

ESCUELA POLITÉCNICA DEL EJÉRCITO SEDE LATACUNGA



CARRERA DE INGENIERIA DE SISTEMAS E INFORMÁTICA

**TEMA: “DISEÑO Y CONSTRUCCIÓN DE SITIOS WEB MEDIANTE LA
TECNOLOGÍA RICH INTERNET APPLICATIONS (RIA).”**

**CASO PRÁCTICO: DISEÑO DEL SITIO WEB DE LA EMPRESA
BABEL SOFTWARE DEL ECUADOR.**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
SISTEMAS E INFORMÁTICA**

**JUAN GABRIEL PEÑAHERRERA SANDOVAL
DIEGO ISMAEL VELASCO SÁNCHEZ**

Latacunga, Agosto 2009

ESCUELA POLITÉCNICA DEL EJÉRCITO

CARRERA DE INGENIERIA DE SISTEMAS E INFORMÁTICA

CERTIFICADO

ING. JAVIER MONTALUISA (DIRECTOR)

ING. JOSÉ CARRILLO (CODIRECTOR)

CERTIFICAN:

Que el trabajo titulado “DISEÑO Y CONSTRUCCIÓN DE SITIOS DE WEB MEDIANTE LA TECNOLOGÍA RICH INTERNET APLICATIONS (RIA). CASO PRÁCTICO: DISEÑO DEL SITIO WEB DE LA EMPRESA BABEL SOFTWARE ECUADOR.” realizado por los señores: Juan Gabriel Peñaherrera Sandoval y Diego Ismael Velasco Sánchez 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 a los señores: Juan Gabriel Peñaherrera Sandoval y Diego Ismael Velasco Sánchez que lo entreguen al Ing. Edison Espinosa, en su calidad de Director de Carrera.

Latacunga, 31 de agosto de 2009.

Ing. Javier Montaluisa
DIRECTOR DE PROYECTO

Ing. José Carrillo
CODIRECTOR DE PROYECTO

ESCUELA POLITÉCNICA DEL EJÉRCITO

CARRERA DE INGENIERÍA DE SISTEMAS E INFORMATICA

AUTORIZACIÓN

Nosotros, Juan Gabriel Peñaherrera Sandoval y Diego Ismael Velasco Sánchez

Autorizamos a la Escuela Politécnica del Ejército la publicación, en la biblioteca virtual de la Institución de la tesis “DISEÑO Y CONSTRUCCIÓN DE SITIOS DE WEB MEDIANTE LA TECNOLOGÍA RICH INTERNET APLICATIONS (RIA). CASO PRÁCTICO: DISEÑO DEL SITIO WEB DE LA EMPRESA BABEL SOFTWARE ECUADOR.” cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y autoría.

Latacunga, 31 de agosto del 2009

Juan Gabriel Peñaherrera Sandoval
CI. No. 0502501976

Diego Ismael Velasco Sánchez
CI. No. 0502603855

DEDICATORIA

Por Juan Peñaherrera:

Dedico este arduo trabajo con todo cariño a mis padres que me han sabido guiar en toda mi vida y en cada uno de mis días, con mucha dedicación y amor, además, a mis hermanos que de una u otra forma han sido un ejemplo y un complemento para mí, a mis abuelitos que han guiado a toda una familia en el bien, a mis tíos y primos que llenan de alegría mi vida. A mi enamorada que es una amiga y compañera, muy cariñosa, comprensiva y un apoyo cada día.

A todos mis profesores que tuvieron el generoso plan de enviarnos a Costa Rica, a la empresa Babel Software, a todos mis jefes que han sabido aconsejarme para lograr mis metas, a mis amigos que siempre han sido mi fuerza a lo largo de la vida, además, a Diego que siempre ha sido un buen amigo y confió en mí para realizar este proyecto.

Y sobre todo a Dios, por darme el regalo de tener una vida llena de seres queridos, de magníficas experiencias y de metas por cumplir, también, por darme una nueva oportunidad de vivir y una nueva visión de la vida; pienso que eso me hace aún más fuerte cada día, teniendo el propósito de mejorar este mundo y de alguna u otra forma la vida de los demás.

Por Diego Velasco:

Dedico el presente proyecto a mis padres Edison y Marcia, hermanos: Paola, Gabriela y Edison Patricio y sobrinos: Ismael, Gabriel y Ana Paula por su amor, cariño e incondicional apoyo en procura de conseguir esta meta personal.

Mi total respeto y gratitud para Juan por invitarme a participar de este proyecto y lograr en conjunto el objetivo de graduarnos.

Igualmente con respeto y gratitud a los ingenieros: José Luis Carrillo, Edison Espinosa, Javier Montaluisa, Edgar Oviedo, y Esteban Oviedo por confiarme junto con sus enseñanzas: su amistad.

Y a todos los buenos amigos, familiares, profesores y compañeros por contribuir generosamente en mi crecimiento personal y mi bienestar; personas con quienes felizmente seguiremos avanzando en los caminos futuros con la bendición de Dios.

CONTENIDO

I. CAPÍTULO I.....	14
SITIO WEB Y DISEÑO WEB.....	14
1.1. INTRODUCCIÓN A SITIO WEB Y DISEÑO WEB.....	14
1.1.1. SITIO WEB.....	14
1.1.2. SITIO WEB ESTÁTICO.....	15
1.1.3. SITIO WEB DINÁMICO.....	15
1.1.4. FUNCIONAMIENTO DE UN SITIO WEB.	15
1.1.5. DISEÑO WEB.	17
1.2. ARQUITECTURAS DE SITIOS WEB.....	18
1.2.1. INTRODUCCIÓN.....	18
1.2.2. ARQUITECTURA WEB BÁSICA [CLIENTE/SERVIDOR]	18
1.2.3. ANÁLISIS DE PLATAFORMAS DE DESARROLLO WEB.....	20
1.2.3.1. Navegador como plataforma.....	20
1.2.3.1.1. HTML/XHTML.....	20
1.2.3.1.2. DHTML.....	21
1.2.3.1.3. AJAX.....	23
1.2.3.2. Plataforma Flash.....	26
1.2.3.3. Otras Plataformas RIA.....	28
1.3. EVOLUCIÓN DEL DISEÑO DE PÁGINAS WEB.	29
1.3.1. GENERACIONES DE DISEÑO.....	29
1.3.1.1. Primera Generación.....	29
1.3.1.2. Segunda Generación.....	31
1.3.1.3. Tercera Generación.....	33
1.3.1.3.1. Plugins.....	34
1.3.1.3.2. CSS.....	35
1.3.1.4. Cuarta Generación.....	35
1.3.2. EVOLUCIÓN WEB.....	36
1.3.2.1. Web 1.0.....	36
1.3.2.2. Web 2.0.....	36
1.3.2.3. Tecnologías de la Web 2.0	38

1.3.2.4.	Mapa Mental Web 2.0.....	38
1.3.2.5.	Cuadro Comparativo de Web1.0 y Web 2.0.....	39
1.3.2.6.	WebTop.....	39
1.3.3.	INTRODUCCIÓN A LAS RICH INTERNET APPLICATION.....	40
1.4.	ARQUITECTURA DE LA INFORMACIÓN.....	41
1.4.1.	DEFINICIÓN.....	41
1.4.2.	LA ARQUITECTURA DE LA INFORMACIÓN DE UN SITIO WEB.....	42
1.4.2.1.	Objetivos.....	42
1.4.2.2.	Organización de la Información en un sitio Web.....	42
1.4.2.2.1.	Esquemas de Organización de la Información. EOI.....	43
1.4.2.2.2.	Estructuras de Organización de la Información. ESOI.....	44
1.4.2.3.	Sistemas de Navegación. SN.....	45
1.4.3.	USABILIDAD.....	46
1.4.4.	ACCESIBILIDAD.....	47
1.4.5.	LA ACCESIBILIDAD EN LAS RIA SEGÚN LA W3C.....	47
1.4.6.	SISTEMA DE GESTIÓN DE CONTENIDOS.....	49
II.	Capítulo II.....	50
	TECNOLOGÍA RICH INTERNET APLICATION RIA.....	50
2.1.	CONCEPTO: RIA.....	50
2.1.1.	DEFINICIÓN DE RIA POR MACROMEDIA.....	51
2.2.	APLICACIONES RIA FRENTE A APLICACIONES WEB TRADICIONALES....	52
2.3.	VENTAJAS Y RESTRICCIONES.....	54
2.3.1.	ALGUNOS BENEFICIOS SOBRE LAS APLICACIONES WEB TRADICIONALES.....	54
2.3.2.	BENEFICIOS EN EL ÁMBITO DE TECNOLOGÍAS DE INFORMACIÓN (TI)	
	57	
2.3.2.1.	Mejorar la productividad de los desarrolladores.....	57
2.3.2.2.	Aprovecha el ambiente de TI existente.....	57
2.3.2.3.	Costos de operación menores.....	57
2.3.3.	RESTRICCIONES.....	57
2.3.3.1.	SandBox.....	57
2.3.3.2.	Scripting Deshabilitado.....	57

2.3.3.3.	Velocidad de procesamiento en el Cliente.....	58
2.3.3.4.	Tiempo de descarga del Script.....	58
2.3.3.5.	Pérdida de visibilidad para los motores de búsquedas.....	58
2.3.3.6.	Dependencia de una conexión a Internet.....	58
2.3.3.7.	Complicaciones de Administración	58
2.4.	ARQUITECTURA RIA.....	60
2.4.1.	CONEXIONES CON EL SERVIDOR DE BASE DE DATOS.....	60
2.4.2.	ACCESO A DATOS	61
2.4.2.1.	Componentes de la capa de acceso a datos.	63
2.4.2.1.1.	Componentes de la capa de datos lógica.	63
2.4.2.1.2.	Ayudadores de datos y Utilidades.....	63
2.4.2.1.3.	Agentes de Servicio.....	63
2.4.3.	CAPA DE NEGOCIO.....	64
2.4.3.1.	Componentes de la capa de negocio.....	64
2.4.3.2.	Entidades del negocio.....	65
2.4.4.	CAPA DE SERVICIOS DE LA APLICACIÓN	66
2.4.4.1.	Componentes de la capa de Servicios.....	66
2.4.4.1.1.	Interfaces de Servicio	66
2.4.4.1.2.	Tipos de mensajes.....	66
2.4.5.	CAPA DE PRESENTACIÓN	67
2.4.5.1.	Componentes de la Interfaz de usuario	67
2.4.5.2.	Procesos de componentes de Interfaz de Usuario	68
2.4.6.	EXPERIENCIA DE USUARIO	68
2.5.	TECNOLOGÍA MICROSOFT RIA: MICROSOFT SILVERLIGHT.....	69
2.5.1.	INTRODUCCIÓN A WPF/E O SILVERLIGHT	69
2.5.2.	EVOLUCIÓN DEL DESARROLLO WEB: AVANZANDO HACIA LA WEB.NEXT	70
2.5.3.	INFORMACIÓN GENERAL SOBRE SILVERLIGHT	72
2.5.3.1.	Características.....	73
2.5.3.1.1.	LINQ.....	75
2.5.3.2.	Crear aplicaciones basadas en Silverlight.....	79
2.5.3.3.	Aplicaciones de Servidor con Silverlight.....	79

2.5.4.	ARQUITECTURA SILVERLIGHT	80
2.5.4.1.	Plataforma Silverlight	81
2.5.4.2.	Arquitectura	83
2.5.4.2.1.	Componentes de presentación básicos	84
2.5.4.2.2.	.Net Framework para Silverlight	86
2.5.4.2.3.	Características de programación adicionales de Silverlight	87
2.5.4.2.4.	Sistemas operativos y exploradores compatibles	88
2.5.4.2.5.	Herramientas y tecnologías relacionadas	89
2.5.4.3.	Competencia más cercana	91
2.6.	UTILIDADES MICROSOFT	91
2.6.1.	CONTROLES DE USUARIO	91
2.6.2.	WINDOWS COMMUNICATION FOUNDATION WCF SERVICES.	95
2.6.2.1.	Arquitectura orientada a servicios.	95
2.6.2.2.	Composición del WCF Service.	96
2.6.2.3.	Comunicación con el servicio.	97
2.6.2.3.1.	Sincrónica	97
2.6.2.3.2.	Asincrónica	98
2.6.2.4.	Estilos XAML	99
III.	CAPÍTULO III	102
	DISEÑO DEL SITIO WEB DE BABEL SOFTWARE ECUADOR	102
3.1.	ANÁLISIS Y DEFINICIÓN DE CARACTERÍSTICAS DEL SITIO WEB.	102
3.1.1.	ANTECEDENTE	102
3.1.2.	ARQUITECTURA DE LA INFORMACIÓN.	103
3.1.2.1.	Estructura de Organización de la Información (ESOI)	103
3.1.3.	USUARIOS.	105
3.1.3.1.	Usuario anónimo.	105
3.1.3.2.	Usuario registrado.	105
3.1.4.	ROLES.	105
3.1.4.1.	Administrador (Web Máster).	105
3.1.4.2.	Editor.	106
3.1.4.3.	Visitante.	106
3.1.5.	CATEGORÍAS DE CONTENIDOS	106

3.1.5.1. Compañía.....	106
3.1.5.2. Servicios.....	107
3.1.5.3. Noticias.....	107
3.1.5.4. Casos de éxito.....	107
3.1.5.5. Contáctenos.....	107
3.1.5.6. Administración.....	107
3.1.6. DISEÑO DE LA APLICACIÓN.....	108
3.1.6.1. Arquitectura.....	108
3.2. DESARROLLO DEL SITIO WEB DE BABEL SOFTWARE ECUADOR.....	109
3.2.1. ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE.....	109
3.2.1.1. Requisitos funcionales.....	109
3.2.1.2. Requisitos no funcionales.....	109
3.2.1.3. Casos de Uso.....	109
Ver Anexo ERS y EDS.....	109
3.2.2. IMPLEMENTACIÓN.....	109
3.2.3. PRUEBAS.....	111
3.2.4. INTEGRACIÓN.....	113
3.2.4.1. Creación de los instaladores.....	113
3.2.4.2. Configuración del servidor.....	116
3.2.4.2.1. Configuración de Keys.....	119
IV. Capítulo IV.....	121
RESULTADOS.....	121
4.1. CONCLUSIONES.....	121
4.2. RECOMENDACIONES.....	123
4.2.1. RECOMENDACIONES SOBRE EL DESARROLLO.....	123
4.2.2. RECOMENDACIONES SOBRE EL PRODUCTO SOFTWARE.....	124
Referencia Bibliográfica.-.....	126
ANEXOS.....	127
ANEXO1.....	127
Carta de Aprobación.....	128
Introducción.....	129
Propósito.....	129

Ámbito del Sistema.....	129
Beneficios.....	129
Funciones del Producto.....	130
Objetivos.....	130
Definiciones, Acrónimos y Abreviaturas.....	131
Actores.....	131
Suposiciones y Dependencias.....	132
Requerimientos funcionales	132
Diagramas General de Casos de Uso	133
Requerimientos no funcionales	148
Requerimientos de Rendimiento.....	148
Restricciones de Diseño.....	149
Atributos del Sistema	150
Otros Requerimientos.....	151
ANEXO 2.....	152
Carta de Aprobación	153
Introducción	154
<i>Propósito</i>	154
<i>Definiciones, Acrónimos y Abreviaturas.....</i>	154
<i>Visión General del Documento</i>	154
Algoritmos	155

CONTENIDO DE IMÁGENES

FIGURA 1.1 Funcionamiento de un sitio web.	17
FIGURA 1.2 Arquitectura web básica [cliente/servidor]	19
FIGURA 1.3 Modelo clásico de Aplicación Web.	20
FIGURA 1.4 Modelo de Aplicación Web con DHTML.	22
FIGURA 1.5 Modelo de Aplicación Web con AJAX.	25
FIGURA 1.6 Modelo de Aplicación Web con Flash.....	27
FIGURA 1.7 Primera Generación.	30
FIGURA 1.8 Segunda Generación.	32
FIGURA 1.9 Tercera Generación.	33
FIGURA 1.10 Mapa Mental Web 2.0.	38
FIGURA 1.11 Ejemplo de WebTop.....	40
FIGURA 1.12 Interacción CMS.....	49
FIGURA 2.1 RIA.....	50
FIGURA 2.2 Aplicación Web Tradicional.	52
FIGURA 2.3 Aplicación RIA.....	53
FIGURA 2.4 Aplicación Web Tradicional frente a RIA.	53
FIGURA 2.5 Complicaciones de Administración.....	59
FIGURA 2.6 Arquitectura RIA.....	60
FIGURA 2.7 Capas de la arquitectura RIA.	62
FIGURA 2.8 Silverlight.	69
FIGURA 2.9 Arquitectura de LINQ	76
FIGURA 2.10 LINQ del proyecto.	76
FIGURA 2.11 Arquitectura Silverlight.	83
FIGURA 2.12 Microsoft Expression Blend.....	89
FIGURA 2.13 Microsoft Visual Studio 2008.....	89
FIGURA 2.14 Programación sin usar User Controls.....	92
FIGURA 2.15 User Control.....	94
FIGURA 2.16 WCF Service.....	96
FIGURA 3.1 Estructura de Organización de la Información (ESOI).	103

FIGURA 3.2 Arquitectura de la aplicación.	108
FIGURA 3.3 Nuevo proyecto Silverlighth.....	109
FIGURA 3.4 Añadir un proyecto Web.....	110
FIGURA 3.5 Solución Silverlight.....	110
FIGURA 3.6 Solución WCF.....	110
FIGURA 3.7 Microsoft Expression Blend 2.0.....	110
FIGURA 3.8 Creación de instaladores Paso 1.....	113
FIGURA 3.9 Creación de instaladores Paso 2.....	114
FIGURA 3.10 Creación de instaladores Paso 3.....	115
FIGURA 3.11 Creación de instaladores Paso 4.....	116
FIGURA 3.12 Configuración del Servidor Paso 1.....	117
FIGURA 3.13 Configuración del Servidor Paso 2.....	117
FIGURA 3.14 Configuración del Servidor Paso 3.....	118
FIGURA 3.15 Configuración del Servidor Paso 4.....	118
FIGURA 3.16 Configuración del Servidor Paso 5.....	119

RESUMEN

El presente proyecto de tesis se ha dividido en 4 capítulos:

Capítulo 1. Habla acerca de lo que son Sitios Web, su diseño, su arquitectura, su evolución y la arquitectura de la información, pieza clave en toda clase de proyectos.

Capítulo 2. Trata sobre la tecnología RIA, sus ventajas frente a aplicaciones Web tradicionales, restricciones, arquitectura y la parte fundamental de este proyecto: Microsoft Silverlight.

Capítulo 3. Es meramente el diseño del Sitio Web, con el análisis y definición de características del mismo y su desarrollo, el cual contempla la base sobre la cual todo producto software debe realizarse: los requisitos del sistema. Finalmente, vienen sus pruebas e integración.

Capítulo 4. Son las conclusiones y recomendaciones a las cuales se ha llegado, al terminar este proyecto de tesis.

PRESENTACIÓN

El presente proyecto explora los beneficios de la tecnología RIA (Rich Internet Applications) en la construcción de Sitios Web tomando como ejemplo práctico el diseño del Sitio Web de Babel Software del Ecuador.

Las herramientas utilizadas son Microsoft Silverlight 2.0, Microsoft Expression Blend 2.0 y Microsoft Visual Studio 2008; las mismas que son compatibles con la tecnología Microsoft .NET y están enfocadas al desarrollo Web; además, cuentan con flexibilidad para incorporar características de WCF (Windows Communication Foundation), LINQ (Microsoft Language Integrated Query).

El proyecto contempla el desarrollo de controles de usuario, animaciones, manejador de contenidos, estilos de diseño XAML; además de las convencionales técnicas y características del desarrollo de software mediante la metodología UML.

Para cumplir con los objetivos del proyecto, ha sido necesario definir las características del Sitio Web de Babel Software del Ecuador, en cuanto a información, estructura y contenidos; a fin de obtener un resultado dinámico, de calidad, óptimo y confiable, que refleje fielmente lo que la empresa Babel representa dentro del ámbito del software a nivel nacional e internacional.

I. CAPÍTULO I

SITIO WEB Y DISEÑO WEB

1.1. INTRODUCCIÓN A SITIO WEB Y DISEÑO WEB.

1.1.1. SITIO WEB.

Conceptualmente un sitio web es un conjunto de páginas web perteneciente a un dominio o subdominio común en la World Wide Web.

Las páginas web, contenidas dentro de un sitio web, están escritas en HTML (Hyper Text Markup Language), o dinámicamente convertidas a éste y se acceden usando un software llamado navegador web, también conocido como un cliente HTTP.

El alojamiento de un sitio web, también conocido como hosting, se realiza en una computadora denominada *servidor web* o *servidor HTTP*; estos términos también son aplicables para el software que se ejecuta en la computadora, encargado de recuperar y entregar las páginas web conforme a las peticiones del usuario; entre los software de servidor web, tenemos Apache (Open Source), y el Internet Information Services (IIS) de Microsoft.

Las páginas del web al principio fueron estáticas; una vez puestas en línea se mantenían inalterables hasta que alguien las cambie. Fue básicamente una pantalla gris con texto negro y algunas imágenes, y mientras evolucionó se utilizaron significativas mejoras como los CGI (common gateway interface)¹ que permiten a personas ejecutar animaciones simples y unir sus páginas web a las bases de datos.

¹ http://es.wikipedia.org/wiki/Interfaz_de_entrada_com%C3%BA

1.1.2.SITIO WEB ESTÁTICO.

Un sitio web estático es aquel cuyo contenido no se espera que cambie frecuentemente, para su escritura se utilizan editores de texto en donde se puede escribir directamente el código HTML o generarlo automáticamente por medio de un editor WYSIWYG (What you see is what you get, Lo que ves es lo que obtienes)².

Generalmente estos sitios presentan un volumen de información limitado y una interactividad con el usuario básica.

1.1.3.SITIO WEB DINÁMICO.

Un sitio web dinámico es aquel que puede tener cambios frecuentes en la información, potenciado por un software CGI residente en el servidor web que ejecuta un script o código de acuerdo a varios parámetros en la petición del cliente como pueden ser: variables, sesiones, perfiles de usuario, etc.

Entre los lenguajes de programación utilizados para generar sitios web dinámicos tenemos: PHP, Active Server Pages (ASP), y Java Server Pages (JSP); también se considera web dinámica DHTML, a la interactividad, basada en JavaScript en el lado del cliente, una vez que las páginas han sido descargadas.

Los sitios dinámicos a menudo incluyen contenido que se recupera de una o más bases de datos conectados por medio de diferentes tecnologías como ADODB, ODBC, y últimamente XML.

1.1.4.FUNCIONAMIENTO DE UN SITIO WEB.

La visualización de un sitio web normalmente comienza tecleando la URL de la página en el navegador web o siguiendo un enlace de hipertexto a esa

² <http://es.wikipedia.org/wiki/WYSIWYG>

página o recurso. En ese momento el navegador comienza una serie de comunicaciones, transparentes para el usuario, para obtener los datos de la página y visualizarla.

El primer paso consiste en traducir la parte del nombre del servidor de la URL en una dirección IP usando la base de datos distribuida de Internet conocida como DNS. Esta dirección IP es necesaria para contactar con el servidor web y poder enviarle paquetes de datos.

El siguiente paso es enviar una petición HTTP al servidor Web solicitando el recurso. En el caso de una página web típica, primero se solicita el texto HTML y luego es inmediatamente analizado por el navegador, el cual, después, hace peticiones adicionales para los gráficos y otros ficheros que formen parte de la página.

Así que se han recibido los ficheros solicitados desde el servidor web, el navegador interpreta la página tal y como se describe en el código HTML, el CSS y otros lenguajes web. Al final se incorporan las imágenes y otros recursos para producir la página que ve el usuario en su pantalla.

Si un usuario accede de nuevo a una página después de un pequeño intervalo, es probable que no se vuelvan a recuperar los datos desde el servidor web, sino de la caché del disco duro que los navegadores almacenan localmente. El navegador enviará una petición HTTP sólo si la página ha sido actualizada desde la última carga, en otro caso, la versión almacenada se reutilizará para agilizar la visualización de la página.

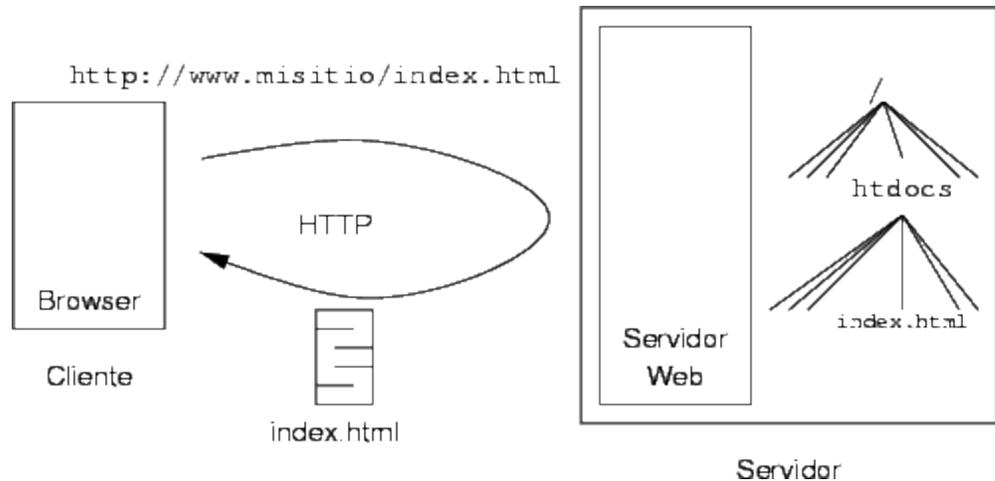


FIGURA 1.1 Funcionamiento de un sitio web.

1.1.5. DISEÑO WEB.

El diseño Web es una actividad que ha ido a la par de desarrollo de la tecnología de Internet; su misión es explorar las diferentes alternativas para la implementación de un sitio en los ámbitos como: conexiones, arquitecturas de información, diseño gráfico, plataformas de desarrollo, lenguajes de servidor, servicios web, protocolos de seguridad, etc.

La web fue específicamente desarrollada para navegar en complejas redes de información usando visualización gráfica y unas simples técnicas de navegación "apunta y pulsa". Este paradigma ha habilitado a usuarios inexpertos a explorar este nuevo medio sin tener que aprender un lenguaje de comandos textuales.

Es así, como de una simple curiosidad dentro de las organizaciones pasó a ser pieza fundamental del crecimiento y razón de ser para muchas organizaciones visionarias que vieron en la Web su plataforma de expansión, comercio, comunicación, etc.

1.2. ARQUITECTURAS DE SITIOS WEB.

1.2.1.INTRODUCCIÓN.

La expansión del internet y el uso de redes corporativas, ha supuesto una transformación en las necesidades de información de las organizaciones, sintetizadas en que:

- La información sea accesible desde cualquier lugar dentro de la organización e incluso desde el exterior.
- La información sea compartida para todas las partes interesadas de acuerdo a su función en todo momento.

Estas necesidades, han provocado un movimiento creciente de cambio de las aplicaciones tradicionales de escritorio hacia las aplicaciones web; que por su idiosincrasia, cumplen a la perfección con las necesidades mencionadas anteriormente. Por lo tanto, los sitios web se han convertido en aplicaciones capaces de una interacción más sofisticada con el usuario, lo que inevitablemente provoca un aumento progresivo en la complejidad de estos sistemas y, por ende, la necesidad de buscar opciones de diseño nuevas que permitan dar con la arquitectura óptima que facilite su construcción.

1.2.2.ARQUITECTURA WEB BÁSICA [CLIENTE/SERVIDOR]

Conceptualmente una aplicación web interactúa a través de los navegadores con un servidor web que evalúa las peticiones y normalmente hace uso de una base de datos.

En un principio el navegador es un simple presentador de información y no presenta procesamiento relacionado con la lógica del negocio, considerando a la arquitectura web un modelo de cliente/servidor simple; sin embargo la

utilización de DHTML, JavaScript, Applets u otros scripts que se ejecuten del lado del cliente, puede flexibilizar la arquitectura hacia una multicapas. No obstante el procesamiento realizado en el cliente suele estar relacionado con aspectos de la interfaz (como ocultar o mostrar secciones de la página en función de determinados eventos) y nunca con la lógica de negocio.

En consideración a ello una aplicación web propondría básicamente 3 capas o niveles.

1. Nivel de Presentación. Es el encargado de generar la interfaz de usuario en función de las acciones llevadas a cabo por el mismo.
2. Nivel de Reglas de Negocio. Contiene toda la lógica que modela los procesos de negocio y es donde se realiza todo el procesamiento necesario para atender a las peticiones de usuario.
3. Nivel de Administración de Datos. Encargado de hacer persistente toda la información, suministra y almacena información para el nivel de negocio.

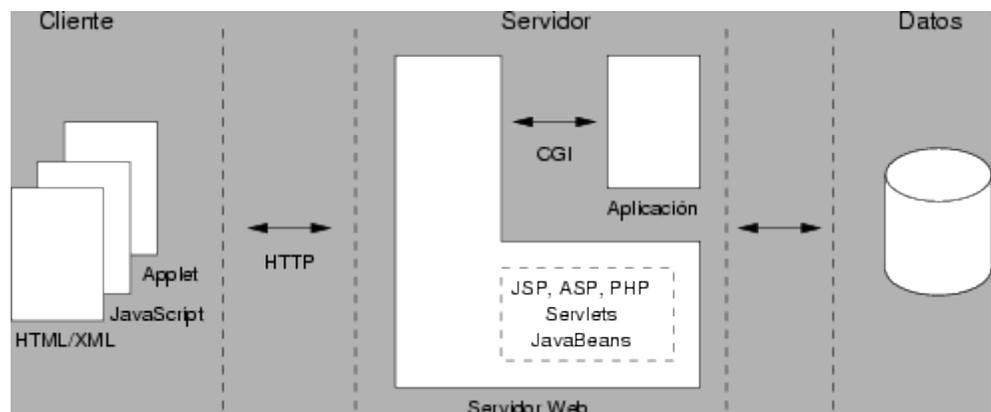


FIGURA 1.2 Arquitectura web básica [cliente/servidor]

1.2.3. ANÁLISIS DE PLATAFORMAS DE DESARROLLO WEB.

Las plataformas de desarrollo web son las diversas formas de implementar un sitio o aplicación web en cuanto a su arquitectura; en la última década este proceso ha tenido un crecimiento sustancial producto de la diversidad de ofertas entre los proveedores de software como Microsoft, Sun Microsystems, Adobe, entre otros desaparecidos como Netscape y Macromedia; conjuntamente, a los esfuerzos de la W3C para estandarizar el uso de los lenguajes de marcado de hipertexto.

El siguiente análisis muestra las características de distintas plataformas de desarrollo web, las ventajas y limitaciones de cada una desde los criterios de riqueza de navegación hacia el usuario, rapidez de repuesta, y dificultad de desarrollo.

1.2.3.1. Navegador como plataforma.

1.2.3.1.1. HTML/XHTML

Considerando a HTML como el primer lenguaje de hipertexto, se creó una extensión del lenguaje más estricta y conforme a las especificaciones XML, este lenguaje se define como XHTML.

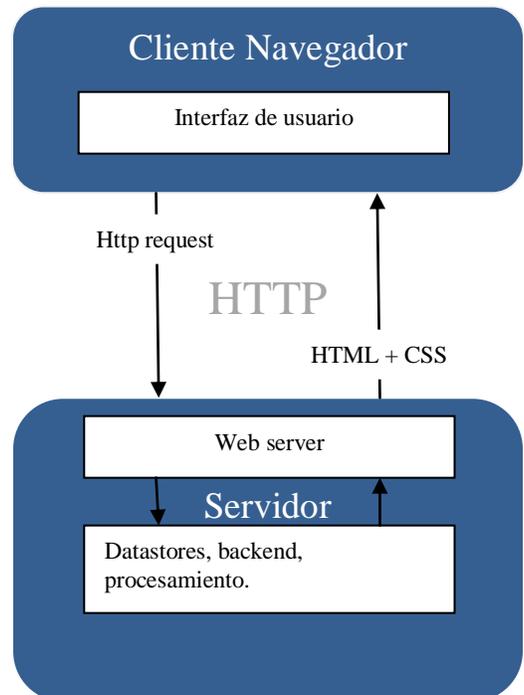


FIGURA 1.3
Modelo clásico de Aplicación Web.

	Ventajas	Limitaciones
Usuario	Perfecta interacción entre HTML y el navegador. Perfectamente accesible a través de los buscadores.	Contenido estático. Limitada riqueza de contenidos y creación de componentes.
Desarrollo	Fácil de aprender a través de las herramientas WYSIWYG. Fácil de desarrollar con un solo servidor, no es necesaria la instalación de clientes o actualizaciones. Bajo costo de desarrollo.	Funcionalidad limitada. Los navegadores tienen diferentes sintaxis de interpretación.
Respuesta	Rápida interpretación de contenidos en los navegadores.	Cada petición y respuesta recarga la totalidad de la página, no hay garantía sobre el arribo e integridad de los datos.

1.2.3.1.2. DHTML.

El HTML dinámico designa el conjunto de técnicas que permiten crear sitios web interactivos utilizando una combinación de lenguaje HTML estático, un lenguaje interpretado en el lado del cliente (como JavaScript), el lenguaje de hojas de estilo en cascada (CSS) y la jerarquía de objetos de un DOM (Document Object Model, Modelo de objetos para la presentación de documentos).³

³ http://es.wikipedia.org/wiki/Document_Object_Model

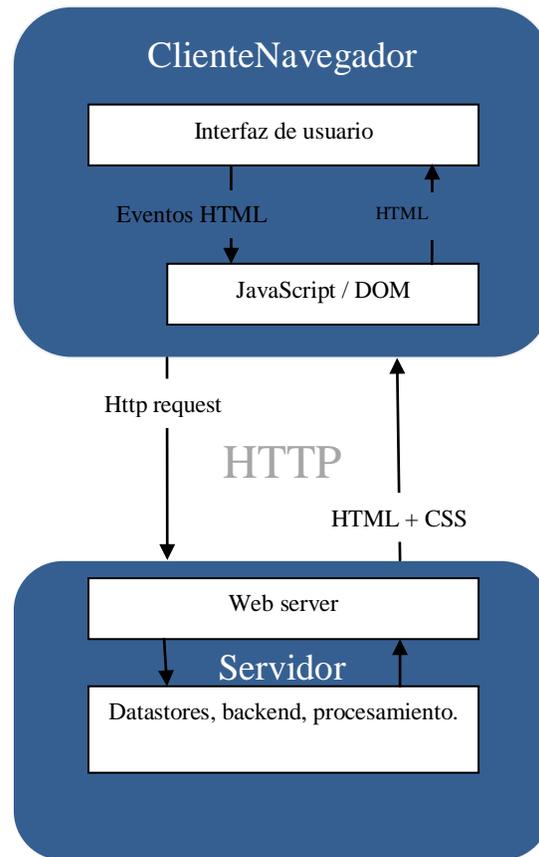


FIGURA 1.4 Modelo de Aplicación Web con DHTML.

	Ventajas	Limitaciones
Usuario	<p>Son familiares los componentes HTML, los estilos cambian, se personalizan, hay pequeñas animaciones.</p> <p>El contenido HTML es fácil de indexar.</p>	<p>No hay soporte sobre multimedia.</p> <p>Los buscadores no pueden diferenciar el contenido de Java Script.</p>
Desarrollo	<p>Fácil de desarrollar con un solo servidor, no es necesaria la instalación de clientes o actualizaciones.</p>	<p>El lenguaje Java Script puede deshabilitarse del navegador.</p> <p>Diferentes navegadores interpretan</p>

	Existen varias herramientas que facilitan el desarrollo. Separación del estilo y del contenido debido al CSS. Fácil de encontrar personal capacitado en desarrollo	diferente el lenguaje Javascript y CSS. Incremento del costo para cubrir la compatibilidad con todos los navegadores.
Respuesta	El despliegue de páginas sigue siendo relativamente rápido, en cuanto a visualización aunque tarda en la carga del código de Javascript. Se reduce la carga sobre el servidor y existe procesamiento del lado del cliente.	

1.2.3.1.3. AJAX.

AJAX, acrónimo de 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.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se solicitan al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página.

JavaScript es el lenguaje interpretado 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 porque está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

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 VBScript, para mostrar e interactuar dinámicamente con la información presentada.
- El objeto XMLHttpRequest para intercambiar datos asincrónicamente 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 (subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML) y

hasta EBML (Meta Lenguaje Binario Extendible almacena y manipula datos de forma jerárquica con campos de longitud variable)⁴.

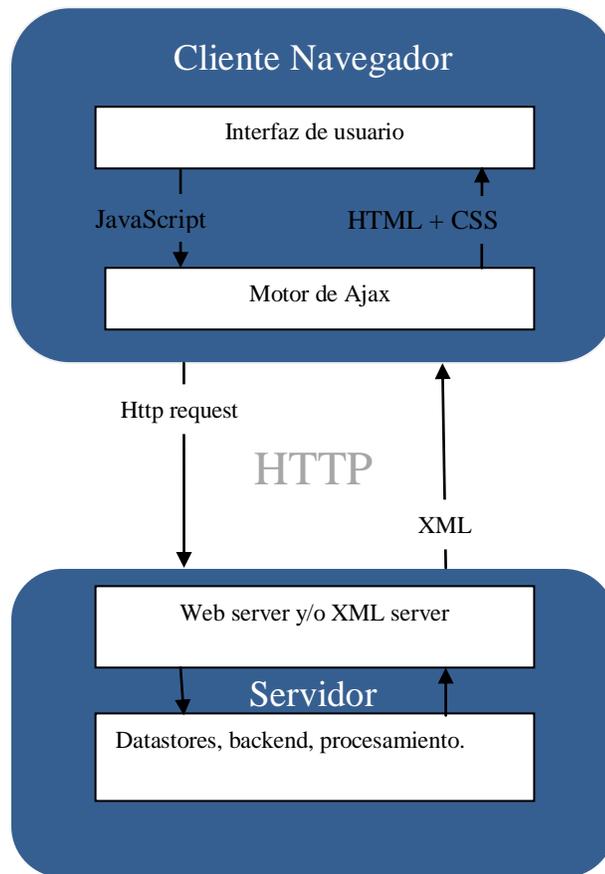


FIGURA 1.5 Modelo de Aplicación Web con AJAX.

	Ventajas	Limitaciones
Usuario	Interacción cliente-servidor no solo después de una completa recarga. Casi todos los navegadores son compatibles.	Es imposible colocar bookmarks sobre las páginas cargadas dinámicamente. Los motores de búsqueda no pueden diferenciar el contenido dinámico y el JavaScript.
Desarrollo	Se incrementa el desarrollo de	Crece la complejidad de

⁴ <http://es.wikipedia.org/wiki/EBML>

	<p>aplicaciones.</p> <p>Se crean equipos de desarrollo para manejar diferente áreas como: arquitectura de información, diseño grafico, etc.</p>	<p>desarrollo.</p> <p>Los costos de desarrollo se incrementan por la implementación de características asincrónicas sobre el diferente estándar de los navegadores.</p>
Respuesta	<p>La comunicación asincrónica, reduce la carga sobre el servidor, para que solo partes de la página sean actualizadas.</p>	<p>No hay conexiones de sockets.</p>

1.2.3.2. Plataforma Flash.

La plataforma Flash crea un ambiente capaz de ejecutar características multimedia con contenidos dinámicos tanto para la Web o como aplicaciones independientes.

Flash provee además características de programación orientada a objetos mediante el lenguaje ActionScript basado en el estándar ECMAScript; actualmente la versión de ActionScript 3 se ejecuta junto con el Flash Player 9 el cual contiene la ActionScript Virtual Machine 2 para procesarlo. En las primeras aplicaciones Flash generalmente toda la carga de procesamiento iba al lado del cliente, para ello era necesario bajar del servidor los archivos Flash que consistían en su mayoría de pequeñas animaciones; no obstante para hacer una aplicación Flash una aplicación RIA, la interacción con el servidor debía ser incluida.

Adobe Flex es un componente servidor que permite a las aplicaciones Flash comunicarse con soluciones de servidor mediante protocolos SOAP

(Simple Object Access Protocol), REST (representational state transfer, transferencia de estado representacional)⁵, o XML vía HTTP. Los controles de Flex están descritos mediante el lenguaje MXML Magic eXtensible Markup Language; y la programación mediante ActionScript 3.

Las soluciones RIA de Flash pueden estar también fuera del navegador mediante el uso de Adobe Integrated Runtime; existen además la plataforma OpenLaszlo que permite una integración con servidores Apache y lenguajes como PHP o CGI; o también con servidores Tomcat y aplicaciones Java; esto ofrece mayor integración de datos como otros Web Services y objetos Java, y conexiones persistentes.

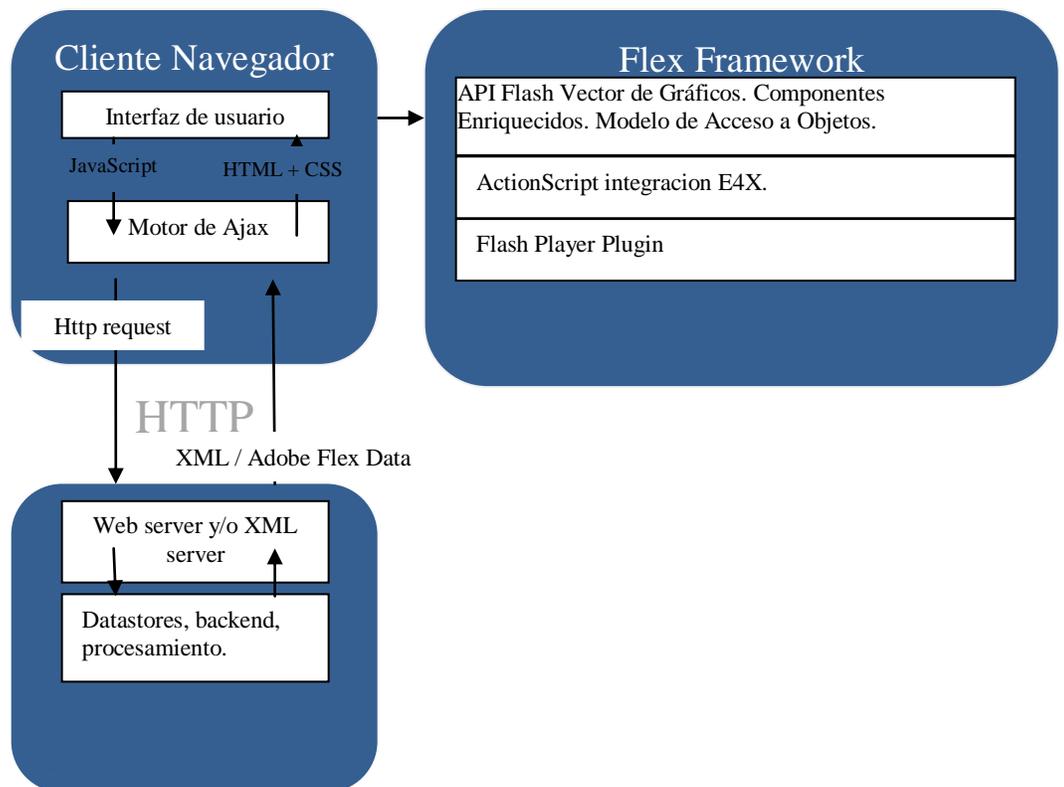


FIGURA 1.6 Modelo de Aplicación Web con Flash.

