How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: “**Capstone_Stage1**”
3. Replace the text in green

Submission Instructions

1. After you’ve completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it “**Capstone Project**”
3. Add this document to your repo. Make sure it’s named “**Capstone_Stage1.pdf**”

GitHub Username: novinc

TV Companion

Description

TV Companion is an easy way for you to keep track of your shows and discover new ones. It uses trakt.tv to keep track of the shows you’ve watched and recommend new shows that you will love. It is easy to browse through popular and trending shows and add them to your shows. For each show you can not only see broad details like number of watchers and a show description, but also episode information like whether you have seen an episode or the percentage of people who loved the episode.

Intended User

This app is intended to be used by all TV lovers to keep track of their shows and discover new ones. It is especially useful for people who follow many shows.

Features

The main features

- Library of TV shows seen (down to episode watched or not watched)
- Recommendations based on watch history
- Browse shows by popular, trending, Most played
- Search for shows

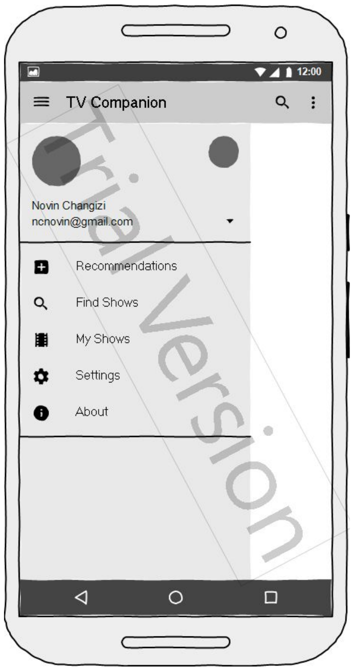
Possible future features

- Notification for new episode of a followed show
- Widget with recommendations

User Interface Mocks

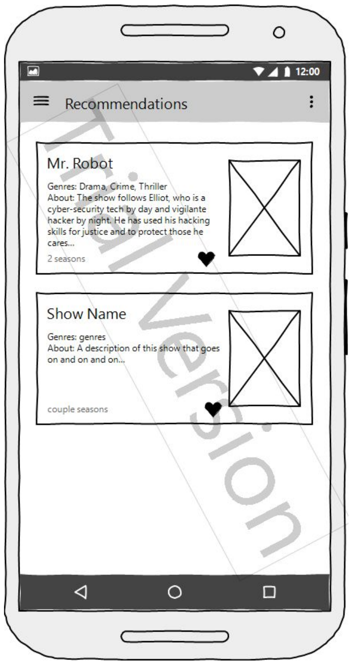
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



The main navigation drawer with trakt.tv account and links to pages in the app

Screen 2

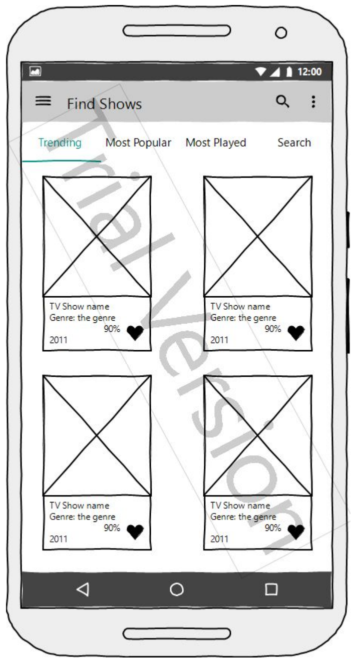


Genres: Drama, Crime, Thriller
About: The show follows Elliot, who is a cyber-security tech by day and vigilante hacker by night. He has used his hacking skills for justice and to protect those he cares...

Genres: genres
About: A description of this show that goes on and on and on...

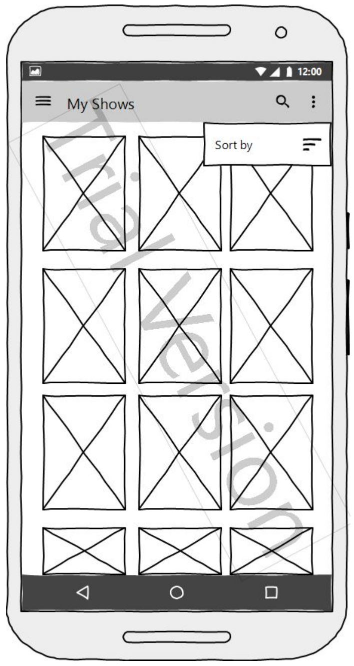
The recommendations screen. Each card is a show that is recommended and the user can click it to view the show details. Cards are larger because we want full focus on each one.

