**PROYECTO GESTOR EQUIPOS DE FUTBOL SALA:**

Realizar lo aprendido este año: MVC con usuarios y registro, e INCORPORAR: UTILIZAR LIBRERIAS DE PHP, nuevas opciones como METODOS DE PAGO,PAYPAL,IDIOMA,MENSAJERIA,MAS OPCIONES:

INTRODUCCION A JS: Videos

UTILIZACION DE SISTEMAS INFORMATICOS ENLAZANDO A AWS AMAZON WEB SERVICE BASE DE DATOS CON WINDOWS Y LINUX.

INGLES:

**LENGUAJE DE MARCAS:** html css+boostrab

**ENTORNOS DE DESARROLLO:**  diagramas y mockups+algoritmia

**BASES DE DATOS:** crear base de datos usuarios, equipos,noticias,

* 6.1.1. Diseño entidades En este apartado listamos todas las entidades que creamos en la base de datos representadas en forma de tabla. También se explican las relaciones que existen entre entidades y la función que realiza en el proyecto.
* Entidad user La primera entidad con la que nos encontramos, que es la entrada a nuestra aplicación, es la entidad user (ver Tabla 6.1) . Esta se encarga de almacenar los datos necesario para la autenticación dentro de nuestra aplicación.
* user Campo Tipo idPK int(20) AI40 usuario varchar(50) nombre(50) apellidos(100) e-mail(100) poner cookie varchar(255)
* Tabla 6.1 Entidad user.
* . Para terminar, al autenticarse en el sistema se crea una cookie42.
* Entidad user\_roles Una vez un usuario se ha autenticado, es la tabla user\_roles (ver Tabla 6.2) la que almacena qué rol ejercerá dentro de la aplicación. Dicho rol le otorgará unos privilegios u otros en función de cómo esté congurado.
* 40AutoIncrement o campo que se auto-incrementa cada vez que se inserta un nuevo elemento 41genera una fecha y hora automáticamente sin necesidad de programación alguna 42pequeña información enviada por un sitio web y almacenada en el navegador del usuario, de manera que el sitio web puede consultar la actividad previa del usuario
* 47
* user\_roles Campo Tipo user\_idPK int(20) FK role int(2)
* Tabla 6.2 Entidad user\_roles.
* Como se puede observar, se trata de una entidad muy sencilla. Lo único que hace es relacionar un usuario con los distintos roles que pueda ejercer. La lista de roles, dado que será difícil que cambien, se ha optado por almacenarla directamente en el código.
* Código 6.1 Roles almacenados en la clase Roles.
* <?php
* class Roles { const ROLE\_BASE = 9; const ROLE\_ADMIN = 1; const ROLE\_ENTRENADOR = 2; const ROLE\_EDITOR = 3; const ROLE\_RESULTADOS = 4; const ROLE\_PLANIFICADOR = 5;

Bases de datos

Una base de datos, o banco de datos, es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital, y por ende se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos. Existen programas denominados sistemas gestores de bases de datos, abreviado SGBD, que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos SGBD, así como su utilización y administración, se estudian dentro del ámbito de la informática. Las aplicaciones más usuales son para la gestión de empresas e instituciones públicas. También son ampliamente utilizadas en entornos científcos con el objeto de almacenar la información experimental. Existen diferentes tipos de bases de datos. Aquí trataremos solo las bases de datos relacionales; son las más utilizadas actualmente. En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia. Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante consultas que ofrecen una amplia flexibilidad y poder para administrar la información. El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, Structured Query Language o Lenguaje Estructurado de Consultas, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

4.2.1. MySQL Dentro de las bases de datos relacionales nos vamos a centrar en MySQL,que es la que usan el proyecto. MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con millones de instalaciones. MySQL AB -desde enero de 2008 una subsidiaria de Sun Microsystems y ésta, a su vez, de Oracle Corporation desde abril de 2009- desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Podríamos decir que casi cualquier hosting mundial posee este tipo de base de datos, es por ello que es de lo más utilizado para pequeños y medianos proyectos; e incluso grandes proyectos también hacen uso, como Facebook.

34

La mayoría de lenguajes de programación en la parte del servidor poseen librerías para interactuar con bases de datos MySQL. Esto se explica por lo anteriormente mencionado; cualquiera que precie su producto ha de permitir interoperatividad con este tipo de base de datos.

