

# **Aplicación Android: FavSeries**

Gabriel Yangüez Parra

CFGS: DAM

IES Serpis

*28 de mayo de 2013*

***Agradecimientos:***

## ***Indice:***

***1.Justificación: ¿Por qué se pone en marcha el proyecto?***

***2.Formulación de objetivos: ¿Qué se pretende conseguir?***

***3.Desarrollo (sin acabar)***

***4.Tecnología utilizada.***

***5.Justificación económica.***

***6.Prevenición de riesgos laborales (sin acabar)***

***7.Evaluación y conclusiones***

***8.Anexo (sin acabar)***

***9.Referencias y Bibliografía.***

***10.Link al proyecto FavSeries***

## **1. Justificación:**

Este proyecto se pone en marcha con el objetivo de finalizar los estudios de Desarrollo de Aplicaciones Multiplataforma. Durante este curso, en el módulo de Programación Multimedia y dispositivos Móviles, me di cuenta que Android es un mundo muy grande y que está cada vez más presente en lo que son Smartphones, Tablets, etc.

El mundo de los ordenadores, móviles y demás siempre me ha atraído y es por eso que este año al estudiar dicho módulo quise aprender y documentarme más y finalmente llevar acabo una Aplicación en Android.

La razón por la que me decidí a hacer la Aplicación FavSeries es porque desde hace 4 o 5 años veo continuamente bastantes series de televisión, pero vía internet y subtituladas en versión original, lo que me hace tener un bloc de notas constantemente para poder recordar por qué temporada y capítulo voy exactamente, que días son lanzados en dicho país el episodio...y gracias a dicha Aplicación, puedo llevar ese listado de manera más dinámica, sin tener un documento constantemente actualizado y además de manera más visual y con el contenido de la Sinopsis de cada serie, etc, y todo ello en un dispositivo móvil el cual puedo llevar a cualquier parte conmigo y llevar también dicha información. De esta manera recuerdo siempre que última temporada y episodio he visto y además saber de qué trata la serie que me interesa o leyendo la sinopsis de una serie que nunca haya visto y así plantearme el ver dicha serie.

## **2. Formulación de objetivos:**

Con la Aplicación FavSeries se pretende conseguir llevar al día las temporadas y series que más le gustan al usuario y así tener una lista actualizada de dicha información sin tener que acceder siquiera al ordenador para consultar en internet sobre la serie. Los objetivos a conseguir eran los de tener una Base de Datos que pudiese ser consultada por la Aplicación y que en esa Base de Datos se almacenases las sinopsis de series, las temporadas y los capítulos y luego en ella se guardasen los datos introducidos por el usuario en referencia a la temporada y capítulo que ya ha visionado de cada serie. Finalmente no ha sido así el desarrollo de la aplicación pero en el apartado Evaluación y conclusiones se especifica más todo.

## **3. Desarrollo**

Las fases por las que he pasado para llevar a cabo dicho proyecto son las siguientes:

### 3.1 Aceptación:

Esta fase la considero la primera de todas, ya que al principio, como es normal, estuve dándole mil y una vueltas a la cabeza con ideas como “no sé si será la idea de proyecto adecuada”, “igual era mejor un proyecto relacionado con páginas web”, etc.

Al principio pensé en una web, pero como tampoco sabía muy bien de que hacerla debido a que ya prácticamente hay webs de todo tipo, me decanté por Android.

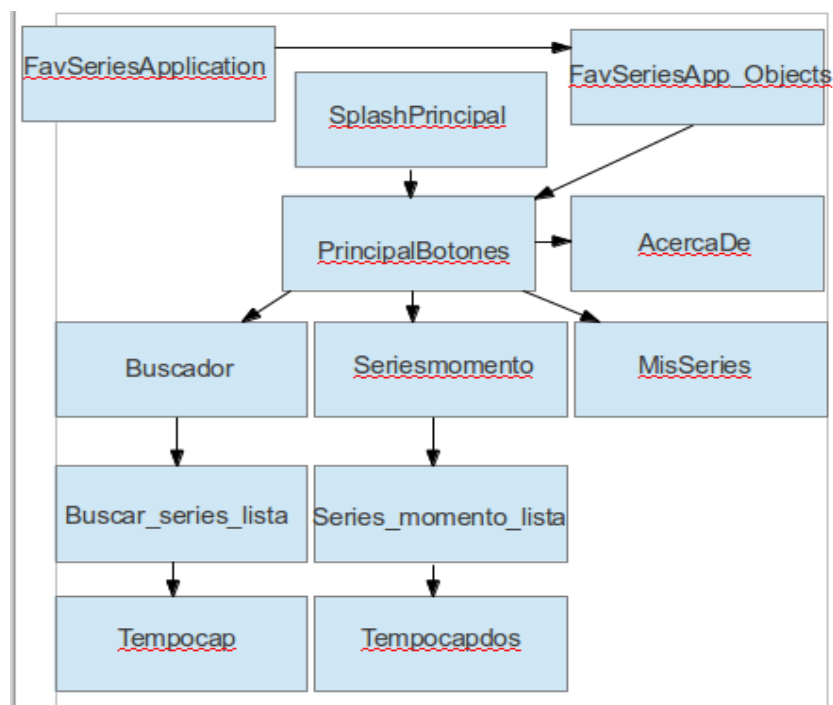
Una vez me di cuenta que de verdad quería llevar a cabo este proyecto y también conseguir superar mis objetivos llevé a cabo la fase de Estructura.

### 3.2 Estructura del proyecto y de la aplicación:

Lo primero que hice fue coger lápiz y papel y pasarme los primeros días y semanas en diseñar lo que sería cada parte de la aplicación, que vienen a ser las Activitys, Layouts, comportamiento de cada botón, la estética que quería para el proyecto, etc.

Luego simplemente era llevar acabo lo diseñado en papel al lenguaje de programación Android mediante Eclipse y a empezar a ver como dicho proyecto iba cogiendo forma según lo planeado.

