TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

SAGARMATHA ENGINEERING COLLEGE

A PROJECT PROPOSAL ON

DEHRA.COM

BY

RODAN RAMDAM SEC074BCT035

RAJ MAHARJAN SEC074BCT032

BISHWAS DANGOL SEC074BCT011

ANU SHARMA SEC074BCT009

A PROJECT PROPOSAL SUBMITTED TO THE DEPARTMENT OF
ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL
FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR IN COMPUTER ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

SANEPA, LALITPUR, NEPAL

September 7, 2020

ABSTRACT

A place to stay is always has been the one of the important parts of human's day to day life. Similarly, in the case of Nepal, according to Annual Household Survey 2016/2017 by CBS, Nepal [1] which was released in 2019, 24.7 percent of people in urban areas don't own their own house; either choosing to stay in an apartment or rent a room. However talking with various people from different places of our locality, It was found that people have problem finding rooms in the place they want and compare for the best price they can get. Applications that do exist which have tried to solve this problem seems to be missing more function which could be integrated in order to make life of people easier.

Even on the context of current situation, where the world is suffering from worldwide COVID-19 pandemic the option of choosing the best option for them and contacting the house owner and renter before the room is actually rented virtually has been even more necessary with the choice of showing the actual way towards the house for a person who want to rent in order for them to come in contact with as less people as possible is more important in current scenario.

Not only for current situation but even in the world when there are no more huge implications by COVID, features as such could help day-to-day people get rooms efficiently and room renters to get their preferred choice i.e. most people prefer certain types of people to rent their rooms such as students.

Keywords: Room Renting Nepal, Google Map API Integration, DBMS(Database Management System), Real Time Chat, MVC Approach(Model View Controller), Flutter Android Application Development, Python Backend with Flask, Object Relational Mapper Tool, ReactJs Web Application Development

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

DBMS   : Database Management System

API   : Application Program Interface

SQL   : Structured Query Language

MVC   : Model View Controller

REST API   : RestFul Application Program Interface

CHAPTER 1

INTRODUCTION

Home has always been a place where one's comfort zone lies. A place where people can rest from their day-to-day stress, think about something new, have time for oneself or their family. Not all people can afford to build a house hence they go to the second option of renting a room or an apartment.

1.1 Background

People tend to rent their place which comforts them the most i.e. people tend to rent near there work place or study place.

1.2 Problem Definition

Finding place where one has never been to is very time consuming and even when searching might not result on finding even on empty room. Even when searching online, there is no proper way of talking to the owner in a secure manner i.e. either a site just give the phone number of the owner to the public or they just send them an email and reveal their location to the public. This has a security flaw since owners personal information is out there in public.

1.3 Problem Solution

The solution to the problem can be seen in following ways:

- Create an application through which owner and renter can talk with each other through the application.

- Owner can approve for showing of their information if they find the request legit through chatting features in the application.

- Renter can book the room after both owner and renter approves after which through google maps integration, renter can use it to find the way to owners house.

1.4   Objectives

This work is conducted in order to:

- To make it easier for average citizen to rent a room

- To make it easier for average house owner to rent out a room

CHAPTER 2

LITERATURE REVIEW

The main idea of this project is to make renting of rooms for average citizen easier through improving several features such as adding way of having live chat, google maps integration allowing users to use the feature of following path to reach the destination, only reveling necessary data such as rent per month, location i.e. place's name only, Image of the place, etc. other data being shared when user allows it to be shared.

Various renting systems such has house renting system[3], car renting system projects have been done one of them being from Bangladesh, house renting system being focused on entire property similar to 'basobas.com'. Other products such as 'rentalnepal.com', 'gharbazar.com', 'gharbheti.com' and even in 'hamrobazar.com' already exist all trying to solve their own ways. All of these have solved one problem but removing some important features from previous products i.e. Some give phone numbers only, one only allows through there emailing service, one totally lacks images of the rooms posted in the site. All of these problems were the main reason this project is viable.

