

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any 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.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

GitHub Username: [cjlws](#)

Reddit Explorer

Description

Reddit Explorer provides a quick and easy way to catch up on the top subreddits as well as helping you discover more about your own reading habits. By displaying easy to read stats on what you engaged with you can determine which subreddits are worth following and which ones are just clogging up your inbox.

Intended User

Users of Reddit, both new and old. For people new to the platform it provides a good way to discover subreddits to engage with. For seasoned users who have potentially built up a lengthy list of subscriptions it allows them to visualise which ones are their favourites and which ones might be worth unsubscribing.

Features

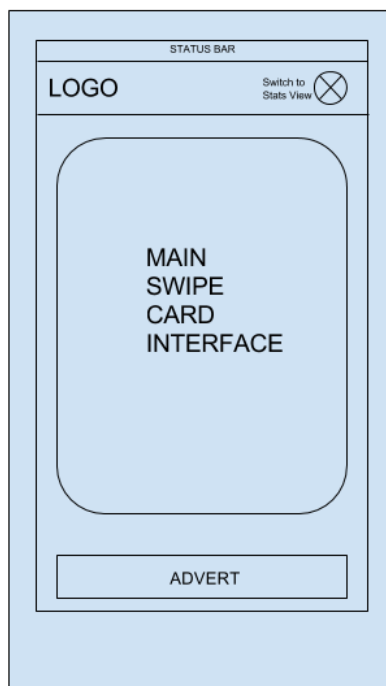
Gathers a reading list from Reddit. If you are logged in it gathers a sample of articles from your subscribed subreddits, if you are not then it gathers articles from the homepage.

Articles are presented one at a time in summary form allowing a user to either dismiss it, mark it for reading later, or read it now.

A separate stats page allows a user to see which subreddits they read the most (and therefore might want to subscribe to) and which ones they dismissed the most (and therefore probably don't want to bother with in future).

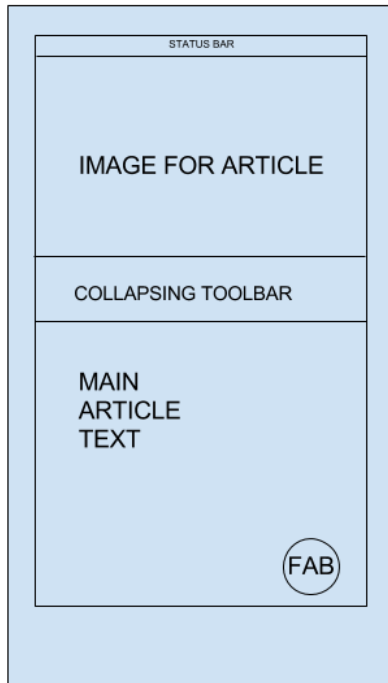
User Interface Mocks

Screen 1 - Main Phone Interface



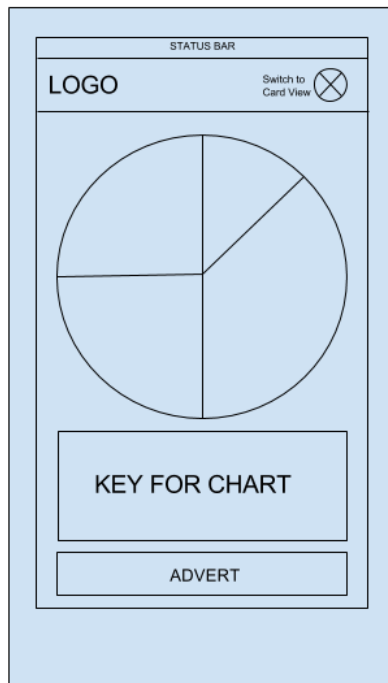
Provides the main interface for you to quickly skim through articles. The main card will show the title of the article, some basic information such as author and subreddit and the opening of the main article's text. Swiping the card off to the left of the screen will dismiss it, swiping it to the right will add it back to the bottom of the list and clicking on it will open the corresponding detail view.

Screen 2 - Phone Article Detail View



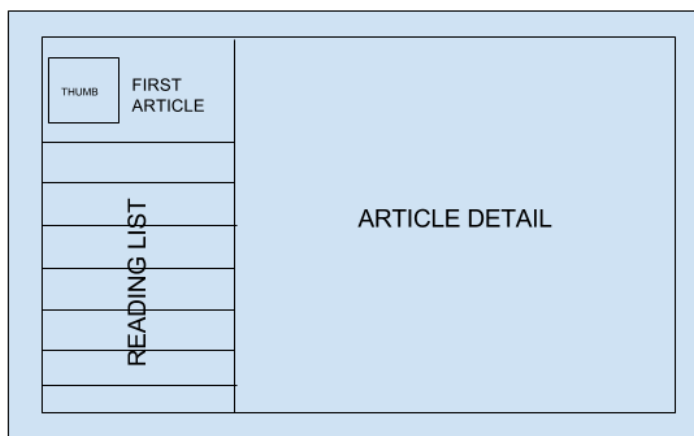
Main page for viewing the article in detail. Will display the article text itself as well as the top most comments. If the article contains media such as videos or web links then selecting them will open them in an external application for viewing via the use of intents. There will be a FAB for sharing the article.

