

Project Title: Social life tracking
Team Name : Team Charlie

Active Member:

1. Khondakar Readul Islam
2. Emdadul Sadik
3. Md. Wazid Ali



Abstract

1. Project overview
2. System characteristics (given criteria from department, our own added characteristics)
2. Architecture of the apps (event based, diagram, control flow)
3. Design consideration
4. Software development tools, gradle, ngcordova, Ionic native, nodejs, angular, mongodb, mongoose, Express generator, CORS,
5. Component description,
6. User interfaces....
7. Dependencies (package.json)... Camera plugin, google api, geolocation, File transfer , JWT
8. Data (file structure, table schema, data mapping, security)... collection, user, location, activity, image,
9. Data flow
10. User Instruction

Introduction. The project “Social life tracking” is part of the “Project praktikum”, which has been implemented by a group of student from “Distributed Software System” at TU Darmstadt. The apps in gen is mobile based reporting system of individual’s where abouts and his connected friends as well. It also capable of reporting individual user’s daily activities. In addition to that the app has a gaming features which utilize the user’s activity to challenge other user to do the same activities to win a prize. The app’s provides an easy way to interact with friends and families while boosting enthusiasm for physical activities.

Intended audience. This document is intended for the project supervisor, customer, user, team member, enthusiastic developer and the other team of tk project praktikum.

System characteristics. Basing on the objectives of the project “Team Charlie” decided the following characteristics of the apps.

- a. User management system
- b. User tracking system
- c. Friend management system
- d. User activities tracking
- e. Statistics of user activities

f. Gaming

a. User management. Users of the app is able to manage personal data, user account and password. He is able to edit/delete any information in his profile. He can reset password and reset account in the event of forgetting password. He can deactivate his account temporarily.

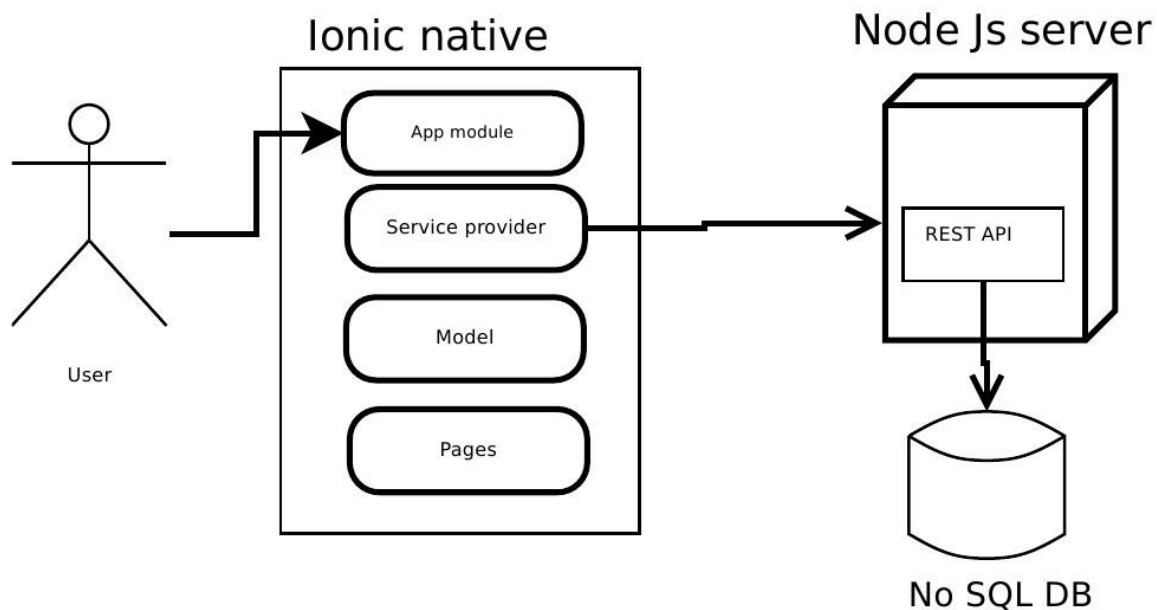
b. User tracking system. User can track his own personal and social activities. He can see the the history of his activities in the time line.

