

UNIVERSITY PARTNER



Project and Professionalism (6CS020)

Final year report
Allbachelor-shop

Student ID: : 1928930
Student Name: : Bishal Khadka
Group: : C3G1
Supervisor : Mr. Prakash Gautam
Cohort : 2/3/4/5
Submitted on : 09/09/2020
Word count : 10605

Declaration Sheet

Award Title: *BSc (Hons) Computer Science*

Declaration Sheet

(Presented in partial fulfillment of the assessment requirements for the above award.)

This work or any part thereof has not previously been presented in any form to the University or to any other institutional body whether for assessment or for other purposes. Save for any express acknowledgements, references and/or bibliographies cited in the work. I confirm that the intellectual content of the work is the result of my own efforts and of no other person.

It is acknowledged that the author of any project work shall own the copyright. However, by submitting such copyright work for assessment, the author grants to the University a perpetual royalty-free license to do all or any of those things referred to in section 16(l) of the Copyright Designs and Patents Act 1988. (viz. to copy work; to issue copies to the public; to perform or show or play the work in public; to broadcast the work or to make an adaptation of the work).

Student Name: Bishal Khadka

Student ID Number: 1928930

Signature:



Date: 09/09/2020

ABSTRACT

In this report/project I try to make recommendation system which is integrated to ecommerce site made up of Django. Recommendation system is one of the major sources for generating revenue in different online companies. Recommendation not only increases the sales of product but it helps to make the system personalized and helps to find out the preferences and interest of particular user to the product. It is currently being used in big tech companies for recommending products, songs, videos, movies, jobs, services etc. It is also implemented in search engine, ranking, friend suggestion, gaming, Point of interest, online advertisements etc. It is one of the most studied and researched topics in the field of artificial intelligence. Researcher and scientist now studying and developing recommendation using deep neural network.

ACKNOWLEDGEMENT

I would first like to sincerely convey my gratitude to my final year project supervisor Mr. Prakash Gautam of Herald college Kathmandu for replying my every e-mail, giving wise advice and reviews, clarify whenever I needed , scheduling meeting and showing me the right path for my steps throughout the development of this project and report writing. I will never forget what he has done to me and I am gratefully indebted to his involvement and contribution. This project will not be completed without his kind cooperation.

I would also like to thank Mr. Biraj Dulal as a reader for this report for his valuable comments. I would also like to acknowledge Mr. Padam Ghimire for introducing me to the topic of recommendation system.

I would like to thank all the contributors who have been enthusiastically participated and decided to share information and knowledge during my survey.

Lastly, I must be grateful to my parents and friends for providing me with trustworthy support and nonstop inspiration and praise. This accomplishment would not have been possible without them.

Thank you.

Author

Bishal Khadka

LIST OF SYMBOLS / ABBREVIATIONS

RS	Recommendation system
CF	Collaborative filtering
CBF	Content Based filtering
RMSE	Root Mean Square Error
MAE	Mean absolute error
MF	Matrix factorization
MBMF	Magnitude bounded Matrix factorization
NMF	Non-negative Matrix Factorization algorithms
AN	Adversarial Networks
SVD	Singular Value Decomposition
B-SVD	Bounded Singular Value Decomposition
AM	Attentional Models
RBM	Restricted Boltzmann Machine
CNN	Convolutional neural network
RNN	Recurrent neural network
DNN	Deep neural network
MLP	Multilayer Perceptron
AE	Autoencoder
NADE	Neural Autoregressive Distribution Estimation
BMC-ADMM	Bounded Matrix Completion in Alternating Direction of Multiplier Method

Table of Contents

Chapter 1. Introduction:	1
1.1 Motivation:	2
1.2 Academic question:	3
1.3 Aims:	3
1.4 Objective:	3
1.5 Artifact:	4
1.5.1 Artifact 1-Web Application:	4
1.5.2 Artifact 2-Product Recommendation:	4
1.6 Scope and limitation of project:	4
1.7 Report Outline:	5
Chapter 2 Literature Review:	6
2.1 Types of recommendation system:	7
2.1.1 Content-based (CB) filtering:	8
2.1.1.1 Term Frequency – Inverse document frequency:	8
2.1.1.2 Cosine similarity matrix:	8
2.2 Collaborative filtering (CF):	9
2.2.1. Neighborhood-based approach:	9
2.2.1.1. Pearson Correlation:	9
2.2.1.2. Constrained Pearson correlation:	10
2.2.1.3. Cosine distance:	10
2.2.1.3. Adjusted Cosine Similarity:	11
2.2.1.4. Mean squared distance:	11
2.2.1.5. Euclidean Distance:	12
2.2.1.6. Tanimoto Coefficient:	12
2.2.1.7. Spearman Correlation:	12
2.2.2 The clustering based Collaborative filtering:	12
2.2.3 latent factor models:	13
Chapter 3: Requirement gathering and analysis:	18
3.1 Sources of data:	18
3.2 fact finding techniques:	18
3.2.1 Research:	18
3.2.2 Observation:	19

3.3 functional Decomposition diagram:	19
3.4 Functional requirement:	20
3.4.1. UI/ UX requirement:.....	20
3.4.2. personalized recommendation system:	20
3.4.3 Management requirements:	20
3.5 Non-functional requirement:	20
3.5.1. Speed:.....	20
3.5.2. Security and privacy:	21
3.5.3. Extensibility:.....	21
3.6 Usability requirement:	21
3.6.1 Ease of use:.....	21
3.6.2 Satisfaction and memorability:.....	21
3.6.3 Graphics:.....	21
Chapter 4: design and architecture:	22
4.1. Process Design:	22
4.1.1 Use case diagram:	22
4.1.2 Class diagram:.....	23
4.1.3 Sequence diagram:.....	24
4.1.4 State Transition diagram:	25
4.1.5 Activity diagram:	26
4.1.5.1 Activity diagram for user:	26
4.1.5.2 Activity diagram for admin:	27
4.2. Database design:.....	28
4.2.1 Entity relation diagram:.....	28
4.2.1 Data dictionary diagram:	29
4.3. User interface:	32
4.3.1 Wireframe:	32
4.4. System architecture:	37
4.4.1 The MVC pattern:.....	37
Chapter 5 implementation and development process:.....	39
5.1.1 Technical specification:	39
5.1.2 Development:	39
Chapter 6: walkthrough of system:	44

6.1.1 Admin panel:	44
6.1.2user view:	50
Chapter 7 Testing and Evaluation:	58
7.1.1 Invalid Credential:	58
7.1.2. mobile friendly testing:	64
7.1.4. Testing for Recommendation system:	69
7.2 Academic question:	72
Chapter 8 conclusion and future work:	73
8.1 Critical evaluation:	73
References	75

Table of figures

Figure 1 ecommerce revenue growth worldwide	1
Figure 2 ecommerce revenue growth in Nepal	1
Figure 3 top 5 countries with highest sell in e-commerce	2
Figure 4 types of recommender system	7
Figure 5 functional decomposition diagram.....	19
Figure 6 Use case diagram.....	22
Figure 7 Class diagram	23
Figure 8 Sequence diagram	24
Figure 9 State transition diagram	25
Figure 10 activity diagram for user	26
Figure 11 activity diagram for admin	27
Figure 12 Entity relation diagram	28
Figure 13 wireframe of index page.....	32
Figure 14 Product detail page	33
Figure 15 profile page	34
Figure 16 My cart page	35
Figure 17 order details	36
Figure 18 MTV architecture Django	37
Figure 19 system architecture for recommendation system	38
Figure 20 Euclidean score and Pearson score	42
Figure 21 admin login.....	44
Figure 22 Admin Dashboard.....	45
Figure 23 Add category.....	45
Figure 24 add sliders.....	45
Figure 25 Add sub category	Error! Bookmark not defined.
Figure 26 Add products.....	46
Figure 27 view users	47
Figure 28 Viewing similar cluster.....	47
Figure 29 viewing profile information of user	48
Figure 30 viewing order detail	48
Figure 31 view orders.....	49
Figure 32 view shopcart of user	49
Figure 33 Update order product information	49
Figure 34 user sign up.....	50
Figure 35 user sign in.....	50
Figure 36 Home page.....	51
Figure 37 Detail homepage.....	51
Figure 38 Search list.....	52
Figure 39 View category wise product.....	52
Figure 40 Product detail page 1	53
Figure 41 product detail page 2.....	53
Figure 42 User profile.....	54

Figure 43 Edit profile	54
Figure 44 change password	55
Figure 45 My cart.....	56
Figure 46 My orders	56
Figure 47 fill order details.....	57
Figure 48 order details	57
Figure 49 password does not match.....	58
Figure 50 user already exist	59
Figure 51 invalid username or password.....	60
Figure 52 invalid password 1	60
Figure 53 Invalid password 2.....	61
Figure 54 invalid password 3.....	62
Figure 55 Invalid password 4.....	63
Figure 56 Quantity limit for add to cart.....	63
Figure 57 mobile friendly testing index page.....	64
Figure 58 product detail page.....	65
Figure 59 mobile testing for login page.....	66
Figure 60 profile page	67
Figure 61 my cart.....	68
Figure 62 rating for user.....	69
Figure 63 rating for similar user.....	70
Figure 64 not showing rating to previously rated product	70
Figure 65 recommendation for test1	71
Figure 66 admin viewing clusters	72

List of tables

Table 1 data dictionary diagram	31
Table 2 comparison table of model.....	43

Chapter 1. Introduction:

Nowadays, Companies and individual are increasingly trying to buy and sell variety of products online. The most of the transaction are conducted with the help of internet. Not only products, but buying and selling of services are quite popular nowadays. According to Statista, the revenue for ecommerce market was US\$1,800,517 globally in 2019 and expected to reach US\$2,553,572 million in 2021.

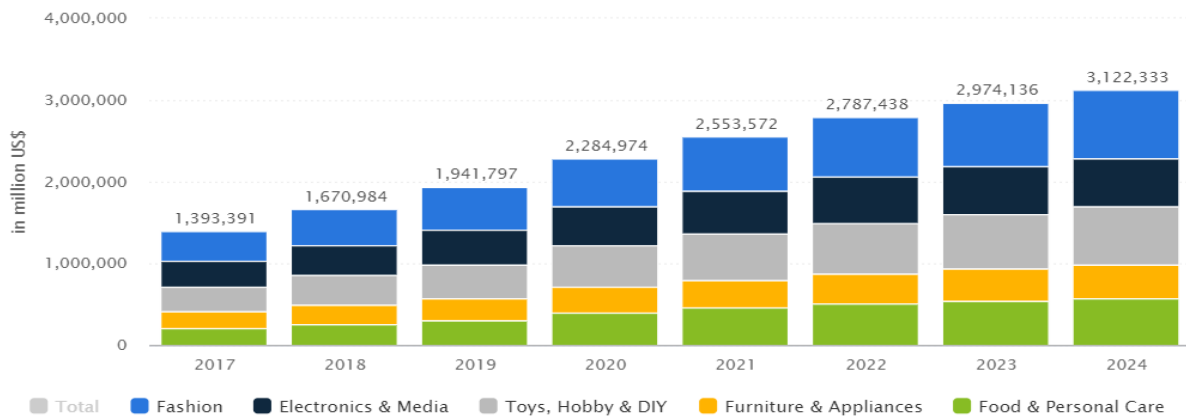


Figure 1 ecommerce revenue growth worldwide

Although the digital market like ecommerce companies are not more here in Nepal. Ecommerce development in Nepal is also in growing order which can be seen in figure below.

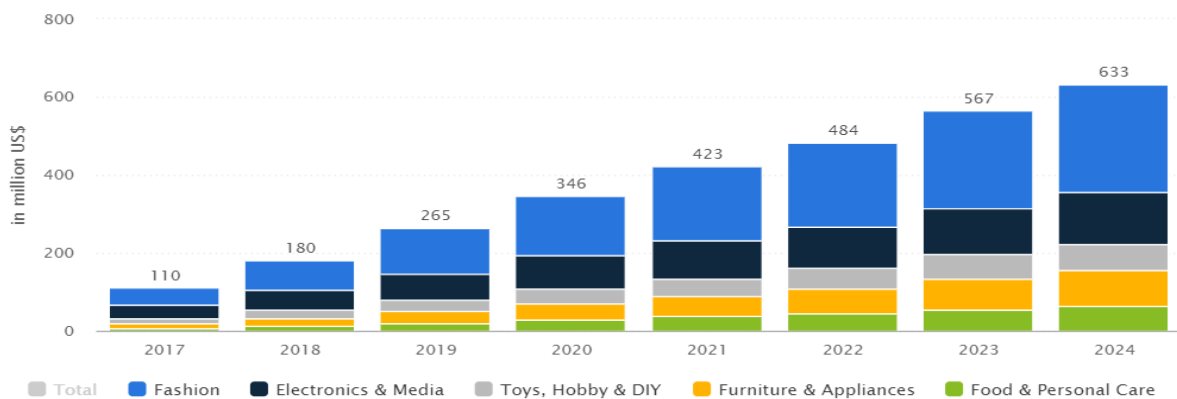


Figure 2 ecommerce revenue growth in Nepal

These figures are projected to increase due to increase in number of mobile internet users which can also be seen in figure above.

The ecommerce market is blooming in advanced and developed country of world. The development of online commerce can also be seen in figure below. (statista, 2020)

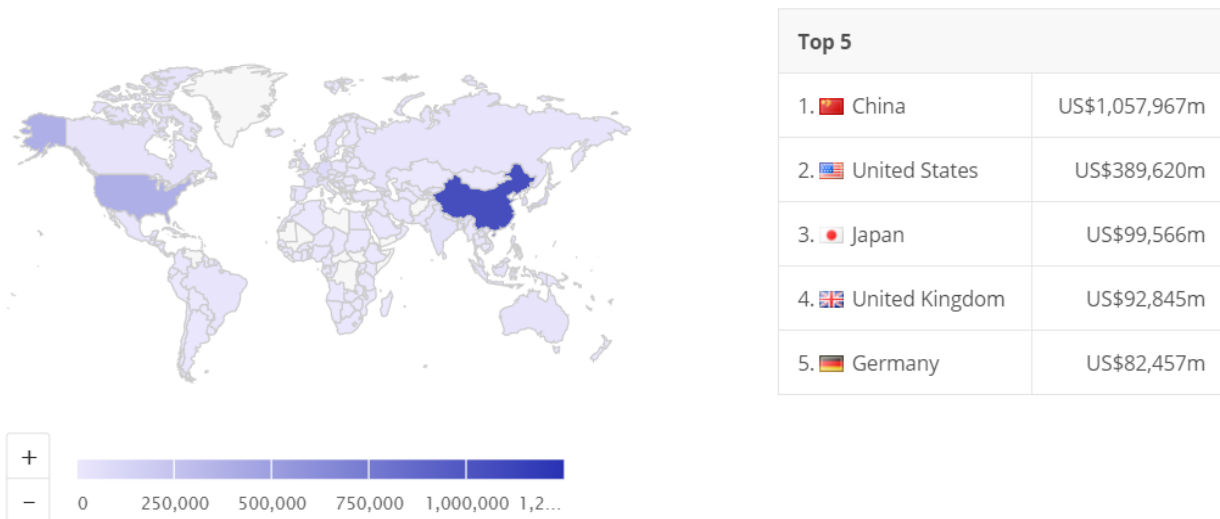


Figure 3 top 5 countries with highest sell in e-commerce

1.1 Motivation:

The idea of building this system is to ease both buyer and seller by recommending products to individual user. Our country Nepal stills lack certain types of ecommerce. There are very limited ecommerce sites in Nepal where user get their desired product. Nearly all of Nepal's ecommerce platform fails to produce satisfactory revenue from the product till this date. Because of this reason I have built an ecommerce framework for purchasing and selling of numerous products in different categories like books, electronics, fashion, cosmetics, baby products and many more.

The one reason behind the stagnation of growth in the present ecommerce site is due to lack of knowledge about the individual interest of the user and the customer's on-site behavior. Sales are not boosting as expected here in Nepal because it is one of the tough tasks to find and target the right audience in ecommerce. To address this type of problem, I researched the topic of recommendation system to generate the personalized recommendation system for each user who uses our system. This build system uses automated recommendation system which solves the mentioned problem. The developed

recommendation system helps the new and existing user to discover relevant and related product recommended from our system based on browsing history, user's behaviors, ratings, demographics and purchase history. The developed system is Business to customer type of ecommerce where customer may order, buy, rate and review different products online posted by the admin of this site. The web application is very easy to use because of simple and comfortable user-centric interface.

1.2 Academic question:

How will a recommendation system that uses machine learning technique be used to predict the interest/desires of users, boost their experience to increase the sale of products, earn revenue in ecommerce sector and what are the possible challenges and issues in real word?

1.3 Aims:

The main aim of this developed project are to:

- design, develop and test the full featured ecommerce web app for viewing, buying, rating and reviewing different product
- build recommendation system to recommend different product to individual user.
- examine, identify and evaluate the past publication or existing system of the related topic.

1.4 Objective:

The main objectives of the study are as follows:

- 1) To study about the B2C type of ecommerce in Nepal and recommendation types that can actually be used in ecommerce site.
- 2) To build the powerful recommendation system that can recommend each and every individual user of the system.
- 3) To provide the online platform to view, buy rate and review different product.
- 4) To implement the MVC (Model, View, controlled) pattern in web-based application.

1.5 Artifact:

To develop all the artifact given below, Django framework with MySQL database is used. The developed system is ecommerce which can recommend products to similar users of the system.

1.5.1 Artifact 1-Web Application:

The developed system has scalable website design with different features like registration, authentication, search products, product details, categories, sub categories, search bar, user generated review, ratings etc. All the front-end is displayed from the database where only admin can upload all the content. This site has also review system where user can give their view about the specific products and can also give rating from 1 to 5. Admin can add the item to display to other users. The system is also made mobile user-friendly and Search engine optimization for the better performance.

1.5.2 Artifact 2-Product Recommendation:

This site recommends relevant product to users individually based upon the user's behaviors, ratings and history of purchase. Various techniques are used and tested to develop the model. Python is used to develop the AI model for this system.

1.6 Scope and limitation of project:

As the main aim of this project is to develop and test the ecommerce system and to integrate recommendation system into web-based application. The scope of the study may include product bought on various part of Nepal but can be used worldwide. The study is not specific to one country or the region. Today's users are demanding very rich system to purchase product online. There is some limitation of the system: I may not work well with newly signed in users due to the fact that the recommendations emerge from a correlation between the intended user and other used focused on collection of ratings. So, it is almost impossible to judge the user and to categorize a user with very few ratings. It only starts recommending things after user rate some of the products from the system. In similar fashion, new item if not rated by any user, it gets recommended at the last part of the recommendation. Since user only see upper part of the recommendation. The

product rated very low also does not appear at the top part of the recommendation. It is also known as “early rater” problem and cold start program. To deal with this problem, homepage of this site does not show recommended things but only shows the latest product from the database and recommended things for the user can only be seen from recommendation webpage. It takes much more time to recommend products to fresh user. The reason behind choosing this project is to show how recommendation can be build, designed and analyzed. The recommendation is that type of system which can be extended in future and the algorithm used during the development of this project can be improved throughout the period of maintenance of this project. Although the system works perfect in larger dataset.

1.7 Report Outline:

The rest of this report is prearranged as: Chapter 2 of this report presents relevant theoretical literature review of papers, journal and books related to recommendation system approaches, technique and algorithm. Chapter 3 covers the requirements gathering and analysis part of the system. Chapter 4 focuses in design and architecture implemented in project. Chapter 5 deals with implementation part, highlighting key aspects and development process. Chapter 6 takes the reader walkthrough of practical implementation of system. Chapter 7 provides Testing and Evaluation of the system. Chapter 8 concludes the report and final section describes future work in related topics.

Chapter 2 Literature Review:

My topic of research is not the new topic but active area of research. Recommendation system (RS) is popular field of study among the researcher and scientist since last few decades. RS to recommend products, books and news is fairly common in large tech companies including amazon and Google. Social media site like Facebook, Tinder and twitter also suggest new friends and advertisements. Not only that but last FM, pandora etc. also offers songs that the user may like. Films and online videos are likewise generating meaningful recommendation by companies such as Netflix, Movie Lens, IMDb, YouTube etc. (aggrawal, et al., 2017)

RS comes into practice in order to address, deeper analyze and resolve the issue and problem of both buyers and sellers by automating the activities based on the quality of user-item communication. (Melville & Sindwani, 2018)

Recommendation engines is totally transforming or reshaping ecommerce industry by helping the customer to choose the relevant product. Recommendation system allows to increase the figures of sells, to offer more varied products, to improve user satisfaction and to fully consider customer needs. In simplest term recommendation engines are the ranked list of products to the user based upon different factors like user's preference and constraints. To first determine the constraints and preference, RS collects user's view with the help of ratings to the particular product in most of the cases. (Ricci, et al., 2011)

RS can also be called as information retrieval as it is used to filter the system and display only item's that match the user's interest. For this, user must give some of interest to the system and system tries to display products user may like. If user does not give any information to the system, the system cannot provide any recommendation. Various tools are available to recommend the product. Most used recommender system in existing online environment are collaborative filtering, content-based filtering, Popularity based, Hybrid approach. (Anantha & Bhattula, 2017)

2.1 Types of recommendation system:

Before making recommendation system, one must know the pros and cons of each type of recommender system which are described in this section. Broadly speaking there are two approaches of recommendation system. They are content based filtering, collaborative filtering and hybrid filtering. Each of this type has their own limitation and benefits which is shown in later part of this section. Collaborative filtering is further divided into two type: Model-based filtering techniques and memory-based filtering techniques. Model-based filtering techniques is further divided into clustering, association, Bayesian and neural networks. Memory based filtering technique is further divided into user-user based collaborative filtering and item-item based collaborative filtering. Hybrid filtering is combination of both collaborative and content-based filtering.

