ACKNOWLEDGEMENT

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

ABSTRACT

As we are at the brim of the Visit Nepal year 2020, the decentralization of tourism in Nepal has become absolutely necessary. Almost ninety percent of the tourists arriving Nepal visit the most popular destinations like Kathmandu, Pokhara, Chitwan and Everest, whereas the remote places like Rara Lake and Phoksundo National Park have not seen satisfactory inflow of tourists despite them being rich in natural beauty and tourism potentiality. There are still a lot of tourist destinations in Nepal that need the attention of the tourists.

Treasure Nepal is a treasure hunt application available in both Android and iOS platforms, where the users travel to different places in order to collect treasures and increase their scores. The app has an integrated map that navigate the users to various tourist destinations in Nepal. The project aims to take the attention of tourists and visitors towards otherwise unnoticed tourists destinations in Nepal. The tourists need to physically reach to a place in order to collect treasures. The tourists who collect treasures at places that are remote and left behind will get higher scores than the ones who visit common popular destinations. The users can compete to get their profiles on top of leaderboard on the basis of the scores they have collected. Moreover, the scores collected by the users will provide them with exclusive offers and discounts on hotels, restaurants, travel agencies, movie tickets, and many more commercial services. On the long run, the project sets its objective to strive for the decentralization of tourists visiting Nepal.

Keywords: Visit Nepal 2020, Treasure hunt, Tourism

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

vi

1. Introduction

Treasure Nepal is an Android / iOS application for a treasure hunt game targeted to internal as well as external tourists visiting several destinations within Nepal. With the view of encouraging tourists to visit remote and unexplored part of the country, the application aims to increase the traffic of tourist in such remote and unnoticed locations as well as promote their tourism.

According to Nepal Rastra Bank, the total contribution of the foreign exchange from tourism to the total Gross Domestic Product (GDP) of Nepal was only 2.2% in the year 2017/18. [?]. Observing at the statistics, the ratio of contribution of tourism to GDP is not satisfactory in Nepal. Looking at the Federal Budget of Nepal of fiscal year 2019/20, only Rs. 2.68 billion out of total Rs. 1.53 trillion budget (0.17%) has been allocated for the development of tourism infrastructures [?]. This also shows that tourism sector is not getting quite satisfactory budget for its development. For a geographically challenged landlocked country like Nepal, tourism can in fact become one of the major backbones of economy of the nation. For developing the tourism sector in future days, collaboration from all parties – from government to the common people — has become absolutely necessary.

The Government of Nepal has taken efforts to celebrate the year 2020 officially as the Visit Nepal Year 2020. The Government aims to bring two million tourists in Nepal during the year 2020 [?]. At the brim of year 2020, tourism in Nepal is largely centralized to a few popular destinations. The places like Kathmandu, Pokhara, Chitwan, Annapurna area and Everest area are largely flocked by tourists while destinations like Rara Lake, Shey Phoksundo National Park or Khaptad National park struggle to get satisfactory inflow of traffic. This project is an endeavor taken to contribute as much as we collective can to uniformly distribute the traffic of tourists in different places, to the success of upcoming Visit Nepal 2020 project and tourism development in Nepal as a whole.

1.1 Project Overview

Treasure Nepal is an installable mobile application available in both Android and iOS platforms. Once the user installs app and registers her account, she can view various local as well as distant treasures in a map within the application. Each of the treasures have some specified number of points associated with them. When the user decides to collect a treasure, she should physically go to the location where the treasure is installed, and then scan QR code installed at that location. Once the user scans the

QR at the treasure location, those points will be awarded to the user's account. The user cannot collect the same treasure again and again. When user's point crosses a certain level, she is provided with several offers, rewards and discounts at various hospitality service providers, preferably the ones that are the nearest.

1.2   Problem Statement

The centralization of traffic of tourists visiting Nepal has largely underestimated the potential and beauty of many travel destinations, specially in remote areas. As a result, these places have very low traffic of tourists. In addition, as majority of people in such places rely solely on tourism industry for their livelihood, this problem has pushed those communities even further down below the poverty line. Furthermore, the tourists visiting different places in Nepal may not get sufficient information about the local attractions that are not briefed to them by their guides/agencies. An approach to reliably provide them information about the unnoticed and unvisited places in Nepal, and to encourage them going to that place has been absolutely necessary.

1.3   Objectives

The project has put forward the following objectives:

- To provide as the deliverables an Android as well as iOS app freely to the users as a means of information about places as well as a method of entertainment.

- To decentralize the tourism traffic and encourage uniform flow of tourists at various destinations across Nepal.

- To promote and encourage the tourism in remote, unnoticed and novel destinations which otherwise are not popular or have low inflow of tourists.

- To promote local businesses and people's life standard in remote places by encouraging the tourists to visit those places.

1.4   Significance of the Study

The study of this project is significant owing to the fact that the solution to the problem we are trying to solve has rarely been ever created, and our idea is one of the first of its kind. Our approach of trying to solve the problem while providing entertainment

to the users will certainly be a reason why people won't hesitate to install and use our application. Also, we are very near to the Visit Nepal Year 2020, and this study will certainly be helpful in some ways in contributing to achieving the objectives set by the Government of Nepal in the year 2020. It is expected that the project will reach to a significant majority of tourists that visit Nepal in 2020, because the inflow traffic of tourists in that year is expected to be higher than usual..

2.   Literature Review

This section consists description of the literature study performed during the development of this project.

2.1   Visit Nepal 2020

The Visit Nepal Year (VNY) 2020 project was officially introduced by Nepal Tourism Board (NTB) in 2015 and aims to bring two million tourists in Nepal during the year 2020 [?]. The year 2020 has been chosen as the national tourism year after the last similar tourism year in 2011. Its slogan – 'Lifetime Experiences' – has been translated to more than ten different languages. The tourism board also had developed a plan to train ten thousand people to provide quality service to the tourists. The three major commitments made by the VNY2020 for sustainability are 'Climate Change', 'Community Based Tourism' and 'Going Green' respectively [?].

Community based tourism has been very popular in recent years in Nepal, especially in the Himalayan region and along the trekking routes. For instance, the numbers of registered home-stays in Nepal increased from 283 in 2017 to 324 in 2018 [?]. This number was just 217 in 2015. This increasing interest in the community based tourism is in fact due to the tourist's wish to see the actual lifestyle and living standard of the people living in various communities in Nepal. These forms of tourism not only provide unforgettable experiences to the visiting tourists, but are also a means of income and living for people in such areas. This project has also taken steps to actually promote such local and community based tourism in various places in Nepal.

2.2   Tourism in Nepal

Tourism is one of the most important service industry contributing to the economy of Nepal. Being rich in natural, cultural and bio-diversity, Nepal comes within the short list of the budget-friendly choices of tourism destinations all around the world. As the country has finally gained stability after a decade long armed-chaos, political instability and the devastating earthquake, the number of tourists interested to visit Nepal is increasing day by day. As of 2019, the average cost per day of traveling to Nepal is only $25, which is the reason Nepal has been seen as budget-friendly tourist destination [?].

The major itineraries of the tourists visiting Nepal include mountaineering, trekking,

religious pilgrimages and holiday spending. The northern part of Nepal has the mountain range with highest elevation in the world, called as the Himalayas [**?**]. These mountains serve as a destination for the tourists seeking mountaineering as well as trekking. The famous trekking destinations in Nepal are Everest Base Camp, Annapurna Circuit and Langtang Trekking Route [**?**]. Other attractions include various lakes, the famous of which are Tilicho Lake, Rara Lake, Phewa Lake and Gosaikunda.

Nepal is also rich in biodiversity. Currently there are 12 national parks, 1 wildlife reserve, 6 conservation areas, 1 hunting reserve and 10 Ramsar sites as the protected areas of Nepal [**?**]. Among them, Chitawan National Park is the most popular one. A lot of tourists visit these areas for the purpose of jungle safari and animal sports.

