



VNIVERSIDAD  
D SALAMANCA

CAMPUS DE EXCELENCIA INTERNACIONAL

# TRABAJO FIN DE GRADO

## Revista On-Line sobre eSports

### RESUMEN

ESCUELA POLITÉCNICA SUPERIOR DE ZAMORA  
INGENIERÍA INFORMÁTICA EN SISTEMAS DE INFORMACIÓN

Departamento de Informática y Automática

Autor: ISRAEL CÁCERES MARCOS

Tutor: DIONISIO TOMÁS RODRÍGUEZ BARRIOS

Fecha adjudicación: MARZO 2019

Fecha presentación: SEPTIEMBRE 2019



UNIVERSIDAD DE SALAMANCA

Escuela **politécnica** superior  
de **Zamora**



## Tabla de contenido

1. INTRODUCCIÓN .....	5
2. OBJETIVOS DEL PROYECTO.....	5
2.1.    OBJETIVOS ESPECIFICADOS POR LOS REQUISITOS.....	5
2.2.    OBJETIVOS PERSONALES .....	6
3. CONCEPTOS TEÓRICOS .....	6
3.1.    Deportes Electrónicos.....	6
3.2.    Servidor Web.....	7
3.3.    Framework.....	7
3.4.    MVC .....	8
3.5.    RBAC .....	8
4. TÉCNICAS Y HERRAMIENTAS .....	9
4.1.    TÉCNICAS.....	9
4.1.1.    POO .....	9
4.1.2.    Modelo cliente-servidor.....	9
4.1.3.    Modelo vista-controlador (MVC).....	9
4.2.    HERRAMIENTAS .....	10
4.2.1.    Yii Framework.....	10
4.2.2.    XAMPP .....	10
5. ASPECTOS RELEVANTES DEL DESARROLLO DEL PROYECTO .....	11
5.1.    ASPECTOS RELACIONADOS CON EL ANÁLISIS .....	11
5.2.    ASPECTOS RELACIONADOS CON EL DISEÑO .....	12
5.3.    ASPECTOS RELACIONADOS CON LA IMPLEMENTACIÓN .....	12
6. CONCLUSIONES Y LÍNEAS DE TRABAJO FUTURAS.....	14
6.1.    CONCLUSIONES.....	14
6.2.    LÍNEAS DE TRABAJO FUTURAS .....	14



# 1. INTRODUCCIÓN

Los eSports, o deportes electrónicos, son competiciones profesionales en videojuegos multijugador competitivos.

Existen prácticamente desde el nacimiento de los videojuegos multijugador, donde empezaron con pequeños torneos en locales, pero han ido evolucionando, sobre todo a partir de los años 2000 con la llegada de los juegos en línea y el enfoque competitivo en el desarrollo de los videojuegos.

En los últimos años, se han profesionalizado hasta el punto de empatar o superar en expectación y audiencia a deportes como el Fútbol o el Baloncesto.

Actualmente las ganancias que puede llegar a obtener un jugador profesional de eSports son equiparables, por ejemplo, al sueldo de un deportista de élite, incluso, superando algunos torneos de eSports en ganancias a competiciones profesionales de gran nombre, como puede ser Wimbledon, en tenis profesional.

## 2. OBJETIVOS DEL PROYECTO

### 2.1. OBJETIVOS ESPECIFICADOS POR LOS REQUISITOS

En el siguiente listado tenemos los objetivos marcados por los requisitos del software:

- Desarrollar una aplicación web destinada a informar a los usuarios sobre la actualidad del mundo que rodea a los deportes electrónicos.
- Ofrecer una aplicación de fácil acceso a los usuarios que sirva como fuente de información, accesible desde internet mediante un ordenador o dispositivo portátil.
- Brindar un entorno web intuitivo y fácil de usar tanto para los usuarios de la web, como los redactores de noticias.
- Garantizar un control de acceso de los usuarios.
- Establecer un sistema de rangos o permisos para la realización de tareas en la aplicación web.