⁵ http://searchsoa.techtarget.com/sDefinition/0,,sid26_gci823682,00.html

	Ventajas	Limitaciones
Usuario	<p>El plugin es gratis.</p> <p>La aplicación Web ofrece mayor experiencia de usuario que AJAX.</p> <p>Como una aplicación AIR Integrada ofrece buena interacción con el escritorio.</p> <p>La interacción es inmediata por el proceso en el lado del cliente.</p> <p>Excelente reproducción de audio y video.</p>	<p>Es necesario el plugin Flash Player.</p> <p>Los contenidos no pueden ser diferenciados por un motor de búsqueda.</p>
Desarrollo	<p>Fácil de acceder y desarrollar para desarrolladores novatos.</p> <p>Rápidos e impresionantes resultados.</p> <p>Riqueza y estabilidad en las herramientas de desarrollo.</p>	<p>Se requieren de herramientas de desarrollo específicas y además costosas.</p> <p>Limitada interacción con otras plataformas.</p>
Respuesta	<p>Se reduce la carga de ejecución en el lado del servidor.</p>	<p>Ralentización de carga.</p>

1.2.3.3. Otras Plataformas RIA.

Con las plataformas Flash y AJAX se inicia la tendencia de desarrollar aplicaciones web enriquecidas en donde la comunicación con el servidor sea hasta cierto punto imperceptible por parte del usuario; emulando a un aplicación de escritorio.

Es así como algunas plataformas Web tradicionales incrementaron su funcionalidad para desarrollar aplicaciones RIA; entre ellas destacan:

- Java: Java Rich Client, Java Web Smart, Java Applets y JavaFx.
- Adobe Flex, Openlaszlo.
- Microsoft SilverLight.

1.3. EVOLUCIÓN DEL DISEÑO DE PÁGINAS WEB.

La evolución del diseño de sitios web describe el proceso de crear un atractivo y usable sitio web, que compagine la visualización con la funcionalidad y la calidad en el acceso a la información. El diseño web también incluye la arquitectura de la información, la cual especifica como ésta se organiza y se enlaza.

1.3.1. GENERACIONES DE DISEÑO.

Según Sergio Lujan Mora, en su libro Programación de Aplicaciones Web: historia, principios básicos y clientes web, existen cuatro generaciones de sitios web basados en el diseño; esta clasificación igualmente complementa a las tres generaciones descritas por David Siegel en 1996 en su libro Creating Killer Web Sites: The Art of Third-Generation Site Desing⁶.

1.3.1.1. Primera Generación.

Abarca desde nacimiento de la Web en 1992, hasta mediados de 1994, en estas etapas tempranas, los navegadores son desarrollados como proyectos de universitarios, de entre ellos se desataca el NSCA Mosaic de Marc Andreessen y Eric Bina realizado en National Center for Supercomputing Applications. Mosaic llegó en 1993 y fue una gran mejora precisamente para poder ver una imagen en un documento de texto entregado vía Internet.

⁶ <http://gplsi.dlsi.ua.es/~slujan/materiales/pi-cliente2-muestra.pdf>

La construcción de sitios web de la primera generación incluyó un conocimiento básico de HTML, escritura de información organizada, diseño rudimentario de gráficos, algo de programación y típicamente conocimiento de UNIX. Estos sitios fueron simples negocios, algo de texto, pocas imágenes y quizás algunos formularios para interactividad.

Los sitios a menudo consistían de unas pocas páginas y fueron muy experimentales. La tecnología usada para construir estos sitios incluyó HTML, y ocasionalmente simples programas CGI para acceder a los sistemas “back end”. Archivos gráficos en dos dimensiones en formato GIF o pequeños archivos de multimedia accedidos vía alguna aplicación de ayuda.

El HTML y navegadores no soportaban colores de fondo, objetos centrados o alguna proporción del control del esquema de la pantalla como tablas o estilos de hojas. La programación consistía de un código simple, la gran mayoría escrita en Perl.

La siguiente figura describe esta primera etapa:

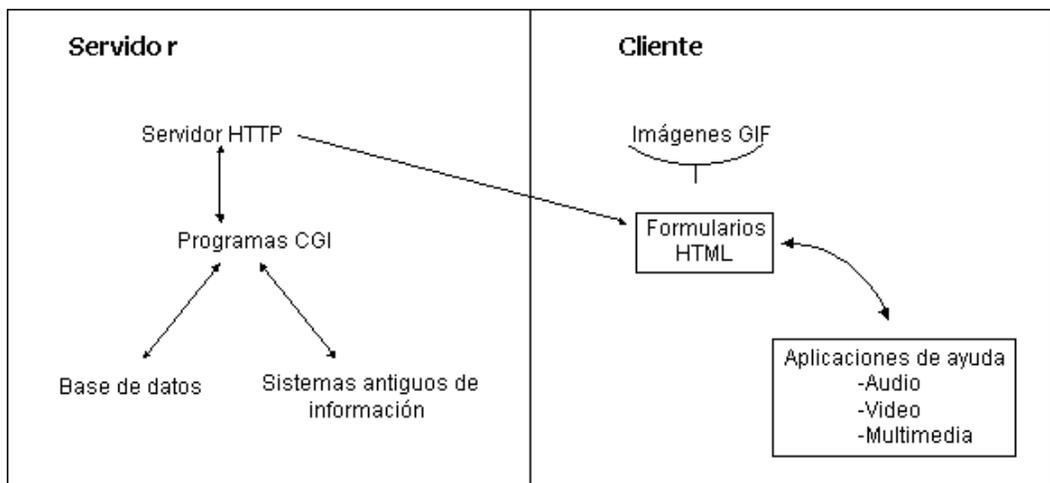


FIGURA 1.7 Primera Generación.

Al inicio las compañías asignaron pequeños o ningún presupuesto para la construcción de los sitios web, más bien estos eran proyectos de los administradores del sistema o directores de marketing en su tiempo libre. Por esto a menudo se podía encontrar constantemente sitios que estaban bajo construcción o completamente abandonados.

1.3.1.2. Segunda Generación.

La segunda generación aparece en 1995 y se considera vigente hasta la actualidad; en ella se prioriza la utilización de gráficos, iconos, animaciones en los diseños de las páginas web llegando en algunos casos a exagerar, lo que provocó que la carga por parte de cliente se ralentice. Aparecen tecnologías multimedia propietarias, que necesitan la instalación de un plug-in (Complemento: Aplicación que se relaciona con otra para aportarle una función nueva y específica)⁷ para su visualización.

En esta etapa empieza también la denominada “guerra de navegadores” entre Netscape Navigator de Netscape Corporation e Internet Explorer de Microsoft, ambos basados en NSCA Mosaic.

Con la introducción de la primera versión de Netscape por el año de 1994, la dinámica del web súbitamente cambió. El navegador de Netscape introdujo nuevas características de HTML, como el centrar, flujo de texto alrededor de gráficos, la propiedad de etiquetas y la posibilidad de escoger fondos de texto. Consecuentemente la presentación de las páginas web mejoró y comenzó a evolucionar.

Junto con Netscape el comercio en el web comenzó a ser más importante. Los diseñadores gráficos estuvieron equipados con nuevas facilidades de diseño HTML, y comenzaron a desempeñar su rol en el web, es en este punto que las conclusiones técnicas y creativas empezaron a divergir.

⁷ [http://es.wikipedia.org/wiki/Complemento_\(inform%C3%A1tica\)](http://es.wikipedia.org/wiki/Complemento_(inform%C3%A1tica))

Los administradores de los sitios web fueron puestos a favor de los diseñadores quienes estaban enmarcados dentro o alrededor de las limitaciones técnicas del web. Los diseñadores gráficos mejoraron la apariencia y la sensación del web pero a menudo a expensas de los aspectos técnicos del sitio. Por esta razón, la mayoría de los sitios fueron mucho más apariencia que funcionalidad.

La siguiente figura describe la segunda etapa de la evolución:

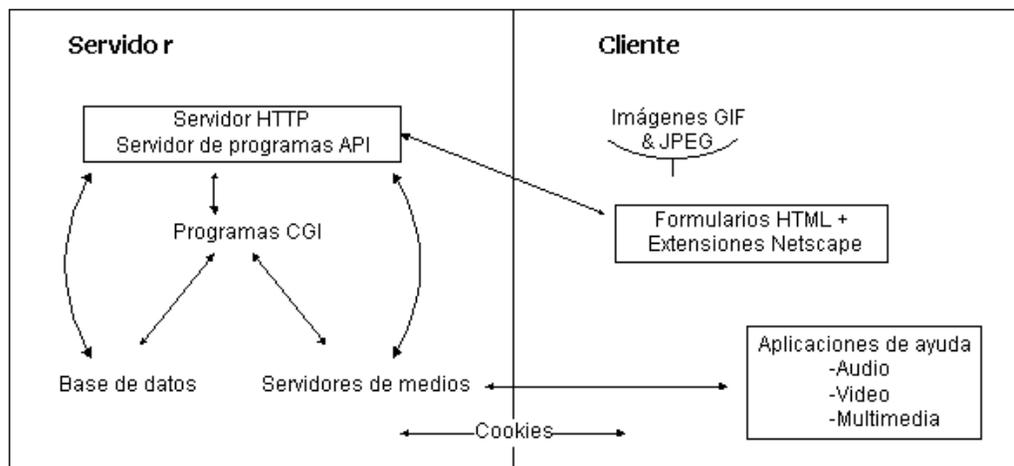


FIGURA 1.8 Segunda Generación.

En la segunda generación muchas personas empezaron en entender como el web podría realmente comunicar, entonces las personas empezaban a visitar los sitios web, y no lo hacían al azar sino que ellos estaban por algún propósito.

La programación del lado del servidor en forma de programas CGI empezó a incrementarse dramáticamente en complejidad. Un obstáculo en la programación de las páginas web fue la falta de preservación del estado o el recordar la información a través de páginas o visitas al sitio, suministrados por los servidores web o los protocolos como el http. La

creación de los cookies ayudó a manejar el problema del estado, pero hasta hoy existen ciertas dificultades.

1.3.1.3. Tercera Generación.

La tercera generación aparece en 1996; se caracteriza porque el énfasis sobre la estructura de la información, los sitios web se crean teniendo en cuenta los posibles usuarios y el objetivo del sitio (informar, vender, entretener, etc.). Se incorporan principios de usabilidad y accesibilidad realizando pruebas con usuarios reales.

En los sitios web de las empresas cobra vital importancia la identidad corporativa, se emplea de forma coherente los iconos, imagen, colores, tipos de letra, etc.

Existen mejoras sobre HTML, se difunden la utilización de Cascade Style Sheets CSS, applets de Java, y JavaScripts para ejecución en el lado del cliente.

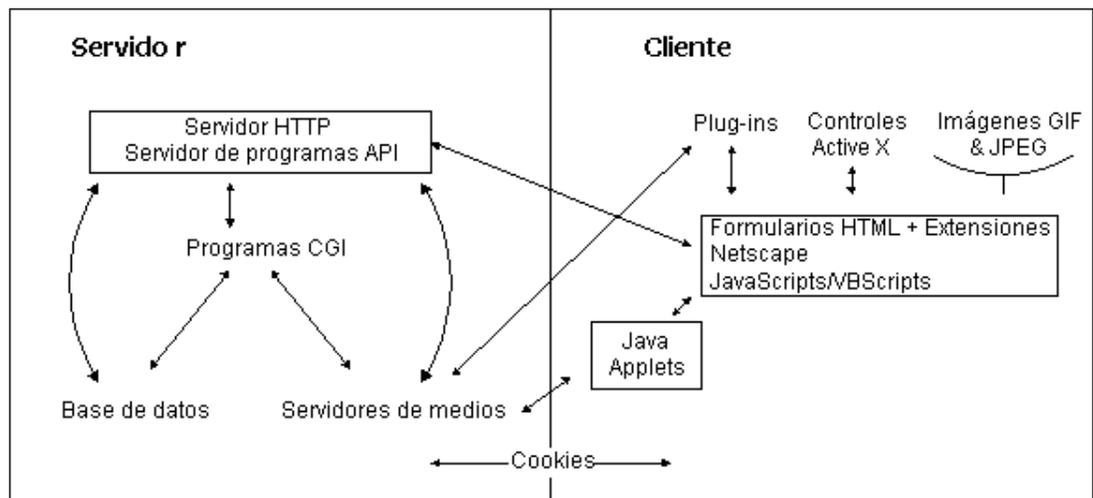


FIGURA 1.9 Tercera Generación.

El modo de generar páginas dinámicas evoluciona, desde la utilización del CGI, Common Gateway Interface, hasta los servlets (es un programa que se ejecuta en un servidor) pasando por tecnologías tipo *JavaServer Pages*. Todas estas tecnologías se encuadran dentro de aquellas conocidas como *Server Side*, ya que se ejecutan en el servidor web.⁸

1.3.1.3.1. Plugins

Existe un grupo de tecnologías orientadas a WWW que ofrecen mayor riqueza de contenidos a los usuarios, en campos como animación, audio y video; estas tecnologías son manejadas a través de plugins o complementos en los navegadores.

Fue Java Applets de Sun Microsystems el pionero en la inclusión de plugins en páginas web en 1995, la temprana intención de Java fue proveer aplicaciones con desempeño similar a las aplicaciones de escritorio a través de Internet, sin embargo su éxito no fue el que Sun Microsystems esperaba. Hacia 1996 se sumo a la tecnología de plugins Flash 1.0 de Macromedia.

Ambas tecnologías, nunca pudieron reemplazar a la tecnología tradicional de hipertexto principalmente porque eran métodos cerrados que los motores de búsqueda de internet no podían interpretar. Sin embargo, el fuerte de la tecnologías enriquecida para Internet era la interactividad con los usuarios a través de una excelente comunicación multimedia con sonidos, gráficos, y complejas visualizaciones; estas características eran ideales para juegos, publicidad, animaciones, etc.

⁸ http://es.wikipedia.org/wiki/Java_Servlet

1.3.1.3.2. CSS.

CSS acrónimo de Cascade Style Sheet u Hojas de Estilo en Cascada, son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML y XHTML, además puede ser aplicado sobre todos los lenguajes extendidos de XML como SVG (scalable vector graphics, describe gráficos bidimensionales vectoriales)⁹ y XUL(lenguaje basado en XML para la interfaz de usuario)¹⁰.

El modo de funcionamiento de las CSS consiste en definir, mediante una sintaxis especial el valor de las propiedades de formato de cada elemento de presentación, creando clases de estilo de las que estos elementos pueden heredar; los estilos de las CSS pueden aplicarse a controles en particular, páginas web o a todo el sitio web.

Utilizar CSS ayuda a mejorar el posicionamiento Web, se logra aumentar la densidad de las palabras clave dentro de los contenidos, ya que muchas de las etiquetas ocuparán muchísimo menos espacio. Esto también supone un menor peso para las páginas Web, lo cual agradecen tanto los robots de búsqueda como los usuarios finales. Adicionalmente se podrá cambiar rápidamente los estilos de ciertas palabras, modificando la importancia que se les quiere otorgar ante los robots de búsqueda.

1.3.1.4. Cuarta Generación.

La cuarta generación aparece en 1999 y existe hasta la actualidad; y se destaca por hacer del sitio web una experiencia multimedia para el usuario.

⁹ http://es.wikipedia.org/wiki/XML-based_User-interface_Language

¹⁰ http://es.wikipedia.org/wiki/XML-based_User-interface_Language

Se extiende el uso de tecnologías nuevas como Dynamic HTML (DHTML), y Flash; se aprovechan las existentes como CSS; la generación de las páginas web es partir de información almacenada en bases de datos.

En este punto, se considera la construcción de un sitio web una tarea de Ingeniería multidisciplinaria en donde participan diversos recursos técnicos como administradores, programadores, diseñadores gráficos, analistas de bases de datos, arquitectos de información, especialistas de marketing, entre otros. (Este tema se trata con mayor profundidad en el capítulo II).

1.3.2.EVOLUCIÓN WEB.

1.3.2.1. Web 1.0

El objetivo primordial de un sitio web, era presentar contenidos informativos, generalmente HTML estáticos, hacia los usuarios; el éxito de los sitios web dependía de webs más dinámicas (a veces llamadas Web 1.5) donde los CMS servían páginas HTML dinámicas creadas desde una base de datos. En ambos sentidos, el conseguir visitas y la estética visual eran considerados como unos factores muy importantes.

En esta etapa el usuario es considerado un cliente y su nivel de participación sobre el contenido es limitado o nulo.

Eventualmente, los nuevos paradigmas que involucran la participación de los usuarios sobre los contenidos de la web disparan la popularidad de los sitios, como por ejemplo Napster.

1.3.2.2. Web 2.0

Web 2.0 es la representación de la evolución de las aplicaciones tradicionales hacia aplicaciones web enfocadas al usuario final. Se trata de

aplicaciones que generen colaboración y de servicios que reemplacen las aplicaciones de escritorio. Es una etapa que ha definido nuevos proyectos en Internet y está preocupada por brindar mejores soluciones para el usuario final.

Aunque el término sugiere una nueva versión de la Web, no se refiere a una actualización de alguna especificación técnica sino a los cambios en la forma en que los desarrolladores y usuarios finales la utilizan, en conclusión es considerada fundamentalmente como una actitud antes que una tecnología.

El término Web 2.0 fue descrito por primera vez por Dale Dougherty de O'Reilly Media quien lo utilizó en una conferencia en la que compartió una lluvia de ideas junto a Craig Cline de MediaLive, esto dio la pauta para que se realizara la primera conferencia sobre Web 2.0 en San Francisco en 2004, en dicha conferencia Tim O'Reilly expresó:

“Web 2.0 is the business revolution in the computer industry caused by the move to the Internet as platform, and an attempt to understand the rules for success on that new platform.” Tim O'Reilly (2006-12-10). "Web 2.0 Compact Definition: Trying Again". Retrieved on 2007-01-20.”¹¹

“Web 2.0 es la revolución de los negocios en la industria informática causada por la utilización del internet como plataforma y un intento para entender las reglas del éxito sobre esta nueva plataforma”

En la misma conferencia se habló de los principios que tenían las aplicaciones Web 2.0:

¹¹ <http://oreilly.com/web2/archive/what-is-web-20.html>

1.3.2.5. Cuadro Comparativo de Web1.0 y Web 2.0

El siguiente cuadro muestra las aplicaciones que pertenecen a la Web2.0, con su antecesor o equivalente basadas en paradigmas anteriores, con similares características de servicio.

Característica	Ejemplo.
Una aplicación ofrece su servicio vía on-line en lugar de utilizar paquetes instalables.	Google Documents ofrece el procesamiento en línea de texto, contrariamente Microsoft Office necesita de un CD de instalación.
Los usuarios contribuyen a mejorar y actualizar el contenido de sitio.	Wikipedia, basa su contenido en el conocimiento de los usuarios, el crecimiento de este conocimiento es diario; Microsoft Encarta actualiza su contenido en cada versión del paquete.
Confianza a los usuarios como co-desarrolladores.	Todo el software en Sourceforge.net
Fomentación de las redes sociales orientadas a Internet.	Foros, bloggers, y wikis.
Aplicaciones web sobre diferentes dispositivos.	Palms, celulares, consolas de juegos, etc.

1.3.2.6. WebTop.

El constante crecimiento de la Web2.0 junto con las redes de comunicación de banda ancha le ha dado un nuevo auge a las tecnologías enriquecidas. La tendencia lleva a depender cada vez menos del navegador y desarrollar aplicaciones con riqueza multimedia que sean

ejecutadas por los usuarios en un entorno independiente, ya sea, de la plataforma del Sistema Operativo o del navegador.

Aparece entonces el término Webtop, para definir a una aplicación que corre como un programa de escritorio y utiliza tecnología RIA; estas aplicaciones ofrecen un soporte multimedia avanzado para una interactividad enriquecida.

Teóricamente las aplicaciones Webtop pueden reemplazar a las aplicaciones tradicionales de escritorio, como ya ha ocurrido con software como iTunes.



FIGURA 1.11 Ejemplo de WebTop.

1.3.3. INTRODUCCIÓN A LAS RICH INTERNET APPLICATION.

La evolución Web conduce a la creación de las Rich Internet Application [Aplicaciones Enriquecidas para Internet], este término, fue utilizado por primera vez por Jeremy Allaire en 2002 a razón del programa Macromedia Flash.

Las soluciones RIA pueden solventar algunas deficiencias de las aplicaciones web tradicionales y asemejarlas a una aplicación de escritorio, tanto en visualización, diseño, rendimiento, multimedia y comunicación. (Este tema se trata con mayor profundidad en el capítulo II).

1.4. ARQUITECTURA DE LA INFORMACIÓN.

1.4.1. DEFINICIÓN.

El término "Arquitectura de la Información" (AI) fue utilizado por primera vez por Richard Saul Wurman en 1976, quién la define como:

“El estudio de la organización de la información con el objetivo de permitir al usuario encontrar su vía de navegación hacia el conocimiento y la comprensión de la información”¹⁴

Si se refiere exclusivamente a la AI en el campo de la Web, una de las definiciones que Louis Rosenfeld y Peter Morville ofrecen en su libro "Information Architecture for the World Wide Web 2nd Edition", puede ser de más fácil comprensión:

“El arte y la ciencia de estructurar y clasificar sitios web e intranets con el fin de ayudar a los usuarios a encontrar y manejar la información.”¹⁵

El concepto "Arquitectura de la Información" no solo engloba la actividad de organizar información, sino también el resultado de dicha actividad.

¹⁴ <http://sociologiac.net/2008/05/29/richard-saul-wurman-y-la-arquitectura-de-la-informacion/>

¹⁵ <http://www.artedinamico.com/articulo/106/5>

Como disciplina no busca definir una metodología de diseño universal sino articular un conjunto de técnicas para ayudar al desarrollo y producción de espacios de información como los sitios web.

1.4.2.LA ARQUITECTURA DE LA INFORMACIÓN DE UN SITIO WEB.

1.4.2.1. Objetivos

Con el fin de que la asimilación de contenidos por parte del usuario sea eficiente y efectiva, y para que el sitio sea accesible y usable, la Arquitectura de la Información como proceso en general, se encarga, durante el desarrollo de definir:

- El objeto, propósito y fines del sistema de información o sitio.
- La definición del público objetivo y los estudios de la audiencia.
- La realización de análisis competitivos.
- El diseño de la interacción.
- El diseño de la navegación y esquemas de facetas.
- El etiquetado o rotulado de los contenidos para acceder a la información.
- La planificación, gestión y desarrollo de contenidos.
- La facilidad de búsqueda y el diseño de la interfaz de búsqueda.
- La usabilidad.
- El feedback del resultado y los procesos de reingeniería del sitio.

1.4.2.2. Organización de la Información en un sitio Web.

La organización de la información en el contexto del desarrollo de sitios web está estrechamente relacionada con la navegación, el etiquetado y la indización.

Los sistemas de organización están compuestos por: esquemas de organización y estructuras de organización.

1.4.2.2.1. Esquemas de Organización de la Información. EOI

Un esquema de organización delimita las características compartidas de los elementos que integran el contenido y las influencias del agrupamiento lógico de estos elementos.

El Objetivo de los EOI es dividir la información en secciones bien definidas y mutuamente excluyentes; existen diferentes tipos EOI según la manera en que organizan la información.

- EOI Exactos, son relativamente fáciles de diseñar y mantener debido a que demandan de un pequeño esfuerzo intelectual para asignar elementos a las categorías.
- Esquemas alfabéticos de organización de la información (EAOI).
- Esquemas cronológicos de organización de la información (ECOI).
- Esquemas geográficos de organización de la información (EGOI).
- EOI ambiguos, dividen la información en categorías que proponen definiciones exactas. Están impregnados por la ambigüedad del lenguaje y de la subjetividad humana. Entre los EOI ambiguos, los más comunes son:
 - Los que organizan la información según el asunto o la materia.
 - Los que organizan la información atendiendo a determinados procesos o tareas.
 - Los que organizan la información según los intereses de la audiencia potencial.
 - Los que organizan la información a través de metáforas de la vida cotidiana.
 - Los esquemas híbridos.

1.4.2.2.2. Estructuras de Organización de la Información. ESOI.

Una estructura de organización define las relaciones entre los elementos de contenidos y los grupos de estos elementos.

Las estructuras de organización de la información son intangibles pero muy importantes en el diseño de un sitio web. Frecuentemente, se interactúa con estructuras de organización pero se tiene poca conciencia de ello en un buen número de casos. Las ESOI definen los caminos primarios sobre los cuales los usuarios podrán navegar.

Las principales ESOI aplicadas a sitios web incluyen: la jerarquía, los modelos orientados a bases de datos y el hipertexto. Cada una tiene sus fortalezas y debilidades; en algunos casos, se usa una o la otra, pero, la mayor parte de las veces, lo más aconsejable es utilizarlas todas para que se complementen entre ellas.

- Jerarquías. La cualidad de las estructuras jerárquicas de brindar una manera simple y familiar de organizar la información permite al usuario sentirse cómodo, porque puede hacerse una idea de la estructura del web sin necesidad de adentrarse en una extensa exploración de los contenidos.
- Hipertexto. El hipertexto es una vía no lineal de estructurar la información. Un sistema de hipertextos contiene dos componentes primarios: los elementos o nodos de información que se enlazarán y los enlaces o vínculos entre estos. Los nodos de hipertextos pueden conectarse de manera jerárquica, no jerárquica o en ambas. Esta estructura de organización proporciona una gran flexibilidad, no obstante puede provocar que los usuarios al navegar por ella se desorienten o pierdan. Al

navegar entre hipervínculos no es posible representarse un modelo mental de la estructura del sitio web.

- **Modelo de Bases de Datos.** En forma muy simple, puede definirse una base de datos como una colección de registros donde cada registro contiene un número determinado de campos asociados. Una de las mayores potencialidades de utilizar una ESOI basada en bases de datos es que se puede brindar la posibilidad de buscar elementos de contenido en campos específicos o varios campos a la vez con un solo intento de búsqueda. Sin embargo, las ESOI basadas en bases de datos tienen sus limitaciones, por ejemplo los registros deben cumplir con reglas estrictas de integridad, cada registro debe tener los mismos campos y cada campo debe contener el mismo tipo de dato (numérico, carácter, booleano, etc.).

1.4.2.3. Sistemas de Navegación. SN.

La razón para diseñar correctamente un sistema de navegación (SN) radica en prevenir que los usuarios puedan hallarse perdidos frente al sitio web.

En el web, un EOI jerárquico bien diseñado evitará la desorientación de los usuarios. De manera complementaria, el diseño de un SN es necesario para brindar un sentido del entorno y dar flexibilidad al movimiento dentro del sitio.

Debe prestarse especial atención al balance que debe existir entre la flexibilidad en la navegación y la inclusión de excesivas opciones que podrían confundir y desorientar al usuario.

Los SN pueden conformarse por varios elementos como son: gráficos, barras de menús horizontales y verticales y menús emergentes. Otros, como las tablas de contenido y el mapa del sitio, permiten el acceso remoto al contenido alojado en el EOI. Cada uno de estos elementos debe aparecer en cada página y la suma de todos ellos conforma el SN.

Existen varios tipos de SN que pueden incluirse en un mismo diseño. Y pueden identificarse cuatro:

- Jerárquicos: Ofrece acceso a los diferentes niveles jerárquicos inferiores a partir de la página inicial.
- Globales: Brinda la posibilidad de navegación tanto a lo profundo como a lo largo del sitio - navegación vertical y horizontal.
- Locales: Se utilizan para sitios complejos, generalmente divididos en sub-sitios o categorías.
- Específicos: Utilizados cuando la complejidad del sitio web impide la clasificación de las páginas.

1.4.3.USABILIDAD.

La usabilidad (usability, el término en inglés), emerge desde raíces interconectadas con factores y disciplinas como: gráfica computarizada, interfaces humanas, procesos cognitivos, ingeniería industrial, entre otros. La gráfica computarizada nació del uso del tubo de rayos catódicos y dispositivos cerrados, desde épocas tempranas en la historia de las computadoras. Esto motivó el desarrollo de varias técnicas de interacción hombre - máquina.

La usabilidad para el web surge a partir del nacimiento y desarrollo de Internet como red de comunicación. El surgimiento de lenguajes de programación para el web ha potenciado la implementación de prestaciones y funcionalidades que posibilitan la creación de interfaces más poderosas, por tanto más

complejas de asimilar y utilizar por parte del usuario. Esta razón motivó la adecuación de los aspectos tradicionales de la usabilidad a la nueva tecnología del web.

Pretender que una aplicación o sitio web sea usable independientemente de quién y cómo la use, corresponde más con una visión o enfoque universalista de la usabilidad (en ocasiones necesaria), que con una visión realista y práctica. Esto es debido a que normalmente toda aplicación se diseña con la intención de satisfacer las necesidades de una audiencia concreta y determinada, por lo que será más usable cuanto más adaptado esté su diseño a esta audiencia específica y por tanto menos lo esté para el resto de personas.

1.4.4. ACCESIBILIDAD.

Un concepto íntimamente ligado al de usabilidad es el de accesibilidad. Éste ya no se refiere a la facilidad de uso, sino a la posibilidad de acceso. En concreto a que el diseño, como prerequisite imprescindible para ser usable, posibilite el acceso a todos sus potenciales usuarios, sin excluir a aquellos con limitaciones individuales - discapacidades, dominio del idioma, o limitaciones derivadas del contexto de acceso, software y hardware empleado para acceder, ancho de banda de la conexión empleada, etc.

Se da la paradoja de que mientras que un diseño usable requiere delimitar a su audiencia potencial con el fin de diseñar para lo concreto, un diseño accesible implica la necesidad de diseñar para la diversidad y heterogeneidad de necesidades de acceso presentadas por esta audiencia específica.

1.4.5. LA ACCESIBILIDAD EN LAS RIA SEGÚN LA W3C.

La W3C estableció la normativa WAI-ARIA para definir los estándares para las Rich Internet Applications.

WAI-ARIA, la Normativa de Accesibilidad para las RIA, define una manera para hacer que el contenido web y las aplicaciones web sean más accesibles para personas con discapacidad. Esto ayuda especialmente con contenidos dinámicos y avanzadas interfaces de usuarios desarrolladas con AJAX, HTML, JavaScript y demás tecnologías relacionadas.

Actualmente ciertas funcionalidades de los sitios web no están disponibles para usuarios con discapacidad. WAI-ARIA se ocupa de estos problemas de accesibilidad, por ejemplo, mediante la definición de nuevas formas de funcionalidad que debe proporcionarse a la tecnología de asistencia. Con WAI-ARIA, los desarrolladores pueden hacer aplicaciones web avanzadas accesibles y utilizables a las personas con discapacidad.

Más concretamente, WAI-ARIA proporciona un marco para añadir los atributos para identificar las características de la interacción del usuario, cómo se relacionan unos con otros, y su estado actual. WAI-ARIA describe las nuevas técnicas de navegación para marcar las regiones y las estructuras comunes de la web como menús, contenido primario, secundario contenido, banner información, y otros tipos de estructuras Web. Por ejemplo, con el WAI-ARIA, los desarrolladores pueden identificar regiones de las páginas y permitir que los usuarios puedan moverlas fácilmente con el teclado, en lugar de tener que pulsar TAB muchas veces.

WAI-ARIA, también incluye tecnologías para mapear controles, conocida como AJAX Live Regions, y eventos de accesibilidad a API, incluyendo los controles para las aplicaciones de RIA. Las técnicas WAI-ARIA se aplican a los controles [artilugios], tales como botones, listas desplegadas, funciones de calendario, árbol de controles (por ejemplo, menús expandibles), y otros.

1.4.6. SISTEMA DE GESTIÓN DE CONTENIDOS.

Un Sistema de gestión de contenidos (Content Management System en inglés, abreviado CMS) es un programa que permite crear una estructura de soporte para la creación y administración de contenidos por parte de los participantes.

La principal ventaja de un CMS es que los sitios web creados con él, proporcionan alta interactividad a los visitantes.

Además de ser profesionales, permiten administrar todo a través de Internet. Esta gestión online significa que para actualizar la página web, sólo se necesita un PC conectado a Internet. La facilidad para agregar información al sitio, convierten al CMS en una poderosa herramienta de difusión de información, marketing online y de negocios en la web. La fortaleza del CMS radica en su facilidad de uso. Y es que una vez instalado y configurado, puede comenzar a funcionar en Internet.



FIGURA 1.12 Interacción CMS.

II. Capítulo II

TECNOLOGÍA RICH INTERNET APPLICATION RIA

2.1. CONCEPTO: RIA.

Rich internet applications o aplicaciones de internet enriquecidas. Son un nuevo tipo de aplicaciones con más ventajas que las tradicionales aplicaciones Web. Surge como una combinación de las ventajas que ofrecen las aplicaciones Web y las aplicaciones tradicionales de escritorio. RIA ofrece lo mejor de la mezcla entre las ricas disponibilidades de las aplicaciones y de la riqueza del contenido.

Se debe pensar en las soluciones tecnológicas en términos de dos características: **Riqueza** habilidad para incorporar interactividad e interfaces de usuario intuitivas en el cliente y **alcance** es la habilidad de la aplicación para estar disponible para cualquier usuario en el lugar del planeta en el que éste se encuentre.

Las RIAs combinan las riquezas ilimitadas de las aplicaciones Web, el alcance extensivo, la facilidad de distribución y mantenimiento, junto con la familiaridad de las eficientes e interesantes interfaces que permiten mejores experiencias para el usuario final de las aplicaciones de escritorio. Además, de las ventajas de la multimedia, con sus múltiples tipos de letra, gráficos vectoriales, animaciones, conferencias on-line, audio y video. Esto genera la riqueza mencionada.



FIGURA 2.1 RIA.

Las aplicaciones de internet enriquecidas proveen al usuario final una interfaz que es más intuitiva que las aplicaciones tradicionales. Muchos frameworks usados para crear estas aplicaciones incluyendo Microsoft Silverlight y Flex, brindan más proceso al cliente en vez de dejar centralizado todo en el servidor. Con esto el navegador no intercambia largos y monolíticos bloques de información. Significa que solo las piezas relevantes de la interfaz necesitan ser actualizadas, permitiendo a los usuarios hacer más cosas y más rápido, que en las aplicaciones tradicionales.

2.1.1. DEFINICIÓN DE RIA POR MACROMEDIA

Son aplicaciones que combinan lo mejor de la funcionalidad de la interfaz de usuario de las aplicaciones de escritorio, con el ancho, rico y barato uso de las aplicaciones Web con lo mejor de la interactividad, de la multimedia y de las comunicaciones. Resultado final: una aplicación que provee una experiencia de usuario más intuitiva, receptiva y efectiva.

Específicamente, lo mejor de las aplicaciones de escritorio son capacidades como: es el proveer de una interfaz de usuario para validaciones y formatos, siendo una interfaz rápida responde muchas veces sin actualizar o recargar páginas, además de los comportamientos de las interfaces comunes de usuario como halar y soltar (drag and drop) y la habilidad de trabajar conectado y desconectado. Lo mejor de la Web incluye capacidades como: uso instantáneo, es decir, sin instalaciones, son multiplataforma, tiene el uso progresivo de descargas para recuperar contenido e información. Lo mejor de las comunicaciones significa incorporar audio y video interactivo.

Con todo esto en una RIA, un cliente está habilitado para hacer más cosas que solo desplegar páginas. Está habilitado para ejecutar cálculos, enviar y recibir información en Background (en segundo plano, sin interactividad humana)¹⁶ de forma asíncrona con peticiones de usuario, redibujar secciones

¹⁶ [http://en.wikipedia.org/wiki/Background_\(computer_software\)](http://en.wikipedia.org/wiki/Background_(computer_software))

de la pantalla, usar audio y video de una forma fuertemente integrada, independientemente del servidor al cual esté conectado.

2.2. APLICACIONES RIA FRENTE A APLICACIONES WEB TRADICIONALES.

El diferenciador más importante entre una RIA y un sitio Web convencional, es que una RIA es una aplicación real que permite al usuario realizar tareas. Estas tareas incluyen: encontrar un producto, personalizar permisos, añadir o modificar información, jugar un juego.

Las aplicaciones web tradicionales basan todo el procesamiento de datos en el servidor, utilizando al cliente (Navegador) solo para mostrar la información generada, teniendo que redibujar todo el contenido de la página entre una interacción y otra, aun así sea un cambio mínimo. El pasar toda la interacción a través del servidor es una gran desventaja al perder velocidad de procesamiento.



FIGURA 2.2 Aplicación Web Tradicional.

Mientras que en las RIAs solo es necesario cargar la interfaz una sola vez, haciendo pedidos específicos de información al servidor sin la necesidad de recargar nada, solo mostrar la respuesta del servidor, reduciendo así el tiempo de espera del usuario y teniendo la posibilidad de integrar una reacción (Feedback) para que el usuario este enterado de que está pasando en la aplicación en cada momento.

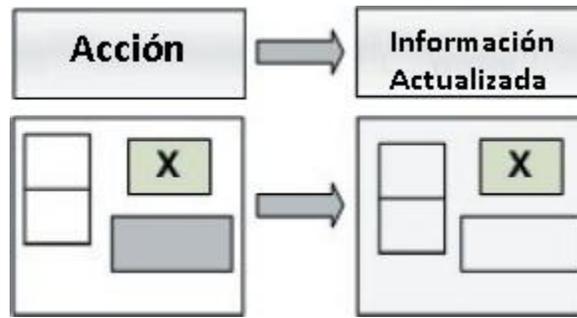


FIGURA 2.3 Aplicación RIA.

Otro beneficio de las RIAs sobre las aplicaciones web tradicionales es sin duda la Interfaz gráfica del usuario (GUI), ya que en las RIAs se puede agregar, eliminar y modificar elementos sin tener que recargar la vista actual.

Una característica común entre todas las RIAs, es que estas han introducido una capa intermedia de código entre el usuario y el servidor, que es usualmente conocida como Cliente Engine (Motor del cliente). Esta actúa como una extensión del servidor y es descargado generalmente al inicio de la aplicación, puede ser complementada por descargas adicionales de código según el progreso de la aplicación. El motor del cliente usualmente toma la responsabilidad de desplegar la interfaz de usuario de la aplicación y de la comunicación con el servidor. Además, ejecuta comunicaciones asíncronas con el servidor.

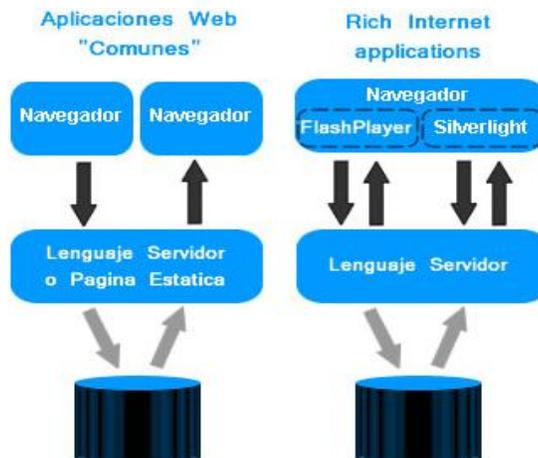


FIGURA 2.4 Aplicación Web Tradicional frente a RIA.

Usualmente las RIAs se ejecutan en un Navegador Web pero pronto también lo harán sin necesidad de este, no necesitan ser instaladas y además, corren localmente en un SandBox (ambiente virtual o de testeo que aísla los cambios en el código no testeado) que provee un ambiente seguro.

2.3. VENTAJAS Y RESTRICCIONES.

2.3.1. ALGUNOS BENEFICIOS SOBRE LAS APLICACIONES WEB TRADICIONALES

- Una RIA es usualmente más rica en funcionalidad también pueden ofrecer interfaces de usuario que usen solo herramientas HTML pero pueden incluir cualquier tecnología usada por el lado del cliente, incluyendo Drag and Drop (arrastrar y soltar), Slider (deslizador) para cambiar información, cálculos ejecutados solo por el cliente sin necesidad del servidor.

- Los comportamientos de interfaz son usualmente más responsivos en comparación a los de una Web estándar y también pueden generar otros beneficios de rendimiento cuando se usa motores de cliente, tales como:
 - Es capaz de hacer un mejor balance entre el cliente y el servidor liberando recursos del servidor permitiendo al mismo hardware del servidor manejar más sesiones de clientes concurrentemente.

 - Es también capaz de realizar una comunicación asíncrona sin esperar al usuario para ejecutar una acción de interfaz como dar clic en un botón o vínculo, de esta manera los diseñadores RIA se sienten libres de mover información entre el cliente y el servidor sin hacer que el usuario espere. Además como ya se ha explicado anteriormente el prefetch (pre-obtener)¹⁷ es la aplicación más común, en la cual una aplicación anticipa una necesidad futura de

¹⁷ <http://en.wikipedia.org/wiki/Prefetching>

información y descarga esta al cliente antes de que el cliente la pida, un ejemplo de esto se lo puede ver en los mapas Google que descargan partes de mapas adyacentes antes que el usuario los recorra.

- Hay otro beneficio en términos de eficiencia de red es que se reduce el tráfico de forma significativa en las RIAs ya que la aplicación mediante el motor del cliente es más inteligente que un navegador Web estándar al momento de decidir qué información necesita intercambiar con el servidor. Esto aumenta las peticiones y respuestas, debido a que hay menos transferencias de información por cada interacción, así que la recarga en toda la red se reduce.
- Las RIAs permiten a las empresas remover la complejidad de las transacciones a través de las interfaces que son como de escritorio y los procesos intuitivos. Las tareas que se pueden realizar son:
 - Desarrollar nuevos tipos de aplicaciones con características o capacidades que antes eran imposibles o difíciles de desarrollar usando las tecnologías tradicionales.
 - Guiar y escuchar a sus consumidores en línea íntimamente y más cercanamente para incrementar en ellos la lealtad a la empresa, mejorar el servicio, profundizar la relación con el cliente, distinguir la compañía, guiar en el desarrollo de productos.
 - Crear fascinantes y atractivos sitios Web usando audio, video, texto y gráficos que generan ventajas, incrementan las ventas, simplifican la comunicación y crean una única experiencia en línea (on Line) que vale la pena y hace que los clientes deseen regresar.

- Simplificar procesos típicos como el de registro, configuración o envío, incrementando ventas, el tiempo y repetición de visitas.
 - Presentar información a todos de una forma limpia, innovadora, intuitiva y efectiva para incrementar la productividad, compartir información, la toma de decisiones y la ventaja competitiva.
 - Provee una altamente interactiva capa de presentación que se conecta a los servicios Web.
 - Reduce los costos de ancho de banda asociados con las recargas frecuentes ahora con los sitios Web de mejor tráfico.
 - Incrementa las ventas dramáticamente de los productos y servicios a través del canal de internet.
 - Construir una RIA reduce costos comparado con el uso de tecnologías Web alternativas.
- Las expectativas de los usuarios continúan creciendo debido a los avances tecnológicos. Ya no es aceptable tener que recorrer cuatro páginas web para hacer una compra en línea. Si tanta traba es encontrada a lo largo de la transacción el usuario la terminará y buscará otro sitio donde el proceso se ejecute sin problemas.
 - Las RIAs eliminan las multipáginas para los múltiples pasos en las transacciones presentando la información pertinente a los usuarios sin dejar el ambiente inicial. Esto simplifica el proceso para los usuarios, quienes continúan con el proceso de compra y regresan porque la experiencia fue rápida, fácil y disfrutable.

