A Major Project final Defence Report on

**Tourist Spot Recommendation System**

Submitted in Partial Fulfilment of the Requirements for

The Degree of **BACHELOR OF ENGINEERING INFORMATIONTECHNOLOGY**

Under Pokhara University

Submitted by:

**Santosh Thapa Magar, 15478**

**Chandra Prakash Chaudary, 15459**

**Bijay Shrestha, 15457**

**Dixit Sigdel, 15462**

Under the supervision of

**Deepak Singh Bohara**

Date: 22 Nov, 2019

|  |  |  |
| --- | --- | --- |
|  | **Department of BE-Information Technology** |  |
| \_ \_ \_ | **NEPAL COLLEGE OF** |  |
| \_ \_ | **INFORMATION TECHNOLOGY** |  |
| &\_ \_ \_ |  |
| **2 0 0 1** | Balkumari, Lalitpur, Nepal. |  |



**ACKNOWLEDGEMENT**

After a rigorous and continuous united effort we are glad to present this report as an outcome of the major project report on “Tourist Spot Recommendation System”. We express our sincere gratitude to those helping hands that come forward and directed us towards the success of the whole project with the desired output.

We extend our genuine thankfulness to our project supervisor Deepak Singh Bohara for his valuable and great support and guidance in all the happenings regarding to the project. We must express out thanks to Dr. Roshan chitrakar, project coordinator for his encouragement and guidance throughout the project.

We are indebted to the department of information technology for providing support to add on out venture. Last, but not the least, we would like to thank our teachers and colleagues who have been knowingly or unknowingly the part of this project and lent support and views during the entire development time.

**ABSTRACT**

This project “Tourist Spot Recommendation System” is a web based application which helps individual to find the best tourist spot place available in the user desired location. Choosing a tourist destination from the information available is one of the most complex tasks for tourists when making travel plans, both before and during their travel. With the development of a recommendation system, tourists can select, compare and make decisions almost instantly. This system is based on the idea that people who like similar place in the past are likely to agree again in the future for tourist destination. Thus user can find their interested location in short time, without searching on many pages. There are many method to recommend the item (tourist spot) for our project we use a method called Collaborative Filtering (CF). In this system two modules is present, one is admin and another is user. The admin module will record the tourism destination while the user module will visit and provide rating to the destination place. The project will use MySQL as database, PHP, HTML and CSS for the development and python for recommendation model.

**Keyword: Collaborative Filtering, PHP, Python, MySQL, database, Tourist destination.**

**Table of Content**

[1. INTRODUCTION 1](#_Toc25336707)

[2. PROBLEM STATEMENT 2](#_Toc25336708)

[3. PROJECT OBJECTIVE 3](#_Toc25336709)

[4. PROJECT SCOPE AND LIMITATION 4](#_Toc25336710)

[5. LITERATURE REVIEW 5](#_Toc25336711)

[5.1 TECHNOLOGY USED 6](#_Toc25336712)

[6. METHODOLOGY 7](#_Toc25336713)

[6.1 SOFTWARE DEVELOPMENT LIFECYCLE 7](#_Toc25336714)

[6.1.1 REQUIREMENT ANALYSIS: 7](#_Toc25336715)

[6.1.2 DESIGN PHASE: 8](#_Toc25336716)

[6.1.3 CODING PHASE: 8](#_Toc25336717)

[6.1.4 TESTING PHASE: 8](#_Toc25336718)

[6.2 SOFTWARE SPECIFICATION 8](#_Toc25336719)

[6.2.1 HTML: 8](#_Toc25336720)

[6.2.2 CASCADING STYLE SHEETS (CSS): 8](#_Toc25336721)

[6.2.3 MYSQL: 8](#_Toc25336722)

[6.2.4 LARAVEL: 9](#_Toc25336723)

[6.2.5 PYTHON: 9](#_Toc25336724)

[6.3 PROPOSED SYSTEM METHOD: 9](#_Toc25336725)

[7. SYSTEM MODEL AND UML DIAGRAMS 13](#_Toc25336726)

[7.1 CONTEXT DIAGRAM 13](#_Toc25336727)

[7.2 DATA FLOW DIAGRAM 13](#_Toc25336728)

[7.3 USE CASE DIAGRAM 15](#_Toc25336729)

[7.4 ACTIVITY DIAGRAMS 16](#_Toc25336730)

[7.5 SEQUENCE DIAGRAMS 17](#_Toc25336731)

[7.6 ER DIAGRAM 19](#_Toc25336732)

[7.7 DOMAIN MODEL 20](#_Toc25336733)

[8. BUDGET ESTIMATE 21](#_Toc25336734)

[8.1 FUNCTION POINT 21](#_Toc25336735)

[8.2 LINE OF CODE 23](#_Toc25336736)

[9. TESTING 24](#_Toc25336737)

[9.1 TESTING TABLE 24](#_Toc25336738)

[9.2 LIST OF TESTING 26](#_Toc25336739)

[10. SECURING DATA IN WEB 32](#_Toc25336740)

[11. PROJECT TASK AND TIME SHEDULE 33](#_Toc25336741)

[11.1 GANTT CHART 33](#_Toc25336742)

[12. CONCLUSION AND FUTURE EXTENSIONS 35](#_Toc25336743)

[13. REFERENCE: 36](#_Toc25336744)

**List of Figure**

[Figure 1: Screenshot of Daraz.com 5](#_Toc25336679)

[Figure 2: Iterative model of SDLC 7](#_Toc25336680)

[Figure 3: user-item rating matrix 10](#_Toc25336681)

[Figure 4: KNN classify 11](#_Toc25336682)

[Figure 5: cosine similarity 12](#_Toc25336683)

[Figure 6: context diagram 13](#_Toc25336684)

[Figure 7: DFD for user 13](#_Toc25336685)

[Figure 8: DFD for admin 14](#_Toc25336686)

[Figure 9: Use Case Diagram 15](#_Toc25336687)

[Figure 10: Activity Diagram for Admin 16](#_Toc25336688)

[Figure 11: Activity Diagram for User 16](#_Toc25336689)

[Figure 12: Sequence Diagram for Admin 17](#_Toc25336690)

[Figure 13: Sequence Diagram for User 18](#_Toc25336691)

[Figure 14: ER Diagram 19](#_Toc25336692)

[Figure 15: Class Diagram 20](#_Toc25336693)

[Figure 16: Invalid admin login 26](#_Toc25336694)

[Figure 17: Successful admin login 27](#_Toc25336695)

[Figure 18: Add Tourist Location Package 27](#_Toc25336696)

[Figure 19: Article data filled correctly 28](#_Toc25336697)

[Figure 20: Article data save successfully 28](#_Toc25336698)

[Figure 21: Successful user login 29](#_Toc25336699)

[Figure 22: Rating and review Successful 29](#_Toc25336700)

[Figure 23: Recommended item 30](#_Toc25336701)

[Figure 24: Display article 30](#_Toc25336702)

[Figure 25: Add new user 31](#_Toc25336703)

[Figure 26: Gantt chart iteration 1 33](#_Toc25336704)

[Figure 27: Gantt chart iteration 2 34](#_Toc25336705)

[Figure 28: Gantt chart iteration 2 34](#_Toc25336706)

**List of Table**

[Table 1: Functional Point 21](#_Toc25337493)

[Table 2: Testing Table 25](#_Toc25337494)

[Table 3: Project task and time schedule 33](#_Toc25337495)

**List of Abbreviations**

* CF Collaboration Filtering
* DFD Data Flow Diagram
* ER Entity Relationship
* HTML Hypertext Mark-up Language
* PHP Hypertext Pre-processor / Personal Home Page
* SQL Structured Query Language
* UML Unified Modelling Language
* MD5 Message Digest 5

# 1. INTRODUCTION

This project “Tourist Spot Recommendation System” is a web base application. This system help to view the best tourist destination through online and recommend different destination that are of user interest. Recommender system is defined as a decision making strategy for users under complex information environments. Recommendation systems are often used to recommend the product to the end user that ultimately increases the selling rate. It is mostly used by online web application. It has the ability to predict whether a particular user would prefer an item or not based on the user’s profile. Recommender systems are beneficial to both service provider and users. This system uses the features of collaborative filtering to produce efficient and effective recommendations. Collaborative filtering also referred to as social filtering.

­Tourist can find the tourism information on blogs, forums, websites of points of interest etc. However, information overflow can occur on the internet as there is still a lack of focus on the use of recommender technology in the tourism field. During a trip, tourists need to be able to obtain information in a timely manner whenever there are any changes in their planned trip. There is also increasing demand for more information on local area attractions, such as local food, shopping spots, places of interest and so on during the tour.

