Project plan and second iteration report

Group 6:

1. Ragunandan Rao Malangully – 45
2. Sandeep Siddhi – 57
3. Raviteja Reddy Kandakatla – 33
4. Pradeep Chaitanya Tirumalsetty - 61

Submitted on 10/19/2015

Introduction

This document intends to provide an overall description of the project named “Travel Guide” in detail. The project schedule and the plan of action is also discussed. The proposal document submitted would give an insight about what the project is about. The main outcome of the second increment is the three modules of the application(weather, task and language convertor modules). As of current state we have not deviated from our initial proposal that we submitted earlier.

Project goal and objectives

Overall goal

The project goal is to put together a set of related and useful web services under one umbrella. The app intends to serve the needs of a person who is in search for some trivial information about a place that he in planning or intends to travel to. The significance of the application lies in the fact that it serves as a one stop shop to interact with various information sources. Additionally the app lets users to interact with social media. The objectives of the application are as follows.

Objectives

- To reduce the user time in searching for information across various sources.
- To develop a system architecture that can integrate the different features that the system proposes.
- To ensure that each iteration is tested thoroughly before heading to the next one.
- To develop a user friendly application.
- To provide user a tool to keep track of items that are required for the travel and the duration of the stay.
- Ensure that the user never misses important information in a touring country by providing the translate feature that translates the text into the user specified language.
- The user would never miss out on famous tourist locations as our app has a feature to suggest famous tourist attractions or hangout spots in a place.
- A tourist never feels away from home. The accommodation finder is a tool that helps the user in selecting his home away from home.
- Currency conversion is one of the major problems that tourists face when visiting other countries. The users of this app have no such worries as we take of currency conversions based on the latest foreign exchange rates.
- The user can prepare for the weather of the place he is visiting. The system predicts the latest weather information of the place of interest.

Project background and related work

The app is not a new invention or a brand new concept. Instead it intends to provide a mash up of the various web services that are present in full or in partial implementation in the below references.

- OpenWeather: This app offers information on the weather intricacies of several places around the world using cloud technology [1]. The application that we are developing uses simple REST API technology rather than the cloud to get information about a

specific place or region of interest. The UI would be significantly different from what the Weather Pro app has in place.

- TripAdvisor: This app offers reviews on hotels, restaurants and places of attraction at the mentioned travel destination [2]. In coherence with this our proposed app is only a suggestion service on the various places of accommodation. But the primary contrasting factor is that in the app being developed there is no ranking or reviews of the places suggested.
- Hostel world: This application gives the list of budget friendly hotels present in and around the place being visited [3]. This application only retrieves a list of hotels or accommodations that are light on the pocket. But in our app we provide a list of all the hotels and accommodations that are returned by the API service.
- XE Currency: This tool is ideal for getting the live currency information [4]. The tool mainly acts akin to a currency conversion chart. The "Travel guide" app on the other hand provides specific rates for the currencies selected by the user.

Significance

The major significance of the application lies in the fact that the all the useful services are provided under one system. This saves time wasted in surfing the web to get the required information from the various apps or websites. The user need not create a new profile or provide redundant data to the system. He can authenticate himself using his Google+ or Facebook account. The system ensures that the user is prepared and has planned his travel with all the required information in hand. The user also has the added advantage of connecting to his social network and sharing his thoughts about the travel and stay at the destination.

Proposed system

Requirements specifications

- The application will provide users facility to login with their Google plus account.
- User should have the flexibility to login with his Facebook account.
- The system will display the user his information along with his profile picture.
- The user shall be provided with six options or features that the user can interact with.
- Feature of currency conversion shall be provided.
- Latest weather predictions of the place of interest shall be displayed to the user.
- Given a place, the system must suggest and display the famous tourist attractions and other places of interests.
- There shall be a feature for the user to keep track of items and tasks that the user intends to buy or plans to do during his course of travel or stay.
- Accommodations options such as hotels and other places to stay shall be suggested to the user, based on user's choice of place.
- Text translation into the language chosen by the user is to be provided

Workflow analysis

The first and the most fundamental workflow of the application is that of the designing the UI for the system. Since our application is intended for mobile platform, Ionic Framework gives a better look. So we have used this latest framework in our application. Each page of the application is to be custom designed for suiting the needs of the service or feature being provided over that page.

Second, the information on selecting API and implementing the API features into the system. This deals with the selection of API from a list of multiple options. This process is to be repeated for each of the service being offered.

The third workflow deals with the creating a mongodb database to store and manipulate the tasks or items to be tracked for the user. This flow ends with the integration of the individual components of the system to end up with a stable application. The appropriate workflow analysis is provided by the sequence diagrams provided in the first increment report.

Technological and architectural requirements

On the technological facet of requirements the system needs to interact with the external REST APIs provided. We require the interaction with service of Google plus and Facebook for the login functionality to be implemented. Constant interaction with services provided through APIs should be present for the system to be available for usage. Most of the interaction would happen over the REST technology and JSON would be the primary medium of interaction. For the service of tracking tasks and items the system would use a local storage and have to interact with the native library features to display notifications on tracked items.

Java would be the primary language for implementation of the business logic. Ionic Framework, Angular JS would form the core framework for the UI layer and UI interaction with the REST services offered through APIs. We have incorporated the Ionic framework and bootstrap JS for multiple platform implementation and better UI experience of the user.

The architectural requirements would focus on the providing an architecture that can integrate the various services offered from external sources. This provides a challenge in coming up with a flexible architecture that has the freedom to integrate various sources of service providers to provide a complete experience to the user.

Import Existing Services/API

Primary services and their features

The following are the primary services that the system offers to the user. The APIs being provided here are initial intuition and may change during the project development phase to provide better service quality. The priority of the tasks are not decided as this moment and would be implemented in the ad hoc fashion. The timeline for each activity is a bit complex to predict and it would be premature to comment now.

- Currency conversion based on the current foreign exchange rates. This service is offered through the "Google finance API". This service provided over the URL "https://developers.google.com/finance/?hl=en". This would require the inputs of currency

codes and the amount to be converted. The output would be a value in another currency. The service implementation is relatively on the easier side.

- The next service offered is the translation of the text provided to a language of user choice. This service is offered by the frengly.com. The URL for the REST service interaction is "http://www.frengly.com/". The translation service takes the input of the source and destination language codes along with the text to be translated. The service generates the translated text. The service implementation although easy would be time consuming.
- For providing the feature of suggesting the places of interest the system would interact with the "Google Maps API" and a third party service that interacts with this map API. "https://developers.google.com/maps/?hl=en" is the URL for interacting with this API. This service implementation would be a tough job as it deals with providing location and details of places that being suggested. The input for this file would be the coordinates of the place being queried by the user. The output would be a map view of the places being suggested.
- The feature of creating, tracking and deleting items and tasks to be done is being built by the system using the mongoDB database hosted on mongo lab hosting service. This service would be designed within the system. This service implementation deviates from the normal process web service consumption. Instead we would develop a web service that stores, retrieves and manipulates data to and from a database. We have implemented this feature using the API calls and we are in the process of writing business logic in Java i.e Rest Services so that time lag faced during the API calls would be reduced using the REST services .
- The weather prediction is based on the data being returned from the web service provided by "Openweathermap.org" in the form an API. The URL for this REST service is "http://openweathermap.org/". The task mostly concentrates on the UI display of the information being retrieved. The service consumes the data of place for which the weather is predicted. The output of the service would be a UI displaying the weather information generated by the service. We have implemented this feature successfully and tested the functionality and the test cases have been successfully passed.
- For the feature of accommodation suggestions we have this feature will be implemented in next phase. This would be winded up pretty soon. This service implementation would require little time as there is a reuse of code and ideology taken from the implementation of the places of interest service. The time factor is moderate.

Framework specifications

The system would be a flexible framework that can incorporate the services that are customized to suit the user and application needs. The system architecture would comprise of a client server architecture in which our application would be a client accessing service hosted or offered through different servers. Internet would be the medium of communication between the application and the services hosted in the web. One of the service that the application offers requires the traditional 3 tier architecture where in the data is stored through the UI layer into the business layer and ends up in the database. The system architecture is represented in the diagram below.

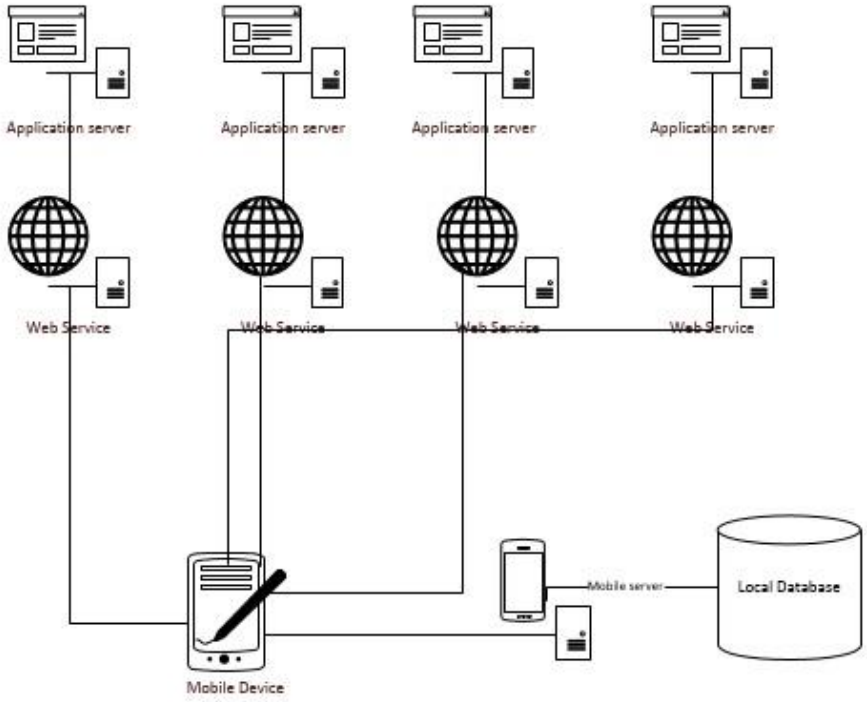


Figure 1: Pictorial representation of the system architecture

Second Increment Report

This document is a report of second iteration of work performed on the Travel Guide Mobile App. The document emphasizes on the details of the app and the progress of the work done so far in developing the app. This document intends to provide an overall description of the project named “Travel Guide” in detail.

