

**GitHub Username:** ktsiounis

# Near Me

## Description

This app is created for travelers who want to find restaurants, bars and other places near to their location. It is a common problem for every traveler who's looking for interesting places when he's in a new place. This app keeps a clean, minimal and easy interface so as to let the user search for places quickly.

## Intended User

This app is mostly for travelers but every person who's interested to explore new places near to him, he can use it too!

## Features

The main features of this app are:

- Uses user's location
- Search for places around a location
- Search for places based on category
- Shows the places on map
- Saves users favorite places

# User Interface Mocks

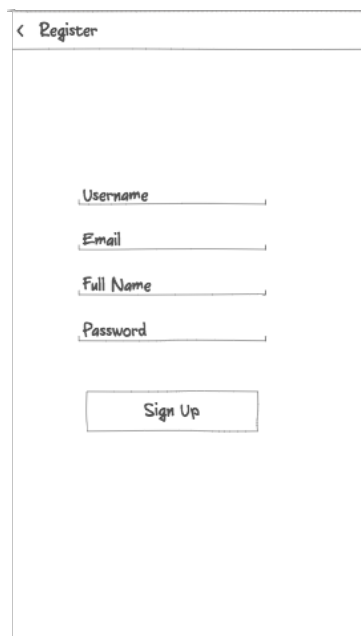
## Screen 1



A mockup of a login screen. At the top, the word "Login" is displayed in a header bar. Below this, there are two input fields: "Username" and "Password". Under the "Password" field is a "Log In" button. At the bottom, there is a link that says "Don't you have an account? [Create a new one!](#)".

First, the user needs to log in.

## Screen 2



A mockup of a register screen. At the top, there is a header bar with a back arrow and the word "Register". Below this, there are four input fields: "Username", "Email", "Full Name", and "Password". Under the "Password" field is a "Sign Up" button.

If the user doesn't have an account, he can create a new one.

## Screen 3

A hand-drawn wireframe for a mobile application screen titled "Search". At the top, there is a header bar with the word "Search" on the left and three dots on the right. Below the header, the main content area contains the text "What is the location you are interested to?" followed by a horizontal input field and a location pin icon. Below this, it says "You can also explore places near to you by categories!". Underneath, there are six rectangular buttons arranged in a 3x2 grid, labeled "Restaurants", "Bars", "Museums", "Gas Stations", "Coffee shops", and "Hotels". At the bottom of the screen, there is a tab bar with two tabs: "Search" (which is currently selected and has a thick underline) and "Favorites".

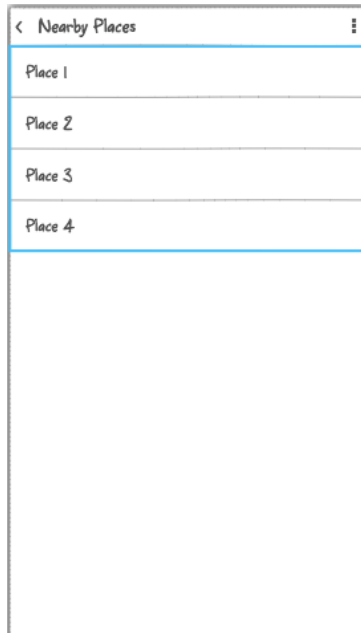
In the main activity, the user can search for places by giving a specific location or by choosing a category. There are also two tabs at the bottom. The first is for search and the second is for the user's favorite places.

## Screen 4

A hand-drawn wireframe for a mobile application screen titled "Favorites". At the top, there is a header bar with the word "Favorites" on the left and three dots on the right. Below the header, the main content area contains a list of four items, each labeled "Place 1", "Place 2", "Place 3", and "Place 4" respectively. Below the list, there is a large empty rectangular box. At the bottom of the screen, there is a tab bar with two tabs: "Search" and "Favorites" (which is currently selected and has a thick underline).

Here is the second tab with the favorite places.

## Screen 5



These are the results from search.

## Screen 6



This is the details activity where the user can see the details for a specific place.



This is the app's widget. It contains user's favorite places.

## Key Considerations

### How will your app handle data persistence?

The data will be handled using Firebase Realtime Database. Every user, has to be logged in so as to use the app and to have his own saved favorite places. The login/register will be implemented using Firebase Authentication.

### Describe any edge or corner cases in the UX.

When the user opens the app, he needs to sign in with his personal account or to create a new one. When he is logged in, he has the ability to choose a category and find places near to him or to search with a specific location. Then, a list of places open and he can click on anyone to see more details about the place.

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

The libraries that I'll use are:

- Picasso (v. 2.5.2) to load and cache images
- Butterknife (v. 8.8.1) for binding views easily
- Retrofit (v. 2.1.0) for API calls
- RecyclerView (v. 27.1.1)
- CardView (v. 27.1.1) for the cards in recycler view

- ConstraintLayout (v. 1.1.2) to create a better layout
- Espresso (v. 3.0.2) for UI tests
- AdMob (v. 15.0.1) for displaying ads

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

I will use Firebase Realtime Database to save data and users, Firebase Authentication to authenticate users, Maps, Places and Location to show places and take current location. I'll also use AdMob to display some ads in free version.

## Next Steps: Required Tasks

### Task 1: Project Setup

- Implement project libraries
- Setup gradle and project dependencies
- Get API key for Maps
- Create Activities and Fragments

### Task 2: Implement UI for Each Activity and Fragment

- Build UI for LoginActivity
- Build UI for RegisterActivity
- Build UI for MainActivity
- Build UI for PlaceDeatilsActivity
- Build UI for PlacesListActivity

### Task 3: Implement Firebase Authentication

- Implement login
- Implement user register

### Task 4: Implement Firebase Realtime Database

- Add favorites places in database

- Remove favorite places
- Retrieve and show them in a view

### **Task 5: Implement Admob**

- Create ad layout in activities

### **Task 6: Implement Maps**

- Add maps in activity
- Implement search with location and category
- Show pins for places on map

### **Task 7: Create Free and Paid version of app**

- Add dependencies in gradle for free and paid version

### **Task 8: Implement AppWidget**

Create app widget to show places.

### **Task 9: Espresso Tests**

Write espresso UI tests.

### **Extra information:**

- The app will be written solely in the Java Programming Language.
- The app will utilize stable release version of all libraries, Gradle and Android Studio.
- All the libraries being used, together with Gradle and Android Studio will belong to stable versions.
- The app will include support for accessibility. That includes content descriptions and navigation using D-pad.
- There won't be hardcoded string in the app. The app will keep all strings in strings.xml file and the other resources in their related files.
- AsyncTask will be used to implement log in and register user procedure.

- The Gradle's version is 4.4 and Android Studio's 3.1.3 .
- Place's, Map's, Nearby's version is 15.0.1 .
- Firebase's Core and Database version is 16.0.1 and 16.0.2 for Auth.