c. Friend management system. Users are capable of selecting their friends. They can see the current location of their friend. Alternatively, users can locate their friend location on google map. They can see the activities of their friend and if any of their friend has thrown any challenge of any activity.

d. User activities tracking. Users can track their physical activities and keep a record of those activities. They can see the activity records in various aggregations. They can also challenged their friends on different activities.

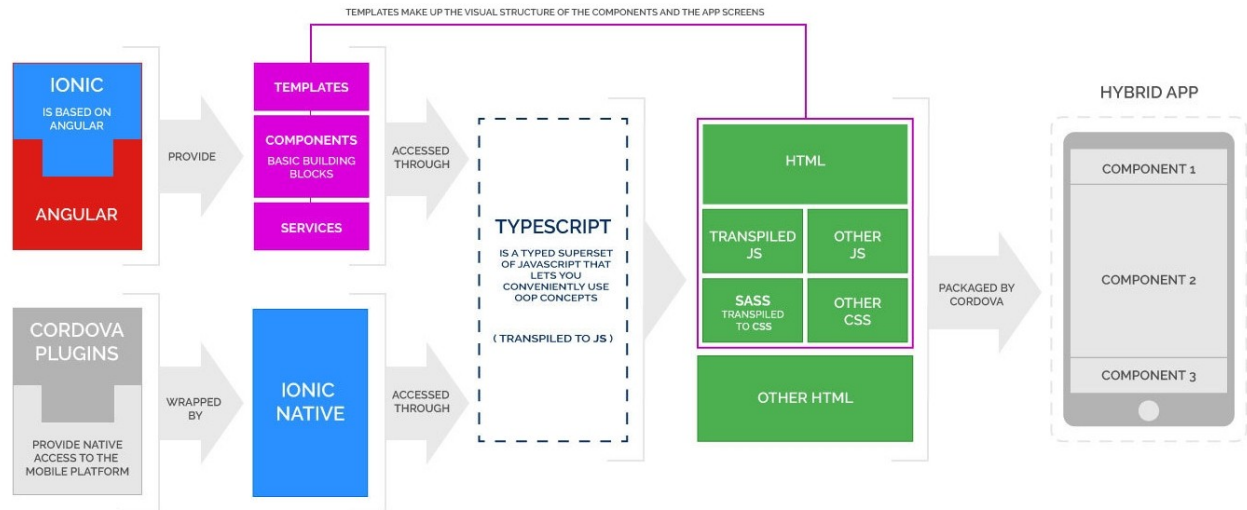
e. Statistics of user activities. Users can see the statistics of their activities.

f. gaming.



Architecture of the app. The app is built in distributed paradigm model. Event based programming is used for communication between server and client. We have used node.js server for back end, angular 4 based ionic3 framework for native android app. For data storage we used mongodb. Basing on the characteristics of the app we divided the app into various modules like User management module, Friend management module, Activities management modules, Gamification modules etc. The basic architecture of the app is following:

Ionic framework is wrapped in cordova for cross platform native mobile functionality. Which intern use typescript for modules and other rest services. We have developed rest API in node js for serving user request.



Design consideration. The most crucial aspect of design consideration was concurrency. As node provide asynchronous functionality so we used node server for asynchronous service which will allow the app to scale in increased number of user and user request. This will provide user with smooth user experience. Ionic provides cross platform build capability of native mobile app, which facilitate same user experience for all user regardless of their individual mobile platform. Ionic provides a set of components which are built in with the framework, provides seamless functionality over all mobile platform. The use of angular 5 helps building powerful and robust application.

