

**MEMORIA FINAL DE PROYECTO**

**COMUNIDAD STARCRAFT**



**CICLO FORMATIVO DE GRADO SUPERIOR**

**DESARROLLO DE APLICACIONES WEB**

**AUTORES**

**Cristian Rivera González**

**Daniel González Peci**

**Adrián Conde Romero**

**TUTOR**

**Ana Isabel Arévalo Rosingana**

## **Licencia**

**Esta obra está bajo una licencia Reconocimiento-Compartir bajo la misma licencia 3.0 España de Creative Commons. Para ver una copia de esta licencia, visite**

**<http://creativecommons.org/licenses/by-sa/3.0/es/> o envíe una carta a Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.**

# INDICE

INTRODUCCIÓN.....	8
ALCANCE DEL PROYECTO Y ANÁLISIS PREVIO.....	9
1 ESTUDIO DE VIABILIDAD.....	10
1.1 ESTADO ACTUAL DEL SISTEMA.....	10
1.1 RESUMEN DE REQUISITOS DEL CLIENTE .....	10
1.1 POSIBLES SOLUCIONES.....	10
1.1 SOLUCIÓN ELEGIDA.....	10
1.1 PLANIFICACIÓN TEMPORAL DE LAS TAREAS DEL PROYECTO COMUNIDAD STARCRAFT.....	11
1.1 PLANIFICACIÓN DE LOS RECURSOS A UTILIZAR .....	11
1 ANÁLISIS.....	12
1.1 DIAGRAMA DE CASOS DE USO.....	12
1.2 MODELO DE DATOS.....	13
1.1 REQUISITOS FUNCIONALES.....	14
1.2 REQUISITOS NO FUNCIONALES.....	14
1 DISEÑO.....	15
1.1 ESTRUCTURA DE LA APLICACIÓN.....	15
1.1 COMPONENTES DEL SISTEMA / ARQUITECTURA DE RED.....	15
1.1 HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS.....	15
1 IMPLEMENTACIÓN.....	17
1.1 IMPLEMENTACIÓN DEL MODELO DE DATOS.....	17
1.1 CARGA DE DATOS.....	18
1.1 CONFIGURACIONES REALIZADAS EN EL SISTEMA.....	18
1.1 IMPLEMENTACIONES DE CÓDIGO REALIZADAS.....	18
1 PRUEBAS.....	19
1.1 CASOS DE PRUEBAS.....	19
1 EXPLOTACIÓN.....	21
1.1 PLANIFICACIÓN.....	21
1.1 PREPARACIÓN PARA EL CAMBIO.....	21
1.1 MANUAL DE USUARIO.....	21
1.1 IMPLANTACIÓN PROPIAMENTE DICHA.....	22
CONTROL DE VERSIONES.....	24

<b>1 FUENTES.....</b>	<b>26</b>
<b>1.1 LEGISLACIÓN.....</b>	<b>26</b>
<b>1.1 BIBLIOGRAFÍA.....</b>	<b>26</b>
<b>2 ANEXOS.....</b>	<b>27</b>

# INDICE de tablas e ilustraciones

## Base de datos: `starcraft`

### Estructura de tabla para la tabla `categoria`

```
CREATE TABLE IF NOT EXISTS `categoria` (  
  `idCategoria` int(11) NOT NULL AUTO_INCREMENT,  
  `nombre` varchar(40) COLLATE utf8_spanish_ci NOT NULL,  
  PRIMARY KEY (`idCategoria`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci  
AUTO_INCREMENT=8 ;
```

### Estructura de tabla para la tabla `datos\_usuario`

```
CREATE TABLE IF NOT EXISTS `datos_usuario` (  
  `idUsuario` int(11) NOT NULL,  
  `password` varchar(16) COLLATE utf8_spanish_ci NOT NULL,  
  `nombre` varchar(50) COLLATE utf8_spanish_ci NOT NULL,  
  `correo` varchar(40) COLLATE utf8_spanish_ci NOT NULL,  
  `direccion` varchar(100) COLLATE utf8_spanish_ci NOT NULL,  
  `cod_postal` int(5) NOT NULL,  
  `sexo` varchar(1) COLLATE utf8_spanish_ci NOT NULL,  
  `f_alta` date NOT NULL,  
  `f_nacimiento` date NOT NULL,  
  `imagenPerfil` varchar(100) COLLATE utf8_spanish_ci NOT NULL,  
  PRIMARY KEY (`idUsuario`),  
  UNIQUE KEY `correo` (`correo`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;
```

### Estructura de tabla para la tabla `noticias`

```
CREATE TABLE IF NOT EXISTS `noticias` (  
  `Titulo` varchar(50) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT NULL,  
  `Contenido` mediumtext CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT  
  NULL,  
  `imagen` varchar(1000) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT  
  NULL,  
  `id` int(255) NOT NULL AUTO_INCREMENT,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

### **Estructura de tabla para la tabla `respuesta`**

```
CREATE TABLE IF NOT EXISTS `respuesta` (  
  `idRespuesta` int(11) NOT NULL AUTO_INCREMENT,  
  `texto` longtext COLLATE utf8_spanish_ci NOT NULL,  
  `idUsuario` int(11) NOT NULL,  
  `idTema` int(11) NOT NULL,  
  `f_creacion` datetime NOT NULL,  
  PRIMARY KEY (`idRespuesta`),  
  KEY `idUsuario` (`idUsuario`,`idTema`),  
  KEY `idTema` (`idTema`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci  
AUTO_INCREMENT=22 ;
```

### **Estructura de tabla para la tabla `tema`**

```
CREATE TABLE IF NOT EXISTS `tema` (  
  `idTema` int(11) NOT NULL AUTO_INCREMENT,  
  `idCategoria` int(11) NOT NULL,  
  `nombre` varchar(50) COLLATE utf8_spanish_ci NOT NULL,  
  `texto` longtext COLLATE utf8_spanish_ci NOT NULL,  
  `f_creacion` datetime NOT NULL,  
  `usuario` int(11) NOT NULL,  
  `cerrado` tinyint(1) NOT NULL DEFAULT '0',  
  PRIMARY KEY (`idTema`),  
  KEY `usuario` (`usuario`),  
  KEY `idCategoria` (`idCategoria`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci  
AUTO_INCREMENT=11 ;
```

### **Estructura de tabla para la tabla `usuario`**

```
CREATE TABLE IF NOT EXISTS `usuario` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nickname` varchar(15) COLLATE utf8_spanish_ci NOT NULL,  
  `rol` varchar(20) COLLATE utf8_spanish_ci NOT NULL DEFAULT 'Miembro',  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `nickname` (`nickname`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci  
AUTO_INCREMENT=3 ;
```

### **Restricciones creadas en las tablas**

```
ALTER TABLE `datos_usuario`  
  ADD CONSTRAINT `datos_usuario_ibfk_1` FOREIGN KEY (`idUserio`) REFERENCES `usuario` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;
```

```
ALTER TABLE `respuesta`  
  ADD CONSTRAINT `respuesta_ibfk_1` FOREIGN KEY (`idUserio`) REFERENCES `usuario` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,  
  ADD CONSTRAINT `respuesta_ibfk_2` FOREIGN KEY (`idTema`) REFERENCES `tema` (`idTema`) ON DELETE CASCADE ON UPDATE CASCADE;
```

