

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 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 various 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 also share their scores and leaderboard status to social media. On the long run, the project sets its objective to take the tourism of Nepal to a new level.

This report has been prepared to demonstrate the progress the team has made after the acceptance of project proposal. It also includes major milestones that will be met in future and estimated date for deployment.

Keywords: Visit Nepal 2020, Treasure hunt, Tourism

TABLE OF CONTENTS

Abstract	i
Table of Contents	ii
1 Introduction	1
1.1 Problem Statement	1
1.2 Project Objectives	2
1.3 Significance of the Study	2
1.4 Scope and Limitations	2
2 Literature Review	4
2.1 Visit Nepal 2020	4
2.2 Attractions in Nepal	4
2.3 Existing Similar Applications	5
2.4 Challenges	5
2.5 Cross Platform App Development	5
2.6 REST API	6
3 Methodology	7
3.1 Proposed Software Development Life Cycle	7
3.2 Technical Architecture	8
3.3 Technologies Used	8
3.3.1 MySQL	8
3.3.2 Django REST Framework	9
3.3.3 React Native	9
3.3.4 Amazon Web Services (AWS)	9
3.3.5 LaTeX	10
4 Tasks Done so Far	11
4.1 Requirement Analysis	11
4.2 Database Design	14
4.3 Data Collection	16
4.4 Backend API	16
4.5 Authentication Module	17
4.6 Application Development	18
5 Results and Discussion	21
6 Tasks Remaining	22
6.1 API Completion	22

6.2	Mobile Application Completion	22
7	Deliverables	23
7.1	RESTful API service	23
7.2	Android client application	23
8	Project Task and Time Schedule	24
	References	25

1. Introduction

Treasure Nepal is a mobile application for a treasure hunt game proposed to be tailored for tourists visiting 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 locations as well as promote their tourism. This document looks forward to providing detailed information about the project, the tasks that have been done so far, and the tasks that are remaining.

Tourism has a great potential to contribute to the Gross Domestic Product (GDP) of Nepal, but having observations at the statistics, the ratio of contribution of tourism to GDP is not satisfactory. According to Nepal Rastra Bank, the total contribution of the foreign exchange from tourism to the total Gross Domestic Product (GDP) of Nepal was 2.2% in the year 2017/18. [1].

The Government of Nepal has taken efforts to celebrate the year 2020 officially as the Visit Nepal Year 2020. At the brim of year 2020, the authors have proposed to build a treasure hunt application specially tailored for the tourists to get their attention to unexplored and remote tourism destinations of Nepal.

There are a few applications somehow similar to the proposed application but none of them are built for tourism purposes. Rather, they are developed only for entertainment purposes. In this context, this application will distinguish itself as unique and first of the kind product in the market.

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.

1.1 Problem Statement

The decentralisation of tourism 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.

1.2 Project Objectives

The project has put forward the following objectives:

- To decentralize the tourism industry and encourage uniform flow of tourists at various destinations across Nepal.
- To promote and encourage the tourism in remote and novel destinations which otherwise are not popular or have low inflow of tourists.
- To explore business and economic opportunities generated by the project if taken to the production level.

1.3 Significance of the Study

The project is significant owing to the fact that we are near the Visit Nepal Year 2020, and it will certainly be fruitful in achieving the objectives set by the Government of Nepal in the year 2020. Since the idea behind is one of the first of its kind, it is expected that the project will reach to a significant majority of tourists that visit Nepal in 2020.

1.4 Scope and Limitations

In the beginning phase, the treasure hunt concept of the app will be implemented and other features are proposed to be added later gradually if possible. Such possible extensions could be addition of forex plugins, itinerary maps, guides, etc. The users will be able to collect coins from collecting the treasures, and their collection will be put in the leaderboards based on the user's local location as well as country-wise and globally. The application will have an integrated map, which enlists the tourist destinations in the locality of tourist's current location. The application will also be connected to third party social networking platforms like Facebook, Twitter, etc. so that the users can share their collection and score.

The scores of the treasures will be calculated based on the factors like the difficulty to reach the destination, its novelty, potentiality to attract new tourists and other similar criteria. The users will receive more amount of score when visiting rural and novel places than visiting urban and frequently visited places.

The following are the limitations of the project that are realized:

- The application will be built on Android and iOS platform, but not for other mobile operating systems like Blackberry and Windows.
- QR code scanning will be the method of collection of treasures and no other validation architecture will be used except for the check of location when the user scans the QR.

2. Literature Review

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

2.1 Visit Nepal 2020

The Visit Nepal 2020 project was officially introduced by Nepal Tourism Board (NTB) in 2015. The project aims to bring two million tourists in Nepal during the year 2020 [2]. The slogan of Visit Nepal 2020 has been translated to more than ten different languages. The board also is planning to train ten thousand people to provide quality service to the tourists.

2.2 Attractions in Nepal

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 [3]. These mountains serve as a destination for the tourists seeking mountaineering as wellll as trekking. The famous trekking destinations in Nepal are Everest Base Camp, Annapurna Circuit and Langtang Trekking Route [4]. 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 [5]. 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 rich in cultural diversity too. A large portion of the tourist inflow in Nepal occurs for religious and pilgrimage purposes. In recent years, the various communities have incorporated home stay programmes 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 Existing 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 [6]. This application distinguishes itself from these applications due to its focus on the tourism industry. The available applications are developed only for entertainment purposes. 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.

