

---

## Contents

Description .....	2
Intended User .....	2
Features.....	2
User Interface Mocks .....	3
Screen 1 .....	3
Screen 4 (widget) .....	3
Screen 2.....	4
Screen 3.....	5
Key Considerations .....	5
How will your app handle data persistence? .....	5
Describe any edge or corner cases in the UX. ....	5
Android Studio .....	5
Programming Language.....	5
Accessibility .....	5
Resources.....	6
Themes.....	6
Themoviedb API .....	6
Describe any libraries you'll be using and share your reasoning for including them. ....	6
Describe how you will implement Google Play Services or other external services. ....	6
Next Steps: Required Tasks.....	7
Task 1: Project Setup .....	7
Task 2: Configure themoviedb APIkey .....	7
Task 3: Implement UI for Each Activity and Fragment .....	7
Task 4: Setup API Calls and Interface .....	7
Task 5: Setup Data layer .....	7
Task 6: Setup UI layer and Uses-cases components .....	8
Task 7: Polish UI.....	8

**GitHub Username:** [Lepatlamass](#)

# LLMOVIES

## Description

LLMOVIES is remaster version of the popular movie app using Google material design principle to show case Movies, TV Show and search them, its main aim is to help users search and view their best movies and series.

## Intended User

Our users are people passionate about film and TV Shows.

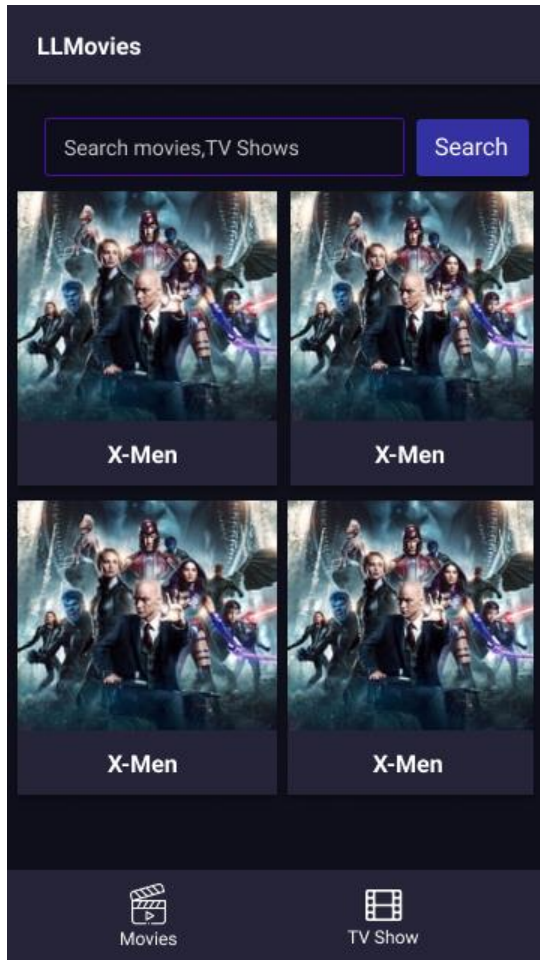
## Features

List the main features of your app. For example:

- Search Movies and TV Shows.
- View Movies and TV Shows.
- Rate Movies and TV Shows.
- View Reviews.

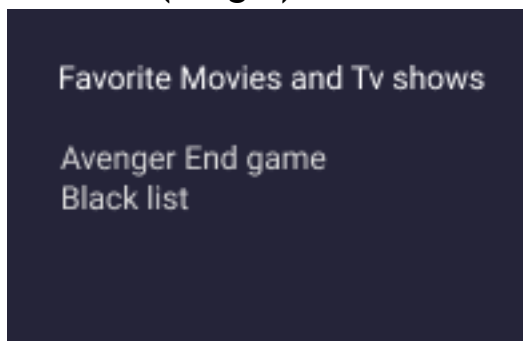
## User Interface Mocks

### Screen 1



When you open the app it will display the latest films, TV shows according to which tab you clicked and also give you the possibility to search movies or films.

