A Major Project Report on

Futsal Reservation System

Submitted in Partial Fulfillment of the Requirements for

the Degree of **Bachelor of Engineering in Computer engineering**

Under Pokhara University

Submitted by:

**Anish B.K., 15350**

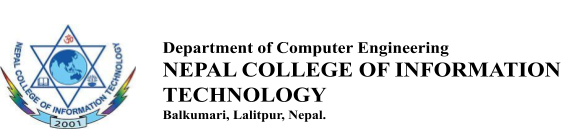
**Nabin Sinkhwal, 15392**

**Bishal Shrestha, 15357**

**Aswin Dhakal, 15353**

Date:

**28November 2019**



**Acknowledgement**

Without the common efforts of many individuals, this project would not have been possible. It has been a pleasure for us to acknowledge the assistance and contributions that were very important and supportive throughout the project. We would like to extend our sincere thanks to all of them. We owe special thanks to a number of people who has devoted much of their time and expertise without which it would have been very difficult for us to complete our project entitled Futsal Reservation System.

We are highly indebted to our Project Supervisor Er.Birendra Bista, for his valuable guidance throughout the project development period and for providing technical support with suggestions which helped our project to grow and foster to a certain level we didn’t think of reaching in such a short period.

Last, but not the least, we would like to thank our teachers and colleagues who have been knowingly or unknowingly the part of this project and lent support and views during the entire development time.

**Abstract**

The project aim is to make the Futsal booking convenient by providing an electronic platform i.e. introducing e-commerce in this field. Many developed countries believe that it is the backbone of development. So, we chose our project related to e-commerce in which booking and obtaining related information through internet. The Futsal administration can handle and manage a website to view the clients request and confirm those requests. It will be also be useful for administration to provide details like upcoming events, update and other related services to the visitors. The aim of our project is to provide online booking facility to the registered members via website.

The development is carried out with the agile principles following the scrum framework of agile methodology. The project will complete in major 4 sprints and will extend up to one and half month.

Contents

[**1.** **Introduction** 1](#_Toc25837769)

[1.1. Problem Statement 2](#_Toc25837770)

[1.2. Project objectives 2](#_Toc25837771)

[1.3. Project scope and limitations 2](#_Toc25837772)

[1.4. Significance of the study 3](#_Toc25837773)

[**2.** **Literature Review** 4](#_Toc25837774)

[**2.1.** **PHP: Hypertext Preprocessor** 4](#_Toc25837775)

[2.2. Existing systems like Futsal Reservation System 5](#_Toc25837776)

[2.3. Google Maps API 6](#_Toc25837777)

[2.4. Solutions Offered to The Existing Deficiencies 7](#_Toc25837778)

[**3.** **Team Members and Divided Roles** 8](#_Toc25837779)

[**4.** **Methodology** 9](#_Toc25837780)

[4.1. Software Development Life Cycle 9](#_Toc25837781)

[**5.** **Performance Analysis** 11](#_Toc25837782)

[**6.** **Achieved Deliverables** 12](#_Toc25837783)

[**7.** **Requirement Analysis** 13](#_Toc25837784)

[7.1. Functional Requirements 13](#_Toc25837785)

[7.2. Non-Functional Requirements 14](#_Toc25837786)

[7.3. Input Requirements 15](#_Toc25837787)

[7.4. Output Requirements 15](#_Toc25837788)

[7.5. Security Requirements 15](#_Toc25837789)

[**8.** **System Design and UML Models** 16](#_Toc25837790)

[8.1. System Architecture 16](#_Toc25837791)

[8.2. ER Diagram 17](#_Toc25837792)

[8.3. Use Case Diagram 18](#_Toc25837793)

[8.4. Class Diagram 19](#_Toc25837794)

[8.5. Data Flow Diagram 20](#_Toc25837795)

[8.6. Context Diagram 21](#_Toc25837796)

[8.7. Sequence Diagram 22](#_Toc25837797)

[8.7.1. Sequence Diagram for Register 22](#_Toc25837798)

[8.7.2. Sequence diagram of Login 23](#_Toc25837799)

[8.7.3. Sequence Diagram of Logout 23](#_Toc25837800)

[8.7.4. Sequence Diagram of Event 24](#_Toc25837801)

[9. Project task and time schedule 25](#_Toc25837802)

[**10.** **Testing** 26](#_Toc25837803)

[10.1. Testing Table 26](#_Toc25837804)

[10.2. Test Evidences 28](#_Toc25837805)

[Test 1: 28](#_Toc25837806)

[Test 2: 29](#_Toc25837807)

[Test 3: 30](#_Toc25837808)

[Test 4: 31](#_Toc25837809)

[**11.** **Conclusion and Future Extensions** 32](#_Toc25837810)

[**12.** **References** 33](#_Toc25837811)

[Figure 1 Scrum (Sprint Periods) 11](#_Toc25575298)

[Figure 2 Agile Testing 14](#_Toc25575299)

[Figure 3 Functional Requirements 16](#_Toc25575300)

[Figure 4 Non-Functional Requirements 17](#_Toc25575301)

[Figure 5 System architecture 19](#_Toc25575302)

[Figure 6 ER Diagram 20](#_Toc25575303)

[Figure 7 Use Case Diagram 21](#_Toc25575304)

[Figure 8 Class Diagram 22](#_Toc25575305)

[Figure 9 Data Flow Diagram 23](#_Toc25575306)

[Figure 10 Context Diagram 24](#_Toc25575307)

[Figure 11 Sequence Diagram For Register 25](#_Toc25575308)

[Figure 12 Sequence Diagram of Login 26](#_Toc25575309)

[Figure 13 Sequence Diagram of LogOut 26](#_Toc25575310)

[Figure 14 Sequence Diagram of Event 27](#_Toc25575311)

[Figure 15 Gantt chart 28](#_Toc25575312)

[Figure 16 Testing Table 30](#_Toc25575313)

[Figure 17 SignUp Test Evidence 31](#_Toc25575314)

[Figure 18 SignIn Test Evidence 32](#_Toc25575315)

[Figure 19 Map Test Evidence 33](#_Toc25575316)

[Figure 20 Update Event Test Evidence 34](#_Toc25575317)

[Figure 21 Sign Up 37](#_Toc25575318)

[Figure 22 Login and Sign Up in web 38](#_Toc25575319)

[Figure 23 Login In Android 38](#_Toc25575320)

[Figure 24 Update Events in Android 39](#_Toc25575321)

[Figure 25 Update events in Web 40](#_Toc25575322)

[Figure 26 Events location in Map (android) 41](#_Toc25575323)

[Figure 27 Home Page (Android) 42](#_Toc25575324)

[Figure 28 User Profile 43](#_Toc25575325)

[Figure 29 Event Detail 44](#_Toc25575326)

[Figure 30 Profile Detail Fill Up 45](#_Toc25575327)

**LIST OF ABBREVIATIONS**

DFD Data Flow Diagram

ER Entity Relationship

FP Function Point

HTML Hyper Text Markup Language

XML eXtensible Markup Language

UML Unified Modeling Language

SQL Structured Query Language

# **Introduction**

Futsal Reservation System is the web application for registering a futsal arena and getting the booking notification whenever a booking has been made by the user. Users, on the other hand, should also make themselves and seek through the futsal and make the bookings.

There will be two types of accounts, customer and arena. Both of the users are verified by their email addresses. Futsal companies show their available timing. The timing is visible to the customers and they select desired time and book the arena for that time

Futsal Companies are given a dash board where they can update their informations anytime they want. Also they can list the offers if there are any.

Recommendation for the companies to provide for discounts are made on the basis of no booking for the futsals using the training algorithm.

## Problem Statement

We came up with this project idea to somehow eliminate the problem we are facing in this reserving a futsal court. Here are the problems that arise and can be reduced to a certain extent.

Many futsal are not being used due to lack of information about those futsals. Whereas many users cannot find their desired futsal on the desired time. For this, this system provides a platform for user as well as futsal companies for the booking to be made.

## Project objectives

The major objectives of this project are

* **Ease the procedure of futsal booking.**

This system helps both the user and the futsal company to ease the booking procedure and keeping the track of the bookings.

* **Saving the booking records in database**

The system will store all the bookings made by the user and to which futsal the booking is made.

## Project scope and limitations

The scope of the system is extended to all the internet users all over Nepal who are interested in booking futsals and to all those companies who possess a futsal court. The scopes are extended to,

* Perfect platform arena booking.
* Connect Users and Futsal Companies together in one convenient place.
* Local advertisement of the event.

The limitations that this system carry is:

* The events are not controlled by the admin panel, so the false information cannot be filtered by us. So non authentic information can flow or include in the platform.
* Proper Recommendation System

## Significance of the study

The project is determined to be beneficial for all the futsal companies and related partners. It was found to be useful to take into the implementation. It is believed to provide the business value by the project. It also implements the different part of the engineering ranging from databases to API.

# **Literature Review**

This section consists the literature study on different topics like Php. It aims to provide readers a theoretical base for the project and also develop an understanding of the nature of the project.

## **PHP: Hypertext Preprocessor**

PHP: Hypertext Preprocessor (or simply PHP) is a general-purpose programming language originally designed for web development. It was originally created by Rasmus Lerdorf in 1994; the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive initialism PHP: Hypertext Preprocessor.

PHP code may be executed with a command line interface (CLI), embedded into HTML code, or used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in a web server or as a Common Gateway Interface (CGI) executable. The web server outputs the results of the interpreted and executed PHP code, which may be any type of data, such as generated HTML code or binary image data. PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control.