2.4 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.

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. 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 [7].

2.5 Cross Platform App Development

Android and iOS are two major mobile development platforms in the market today. Any application that is developed targeting both of these platforms is known as a cross platform application. 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 can be used by a web based portal, an Android mobile phone, an iPhone, iPad, or even a desktop computer software.

When cross platform applications are needed to be developed, the two most popular frameworks are Flutter and React Native. React Native is relatively older technology first released in 2013. Currently it is managed by Facebook. Flutter, a new technology, is the similar framework from Google. React Native uses JavaScript as its language while Flutter uses Dart. 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.

2.6 REST API

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 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.

API stands for Application Programming Interface. It is a kind of abstract interface or a communication protocol between a client and a server to ease client-server communication. An API can be Web based, OS based, or for a database management system. Web based APIs are those which provide the interface between the client and the database. 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).

3. Methodology

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

3.1 Proposed Software Development Life Cycle

The project has been developed as per the waterfall model of software development life cycle as depicted in Figure 1. 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.

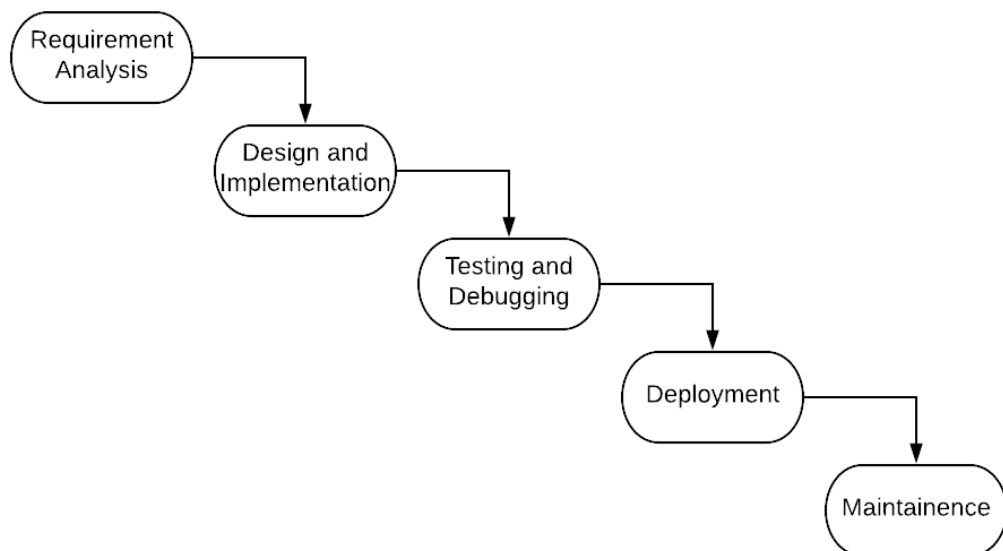


Figure 1: Proposed software development life cycle

The life cycle begins when the team collects and evaluates the requirements that will be expected from the application. The design and implementation phase will be to design and build both API services and client applications. By the end of this phase, a minimal viable product (MVP) will already have been constructed. In the testing and debugging phases, the quality control methods will be applied to both API and application. Finally, the application will be deployed to the Amazon Web Services at the end of the deployment phase. However, there might be slight modifications in the original waterfall model where the design and implementation may be changed slightly after the testing phase if seen reasonable.

3.2 Technical Architecture

The application has been built upon the client-server web architecture, as illustrated in Figure 2.

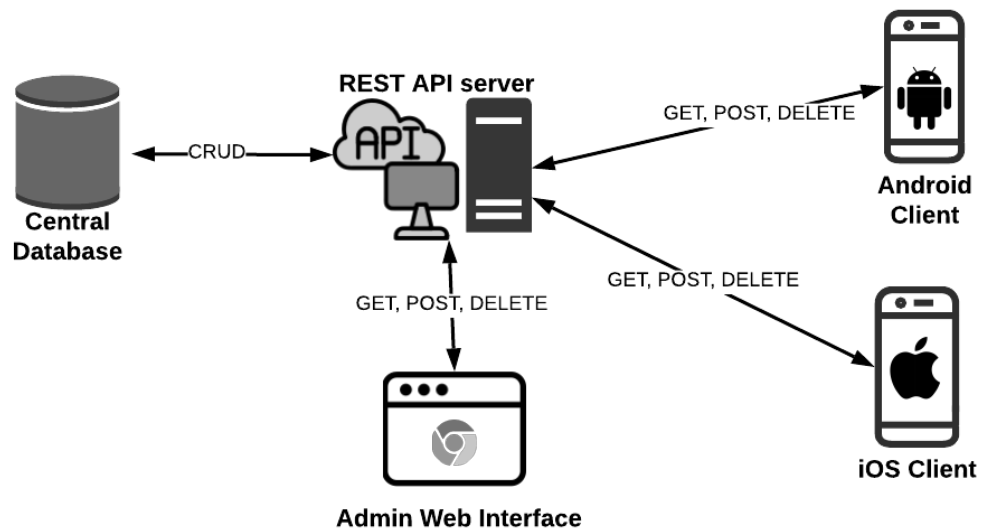


Figure 2: Technical architecture of the application

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 1 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.