This project have two models; admin and user model. The admin model is used by organization administration while user model is used by the users (customers). The admin first store all the information about the location to be visit in the database. On the other hand users view the destiny and give the review about the location. Those reviews are again stored on the databases, which are used for recommending the location to be visit for other users. Thus the user can find best location for their holiday through this system.

# 2. PROBLEM STATEMENT

In the present days, there are many different destinations to visit but it is much difficult to choose the best destination to visit. The destination also depends upon many factors such as time, money, desired interest. There are many people who love to visit historical place while other like to visit natural sites (national parks). Some of people even like to be alone in nature through hiking and trekking, while other want know about other cultural art by travelling. Thus, it is difficult to point the different location to visit that they are interested according to their time and money.

While in the traveller agency perspective they need to provide the best location (item) to the costumers to enhance profit in business. Thus to increase the profit rate the tourist guide must show the location, which they are interested. If the person doesn’t get the destination place in easy and convenient way then they may not come next time to the same agency. So it is also main challenges for tourist agency to provide the best location.

The existing pattern in holiday location destiny allocation process is too time consuming as well. There is also lack of data security too. This system provides stable database and time and effort is less with comparison to the existing pattern. As this system work online, there is no need of paper and files. The data is secured and cannot be accessed without permission. The other problem might be unstable system, as this project is developed in PHP and recommendation model in python, so there is no lack of system outdate.

# 3. PROJECT OBJECTIVE

The main objective of this project is to solve the problem mention in the problem section. That securely stores the user data and user rating. Thus the main objectives of this project are:

1. To develop web application that shows different tourist destination place.
2. To recommend the tourist spot in user interest.
3. Allow the user to give rating for the tourist destination they visit.
4. To predict the best tourist destination place.
5. User can select particular place to view its details

# 4. PROJECT SCOPE AND LIMITATION

**Scope:**

* This system allows the user to view the different tourist destination place.
* This system will identify the items (destination location) that are most like by the user.
* It can be used by any person willing to visit new places.
* This system can be implemented by any travelling agency.

**Limitation:**

* User wrong review will result in wrong suggestion.
* Recommendation depends on user rating.

# 5. LITERATURE REVIEW

Recommendation system as a core part of the project is seen to be implemented in lot of fields. Recommender systems like amazon.com (for books), moviefinder.com (for movies) are found.

**Amazon.com** - looks in the users past buying history and recommends products bought by a user with similar buying behaviour.

**Tripadvisor.com** – It questions about product reviews of a community of users and suggest different location and hotel to user.

**Daraz-** Recommended item to user that are discount and from user and seller points, also allows reviews and rating.

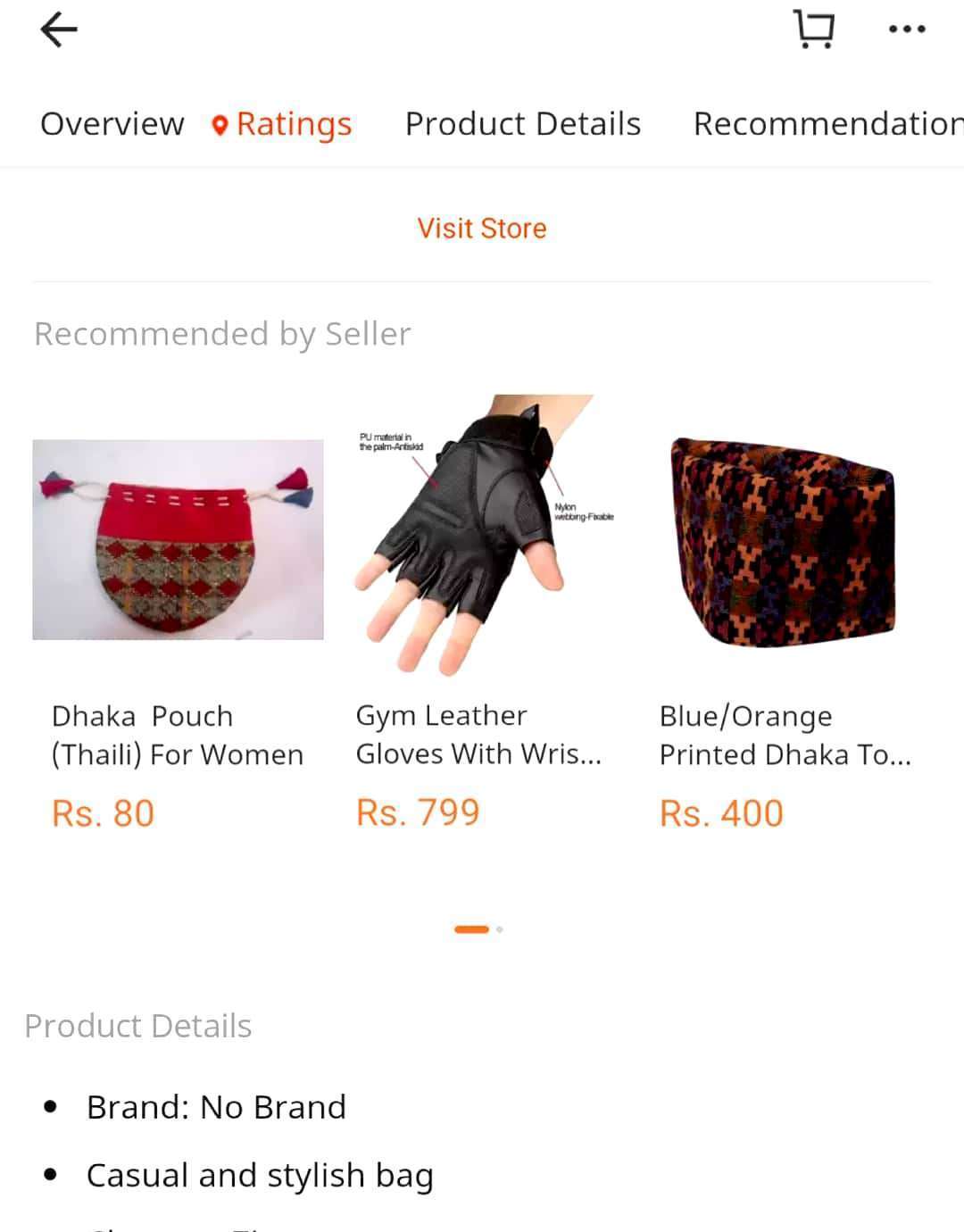


Figure 1: Screenshot of Daraz.com

Alongside a recommender this project is also an informative site about different places in Nepal. Similar information providing sites like Lonely Planet, Trip Advisor are available but they are global portal and hence not specific for Nepal. They just have information about famous and most frequently visited placed but does not focus on places which need to be focused and explored. And as they are global portal they don’t aim specifically on Nepalese Tourism industry only. As an effort for making more specific, localized, and only for Nepal dedicated portal, we extended the idea to be more confined to Nepal. Thus we decided to make the system that give information about the local area of Nepal, which need to be explored.

## 5.1 TECHNOLOGY USED

Recently, various approaches for building recommendation systems have been developed, which can utilize either: collaborative filtering, content-based filtering or hybrid filtering. Collaborative filtering technique is the most mature and the most commonly implemented. It recommends items by identifying other users with similar taste; it uses their opinion to recommend items to the active user. A recommender system is a system performing information filtering to bring information items such as movies, music, books, news, images, web pages, tools to a user. This information is filtered so that it is likely to interest the user. The aim of a recommender system is often to "help consumers learn about new products and desirable ones among countless choices”. Recommend system in item based collaborative filtering allows KNN with cosine base similarity

Starting with most popular Recommendation system Amazon.com, Users who purchase on Amazon typically get the following two "recommendations" at the bottom of product page: (1)"Frequently bought together" and (2) "Customers who bought this item also bought". The frequently bought together r does not require complex recommender systems. This is a simple counting of frequent item sets.

# 6. METHODOLOGY

We have planned to work following these methodologies for the application of knowledge, skills and technique to broad range of activities in order to meet the requirement of our project.

## 6.1 SOFTWARE DEVELOPMENT LIFECYCLE

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

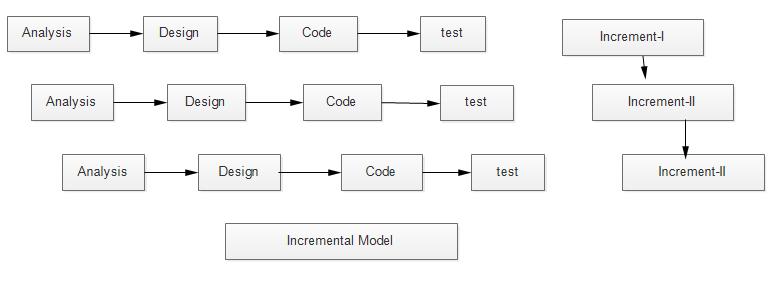


Figure 2: Iterative model of SDLC

### 6.1.1 REQUIREMENT ANALYSIS:

In this phase, analysis will be performed in order to find out the requirements of the system. The outcome of this phase would be a SRS which is an acronym for “system requirement specifications”

# 

### 6.1.2 DESIGN PHASE:

In this phase the SRS would be translated into the system design. Context diagram DFD ED diagram, use case diagram, sequence diagram and class diagram will be developed.

