

Manual de instalación de FilmUOC

Diego González Puga

Máster universitario de desarrollo de aplicaciones para dispositivos móviles
Trabajo Final del Máster DADM

Francesc D'Assís Giralte Queralt
Carles Garrigues Olivella

30/12/2020

Índice

1. Introducción.....	1
2. Firebase	2
3. TheMovieDB.....	5
4. Ejecución.....	7

Lista de figuras

Ilustración 1: Aplicación en Firebase	2
Ilustración 2: Google service en proyecto	3
Ilustración 3: Configuración de base de datos	4
Ilustración 4: Configuración identificación	4
Ilustración 5: Configuración TheMovieDB	5
Ilustración 6: Clave TheMovieDB	6
Ilustración 7: Ejecutar en Android Studio	7

1. Introducción

En este documento se describirán las instrucciones para compilar y poder ejecutar la aplicación. El documento definirá la configuración de **Firestore** y de la API de **TheMovieDB**, aunque por defecto el proyecto incluye los valores utilizados para el desarrollo y que serán suficientes para el funcionamiento de la aplicación.

2. Firebase

Para la configuración de Firebase, lo primero que se debe hacer es crear un proyecto dentro de esta plataforma.

Una vez que el proyecto se encuentre creado, se debe crear una aplicación a través de la configuración del proyecto. Por seguridad, se debe añadir la huella digital del certificado SHA, de manera que nos aseguremos de que solamente se pueda acceder desde ella y no puedan hacer uso de este proyecto aplicaciones externas, obteniendo un resultado como el que se muestra a continuación:

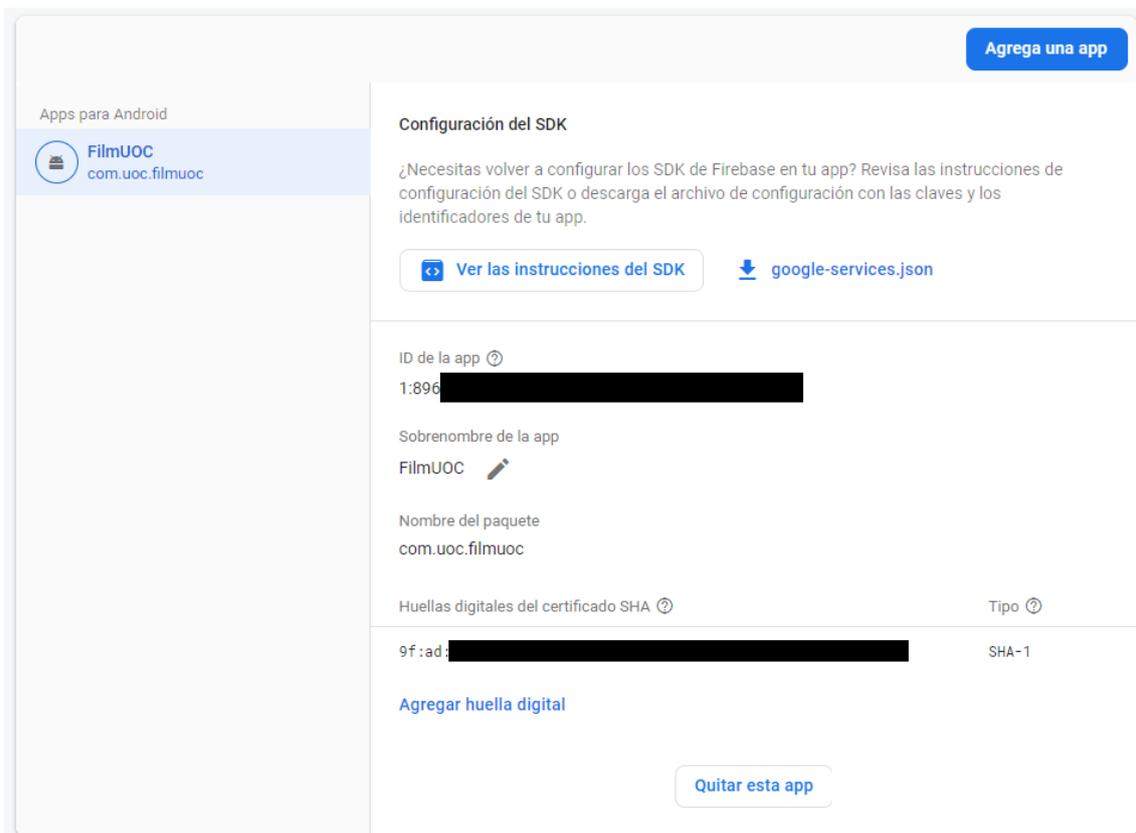


Ilustración 1: Aplicación en Firebase

Con la aplicación ya creada, hay que pulsar sobre la opción de descargar el archivo google-services.json que se ve en la pantalla. Dicho archivo se utilizará para remplazar el actual de la aplicación. Para encontrarlo, se deberá poner el visualizador del proyecto con la opción “**Project**” y estará situado dentro de la carpeta “**app**”:

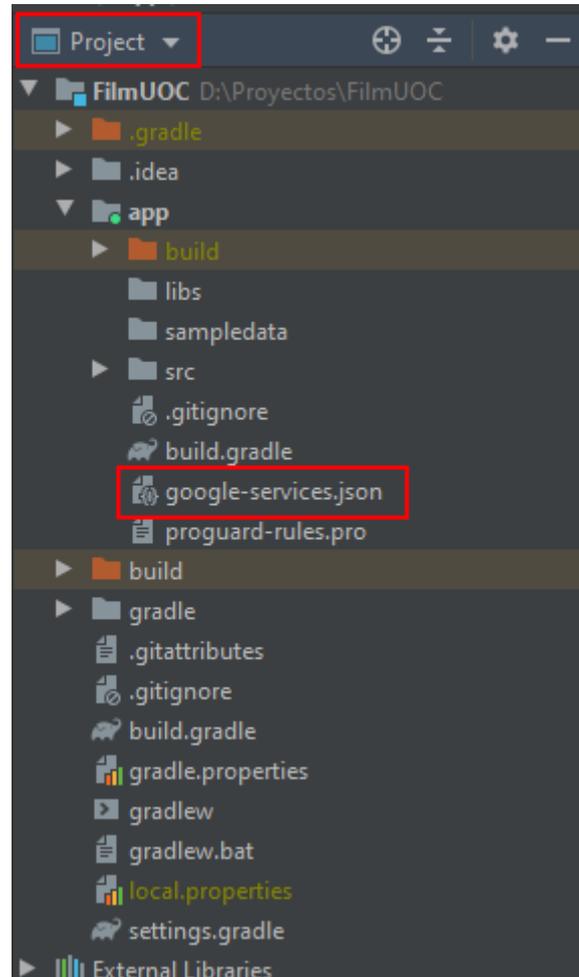


Ilustración 2: Google service en proyecto

Volviendo a Firebase, se procede a configurar la base de datos. Para ello, se debe acceder a la opción **“Realtime Database → Reglas”** y dar permiso de lectura y escritura a la aplicación:

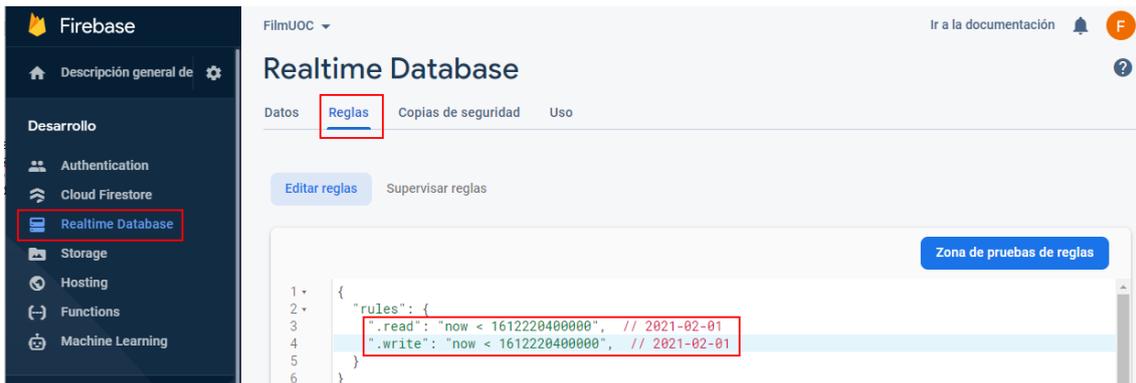


Ilustración 3: Configuración de base de datos

Por último, esta aplicación cuenta con identificación de usuarios a través de correo electrónico y de Google. Para que esto funcione se debe acceder a la sección de **“Authentication → Sign-in method”** y habilitar las opciones de **“Correo electrónico/contraseña”** y **“Google”**:

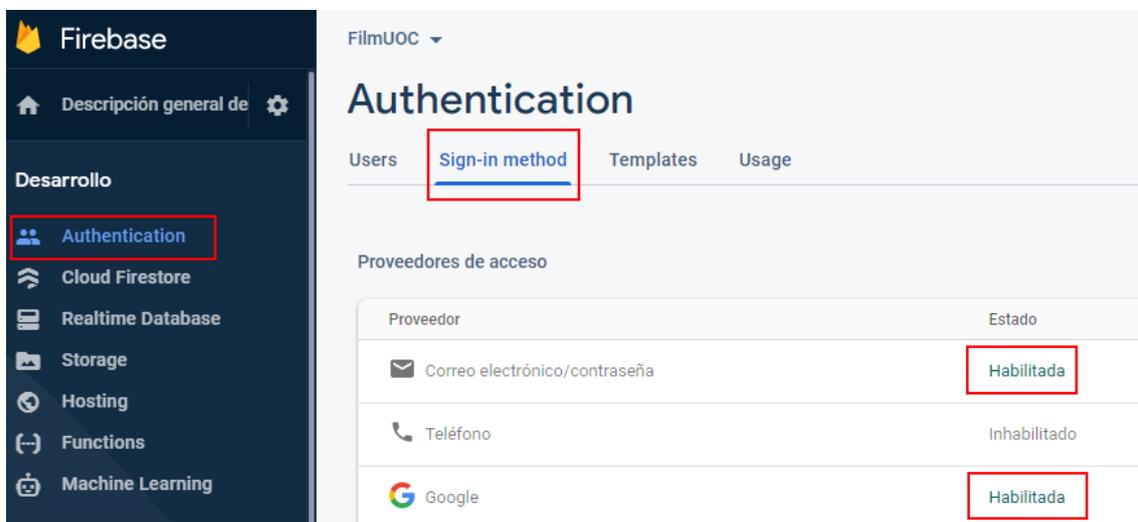


Ilustración 4: Configuración identificación

3. TheMovieDB

La aplicación hace uso de la base de datos de TheMovieDB accediendo a través de su API. Para obtener una clave para poder usarla, se debe acceder a su página web (<https://www.themoviedb.org/>) y crearse una cuenta.

Una vez que la cuenta se haya creado, accedemos a la sección “**Configuración** → **API**” donde se solicitará una serie de datos para conceder el acceso a la API. Una vez rellenados correctamente, se mostrará toda la información necesaria para hacer uso de esta:

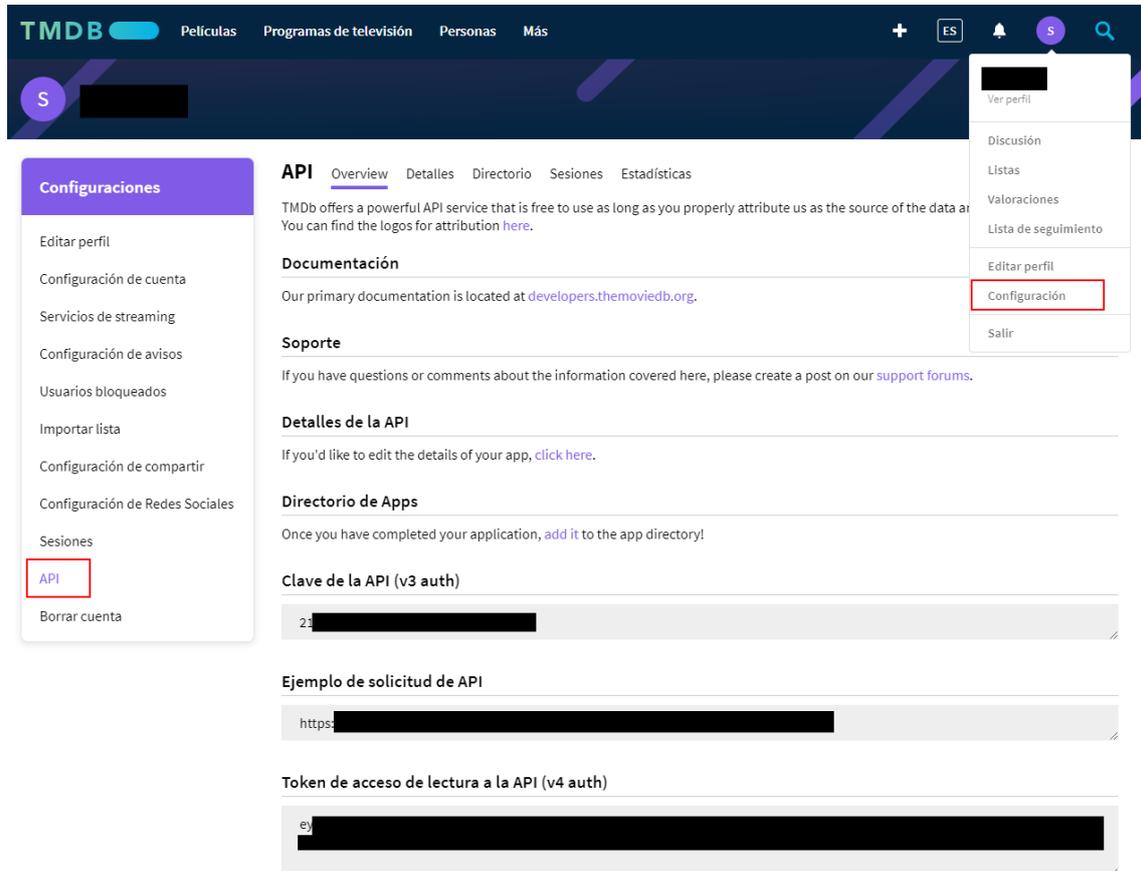
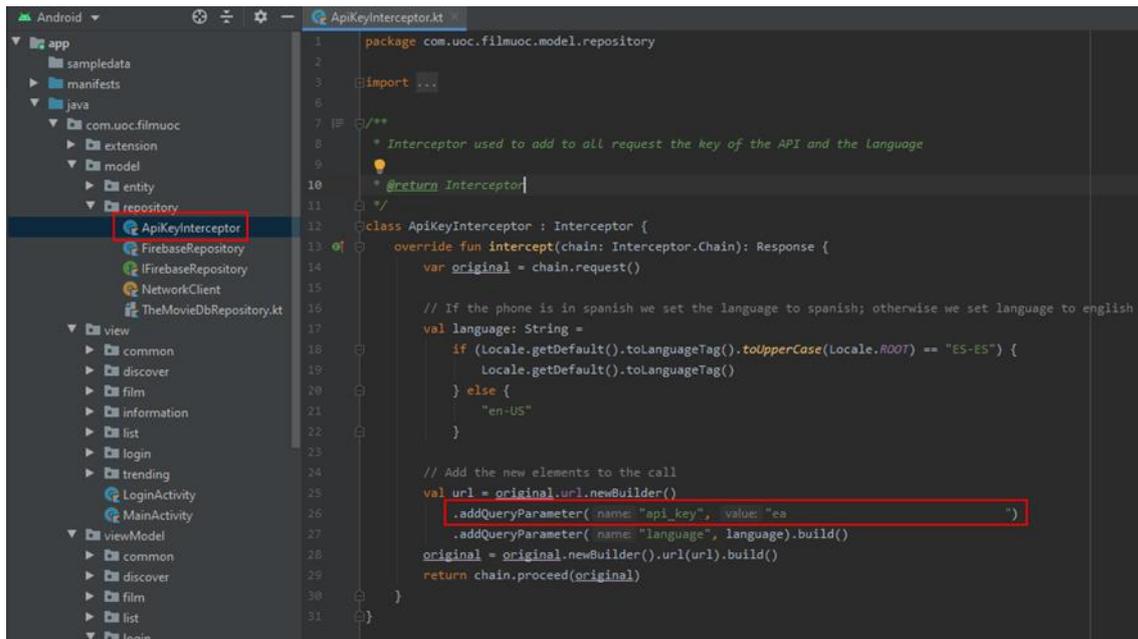


