BUILDING A HOTEL RECOMMENDATION SYSTEM USING MACHINE LEARNING

Submitted for partial fulfillment of the requirements

for the award of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

by

Bonkuri Bhavani Prasad	**--**	**16BQ1A0523**
Bathuri Ayyappa	**--**	**16BQ1A0516**
Gadiparthi Venkatesh	**--**	**16BQ1A0552**
Anangi Krishna Teja	**--**	**16BQ1A0504**

Under the guidance of

Mr. R. Chitti Babu

Assist. Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(B.Tech Program is Accredited by NBA)

VASIREDDY VENKATADRI INSTITUTE OF TECHNOLOGY

Permanently Affiliated to JNTU Kakinada, Approved by AICTE

Accredited by NAAC with ‘A’ Grade, ISO 9001:2008 Certified

NAMBUR(V), PEDAKAKANI(M), GUNTUR-522 508

Tel no: 0863-2118036, url:www.vvitguntur.com,

April 2020.

DECLARATION

We, Mr. B. Bhavani Prasad, Mr. B. Ayyappa, Mr. G.Venkatesh, Mr. A. Krishna Tejahereby declare that the Project Report entitled “Building A Hotel Recommendation System Using Machine Learning” done by us under the guidance of Mr. R. Chitti Babu, Associate Professor, CSE at Vasireddy Venkatadri Institute of Technology is submitted for partial fulfillment of the requirements for the award of Bachelor of Technology in Computer Science & Engineering. The results embodied in this report have not been submitted to any other University for the award of any degree.

DATE :

PLACE :

SIGNATURE OF THE CANDIDATE(S)

(B. Bhavani Prasad)

(B. Ayyappa)

(G.Venkatesh)

(A.Krishna Teja)

VASIREDDY VENKATADRI INSTITUTE OF TECHNOLOGY
Permanently Affiliated to JNTU Kakinada, Approved by AICTE
Accredited by NAAC with 'A' Grade, ISO 9001:2008 Certified
Nambur, Pedakakani (M), Guntur (Dt) - 522508
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
B.Tech Program is Accredited by NBA

CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Mr. B. Bhavani Prasad, Mr. B. Ayyappa, Mr. G. Venkatesh, Mr. A.. Krishna Teja** bearing **Reg. No. 16BQ1A0523, 16BQ1A0516, 16BQ1A0552, 16BQ1A0504** who had carried out the project entitled "Building Hotel Recommendation System Using Machine Learning" under our supervision.

Project Guide	**Head of the Department**
(R. Chitti Babu)	(Dr. R. Eswaraiah)

Submitted for Viva voce Examination held on ________________________

External Examiner

ACKNOWLEDGEMENT

I take this opportunity to express my deepest gratitude and appreciation to all those people who made this project work easier with words of encouragement, motivation, discipline, and faith by offering different places to look to expand my ideas and helped me towards the successful completion of this project work.

First and foremost, I express my deep gratitude to **Mr. Vasireddy Vidya Sagar**, Chairman, Vasireddy Venkatadri Institute of Technology for providing necessary facilities throughout the B.Tech programme.

I express my sincere thanks to **Dr. Y. Mallikarjuna Reddy**, Principal, Vasireddy Venkatadri Institute of Technology for his constant support and cooperation throughout the B.Tech programme.

I express my sincere gratitude to **Dr. R. Eswarariah**, Professor & HOD, Computer Science & Engineering, Vasireddy Venkatadri Institute of Technology for his constant encouragement, motivation and faith by offering different places to look to expand my ideas.

I would like to express my sincere gratefulness to my guide **Mr. R. Chitti babu** for his insightful advice, motivating suggestions, invaluable guidance, help and support in successful completion of this project.

I would like to take this opportunity to express my thanks to the **teaching and non-teaching** staff in Department of Computer Science & Engineering, VVIT for their invaluable help and support.