2.3.2. BENEFICIOS EN EL ÁMBITO DE TECNOLOGÍAS DE INFORMACIÓN (TI)

2.3.2.1. Mejorar la productividad de los desarrolladores

Maximiza la productividad de todo el equipo de desarrollo a través de herramientas accesibles, plantillas pre-construidas y componentes y de un desarrollo de aplicaciones rápido usando lenguajes script de cliente y de servidor.

2.3.2.2. Aprovecha el ambiente de TI existente

Tiene un soporte nativo para liderar la industria de los estándares de internet, como con Java y .Net reduce tiempos y riesgos en el desarrollo.

2.3.2.3. Costos de operación menores

Como ya se ha explicado anteriormente, reduce el uso del ancho de banda y de la carga en el servidor, moviendo el procesamiento al navegador Web, resultando en menores pedidos de servidor y transferencias de datos comprimidos.

2.3.3. RESTRICCIONES

A pesar de tener una gran ventaja por encima de las aplicaciones Web estándar, Las RIAs tienen algunos defectos y restricciones.

2.3.3.1. SandBox

Como las RIAs se ejecutan en un ambiente virtual, se necesita una correcta operación del SandBox para que una RIA se ejecute correctamente. Si el acceso a recursos esta incorrecto, la RIA fallará para operar con recursos del sistema.

2.3.3.2. Scripting Deshabilitado

Para correr una RIA cualquier lenguaje script es esencial, en el caso de que esté deshabilitado el lenguaje script en el navegador, no habrá funcionamiento de la RIA.

2.3.3.3. Velocidad de procesamiento en el Cliente

Algunas RIA usan scripts del lado del cliente escritos en lenguajes interpretados que moderan la velocidad de rendimiento, mientras que el lenguaje compilado del lado del cliente en las aplicaciones tradicionales no tiene relación con la velocidad.

2.3.3.4. Tiempo de descarga del Script

Si el usuario descarga cualquier documento, este tiene que ser transferido por lo menos una vez desde el Sitio Web hasta la memoria caché del sistema. Aunque el documento descargado no necesite ser instalado en el sistema todavía este toma algún tiempo inesperado. Los desarrolladores RIA pueden reducir el tiempo de espera comprimiendo los Scripts y repartiendo su envío sobre múltiples páginas de una aplicación.

2.3.3.5. Pérdida de visibilidad para los motores de búsquedas

Los motores de búsqueda no están habilitados para indexar el contenido de texto de las RIA.

2.3.3.6. Dependencia de una conexión a Internet

Como las aplicaciones Web tradicionales, también las RIAs necesitan conexión a internet, la velocidad de operación de una RIA depende de la conexión de red. Una conexión de red ideal es usualmente apropiada para que una RIA se ejecute sin problemas, en otros casos esto será un dolor de cabeza.

2.3.3.7. Complicaciones de Administración

Las aplicaciones Web Tradicionales son más simples porque solo tiene un formato de construcción estándar HTML, mientras que una RIA es más difícil de manejarla. La complejidad adicional de una RIA la hace más difícil de diseñarla, testearla, medirla o darle soporte. Estas

complicaciones alargan el proceso de desarrollo de Software, a pesar de la particular metodología empleada. Debido al lento procesamiento este se vuelve difícil de testarlo y las pruebas incompletas hacen que la aplicación tenga una menor calidad y fiabilidad.

Claro que ahora han nacido nuevas herramientas que ayudan en estos procesos tales como VS 2008, Silverlight, entre otras que obviamente necesitan un mejor computador y más memoria por su alto contenido gráfico y de procesamiento.

Una RIA puede ser una Aplicación basada en la Web que es lenta al perder mucho tiempo real de información, al ser construida para varias plataformas o dispositivos en los cuales no siempre se la va a usar.

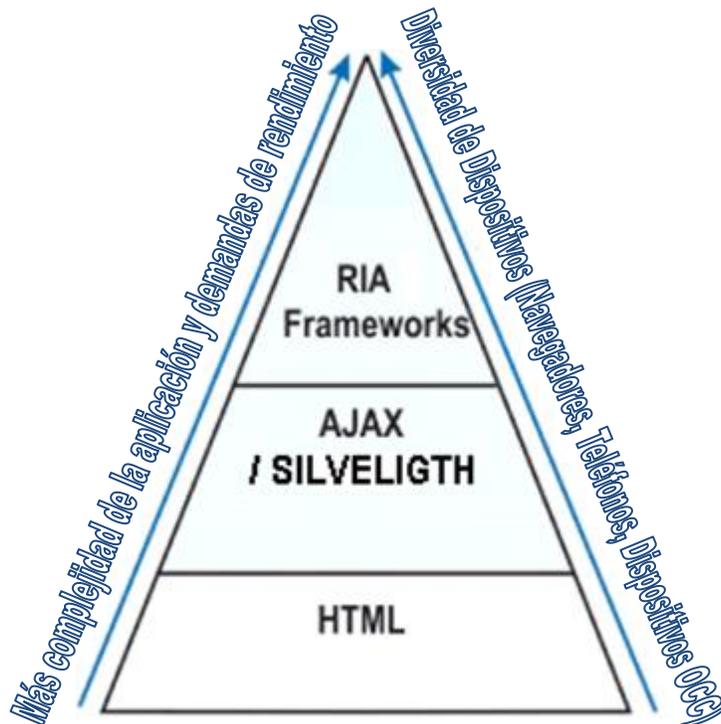


FIGURA 2.5 Complicaciones de Administración.

2.4. ARQUITECTURA RIA



FIGURA 2.6 Arquitectura RIA.

2.4.1. CONEXIONES CON EL SERVIDOR DE BASE DE DATOS

Las conexiones con las fuentes de datos son una parte fundamental de la capa de datos. El crear y manejar conexiones usa recursos valiosos en ambas capas, la de datos y la de la fuente de los datos. Para maximizar el rendimiento se debe seguir estos aspectos:

- Se deben abrir conexiones lo más tarde posible y cerrarlas lo más pronto posible.
- Ejecutar transacciones a través de una sola conexión si es posible.

- Usar transacciones que permitan retroceder o rehacer acciones en el caso de pérdidas de conexión.

Las RIA deben usar el modelo de llamadas asíncronas para servicios, con el fin de evitar bloqueos en los procesos del navegador. Hay que asegurarse de que se usan aplicaciones multi-plataforma. Hay que considerar usar sockets, para poner la información en el servidor ya que es más eficiente que usar servicios.

2.4.2. ACCESO A DATOS

Las implementaciones de acceso a datos se dan en forma similar a las aplicaciones Web. Las RIAs deben pedir datos del servidor Web a través de servicios al igual que lo hace un cliente AJAX. Después de que los datos lleguen al cliente, este puede ser puesto en memoria caché para maximizar el rendimiento.

- Hay que minimizar el número de idas y regresos al servidor, mientras se provee una respuesta en la interfaz de usuario
- Filtrar la información en el servidor en vez de en el cliente, para reducir la cantidad de información que debe ser enviada en la red.}

Diseñar una aplicación para usar una capa de acceso a datos de forma separada es importante para mantenimientos y extensibilidad. La capa de acceso a datos debería ser responsable de administrar las conexiones y ejecutar comandos con la fuente de datos.

- Hay que esforzarse por tener integridad en la información en la base de datos y no a través del código en la capa de datos.

- Hay que mover el código que toma decisiones de negocio a la capa de negocio.
- Evitar el acceso a la base de datos directamente desde diferentes capas en la aplicación, en vez de eso toda la interacción con la base de datos debería estar hecha a través de la capa de acceso a datos.

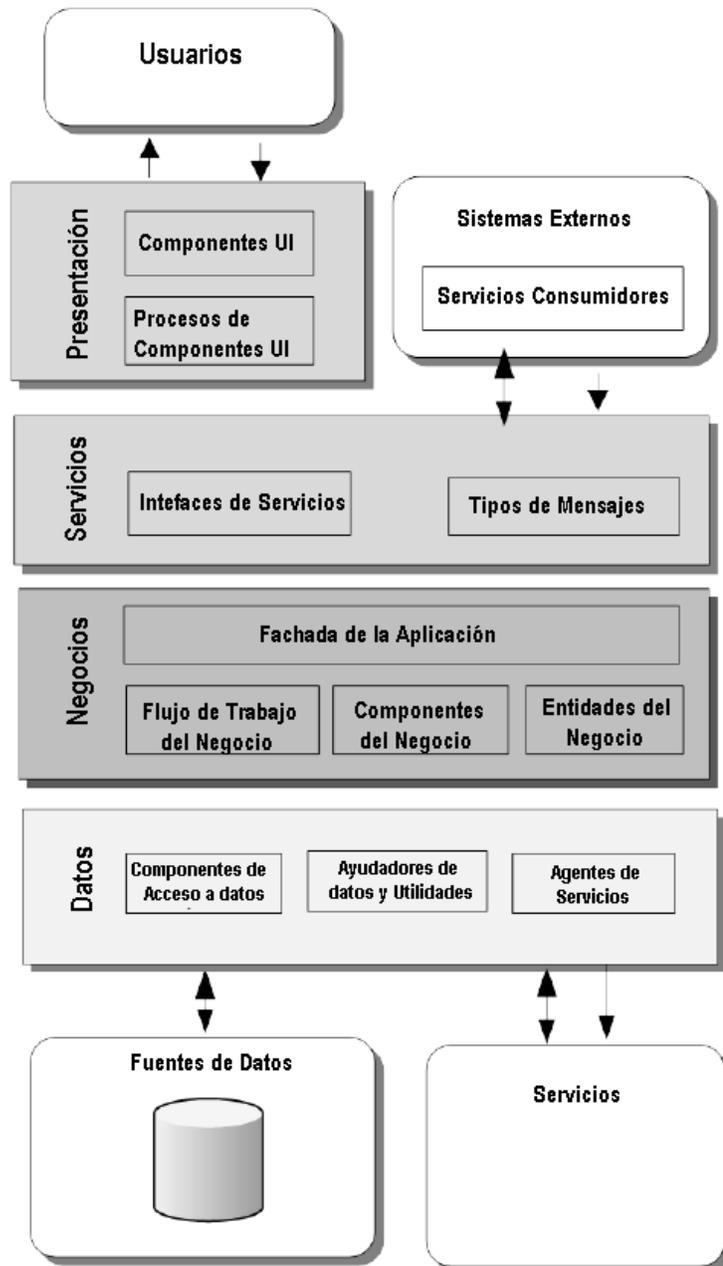


FIGURA 2.7 Capas de la arquitectura RIA.

2.4.2.1. Componentes de la capa de acceso a datos.

2.4.2.1.1. Componentes de la capa de datos l3gica.

Estos componentes abstraen la l3gica necesaria para acceder a los almacenes de datos. Haciendo esto, se centraliza la funcionalidad de acceso a datos lo cual hace que la aplicaci3n sea m3s f3cil de configurar y mantener.

2.4.2.1.2. Ayudadores de datos y Utilidades.

Estas asisten en la manipulaci3n de datos, en su transformaci3n y en acceso en la capa. Estos consisten en librer3as especializadas y/o rutinas especialmente dise1adas para maximizar el rendimiento en el acceso a datos y reducir los requerimientos de desarrollo de los componentes l3gicos y de las partes del agente de servicio de la capa.

2.4.2.1.3. Agentes de Servicio

Cuando un componente de negocio debe usar funcionalidades expuestas por un servicio externo, se necesita crear c3digo que administre la sem3ntica de la comunicaci3n con ese servicio. Los agentes de servicios a3slan la aplicaci3n de la idiosincrasia de llamar diversos servicios y pueden proveer servicios adicionales como el mapeo b3sico entre el formato de la informaci3n expuesta por el servicio y el formato que la aplicaci3n requiere.

2.4.3. CAPA DE NEGOCIO.

Las implementaciones RIA proveen la capacidad de mover el proceso del negocio al cliente, considerando que mover la lógica mejora la experiencia de usuario o el rendimiento de la aplicación como un todo.

- Hay que considerar empezar con la lógica del negocio en el servidor a través de servicios. Solo mover la lógica del negocio al cliente para mejorar la totalidad del rendimiento del sistema o la capacidad de respuesta de la interfaz de usuario.
- Si se tiene una lógica duplicada hay que tratar de usar el mismo código tanto el cliente como en el servidor.
- Se puede mantener el código de proceso en lenguajes de programación más estructurados, poderosos o familiares como C#, actualmente Silverlight lo permite.

2.4.3.1. Componentes de la capa de negocio.

Los componentes de la capa de negocio implementan reglas del negocio en diversos patrones, aceptan y retornan simples o complejas estructuras de datos. Los componentes de negocio exponen la funcionalidad de una forma que es agnóstica a los almacenamientos de datos y servicios requeridos para ejecutar el trabajo. Si se falla en el diseño de los componentes de negocio el resultado es un código imposible de dar mantenimiento.

- Hay que evitar mezclar la lógica del acceso a datos con la de negocio en los componentes del negocio.
- No hay que sobrecargar la capa de negocio con funcionalidades no vinculadas con el negocio o mixtas.
- Si hay reglas de negocio volátiles, hay que almacenarlas en un motor de reglas.

- Si los procesos de negocio envuelven múltiples pasos o transacciones de larga ejecución, hay que considerar usar componentes de flujo de trabajo (Workflow).

2.4.3.2. Entidades del negocio.

Las entidades del negocio almacenan valores de la información y las exponen a través de propiedades, estas proveen acceso programable a la información del negocio y a funcionalidades relacionadas. Por tanto, diseñar o escoger las entidades de negocio apropiadas es vitalmente importante para maximizar el rendimiento y la eficiencia de la capa de negocio.

- Hay que escoger formatos de datos apropiados para las entidades de negocio. Como una regla general hay que usar objetos tradicionales. Sin embargo, para aplicaciones de manejo de datos pequeñas se puede considerar usar XML.
- Hay que analizar los requerimientos y la complejidad asociada con el diseño en el Modelo de dominio antes de escoger usar algo. Un modelo de dominio (Modelo Conceptual de un sistema que describe las entidades que participan en dicho sistema y sus relaciones)¹⁸ es muy bueno para manejar reglas de negocio complejas y trabajar mejor con una aplicación estable.
- Hay que usar serialización, las entidades deben soportar esta clase de requerimientos.

¹⁸ http://en.wikipedia.org/wiki/Domain_model

2.4.4. CAPA DE SERVICIOS DE LA APLICACIÓN

Cuando se provea funcionalidades de la aplicación a través de servicios, es importante separar la funcionalidad de los servicios dentro de una capa de servicios separada. En esta se define e implementa la interfaz del servicio, y se provee de componentes traductores, que como su nombre lo indica traducen formatos de información entre la capa de negocio con información externa. Uno de los conceptos más importantes a tener en cuenta es que un servicio nunca debe exponer información interna que se usa en la capa de negocio, para esto se usan las interfaces que son como fachadas.

2.4.4.1. Componentes de la capa de Servicios

2.4.4.1.1. Interfaces de Servicio

Los servicios exponen una interfaz de servicio donde todos los mensajes de entrada son enviados. La definición de este conjunto de mensajes que deben ser enviados con un servicio, es un contrato, porque el servicio ejecuta una tarea específica de negocio. Una interfaz de servicio es una fachada que expone la lógica de negocio implementada en el servicio para los potenciales consumidores.

2.4.4.1.2. Tipos de mensajes

Cuando se envía información a través de la capa de servicio, la estructura de los datos es envuelta por la estructura del mensaje que soporta diferentes tipos de operaciones, por ejemplo, se puede tener un mensaje de comando, un mensaje de documento u otro tipo de mensaje. Estos tipos de mensajes son los mensajes de contrato para la comunicación entre los servicios de consumidores y proveedores.

2.4.5. CAPA DE PRESENTACIÓN

Las aplicaciones RIA se ejecutan mejor cuando son diseñadas como una interfaz central. Una Interfaz de usuario de múltiples páginas requiere consideraciones, en cómo que se conectará entre páginas. El posicionar elementos en una página puede afectar tanto en el diseño como en el rendimiento de la aplicación RIA.

Hay que considerar los siguientes puntos para diseñar:

- Para evitar problemas que causen difuminación hay que acomodar los componentes de interfaz de usuario para que ocupen la totalidad de los pixeles. Hay que poner atención en centrar y posicionar con rutinas matemáticas. Se puede crear una rutina que chequee los pixeles fraccionados y los convierte al pixel completo más cercano.
- Hay que tener cuidado con los eventos de los botones de adelante y atrás para evitar navegación unidireccional afuera de la página.
- Para interfaces de usuario de múltiples páginas hay que usar métodos de conexión profundos para permitir una única identificación y navegación a páginas individuales de la aplicación.

2.4.5.1. Componentes de la Interfaz de usuario

Los componentes de la interfaz de usuario proveen una vía para que los usuarios interactúen con la aplicación. Estos despliegan y dan formato información para los usuarios. Estos también adquieren y validan información de entrada del usuario.

2.4.5.2. Procesos de componentes de Interfaz de Usuario

Estos procesos sincronizan y administran las interacciones del usuario. Separar los procesos puede ser útil si se tiene una interfaz de usuario complicada. Implementar patrones de interacción común como un proceso separado de componentes permite rehusarlos en múltiples interfaces.

2.4.6. EXPERIENCIA DE USUARIO

Diseñar para una experiencia de usuario efectiva puede ser crítico. Si la navegación es difícil o si los usuarios son dirigidos a páginas inesperadas la experiencia de usuario sería negativa.

- Se debe usar patrones compuestos para el estilo y el sentido, como MVC (Modelo vista controlador, patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos)¹⁹, controles supervisores, vistas pasivas para el procesamiento de interfaces de usuario.
- Diseñar la interfaz de tal forma que cada página o sección sea destinada para determinada acción.
- Es mejor tener varias páginas pequeñas que una sola y larga con muchas funcionalidades.
- Una buena experiencia de usuario puede hacer la diferencia entre una aplicación usable y otra inútil.
- Hay que tener en cuenta los estudios de usabilidad, encuestas y entrevistas para entender que es lo que el usuario requiere y espera de la aplicación y diseñar con este objetivo en mente.

¹⁹ http://es.wikipedia.org/wiki/Modelo_Vista_Controlador

2.5. TECNOLOGÍA MICROSOFT RIA: MICROSOFT SILVERLIGHT.



FIGURA 2.8 Silverlight.

2.5.1.INTRODUCCIÓN A WPF/E O SILVERLIGHT

Silverlight es una nueva tecnología de presentación web creada para su ejecución en distintas plataformas (Multiplataforma). Hace posible un uso más completo, atractivo visualmente e interactivo y se puede ejecutar en todos los entornos: como exploradores, en múltiples dispositivos y sistemas operativos de escritorio (como en Macintosh de Apple). Al igual que WPF (Windows Presentation Foundation), la tecnología de presentación en Microsoft .NET Framework 3.0 (la infraestructura de programación de Windows), XAML (eXtensible Application Markup Language, lenguaje de marcado de aplicaciones extensible), constituye la base de la capacidad de presentación de Silverlight.

Anteriormente conocido como WPF/E: 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.

Silverlight fue creada para presentar la siguiente generación de de experiencias en medios basados en .NET y aplicaciones altamente interactivas para la Web. 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 Mac OS 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.

Esencialmente, Silverlight es un complemento de navegador que 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 y, por otra parte, un 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.

2.5.2.EVOLUCIÓN DEL DESARROLLO WEB: AVANZANDO HACIA LA WEB.NEXT

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.

Como XAML es XML, está basado en texto, lo cual proporciona una descripción de sus ricos contenidos fácil de inspeccionar y totalmente compatible con los firewall.

Aunque se pueden usar otras tecnologías como subprogramas en Java, ActiveX y Flash para implementar contenidos más ricos que DHTML, CSS y JavaScript, todas ellas envían contenidos binarios al explorador. Esto resulta difícil de auditar, sin mencionar la dificultad de actualización, ya que para cualquier cambio es necesario que la aplicación se vuelva a instalar, lo que no es tan fácil para el usuario y podría provocar estancamientos en las páginas. Cuando se usa Silverlight y es necesario realizar un cambio en el contenido enriquecido, se genera un nuevo archivo XAML en el servidor. La próxima vez que el usuario vaya a la página, se descargará este XAML y se actualizará la experiencia sin ninguna reinstalación.

En el núcleo de Silverlight está el módulo de mejora del explorador que genera XAML y ofrece los gráficos resultantes en la superficie del explorador. Se trata de una descarga pequeña (menos de 4 MB), que se puede instalar cuando el usuario se encuentra con un sitio con contenido de Silverlight. Este módulo expone el marco de trabajo subyacente de la página XAML a los desarrolladores de JavaScript, de forma que se hace posible la interacción con el contenido en la página y, por tanto, el desarrollador puede, por ejemplo,

escribir controladores de eventos o manipular los contenidos de la página XAML mediante el código de JavaScript.

2.5.3. INFORMACIÓN GENERAL SOBRE SILVERLIGHT

Como ya se explicó anteriormente Microsoft Silverlight es una implementación multiplataforma de .NET Framework que se puede ejecutar en distintos exploradores para crear y proporcionar la nueva generación de experiencias multimedia y aplicaciones interactivas enriquecidas para la Web. Silverlight unifica las funciones del servidor, la Web y el escritorio, del código administrado y de los lenguajes dinámicos, de la programación declarativa y la tradicional, así como la eficacia de WPF (Windows Presentation Foundation)²⁰.

Silverlight permite crear aplicaciones de vanguardia con las siguientes características:

- Es una tecnología multiplataforma que se ejecuta en varios exploradores. Se ejecuta en todos los exploradores web conocidos, como Microsoft Internet Explorer, Mozilla Firefox y Apple Safari, así como en Microsoft Windows y Apple Mac OS X.
- Proporciona una experiencia coherente independientemente de dónde se ejecute.
- Es una descarga muy pequeña que se instala en pocos segundos.
- Transmite los contenidos de audio y vídeo por secuencias. Ajusta el contenido de vídeo a todo tipo de calidades, desde dispositivos móviles hasta exploradores de escritorio y modos de vídeo HDTV de 720p.

²⁰ http://es.wikipedia.org/wiki/Windows_Presentation_Foundation

- Incluye gráficos atractivos que los usuarios pueden manipular (arrastrar, girar y acercar o alejar) directamente en el explorador.
- Lee datos y actualiza la pantalla, pero no interrumpe al usuario al actualizar la página completa.
- Aplicación basada en Silverlight con gráficos enriquecidos e interacción con el usuario.

Los desarrolladores web y diseñadores de gráficos pueden crear aplicaciones basadas en Silverlight de diversas maneras. Se puede utilizar el marcado de Silverlight para crear elementos multimedia y gráficos y manipularlos con lenguajes dinámicos y código administrado. Silverlight también permite utilizar herramientas de calidad profesional, como Visual Studio para la codificación y Microsoft Expression Blend para la disposición y el diseño gráfico.

2.5.3.1. Características

Silverlight combina varias tecnologías en una sola plataforma de desarrollo que permite seleccionar las herramientas y el lenguaje de programación apropiados según las necesidades del usuario.

Silverlight ofrece las siguientes características:

- WPF y XAML.

Silverlight incluye un subconjunto de la tecnología WPF (Windows Presentation Foundation), que extiende en gran medida los elementos en el explorador para crear la interfaz de usuario. WPF permite crear gráficos, animaciones y elementos multimedia fascinantes, así como otras características de cliente enriquecidas, extendiendo la interfaz de usuario basada en explorador más allá de lo que está disponible

únicamente con HTML. El Lenguaje de marcado de aplicaciones extensible (XAML) proporciona una sintaxis de marcado declarativa para crear elementos.

- Extensiones a JavaScript.

Silverlight proporciona extensiones al lenguaje de scripting de explorador universal que permiten controlar la interfaz de usuario del explorador, incluida la capacidad para trabajar con elementos WPF.

- Compatibilidad con varios exploradores y plataformas.

Silverlight se ejecuta de la misma manera en todos los exploradores conocidos (en cualquier plataforma). Es posible diseñar y desarrollar aplicaciones sin tener que preocuparse del explorador o de la plataforma de los usuarios.

- Integración con aplicaciones existentes.

Silverlight se integra perfectamente con el código JavaScript y ASP.NET AJAX existente de modo que complementa la funcionalidad ya creada. Se pueden crear recursos basados en servidor en ASP.NET y utilizar las funciones AJAX de ASP.NET para interactuar con los recursos basados en servidor sin tener que interrumpir al usuario.

- Acceso al modelo de programación de .NET Framework y a las herramientas asociadas.

Se pueden crear aplicaciones basadas en Silverlight mediante lenguajes dinámicos, como IronPython, y lenguajes como C# y Visual

Basic. Se pueden utilizar herramientas de desarrollo como Visual Studio para crear aplicaciones basadas en Silverlight.

- Compatibilidad de red.

Silverlight incluye compatibilidad con HTTP sobre TCP. Se puede conectar a los servicios WCF (Windows Communication Foundation), SOAP (Simple Object Access Protocol es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML) o ASP.NET AJAX y recibir datos XML, JSON (subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML)²¹ o RSS (familia de formatos de fuentes web codificados en XML).

2.5.3.1.1. LINQ.

Silverlight incluye LINQ (Language Integrated Query), que permite programar el acceso a datos utilizando una sintaxis nativa intuitiva y objetos con establecimiento inflexible de tipos en los lenguajes de .NET Framework.

LINQ define operadores de consulta estándar que permiten a lenguajes habilitados con LINQ filtrar, enumerar y crear proyecciones de varios tipos de colecciones usando la misma sintaxis. Tales colecciones pueden incluir arreglos, clases enumerables, XML, conjuntos de datos desde bases de datos relacionales y orígenes de datos de terceros.

LINQ to SQL es una implementación de O/RM (object relational mapping, mapeador de objetos relacionales)²² que viene con la versión

²¹ <http://es.wikipedia.org/wiki/JSON>

²² <http://es.wikipedia.org/wiki/ORM>

“Orcas” del .NET Framework, y permite modelar bases de datos relacionales con clases de .NET.

Se puede consultar bases de datos con LINQ, así como actualizar/añadir/borrar datos de estas.

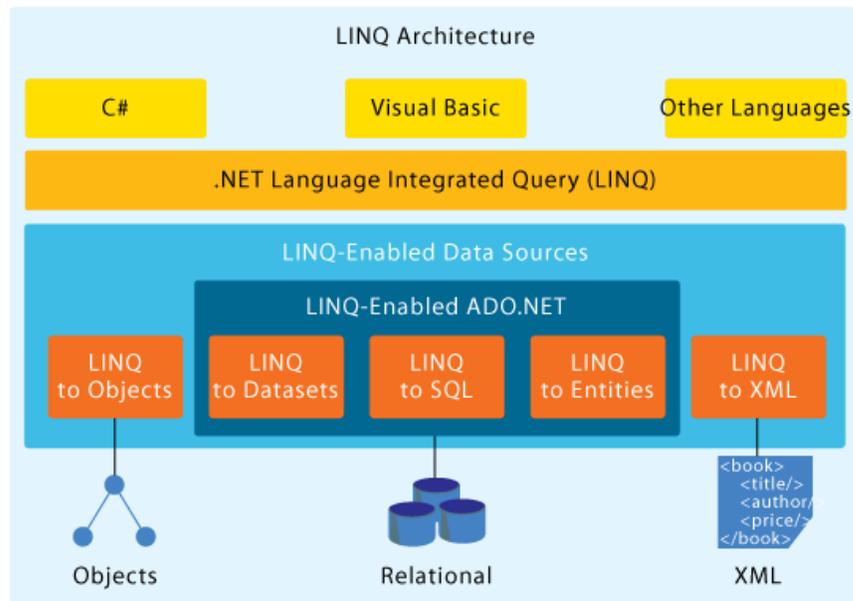


FIGURA 2.9 Arquitectura de LINQ

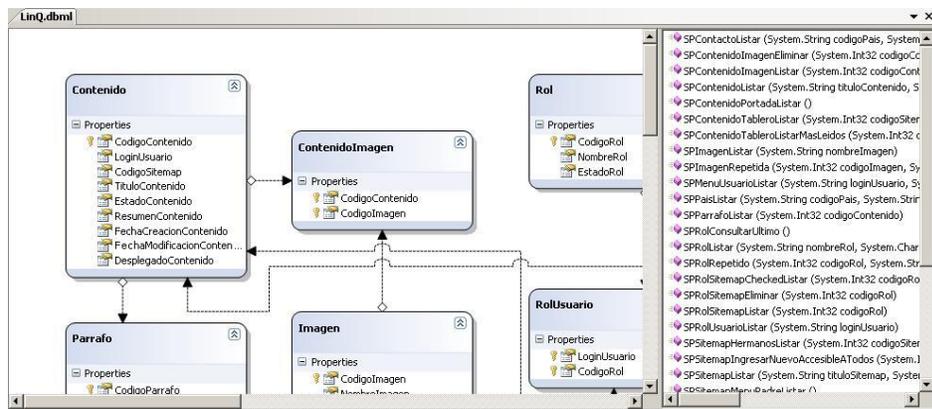


FIGURA 2.10 LINQ del proyecto.

Los desarrolladores pueden usar LINQ con cualquier fuente de datos. Pueden expresar consultas eficientemente en los lenguajes de

programación que elijan, opcionalmente transformar/incrustar los resultados de las consultas en el formato que quieran, y entonces manipular fácilmente los resultados.

Los lenguajes habilitados para LINQ pueden aportar seguridad de tipos y chequeo en tiempo de compilación en las expresiones de consulta, y desarrollar herramientas que aporten intellisense, debugging.

LINQ soporta un modelo de extensibilidad muy rico que facilita la creación de operadores eficientes para fuentes de datos. La versión “Orcas” del .NET Framework viene con librerías que habilitan LINQ sobre objetos, XML y bases de datos.

Ejemplos:

1. El siguiente código usa una consulta LINQ para obtener una secuencia IEnumerable de objetos Sitemaps.

```
var q = from c in db.Sitemaps
        where c.CodigoSitemap == i
        select c;
```

2. El siguiente código usa una consulta LINQ para obtener una secuencia IEnumerable de objetos de la entidad Contenido, mediante un método LINQ SPContenidoListar que implementa un procedimiento almacenado de SQL.

```
var q = from c in db.SPContenidoListar(item.Codigo, item.Estado)
        select c;
```

3. El código siguiente muestra cómo obtener un elemento de la entidad Contenido, de la base de datos, actualizar su estado y guardar los cambios en la base de datos:

```
Contenido objContenido = db.Contenidos.Single(c =>
c.CodigoContenido == pvoContenido.CodigoContenido);

objContenido.EstadoContenido = pvoContenido.EstadoContenido;

db.SubmitChanges();
```

4. El código siguiente muestra la inclusión de un registro de la entidad Contenido, en la base de datos.

```
Contenido objContenido = new Contenido();

objContenido.CodigoSitemap = item.CodigoSitemap;
objContenido.EstadoContenido = item.EstadoContenido;
objContenido.TituloContenido = item.TituloContenido;
objContenido.FechaCreacionContenido = item.FechaCreacionContenido;

db.Contenidos.InsertOnSubmit(objContenido);

db.SubmitChanges();
```

5. El código siguiente muestra la eliminación de un registro de Contenido específico en la base de datos.

```
Contenido objContenido = db.Contenidos.Single(c => c.CodigoContenido ==
item.CodigoContenido);

db.Contenidos.DeleteOnSubmit(objContenido);

db.SubmitChanges();
```

2.5.3.2. Crear aplicaciones basadas en Silverlight

Se pueden crear aplicaciones basadas en Silverlight mediante cualquier lenguaje compatible con .NET Framework (incluidos Visual Basic, C# y JavaScript). Visual Studio 2008 y Expression Blend admiten el desarrollo de aplicaciones basadas en Silverlight. Con Silverlight, se pueden crear páginas web con elementos HTML y WPF. Al igual que HTML, XAML permite crear la interfaz de usuario de las aplicaciones basadas en web mediante una sintaxis declarativa, pero XAML proporciona elementos mucho más eficaces.

2.5.3.3. Aplicaciones de Servidor con Silverlight

No es correcto pensar en Silverlight como una tecnología cliente hecha para incrustar contenido enriquecido en el navegador. Este tipo de contenido enriquecido ya es posible mediante el uso de complementos cerrados generados como subprogramas de Java, controles ActiveX o aplicaciones Flash. Silverlight, por su parte, es una tecnología abierta en el sentido que la interfaz de usuario se define en XAML basado en texto y la capacidad de programación se consigue con JavaScript, VB o C#. Esto permite a los desarrolladores generar fácilmente aplicaciones que interactúen con servidores back-end (parte que procesa la entrada desde el front-end (donde interactúa el usuario))²³.

Por ejemplo, si se deseara producir una aplicación meteorológica en el cliente, podría generar un subprograma de Java, un control ActiveX o una aplicación Flash que use un servicio Web y después se requiera implementar este subprograma en el cliente. Sin embargo, esto aumentaría las necesidades de comunicación entre el cliente y el servidor. ¿Qué sucede si este origen de datos fuera un servicio de suscripción de

²³ http://es.wikipedia.org/wiki/Front-end_y_back-end

pago? La persona que implementa la aplicación tiene que encargarse de autenticar las licencias de todos los clientes que obtienen acceso al servicio de datos y eso quita tiempo para crear lógica empresarial específica del dominio.

Sin embargo, si la aplicación se puede ensamblar en el servidor y todos los datos se pueden volver a pasar al cliente, entonces se reducirá la carga de trabajo. Al usar una herramienta como Expression Blend se puede ensamblar una plantilla para la interfaz de usuario de la aplicación y expresarla como XAML. A continuación, el desarrollador, en tiempo de ejecución, toma los datos pertinentes y los inserta en la plantilla, devolviendo el XAML terminado al cliente que lo representa. No se requiere ninguna lógica de conectividad al cliente (aparte de la conectividad inicial con el servidor) y la implementación y administración del cliente se mantienen relativamente sencillas.

2.5.4. ARQUITECTURA SILVERLIGHT

En la presente tesis se utiliza la versión de Silverlight 2.0 y esta versión admite dos modelos de programación: la API de JavaScript para Silverlight y la API administrada para Silverlight. La API administrada se basa en un subconjunto de .NET Framework y es la que se usará en este proyecto.

Silverlight no es solo un lienzo atractivo donde se puede mostrar a los usuarios finales contenidos web y multimedia enriquecidos e interactivos. También es una plataforma eficaz y, a la vez, ligera, donde se puede desarrollar aplicaciones portátiles, multiplataforma en red que integran datos y servicios de numerosos orígenes. Además, Silverlight permite generar

interfaces de usuario que mejorarán de manera significativa la experiencia del usuario final típico, en comparación con las aplicaciones web tradicionales.

Aunque Silverlight parece simple y de tamaño compacto cuando se considera como un entorno en tiempo de ejecución del lado cliente, la plataforma de desarrollo de Silverlight integra diversas características y tecnologías complejas, que pone a disposición de los programadores. Los programadores necesitan conocimientos prácticos de la arquitectura de la plataforma para poder crear aplicaciones efectivas basadas en Silverlight.

2.5.4.1. Plataforma Silverlight

La plataforma Silverlight en su conjunto se compone de dos partes principales, del marco de trabajo de presentación básico y del .Net Framework, además de un componente de instalador y actualización.

Componente	Descripción
Marco de trabajo de presentación básico	Componentes y servicios orientados a la interfaz de usuario y la interacción con el usuario, incluidos los datos proporcionados por el usuario, controles de interfaz de usuario ligeros para su uso en las aplicaciones web, reproducción de elementos multimedia, administración de derechos digitales, enlaces de datos; y características de presentación, incluidos gráficos vectoriales, texto, animaciones e imágenes. También incluye el lenguaje de marcado de aplicaciones extensible (XAML) para especificar el diseño.
.NET Framework para Silverlight	Subconjunto de .NET Framework que contiene componentes y bibliotecas, que incluyen integración de datos, controles de Windows extensibles, funciones de red, bibliotecas de clases

	<p>base, recolección de elementos no utilizados y CLR (Lenguaje común en tiempo de ejecución).</p> <p>Algunas partes de .NET Framework para Silverlight se implementan con la aplicación. Estas bibliotecas de Silverlight son ensamblados que no se incluyen en el motor en tiempo de ejecución de Silverlight, sino que se distribuyen en el SDK (Kit de desarrollo de Software)²⁴ de Silverlight.</p> <p>Cuando se utilizan bibliotecas de Silverlight en la aplicación, se empaquetan con esta última y se descargan al explorador.</p> <p>Incluyen nuevos controles de interfaz de usuario, XLINQ (LINQ a XML), distribución en formatos RSS o Atom, serialización XML y DLR (Lenguaje dinámico en tiempo de ejecución).</p>
<p>Componente de instalador y actualización</p>	<p>Control de instalación y actualización que simplifica el proceso de instalar la aplicación para los usuarios nuevos y proporciona actualizaciones automáticas de bajo impacto.</p>

²⁴ <http://es.wikipedia.org/wiki/SDK>

2.5.4.2. Arquitectura

Componentes y servicios relacionados.

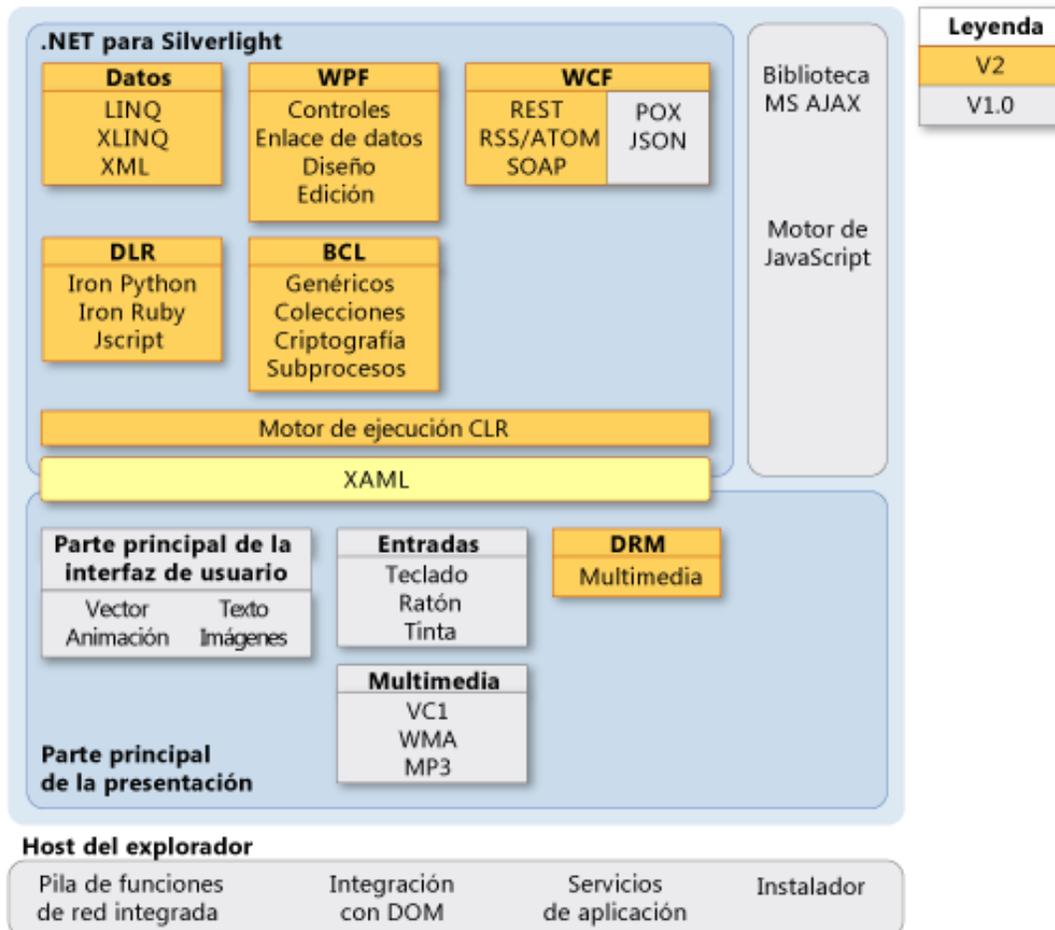


FIGURA 2.11 Arquitectura Silverlight.²⁵

Las herramientas, tecnologías y servicios que se incluyen en Silverlight encierran un valor concreto:

Facilitar a los programadores la tarea de crear aplicaciones enriquecidas e interactivas en red.

Aunque es posible generar aplicaciones RIA mediante las herramientas y tecnologías web actuales, el trabajo de los programadores se ve

²⁵ <http://msdn.microsoft.com/es-es/library/bb404713%28VS.95%29.aspx>

ralentizado por las numerosas dificultades técnicas, tales como plataformas incompatibles, protocolos y formatos de archivo dispares y diversidad de exploradores web que presentan las páginas y administran los scripts de manera diferente.

Una aplicación web enriquecida que se ejecuta perfectamente en una combinación de sistema y explorador puede funcionar de manera muy distinta, e incluso no funcionar, en otro sistema u otro explorador. Con la gran variedad actual de herramientas, protocolos y tecnologías, es un esfuerzo titánico y, con frecuencia, económicamente prohibitivo, generar una aplicación que pueda proporcionar al mismo tiempo las ventajas siguientes:

- Capacidad para crear la misma experiencia del usuario en la totalidad de exploradores y plataformas.
- Integración de datos y servicios procedentes de diversas ubicaciones de red en una misma aplicación mediante las clases y funcionalidades de .NET Framework.
- Una interfaz de usuario rica en elementos multimedia, atractiva y accesible.
- Silverlight facilita crear este tipo de aplicaciones a los desarrolladores, porque supera muchas de las incompatibilidades actuales entre las tecnologías y proporciona dentro de una misma plataforma, las herramientas necesarias para crear aplicaciones enriquecidas, multiplataforma e integradas.

2.5.4.2.1. Componentes de presentación básicos

Características de presentación básicas de la plataforma Silverlight:

Característica	Descripción
Entrada	Administra datos de entrada procedentes de distintos dispositivos hardware.
Representación de la interfaz de usuario	Representa gráficos vectoriales y de mapa de bits, animaciones y texto.
Multimedia	Reproducción y administración de varios tipos de archivos de audio y vídeo, como: .WMP y .MP3.
Controles	Admite controles extensibles que se pueden personalizar aplicando estilos y plantillas.
Diseño	Inserción dinámica de elementos de interfaz de usuario.
Enlace de datos	Vinculación de objetos de datos y elementos de la interfaz de usuario.
DRM (Administrador de derechos digitales)	Administración de derechos digitales de los recursos multimedia.
XAML	Proporciona un analizador para el marcado XAML.