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 that it is completely open source. Some of the established tech corporates like Facebook, Twitter, Youtube etc. use

Table 1: Technologies proposed to be used

Subject	Proposed Technology
Backend Database	MySQL
REST API Service	Django REST Framework
Android Client	React Native
Deployment Platform	Amazon Web Services (AWS)
Documentation	LaTeX

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 language used is Python.

3.3.3 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.

3.3.4 Amazon Web Services (AWS)

Amazon Web Services (AWS) is a common cloud platform that provides cloud based services to individuals, governments and corporates on the pay-as-you basis. It is a subsidiary of Amazon Inc.

3.3.5 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.

4. Tasks Done so Far

This section enlists the major breakthroughs achieved during the project. Among the total works proposed in the proposal, about two thirds of the progress have already been achieved and the foundation for the next one third have also been laid out.

4.1 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 classified into two categories: functional and non functional requirements. Functional requirements are those requirements which define a system or a component by the functions it should perform. Non functional requirements are those which describe quality attributes in a system.

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.

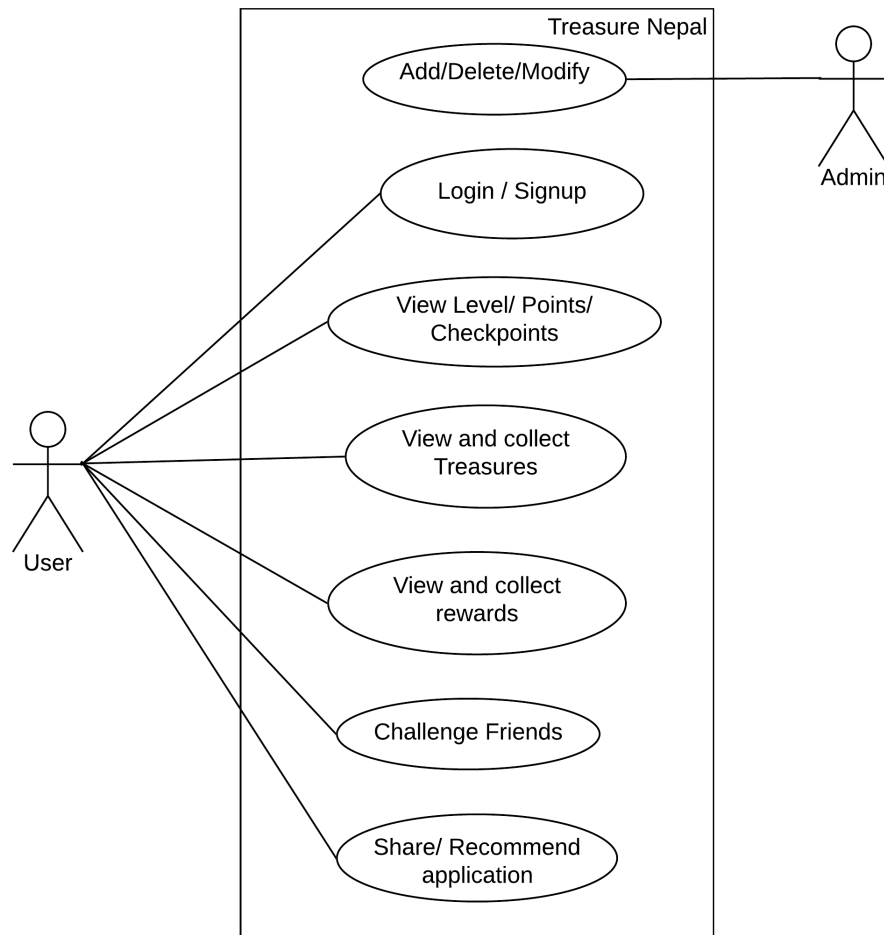


Figure 3: Use case diagram for overall application

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.