```
ALTER TABLE `tema`  
  ADD CONSTRAINT `tema_ibfk_2` FOREIGN KEY (`usuario`) REFERENCES `usuario` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,  
  ADD CONSTRAINT `tema_ibfk_3` FOREIGN KEY (`idCategoria`) REFERENCES `categoria` (`idCategoria`) ON DELETE CASCADE ON UPDATE CASCADE;
```

## INTRODUCCIÓN

Este documento recoge el trabajo realizado para el **módulo de Proyecto** del CFGS en Desarrollo de Aplicaciones Web.

Este módulo profesional complementa la formación establecida para el resto de los módulos profesionales que integran el título en las funciones de **análisis** del contexto, **diseño** o **desarrollo** del proyecto y organización de la **ejecución**.



## **ALCANCE DEL PROYECTO Y ANÁLISIS PREVIO**

El propósito del proyecto es, ante todo, fomentar el trabajo en equipo y crearlo a partir de ideas de todos los participantes.

Queremos realizar una comunidad online ligada a un videojuego actual (Starcraft II). Con esto pretendemos demostrar conocimientos en PHP, javascript, JQuery, DB, etc. Además, pretendemos demostrar lo que hemos aprendido también por nuestra cuenta, y utilizar un sistema de control de versiones o protocolos y métodos que no se hayan visto en clase.

Realizando un proyecto basado en una Comunidad y en un foro queremos desarrollar una herramienta que sirva a los usuarios como portal de enlace a ver streamings de jugadores profesionales de Starcraft II, y a su vez, que puedan comunicar sus impresiones sobre el juego, y todas las características que éste contiene.

El desarrollo de éste proyecto se ha llevado a cabo en varias fases: estudio de viabilidad, análisis, diseño, implementación y pruebas, y explotación o ejecución. A continuación se detallan las actividades/tareas/procedimientos de cada una de estas fases.

-Hemos iniciado el proyecto con una fase de estudio de viabilidad y análisis, hemos evaluado si el proyecto era factible en el tiempo indicado y con los conocimientos que tenemos o que podemos adquirir y; una vez hecho esto, hemos analizado las ideas de cada uno y las hemos ido dando forma poco a poco.

# **1 ESTUDIO DE VIABILIDAD**

En conocimiento del alcance que podría tener el proceso de generado de un foro, sabiendo que es posible implementar características y funcionalidades adicionales se ha procedido a realizar un desarrollo escalado del mismo. Con ello conseguimos que los requisitos mínimos se cumplan, y a partir de ahí, desarrollar extras con el fin de dar más funcionalidades a la aplicación web.

## **1.1 *Estado actual del sistema***

El sistema actual está basado en las especificaciones básicas de un foro

## **1.1 *Resumen de requisitos del cliente***

La Comunidad Starcraft está orientada a jugadores y seguidores de Starcraft II para satisfacer sus necesidades de comunicación con otros jugadores de habla hispana, así como de proporcionarles herramientas para aprender y mejorar sus estilos de juego.