Bonkuri Bhavani Prasad (16BQ1A0523)

Bathuri Ayyappa (16BQ1A0516)

Gadiparthi Venkatesh(16BQ1A0552)

Anangi Krishna Teja (16BQ1A0504)

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

In this project, we aim to create the optimal hotel recommendations for users that are searching for a hotel to book. We will model this problem as a multi-class classification problem and train the model using algorithms Like SVM,K-Nearest Neighbour,Naivebayes,LogisticRegression is ensemble method to predict which hotel the user is likely to book, given his (or her) entered details.

Keywords:
numpy, pandas, matplotlib, python-flask, scikit-learn, seaborn packages.

CHAPTER-1
INTRODUCTION

We would like to recommend hotels based on the hotels that a user has already booked or viewed using the cosine similarity. We would recommend hotels with the largest similarity to the ones previously booked or viewed or showed interest by the user. Our recommendation system is highly dependent on defining an appropriate similarity measure. Eventually, we select a subset of hotels to display to the user or to determine an order in which to display the hotels.

One of the first things to do while planning a trip is to book a good place to stay. Booking a hotel online can be an overwhelming task with thousands of hotels to choose from, for every destination. Motivated by the importance of these situations, we decided to work on the task of recommending hotels to users. We used Expedia's hotel recommendation dataset, which has a variety of features that helped us achieve a deep understanding of the process that makes a user choose certain hotels over others. The aim of this hotel recommendation task is to predict and recommend five hotel clusters to a user that he/she is more likely to book given hundred distinct clusters.

1.1 MACHINE LEARNING

Machine learning (ML) is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task.Machine learning algorithms are used in a wide variety of applications, such as filtering and computer vision, where it is difficult or infeasible to develop a conventional algorithm for effectively performing the task.

Machine learning is closely related to computational statistics, which focuses on making predictions using computers. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a field of study within machine learning, and focuses on exploratary data analysis through unsupervised Learning. In its application across business problems, machine learning is also referred to as predictive analytics.

1.1.1 Machine Learning tasks:

A **machine learning task** is the type of prediction or inference being made, based on the problem or question that is being asked, and the available data. For example, the classification task assigns data to categories, and the clustering task groups data according to similarity.

Machine learning tasks are classified into several broad categories.InSupervised learning, the algorithm builds a mathematical model from a set of data that contains both the inputs and the desired outputs.In unsupervised learning, the algorithm builds a mathematical model from a set of data that contains only inputs and no desired output labels. Unsupervised learning algorithms are used to find structure in the data, like grouping or clustering of data points.