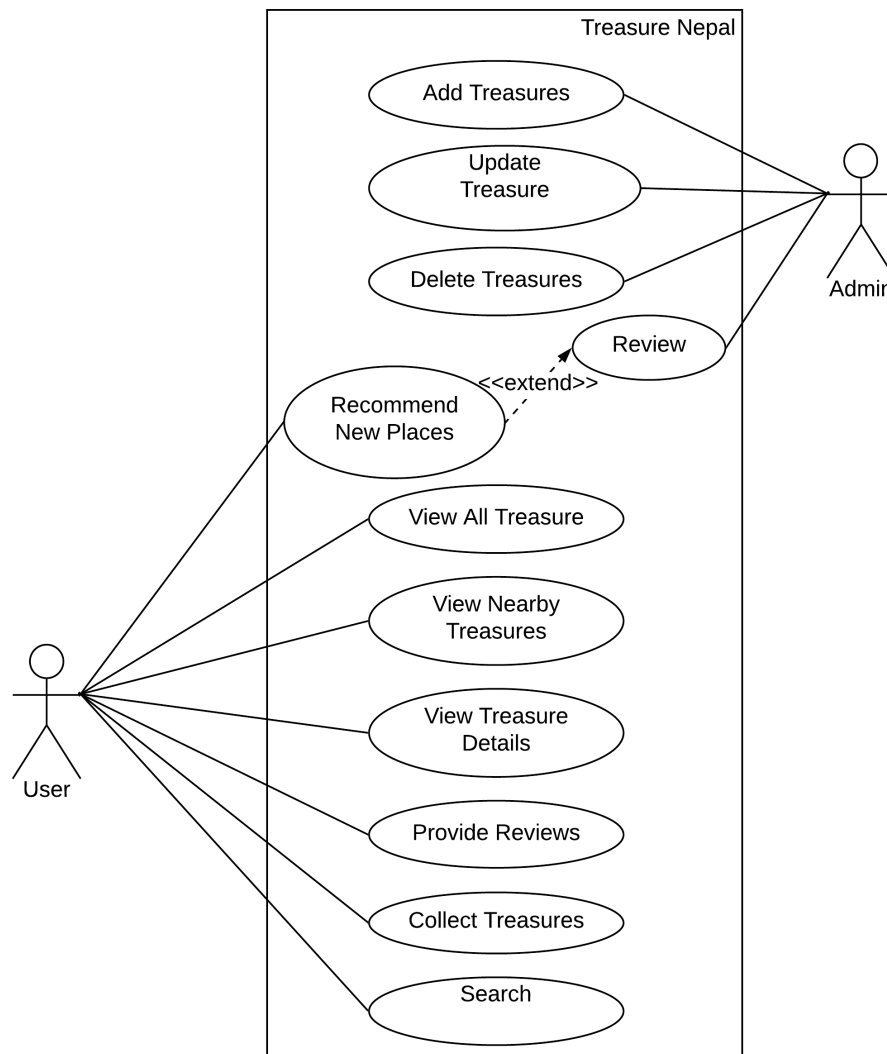


Figure 4: Use case diagram for treasure collection

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.

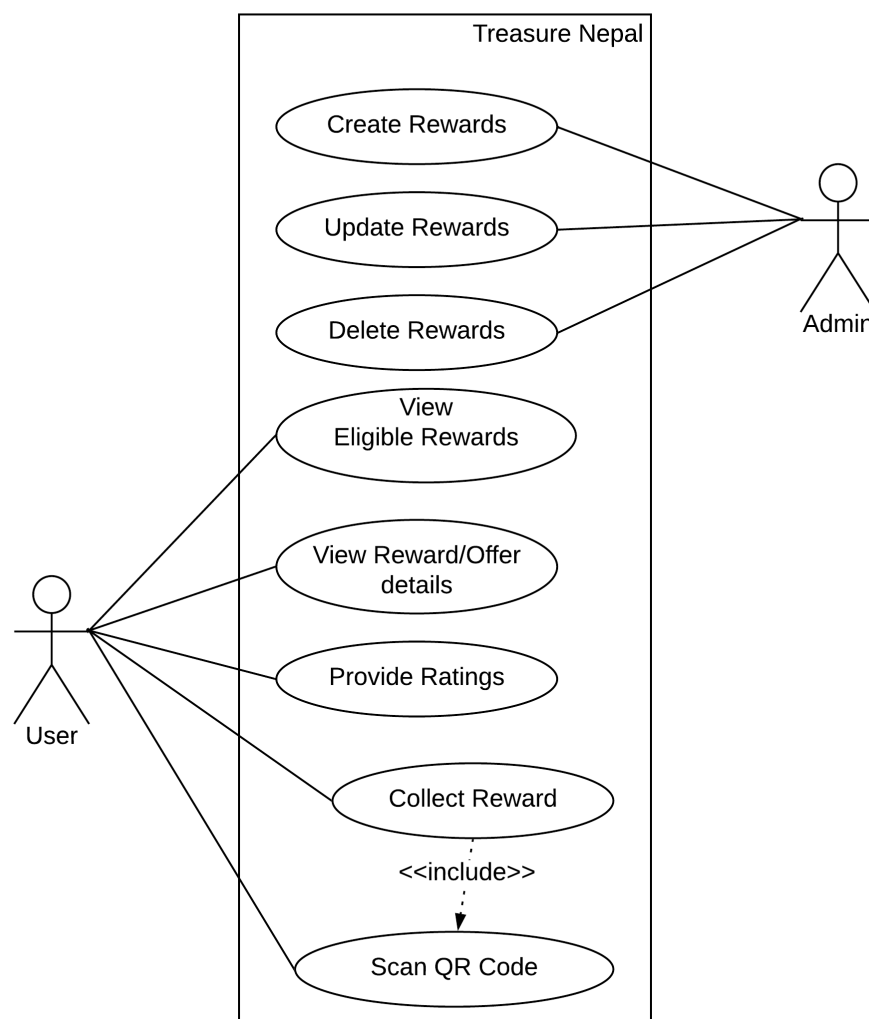


Figure 5: Use case diagram for reward collection

4.2 Database Design

The following class diagram illustrates the database schema used for the application.

