
GitHub Username: ram101

Travel Locker

Description

Write a brief summary of what your app does. What problem does your app solve?

Not sure how to write a good description? Search 5-star apps on the Play Store for inspiration.
Do you travel a lot ?, Are you hesitant about plotting locations on your ordinary maps app, does your recent travel location get buried in your map's travel history, fear not this is the app just for you. Travel Locker saves all your personal locations offline within the app. Using travel locker you can navigate to a location from the app using google maps. The personalised content you store within the app regarding the location would only be stored inside the app and nowhere else. We respect your privacy, so we make sure that the content is not shared with anyone, ever.

Intended User

Who is your intended user? (For example, is this an app for dog owners? Families? Students? Travelers?)
Travellers

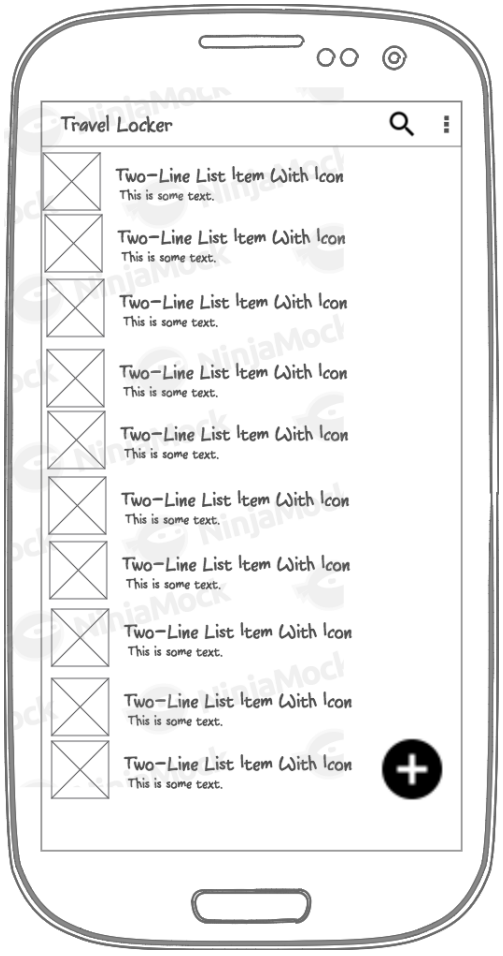
Features

- Saves information against the location
- Takes pictures
- Search and index locations in app

User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1 (Location List View)



The landing page of the app. Here all the locations stored in the app is displayed in a listview. The user can add new locations or search for location based on tags or location name.

Screen 2 (Location Detail View)

On click the list item the user is taken to the location detail view. Here the users can view the details of the location stored in the app. The details include the location , pictures, description and name. Upon clicking the navigate button, the user can navigate to the location via google maps.

Screen 3 (Location Add View)

This screen can be spawned from the location list view or the detail view. Once the user clicks the action button the user is taken here. The user can add/edit the following details here name,location, description and photos.

Key Considerations

How will your app handle data persistence?

All the user data is stored within the app in the Realm database with the help of the Content Provider in non exported form so that other apps can't use it.

Describe any edge or corner cases in the UX.

For example, how does the user return to a Now Playing screen in a media player if they hit the back button?

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

Realm Java enables you to efficiently write your app's model layer in a safe, persisted and fast way. I'll be using this as the database component for the app

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

I'll be using google maps api to get the device location and also fire an intent to the google map to navigate the user from current location he is in.

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

- Setting up the architecture components
- Designing models/objects of the system
- Configuring required services (Realm , google maps)

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:

- Build UI for LocationListView
- Build UI for LocationDetailView
- Build UI for LocationAddView

Task 3: Building the backend for the App

Creating the database required to store the location and associated data within the app