Software development tools. A number of software development tools has been used for this project. Some of the worth mentioning are following:

- gradle. We used Gradle to build the app for native device. Gradle It's a build automation tool. In our case gradle is building android APK file that could be installed in Android devices.
- ngCordova
- Ionic native. Ionic native Is an Angular and Apache Cordova based multi platform development framework. We have used Ionic 3 which came along with Angular 5.
- mongodb. It's document based database. Highly scalable because of it's nature of loosely typed schema. Data comes prejoined or embedded. It's probably the best NoSQL database in the market.

- e. **Mongoose** Mongoose is an ODM (Object Data Mapper) that compensates for monodb's lack of tight schema definition. With mongoose a document to model/object is possible and driver level validation is achievable. Also data manipulation becomes handy.
- f. **Express**. Express is a nodejs based web development MVC framework. Mostly used to develop highly decoupled REST backend.
- g. **Nodejs**. Node js is javascript runtime environment. It's an eventdriven architecture with non-blocking I/O which enables greater scalability and higher throughput. Service based design is one of the best approaches in Nodejs development. So is it in our case.

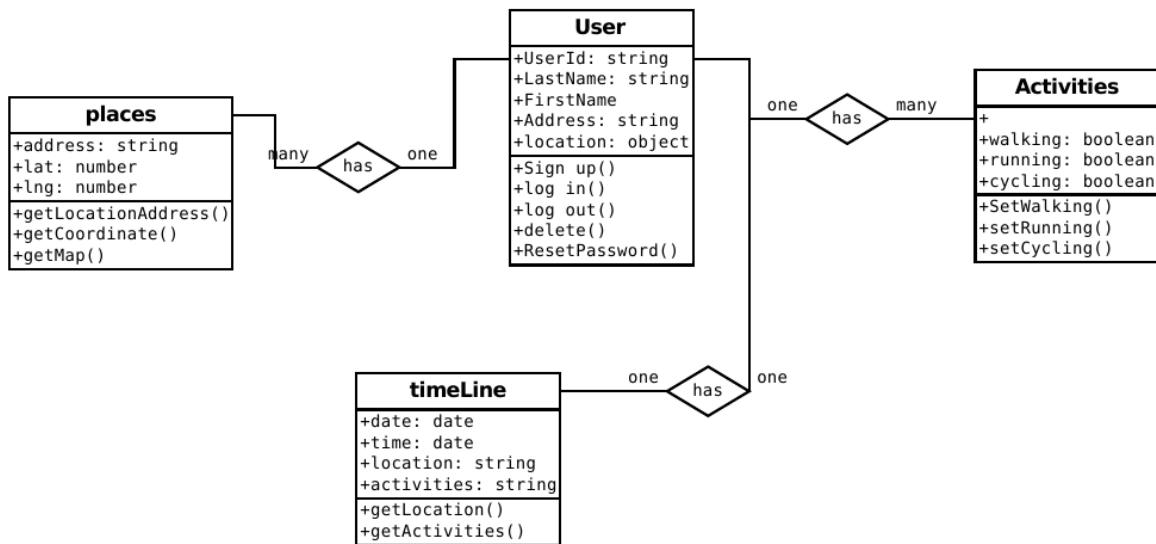
Entities. This section of the document described the entities used in this application and the relation and constraint among them.

a. User. The most crucial entity of the application. All the functionality of the application revolve around this entity. The user entity has the capability of creating a new user, delete user, edit user information, reset user password. This entity holds the attributes of lastName, firstName, userAddress, userLocation. Sign up to the application, log in to the system, logout, resetpassword, deleteUser, etc. functionality are provided by this entity. This entity has 'has a' relationship with place, timeline and activities entity.

b. place. The place entity got 'many to one' relationship with user entity. It has the functionality of getting address through google api by latitude and longitude and vice versa. This entity also has the functionality of calling google map and display location on map.

c. Activities. Another important entity of the system provides users activity records. It has a 'many to one' relationship with user entity. It provides the functionality of setting user activities like walking, running or cycling. It also has the same attributes as running, walking, cycling.

