

**ESCUELA POLITÉCNICA DEL EJÉRCITO**

**SEDE LATACUNGA**



**CARRERA DE INGENIERÍA DE SISTEMAS E  
INFORMÁTICA**

**TEMA: "UTILIZACIÓN DE LA TECNOLOGÍA DE APLICACIONES  
DE INTERNET ENRIQUECIDAS (RIA) DE MICROSOFT PARA EL  
DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA EN LÍNEA  
PARA EL SEGUIMIENTO Y ADMINISTRACIÓN DE  
ENCOMIENDAS, RUTAS, TARIFAS, BOLETOS, VEHÍCULOS, Y  
FUNCIONARIOS PARA LA COOPERATIVA INTERPROVINCIAL DE  
TRANSPORTES MACUCHI"**

**TESIS PRESENTADA COMO REQUISITO PREVIO A LA OBTENCIÓN DEL  
TÍTULO DE INGENIERO DE SISTEMAS E INFORMÁTICA**

**CRISTÓBAL GEOVANNI GÓMEZ CARRERA**

**Latacunga, Diciembre 2011**

**ESCUELA POLITÉCNICA DEL EJÉRCITO**

**CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**CERTIFICADO**

**ING. SANTIAGO JACOME (DIRECTOR)**

**ING. FABIAN MONTALUISA (CODIRECTOR)**

**CERTIFICAN:**

Que el trabajo titulado "UTILIZACIÓN DE LA TECNOLOGÍA DE APLICACIONES DE INTERNET ENRIQUECIDAS (RIA) DE MICROSOFT PARA EL DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA EN LÍNEA PARA EL SEGUIMIENTO Y ADMINISTRACIÓN DE ENCOMIENDAS, RUTAS, TARIFAS, BOLETOS, VEHÍCULOS, Y FUNCIONARIOS PARA LA COOPERATIVA INTERPROVINCIAL DE TRANSPORTES MACUCHI" realizado por el señor: Cristóbal Geovanni Gómez Carrera ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Escuela Politécnica del Ejército.

Debido a que constituye un trabajo de excelente contenido científico que coadyuvará a la aplicación de conocimientos y al desarrollo profesional, **SI** recomiendan su publicación.

El mencionado trabajo consta de UN empastado y UN disco compacto el cual contiene los archivos en formato portátil de Acrobat. Autorizan al señor: Cristóbal Geovanni Gómez Carrera que lo entregue al Ing. Santiago Jácome, en su calidad de Director de Carrera.

Latacunga, 06 de Diciembre de 2011.

-----  
Ing. Santiago Jácome

DIRECTOR DE PROYECTO

-----  
Ing. Fabian Montaluisa

CODIRECTOR DE PROYECTO

## CERTIFICACIÓN

Se certifica que el presente trabajo fue desarrollado por Cristóbal Geovanni Gómez Carrera, bajo nuestra supervisión.

---

Ing. Santiago Jácome  
DIRECTOR DE PROYECTO

---

Ing. Fabián Montaluisa  
CODIRECTOR DE PROYECTO

---

Ing. Santiago Jácome  
COORDINADOR DE LA CARRERA DE SISTEMAS E INFORMÁTICA

---

Dr. Rodrigo Vaca  
SECRETARIO ACADEMICO

## DECLARACIÓN DE RESPONSABILIDAD

Yo, Cristóbal Geovanni Gómez Carrera,

### DECLARO QUE:

El proyecto de grado denominado "UTILIZACIÓN DE LA TECNOLOGÍA DE APLICACIONES DE INTERNET ENRIQUECIDAS (RIA) DE MICROSOFT PARA EL DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA EN LÍNEA PARA EL SEGUIMIENTO Y ADMINISTRACIÓN DE ENCOMIENDAS, RUTAS, TARIFAS, BOLETOS, VEHÍCULOS, Y FUNCIONARIOS PARA LA COOPERATIVA INTERPROVINCIAL DE TRANSPORTES MACUCHI" ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan al pie de las páginas correspondientes.

Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance científico del proyecto de grado en mención.

Latacunga, 31 de Agosto del 2009.

-----  
Cristóbal Gómez Carrera

C.I. No.-0502978323

## **A U T O R I Z A C I Ó N**

Yo, Cristóbal Geovanni Gómez Carrera

Autorizo a la Escuela Politécnica del Ejército la publicación, en la biblioteca virtual de la Institución de la tesis "UTILIZACIÓN DE LA TECNOLOGÍA DE APLICACIONES DE INTERNET ENRIQUECIDAS (RIA) DE MICROSOFT PARA EL DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA EN LÍNEA PARA EL SEGUIMIENTO Y ADMINISTRACIÓN DE ENCOMIENDAS, RUTAS, TARIFAS, BOLETOS, VEHÍCULOS, Y FUNCIONARIOS PARA LA COOPERATIVA INTERPROVINCIAL DE TRANSPORTES MACUCHI" cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y autoría.

Latacunga, 06 de Diciembre del 2011

-----

Cristóbal Geovanni Gómez Carrera

C.I. No. 0502978323

## DEDICATORIA

Este arduo trabajo que ha costado mucho esfuerzo, va dedicado para todas aquellas personas que de una u otra manera pasaron por mi vida, dejando su huella en mí. Cada una de estas personas contribuyeron para llegar a ser lo que soy; gracias a ustedes.

Pero no puedo dejar de nombrar a gente demasiado importante en mi vida, quienes son la razón de mí ser y de mis esfuerzos. A mi mamá, la luz de mi vida, mi ejemplo de todo lo bueno que se puede hacer en la vida y artífice de todo lo que he logrado. Aunque a veces no la diga que la quiero, pero la amo como a ninguna otra mujer, gracias a ti estoy redactando estas breves palabras. A mi hermana, ejemplo de tenacidad, valentía, responsabilidad y muchas cosas más. A mis doblemente padres, Laura, Rosita, Luis y Cristóbal, que han sabido llenar espacios muy grandes en mi vida. A mis tíos que siempre han estado conmigo y me han tratado como un hermano más. A Lupita, otro gran ejemplo de responsabilidad y valentía, quien me supo acoger en su hogar como un hijo más. A todos mis primos, con quienes he compartido muchas cosas buenas y malas; adelante, persigan sus sueños tómenlos y háganlos realidad. A mi padre, a quien le debo mi templanza e independencia.

Finalmente, hubiera querido festejar este logro con dos personas que desde el cielo seguro nos cuidan. Mi primo Francisco Gómez, fuiste un amigo sincero y bueno, siempre te voy a extrañar. Mario Benítez, quien fue como mi padre, siempre te estaré agradecido por la confianza, por el apoyo, por las enseñanzas y a quien llevaré en mi corazón siempre. Ustedes no se fueron, se quedaron en nuestros corazones. Algún rato nos volveremos a ver...

Gracias padre celestial por todo esto.

## **A G R A D E C I M I E N T O**

Todos los éxitos conseguidos hasta ahora y los que se obtendrán en el futuro tendrán como primer responsable a la Universidad quién me proporcionó las herramientas para desempeñarme adecuadamente en mi profesión, para ella son los primeros agradecimientos.

Mi familia fueron quienes patrocinaron el proyecto universitario en todos los sentidos. Sin su apoyo no habría sido posible el logro de una meta tan importante como lo es terminar una carrera profesional, a ellas un sincero agradecimiento por su esfuerzo y dedicación.

Finalmente al director y codirector del proyecto que brindaron todas las facilidades del caso.

Cristóbal Gómez Carrera

## **R E S U M E N**

El presente proyecto de tesis se ha dividido en 5 capítulos:

Capítulo 1. Habla acerca de la evolución de las aplicaciones web, sus características, su importancia, cual es la tendencia actual y lo que nos pudiera esperar en un futuro próximo.

Capítulo 2. Trata sobre las RIA (Rich Internet Application), sus características, beneficios, objetivos, arquitectura y las tecnologías para el desarrollo de RIA

Capítulo 3. Se abarca más a profundidad la tecnología Silverlight, orígenes, evolución, características, uso junto con WCF y sobre las herramientas que nos permiten trabajar con esta.

Capítulo 4. Es meramente es la construcción de la aplicación para la cooperativa de transportes "Macuchi", aplicando los principios metodológicos de Extreme Programming.

Capítulo 5. Son las conclusiones y recomendaciones a las cuales se ha llegado, al terminar este proyecto de tesis.



## **A B S T R A C T**

This thesis project has been in 5 chapters:

Chapter 1: This chapter talks about the evolution of the web applications, their features, its importance, which is the current trend and what we might expect in the future.

Chapter 2: This chapter attempts to explain the Rich Internet Applications, their features, benefits, goals, architecture and the technologies used on this.

Chapter 3: This chapter talks more in-depth about the Silverlight technology, origins, evolution, characteristics, WCF compatibility and about the tools to work with it.

Chapter 4: This chapter represents the construction of the web application to Cooperativa de Transportes "Macuchi", applying the Extreme Programming methodology.

Chapter 5: This chapter describes the conclusions and the recommendations of the actual thesis project.

## INTRODUCCIÓN

El presente proyecto muestra los beneficios de la aplicación tecnología para crear Rich Internet Applications de Microsoft para la construcción de una aplicación web para la cooperativa de transportes "Macuchi".

Como herramienta de desarrollo, se uso Microsoft Visual Studio 2008, para diseño Microsoft Expression Blend 3.0, junto Silverlight 3.0 que están enfocadas al desarrollo de aplicaciones RIA. Como fuente de datos, se uso Microsoft SQL Server 2008 y para comunicar al cliente con el servidor, WCF (Windows Communication Foundation).

La aplicación contempla el desarrollo de un sitio web para el seguimiento y administración de encomiendas, rutas, tarifas, boletos, vehículos, oficinas y funcionarios de la cooperativa de transportes "Macuchi". Usando la metodología de desarrollo XP (Extreme Programming).

## INDICE GENERAL

CAPÍTULO 1.....	1
EVOLUCIÓN DE LAS APLICACIONES WEB .....	1
1.1. INTRODUCCIÓN .....	1
1.2. DEFINICIÓN .....	2
1.3. WEB 1.0 .....	5
1.3.1. DEFINICIÓN .....	5
1.3.2. CARACTERÍSTICAS DE WEB 1.0 .....	7
1.3.3. TECNOLOGÍAS .....	7
1.4. WEB 2.0 .....	12
1.4.1. DEFINICIÓN .....	12
1.4.2. IMPORTANCIA DE LA WEB 2.0 .....	20
1.4.3. CARACTERÍSTICAS .....	22
1.4.4. TECNOLOGÍAS .....	22
1.4.5. INTERNET SOCIAL (REDES SOCIALES) .....	33
1.4.6. WIKIS .....	38
1.4.7. BLOGGING (WEBLOG) .....	46
1.4.8. (REALLY SIMPLE SYNDICATION) RSS FEEDS .....	53
1.4.9. ESCRITORIOS VIRTUALES .....	55
1.5. WEBS DEL FUTURO .....	62
CAPITULO 2 .....	65
RICH INTERNET APPLICATION (RIA) .....	65
2.1. INTRODUCCIÓN .....	65
2.2. CARACTERÍSTICAS .....	67
2.3. OBJETIVOS .....	68
2.4. BENEFICIOS .....	69
2.5. ARQUITECTURA DE UNA APLICACIÓN RIA .....	70
2.6. HERRAMIENTAS DE DESARROLLO PARA RIA .....	72
CAPITULO 3 .....	73
TECNOLOGÍA SILVERLIGHT 3 .....	73
3.1. INTRODUCCIÓN .....	73
3.2. EVOLUCIÓN .....	78
3.3. FUNDAMENTOS DE SILVERLIGHT .....	80
3.3.1. CARACTERÍSTICAS DE SILVERLIGHT 3 .....	81

3.4. MICROSOFT .NET...	84
3.4.1. BCL...	85
3.4.2. CLR...	87
3.5. COMUNICACIONES CON WCF...	90
3.6. HERRAMIENTAS DE DESARROLLO...	94
3.6.1. VISUAL STUDIO 2008...	94
3.6.2. MICROSOFT EXPRESSION BLEND 3...	96
CAPITULO 4...	99
CONSTRUCCIÓN DEL SISTEMA EN LÍNEA PARA LA COOPERATIVA "MACUCHI"...	99
4.1. PLANIFICACIÓN DE PROYECTOS...	100
4.1.1. HISTORIAS DE USUARIO...	100
4.1.2. REALEASE PLANNING...	140
4.1.3. VELOCIDAD DEL PROYECTO...	141
4.1.4. ITERACIONES...	142
4.1.5. PROGRAMACIÓN EN PAREJA...	146
4.2. DISEÑO...	146
4.2.1. DISEÑOS SIMPLES...	146
4.2.2. GLOSARIO DE TÉRMINOS...	147
4.2.3. RIESGOS...	148
4.2.4. FUNCIONALIDAD EXTRA...	149
4.2.5. REUNIONES DIARIAS...	149
4.2.6. DIAGRAMA DE CLASE...	150
4.2.7. MODELO ENTIDAD RELACIÓN...	150
4.3. CODIFICACIÓN...	150
4.4. PRUEBAS...	152
CAPITULO 5...	153
CONCLUSIONES Y RECOMENDACIONES...	153
5.1. CONCLUSIONES...	153
5.2. RECOMENDACIONES...	156
BIBLIOGRAFÍA...	157
ANEXOS...	159

## CONTENIDO DE IMÁGENES

FIGURA 1.1 MAPA MEME DE LA WEB 2.0 ... ..	14
FIGURA 2.1 ARQUITECTURA DE UNA APLICACIÓN RIA ... ..	71
FIGURA 4.1 FASES DE LA METODOLOGÍA EXTREME PROGRAMMING ... ..	99
FIGURA 4.2 DISEÑO SIMPLE DE LAS PANTALLAS EN EL SISTEMA ... ..	147
FIGURA 4.3 ARQUITECTURA DE LA APLICACIÓN ... ..	152

## CAPÍTULO 1

### 1. EVOLUCIÓN DE LAS APLICACIONES WEB

#### 1.1. Introducción.

Desde sus inicios hasta la actualidad, la Web ha ido evolucionando. Hoy, nos encontramos en una etapa de esta evolución marcada por servicios y características particulares que requieren nuevas técnicas de desarrollo de aplicaciones Web y también nuevas tecnologías o, en algunos casos, nuevos usos de tecnologías existentes.

En sus orígenes, la Web era utilizada por los navegantes para obtener información en un entorno estático, con páginas en HTML que sufrían pocas actualizaciones y no tenían interacción con el usuario. La unidad de trabajo era la página Web y, de esta forma, los usuarios navegaban entre múltiples páginas a través de hipervínculos. En esta conceptualización, el elemento central es el servidor Web con su información. Generalmente, estos sitios consistían en sitios corporativos, sitios de noticias, sitios de información específica.

El siguiente paso en esta evolución consistió en utilizar la red como comunidades, en las que los propios navegantes o usuarios de la Web pudieran interactuar, ya sea entre ellos o con otros sitios que ofrecían servicios. Durante esta fase aparecieron servicios tales como foros de discusión, servicios de e-banking, servicios de mensajería, compras electrónicas basadas en comunidades, etc.

El paso actual, es la visión de la Web como una plataforma, es decir, como el medio para la distribución de servicios y para la

construcción de información, su selección, etc. También, detrás de esta nueva visión, surgen nuevos modelos de negocio basados en la web como plataforma. Si bien no existe una forma única en que las aplicaciones de la Web 2.0 deben desarrollarse, existen algunas herramientas y tecnologías que permiten su desarrollo.

Parte de la nueva tendencia para el desarrollo de aplicaciones web consiste en:

- Transformar las aplicaciones de escritorio en aplicaciones basadas en la Web. En este sentido, el software como servicio es uno de los modelos de negocio que apoya esta transformación.
- Estandarizar las interfaces, utilizando XHTML, XML u otros estándares.
- Separación del contenido de la presentación.
- Utilizar sindicación de contenidos.
- Incorporar la técnica AJAX en las aplicaciones, de forma tal de poder trabajar con unidades de navegación más granulares y específicas, en lugar de navegar a través de páginas HTML.

## **1.2. Definición.**

Se puede definir como aquellas aplicaciones 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.

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.

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.

En la actualidad existen disponibles una gran variedad de lenguajes y tecnologías disponibles en el mercado, sean estas open source o software propietario, en las cuales podemos crear y desarrollar completamente una aplicación web con muchas ventajas visuales atractivos al usuario. Como ejemplos más visibles de aplicaciones web, podemos destacar los web mail, ventas online, subastas online, wikis, foros de discusión, web logs, redes sociales, juegos, etc.

Prácticamente no hay limitaciones, las aplicaciones web pueden hacer casi todo lo que está disponible para aplicaciones tradicionales: acceder al mouse, al teclado, ejecutar audio o video, mostrar animaciones, soporte para arrastrar y soltar, y otros tipos de tecnologías de interacción usuario-aplicación.



### **Ventajas generales de las aplicaciones Web.**

- El usuario puede acceder fácilmente a estas aplicaciones empleando un navegador web o similar.
- Si es por internet, el usuario puede entrar desde cualquier lugar del mundo donde tenga un acceso a internet.
- Pueden existir miles de usuarios pero una única aplicación instalada en un servidor, por lo tanto se puede actualizar y mantener una única aplicación y todos sus usuarios verán los resultados inmediatamente.
- Emplean tecnologías como Java, JavaFX, JavaScript, DHTML, Flash, Ajax, Silverlight que dan gran potencia a la interfaz de usuario.
- Emplean tecnologías que permiten una gran portabilidad entre diferentes plataformas. Por ejemplo, una aplicación web flash podría ejecutarse en un dispositivo móvil, en una computadora con Windows, Linux u otro sistema, en una consola de videojuegos, etc.
- Las aplicaciones web requieren poco o nada de espacio en disco. Además suelen ser livianas.

### **Desventajas generales de las aplicaciones Web.**

- Las aplicaciones web requieren navegadores web totalmente compatibles para funcionar. Incluso muchas veces requieren las extensiones apropiadas y actualizadas para operar.
- Muchas veces requieren una conexión a internet para funcionar, si la misma se interrumpe, no es posible utilizarla más. De todas maneras, en ocasiones, pueden ser descargadas e instaladas localmente para su uso offline.
- Muchas no son de código abierto, perdiendo flexibilidad.

- La aplicación web desaparece si así lo requiere el desarrollador o si el mismo se extingue. Las aplicaciones tradicionales, en general, pueden seguir usándose en esos casos.
- El usuario, en general, no tiene libertad de elegir la versión de la aplicación web que quiere usar. Un usuario podría preferir usar una versión más antigua, hasta que la nueva sea probada.
- En teoría, el desarrollador de la aplicación web puede rastrear cualquier actividad que el usuario haga. Esto puede traer problemas de privacidad.

### **1.3. Web 1.0.**

#### **1.3.1. Definición.**

La Web 1.0 estuvo presente desde el año 1991 al 2003, es la forma más básica que existe con navegadores de sólo texto bastante rápidos. Después surgió el HTML que hizo las páginas web más agradables a la vista, así como los primeros navegadores visuales tales como Internet Explorer y Netscape.

Las aplicaciones web 1.0 son de sólo lectura. El usuario no puede interactuar con el contenido de la página, es decir nada de comentarios, respuestas, citas, etc., estando totalmente limitado a lo que el Web máster sube a ésta. Web 1.0 se refiere a un estado de la World Wide Web, y cualquier página web diseñada con un estilo anterior del fenómeno de la Web 2.0. Es en general un término que ha sido creado para describir la Web antes del impacto de la fiebre punto com en el 2001, que es visto por muchos como el momento en que el internet dio un giro.

Inicialmente las páginas Web se escribían en el lenguaje de etiquetas HTML y permitían visualizar texto, imágenes e hipervínculos, lo cual era suficiente para que empresas y particulares pudieran exponer y vender sus productos a través de Internet, emergiendo así con extraordinaria fuerza el Comercio electrónico.

En este periodo de tiempo las páginas Web eran estáticas, es decir cualquier persona del mundo que visitaba una web, podía ver la misma información o la misma página web. En la web 1.0 el usuario es pasivo, solo puede visitar la web, consultarla y como máximo enviar algún dato al servidor mediante formularios.

Cuando la información en las webs se hizo muy grande y muy difícil de consultar, se comenzaron a utilizar lenguajes de programación que permitían al Servidor crear páginas a medida para cada usuario, las cuales contenían solo la información que este buscaba, es decir se añadió inteligencia al servidor para que el decidiera qué información debía enviar al usuario.

Esta inteligencia del Servidor la proporcionan programas escritos en lenguajes como Perl, PHP, ASP, Python, entre otros. Mientras que el lenguaje JavaScript permitía hacer las páginas web más animadas. La información con la que se construyen estas páginas, llamadas dinámicas, reside en una base de datos alojada en el servidor.

Estas páginas web se llaman dinámicas porque presentan distinta información dependiendo del internauta que las visita, aquí el usuario no es totalmente pasivo, porque puede

modificar la información que quiere ver, esta nueva tecnología se podría denominar Web 1.5, algo intermedio y que comenzó a dar origen a la Web 2.0.

#### **1.3.2. Características de Web 1.0.**

- Las páginas web son documentos estáticos que jamás se actualizaban.
- Se extendió el uso de framesets o Marcos para el desarrollo de páginas webs.
- Extensiones propias del HTML como el parpadeo y las marquesinas, etiquetas introducidas durante la guerra de navegadores web.
- Libros de visitas online o guest books
- Botones GIF, casi siempre a una resolución típica de 88x31 píxeles en tamaño promocionando navegadores web u otros productos.
- Formularios HTML enviados vía email. Un usuario llenaba un formulario y después de hacer clic se enviaba a través de un cliente de correo electrónico, con el problema que en el código se podía observar los detalles del envío del correo electrónico.

#### **1.3.3. Tecnologías.**

##### **HTML**

En la era de la Web 1.0, la tecnología predominante para el desarrollo de páginas web era el HTML que es acrónimo de HyperText Markup Language (*Lenguaje de Marcado de Hipertexto*). Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en

forma de etiquetas, rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script por ejemplo Javascript, el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

Fue creado en 1986 por el físico nuclear Tim Berners Lee, el cual tomo dos herramientas preexistentes: El concepto de Hipertexto conocido también como link o ancla, permite conectar dos elementos entre si y el SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML no es propiamente un lenguaje de programación como C++ o Visual Basic, sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto algún error de sintaxis que se presente éste no lo detectará y se visualizara en la forma como éste lo entienda.

El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos Windows, Bloc de notas, en Unix, editor vi o simplemente MS Office Word. El conjunto de etiquetas que se creen, se deben guardar con la extensión .htm o .html. Estos documentos pueden ser mostrados por los browsers de páginas Web en Internet, como Netscape Navigator, Opera, Microsoft Internet Explorer, Google Chrome, entre otros.

También existe el HTML Dinámico (DHTML), que es una mejora de Microsoft de la versión 4.0 de HTML que le permite crear efectos especiales como, por ejemplo, texto que vuela desde la página palabra por palabra o efectos de transición al estilo de anuncio publicitario giratorio entre página y página.

Para crear una página web se pueden utilizar varios programas especializados en esto, como por ejemplo, el Microsoft Front Page o el Adobe Dreamweaver. Otra forma de diseñar un archivo .html, es copiar todo en el Bloc de Notas del Windows, ya que este sencillo programa cumple con un requisito mínimo que es la posibilidad de trabajar con las etiquetas con las que trabaja este lenguaje.

### **Estructura de los documentos de HTML**

Si se tiene en cuenta el contenido del documento, todos los documentos de HTML bien escritos comparten una estructura en común. Un documento de HTML empieza con la etiqueta <HTML>, que es la que encerrará el documento actual. Contiene dos secciones primordiales: la cabecera y el cuerpo encerrados respectivamente por los elementos <HEAD> cabeza y <BODY> cuerpo.

La cabecera puede contener información y siempre contiene el título del documento encerrado por el elemento <TITLE>. En el cuerpo se encuentra todo el contenido del documento, ya sea, texto, imágenes, sonidos, hipervínculos, etc.

Un documento escrito en HTML contiene las siguientes etiquetas en el siguiente orden:

Ejemplo:

```
<HTML>
  <HEAD>
    <TITLE> Título de mi página de Internet
  </TITLE>
  </HEAD>
  <BODY>
    <H1>
      <CENTER> Primera página </CENTER>
    </H1>
    <P>
      Texto que se va a mostrar en la página
      web!!! </P>
    </BODY>
</HTML>
```

En el ejemplo anterior, dentro de las etiquetas <BODY> y </BODY>, vamos a especificar todo el contenido a mostrar en la página web mediante la combinación de diferentes etiquetas que nos van a permitir dar formato las letras, ordenar el texto por párrafos, dar la apariencia de diferentes títulos, incrustar imágenes, agregar caracteres especiales, permitir la navegación a otras páginas con la incrustación de Hipervínculos, entre otras muchas etiquetas que son parte del HTML.

Sin HTML no existiría el Internet que hoy conocemos, sino se hubiera inventado, seguiríamos con el aburrido FTP (Protocolo de Transferencia de Ficheros) mandándonos o bajando archivos que a veces ni sabíamos lo que eran, sin tener la posibilidad de visualizarlo antes. Ahora por medio de

una página llena de colores y botones por la cual se puede navegar, se puede tener acceso a bastante información y archivos.

El lenguaje de HTML abrió una puerta al mundo permitiéndoles a las personas expresar sus ideas por medio de páginas y mostrárselas a todas las personas de todos los países.

### **G I F**

La tecnología G I F (Graphics Interchange Format) usada en la Web 1.0, es un formato gráfico utilizado ampliamente en la World Wide Web, tanto para imágenes como para animaciones.

El formato fue creado por CompuServe en 1987 por una compañía de comunicaciones estadounidense para dotar de un formato de imagen en color para sus áreas de descarga de ficheros, sustituyendo su temprano formato RLE en blanco y negro. G I F llegó a ser muy popular porque podía usar el algoritmo de compresión LZW (Lempel Ziv Welch) para realizar la compresión de la imagen, que era más eficiente que el algoritmo Run-Lenght Encoding (RLE) usado por los formatos PCX y MacPaint. Por lo tanto, imágenes de gran tamaño podían ser descargadas en un razonable periodo de tiempo, incluso con módems muy lentos.

G I F es un formato sin pérdida de calidad para imágenes con hasta 256 colores, limitados por una paleta restringida a este número de colores. Por ese motivo, con imágenes con más de 256 colores, debe adaptarse reduciendo sus colores, produciendo la consecuente pérdida de calidad.



## 1.4. Web 2.0.

### 1.4.1. Definición.

La Web 2.0 es la representación de la evolución de las aplicaciones tradicionales hacia aplicaciones web enfocadas al usuario final. El Web 2.0 es una actitud y no precisamente una tecnología. También se la puede definir como la transición que se ha dado de aplicaciones tradicionales hacia aplicaciones que funcionan a través de la web. Dichas aplicaciones están enfocadas al usuario final. Se trata de aplicaciones que generen colaboración y de servicios que reemplacen las aplicaciones de escritorio.

El término Web 2.0 nos remonta al momento en que Dale Dougherty de <sup>1</sup>O'Reilly Media utilizó este término en una conferencia en la que compartió una lluvia de ideas junto a Craig Cline de MediaLive. En dicho evento se hablaba del renacimiento y evolución de la web. Esta conferencia dio la pauta para la "Web 2.0 conference" que se ha venido realizando anualmente en la ciudad de San Francisco en Estados Unidos de América, con eventos adicionales utilizando la marca en otros países.