Nepal is quite rich in cultural diversity too. Some of the major ethnic groups in Nepal are Kshetri, Brahmins, Magar, Tharu, Tamang, Newar, Kami, Musalman, Yadav, Rai, Gurung, Thakuri, Limbu, etc. A large portion of the tourist inflow in Nepal, especially from south -east Asian countries like India, Bhutan, Thailand, Myanmar, etc. occurs for religious and pilgrimage purposes. In recent years, the various communities have incorporated home stay programme to host tourists in their home and offer their cultural courtesy, which has sent setup good environment for the cultural tourism in Nepal. The festivals like Dashain, Lhosar, Chhath, Gai Jatra, Buddha Jayanti, etc. are some of the most popular festivals of Nepal.

2.3   Hospitality Industry in Nepal

Hospitality industry is a collection of service based industries that are focused on providing hospitality services to the consumers. The industries like hotels, lodges, restaurants, food and drink services, event planning services, amusements, adventures, travel and transportation services etc. all come under the common umbrella of hospitality industry. The industry of hospitality is specially important to be studied for this project, because it is those service providers that will provide offers and discounts to the tourists who use the application.

The hospitality industry is largely operated by private and Non Governmental Organization (NGO) sectors in Nepal. Some of the major businesses in the field of tourism in Nepal are hotels, restaurants, lodges, travel agencies, ticket providers, tour guides, rafting agencies, mountaineering agencies, trekking agencies, recreational aviation, etc. Some statistics on number of different businesses providing hospitality services are listed in Table **??**.

Table 1: Tourism Industry Statistics in Nepal in Year 2075 BS

S.N.	Description	Number
1.	Star Hotels	129
2.	Tourist Standard Hotels	1125
3.	Total Hotels	1254
4.	Travel Agencies	3508
5.	Trekking Agencies	2649
6.	Rafting Agencies	73
7.	Tourist Transportation Services	77

Source: Tourism Statistics, 2018. Ministry of Culture, Tourism and Civil Aviation. [**?**]

2.4   Cross Platform App Development

Any application that is developed targeting both of these platforms is known as a cross platform application. Android and iOS are two major mobile development platforms in the market today. In today's market, the businesses have opted to allow all of their users use their services through all of the technological platforms available to the client. For example, the same services of a company can be used by a web based portal, an Android mobile phone, an iPhone, iPad, or even a desktop computer software.

When the matter comes to cross-platform mobile application development, the most popular frameworks available today are Flutter, Ionic and React Native [**?**]. The two frameworks are compared in Table **??**. React Native is relatively older technology first released in 2013 and currently managed by Facebook Inc. Flutter, a new technology, is the similar framework from Google. React Native uses JavaScript as its language while Flutter uses Dart. So, to use Flutter, one has to learn Dart first, which itself is not a primary language many developers learn in early stages of their programming career. React Native being older has relatively wider support and user base than flutter. Flutter is rapidly increasing its user base too. The choice between one of them depends upon the programmer's preferences and choices. We opted to use React Native in stead of Flutter in this project because of the team member's pre-familiarity with Javascript language and better community support.

Table 2: Comparison between React Native, Ionic and Flutter

S.N.	Category	React Native	Ionic	Fluttter
1.	Developed By	Facebook and Community	Drifty Co.	Google and Community
2.	Released	2013 (Open Sourced in 2015)	2013	2017
3.	Language	JavaScript	HTML, CSS, JavaScript	Dart
4.	Community Support	Old and Strong	Old and Strong	Relatively new
5.	Used By	Facebook, Instagram, Tesla, Uber, Walmart, Airbnb	Marketwatch, NHS, SWorkit, Untapped	Alibaba, Google Ads, Tencent

Source: CodeBurst.io [**?**]

2.5   Client Server Architecture

Client-Server architecture is a model of computing where the services and resources to be provided are managed by a computer called a server and the provided services are utilized and used in the computer called a client, as shown in Figure **??**. The clients use the services hosted by the server with the use of a network or internet connection. There are several noticeable advantages of using a client server architecture in stead of a single tier architecture. The most important of them is the separation of workloads between the two computers. The maintenance also becomes easy as all the services and resources are hosted by a single computer. The separation of the client and server enables the developers to employ security strategies in both client and server layer and thus makes it more secure. Data and resource sharing is also improved by the use of a central server where all the clients can have access to.

In our project, the server tier is the API server that will handle all the data requests received from mobile device and the client will the application running in Android phone or iPhone.
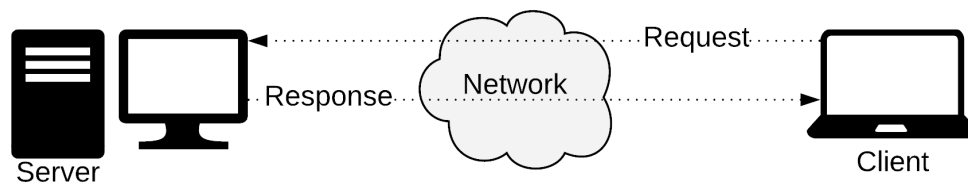
Figure 1: Client Server Architecture

2.6 REST API

API stands for Application Programming Interface. It is a set of of functions, procedures and protocols between a client and a server so that the client can use the application and services of a server. An API can be Web Service based, Operating System based or Database Management System (DBMS) based. Web based APIs are those which provide the interface between the client (Presentation logic) and the database (data and business logic). A client can initiate either a read or a write operation request the API server on some data. Each time a client sends a request to the API server, it breaks it down to different database operations and then read/write data to the database. The API server also sends back the data it receives from the database in formats like XML(Extensive Markup Language) or JSON(JavaScript Object Notation).

REST stands for Representational State Transfer. It is a software architecture that defines a set of constraints for creation of web services and APIs. The web services that abide by these constraints are commonly called RESTful. The web services are the programs that are run in the server that respond to various HTTP requests like GET, PUT, POST, DELETE, etc. and send response to the client of those services.

Hence, a RESTful API is defined as a software architecture where an API server provides its clients with various endpoints through which the client can read and write data from/to the database. The API server is programmed in such a way that it returns a specific set of dynamic data when one of it's endpoint is hit with correct method, authentication header, body and parameters. Table **??** shows commonly used convention in request-response model of a RESTful API.

Table 3: Commonly used convention in API Programming

URL	Method	Parameters	Body	Commonly Used For
/endpoint/	GET			Get all entries from the database
/endpoint/{id}	GET			Get the entry corresponding to the provided id
/endpoint/	GET	params		Get a list of entries that are filtered by provided parameters
/endpoint/	POST		entry	Add the new entry into the database
/endpoint/{id}	DELETE			Delete the entry corresponding to the provided id
/endpoint/{id}	PUT		new_entry	Update the entry corresponding to the given id with new entry
/endpoint/{id}	PATCH		{attr: value}	Update only the specified attributes of the entry corresponding to provided id with new values

2.7   JavaScript Object Notation

JavaScript Object Notation (JSON) is an open-standard lightweight method of data interchange. It is one of the widely used method of exchange of data between client and server in RESTful API architecture. JSON makes use of key-value pairs of strings to represent data. The collection of unique keys and corresponding values is known as JSON object. A collection of JSON objects is known as a JSON array. A JSON object is enclosed in curly braces '{ }' whereas the JSON array is enclosed inside square brackets '[ ]'. In Figure **??**, an example of JSON array consisting of two JSON objects is shown.

```
[
  {
    "id" : "1",
    "name" : "Bikalpa",
    "roll" : "15395"
  },
  {
    "id" : "2",
    "name" : "Avinash",
    "roll" : "15306"
  }
]
```

Figure 2: The format of JSON object and array

It is quite important to note the difference between an actual JavaScript Object and a JSON formatted object. The first and foremost difference is that JSON is just a format of transferring text, that just happens to be formatted in a way JavaScript objects are written. The JSON object doesn't directly correspond to a JavaScript object. In fact, JSON only supports strings as its key and value, whereas the values in JavaScript object can be any data type. A parser is needed to first convert a JSON string to actual JavaScript object if it is to be used inside JavaScript code.

