UMKC

Spring-2017

Advanced Software Engineering

Final Report

Submitted by Team19:

Gattu, Sampath

Kallakuri, Sadanand

Pottabathula, Prathyusha

Nalla, Abhiram Reddy

Table of Content

1. Introduction

Travel buddy is an application, one stop to all the requirements for those who loves to travel. Now in a smart world there is requirement for a smart application which provides help to users traveling. Travel buddy provides a unique platform which gives information about weather, accommodation, currency conversions, twitter timelines for activities around you and many more.

We planned to create an application which has a registration page and login page. For the First time, users are supposed to register for further access. We are also including the social sign in plugins so that any user can login without the hassle of signup/registration forms. Home page has different functionalities such as weather details, hotel search, twitter timeline, translate, emergency services . etc.

2. Project Goals and Objective

2.1 Motivation

Travelling has become an integral part of our lives and has become so much more than just purchasing a ticket and boarding a plane with a map and a compass. With the immense technology, we have a better chance of pump up our escapade on an adventure trip. Therefore, we have chosen to develop Travel Buddy (travel guide) for easy travel.

2.2 Significance/Uniqueness

Currently, there are so many applications for travelling but our application will be somewhat different compared to them. At present, there are applications which are created showing the nearest restaurants, best places to visit. In our application, all features are integrated into a one application. In addition to the above specifications, our app will try to implement many other specifications such as if a person wants to travel from Kansas City to Chicago and he has a car and travelling alone, then he can post his date of traveling in the app, so that others can see and make a ride which will be less than the usual cost of transport.

2.3 Objectives

- The objective of our project is to develop an application where the user can login and make his travelling easy using our application. The application should have a good user interface so that user has easy feel of using the application.
- User's effort to get multiple information is reduced as we provide all information in one application.
- User's effort for searching the new places is reduced.

2.4 System Features

User should able to login to application securely.

Accommodation: This feature provides the user for selecting his stay of visit based on his expenditures and his preferences.

Car Rental: It would provide options for taking rental cars.

Reviews: Travelers are given chance to give their rating for the hotels and nearby best places to visit.

Searching for passengers: Person who has a car traveling from one place to another can ask for friends if they wanted to travel to same place.

Language Translator: This feature provides user to translate from any other language to the language that he prefers. This features provides to translate to more than one language at a time.

Weather Report: The information about the weather will be provided.

3. Related Work

Weather Pro: Weather pro application provides information on the weather of several places provided by user.

Trip Advisor: This application offers user reviews on restaurants, hotels and attraction places which can be visited at the location.

Hostel world: This application provides the user the best hotel for their available budget.

XE Currency: This application gives the currency exchange at present.

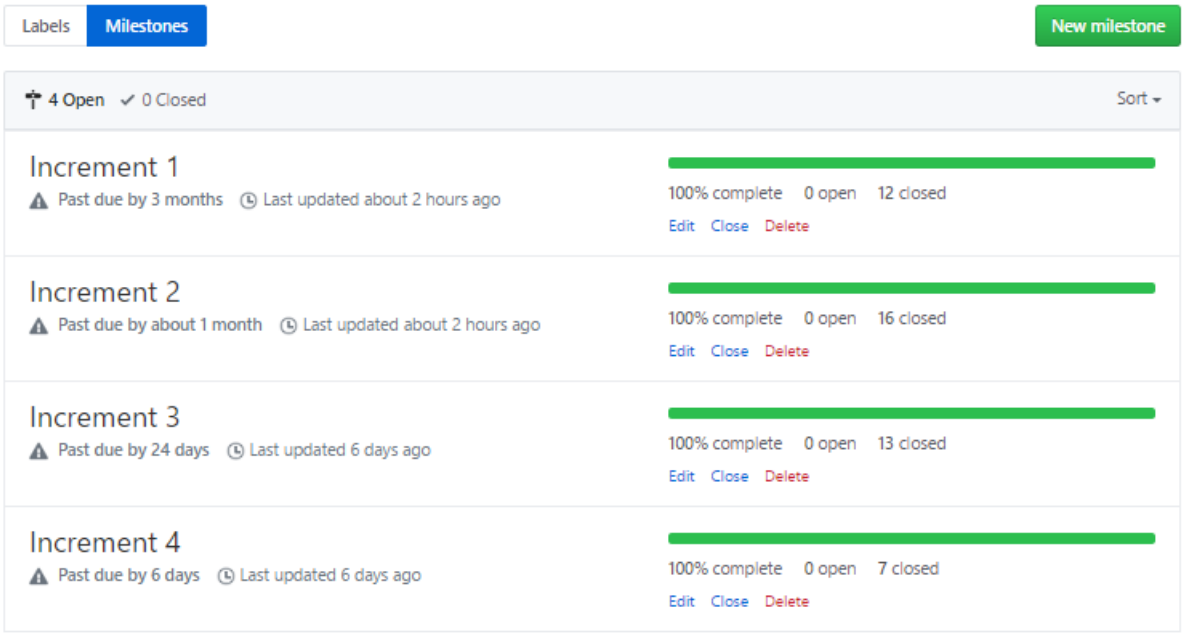
4. Project Plan

Agile Model is used to develop the project which helped us to develop the project in increments. In agile model, each increment is tested to maintain the quality of application. Due to Agile process changes are easily adapted.

The entire project is divided into **4 increments** and issues are created under each increment. The tasks are assigned to individual team members. Once the tasks are completed it is moved to closed state. The progress of each increment is visualized in Burndown Chart.

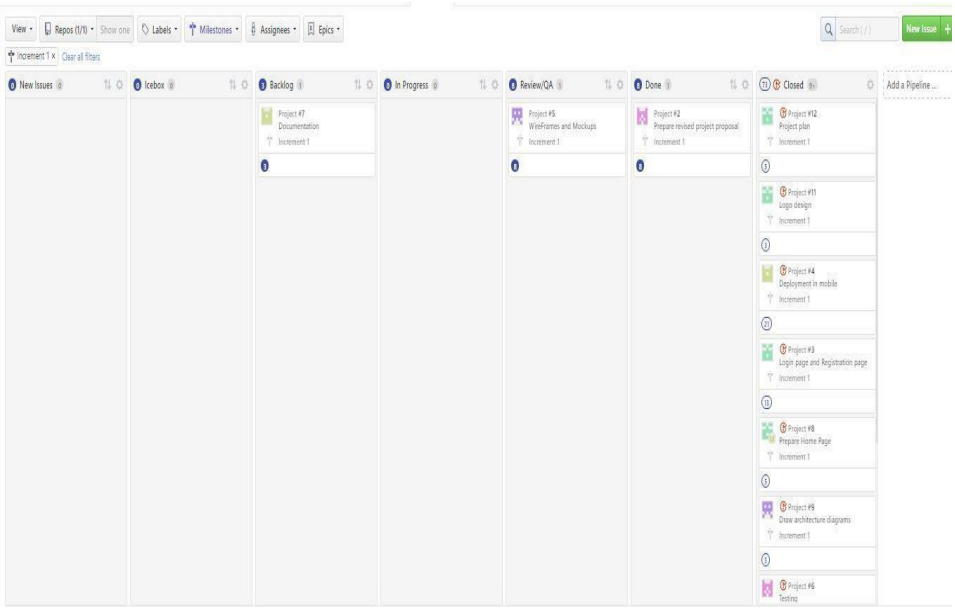
Increment	Start Date	End Date	Work Planned
Increment 1	02/07/2017	02/17/2017	Design and Architecture of the Application. Login, Registration and home page. Created Class diagram, Use case, Sequence diagram, Architecture diagram. Created increments in Zenhub. Created wireframes.
Increment 2	02/18/2017	03/10/2017	User credentials stored in mongo DB,Weather API , Translator, Hotel search implementation
Increment 3	03/11/2017	04/10/2017	Ionic Login, Registration and home page Ionic Login page register page validation, Twitter Timeline, Currency Converter
Increment 4	04/11/2017	04/28/2017	Event API, Mongo retrieval, To-do List.

In ZenHub, 4 Milestones are created for each increment and issues are created under each milestone and assigned to respective team member.