En la charla inicial del Web Conference se planteó los principios que debían tener las aplicaciones Web 2.0:

- **La web es la plataforma.**- Es el pilar fundamental en el cual se sustentan los demás principios. La propia red la que hace de base, y es allí donde los programas van a ejecutarse liberándose así del enorme lastre que supone la portabilidad a/desde las diferentes plataformas, del pago

---

<sup>1</sup> Es una empresa editorial estadounidense creada por Tim O'Reilly principalmente enfocada a libros relacionados con la programación informática

por licencia, de las instalaciones en el cliente, de los requisitos mínimos y de las actualizaciones de software en el cliente. El software y hardware propio ya no va a ser una barrera que saltar, puesto que las aplicaciones van a correr en la red y por tanto bastará con una conexión a internet y algún dispositivo de acceso (PC, móvil, PDA, iPod Touch, Laptop) para poder interactuar con ellas.

Todas las herramientas de la web 2.0 están basadas en que tanto el software como la información están alojados en internet y no en el disco rígido de la PC del usuario. Lo que implica un paso del modelo Desktop al Modelo Webtop. Nace entonces el concepto de web participativa, donde hay un usuario que deja de ser netamente consumidor para convertirse en proveedor de contenidos y estos contenidos se alojan ya no en forma privada sino que quedan en bases de datos que son compartidas entre varios usuarios. Como ejemplo podemos tomar los portales de YouTube, Slide Share, Flickr, entre otros.

En este principio son los servicios y no los productos los que van a cobrar una mayor importancia, puesto que éstos van a superar a las aplicaciones tradicionales en todo momento. El software como servicio no va a tener ninguna limitación de hardware en el cliente y por tanto va a poder permitir que el trabajo en red sea casi transparente y muy sencillo para el usuario final, evitando grandes inversiones de hardware, su permanente mantenimiento, configuración y actualización, en definitiva, nos va a permitir crear, mantener y simplificar redes que funcionando con las

plataformas tradicionales en muchos casos hubiera sido imposible crear o mantener.

<sup>2</sup> Se puede visualizar Web 2.0 como un sistema de principios y prácticas que conforman un verdadero sistema solar de sitios que muestran algunos o todos esos principios, a una distancia variable de ese núcleo.

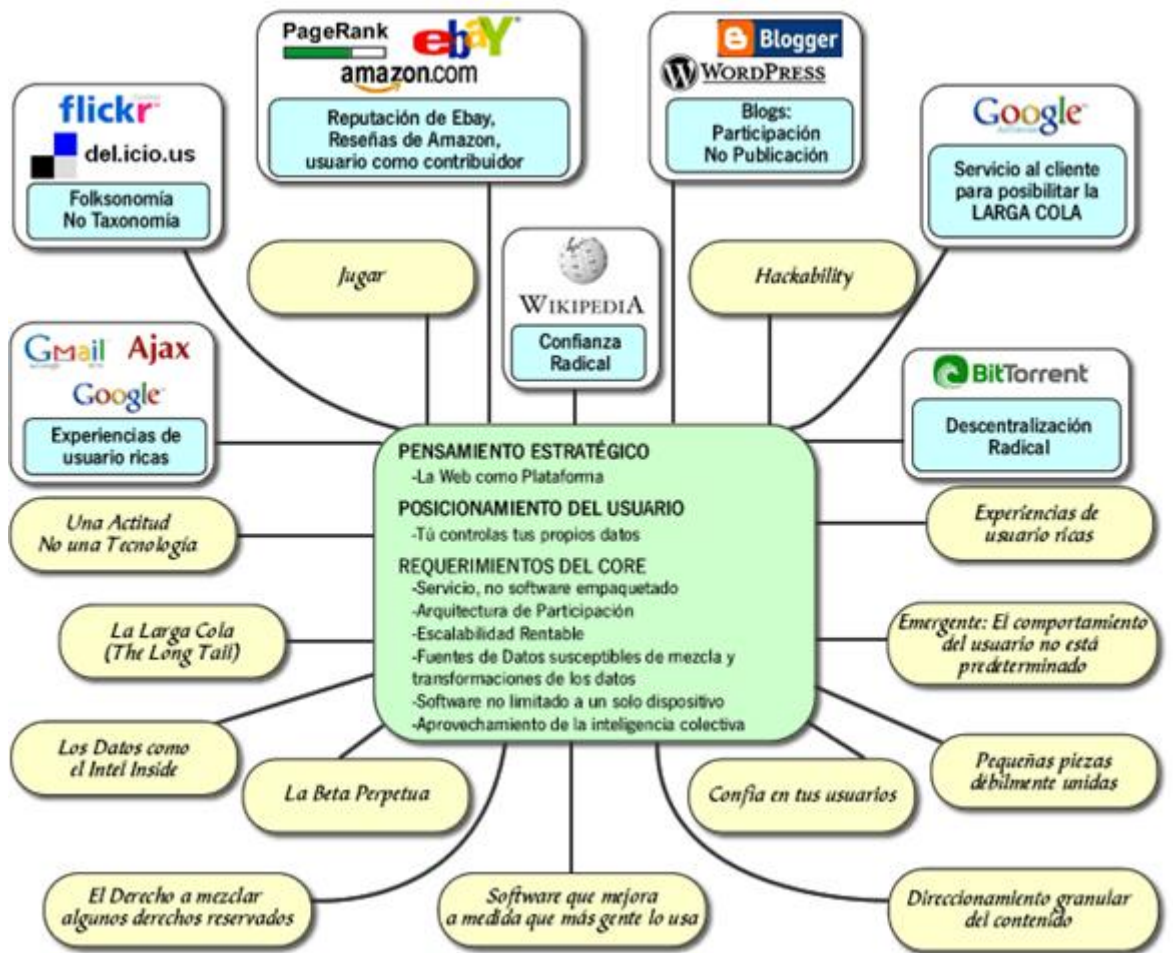


Figura 1.1 Mapa meme de la Web 2.0

La figura anterior muestra un Mapa meme que fue desarrollado mediante una lluvia de ideas en una conferencia en O'Reilly Media en el año 2004. Que

<sup>2</sup> Traducción: <http://oreilly.com/web2/archive/what-is-web-20.html>

manifiesta muchas ideas que irradia desde el núcleo de la Web 2.0 para representar sus principales cambios.

El pionero en adoptar el término "La web como plataforma" fue Netscape en términos del viejo paradigma del software. Su buque insignia era el navegador web, una aplicación de escritorio, y su estrategia era utilizar su dominio en el mercado de los navegadores para crear un mercado de productos de servidor de gama alta. Netscape, el abanderado de la Web 1.0 promovió un webtop para sustituir al escritorio o desktop, y planeó poblar ese webtop con las actualizaciones de información y applets insertados en el webtop por los proveedores de información que comprarían los servidores de Netscape. Sin embargo, al final, los navegadores web y los servidores web resultaron ser productos básicos, y el valor se desplazó hacia los servicios ofrecidos sobre la plataforma web.

Google, por el contrario, comenzó su vida como una aplicación web nativa, nunca vendida o empaquetada, sino siempre entregada como un servicio, con clientes pagando, directamente o indirectamente, por el uso de ese servicio. Ninguna de las dificultades de la vieja industria del software están presentes. No hay programación de las actualizaciones de las versiones del software, sencillamente mejora continua. Ninguna licencia o venta, sencillamente uso. Ningún tipo de portabilidad a diferentes plataformas de forma que los clientes puedan ejecutar el software en su propio equipo, sencillamente, una colección masiva de PCs escalables en los que corren sistemas

operativos de software abierto junto con aplicaciones y utilidades de su propia cosecha que nunca nadie de fuera de la compañía consigue ver.

- **La información es lo que mueve al Internet.-** Existe una diferencia entre los símbolos tanto de la Web 1.0 y de la Web 2.0, Google requiere una capacidad que Netscape nunca necesitó: gestión de la base de datos. Google no es sencillamente una colección de herramientas software, es una base de datos especializada. Sin los datos, las herramientas son inútiles; sin el software, los datos son inmanejables.

El licenciamiento del software y el control sobre las APIs, que era la palanca de poder en la Web 1.0, es irrelevante porque el software no necesita ser distribuido sino ejecutado, y también porque sin la capacidad de recoger y de gestionar los datos, el software es de poca utilidad. De hecho, el valor del software es proporcional a la escala y al dinamismo de los datos que ayuda a gestionar.

- **Efectos de la red movidos por una arquitectura de participación.-** DoubleClick, una empresa que brinda soluciones de publicidad en internet, se vio limitado por su modelo de negocio. Apoyó en los años 90 el concepto de que la web trataba de publicación, no participación; que los publicistas, no los consumidores, deben ser los que deciden; que el tamaño importaba, y que Internet cada vez estaba más dominada por los sitios web situados en la cima según las estadísticas de MediaMetrix y otras compañías que valoraban los anuncios de la web.

Como consecuencia, DoubleClick cita orgulloso en su web más de 2000 implementaciones exitosas de su software, mientras que Yahoo Search Marketing y Google AdSense, empresas dedicadas a la publicidad sobre internet, dan cada uno servicio a centenares de millares de publicistas.

El éxito de Yahoo y de Google fue fruto de la comprensión de lo que <sup>3</sup>Chris Anderson cita como "the long tail" (la larga cola), el poder colectivo de los sitios web pequeños que conforman la gran mayoría del contenido de la web. Las ofertas de DoubleClick requieren un contrato formal de venta, limitando su mercado a unos pocos miles de sitios web grandes. Yahoo y Google se las ingenieron para permitir la colocación del anuncio prácticamente en cualquier página web. Lo que es más, evitaron los formatos de publicidad preferidos por los publicistas y las agencias de publicidad como banners y popups (ventanas emergentes), en favor de los anuncios de texto, mínimamente intrusivos, sensibles al contexto y amigables para el consumidor.

En fin, se puede citar muchas historias de éxito de la Web 2.0 donde se demuestra este mismo comportamiento. Un ejemplo claro de esto es <sup>4</sup>eBay, permitiendo las transacciones ocasionales de tan solo algunos dólares entre simples individuos, actuando como un intermediario automatizado. <sup>5</sup>Napster aunque cerrado por razones legales construyó su red no mediante la construcción de una base de datos centralizada de canciones, sino

---

<sup>3</sup> Periodista, escritor y conferenciante, es el editor en jefe de la revista Wired.

<sup>4</sup> Sitio destinado a la subasta de productos a través de Internet.

<sup>5</sup> Fue un servicio de distribución de archivos de música (en formato MP3)

creando un sistema en el que cada individuo que descargaba algo también se convertía en un servidor, es decir, alguien del que otros se descargaban algo, y así creció la red.

- **La innovación surge de características distribuidas por desarrolladores independientes.**- Aquí se aprovecha la denominada "Inteligencia Colectiva" que es el principio fundamental de los gigantes que lideran la Web 2.0.

Los hipervínculos constituyen los cimientos de la web debido a que a medida que los usuarios agregan nuevo contenido, y sitios web nuevos, se enlazan con la estructura de la web gracias a otros usuarios que descubren el contenido y enlazan con él. De forma muy parecida a la sinapsis del cerebro, donde las asociaciones llegan a ser más fuertes a través de la repetición o la intensidad, la red de conexiones crece orgánicamente como resultado de la actividad colectiva de todos los usuarios de la web.

Compañías que adoptan esta idea, han dejado su marca en la web:

Yahoo!, la primera gran historia del éxito de Internet, nació como un catálogo, o un directorio de enlaces o links, un agregado del mejor trabajo de millares, después millones de usuarios de la web. A pesar de que Yahoo! ha avanzado hacia el negocio de crear muchos tipos de contenido, su papel como portal del trabajo colectivo de los usuarios de la red sigue siendo la base de su valor.

La innovación de Google en la búsqueda, que rápidamente le convirtió en el indiscutible líder del mercado de la búsqueda, fue PageRank que es una marca registrada y patentada por Google que ampara una familia de algoritmos utilizados para asignar de forma numérica la relevancia de los documentos o páginas web indexados por un motor de búsqueda para proporcionar mejores resultados de búsqueda, en lugar de usar sólo las características de los documentos.

El producto de eBay es la actividad colectiva de todos sus usuarios; como la web en sí misma, eBay crece orgánicamente en respuesta a la actividad del usuario, y el papel de la compañía es el de habilitador de un contexto en el cual pueda tener lugar esa actividad del usuario. Lo que es más, la ventaja competitiva de eBay proviene casi enteramente de la masa crítica de compradores y de vendedores, que convierte a cualquier nuevo competidor que ofrezca servicios similares significativamente menos atractivo.

Wikipedia, una enciclopedia libre y en línea basada en la inverosímil idea de que una entrada puede ser agregada por cualquier usuario de la web, y corregida por cualquier otro, es un experimento radical de confianza, aplicando la máxima de <sup>6</sup> Eric Raymond de que "con ojos suficientes, todos los fallos son superficiales" para la generación de contenido. Wikipedia está ya entre las 100 webs más visitadas, y muchos piensan que llegará a estar entre las 10 de la cima en poco tiempo.

---

<sup>6</sup> Figura líder en el movimiento del Open Source



- **La web para todos.**- La universalización en el acceso a los medios tecnológicos exige nuevas herramientas de colaboración al alcance de todos. La Web 2.0 permite, mediante mecanismos muy simples que cualquier individuo pueda publicar. Esto mismo democratiza el uso de internet al ampliar las posibilidades de sólo lectura.

#### **1.4.2. Importancia de la web 2.0.**

A pesar de la relativa corta antigüedad de la web, la influencia que esta ejerce sobre gran cantidad de aspectos cotidianos, es cada día mayor. Hace apenas un par de años el correo electrónico era reservado a unos cuantos, contar con una representaba cierta posición dentro de algún ámbito académico o laboral. Con la rápida evolución de las comunicaciones en la última década, se ha situado como un aspecto básico del mundo informático.

Antiguamente los sitios web eran desarrollados principalmente por empresas, gobiernos y organizaciones con cierta madurez en el mercado. Los navegadores eran simples espectadores del contenido proporcionado por unos pocos editores privilegiados de sitios webs. La información difundida en la red, contenía exclusivamente el punto de vista del escritor, o redactor.

Con una lenta y prácticamente imperceptible evolución, la mayoría de los buscadores pioneros en la red, se constituyeron en portales que proporcionaban servicios que iban desde cuentas de correo donde 10 Mb de almacenamiento resultaba sorprendente, servidores de almacenaje de páginas estáticas, como el caso de

geocities.com ; pasando por los servicios de salones de chat y su posterior evolución a la mensajería instantánea.

Web 2.0 puede a primera vista parecer un concepto redundante y falta de importancia. Adaptarse a los cambios siempre ha resultado difícil, más aun con el apresurado estilo de vida actual, en el cual se nos haría muy difícil regresar a la forma de utilizar la red de hace cinco años, o cuando el administrador del sitio en donde tenemos nuestras cuentas de mail nos esté recordando diariamente que se ha excedido los 5 MB de espacio disponible. Abordar la evolución de la red, no se refiere exclusivamente a las mejoras en materia de telecomunicaciones, el abaratamiento de los dispositivos de almacenamiento, el incremento de los servidores actuales y las aplicaciones web en auge.

La verdadera importancia de la Web 2.0 radica en el cambio de mentalidad de los usuarios referente a las nuevas tecnologías y a su correcto aprovechamiento en función del beneficio de los cibernautas. Los usuarios dejaron de ser simples espectadores; se han convertido en escritores, revisores, editores, críticos y productores del propio contenido de la web.

Uno de los aspectos más relevantes es que se ha definido nuevos e interesantes proyectos en Internet y se preocupa por brindar mejores soluciones para el usuario final. Muchos aseguran que hemos reinventado lo que era el Internet, otros hablan de burbujas e inversiones, pero la realidad es que la evolución natural del medio realmente ha propuesto cosas más interesantes.

#### 1.4.3. Características.

- La web como plataforma.
- Aprovechar la Inteligencia Colectiva.
- Gestión de Base de Datos como competencia básica.
- Fin del ciclo de actualizaciones de software.
- Modelos de programación ligera, fácil plantillado.
- Software no limitado a un solo dispositivo.
- Experiencias enriquecedoras del usuario.

#### 1.4.4. Tecnologías.

Existe una gran variedad de tecnologías para el desarrollo de aplicaciones Web 2.0, pero a continuación se van a detallar las más sobresalientes:

##### **CSS**

Cascading Style Sheets, que en español quiere decir Hojas de Estilo en Cascada, es un lenguaje con el cual especificamos el diseño que se va a aplicar a los diferentes controles que contiene nuestra página web. Con esto logramos una separación de diseño y contenido.

Se puede especificar mediante un archivo con extensión .css, siendo esta la manera de programar más potente, porque separa completamente las reglas de formateo para la página HTML de la estructura básica de la página. En este archivo se especifica cómo se ha de mostrar las diferentes etiquetas HTML en parámetros de color, fuente, alineación del texto, tamaño y otras características no visuales como definir el volumen de un sintetizador de voz, entre otras características.

Otra forma de aplicar las reglas de estilos es incrustarla dentro de un documento HTML dentro del elemento <head>. De esta manera se obtiene el beneficio de separar la información del estilo, del código HTML propiamente dicho. Se puede optar por copiar la hoja de estilo incrustada de una página a otra. En general, la única vez que se usa una hoja de estilo interna, es cuando se quiere proporcionar alguna característica a una página Web en un simple fichero.

La opción menos recomendada si se quiere tener páginas web realmente bien estructuradas, es mantener los estilos en línea, que es un método para insertar el lenguaje de estilo de página, directamente, dentro de una etiqueta HTML. Esta manera de proceder no es totalmente adecuada. El incrustar la descripción del formato dentro del documento de la página Web, a nivel de código se convierte en una tarea larga, tediosa y poco elegante de resolver el problema de la programación de la página. Este modo de trabajo se podría usar de manera ocasional si se pretende aplicar un formato con prisa, al vuelo. No es todo lo claro o estructurado que debería ser, pero funciona. Este es el método recomendado para maquetar correos electrónicos en HTML.

Como ventajas del uso de CSS se puede apuntar:

- Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Los navegadores permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio web, con lo que aumenta considerablemente la accesibilidad. Por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo

para aumentar el tamaño del texto o remarcar más los enlaces.

- Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario. Por ejemplo, para ser impresa, mostrada en un dispositivo móvil, o ser leída por un sintetizador de voz.
- El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño siempre y cuando no se utilice estilo en línea.

## **A J A X**

Asynchronous Javascript And XML (Javascript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

Fue creado en 2005, la historia de las tecnologías que permiten Ajax se remonta a una década antes con la iniciativa de Microsoft en el desarrollo de Scripting Remoto. Sin embargo, las técnicas para la carga asíncrona de contenidos en una página existente sin requerir recarga completa remontan al tiempo del elemento iframe introducido en Internet Explorer 3 en 1996 y el tipo de elemento layer introducido en Netscape 4 en 1997, abandonado durante las primeras etapas de desarrollo de Mozilla. Ambos tipos de elemento tenían el atributo src que podía tomar cualquier

dirección URL externa, y cargando una página que contenga Javascript que manipule la página paterna, pueden lograrse efectos parecidos al Ajax.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se requieren al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado o scripting language en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores, dado que está basado en estándares abiertos como Javascript y Document Object Model (DOM).

### **Tecnologías incluidas en Ajax**

Ajax es una combinación de cuatro tecnologías ya existentes:

- XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño que acompaña a la información.
- Document Object Model (DOM) accedido con un lenguaje de scripting por parte del usuario, especialmente implementaciones ECMAScript como JavaScript y JScript, para mostrar e interactuar dinámicamente con la información presentada.

- El objeto XMLHttpRequest para intercambiar datos de forma asíncrona con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto iframe en lugar del XMLHttpRequest para realizar dichos intercambios.
- XML es el formato usado generalmente para la transferencia de datos solicitados al servidor, aunque cualquier formato puede funcionar, incluyendo HTML preformateado, texto plano, JSON y hasta EBM L.

Ajax no constituye una tecnología en sí, sino que es un término que engloba a un grupo de éstas que trabajan conjuntamente.

#### **Problemas e Inconvenientes**

- Las páginas con AJAX son más difíciles de desarrollar que las páginas estáticas.
- Las páginas creadas dinámicamente mediante peticiones sucesivas AJAX, no son registradas de forma automática en el historial del navegador, así que haciendo clic en el botón de "Volver" del navegador, el usuario no será devuelto a un estado anterior de la página, en cambio puede volver a la última página que visitó. Soluciones incluyen el uso de IFrames invisible para desencadenar cambios en el historial del navegador y el cambio de la porción de anclaje de la dirección después de un #.
- Los motores de búsquedas no entienden JavaScript. La información en la página dinámica no se almacena en los registros del buscador.

- Hay problemas usando Ajax entre nombres de dominios. Eso es una función de seguridad.
- El sitio con Ajax usa más recursos en el servidor.
- Es posible que páginas con Ajax no puedan funcionar en teléfonos móviles, PDA u otros aparatos. Ajax no es compatible con todo el software para ciegos u otras discapacidades.

### **Adobe Flex**

Flex es un framework open source para el desarrollo de aplicaciones web intuitivas que hacen mucho más fácil a las personas interpretar y ver los datos. Es un marco de trabajo de código abierto gratuito y altamente productivo para la creación de aplicaciones web expresivas que se implantan coherentemente en exploradores, ordenadores de sobremesa y sistemas operativos, aprovechando los tiempos de ejecución de Adobe Flash Player y Adobe AIR. Aunque pueden crearse aplicaciones Flex únicamente mediante el marco de trabajo de Flex, Adobe Flash Builder (anteriormente, Adobe Flex Builder) puede acelerar el desarrollo mediante funciones como códigos inteligentes, depuración interactiva estratificada y diseño visual del aspecto de la interfaz de usuario.

Además es un término que agrupa una serie de tecnologías publicadas desde Marzo de 2004 por Macromedia, ahora bajo el nombre de Adobe, para dar soporte al despliegue y desarrollo de Aplicaciones Enriquecidas de Internet, basadas en su plataforma propietaria Flash.



Flex fue inicialmente liberado como una aplicación de la J2EE o biblioteca de etiquetas JSP que compilaba el lenguaje de marcas Flex (MXML) y ejecutaba mediante ActionScript aplicaciones Flash (archivos SWF binarios). Versiones posteriores de Flex soportan la creación de archivos estáticos que son compilados, y que pueden ser distribuidos en línea sin la necesidad de tener una licencia de servidor.

El objetivo de Flex es permitir a los desarrolladores de aplicaciones web construir rápida y fácilmente Aplicaciones de Internet Ricas, también llamadas RIAs. En un modelo multicapa, las aplicaciones Flex son el nivel de presentación.

Flex pone en relieve el desarrollo de Interfaces gráficas de usuario usando un lenguaje XML llamado MXML. Tiene varios componentes y características que aportan funcionalidades tales como Servicios Web, objetos remotos, arrastrar y soltar, columnas ordenables, gráficas, efectos de animación y otras interacciones simples. El cliente solo carga la aplicación una vez, mejorando así el flujo de datos frente a aplicaciones basadas en HTML (PHP, ASP, JSP, CFMX), las cuales requieren de ejecutar plantillas en el servidor para cada acción. El lenguaje y la estructura de archivos de Flex buscan el desacoplamiento de la lógica y el diseño.

El servidor Flex también actúa como un gateway permitiendo al cliente comunicarse con servicios web XML y objetos remotos (tales como Coldfusion CFCs, clases Java, y cualquiera que soporte el formato de mensajes de acciones). Las aplicaciones desarrolladas sobre la plataforma Flex pueden interactuar con otras tecnologías del lado servidor a través de la invocación de Web Services, REST o bajo el

formato binario AMF. La interacción a través de AMF es bastante eficiente y de éste se han realizado implementaciones para diferentes lenguajes siendo el más popular Java.

A través de los años y de la evolución de Flex como tecnología, se han presentado diferentes versiones que a continuación se detalla:

### **Flex 2**

- Flex 2 cambia el modelo de licencias para abrir la puerta a una versión libre de esta tecnología, denominada Flex Framework.
- El nuevo Flex Builder 2 está basado en el entorno de desarrollo Eclipse. Los servicios orientados a empresas seguirán estando disponibles para aquellos que necesitan características avanzadas, tales como el testeo automático.
- Flex 2 introduce el uso de una nueva versión del lenguaje de scripts ActionScript 2, ActionScript 3, que requiere reproductor Flash 9 o posterior para su funcionamiento.
- Flex será el primer producto de Macromedia en ser etiquetado como producto de Adobe, empezando por la versión 2.0.

### **Flex 3**

El 13 de febrero del 2008, Adobe anunció el lanzamiento de la versión 3 de su kit de desarrollo (SDK) para la plataforma Adobe Flex. La característica más notable de este lanzamiento es su integración nativa con la plataforma Adobe Air que permite ejecutar aplicaciones RIA directamente en el escritorio. Las nuevas funcionalidades de Adobe Flex 3 incluyen:

- Soporte nativo para Adobe Air.
- Nuevo Framework para manejar caches persistentes dentro de la plataforma Flash.
- Mejoras en el IDE comercializado por Adobe, Flex Builder 3.
- Integración con Creative Suite 3.
- Inclusión del componente Advanced DataGrid.
- Cambios en el tipo de licenciamiento (primeros pasos en el proceso de migración del SDK a código abierto).

#### **Flex 4**

La versión 4 de Flex fue distribuida al público en Marzo del 2010. Esta nueva versión busca facilitar el flujo de trabajo entre diseñadores y desarrolladores de aplicaciones RIA. Los principales objetivos Flex 4 son los siguientes:

- Diseño en mente.- La arquitectura de personalización (Skinning en inglés) se simplificó.
- Productiva del desarrollador.- Se mejoró el desempeño del compilador y se perfeccionó el proceso de enlazamiento de datos (data binding en inglés) de los componentes.
- Evolución del framework.- Se añadieron nuevos componentes y se modificó el SDK para aprovechar las características del nuevo Flash Player 10.

Uno de los cambios más notables en Flex 4 es la incorporación de una nueva arquitectura de componentes llamada Spark; ésta reemplaza a algunos de los componentes que existían en el espacio de nombres MX que era usado en las versiones anteriores de Flex.

## Java FX

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 planeados.

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. El objetivo de JavaFX es el de simplificar en buen grado la complejidad que de otra forma demanda Java a la hora del desarrollo en sí.

Un factor muy importante que hace a JavaFX realmente atractivo es su costo. Si uno desea desarrollar aplicaciones en Flash, posee ciertas funciones que son gratuitas, pero el paquete entero de desarrollo posee un costo que para algunos puede ser prohibitivo bajo ciertas circunstancias. En cambio, JavaFX es completamente gratuito. Puedes descargar de inmediato el SDK desde su página oficial, y aprovechar la versión 1.1 recientemente publicada.

Si bien tiene varias cosas que parecen atractivas, serán los desarrolladores quienes determinen el futuro de JavaFX. Existen algunas limitaciones, como que JavaFX sólo es compatible con Windows y Mac de momento, pero habrá que

ver más allá de los sistemas operativos compatibles. Flash está muy firme, mientras que Silverlight está haciendo lo posible por robar terreno. Si JavaFX desea entrar a la lucha, deberá convencer a los desarrolladores de que es una herramienta más adecuada, más flexible y más productiva.

### **OpenLaszlo**

OpenLaszlo es una plataforma de código libre que ofrece una tecnología de libre distribución para el desarrollo de Rich Internet Applications. Los dos pilares de OpenLaszlo son el lenguaje LZX y el Laszlo Presentation Server.

