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: Rashwan

Reddit Client

Description

The app uses the Reddit API to present the user with posts from all the subreddits the user is subscribed to.

Intended User

Any existing reddit users or new users who want to try reddit on mobile

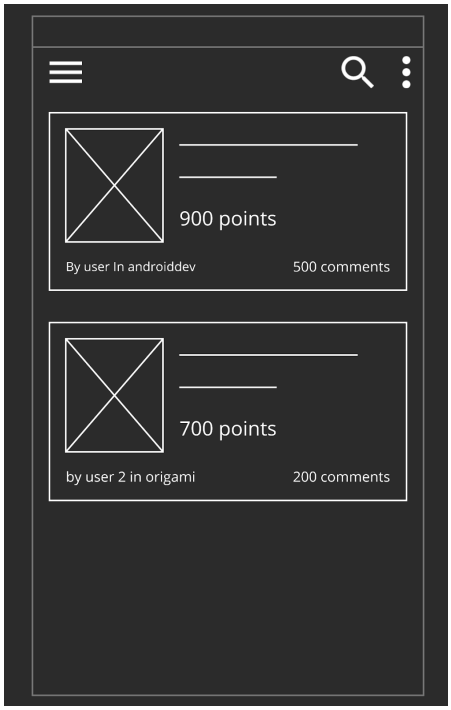
Features

- Showing the reddit front page for unregistered users.
- Users can choose a post to see its details.
- Users can search for subreddits and see its content.
- Users can login and see their subscribed subreddits .(depends on available time)
- Users can comment or +1 posts .(depends on available time)

User Interface Mocks

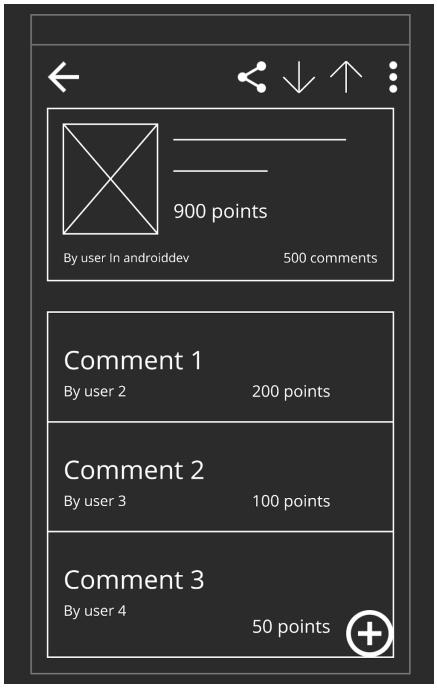
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



Main screen showing all the posts from the reddit front page . each post has an image ,title, the number of upvotes(points) , number of comments, the creator username and the subreddit which contains the the post. (The same layout can be used to show the user all posts in a selected subreddit but without showing the subreddit name in each card as it will be the same)

Screen 2



Details screen contains the selected post then all the comments on this post in scrollable view with each comment having its creator and the number of upvotes for it. The screen also contains a FAB for the user to add a comment and an upvote and downvote for the post.

Key Considerations

How will your app handle data persistence?

The app will use a sqlite database to persist data , no content provider is needed at least for initial development.

Describe any corner cases in the UX.

The app uses traditional list-to-details navigation pattern so nothing new is there. The only corner case would be the login screen (if there is time to implement it) it will use a chrome custom tab to allow the user to login then navigate back to the main screen if the login is successful.

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

- Picasso (for loading images)
- Retrofit (for interacting with the reddit api)

- Moshi (for serialization and deserialization of json)
- Okhttp (for handling network calls)
- SqlBrite (for using reactive stream semantics in sqlite queries)
- SqlDelight (for generating Java models from SQL `CREATE TABLE` statements. These models give you a typesafe API to read & write the rows of your tables)
- RxJava (Reactive extension that has many advantages in android development)
- RxAndroid (Reactive extension that has many advantages in android development)
- RetroLambda (Backport of Java 8's lambda expressions to Java 7, 6 and 5)
- AutoValue (provides an easier way to create immutable value classes, with a lot less code and less room for error)
- Dagger 2 (fully static, compile-time dependency injection framework for both Java and Android)
- ButterKnife (Field and method binding for Android views)
- Timber (logger with a small, extensible API which provides utility on top of Android's normal Log class)

Describe how you will implement Google Play Services.

- Google Analytics (to provide more insight about how users are interacting with the app).
- Google Ads (to show ads to the user).

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

- Setup project to use latest gradle & support library version.
- Setup a github repo for the project.
- Setup 3rd-party libraries .
- Build the basic MVP architecture that will be used throughout the app.

Task 2: Implement UI for Each Activity and Fragment

- Implement the ui for the main screen with mocked data.

- Replace the mocked data with real data using the api.
- Implement Details screen using the same procedure.

Task 3: Login

- Implement login flow (depends on available time)
- Implement registered user actions like commenting or +1 (depends on available time).

Task 4: Google play services

- Integrate google Ads in the app .
- Integrate google analytics in the 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**"