



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“CREACIÓN DE UN CONTENEDOR DE WIDGETS”

TESINA DE SEMINARIO

Previo a la obtención del Título de:

INGENIERO EN COMPUTACIÓN ESPECIALIZACIÓN

SISTEMAS TECNOLÓGICOS

Presentado por:

GRACIELA DENISSE SALTOS CONTRERAS

TIZIANA MARÍA HERRERA MEDINA

Guayaquil – Ecuador

2010

AGRADECIMIENTO

A Dios por su eterna misericordia,
por abrirnos las puertas del triunfo
y enseñarnos el camino que
debemos seguir.

A nuestros padres, hermanos, y
familiares por su apoyo y
confianza.

DEDICATORIA

A Dios a nuestros padres,
hermanos y familiares.

A Juan Carlos por su paciencia,
comprensión y apoyo
incondicional.

A mis hijas, por ser mi fuente de
inspiración y fortaleza para lograr
mis objetivos.

TRIBUNAL DE SUSTENTACIÓN

MSc. Fabricio Echeverría

PROFESOR DEL SEMINARIO DE GRADUACIÓN

MSc. Carmen Vaca

PROFESOR DELEGADO POR EL DECANO

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este Trabajo de Graduación, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITECNICA DEL LITORAL”.

Graciela Denisse Saltos Contreras

Tiziana María Herrera Medina

RESUMEN

En el presente trabajo se realizó el análisis, diseño, implementación y pruebas de un Repositorio de Widgets junto con cuatro widgets, los mismos que son aplicaciones diseñadas para mostrar información a manera de texto, gráfico, audio o video y pueden ser utilizados en blogs o sitios web.

En el primer capítulo, se detalla una visión general del Google Web Toolkit (GWT), que es el Framework utilizado para el desarrollo de cada widget.

En el segundo capítulo, se describe la funcionalidad del repositorio de widgets junto con las cuatro aplicaciones creadas, detallando cada caso de uso así como también el respectivo diagrama de interacción de objetos.

En el tercer capítulo, se explica la arquitectura del repositorio de widgets junto con el diseño de la interfaz de los Widgets.

En el cuarto capítulo, se detalla las herramientas utilizadas y las soluciones que se plantearon en el transcurso de la implementación de las aplicaciones.

Finalmente el capítulo 5, indica las pruebas ejecutadas con los estudiantes y la encuesta realizada, esperando como resultado que los widgets sean del agrado del estudiante y de fácil uso.

ÍNDICE GENERAL

RESUMEN	VI
ÍNDICE GENERAL	VII
ABREVIATURAS	IX
ÍNDICE DE FIGURAS	X
ÍNDICE DE TABLAS	XI
INTRODUCCIÓN	XII
Capítulo 1	1
1. Visión General	1
1.1 Introducción	1
1.2 Arquitectura de Google Web Toolkit	2
1.3 Componentes básicos de una aplicación desarrollada con GWT	4
Capítulo 2	5
2. Análisis del Requerimiento	5
2.1 Introducción	5
2.2 Funcionalidad	5
2.3 Descripción de los Casos de Uso	8
2.3.1 Caso de Uso Visualizar Contador de Sesiones	9
2.3.2 Caso de Uso Visualizar Contador de Visitas	9
2.3.3 Caso de Uso Consultar Ranking Académico	10
2.3.4 Caso de Uso Visualizar Nube de Palabras	11
2.4 Diagrama de Interacción de Objetos	12
2.4.1 Escenario 1.1: Conteo exitoso de visitas del usuario	12
2.4.2 Escenario 2.1: Conteo exitoso de visitas a la página	13
2.4.3 Escenario 3.1: Consulta exitosa del Ranking Académico	14
2.4.4 Escenario 4.1: Visualización exitosa de la Nube de Palabras	15

Capítulo 3	16
3	Diseño de la Aplicación 16
3.1	Arquitectura del Repositorio de Widgets 16
3.2	Estructura básica de una aplicación web usando GWT 17
3.3	Diseño de la Interfaz de Widgets 18
3.3.1	Interfaz del Repositorio de Widgets 20
3.3.2	Interfaz del Contador de Sesiones 21
3.3.3	Interfaz del Contador de Vistas 22
3.3.4	Interfaz del Ranking Académico 23
3.3.5	Interfaz de la Nube de Palabras 24
Capítulo 4	25
4	Implementación de la Aplicación 25
4.1	Introducción 25
4.2	Requerimientos de Hardware 25
4.3	Requerimiento de Software 26
4.4	Detalle de la implementación 26
4.4.1	Herramientas de desarrollo 27
4.4.2	Problemas en la implementación 27
Capítulo 5	29
5	Pruebas y Resultados 29
5.1	Introducción 29
5.2	Pruebas 29
5.3	Resultados 33
CONCLUSIONES Y RECOMENDACIONES	36
ANEXO	37
BIBLIOGRAFÍA	38

ABREVIATURAS

GWT	Google Web Toolkit
RPC	Remote Procedure Call
API	Aplication Programming Interface
IDE	Integrated Development Environment
RIA	Aplicaciones Ricas en Internet
JDK	Java Development Kit
JSP	JavaServer Pages
HTML	HyperText Markup Language
CSS	Cascading Style Sheets

ÍNDICE DE FIGURAS

FIGURA	DESCRIPCIÓN	PÁG.
Figura 1	Arquitectura GWT	3
Figura 2	Diagrama de Casos de Uso	8
Figura 3	Diagrama Interacción Objeto Escenario 1.1	12
Figura 4	Diagrama Interacción Objeto Escenario 2.1	13
Figura 5	Diagrama Interacción Objeto Escenario 3.1	14
Figura 6	Diagrama Interacción Objeto Escenario 4.1	15
Figura 7	Estructura de aplicación web usando GWT	17
Figura 8	Repositorio de Widgets	20
Figura 9	Contador de Sesiones	21
Figura 10	Contador de Visitas	22
Figura 11	Ranking Académico	23
Figura 12	Nube de Palabras	24
Figura 13	Copia de código del widget	30
Figura 14	Ventana de administrador de blog	31
Figura 15	Cargar Widget de texto	31
Figura 16	Pegar código de widget	32
Figura 17	Blog con widget Ranking Académico	32
Figura 18	Estadística de uso del widget	33
Figura 19	Estadística de cuál llamó más la atención	34
Figura 20	Estadística de facilidad de uso	34
Figura 21	Estadística de mejoras del widget	35
Figura 22	Estadística de recomendación	35

ÍNDICE DE TABLAS

TABLA	DESCRIPCIÓN	PÁG.
Tabla I	Rango de Notas	6
Tabla II	Rango de repetición de palabras	7
Tabla III	Caso de Uso 1	9
Tabla IV	Escenario 1.1	9
Tabla V	Escenario 1.2	9
Tabla VI	Caso de Uso 2	9
Tabla VII	Escenario 2.1	10
Tabla VIII	Escenario 2.2	10
Tabla IX	Caso de Uso 3	10
Tabla X	Escenario 3.1	10
Tabla XI	Escenario 3.2	11
Tabla XII	Escenario 3.3	11
Tabla XIII	Caso de Uso 4	11
Tabla XIV	Escenario 4.1	11
Tabla XV	Escenario 4.2	12
Tabla XVI	Interfaz del Repositorio de Widgets	20
Tabla XVII	Interfaz Contador de Sesiones	21
Tabla XVIII	Interfaz Contador de Visitas	22
Tabla XIX	Interfaz Ranking Académico	23
Tabla XX	Interfaz Nube de Palabras	24

INTRODUCCIÓN

La Web 2.0 está basada en la interacción que se logra a partir de diferentes aplicaciones web, permitiendo a sus usuarios interactuar con otros usuarios, en contraste a sitios web no-interactivos donde los usuarios se limitan a la visualización pasiva de información que se les proporciona. [1]

Los Widgets o Gadgets son pequeñas aplicaciones, basadas en Web 2.0, que pueden ser instaladas en un blog, red social favorita o descargarlo en tu computador, permitiéndote recibir contenidos que pueden ser mostrados en formato imágenes, video, audio y texto de cualquier temática o funcionalidad.