2.8 Global Positioning System

Global Positioning System, also abbreviated as GPS is a global system of satellite based navigation where about a total of 33 satellites orbit the earth to provide geolocation and time information to the receiver anywhere in the earth. As of 2019, the GPS is owned by the United States Government and operated by United States Air Force [?]. It is freely available for use to everyone with the use of a GPS receiver. Today, almost all of the smartphones have a built-in GPS sensor which they use as the method of navigation.

The GPS satellites any particular time, a GPS receiver is directly in line-of-sight with at least four satellites. All of the 33 GPS satellites regularly emit radio waves carrying information about their current position and time. When a GPS receiver receives those radio waves, it can figure out how far away from those satellites it currently is. Based on the distance of the GPS receiver from at least three satellites in its line of sight, the receiver can deduce its location by the method called as trilateration [?].

Let us consider a GPS receiver that is in line of sight to three GPS satellites at some place in the world. When the receiver figures out its distance from one of the satellites, its actual position is somewhere in the surface of the sphere centered at the position of satellite, with radius equal to the distance from the satellite. If the receiver knows its distance from all of three satellites, its position is exactly at the intersection of three

spheres centered at each of the satellites with radius equal to the distance between them and the receiver, as shown in Figure **??**. In this way, the accuracy of the position of receiver can be pin-pointed to the accuracy of about 30 centimeters [**?**].
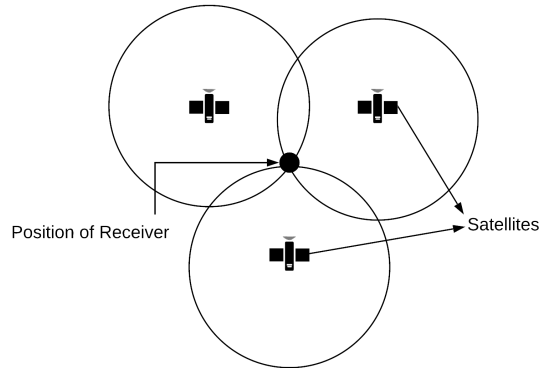

Figure 3: Identification of geolocation by trilateration

2.9   Longitude and Latitude

The position of any point in the world can be precisely described by two real numbers called longitude and latitude. To make this possible, 180 equidistant and parallel horizontal imaginary lines are drawn over the surface of the earth, which are called latitudes. The longitudes, also called as meridians are the 360 vertical lines that have their ends at the two poles of the earth and span through the surface of the earth. As a reference, the longitude that passes through the center of the earth , also called the equator, is considered as 0 degrees latitude. In the similar way, the longitude line that passes through the British Royal Observatory in Greenwich England is considered as 0 degrees longitude. The distance between two consecutive latitude or longitude is considered as one degree. As a convention, the northern hemisphere is counted as positive latitude and the southern hemisphere as negative latitude. Similarly, the the hemisphere to the east of 0 degree is considered positive and the western hemisphere is considered to have negative longitude. To precisely locate a point in the earth, we provide a coordinates of two real numbers, which describe the distance of the point in degrees from the 0 degree latitude and 0 degree longitude lines respectively.

Given the coordinates of two points, we need to find the distance between those two points on the surface of the earth when we have to verify whether the tourist collected the treasure from within a permitted area around the actual treasure location or not. This distance between the two points on the surface of earth, also called as the great-circle distance can be calculated by using Haversine Formula [**?**].

Let us consider two points *A* and *B* on the surface of the earth with co-ordinates

(ψ_1, ϕ_1) and (ψ_2, ϕ_2). Let R be the radius of the earth. Then according to the Haversine Formula, the great-circle distance d between the points A and B can be calculated as:

$$d = 2R\sin^{-1}\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos\phi_1\cos\phi_2\sin^2\left(\frac{\psi_2 - \psi_1}{2}\right)}\right) \qquad (1)$$

2.10 Similar Applications

There are several treasure hunt applications available in the application stores developed for entertainment purposes. Some of them are GooseChase, Locandy, Huntzz, Scavify and Geocaching [**?**]. This application distinguishes itself from these applications due to its focus on the tourism industry. The available applications are developed only for entertainment purposes. The existing applications are actually used for challenging friend / family to solve some puzzle and collect treasures within a small amount of area. In contrast, our application covers the whole country. Also, the users themselves have to set treasures and spots and enter them into the application to create a challenge and then challenge someone else to find and collect them. In contrast, the users of the proposed application are already provided with the treasures and they don't actively take participation in creating one.

2.11 Challenges

One of the major challenges realized is the validation of the collection of treasures by the users. If only QR code is used for validating that a tourist has in fact reached a destination, there is a high chance that the QR codes get shared among people and people will remotely validate themselves having gone to a place and collected a treasure just by scanning the photo of the QR from a remote location. A countermeasure that can be used is to add actual location data from the user's phone's GPS sensor as an additional parameter for validation. A treasure is only considered to be collected if a user scans the QR from within a specific distance from the actual treasure location. That distance can be calculated using the Haversine formula discussed in section **??**.

GPS spoofing is one of the major challenges for any system that has used GPS for the validation. GPS spoofing is the process of modifying a GPS receiver unit so that it broadcasts incorrect GPS signal. The solution to GPS spoofing problem is quite complex and out of scope right now. However, some countermeasures to tackle GPS spoofing are monitoring absolute as well as relative GPS signal strength; checking time intervals and performing comparison; and performing sanity checks [**?**].

3. Methodology

This section describes the methodology that have been followed during the development of the project.

3.1 Software Development Life Cycle

The project has been developed as per the waterfall model of software development life cycle as depicted in Figure **??**. The reason for choosing this model is the lack of sufficient time duration for agile and iterative methods, as well as very low chances of the changes of requirements in the process of development.
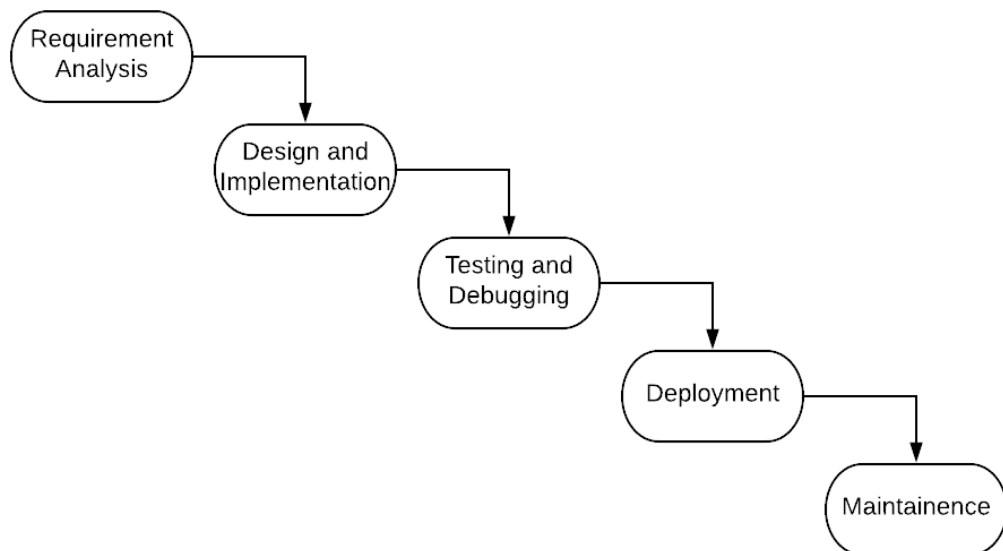
Figure 4: Proposed software development life cycle

The life cycle began when the team collected and evaluated the requirements expected from the application. The design and implementation phase was to design and build both API services and client applications. By the end of this phase, a minimal viable product (MVP) was already constructed. In the testing and debugging phases, the quality control methods were be applied to both API and application. Finally, the application was deployed to the local server at the end of the deployment phase.