Los programadores pueden interactuar con estas características de presentación utilizando XAML para especificar los detalles de presentación. XAML es el punto primario de interacción entre .NET Framework y la capa de presentación. Los programadores pueden manipular la capa de presentación mediante programación con código administrado (dinámicamente).

2.5.4.2.2. .Net Framework para Silverlight

Características de .NET Framework para Silverlight

Característica	Descripción
Datos	Características de LINQ y de LINQ a XML, que facilitan el proceso de integrar y trabajar con datos procedentes de orígenes dispares. Admite el uso de las clases de serialización y XML para administrar los datos.
Biblioteca de clases base (BCL)	Conjunto de bibliotecas de .NET Framework que proporcionan las funciones de programación esenciales, como la administración de cadenas, expresiones regulares, entrada y salida, reflexión, colecciones y globalización.
Windows Communication Foundation (WCF)	Simplificar el acceso a los servicios y datos remotos. Incluye un objeto de explorador, un objeto de solicitud y respuesta HTTP, compatibilidad con solicitudes HTTP entre dominios, compatibilidad con fuentes de distribución RSS o Atom así como compatibilidad con los servicios JSON, POX (término usado para describir XML básico) ²⁶ y SOAP.
CLR (Lenguaje común de tiempo de ejecución)	Administración de memoria, recolección de elementos no utilizados, comprobación de seguridad de tipos y control de excepciones.
Controles WPF	Conjunto enriquecido de controles, como: Button, Calendar, CheckBox, DataGrid, DatePicker, HyperlinkButton, ListBox, RadioButton y ScrollViewer.
DLR (Lenguaje dinámico en tiempo de ejecución)	Admite la compilación y ejecución dinámicas de lenguajes de scripting como JavaScript y IronPython para programar aplicaciones Silverlight. Incluye un modelo conectable que aporta compatibilidad de otros lenguajes con Silverlight.

²⁶ http://es.wikipedia.org/wiki/Plain_Old_XML_%28POX%29

.NET Framework para Silverlight es un subconjunto de .NET Framework. Proporciona los fundamentos del desarrollo robusto y orientado a objetos para tipos de aplicaciones (como las de Internet) que tradicionalmente no contaban con este tipo de recursos.

Los desarrolladores pueden interactuar con la capa de .NET Framework para Silverlight escribiendo código administrado en C# y Visual Basic, también pueden tener acceso a la capa de presentación creando aplicaciones en Visual Studio 2008 o Microsoft Expression Blend.

2.5.4.2.3. Características de programación adicionales de Silverlight

Silverlight proporciona varias características adicionales que ayudan a los programadores a crear aplicaciones RIA.

Característica	Descripción
Almacenamiento aislado	Acceso seguro del cliente de Silverlight al sistema de archivos del equipo local. Permite el almacenamiento local y el almacenamiento en caché de datos aislados para un usuario determinado.
Programación asíncrona	Un subproceso de trabajo en segundo plano lleva a cabo tareas de programación, mientras la aplicación queda libre para interactuar con el usuario.
Administración de archivos	Proporciona un cuadro de diálogo Abrir archivo, a fin de facilitar el proceso de update (subidas al servidor).
Interacción entre HTML y código administrado	Manipular directamente los elementos de la interfaz de usuario en el DOM HTML de una página web. Utilizar JavaScript para efectuar llamadas directas al código administrado y tener acceso a los objetos, propiedades, eventos y métodos que admiten el uso de scripts.

Serialización	Serialización de los tipos CLR a JSON y XML.
Empaquetar	Proporciona la clase Application (Administración de la aplicación Silverlight) ²⁷ y herramientas de compilación para crear paquetes .xap. El paquete .xap contiene la aplicación y el punto de entrada para que se ejecute el control del complemento Silverlight.
Bibliotecas XML	Las clases XmlWriter y XmlReader simplifican el trabajo con los datos XML de los servicios web. La característica XLINQ permite a los desarrolladores consultar los datos XML directamente en los lenguajes de programación.

2.5.4.2.4. Sistemas operativos y exploradores compatibles

Dado que las aplicaciones basadas en Silverlight son multiplataforma, se ejecutan en la mayoría de los exploradores web modernos. Sistemas operativos y exploradores compatibles.

Sistema operativo²⁸	Internet Explorer 7	Internet Explorer 6	Firefox 1.5, 2.x y 3.x	Safari 2.x y 3.x
Windows Vista	Sí	-	Sí	-
Windows XP SP2	Sí	Sí	Sí	-
Windows XP SP3	Sí	Sí	Sí	-
Windows 2000	-	Sí	-	-
Windows Server 2003 (excepto IA-64)	Sí	Sí	Sí	-
Mac OS 10.4.8+ (PowerPC)	-	-	-	-
Mac OS 10.4.8+ (basado en Intel)	-	-	Sí	Sí

²⁷ <http://msdn.microsoft.com/es-es/library/system.windows.application%28VS.95%29.aspx>

²⁸ [http://msdn.microsoft.com/es-es/library/bb404713\(VS.95\).aspx](http://msdn.microsoft.com/es-es/library/bb404713(VS.95).aspx)

2.5.4.2.5. Herramientas y tecnologías relacionadas

Las aplicaciones Microsoft siguientes incluyen características especiales para el desarrollo en Silverlight:

- Microsoft Expression Blend. Se puede utilizar para crear y modificar la capa de presentación de una aplicación; permite manipular el lienzo y los controles XAML, trabajar con gráficos y programar la capa de presentación mediante un lenguaje dinámico, como JavaScript, C# o VB Net.



FIGURA 2.12 Microsoft Expression Blend.

- Visual Studio 2008. Proporciona herramientas de productividad para desarrollar aplicaciones mediante código administrado. Todas las características existentes de Visual Studio están disponibles para Silverlight. Además, incluye las características específicas de Silverlight, tales como IntelliSense (autocompletar), depuración, y plantillas de proyecto Silverlight que crean y vinculan todos los archivos necesarios.



FIGURA 2.13 Microsoft Visual Studio 2008.

Como las aplicaciones basadas en Silverlight se ejecutan en un entorno de tiempo de ejecución en el equipo cliente, no es necesario que haya

ninguna aplicación concreta instalada en el servidor. Sin embargo, para mejorar el crear aplicaciones enriquecidas que integran en el servidor servicios y datos procedentes de diversos orígenes, existe la integración de los siguientes tipos de servicios y aplicaciones de servidor en las aplicaciones basadas en Silverlight:

- ASP.NET AJAX. Incluye un conjunto de controles, servicios y bibliotecas para crear aplicaciones ricas e interactivas basadas en web.
- Microsoft ASP.NET 3.5 Extensions Preview. Proporciona funciones adicionales para mejorar las aplicaciones ASP.NET AJAX. Incluye dos controles que son útiles para generar aplicaciones basadas en Silverlight como parte de una aplicación ASP.NET:
 - Control de servidor MediaPlayer de ASP.NET
 - Control de servidor Silverlight de ASP.NET
- Servicios de Microsoft Windows Communication Foundation (WCF).
- Servidores de Internet, incluidos Microsoft Internet Information Services (IIS) y el servidor web Apache.
- Aplicaciones y servicios basados en Internet, como Microsoft ASP.NET, PHP, servicios de transmisión multimedia por secuencias de Windows, servicios de Windows Live y otros servicios web abiertos.

2.5.4.3. Competencia más cercana

Adobe Flex.- Término que agrupa una serie de tecnologías publicadas desde 2004 por Macromedia (ahora Adobe) para dar soporte al despliegue y desarrollo de aplicaciones de Internet basadas en su plataforma Flash.

Nexaweb.- Framework de desarrollo y distribución de aplicaciones en Internet donde se pueden mantener interfaces de usuario con Java y Ajax.

OpenLaszlo.- Plataforma de código abierto que no requiere instalación para aplicaciones web con capacidades de interfaz de usuario del software del cliente y cuyos programas en XML y JavaScript se compilan de forma transparente a Flash y DHTML.

2.6. UTILIDADES MICROSOFT

2.6.1. CONTROLES DE USUARIO

Los controles de usuario hacen del desarrollo mucho más simple, al ser reusables y permitir su personalización, estos son contenedores en los que se puede colocar controles. Se puede tratar el control de usuario como una unidad y definir propiedades y métodos para el mismo.

Lo interesante de estos controles es que al ser personalizables y al unirlos con la tecnología RIA permiten diseños novedosos con ahorro de tiempo y esfuerzos, haciendo que las funcionalidades o procesos de las páginas web utilicen menos esfuerzo del usuario.

Los controles de usuario son la unidad básica de XAML reusable y del código que viene con este. Un control de usuario tiene su parte XAML representada por el archivo de extensión .xaml y su parte de código dinámico de programación (Code-Behind) representada por el archivo de extensión

.xaml.cs o .xaml.vb dependiendo si está en Csharp o en Visual Basic.Net respectivamente.

De esta manera es como se presenta en código XAML:

```
<UserControl x:Class="nombre"  
xmlns="http://schemas.microsoft.com/client/2007"  
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"/>
```

Y el code-behind viene de esta forma

```
namespace nombre  
{  
public partial class Page : UserControl  
{  
}  
}
```

Se añade controles de usuario a una aplicación web para encapsular alguna funcionalidad que se desea reutilizar.

Ejemplo gráfico:

Un usuario puede poseer varios domicilios.

Domicilios que tiene un usuario

Domicilio 1:
Tipo:
Extensión:
Dirección:

Domicilio 2:
Tipo:
Extensión:
Dirección:

Domicilio 3:
Tipo:
Extensión:
Dirección:

FIGURA 2.14 Programación sin usar User Controls.

Código XAML:

```
<UserControl x:Class="Ejemplodeusercontrol.Page"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml" Width="400" Height="600">
  <Grid x:Name="LayoutRoot" Background="White">
    <TextBlock >Domicilios que tiene un usuario</TextBlock>

    <TextBlock Text="Domicilio 1:" />
    <TextBlock Text="Tipo:" />
      <ComboBox x:Name="comboboxTipo1"/>
    <TextBlock Text="Extensión:"></TextBlock>
      <ComboBox x:Name="comboboxExtension1"></ComboBox>
    <TextBlock Text="Dirección:"/>
      <TextBox x:Name="TextBoxDireccion1"/>

    <TextBlock Text="Domicilio 2:" />
    <TextBlock Text="Tipo:" />
      <ComboBox x:Name="comboboxTipo2"/>
    <TextBlock Text="Extensión:"></TextBlock>
      <ComboBox x:Name="comboboxExtension2"></ComboBox>
    <TextBlock Text="Dirección:"/>
      <TextBox x:Name="TextBoxDireccion2"/>

    <TextBlock Text="Domicilio 3:" />
    <TextBlock Text="Tipo:" />
      <ComboBox x:Name="comboboxTipo3"/>
    <TextBlock Text="Extensión:"></TextBlock>
      <ComboBox x:Name="comboboxExtension3"></ComboBox>
    <TextBlock Text="Dirección:"/>
      <TextBox x:Name="TextBoxDireccion3"/>

  </Grid>
</UserControl>
```

Si no se crea un control de usuario entonces habrá que programar las veces que sean necesarias los campos para cada domicilio. Se puede apreciar que hay 3 Combobox tipo, 3 Combobox extensión y 3 textos dirección. En cambio, si se crea un control de usuario que podría llamarse información del domicilio, basta con llamarlo las veces que se necesiten y se ahorrarán líneas de código.



FIGURA 2.15 User Control.

```
<UserControl x:Class="Ejemplodeusercontrol.UsarUserControl"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:UsarControl="clr-namespace:Ejemplodeusercontrol" Width="400" Height="300">
  <Grid x:Name="LayoutRoot" Background="White">
    <TextBlock>Domicilios que tiene un usuario</TextBlock>
    <UsarControl:InformacionDelDomicilio
  x:Name="Uno"></UsarControl:InformacionDelDomicilio>
    <UsarControl:InformacionDelDomicilio
  x:Name="Dos"></UsarControl:InformacionDelDomicilio>
    <UsarControl:InformacionDelDomicilio
  x:Name="xxxx"></UsarControl:InformacionDelDomicilio>
  </Grid>
</UserControl>
```

Se está usando el control de usuario llamado Información del domicilio, el cual ya consta de la programación requerida y basta solo con llamarlo las veces que se necesiten. Este es un ejemplo no dinámico, solo se ha usado código xaml, pues habría programación dinámica (code-behind) que incluso podría controlar esto con menos líneas de código.

2.6.2. WINDOWS COMMUNICATION FOUNDATION WCF SERVICES.

Es una estructura de programación usada para construir aplicaciones que se comunican entre sí. WCF es la parte del Framework de .NET dedicada a las comunicaciones.

El WCF es una de las cuatro nuevas APIs (WCF, WPF, Card Space, WF) que presentó con. Framework .NET 3.0, que se publicó en diciembre de 2006.

Framework .NET es la tecnología de Microsoft que se carga en los sistemas operativos Windows (cliente, servidor y plataformas móviles). Al ser parte de Framework, WCF puede ser desarrollado en cualquier lenguaje de programación de .NET.

WCF unifica los modelos de comunicación de programación en un solo modelo. El Framework .NET 2.0 proveía APIs separadas para la comunicación basada en SOAP mediante Web Services, optimización binaria para la comunicación entre aplicaciones corriendo en máquinas Windows mediante .Net Remoting, comunicaciones transaccionales mediante Distributed Transactions, y comunicación asincrónica mediante Message Queues. Ahora WCF unifica estas capacidades de estos mecanismos en un único modelo de programación orientada al servicio.

2.6.2.1. Arquitectura orientada a servicios.

WCF está diseñado de acuerdo a los principios de la arquitectura orientada a servicios. Para soportar aplicaciones distribuidas donde los servicios son consumidos por los clientes. Los clientes pueden consumir múltiples servicios y los servicios pueden ser consumidos por múltiples clientes.

Los servicios tienen una interfaz WSDL (Lenguaje de descripción de Web Services)²⁹ la misma que cualquier cliente puede usar para consumir el servicio, independientemente de la plataforma en que se encuentra en servicio. La interfaz asemeja una fachada a la cual los clientes tienen acceso.

2.6.2.2. Composición del WCF Service.

Service Class, donde se implementa el servicio a ser suministrado, un **Host Environment**, para albergar el servicio y **Endpoints**, que son puntos de conexión para el cliente. Todas las comunicaciones con el servicio pasan a través de los endpoints, estos especifican un contrato que define cuales métodos del Service Class van a ser accesibles, además define los criterios de valoración **binding**³⁰ que especifica cómo el cliente se comunicará con el servicio y la dirección donde se encuentra el endpoint.

Ejemplo:

```
<service behaviorConfiguration="SitemapBehavior" name="Sitemap">
  <endpoint address="" binding="basicHttpBinding" contract="ISitemap">
    <identity>
      <dns value="localhost"/>
    </identity>
  </endpoint>
</service>
```

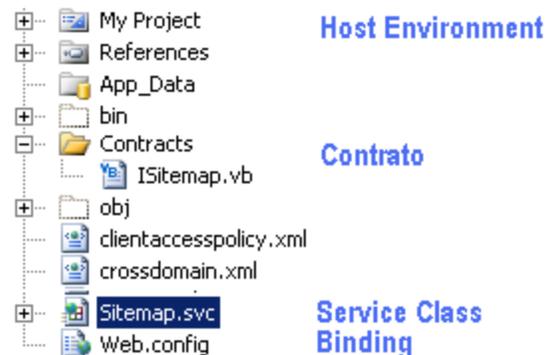


FIGURA 2.16 WCF Service.

²⁹ <http://es.wikipedia.org/wiki/WSDL>

³⁰ <http://msdn.microsoft.com/en-us/library/ms733107.aspx>

2.6.2.3. Comunicación con el servicio.

La comunicación con un servicio WCF puede ser sincrónica o asincrónica.

2.6.2.3.1. Sincrónica.

Un cliente puede comunicarse con un servicio WCF utilizando cualquiera de los mecanismos basados en RPC en el que el servicio puede ser invocado como un método de llamada. Usando comunicación sincrónica el llamado al servicio puede ser bloqueado, es decir, este tipo de comunicación puede detener la ejecución del cliente hasta que el servicio procese el pedido.

Ejemplo:

'Primero se llama al método listar

```
Private Sub UserControlSitemap_Loaded(ByVal sender As Object, ByVal e As
System.Windows.RoutedEventArgs) Handles Me.Loaded
    ListarSitemap()
End Sub
```

```
Public Sub ListarSitemap()
```

'Listar Sitemaps

'Comunicación Sincrónica

```
Dim objWCFSitemap As New WCFsitemap.WCFSitemapClient()
```

'llamado sincrónico para llenar el grid

```
myGrid.ItemsSource = objWCFSitemap.ListarSitemap(objParametrosSitemap)
```

'Aquí se detiene la ejecución del cliente aún si hay más líneas de código, hasta que se complete el llamado al método.

```
End Sub
```

2.6.2.3.2. Asincrónica.

WCF también soporta comunicación asincrónica la cual no bloquea las llamadas entre cliente y servidor. Los mecanismos para utilizar comunicación asincrónica pueden ser: Message Queues como transporte para la entrega y recepción de mensaje, o por Multi-threads que genera un proxy para WCF en el lado del cliente.

Ejemplo:

'Primero se llama al método listar

```
Private Sub UserControlSitemap_Loaded(ByVal sender As Object, ByVal e As System.Windows.RoutedEventArgs) Handles Me.Loaded
    ListarSitemap()
End Sub
```

```
Public Sub ListarSitemap()
```

```
    'Listar Sitemaps
```

```
    'Comunicación Asincrónica
```

```
    Dim binding As New System.ServiceModel.BasicHttpBinding()
```

```
    Dim url As String = App._InitP("WCFsitemap")
```

```
    Dim address As New System.ServiceModel.EndpointAddress(New Uri(url))
```

```
    objWCFsitemap = New SitemapClient(binding, address)
```

```
    AddHandler objWCFsitemap.ListarSitemapCompleted, AddressOf
```

```
    ListarSitemap_Completed
```

```
    'Llamado asincrónico al método.
```

```
    objWCFsitemap.ListarSitemapAsync(objParametrosSitemap) 'Si existiera más líneas de código la ejecución seguiría y el cliente no se quedaría bloqueado esperando que se complete el pedido al método.
```

```
End Sub
```

```
Private Sub ListarSitemap_Completed(ByVal sender As Object, ByVal e As
```

```
ListarSitemapCompletedEventArgs)
```

'Cuando se haya procesado el pedido en el Servicio WCF, se retorna la respuesta, sin que se haya bloqueado el cliente aquí en el Completed

```
'Uso de la respuesta del servicio = e.Result para llenar el grid
    myGrid.ItemsSource = e.Result
    RemoveHandler objWCFSitemap.ListarFamiliarCompleted, AddressOf
ListarSitemapCompletedEventArgs
    objCliente = Nothing
End Sub
```

El objeto DataGrid.ItemsSource es donde el cliente toma el resultado del WCF Service.

2.6.2.4. Estilos XAML

Silverlight soporta un mecanismo de estilos que permite encapsular valores de propiedades de control como un recurso reutilizable. Se puede almacenar estas declaraciones de estilos en archivos separados de las páginas y rehusarlos a través de múltiples controles y páginas en una o varias aplicaciones. Es similar a usar CSS (Hojas de estilo en cascada) con HTML, cuando se hacen personalizaciones de escenarios básicos.

Además, de definir ajustes de propiedades básicas como color, tipo de letra, tamaño, márgenes, etc. Los estilos en Silverlight pueden también ser usados para definir y rehusar plantillas de controles, los que permiten crear ricos escenarios, como ya se había explicado anteriormente es aquí donde se unen las habilidades de los diseñadores a la de los programadores.

Los estilos son extremadamente importantes porque permiten a los desarrolladores controlar el look y el diseño de los controles en la aplicación. Usando estilos se puede simplemente cambiar la declaración del estilo y todos los controles en la aplicación serán actualizados automáticamente. Es una alternativa mucho más rápida que actualizar a cada uno de los controles en la aplicación de forma individual.

En Silverlight, los estilos son declarados en los controles mediante esta `<UserControl.Resources>` etiqueta, pero si se desea hacerlos globales, es decir, que sirvan para todos los controles se la puede poner en el archivo App.Xaml dentro de las etiquetas `<Application.Resources>`.

A continuación se ha creado un estilo global en el cual se ha programado un estilo para los controles **TextBlocks**.

```
<Application xmlns=http://schemas.microsoft.com/winfx/2006/xaml/presentation
xmlns:x=http://schemas.microsoft.com/winfx/2006/xaml x:Class="MiAplicacion.App">
  <Application.Resources>
    <Style x:Name="EstiloTextBlock" TargetType="TextBlock">
      <Setter Property="FontFamily" Value="Calibri.ttf#Calibri"></Setter>
      <Setter Property="FontSize" Value="16"></Setter>
    </Style>
  </Application.Resources>
</Application>
```

A continuación se explica la estructura de este archivo APP.Xaml:

1. Declaración del estilo `<Style></Style>` dentro de la sección de las etiquetas `<Application.Resources>`.
2. La propiedad Name pone el nombre del estilo `x:Name="EstiloTextBlock"` para poder referenciarlo desde los controles.
3. La propiedad TargetType pone el nombre del elemento de framework para el cual se está creando el estilo, en este caso el elemento **TextBlock**
4. A cada propiedad que se desea poner en el estilo se la declara dentro de la etiqueta Setter. Se puede modificar los atributos de múltiples propiedades

mediante múltiples etiquetas Setter. Por ejemplo FontSize (tamaño de letra), FontFamily (Tipo de letra), etc.

El siguiente XAML es de un control de usuario que muestra tres elementos TextBlock aplicados el estilo que se creó anteriormente.

```
<TextBlock x:Name="TextBlock1" Text="Babel Software"  
Style="{StaticResource EstiloTextBlock}"></TextBlock>
```

```
<TextBlock x:Name="TextBlock2" Text="Ecuador" Style="{StaticResource  
EstiloTextBlock}"></TextBlock>
```

```
<TextBlock x:Name="TextBlock3" Text="Tesis de Grado"  
Style="{StaticResource EstiloTextBlock}"></TextBlock>
```

Con este estilo aplicado **EstiloTextBlock** los tres TextBlock tendrán el tipo de letra Calibri y tendrán un tamaño de 16.

III. CAPÍTULO III

DISEÑO DEL SITIO WEB DE BABEL SOFTWARE ECUADOR

3.1. ANÁLISIS Y DEFINICIÓN DE CARACTERÍSTICAS DEL SITIO WEB.

3.1.1. ANTECEDENTE.

Este proyecto consiste en diseñar el sitio Web de Babel Software Ecuador. Sin embargo; no basta con tener un sitio Web funcional y atractivo para los usuarios, sino que existe también un aspecto aún más importante que determinará el éxito del sitio Web y esto se llama: Marketing o Mercadeo Web, la cual tiene como misión, optimizar el sitio, a partir de estrategias que permitan información confiable y actualizada, poniendo especial hincapié en la calidad del trabajo, el cual se consigue con el máximo cuidado en los detalles.

Un sitio desfasado, dice poco de la empresa. En cambio uno actual, hará que los clientes vean que la empresa es dinámica y que está al día con las últimas tecnologías. De esta forma, los clientes tendrán una idea positiva de la empresa y podrá de esta forma conseguir, atraer y fidelizar a nuevos clientes con un buen diseño Web.

La idea es tener en cuenta el objetivo final del sitio Web pasando los máximos controles de calidad en cuanto a su usabilidad y contenido.

Esto con el fin, de conseguir que el cliente potencial al entrar a la página Web se sienta interesado por el contenido, encuentre rápido lo que busca y de esta manera confíe en la empresa, lo que puede traducirse en una venta para la misma.

Las consideraciones relevantes para diseñar el sitio Web son:

1. Explotar las líneas de negocio que se tienen.
2. Mención de clientes.
3. Casos de éxitos.
4. Información sobre la presencia que se posee actualmente a nivel nacional e internacional.
5. Informar la importancia de contar con el respaldo de la certificación Microsoft Gold Partner que tiene la empresa y sus beneficios.
6. Filosofía organizacional (Historia, Misión, Visión, Objetivos y Valores).

3.1.2. ARQUITECTURA DE LA INFORMACIÓN.

3.1.2.1. Estructura de Organización de la Información (ESOI)

El sitio Web de Babel Software del Ecuador utiliza una estructura de la organización mediante el modelo de base de datos relacionada, la misma que utiliza el siguiente diagrama:

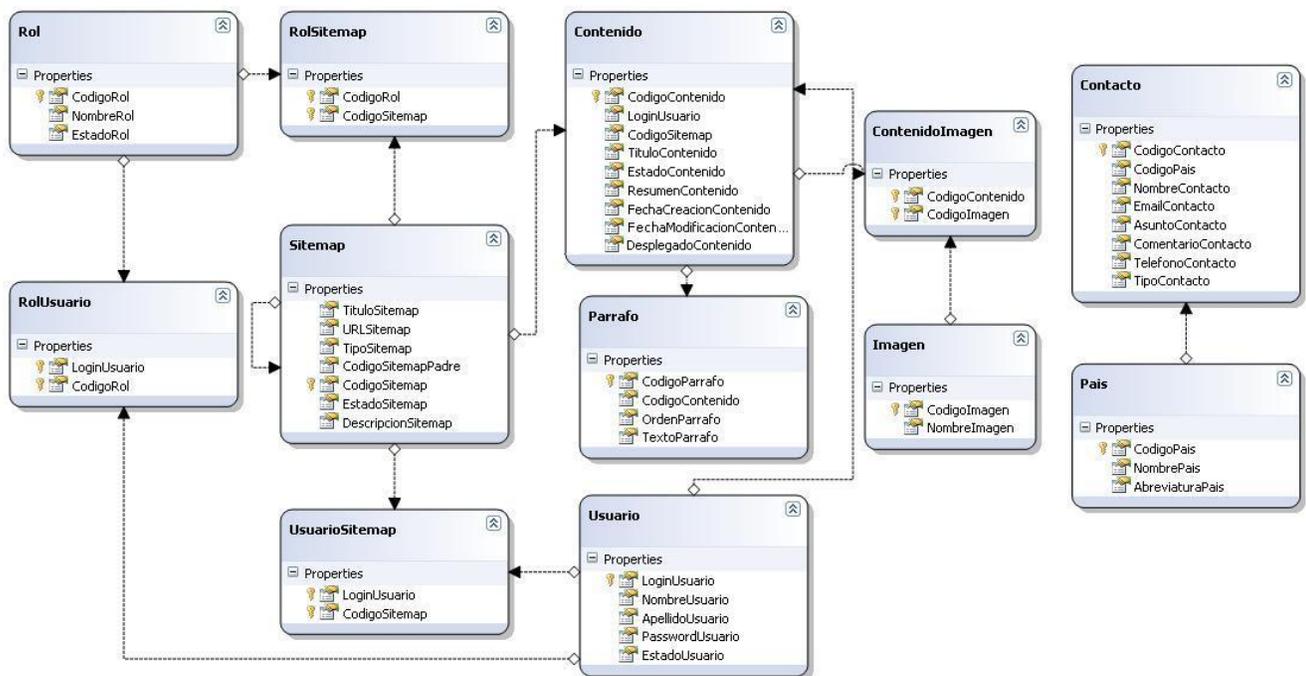


FIGURA 3.1 Estructura de Organización de la Información (ESOI).

Usuario: define a una persona que utiliza el sistema, a este usuario se le asigna un Rol y un Sitemap (Categoría). Estas asignaciones definen los permisos de edición, lectura, y/o accesos a páginas que tiene dicha persona.

Rol: define un perfil para usuario asociados con los permisos de edición, lectura, y/o acceso a páginas dentro del sistema. Un rol puede permanecer inactivo dentro del sistema.

Sitemap: (Categoría) se define a la interfaz entre el usuario del sistema y una página o páginas de contenidos.

Contenido: se define sintéticamente como una página web dentro del sistema.

Párrafo: define un texto con la descripción del contenido de una página web del sistema.

Imagen: define una ruta con una parte grafica del contenido de una página web del sistema.

País: agrupa los comentarios recibidos por los visitantes según su país.

Contacto: define los comentarios recibidos por los visitantes en el sitio web.

La interacción de las entidades se define mediante las relaciones existentes entre ellas y las entidades que las relacionan como: RolSitemap, RolUsuario, UsuarioSitemap y ContenidoImagen.

3.1.3.USUARIOS.

3.1.3.1.Usuario anónimo.

Son todas las posibles personas que vayan a entrar en el sitio Web, ya sea para obtener información de la empresa o para contactarla.

Los visitantes del sitio Web que no estén registrados podrán enviar un comentario o pregunta en la sección contáctenos pues podría ser un posible cliente.

3.1.3.2.Usuario registrado.

Serán los propios empleados de la empresa quienes poseerán su nombre de usuario y contraseña para registrarse en el sitio Web. Estos usuarios tendrán acceso a la sección administrativa, en donde podrán modificar el contenido del sitio de acuerdo al rol y permisos asignados a su cuenta.

3.1.4.ROLES.

Rol de usuario son los permisos que se asignan a un usuario que se registra en el sistema (Sitio Web), como ya se había mencionado anteriormente el Sitio Web de Babel Software del Ecuador poseerá usuarios registrados y anónimos, pero solo a los usuarios registrados se les asigna un rol, los roles se clasificarán en:

3.1.4.1.Administrador (Web Máster).

Los usuarios con rol de administrador podrán subir casos de éxito y también noticias, además podrán modificar o eliminar las noticias o casos de éxito que ya no tengan relevancia, administrar todas las ventajas que brinda el manejador de contenidos, como es crear nuevas categorías y contenidos, controlar la información de los usuarios, crear, modificar y eliminar usuarios.

3.1.4.2.Editor.

Los editores podrán crear contenidos, además de hacerlos accesibles para los roles ya creados como el de visitante.

3.1.4.3.Visitante.

Los usuarios con rol de visitante podrán ver contenidos que se creen por los usuarios con rol de administrador o editor.

3.1.5.CATEGORÍAS DE CONTENIDOS

El sitio Web de Babel Software Ecuador se ha organizado en base a sus necesidades de darse a conocer y de mostrar los servicios que ofrece.

Consta de un CMS con un ESOI de modelo de base de datos que permite disponer de una estructura con contenidos, usuarios, roles, y categorías dinámicos; flexible en su crecimiento y liviano en su presentación.

La información del sitio Web de Babel Software del Ecuador dispuesta en categorías es la siguiente:

3.1.5.1.Compañía.

- **Oficinas Corporativas.-**
Se brindará información sobre las oficinas corporativas de Babel Software y sus ubicaciones.
- **Perfil Corporativo.-**
Habrá información sobre, la reseña histórica, misión, visión, valores y el enfoque de negocios (Mención de los sectores en los que incursiona la empresa)

3.1.5.2.Servicios.

Constará con información de los siguientes temas:

- Desarrollo de Software a la medida
- Inteligencia de negocios
- Soporte de aplicaciones
- Outsourcing
- Software Factory

3.1.5.3.Noticias.

Será una sección para mantener actualizado a los funcionarios de la empresa y potenciales clientes sobre el dinamismo y el constante cambio de la misma.

3.1.5.4.Casos de éxito.

Con este contenido lo que se pretende es crear confianza y credibilidad en los clientes. Mostrando información sobre los proyectos que ha tenido la empresa.

3.1.5.5.Contáctenos.

Este apartado permitirá a los visitantes enviar una consulta o comentario acerca de los productos o servicios que ofrece que ofrece Babel Software del Ecuador. Además, el visitante deberá insertar su información para ser contactado, ya sea email o teléfono.

3.1.5.6.Administración.

En esta sección los administradores del sitio Web podrán dar mantenimiento a los usuarios, a las categorías y a los contenidos del sistema, además podrán ver los contactos.

3.1.6. DISEÑO DE LA APLICACIÓN.

3.1.6.1. Arquitectura.

La aplicación consta de dos soluciones definidas de la siguiente manera:

UIBabelSoftware. Es una solución tipo Aplicación de Silverlight que contiene a la Interfaz de Usuario mediante dos proyectos:

- BabelSoftware.Web, de tipo ASPX que sirve como contenedor.
- BabelSoftware, de tipo Silverlight en donde se generan las pantallas mediante código XAML.

WCFBabelSoftware. Es una solución de clases de .NET que contiene 3 proyectos.

- BEUBabelSoftware, proyecto de clases, que refleja las entidades definidas en la base de datos.
- BLLBabelSoftware, proyecto de clases, que contiene la conexión persistente con la base de datos mediante LINQ y la definición de los métodos para el mantenimiento y actualización de la misma, además se usan Stored Procedures.
- WCFServicio, proyecto tipo WCF, para la definición de servicios de comunicación desde la solución de UIBabelSoftware hacia la capa de conexión BLLBabelSoftware.

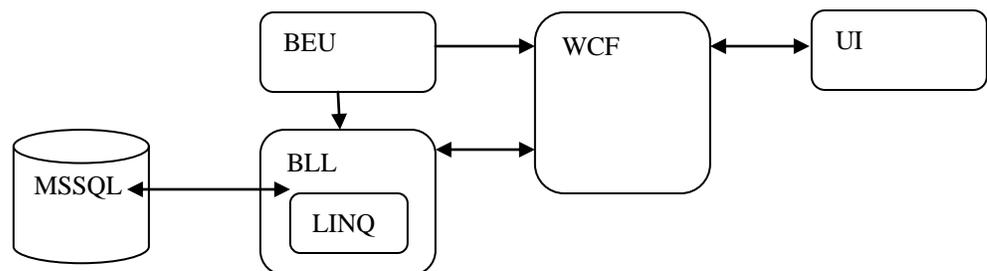


FIGURA 3.2 Arquitectura de la aplicación.

3.2. DESARROLLO DEL SITIO WEB DE BABEL SOFTWARE ECUADOR.

3.2.1.ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

3.2.1.1.Requisitos funcionales

3.2.1.2.Requisitos no funcionales

3.2.1.3.Casos de Uso

Ver Anexo ERS y EDS.

3.2.2.IMPLEMENTACIÓN

El sitio Web ha sido programado en C# mediante la tecnología Silverlight 2.0, los WCF igualmente se los ha programado en C#, ambos en Visual Studio 2008.

El sitio Web es un proyecto de aplicación Web en Silverlight, consta de su parte Web, que es la contenedora del archivo .xap, el cual es el comprimido de todo el proyecto Silverlight.

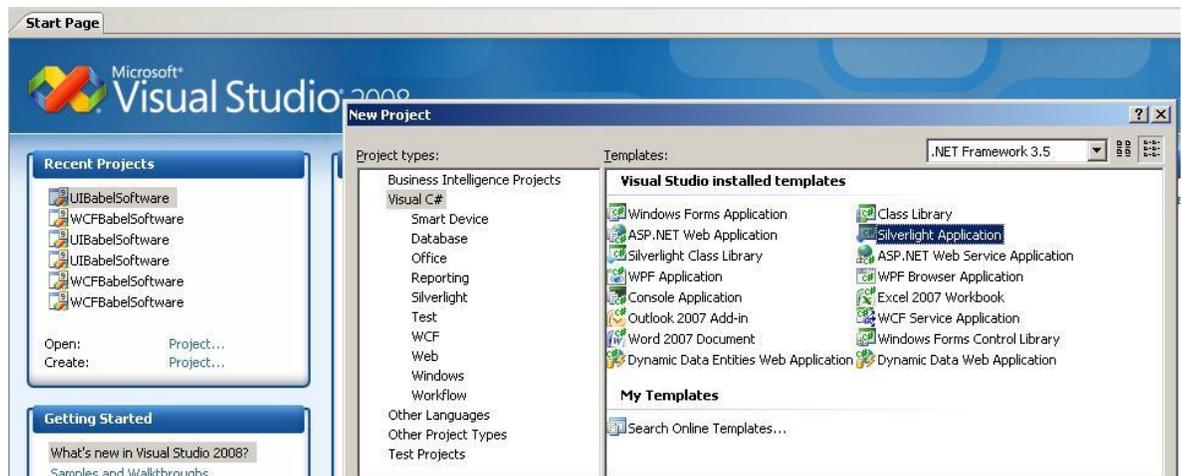


FIGURA 3.3 Nuevo proyecto Silverlight.

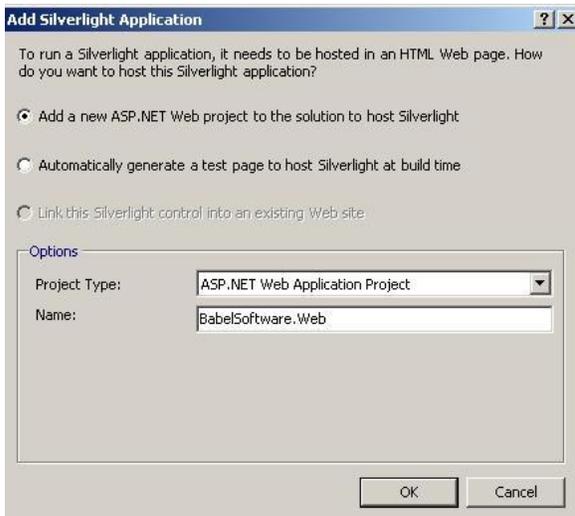


FIGURA 3.4 Añadir un proyecto Web.



FIGURA 3.5 Solución Silverlight.

El proyecto WCF como fue explicado en la parte de arquitectura del sitio Web consta de las capas BEU, BLL y WCF.



FIGURA 3.6 Solución WCF.

La parte de diseño se la ha realizado en Microsoft Expression Blend 2.0.

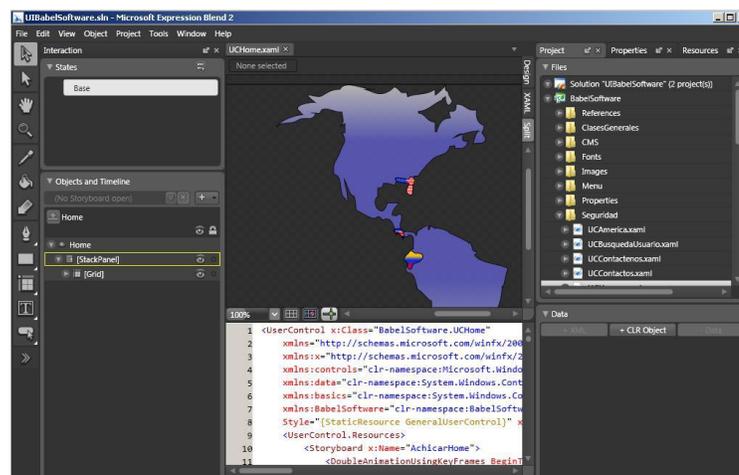


FIGURA 3.7 Microsoft Expression Blend 2.0.

La base de datos se encuentra en SQL 2008 y consta además con Stored Procedures (programa que es ejecutado directamente en la bases de datos)³¹.

3.2.3.PRUEBAS

Las pruebas sobre el sitio Web se realizaron en base a los casos de uso extendidos, para certificar la ejecución de cada requisito. Conjuntamente, se reviso cada pantalla en base a los requerimientos no funcionales para certificar que se maneje un estándar de diseño en todo el sitio.

De estas pruebas se obtuvieron los siguientes resultados.

	Acción	Comentario
	Presentar Sitio web	Si
	Presentar Menú	Si
	Crear categoría	Si
	Consultar categoría	Si
	Modificar categoría	Si
	Listar categoría	Si
	Mostrar contenido de compañía	Si
	Mostrar contenidos de servicios	Si
	Mostrar contenidos de casos de éxito	Si
	Mostrar contenidos de noticias	Si
Error	Crear contenido	Al guardar un contenido, con la primera imagen mostrada, no se guardaba la imagen.
Solución		Se determinó que no se enviaba el código de imagen, si el usuario no presionaba los botones atrás o adelante en las imágenes. Error resuelto.
	Consultar contenido	Si

³¹ http://es.wikipedia.org/wiki/Procedimiento_almacenado

	Modificar contenido	Si
	Listar contenido	Si
	Desactivar contenido	Si
	Enviar correo	Si (Está configurado con los parámetros del Servidor SMTP de Babel Software)
Error	Agregar párrafo	Al presionar el botón Agregar se quedaba el Mensaje en 0/0. Y no en 1/1
Solución		Se arregló el error al agregar se aumenta el número de párrafo a su correspondiente
	Consultar párrafo	Si
	Modificar párrafo	Si
	Eliminar párrafo	Si
	Subir imagen	Si
	Reemplazar imagen	Si
	Consultar imagen	Si
	Listar imagen	Si
	Crear usuario	Si
	Consultar usuario	Si
	Modificar usuario	Si
	Listar usuario	Si
	Crear rol	Si
	Consultar rol	Si
	Modificar rol	Si
	Listar roles	Si
	Asociar usuario a categoría.	Si
	Asociar usuario a rol.	Si
	Asociar rol a categorías.	Si
	Iniciar sesión.	Si
	Consultar correos recibidos	Si (Consultar Contactos)

	Listar correos recibidos	Si (Listar Contactos)
	Correspondencia con requerimientos no funcionales.	Si

3.2.4. INTEGRACIÓN

Para poder ejecutar el sistema desde el internet, se lo debe instalar en un servidor Web en este caso por usar la tecnología Microsoft Silverlight, es necesario el Internet Information Server 6.0 o superior.

3.2.4.1. Creación de los instaladores

Primer paso: Agregar un Nuevo Proyecto en la Solución, este debe ser del tipo Setup Wizard.

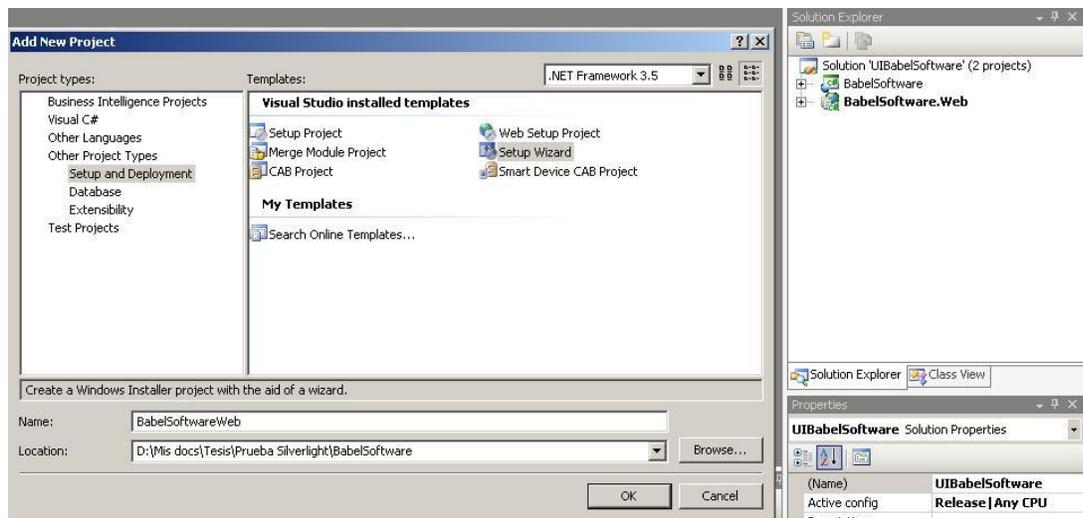


FIGURA 3.8 Creación de instaladores Paso 1.