Estas aplicaciones llegan en la actualidad a 178 millones de usuarios de Internet y tienen como objetivo dar fácil acceso a funciones que son utilizadas frecuentemente, así como también el proveer información visual en tiempo real. [2]

En este proyecto se han creado cuatro Widgets, los cuales se pueden incorporar en su sitio web o blog con tan solo copiar un pedazo de código y automáticamente ya está en funcionamiento. Para la realización de los widgets se utilizó el Framework Google Web Toolkit, la misma que hace posible desarrollar y depurar aplicaciones Web 2.0, utilizando Java como lenguaje de programación.

Esta aplicación tiene como objetivos:

- Diseñar e implementar una interfaz web que servirá para mostrar y descargar los widgets creados.
- Diseñar e implementar cuatro widgets para uso del estudiante en su sitio web o blog.
- Evaluar la funcionalidad de la interfaz web y los widgets con ayuda de encuestas realizadas a un grupo de estudiantes.

Capítulo 1

Visión General

1.1 Introducción

Los widgets son pequeñas aplicaciones portables de fácil instalación y uso, los mismos que brindan funcionalidades o características atractivas tanto a páginas web, computadores o telefonía móvil.

Según su funcionamiento y entorno los mismos se clasifican en:

- Widgets para web.- Son pequeñas aplicaciones portables de fácil uso que atraen a más visitantes a nuestro sitio, este tipo de widgets hace que el sitio tenga más funcionalidades, brinde más servicios, sin la necesidad de una ardua programación, o modificación de la misma.
- Widgets de escritorio.- Este tipo de widgets son descargados e instalados en las máquinas de los clientes por lo general permiten realizar búsquedas en el ordenador instalado u obtener algún tipo de información que el usuario crea conveniente como la hora, el clima, etc.
- Widgets Móviles: Son similares a los de escritorio con la diferencia de que los mismos son instalados en dispositivos móviles o celulares. [2]

Una de las maneras de desarrollar los widgets es mediante el Framework OpenSource GWT por sus siglas en inglés (Google Web Toolkit). Se escogió utilizar este Framework ya que en este proyecto se implementan widgets web y GWT hace posible la creación de los mismos, en tanto que con Yahoo Widgets sólo se pueden implementar widgets de escritorios, los cuales son desarrollados en sus servidores.

Cabe recalcar que el Framework es libre pero su compilador no, dicho compilador es el corazón del Framework, el cual permite generar código JavaScript a partir de código Java, el objetivo principal de este Framework es el desarrollo de aplicaciones RIA's (Aplicaciones Ricas en Internet), considerando que una de las tecnologías más importantes en el desarrollo de este tipo de aplicaciones es el AJAX, GWT está orientado a manejar esta tecnología, para que la misma sea fácil de usar, depurar y dar mantenimiento, además de poder crear el front-end de dichas aplicaciones sin preocuparnos por : JSP, JavaScript, HTML, etc. Es decir, todo lo que una aplicación de este tipo conlleva y resulta muchas veces muy complicada su codificación, depuración y mantenimiento.

1.2 Arquitectura de Google Web Toolkit

La arquitectura de GWT está compuesta por 4 componentes como muestra el siguiente gráfico:



Figura 1. Arquitectura GWT. [3]

- JRE Librerías de Emulación.- Está compuesta de las principales librerías para la creación de programas Java como son java.lang y un subconjunto de las java.util, el resto de las clases no son soportadas por GWT.
- Librerías GWT UI.- Contiene los componentes necesarios para la creación de la interfaz de usuario.
- Compilador GWT (Java a JavaScript).- Permite la traducción del código generado en Java a JavaScript el mismo es usado cuando ejecutamos la aplicación desarrollada en modo web.
- Navegador Web embebido.- Permite ejecutar la aplicación en la máquina virtual de Java sin necesidad de traducirla a este modo de ejecución, se le conoce como ejecución en modo host. [3]

1.3 Componentes básicos de una aplicación desarrollada con GWT

Las aplicaciones desarrolladas con GWT, constan de los siguientes componentes:

- **Módulo.-** Es un paquete distribuible de la aplicación, debido a que lo constituyen los servicios, librerías, archivos, etc. que se encuentran del lado del servidor, estos pueden heredar de otros módulos y compartir recursos con los mismos.
- **La interfaz EntryPoint.-** Esta interfaz contiene un solo método a implementar que es el `onModuleLoad()`, la clase que implemente este método es el punto de entrada de la aplicación lo cual sería equivalente a el método `main()` en una aplicación Java StandAlone.
- **La página Host.-** Es la página HTML generada por el Framework, y donde se carga la aplicación, ésta posee una referencia al archivo llamado `Module.nocache.js` el mismo que contiene la referencia a archivos externos o librerías.
- **La clase RootPanel.-** Es el componente principal de la interfaz de usuario dentro del mismo se visualizan o agregan todos los widgets a ser vistos por el usuario, debido a que él es el acceso directo a la sección `body` de la página `host HTML`, todos los widgets de una aplicación son agregados a este panel principal de manera explícita o implícita. [4]

Capítulo 2

Análisis del Requerimiento

2.1 Introducción

En este capítulo se realizará un análisis de la funcionalidad del Repositorio de Widgets y sus aplicaciones creadas. Así mismo, se detallará cada Caso de Uso y sus respectivos Diagramas de Interacción de Objetos.

2.2 Funcionalidad

El Repositorio de widgets contiene cuatro aplicaciones que cumplen funcionalidades específicas, las mismas que se detallan a continuación:

- Repositorio de Widgets.- Tiene como finalidad explicar en breves pasos qué es un Widget y cómo cargarlo, así mismo muestra el código de los widgets creados para que sean utilizados por los usuarios.
- Widget Contador de Sesiones.- Muestra el número de veces que el usuario ha ingresado a la página donde se encuentra cargado el widget.
- Widget Contador de Visitas.- Cuenta la cantidad de usuarios que han visitado el blog del estudiante y funciona capturando la

dirección de origen de la página donde esté cargado el Widget, valida que exista dentro de la base de datos para continuar el conteo y en el caso de no existir crea un nuevo registro para dicha página.

- Widget Ranking Académico.- En este widget, el estudiante debe ingresar su número de matrícula, se valida que se encuentre en la base de datos y mediante RPC se envía como resultado las materias en que se encuentra registrado junto con gráficos de caritas felices o tristes según sea la nota.

Se tiene cuatro gráficos diferentes que se muestra dependiendo del rango en que se encuentra la nota del estudiante.

El rango está dado por:

Rango de Notas	Resultados
0 – 39	
40 – 59	
60 – 79	
80 – 100	

Tabla I. Rango de Notas

- Widget Nube de Palabras.- Esta aplicación cuenta las veces que se repite una palabra de un archivo tipo texto y la muestra en diferentes tamaños y colores, dependiendo de la cantidad de veces que se repita.

El archivo debe ser de extensión .txt y al iniciar la aplicación debe ser subido en el Widget para obtener el conteo de las palabras.

Este widget hace la transferencia del archivo mediante Servlets debido a la limitante que tiene el GWT al no poder transferir un archivo usando RPC. Para realizar este Widget se cambió el IDE de desarrollo, ya que el Plugin que se utiliza en NetBeans no está lo suficientemente estable para realizar la transferencia del archivo, motivo por el cual utilizamos Eclipse como segunda opción de desarrollo para realizar la transferencia.

Rango de repetición	Tamaño Letra	Color Letra
0 – 10	10 px	Rojo
11 – 20	12 px	Verde
21 – 30	14 px	Gris
31 – 40	16 px	Azul
>= 40	18 px	Naranja

Tabla II. Rango de repetición de palabras

2.3 Descripción de los Casos de Uso

Los Casos de Uso nos permiten visualizar de manera gráfica la interacción entre el sistema desarrollado y el usuario. A continuación se muestra el diagrama de Casos de Uso del Repositorio de Widgets.