## 2.2. OBJETIVOS PERSONALES

En el siguiente listado tenemos los objetivos personales marcados para la realización del trabajo de fin de grado:

- Aprender y obtener conocimientos sobre la programación y diseño de aplicaciones web.
- Ampliar los conocimientos sobre el uso de un Framework en el desarrollo de aplicaciones web.
- Poner en práctica los conocimientos obtenidos durante el grado sobre las bases de datos relacionales.
- Mejorar los conocimientos sobre el lenguaje PHP, en el cual está basada esta aplicación web.
- Aplicar el uso de un patrón de diseño MVC (Modelo-Vista-Controlador).
- La consecución del título de Grado en Ingeniería Informática en Sistemas de Información.

## 3. CONCEPTOS TEÓRICOS

### 3.1. Deportes Electrónicos

Los eSport o “Electronic Sport” son una novedad en el campo de los videojuegos. Este concepto se utiliza para nombrar las competiciones organizadas a nivel profesional. Son competiciones multijugador de diferentes disciplinas de videojuegos. Su funcionamiento es muy sencillo: cada esport tiene sus propias reglas y tienen acceso a diferentes dispositivos y plataformas de forma online u offline.

Aunque en un primer momento pudiese parecer que no tendría éxito el formato —una audiencia que no disfrutaría tanto viendo jugar como jugando ellos mismos— ya se ha podido comprobar cómo estas competiciones de videojuegos en las que los espectadores observan cómo juegan otros gamers, enganchan.<sup>1</sup>

---

<sup>1</sup> **Galiana, Patricia.** [En línea] <https://www.iebschool.com/blog/que-es-esports-marketing-digital/#esports>.

### 3.2. Servidor Web

Un servidor Web es un programa que utiliza el protocolo de transferencia de hiper texto, HTTP (Hypertext Transfer Protocol), para servir los archivos que forman páginas Web a los usuarios, en respuesta a sus solicitudes, que son reenviados por los clientes HTTP de sus computadoras. Las computadoras y los dispositivos dedicados también pueden denominarse servidores Web.

El proceso es un ejemplo del modelo cliente/servidor. Todos los equipos que alojan sitios Web deben tener programas de servidor Web. Los principales servidores Web incluyen Apache (el servidor Web más ampliamente instalado), Internet Information Server (IIS) de Microsoft y nginx (que se pronuncia engine X) de NGNIX. Otros servidores Web incluyen el servidor NetWare de Novell, el servidor Web de Google (GWS) y la familia de servidores Domino de IBM.

Los servidores Web a menudo forman parte de un paquete más amplio de programas relacionados con internet e intranet para servir correo electrónico, descargar solicitudes de archivos de protocolo de transferencia de archivos (FTP) y crear y publicar páginas Web. Las consideraciones al elegir un servidor Web incluyen cuán bien funciona con el sistema operativo y otros servidores, su capacidad para manejar la programación del servidor, las características de seguridad y las herramientas particulares de publicación, motor de búsqueda y creación de sitios que vienen con él.<sup>2</sup>

### 3.3. Framework

El concepto de framework engloba un conjunto de componentes o módulos que pueden ser personalizados para desarrollar una aplicación.

En este proyecto se ha utilizado un framework web, orientado al desarrollo de aplicaciones web llamado Yii Framework.

---

<sup>2</sup> **Rouse, Margaret.** <https://searchdatacenter.techtarget.com/es/definicion/Servidor-Web>.

Gracias al uso de un framework, es posible acelerar el desarrollo de un proyecto, además de poder reutilizar código y aplicar patrones de desarrollo fácilmente, como puede ser Modelo-Vista-Controlador en nuestro caso.

### 3.4. MVC

Modelo, Vista, Controlador, es un patrón de diseño de software cuya principal característica es la separación del código en función de su finalidad.

Se divide en Modelos, dónde se incluyen las funciones que trabajan con los datos, las Vistas, que generarán la salida visual al usuario de la aplicación, y los Controladores, que responderán a la llamada de funciones de la aplicación web.

Los controladores sirven de enlace entre los modelos y las vistas, pues son capaces de generar una salida al usuario accediendo a los datos de un modelo y ofreciendo estos a una vista.