Segundo paso: escoger la creación del instalador para aplicaciones Web.

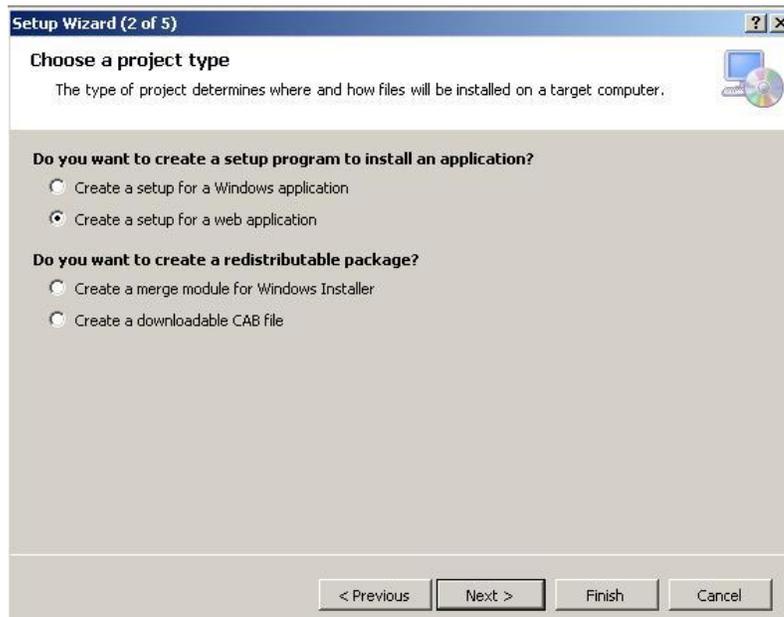


FIGURA 3.9 Creación de instaladores Paso 2.

Tercer paso: Escoger la información que se necesita de Salida para el instalador, por lo general solo se debería seleccionar la opción de salida principal del proyecto Web (Primary Output from BabelSoftware.Web), pero además, se necesita que esté el contenido de este proyecto (Content Files from BabelSoftware.Web) para enviar la carpeta de imágenes y archivos necesarios para la subida de imágenes.

Además, incluir el primary output del Proyecto Silverlight. Los datos de la opción XML Serialization es para la comunicación asíncrona que trae datos serializados.

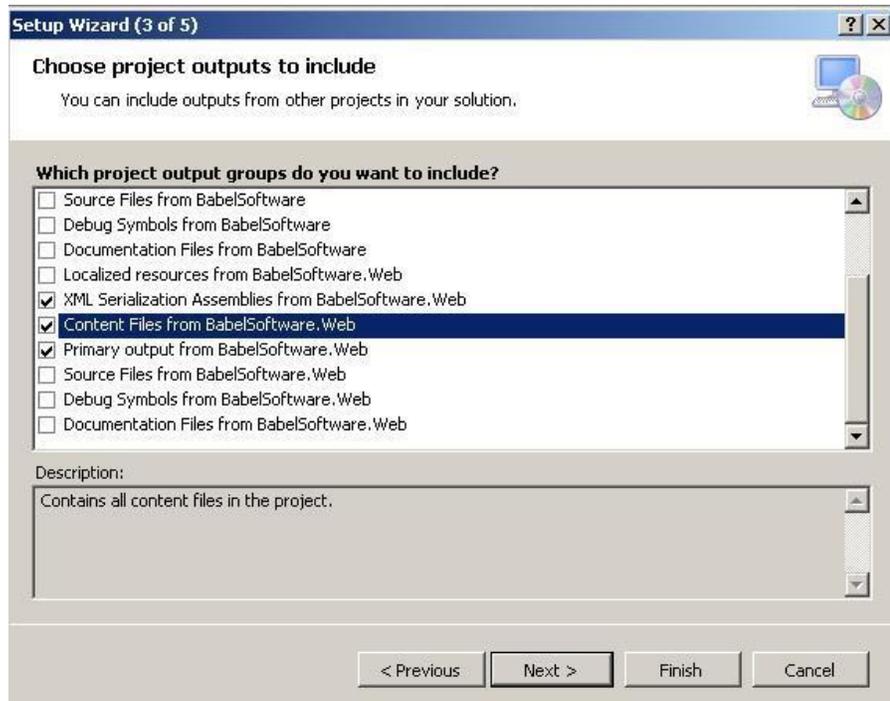


FIGURA 3.10 Creación de instaladores Paso 3.

En el caso del proyecto WCF se debe escoger también el contenido del Proyecto WCF para que se publiquen los servicios. Y solo los Primary Outputs de los proyectos BEU y BLL. Además, se debe constatar que las DLL de los proyectos BEU y BLL vayan en la carpeta BIN del instalador del WCF caso contrario agregarlas en el paso 4 de 5 del Setup Wizard.

Cuarto paso: se deberá escoger el tipo de compilación Release (esta opción optimiza al sitio para su máxima velocidad de respuesta)³² en vez de Build (esta es más lenta pero permite hacer pruebas (Debugging), por tal razón no se usa para crear los instaladores) y compilar el proyecto de tipo Setup.

³² http://www.ehu.es/mrodriguez/archivos/Articulos_alumnos/instalador.pdf

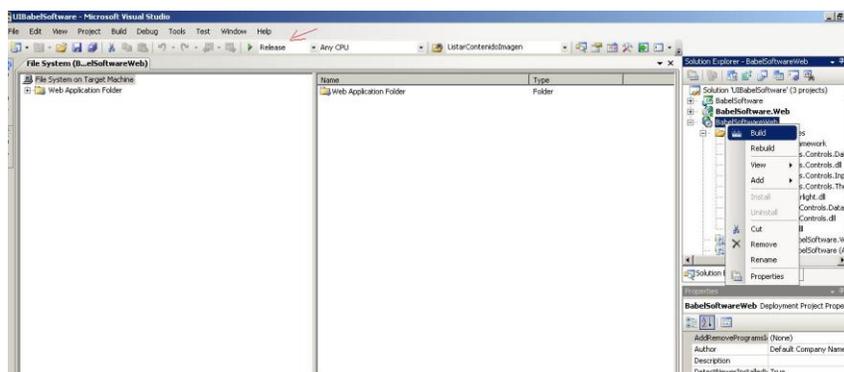


FIGURA 3.11 Creación de instaladores Paso 4.

Con esto se habrá creado en la carpeta Release, el instalador (Setup.exe), el cual se lo puede ejecutar en el servidor Web.

Se deberán instalar y publicar los servicios WCF, en un servidor Web (puede ser el mismo que se usó para instalar el sitio Web), en este caso Internet Information Server 6.0.

Dependiendo en que puerto se los publique (se puede escoger el puerto), se deberá configurar el archivo Web.Config con los Key (es un parámetro que permite acceder a una pieza de información)³³ que usa este sitio Web, esto se verá con detalle en el siguiente tema.

3.2.4.2. Configuración del servidor

Para esto se debe configurar el Servidor Web con las extensiones necesarias para Silverlight.

- .xap application/x-silverlight-app (Silverlight)
- .xaml application/xaml+xml (XAML)
- .xbap application/x-ms-xbap (WPF)

³³ <http://www.odetocode.com/Articles/345.aspx>

Primer paso: ir a Inicio/Herramientas Administrativas y ejecutar el Administrador de Internet Information Server



FIGURA 3.12 Configuración del Servidor Paso 1.

Segundo paso: dar clic derecho sobre el nombre del Servidor Web y seleccionar propiedades.

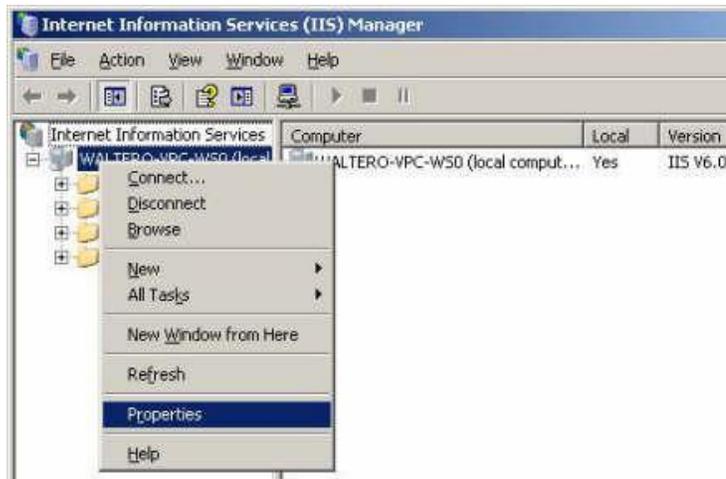


FIGURA 3.13 Configuración del Servidor Paso 2.

Tercer paso: dar clic en el botón de los MIME Types (estándar que especifica cómo debe un programa transferir cualquier archivo que no esté codificado en US-ASCII)³⁴.

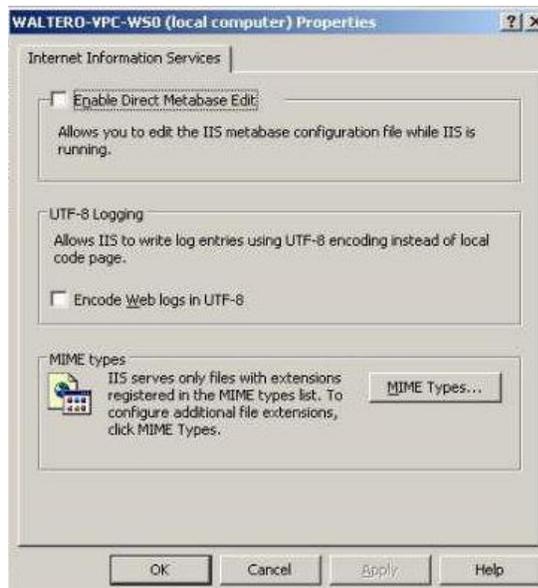


FIGURA 3.14 Configuración del Servidor Paso 3.

Cuarto paso: presionar el botón nuevo.

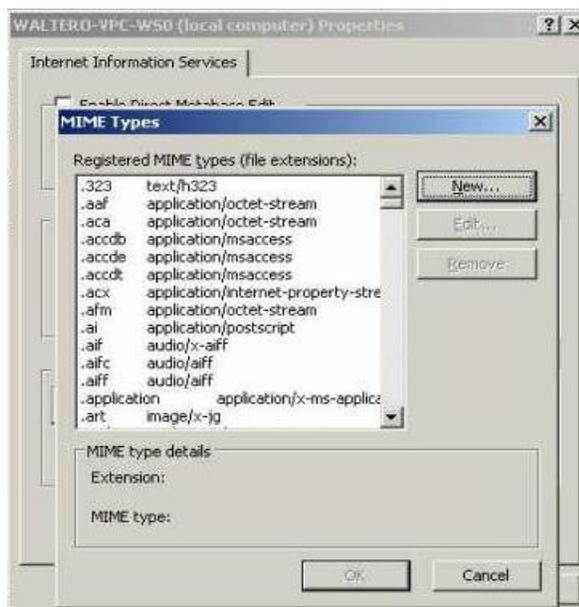


FIGURA 3.15 Configuración del Servidor Paso 4.

³⁴ http://en.wikipedia.org/wiki/Mime_types

Quinto paso: insertar las extensiones y MIME Types indicados anteriormente

- .xap application/x-silverlight-app (Silverlight)
- .xaml application/xaml+xml (XAML)
- .xbap application/x-ms-xbap (WPF)

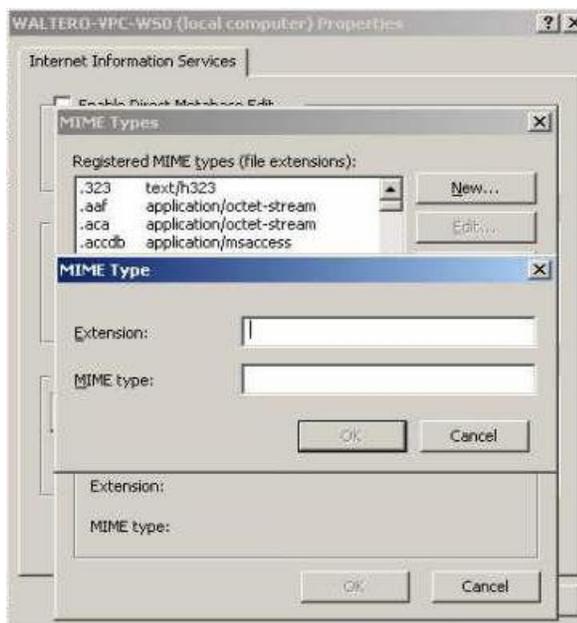


FIGURA 3.16 Configuración del Servidor Paso 5.

3.2.4.2.1. Configuración de Keys

Cambiar localhost:55887/ por los datos del servidor

Keys:

```
<add key="Contenido" value="http://localhost:55887/Contenido.svc"/>
<add key="ContenidoImagen" value="http://localhost:55887/ContenidoImagen.svc"/>
<add key="Sitemap" value="http://localhost:55887/Sitemap.svc"/>
<add key="Rol" value="http://localhost:55887/Rol.svc"/>
<add key="RolSitemap" value="http://localhost:55887/RolSitemap.svc"/>
<add key="Usuario" value="http://localhost:55887/Usuario.svc"/>
<add key="UsuarioSitemap" value="http://localhost:55887/UsuarioSitemap.svc"/>
```

```
<add key="RolUsuario" value="http://localhost:55887/RolUsuario.svc"/>  
<add key="Parrafo" value="http://localhost:55887/Parrafo.svc"/>  
<add key="Imagen" value="http://localhost:55887/Imagen.svc"/>  
<add key="Pais" value="http://localhost:55887/Pais.svc"/>  
<add key="Contacto" value="http://localhost:55887/Contacto.svc"/>  
<add key="Correo" value="http://localhost:55887/Correo.svc"/>
```

Nota.-El Key para mostrar imágenes y el que sirve para realizar la subida de imágenes al servidor, dependen del puerto del Sitio Web más no de los servicios. (Cambiar localhost: 1265/ por los datos del servidor)

```
<add key="KeyImágenes" value="http://localhost:1265/Images/" />  
<add key="KeyReceiver" value="http://localhost:1265/receiver.ashx"/>
```

Recordar que se debe habilitar el servidor SMTP, para el envío de mails desde el servidor y dar permisos a los usuarios del servidor, como son el Invitado de Internet y el usuario de internet del servidor, esto para permitir la subida de imágenes al mismo.

IV. Capítulo IV

RESULTADOS

4.1. CONCLUSIONES

- El presente proyecto evidenció el avance en el desarrollo de sitios web; la tendencia actual ha llevado a identificar dos elementos sustanciales que complementan la interacción de los sitios web con los usuarios: por un lado el progreso de la presentación; mediante mejoras en la arquitectura, usabilidad, arquitectura de la información, comunicaciones asíncronas, integración con lenguajes de programación, programación orientada a objetos, web services, agregados de plug in's gráficos y multimedios, mejoras en la conectividad, etc. ; y por otro lado, aún más importante, la oportunidad de los usuarios para aportar en los contenidos mediante blogs, foros, sistemas manejadores de contenidos, redes sociales, etc.
- El diseño de la interfaz fue fortalecida por la utilización del IDE Microsoft Expression Blend 2.0 para el lenguaje XAML, el mismo que facilita el desarrollo de estilos, animaciones y diseños por medio de una interfaz amigable.
- La utilización de XAML para el desarrollo de interfaces de usuario permite separar de forma ordenada la estructura del elemento diseñado: por un lado sus propiedades de diseño y por otro sus características de programación.
- Este sitio web es en esencia un Sistema Manejador de Contenidos; lo cual, le da un valor de flexibilidad en cuanto al manejo de la información que presenta, haciendo que la misma pueda ser actualizada de forma dinámica tanto a nivel de contenidos, categorías, y permisos de accesibilidad. Esta

fortaleza demuestra que el usuario que administra el sitio web no debe necesariamente tener altos conocimientos de informática.

- La estructura de organización de la información fue basada en un modelo de base de datos relacional, dicho modelo fue sumamente eficaz para determinar los niveles de accesibilidad y facilitó también la generación dinámica de menús y contenidos.
- En armonía con la arquitectura Microsoft, se utilizó la estructura de programación WCF, lo cual le da mayor adaptabilidad a la solución, ya que esta, eventualmente puede ser asociada a cualquier “front end” de distinta plataforma.
- La metodología UML respalda el desarrollo de software mediante la documentación, y evidencia la interacción de los usuarios [actores] con el sistema; esta documentación representada en casos de uso, ayuda al desarrollo del sitio web en cuanto a sus pantallas, llevando interiormente a la estandarización, y además permite construir un proceso de verificación de calidad.
- LINQ es un novedoso método mediante el cual se puede implementar un conjunto de clases que reflejan el modelo de datos. El uso de LINQ simplifica las tareas de inserción, eliminación, actualización y consulta mediante funciones estándares, con ello se reduce significativamente el tiempo de desarrollo.
- Los controles de usuario permiten la reutilización de código mediante la instancia; un control de usuario tiene asociado una estructura de software con métodos, propiedades y funciones que pueden utilizarse de manera genérica dependiendo de la necesidad en cualquier sitio del sistema.

- La ventaja de la aplicación RIA mediante la tecnología Silverlight de Microsoft radica en el orden de la disposición de código, la misma que asemeja una programación orientada a objetos, permitiendo que un desarrollador, sin experiencia, puede asimilarla fácilmente; contrario a Action Script de Adobe donde se debe tener un alto conocimiento de la arquitectura para explotar su funcionalidad.

4.2. RECOMENDACIONES

4.2.1. RECOMENDACIONES SOBRE EL DESARROLLO

- Planificar el desarrollo del proyecto software de tal forma que permita: tener un respaldo basado en documentos, identificar la meta del proyecto, establecer el avance en el desarrollo, distribuir las tareas y definir fechas e hitos entregables.
- Utilizar estándares de programación que ayuden a ordenar de forma efectiva al sitio Web, tanto la parte de diseño como en la parte lógica, estos pueden ser estándares preestablecidos, propios o adaptaciones de estándares, según la necesidad del proyecto y de sus ejecutantes.
- Establecer un proceso de identificación de necesidades de la organización, que permita definir el alcance y la arquitectura de información ideal del sitio web.
- Optimizar el tiempo de desarrollo de software mediante la utilización de IDE como Microsoft Blend, SQL Server Management Studio, Microsoft Visual Studio, lenguajes como LINQ que facilita y simplifica el acceso a datos y herramientas que faciliten la comunicación entre capas y brinden seguridades como WCF.

- Emplear UML haciendo que la descripción de cada requerimiento brinde al programador una visión muy clara de que es lo que va a programar y cuáles serán los resultados esperados.
- Crear y probar prototipos de software para toda la arquitectura del sistema con una funcionalidad básica y que permitan, posteriormente, incrementar las demás funcionalidades del proyecto.
- Diseñar una interfaz bajo recomendación de diseñadores gráficos, de tal suerte que no afecte a la vista del usuario y más bien sea motivo de elegancia y sobriedad, con una buena disposición de los elementos de interacción en pantalla.
- Desarrollar el producto software de tal forma que sea de fácil manejo y atractivo para los usuarios, sin procesos largos que lo cansen o aburran, sino con subprocesos que hagan de los procedimientos un tema interesante.
- Establecer un plan de pruebas donde el programador que desarrolló una parte del sistema no sea el mismo que la revisa (pruebas no dirigidas), esto siempre ayuda a encontrar falencias en otros casos imperceptibles.

4.2.2. RECOMENDACIONES SOBRE EL PRODUCTO SOFTWARE.

- Incrementar la funcionalidad del sistema CMS que permita un nexo más fuerte con los visitantes, como por ejemplo la implementación de un foro, un chat, o un repositorio de código, bolsa de trabajo, etc.

- Explotar las ventajas de las aplicaciones RIA para este proyecto mediante la inclusión de elementos multimedia como videos, sonidos, y animaciones.
- Convertir el presente proyecto en un framework para futuras aplicaciones tipo CMS que utilicen Microsoft Silverlight.
- Incluir la técnica SEO, Search Engine Optimization para posicionar el URL del sitio web en los primeros lugares de los buscadores de Internet.
- Incluir técnicas de seguridad más invulnerables para el manejo de las cuentas de usuario.
- Ampliar el idioma del sitio a inglés para tener un marco potencial de visitantes más alto.
- Agregar funcionalidad de acceso tipo WAI-ARIA para personas con habilidades especiales.

Referencia Bibliográfica.-

<http://www.techweb.com/encyclopedia/defineterm.jhtml?term=RIA&x=&y=>
<http://en.wikipedia.org/wiki/Silverlight>
<http://www.desarrolloweb.com/articulos/993.php>
<http://dojotoolkit.org/book/dojo-book-0-4/part-1-introduction/dojo-what-it>
<http://www.kirupa.com/developer/mx/sharedobjects.htm>
<http://www.roseindia.net/Technology-revolution/ria/rich-internet-application.shtml>
[http://msdn.microsoft.com/es-es/library/bb404700\(VS.95\).aspx](http://msdn.microsoft.com/es-es/library/bb404700(VS.95).aspx)
http://en.wikipedia.org/wiki/Site_map
<http://www.rioserver.com/comunidad/index.php/tutoriales/38-tutoriales-videos-software-aplicaciones-online/86-creacion-de-un-sitemap-para-google.html>
<http://www.alegsa.com.ar/Dic/usuario.php>
<http://es.wikipedia.org/wiki/Usuario>
<http://silverlight.net/learn/tutorials/UserControls.aspx>
<http://weblogs.asp.net/scottgu/pages/silverlight-tutorial-part-4-using-style-elements-to-better-encapsulate-look-and-feel.aspx>
http://en.wikipedia.org/wiki/Windows_Communication_Foundation
<http://www.rioserver.com/comunidad/index.php/tutoriales/38-tutoriales-videos-software-aplicaciones-online/86-creacion-de-un-sitemap-para-google.html>
<http://silverlight.net/learn/tutorials/UserControls.aspx>
<http://weblogs.asp.net/scottgu/pages/silverlight-tutorial-part-4-using-style-elements-to-better-encapsulate-look-and-feel.aspx>
<http://www.w3c.es/Consortio/>
http://en.wikipedia.org/wiki/Windows_Communication_Foundation
<http://learn.iis.net/>

ANEXOS

ANEXO1



Especificación de Requerimientos

Sitio web Babel Software Ecuador

Versión 1.0.0

2009-01-16

Carta de Aprobación

Por medio de la firma de esta carta de aceptación los requerimientos aquí descritos se aprueban y serán desarrollados. Se acepta que no quedan requerimientos fuera del alcance del proyecto.

Tnlg. Maribel Silva
Gerente – BABEL ECUADOR

Introducción

Este proyecto consiste en diseñar el sitio Web de Babel Software del Ecuador. Sin embargo; no basta con tener un sitio Web funcional y atractivo para los usuarios, sino que existe también un aspecto aún más importante que determinará el éxito del sitio Web y esto se llama: Marketing o Mercadeo Web, la cual tiene como misión, optimizar el sitio, a partir de estrategias que permitan información confiable y actualizada, poniendo especial hincapié en la calidad del trabajo, el cual se consigue con el máximo cuidado en los detalles.

Un sitio desfasado, dice poco de la empresa. En cambio uno actual, hará que los clientes vean que la empresa es dinámica y que está al día con las últimas tecnologías. De esta forma, los clientes tendrán una idea positiva de la empresa y podrá de esta forma conseguir, atraer y fidelizar a nuevos clientes con un buen rediseño Web.

La idea es tener en cuenta el objetivo final del sitio Web pasando los máximos controles de calidad en cuanto a su usabilidad y contenido.

Esto con el fin, de conseguir que el cliente potencial al entrar al sitio Web se sienta interesado por el contenido, encuentre rápido lo que busca y de esta manera confíe en la empresa, lo que puede traducirse en una venta para la misma.

Propósito

Una empresa, que se considere de vanguardia, debe poseer los beneficios de la tecnología Web y saberlas explotar de manera sustancial; es así como surge la necesidad de la empresa Babel Software Ecuador para diseñar su sitio Web acorde a las últimas tendencias en el Desarrollo Web y exponer al mercado sus clientes, productos, alianzas, proyectos y desarrollo.

La estrategia para cubrir la necesidad de Babel Software Ecuador su sitio Web, es diseñarlo mediante la utilización de tecnología Rich Internet Application RIA; la misma que permite crear sitios Web que emulan a las aplicaciones de escritorio, efectivizando además los tiempos de respuesta; y en donde prevalecen las características de dinamismo, flexibilidad, usabilidad, y navegabilidad.

Ámbito del Sistema

El proyecto contempla el desarrollo de controles de usuario, animaciones, manejador de contenidos, estilos de diseño XAML, uso de WCF y LINQ; además de las convencionales técnicas y características del desarrollo de software mediante la metodología UML.

Para cumplir con los objetivos del proyecto, es necesario definir las características del Sitio Web, en cuanto a información, estructura y contenidos; a fin de obtener un resultado de calidad, óptimo y confiable, que refleje fielmente lo que la empresa Babel representa dentro del ámbito del software a nivel nacional e internacional.

Beneficios

- Añadir una segunda puerta principal de acceso a la empresa.
- Será una especie de arte publicitario ofrecido las 24 horas al día y 7 días a la semana.

- Mejorar y posicionar la imagen de la empresa.
- Ofrecer servicios que la competencia no tiene.
- Comunicar las ventajas competitivas de la empresa.
- Ampliar el mercado desde un nivel local hasta un nivel nacional o internacional.
- Promover la comunicación con los clientes y proveedores.
- Optimizar la labor de Marketing, orientada a los objetivos propuestos (captura de clientes, comunicación efectiva, etc.)

Funciones del Producto

El presente proyecto explora los beneficios de la tecnología RIA en la construcción de Sitios Web tomando como ejemplo práctico el diseño en Babel Software del Ecuador.

Las principales funciones del sitio Web son:

- Actualizar toda la información que la empresa Babel Software del Ecuador desea mostrar a sus clientes o cualquier visitante.
- Mantener un nuevo medio de comunicación y contacto con posibles clientes.
- Demostrar que Babel Software del Ecuador usa tecnología de vanguardia hacia el público en general.

Objetivos

OBJ-001	<i>Presentar Información de la Empresa</i>
Descripción	El sitio web debe mostrar una página de bienvenida con un mensaje del presidente, sus diferentes sucursales y principales competencias.
Importancia	Alta
Comentarios	Es necesario dar a conocer esta información de una forma elegante, interactiva y llamativa.

OBJ-002	<i>Presentar Información de Productos y Servicios</i>
Descripción	El sitio Web debe mostrar información sobre los paquetes comerciales y servicios que ofrece la empresa.
Importancia	Alta
Comentarios	Mostrar lo que se produce en la empresa, es de principal importancia para llamar la atención del visitante.

OBJ-003	<i>Presentar Información de Noticias</i>
Descripción	El sitio Web debe mostrar información noticiosa, casos de éxitos y artículos de prensa.
Importancia	Alta
Comentarios	El objetivo del sitio Web es proporcionar información actual de la empresa

OBJ-004	<i>Contactar a Babel Software Ecuador con posibles clientes</i>
Descripción	El sitio Web debe permitir a los visitantes dejar comentarios o pedidos e información de teléfono o correo electrónico para poderlos contactar
Importancia	Alta
Comentarios	

Definiciones, Acrónimos y Abreviaturas

- RIA: Aplicaciones Web enriquecidas
- Microsoft SilverLight: Conjunto de librerías que permiten desarrollar RIA en los sitios Web
- Microsoft Blend: Herramienta robusta para diseño
- CMS: Sistema Manejador de Contenidos
- XML: Lenguaje de marcas extensible
- XAML: Lenguaje de Formato para Aplicaciones Extensibles
- EDS: Especificación de diseño de software
- ERS: Especificación de requisitos de software
- CU: Caso de Uso
- ACT: Actor
- RQF: Requerimientos funcionales
- RQR: Requerimientos no funcionales de rendimiento
- RTD: Requerimientos no funcionales de estilos de diseño
- FIA: Requerimientos no funcionales de funcionamiento correcto del sistema
- PRT: Requerimientos no funcionales de portabilidad.
- PRT: Requerimientos no funcionales de permisos de accesos.
- DSP: Requerimientos no funcionales de Disponibilidad
- RQN: Requerimientos no funcionales

Actores

ACT-001	Administrador (Web Máster)
Descripción	Un Web Máster es un funcionario de la empresa Babel Software con permisos de usuario administrador en el Sitio Web.
Comentarios	Encargado de registrar la información de proyectos y noticias; además de gestionar los registros de noticias, y administrar usuarios, categorías y permisos.

ACT-002	Visitante
Descripción	Un Visitante es cualquier persona que accede al sitio Web de Babel Software.
Comentarios	

ACT-003	Editor
Descripción	Un Editor es un funcionario de Babel Software con permisos para crear un contenido en el sitio Web.
Comentarios	

Suposiciones y Dependencias

SUP-001	Navegador Web
Descripción	Los navegadores comúnmente utilizados son Internet Explorer y Mozilla Firefox
Importancia	Alta
Comentarios	Silverlight está soportado en estos navegadores, en caso de que el sitio Web sea accesado por otro browser habrá funcionalidades y hasta páginas enteras que no serán desplegadas.

SUP-002	Servidor de Base de Datos
Descripción	Se debe contar con un servidor de Base de Datos
Importancia	Alta
Comentarios	Si no se tiene, puede afectar el rendimiento del sistema

SUP-003	Base de Datos
Descripción	El servidor de base de datos debe tener instalado SQL Server 2008.
Importancia	Alta
Comentarios	Si se cambia esto, provocará nuevos requerimientos

SUP-004	Sistema Operativo de los Servidores
Descripción	El sistema operativo para los servidores debe ser Windows 2003 Server con el servicio IIS instalado.
Importancia	Alta
Comentarios	Si se cambia el sistema operativo, provocará nuevos requerimientos

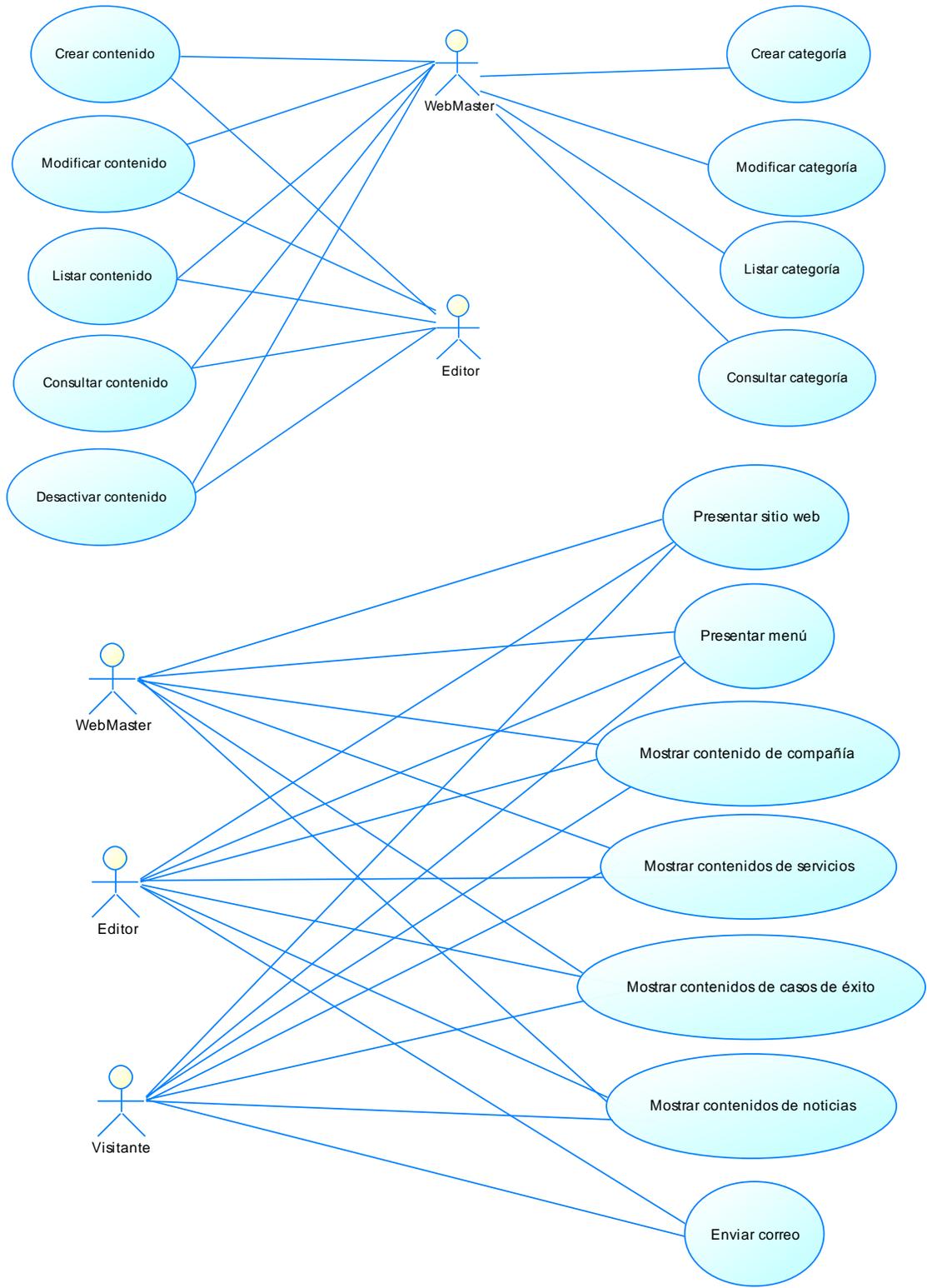
Requerimientos funcionales

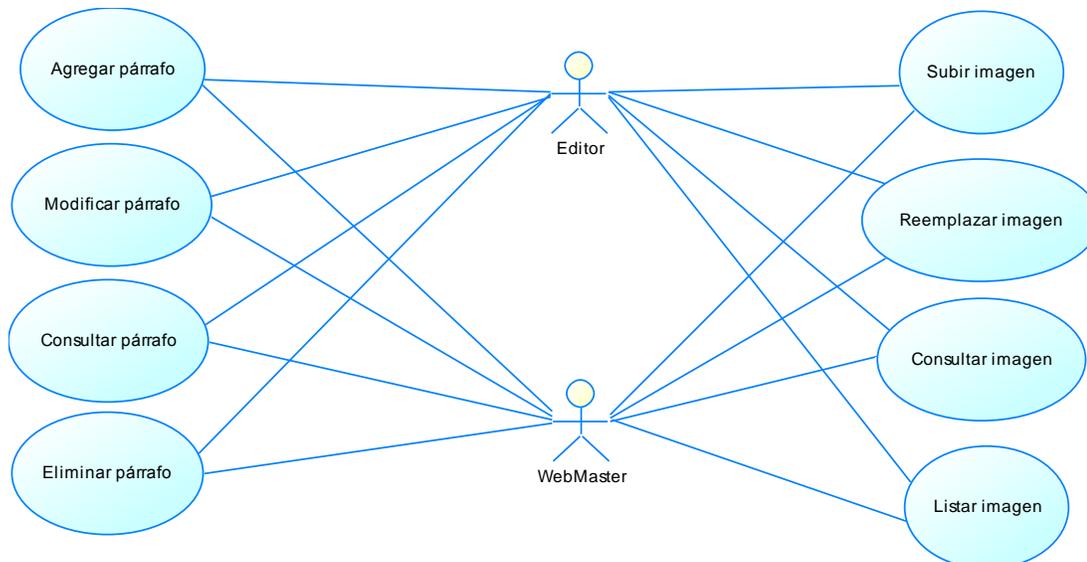
Este proyecto consiste en diseñar el sitio Web mostrando toda la información relevante de la empresa, como son los apartados de:

1. Compañía
2. Servicios
3. Noticias
4. Casos de éxito
5. Contáctenos
6. Administración

Se necesita además, en apartados como en el de casos de éxito, tener información de testimonios satisfechos, esto con el fin, de conseguir que el cliente potencial al entrar al sitio Web se sienta interesado por el contenido, encuentre rápido lo que busca y de esta manera confíe en la empresa, lo que puede traducirse en una venta para la misma.

Diagramas General de Casos de Uso





RQF-001	Presentar Sitio web
Descripción	El sitio web muestra información relevante de los servicios de la empresa, un saludo del Presidente, un menú de navegación, los logos de los partners, e información de teléfonos.
Objetivo	OBJ-001
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	La información sobre contactos y teléfonos juntos con los logos de la empresa y partners serán siempre visible.

CU-001	Presentar Sitio Web
Requerimientos	RQF-001
Casos de Uso Relacionados	Ninguno
Actores	ACT-001, ACT-002, ACT-003
Tipo	Primario
Descripción	En el inicio, el sitio Web mostrará el mensaje "Hablamos el lenguaje de su empresa", "Latacunga cuna del software", el logo de Babel Software, el logo de el partner Microsoft, un apartado con el saludo inicial del Presidente, una descripción de los principales servicios de la empresa (estas descripciones son dinámicas), un menú de navegación dinámico, y la información de teléfonos de contacto.

RQF-002	Presentar Menú
Descripción	El sitio Web presentará un menú en la parte superior izquierda, en donde se mostrarán links de acceso hacia los contenidos.
Objetivo	OBJ-001
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	El menú es dinámico y solo listará los ítems activos

CU-002	Presentar Menú
Requerimientos	RQF-002
Casos de Uso Relacionados	CU-001
Actores	ACT-001, ACT-002, ACT-003
Tipo	Primario
Descripción	La estructura del menú en una primera instancia es: compañía, servicios, casos de éxito, noticias, contáctenos y administración. El menú debe ser dinámico.

RQF-003	Crear categoría
Descripción	El sistema debe permitir la creación de categorías de contenidos, las mismas que pueden visualizarse a través del menú de navegación.
Objetivo	OBJ-001
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Se debe controlar que no se inserten categorías con el mismo título. Cuando se cree una nueva categoría permitir si el usuario desea, que sea accesible para todos.

CU-003	Crear categoría
Requerimientos	RQF-003
Casos de Uso Relacionados	CU-040
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario ingresará los campos para crear una nueva categoría: Título, Descripción, Tipo, Estado, y URL. La URL para contenidos dinámicos va hacer insertada de forma imperceptible para el usuario y en el caso de que se creen nuevas pantallas en dicha instalación, la URL de la nueva pantalla deberá ser ingresada de forma manual por el administrador o editor con permisos, para esto deberá existir dicha opción.

RQF-004	Consultar categoría
Descripción	El sistema debe permitir modificar categorías de contenidos.
Objetivo	OBJ-001
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-004	Consultar categoría
Requerimientos	RQF-004
Casos de Uso Relacionados	CU-006
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario selecciona la categoría a modificar y actualiza los campos para

	crear una nueva categoría: Título, Descripción, Tipo, Estado, y URL.
--	--

RQF-005	Modificar categoría
Descripción	El sistema debe permitir modificar categorías de contenidos.
Objetivo	OBJ-001
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Se debe controlar que no se cambie el título a uno ya creado. (no repetidos)

CU-005	Modificar categoría
Requerimientos	RQF-005
Casos de Uso Relacionados	CU-004
Actores	ACT-001, ACT-003
Tipo	Secundario
Descripción	El usuario selecciona la categoría a modificar y actualiza los campos para crear una nueva categoría: Título, Descripción, Tipo, Estado, y URL.

RQF-006	Listar categoría
Descripción	El sistema debe permitir listar categorías de contenidos.
Objetivo	OBJ-001
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-006	Listar categoría
Requerimientos	RQF-006
Casos de Uso Relacionados	CU-040
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario seleccionar la opción para listar categorías, y el sistema le muestra todas las categorías guardadas para el sitio Web.

RQF-007	Mostrar contenido de compañía.
Descripción	El sistema debe mostrar un contenido para la categoría de Compañía.
Objetivo	OBJ-001
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Cada uno de los contenidos pueden contener fotografía alusiva y descripción.

CU-007	Mostrar contenido de compañía.
Requerimientos	RQF-007
Casos de Uso Relacionados	CU-001, CU-002
Actores	ACT-001, ACT-002, ACT-003

Tipo	Primario
Descripción	Este apartado contendrá información sobre las oficinas corporativas de Babel Software en Costa Rica, Ecuador y Estados Unidos, además de un mensaje de bienvenida por parte del presidente junto con su foto, una reseña histórica de la empresa y la parte del perfil corporativo de la empresa con: misión, visión, valores y enfoque de negocios.

RQF-008	Mostrar contenidos de servicios
Descripción	El sistema debe mostrar un contenido para la categoría de Servicios
Objetivo	OBJ-002
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Cada uno de los servicios pueden contener fotografía alusiva y descripción.

CU-008	Mostrar contenido de servicios.
Requerimientos	RQF-008
Casos de Uso Relacionados	CU-001, CU-002
Actores	ACT-001, ACT-002, ACT-003
Tipo	Primario
Descripción	Este apartado contendrá información sobre los servicios que ofrece Babel Software en el ámbito informático: Desarrollo a la media, Outsourcing, inteligencia de negocios, Software Factory y Soporte de Aplicaciones.

RQF-009	Mostrar contenidos de casos de éxito
Descripción	El sistema debe mostrar un contenido para la categoría de Casos de éxito.
Objetivo	OBJ-002
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Cada uno de los casos de éxito pueden contener fotografía alusiva, descripción. Los contenidos estarán ordenados cronológicamente y existirá un filtro para las fecha de inicio y fecha final.

CU-009	Mostrar contenido de casos de éxito.
Requerimientos	RQF-009
Casos de Uso Relacionados	CU-001, CU-002
Actores	ACT-001, ACT-002, ACT-003
Tipo	Primario
Descripción	Este apartado contendrá testimonios de los clientes de Babel Software a fin de incrementar la confianza y credibilidad de los potenciales clientes.

RQF-010	Mostrar contenidos de noticias
Descripción	El sistema debe mostrar un contenido para la categoría de Noticias
Objetivo	OBJ-002
Importancia	Alta
Estado	Aprobado

Estabilidad	Alta
Comentario	Cada una de las noticias pueden contener fotografía alusiva, descripción.

CU-010	Mostrar contenido de noticias.
Requerimientos	RQF-010
Casos de Uso Relacionados	CU-001, CU-002
Actores	ACT-001, ACT-002, ACT-003
Tipo	Primario
Descripción	Este apartado contendrá reportajes alusivos al área informática, las noticias estarán ordenadas cronológicamente, y existirá un filtro para las fecha de inicio y fecha final.

RQF-011	Crear contenido
Descripción	El sistema debe permitir crear contenidos para mostrarlos en el sitio web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Pueden existir contenidos con el mismo título, pues dependen de su fecha de creación, autor, categoría a la cual pertenezcan, imagen y párrafos. Enviar un correo al administrador del sistema, indicando que se creó un nuevo contenido con el Título, Categoría y el Usuario que creó el contenido.

CU-011	Crear contenido
Requerimientos	RQF-011
Casos de Uso Relacionados	CU-040
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario ingresará los siguientes campos de contenido: Título, Resumen, Estado y Categoría, además se guardarán automáticamente la Fecha de Creación, Fecha de Modificación, y Usuario que creó el contenido. El Estado puede ser Activo, Inactivo o Portada, en este caso este contenido se mostrará en la página de inicio.