The outcome of the second increment is weather, language converter and task modules with the functionality and basic look and feel. As of current state we have not deviated from our initial proposal that we submitted earlier. We have taken care of the implementation of app as explained in the class diagrams and sequence diagrams.

The detailed development process include the Login Page using Facebook API , weather, language translator and task manager module. Completed the Unit Testing of the login module. The other page include the Home Page where we have covered the profile part and design of the functionalities which we are going to implement in the next phases.

We have also checked the performance of the app by installing in the mobile device. we have concentrated more on the functionality rather than the look and feel for the app. The look and feel will be concentrated as part of the next increment.

Class diagram for the high level design of the system

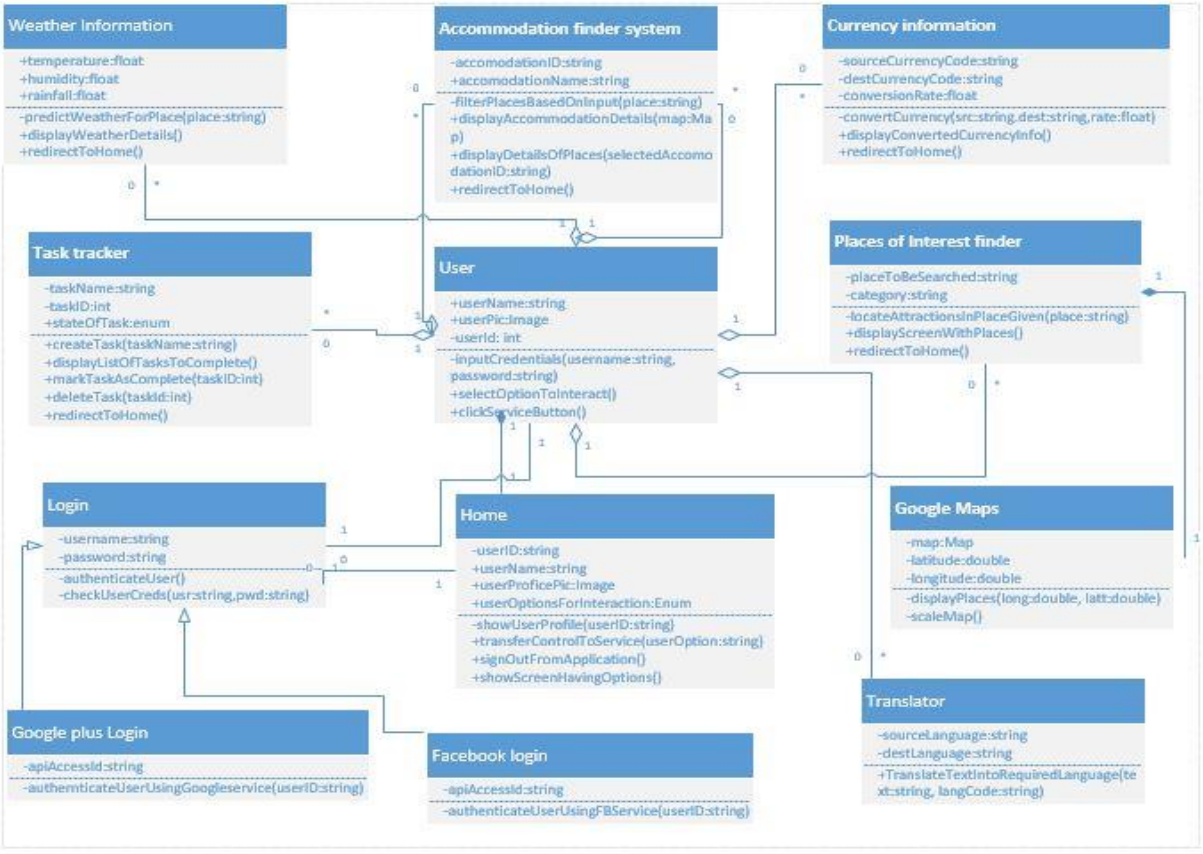


Figure 1

Figure 1 demonstrates a class diagram which explains High Level Design Architecture of the System. In the system the Home controller is the central control for the smooth operation of the app. Each class in the system is an implementation of an API that is integrated into the system. The user uses the login controller to reach the home controller and then the interaction happens between the various classes and the home controller class. The places of interest finder class uses the Google maps instance to provide the desired results. There are 2 ways to login to the system. One is the normal routine way of providing the username and the password. The other way is to use social networking like Facebook and Google plus. The user class interacts with the other components through the home controller.

Sequence diagram for each of the web service interaction

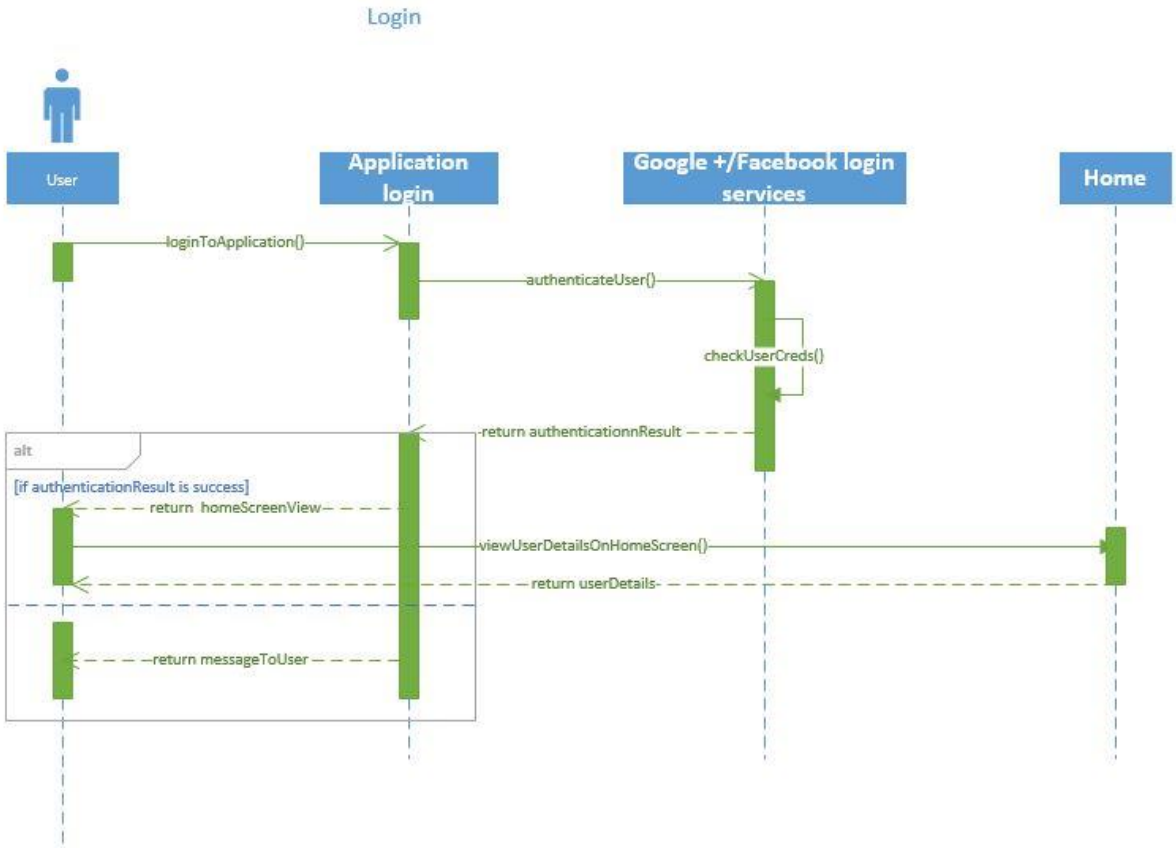


Figure 2

Figure 2 demonstrates a sequence diagram about how a user can Login to the App by using different Sign In options like Facebook, Google+. User has to provide credentials to the social media login page. In case of the Facebook login the user's phone if has Facebook then there is no need to provide the credentials for the app. The case for google plus is normal wherein the user is required to provide credentials.

