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: jcoro

Quizlet Math Plus

Description

The Quizlet Math Plus app allows user to add mathematical formulas, links, and custom branding to their Quizlet.com flash cards.

Intended User

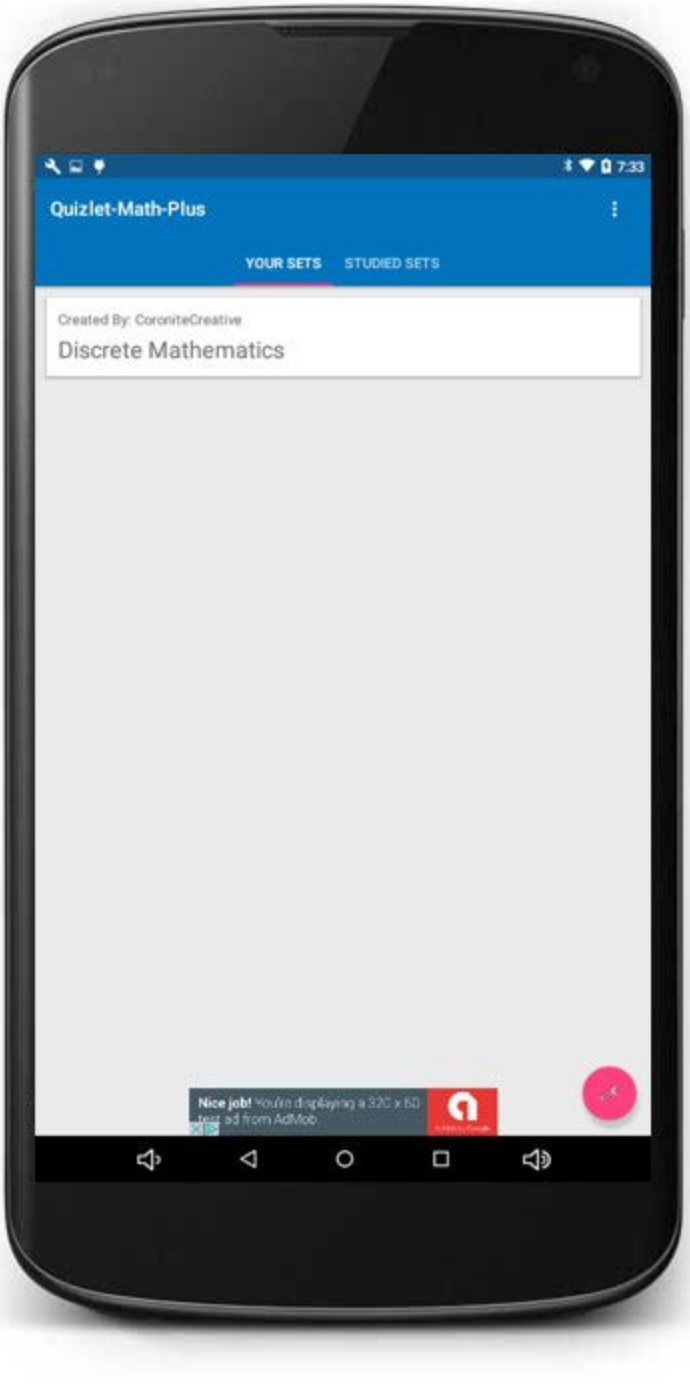
Quizlet Math Plus is perfect for students and teachers looking for an added level of functionality from their Quizlet.com flash cards.

Features

- Uses the MathJax JavaScript library to display mathematical formulas in Quizlet.com flash cards.
- Allows teachers and students to add links to relevant online content for each card.
- Allows the creator of the card set to add “branding” links for businesses and schools.
- Toggles the flash card term and description in the same view, so both are visible at the same time.
- Allows the user to display either the term or definition with the other hidden.
- Displays two lists of flash card sets--those created by the user and those currently being studied by the user.
- Properly displays “standard format” flash card data as well.