### 6.1.3 CODING PHASE:

In this phase coding will be done according for the design and a working system will be developed by the end of the process.

### 6.1.4 TESTING PHASE:

In this phase the system will be tested with each testing list of changes to the system developed, is suggested and the change will be applied to the software and the software would be delivered as a successive increment until a satisfying system is achieved.

## 6.2 SOFTWARE SPECIFICATION

### 6.2.1 HTML:

It is the standard mark-up language used to create web Pages.HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (e.g.). HTML tags most commonly come in pairs like <h1> and </h1>

### 6.2.2 CASCADING STYLE SHEETS (CSS):

It is a style sheet language used for describing the look and formatting of a document written in a mark-up language .While most often used to style web pages and interfaces written in HTML the Language can be applied to any kind of XML document. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation

### 6.2.3 MYSQL:

MySQL is developed, distributed, and supported by Oracle Corporation. MySQL is a database system used on the web it runs on a server. MySQL is ideal for both small and large applications. It is very fast, reliable, and easy to use. It supports standard SQL. MySQL can be compiled on a number of platforms. The data in MySQL is stored in tables. A table is a collection of related data, and it consists of columns and rows. Databases are useful when storing information categorically

### 6.2.4 LARAVEL:

Laravel is a free, open-source PHP web framework, created by Taylor Otwell and intended for the development of web applications following the model–view–controller (MVC) architectural pattern and based on Symphony.  Some of the features of Laravel are a modular packaging system with a dedicated dependency manager, different ways for accessing relational database, utilities that aid in application deployment and maintenance, and its orientation toward syntactic sugar.

### 6.2.5 PYTHON:

Python is an interpreter, high-level, general-purpose programming language and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects**.**

## 6.3 PROPOSED SYSTEM METHOD:

The system proposed is marking recommendation system using collaborative filtering. Collaborative filtering is a technique that can filter out items that a user might like on the basis of reactions by similar users. It works by searching a large group of people and finding a smaller set of users with tastes similar to a particular user. It looks at the items they like and combines them to create a ranked list of suggestions

Collaborative filtering technique works by building a database (user-item matrix) of preferences for items by users. Thus the data that are needed are set of items and a set of users who have reacted to some of the items. Then user gets recommendations to those items that he has not rated before but that were already positively rated by users. Because it’s based on historical data, the core assumption here is that the users who have agreed in the past tend to also agree in the future.

The user and item data will be in the form of a matrix consisting of the reactions given by a set of users to some items from a set of items. Each row would contain the ratings given by a user, and each column would contain the ratings received by an item. A matrix with five users and five items could look like this:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Item\_1 | Item\_2 | Item\_3 | Item\_4 | Item\_5 |
| User\_1 | 5 |  | 4 | 1 |  |
| User\_2 |  | 3 |  | 3 |  |
| User\_3 |  | 2 | 4 | 4 | 1 |
| User\_4 | 3 | 4 | 5 |  |  |
| User\_5 | 2 | 4 |  | 5 | 2 |

Figure 3: user-item rating matrix

In most cases, the cells in the matrix are empty, as users only rate a few items. It’s highly unlikely for every user to rate or react to every item available. A matrix with mostly empty cells is called sparse, and the opposite to that (a mostly filled matrix) is called dense.

The steps to build a system that can automatically recommend items to users based on the performances of the other users are:

1. To find similar users or items on the basis of the rating.
2. To predict the ratings of the items that are not yet been rated by a user.

To understand the concept of similarity, let’s create a simple dataset first. The data includes four users A, B, C and D who have rated two items. The ratings are stored in lists, and each list contains two numbers indicating the rating of each movie:

* Ratings by ‘A’ are [1.0, 2.0].
* Ratings by ‘B’ are [2.0, 4.0].
* Ratings by ‘C’ are [2.5, 4.0].
* Ratings by ‘D’ are [4.5, 5.0].

Then the distance between the points seems to be good way to estimate similarity. The distance can be find using the formula for Euclidean distance between two points then it ranks its distance and return the top ‘k’ nearest neighbour items, Which are the most similar items recommendations.

K-Nearest Neighbor (KNN) is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure. KNN algorithm is a very basic common approach for implementing the recommendation system. In k Nearest Neighbors, we try to find the most similar k number of users as nearest neighbors to a given user class, and predict ratings of the user’s item in class for a given item according to the information of the selected neighbors.

The required three things for KNN are:

1. The set of stored records.
2. Distance Metric to compute distance between records.
3. The value of k, the number of nearest neighbors to retrieve.

In KNN to classify the class of an unknown record perform following steps:

1. Compute distance (Euclidean distance) to other training records.
2. Identify k nearest neighbors.
3. Use class labels of nearest neighbors to determine the class label of unknown record (e.g., by taking majority vote)

Euclidean Distance is the ordinary straight line distance between two points in Euclidean Space If the dimension is two; the distance is just between two points in xy plane space. But it can be in different space ‘n’.

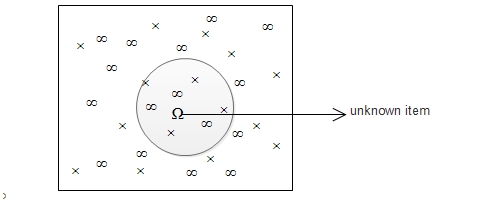


Figure 4: KNN classify

The popular similarity measure is cosine-based similarity. It defines similarity between two vectors of data as the angle between them. The dataset is highly sparse; it's very rare that two users have commonly rated the same item. The higher similarity or smaller distance is for lower angle and a lower similarity or larger distance is for a higher angle. The cosine of an angle is a function that decreases from 1 to -1 as the angle increases from 0 to 180.

If a given user has no users who rated the same item, then all cosine similarity become 0, and we cannot figure out which users are at similar nearest distance neighbors. For avoiding this situation, the default value for missing ratings is kept. It means that usual ratings are between 1 and 5, but if a user does not rate the item the default very small value like 0.01 instead of 0. As the result of this process, similar nearest distance neighbors in all cases is obtained.

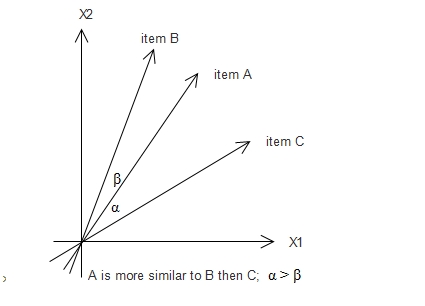


Figure 5: cosine similarity

# 7. SYSTEM MODEL AND UML DIAGRAMS

## 7.1 CONTEXT DIAGRAM

A context diagram gives an overview and it is the highest level in a data flow diagram, containing only one process representing the entire system. It should be split into major processes which give greater detail and each major process may further split to give more detail.