A continuación se explican las clases de la app y su función en ella:



## Clases de la Aplicación FavSeries:

1.FavSeriesApplication: En esta clase establezco los Strings de Nombre, Sinopsis, los temporadas y capitulos, y el Int de Banner para cargar las imágenes en la ficha de cada serie.

```
public class FavSeriesApplication implements Parcelable{
    String nombre;
    String sinopsis;
    int banner;
    String[] spinnertemporadas;
    String[] spinnercapitulos;

    public FavSeriesApplication(String nombre, String sinopsis, int banner,
        String[] spinnertemporadas, String[] spinnercapitulos) {
        super();
        this.nombre = nombre;
        this.sinopsis = sinopsis;
        this.banner = banner;
        this.spinnertemporadas = spinnertemporadas;
        this.spinnercapitulos = spinnercapitulos;
    }
}
```

2.FavSeriesApplication\_Objects: En esta clase tengo dos arrays de objetos, uno para las series de la parte buscador y otro para las series de la parte de series del momento.

```
public class FavSeriesApplication_Objects {

    public static FavSeriesApplication[] arrayObjetos(Context contexto){
        FavSeriesApplication array[] = {
            new FavSeriesApplication(contexto.getString(R.string.anatomiaGrey),
                contexto.getString(R.string.anatomiaGreySinopsis),
                R.drawable.anatomiagreybanner,
                new String[]{"Temporada 1", "Temporada 2", "Temporada 3", "T
                Temporada 6", "Temporada 7", "Temporada 8", "Temporada 9", "Tem
                Temporada 12", "Temporada 13", "Temporada 14", "Temporada 15",
                Temporada 18", "Temporada 19", "Temporada 20", "Temporada 21",
                Temporada 24", "Temporada 25"},
                new String[]{"Capitulo 1", "Capitulo 2", "Capitulo 3", "Capi
                Capitulo 6", "Capitulo 7", "Capitulo 8", "Capitulo 9", "Capitulo
        }
    }
}
```

3.SplashPrincipal: Esta es la pantalla principal de presentación que después de 10 segundos te lleva a la pantalla PrincipalBotones para empezar a usar la aplicación.

```
public class SplashPrincipal extends Activity {
    private static final int SPLASH_TIME = 5 * 2000;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.splash);
        try {
            new Handler().postDelayed(new Runnable() {
                @Override
                public void run() {
                    startActivity(new Intent(SplashPrincipal.this,
                        PrincipalBotones.class));
                    finish();
                }
            }, SPLASH_TIME);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

4.PrincipalBotones: Aquí tengo los 3 botones Buscar serie, Series momento y Mis series.

```
public class Principalbotones extends Activity {  
  
    private Button bn1,bn2,bn3;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.principal2);  
  
        bn1 = (Button) findViewById(R.id.button1);  
        bn2 = (Button) findViewById(R.id.button2);  
        bn3 = (Button) findViewById(R.id.button3);  
    }  
}
```

5.AcercaDe: Explico básicamente lo que hace la aplicación.

```
public class AcercaDe extends Principalbotones {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.acercade);  
    }  
}
```

6.Buscador: Te muestra una lista con el nombre de las series cogiendo el String de Nombres de las series.

```
public class Buscador extends Activity {  
  
    private ListView listaSeries;  
    private EditText buscador;  
    AdaptadorLista adaptador;  
    String intentString;  
    String buscar_series;  
  
    FavSeriesApplication arraySeries[];  
    private final String BUNDLE = "Bundle";  
}
```

7.Series momento: Te muestra una lista de las 10 series más exitosas pero esta vez cogiendo el String de Banner para así mostrar imágenes en vez de texto y darle un mejor aspecto a esta sección.

```
public class Seriesmomento extends Activity {  
    private ListView listaSeries2;  
    AdaptadorLista2 adaptador2;  
    String intentString;  
    FavSeriesApplication arraySeries2[];  
    private final String BUNDLE2 = "Bundle2";  
}
```



8.Mis Series: En esta Activity se podrá en un futuro visualizar la serie añadida previamente una vez se accede a la ficha de la serie.

9.Buscar\_serie\_lista: Aquí tengo 3 botones, un ImageView y 2 TextView, uno para el Titulo de la serie y otro para la sinopsis a mostrar.

```
public class Buscar_serie_lista extends Activity{  
    Button bn9,bn10,bn16;  
    ImageView feed_image;  
    private final String BUNDLE = "Bundle";  
    private final String BUNDLE_TEMPcap = "BundleTempCap";  
    FavSeriesApplication objeto;  
    ..  
}
```

10.Series\_momento\_lista: El mismo contenido que en la Activity Buscar\_serie\_lista.

```
public class Series_momento_lista extends Activity {  
    Button bn11,bn12,bn14;  
    ImageView feed_image2;  
    private final String BUNDLE2 = "Bundle2";  
    private final String BUNDLE_TEMPcap2 = "BundleTempCap2";  
    FavSeriesApplication objeto2;  
    ..  
}
```

11.Tempocap: Contiene 2 Spinners, uno para las Temporadas y otro para los capítulos de la parte de Buscar Series. Contiene también un botón guardar y los SharedPreferences correspondientes para guardar la posición del spinner y que al entrar de nuevo en la aplicación, la posición del Spinner siga en la misma que al guardar.

```
public class Tempocap extends Activity {  
    private Button bn15;  
    private Spinner s1,s2;  
    private SharedPreferences prefs,prefs1;  
    private String prefName = "spinner_value";  
    private String prefName2 = "spinner_value2";  
    int id=0;  
    int id2=0;  
    void showToast(CharSequence msg) {  
        Toast.makeText(this, msg, Toast.LENGTH_SHORT).show();  
    }  
    private final String BUNDLE_TEMPcap = "BundleTempCap";  
    FavSeriesApplication objeto;  
    ..  
}
```

12.Tempocapdos: Contiene 2 Spinners, uno para las Temporadas y otro para los capítulos de la parte de Series momento. Como en Tempocap, contiene también un botón guardar y los SharedPreferences correspondientes para guardar la posición del spinner y que al entrar de nuevo en la aplicación, la posición del Spinner siga en la misma que al guardar.