### Screen 4 (widget)



showing favorite shows and movies.

## Screen 2



**Black Panther**  
(2018)

**Overview**

King T'Challa returns home from America to the reclusive, technologically advanced African nation of Wakanda to serve as his country's new leader. However, T'Challa soon finds that he is challenged for the throne by factions within his own country as well as without. Using powers reserved to Wakandan kings, T'Challa assumes the Black Panther mantle to join with girlfriend Nakia, the queen-mother, his princess-kid sister, members of the Dora Milaje (the Wakandan 'special forces') and an American secret agent, to prevent Wakanda from being dragged into a world war.

**User Score**

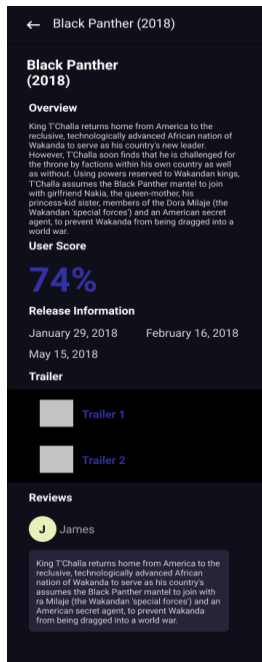
**74%**

**Release Information**

January 29, 2018    February 16, 2018  
May 15, 2018

Shows all the details of TV show or Movie

## Screen 3



## Key Considerations

How will your app handle data persistence?

Expenses data will be persisted using Room Database, use also Shared Preferences to store other necessary information.

Describe any edge or corner cases in the UX.

The app will always load latest movies on launching.

## Android Studio

The project will be completed using Android Studio (version 3.3+).

## Programming Language

The app will be written solely in the Java Programming Language.

## Accessibility

The app will support Android's accessibility features, through content descriptions. App enables RTL layout switching on all layouts.

## Resources

Values will not be hardcoded, instead they will be stored in a **strings.xml** file in **res/values** for Strings values. This will hold true for dimensional values, integer and Booleans, all in their respective folders.

## Themes

The app will use a personalized theme that extend the Theme.MaterialComponents with all colors resources such as Color Primary, Color Secondary and so on.

## Themoviedb API

The app will use themoviedb API to pull movies and TV shows.

Describe any libraries you'll be using and share your reasoning for including them.

- **Room Database:** to store our expenses data without SQLite boilerplate code. (version: 1.1.1)
- **Lifecycle:** to make UI automatically responds to lifecycle events. (version: 1.1.1)
- **Glide:** to load and cache our images. (version: 4.8.0)
- **Retrofit:** to turn Http API into java interface. (version: 2.4.0)
- **Material Component:** to use google material design component. (version: 1.0.0)
- **Butterknife:** to bind views. (version: 10.0.1)
- **Constraint-layout:** to well organized are layout. (version: 1.1.3)
- **Support Library:** Android support library (version: 28.0.0)

Describe how you will implement Google Play Services or other external services.

- **Google adMobs services** to display ads banner within the app.

- **Firestore Analytics:** analytics to track the usage of my application.

## Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

### Task 1: Project Setup

- Create the project on Android Studio
- Add gradle Dependencies for third-party libraries
- Create the app structure by adding packages for app components.

### Task 2: Configure themoviedb APIkey

- Create account as a developer at themoviedb
- Create and application
- Get the Apikey
- Configure your gradle Apikey.

### Task 3: Implement UI for Each Activity and Fragment

- Build UI for MainActivity to search and display latest films or Series
- Build UI to display a Movie or TV Show
- Build UI to display search movie and TV Show.

### Task 4: Setup API Calls and Interface

- Create class for Api calls
- Create Interface for the services.

### Task 5: Setup Data layer

- Create Data classes
- Create Room database components: DAO, database
- Create expense data repository to expose data to our viewModel.

## Task 6: Setup UI layer and Uses-cases components

- Create all UI classes and their related xml layouts
- Add related uses cases in viewModel classes.

## Task 7: Polish UI

- Use material components to make the app beautiful.