In order for this project to go on forward Web application development, Android application development and knowledge about DBMS was necessary. For all of these, React Js., Flutter, Flask and PostgreSQL were all the technology planned to be used with the help of Object Relational Mapper Tool such as SQLAlchemy. In optional technology, Node Js. is kept so that all the planned features could be integrated even if there is any problem which may occur during backend using Flask following the Model View Controller(MVC) Architecture.

CHAPTER 3

METHODOLOGY

3.1 Process Model

In this project, we opted to follow agile model[2] since we felt it was the best way of going forward learning the technology, easier way of learning together and is used during production on companies. Since in this model we can build our application in various iterations, working application features needed later on if found important could be integrated easily. This model also promotes teamwork and allow for cross-training i.e. allows us to go on with our plan of making both web application through React Js. and Android Application through Flutter. Debugging should also be easier using this model since at the beginning small application of doing main task of booking the room could be developed then including the other parts on later iterations making it easier and helping to find out the bug on the software quicker. Documentation could also be done easily which is important since the project is to be done in such time constraint.
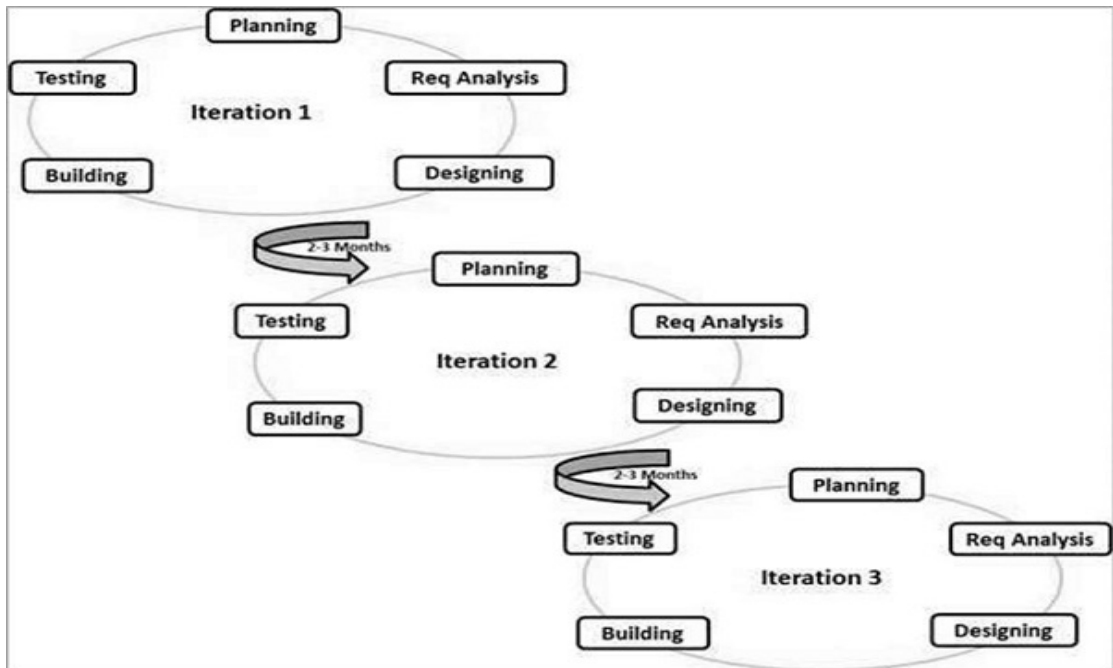

Figure 3.1: Agile Model[2]

3.2 Tools to be used

3.2.1 Android

Android is an Linux kernel based mobile operating system managed by Google. It is used by several smartphones and tablets. Example includes Samsung Galaxy, OnePlus, etc.

Unlike Apple's iOS, Android is an open source i.e. developer can modify or customize the OS for each phone. Even this is the case, application on similar android version can run on all the platform regradless of the vendor of the phone and unlike iOS, application can be deployed for testing and presenting purpose on an actual phone though any PC while iOS needs a MAC.

3.2.2 Android Studio

Android Studio is an official Integrated Development Environment(IDE) for Android, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio even offers more features that enhances your productivity with building Android Application.

Virtual android device is one of the important features which allow testing of files in different virtual android devices. Using developer mode's USB debugging mode feature.