```
public class Tempocapdos extends Activity {  
  
    private Button bn18;  
    Spinner s3,s4;  
    private SharedPreferences prefs2,prefs3;  
    private String prefName3 = "spinner_value3";  
    private String prefName4 = "spinner_value4";  
    int id3=0;  
    int id4=0;  
    void showToast(CharSequence msg) {  
        Toast.makeText(this, msg, Toast.LENGTH_SHORT).show();  
    }  
  
    private final String BUNDLE_TEMP2 = "BundleTempCap2";  
    FavSeriesApplication objeto2;
```

### 3.3 Recopilación de información:

Esta fase ha llevado bastante trabajo, ya que en un principio quería empezar con unas 15 series para que en el proyecto se viese reflejado que los objetivos prometidos se podían cumplir. Pero acabé dándome cuenta que ya que hacía el proyecto, lo quería realizar con un buen contenido así que fueron también muchos días y semanas de buscar por Google y por muchísimas Webs información de las temporadas, imágenes para cada una de ellas, las sinopsis,etc. Finalmente acabé con un total de 35 series en el buscador global y 10 series más vistas basándome en un ranking de internet.

### 3.4 Fase de Testeo:

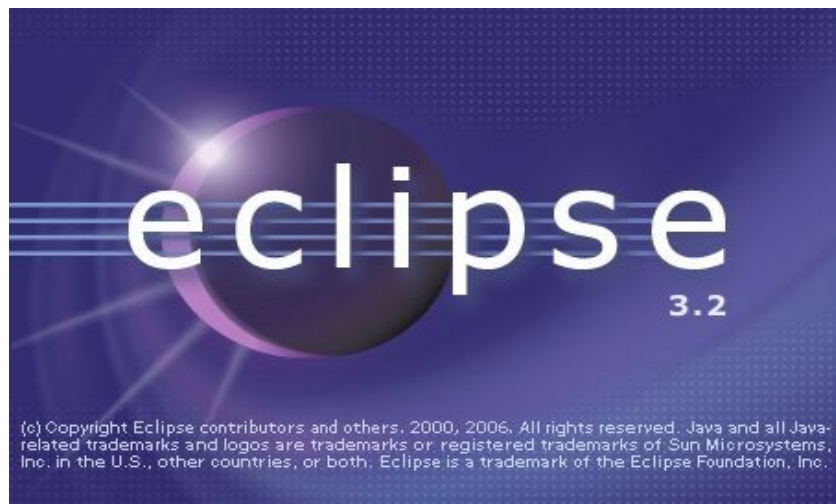
Esta fase también ha sido complicada debido a que realicé durante bastante tiempo numerosas pruebas de testeo de la Aplicación instalándola en varias versiones de Android (2.1, 3.2, 4.2), de diferentes marcas de teléfonos móviles, de Tablets para ver si el diseño seguía por el buen camino o se visualizaba con problemas en el diseño y en funcionalidad, de si cada botón o acción se comportaba correctamente a su funcionalidad diseñada con anterioridad,etc.

#### **4. Tecnología utilizada.**

Para el desarrollo de mi proyecto Android he usado las siguientes tecnologías:

**[1] Eclipse:** Eclipse es una potente y completa plataforma de programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java.

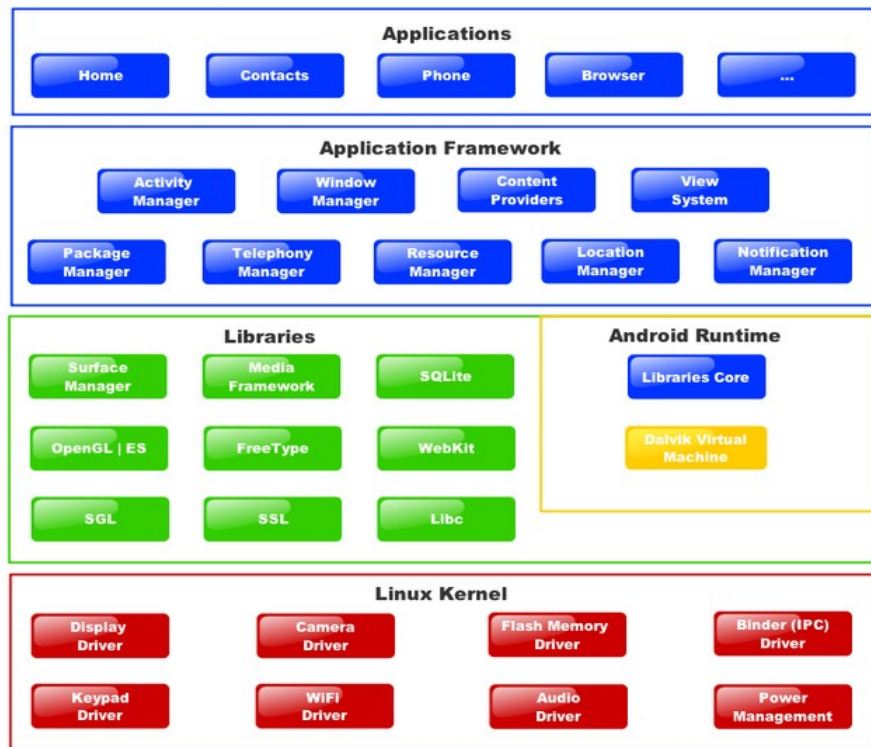
Eclipse es un entorno de desarrollo integrado (IDE) en el que encontrarás todas las herramientas y funciones necesarias para tu trabajo, recogidas además en una atractiva interfaz que lo hace fácil y agradable de usar.



Eclipse cuenta con un editor de texto donde puedes ver el contenido del fichero en el que estás trabajando, una lista de tareas, y otros módulos similares.

Si bien las funciones de Eclipse son más bien de carácter general, las características del programa se pueden ampliar y mejorar mediante el uso de plugins.

**[2] Android:** Android es un sistema operativo inicialmente pensado para teléfonos móviles, al igual que iOS, Symbian y Blackberry OS. Lo que lo hace diferente es que está basado en Linux, un núcleo de sistema operativo libre, gratuito y multiplataforma.



El sistema permite programar aplicaciones en una variación de Java llamada Dalvik. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las llamadas, la agenda, etc.) de una forma muy sencilla en un lenguaje de programación muy conocido como es Java.

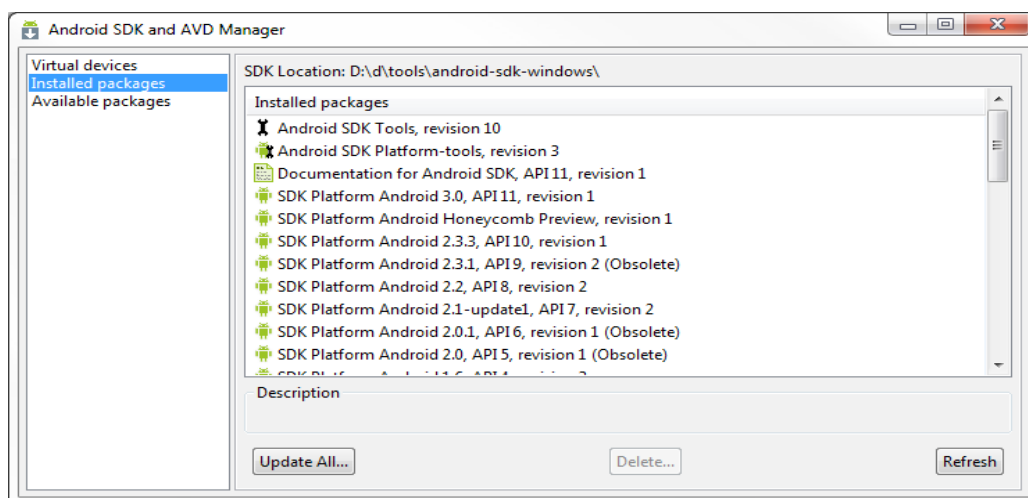
Esta sencillez, junto a la existencia de herramientas de programación gratuitas, hacen que una de las cosas más importantes de este sistema operativo sea la cantidad de aplicaciones disponibles, que extienden casi sin límites la experiencia del usuario.

Una de las mejores características de este sistema operativo es que es completamente libre. Es decir, ni para programar en este sistema ni para incluirlo en un teléfono hay que pagar nada. Y esto lo hace muy popular entre fabricantes y desarrolladores, ya que los costes para lanzar un teléfono o una aplicación son muy bajos.