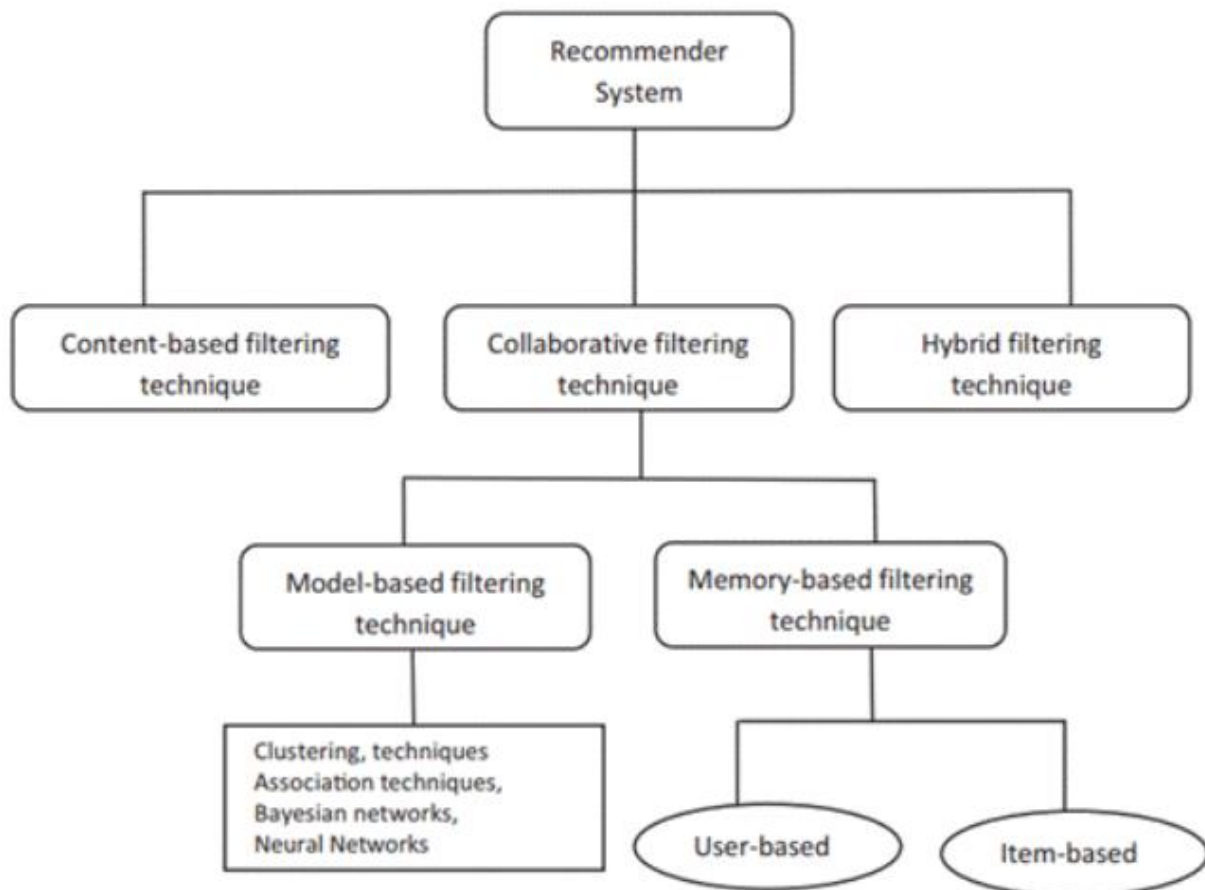


Figure 4 types of recommender system

2.1.1 Content-based (CB) filtering:

Content-based (CB) filtering, also known as cognitive filtering recommends similar products that the user had already bought from the system. Content-based filtering can also recommend product to user with other factors like age, gender, geography, review, usage pattern etc. The preferences of particular users can be determined by retrieval method such as cosine similarity matrix, term frequency-inverse document frequency, Long Distance Affair etc. or by using machine learning techniques like support vector machines, Naïve Bayes, decision trees etc. (Pazzani & Billsus, 2015)

2.1.1.1 Term Frequency – Inverse document frequency:

TF-IDF techniques is mostly used in knowledge extraction and processing of data. Term frequency as the name indicates, measures the frequency at which the word occurs which can be calculated as the ratio of total number of particular words occurs in a document and total number of words in the document. (Beel, et al., 2017) Inverse document frequency measures the significance of term in a document and calculated as:

IDF (particular word) = \log_e (total no. of documents / no of documents with particular word in it)

This can be calculated as:

$$tfidfweight = \sum_{i \in d} tf_{i,d} * \log \left(\frac{N}{df_i} \right)$$

This technique can be used and effective only in the field where there is more textual data.

2.1.1.2 Cosine similarity matrix:

This approach offers the similarity estimation between two items by angle calculation of two vectors. Cosine similarity can be calculated as:

$$similarity = \cos(\theta) = \frac{A \cdot B}{|A||B|}$$

It is the decision which is based on orientation and not on magnitude.

(Sitikhu, et al., 2019)

2.2 Collaborative filtering (CF):

The developers at xerox first coined and developed the term collaborative filtering. Since then, has been enhanced and developed by using different technology and algorithms. The main goal of recommendation system is to suggest the content based on individuals' desires. Collaborative filtering (CF) suggests the item which are preferred by other related users. There are various types of algorithms that can recommend the item and every algorithm has different set of features. Different algorithm may be great or bad for various datasets depending upon the features. The algorithm can take more time to function better in particular datasets. (HERLOCKER, et al., 2004) Some of the algorithm are described briefly:

2.2.1. Neighborhood-based approach:

Neighborhood-based algorithm can be used to predict the user's desire and opinion. It also analyzes the existing users from the system. In this method, the group of similar users is segregate and their average rating is computed from all users that are close to the actual logged in user. In simple form collaborative filtering is based upon the assumption that similar customers like similar products. So, the main task is to find the similar customer from the set of all the customer available in the database. The one way to find similar customer is from the ratings and reviews given by the customer to different products. Another way is to track all the user's activity in the web application. For the recommendation part first of all, similarity between separated user and active user is calculated by using different techniques which are discussed below:

2.2.1.1. Pearson Correlation:

This similarity was introduced in 1994 by Group Lens and is considered as the best-known algorithm to find the similarity between the users.

Let the profile of the targeted user be x (represented as a vector) and the profile of another user be y ($x, y \in T$). Here the score of similarity between the target user x and the neighbor y can be calculated by using the following formula.

$$R_{x,y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 (y_i - \bar{y})^2}}$$

Where,

x_i is the rating given by user to item i ,

n is the total number of items and

\bar{x} is the mean rating given by a user and calculated by using the following formula:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

The value of $R_{x,y}$ comes between -1 and +1. If a value is equal to -1, there is a negative correlation between two values and if the value is equal to or near to 1 there is positive correlation. Here x and y are the user whose similarity needs to be computed. (Katarya & verma, 2016)

2.2.1.2. Constrained Pearson correlation:

(Shardanand & Maes, 1995) proposed constrained Pearson correlation to develop the song's recommendation systems named RINGO. They developed this algorithm from Pearson correlation approach replacing the average rating with average positive and negative level from 1(dislike) to 7(like) and neutral (neither preferred nor hated). The Constrained Pearson can be calculated as:

$$s(u, v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_z)(r_{v,i} - r_z)}{\sqrt{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_z)^2} \sqrt{\sum_{i \in I_u \cap I_v} (r_{v,i} - r_z)^2}}$$

Where u and v are two users whose similarity needs to be computed

$R_{u,i}$ is the rating given by user u to item i and $R_{v,i}$ is the rating given by user v to item i .

2.2.1.3. Cosine distance:

The user similarity can also be calculated by taking ratings of users and putting two user's rating as vector in an m -dimensional space and compute the cosine of angle between two users as shown in equation below:

$$w_{a,u} = \cos(\vec{r}_a, \vec{r}_u) = \frac{\vec{r}_a \cdot \vec{r}_u}{\|\vec{r}_a\|_2 \times \|\vec{r}_u\|_2} = \frac{\sum_{i=1}^m r_{a,i} r_{u,i}}{\sqrt{\sum_{i=1}^m r_{a,i}^2} \sqrt{\sum_{i=1}^m r_{u,i}^2}}$$

where $w(a,u)$ represents the measure of similarity between user a and the active user u .

I is the set of item rated by user a and u

R_{ui} is the rating given by user u to item i and r is the mean rating given by user i

+1 (same direction) and -1 (opposite direction) represents high correlation and 0 (irrelevant) represents no correlation. This approach gives both negative ranking and blank rated product as zero rating. (Liu, et al., 2014)

2.2.1.3. Adjusted Cosine Similarity:

Adjusted Cosine similarity is necessary to recommend the user based upon the rating of products. The only problem was the user rate products differently to different products. The solution to this problem is to subtract average user rating from rating provided by the user in each product. So, the adjusted cosine similarity is calculated by using the following formula:

$$s(u, v) = \frac{\sum_{i \in I_{uv}} (r_{ui} - \bar{r}_i)(r_{vi} - \bar{r}_i)}{\sqrt{\sum_{i \in I_u} (r_{ui} - \bar{r}_i)^2 \sum_{i \in I_v} (r_{vi} - \bar{r}_i)^2}}$$

Where, u and v are the two users u is the user to be predicted and v is the active user of the system

2.2.1.4. Mean squared distance:

Mean squared distance, as name suggests, this approach adds the square difference between the rating of both user u and v alongside their common products which is divided by the total number of common products. This approach underlines the penalizing divergence in taste over resemblance of test. The mean-squared distance can be calculated by using the formula below:

$$s(u, v) = \frac{\sum_{i \in I_u \cap I_v} (r_{u,i} - r_{v,i})^2}{|I_u \cap I_v|}$$

Where u is the active user,

V is the any other user,

s(u, v) - similarity between active user u and user v,

R_{ui} is the rating given by user u to item i to item I,

R_{vi} is the rating given by user v to item i

(Krishnan & Brahma, n.d.)

2.2.1.5. Euclidean Distance:

Euclidean Distance can also be used to find the similarity between two users. The distance between two users can be calculated as:

$$(x, y) = \sqrt{\sum_1^n (x_i - y_i)^2}$$

Where x_i and y_i are the score of products given by two user u and v for same product n . (Sondur & Chigadani, 2016)

2.2.1.6: Tanimoto Coefficient:

This is commonly known as Jaccard similarity coefficient. It does not take the ratings of items explicitly but it only takes if two users show some preference with one another and calculated by using the following formula:

$$sim(i, j) = \frac{|f_i \cap f_j|}{|f_i| + |f_j| - |f_i \cap f_j|}$$

Where $f_i \cap f_j$ is intersection between product i and product j for items where users prefer Tanimoto coefficient, $sim(i, j)$ is 1 if two products are completely overlapped and if two products are not overlapped, $sim(i, j)$ is 0. (Lee, 2015)

2.2.1.7. Spearman Correlation:

This approach is similar to Pearson correlation, the only difference is instead of finding the similarity between users, this system finds the rank of the user. The spearman correlation can be calculated as:

$$s(u, v) = \frac{\sum_{i \in I_u \cap I_v} (k_{u,i} - \bar{k}_u)(k_{v,i} - \bar{k}_v)}{\sqrt{\sum_{i \in I_u \cap I_v} (k_{u,i} - \bar{k}_u)^2 \sum_{i \in I_u \cap I_v} (k_{v,i} - \bar{k}_v)^2}}$$

$k_{u,i}$ is the i 's ranking for the ratings given by use.

(Sivaramakrishnan, et al., 2018)

2.2.2 The clustering based Collaborative filtering:

This type of recommender system uses cluster of similar users portioned or divided into a different set of data as compared to all other existing users in system. In simpler term, clustering can be called as the grouping of similar user based upon their preference.

There are different types of clustering. Centroid-Based clustering, Distributed-Based Clustering, Connectivity-based Clustering, Density-based clustering are some of them. Any of the mentioned distances can be used for the measurement of user similarity distance. The prediction can be calculated by using the following formula:

$$P_{a,i} = A(\bar{r}_{a,l}) + \frac{\sum_k \{w_{a,k} \times (r_{k,i} - A(\bar{r}_{k,l}))\}}{\sum_k |w_{a,k}|}$$

Where $P_{a,i}$ is the prediction of user a to item i. $A(\bar{r}_{a,l})$ is the average rating of user a to product l and $A(\bar{r}_{k,l})$ is the average ratings of user k to item l. (Kim & Yang, 2015)

2.2.3 latent factor models:

Latent factor model is generally called as matrix Factorization models. This model connects user and products in a latent feature space of low dimension $F \in \mathbb{R}^d$ where F is the hidden characteristics and d is the latent feature space dimension. For an example, shoes products can be categorized by their features such as ankle boot, army boots, athletics shoes, ballet shoes, boat shoes, heels etc. A user has hundreds of products could have been bought from 20 category. So, 20 categories are adequate to explain their preferences. The same thing can be applied to products. Singular value decomposition is one of the common techniques for dimension reduction to identify latent factors. The tensor factorization, Bayesian network are other methods for identifying the latent factor. (Jingying, 2017)

One disadvantage of using content-based recommendation is the system shows always similar type of product all the time. The limitations of content-based recommendation systems are overcome by using collaborative filtering. If the content is not available for the product or if the content is difficult to get still gets recommended to users. No matter what the products look, if purchased user gives good rating to the product it gets recommended to other similar user as well. This system suggests product by looking at the other users(neighbors) that have similar pattern of rating. The neighborhood-based recommendation is favored by various researcher and developer because of its simplicity, efficiency, stability and justifiability. (Desrosiers & Karypis, 2017)

Nowadays, many companies are also trying to use deep neural networks like RBM, Auto encoders, KGE, MLP, AN, AM, RNN, CNN, CF_NADE etc. There are a greater number of hidden layers and better techniques for initialization parameters in Deep neural network. A DNN with large number of hidden units can be more powerful and effective in modeling. Deep neural network in recommendation system is one of the fields that is on the way to perfectionist. Different scientists and researcher are studying and implementing deep neural network for recommendation. Most of the recent work on recommendation are based upon the deep learning-based recommendation system.

I have studied some existing papers, books related to recommendation system and techniques, experiments to complete this project. Some of the techniques, application and evaluation from the papers related to recommendation system are listed below.

1. One of the journals purposed by (Sharma & Mahajan, 2017) made recommendation system for GitHub. They used collaborative filtering with Pearson coefficient and cosine similarity to find the similar user of GitHub to recommend code to the user. Another paper (Shao, et al., 2020) recommended GitHub repository for paper of academic by using item-item recommender system. They used Graph convolutional Network for the classification purpose on 11,272 papers and 7,516 repositories initially and increased to 32,029 papers to evaluate the performance in large number of datasets. To recommend better same repository must be stared by multiple user which leads to cold start problem. So, new user repository until and unless stared by other user is not recommended to other user which is the limitation of paper. Despite of this limitations, Graph Convolution Network works much better than other traditional recommender system when there are large number of constrains and features to watch because they can encode both information of node and structure of graph at the same time.

2. (Wang & Qiang, 2017) have presented a probabilistic framework of neighbors-based recommendation system and multi-layer similarity approach. They also evaluated the model produced from proposed framework using RMSE. A map-based recommender with customer centric interface was done by (Park, et al., 2007) using proposed Bayesian based recommendation system.

3. In the paper (Jiang, et al., 2018) used novel matrix factorization algorithm and they called their algorithm as Magnitude bounded Matrix factorization to fix user, product prediction fluctuation preceded by conversion of constrained function to unconstrained task. To solve the unconstrained task, they have added a stochastic gradient descent method. They used improved matrix factorization in three real dataset and their algorithm, MBMF indicated strong stability which means this algorithm can be used in larger set of dataset since this algorithm does not repeat the process to determine the best suitable value of k. Bounded matrix factorization, MBMF, Feature-wise updated Cyclic Coordinate Descent method, BMC-ADMM, Bounded SVD gained more accuracy than low level matrix factorization, Non- negative matrix factorization. They used RMSE, MAE and F1-score to compare the result.

4. A research paper published by Phonexay, et al to estimate the performance of movie recommendation. They compared proposed improved k-clique methods with k-nearest neighbors, the maximal clique methods and the k-clique methods in movie lens dataset. They performed the test in 10 random datasets from the rating of 800 users to the movies. They have taken the user with 200 rated movies, 100 rated movies, 50 rated movies and 20 rated movies in four different parts. To predict the precision in the recommended dataset they used Mean absolute percentage error (MAPE).The table below shows the MAPE values of all the algorithms for the user rated 200 movies, 100 movies, 50 movies and 20 movies:

S.N.	algorithm	MAPE (200 ratings)	MAPE (100 ratings)	MAPE (50 ratings)	MAPE (20 ratings)
1.	KNN	22.08%	34.45%	40.90%	54.30%
2.	Maximal clique	30.30%	39.62%	48.04%	59.63%
3.	k-clique	20.47%	31.52%	38.57%	53.01%
4.	Improved k clique	18.85%	29.80%	36.57%	49.28%

From this experiment compared to than other methods like KNN, Maximal clique, k-clique, it was found that the proposed improved k-clique method achieved more accuracy than others. K-clique approach would also increase the precision of the recommendation system but the only downside is measuring the k-clique approach takes a log time compared to other approach. Right now, KNN is the correct algorithm to use than maximal clique method which is shown in above table. (Phonexay, et al., 2018)

5. A prototype was made to recommend the users based upon the voice input from the user. by (Kang, et al., 2017) on movie Lens dataset. They collected 347 user's datasets with the help of survey. This is the first recommendation that use natural language processing to detect the desire of the user. They try to solve the real-world problem like what the user really wants from the recommender system not the queries that will work or not in a particular system. The main goal behind this research was to find the subject bias and give the clearer idea to other researcher that the Natural language processing can also be the future to recommendation system. Similar concept was also used in (Chnag, et al., n.d.) where movies are recommended based upon the tag-based preference of users. They recommended the movies based upon the formula:

$$pref_{u,t} = \frac{\sum_{m \in M_u} rating_{u,m} * rel_{m,t} + K * avg_u}{\sum_{m \in M_u} rel_{m,t} + K}$$

Where, m is the movie in set of movies M_u ,

rating_{u,m} is the rating given by user u to movie m

and rel_{m,t} is the relevance score of movies tag

6. In the paper (Zhou, et al., 2018) tried to recommend the display of appropriate advertisement to user in ecommerce company by using very large number of data. They presented the average of five times experimented result from dataset of both Amazon and movie lens dataset. They used deep learning for the prediction of Click Through rate. They developed the base model by using embedding layer to alter high dimensional vectors to low dimensional dense representations followed by pooling layer and Concat layer to deal with various user and their various behavior. To be specific, they used two pooling layers, one is sum pooling for the sum operation and another is average pooling

for average operations to embedding vectors. Fully connected layers are used to practice the feature mixing. They used two training techniques: Mini-batch aware regularization and data adaptive activation function. After training with amazon dataset, movie lens dataset and Alibaba dataset. They compared the deep interest network (designed model) model with other existing deep networks like Base Model, Wide and Deep, Deep FM and PNN. The result is better in the developed deep interest network. They used AUC metric to compare the result with another model and optimizer. They later deployed Deep Interest network (DIN) in Alibaba ecommerce company to display the advertisement to the wide variety of user. Similar to this study, (Zhou, et al., 2019) tries in CTR to estimate the probability of user can click on the product.by using Deep Interest Evolution network(DIEN) and compared this model with other model mentioned above and found out that the Deep Interest evolution Network captures sequential interest efficiently and also models the evolving interest process related to particular product that enforces the performance of large CTR prediction in the advertisement system.

Chapter 3: Requirement gathering and analysis:

Analysis of requirement can be defined as to identify system requirements for investigation, documentation all the requirements and analysis of those collected requirement. The major problem behind this project is to build the ecommerce webapp and integrate recommendation system.

In this section we discuss about the fact-finding techniques, function requirement, non-functional requirement and usability requirements.

3.1 Sources of data:

The data for the ecommerce site is collected from different existing national ecommerce websites like Daraz, Sasto deal, esewapasal etc. and foreign sites like flipcart, amazon, ebay, Walmart etc. The data for recommendation system is the data provided by amazon. For making recommendation only, I have also made sample data by myself.

3.2 fact finding techniques:

I have researched on different topic related to ecommerce and recommendation system. There are many fact findings technique, but I focused in only two of them. I did not choose interview and questionnaires as the fact-finding techniques rather I chose observation, research and studying ecommerce companies as fact finding technique. One must observe the similar system to get the insights of competitors fact.

3.2.1 Research:

I think research also plays a vital role to find what the user really wants from the user. Before this study I have not known about this problem domain, I have manually researched and read papers on the main topic needed for the ecommerce site. So, I decided to view, understand and manipulate internal mechanism of existing system similar to my system as much as possible. I have found that there are many topics rather than just purchasing product in ecommerce site like searching, sorting, Wishlist etc. Recommended product is also one of them. I have also researched about the existing web application like flip cart, amazon, Daraz etc. Although these sites are very large site with millions of transactions and billions of activities in a day, basic features of ecommerce are almost similar to small enterprise ecommerce system.

3.2.2 Observation:

Observation has been used to observe app behavior explicitly. I have examined, observed, viewed and displayed numerous web-based application similar to ecommerce system. I have attempted to make the system similar to my research as much as possible to make this system as much as practicable in real world.