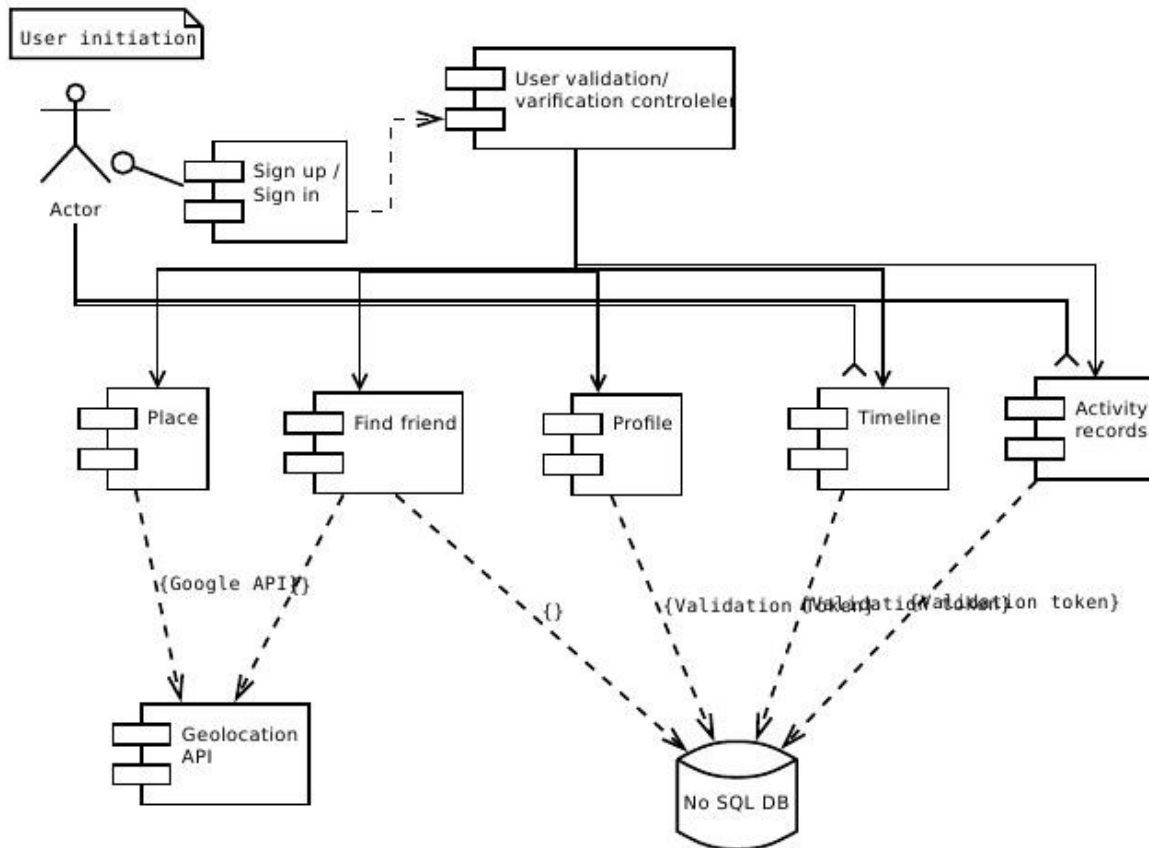
d. Timeline. The timeline entity has a 'one to many relationship' with user entity. This entity is responsible for showing the user location and activities in a timely manner. It has the attributes of date, time, location and activities. It provides its functionality through the methods getLocation and getActivities.



