

Trabajo Práctico

El objetivo del trabajo es desarrollar una aplicación Android utilizando los conceptos aprendidos durante el seminario.

El trabajo consta de dos partes:

- Un enunciado principal **requerido para poder aprobar el seminario**, donde se pondrán aplicar los conceptos básicos vistos durante el seminario.
- Una serie de enunciados **opcionales** (desafíos) que buscarán poner a prueba conceptos más avanzados, algunos vistos durante el seminario, y otros que requerirán investigar material extra al explicado durante el seminario. La realización de algunos de estos desafíos contribuirá a la calificación final del trabajo práctico entregado.

Contexto

La aplicación a desarrollar se trata de una aplicación de películas, donde el usuario puede buscar películas, ver su descripción, sus actores, su foto de portada, etc. La información de las películas será provista por [The Movie Database](https://www.themoviedb.org/)¹ (TMBD), la cual cuenta con un gran catálogo de películas accesible mediante una API. Para poder acceder a esta API necesitan obtener un [Token](https://developer.themoviedb.org/docs/authentication-application)², el cual pueden obtener de manera gratuita registrándose en la página. A lo largo de este documento encontrará notas al pie de cada página con enlaces a documentación sobre cada uno de los servicios que se van mencionando.

Enunciado principal

La aplicación deberá tener una pantalla principal en donde el usuario pueda visualizar una lista de películas populares ([movie-popular-list](https://developer.themoviedb.org/reference/movie-popular-list)³). Esta lista deberá mostrar mínimamente el título y una imagen de la película (puede mostrar más datos si así lo desean).

Al seleccionar una película de la lista, se deberá navegar a una segunda pantalla donde se podrá visualizar más información sobre la misma ([movie-details](https://developer.themoviedb.org/reference/movie-details)⁴). Esta pantalla deberá mostrar (como mínimo) los siguientes datos: el título de la película, la portada ([image-basics](https://developer.themoviedb.org/docs/image-basics)⁵), la sinopsis / resumen, los géneros, los actores que actúan en ella, y el rating de la película. Pueden mostrar más información si así lo desean.

¹ <https://www.themoviedb.org/>

² <https://developer.themoviedb.org/docs/authentication-application>

³ <https://developer.themoviedb.org/reference/movie-popular-list>

⁴ <https://developer.themoviedb.org/reference/movie-details>

⁵ <https://developer.themoviedb.org/docs/image-basics>

Ambas pantallas deberán mostrar indicadores de carga mientras consultan los servicios correspondientes. En caso de que alguna consulta falle, ya sea al obtener el listado de películas populares o los detalles de una película seleccionada, deberán mostrarse los errores correspondientes en pantalla y darle la opción al usuario de reintentar la carga de datos.

Enunciados secundarios

A continuación se enumeran algunos desafíos opcionales para profundizar / aprender algunos conceptos. Los enunciados son independientes el uno del otro: no es necesario hacerlos todos ni hacerlos en el orden en el que se listan.

a) Búsqueda de películas

Agregar un campo de búsqueda en la pantalla principal que permita al usuario buscar una película en el catálogo ([search-movie](https://developer.themoviedb.org/reference/search-movie)⁶). Mientras se busca una película, se ocultan las películas “populares” y se muestra la lista resultante de la búsqueda. Si la lista resultante está vacía (es decir, si no se encuentran películas con el título introducido) se debe mostrar un mensaje acorde (por ejemplo: “No se encuentra la película buscada” o “No hay resultados para la búsqueda”). Si el usuario borra la búsqueda y la deja vacía, se debe volver a mostrar la lista de películas populares.

b) Paginado de resultados

El servicio para listar las películas populares es paginado ([movie-popular-list](https://developer.themoviedb.org/reference/movie-popular-list)⁷). Esto es que acepta un query parameter opcional (*page*) para indicar el número de página que se desea cargar (por defecto es 1). Se desea implementar la lógica para que si el usuario llega al final de la lista, automáticamente se consulte por la segunda página, luego por la tercera, y así sucesivamente. Las películas de cada página consultada deberán mostrarse debajo de las películas de las páginas anteriores. Existen múltiples formas de implementar esta lógica, pero les sugerimos utilizar la popular librería [Paging](https://developer.android.com/topic/libraries/architecture/paging/v3-overview)⁸ provista por Android.

c) Lista de deseos

Agregar a la pantalla principal pestañas / tabs (requiere investigar sobre Tabs o BottomNavigation). En uno de los tabs se mostrará la lista de películas populares que se estaba mostrando originalmente. En la segunda pestaña se mostrará una “lista de deseos” (lease, películas que quiero ver). Al entrar a los detalles de una película, el usuario deberá poder agregar/quitar la película de dicha lista. Para almacenar la lista de películas, no será necesario almacenar todos los datos de dicha película, sinó únicamente el ID de la película y algunos datos mínimos necesarios para mostrar la película en la pantalla de “lista de deseos” (similar a la “lista de películas populares”). Al seleccionar una película de la “lista