4.2.2. Entidad - Relación En el proyecto se utiliza un tipo de base de datos relacional: MySQL. Esto signica que tiene unos datos almacenados en estructuras (entidad) y estos se pueden relacionar (no es necesario) con otras entidades, (Relación). Para entender este concepto se representan dos entidades de la base de datos utilizada en el proyecto. La forma habitual de representar una entidad grácamente es en forma de tabla. Así que podemos simplicar diciendo que tenemos dos tablas:

Equipos Nombre Tipo id INT nombre Varchar diminutivo Varchar dia INT hora TIME id\_liga INT id\_temoporada INT nuestro TINYINT foto VARCHAR

Ligas Nombre Tipo id INT nombre Varchar grupo INT id\_categoria INT id\_temporada INT puntos\_ganar INT puntos\_empatar INT puntos\_perder INT mostrar TINYINT orden TINYINT

Tabla 4.1 Tablas de Equipos y Ligas.

Tal y como observamos en las tablas (Tablas 4.1), existe una relación entre la entidad Equipos y la entidad Ligas. Un equipo pertenece a una liga. La forma en que los relacionamos es añadiendo un campo en la estructura de los equipos que hace referencia a la liga a la que pertenecen: id\_liga.

4.2.3. Conguración de una entidad Como se ve en las tablas (Tablas 4.1), existe un campo que nos indica el tipo de dato para cada la. Cada la de la tabla es un dato que se almacenará dentro de la entidad. Dicho dato tiene una serie de atributos que hay que especicar:

1. Tipo de dato: para optimizar la forma de almacenar los datos se nos solicita qué tipo de dato va a almacenar. Así podemos tener diferentes tipos como: VARCHAR, INT, LONGINT, TINYINT, DATETIME... Tipos bastante genéricos y explicativos por su propio nombre.

2. Índice: debemos indicar si se trata de un índice. En cuyo caso el Sistema Gestor de Bases de Datos (en nuestro caso MySQL) ordenará por dicho elemento para optimizar las búsquedas.

35

3. Única: indicamos que no puede repetirse el valor dentro de la misma entidad para ese elemento. Por ejemplo, el campo e-mail para un registro de usuarios nos interesa que no esté repetido.

4. Clave primaria: o primary key (PK) en inglés. Es obligatoria y se trata del elemento (o unión de elementos) que identica de forma única a la totalidad de una la dentro de la entidad.

5. Otros: podemos indicar también que el elemento pueda o no ser nulo, que la PK se auto-incremente (típico ejemplo de generación de ID), el esquema predeterminado de los datos (ejemplo 0000-00-00 para fechas)...

--------------------------------------------------------------------------------------

Zona de administración (backend)

El diseño de la zona de administración, o backend, se realiza utilizando diagramas de casos de uso. Estos diagramas indican, dependiendo del usuario que trate, las posibles acciones que este puede realizar contra el sistema.

6.2.1. Usuario no autenticado Un usuario no autenticado es un usuario que no sabemos quién es. El primer paso antes de poder utilizar una aplicación, es decirle a la misma quién eres. Esta acción se describe en el siguiente diagrama (Figura 6.2):

55

Un usuario con rol administrador dispondrá de las acciones que se detallan en el siguiente diagrama (Figura 6.7):

**Figura 6.7 Diagrama rol administrador.**

Ver usuarios Ver un listado de las usuarios almacenados en la base de datos.

Flujo de eventos:

1. El usuario accede a la opción usuarios.

2. El sistema muestra la lista de los usuarios con botones de acción.

Asignar roles Asignar un rol o roles a un usuario en concreto.

Flujo de eventos:

1. El usuario accede a la opción usuarios.

2. El sistema muestra la lista de los usuarios con botones de acción.

3. El usuario elige la opción Asignar rol.

4. El usuario elige un rol de la lista.

5. El sistema asigna el rol al usuario.

Asignar equipo a entrenador Asignar equipo o equipos a un usuario con rol entrenador.

Flujo de eventos:

1. El usuario accede a la opción entrenadores.

2. El sistema muestra la lista de los usuarios con botones de acción.

3. El usuario elige la opción Asignar equipo.

4. El usuario elige un equipo de la lista.

5. El sistema asigna el equipo al entrenador.

Editar información parte pública Edita información de la parte pública.

