TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

THAPATHALI CAMPUS

A Minor Project Report
On
Tourist Destination Recommender

Submitted By:

AMAN SHRESTHA	(THA075BCT006)
BIBEK JOSHI	(THA075BCT015)
DIPIN MAINALI	(THA075BCT021)
MANOHAR DAHAL	(THA075BCT026)

Submitted To:

Department of Electronics and Computer Engineering

Thapathali Campus

Kathmandu, Nepal

March, 2022

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

THAPATHALI CAMPUS

A Minor Project Report
On
Tourist Destination Recommender

Submitted By:

AMAN SHRESTHA	(THA075BCT006)
BIBEK JOSHI	(THA075BCT015)
DIPIN MAINALI	(THA075BCT021)
MANOHAR DAHAL	(THA075BCT026)

Submitted To:

Department of Electronics and Computer Engineering,
Thapathali Campus,
Kathmandu, Nepal

In partial fulfillment for the award of the Bachelor's Degree in Computer Engineering.

Under the Supervision of

Er.Suramya Sharma Dahal

March, 2022

DECLARATION

We hereby declare that the report of the project entitled **"Tourist destination recommender"** which is being submitted to the **Department of Electronics and Computer Engineering, IOE, Thapathali Campus**, in the partial fulfillment of the requirements for the award of the Degree of Bachelor of Engineering in **Computer Engineering**, is a bona fide report of the work carried out by us. The materials contained in this report have not been submitted to any University or Institution for the award of any degree and we are the only author of this complete work and no sources other than the listed here have been used in this work.

Aman Shrestha (Class Roll No: THA075BCT006) ________________________

Bibek Joshi (Class Roll No: THA075BCT015) ________________________

Dipin Mainali (Class Roll No: THA075BCT021) ________________________

Manohar Dahal (Class Roll No: THA075BCT026) ________________________

Date: March, 2022

CERTIFICATE OF APPROVAL

The undersigned certify that they have read and recommended to the **Department of Electronics and Computer Engineering, IOE, Thapathali Campus**, a minor project work entitled "**tourist destination recommender**" submitted by **Aman Shrestha, Bibek Joshi, Dipin Mainali** and **Manohar Dahal** in partial fulfillment for the award of Bachelor's Degree in Computer Engineering. The Project was carried out under special supervision and within the time frame prescribed by the syllabus.

We found the students to be hardworking, skilled and ready to undertake any related work to their field of study and hence we recommend the award of partial fulfillment of Bachelor's degree of Computer Engineering.

____

Project Supervisor

Er. Suramya Sharma Dahal

Department of Electronics and Computer Engineering, Thapathali Campus.

____

External Examiner

Er. Anil Verma

Department of Electronics and Computer Engineering, Pulchowk Campus.

____

Project Co-ordinator

Mr. Umesh Kanta Ghimire

Department of Electronics and Computer Engineering, Thapathali Campus.

____

Mr. Kiran Chandra Dahal

Head of the Department,

Department of Electronics and Computer Engineering, Thapathali Campus

March, 2022

COPYRIGHT

The author has agreed that the library, Department of Electronics and Computer Engineering, Thapathali Campus, may make this report freely available for inspection. Moreover, the author has agreed that the permission for extensive copying of this project work for scholarly purpose may be granted by the professor/lecturer, who supervised the project work recorded herein or, in their absence, by the head of the department. It is understood that the recognition will be given to the author of this report and to the Department of Electronics and Computer Engineering, IOE, Thapathali Campus in any use of the material of this report. Copying of publication or other use of this report for financial gain without approval of the Department of Electronics and Computer Engineering, IOE, Thapathali Campus and author’s written permission is prohibited.

Request for permission to copy or to make any use of the material in this project in whole or part should be addressed to department of Electronics and Computer Engineering, IOE, Thapathali Campus.

ACKNOWLEDGEMENT

We owe our deepest gratitude towards the department of Electronics and Computer Engineering, Thapathali campus for providing us the wonderful opportunity to undertake this project. It is an honor for us to express respect and gratitude to our project supervisor, Er. Suramya Sharma Dahal, for providing us with the constant support and feedback related to the project. We'd like to thank our parents and elders for their support and motivation.

Last but not the least, we would like to thank all our teachers and colleagues for their direct and indirect help related to our project.

Any kind of suggestion or criticism will be highly appreciated and acknowledged.

Aman Shrestha	(THA075BCT006)
Bibek Joshi	(THA075BCT015)
Dipin Mainali	(THA075BCT021)
Manohar Dahal	(THA075BCT026)

ABSTRACT

The purpose of the project is to design and develop a recommender system based on web technologies, which utilizes a hybrid recommendation filtering for the smart tourism industry. For the country Nepal where Tourism had been and always have been the major economy booster for the country, introduction of new technologies and web services in this field is a must. Modern technologies of classical recommender system, such as collaborative filtering and Content based filtering are considered to be effectively adopted in the tourism domain. Through our project, recommendation shall be done on basis of composition of data of tourists who have already visited different places of Nepal and the users' own interests using cosine similarity, k-NN and Matrix Factorization algorithm. The project will also act as an informative system with description about many places of Nepal. Our project will help Tourists coming to Nepal act as their own travel agent and plan their trip with effective recommendation provided as well as description of places that are to be visited.

Key Words: Collaborative filtering, content-based filtering, k-NN algorithm, matrix factorization, cosine similarity, hybrid recommendation filtering.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AI	Artificial Intelligence
CB	Content Based
CF	Collaborative Filtering
CSS	Cascading Style Sheets
GDP	Gross Domestic Product
HTML	Hypertext Markup Language
k-NN	k-Nearest Neighbor
MF	Matrix Factorization
ML	Machine Learning
RS	Recommender System

1. INTRODUCTION

1.1 Background

The tourism industry has experienced tremendous growth in recent years. Such a massive leap has been partly attributed to the rapid development of communication and information technology across the globe as well as the widespread use of the internet, which has simplified the process of accessing large amounts of global data from potential customers (tourists) on points of interest, travel plans, and destinations. Currently, e-tourism thrives in both the social and economic sector. This is true for our country Nepal as well. Though a small country, but a blessed country with xanthic natural beauty, tourism industry can be made more flourished with all added benefits to the country. Every year crores of the national revenue are generated from tourism industry with additional benefit of exposure to the world, globalization and exchange of culture with foreigners.

However, in the recent times, global outbreak of pandemic has caused a significant decrease in the number of tourists and their average length of stay. To compensate this loss, an effective system is required which should be able to attract more tourists and retain tourists for more number of days. Exploration of more tourist-undiscovered places within Nepal with a potential of becoming one of the major tourist attractions in the recent future should be done in order to retain tourists for more days and to compensate the recent loss. To achieve this, use of IT is a most.

Table 1-1: Tourist arrival in Nepal 2019/2020

Indicators	**2019**	**2020**	**% change**
Tourists Arrivals By:			
Air	995884	183130	-81.6
Land	201307	46955	-76.7

Total	1197191	230085	-80.8
Average Length of Stay	12.7	15.1	18.9
Sex			
male	626866	124048	-80.2
female	124048	106037	-14.5
Purpose of visit			
Holiday/pleasure	778173	139202	-82.1
Pilgrimage	197786	28530	-85.6
Trekking and mountaineering	171937	35893	-79.1
others	49301	26460	-46.3

In this technological world which is moving each day towards more and more sophisticated technology, esp. achieving its recognition as an "IT world", introducing IT in Tourism Industry can be very fruitful. Our country Nepal is a developing country moving towards the era of modernization and technology but has not achieved much. Regarding the technologies introduced in Tourism industry in Nepal, all we can get is the website maintained by Nepal Tourism Board (NTB), travel agencies with well adapted technologies like online booking of flights, hotels etc. for tourists but serve for pure business purpose, information maintained in sites like Wikipedia, Lonely planets which are not enough and other similar technological acquaintances which are not quite adequate to make the tourism industry as e-tourism.

Interim agency-based software plays a crucial role in bridging the gap between the customers and the system by recommending likely holiday packages and exhilarating tours. Furthermore, such software provides assistance to customers within a specified territory. However, in order to assist customers, many software agencies have developed innovative strategies that provide these customers with information that is useful for planning holiday trips and selecting holiday destinations. Developing an efficient software agency depends strongly on the recommender system (RS) or filtering approach.

In this system, RS plays an important role in improving the customers' level of satisfaction. Generally, RS directly assists customers in finding reputable services, hotels, tours, tickets, restaurants, and others by aggregating and analyzing the demographic data of the customers' reviews. In this system, an analytic technology will be used to compute the probability that a customer will like to travel to a particular destination. In travel agencies, RS-based software suggested suitable holiday packages (tickets, hotel, tour, time) for the customer. Furthermore, RS helps improve the customer's loyalty and satisfaction by recommending the acceptable option which is more related to the request of customer. RS is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item. Generally, recommendation systems are classified into three types, which are: content based filtering (CB), collaborative filtering (CF), and hybrid filtering, which is a combination of former two types.

Content-based filtering methods are based on a description of the item and a profile of the user's preferences. These methods are best suited to situations where there is known data on an item (name, location, description, etc.), but not on the user. Content-based recommenders treat recommendation as a user-specific classification problem and learn a classifier for the user's likes and dislikes based on an item's features. Collaborative filtering filters information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc. They are based on collecting and analyzing a large amount of information on user behaviors, activities or preferences and predicting what users will like based on their similarity to other users**.**

The project "Tourist recommender system" is hence an approach to be of some worth in introducing IT and computerization in field of Tourism. As an IT project, this project could help in promoting tourism industry of Nepal. And along with this, this project as a part of this academic project, which helped us to use knowledge we have learned practically and learn new techniques and trends to develop an application and help us in the academic side too. So, “Tourist recommender system” basically aims in fulfilling both academic and social responsibility and help both tourists and concerned authorities too.

1.2 Motivation

There are various recommender systems in tourism industry, especially in maps which recommend places according to the distance of the place rather than interests of tourists. There are no abundant non-commercial recommender systems to recommend next site to a tourist by analyzing the pattern formed by the places which are already visited by him/her. It would be great if a tourist gets recommendations not solely according to the location or distance or economy, but according to the places he/she visits, i.e. his/her interests and the interests of tourists significantly similar to him/her. This is the motivation which drove us into creating this project.

1.3 Problem Statement

The tourism industry is one of the significant contributors to the country's gross domestic product. According to a report of the World Travel and Tourism Council, the total contribution by the travel and tourism industry to the national gross domestic product (GDP) and employment stood at 6.7 percent and 6.9 percent in 2019. There’s been a significant decrease in the number of tourists visiting Nepal and length of their stay during the pandemic in 2020 and after. The loss occurred due to this problem should be compensated. Tourism industry should be scaled to compensate the loss. If tourism industry is not scaled, there will only be a few limited tourism sites where tourist will come, visit and return. The obscurity of the tourist attractions of Nepal is one of the major cause of tourism industry not reaching its potential and failing to scale properly. So, to retain the incoming tourists for more days and generate more income through tourism industry, obscurity problem of tourism sites should be solved. Also, the interests and mindset of tourists aren’t analyzed properly. If the mindset of tourists is analyzed and approximation of their wishes of next visiting site is done, tourists will be more satisfied and ultimately, tourism industry will grow to its potential.

Our project, “Tourist Destination Recommender”, creates a mathematical model that analyzes the psychology of Tourists according to their visits. Also, our project has a huge database covering famous places as well as many potential tourism destinations which haven’t yet made their way to fame. So, through this project, we are aiming to systematize the tourism sector of the country by recommending the tourists particular destinations depending upon their interests and also helping them discover new places that might be under the horizon of their interests.

1.4 Objective

- To design and develop a web based travel destination recommender system.
- To utilize a combination of k-NN, cosine similarity and matrix factorization algorithm.

1.5 Scopes and Applications

Web-based applications, informative sites etc. in tourism have been a source for visitors to travel across countries and places, especially in country blessed with beauty like Nepal. But with very few information on the web, this system can appear as a boon of people eager to know about Nepal and travel here to get hypnotized in its beauty.

In this project, there will be platform with information and recommendation both embedded into one single application. Similarly, data mining on the records possessed and extraction of information hidden within them will just add a flavor both to tourists and the authorities over here. Hence the scope of this system can be seen broad for boosting tourism in Nepal.

1.6 Report Organization

1 deals with introduction of project, problem statement it intends to solve, projects’ significance in area of problem as well as, as an academic part, objectives it intends to meet and overall need of the project or the system.

2 gives literature review of the project i.e. existing system that can be compared to, related works, information about the existing recommender systems are included.

In 3, theoretical background stating Tourism status and analysis in Nepal, Recommender system's importance and related behaviors are described.

4, requirement analysis, has all the information about the requirements and feasibility analysis for this project.

5, Methodology, gives the information about the system architecture, Use case diagram, class diagram, sequence diagram and ER-diagram.

In 6, Implementation of project is detailed.

7 has the information about the tools that we included in this project. All the tools that we used in project are detailed out.

8 has the results and conclusions of the project.

2. LITERATURE REVIEW

Clearly, we are not the first ones to point out potential benefits of combining collaborative and content based filtering techniques. There has been many field where recommender system has had its role.

In 2006, the online DVD rental company Netflix announced a contest to improve the state of its recommender system [1]. To enable this, the company released a training set of more than 100 million ratings spanning about 500,000 anonymous customers and their ratings on more than 17,000 movies, each movie being rated on a scale of 1 to 5 stars. Participating teams submit predicted ratings for a test set of approximately 3 million ratings, and Netflix calculates a root-mean-square error (RMSE) based on the held-out truth. The first team that can improve on the Netflix algorithm's RMSE performance by 10 percent or more wins a $1 million prize. If no team reaches the 10 percent goal, Netflix gives a $50,000 Progress Prize to the team in first place after each year of the competition. On September 21, 2009, the grand prize of US$1,000,000 was given to the BellKor's Pragmatic Chaos team which bested Netflix's own algorithm for predicting ratings by 10.06%[2].

[3] describes the different methods of computing similarities between item vectors like cosine similarity, Euclidean distance and Pearson's coefficient of correlation. The item vectors can be both the movies or the users which are called user based or item based collaborative filtering respectively. If A and B are the vectors in n-dimensional hyperspace, then the cosine similarity between them can be calculated as

$$\text{cosine similarity} = S_C(A,B) := \cos(\theta) = \frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2}\sqrt{\sum\limits_{i=1}^{n} B_i^2}}, \qquad \text{………(2.1)}$$

Correlation between two users x and y can be measured by computing the Pearson correlation which measures the extent to which two variables linearly relate with each other. For the user-based algorithm, the Pearson correlation between users x and y is

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \qquad \text{………(2.2)}$$

where the i ∈ I summations are over the items that both the users x and y have rated and x̅ and ȳ is the average rating of the co-rated items of the xth and yth user.

[4] describes the k-NN and Naïve Bayes classifier algorithm for text and document mining. Bayesian classifiers are based on a statistical principle. Each processed term is assigned a probability that it belongs to a certain category. This probability is calculated from the occurrences of the term in the training documents where the categories are already known. Given a set of r document vectors D = {d1, …, dr}, classified along a set C of q classes, C={c1, …, cr}, Bayesian classifiers estimate the probabilities of each class ck given a document dj as:

$$P(ck|dj) = \frac{P(ck) * P(dj|ck)}{P(dj)} \dots\dots\dots .(2.3)$$

Where, P(dj) is the probability that a randomly picked document has vector dj as its representation, and P(ck) the probability that a randomly picked document belongs to vector ck.

k-Nearest Neighbor algorithm (k-NN) is a non-parametric classification method first developed by Evelyn Fix and Joseph Hodges in 1951, and later expanded by Thomas Cover. It is used for classification and regression. In both cases, the input consists of the k closest training examples in a data set. A commonly used distance metric for continuous variables is Euclidean distance. For discrete variables, such as for text classification, another metric can be used, such as the hamming distance. In the context of gene expression microarray data, for example, k-NN has been employed with correlation coefficients, such as Pearson and Spearman, as a metric.

[5] describes Stochastic Gradient Descent (SGD) as an iterative algorithm mostly used in matrix factorization and machine learning. SGD applies gradient regarding vectors of the factor matrices to optimize the loss function. The derivatives of the loss function based on W and H are calculated, and the update rules are obtained as following:

$$Wi = Wi - \alpha(ei,jHj - \lambda wHj) \dots\dots.(2.4)$$

$$Hi = Hi - \alpha(ei,j\ Wj - \lambda HWi) \dots\dots.(2.5)$$

where ei,j is the loss (or error) calculated for the rating ri,j in the current iteration. α is the learning rate that can be selected different for each factor matrix, and λ is the regularization parameter used to avoid overfitting.

In the tourism industry of China, [6] has implemented a web-based tourism recommender system consisting of methods like web-crawlers, collaborative filtering and cosine similarity for recommending tourism sites to the users.

Cold start problem in recommendation [7] can be item based and user-based, item based refers to addition of new item in the system and user based refer to addition of new user and it is discussed with the solution. [8] describes how we can combine both content based and collaborative filtering techniques. It discusses cosine similarities and Karl Pearson’s correlation coefficient as a medium for mapping of different users.