## Existing systems like Futsal Reservation System

Previously, MountDigit Technology has developed a dynamic and interactive website for Futsal Arena, Thamel. In this website, they mainly emphasized on online booking of a pitch (rates, available shifts, etc.), information about their services (trainings and tournaments) and their mini sports store (moreover like an informative site of the store). Team registrations, viewing tie-sheets for a tournament is also possible through the website. Besides these, there are informative web pages which contain information, news, gallery and contact information regarding the Futsal Arena.

A similar type of website has been designed for Zion Futsal Club, Singapore. Like the previously mentioned one, this website also includes a wide range of information about the Futsal Club along with weather forecast and advertisements from third parties. They also provide the online shopping facility in their own sports materials store (includes online purchase via master cards).

**PROS**

* User and Arena registration easy and hassle free.
* Both user and futsal company is notified about the booking.
* Futsal company can make the changes in its required field.

**CONS**

* Not everyone is connected with the internet.
* Fake futsals may register by a fake email.
* There's no guarantee that the commitment a user makes to attend by clicking 'book' will be there on time.

## Google Maps API

Google Maps API was launched by Google in June 2005. By using Google Maps API, it is possible to embed Google Maps site into an external website, on to which site-specific data can be overlaid. Google API allow for developer to integrate Google Maps into their websites. The Google Maps API is free for commercial use, provided that the site on which it is being used is publicly accessible and does not charge for access, and is n ot generating more than 25000 map accesses a day.

## Solutions Offered to The Existing Deficiencies

EVENTer is looking forward to bring all the events that are being hosted around any location to make searching events easy and intuitive. Interactive maps let attendees zoom in and out, tap on pins for exhibitor and session info, and include links to full profile and makes getting around your event with interactive maps with details of the event, time, venue, etc. keeping track of events updates with the Activity Feed. And also, EVENTer system will provide recommendation platform. Furthermore, our system will incorporate Voice enabled interface to provide more better and efficient User Experience (UX). EVENTer will have a built-in Smart Recommendation Algorithm and Collaborative Filtering Algorithm to suggest and recommend the users about the popular, interested and nearby events by collecting past data and experiences of users. And also Sponsors and Partners will be valuable party for any event. Also, when any event needs volunteers for specific time and with limitation this system will provide these services by the interests of Uses.

# **Team Members and Divided Roles**

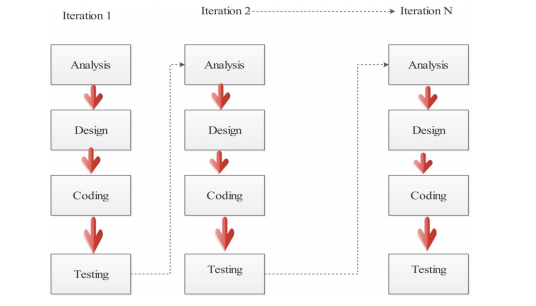
|  |  |  |
| --- | --- | --- |
| Name | Roles | Responsibilities |
| Anish B.K. | System/ Back end developer  Database Administrator | * Test System Interfaces * Define and execute development requirements * Develop, maintain and implement policies and procedures necessary to ensure the security and integrity of the corporate database |
| Nabin Sinkhwal | Back end Developer  Database Administrator | * Develop Use-Friendly Interface and work through design revisions * Evaluating existing applications and performing updates and modifications |
| Aswin Dhakal | Front end Developer | * Develop website * Develop Use-Friendly Interface and work through design revisions. |
| Bishal Shrestha | Project Manager  Documentation | * Review and approve all project deliverables * Day to Day responsibility to keep project on track for the successful delivery of Futsal Reservation System. * Develop Documentation |

# **Methodology**

We have planned to work following these methodologies for the application of skills, tools, knowledge and techniques to a broad range of activities in order to meet the requirement of our project. This section presents a detailed information about the software development process, project approach and the tool that we used for our project.

## Software Development Life Cycle

The framework we will be using for developing this project is iterative model of software development life cycle. in this model, a simple and primitive implementation of very small set of software requirement is done at first, which is followed by the iterative enhancement in the primitive model until all requirements are fulfilled and the software is ready for deployed. The following sub section briefly describe various phase in iterative model of SDLC that was applied in the development of system.



Tools and Technologies used

|  |  |  |
| --- | --- | --- |
| S.no | Technologies | Used For |
|  | PHP | Web framework for our website |
|  | MySQL | Database for application development |
|  | PHP | Building WEB API |
|  | Bootstrap | Web design frame-working |
|  | HTML, CSS & JS | Web designing technologies |

|  |  |  |
| --- | --- | --- |
|  | VS code | Frontend development |
|  | Sublime Text | Backend PHP |
|  | Adobe Photoshop | Graphical works |
|  | Google Group | Team communication |
|  | GitHub | Code repository |

# **Performance Analysis**

The system/ Software is tested using the agile methodology called agile testing. It aligns with the iterative software development. In this method, the testing is integrated alongside development called test driven development (TDD).

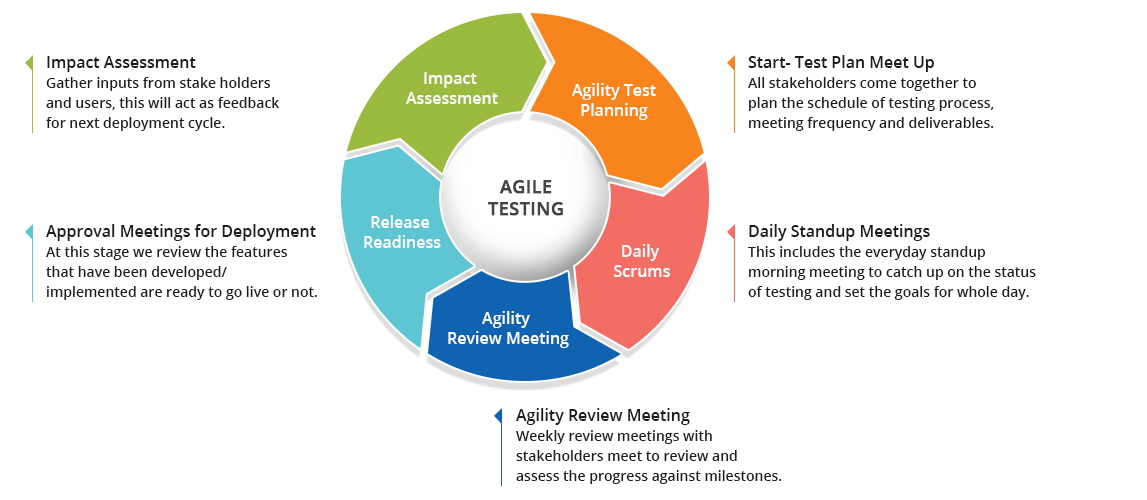


Figure 2 Agile Testing

# **Achieved Deliverables**

* **Website of EVENTer**

The website is our major part of development. The website is integrated with android application using API. The website is user friendly, clean functioning and in accordance with the latest technology.

* **Android Application called EVENTer**

The API integration android application is the major attraction in our project. All the features are also integrated in the android application.

# **Requirement Analysis**

Requirement analysis, in software engineering encompasses those tasks that go into determining the need and conditions to meet for a new or altered product, taking account of possibly conflicting requirements of the various stakeholders, such as beneficiaries and users. It is the early stage activity of requirement engineering which consist all the activities with eliciting, analyzing, documenting, validating, and managing requirements.

## Functional Requirements

|  |  |  |
| --- | --- | --- |
| S.N. | Requirements | Priority |
| 1 | User interface for the users to view events and users | High |
| 2 | User is able to create events (as individuals or organizational) | Essential |
| 3 | User is able to sign in to view event details | High |
| 4 | User is able to sign out from the application | Optional |
| 5 | User must be logged in to create an event | Essential |
| 6 | User can manage their own profile | Optional |
| 7 | User can’tconnect without internet | Essential |
| 8 | User can interact with any event by activities such as comment, like, interested, going etc. | High |

Figure 3 Functional Requirements

## Non-Functional Requirements

|  |  |  |
| --- | --- | --- |
| S. N | Requirements | Priority |
| 1 | The system needs android phone for use mobile app and browser for website | Essential |
| 2 | The application should be user friendly | High |
| 3 | The system is implemented using tools specified (Android Studio 3.5.2, Advanced REST Client, E-Draw, Google Chrome, Xampp, Sublime Text Editor, Adobe Photoshop) | Essential |
| 4 | Applications need to operate successfully (or degrade gracefully) within a wide spectrum of operating conditions, such as a range of supported screen resolutions and form factors, network bandwidth situations and network types (2G/3G/4G/Wi-Fi), etc. | High |
| 5 | Application should emphasize on High Performance, High Responsiveness, Good Scalability, Good Usability, High Reliability, Good Security, Modifiability and Maintainability. | High |

Figure 4 Non-Functional Requirements

## Input Requirements

**Source of Data:**

* The information will be given by user themselves and the events have to full filled detailed requirements.

**Data Required:**

* User Information

This contains user’s full name, username, email, password and address.

* Events

This contains the events details (Title, Date, Time, Venue, Organizer, Category, Price) input by the user.

## Output Requirements

Output Objectives:

The user requires following output from our system

1. Broader details of all the events in the newsfeed and map
2. Full detail on the events they have joined to
3. Event recommendation based on user’s past and consumption data and pattern

## Security Requirements

Users are to sign up and then only get access to the information regarding the events so that each user can have their separate record to the events they are interested in or joined to and only authorized user will be able to update the events. Therefore, each user has been given their own account with username and password to login. Password are saved in database, which are hashed using Sha-1 protocol. This will keep the information safe from intruders as well as those who are authorized to manage database only

# **System Design and UML Models**

Designing according to the requirement specification, we have made an attempt to make sure that the system actually confirms the user requirements of the system.