Flujo de eventos:

1. El usuario accede a la opción información web.

2. El sistema muestra la lista de los campos de información.

3. El usuario elige la opción Editar.

4. El sistema actualiza la información.

Añadir avisos a usuarios Añade avisos a los usuarios dependiendo de su rol.

Flujo de eventos:

1. El usuario accede a la opción avisos.

2. El sistema muestra un formulario a rellenar.

3. El usuario rellena el formulario y lo envía.

4. El sistema actualiza los avisos.

Ver temporadas Ver en sección 6.2.5

Seleccionar temporada de trabajo Ver en sección 6.2.5

Añadir temporada Ver en sección 6.2.5

Editar temporada Ver en sección 6.2.5

Borrar temporada Ver en sección 6.2.5

Ver categorías Ver en sección 6.2.5

Añadir categoría Ver en sección 6.2.5

Editar categoría Ver en sección 6.2.5

Borrar categoría Ver en sección 6.2.5

Ver ligas Ver en sección 6.2.5

Añadir liga Ver en sección 6.2.5

Editar liga Ver en sección 6.2.5

Borrar liga Ver en sección 6.2.5

Ver equipos Ver en sección 6.2.5

Añadir equipo Ver en sección 6.2.5

Editar equipo Ver en sección 6.2.5

Borrar equipo Ver en sección 6.2.5

Asignar foto a equipo Ver en sección 6.2.5

Cambiar foto a equipo Ver en sección 6.2.5

Quitar foto a equipo Ver en sección 6.2.5

Añadir jornada Ver en sección 6.2.5

Borrar jornada Ver en sección 6.2.5

Ver resultados Ver en sección 6.2.4

Añadir resultados Ver en sección 6.2.4

Editar resultados Ver en sección 6.2.4

Ver clasicaciones Ver en sección 6.2.4

Ordenar clasicaciones Ver en sección 6.2.4

Salir

**Figura 6.2 Diagrama de usuario no autenticado.**

Como se puede apreciar, un usuario no autenticado solo dispone de dos acciones posibles:

Loguearse

Normal con registro

con Google Accede al sistema usando su cuenta de Google.

Flujo de eventos:

1. El usuario selecciona la opción de Google.

2. El sistema le redirecciona al sistema de autenticación de Google para que introduzca sus credenciales.

3. El sistema recoge las credenciales.

4. Si el usuario no se ha logueado nunca en el sistema, añade el usuario a la base de datos guardando su dirección de correo y el ID de Google.

5. Si el usuario ha marcado la opción de Recordarme, se genera una cookie en el navegador del usuario.

6. El sistema carga la página principal.

Flujos alternativos:

En el paso 4, si el usuario ya se ha logueado antes:

4. se recupera el rol que le ha asignado un administrador.

En el paso 5, si el usuario no ha marcado la opción Recordarme:

5. no se genera cookie y se elimina cualquier posible resto de alguna anterior.

Loguearse con Facebook Accede al sistema usando su cuenta de Facebook.

Flujo de eventos:

1. El usuario selecciona la opción de Facebook.

56

2. El sistema le redirecciona al sistema de autenticación de Facebook para que introduzca sus credenciales. 3. El sistema recoge las credenciales. 4. Si el usuario no se ha logueado nunca en el sistema, añade el usuario a la base de datos guardando su dirección de correo y el ID de Facebook. 5. Si el usuario ha marcado la opción de Recordarme, se genera una cookie en el navegador del usuario. 6. El sistema carga la página principal. Flujos alternativos:

En el paso 4, si el usuario ya se ha logueado antes:

4. se recupera el rol que le ha asignado un administrador.

En el paso 5, si el usuario no ha marcado la opción Recordarme:

5. no se genera cookie y se elimina cualquier posible resto de alguna anterior.

Un usuario con rol **entrenador** dispondrá de las acciones que se detallan en el siguiente diagrama (Figura 6.4):

Figura 6.4 Diagrama rol entrenador

Añadir resultados Añadir resultados de partidos ya disputados.

