|  |
| --- |
| Servidor Web  **Mediante Sockets** |
| Miguel González Gómez |
| Madrid 31 de Mayo de 2013 |

# INDICE

[INDICE 1](#_Toc357764578)

[Descripción del proyecto 2](#_Toc357764579)

[Características de la aplicación 2](#_Toc357764580)

[Interfaces de la aplicación 3](#_Toc357764581)

[Interfaz de la aplicación por primera vez 3](#_Toc357764582)

[Ejemplo de creación de un servidor con un servicio 3](#_Toc357764583)

[Interfaz de la aplicación con un servidor web 4](#_Toc357764584)

[Interfaz de la aplicación para acceder a las opciones de un servidor 4](#_Toc357764585)

[Instalación de los demos 5](#_Toc357764586)

[Guardado 5](#_Toc357764587)

[Servidor estático 6](#_Toc357764588)

[Servidor con servicio web 7](#_Toc357764589)

# Descripción del proyecto

El proyecto consiste en crear un Servidor Web utilizando Sockets en Java.

Además debe ser capaz de procesar peticiones POST y GET, formularios y Cookies.

# Características de la aplicación

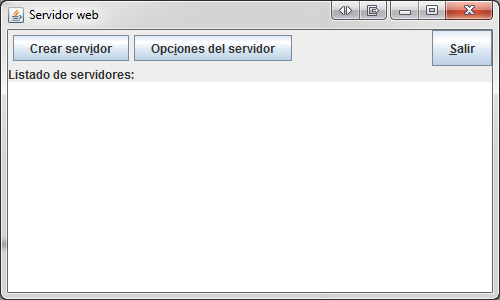
Se ha tomado como guía el servidor Tomcat, es decir, el servidor permite servir páginas estáticas o permite a través de una interfaz implementar una serie de servicios para añadir funcionalidades (acceso base de datos, gestión de cookies, etc.).

Además, a diferencia de cualquier servidor web, se ha implementado una interfaz para la administración del servidor utlizando un patrón MVC, con las siguientes funcionalidades:

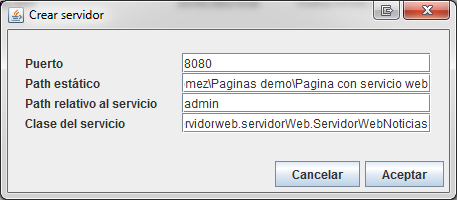
* Configurar el servidor web
  + Número del pool de sockets
  + Tiempo del KeepAlive para mantener conexiones abiertas
* Crear un nuevo servidor
  + Estático
  + Con un servicio indicando su clase java y el path donde actua el servicio
* Editar un servidor existente
* Borrar un servidor existente
* Iniciar un servidor
* Parar un servidor

# Interfaces de la aplicación

### Interfaz de la aplicación por primera vez



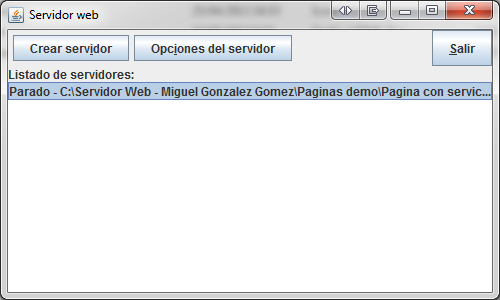
### Ejemplo de creación de un servidor con un servicio



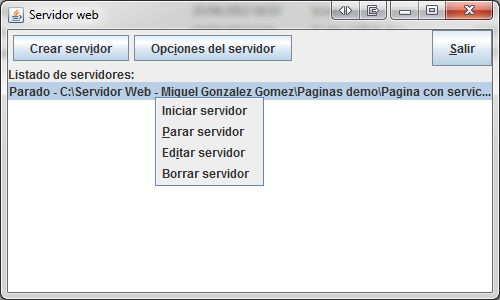
El path relativo al servicio significa que al intentar acceder a un recurso en esa ruta se gestionará por el servicio. Por defecto en la aplicación viene un servicio muy básico llamado ServidorWebNoticias cuya clase se encuentra en:

*servidorweb.servidorWeb.ServidorWebNoticias*

### Interfaz de la aplicación con un servidor web



### Interfaz de la aplicación para acceder a las opciones de un servidor



# Instalación de los demos

En el proyecto existe una carpeta con dos demos:

1. Pagina con servicio web
2. Página estática

Para hacerlos funcionar hay que crear dos servidores, en el ejemplo voy a partir que la carpeta del proyecto está en *C:\*.