3.3 functional Decomposition diagram:

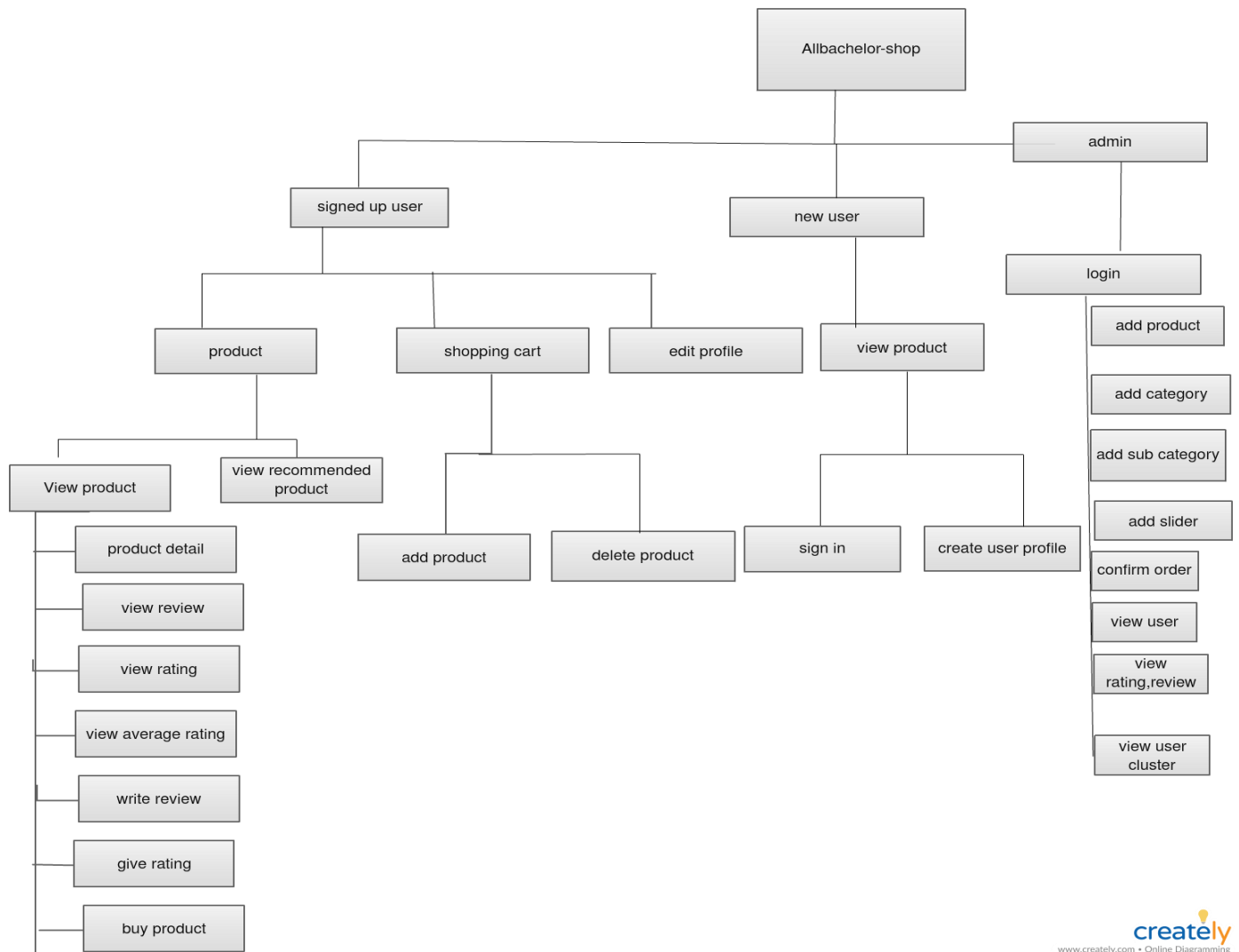


Figure 5 functional decomposition diagram

3.4 Functional requirement:

Functional requirements are the system's technical features or what the system is meant to do. Some of the functional requirements are described below:

3.4.1. UI/ UX requirement:

Attractive design, easy to navigate and speedy page loading are three features an e-commerce site must have. Apart from that, use of color, fonts and images also plays a vital role to engage the user. These issues must be resolved. Mobile devices are the great source of users nowadays. So, this system must be mobile friendly and scalable to different screen sizes. This system must have features of search and filter products based upon categories and sub-categories. To ease the use for the native user, this system must have a language option to change the language from English to Nepali. Customers always want to hear from other customers how the shopping suits their needs and expectations. To solve this problem, reviews and ratings must be used. This system must have features of shopping cart and checkout for both logged-in users and anonymous users.

3.4.2. personalized recommendation system:

Most popular systems are dynamic and personalized to users. It is one of the key factors to engage the user with the system. Products are recommended to users based upon their preferences, user history and ratings. Products already rated by the user must not be recommended to the user.

3.4.3 Management requirements:

Management requirements contain only registered administrators who can access the admin panel and login to the admin panel of the system. Only administrators can upload and manage products, and customers can buy needed products from the system.

3.5 Non-functional requirement:

Non-functional requirements specify whether a particular task will be performed or not. Some of the non-functional requirements are listed below:

3.5.1. Speed:

Speed is actually much more complicated than it seems. Speed is one of the non-functional requirements that must be considered. The speed may lag because of

many reasons. Some of the common reason for slow load is due to high resolution unoptimized images, complex order entry process, use of many API calls to render, unclean code etc. All of these issues must be solved.

3.5.2. Security and privacy:

Since this web app use most sensitive data of the user. The system must be protected from vulnerabilities. The authentication must be secured from data breach. To protect the system from code manipulation, the software must be coded with much complexity. The privacy must be protected. Policies, terms and conditions must be able to update from the admin side.

3.5.3. Extensibility:

One of the strong points of this system is this system can be extended to large number of users and large number of products with different categories and sub categories because admin panel is responsible for the flash sale, slider, product, categories, subcategories and many more.

3.6 Usability requirement:

3.6.1 Ease of use:

The system must be easy to use and understand the system or the casual user. The navigation and function must be simple.

3.6.2 Satisfaction and memorability:

The design of the system must be pleasant to view and the user must be able to recall the site when user return after a certain period of time.

3.6.3 Graphics:

Graphics is considered as one of the components that helps user to better connect to the system. It mostly deals with data rendering and displaying high quality of data. The code must be optimized and should able to show similar quality in any hardware devices.

Chapter 4: design and architecture:

Different design and architecture are designed in this section. All the process database and system are designed in this section.