LZX es un lenguaje descriptivo basado en XML y JavaScript que se ocupa de la capa de presentación de las aplicaciones RIA y ofrece una definición de interfaces GUI simple y extensa. El Laszlo Presentation Server o LPS es un servlet Java que compila las aplicaciones LZX en ficheros ejecutables SWF. Además sirve y cachea esas aplicaciones para ser ejecutadas por cualquier navegador Web con el plugin de Macromedia, Flash Player 5 o superior, activado.

Con OpenLaszlo, los desarrolladores serán capaces de:

- Desarrollar RIAs basadas en lenguajes de programación estándar como lo son XML y JavaScript.
- Desplegar las aplicaciones en cualquier servidor de aplicaciones basados en J2EE o en cualquier contenedor de Servlets Java corriendo bajo Linux, Unix, Windows o MacOS X.
- Mostrar las aplicaciones en cualquier navegador Web con el plugin Flash Player 5 o superior activado. Más o

menos el 97% de los usuarios de Internet disponen de este plugin en su navegador habitual.

#### **1.4.5. Internet Social (Redes sociales).**

Son estructuras sociales compuestas de grupos de personas, las cuales están conectadas por uno o varios tipos de relaciones, tales como amistad, parentesco, intereses comunes o que comparten conocimientos. Las Redes son formas de interacción social, definida como un intercambio dinámico entre personas, grupos e instituciones en contextos de complejidad. Un sistema abierto y en construcción permanente que involucra a conjuntos que se identifican en las mismas necesidades y problemáticas y que se organizan para potenciar sus recursos.

En las redes sociales en Internet tenemos la posibilidad de interactuar con otras personas aunque no las conozcamos, el sistema es abierto y se va construyendo obviamente con lo que cada suscriptor aporta a la red, cada nuevo miembro que ingresa transforma al grupo en otro nuevo. La red no es lo mismo si uno de sus miembros deja de ser parte.

Intervenir en una red social empieza por hallar allí otros con quienes compartir nuestros intereses, preocupaciones o necesidades y aunque no sucediera más que eso, eso mismo ya es mucho porque rompe el aislamiento que suele aquejar a la gran mayoría de las personas, lo cual suele manifestarse en retraimiento y otras veces en excesiva vida social sin afectos comprometidos.

Las redes sociales en Internet suelen posibilitar que pluralidad y comunidad se conjuguen y allí quizás esté gran parte de

toda la energía que le da vida a los grupos humanos que conforman esas redes. Las redes sociales dan al anónimo popularidad, al discriminado integración, al diferente igualdad, al mal humorado educación y así muchas cosas más. La fuerza del grupo permite sobre el individuo cambios que de otra manera podrían ser difíciles y genera nuevos vínculos afectivos y de negocios.

Alrededor de 2001 y 2002 surgen los primeros sitios que fomentan redes de amigos. Hacia 2003 se hacen populares con la aparición de sitios tales como Friendster, Tribe y Myspace. Rápidamente algunas empresas ingresan a las redes sociales. Google lanza en enero de 2004 Orkut apoyando un experimento que uno de sus empleados realizaba en su tiempo libre. En 2005 ingresan Yahoo 360° y otros.

Básicamente el funcionamiento comienza cuando una vez montado el soporte técnico, un grupo de iniciadores invitan a amigos y conocidos a formar parte de la red social, cada miembro nuevo puede traer consigo muchos nuevos miembros y el crecimiento de esa red social puede ser geométrico.

Y he aquí que se transforma en un interesante negocio. Un buen ejemplo de esto es Facebook, una red social enfocada a estudiantes, muy similar a Myspace, con millones de usuarios registrados y donde ha habido una importante inversión publicitaria de parte de Microsoft.

El software germinal de las redes sociales parte de la teoría de los Seis grados de separación, según la cual toda la gente del planeta está conectada a través de no más de seis personas. De hecho, existe una patente en EEUU conocida

como "Six Degrees Patent" por la que ya han pagado Tribe y LinkedIn. Hay otras muchas patentes que protegen la tecnología para automatizar la creación de redes y las aplicaciones relacionadas con éstas.

Estas redes sociales se basan en la teoría de los seis grados, seis grados de separación es la teoría de que cualquiera en la Tierra puede estar conectado a cualquier otra persona en el planeta a través de una cadena de conocidos que no tiene más de seis intermediarios. La teoría fue inicialmente propuesta en 1929 por el escritor húngaro Frigyes Karinthy en una corta historia llamada Chains. El concepto está basado en la idea que el número de conocidos crece exponencialmente con el número de enlaces en la cadena, y sólo un pequeño número de enlaces son necesarios para que el conjunto de conocidos se convierta en la población humana entera.

El término red social es acuñado principalmente a los antropólogos ingleses John Barnes y Elizabeth Bott, ya que, para ellos resultaba imprescindible considerar lazos externos a los familiares, residenciales o de pertenencia a algún grupo social. Los fines que han motivado la creación de las llamadas redes sociales son varios, principalmente, es el diseñar un lugar de interacción virtual, en el que millones de personas alrededor del mundo se concentran con diversos intereses en común.

Recogida también en el libro "Six Degrees: The Science of a Connected Age" del sociólogo Duncan Watts, y que asegura que es posible acceder a cualquier persona del planeta en tan solo seis saltos.

Según esta Teoría, cada persona conoce de media, entre amigos, familiares y compañeros de trabajo o escuela, a unas



100 personas. Si cada uno de esos amigos o conocidos cercanos se relaciona con otras 100 personas, cualquier individuo puede pasar un recado a 10.000 personas más tan solo pidiendo a un amigo que pase el mensaje a sus amigos.

Estos 10.000 individuos serían contactos de segundo nivel, que un individuo no conoce pero que puede conocer fácilmente pidiendo a sus amigos y familiares que se los presenten, y a los que se suele recurrir para ocupar un puesto de trabajo o realizar una compra. Cuando preguntamos a alguien, por ejemplo, si conoce una secretaria interesada en trabajar estamos tirando de estas redes sociales informales que hacen funcionar nuestra sociedad. Este argumento supone que los 100 amigos de cada persona no son amigos comunes. En la práctica, esto significa que el número de contactos de segundo nivel será sustancialmente menor a 10.000 debido a que es muy usual tener amigos comunes en las redes sociales.

Si esos 10.000 conocen a otros 100, la red ya se ampliaría a 1.000.000 de personas conectadas en un tercer nivel, a 100.000.000 en un cuarto nivel, a 10.000.000.000 en un quinto nivel y a 1.000.000.000.000 en un sexto nivel. En seis pasos, y con las tecnologías disponibles, se podría enviar un mensaje a cualquier lugar individuo del planeta.

En 1967, el psicólogo estadounidense Stanley Milgram ideó una nueva manera de probar la Teoría, que él llamó "el problema del pequeño mundo". El experimento del mundo pequeño de Milgram consistió en la selección al azar de varias personas del medio oeste estadounidense para que enviaran tarjetas postales a un extraño situado en Massachusetts, situado a varios miles de millas de distancia.

Los remitentes conocían el nombre del destinatario, su ocupación y la localización aproximada. Se les indicó que enviaran el paquete a una persona que ellos conocieran directamente y que pensarán que fuera la que más probabilidades tendría, de todos sus amigos, de conocer directamente al destinatario. Esta persona tendría que hacerlo mismo y así sucesivamente hasta que el paquete fuera entregado personalmente a su destinatario final.

Aunque los participantes esperaban que la cadena incluyera al menos cientos de intermediarios, la entrega de cada paquete solamente llevó, como promedio, entre cinco y siete intermediarios. Los descubrimientos de Milgram fueron publicados en "Psychology Today" e inspiraron la frase seis grados de separación.

El origen de las redes sociales se remonta, al menos, a 1995, cuando Randy Conrads crea el sitio web classmates.com. Con esta red social se pretende que la gente pueda recuperar o mantener el contacto con antiguos compañeros del colegio, instituto, universidad, etcétera.

En estas comunidades, un número inicial de participantes envían mensajes a miembros de su propia red social invitándoles a unirse al sitio. Los nuevos participantes repiten el proceso, creciendo el número total de miembros y los enlaces de la red. Los sitios ofrecen características como actualización automática de la libreta de direcciones, perfiles visibles, la capacidad de crear nuevos enlaces mediante servicios de presentación y otras maneras de conexión social en línea. Las redes sociales también pueden crearse en torno a las relaciones comerciales.

Las herramientas informáticas para potenciar la eficacia de las redes sociales online (software social), operan en tres ámbitos, "las 3 Cs", de forma cruzada:

- Comunicación (nos ayudan a poner en común conocimientos).
- Comunidad (nos ayudan a encontrar e integrar comunidades).
- Cooperación (nos ayudan a hacer cosas juntos).

Las redes sociales continúan avanzando en Internet a pasos agigantados, especialmente dentro de lo que se ha denominado Web 2.0 y Web 3.0, y dentro de ellas, cabe destacar un nuevo fenómeno que pretende ayudar al usuario en sus compras en Internet: las redes sociales de compras. Las redes sociales de compras tratan de convertirse en un lugar de consulta y compra. Un espacio en el que los usuarios pueden consultar todas las dudas que tienen sobre los productos en los que están interesados, leer opiniones y escribirlas, votar a sus productos favoritos, conocer gente con sus mismas aficiones y, por supuesto, comprar ese producto en las tiendas más importantes con un solo clic. Esta tendencia tiene nombre, se llama Shopping 2.0.

#### **1.4.6. Wikis.**

Un wiki o una wiki (del hawaiano wiki, "rápido") es un sitio web cuyas páginas pueden ser editadas por múltiples voluntarios a través del navegador web. Los usuarios pueden crear, modificar o borrar un mismo texto que comparten. Los textos o páginas wiki tienen títulos únicos. Si se escribe el título de una página wiki en algún lugar del wiki entre dobles corchetes, esta palabra se convierte en un enlace web a la página wiki.

En una página sobre alpinismo, por ejemplo, puede haber una palabra como piolet o brújula que esté marcada como palabra perteneciente a un título de página wiki. La mayor parte de las implementaciones de wikis indican en el URL de la página el propio título de la página wiki (en Wikipedia ocurre así: <http://es.wikipedia.org/wiki/Alpinismo>), facilitando el uso y comprensibilidad del link fuera del propio sitio web. Además, esto permite formar en muchas ocasiones una coherencia terminológica, generando una ordenación natural del contenido.

La aplicación de mayor peso y a la que le debe su mayor fama hasta el momento ha sido la creación de enciclopedias colectivas, género al que pertenece la Wikipedia. Existen muchas otras aplicaciones más cercanas a la coordinación de informaciones y acciones, o la puesta en común de conocimientos o textos dentro de grupos.

La mayor parte de los wikis actuales conservan un historial de cambios que permite recuperar fácilmente cualquier estado anterior y ver qué usuario hizo cada cambio, lo cual facilita enormemente el mantenimiento conjunto y el control de usuarios destructivos. Habitualmente, sin necesidad de una revisión previa, se actualiza el contenido que muestra la página wiki editada.

El origen de los wikis está en la comunidad de patrones de diseño, cuyos integrantes los utilizaron para escribir y discutir patrones de programación. El primer WikiWeb fue creado por Ward Cunningham, quien inventó y dio nombre al concepto wiki, y produjo la primera implementación de un

servidor WikiWiki para el repositorio de patrones del Portland (Portland Pattern Repository) en 1995. En palabras del propio Cunningham, un wiki es la base de datos en línea más simple que pueda funcionar (the simplest online database that could possibly work). El wiki de Ward aún es uno de los sitios wiki más populares.

### **Ventajas**

La principal utilidad de un wiki es que permite crear y mejorar las páginas de forma instantánea, dando una gran libertad al usuario, y por medio de una interfaz muy simple. Esto hace que más gente participe en su edición, a diferencia de los sistemas tradicionales, donde resulta más difícil que los usuarios del sitio contribuyan a mejorarlo.

Dada la gran rapidez con la que se actualizan los contenidos, la palabra "wiki" adopta todo su sentido. El documento de hipertexto resultante, denominado también "wiki" o "WikiWikiWeb", lo produce típicamente una comunidad de usuarios. Muchos de estos lugares son inmediatamente identificables por su particular uso de palabras en mayúsculas, o texto capitalizado que consiste en poner en mayúsculas las letras iniciales de las palabras de una frase y eliminar los espacios entre ellas como por ejemplo en EsteEsUnEjemplo. Esto convierte automáticamente a la frase en un enlace. Este wiki, en sus orígenes, se comportaba de esa manera, pero actualmente se respetan los espacios y sólo hace falta encerrar el título del enlace entre dos corchetes.

## Características

- Un wiki permite que se escriban artículos colectivamente por medio de un lenguaje de wikitexto editado mediante un navegador.
- Una página wiki singular es llamada página wiki, mientras que el conjunto de páginas normalmente interconectadas mediante hipervínculos es "El Wiki". Es mucho más sencillo y fácil de usar que una base de datos.
- Facilidad con que las páginas pueden ser creadas y actualizadas. En general no hace falta revisión para que los cambios sean aceptados.
- La mayoría de wikis están abiertos al público sin la necesidad de registrar una cuenta de usuario. A veces se requiere hacer login para obtener una cookie de wiki-firma, para autofirmar las ediciones propias. Otros wikis más privados requieren autenticación de usuario.

En un wiki tradicional existen tres representaciones por cada página:

- El código fuente, que pueden editar los usuarios. Es el formato almacenado localmente en el servidor. Normalmente es texto plano, sólo es visible para el usuario cuando lo muestra la operación Editar.
- Una plantilla, en ocasiones generada internamente que define la disposición y elementos comunes de todas las páginas.
- El código HTML, puesto en tiempo real por el servidor a partir del código fuente cada vez que la página se solicita.

El código fuente es potenciado mediante un lenguaje de marcado simplificado para hacer varias convenciones visuales y estructurales. Por ejemplo, el uso del asterisco (\*) al empezar una línea de texto significa que se generará una lista desordenada de elementos o bullet - list. El estilo y la sintaxis pueden variar en función de la implementación, alguna de las cuales también permite etiquetas HTML.

La razón principal para no usar HTML en el diseño es que con muchas de sus etiquetas crípticas, no es fácil de leer por usuarios no técnicos. Hacer visibles las etiquetas de HTML provoca que el texto en sí sea difícil de leer y editar para la mayoría de usuarios. Por lo tanto, se promueve el uso de edición en texto llano con convenciones para la estructura y el estilo fáciles de comprender.

A veces es beneficioso que los usuarios no puedan usar ciertas funcionalidades que el HTML permite, tales como JavaScript, CSS y XML. Se consigue consistencia en la visualización, así como seguridad extra para el usuario. En muchas inserciones de wiki, un hipervínculo es exactamente tal como se muestra, al contrario de lo que ocurre en el HTML.

### **Estándar**

Durante años el estándar de facto fue la sintaxis del WikiWeb original. Actualmente las instrucciones de formato son diferentes dependiendo del motor del wiki. Los wikis simples permiten sólo formato de texto básico, mientras que otros más complejos tienen soporte para

cuadros, imágenes, fórmulas e incluso otros elementos más interactivos tales como encuestas y juegos. Debido a la dificultad de usar varias sintaxis, se están haciendo esfuerzos para definir un estándar de marcado.

### **Vincular y crear páginas**

Los wikis son un auténtico medio de hipertexto, con estructuras de navegación no lineal. Cada página contiene un gran número de vínculos a otras páginas. En grandes wikis existen las páginas de navegación jerárquica, normalmente como consecuencia del proceso de creación original, pero no es necesario usarlas. Los vínculos se usan con una sintaxis específica, el patrón de vínculos.

### **CamelCase**

Originalmente gran parte de wikis usaban CamelCase como patrón de vínculos, poniendo frases sin espacios y poniendo la primera letra de cada palabra en mayúscula, por ejemplo, la palabra "CamelCase". Este método es muy fácil, pero hace que los links se escriban de una manera que se desvía de la escritura estándar. Los wikis basados en CamelCase se distinguen instantáneamente por los links con nombres como: TablaDeContenidos, PreguntasFrecuentes. Por consiguiente, comenzaron a desarrollarse otras soluciones.

### **Vínculos libres**

Los vínculos libres, usados por primera vez por Cliko, usan un formato tipo `_vínculo`. Por ejemplo, `_Tabla de contenidos`,



\_ Preguntas frecuentes. Otros motores de wiki usan distintos signos de puntuación.

### **Interwiki**

Interwiki permite vínculos entre distintas comunidades wiki. Las nuevas páginas se crean simplemente creando un vínculo apropiado. Si el vínculo no existe, se acostumbra a remarcar como vínculo roto. Siguiendo el vínculo se abre una página de edición, que permite al usuario introducir el texto para la nueva página wiki. Este mecanismo asegura que casi no se generen páginas huérfanas, es decir, páginas que no tienen ningún vínculo apuntando a ellas. Además se mantiene un nivel alto de conectividad.

### **Búsqueda**

La mayoría de wikis permite al menos una búsqueda por títulos, a veces incluso una búsqueda por texto completo. La escalabilidad de la búsqueda depende totalmente del hecho de que el motor del wiki disponga de una base de datos o no. Es necesario el acceso a una base de datos indexada para hacer búsquedas rápidas en wikis grandes. En Wikipedia el botón "Ir" permite a los lectores ir directamente a una página que concuerde con los criterios de búsqueda. El motor de MetaWiki se creó para habilitar búsquedas en múltiples wikis.

### **Control de cambios**

Los wikis suelen diseñarse con la filosofía de aumentar la facilidad de corrección de los errores, y no la de reducir la dificultad de cometerlos. Los wikis son muy abiertos, pero aun

así proporcionan maneras de verificar la validez de los últimos cambios al contenido de las páginas. En casi todos los wikis hay una página específica, Cambios recientes, que enumera las ediciones más recientes de artículos, o una lista con los cambios hechos durante un período. Algunos wikis pueden filtrar la lista para deshacer cambios hechos por vandalismo.

Desde el registro de cambios suele haber otras funciones, el Historial de revisión muestra versiones anteriores de la página, y la característica "diff" destaca los cambios entre dos revisiones. Usando el historial, un editor puede ver y restaurar una versión anterior del artículo, y la característica "diff" se puede usar para decidir cuándo eso es necesario. Un usuario normal del wiki puede ver el diff de una edición listada en Cambios recientes y, si es una edición inaceptable, consultar el historial y restaurar una versión anterior. Este proceso es más o menos complicado, según el software que use el wiki.

En caso de que las ediciones inaceptables se pasen por alto en Cambios recientes, algunos motores de wiki proporcionan control de contenido adicional. Se pueden monitorizar para asegurar que una página o un conjunto de páginas mantienen la calidad. A un usuario dispuesto a mantener esas páginas se le avisará en caso de modificaciones, y así se le permitirá verificar rápidamente la validez de las nuevas ediciones.

### **Vandalismo**

Consiste en hacer ediciones generalmente hechas por desconocidos que borran contenido importante, introducen errores, agregan contenido inapropiado u ofensivo tal como insultos o simplemente incumplen flagrantemente las normas

del wiki. También son frecuentes los intentos de spam, por ejemplo:

- La introducción de enlaces en un wiki con el fin de subir en los buscadores de Internet.
- Los intentos de publicitarse o hacer proselitismo de su ideología, religión u otros a través del wiki.
- Ingresar material que viola derechos de autor.

Algunas soluciones que se utilizan para luchar contra el vandalismo son:

- Revertir rápidamente sus cambios, para que así se desanimen.
- Bloquearlos temporalmente por su nombre de usuario o dirección IP, de tal forma que no puedan seguir editando. Esta solución se ve dificultada por las IPs dinámicas y el uso de proxies abiertos, que, al ser bloqueados, pueden afectar también a personas inocentes.
- Si se produce siempre en una misma página, la protección de esa página.
- No permitir que editen páginas usuarios que no estén registrados en la wiki.
- En casos extremos como ataques por medio de herramientas automáticas, bloquear la base de datos del wiki, sin permitir ningún tipo de edición.

#### **1.4.7. Blogging (Weblog).**

Una de las características más altamente pregonadas de la era de la Web 2.0 es el auge del blogging. En su aspecto más básico, un blog es sencillamente una “**home page**” personal

en formato de diario. Pero tal y como apunta Rich Skrenta <sup>7</sup>, la organización cronológica de un blog parece una diferencia trivial, pero conduce a una cadena de reparto, de publicidad y de valor completamente distinta.

También conocido como Bitácora, es un sitio web periódicamente actualizado que recopila cronológicamente textos o artículos de uno o varios autores, apareciendo primero el más reciente, donde el autor conserva siempre la libertad de dejar publicado lo que crea pertinente. El nombre bitácora está basado en los cuadernos de bitácora, cuadernos de viaje que se utilizaban en los barcos para relatar el desarrollo del viaje y que se guardaban en la bitácora. Aunque el nombre se ha popularizado en los últimos años a raíz de su utilización en diferentes ámbitos, el cuaderno de trabajo o bitácora ha sido utilizado desde siempre.

Este término inglés blog o weblog proviene de las palabras web y log ('log' en inglés = diario). El término bitácora, en referencia a los antiguos cuadernos de bitácora de los barcos, se utiliza preferentemente cuando el autor escribe sobre su vida propia como si fuese un diario, pero publicado en la web (en línea).

Habitualmente, en cada artículo de un blog, los lectores pueden escribir sus comentarios y el autor darles respuesta, de forma que es posible establecer un diálogo. No obstante es necesario precisar que ésta es una opción que depende de la decisión que tome al respecto el autor del blog, pues las herramientas permiten diseñar blogs en los cuales no todos los internautas o incluso ninguno puedan participar agregando

---

<sup>7</sup> Es un programador de computadoras y un empresario en Silicon Valley

comentarios. El uso o tema de cada blog es particular, los hay de tipo: periodístico, empresarial o corporativo, tecnológico, educativo o edublogs, políticos, personales con contenidos de todo tipo, etc.

## **Historia**

Para contar la historia de cómo fue evolucionando el Blogging, se la puede separar en periodos, cada uno con características específicas.

### **1994-2000**

El blog moderno es una evolución de los diarios en línea, donde la gente escribía sobre su vida personal, como si fuesen un diario íntimo pero en red. Las páginas abiertas Webring incluían a miembros de la comunidad de diarios en línea. Justin Hall, quien escribió desde 1994 su blog personal, mientras era estudiante de la Universidad de Swarthmore, es reconocido generalmente como uno de los primeros blogueros.

También había otras formas de diarios en línea. Un ejemplo era el diario del programador de juegos John Carmack, publicado mediante el protocolo Finger. Los sitios web, como los sitios corporativos y las páginas web personales, tenían y todavía tienen a menudo secciones sobre noticias o novedades, frecuentemente en la página principal y clasificados por fecha. Uno de los primeros precursores de un blog fue el sitio web personal de Kibo, actualizado mediante USENET.

Los primeros blogs eran simplemente componentes actualizados de sitios web comunes. Sin embargo, la

evolución de las herramientas que facilitaban la producción y mantenimiento de artículos web publicados y ordenados de forma cronológica, hizo que el proceso de publicación pudiera dirigirse hacia muchas más personas, y no necesariamente a aquellos que tuvieran conocimientos técnicos. Últimamente, esto ha llevado a que en la actualidad existan diversos procedimientos para publicar blogs. Por ejemplo, el uso de algún tipo de software basado en navegador, es hoy en día un aspecto común del blogging. Los blogs pueden ser construidos y almacenados usando servicios de alojamiento de blogs dedicados, o pueden ser concretados y accedidos mediante software genérico para blogs, como por ejemplo usando los productos Blogger o LiveJournal, o mediante servicios de alojamiento web corrientes.

El término "**w e b l o g**" fue acuñado por Jorn Barger el 17 de diciembre de 1997. La forma corta, "**b l o g**", fue acuñada por Peter Merholz, quien dividió la palabra weblog en la frase "**w e b l o g**" en la barra lateral de su blog Peterme.com en abril o mayo de 1999. Rápidamente fue adoptado tanto como nombre y verbo asumiendo "bloguear" como "editar el weblog de alguien o añadir un mensaje en el weblog de alguien".

Tras un comienzo lento, los blogs ganaron popularidad rápidamente; el sitio Xanga, lanzado en 1996, sólo tenía 100 diarios en 1997, pero más de 50.000.000 en diciembre de 2005. El uso de blogs se difundió durante 1999 y los siguientes años, siendo muy popularizado durante la llegada casi simultánea de las primeras herramientas de alojamiento de blogs:

- Open Diary lanzado en octubre de 1998, pronto creció hasta miles de diarios online. Innovó en los

comentarios de los lectores, siendo el primer blog comunitario donde los lectores podían añadir comentarios a las entradas de los blogs.

- Brad Fitzpatrick comenzó LiveJournal en marzo de 1999.
- Andrew Smales creó Pitas.com en julio de 1999 como una alternativa más fácil para mantener una página de noticias en un sitio web, seguido de Diaryland en septiembre de 1999, centrándose más en la comunidad de diarios personales.
- Evan Williams y Meg Hourihan lanzaron Blogger.com en agosto de 1999, y que fue adquirido por Google en febrero de 2003

El blogging combinaba la página web personal con herramientas para poder enlazar con otras páginas más fácilmente, en concreto permalinks, blogrolls y trackbacks. Esto, junto a los motores de búsqueda de weblogs permitió a los blogueros llevar un seguimiento de los hilos que les conectaban con otros usuarios con intereses similares.

#### **2000-2006**

Los primeros blogs estadounidenses populares aparecieron en 2001, tales como AndrewSullivan.com de Andrew Sullivan, Politics1.com de Ron Gunzburger, Political Wire de Taegan Goddardy, MyDD de Jerome Armstrong que trataban principalmente temas políticos.

En 2002, el blogging se había convertido en tal fenómeno que comenzaron a aparecer manuales, centrándose principalmente en la técnica. La importancia de la comunidad de blogs y su relación con una sociedad más grande cobró

importancia rápidamente. Las escuelas de periodismo comenzaron a investigar el fenómeno de los blogs, y a establecer diferencias entre el periodismo y el blogging.

En 2002, el amigo de Jerome Armstrong y ex-socio Markos Moulitsas Zúniga comenzó DailyKos. Con picos de hasta un millón de visitas diarias, ese espacio web se ha convertido en uno de los blogs con más tráfico de Internet.

En el año 2006, se escogió la fecha del 31 de agosto, para celebrar en toda la red, el llamado "día internacional del Blog". La idea nació de un bloguero o usuario de blog israelí llamado Nir Ofir, que propuso que en esta fecha, los blogueros que desarrollaban bitácoras personales enviaran cinco invitaciones de cinco blogs de diferentes temáticas a cinco diferentes contactos, para que así los internautas dieran difusión a blogs que seguramente otras personas desconocían, y que eventualmente les pudiera resultar interesantes.

#### **2007-Presente**

Hoy en día el blogging es uno de los servicios más populares en el Internet, llegando hasta el punto que cantantes o actores famosos tienen blogs, así también como empresas Internacionales inclusive.

Entre los servidores de blogs más populares se encuentran Blogger y Wordpress.

Algunos blogueros se han ido trasladando a la televisión y a los medios de prensa: Duncan Black, Markos Moulitsas Zúniga, Alex Steffen, Ana Marie Cox, Ezra Klein que se



integró primeramente al blog American Prospect, después a The Washington Post.

### **Herramientas para su creación y mantenimiento**

Existen variadas herramientas de mantenimiento de blogs que permiten, muchas de ellas gratuitamente y sin necesidad de elevados conocimientos técnicos, administrar todo el weblog, coordinar, borrar, o reescribir los artículos, moderar los comentarios de los lectores, etc., de una forma casi tan sencilla como administrar el correo electrónico. Actualmente su modo de uso se ha simplificado a tal punto, que casi cualquier usuario es capaz de crear y administrar un blog personal.

