

Description

Intended User

Features

User Interface Mocks

Screen 1

Screen 2

Screen 3

Screen 4

Screen 5

Key Considerations

How will your app handle data persistence?

Describe any edge or corner cases in the UX.

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

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

Next Steps: Required Tasks

Task 1: Project Setup

Task 2: Implement UI for Each Activity and Fragment

Task 3: Implement the logic behind each of the features.

Task 4: Implement the widget

Task 5: Recommendation Engine

Task 6: Test the application

Task 7: Add accessibility support

**GitHub Username:** madhurimamalla

## What next?

### Description

People generally look for personalized recommendations, a service that understands their likes and dislikes and then recommends new content to them.

This application solves the problem by understanding a user's preferences and then suggest new content recommendations to the user.

## Intended User

A user would be anyone who's interested in watching some new content.

## Features

- Discovers and shows new content to the user
- Saves preferences of the user
- Stores a watchlist of content
- Searches for any content typed by the user

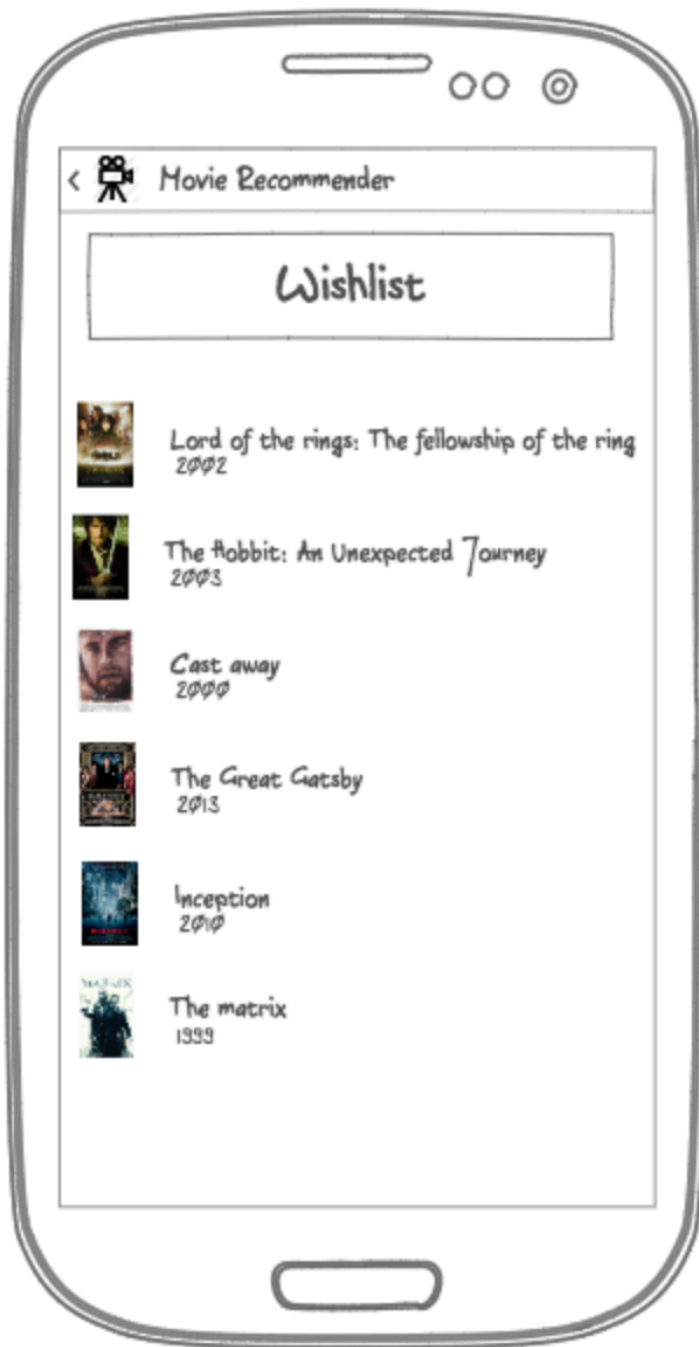
# User Interface Mocks

## Screen 1



This is the main screen the user will see when they launch the application

## Screen 2



This is the screen the user will see when he clicks on the Wishlist option on Screen 1