## System Architecture

The EVENTer system generally revolves around 3 groups of users- Event organizer, Participants and the group of supporters, sponsors and volunteers. The general interaction between them is through this system in the following way.

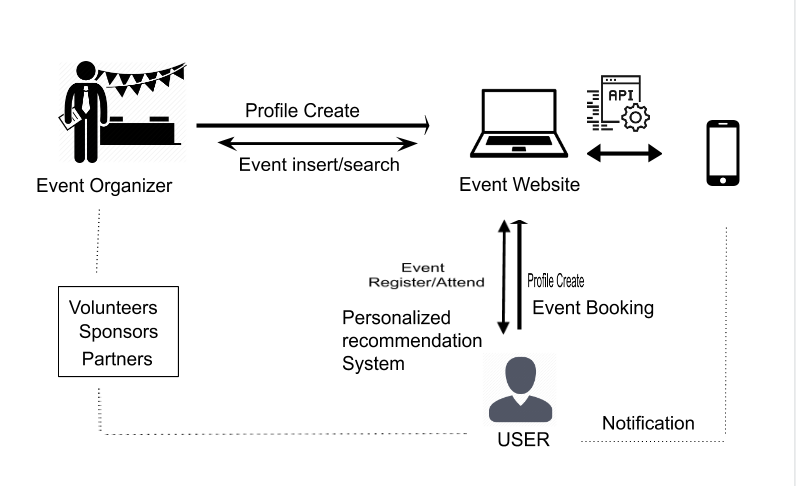


Figure 5 System architecture

All the events organizing companies can create an account in the event system and register or insert the details of the events according to the category. The other user can check the events and add their own events too.

The users/ companies who want to volunteer or became the partners and sponsors can register through this site. The registered events can be altered i.e. can be modified according to the needs of the events.

## ER Diagram

The ER Diagram is a pictorial representation of the overall logical structure of the system’s database. The ER Diagram of our system is given below. It shows the relationship among the five entities of our system. The entities are represented in the rectangle, their attributes are represented in the oval and the attributes that are underlined are the primary keys.

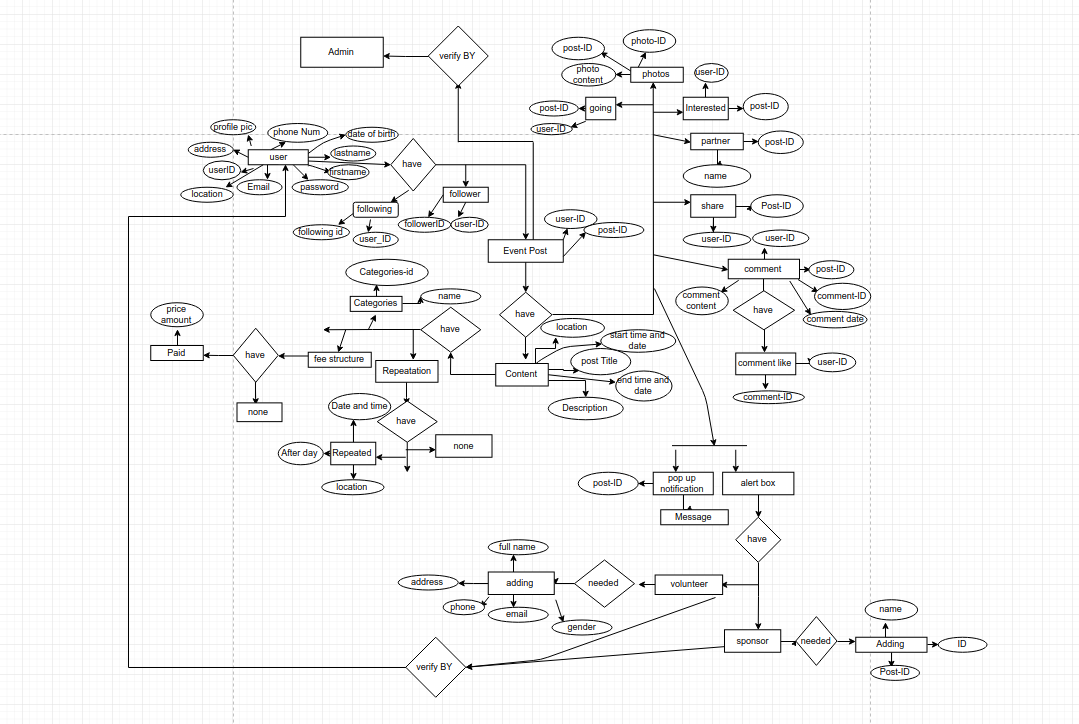


Figure 6 ER Diagram

## Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. The actors for our system are: User, System and APIs. The simplified and graphical representation of what our system must actually do is represented below:

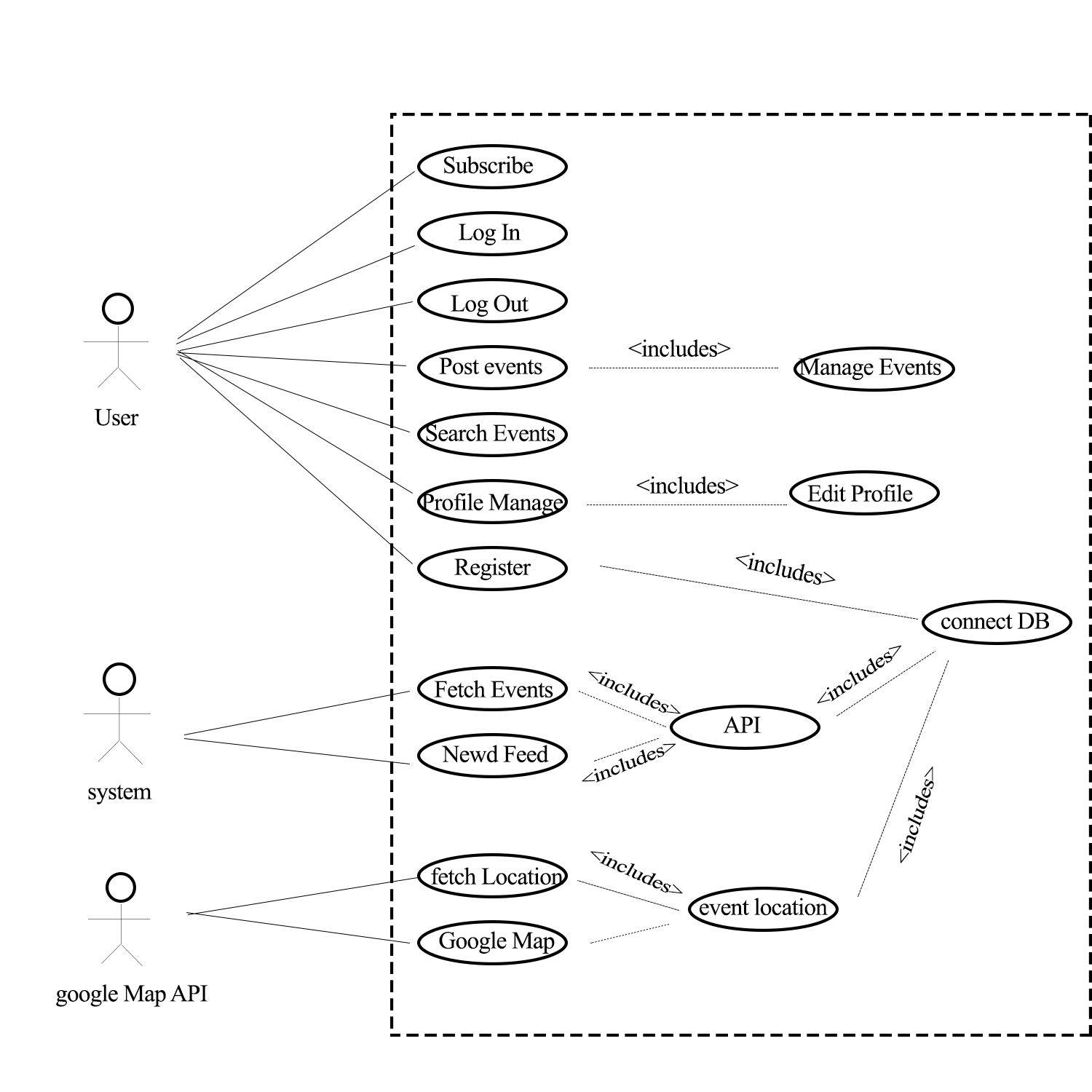


Figure 7 Use Case Diagram

## Class Diagram

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system. We designed the following class diagram to illustrate the system's classes, their attributes, operations (or methods), and the relationships among objects.