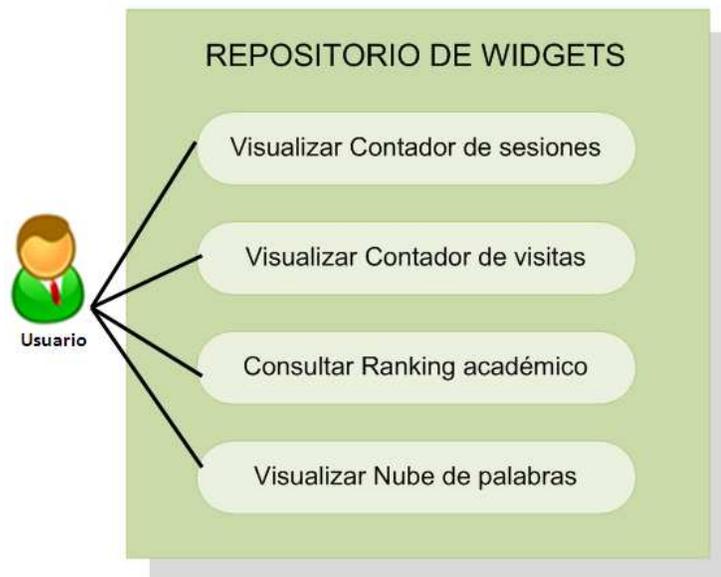


Figura 2. Diagrama de Casos de Uso

Luego de listar los Casos de Uso del Repositorio de Widgets, procederemos a describir cada uno junto con sus escenarios más importantes.

2.3.1 Caso de Uso Visualizar Contador de Sesiones

Caso de Uso 1	
Nombre	Visualizar Contador de Sesiones
Descripción	El usuario carga su página web o blog y el widget presenta como resultado cuantas veces el usuario ha visitado la página.
Actores	Usuario

Tabla III. Caso de Uso 1

Escenario 1.1	
Nombre	Conteo exitoso de visitas del usuario
Descripción	El usuario carga su página o blog que tiene instalado el Widget y se presenta el número de veces que él ha visitado la página.
Pre-condiciones	Ninguna
Post-condiciones	Widget muestra el número de veces que el usuario ha estado en dicha página

Tabla IV. Escenario 1.1

Escenario 1.2	
Nombre	Conteo no exitoso de visitas del usuario
Descripción	El usuario carga su página o blog que tiene instalado el Widget y no se presenta el número de veces que él ha visitado la página.
Pre-condiciones	Ninguna
Post-condiciones	Se presenta un mensaje de error indicando que no existe conexión con el servidor

Tabla V. Escenario 1.2

2.3.2 Caso de Uso Visualizar Contador de Visitas

Caso de Uso 2	
Nombre	Visualizar Contador de Visitas
Descripción	El usuario carga su sitio web o blog y el Widget presenta cuantas visitas ha tenido la página
Actores	Usuario

Tabla VI. Caso de Uso 2

Escenario 2.1	
Nombre	Conteo exitoso de visitas a la página
Descripción	El usuario carga su página o blog y se presenta el número de veces que el sitio o blog ha sido visitado por los usuarios.
Pre-condiciones	Ninguna
Post-condiciones	Widget muestra el número de visitas que ha tenido la página.

Tabla VII. Escenario 2.1

Escenario 2.2	
Nombre	Conteo no exitoso de visitas a la página
Descripción	El usuario carga su página o blog y no se presenta el número de veces que el sitio o blog ha sido visitado por los usuarios.
Pre-condiciones	Ninguna
Post-condiciones	Se presenta un mensaje de error indicando que no existe conexión con el servidor

Tabla VIII. Escenario 2.2

2.3.3 Caso de Uso Consultar Ranking Académico

Caso de Uso 3	
Nombre	Consultar Ranking Académico
Descripción	El usuario carga su sitio web o blog y el Widget presenta un cuadro de texto donde el usuario ingresa su número de matrícula para consultar las notas obtenidas en dicho semestre.
Actores	Usuario

Tabla IX. Caso de Uso 3

Escenario 3.1	
Nombre	Consulta exitosa del Ranking Académico
Descripción	El usuario desea visualizar sus materias registradas con sus respectivas notas. El Widget pide el ingreso del número de matrícula.
Pre-condiciones	Número de matrícula existe
Post-condiciones	Presenta el nombre de la materia e imágenes que identifican la nota obtenida de acuerdo a un rango ya establecido.

Tabla X. Escenario 3.1

Escenario 3.2	
Nombre	Consulta no exitosa del Ranking Académico
Descripción	El usuario desea visualizar sus materias registradas con sus respectivas notas y no ingresa número de matrícula o ingresa un número de matrícula erróneo
Pre-condiciones	No ingresa número de matrícula o matrícula no existe
Post-condiciones	Se muestra el siguiente mensaje: "Número de matrícula no existe".

Tabla XI. Escenario 3.2

Escenario 3.3	
Nombre	Consulta no exitosa debido a fallas técnicas
Descripción	El usuario desea visualizar sus materias registradas con sus respectivas notas. El Widget pide el ingreso del número de matrícula.
Pre-condiciones	Número de matrícula existe
Post-condiciones	Error en la conexión con la base de datos y se presenta un mensaje indicando que no se pudo establecer la conexión para la obtención de datos.

Tabla XII. Escenario 3.3

2.3.4 Caso de Uso Visualizar Nube de Palabras

Caso de Uso 4	
Nombre	Visualizar Nube de Palabras
Descripción	El usuario carga su sitio web o blog y el Widget presenta un cuadro de texto para que el usuario ingrese la ruta donde se encuentra su archivo tipo texto que desea cargar al widget.
Actores	Usuario

Tabla XIII. Caso de Uso 4

Escenario 4.1	
Nombre	Visualización exitosa de la Nube de Palabras
Descripción	El usuario carga un archivo tipo texto
Pre-condiciones	Archivos tipo texto
Post-condiciones	Se muestra cada palabra del archivo con diferente tamaño y color, dependiendo del número de veces que se repite la palabra.

Tabla XIV. Escenario 4.1

Escenario 4.2	
Nombre	Visualización no exitosa de la Nube de Palabras
Descripción	El usuario no carga un archivo tipo texto
Pre- condiciones	Archivo no tipo texto
Post- condiciones	Se muestra un mensaje de error indicando que no se ha podido subir el archivo.

Tabla XV. Escenario 4.2

2.4 Diagrama de Interacción de Objetos

Mediante este tipo de diagramas podemos visualizar la interacción de los objetos a través del tiempo en un contexto dado, que métodos son utilizados, los parámetros de envío, mensajes entre objetos, así como la respuesta de los mismos. [5]

A continuación mostramos los Diagramas de Interacción de Objeto de los escenarios más representativos.

2.4.1 Escenario 1.1: Conteo exitoso de visitas del usuario

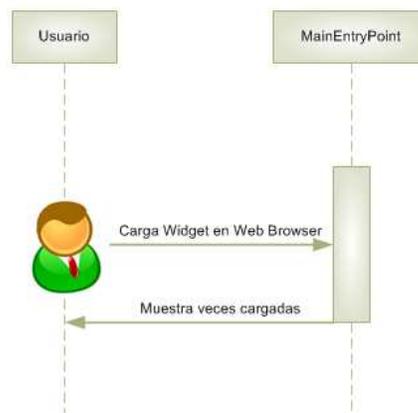


Figura 3. Diagrama Interacción Objeto Escenario 1.1

En la Figura 3 se describe el Diagrama Interacción Objeto del escenario conteo exitoso de visitas del usuario. El usuario carga la página donde se aloja el widget y al momento de ser cargado este aumenta el contador en uno.