RQF-012	Consultar contenido
Descripción	El sistema debe permitir consultar un contenido.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Corregir posibles errores que pudieron ser subidos sin una correcta revisión o actualizar datos en una misma noticia.

CU-012	Consultar contenido
Requerimientos	RQF-012
Casos de Uso Relacionados	CU-014

Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario consulta un contenido. Cuando se consulte un contenido desde las secciones de mostrar contenidos, se sumará en 1 el campo DesplegadoContenido, para registrar cuantas veces ha sido leído.

RQF-013	Modificar contenido
Descripción	El sistema debe permitir modificar los campos de un contenido del sitio Web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Corregir posibles errores que pudieron ser subidos sin una correcta revisión o actualizar datos en una misma noticia.

CU-013	Modificar contenido
Requerimientos	RQF-013
Casos de Uso Relacionados	CU-012
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	Este apartado contendrá reportajes alusivos al área informática, las noticias estarán ordenadas cronológicamente, y existirá un filtro de para las fecha de inicio y fecha final.

RQF-014	Listar contenido
Descripción	El sistema debe permitir listar todos los contenidos que estén alojados en el sistema.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Inactivar información obsoleta o que ya no sea de interés para la empresa. Además, se podrá colocar información en la portada.

CU-014	Listar contenido
Requerimientos	RQF-014
Casos de Uso Relacionados	CU-040
Actores	ACT-001, ACT-002, ACT-003
Tipo	Primario
Descripción	El listado puede ser filtrado por categoría, estado, título y usuario. Existe un segundo listar contenido que es dinámico, en el cual se muestran los contenidos de cada categoría y estarán ordenadas cronológicamente o por más leídos. Todos los mostrar contenido lo harán por fecha o por orden de más leídos, con un máximo de 20 contenidos.

RQF-015	Desactivar contenido
Descripción	El sistema debe permitir desactivar un contenido para que no pueda ser visto dentro del sitio Web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Inactivar información obsoleta o que ya no sea de interés para la empresa. El contenido puede ser desactivado por un usuario Web Máster o el Editor del contenido, en este caso, éste debe ser su autor.

CU-015	Desactivar contenido
Requerimientos	RQF-015
Casos de Uso Relacionados	CU-001, CU-002, CU-013
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El estado de un contenido puede ser actualizado a un estado desactivado para que no sea mostrado en el sitio Web.

RQF-016	Enviar correo (Contáctenos)
Descripción	El sistema debe permitir el envío de mensajes por correo electrónico de parte de un visitante hacia el web Máster. Además, esta información se guardará en la base de datos.
Objetivo	OBJ-004
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Se deberá enviar un email con esta información a infoecuador@grupobabel.com

CU-016	Enviar correo
Requerimientos	RQF-016
Casos de Uso Relacionados	Ninguno
Actores	ACT-002
Tipo	Secundario
Descripción	Este apartado pedirá al visitante: su nombre, correo electrónico, Teléfono o Fax, País, cómo desea ser contactado (Teléfono, Correo), Asunto, Comentario.

RQF-017	Agregar párrafo
Descripción	El sistema debe permitir agregar párrafo asociados a un contenido.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Debido a que el párrafo es un campo con contenido extenso no se deberán controlar repetidos.

CU-017	Agregar párrafo
Requerimientos	RQF-017
Casos de Uso Relacionados	CU-011
Actores	ACT-001, ACT-003
Tipo	Secundario
Descripción	El usuario agrega un párrafo asociado a un contenido, definiendo el texto.

RQF-018	Consultar párrafo
Descripción	El sistema debe permitir consultar párrafo asociados a un contenido.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-018	Consultar párrafo
Requerimientos	RQF-018
Casos de Uso Relacionados	CU-017
Actores	ACT-001, ACT-003
Tipo	Secundario
Descripción	El usuario consulta un párrafo asociado a un contenido.

RQF-019	Modificar párrafo
Descripción	El sistema debe permitir modificar un párrafo asociados a un contenido.
Objetivo	OBJ-005
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-019	Modificar párrafo
Requerimientos	RQF-019
Casos de Uso Relacionados	CU-018
Actores	ACT-001, ACT-003
Tipo	Secundario
Descripción	El usuario actualiza el texto del párrafo.

RQF-020	Eliminar párrafo
Descripción	El sistema debe permitir eliminar un párrafo asociados a un contenido.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-020	Eliminar párrafo
Requerimientos	RQF-020
Casos de Uso Relacionados	CU-018
Actores	ACT-001, ACT-003
Tipo	Secundario
Descripción	El usuario puede eliminar un párrafo asociado a un contenido.

RQF-021	Subir imagen
Descripción	El sistema debe permitir subir imágenes hacia el servidor.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	No pueden existir dos imágenes con el mismo nombre. Las imágenes permitidas son *.jpg, *.png.

CU-021	Subir imagen
Requerimientos	RQF-021
Casos de Uso Relacionados	Ninguno
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario selecciona una imagen de su disco duro, que puede subirse hacia el servidor web. (*.jpg, *.png)

RQF-022	Reemplazar imagen
Descripción	El sistema debe permitir reemplazar una imagen previamente subida al servidor por otra.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	No pueden existir dos imágenes con el mismo nombre

CU-022	Reemplazar imagen
Requerimientos	RQF-022
Casos de Uso Relacionados	CU-021, CU-023
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario selecciona la imagen a reemplazar y sube otra que la reemplazará.

RQF-023	Consultar imagen
Descripción	El sistema debe permitir consultar imagen previamente subida al servidor por otra.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta

Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-023	Consultar imagen
Requerimientos	RQF-023
Casos de Uso Relacionados	CU-024
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario selecciona la imagen de un listado.

RQF-024	Listar imagen
Descripción	El sistema debe permitir listar todas las imágenes subidas al servidor.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-024	Listar imagen
Requerimientos	RQF-024
Casos de Uso Relacionados	
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario visualiza el listado de imágenes.

RQF-025	Crear usuario
Descripción	El sistema debe permitir crear usuarios para interactuar con el sitio web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Cada usuario tendrá su rol y sus respectivos permisos de acceso. Se debe controlar que no se inserten usuarios con el mismo Nombre de Usuario.

CU-025	Crear usuario
Requerimientos	RQF-025
Casos de Uso Relacionados	
Actores	ACT-001
Tipo	Primario
Descripción	Cada usuario será definido con los siguientes campos: Nombres, Apellidos, Nombre de usuario, Contraseña, y Estado.

RQF-026	Consultar usuario
Descripción	El sistema debe permitir consultar un usuario registrado.
Objetivo	OBJ-001, OBJ-002, OBJ-003

Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-026	Consultar usuario
Requerimientos	RQF-026
Casos de Uso Relacionados	CU-028
Actores	ACT-001
Tipo	Primario
Descripción	El Web Máster selecciona el usuario a consultar desde la lista correspondiente, inmediatamente se le presentará la información asociada al usuario consultado.

RQF-027	Modificar usuario
Descripción	El sistema debe permitir modificar un usuario previamente registrado en el sitio web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	No se puede modificar el nombre de usuario, éste será usado como su referencia.

CU-027	Modificar usuario
Requerimientos	RQF-027
Casos de Uso Relacionados	CU-026
Actores	ACT-001
Tipo	Primario
Descripción	El Web Máster selecciona el usuario a modificar y actualiza sus campos.

RQF-028	Listar usuarios
Descripción	El sistema debe permitir listar los usuarios registrados en el sitio web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-028	Listar usuario
Requerimientos	RQF-028
Casos de Uso Relacionados	
Actores	ACT-001
Tipo	Primario
Descripción	El listado de usuarios puede ser filtrado por Nombres, Apellidos, Nombre de usuario, y Estado.

RQF-029	Crear rol
Descripción	El sistema debe permitir crear roles para asociarlos a los usuarios.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Cada rol tendrá sus respectivos permisos de acceso. Se debe controlar que no se inserten roles con el mismo nombre. Solo los administradores pueden crear roles.

CU-029	Crear rol
Requerimientos	RQF-029
Casos de Uso Relacionados	
Actores	ACT-001
Tipo	Primario
Descripción	Cada rol será definido con los siguientes campos: Nombre y Estado.

RQF-030	Consultar rol
Descripción	El sistema debe permitir consultar un rol.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-030	Consultar rol
Requerimientos	RQF-030
Casos de Uso Relacionados	CU-032
Actores	ACT-001
Tipo	Primario
Descripción	El Web Máster selecciona el rol de la lista correspondiente, e inmediatamente se le presentará la información asociada.

RQF-031	Modificar rol
Descripción	El sistema debe permitir modificar un rol previamente registrado en el sitio web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Se debe controlar que no se cambie el rol a uno ya creado. (no repetidos)

CU-031	Modificar rol
Requerimientos	RQF-031
Casos de Uso Relacionados	CU-030
Actores	ACT-001
Tipo	Primario

Descripción	El Web Máster selecciona el rol a modificar y actualiza sus campos.
--------------------	---

RQF-032	Listar roles
Descripción	El sistema debe permitir listar los roles registrados en el sitio web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-032	Listar roles
Requerimientos	RQF-032
Casos de Uso Relacionados	
Actores	ACT-001
Tipo	Primario
Descripción	El listado de roles puede ser filtrado por Nombre y Estado.

RQF-033	Asociar usuario a categoría.
Descripción	El sistema debe permitir asociar las categorías a un usuario para que pueda acceder a ellos.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Los editores no pueden modificar permisos de los administradores.

CU-033	Asociar usuario a categoría
Requerimientos	RQF-033
Casos de Uso Relacionados	CU-006, CU-028
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario elige un usuario de un listado y lo asocia a las categorías seleccionados del listado correspondiente.

RQF-034	Asociar usuario a rol.
Descripción	El sistema debe permitir asociar un rol a un usuario.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Los editores no pueden modificar permisos de los administradores

CU-034	Asociar usuario a rol.
Requerimientos	RQF-034
Casos de Uso Relacionados	CU-028, CU-032
Actores	ACT-001, ACT-003

Tipo	Primario
Descripción	El usuario elige un usuario de un listado y lo asocia a un rol.

RQF-035	Asociar rol a categorías.
Descripción	El sistema debe permitir asociar un rol a categorías.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Si se cambian los permisos de un rol, preguntar al usuario, si desea que los cambios se apliquen en los usuarios que tengan dicho rol también. Los editores no pueden modificar permisos de los administradores.

CU-035	Asociar rol a categorías.
Requerimientos	RQF-039
Casos de Uso Relacionados	CU-006, CU-032
Actores	ACT-001, ACT-003
Tipo	Primario
Descripción	El usuario elige un rol de un listado y lo asocia a las categorías seleccionados del listado correspondiente.

RQF-036	Iniciar sesión.
Descripción	El sistema debe permitir iniciar sesión con sus diferentes usuarios.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	Iniciar sesión solo podrán hacerlo usuarios activos.

CU-036	Iniciar sesión.
Requerimientos	RQF-040
Casos de Uso Relacionados	CU-008, CU-001, CU-002
Actores	ACT-002
Tipo	Primario
Descripción	El usuario ingresa su Nombre de usuario y Contraseña, al sistema, y muestra un mensaje de bienvenida si los parámetros están correctos.

RQF-037	Consultar correos recibidos (Consultar Contactos)
Descripción	El sistema debe permitir consultar los correos recibidos.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-037	Consultar correos recibidos
Requerimientos	RQF-037
Casos de Uso Relacionados	CU-038
Actores	ACT-001
Tipo	Primario
Descripción	El Web Máster selecciona el correo de la lista correspondiente, e inmediatamente se le presentará la información asociada.

RQF-038	Listar correos recibidos (Listar Contactos)
Descripción	El sistema debe permitir listar los correos recibidos en el sitio web.
Objetivo	OBJ-001, OBJ-002, OBJ-003
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

CU-038	Listar correos recibidos
Requerimientos	RQF-038
Casos de Uso Relacionados	
Actores	ACT-001
Tipo	Primario
Descripción	El listado de correos puede ser filtrado por nombre, asunto, email, país y por como desea ser contactado el usuario

Requerimientos no funcionales

Requerimientos de Rendimiento

RQR-001	Tiempo de despliegue de páginas
Descripción	El tiempo de despliegue de cada página no debe durar más de un minuto.
Objetivo	NA
Importancia	Alta
Estado	Aprobada
Estabilidad	Media
Comentario	

RQR-001	Tiempo de respuesta del registro en un mantenimiento de datos
Descripción	El tiempo de respuesta del registro en un mantenimiento de datos debe estar en un rango de uno a cincuenta segundos.
Objetivo	NA
Importancia	Alta
Estado	Aprobada
Estabilidad	Baja
Comentario	

RQR-003	Proceso de consultas de la aplicación
Descripción	El proceso de consultas de la aplicación debe tener un tiempo máximo de 45 segundos.
Objetivo	NA
Importancia	Alta
Estado	Aprobada
Estabilidad	Baja
Comentario	

RQR-004	Disponibilidad del sistema
Descripción	El sistema debe estar disponible 24 / 7.
Objetivo	NA
Importancia	Alta
Estado	Aprobada
Estabilidad	Baja
Comentario	

RQR-005	Cantidad de usuarios en el sistema
Descripción	Al ser un sistema Web debe soportar como mínimo 100 usuarios simultáneos
Objetivo	NA
Importancia	Alta
Estado	Aprobada
Estabilidad	Baja
Comentario	

Restricciones de Diseño

RTD-001	Estilo del sistema
Descripción	El sistema debe ser un sitio Web RIA (Rich internet application).
Referencia	NA
Importancia	Alta
Estado	Aprobada
Estabilidad	Alta
Comentario	Deberá contener los colores de Babel Software (Azul y Blanco)

RTD-002	Navegadores de Internet a utilizar
Descripción	El sistema debe navegar en Internet Explorer 6 o superior y Firefox 2 o superior. No se garantiza el buen funcionamiento del sistema en versiones anteriores a IE7 o en otros navegadores de Internet.
Referencia	NA
Importancia	Alta
Estado	Aprobada
Estabilidad	Media
Comentario	

RTD-003	Sistema Operativo
Descripción	El sistema operativo a utilizar es Windows XP SP1 o superiores
Referencia	http://www.microsoft.com/silverlight/resources/install.aspx#sysreq

	http://msdn.microsoft.com/en-us/library/ms143506.aspx
Importancia	Alta
Estado	Aprobada
Estabilidad	Alta
Comentario	

RTD-004	Herramienta de desarrollo
Descripción	El sistema va a ser realizado utilizando la herramienta de Desarrollo Microsoft Visual Studio 2008 y Microsoft Expression Blend 2.0
Referencia	NA
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

RTD-005	Servidor de base de datos
Descripción	El sistema debe tener un servidor de base de datos que tenga instalado SQL Server 2008.
Referencia	NA
Importancia	Alta
Estado	Aprobado
Estabilidad	Alta
Comentario	

RTD-006	Despliegue
Descripción	El sistema requiere una resolución VGA o superior, utilizando como resolución mínima de 1024x768 pixeles.
Referencia	http://msdn.microsoft.com/en-us/library/ms143506.aspx
Importancia	Alta
Estado	Aprobada
Estabilidad	Alta
Comentario	No se garantiza que en pantallas de mucho menor resolución el sistema se despliegue con la misma calidad.

Atributos del Sistema

FIA-001	Funcionamiento correcto del sitio
Descripción	Para asegurar el funcionamiento correcto del sitio se contará con una etapa completa de pruebas exhaustivas, esto para que antes de que el sitio se publique, se hayan depurado y corregido todos los problemas presentes.
Importancia	Alta
Estabilidad	Alta
Comentario	

PRT-002	Portabilidad del sistema
Descripción	El sistema será desarrollado en una plataforma Web, lo cual asegura que el mismo se pueda ver o utilizar en varios sistemas operativos sin problemas (Lo único que se necesita es un explorador Web).

Importancia	Alta
Estabilidad	Alta
Comentario	

SGD-003	Usuarios
Descripción	Al sistema lo podrán usar usuarios invitados y registrados.
Importancia	Alta
Estabilidad	Alta
Comentario	Los usuarios que tengan permisos para crear contenidos, no necesariamente deberán tener conocimientos informáticos, incluso lo podrán realizar usuarios con conocimientos básicos.

SGD-004	Permiso de funcionalidad
Descripción	El Web Máster del sistema dará permiso a usuarios para diversas Categorías.
Importancia	Alta
Estabilidad	Alta
Comentario	N/A

DSP-005	Disponibilidad del sistema
Descripción	El sistema siempre debe estar disponible para los usuarios, sin importar la época del año.
Importancia	Alta
Estabilidad	Alta
Comentario	

Otros Requerimientos

RQN-001	Idioma del sistema
Descripción	El sistema debe estar escrito bajo el idioma español.
Objetivo	NA
Importancia	Baja
Estado	Pendiente de Aprobación
Estabilidad	Alta
Comentario	N/A

ANEXO 2



Especificación de Diseño Físico

Sitio web Babel Software Ecuador
Versión 1.0.0

2009-07-12

Carta de Aprobación

Por medio de la firma de esta carta de aceptación los requerimientos aquí descritos se aprueban y serán desarrollados. Se acepta que no quedan requerimientos fuera del alcance del proyecto.

Tnlg. Maribel Silva
Gerente – BABEL ECUADOR

Introducción

En este documento se detallan los casos de uso de alto nivel especificados en el documento de requerimientos (ERS).

La documentación presente en este documento detalla aspectos de programación como son los casos de uso extendido complementándose con los diagramas de interacción detallados por cada uno de los casos de uso desarrollados. Esta información se presenta detallada por cada uno de los módulos que componen el sitio Web.

Propósito

En este documento se extiende el contenido de los casos de uso de alto nivel del documento de requerimientos (ers), con los casos de uso extendidos y los diagramas de interacción.

Los casos de uso extendido están dirigidos a los usuarios encargados de los módulos y patrocinador del sistema.

Los casos de uso extendido, conjuntamente con los diagramas de interacción están dirigidos a líderes del proyecto, analistas, desarrolladores y cualquier persona que desee consultar el documento.

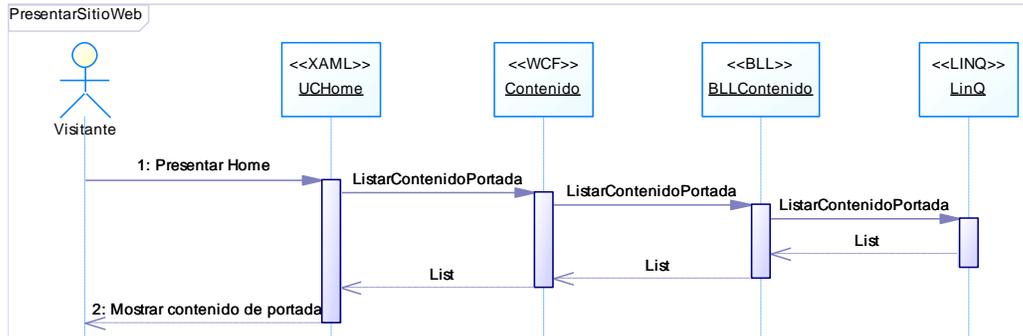
Definiciones, Acrónimos y Abreviaturas

- EDS: Especificación de diseño de software
- ERS: Especificación de requisitos de software
- CU: Caso de Uso
- ACT: Actor
- RQF: Requerimientos funcionales
- RQR: Requerimientos no funcionales de rendimiento
- RTD: Requerimientos no funcionales de estilos de diseño
- FIA: Requerimientos no funcionales de funcionamiento correcto del sistema
- PRT: Requerimientos no funcionales de portabilidad.
- PRT: Requerimientos no funcionales de permisos de accesos.
- DSP: Requerimientos no funcionales de Disponibilidad
- RQN: Requerimientos no funcionales

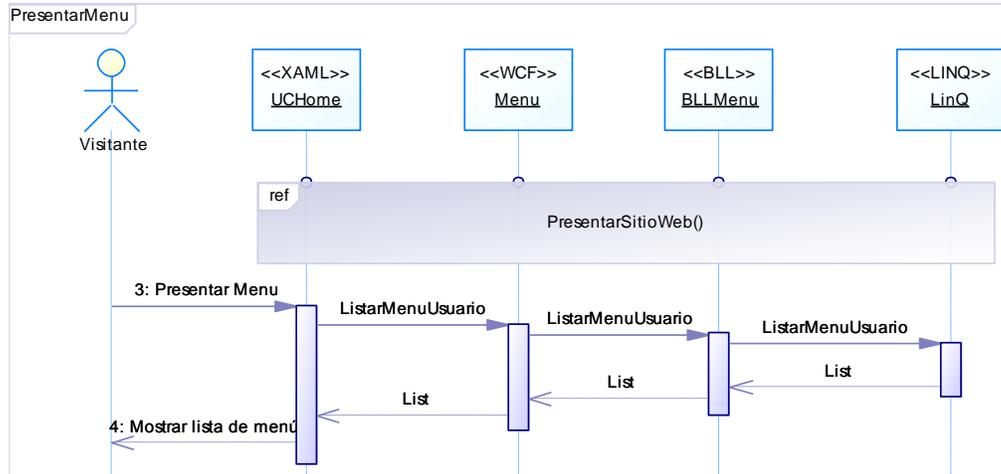
Visión General del Documento

El apartado de algoritmos y su respectiva información se presenta agrupado por cada uno de los módulos que componen el Sitio. A continuación se encuentra el apartado del diccionario de datos, donde se detalla la estructura física del repositorio de los datos.

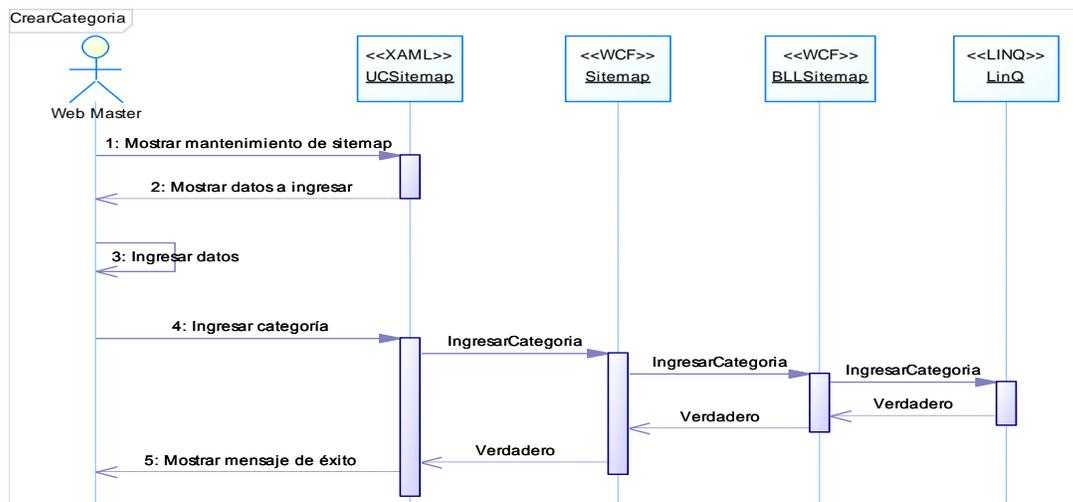
Algoritmos



CU-001	Presentar Sitio web	
Requerimientos	RQF-001	
Casos de Uso Relacionados		
Actores	ACT-001, ACT-002, ACT-003	
Tipo	Primario	
Descripción	El sistema debe presentar el sitio web con una página denominada Home.	
Precondiciones	1. El Visitante debe tener un navegador web instalado en su computador.	
Postcondiciones	1. El Visitante visualiza el sitio web.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando el Visitante digita la dirección URL www.babel.com.ec en su navegador.	2. El sitio web muestra información relevante de los servicios de la empresa, un saludo del Presidente, un menú de navegación, los logos de los partners, e información de teléfono.
	Curso Alternativo:	
	Acción 1: El Visitante cancela la operación. El navegador no muestra ninguna página web.	
	Acción 2: Se presenta un fallo de conexión. El navegador no muestra ninguna página web.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	

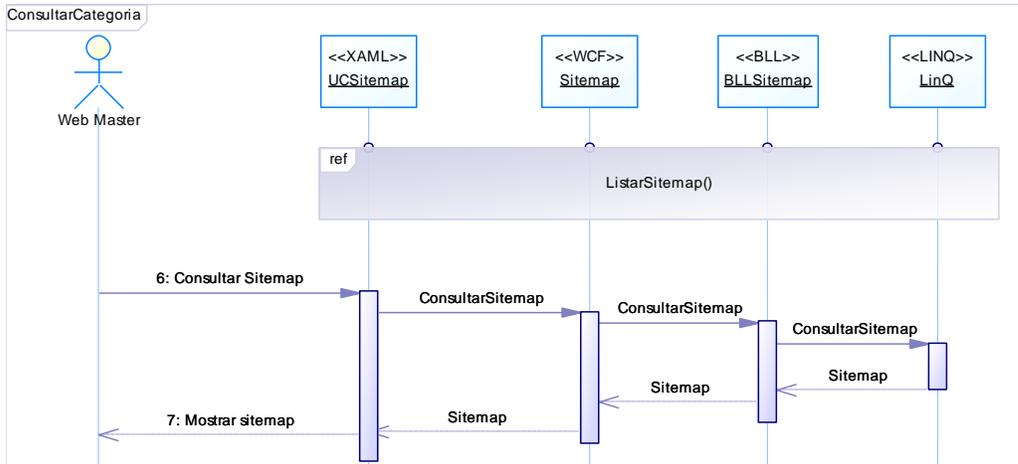


CU-002	Presentar Menú	
Requerimientos	RQF-002	
Casos de Uso Relacionados	CU-040	
Actores	ACT-001, ACT-002, ACT-003	
Tipo	Primario	
Descripción	El sistema debe presentar el menú junto con la página del Home.	
Precondiciones	1. El visitante debe llamar a la página de Home.	
Postcondiciones	1. El usuario visualiza el menú principal.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando el usuario digita la dirección URL www.babel.com.ec en su navegador.	2. El sitio web muestra el menú principal con la estructura: compañía, servicios, casos de éxito, noticias, y contáctenos.
	Curso Alternativo:	
	Acción 1: El usuario cancela la operación. El navegador no muestra ninguna página web.	
	Acción 1: El usuario inicia la sesión. El sistema genera las opciones de menú de acuerdo a su perfil y permisos.	
	Acción 2: Se presenta un fallo de conexión. El navegador no muestra ninguna página web.	
Importancia	Alta	
Comentarios	Recordar que el menú es dinámico y solo listará los ítems activos.	
Acción	Salida	

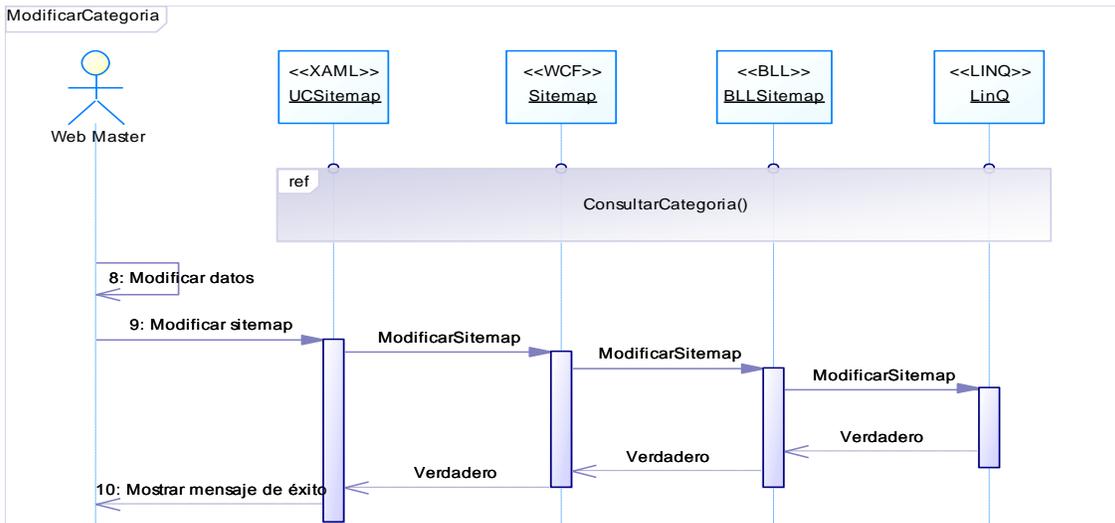


CU-003	Crear categoría	
Requerimientos	RQF-003	
Casos de Uso Relacionados	CU-040	
Actores	ACT-001, ACT-003	
Tipo	Primario	
Descripción	El sitio web debe permitir el ingreso de una categoría.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. La categoría queda registrada en el sitio web.	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de categorías y decide ingresar una nueva.</p> <p>3. El Web Máster llena los campos solicitados.</p> <p>4. El Web Máster presiona el botón guardar.</p>	<p>2. El sitio web muestra un formulario que contiene los siguientes campos:</p> <ul style="list-style-type: none"> a. Título. b. Descripción. c. Tipo. d. Estado. e. URL. f. Categoría padre. <p>5. El sistema va a base de datos y guarda el registro. Automáticamente se genera un código de categoría.</p> <p>6. El sistema muestra un mensaje de transacción exitosa.</p> <p>7. El sistema muestra la pantalla de mantenimiento de categorías.</p>
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de categorías.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	

Comentarios	Sitemap = Categoría. Cuando se cree una nueva categoría permitir si el usuario desea, que sea accesible para todos.
Acción	Entrada

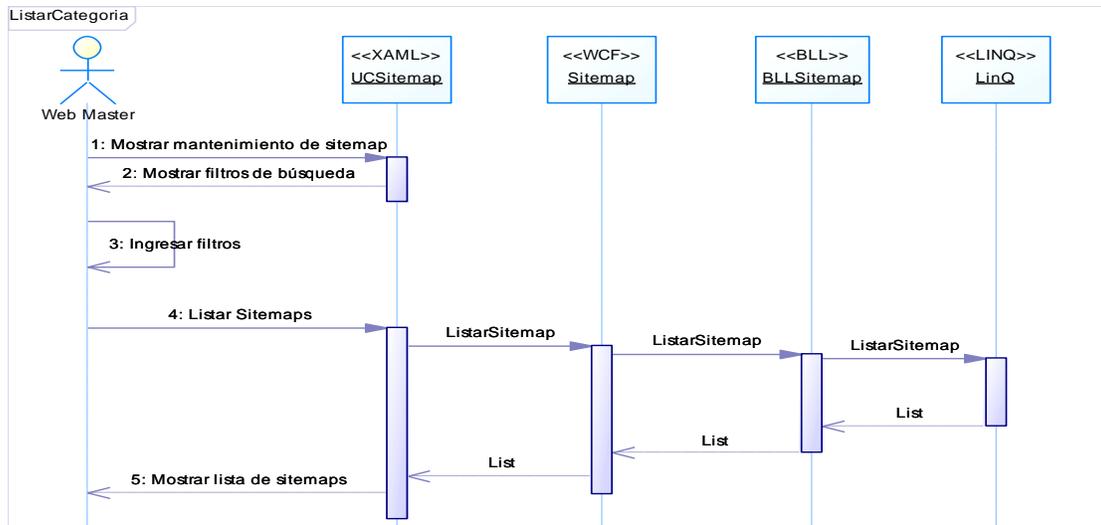


CU-004	Consultar categoría	
Requerimientos	RQF-004	
Casos de Uso Relacionados	CU-006	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir hacer una consulta de categorías.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster. 2. El registro a consultar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El sistema muestra los datos consultados. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de categorías y decide consultar una de ellas. 3. El Web Máster selecciona un registro de la lista. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado. 4. El sistema muestra los datos específicos para el registro seleccionado: <ol style="list-style-type: none"> a. Título b. Descripción c. URL d. Tipo e. Estado f. Categoría padre.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de categorías.	
	Acción 4: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	



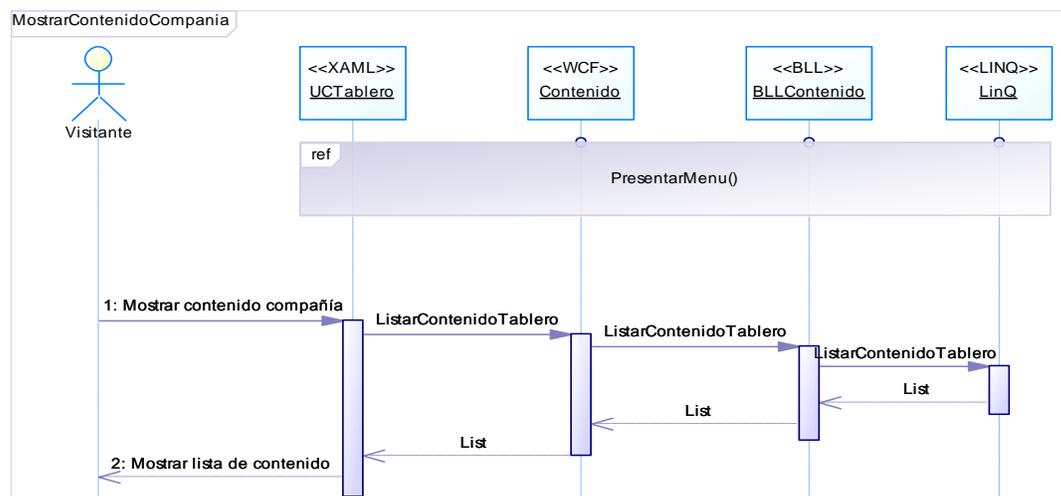
CU-005	Modificar categoría	
Requerimientos	RQF-005	
Casos de Uso Relacionados	CU-004	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir modificar una categoría.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster. 2. El registro a modificar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El registro queda actualizado en el sistema. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de categorías y decide modificar una de ellas. 3. El Web Máster selecciona un registro de la lista. 5. El Web Máster realiza los cambios necesarios sobre los campos que está habilitado a modificar: <ol style="list-style-type: none"> a. Título b. Descripción c. URL d. Tipo e. Estado f. Categoría padre 6. El Web Máster presiona el botón de guardar. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado de categorías. 4. El sistema muestra los datos específicos para el registro seleccionado. 7. El sistema va a base de datos y actualiza el registro. 8. El sistema muestra un mensaje de transacción exitosa. 9. El sistema muestra la pantalla de mantenimiento de categorías.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna	

	operación y muestra la pantalla de mantenimiento de categorías.
	Acción 5: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.
	Acción 7: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
Importancia	Alta
Comentarios	NA
Acción	Entrada



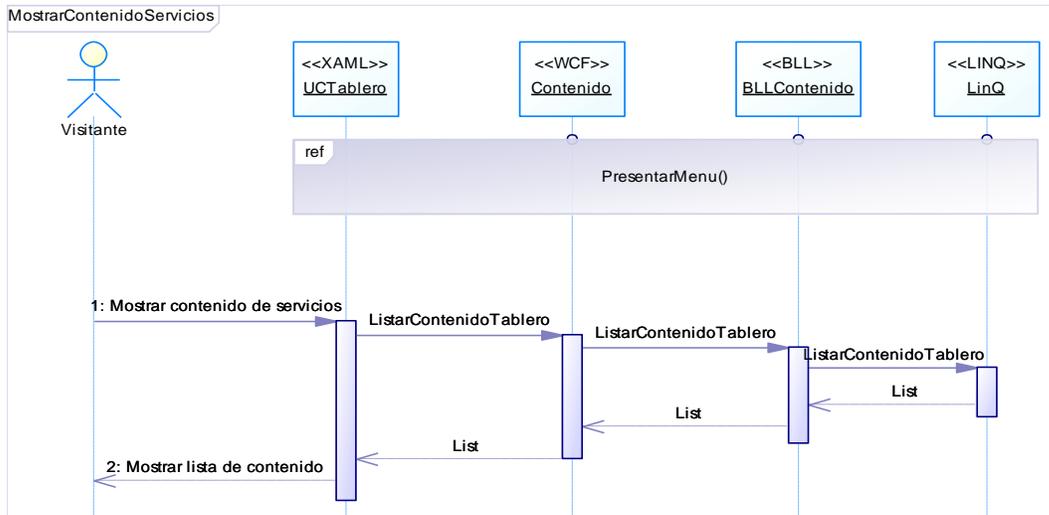
CU-006	Listar categorías	
Requerimientos	RQF-006	
Casos de Uso Relacionados	CU-040	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir ver el listado de categorías.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El sistema muestra un listado de los datos solicitados, basándose en los criterios dados por el Web Máster.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de categorías y decide ver un listado de las mismas. 2. El Web Máster ingresa los datos solicitados. 3. El Web Máster presiona el botón de buscar.	2. El sistema muestra una pantalla con los siguientes criterios: <ul style="list-style-type: none"> a. Título. b. Descripción. c. URL d. Estado e. Categoría padre. 3. El sistema muestra un listado con los siguientes datos: <ul style="list-style-type: none"> a. Título. b. Descripción. c. URL

		d. Estado e. Categoría padre.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de categorías.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	



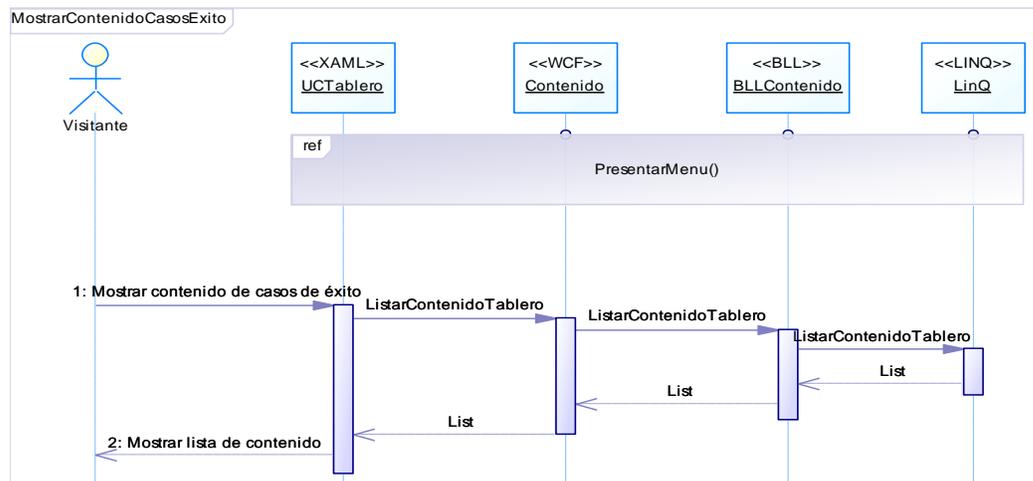
CU-007	Mostrar contenido de compañía.	
Requerimientos	RQF-007	
Casos de Uso Relacionados	CU-001, CU-002	
Actores	ACT-001, ACT-002, ACT-003	
Tipo	Primario	
Descripción	El sistema debe permitir ver el contenido de compañía.	
Precondiciones	1. El usuario debe tener acceso al contenido en el menú principal.	
Postcondiciones	1. El sistema muestra el contenido de compañía	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el usuario accede al menú principal y decide ver el contenido de compañía.</p> <p>3. El usuario selecciona si desea listar los últimos contenidos o los contenidos más leídos.</p>	<p>2. El sistema muestra el contenido de compañía, este apartado contendrá información sobre las oficinas corporativas de Babel Software en Costa Rica, Ecuador y Estados Unidos, además de un mensaje de bienvenida por parte del presidente junto con su foto, una reseña histórica de la empresa, misión, visión, valores y enfoque de negocios. El sistema muestra las opciones de los últimos contenidos o los contenidos más leídos para filtrarlos.</p>

		4. El sistema consulta la base de datos y muestra los contenidos que estén dentro de los filtros de búsqueda seleccionados.
	Curso Alternativo:	
	Acción 2: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	



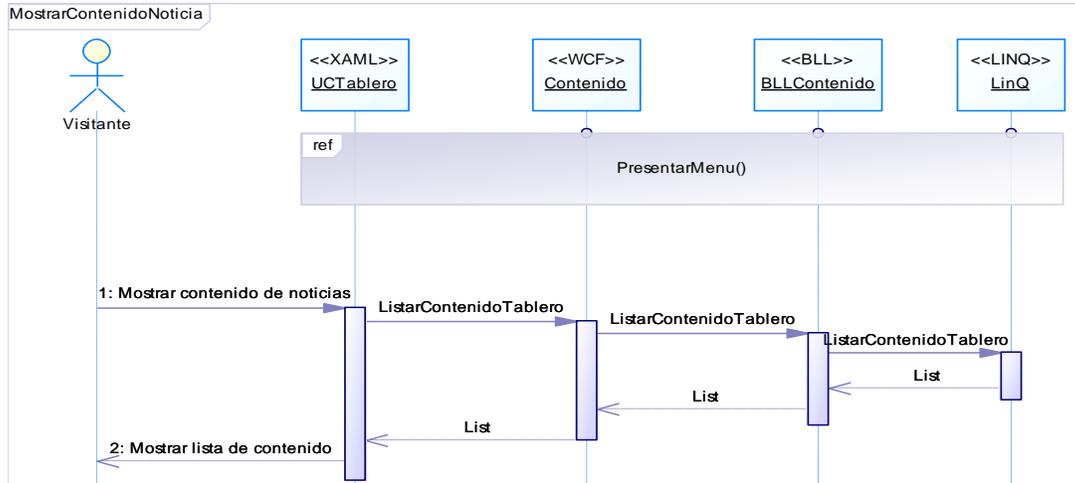
CU-008	Mostrar contenido de servicios	
Requerimientos	RQF-008	
Casos de Uso Relacionados	CU-001, CU-002	
Actores	ACT-001, ACT-002, ACT-003	
Tipo	Primario	
Descripción	El sistema debe permitir ver el contenido de servicios.	
Precondiciones	1. El debe tener acceso al contenido en el menú principal.	
Postcondiciones	1. El sistema muestra el contenido de servicios	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el usuario accede al menú principal y decide ver el contenido de servicios.</p> <p>3. El usuario selecciona si desea listar los últimos contenidos o los contenidos más leídos.</p>	<p>2. El sistema muestra el contenido de servicios, este apartado contendrá información sobre los servicios que ofrece Babel Software en el ámbito informático: Desarrollo a la media, Outsourcing, inteligencia de negocios, Software Factory, y Soporte de Aplicaciones. El sistema muestra las opciones de los últimos contenidos o los contenidos más leídos para filtrarlos.</p> <p>4. El sistema consulta la base de datos y muestra los contenidos que estén dentro</p>

	de los filtros de búsqueda seleccionados.
	Curso Alternativo:
	Acción 2: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
Importancia	Alta
Comentarios	NA
Acción	Salida