Las herramientas de mantenimiento de weblogs se clasifican, principalmente, en dos tipos: aquellas que ofrecen una solución completa de alojamiento, gratuita como Freewebs, Blogger y LiveJournal, y aquellas soluciones consistentes en software que, al ser instalado en un sitio web, permiten crear, editar, y administrar un blog, directamente en el servidor que aloja el sitio como es el caso de WordPress o de Movable Type. Este software es una variante de las herramientas llamadas Sistemas de Gestión de Contenido (CMS), y muchos son gratuitos. La mezcla de los dos tipos es la solución planteada por la versión multiusuario de WordPress (WordPress MU) a partir de la cual se pueden crear plataformas como Rebuscando.info, Wordpress.com o CyberBlog.es o Blog total.

Las herramientas que proporcionan alojamiento gratuito asignan al usuario una dirección web por ejemplo, en el caso de Blogger, la dirección asignada termina en "blogspot.com", y le proveen de una interfaz, a través de la cual se puede

añadir y editar contenido. Obviamente, la funcionalidad de un blog creado con una de estas herramientas, se limita a lo que pueda ofrecer el proveedor del servicio, o hosting.

Un software que gestione el contenido, en tanto, requiere necesariamente de un servidor propio para ser instalado, del modo en que se hace en un sitio web tradicional. Su gran ventaja es que permite control total sobre la funcionalidad que ofrecerá el blog, posibilitando así adaptarlo totalmente a las necesidades del sitio, e incluso combinarlo con otros tipos de contenido.

#### **1.4.8. (Really Simple Syndication) RSS Feeds.**

RSS son las siglas de Really Simple Syndication, un formato XML conforme especificaciones publicadas por el World Wide Web Consortium (W3C) para syndicar o compartir contenido en la web. Se utiliza para difundir información actualizada frecuentemente a usuarios que se han suscrito a la fuente de contenidos. El formato permite distribuir contenidos sin necesidad de un navegador, utilizando un software diseñado para leer estos contenidos RSS. A pesar de eso, es posible utilizar el mismo navegador para ver los contenidos RSS. Las últimas versiones de los principales navegadores permiten leer los RSS sin necesidad de software adicional. RSS es parte de la familia de los formatos XML desarrollado específicamente para todo tipo de sitios que se actualicen con frecuencia y por medio del cual se puede compartir la información y usarla en otros sitios web o programas. A esto se le conoce como redifusión web o sindicación web, una traducción incorrecta, pero de uso muy común.

To syndicate literalmente significa syndicar o formar parte de un sindicato. En inglés tiene otro significado: publicar artículos simultáneamente en diferentes medios a través de una fuente a la que pertenece.

Es el avance más significativo de la arquitectura básica de la web desde que los primeros hackers se dieron cuenta de que el CGI se podía utilizar para crear sitios web basados en bases de datos. RSS permite que alguien no sólo enlace con una página, sino suscribirse a la misma, con notificaciones cada vez que la página cambia. Skrenta denomina a esto como la web incremental. Otros lo llaman la web viva.

Es una forma de facilitar contenidos desde cualquier sitio en la red para su inserción fácil en una página web o en un lector de tu escritorio. El RSS es un paso más muy importante en la interconexión de la información y su acceso por los usuarios. En cualquier página web pueden ser vistos los titulares actualizados de The New York Times, BBC, Yahoo, Rolling Stone o de un blog. Ejemplo Noticias, información, artículos, novedades, entretenimiento sin tener que navegar por cada sitio y buscar sus actualizaciones. Con RSS puedes apuntarte a aquellas noticias y novedades con un reader o lector el cual automáticamente chequea todos los sitios que has indicado y va mostrando los nuevos contenidos que son añadidos.

La sindicación es un paso más en la actualización de contenidos superando a las listas de correo, un acceso a la información sin la navegación, permitiendo que en el escritorio de su ordenador tenga actualizada toda la información que se seleccione como por ejemplo los titulares de las noticias.

Constituyen una familia de documentos tipo para actualizar las novedades y noticias de un sitio web. Los documentos

generalmente llamados "RSS feeds" son leídos mediante lectores RSS readers denominados agregadores o aggregators, aunque recientemente se ha anticipado que las funciones de los agregadores estarán incorporadas en los browsers de los navegadores web.

Generalmente es utilizado para publicar los titulares de noticias, las entradas de los blog y otras informaciones. Nuevos usos están siendo encontrados continuamente, especialmente en el mundo de la educación. Se habla de una segunda era de Internet, tras el desarrollo de redes hipervinculadas que abre potencialidades muy grandes en el acceso a la información.

#### **Ventajas de un sitio web con RSS**

- Mantener una web fresca y actualizada. Utilizando herramientas tales como FeedRoll, se pueden importar alimentación de noticias desde otros sitios y mostrarlas en una web actualizada permanentemente.
- Publicar tus páginas sean estas de noticias, artículos, entre otros como archivos RSS para su difusión. Publicando un feed se da un paso más en el acceso a los contenidos y en una forma más directa de acceder a la información.

#### **1.4.9. Escritorios virtuales.**

Es un término nuevo, introducido en la década de los 90, que describe el proceso de separación entre el escritorio, que engloba los datos y programas que utilizan los usuarios para trabajar, de la máquina física. El escritorio virtualizado es almacenado remotamente en un servidor central en lugar de

en el disco duro del ordenador personal. Esto significa que cuando los usuarios trabajan en su escritorio desde su portátil u ordenador personal, todos sus programas, aplicaciones, procesos y datos se almacenan y ejecutan centralmente, permitiendo a los usuarios acceder remotamente a sus escritorios desde cualquier dispositivo capaz de conectarse remotamente al escritorio, tales como un portátil, PC, Smartphone o cliente ligero.

La virtualización de escritorio es encapsular y entregar bien el acceso a la totalidad de un entorno de un sistema de información o del entorno en sí a un dispositivo remoto. Este dispositivo puede estar basado en una arquitectura de hardware completamente diferente a la utilizada por el entorno de escritorio proyectado. También puede tener como base un sistema operativo totalmente diferente.

También se la puede definir como el uso de máquinas virtuales para permitir que los múltiples usuarios de la red mantengan sus escritorios individuales en un único servidor u ordenador central. El equipo central puede estar localizado en una residencia, en la empresa o en un centro de datos. Los usuarios pueden estar geográficamente dispersos y estar conectados al equipo central mediante una red de área local (LAN), una red de área extensa (WAN) o mediante Internet.

La experiencia que tendrá el usuario está orientada para que sea idéntica a la de un PC estándar, pero desde un dispositivo de cliente ligero o similar, y desde la misma oficina o remotamente.

Muchas soluciones comerciales también agregan la capacidad de conmutar algunas sesiones de cliente entrantes usando un software de rompimiento de conexión en dirección a sistemas de escritorio compartido tradicionales como el Terminal Services de Microsoft o servidores de aplicaciones de Citrix, servidores Blade, o incluso computadores de escritorios físicos e individuales no usados.

Es complejo instalar y mantener equipos de trabajo separados, además los usuarios han disfrutado de una capacidad casi ilimitada tanto para instalar como para borrar software. Debido a esto, los departamentos corporativos de tecnología de información y los usuarios han usado Terminal Services o Presentation Server de Citrix para proporcionar un entorno de escritorio estable y simple, pudiendo los usuarios estar usando tanto un PC de escritorio como un cliente ligero.

Proporciona muchas de las ventajas de un servidor de terminales, si es configurado por los administradores de sistemas pueden proporcionar a los usuarios mucha más flexibilidad, por ejemplo, cada uno puede tener permitido instalar y configurar sus propias aplicaciones. Los usuarios también disponen de la capacidad de acceder a sus escritorios virtuales basados en servidor desde otras localizaciones.

Ofrece ventajas con respecto al modelo tradicional, en el que cada ordenador funciona como una unidad completa y autónoma con su propio sistema operativo, periféricos y aplicaciones. En general los gastos se reducen porque los recursos pueden ser compartidos y se asignan a los usuarios en función de sus necesidades. La integridad de la

información del usuario se mejora por que todos los datos se almacenan y se guardan copias de seguridad en el centro de datos. Los conflictos en el software se minimizan mediante la reducción del número total de los programas almacenados en los equipos.

A pesar de la distribución de los recursos, todos los usuarios pueden personalizar y modificar sus equipos de escritorio para satisfacer sus necesidades específicas. De esta manera, la virtualización de escritorio ofrece una mayor flexibilidad en comparación con la paradigma cliente / servidor.

#### **Limitaciones**

Las limitaciones que tiene la virtualización de escritorios son los riesgos de seguridad que se generan si la red no está bien gestionada, conlleva una pérdida de autonomía y privacidad del usuario, los desafíos en la creación y el mantenimiento de los controladores para las impresoras y otros periféricos, dificultad en la gestión de complejas aplicaciones como las multimedia y problemas de mantenimiento de direcciones de los usuarios de máquinas virtuales en correspondencia con los del centro de datos.

#### **Beneficios**

Al igual que cualquier otra tecnología, la virtualización de escritorios proporciona una serie de beneficios clave que hace que esta tecnología sea la elegida por un gran número de usuarios:

- **Aumenta la seguridad de los escritorios y disminuye los costes de soporte.-** Las organizaciones tienen una gran cantidad de problemas otorgando permisos a los usuarios de sus equipos informáticos. Resulta complicado otorgar los mínimos permisos posibles y ofrecer a los usuarios un entorno de trabajo flexible con todas las funcionalidades que requieren para realizar su trabajo, sobre todo a aquellos usuarios que viajan. La virtualización del escritorio posibilita centralizar los escritorios en servidores centrales y gestionar dichos escritorios individualmente de manera remota. Esto provee a los usuarios una experiencia de escritorio completa y permite al personal de IT gestionar los escritorios virtualmente en lugar de físicamente. Disminuye radicalmente el coste de mantenimiento de los escritorios ya que no permite a sus usuarios modificar nada del Sistema Operativo, principal causa de los problemas en las organizaciones.
  
- **Reduce los costes generales de Hardware.-** La virtualización de escritorio implica que usted no tiene que tener equipos de última tecnología, porque todas las aplicaciones del escritorio son ejecutadas centralmente en un servidor remoto. Esto significa que los equipos que los usuarios están utilizando para conectarse al servidor tienen un periodo de vida mayor. Estos costes pueden reducirse aún más si para acceder a los escritorios virtuales utilizamos clientes ligeros, que son mucho menos costosos que los PC tradicionales.



- **Asegurar la continuidad del negocio.-** Un estudio reciente indica que la gran mayoría de las organizaciones no tienen planes de continuidad del negocio actuales. El estudio se basa en una encuesta realizada a 100 administradores de IT de Pymes del Reino Unido, empresas con 30-500 empleados, llevada a cabo por ICM Investigación. La encuesta sugiere que ante la posibilidad de un fallo en el transporte público o desastre en el lugar de trabajo, muchas empresas no estarían en condiciones reabrir el negocio, mientras que los usuarios de la virtualización de escritorios serían capaz de ofrecer a sus usuarios una clave personal para acceder a sus escritorios desde cualquier ubicación.
- **Alternativa ecológica.-** Los escritorios virtuales almacenados en el servidor central son una alternativa ecológica a los PC tradicionales. Un estudio realizado en 2008 por The Climate Group indica que los PC consumen alrededor de tres veces más energía que el centro de datos. Uno de los mayores beneficios de la virtualización de servidores es el ahorro de energía que se produce cuando varios servidores a baja utilización ejecutando aplicaciones se combinan en una única pieza física de hardware.
- **Mejorar la seguridad de los datos.-** La virtualización de escritorio hace que todos los datos de los usuarios de los escritorios, y por lo tanto de las organizaciones, se almacenen centralmente en los servidores, absolutamente nada se almacena a nivel local. Por lo que si el empleado pierde o le roban el portátil, no se

pierden los datos y tampoco dichos datos pasan a manos peligrosas.

### **Ventajas**

Las principales ventajas son las siguientes:

- Implementación instantánea de nuevos escritorios.
- Prácticamente cero tiempo de inactividad en caso de fallos de hardware.
- Reducción significativa del coste de los nuevos despliegues.
- Sólida capacidad de gestión de la imagen del escritorio.
- El ciclo de actualización de los PC pasa a ampliarse de 2-3 años a 5-6 años o más.
- Escritorios existentes incluyen múltiples monitores, audio/video bidireccional, streaming de video, soporte a los puertos USB, etc.
- Capacidad de acceso a los usuarios de empresas a sus escritorios virtuales desde cualquier PC, incluida el PC de la casa del empleado.
- Recursos a medida de las necesidades del escritorio.
- Múltiples escritorios bajo demanda.

### **El escritorio como un servicio**

El escritorio como un servicio es la disposición de la virtualización de escritorio como un servicio; transformando los escritorios en servidor de nube o el muy conocido cloud service. Hay varias soluciones diferentes basadas en la

ejecución local o distribuida y otras basadas en la ejecución centralizada o remota del escritorio.

El escritorio del usuario se ejecuta en el servidor, que puede ser un PC normal, un servidor dedicado o incluso un dispositivo de hardware fabricado específicamente para realizar esa tarea. Las órdenes del usuario se transmiten al servidor a través de la red, y el servidor envía de vuelta los resultados de las acciones. Según el modelo utilizado, el servidor puede encargarse de ejecutar todas y cada una de las tareas que requiera el usuario, o puede delegar algunas de las tareas al ordenador que esté utilizando el usuario, a fin de reducir la carga de trabajo del servidor y el tráfico de la red.

El usuario puede tener un ordenador completo que haya sido configurado para depender total o parcialmente de las instrucciones del servidor. O puede tener un aparato llamado terminal que haya sido diseñado para esta tarea, al cual conectará un monitor, un teclado, y un ratón, y con los cuáles usará la terminal como si se tratase de un ordenador completo. Esta terminal sólo funcionará cuando tenga conexión con un servidor adecuado.

#### **1.5. Webs del futuro.**

Como partes de las webs del futuro, podemos tomar en cuenta conceptos como la Web 3.0. Es un neologismo que apareció por primera vez en 2006 en un artículo de Jeffrey Zeldman, crítico de la web 2.0 y asociado a tecnologías como Ajax. Se utiliza para describir la evolución del uso y la interacción en la red a través de diferentes caminos. Ello incluye, la transformación de la red en una base de datos, un movimiento a hacer los contenidos

accesibles por múltiples aplicaciones sin necesidad de navegador, el empuje de las tecnologías de inteligencia artificial, la web semántica, la web geoespacial, o la web 3D.

Frecuentemente es utilizado por el mercado para promocionar las mejoras respecto a la web 2.0. Actualmente existe un debate considerable en torno a lo que significa web 3.0, y cuál es la definición acertada. Las tecnologías de la Web 3.0, como programas inteligentes, que utilizan datos semánticos, se han implementado y usado a pequeña escala en compañías para conseguir una manipulación de datos más eficiente. En los últimos años, sin embargo, ha habido un mayor enfoque dirigido a trasladar estas tecnologías de inteligencia semántica al público general.

Otro término que se analizará, es el de Web Semántica cuyo precursor es Tim Berners-Lee, es una Web extendida, dotada de mayor significado en la que cualquier usuario en Internet podrá encontrar respuestas a sus preguntas de forma más rápida y sencilla gracias a una información mejor definida. Al dotar a la Web de más significado y, por lo tanto, de más semántica, se pueden obtener soluciones a problemas habituales en la búsqueda de información gracias a la utilización de una infraestructura común, mediante la cual, es posible compartir, procesar y transferir información de forma sencilla.

Esta Web extendida y basada en el significado, se apoya en lenguajes universales que resuelven los problemas ocasionados por una Web carente de semántica en la que, en ocasiones, el acceso a la información se convierte en una tarea difícil y frustrante.

Esas informaciones adicionales describen el contenido, el significado y la relación de los datos y se deben proporcionar de manera formal, para que así sea posible evaluarlas automáticamente por máquinas de procesamiento. El objetivo es mejorar Internet ampliando la interoperabilidad entre los sistemas informáticos usando agentes inteligentes. Agentes inteligentes son programas en las computadoras que buscan información sin operadores humanos.

## CAPÍTULO 2

### 2. RICH INTERNET APPLICATION (RIA)

#### 2.1. Introducción

Son aplicaciones web que se combinan con las características y funcionalidades de aplicaciones tradicionales de escritorio. Las RIA's típicamente transfieren el proceso necesario para la interfaz de usuarios a un cliente web, pero protegen los volúmenes de datos manteniendo el estado del programa o los registros en el servidor de aplicación. Utilizan un navegador web estandarizado para ejecutarse y por medio de "plugin" o independientemente una "virtual machine".

Consisten en el aprovechamiento de la experiencia del usuario en herramientas y funciones de escritorio tan naturales como copiar, cortar y pegar, redimensionar columnas, y ordenar etc., con el alcance y la flexibilidad de presentación y despliegue que ofrecen las aplicaciones o páginas Web junto con lo mejor de la multimedia voz, vídeo, etc.

Se puede decir que las RIA son la nueva generación de las aplicaciones y es una tendencia ya impuesta por empresas como Macromedia, Magic Software, Sun o Microsoft que se encuentran desarrollando recursos para hacer de este tipo de aplicaciones una realidad. Estas aplicaciones están basadas en plataformas J2EE o .NET, con un front-end Flash, Java swing o Silverlight y utilizan una arquitectura Cliente/Servidor asíncrona, segura y escalable, junto con una interfaz de usuario Web.

Entre los beneficios principales de aplicaciones RIA tenemos una mejora importante en la experiencia visual, que hacen del uso de la aplicación algo muy sencillo, ofrece mejoras en la conectividad y despliegue instantáneo de la aplicación, agilizando su acceso, garantizan la desvinculación de la capa de presentación es decir acceso a la aplicación desde cualquier computador en cualquier lugar del mundo.

Normalmente en las aplicaciones Web, hay una recarga continua de páginas cada vez que el usuario pulsa sobre un enlace. De esta forma se produce un tráfico muy alto entre el cliente y el servidor, llegando muchas veces, a recargar la misma página con un mínimo cambio. Otra de las desventajas de las tradicionales aplicaciones Web es la poca capacidad multimedia que posee. Para ver un vídeo es necesario usar un programa externo para su reproducción.

En los entornos RIA, en cambio, no se producen recargas de página, ya que desde el principio se carga toda la aplicación, y sólo se produce comunicación con el servidor cuando se necesitan datos externos como datos de una Base de Datos o de otros ficheros externos.

Comúnmente las RIA's:

- Corren en un navegador web.
- No requieren instalación de software.
- Corren localmente en un ambiente seguro llamado "sandbox".
- Sus clientes esperan interfaces potentes y útiles que ofrezcan la respuesta de las aplicaciones de escritorio tradicionales.

- Buscan mejorar la experiencia del usuario.

## 2.2. Características.

- **Accesibilidad.-** AJAX nativo en los navegadores web y es el único "RIA framework" que puede ser encontrado por los diferentes motores de búsqueda.
- **Comunicaciones avanzadas.-** Con servidores que soporten nuevas tecnologías se puede mejorar la experiencia del usuario al utilizar protocolos de red optimizados y entradas y salidas asíncronas. Se requiere de una conexión de banda ancha confiable.
- **Complejidad.-** Soluciones avanzadas puede ser más difíciles de diseñar, desarrollar, implementar y depurar que las aplicaciones web tradicionales
- **Consistencia.-** La interfaz de usuario y las experiencias pueden ser controlada por el Sistema Operativo, el monitoreo del rendimiento y diagnostico de errores puede ser difícil.
- **Instalación y mantenimiento.-** Se requiere de la instalación de plugin, virtual machine o sandbox, que generalmente es más rápida que la instalación de una aplicación tradicional y esta no se puede automatizar. Las actualizaciones son automáticas.
- **Offline.-** Puede ser soportada reteniendo el estado en la maquina cliente.
- **Seguridad.-** Se mejora la seguridad por medio de actualizaciones automáticas y sandbox. Pero las extensiones por sí mismas son sujetas a vulnerabilidades.
- **Rendimiento.-** Puede ser mejorado dependiendo de la aplicación y de las características de la red. Aplicaciones que pueden procesar localmente en el cliente evitan viajes



hacia el servidor lo que incrementa su rendimiento. Dándole más trabajo al cliente también se puede incrementar el rendimiento del servidor. Por el contrario los recursos necesarios hacen difícil que las aplicaciones puedan ejecutarse en dispositivos pequeños, móviles o embebidos.

- **Riqueza.**- Añadiendo características que no son nativas en los navegadores web como captura de video, contenido multimedia entre otros aspectos gráficos que destacan las aplicaciones RIA de las aplicaciones web tradicionales.
- La mayoría de tecnologías RIA se basan en el lenguaje de programación XML, tanto para las interfaces gráficas como para el intercambio de datos.

### 2.3. Objetivos.

- Hacer las cosas que no puede hacer HTML y otras tecnologías y hacerlo en una manera que es muy fácil e implementar para los desarrolladores.
- Dar la sensación al usuario de que está en el escritorio cuando en realidad está en la web,
- Llevar a los usuarios una experiencia de navegación superior al mismo tiempo que reduce la necesidad de utilizar componentes de terceras partes.
- Unificar en nuestras aplicaciones gráficos, multimedia, animaciones, enlace a datos y comunicaciones de un modo rico y orientado al usuario final dándole una gran relevancia al diseño de interfaces y a los conceptos de usabilidad.
- Incrementar la agilidad empresarial con una mayor satisfacción de los desarrolladores y aplicaciones nuevas o ampliadas de gran calidad.

#### 2.4. Beneficios.

Debido a que las RIA's emplean un cliente para interactuar con el usuario, ellas son:

- **Enriquecidas.-** Pueden ofrecer el funcionamiento de una interfaz de usuario no obtenible solo con elementos html disponibles para navegadores web basados en las aplicaciones web estándares. Estas funciones enriquecidas pueden incluir cualquier cosa que pueda ser implementada en la tecnología siendo usada del lado del cliente, incluyendo drag&drop, utilización de slider para cambiar datos, cálculos que requiere solo el cliente y no requieren ser enviados de vuelta al servidor, por ejemplo, una calculadora.
- **Mayor respuesta.-** El comportamiento de la interfaz presenta típicamente mejor respuesta que una aplicación web estándar de navegador web que debe siempre interactuar con un servidor remoto. Los más sofisticados ejemplos de RIA's exhiben un "look and feel" cercano al de un ambiente de escritorio.
- **Balance de cliente/servidor.-** La demanda del cliente y servidor está mejor balanceada, así que el servidor web no necesita estar siempre sobrecargado, lo que ocurre con las aplicaciones web tradicionales. Esto libera recursos del servidor, permitiendo que el mismo hardware de servidor pueda manejar mayor número de sesiones de clientes concurrentes.
- **Comunicación Asíncrona.-** El cliente puede interactuar con el servidor sin esperar que el usuario ejecute una acción como presionar un botón o un hipervínculo. Esto permite que a los usuarios ver e interactuar con la página asincrónamente para la

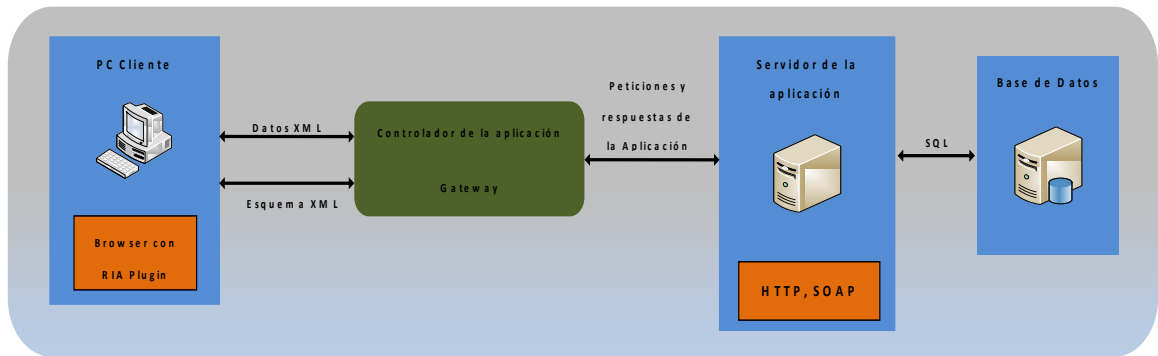
comunicación entre el cliente y el servidor. Esta opción permite a los diseñadores de RIA's mover datos entre el cliente y el servidor sin hacer esperar al usuario. Sin embargo la aplicación más común de esto es la precarga, en la cual una aplicación anticipa que el futuro necesitará de cierto dato, y lo descarga en el cliente antes de que el usuario lo solicite, de este modo acelera una respuesta subsecuente. Google Maps usa esta técnica para moverse a segmentos adyacentes de mapa antes que el cliente use el desplazamiento para verlos.

- **Eficiencia de la red.-** El tráfico de la red puede ser significativamente reducido porque un cliente de aplicación específico puede ser más inteligente que una aplicación Web estándar cuando decida qué datos necesita para ser intercambiados con los servidores. Esto puede acelerar las respuestas y solicitudes individuales, porque disminuye los datos que empezaran a transferirse.

## **2.5. Arquitectura de una aplicación RIA .**

La arquitectura de una RIA no dista mucho del esquema de una aplicación Web tradicional.

Primero tenemos el navegador Web, que contiene el plugin necesario para interpretar los datos que le llegan de la aplicación RIA, ya sea ficheros de video, textos, audio, etc. En esta parte de la arquitectura, a diferencia que la de las aplicaciones Web tradicionales, se almacena la parte gráfica de la aplicación que viene en un esquema XML. Con esto evitamos tener que renderizar una nueva página Web cada vez que queremos mostrar nuevos datos en pantalla.



**Figura 2.1 Arquitectura de una aplicación RIA**

En segundo lugar tenemos el controlador de la aplicación y el gateway. El controlador de la aplicación sería la parte de la aplicación que se encarga de interactuar con el cliente y viceversa. Luego esta parte de la aplicación utilizaría otras partes para realizar y ofrecer los servicios solicitados. La pasarela de información o gateway se encargaría de transformar los datos que maneja la aplicación a un formato de datos, en nuestro caso XML, que pudiera entender el navegador del cliente.

El siguiente elemento de la arquitectura de una RIA sería el servidor donde se alojaría la aplicación. Este servidor de aplicaciones compilará la aplicación Web en caso necesario y permitiría la comunicación de nuestra aplicación RIA con otras aplicaciones. Esta parte de la arquitectura se encargaría también de obtener los datos necesarios para la aplicación, ya sea de una base de datos o de ficheros.

Por último tenemos el almacenaje de los datos que habitualmente será una base de datos donde se almacenarán los datos relacionados con el servicio que está ofreciendo la aplicación. Esta parte de la arquitectura no siempre se requiere ya que puede haber aplicaciones que no lo necesiten.

## **2.6. Herramientas de desarrollo para RIA .**

Entre las herramientas para la creación de entornos RIA, se pueden mencionar las plataformas Flash y Flex de adobe, Ajax, Open Laszlo que es una herramienta open source, Silverlight de Microsoft, JavaFX script de Sun Microsystems y Bindows de mb technologies. Muchas de ellas ya se explicaron en el capítulo anterior, a excepción de Silverlight del cual se hablará en el siguiente capítulo

## CAPITULO 3

### 3. TECNOLOGÍA SILVERLIGHT 3

#### 3.1. Introducción.

Silverlight en general es parte de una generación de desarrollo de diseño espectacular en la web que ha llegado a nosotros desde ya hace algún tiempo, como competencia principal de Flash de Adobe. Es una nueva tecnología de presentación web que nos permite crear soluciones RIA para su ejecución en distintas plataformas es decir Multiplataforma, tales como Mac OS X, Windows y Linux GNU. En el caso de Linux particularmente, el componente que brinda Silverlight para este sistema operativo se llama Moolight que es creado y soportado por Novell y respaldado por Microsoft.