Component description. The worth mentioning compnents of the app are

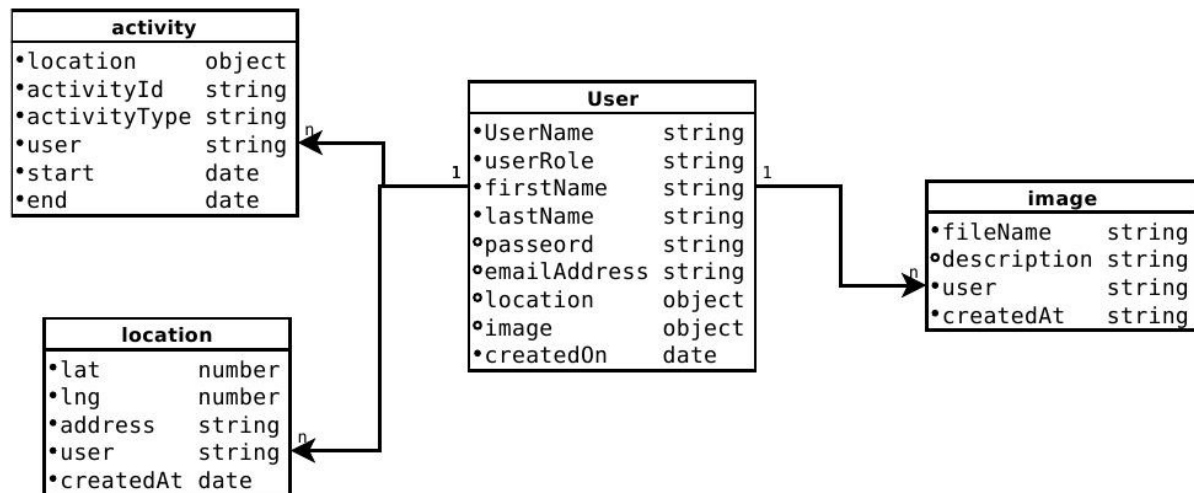
- signup components. Sign up component is primarily responsible for user login and singing up of new users. In addition to that this components also provides user validation and verification. It has its own set of rules for user verification which provides an extra layer of security of the apps.
- profile components. This component is used for viewing user profile. Users are able to edit their profile information through this component. This component is also used to show the friends profile of a user.
- Find friend components. Through find friends component users can locate friends whose current locations are around the users. This component use google geolocation API to locate own position and friends position. This component then shows these locations on Google map.
- place components. Through this component user can save picture and position of a particular place as his favorite place. User can locate himself on map anytime and take a picture through mobile camera and upload the picture with location information as his favorite place.
- Activity records components. Activity records help user to keep records of his/her workout. When ever user do some activities he needs to activate the activity through this component and the component automatically keeps tracks of that activity i.e. distance, time etc. Later user is able to see the aggregated records of his activities.
- Social component. Need to write

g. Time line component . Through time line component user can see the history of his location and activities in a timely manner. All the location data and activity history is shown in terms of time line.



