

How to Use this Template

1. Create a new document, and copy and paste the text from this template into your new document [Select All → Copy → Paste into new document]
 2. Name your document file: “**Capstone_Stage1**”
 3. Replace the text in green
-

Description

Intended User

Features

User Interface Mocks

Screen 1 Phone Version

Screen 2 Phone Version

Screen 3 Tablet Version - Widget

Key Considerations

How will your app handle data persistence?

Describe any edge or 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 or other external services.

Next Steps: Required Tasks

Task 1: Project Setup

Task 2: Implement UI for Each Activity and Fragment

Task 3: Implement Reddit communication

Task 4: Connect Services

Task 5: Testing and APK in Google Play Store

GitHub Username: [benepell](#)

RcapStone

Description

A beautifully client for browse [reddit](#) discussion with an easy to use interface.

Posts are organized by subject into user-created boards called "subreddits", which cover a variety of topics including news, science, movies, video games, music, books, fitness, food, and image-sharing. Registered users can upvote or downvote posts which will make them more/less visible to other users.

Intended User

Get the Best of What Really Interests You

The app's designated for all users, is present adult content restrictions, however it can be disabled. There's a subreddit about everything, everyone to post, share, vote and discuss just about every topic and subject imaginable. Share your passion for your favorite books, movies, TV shows, music and technology with millions of users in thousands of subreddits.

Features

List the main features of your app.

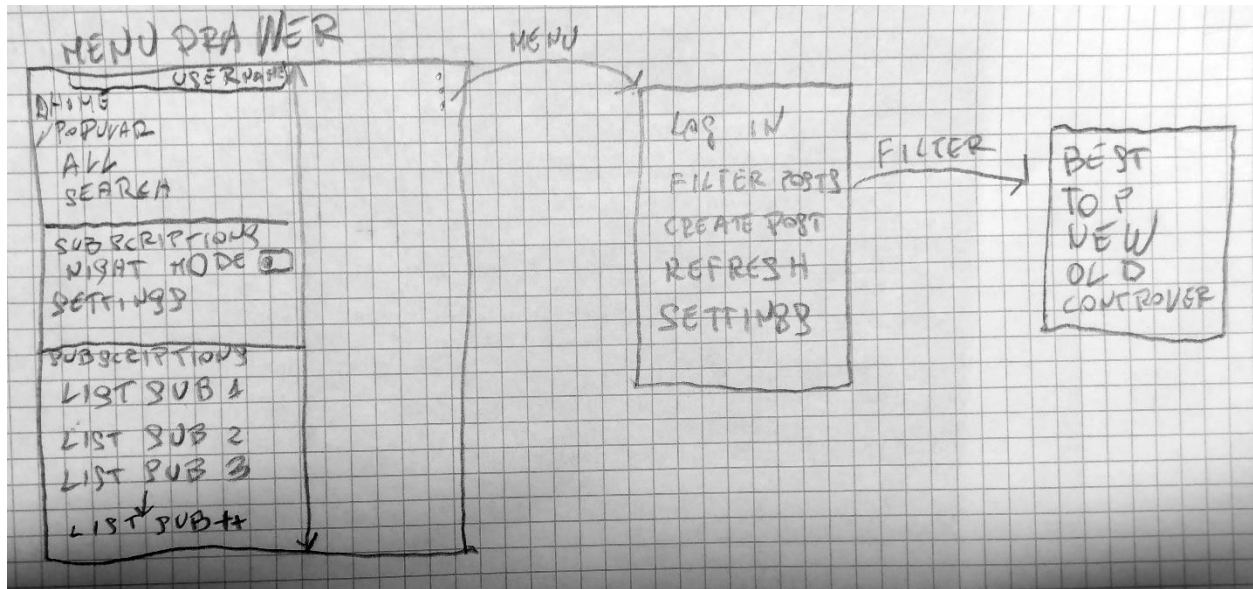
- Connect to Reddit [API](#)
- Automatic data saving options based on network
- Subscribe to subreddits or add to favorites
- Presents posts from selected subreddits.
- Intuitive Material Design user interface
- Restrictions by adult content (enabled by default)
- Filter by subreddit, domain, author or keywords in titles
- Filter posts by content type: image, album, gif, video, text posts or links
- Media previews: images and videos with download and share options
- Switch between day and night modes
- Scroll comments with volume keys
- Buttons to navigate between comments
- Swipe back from comments instead of using the back button
- Split-screen mode for tablets
- Beautiful home screen **Widget**
- Add free and paid flavors to an app
- Banner ads will be shown only in a free version