### 3.5. RBAC

El control de acceso basado en roles (Role-based Access control, RBAC) es una forma de restringir el acceso a la red según el rol que tenga una persona dentro de una organización.

Los roles en RBAC representan los niveles de acceso que tienen los empleados de una organización en la red.

De esta forma, cada empleado podrá tener acceso solo a la información que necesite según su autoridad o responsabilidad en la organización.

De esta forma, se puede controlar el acceso de una persona a la habilidad de ver, crear o modificar un archivo.

Normalmente, en una organización, los empleados de nivel bajo no tienen acceso a datos sensibles en la organización.

En este proyecto, utilizamos RBAC para establecer los distintos roles de los diferentes tipos de usuarios que pueden acceder a la aplicación web.



## 4. TÉCNICAS Y HERRAMIENTAS

### 4.1. TÉCNICAS

#### 4.1.1. POO

La programación orientada a objetos (POO, u OOP según sus siglas en inglés) es un paradigma de programación que viene a innovar la forma de obtener resultados. Los objetos manipulan los datos de entrada para la obtención de datos de salida específicos, donde cada objeto ofrece una funcionalidad especial.

Muchos de los objetos prediseñados de los lenguajes de programación actuales permiten la agrupación en bibliotecas o librerías, sin embargo, muchos de estos lenguajes permiten al usuario la creación de sus propias bibliotecas.

Está basada en varias técnicas, como las siguientes: herencia, cohesión, abstracción, polimorfismo, acoplamiento o encapsulación.

#### 4.1.2. Modelo cliente-servidor

El modelo o arquitectura cliente-servidor se basa en la realización de peticiones por parte de un usuario (cliente) a un proceso o recurso (servidor), el cual devuelve una respuesta a este.

Esta aplicación web se apoya en esta arquitectura: Nuestro servidor contiene los recursos a los que acceden los usuarios de la web, y a cada uno se le dará una respuesta en forma de vista según sea solicitado.

#### 4.1.3. Modelo vista-controlador (MVC)

Este proyecto se basa en el patrón de diseño MVC, pues se apoya en el framework Yii.

Yii implementa el diseño de patrón modelo-vista-controlador (model-view-controller MVC) el cual es adoptado ampliamente en la programación Web. MVC tiene por objeto separar la lógica del negocio de las consideraciones de la interfaz de usuario para que los desarrolladores puedan modificar cada parte más fácilmente sin afectar a la otra. En MVC el modelo representa la información (los datos) y las reglas del negocio; la vista contiene elementos de la interfaz de usuario como textos, formularios de entrada; y el controlador administra la comunicación entre la vista y el modelo.

## 4.2. HERRAMIENTAS

### 4.2.1. Yii Framework

Yii es un framework PHP basado en componentes, para la realización de aplicaciones web.

Es un framework muy versátil en relación al gran abanico de aplicaciones web para el que es apropiado su uso: Desde páginas web, foros, sistemas de gestión de contenidos y aplicaciones de comercio electrónico.

Yii implementa el patrón MVC, por lo que la organización de código está basada en este.

Además, Yii es fácilmente extensible personalizando o añadiendo nuevos módulos y extensiones.

### 4.2.2. XAMPP

XAMPP es un paquete de software, que engloba los programas necesarios para crear un entorno de desarrollo, en este caso, en PHP.

Su nombre es un acrónimo: X (que representa la compatibilidad multi-sistema operativo), Apache, MySQL, PHP y Perl.

#### 4.2.2.1. *Apache*

Apache nos ofrece las funciones de servidor web. Es gratuito y de código abierto.

Se estima que prácticamente la mitad de sitios webs del mundo son ejecutados en apache.

#### 4.2.2.2. *MySQL*

MySQL es un sistema gestor de bases de datos para bases de datos relacionales.

Es código abierto, gratuito y multiplataforma.

#### 4.2.2.3. *PHP*

PHP es un lenguaje de programación que puede utilizarse de forma estructurada, en programación orientada a objetos o scripts.

Es un lenguaje interpretado que se ejecuta en el lado del servidor.