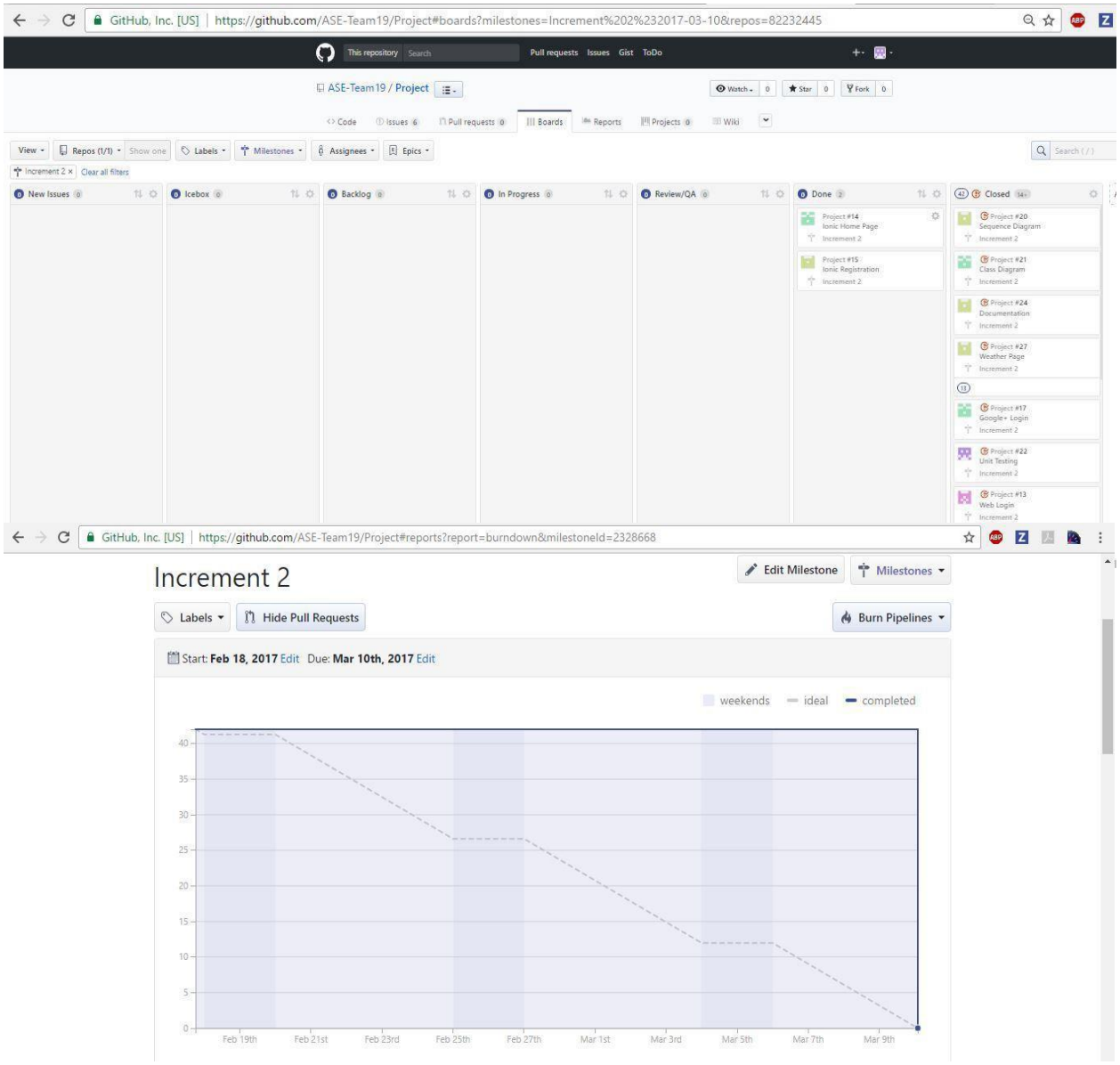


4.1 ZenHub Issues Burndown Charts

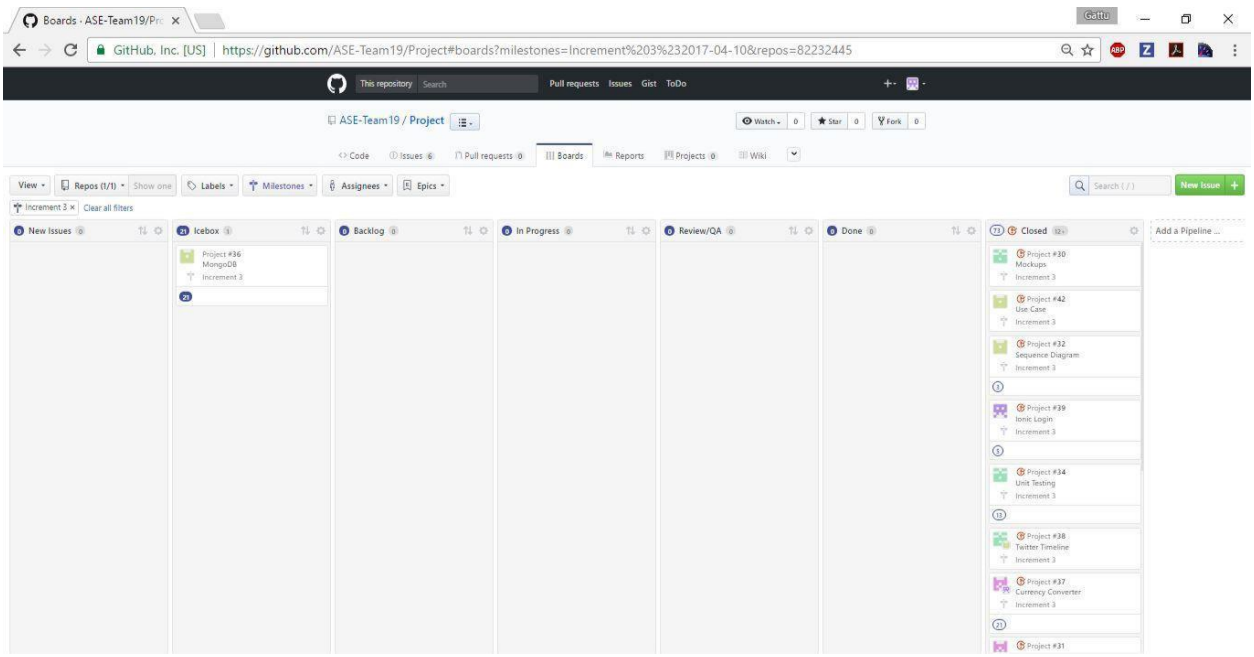
4.1.1. Project Increment 1

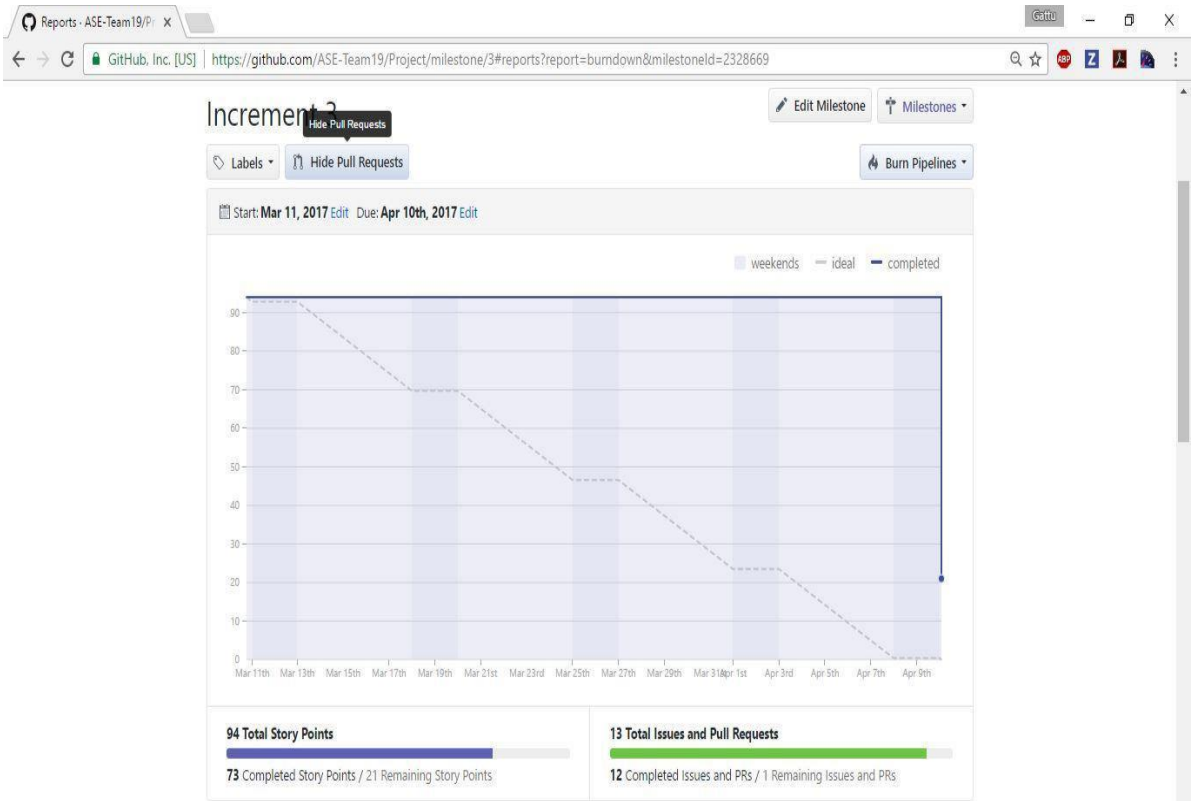


4.1.2 Project Increment 2

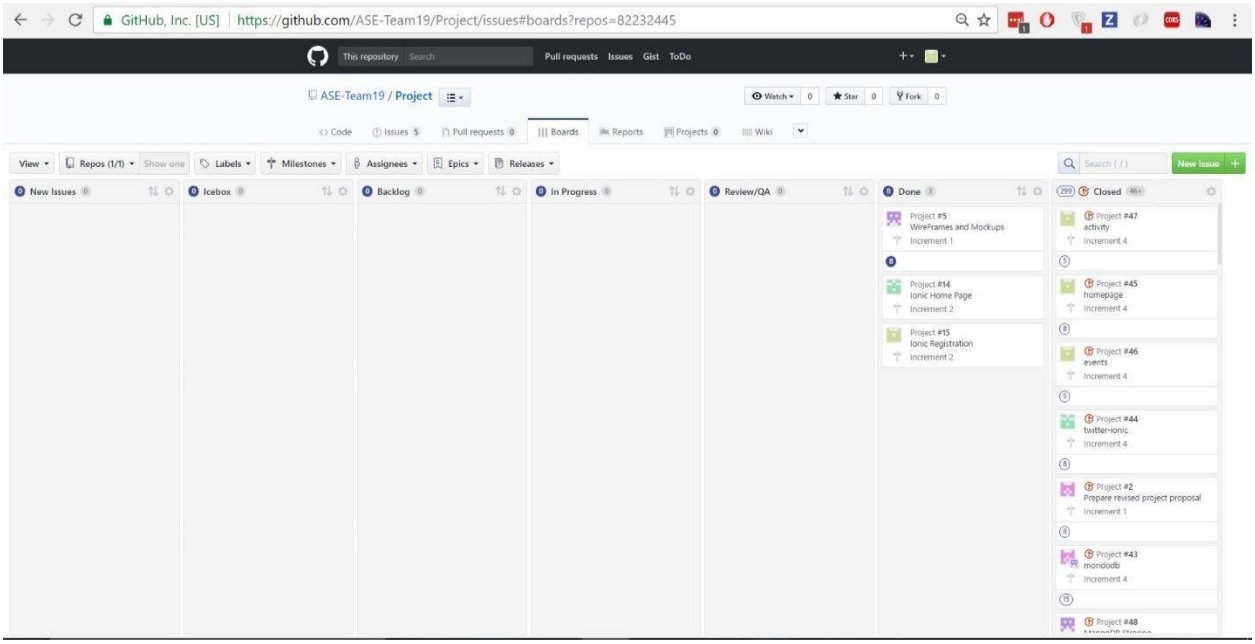


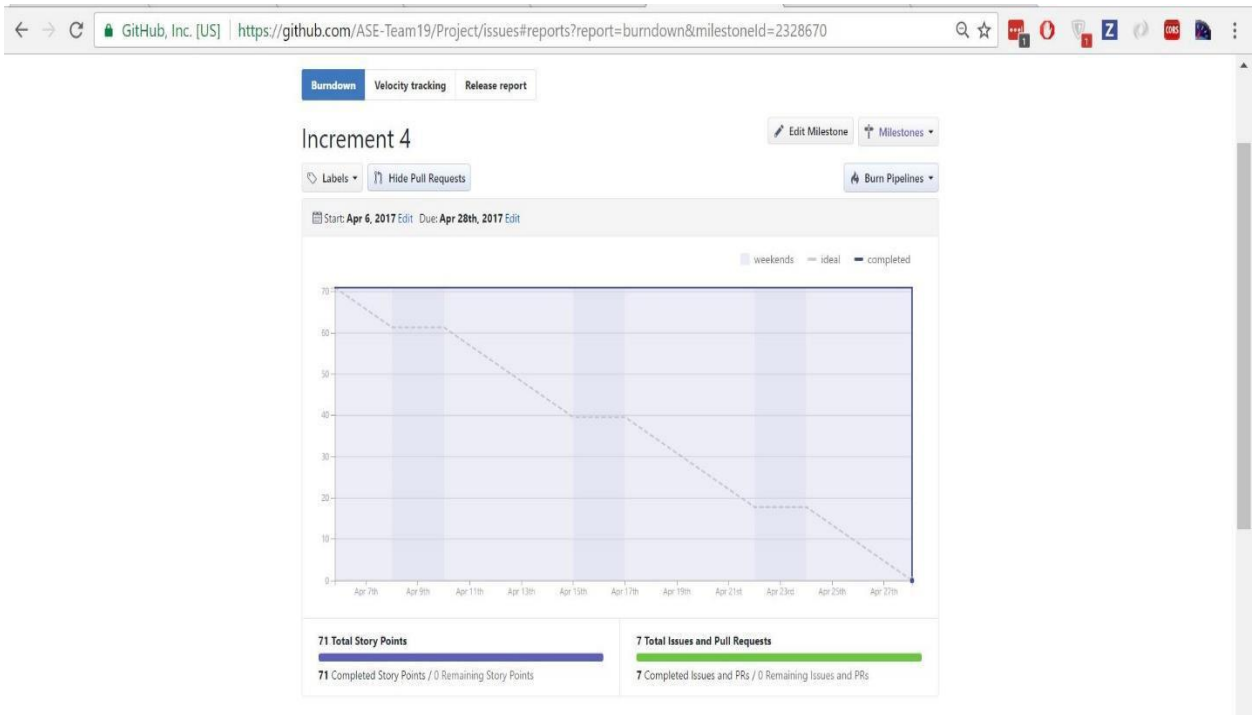
4.1.3 Increment 3





4.1.4 Increment 4





5. Technologies Used & Internal Flow of System

5.1 System Requirements

Travel Buddy is an ionic application which uses **WebStorm / Brackets** as IDE. The front end of the application is developed using **HTML, CSS, JSP** and back-end logic is handled in **Angular JS**. **MongoDB** is used to manage user details such as user registration, validating user login details, updating user details.