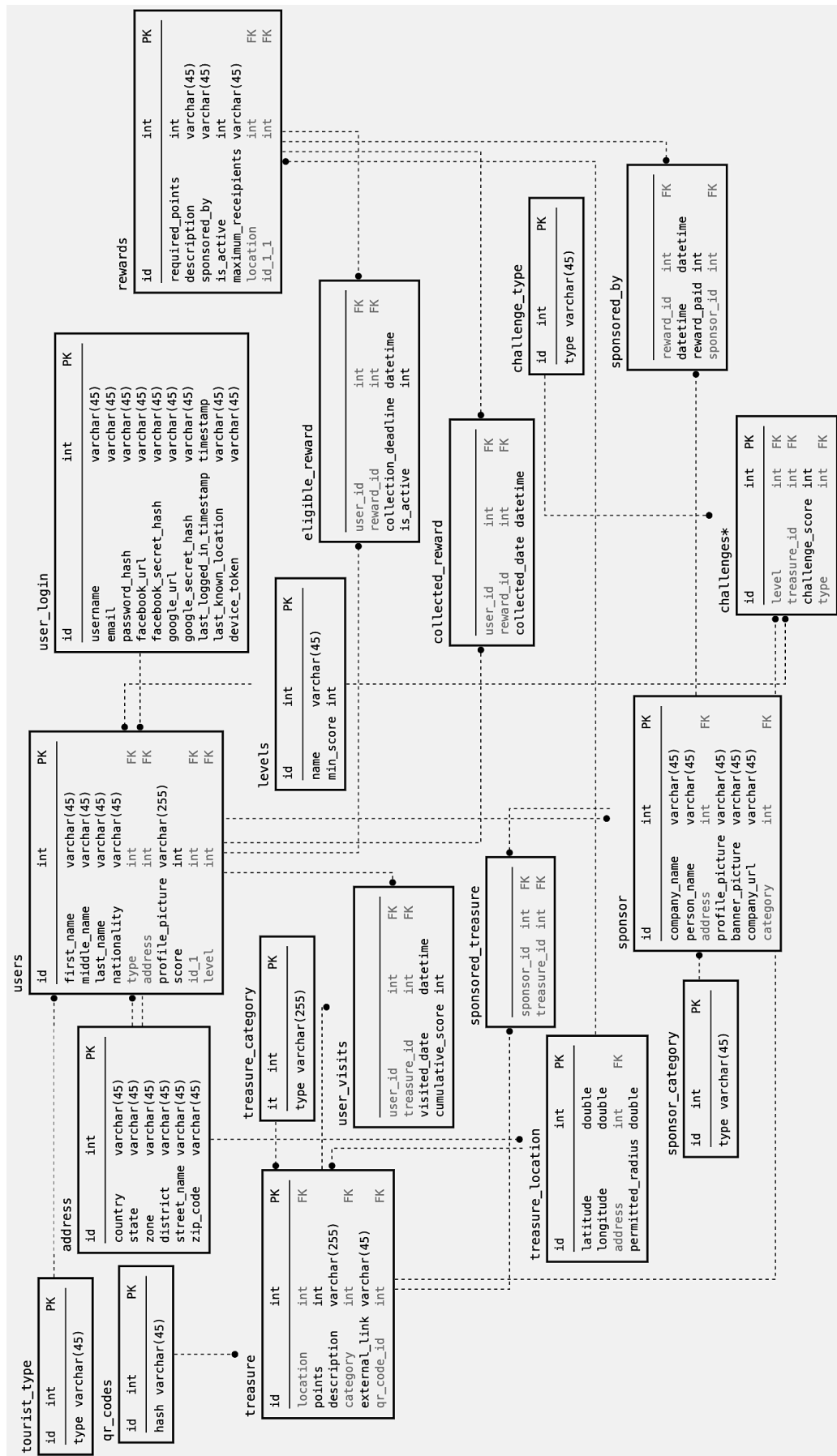


Figure 6: Schema used in backend database

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.

4.3 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.

The following 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 south	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 7: Model of data collection

4.4 Backend API

The API endpoints for creating, updating, reading and deleting the data from all of the tables shown in Figure 6 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.

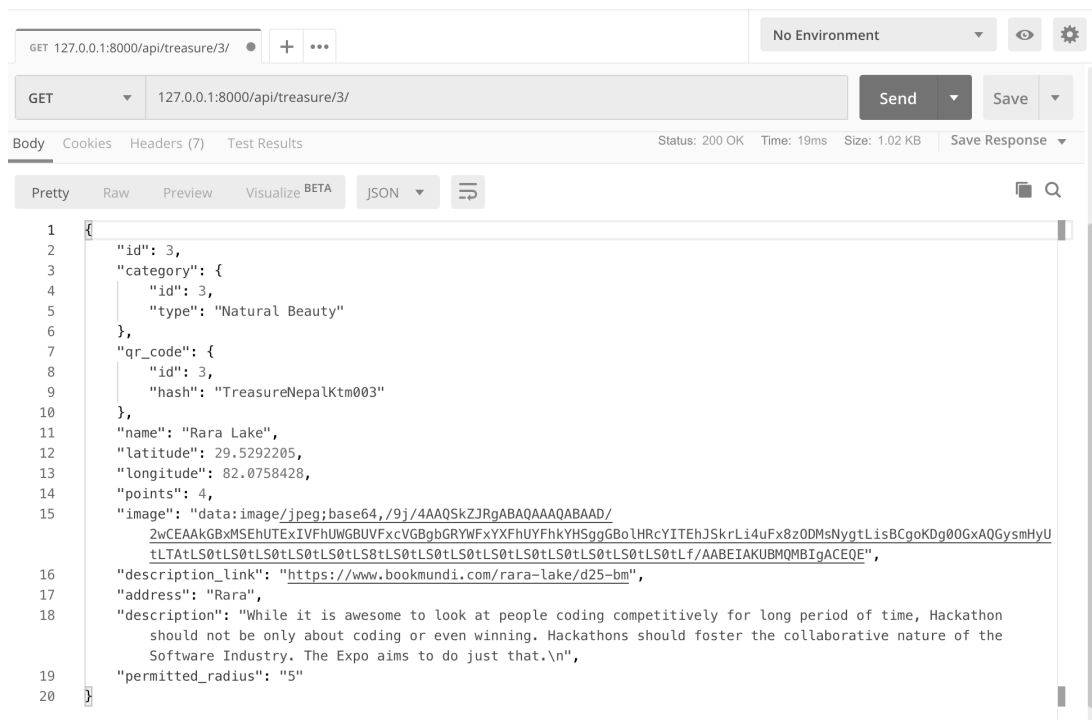


Figure 8: Sequence diagram for user authentication

4.5 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.