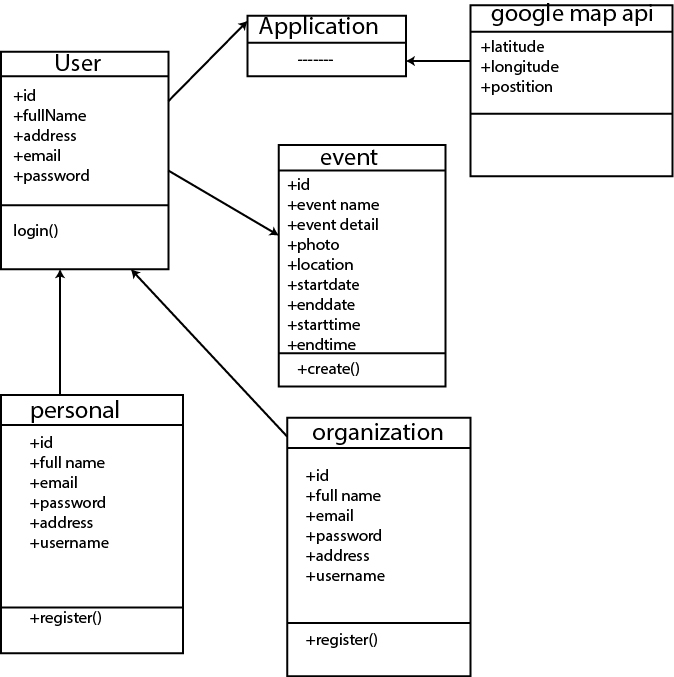


Figure 8 Class Diagram

## Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. We used DFD as a preliminary step to create an overview of the system, which can later be elaborated also be used for the visualization of data processing (structured design)

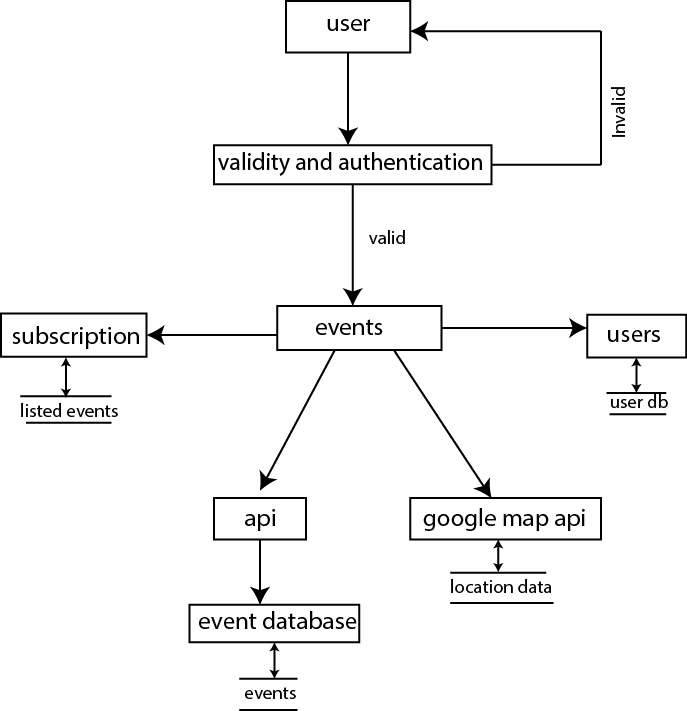


Figure 9 Data Flow Diagram

## Context Diagram

The overall explanation of a system is represented by a context diagram. Using this diagram, we define the boundary between the system, or part of a system, and its environment, showing the entities that interact with it. The diagrammatic representation of EVENTer’s context diagram is below:

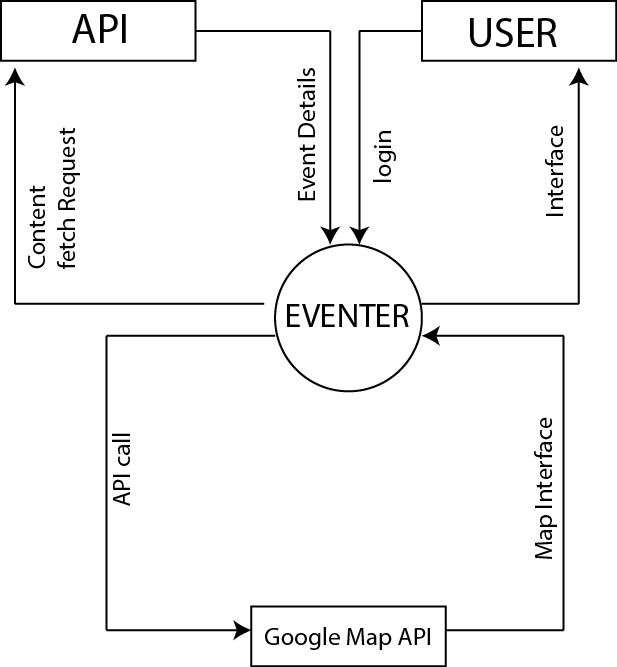


Figure 10 Context Diagram

## Sequence Diagram

Sequence Diagram is an interaction diagram. It shows how the events occur and in what order. For our system we have designed sequence diagrams for most critical and influential activities which are shown below:

### Sequence Diagram for Register

The sequence diagram for a new user to register is represented as below. The user may choose to sign via manually to register his/her details.

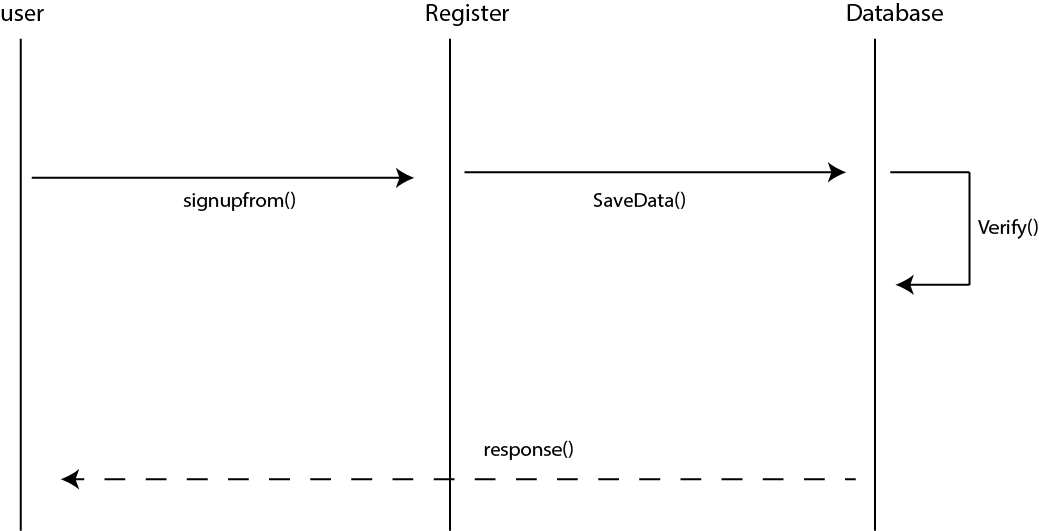


Figure 11 Sequence Diagram For Register

### Sequence diagram of Login

The Sequence Diagram for depicting Login Activity for user is represented below:

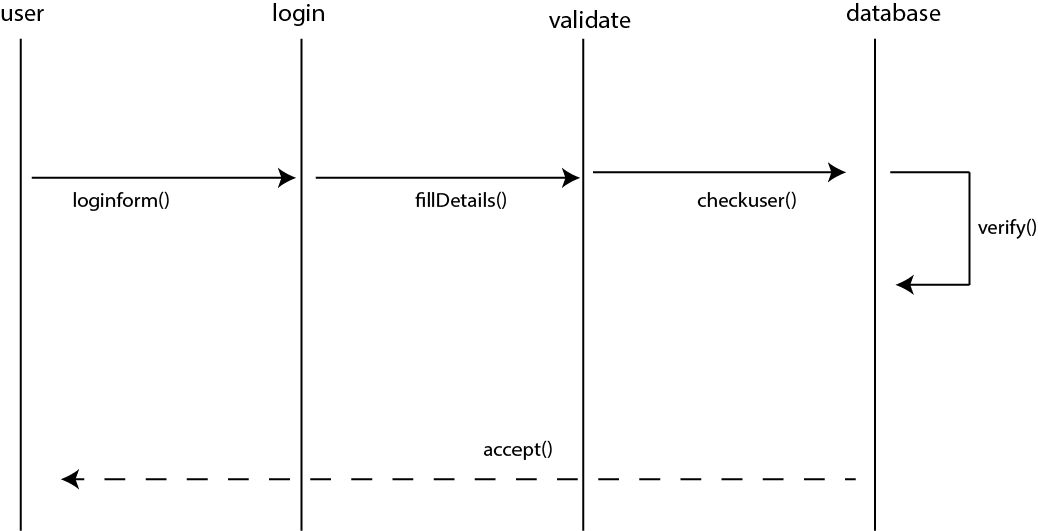


Figure 12 Sequence Diagram of Login

### Sequence Diagram of Logout

The Sequence Diagram for Logout Activity of user is represented as following:

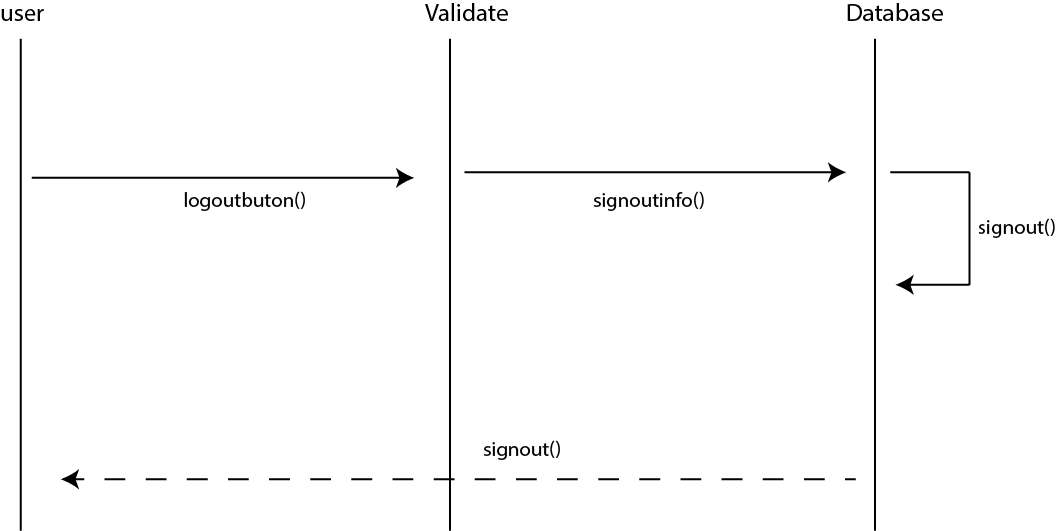


Figure 13 Sequence Diagram of LogOut

### Sequence Diagram of Event

The sequence diagram for any event is below:

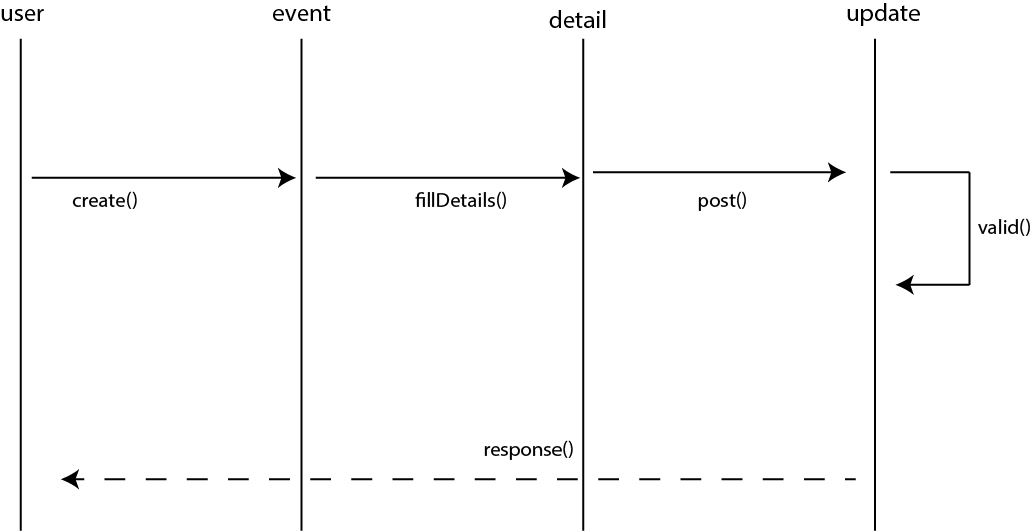


Figure 14 Sequence Diagram of Event

# Project task and time schedule



Figure 15 Gantt chart

# **Testing**

All the elements of the developed worked must be functioned properly. For this we created a test cases for our work, in which elements such as validation, reliability and user acceptance will be tested.

## Testing Table

Each unit to be tested of our system are illustrated in the table. Which is as below:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test  No. | Unit | Test | Expected  Result | Test  Outcome | Evidence | If  Failure |
| 1 | Sign Up | Check whether a new account can be created on filling up required details | Account Successfully created | Successful | Test: 1 |  |
| 2 | Login | Check login credentials for valid email and password | User is successfully logged in | Successful | Test: 2 |  |
| 3 | Google Maps API(web) | Display the respective markers of the updated events | The event markers are placed on the event location | Successful | Test: 3 |  |
| 4 | Google Maps API(Android) | Display the respective markers of the updated events | The event markers are placed on the event location | Failed (API Authorization failure) | Test: 4 | Changed API Key Perform Test no.5 |
| 5 | Google Maps Activity (Android) | Display the respective markers of the updated events | The event markers are placed on the event location | Successful | Test:5 |  |
| 6 | Update Event | Check whether the events updated are successfully registered or not. | Event is successfully updated | Successful | Test:6 |  |

Figure 16 Testing Table

## Test Evidences

### Test 1:

Unit: Sign Up

Purpose: Check whether a new account can be created on filling up required details.

Expected Output: Account Successfully Created

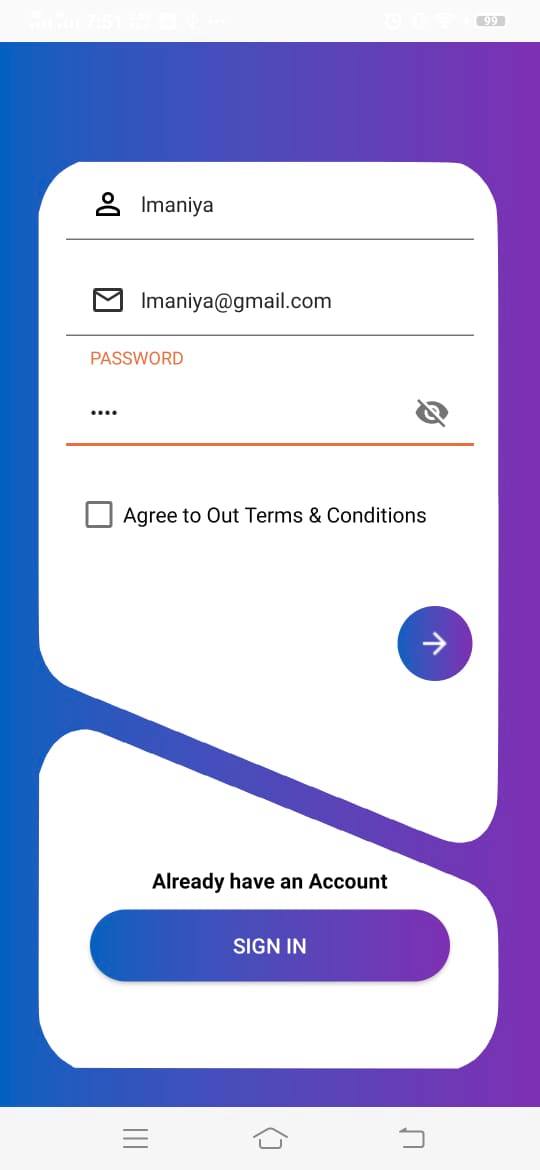


Figure 17 SignUp Test Evidence

### Test 2:

Unit: Login

Purpose: check Login credentials for valid email and password

Expected Output: User is successfully logged in

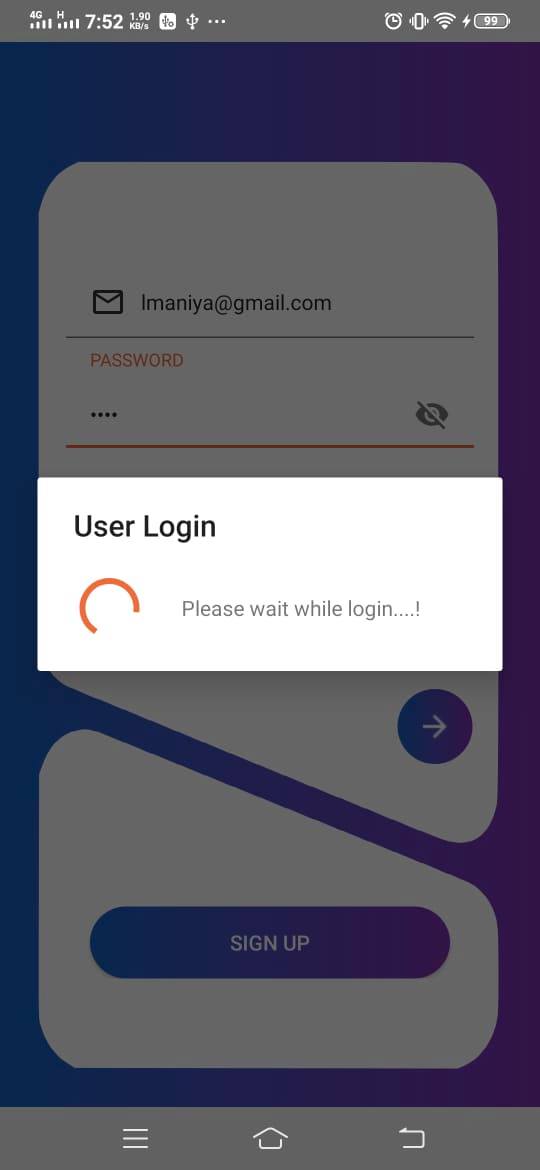


Figure 18 SignIn Test Evidence

### Test 3:

Unit: Google Maps API

Purpose: Display the respective markers of the updated events

Expected Output: The event markers are placed on the event location



Figure 19 Map Test Evidence

### Test 4:

Unit: Update Event

Purpose: To know what happens if all the required field/s are filled correctly by user

Input: All required fields were properly filled

Expected Output: Event is successfully updated.