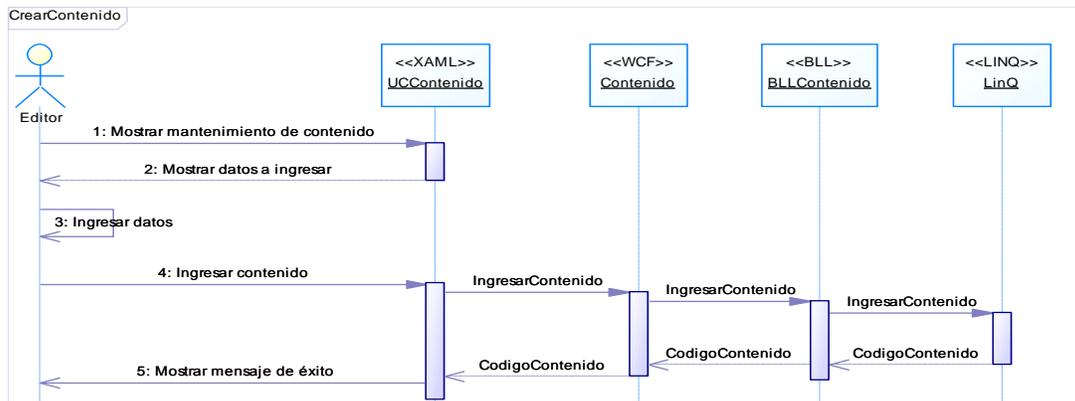


CU-009	Mostrar contenido de casos de éxito.	
Requerimientos	RQF-009	
Casos de Uso Relacionados	CU-001, CU-002	
Actores	ACT-001, ACT-002, ACT-003	
Tipo	Primario	
Descripción	El sistema debe permitir ver el contenido de casos de éxito.	
Precondiciones	1. El debe tener acceso al contenido en el menú principal.	
Postcondiciones	1. El sistema muestra el contenido de casos de éxito.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando el usuario accede al menú principal y decide ver los casos de éxito. 3. El usuario ingresa valores de fecha de inicio y fecha final o si desea listar los contenidos más leídos.	2. El sistema muestra el contenido de casos de éxito, este apartado contendrá testimonios de los clientes de Babel Software a fin de incrementar la confianza y credibilidad de los potenciales clientes. El sistema muestra las opciones de fecha de inicio y fecha final para filtrar los casos de éxito. Y las opciones de los últimos contenidos o los contenidos más leídos para filtrarlos. 4. El sistema consulta la base de datos y muestra los casos de éxito que están dentro de los rangos.
	Curso Alternativo:	
	Acción 2: Se presenta un fallo de conexión. El sistema muestra un mensaje	

	de retroalimentación.
Importancia	Alta
Comentarios	NA
Acción	Salida

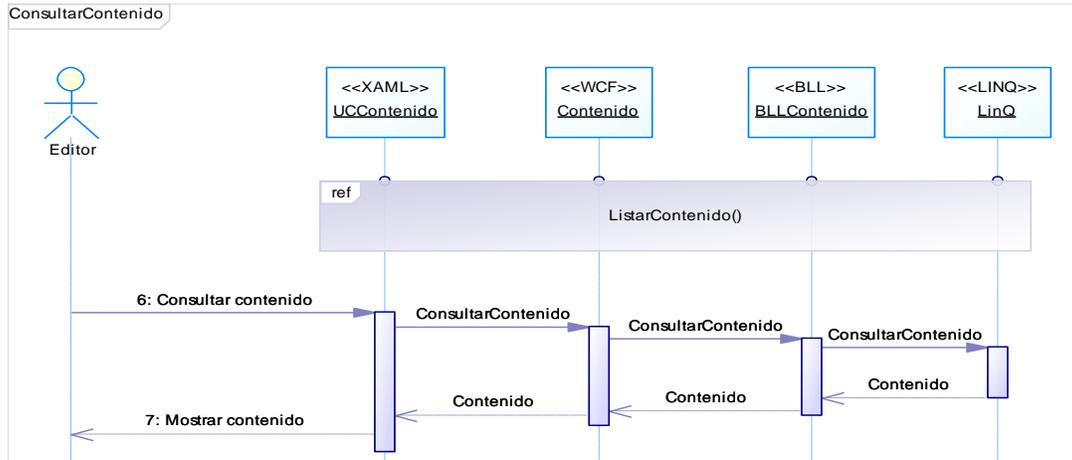


CU-010	Mostrar contenido de noticias.	
Requerimientos	RQF-010	
Casos de Uso Relacionados	CU-001, CU-002	
Actores	ACT-001, ACT-002, ACT-003	
Tipo	Primario	
Descripción	El sistema debe permitir ver el contenido de noticias.	
Precondiciones	1. El debe tener acceso al contenido en el menú principal.	
Postcondiciones	1. El sistema muestra el contenido de noticias.	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el usuario accede al menú principal y decide ver las noticias.</p> <p>3. El usuario ingresa valores de fecha de inicio y fecha final o si desea listar los contenidos más leídos.</p>	<p>2. El sistema muestra el contenido de noticias, este apartado contendrá reportajes alusivos al área informática, las noticias estarán ordenadas cronológicamente. El sistema muestra las opciones de fecha de inicio y fecha final para filtrar las noticias. Y las opciones de los últimos contenidos o los contenidos más leídos para filtrarlos.</p> <p>4. El sistema consulta la base de datos y muestra las noticias que están dentro de los rangos.</p>
	Curso Alternativo:	
	Acción 2, 4: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
	Acción 3: El usuario ingresa valores erróneos. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	



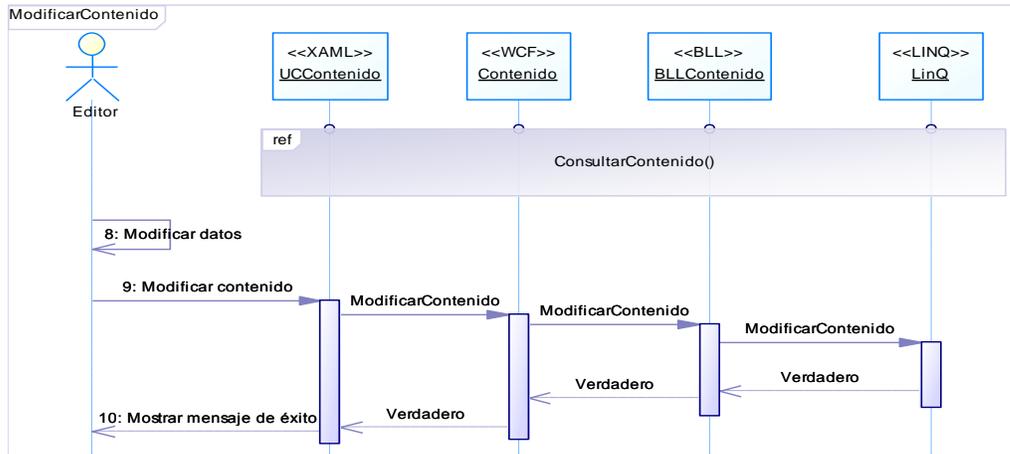
CU-011	Crear contenido	
Requerimientos	RQF-011	
Casos de Uso Relacionados	CU-040	
Actores	ACT-001, ACT-003	
Tipo	Primario	
Descripción	El sitio web debe permitir el ingreso de nuevos contenidos.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster o Editor.	
Postcondiciones	1. El contenido queda registrado en el sitio web.	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster o el Editor accede a la pantalla de mantenimiento de contenidos y decide ingresar uno nuevo. 3. El Web Máster llena los campos solicitados. 4. El Web Máster presiona el botón guardar. 	<ol style="list-style-type: none"> 2. El sitio web muestra un formulario que contiene los siguientes campos: <ol style="list-style-type: none"> a. Título. b. Resumen. d. Estado. e. Categoría y Sub categoría si la hay. 5. El sistema va a base de datos y guarda el registro. Automáticamente se genera un código de contenido. 6. El sistema muestra un mensaje de transacción exitosa. 7. El sistema muestra la pantalla de mantenimiento de contenidos.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de contenido.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	

Comentarios	Un contenido que tenga el estado de portada aparecerá dinámicamente en la página de inicio. Enviar un correo al administrador del sistema, indicando que se creó un nuevo contenido con el Título, Categoría y el Usuario que creó el contenido.
Acción	Entrada



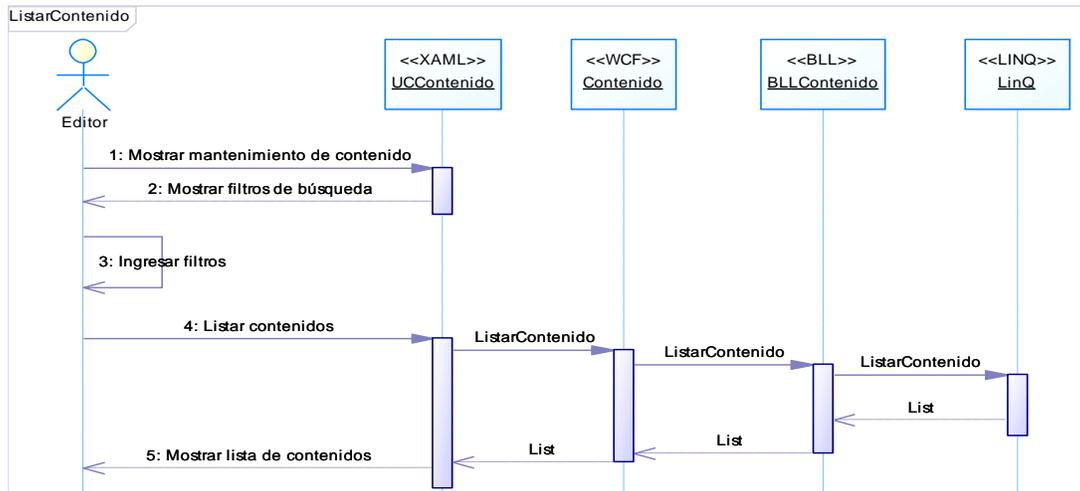
CU-012	Consultar contenido					
Requerimientos	RQF-012					
Casos de Uso Relacionados	CU-014					
Actores	ACT-001, ACT-003					
Tipo	Secundario					
Descripción	El sistema debe permitir hacer una consulta de contenido.					
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster o Editor. 2. El registro a consultar debe estar previamente ingresado en el sistema. 					
Postcondiciones	<ol style="list-style-type: none"> 1. El sistema muestra los datos consultados. 					
Curso Típico de Eventos	<table border="1"> <thead> <tr> <th>Actor</th> <th>Sistema</th> </tr> </thead> <tbody> <tr> <td> <ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster o el Editor accede a la pantalla de mantenimiento de contenido y decide consultar uno de ellos. 3. El Web Máster o el Editor selecciona un registro de la lista. </td> <td> <ol style="list-style-type: none"> 2. El sistema muestra un listado. 4. El sistema muestra los datos específicos para el registro seleccionado: <ol style="list-style-type: none"> a. Título b. Resumen c. Estado d. Categoría y Sub categoría si la hay. </td> </tr> </tbody> </table>	Actor	Sistema	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster o el Editor accede a la pantalla de mantenimiento de contenido y decide consultar uno de ellos. 3. El Web Máster o el Editor selecciona un registro de la lista. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado. 4. El sistema muestra los datos específicos para el registro seleccionado: <ol style="list-style-type: none"> a. Título b. Resumen c. Estado d. Categoría y Sub categoría si la hay. 	
Actor	Sistema					
<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster o el Editor accede a la pantalla de mantenimiento de contenido y decide consultar uno de ellos. 3. El Web Máster o el Editor selecciona un registro de la lista. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado. 4. El sistema muestra los datos específicos para el registro seleccionado: <ol style="list-style-type: none"> a. Título b. Resumen c. Estado d. Categoría y Sub categoría si la hay. 					
	Curso Alternativo:					
	Acción 3: El Web Máster o el Editor cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de contenidos.					
	Acción 4: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.					
Importancia	Alta					
Comentarios	Cuando se consulte un contenido desde las secciones de mostrar contenidos, se sumará en 1 el campo DesplegadoContenido, para registrar cuantas veces					

	ha sido leído.
Acción	Salida



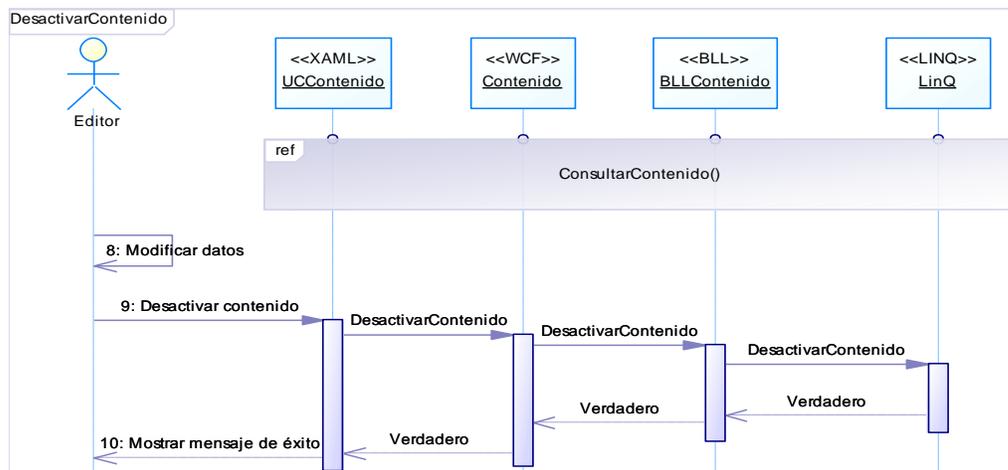
CU-013	Modificar contenido	
Requerimientos	RQF-013	
Casos de Uso Relacionados	CU-012	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir modificar un contenido.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster o Editor. 2. El registro a modificar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El registro queda actualizado en el sistema. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de contenidos y decide modificar uno de ellos. 3. El Web Máster o el Editor selecciona un registro de la lista. 5. El Web Máster realiza los cambios necesarios sobre los campos que está habilitado a modificar: <ul style="list-style-type: none"> a. Título b. Resumen c. Estado d. Categoría y Sub categoría si la hay. 6. El Web Máster o el Editor presiona el botón de guardar. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado de contenidos. 4. El sistema muestra los datos específicos para el registro seleccionado. 7. El sistema va a base de datos y actualiza el registro. 8. El sistema muestra un mensaje de transacción exitosa. 9. El sistema muestra la pantalla de mantenimiento de contenidos.
	Curso Alternativo:	
	Acción 3: El Web Máster o el Editor cancela la operación. El sistema no realiza	

	ninguna operación y muestra la pantalla de mantenimiento de contenidos.
	Acción 5: El Web Máster o el Editor ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.
	Acción 7: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
Importancia	Alta
Comentarios	NA
Acción	Entrada



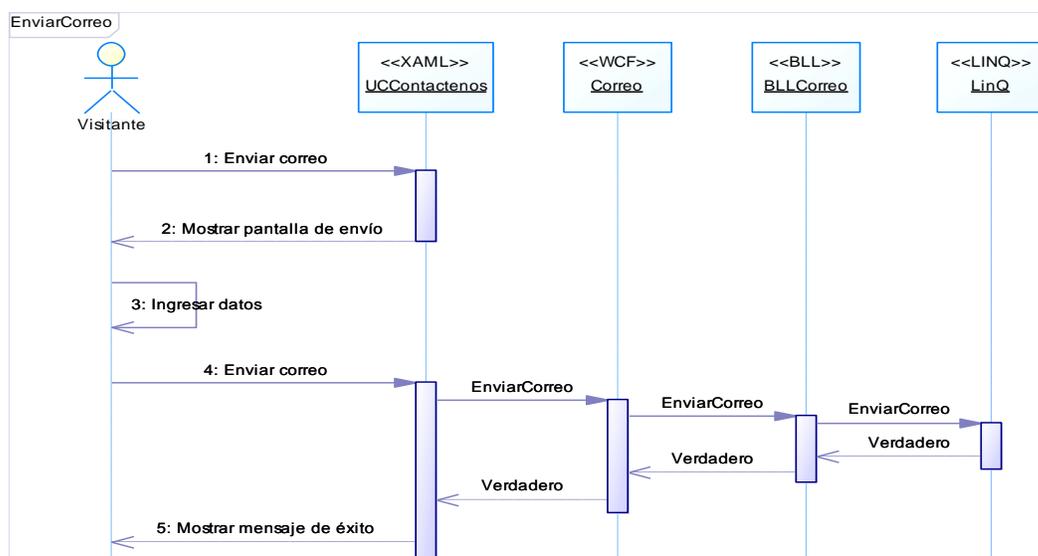
CU-014	Listar contenidos	
Requerimientos	RQF-014	
Casos de Uso Relacionados	CU-040	
Actores	ACT-001, ACT-002, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir ver el listado de contenidos.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster o Editor.	
Postcondiciones	1. El sistema muestra un listado de los datos solicitados, basándose en los criterios dados por el Web Máster o el Editor.	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster o el Editor accede a la pantalla de mantenimiento de contenidos y decide ver un listado. 3. El Web Máster ingresa los datos solicitados. 4. El Web Máster o el Editor presiona el botón de buscar. 	<ol style="list-style-type: none"> 2. El sistema muestra una pantalla con los siguientes criterios: <ol style="list-style-type: none"> a. Título. b. Resumen. c. Estado. d. Categoría. 5. El sistema muestra un listado con los siguientes datos: <ol style="list-style-type: none"> a. Título. b. Resumen. c. Estado d. Categoría y Sub categoría si la hay.

	Curso Alternativo:
	Acción 3: El Web Máster o el Editor cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de contenidos.
	Acción 3: El Web Máster o el Editor ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
Importancia	Alta
Comentarios	Un contenido que tenga el estado de portada aparecerá dinámicamente en la página de inicio. Existe un segundo listar contenido que es dinámico, en el cual se muestran los contenidos de cada categoría y estarán ordenadas cronológicamente o por más leídos. Todos los mostrar contenido lo harán por fecha o por orden de más leídos, con un máximo de 20 contenidos.
Acción	Salida



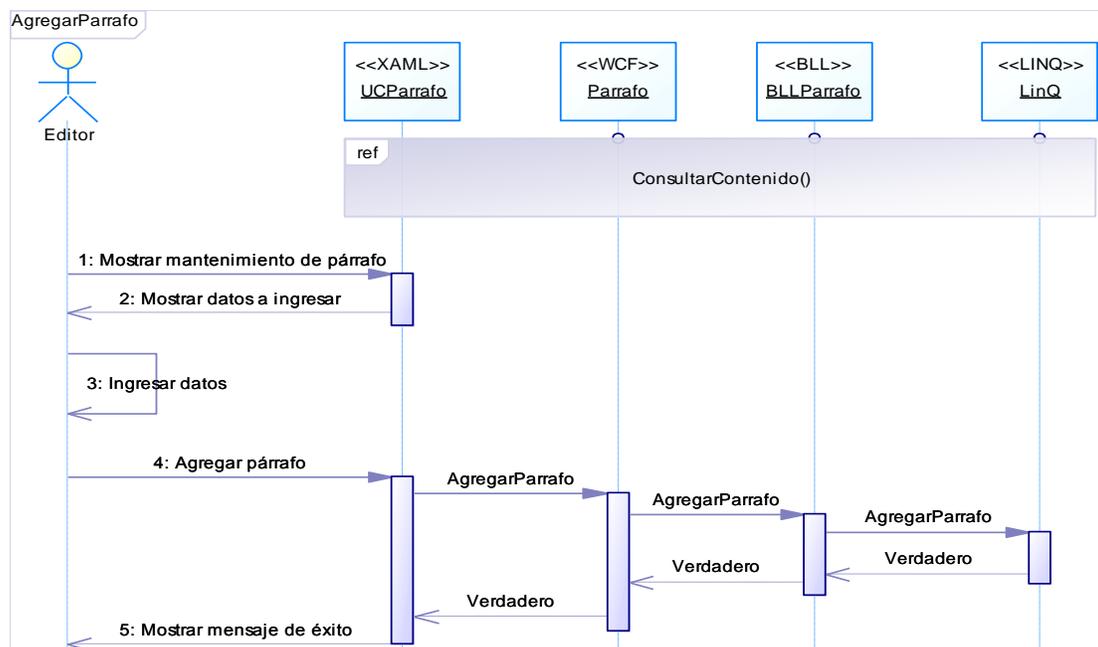
CU-015	Desactivar contenido	
Requerimientos	RQF-015	
Casos de Uso Relacionados	CU-012	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir desactivar un contenido.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster o Editor. 2. El registro a desactivar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El registro queda actualizado en el sistema. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster o el Editor accede a la pantalla de mantenimiento de contenidos y decide modificar uno de ellos. 3. El Web Máster o el Editor selecciona un registro de la lista. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado de contenidos. 4. El sistema muestra los datos específicos para el registro seleccionado. 7. El sistema va a base de datos y actualiza el registro. 8. El sistema muestra un mensaje de

	<p>5. El Web Máster o el Editor modifica el estado del contenido.</p> <p>6. El Web Máster o el Editor presiona el botón de guardar.</p>	<p>transacción exitosa.</p> <p>9. El sistema muestra la pantalla de mantenimiento de contenidos.</p>
	Curso Alternativo:	
	Acción 3: El Web Máster o el Editor cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de categorías.	
	Acción 7: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Entrada	



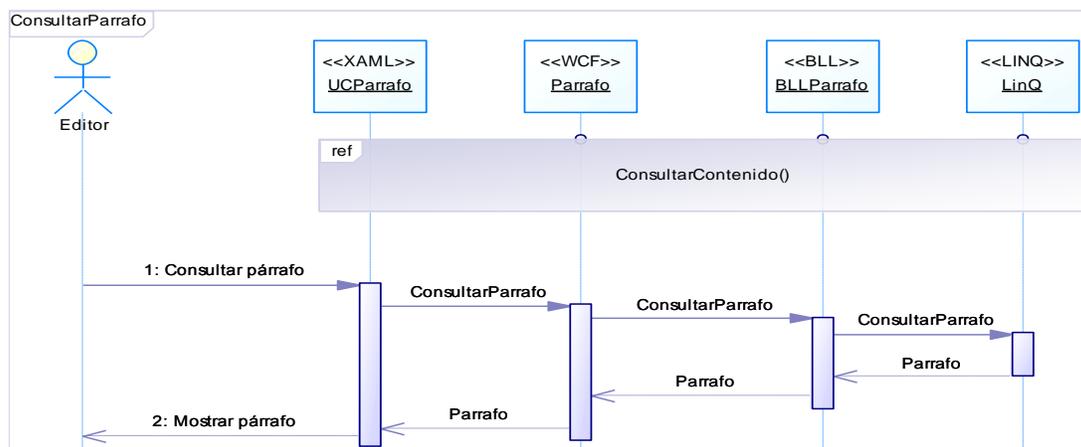
CU-016	Enviar correo (Contáctenos)	
Requerimientos	RQF-016	
Casos de Uso Relacionados		
Actores	ACT-002	
Tipo	Secundario	
Descripción	El sistema debe permitir desactivar un contenido.	
Precondiciones	1. El usuario debe tener acceso al contenido en el menú principal.	
Postcondiciones	1. El correo es enviado hacia el Web Máster.	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el usuario accede al menú principal y decide contactar a la empresa</p> <p>3. El usuario llena los campos solicitados.</p> <p>4. El usuario presiona el botón Enviar.</p>	<p>2. El sistema muestra un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> a. Nombre. b. correo electrónico. c. Teléfono o Fax d. País e. cómo desea ser contactado (Teléfono, Correo)

		f. Asunto. g. Comentario. 5. El sistema envía el mail, guarda los datos y muestra un mensaje de transacción exitosa.
	Curso Alternativo:	
	Acción 3: El usuario cancela la operación. El sistema no realiza ninguna operación y muestra la página Home.	
	Acción 7: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Entrada	



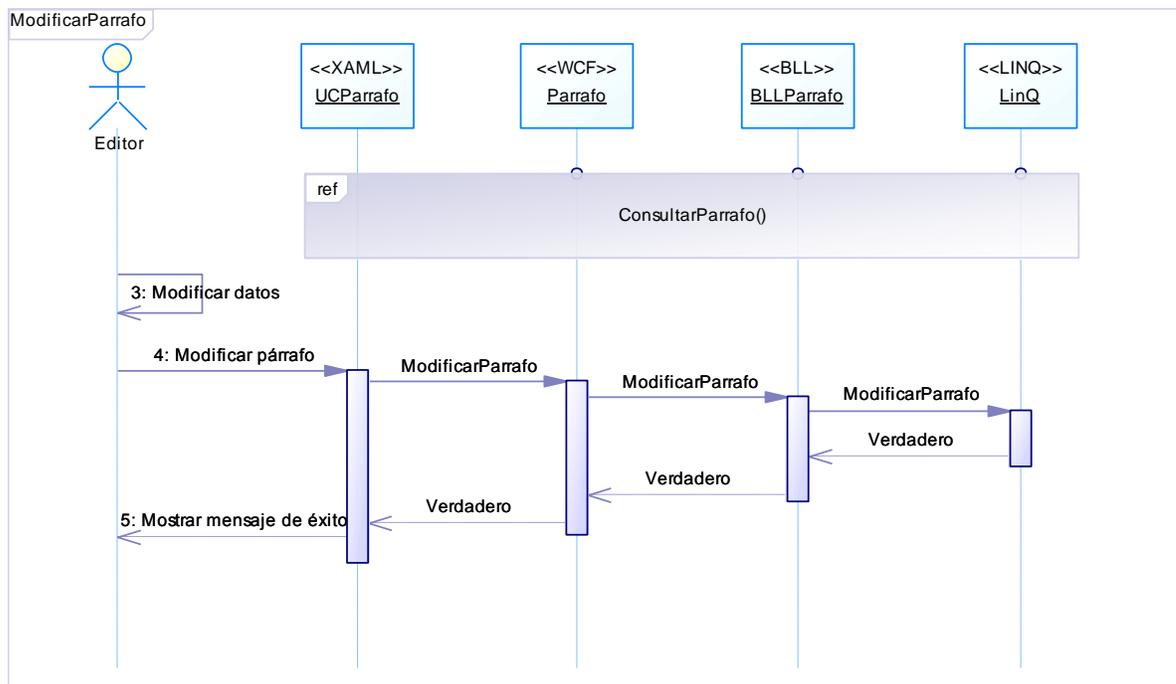
CU-017	Agregar párrafo	
Requerimientos	RQF-017	
Casos de Uso Relacionados	CU-011	
Actores	ACT-001, ACT-003	
Tipo	Primario	
Descripción	El sitio web debe permitir el ingreso de párrafos asociados a un contenido.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Editor.	
Postcondiciones	1. El párrafo queda registrado en el sitio web.	
Curso Típico de	Actor	Sistema

Eventos	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Editor accede a la pantalla de mantenimiento de contenidos y decide ingresar un nuevo párrafo. 3. El Editor llena el texto del párrafo. 4. El Editor presiona el botón guardar. 	<ol style="list-style-type: none"> 2. El sitio web muestra un formulario que contiene el campo: <ol style="list-style-type: none"> a. Texto. 5. El sistema va a base de datos y guarda el registro. Automáticamente se genera un código de párrafo. 6. El sistema muestra un mensaje de transacción exitosa. 7. El sistema muestra la pantalla de ingreso de párrafo, en modo de inserción de datos.
	Curso Alternativo:	
	Acción 3: El Editor cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de ingreso de párrafo.	
	Acción 3: El Editor ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	Recordar que un contenido posee uno o varios párrafos.	
Acción	Entrada	



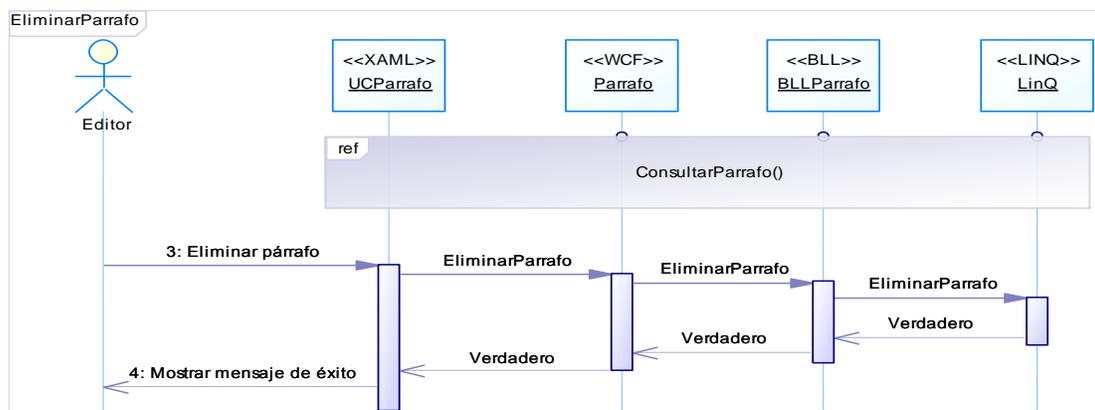
CU-018	Consultar párrafo	
Requerimientos	RQF-018	
Casos de Uso Relacionados	CU-017	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir hacer una consulta de párrafo.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Editor. 2. El registro a consultar debe estar previamente ingresado en el sistema. 	
Postcondiciones	1. El sistema muestra los datos consultados.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando el Editor accede a la pantalla de	2. El sistema muestra un primer párrafo asociado al contenido.

	mantenimiento de párrafos.
	Curso Alternativo:
	Acción 1: El Editor cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de contenidos.
	Acción 2: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
	Acción 2: El editor presiona el botón siguiente. El sistema muestra el siguiente párrafo asociado al contenido.
	Acción 2: El editor presiona el botón anterior. El sistema muestra el párrafo anterior asociado al contenido.
Importancia	Alta
Comentarios	NA
Acción	Salida



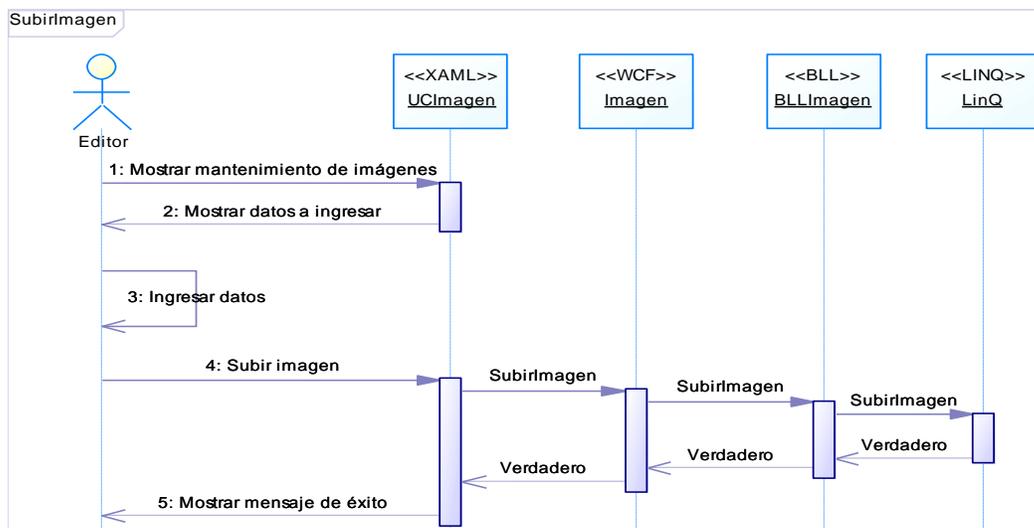
CU-019	Modificar párrafo	
Requerimientos	RQF-019	
Casos de Uso Relacionados	CU-018	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir modificar un contenido.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Editor. 2. El registro a modificar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El registro queda actualizado en el sistema. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Editor accede a la pantalla de 	<ol style="list-style-type: none"> 2. El sistema consulta y muestra el párrafo a modificar.

	<p>mantenimiento de párrafos y decide modificar uno de ellos.</p> <p>3. El Editor realiza los cambios necesarios sobre el texto del párrafo.</p> <p>4. El Editor presiona el botón de guardar.</p>	<p>5. El sistema va a base de datos y actualiza el registro.</p> <p>6. El sistema muestra un mensaje de transacción exitosa.</p> <p>7. El sistema muestra la pantalla de ingreso de párrafo, en modo de inserción de datos.</p>
	Curso Alternativo:	
	Acción 1: El Editor cancela la operación. El sistema no realiza ninguna operación.	
	Acción 3: El Editor ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	Recordar que un contenido posee uno o varios párrafos.	
Acción	Entrada	



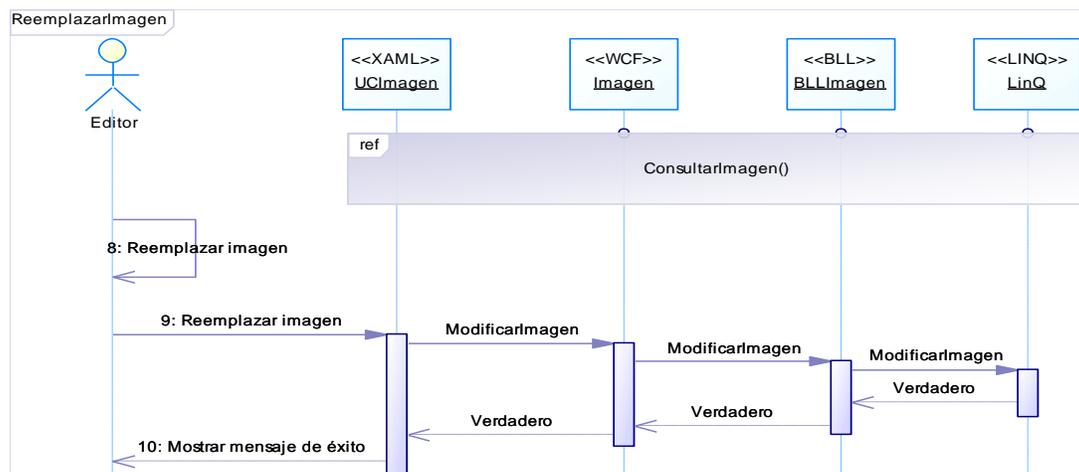
CU-020	Eliminar párrafo	
Requerimientos	RQF-020	
Casos de Uso Relacionados	CU-018	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir eliminar un párrafo.	
Precondiciones	<p>1. El usuario debe estar debidamente identificado en el sitio web con el rol de Editor.</p> <p>2. El registro a eliminar debe estar previamente ingresado en el sistema.</p>	
Postcondiciones	1. El registro queda actualizado en el sistema.	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el Editor accede a la pantalla de mantenimiento de párrafos y decide eliminar uno de ellos.</p> <p>3. El Editor presiona el botón de eliminar.</p>	<p>2. El sistema consulta y muestra el párrafo a eliminar.</p> <p>4. El sistema muestra un mensaje para confirmación, va a base de datos y actualiza el registro.</p> <p>5. El sistema muestra un mensaje de</p>

		transacción exitosa. 6. El sistema muestra la pantalla de ingreso de párrafo, en modo de inserción de datos.
	Curso Alternativo:	
	Acción 1: El Editor cancela la operación. El sistema no realiza ninguna operación.	
	Acción 3: El Editor cancela la eliminación del párrafo. El sistema muestra el párrafo.	
	Acción 4: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Entrada	



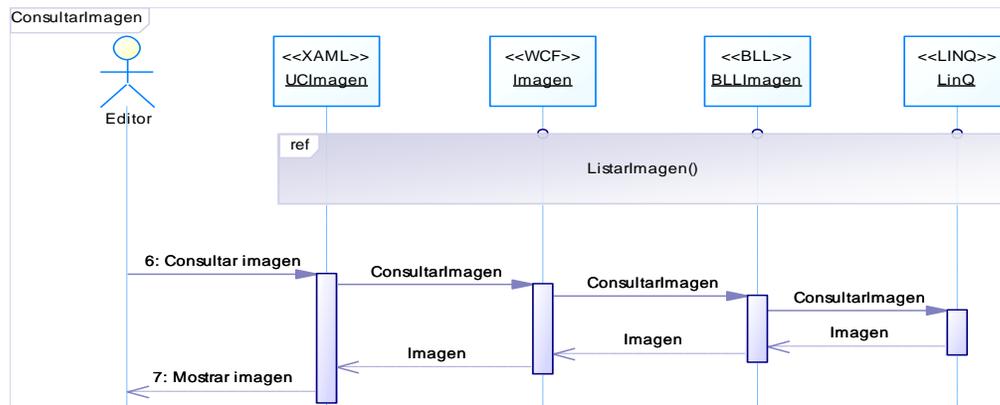
CU-021	Subir imagen	
Requerimientos	RQF-021	
Casos de Uso Relacionados	CU-021	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sitio web debe permitir reemplazar una imagen.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Editor o Web Máster.	
Postcondiciones	1. La imagen queda reemplazada en el sitio web.	
Curso Típico de	Actor	Sistema

Eventos	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Editor o el Web Máster accede a la pantalla de mantenimiento de imágenes y decide reemplazar una imagen. 3. El Editor o el Web Máster llena el título y carga la imagen. 4. El Editor o el Web Máster presiona el botón guardar. 	<ol style="list-style-type: none"> 2. El sitio web muestra un formulario que contiene el campo: Título y la opción de carga de la imagen. 5. El sistema va a base de datos y guarda el registro, además de cargar la imagen en el servidor. 6. El sistema muestra un mensaje de transacción exitosa. 7. El sistema muestra la pantalla de mantenimiento de imágenes.
	Curso Alternativo:	
	Acción 3: El Editor o el Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de ingreso de párrafo.	
	Acción 3: El Editor o el Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo en la carga al servidor. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	N/A	
Acción	Entrada	



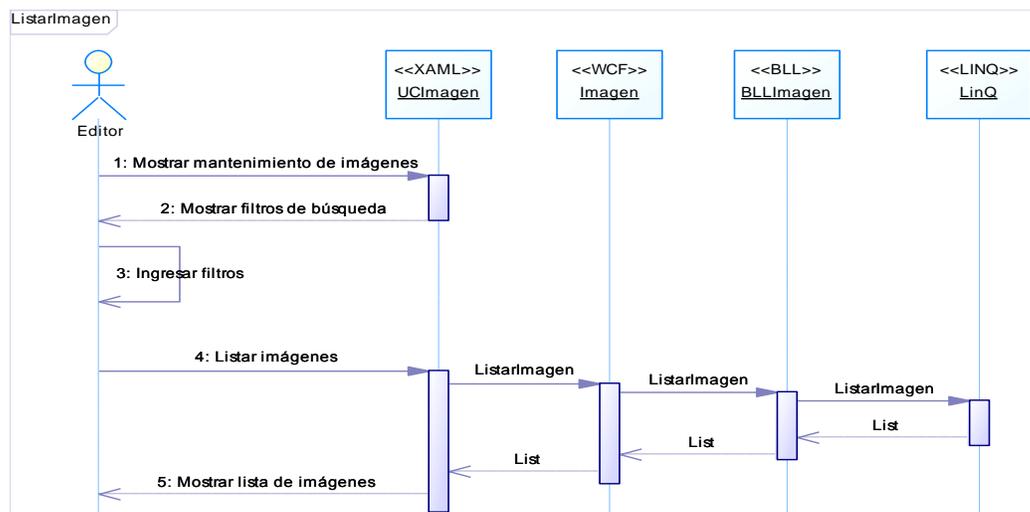
CU-022	Reemplazar imagen
Requerimientos	RQF-019
Casos de Uso Relacionados	CU-018
Actores	ACT-001, ACT-003
Tipo	Secundario
Descripción	El sistema debe permitir reemplazar una imagen.
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Editor o Web Máster. 2. La imagen a reemplazar debe estar previamente ingresado en el sistema.
Postcondiciones	<ol style="list-style-type: none"> 1. La imagen queda actualizada en el sistema.

Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Editor accede a la pantalla de mantenimiento de párrafos y decide modificar uno de ellos. 3. El Editor o el Web Máster realiza el reemplazo de la imagen. 4. El Editor o el Web Máster presiona el botón de guardar. 	<ol style="list-style-type: none"> 2. El sistema consulta y muestra la imagen a reemplazar. 5. El sistema va a base de datos y actualiza el registro, además carga la imagen. 6. El sistema muestra un mensaje de transacción exitosa. 7. El sistema muestra la pantalla de mantenimiento de imágenes.
	Curso Alternativo:	
	Acción 1: El Editor o el Web Máster cancela la operación. El sistema no realiza ninguna operación.	
	Acción 3: El Editor o el Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo en la carga al servidor. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Entrada	



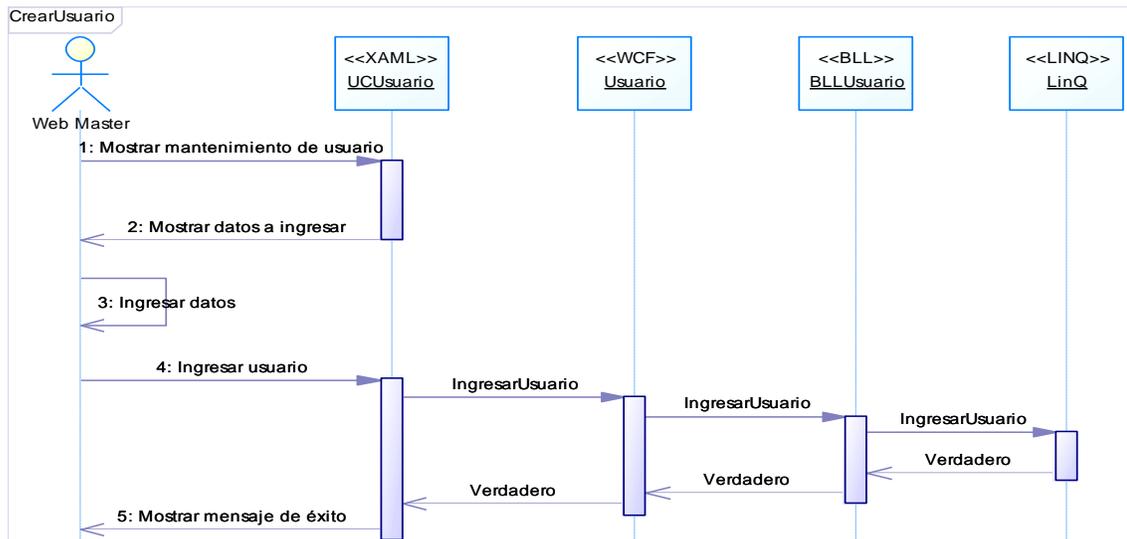
CU-023	Consultar imagen	
Requerimientos	RQF-023	
Casos de Uso Relacionados	CU-024	
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir hacer una consulta de imagen.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Editor o Web Máster. 2. La imagen a consultar debe estar previamente ingresada en el sistema. 	
Postcondiciones	1. El sistema muestra la imagen consultada.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando	2. El sistema muestra un listado.

