

## **Capítulo 5. Implementación de Software**

En el presente capítulo, se explicarán, describirán y mostrarán las complicaciones que se tuvieron para el desarrollo del Módulo Administrativo de Portales Interactivos. Dentro de las pruebas se simuló pruebas de concurrencia y un tipo de pruebas de funcionalidad, las pruebas de concurrencia fueron simuladas desde diferentes computadoras accediendo al mismo sitio construido con el Módulo Administrativo de Portales, ya que no se cuenta con un software especial para realizar este tipo de pruebas.

### **V.1. Herramientas del Sistema.**

Antes de explicar las pruebas y contrariedades que se tuvieron para el desarrollo del software tenemos que empezar por mostrar el resultado del trabajo hecho. Empezamos por el Back End, el cual en su pantalla principal cuenta con todas las opciones existentes dentro del software, en este capítulo no mencionaremos cómo utilizar cada una de las opciones, eso lo haremos en el manual de usuario.



Figura 59. Pantalla principal del Back End.

Como se puede ver contamos con 11 diferentes opciones, pero las importantes en esta tesis son las de Base de Datos, Secciones, Artículos, Templates y Noticias. Podemos ver que en la parte inferior de la pantalla existe un botón para salir de la aplicación.

Cada una de las pantallas de Base de Datos, Secciones, Artículos, Templates y Noticias contiene con tres botones en la parte superior, cada uno de ellos corresponde a la opción de agregar, eliminar o modificar. Debajo de ellos se encuentra el título de herramienta escogida, por debajo de esta se encuentra un botón en la parte izquierda, el cual está ligado a la pantalla principal de la opción, es decir si estamos en la opción artículos este botón nos llevara únicamente a la pantalla principal de artículos; del lado contrario de este botón existe otro botón en forma de casa, el cual nos lleva a la pantalla principal del Back End.

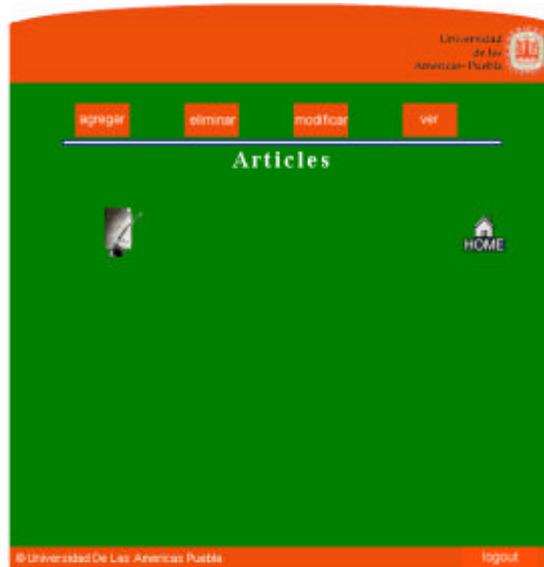


Figura 60. Pantalla de segundo nivel del Back End.

Todas y una cada una de las pantallas del BackEnd cuentan con esta estructura, Nereo la información que se despliega para cada opción del sistema es la correspondiente.

## V.2. Ejemplo de Portal desarrollado con MACP.

Como ejemplo de un sitio construido con el Módulo Administrativo de Portales mostraremos un portal pequeño, el cual contiene todos y cada uno de los capítulos de esta tesis. Este sitio cuenta con un menú en el lado izquierdo, en el cual se encuentra el menú de secciones (ligas a cada uno de los capítulos), además la página principal de este sitio cuneta con un botón llamado “correo”, por medio del cual se puede mandar un mail al webmaster del sitio.



Figura 61. Pantalla principal (index) del sitio generado con el MACP.

Como dijimos anteriormente el menú de la parte izquierda son ligas a cada una de esas secciones. Cada sección cuenta con un template muy parecido, en el cual se guarda el mismo espacio para desplegar el menú, pero en la parte central de la página se encuentra la información correspondiente a la sección; otro característica que tiene los template de las secciones es que en la parte superior derecha existen dos Borne en forma de flecha, los cuales tienen la funcionalidad de scrollbar. En la parte inferior del template no sólo se muestra el botón llamado “correo”, sino que también se muestra un botón llamado “regresar”, el cual nos enviará a la página que se navegó anteriormente; Otro botón que aparece es el botón llamado “inicio”, el cual nos direcciona al “home” del sitio.