Screen 3 - Phone Stats Screen



The stats page will display statistics on the user's reading habits, displaying insights into which subreddits they engaged with the most and which ones were dismissed the most. By using this system it will be easy to discover the most relevant subreddits to subscribe to and which ones to ignore.

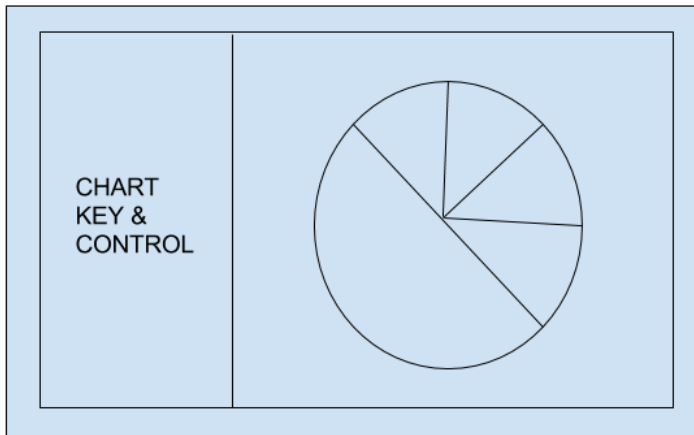
Screen 4 - Tablet List and Detail View



Tablet version of the main interface. The card system doesn't translate too well to a full screen interface so instead tablets will get a two-pane interface. The reading list will work in the same

way as the cards (ie, left and right swipes and clicks) but with the entire reading list visible at once and clicked articles opening in the right hand pane. The right hand content will be the same as for the phone detail view.

Screen 5 - Tablet Stats View



Tablet version of the stats page. The legend for the chart will be visible in the left pane and the chart visualisation itself on the right.

Key Considerations

How will your app handle data persistence?

A content provider and local database will be used to record the subreddit engagement and generate the stats. Another database will be used for the caching of articles and feeding the adapter for display. Shared Preferences will be used to store user settings.

Describe any edge or corner cases in the UX.

The first time the app is loaded there will be no data and therefore nothing on the main interface. To get past this, on the first load, the first few cards will be hard coded and act as a tutorial that the user can work through and familiarise themselves with the app.

If there is not enough data to make a meaningful chart then the stats activity will display feedback to that effect rather than load, say, a pie chart with one segment.

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

Picasso to handle the loading and caching of images.

Volley for connecting with the Reddit API.

A yet to be determined graphing library, currently likely to be MPAndroidChart.

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

Reddit API will be used to gather articles and connect to accounts using OAuth.

Google Analytics to measure app engagement.

Google AdMob to display adverts on the main page

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

- Configure Gradle files for the correct versions of libraries being used
- Gather generic admob credentials from <https://developers.google.com/admob/android/quick-start>

Task 2: Implement UI for Each Activity and Fragment

Build skeleton of app activities including:

- Main Activity with placeholder for swipe card and adverts
- Stats Activity for stats page with placeholder for charts
- Detail activity for main display of individual articles
- Tablet two-pane version of main activity

Task 3: Main Content Provider

- Build content provider to feed in articles to Main Activity. Use temporary test data to cover the main types of posts being supported (eg: video, text, etc).

Task 4: Custom View

- Build custom swipe card view to enable the app's central left/right/click interface. Wire up to the main content provider for to feed data to the swipe card, replacing the temporary test data.

Task 5: Intents

- Create intents to pass data from Main Activity to Detail View. Wire up detail view to load the entire article, parsing comments.

Task 6: Stats Database

- Create database system to store reading habits. Add dummy data. Create content provider to feed the stats display.

Task 7: Stats Display

- Create fragment for displaying charts. Wire up to database via content provider.

Task 8: Connect to Reddit

- Build Reddit api connection. Store articles to database using main content provider. Ensure card interface now displaying live data.

Task 9: Produce Stats

- Code main swipe control to produce stats data regarding article readership and feed stats database.

Task 10: Display Stats

- Integrate graphing library and add graph to stats page. Connect to stats database to use real data.

Task 11: Other

- Integrate Admob and analytics
- Create tutorial cards
- Create custom views and resource files for tablet display

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

Submission Instructions

- After you've completed all the sections, download this document as a PDF [File → Download as PDF]
 - Make sure the PDF is named "**Capstone_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"