⁶ <https://developer.themoviedb.org/reference/search-movie>

⁷ <https://developer.themoviedb.org/reference/movie-popular-list>

⁸ <https://developer.android.com/topic/libraries/architecture/paging/v3-overview>

de deseados” se deberá navegar a la pantalla que mostrará los detalles de la película (es decir, la misma pantalla que se muestra al seleccionar una película de la “lista de películas populares”).

d) Lista de tendencias

Agregar a la pantalla principal pestañas / tabs (requiere investigar sobre Tabs o BottomNavigation). En uno de los tabs se mostrará la lista de películas populares que se estaba mostrando originalmente. En la segunda pestaña se mostrará la lista de películas que fueron tendencia en la última semana ([trending-movies](https://developer.themoviedb.org/reference/trending-movies)⁹). Al seleccionar una película (ya sea del listado de populares o del listado de tendencias) se deberá navegar a la pantalla que muestra los detalles de dicha película.

e) Idioma en las solicitudes

Los servicios de TMBD aceptan un query parameter opcional (*language*) para indicar el lenguaje de la respuesta (por defecto es en-US). Para mejorar la experiencia del usuario, se desea especificar en las consultas el idioma establecido en el dispositivo para que el idioma de la información sea el mismo utilizado en el celular. Para obtener el lenguaje del celular se aconseja revisar la referencia de [Locale](#)¹⁰. Para especificar el query parameter en cada solicitud pueden utilizar las anotaciones de [Retrofit](#)¹¹ o implementar un [Interceptor](#)¹² que intercepte las solicitudes HTTP salientes y les agregue el query parameter de *language*.

f) Recomendaciones y películas similares

En la pantalla donde se muestran los detalles de una película, mostrar la lista de películas similares ([movie-similar](#)¹³) y/o la lista de películas recomendadas ([movie-recommendations](#)¹⁴) a partir de la premisa de que al usuario le gustó la película que tiene seleccionada. Si el usuario selecciona alguna de las películas recomendadas o similares se deberá navegar a la pantalla para mostrar los detalles de esa película. Al navegar hacia atrás desde los detalles de la película recomendada/similar, se deberá volver a ver los detalles de la película inicial.

g) Layout adaptativo

Si el usuario rota el dispositivo y lo coloca de manera horizontal, la pantalla principal donde se listan las películas populares deberá cambiar su layout para aprovechar mejor el espacio. En lugar de mostrar las películas en una lista vertical (una debajo de la otra), deberá mostrar las películas en una especie de grilla con dos columnas.

⁹ <https://developer.themoviedb.org/reference/trending-movies>

¹⁰ <https://developer.android.com/reference/java/util/Locale>

¹¹ <https://square.github.io/retrofit/>

¹² <https://square.github.io/okhttp/features/interceptors/>

¹³ <https://developer.themoviedb.org/reference/movie-similar>

¹⁴ <https://developer.themoviedb.org/reference/movie-recommendations>

h) Interacción con otras apps

En la pantalla que muestra los detalles de una película se debe mostrar a) un botón para compartir la película (para compartir por ejemplo por whatsapp o por alguna red social) y b) un botón para abrir la homepage de la película (en caso de que la película tenga alguna homepage). Al tapear el botón para compartir la película se deberá generar un mensaje con información de la película (título y sinopsis). Si desean pueden enviar información extra, como la imagen de la portada. Al tapear el botón de homepage se deberá abrir el link de la página en el navegador web predeterminado del dispositivo (por ejemplo, Chrome, Firefox, etc).

¿Qué se evaluará?

- Que el código sea legible y de fácil manipulación, utilizando lenguaje Kotlin y aplicando correctamente los patrones estándares de la tecnología explicados durante el seminario (MVVM y Repository Pattern)
- Que la aplicación sea robusta, es decir, bajo ningún punto de vista la aplicación debería cerrarse inesperadamente ni la UI deberá quedarse colgada debido a una operación de I/O en disco o en la red.
- Que los diseños de las vistas se adapten bien a distintos tamaños de celulares o a distintas configuraciones (modo oscuro y modo claro; idioma inglés e idioma español).
- Que la aplicación sea intuitiva y muestre los mensajes de error correspondientes cuando no se pueda mostrar la información solicitada. Doble mérito si se identifica el origen del error (por ejemplo, si se identifica y se le informa al usuario cuando NO tiene internet).
- Se valorará positivamente el esfuerzo y el tiempo dedicado en la realización de los enunciados secundarios.