3.2.3 VSCode

Visual Studio Code is a lightweight but powerful source code editor. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).

This IDE also allow android application development though testing in users mobile through the developer mode's USB debugging mode feature.

3.2.4 Flask

Flask is an microframework made from Python. Here the word 'micro' doesn't mean that it lacks functionality but is simple in it's core but is extensible. Decisions such as which database to use is not decided by flask and decisions of flask such as it's template engine of Jinja can be changed easily.

3.2.5 React Js.

React is a JavaScript library for building user interfaces. React has been designed from the start for gradual adoption, and you can use as little or as much React as you need.

3.2.6 Flutter

Flutter is Google's UI toolkit for building beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. In it's core, it uses dart programming language. It has features of fast development, good native performance with expressive and flexible UI.

3.2.7 PostgreSQL

PostgreSQL is a powerful, open source object-relational database system that uses and extends the SQL language combined with many features that safely store and scale the most complicated data workloads.
PostgreSQL has earned a strong reputation for its proven architecture, reliability, data integrity, robust feature set, extensibility, and the dedication of the open source community behind the software to consistently deliver performance and innovative solutions. PostgreSQL runs on all major operating systems, has been ACID-compliant since 2001, and has powerful add-ons such as the popular PostGIS geospatial database extender.

3.2.8 SQLAlchemy

SQLAlchemy is the Python SQL toolkit and Object Relational Mapper that gives application developers the full power and flexibility of SQL.

It provides a full suite of well known enterprise-level persistence patterns, designed for efficient and high-performing database access, adapted into a simple and Pythonic domain language.

3.3 Block Diagram
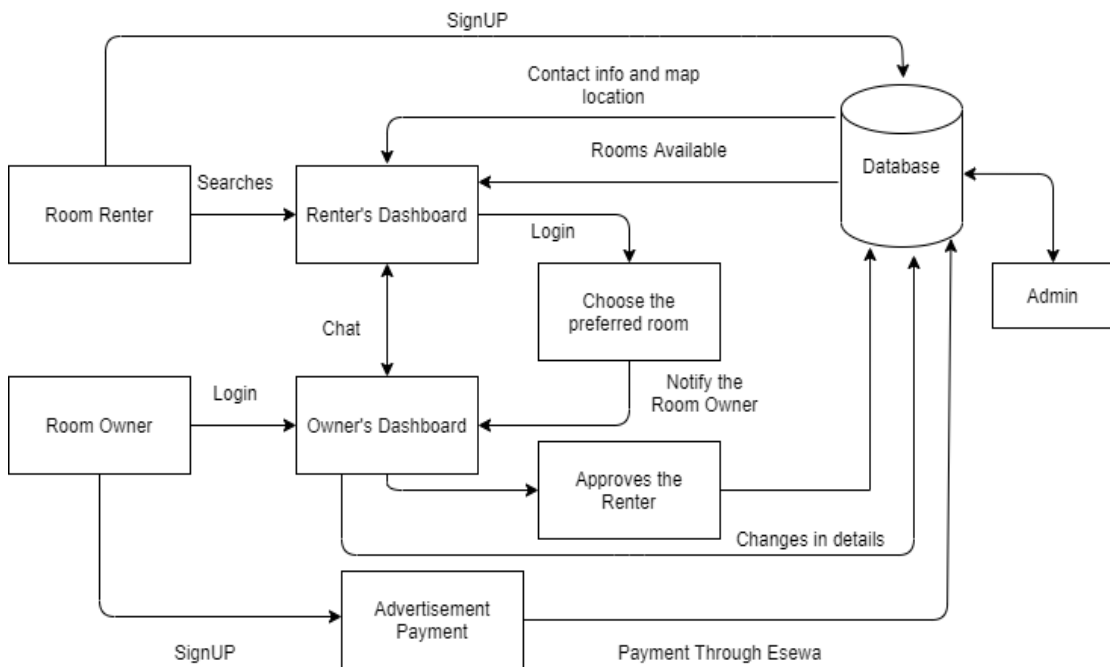
Figure 3.2: System Block Diagram

When room renter opens the application an interface which allows the user to check rooms available. In order to choose the room they prefer login is needed and if there is no account then room renter has to signup. After the room renter has signed in and choosen there prefered room, notification is sent to the room owner. When Room Owner opens the application, owner has to signup initally and choose advertisement plan then do online payment through the application. Next time they can just login then be presented to owner's dashboard where they can update more information about their room. If renter chooses their room then they can chat with the room renter.