3.2 Technical Architecture

The application has been built upon the client-server web architecture, as illustrated in Figure **??**.
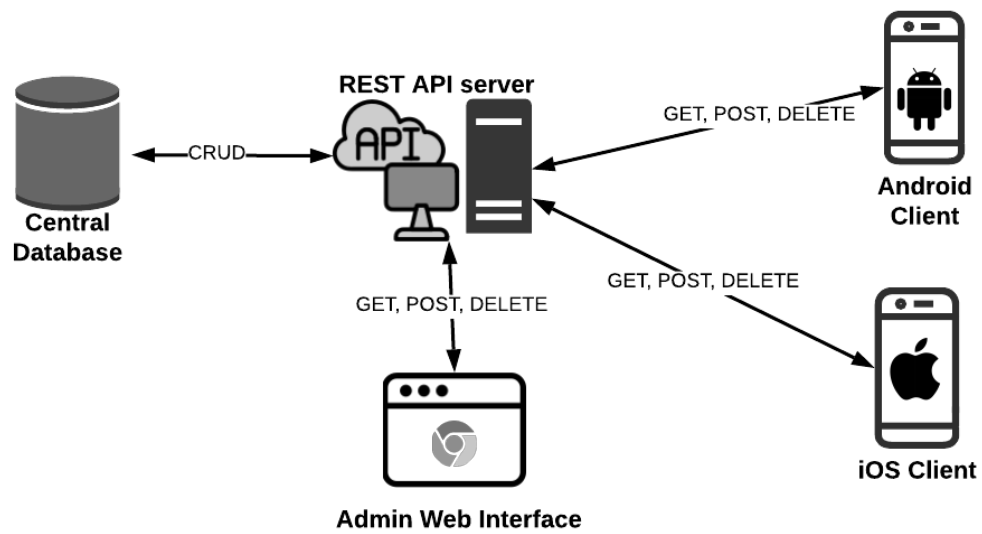
Figure 5: Technical architecture of the application

At the heart of the architecture lies the RESTful web service which communicates directly with the central database where all the data is stored. The mobile applications as well as the admin web interface do not access the database directly, but via the API service. The clients send HTTP requests like GET, POST and DELETE, while the API service processes those requests and return the data in JSON format.

3.3 Technologies Used

Table **??** consists of the major technologies that are proposed to be used during development and deployment of the application. They are briefly described in the subsections that follow.

Table 4: Technologies proposed to be used

Subject	Tools and Technologies Used
Backend Database	MySQL
REST API Service	Django REST Framework; Postman
Android Application	React Native; Android SDK
Android / iOS application	React Native; iOS SDK; XCode
IDE / Code Editor	Sublime Text; JetBrains WebStorm
Version Control System	GitHub
Documentation	LaTeX

3.3.1   MySQL

MySQL is one of the most widely used relational database management system (RDBMS). The major reason for us to choose this database is our previous familiarity with the database and it being completely open source. Some of the established tech corporates like Facebook, Twitter, Youtube etc. use MySQL. It also has very good user base and the usage is easier thanks to comprehensive documentation and support.

3.3.2   Django REST Framework

Django REST framework is an open source framework for building Web APIs. The framework provides features like authentication, tokenization, session handling, serialization, etc. for users to build API in short amount of time. The programming language used is Python. The reason for choosing this framework is the team members previous familiarity and experience in working with Python programming language.

3.3.3   Postman

Postman is an application software that is used for testing API web services. It lets users provide URL, parameters, body and headers to send the request and also shows the response in raw as well as pretty form.

3.3.4   React Native

React Native is an open source framework for building mobile applications using Javascript language. In addition to the usage of widely used language like Javascript, it also has cross platform support so that both Android and iOS applications can be built using same code base. Compared to its immediate rival – Flutter – it uses relatively easy and familiar language as compared to Dart used by Flutter. We chose this framework due to lack of time to learn Dart language and our familiarity with JavaScript language.

3.3.5   Android SDK

Android Software Development Kit is a set of build tools and libraries developed by Google Inc. to let developers develop applications in the Android platform. React Native also uses the Android build tools to eventually compile the application code and install it on the android device.

3.3.6   iOS SDK

iOS Software development Kit is a set of libraries and build and debugging tools developed by Apple Inc. to allow developers to develop applications and software in the iOS platform. React Native requires iOS SDK in order to build the compiled iOS application that could run on an iOS device.

3.3.7   XCode

XCode is a proprietary Integrated Development Environment (IDE) for macOS, primarily used for developing applications for the iOS and Mac platforms. XCode is frequently required to change native code for asking permissions, interacting with device sensors, installing third party APIs with keys, etc.

3.3.8   Sublime Text

Sublime Text is a light-weight cross-platform code editor written in C++ and Python. It is very popular among the developers due to ability of customization and plugins. This editor was chosen for development of backend due to our early experience working on it.

3.3.9   JetBrains WebStorm

JetBrains WebStorm is a commercial and proprietary IDE developed by JetBrains for JavaScript development. This IDE was chosen because of it's amazing code completion feature and easier debugging.

3.3.10   GitHub

GitHub is a platform for hosting and sharing software development version control by using Git. Github was acquired by American company Microsoft Corporation in 2018. The reason for using this platform was our early experience with it.

3.3.11   LaTeX

LaTeX is widely used documentation preparation system for preparation of scientific documents, books and technical papers. It uses plain text for formatting unlike other document creation systems. The source code is compiled by a compeller to generate the printable/viewable document. The reason for using LaTeX for documentation was to learn this new form of documentation.

3.4   APIs and Libraries Used

In course of developing this project, we have integrated several open source as well as proprietary APIs and libraries into our application. Some of the major APIs and libraries used in this project are listed in Table **??**.

Table 5: List of APIs and libraries used in the project

S.N.	Name	Author / Developer	Used For
1.	Google Maps SDK (Android / iOS)	Google Inc.	Showing Google maps inside our application for users to view treasures.
2.	React Native Paper	Callstack	Material Design UI library for developing Android and iOS application
3.	react-native-maps	React Native Community	Integration of Google Maps / Apple maps inside of mobile application
4.	react-navigation	React Navigation	Navigating users from one screen to another inside mobile application
5.	redux	Dan Abramov and Andrew Clark	State management in mobile application
6.	react-native-camera	React Native Community	Access mobile device's camera and scan QR codes
7.	react-native-vector-icons	Joel Arvidsson	Showing vector icons at various places in mobile application
8.	react-redux	Redux	Using redux library inside of a React Native application

3.5   Method of Data Collection

Our team has collected data about the various tourist destinations and places (the data include the location, the photos, latitude, longitude, how to get to that place, entry fee, etc.) so that we can add treasures to those places and add them to our database.

In this project, we have used secondary source of data collection. We did not go to those tourist attractions by ourselves but collected the information about them through the use of Internet. The latitude and longitude information were collected by the help of Google Maps, while the address, description, etc were collected from various tourist blogs and websites. As a result, the collected data are not quite reliable and are used only for demo purpose. For more reliability, we can use primary source of information by directly going to thos places and collecting data.

Figure **??** shows a sample of data collected by our team.

