4. 6.17.



Fuente: Elaboración Propia

**Figura 4. 6.17.** Árbol de tablas de la base de datos  
“Panaderia\_carmencita”

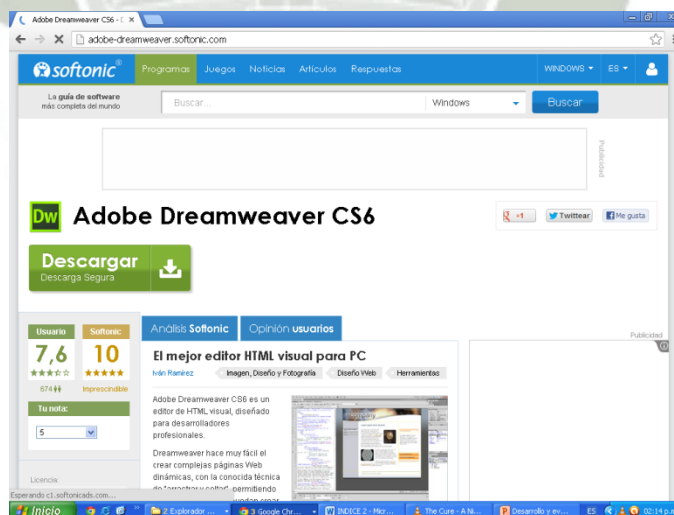
## 4.7. INSTALACIÓN Y CONFIGURACIÓN DEL SITIO WEB

Para comenzar con la construcción del sitio fue necesaria la instalación y configuración adecuada de las herramientas a utilizar.

### Instalación de herramientas

Como ya se mencionó hemos utilizado el programa Dreamweaver en la versión CS5 para la escritura de código, debido a su gran integración de herramientas y practicidad. Por tanto se instaló una versión principal de Adobe (Ver figura 4.7.1). Siguiendo estos pasos para instalar en un sistema Windows:

1. Se introdujo el CD de Dreamweaver en la unidad de CD-ROM del ordenador.
2. Disponemos de las opciones siguientes:
  - Elegimos Inicio > Ejecutar. Hacemos clic en Examinar y elegimos el archivo Dreamweaver CS5 Installer.exe del CD de Dreamweaver. En el cuadro de diálogo. Ejecutar, hacemos clic en Aceptar para comenzar la instalación.
3. Seguimos las instrucciones que aparecen en la pantalla.
4. Reiniciamos el ordenador si el sistema lo solicita.



Fuente: Elaboración Propia

**Figura 4.7.1.** Página de descarga de la herramienta Dreamweaver CS5

Posteriormente, mediante el asistente de configuración, se establece los parámetros básicos de instalación.

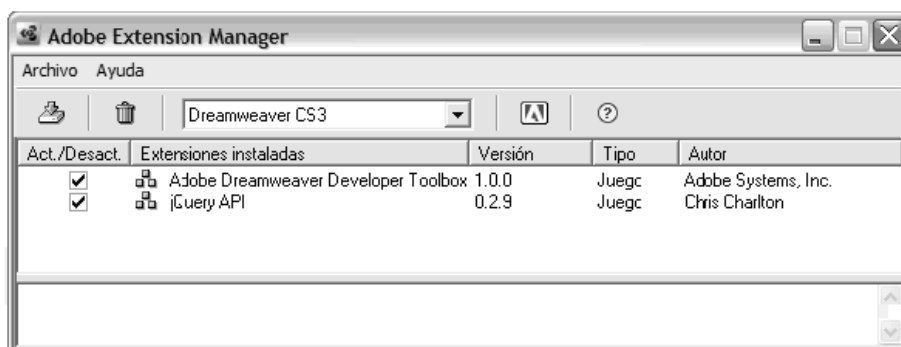
Cuando la instalación finaliza automáticamente se ejecuta la aplicación. En la figura 4.7.2 se muestra la vista principal de la herramienta.



**Fuente: Elaboración Propia**

**Figura 4.7.2.** Vista principal de Dreamweaver CS5

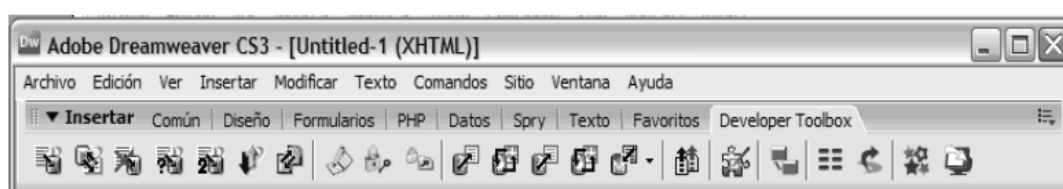
Para los propósitos de la presente Tesis fue necesario instalar las extensiones Developer Toolbox (para operaciones con la base de datos) y jQuery API (para la codificación de la funcionalidad AJAX). En la figura 4.7.3 se muestra el administrador de extensiones con las extensiones ya descargadas e instaladas.



**Fuente:** Elaboración propia

**Figura 4.7.3.** Administrador de extensiones de DW

Después de reiniciar el producto se muestran las herramientas adicionales en la barra de herramientas de DW (Figura 4.7.4.).



**Fuente:** Elaboración Propia

**Figura 4.7.4.** Herramientas para la extensión Developer Toolbox.

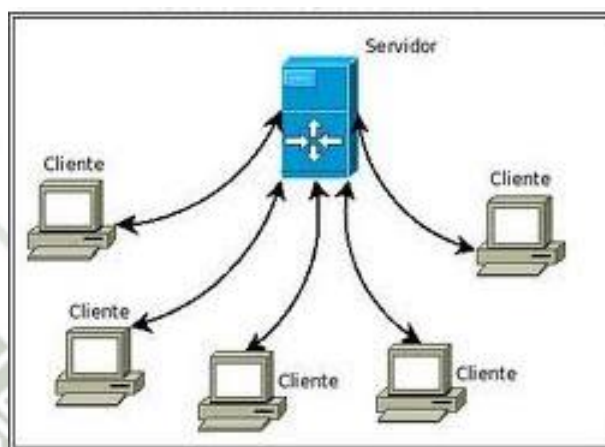
Otra herramienta que se emplea es APPSERV, su instalación y aplicación se ha explicado en el apartado del Back-End, a este momento se verifica que esté funcionando correctamente.

### Creación del sitio

Después de haber instalado adecuadamente las herramientas descritas anteriormente, se procedió a dar inicio a la creación del sitio web para la microempresa de panadería “Carmencita”. Sitio Web que exactamente es una carpeta que contiene los archivos para su funcionamiento.

El método más difundido para crear un sitio Web utilizando DreamWeaver consiste en:

- (1) Crear y editar páginas en el disco local, guardándolas en nuestro propio equipo.
- (2) Después conectarnos a un servidor Web remoto donde se alojarán.
- (3) Cargar una copia del sitio Web y los documentos (páginas, imágenes, js, etc.) en el servidor para que sean de dominio público y
- (4) que el servidor los “reparta”. (Ver Figura 4.7.5)



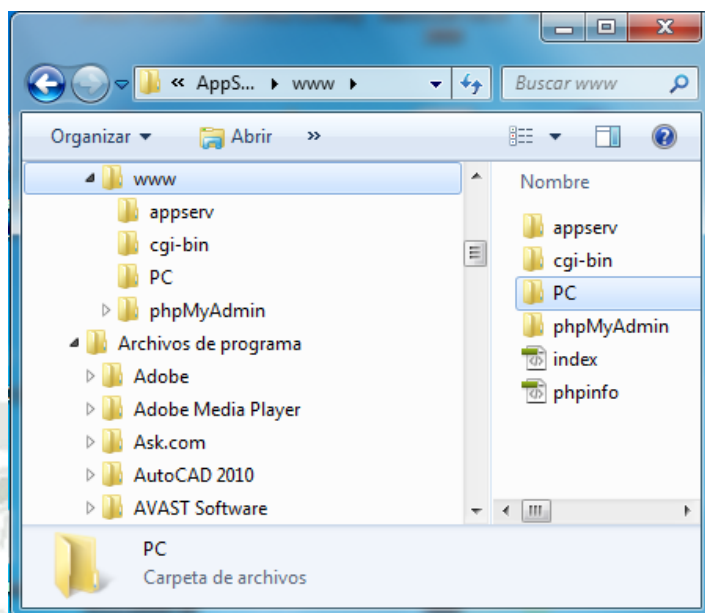
Fuente: Elaboración Propia

**Figura 4.7.5.** Método típico para la difusión de sitios Web

Lo primero que se realizó fue crear la carpeta del sitio que contenga los archivos necesarios para después subirlos al servidor. Dicha carpeta se denomina carpeta raíz.

La carpeta del sitio se llamará “PC” (*Panaderia carmencita*), esta carpeta se crea en la raíz de la carpeta principal “www” del servidor de prueba que se configuró previamente en la instalación de APPSERV. (Ver Figura 4.7.6).

El nombre de la carpeta no necesariamente se relaciona con el nombre del sitio que se configura luego en Dreamweaver.

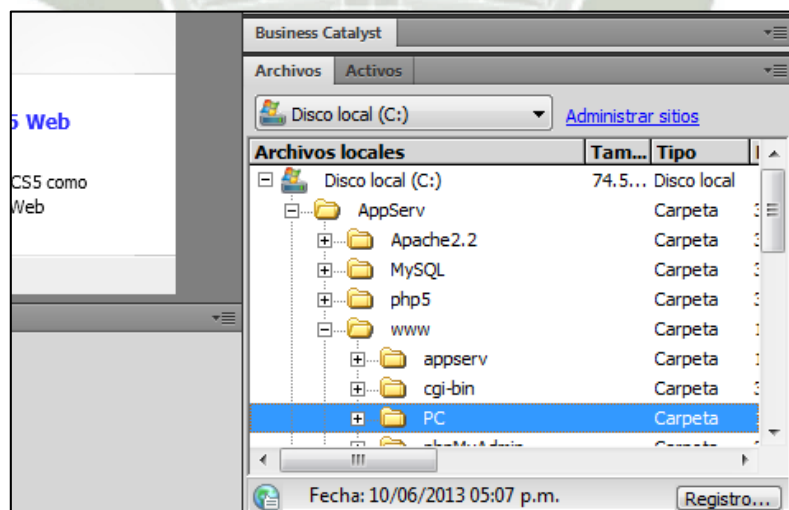


Fuente: Elaboración Propia

**Figura 4.7.6.** Creación de la carpeta del Sitio Web

Después de crear la carpeta se configuró propiamente el sitio en Dreamweaver mediante los siguientes pasos:

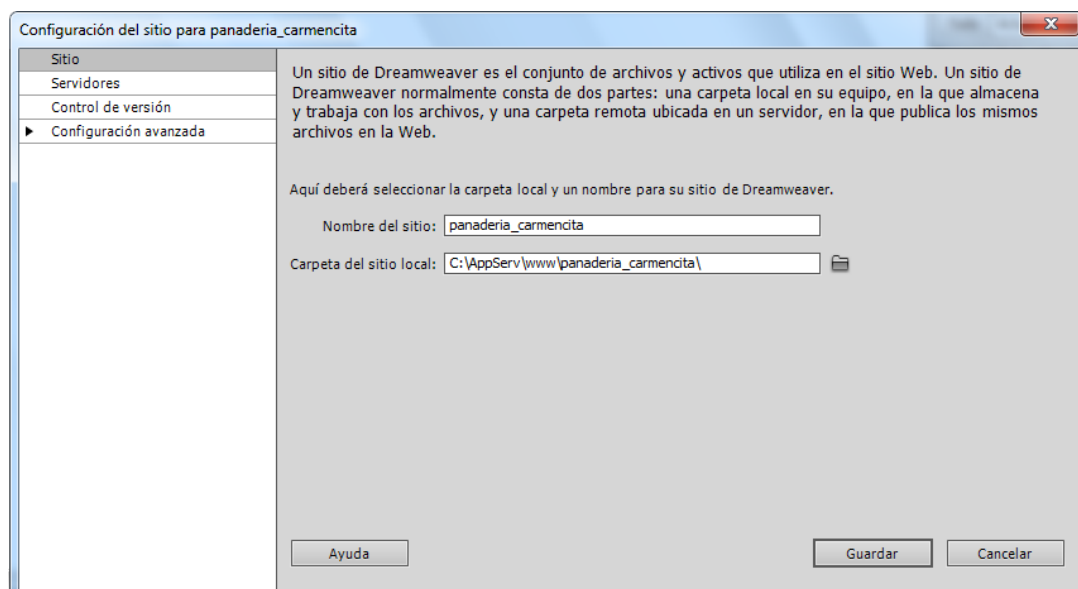
- Localizando la carpeta raíz que se creó previamente.
- Abrir la herramienta de administración de sitios de DW (Ver Fig. 4.7.7).
- Configurar el sitio



Fuente: Elaboración Propia

**Figura 4.7.7.** Ubicación del administrador de sitios en DW

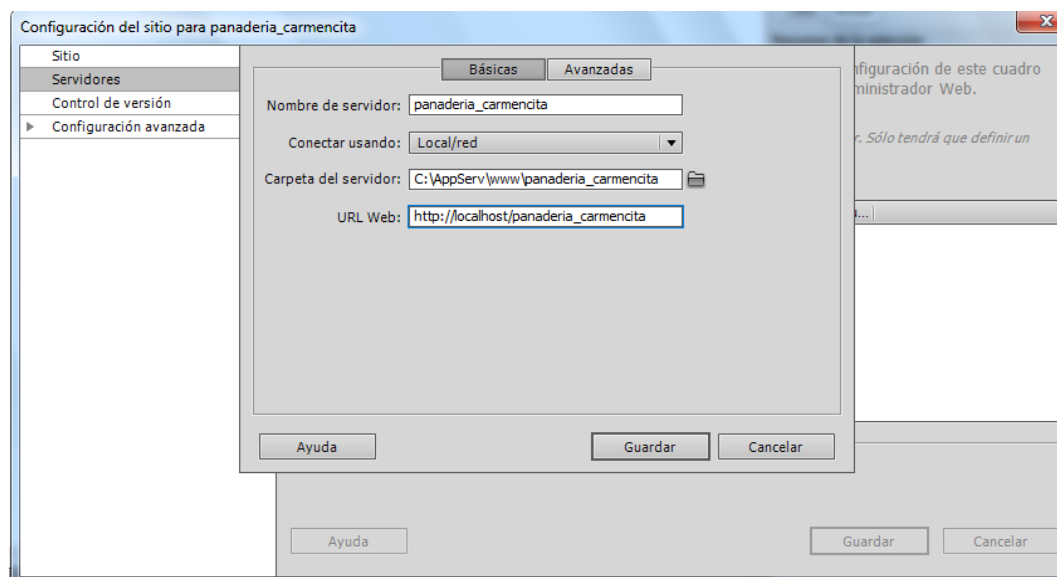
*Configuración de datos locales.* Se elige el nombre, en este caso se le llamo de la misma forma que la carpeta raíz, posteriormente se ubicó la carpeta raíz local e introdujo la dirección http para poder acceder a esta ubicación desde el navegador.  
(Ver Fig. 4.7.8).



**Fuente:** Elaboración Propia

**Figura 4.7.8.** Configuración de los datos locales

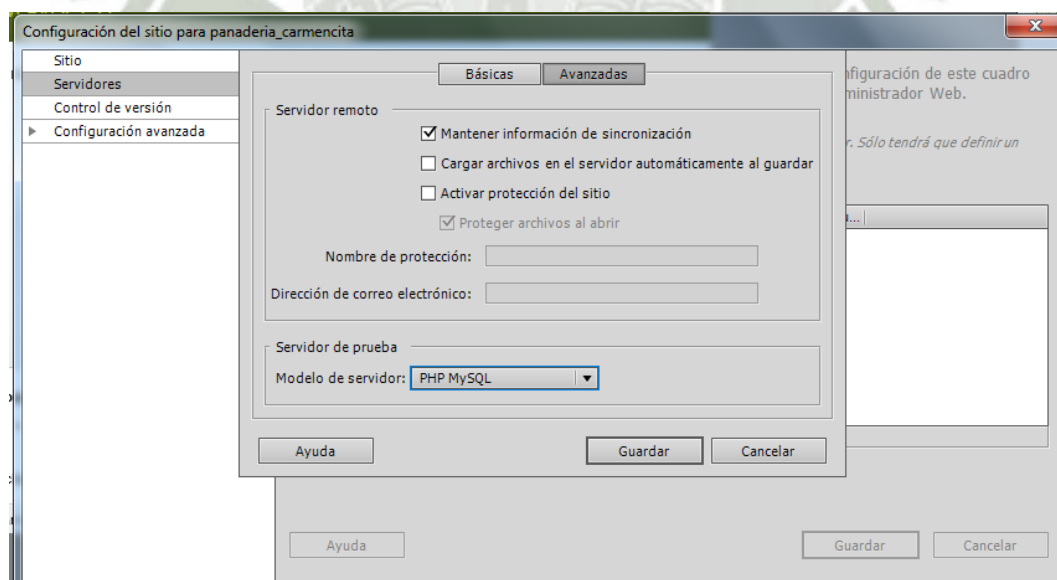
*Configuración de datos remotos.* Se configura el tipo de acceso por el cual se realizó el enlace al sitio, en este caso por ser servidor de prueba local fue por medio de la red intranet (opción Local/red). Y se ubicó la carpeta remota (en el servidor local) que contiene al sitio (Ver Fig. 4.7.9).



Fuente: Elaboración Propia

**Figura. 4.7.9.** Configuración de los Datos remotos

*Servidor de prueba.* Se determinó el tipo de servidor en base a los lenguajes y sistema de archivos que soporte. En este caso se eligió la opción de PHP y MySQL con acceso por red (Ver Fig. 4.7.10).



Fuente: Elaboración Propia

**Figura 4.7.10.** Configuración del Servidor de prueba.

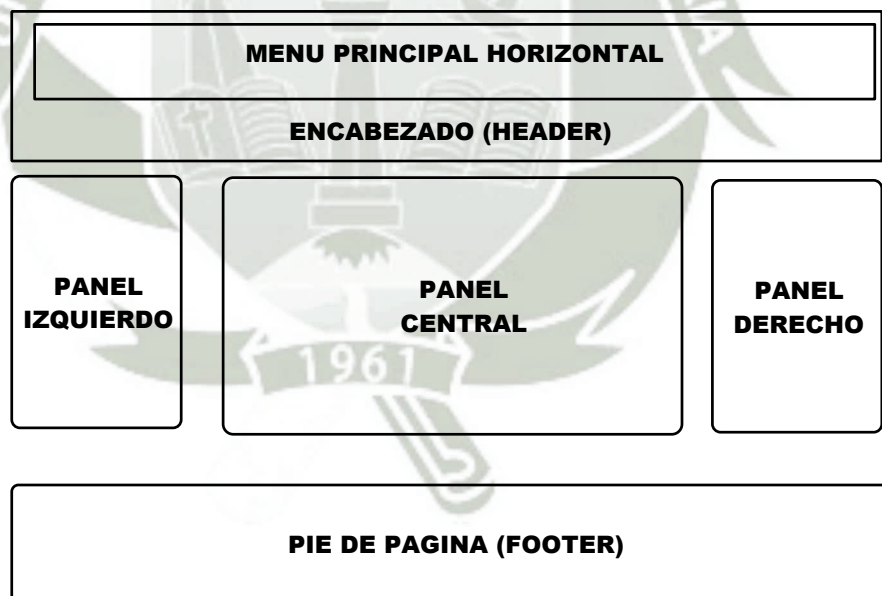
Esta configuración es suficiente para los propósitos del presente trabajo. Después de haber realizado los procedimientos anteriores aparece en el panel Archivos de DW la carpeta raíz para trabajar con los archivos del sitio (la cual está vacía). En otras palabras, este sitio en DW es un acceso a la carpeta del sitio.

#### 4.8. DISEÑO Y CONSTRUCCIÓN DEL FRONT - END

Después de haber diseñado e implementado el back-end y para poder obtener su utilidad se procedió a realizar el Front-end, que interactúa con todo los datos anteriormente descritos.