3. THEORETICAL BACKGROUND

3.1 Tourism in Nepal

Nepal is the "Land of Paradise" for cultural tourists, trekkers and mountaineers. The natural landscape of mountains, rivers, glacial lakes, forests, wild animals, favorable climate and the hospitality of the Nepalese people and their culture are the main attractions. To sum up Nepal in a single sentence is: Nepal is a mosaic of culture, tradition, adventure, and its ever smiling people nestled beneath towering Himalayas, beautified by rare flora and fauna.

With such rich culture and nature’s gift, Nepal is the best destination of all types of travelers. Tourism in Nepal is a well-fostered business with yearly millions of tourists coming to Nepal from different parts of the world.

People from number of countries visit Nepal every year generating a revenue of millions of dollars and being strong backbone of country’s economy. Tourism in Nepal need to be developed more to get best of it and let the country’s name and beauty prosper in every corner of the world.

3.2 Most visited places in Nepal

Among a lot more places to be explored and visited, very few famous places among them are the targets of many tourists coming to Nepal. Natural beauty like Pokhara, world heritages like Pashupatinath, Swayambhunath, Durbar Squares, Lumbini, birthplace of Gautam Buddha, are the mostly visited places. There are other places like Rara lake, extravagantly beautiful place not much visited, Palpa blessed with immense pleasure of nature, Ilam known for its beautiful tea gardens etc. and many more are places that are equally eligible for attracting tourists but are less visited upon and by very few. So it’s an utmost need to get these places explored and highlighted and enhance the prospect of Tourism in Nepal.

3.3 Tourists’ arrival statistics

Tourists from different countries visit to Nepal. As the ‘Visit Nepal 2021’ slogan became popular across the world, tourist’s arrival increased rapidly from 2015 to 2019.

However, due to global pandemic outburst in late 2019, tourist approval drastically decreased. Nepal government significantly decreased its budget allocated for tourism as well. Also, the overall revenue of Nepal stopped increasing as major revenue collector is the tourism industry. Statistics of tourists' arrival in recent years is as shown below:

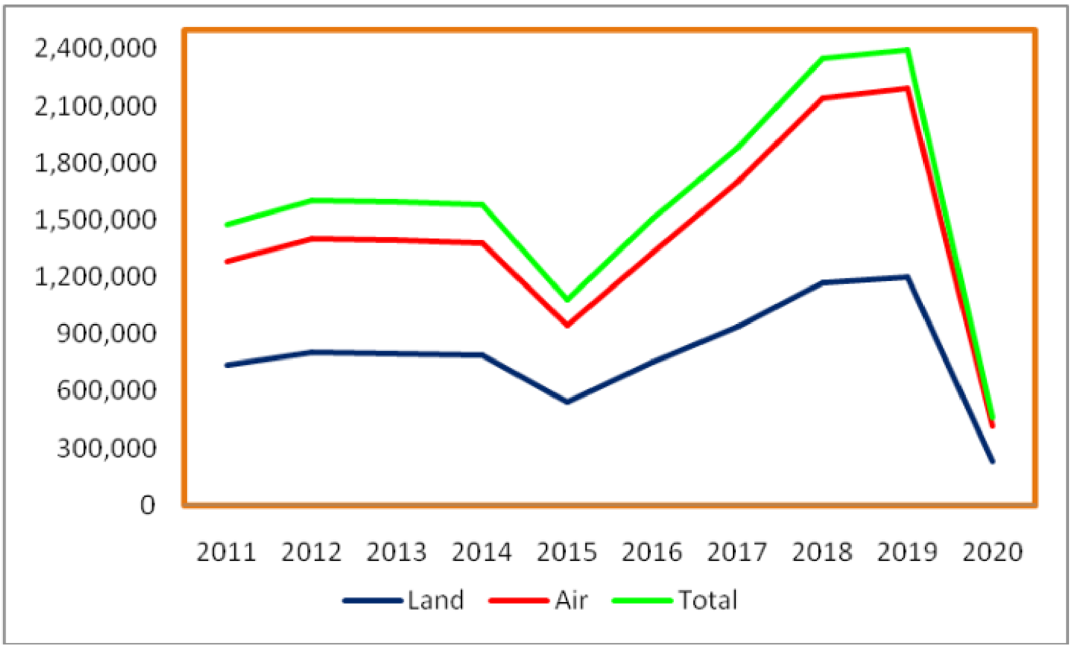


Figure 3-1: Tourists' arrival in Nepal from 2011 to 2020

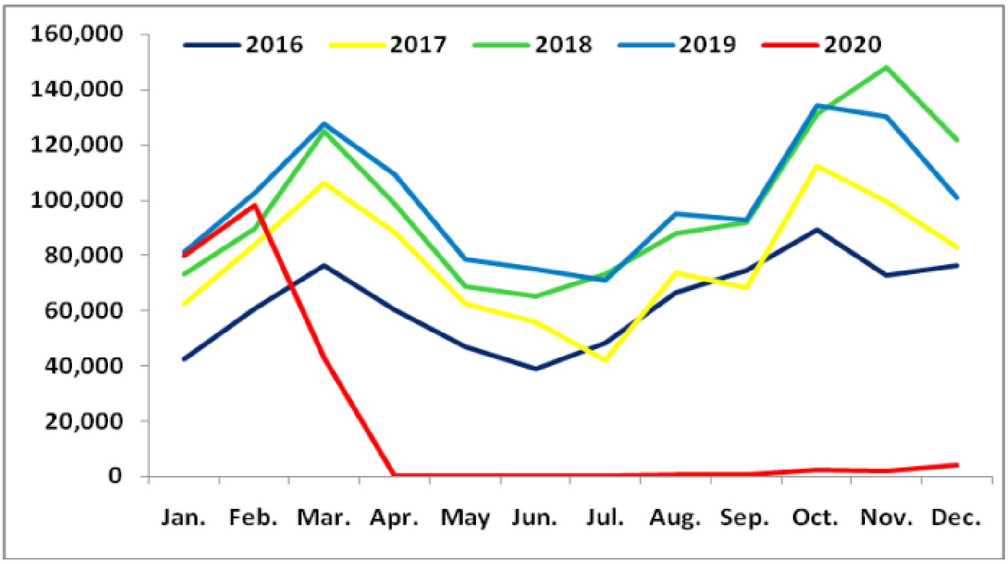


Figure 3-2: Tourists' arrival by month

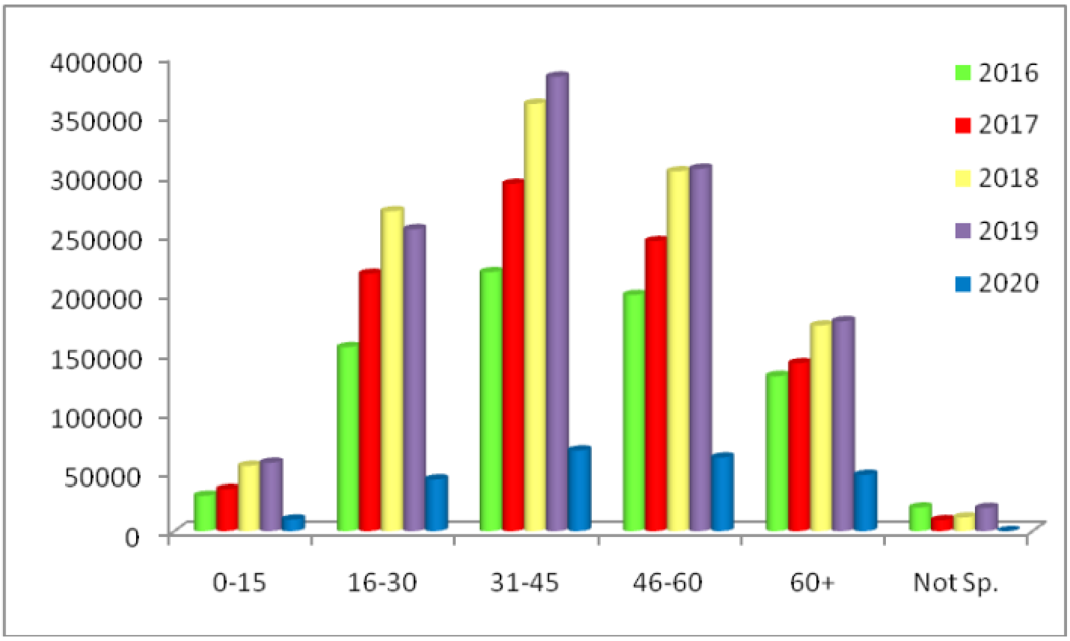


Figure 3-3: Tourists' arrival by age group

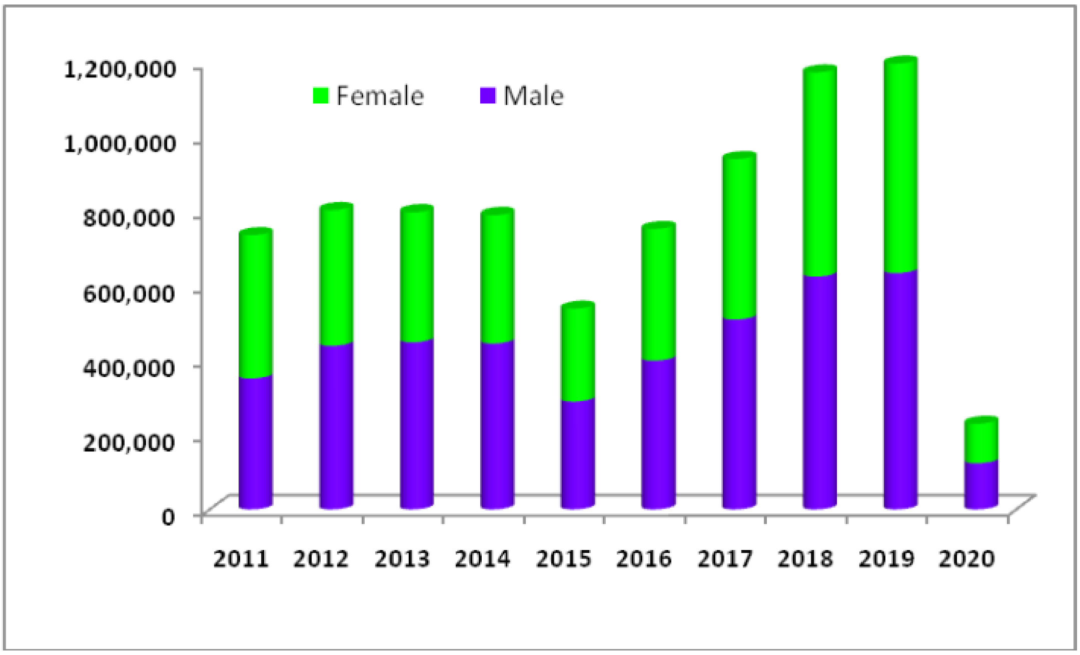
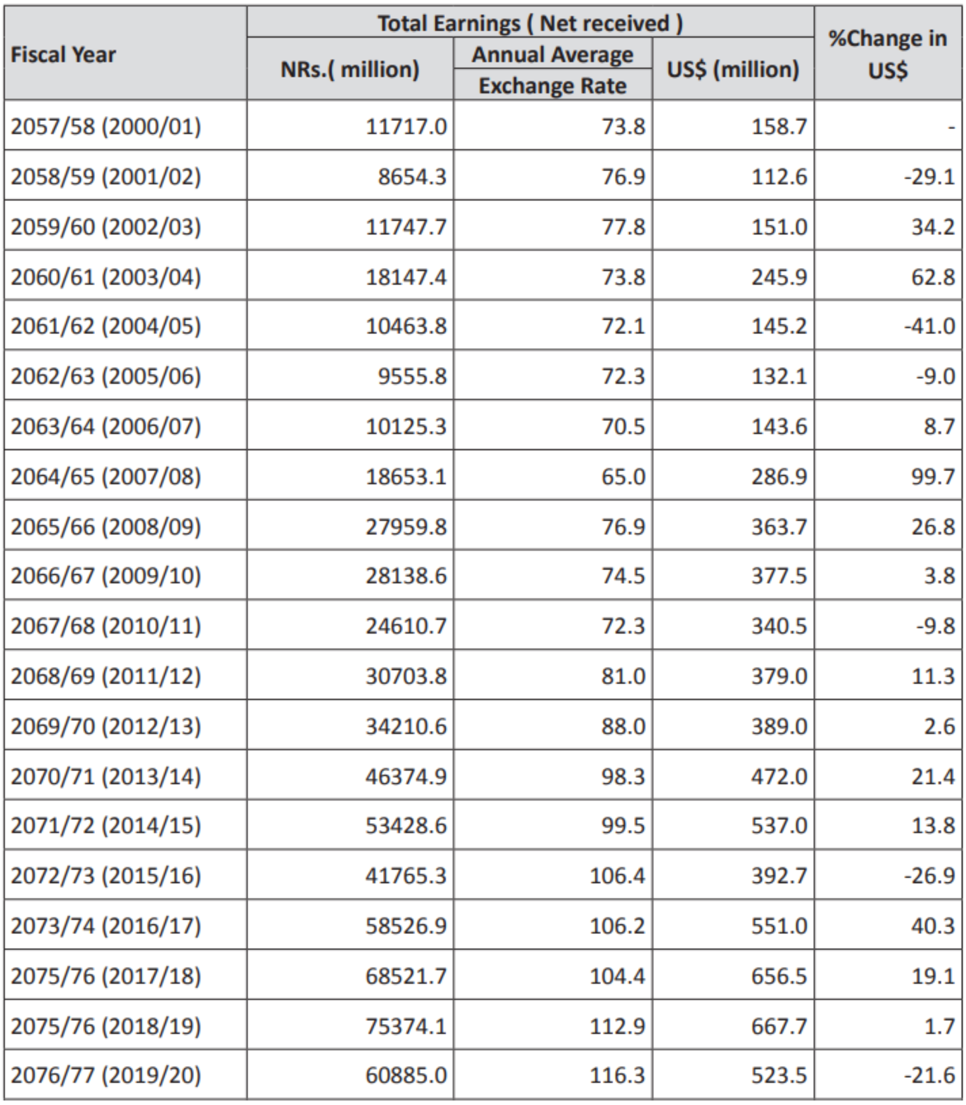


Figure 3-4: Tourists' arrival by sex

3.4 Revenue from Tourism

Tourism industry in Nepal is the major contributing factor to country's economy. It is the backbone for country's economy. Tourism industry yearly generates millions of dollars to the country Nepal. Tourism sector earnings directly correlated with Tourist Arrivals. Due to COVID 19 Tourist Arrivals in 2020 decreased by more than 80%, earning from tourism sectors also decreased by almost 73 % in compare to last year. In 2020, gross foreign exchange earning was NRs. 249595 thousand, (Around 217007 thousand US$). Contrary to decrease in total gross foreign earning, per day expenditure slightly increased to 65 USD per tourist per day in 2020. A statistical report of revenue generation from Tourism industry is as shown in table below:

Table 3-1: Gross foreign exchange earnings from tourism



Fiscal Year	Total Earnings (Net received)			%Change in US$
	NRs.(million)	Annual Average Exchange Rate	US$ (million)	
2057/58 (2000/01)	11717.0	73.8	158.7	-
2058/59 (2001/02)	8654.3	76.9	112.6	-29.1
2059/60 (2002/03)	11747.7	77.8	151.0	34.2
2060/61 (2003/04)	18147.4	73.8	245.9	62.8
2061/62 (2004/05)	10463.8	72.1	145.2	-41.0
2062/63 (2005/06)	9555.8	72.3	132.1	-9.0
2063/64 (2006/07)	10125.3	70.5	143.6	8.7
2064/65 (2007/08)	18653.1	65.0	286.9	99.7
2065/66 (2008/09)	27959.8	76.9	363.7	26.8
2066/67 (2009/10)	28138.6	74.5	377.5	3.8
2067/68 (2010/11)	24610.7	72.3	340.5	-9.8
2068/69 (2011/12)	30703.8	81.0	379.0	11.3
2069/70 (2012/13)	34210.6	88.0	389.0	2.6
2070/71 (2013/14)	46374.9	98.3	472.0	21.4
2071/72 (2014/15)	53428.6	99.5	537.0	13.8
2072/73 (2015/16)	41765.3	106.4	392.7	-26.9
2073/74 (2016/17)	58526.9	106.2	551.0	40.3
2075/76 (2017/18)	68521.7	104.4	656.5	19.1
2075/76 (2018/19)	75374.1	112.9	667.7	1.7
2076/77 (2019/20)	60885.0	116.3	523.5	-21.6

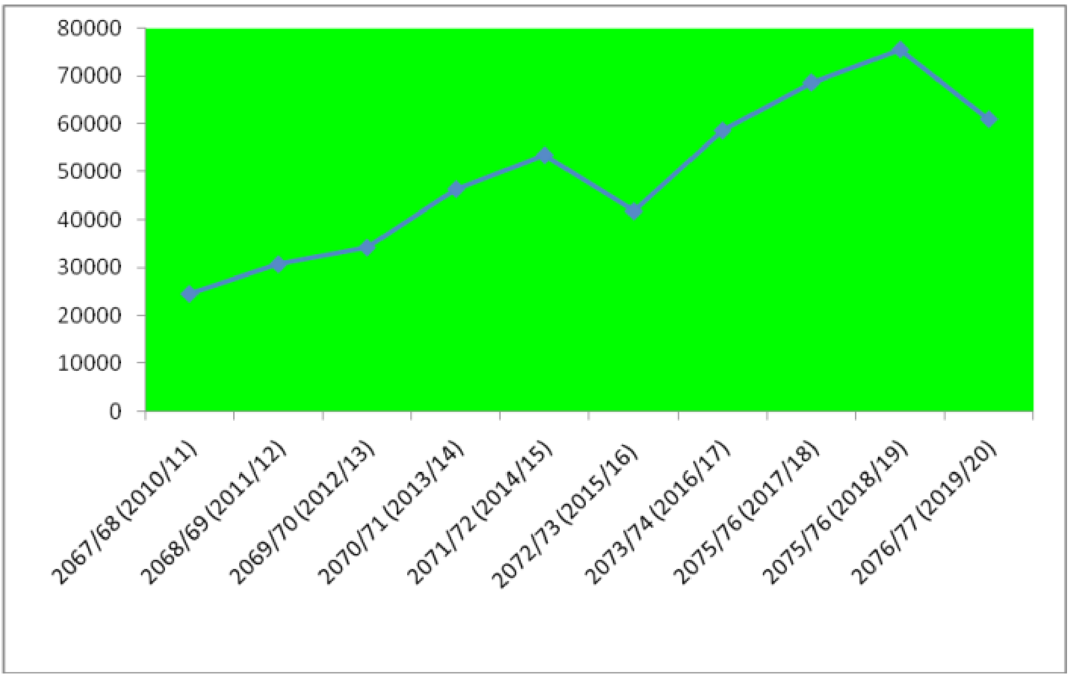


Figure 3-5: Gross foreign exchange earnings from tourism

4. REQUIREMENT ANALYSIS

4.1 Requirement specification

The functional & non-functional requirement for this system are specified below.