Screen 3



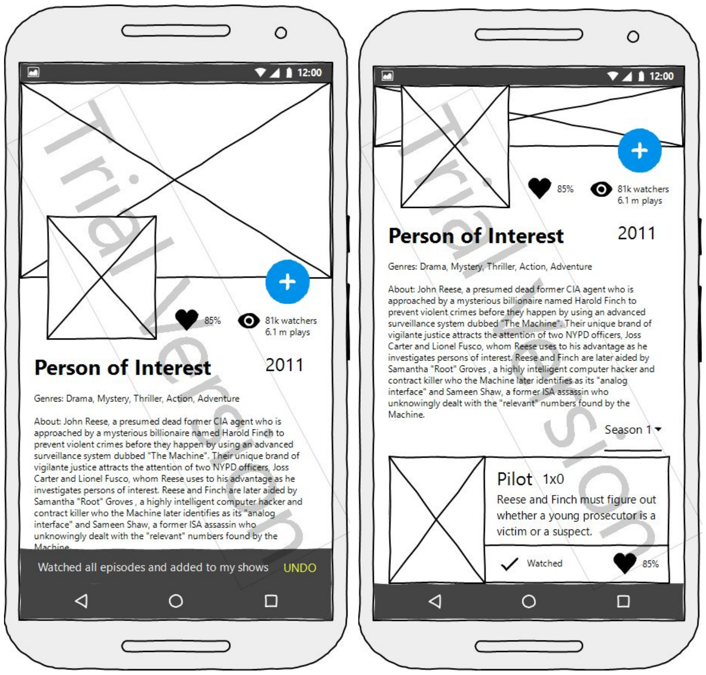
The find shows screen where you can search or look through trending and popular shows. Smaller cards because there are a lot of shows and each one doesn't deserve full attention.

Screen 4



The my shows screen. This is a library of all the shows you have seen and are watching. Small image only cards because these are shows you are familiar with and because library could be very large and we can fit a lot on screen. Searchable and sortable.

Screen 5



Person of Interest 2011

Genres: Drama, Mystery, Thriller, Action, Adventure

About: John Reese, a presumed dead former CIA agent who is approached by a mysterious billionaire named Harold Finch to prevent violent crimes before they happen by using an advanced surveillance system dubbed "The Machine". Their unique brand of vigilante justice attracts the attention of two NYPD officers, Joss Carter and Lionel Fusco, whom Reese uses to his advantage as he investigates persons of interest. Reese and Finch are later aided by Samantha "Root" Groves , a highly intelligent computer hacker and contract killer who the Machine later identifies as its "analog interface" and Sameen Shaw, a former ISA assassin who unknowingly dealt with the "relevant" numbers found by the Machine.

Watched all episodes and added to my shows

Pilot 1x0

Reese and Finch must figure out whether a young prosecutor is a victim or a suspect.

The show details screen. The fab button adds the show to library and marks all the episodes as watched. Scroll down to see seasons and episode info. Fine grained episode and season watching can be added later as it all relies on trak.tv

Add as many screens as you need to portray your app’s UI flow.

Key Considerations

How will your app handle data persistence?

The app will use the onboard database to store watched shows. The app will pull most of its information from the trakt.tv api. When the app syncs, all of the watched shows and episodes on trak.tv will be added to the database. It will most likely use two tables. One for the shows and one for the episodes.

Describe any corner cases in the UX.

On bigger devices a two pane layout should be used and the show information should be a popup instead of a full new screen. A login screen should appear before the app is used for the first time. Instructions to go find shows if the user’s shows is empty.

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

Glide used to handle image loading
Retrofit, OkHttp, RxJava used to handle communication with the trakt.tv api
Butterknife to simplify layout handling
greenDAO as an ORM for the database

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

Add libraries into gradle build
Create strings for UI elements
Create dimensions for UI elements
Create colors for app

Task 2: Implement UI for Each Activity and Fragment

Make the UI for the different pages

- Build UI for main drawer
- Build UI for recommendations screen
- Build UI for find shows screen
- Build UI for my shows screen
- Build UI for show details screen
- Build UI for sign in screen
- Build UI for settings

Task 3: Database setup

- Create shows table
- Create episodes table

Task 4: Retrofit setup

Create a trakt.tv api interface

- Follow this guide: https://guides.codepath.com/android/Consuming-APIs-with-Retrofit

Task 5: Connect UI with database and retrofit

A ton of Java code:

- Recommendations, using trakt.tv api interface to get recommendations
- Find shows, using api interface to get popular, search, etc…
- My shows, using the database
- Show details using the database to show episode watched or not and adding new shows to database (fab watches all episodes, watching and unwatching episodes can be added later as that is not a main feature)
- Make a login page for users to login to trakt.tv
- Make sign out in settings and clear database on sign out
- Sync trakt.tv data with database on sign in

Task 6: Tablet layout

Make a layout for large screens

- Two pane drawer/content mode
- Show details a popup rather than new screen

Task 7: Make production ready

Sign with key

Add as many tasks as you need to complete your app.

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"

3. Add this document to your repo. Make sure it’s named “**Capstone_Stage1.pdf**”