## **1.1 *Posibles soluciones***

En la actualidad la Comunidad de Starcraft II más extendida se encuentra en la web del propio juego: <http://us.battle.net/sc2/es/> . Como es lógico nuestra web no pretende competir con dicha comunidad, sino ofrecer características adicionales, como podría ser el visionado de Streams por medio de la plataforma Twitch.

## **1.1 *Solución elegida***

Nos decantamos por la implementación de una comunidad con un foro y características multimedia con el fin de aunar dos conceptos que en la web oficial del juego Starcraft II no son implementados. Ofrecemos encuentros en tiempo real con el fin de que los usuarios de nuestra aplicación web puedan aprender de jugadores profesionales, puedan comunicarse con personas con su misma afición, y puedan encontrar contenidos generados por otros usuarios que puedan considerar beneficiosos o entretenidos.

## **1.1 Planificación temporal de las tareas del proyecto Comunidad Starcraft**

### **1.1 Planificación de los recursos a utilizar**

Los recursos a utilizar son los tres estudiantes encargados de realizar este proyecto realizando una organización dividida en 4 ramas de desarrollo, pudiendo realizar tareas de otra rama ayudándonos entre nosotros.

Los sectores en los que se dividen el proyecto son:

- **Rama diseño web:**

En esta rama se realizarán los editados de los estilos, así como la implementación y colocación de los componentes diseñados en el proyecto. Para ello son usadas las tecnologías .css o stylesheet, jQuery y Javascript combinándolas con HTML.

- **Rama desarrollo “MainPage” o portada:**

Esta rama abarca el desarrollo de la página principal de la Comunidad Starcraft. Esto implica el desarrollo de módulos personalizados para conectar con la plataforma Twitch, con Twitter y creación de un slider personalizado para mostrar noticias actuales de la web oficial del juego Starcraft II. Son necesarias las tecnologías PHP, Curl, MySQL para el desarrollo de la misma.

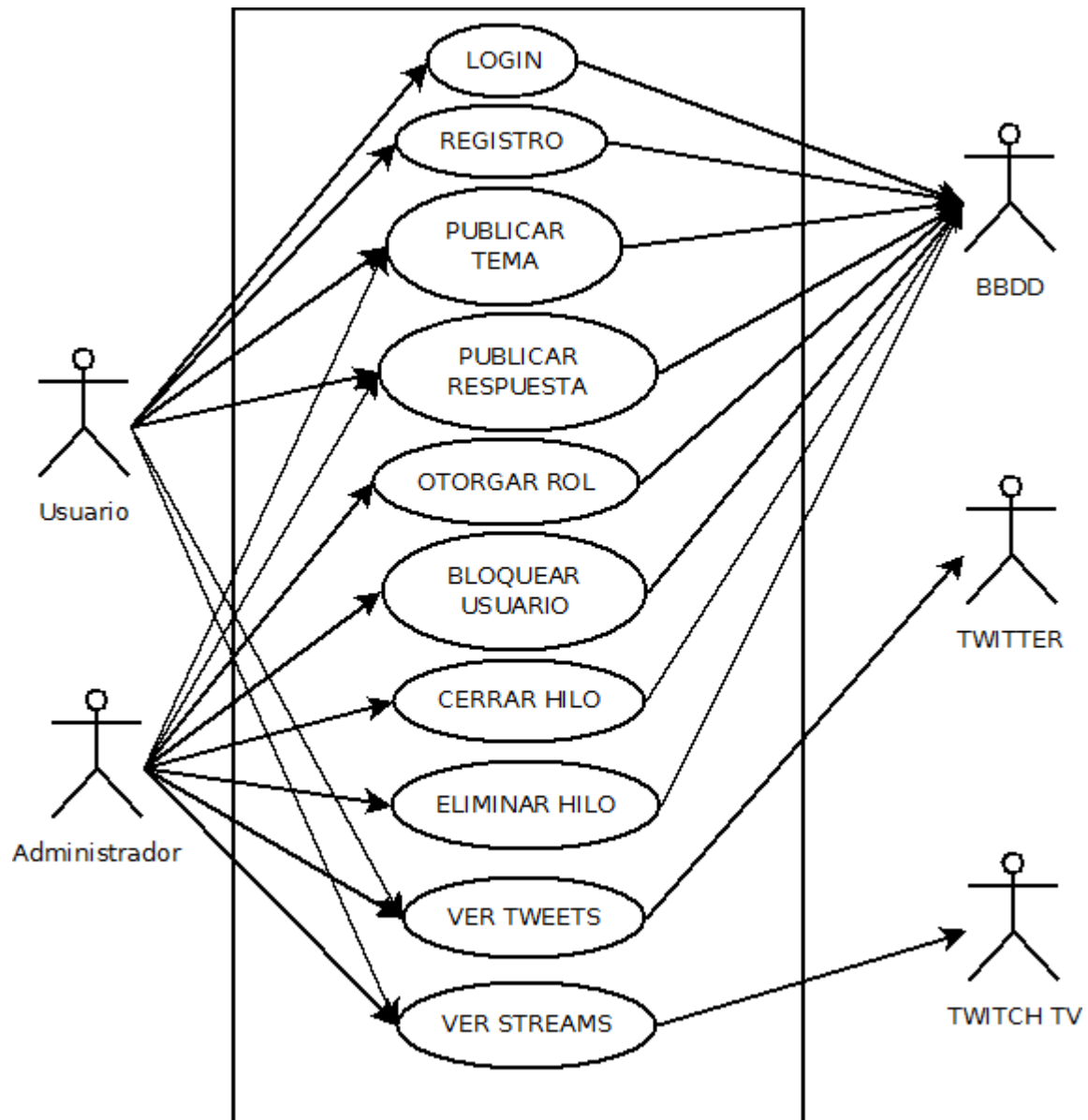
- **Rama desarrollo Foro:**

En esta rama se realiza el desarrollo de un foro, creación de hilos, creación de temas, y tareas administrativas asociadas a ello, como pudiera ser por ejemplo bloqueos o eliminaciones de temas. En este caso se utiliza MySQL y PHP para realizar este desarrollo.

- **Rama desarrollo Panel de Control y Perfil:**

Abarca todo el muestreo de los datos de un registro de usuario, y en el caso de un administrador de la aplicación web, el mantenimiento de los usuarios, el bloqueo de los mismos, y la administración de los usuarios otorgándoles cambios de rol en la web. Para ello usaremos la combinación PHP con MySQL.

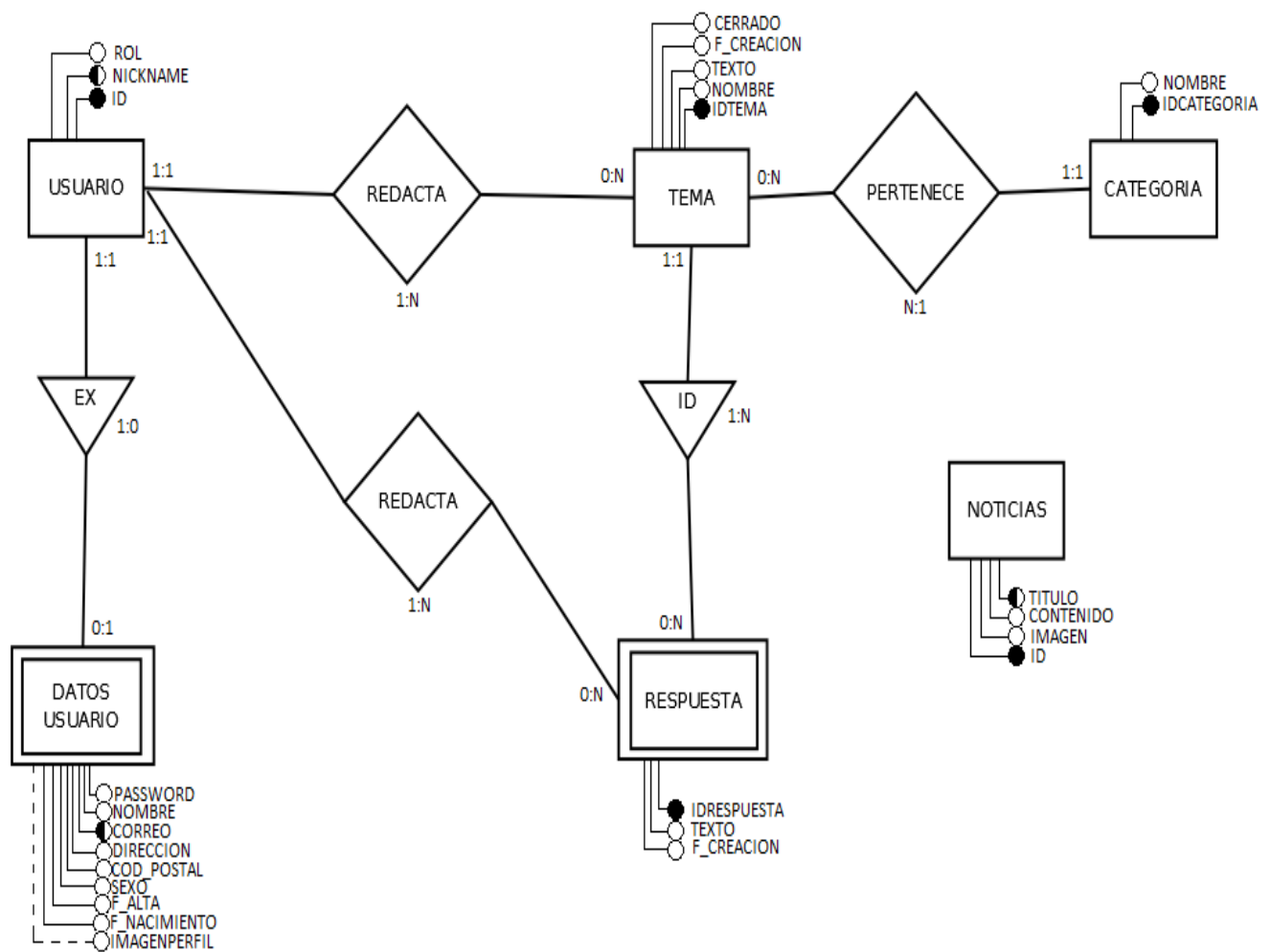
## 1.1

*Diagrama de casos de uso.*

## 1.2

## Modelo de datos

### DIAGRAMA ENTIDAD-RELACION METODOLOGIA CHENG:



## **1.1**

### ***Requisitos funcionales***

El sistema deberá mostrar información referente a un foro con un usuario con tipo de rol “miembro” de una forma rápida y eficaz permitiendo enviar respuestas si el hilo no estuviera cerrado. Si el acceso se realiza con un usuario administrador debe permitir cerrar y eliminar dichos hilos. En caso de que el acceso sea realizado con el rol moderador permite cerrar y abrir hilos.