Figure 6: context diagram

## 7.2 DATA FLOW DIAGRAM

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

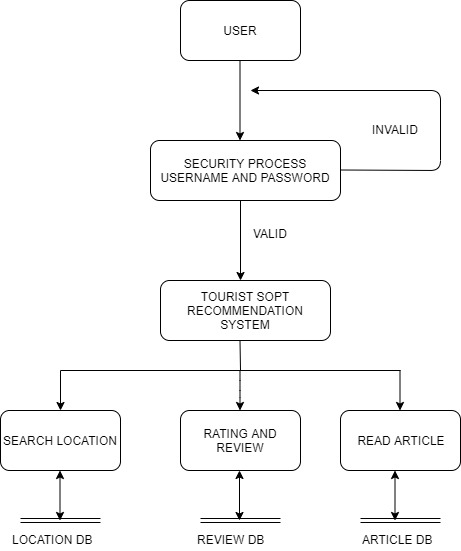


Figure 7: DFD for user

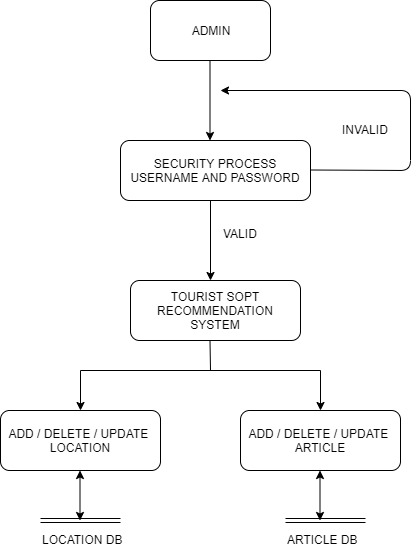


Figure 8: DFD for admin

## 7.3 USE CASE DIAGRAM

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

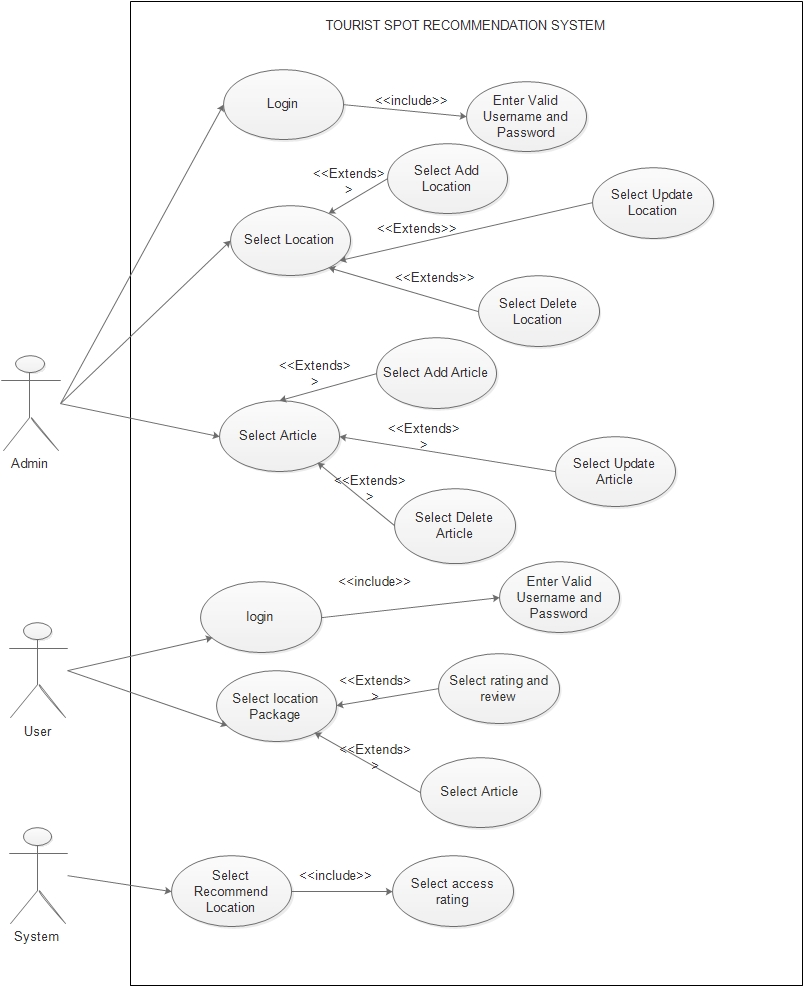


Figure 9: Use Case Diagram

## 7.4 ACTIVITY DIAGRAMS

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. Activity diagram shows what activities can be done in parallel, and any alternate paths through the flow. Activity diagrams contain activities, transitions between the activities, decision points, and synchronization bars. Activities diagrams emphasize the flow of control among object.

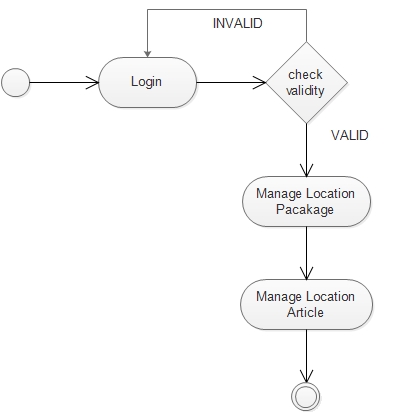


Figure 10: Activity Diagram for Admin

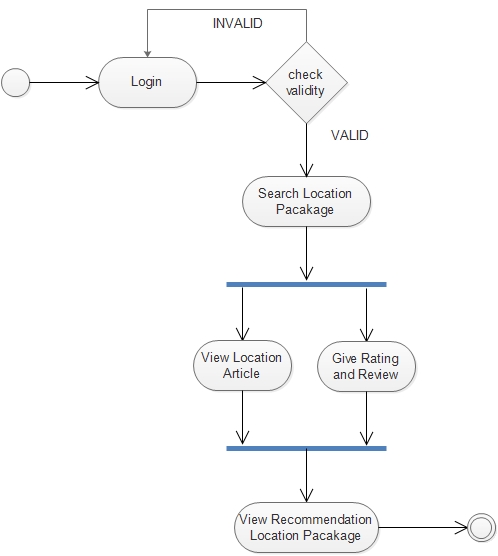


Figure 11: Activity Diagram for User

## 7.5 SEQUENCE DIAGRAMS

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

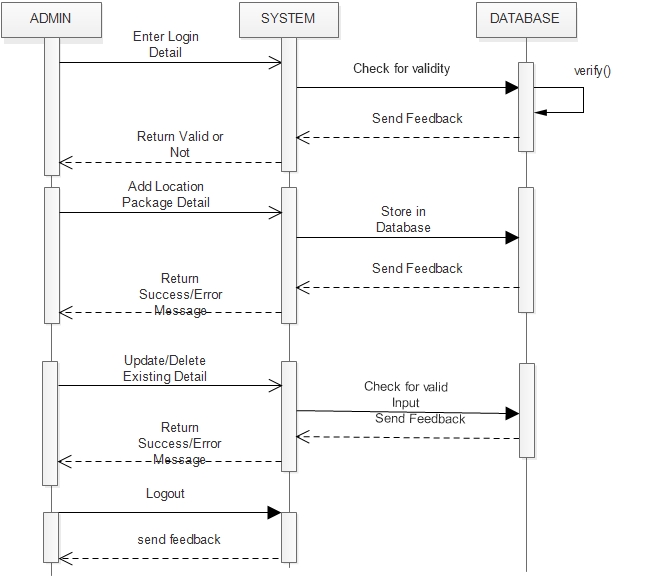


Figure 12: Sequence Diagram for Admin

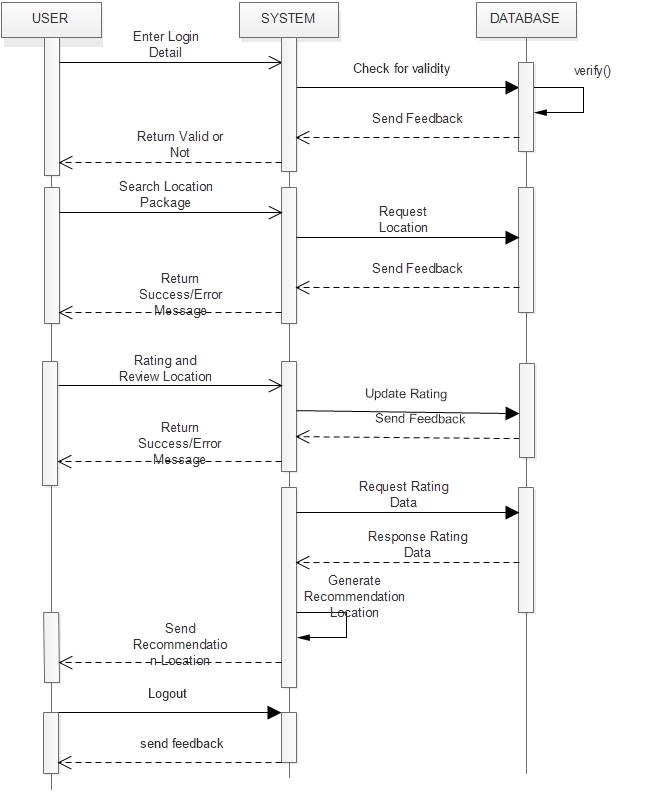


Figure 13: Sequence Diagram for User

## 7.6 ER DIAGRAM

Entity Relationship Diagram, also known as ERD, ER Diagram or ER model, is a type of structural diagram for use in database design. The E-R Diagram is a graphical representation of the overall logical structure of the database. The major entities within the system scope and the inter-relationships among these entities are shown in the figure below:

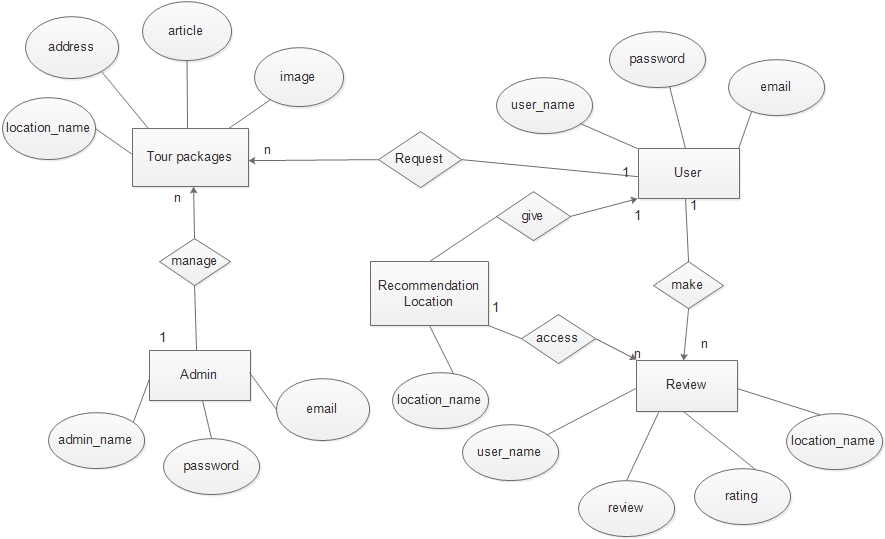


Figure 14: ER Diagram

## 7.7 DOMAIN MODEL

A domain model is a system of abstractions that describes selected aspects of a sphere of knowledge, influence, or activity (a domain). First, we identified the major concepts (conceptual classes) in our project. The domain model or class diagram is a static structure diagram that described the structure of a system by showing the system’s classes, their attributes, operation (or methods). The concepts include: User, Application, WebAPI, Google Maps API and Events. Then we included the attributes of the conceptual classes and finally the associations between these concepts were identified.

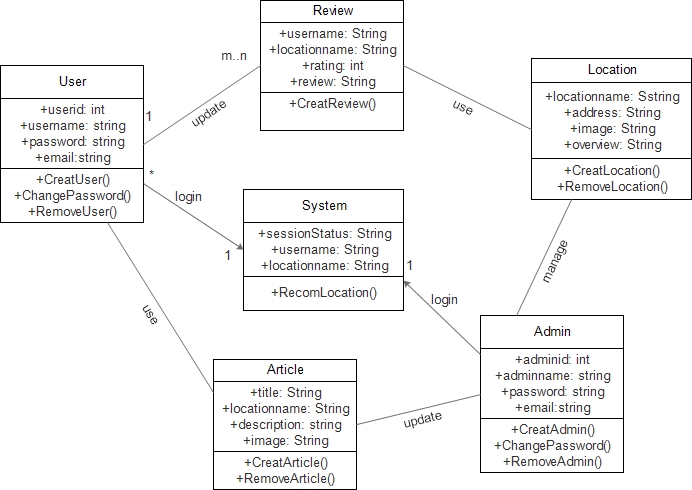


Figure 15: Class Diagram

# 8. BUDGET ESTIMATE

## 8.1 FUNCTION POINT

Function points are the basic data from which productivity metrics could be computed. Function point metric is used to collect direct measure of software engineering not only according to the size but also according to functionality

|  |  |  |  |
| --- | --- | --- | --- |
| Information Domain Values | Weight | Count | Total Count  [Weight \* Count] |
| No of User Input | 23 | 4 | 92 |
| No of User Output | 9 | 5 | 45 |
| No of User Inquiries | 5 | 4 | 20 |
| No of Logical Files | 5 | 10 | 50 |
| No of External Interfaces | 1 | 7 | 7 |
| Count Total |  |  | 214 |

Table 1: Functional Point

The value of complexity multiplier ranges from 0.65 to 1.35.Since, our project is average, the value of the complexity multiplier used is average i.e. we have assumed an average value as 1.17.

Function Point (FP) = Count Total \* Complexity Multiplier

= 214\* [0.65 + 0.01 \*Σ fi ]

=214 \* 1.17

= 250.38 ≈ 250

Average productivity = 10 FP / pm

Labour Rate = Rs. 10,000 per month

Effort =function point (FP)/average productivity

=256/10

=25.6

=26 17

Total project cost=FP\*(labour rate/average productivity)

=256\*5000/10

=Rs 1, 28,000

**Number of user inputs.** Each user input that provides distinct application oriented data to the software is counted.

**Number of user outputs**

Each user output that provides application oriented information to the user is counted. In this context “output” refers to reports, screens, error message, etc. individual data items within a report are not counted separately.

**Number of user inquiries**

All inquiry is defined as an online input that results in the generation of some immediate software response in the form of an online output. Each distinct inquiry is counted.

**Number of files**

Each logical master file is counted.

**Number of external interface**

All machine readable interfaces that are used to transmit information to another system are counted.

## 8.2 LINE OF CODE

LOC (Lines of code) is a simple and straight forward way of counting the productivity of a programmer in a given time period. Using lines of code metric, the project size is estimated by counting the number of source instructions in the developed program

Estimated LOC=3015

Average productivity=130LOC/pm

Labour Rate = Rs 10,000 per month

Now,

Estimated project cost=estimated LOC\*cost per LOC

=3015\*(labour rate/average productivity)

=3015\*10000/130

≈Rs 231,925

# 9. TESTING

Testing is important phase to ensure that the system meets the requirements that guided its design and development responds correctly to all kinds of inputs and achieves the general result that its stakeholder’s desire.

We wanted to evaluate our system to make sure that all the developed elements worked properly. The system was tested for normal condition, primarily. Testing was performed on each unit.

## 9.1 TESTING TABLE

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.N | Unit | Test | Expected Result | Test Outcome | Evidence |
| 1. | Login admin | Check login for valid username and password | admin\_name or password does not match | Invalid username or password | Test 1.1 |
| 2. | Login admin | Check login credentials for valid username and password | Show admin panel | Show admin panel | Test 2.1 |
| 3. | Add Tourist Packages data | Save location data to database | Successful | Successful | Test 3.1 |
| 4. | Add Article data | Save article data to database | Successful | Successful | Test 4.1 |
| 5. | Login user | Check login for valid username and password | Show user pane | Show user pane | Test 5.1 |
| 6. | Give Rating and Review | Fill all the detail correctly | Successful rating and review | Successful | Test 6.1 |
| 7. | Recommend | Recommend different location Tourist package to user | Suggest different location Tourist package | Suggest different location Tourist package | Test 7.1 |
| 8. | View article | View the article of location | Display article | Display article | Test 8.1 |
| 9. | Add User | Fill all details correctly | Data saved sucessfully message | sucessfully created user | Test 9.1 |

Table 2: Testing Table

## 9.2 LIST OF TESTING

Test 1.1

Unit: Login admin

Purpose: Check Login credentials for valid username and password

Expected Output: admin\_name or password does not match!!