Hace posible un uso más completo, atractivo visualmente e interactivo y se puede ejecutar en todos los entornos como en los exploradores Firefox, Internet Explorer, Google Chrome, Opera, Safari y en múltiples dispositivos móviles.

Silverlight no es dependiente del Framework .NET ni del lado del cliente ni del lado del servidor. Del lado del cliente se puede utilizar Mac OS X, Windows, Linux. Así mismo del lado del servidor se puede utilizar Internet Information Service (IIS), Apache Tomcat o cualquier servidor web con cualquiera de las tecnologías web como son asp, php, jsp.

Al igual que WPF (Windows Presentation Foundation), la tecnología de presentación en Microsoft .NET Framework 3.0, la

infraestructura de programación de Windows, XAML (acrónimo pronunciado xammel del inglés eXtensible Application Markup Language, Lenguaje Extensible de Formato para Aplicaciones en español), constituye la base de la capacidad de presentación de Silverlight.

Silverlight ofrece un modelo de programación flexible que soporta AJAX, VB, C#, Python y Ruby, y se integra con las aplicaciones de Web existentes. Silverlight soporta la descarga rápida y eficiente de video de alta calidad a todos los principales navegadores que corren en MacOS o Windows. Tiene la ventaja de que no depende de otros productos como podrían ser el Windows Media Player para reproducir video ni de Microsoft .NET Framework 3.0 para el análisis de XAML. Ciertamente es que es una tecnología del lado del cliente, pero encaja en una estrategia mayor orientada a servidor, incluida la capacidad de usarse en servidores que ejecutan PHP o Java.

A Silverlight se lo puede interpretar desde 2 puntos de vista:

- **Del lado del cliente.-** Cuando los cibernautas se encuentran por primera vez con una página web que tiene contenido Silverlight, lo primero que nos pide el navegador web, sea cual fuere este, es descargarnos un plugin llamado Silverlight que se instala en el navegador web que se esté utilizando no importa cuál sea, Internet Explorer, Firefox, Google Chrome, Safari, entre otros. No por el hecho de ser producto de Microsoft no se puede visualizar en otros navegadores, al contrario Silverlight está pensado para ser multiplataforma y no solo abarcar hacia los productos de Microsoft. Muchos han llegado a pensar que es un software malicioso. Los que si se han atrevido a experimentar la descarga del plugin se han sorprendido

por el peso de este, alrededor de 4.8 MB para Windows y 8,7 MB para Mac OS X, es bastante pequeño en cuanto al tamaño de la descarga comparado con otros plugins

El resultado que se obtiene al visualizar una página que tenga contenido Silverlight es que ofrece una rica experiencia de usuario, así como contenido multimedia y creación de ricas aplicaciones interactivas y con grandes animaciones para la web. Todas las aplicaciones hechas con esta herramienta van a tener un cambio sumamente notable para los usuarios ya que se va a notar videos de muy buena calidad y sobre todo se podrán ver paginas con un diseño extremadamente increíble y atractivas para los usuarios así también se podrá visualizar muy buenas animaciones, por supuesto creadas con Silverlight.

Este complemento del navegador, representa XAML exponiendo su modelo de objetos del documento (DOM) interno y el modelo de evento al explorador de manera traducible en secuencias de comando. De esta manera, un diseñador puede reunir un documento XAML que contenga gráficos, animaciones y escalas de tiempo.

Este plugin no causa conflictos con otros plugins previamente instalados, sin que haya necesidad de desinstalarlos. Silverlight se integra fácilmente con otras aplicaciones

- **Del lado de los desarrolladores y diseñadores.-** Todos los elementos gráficos que se implementó mediante el lenguaje XAML, el desarrollador puede adjuntarlos al código de una página para implementar la funcionalidad. Dado que XAML se basa en XML, el documento que define



la interfaz de usuario que se descarga al cliente se basa en texto. Esto implica que no es un problema para los motores de búsqueda ni firewalls. Además, puede ensamblarse y emitirse en tiempo de ejecución mediante una aplicación de servidor, lo que no sólo ofrece una experiencia gráfica enriquecida, sino también una experiencia personalizable y dinámica.

Un diseñador puede reunir un documento XAML que contenga gráficos, animaciones y escalas de tiempo y, por otra parte, un desarrollador puede adjuntarlos al código de una página para implementar la funcionalidad.

Como Silverlight se basa principalmente en crear aplicaciones web con una buena presentación los diseñadores pueden ocupar Expression Blend que nos ayudara a crear animaciones o a realizar el diseño en muy pocos minutos y obteniendo resultados asombrosos todo esto simplemente para satisfacer a los usuarios que vayan a visitar la aplicación web y puedan ver el excelente trabajo que se realiza con Silverlight, además que con esto se va a lograr tener numerosas páginas con buena calidad para los usuarios.

Para los desarrolladores vamos a trabajar con Visual Studio que nos permite trabajar con Silverlight y que además es una buena herramienta de trabajo ya que el vs soporta Intellinsense para Silverlight así que es muy sencillo desarrollar en Silverlight.

Podemos mencionar que existe una buena interacción desarrolladores diseñadores, ambos pueden trabajar al

mismo tiempo para poder crear algún proyecto, como se consigue esto, pues muy sencillo tanto los programadores como los diseñadores van a tener algo en común al trabajar en un proyecto de Silverlight, al estar trabajando con el Visual Studio que es el que nos permite trabajar con Silverlight ya que el 2008 lo trae completamente nativo.

En la parte del diseño se trabaja con Expression Blend que desde nuestro proyecto de Silverlight en el Visual Studio podemos abrirlo en el Blend, por esta razón puede existir una buena integración entre diseñador y programador. De esta manera es como se logran estas aplicaciones impresionantes, todas las modificaciones que se estén haciendo en cuanto a la animación o diseño del proyecto en Blend se van a estar actualizando en Visual estudio mediante XAML.

Una de las características principales para los desarrolladores es que Silverlight puede ser ocupado con cualquier lenguaje de programación soportados por .NET así que por esta parte no se limita a los desarrolladores debido a que pueden elegir entre un buen número de lenguajes de programación que deseen ocupar con Silverlight tales como C#.NET y Visual Basic.NET. Además podemos utilizar todas las funcionalidades que brinda Framework 3.0 para el desarrollo de aplicaciones Silverlight, dando como resultado que el implementar un proyecto web de este tipo es relativamente sencillo.

## **X A M L**

XAML es el lenguaje de formato para la interfaz de usuario para la Base de Presentación de Windows (WPF por sus

siglas en inglés) y Silverlight en sus diferentes versiones, el cual es uno de los pilares de la interfaz de programación de aplicaciones .NET en su versión 3.0.

XAML es un lenguaje declarativo basado en XML, optimizado para describir gráficamente interfaces de usuarios visuales ricas desde el punto de vista gráfico, tales como las creadas por medio de Adobe Flash. XUL y UIML son otros ejemplos de lenguajes de interfaz basados en XML

En su uso típico, los archivos tipo XAML serían producidos por una herramienta de diseño visual, como Microsoft Visual Studio o Microsoft Expression. El XML resultante es interpretado en forma instantánea por un sub-sistema de despliegue de Windows Vista que reemplaza al GDI de las versiones anteriores de Windows. Los elementos de XAML se interconectan con objetos del Entorno Común de Ejecución para Lenguajes (CLR). Los atributos se conectan con propiedades o eventos de esos objetos.

XAML fue diseñado para soportar las clases y métodos de la plataforma de desarrollo .NET que tienen relación con la interacción con el usuario, en especial el despliegue en pantalla. El acrónimo XAML originalmente significaba Extensible Avalon Markup Language, Lenguaje Extensible de Formato de Avalon; habiendo sido Avalon el nombre clave original de la Base de Presentación de Windows, nombre que engloba a este grupo de clases de .NET.

### **3.2. Evolución.**

Silverlight anteriormente era conocido como WPF/E acrónimo de WPF everywhere (WPF donde sea), Silverlight está diseñado para ofrecer un amplio contenido enriquecido con el resto del

entorno de desarrollo Web, incluido ASP.NET. Fue creado para presentar la siguiente generación de experiencias en medios basados en .NET y aplicaciones altamente interactivas para la Web.

Cuando Tim Berners-Lee inventó la Web moderna en CERN, se pretendía que ésta sirviera como un sistema que hiciera posible que los documentos estáticos se almacenaran en un sistema basado en la red con vínculos al mismo. Con el paso de los años, se fueron incorporando innovaciones y el siguiente paso lógico fueron los documentos activos que se generan en el momento en que se solicitan, con información específica del momento o del usuario. Tecnologías como CGI activaron este aspecto. Más adelante, la capacidad para generar documentos en la Web se convirtió en algo primordial y la tecnología evolucionó de CGI, a Java, ASP y, posteriormente, a ASP.NET.

ASP.NET supuso un hito en la capacidad de los desarrolladores para desarrollar aplicaciones web de calidad con un paradigma de desarrollo de servidores y con las mejores herramientas de la línea de productos Visual Studio.

La experiencia del usuario resultó ser una gran barrera en las aplicaciones web, donde las restricciones técnicas evitaban que éstas pudieran ofrecer la misma variedad de uso que una aplicación cliente con datos locales.

El objeto XMLHttpRequest, que Microsoft lanzó como parte de Internet Explorer en 2000, se convirtió en la base de la tecnología asíncrona de JavaScript y XML (AJAX) que hizo posible que las aplicaciones web pudieran proporcionar una respuesta más dinámica a las entradas de los usuarios, mediante la actualización

de pequeñas porciones de una página web sin necesidad de volver a cargar totalmente los contenidos.

Las soluciones innovadoras que se generaron con AJAX, como los mapas regionales de Windows Live, llevaron a las aplicaciones web a dar un paso adelante en su capacidad para lograr un uso similar a las aplicaciones cliente.

Silverlight representa el paso siguiente en el desarrollo del potencial de riqueza en utilización que los desarrolladores y diseñadores de aplicaciones pueden presentar a sus clientes. Esto se consigue permitiendo a los diseñadores expresar su creatividad y guardar su trabajo en un formato que funcione directamente en la Web. En el pasado, los diseñadores creaban un sitio web y ofrecían una utilización a los usuarios a través de herramientas que proporcionaban resultados variados, pero el desarrollador tenía que solventar las restricciones de la plataforma web siendo capaz de ofrecer tales resultados.

En el modelo de Silverlight, los diseñadores pueden generar el tipo de utilización para el usuario que desean y expresarlo como XAML. Un desarrollador puede incorporar este XAML directamente en una página web mediante el tiempo de ejecución de Silverlight. De esta forma, ambos pueden trabajar más estrechamente que antes con el fin de proporcionar una utilización completa y variada al usuario.

### **3.3. Fundamentos de Silverlight**

Silverlight es una tecnología web y como tal necesita por lo menos una página html, aunque puede ser una aspx, php, json,

cgi, para hospedarse, por lo que no hay ningún tipo de dependencia de .NET por parte del servidor.

Un .xap es básicamente una aplicación de Silverlight compilada. El archivo es en realidad un archivo .zip que contiene todos los archivos necesarios para la aplicación. Para poder ver el contenido, es tan solo necesario cambiar la extensión del archivo de .xap a .zip y abrirlo como un archivo comprimido. Los archivos .xap van a contener como mínimo un ensamblado .dll que representa el compilado de la aplicación y un manifiesto (AppManifest.xaml) creado de una manera automática. Este manifiesto indica todas aquellas partes que constituyen la aplicación Silverlight, tales como cuales es el ensamblado, el punto de entrada, la clase en el ensamblado en el cual va a iniciar la aplicación, la versión mínima de Silverlight que debe tener instalado el cliente para poder ejecutar la aplicación. Todo este manifiesto está en lenguaje xml.

### **3.3.1. Características de Silverlight 3.**

- **Reducción de tamaño de los archivos XAP.**- El tamaño de los archivos XAP se reducirá en un 10% a 30%. Esto significa que podemos utilizar menos ancho de banda para nuestras aplicaciones Silverlight debido a su ratio de compresión más alto. Eso sí, siempre hay que recordar que una aplicación Silverlight debe ser lo más pequeña posible.
- **OOB (Out-Of Browser).**- Con esta característica podemos ejecutar nuestras aplicaciones de Silverlight 3 fuera de la ventana del Navegador, simulando una aplicación de escritorio con acceso directo en nuestro escritorio y un icono que la represente. No obstante, la aplicación sigue

ejecutándose en un Sandbox tal y como sucede cuando la ejecutamos dentro del navegador.

Esta característica es particularmente interesante ya que se nos abre un abanico grande de oportunidades al crear aplicaciones con esta plataforma. Aplicaciones RIA que incluso puedan detectar cuando están ejecutándose de manera desconectada para ofrecer a nuestros usuarios una experiencia de uso más potente. Tal vez ofreciendo una versión offline que guarde datos usando datos de un cache local. La creación de este tipo de aplicaciones es bastante sencilla ya que esta opción la establecemos vía el Manifiesto de la aplicación.

- **Aceleración vía GPU.**- Las aplicaciones Silverlight 3 podrán utilizar el GPU del equipo en donde estén ejecutándose, liberando así recursos del CPU. Las aplicaciones que hagan uso de gráficas sofisticadas o video detallado son las que disfrutarán más de esta característica.
- **Perspectivas 3D.**- Podemos poner objetos 2D en espacios 3D. Las animaciones que hagan uso de esta característica podrán ofrecer efectos visuales bastante interesantes tales como menús desplegados en 3D, rotación de elementos, entre otros. Lo que hace que esta característica sea bastante poderosa es el hecho que podemos establecer este tipo de perspectivas incluso en los controles, no solo en elementos.
- **Estilos BasedOn.**- Estilos basados en otros.
- **Estilos modificables.**- Podemos modificar en tiempo de ejecución el estilo de un elemento o control.

- **Enlace entre Elementos.-** Enlace o Data Binding entre controles. Esta característica está presente en WPF y ahora es una realidad en Silverlight.
- **SaveFileDialog.-** La caja de diálogo pareja de OpenFileDialog, con la cual podemos ofrecer al usuario un mecanismo de selección de una ruta y archivo para escritura. Esta característica es especialmente necesaria cuando queremos permitir a nuestros usuarios que guarden archivos como texto, imágenes, dentro de su equipo. Su funcionalidad es auto-explicable ya que es un control que hemos usado desde siempre en las aplicaciones Windows Forms.
- **Pixel Shaders.-** Los Shaders permiten cambiar un pixel a otro tal vez de un color a otro para hacer efectos de sobre posición de imágenes o videos.
- **Framework de Navegación.-** Un paso más para subsanar las aplicaciones Silverlight con respecto a los buscadores. Este framework permite que tengamos diferentes Url's en una misma aplicación de Silverlight, para que sean indexables. Además funciona con las características de "Atrás" y "Adelante" de los Navegadores.
- **Mensajería Local (Local Messaging).-** Comunicación entre diferentes plugins de Silverlight en la misma página, o entre diferentes tabs del Navegador, o incluso: entre diferentes Navegadores. Esta característica está basada en mensajes tipo string entre un plugin y otro con un modelo suscriptor-publicador.



- **Efectos.-** Con ellos podemos impactar el comportamiento visual pero no la funcionalidad de nuestras aplicaciones. Incluidos en Silverlight 3 están dos: DropShadow y Blur; aunque podremos desarrollar nuestros propios efectos por medio de Pixel Shaders.
- **RIA Services.-** Una plataforma que promete mucho si es orientada correctamente: Los RIA Services. Estos servicios ofrecen a las aplicaciones de Silverlight 3 un mecanismo sencillo para comunicación con fuentes de datos.

### 3.4. Microsoft .NET.

Es un framework o componente de Microsoft que puede ser añadido al sistema operativo Windows y hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permita un rápido desarrollo de aplicaciones.

.NET podría considerarse una respuesta de Microsoft al creciente mercado de los negocios en entornos Web, como competencia a la plataforma Java de Oracle Corporation y a los diversos framework de desarrollo web basados en PHP. Su propuesta es ofrecer una manera rápida y económica, a la vez que segura y robusta, de desarrollar aplicaciones o soluciones permitiendo una integración más rápida y ágil entre empresas y un acceso más simple y universal a todo tipo de información desde cualquier tipo de dispositivo.

Provee un extenso conjunto de soluciones predefinidas para necesidades generales de la programación de aplicaciones, y administra la ejecución de los programas escritos específicamente con la plataforma. Esta solución es el producto principal en la

oferta de Microsoft, y pretende ser utilizada por la mayoría de las aplicaciones creadas para la plataforma Windows. .NET Framework se incluye en Windows Server 2008, Windows Vista y Windows 7. De igual manera, la versión actual de dicho componente puede ser instalada en Windows XP, y en la familia de sistemas operativos Windows Server 2003. Una versión reducida de .NET Framework está disponible para la plataforma Windows Mobile, incluyendo teléfonos inteligentes.

### **Componentes**

Los principales componentes del marco de trabajo son:

- La Biblioteca de Clases Base o BCL
- El Entorno Común de Ejecución para Lenguajes o CLR por sus siglas en inglés.

#### **3.4.1. BCL.**

La Biblioteca de Clases Base (BCL por sus siglas en inglés) maneja la mayoría de las operaciones básicas que se encuentran involucradas en el desarrollo de aplicaciones. Es una librería estándar disponible para todos los lenguajes que usan .NET Framework. En otras palabras son un enorme conjunto de clases, más de 4000, que poseen una amplia funcionalidad y que nos servirán para desarrollar cualquier tipo de aplicación que nos propongamos. Adicionalmente, podemos desarrollar nuestras propias clases y con eso, podemos contribuir con nuestra experiencia a nuestros desarrollos.

.NET incluye el BCL para encapsular un gran número de funciones comunes, tales como archivo de lectura y escritura,

representación gráfica, la interacción de bases de datos y la manipulación de documentos XML, lo que hace el trabajo del programador más fácil. Las clases contenidas BCL permiten acceder a los servicios ofrecidos por el CLR y a las funcionalidades más frecuentemente usadas a la hora de escribir aplicaciones.

Es mucho más grande en su alcance que las bibliotecas estándar para la mayoría de otros lenguajes, incluyendo C++, y es comparable en el ámbito de aplicación a las bibliotecas estándar de Java. El BCL es a veces incorrectamente denominado como Framework Class Library (FCL), que es un super conjunto incluyendo el nombre de espacios de Microsoft. El BCL se actualiza con cada versión de .NET Framework.

Microsoft .NET Framework, no deja de ser por lo tanto en parte, un enorme repositorio de clases listas para usar desde que instalamos el entorno .NET. Dado la amplitud de la BCL, ha sido necesario organizar las clases en ella incluida en espacios de nombres que agrupen clases con funcionalidades similares. Un ejemplo de esto es el espacio de nombres System que nos proporciona tipos que son usados muy frecuentemente, como los tipos de datos básicos, entrada/salida en consola, excepciones, tablas, etc. A su vez el espacio de nombres System contiene otros espacios de nombres de uso frecuente.

A partir de las clases contenidas en el BCL, el programador puede crear nuevas clases que extiendan o usen su funcionalidad y se integren a la perfección con el resto de las clases de la BCL. Es tal la riqueza de servicios que ofrece

esta biblioteca de clases que pueden crearse lenguajes que carezcan de biblioteca de clases propia y sólo usen la BCL como por ejemplo C#; es por ello que cuando hablamos de una clase de C#, realmente estamos hablando de una clase de la BCL.

#### **3.4.2. CLR.**

El Common Language Runtime (Lenguaje común en tiempo de ejecución) es el componente de máquina virtual de la plataforma .NET Framework que proporciona un entorno en tiempo de ejecución que ejecuta el código y proporciona servicios que facilitan el proceso de desarrollo.

Los compiladores y las herramientas exponen la funcionalidad de Common Language Runtime y permiten escribir código con las ventajas que proporciona este entorno de ejecución administrado. El código desarrollado con un compilador de lenguaje orientado al tiempo de ejecución se denomina código administrado. Este código se beneficia de características como: la integración entre lenguajes, el control de excepciones entre lenguajes, la seguridad mejorada, la compatibilidad con la implementación y las versiones, un modelo simplificado de interacción y servicios de generación de perfiles y depuración.

Para permitir al motor en tiempo de ejecución proporcionar servicios al código administrado, los compiladores de lenguajes deben emitir metadatos que describen los tipos, los miembros y las referencias del código. Los metadatos se almacenan con el código es decir cada archivo ejecutable portable (PE) de Common Language Runtime cargable

contiene metadatos. El motor en tiempo de ejecución utiliza los metadatos para localizar y cargar clases, colocar instancias en memoria, resolver invocaciones a métodos, generar código nativo, exigir mecanismos de seguridad y establecer los límites del contexto en tiempo de ejecución.

En tiempo de ejecución controla automáticamente la disposición de los objetos y administra las referencias a éstos, liberándolos cuando ya no se utilizan. Los objetos cuya duración se administra de esta forma se denominan datos administrados. La recolección de elementos no utilizados elimina pérdidas de memoria así como otros errores habituales de programación. Con un código administrado se pueden utilizar datos administrados, datos no administrados o estos dos tipos de datos en una aplicación .NET Framework. Como los compiladores de lenguajes proporcionan sus propios tipos, como tipos primitivos, no siempre se sabe o no es necesario saber si los datos se están administrando.

Common Language Runtime facilita el diseño de los componentes y de las aplicaciones cuyos objetos interactúan entre lenguajes distintos. Los objetos escritos en lenguajes diferentes pueden comunicarse entre sí, lo que permite integrar sus comportamientos de forma precisa. Por ejemplo, puede definir una clase y, a continuación, utilizar un lenguaje diferente para derivar una clase de la clase original o llamar a un método de la clase original. También se puede pasar al método de una clase una instancia de una clase escrita en un lenguaje diferente. Esta integración entre lenguajes diferentes es posible porque los compiladores y las herramientas de lenguajes orientados al motor en tiempo de ejecución utilizan un sistema de tipos común definido por el motor en tiempo de

ejecución, y los lenguajes siguen las reglas en tiempo de ejecución para definir nuevos tipos, así como para crear, utilizar, almacenar y enlazar tipos.

Como parte de los metadatos, todos los componentes administrados contienen información sobre los componentes y los recursos utilizados en su compilación. El motor en tiempo de ejecución utiliza esta información para garantizar que el componente o la aplicación contienen las versiones especificadas de todo lo necesario, por lo que hay menos posibilidades de que la ejecución del código se interrumpa debido a una dependencia inadecuada. La información de registro y los datos de estado ya no se almacenan en el Registro, donde puede ser difícil establecer y mantener datos. En su lugar, la información sobre tipos definidos por el usuario y sus dependencias se almacena en el código como metadatos y, de este modo, las tareas de replicación y eliminación de componentes es mucho menos complicada.

Las herramientas y los compiladores de lenguajes exponen la funcionalidad del motor en tiempo de ejecución de forma que resulte útil e intuitiva para los programadores. Esto significa que algunas características en tiempo de ejecución pueden ser más evidentes en un entorno que en otro. El funcionamiento del motor en tiempo de ejecución depende de las herramientas y los compiladores utilizados. Por ejemplo, un programador de Visual Basic observará que con Common Language Runtime, el lenguaje Visual Basic contiene más características orientadas a objetos que antes.

El motor en tiempo de ejecución ofrece las siguientes ventajas:

- Mejoras en el rendimiento.
- Capacidad para utilizar fácilmente componentes desarrollados en otros lenguajes.
- Tipos extensibles que proporciona una biblioteca de clases.
- Características del lenguaje como herencia, interfaces y sobrecarga para la programación orientada a objetos.
- Compatibilidad con subprocesamiento libre explícito que permite la creación de aplicaciones multiproceso que sean escalables.
- Compatibilidad con el control de excepciones estructurado.
- Compatibilidad con atributos personalizados.
- Recolección de elementos no utilizados.
- Emplea delegados en lugar de punteros a funciones para mayor seguridad y protección de tipos.

### **3.5. Comunicaciones con WCF.**

Windows Communication Foundation (WCF) es un marco de trabajo para la creación de aplicaciones orientadas a servicios. Con WCF, es posible enviar datos como mensajes asincrónicos de un extremo de servicio a otro. Un extremo de servicio puede formar parte de un servicio disponible continuamente hospedado por IIS, o puede ser un servicio hospedado en una aplicación. Un extremo puede ser un cliente de un servicio que solicita datos de un extremo de otro servicio. Los mensajes pueden ser tan simples como un carácter o una palabra que se envía como XML, o tan complejos como una secuencia de datos binarios. A continuación se indican unos cuantos escenarios de ejemplo:

- Un servicio seguro para procesar transacciones comerciales.
- Un servicio que proporciona datos actualizados a otras personas, como un informe sobre tráfico u otro servicio de supervisión.
- Un servicio de chat que permite a dos personas comunicarse o intercambiar datos en tiempo real.
- Una aplicación de panel que sondea los datos de uno o varios servicios y los muestra en una presentación lógica.
- Exponer un flujo de trabajo implementado utilizando Windows Workflow Foundation como un servicio WCF.
- Una aplicación de Silverlight para sondear un servicio en busca de las fuentes de datos más recientes.

Si bien era posible crear tales aplicaciones antes de que existiera WCF, el desarrollo de extremos resulta más sencillo que nunca. En resumen, WCF se ha diseñado para ofrecer un enfoque manejable para la creación de servicios web y clientes de servicios web.

#### **Características de WCF**

**Orientación a servicios.-** Como consecuencia del uso de los estándares de Web Services, WCF le permite crear aplicaciones orientadas a servicios. SOA, la arquitectura orientada a servicios es el uso de servicios web para enviar y recibir datos. Los servicios tienen la ventaja general de estar débilmente acoplados entre una aplicación y otra en lugar de incluidos en el código. Una relación de acoplamiento débil implica que cualquier cliente creado en cualquier plataforma puede conectar con cualquier servicio siempre y cuando se cumplan los contratos esenciales.



**Interoperabilidad.-** WCF implementa estándares modernos de la industria para la interoperabilidad de servicios web.

**Varios modelos de mensajes.-** Los mensajes se intercambian mediante uno de los distintos modelos. El más común es el de solicitud/respuesta, en que un extremo solicita datos de otro extremo y el otro extremo responde. Existen otros modelos, como un mensaje unidireccional, en que un único extremo envía un mensaje sin esperar ninguna respuesta. Un modelo más complejo es el modelo de intercambio de mensajes dúplex, en donde dos extremos establecen una conexión y se envían datos entre sí, como ocurre con un programa de mensajería instantánea.

**Metadatos de servicios.-** WCF admite la publicación de metadatos de servicios utilizando los formatos especificados en los estándares de la industria, como WSDL, Esquemas XML y WS-Policy. Estos metadatos pueden utilizarse para generar y configurar automáticamente clientes para el acceso a los servicios de WCF. Los metadatos se pueden publicar sobre HTTP y HTTPS, o utilizando el estándar Intercambio de metadatos de servicios web.

**Contratos de datos.-** Dado que WCF se basa en .NET Framework, también incluye métodos con código sencillo para proporcionar los contratos que desea aplicar. Uno de los tipos de contrato universales es el contrato de datos. Básicamente, mientras se escribe el código del servicio utilizando Visual C# o Visual Basic, la forma más sencilla de controlar los datos consiste en crear clases que representan una entidad de datos con propiedades que pertenecen a la misma. WCF incluye un completo sistema para trabajar con los datos de esta manera tan sencilla. Cuando se han creado las clases que representan los

datos, el servicio genera automáticamente los metadatos que permiten a los clientes ajustarse a los tipos de datos que se han diseñado.