Cualquiera puede bajarse el código fuente, inspeccionarlo, compilarlo e incluso cambiarlo. Esto da una seguridad a los usuarios, ya que algo que es abierto permite detectar fallos más rápidamente. Y también a los fabricantes, pues pueden adaptar mejor el sistema operativo a los terminales.

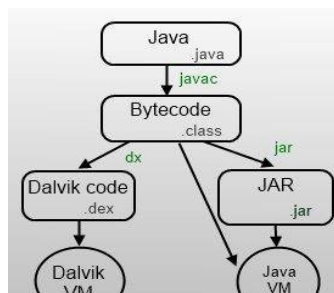


**[3] Android Developer Tools (Plugin Android para Eclipse):** es un plugin para Eclipse que proporciona un conjunto de herramientas que se integran con el IDE de Eclipse. Ofrece acceso a muchas características que ayudan a desarrollar aplicaciones de Android rápidamente. ADT ofrece acceso GUI a muchas de las herramientas de línea de comando SDK, así como una herramienta de diseño de interfaz de usuario para la creación rápida de prototipos, diseño y construcción de la interfaz de usuario de la aplicación.



**[4] Dalvik:** es la máquina virtual que utiliza la plataforma para dispositivos móviles Android. Dalvik ha sido diseñada por Dan Bornstein con contribuciones de otros ingenieros de Google.

Dalvik está optimizada para requerir poca memoria y está diseñada para permitir ejecutar varias instancias de la máquina virtual simultáneamente, delegando en el sistema operativo subyacente el soporte de aislamiento de procesos, gestión de memoria e hilos.



A menudo Dalvik es nombrada como una máquina virtual Java, pero esto no es estrictamente correcto, ya que el bytecode con el que opera no es Java bytecode. Sin embargo, la herramienta dx incluida en el SDK de Android permite transformar los archivos Class de Java compilados por un compilador Java al formato de archivos Dex.

**[5] Java:** es un lenguaje de programación con el que podemos realizar cualquier tipo de programa. En la actualidad es un lenguaje muy extendido y cada vez cobra más importancia tanto en el ámbito de Internet como en la informática en general. Está desarrollado por la compañía Sun Microsystems con gran dedicación y siempre enfocado a cubrir las necesidades tecnológicas más punteras.

Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si hacemos un programa en Java podrá funcionar en cualquier ordenador del mercado.

Es una ventaja significativa para los desarrolladores de software, pues antes tenían que hacer un programa para cada sistema operativo, por ejemplo Windows, Linux, Apple, etc.



Esto lo consigue porque se ha creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente.

**[6] Git:** es un sistema de control de versiones diseñado para manejar proyectos muy grandes con velocidad y eficiencia, pero igual de apropiado para repositorios pequeños; es especialmente popular con la comunidad open source, sirviendo como plataforma de desarrollo para proyectos como el Kernel Linux, Ruby on Rails, WINE o X.org.



Cada directorio de trabajo de Git es un repositorio completo con historial y capacidades totales de tracking de revisiones, independiente de acceso de red o un servidor central. Aún así, Git es extremadamente rápido y eficiente con el espacio.

### ***5. Justificación económica.***

Como justificación económica para mi aplicación Android me voy a basar en 4 puntos principales:

- Cuenta desarrollador de Google Play
- La publicidad
- Precio estipulado para la aplicación en Google Play
- Venta de la aplicación a algún interesado.

#### -Cuenta desarrollador de Google Play:

Lo primero de todo, para que una aplicación pueda estar visible y disponible para su descarga en Google Play hay que crearse una cuenta de desarrollador.

El precio de la cuenta es de 25 dólares (unos 19 euros). Se paga una única vez y desde ese momento ya podemos empezar a subir nuestras aplicaciones a la tienda de Google.