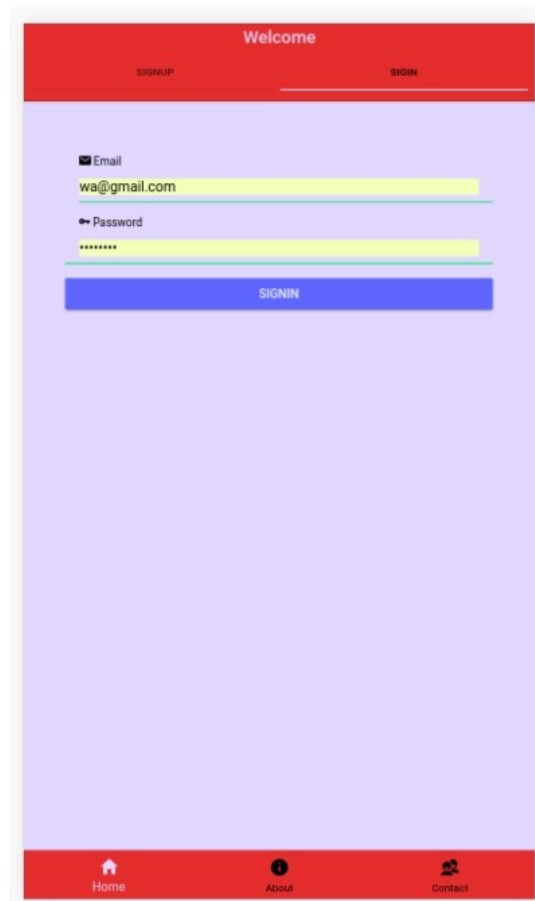
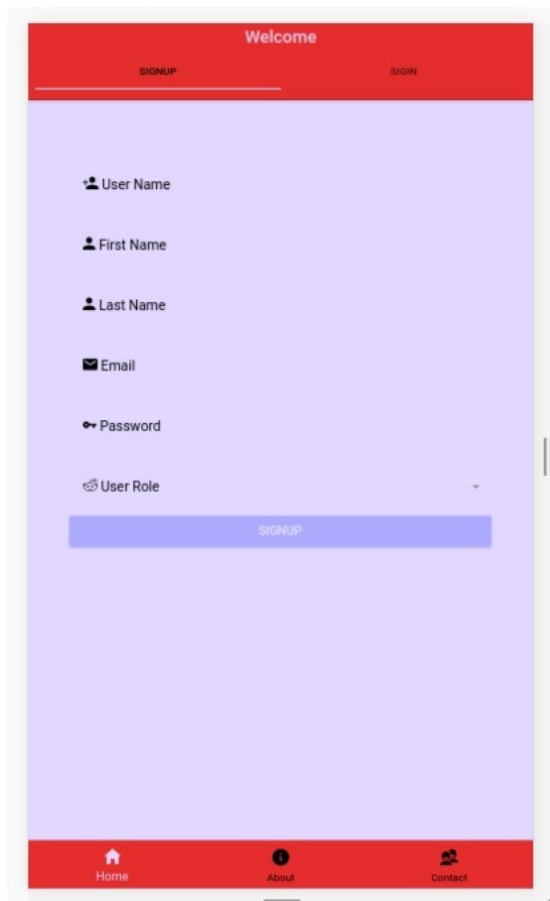
Dependencies. This project has number of dependencies which includes Google's geo lactation api, mobile camemra api , mobile sensor, JWD etc.

Data. No SQL database i.e. mongodb was fond most suitable for this app. We have used mongoose for data intigrity and data manipulation. We have used number of schema for our data storage i.e. user schema, activity schema, image schema, location schema. The details of collections and data association are shown in the schema diagram.