**Seguridad.-** Es posible cifrar los mensajes para proteger la privacidad, así como obligar a los usuarios a que se autentiquen antes de permitirles recibir mensajes. La seguridad puede implementarse utilizando estándares conocidos como SSL o WS-SecureConversation.

**Varios transportes y codificaciones.-** Los mensajes pueden enviarse con cualquiera de los protocolos y codificaciones integrados. La combinación más frecuente de protocolo y codificación consiste en enviar mensajes SOAP codificados de texto utilizando el Protocolo de transferencia de hipertexto (HTTP) usado en World Wide Web. WCF también le permite enviar mensajes sobre TCP, canalizaciones con nombre o MSMQ. Estos mensajes pueden codificarse como texto o utilizando un formato binario optimizado. Los datos binarios pueden enviarse de manera eficaz utilizando el estándar MTOM.

**Mensajes confiables y en cola.-** WCF admite el intercambio de mensajes confiable utilizando sesiones confiables implementadas sobre WS-Reliable Messaging y utilizando MSMQ.

**Mensajes duraderos.-** Un mensaje duradero es aquel que nunca se pierde debido a una interrupción de la comunicación. Los mensajes que forman parte de un modelo de mensajes duraderos siempre se guardan en una base de datos. Si se produce una interrupción, la base de datos le permite reanudar el intercambio de mensajes cuando se restablezca la conexión. También puede

crear un mensaje duradero utilizando Windows Workflow Foundation (WF).

**Transacciones.-** WCF también admite las transacciones que usan uno de los tres modelos de transacción: las transacciones WS-Atomic, las API del espacio de nombres System.Transactions y Microsoft DTC (Coordinador de transacciones distribuidas).

**Compatibilidad con AJAX y REST.-** REST es un ejemplo de una tecnología de web 2.0 en evolución. WCF se puede configurar para procesar datos XML sin formato que no se ajustan en un sobre SOAP. WCF también se puede extender para admitir formatos XML concretos, como ATOM un estándar popular de RSS e incluso formatos que no sean XML, como JavaScript Object Notation (JSON).

**Extensibilidad.-** La arquitectura de WCF tiene varios puntos de extensibilidad. Si se necesita una función adicional, existen una serie de puntos que le permiten personalizar el comportamiento de un servicio.

### **3.6. Herramientas de desarrollo.**

#### **3.6.1. Visual Studio 2008.**

Microsoft Visual Studio 2008, publicado en noviembre de 2007, mientras que la versión en castellano no fue publicada hasta el 2 de Febrero de 2008. El nuevo framework (.Net 3.5) está diseñado para aprovechar las ventajas que ofrece el nuevo sistema operativo Windows Vista a través de sus subsistemas Windows Communication Foundation (WCF) y Windows Presentation Foundation (WPF). El primero tiene como objetivo la construcción de aplicaciones orientadas a

servicios mientras que el último apunta a la creación de interfaces de usuario más dinámicas que las conocidas hasta el momento.

Es la primera aplicación en incluir un diseñador visual para Windows Presentation Foundation (WPF) con líneas de ajuste y tablas de eventos para el desarrollo en estilo RAD. También ayuda a los desarrolladores cuyo objetivo es el sistema de Microsoft Office 2007 con diseñadores para menús de Fluent y Outlook. Visual Studio Team System, diseñado para equipos multidisciplinares, ofrece una mejor escalabilidad e incluía soporte para una integración continua.

Asimismo, proporciona a los desarrolladores web las herramientas de formato CSS, la capacidad para codificar en ASP.NET AJAX y la compatibilidad con IntelliSense para JavaScript. La introducción de Language Integrated Query (LINQ) permite la manipulación y el procesamiento de datos avanzado en Visual Basic y C#. Para mejorar el rendimiento de las aplicaciones, Visual Studio 2008 ofrece herramientas de análisis de código, entre las que se incluyen métricas de código que pueden identificar áreas codificadas de una manera no eficiente o con otros problemas. Un sistema de compilación integrado también incluye la compatibilidad con multithreading tanto para la creación como para la depuración.

Visual Studio 2008 permite la creación de soluciones multiplataforma adaptadas para funcionar con las diferentes versiones de .Net Framework: Versión 2.0. Incluido con Visual Studio 2005, Versión 3.0 incluido en Windows Vista y la Versión 3.5 incluido con Visual Studio 2008.

## **Versiones**

**Visual Studio Professional.-** Ofrece un entorno de desarrollo completo para aplicaciones de Microsoft Office, Windows Mobile, Windows Server y basadas en web

**Visual Studio Team System .-** Proporciona un conjunto completo de herramientas para cada uno de los miembros de cualquier proyecto en equipo, incluidos arquitectos, diseñadores, desarrolladores y evaluadores.

Ofrece las siguientes ediciones cliente:

- Architecture Edition
- Database Edition
- Development Edition
- Test Edition

**Team Foundation Server.-** Es una plataforma de colaboraciones en grupo que combinaba un portal de equipos, control de versiones, seguimiento de elementos de trabajo, administración de compilaciones, guía de procesos y Business Intelligence en un servidor unificado. Permitía que todos los miembros del equipo colaboraran de forma más efectiva y crearan un software de calidad.

### **3.6.2. Microsoft Expression Blend 3.**

Expression Blend antes llamado Expression Interactive Designer, cuyo nombre en código era Sparkle, es una herramienta profesional desarrollado por Microsoft, de diseño que le permite controlar la eficacia del XAML, .NET y Silverlight para proporcionar experiencias de usuario

atractivas en escritorios conectados y Web, abriendo la puerta a la creación de diseños de interfaces de usuario mediante una amplia gama de tipos de medios, trabajando en un entorno de diseño en tiempo real. Su versión 3 fue lanzada el 10 de julio de 2009, esta incluye una herramienta muy útil, SketchFlow que revoluciona la velocidad con la que se puede prototipar la visión de una aplicación.

SketchFlow, una característica de Expression Studio Ultimate, permite presentar ideas, flujos de interacción, pantallas y funcionalidad de la aplicación a los clientes de forma rápida y con un prototipo efectivo. Transmite tus ideas de forma informal mediante dibujos y componentes esbozados que ayudarán al cliente a centrarse en la funcionalidad del prototipo más que en el detalle, mientras que el reproductor de SketchFlow le dará al cliente la oportunidad de probar el prototipo y devolver feedback o retroalimentación de su opinión, que puede ser visto directamente en contexto o en la superficie de diseño de Blend.

Los prototipos de SketchFlow son proyectos reales de WPF o Silverlight así que una vez que el diseño se ha terminado se debe que elegir. ¿Cuánto de tu prototipo quieres re usar? ¿Quieres tirarlo o usar elementos individuales del prototipo como la librería de recursos o Behaviors, o llevar directamente el prototipo hacia producción.

Los Behaviors son potentes bloques de construcción de interacción que traen un proyecto a la vida añadiendo control a la interacción, animación y transiciones de tus aplicaciones. Los desarrolladores pueden encapsular complejas interacciones e ideas en Behaviors, listos para ser reusados

en producción, que pueden ser refinados visualmente a través del visor de propiedades sin escribir código. GoToStateAction, ChangePropertyAction, MultiTouch, acciones condicionales, añadir física y MouseDrag son ejemplos perfectos de Behaviors que pueden ser compartidos por los equipos de diseño y desarrollo para asegurar la consistencia, reducir la duplicación de esfuerzos y permitir a los equipos maximizar su innovación.

Expression Blend incluye potentes importadores para Adobe Photoshop y Adobe Illustrator que permiten la sencilla integración con el flujo de trabajo que están usando los diseñadores visuales hoy. Durante la importación existe la libertad de ver e importar archivos de Photoshop capa a capa. Las capas pueden agruparse fácilmente y los elementos mantienen sus formatos originales; capas, posición de capas, texto editable y vectores.

## CAPITULO 4

### 4. CONSTRUCCIÓN DEL SISTEMA EN LÍNEA PARA LA COOPERATIVA “MACUCHI”

En el presente capítulo se mostrará la aplicación de la metodología XP en el desarrollo del sistema en línea para el seguimiento y administración de encomiendas, rutas, tarifas, boletos, vehículos, y funcionarios para la cooperativa de transportes “Macuchi”. Como parte de la metodología se ha creado historias de usuarios, plan de entregas o release planning de los módulos del sistema que deberán estar totalmente funcionales, el diagrama de clases, el modelo entidad relación. Así mismo se han identificado los riesgos, el glosario de términos especiales que no son del dominio general, es bosquejo del diseño preliminar que van a tener las pantallas que conforman el sistema, entre otros. El siguiente cuadro muestra las fases de las que consta la metodología Extreme Programming.

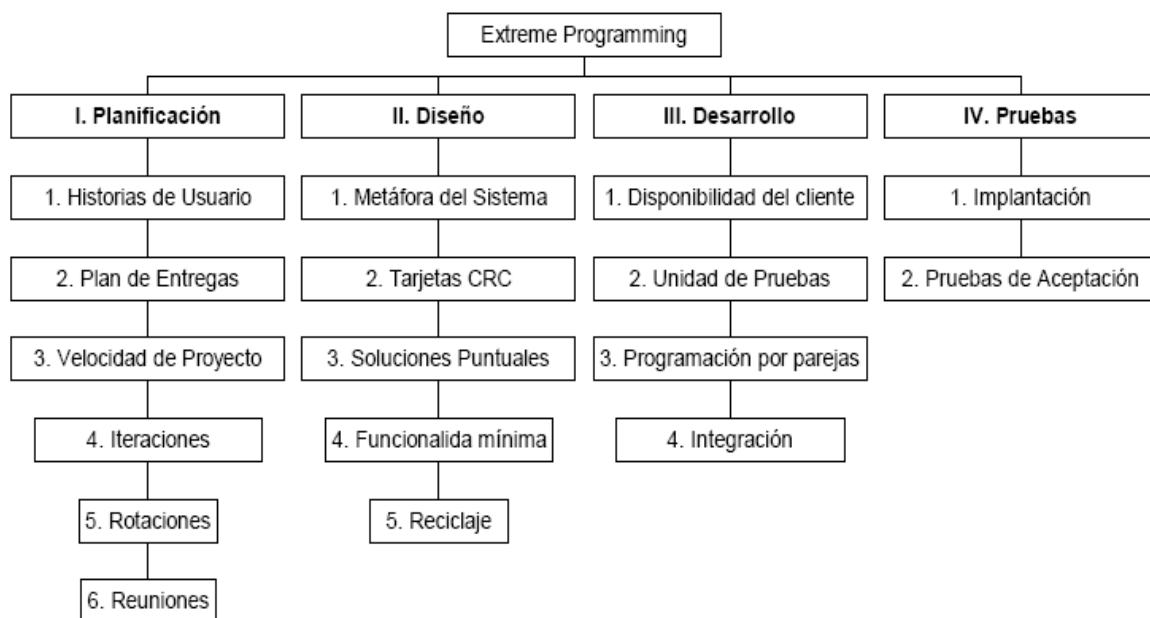


Figura 4.1 Fases de la metodología Extreme Programming.



#### 4.1. Planificación de Proyectos.

##### 4.1.1. Historias de Usuario.

Historia de Usuario No. 001 - Agregar una Ciudad			
<b>Nombre Historia:</b>	Agregar una Ciudad		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>El usuario va a registrar el nombre de la ciudad y si esta va a tener una oficina dentro del sistema, para motivos de facturación, se deberá ingresar una identificación que constará de 3 números. Ejemplo 001. El usuario ejecuta la acción de guardar entonces el sistema automáticamente guarda el registro y se muestra un listado con todas las ciudades y sus respectivos números de identificaciones.</p>			
<b>Notas:</b>			
-----			
Analista		Contraparte	

Historia de Usuario No. 002 - Listar Ciudades			
<b>Nombre Historia:</b>	Listar Ciudades		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción Ciudades. Como resultado de esta búsqueda va a obtener una lista de las Ciudades ya almacenadas en la base de datos.</p>			
<b>Notas:</b>			
-----			
Analista		Contraparte	

Historia de Usuario No. 003 - Consultar Ciudades			
<b>Nombre Historia:</b>	Consultar Ciudades		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	1 Horas
<b>Descripción:</b>			
El usuario selecciona una ciudad de la lista obtenida en la historia de usuario 2, e inmediatamente se le presentará la información asociada a la ciudad en modo de edición.			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <span>----- <b>Analista</b></span> <span>----- <b>Contraparte</b></span> </div>			

Historia de Usuario No. 004 - Modificar una Ciudad			
<b>Nombre Historia:</b>	Modificar una Ciudad		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
El usuario va a seleccionar una ciudad del listado mostrado, según historia de usuario número 3 y va a modificar los datos de esta. El usuario ejecuta la acción de guardar y el sistema automáticamente guarda el registro. El listado con todas las ciudades se va a actualizar.			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <span>----- <b>Analista</b></span> <span>----- <b>Contraparte</b></span> </div>			

Historia de Usuario No. 005 - Eliminar una Ciudad			
<b>Nombre Historia:</b>	Eliminar una Ciudad		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Hora
<b>Descripción:</b>			
<p>El usuario podrá eliminar una ciudad siempre y cuando no tenga datos relacionados. Una vez que se elimine la ciudad, se actualizará la lista de las ciudades.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="float: right;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 006 - Ingresar Rutas y tarifas			
<b>Nombre Historia:</b>	Ingresar Rutas y tarifas		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>El usuario deberá registrar en el sistema la ciudad de origen y de destino, la descripción, el costo de la tarifa normal y de la tarifa especial, la duración del viaje especificada como un texto (4 Horas) y se debe seleccionar si es una ruta directa. Esto se lo debe especificar si la ruta que se está creando es la ruta principal especificando la ciudad desde donde nace la ruta hasta la ciudad donde termina la ruta. El usuario ejecuta la acción de guardar y el sistema automáticamente guarda el registro.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="float: right;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 007 - Listar Rutas y Tarifas			
<b>Nombre Historia:</b>	Listar Rutas y Tarifas		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción Rutas y tarifas, si desea ingresa los filtros Ciudad origen, ciudad destino, descripción, definir si es una ruta directa (en el caso de no seleccionar esta opción, se filtran las rutas intermedias), el estado (todas, activas, inactivas).</p> <p>Como resultado de esta búsqueda va a obtener una lista de las rutas y tarifas ya almacenadas en la base de datos.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Analista</div> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Contraparte</div> </div>			

Historia de Usuario No. 008 - Consultar Rutas y Tarifas			
<b>Nombre Historia:</b>	Consultar Rutas y Tarifas		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario selecciona una ruta de la lista obtenida en la historia de usuario 7, e inmediatamente se le presentará la información asociada a la ruta en modo de edición.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Analista</div> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Contraparte</div> </div>			

Historia de Usuario No. 009 - Editar Rutas y Tarifas			
<b>Nombre Historia:</b>	Editar Rutas y Tarifas		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario después de consultar una Ruta, según la historia de usuario 8, procederá a modificar la información asociada a esta. No hay restricción respecto a los campos que puede modificar. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.</p>			
<b>Notas:</b>			
<p>-----</p> <p style="display: flex; justify-content: space-between;"><b>Analista</b> <b>Contraparte</b></p>			

Historia de Usuario No. 010 - Relacionar Rutas			
<b>Nombre Historia:</b>	Relacionar Rutas		
<b>Fecha:</b>	05/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>Esta opción nos permitirá relacionar las rutas directas con las rutas intermedias, para esto el usuario deberá escoger la opción Rutas y tarifas, seleccionar la opción Rutas relacionadas. Aquí deberá seleccionar la ruta directa y automáticamente se debe cargar las rutas intermedias relacionadas en el caso de tenerlas. Deberá seleccionar las rutas intermedias desde un listado e ir agregándolas a la lista de rutas intermedias. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.</p>			
<b>Notas:</b>			
<p>-----</p> <p style="display: flex; justify-content: space-between;"><b>Analista</b> <b>Contraparte</b></p>			

Historia de Usuario No. 011 - Ingresar oficinas			
<b>Nombre Historia:</b>	Ingresar oficinas		
<b>Fecha:</b>	06/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 horas
<b>Descripción:</b>			
<p>El usuario va a seleccionar la ciudad mostrándose automáticamente su número de identificación, un nombre o descripción de la oficina, la dirección, hasta 2 números de teléfono convencional, número de celular, número de Ruc para la oficina, el número de identificación de la oficina que así mismo va a servir para facturación de encomiendas, constará de 3 dígitos. Ejemplo: 001.</p> <p>Adicional a esto, debemos agregar las rutas relacionadas a esta oficina, esto para que no se puedan facturar encomiendas o boletos en rutas que no sean despachadas desde la oficina creada. El usuario ejecuta la acción de guardar y el sistema automáticamente guarda el registro.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 012 - Listar oficinas			
<b>Nombre Historia:</b>	Listar oficinas		
<b>Fecha:</b>	06/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción de listar oficinas, obteniendo una lista de las oficinas almacenadas en la base de datos, pudiendo filtrar por nombre y por estado (todos, activos, inactivos).</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 013 - Consultar una Oficina			
<b>Nombre Historia:</b>	Consultar una Oficina		
<b>Fecha:</b>	06/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
El usuario selecciona una oficina de la lista obtenida en la historia de usuario 12, e inmediatamente se le presentará la información asociada a la oficina seleccionada permitiendo modificar los datos.			
<b>Notas:</b>			
-----			
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 014 - Modificar Oficina			
<b>Nombre Historia:</b>	Modificar Oficina		
<b>Fecha:</b>	06/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
El usuario después de seleccionar una oficina según la historia de usuario 13, procederá a modificar la información asociada a esta. No hay restricción respecto a los campos que puede modificar. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.			
<b>Notas:</b>			
-----			
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 015 - Ingresar datos del Funcionario			
<b>Nombre Historia:</b>	Ingresar datos del Funcionarios		
<b>Fecha:</b>	12/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	10 Horas
<b>Descripción:</b>			
<p>El usuario va registrar el número de cédula, nombres, apellidos, dirección, número de teléfono convencional, número de teléfono celular, fecha de nacimiento, sexo, fecha de ingreso a la cooperativa "Macuchi", tipo de funcionario (Socio, chofer, oficial controlador, secretaria y despachador) y el estado del funcionario (Activo, Inactivo). El usuario ejecuta la acción de guardar y el sistema automáticamente guarda el registro.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 016 - Listar datos del Funcionario			
<b>Nombre Historia:</b>	Listar datos del Funcionario		
<b>Fecha:</b>	12/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción de listar funcionarios, obteniendo una lista de los funcionarios ya almacenadas en la base de datos, pudiendo filtrar por número de cédula, nombres, apellidos, sexo, fecha de ingreso, tipo de funcionario, estado del funcionario.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			



Historia de Usuario No. 017 - Consultar un Funcionario			
<b>Nombre Historia:</b>	Consultar un Funcionario		
<b>Fecha:</b>	12/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
El usuario selecciona un funcionario de la lista obtenida en la historia de usuario 16, e inmediatamente se le presentará la información asociada al funcionario seleccionado.			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Analista</div> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Contraparte</div> </div>			

Historia de Usuario No. 018 - Modificar un Funcionario			
<b>Nombre Historia:</b>	Modificar un Funcionario		
<b>Fecha:</b>	12/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
El usuario después de seleccionar un Funcionario, según la historia de usuario 17, procederá a modificar la información asociada a este. No hay restricción respecto a los campos que puede modificar. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Analista</div> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Contraparte</div> </div>			

Historia de Usuario No. 019 - Cambio de contraseña			
<b>Nombre Historia:</b>	Cambio de contraseña		
<b>Fecha:</b>	12/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>La primera vez que el usuario ingresa al sistema, debe cambiar la contraseña que fue proporcionada por el administrador del sistema, para esto deberá ingresar la contraseña actual, la nueva contraseña e ingresarla nuevamente para confirmar la nueva contraseña. El usuario ejecuta la acción de guardar y el sistema automáticamente guarda los cambios. Para el siguiente inicio de sesión, el usuario deberá ingresar con la nueva contraseña.</p>			
<b>Notas:</b>			
<p>-----</p> <p style="display: flex; justify-content: space-between;"><b>Analista</b> <b>Contraparte</b></p>			

Historia de Usuario No. 020 - Permisos a funcionarios			
<b>Nombre Historia:</b>	Permisos a funcionarios		
<b>Fecha:</b>	13/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	8 Horas
<b>Descripción:</b>			
<p>El usuario podrá administrar las pantallas que pueden ver o no los diferentes usuarios que tendrán acceso al sistema. Para esto se deberá seleccionar un funcionario y seleccionar las opciones del sistema a las cuales tendrá acceso. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.</p> <p>Los cambios se reflejarán en el siguiente inicio de sesión del usuario.</p>			
<b>Notas:</b>			
<p>-----</p> <p style="display: flex; justify-content: space-between;"><b>Analista</b> <b>Contraparte</b></p>			

Historia de Usuario No. 021 - Ingresar datos de los vehículos			
<b>Nombre Historia:</b>	Ingresar datos de los vehículos		
<b>Fecha:</b>	12/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	5 Horas
<b>Descripción:</b>			
<p>El usuario va registrar el socio al que pertenece, el número de disco (no debe repetirse), el número de asientos, la placa (no debe repetirse), el año de fabricación, la marca, el tipo, la clase, el modelo, el número de motor (no debe repetirse), el número de chasis (no debe repetirse), el estado (Activo, inactivo) y de disponer podrá agregar una imagen. El usuario ejecuta la acción de guardar y el sistema automáticamente guarda el registro.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 022 - Listar datos de los vehículos			
<b>Nombre Historia:</b>	Listar datos de los vehículos		
<b>Fecha:</b>	19/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción de listar vehículos, obteniendo una lista de los vehículos ya almacenados en la base de datos, pudiendo filtrar por el funcionario, marca del vehículo, estado (activo, inactivo).</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 023 - Consultar datos de un vehículo			
<b>Nombre Historia:</b>	Consultar datos de un vehículo		
<b>Fecha:</b>	19/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
El usuario selecciona un vehículo de la lista obtenida en la historia de usuario 22, e inmediatamente se le presentará la información asociada al vehículo seleccionado permitiendo modificar los datos.			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Analista</div> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Contraparte</div> </div>			

Historia de Usuario No. 024 - Modificar datos de un vehículo			
<b>Nombre Historia:</b>	Modificar datos de un vehículo		
<b>Fecha:</b>	19/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
El usuario después de seleccionar un Vehículo, según la historia de usuario 23, procederá a modificar la información asociada a este. No hay restricción respecto a los campos que puede modificar. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Analista</div> <div style="width: 45%; border-top: 1px dashed black; text-align: center;">Contraparte</div> </div>			

Historia de Usuario No. 025 - Ingreso de Turnos			
<b>Nombre Historia:</b>	Ingreso de Turnos		
<b>Fecha:</b>	20/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario deberá registrar en el sistema la ruta junto con la hora de salida. Ejecutar la acción de guardar y el sistema automáticamente guarda el registro.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="float: right;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 026 - Listar Turnos			
<b>Nombre Historia:</b>	Listar Turnos		
<b>Fecha:</b>	20/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción Listar Turnos, si desea selecciona la ruta de la cual desea obtener los turnos. Si no selecciona nada, muestra todos los turnos.</p> <p>Como resultado de esta búsqueda va a obtener una lista de los turnos ya almacenadas en la base de datos.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="float: right;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 027 - Consultar Turno			
<b>Nombre Historia:</b>	Consultar Turno		
<b>Fecha:</b>	20/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
El usuario selecciona un turno de la lista obtenida en la historia de usuario 26, e inmediatamente se le presentará la información asociada al turno en modo de edición.			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <span>----- Analista</span> <span>----- Contraparte</span> </div>			

Historia de Usuario No. 028 - Editar Turno			
<b>Nombre Historia:</b>	Editar Turno		
<b>Fecha:</b>	20/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
El usuario después de consultar un turno, según la historia de usuario 27, procederá a modificar la información asociada a esta. No hay restricción respecto a los campos que puede modificar. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <span>----- Analista</span> <span>----- Contraparte</span> </div>			

Historia de Usuario No. 029 - Crear Cuadro de Trabajo Semanal			
<b>Nombre Historia:</b>	Crear Cuadro de Trabajo Semanal		
<b>Fecha:</b>	26/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	16 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción Asignación de Turnos. El cuadro de trabajo va a ser válido para la semana en curso, por lo que se deberá calcular automáticamente la fecha de inicio que equivaldrá a la fecha del lunes de la semana en tránsito junto con la fecha final que equivaldrá al domingo de la misma semana.</p> <p>Además se van a listar todos los turnos previamente ingresados junto con las horas para cada uno de ellos. Como cabecera van a estar los días de la semana para los cuales se deben ingresar el número de unidad para cada turno en un formato de matriz. (Ver Anexo A).</p> <p>Una vez que se hayan ingresado el número de unidades para los respectivos turnos en los diferentes días, los datos se van a guardar en la base de datos.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: flex-end; margin-top: 50px;"> <div style="width: 45%; border-top: 1px dashed black; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; border-top: 1px dashed black; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 030 - Ingreso de vehículos que están de parada			
<b>Nombre Historia:</b>	Ingreso de vehículos que están de parada		
<b>Fecha:</b>	26/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	8 Horas
<b>Descripción:</b>			
<p>El usuario al mismo tiempo que va ingresando el cuadro de Trabajo semanal según historia de usuario 29, puede de manera similar ingresar el número de unidad que está de parada para un determinado día.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="float: right;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 031 - Buscar cuadro de trabajo semanal			
<b>Nombre Historia:</b>	Buscar cuadro de trabajo semanal		
<b>Fecha:</b>	27/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	5 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción Asignación de Turnos, para consultar un cuadro de trabajo semanal se debe seleccionar en la fecha inicial una fecha que sea en inicio de cualquier semana equivalente a lunes. La fecha final se calculará automáticamente 7 días después que equivaldrá al día domingo fin de semana. Un vez que se realice este proceso, se buscarán los datos registrados en base a estos parámetros y se presentaran en modo de edición o en modo de solo lectura dependiendo de la fecha. Esto es que permitirá editar los registros desde la fecha actual en adelante.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="float: right;"><b>Contraparte</b></span></p>			



Historia de Usuario No. 032 - Buscar vehículos que están de parada			
<b>Nombre Historia:</b>	Buscar vehículos que están de parada		
<b>Fecha:</b>	27/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>Al mismo tiempo que se ejecuta la historia de usuario 31, se va listar los vehículos que están de parada en base a los mismos filtros con los que se busca el cuadro de trabajo semanal. De la misma manera solo se podrán editar los registros desde la fecha actual en adelante.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="margin-left: 200px;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 033 - Editar cuadro de trabajo semanal			
<b>Nombre Historia:</b>	Editar cuadro de trabajo semanal		
<b>Fecha:</b>	27/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
<p>El usuario después de consultar el cuadro de trabajo semanal, según la historia de usuario 31, procederá a modificar la información asociada a esta. La única restricción es que el sistema solo permitirá editar los registros desde la fecha actual en adelante. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="margin-left: 200px;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 034 - Editar vehiculos que están de parada			
<b>Nombre Historia:</b>	Editar vehiculos que están de parada		
<b>Fecha:</b>	27/02/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Alta
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
<p>El usuario después de consultar los vehiculos que están de parada, según la historia de usuario 32, procederá a modificar la información asociada a esta. La única restricción es que el sistema solo permitirá editar los registros desde la fecha actual en adelante. El usuario ejecuta la acción de guardar y el sistema guarda los cambios.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; border-top: 1px dashed black; margin-top: 20px;"> <span>-----</span> <span>-----</span> </div> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <span><b>Analista</b></span> <span><b>Contraparte</b></span> </div>			

