

Contents

Description	1
Intended User	2
Features.....	2
User Interface Mocks	2
Screen 1	2
Screen 2.....	3
Screen 3.....	3
Key Considerations.....	4
How will your app handle data persistence?	4
Describe any corner cases in the UX.	4
Describe any libraries you'll be using and share your reasoning for including them.	4
Next Steps: Required Tasks.....	4
Task 1: Project Setup	4
Task 2: Implement UI for Each Activity and Fragment	4
Task 3: Data Model and Database	5
Task 4: Implement Code for Wearables.....	5
Task 5: Others.....	5
Task 6: Testing.....	5

GitHub Username: sanchayitam@github.com

Vocabulary App

Description

Vocabulary App displays flashcards of random English words. On click of the word flashcard, you see the **definition** and **synonyms** for that word as well as an Audio pronunciation. These synonyms are very helpful especially for those who are looking for a way to expand their vocabulary. You can also mark words as favorite and have access to them from “Favorites” section of the app.

Word of the day is a great feature runs on hand held mobile and wearables.

Intended User

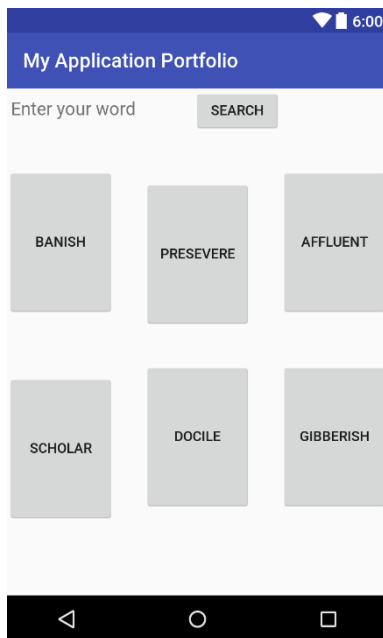
Students

Features

- Random Word Generator
- Instant access to Dictionary
- Definitions and Synonyms
- Word of the day
- Home Screen widget
- Mark words as favorite
- Show phonetics and audio pronunciations

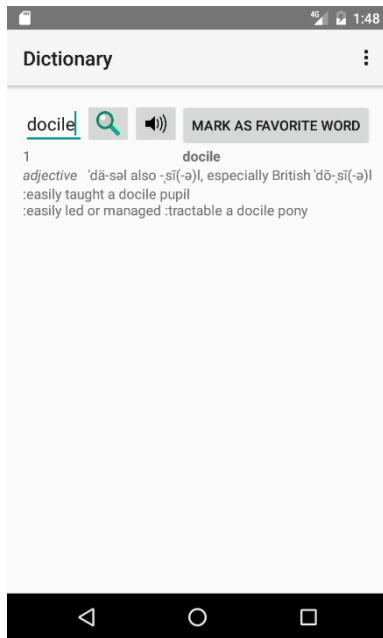
User Interface Mocks

Screen 1



The main activity screen displays random words and dictionary look up. Menu drop down option includes “Refresh” and “Show Favorite Word List”

Screen 2



On click of the word displays its pronunciation, synonyms, definition.

Screen 3



Word of the day displayed on watchface.

Key Considerations

How will your app handle data persistence?

App implements a Content Provider to access locally stored data. The app pulls data from API and uses an Intent Service. It also uses an Async Task.

Describe any corner cases in the UX.

3D GridView

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

Latest Google Play Service
Android Wearable

.

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

Update Android Studio with latest version
Update SDK with Google Play services
Configure libraries
Register Merriam Webster Developer Center for an API account
Generate API key

Task 2: Implement UI for Each Activity and Fragment

Subtasks for the project:

- Setup Menubar
- Build top part of each activity/fragment (AppBar, coordinatorLayout)
- Build Master/Detail for Tablet Support
- Build UI layout for multiple Activities.
- Launch these Activities via Intent.
- Fetch data from dictionary API

Task 3: Data Model and Database

Build up the data models and implement data persistence.

Subtasks:

- Create data model classes
- SQLite database setup
- Loader/adapters
- Notification

Task 4: Implement Code for Wearables

Incorporate “Word of the Day” functionality into an Android Wear watchface.

Task 5: Others

Subtasks:

- Widget
- Accessibility
- RTL

Task 6: Testing

Testing Subtasks:

- Rotation
- Phone vs. Tablet
- Performance

Task 7: Google Service Play

- Admob for advertisements and generate revenue
- Google Analytics to track and report traffic

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**”