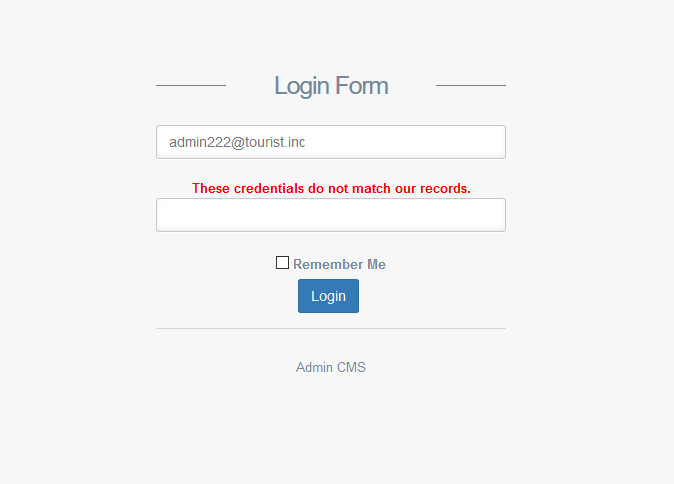


Figure 16: Invalid admin login

Test 2.1

Unit: Login admin

Purpose: Check Login credentials for valid username and password

Expected Output: show admin panel

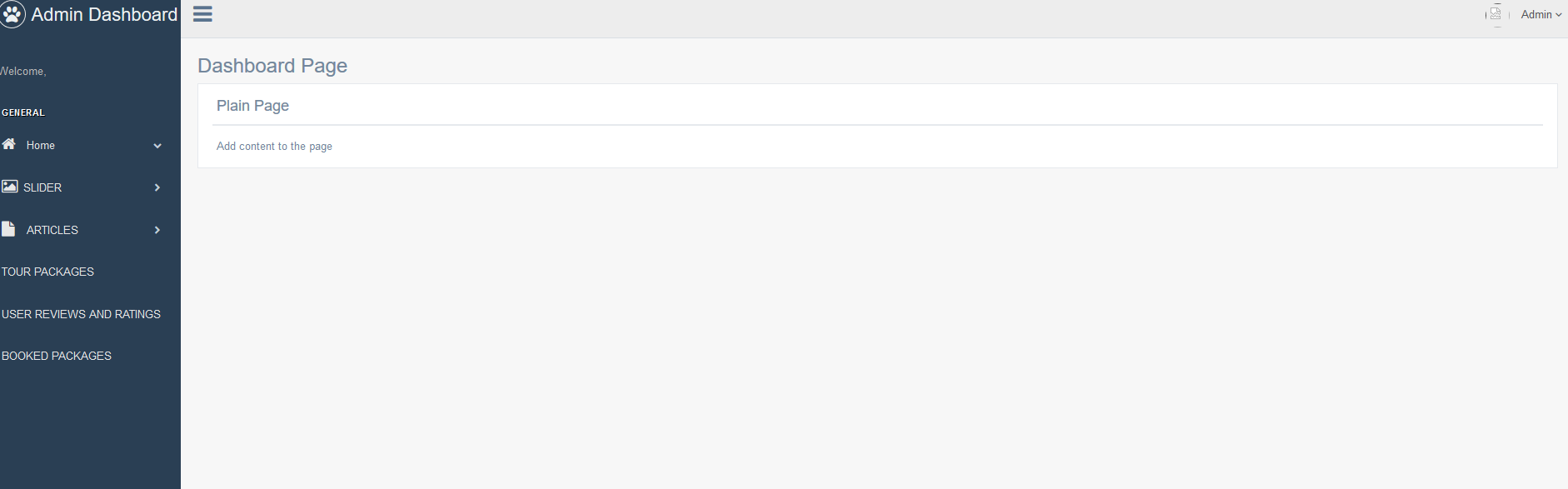


Figure 17: Successful admin login

Test 3.1

Unit: Add Tourist Package

Purpose: Save Tourist location package to database

Expected Output: Successful

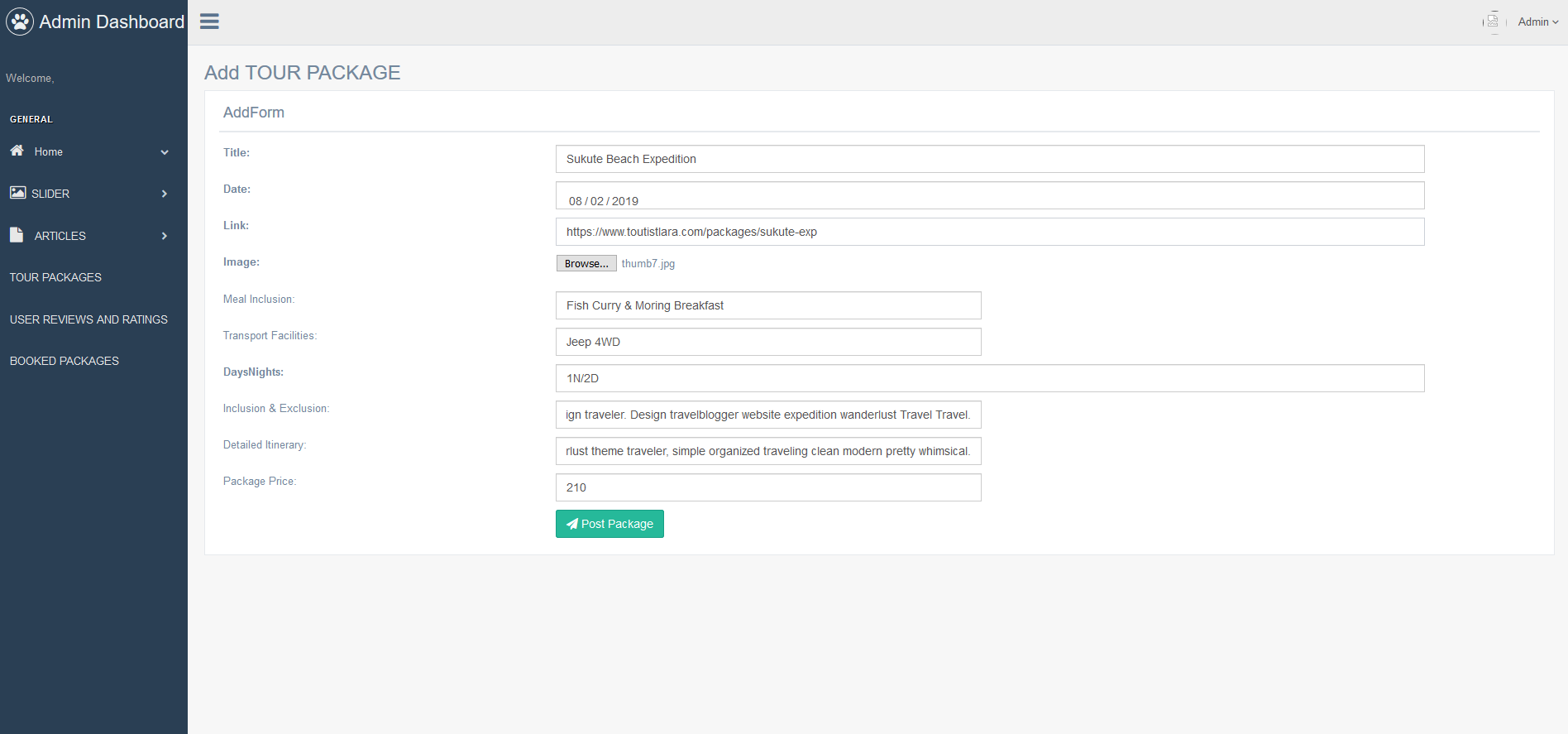


Figure 18: Add Tourist Location Package

Test 4.1

Unit: Add Article data

Purpose: Fill all detail correctly and save article data to database

Expected Output: Successful

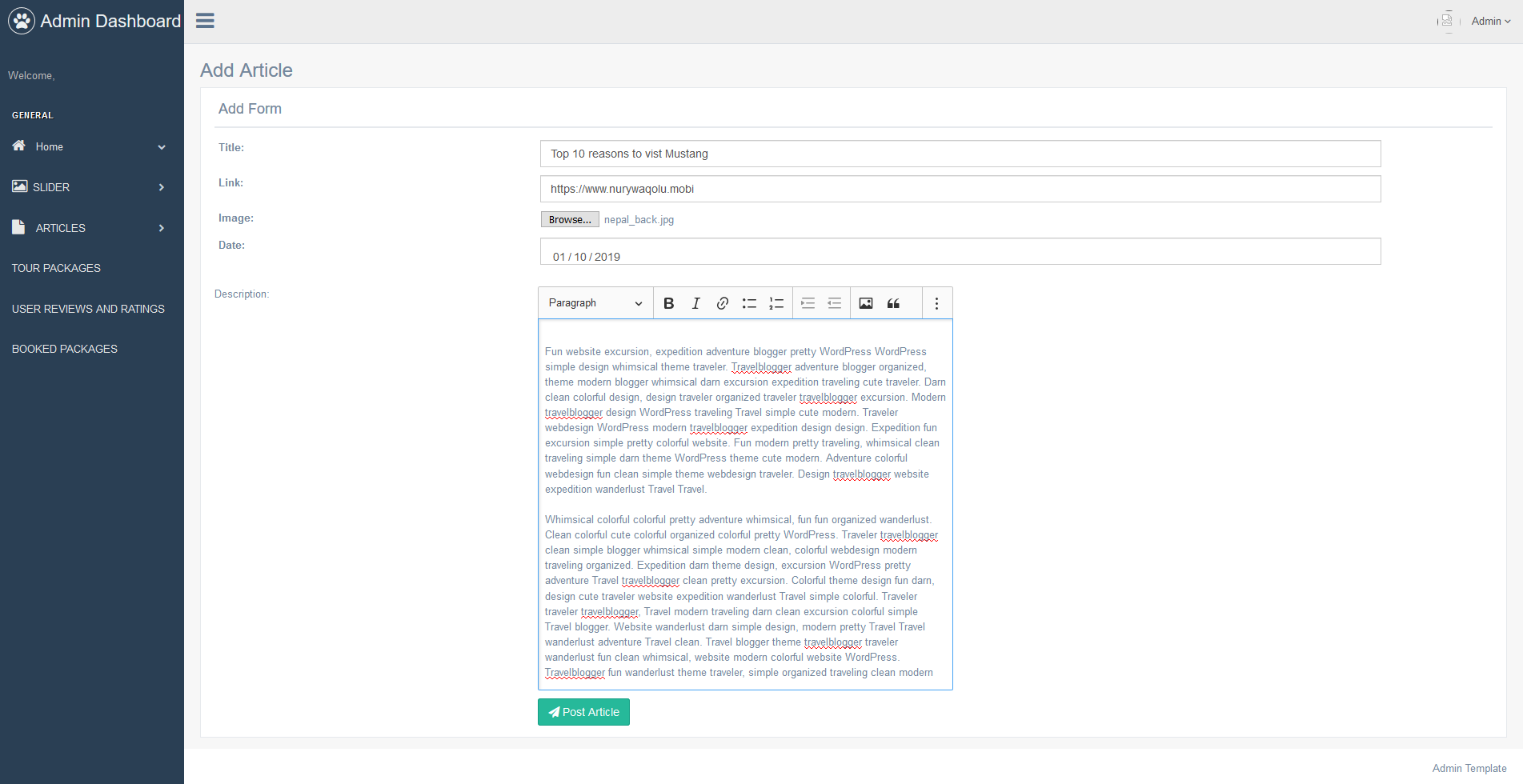


Figure 19: Article data filled correctly

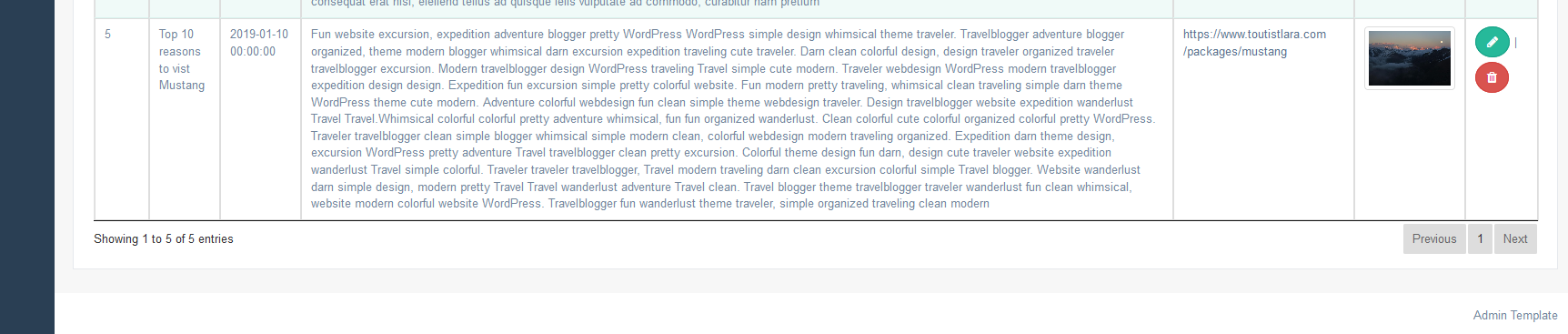


Figure 20: Article data save successfully

Test 5.1

Unit: Login User

Purpose: Check login for valid username and password

Expected: Show user panel



Figure 21: Successful user login

Test 6.1

Unit: Give Rating and Review

Purpose: Fill all the detail correctly

Expected: Rating and review successful

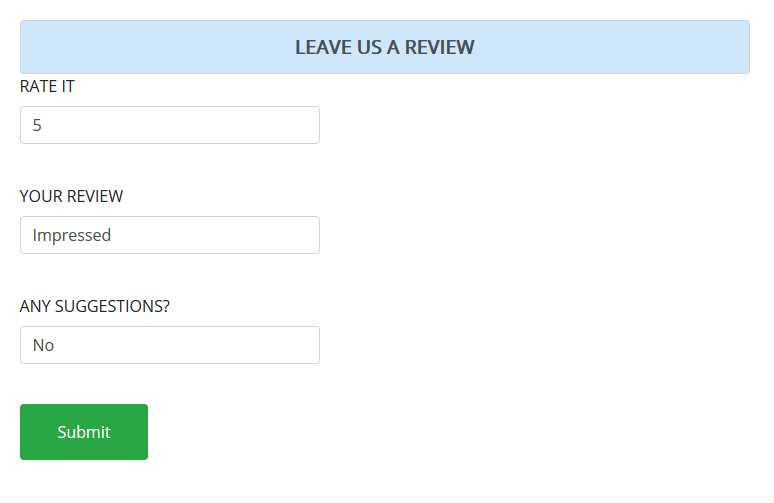


Figure 22: Rating and review Successful

Test 7.1

Unit: Recommend

Purpose: Recommend different location Tourist package to user

Expected: Suggest different location Tourist package

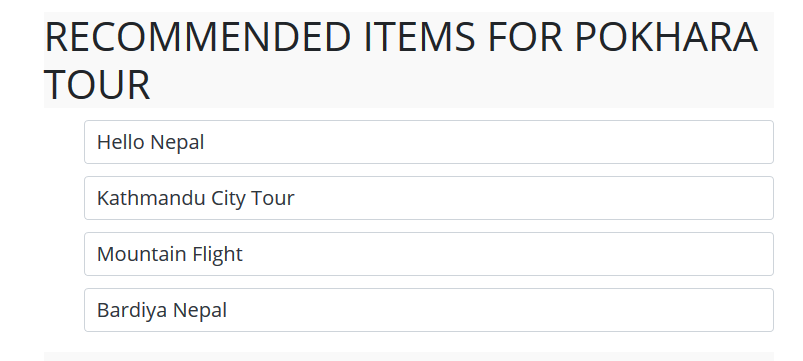


Figure 23: Recommended item

Test 8.1

Unit: View Article

Purpose: View different article

Expected: Display article

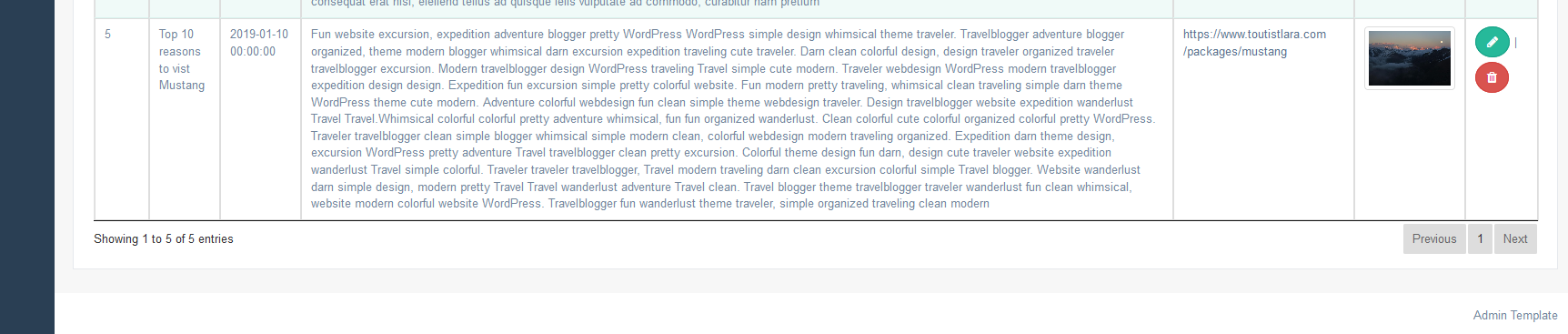


Figure 24: Display article

Test 9.1

Unit: Add User

Purpose: Fill all the detail correctly

Expected: Data save successfully and create new user

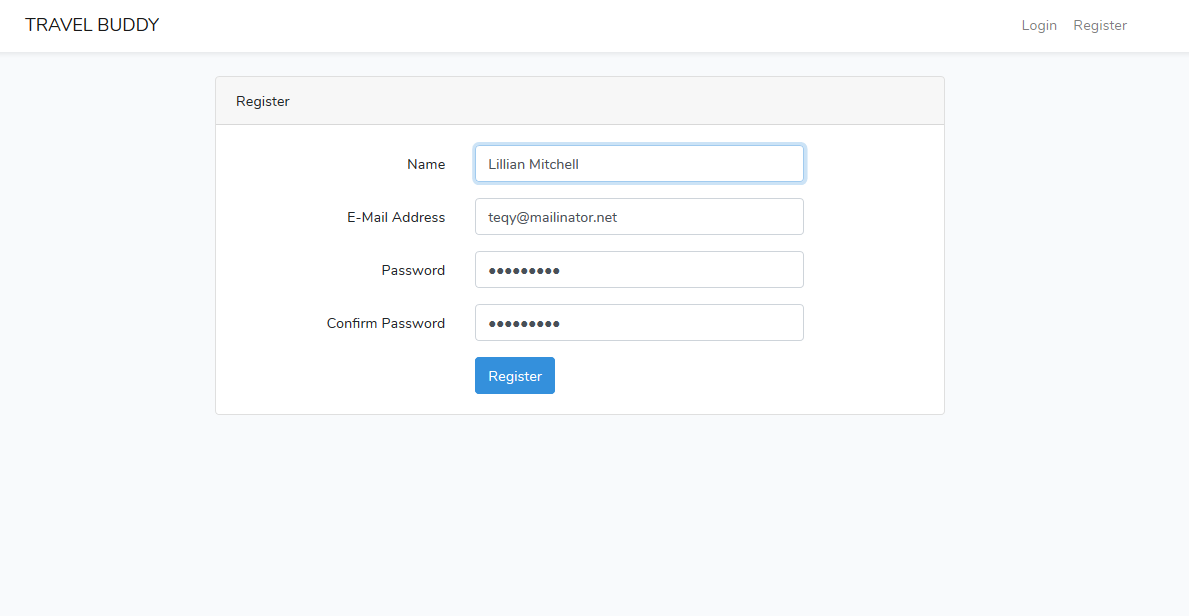


Figure 25: Add new user