After chatting through the application, if both come to an agreement, owner can then share their number and location by approving the renter. Renter can then find the rented room through the Google Map route from the application.

Admin can review all the transactions and could change data in database for improvement.

3.4 Algorithm

Algorithm 1: For Room Renter

Step1: Looks for the content

Step2:**while** *User Finds proper room or exits* **do**

  Step3: **if** *User Logged in* **then**
  | Goto Step6

  else if *User account exists* **then**
  | Goto Step5

  else
  | Goto Step4

  Step4: Signup for new account

  Step5: Login

  Step6: Choose Preferred room(Notification is sent to the Room owner)

  Step7: Chatting with the Room owner through the application

  Step8: **if** *Room Owner Agrees* **then**
  | Recieves detail i.e. Map and contact info.

  else
  | Recieves info that the request has been denied.

This Algorithm shows how a room renter can use our application.

Algorithm 2: For Room Owner

Step1: **if** *User Logged in* **then**
  | Goto Step4

else if *User account exists* **then**
  | Goto Step3

else
  | Goto Step2

Step2: Signup for new account

Step3: Registers for Advertisement

Step4: Login

Step5: **if** *User want to change detail* **then**
  | Change detail

Step6: **while** *No requests* **do**
  | No requests

Step7: Chatting with the Room Renter through the application. Step8: **if**

  Approves **then**
  | Goto Step9

else
  | Rejects the applicant and they get the information

  | Goto Step6

Step9: Option of sending phone number, Location of the house through

  maps.

This Algorithm shows how a room owner can use our application.

CHAPTER 4

FEASIBILITY STUDY

A feasibility analysis is used to determine the viability of an idea, such as ensuring a project is legally and technically feasible as well as economically justifiable. It tells us whether a project is worth the investment—in some cases, a project may not be doable. There was research done in order to check the feasibility.

4.1 Technical Feasibility

The task focused include front end in React Js. and Flutter, backend using Flask and Node js.(when needed) and DBMS using PostgreSQL, for map Google provides there Google Maps API. All of these are chosen due to there application on various stuffs like making better design to being easily extensible and main point being open source. All of these make it feasible to apply on software level. For hardware level following are the chosen:

4.1.1 PC

Computer with following features are chosen:

(a) Processor: Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz, 2401 Mhz, 2 Core(s), 4 Logical Processor(s)

(b) Physical Memory: 4.00 GB

(c) Graphics Card: AMD Radeon (TM) R5 M330

4.1.2 Android

Flutter Supports all the application above Android Jelly Bean 4.1.x or newer but in our case Marshmellow 6.0 is used.

4.2 Economic Feasibility

Since most of the software/program/framework being used is open source it doesn't have a big effect in the total budget of the software development. One which does require initial investment is Google's Map API which depends upon features chosen but afterwards Google gives 200$ credit per month which enough in per month basis unless there is a overload where additional payment might be necessary. Only other investment is for the labour work. All of the initial investment could be recovered though the owners sharing their rooms in our application.

4.3 Time Feasibility

Since our work flow is planned through agile model, working application can be released soon and features could be added eventually. To reach our goals, a gannt chart was developed planning the estimated time for each task. Each task were given bit more time as needed so that if needed it goal could be reached easily finishing the project on time.

4.4 Operational Feasibility

Although due to COVID-19 pandemic physical interview could not be held but still though the mediums like Facebook, interviews were conducted. Several people pointed us to the fact that there are applications which already exists and approved on the point that most of them lacks feature we planned to include and that it would help them if they were to rent a room and would use application that is easier for them.