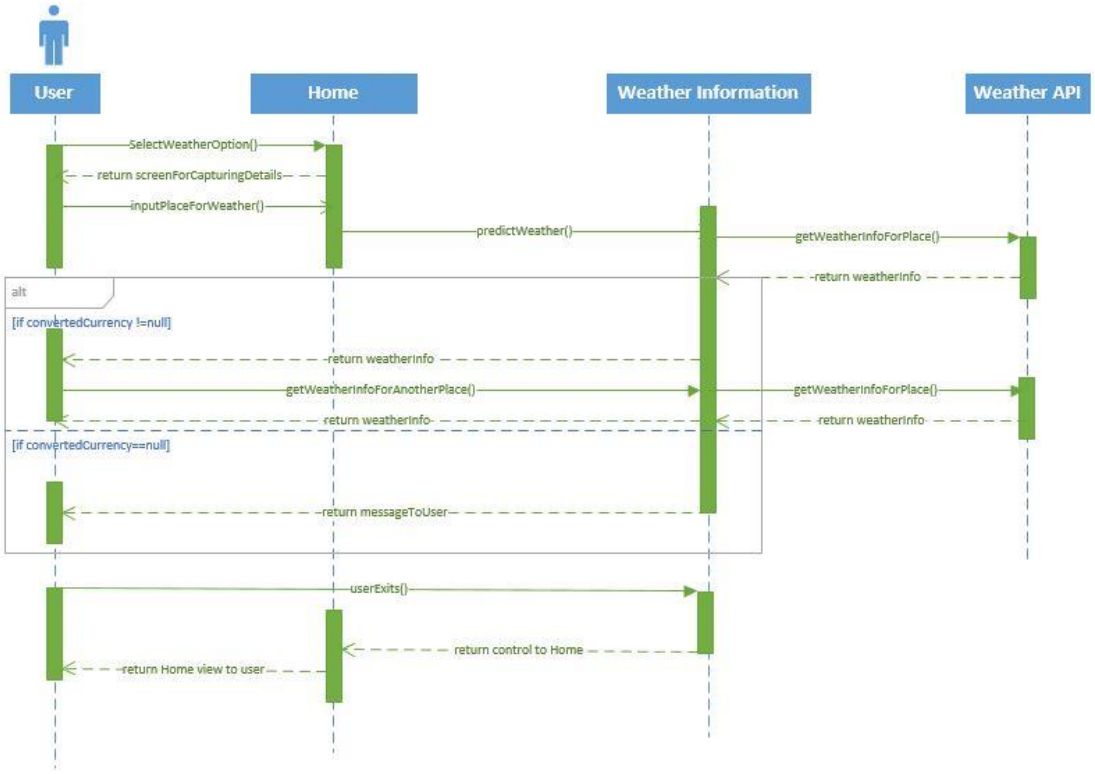


Figure 3

Figure 3 demonstrates a Sequence diagram about how a user can get the Weather of a particular location of user's choice. Once user selects a place, app communicates with API and displays the temperature and weather information on the screen

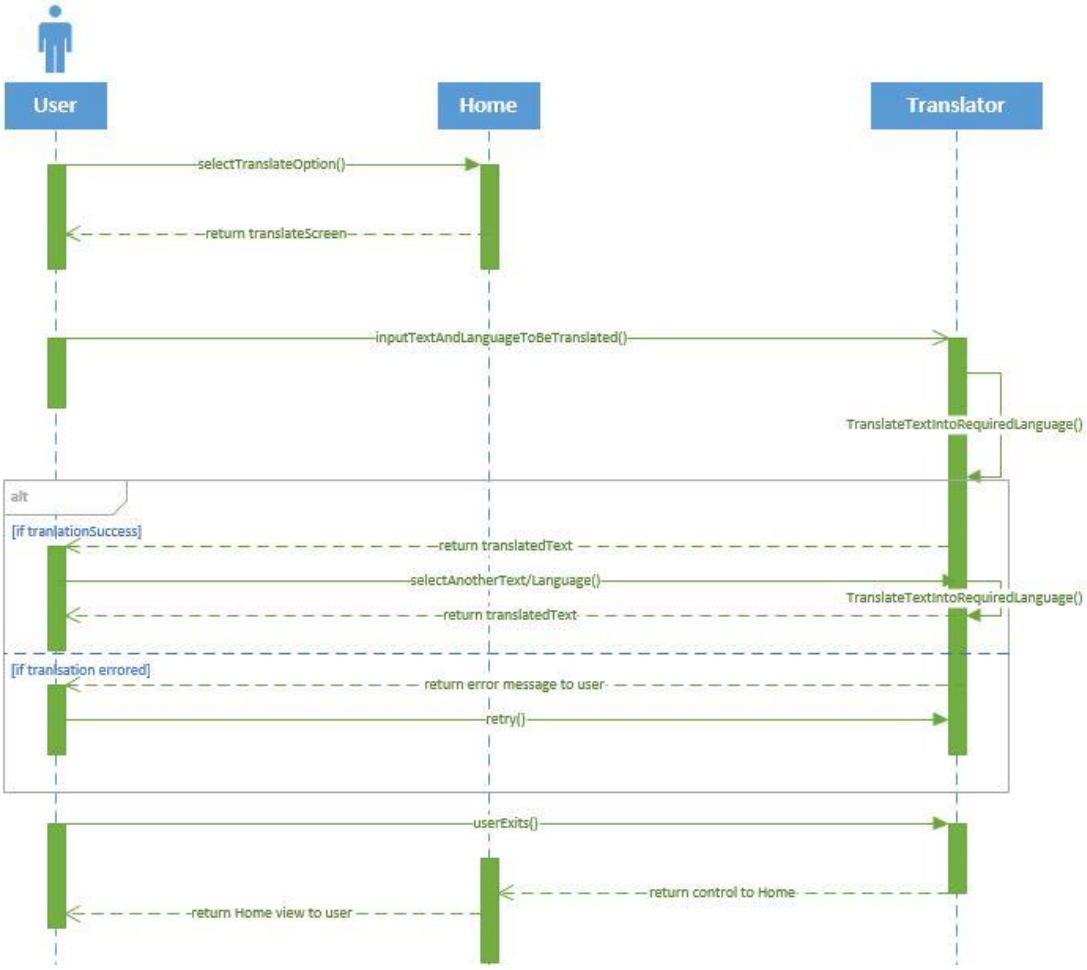


Figure 4

Figure 4 demonstrates a sequence diagram about how a user can use this App to translate the text from a Language to a Language of user's choice. App uses API from frengly.com for language conversion. The choice of the source and the destination language can be a user choice. User has to enter text in one language and select the language that he wants to be converted.

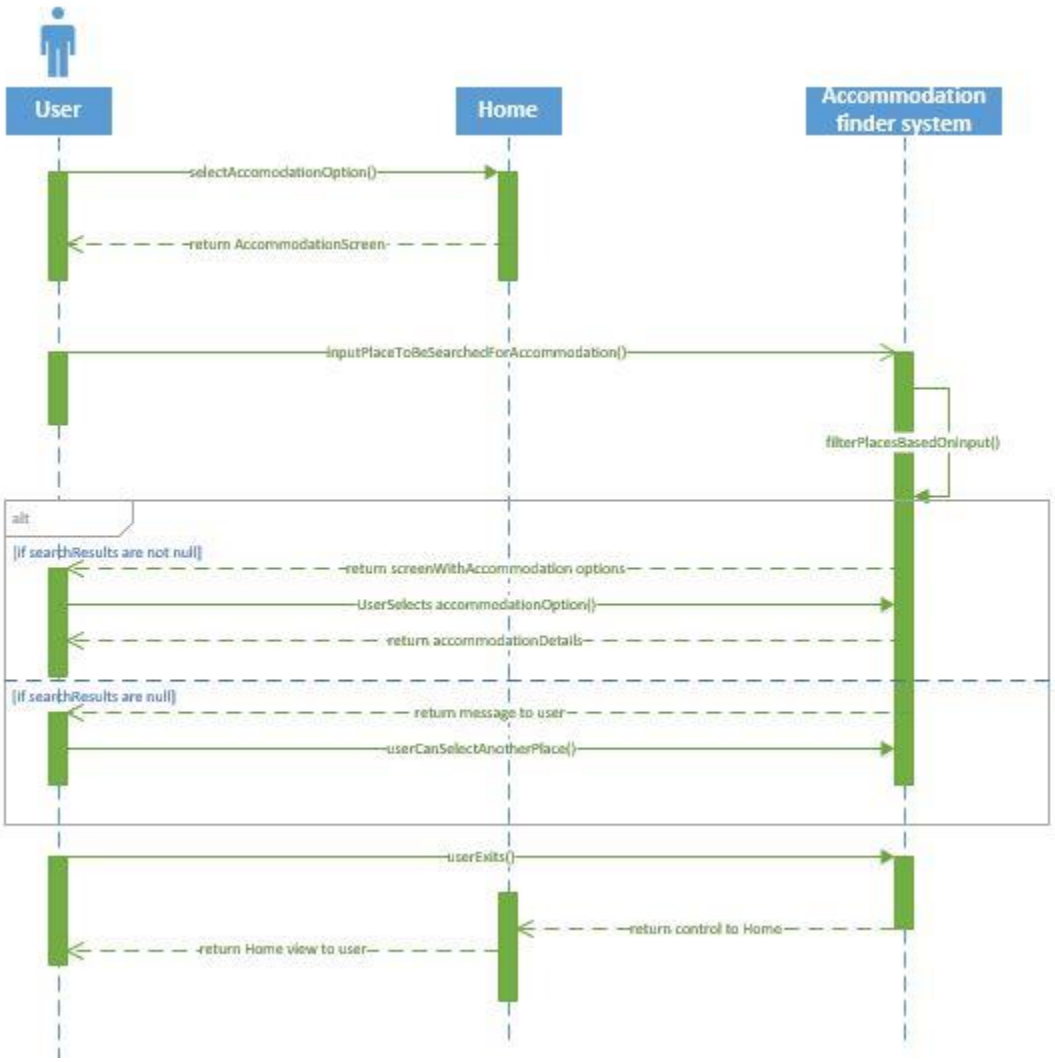


Figure 5

Figure 5 demonstrates a sequence diagram about how a user can get the Accommodation details of a particular location of user's choice. This feature makes use of the google maps service to improve the results provided by the location basis. Each of the place is indicated in the map.