##### 4.8.1. DISEÑO DEL FRONT-END

Se procedió a realizar un bosquejo de todo el sitio Web así como de los paneles que se requieren, todos estos con base en los requisitos planteados. En la figura 4.8.1.1 se muestran los paneles que contiene la página principal, sobre la cual se implementan los módulos respectivos.



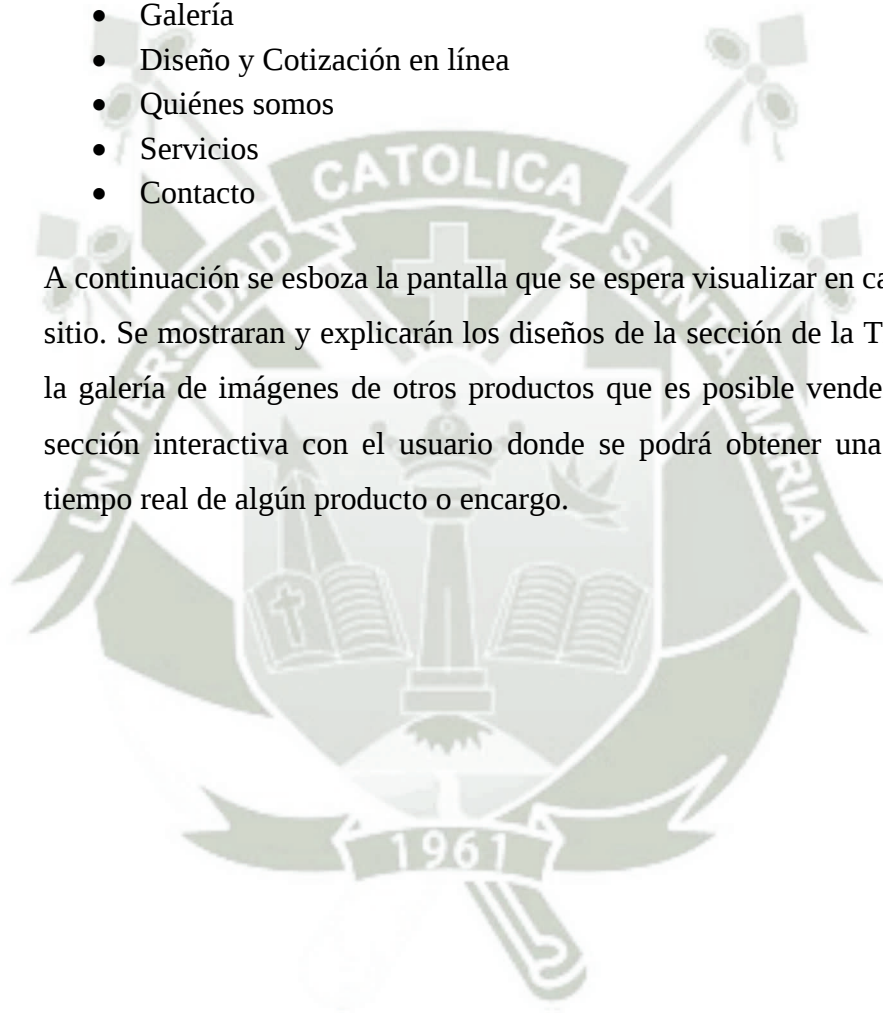
Fuente: Elaboración Propia

Figura 4.8.1.1. Paneles principales del sitio web.

Se cuenta con cinco paneles principales, cada panel muestra de forma asíncrona las diferentes páginas o archivos PHP correspondientes que se llamarán cuando sean necesarios mediante enlaces para evitar el recargo de la página completa. El modelo de la página web de la Microempresa de panadería “Carmencita” cuenta con las siguientes elementos que lo conforma

- Inicio
- Tienda en línea
- Galería
- Diseño y Cotización en línea
- Quiénes somos
- Servicios
- Contacto

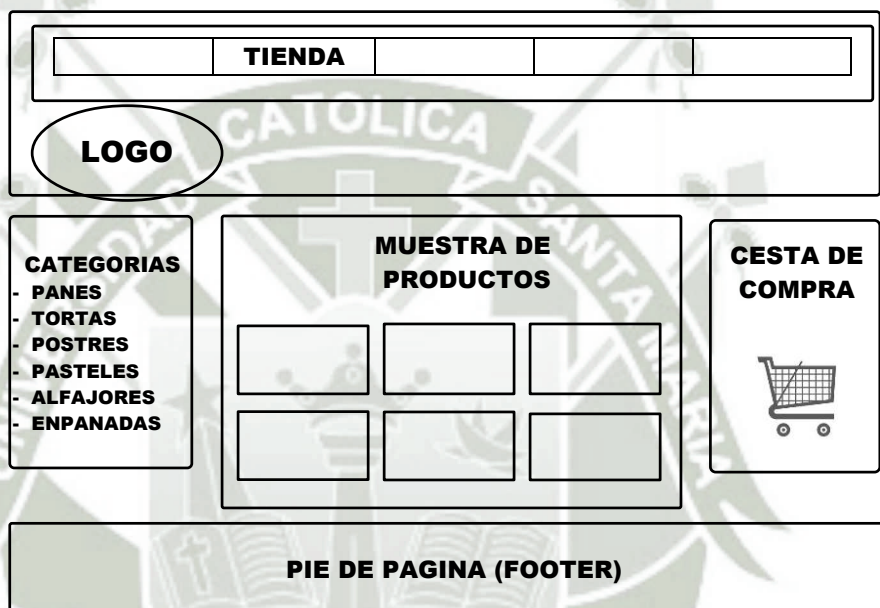
A continuación se esboza la pantalla que se espera visualizar en cada sección del sitio. Se mostrarán y explicarán los diseños de la sección de la Tienda en línea, la galería de imágenes de otros productos que es posible vender en línea y la sección interactiva con el usuario donde se podrá obtener una cotización en tiempo real de algún producto o encargo.



### Tienda en línea

**Objetivo:** Mostrar, desplegar las imágenes de los productos de la lista de categorías para que el cliente siempre tenga conocimiento de los artículos en oferta

En la Figura 4.8.1.2 se muestra la pantalla que se pretende visualizar cuando se elija tienda en línea.



Fuente: Elaboración Propia

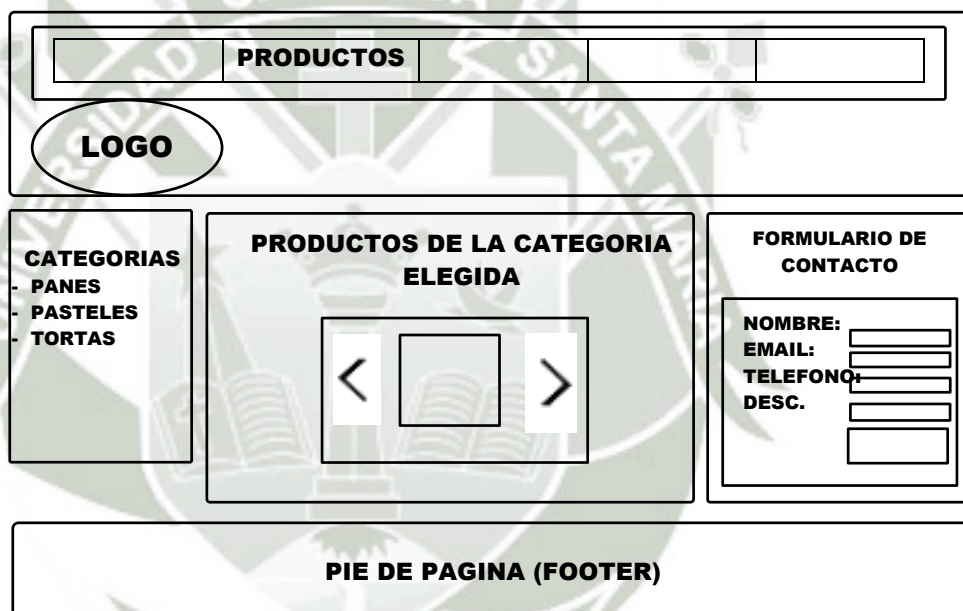
**Figura 4.8.1.2** Pantalla principal para la tienda en línea

En el Panel Izquierdo se muestra la lista de categorías de los productos en venta, cada elemento de la lista es un vínculo, que al hacerle click despliega las imágenes de los productos de la categoría elegida dentro del panel central, al mismo tiempo en el panel derecho aparecerá la cesta de compra o carrito de compras donde de forma visual el cliente siempre tendrá conocimiento de los artículos que va ingresando.

## Galería

**Objetivo:** Visualizar las imágenes en un tamaño mayor para mejor detalle y mostrar el formulario de contacto.

Existen productos o servicios que la empresa de panadería “Carmencita” desea se muestren a los usuarios a manera de galería de imágenes, esta galería también se mostrará por categorías, y además se ha de agregar un formulario de contacto para que el usuario pueda solicitar alguna cotización de algún producto que visualiza en la galería. (Ver figura 4.8.1.3)



PRODUCTOS

LOGO

CATEGORIAS

- PANES
- PASTELES
- TORTAS

PRODUCTOS DE LA CATEGORIA ELEGIDA

FORMULARIO DE CONTACTO

NOMBRE:

EMAIL:

TELEFONO:

DESC.

PIE DE PAGINA (FOOTER)

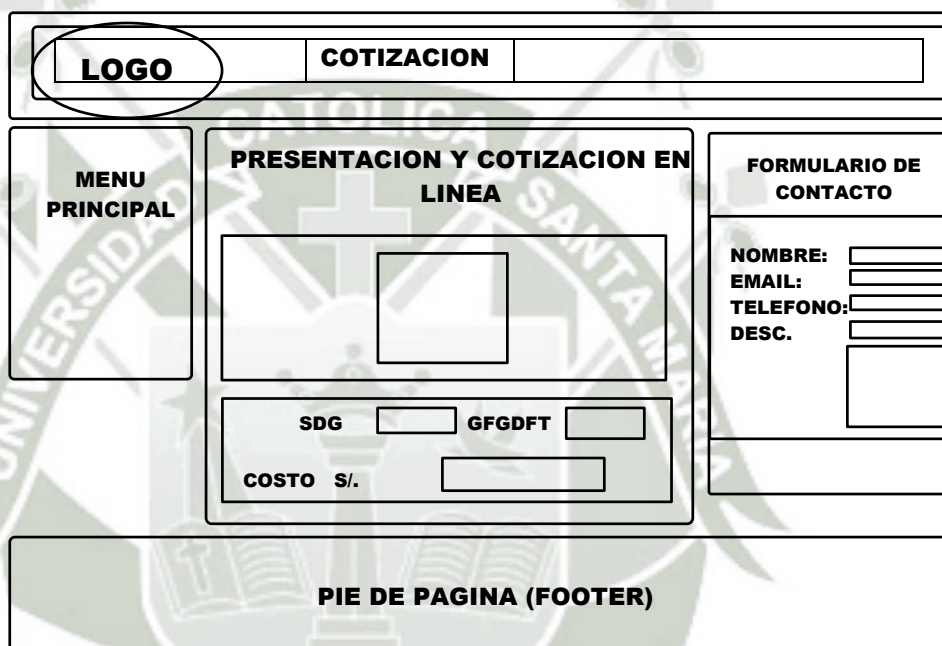
Fuente: Elaboración Propia

Figura 4.8.1.3 Pantalla principal para la Galería de imágenes

### Presentación y cotización en línea

**Objetivo:** Permitir a la empresa modificar las dimensiones y el área de ciertos artículos, y de esta manera obtener visualmente el costo en tiempo real

Esta sección es una respuesta a la necesidad de solicitud de cotización en línea en tiempo real por parte de los clientes. Para algunos productos la cotización de un producto depende del área requerida del producto final. (Ver Figura 4.8.1.4).



|                               |                                                                                                                                                                   |                                                                                                                                                              |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>LOGO</b>                   | <b>COTIZACION</b>                                                                                                                                                 |                                                                                                                                                              |
| <b>MENU PRINCIPAL</b>         | <b>PRESENTACION Y COTIZACION EN LINEA</b><br><div></div> <div>SDG <input type="text"/> GFGDFT <input type="text"/></div> <div>COSTO S/ <input type="text"/></div> | <b>FORMULARIO DE CONTACTO</b><br>NOMBRE: <input type="text"/><br>EMAIL: <input type="text"/><br>TELEFONO: <input type="text"/><br>DESC. <input type="text"/> |
| <b>PIE DE PAGINA (FOOTER)</b> |                                                                                                                                                                   |                                                                                                                                                              |

Fuente: Elaboración Propia

**Figura 4.8.1.4** Pantalla principal para el diseño y cotización de productos en línea

Además el cliente o visitante va a contar con la sección de un Formulario de contacto para solicitar en cualquier momento una cotización para las necesidades particulares del cliente.

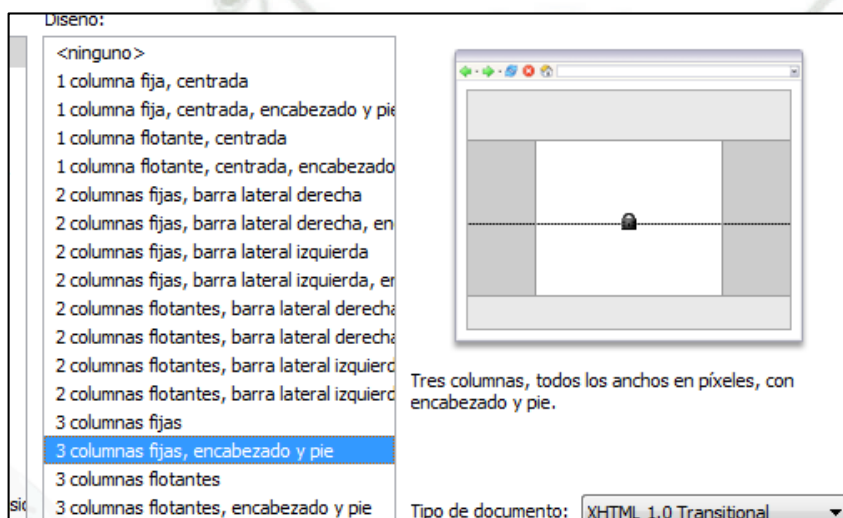
Para las secciones restantes (Quiénes somos, Servicio y Contacto) solo se utiliza información de presentación de la empresa que se mostrará en el Panel Central.

#### 4.8.2. CONSTRUCCION DEL FRONT-END

### Maquetado

A continuación se realizó el maquetado de la página principal con base en el bosquejo del inicio de la sección.

Debido a la practicidad que ofrece DW se eligió una plantilla entre las pre configuradas que cumpla lo más posible con los propósitos perseguidos en cuando a maquetación y distribución de las secciones (Ver figura 4.8.2.1). Ésta solo nos sirve como guía en cuanto a distribución (en píxeles) de una visión general de la página. Posteriormente se adecuo a las necesidades particulares.



Fuente: Elaboración Propia

**Fig. 4.8.2.1** Plantilla DW de tres columnas fijas, con encabezado y pie.

Para mantener la separación de la lógica y la presentación se trabajó aparte con un archivo de diseño (CSS) que se llamará “*estilos.css*”, que de igual manera se guarda en la carpeta raíz. Posteriormente se hace referencia a este archivo en la página principal del sitio o las páginas que requieran de este diseño.

### 4.8.3. TIENDA

A continuación se describe la manera en la que se realizó los módulos o secciones de la tienda .Cada módulo es un archivo o conjunto de archivos que se guardan en la carpeta raíz.

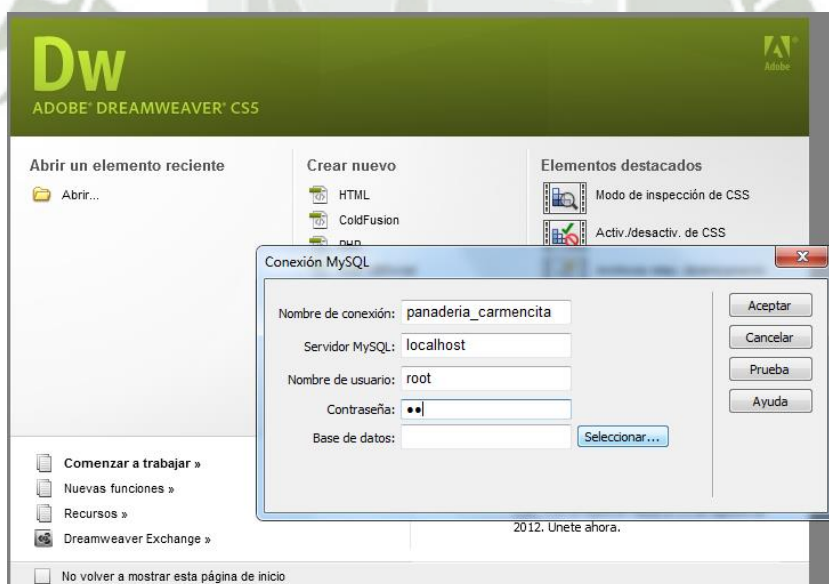
### Generación del catálogo en línea

Para poder implementar el catálogo de los productos que se venderán en línea, primero es necesario mostrar la lista de las categorías de los productos que se desean vender.

Esta lista de categorías se obtuvo de la base de datos previamente creada, para lo cual fue necesario establecer una conexión a la base de datos llamada Panadería “Carmencita”.

### Conexión a la base de datos

Esta conexión a la base de datos se realizó con ayuda de las herramientas básicas de Dreamweaver. En la sección de Bases de datos del panel de aplicación, al pulsar se abrirá un cuadro de diálogo donde se pueden establecer los parámetros necesarios para la conexión. (Ver figura 4.8.3.1)



**Fuente:** Elaboración Propia

**Figura 4.8.3.1.** Establecimiento de los parámetros de conexión

Después de haber pulsado el botón aceptar se genera de forma automática una carpeta con un archivo PHP con el nombre de la conexión que contendrá los parámetros necesarios para llevar a cabo dicha conexión.

El contenido del archivo se muestra en el listado 4.8.3.1

```
<?php
# FileName="Connection_php_mysql.htm"
# Type="MYSQL"
# HTTP="true"
$hostname_panaderia_carmencita = "localhost";
$databse_panaderia_carmencita = "panaderia_carmencita";
$username_panaderia_carmencita = "root";
$password_panaderia_carmencita = "***";
$panaderia_carmencita =
mysql_pconnect($hostname_panaderia_carmencita,
$username_panaderia_carmencita, $password_panaderia_carmencita)
or trigger_error(mysql_error(),E_USER_ERROR);
?>
```

**Listado 4.8.3.1.** Código generado del archivo *panadería\_carmencita.php*

Con base en el archivo anterior se llevan a cabo todas las interacciones con la base de datos, por lo tanto cualquier archivo que requiera algún conjunto de registros de la base de datos debe hacer referencia a este archivo con la siguiente línea de código:

```
<?php require_once('Connections/panaderia_carmencita.php'); ?>
```

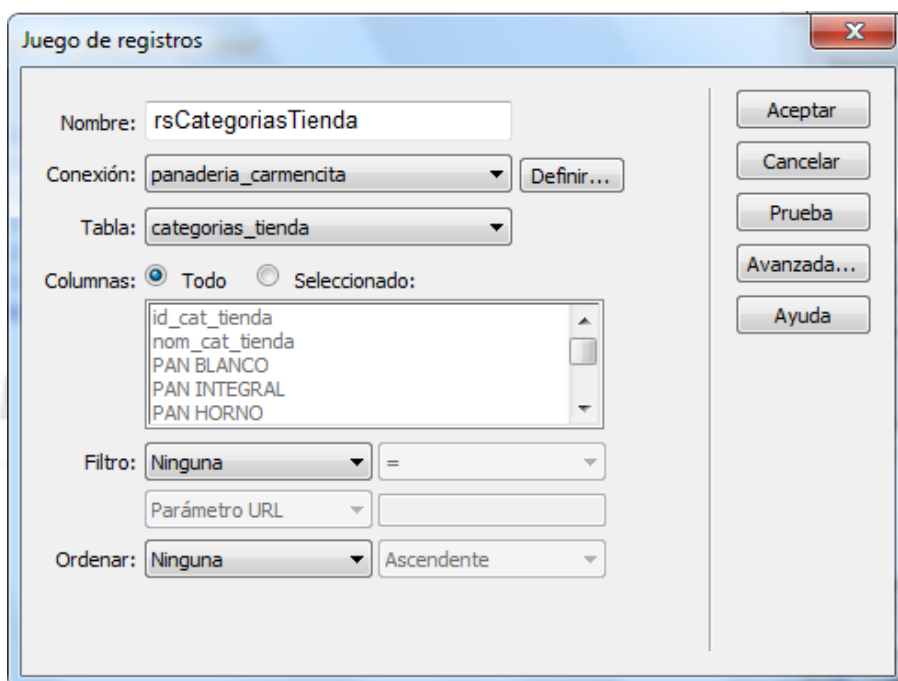
### Listado de categorías

Para este módulo se creó primero un archivo llamado *MenuTienda.php* que genera el listado de todas las categorías de los productos.