Servidor con servicio web

Puerto

8080

Path estático

C:\Servidor Web - Miguel Gonzalez Gomez\Paginas demo\Pagina con servicio web

Path relativo al servicio

admin

Clase del servicio

servidorweb.servidorWeb.ServidorWebNoticias

Servidor estático

Puerto

9090

Path estático

C:\Servidor Web - Miguel Gonzalez Gomez\Paginas demo\Pagina estática

Path relativo al servicio

-- Vacío --

Clase del servicio

-- Vacío --

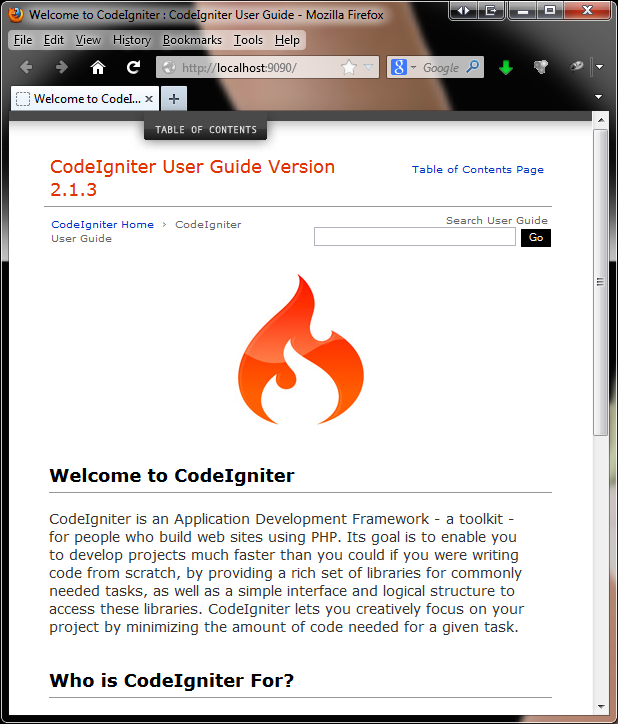
### Guardado

Al guardar los cambios se crea un fichero “servidorWeb.xml” con las opciones. Se puede configurar el fichero con un editor en modo texto o mediante la aplicación.

# Servidor estático

El servidor estático en la demo es la documentación del Framework PHP Code Igniter. Una vez iniciado se navega a través de él con la url:

*http://localhost:9090*



# Servidor con servicio web

Se ha desarrollado una pequeña aplicación web con contenido estático y una zona de administración, en la cual, para acceder hay que reyenar un formulario. Los datos de acceso son:

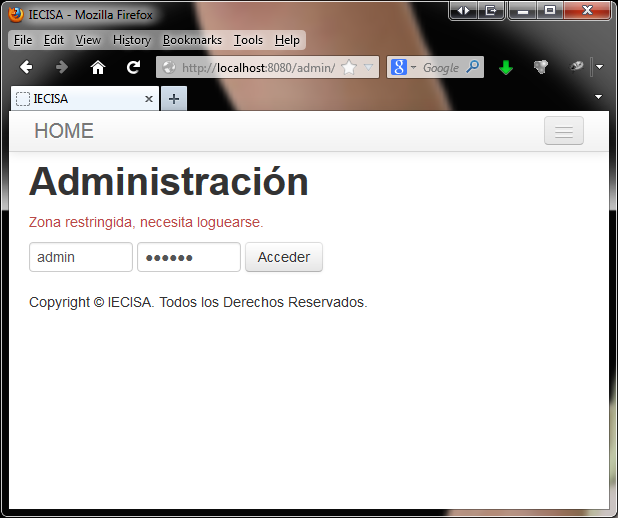
Usuario: *admin*

Password: *entrar*

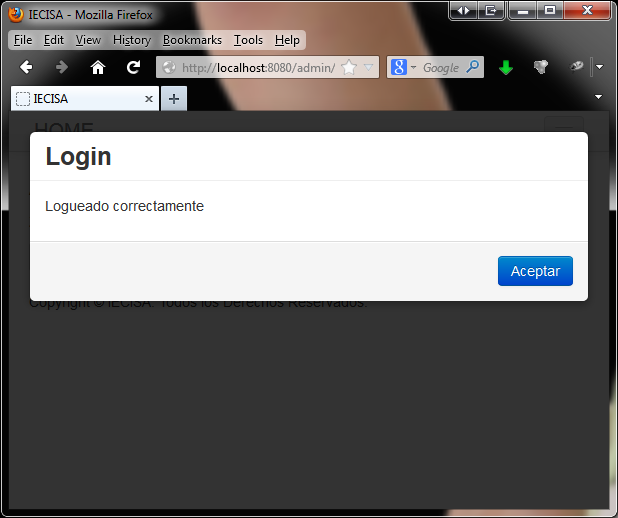
Home



Administración



Al clickar en Login con los datos correctos



Zona de administración controlada con una Cookie