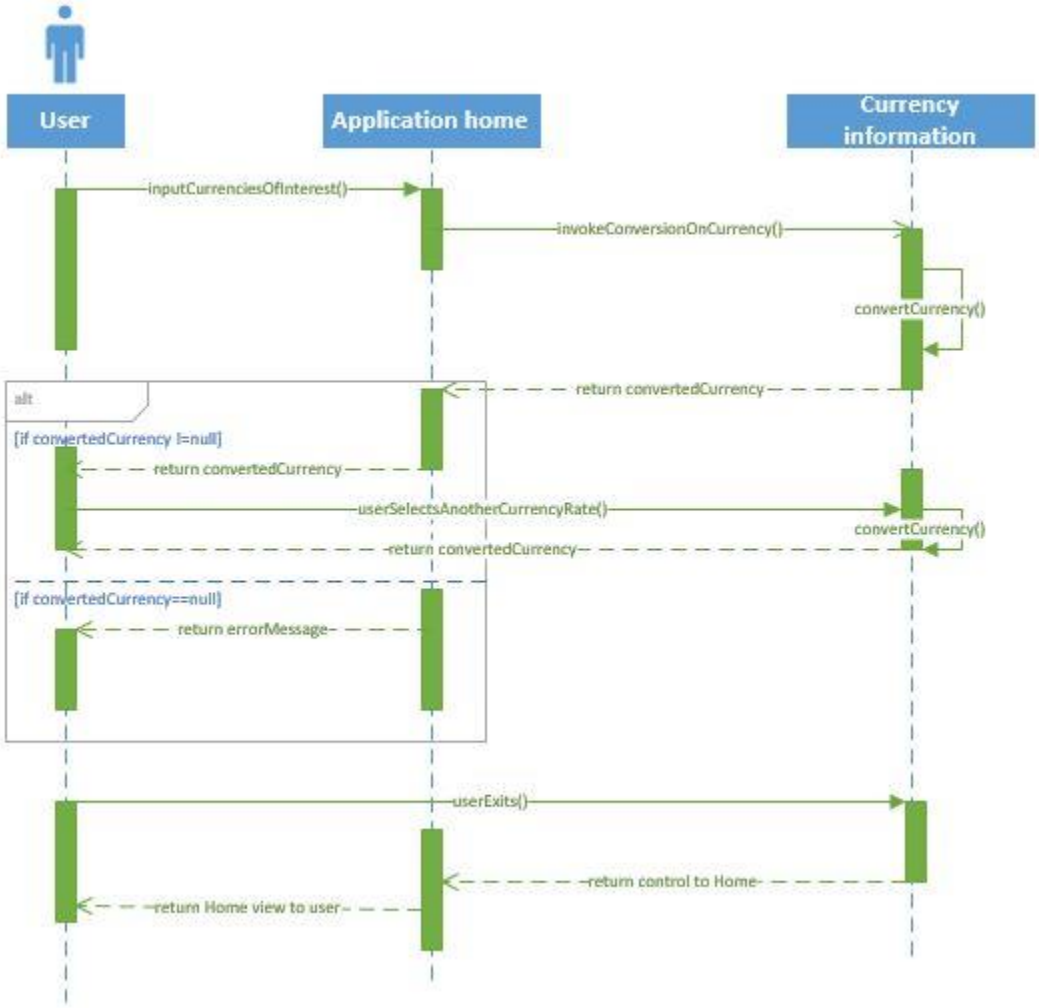


Figure 6

Figure 6 demonstrates a sequence diagram about how a user can get the varying currency rates of the Specified countries on that particular day. The currency exchange rates are updated per transaction basis. The user selects a particular currency and enters the value. The converted rate is displayed to the user.

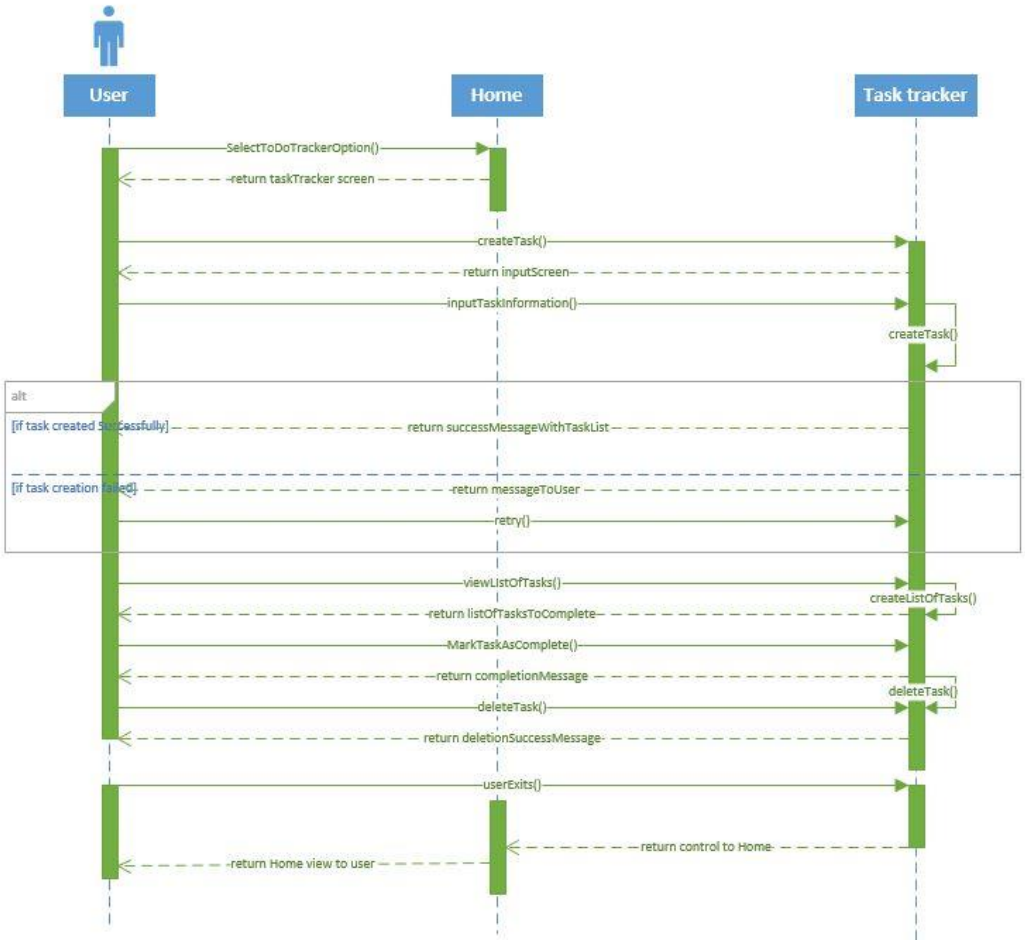


Figure 7

Figure 7 demonstrates a sequence diagram about how a user can create task. There is a feature to update the task title and the description of the task. Each task requires title and a description from the user. Each task can be scheduled for a notification so that a reminder can be provided to a user.

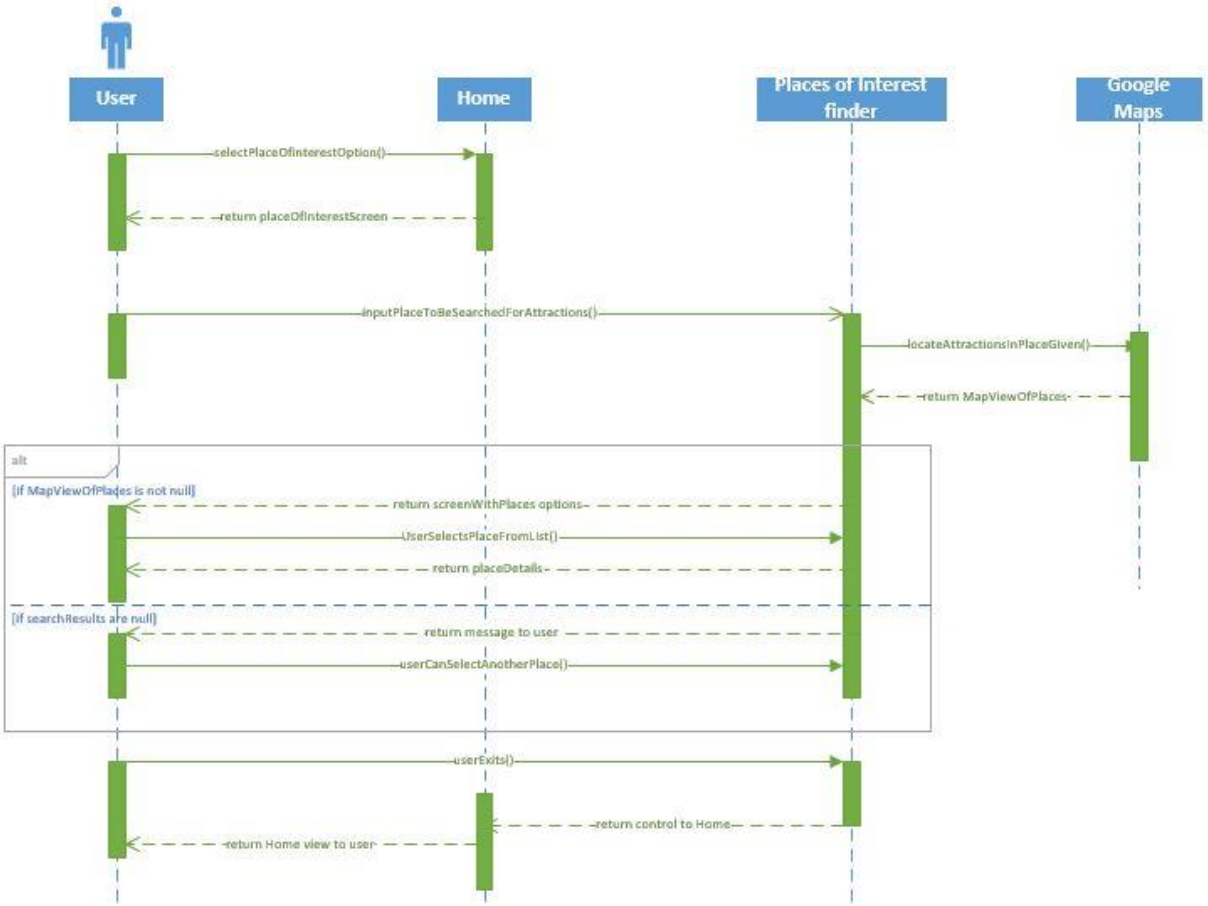


Figure 8

Figure 8 demonstrates a sequence diagram about how a user can get the places around a particular location of user's choice.