CHAPTER 5

WORK SCHEDULE
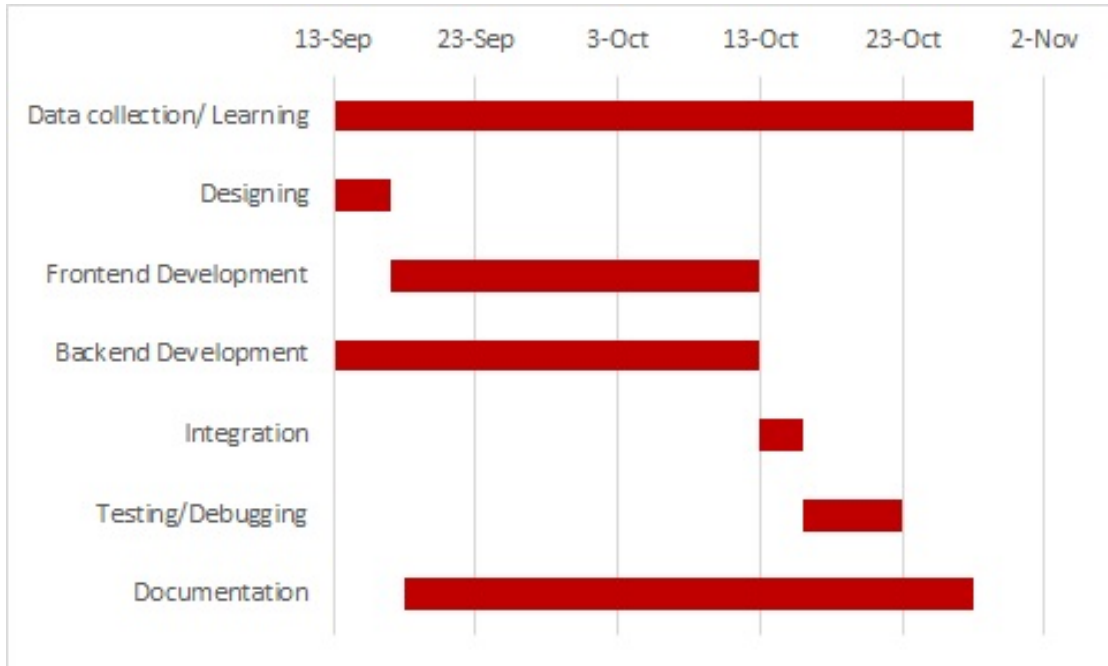
Figure 5.1: Time schedule

(a) Data Collection/Learning refers to the data being added to the database in order to make the application and learning how to apply them. Since this can go concurrently with other works being done and takes less time than others, it is taking full project time.

(b) Designing is the time for making UI/UX for frontend developement. It has been given 4 days since we are using Agile model, this could be improved on later iteration. It starts on the first day of the project.

(c) Frontend Development is the part to code the interface costumers see and act on. Since it is connected with designing part it's given 26 days giving interface development 1 month to complete.

(d) Backend Development is the part where all of the data are stored and is needed to be kept secure. Hence, all of the 1 month is given to this task and

it starts on the first day of the project. Team is divided into frontend and backend so that things could be done concurrently.

(e) Integration is the part of combining frontend and backend of the program. Several Errors might occur hence, 3 days are given for this.

(f) Testing/Debugging is the part where anyone except the development group is given the application to try and if bug is found it is patched out.

(g) Documentation is the part of writing down everything that has been done in the project hence making it easier for debugging and changing on future iterations.

CHAPTER 6

EXPECTED OUTPUT

Dehra is an multi platform application to make renting rooms easier for average citizen. The room renter can find rooms in any location s/he prefers and can contact the owner through the application which removes calls from unknown number for owner making it easier for owner to know who are interested. Although the location is given, to give map of the owner's room, owner needs to approve making it more secure for the owner and the map navigation feature helps the renter reach the destination easier. Both renter and owner can provide feedback which can be analysed to make the application better on future versions.

REFERENCES

[1] Central Bureau of Statistics Nepal. Annual household survey 2016/2017. Technical Report 3, 2019.

https://cbs.gov.np/wp-content/upLoads/2019/01/Annual-Household-Survey-2016_17.pdf

[2] Agile method(sdlc).

https://partheniumprojects.com/home-renting-system/

[3] Renting out house.

https://partheniumprojects.com/home-renting-system/