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

GitHub Username: lwach87

Earthquake application

Description

Application which helps you track latest earthquakes from around the globe by showing latest earthquakes of 500+ significance level. Data is shown on maps, list and application also shows earthquake of the highest magnitude near your place of living.

Intended User

General audience, all people interested in earthquakes.

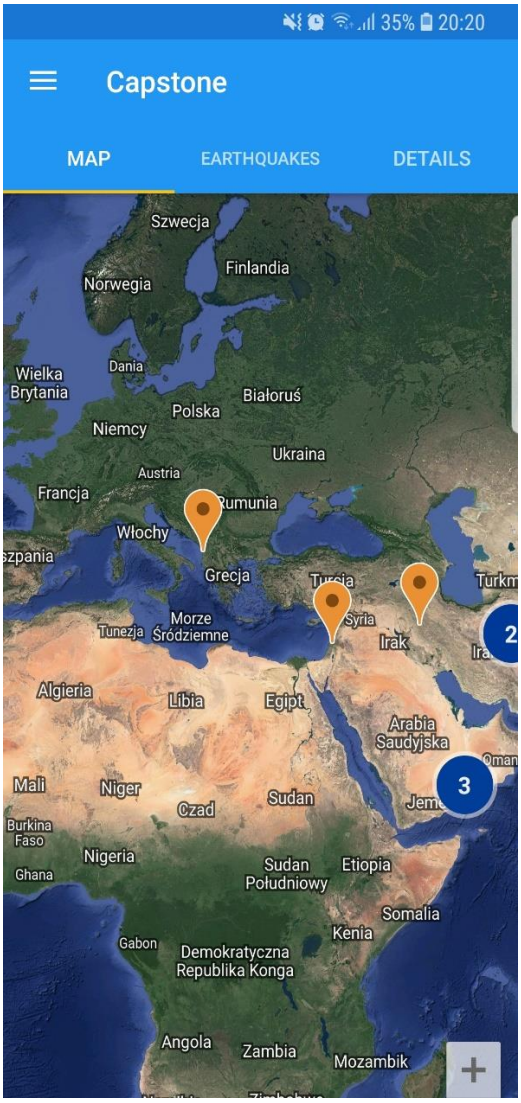
Features

List the main features of your app. For example:

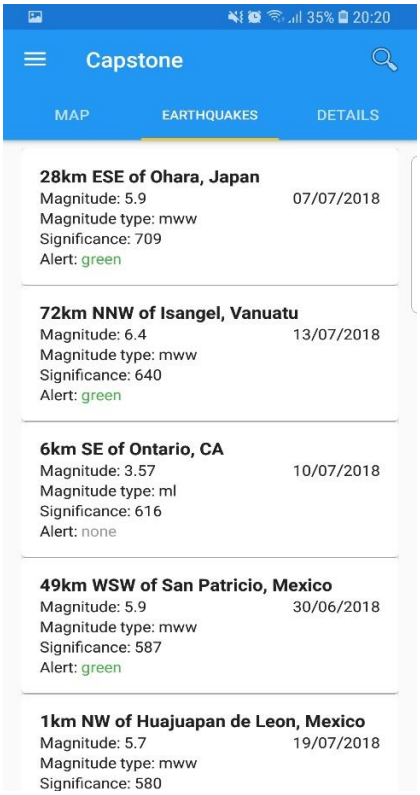
- Shows up to 25 earthquakes of the highest magnitude on map
- Has a list containing details of each earthquake
- Searching for the highest earthquake near your place of living and displaying all the details

User Interface Mocks

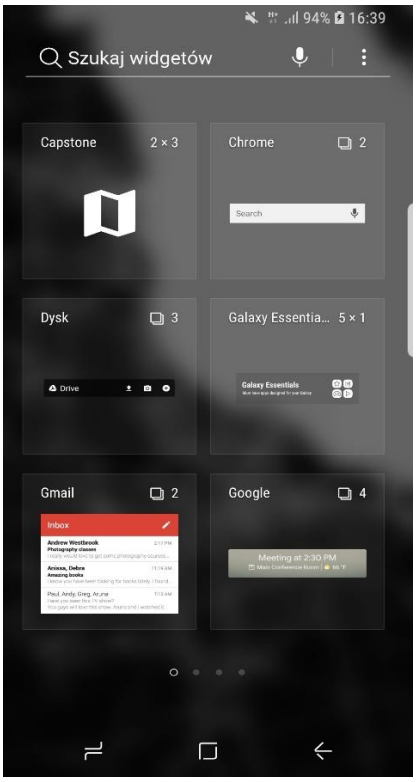
Screen 1



Screen 2



Widget icon



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

Key Considerations

How will your app handle data persistence?

Data will be stored in Room database.

Describe any edge or corner cases in the UX.

- When offline, app will be showing snack bar with offline statement
- For offline situations app will be displaying data from database
- App will be only available when user accept gps permission

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

- Gson for Json parsing of response data
- Retrofit2 combined with RxJava2 for network calls
- Android Architecture Components for persisting orientation changes in application and for offline mode.
- Dagger2 for dependency injection
- Firebase JobDispatcher for periodical calls to server

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

- Google Maps API services
- Google Location services
- Firebase JobDispatcher

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 project
- Configure libraries
- Add permissions
- Get Google API key from console
- Enable API services
- Get configuration files for google services and add it to app

Task 2: Connect to network and download the data

- Create database, dao and repository
- Create network call using retrofit in repository and save data in database
- Add Firebase's JobDispatcher for update the data at regular intervals
- Create View Model for an activity
- Use Google Location service to get current location

Task 3: Configure dagger and create adapters

- Create modules and components for dependency injection
- Implement View Model Factory in Main Activity
- Create three fragments and View Pager Adapter
- Create List Adapter for second fragment
- Implement repository in each fragment

Task 4: Implement UI for Each Activity and Fragment

- Show data on map in first fragment
- Fetch data with recycler view in second fragment
- Fetch third fragment with the highest magnitude earthquake
- Calculate shortest distance between you and latest earthquakes

Task 5: Add widget and improve UI

- Build widget to show earthquake with the highest magnitude
- Handle critical cases like gps off, internet off
- Code refactor
- Make App production ready

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**"