State diagram of the system

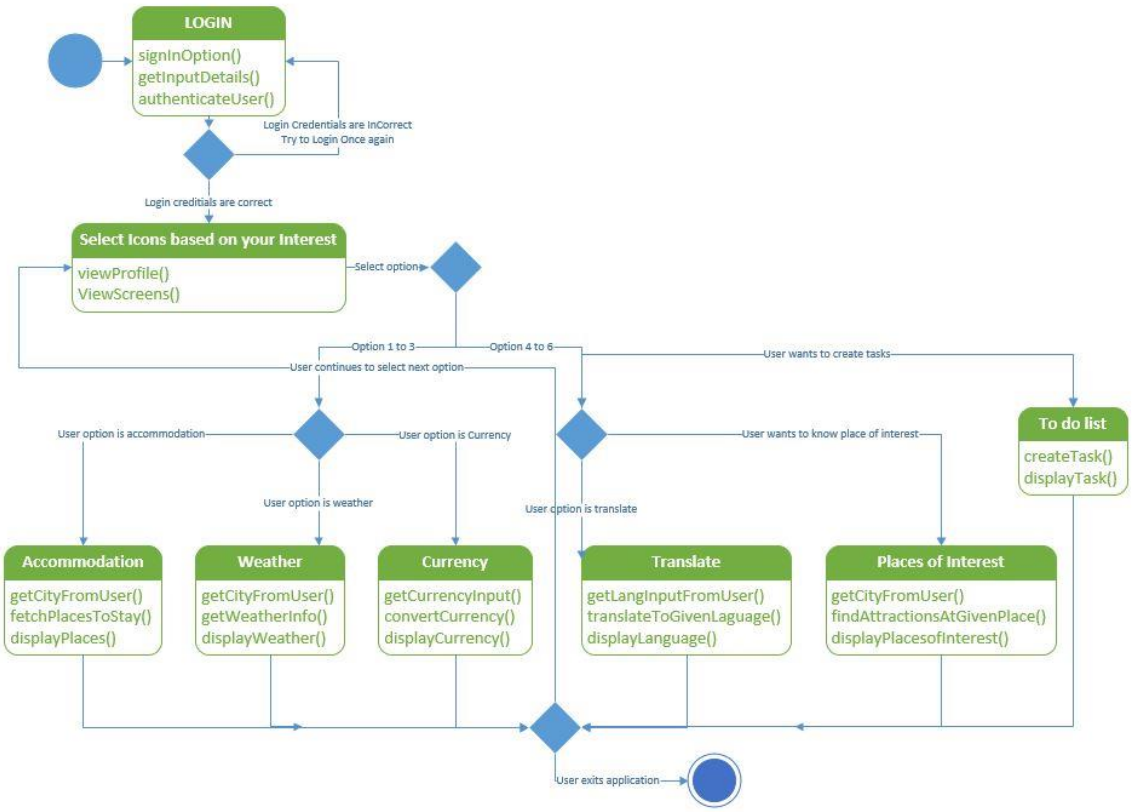


Figure 9

Figure 9 demonstrates a state diagram which explains High Level Design Architecture of the System. The initial step of the system is to login to the system. If the user has a social networking account like google plus or facebook he can login to the system. Else he could register himself with the app by providing some basic details. After successful login to the system the user views the home page where he can choose the feature he wants to interact with. Here the user has six options to choose from. Based on what he chooses the users follows the steps to interact with the corresponding feature. After successful interaction with a particular feature the user can end his session and close application. On the other hand he could visit the home page and begin interaction with another service.

Wireframes for the application

Figure 10

Figure 10 represents the startup screen for the travel guide APP. A user can use either Facebook or Google+ credentials to login into the app. The app accesses only display picture and basic details from the social networking application for its use and it will be confidential.

Figure 11 represents a wireframe for Homepage. ALl the features can be accesible from the home page. every button represents a feature. User will be taken to the feature specific page.

Figure 12

Figure 12 represents the Currency Converter Screen. User can use this feature to know the current value of a particular currency.

Figure 13

Figure 13 represents the Wireframe for Places of Interest. User can find the list of places in and around a place.Restaurants, movies, muesuems are some of the items that will be marked on the map.

Figure 14

Figure 14 represents the wireFrame of CheckList Screen. This feature works as a reminder to the user about the taks he has to do.

Figure 15

Figure 15 represents the WireFrame for Language Translator Screen.This feature helps user to learn foreign language for basic communication. Language translator support is given for a set of languages.

Figure 16

Figure 16 represents the WireFrame of Accomodation Finder Screen.

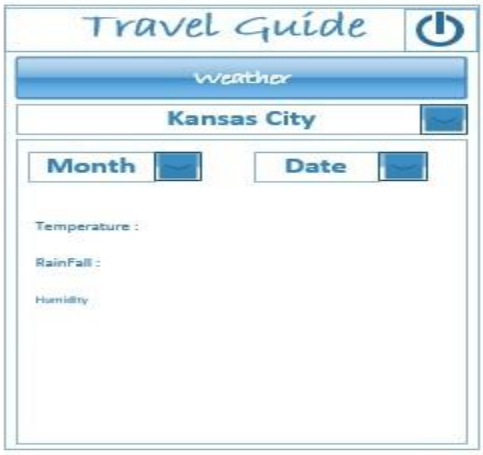


Figure 17

Figure 17 represents the WireFrame of Weather Screen.This feature gives weather information for a place for period of 7 days.

Mockups for the application

Figure 18

Figure 18 represents the Mockup for Login Screen.This figure depicts the login screen of travel guide.

Figure 19

Figure 19 represents the Mockup for Homepage Screen.

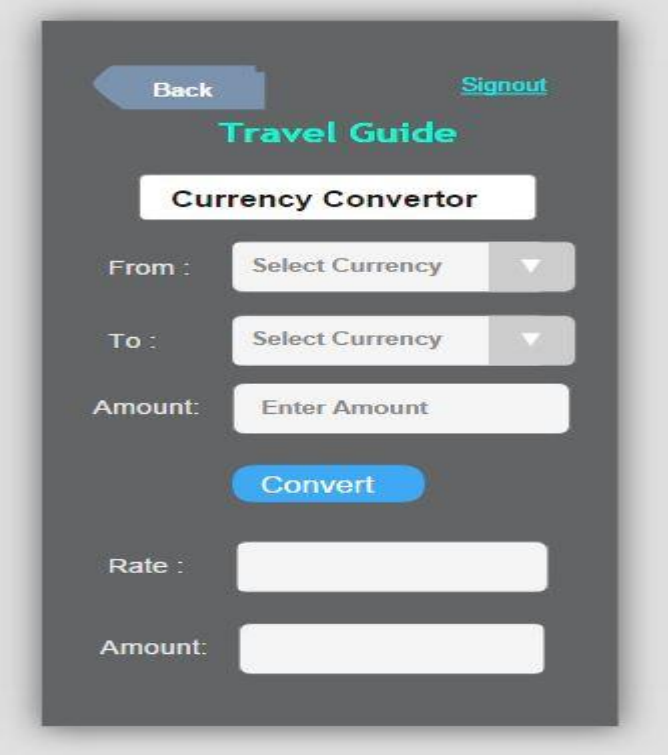


Figure 20

Figure 20 Represents the Mockup for Currency Converter Screen.

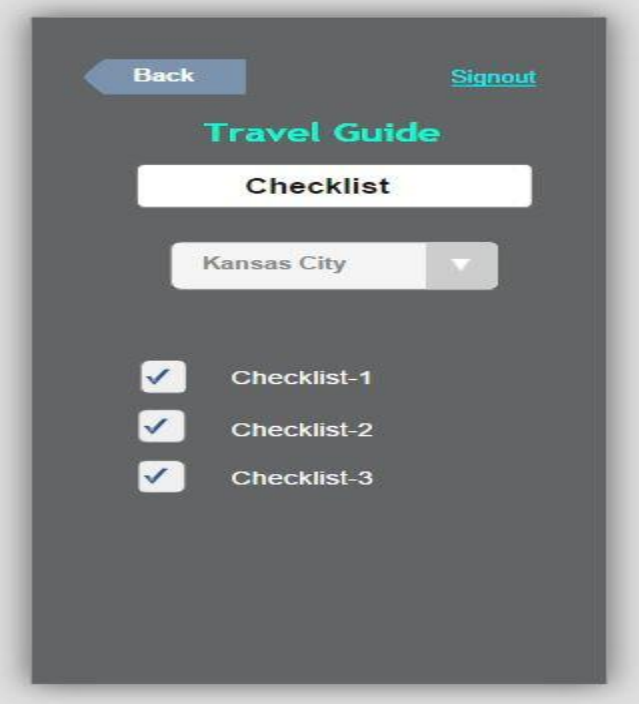


Figure 21

Figure 21 represents the Mockup for Checklist Screen.

Figure 22

Figure 22 represents the Mockup for Accommodation Finder Screen.

Figure 23

Figure 23 represents the Mockup for Weather Screen

Figure 24

Figure 24 represents the Mockup for Places of Interest Screen.

Figure 25

Figure 25 represents the Mockup for Language Translator Screen.

