
GitHub Username: nasreekar (https://github.com/nasreekar)

GEONOTES

Description

A simple mobile application that allows users to save location based notes on a map. These notes can be displayed on the map where they were saved and viewed by the user that created the note.

Intended User

Anyone who wants to save quick notes with a twist (location included)

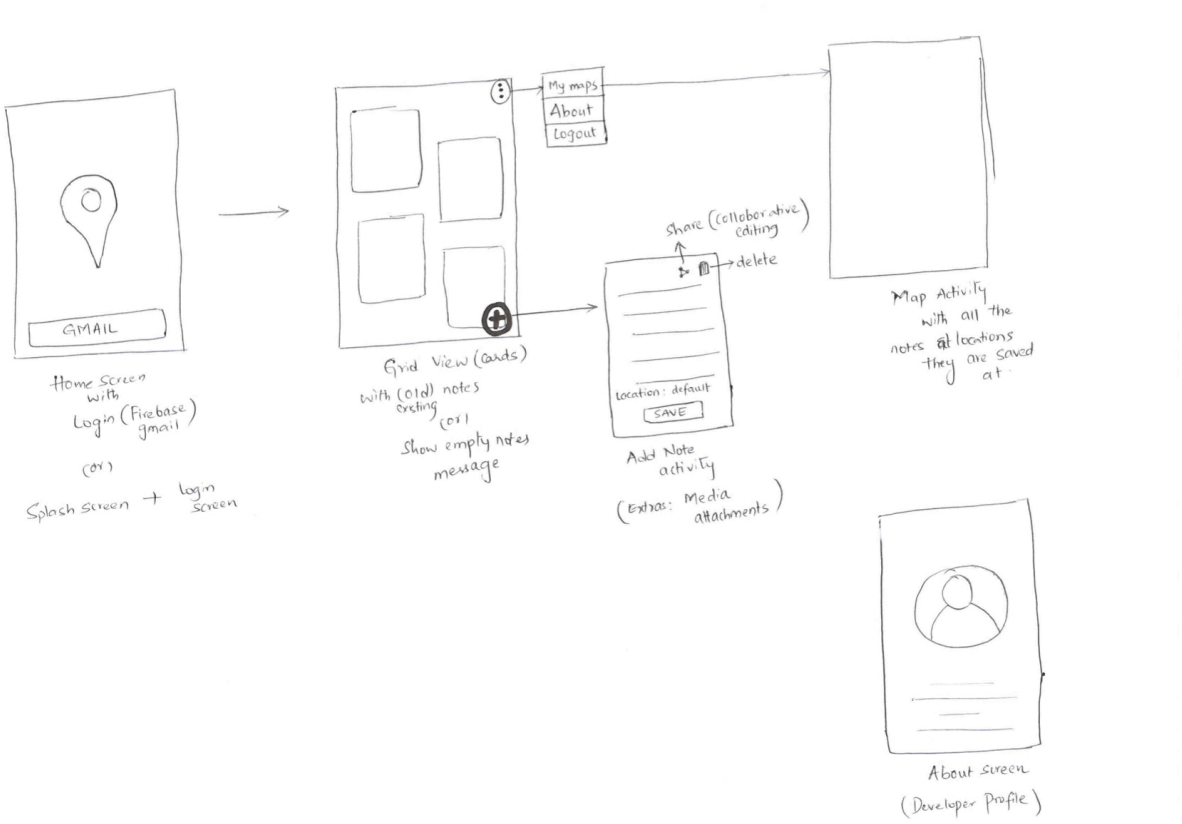
Features

List the main features of your app. For example:

- Google+ login
- Display list of notes
- Location Based
- Google Maps Clustering

User Interface Mocks

High level flow of the app:

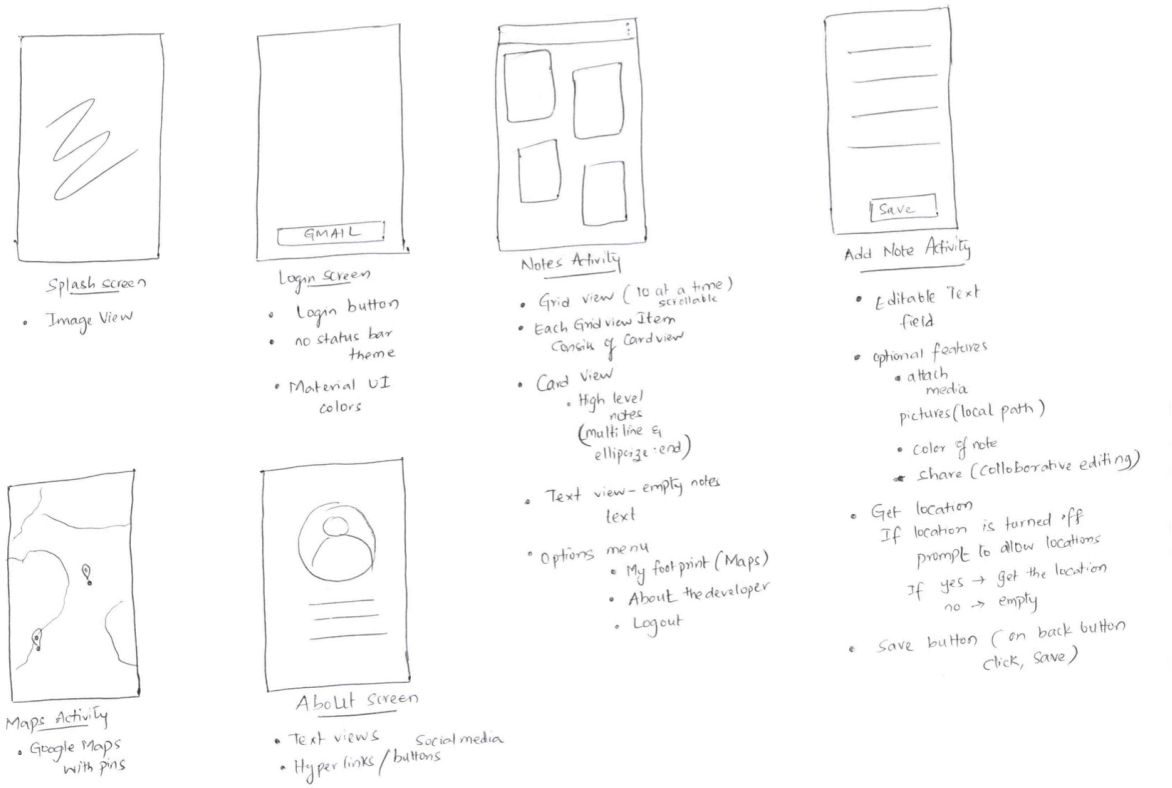


The above picture explains the high level flow of the app. Major components of the app are:

- Login Functionality
- Add Notes
- Location feature
- Google maps with markers
- About the app

Sample Demo of the prototype: https://vimeo.com/user101382508/review/351366972/a5cfbdb578

Screen 2



A brief explanation of how the app works:

Upon login using Gmail account (Firebase), the user will be navigated to home screen which displays the list of all previous notes he has added (if there are no notes added previously, respective text message will be displayed). The Notes List screen also have a FAB button which allows the user to add a new note (location will be collected when location permissions are given).

Home screen will also have options menu with My Footprint and About Developer CTA's. Upon clicking the MyFootPrint button, user will be navigated to Maps Activity with all the notes tagged at the locations they are created at. The pins are color coded to differentiate own notes and other users notes.

About Developer is an activity which contains the developer details and social media links.

Key Considerations

How will your app handle data persistence?

For the MVP, data persistence is handled using Firestore

Describe any edge or corner cases in the UX.

Following the existing notes available apps in market, hitting the back button on Adding Notes activity will behave similar to save button on the screen provided there is some text.

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

For MVP, media attachments are not considered and so no Image caching and download library is being used. However for the splash screen animation, I'm considering to use Lottie library.

For view injection, I'm considering to use Butterknife.

Retrofit - Type safe HTTP client library used to wrap around any Web API calls

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

As the app is location based , will be using google maps SDK and Admob for banner ads.

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 new Android Studio project using the following information:
Application name: GeoNotes
Company Domain: nanodegreeapp.abhijith.geonotes.com
Package name: com.abhijith.nanodegree.geonotes
Platform : Phone and Tablet Minimum SDK: API 16 - API 28

2. Modify build.gradle file in app folder to include all the dependencies libraries.

Task 2: Implement UI for Each Activity and Fragment

List the subtasks.

- Build UI for SplashActivity
- Build UI for LoginActivity
- Build UI for NotesListActivity
- Build UI for AddNote Activity
- Build UI for AboutMe Activity
- Build UI for MyFootprint Activity

Task 3: Implement Google play services

1. Add identity verification using google play services to the Login Screen
2. Add an add banner to the add note screen
3. Add Google maps

Task 4: Implement Data layer

- Implement Firestore structure
- Add a layer in the code to call all the required data

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**"