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: exitmusic

LocalLore

Description

LocalLore allows you to read and create short stories tied to your surroundings.

Intended User

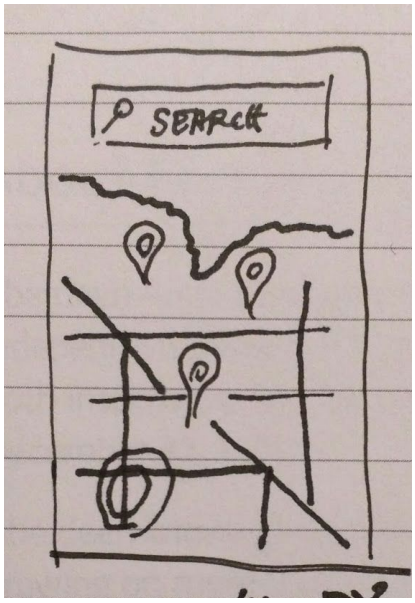
The intended users are curious travelers with an interest in local stories.

Features

- Browse stories using a map interface
- Add new stories and pin them to a map

User Interface Mocks

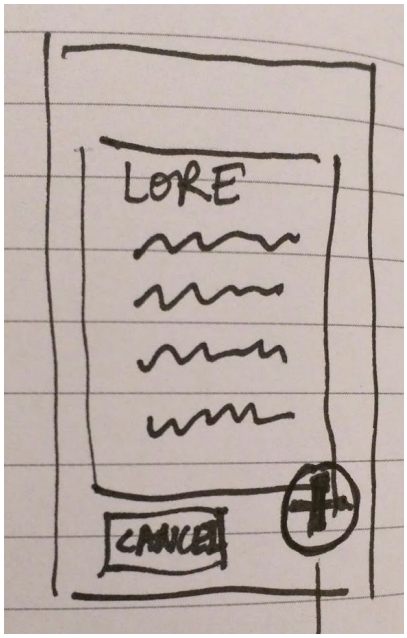
Screen 1



Main screen

- Google maps interface
- Browse existing story locations
- Tap a pin to bring up story text
- Press and hold a new location to bring up the add story screen
- Search for location to recenter map

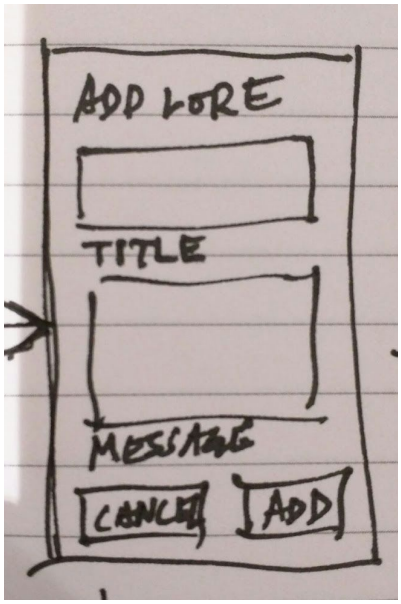
Screen 2



Read story

- Tapping on a pin on the main screen opens up the story reader
- Tap + to open up the add story screen to add another story to the same location
- Tap cancel to exit back to main screen

Screen 3



Add story

- Press and hold a new location on the main screen opens the add story screen

- Add title
- Add story text
- Tap add to save
- Tap cancel to exit back to previous screen (main or read story)

Key Considerations

How will your app handle data persistence?

It will store and access data using a ContentProvider backed by a local SQLite database, using Schematic.

Describe any corner cases in the UX.

Tapping on cancel in the add story screen should prompt user before exiting because the app will not save stories in progress.

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

Butterknife - field and method binding for Android views
Schematic - generate a ContentProvider backed by a SQLite database
Retrofit - make HTTP API calls using a Java interface

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

- Add dependencies to the gradle file and configure libraries
- Register for Google Services API keys: Location and Maps

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity (main maps screen)

- Build UI for ReadStoryActivity
- Build UI for AddStoryActivity

Task 3: Build Content Provider

Use Schematic to build the content provider for saving user stories

- Research how to use Schematic
- Create data model

Task 4: Set Up Google Maps Services

Integrate Google Maps interface with main screen

- Research how to use Google Maps
- Research how to use Google Maps API
- Integrate Maps with main screen

Task 5: Implement Add Story Screen

Add logic to the add story screen

- Wire up save button to use the content provider

Task 6: Implement Read Story Screen

Add logic to the read story screen

- Wire up read story screen
- Use a CursorLoader to surface data from the ContentProvider to the View

Add as many tasks as you need to complete your app.

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**"