También deberá mostrar una portada con unos contenidos dinámicos correspondientes a la cuenta de twitter `spain_starcraft`, la cual está relacionada con la comunidad.

El sistema mostrará streams en directo de jugadores profesionales muy conocidos por los aficionados a starcraft.

Se les ofrecerá a los usuarios acceso a su perfil para ver sus datos de registro, y poder subir un avatar que les identifique como usuario, siempre y cuando no superen unas medidas de peso de avatar.

El sistema proporcionará herramientas de administración para los administradores de la aplicación web para poder cambiar rol a usuarios que no sean ellos mismos, así como poder bloquear y desbloquear usuarios en la web.

Por último el sistema proporcionara herramientas a los administradores para poder abrir y cerrar hilos, y para poder eliminar éstos de una forma definitiva.

## **1.2**

### ***Requisitos no funcionales***

La aplicación web deberá tener un diseño intuitivo, fomentando la fácil inclusión de información por parte de los foreros, y también el acceso a la misma.

## **1.1 Estructura de la aplicación**

Aplicación web con fines no comerciales.

### **1.1 Componentes del sistema / arquitectura de red**

El sistema funcionará bajo un servidor Xampp, el cual a su vez tendrá activo un servidor MySQL y un servidor Apache.

Opcionalmente, y como extra, en caso de disponer de tiempo suficiente de desarrollo de proyecto, se podrá hacer uso de servidores Google para envío de correo automático desde el código PHP.

### **1.1 Herramientas y tecnologías utilizadas.**

Se han utilizado las siguientes tecnologías y herramientas:

MySQL: usado como Sistema Gestor de Bases de Datos para realizar un guardado fijo de datos.

PHP: ha sido el lenguaje elegido para el desarrollo de la aplicación web. Como lenguaje de programación de bajo nivel de software libre se adapta a los requisitos que necesitábamos.

Curl: librería de funciones PHP que permiten realizar conexiones a BBDD a través de Url's.

HTML: lenguaje de hipertextos basados en marcas básico en la creación de páginas web. Era necesaria su inclusión.

Css: para realizar la maquetación de la aplicación web se ha usado StyleSheets u hojas de estilos.

JavaScript: lenguaje de programación del lado cliente el cual hemos usado para realizar efectos ante la interacción del cliente.

Jquery: framework del lenguaje de programación Javascript. Es una herramienta más potente que éste y de uso más sencillo. Ha sido utilizada en multitud de efectos realizados en la aplicación web, como por ejemplo el slider de las noticias.

Bluefish: utilizado por Adrián Conde como editor de textos.

Sublime Text: utilizado por Cristian Rivera como editor de textos.