- Google analytics for tracking

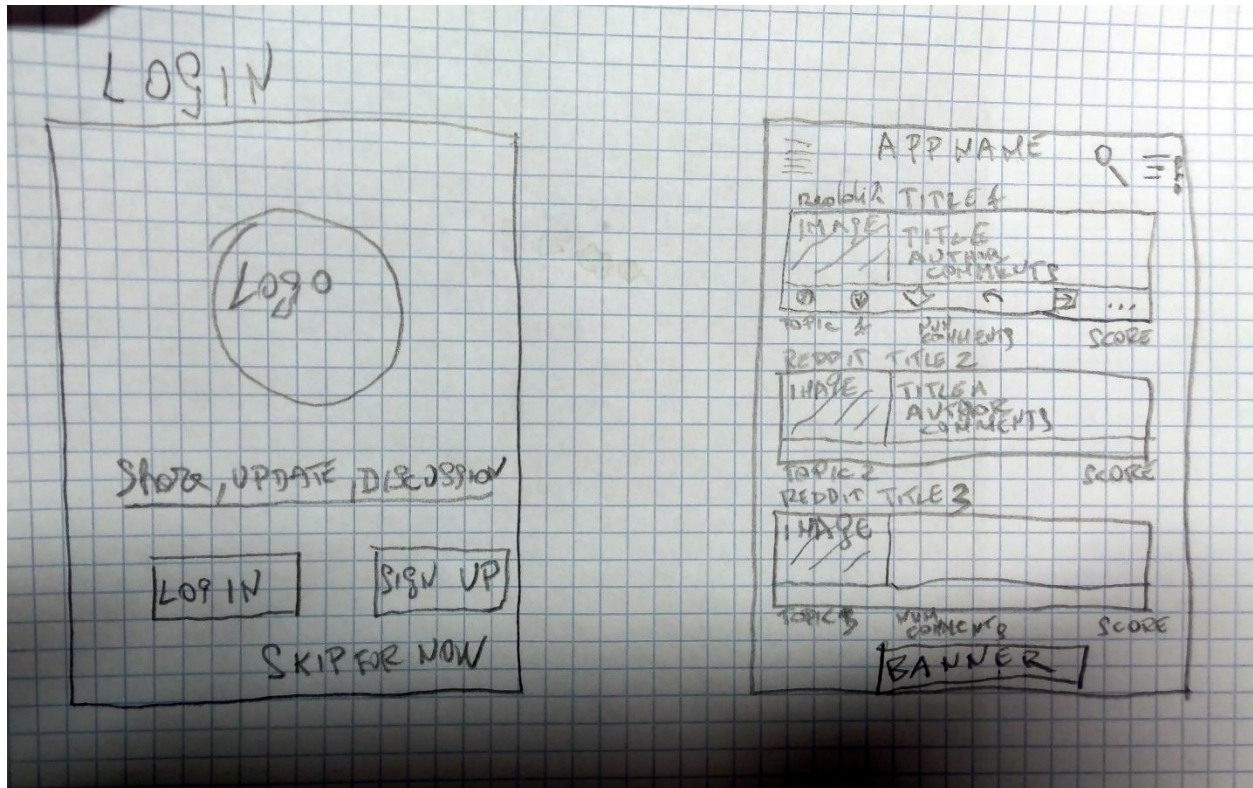
User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1 Phone Version

