GitHub Username: ladanisavan

ShowTime

Description.

ShowTime is TV series tracker app. These days with our busy life and quite a lot of options available in TV series. Many Times it difficult to decide which series to watch and to be sure of not forgetting any episode of our favorite shows. This application aims to solve all of that for the user, at the same time that makes the hobby of watching and following TV series something much more enjoyable and delightful.

Intended User

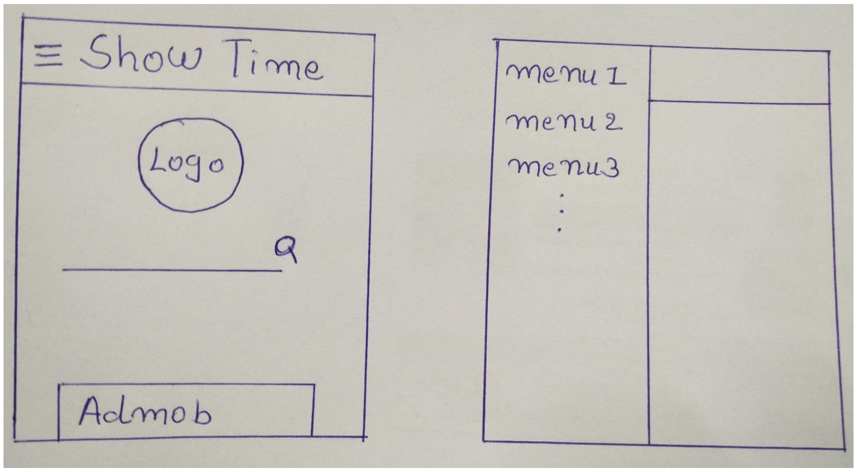
This app is intended to TV series lovers, specially to those people who want to keep track of their favorite shows, to stay always updated and to know about new series to watch.

Features

- Search TV series by keywords.
- See the details of a certain show or episode (including other users comments)
- Mark (and consult) the series (or episodes) you want to watch in the future (the watchlist).
- Mark (and consult) the series you have already seen (the collection). The user will be able to assign a grade to each show.
- Mark (and consult) the series you are currently watching.
- Receive updated info about popular TV series.
- Personally configure some application settings: default order used in the results screen, initial screen opened when starting the app, etc.

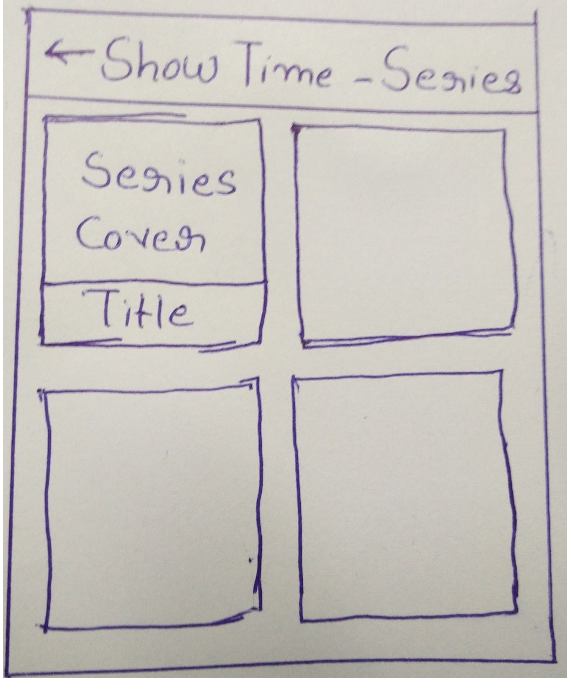
User Interface Mocks

Main screen (search)



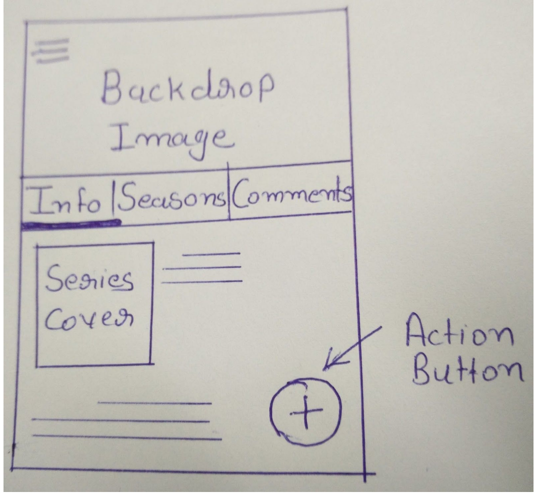
When the user introduces some search terms in the corresponding field, the app navigates to search results screen.

Search results screen



This screen will be also used for showing the user's series lists: the watchlist, the collection and the currently watching series. When the user clicks on some of the series cards, the app will navigate to the series details screen.

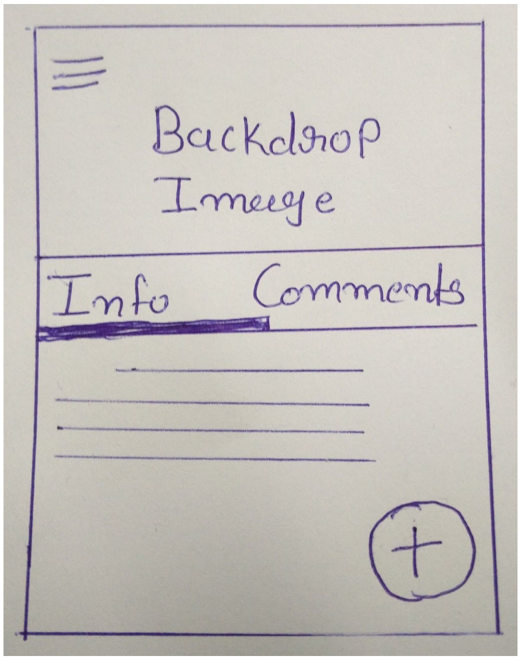
Series details screen



Seasons tab will present the list of available seasons (and their corresponding episodes) for the selected series. Comments tab will present a list of comments about the selected series, coming from trakt.tv users.