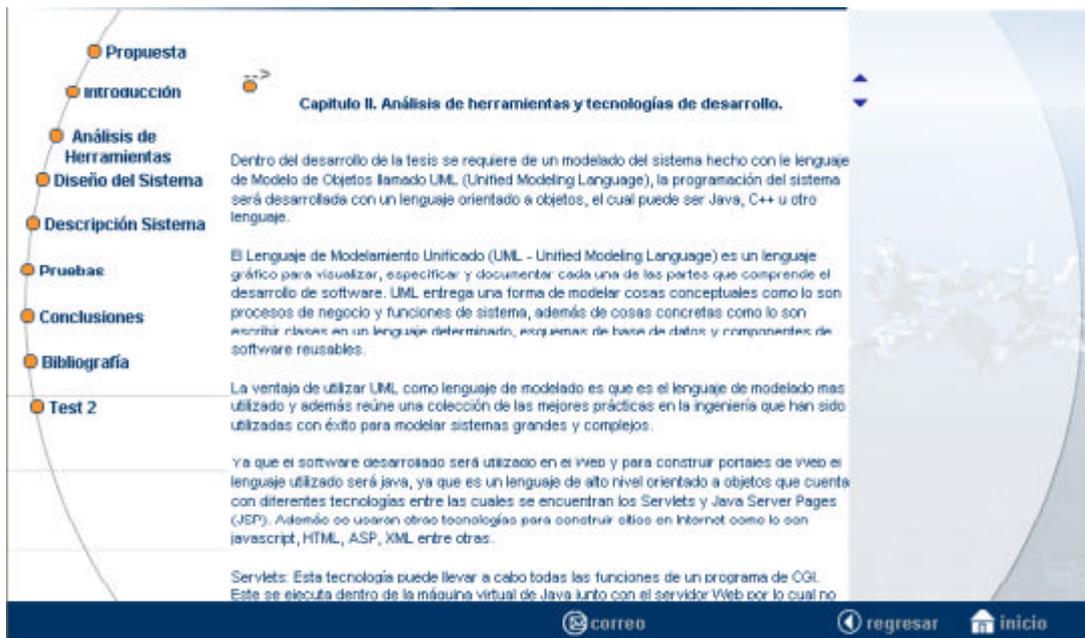


Figura 62. Pantalla de una sección del sitio desarrollado con el MACP.

Recordemos que el sitio que se presentó anteriormente se diseñó exclusivamente para mostrar la información de esta tesis, por lo cual la funcionalidad aplica totalmente al sitio, es decir, cada sitio tendrá una funcionalidad específica y no quiere decir que este tipo de sitios sean los únicos que pueden desarrollarse con esta herramienta, la funcionalidad, el diseño gráfico y la distribución de la información depende de cada sitio.

Comenzado con los parámetros establecidos, se construyeron cuatro servlets (BackEnd) para controlar la información, dos java beans (FrontEnd) para desplegar la información en el sitio. Estos componentes interactúan con algunos componentes desarrollados dentro de la Tesis de Juan Carlos Korzi, los cuales se encargan de controlar las políticas de uso del Módulo Administrativo de Portales Interactivos, de igual manera se utilizó una Base de Datos en la cual se encuentra la información tanto de las políticas de

uso (Tesis Juan Korzi) del sistema, como del Administrador de Contenido de Portales Interactivos (Tesis Luis Vidrio).