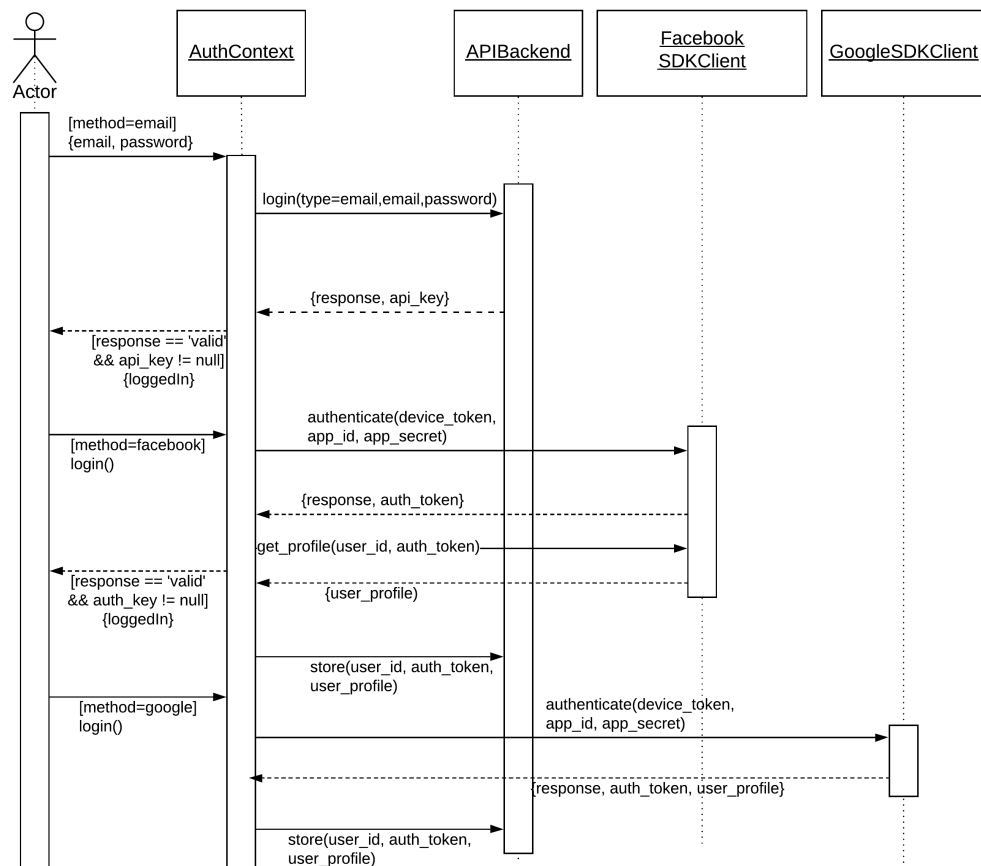
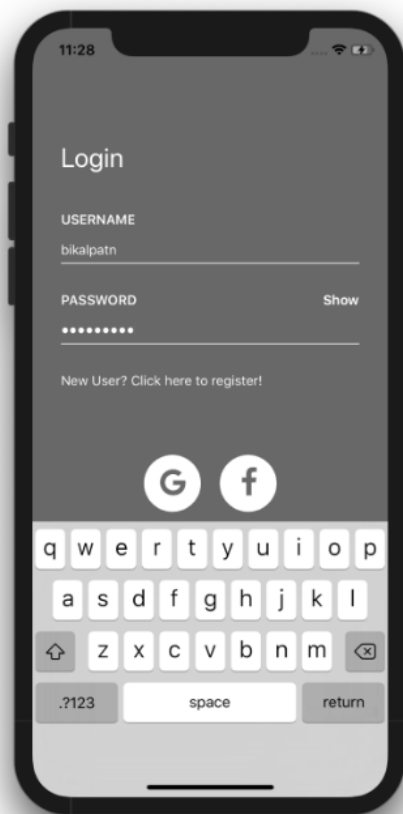


Figure 9: Sequence diagram for user authentication

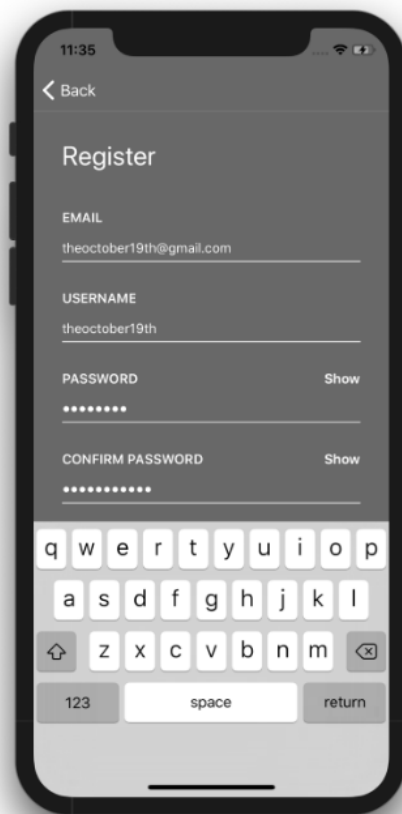
4.6 Application Development

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.