	<p>el Web Máster o el Editor accede a la pantalla de mantenimiento de imágenes y decide consultar una de ellas.</p> <p>3. El Web Máster o el Editor selecciona un registro de la lista.</p>	<p>4. El sistema muestra los datos específicos para el registro seleccionado:</p> <p>a. Título</p> <p>b. Imagen registrada.</p>
	<p>Curso Alternativo:</p> <p>Acción 1: El Editor o el Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de imágenes.</p> <p>Acción 2: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.</p>	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	



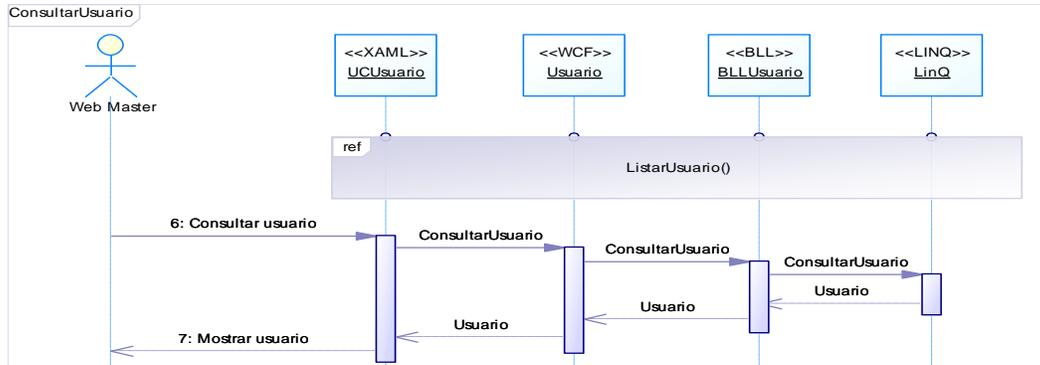
CU-024	Listar imágenes	
Requerimientos	RQF-024	
Casos de Uso Relacionados		
Actores	ACT-001, ACT-003	
Tipo	Secundario	
Descripción	El sistema debe permitir ver el listado de imágenes.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster o Editor.	
Postcondiciones	1. El sistema muestra un listado de los datos solicitados, basándose en los criterios dados por el Web Máster o el Editor.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando el Web Máster o el Editor accede a la pantalla de mantenimiento de imágenes.	2. El sistema muestra un listado con los siguientes datos: a. Título.

	Curso Alternativo:
	Acción 2: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
Importancia	Alta
Comentarios	NA
Acción	Salida

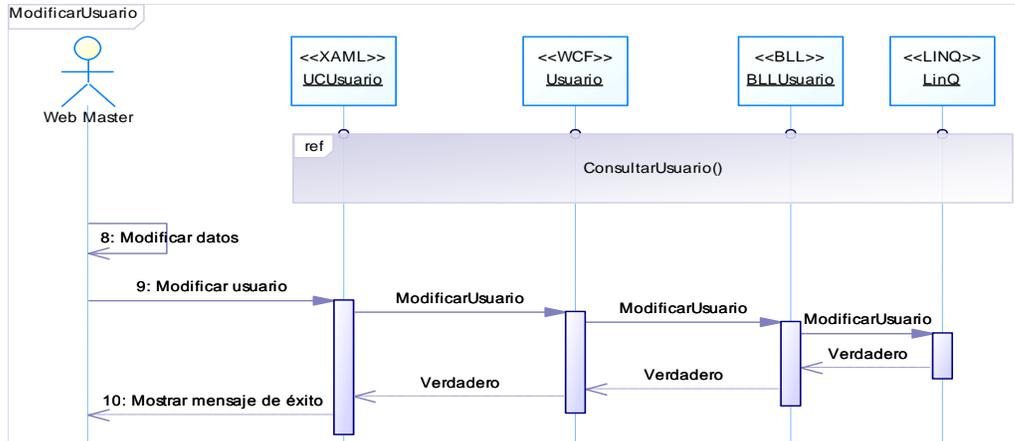


CU-025	Crear usuario	
Requerimientos	RQF-025	
Casos de Uso Relacionados		
Actores	ACT-001	
Tipo	Primario	
Descripción	El sitio web debe permitir el ingreso de un usuario.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El usuario queda registrado en el sitio web.	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de usuarios y decide ingresar uno nuevo.</p> <p>3. El Web Máster llena los campos solicitados.</p> <p>4. El Web Máster presiona el botón guardar.</p>	<p>2. El sitio web muestra un formulario que contiene los siguientes campos:</p> <ul style="list-style-type: none"> a. Nombres. b. Apellidos. c. Nombre de usuario. d. Contraseña. e. Estado <p>5. El sistema va a base de datos y guarda el registro.</p> <p>6. El sistema muestra un mensaje de transacción exitosa.</p> <p>7. El sistema muestra la pantalla de mantenimiento de usuarios.</p>
	Curso Alternativo:	

	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de usuarios.
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
Importancia	Alta
Comentarios	N/A
Acción	Entrada

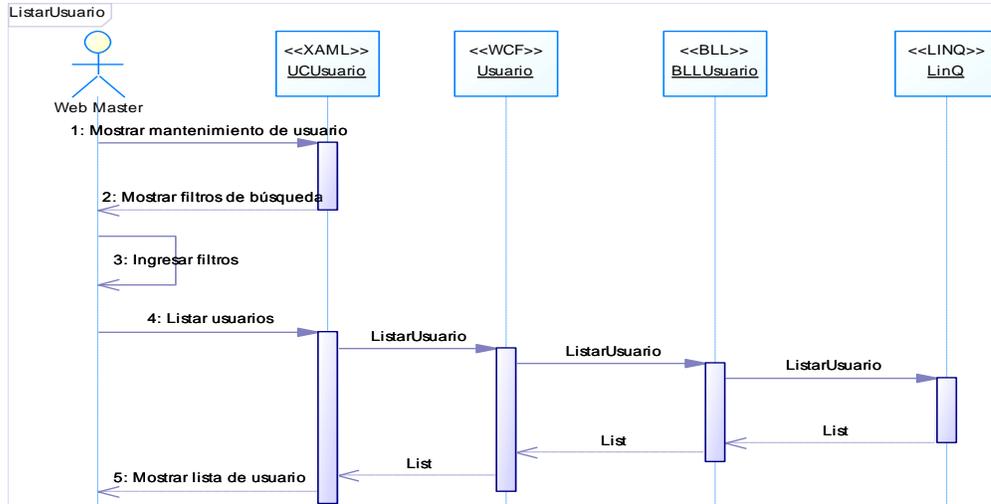


CU-026	Consultar usuario	
Requerimientos	RQF-026	
Casos de Uso Relacionados	CU-028	
Actores	ACT-001	
Tipo	Secundario	
Descripción	El sistema debe permitir hacer una consulta de usuario.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster. 2. El registro a consultar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El sistema muestra los datos consultados. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de usuarios y decide consultar uno de ellos. 3. El Web Máster selecciona un registro de la lista. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado. 4. El sistema muestra los datos específicos para el registro seleccionado: <ol style="list-style-type: none"> a. Nombres b. Apellidos c. Nombre de usuario d. Contraseña. e. Estado
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de usuarios.	
	Acción 4: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	



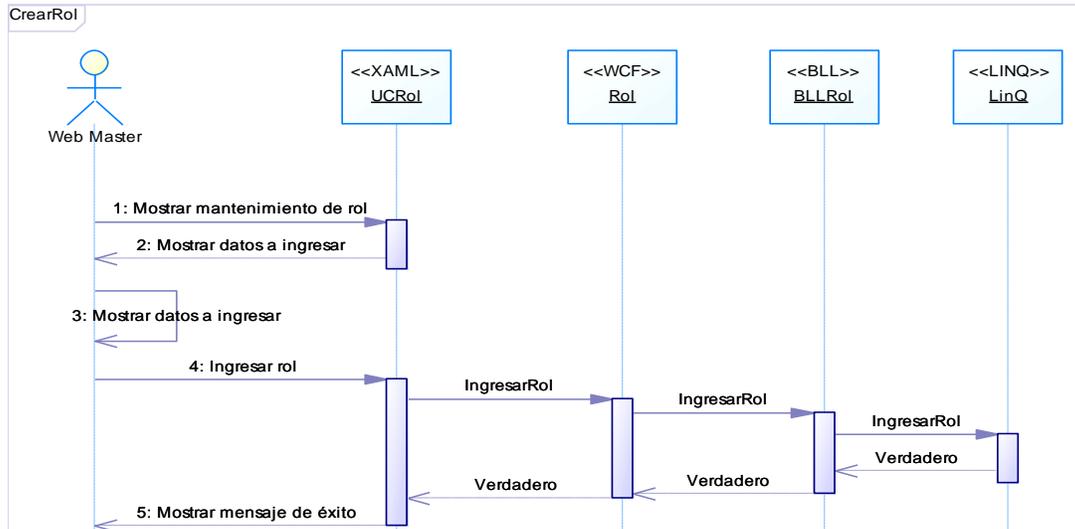
CU-027	Modificar usuario	
Requerimientos	RQF-027	
Casos de Uso Relacionados	CU-026	
Actores	ACT-001	
Tipo	Secundario	
Descripción	El sistema debe permitir modificar un usuario.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster. 2. El registro a modificar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El registro queda actualizado en el sistema. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de usuarios y decide modificar uno de ellos. 3. El Web Máster selecciona un registro de la lista. 5. El Web Máster realiza los cambios necesarios sobre los campos que está habilitado a modificar: <ul style="list-style-type: none"> a. Nombres b. Apellidos c. Contraseña d. Estado 6. El Web Máster presiona el botón de guardar. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado de usuarios. 4. El sistema muestra los datos específicos para el registro seleccionado. 7. El sistema va a base de datos y actualiza el registro. 8. El sistema muestra un mensaje de transacción exitosa. 9. El sistema muestra la pantalla de mantenimiento de usuarios.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de usuarios.	
	Acción 5: El Web Máster o el Editor ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 7: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	

Comentarios	El Nombre de usuario no se puede modificar, ya que este se lo utiliza como referencia.
Acción	Entrada



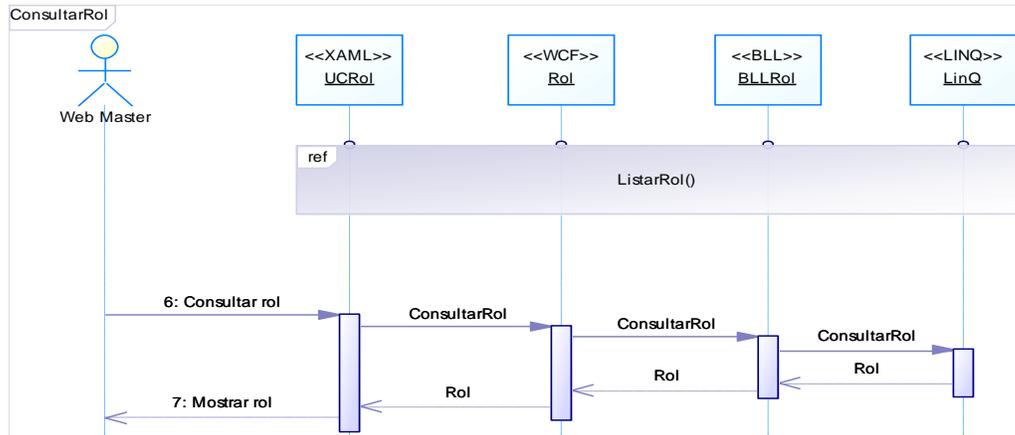
CU-028	Listar usuarios	
Requerimientos	RQF-028	
Casos de Uso Relacionados		
Actores	ACT-001	
Tipo	Secundario	
Descripción	El sistema debe permitir ver el listado de usuarios.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El sistema muestra un listado de los datos solicitados, basándose en los criterios dados por el Web Máster.	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de usuarios y decide ver un listado.</p> <p>3. El Web Máster ingresa los datos solicitados.</p> <p>4. El Web Máster presiona el botón de buscar.</p>	<p>2. El sistema muestra una pantalla con los siguientes criterios:</p> <ul style="list-style-type: none"> a. Nombre b. Apellido c. Nombre de usuario e. Estado <p>5. El sistema muestra un listado con los siguientes datos:</p> <ul style="list-style-type: none"> a. Nombre. b. Apellido. c. Nombre de usuario d. Contraseña e. Estado.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de usuarios.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	

	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
Importancia	Alta
Comentarios	NA
Acción	Salida

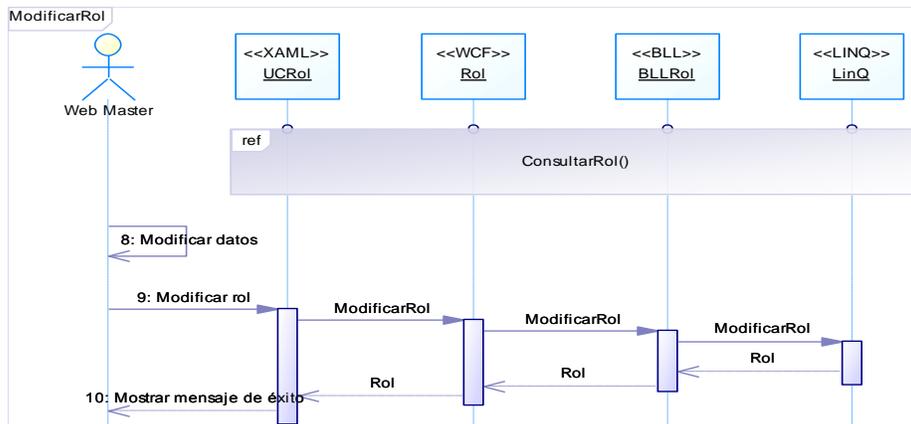


CU-029	Crear rol	
Requerimientos	RQF-029	
Casos de Uso Relacionados		
Actores	ACT-001	
Tipo	Primario	
Descripción	El sitio web debe permitir el ingreso de un rol.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El rol queda registrado en el sitio web.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de roles y decide ingresar uno nuevo.	2. El sitio web muestra un formulario que contiene los siguientes campos: a. Nombre b. Estado
	3. El Web Máster llena los campos solicitados.	5. El sistema va a base de datos y guarda el registro.
	4. El Web Máster presiona el botón guardar.	6. El sistema muestra un mensaje de transacción exitosa. 7. El sistema muestra la pantalla de mantenimiento de usuarios.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de roles.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	

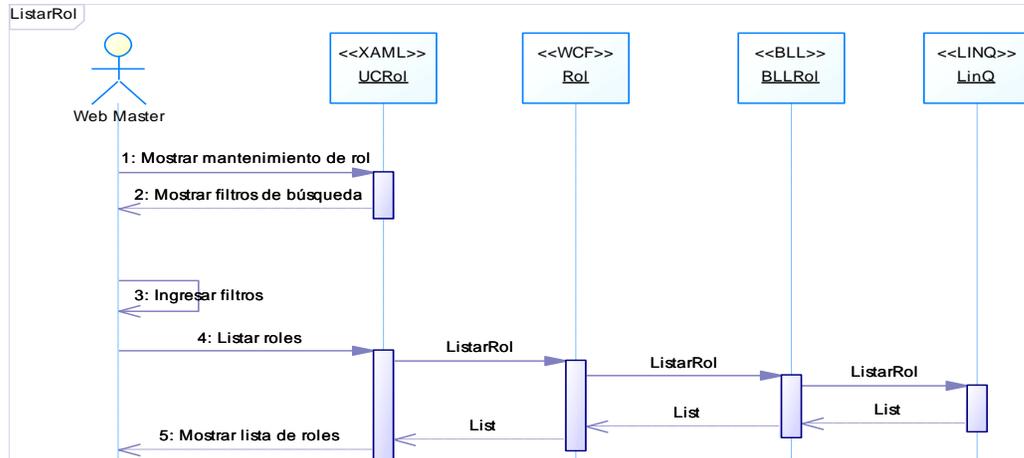
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.
Importancia	Alta
Comentarios	Solo los administradores pueden crear roles.
Acción	Entrada



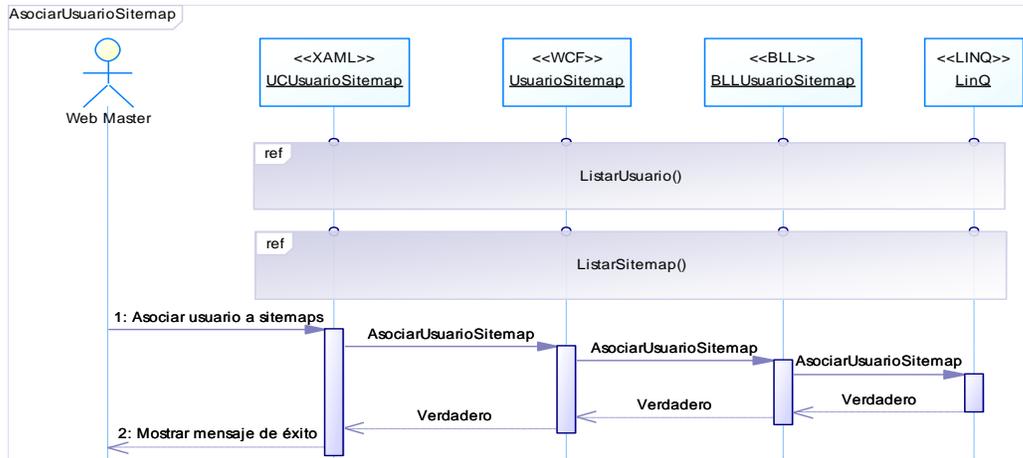
CU-030	Consultar rol	
Requerimientos	RQF-030	
Casos de Uso Relacionados	CU-032	
Actores	ACT-001	
Tipo	Secundario	
Descripción	El sistema debe permitir hacer una consulta de usuario.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster. 2. El registro a consultar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El sistema muestra los datos consultados. 	
Curso Típico de Eventos	<p style="text-align: center;">Actor</p> <ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de roles y decide consultar uno de ellos. 3. El Web Máster selecciona un registro de la lista. 	<p style="text-align: center;">Sistema</p> <ol style="list-style-type: none"> 2. El sistema muestra un listado. 4. El sistema muestra los datos específicos para el registro seleccionado: <ol style="list-style-type: none"> a. Nombre b. Estado
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de roles.	
	Acción 4: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	



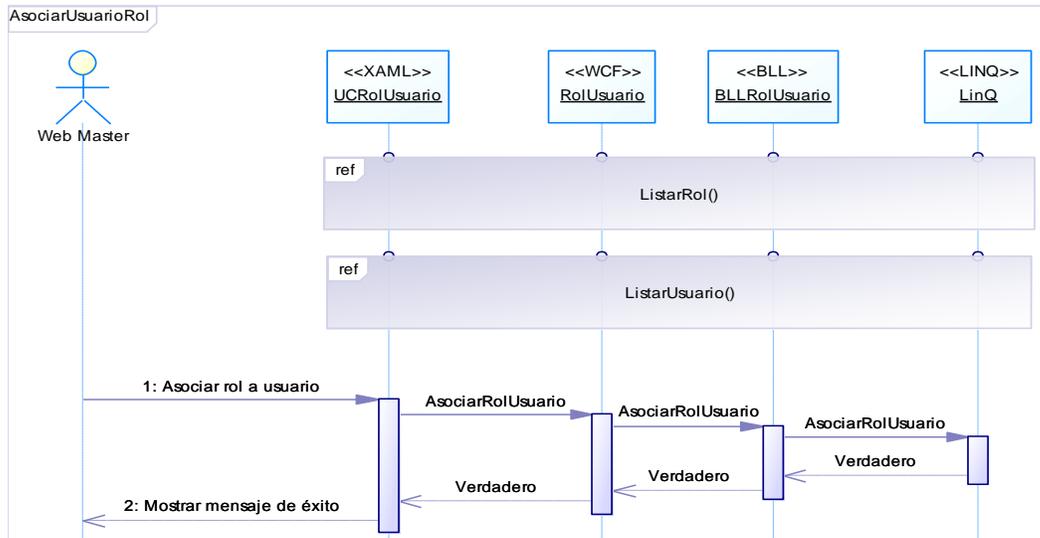
CU-031	Modificar rol	
Requerimientos	RQF-031	
Casos de Uso Relacionados	CU-030	
Actores	ACT-001	
Tipo	Secundario	
Descripción	El sistema debe permitir modificar un rol.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster. 2. El registro a modificar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El registro queda actualizado en el sistema. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de roles y decide modificar uno de ellos. 3. El Web Máster selecciona un registro de la lista. 5. El Web Máster realiza los cambios necesarios sobre los campos que está habilitado a modificar: <ol style="list-style-type: none"> a. Nombre b. Estado 6. El Web Máster presiona el botón de guardar. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado de roles. 4. El sistema muestra los datos específicos para el registro seleccionado. 7. El sistema va a base de datos y actualiza el registro. 8. El sistema muestra un mensaje de transacción exitosa. 9. El sistema muestra la pantalla de mantenimiento de roles.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de roles.	
	Acción 5: El Web Máster o el Editor ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 7: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Entrada	



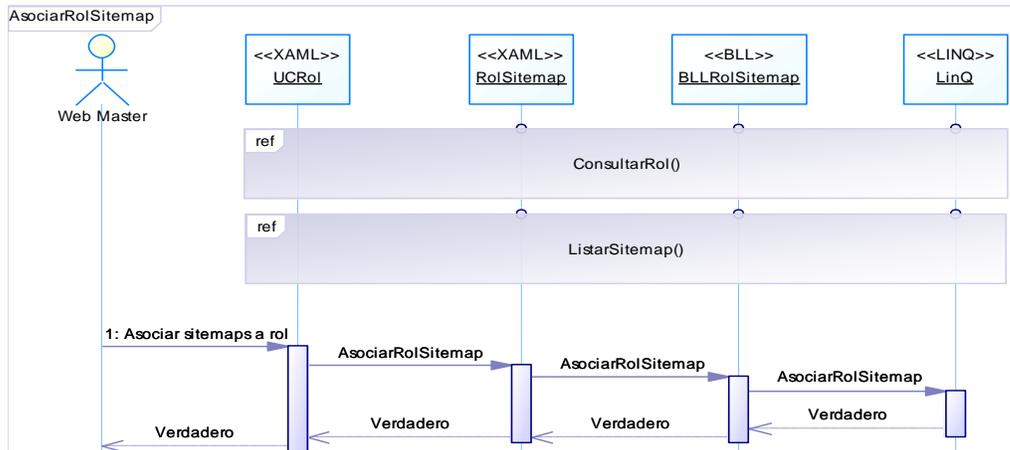
CU-032	Listar roles	
Requerimientos	RQF-032	
Casos de Uso Relacionados		
Actores	ACT-001	
Tipo	Secundario	
Descripción	El sistema debe permitir ver el listado de roles.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El sistema muestra un listado de los datos solicitados, basándose en los criterios dados por el Web Máster.	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de roles y decide ver un listado.</p> <p>3. El Web Máster ingresa los datos solicitados.</p> <p>4. El Web Máster presiona el botón de buscar.</p>	<p>2. El sistema muestra una pantalla con los siguientes criterios:</p> <p>a. Nombre</p> <p>b. Estado</p> <p>5. El sistema muestra un listado con los siguientes datos:</p> <p>a. Nombre.</p> <p>b. Estado.</p>
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de roles.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	



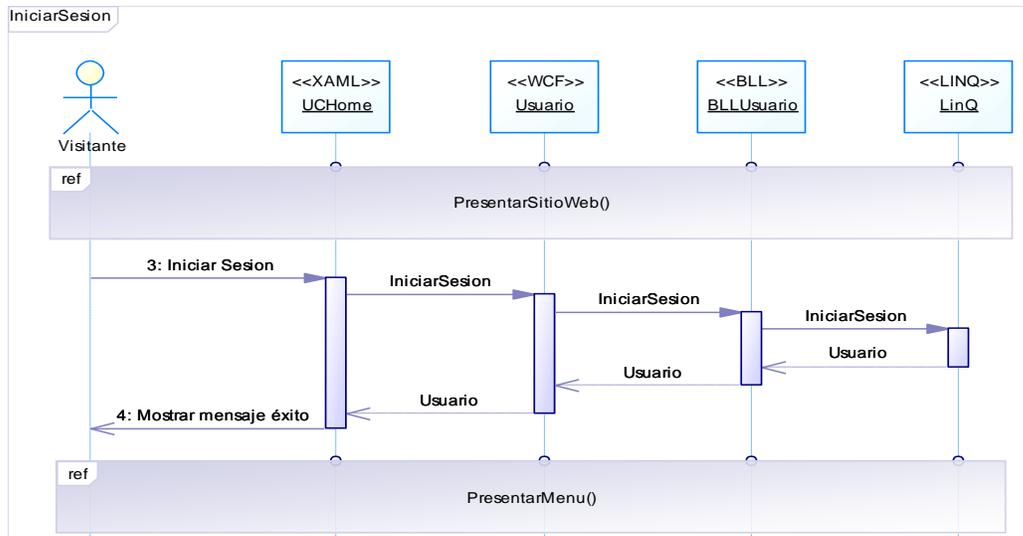
CU-033	Asociar usuario a categoría	
Requerimientos	RQF-033	
Casos de Uso Relacionados	CU-006, CU-028	
Actores	ACT-001, ACT-003	
Tipo	Primario	
Descripción	El sitio web debe permitir asociar un usuario a una categoría.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El usuario queda asociado a las categorías.	
Curso Típico de Eventos	Actor	Sistema
	<p>1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de roles y decide asociar un usuario a categorías.</p> <p>3. El Web Máster selecciona un usuario de la lista y selecciona varias categorías del listado correspondiente.</p> <p>4. El Web Máster presiona el botón guardar.</p>	<p>2. El sitio web muestra un formulario que contiene un listado de usuarios y listados de categorías.</p> <p>5. El sistema va a base de datos y guarda los registros.</p> <p>6. El sistema muestra un mensaje de transacción exitosa.</p> <p>7. El sistema muestra la pantalla de mantenimiento de roles.</p>
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de roles.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	Los editores no pueden modificar permisos de los administradores.	
Acción	Entrada	



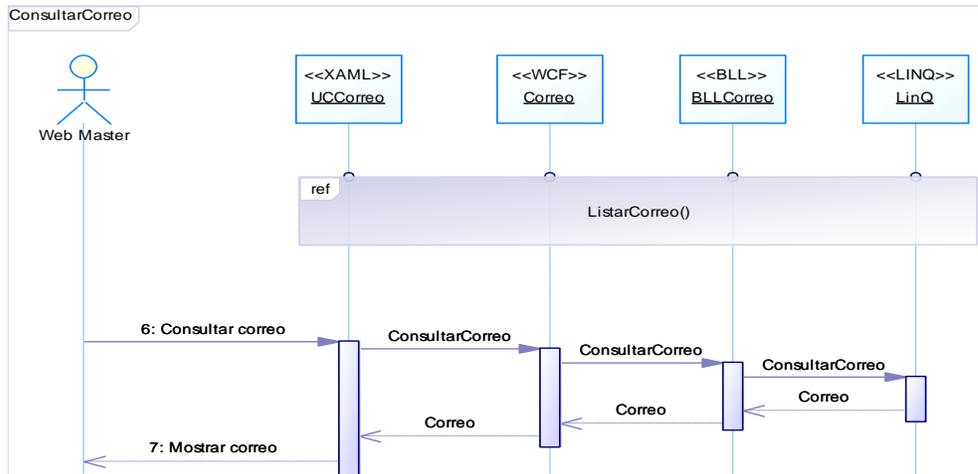
CU-034	Asociar usuario a rol	
Requerimientos	RQF-034	
Casos de Uso Relacionados	CU-028, CU-032	
Actores	ACT-001, ACT-003	
Tipo	Primario	
Descripción	El sitio web debe permitir asociar un usuario a un rol.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El usuario queda asociado al rol.	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de roles y decide asociar un usuario a un rol. 3. El Web Máster selecciona un usuario de la lista y selecciona un rol del listado correspondiente. 4. El Web Máster presiona el botón guardar.	2. El sitio web muestra un formulario que contiene un listado de usuarios y listados de roles. 5. El sistema va a base de datos y guarda los registros. 6. El sistema muestra un mensaje de transacción exitosa. 7. El sistema muestra la pantalla de mantenimiento de roles.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de roles.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	Los editores no pueden modificar permisos de los administradores	
Acción	Entrada	



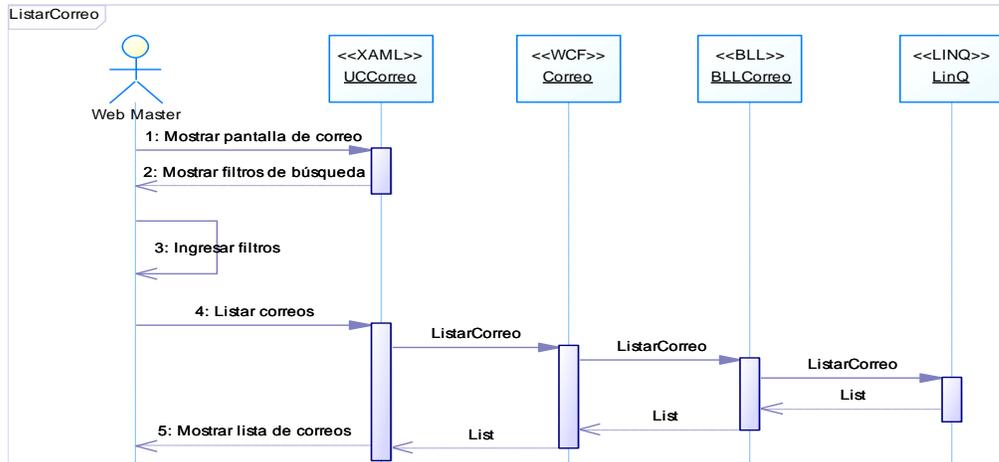
CU-035	Asociar rol a categorías	
Requerimientos	RQF-035	
Casos de Uso Relacionados	CU-006, CU-032	
Actores	ACT-001, ACT-003	
Tipo	Primario	
Descripción	El sitio web debe permitir asociar un rol a categorías.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El rol queda asociado a las categorías.	
Curso Típico de Eventos	<p>Actor</p> <ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de mantenimiento de roles y decide asociar un rol a varios categorías. 3. El Web Máster selecciona un rol de la lista y selecciona varias categorías de la lista correspondiente. 4. El Web Máster presiona el botón guardar. 	<p>Sistema</p> <ol style="list-style-type: none"> 2. El sitio web muestra un formulario que contiene un listado de roles y listados de categorías. 5. El sistema va a base de datos y guarda los registros. 6. El sistema muestra un mensaje de transacción exitosa. 7. El sistema muestra la pantalla de mantenimiento de roles.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de mantenimiento de roles.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	Si se cambian los permisos de un rol, preguntar al usuario, si desea que los cambios se apliquen en los usuarios que tengan dicho rol también. Los editores no pueden modificar permisos de los administradores.	
Acción	Entrada	



CU-036	Iniciar sesión	
Requerimientos	RQF-036	
Casos de Uso Relacionados	CU-008, CU-001	
Actores	ACT-002	
Tipo	Primario	
Descripción	El sitio web debe permitir iniciar sesión a cualquier usuario.	
Precondiciones	1. La pagina Home debe estar desplegada	
Postcondiciones	1. El usuario es identificado en el sistema y se le asocian los permisos correspondientes	
Curso Típico de Eventos	Actor	Sistema
	1. El caso de uso inicia cuando usuario accede a la pagina Home y decide iniciar la sesión.	2. El sistema va a base de datos e identifica al usuario. 3. El sistema actualiza los permisos y accesos del usuario de acuerdo a su rol y permisos asociados a categorías.
	Curso Alternativo:	
	Acción 1: El usuario cancela la operación. El sistema no realiza ninguna operación y muestra la página Home.	
	Acción 1: El usuario ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 2: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Acción 3: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.		
Importancia	Alta	
Comentarios	Recordar que iniciar sesión solo podrán hacerlo usuarios activos	
Acción	Entrada	



CU-037	Consultar correos recibidos (Consultar Contactos)	
Requerimientos	RQF-037	
Casos de Uso Relacionados	CU-038	
Actores	ACT-001	
Tipo	Secundario	
Descripción	El sistema debe permitir hacer una consulta de un correo recibido.	
Precondiciones	<ol style="list-style-type: none"> 1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster. 2. El registro a consultar debe estar previamente ingresado en el sistema. 	
Postcondiciones	<ol style="list-style-type: none"> 1. El sistema muestra los datos consultados. 	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de lectura de correos recibidos y decide consultar uno de ellos. 3. El Web Máster selecciona un registro de la lista. 	<ol style="list-style-type: none"> 2. El sistema muestra un listado. 4. El sistema muestra los datos específicos para el registro seleccionado: <ol style="list-style-type: none"> a. Nombre. b. correo electrónico. c. Teléfono o Fax d. País e. cómo desea ser contactado (Teléfono, Correo) f. Asunto. g. Comentario.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de lectura de correos recibidos.	
	Acción 4: Se presenta un fallo de conexión. El sistema muestra un mensaje de retroalimentación.	
Importancia	Alta	
Comentarios	NA	
Acción	Salida	

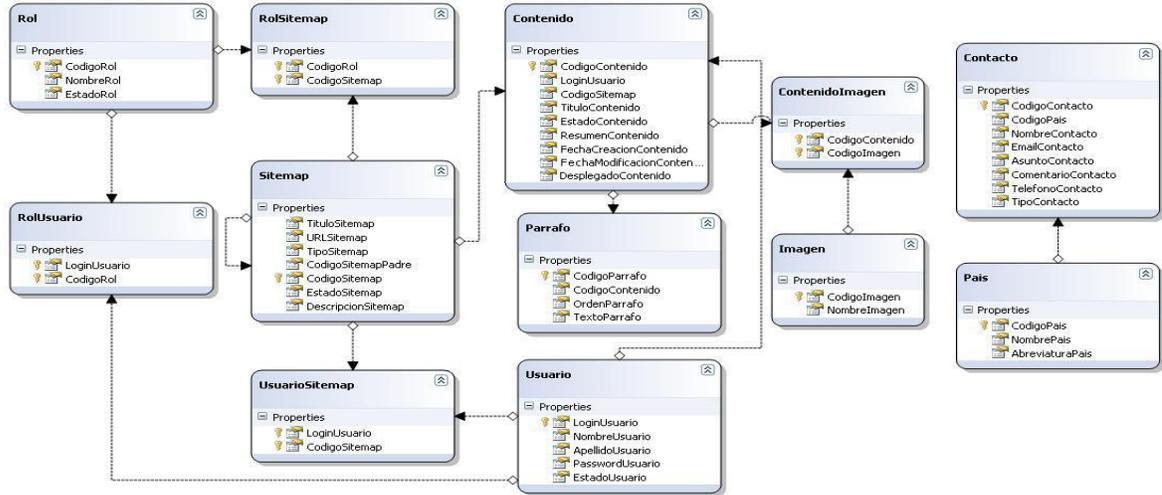


CU-038	Listar correos recibidos (Listar contactos)	
Requerimientos	RQF-038	
Casos de Uso Relacionados		
Actores	ACT-001	
Tipo	Secundario	
Descripción	El sistema debe permitir ver el listado de los correos recibidos por el web Máster.	
Precondiciones	1. El usuario debe estar debidamente identificado en el sitio web con el rol de Web Máster.	
Postcondiciones	1. El sistema muestra un listado de los datos solicitados, basándose en los criterios dados por el Web Máster.	
Curso Típico de Eventos	Actor	Sistema
	<ol style="list-style-type: none"> 1. El caso de uso inicia cuando el Web Máster accede a la pantalla de lectura de correos recibidos. 3. El Web Máster ingresa los datos solicitados. 4. El Web Máster presiona el botón de buscar. 	<ol style="list-style-type: none"> 2. El sistema muestra una pantalla con los siguientes criterios: <ol style="list-style-type: none"> a. Nombre. b. correo electrónico. c. País d. cómo desea ser contactado (Teléfono, Correo) e. Asunto. 5. El sistema muestra un listado con los siguientes datos: <ol style="list-style-type: none"> a. Nombre. b. correo electrónico. c. Teléfono o Fax d. País e. cómo desea ser contactado (Teléfono, Correo) f. Asunto. g. Comentario.
	Curso Alternativo:	
	Acción 3: El Web Máster cancela la operación. El sistema no realiza ninguna operación y muestra la pantalla de lectura de correos.	
	Acción 3: El Web Máster ingresa datos incorrectos. El sistema muestra un mensaje de retroalimentación.	
	Acción 5: Se presenta un fallo de conexión. El sistema muestra un mensaje	

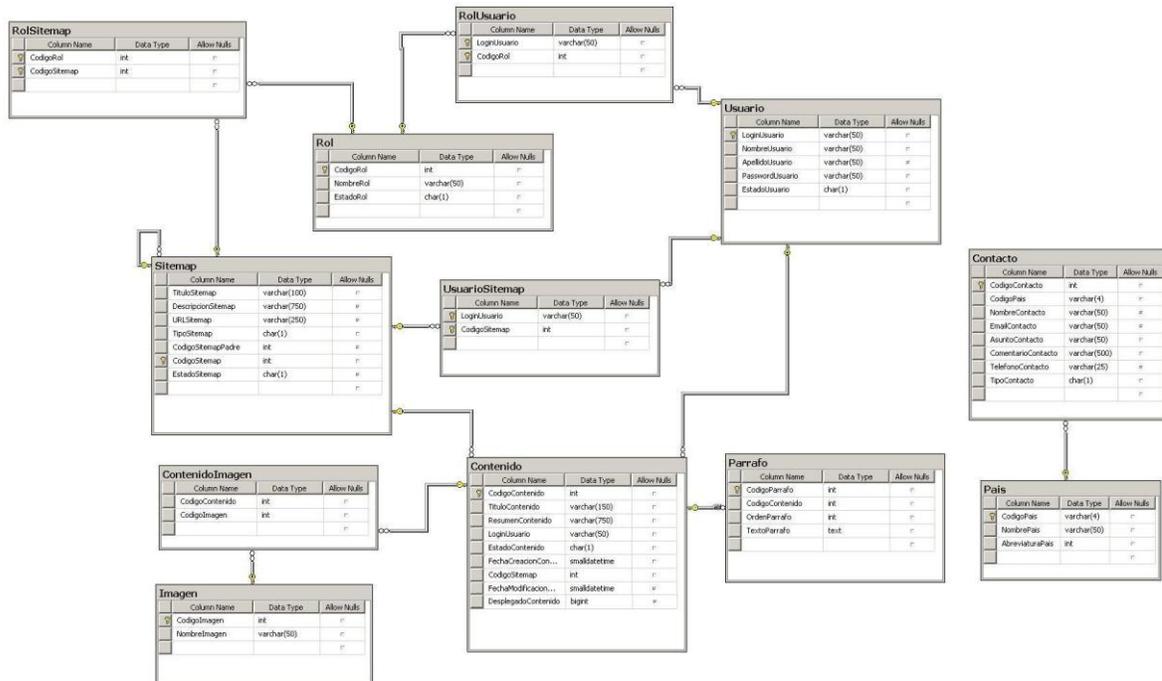
	de retroalimentación.
Importancia	Alta
Comentarios	NA
Acción	Salida

Diccionario de datos

Base de datos



Modelo físico de la base de datos



Lista de tablas

Nombre	Código
Contacto	CONTACTO
Contenido	CONTENIDO
ContenidoImagen	CONTENIDOIMAGEN
Imagen	IMAGEN
Pais	PAIS
Parrafo	PARRAFO
Rol	ROL
RolCategoría	ROLCATEGORÍA
RolUsuario	ROLUSUARIO
Categoría	CATEGORÍA
Usuario	USUARIO
UsuarioCategoría	USUARIOCATEGORÍ
	A

Lista de tablas y campos

Contacto

Nombre	Código	Tipo de datos
CodigoContacto	CODIGOCONTACTO	Serial
NombreContacto	NOMBRECONTACTO	Variable characters (50)
EmailContacto	EMAILCONTACTO	Variable characters (25)
AsuntoContacto	ASUNTOCONTACTO	Variable characters (50)
ComentarioContacto	COMENTARIOCONTACTO	Variable characters (500)
TelefonoContacto	TELEFONOCONTACTO	Variable characters (25)
TipoContacto	TIPOCONTACTO	Characters (1)

Contenido

Nombre	Código	Tipo de datos
CodigoContenido	CODIGOCONTENIDO	Serial
TituloContenido	TITULOCONTENIDO	Variable characters (150)
ResumenContenido	RESUMENCONTENIDO	Variable characters (450)
EstadoContenido	ESTADOCONTENIDO	Characters (1)
FechaCreacionContenido	FECHACREACIONCONTENIDO	Date
FechaModificacionContenido	FECHAMODIFICACIONCONTENID O	Date
DesplegadoContenido	DESPLEGADOCONTENIDO	Long

Imagen

Nombre	Código	Tipo de datos
CodigoImagen	CODIGOIMAGEN	Serial
NombreImagen	NOMBREIMAGEN	Variable characters (50)

Pais

Nombre	Código	Tipo de datos
CodigoPais	CODIGOPAIS	Serial
NombrePais	NOMBREPAIS	Variable characters (50)
AbreviaturaPais	ABREVIATURAPAIS	Integer

Párrafo

Nombre	Código	Tipo de datos
CodigoParrafo	CODIGOPARRAFO	Serial
OrdenParrafo	ORDENPARRAFO	Integer
TextoParrafo	TEXTOPARRAFO	Variable characters (750)

Rol

Nombre	Código	Tipo de datos
CodigoRol	CODIGOROL	Serial
NombreRol	NOMBREROL	Variable characters (50)
EstadoRol	ESTADOROL	Characters (1)

Categoría

Name	Code	Data Type
CodigoCategoría	CODIGOCATEGORÍA	Integer
TituloCategoría	TITULOCATEGORÍA	Variable characters (100)
DescripcionCategoría	DESCRIPCIONCATEGORÍA	Variable characters (750)
URLCategoría	URLCATEGORÍA	Variable characters (250)
TipoCategoría	TIPOCATEGORÍA	Characters (1)
EstadoCategoría	ESTADOCATEGORÍA	Characters (1)

Usuario

Name	Code	Data Type
NombreDeUsuario	NOMBREDEUSUARIO	Variable characters (50)
NombreUsuario	NOMBREUSUARIO	Variable characters (50)
ApellidoUsuario	APELLIDOUSUARIO	Variable characters (50)
ContraseñaUsuario	CONTRASEÑADUSUARIO	Variable characters (50)
EstadoUsuario	ESTADOUSUARIO	Characters (1)

ESCUELA POLITÉCNICA DEL EJÉRCITO

CARRERA DE INGENIERÍA DE SISTEMAS E INFORMATICA

DECLARACIÓN DE RESPONSABILIDAD

Nosotros, Juan Gabriel Peñaherrera Sandoval y Diego Ismael Velasco Sánchez

DECLARAMOS QUE:

El proyecto de grado denominado “DISEÑO Y CONSTRUCCIÓN DE SITIOS DE WEB MEDIANTE LA TECNOLOGÍA RICH INTERNET APLICATIONS (RIA). CASO PRÁCTICO: DISEÑO DEL SITIO WEB DE LA EMPRESA BABEL SOFTWARE DEL ECUADOR” 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.

Juan Gabriel Peñaherrera Sandoval
C.I. No.-0502501976

Diego Ismael Velasco Sánchez
CI. No. 0502603855

CERTIFICACION

Se certifica que el presente trabajo fue desarrollado por Juan Gabriel Peñaherrera Sandoval y Diego Ismael Velasco Sánchez, bajo nuestra supervisión.

Ing. Javier Montaluisa
DIRECTOR DE PROYECTO

Ing. José Carrillo
CODIRECTOR DE PROYECTO

Ing. Edison Espinosa
COORDINADOR DE LA CARRERA DE SISTEMAS E INFORMÁTICA

Dr. Rodrigo Vaca
SECRETARIO ACADEMICO