4.1.1 Functional requirements

Functional requirements define what the system or application will do - specifically in the context of an external interaction (with a user, or with another system). Our system is all about storing the user details and customer profile data, processing these data, analyzing them and finally producing recommendations through the dataset.

So the system has following functional requirements:

- The system shall store the user interests (tags and parameters of previously visited locations) in suitable format.
- The system shall predict when a particular tourist is likely to favor different kind of places and recommend accordingly.
- The user shall get information about the places.
- The user shall have provision of writing blogs, comments and rating the places.
- If a user is new, s/he shall get recommendation on the basis of their initial preferences.
- User shall be able to see blogs written by other users regarding certain place.
- User shall be able to search and find places from the collection of data.
- User shall be able to create user account and login through registration process and authentication.
- User shall be able to create and edit his/her profile information.

4.1.2 Non-functional requirements

Non-functional requirements are not concerned with the functions of the system. Instead, they look at the criteria to which the software is expected to conform to. Non-functional requirements can include things like response time, security and reliability. It can also be closely tied to user satisfaction. So this system has following non-functional requirements:

- The system shall be easy to use and user friendly with minimum design as possible, so as to make user comfortable with the interface.
- This system shall run on all operating system platforms supporting web surfing.
- This system shall be highly portable and can be used from different web browsers.
- The system should produce content based recommendations in the real time.
- The system should produce results of collaborative filtering in a scheduled time and combine it with the content based recommendations in real time.
- Only the authorized person shall be able to modify the data.
- The system shall be immune to several cyber-attacks.
- There can be no unhandled exceptions from incorrect user input.

4.2 Feasibility Analysis

4.2.1 Economic feasibility

The economic feasibility of the project includes its economic appropriateness with respect to its presented output. If a project provides results of lower significance but requiring higher budgets, then that project can’t be economically viable. For the case of this project, the investment is not a lot in terms of economic aspects. The only expense was of time and skill and the return is large if the project is favored by potential customers.

4.2.2 Technical feasibility

The technical feasibility of the project deals with the availability and its actual implementation ability with the existing tools and techniques available in the software market world. The following are the notable points about the technical feasibility of the project:

- For frontend, we used HTML, CSS, JavaScript, Bootstrap, Tailwind and for backend we used Python and Django.
- For collaborative and content based filtering, we used python libraries like numpy, pandas, scipy, etc.
- SQLite is used to handle our database.

With all these perspectives taken into consideration, the project is technically feasible to implement.

4.2.3 Operational feasibility

As this product is related to database handling & data mining, it relies mainly in the data. So, without data, its existence is impossible. The collection of data is the most challenging part in the application. The required data was collected from Nepal Tourism Board and different other websites. As per our problem statement, we have also collected information regarding the places which have not yet climbed the hill of fame. We collected all the data from various suggestions and information regarding potential tourism sites from journals, blogs, articles, etc. We studied about each new place we discovered and parameterized them in our database. Also, we collected user ratings for various places by conducting a survey.

Since we collected all the data required for the system, the project is operationally feasible to implement.

5. METHODOLOGY

5.1 System block diagram

We have used both content based and collaborative filtering techniques of data mining to recommend travel destination to tourists. k-NN, cosine similarity and matrix factorization algorithm is used using the users' data. During the registration of user, the field of interests of the user is recorded in user details according to which initial recommendation is performed. As the user visits new places and give their rating of that place, k-NN algorithm comes in handy and recommends next recommendation depending upon the given rating recorded in the database. The recommendation provided will be based upon the cosine similarity of places with the user profile.

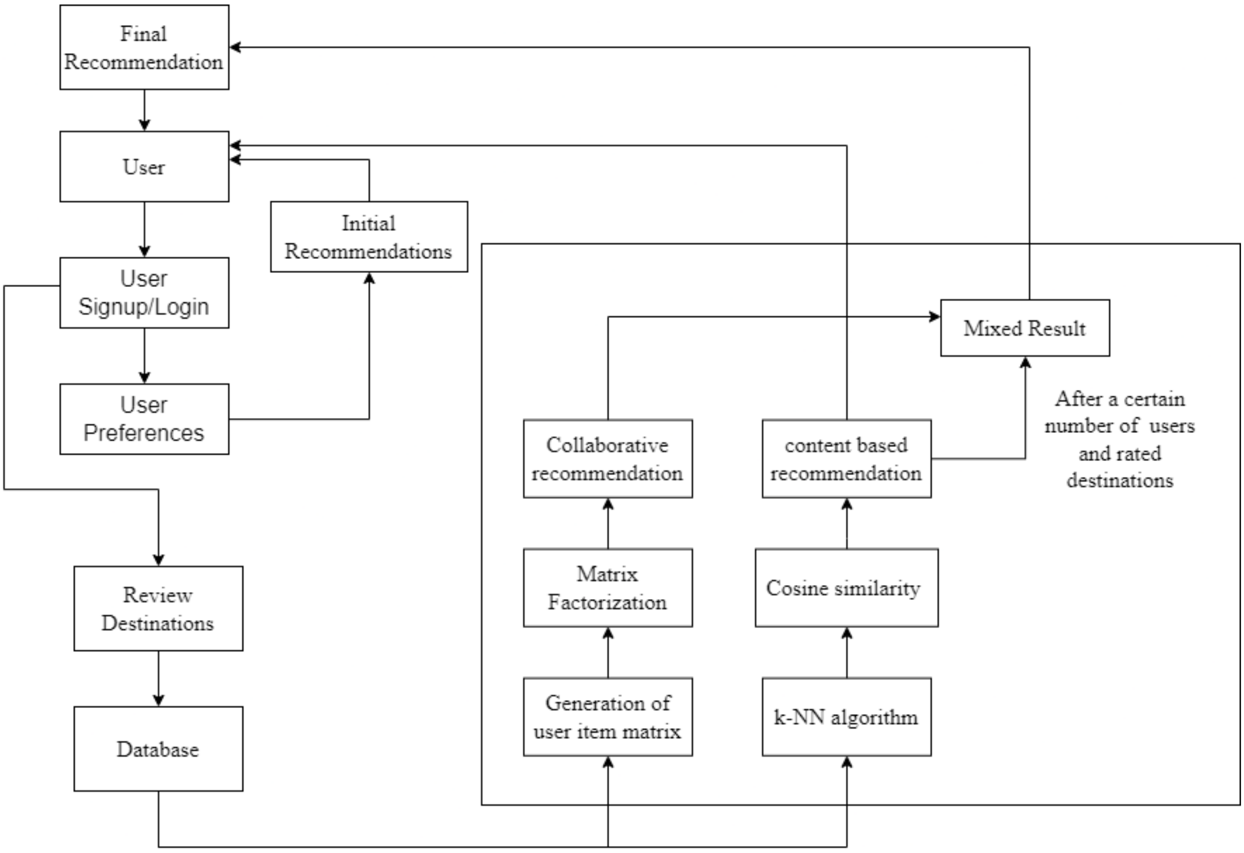


Figure 5-1: System Block Diagram

We have used collaborative filtering technique (matrix factorization) by collecting sufficient number of users and their ratings. The purpose of using content based filtering techniques is to avoid the “cold start” problem. Hence, the union of both gives the ultimate recommendation which will be recommended to the user.

5.2 Description of Working Principle

5.2.1 K-NN Algorithm

It is a machine learning algorithm which calculates minimum distance in multidimensional space to find the nearest neighbor. The algorithm stores all the available cases (unrated destinations) and classifies new cases by majority votes of its K neighbors. When implementing k-NN, the first step is to transform data points into their mathematical values (vectors). The algorithm works by finding the distance between the mathematical values of these points. It computes the distance between each data point and the test data and then finds the probability of the points being similar to the test data. Classification is based on which points share the highest probabilities. The distance function can be Euclidean, Minkowski or the Hamming distance. For our project, we will be using Euclidean distance given as

$$\text{distance} = \sqrt{\left(\sum_{k=0}^{n} (xs_k - x_k)^2 \right)} \quad \ldots\ldots..(5.1)$$

where xs1, xs2, xs3,… xsn are the parameters of unrated destination and x1, x2, x3,… xk are the parameters of rated destinations. For a given value of K, the algorithm will find the k-nearest neighbors of the data point and then it will assign the class to the data point by having the class which has the highest number of data points out of all classes of the k neighbors. After computing the distance, the input x gets assigned to the class with the largest probability.

For the brute-force neighbor search of the k-NN algorithm, we have a time complexity of O(n×m), where n is the number of training examples and m is the number of dimensions in the training set.

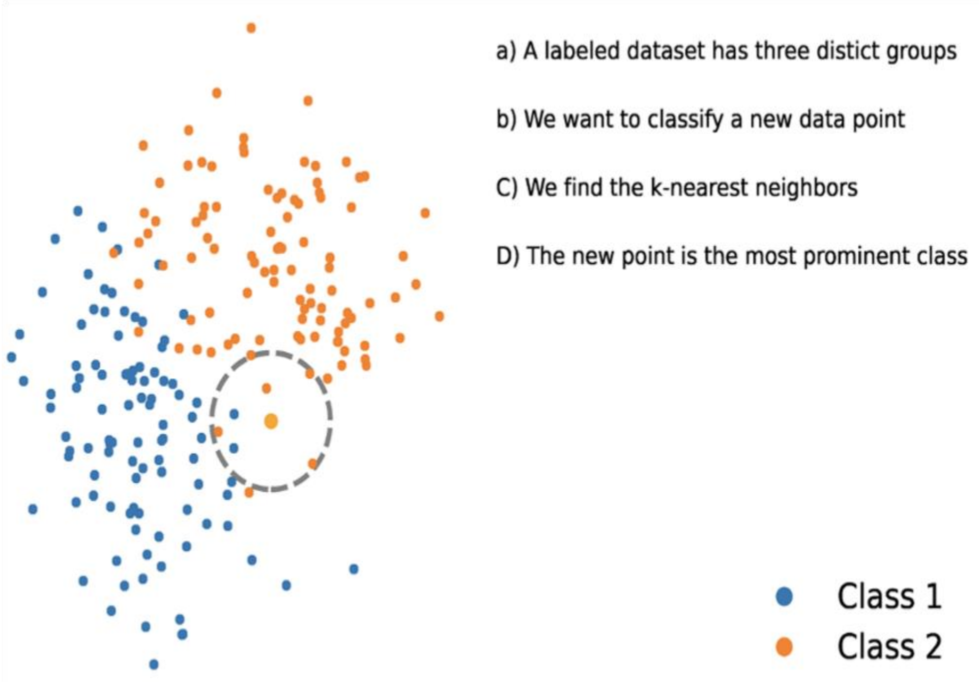


Figure 5-2: k-NN algorithm

.

5.2.2 Matrix factorization

Matrix factorization is a class of collaborative filtering algorithms used in recommender systems. Matrix factorization algorithms work by decomposing the user-item interaction matrix into the product of two lower dimensionality rectangular matrices. In recommendations, the interactions between users and items, or in this case visited destinations, are usually represented as a user-item matrix, where each row/column represents a user/item, and each entry refers to a user’s preference for a particular item. Given a user-item matrix, MF, which is a type of latent factor model, is one of the most popular recommendation methods because of its competence in addressing data scarcity and implicit feedback.

MF decomposes a user-item matrix with unknown entries into a user matrix, which represents the latent factors of the users, and an item matrix, which represents the latent factors of the items; thus, the users and items are mapped onto a low-rank latent space. Then, by estimating the entries in the user matrix and the item matrix, a user-item matrix that is closest to the real one can be obtained through an optimization process, and the unknown entries in the original user-item matrix can be filled.

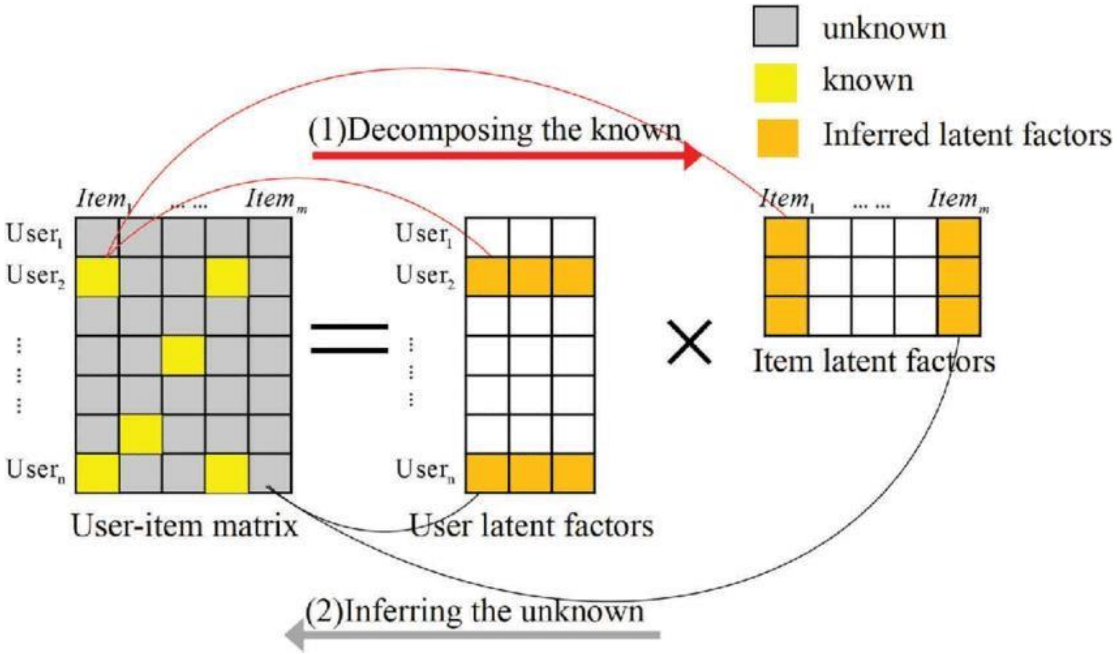


Figure 5-3: Matrix Factorization

5.2.3 Cosine Similarity

Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space. It is defined to equal the cosine of the angle between them, which is also the same as the inner product of the same vectors normalized to both have length 1. From the latter definition, it follows that the cosine similarity depends only on the angle between the two non-zero vectors, but not on their magnitudes. The cosine similarity is bounded in the interval [-1, 1] for any angle θ.