Ilustración 5: Configuración TheMovieDB

Finalmente, se deberá substituir la clave utilizada en la aplicación con la que se muestra bajo la etiqueta “**Clave de la API (v3 auth)**”. Dicha clave, en el código, se encuentra situada en la clase **ApiKeyInterceptor** del modelo:



```
1 package com.uoc.filmuoc.model.repository
2
3 import ...
4
5
6
7 /**
8  * Interceptor used to add to all request the key of the API and the Language
9  */
10 @return Interceptor
11
12 class ApiKeyInterceptor : Interceptor {
13     override fun intercept(chain: Interceptor.Chain): Response {
14         var original = chain.request()
15
16         // If the phone is in spanish we set the language to spanish; otherwise we set language to english
17         val language: String =
18             if (Locale.getDefault().toLanguageTag().toUpperCase(Locale.ROOT) == "ES-ES") {
19                 Locale.getDefault().toLanguageTag()
20             } else {
21                 "en-US"
22             }
23
24         // Add the new elements to the call
25         val url = original.url.newBuilder()
26             .addQueryParameter( name: "api_key", value: "ea" )
27             .addQueryParameter( name: "language", language).build()
28         original = original.newBuilder().url(url).build()
29         return chain.proceed(original)
30     }
31 }
```

Ilustración 6: Clave TheMovieDB

4. Ejecución

Tras abrir el código adjunto con Android Studio, este comenzará a descargar todas las dependencias necesarias a través del Gradle, por lo que una vez que termine este proceso solamente habrá que seleccionar el dispositivo en el que se quiere lanzar la aplicación (emulador o dispositivo físico) y pulsar el botón de play.

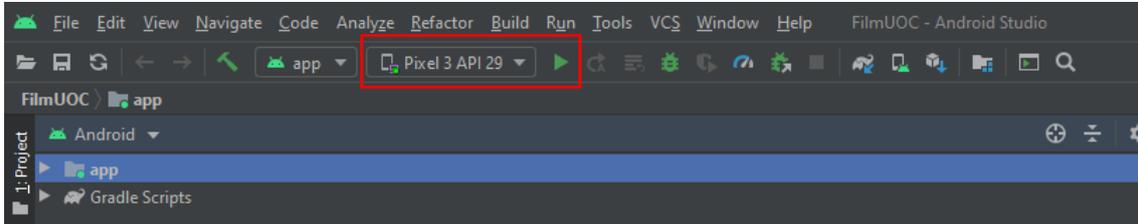


Ilustración 7: Ejecutar en Android Studio