

Proyecto fin de curso

Lolinfo



Índice

Introducción	3
Plan de empresa y presupuesto	4
Base de datos	5
Funcionamiento	7
Código Android	15
Bibliografía	16

Introducción

-Idea inicial

Hemos decidido llamar a nuestro proyecto "**Lolinfo**" porque va sobre el videojuego "League of Legends" y la función que tiene es dar información sobre los próximos partidos de liga, la clasificación actual, los equipos y sus jugadores.

-Planteamiento del proyecto

El videojuego "League of Legends", conocido como Lol, tiene más de 70 millones de jugadores registrados y es el más jugado del mundo. Viendo el calado que tiene entre gente joven (14-25 años), los cuales tienen en su mayoría un teléfono móvil con Android, y que no hay ninguna aplicación de este estilo ni que se le parezca, decidimos crear una nosotros.

-Propósito del proyecto

El propósito de este proyecto es desarrollar una aplicación útil, que podamos vender a "Riot Games", empresa propietaria de este videojuego.

-Responsable del proyecto

El responsable de este proyecto es Carlos Contreras.

-Herramientas del proyecto

Para desarrollar este proyecto, se necesita un programa para escribir el código de la aplicación (Android Studio) y un dominio en internet que pueda gestionar bases de datos (cuenta creada en Hostinger).

-Requisitos

Para el uso de esta aplicación solo hará falta tener un dispositivo con Android, puesto que se podrá descargar de la PlayStore (tienda de aplicaciones de Android de Google).

-Copias de seguridad

Hemos trabajado siempre con Dropbox, para que los archivos siempre estén en la nube y se pueda acceder a ellos desde cualquier equipo.

Plan de empresa y presupuesto

-Plan de empresa

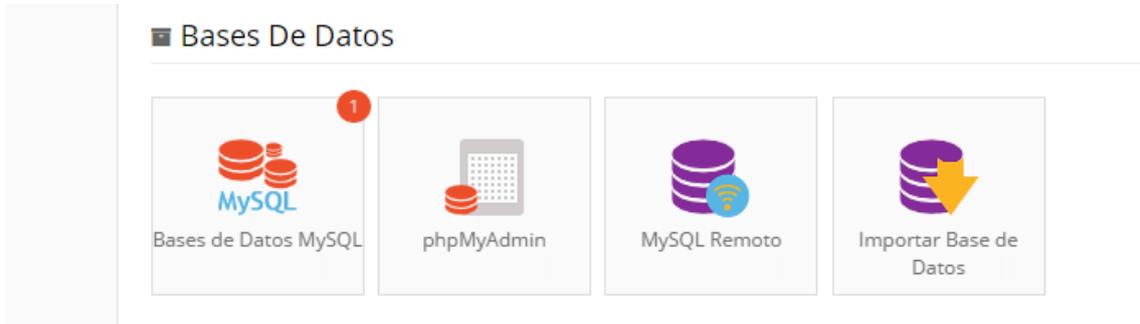
Nuestro plan de empresa es que cuando tengamos la aplicación funcional, contactar con Riot Games e intentar venderles nuestro proyecto, aunque aparte de eso intentaremos que nos mantengan como equipo de soporte y desarrollo de la aplicación una vez la hayamos vendido.

-Presupuesto

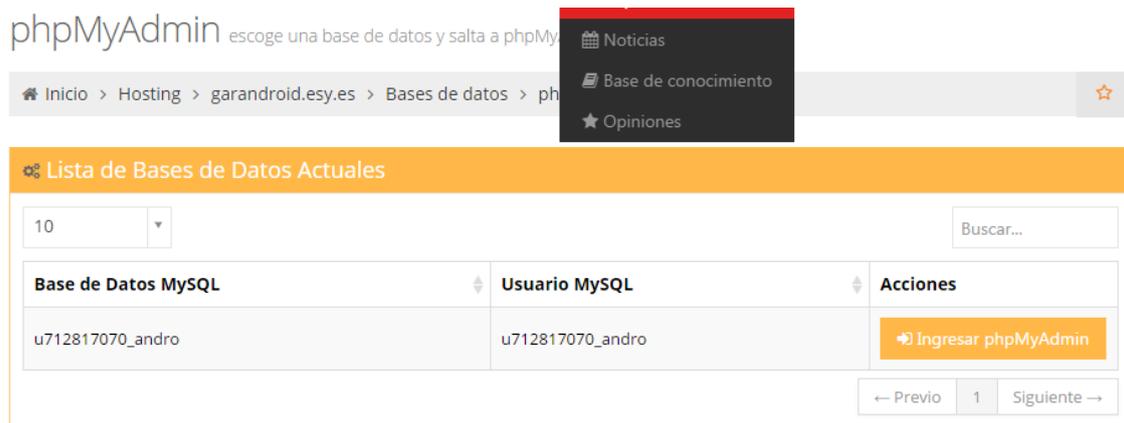
Nuestro presupuesto para hacer esto ha sido de 0€, pues todos los programas que hemos usado son gratuitos. Sin embargo, a la hora de vender la aplicación, hemos calculado que hemos dedicado unas 100 horas a la realización del desarrollo, por lo que esas horas irán incluidas en el precio de venta de la aplicación.