Los nombres para dicha lista se obtienen de la base de datos, específicamente de la columna *nom\_cat\_tienda* de la tabla *categorias\_tienda* que se encuentra en la base de datos *panadería\_carmencita*.

En el mismo archivo *MenuTienda.php* fue necesario crear un juego de registros con el nombre *CategoriasTienda* para poder efectuar interacciones (consultas) con la base de datos de forma más práctica. Para poder generar este juego de registros se utilizó la herramienta “Juego de registros” a la cual se accede

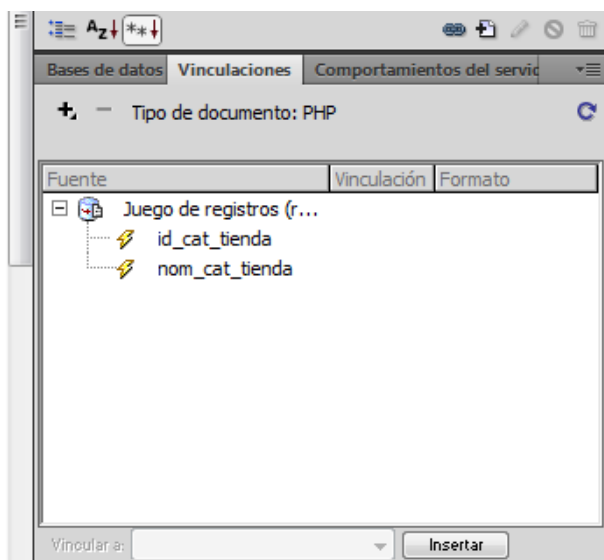
mediante Insertar>objeto de datos>Juego de registros. Luego de pulsar esta opción se muestra un cuadro de diálogo para introducir los parámetros de conexión. (Ver figura 4.8.3.2)



**Fuente:** Elaboración Propia

**Figura 4.8.3.2.** Parámetros del juego de registros.

Después de aceptar se agregó el juego de registros en el panel vinculaciones como se observa en la figura, que permite utilizar los valores de los registros de la tabla. Esta herramienta ofrece la posibilidad de utilizar los datos de la tabla con solo arrastrarla desde este árbol gráfico los registros y soltarlos en el área de la página donde se desea que se muestre (ver figura 4.8.3.3).



Fuente: Elaboración Propia

**Figura 4.8.3.3. Árbol de registros**

Lo que realmente hace esta herramienta de interfaz gráfica para obtener los datos de la base de datos, es ejecutar la consulta sql: “SELECT \* FROM categorias\_tienda”, con la cual se muestran todas las columnas de la tabla *categorias\_tienda*. Mediante el código del listado 4.8.3.2.

```
mysql_select_db($database_panaderia_carmencita,
$panaderia_carmencita);
$query_rsCategoriasTienda = "SELECT * FROM categorias_tienda";
$rsCategoriasTienda = mysql_query($query_rsCategoriasTienda,
$panaderia_carmencita) or die(mysql_error());
$row_rsCategoriasTienda =
mysql_fetch_assoc($rsCategoriasTienda);
$totalRows_rsCategoriasTienda =
mysql_num_rows($rsCategoriasTienda);
```

**Listado 4.8.3.3. Código para la conexión a la BD**

La cual se genera en la sección <header> de la pagina *MenuTienda.php*. Así mismo generará en la sección <body> el código siguiente:

```
<?php echo $row_rsCategoriasTienda['nom_cat_tienda']; ?>
```

En el cual se observa que con el comando *echo* de php muestra en el navegador el primer valor contenido en el arreglo `$row_rsCategTienda` en el que fueron almacenados los nombres de las categorías.

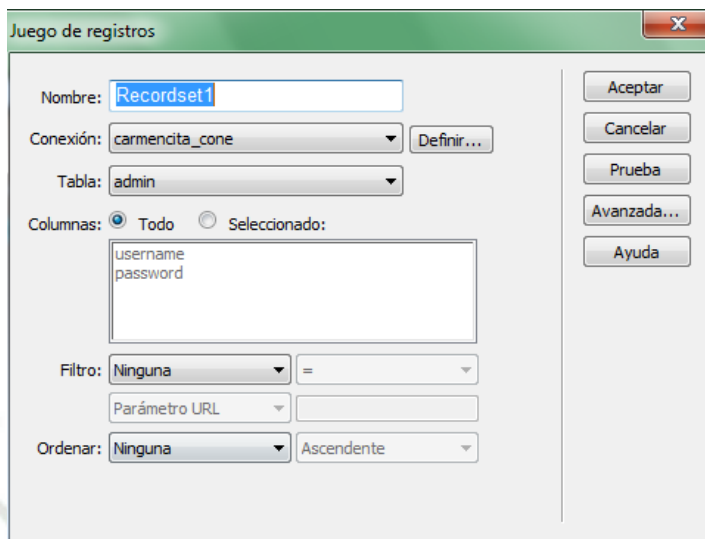
Con lo mencionado se pudo obtener el nombre de la primera categoría, por tanto fue necesario hacer un ciclo *do while* para obtener todos los valores de los registros. Antes de la ejecución del ciclo *do while* se añadieron las etiquetas `<div></div>` para indicar al navegador que no muestre los elementos en forma lineal sino que los muestre en forma vertical. Quedando el código como se muestra a continuación:

```
<?php do { ?>
<div><?php echo $row_rsCategoriasTienda['nom_cat_tienda'];
?></div>
<?php } while ($row_rsCategoriasTienda =
mysql_fetch_assoc($rsCategoriasTienda)); ?>
```

#### **Listado 4.8.3.4.** Código del ciclo *do while* para obtener todos los registros.

En el código del listado 4.8.3.4. se observa lo siguiente: se muestra en pantalla cada elemento del arreglo `$row_rsCategoriasTienda` mientras existan elementos que por medio del comando `mysql_fetch_assoc` devuelva la matriz asociativa que corresponde a la fila recuperada y traslada el apuntador de datos interno hacia adelante.

Para obtener la operación anterior de forma rápida se selecciono el registro, arrastrado y le aplicamos una región de repetición, desde el panel de Aplicación (Ver Figura 4.8.3.4).



**Juego de registros**

Nombre:

Conexión:  Definir...

Tabla:

Columnas: ☒ Todo ☐ Seleccionado:

Filtro:  =

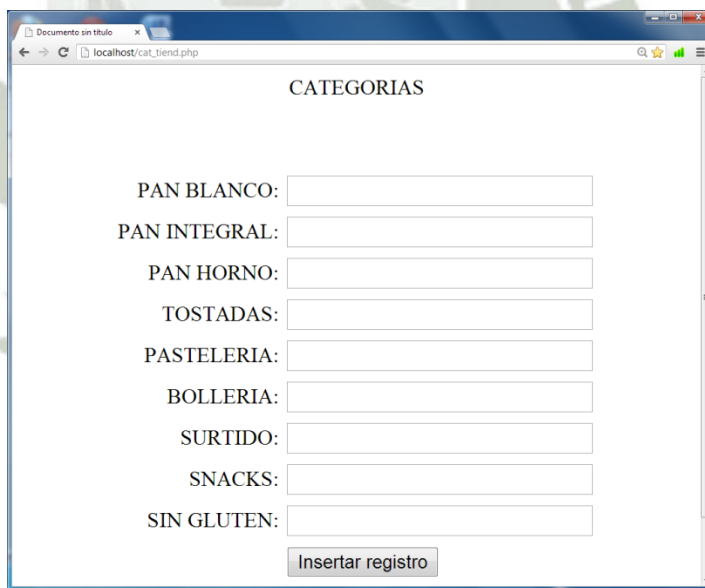
Ordenar:

Aceptar  
Cancelar  
Prueba  
Avanzada...  
Ayuda

Fuente: Elaboración Propia

**Figura 4.8.3.4.** Región de repetición

De esa forma fue que se obtuvo la lista completa de todas las categorías de los productos desde la base de datos, que en el navegador aparecerá como se muestra en la Figura 4.8.3.5.



Documento sin título

localhost/cat\_tienda.php

**CATEGORIAS**

PAN BLANCO:

PAN INTEGRAL:

PAN HORNO:

TOSTADAS:

PASTELERIA:

BOLLERIA:

SURTIDO:

SNACKS:

SIN GLUTEN:

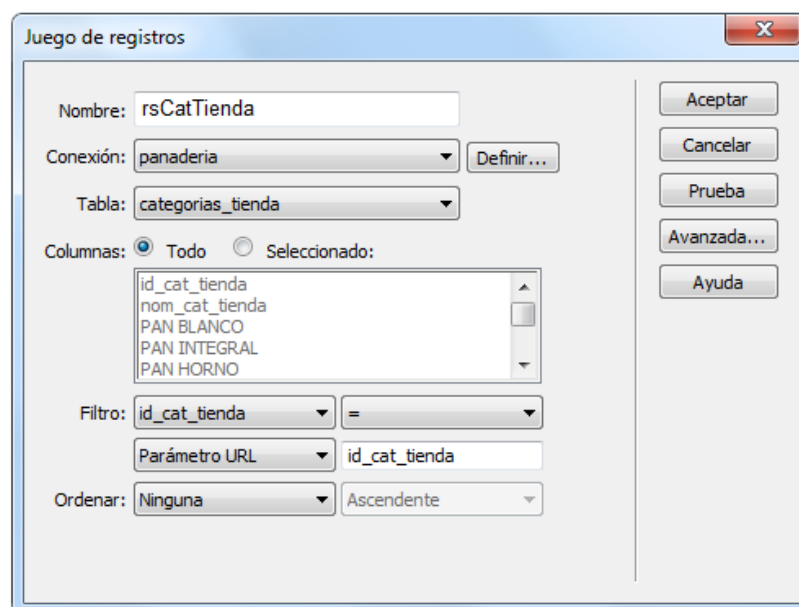
Insertar registro

Fuente: Elaboración Propia

**Figura 4.8.3.5.** Vista en el navegador de la lista de categorías generada.

### Generación del catálogo de productos

Para este módulo se creó un nuevo documento llamado *CatalogoTienda.php*. Antes de la generación del listado de los productos se obtuvo un título dinámico, que se muestra en la parte superior del área donde se despliegan los productos la categoría de los productos que se hayan seleccionado de la lista de categorías del módulo anterior, se recurre a la misma operación anterior (generando un juego de registros), pero esta vez aplicando un filtro para obtener el nombre de la categoría con base en su identificador único, en este caso la columna *id\_cat\_tienda* tendrá que ser igual al parámetro URL *id\_cat\_tienda*. Como se muestra en la Figura 4.8.3.6.



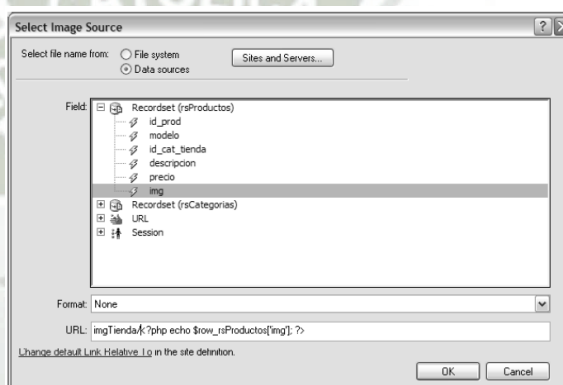
Fuente: Elaboración Propia

**Figura 4.8.3.6.** Configuración del Juego de registros con filtro

Para conseguir el desplegado de las imágenes de los productos de cada categoría se necesita nuevamente de un juego de registros ya que se requiere de los datos de la columna *img* de la tabla productos.

Este nuevo juego de registros de la misma manera fue filtrado por medio del parámetro *id\_cat\_tienda* que se ha seleccionado en el archivo *MenuTienda.php*.

La forma de listar los productos de cada categoría es similar al proceso del listado de categorías. Ahora no será necesario mostrar una lista de texto donde se muestren los modelos de los productos de cierta categoría, sino que, de acuerdo a las especificaciones establecidas previamente, fue necesario obtener las imágenes de los productos de la categoría elegida y posteriormente en un siguiente proceso se obtuvieron los detalles de los productos, por tanto de manera similar se insertó una imagen donde elegimos como origen el registro *img* del conjunto de registro previamente obtenidos, así como la ubicación física de la carpeta donde se obtienen las imágenes (ver Figura 4.8.3.7).



**Fuente: Elaboración Propia**

**Figura 4.8.3.7.** Selección del origen de la imagen  
Posteriormente

Luego de igual forma se le aplicó una región de repetición para obtener todos los productos de la categoría deseada.

Se genera el catálogo de los productos listados por categoría, a continuación se describe el proceso de la obtención de los detalles de cada producto.

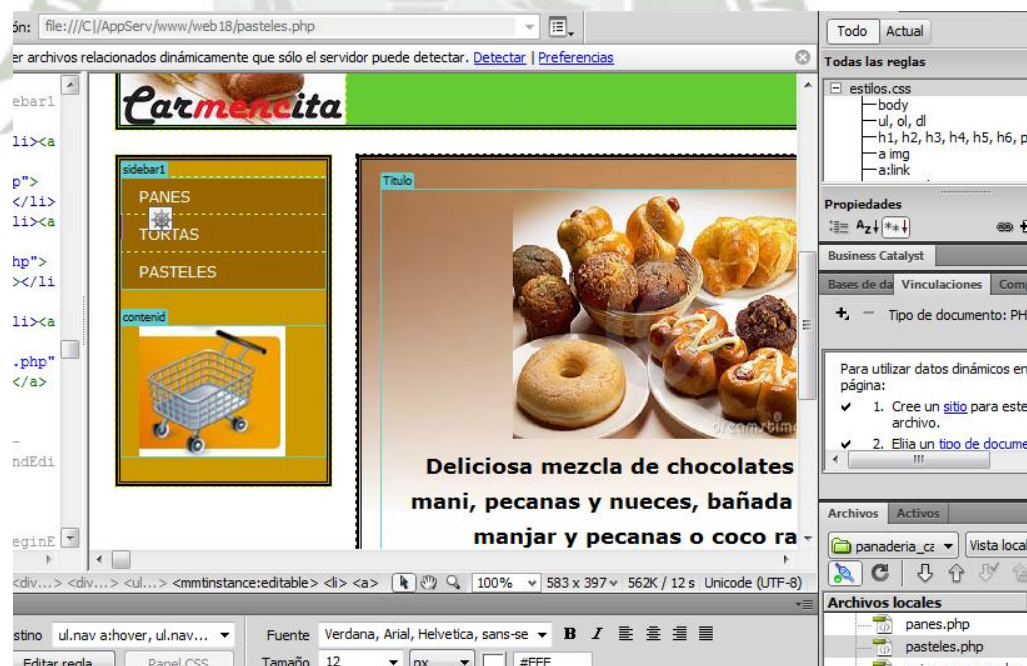
### Mostrar detalles de producto

De acuerdo a los requerimientos fue necesario contar con una interfaz intuitiva al usuario para la obtención de los detalles un producto. Se propuso la utilización de lo que se denomina “*tooltips*”, que es una descripción emergente, siendo una

herramienta de ayuda visual que funciona al situar el cursor sobre algún elemento gráfico, mostrando una ayuda adicional para informar al usuario de los detalles del producto como: el precio, la descripción y el modelo del elemento sobre el que se encuentra, proporcionando esta información adicional sin necesidad de que sea solicitada.

Al lograr obtener el valor de la *form variable* 'img' se hará una matriz asociativa con los valores del modelo, descripción y precio obtenidos de la consulta filtrada "SELECT \* FROM productos WHERE img='".\$img.'"', hecha a la base de datos. Con lo cual es suficiente para que funcione la transferencia de datos para el *tooltip*, lo que queda es darle presentación a la información, esto se logra con la definición de los parámetros de la clase "tooltip" en la hoja de estilos CSS a la que está ligada la página principal.

Así de esa manera se puso en funcionamiento el *tooltip* con los detalles del producto que será accionado al situar el ratón sobre alguna imagen del producto (Ver figura 4.8.3.8).



Fuente: Elaboración Propia

**Figura 4.8.3.8.** *Tooltip* en funcionamiento

### Carro de compra

Para el desarrollo de esta parte se empleó la funcionalidad **drag and drop** (arrastrar y soltar) para arrastrar aquellos artículos que se desean comprar a la “cesta de la compra”.

El usuario interactuante con la Web de forma novedosa y vistosa, puede ver en todo momento el resultado de sus acciones, pudiendo modificar o consultar sus compras sin a molestia de volver a cargar la página o de abandonarla. Lo cual tiene la ventaja de mantener actualizado el precio total de sus compras cuando añade o cuando elimina un producto.

Teniendo fundamento en el procedimiento descrito anteriormente para el tooltip, en ese mismo archivo scrip.js se agregaron otras funciones que permiten obtener las funcionalidades deseadas.

**Funcionalidad Drag.** Para hacer el efecto de *drag* o arrastre de los artículos se recurrió la función *draggable()* del framework, esta se aplicó con sus respectivos parámetros de configuración, al elemento *img* que se encuentra dentro de los *<div>* que contienen la clase *product*.

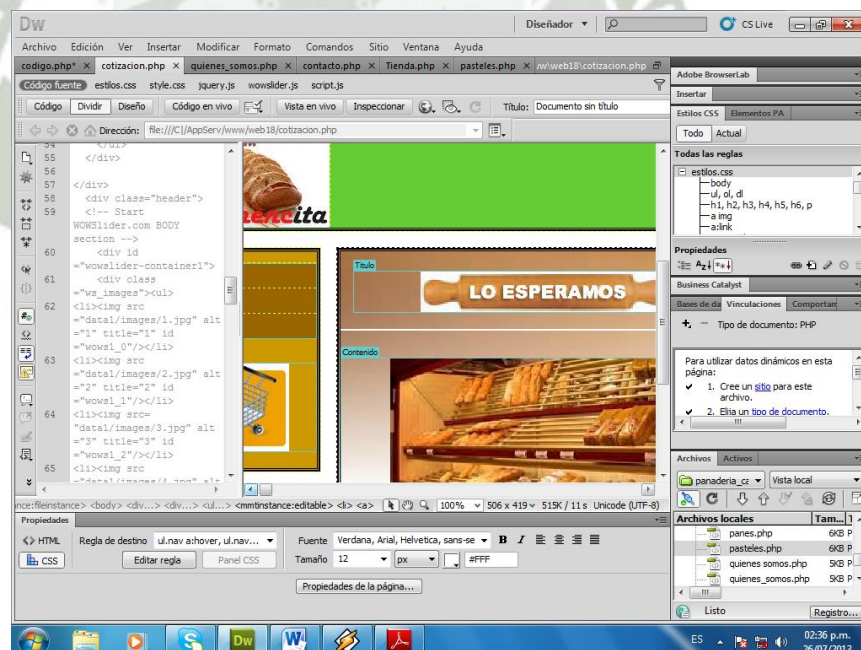
**Funcionalidad Drop (soltar)** La función asociada es *droppable()* la cual se aplicó al área donde deseamos sea “colocada” la imagen del producto. El área que se habrá de establecer para ser empleada dicha funcionalidad está establecida por las etiquetas *<div>* en las que se aplica la clase *content drop-here*, dentro de esa sección *div* se encuentra la imagen de un carro de compras para indicar al usuario donde se encuentra el área *droppable*. Esta imagen del carro de compras está en el archivo *cesta.php* que se configura posteriormente para ser mostrado mientras se despliega el catálogo de productos, de tal manera que el usuario pueda tenerlo disponible mientras hace las elecciones de sus productos.