Para poder amortizar la aplicación podemos hacerlo de 2 maneras distintas: con la publicidad y generando ingresos mediante Admob por ejemplo, y estableciendo un precio de venta de la aplicación desde el mismo Google Play.



### -La publicidad:

Mediante la publicidad lo que hacemos es mostrar anuncios de otras aplicaciones y es gracias a que promocionamos esas aplicaciones que si un usuario hace click en el banner y además genera impresiones de la publicidad conseguimos el dinero.

Si la aplicación no tiene muchas descargas será difícil generar ingresos mediante publicidad.

### -Precio estipulado para la aplicación en Google Play:

Para poner nuestra aplicación de pago en Google Play lo que hay que hacer es después de registrarnos como desarrolladores, acudir al apartado de Google Merchant y darnos de alta. Se rellena un formulario y después establecemos el precio estipulado que ponemos a nuestra Aplicación.

Yo para mi aplicación establecería un precio de 50 cent,por lo menos al empezar. La aplicación cumple sus funciones pero para darla a conocer y empezar a generar ingresos y conocer un poco el mundo de Google Pay y demás estaría bien. Si en un futuro incorporo las mejoras de Twitter,Facebook,trailers etc el precio lo establecería en 1€ e iría actualizando cada cierto tiempo la aplicación con nuevas funciones, más series, mejor contenido visual,etc.

### -Venta de la aplicación a algún interesado:

Es difícil generar ingresos en Google Play con la de millones de aplicaciones que ya hay, pero si fuese el caso de que algún interesado quisiese comprar la aplicación, el precio de venta lo haría calculando las horas y el esfuerzo que me ha llevado la aplicación y según el éxito que estuviese cosechando en la tienda.

A la aplicación le habré dedicado entre plantearme el diseño de la aplicación, buscar

información, desarrollarla, testearla y demás unas 50 o 60 horas. Sé que es más de lo que tocaba pero hay veces que te atascas en un sitio y puedes encontrar la solución en unas horas como en días.

Como he dicho antes no podría calcular por cuando la vendería pero haría los cálculos basándome en función de las horas trabajadas en la aplicación, de los beneficios que dejaría de obtener al tener que quitarla de Google Play para cederla al comprador, etc etc.

Otra solución también podría ser venderla a un precio no muy caro y quedarme con un % de las ganancias generadas por la publicidad,etc.

## **6.Prevencción de riesgos laborales (Sector de la Informática)**

A continuación explicaré los siguientes riesgos laborales que se suelen presentar en el sector de la informática:

### 2.1. CAÍDAS AL MISMO NIVEL

- Los suelos de los locales de trabajo deben ser fijos, estables y no resbaladizos, sin irregularidades ni pendientes peligrosas.
- Se debe canalizar todo el cableado de ordenadores y demás instalaciones.
- Mantener las zonas de paso despejadas.

### 2.2. CAÍDAS A DISTINTO NIVEL

- Las escaleras de mano deben subirse con precaución, siempre de frente a ellas, tanto al subir como al bajar agarrándose con las dos manos.
- Facilitar el acceso a zonas de almacenamiento elevadas mediante escaleras fijas o móviles perfectamente aseguradas.

### 2.3. GOLPES CONTRA OBJETOS

- Mantener despejados de objetos los pasillos y las zonas de paso.

### 2.4. TRASTORNOS MUSCULOESQUELÉTICOS

- Las mesa de trabajo debe ser lo suficientemente amplia y espaciosa para que en ella se puedan depositar cómodamente todos los utensilios necesarios para el desarrollo de la tarea. Las dimensiones aproximadas deben ser: 160 cm de ancho por 80 cm de alto y 100 cm de profundidad.