2.4.2 Escenario 2.1: Conteo exitoso de visitas a la página

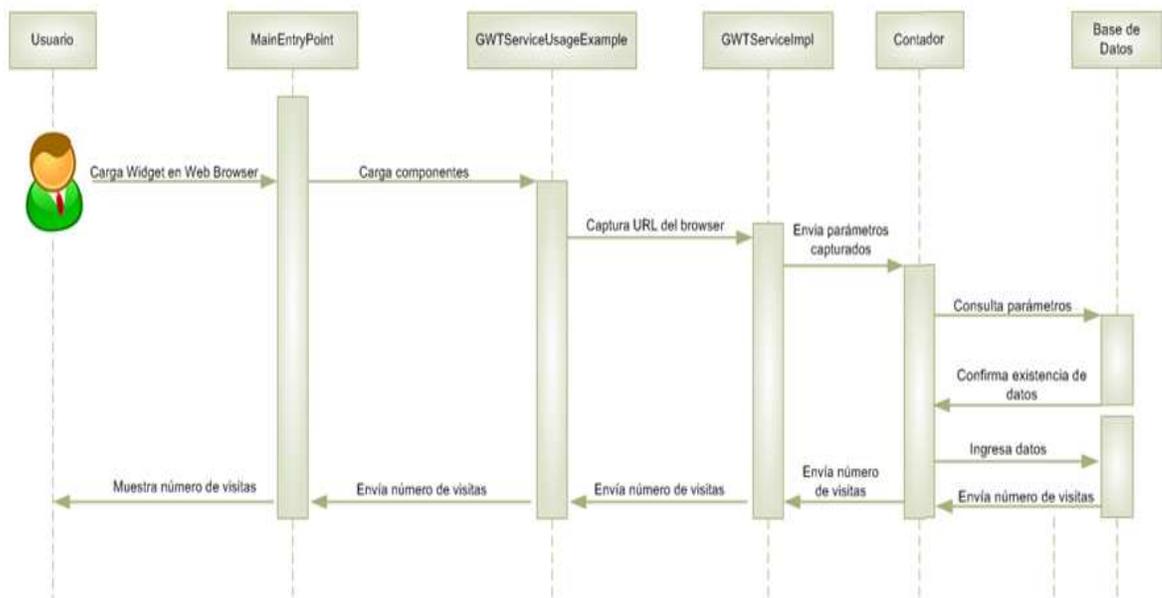


Figura 4. Diagrama Interacción Objeto Escenario 2.1

En la Figura 4, el usuario carga el widget al abrir la página en la que se encuentra alojado, disparando el método OnLoad que permite capturar el URL del blog o sitio web que contiene al widget, dicho método envía el URL al servidor y valida su existencia, si no existe lo crea caso contrario, aumenta el contador.

2.4.3 Escenario 3.1: Consulta exitosa del Ranking Académico

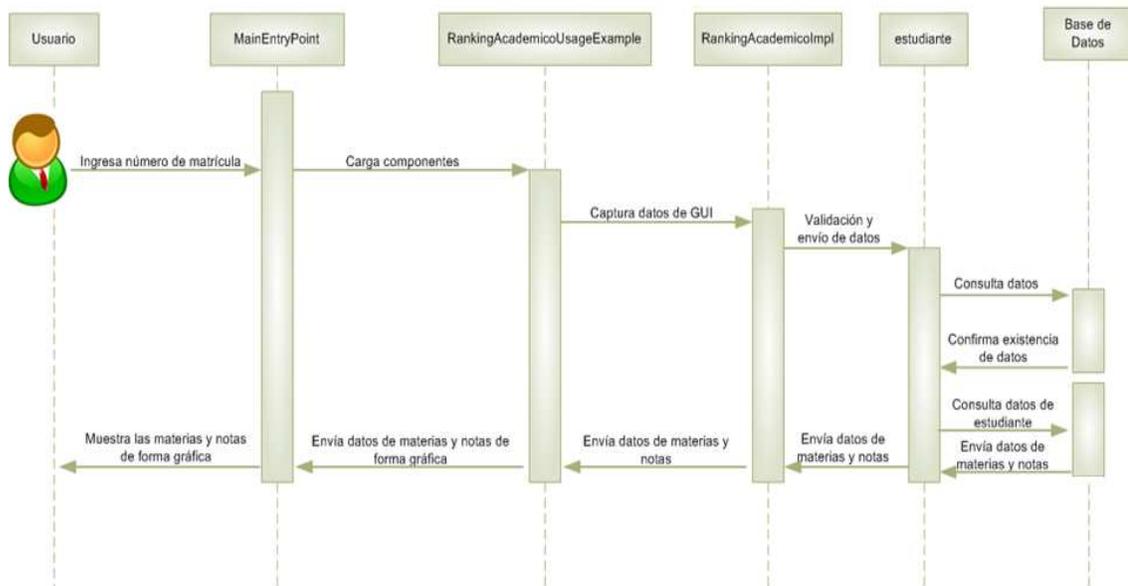


Figura 5. Diagrama Interacción Objeto Escenario 3.1

En la Figura 5, el usuario carga el widget al abrir el blog en el que se encuentra alojado, ingresa el número de matrícula a consultar, dichos datos son capturados y enviados al servidor, el mismo que valida la existencia de los datos y si la información es correcta, retorna los datos consultados caso contrario devuelve un mensaje de error.

2.4.4 Escenario 4.1: Visualización exitosa de la Nube de Palabras

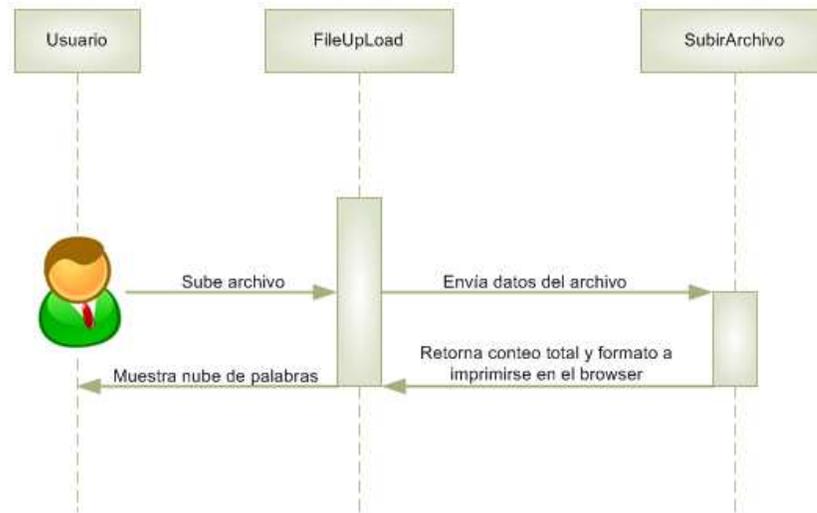


Figura 6. Diagrama Interacción Objeto Escenario 4.1

La Figura 6, detalla el Diagrama de Interacción del escenario visualización exitosa de la Nube de Palabras, que indica que el usuario ingresa al blog para visualizar el widget, ingresa la ruta del archivo del que se desea obtener la Nube de Palabras, el mismo es leído y analizado, contando el número de repeticiones de cada palabra dentro del archivo y clasificándolas por tamaño y color, estos son encapsulado en sentencias HTML y devueltas por medio de mensajes a la interfaz de usuario para de ahí poder ser visualizados.

Capítulo 3

Diseño de la Aplicación

3.1 Arquitectura del Repositorio de Widgets

El modelo de desarrollo de software utilizado para realizar el Repositorio de Widgets es el modelo de prototipos, debido a que el mismo como primera parte permite definir los objetivos globales del repositorio, para posteriormente identificar las necesidades. Una vez que estas se identifican, se puede determinar la iteración y construcción de los prototipos y de esta manera presentar un diseño rápido, el cual se enfoca en una representación de aquellos aspectos del software que serán visibles para el cliente o el usuario final. [6]

Para que el usuario interactúe adecuadamente con el repositorio, este se implementó como una aplicación web para permitirle el uso de cada uno de los widgets almacenados en el mismo y explicar de manera puntual, la forma de utilizarlos.