The cosine of two non-zero vectors can be derived by using Euclidian cosine formula:

$$A.B = ||A||\ ||B||\ \cos\theta \ \ldots\ldots(5.2)$$

Given two vectors of attributes, A and B, the cosine similarity, cos(θ), is represented using a dot product and magnitude as

$$\text{cosine similarity} = S_C(A,B) := \cos(\theta) = \frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2}\sqrt{\sum\limits_{i=1}^{n} B_i^2}}, \qquad \ldots\ldots\ldots(5.3)$$

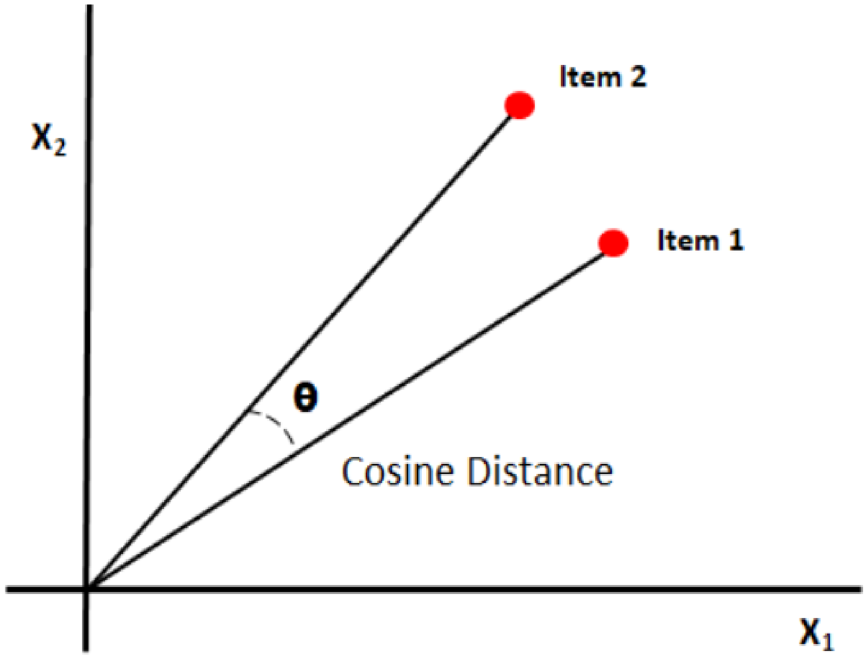


Figure 5-4: Cosine Similarity

5.3 Data Flow Diagrams

A data flow diagram (DFD) illustrates how data is processed by a system in terms of inputs and outputs. As its name indicates its focus is on the flow of information, where data comes from, where it goes and how it gets stored. The below two diagram depicts the level 0 and level 1 data flow diagram of our system respectively. It shows how external entities, user and admin, flows the data to different processes.

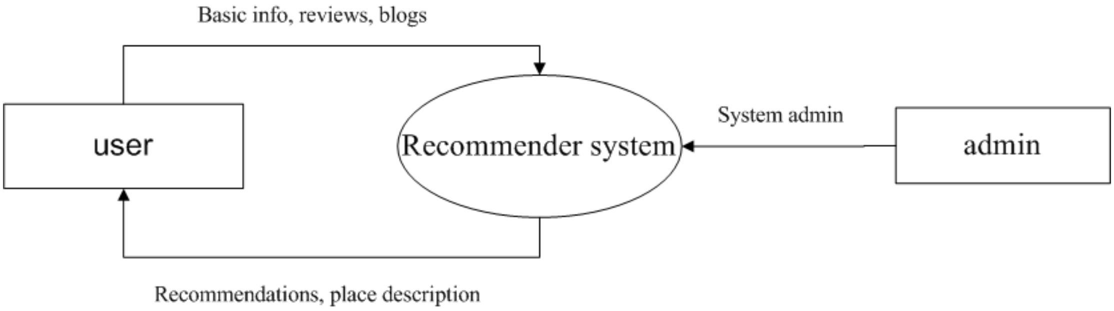

Basic info, reviews, blogs
user
Recommender system
System admin
admin
Recommendations, place description

Figure 5-5: level 0 DFD

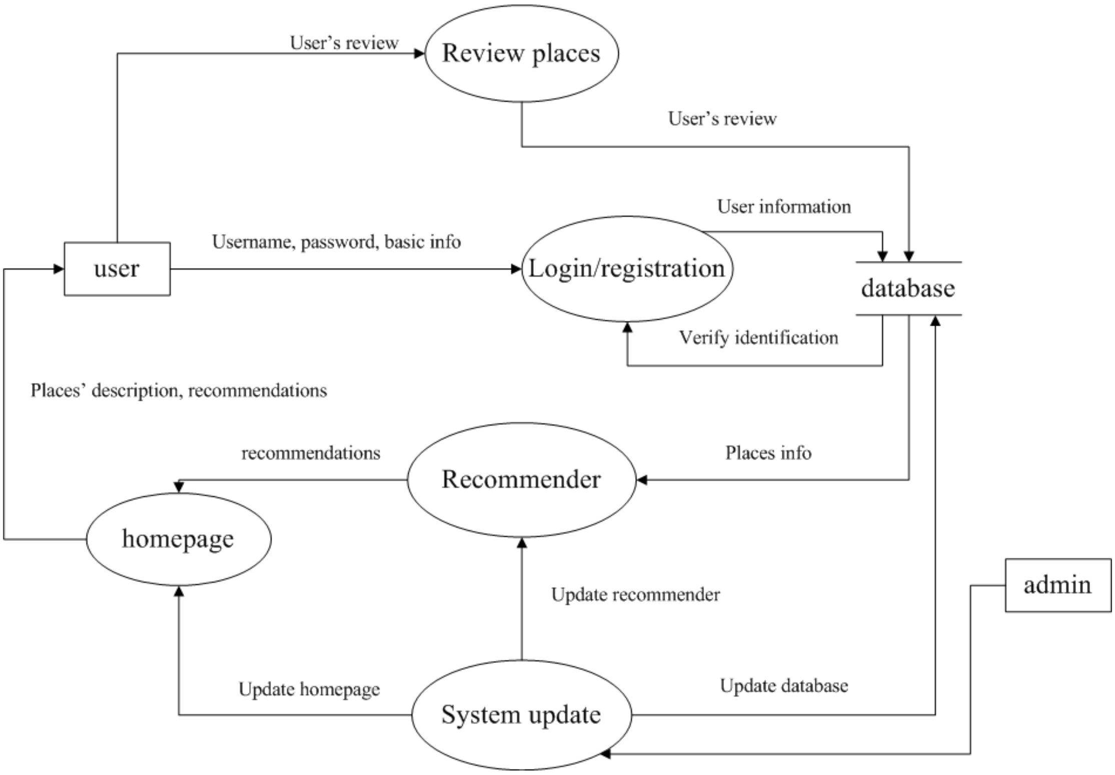


Figure 5-6: level 1 DFD

5.4 Use case diagram

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specification of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. A use case is a list of steps, typically defining interactions between an actor and a system, to achieve a goal. The actor can be a human, an external system, or time.

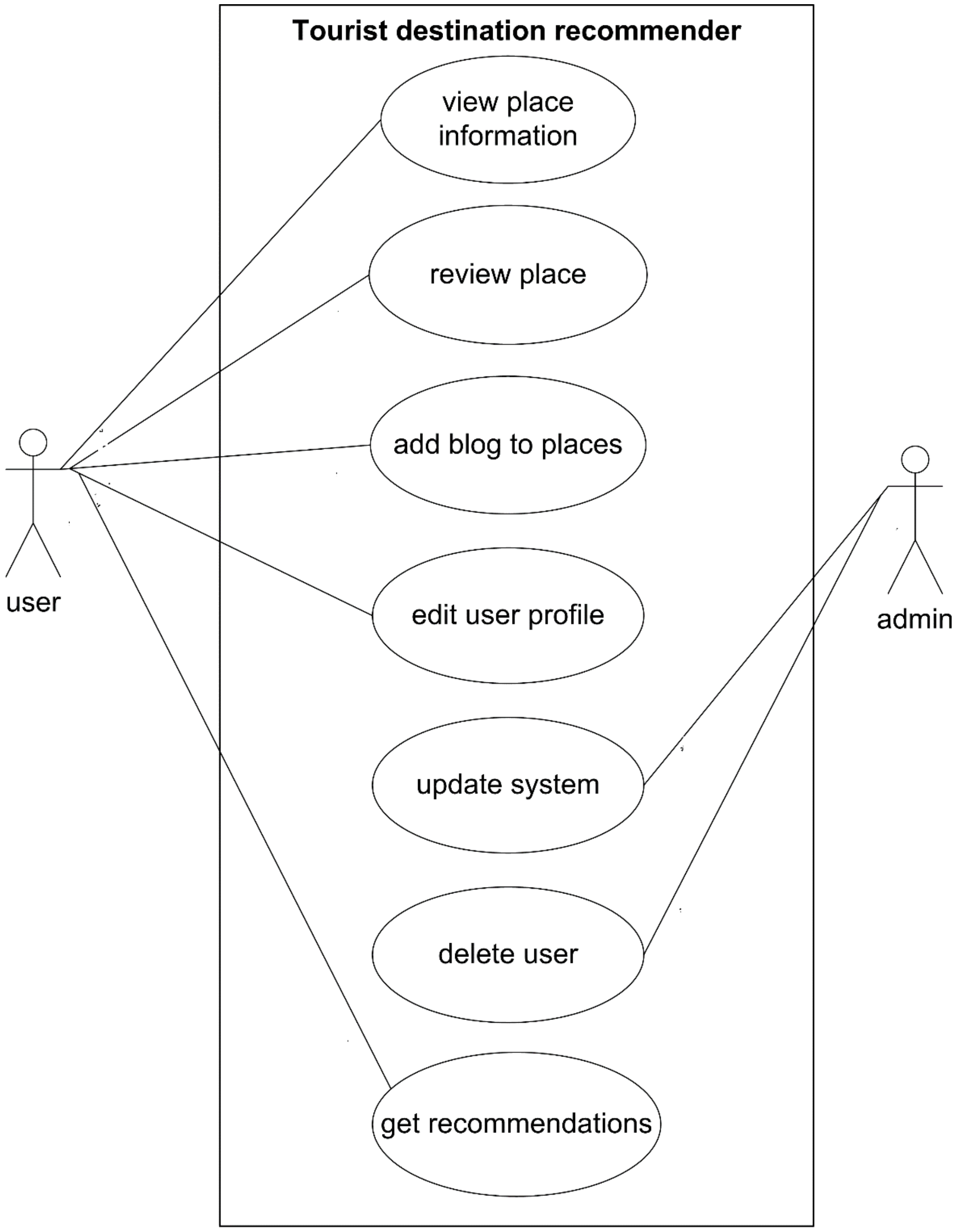


Figure 5-7: Use case diagram

The diagram above depicts the use case diagram of the system that shows the actors and the system. This diagram explains the service provided to the actors by the system. Basically, two kind of actors are explained in the system viz. registered user and system admin. A registered user is allowed to view the places, review the place, rate the place, add blog to the place and avail the service of recommendation and suggestion of places. The user can also update its account. The system admin is responsible for system update and removing of users.

5.5 Sequence diagram

Sequence diagram represents the step by step operations in any system. This explains the sequential steps followed in any system to perform an operation. Sequence diagram is an interaction diagram that show how process operate with one another and in what order. Sequence diagram models the interaction of objects arranged in time sequence. It carries the request of the one object as message and gets reply in the same form if necessary, and the operation(s) are carried out thereafter. Sequence diagrams are sometimes called, event diagrams, or event scenarios.

The figure below shows the sequence diagram of this system that displays the agents, and the operations of the different objects and their sequences of time. In the figure, there is an agent, registered user. The registered users are allowed to view as well as rate and review about the places, add blogs and also he gets recommendation from the system according to his attributes.

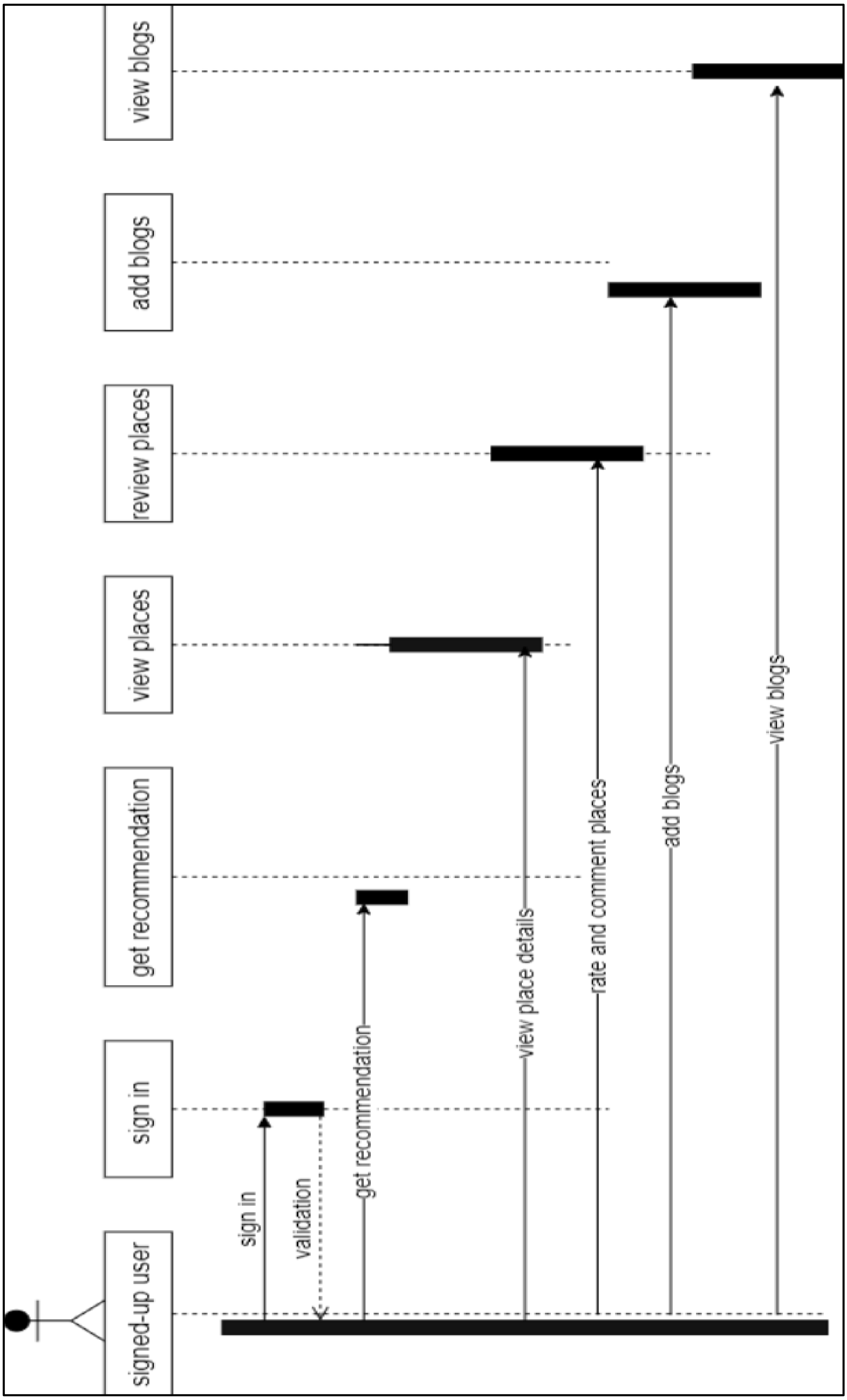


Figure 5-8: sequence diagram

5.6 Class diagram

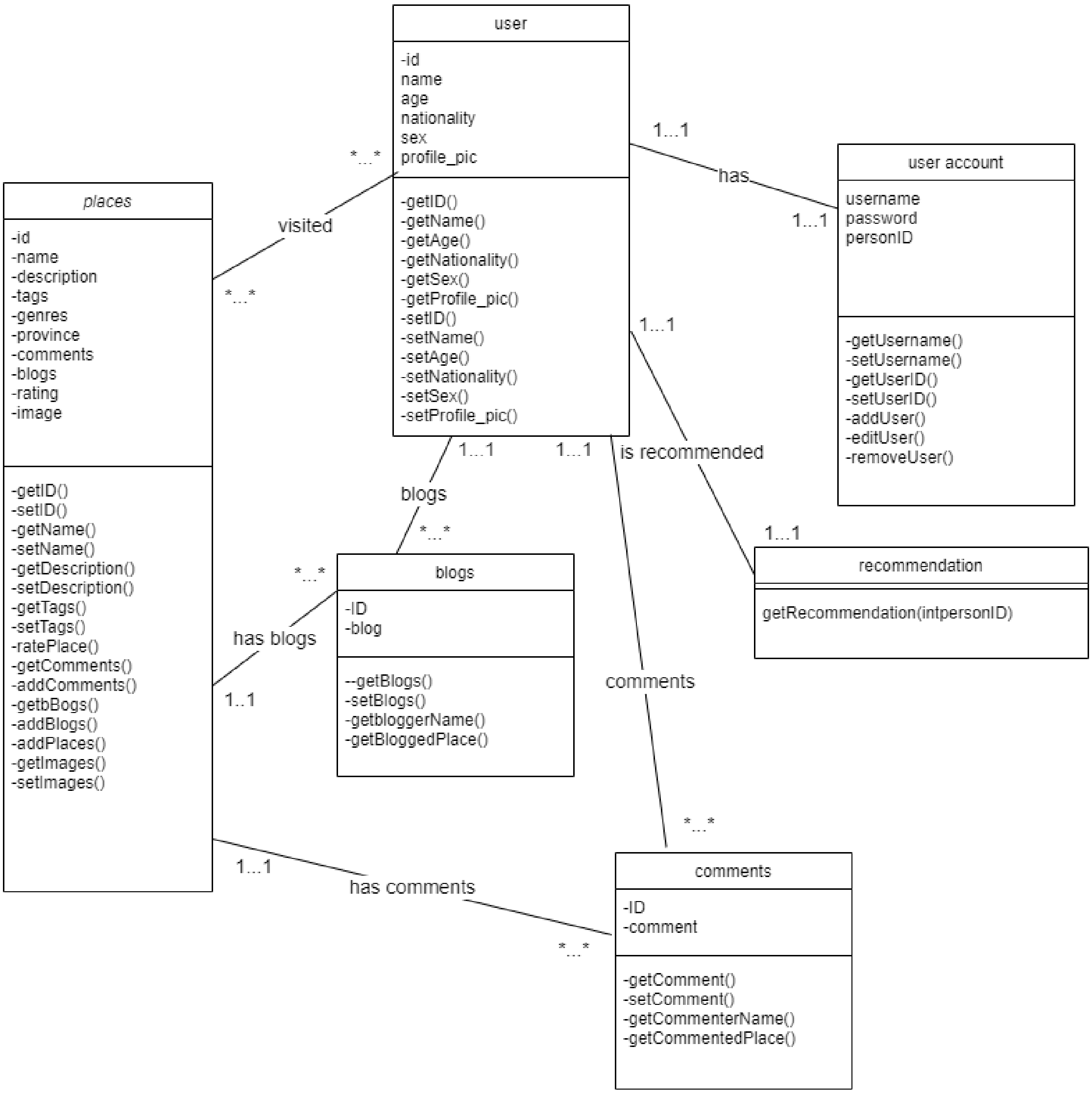


Figure 5-9: class diagram

The above class diagram shows the interaction occurring among the different classes. There are six different classes viz. places, user, user account, blogs, recommendation and comment. These classes are developed with the purpose of breaking down the system into smaller making the system easier to build. The Place class is used to keep the details about places and the operations regarding the places like rating, adding blogs and commenting. The user class provides the details about the user. The user account class updates the person’s information. This also validates a user during login. The Comment class keeps the records of comments provided by the users about the places and help the system to add/display the comments. The Blog class keeps the records of blogs provided by the users about the places and help the system to add/display the blogs.