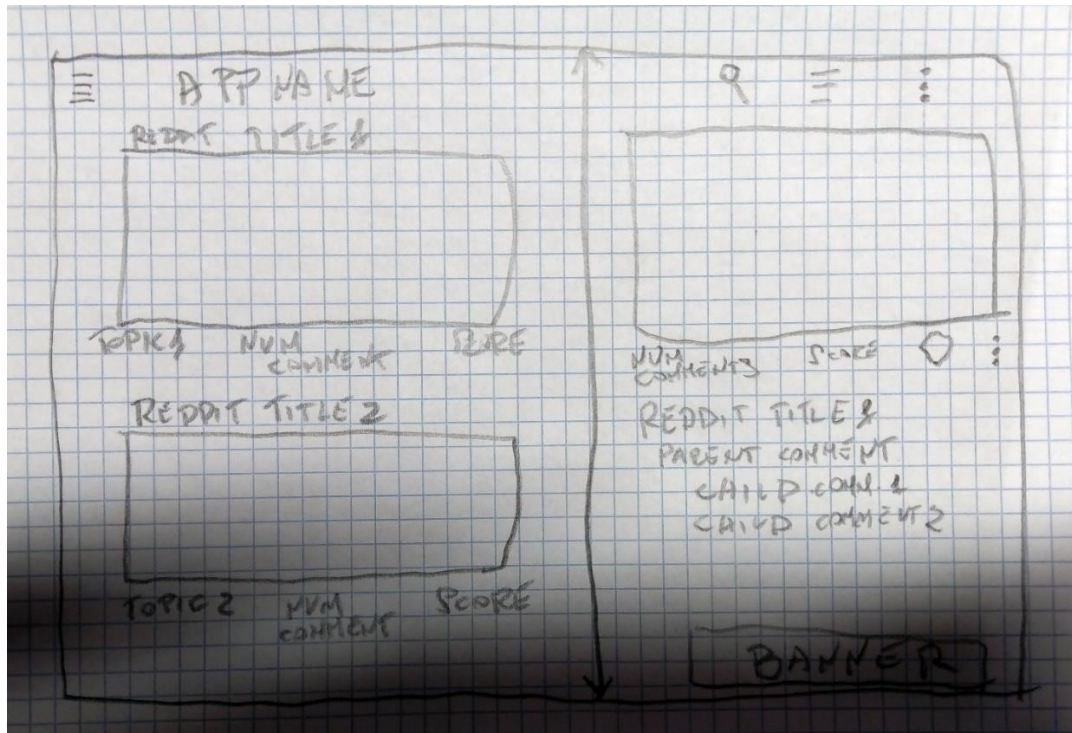


Screen 2 Phone Version

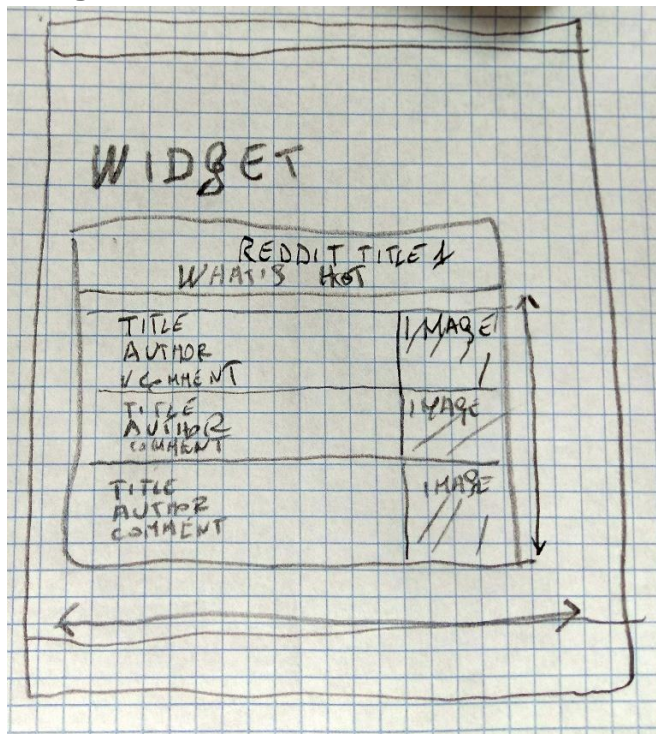


Add as many screens as you need to portray your app's UI flow.