- Como regla general la pantalla debe estar a unos 40 cm y la parte superior de la pantalla al altura de los ojos.
- Establecer pausas planificadas, cuya duración dependerá de las exigencias concretas de cada tarea.

### 2.5. GOLPES Y/O CORTES CON HERRAMIENTAS

- Cumplir las normas de seguridad indicadas por el fabricante.
- Guardar las herramientas cortantes en fundas y/o soportes adecuados.
- En general, se utilizarán las herramientas de acuerdo con su función, y de una manera prudente.
- Las herramientas se mantendrán en buen estado.

### 2.6. CONTACTO ELÉCTRICO

- La instalación eléctrica dispondrá de protección magnetotérmica, diferencial y toma de tierra.
- Evitar sobrecargar los enchufes con ladrones.
- Utilizar para los elementos portátiles tensiones de seguridad.
- Realizar un mantenimiento periódico de las instalaciones por instalador autorizado.

- No utilizar aparatos eléctricos con las manos húmedas.
- Separar el cableado de las fuentes de calor.
- No realizar operaciones de mantenimiento de los equipos electrónicos sin desconectarlos de la red.

## 2.7. FATIGA VISUAL

- Se graduará el brillo y el contraste mediante los mandos de la pantalla.
- Se colocará la pantalla de tal manera que están situadas paralelamente a ella las fuentes de iluminación.

## 2.8. CONFORT ACUSTICO

- Se regularán los timbres de los teléfonos.
- El ruido ambiental no debe superar los 50 db (A)

## 2.9. CONFORT TERMICO

- La temperatura de los locales se mantendrá entre 17 y 27 °C, y la humedad relativa entre el 30 y el 70%.
- Se instalarán si es necesario sistemas de aire acondicionado que mantengan la temperatura de los locales entre estos valores.

## 2.10. FACTORES PSICOSOCIALES

- Se intentará que el trabajador tenga la máxima información sobre la totalidad del proceso en el que está trabajando.
- Distribuir claramente las tareas y competencias.
- Planificar los diferentes trabajos de la jornada, teniendo en cuenta una parte para imprevistos.

-Realizar pausas o alternancia de tareas para evitar la monotonía del trabajo.

## 2.11. INCENDIOS

- Existencia de extintores de incendios adecuados a la clase de fuego.
- Mantenimiento periódico de extintores y demás equipos contra incendios.
- Revisar y mantener las instalaciones eléctricas aisladas y protegidas.
- Instalar sistemas de detección y alarma.
- Señalizar las zonas de riesgo de incendio.
- Señalizar y dejar libres las salidas de emergencia.
- Realizar planes de emergencia e implantarlos.

## **7.Evaluación y conclusiones:**

Después de realizar este proyecto, he dado por concluido el logro de los objetivos que al principio de la realización del mismo me había planteado. No quise ponerme tampoco unas metas muy complicadas debido a que no sabía con certeza si podría tenerlo todo terminado para el plazo acordado, ya fuese por el tiempo o por si encontraba una dificultad por el camino.

Al principio la idea era de almacenar las sinopsis, los banners de imágenes, los contenidos de los Spinners y demás en una base de datos de manera que sería más fácil acceder a esos datos,modificarlos,borrarlos... pero a mitad de camino me empezaron a dar problemas con la Activity que tenía de dicha BD y en el ordenador de sobremesa no se abría bien,y en el portátil tengo el disco duro un poco dañado y a veces hay ficheros que los detecta y otros no y la aplicación no se abría bien, así que lo solucioné haciendo cada spinner en un string-array diferente, las sinopsis igual y que cada sinopsis fuese un ítem, los banners igual, etc.