4.1. Process Design:

4.1.1 Use case diagram:

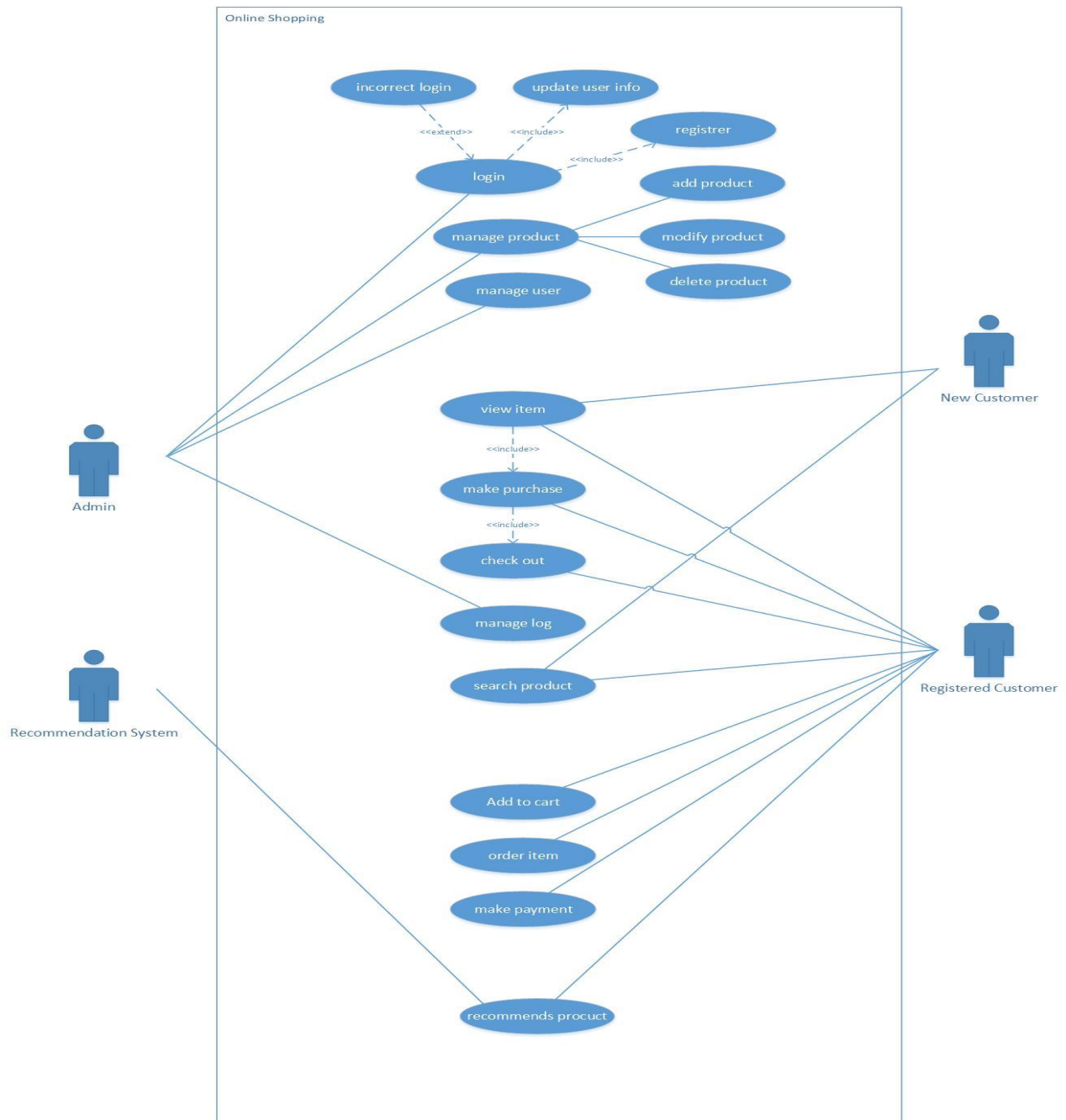


Figure 6 Use case diagram

4.1.2 Class diagram:

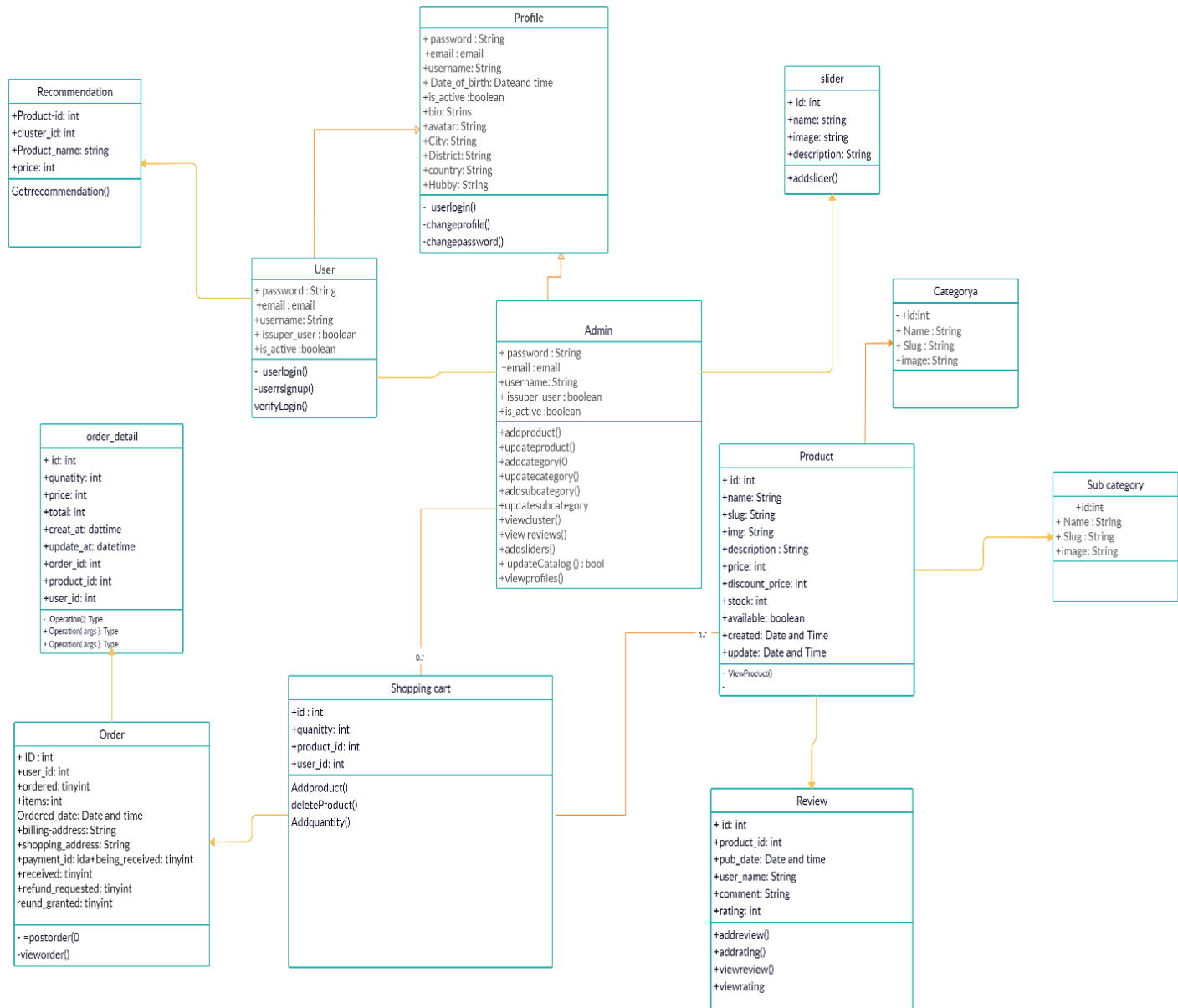


Figure 7 Class diagram

4.1.3 Sequence diagram:

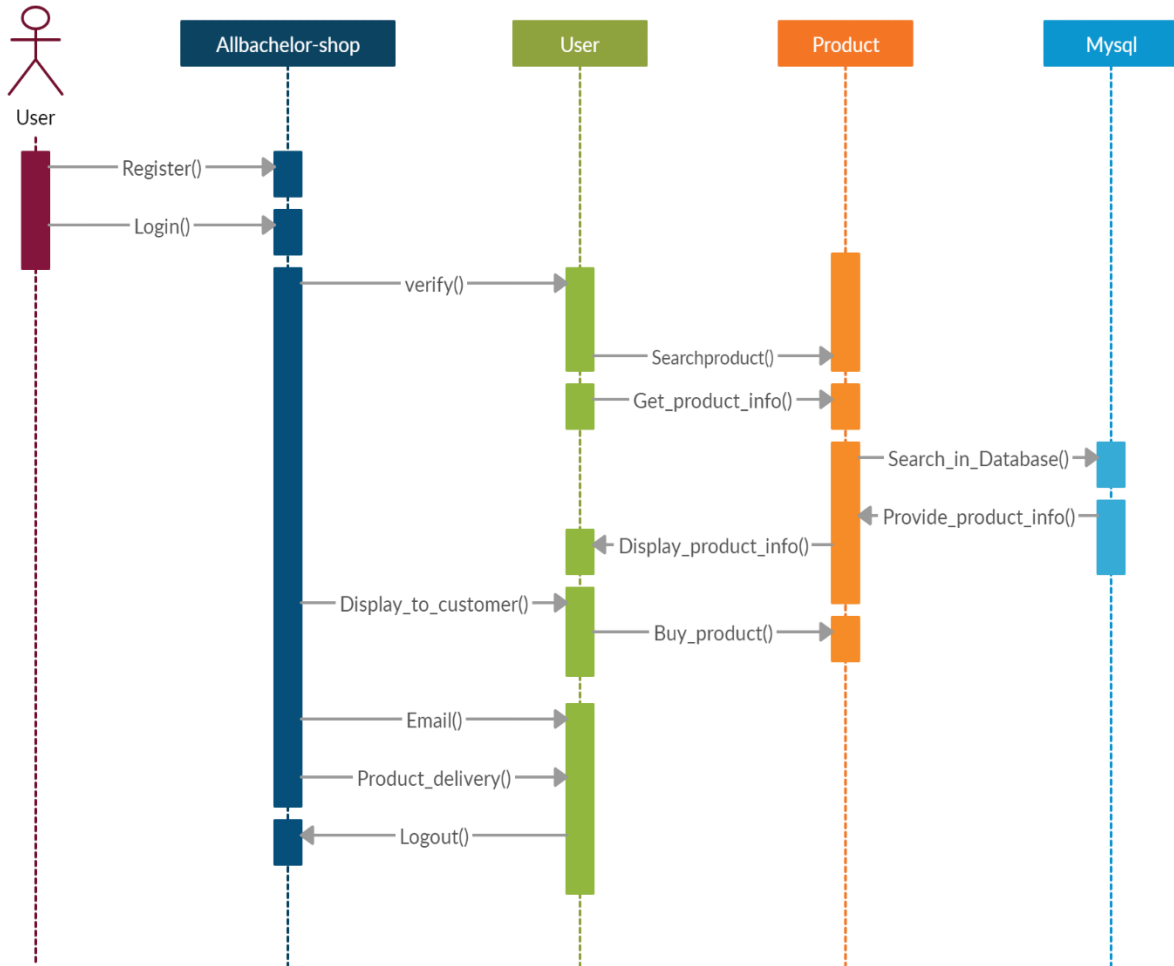


Figure 8 Sequence diagram

4.1.4 State Transition diagram:

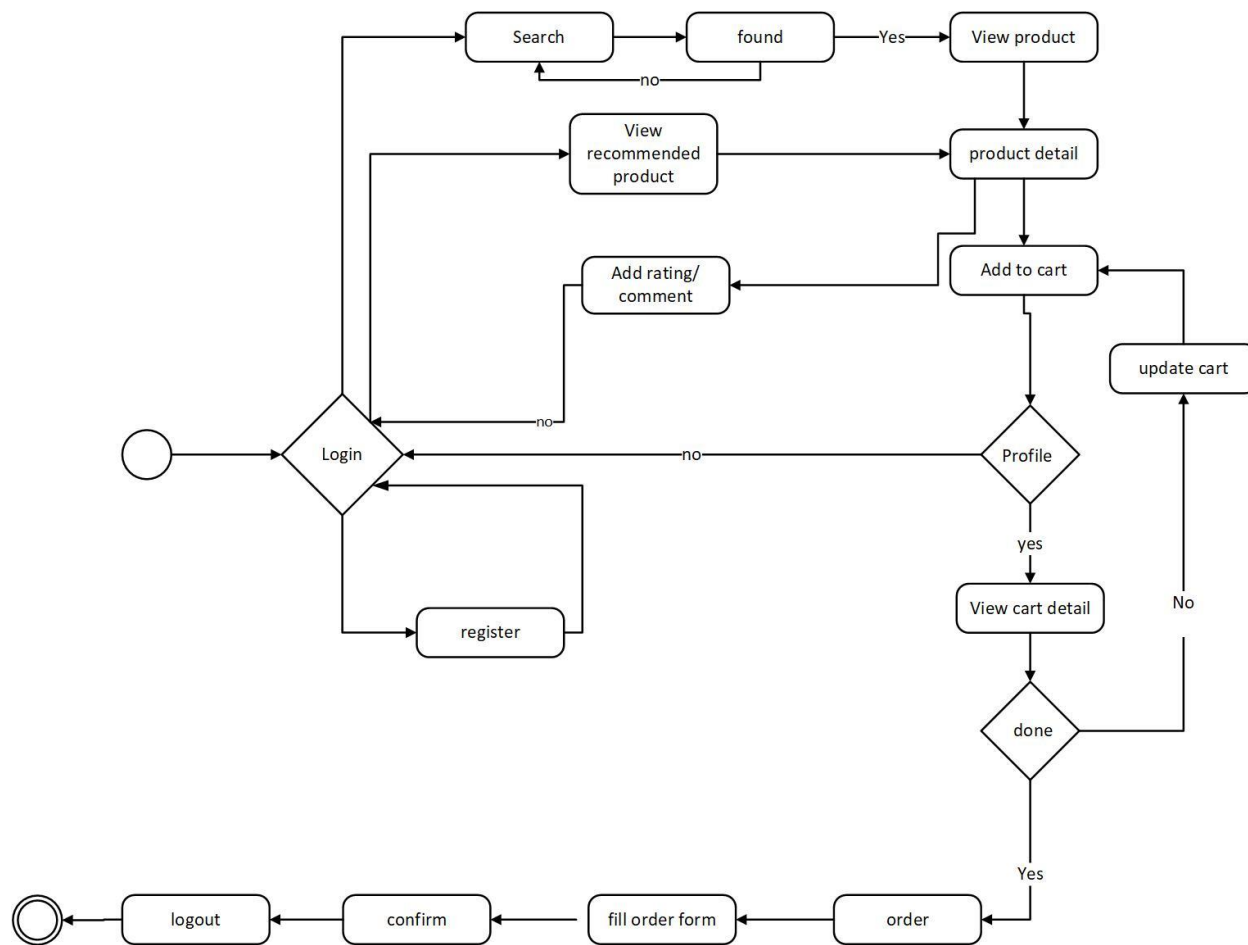


Figure 9 State transition diagram

4.1.5 Activity diagram:

4.1.5.1 Activity diagram for user:

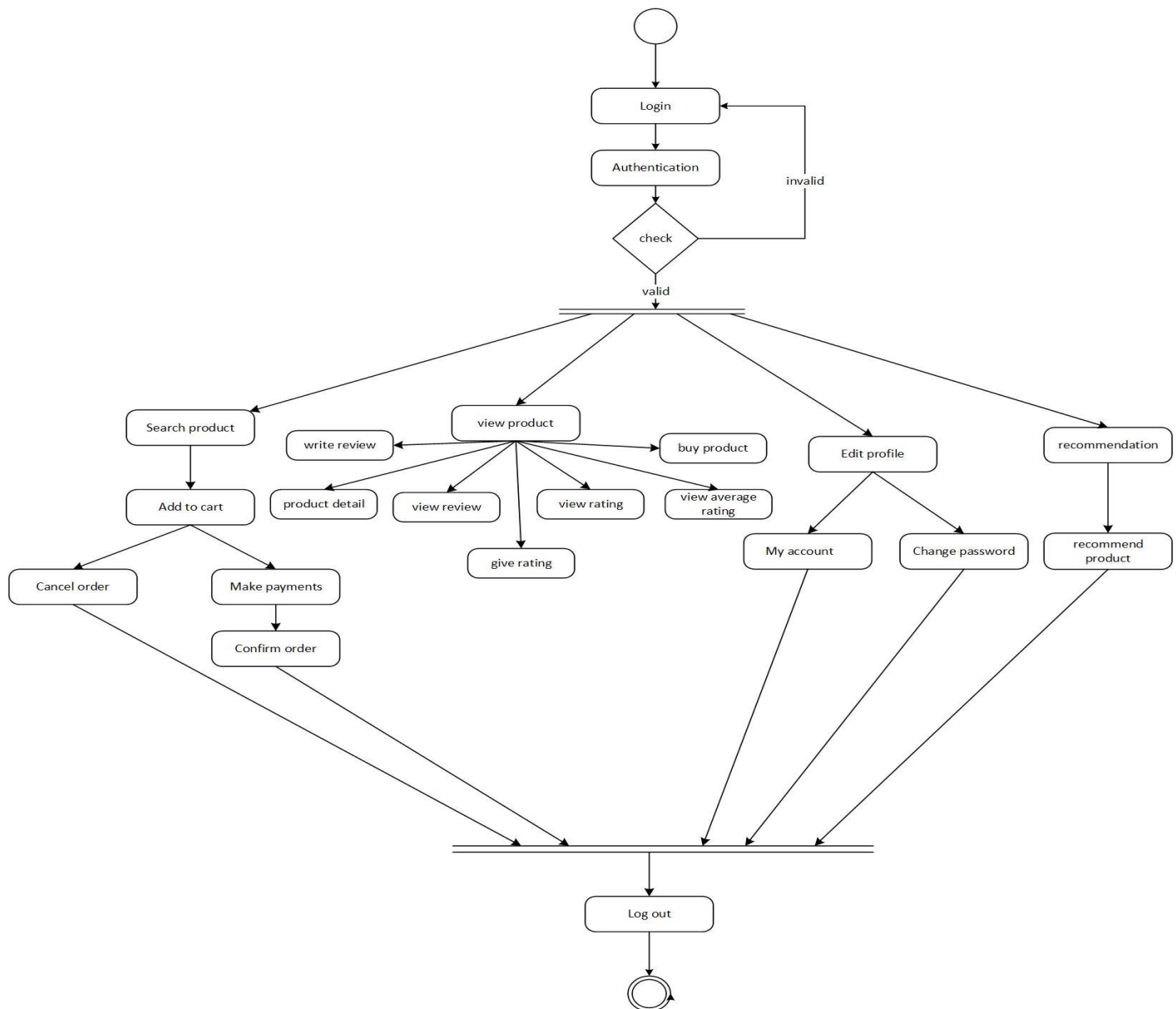


Figure 10 activity diagram for user

4.1.5.2 Activity diagram for admin:

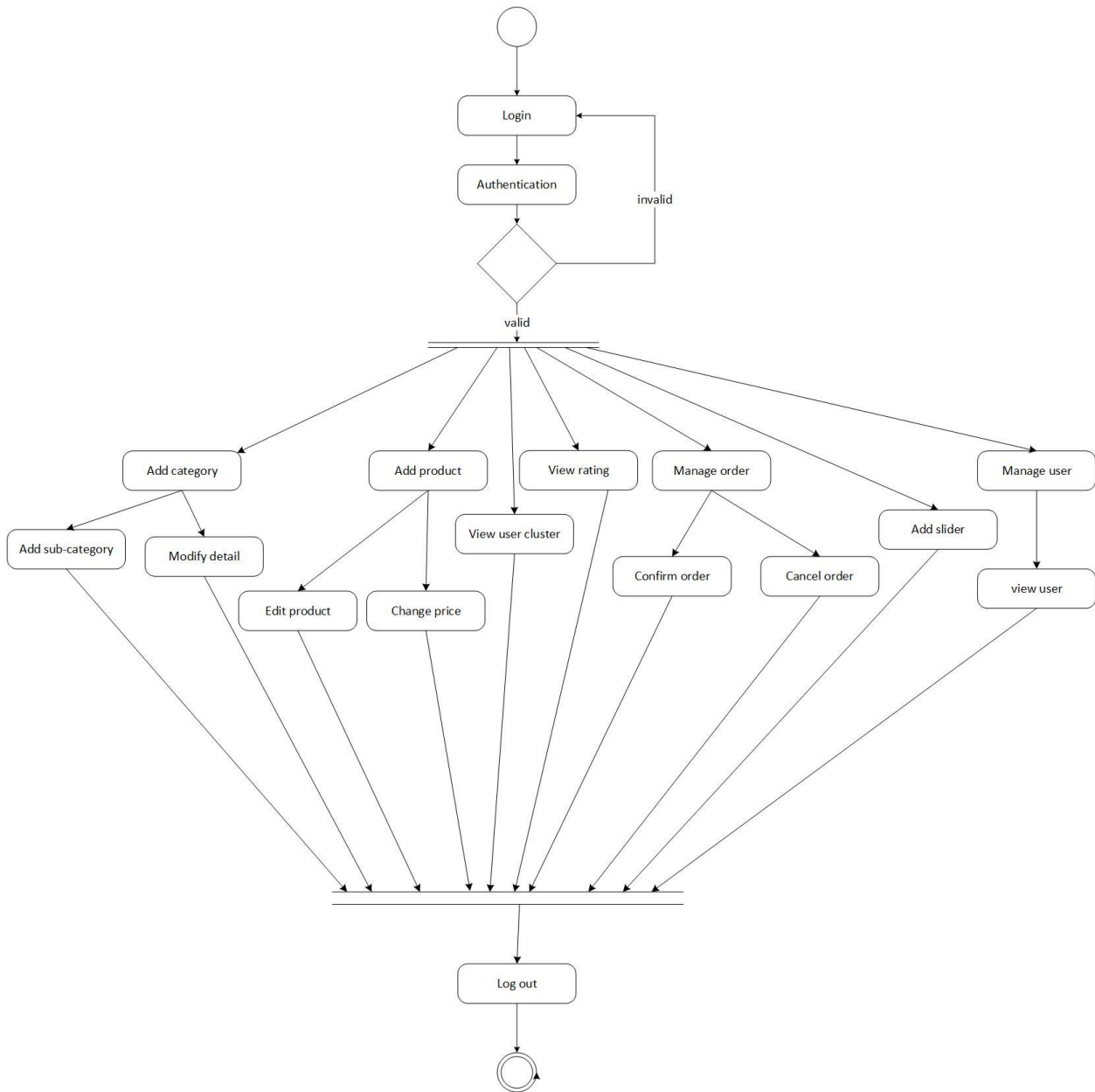


Figure 11 activity diagram for admin

4.2. Database design:

4.2.1 Entity relation diagram:

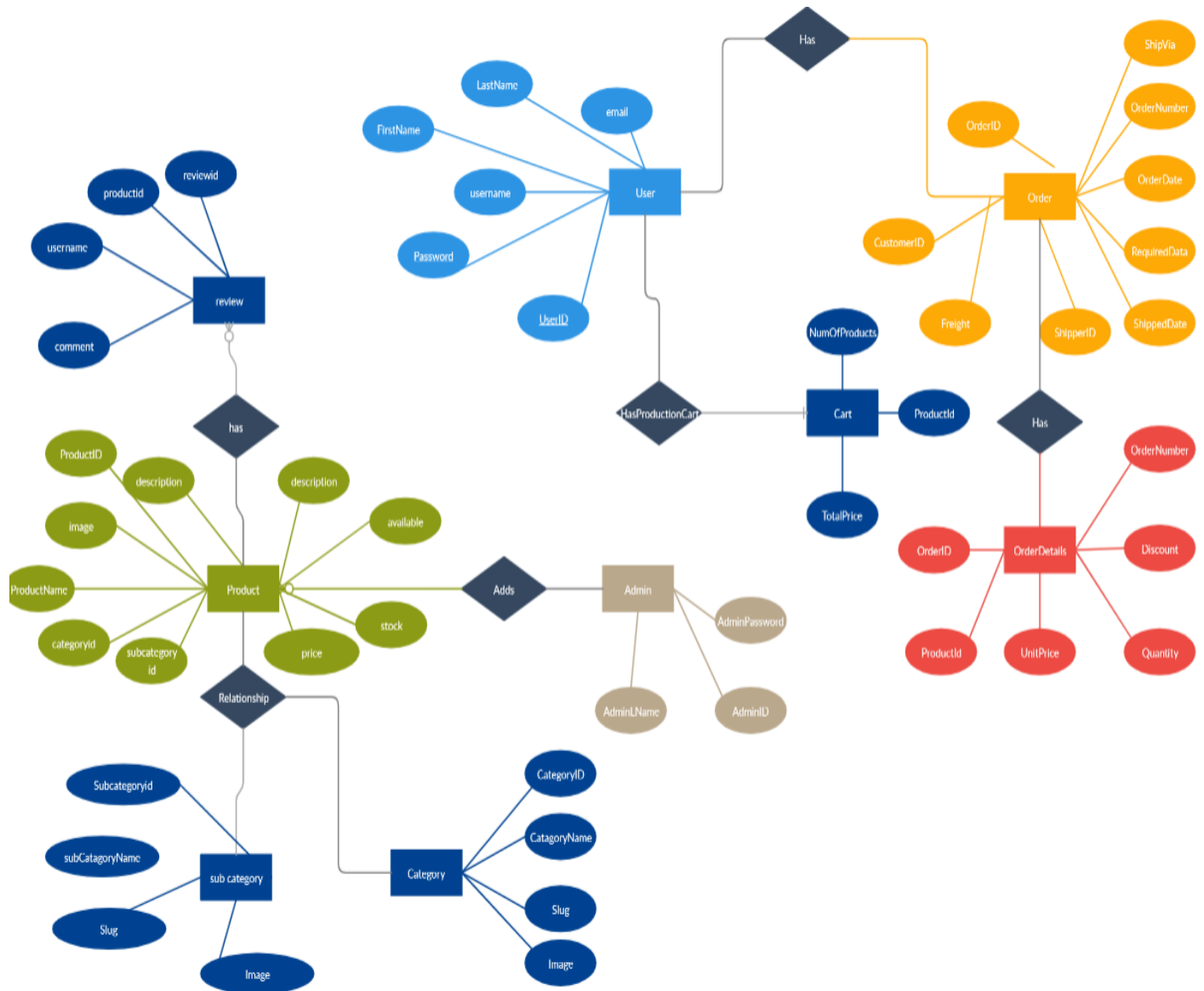


Figure 12 Entity relation diagram

4.2.1 Data dictionary diagram:

Table	Column	Datatype	Length	Nullable	Primary key
User	id	int	11	No	Yes
	password	varchar	128	No	No
	Last_login	datetime	6	Yes	No
	Is_superuser	tinyint	1	No	No
	username	varchar	150	No	No
	First_name	varchar	100	No	No
	Last_name	varchar	150	No	No
	email	varchar	254	No	No
	Is_staff	tinyint	1	No	No
	Is_active	tinyint	1	No	No
	Date_joined	Date_joined	6	No	No
	Profile	Id	int	11	No
First_name		varchar	100	No	No
Last_name		varchar	150	No	No
email		varchar	100	No	No
Date_of_birth		date	6	No	No
bio		text	-	No	No
avatar		varchar	100	No	No
slug		varchar	100	No	No
city		varchar	100	No	No
hobby		varchar	100	No	No
district		varchar	100	No	No
category	country	varchar	100	No	No
	id	int	11	No	Yes
	name	varchar	100	No	No
	slug	varchar	100	No	No
	image	varchar	100	No	No

	id	int	11	No	Yes
subcategory	name	varchar	100	No	No
	slug	varchar	100	No	No
	image	varchar	100	No	No
	id	int	11	No	Yes
slider	name	varchar	100	No	No
	description	varchar	100	No	No
	image	varchar	100	No	No
	Id	int	11	No	Yes
	name	varchar	150	No	No
Product	slug	varchar	100	No	No
	image	varchar	100	No	No
	description	varchar	100	No	No
	price	int	11	No	No
	Discount_price	int	11	No	No
	stock	int	11	No	No
	available	tinyint	1	No	No
	Created	datetime	11	No	No
	updated	datetime	11	No	No
	id	int	11	No	Yes
	Product_id	int	11	No	No
	Pub_date	datetime	11	Yes	No
review	comment	varchar	200	No	No
	username	varchar	100	No	No
	rating	int	11	No	No
	id	int	11	No	Yes
	User_id	int	11	No	No
	Ref_code	varchar	100	Yes	No
	ordered	tinyint	1	No	No
	items	int	11	No	No

	Start_date	datetime	11	No	No
Order	Ordered_date	datetime	11	No	No
	Billing_address	varchar	100	No	No
	Shipping-address	varchar	100	No	No
	received	tinyint	1	No	No
	Being_received	tinyint	1	No	No
	id	int	11	No	Yes
	User_id	int	11	No	No
order item	Item_id	int	11	No	No
	quantity	int	11	No	No
	ordered	tinyint	1	No	No