User Stories:

- As a User, I can see the UI of TravelGuide App.
- As a user, I can see the Login Screen.
- As a user, I can enter my Details and register to the App.
- As a user, I can signin with Facebook.
- As a user, I can see the Icons on Home Page like Weather, Currency Converter, Task Reminder, Accommodation and Language Translator.
- As a User, I can enter the City that I want to see its Weather Conditions.
- As a User, I can see the Weather of a Specified City.
- As a User, I can create tasks.
- As a User, I can choose the Languages to get the Text converted into my Preferred Language.
- As a User, I am able to see the language translated to my preferred Language.
- As a User, I can go to preferred location based on the icon clicked.
- As a user, I can enter the city to find Place of Accommodation.
- As a User, I can Query the Places of Interest near a Specified City.
- As a User, I can see the Places of Interest of that City.
- As a User, I can enter the city that I want to know about the Currency of that City.
- As a User, I am able to see the Currency of the Specified City.

As a user, I can Sign-out from the Application.

Testing:

We have performed code walkthroughs,peer reviews and integration testing.

We have also performed manual testing, Smoke test on overall App to ensure the correctness of the App functionality.

- Unit Testing

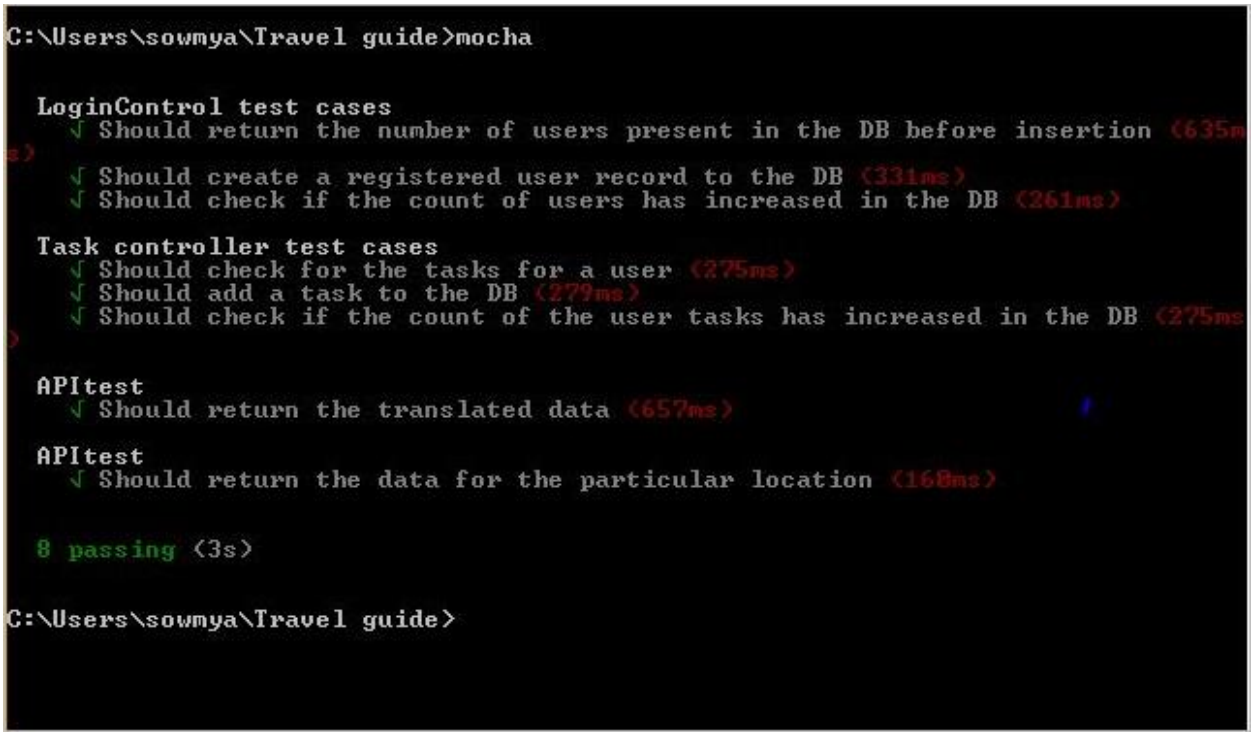


Figure 38 : Unit Test cases

LoginController Test Case: We have performed 3 testcases for testing Login functionality.

i) To fetch the number of users present in the database.

ii) To check whether the registered user details are mapped to the mongodb.

iii) To check whether the count has increased in the users table.

TaskController Test Case : We have performed 3 testcases for testing Task functionality.

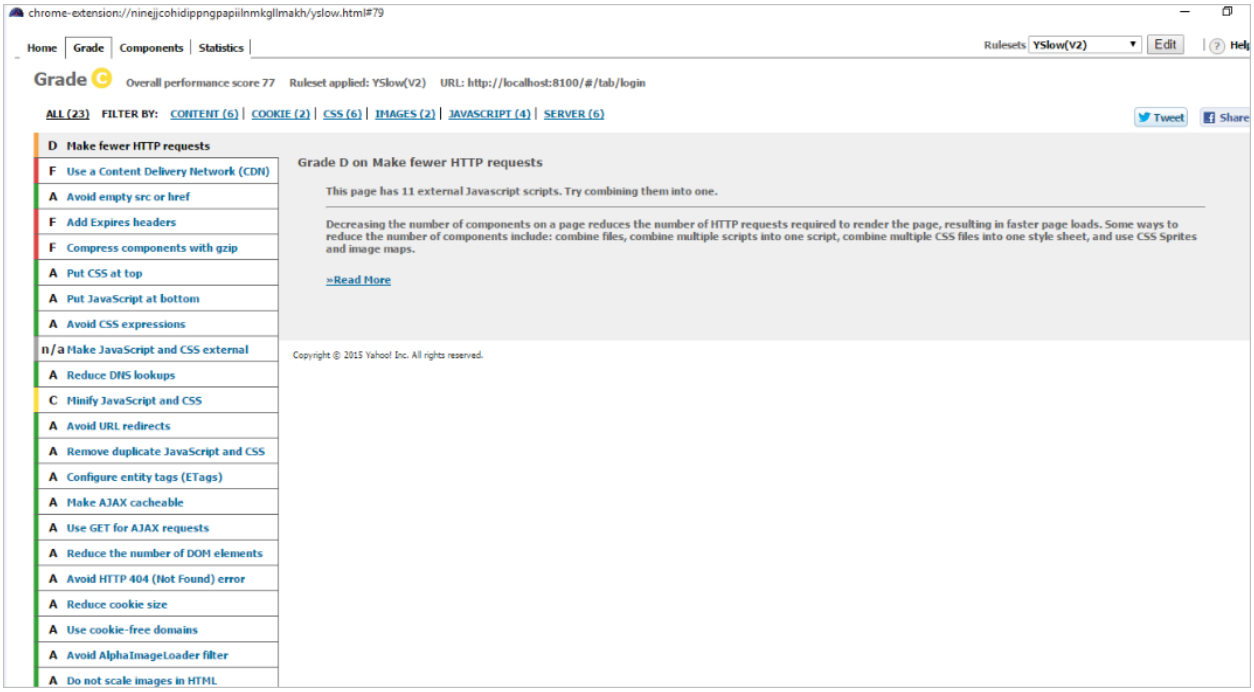
i) To fetch the tasks of a user present in the database.

ii) To check whether the record is added upon succesfull creation of task into the mongodb.

iii) To check whether the count has increased in the tasks table.

APITest cases : We have tested the api services used for weather and language translator whether it providng the values.

- Performance Testing



Project Management Site Link

https://pradeep123.kanbantool.com/boards/180576-welcome-to-kanbantool

Work completed

The detailed development process include the Login Page using Facebook API's, weather, language translator and task manager module. Completed the Unit Testing of each module separately. The other page include the Home Page where we have covered the profile part and design of the functionalities which we are going to implement in the next phases.

- **Weather Module:**

Description: successfully developed weather module with clear picture displaying temperatures with the help of a graphical notation. The weather information is display for the 7 days in future for any city selected by the user.

Responsibility: RaviTeja

Time Taken: 75 hours

- **Language Translator**

Description: successfully developed language translator for converting phrases of a languages to other set of languages

Responsibility: Pradeep

Time Taken: 75 hours

- **Task Module**

Description: Successfully developed task module where in user can creates tasks and can add reminders to the tasks.

Responsibility: Ragunandan

Time Taken: 75 hours

- **User Interface,MongoDB setup**

Description: Completed implementation the IONIC Framework for UI as we previously implemented standard HTML.Successfully written API calls for storing the values in the backend (Mongo DB).Completed the documentation and few Rest Services to store the data in the backend without using API calls.

Time Taken: 75 hours

Responsibility: Sandeep

Work to be completed

The modules scheduled for next phase

Place of interest

Description: App should provide the list of interested places in and around a particular place

Responsibility: RaviTeja

Estimated Time: 80 hours

- **Accommodation module**

Description: App should provide the list of available accommodation options around a particular place. Implementing REST services by writing the business logic to store the values without using API calls.

Responsibility: Sandeep

Estimated time: 80 hours

- **Session Management**

Description: App should be consistent with the session management for a particular user

Responsibility: Ragunandan

Estimated Hours: 80 hours

- **Currency Converter**

Description: App should provide user with the currency conversion details of various currencies.

Responsibility: Pradeep

Estimated hours: 80 hours

Issues/Concerns:

Test cases in karma and Jasmine for services.

We have also checked the performance of the app by installing in the mobile device. We have concentrated more on the functionality rather than the look and feel for the app. The look and feel will be concentrated as part of the next increment.