Me costó un tiempo encontrar la manera de hacer que cada sinopsis y cada banner

mostrasen diferentes contenidos utilizando un único TextView e ImageView para todas las series, pero con la clase Application todo fue muchísimo más sencillo.

Luego vino un problema y es que aparte de ser un problema estético, me di cuenta de que programarlo con la clase Application y luego tener en las clases Buscador\_serie\_lista y Series\_momento\_lista un switch case me había dado la solución de mostrar bien los banners de imágenes pero el problema estaba en que cada vez que añadiese más y más series acabaría con un código kilométrico así que hice la clase FavSeriesApplication e implementando Parcelable y FavSeriesApplication\_Objects con los arrays de cada serie y haciendo simplemente los String de Nombre y Sinopsis y los Int para las Imágenes y los Spinners de temporadas y capítulos podía tener el código mejor estructurado y luego simplemente con pasar cada objeto de la clase FavSeriesApplication\_Objects obtenía lo mismo con la clase Application pero en únicamente una línea de código.

Por falta de tiempo, me queda pendiente en un futuro crear la opción de añadir la serie, de manera que al entrar en la ficha de la serie y pulsar el botón “Añadir Serie” se pasaría esa serie a la clase “MisSeries.java” en forma de lista pero lo he dejado al final por eso, por falta de tiempo y porque para hacerlo bien querría hacerlo con una BD que aparte de pasar la serie a dicha Activity luego pudiese seleccionar esa serie y eliminarla en el caso de no querer tenerla más en la lista de “Mis Series”, etc.

Los Spinners los he rellenado con 25 temporadas y 40 capítulos por igual para todas las series, así que también queda pendiente y como mejora el rellenar los Spinners de Temporadas y Capítulos de manera correcta, es decir, si por ejemplo la serie “Anatomía de Grey” tiene 4 temporadas y su primera temporada tiene 21 capítulos, haría que el spinner de capítulos dependa de la temporada que seleccione y que me saque automáticamente al pinchar una temporada, la cantidad de capítulos exacta que tiene.

En el futuro también gustaría compartir vía Twitter, Facebook, etc, es decir, la información que ha introducido el usuario y que así pueda compartir las series y temporadas que está visionando en esos momentos. Lo marco como algo futuro porque por ahora no lo veo del todo necesario.

También crearía un botón por ejemplo “trailer” con el que poder ver un pequeño trailer de la serie y así saber más de que trata.

## 8.Anexo (sin acabar)

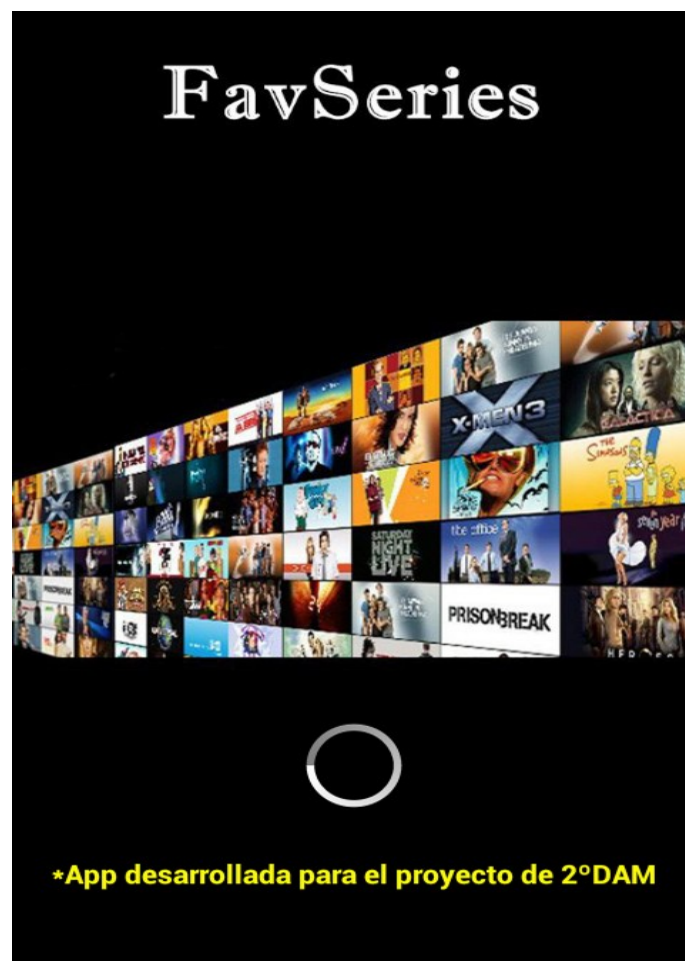
Con FavSeries podrás buscar información sobre la serie que tú necesites ,saber cuáles son las más exitosas del momento y además hacer un seguimiento diario de tus series.

Aplicación desarrollada por Gabriel Yangüez.

### MANUAL DE USUARIO DE LA APLICACIÓN

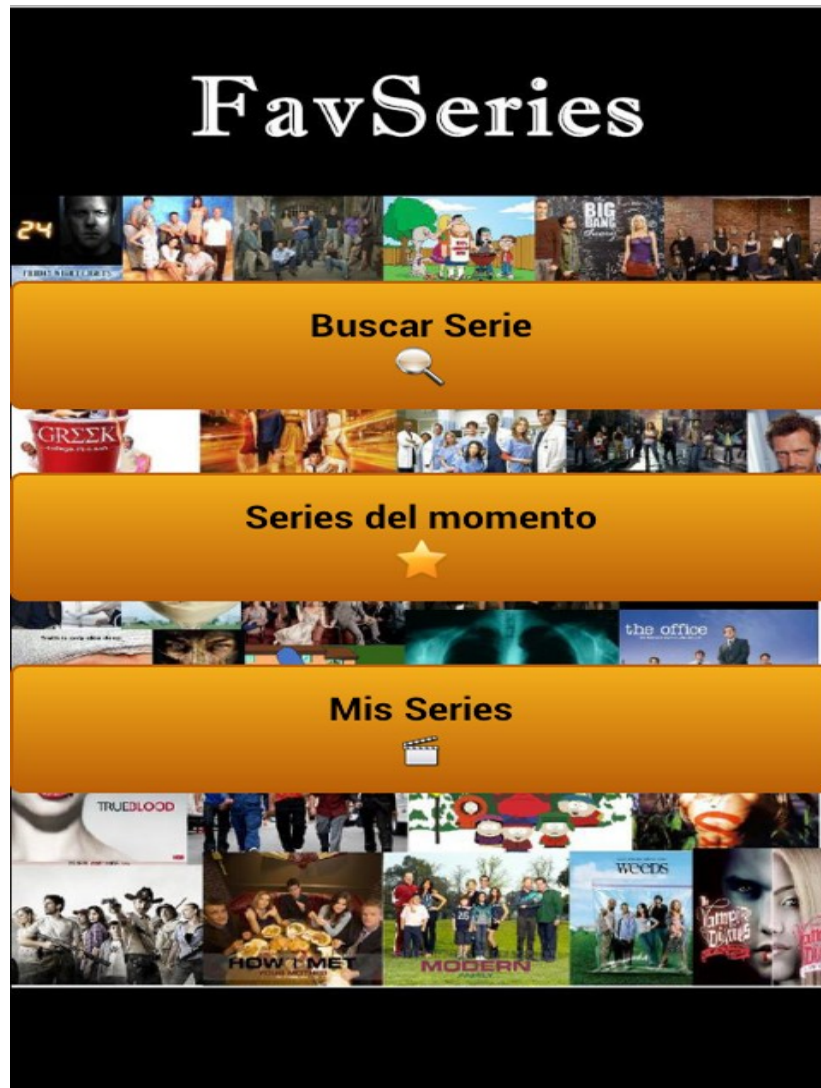
#### 1.Comienzo de uso de la Aplicación FavSeries:

Lo primero que encontramos al abrir la aplicación es una pantalla de presentación con la que intuimos que contendrá información sobre series, imágenes etc.



2.La pantalla que nos da acceso a las opciones de la aplicación:

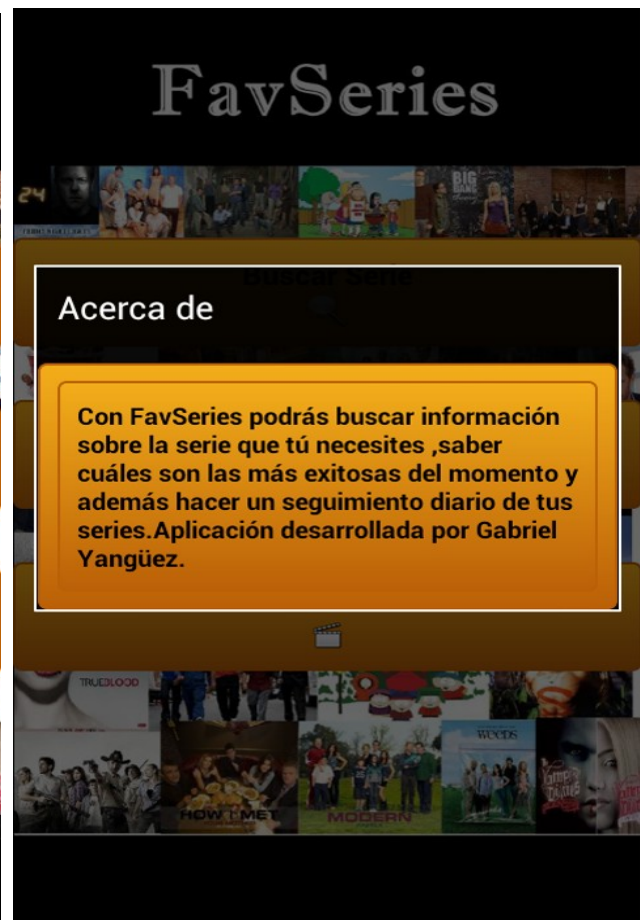
Después de la pantalla de presentación llegamos a la siguiente pantalla que nos permite ya empezar a dar uso de la misma. Encontramos 3 botones: Buscar serie, Series del momento, y Mis Series (futura implementación de éste).





### 3. Acerca de la Aplicación:

Si estamos en la pantalla de inicio y en el smartphone apretamos el botón de menú habitual nos saldrán dos opciones, de las cuales una es el Acerca de, con la que podemos saber un poco de información de la aplicación, lo que hace, quién la ha desarrollado, etc.