Notepad++: utilizado por Daniel González como editor de textos.

DropBox: en primera instancia fue usado para albergar el desarrollo y las versiones del proyecto. Debido a su baja fiabilidad en estos aspectos fue sustituido.

GitHub: programa que hemos usado para albergar los archivos desarrollados, ya que es gratuito y realiza un control de versiones adecuado a lo necesitado.



# 1

## IMPLEMENTACIÓN

### 1.1

### *Implementación del modelo de datos*

NOTICIAS ( ID , TITULO , CONTENIDO , IMAGEN )

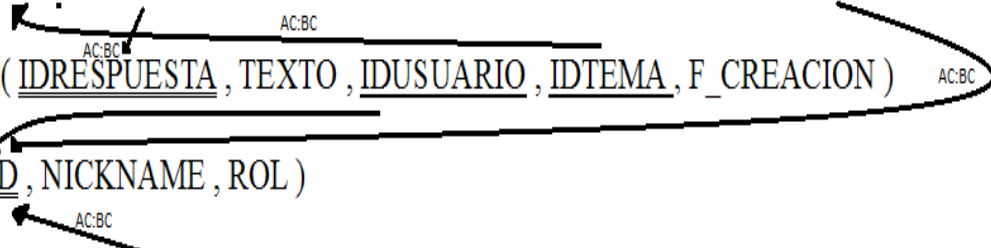
CATEGORIA ( IDCATEGORIA , NOMBRE )

TEMA ( IDTEMA , IDCATEGORIA , NOMBRE , TEXTO , F\_CREACION , USUARIO )

RESPUESTA ( IDRESPUESTA , TEXTO , IDUSUARIO , IDTEMA , F\_CREACION )

USUARIO ( ID , NICKNAME , ROL )

DATOS USUARIO ( IDUSUARIO , PASSWORD , NOMBRE , CORREO , DIRECCION ,  
COD\_POSTAL , SEXO , F\_ALTA , F\_NACIMIENTO , IMAGENPERFIL\* )



## **1.1 Carga de datos**

Tras las configuraciones iniciales, la carga de datos se realiza a través del script de configuración entregado con el proyecto.

## **1.1 Configuraciones realizadas en el sistema**

Es necesaria la instalación de Xampp, junto con la instalación de MySQL y Apache, junto con sus respectivos servicios. Es necesario, a su vez, la activación en el archivo de configuración del servidor Apache llamado php.ini de las directivas curl y short open tag. Esto se realiza des comentando las líneas pertinentes.

## **1.1 Implementaciones de código realizadas**

El código ha sido diseñado por los alumnos Cristian Rivera, Adrián Conde y Daniel González. Hemos intentado comentar los puntos que pudieran ser más conflictivos en el mismo con el fin de facilitar la lectura del mismo.

# 1

## PRUEBAS

Son muchas las pruebas que pueden realizarse en un proyecto para eliminar los posibles errores y garantizar su correcto funcionamiento. Los casos de prueba establecen las condiciones/variables que permitirán determinar si los requisitos establecidos se cumplen o no.

A continuación se detallan algunos de los casos de prueba que se ejecutarán para comprobar la correcta construcción de este proyecto.

### 1.1 *Casos de pruebas*

Como ejemplo se indican un caso de prueba no testados mediante programación:

Caso de prueba: Crear tema

ID: 1

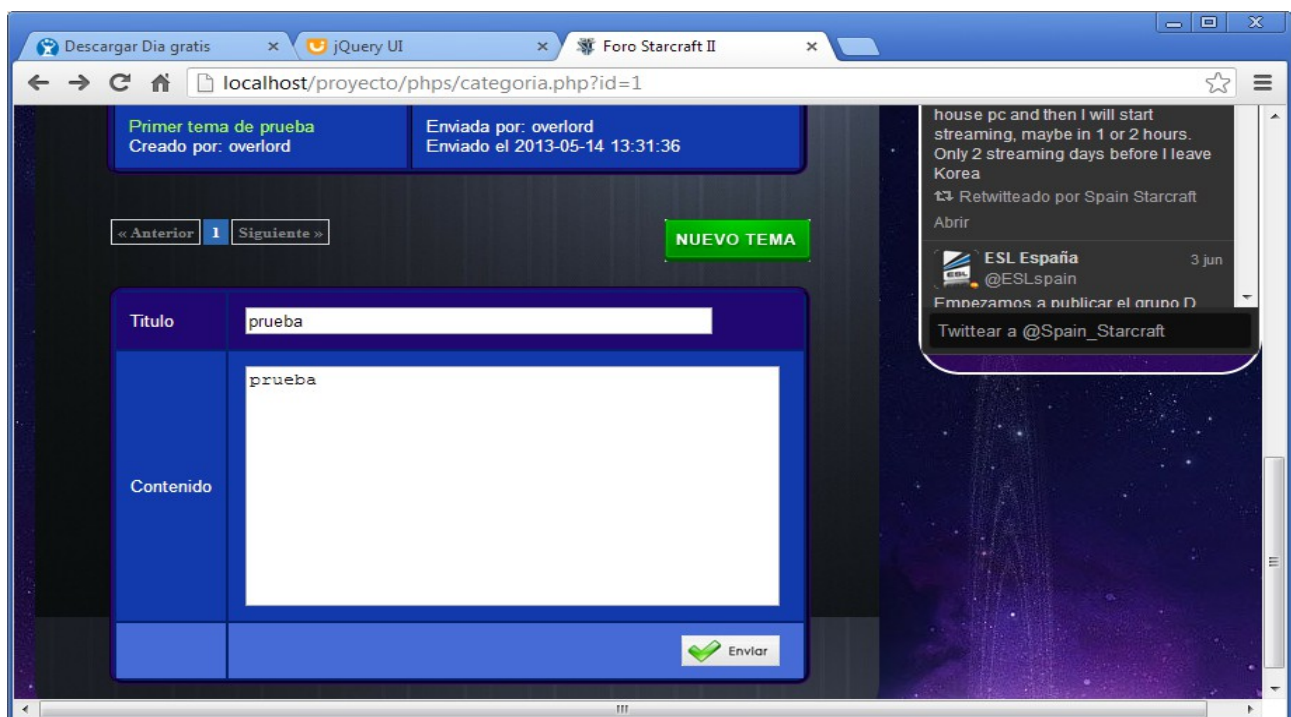
Descripción: Caso de prueba para rellenar un tema.

Objetivo: establecer si es posible crear temas.

Condiciones de ejecución: conectividad a Internet y a BBDD. Estar Logueado en la web.

Entrada: Nombre tema: prueba

Texto: prueba



Valor esperado: creacion de tema

Valor obtenido:



## **1 EXPLOTACIÓN**

La implantación es la fase más crítica del proyecto ya que el sistema entra en producción, es decir opera en un entorno real, con usuarios reales.

### **1.1 *Planificación***

Recomendamos el uso de servidores Windows. Debido a la poca carga que va a ocasionar la aplicación web no es necesario que haya instalado un Windows Server, pudiéndose usar perfectamente bajo cualquier versión comercial de Windows.

