****

**CIE203**

**Software Engineering**

Project Name

Software Implementation Document

2M&KH

Khaled Hefnawy

12 / 2016

Contents

[Team 3](#_Toc468897585)

[I. Document Purpose and Audience 3](#_Toc468897586)

[II. Program Structure 3](#_Toc468897587)

[III. Installation and Setup Guide 4](#_Toc468897588)

[IV. User Manual 4](#_Toc468897589)

[Team members roles 4](#_Toc468897590)

[Ownership Report 5](#_Toc468897591)

[Policy Regarding Plagiarism: 5](#_Toc468897592)

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 201300812 | Khaled hefnawy | [s-Khaled.mohamed@zewailcity.edu.eg](mailto:s-Khaled.mohamed@zewailcity.edu.eg) | 01066394056 |
| 201301691 | Mohamed wasfy | [s-mohamed.safieddin@zewailcity.edu.eg](mailto:s-mohamed.safieddin@zewailcity.edu.eg) | 01110594119 |
| 1201305056 | Mohamed alaa el-din Fargaly | [s-mohamed.fargaly@zewailcity.edu.eg](mailto:s-mohamed.fargaly@zewailcity.edu.eg) | 01002680520 |
| 201306057 | Hazem Hamdy | S-hazem.alromisy@zewailcity.edu.eg | 01288449661 |

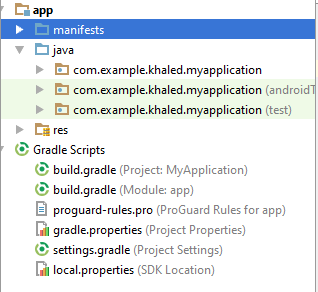
# I. Document Purpose and Audience

This Document is for the users of the app whether an admin or a normal user. It is expected to find the full instruction how to use the app

Also this Document show the general structure of the code and what each part does and how they are connected

# II. Program Structure

**General Structure of our program**



**Package installed**

The packages that we used are installed on the android studio. We installed two packages that are Google maps package and Parse Database Package both of them are embedded in the code. We Cannot separated our activities to packages as most of our activities use It.

**Composition of our program**

Our program is an android program so it is composed of Activities. Each Activity is Composed of Java Activity file and two files for the Xml which represents the layout of the activity. One activity represents the code of a page in the program and its layout. Our code is composed of 7 java code files and 12 layout files as there are classes that are used without layout as they are only classes for making object they are PlaceAdapter and Place those two files have no layouts. They were used to make the row file which meant to make a good shape list view which shows the pictures of the users and their names. However it wasn’t completed because of time

**Small note:** for each java code we have to xml layout files just for simplifying the design

Our program has a manifest file that we declare all the activities in it and ask for the required permissions that will be shown for the user at installation.

We have the gradle file this file has the code for installing the packages we add the parse and google package using this file and then we compile it to have the classes when we code the program

**Explanation of each package**

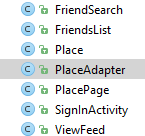
we don’t need to divide our code to other packages than the google api package and parse database package as our java code files has self-explanatory names and don’t need to be in one package

google api package is the one responsible for giving us information about nearest places and we call an intent that open google api. Then we get the place and save the rating and the place on our database.

Parse package provides us with simple connection with the database. We just need to put the app id and the link of the database to make the connection.it also provides us with several types of queries that simplify the way we retrieve data from the database. It provides us with functions of different functions of registeration such as sign in and register that could check if the username is distinct on the database. We don’t need to write this code ourselves

We used the Gui package in our layouts of the pages

**Java files**



Those are the java codes that control the flow of the program. FriendSearch is the code that is responsible for the retrieval of users data on database and show them when a user search for friend. Its layouts files are activity\_friends\_search and content\_friends\_search.

FriendsList we used this file to query the actual friends of a specific user. We showed them in the friendsList layout file. We then use this list to show their feed in the viewFeed Activity Page. Its layouts files are activity\_friends\_list and content\_friends\_list.