3.2 Estructura básica de una aplicación web usando GWT

A continuación se muestra la estructura básica de una aplicación web usando GWT:

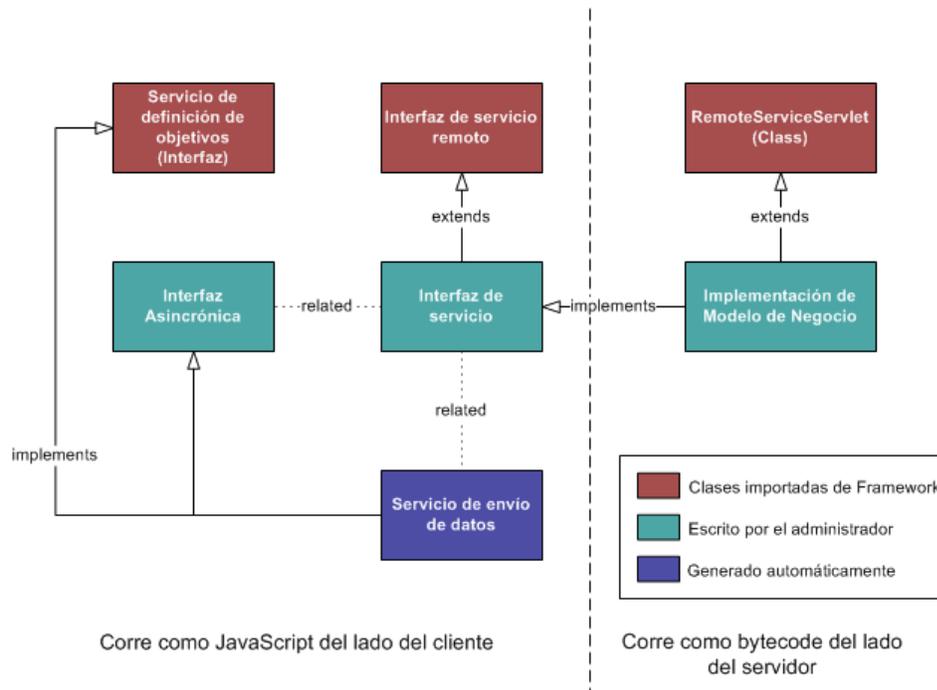


Figura 7. Estructura de aplicación web usando GWT. [7]

En la parte superior del gráfico se encuentran las interfaces y clases importadas de la herramienta, para la generación adecuada del widgets. En la parte central se encuentran las clases e interfaces implementadas por nosotros, de izquierda a derecha tenemos:

1.- Interfaz de servicios asincrónicos.- La cual nos permite describir los métodos a implementar por la clase del servidor.

2.- La clase principal o GUI de usuario.- Esta nos permite colocar los elementos visuales del widget así como la invocación a métodos para transferir información al servidor.

3.- La clase del lado del servidor.- La cual maneja la información suministrada por el lado del cliente y entregado resultados.

3.3 Diseño de la Interfaz de widgets

La herramienta GWT permite realizar pequeñas aplicaciones mediante el uso de las siguientes clases:

La primera clase debe contener el método `onModuleLoad()` dicho método permite contener paneles, y otras características básicas de un widget como pueden ser color, fondo, texto, así como llamada a otros métodos y creación de objetos.

La segunda clase permite la implementación de los métodos establecidos en la interfaz, dichos métodos son los que permiten la comunicación entre cliente-servidor y los cuales contiene embebido la comunicación RPC.

La interfaz como tal contiene la descripción de los métodos a ser implementados dentro de las clases del servidor, los mismos pueden ser asincrónicos o sincrónicos.

Todos los widgets alojados en el servidor utilizan métodos asincrónicos que dependen de la interacción del usuario para dar una respuesta en la interfaz.

3.3.1 Interfaz del Repositorio de Widgets

En esta sección se describe las pantallas más relevantes en el diseño tanto del repositorio como los widgets.

Interfaz Repositorio
Descripción
Sirve como mecanismo de acceso a los widgets desarrollados, este explica y contiene el código necesario para que el usuario pueda utilizarlos de manera adecuada.
Imagen
 <p>The screenshot displays the 'Repositorio de Widgets' website. At the top, there is a navigation bar with links for 'Inicio', 'Twitter', 'Mail', 'Blog', and 'IPTV'. The main content area is titled 'Repositorio de Widgets' and includes a globe icon. Below this, there are several sections: '¿Qué es un Widget?', '¿Cómo subir un Widget a mi sitio?', and 'Visualizar'. The '¿Qué es un Widget?' section explains that widgets are small applications that can be used on a website or desktop. The '¿Cómo subir un Widget a mi sitio?' section provides instructions on how to add a widget to a website, including steps like 'Seleccionar' (Select), 'Copiar y Pegar' (Copy and Paste), and 'Visualizar' (View). The 'Visualizar' section shows a preview of a widget on a website. At the bottom of the page, there is a footer with copyright information: 'Foto - Twitter y Condiciones - Recursos - Contacto Copyright © 2008. All Rights Reserved. Design by Denise & Trina'.</p>

Figura 8. Repositorio de Widgets

Tabla XVI. Interfaz del Repositorio de Widgets

3.3.2 Interfaz del Contador de Sesiones

Interfaz Contador de Sesiones	
Descripción	
	Esta interfaz permite al usuario conocer de manera visual cuántas veces él ha accedido a su página o blog, mediante la implementación de un cookie que guarda dicho registro, puesto que no es necesario tener un historial en el servidor del número de visitas realizadas por él. El cookie tiene un tiempo de expiración de un año para poder realizar estadísticas veraces del sitio.
Imagen	
	 The image shows a small rectangular window titled "Contador de sesiones". Inside the window, there is a 3D cartoon turtle character with a yellow body and green limbs, standing and waving. Below the turtle, the number "29" is displayed in a green, digital-style font.
	Figura 9. Contador de Sesiones

Tabla XVII. Interfaz Contador de Sesiones

3.3.3 Interfaz del Contador de Visitas

Interfaz Contador de Visitas	
Descripción	
	Esta interfaz permite al usuario conocer de manera visual cuántas veces han visitado su página o blog, el mismo se desarrolló utilizando métodos asincrónicos de envío de datos y almacenamiento de la información en una pequeña base de datos debido a que se debe mantener un registro permanente de las visitas realizadas al o los sitios.
Imagen	
	 <p>Visitante: 35</p>

Figura 10. Contador de Visitas

Tabla XVIII. Interfaz Contador de Visitas

La diferencia entre los dos contadores antes mencionados es que el de sesiones fue implementado mediante cookies y permite conocer cuantas veces el autor del blog o página web ha accedido al mismo, mientras el contador de visitas nos permite saber el número general de veces que el sitio o blog ha sido visitado, si restamos el valor obtenido por el contador de sesiones con el valor obtenido por el contador de visitas tendremos el número real de visitas realizadas al sitio no hechas por el autor del mismo.

3.3.4 Interfaz del Ranking Académico

Interfaz Ranking Académico	
Descripción	<p>Esta interfaz permite al usuario tener una idea general de las calificaciones obtenidas durante el semestre de forma visual con tan solo visitar su blog, el mismo se desarrolló utilizando los métodos RPC encapsulados en la herramienta para realizar la transferencia de información entre el cliente y el servidor para así ser categorizada y mostrar al estudiante.</p> <p>Debemos tener en consideración que la utilización de métodos RPC se realiza cuando se necesita realizar la transferencia de la información y llamadas a métodos alojados en el servidor.</p> <p>Este widget posee una pequeña base de datos de prueba creada en MySQL.</p>
Imagen	 <p>The image shows a web interface for an academic ranking system. At the top, there is a text input field containing the number '200109399' and a button labeled 'Send to server'. Below this, there is a list of subjects with corresponding smiley face icons indicating their status or ranking:</p> <ul style="list-style-type: none">Ecología: Happy face (😊)Lenguajes de programación: Neutral face (😐)Cálculo I: Neutral face (😐)Estadística: Sad face (😞)