Operating System: Windows 7 or above

IDE: WebStorm / Brackets

Languages Used: Angular JS, HTML, CCS, JSP Ionic Framework

Databases: Firebase, MongoDB

Weather Report: Weatherunderground api

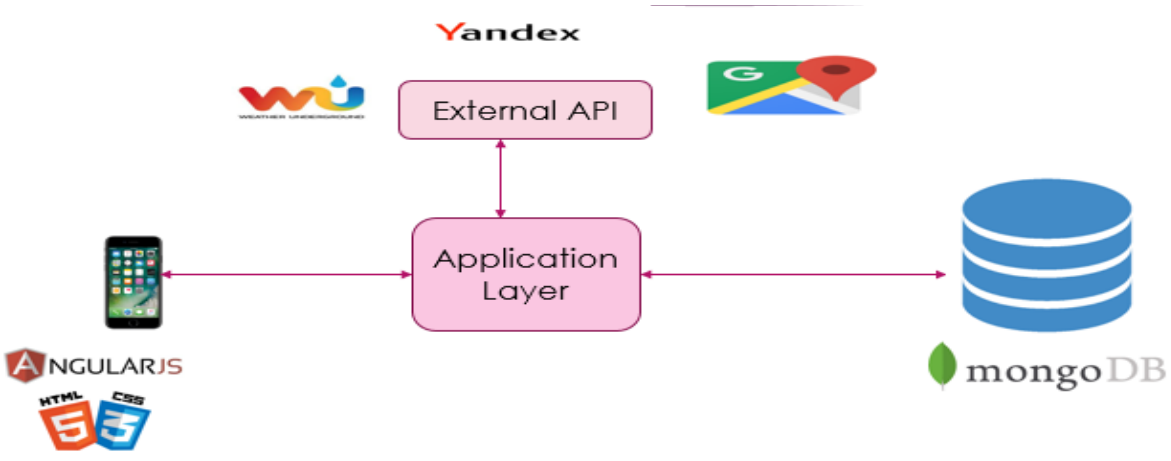
Language Translator: Yandex API

Currency Converter: Fixer api

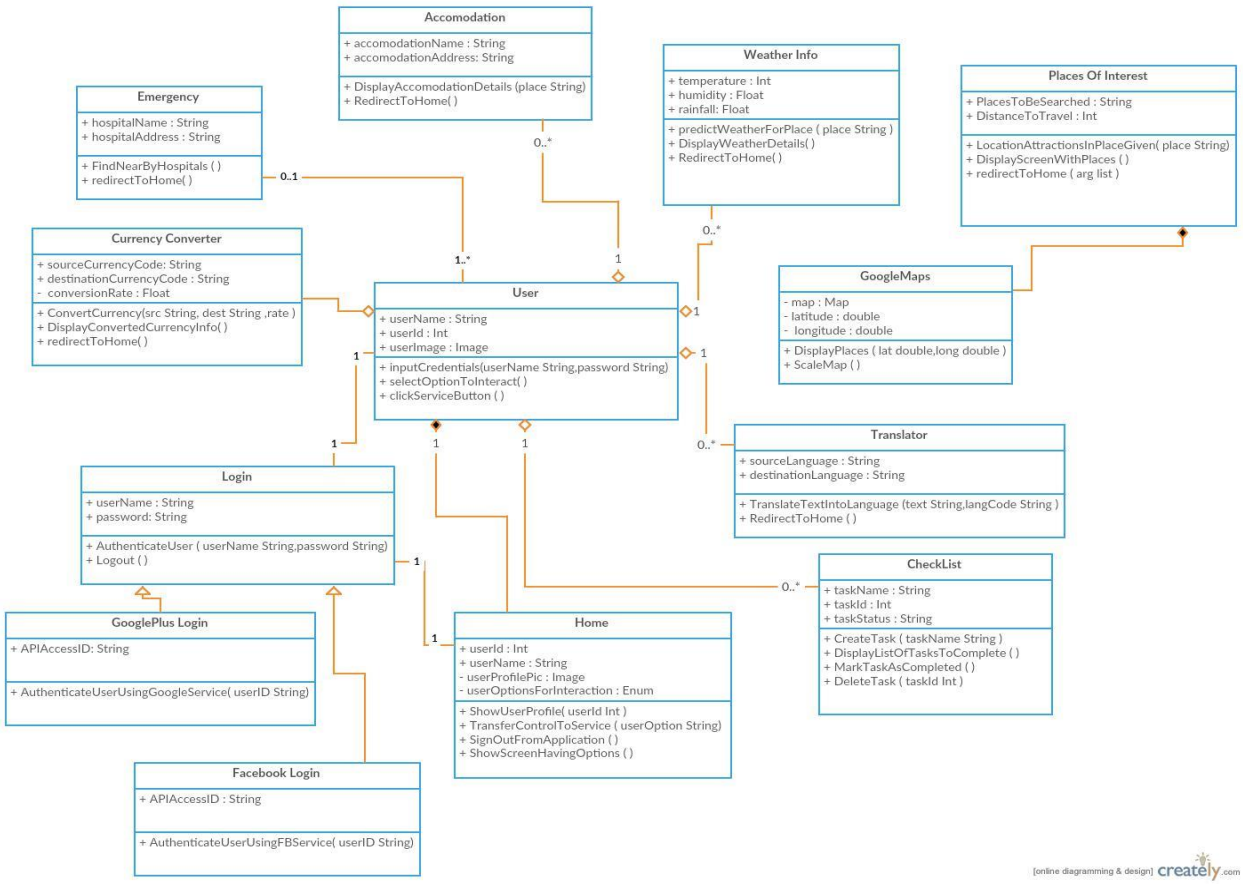
Events: Eventful api

Other geolocation: Google maps api

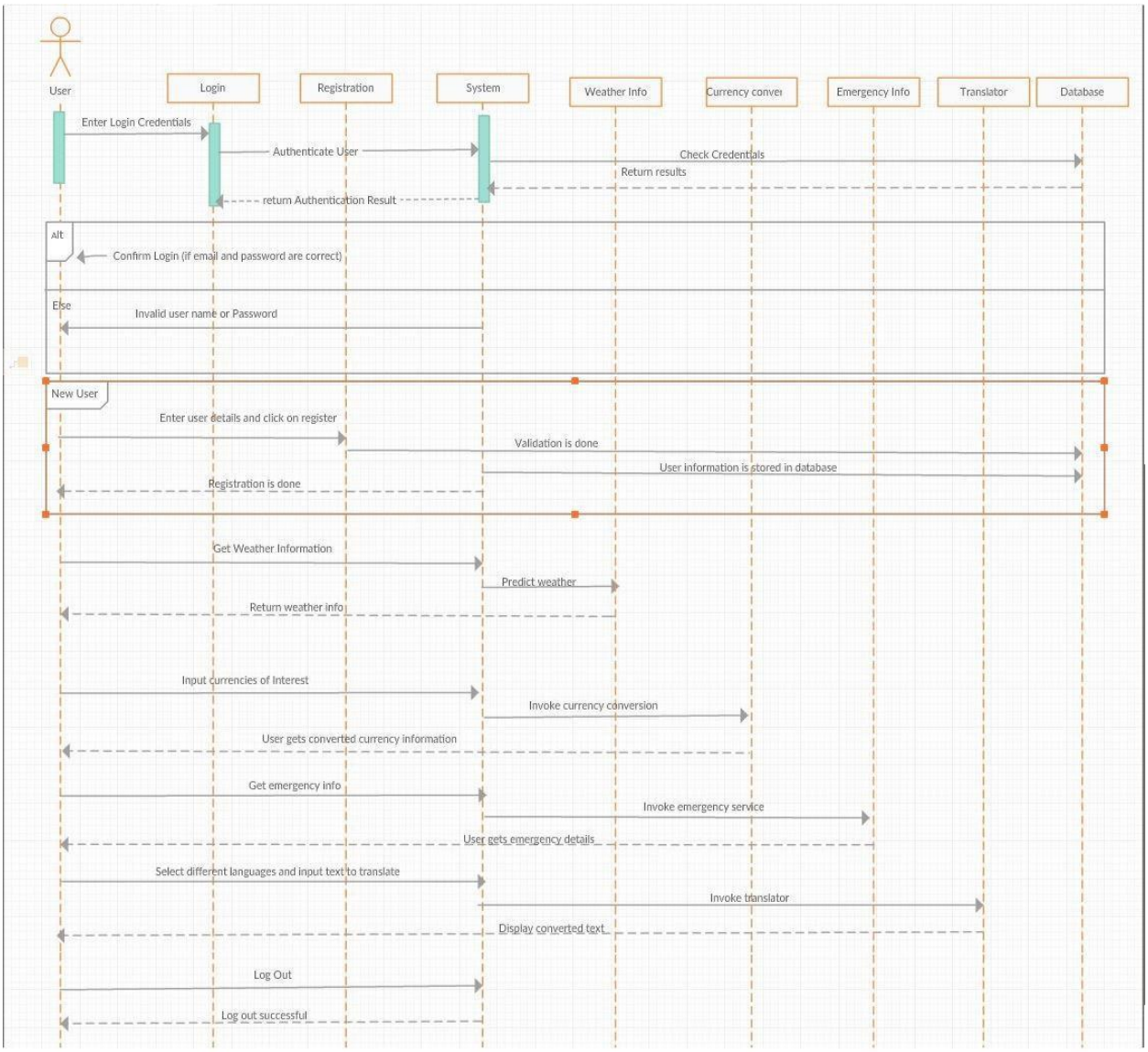
5.2 System Architecture



5.3 Class Diagram

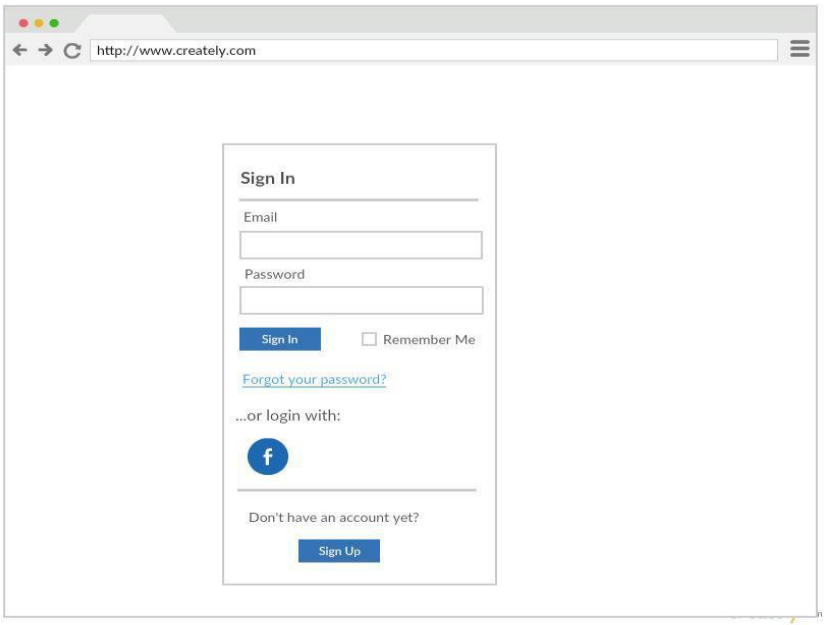


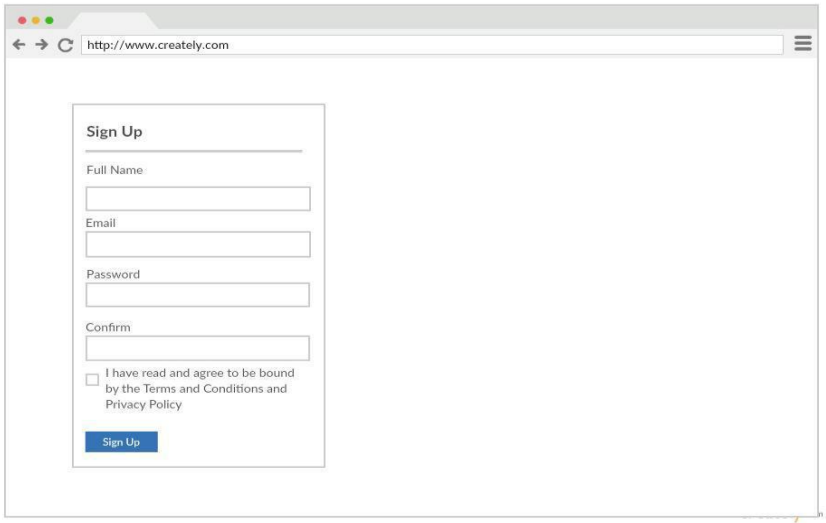
5.4 Sequence Diagram

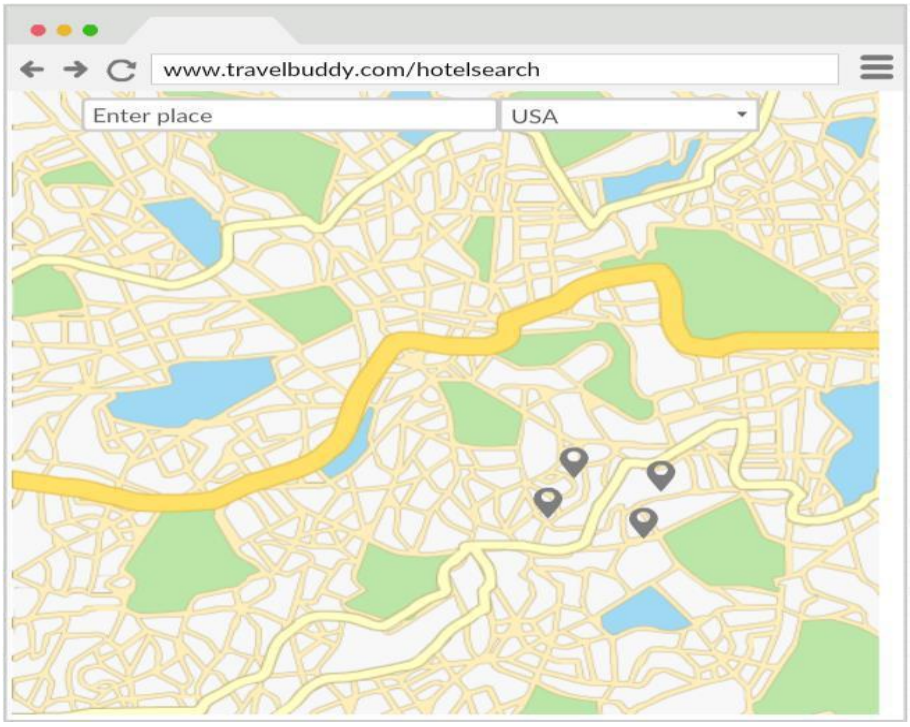


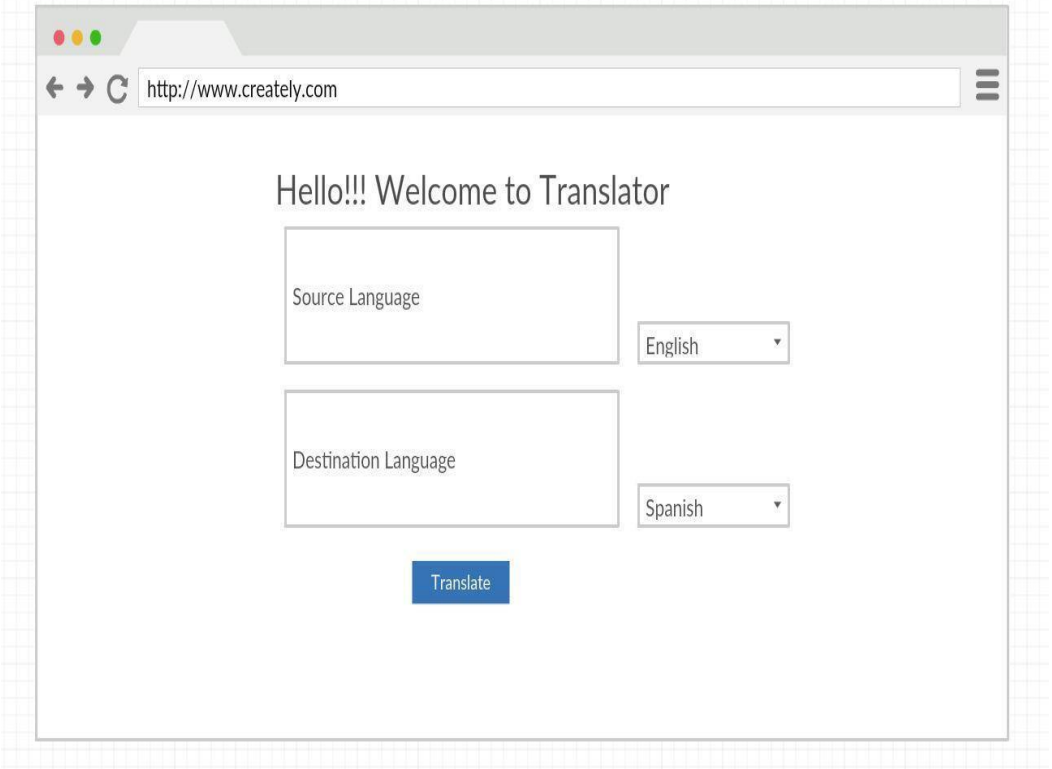
6. Detail Design of Features

6.1 Wireframes






www.travelbuddy.com/hotelsearch
Enter place
USA


http://www.creately.com