User Interface Mocks

Screen 1

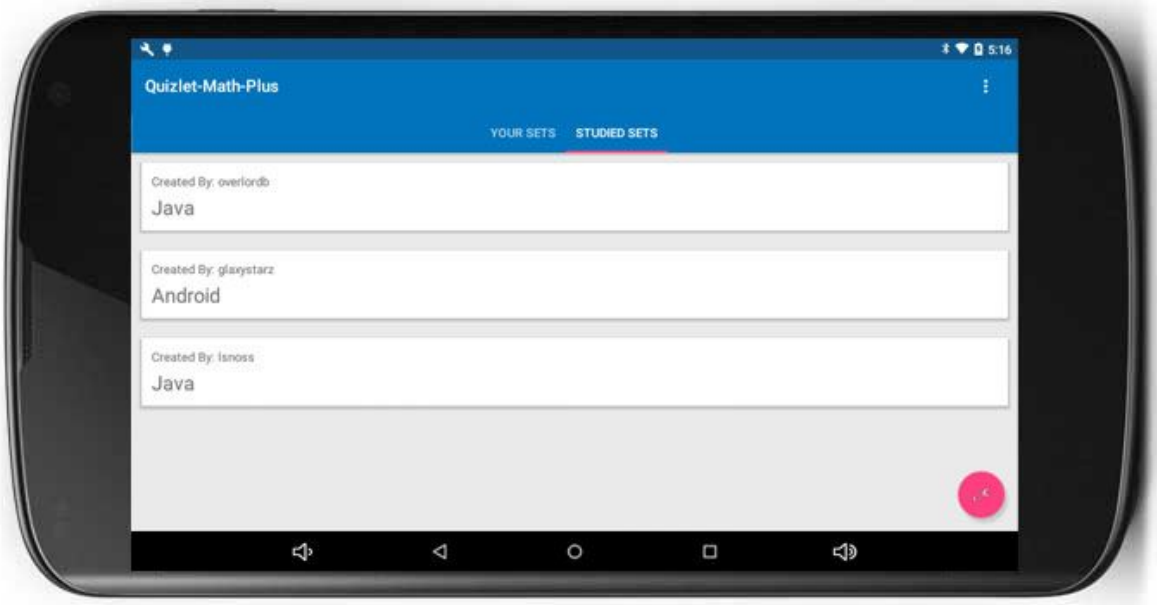


The MainActivity contains a tab layout where users can swipe between groups of flashcard sets.

Screen 2

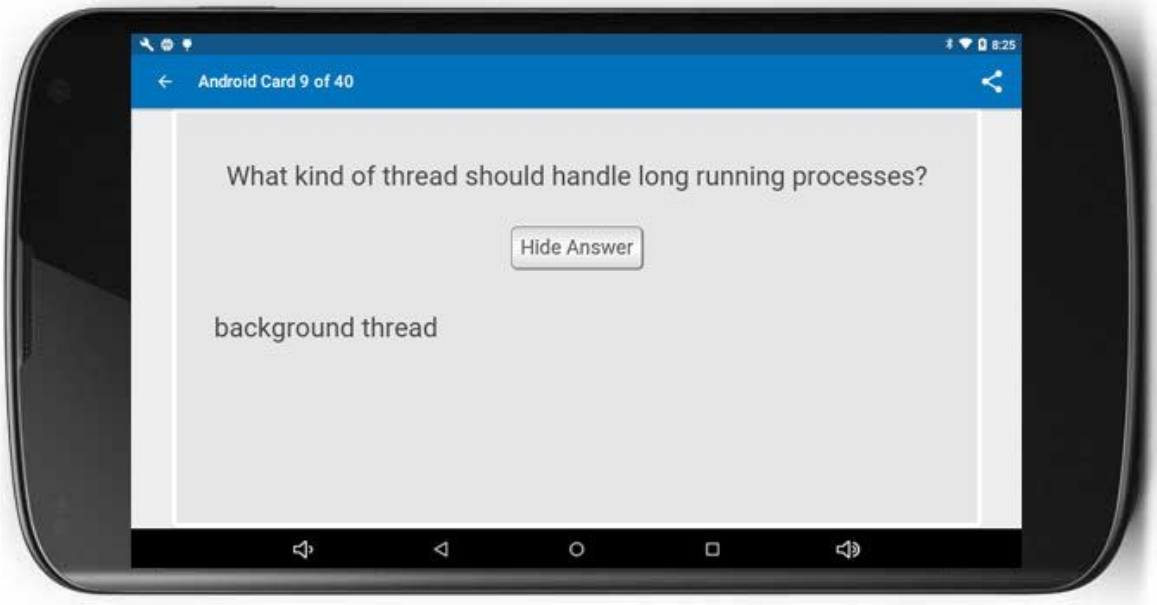
The DetailActivity contains a ViewPager for the user to swipe between cards. The view is a custom WebView with links to relevant coursework and branding. The user can toggle the answer by hitting the Show Answer button, and the user can share the card data via a share Intent.