The purpose of using the Recommendation class is to provide the recommendation to the user according to the characteristics as well as history of the user. The relations between the classes are shown by the lines connected through the classes. The label on the line explains the relation between the classes. The multiplicity is displayed at the end of each line, which represents how the classes are related to each other. In the above diagram, we can say that a person can have only one user account, and a user account can have details about only one person. Similarly, a person can have zero or more than zero places visited, and a place can be visited by zero or more than one person.

5.7 ER- diagram

ER diagram (entity-relationship diagram) is a diagram which shows entities, attributes of entities and relationship between them. ER diagram of our system is as shown below:

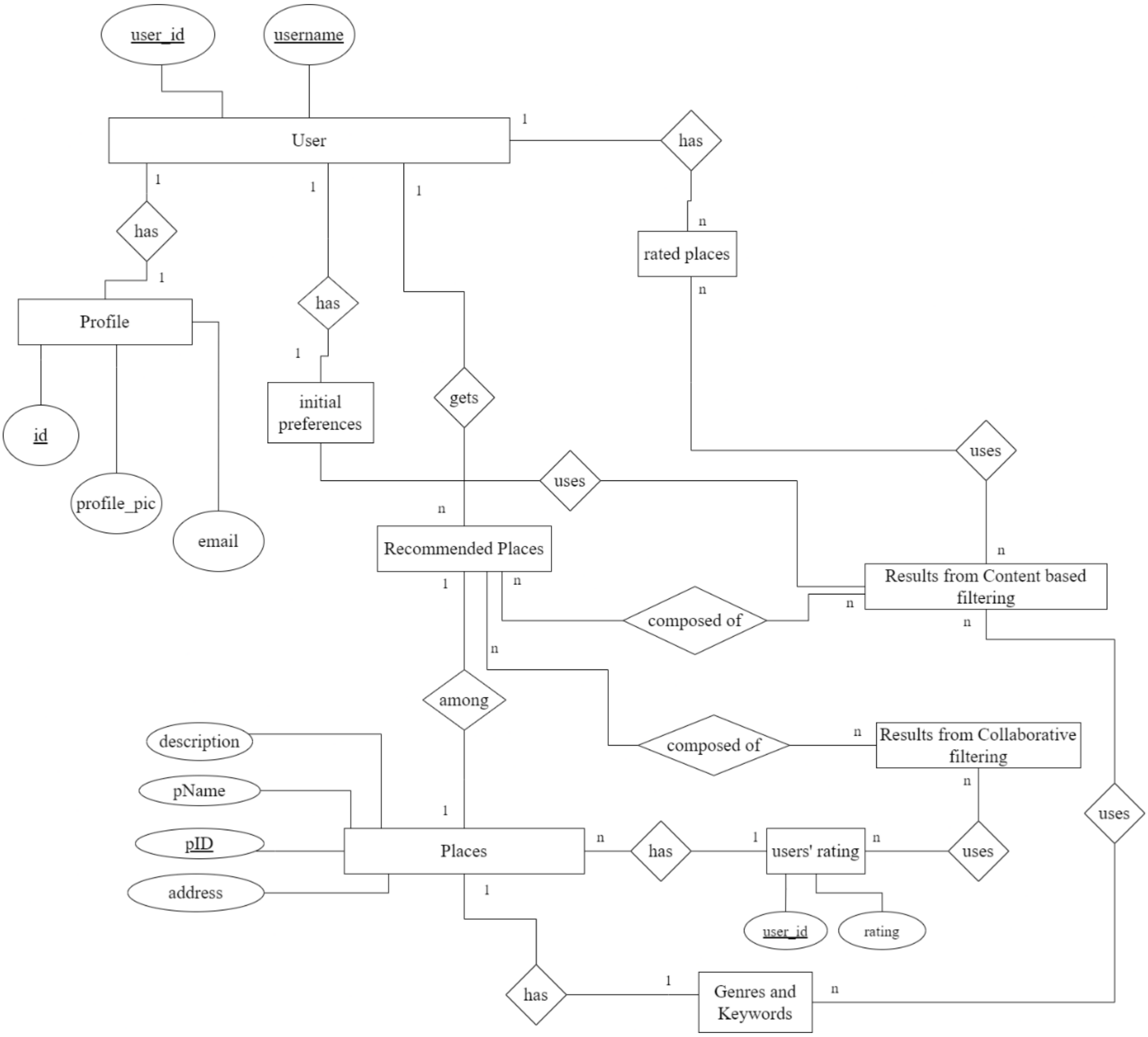


Figure 5-10: ER diagram

6. IMPLEMENTATION DETAILS

6.1 Data Collection

Data Collection was the main challenging part of this project. Data domain needed for this project includes many places with genres and keywords related to each place. Those genres and keywords were given according to the reviews and information of each place from different blogs, websites and social media. 503 different places from all 7 provinces of Nepal are collected in our current database.

Each place collected in the dataset was given a unique place ID and classified into genres. A place may be constituted of many genres. For example, a single place can be adventurous as well as religious. Also, keywords were given to each place according to the nature of place and things that they are famous for. Later on, the genres and keywords were converted into binary format and combined as one to give 'parameters' of each place. Those parameters are later used as a vector with n dimensions, with n being the number of elements in the parameters, for the distance calculation in recommender system.

We also collected many pictures for each place, and we collected general information about each place. Images were of two types – the thumbnail image and the destimages (images of the destination), which were to be displayed inside the detailed view of each place in our website.

For content-based recommendation, there was no need to collect the ratings prior to the ratings given by a specific user of the website, because it didn't depend upon the ratings given by other users. However, to implement the collaborative filtering, only this much data wasn't sufficient as there were no ratings in the places prior to the deployment of our website. Thus, we had to conduct a survey to collect ratings before implementing collaborative filtering.

We took made 10 samples of dataset, each sample including 50 random places from our database, and we sent each sample to different set of people to collect ratings of the places that they already had been to. We collected ratings of about 100 users prior to implementing the collaborative filtering algorithm.

6.2 Design of database

The database of the system uses the SQLite database, mainly related to the user's information, data including ratings and information of the places, similarity ranking information, etc.

The main data models created in our project till now are listed below:

Table 6-1: Genre-info of Places

Field	Type	Description
pID	IntegerField	Primary key for unique identification
place	OneToOneField	Linking Genre info with description and images of place
pName	CharField	Name of the place
Culture	BooleanField	Whether the place belongs to the genre 'culture'
Wildlife	BooleanField	Whether the place belongs to the genre 'wildlife'
Sightseeing	BooleanField	Whether the place belongs to the genre 'sightseeing'
Adventure	BooleanField	Whether the place belongs to the genre 'adventure'

History	BooleanField	Whether the place belongs to the genre ‘history’
religious	BooleanField	Whether the place belongs to the genre ‘religious’
Child_friendly	BooleanField	Whether the place belongs to the genre ‘child_friendly’
Tags	CharField	Tags or Keywords related to the place
Province	IntegerField (1 to 7)	Province number of the location of place.

Table 6-2: Description of the places

Field	Type	Description
ID	IntegerField	Primary key for unique identification
Genre-info	OneToOneField	Linking the place and its genre-info
Name	CharField	Name of the place
Description	TextField	Description of the place
Thumbnail	ImageField	Thumbnail of the places

Table 6-3: User Profile

Field	Type	Description
FirstName	CharField	First name of the user
MiddleName	CharField	Middle name of the user
LastName	TextField	Last name of the user
username	CharField	Unique username of the user
email	EmailField	Email of the user
age	IntegerField	Age of the user
sex	CharField(choices)	Gender of the user
Profile_pic	ImageField	Profile pic of the user
Phone_no	CharField	Phone number of the user
Nationality	CharField	Nationality of the user

Similarly, our other data models are:

- **Comments:** It stores the comments provided by the users associated with each object of places model.
- **Destimages:** It stores the url pointing towards the images associated with every instance of places model.
- **Placerating:** It stores the ratings provided by the users to each instance of places model.
- **Blog:** It stores the blogs posted by users associating with instances of places model.

- **Preferences:** It stores the preferences chosen by every authenticated user initially. These preferences are used to show the initial recommendations.
- **Blogimages:** It stores the images posted in the blogs created by the users.
- **Mf_result:** It stores the result given by the matrix factorization algorithm.

These models were created inside 'models.py' of different apps with the help of Django development platform. Different classes were created for each model. Inside the blog app, blog model is connected to blogimages model through foreignkey, i.e. one to many relationships. That means, each blog may contain many blog images but each blog image will be associated with only one blog. Also, destinations (places) model is related to destimages model in a similar way. Similarly, blog model is connected with Django's inbuilt user model through one-to-one relationship, i.e. each blog is associated with only one user. The 'comment' model is connected to user as well as places model, i.e. Each comment is associated with one user and one place. There are similar other connections between these models which are achieved through inheritance of different classes.

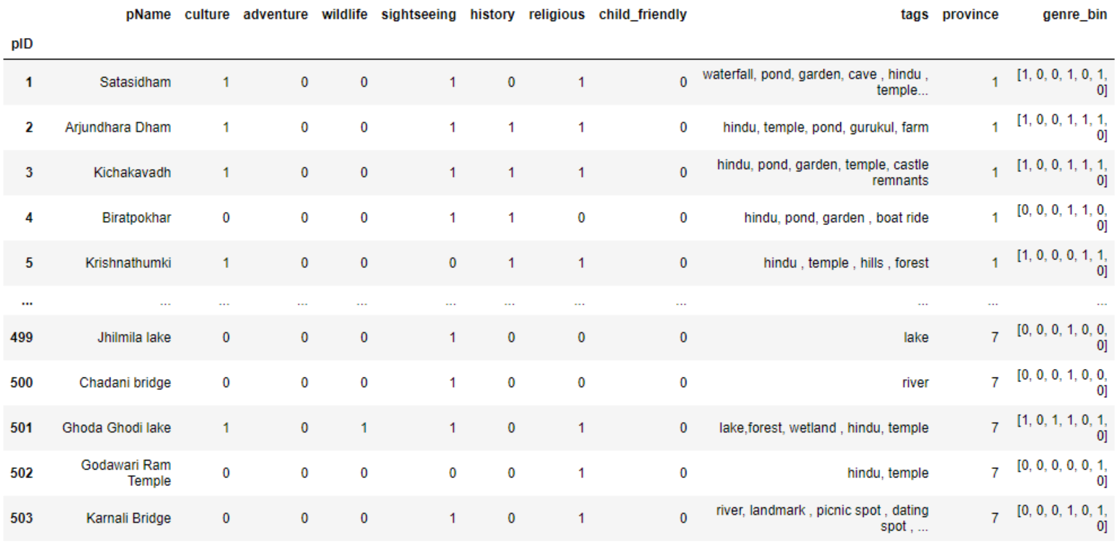
6.3 Implementation of Recommendations:

Recommendations was the core part in this project. For any recommender system there requires user's mood or interest. Recommending outside the user's interest makes the system unusable. The problem was "How to recommend?" In this system we observed 2 types of users.

6.3.1 New users:

The users who just signed up to our website are new users. The main problem faced while recommending places to new users is that new users will not have given ratings to places already. So, we collect preferences of user during his/her first log in. The preferences is a list of binary values of length 7, which is equal to the total number of genres of places that we have. The list is used as a 7-dimensional vector point to calculate the cosine similarity between each place and the user's preferences. The place's genres are also converted into list called 'genre_bin' prior to calculating the initial preferences. Our dataframe just before computing the initial recommendations looks like this:

Table 6-4: dataframe after assigning genre_bin



pID	pName	culture	adventure	wildlife	sightseeing	history	religious	child_friendly	tags	province	genre_bin
1	Satasidham	1	0	0	1	0	1	0	waterfall, pond, garden, cave , hindu , temple...	1	[1, 0, 0, 1, 0, 1, 0]
2	Arjundhara Dham	1	0	0	1	1	1	0	hindu, temple, pond, gurukul, farm	1	[1, 0, 0, 1, 1, 1, 0]
3	Kichakavadh	1	0	0	1	1	1	0	hindu, pond, garden, temple, castle remnants	1	[1, 0, 0, 1, 1, 1, 0]
4	Biratpokhar	0	0	0	1	1	0	0	hindu, pond, garden , boat ride	1	[0, 0, 0, 1, 1, 0, 0]
5	Krishnathumki	1	0	0	0	1	1	0	hindu , temple , hills , forest	1	[1, 0, 0, 0, 1, 1, 0]
...	...	...	...	...	...	...	...	...	...	...	...
499	Jhilmila lake	0	0	0	1	0	0	0	lake	7	[0, 0, 0, 1, 0, 0, 0]
500	Chadani bridge	0	0	0	1	0	0	0	river	7	[0, 0, 0, 1, 0, 0, 0]
501	Ghoda Ghodi lake	1	0	1	1	0	1	0	lake,forest, wetland , hindu, temple	7	[1, 0, 1, 1, 0, 1, 0]
502	Godawari Ram Temple	0	0	0	0	0	1	0	hindu, temple	7	[0, 0, 0, 0, 0, 1, 0]
503	Karnali Bridge	0	0	0	1	0	1	0	river, landmark , picnic spot , dating spot , ...	7	[0, 0, 0, 1, 0, 1, 0]

Cosine distance between the genre_bin and user's preferences is calculated and is sorted in ascending order (from nearest to farthest) and top 30 of the result is shown in the home-page.

6.3.2 Existing users

For the existing users, i.e. users who have already rated some places, we've used a mixture of content based as well as collaborative filtering.

a. Content based filtering

For content based filtering, the 'tags' was converted into binary list (vector) and combined tags and genre_bin to give parameters or profile of each place. Then, a user's profile was generated based on the places he/she has rated. A 'weighted profile' of every place was generated using the rating data.

For example, if user visited 5 places containing 2 places of parameter A and 3 places of parameter B, then user's profile becomes: A = 2/5, B = 3/5. This gives a probability of finding place of a genre in the user's rating data.

After that, weighted profile of each place was generated using ratings given by the user to places of different genres. For example, out of 2 places he visited of parameter A, he rated them 1 and 4. Also, out of 3 places he visited of parameter B, he rated them 2, 3 and 5. Now,

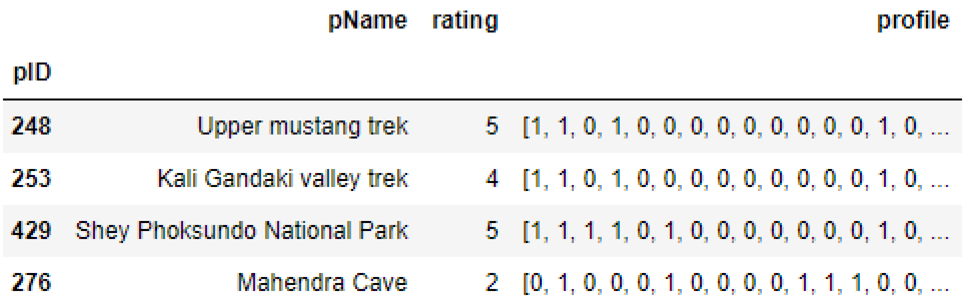
Weighted profile for parameter A = (-2+1)/2 = -0.5

Weighted profile for parameter B = (-1+0+3)/3 = 2/3

Here, since 3 rating is an average among the ratings, it represents neutral preference. Ratings 4 and 5 are considered as liked and ratings 2 and 1 are considered as disliked. The ratings used in calculating the weighted profile is calculated by subtracting the average, i.e. 3 from the ratings given by the user. Then, averaging is done inside each parameter to calculate its weighted profile. After that, these waited profile of parameters are assigned to each place having these parameters and are converted to list or vector.