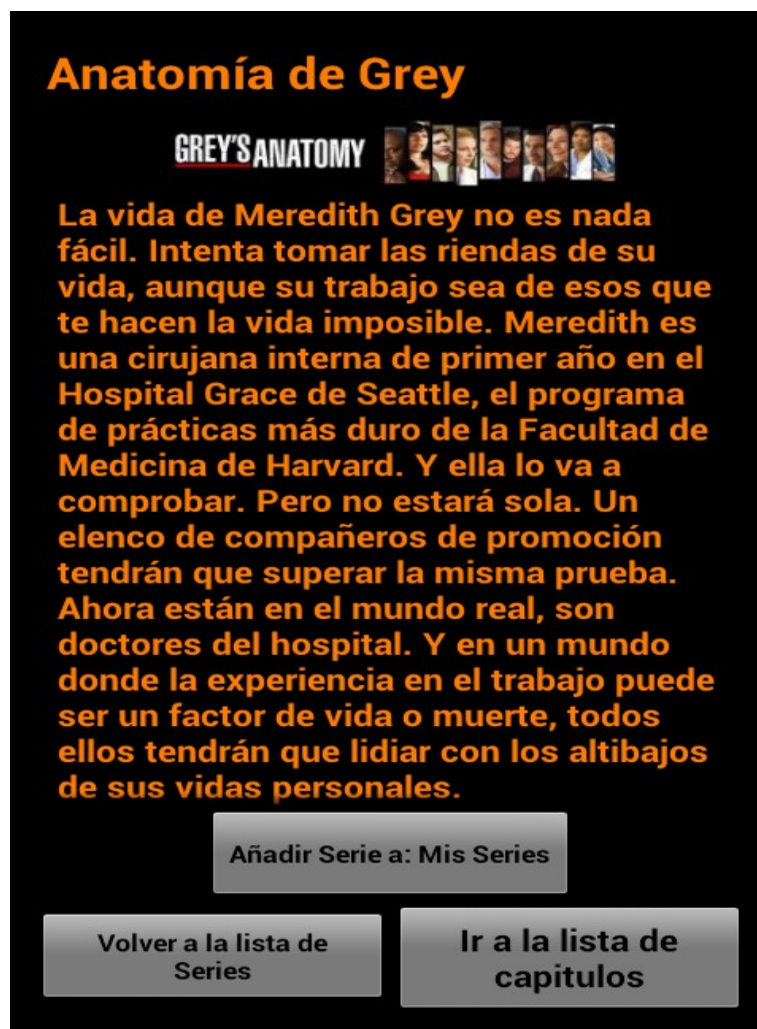
4.La lista de series en “Buscar serie”:

Una vez accedemos al botón “Buscar Serie” podemos ver un listado de más de 30 series de televisión ordenadas alfabéticamente.

<b>Anatomia de Grey</b>
<b>Arrow</b>
<b>Bones</b>
<b>Breaking Bad</b>
<b>Cálculo Electrónico</b>
<b>Castle</b>
<b>Como conocí a vuestra madre</b>
<b>Dexter</b>
<b>Dos hombres y medio</b>

## 5. La Ficha de la serie seleccionada:

Al seleccionar una serie cualquiera, nos aparecerá una pantalla en la que podremos ver el título de dicha serie, un banner con una imagen de la misma, y la sinopsis en la que nos describe un poco por encima el tema de la serie. Los 3 botones nos permiten “Añadir la serie a Mis Series” (no implementado), “Ir a la lista de capítulos y temporadas”, y luego “Volver a la lista de Series” para seguir viendo las demás.



## 6. Las Temporadas y Capítulos tanto en “Buscar serie” como en “Series momento”:

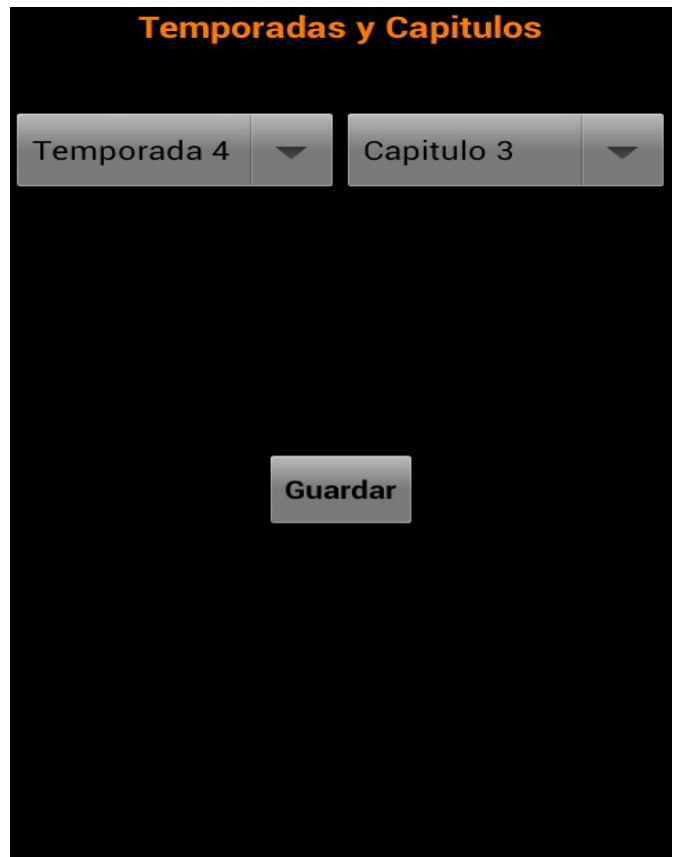
Al pinchar en el botón “Ir a la lista de capítulos” nos mostrará 2 Spinners en el que podremos ver tanto el número de temporadas para la serie escogida como el número de capítulos. En la parte de “Series momento” el contenido también se ve similar, pero son Spinners diferentes con lo cual la elección de uno no alterará los otros. Una vez elegidas las temporadas y capítulos pulsamos “Guardar” y aunque cerremos la aplicación, al volver a ver los datos de la misma, permanecerá la temporada y capítulos seleccionados anteriormente.

Temp. y Capítulos de “Buscar Serie”



The screenshot shows a dark-themed interface with the title "Temporadas y Capítulos" in orange at the top. Below the title, there are two spinner controls. The first spinner is labeled "Temporada: 5" and the second is labeled "Capítulo: 4". Both spinners have a downward arrow icon. At the bottom center, there is a grey button labeled "Guardar".

Temp. y Capítulos de “Series Momento”



The screenshot shows a dark-themed interface with the title "Temporadas y Capítulos" in orange at the top. Below the title, there are two spinner controls. The first spinner is labeled "Temporada 4" and the second is labeled "Capítulo 3". Both spinners have a downward arrow icon. At the bottom center, there is a grey button labeled "Guardar".

### 7. La lista de series en “Series momento”:

Si en la pantalla de inicio pinchamos en “Series del momento” nos llevará a una lista de las 10 series más exitosas en los últimos tiempos y de esta manera conocerlas mejor. Al estar la lista compuesta por banners de imágenes y no del título de la serie, aparte de dar una visualización diferente, capta más la atención y se puede intuir de que puede tratar la serie con ver dicha imagen. Al pinchar en una serie, nos llevará a su ficha como también pudimos ver anteriormente en “Buscar Serie”



## **9. Referencias y Bibliografía:**

### Aplicaciones de Referencia:

Para desarrollar la aplicación FavSeries he tomado como referencia algunas de las aplicaciones de Google Play que más o menos ejercían la misma función como por ejemplo Series.ly Mobile, TV Show Favs, etc.

### Webs de Referencia:

<http://stackoverflow.com/>

<http://developer.android.com/guide/components/index.html>

<http://www.androidhive.info/2011/10/android-listview-tutorial/>

<http://a2zandroidtutorials.blogspot.com.es/>

<http://jarroba.com/listview-o-listado-en-android/>

<http://thetvdb.com/>

**10.Link al proyecto FavSeries:**

<https://github.com/gabigoxi/proyectogabi>