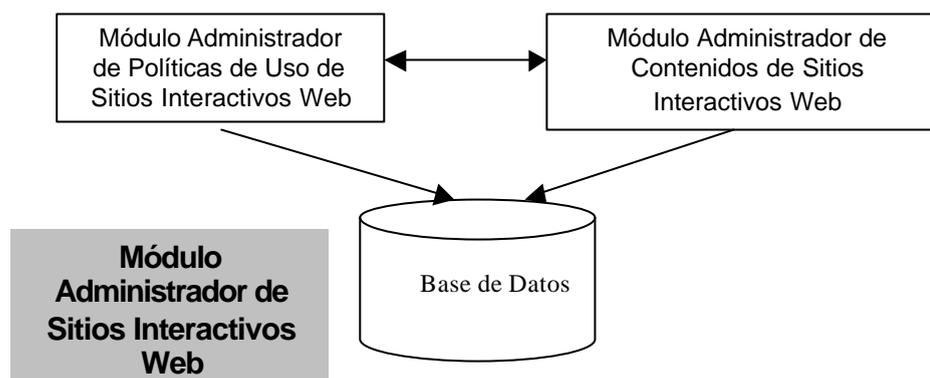


Figura 63. Diagrama general del proyecto Módulo Administrativo de Portales.

### V.3. Problemas de Implementación Sistema.

Otro tipo de pruebas que se realizó, gracias al diseño de la aplicación fue la prueba del sistema con diferentes tipos de base de datos como lo fueron MSQlServer 7.0, MySQL, Oracle y Access. Cabe mencionar que las pruebas que se realizaron con estas pruebas fueron sencillas ya que únicamente se realizó la conexión con la base de datos y se insertó información en algunos registros, es decir, no se construyó un sitio con cada manejador de base de datos; El sitio construido fue con la base de datos MSQlServer 7.0. Un problema que se presenta al utilizar una base de datos diferente es el conseguir el driver adecuado para cada base de datos y su versión. Como ya se mencionó la tesis fue desarrollada con la base de datos MSQlServer 7.0 y su driver llamado Ashna, por lo cual quien utilice este software y requiera utilizar otro tipo de base de datos necesitara conseguir

el driver adecuado y configurar el sistema como se menciona en el manual de usuario, en la parte de base de datos.

DBname:	bowm
password:	bowm
port:	1433
user:	bowm_admin
connUrl:	jdbc:Turbo://140.148.4.38:1433/bowm
server:	140.148.4.38
jdbcDriver:	com.asima.jdbc.driver.Driver

Figura 64. Pantalla del Back End para la Configuración de Base de Datos.

Dentro de la estructura de la base de datos, es decir, las tablas, se tuvieron que modificar dos de ellas para lograr que el Módulo Administrativo de Portales fuese lo más genérico posible y se puedan desarrollar portales pequeños y portales medianos. Las tablas que fueron modificadas son la tabla **pg\_seccion**, la tabla **pg\_articulo** y se creó la tabla **pg\_idioma**.

La tabla **pg\_idioma** es aquella en la que se almacenan los diferentes idiomas a utilizar para crear un sitio.

### PG\_IDIOMA

Id_idioma (PK)	Numeric(6,0)	Not null
descripcion	Varchar(50)	null

Tabla 4. Tabla pg\_idioma de Base de Datos.

La tabla **pg\_seccion** es la tabla que arma toda la estructura del portal a desarrollar con el Módulo Administrativo de Portales, se debe tener cuidado al eliminar una sección ya que esta tabla es la que cuenta con más relaciones. Las modificaciones que sufrió fueron el aumentar id\_idioma para poder tener una misma sección en diferentes idiomas.

### PG\_SECCION

Id_seccion(PK)	Numeric(3)	Not null
Secc_padre(PK)	Numeric(3)	Null
Id_template(FK)	Numeric(3)	Not null
Id_idioma(FK)	Numeric(3)	Not null
Descripcion	Varchar(100)	Null
nivel	Numeric(3)	Not null

Tabla 5. Tabla pg\_seccion de Base de Datos.

La tabla **pg\_articulo** es tabla que contendrá más información ya que contendrá la información final del sitio a mostrar al usuario. Se le han agradao los campos de *id\_idioma* para tener los Articulos en más de un idioma y para poder identificarlos en cada versión; el campo *resumen* sirve para tener una pequeña descripción del artículo; el campo *ticker* sirve para identificar si un artículo también funciona como noticia, las cuales son manejadas por medio de un componente del sistema.