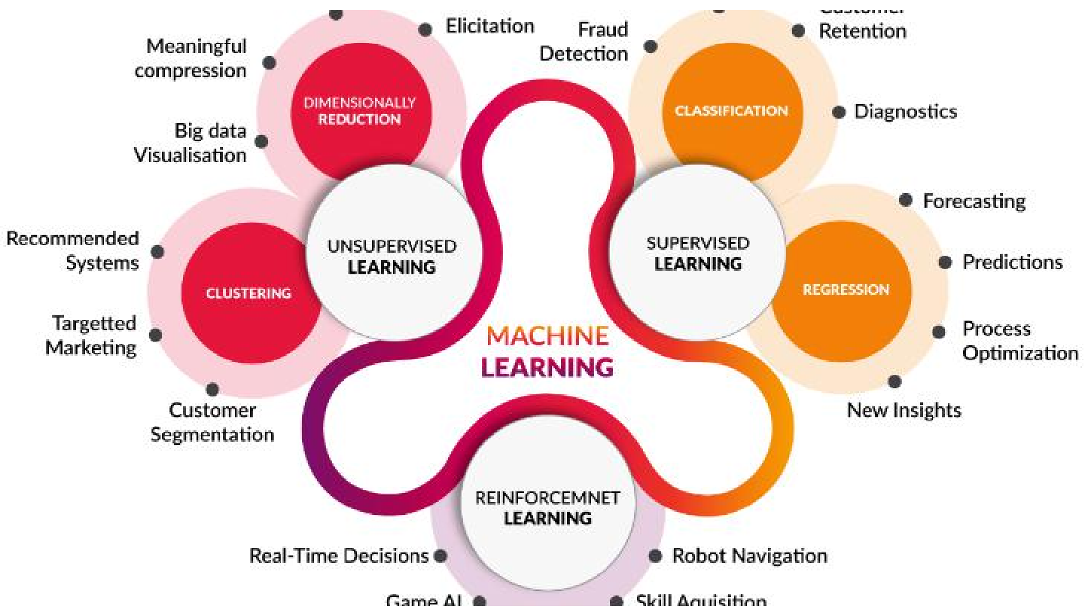


Fig.1.1: Machine Learning Tasks

In supervised Learning, the algorithm builds a mathematical model from a set of data that contains both the inputs and the desired outputs. For example, if the task were determining whether an image contained a certain object, the training data for a supervised learning algorithm would include images with and without that object (the input), and each image would have a label (the output) designating whether it contained the object. In special cases, the input may be only partially available, or restricted to special feedback.

In unsupervised learning, the algorithm builds a mathematical model from a set of data that contains only inputs and no desired output labels. Unsupervised learning algorithms are used to find structure in the data, like grouping or clustering of data points. Unsupervised learning can discover patterns in the data, and can group the inputs into categories, as in feature learning. Dimensionality reduction is the process of reducing the number of "features", or inputs, in a set of data.

Reinforcement Learning algorithms are given feedback in the form of positive or negative reinforcement in a dynamic environment and are used in autonomous learning or in learning to play a game against a human opponent.Other specialized algorithms in machine learning include topic modeling, where the computer program is given a set of natural language documents and finds other documents that cover similar topics.

Applications of Machine Learning:

- Traffic Alerts
- Social Media
- Transportation and commuting
- Products Recommendations
- Virtual personal Assistants
- Self Driving Cars
- Dynamic Pricing
- Google Translate
- Online Video Streaming
- Fraud Detection

Machine Learning Workflow:

Machine learning in production happens in five phases. (There are few standardized best practices across teams and companies in the industry. Most machine-learning systems are ad hoc.)

Phases in Machine Learning Workflows

- Use Case Conception and Formulation
- Feasibility Study and Exploratory Analysis
- Model Design, Training, and Offline Evaluation
- Model Deployment, Online Evaluation, and Monitoring
- Model Maintenance, Diagnosis, and Retraining