Figura 11. Ranking Académico

Tabla XIX. Interfaz Ranking Académico

3.3.5 Interfaz de la Nube de Palabras

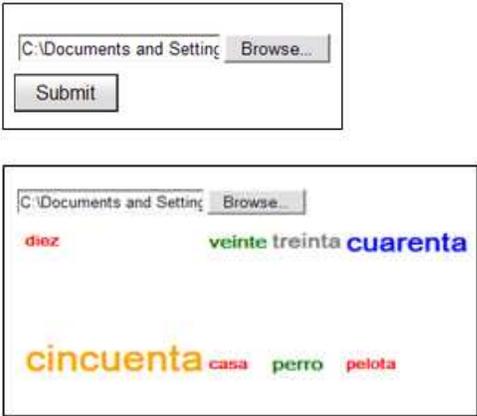
Interfaz Nube de Palabras	
Descripción	
	Esta interfaz permite ver de forma visual una nube de palabras generada de un archivo de texto ingresado por el usuario, esto es realizado en el lado del servidor y de vuelta al GUI.
Imagen	 <p>The image shows two screenshots of a web application interface. The top screenshot displays a file selection dialog with a text input field containing 'C:\Documents and Settings\ Browse...' and a 'Submit' button. The bottom screenshot shows the resulting word cloud, with words of varying sizes and colors: 'diez' (red), 'veinte treinta cuarenta' (green and blue), and 'cincuenta casa perro pelota' (yellow, red, green, and red).</p>

Figura 12. Nube de Palabras

Tabla XX. Interfaz Nube de Palabras

Capítulo 4

Implementación de la Aplicación

4.1 Introducción

La implementación del proyecto se divide en dos partes, una implementación para el contenedor de los Widgets y otra implementación para los Widgets.

La función del contenedor de los Widgets es instruir en breves pasos, al usuario, sobre cómo cargar los Widgets y a su vez mostrar la imagen y código del Widget a cargar en el sitio web del usuario.

En el caso de los Widgets, dependiendo de cual escoja el usuario su función es mostrar información de visitas, notas del estudiante y número de repeticiones de las palabras de un archivo texto cargado por el usuario.

4.2 Requerimientos de Hardware

Para poder crear y cargar los Widgets, es necesario trabajar en una máquina que tenga las siguientes características:

- Procesador: Intel Core 2 duo
- Memoria RAM: Mínimo 2Gb
- Disco Duro: 120Gb

- Sistema Operativo: Windows XP / Windows Vista / Windows 7.

4.3 Requerimiento de Software

Para la creación del Repositorio y de los Widgets se requiere:

- Instalar IDE NetBeans 6.0 o Eclipse (Galileo)
- Instalar Framework Google Web Toolkit

Para poder usar los Widgets es necesario que el usuario tenga:

- Creado un blog o sitio web.
- Acceso a Internet

4.4 Detalle de la implementación

En la realización de los widgets web se tuvieron 2 posibilidades de implementación las cuales fueron el widgets bajo la perspectiva de Yahoo o el desarrollo de los mismos bajo la perspectiva de Google. Se optó por la realización de los mismos usando el Framework de Google GWT, ya que permite desarrollarlos con el lenguaje Java y posteriormente ser publicados en la web, mientras que con Yahoo nos veíamos obligados a realizarlos en sus servidores, sin mucha flexibilidad.

Dentro del proceso de implementación se utilizaron varias herramientas, para el desarrollo de los widgets, comenzando con el IDE NetBeans 6.5.1, debido a la experiencia que se tenía en dicho IDE. Posteriormente

se utilizó Eclipse Galileo ya que surgieron varios inconvenientes con la transferencia y análisis de archivos.

Para el almacenamiento de información utilizamos el motor de base de datos MySQL, ya que es libre bajo cierto número de restricciones como la cantidad de tablas que el mismo maneja.

4.4.1 Herramientas de desarrollo

Para la creación del Repositorio de widgets se utilizó una plantilla CSS, la misma que se modificó acorde a lo que se va a presentar al usuario. Para desarrollar los Widgets se hizo uso de herramientas Open Source que se listan a continuación:

- IDE NetBeans 6.5.1
- Lenguaje de programación Java
- Google Web Toolkit
- Eclipse (Galileo)
- Base de datos MySQL

4.4.2 Problemas en la implementación

Una de las mayores dificultades que se presentó en la implementación de las aplicaciones, fue con el widget Nube de palabras. El Plugin para NetBeans, no permitía la transferencia del archivo y como solución, se tuvo que desarrollar esta aplicación utilizando otro IDE (Eclipse).

Otro inconveniente, fue al momento de cargar los widgets en un blog de la ESPOL creado para hacer pruebas de funcionalidad previa a las pruebas con los estudiantes. El blog de la ESPOL tenía bloqueados los permisos para el uso del tag script lo que ocasionaba que al momento de guardar el código para cargar el Widget, se borrara parte del mismo. Por lo que se pidió ayuda al Administrador del Blog para que habilite los permisos para el uso del tag script. Una vez habilitados los permisos se procedió nuevamente con las pruebas teniendo como resultado poder utilizar el tag mencionado.

Capítulo 5

Pruebas y Resultados

5.1 Introducción

En este capítulo se detallan las pruebas y los resultados obtenidos de las encuestas realizados a un grupo de estudiantes.

5.2 Pruebas

Como se indicó en el capítulo anterior, previo a las pruebas se necesitó ayuda del Ing. José Rodríguez, Administrador del blog de la ESPOL, puesto que el blog tiene restricciones en cuanto al uso del tag script y tag iframe, las cuales sólo pueden ser habilitados por el administrador.

Una vez habilitados los permisos en el blog y habiendo hecho los cambios necesarios, se procedió a las pruebas en la Facultad de Ingeniería Mecánica y Ciencias de la Producción bajo la tutela del Ing. Juan Carlos Delgado quien solicitó a los estudiantes que lleven sus máquinas teniendo un total de 10 computadoras disponibles para las pruebas. Se les explicó a los estudiantes qué es un widget, el uso de cada uno de ellos y cómo cargarlos. Las pruebas duraron alrededor de

30 minutos y se necesitó de 5 minutos más para responder la encuesta.

La encuesta se encuentra en el ANEXO A, sección A.1

Para las pruebas los estudiantes realizaron los siguientes pasos:

1. Ingresar a <http://hobbes:8080/ServerWidget/>
2. Escoger el widget a cargar en el blog
3. Copiar el código del widget escogido

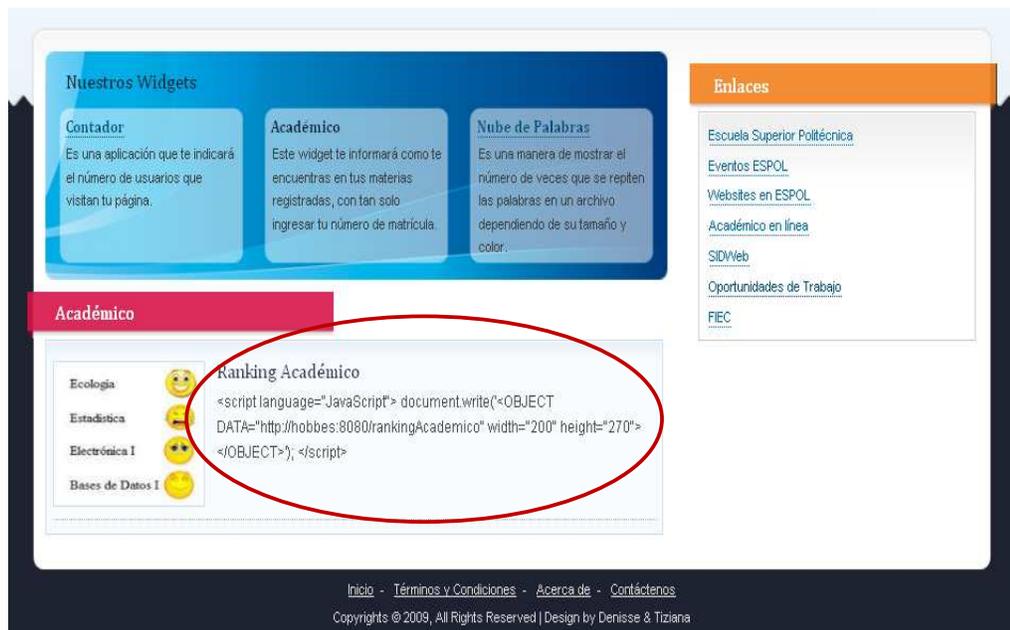


Figura 13. Copia de código del widget

4. Ingresar al administrador del blog, en el menú del lado izquierdo dar clic en Apariencia, opción Widgets.



Figura 14. Ventana de administrador de blog

5. De los Widgets que están disponibles, seleccionamos el de texto y lo arrastramos a la columna de la derecha para activarlo dentro del Blog.