Table 1 data dictionary diagram

4.3. User interface:

4.3.1 Wireframe:

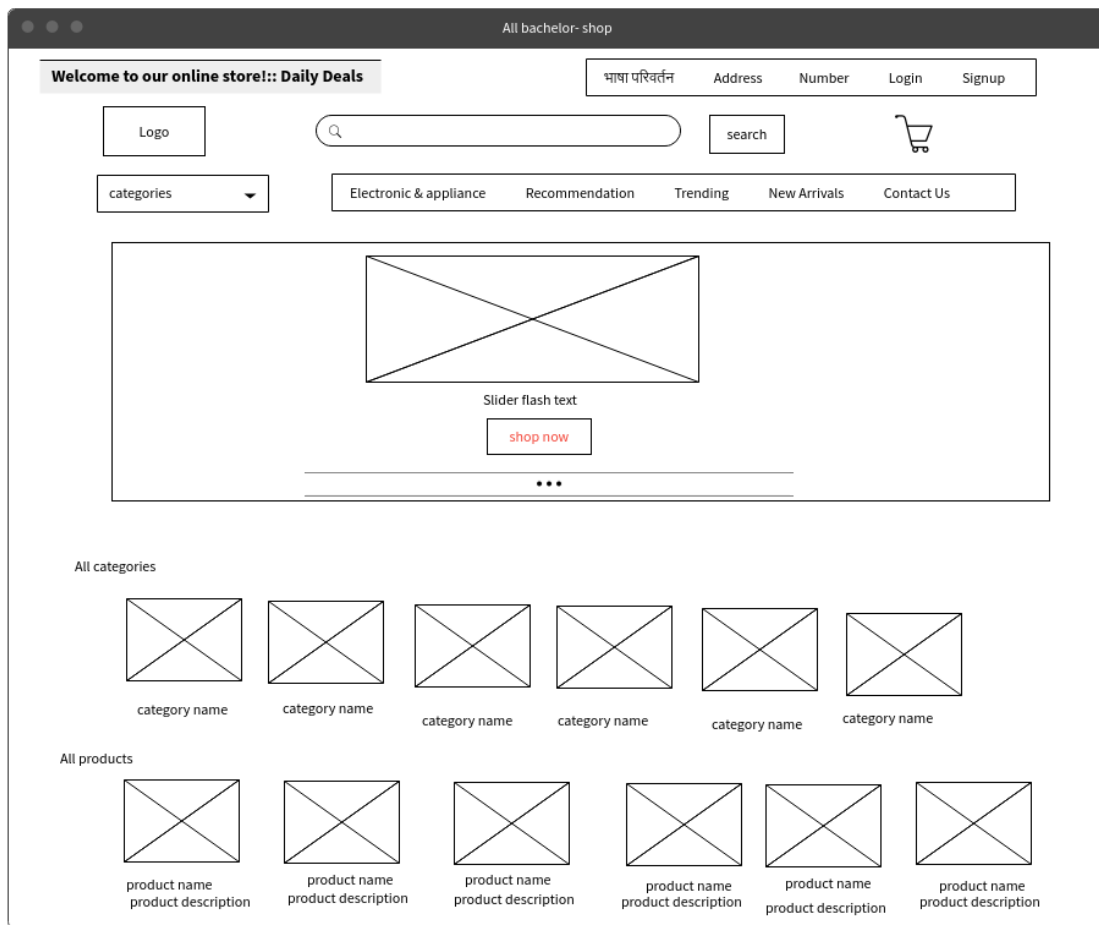


Figure 13 wireframe of index page

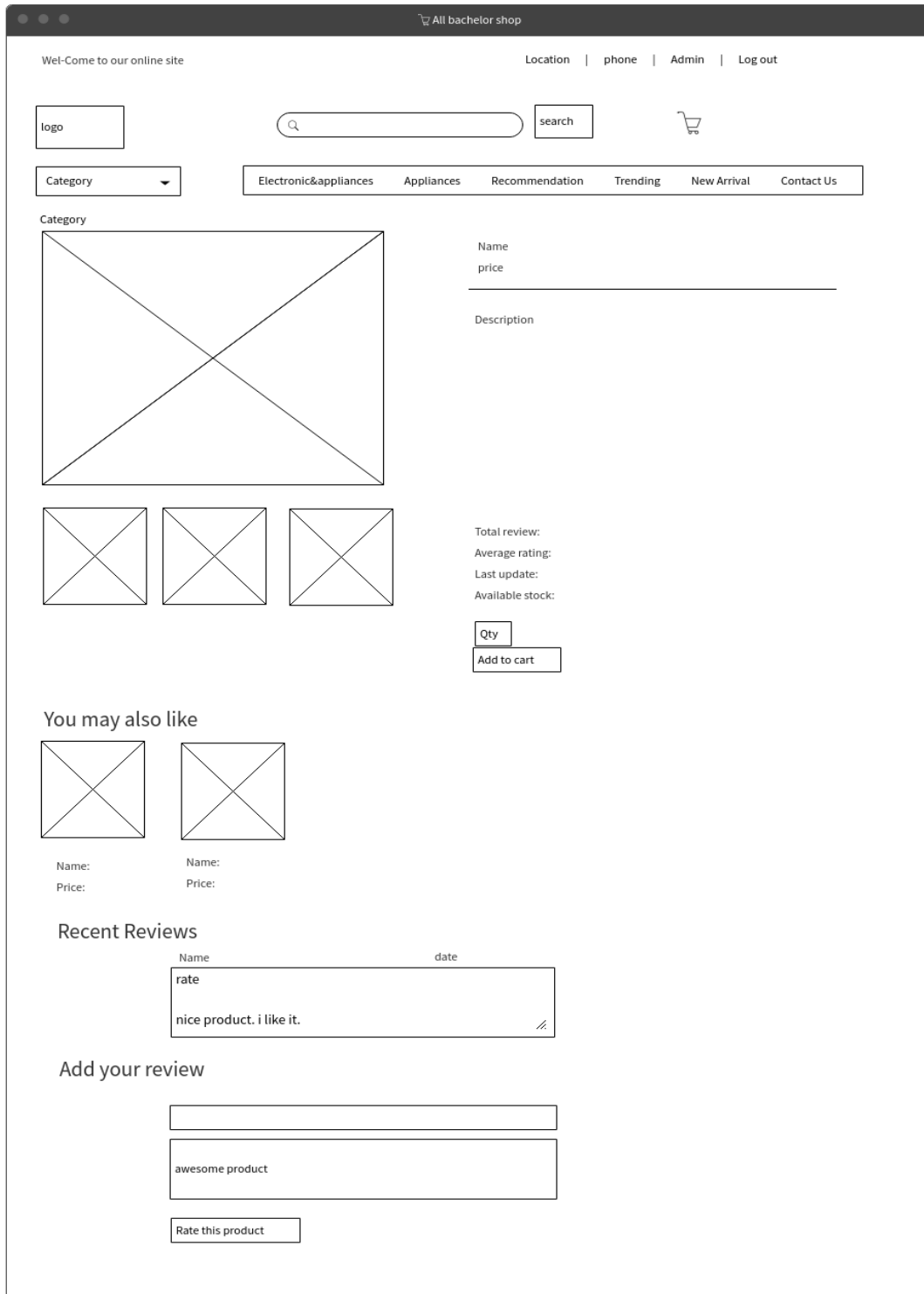


Figure 14 Product detail page

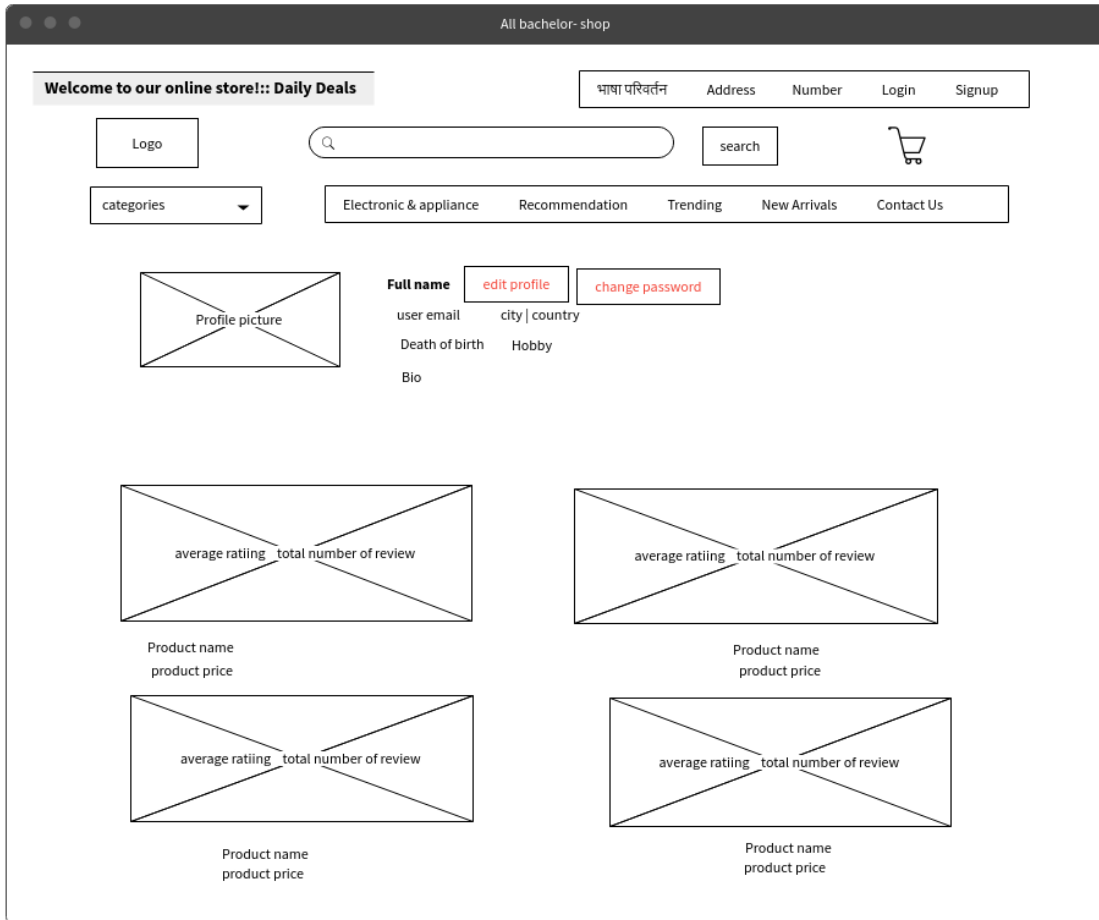


Figure 15 profile page

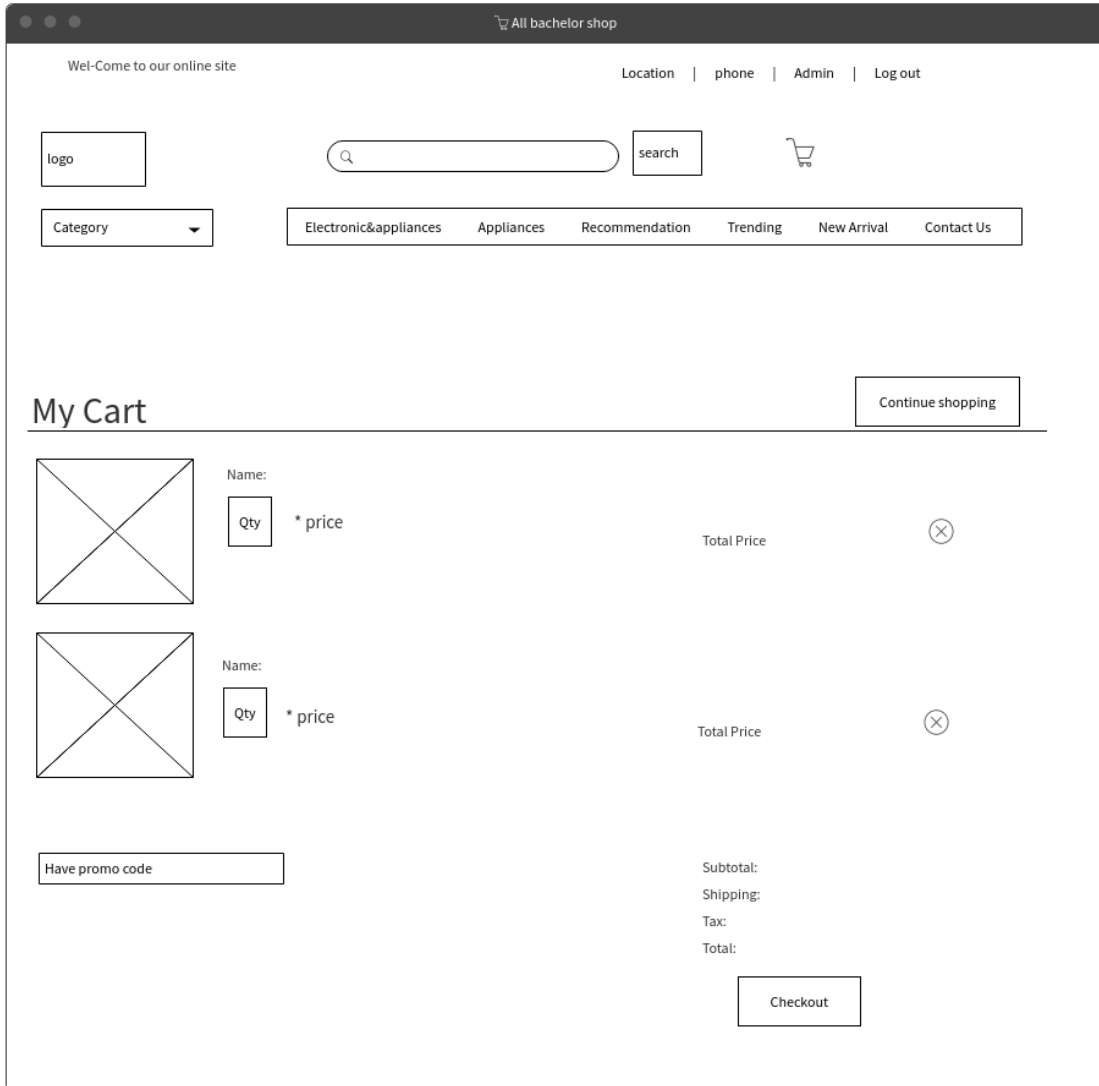


Figure 16 My cart page

All bachelor shop

Wel-Come to our online site

Location | phone | Admin | Log out

logo search

Category

Order details

Name of the card

Surname on card

Address

City

Contact Number

Total payment

Order review

	Name: <input type="text" value="1"/>	Price: <input type="text"/>	<input type="button" value="X"/>
--	--------------------------------------	-----------------------------	----------------------------------

Figure 17 order details

4.4. System architecture:

4.4.1 The MVC pattern:

This site uses MVC (model, view, controller) pattern instead of traditional server-side code. As I am using Django as a framework, Django differentiate business logic from representation. This architecture splits an application into three strata/layer. They are: Model, View and controller. Model mostly deals with the database. To be more specific, models defines the table of database. Views are what we see in browser or what the user views when visited. View mostly deals with the client or end user of the system. Controller helps to control information flow. The controller is the center of the MVC architecture, as it helps to connect models with views. Thus, MVC architecture helps us to create and maintain complex application. Django uses MTV (model, template, view) architecture. Only the terminology used in Django are different. The template is used instead of views and views instead of controller.

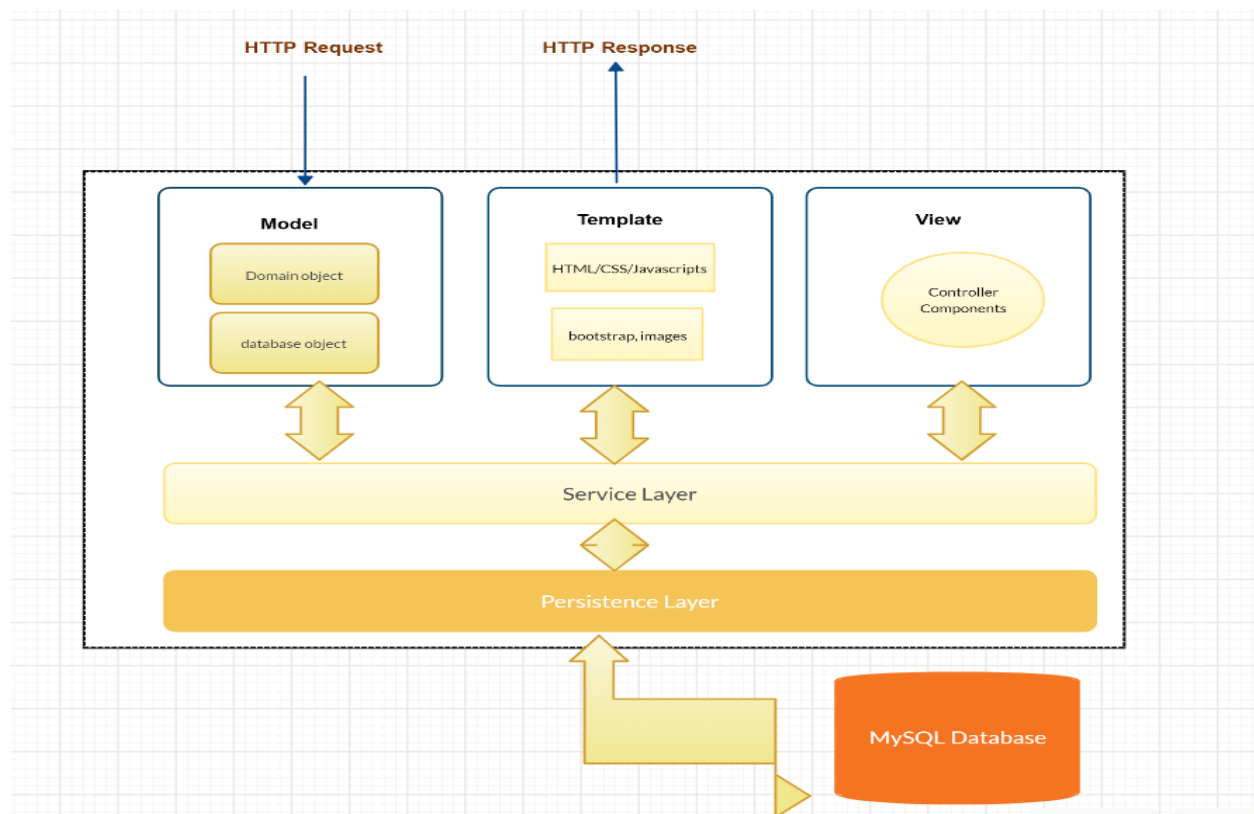


Figure 18 MTV architecture Django

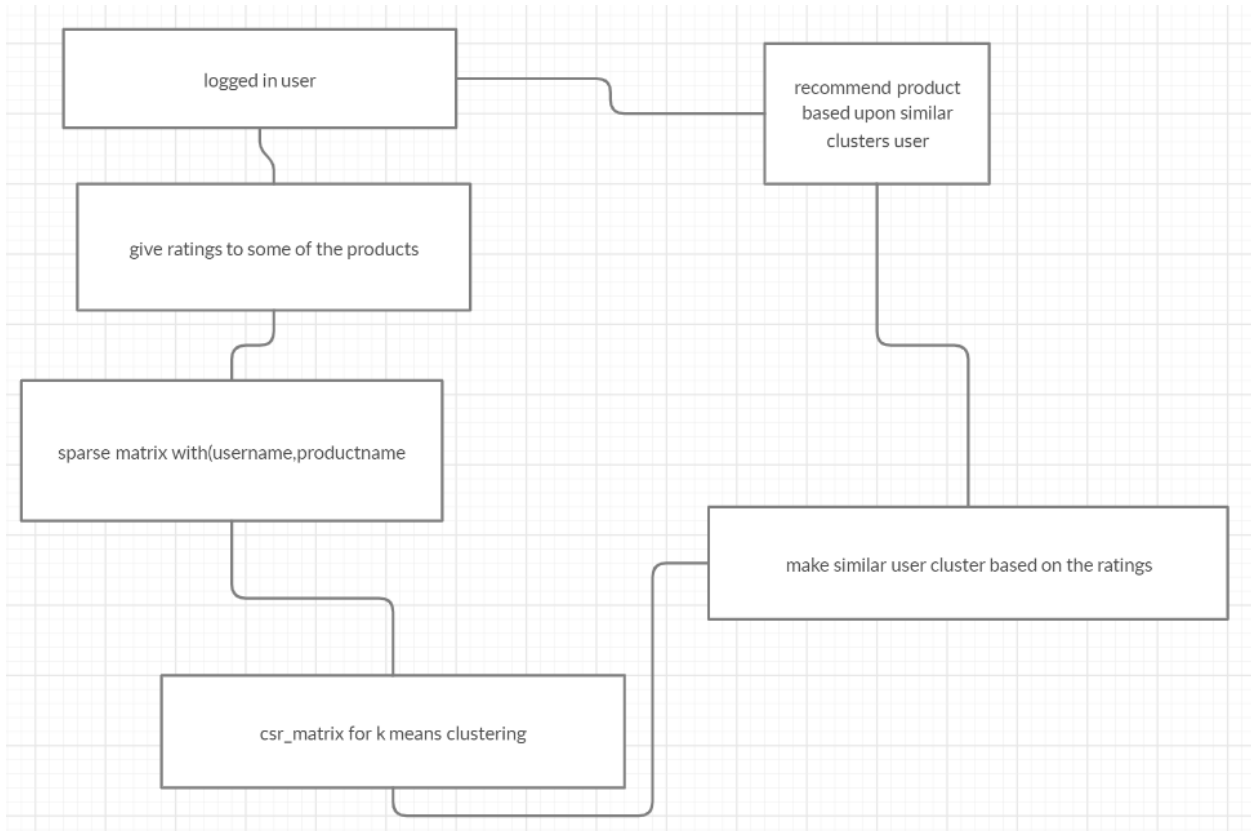


Figure 19 system architecture for recommendation system

Chapter 5 implementation and development process:

One of the key features of this system is to develop the recommendation for a given active user. The development component is then split into three major phases: firstly, the implementation of ecommerce system; secondly, the development of a recommendation system that can give reliable products independent of the size of the dataset and thirdly integrating the recommendation model to web app. First step is the downloading and installing all the necessary packages needed to develop the tool. Before development of the system, some of the software is installed as shown in technical specification below:

5.1.1 Technical specification:

Framework: Django 3.2

Library: bootstrap3, numpy, pandas, matplotlib, scipy, scikit-learn

IDE: Pycharm, Jupyter notebook, spyder

Text editor: sublime text

Language: python 3.8.3

Database: MySQL 8.0

Browser: Google Chrome

Datasets: Amazon dataset, user generated dataset

Front end: Html, CSS, JavaScript

Documentation, diagram and presentation: Microsoft Office, PowerPoint

Visual Prototyping tool: creatly, visual studio, Mockups

5.1.2 Development:

After installing all above packages and library, it's time to develop the ecommerce system. First front-end part is made just like the wireframe mentioned above with the help of html, css, javascript and bootstrap using sublime as a text editor. After that front end is connected to backend database MySQL with the help of python and Django. The database is designed like the above-mentioned entity relation database. All of these

features are discussed in next chapter in detail. The only problem faced during the development of this system is generating recommendation as I have very less number of data and my database does not match with the dataset given by any dataset provider. After successfully developing ecommerce site. I tried and tested all the recommendation system in Jupyter notebook and Spyder which is also discussed in next chapter.

The product has different features like productid, product description, rating, comments, category, sub category etc. The recommendation is completely based upon the rating. Rating 1, 2 means not liked product and rating 4,5 are considered as the liked product. So, I made the recommendation system by using cluster of similar users. First the query set is done to the logged in active user for all the rating that the logged in user has provided. The data is fetched by using the Review class. Review class has review id, rating, user_id, comment. In similar fashion all the information of user. Only fetching the data from database and recommending the random products from the database is not the wise way and I never wanted to do that. First things needed to recommend the product is to find out preferences of the user. So, for this purpose the newly logged in fresh user is asked to rate some of the products of the system. The data gets updated once the user rate any new product. The similar user is calculated by the rating given the user u to product p assumed that the user in particular cluster have same preferences with each other as they have rated the products in similar manner. In similar fashion, other logged in user are also categorized in cluster. All the inactive users of the system are placed in one of the clusters. The users with no ratings in any of the products are also considered as inactive users, those users are encouraged to rate the product to get the recommendation. This system works perfectly in only three condition: product is not deleted in database, the product has been rated positive by the similar user which are stored in same clusters as in logged in user and lastly , the products that were never reviewed by the logged in user are shown to the user. The already rated product by user is never shown in recommended part of the system. After successfully making and updating the cluster based upon the user preferences. Similar user's cluster data are fetched to the system to recommend the product. Then after all the product rated by similar user excepts the rating already given by the user. Lastly, based upon the average rate, the products are displayed in user's recommendation page.

To automate the task, instead of making cluster by only the admin, the cluster must be updated by the user's activity in the system. Every time it would not be feasible to view and update the similar clusters from the admin side. Recommendation works with data and knowledge provided by the user to the system. I used clustering based recommendation because it is not always feasible to compare the logged in user with any other user in the system whenever recommendation is required. So, to increase the speed of the prediction, I use cluster of the similar user based upon the preferences, system tries to predict the rating for the system. The cluster must not be calculated immediately after a user rate new product, otherwise it would lead to scalability problem which was one of the reasons behind choosing clustering technique.

The system works as shown in figure above in system architecture. First of all, the sparse matrix is created using the rating provided by the user and their user information. I have made the matrix with row of username and column with product name. in simpler term, it can be said that the matrix R_1, R_2, \dots, R_n is the username1, username2...username n and c_1, c_2, \dots, c_n is the productname1, productname2 to till n. For each element (r,c) contains username with their corresponding rating to the user. For each user, there will be rating for each product. In order to make the sparse matrix, I used to prebuild function `dok_matrix` of scipy library. I don't want to make this system complex. I want to make the project simple, so made 1/10th of the user as cluster. This code must be changed if the system is used for the large number of user and rating. `dok_matrix` is converted to `csr_matrix` to calculate the k means clustering. Both `dok_matrix` and `csr_matrix` are class of scipy library which can be found on the documentation page of the scipy library.

The review, rating and user information can also be passed from csv dataset and read from csv files using panda's library. Since I don't want to make this system more complex. I have used in the real database instead prebuild dataset. The cluster is then updated in database.

There are some limitations of using this system which I have discussed in chapter 8. K-means is not most common model for recommending product. The only reason behind choosing this algorithm is simplicity and fast recommendation as it uses parallel and

scalable approach. It is best for the developer like me to understand and is very easy to maintain.

Another recommendation that I have made during this process is popularity based recommendation system which can be shown to the user if no recommendation are shown to the user. The popularity is calculated by using the sum of rating times the total number of rating. however, this popularity-based system can be improved.

I have also used tf-idf technique for recommending products that has similar word in the description box. The similar products are displayed so, it is a content-based filtering.

Before making all of this recommendation, I have calculated and used different approach and techniques of recommendation which can be seen in Jupyter notebook. First, I tried to calculate the score similarity between users in the system. For simplicity, I used two users from the dataset and try to find out the similarity between those two users who have rated already.

```
] cosine_similarity('user1', 'user4')
```

```
] 0.5412746144356352
```

```
] euclidian_similarity('user1', 'user4')
```

```
] 6.557438524302
```

```
] kerl_person_coefficient('user1', 'user4')
```

```
] -0.13710212427677046
```

Figure 20 Euclidean score and Pearson score

After finding, similarity score, products are displayed to the similar user based upon the rating provided by the similar

I later made the algorithm based upon collaborative algorithm on amazon dataset on both user-user based and item- item based algorithm. To increase the accuracy, I removed all the ratings with 3 values from the dataset as three ratings does not directly determine either the user rated positive or negative.

I replaced all the zero-value rated product in a sparse matrix with mean value of the column to deal with sparsity and tried to compare the result with the one that was replaced with baseline rating (3) because the sparsity of the dataset becomes 0.99367, That means only 0.067% matrix has the value.