S.N	Destination Name	Category	Latitude	Longitude	Description	District	Points
1	Patan Durbar Square	Durbar Square	27.6727352	85.3231056	Patan Durbar	Lalitpur	3
2	Krishna Mandir	Temple	27.673603	85.3227425	The Krishna te	Lalitpur	2
3	Mahaboudha	Temple	27.6685903	85.3250747	Mahabuddha	Lalitpur	5
4	Hirenya Verna Mahavihar	Temple	27.6752289	85.3223596	Hiranyavarna	Lalitpur	5
5	Kumbheshwor Jagatnarayan Temple	Temple	27.6788188	85.3285167	On the bank o	Lalitpur	3
6	Rudra Varna Mahavihar	Temple	27.6788423	85.3219506	This is one of	Lalitpur	3
7	Ashok Stupa	Temple				Lalitpur	5
9	Machhendranath and Minnath Temples	Temple	27.669063	85.3246662	Just 200m soutl	Lalitpur	10
10	Centra Zoo		27.6727087	85.3096407		Lalitpur	6
11	Patan Industrial Estate		27.6615469	85.3233106		Lalitpur	2
12	Bajra Barahi	Temple	27.6060558	85.3271305		Lalitpur	4
13	Godawari Botanical Garden		27.5969371	85.3779545		Lalitpur	15
14	Phulchoki		27.6354045	85.3779545		Lalitpur	30
16	UN Park	Park	27.6855826	85.3240775		Lalitpur	2
17	Sundari Chowk		27.6727215	85.3229108		Lalitpur	3

Figure 6: Model of data collection

3.6   Team Members and Role Division

TODO: EDIT THIS

4. Requirement Analysis

The requirement analysis of the product to be developed was done before everything else during the project. During this phase, our team worked together to find out what features were expected of the product to be developed. It was also helpful to filter what is important and what is not important features to be added in the application. The requirements were widely categorized into functional and non functional requirements. The process we used during this phase are described in the sections that follow.

4.1 Requirements Elicitation

Requirement elicitation is the process of collecting and noting down several types of requirements that are expected of the product from various sources like development team, the consumer, users, experts, etc. Prioritizing the collected requirements is also an important task done in this phase. Elicitation is the first step in developing the requirements documentation in any software project. In this project, the major sources for eliciting the requirements where the team members and the supervisor assigned to the project team.

During requirement elicitation, the team members conducted elicitation meetings with the supervisor to find out and note out requirements from 'absolutely necessary' to 'desirable'. During the process, each of us acted as an end user and described what an end user would expect the product to do. We would then discuss about whether the requirement is necessary or not, whether it is feasible to implement in our project and then note them in neat and tidy way. The team members then categorized the requirement into several categories that are described in the subsections that follow.

4.1.1 Functional Requirements

Functional requirements are those requirements which define a system or a component by the functions it should perform. These are the requirements that are absolutely necessary to be in the product. The functional requirements of our project are listed in Table **??**.

Table 6: List of functional requirements of the project

S.N.	Functional Requirements	Priority
1.	The user should be able to download and install the application in a Android Phone or iPhone	Very High
2.	The user should be able to register his/her account in the application and login at any time with valid credentials	Very High
3.	The user should be able to view treasures in a map and get information about the treasures.	High
4.	The user should be able to view rewards that he/she is eligible to collect and read their details.	High
5.	The user should be able to scan a treasure and have the points collected in his/her account.	High
6.	It will be possible to scan a treasure once and only once and only when the user is in vicinity of the treasure's location	Very High
7.	The points provided to a particular treasure should be practical and dependent on factors like reachability, cost, etc.	High

4.1.2 Non-Functional Requirements

Non functional requirements are those which describe quality attributes in a system. These are the requirements that are not absolutely necessary, but are desirable for good quality of the proposed product. The non-functional requirements of our project are listed in Table **??**.

Table 7: List of non functional requirements of the project

S.N.	Non Functional Requirements	Desirability
1.	The permitted distance from the location of the treasure up-to which the collection of treasure is allowed could be specified and entered into database	High
2.	The users should be able to login via email as well as Google and Facebook	Medium
3.	The user will be able to share the application via email or social media to invite other people to play	High
4.	The rewards offered to the users could be provided on the basis of user's personal preference.	Medium

4.2   Requirements Specification

The most widely used tool during the requirement analysis phase in our project was the use case descriptions and diagrams. The project team members first described the actions a user would perform on the system during a particular scenario, and then it was converted to diagrammatic form.

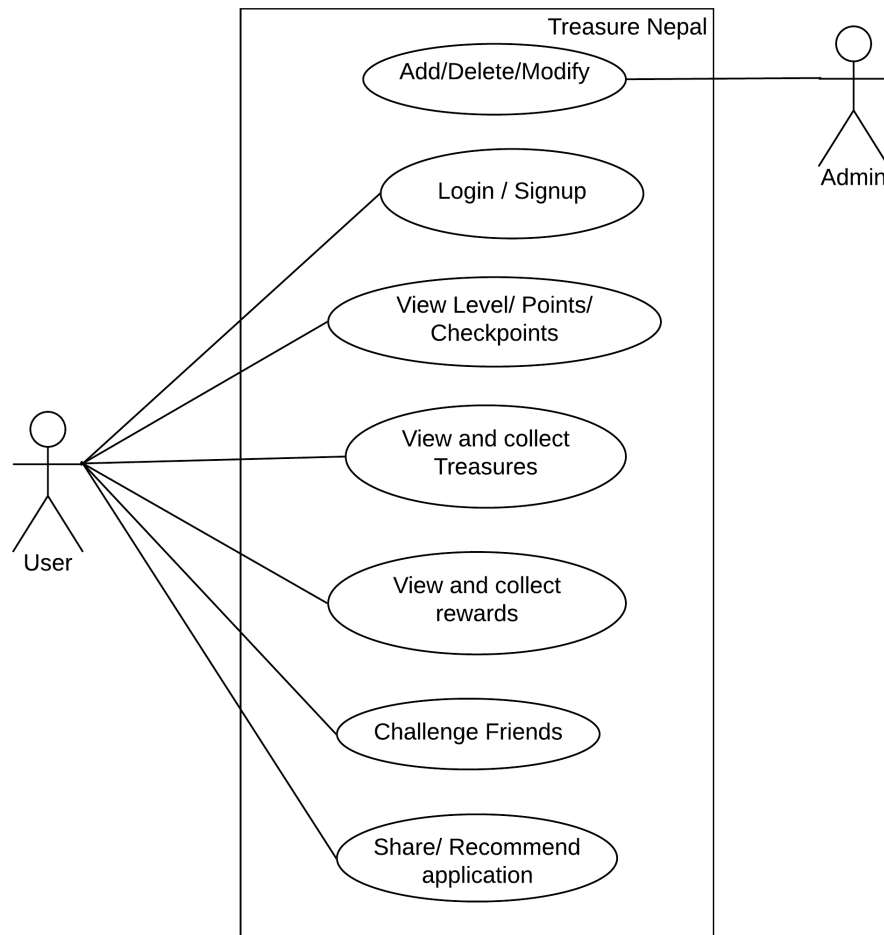| **Use Case: Use Application** |
|---|
| When the user first installs the app, she should be able to sign up for a new account. For so, the user can either use email address or use Facebook and Google for authentication. Once the user registers her account, she will enter her credentials and login to the application. The user now can see her points (which will be 0 at the start), and all the treasures around her location. She will also be able to get the information about the particular treasure. To collect the treasure, the user will physically have to reach the location where the treasure is installed. When the user checks in at the treasure location, she will be awarded with the points associated with that treasure. The user will be able to see different rewards and offers that she is eligible to collect. She will be able to collect any of the permitted rewards by scanning the QR provided by the offerer. The user can challenge other friends as well as share the app in the social media. |

Figure 7: Use case diagram for overall application

This use case is prepared at the highest level of abstraction – that is when the user uses the application as a whole. After a series of in depth analysis, we figured out that the two major parts in our application are the collection of treasures and rewards. So, we prepared use cases each for how a user would collect them.