Flujo de eventos:

1. El usuario accede a la opción resultados.

2. El sistema muestra las ligas a las que tiene acceso.

3. El usuario elige la liga en la que quiere añadir resultados.

4. El sistema le muestra la última jornada a la que le faltan los resultados.

5. El usuario inserta los resultados.

6. El usuario le da al botón guardar.

7. El sistema guarda los resultados.

8. El sistema calcula las clasicaciones.

9. El sistema pasa a la siguiente jornada.

Flujos alternativos:

En el paso 5, si el usuario no quisiera esa jornada:

61

5. El usuario elige la jornada que desea.

6. El usuario inserta los resultados.

7. El usuario le da al botón guardar.

8. El sistema guarda los resultados.

9. El sistema calcula las clasicaciones.

10. El sistema pasa a la siguiente jornada.

Añadir jornadas Añadir jornadas para la confección del calendario deportivo.

Flujo de eventos:

1. El usuario accede a la opción jornadas.

2. El sistema muestra las ligas a las que tiene acceso.

3. El usuario elige la liga en la que quiere añadir jornadas.

4. El sistema le muestra la última jornada a la que le faltan los enfrentamientos.

5. El usuario inserta los enfrentamientos.

6. El usuario pone las fechas del partido de ida y vuelta.

7. El usuario le da al botón guardar.

8. El sistema guarda los resultados.

9. El sistema pasa a la siguiente jornada.

Salir

**PROGRAMACION:**  lo primero hacer loggin y registro+implementar

**SISTEMAS INFORMATICOS:**

PROYECTO:

* Montando la Base de datos con AWS:

Adapta tu aplicación que estas desarrollando para el proyecto integrado para que toda la infraestructura que utilizas de base de datos este en AWS. Para ello, tendrás que configurar una base de datos en AWS, y puedes hacerlo de la forma que prefieras según los vıdeos: configurando una instancia LAMP o utilizando el servicio RDS, pero tendrás que argumentar tu eleccion. Para acceder a la base de datos puedes utilizar la consola de MySQL o hacerlo mediante PHPMyadmin. Una vez montada la infraestructura, si ya tienes datos en tu copia local, puedes importarlos a tu base de datos en AWS. Se entregar´a el codigo fuente de la aplicacion.

VIDEOS:

* **Conexión a MySQL desplegado en EC2**
* Acceso a PhpMyAdmin desplegado en EC2
* Conexión desde PHP a MySQL desplegado en EC2
* Servicio RDS
* Exportaci´on e importación en MySQL

Referencias:

<https://aws.amazon.com/es/rds>

Documentación oficial de RDS

<http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.RDSSecurityGroups.html>

Grupos de seguridad con RDS

<https://aws.amazon.com/documentation/rds/>

mysqldump

<https://dev.mysql.com/doc/refman/5.7/en/mysqldump.html>

1-CONEXIÓN MAQUINA LAMP DESPLEGADA EN EC2 con AWS AMAZON WEB SERVICE:

* Iniciamos Instancias maquina ec2
* Conectamos con putty-putty-gen o virtualbox Linux

Accedemos: documentación bitnami

* **Desde / cd /opt/bitnami/mysql**

**Cd bin** (están los ejecutables)

(conectamos con comandos) **mysql -u root -p**

(elegimos contrseña cn) ./bnconfig –userpassword a12345678

**mysql -u root -p**

**a12345678 estamos dentro: usar base de datos ,crear…**

* 2- Acceso a PhpMyAdmin desplegado en EC2

miercoles

TRABAJO:

1:SHELL SCRYPTS: hacer

2:Windows Server: si se puede ver….

2:Hecho **Amazon Web Service**

Desde AWS, montar maquina Linux y conectar con ssh

De L a L🡪 ssh -i

De W a L🡪 Putty

AWS: requena7

1-Trabajar con la consola de administración de AWS.

2- Utilizar el servicio EC2 para el aprovisionamiento de máquinas virtuales.