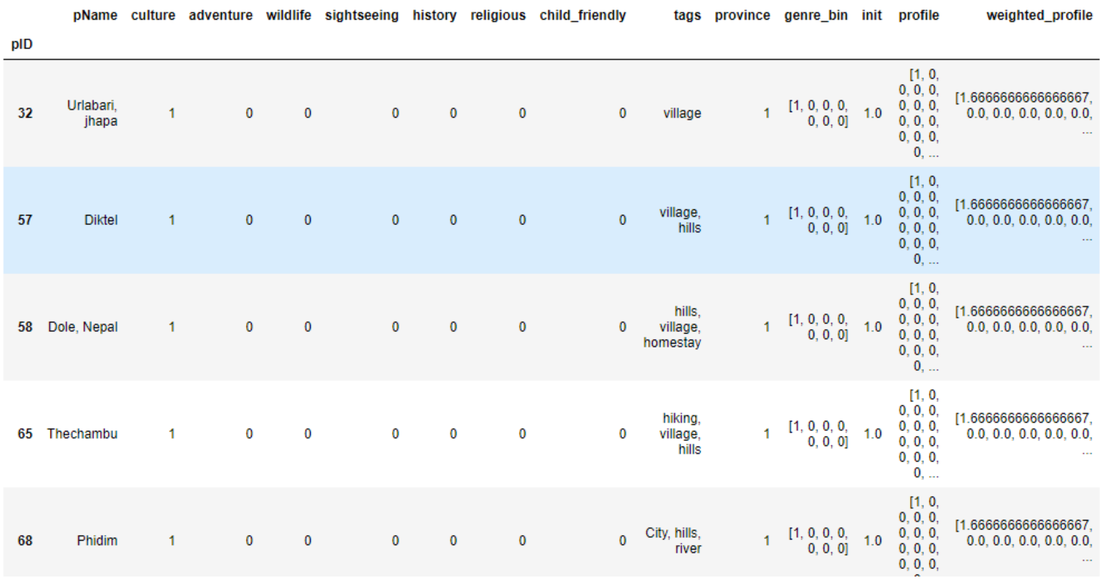
An example of dataframe for a specific user after calculating weighted profile for each place is as shown below:

Table 6-5: sample ratings provided by a user



	pName	rating	profile
pID			
248	Upper mustang trek	5	[1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, ...
253	Kali Gandaki valley trek	4	[1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, ...
429	Shey Phoksundo National Park	5	[1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, ...
276	Mahendra Cave	2	[0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, ...

Table 6-6: Sample dataframe after weighted profile is calculated for each place



	pName	culture	adventure	wildlife	sightseeing	history	religious	child_friendly	tags	province	genre_bin	init	profile	weighted_profile
pID														
32	Urlabari, jhapa	1	0	0	0	0	0	0	village	1	[1, 0, 0, 0, 0, 0, 0]	1.0	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1.6666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, ...
57	Diktel	1	0	0	0	0	0	0	village, hills	1	[1, 0, 0, 0, 0, 0, 0]	1.0	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1.6666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, ...
58	Dole, Nepal	1	0	0	0	0	0	0	hills, village, homestay	1	[1, 0, 0, 0, 0, 0, 0]	1.0	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1.6666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, ...
65	Thechambu	1	0	0	0	0	0	0	hiking, village, hills	1	[1, 0, 0, 0, 0, 0, 0]	1.0	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1.6666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, ...
68	Phidim	1	0	0	0	0	0	0	City, hills, river	1	[1, 0, 0, 0, 0, 0, 0]	1.0	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,	[1.6666666666666667, 0.0, 0.0, 0.0, 0.0, 0.0, ...

Now we've got a user's profile which ranges from 0 to 1 and place profile ranging from -2 to 2 in each parameters. Both profiles or vectors are of same dimensions.

Firstly, for each of the total places in our database, we calculated 'k' number of neighbors from the rated places. For optimum result, we've used k = square root of n where n=number of places rated. If k is in float, we convert it into integer and if it's even, we make it odd. Then, out of k nearest neighbors, we find out the frequency of liked places and disliked places. Ratings greater than or equal to 3 is considered liked and disliked otherwise. So, each of the total places are classified into 'likeable' and 'dislikeable' classes. We only filtered the places with 'likeable' classes and passed them on to further processing.

The places of likeable classes were taken and cosine similarity of each place with user's profile was calculated.

$$cosine\ similarity = \frac{\vec{i}\,.\vec{j}}{i * j} \ldots\ldots\ldots (6.1)$$

Where 'i' is the user's profile and 'j' is the places' profile. Normalization of places' profile wasn't really necessary because it came in the range of -2 to 2 and dividing by 2 doesn't make a difference because it's simply divided in the nominator and denominator and cancelled out. However, the cosine similarity has the range of -1 to 1 because of the negative values.

At last, the likeable places were sorted in descending order according to the cosine similarity and top 30 results were returned.

b. Collaborative filtering

since content based filtering perform recommendation only based upon user past history and activity. It cannot help users to discover new interests. This was solved by using collaborative filtering technique in which matrix factorization was performed on user rating matrix. Matrix factorization is simply a mathematical tool to factorize a matrix into two low dimensional matrices. In collaborative filtering recommendation, various users are taken into account. Here, recommendation in done using the feedback matrix collected from number of user from their activity.

There is no domain necessary- the embeddings are naturally learned.

Matrix factorization is a way to generate latent features when multiplying two different kinds of entities. Collaborative filtering is the application of matrix factorization to identify the relationship between items (here places) and users' entities. With the input of users' ratings on the places, we would like to predict how the users would rate the places so the users can get the recommendation based on the prediction.

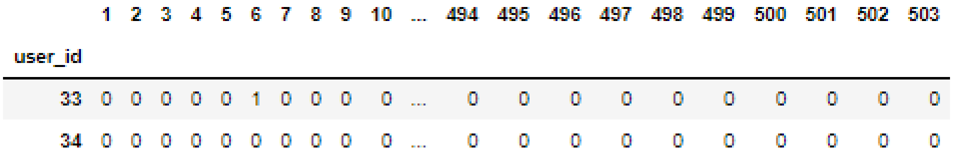
We Defined a set of Users (U), places (D), R size of |U|, and |D|. The matrix |U|*|D| includes all the ratings given by users. The goal is to discover K latent features. Given with the input of two matrices matrices P (|U|*k) and Q (|D|*k), it would generate the product result R. Here k signifies latent features

i.e hidden features which are calculated according to user rating. Here, we have taken k=7. Given with the input of two matrices matrices P (|U|*k) and Q (|D|*k), it would generate the product result R.

Matrix P represents the association between a user and the features while matrix Q represents the association between an item and the features. We can get the prediction of a rating of a place by the calculation of the dot product of the two vectors corresponding to u_i and d_j. To get two entities of both P and Q, we initialize the two matrices randomly and calculated the difference of the product named as matrix M. Next, we minimize the difference through the iterations.

The method is called stochastic gradient descent, aiming at finding a minimum of the difference i.e. minimizing RMSE (cost function). Places not rated by user are assigned with 0 in the user place rating matrix which was predicted by dot product of two matrices and places are recommended to the user to those places whose ratings were predicted greater or equal to 3. Cold start problem in recommendation in handled by content based recommendation.

Table 6-7: sample user's ratings in places

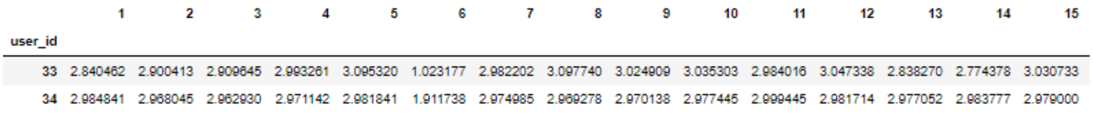


	1	2	3	4	5	6	7	8	9	10	...	494	495	496	497	498	499	500	501	502	503
user_id																					
33	0	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

Here rows indicate users and column indicates places and cell value indicates rating of the user. The predicted matrix is generated below. As you can see, the predicted matrix has similar output with the true values, and the 0 ratings are replaced with the prediction based on the similar users' preferences on places.

We can see that for existing ratings we have the approximations very close to the true values, and we also get some 'predictions' of the unknown values. With the feature given as 7, the algorithm is able to associate the users and items to three different features, and the predictions also follow these associations.

Table 6-8: sample result obtained from matrix factorization



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
user_id															
33	2.840462	2.900413	2.909645	2.993261	3.095320	1.023177	2.982202	3.097740	3.024909	3.035303	2.984016	3.047338	2.838270	2.774378	3.030733
34	2.984841	2.968045	2.962930	2.971142	2.981841	1.911738	2.974985	2.969278	2.970138	2.977445	2.999445	2.981714	2.977052	2.983777	2.979000

Hence we can recommend places to the specific user. Recommendation in done with matrix factorization in our system only when the number of rating given by a user exceeds 50 and no. of users exceeds 50 to increase the prediction of the algorithm. Since sparsity of the matrix decreases the performance of the algorithm

Since matrix factorization in our system performs training continuously on users rating matrix recommendation is not possible at runtime hence we schedule computation of it at midnight daily and all the ratings are taken in consideration at that computation. Django apscheduler library was used to schedule the task periodically.

c. Mixing the results

After sufficient number of ratings, there are two results obtained from content based and collaborative filtering. We've combined them and shown in the front page. Combination of two results are shown in the 'top recommendations' and 'other recommendations' section. In the 'top recommendations' section, there are places which is among the top 15 of result of content based filtering and also calculated rating above 4 by matrix factorization. The semantic meaning of this mixing can be stated as, "if a place is among top 15 of the nearest likeable places in terms of content and also its probable rating is above 4 calculated by collaborative filtering, then it deserves to be on top recommendation as both filtering have shown." Then, the remaining results are shown as a union and is shuffled in every reload and shown random 30 out of all recommendations, so that the user isn't bored by getting same thing in his homepage over and over again. The places in the 'top recommendations' section are shuffled among themselves and the places in the 'other recommendations' section are shuffled among themselves.

6.4 System security

Django comes with different inbuilt security features for websites. Those features prevent attacks and data leaks through anonymous users. Some of those security functions are:

6.4.1 CSRF (Cross site request forgery) protection

CSRF attacks allow a malicious user to execute actions using the credentials of another user without that user's knowledge or consent. Django has built-in protection against most types of CSRF attacks, if enabled and used appropriately. CSRF protection works by checking for a secret in each POST request. This ensures that a malicious user cannot "replay" a form POST to the website and have another logged in user unwittingly submit that form. The malicious user would have to know the secret, which is user specific (using a cookie).

In each HTML, while sending and receiving data, we've used CSRF tokens so that all the data will be protected from CSRF attacks. Django forms are used to take data from authorized users.

6.4.2 SQL injection protection

SQL injection is a type of attack where a malicious user is able to execute arbitrary SQL code on a database. This can result in records being deleted or data leakage.

Django's querysets are protected from SQL injection since their queries are constructed using query parameterization. A query's SQL code is defined separately from the query's parameters. Since parameters may be user-provided and therefore unsafe, they are escaped by the underlying database driver.

6.4.3 Password Hashing

Passwords of each user is saved in database with the help of strong hashing function which can't be viewed even by the system admin. It's too strong to be decoded. It uses SHA256 and PBKDF2 algorithm for password hashing.

7. TOOLS AND PLATFORMS USED

7.1 Languages

7.1.1 Python

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small- and large-scale projects. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library. Python is considered the best language for data science and machine learning due to its adequate number of mathematical and machine learning libraries like numpy, pandas, scipy, matplotlib, etc. and is used for content and collaborative filtering in our project.

7.1.2 Javascript

JavaScript often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. Over 97% of websites use JavaScript on the client side for web page behavior, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on users' devices. JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard. It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). Javascript is used in our project for DOM manipulation.

7.1.3 HTML

The Hyper Text Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content.

7.1.4 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting. In our project, CSS is used for styling html forms and containers.

7.2 Libraries

7.2.1 Numpy

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. The ancestor of NumPy, Numeric, was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open-source software and has many contributors. NumPy is a NumFOCUS fiscally sponsored project. We have used Numpy for both content and collaborative based recommendations.

7.2.2 Pandas

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself. Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007 to 2010. Pandas is also used in our project for data manipulation and recommendation generation.

7.2.3 SciPy

SciPy is a free and open-source Python library used for scientific computing and technical computing. Scipy contains modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal and image processing, ODE solvers and other tasks common in science and engineering. We have used Scipy in our project for calculation Euclidian distance for KNN algorithm for content based recommendation.

7.2.4 Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of Matplotlib. We have used matplotlib in our project for data visualization and data plotting.

7.2.5 APScheduler

Advanced Python Scheduler (APScheduler) is a Python library that lets you schedule your Python code to be executed later, either just once or periodically. You can add new jobs or remove old ones on the fly as you please. If you store your jobs in a database, they will also survive scheduler restarts and maintain their state. When the scheduler is restarted, it will then run all the jobs it should have run while it was offline. APscheduler is used in our project to schedule the running of collaborative filtering algorithm (matrix factorization).

7.3 Frameworks

7.3.1 Django

Django is a Python-based free and open-source web framework that follows the model–template–views (MTV) architectural pattern. It is maintained by the Django Software Foundation (DSF), an independent organization established in the US as a 501(c)(3) non-profit. Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes reusability and "pluggability" of components, less code, low coupling, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings, files, and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models.

7.3.2 Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components. As of August 2021, Bootstrap is the tenth most starred project on Github, with over 152,000 stars, behind freeCodeCamp (over 328,000 starts), Vue.js Framework, React library, TensorFlow and others. Bootstrap has been used extensively in our project for page styling and responsiveness.

7.3.3 Tailwind

Tailwind is a utility-first CSS framework packed with classes flex, pt-4, text-center and rotate-90 that can be composed to build any design, directly in our markup. We have used tailwind for creating containers and divs for blogs, “about us” page and recommendation lists.

7.4 Tools

7.4.1 Jupyter Notebook

Jupyter Notebook (formerly IPython Notebooks) is a web-based interactive computational environment for creating notebook documents. A Jupyter Notebook document is a browser-based REPL containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media. Underneath the interface, a notebook is a JSON document, following a versioned schema, usually ending with the ".ipynb" extension. It is used as an interpreter for running codes for content and collaborative recommendation in our project.

7.4.2 Visual Studio Code

Visual Studio Code is a source code editor by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. In the Stack Overflow 2021 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool, with 70% of 82,000 respondents reporting that they use it. We have written our overall project in VS Code.

7.4.3 Ms Visio

Microsoft Visio (formerly Microsoft Office Visio) is a diagramming and vector graphics application and is part of the Microsoft Office family. It was used to develop different diagrams for project management. Different diagrams like class diagram, use case diagram, sequence diagram, and timeline chart. This tool was a very efficient tool for developing such diagrams that really explains the purpose of the system, as well the properties, functions and different aspects of the system are clear through such diagrams.

8. RESULTS AND ANALYSIS

The figure below represents the functional structure of our system.