Figura 15. Cargar Widget de texto

6. En la ventana de texto, se debe poner un título al Widget y pegar el código del Widget seleccionado en el punto 3 y guardamos.

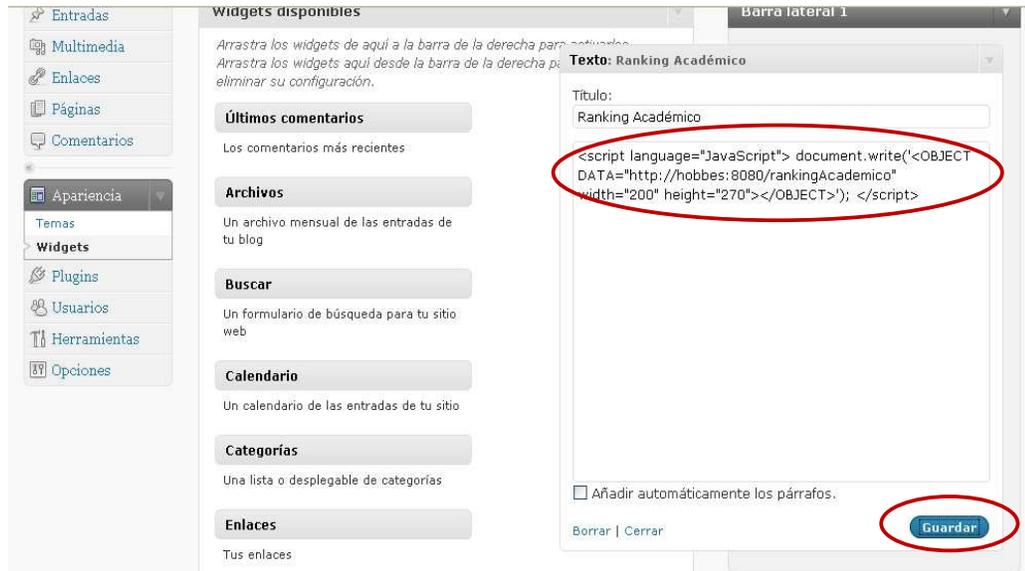


Figura 16. Pegar código de widget

7. Finalmente cargamos el blog y podemos visualizar el widget.

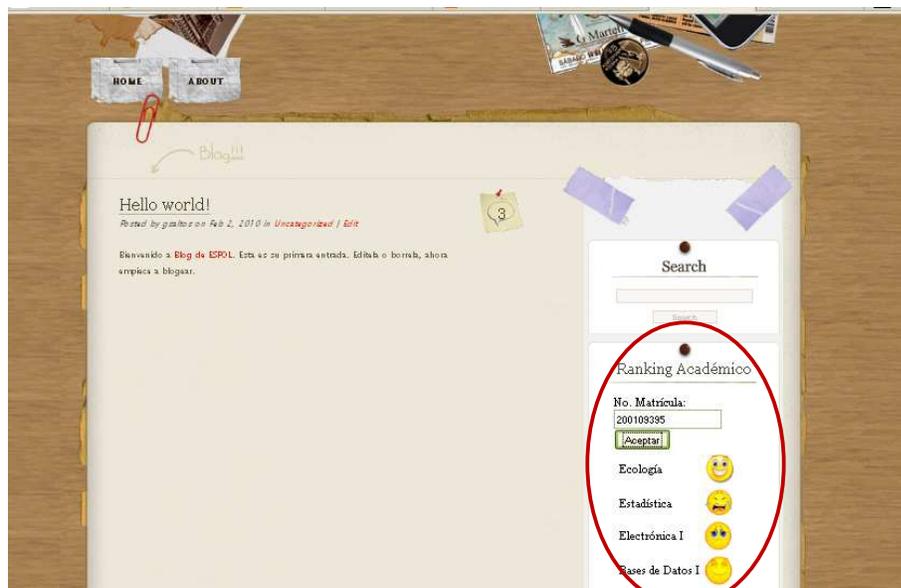


Figura 17. Blog con widget Ranking Académico

En el caso del widget Ranking Académico, utilizamos una base de prueba, motivo por el cual se les dio a los estudiantes dos números de matrículas para ser ingresadas en el widget y comprobar su funcionalidad.

5.3 Resultados

Según la encuesta realizada el día viernes 3 septiembre del año en curso a un grupo heterogéneo de 32 estudiantes de las diferentes Facultades de la ESPOL, podemos concluir que el 72% de los estudiantes aseguran que sí utilizarían los widgets en su blog.

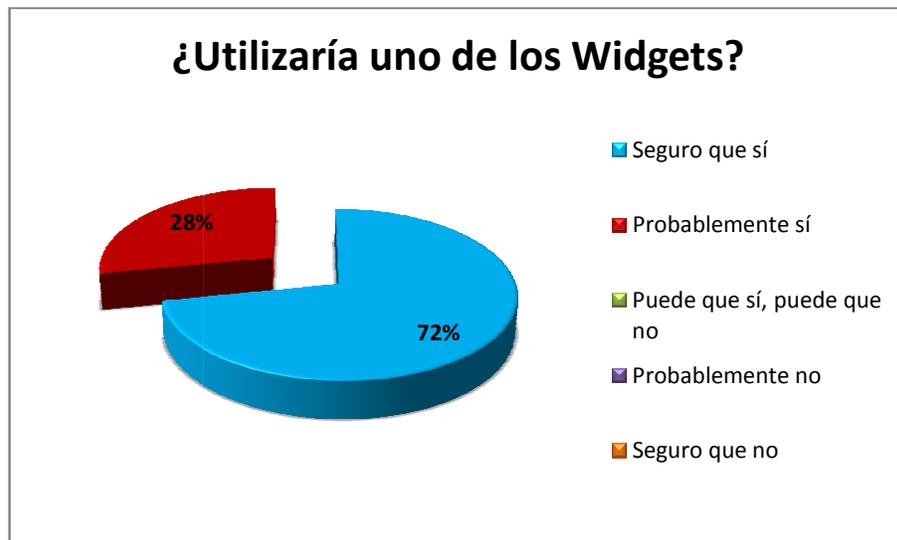


Figura 18. Estadística de uso del widget

Según la encuesta, al 100% de los encuestados les llamó más la atención el Widget Académico y el 18.75% también votó por el Widget Contador de Visitas.

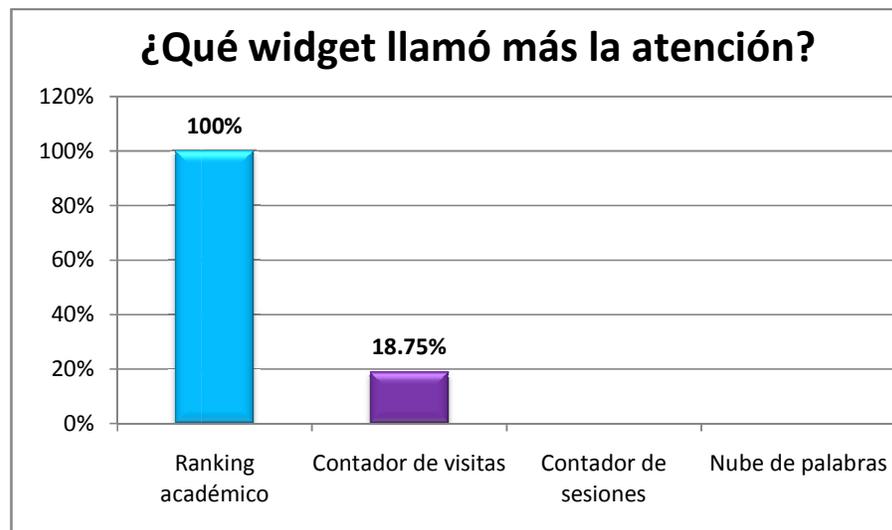


Figura 19. Estadística de cuál llamó más la atención

Otro resultado reflejado por esta encuesta es que en promedio, el 78% lo considera muy fácil de usar.