Use Case: Treasure Collection

After being logged in, the user can view all nearby treasure locations in a map. The user can also search treasures using name and location. The user can also view information about the place she will be visiting in advance, and about the treasure available at that location. All the data related to the treasure are added and updated by an admin who has access to the database. When the user physically reaches the location where the treasure is installed, she uses the application to check in at that place. Based on the location of the place, the validity of collection is determined and the scores are attributed to the user. The user can write reviews about a particular treasure location and share it. She can also recommend to add some new places to the treasure database, which will be reviewed by the admin team.
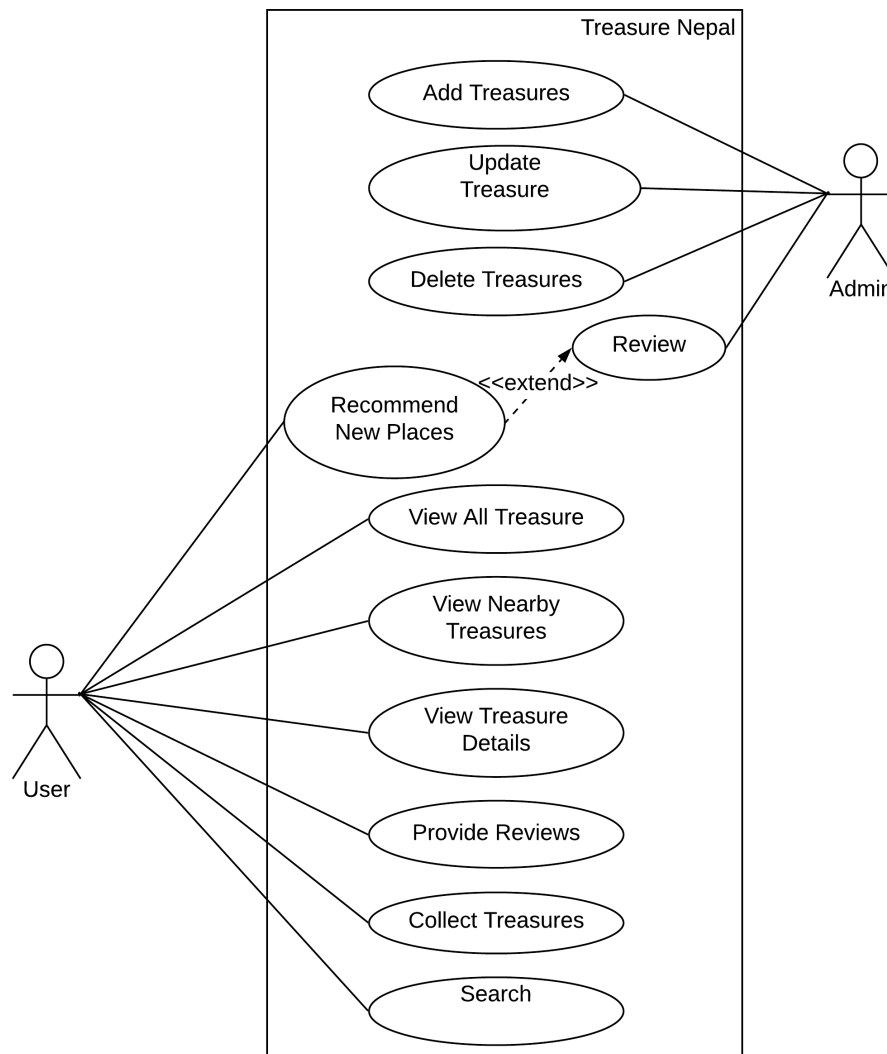
Figure 8: Use case diagram for treasure collection

Use Case: Reward Collection
The application will have a number of rewards and offers from different business partners like hotels, resorts, restaurants, etc. The eligibility of a user to claim these rewards will be based on their points. After the points cross a certain level, several of the rewards will be unlocked. To redeem those awards, the user have to visit the reward location, and scan QR code at that location. An admin will be responsible for adding and updating the rewards and offers in the central database. She will also be able to provide reviews and ratings to the rewards she collects.
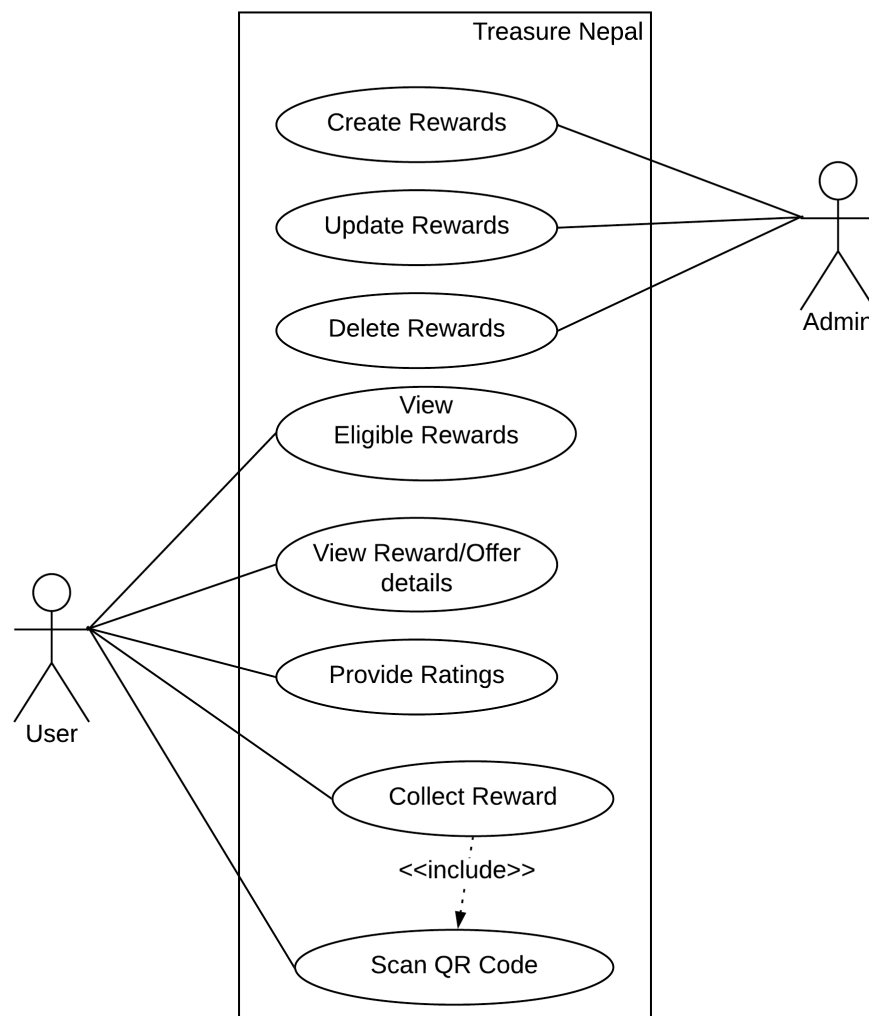
Figure 9: Use case diagram for reward collection

4.3   Requirement Validation

5. Design

5.1 Database Design

The database used for the API backend in the project is MySQL. Since the mobile applications do not access the data directly from the database, the API server is responsible for fetching the data in accordance to the API query and send it to the mobile device.