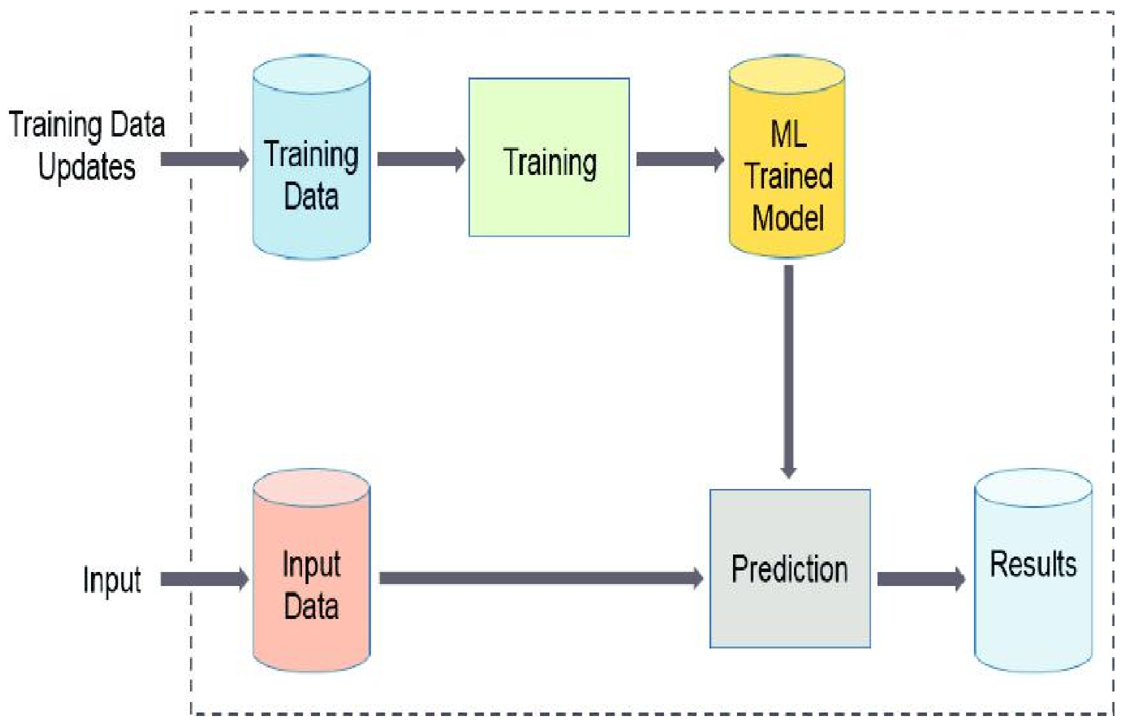


Fig.1.2.Machine Learning Workflow

CHAPTER-2
AIM AND SCOPE

2.1 EXISTING SYSTEM :

Most of the existing hotel recommendation systems are designed using Naive bayes algorithm. This algorithm is used for recommended the best hotel. By using this algorithm the output may not be accurate and correct.

Naive Bayes Algorithm :

It is a recommendation technique based on Bayes theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as 'Naive'.

Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods. Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:

$$\underset{\text{Posterior Probability}}{P(c \mid x)} = \frac{\overset{\text{Likelihood}}{P(x \mid c)}\,\overset{\text{Class Prior Probability}}{P(c)}}{\underset{\text{Predictor Prior Probability}}{P(x)}}$$

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c$$

● P(c|x) is the posterior probability of class (c, target) given predictor (x, attributes).

● P(c) is the prior probability of class.

● P(x|c) is the likelihood which is the probability of predictor given class.

● P(x) is the prior probability of predictor.

Advantages :

- It is easy and fast to predict class of test data set It also perform well in multi class Prediction
- When assumption of independence holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need less training data.
- It performs well in case of categorical input variables compared to numerical variable(s). For numerical variable, normal distribution is assumed (bell curve, which is a strong assumption).

Disadvantages :

- If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as "Zero Frequency". To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.
- On the other side naive Bayes is also known as a bad estimator, so the probability outputs from predict proba are not to be taken too seriously.
- Another limitation of Naive Bayes is the assumption of independent predictors. In real life,

Some of the existing hotel recommended system are

1. **Hotel Recommendations Using Machine Learning:**

Expedia users who prefer the same types of hotels presumably share other commonalities (i.e., non-hotel commonalities) with each other. With this in mind, Kaggle challenged developers to recommend hotels to Expedia users. Armed with a training set containing data about 37 million Expedia users, we set out to do just that. Our machine-learning algorithms ranged from direct applications of material learned in class to multi-part algorithms with novel combinations of recommender system techniques. Kaggle's benchmark for randomly guessing a user's hotel cluster is 0.02260, and the mean average precision K = 5 value for na¨ıve recommender systems is 0.05949. Our best combination of machine-learning algorithms achieved a fifigure just over 0.30. Our results provide insight into performing multi-class classifification on data sets that lack linear structure.

2.2 PROPOSED SYSTEM :

Hotel recommended system are different kind of all the web sites, Our project is recommended the best hotel to know the requirements of user. This works very accurately than compared to other hotel recommended system. As soon as we enter the information it redirects to the website that they want. In our project we use HTML language, python(flask) by using these techinques and features provide the hotel recommended system works very accurately.