Hello!!! Welcome to Translator
Source Language
English
Destination Language
Spanish
Translate

7. How to use Travel Buddy

7.1 Registration Process

Initially new users should create account by giving User Name, Password, Confirm Password, Mobile Number, Email Id. If user has not entered all the details validation will be thrown. Also, validation is thrown if the user entered invalid email id, Mobile number and if password and confirm password doesn't match. If the user selected already registered email id, then validation will be thrown.

Registration Details Requirements:

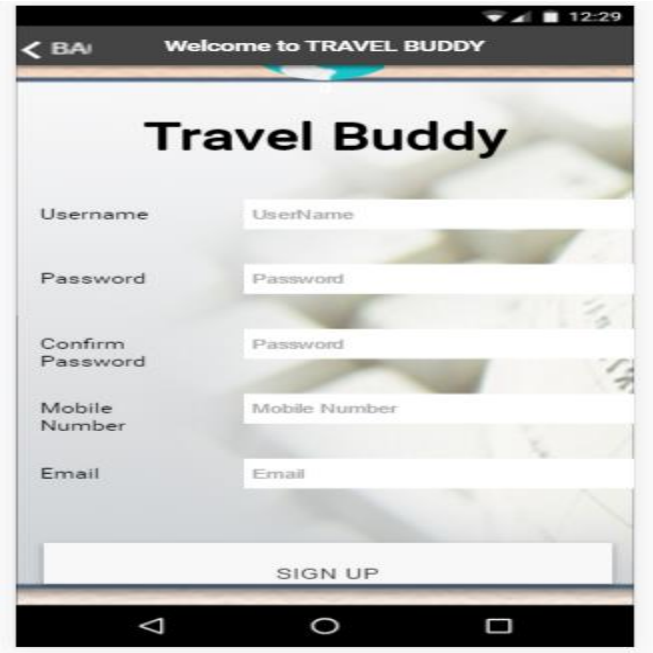
All the details are mandatory

Phone Number should be only 10 digit numeric

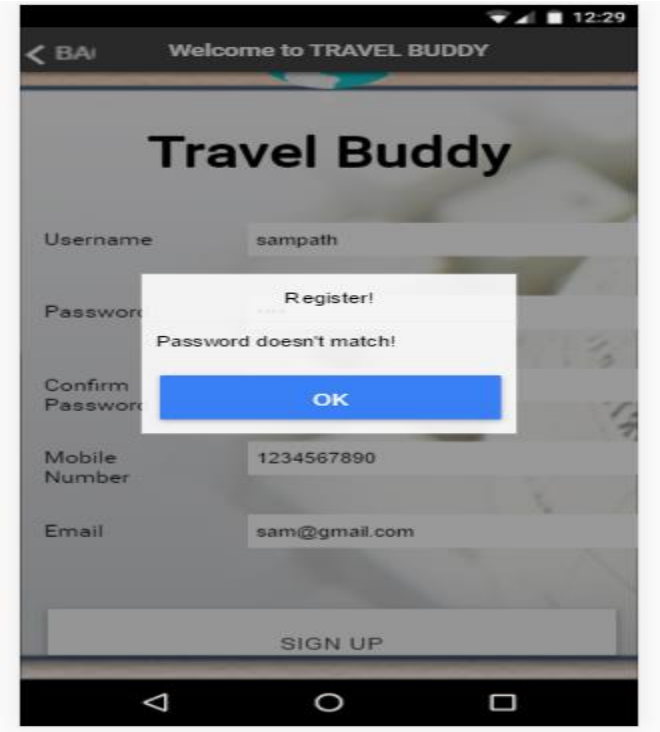
Email address should contain '@' and '.'

