GitHub Username: baguzzzaji

Dramania (Drama Tracking)

Description

People love watching dramas. Watching many dramas at a time make it hard to keep track on which episode we are on. This app will help users on managing dramas collection, tracking dramas episode watched, discover interesting dramas, and much more.

Intended User

Users who love to watch dramas.

Features

- View latest dramas

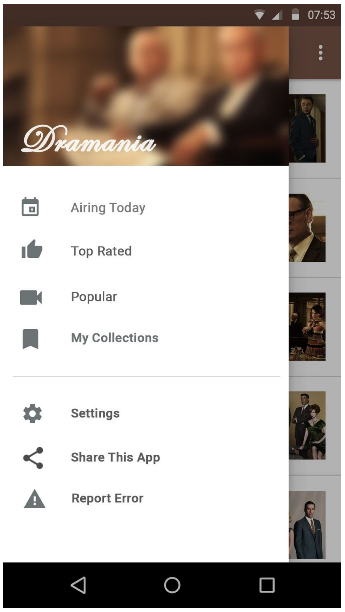
- View trending dramas
- View completed drama
- View currently airing dramas
- View drama detail information
- View actors/actresses information
- Save dramas to collection
- Filters drama (by countries and categories)
- Track episodes watched
- Share watched drama

User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

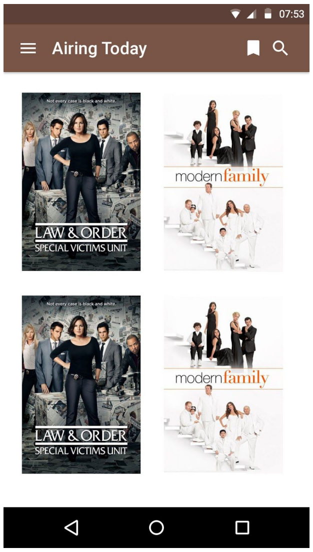
Screen 1

Navigation Drawer



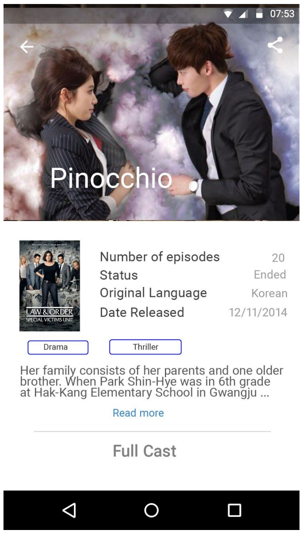
Screen 2

Display list of dramas in GridLayout



Screen 3

Display drama detail information including title, number of episodes, status, language, categories, synopsis and casts.



Pinocchio

Number of episodes	20
Status	Ended
Original Language	Korean
Date Released	12/11/2014

Drama Thriller

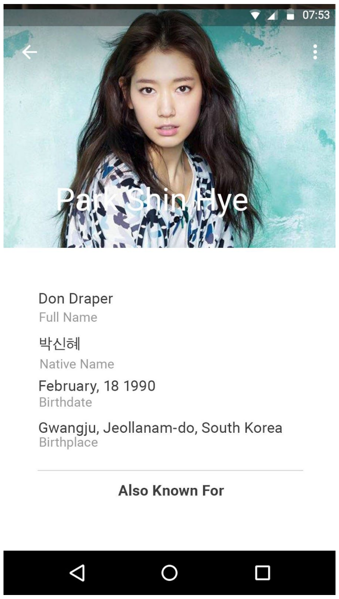
Her family consists of her parents and one older brother. When Park Shin-Hye was in 6th grade at Hak-Kang Elementary School in Gwangju ...

Read more

Full Cast

Screen 4

If users click on cast they will open actors detail page. This page contains a list of dramas which the actor play for.



Don Draper
Full Name

박신혜
Native Name

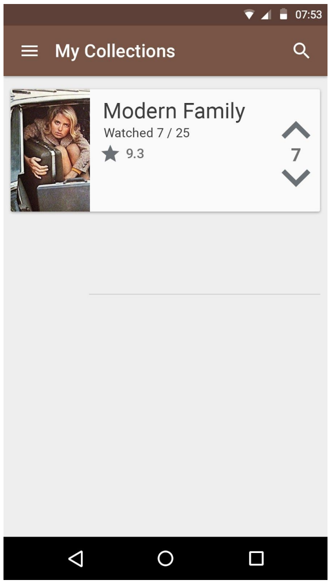
February, 18 1990
Birthdate

Gwangju, Jeollanam-do, South Korea
Birthplace

Also Known For

Screen 5

My Collections contains dramas for offline views and users can update the watched episodes here.



Modern Family
Watched 7 / 25
★ 9.3
7

Key Considerations

How will your app handle data persistence?

I will build it using Content Provider to save episodes and drama. In order to search for favorites dramas, this app will use AsyncTask.

Describe any corner cases in the UX.

Users could move around using navigation drawer. For drama and actor/actress detail screen, user could go back using up button.

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

- Picasso for loading images
- Retrofit to get data from server
- RecyclerView for displaying list of dramas
- ButterKnife to reduce findViewById boilerplate

Describe how you will implement Google Play Services.

I will use Google AdMob and Analytics. AdMob to get ads on the app to get money and analytics to analyze how users using the app.

Next Steps: Required Tasks

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

Task 1: Project Setup

- Configure build.gradle to include required libraries
- Get new API key from themoviedb.org
- Add API key to gradle.properties

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity/TrendingDramas
- Build UI for CurrentlyAiringDrama
- Build UI for CompletedDrama
- Build UI for MyDramaCollection
- Build UI for Navigation Drawer
- Build UI for Settings
- Build UI for DetailDrama
- Build UI for DetailActor

Task 3: Implement Navigation pattern

- Configure Navigation Drawer to correctly navigate users to various screen
- Configure Detail screen to use parent - child pattern

Task 4: Show Dramas

- Configure MainActivity/TrendingDrama to show trending dramas using RecyclerView in GridLayout. Set maximum dramas show in the activity to 50.
- Configure CurrentlyAiringDrama to show only the drama that haven't finished yet. Let users browse to all of them using *scroll to view more*.
- Configure CompletedDrama to show all completed drama. Let users browse to all of them using *scroll to view more*.
- Configure MyDramaCollection to show favourite drama and buttons to update the numbers of episode watched.
- Get drama data from servers using Retrofit.
- Save drama to database if user add them to MyDramaCollection for offline view.

Task 5: Implement Settings

- Configure xml for preference configuration
- Configure Settings to change users configuration

Task 6: Configure Filters

- Allow users to filter dramas by categories or countries.

Task 7: Implement Tablet Views

- Implement tablet view to show menu on the left side and detail activity on the right side.