Base de datos

La base de datos la hemos gestionado a través de phpmyadmin en nuestra cuenta de Hostinger.



Hemos creado una base de datos.



Hemos creado tres tablas, una por layout.



Clasificación:

- equipo : Nombre del equipo, primary key.
- ganadas: Partidas ganadas.
- perdidas: Partidas perdidas.

Equipos:

- idequipo : Id del equipo, primary key.
- nequipo : Nombre del equipo.
- sequipo : Siglas equipo.
- csuperior : Jugador del top.
- jungla : Jugador de la jungla.
- ccentral : Jugador del medio.
- adcarry : Jugador de adc.
- apoyo : Jugador de apoyo.
- imgurl : Url de la imagen del equipo.

Partidos:

- idpartido : Id del partido, primary, autoincrement.
- fecha : Fecha del partido.
- hora : Hora del partido.
- equipo1 : Nombre del equipo 1.
- equipo2 : Nombre del equipo 2.
- gano : nombre del equipo ganador.
- perdio : nombre del equipo perdedor.

Para introducir nuevos partidos, equipos y demás, sólo hay que hacer inserts o updates.

Funcionamiento

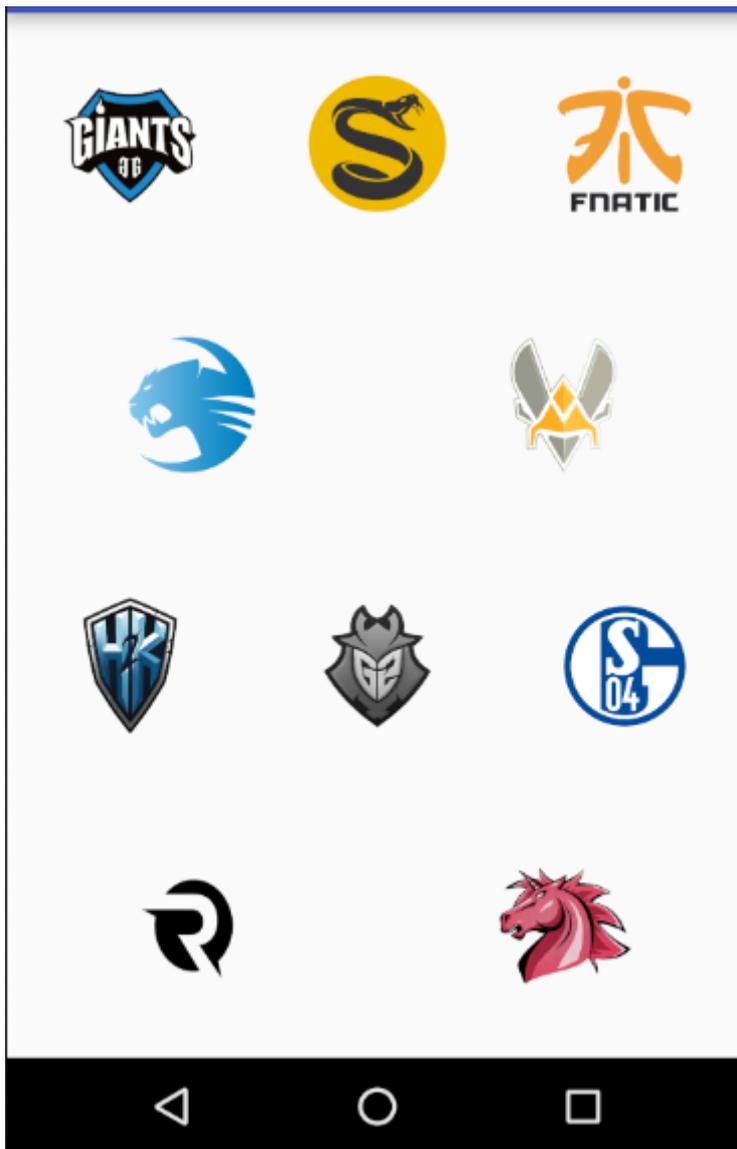
Cuando abrimos la aplicación, esta es la pantalla de inicio:



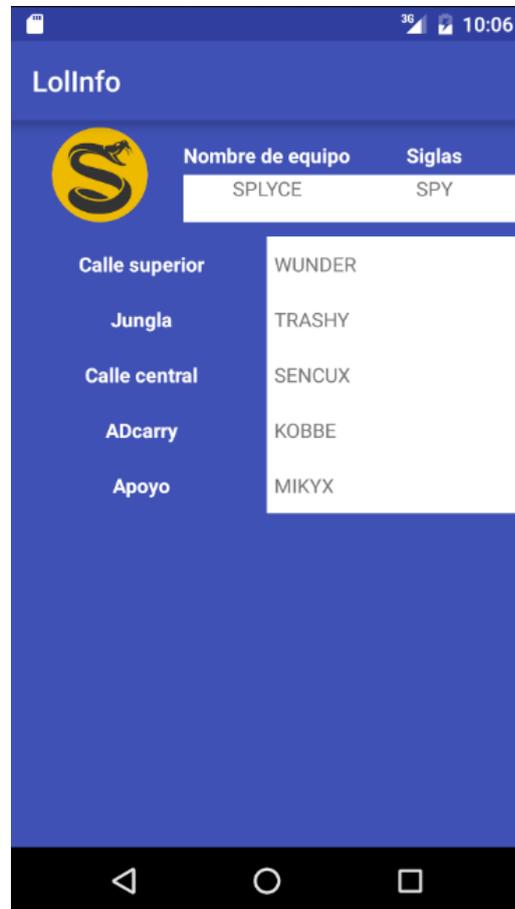
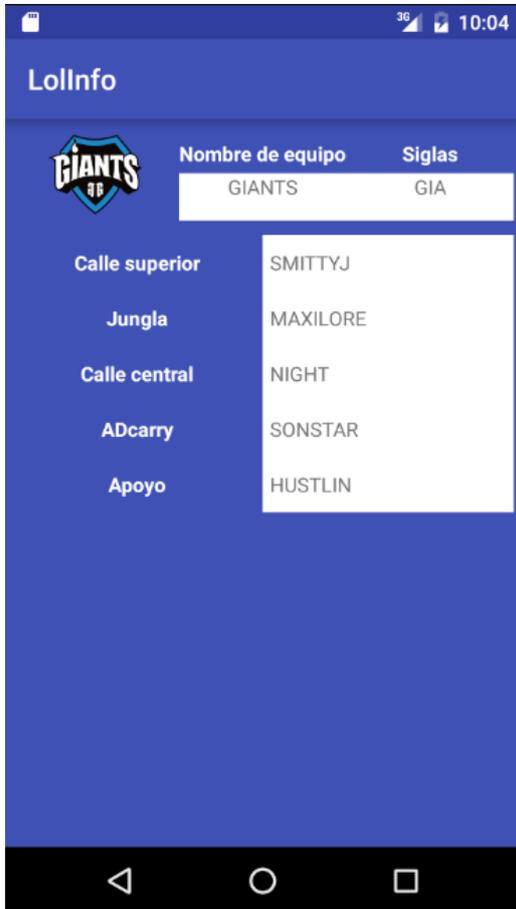
Tenemos 3 partes diferenciadas, equipos, partidos y clasificación.

Equipos:

Al entrar en equipos, nos salen el logo de cada equipo, al pulsar sobre cualquiera de ellos nos lleva a una página de información sobre el equipo.



Ejemplo:



Partidos:

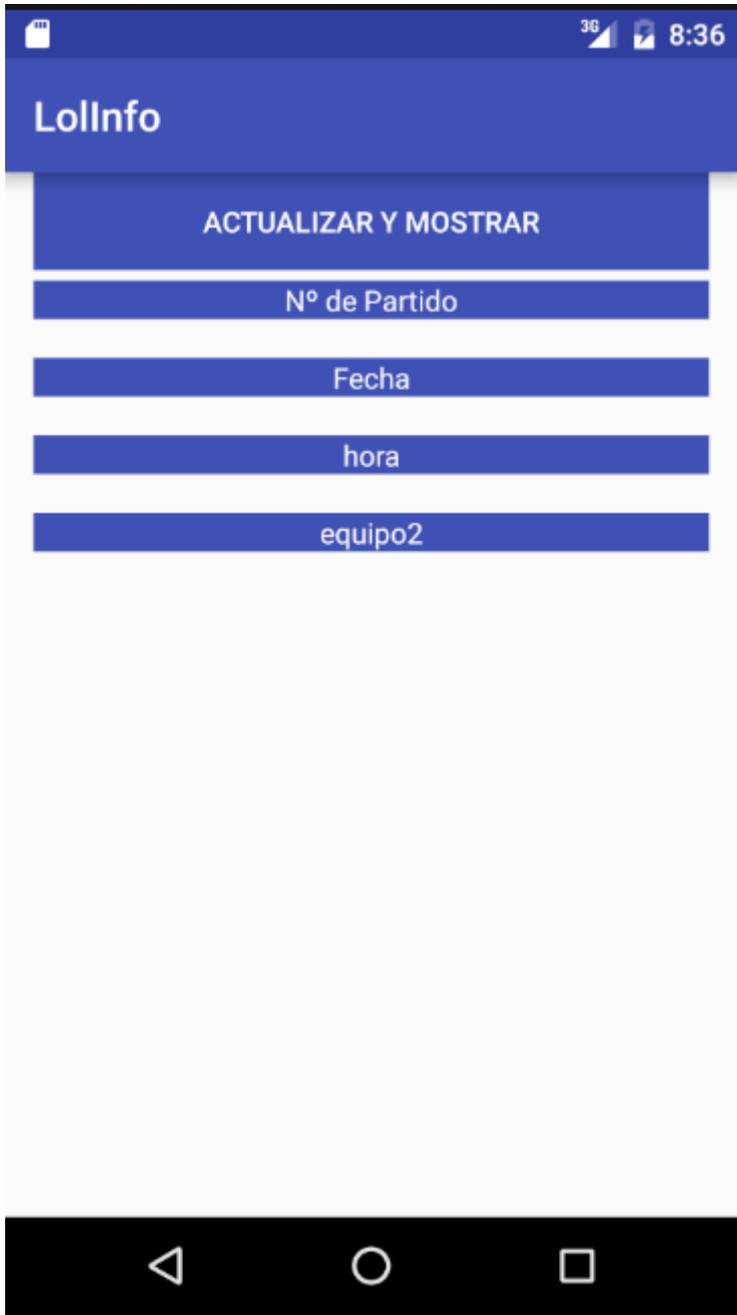
Al entrar en partidos, tenemos 3 opciones, los próximos partidos, los que se juegan hoy, o los ya jugados.



Próximos:

fecha	equipo vs.	equipo
2016-06-09	 VIT	 OG
2016-06-09	 H2K	 GIA
2016-06-09	 ROC	 G2
2016-06-09	 FNC	 UOL
2016-06-09	 SPY	 S04
2016-06-10	 VIT	 ROC

Hoy:



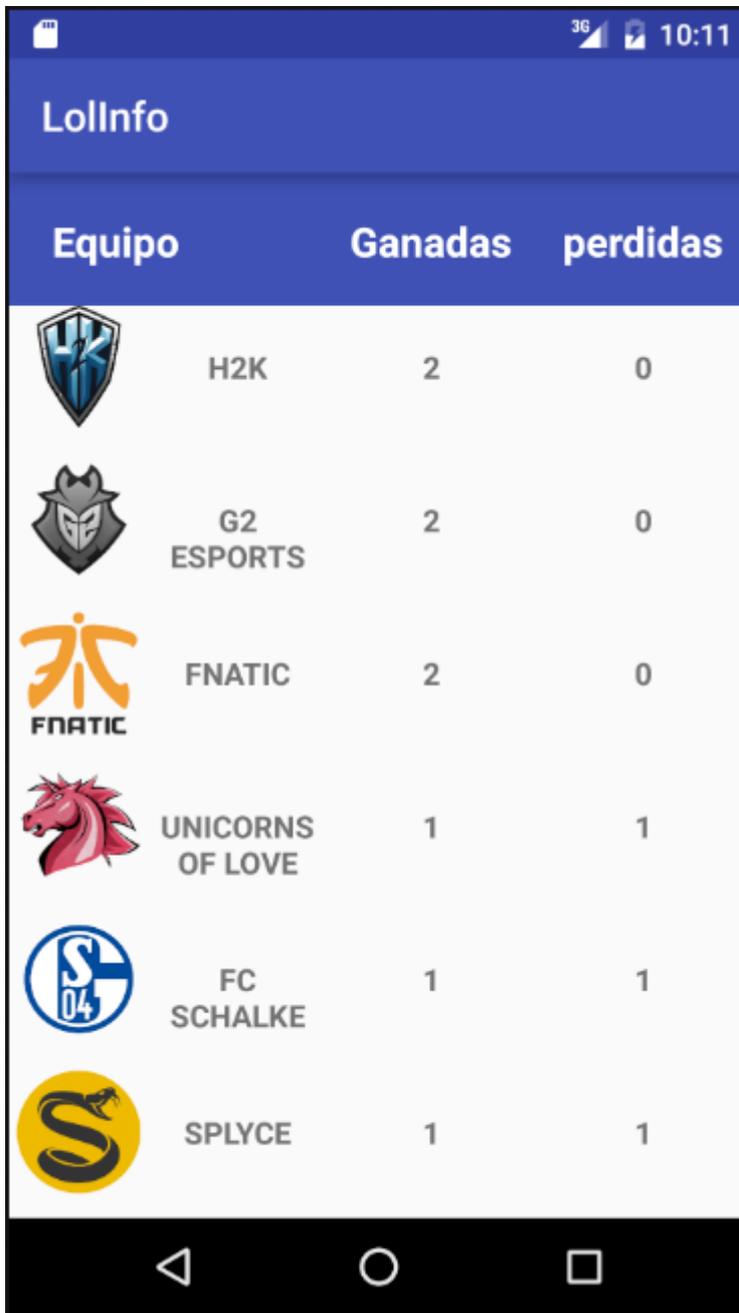
Jugados:

fecha		gano		perdio
2016-06-02		VIT		SPY
2016-06-02		S04		UOL
2016-06-02		G2		OG
2016-06-02		H2K		ROC
2016-06-02		FNC		GIA
2016-06-03		SPY		ROC

Sale cada partido con la fecha al lado.

Clasificación:

Al dar a clasificación, nos lleva a una pantalla donde salen los equipos, con sus correspondientes victorias y derrotas.

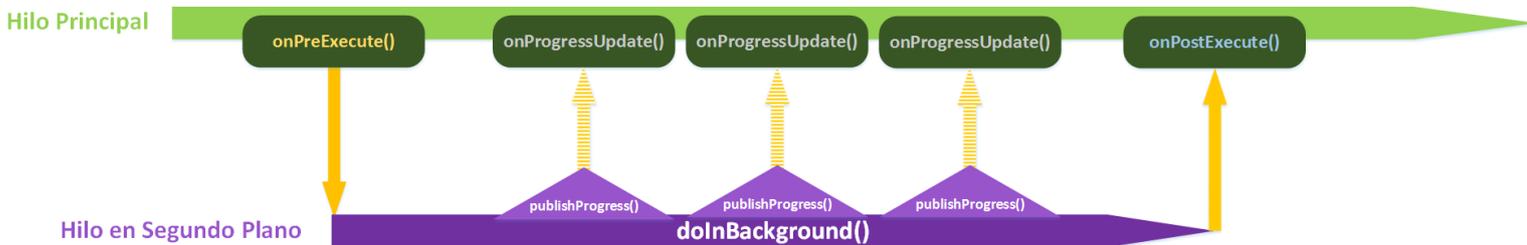


Equipo	Ganadas	perdidas
 H2K	2	0
 G2 ESPORTS	2	0
 FNATIC	2	0
 UNICORNS OF LOVE	1	1
 FC SCHALKE	1	1
 SPLYCE	1	1

Código Android

Async Task

Ciclo de vida de la clase AsyncTask.



Trabajar con el hilo principal solo para la interfaz gráfica y el resto en segundo plano
Los métodos usados son:

onPreExecute() que se ejecuta sobre el hilo principal. Al terminar este AsyncTask crea un hilo secundario y ejecuta su trabajo dentro de **doInBackground()**.

doInBackground(). Durante la ejecución en segundo plano podemos realizar llamadas al hilo principal (Barras de carga.) con ayuda de `publishProgress()`, a un método sobrescrito llamado **onProgressUpdate()**

onProgressUpdate(), el método `publishProgress()` no es obligatorio llamarlo, pudiendo no llamarlo nunca con lo que nunca entraremos en `onProgressUpdate()`. Al terminar de ejecutarse `doInBackground()` se llama de inmediato a **onPostExecute()**,

onPostExecute() se da por acabado a este hilo en segundo plano.

En nuestra aplicación funciona de la siguiente forma:

1. **onPreExecute**: preparamos las variables y el recyclerView donde se mostraran los datos.
2. **doInBackground**: llamamos a la página web que nos devolverá el resultado en JSON.
3. **onProgressUpdate**: se va añadiendo cada valor ya convertido de JSON a texto a el Array y lo pasamos a la actividad principal.
4. **onPostExecute**: cerramos el proceso..

Bibliografía

<http://www.w3schools.com>

<http://www.sgoliver.net/blog/curso-de-programacion-android/indice-de-contenidos/>

d-codepages.com

<https://www.simplifiedcoding.net/>

<http://www.kosalgeek.com/>

Apuntes facilitados por el tutor.