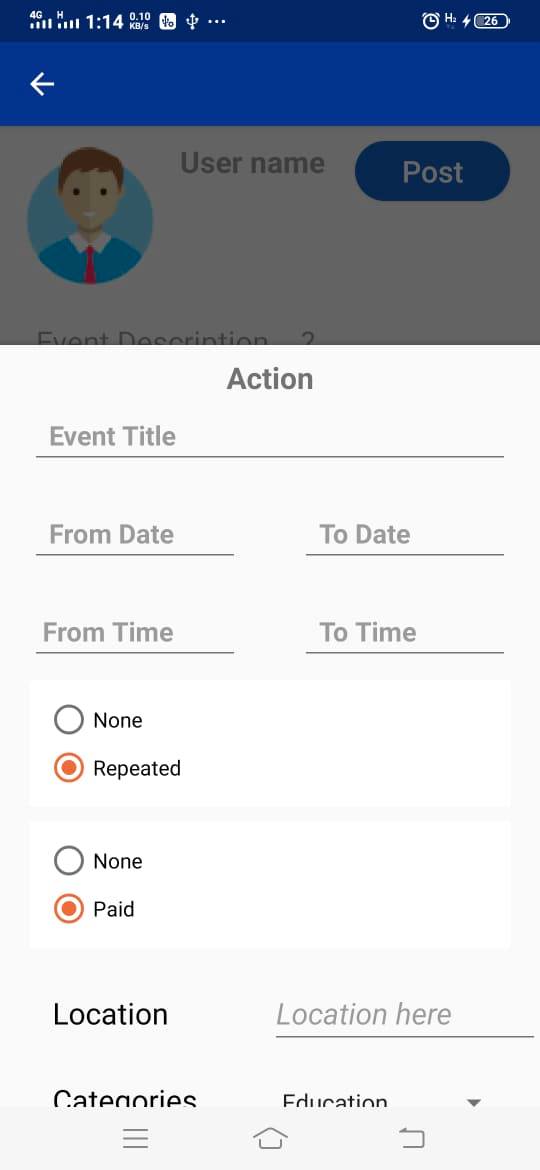


Figure 20 Update Event Test Evidence

# **Conclusion and Future Extensions**

Our system **“EVENTer”** is now at initial phase with its beta version having most of the basic functionalities discussed before. All the modules have been working after integrating and are ready for the demo. As the features adding up the level of complexity has been increasing as well. However, it is not complete with the ideas we have put through and might need more improvisation in the upcoming days as well. This makes us think about the future extensions that we are going to implement in this system.

The following future extensions will be implemented in our system in upcoming days:

1. Recommendation system
2. Volunteer, Sponsors and Partners integration
3. Online payment for paid event

# **References**

- "Google Maps API – Terms of use". [Online]. Available <https://developers.google.com/maps/terms?csw=1>

- "Official Google Blog: The world is your JavaScript-enabled oyster".[Online]. Available <https://googleblog.blogspot.com/2005/06/world-is-your-javascript-enabled_29.html>

- "Googles Map API “<https://developers.google.com/maps/faq?csw=1#tos_commercial>

- “Facebook Events- Your guide to hosting great events with Facebook” Available <https://events.fb.com/#create-a-great-event>

- “Events in Nepal.com –About-us “http://www.eventsinnepal.com/#

- “Eventbrite-about-us” <https://www.eventbrite.com/about/>

- “Volley Library for Android” <https://developer.android.com/training/volley>

- “PHP framework for API” <https://www.codeofaninja.com/2017/02/create-simple-rest-api-in-php.html>

- “UML design “ <https://www.smartdraw.com/uml-diagram/>

- “Event Management Basic” <https://nevonprojects.com/online-event-management-system/>