### PG\_ARTICULO

Id_articulo	Numeric(3)	Not null
Id_idioma	Numeric(3)	Not null
Id_seccion	Numeric(3)	Null
Id_template	Numeric(3)	Null
Titulo	Varchar(255)	Not null
Resumen	Varchar(255)	Null
Expira	Bit default 0	Not null
Ticker	Bit default 0	Not null

Tabla 6. Tabla pg\_articulo de base de datos.

Dentro del modulo de templates, se hicieron cambios en la manera de recibir los archivos para almacenar un template dentro del servidor, ya que en un principio, se solicitaban los siguientes datos:

- ?? Nombre Template.
- ?? Número de Campos.
- ?? Número de Imágenes.
- ?? Archivo HTML.
- ?? Imágenes.
- ?? ScrennShot.

Pero para que el sistema fuese genérico se tuvo que modificar recibiendo un archivo \*.zip el cual contiene tanto los las imágenes a utilizar por el template, como el archivo HTML, los archivos javascript a utilizar por el template y la hija de Estilos (\*.css) a utilizar por el template. Y el screenshot se recibe como una sola imagen.



Figura 65. Pantalla del BackEnd para agregar un nuevo Template.

Este cambio requirió un trabajo extra en Java, ya que por medio de este lenguaje el archivo \*.zip es descomprimido y puesto en una carpeta temporal del servidor para que el sistema valide si los archivos enviados por el cliente son archivos válidos (html, jpg, gif, swf, js, css), en caso de que no sean archivos válidos el sistema mandará un mensaje y el template no podrá ser almacenado en el servidor, ni en la base de datos. Pero cuando los archivos son válidos, los archivos pasan de la carpeta temporal del servidor y la carpeta correspondiente del template. La estructura de un templates es la siguiente:

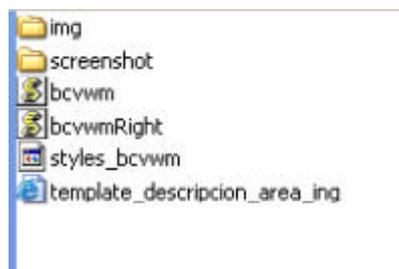


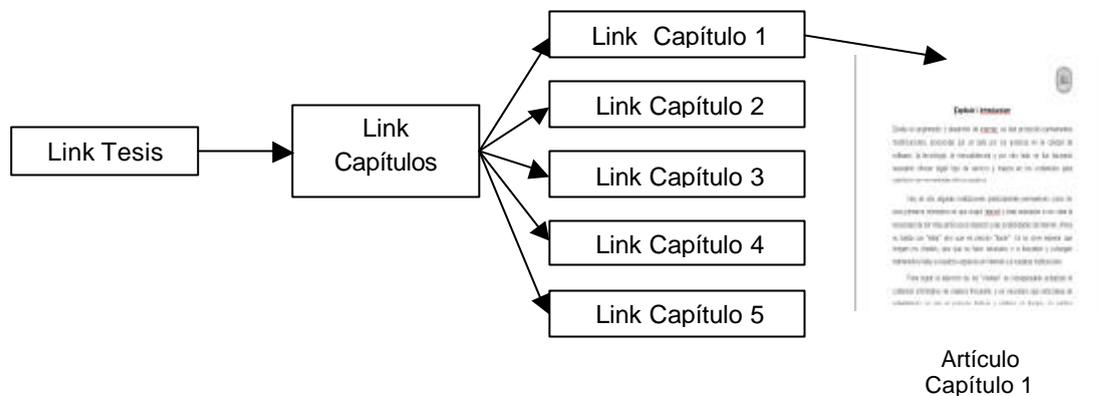
Figura 66. Estructura de Directorios para el archivo zip para un template.

Una vez corregidos todos los puntos anteriores, se desarrollo un mini sitio con cuatro secciones, y una de estas secciones con dos subsecciones, y varios artículos. A este mini sitio se realizaron varios tipos de pruebas, como lo son las pruebas de funcionalidad, pruebas de “conurrencia” y pruebas de “stress”.