Password and Confirm password must match

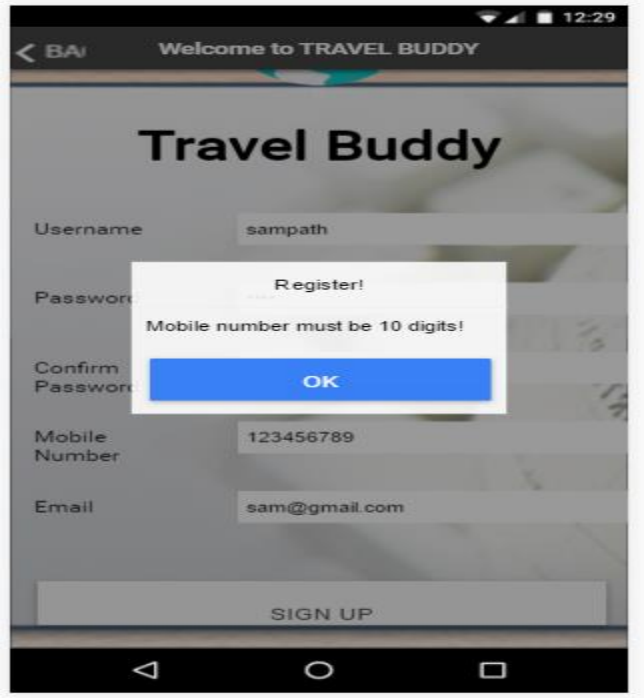
Below is the Registration page where user must enter all valid details.



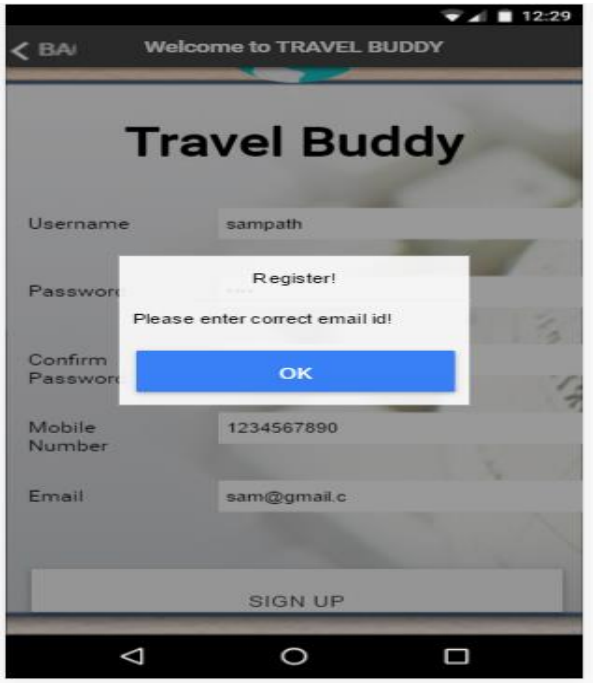
If the password and confirm password doesn't match, validation is thrown.



If the mobile number is not exactly 10 digits, validation is thrown



If the email id is not correct, validation is thrown.



The user details entered during registration is stored in MongoDB. The details are further used to validate user login.

7.2 Login Process

Once the new user is created successfully, login to the application through login page.

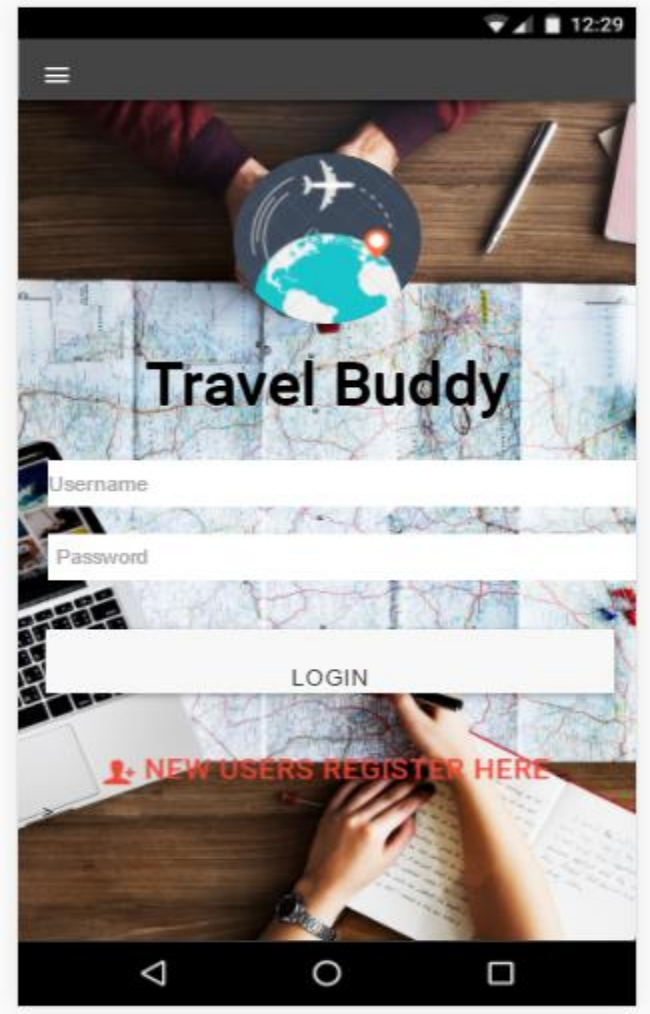
Login Page Requirements:

All the details are mandatory

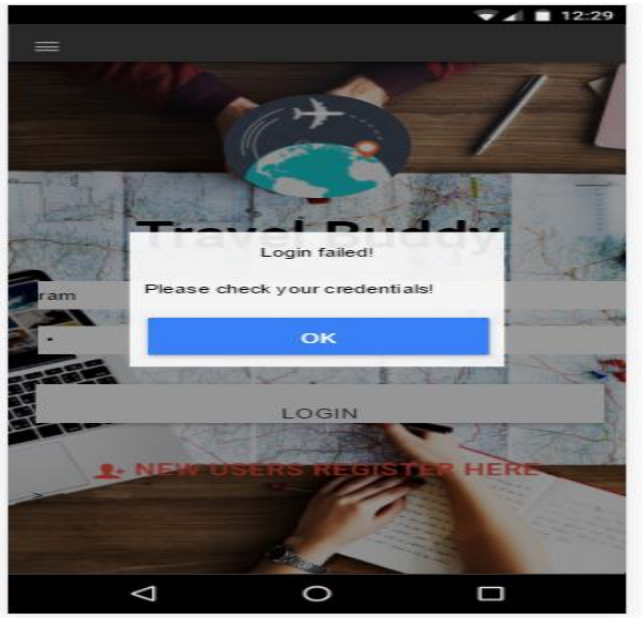
Only the registered user is valid (Details stored in MongoDB)

Password should be same as password entered during registration

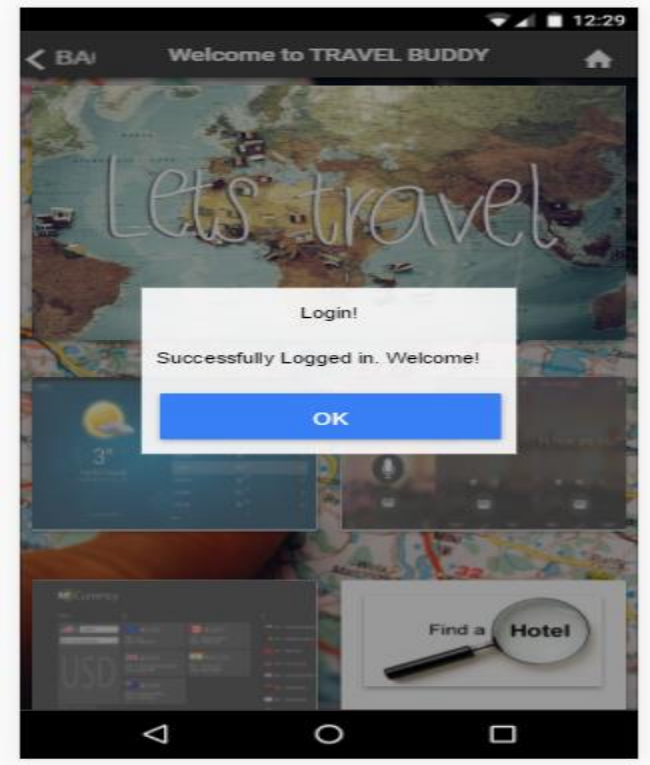
The below is the Login page where user can to login using valid user name and password.



Validation is thrown if the user name and password is not in the Mongo database



If the correct user name and password is provided, user successfully logged in the application and main page is displayed.

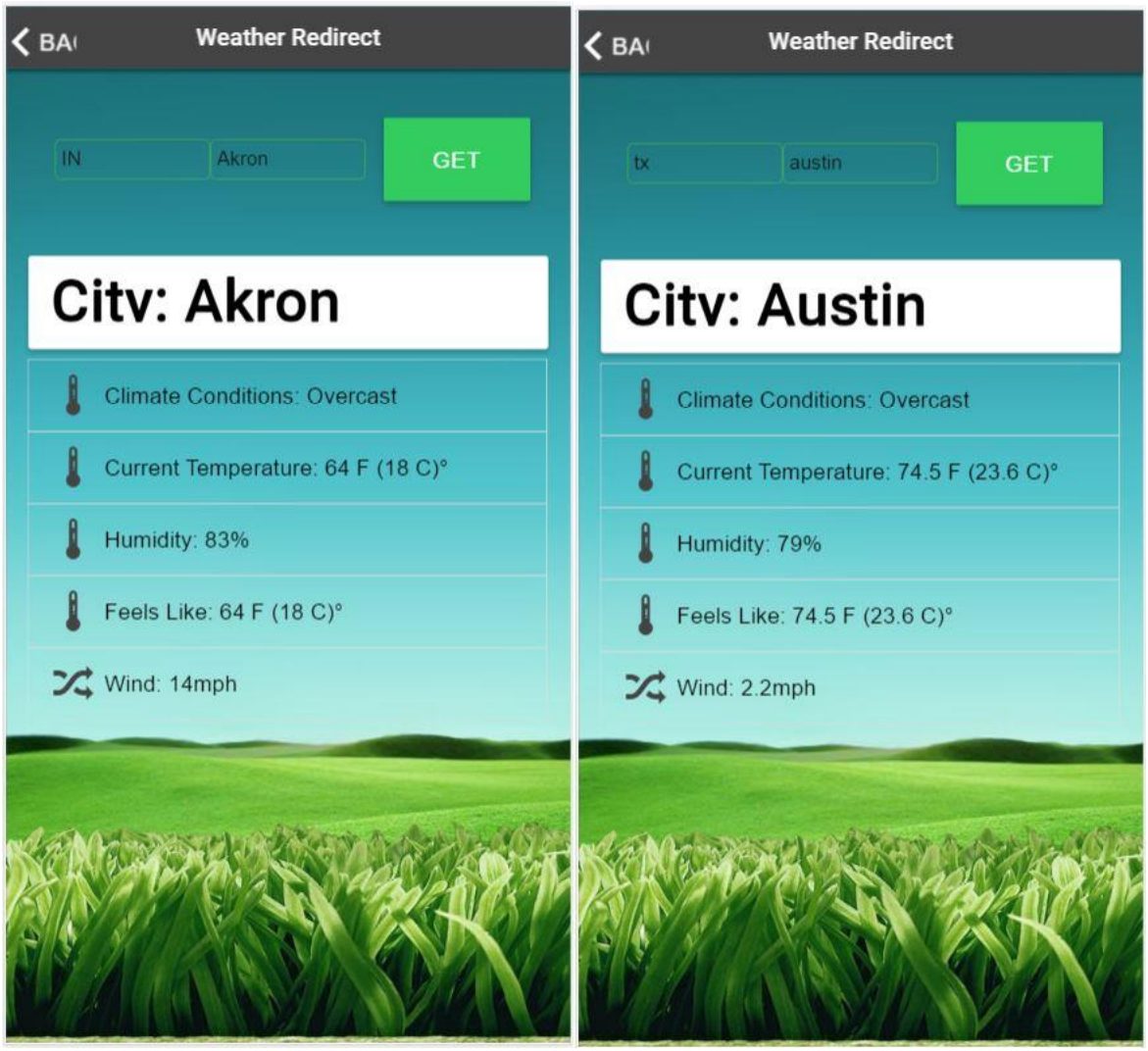


7.3 Weather Forecast

Every traveler wishes to know the weather conditions at a place beforehand so that he can plan his visit to that place. So, we have incorporated this feature into our application so that every traveler has a pleasant experience.