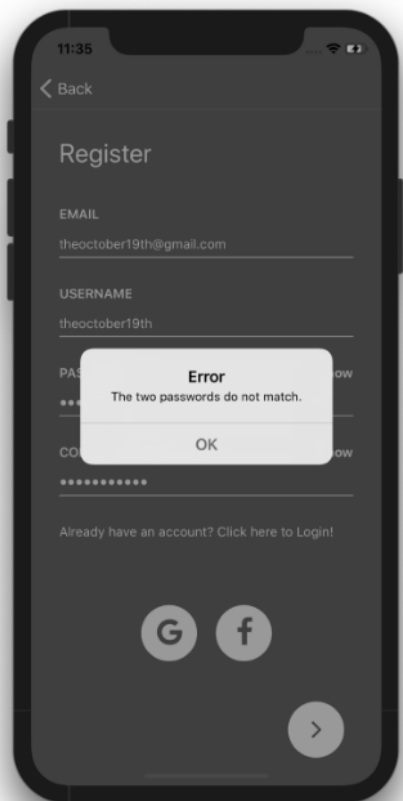
The following figures show some screenshots of the application developed so far.



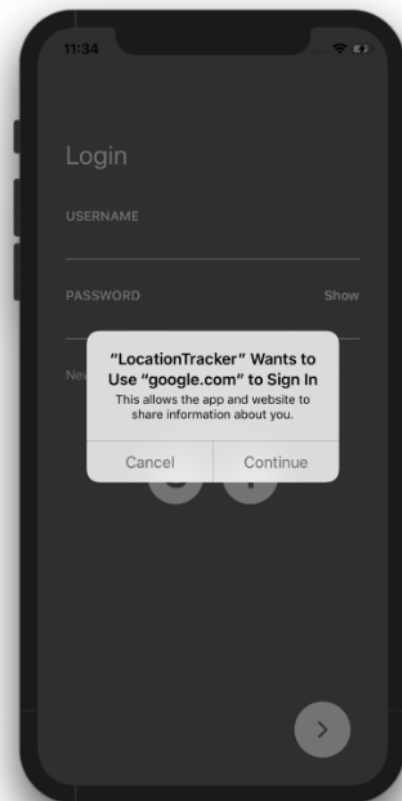
iPhone X — 12.4



iPhone X — 12.4



iPhone X — 12.4



iPhone X — 12.4

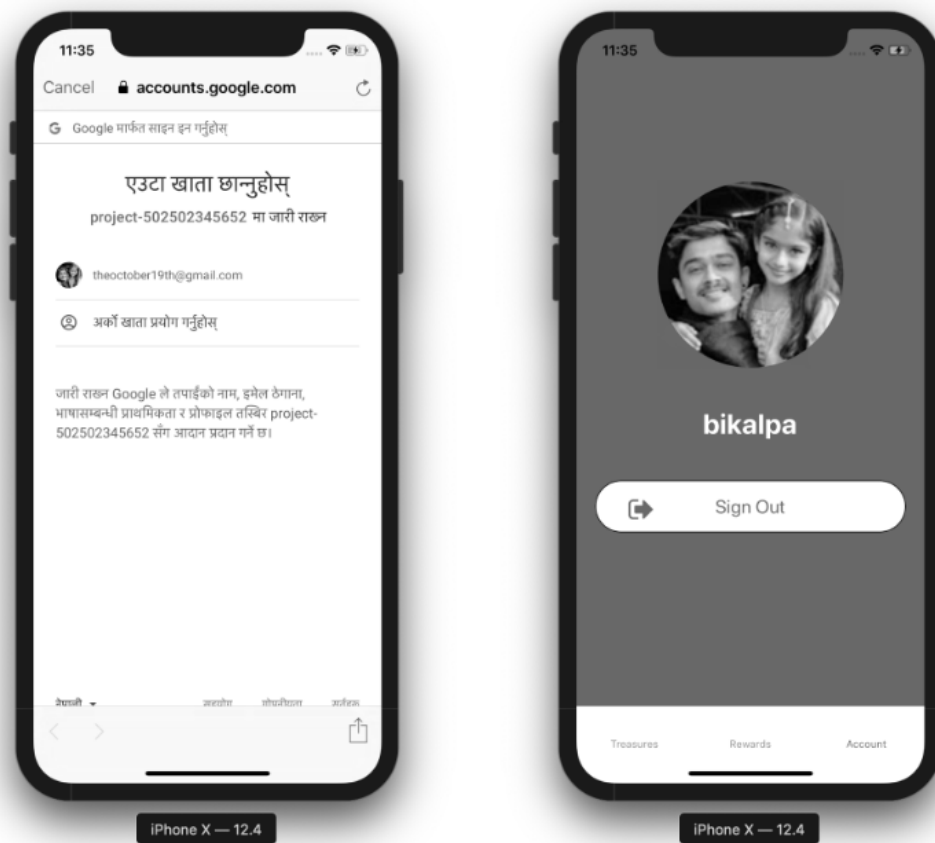


Figure 10: Screenshots of the application developed

5. Results and Discussion

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 analysing 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.