Historia de Usuario No. 035 - Ingreso de una autorización del SRI			
<b>Nombre Historia:</b>	Ingreso de un número de autorización del SRI		
<b>Fecha:</b>	12/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>El usuario deberá registrar en el sistema el número de autorización que genera el SRI tanto para facturar Encomiendas como para facturar boletos. Los datos que se deben registrar son: Autorización del SRI, fecha de caducidad, numeración inicial, numeración final, agregamos la o las oficinas para la cual aplica dicha autorización y finalmente se debe escoger si es para las encomiendas o para los boletos. El usuario ejecuta la acción de guardar y el sistema automáticamente guarda el registro.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; border-top: 1px dashed black; margin-top: 20px;"> <span>-----</span> <span>-----</span> </div> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <span><b>Analista</b></span> <span><b>Contraparte</b></span> </div>			

Historia de Usuario No. 036 - Listar autorizaciones del SRI			
<b>Nombre Historia:</b>	Listar de autorizaciones del SRI		
<b>Fecha:</b>	12/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción de Autorizaciones del SRI, si desea ingresa los filtros de Autorización del SRI, fecha de caducidad o la Oficina a la cual se le asignó un número de Autorización.</p> <p>Como resultado de esta búsqueda va a obtener una lista de las autorizaciones ya almacenados en la base de datos.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="float: right;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 037 - Consulta de una autorización del SRI			
<b>Nombre Historia:</b>	Consulta de una número de autorización del SRI		
<b>Fecha:</b>	12/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>El usuario selecciona un funcionario de la lista obtenida en la historia de usuario 36, e inmediatamente se le presentará la información asociada al número de autorización en modo de solo lectura.</p>			
<b>Notas:</b>			
<p style="text-align: center;">-----</p> <p style="text-align: center;"><b>Analista</b> <span style="float: right;"><b>Contraparte</b></span></p>			

Historia de Usuario No. 038 - Facturación de Encomiendas			
<b>Nombre Historia:</b>	Facturación de Encomiendas		
<b>Fecha:</b>	13/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	12 Horas
<b>Descripción:</b>			
<p>El usuario va registrar los nombres, apellidos, dirección y número de cédula de la persona que envía, el nombre de la persona que recibe, el valor declarado, la ciudad destino, la fecha de envío, el turno junto con la unidad en que va a viajar la encomienda, el contenido, Motivo de traslado y algún comentario de existir.</p> <p>Además de esto debe seleccionar si es flete pagado o flete al cobro, ingresar la cantidad de paquetes, bultos o sobres, junto con el valor unitario. Se va a calcular automáticamente el valor total. Como parte del control, el usuario va a ingresar la cantidad de dinero con la cual el cliente paga calculándose automáticamente el valor de vuelto.</p> <p>El número de Factura se va a generar automáticamente de acuerdo a la oficina en la que se encuentre. El estado al crear la factura va a ser Activa.</p> <p>El usuario ejecuta la acción de guardar, el sistema automáticamente guarda la información e imprime la factura correspondiente conforme al formato establecido.</p> <p>Ver Anexo B.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 039 - Listar las Facturas de las Encomiendas			
<b>Nombre Historia:</b>	Listar las Facturas de las Encomiendas		
<b>Fecha:</b>	19/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción de Facturación de encomiendas, si desea ingresa los filtros de número de factura, apellidos y nombres de la persona que envía la encomienda, nombre de la persona que recibela encomienda, estado del proceso de la factura (Todas, Recibidas, Viajando, En destino, Entregadas), si están Activas o Anuladas, por rango de fechas y si desea ver las encomiendas de la oficina a la que pertenece o de todas las oficinas.</p> <p>Como resultado de esta búsqueda va a obtener una lista de las encomiendas ya almacenadas en la base de datos.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 040 - Consultar una Factura de Encomienda			
<b>Nombre Historia:</b>	Consultar una Factura de Encomienda		
<b>Fecha:</b>	19/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
<p>El usuario selecciona una Encomienda de la lista obtenida en la historia de usuario 39, e inmediatamente se le presentará la información asociada a la encomienda seleccionada dejándonos como única opción el poder anular la factura de la encomienda siempre y cuando la encomienda no haya sido despachada y cuando la factura haya sido creada en la misma oficina a la cual pertenece el usuario. Caso contrario solo se puede la información mostrada es solo de lectura.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 041 - Anular una factura de encomienda			
<b>Nombre Historia:</b>	Anular una factura de encomienda.		
<b>Fecha:</b>	20/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario después de seleccionar una Encomienda según la historia de usuario 40, procederá a modificar la información asociada a esta siempre y cuando la encomienda no haya sido despachada y cuando la factura de la encomienda haya sido creada en la misma oficina a la cual pertenece el usuario. Si cumple estas 2 condiciones, el usuario podrá seleccionar la opción para Anular la factura, ingresar una descripción del motivo por el cual se anula la factura y presionará el botón anular.</p> <p>Después de esto, la información de dicha factura, quedará disponible solo para consulta y no será incluida para la elaboración de la hoja de ruta.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 042 - Generar Hoja de Ruta			
<b>Nombre Historia:</b>	Generar Hoja de Ruta		
<b>Fecha:</b>	26/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	6 Horas
<b>Descripción:</b>			
<p>El usuario debe escoger la ciudad de destino, la fecha de envío, el número de unidad y el turno. El sistema va a mostrar automáticamente la ciudad origen a la cual pertenece el usuario que va a estar logueado, después de seleccionar el número de unidad y el turno, se va a mostrar cual es el propietario del vehículo.</p> <p>El usuario deberá presionar el botón buscar para listar todas las facturas de las encomiendas que se registraron para el turno seleccionado. De existir podrá imprimir los datos mostrados, según el formato especificado. Ver Anexo C.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			



Historia de Usuario No. 043 - Registro de Encomiendas que Llegan			
<b>Nombre Historia:</b>	Registro de Encomiendas que Llegan		
<b>Fecha:</b>	26/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	8 Horas
<b>Descripción:</b>			
<p>El usuario deberá registrar las encomiendas que llegan en un determinado turno. Para esto deberá escoger la ciudad origen, la fecha de envío, el número de unidad y el turno en el cual llegaron las encomiendas registradas en el sistema.</p> <p>Una vez ingresados todos estos datos, se deberá mostrar un listado con las encomiendas registradas que van a viajar en ese turno. De ese listado el usuario deberá seleccionar las encomiendas que llegan y guardar en el sistema. En caso de que no se seleccione alguna encomienda por algún motivo, después el usuario podrá regresar, ingresar los datos, listar las encomiendas y registrar el ingreso de las encomiendas a la oficina.</p> <p>Después de este registro, las encomiendas estarán disponibles en el sistema para registrar las entregas.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: flex-end; margin-top: 50px;"> <div style="text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 044 - Entrega de Encomiendas			
<b>Nombre Historia:</b>	Entrega de Encomiendas		
<b>Fecha:</b>	27/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	16 Horas
<b>Descripción:</b>			
<p>El usuario deberá poder registrar la entrega de las encomiendas creadas en el sistema. Para esto, deberá buscar la encomienda en base el número de factura, los nombres, apellidos de la persona que envía, el nombre de la persona que recibe, el lugar desde donde le envían o la fecha de envío. En base a estos filtros, el sistema deberá mostrar una o varias facturas. Con este listado el usuario va a seleccionar una de ellas, y se va a cargar la información referente a la encomienda permitiendo ingresar el nombre y número de cédula de la persona que retira la encomienda.</p> <p>Una vez ingresado esta información, el sistema deberá registrar la entrega de la encomienda.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

<b>Historia de Usuario No. 045 - Ingresar Encomiendas de Oficinas sin Sistema</b>			
<b>Nombre Historia:</b>	Registro de Encomiendas que llegan de Oficinas sin Sistema		
<b>Fecha:</b>	27/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	8 Horas
<b>Descripción:</b>			
<p>El usuario deberá registrar en el sistema las encomiendas que vienen desde oficinas que no tienen acceso al sistema. Para esto deberá ingresar los siguientes datos.</p> <p>Número de factura, nombre y cédula del remitente, nombre y cedula del destinatario, la ciudad desde donde envían la encomienda, la fecha, la unidad y el turno en el que vino, la fecha de llegada, el valor total, y la descripción. Estos datos se deberán almacenar en el sistema.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 046 - Listar encomiendas de Oficinas sin Sistema			
<b>Nombre Historia:</b>	Listar encomiendas que llegan de Oficinas sin Sistema		
<b>Fecha:</b>	27/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>El usuario deberá filtrar el listado de encomiendas que llegan de oficinas sin sistema en base al número de factura, nombre de la persona quien envía, nombre de la persona quien recibe, el estado de las encomienda ("Recibida", "Entregada"), rango de fechas e indicar si quiere mostrar las encomiendas registradas en todas las oficinas o de la oficina actual.</p> <p>Como resultado se mostrará un listado de las encomiendas que coincidan con los parámetros de búsqueda.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 047 - Entregar encomienda que llega de oficina sin sistema			
<b>Nombre Historia:</b>	Entregar encomienda que llega de oficina sin sistema		
<b>Fecha:</b>	27/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>Para registrar la entrega de una encomienda que llegó de una oficina que no tiene acceso al sistema, el usuario deberá seleccionar un registro del listado obtenido en la historia de usuario número 46.</p> <p>Una vez consultado, se muestran los datos relacionados permitiendo ingresar la cédula de la persona que retira la encomienda en caso de que sea la misma especificada como destinatario. En caso de que otra persona se la que va a retirar, se debe especificar el nombre y la cédula de la persona que retira. Además se debe indicar la fecha en la que se retira la encomienda. Después de ingresar estos datos, se debe registrar la entrega de la encomienda en el sistema.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 048 - Reporte diario de Encomiendas por Oficina			
<b>Nombre Historia:</b>	Reporte diario de Encomiendas por Oficina		
<b>Fecha:</b>	27/03/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	6 Horas
<b>Descripción:</b>			
<p>El usuario va a poder obtener un reporte impreso sobre el reporte diario de las encomiendas facturadas por oficina. Para esto el usuario deberá escoger la oficina y la fecha sobre la cual quiere obtener el resumen de las encomiendas. El reporte se va a mostrar en el formato especificado en el Anexo D.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: flex-end; margin-top: 20px;"> <div style="width: 45%; border-top: 1px dashed black; text-align: center;"> <p>Analista</p> </div> <div style="width: 45%; border-top: 1px dashed black; text-align: center;"> <p>Contraparte</p> </div> </div>			

Historia de Usuario No. 049 - Ingresar una Reserva de Boletos			
<b>Nombre Historia:</b>	Ingresar una Reserva de Boletos		
<b>Fecha:</b>	02/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Baja
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	12 Horas
<b>Descripción:</b>			
<p>El usuario va seleccionar la ciudad de origen, la ciudad de destino, la fecha de la reserva, el turno (Número de unidad y hora de salida). Una vez seleccionada esta información, se debe mostrar automáticamente el número de unidad, la capacidad de pasajeros de la unidad, la hora de salida y los asientos vendidos así como reservados. Se debe ingresar el número de cédula, el nombre del pasajero, el número telefónico para el contacto y seleccionar los asientos a reservar. El usuario ejecuta la acción de guardar y el sistema automáticamente guarda el registro.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 050 - Listar reserva de boletos			
<b>Nombre Historia:</b>	Listar reserva de boletos		
<b>Fecha:</b>	02/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Baja
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción de listar reserva de boletos, obteniendo una lista de las reservas de boletos, pudiendo filtrar por el nombre de la persona que reservó el boleto, por la oficina de origen y destino, y por la fecha de reserva.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 051 - Cancelar reserva de boletos			
<b>Nombre Historia:</b>	Cancelar reserva de boletos		
<b>Fecha:</b>	02/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Baja
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	2 Horas
<b>Descripción:</b>			
<p>El usuario puede Cancelar la reserva de un boleto reservado de la lista obtenida en la historia de usuario 50. Con esto se liberan los asientos reservados.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			



Historia de Usuario No. 052 - Facturar un Boleto			
<b>Nombre Historia:</b>	Facturar un Boleto		
<b>Fecha:</b>	03/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	6 Horas
<b>Descripción:</b>			
<p>El usuario va a seleccionar si se va a facturar con su propio número de factura o con el número de factura de la cooperativa, ingresar el número de cédula y el nombre de la persona que compra el boleto, el lugar a donde viaja, en caso de ser para una fecha posterior deberá cambiar la fecha, seleccionar el turno (número de unidad y hora de salida), escoger si se va a aplicar la tarifa normal o la tarifa especial para el cobro del valor de los boletos. Para finalizar deberá escoger los asientos que compra el pasajero y se calculará automáticamente el valor total de los boletos dejando la opción de modificar dicho valor.</p> <p>El usuario ejecuta la acción de guardar, el sistema automáticamente guarda la información e imprime la factura correspondiente conforme al formato establecido. Ver Anexo E.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 053 - Listar Boletos			
<b>Nombre Historia:</b>	Listar Boletos		
<b>Fecha:</b>	03/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>El usuario va a ingresar al sistema y va a escoger la opción de listar boletos, obteniendo una lista de los boletos facturados, pudiendo filtrar por el nombre de la persona que compró el boleto, por Oficina (boletos facturados en la oficina actual o en todas), y por rango de fechas.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center; margin-top: 20px;"> <div style="border-top: 1px dashed black; width: 40%;"></div> <div style="border-top: 1px dashed black; width: 40%;"></div> </div> <div style="display: flex; justify-content: space-between; margin-top: 5px;"> <div style="width: 40%; text-align: center;"><b>Analista</b></div> <div style="width: 40%; text-align: center;"><b>Contraparte</b></div> </div>			

Historia de Usuario No. 054 - Consultar Boletos			
<b>Nombre Historia:</b>	Consultar Boletos		
<b>Fecha:</b>	03/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	3 Horas
<b>Descripción:</b>			
<p>El usuario selecciona un Boleto de la lista obtenida en la historia de usuario 53, e inmediatamente se le presentará la información asociada al boleto seleccionado en modo de solo lectura.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; align-items: center; margin-top: 20px;"> <div style="border-top: 1px dashed black; width: 40%;"></div> <div style="border-top: 1px dashed black; width: 40%;"></div> </div> <div style="display: flex; justify-content: space-between; margin-top: 5px;"> <div style="width: 40%; text-align: center;"><b>Analista</b></div> <div style="width: 40%; text-align: center;"><b>Contraparte</b></div> </div>			

Historia de Usuario No. 055 - Facturar un boleto reservado			
<b>Nombre Historia:</b>	Facturar un boleto reservado		
<b>Fecha:</b>	03/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	6 Horas
<b>Descripción:</b>			
<p>El usuario previamente debe listar los boletos reservados según historia de usuario 48. Debe seleccionar algún registro desplegado y automáticamente se cargaran los campos especificados previamente para la facturación de los boletos en la historia de usuario 50, junto con los asientos reservados, teniendo la oportunidad de modificarlos.</p> <p>El usuario ejecuta la acción de guardar, el sistema automáticamente guarda la información e imprime la factura correspondiente conforme al formato establecido. Ver Anexo E.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 056 - Listado de Pasajeros			
<b>Nombre Historia:</b>	Listado de Pasajeros		
<b>Fecha:</b>	03/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Media
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	6 Horas
<b>Descripción:</b>			
<p>El usuario podrá generar la lista de pasajeros para despachar el turno. Para esto se debe seleccionar la ciudad de origen, la fecha, el turno (número de unidad y hora de salida). Al seleccionar los campos anteriores, se deberá mostrar automáticamente el nombre del propietario, la capacidad de la unidad y el número de placa. Después de mostrar el listado de los boletos facturados, se debe mostrar el número total de pasajeros y el valor total por la venta de los boletos.</p> <p>El usuario ejecuta la acción Imprimir, e imprime la lista de pasajeros correspondiente conforme al formato establecido. Ver Anexo F</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 057 - Reporte de venta de boletos por oficina			
<b>Nombre Historia:</b>	Reporte de venta de boletos por oficina		
<b>Fecha:</b>	03/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Baja
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	6 Horas
<b>Descripción:</b>			
<p>El usuario va a poder obtener un reporte impreso sobre el reporte de venta de boletos por oficina. Para esto el usuario deberá escoger la oficina y la fecha sobre la cual quiere obtener el reporte. El reporte se va a mostrar en el formato especificado. Ver Anexo G.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="width: 45%; text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 058 - Seguimiento de Encomiendas			
<b>Nombre Historia:</b>	Seguimiento de Encomiendas		
<b>Fecha:</b>	09/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Baja
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	4 Horas
<b>Descripción:</b>			
<p>El cliente va a poder realizar el seguimiento en línea de la encomienda enviada. Esto lo va a hacer a través de la página web que va a estar a disposición de todo el mundo en la dirección web: <a href="http://www.transportesmacuchi.com">http://www.transportesmacuchi.com</a> para lo cual se deberá ingresar el número de factura. El sistema le indicará al cliente el lugar en el cual se encuentra la encomienda, pudiendo variar entre: la ciudad de origen, viajando a la ciudad de destino, en la ciudad de destino y si ya fue entregada.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 059 - Reserva de Boletos en Línea			
<b>Nombre Historia:</b>	Reserva de Boletos en Línea		
<b>Fecha:</b>	09/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Baja
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	40 Horas
<b>Descripción:</b>			
<p>El cliente va a poder realizar la reserva de un boleto en línea. Esto lo va a hacer a través de la página web que va a estar a disposición de todo el mundo en la dirección web: <a href="http://www.transportesmacuchi.com">http://www.transportesmacuchi.com</a>, seleccionar la opción "Reserva de Boletos". Aquí se deberá especificar la ciudad de origen y la ciudad de destino, se deberá escoger la fecha para la cual se quiere hacer la reserva (la fecha deberá ser mayor o igual a la fecha actual) y se mostrarán los turnos disponibles para esa fecha. El sistema deberá mostrar el número de unidad, la hora de salida, la cantidad de asientos del bus. El cliente deberá ingresar el número de cédula, el nombre del pasajero, el número telefónico para contactarle al cliente, seleccionar los asientos que desea reservar con un máximo de 5 asientos. Para realizar la reserva el cliente tendrá un máximo de 5 minutos. Una vez ingresados todos los datos, el cliente realizará la reservación de los boletos y se almacenarán en el sistema.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	

Historia de Usuario No. 060 - Confirmación de Reserva de Boletos			
<b>Nombre Historia:</b>	Confirmación de Reserva de Boletos		
<b>Fecha:</b>	09/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Baja
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	40 Horas
<b>Descripción:</b>			
<p>En la computadora de un operador de la cooperativa, deberá estar instalado un programa que de aviso a penas se realice la reserva de un boleto, Se deberá mostrar un listado con los nombres de las personas que reservaron el boleto, junto con la ruta, el número de asientos y el teléfono de contacto. Como parte del proceso el operador procederá a llamar telefónicamente al cliente para confirmar la reserva y que está se haga efectiva. Se realizaran un máximo de 2 llamadas, si no contestó o no se confirmó, se procederá a cancelar la reservación liberando de esta manera los asientos reservados.</p>			
<b>Notas:</b>			
<div style="display: flex; justify-content: space-between; margin-top: 50px;"> <div style="text-align: center;"> <p>-----</p> <p><b>Analista</b></p> </div> <div style="text-align: center;"> <p>-----</p> <p><b>Contraparte</b></p> </div> </div>			

Historia de Usuario No. 061 – Diseño de la Página web			
<b>Nombre Historia:</b>	Diseño de la Página web		
<b>Fecha:</b>	10/04/2011	<b>Usuario:</b>	Mónica Molina
<b>Riesgo en Desarrollo:</b>	Alta	<b>Prioridad:</b>	Baja
<b>Programador asignado:</b>	Cristóbal Gómez	<b>Estimado:</b>	40 Horas
<b>Descripción:</b>			
<p>En la computadora de un operador de la cooperativa, deberá estar instalado un programa que de aviso a penas se realice la reserva de un boleto, Se deberá mostrar un listado con los nombres de las personas que reservaron el boleto, junto con la ruta, el número de asientos y el teléfono de contacto. Como parte del proceso el operador procederá a llamar telefónicamente al cliente para confirmar la reserva y que está se haga efectiva. Se realizaran un máximo de 2 llamadas, si no contestó o no se confirmó, se procederá a cancelar la reservación liberando de esta manera los asientos reservados.</p>			
<b>Notas:</b>			
-----		-----	
<b>Analista</b>		<b>Contraparte</b>	



#### 4.1.2. Realease Planning.

A continuación se va a describir el plan de entregas de los módulos del sistema. Estos deberán estar totalmente funcionales en las fechas establecidas de acuerdo a los tiempos estimados para cada una de las historias de usuario que comprenden cada uno de los planes.

La prioridad de los planes se determinó de acuerdo a las dependencias que presenten con los módulos posteriores. Para elaborar la tabla de release plan, se agrupó las historias de usuarios por pantallas dentro del sistema.

Pantalla	Tiempo Esti.		Iteración Asignada				
	Días	Horas Hom.	1	2	3	4	5
1. Gestión de ciudades (001,002,003,004,005)	1,5	12	x				
2. Rutas y tarifas (006, 007, 008, 009, 010)	1,5	12	x				
3. Gestión de Oficinas (011, 012, 013, 014)	1,5	12	x				
4. Gestión de Funcionarios (015, 016, 017, 018 019, 020)	4	32		x			
5. Gestión de Vehículos (021, 022, 023, 024)	4	12		x			
6. Gestión de Turnos (025, 026, 027, 028)	1	8		x			
7. Cuadro de trabajo semanal (029, 030, 031, 032, 033, 034)	5	40		x			

8. Autorizaciones del SRI (035, 036, 037)	1	8			x		
9. Datos de las Encomiendas (038, 039, 040, 041, 042, 043, 044, 045, 046, 047, 048)	9	72			x		
10. Datos de Boletos (049, 050, 051, 052, 053, 054, 055, 056, 057)	6	48				x	
11. Reserva de Boletos en Línea (059)	5	40				x	
12. Confirmación de reserva de boletos. (060)	5	40					x
13. Seguimiento de Encomiendas (058)	0,5	4					x
14. Diseño página web (061)	10	80					x

#### 4.1.3. Velocidad del Proyecto.

Representa la el numero de historias que se van evacuando en cada iteración. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias

Para el presente proyecto, previamente se fue definiendo la velocidad de cada iteración después de la especificación de cada tabla con sus respectivas historias de usuario.

#### 4.1.4. Iteraciones.

Iteración 1			
No. His.	Nombre Historia	Horas	Progr. Asignado
001	Agregar una Ciudad	3	Cristóbal Gómez
002	Listar Ciudades	2	Cristóbal Gómez
003	Consultar Ciudades	2	Cristóbal Gómez
004	Modificar una Ciudad	3	Cristóbal Gómez
005	Eliminar una Ciudad	2	Cristóbal Gómez
006	Ingresar Rutas y tarifas	3	Cristóbal Gómez
007	Listar Rutas y Tarifas	2	Cristóbal Gómez
008	Consultar Rutas y Tarifas	2	Cristóbal Gómez
009	Editar Rutas y Tarifas	2	Cristóbal Gómez
010	Relacionar Rutas	3	Cristóbal Gómez
011	Ingresar oficinas	4	Cristóbal Gómez
012	Listar oficinas	2	Cristóbal Gómez
013	Consultar una Oficina	2	Cristóbal Gómez
014	Modificar Oficina	4	Cristóbal Gómez

Velocidad de la Iteración 1: 14

Iteración 2			
No. His.	Nombre Historia	Horas	Progr. Asignado
015	Ingresar datos del Funcionarios	10	Cristóbal Gómez
016	Listar datos del Funcionario	4	Cristóbal Gómez
017	Consultar un Funcionario	3	Cristóbal Gómez
018	Modificar un Funcionario	4	Cristóbal Gómez
019	Cambio de contraseña	3	Cristóbal Gómez
020	Permisos a funcionarios	8	Cristóbal Gómez
021	Ingresar datos de los vehículos	5	Cristóbal Gómez
022	Listar datos de los vehículos	2	Cristóbal Gómez
023	Consultar datos de un vehículo	2	Cristóbal Gómez
024	Modificar datos de un vehículo	3	Cristóbal Gómez
025	Ingreso de Turnos	2	Cristóbal Gómez
026	Listar Turnos	2	Cristóbal Gómez
027	Consultar Turno	2	Cristóbal Gómez
028	Editar Turno	2	Cristóbal Gómez
029	Crear Cuadro de Trabajo Semanal	16	Cristóbal Gómez
030	Ingreso de vehículos que están de parada	8	Cristóbal Gómez
031	Buscar cuadro de trabajo semanal	5	Cristóbal Gómez
032	Buscar vehículos que están de parada	3	Cristóbal Gómez
033	Editar cuadro de trabajo semanal	4	Cristóbal Gómez
034	Editar vehículos que están de parada	4	Cristóbal Gómez

Velocidad de la Iteración 2: 20

Iteración 3			
No. His.	Nombre Historia	Horas	Progr. Asignado
035	Ingreso de un número de autorización del SRI	3	Cristóbal Gómez
036	Listar de autorizaciones del SRI	2	Cristóbal Gómez
037	Consulta de un número de autorización del SRI	3	Cristóbal Gómez
038	Facturación de Encomiendas	12	Cristóbal Gómez
039	Listar las Facturas de las Encomiendas	4	Cristóbal Gómez
040	Consultar una Factura de Encomienda	8	Cristóbal Gómez
041	Anular una factura de encomienda	2	Cristóbal Gómez
042	Generar Hoja de Ruta	6	Cristóbal Gómez
043	Registro de Encomiendas que llegan	8	Cristóbal Gómez
044	Entrega de Encomiendas	16	Cristóbal Gómez
045	Ingresar encomiendas de Oficinas sin Sistema	8	Cristóbal Gómez
046	Listar encomiendas de Oficinas sin Sistema	4	Cristóbal Gómez
047	Entregar encomienda de oficina sin sistema	2	Cristóbal Gómez
048	Reporte diario de Encomiendas por Oficina	6	Cristóbal Gómez

Velocidad de la Iteración 3: 14

<b>Iteración 4</b>			
<b>No. His.</b>	<b>Nombre Historia</b>	<b>Horas</b>	<b>Progr. Asignado</b>
049	Ingresar una Reserva de Boletos	12	Cristóbal Gómez
050	Listar reserva de boletos	4	Cristóbal Gómez
051	Cancelar reserva de boletos	2	Cristóbal Gómez
052	Facturar un Boleto	6	Cristóbal Gómez
053	Listar Boletos	3	Cristóbal Gómez
054	Consultar Boletos	3	Cristóbal Gómez
055	Facturar un boleto reservado	6	Cristóbal Gómez
056	Listado de Pasajeros	6	Cristóbal Gómez
057	Reporte de venta de boletos por oficina	6	Cristóbal Gómez
059	Reserva de Boletos en Línea	40	Cristóbal Gómez

**Velocidad de la Iteración 4: 10**

<b>Iteración 5</b>			
<b>No. His.</b>	<b>Nombre Historia</b>	<b>Horas</b>	<b>Progr. Asignado</b>
058	Seguimiento de Encomiendas	4	Cristóbal Gómez
060	Confirmación de Reserva de Boletos	40	Cristóbal Gómez
061	Diseño de la página web	80	Cristóbal Gómez

**Velocidad de la Iteración 5: 3**

#### **4.1.5. Programación en Pareja.**

La metodología XP aconseja la programación en parejas pues incrementa la productividad y la calidad del software desarrollado. El trabajo en pareja involucra a dos programadores trabajando en el mismo equipo; mientras uno codifica haciendo hincapié en la calidad de la función o método que está implementando, el otro analiza si ese método o función es adecuado y está bien diseñado. De esta forma se consigue un código y diseño con gran calidad.