Aquí está el mecanismo que hace que cuando se suelte la imagen sobre el elemento *droppable()* se active el ciclo *for* para que se incremente en uno la

cantidad de los artículos que se van comprando así mismo se irá actualizando el valor del precio total, mostrándolo en el elemento `#item.list` con la función `append()`. La función `remove()` y `change()` esta implementada para darle al usuario la opción de eliminar algún artículo del carro de compra o cambiar el número de productos que está adquiriendo, actualizando con la función `update_total()` el valor del precio total, el número y tipo de los productos dentro del carro.

Guando la variable `total price` contenga algún valor, se muestra el botón para ofrecer al usuario pasar a la caja. `$(a.button).css('display','block');` La función `addlist()` anteriormente mostrada devuelve el número y tipo de productos así como el costo total de la compra, estos valores son empleados por el archivo `addtocart.php` para mostrar al usuario dicha información. Aplicando posteriormente hojas de estilo CSS se podrá ver la cesta como se muestra en la figura 4.8.3.9.

**Figura 4.8.3.9.** Cesta de compra aplicando los estilos css



**Fuente:** Elaboración Propia

El archivo `order.php` lo que hace esencialmente es presentar los valores del código JavaScript.

#### 4.8.4. GALERIA DE IMAGENES

Uno de los objetivos que pretenden alcanzar es la promoción de los productos y lostipos de servicios que ofrece la PYME “Panaderia Carmencita”no fue suficiente con mostrar los productos de la tienda en línea, y no todos los productos pueden venderse por Internet, es por eso que existe una galería de imágenes en la que se muestra esta clase de productos y servicios que se ofrece. Al generar una galería de imágenes se recurre a un proceso análogo a la generación del catálogo para la tienda en línea, es decir, se cuenta con un menú de categorías cuyos valores se obtienen de la base de datos, para los productos de la galería y la galería propiamente donde se muestran los productos de la categoría elegida.

Las imágenes de los productos se guardaron en una carpeta distinta llamada *imgGaleria*. Se recurrió a la reutilización de código y la galería es capaz de generar las vistas en miniatura en el momento de ejecución y sin la necesidad de almacenar tres versiones de la misma imagen en el servidor debido a estas funcionalidades es que se reutilizo el código gratuito de una galería basada en jQuery que se descarga libremente de Internet.

Como primera instancia y después de descargarla se hizo referencia a las librerías que necesita la galería jQuery, dicha referencia esta dentro de la sección `<header>` del archivo *galeria.php*

La principal funcionalidad se encuentra en el archivo *galleria.js* Posteriormente se agrega un elemento `<div>` con un identificador único que es donde se muestra la galería de imágenes, dentro dichas etiquetas se declararon las imágenes, (mediante el proceso análogo a la Tienda en línea; se generarán de forma dinámica).

También se presenta en el panel derecho de la página principal un formulario para que este siempre visible mientras la galería se muestra, en donde el usuario

pueda obtener ayuda o asesoría adicional por parte del personal capacitado y así ponerse en contacto con la PYME “Panadería Carmencita”.

#### 4.8.5. DISEÑO Y COTIZACIÓN EN LÍNEA

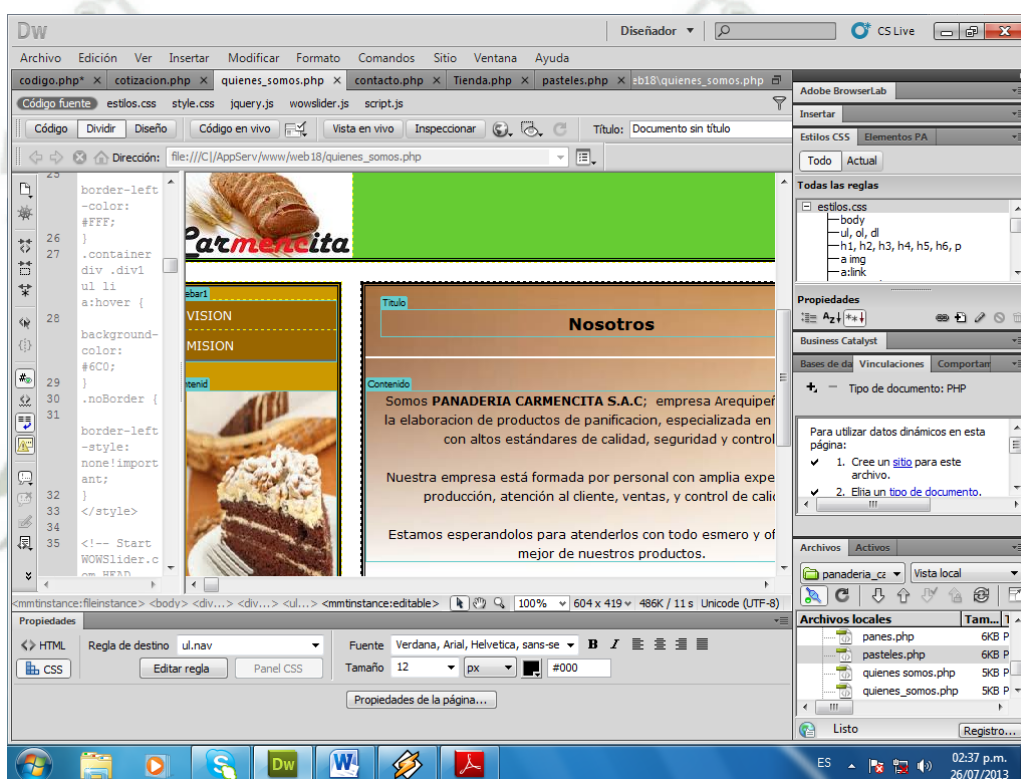
Las tendencias de negocio para el mercado actual que se visualiza en el futuro de las empresas, es la venta por Internet de los productos, sino además el ofrecimiento al cliente de poder personalizar sus propios productos. Donde tendrán la libertad de elegir sus propios diseños, medidas, etc.

Con referencia a la sección de requisitos que debe cubrir el sitio del presente trabajo, se requiere que si el cliente no encuentra lo que busca tenga la opción de hacer su propio diseño o redimensionamiento visual de algún producto para su posterior elaboración, indicando algunas características del producto adicionales para ser enviadas al departamento de diseño y posterior elaboración; previo pago. Además de esto se dispondrá de un mecanismo que ofrezca instantáneamente la cotización en línea al mismo tiempo que redimensiona el producto. Ese costo será proporcionado por medio de un factor, que se ha de establecer por la PYME con base en las características propias del producto. Para obtener dicho mecanismo se recurrirá a un proceso similar al anterior, es decir, se reutilizará código del Framework jQuery; ahora usando la función *resizable()* para poder redimensionar la imagen en el browser con solo arrastrar y soltar una esquina de la misma.

Al mismo tiempo se ejecuta en segundo plano otra función para conocer las dimensiones finales que el usuario está modificando y simultáneamente mostrando en pantalla tanto las medidas proporcionales como el costo proporcional al área que se obtiene, esto es posible gracias al tipo de requisitos que se solicitan. Es así como a manera de versión de prueba para la PYME se implementado dicha funcionalidad. Posteriormente se podría implementar que el usuario pueda elegir entre un mayor número de diseños.

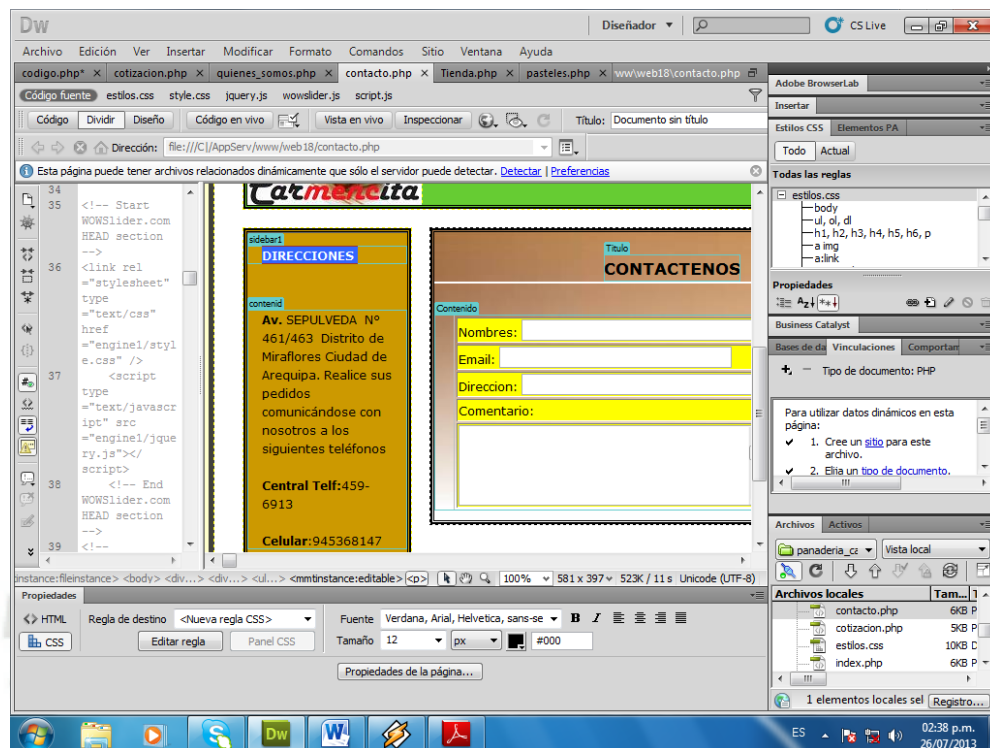
#### 4.8.6. INFORMACIÓN DE LA PYME

Para las secciones restantes que son: quienes somos, productos y contacto, solo se muestra información de presentación de la empresa que se mostrará en el Panel Central. Cada sección se diseñará en un archivo individual. Se agregan imágenes, texto y vínculos a los módulos para que el usuario pueda contactar o conocer más de la empresa. Posteriormente se aplicarán hojas de estilo CSS. (Ver figura 4.8.6.1.).



Fuente: Elaboración Propia

Figura 4.8.6.1. Aplicación de los estilos css



Fuente: Elaboración Propia

**Figura 4.8.6.2.** Secciones informativas de la PYME

### Vínculos entre las distintas paginas

Tras ejecutar el evento *click* con el Mouse sobre el vínculo se ejecutará la carga de un archivo php dentro de un elemento `<div>` que posee un identificador único. Este script se adaptará en cada parte de la página donde se necesite una vinculación para cargar peticiones en segundo plano.

Una característica de esta funcionalidad es que al ejecutarse las peticiones al servidor en segundo plano, el usuario no tendrá conocimiento de esta actividad a menos que se le indique de forma visual, por lo que se agregará una funcionalidad para mostrar al usuario cuando se está llevando a cabo su petición o comunicación con el servidor y cuando esta ha terminado

De donde se observa que el uso de las propiedades CSS para mostrar el mensaje dentro de un elemento `<span>` que podrá ubicarse en cualquier lugar de la página principal, y ocultarlo cuando finalice la carga del archivo. Después de lo anterior será necesario establecer las reglas de la disposición de los elementos en la página principal, es decir, la manera que se mostrarán los archivos de los distintos módulos en los paneles de la página principal. Por ejemplo, en el caso que se desee mostrar el carrito de compras, éste solo se mostrará cuando en el panel central se visualice el catálogo de los productos, o se mostrarán las promociones especiales cuando se visualice la página principal, etc.

Los vínculos a los principales módulos se harán desde un menú horizontal que se cargará en el encabezado de la página principal.

#### 4.8.7. INTERFAZ DE LA INFORMACIÓN

Una modificación habitual que puede realizarse en la tienda en línea es cambiar el precio de un producto (por ejemplo para ofertas especiales o rebajas), y agregar artículos a la galería de imágenes por lo que se diseñará una interfaz de administración para los usuarios administradores del sitio Web, que no requieran de conocimientos de programación para poder dirigirlo que permitirá añadir, eliminar y modificar productos y categorías almacenados en la base de datos, que son los módulos básicos, ya que se pueden administrar gran cantidad de aspectos del sitio, que estarían fuera del alcance del presente trabajo.

La interfaz de administrador será una versión ligeramente distinta de la página que es visible para cualquier tipo de usuario, solamente que se agregarán las funcionalidades propias del administrador y serán visibles solo después de haberse registrado previamente con las credenciales autorizadas.

##### **Inicio de sesión**

Todas las cuentas de usuario se almacenan en la tabla **admin** que contendrá la información básica como el nombre de usuario y password.

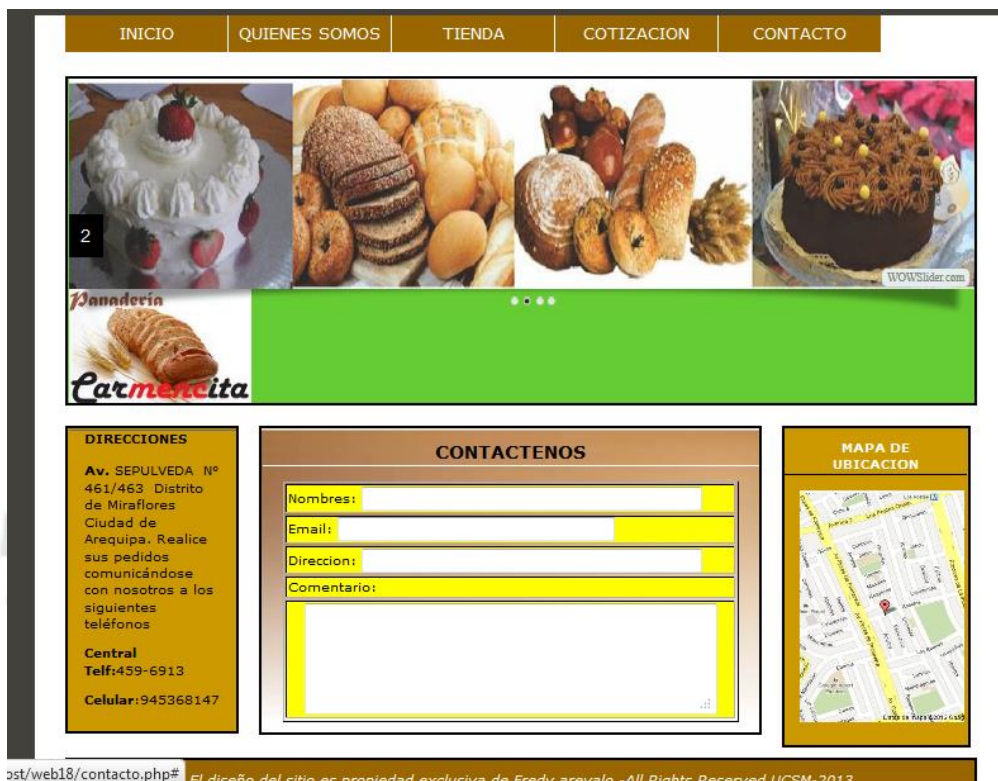
**Proceso del inicio de sesión:**

La secuencia de comandos verifica si la combinación de nombre de usuario y contraseña exista en la base de datos. En caso que la combinación exista en la base de datos, podrá ir a la página principal de administración. Si no es así se mostrará un error.

Para implementar la interfaz es necesario que la tabla *admin* de la base de datos no esté vacía, de lo contrario se agregarán valores para realizar las pruebas. También ha de tomarse en cuenta la manera en que se guardarán las contraseñas, por motivos de seguridad deben encriptarse con algoritmos como MD5, del cual puede configurarse con las herramientas proporcionadas por la extensión de Dreamweaver. Por tanto se agregarán los módulos para dichas tareas sobre la misma interfaz de usuario que se ha desarrollado, posteriormente se ocultarán mediante la configuración de variables de sesión para que dichos módulos puedan ser visibles exclusivamente los usuarios que se hayan autenticado con privilegios de administrador. Es así como se agregará un módulo para agregar una nueva categoría, justo debajo del desplegado de las categorías existentes para la tienda.

Finalmente se harán las configuraciones de registro desde el panel de control de la extensión instalada en Dreamweaver donde se definirán las opciones y acciones para el sitio como son las tablas de donde se obtendrán los usuarios

**Figura 4.8.7.1.** Configuración del inicio de sesión de un usuario administrador



**Fuente:** Elaboración Propia

Después de haber implementado todas las funcionalidades solicitadas y hacer pruebas de funcionamiento se aplica la estética de presentación mediante las hojas de estilos a la versión final. Estas hojas de estilos CSS se guardarán en documentos aparte pero dentro de la misma carpeta raíz. Al visualizar la versión final de la página en el navegador queda como se muestra en la figura.

**Figura 4.8.7.2.** Versión final del proyecto visto en el navegador



Fuente: Elaboración Propia

## 4.9. PRUEBAS E INTEGRACIÓN

Aspectos como la comprobación y la revisión del código son básicos en la ingeniería de Software y no se suele pasar por alto en programación Web. Es muy habitual tratar de ejecutar el sistema con dos o tres pruebas y será necesario comprobar que se ha revisado y probado exhaustivamente distintos casos antes de preparar una producción final del proyecto. Existen dos enfoques para reducir los errores del código:

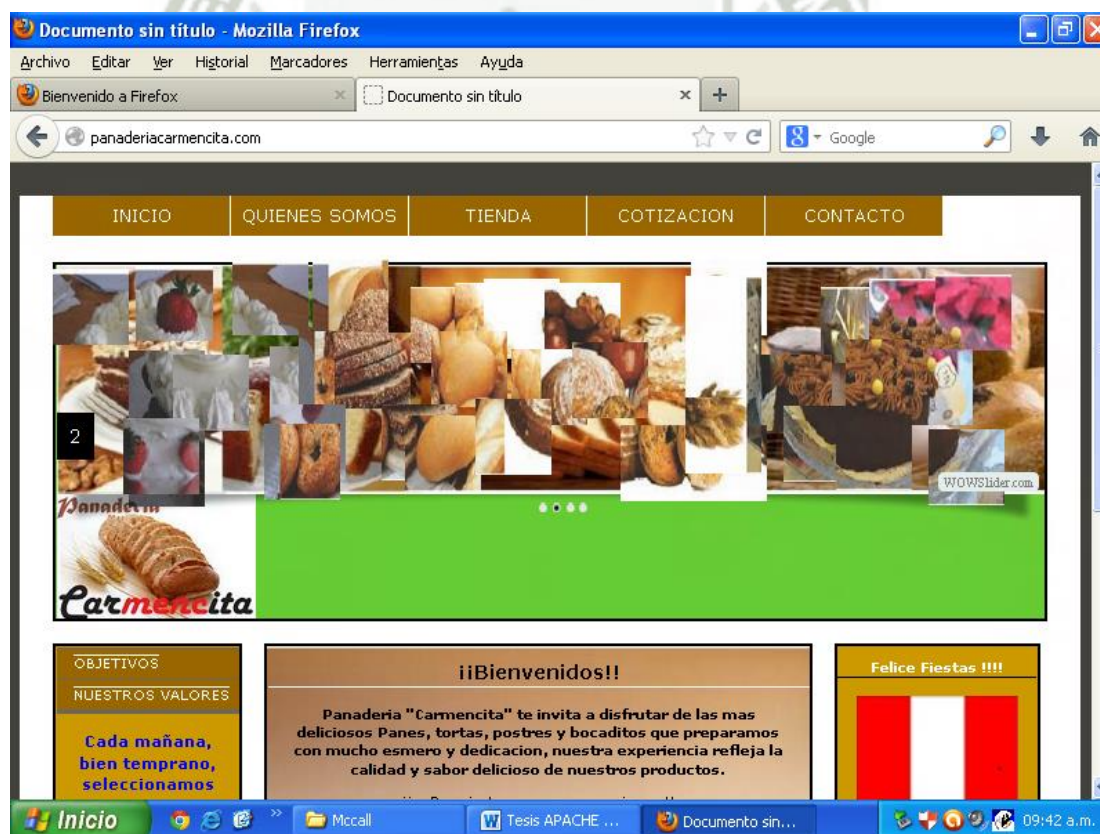
Adoptar una práctica de revisión de código. Gracias a esto se pueden encontrar: errores que se han pasado por alto, comprobaciones sobre las que no se ha pensado, optimización, mejoras de seguridad, componentes existentes que se pueden utilizar con el objetivo de mejorar un fragmento de código existente, funcionalidad adicional. Localizar personas que prueben las aplicaciones Web y que representen a los usuarios finales del producto.