Recomendamos que la persona inicial a la que se le asigne el rol “Administrador” posea altos conocimientos en navegación web, y de cómo funciona un foro, pues será quien vaya otorgando roles a los demás usuarios.

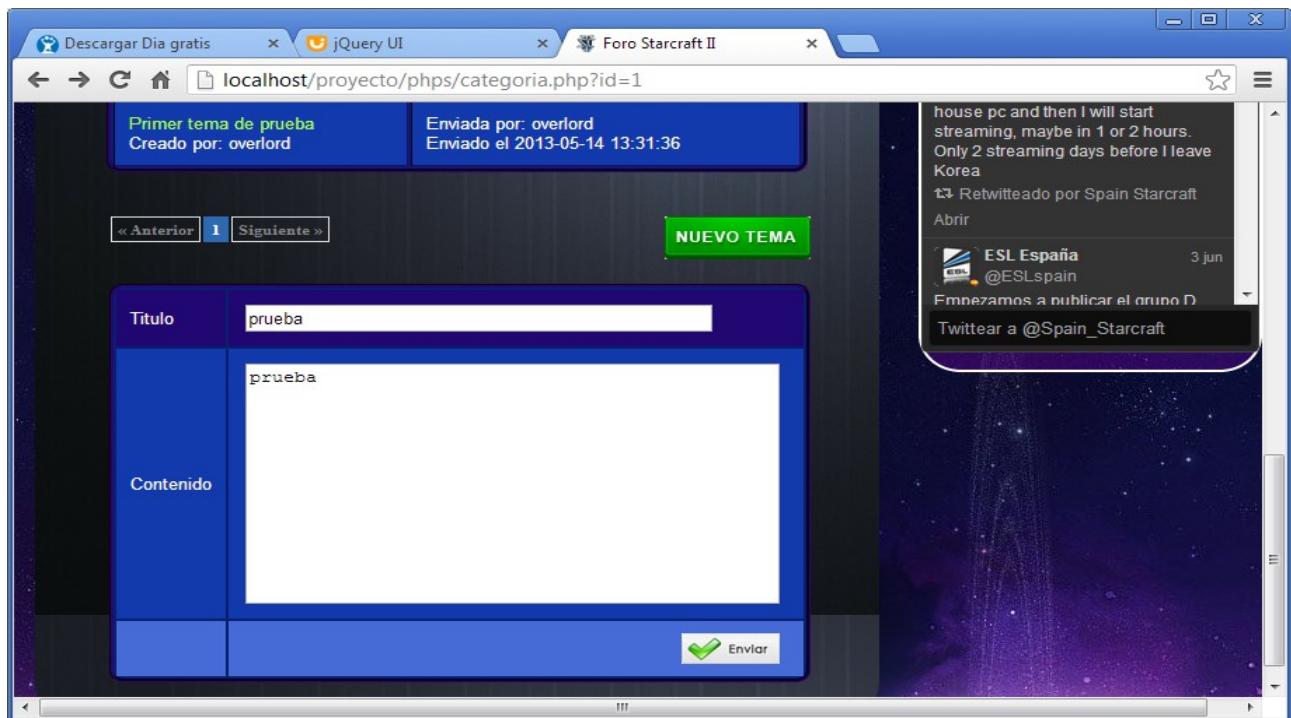
### **1.1 *Preparación para el cambio***

Los cambios estéticos en la web no siempre son bien llevados por los usuarios. En cuanto a los cambios de funcionalidad se debe evaluar si son o no necesarios, y si la aceptación puede ser buena entre la Comunidad.

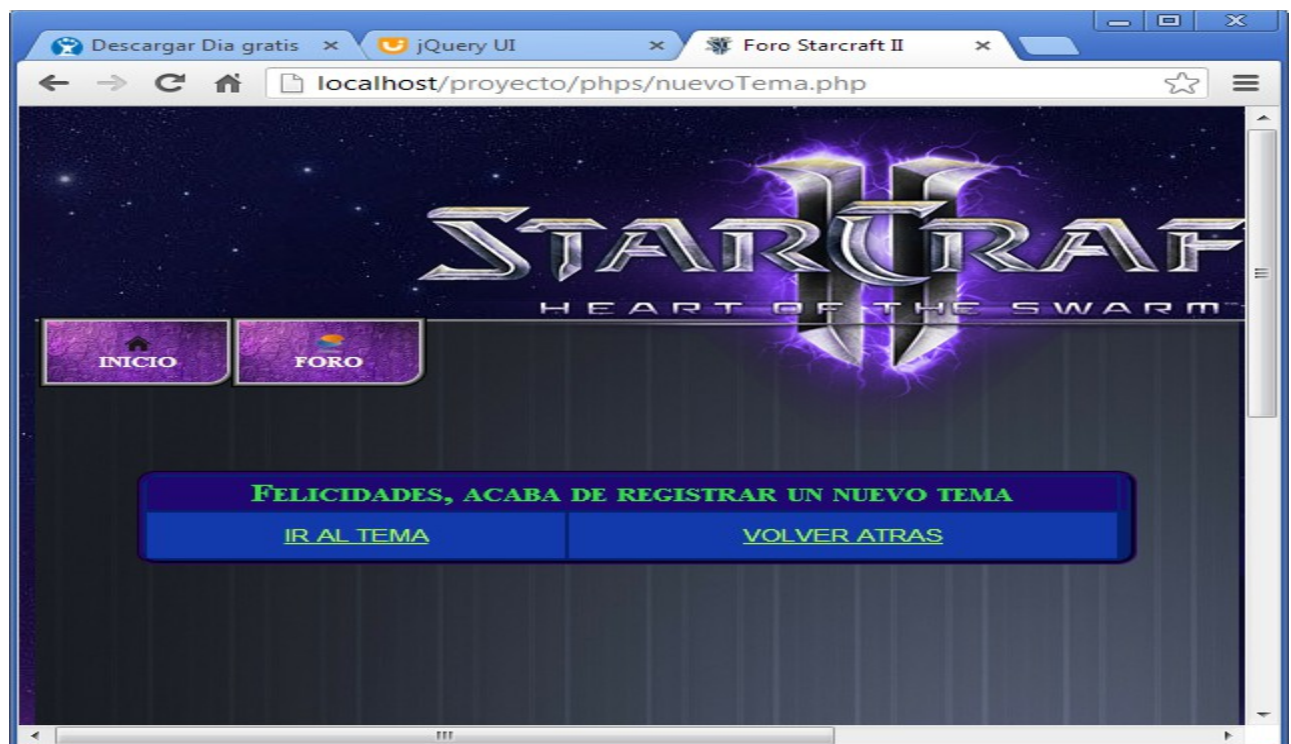
### **1.1 *Manual de usuario***

Incluido como anexo

## 1.1 *Implantación propiamente dicha*



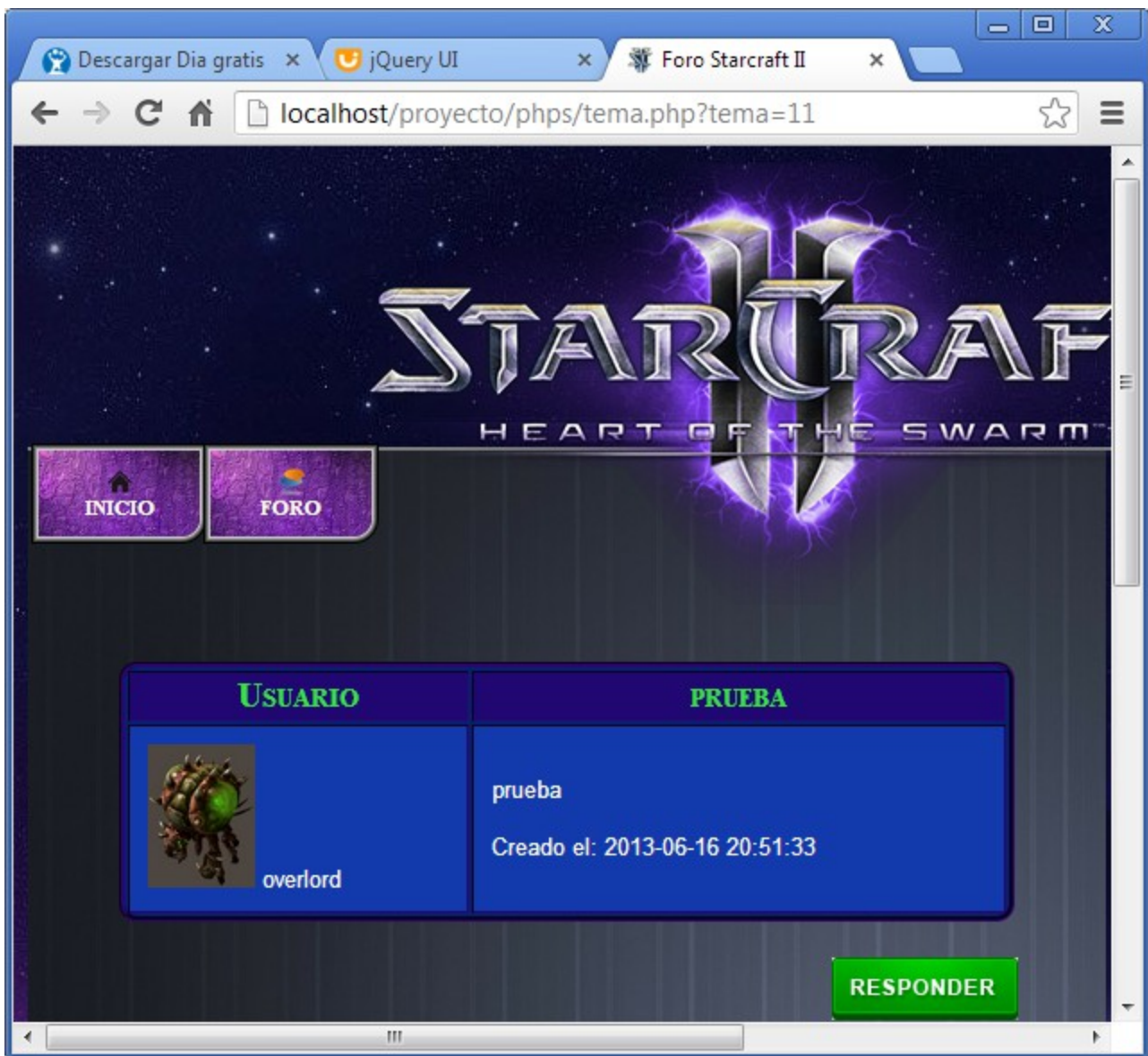
Por ejemplo funcionamiento a la hora de crear un tema nuevo





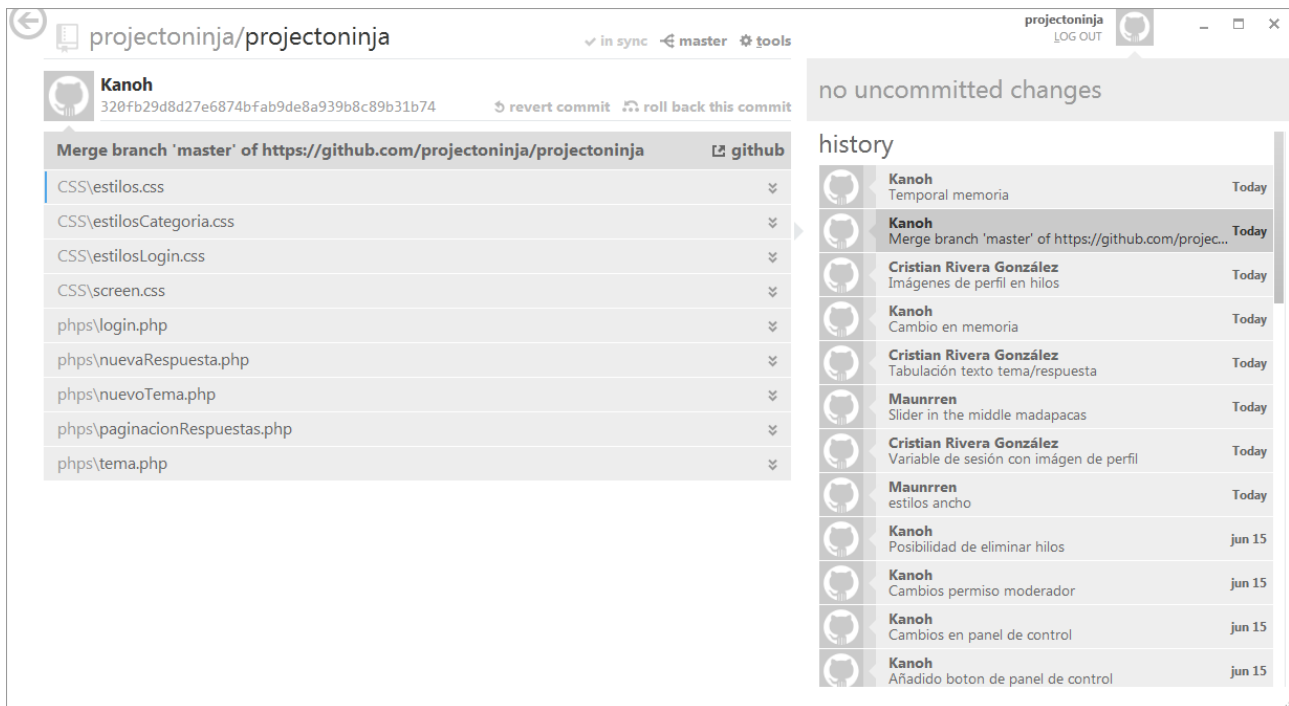
## COMUNIDAD STARCRAFT

Demostración de funcionamiento:



## CONTROL DE VERSIONES

### Uso de GitHub:



En esta captura se observa la solución de una bifurcación o “merge” por medio de github. A su vez se puede observar los diferentes commits realizados por los desarrolladores.



## COMUNIDAD STARCRAFT

projectoninja/projectoninja

in sync master tools

Kanoh

320fb29d8d27e6874bfab9de8a939b8c89b31b74

revert commit roll back this commit

Merge branch 'master' of https://github.com/projectoninja/projectoninja

github

CSS\estilos.css

CSS\estilosCategoria.css

CSS\estilosLogin.css

CSS\screen.css

phps\login.php

phps\nuevaRespuesta.php

phps\nuevoTema.php

phps\paginacionRespuestas.php

phps\tema.php

no uncommitted changes

Maunrren

titulo perfil

jun 15

Cristian Rivera González

Arreglado error login

jun 15

Maunrren

estilo categoria

jun 15

Maunrren

arreglo perfil

jun 15

Maunrren

perfil

jun 15

Cristian Rivera González

Cambios login

jun 15

Kanoh

Creacion de panel de control y editado de roles

jun 15

Cristian Rivera González

Reparado error perfil

jun 15

Kanoh

Errores estilos dependientes de clases en etiquetas

jun 15

Kanoh

Error login que transformaba el rol del usuario

jun 15

Kanoh

Corregidos errores en los estilos

jun 15

Maunrren

Merge branch 'master' of https://github.com/projec...

jun 15

Maunrren

ramkin estline tahlse

jun 15

I.E.S. Rey Fernando VI

Curso: 2013

## 1 FUENTES

### 1.1 *Legislación*

Enseñanzas mínimas: Real Decreto 686/2010, de 20 de mayo (BOE 12/06/2010)

[http://pdf/IFCS03/titulo/RD20100686\\_TS\\_Desarrollo\\_Aplicaciones\\_Web.pdf](http://pdf/IFCS03/titulo/RD20100686_TS_Desarrollo_Aplicaciones_Web.pdf)

Currículo: [Decreto 1/2011, de 13 de enero \(BOCM 31/01/2011\)](#)

[http://pdf/IFCS03/curriculo/D20110001\\_TS\\_Desarrollo\\_Aplicaciones\\_Web.pdf](http://pdf/IFCS03/curriculo/D20110001_TS_Desarrollo_Aplicaciones_Web.pdf)

Definición de procedimientos de control y evaluación:

- ⌚ <http://www.xperta.es/es/descripcion.asp>
- ⌚ <http://www.xperta.es/es/aquienvadirigido.asp>
- ⌚ <http://churriwifi.wordpress.com/2010/04/10/gestion-de-incidencias/>