The RMSE in later model where nan value is replaced by 3 ratings (baseline rating) extremely diminishes and the prediction for the test dataset is also very remarkable and notable. The second model says that the use who have not rated the product may give the baseline rating but first model says that the user who have not rated surely dislike the product. The RMSE score is also tabulated below:

Approach	RMSE value for normal rating	RMSE for baseline rating
User-user based CF	4.35927000050732	1.7277263715301425
Item-Item based CF	4.35927000050732	1.7277263715301425
Latent factor model	4.2112786441410694	1.6781041043440479

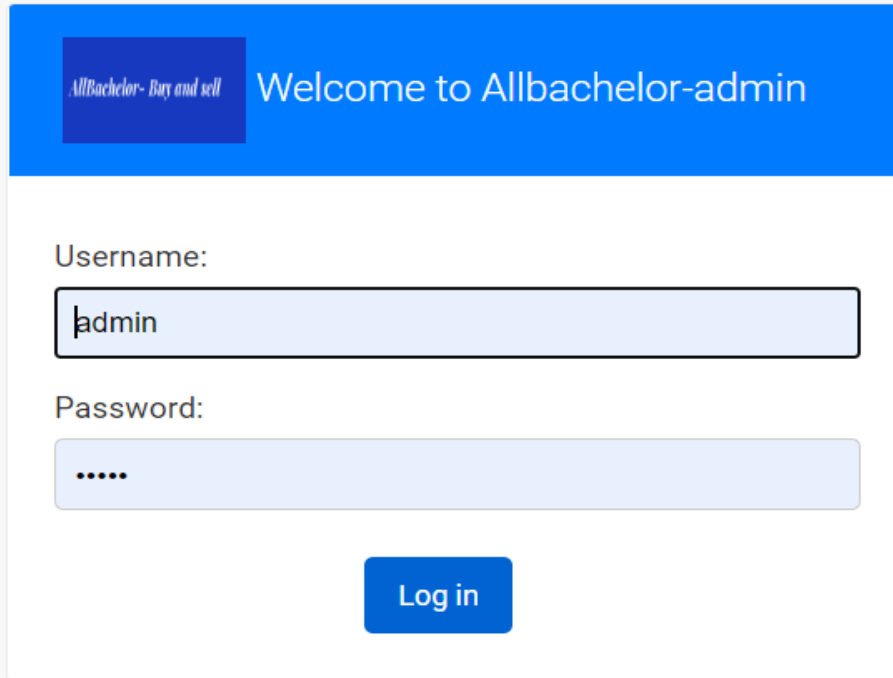
Table 2 comparison table of model

The performance and RMSE value for normal ration is very high than that of using baseline rating. Also, Model based approach latent factor model's RMSE for both normal rating and baseline rating is very good. So, latent factor model is also used in this system.

Chapter 6: walkthrough of system:

A walkthrough of the application is illustrated in this chapter. Examples of features are presented for each user type: new user, returned user and administrators. Additional functionality and features that are common to all users can be found in Appendix.

6.1.1 Admin panel:



AllBachelor- Buy and sell Welcome to Allbachelor-admin

Username:

Password:

Figure 21 admin login

allbachelor by mail Welcome to Allbachelor-admin WELCOME, ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Site administration

AUTHENTICATION AND AUTHORIZATION	
Groups	+ Add Change
Users	+ Add Change

PROFILES	
Profiles	+ Add Change

PYTHON SOCIAL AUTH	
Associations	+ Add Change
Nonces	+ Add Change
User social auths	+ Add Change

SHOP	
Categories	+ Add Change
Clusters	+ Add Change
Products	+ Add Change
Reviews	+ Add Change
Sliders	+ Add Change
Subcategories	+ Add Change

Recent actions

My actions

- + samsungj1 Product
- ✖ J1 Product
- ✖ samsungj1 Product
- ✖ Bishal khadka Product
- 🔧 samsungj1 Product
- 🔧 samsungj1 Product
- 🔧 J1 Product
- 🔧 samsungj1 Product
- 🔧 Bisha khadka Profile
- + Bishal khadka Profile

Figure 22 Admin Dashboard

allbachelor by mail Welcome to Allbachelor-admin WELCOME, ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Shop > Categories > Add category

Add category

Name:

Slug:

Image: a6.jpg

Figure 23 Add category

allbachelor by mail Welcome to Allbachelor-admin WELCOME, ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Shop > Sliders > Add slider

Add slider

Name:

Image: b1.jpg

Description:

Figure 24 add sliders

lms-bachelor-admin | Welcome to Allbachelor-admin | WELCOME, ADMIN | VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Shop > Subcategories > Add subcategory

Add subcategory

Name:

Slug:







Category:  

Figure 25 Add sub category

Home > Shop > Products > Add product

Add product

Category:  

SubCategory:  

Name:

Slug:

Image:

Description:

Price:

Discount price:

Stock:

Available

Figure 26 Add products

Welcome to Allbachelor-admin

WELCOME ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home - Authentication and Authorization - Users

Select user to change ADD USER +

Q Search

Action: Go 0 of 13 selected

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	admin	bishalkhadka75919@gmail.com			✔
<input type="checkbox"/>	admin1	bisf			✘
<input type="checkbox"/>	bishal	bishalkhadka75919@gmail.com			✘
<input type="checkbox"/>	bishal1234	bishalkhadka75919@gmail.com	Bishal	khadka	✘
<input type="checkbox"/>	bishalkhadka75919	bishalkhadka75919@gmail.com	bishal	khadka	✘
<input type="checkbox"/>	hawa	mareshworhamal@yahoo.com			✘
<input type="checkbox"/>	herald				✘
<input type="checkbox"/>	ma	mareshworhamal@yahoo.com			✘
<input type="checkbox"/>	mummy	bishal.khadka46@yahoo.com			✘
<input type="checkbox"/>	poonam	bishalkhadka75919@gmail.com			✘
<input type="checkbox"/>	punam	bishal.khadka46@yahoo.com			✘
<input type="checkbox"/>	test	test@h.c			✘
<input type="checkbox"/>	test1	test@gmail.com			✘

13 users

FILTER

By staff status

All
Yes
No

By superuser status

All
Yes
No

By active

All
Yes
No

Figure 27 view users

Welcome to Allbachelor-admin

WELCOME ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home - Shop - Clusters

✔ The cluster "Cluster object (23)" was changed successfully.

Select cluster to change ADD CLUSTER +

Action: Go 0 of 3 selected



<input type="checkbox"/>	NAME	GET MEMBERS
<input type="checkbox"/>	2	admin bishal
<input type="checkbox"/>	1	bishal
<input type="checkbox"/>	0	admin1 bishal1234 bishalkhadka75919 hawa herald ma punam test

3 clusters

Figure 28 Viewing similar cluster

Home > Profiles > Profiles > Bishal khadka

Change profile HISTORY VIEW ON SITE >


User:  


First name:

Last name:

Email:

Email verify:

Date of birth: Today 
Note: You are 10.75 hours ahead of server time.

Bio: 

Avatar: Currently: avatar/photo-1511765224389-3710e77c0eb.jpg Clear
 Change: No file chosen

City:

District:

Figure 29 viewing profile information of user

allbachelor Welcome to Allbachelor-admin WELCOME: ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Order > Order details

Select order detail to change ADD ORDER DETAIL +

Action: 0 of 6 selected

<input type="checkbox"/>	USER	PRODUCT	PRICE	TOTAL	UPDATE AT
<input type="checkbox"/>	admin	Gaming pc	150000	140000	May 26, 2020, 3:42 a.m.
<input type="checkbox"/>	admin	samsungl1	12000	11000	May 26, 2020, 3:42 a.m.
<input type="checkbox"/>	admin	Gaming pc	150000	140000	May 26, 2020, 3:42 a.m.
<input type="checkbox"/>	admin	samsungl1	12000	11000	May 26, 2020, 1:03 a.m.
<input type="checkbox"/>	punam	samsungl1	12000	11000	May 25, 2020, 11:29 p.m.
<input type="checkbox"/>	punam	Gaming pc	150000	140000	May 25, 2020, 11:29 p.m.

8 order details

Figure 30 viewing order detail

allbachelor Welcome to Allbachelor-admin WELCOME, ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home » Order » Orders

Select order to change ADD ORDER +

Action: Go 0 of 13 selected

<input type="checkbox"/>	USER	NAME	SURNAME	CITY	PHONE	TOTAL	STATUS
<input type="checkbox"/>	admin	Bishal	khadka	Kathmandu	9808349417	140000	New
<input type="checkbox"/>	admin	Bishal	khadka	Kathmandu	9808349417	151000	New
<input type="checkbox"/>	admin	Bishal	khadka	Kathmandu	9808349417	11000	OnShipping
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New
<input type="checkbox"/>	punam	Bishal	khadka	Kathmandu	2147483647	151000	New

13 orders

FILTER

By status

- All
- New
- Accepted
- Preparing
- OnShipping
- Completed
- Cancelled

By create at

- Any date
- Today
- Past 7 days
- This month
- This year

Figure 31 view orders

allbachelor Welcome to Allbachelor-admin WELCOME, ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home » Order » Shop carts

Select shop cart to change ADD SHOP CART +

Action: Go 0 of 1 selected

<input type="checkbox"/>	USER	PRODUCT	QUANTITY
<input type="checkbox"/>	punam	Gaming pc	1

1 shop cart

Figure 32 view shopcart of user

allbachelor Welcome to Allbachelor-admin WELCOME, ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home » Order » Orders » Status

Change order HISTORY

Note:

This product is on the way to your specified place

Status:

Name:

Surname:

Address:

City:


Phone:

Total:

User:

Figure 33 Update order product information

6.1.2 user view:



CREATE AN ACCOUNT

Username
test1

Email
test@gmail.com


Password
.....

Re Password
.....

Already an account [Sign in](#)

SUBMIT

Figure 34 user sign up



WELCOME BACK

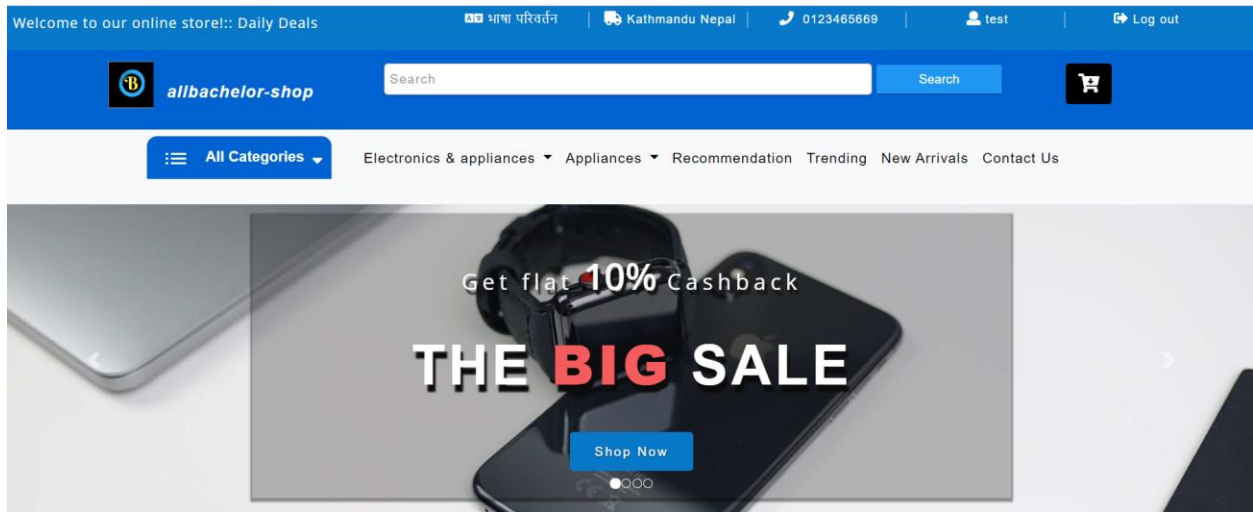
Username
test

Password
.....

Want to sign up with email? [Sign Up](#)

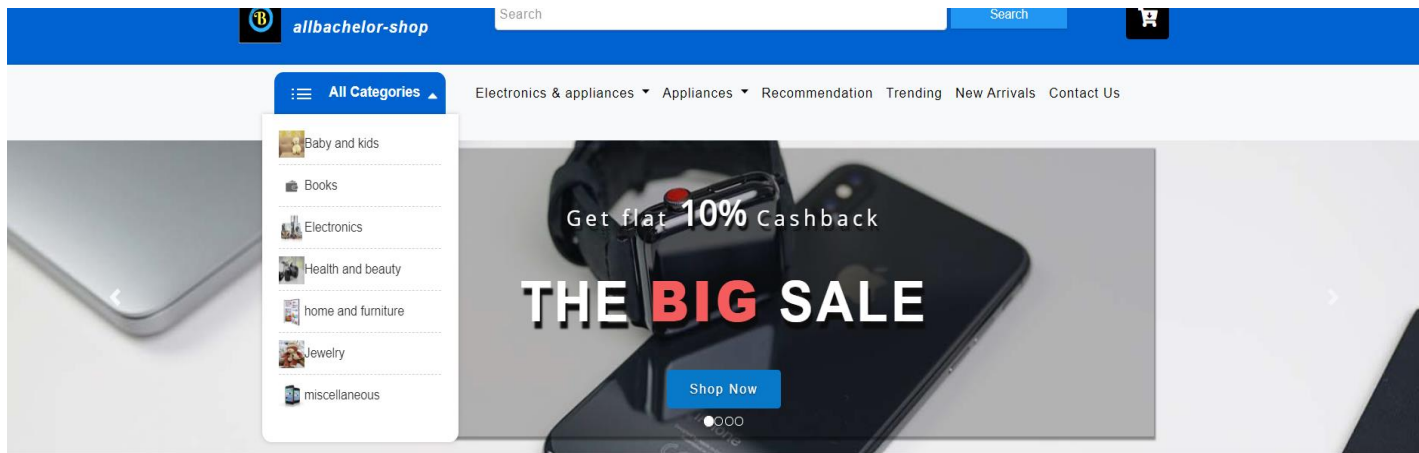
SUBMIT

Figure 35 user sign in



ALL CATEGORIES _____

Figure 36 Home page



ALL CATEGORIES _____



Figure 37Detail homepage

Result for gaming_pc

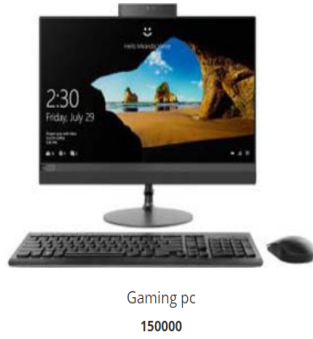


Figure 38 Search list


Electronics



Figure 39 View category wise product

All Categories | Electronics & appliances | Appliances | Recommendation | Trending | New Arrivals | Contact Us

PRODUCT / ELECTRONICS /



Dell Inspiron 5477 i7 8th gen GTX 1050

Price: ~~Rs.146000~~ Rs.140000

Product model:X7 Macros Define the game mouse 100% new mouse game High-end brand macro definition gaming mouse Human body engineering design 6 programmable keys Cool LED variable light effect Item Size : 124*76*43mm(L * W * H) Support Windows 2000 / XP / win7 / win8 /win10 Vista 32bit IOS or latest.

Total reviews:0 reviews
Average rating: nan average rating
Last updated: May 26, 2020, 4:07 a.m.
Available stock: 12 only

1 **ADD TO CART**

Figure 40 Product detail page 1

You may also like



Gaming mouse

Rs.190.99

Recent reviews

punam verified comment May 26, 2020, 3:07 p.m.

Rated 3 of 5 by punam

Good for budget gaming mice ! good weight and performance but the software is little difficult to use and some of the software specification isn't working at all BUT got the best product of the

[Read more](#)

Add your review

1

I don't like this

Rate this product

Figure 41 product detail page 2

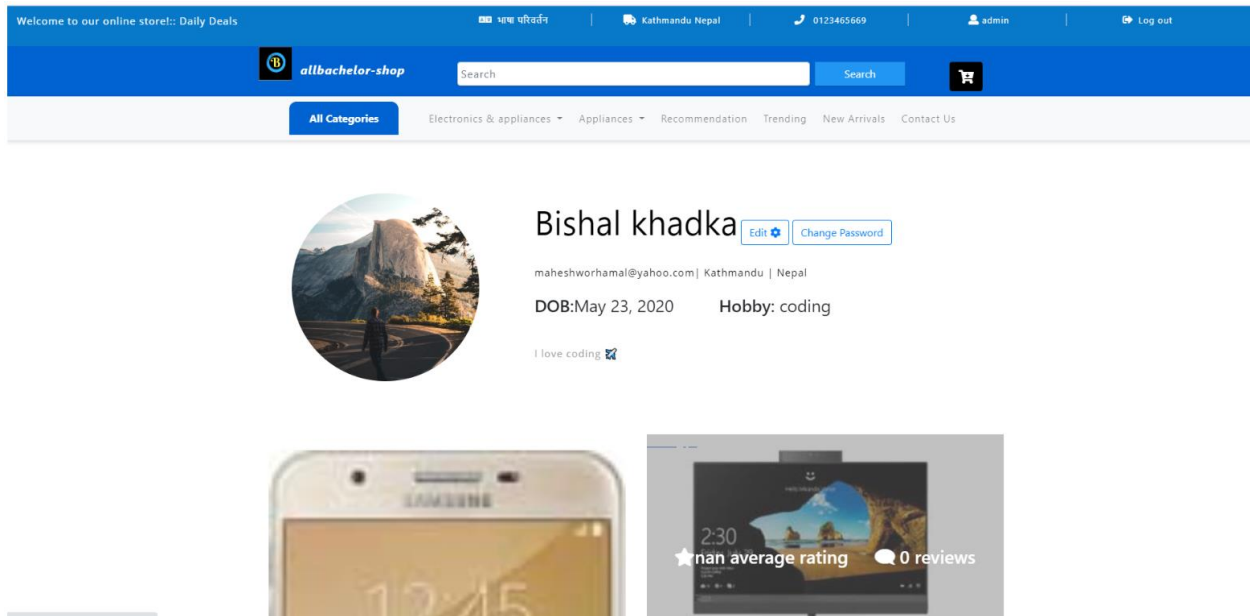


Figure 42 User profile

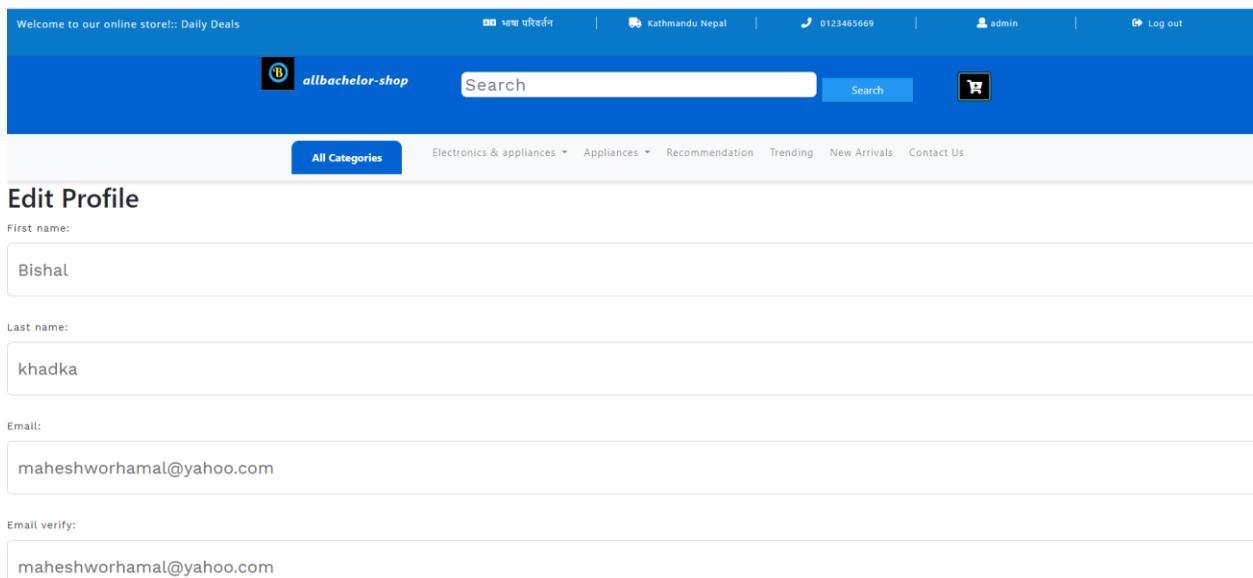


Figure 43 Edit profile

Change Password

Old password:

New password1:

Warning This password would take to crack.

- Your password can't be too similar to your other personal information
- Your password must contain at least 14 characters
- Your password can't be a commonly used password
- Your password can't be entirely numeric
- Your password must not be the same as the current password
- Your password must use of both uppercase and lowercase letters
- Your password must include of one or more numerical digits
- Your password must include of special characters, such as @, #, \$

New password2:

Warning Your passwords don't match.

Figure 44 change password

My Cart

Continue Shopping >



#QUE-007544-002
GAMING PC

1 X ~~Rs.150000~~ Rs.140000

= **RS. 140000**



#QUE-007544-002
SAMSUNGJ1

1 X ~~Rs.12000~~ Rs.11000

= **RS. 11000**

SUBTOTAL Rs. 151000

SHIPPING Free Shipping

TAX 21140.00000000000004

TOTAL 172140.0

Have A Promo Code?

Checkout >

Figure 45 My cart

My orders

ID	Full Name	Date	mobile number	city	Total	status	detail
15	Bishal khadka	May 26, 2020, 1:03 a.m.	9808349417	Kathmandu	Rs. 11000	OnShipping	Detail
16	Bishal khadka	May 26, 2020, 3:42 a.m.	9808349417	Kathmandu	Rs. 151000	New	Detail
17	Bishal khadka	May 26, 2020, 3:42 a.m.	9808349417	Kathmandu	Rs. 140000	New	Detail

Figure 46 My orders

Welcome to our online store:: Daily Deals

भाषा परिवर्तन | Kathmandu Nepal | Track My order | admin | Log out

allbachelor-shop

Search

All Categories | Electronics & appliances | Appliances | Recommendation | Trending | New Arrivals | Contact Us

Order details

Name on Card
Bishal

Surname on Card
Last Name

Address
Address

City
City

Contact Number
Telephone

Total payment: Order Total : \$172140.0

Ready to buy ?