El segundo enfoque es sin duda el más importante ya que la principal diferencia entre aplicaciones Web y aplicaciones de escritorio es que cualquiera debe poder utilizar una aplicación Web. Teniendo esto en mente no se debe asumir que los usuarios estarán familiarizados con la informática. Las aplicaciones Web se deben documentar y explicar por sí mismas. Se debe pensar en cómo utilizarán los usuarios la aplicación.

Para el presente trabajo se han hecho algunas pruebas tanto de navegación como de desempeño en diferentes navegadores también se han elaborado otras métricas para verificar que el tráfico de datos sea eficiente. Para esta primera parte de la prueba se verificó que el sitio Web pueda mostrarse de la misma forma en la mayoría de los navegadores existentes.

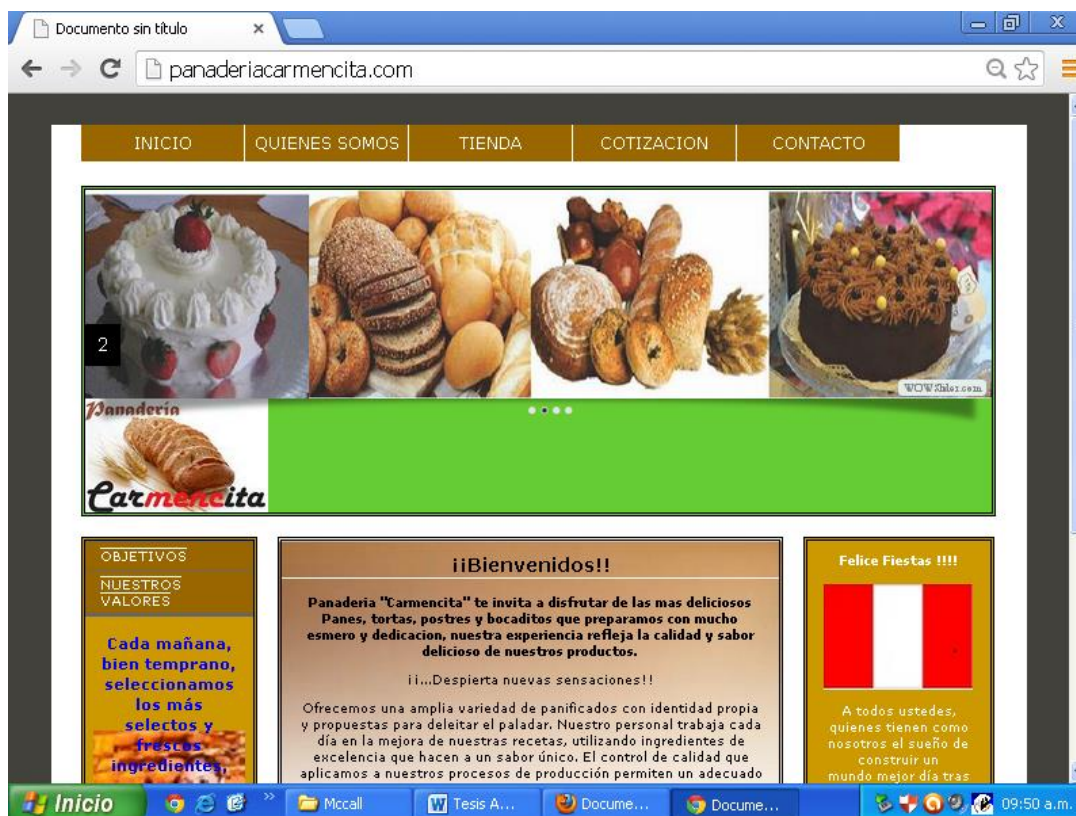
Los navegadores donde fue probado el sitio fueron: FIREFOX, Internet Explorer 8, Opera 10.62, Google Chrome 9.

**Figura 4.9.1.** Sitio Web visto en Firefox 3.6.8



Fuente: Elaboración Propia

Figura 4.9.2. Sitio Web mostrado en google chrome



Fuente: Elaboración Propia

## CAPITULO V

### EVALUACIÓN DE FACTORES DE CALIDAD

Para evaluar la calidad del software, primero deben establecerse los requerimientos de calidad, después especificar, diseñar y efectuar la evaluación. El proceso completo se muestra a continuación donde se indican las actividades de cada etapa. Es necesario mencionar que este proceso de evaluación fue la referencia y el camino a seguir en el presente trabajo, ya que el objetivo era obtener el nivel de calidad de software para la página web Panadería Carmencita.

#### 5.2. ESTABLECIMIENTO DE LOS REQUERIMIENTOS DE EVALUACIÓN.

Fue necesario determinar los requerimientos de evaluación suficientes para definir el propósito de la evaluación del software.

##### 5.1.1. Establecimiento del propósito de la evaluación.

El objetivo principal de la evaluación del software construido, fue medir la calidad externa del producto en este caso la página web de la panadería Carmencita.

El propósito de la evaluación de la calidad de la página de E-commerce fue con el fin de:

- Comparar algunos de los productos de similares disponibles en el mercado.
- Conocer el porcentaje de calidad externa de nuestro producto de software.

Después de analizar las cualidades específicas, se compararon con las características y subcaracterísticas de calidad definidas por la Norma ISO-IEC 9126 y con las definidas en la metodología de aseguramiento de calidad McCall, y posteriormente, se establecieron las características del modelo de calidad aplicado al sitio web de la Panadería Carmencita.

##### 5.1.2. Identificación del producto a evaluar.

El producto evaluado en el caso de estudio, denominado “sitio web Panadería Carmencita”, es una aplicación Web que ofrece servicios típicos del negocio de la panificación, permitiendo a los clientes y público en general llevar adelante el proceso de compra, consulta, cotización, contacto reservas etc. empleando recursos y servicios provistos típicamente por la Web.

### 5.1.3. Especificación del Modelo de Calidad

Para definir los requerimientos de evaluación es importante especificar el modelo de calidad apropiado; para esto fue necesario tomar en cuenta las cualidades del software que se desea medir. Después de analizar las cualidades específicas, estas se compararon con las características y subcaracterísticas de calidad definidas por la Norma ISO-IEC 9126 y con las definidas en la metodología de aseguramiento de calidad McCall, y posteriormente, se establecieron las características del modelo de calidad aplicado a la página web implementada. A continuación se muestra la relación de algunas cualidades que se evaluaron con las características, subcaracterísticas y atributos de calidad que corresponde.

| CUALIDAD A<br>EVALUAR | CARACTERISTICAS | SUBCARACTERISTICAS | ATRIBUTOS |
|-----------------------|-----------------|--------------------|-----------|
|-----------------------|-----------------|--------------------|-----------|

|                                                  |               |              |                             |
|--------------------------------------------------|---------------|--------------|-----------------------------|
| Confiabilidad en los resultados de la Pagina web | Confiabilidad | Exactitud    | Exactitud de respuesta      |
| Como es el acceso al producto                    | Usabilidad    | Comprensión  | Interfaz de usuario         |
| Tiempo de respuesta en realizar los procesos     | Usabilidad    | Operabilidad | Tiempo de operación         |
| Grado de visualización de los resultados         | Usabilidad    | compresión   | Visualización de resultados |
| Usa terminología adecuada                        | Usabilidad    | compresión   | Terminología adecuada       |
| Tiene tutorial para la operatividad del sistema  | Usabilidad    | aprendizaje  | Tutorial                    |

Fuente COLOMES FORMOS, XAVIER. (2009).

**Tabla 5.1.** Relación entre cualidades a medir y características

Hemos encontrado durante la comparación de estas cualidades con las métricas propuestas por la norma, que dentro de las cualidades a medir faltan aspectos que no se consideran, por lo cual se añadió nuevas características necesarias de acuerdo a nuestro sitio.

En la Figura 5.1, se presentan las características y subcaracterísticas consideradas de la norma ISO 9126 y la Metodología de Calidad McCall para posteriormente considerarlo como modelo de calidad para del sitio web.

Cabe mencionar que se seleccionaron estas características, por considerarlas necesarias para evaluar la página web de la panadería Carmencita, con el fin de reducir en lo posible el número de métricas a evaluar.

**Tabla. 5.2.** Relación de características y subcaracterísticas que conforman el modelo de calidad. Parte 1.

| PUNTOS DE | FACTORES | CRITERIOS/METRICAS |
|-----------|----------|--------------------|
|-----------|----------|--------------------|

| VISTA O EJES           |                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|------------------------|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OPERACIÓN DEL PRODUCTO | Facilidad de uso | <ol style="list-style-type: none"> <li><b>Facilidad de operación-</b> Facilidad de comunicación: Atributos del software que proporcionan entradas y salidas fácilmente asimilables.</li> <li><b>Facilidad de aprendizaje:</b> Atributos del software que facilitan la familiarización inicial del usuario con el software y la transición del modo actual de operación.</li> <li><b>Formación:</b> El grado en que el software ayuda para permitir que nuevos usuarios apliquen el sistema.</li> </ol>                                                                                                                                                                                          |
|                        | Integridad       | <ol style="list-style-type: none"> <li><b>Control de accesos.</b> Atributos del software que proporcionan control de acceso al software y los datos que maneja.</li> <li><b>Facilidad de auditoría:</b> Atributos del software que facilitan la auditoría de los accesos al software.</li> <li><b>Seguridad:</b> La disponibilidad de mecanismos que controlen o protejan los programas o los datos.</li> </ol>                                                                                                                                                                                                                                                                                 |
|                        | Corrección       | <ol style="list-style-type: none"> <li><b>Compleitud:</b> Atributos del software que proporcionan la implementación completa de todas las funciones requeridas.</li> <li><b>Consistencia:</b> Atributos del software que proporcionan uniformidad en las técnicas y notaciones de diseño e implementación.</li> <li><b>Trazabilidad o rastreabilidad:</b> Atributos del software que proporcionan una traza desde los requisitos a la implementación con respecto a un entorno operativo concreto.</li> </ol>                                                                                                                                                                                   |
|                        | Fiabilidad       | <ol style="list-style-type: none"> <li><b>Precisión:</b> Atributos del software que proporcionan el grado de precisión requerido en los cálculos y los resultados.</li> <li><b>Consistencia.</b></li> <li><b>Tolerancia a fallos:</b> Atributos del software que posibilitan la continuidad del funcionamiento bajo condiciones no usuales.</li> <li><b>Modularidad:</b> Atributos del software que proporcionan una estructura de módulos altamente independientes.</li> <li><b>Simplicidad:</b> Atributos del software que posibilitan la implementación de funciones de la forma más comprensible posible.</li> <li><b>Exactitud:</b> La precisión de los cálculos y del control.</li> </ol> |
|                        | Eficiencia       | <ol style="list-style-type: none"> <li><b>Eficiencia</b> en ejecución: Atributos del software que minimizan el tiempo de procesamiento.</li> <li><b>Eficiencia</b> en almacenamiento: Atributos del software que minimizan el espacio de almacenamiento necesario.</li> </ol>                                                                                                                                                                                                                                                                                                                                                                                                                   |

| PUNTOS DE | FACTORES | CRITERIOS/METRICAS |
|-----------|----------|--------------------|
|-----------|----------|--------------------|

| VISTA O<br>EJES       |                            |                                                                                                                                                                                                                                                                                                                                                             |
|-----------------------|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| REVISIÓN DEL PRODUCTO | Facilidad de mantenimiento | <p>20. Modularidad.</p> <p>21. Simplicidad.</p> <p>22. Consistencia.</p> <p>23. <b>Concisión:</b> Atributos del software que posibilitan la implementación de una función con la menor cantidad de códigos posible.</p> <p>24. <b>Auto descripción:</b> Atributos del software que proporcionan explicaciones sobre la implementación de las funciones.</p> |
|                       | Facilidad de prueba        | <p>25. Modularidad.</p> <p>26. Simplicidad.</p> <p>27. Auto descripción.</p> <p>28. <b>Instrumentación:</b> Atributos del software que posibilitan la observación del comportamiento del software durante su ejecución para facilitar las mediciones del uso o la identificación de errores.</p>                                                            |
|                       | Flexibilidad               | <p>29. Auto descripción.</p> <p>30. <b>Capacidad de expansión:</b> Atributos del software que posibilitan la expansión del software en cuanto a capacidades funcionales y datos.</p> <p>31. <b>Generalidad:</b> Atributos del software que proporcionan amplitud a las funciones implementadas.</p> <p>32. Modularidad.</p>                                 |

| PUNTOS DE VISTA O EJES  | FACTOR            | CRITERIOS/METRICAS                                                                                                                                                                                                                                                                                                                                                                                                          |
|-------------------------|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TRANSICION DEL PRODUCTO | Reusabilidad      | 33. Auto descripción.<br>34. Generalidad.<br>35. Modularidad<br>36. <b>Independencia</b> entre sistema y software: Atributos del software que determinan su dependencia del entorno operativo.<br>37. <b>Independencia del hardware:</b> Atributos del software que determinan su dependencia del hardware.                                                                                                                 |
|                         | Interoperabilidad | 38. Modularidad.<br>39. <b>Compatibilidad de comunicaciones:</b> Atributos del software que posibilitan el uso de protocolos de comunicación e interfaces estándar.<br>40. <b>Compatibilidad de datos:</b> Atributos del software que posibilitan el uso representaciones de datos estándar.<br>41. <b>Estandarización en los datos:</b> El uso de estructuras de datos y de tipos estándar a lo largo de todo el programa. |
|                         | Portabilidad      | 42. Auto descripción.<br>43. Modularidad.<br>44. Independencia entre sistema y software.<br>45. Independencia del hardware.                                                                                                                                                                                                                                                                                                 |

Fuente: COLOMES FORMOS, XAVIER. (2009).

## 5.2. ESPECIFICACIÓN DE LA EVALUACIÓN.

El propósito de la especificación de la evaluación del software fue determinar el alcance de la medición. En este caso las características y subcaracterísticas establecidas en el modelo de calidad propuesto en la sección anterior, determinan el punto de partida para la selección de los atributos y/o métricas propuestos para la evaluación.

### 5.2.1. Métricas para la evaluación.

Para llevar a cabo el proceso de evaluación de nuestro software fue necesario evaluar una serie de métricas así como contabilizar y obtener un promedio y el porcentaje de calidad del software. En el modelo McCall se muestran 43 métricas para la etapa de aceptación del software.

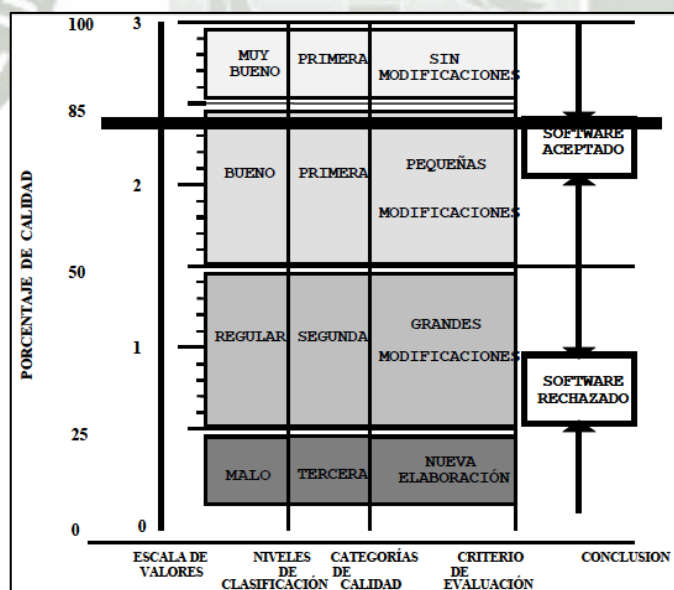
Los criterios para seleccionar las métricas necesarias son:

1. Reducir al máximo el número de métricas a evaluar por facilidad al usuario.
2. Incluir un mayor número de métricas críticas cuyas respuestas sean de forma afirmativa y negativa, para evitar una calificación relativa al criterio del usuario que está evaluando el software.
3. Incluir métricas exclusivas de la parte de operación del software, y algunas relacionadas al mantenimiento del software.

Finalmente, las métricas consideradas en el presente trabajo fueron 43, las mínimas necesarias para medir la calidad en software de nuestro sitio web.

### 5.2.2. Establecimiento de criterios para la valoración.

La Figura 5.1. Muestra la relación entre la escala de calificación y el valor del porcentaje, en cuanto a calidad se refiere. Una vez realizada la medición de las métricas en la etapa de ejecución de la evaluación, se obtuvieron los criterios de calidad como son el nivel de clasificación, la categoría de calidad y la conclusión.



Fuente: COLOMES FORMOS, XAVIER. (2009).

**Figura 5.1. Esquema de criterios de evaluación para la página web Panadería Carmencita**

### 5.3. DISEÑO Y PROCEDIMIENTO DEL PLAN DE EVALUACIÓN.

Dentro del plan para medirlos factores de calidad de la página web Panadería Carmencita, se estableció el siguiente procedimiento:

- a) El cuestionario de métricas evaluado, está clasificado en 11 características principales, que son Facilidad de uso, integridad, corrección, Fiabilidad, Eficiencia, Facilidad de mantenimiento, facilidad de prueba, flexibilidad, reusabilidad, interoperabilidad y portabilidad.
- b) La evaluación fue por cada una de las características establecidas.
- c) El cuestionario contiene los tipos de métricas, con preguntas directas o críticas por lo que el evaluador contesta SI o NO.
- d) Una vez que se midieron todas las métricas, se calcularon subtotaes de cada una de las características en forma de promedio, es decir la suma de todas las métricas de dicha característica y se promedió entre el número de métricas.
- e) Finalmente se sumaron los valores promedios de las características y se obtuvo el promedio de éstas, el resultante fue el número de valoración.
- f) Con este número de valoración se calculó el porcentaje de calidad considerando la escala propuesta en la Figura 5.1, donde el valor máximo 3 equivale al 100%, entonces para calcular el porcentaje bastó con dividir este valor entre 3 y después multiplicarlo por 100.
- g) Con este número de valoración y el porcentaje de calidad externa del software, lo que se hizo fue comparar éste en base a los criterios establecidos en la Figura 5.1 y así definimos los niveles de clasificación, categorías de calidad, criterios de evaluación y conclusión.

- h) Como resultado se realizó un reporte que contiene el valor para cada una de las características medidas al software y el porcentaje de cumplimiento de calidad externa del software, así como su interpretación.

## 5.4. EJECUCIÓN DE LA EVALUACIÓN

### 5.4.1. Ejecución del instrumento.

Una vez realizado el plan de evaluación de la página web, y según lo establecido en la especificación, donde se muestran las métricas mínimas necesarias, se procedió a realizar la medición de la calidad usando el instrumento de aplicación. A continuación se muestra el proceso de medición de la calidad realizada para la página web para la Panadería “Carmencita”.