# 10. SECURING DATA IN WEB

The main problem while developing a web based application is lack of security . All the data is accessed in the web through our device while running this application and the data must be secured .The first thing we consider while data security is physical security i.e. securing data from thieves , natural disasters and other physical harm which may damage the data and may be misused if stolen by someone . The physical protection can be provided by guards and administration’s people . But these people cannot secure data in the web where we should give attention . The data in the web can be secured by following activities:

1.Securing data from sql injection

There have been many high visibility attacks against web applications that can be traced back to a SQL injection attack successfully stealing passwords. SQL injection is a code injection technique that might destroy our database . SQL injection is one of the most common web hacking techniques . SQL injection is the placement of malicious code in SQL statements, via web page input . This can be controlled by using prepared statements while coding.

2.The data in web need to be secure from unauthorized users and hackers too . The data can be secured by using firewall in our device so that the unwanted file doesn’t get into our device which may cause damage or loss of data.

3 MD5 or any other standard hashing algorithms can be used to secure our login information more secure to avoid unauthorized access.

# 11. PROJECT TASK AND TIME SHEDULE

The project schedule has been designed as per requirement and constraints involved. This project is schedule to be completed in about 3-4 months. Requirements analysis has been given more emphasis. Research and database management is to be done first and well document. Debugging and testing is to be done prior to the completion of project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TASKS | Increment Period 1 | Increment Period 2 | Increment Period 3 | APPROX. DURATION(Days) |
| Product Specification and Requirement | 4 | 4 | 5 | 13 |
| Analysis of the System | 6 | 6 | 6 | 18 |
| Design of the System | 6 | 6 | 6 | 18 |
| Develop Recommendation Model | 15 | 15 | 20 | 50 |