The following class diagram illustrates the database schema used for the application.
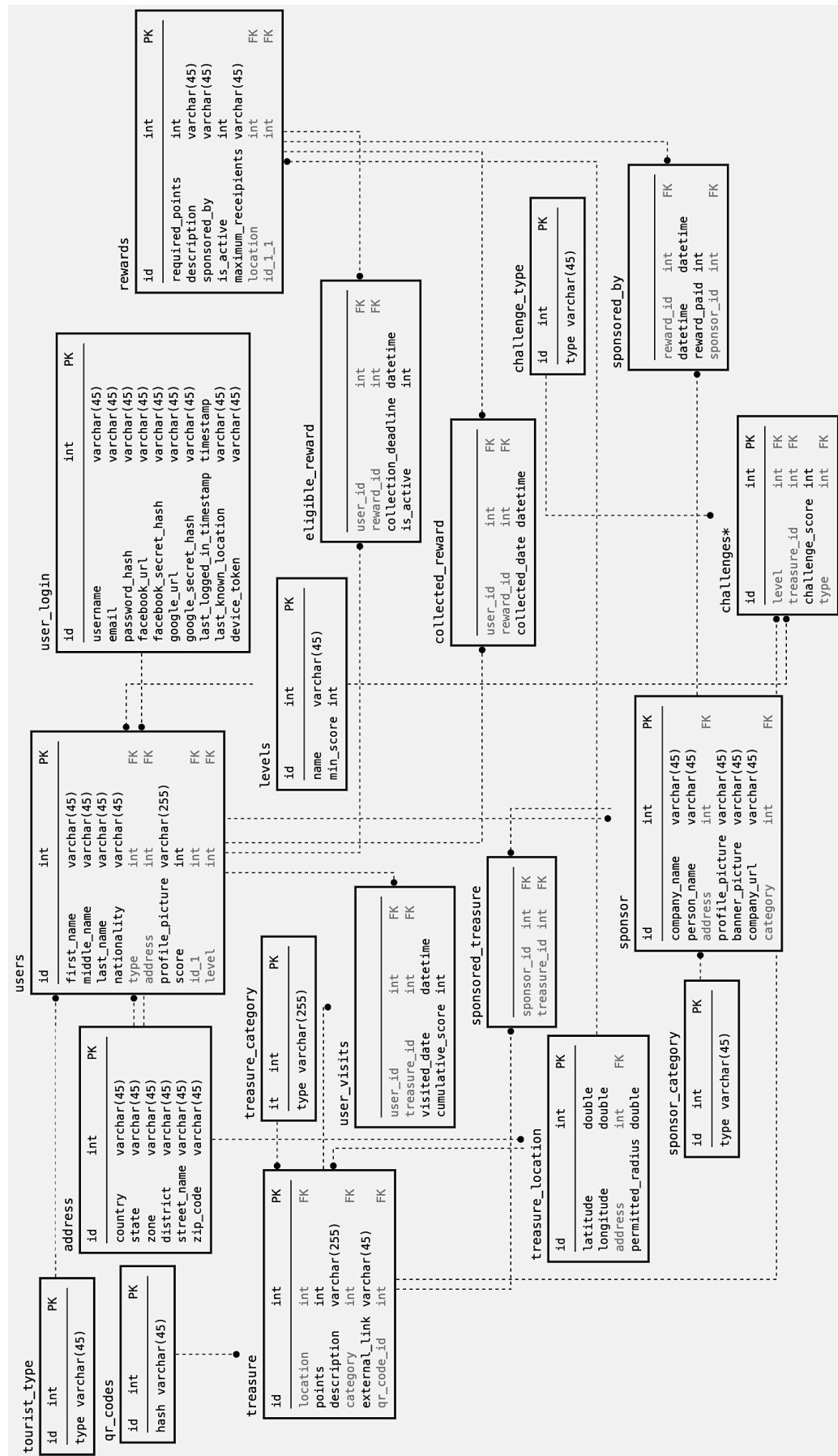
Figure 10: Schema used in backend database

6. Coding and Implementation

6.1   Backend API

The API endpoints for creating, updating, reading and deleting the data from all of the tables shown in Figure **??** have already been created. The following table shows the syntax of different endpoints of the API.

Endpoint	Method	Body	Response
/{table_name}	GET	-	List of all data entries in the table
/{table_name}/{id}	GET	-	The entry in the table corresponding to provided ID
/{table_name}/	POST	data object	Entry for data object created in the database
/{table_name}/{id}	PUT	data object	Update the entry with provided id with new data object
/{table_name}/{id}	DELETE	-	Delete the entry corresponding to the provided id

The following figure shows an instance of API request and response.
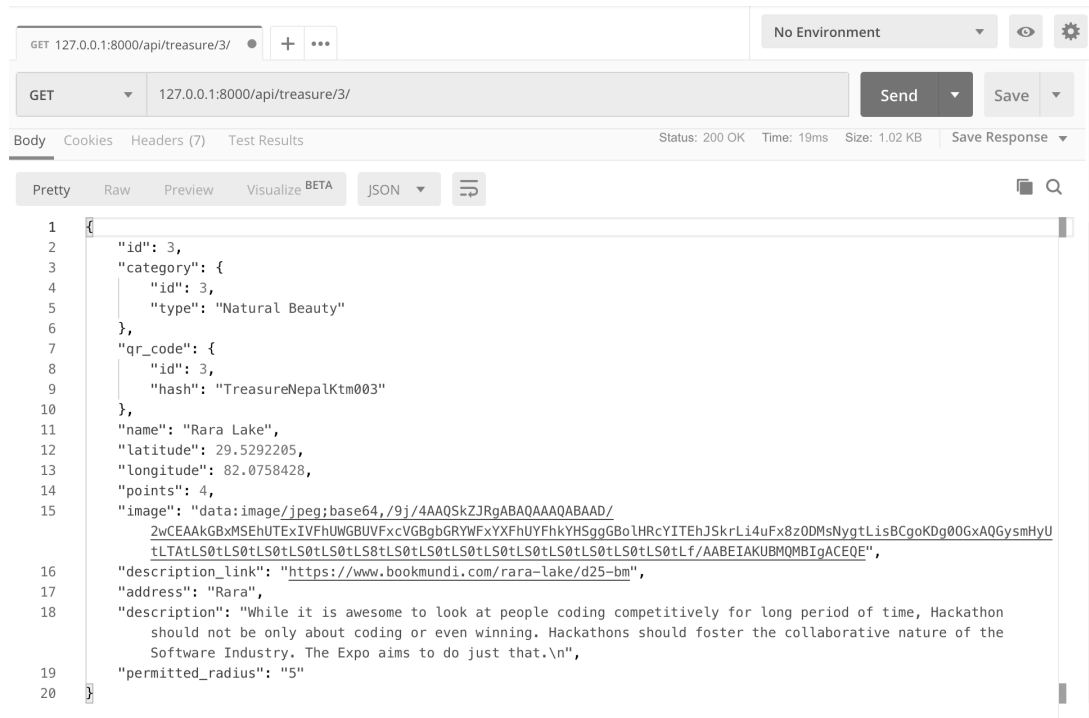
Figure 11: Screenshot of API request and response using Postman

6.2 Authentication Module

The user authentication module has been completed in the backend, using email, Facebook as well as Google. In the backend, we have use django-rest-auth module for authentication management. This gives the user a unique api key whenever she logs in to the application. The user should then include that key in every request she sends to the API server.

The following sequence diagram shows how a user authenticates with the application.
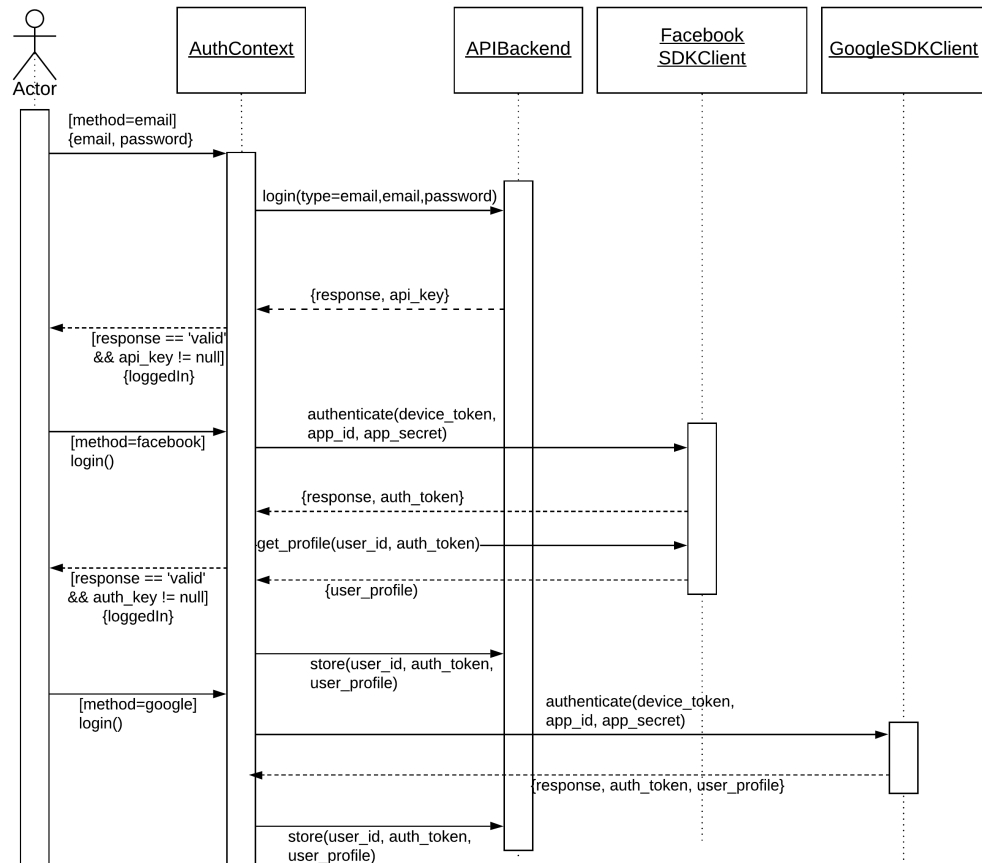
Figure 12: Sequence diagram for user authentication