### INSTRUMENTO DE EVALUACIÓN SEGÚN LAS MÉTRICAS DE McCall

## 1. OPERACIÓN DEL PRODUCTO

| FACTOR              | MÉTRICA                                                                                                          |   |   |
|---------------------|------------------------------------------------------------------------------------------------------------------|---|---|
| FACILIDAD DE<br>USO | 1. ¿La planeación y diseño permite atender las necesidades de información y servicios del usuario?               |   | X |
|                     | 2. ¿Los nuevos usuarios tienen una interacción efectiva y familiar en base a conocimientos previos con el sitio? |   | X |
|                     | 3. ¿El entorno ayuda para permitir que nuevos usuarios apliquen el sistema?                                      | X |   |
| INTEGRIDAD          | 4. ¿La página web posee atributos de control de accesos?                                                         | X |   |
|                     | 5. ¿El sistema posee atributos que faciliten la auditoria de los accesos al software?                            |   | X |
|                     | 6. ¿Hay disponibilidad de mecanismos que controlen o protejan los datos del sitio?                               | X |   |
| CORRECCION          | 7. ¿Tiene atributos que proporcionan la implementación completa de todas las funciones requeridas?               | X |   |
|                     | 8. ¿Posee cualidades que proporcionan uniformidad en las técnicas y notaciones de diseño e implementación?       |   | X |
|                     | 9. ¿Proporciona una traza desde los requisitos a la implementación con respecto a un entorno operativo concreto? |   | X |
| FIABILIDAD          | 10. ¿Hay precisión en los resultados obtenidos?                                                                  |   | X |
|                     | 11. ¿Los resultados son consistentes con los requerimientos?                                                     |   | X |
|                     | 12. ¿Existen fallas cuando la página web esta en uso?                                                            | X |   |
|                     | 13. ¿Tiene atributos que proporcionan una estructura de módulos altamente independientes.?                       | X |   |
|                     | 14. ¿Puede el usuario restablecer fácilmente la página cuando está en uso?                                       |   | X |
|                     | 15. ¿Hay exactitud en los resultados obtenidos?                                                                  |   | X |
| EFICIENCIA          | 16. ¿Las respuestas del sistema son rápidas?                                                                     |   | X |
|                     | 17. ¿Puede el usuario guardar su información generada con el software, en la memoria del sistema?                | X |   |

Tabla. 5.3 Medición de las Características de Operación de la Pagina web

## 2. REVISIÓN DEL PRODUCTO

| FACTOR                        | MÉTRICA                                                                                                                                  |   |   |
|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|---|---|
| FACILIDAD DE<br>MANTENIMIENTO | 18. ¿El Sitio está dividido en componentes funcionales tiene acoplamiento y cohesión?                                                    |   | X |
|                               | 19. ¿El sitio está orientado hacia la simplicidad y facilidad operativa, en su manejo y mantenimiento?                                   |   | X |
|                               | 20. ¿Las aplicaciones son consistentes con conocimientos previos?                                                                        |   | X |
|                               | 21. ¿El software posibilita la implementación de una función con la menor cantidad de código posible?                                    | X |   |
|                               | 22. ¿Posee atributos que proporcionan explicaciones sobre la implementación de las funciones?                                            |   | X |
| FACILIDAD DE<br>PRUEBA        | 23. ¿Provee una plataforma modular de prueba para solucionar problemas rápidamente?                                                      | X |   |
|                               | 24. ¿Resulta simple que el software demuestre sus fallas a través de la ejecución de pruebas?                                            |   | X |
|                               | 25. ¿Posee atributos autodescriptivos que permiten realizar pruebas sobre el sitio?                                                      | X |   |
|                               | 26. ¿Es posible observar el comportamiento del software durante su ejecución y facilitar mediciones del uso e identificación de errores? |   | X |
| FLEXIBILIDAD                  | 27. ¿Existe variedad de posibilidades con las que el usuario y el sitio puedan intercambiar información?                                 | X |   |
|                               | 28. ¿Hay la posibilidad de la expansión del software en cuanto a capacidades funcionales y datos?                                        |   | X |
|                               | 29. ¿Permite el software proporcionar amplitud a las funciones implementadas?                                                            |   | X |
|                               | 30. ¿Tiene una serie de módulos que le dan potencia y flexibilidad?                                                                      | X |   |

Figura. 5.4. Medición de las Características de Revisión.

### 3. TRANSICIÓN DEL PRODUCTO

| FACTOR            | MÉTRICA                                                                                                                           |   |   |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------|---|---|
| REUSABILIDAD      | 31. ¿La página tiene varias opciones para ingresar los datos: Archivos tipo texto, archivos provenientes de otras bases de datos? |   | X |
|                   | 32. ¿El usuario entiende fácilmente los mensajes que envía el software?                                                           |   | X |
|                   | 33. ¿La página contiene módulos para visualizar gráficamente resultados?                                                          |   | X |
|                   | 34. ¿El sistema es independiente del Software?                                                                                    | X |   |
|                   | 35. ¿Los atributos del Software son independientes del hardware?                                                                  | X |   |
| INTEROPERABILIDAD | 36. ¿El sistema dispone de mecanismos que permitan intercambiar procesos y/o datos?                                               |   | X |
|                   | 37. ¿Los Atributos posibilitan el uso de protocolos de comunicación e interfaces estándar?                                        |   | X |
|                   | 38. ¿Los Atributos permiten el uso representaciones de datos estándar?                                                            |   | X |
|                   | 39. ¿Usa estructuras de datos y de tipos estándar a lo largo de todo el programa?                                                 | X |   |
| PORTABILIDAD      | 40. ¿El sistema se puede adaptar a cualquier manejador de basa de datos?                                                          |   | X |
|                   | 41. ¿Al realizar mejoras del sistema, las funciones y módulos son similares?                                                      |   | X |
|                   | 42. ¿La construir la página el sistema operativo es un factor primordial?                                                         | X |   |
|                   | 43. ¿Al desarrollar el sistema el hardware es un factor primordial?                                                               |   | X |

**Tabla 5.5. Medición de las Características de Transición de la Página web.**

#### 5.4.2. Resultados de la evaluación

Una vez contestadas las métricas, se obtuvo el reporte final donde se muestra el valor obtenido por cada una de las características y una calificación total; estos valores están dentro de un rango de 0 a 3, con ello se obtiene el porcentaje de calidad de la siguiente forma:

#### 1. Operación del producto

Se realizó el recuento aplicando la formula respectiva:

$$\text{Factor Facilidad de uso} = \frac{\sum N^{\circ}si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3+3}{3} \right) = 2$$

$$\text{Factor integridad} = \frac{\sum N^{\circ}si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3}{3} \right) = 1$$

$$\text{Factor Corrección} = \frac{\sum N^{\circ}si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3+3}{3} \right) = 2$$

$$\text{Factor Fiabilidad} = \frac{\sum N^{\circ}si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3+3+3+3}{6} \right) = 2$$

$$\text{Factor eficiencia} = \frac{\sum N^{\circ}si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3}{2} \right) = 1.5$$

## 2. Revisión del producto

$$\text{Factor Facilidad de Mantenimiento} = \frac{\sum N^{\circ} si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3 + 3 + 3 + 3}{5} \right) = 2.4$$

$$\text{Factor Facilidad de Prueba} = \frac{\sum N^{\circ} si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3 + 3}{4} \right) = 1.5$$

$$\text{Factor Flexibilidad} = \frac{\sum N^{\circ} si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3 + 3}{4} \right) = 1.5$$

### 3. Transición del producto

$$\text{Factor Reusabilidad} = \frac{\sum N^{\circ} si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3 + 3 + 3}{5} \right) = 1.8$$

$$\text{Factor Reusabilidad} = \frac{\sum N^{\circ} si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3 + 3 + 3}{4} \right) = 2.25$$

$$\text{Factor Portabilidad} = \frac{\sum N^{\circ} si}{\text{Cantidad de preguntas}}$$

$$= \left( \frac{3 + 3 + 3}{4} \right) = 2.25$$

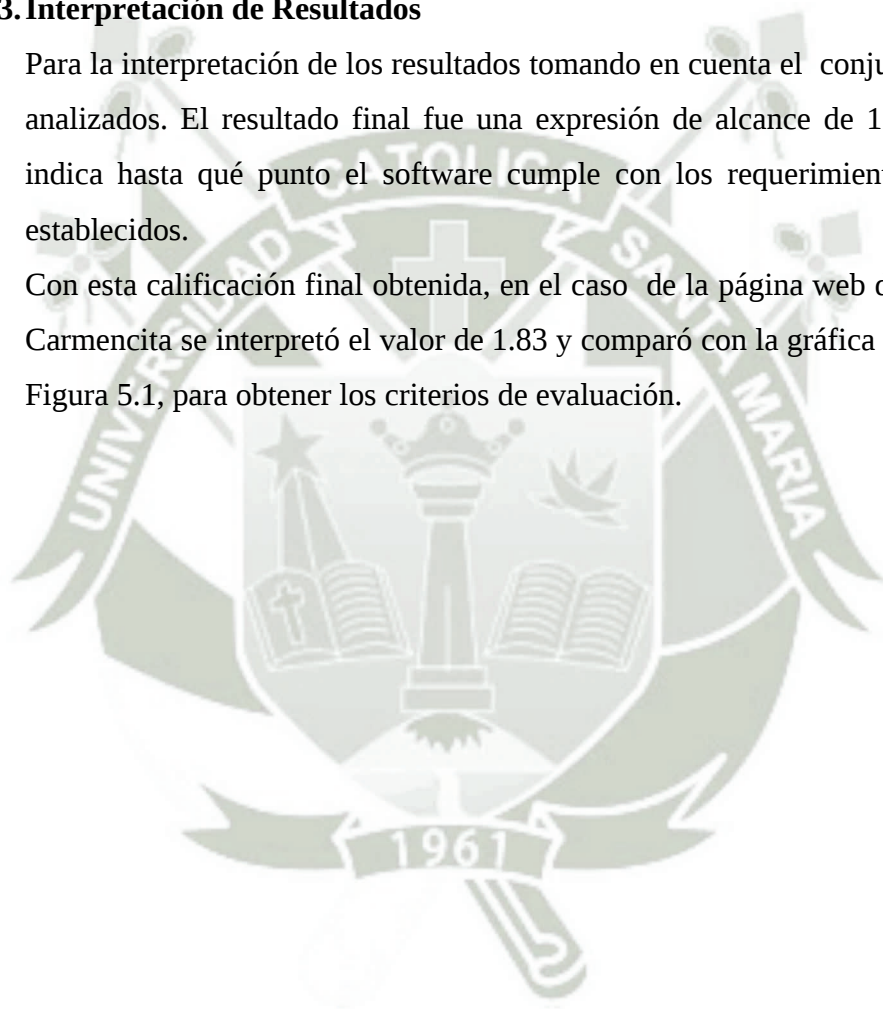
### Medición del porcentaje de calidad

$$\begin{aligned}\% \text{ calidad} &= \left( \frac{(2+1+2+2+1.5)+(2.4+1.5+1.5)+(1.8+2.25+2.25)}{11} \right) \\ &= \left( \frac{(8.5) + (5.4) + (6.3)}{11} \right) \\ &= \left( \frac{20.2}{11} \right) = 1.8363 \sim 61.21\%\end{aligned}$$

### 5.4.3. Interpretación de Resultados

Para la interpretación de los resultados tomando en cuenta el conjunto de valores analizados. El resultado final fue una expresión de alcance de 1.8383 que nos indica hasta qué punto el software cumple con los requerimientos de calidad establecidos.

Con esta calificación final obtenida, en el caso de la página web de la panadería Carmencita se interpretó el valor de 1.83 y comparó con la gráfica mostrada en la Figura 5.1, para obtener los criterios de evaluación.



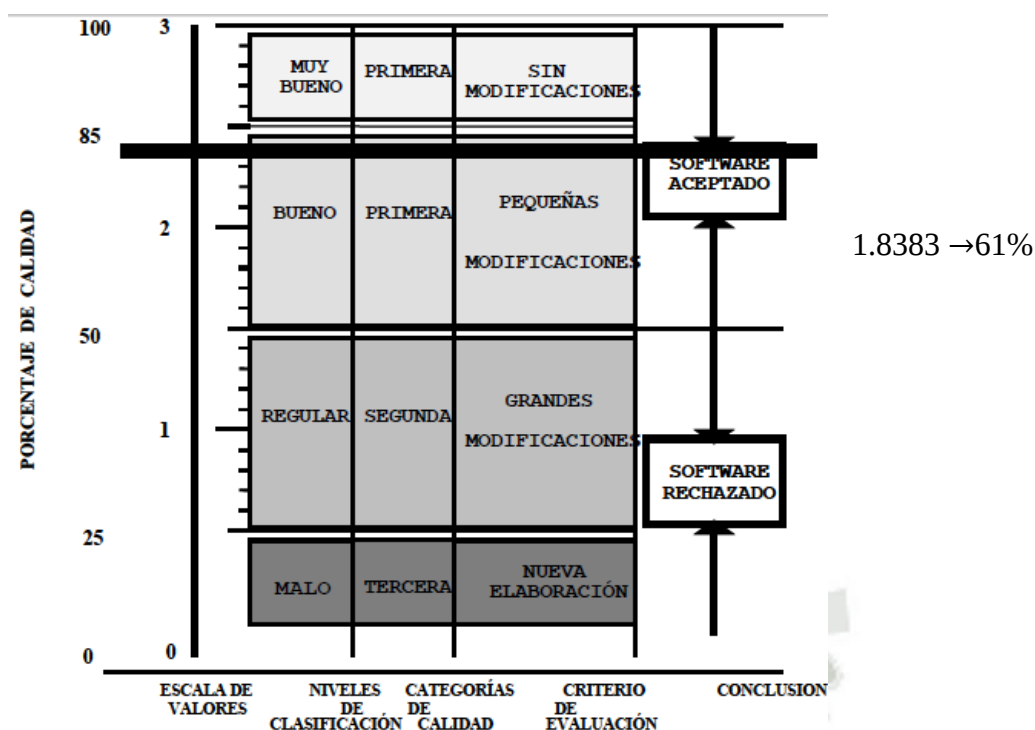


Fig.5.2. Ubicación del nivel de calidad de la página web según escala

El nivel de Clasificación del Software : BUENO  
 La Categoría de Calidad : PRIMERA  
 El Criterio de Evaluación : PEQUEÑAS DEFICIENCIAS  
 Conclusión : SOFTWARE ACEPTADO

## CONCLUSIONES

1. Se ha desarrollado una herramienta de E-marketing para el incremento de las ventas en la microempresa de panadería “Carmencita” de la ciudad de Arequipa haciendo uso de diversas aplicaciones virtuales para el comercio electrónico lo que representa grandes ventajas sobre la empresa y pueda obtener ventajas competitivas sobre otras empresas por las características y beneficios que brinda.
2. Se ha construido una página web dinámica para la microempresa Panadería Carmencita que ya cuenta con un sitio en internet, lo que le otorga un plus en competitividad y ventajas económicas, ya que ahora puede ofrecer más beneficios a sus clientes, visitantes y público en general.
3. Se ha desarrollado un modelo de E-Marketing electrónico de óptimo funcionamiento en internet, por la tecnología utilizada tiene la posibilidad de efectuar cambios y modificaciones sin necesidad de herramientas complicadas, con acceso sencillo por parte del cliente hacia el servidor, la administración, puede realizar seguimientos del proceso de ventas desde el principio hasta el final mediante tecnologías web como PHP, MySQL.
4. Se han utilizado técnicas y herramientas de Ingeniería de software en la implementación de la página, que están a la vanguardia de la construcción de sitios web e E-Business, acorde a los requerimientos de un mercado que cada día se hace más competitivo, asegurando así, una competencia frente a otras empresas pertenecientes al mismo nicho de mercado, consiguiendo la continuidad y crecimiento del negocio.

## RECOMENDACIONES

1. Para satisfacer completamente a los clientes de la Panadería “Carmencita” se debe implementar funcionalidad en nuevos navegadores para Iphones y Smartphones, con el objetivo de ampliar el acceso desde estas nuevas versiones y qué navegadores se pueden utilizar para navegar en el sistema. Como se señala en el documento actualmente solo se soportan Internet Explorer desde la versión 7.0, Mozilla Firefox desde la versión 3.6 y Google Chrome desde la versión 5.0.
2. Se recomienda implementar, en el futuro, nuevos módulos de ayuda y soporte técnico, que permitan a los usuarios registrar los problemas que se presentan durante el uso de la aplicación, de modo que se brinde información adecuada sobre las posibles deficiencias.
3. Con el objetivo de brindar una mejor experiencia de utilización al usuario, se sugiere agregar un bloque de Google Analytics para tener un conjunto de cálculos y métricas de las visitas al sitio web; se puede también incluir un bloque de formulario de contacto, el cual estaría direccionado al correo electrónico del propietario del sitio web. Así también, podría tener un menú con las estadísticas de los bloques consumidos, los tipos de estos bloques, y de igual forma el número de los bloques utilizados por página.
4. Para administrar una sede virtual, la empresa Panadería “Carmencita” debe considerar la capacitación de empleados y socios por medio de empresas especializadas, que puedan manejar y entender las herramientas que se utilizan o buscar ayuda externa en la gestión y administración de páginas virtuales sobre las herramientas empleadas.

## BIBLIOGRAFÍA

- AVELLO, R. (21 de julio del 2007). “Exe; útil herramienta para la construcción de contenidos educativos”, Valencia, España. Recuperado el 14 de octubre del 2013 de URL: <http://quadernsdigitals.net/>
- BUD E. SMITH, RYAN C. WILLIAMS 2009 Google Business Solutions All-in-One For Dummies Indianapolis, Indiana. USA: Wiley Publishing, Inc.
- CAMACHO, A. (04 de Agosto 2013).SISTEMA DE INFORMACIÓN PARA LA MICRO EMPRESA. Recuperado el 08 de octubre del 2013 de URL:<http://es.scribd.com/doc/57721594/SISTEMA-DE-INFORMACION-PARA-LA-MICRO-EMPRESA>.
- COLOMES FORMOS, XAVIER. (2009).CSS, DHTML y AJAX, Edit. ANAYA Multimedia.320 págs.
- CUBERO TORRES, SERGIO (2008) “Elaboración de contenidos con Exe learning”, Valencia, España, Disponible en:<http://www.uv.es/scubero/recursos/gestioncontenidos/eXelearning.pdf>
- DE LA TORRE, ANIBAL. (2006). “Introducción a la plataforma Moodle”, New York, USA. Disponible en:[http://www.adelat.org/media/docum/moodle/docum/23\\_cap03.pdf](http://www.adelat.org/media/docum/moodle/docum/23_cap03.pdf)
- DOLADO, JOSÉ JAVIER, 2000. “Medición para la gestión en la Ingeniería del Software”. C.Ed. RA-MA España, Febrero.345pp.

- ESPINOZA, N. (07 de Mayo 2008).LAS PEQUEÑAS Y MICROEMPRESAS EN EL PERU. Recuperado el 07 de Mayo del 2013 de URL:<http://www.slideshare.net/Nadisina/pymes-en-el-peru>
- GILMORE W, JASON 2008 Beginning PHP and MySQL: From Novice to Professional (Third Edition) EEUU, New York: Apress.
- GONZALES. Ricardo. 2013. CONCEPTOS BASICOS DE ENRUTAMIENTO. Fecha: 13 de abril 2012 : Fecha de consulta 30 de Noviembre del 2013 URL:<http://www.ditae.uat.edu.mx/ver2013/xtras/ConceptosBasicosRoutingRicardo.pdf>
- KEVIN POTTS 2008 Web Design and Marketing Solutions for Business Websites New York, USA: Springer-Verlag, Inc.
- LUKE WELLINGTHOMSON –LAURA. 2009. Desarrollo Web con PHP y MySQL Edit. ANAYA Multimedia.976 págs.