#### **V.4. Pruebas de Funcionalidad.**

Las pruebas de funcionalidad fueron importantes ya que por medio de ellas se verifico que el sitio desarrollado contara con la funcionalidad desea, es decir, al oprimir un link nos enviará la página correspondiente y ejecutará las acciones correspondientes a los botones de home, back, language y mapa del sitio.

Un problema que se presento dentro de las pruebas de funcionalidad fue el que los botones de back y mapa del sitio no funcionaban adecuadamente ya que el código de las variables de ambiente no se mostraba correctamente por un error de programación, pero el error fue corregido y ahora funciona correctamente.



**Diagrama de Pruebas de Funcionalidad**

Figura 67. Diagrama para representar las pruebas de funcionalidad realizadas al MACP.

### V.5. Pruebas de Concurrencia.

Las pruebas de “conurrencia” se simularon conectándose desde 10 computadoras de la Universidad al sitio desarrollado (<http://140.148.4.38/bcvwm/index.jsp>), para ello se accedía al link de capítulos y posteriormente al link de capítulo 2, esto fue realizado 10 veces por cada usuario. Con ello se asegura que el sistema responde oportunamente a las peticiones de los usuarios y aunque en algunas ocasiones tarda un poco más de tiempo en contestar, pero no se pierde la comunicación entre el cliente y servidor.

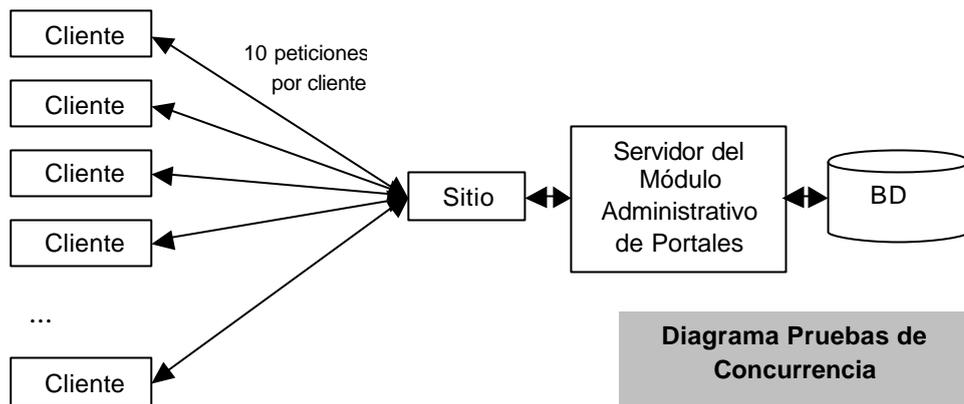


Figura 68. Diagrama de representación de las pruebas de concurrencia realizadas en el MACP.

Para mejorar el tiempo de respuesta dentro de las pruebas de concurrencia se aumento el tiempo de Intercambio de Datos del ServletExec ISAPI (Engine que ejecuta los Servlets), el cual tenia un tiempo por default de 10 segundos y fue aumentada a 30 segundos, lo cual mejoro el tiempo de respuesta en las peticiones de los clientes.

<b>Max Residents:</b>	<input type="text" value="1024"/>
<b>Max Sessions:</b>	<input type="text" value="1024"/>
<b>Swap Directory:</b>	<input type="text" value="sessionSwap"/>
<b>Swap Interval:</b>	<input type="text" value="30"/> Seconds
<b>Invalidation Interval:</b>	<input type="text" value="10"/> Seconds
<b>Invalidation Time:</b>	<input type="text" value="30"/> Minutes

Figura 69. Pantalla de Configuración el Servlet Exec ISAPI para aumentar el tiempo de Swap.

Actualmente el sistema funciona sin ningún error de programación, los problemas posteriores que pueden presentarse son respecto a agregar nuevos módulos para controlar información. Seguramente los errores más comunes al construir un sitio serán por falta de experiencia en esta herramienta, pero para ello se cuenta con un manual de usuario.