Aunque es un lenguaje de programación de propósito general, se utiliza principalmente en el desarrollo web, por su facilidad para trabajar con bases de datos y HTML.

## 5. ASPECTOS RELEVANTES DEL DESARROLLO DEL PROYECTO

### 5.1. ASPECTOS RELACIONADOS CON EL ANÁLISIS

Durante la fase de análisis se empiezan a dar pinceladas de la ruta que tomara la solución.

Desde el principio tenemos claro que la revista tomará forma de aplicación web, por tanto, hay que elegir como llevaremos a cabo la implementación.

Para agilizar el proceso de desarrollo de la aplicación web, está claro que nos apoyaremos sobre un framework.

Finalmente se elige Yii Framework ya que no es nuevo para mí, pues hemos tenido prácticas con él durante la carrera.

La elección del software que nos sirve de apoyo durante el desarrollo tiene como objetivo encontrar software libre y de fácil puesta en marcha, por ello elegimos XAMPP, un paquete enfocado a la web, que incluye:

- Servidor Web, Apache. Como ya cité anteriormente, es uno de los servidores más usados en internet.
- Gestor de Bases de datos, MySQL.
- Motor PHP, ya que vamos a usar framework Yii, necesitamos que nuestro software de apoyo sea capaz de interpretar PHP

Yii Framework se basa en el patrón de diseño MVC, el cual creo que es el idóneo para una aplicación web que tendrá una parte pública, ya que separar la interfaz de la funcionalidad nos permite hacer actualizaciones en el diseño de manera sencilla.

## 5.2. ASPECTOS RELACIONADOS CON EL DISEÑO

El principal objetivo de estos aspectos está relacionado con el diseño de la base de datos que será el corazón de la aplicación web.

Se trabajará con una base de datos relacional, que nos permite asegurar la integridad referencial (Si se elimina un registro, también lo harán los demás registros relacionados), además brinda herramientas para evitar la duplicidad de registros.

También favorece la normalización por ser más comprensible y aplicable.

Otro aspecto a destacar es que cada tabla de la base de datos tendrá asociado un controlador en nuestro sistema basado en MVC.

Desde cada controlador podemos acceder a las funciones de creación, actualización y borrado de datos, es decir, podemos mantener la base de datos desde nuestra implementación.

Respecto al diseño visual de cara al usuario, se ha decidido hacer un diseño simple y de fácil uso, siguiendo un patrón muy similar en todas las vistas que ofrece la aplicación web.

## 5.3. ASPECTOS RELACIONADOS CON LA IMPLEMENTACIÓN

Durante la fase de implementación surgen distintos interrogantes, que se han intentado resolver de la mejor forma posible.

A continuación, vamos a describir algunos de las cuestiones más importantes relacionadas con este aspecto.

La primera encrucijada que apareció fue como gestionar los distintos permisos o roles de la aplicación web para manejar el acceso a los distintos apartados. Yii Framework ofrece un sistema RBAC, que debemos configurar convenientemente. Para ello se establece una lista de roles fijos, ciertas acciones, y se graban en la base de datos, en una tabla establecida para ello. Luego, estos roles se asignan a cada usuario mediante otra tabla que los relaciona.

Por otro lado, uno de los objetivos es que, en el registro del usuario, el email que se introduce tenga que ser verificado de alguna forma. Una vez más, Yii Framework nos brinda un componente que, configurado de forma correcta, nos permite enviar correo electrónico. El mecanismo de validación del email, consiste en lo siguiente:

El usuario se registra en la aplicación web rellenando un formulario con sus datos. Estos datos incluyen campos que no cambiarán, como el id, el nick o la fecha de registro. Con estos campos, concatenados, creamos un hash md5, que nos servirá como código de control. Este control será enviado incrustado en un enlace para el usuario, junto con el identificador, que, al acceder a este, la acción correspondiente comprobara que el código de control es correcto para ese usuario, y si lo es, el usuario pasará a estar confirmado en la base de datos.

Con un mecanismo similar se implementa la opción de recuperar la contraseña introduciendo el usuario el correo electrónico.