## ANEXOS

a)

### CODIGO DE LA BASE DE DATOS

```
<?php require_once('../Connections/carmencita_cone.php'); ?>

<?php

if (!function_exists("GetSQLValueString")) {

function GetSQLValueString($theValue, $theType,
    $theDefinedValue = "", $theNotDefinedValue = "")
{
    if (PHP_VERSION < 6) {
        $theValue = get_magic_quotes_gpc() ?
stripslashes($theValue) : $theValue;
    }

    $theValue = function_exists("mysql_real_escape_string") ?
mysql_real_escape_string($theValue) :
mysql_escape_string($theValue);

    switch ($theType) {
        case "text":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" :
"NULL";
            break;

        case "long":

        case "int":
            $theValue = ($theValue != "") ? intval($theValue) :
"NULL";
```

```

break;

case "double":

    $theValue = ($theValue != "") ? doubleval($theValue) :
    "NULL";

break;

case "date":

    $theValue = ($theValue != "") ? "'" . $theValue . "'" :
    "NULL";

break;

case "defined":

    $theValue = ($theValue != "") ? $theDefinedValue :
    $theNotDefinedValue;

break;

    }

return $theValue;

}

}

mysql_select_db($database_carmencita_cone, $carmencita_cone);

$query_Recordset1 = "SELECT * FROM `admin`";

$Recordset1 = mysql_query($query_Recordset1,
$carmencita_cone) or die(mysql_error());

$row_Recordset1 = mysql_fetch_assoc($Recordset1);

$totalRows_Recordset1 = mysql_num_rows($Recordset1);

mysql_select_db($database_carmencita_cone, $carmencita_cone);

```

```
$query_Recordset2 = "SELECT * FROM cat_galeria";

$Recordset2 = mysql_query($query_Recordset2,
$carmencita_cone) or die(mysql_error());

$row_Recordset2 = mysql_fetch_assoc($Recordset2);

$totalRows_Recordset2 = mysql_num_rows($Recordset2);

mysql_select_db($database_carmencita_cone, $carmencita_cone);

$query_Recordset3 = "SELECT * FROM productos";

$Recordset3 = mysql_query($query_Recordset3,
$carmencita_cone) or die(mysql_error());

$row_Recordset3 = mysql_fetch_assoc($Recordset3);

$totalRows_Recordset3 = mysql_num_rows($Recordset3);

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />

<title>Documento sin título</title>

</head>

<body>

</body>

</html>
```

```
<?php  
  
mysql_free_result($Recordset1);  
  
mysql_free_result($Recordset2);  
  
mysql_free_result($Recordset3);  
?>
```



### CODIGO DE CONEXIÓN A LA BASE DE DATOS

```
<?php  
  
# FileName="Connection_php_mysql.htm"  
  
# Type="MYSQL"  
  
# HTTP="true"  
  
$hostname_carmencita_cone = "localhost";  
$database_carmencita_cone = "carmencita";  
  
$username_carmencita_cone = "root";  
$password_carmencita_cone = "sa";  
  
$carmencita_cone = mysql_pconnect($hostname_carmencita_cone,  
$username_carmencita_cone, $password_carmencita_cone) or  
trigger_error(mysql_error(),E_USER_ERROR);  
  
?>
```

## **CODIGO FUENTE DE LA PAGINA**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"><!-- InstanceBegin
template="/Templates/Plantilla.dwt.php"
codeOutsideHTMLOIsLocked="false" -->

<head>

<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />

<!-- InstanceBeginEditable name="doctitle" -->
<title>Documento sin título</title>
<!-- InstanceEndEditable -->

<link href="estilos.css" rel="stylesheet" type="text/css" />
<style type="text/css">
.container div .div1 ul li {
    float: left;
    list-style-type: none;
}
.container div .div1 ul li a {
    color: #FFF;
    text-decoration: none;
    background-color: #960;
    display: block;
    height: 36px;
    width: 160px;
```

```

text-align: center;

line-height: 36px;

border-left-width: thin;

border-left-style: solid;

border-left-color: #FFF;

}

.container div .div1 ul li a:hover {

    background-color: #6C0;

}

.noBorder {

    border-left-style: none!important;

}

</style>

<!-- Start WOWSlider.com HEAD section -->

<link rel="stylesheet" type="text/css"
href="engine1/style.css" />

    <script type="text/javascript"
src="engine1/jquery.js"></script>

    <!-- End WOWSlider.com HEAD section -->

<!-- InstanceBeginEditable name="head" -->

<style type="text/css">

.div7 {

    font-size: 14px;

    text-align: center;

```

```
background-repeat: no-repeat;

background-image: url(images/imadiv7.jpg);

background-position: center;

color: #00F;

}

</style>

<!-- InstanceEndEditable -->

</head>

<body>

<div class="container">
<div>
<div class="div1">
<ul>
<li><a href="index.php" class="noBorder">INICIO</a></li>
<li><a href="quienes_somos.php">QUIENES SOMOS</a></li>
<li><a href="tienda.php">TIENDA</a></li>
<li><a href="cotizacion.php">COTIZACION</a></li>
<li><a href="contacto.php">CONTACTO</a></li>
</ul>
</div>

</div>
```

```
<div class="header">

<!-- Start WOWSlider.com BODY section -->

    <div id="wowslider-container1">

        <div class="ws_images"><ul>

<li></li>

<li></li>

<li></li>

<li></li>

</ul></div>

<div class="ws_bullets"><div>

<a href="#" title="1">1</a>

<a href="#" title="2">2</a>

<a href="#" title="3">3</a>

<a href="#" title="4">4</a>

</div></div>

<span class="wsl"><a href="http://wowslider.com">Slideshow
Javascript Code</a> by WOWSlider.com v4.0</span>

    <div class="ws_shadow"></div>

</div>
```

```
<script type="text/javascript"
src="engine1/wowslider.js"></script>
```

```
<script type="text/javascript"
src="engine1/script.js"></script>
```

```
<!-- End WOWSlider.com BODY section -->
```

```
<a href="#"></a>
```

```
<!-- end .header --></div>
```

```
<div class="sidebar1">
```

```
<ul class="nav">
```

```
<!-- InstanceBeginEditable name="sidebar1" -->
```

```
<li><a href="objetivos.php">OBJETIVOS</a></li>
```

```
<li><a href="nuestros_valores.php">NUESTROS VALORES</a></li>
```

```
<li></li>
```

```
<li></li>
```

```
<!-- InstanceEndEditable -->
```

```
</ul>
```

```
<!-- InstanceBeginEditable name="contenid" -->
```

```
<p class="div7"><strong> Cada mañana, bien temprano,
seleccionamos los más selectos y frescos ingredientes, para
preparar exquisitos productos, con todo nuestro cariño y
dedicación, pensando en ofrecerle lo mejor al
convertir un producto rico en un producto
especial.</strong></p>
```

```
<!-- InstanceEndEditable --><!-- end .sidebar1 --></div>

<div class="content">

<h2><!-- InstanceBeginEditable name="Titulo" --
><r>;Bienvenidos!!</r><!-- InstanceEndEditable --></h2>

<!-- InstanceBeginEditable name="Contenido" -->

<p><strong>Panaderia &quot;Carmencita&quot; te invita a
disfrutar de las mas deliciosos Panes, tortas, postres y
bocaditos que preparamos con mucho esmero y dedicacion,
nuestra experiencia refleja la calidad y sabor delicioso de
nuestros productos.</strong></p>

<p>;...Despierta nuevas sensaciones!!</p>

<p>Ofrecemos una amplia variedad de panificados con identidad
propia y propuestas para deleitar el paladar. Nuestro
personal trabaja cada día en la mejora de nuestras recetas,
utilizando ingredientes de excelencia que hacen a un sabor
único.

El control de calidad que aplicamos a nuestros procesos de
producción permiten un adecuado servicio de acuerdo a la
exigente demanda, optimizando la calidad de cada uno de los
productos.

Así mismo, estamos desarrollado un sistema de distribución a
comercios minoristas y mayoristas de la ciudad de Arequipa,
garantizando lealtad y cumplimiento en los plazos de
entrega.</p>

<!-- InstanceEndEditable -->

<!-- end .content --></div>

<div class="sidebar2">
```

```
<h4><!-- InstanceBeginEditable name="enunciado" -->Felice
Fiestas !!!!<!-- InstanceEndEditable --></h4>
```

```
<!-- InstanceBeginEditable name="cont" -->
```

```
<p><a href="#"></a></p>
```

```
<p>A todos ustedes, quienes tienen como nosotros el sueño de
construir un mundo mejor día tras día, les deseamos unas
muy FELICES FIESTAS PATRIAS! Brindemos juntos por un país
repleto de AMOR, PAZ y mucha FELICIDAD!</p>
```

```
<!-- InstanceEndEditable --><!-- end .sidebar2 --></div>
```

```
<div class="footer">
```

```
<p>El diseño del sitio es propiedad exclusiva de Fredy
arevalo -All Rights Reserved UCSM-2013 </p>
```