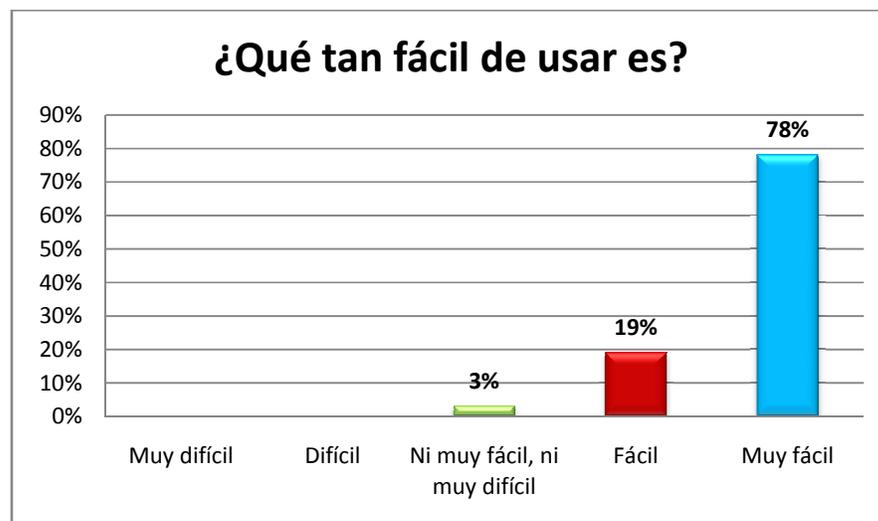


Figura 20. Estadística de facilidad de uso

El widget Nube de palabras fue el de mayor votos en lo que respecta a cambios en tipo de letra (21.8%), color de letra (16.6%), Imagen (12.5%) y color de fondo (12.5%). Cabe recalcar que la suma de los porcentajes no da un 100% porque las respuestas fueron de selección múltiple.

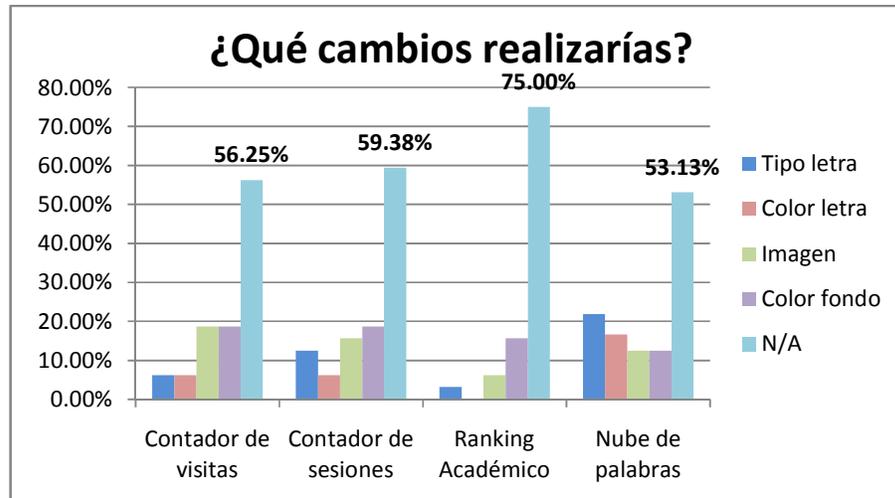


Figura 21. Estadística de mejoras del widget

El resultado más expresivo y alentador de esta encuesta es que el 100% de los estudiantes encuestados recomendarían el uso de los Widgets en los blogs.



Figura 22. Estadística de recomendación

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

De la experiencia adquirida en el desarrollo de aplicaciones web utilizando el Framework GWT podemos concluir que:

1. Con GWT podemos generar aplicaciones web de manera fácil, rápida y sencilla evitando así todo el proceso de programación en capas y permitiendo la creación de este tipo de aplicaciones en un solo lenguaje como es Java.
2. Para crear aplicaciones que permitan cargar, leer y analizar archivos, es mucho mejor trabajar con Eclipse ya que el Framework realiza dichas operaciones de manera exitosa sobre el mismo.
3. La ejecución de una aplicación en el compilador de GWT es más rápida que la ejecución de la misma aplicación en un browser, ahorrando tiempo de depuración en el desarrollo e implementación.

RECOMENDACIONES

1. Una manera de difundir información sobre charlas, seminarios, oportunidades de trabajo, etc., se lo podría lograr a través de la creación de una red de publicidad con widgets.
2. Para incrementar el uso de estas aplicaciones, se recomienda implementar widgets que permitan al estudiante tener información de su interés de forma rápida.

ANEXO

ANEXO A

A.1 Encuesta a los estudiantes

Repositorio de Widgets

Valoración de los Widgets

CARRERA:

Dedique unos minutos a completar esta pequeña encuesta.

Sus respuestas serán tratadas de forma confidencial y serán utilizadas únicamente para mejorar el servicio que le proporcionamos.

Esta encuesta dura aproximadamente cinco minutos.

1. ¿Utilizaría uno de los widgets en su blog?

- Seguro que sí
- Probablemente sí
- Puede que sí, puede que no
- Probablemente no
- Seguro que no

2. ¿Cuál de los widgets te llamó más la atención?

- Widget Contador de Visitas
- Widget Contador de Sesiones
- Widget Ranking Académico
- Widget Nube de Palabras

3. Por favor, indiquenos que tan fácil de usar ve usted al widget en una escala del 1 al 5, donde 1 es muy difícil y 5 es muy fácil

	1	2	3	4	5
Widget Contador de Visitas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Widget Contador de Sesiones	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Widget Ranking Académico	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Widget Nube de Palabras	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

3. Por favor, indiquenos que tan fácil de usar ve usted al widget en una escala del 1 al 5, donde 1 es muy difícil y 5 es muy fácil

	1	2	3	4	5
Widget Contador de Visitas	<input type="radio"/>				
Widget Contador de Sesiones	<input type="radio"/>				
Widget Ranking Académico	<input type="radio"/>				
Widget Nube de Palabras	<input type="radio"/>				

4. Por favor, indicanos que cambiarías de los widgets

	Tipo de letra	Color de letra	Imagen	Color de fondo	N/A
Widget Contador de Visitas	<input type="checkbox"/>				
Widget Contador de Sesiones	<input type="checkbox"/>				
Widget Ranking Académico	<input type="checkbox"/>				
Widget Nube de Palabras	<input type="checkbox"/>				

5. ¿Recomendaría usted el uso de los widgets a otras personas?

- Sí
- No

BIBLIOGRAFIA

1. **Wikipedia**, Web 2.0, http://es.wikipedia.org/wiki/Web_2.0, 5 de oct 2010
2. **Microsoft**, ¿Qué son los Widgets y cuáles son sus oportunidades de negocio?, <http://www.microsoft.com/business/smb/es-es/internet/widgets.msp>, 12 julio 2010
3. **Google web Toolkit**, La arquitectura de Google Web Toolkit, <http://esgooglewebtoolkit.blogspot.com/2007/09/la-arquitectura-de-google-web-toolkit.html>, 10 dic de 2010
4. **Epidata Consulting**, Bloque de Conocimiento de GWT, http://www.epidataconsulting.com/tikiwiki/tiki-index.php?page=GWT#Qu_es_un_m_dulo_, 14 de ene 2010
5. **Joaquín**, UML - Diagramas de interacción, <http://jms32.eresmas.net/tacticos/UML/UML06/UML0601.html>, 18 de 09 de 2003
6. **Wikipedia**, Modelo de prototipos, http://es.wikipedia.org/wiki/Modelo_de_prototipos, 1 de agosto 2010
7. **Google Web Toolkit**, Anatomía de RPC, <http://esgooglewebtoolkit.blogspot.com/2007/09/anatoma-de-rpc.html>, 6 de nov 2009