Order Review

#QUE-007544-002
GAMING PC
1 x Rs.350000 Rs.140000 = RS. 140000

Figure 47 fill order details

Welcome to our online store:: Daily Deals

भाषा परिवर्तन | Kathmandu Nepal | Track My order | admin | Log out

allbachelor-shop

Search

All Categories | Electronics & appliances | Appliances | Recommendation | Trending | New Arrivals | Contact Us

Name	Bishal
Surname	khadka
Address	putalisadak kathmandu
City	Kathmandu
City	9808349417
Date	May 26, 2020, 1:03 a.m.
Note	This product is being delivered and on the way
Status	OnShipping

Order Review


Product	Price	Quantity	Total
 samsung1	\$ 12000	1	\$ 11000

Figure 48 order details

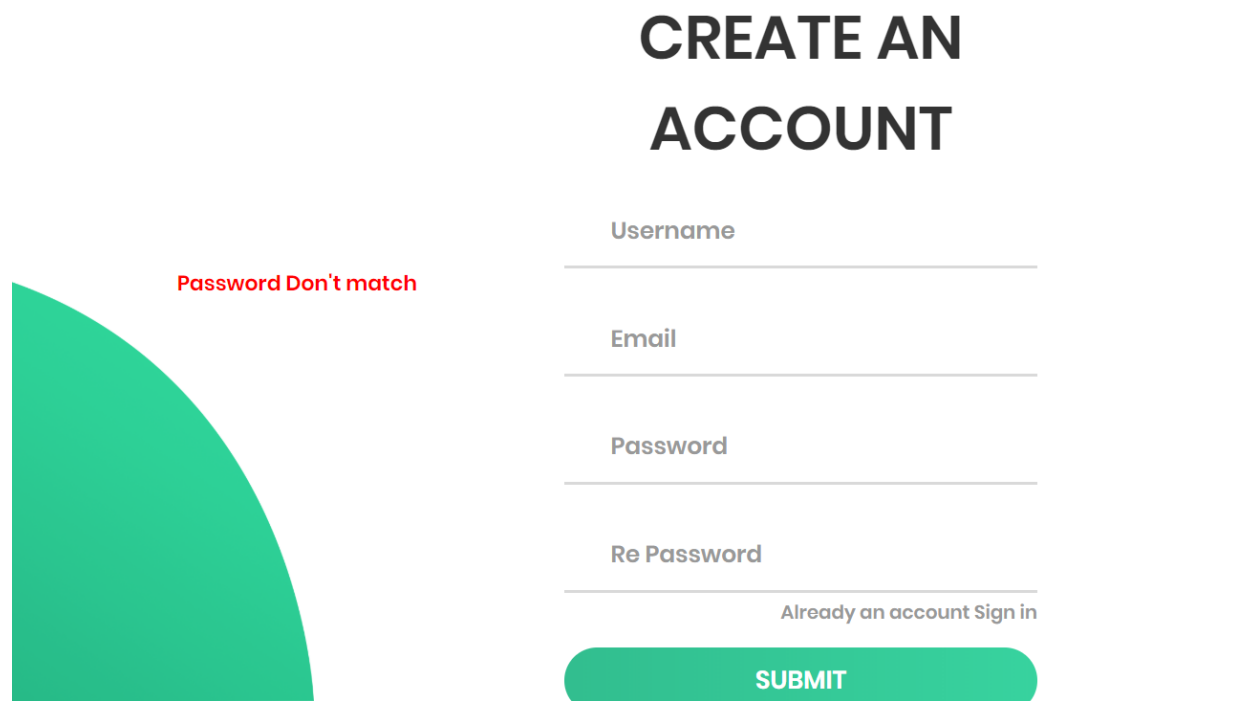
Chapter 7 Testing and Evaluation:

Different testing and evaluation are done in this section.

7.1.1 Invalid Credential:

I have used following criteria to test the system.

1. Password wrong:



The image shows a web form titled "CREATE AN ACCOUNT". The form has four input fields: "Username", "Email", "Password", and "Re Password". Below the "Re Password" field, there is a link that says "Already an account Sign in". At the bottom of the form is a green "SUBMIT" button. To the left of the form, there is a large green curved shape. Above this shape, the text "Password Don't match" is displayed in red, indicating an error message.

Figure 49 password does not match

2. If username matches with another user in database.



CREATE AN ACCOUNT

User already exist

Username

Email

Password

Re Password

Already an account [Sign in](#)

SUBMIT



Figure 50 user already exist

3. Invalid username or password.

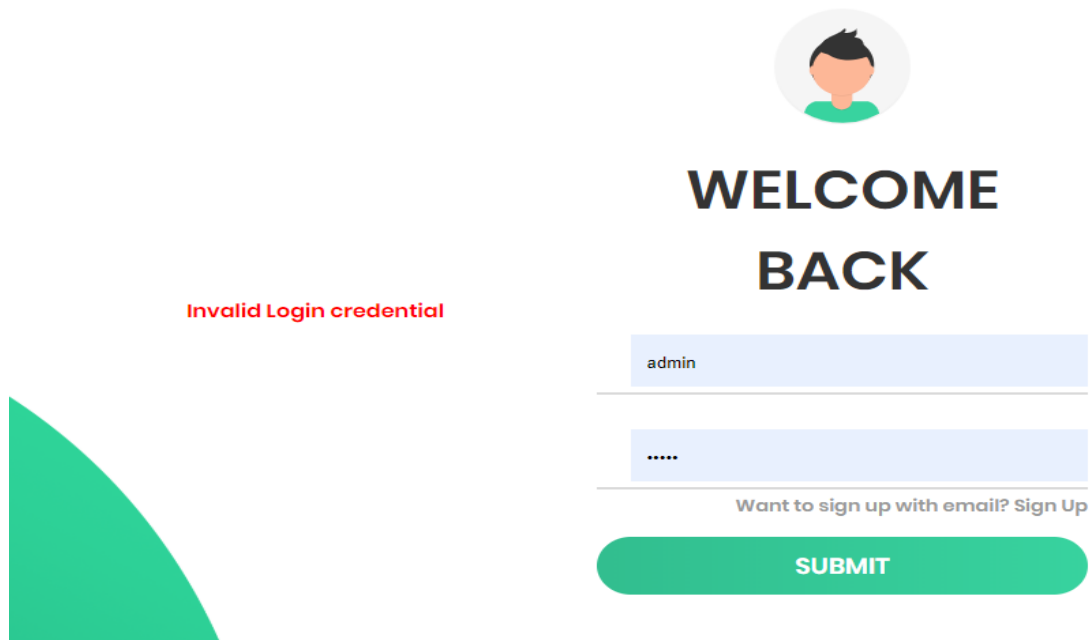


Figure 51 invalid username or password

4. The older password was entered wrong.
5. New password and confirmation password do not match.
6. Without the use of uppercase and lowercase letters.

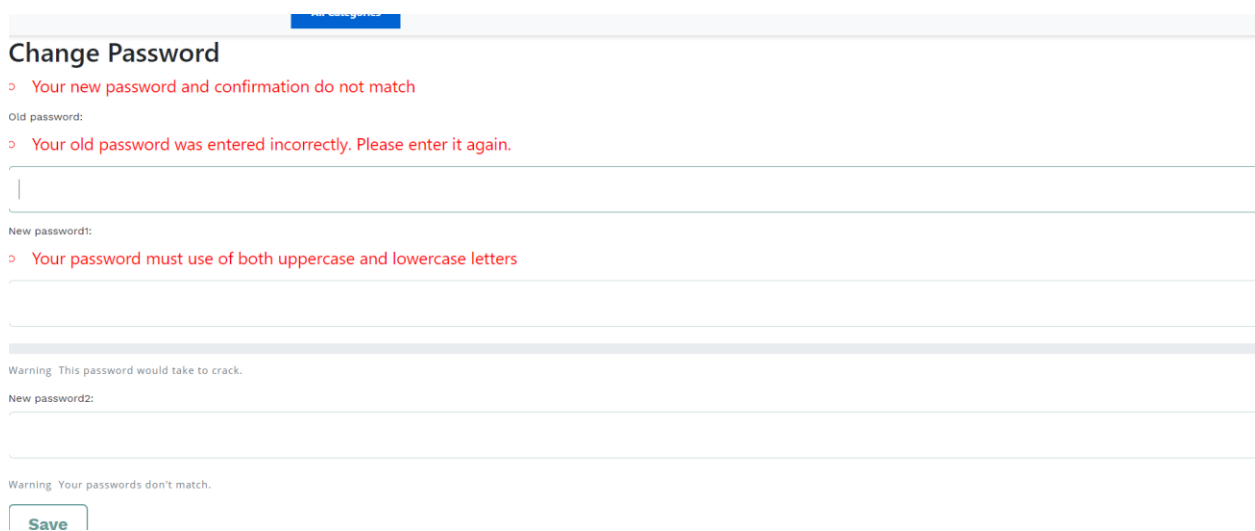


Figure 52 invalid password 1

7. password is too short and similar to other content of the user information.

All Categories Electronics & appliances ▾ Appliances ▾ Recommendation Trending New Arrivals Contact Us

Change Password

Old password:

New password1:

- This password is too short. It must contain at least 8 characters.

Warning This password would take to crack.

New password2:

- The password is too similar to the username.
- This password is too short. It must contain at least 8 characters.

Warning Your passwords don't match.

Save

Figure 53 Invalid password 2

8. The same password as used before.

Old password:

New password1:

○ Your password must not be the same as the current password

Warning This password would take to crack.

New password2:

Warning Your passwords don't match.

Figure 54 invalid password 3

9. The password is too common or the password is entirely numeric.

All Categories Electronics & appliances Appliances Recommendation Trending New Arrivals Cc

Change Password

Old password:

New password1:

- This password is too common.
- This password is entirely numeric.

Warning This password would take to crack.

New password2:

- This password is too common.
- This password is entirely numeric.

Warning Your passwords don't match.

Save

Figure 55 Invalid password 4

10. During ordering the product from product detail page, the quantity must not be greater than the stack of the product. The stock for the gaming pc is only 23 in our database.

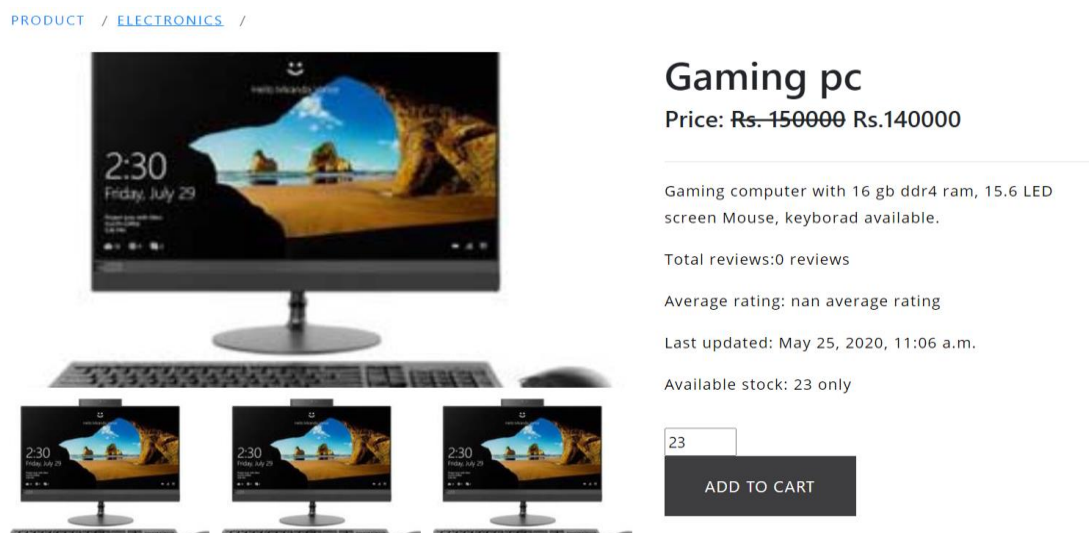


Figure 56 Quantity limit for add to cart

7.1.2. mobile friendly testing:

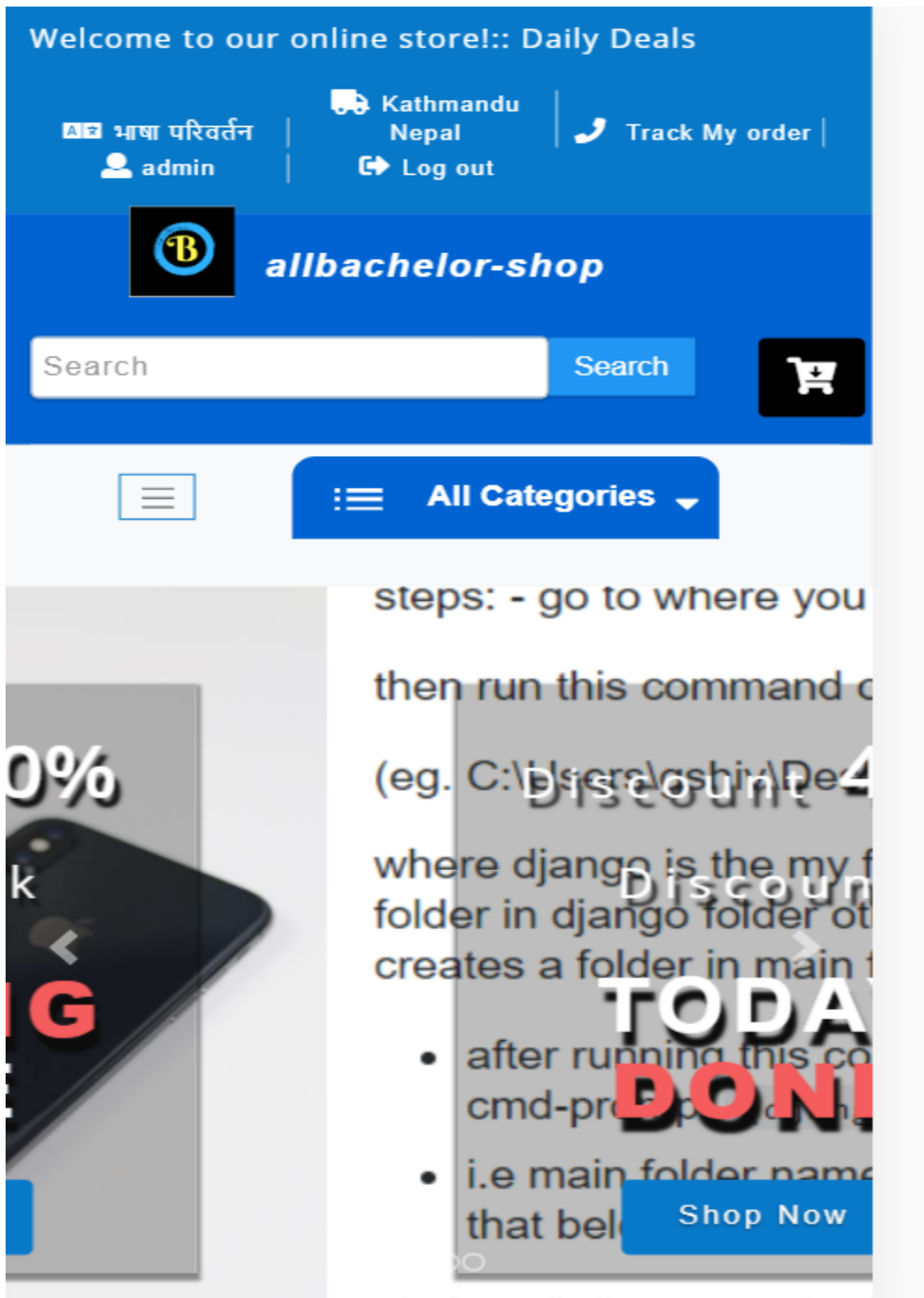


Figure 57 mobile friendly testing index page

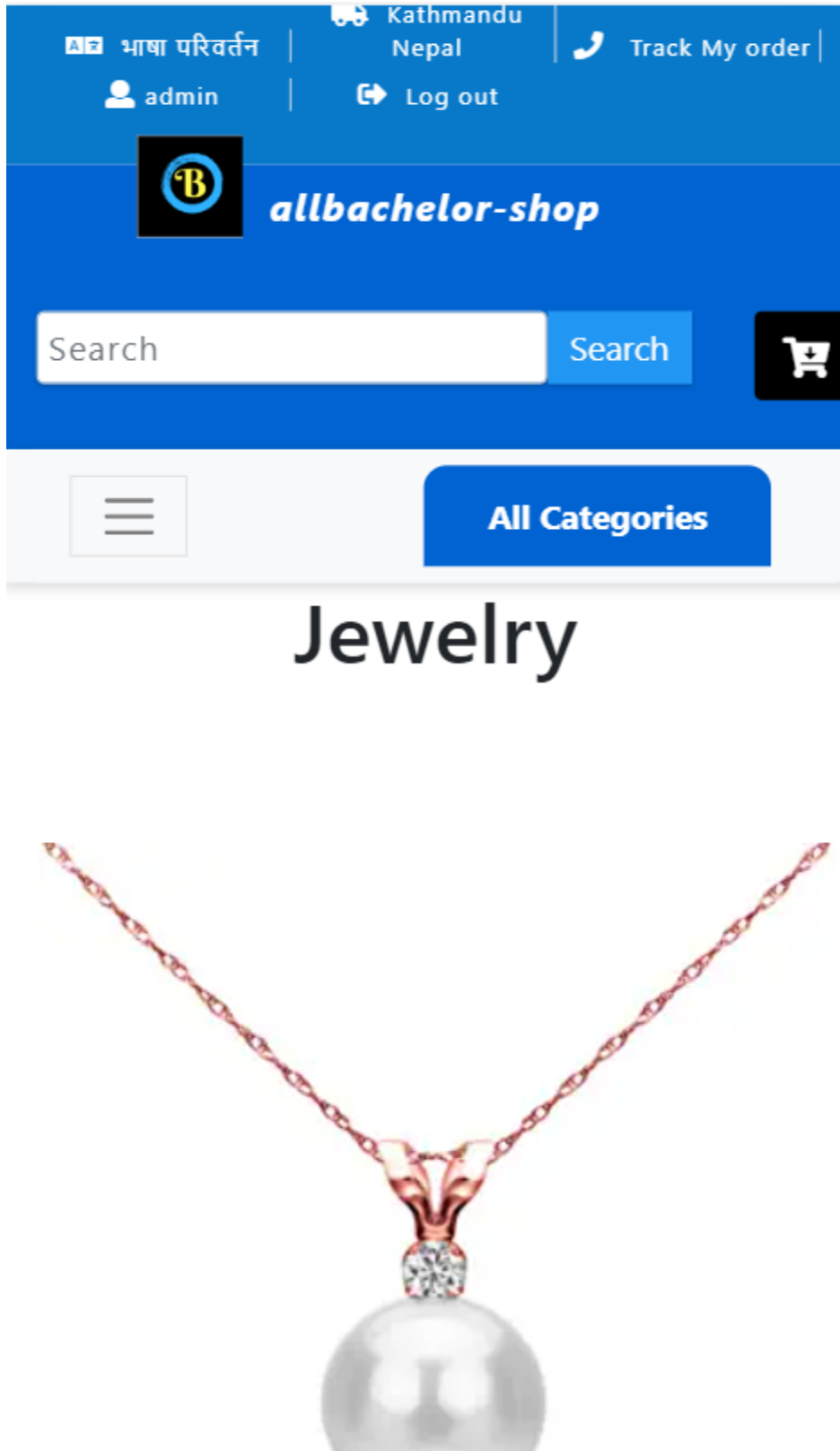


Figure 58 product detail page



WELCOME BACK

admin


.....


Want to sign up with email? [Sign Up](#)

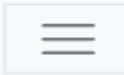
SUBMIT

Figure 59 mobile testing for login page

भाषा परिवर्तन | Nepal | Track My order |
admin | Log out


 **allbachelor-shop**

Search Search 

 [All Categories](#)



**Bishal
Khadka**

[Edit](#) 

[Change Password](#)

I love coding 

maheshworhamal@yahoo.com | Kathmandu | Nepal

DOB:
May 23, 2020


Hobby:
coding

Figure 60 profile page

☰
All Categories

My Cart

Continue Shopping >



#QUE-007544-002

UNTAMED

X ~~Rs.1200~~ Rs.1100

Have A Promo Code?

>

SUBTOTAL	Rs. 1100
SHIPPING	Free Shipping
TAX	154.00000000000000
TOTAL	1254.0

Checkout >

Figure 61 my cart

7.1.4. Testing for Recommendation system:

For this purpose, I have created three new users as admin, test and test1. I have rated at least four items and try to get the recommendation from the system. Since this recommendation system tries to find the similar user. The similar user must be clustered in database and admin must be able to view the clusters from admin panel. Each time when some user rate different items, the cluster must be updated. The cluster with zero rating and no rating at all must be placed in separate cluster.

To show the recommendation actually working I have rated some of the product from three different user's admin, test and test1. First admin(user) has rated three books including one mouse and other chain. The test1 also rated three books but two of them are different from admin. Later test user rated exactly the three books including gaming pc. By viewing this pattern, we can also say that the preferences of both admin and test look similar. On other hand test 1 user has rated completely different products than admin and test. So, the similar user for the test is admin. So, If test user clicks the recommendation link, He get the product rated by the admin but not by test user as shown I figure below.