```
<!-- end .footer --></div>
```

```
<!-- end .container --></div>
```

```
</body>
```

```
<!-- InstanceEnd --></html>
```

b)

**INSTRUMENTO DE EVALUACIÓN SEGÚN LAS MÉTRICAS DE MCCALL**

|                     |         | EVALUADOR |   |   |   |   |     |      |
|---------------------|---------|-----------|---|---|---|---|-----|------|
| FACTOR              | MÉTRICA | 1         | 2 | 3 | 4 | 5 | TOT | PROM |
| FACILIDAD DE<br>USO | 1.      | 1         | 2 | 3 | 3 | 1 | 10  | 2    |
|                     | 2.      | 3         | 3 | 3 | 1 | 2 | 12  | 2.4  |
|                     | 3.      | 3         | 3 | 3 | 2 | 3 | 14  | 2.8  |
| INTEGRIDAD          | 4.      | 3         | 3 | 3 | 3 | 3 | 15  | 3    |
|                     | 5.      | 1         | 2 | 3 | 3 | 1 | 10  | 2    |
|                     | 6.      | 3         | 3 | 3 | 1 | 2 | 12  | 2.4  |
| CORRECCION          | 7.      | 3         | 3 | 3 | 2 | 3 | 14  | 2.8  |
|                     | 8.      | 3         | 3 | 3 | 3 | 3 | 15  | 3    |
|                     | 9.      | 1         | 2 | 3 | 3 | 1 | 10  | 2    |
| FIABILIDAD          | 10.     | 3         | 3 | 3 | 1 | 2 | 12  | 2.4  |
|                     | 11.     | 3         | 3 | 3 | 2 | 3 | 14  | 2.8  |
|                     | 12.     | 3         | 3 | 3 | 3 | 3 | 15  | 3    |
|                     | 13.     | 3         | 3 | 3 | 1 | 2 | 12  | 2.4  |
|                     | 14.     | 3         | 3 | 3 | 2 | 3 | 14  | 2.8  |
|                     | 15.     | 3         | 3 | 3 | 3 | 3 | 15  | 3    |
| EFICIENCIA          | 16.     | 1         | 2 | 3 | 3 | 1 | 10  | 2    |
|                     | 17.     | 3         | 3 | 3 | 1 | 2 | 12  | 2.4  |

| FACTOR                     | MÉTRICA | 1 | 2 | 3 | 4 | 5 | TOT | PROM |
|----------------------------|---------|---|---|---|---|---|-----|------|
| FACILIDAD DE MANTENIMIENTO | 18.     | 3 | 3 | 3 | 1 | 2 | 12  | 2.4  |
|                            | 19.     | 3 | 3 | 3 | 2 | 3 | 14  | 2.8  |
|                            | 20.     | 3 | 3 | 3 | 3 | 3 | 15  | 3    |
|                            | 21.     | 1 | 2 | 3 | 3 | 1 | 10  | 2    |
|                            | 22.     | 3 | 3 | 3 | 1 | 2 | 12  | 2.4  |
| FACILIDAD DE PRUEBA        | 23.     | 3 | 3 | 3 | 1 | 2 | 12  | 2.4  |
|                            | 24.     | 3 | 3 | 3 | 2 | 3 | 14  | 2.8  |
|                            | 25.     | 3 | 3 | 3 | 3 | 3 | 15  | 3    |
|                            | 26.     | 3 | 3 | 3 | 1 | 2 | 12  | 2.4  |
| FLEXIBILIDAD               | 27.     | 3 | 3 | 3 | 2 | 3 | 14  | 2.8  |
|                            | 28.     | 3 | 3 | 3 | 3 | 3 | 15  | 3    |
|                            | 29.     | 3 | 3 | 3 | 2 | 3 | 14  | 2.8  |
|                            | 30.     | 3 | 3 | 3 | 3 | 3 | 15  | 3    |

|                   |     |   |   |   |   |   |    |     |
|-------------------|-----|---|---|---|---|---|----|-----|
| REUSABILIDAD      | 31. | 3 | 3 | 3 | 1 | 2 | 12 | 2.4 |
|                   | 32. | 3 | 3 | 3 | 2 | 3 | 14 | 2.8 |
|                   | 33. | 3 | 3 | 3 | 3 | 3 | 15 | 3   |
|                   | 34. | 3 | 3 | 3 | 1 | 2 | 12 | 2.4 |
|                   | 35. | 3 | 3 | 3 | 2 | 3 | 14 | 2.8 |
| INTEROPERABILIDAD | 36. | 3 | 3 | 3 | 3 | 3 | 15 | 3   |
|                   | 37. | 3 | 3 | 3 | 1 | 2 | 12 | 2.4 |
|                   | 38. | 3 | 3 | 3 | 2 | 3 | 14 | 2.8 |
|                   | 39. | 3 | 3 | 3 | 3 | 3 | 15 | 3   |
| PORTABILIDAD      | 40. | 3 | 3 | 3 | 1 | 2 | 12 | 2.4 |
|                   | 41. | 3 | 3 | 3 | 2 | 3 | 14 | 2.8 |
|                   | 42. | 3 | 3 | 3 | 3 | 3 | 15 | 3   |
|                   | 43. | 3 | 3 | 3 | 1 | 2 | 12 | 2.4 |

### c) EVALUACIÓN DE CALIDAD DE SOFTWARE POR EXPERTOS

#### UNIVERSIDAD CATÓLICA DE SANTA MARIA INSTRUMENTO DE EVALUACION CALIDAD DEL SOFTWARE

Nombre del evaluador: DONALI CHARCA CANO Empresa: D. y F. SUTEM

#### 1. OPERACIÓN DEL PRODUCTO

| FACTOR           | METRICA                                                                                                          | 0 | 1 | 2 | 3 |
|------------------|------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| FACILIDAD DE USO | 1. ¿La planeación y diseño permite atender las necesidades de información y servicios del usuario?               |   |   |   | X |
|                  | 2. ¿Los nuevos usuarios tienen una interacción efectiva y familiar en base a conocimientos previos con el sitio? |   |   |   | X |
|                  | 3. ¿El entorno ayuda para permitir que nuevos usuarios apliquen el sistema?                                      |   |   |   | X |
| INTEGRIDAD       | 4. ¿La página web posee atributos de control de accesos?                                                         | X |   |   |   |
|                  | 5. ¿El sistema posee atributos que faciliten la auditoria de los accesos al software?                            | X |   |   |   |
|                  | 6. ¿Hay disponibilidad de mecanismos que controlen o protejan los datos del sitio?                               |   |   |   | X |
| CORRECCION       | 7. ¿Tiene atributos que proporcionan la implementación completa de todas las funciones requeridas?               |   |   |   | X |
|                  | 8. ¿Posee cualidades que proporcionan uniformidad en las técnicas y notaciones de diseño e implementación?       |   |   |   | X |
|                  | 9. ¿Proporciona una traza desde los requisitos a la implementación con respecto a un entorno operativo concreto? | X |   |   |   |
| FIABILIDAD       | 10. ¿Hay precisión en los resultados obtenidos?                                                                  |   |   |   | X |
|                  | 11. ¿Los resultados son consistentes con los requerimientos?                                                     |   |   |   | X |
|                  | 12. ¿Existen fallas cuando la página web esta en uso?                                                            | X |   |   |   |
|                  | 13. ¿Tiene atributos que proporcionan una estructura de módulos altamente independientes.?                       | X |   |   |   |
|                  | 14. ¿Puede el usuario restablecer fácilmente la página cuando está en uso?                                       |   |   |   | X |
|                  | 15. ¿Hay exactitud en los resultados obtenidos?                                                                  |   |   |   | X |
| E F I C          | 16. ¿Las respuestas del sistema son rápidas?                                                                     |   |   |   | X |

|  |                                                                                                   |   |  |  |  |
|--|---------------------------------------------------------------------------------------------------|---|--|--|--|
|  | 17. ¿Puede el usuario guardar su información generada con el software, en la memoria del sistema? | X |  |  |  |
|--|---------------------------------------------------------------------------------------------------|---|--|--|--|

## 2. REVISION DEL PRODUCTO

| FACTOR                     | METRICA                                                                                                                                  | 0 | 1 | 2 | 3 |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| FACILIDAD DE MANTENIMIENTO | 18. ¿El Sitio está dividido en componentes funcionales tiene acoplamiento y cohesión?                                                    |   |   |   | X |
|                            | 19. ¿El sitio está orientado hacia la simplicidad y facilidad operativa, en su manejo y mantenimiento?                                   |   |   |   | X |
|                            | 20. ¿Las aplicaciones son consistentes con conocimientos previos?                                                                        |   |   |   | X |
|                            | 21. ¿El software posibilita la implementación de una función con la menor cantidad de código posible?                                    | X |   |   |   |
|                            | 22. ¿Posee atributos que proporcionan explicaciones sobre la implementación de las funciones?                                            |   |   |   | X |
| FACILIDAD DE PRUEBA        | 23. ¿Provee una plataforma modular de prueba para solucionar problemas rápidamente?                                                      | X |   |   |   |
|                            | 24. ¿Resulta simple que el software demuestre sus fallas a través de la ejecución de pruebas?                                            |   |   |   | X |
|                            | 25. ¿Posee atributos autodescriptivos que permiten realizar pruebas sobre el sitio?                                                      | X |   |   |   |
|                            | 26. ¿Es posible observar el comportamiento del software durante su ejecución y facilitar mediciones del uso e identificación de errores? |   |   |   | X |
| FLEXIBILIDAD               | 27. ¿Existe variedad de posibilidades con las que el usuario y el sitio puedan intercambiar información?                                 | X |   |   |   |
|                            | 28. ¿Hay la posibilidad de la expansión del software en cuanto a capacidades funcionales y datos?                                        |   |   |   | X |
|                            | 29. ¿Permite el software proporcionar amplitud a las funciones implementadas?                                                            |   |   |   | X |
|                            | 30. ¿Tiene una serie de módulos que le dan potencia y flexibilidad?                                                                      | X |   |   |   |

### 3. TRANSICION DEL PRODUCTO

| FACTOR            | METRICA                                                                                                                           | 0 | 1 | 2 | 3 |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| REUSABILIDAD      | 31. ¿La página tiene varias opciones para ingresar los datos: Archivos tipo texto, archivos provenientes de otras bases de datos? |   |   |   | X |
|                   | 32. ¿El usuario entiende fácilmente los mensajes que envía el software?                                                           |   |   |   | X |
|                   | 33. ¿La página contiene módulos para visualizar gráficamente resultados?                                                          |   |   |   | X |
|                   | 34. ¿El sistema es independiente del Software?                                                                                    | X |   |   |   |
|                   | 35. ¿Los atributos del Software son independientes del hardware?                                                                  | X |   |   |   |
| INTEROPERABILIDAD | 36. ¿El sistema dispone de mecanismos que permitan intercambiar procesos y/o datos?                                               |   |   |   | X |
|                   | 37. ¿Los Atributos posibilitan el uso de protocolos de comunicación e interfaces estándar?                                        |   |   |   | X |
|                   | 38. ¿Los Atributos permiten el uso representaciones de datos estándar?                                                            |   |   |   | X |
|                   | 39. ¿Usa estructuras de datos y de tipos estándar a lo largo de todo el programa?                                                 | X |   |   |   |
| PORTABILIDAD      | 40. ¿El sistema se puede adaptar a cualquier manejador de basa de datos?                                                          |   |   |   | X |
|                   | 41. ¿Al realizar mejoras del sistema, las funciones y módulos son similares?                                                      |   |   |   | X |
|                   | 42. ¿La construir la página el sistema operativo es un factor primordial?                                                         | X |   |   |   |
|                   | 43. ¿Al desarrollar el sistema el hardware es un factor primordial?                                                               |   |   |   | X |

El nivel de Clasificación del Software : BUENA

La Categoría de Calidad : BUENA

El Criterio de Evaluación : BUENA

Conclusión : BUENA SOFTWARE ACEPTADO

  
Daniel Chama Cano  
PROGRAMADOR  
C.C. Sta. Teresita Ciudad 45

UNIVERSIDAD CATÓLICA DE SANTA MARÍA  
INSTRUMENTO DE EVALUACION CALIDAD DEL SOFTWARE

Nombre del evaluador: Francisco Abel Chambi U. Empresa: Aceosonios SF System SA

1. OPERACIÓN DEL PRODUCTO

| FACTOR           | METRICA                                                                                                          | 0( | 1 | 2 | 3 |
|------------------|------------------------------------------------------------------------------------------------------------------|----|---|---|---|
| FACILIDAD DE USO | 1. ¿La planeación y diseño permite atender las necesidades de información y servicios del usuario?               |    |   |   | X |
|                  | 2. ¿Los nuevos usuarios tienen una interacción efectiva y familiar en base a conocimientos previos con el sitio? |    |   |   | X |
|                  | 3. ¿El entorno ayuda para permitir que nuevos usuarios apliquen el sistema?                                      | X  |   |   |   |
| INTEGRIDAD       | 4. ¿La página web posee atributos de control de accesos?                                                         | X  |   |   |   |
|                  | 5. ¿El sistema posee atributos que faciliten la auditoria de los accesos al software?                            |    |   |   | X |
|                  | 6. ¿Hay disponibilidad de mecanismos que controlen o protejan los datos del sitio?                               | X  |   |   |   |
| CORRECCION       | 7. ¿Tiene atributos que proporcionan la implementación completa de todas las funciones requeridas?               | X  |   |   |   |
|                  | 8. ¿Posee cualidades que proporcionan uniformidad en las técnicas y notaciones de diseño e implementación?       |    |   |   | X |
|                  | 9. ¿Proporciona una traza desde los requisitos a la implementación con respecto a un entorno operativo concreto? |    |   |   | X |
| FIABILIDAD       | 10. ¿Hay precisión en los resultados obtenidos?                                                                  |    |   |   | X |
|                  | 11. ¿Los resultados son consistentes con los requerimientos?                                                     |    |   |   | X |
|                  | 12. ¿Existen fallas cuando la página web esta en uso?                                                            | X  |   |   |   |
|                  | 13. ¿Tiene atributos que proporcionan una estructura de módulos altamente independientes.?                       | X  |   |   |   |
|                  | 14. ¿Puede el usuario restablecer fácilmente la página cuando está en uso?                                       |    |   |   | X |
|                  | 15. ¿Hay exactitud en los resultados obtenidos?                                                                  |    |   |   | X |
| E F I C          | 16. ¿Las respuestas del sistema son rápidas?                                                                     |    |   |   | X |

|                                                                                                   |   |  |  |  |
|---------------------------------------------------------------------------------------------------|---|--|--|--|
| 17. ¿Puede el usuario guardar su información generada con el software, en la memoria del sistema? | X |  |  |  |
|---------------------------------------------------------------------------------------------------|---|--|--|--|

## 2. REVISION DEL PRODUCTO

| FACTOR                     | METRICA                                                                                                                                 | 0 | 1 | 2 | 3 |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| FACILIDAD DE MANTENIMIENTO | 18.¿El Sitio está dividido en componentes funcionales tiene acoplamiento y cohesión?                                                    |   |   |   | X |
|                            | 19.¿El sitio está orientado hacia la simplicidad y facilidad operativa, en su manejo y mantenimiento?                                   |   |   |   | X |
|                            | 20.¿Las aplicaciones son consistentes con conocimientos previos?                                                                        |   |   |   | X |
|                            | 21.¿El software posibilita la implementación de una función con la menor cantidad de código posible?                                    | X |   |   |   |
|                            | 22.¿Posee atributos que proporcionan explicaciones sobre la implementación de las funciones?                                            |   |   |   | X |
| FACILIDAD DE PRUEBA        | 23.¿Provee una plataforma modular de prueba para solucionar problemas rápidamente?                                                      | X |   |   |   |
|                            | 24.¿Resulta simple que el software demuestre sus fallas a través de la ejecución de pruebas?                                            |   |   |   | X |
|                            | 25.¿Posee atributos autodescriptivos que permiten realizar pruebas sobre el sitio?                                                      | X |   |   |   |
|                            | 26.¿Es posible observar el comportamiento del software durante su ejecución y facilitar mediciones del uso e identificación de errores? |   |   |   | X |
| FLEXIBILIDAD               | 27. ¿Existe variedad de posibilidades con las que el usuario y el sitio puedan intercambiar información?                                | X |   |   |   |
|                            | 28.¿Hay la posibilidad de la expansión del software en cuanto a capacidades funcionales y datos?                                        |   |   |   | X |
|                            | 29.¿Permite el software proporcionar amplitud a las funciones implementadas?                                                            |   |   |   | X |
|                            | 30.¿Tiene una serie de módulos que le dan potencia y flexibilidad?                                                                      | X |   |   |   |

### 3. TRANSICION DEL PRODUCTO

| FACTOR            | METRICA                                                                                                                           | 0 | 1 | 2 | 3 |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| REUSABILIDAD      | 31. ¿La página tiene varias opciones para ingresar los datos: Archivos tipo texto, archivos provenientes de otras bases de datos? |   |   |   | ✓ |
|                   | 32. ¿El usuario entiende fácilmente los mensajes que envía el software?                                                           |   |   |   | ✗ |
|                   | 33. ¿La página contiene módulos para visualizar gráficamente resultados?                                                          |   |   |   | ✗ |
|                   | 34. ¿El sistema es independiente del Software?                                                                                    | ✗ |   |   |   |
|                   | 35. ¿Los atributos del Software son independientes del hardware?                                                                  | ✗ |   |   |   |
| INTEROPERABILIDAD | 36. ¿El sistema dispone de mecanismos que permitan intercambiar procesos y/o datos?                                               |   |   |   | ✗ |
|                   | 37. ¿Los Atributos posibilitan el uso de protocolos de comunicación e interfaces estándar?                                        |   |   |   | ✗ |
|                   | 38. ¿Los Atributos permiten el uso representaciones de datos estándar?                                                            |   |   |   | ✗ |
|                   | 39. ¿Usa estructuras de datos y de tipos estándar a lo largo de todo el programa?                                                 | ✗ |   |   |   |
| PORTABILIDAD      | 40. ¿El sistema se puede adaptar a cualquier manejador de basa de datos?                                                          |   |   |   | ✗ |
|                   | 41. ¿Al realizar mejoras del sistema, las funciones y módulos son similares?                                                      |   |   |   | ✗ |
|                   | 42. ¿La construir la página el sistema operativo es un factor primordial?                                                         | ✗ |   |   |   |
|                   | 43. ¿Al desarrollar el sistema el hardware es un factor primordial?                                                               |   |   |   | ✗ |

El nivel de Clasificación del Software :.....Buena.....

La Categoría de Calidad :.....Buena.....

El Criterio de Evaluación :.....Buena Pequeñas Deficiencias.....

Conclusión :.....Buena Software Aceptado.....



UNIVERSIDAD CATÓLICA DE SANTA MARÍA  
INSTRUMENTO DE EVALUACION CALIDAD DEL SOFTWARE

Nombre del evaluador: Patricia ANDREA Lopez Empresa STUDIO 14 E.I.R.L.

1. OPERACIÓN DEL PRODUCTO

| FACTOR           | METRICA                                                                                                          | 0 | 1 | 2 | 3 |
|------------------|------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| FACILIDAD DE USO | 1. ¿La planeación y diseño permite atender las necesidades de información y servicios del usuario?               |   |   |   | X |
|                  | 2. ¿Los nuevos usuarios tienen una interacción efectiva y familiar en base a conocimientos previos con el sitio? |   |   |   | X |
|                  | 3. ¿El entorno ayuda para permitir que nuevos usuarios apliquen el sistema?                                      |   |   |   | X |
| INTEGRIDAD       | 4. ¿La página web posee atributos de control de accesos?                                                         | X |   |   |   |
|                  | 5. ¿El sistema posee atributos que faciliten la auditoria de los accesos al software?                            | X |   |   |   |
|                  | 6. ¿Hay disponibilidad de mecanismos que controlen o protejan los datos del sitio?                               |   |   |   | X |
| CORRECCION       | 7. ¿Tiene atributos que proporcionan la implementación completa de todas las funciones requeridas?               |   |   |   | X |
|                  | 8. ¿Posee cualidades que proporcionan uniformidad en las técnicas y notaciones de diseño e implementación?       |   |   |   | X |
|                  | 9. ¿Proporciona una traza desde los requisitos a la implementación con respecto a un entorno operativo concreto? | X |   |   |   |
| FIABILIDAD       | 10. ¿Hay precisión en los resultados obtenidos?                                                                  |   |   |   | X |
|                  | 11. ¿Los resultados son consistentes con los requerimientos?                                                     |   |   |   | X |
|                  | 12. ¿Existen fallas cuando la página web esta en uso?                                                            | X |   |   |   |
|                  | 13. ¿ Tiene atributos que proporcionan una estructura de módulos altamente independientes.?                      | X |   |   |   |
|                  | 14. ¿Puede el usuario restablecer fácilmente la página cuando está en uso?                                       |   |   |   | X |
|                  | 15. ¿Hay exactitud en los resultados obtenidos?                                                                  |   |   |   | X |
| E F I C          | 16. ¿Las respuestas del sistema son rápidas?                                                                     |   |   |   | X |

|  |                                                                                                   |   |  |  |  |
|--|---------------------------------------------------------------------------------------------------|---|--|--|--|
|  | 17. ¿Puede el usuario guardar su información generada con el software, en la memoria del sistema? | ✓ |  |  |  |
|--|---------------------------------------------------------------------------------------------------|---|--|--|--|

## 2. REVISION DEL PRODUCTO

| FACTOR                     | METRICA                                                                                                                                  | 0 | 1 | 2 | 3 |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| FACILIDAD DE MANTENIMIENTO | 18. ¿El Sitio está dividido en componentes funcionales tiene acoplamiento y cohesión?                                                    |   |   |   | ✓ |
|                            | 19. ¿El sitio está orientado hacia la simplicidad y facilidad operativa, en su manejo y mantenimiento?                                   |   |   |   | ✓ |
|                            | 20. ¿Las aplicaciones son consistentes con conocimientos previos?                                                                        |   |   |   | ✓ |
|                            | 21. ¿El software posibilita la implementación de una función con la menor cantidad de código posible?                                    | ✓ |   |   |   |
|                            | 22. ¿Posee atributos que proporcionan explicaciones sobre la implementación de las funciones?                                            |   |   |   | ✓ |
| FACILIDAD DE PRUEBA        | 23. ¿Provee una plataforma modular de prueba para solucionar problemas rápidamente?                                                      | ✓ |   |   |   |
|                            | 24. ¿Resulta simple que el software demuestre sus fallas a través de la ejecución de pruebas?                                            |   |   |   | ✓ |
|                            | 25. ¿Posee atributos autodescriptivos que permiten realizar pruebas sobre el sitio?                                                      | ✓ |   |   |   |
|                            | 26. ¿Es posible observar el comportamiento del software durante su ejecución y facilitar mediciones del uso e identificación de errores? |   |   |   | ✓ |
| FLEXIBILIDAD               | 27. ¿Existe variedad de posibilidades con las que el usuario y el sitio puedan intercambiar información?                                 | ✓ |   |   |   |
|                            | 28. ¿Hay la posibilidad de la expansión del software en cuanto a capacidades funcionales y datos?                                        |   |   |   | ✓ |
|                            | 29. ¿Permite el software proporcionar amplitud a las funciones implementadas?                                                            |   |   |   | ✓ |
|                            | 30. ¿Tiene una serie de módulos que le dan potencia y flexibilidad?                                                                      | ✓ |   |   |   |

### 3. TRANSICION DEL PRODUCTO

| FACTOR            | METRICA                                                                                                                           | 0 | 1 | 2 | 3 |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| REUSABILIDAD      | 31. ¿La página tiene varias opciones para ingresar los datos: Archivos tipo texto, archivos provenientes de otras bases de datos? |   |   |   | X |
|                   | 32. ¿El usuario entiende fácilmente los mensajes que envía el software?                                                           |   |   |   | X |
|                   | 33. ¿La página contiene módulos para visualizar gráficamente resultados?                                                          |   |   |   | X |
|                   | 34. ¿El sistema es independiente del Software?                                                                                    | X |   |   |   |
|                   | 35. ¿Los atributos del Software son independientes del hardware?                                                                  | X |   |   |   |
| INTEROPERABILIDAD | 36. ¿El sistema dispone de mecanismos que permitan intercambiar procesos y/o datos?                                               |   |   |   | X |
|                   | 37. ¿Los Atributos posibilitan el uso de protocolos de comunicación e interfaces estándar?                                        |   |   |   | X |
|                   | 38. ¿Los Atributos permiten el uso representaciones de datos estándar?                                                            |   |   |   | X |
|                   | 39. ¿Usa estructuras de datos y de tipos estándar a lo largo de todo el programa?                                                 | X |   |   |   |
| PORTABILIDAD      | 40. ¿El sistema se puede adaptar a cualquier manejador de basa de datos?                                                          |   |   |   | X |
|                   | 41. ¿Al realizar mejoras del sistema, las funciones y módulos son similares?                                                      |   |   |   | X |
|                   | 42. ¿La construir la página el sistema operativo es un factor primordial?                                                         | X |   |   |   |
|                   | 43. ¿Al desarrollar el sistema el hardware es un factor primordial?                                                               |   |   |   | X |

El nivel de Clasificación del Software : BUENA.....  
 La Categoría de Calidad : BUENA.....  
 El Criterio de Evaluación : BUENA ALGUNAS DEFICIENCIAS.....  
 Conclusión : BUENA SOFTWARE ACEPTADO.....



Dni 29719434

UNIVERSIDAD CATÓLICA DE SANTA MARÍA  
INSTRUMENTO DE EVALUACION CALIDAD DEL SOFTWARE

Nombre del evaluador: Rafael Mamani Empresa Producciones Ivan

1. OPERACIÓN DEL PRODUCTO

| FACTOR           | METRICA                                                                                                          | 0( | 1 | 2 | 3 |
|------------------|------------------------------------------------------------------------------------------------------------------|----|---|---|---|
| FACILIDAD DE USO | 1. ¿La planeación y diseño permite atender las necesidades de información y servicios del usuario?               |    |   |   | X |
|                  | 2. ¿Los nuevos usuarios tienen una interacción efectiva y familiar en base a conocimientos previos con el sitio? |    |   |   | X |
|                  | 3. ¿El entorno ayuda para permitir que nuevos usuarios apliquen el sistema?                                      |    |   |   | X |
| INTEGRIDAD       | 4. ¿La página web posee atributos de control de accesos?                                                         | X  |   |   |   |
|                  | 5. ¿El sistema posee atributos que faciliten la auditoria de los accesos al software?                            | X  |   |   |   |
|                  | 6. ¿Hay disponibilidad de mecanismos que controlen o protejan los datos del sitio?                               |    |   |   | X |
| CORRECCION       | 7. ¿Tiene atributos que proporcionan la implementación completa de todas las funciones requeridas?               |    |   |   | X |
|                  | 8. ¿Posee cualidades que proporcionan uniformidad en las técnicas y notaciones de diseño e implementación?       |    |   |   | X |
|                  | 9. ¿Proporciona una traza desde los requisitos a la implementación con respecto a un entorno operativo concreto? | X  |   |   |   |
| FIABILIDAD       | 10. ¿Hay precisión en los resultados obtenidos?                                                                  |    |   |   | X |
|                  | 11. ¿Los resultados son consistentes con los requerimientos?                                                     |    |   |   | X |
|                  | 12. ¿Existen fallas cuando la página web esta en uso?                                                            | X  |   |   |   |
|                  | 13. ¿ Tiene atributos que proporcionan una estructura de módulos altamente independientes.?                      | X  |   |   |   |
|                  | 14. ¿Puede el usuario restablecer fácilmente la página cuando está en uso?                                       |    |   |   | X |
|                  | 15. ¿Hay exactitud en los resultados obtenidos?                                                                  |    |   |   | X |
| E F I C          | 16. ¿Las respuestas del sistema son rápidas?                                                                     |    |   |   | X |

|  |                                                                                                   |   |  |  |  |  |
|--|---------------------------------------------------------------------------------------------------|---|--|--|--|--|
|  | 17. ¿Puede el usuario guardar su información generada con el software, en la memoria del sistema? | X |  |  |  |  |
|--|---------------------------------------------------------------------------------------------------|---|--|--|--|--|

## 2. REVISION DEL PRODUCTO

| FACTOR                     | METRICA                                                                                                                                  | 0 | 1 | 2 | 3 |
|----------------------------|------------------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| FACILIDAD DE MANTENIMIENTO | 18. ¿El Sitio está dividido en componentes funcionales tiene acoplamiento y cohesión?                                                    |   |   |   | X |
|                            | 19. ¿El sitio está orientado hacia la simplicidad y facilidad operativa, en su manejo y mantenimiento?                                   |   |   |   | X |
|                            | 20. ¿Las aplicaciones son consistentes con conocimientos previos?                                                                        |   |   |   | X |
|                            | 21. ¿El software posibilita la implementación de una función con la menor cantidad de código posible?                                    | X |   |   |   |
|                            | 22. ¿Posee atributos que proporcionan explicaciones sobre la implementación de las funciones?                                            |   |   |   | X |
| FACILIDAD DE PRUEBA        | 23. ¿Provee una plataforma modular de prueba para solucionar problemas rápidamente?                                                      | X |   |   |   |
|                            | 24. ¿Resulta simple que el software demuestre sus fallas a través de la ejecución de pruebas?                                            |   |   |   | X |
|                            | 25. ¿Posee atributos autodescriptivos que permiten realizar pruebas sobre el sitio?                                                      | X |   |   |   |
|                            | 26. ¿Es posible observar el comportamiento del software durante su ejecución y facilitar mediciones del uso e identificación de errores? |   |   |   | X |
| FLEXIBILIDAD               | 27. ¿Existe variedad de posibilidades con las que el usuario y el sitio puedan intercambiar información?                                 | X |   |   |   |
|                            | 28. ¿Hay la posibilidad de la expansión del software en cuanto a capacidades funcionales y datos?                                        |   |   |   | X |
|                            | 29. ¿Permite el software proporcionar amplitud a las funciones implementadas?                                                            |   |   |   | X |
|                            | 30. ¿Tiene una serie de módulos que le dan potencia y flexibilidad?                                                                      | X |   |   |   |

### 3. TRANSICION DEL PRODUCTO

| FACTOR            | METRICA                                                                                                                           | 0 | 1 | 2 | 3 |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|
| REUSABILIDAD      | 31. ¿La página tiene varias opciones para ingresar los datos: Archivos tipo texto, archivos provenientes de otras bases de datos? |   |   |   | X |
|                   | 32. ¿El usuario entiende fácilmente los mensajes que envía el software?                                                           |   |   |   | X |
|                   | 33. ¿La página contiene módulos para visualizar gráficamente resultados?                                                          |   |   |   | X |
|                   | 34. ¿El sistema es independiente del Software?                                                                                    | X |   |   |   |
|                   | 35. ¿Los atributos del Software son independientes del hardware?                                                                  | X |   |   |   |
| INTEROPERABILIDAD | 36. ¿El sistema dispone de mecanismos que permitan intercambiar procesos y/o datos?                                               |   |   |   | X |
|                   | 37. ¿Los Atributos posibilitan el uso de protocolos de comunicación e interfaces estándar?                                        |   |   |   | X |
|                   | 38. ¿Los Atributos permiten el uso representaciones de datos estándar?                                                            |   |   |   | X |
|                   | 39. ¿Usa estructuras de datos y de tipos estándar a lo largo de todo el programa?                                                 | X |   |   |   |
| PORTABILIDAD      | 40. ¿El sistema se puede adaptar a cualquier manejador de basa de datos?                                                          |   |   |   | X |
|                   | 41. ¿Al realizar mejoras del sistema, las funciones y módulos son similares?                                                      |   |   |   | X |
|                   | 42. ¿La construir la página el sistema operativo es un factor primordial?                                                         | X |   |   |   |
|                   | 43. ¿Al desarrollar el sistema el hardware es un factor primordial?                                                               | X |   |   |   |

El nivel de Clasificación del Software : Buena

La Categoría de Calidad : Buena

El Criterio de Evaluación : Buena

Conclusión : Buena Software aceptado

  
Rafael Mamani Mamani  
PROGRAMADOR  
C.C. Sta. Teresita Stand 143