How to Use this Template

- Create a new document, and copy and paste the text from this template into your new document [Select All → Copy → Paste into new document]
- Name your document file: "**Capstone_Stage1**"
- Replace the text in green

GitHub Username: Hassan-Gaad

Fosha

Description

Fosha is a vacation app , you can quickly peruse thousands of amazing places in Egypt , With the cheapest price , You can use to find well reviewed placed .

Intended User

This App for any one who intended to spend his/her vacation in a cheap and near place in Egypt

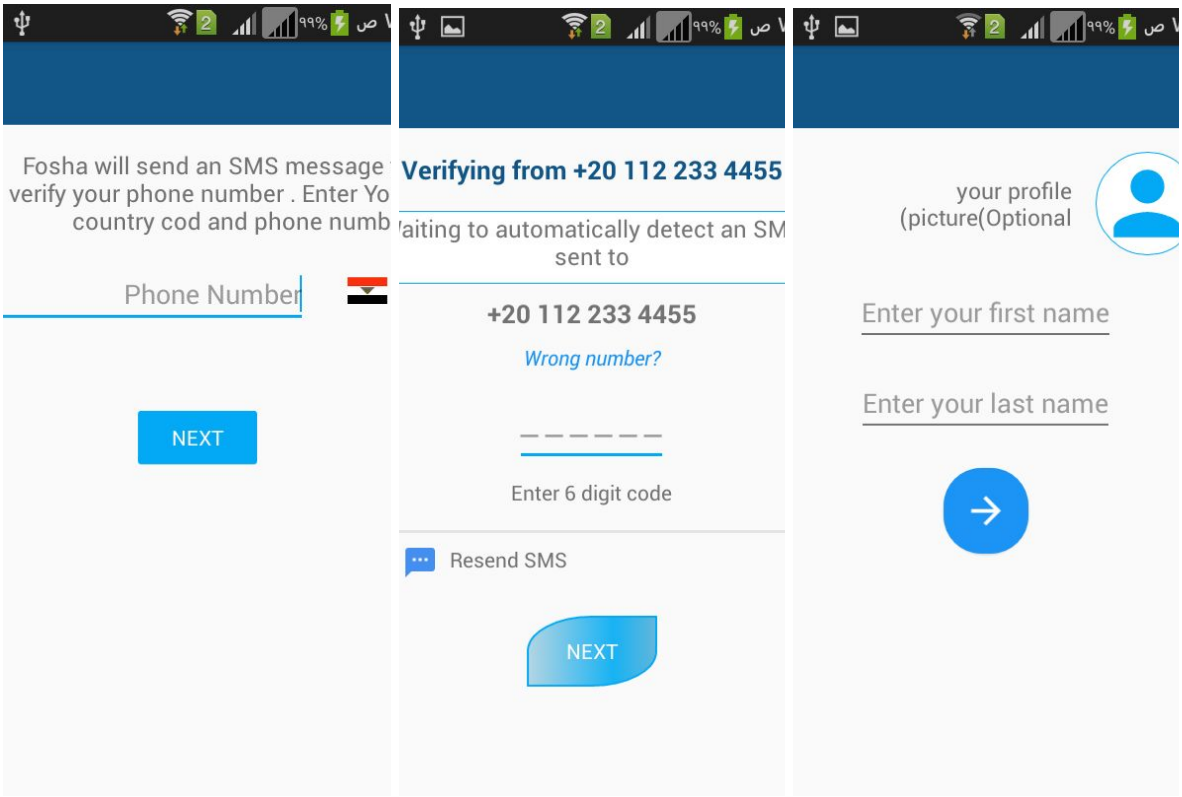
Features

- this app uses only Java programming language.
- To look at the overview of the available places
- To bookmark some place as favorite
- To filter the places by its closeness and price
- To look into more details of the place
- Add your feedback of a place
- Add new place

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.

This is a real screenshots from the project



Login screens for Phone authentication

MainActivity and DetailActivity that shows the place and the detail

This is the list of places that can be filtered according to price Under each item there is a like button ,,price and address of the place Each item uses grid to show images.
It also contains a floating Add button .

When click on item in previous screen it shows this screen with the full image's album of the place
At the bottom contains : images , descriptions and comments

Widget screen that show top place

Key Considerations

How will your app handle data persistence?

It will be using firebase cloud database for
persistence It will have Shared preferences to store
all setting data

Describe any edge or corner cases in the UX.

If user hit list item images , it will get bigger as app move to detail screen .
After user hit favorite button it will show a toast message that this place added to favorites

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

Firebase for cloud database .

- Cloud storage for storing images
- Realtime database
- Authentication

`com.google.firebase:firebase-auth`	`17.1.0`
`com.google.firebase:firebase-storage:`	`19.0.0`
Country Code Picker Library ***com.hbb20:ccp***	`2.3.1`
`com.github.bumptech.glide:glide`	`4.9.0`
`com.firebaseui:firebase-ui-database`	`4.3.2`
`com.google.android.material:material`	`1.0.0`

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

Firebase for cloud database .

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

Identify libraries and Configure build.gradle on the required dependencies

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity that holds Places Item
- Build UI for detail activity and construct its toolbar
- Build UI for Add new Place activity

Task 3: Implement recyclerView

Create RecyclerView that holds Place item and upload files to cloud storage

Task 4: Implement Data Persistence with Firebase and Test Scripts

- Declare the classes
- Declare data adapters

Task 5:Implement add button for adding new places

Create add button

Task 6:implement JobDispacter that sends images to API, app updates data in its cache at regular intervals

Task 7 : keeps all strings in a strings.xml file and enables RTL layout switching on all layouts.

App includes support for accessibility. That includes content descriptions, navigation using a D-pad.

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**"