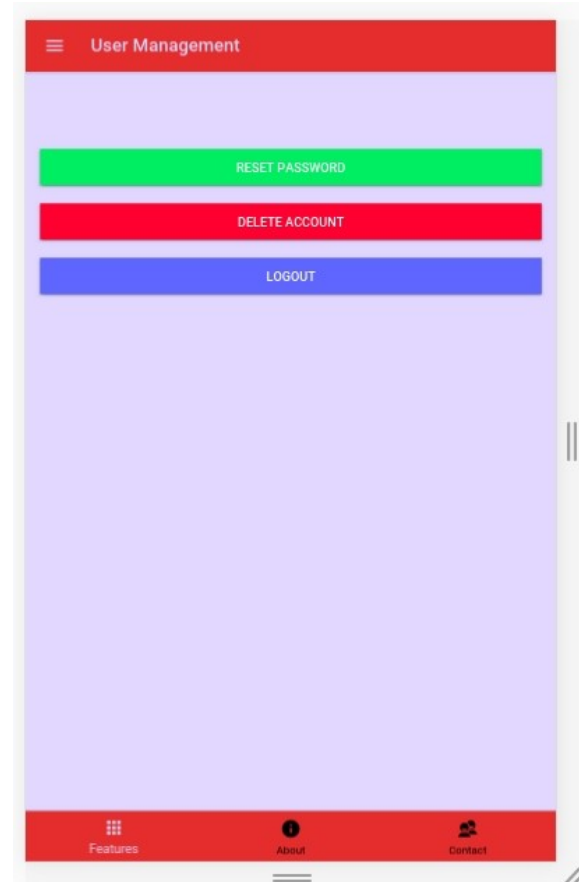
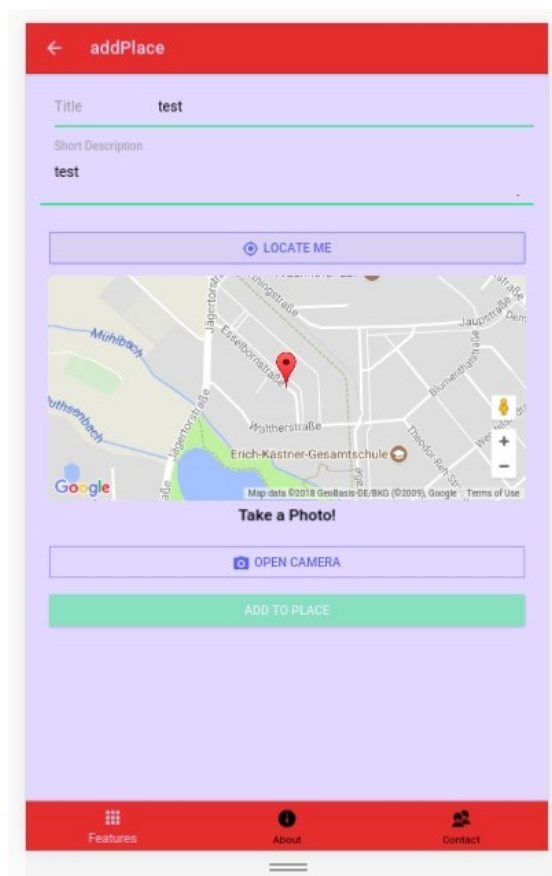
User manual. This section of the document describe the how to use the app.

a. User sign up. Before using any feature of the app user has to create an account by signing up to the system. User has to provide some basic user information of the user in order to sign up. During sign up the app will validate the user name and password. After successful validation the provided information user will be able to log in to the system and use the functionality of the system. To sign up to the system provide the user name, first name, last name. Email, password and role and then tap sign up button as shown on the picture. You will be registered as a new user of the system and get a success message.



b. Log in. User can log in to the system from log in page by providing the appropriate log in credential. To log in to the system go to the log in page enter your user name and password. Then tap sign in button. You will get a success message. After successful login user will be provided with the main tab page where he will be able to scroll through the favorite place, user management, time line, activity records, gamification pages. Apart from that user will also be able to navigate through the features menu to add favorite place picture to the app, find friend page, set activity page.

c. Add favorite place. In this page user can add a place as his favorite place and a picture of the place in his profile. To do that tap enter button at the favorite place tab. This will navigate you to the favorite place button. Tap the + button at the top right corner. Give a name of the place in title field, add a short description of the place. tap locate me button, the app will locate your current position on the map. After confirming the location tap the open camera button which will open the device camera. Take picture of the place and tap 'Add to place' button. This will save the picture, location and provided description of the place.



d. User management. To navigate to user management page tap on enter button of user management tab of features page. You will be provided with the logout, reset password and delete account. You can log out, reset password or completely delete the account from these button.

c. Time line. User can navigate to the time line page through tapping the enter button on time line tab on features page. On the time line page user can see his location history and activity history in a timely manner.

d. Activity records. User can see the history of his activity either daily, monthly or weekly basis from this page. Go to Activity records from features menu. Select the activity you want to see and aggregation interval. An aggregated history of the activities will be shown on the screen.

e. Find Friends. Users can locate nearby friends from this page. Go to find friend page from features menu. Tap to locate me button. After locating the users position tap find friend button. The app will show your nearby friends location on the map.

Find Friends



LOCATE ME

FIND FRIENDS

BACK