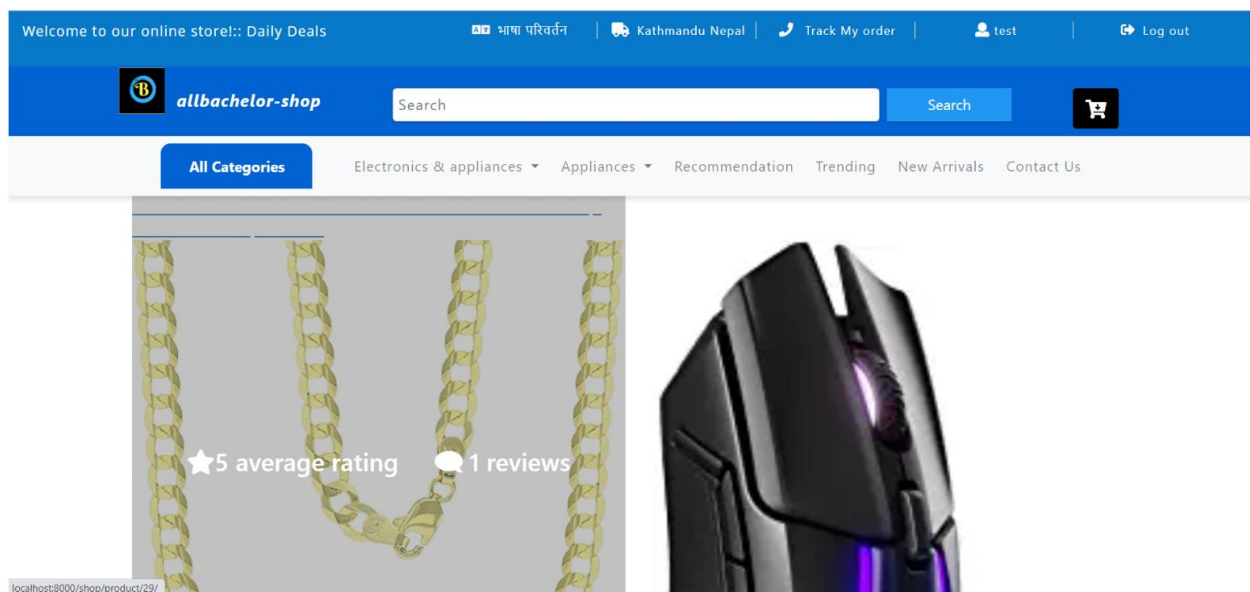


Figure 62 rating for user

Looking at the other side, as both admin and test are considered as similar user. The admin should also get the recommendation that is rated positive by test. In our case test

user has also rated positive for gaming pc. So, admin must also get the recommendation of gaming pc which is shown in figure below.

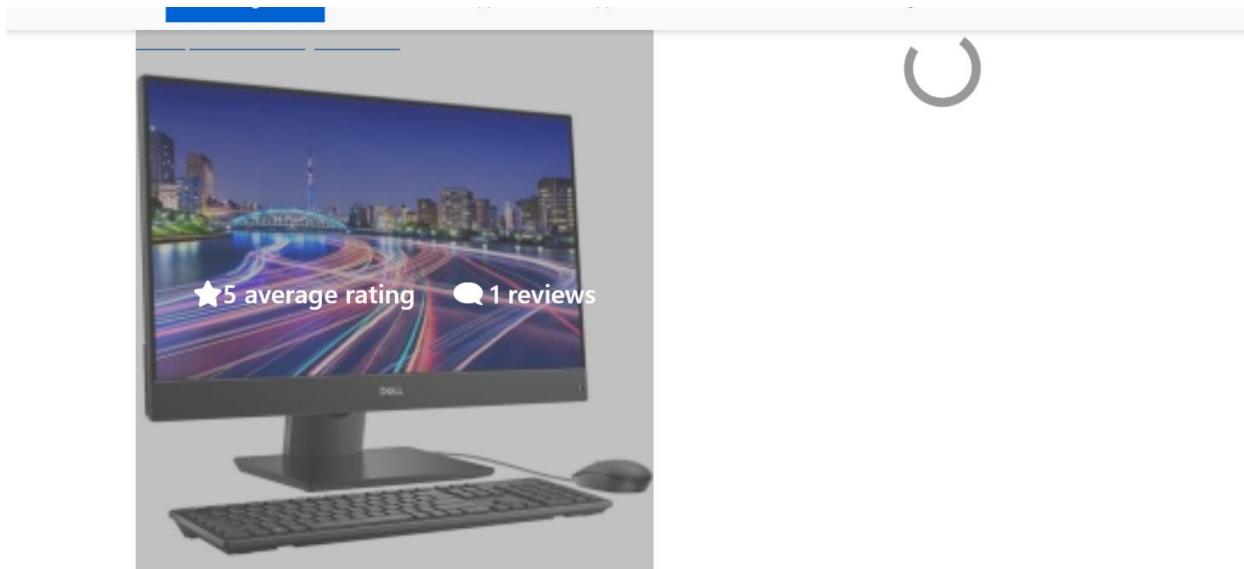


Figure 63 rating for similar user

If test user rates one of the product from the recommendation page, it must not be recommended again. To make this point valid, I have rated mouse as positive from test user.

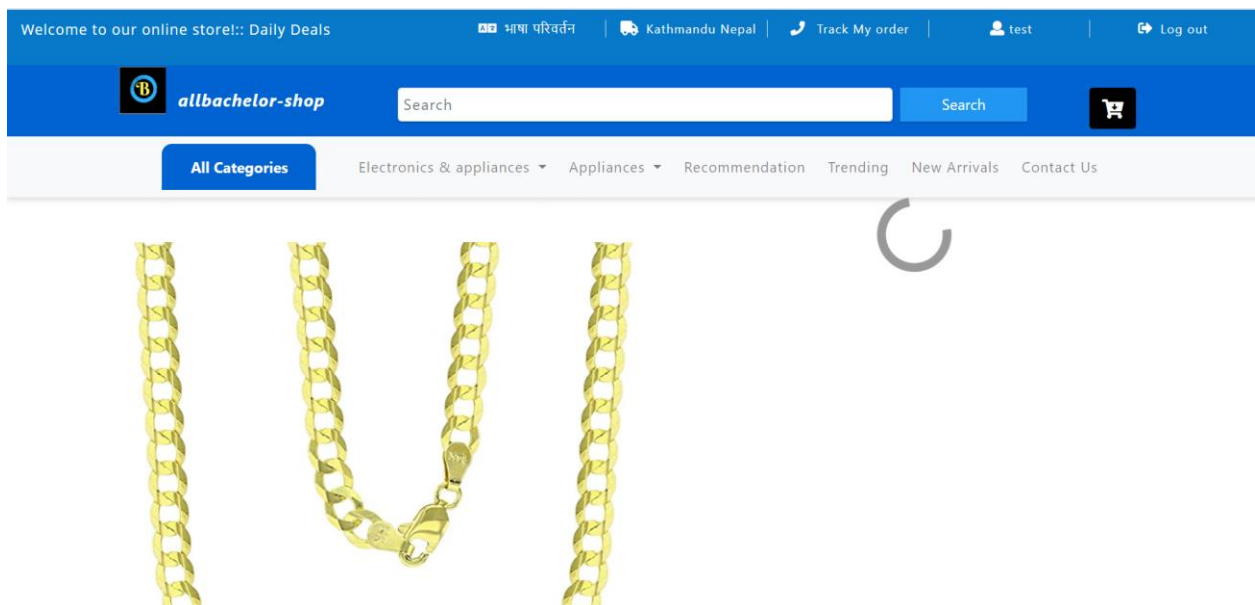


Figure 64 not showing rating to previously rated product

For another part of testing, I will try to show test and test1 as similar user. For that test1 must rate some of the product of test and test must also rate some of the product rated positive by test1. Test1 rated gaming pc as positive and test rated two of the books rated positive by test1 as positive. Our system gets confused because the system could not find is the test similar to admin or test1. For this both test and test1 must rate positive in one product that was not rated by admin. After rating a health product similar by both users. The recommendation for test1 are shown as shown in figure below. These all products are rated positive by test user.

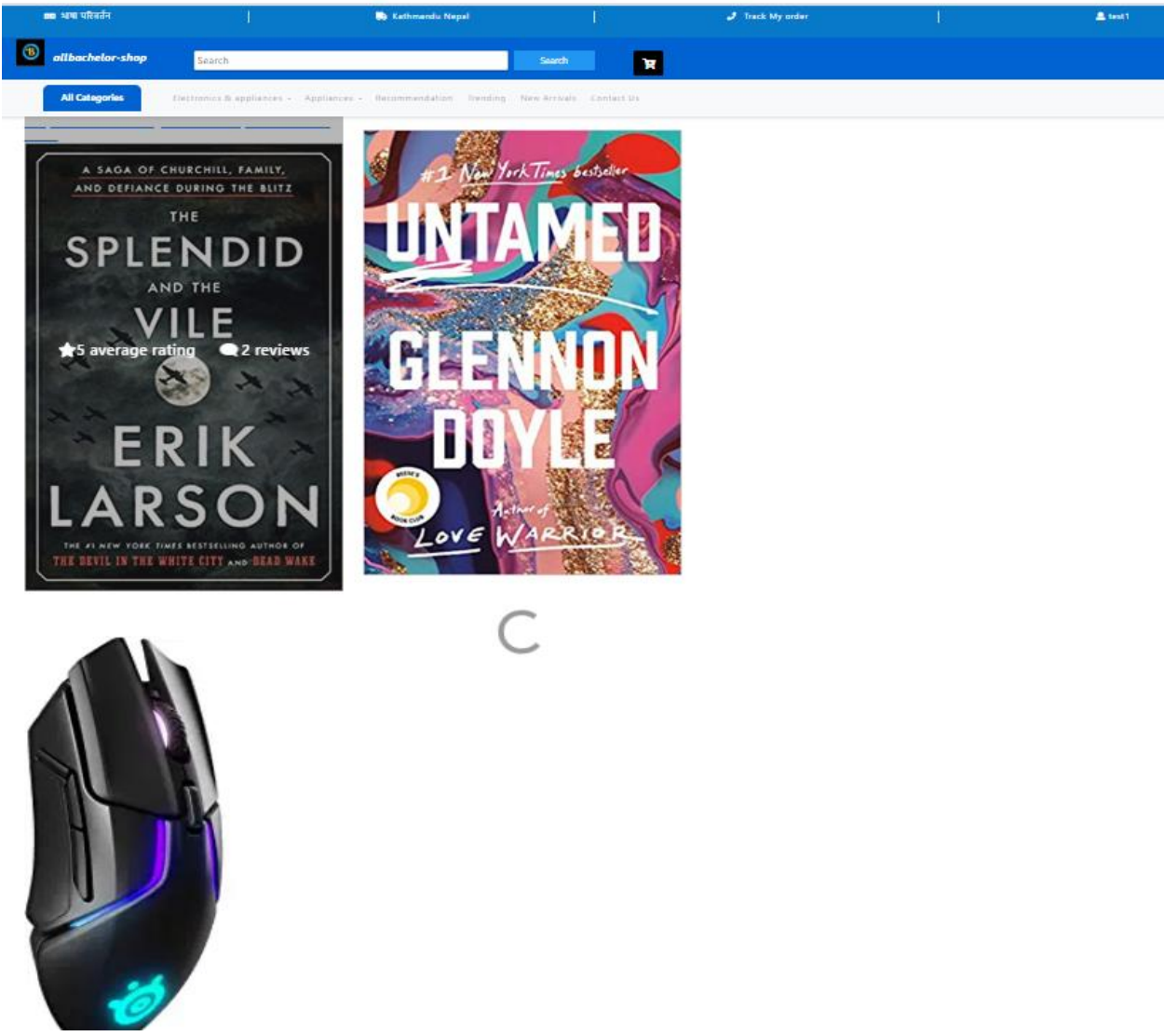


Figure 65 recommendation for test1

This algorithm will work efficiently if different user has rated different product. The updated information can also be seen from admin side. As shown in figure below:

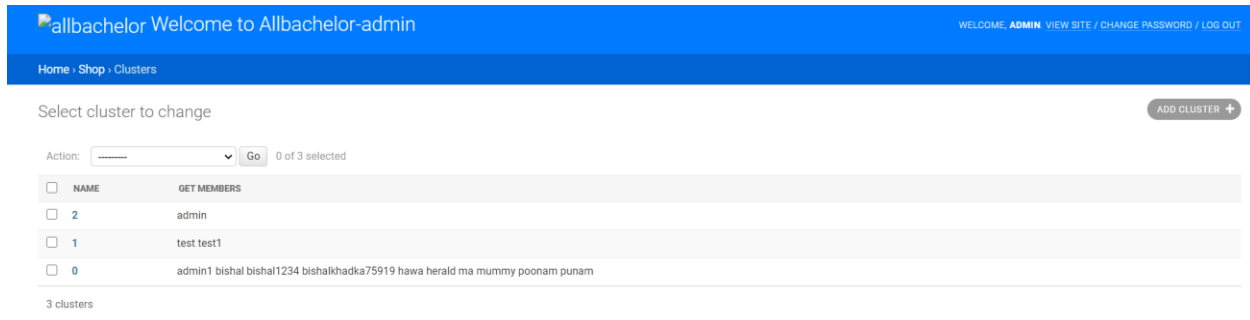


Figure 66 admin viewing clusters

7.2 Academic question:

The main motto behind this project is to find out that how to recommend product based upon the user's preference or interest. To recommend products to the user, rating component is chosen. After finding the preference of the user for the particular product, it's time to predict whether the user may like other products or not based upon the past rating s/he has given. So, different techniques are used to find the similar user and their distance which is discussed in critical evaluation part. I used latent based model for recommending products to user. So, the load time for recommending product is pretty fast as compared to user distance finding techniques like Euclidean distance, Jaccard similarity and Pearson correlation. Apart from that clusters are also made based upon the user in the system. The total number of clusters is calculated by using the total number of users. The clusters is calculated by dividing number of users in the system divided by two and adding 1 So that the algorithm works for small number of users also. The cluster are then updated to the database. I tried to increase the performance of the system by adding k- means clustering to find the number of clusters. The system can be extended. Apart from this, this system also uses popularity-based recommendation system which is not personalized recommendation and tfidf based recommendation system which is a content-based recommendation is also made.

This system has some of the issues and real-world challenges which are described in critical evaluation chapter below.

Chapter 8 conclusion and future work:

In this final semester, a full-fledged ecommerce system with different products from different categories and powerful recommendation system is integrated. I used user clustering technique which boost the performance of the system. The system can highly be improved.

Some of the work for future work is to extend the search features and use different filters to the system. Trending list can be generated based upon the popularity of the product in certain period of time in certain area. To increase the sell, flash sell and promo code for certain discount must be integrated. Password strength checker can also be integrated for better security. The option for language can be increased in future to sell more product to the native people of different community. There are many possibilities for the development of the system.

Just like other software, this software also has a lot of unseen bug. I will regularly fix and upgrade the software during maintenance and deployment of the system in real life.

8.1 Critical evaluation:

Recommendation systems are the hot topic in online big tech. Constantly, recommendation systems are changing to user's behavior in the system. The recommendation systems are of different types as described in literature review part. The distance between similar user can be calculated by using different approach as shown in Chapter 2. I tried some of the approach for calculating distance. I used latent factor model for recommendation purpose. The matrix is used with user's and their particular rating in product in this system. Most of the ecommerce site uses rating as one of the main components to get the user's preferences on particular topic. So, I did the same thing. I used the rating as the main component for determining or predicting the rating of the particular user. Apart of rating, other implicit features such as like, comment, purchase record, purchase feedback etc. can also be used to recommend products more accurately.

The load time for loading the recommendation is pretty good. To make the system even faster caching can be used. Further enhancements to this system would be to integrate

deep learning and hybrid approach for recommending the product. Natural language processing can also be used in comment section of product to detect the user's preferences on certain products.

This system does not have function of editing the review and cannot even reply to the comment. To work recommendation part of this system, the system must not delete any product and users from database since, foreign key of product i.e. product_id and foreign key of user i.e. user_id is used in separate review table of database. Like and dislike system can also be used instead of rating system. Recommendation can also be given to message section. Message replying can also be automated. Chatbot and blog section are one of the main components of ecommerce which can also be integrated.

References

- Shao, H. et al., 2020. *paper2repo: GitHub Repository Recommendation for Academic papers*, Taipei, Taiwan: Amazon Web Services Deep Learning.
- aggrawal, P., Tomar, V. & Kathuria, A., 2017. Comparing Content Based and Collaborative. *International Journal of New Technology and Research (IJNTR)*, 3(4), pp. 65-67.
- Anantha, L. & Bhattula, B. P., 2017. A REVIEW ON RECOMMENDATION SYSTEM USING RATING DATASET. *International Journal of Pure and Applied Mathematics*, 116(5), pp. 133-135.
- Beel, J., Breitinger, C. & Langer, S., 2017. *Evaluating the CC-IDF citation-weighting scheme: How effectively can 'Inverse Document Frequency' (IDF) be applied to references?*, s.l.: iConference.
- Chnag, S., Harper, F. M. & Terveen, L., n.d. *Crowd-Based Personalized Natural Language Explanation for recommendation*, Minnesota: The GroupLens Center for Social and Human-Centered Computing.
- Desrosiers, C. & Karypis, G., 2017. *A comprehensive survey of neighborhood-based*, s.l.: s.n.
- Guru99, 2020. *What is BIG DATA? Introduction, Types, Characteristics & Example*. [Online] Available at: <https://www.guru99.com/what-is-big-data.html#2> [Accessed 07 May 2020].
- HERLOCKER, J., KONSTAN, J., TERVEEN, L. & RIEDL, J., 2004. Evaluating Collaborative Filtering Recommender systems. *ACM Transactions on Information System*, 22(1), pp. 5-10.
- Jiang, S., Li, K., Yi, R. & Xu, D., 2018. Magnitude Bounded Matrix Factorisation for Recommender Systems. *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS*, pp. 1-10.
- Jingying, Z., 2017. *Latent Factor Models for Recommender Systems and Market Segmentation Through Clustering*, Ohio : The Ohio State University.
- Kang, J. et al., 2017. *Understanding How People Use Natural Language to Ask for Recommendations*, s.l.: University of Minnesota.
- Katarya, R. & verma, O. P., 2016. *A collaborative recommender system enhanced with particle swarm optimization technique*, New York: Springer Science+Business Media.
- Kim, T.-H. & Yang, S.-B., 2015. *An Effective Recommendation Algorithm for Clustering-Based Recommender Systems*, Shinchon-dong, Seodaemun-gu, Seoul: Dept. of Computer Science, Yonsei University.
- Krishnan, N. & Brahma, E., n.d. *Survey of Neighborhood-based Collaborative Filtering Techniques for a movie recommendation engine*, Baltimore, MD 21218, USA: Johns Hopkins University.
- Lee, Y., 2015. *RECOMMENDATION SYSTEM USING COLLABORATIVE FILTERING*, s.l.: San Jose State University.
- Liu, H. et al., 2014. A new user similarity model to improve the accuracy of collaborative filtering. *State Key Laboratory of Networking and Switching Technology*, pp. 156-166.

Melville, P. & Sindwani, V., 2018. *Recommender systems*, Yorktown Heights: IBM T.J. Watson Research Center.

Park, M.-H., Hong, J.-H. & Cho, S.-B., 2007. *Location-Based Recommendation System Using Bayesian User's Preference Model in Mobile Devices*, Shinchon-dong, Seodaemun-ku, Seoul Korea: Dept. of Computer Science, Yonsei University .

Pazzani, M. J. & Billsus, D., 2015. *Content-based Recommendation Systems*, Palo Alto, CA: Palo Alto Laboratory, Inc.

Ricci, F., Rokach, L., Shapira, B. & Kantor, P. B., 2011. *Recommender Systems Handbook*. Springer New York Dordrecht Heidelberg London: Springer.

Shardanand, U. & Maes, P., 1995. *Social Information Filtering: Algorithms for Automating "Word of Mouth"*, Cambridge, MA: MIT Media-Lab.

Sharma, S. & Mahajan, A., 2017. A Collaborative Filtering Recommender System for Github. *International Journal of Emerging Technologies in Engineering Research (IJETER)*, 5(8).

Sitikhu, P., Pahi, K., Thapa, P. & shakya, s., 2019. *A Comparison of Semantic Similarity Methods for Maximum Human Interpretability*, Pulchowk Lalitpur: Pulchowk Campus, IOE, Nepal.

Sivaramakrishnan, N. et al., 2018. NEIGHBORHOOD-BASED APPROACH OF COLLABORATIVE FILTERING TECHNIQUES FOR BOOK RECOMMENDATION SYSTEM. *International Journal of Pure and Applied Mathematics Academic Publishing Ltd,,* pp. 13241-13250.

Sondur, S. D. & Chigadani, A. P., 2016. Similarity Measures for Recommender Systems:A Comparative Study. *Journal for Research*, 2(3), pp. 76-80.

statista, 2020. *eCommerce*. [Online]

Available at: <https://www.statista.com/outlook/243/100/ecommerce/worldwide#market-globalRevenue>

[Accessed 20 5 2020].

Wang, J. & Qiang, T., 2017. *A Probabilistic View of Neighborhood-based Recommendation Methods*, s.l.: University of Luxembourg.

Zhou, G. et al., 2019. *Deep Interest Evolution Network for Click-Through Rate Prediction*, Beijing,China: Alibaba Inc..

Zhou, G. et al., 2018. *Deep Interest Network for Click-Through Rate Prediction*, s.l.: Ali baba group.