Screenshots of the initial application UI

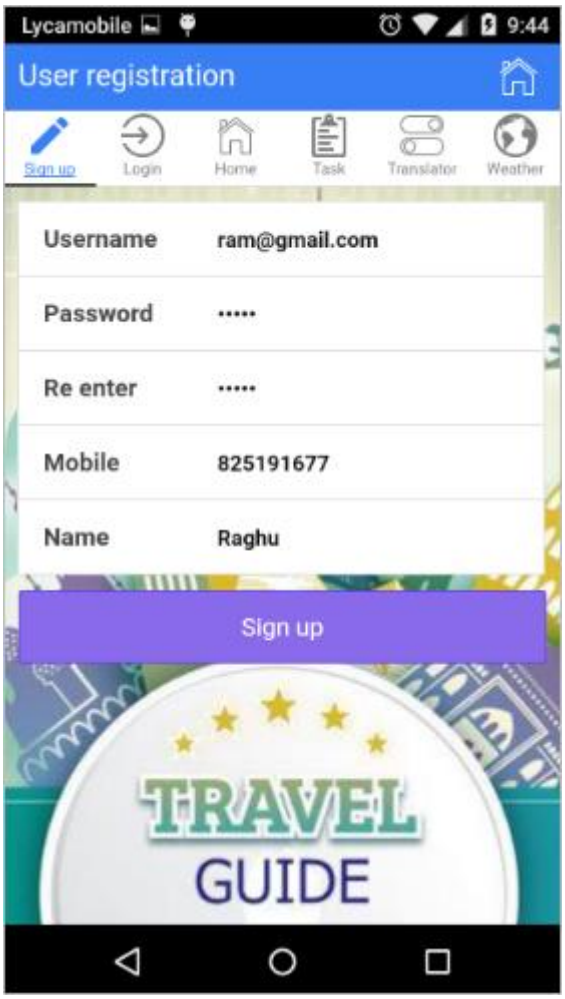


Figure 26 : Registration page

The above figure shows the registration page where the user needs to provide the required information to register with the app.The details will be stored in the MongoDB hosted in Mongolab.

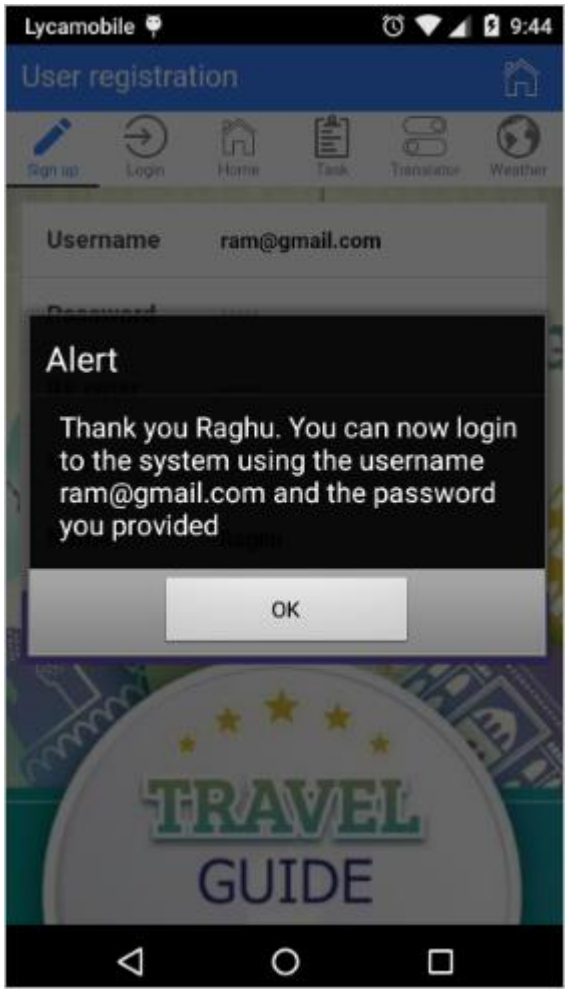


Figure 27: Successful Registartion Prompt

Once the user registers with the app ,upon successful registration the user will be prompter with above message. This goves the user a confirmation that he can now login to the system using the username and password that he has provided during registration.

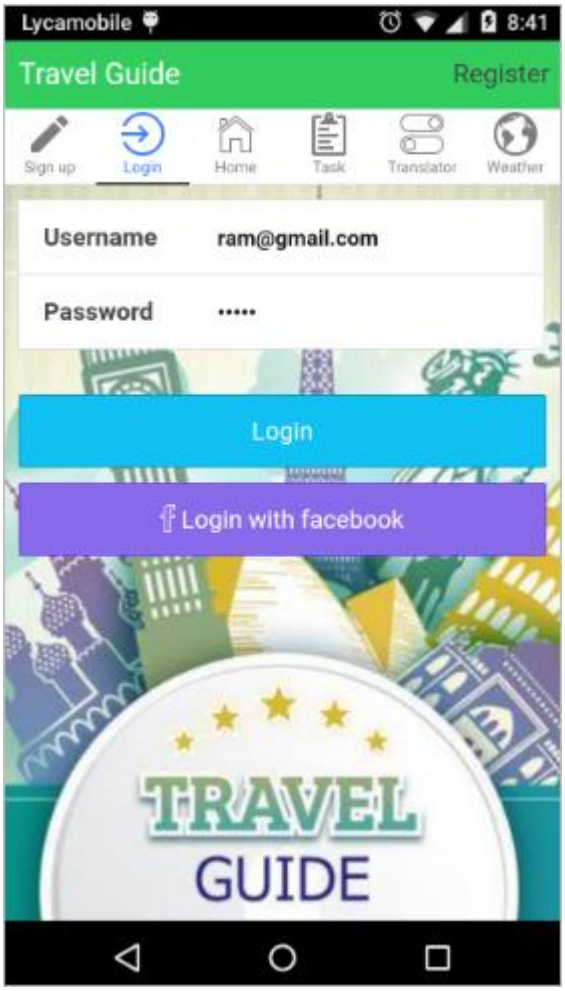


Figure 28 : Login Sceen

User has to provide the credentials registered with the App to login.The details will be fetched from the MongoDb which has been stored earlier while registration. The other option of login to the system is through the usage of facebook credentials. In this case if the user has facebook app already on his device need not provide his credentials again.

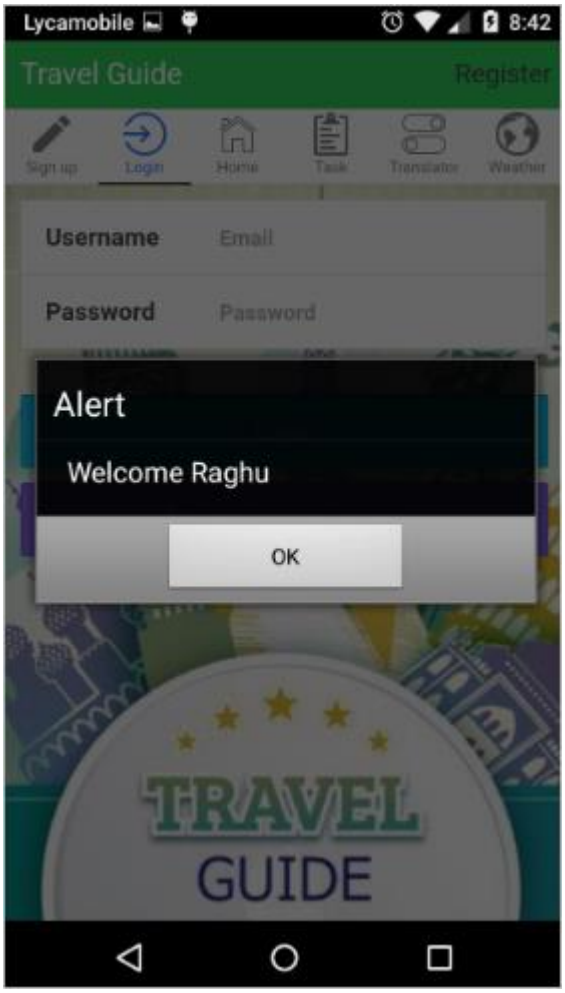


Figure 29 : Welcome message

The above welcome message will be prompted upon Successful Login. This message only pops up when the user has entered valid credentials while login to the system. The username of the logged in user is displayed in the messafe.

Figure 30: Home Screen

The HomeScreen consists of Icons for the various features of the APP starting from top Accomodation,Currency Convertor,Checklist,Places of Interest,Language Translator, Weather . This is the central hub for the hub. The user has to interact with the features of the system through the home page. There is an option to logout of the system here.

Figure 31 : Weather Screen

The weather feature will provide the Weather Forecast for a week starting from today for any place in graphical Notation. The minimum and the maxiumum temperature for that day is provided. The graph provided the range of the maximum and the minimum temperatures for the a range of one week for the place of interest.

Figure 32: Language Translator Screen

The Language Translator provides the user a feature to translate Phrases into a foreign Language. The user has to enter the text in the language he knows. Then he can select the language for translation. The translated text is displayed in the text are below.