6. Tasks Remaining

This section enlists the tasks that are yet to be done before the final defense of the project.

6.1 API Completion

Although the authentication module is already working fine, we need to incorporate JWT (Java Web Token) model to authenticate every request to the backend API. Since we have already written logic for the generation of the key, the only thing remaining is to add a parameter called 'key' to every request, and then validate that key by API server before sending response to a client.

Similarly, the end points for the search queries have not yet been written. The API should be updated with provision of search queries for treasures as well as rewards. After completion of these two features, the backend API part of our project will be complete. On overall, about 80% of work has been completed in the backend portion.

6.2 Mobile Application Completion

The mobile application part of the project has been completed to about 50% of the total work required. Right now, the authentication module is working fine, which needs slight modifications for incorporating the JWT tokens mentioned earlier. The major work remaining in the application portion is to show treasure locations in the map and then write logic to collect those treasures when the user checks in using their app at that location.

7. Deliverables

The following will be the major deliverables that will be produced at the end of this project.

7.1 RESTful API service

There will be a running instance of RESTful API service developed and deployed at the end of the project. This API will be responsible for communicating between the client applications and the central database server.

7.2 Android client application

The application will be developed integrating all the features proposed earlier. The users will be able to use the application to scan QR codes at different travel destinations which will reward them with scores in their accounts. An interactive map will be integrated with the application that shows the treasures installed at various tourist destinations near to the tourist's current location. The application will also be integrated with social networking platforms like Facebook, Twitter, Instagram etc. so the tourists can share their scores publicly to their friends and acquaintances.

8. Project Task and Time Schedule

The working time period for the project is three months. The project will be completed by the end of the spring semester as per the requirements of the university. The major task division among the team members is mentioned in Table 2.

Table 2: Division of tasks among project team members

Team Member	Assigned Tasks
Bikalpa Dhakal	Project Management Android Application Development Deployment
Avinash Shreshtha	API Development Data Management Project Documentation

The time schedule of the project is illustrated in Figure 11.

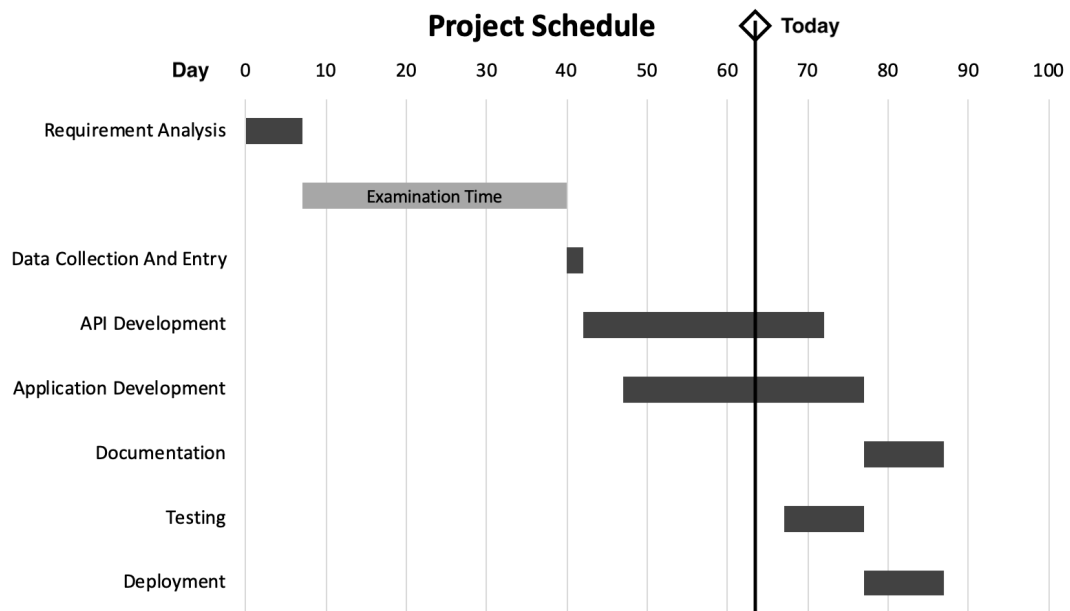


Figure 11: Proposed project schedule

References

- [1] *Nepal Tourism Statistics 2018*. Ministry of Culture Tourism and Civil Aviation, 2019.
- [2] T. H. Times, “Visit nepal 2020: Promises vs groundwork,” 2019. [Online]. Available: <https://thehimalayantimes.com/business/perspectives/visit-nepal-2020-promises-vs-groundwork/>
- [3] “Himalayas,” 2019. [Online]. Available: <https://www.britannica.com/place/Himalayas>
- [4] “Best trekking routes in nepal,” 2019. [Online]. Available: <https://nepalecoadventure.com/blog/best-trekking-routes-in-nepal/>
- [5] “Protected areas of nepal,” 2019. [Online]. Available: https://en.wikipedia.org/wiki/Protected_areas_of_Nepal
- [6] A. Zhukova, “The best scavenger hunt apps and ideas,” 2019. [Online]. Available: <https://www.makeuseof.com/tag/scavenger-hunt-apps-ideas/>
- [7] J. S. Warner and R. G. Johnston, “Gps spoofing countermeasures,” *Homeland Security Journal*, vol. 25, no. 2, pp. 19–27, 2003.