When the user clicks in some of the episodes shown on Seasons tab, the app will navigate to the episode details screen.

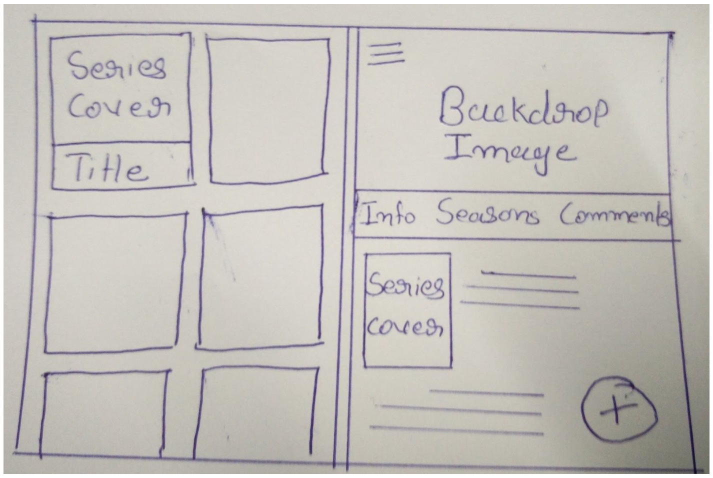
Episode details screen



Comments tab will present a list of comments about the selected episode, coming from trakt.tv users

Tablet specific screens

The app will provide an alternate layout for tablet users, in order to leverage those wider screens in a better way. Namely, in that case the search results and the details screen will be shown together, side by side:



Key Considerations

How will your app handle data persistence?

Data persistence will be handled by the use of a local database (SQLite) in the user device, with the data stored in it exposed through a custom Content Provider. Later, a Cursor Loader will be used in order to retrieve the data to be shown on the different app layouts.

Describe any corner cases in the UX.

The app implementation will try to consider any possible corner case, in order to avoid strange behaviours from the user point of view. For example, when using the app without any network available the local data will still be accessible. And when no data could be retrieved for any reason, placeholder images will be used to point out this situation. In addition, the possibility of not having any available image for a certain series or episode will be taken in consideration, in order to avoid strange or broken layouts.

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

Retrofit will be used to manage in a cleaner and more efficient way the API calls (and related JSON encoding/decoding) needed to retrieve series data. Glide will be used to handle the loading and caching of images. ButterKnife will be used for field and method binding. This way, the resultant code is much more cleaner and readable.

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

- Create a new empty Android Studio project for WatchThemAll app.

- Create a new Git/GitHub repository for the project. Remember including .gitignore and README.md files.
- Configure used libraries (following in each case the pertinent instructions), dependencies and other aspects in the Gradle files.
- Configure the use of trakt API: register the new API app on trakt.tv and the movie DB webpages in order to obtain the API keys needed to make API calls.
- Analyze thoroughly trakt and tmdb APIs documentation, in order to determine the calls that will be used to cover all the app features.
- Configure the use of the API key in the project in a way that allows easily not to include it in the repository.

Task 2: Build data persistence support

Implement all the classes needed to handle data persistence: Content Provider, Cursor Adapters, Database classes, etc

Task 3: Implement UI for Each Activity and Fragment

- Build UI for Search screen (MainActivity).
- Build UI for Search results screen.
- Build UI for Series details screen.
- Build UI for Episode details screen.
- Build tablet specific layouts.
- Build UI for About and Settings screens.

All the UI implementation will be made taking in consideration the use of Material Design guidelines, features and components (such as Floating Action Button, Floating Action Menu, Coordinator Layout, Collapsing Toolbar Layout, etc).

Task 4: Implement trakt API integration

Implement classes needed for series data retrieving:

- A SyncAdapter in order to regularly get updated info about popular TV series. This data will be used later to feed the widget provided by the app.
- An AsyncTask in order to handle the on-demand requests for searching series by keywords from the app main screen or the top search box.

Task 5: Implement Google Play Services integration

- Build Google Admob integration. The app will show banner ads in the free variant.
- Build Google Analytics integration. (if time permits)

Task 6: Accessibility and internationalization

- Ensure that the app offers a good enough experience to users with disabilities, through the correct use of content descriptions, consistent and coherent focus navigation, etc.
- Keep all the app strings in XML files, including internationalized versions for, at least, english (which will be the default language).
- Ensure that the app supports RTL layout usage.

Task 7: Implement Popular Series widget

The app must provide a widget from which users will be able to consult updated info related to the most popular TV series.

Task 8: Configure app building

- Configure app signing, including the keystore and passwords in the repository.
- Ensure that app builds and deploys using the ‘installRelease’ Gradle task.

Task 9: Future lines and improvements

I want to point out in this section some additional features that I have already planned for Show Time application. Some of them may be included (depending on the available time) in the first basic version of the app (the one that will be presented as the Capstone project in the Nanodegree). And the rest will be increasingly included in the app in the future, as I am planning to continue with its development and improvement.

- Create app variants. The app will be offered in two flavors: free and paid. The difference between them will be in the use or not of ads when the app is running.

- Implement sharing functionality. A Share Action button will be provided in details screens in order to let users share some info about their favorite shows with their friends and family.
- Implement notifications. The app will provide the option to receive daily notifications with info about the series episodes being aired each day.
- Implement a watch face for Android Wearables, in order to get the app also integrated with this new kind of devices, making the user experience even more enjoyable.