Screen 3 Tablet Version - Widget



Widget



Key Considerations

How will your app handle data persistence?

- Shared Preferences will be used to store the user's custom list of subreddits.
- A content provider will be created to manage the favorite's SQLite database.

Describe any edge or corner cases in the UX.

- If there is no network the app will display a Snack Bar message warning the user. If a user enters an invalid subreddit they will be warned by a Snack Bar/Toast message.
- A Navigation Drawer will provide access to the different functions.
- When required an icon will generated material design iconic font.

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

- Android Design Support Libraries – Material based design and other fantastic features.
- Retrofit 2 – Used to download data from the data source, simplifies http client and fetching JSON from URL
- Glide – Automatic memory and disk caching provides optimized image download, transform images to better fit into UI layout, supports preview and error placeholder images, downloading it in separate thread, optimize with OkHttp integration for bandwidth and faster loading.
- EXOPLAYER: Movies player with bandwidth and caching limit.
- Butterknife – Optimize code with DataBinging and save lines code.
- Timber Log.: Optimize code for logfiles.
- Apache-common-io: library for files and folder.
- mikepenzIconics and mikepenzMaterialDesignIconics: get icons UI Layout optimized

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

- Firebase Services Admob to display test ads
- Firebase Services Analytics to analyse user behavior.
- firebase-jobdispatcher add Service sync / schedule
- firebase test lab - Test app on devices hosted in a Google data center.

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

Create a project with minimum API level 21 (Lollipop),

Set uses-permission

INTERNET,ACCESS_NETWORK_STATE,RECEIVE_BOOT_COMPLETED,WRITE_EXTERNAL_STORAGE in AndroidManifest.xml

Set provider, service firebase, service widget, receiver in AndroidManifest.xml

Set Activity in AndroidManifest.xml

My setup for the projects implementation the library:

- [Glide](#)
- [Exoplayer](#)
- [OkHttp](#)
- [Retrofit 2](#)
- [Butter Knife](#)
- [Apache Common Version](#)
- [Android-Iconics](#)
- [Firebase jobdispatcher](#)
- [Firebase Analytics](#)
- [Firebase Ads](#)
- [Firebase Crash](#)

Task 2: Implement UI for Each Activity and Fragment

- Build UI for drawer's list of subreddits with an option to subscribe/unsubscribe
- Build UI for drawer's Login and Logout
- Build UI for Main activity with list of post
- Build UI for detail activity and detail fragment
- Build UI for advanced menu settings
- Build UI for login

Task 3: Implement Reddit communication

- Implement reddit API with retrofit 2. Documentation can be found in <https://www.reddit.com/dev/api>
- Login to reddit site, with user's customizes
- Store user credentials in md5
- Updating list subscription to reddit
- Filter top post data with parsing JSON – (use: retrofit 2)
- Get information for the view, thumbnail images, number of comments, etc.
- Get detail information for the view selected with images, comments, video, etc.
- Manage the list of subreddits with shared preferences

Task 4: Connect Services

- Connect Retrofit endpoints
- Implement image library with control cache, error communication, clear cache, cache size.
- Connect firebase service from Firebase console : ads (get google-service.json and configure it) , analytics, job-dispatcher, crash
- App updates data in its cache at regular intervals using a job-dispatcher.

Task 5: Testing and APK in Google Play Store

- Implement Unit tests and functional UI tests Espresso libraries.
- Firebase Test Robo
- Improve code with Lint
- Generate signed APK and publish in Google Play Console
- Create alpha/beta/stable project in google play console

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**"