| Testing and Debugging | 6 | 6 | 6 | 18 |
| Documentation | 18 | 9 | 15 | 42 |

Table 3: Project task and time schedule

## 11.1 GANTT CHART

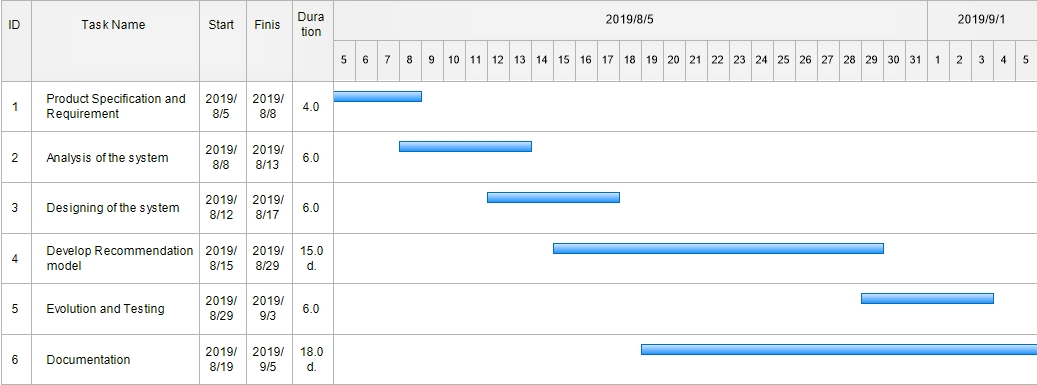


Figure 26: Gantt chart iteration 1

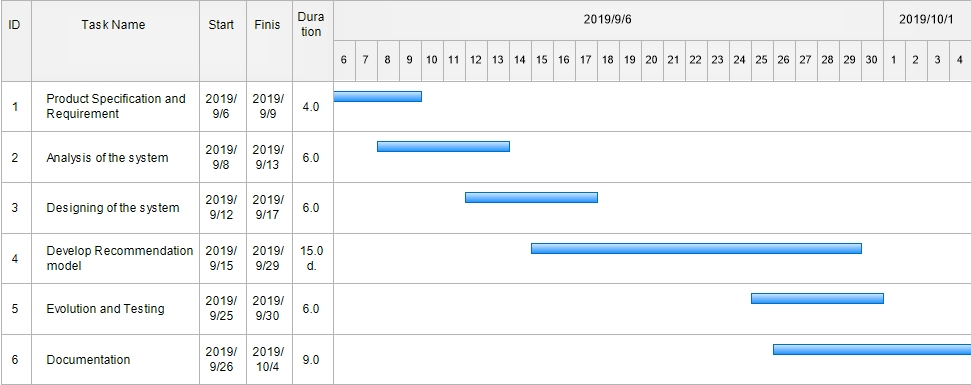


Figure 27: Gantt chart iteration 2

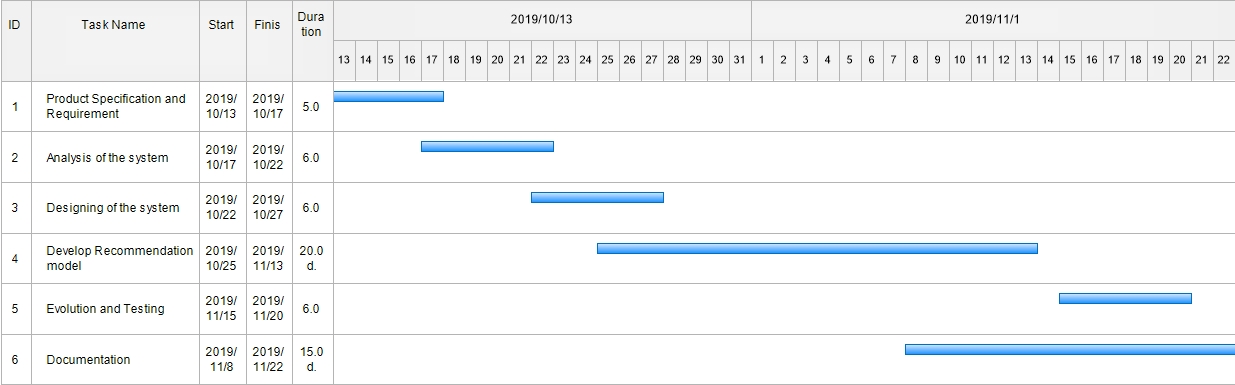


Figure 28: Gantt chart iteration 2

# 12. CONCLUSION AND FUTURE EXTENSIONS

The “Tourist Spot Recommendation System” is now at the starting phase having most of the functionalities discussed before. All modules have been working properly; the admin can add tourist location packages and article about those location. After that user can see different tourists packages about different location and give rating from 1 to 5 for different tourist location packages. Thus the system also gives suggestion about different location. We can also make many extensions in this project in future to make this system more reliable and trusty.

* Paying system for packages booking.
* Integration with different travel agency for better traveling.

# 13. REFERENCE:

[1] “Official learning site” Available at <https://www.w3schools.com>

[2] “Project Proposal Formats” [Online] Available <https://drive.google.com/open?id=13m3kAbj37IZvBpqM5xmU150m8CvuD7_W>

[3] Reger S.Pressman, Ph.D. Seventh Edition, Software Engineering: A Practitioner’s Approach, McGraw Hill, 2010

[4] Database Connection Tutorials Available at: https://www.w3schools.com/php/php\_mysql\_connect.asp [Accessed: 1-Aug-2018]

[5] Concept of building secure web application Available at: <https://www.itproportal.com/2013/08/06/7-steps-for-building-a-secure-web-application/>

[6] Algorithm KNN: <https://www.hindawi.com/journals/cin/2016/1291358/>

[7] Build a Recommendation Engine with Collaborative Filtering (Online Available) <https://realpython.com/build-recommendation-engine-collaborative-filtering/?fbclid=IwAR2slYRQR7nEE2UcvNGyWdhaQvySX581ptJi9ex2j-VtoZHBhZweN3PZ88w>