For this project, we have used **weatherunderground api** for retrieving the weather conditions for a place given by the user. Our weather page provides information about the climate conditions, temperature, wind speed, humidity, visibility, also a link is provided for some additional information.

APP SCREENS:



7.4 Language Translator

One of the major discomforts, one may encounter while travelling to a new place, is the local language. As the traveler might not know all the languages, talking to local people will not be an easy task. So, we have added language translator in our application, so that travelers can interact with new people beyond language barriers.

We have used Yandex API to language translation. The API gives access to the Yandex which is an online machine interpretation benefit. It underpins more than 70 dialects and can decipher isolate words or finish writings. The API makes it conceivable to insert Yandex. Interpret in a portable application or web benefit for end clients. Or, then again decipher expansive amounts of content, for example, specialized documentation.

The user first need to enter the text which he wants to translate and selects the destination language into which he wants the text to be translated. Once, he/she clicks on translate, the text is translated to corresponding language.

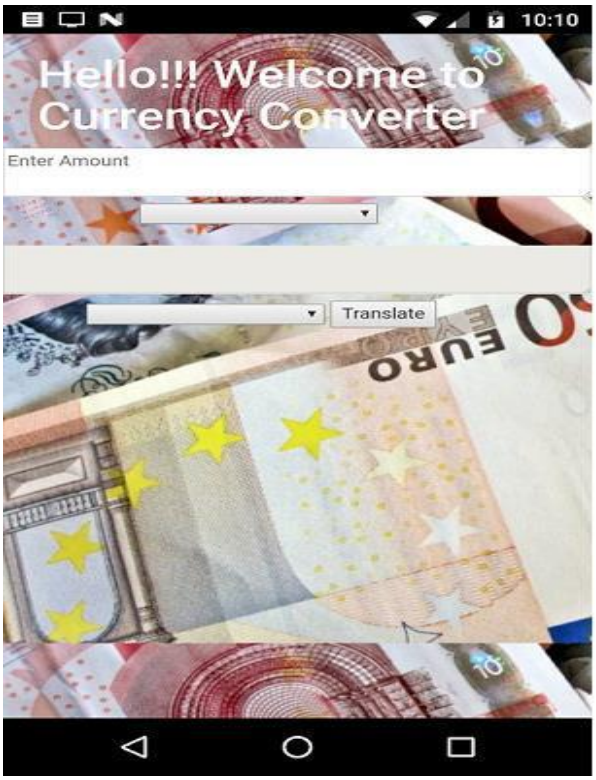
APP SCREENS:

7.5 Currency Converter

We have used pi.fixer.io to get the current Currency values base to EUR. Then we get the source and destination currency divide both and multiply with the given base number to get the destination currency value.

The user must select the source currency and must enter the amount that has to be converted into destination currency. He/she then clicks on translate button to get the required converted value.

APP SCREENS:



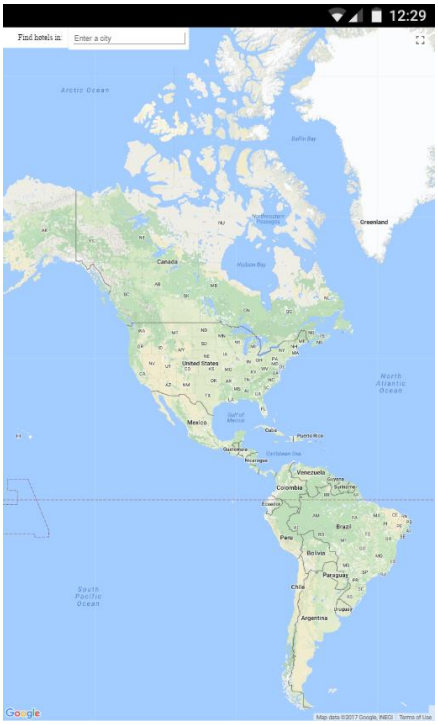
7.6 Hotel Search

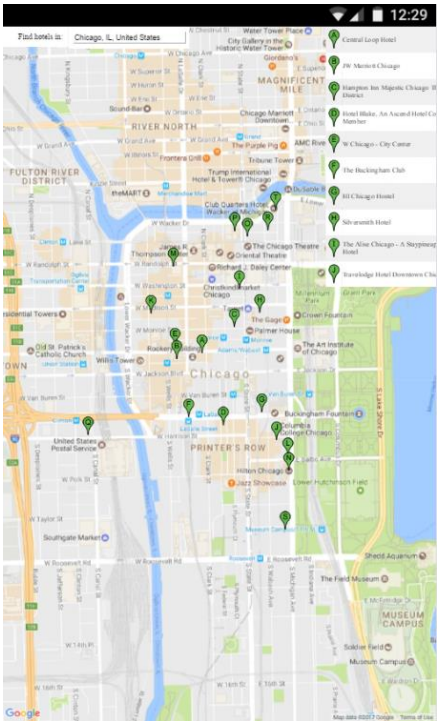
API :

We used google maps API to implement hotel search . Application has a text field with place auto complete function powered by google . we used google API to find near by search places to get details of Near by hotels .

If we navigate to hotel search in home page we can see hotel search interface .

APP SCREENS:

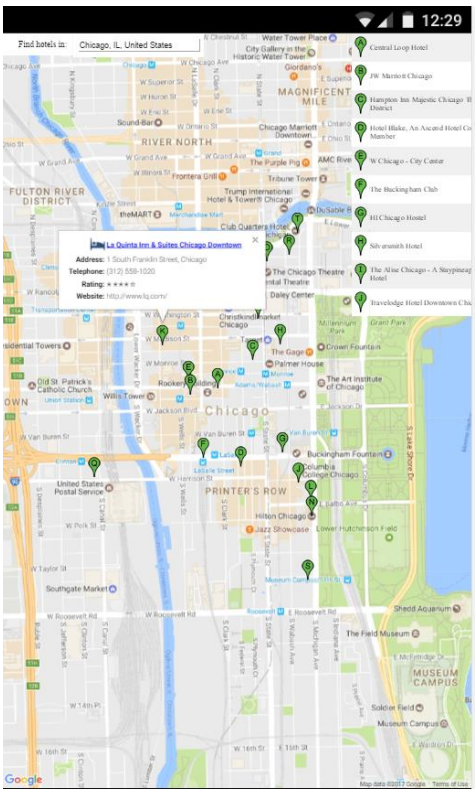


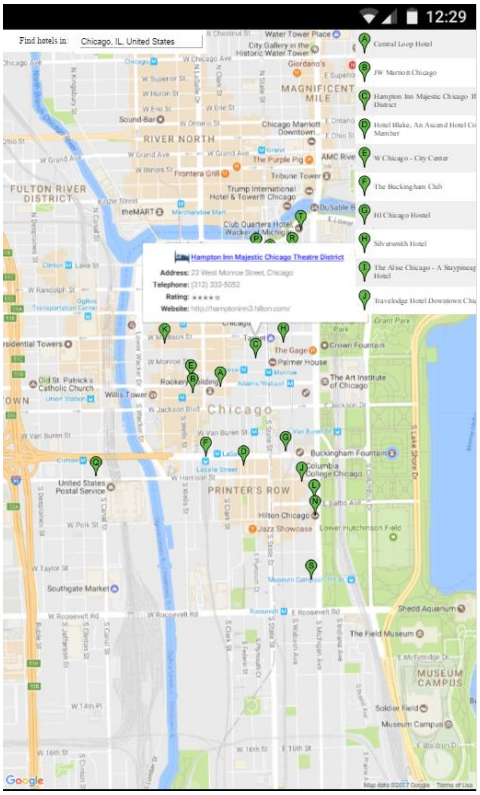


This Is hotel search page. We can search for hotel near by giving places details in the text field. We used google place search API to search for hotels.

We gave Chicago as the place and it shows the hotel near by Chicago with green marker as shown above.

Here we can see place details by clicking on marker. we can also select a hotel from window on the side. the details window will be popped up. it shows details such as rating phone address and website as shown below.

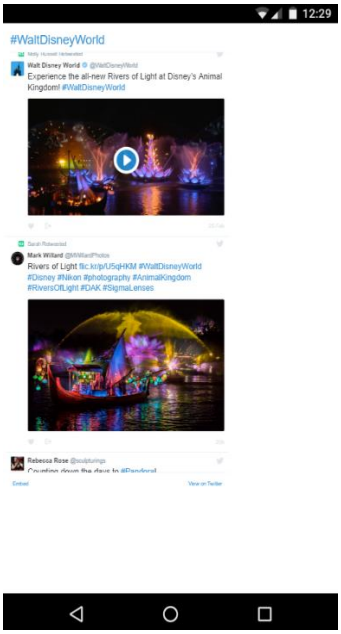




7.7 Twitter Timeline

This is the home page for twitter time line. we can choose our interest and the page is redirected to top places of interest in USA for the desired result. We have many choices like skiing, trekking, rafting, wildlife, theme parks etc. If we choose any of the above the page is redirected to page with top results of places of interest for selected category.