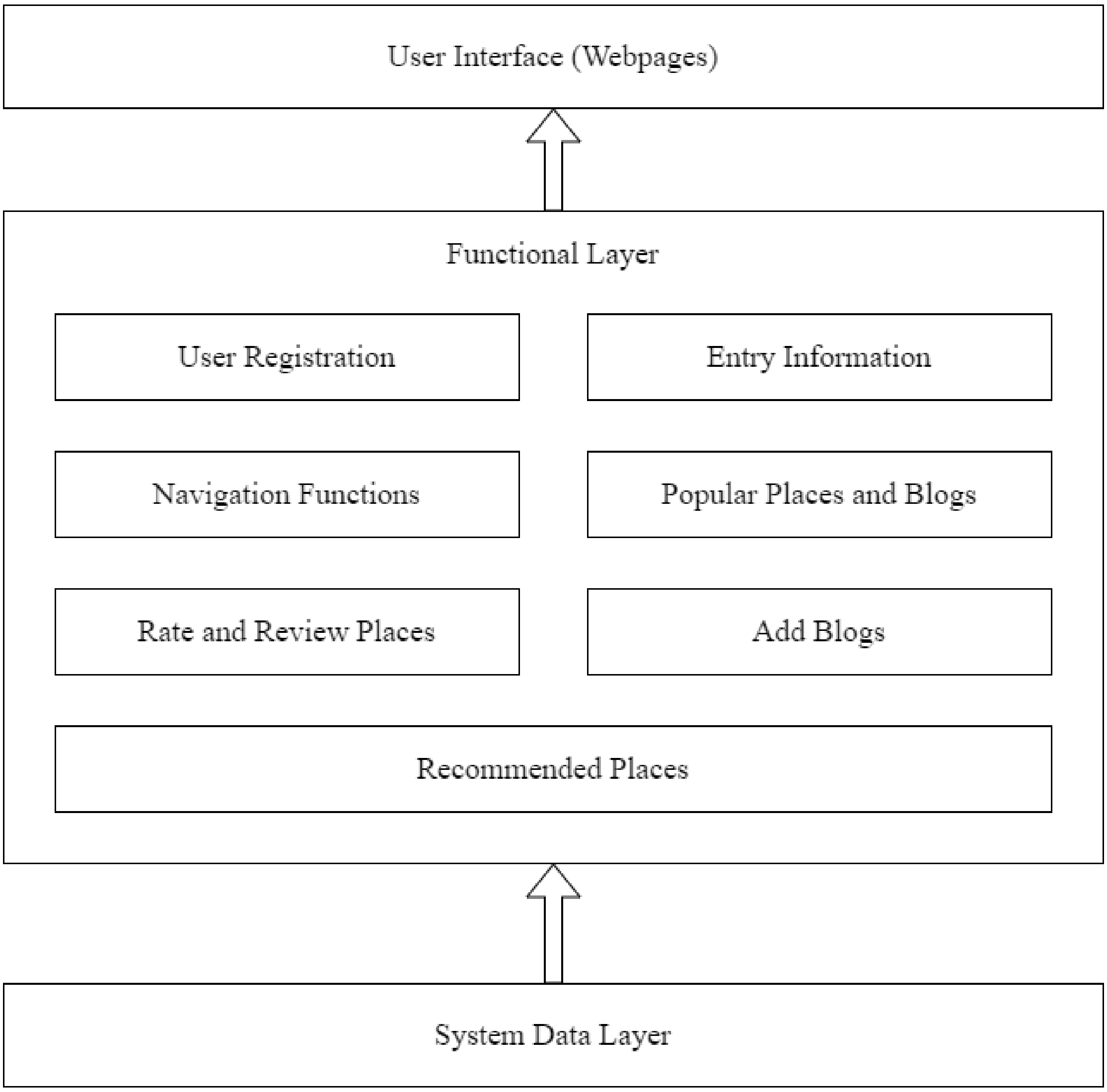


Figure 8-1: System function structure

The function modules are described as follows:

8.1 User registration:

Whenever a user enters to our website, initially, he/she is directed to the login page. There’s a form with the fields to enter username and password. After entering and verifying the information, he/she is redirected to the home page customized for the user. If he/she hasn’t registered already, then there’s a link directing towards the Sign-up page which has a form to register the user. The registration form contains fields to enter unique username, email, password1 and password2 (confirm password). After registering, the user will again be redirected to the login page to log in with the newly created account.

8.2 Entry Information:

After the login, if the user is new, the user will be redirected to a form consisting of fields to enter profile information of the user along with his/her preferences. The user can enter information such as firstname, middlename, lastname, profilepic, age, sex, nationality, phone number and preferences. The preferences include the user's likeability of each genre in terms of integer values from 0 to 5.

8.3 Navigation functions

These functions include the functions to redirect the user to different pages. The pages included in the navigation section are Homepage, Blogs, Profile, about us, Contact us with Log-out button.

8.4 Popular places and blogs

List Popular places and blogs are shown in the homepage and blog pages respectively. The list is shown with the thumbnails associating with the places and blogs. When a user clicks any of those thumbnails, the user will be redirected to a page consisting of detailed information of the clicked instance of place or blog.

8.5 Recommended Places

8.5.1 Initial recommendation

For a new user, when he/she creates an account and logs in for the first time, he/she gets redirected to profile setup form where he/she can select preference from different categories. And according to those preferences, he/she gets initial recommendation in the homepage. Main page consists of popular places and recommended places.

8.5.2 Content based recommendation

After the user rates certain places, according to those ratings, list of recommendations is displayed to user under “recommendations for you”. It uses content based filtering techniques. Content will run for the user whose number of rating is less.

8.5.3 Hybrid recommendation

Using matrix factorization technique, collaborative recommendation is formed and its intersection with the content based recommendation with some conditions is provided to the user under “top recommendations for you” and remaining of list of matrix factorization is combined with content based in the ‘recommend for you’ section.

8.6 Rate and Review Places

After getting recommendations, the user can view, visit and rate the places. He/she can also add comments as a form of review. The user can also search for places on his/her own and rate or review the place which will be reflected in the database.

8.7 Add blogs

User can add blogs related to certain place.

9. FUTURE ENHANCEMENTS

Until now, We've collected data for the places and created various data models to store the data and also to link them with other models. We've created a secure user registration and log-in system and created an interface for the users in form of a website, from where they can view information of some places and some blogs related with those places.

We have also provided each user a customized or personalized homepage which shows recommendation unique to only the user. The objectives of our project have successfully been achieved until now. However, to enhance our project furthermore, lots of things can be done. Some of them are as follows:

- Web-crawlers can be used to give more specific recommendations
- Implicit ratings (view-time, no. of views, clicks, searches, etc.) can be embedded in the system so as to enhance the recommendations.
- Hotel recommendations can be added with an economical point of view.
- Neural network and advanced machine learning can be used for predicting the user's exact preferences.

10. APPENDICES

Appendix A. Used terminal commands:

virtualenv . : to setup the virtual environment for the project

activate : to activate the installed virtual environment

pip install django : to install django in the virtual environment

django-admin startproject <projectname> : starts the project with projectname

python manage.py createsuperuser : to create a superuser to access the admin panel

python manage.py startapp <appname> : to create an app with appname

python manage.py makemigrations <appname> : to make migrations of models created in appname.

python manage.py migrate: to create and update models in database.

python manage.py runserver: to run the server.

Python manage.py shell_plus –notebook: to run notebook with Django environment

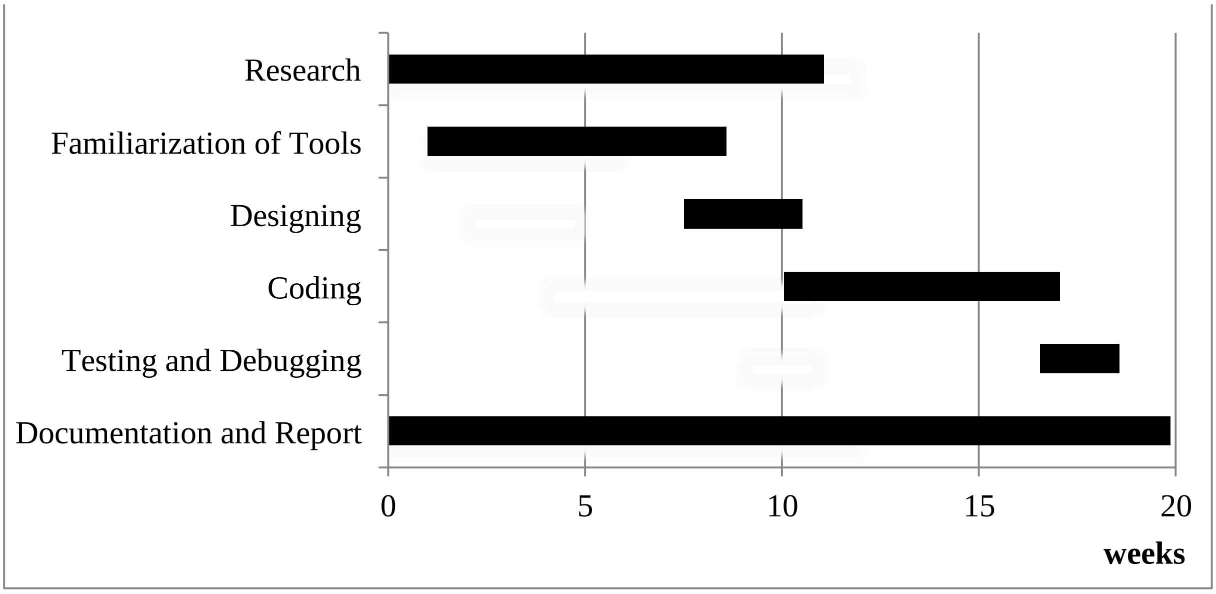
Pip install Django-apscheduler: to run the scheduling task in the django

Pip install Django-extensions: to install django kernel and other extensions

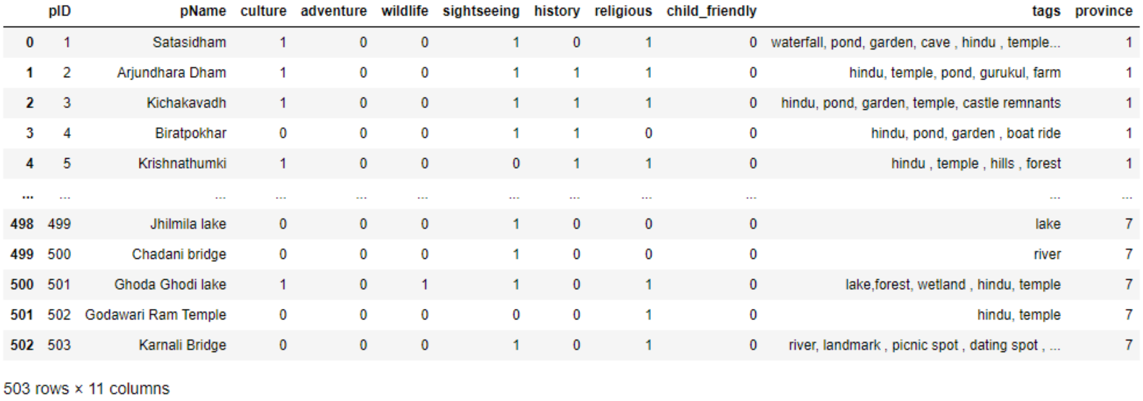
Appendix B. Project Schedule:

Our project schedule is explained by the gantt chart shown below:

Table 10-1: Gantt Chart



Appendix C. Original Collected Dataset:

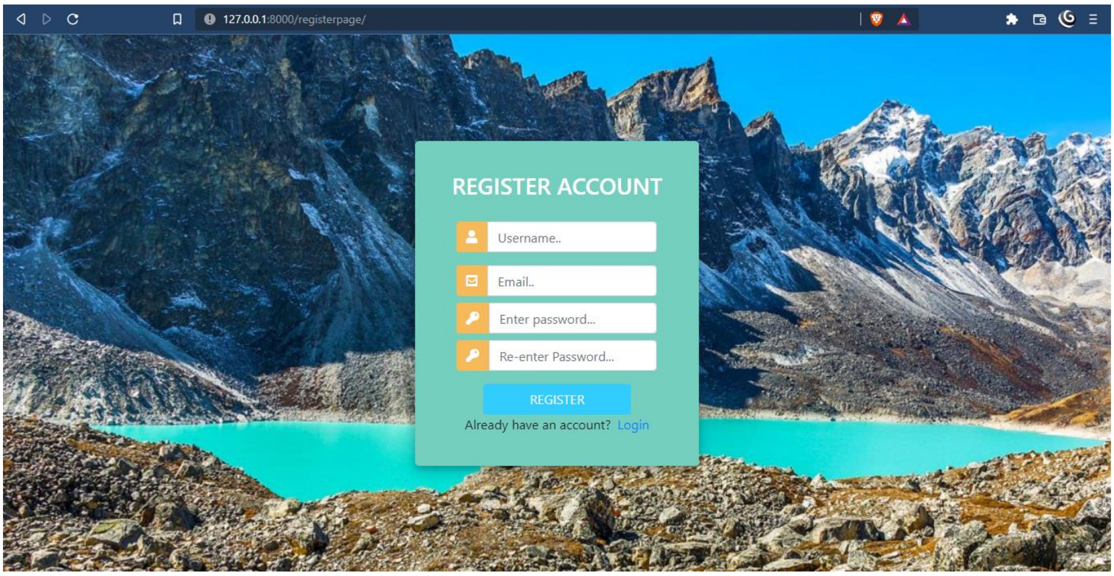


	pID	pName	culture	adventure	wildlife	sightseeing	history	religious	child_friendly	tags	province
0	1	Satasidham	1	0	0	1	0	1	0	waterfall, pond, garden, cave , hindu , temple...	1
1	2	Arjundhara Dham	1	0	0	1	1	1	0	hindu, temple, pond, gurukul, farm	1
2	3	Kichakavadh	1	0	0	1	1	1	0	hindu, pond, garden, temple, castle remnants	1
3	4	Biratpokhar	0	0	0	1	1	0	0	hindu, pond, garden , boat ride	1
4	5	Krishnathumki	1	0	0	0	1	1	0	hindu , temple , hills , forest	1
...	...	...	...	...	...	...	...	...	...	...	...
498	499	Jhilmila lake	0	0	0	1	0	0	0	lake	7
499	500	Chadani bridge	0	0	0	1	0	0	0	river	7
500	501	Ghoda Ghodi lake	1	0	1	1	0	1	0	lake,forest, wetland , hindu, temple	7
501	502	Godawari Ram Temple	0	0	0	0	0	1	0	hindu, temple	7
502	503	Karnali Bridge	0	0	0	1	0	1	0	river, landmark , picnic spot , dating spot , ...	7

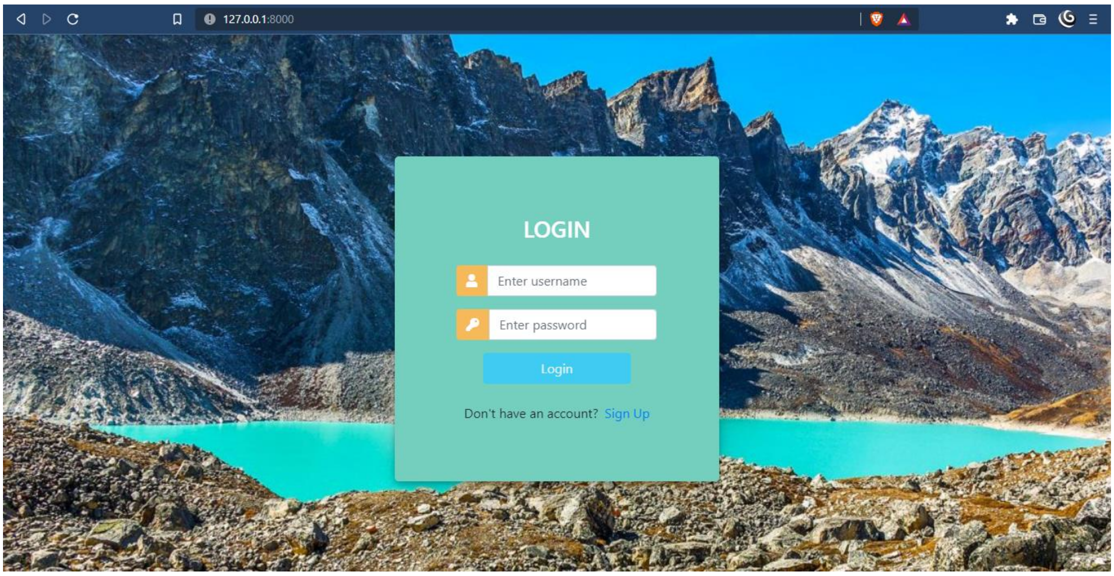
503 rows × 11 columns

Appendix D. Some Screenshots of our Website:

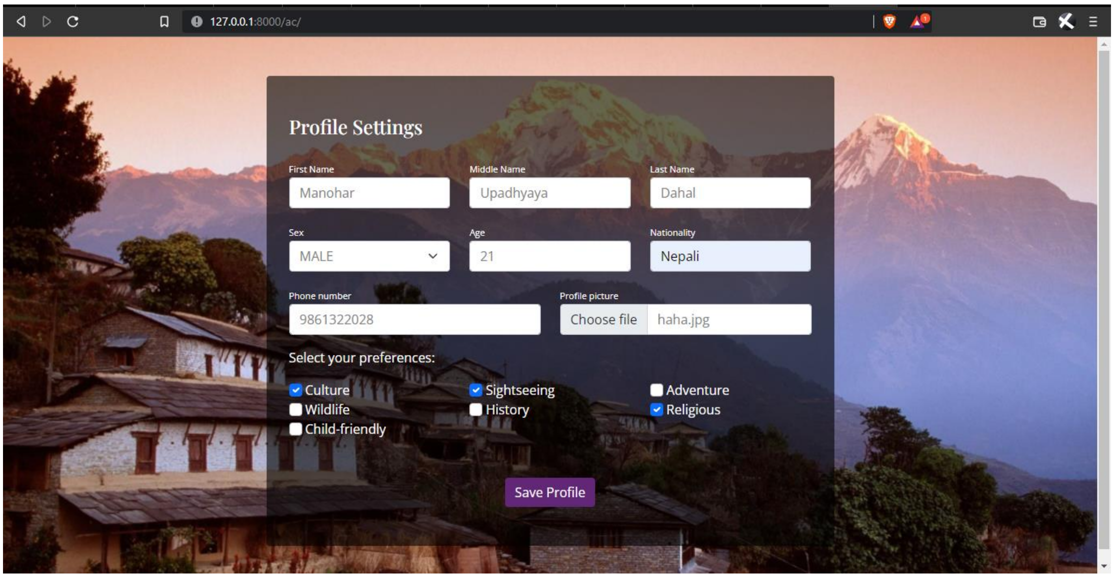
Signup page:



Login page:



Profile settings page:



Profile Settings

First Name: Manohar
Middle Name: Upadhyaya
Last Name: Dahal

Sex: MALE
Age: 21
Nationality: Nepali

Phone number: 9861322028
Profile picture: Choose file haha.jpg

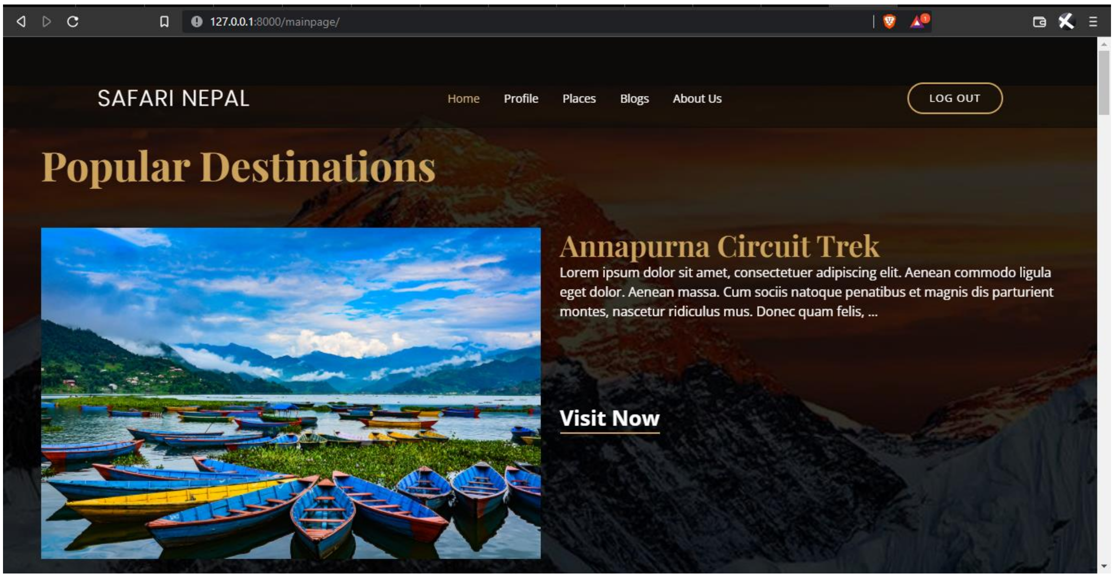
Select your preferences:

☑ Culture
□ Wildlife
□ Child-friendly

☑ Sightseeing
□ History

□ Adventure
☑ Religious

Homepage:

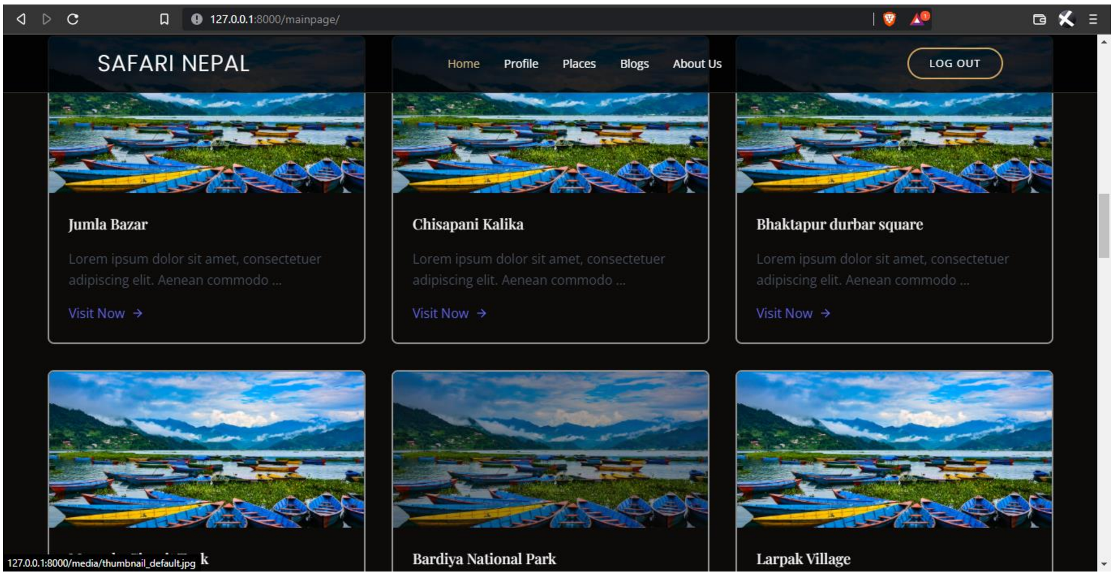


Popular Destinations

Annapurna Circuit Trek

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ...

Recommendations:



Jumla Bazar

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ...

Chisapani Kalika

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ...

Bhaktapur durbar square

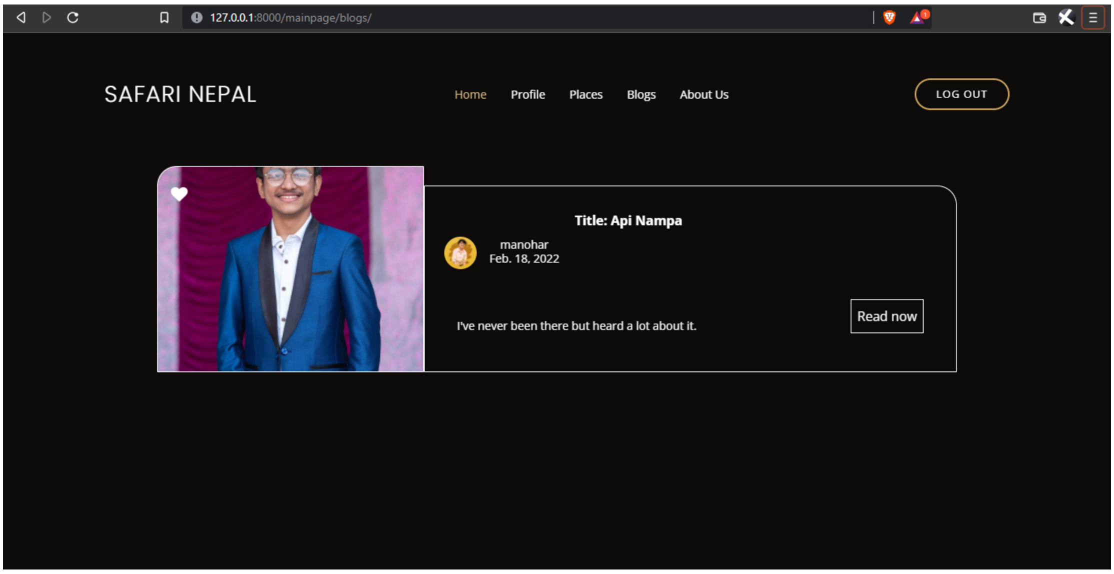
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ...

k

Bardiya National Park

Larpak Village

Blogs:

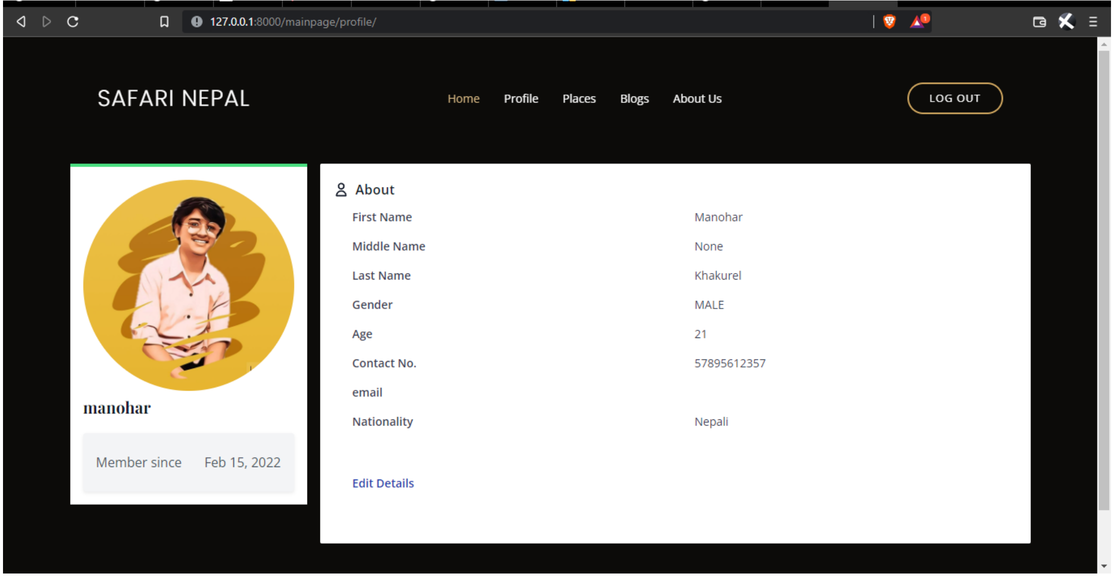


Title: Api Nampa

manohar
Feb. 18, 2022

I've never been there but heard a lot about it.

Profilepage:



manohar

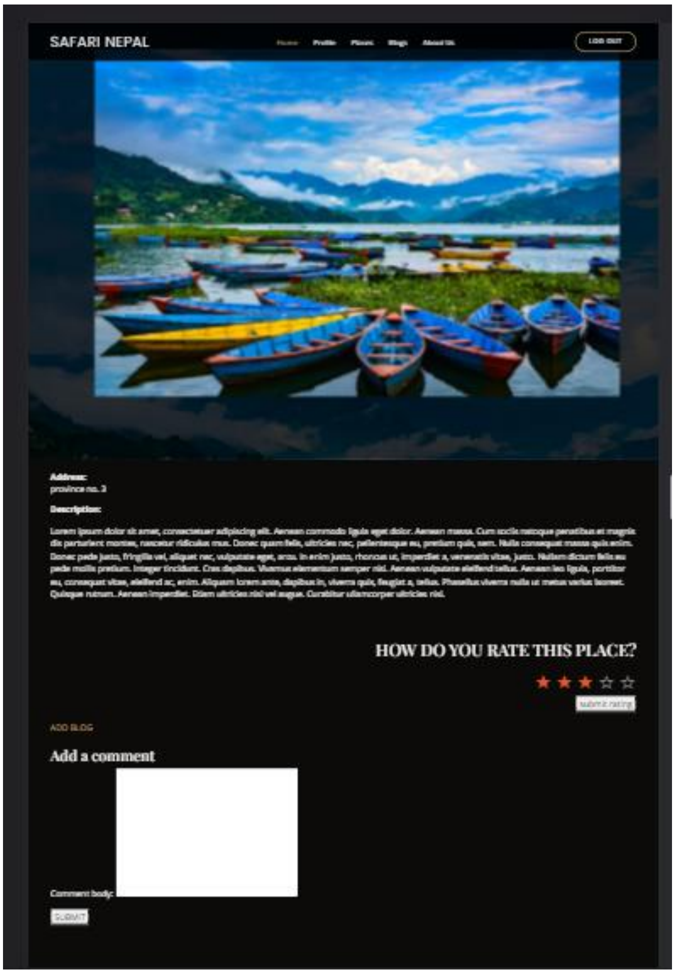
Member since Feb 15, 2022

About

First Name	Manohar
Middle Name	None
Last Name	Khakurel
Gender	MALE
Age	21
Contact No.	57895612357
email	
Nationality	Nepali

Edit Details

Place detailview page:



Address:
province no. 3

Description:

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium. Integer tincidunt. Cras dapibus. Vivamus elementum semper nisi. Aenean vulputate eleifend tellus. Aenean leo ligula, porttitor eu, consequat vitae, eleifend ac, enim. Aliquam lorem ante, dapibus in, viverra quis, feugiat a, tellus. Phasellus viverra nulla ut metus varius laoreet. Quisque rutrum. Aenean imperdiet. Etiam ultricies nisi vel augue. Curabitur ullamcorper ultricies nisi.

HOW DO YOU RATE THIS PLACE?

★★★☆☆

submit rating

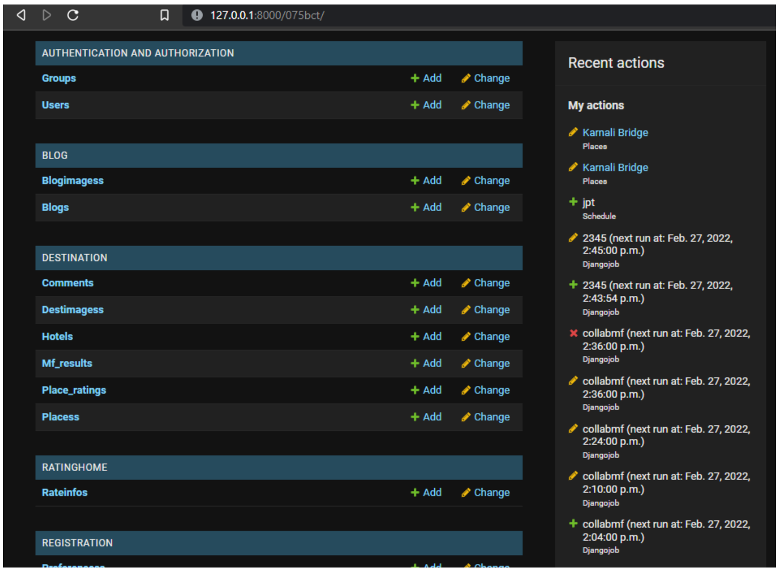
ADD BLOG

Add a comment

Comment body:

SUBMIT

Admin panel:



127.0.0.1:8000/075bct/

AUTHENTICATION AND AUTHORIZATION		
Groups	+ Add	✎ Change
Users	+ Add	✎ Change

BLOG		
Blogimagess	+ Add	✎ Change
Blogs	+ Add	✎ Change

DESTINATION		
Comments	+ Add	✎ Change
Destimagess	+ Add	✎ Change
Hotels	+ Add	✎ Change
Mf_results	+ Add	✎ Change
Place_ratings	+ Add	✎ Change
Placess	+ Add	✎ Change

RATINGHOME		
Rateinfos	+ Add	✎ Change

REGISTRATION		
Preferencess	+ Add	✎ Change

Recent actions

My actions

- ✎ Karnali Bridge
  Places
- ✎ Karnali Bridge
  Places
- + jpt
  Schedule
- ✎ 2345 (next run at: Feb. 27, 2022, 2:45:00 p.m.)
  Djangojob
- + 2345 (next run at: Feb. 27, 2022, 2:43:54 p.m.)
  Djangojob
- ✗ collabmf (next run at: Feb. 27, 2022, 2:36:00 p.m.)
  Djangojob
- ✎ collabmf (next run at: Feb. 27, 2022, 2:36:00 p.m.)
  Djangojob
- ✎ collabmf (next run at: Feb. 27, 2022, 2:24:00 p.m.)
  Djangojob
- ✎ collabmf (next run at: Feb. 27, 2022, 2:10:00 p.m.)
  Djangojob
- + collabmf (next run at: Feb. 27, 2022, 2:04:00 p.m.)
  Djangojob

References

[1] I. Pilaszy, B. Nemmeth, D. Tikk, G. Takács, Matrix Factorization and Neighbor Based Algorithms for the netflix prize problem, ACM, 2009.

[2] A. T. a. M. Jahrer, "The BigChaos Solution to the Netflix Grand Prize," 2009.

[3] R. H. Singh, S. Maurya, T. Tripathi, T. Narula, Gaurav Srivastav, "Movie Recommendation System using Cosine Similarity and Knn," *International Journal of Engineering and Advanced Technology,* vol. 9, no. 5, 2020.

[4] V Bijalwan, P. Kumari, J. Pascual, "KNN based Machine Learning Approach for Text and Document Mining," *International Journal of Database Theory and Application,* vol. 7, pp. 61-70, 2014.

[5] Ö. F. Aktulum, "MATRIX FACTORIZATION WITH STOCHASTIC GRADIENT DESCENT FOR RECOMMENDER SYSTEMS," 2019.

[6] Z. L. B. Wang, "Tourism recommendation system based on data mining," *Journal of Physics: Conference Series,* 2019.

[7] S. R. V. P. Vinodhini G, "A State Of The Art Survey On Cold Start Problem In A Collaborative Filtering System," *International Journal of Scientific and Technology Research,* vol. 9, pp. 2606-2612, 2020.

[8] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence,* vol. 2009, pp. 1-19, 2009.