1. **Appendix**

System Snapshots

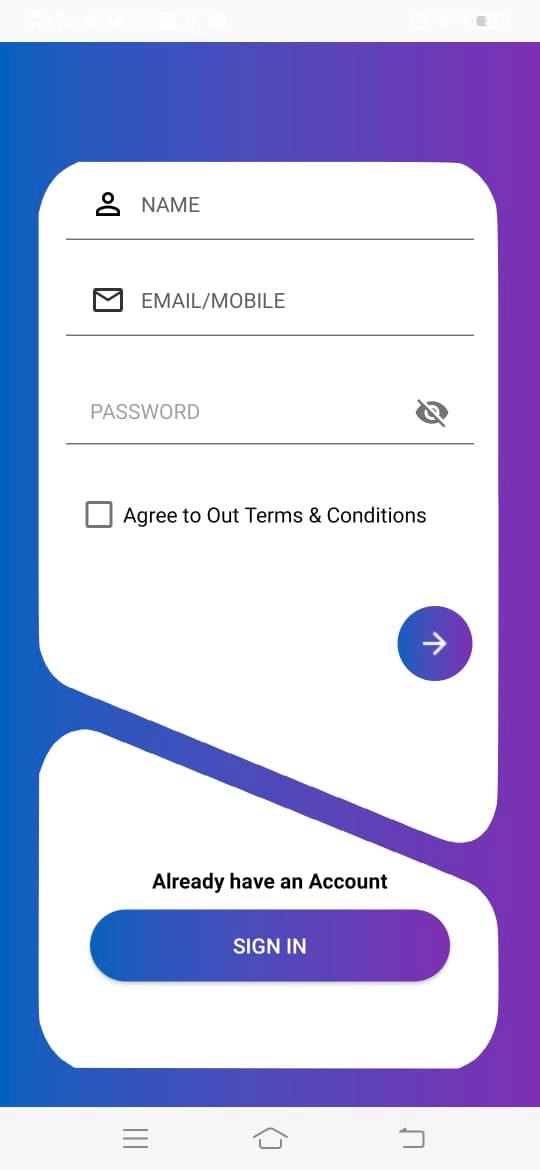


Figure 21 Sign Up

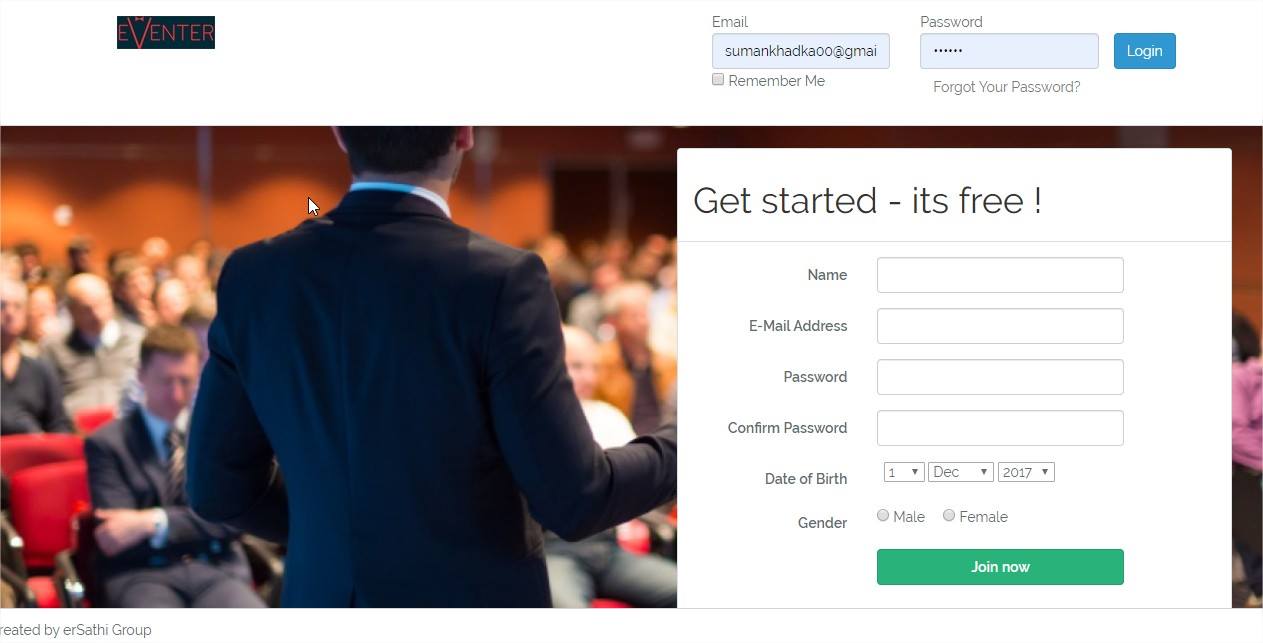


Figure 22 Login and Sign Up in web

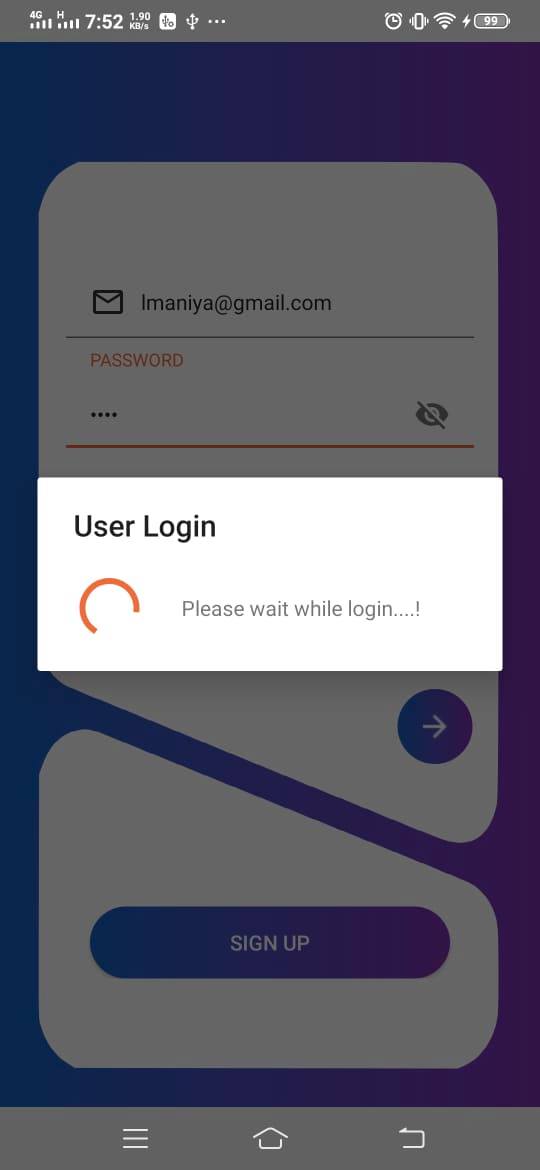


Figure 23 Login In Android

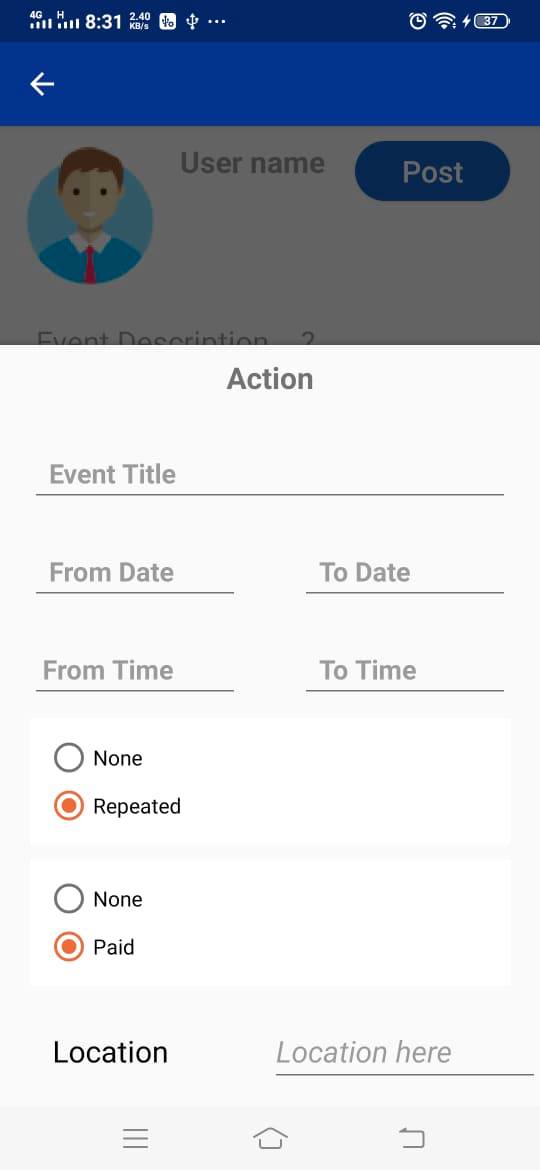


Figure 24 Update Events in Android

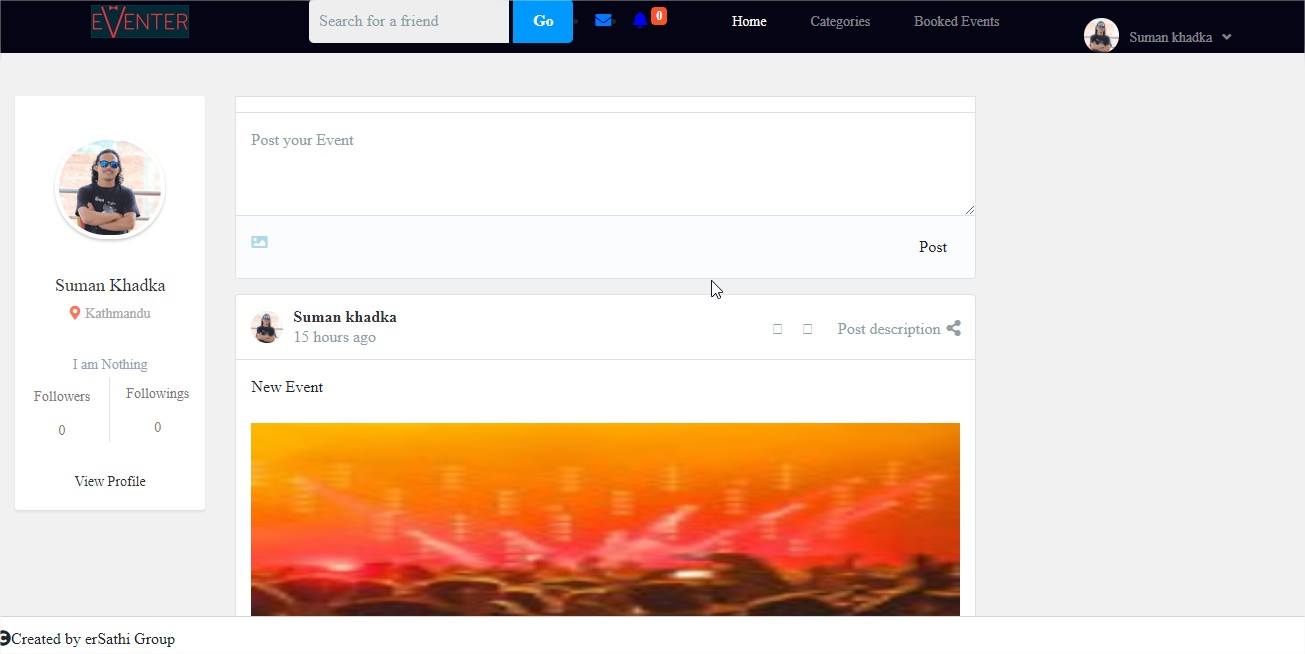


Figure 25 Update events in Web



Figure 26 Events location in Map (android)

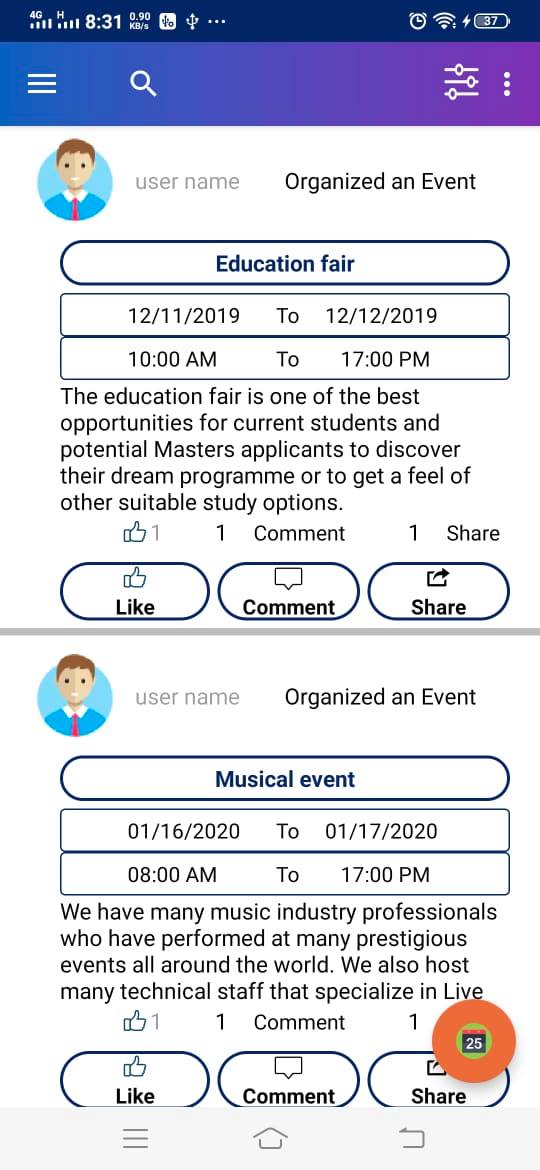


Figure 27 Home Page (Android)