- ⌚ [http://es.wikipedia.org/wiki/Control\\_de\\_versiones](http://es.wikipedia.org/wiki/Control_de_versiones)
- ⌚

### 1.1 *Bibliografía*

Manual PHP: <http://php.net/manual/es/index.php>

Manual JQuery: <http://jquery.com/>

Manual jQuery UI: <http://jqueryui.com/>

## 2

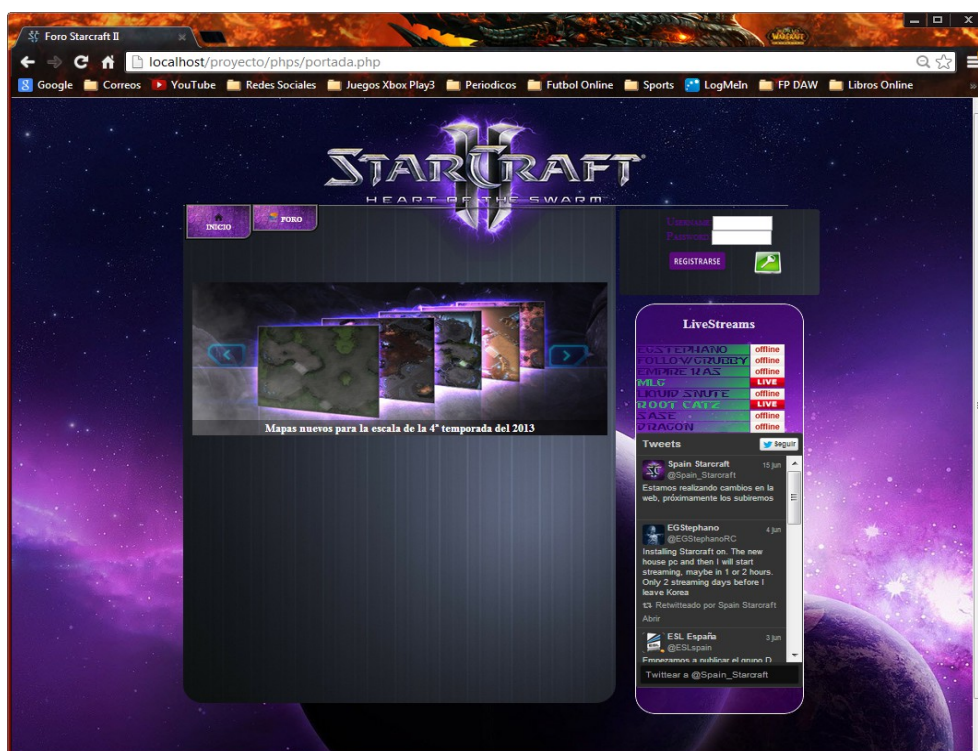
## ANEXOS

### MANUAL DE USUARIO

*Comunidad Starcraft* está diseñado para que tanto usuarios registrados como no tengan acceso a ciertos contenidos, teniendo estos últimos accesos a zonas extra de la aplicación web.

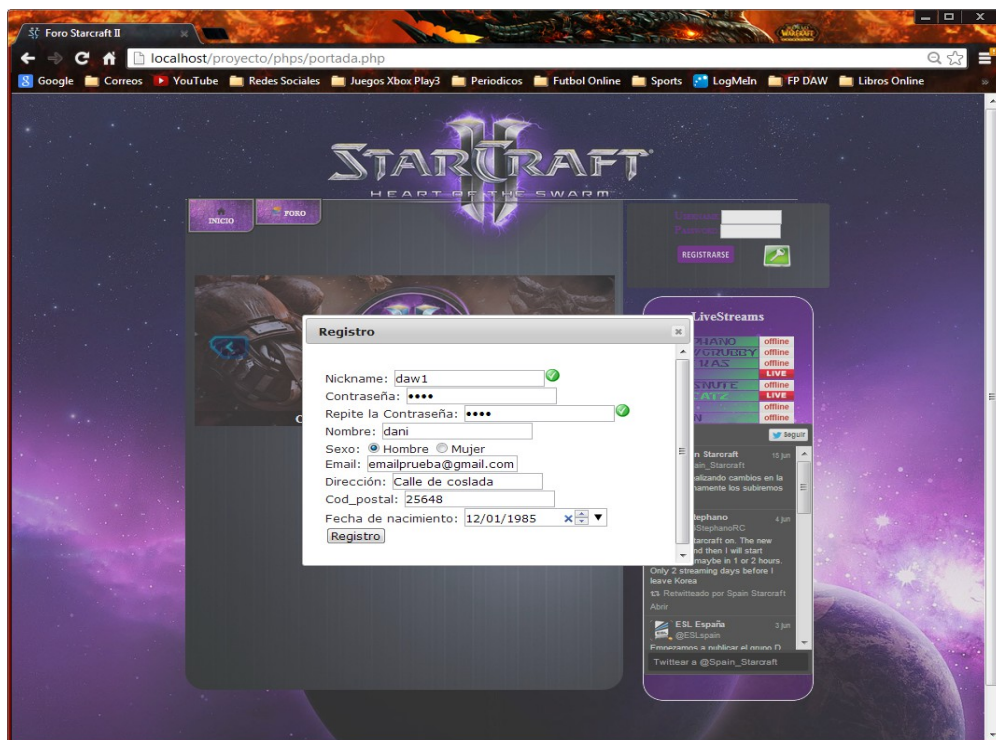
Si estas logueado podrás tener acceso a la portada de la página para ver el slider de noticias, y a la zona de acceso a tweets y visionado de streams. Por el contrario solo los usuarios registrados y debidamente logueados tendrán acceso al foro de la comunidad, así como a la zona privada del perfil.

1. Como crear un usuario en Comunidad Starcraft:

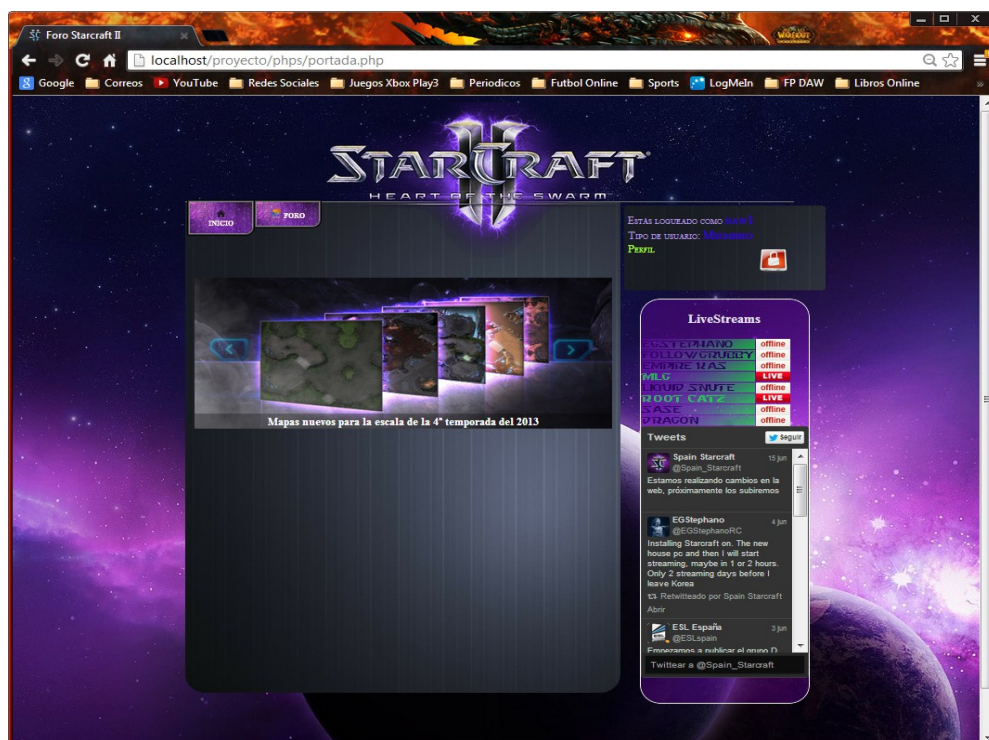


Será necesario clicar en el botón morado para registrarse de la parte superior-derecha de la aplicación. Tras ello aparecerá una ventana en la que habrá que incluir todos los datos de una forma correcta, para que el botón que valida el registro se active.

## COMUNIDAD STARCRAFT



Tras clicar en el botón registrado serás redireccionado de nuevo a la portada, pero ahora podrás observar en la parte superior-derecha que ya estarás “logueado” y preparado para disfrutar de todos los contenidos de la web.



## COMUNIDAD STARCRAFT

En el caso de que ya poseas un usuario en nuestra web, simplemente tienes que acceder con el usuario y la contraseña respectiva y serás automáticamente logueado.

### 2. Primer vistazo a la web.

Muy bien. Ahora que estás logueado tienes que decidir qué acciones quieres realizar, si ver streams, echar un vistazo a los tweets de la Comunidad Starcraft o simplemente interactuar con el foro. Además de ello bajo tu nombre de login (ventana superior-derecha) podrás ver tu perfil único de usuario. Dependiendo del rol que poseas es posible que no puedas usar acciones especiales en el perfil. Éstas serán explicadas con detenimiento más adelante.

### 3. Visionado de tweets.

En cualquiera de las ventanas de la Comunidad Starcraft, en la parte inferior derecha puedes ver los ultimos tweets y re-tweets que la cuenta oficial de la Comunidad Starcraft ha creado. Puedes incluso dar al enlace en cualquiera de los usuarios de Twitter para acceder directamente a dicha plataforma.

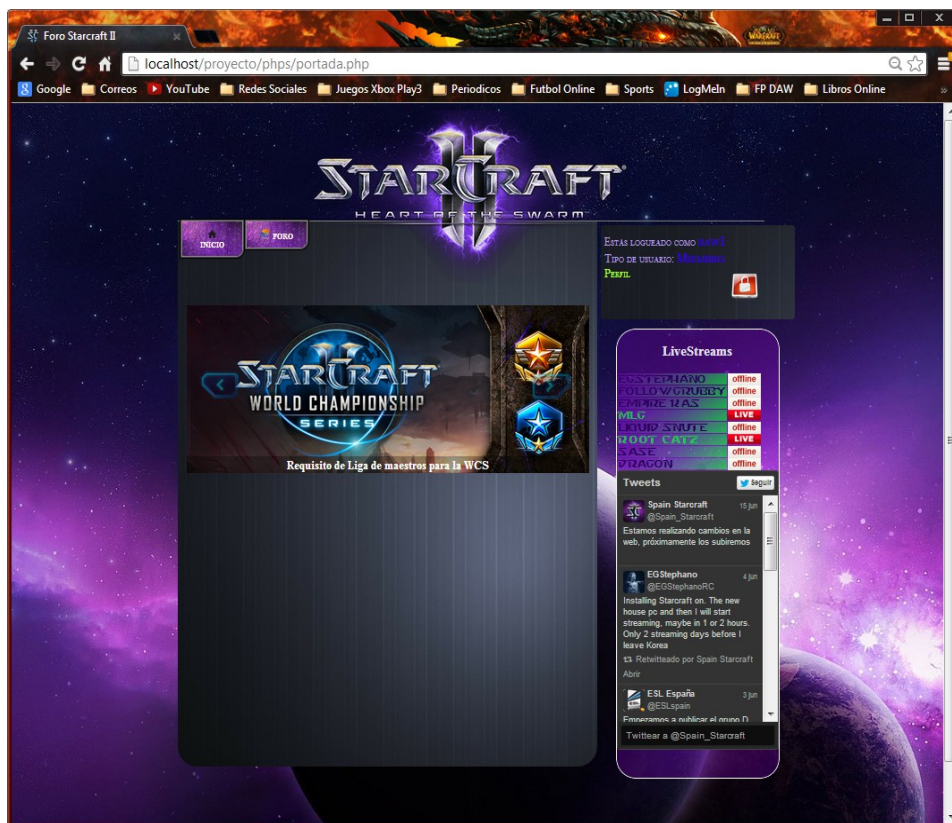




## COMUNIDAD STARCRAFT

### 4. Visionado de streams de jugadores profesionales:

En nuestra aplicación web hemos incluido acceso a un listado de los jugadores profesionales más famosos de Starcraft II. En caso de que alguno de ellos esté emitiendo en directo aparecerá, en la parte derecha de la web, un botón indicando live junto a su Alias de Starcraft II.

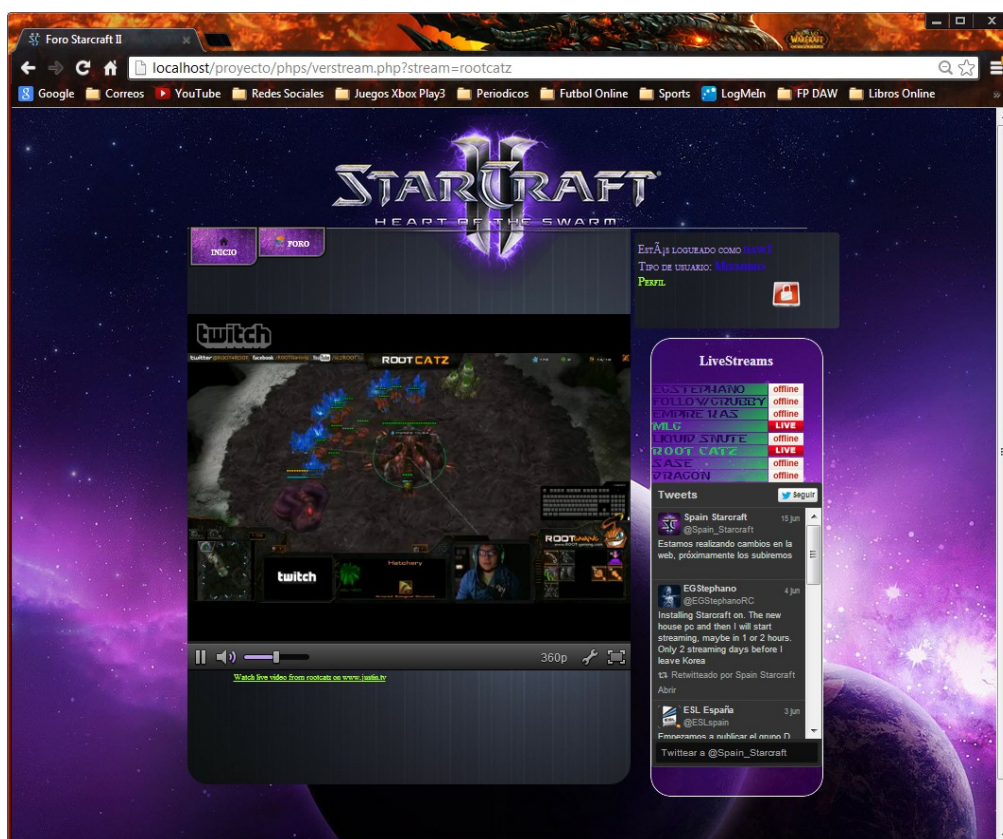