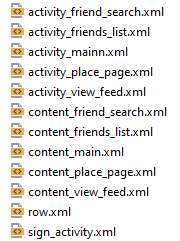
Place and PlaceAdapter were not used in the present Code. They were intended to make the nice view of users with pictures. They were examples of such code. However we couldn’t complete this part because of time

PlacePage activity is the one responsible for Searching a place to find its rating. . Its layouts files are activity\_Place\_Page and content\_ Place\_Page.

SignInActivity is the java code that is responsible for signin, Register and forget password. It calls the functions of sending emails on the database. Its layout file is sign\_activity.

ViewFeed is the code for showing the feed of otherUsers and its layout files are activity\_view\_feed and content\_view\_feed

**Layout files**

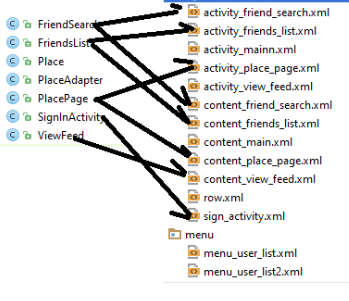


**Further folders needed for Our program**

We have the **drawable folder** that stores the required images needed in the program such as the ones that appears on the buttons.

We have the menu folder needed to create the layout and of the menu that appear on the action bar. Actionbar is the top of page. That has the back button, logout and showing the feed

Values folder stores the required strings, Colors and styles of the pages needed for our app. This folder gives high opportunity to make generic apps that could be translated to any language. The best practice is to define every string with a variable name and then write the name in the string file. If then in forward development it is needed to be translated to another language we just change the strings file with the same name of variable but in another language.



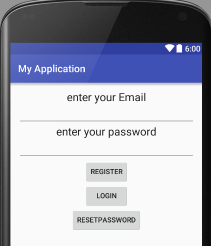
# III. Installation and Setup Guide

For the end users they just need to download the apk of the program and accept the required permission of the program. The database is initialized automatically when the app is installed. The code of the program has the required initialization of the database.

As it is an android program it doesn’t require complex installation. It is just an accept click on the apk to install it

# IV. User Manual

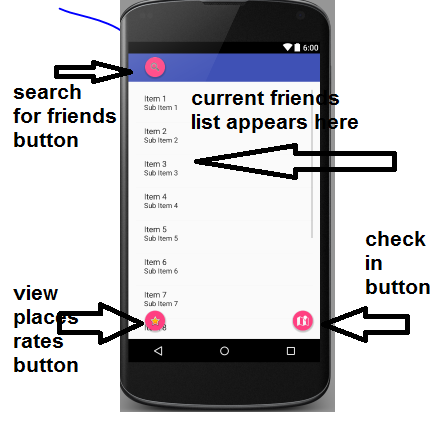
We have a simple user interface for login, register and forget password. When a user register he should verify your email by opening the email and press on the verification link. When a user want to reset your password you enter the email address and press forget password. A link is then provided that enables the user to reset his password



When the user logins, he is forward to the viewFeed page. He could then go to the friendsList and make checkin, Search for a friend or find the rating of specific place.



This is the friends list page. Most of the function that the user could do is here. We could then later change these function to be in the viewFeed page



There is one kind of user in our app that is the normal user. Premium user wasn’t implemented at this stage

Almost we have no crash errors. We were so keen to solve all the crash error that could hinder the user experience. Also we show messages for the user when he forget to enter the password. We show message of what the user should do. We show message if there is incorrect combination of username and password.

We also show a message for short passwords and invalid mails that doesn’t contain @ character

The user is required to open the internet and location in the mobile to be able to retrieve data and store his place

# Team Members Roles

* **In this part you should mention the role of each member in your team in this phase**

|  |  |
| --- | --- |
| **Team member** | **Role** |
| Khaled Hefnawy | *Code integration, Organizer of the work and gitlab* |
| Wasfy | *GUI of the app* |
| Mohamed alaa | *Algorithms implementation and api's research* |
| Hazem hamdy | *database implemention and creation* |