6.3   Application Development

The authentication has been integrated in the application development and as of now, it is fully working in both Android and iOS devices. The user can successfully login and out either using email, Facebook or Google sign in options.

The following figures show some screenshots of the application developed so far.
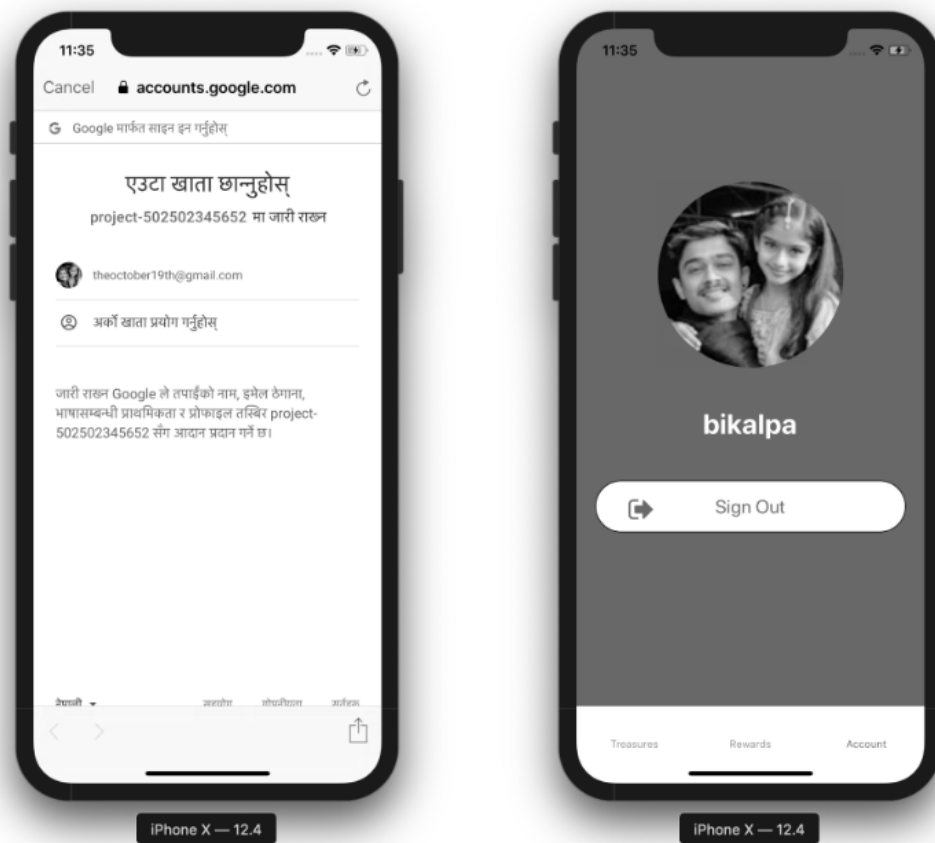
Figure 13: Screenshots of the application developed

7.   Testing and Debugging

8.   Deployment

34 / ??

9.   Project Task and Time Schedule

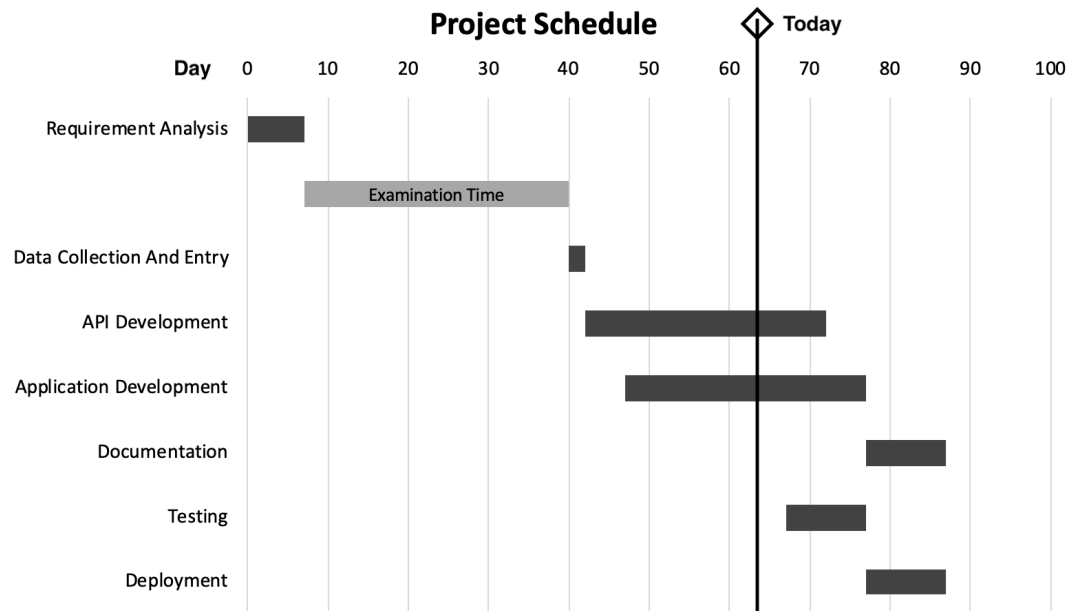The time schedule of the project is illustrated in Figure **??**.

Figure 14: Proposed project schedule

10. Conclusion

At the point of submission of this document, the project is almost 65% complete. The backend part is almost complete while some amount of task is remaining in application development part. After analyzing the project schedule submitted earlier during proposal defense, our project is right on its proposed time schedule.

At this point in time, the project team has learnt a lot about the cross platform development and the basics of RESTful services as compared to when the project was just started. At first, the project team had used SQLite as the database due to its small size and easier integration and we later switched to the MySQL database. We had initially used SQLite only for test purpose. However during the migration of database we had to tackle a lot of portability issues and also required to re-enter the data to the database. We also had encountered problems while integrating third party authentication service (like Facebook and Google) to our project, partly because we were configuring our project to run both on Android and iOS devices simultaneously. However we were able to solve to the problem with team work and mutual discussion.

11.   Further works and Recommendations

At the point of submission of this document, the project is almost 65% complete. The backend part is almost complete while some amount of task is remaining in application development part. After analyzing the project schedule submitted earlier during proposal defense, our project is right on its proposed time schedule.

At this point in time, the project team has learnt a lot about the cross platform development and the basics of RESTful services as compared to when the project was just started. At first, the project team had used SQLite as the database due to its small size and easier integration and we later switched to the MySQL database. We had initially used SQLite only for test purpose. However during the migration of database we had to tackle a lot of portability issues and also required to re-enter the data to the database. We also had encountered problems while integrating third party authentication service (like Facebook and Google) to our project, partly because we were configuring our project to run both on Android and iOS devices simultaneously. However we were able to solve to the problem with team work and mutual discussion.