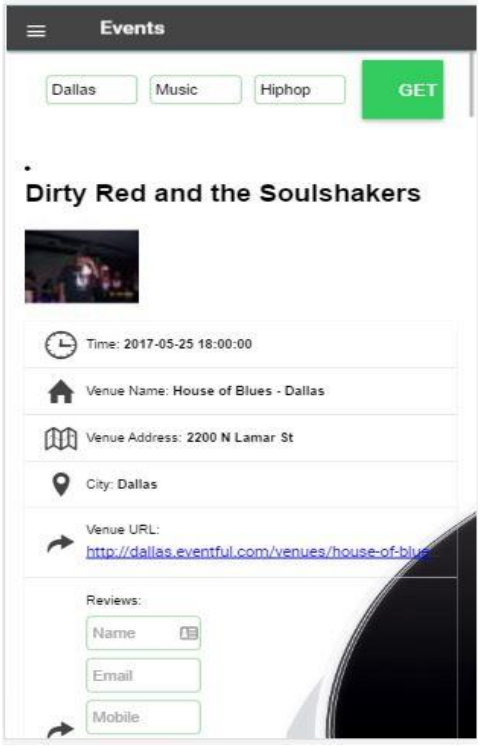
APP SCREENS:

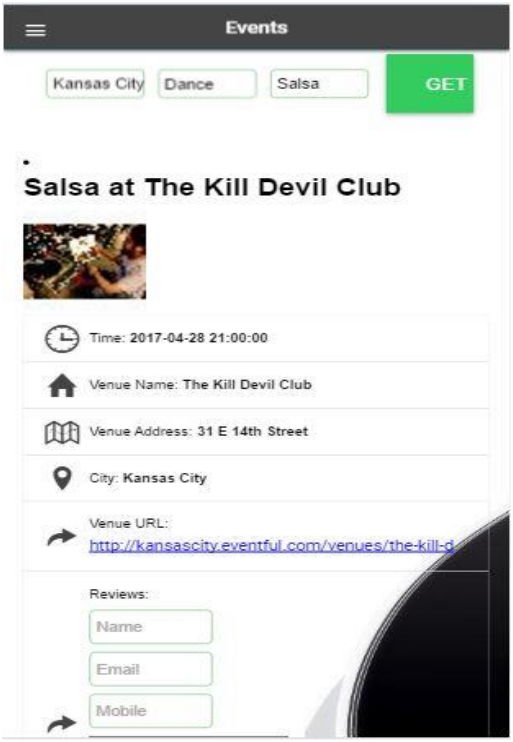


7.8 Event

When traveling to a new place, a person likes to know about the events which are taking place in his vicinity. The events page in this application, returns all the live events taking place according to the user's input. Also, a link is provided for the user to buy tickets for any desired event. The app also provides an option for the users to give reviews based on their experience at an event. The user is required to enter the details of city, category and keyword. We have used **"Eventful API"** for this purpose. This page returns information about the time, date, venue address of the events. Some of the sample screenshots of this page are shown below.

APP SCREENS:

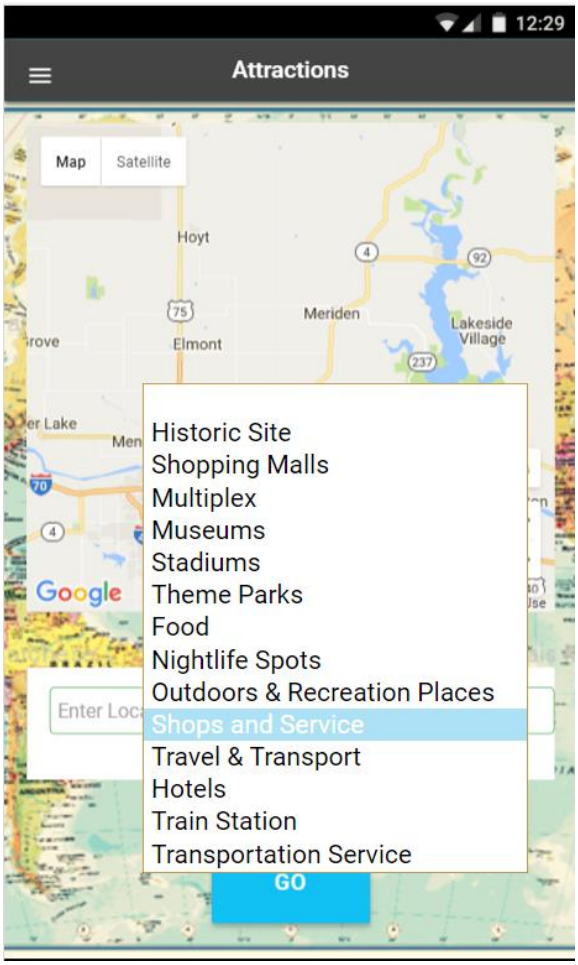


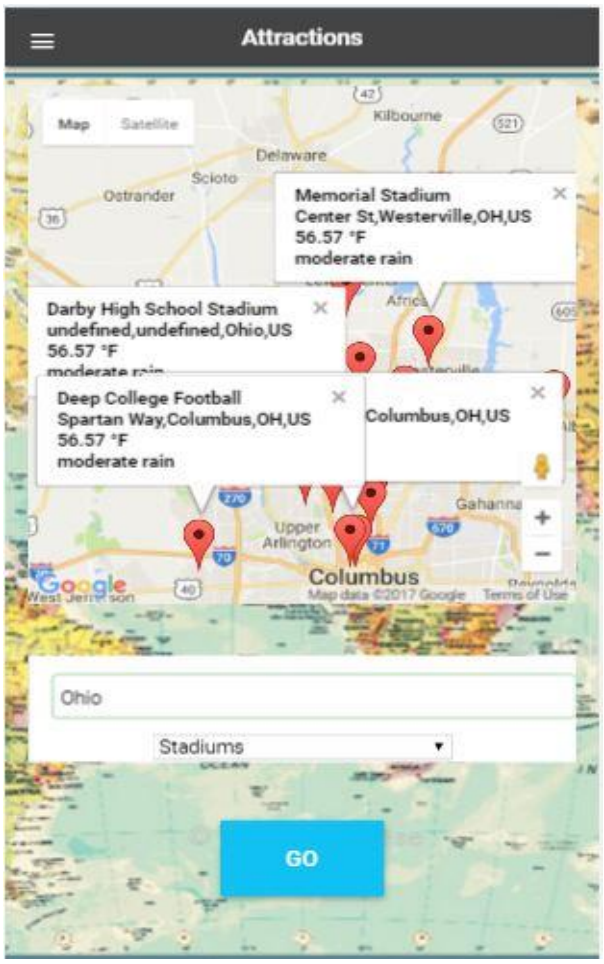


7.9 Attractions

Every traveller is enthusiastic about knowing the tourist spots around him, when he travels to a new place.The attractions page of this application serves this purpose. It shows information about location of musuems, theatres along with many other interesting locations for the users. The application also gives the address of the selected tourist spot along with it the climatic conditions at that particular location, so that he can plan his visit. Some of the sample screenshots of this page are shown below. We have used foursquare api and weather underground api for this page.

APP SCREENS:

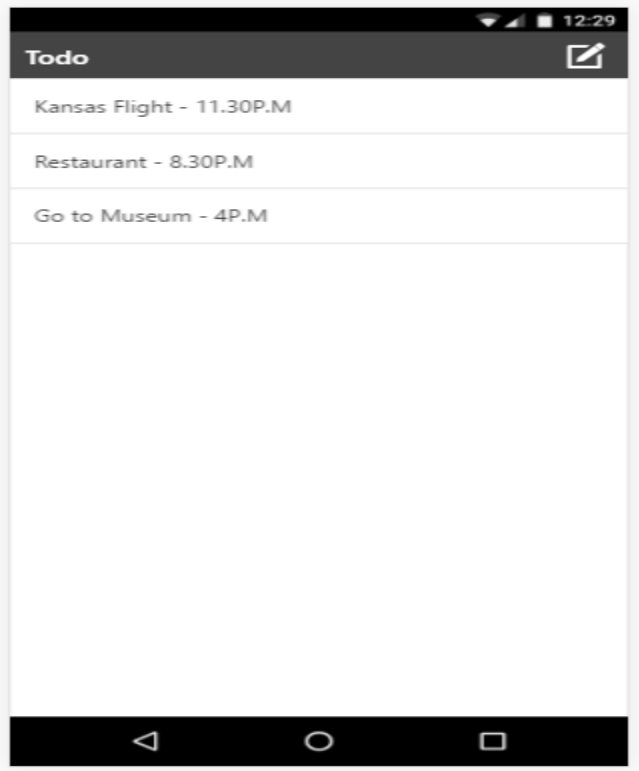




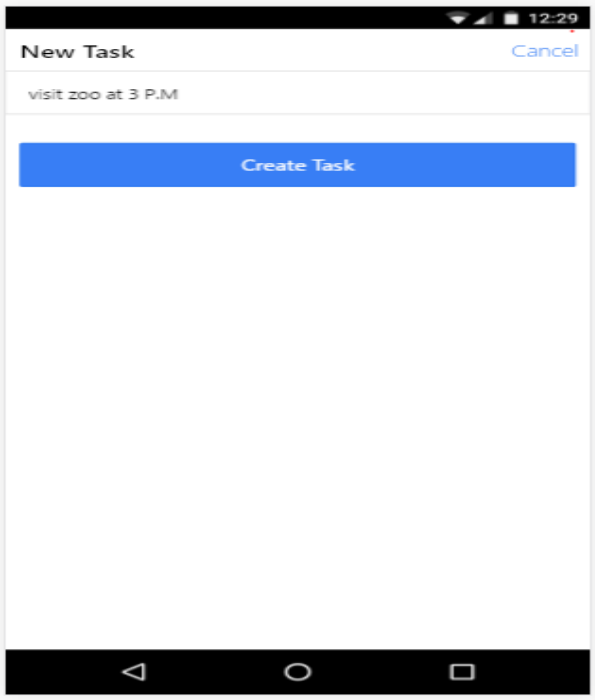
7.10 To-do list

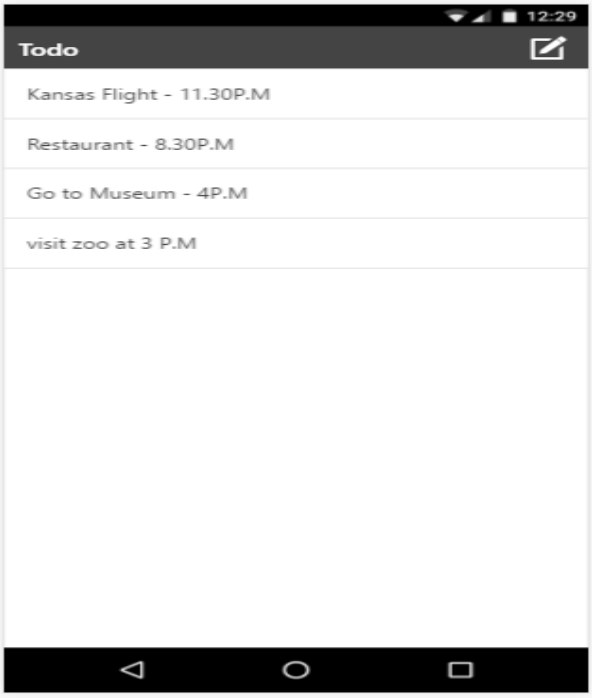
To-do list is the one features where the user can save the tasks and get notification when the deadline is near.

Below screenshot is the To-do list page, where user can see all his tasks.