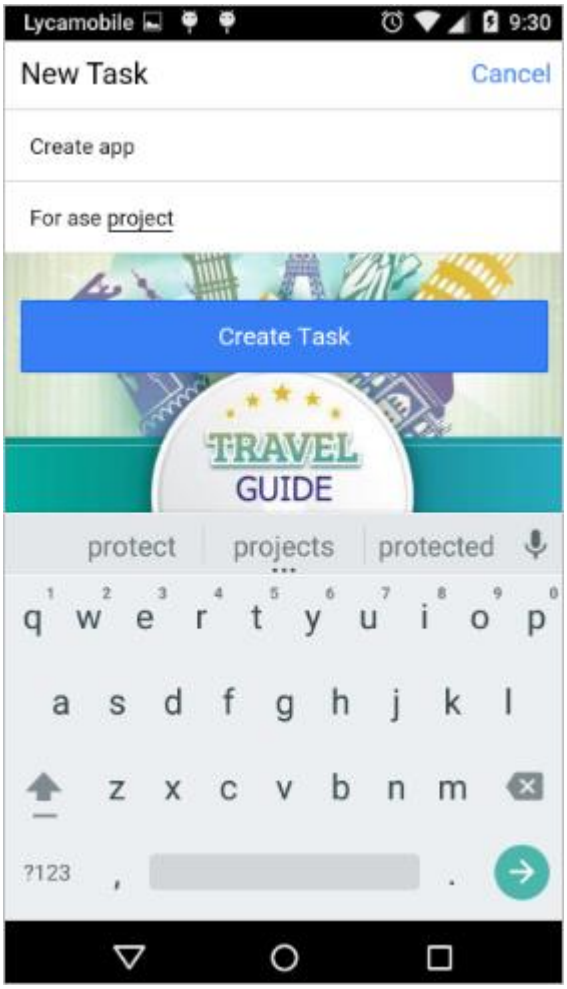


Figure 33: Create Task Screen

The Task module helps users to create a checklist of tasks and set notiifications so that a reminder will be created. The user can udpate the task in case he has any changes to be tracked. The user can set reminder for his task so that he can get a notification when he wants for that task.

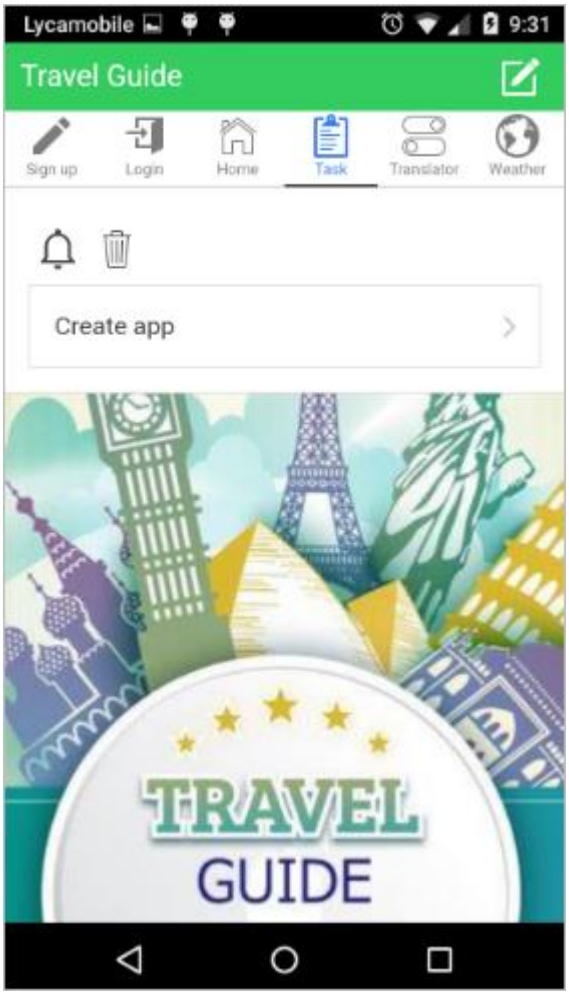


Figure 34 : Tasks List

The above screen consists of the list of tasks created by the user.”Create app” is a task in the tasks list.

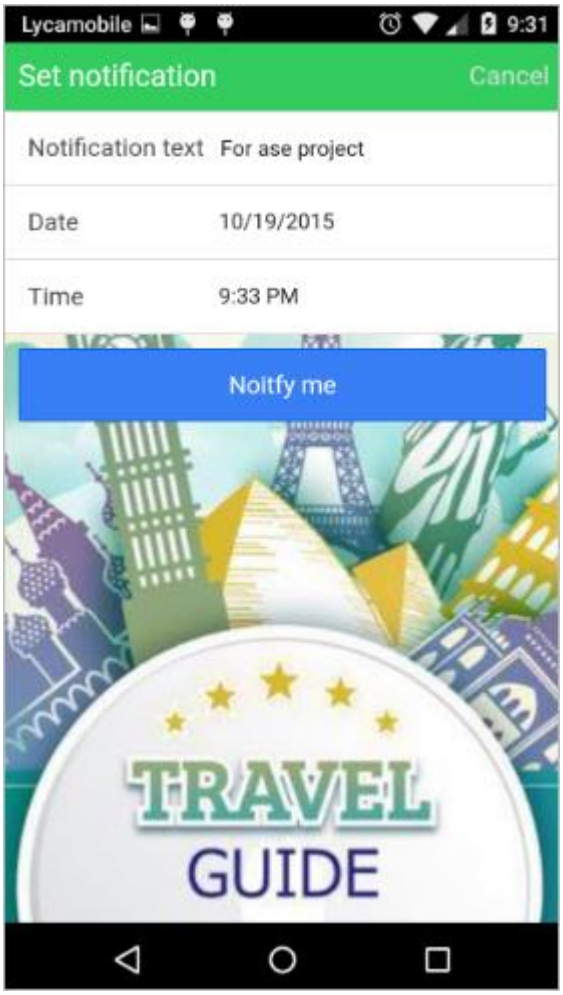


Figure 35: Setting Notifications

To generate reminders for a certain Task, a notification needs to set providing the date and time.

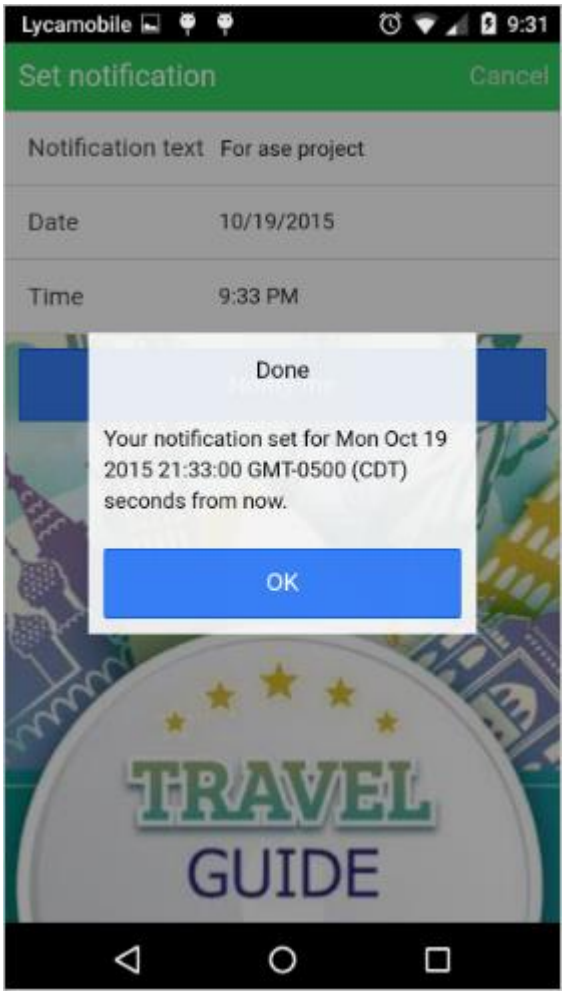


Figure 36 : Setting Notification

The above screen shows the Notification set for a “Ase Project” Task.

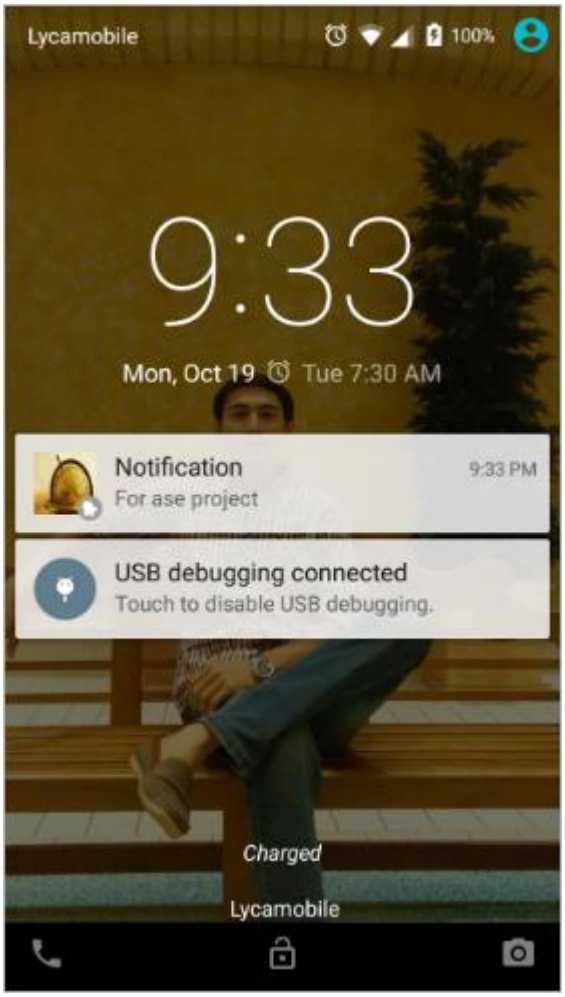


Figure 37: Notification Alert

The above screen shows the notification alert for a “Ase Project” Task

Bibliography

[1], [2], [3] and [4] – Content referred from the article “The world’s 50 best travel apps” by the Time out group. URL: http://www.timeout.com/travel/features/1169/the-worlds-50-best-travel-apps

References:

“Packing list” by Travellerspoint Foundation. URL: http://www.travellerspoint.com/packing-list.cfm
“Kanban tool” used for the planning phase, URL: http://kanbantool.com/
The guide for Google plus login URL: https://developers.google.com/+/web/signin/
Facebook sign in option guided by URL: https://developers.facebook.com/docs/facebook-login