Design Document

List of technologies

- **Google Play Services**
  The client library contains the interfaces to the individual Google services and allows you to obtain authorization from users to gain access to these services with their credentials. It also contains APIs that allow you to resolve any issues at runtime, such as a missing, disabled, or out-of-date Google Play service
  With Google Play services, the app can take advantage of the latest, Google-powered features such as Maps, Google+, and more, with automatic platform updates distributed as an APK through the Google Play store.
- **Volley networking library**
  Android volley is a networking library was introduced to make networking calls much easier, faster without writing tons of code. By default all the volley network calls works asynchronously, so we don't have to worry about using asynctask anymore.
- **GSON**
  Gson is a Java library that can be used to convert Java Objects into their JSON representation. It can also be used to convert a JSON string to an equivalent Java object. Gson is an open-source project.
- **Google SDK tools**
  The Android SDK (software development kit) is a set of development tools used to develop applications for Android platform. The Android SDK includes the following: Required libraries. Debugger. An emulator.
- **Android Studio**
  Android Studio is the official integrated development environment (IDE) for Android platform development. It was announced on May 16, 2013 at the Google I/O conference. Android Studio is freely available under the Apache License 2.0
- **Support Design Library**
  The Android Support Library offers a number of features that are not built into the framework. These libraries offer backward-compatible versions of new features, provide useful UI elements that are not included in the framework, and provide a range of utilities that apps can draw on.
- **AppCompat**
  When new versions of android is published, Google will have to support the older versions of android. So AppCompat is a set of support libraries which can be used to make the apps developed with newer versions to work with older versions.
- **CardView**
  The new support library in Android L introduced two new UI widgets: RecyclerView and CardView. The RecyclerView is a more advanced and more flexible version of the ListView. This new component is a big step because the ListView is one of the most used UI widgets.

- **Google Materials Design**
  Material Design (codenamed Quantum Paper) is a design language developed in 2014 by Google. Expanding upon the "card" motifs that debuted in Google Now, Material Design makes more liberal use of grid-based layouts, responsive animations and transitions, padding, and depth effects such as lighting and shadows.
- **Mockito**
  Mockito is a popular mock framework which can be used in conjunction with JUnit. Mockito allows you to create and configure mock objects. Using Mockito simplifies the development of tests for classes with external dependencies significantly.
- **JUnit**
  JUnit is a unit testing framework for the Java programming language. JUnit has been important in the development of test-driven development, and is one of a family of unit testing frameworks which is collectively known as xUnit that originated with SUnit.
- **Robolectric**
  Robolectric is a framework that allows you to write unit tests and run them on a desktop JVM while still using Android API. Robolectric mocks part of the Android framework contained in the android.jar file.
- **Espresso**
  The Espresso test framework. Espresso is a testing framework for Android to make it easy to write reliable user interface tests. Google released the Espresso framework in Oct. 2013. Since its 2.0 release Espresso is part of the Android Support Repository.
- **Ubuntu**
  Ubuntu is a Debian-based Linux operating system and distribution for personal computers, smartphones and network servers. It uses Unity as its default user interface.

Architecture and design implementations

Networking

I decided to use Volley for networking because of the JSON API passing on and off.
Volley is one of the most used libraries for networking, it is extremely performative and productive to implement.

Google Maps(Google Play Services)

I decided to use Google Maps as the most used and official operating system tool. I've used it on several projects and always met the requirements.

Google Materials Design

I've tried implementing the application screens according to google materials design guidelines.

Tests

In this project I implemented unit tests of the controllers and automated tests of the login and signin screen.

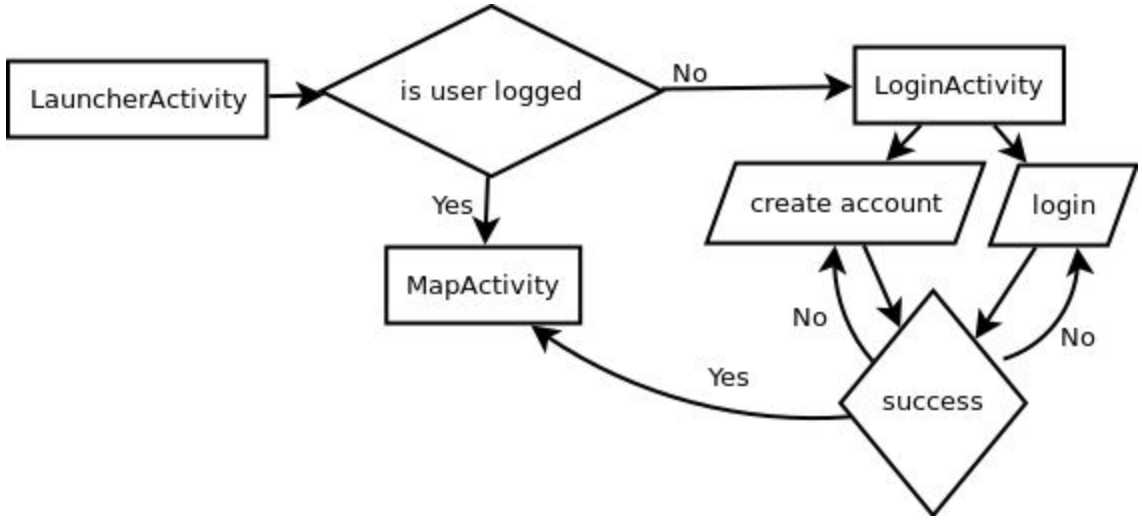
For automated testing I used the Espresso framework. This framework makes it much easier to implement the automated test because it provides more methods of accessing the components of the screen.

For the unit tests I used Robolectric because it is more robust. Only Mockito is not enough because it does not provide mock for native classes like JSON and other important ones.

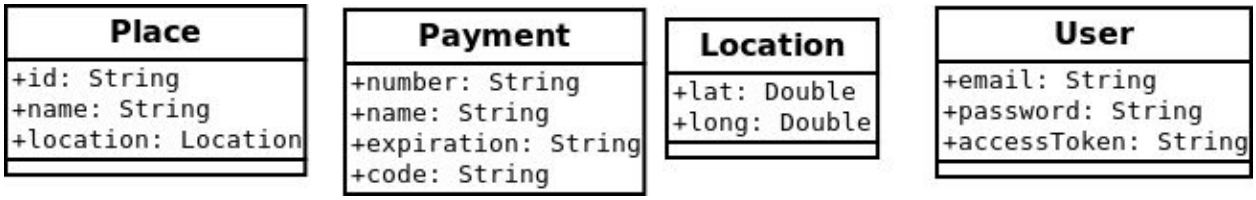
GSON parser

I chose to use GSON because it is very fast to run parse and extremely productive to work with.

Activity Diagram



Class Diagram


Place
+id: String
+name: String
+location: Location
Payment
+number: String
+name: String
+expiration: String
+code: String
Location
+lat: Double
+long: Double
User
+email: String
+password: String
+accessToken: String