GitHub Username: ruiguo11

POI finder

Description

Looking for a good restaurant, an interesting place, or a nice clean hotel, POI finder will help you sort it out, and also tell you how to get there. And you also can save an interesting place to your own favourites list, so you can get the information without internet connection. And share interesting place with your friends.

Intended User

The intended user are the travellers, and people want to explore new places.

Features

- The app will all allow the user to select several places which include attractions, restaurants, hotels
- Allow the user to save their favourite places, and can read the details of the place offline, or share with other via email.
- Get directions to how to go to a place of interesting.
- This app will show point of interest nearby or search in a city.

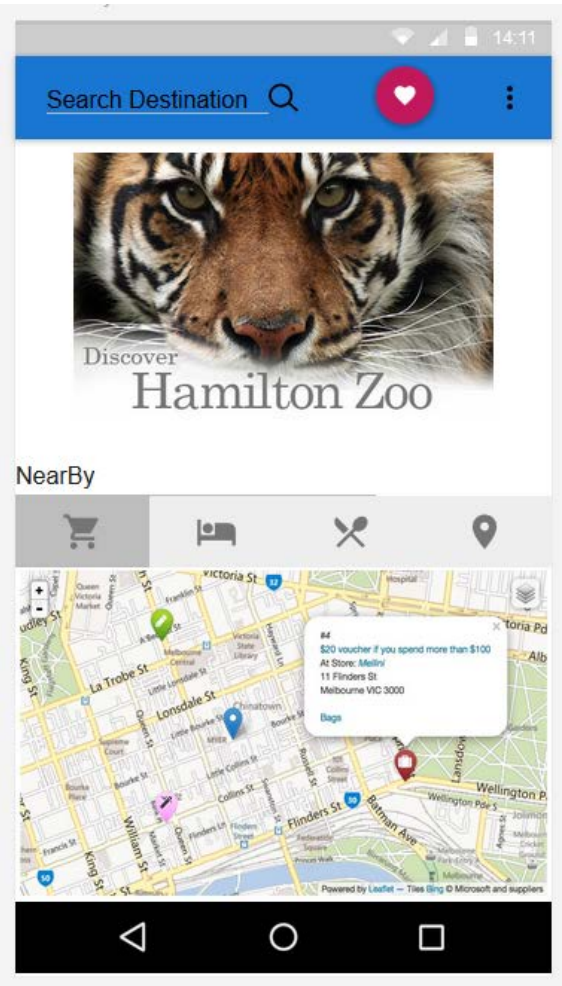
List the main features:

- Saves information
- Share information
- Route planning
- Searching information

User Interface Mocks

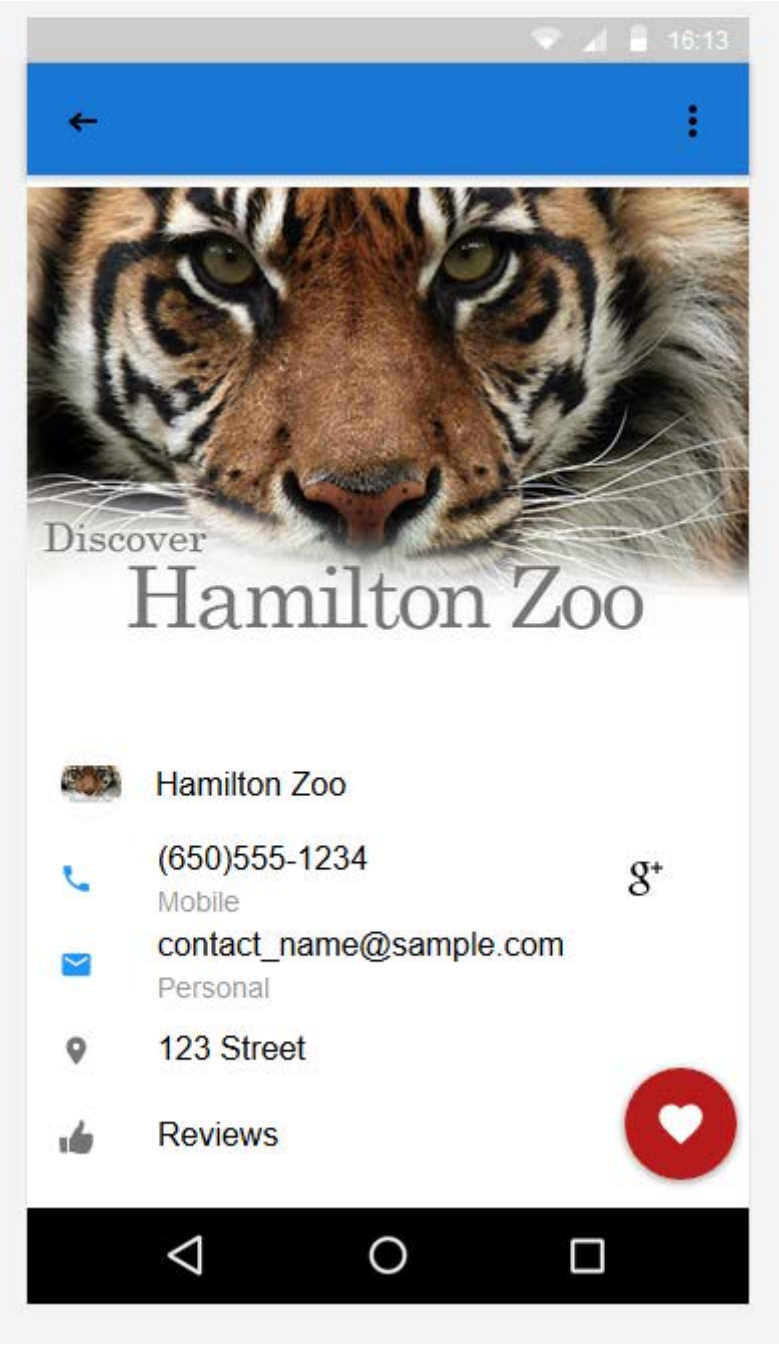
Screen 1

This is the main activity screen. When the app launch, the map will display the nearby shopping, attractions, restaurants, hotels. The user can click the mark on the map to get details of the selected point of interested. (see screen 2)



Screen 2

This is the details screen. When the user clicks the address or the icon by the address, it will display the location and the direction to go there. ()
The FAB will allow the user to add the place to his favourites list, the user can view offline.
This screen will use CollapsingToolBar Layout, and NestedScollView



Screen 3

Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
Provide descriptive text for each screen

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

Key Considerations

How will your app handle data persistence?

Describe how your app with handle data. (For example, will you build a Content Provider or connect to an existing one?)
I will build a Content Provider to handle data.

Describe any 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.

Use Picasso to handle the loading and caching of images.

Describe how you will implement Google Play Services.

Using Google location & Content API
Google Map, Direction, Place API

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

Write out the steps you will take to setup and/or configure this project. See previous implementation guides for an example.

You may want to list the subtasks. For example:

- Configure libraries
- Enable selected google API

If it helps, imagine you are describing these tasks to a friend who wants to follow along and build this app with you.

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:

- Build UI for MainActivity
- Build UI for DetailsActivity

Task 3: Your Next Task

- Implement Google Play Services
- Handel Errors

Task 4: Your Next Task

Describe the next task. List the subtasks. For example:

- Create layout
- Something else

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

Appendix:
Photo 1 :
http://www.pedfieldcountryhouse.co.nz/uploads/100922/images/213227/HamiltonZoo.jpg

---

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**"