En este caso Root Catz está emitiendo en directo. Clicando en su nombre accederemos a la ventana de visionado de streams. Estos streams son proporcionados por la plataforma Twitch TV.

Recomendamos ajustar la definición del vídeo dependiendo de tu ancho de banda, y tras ello dar al play para disfrutar del stream.

Es posible que en algún caso no encuentres ningún stream en directo.

## COMUNIDAD STARCRAFT

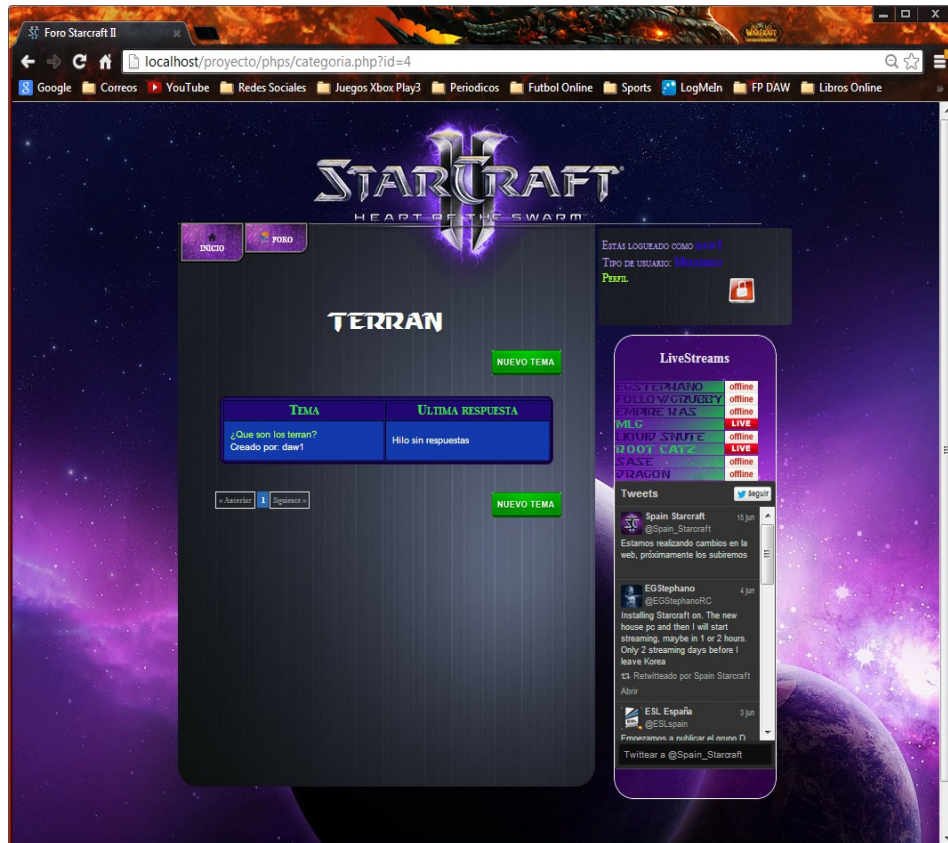
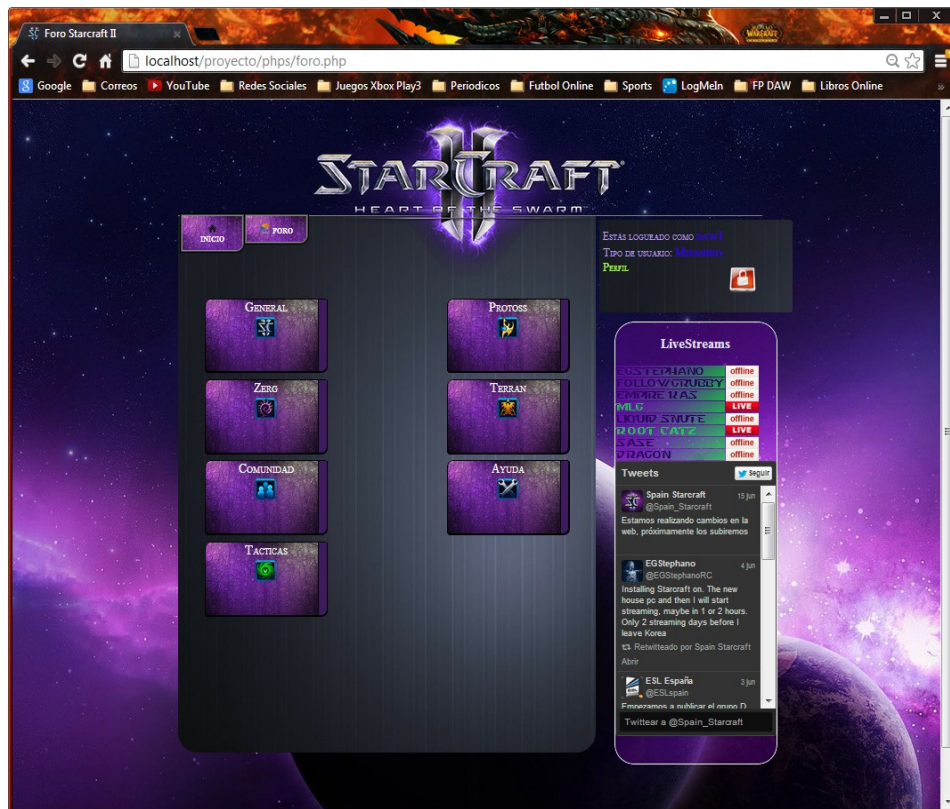


### 5. Disfrutando del foro.

Para acceder al foro has de dar al botón superior de la web en el que pone foro. Una vez dentro tendrás que elegir entre las distintas categorías que hemos creado para la comunidad.

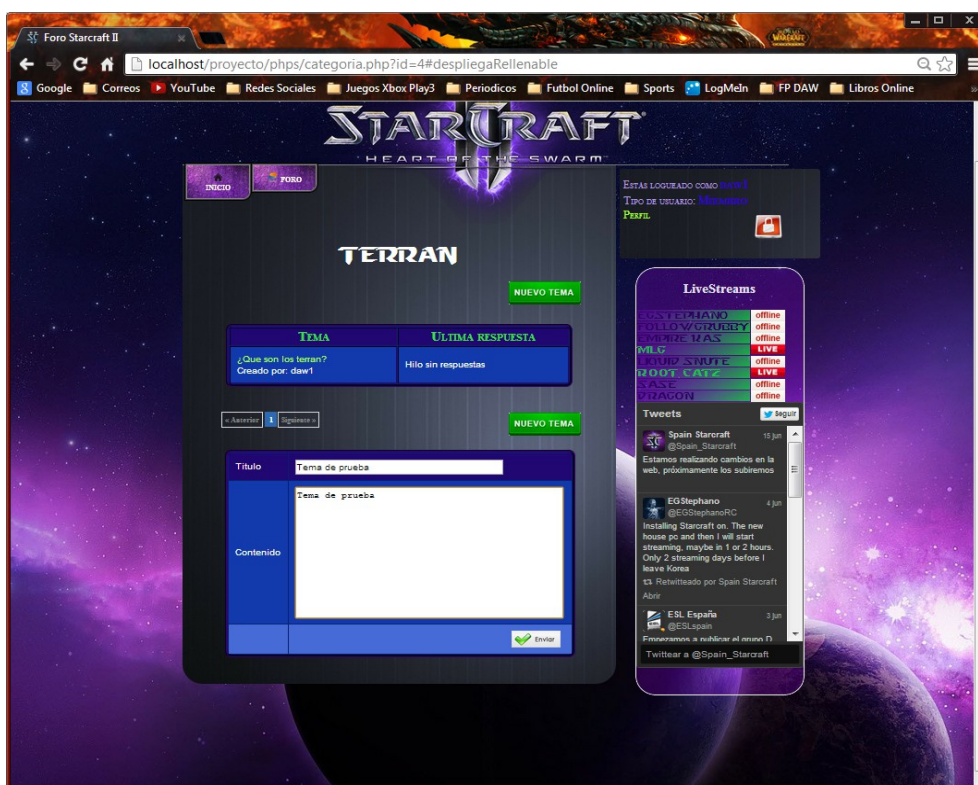
Dentro del foro que hayas seleccionado tendrás la opción de leer cualquier hilo (en caso de que los hubiera creados con anterioridad) o de crear un nuevo hilo, clicando en el botón Nuevo Tema tal y como se muestra en la imagen inferior.

## COMUNIDAD STARCRAFT





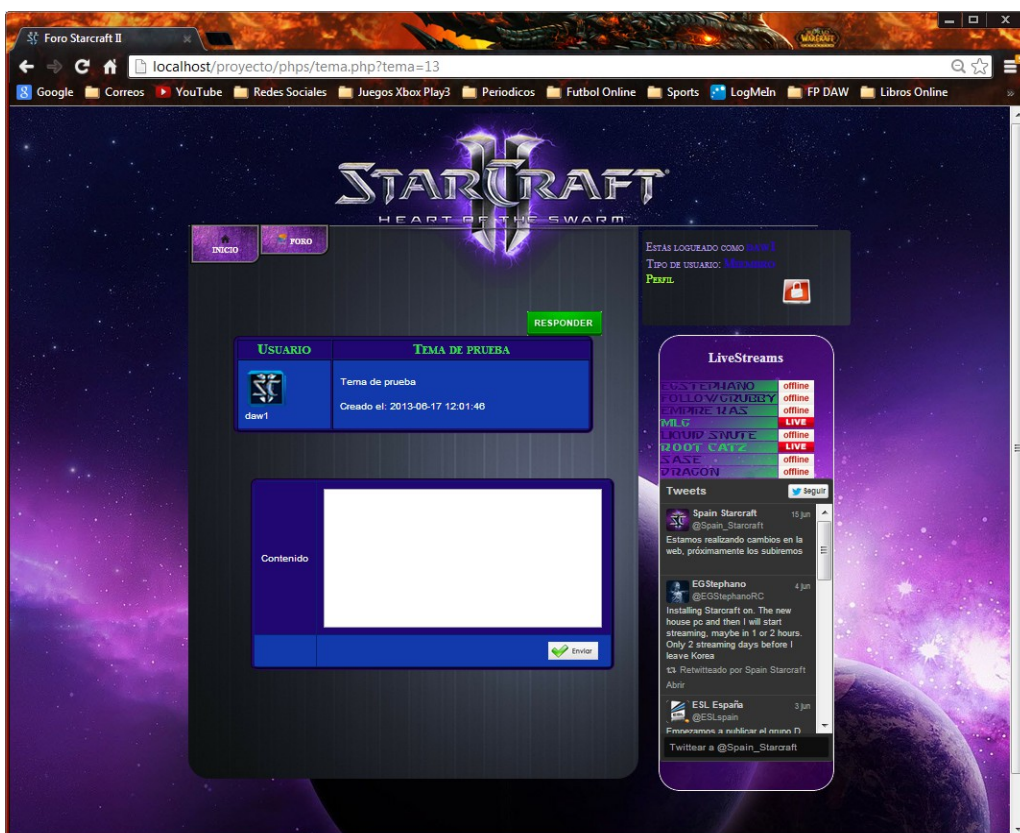
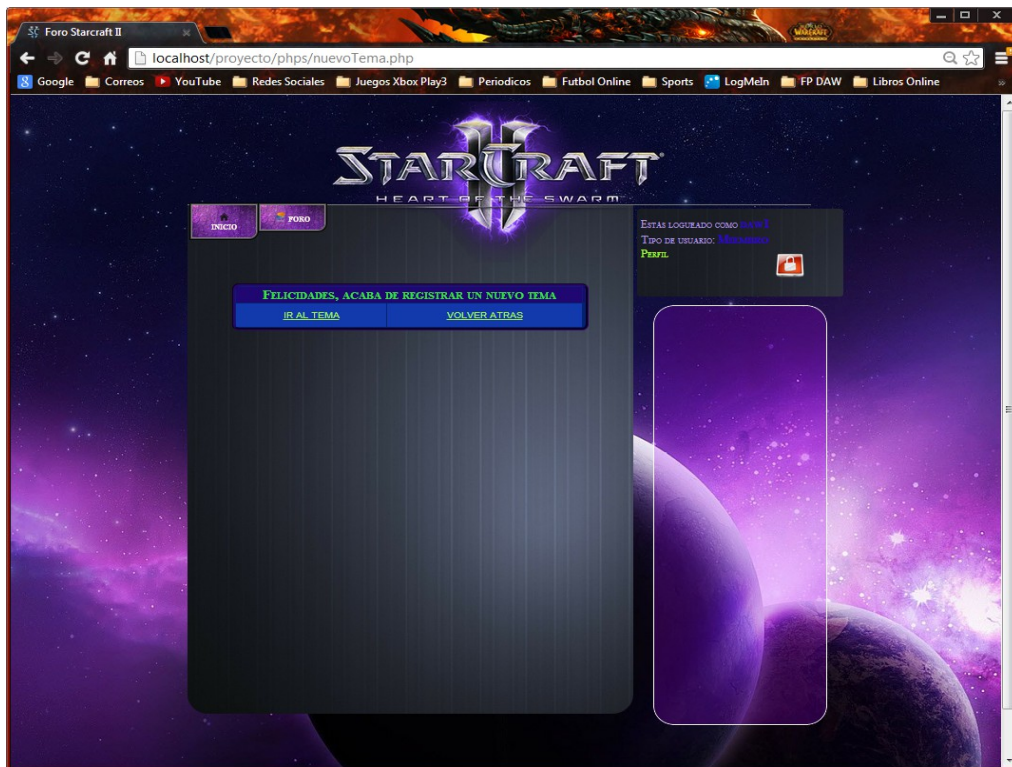
## COMUNIDAD STARCRAFT



Tras realizar estas acciones serás reenviado a la ventana de éxito en la cual se te preguntara si quieres volver a la categoría en cuestión, o ser reenviado al hilo recién creado.

En este último, tanto si eres tú el creador del tema, como si no, podrás escribir una nueva respuesta, siempre y cuando el hilo no haya sido cerrado por un Administrador o un Moderador de la web.

## COMUNIDAD STARCRAFT

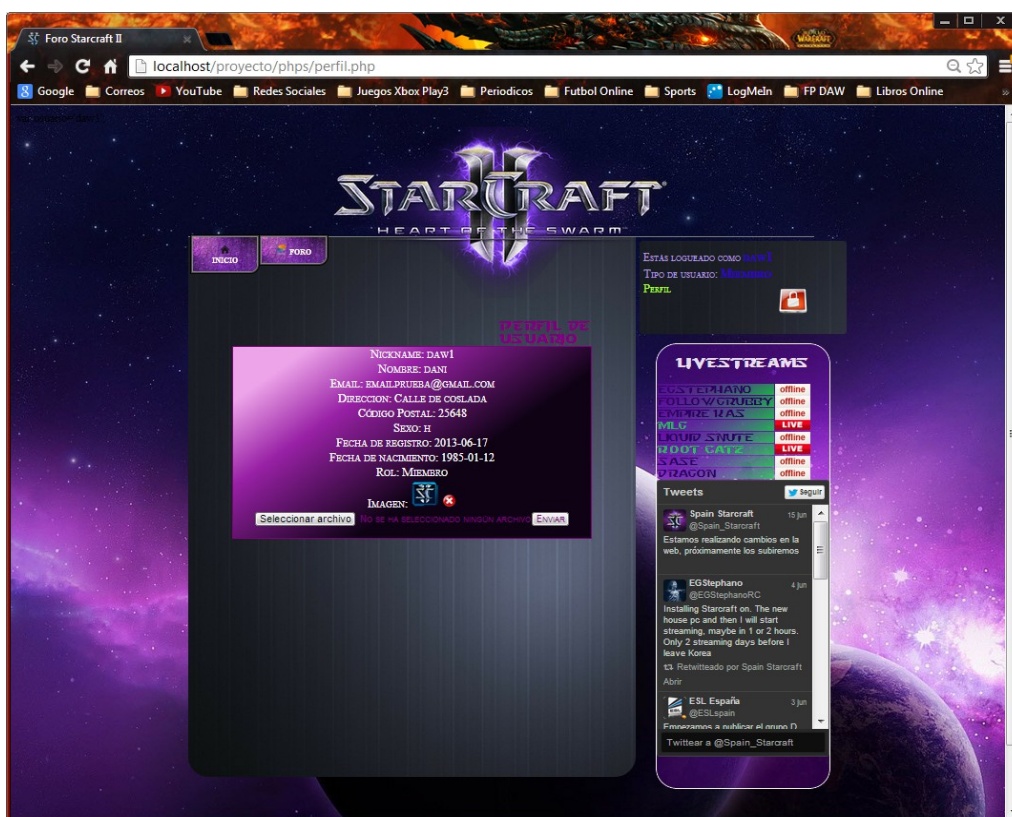


## COMUNIDAD STARCRAFT

Con esto conoces las funcionalidades básicas del foro de la Comunidad Starcraft II.

### 6. Perfil de usuario (Miembros comunes)

Para acceder al Perfil clicaremos en el enlace llamado Perfil, junto a tu nombre de usuario en la ventana superior-derecha. Tras ello nos encontraremos con nuestros datos de usuario incluidos en el registro, y con un botón en el cual podremos subir un avatar (la imagen debe tener un tamaño propio de avatar).

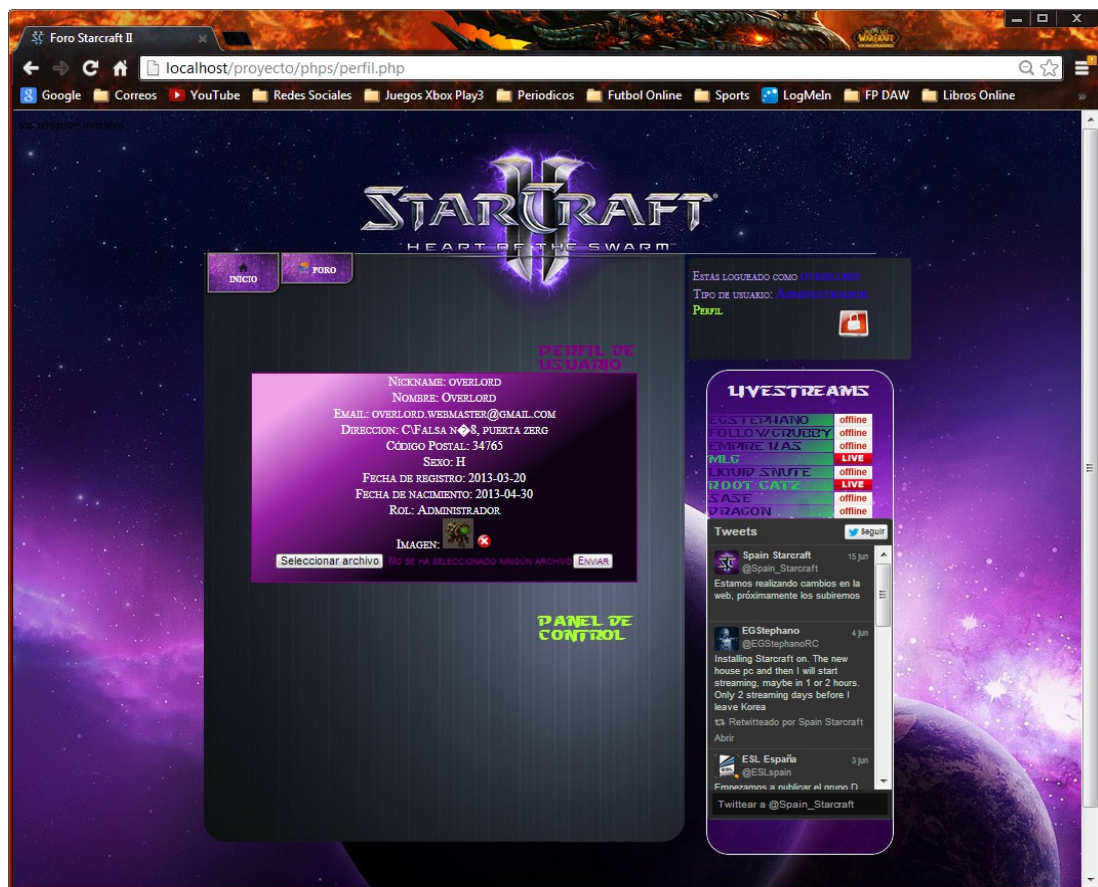


### 7. Perfil de usuario + Panel de Control (Administradores)

En el caso de que tu rol sea administrador, accediendo a la ventana de Perfil podrás realizar tareas de administración sobre otros usuarios que no seas tu, tales como otorgar roles a los mismos, y de manera adicional, bloquear y desbloquear usuarios.

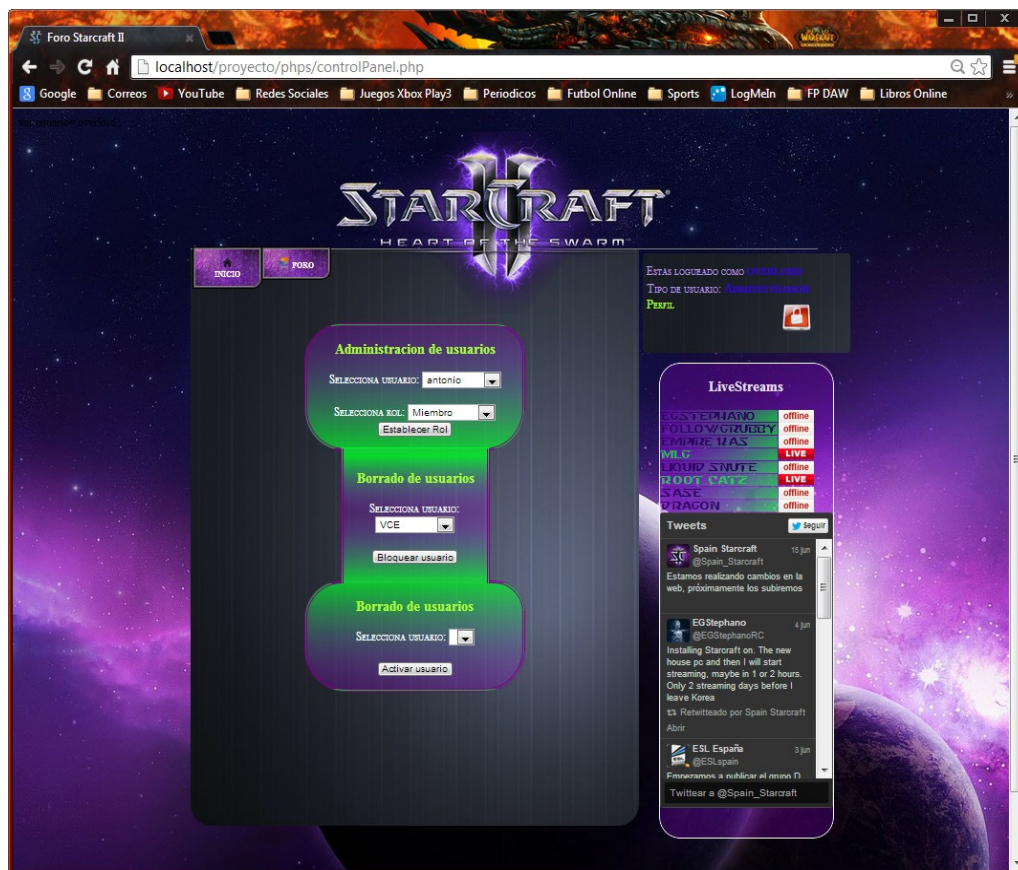


## COMUNIDAD STARCRAFT



Tras clicar en el enlace verde que indica PANEL DE CONTROL podrás acceder a la sección correspondiente y realizar dichas tareas.

## COMUNIDAD STARCRAFT



Recomendamos que el administrador del sistema tenga amplios conocimientos sobre foros, pues estas acciones deben de ser acordes con el mantenimiento del mismo.

Recomendamos, a su vez, la navegación en la Comunidad Starcraft en el navegador Chrome de Google.

## **COMUNIDAD STARCRAFT**

Trabajo realizado por los alumnos de Desarrollo de Aplicaciones Web:

- Adrián Conde.
- Cristian Rivera.
- Daniel González.