En este caso en concreto, no aplicaría esta premisa debido a que el presente proyecto es desarrollado únicamente por una persona, quedando bajo su responsabilidad la codificación y la calidad que se le aplique a esta.

#### **4.2. Diseño.**

##### **4.2.1. Diseños Simples.**

Como diseño base de las pantallas de mantenimiento de datos, se propone tener 2 secciones totalmente marcadas. La primera, "Búsquedas", que va a contener los campos necesarios para filtrar la información resultante que se debe mostrar en una grilla de datos. La segunda, "Mantenimiento", que va a servir para el ingreso de los datos. A continuación se muestra un bosquejo de cómo deben quedar las pantallas.



Figura 4.2 Diseño simple de las pantallas en el sistema

#### 4.2.2. Glosario de Términos.

Para una mejor comprensión de la problemática que el sistema planea resolver, se detalla ciertos términos que pueden prestarse para confusiones.

Término	Definición
<b>Ruta</b>	Especifica una ciudad de origen y otra de destino entre las cuales va a viajar una unidad de la cooperativa.
<b>Turno</b>	Hora y fecha de salida de la unidad cubrir una ruta.
<b>Cuadro de trabajo</b>	Distribución de los turnos para cada una de las rutas.



#### 4.2.3. Riesgos.

Uno de los principales riesgos identificados dentro de la temática del desarrollo del sistema es el grado de dificultad técnica para la implementación de una determinada característica. Al estar usando una tecnología especialmente orientada a la implementación de Rich Internet Application, la parte visual juega un papel muy importante dentro del sistema como tal, por lo que ciertas características como la reserva de boletos en línea, puede representar un reto grande al momento de implementarla debido a la gran cantidad de tiempo invertido para la investigación que de seguro va a hacer falta.

Otro riesgo identificado es caer en retrasos para la entrega de la aplicación. Para tratar de eliminar este riesgo se planteó un cronograma más flexible referente a los tiempos, que considere posibles inconvenientes que se presenten en el transcurso del proyecto.

Una mala abstracción de las historias de usuario puede desembocar en uno de los riesgos más peligrosos poniendo en juego la razón de ser del proyecto. Para mitigar esto, se plantea la necesidad y obligación de realizar cuantas reuniones sean requeridas para plasmar el verdadero requerimiento del usuario y por consiguiente del negocio.

#### **4.2.4. Funcionalidad Extra.**

Uno de las prácticas planteadas por la metodología XP es la simplicidad, la cual dicta que solo se realice lo necesario para que la aplicación cumpla con la funcionalidad requerida por el usuario. No es necesario por parte de los desarrolladores trabajar en características que tal vez nunca sean usadas por el cliente.

Los proyectos realizados bajo esta metodología cumplen con lo estrictamente necesario en su funcionalidad en el momento requerido, es decir, hace lo que se necesita cuando se necesita. En XP no conviene precipitarse o adelantarse a las tareas que se han establecido previamente sin el consentimiento del cliente.

Después de lo expresado anteriormente, no se identifica funcionalidades adicionales a las descritas en las historias de usuario.

#### **4.2.5. Reuniones Diarias.**

Debido a la naturaleza del proyecto, a la situación geográfica en la cual se encuentra el desarrollador como el cliente, se ha omitido esta recomendación para dar paso a las reuniones cada que sean necesarias. Estas reuniones sirven para evacuar dudas, para realizar aclaraciones, para redactar historias de usuario, para presentar avances del sistema o para lo que requiera el proyecto.

El resultado de dichas reuniones se encuentra plasmado en todo este documento.

#### **4.2.6. Diagrama de Clase**

**Ver Anexo H.**

#### **4.2.7. Modelo Entidad Relación**

**Ver Anexo I.**

### **4.3. Codificación.**

En este apartado se recomienda revisar el capítulo en el cual se habla sobre las herramientas que se emplearon en el desarrollo.

La codificación es un proceso que se realiza en forma paralela con el diseño y la cual está sujeta a varias observaciones por parte de XP consideradas controversiales por algunos expertos tales como la rotación de los programadores o la programación en parejas. Para este caso ninguna de las 2 premisas antes mencionadas aplica debido a que el proyecto se desarrolla íntegramente por una sola persona.

Se utilizó una arquitectura en 4 capas para el desarrollo de la aplicación. La programación por capas es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño. Toda aplicación contiene código de presentación, código de procesamiento de datos y código de almacenamiento de datos. La arquitectura de las aplicaciones difiere según como está distribuido este código.

Las capas que se utilizaron para el desarrollo de la aplicación web para la cooperativa interprovincial de transportes "Macuchi" es la siguiente:

**1. Capa de presentación.-** También conocida como GUI (Interfaz gráfica de usuario) es la que ve el usuario, también se la denomina "capa de usuario", presenta el sistema al usuario, le comunica la información y captura los datos del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de servicios que en este caso se implementó mediante WCF Services. También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario. Esta capa se va a desarrollar mediante la tecnología Silverlight de Microsoft.

**2. Capa de Servicios.-** Se la conoce como WS (Web Services o Servicios Web), es la capa que sirve de enlace entre la interfaz de usuario y la capa de Negocios. Esta capa se va a implementar con Windows Communication Foundation (WCF).

**3. Capa de Negocios.-** El acrónimo usado comúnmente es BLL (Business Logic Layer) que es donde residen las clases que contienen la lógica propia del negocio que se requiere automatizar. Se reciben las peticiones del usuario que llegan a través de la capa de servicios, se interactúa con la base de datos y se envían las respuestas tras el proceso.

**4. Capa de Datos.-** También conocida por el acrónimo en inglés DL (Data Layer). Es la capa designada para realizar la conexión al gestor de la base de datos. Aquí se define las clases y métodos necesarios para interactuar con el repositorio de datos.

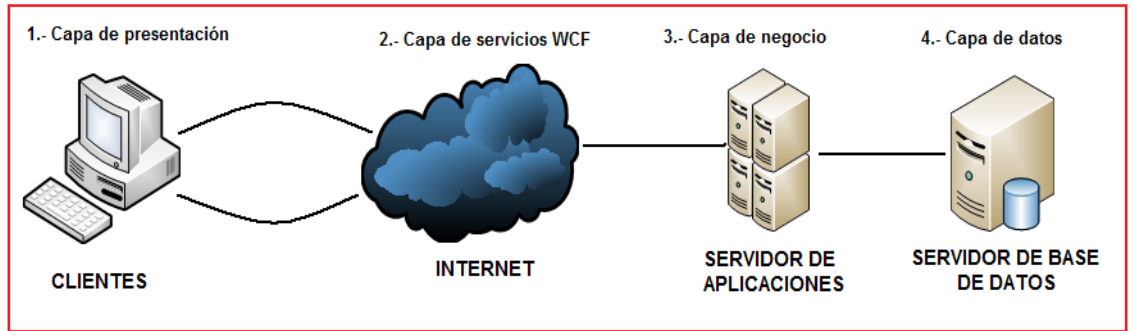


Figura 4.3 Arquitectura de la aplicación.

#### 4.4. Pruebas.

Se plantean 2 tipos de pruebas a realizarse en el sistema.

El primero corresponde a las pruebas unitarias de exclusiva propiedad del desarrollador, el cual será el encargado de ir probando todo lo que haga referente a la historia de usuario en la cual se encuentre trabajando, asegurándose así una buena calidad del sistema en términos de rendimiento, sintaxis así como de 0 excepciones cuando corra el sistema.

Para las pruebas de aceptación se ha elaborado planes de pruebas a seguir, de acuerdo a la pantalla en que se realice las actividades de testing. Dichas pruebas están completamente documentadas en el Anexo J.

## CAPÍTULO 5

### 5. CONCLUSIONES Y RECOMENDACIONES

#### 4.5. Conclusiones.

- El objetivo general de Desarrollar una aplicación web mediante el uso de la tecnología Silverlight de Microsoft, para la administración de encomiendas, rutas, tarifas, boletos, vehículos, y funcionarios para la cooperativa de interprovincial de transportes "Macuchi, se cumplió a cabalidad. Esto quedó demostrado con el desarrollo completo de la aplicación web. Esto evidenció la importancia para las organizaciones de toda índole el mantener su información en internet para facilitar así su acceso, portabilidad y centralización tanto de datos como de software facilitando enormemente el mantenimiento y la actualización.
- El aporte de esta tesis es investigativo y metodológico. Investigativo porque se recaudó información acerca de una reciente tecnología para el desarrollo de aplicaciones de internet que cuentan con mucho potencial. Un claro ejemplo lo establece Microsoft Silverlight en su versión 3, integrado con un potente lenguaje de programación orientado a objetos de Microsoft como lo es C Sharp, con lo último en servicios web representados en Windows Communication Foundation.
- Metodológico porque se siguieron procedimientos organizados basados en la metodología XP que permitieron desarrollar dicha aplicación, con el objetivo de mejorar la innovación y el estado tecnológico de una empresa representante del sector de transporte. XP al ser una metodología de desarrollo ágil fue

de gran utilidad para el presente proyecto debido a la poca cantidad de tiempo con la que se contaba. El desarrollo de la aplicación como tal esta priorizado sobre la cantidad de documentos a generar sin dejar de lado la documentación básica que la misma XP pide. Esto permite obtener productos funcionales cada cierto periodo corto de tiempo, representado un éxito en el desarrollo del proyecto ya que mantenía el interés del cliente en continuarlo, debido a que estaba viendo resultados en el corto plazo.

- Una de las características de las aplicaciones de la web 2.0, es la presentación visual haciendo énfasis en el diseño de la página como tal mediante el lenguaje XAML, utilizando elementos multimedia mediante animaciones, estilos y templates. Para conseguir todo lo descrito de una manera relativamente sencilla, se usó el IDE Microsoft Expression Blend 3.
- Al ser una tecnología de desarrollo web relativamente nueva, Silverlight no trata de complicar la vida del desarrollador ni del diseñador web. Más bien separa de una manera muy clara a las partes involucradas, por un lado el diseñador se concentra en trabajar con el lenguaje XAML y por otro, el desarrollar implementa la lógica del negocio, validaciones y demás características de programación en el lenguaje de su preferencia que los puede escoger entre los soportados por Microsoft.
- Al tener como parte de la arquitectura de la aplicación una capa de WCF services, dejamos abierta la posibilidad de brindar mayor adaptabilidad a la solución, por lo que se puede

eventualmente crear otro "frontend" de distinta plataforma requiriendo únicamente compatibilidad con los servicios WCF.

- Se estableció el internet como medio de conocimiento general. Por lo que, permitió encontrar información referente a la temática tratada en el presente proyecto. Constituyéndose así como una de las principales herramientas de soporte en lo técnico, teórico así como un mecanismo de comunicación entre las partes involucradas.



#### 4.6. Recomendaciones.

- Se deben fijar una serie de reglas generales en la comunicación con el cliente, ya que por el grado de informalidad que la metodología presenta, pueden surgir diferencias que pongan en peligro la culminación exitosa del proyecto.
- Tener un buen conocimiento técnico de las herramientas para llevar a cabo la implementación antes de iniciar dicha etapa. Esto evitará los retrasos en el proyecto así como los tiempos dedicados a la investigación.
- Emplear la metodología XP, para el desarrollo de un proyecto software de mayor envergadura, con un número de participantes superior al presente desarrollo, con el fin de evaluar el real desempeño de la metodología en proyectos a mayor nivel.
- Utilizar estándares de programación para ordenar de forma correcta la parte de diseño así como la lógica, estos pueden ser modelos preestablecidos, propios o adaptados, esto variará según la necesidad del proyecto y de sus participantes.
- Implementar más funcionalidades que la cooperativa como negocio requiera, para aprovechar de mejor manera todas las capacidades tecnológicas con las que contaría desde el momento de la ejecución.

## **Bibliografía.**

1. Introducción a Silverlight;  
<http://msdn.microsoft.com/es-es/library/bb404300.aspx>
2. Microsoft Silverlight  
[http://es.wikipedia.org/wiki/Microsoft\\_Silverlight](http://es.wikipedia.org/wiki/Microsoft_Silverlight)
3. Silverlight  
<http://silverlight.net/>
4. The Official Site of Silverlight  
<http://www.microsoft.com/silverlight/>
5. Aplicaciones de Internet Ricas  
[http://es.wikipedia.org/wiki/Aplicaciones\\_de\\_Internet\\_Ricas](http://es.wikipedia.org/wiki/Aplicaciones_de_Internet_Ricas)
6. ¿Por qué crear aplicaciones RIA?  
[http://www.adobe.com/es/resources/business/rich\\_internet\\_apps/getting\\_started/#](http://www.adobe.com/es/resources/business/rich_internet_apps/getting_started/#)
7. Rich Internet Applications  
<http://www.microsoft.com/silverlight/overview/ria/default.aspx>
8. Programación por capas  
[http://es.wikipedia.org/wiki/Programación\\_por\\_capas](http://es.wikipedia.org/wiki/Programación_por_capas)
9. Una sencilla introducción a los Servicios Web con WCF  
<https://willyxoft.wordpress.com/articulos/servicios-web-wcf/>
10. ¿Qué es Windows Communication Foundation?  
<http://msdn.microsoft.com/es-es/library/ms731082.aspx>
11. Multimedia  
<http://es.wikipedia.org/wiki/Multimedia>
12. Microsoft Visual Studio  
[http://es.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](http://es.wikipedia.org/wiki/Microsoft_Visual_Studio)
13. Información general de Visual Studio 2008  
<http://msdn2.microsoft.com/es-es/vstudio/products/bb931331.aspx>
14. Web 2.0  
[http://es.wikipedia.org/wiki/Web\\_2.0](http://es.wikipedia.org/wiki/Web_2.0)
15. Aplicación Web  
[http://es.wikipedia.org/wiki/Web\\_2.0](http://es.wikipedia.org/wiki/Web_2.0)

16. Qué es la Web 2.0

[http://www.microsoft.com/business/smb/es-es/internet/web\\_2.mspx](http://www.microsoft.com/business/smb/es-es/internet/web_2.mspx)

17. Expression Blend

[http://es.wikipedia.org/wiki/Expression\\_Blend](http://es.wikipedia.org/wiki/Expression_Blend)

18. Expression Blend 2

<http://www.microsoft.com/spain/expression/products/Overview.aspx?key=blend>

19. What is Extreme Programming?

<http://xprogramming.com/what-is-extreme-programming>

# **A N E X O S**

Anexo A.- Cuadro de trabajo

CUADRO DE TRABAJO DE LA SEMANA DEL 30 AL 06 DE DICIEMBRE DEL 2.009									
RUTAS	HORAS	LUN	MAÑ	MIÉ	UE	VIERS	SAB	DOM	RETORNOS
LA MANA - QUITO	1:00	12	14	15	16	17	18	20	09:35 BALZAR RET. 08:10
LA MANA - QUITO	2:00	05	06	07	08	12	14	15	10:00 Echeandia Ret. 05:45
LA MANA - QUITO	3:00	33	34	36	37	38	30	42	10:40 CORZ. RET. 6H00 POR QUZAL.
LA MANA - QUITO	3:30	06	07	08	12	14	15	16	11:20 -
QUEVEDO - QUITO	5:20	22	24	27	28	29	33	34	12:00 CORAZON RET. 07:00
QUEVEDO - QUITO	6:00	71	72	73	75	01	02	03	13:00 CORAZON RET. 10:00
QUEVEDO - QUITO	6:30	48	49	51	55	56	57	58	12:20 -
QUEVEDO - QUITO	7:00	18	20	21	22	24	27	28	14:30 Chasojuan Ret. 08:00
QUEVEDO - QUITO	7:15	58	59	61	62	63	64	65	13:30 -
QUEVEDO - QUITO	7:30	03	05	06	07	08	12	14	14:00 -
QUEVEDO - QUITO	8:00	30	42	44	45	46	47	48	15:20 CORAZ. RET. 11:00 AM.
QUEVEDO - QUITO	8:30	55	56	57	58	59	61	62	15:00
QUEVEDO - QUITO	8:50	29	33	34	36	37	38	30	15:45
QUEVEDO - QUITO	9:20	65	67	68	70	71	72	73	16:30
QUEVEDO - QUITO	9:40	08	12	14	15	16	17	18	17:00
QUEVEDO - QUITO	10:00	21	22	24	27	28	29	33	17:30
QUEVEDO - QUITO	10:30	73	75	01	02	03	05	06	18:00
QUEVEDO - QUITO	11:00	17	18	20	21	22	24	27	18:30
QUEVEDO - QUITO	11:40	62	63	64	65	67	68	70	19:15
QUEVEDO - QUITO	12:20	36	37	38	30	42	44	45	21:00
QUEVEDO - QUITO	13:00	70	71	72	73	75	01	02	5:00
QUEVEDO - QUITO	13:30	45	46	47	48	49	51	55	6:30
QUEVEDO - QUITO	14:10	38	30	42	44	45	46	47	7:15
QUEVEDO - QUITO	15:00	64	65	67	68	70	71	72	8:00
QUEVEDO - QUITO	16:00	57	58	59	61	62	63	64	06:00 CORZ. RET. 16:00 El Corazón-La Maná
El Coraz-Quev-Quit	16:30	16	17	18	20	21	22	24	9:15 -
QUEVEDO - QUITO	17:00	51	55	56	57	58	59	61	07:00 CORAZON RET. 4:00 AM.
QUEVEDO - QUITO	18:00	61	62	63	64	65	67	68	08:40 SAN LUIS Ret. 05:15 AM.
QUEVEDO - QUITO	18:00	-	-	-	-	-	-	22	10:20
QUEV- ESMERALDA	6:15	42	44	45	46	47	48	49	12:40
QUEV- ESMERALDA	8:15	28	29	33	34	36	37	38	14:25
QUEV- ESMERALDA	11:15	47	48	49	51	55	56	57	5:15 PM
QUEV- ESMERALDA	15:15	27	28	29	33	34	36	37	5:35 AM
Quevedo - Morasp.	18:15	02	03	05	06	07	08	12	4:45 Guar. x Quinz. Ret. 14:30 a Moras.
LA MANA - PUCA. X P	13:00	50	43	50	43	50	43	50	15:00
LA MANA - PUCA. X J	15:00	43	50	43	50	43	50	43	17:00
LA MANA - PUCA. X J	18:00	67	68	70	71	72	73	75	6:30
QUEVEDO - GUAYAQUIL	3:45	-	-	-	-	-	-	-	9:00
QUEVEDO - GUAYAQUIL	7:40	-	-	-	-	-	-	-	11:50
QUEVEDO - GUAYAQUIL	9:00	-	-	-	-	-	-	-	13:30
QUEVEDO - GUAYAQUIL	11:50	-	-	-	-	-	-	-	18:00
QUEVEDO - GUAYAQUIL	13:30	-	-	-	-	-	-	-	3:45
QUEVEDO - GUAYAQUIL	18:00	-	-	-	-	-	-	-	7:40
La Unidad que va de 16:30 de Quevedo a Quito, deberá ir a sacar el turno de El Corazón a las 09:00 por la vía Quinz. Y retornará de El Corazón a Quito a las 14:00 por la misma vía.	UNIDADES QUE ESTAN DE PARADAS								
	14	15	16	17	18	20	21		
	07	08	12	14	15	16	17		
	46	47	48	49	51	55	56		
	20	21	22	24	27	28	29		
	72	73	75	01	02	03	05		
	34	36	37	38	30	42	44		

**Anexo B. Facturación de Encomiendas**



**COOPERATIVA INTERPROVINCIAL DE TRANSPORTES**

**MACUCHI**

RUC. 0590047945001 - AUT. SRI. 1108206351

MATRIZ : La Maná, Av. 19 de Mayo # 346 y Medardo Angel Silva - Telf. 032 688142

SUCURSAL: Quito, Quicentro J. de Dios Morales S/N y Maldonado – Telf. 022 957128

SUCURSAL: Quevedo Terminal Terrestre – Telfs. 05-2 750-820 052 750-806

**FACTURA No. 001-001- 0035300**

La Maná, \_\_\_\_\_ del 20 \_\_\_\_\_

Remite: \_\_\_\_\_ Dirección \_\_\_\_\_ RUC. \_\_\_\_\_

Envío desde La Maná a \_\_\_\_\_ Para: \_\_\_\_\_

Motivo del traslado \_\_\_\_\_ Punto de llegada: \_\_\_\_\_ Fecha \_\_\_\_\_

Contenido: \_\_\_\_\_ Valor declarado \_\_\_\_\_ Unid. N°. \_\_\_\_\_ H. Salida \_\_\_\_\_

FLETE PAGADO		
CANT.	DETALLE	VALOR
	CARTAS / SOBRES	
	PAQUETES / BULTOS	
	OTROS	
	GUIA	
	Sub Total	12%
	Sub Total	0%
	Descuento	
	Sub Total	
	IVA, 12%	
	VALOR TOTAL	

Imprenta "QUEVEDO" - Zamora Oribe Luis Fernando  
RUC 1703804706001 - Aut. SRI. 1161 - Calif. Art. 37935  
34301 / 36300 - Emisión: Mayo 04 del 2010  
VALIDO HASTA: MAYO DEL 2011

FLETE AL COBRO		
CANT.	DETALLE	VALOR
	CARTAS / SOBRES	
	PAQUETES / BULTOS	
	OTROS	
	GUIA	
	Sub Total	12%
	Sub Total	0%
	Descuento	
	Sub Total	
	IVA, 12%	
	VALOR TOTAL	

ORIGINAL Cliente - COPIA1: Emisor - COPIA 2: SV Tributario

RECIBI CONFORME

AGENTE \_\_\_\_\_ CHOFER \_\_\_\_\_ REMITENTE \_\_\_\_\_ CED. ID. # \_\_\_\_\_

**IMPORTANTE:** Cooperativa Macuchi no se responsabiliza por deterioro, roturas, manchas, falsedad de declaración de su contenido o ilegalidad en los artículos enviados o recibidos.  
El usuario podrá asegurar su envío pagando la prima adicional del 10% del valor declarado, en este caso Coop. Macuchi reconocerá su valor, de lo contrario solo reconocerá \$ 2,00.  
Cooperativa Macuchi no se responsabiliza por envíos que no hayan sido retirados después de 48 horas, reservándose el derecho de cobrar bodega y de no hacerlo en 30 días se dará de baja. Cooperativa Macuchi no se responsabiliza por dineros enviados dentro de sobres o paquetes. Coop. "MACUCHI" cumple la Ley de Monopolio Postal R.O. N°. 86 de Diciembre de 1958.

Anexo C.- Hoja de Ruta



**COOPERATIVA INTERPROVINCIAL DE TRANSPORTES  
"MACUCHI"**

FUNDADA EL 15 DE ENERO DE 1964, ACUERDO MINISTERIAL N°. 839  
LA MANA - PROVINCIA DE COTOPAXI

**HOJA DE RUTA**

Disco N°:19

**Viaje de:** QUEVEDO - QUITO  
**Fecha:** 12/04/2011

**Propietario:** Rolando Vinicio Conrado Villagomez  
**Hora de salida del Turno:** 08:00

DESTINATARIO	N° FACTURA	DETALLE ENCOMIENDAS	DESTINO	VALOR
MARICELA YANEZ	003-002-0000001	DOCUMENTOS	QUITO	3,25
PABLO CERDA	003-002-0000002	DOCUMENTOS	QUITO	5,25
<b>Total:</b>				8,50

**RECIBI CONFORME**

\_\_\_\_\_  
CHOFER

Recibí conforme bajo mi responsabilidad

Fecha Impresión: 04/12/2011 0:49:24

Página: 1 de: 1

Anexo D.- Reporte diario de Encomiendas por Oficina



**COOPERATIVA INTERPROVINCIAL DE TRANSPORTES  
"MACUCHI"**

FUNDADA EL 15 DE ENERO DE 1964, ACUERDO MINISTERIAL N°. 839  
LA MANA - PROVINCIA DE COTOPAXI

**Reporte diario de las Encomiendas Enviadas**

**Oficina:** Oficina Quevedo Encomiendas

**Fecha:** 04/12/2011

N°. Unidad	Ruta	Hora de salida	Total
19	La Maná - Quito	08:00	8.50
28	Quevedo - La Maná	09:10	1.25
<b>Total:</b>			9.75

Fecha Impresión: 12/4/2011 12:10:31 AM

Página: 1 de: 1

**Anexo E.- Factura de un Boleto**

**ORELLANA LOZADA JORGE LUIS**  
 C/Esq. de Della Ibarra Nieto, Calle Unidad Nacional # 105 y Argentina  
 QUEVEDO - LOS RIOS - ECUADOR  
**Coop. Interprovincial de Transportes**  
**Macuchi**

RUC: 17091980102001  
 Aut. SRI: 1108787385

**FACTURA 001-001**  
**0001100**

Hora de salida: \_\_\_\_\_ De: \_\_\_\_\_ A: \_\_\_\_\_  
 Nombre(s): \_\_\_\_\_ RUC: D.C.I. \_\_\_\_\_  
 Asiento: \_\_\_\_\_ Cargo No. \_\_\_\_\_ Fecha: \_\_\_\_\_

Cant.	Descripción	Precio Unit.	V. de Venta
	PASAJE(S)		
SUB-TOTAL			
IIVA - %			
TOTAL			

ZARDAÑA OSORIO L. E. FERNÁNDEZ IMPRENTA "QUEVEDO" RUC: 05088200005-ALY 9511181  
 RT. 915171198 - (13042) - P. de Asesorías: 3400-3400(1) - VALDO PASAÑA (3400-3400(1))

ORIGINAL: C. Verde  
 COPIA: Em. Gris

**Anexo F.- Listado de Pasajeros**



**COOPERATIVA INTERPROVINCIAL DE TRANSPORTES**  
**"MACUCHI"**

FUNDADA EL 15 DE ENERO DE 1964, ACUERDO MINISTERIAL N°. 839  
 LA MANA - PROVINCIA DE COTOPAXI

**LISTA DE PASAJEROS**

**Fecha:** 04/12/2011 **Placa:** XAG-093  
**Hora:** 10:00 **Disco N°:** 43  
**Conductor:** Romulo Eduardo Ricachi Palma **Capacidad:** 40  
**Origen:** Quevedo  
**Destino:** Quito

N° de Asiento	Nombres	Valor
1	Monica Sanchez	5.00
2	Monica Sanchez	5.00
3	Monica Sanchez	5.00
4	Monica Sanchez	5.00
9	Juan Carlos Suarez	5.00
<b>Total de pasajeros:</b>		<b>5</b>



Anexo G.- Reporte de venta de Boletos por Oficina



**COOPERATIVA INTERPROVINCIAL DE TRANSPORTES  
"MACUCHI"**

FUNDADA EL 15 DE ENERO DE 1964, ACUERDO MINISTERIAL N°. 839  
LA MANA - PROVINCIA DE COTOPAXI

**Reporte diario de los Boletos Facturados**

**Oficina:** Oficina Quevedo Encomiendas

**Fecha:** 04/12/2011

N°. Unidad	Ruta	Hora de salida	Valor Total
43	Quevedo - Quito	10:00	25.00
<b>Total:</b>			25.0000

Fecha Impresión: 12/4/2011 12:36:36 AM

Página: 1 de: 1

## **A n e x o H . - D i a g r a m a d e C l a s e s**



**A n e x o 1.- M o d e l o e n t i d a d - r e l a c i ó n**

## **A n e x o J . - C a s o s d e P r u e b a**