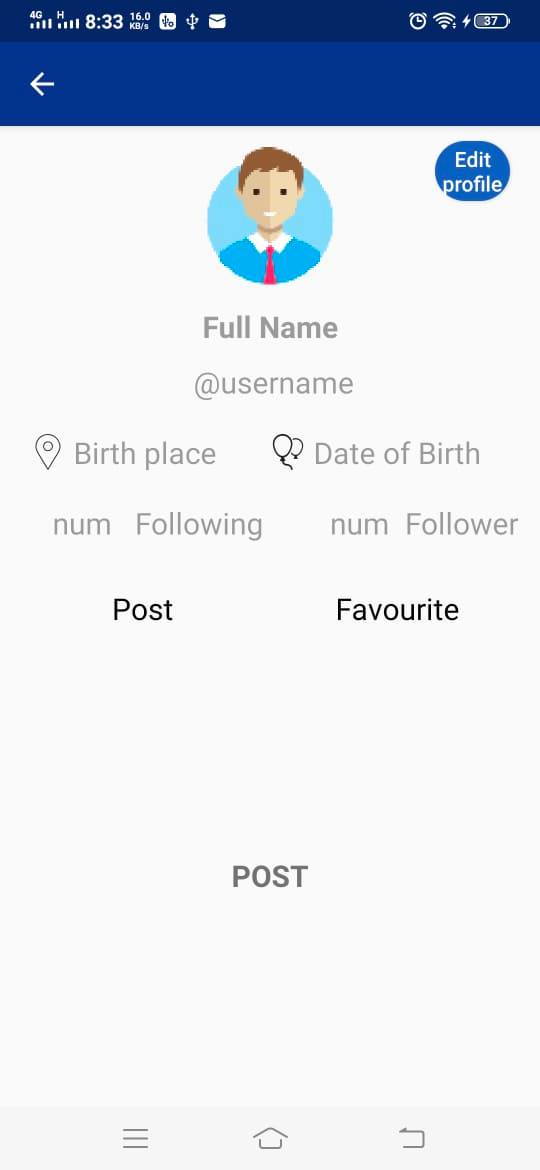


Figure 28 User Profile

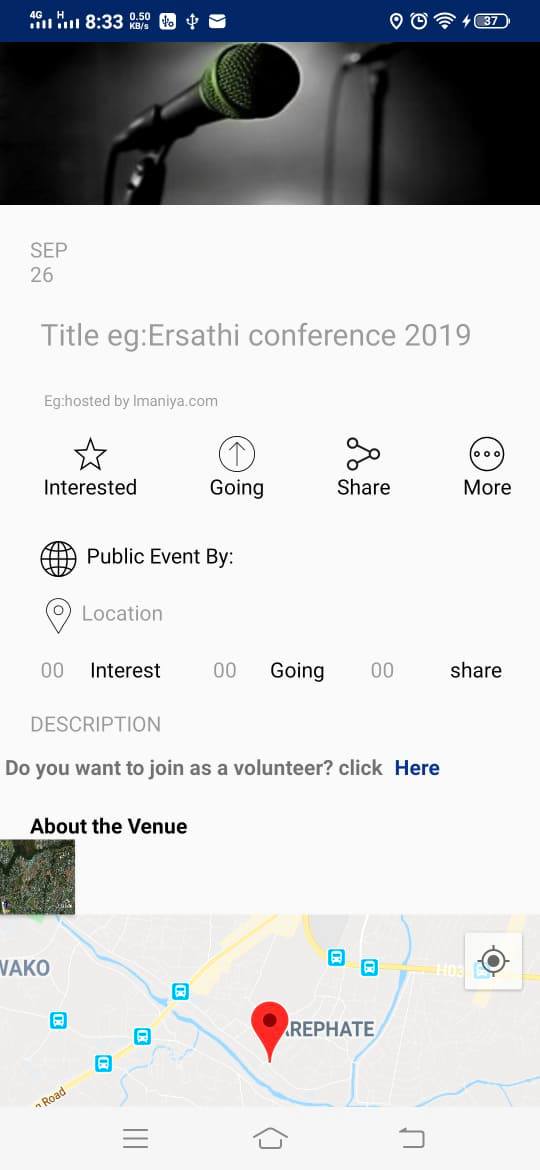


Figure 29 Event Detail

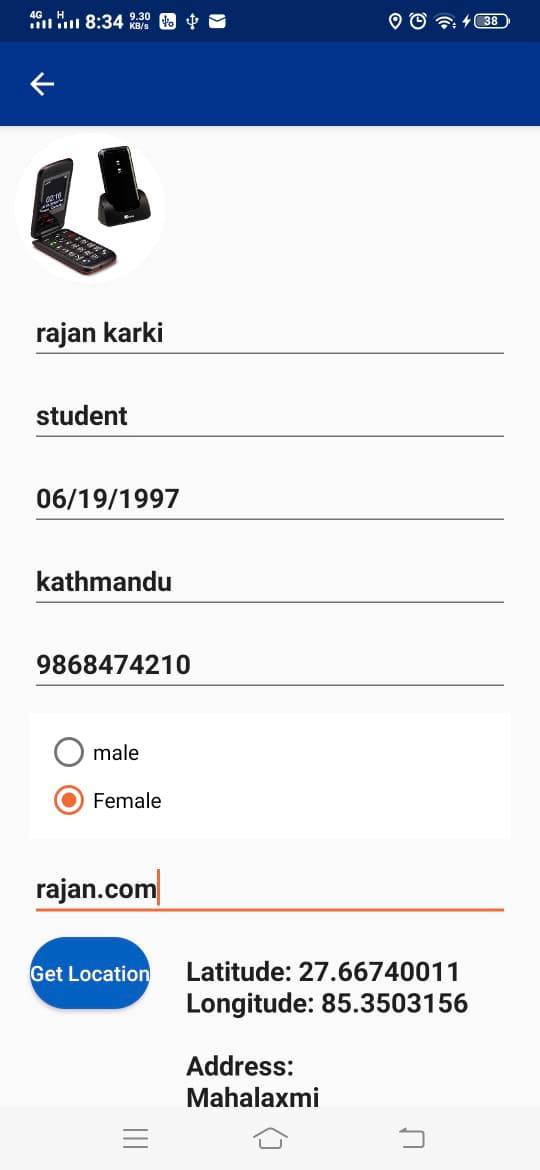
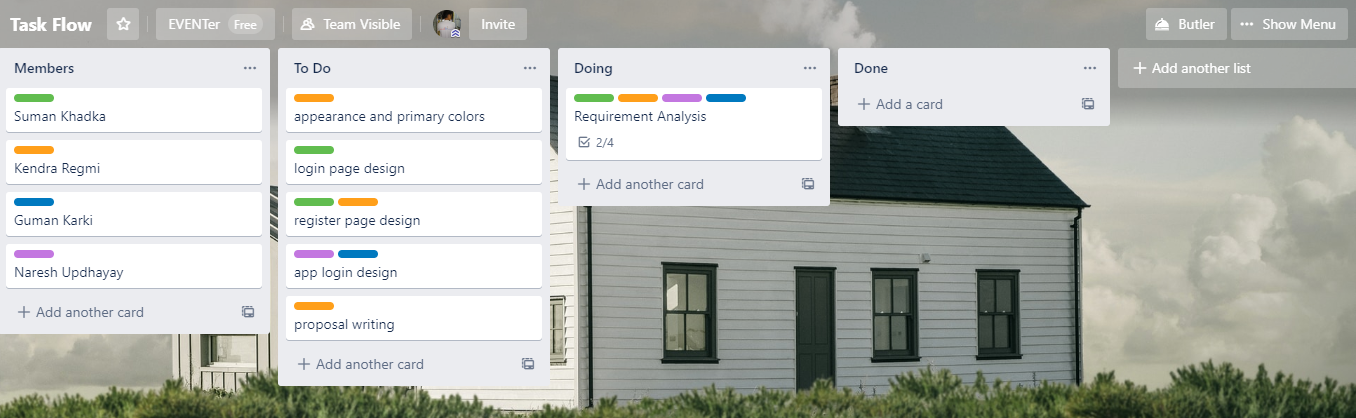
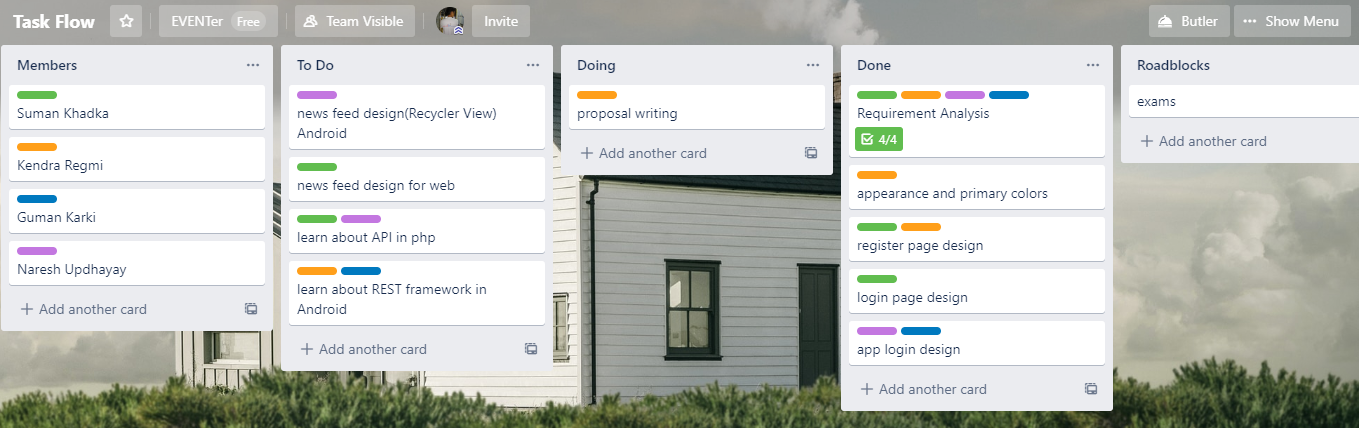


Figure 30 Profile Detail Fill Up





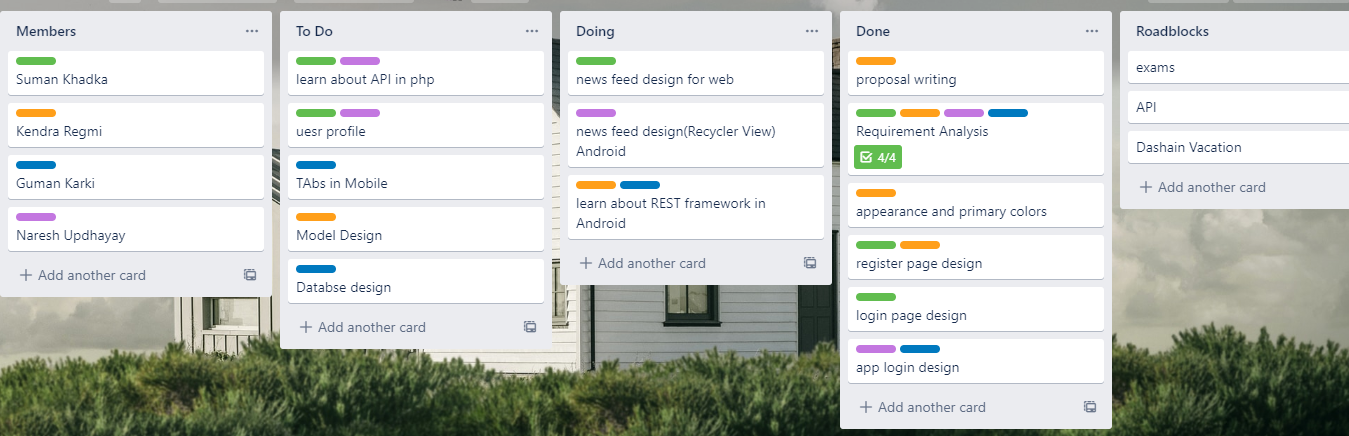


Figure Scrum Board In trello