Screen 3



Users can swipe between sets they have created and those they are studying.

Screen 4



Card Sets without mathematical formulas (like the one above) display normally.

Key Considerations

How will your app handle data persistence?

Quizlet Math Plus contains a SQLite database, Content Provider, and Syncs data via a SyncAdapter.

Describe any corner cases in the UX.

The Settings menu is available from the MainActivity (both as a FAB and as a Toolbar Menu). This allows the app to refresh data from Quizlet if the username is changed. If the option to show definitions and hide terms is selected, the user's flashcard sets are not automatically refreshed--this change changes the UI in the DetailActivity.

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

Quizlet Math Plus uses the Retrofit2 library to facilitate the retrieval of flash card data from the Quizlet.com API. The app also includes a highly-customized version of a MathView library to create a custom WebView to utilize the MathJax JavaScript library for displaying mathematical formulas. The original MathView library on which the app's library is based can be found here: https://github.com/kexanie/MathView

Describe how you will implement Google Play Services.

Quizlet Math Plus uses Firebase AdMob to display test ads and analytics to track the apps usage.

Next Steps: Required Tasks

Task 1: Project Setup

Where this app requires data from the Quizlet.com API, it requires that data be entered into Quizlet in a custom fashion. It also requires the creation of a custom library to display mathematical formulas and links on each flashcard.

Steps to accomplish this are as follows:

- Create a JSON data structure for Quizlet data which will then be parsed by the app.

- Create a custom MathView library which includes the HTML and JavaScript (including a custom MathJax.js implementation) for rendering mathematical formulas in the app.

Task 2: Build the MainActivity

Build the MainActivity:

- Build the UI for the MainActivity which includes a RecyclerView listing the User's flash card sets from Quizlet.
- Implement the Retrofit2 library to access the user's flash card sets from the Quizlet.com API.
- Create a TabLayout which will allow the user to swipe between the flash card sets they've created and the ones they are currently studying.
- On the first launch of the app, prompt the user to enter a Quizlet username.

Task 3: Create Parcelable Data Models

To efficiently handle the data received from the Quizlet.com API, requires parcelable data models for:

- User-created flashcard sets
- The user's studied sets
- Flash card terms

Task 4: Implement Database, ContentProvider, and Loaders

- Implement a SQLite Database for storing Sets
- Implement a ContentProvider for database access.
- Implement a loader in the DetailActivity to display the card data.

Task 5: Create The SyncAdapter

- Implement a SyncAdapter to check for updated Quizlet data periodically.
- Save the data in the database.
- Allow the app to fetch fresh data immediately if the username is changed.

Task 6: Create the PagerView

When the user clicks on a flash card set, they are taken to a PagerView which allows them to swipe through each card in the set.

Task 7: Create the Custom WebView

- Create a custom WebView (HTML, CSS, and JavaScript) which will display the flashcard data.
- Include the Chunk template engine for Java to pass data to an HTML template rendered in the WebView.

Task 8: Implement SharedPreferences

- Allow the user to enter and alter their username.
- Allow the user to show either the flash card set term or definition first, while hiding the other.

Task 9: Implement Google Play Services

- Implement AdMob.
- Implement Firebase Analytics to test some user interactions.

Task 10: Optimize UI

- Alter card width on large landscape screens.
- Optimize text size for readability.

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**"