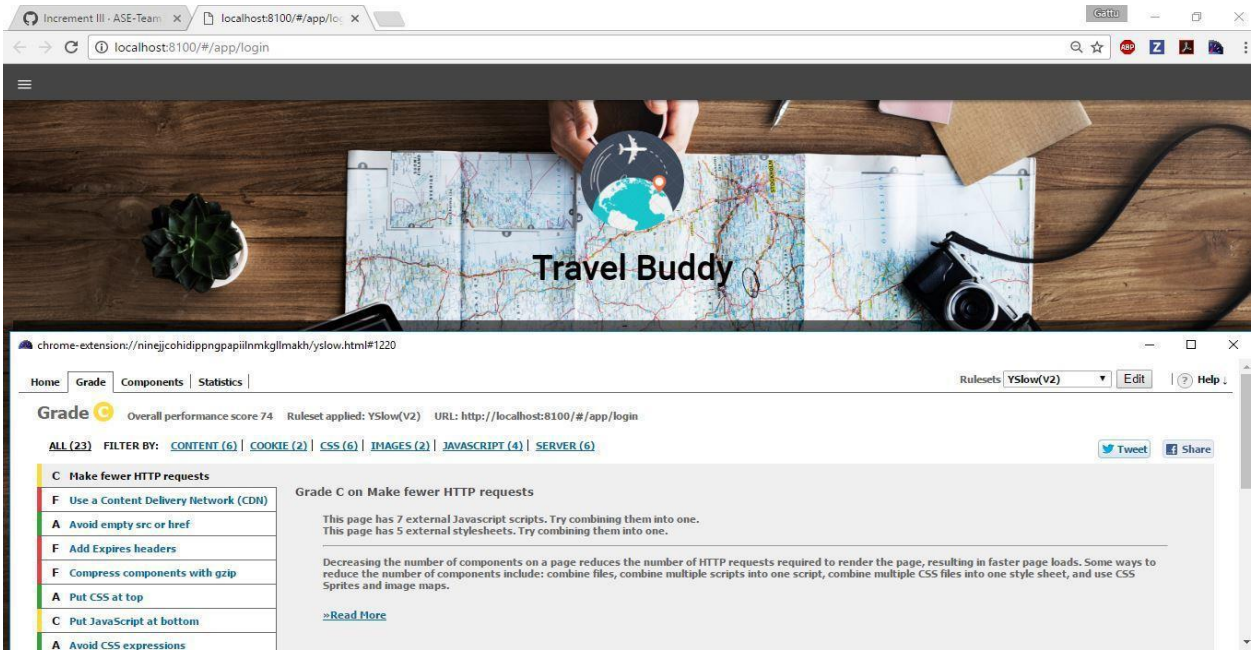
Once, the user clicks on Create task button on top-right, he/she can create a new task which will be added to the existing tasks.

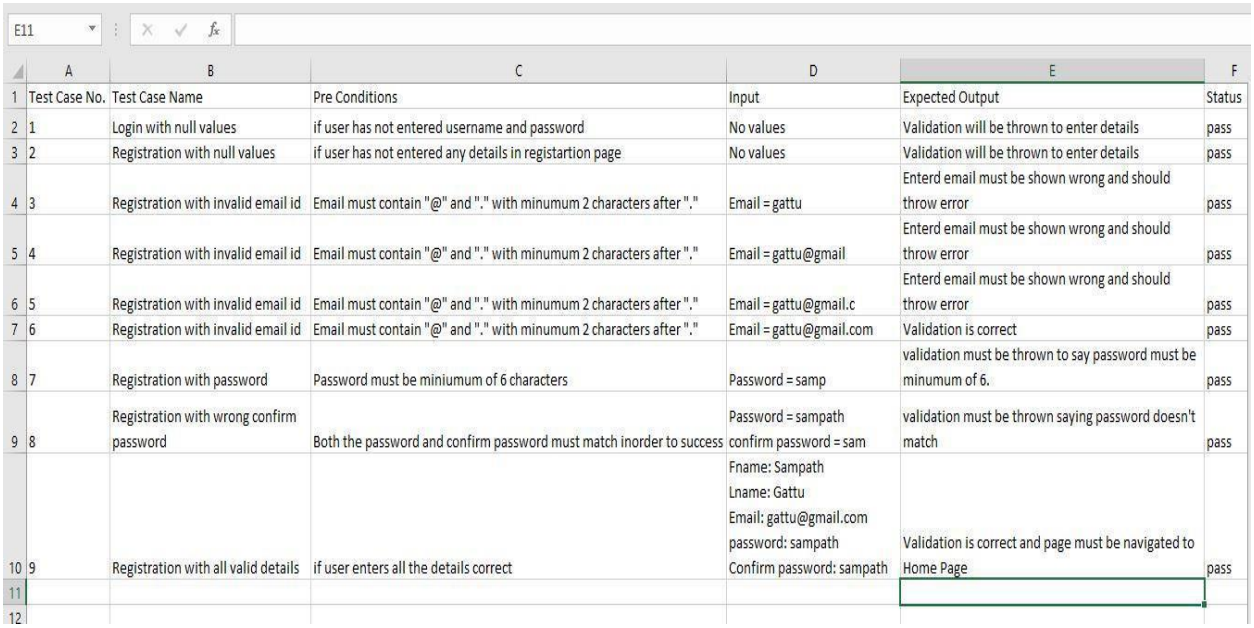




8. Testing

Page performance and ranking is checked using YSLOW analyzer





	A	B	C	D	E	F
1	Test Case No.	Test Case Name	Pre Conditions	Input	Expected Output	Status
2	1	Login with null values	if user has not entered username and password	No values	Validation will be thrown to enter details	pass
3	2	Registration with null values	if user has not entered any details in registartion page	No values	Validation will be thrown to enter details	pass
4	3	Registration with invalid email id	Email must contain "@" and "." with minumum 2 characters after "."	Email = gattu	Enterd email must be shown wrong and should throw error	pass
5	4	Registration with invalid email id	Email must contain "@" and "." with minumum 2 characters after "."	Email = gattu@gmail	Enterd email must be shown wrong and should throw error	pass
6	5	Registration with invalid email id	Email must contain "@" and "." with minumum 2 characters after "."	Email = gattu@gmail.c	Enterd email must be shown wrong and should throw error	pass
7	6	Registration with invalid email id	Email must contain "@" and "." with minumum 2 characters after "."	Email = gattu@gmail.com	Validation is correct	pass
8	7	Registration with password	Password must be miniumum of 6 characters	Password = samp	validation must be thrown to say password must be minumum of 6.	pass
9	8	Registration with wrong confirm password	Both the password and confirm password must match inorder to success	Password = sampath confirm password = sam	validation must be thrown saying password doesn't match	pass
10	9	Registration with all valid details	if user enters all the details correct	Fname: Sampath Lname: Gattu Email: gattu@gmail.com password: sampath Confirm password: sampath	Validation is correct and page must be navigated to Home Page	pass
11						
12						

9. Related Information

GITHUB	https://github.com/ASE-Team19/Project/
FIRST INCREMENT REPORT	https://github.com/ASE-Team19/Project/blob/master/Documentation/Increment1/Increment%20-1.pdf
SECOND INCREMENT REPORT	https://github.com/ASE-Team19/Project/blob/master/Documentation/Increment%20II/increment%202.pdf
THIRD INCREMENT REPORT	https://github.com/ASE-Team19/Project/blob/master/Documentation/Increment%20III/Increment-3.pdf

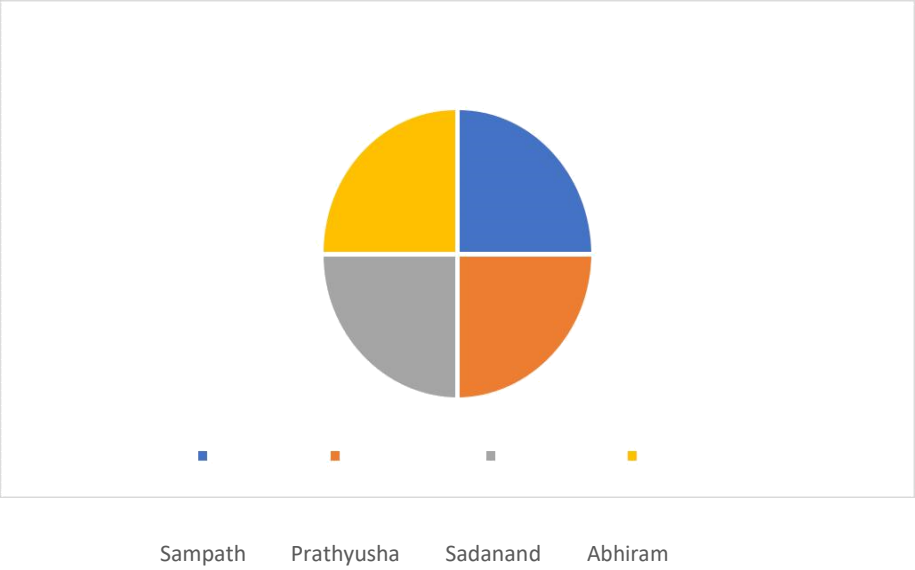
FOURTH INCREMENT REPORT	https://github.com/ASE-Team19/Project/blob/master/Documentation/increment%20IV/Increment-4.pdf
PRESENTATION	https://drive.google.com/file/d/0ByHMby6P_UfqLUhVSGhacFBjQ0k/view
YOUTUBE	https://www.youtube.com/watch?v=w7LqwtOQwGo&feature=youtu.be

10. Project Management

10.1 Team Members Contribution

S.No	Team member name	Contribution	Status
1	Sadanand Kallakuri	Weather Forecast, Events Search, Attractions, IONIC Home page, UI for respective pages, Testing for respective modules, Documentation.	Completed
2	Abhiram Reddy Nalla	Twitter timeline, Hotel Search, IONIC Registration, UI for respective pages, Testing for respective modules, Documentation.	Completed

3	Prathyusha Pottabathula	Currency Converter, UI for respective pages, Data validations in login and registration, Testing for respective modules, Documentation.	Completed
4	Sampath Gattu	Language Translator, MongoDB Implementation, IONIC main Page, UI for respective pages, Testing for respective modules, Documentation.	Completed



10.2 Final Project Evaluation

The project IM Word is successfully completed in the scheduled plan (as shown on Project Plan). We had good interaction with each other and always maintained one to one communication to maintain same understanding of entire project. Following Agile process model helped us to add few changes in the middle of project without facing any major milestone. Agile Process requires less documentation and more interactive between team members.

The work is equally divided between the team members so that we could able to meet the prescribed timelines without any delay. Testing each and every module efficiently helped us to reduce the errors during final integration of the project. Because of this we could reach all the goals and developed the project as proposed in the Project Proposal.

The success of every project involves in good collaboration with team members, maintaining quality of project, having good understanding of requirements, testing of every module efficiently, identifying blockers early and trying to solve them immediately.

11. Future Work

We can add an innovative intelligent option for the users which plans his itinerary given a start and end date. Along with the details of when to visit an particular tourist spot, so that the traveler can get the best experience.

Also, we can incorporate location based emergency services, such as providing feature for the user to call emergency services such as police, ambulance etc when in distress.

Taxi services can also be incorporated in this application, so that whenever his car breaks down, he can request a cab.

12. Bibliography

1. https://creately.com
2. https://w3layouts.com/
3. https://startbootstrap.com/
4. http://fixer.io/
5. www.w3schools.com
6. https://www.tutorialspoint.com/ionic/ionic_environment_setup.htm
7. https://startbootstrap.com/
8. www.Yandex.com
9. https://openweathermap.org/
10. https://www.wunderground.com/weather/api/
11. https://developer.foursquare.com/
12. http://api.eventful.com/