Durante la fase de conexión de usuario en la web, se ha implementado un sistema que, si la introducción de las credenciales es errónea, se procederá a bloquear al usuario X tiempo de acceder a la web. El problema que apareció fue dónde establecer ese tiempo de bloqueo, y se solventó creando una tabla de configuraciones en la base de datos, en la que se almacena un par variable-valor. Esta tabla, de hecho, será utilizada en más configuraciones que han surgido a lo largo de la implementación.

Otro de los puntos a solventar durante la implementación fue el hecho de que, al ser una revista, los redactores tendrán que escribir unos artículos con formato. La opción que se elige es que, en la base de datos se guarden las etiquetas HTML que den formato al texto del artículo que escriba el usuario de la aplicación web, pero, el usuario no tiene por qué conocer el lenguaje HTML, por ello, se piensa en incluir un módulo para Yii que nos permite incrustar un editor de textos con formato en un formulario. Este módulo se llama “yii2-redactor”, y se puede encontrar en <https://github.com/yiidoc/yii2-redactor>.

Finalmente, respecto al apartado de copias de seguridad, hemos elegido el módulo “yii2db-manager” (<https://github.com/Beaten-Sect0r/yii2-db-manager>), el cual solo puede ser accedido por el rol de administrador del sistema desde la aplicación web. Se ha modificado para que, además de la base de datos, también haga una copia de seguridad de todas las imágenes que van asociadas a las diferentes entradas que existen en la base de datos, como pueden ser las imágenes relacionadas con los artículos o categorías que se pueden encontrar en la aplicación web.

## 6. CONCLUSIONES Y LÍNEAS DE TRABAJO FUTURAS

### 6.1. CONCLUSIONES

En este proyecto se han puesto en práctica conocimientos que se han obtenido en distintas asignaturas de la carrera, y han aparecido diversos problemas que se han afrontado de la mejor manera posible. A mi parecer, ha servido como simulacro de una situación real en la vida laboral de un informático.

Además, a mi parecer, en el mundo de la informática, la formación ha de ser constante, y, este proyecto, ha servido como formación en la rama del desarrollo web.

### 6.2. LÍNEAS DE TRABAJO FUTURAS

Este trabajo en principio cumple con los objetivos que se establecieron a la hora de la propuesta, pero si es cierto que hay muchos aspectos que podría mejorarse, sobre todo si la web se hace pública:

- **Nombre corporativo:** Si la aplicación web se llega a publicar en internet, sería necesario establecer un nombre corporativo. Si bien es cierto que parece una tarea sencilla, no lo es en absoluto. Escoger un bueno nombre requiere estudio. El nombre de una marca debe diferenciarse de lo existente, tiene que ser sugerente y fácil de pronunciar. Además, hay que comprobar que el nombre este libre a la hora de registrar el dominio en la web.

- **Diseño:** El diseño de la web de cara al usuario puede mejorarse, ser más vistoso y accesible.
  - **Diseño de emails:** Crear una plantilla personalizada para el envío de emails, con el logotipo corporativo, por ejemplo.
- **Separación entre front-end y back-end:** Yii ofrece la posibilidad de separar front-end y back-end. Sería interesante ampliar la parte de administración del sistema y separarla del front-end.
- **Perfil de usuarios y funciones sociales:** En la aplicación web no se ha contemplado la interacción entre usuarios. Sería interesante la inclusión de un sistema de mensajería privada, o un foro de debate.
- **Integración con redes sociales:** Otro aspecto que quedó en el tintero fue la integración con redes sociales, como enlaces para compartir, o comentar los artículos con “Facebook” o similar.
- **Sistema de torneos online:** En principio se pensó en implantar un sistema de torneos en línea entre usuarios. Tras un estudio inicial, se descartó debido a que, para realizar torneos privados en ciertos videojuegos, es necesario pedir permiso al desarrollador, o, tener una cierta audiencia para poder obtener el beneplácito de la marca, pero si la web sale al mercado, es una opción muy atractiva para el usuario.
  - **Premios:** Implantando un sistema de torneos, también se podría considerar la inclusión de premios para los ganadores.