3- Conectarse a instancias EC2 basadas en **Linux y en Windows**

Ya tenemos cuenta aws con 100 $ 🡪 Despliega una instancia LAMP=Linux, Apache, MySQL y PHP…

Desplegar máquinas virtuales🡪 AMI amazon machine image que ya exista,

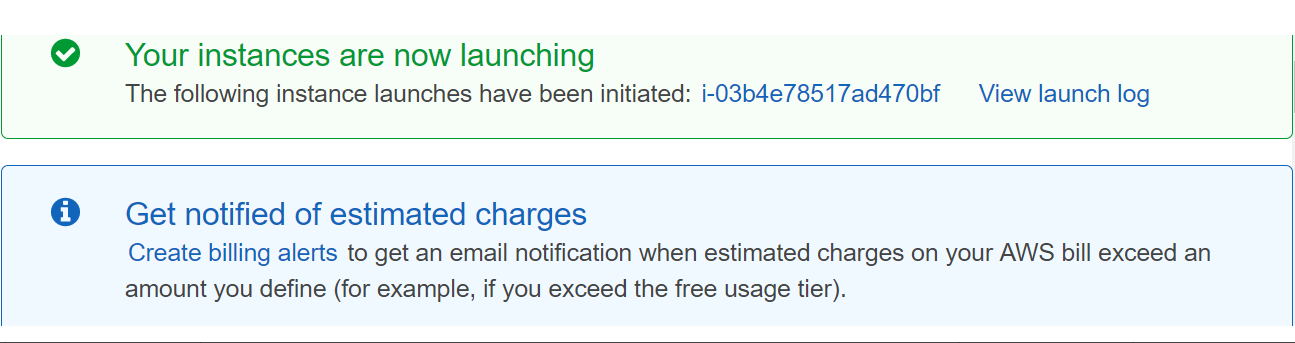
1. **Construir un par de claves:** UNA Pública y otra privada para poder entrar en nuestra maquina con ssh si poner contraseña
2. **Acceder al servicio EC2**
3. Menú global, lateral y vamos a **KEY PAIRS, crea clave** la CLAVE PUBLICA EN REMOTO, LA PRIVADA EN NUESTRA MAQUINA……… si\_guillermo.pem privada
4. **SELECCIONA GRUPO DE SEGURIDAD:**

Es una configuración de cortafuegos sobre un grupo de instancias

5: DESPLEGAR UNA AMI EXISTENTE:

* AMI BASDA EN S3…. Los cambios q hagas se acaban al apagar maquina
* AMI BASADA EN EBS…volúmenes-cambos se mantienen incluso maquina apagada

5: seleccionamos la maquina con ebs tt.micro y continuamos le pasamos key pairs cread+puertos



Then ec2🡪 wiew instances maquina-servicios corriendo

Ver descripción-> importante IP Publica y los DNS… IPV4 normal

**CONNECT: SALE LOS PASOS PARA PODER CONECTAR POR SSH DESDE OTRA MAQUINA**

**Ya viene conexión con shh🡪 pongo stop maquina…**

Descargar clave.pem para hacer ssh desde Ubuntu cambiamos permisos a .pem y lanzamos el ssh -i mas la ruta que pone en connect.

**DESDE WINDOWS DESCARGAS PUTTYGEN Y LUEGOPUTTY**

BUSCAS LA IPV4 PUBLICA DE LA MAQUINA

GENERAMOS EN PUTTYGEN🡪 archivo keyssh.pkk y cargamos en el putty normal en ssh,

Y en user pasamos la dirección ip de la mamquina

-clave publica y privada para poder conectar sin poner contraseña

Clave prvada almacena en maquina local y copias clave publica desde otra maquina

En menú ECS--🡪 KEY PEIRS GENERAS CLAVE PUBLICA Y RIVADA Y SE DESCARGA va a descargas y la a de tener la clave remote en nuestra maquina publica en la otra

Si\_guillermo.pem

Se inyectara a la maquina??

2 CREAR GRUPO DE SEGURIDAD CON SEGURITO GROUP

Se pone a una instancia y hace de cirtafuegos asociar cortafuegos

SE VA A SEGURITY GROUP

ES UNA CONFIGURACION DE CORTAFUEGOS,,, VAS A CREATE SECURITY GROUP

Y ELIGES SSH,,,+ IP T2 CREAS Y TE SALE CON EL NOMBRE🡪DESCRIPCION EN SEGURITY GROUP pudes seleccionar azul y abajo menú para editar puertos

3- SELECCIONAR DESPLEGAR LA MAQUINA EC2 AMIS basadas en s3 instancias ec2 se borra todo almacenamiento temporal otr🡪basadas en EBS permanente

Seleccionar maquina en INSTANCES LAUNCH ISTANCE sea free trieer 12 mses