### Screen 3



This is the screen seen when the user clicks on the Discover option and here's where the application will recommend new content

## Screen 4



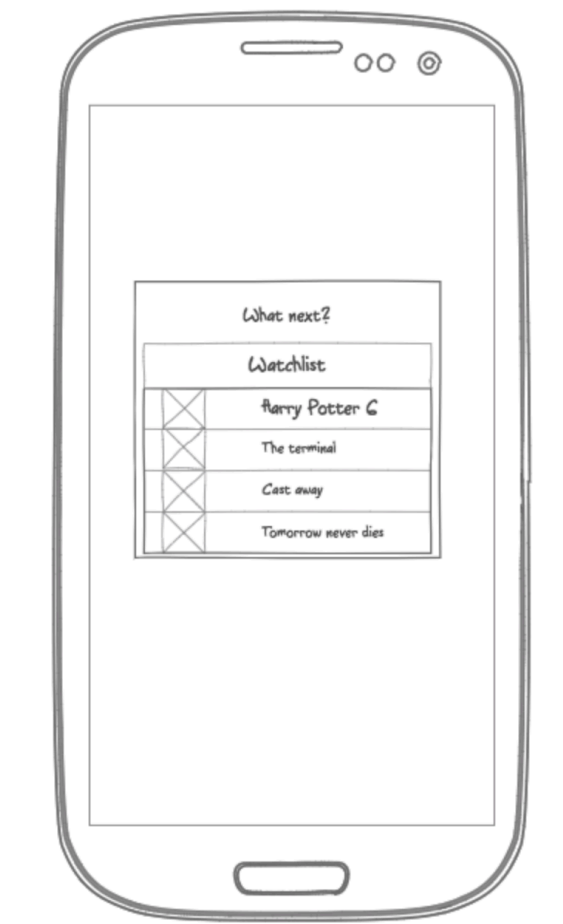
This is the screen seen when the user clicks on the Discover option and here's where the application will recommend new content.

On pulling up the bottom, a user can read the plot summary of the content.

Using the options above, the user can segregate the content shown as seen-like, seen-dislike, not-seen-don't-want-to-see & not-seen-love-to-see.

## Screen 5

This is the widget screen which displays the watchlist of the user. On clicking the widget, it's opens the application on the watchlist screen.



## Key Considerations

### How will your app handle data persistence?

We will store the user preferences on the Firebase realtime database using the Firebase authentication.

Based on that information stored, the application's recommendation engine will then discover content and present it to the user.

### Describe any edge or corner cases in the UX.

When there is no network connectivity, the application will show only your watchlist and the widget with the watchlist. For the features like Discover and Search, the user will need to turn on the network.

On bigger screens such as a tablet, the layout will be different to optimize according to the space available. On rotation, the screens will save the state and load the layout smoothly without any crashes.

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

Will be using Picasso to handle the loading and caching of images.

Will be using Room for the persistence library as it abstracts a layer and eases dealing with the DB.

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

The application will store the user preferences on the Firebase realtime database. The application will also use [Firebase Auth](#) to help the user access his information from multiple devices.

## 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

- Check feasibility of the application screens UI such as Discover screen etc



- Implement the application using Java
- Setup the project in Android Studio using only stable libraries.
- Apply for an API key from TMDB and set that up with my application
- Configure libraries such as Picasso and Room
- Decide the schema for the SQLite DB and configure the room library

## **Task 2: Implement UI for Each Activity and Fragment**

- Build the on-boarding screen which explains how to use the application
- Build UI for SplashActivity (Main Activity)
- Build another Activity UI with fragments for Wishlist, Discover, Search & History

## **Task 3: Implement the logic behind each of the features.**

- Create a short duration, on-demand request for search feature where the application is using an AsyncTask.

## **Task 4: Implement the widget**

- Implement the widget which displays the users watchlist.
- On clicking the widget, open the Watchlist Screen.

## **Task 5: Recommendation Engine**

- Implement the recommendation engine
- Test the recommendation engine results and optimize it

## **Task 6: Test the application**

- Implement tests to test that recommendation engine
- Write UI tests to test the application

## **Task 7: Add accessibility support**

- Add support to accessibility by using contentDescription attribute
- Make sure the application uses strings.xml and removes any hard-coded values in the layouts files etc.