2.3 FEASIBILITY STUDY:

The feasibility of the project is analyzed in this phase and the proposal is put forth with a general plan for the project and some cost estimates. During System analysis the feasibility study of the proposed system to be carried out. This is to ensure that the proposed system is not a burden to already existing system. For feasibility analysis, some understanding of the major requirements for the system essential.

Three key considerations involved in feasibility are:

1.Technical feasibility

2.Operational feasibility

3.Economic feasibility

2.3.1 Technical feasibility :

Technical feasibility study is the complete study of the project in terms of input, processes, output, fields, programs and procedures. It is a very effective tool for long term planning and troubleshooting. The technical feasibility study should most essentially support the financial information of an organization.

After knowing the features the hotel recommendation system should support, technical requirements can be extracted and appropriate technology chosen for the implementation.

2.3.2 Operational feasibility:

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. It takes care that the management and the users support the project.

2.3.3 Economic feasibility:

Generally, it means whether a business or a project feasible cost wise and logistically. Economics calculate economic feasibility by analyzing the costs and revenues a business would incur by undertaking a certain project. There is no need for user to credit the money for accessing the website the return for investment can be calculated by the number of times website is used.

CHAPTER-3

CONCEPTS AND METHODS

3.1 PROBLEM DESCRIPTION:

Now-a-days people are habituated to modern technologies which gives the results with in seconds.Due to time constraints and their busy schedule they are willing to complete all the tasks within a short span of time, but searching for different hotels in different websites is time consuming and is of long process.The mindset of people will be like their work should be completed with less effort and the result must be with in seconds.

3.2 PROBLEM SOLUTION :

To overcome the above problem we creating an efficient web interface which interact with the user in friendly manner. In this the user can search for the hotel which they want to stay. when the user enters the requirements then it displays the best hotels to stay. The proposed system will give the best results with in seconds. The proposed system uses the machine learning algorithms(SVM), and python flask.

3.2.1 ALGORITHMS:

1.K-Nearest Neighbours(KNN):

K-nearest neighbors (KNN) algorithm is a type of supervised ML algorithm which can be used for both classification as well as regression predictive problems. However, it is mainly used for classification predictive problems in industry. The following two properties would define KNN well −

1.Lazy learning algorithm − KNN is a lazy learning algorithm because it does not have a specialized training phase and uses all the data for training while classification.

2.Non-parametric learning algorithm − KNN is also a non-parametric learning algorithm because it doesn't assume anything about the underlying data.

Working of KNN Algorithm

K-nearest neighbors (KNN) algorithm uses 'feature similarity' to predict the values of new datapoints which further means that the new data point will be assigned a value based on how closely it matches the points in the training set. We can understand its working with the help of following steps −

Step 1 − For implementing any algorithm, we need dataset. So during the first step of KNN, we must load the training as well as test data.

Step 2 − Next, we need to choose the value of K i.e. the nearest data points. K can be any integer.

Step 3 − For each point in the test data do the following −

3.1--Calculate the distance between test data and each row oftraining datawith the help of any of the method namely: Euclidean.

3.2 − Now, based on the distance value, sort them in ascending order.

3.3 − Next, it will choose the top K rows from the sorted array.

3.4 − Now, it will assign a class to the test point based on mostfrequent class of these rows.

Step 4 − End

2.Logistic Regression:

Logistic regression is a supervised learning classification algorithm used topredict the probability of a target variable. The nature of target or dependent variable is dichotomous, which means there would be only two possible classes.

In simple words, the dependent variable is binary in nature having data coded as either 1 (stands for success/yes) or 0 (stands for failure/no).

Mathematically, a logistic regression model predicts P(Y=1) as a function of X. It is one of the simplest ML algorithms that can be used for various classification problems such as spam detection, Diabetes prediction, cancer detection etc.

Types of Logistic Regression

Generally, logistic regression means binary logistic regression having binary target variables, but there can be two more categories of target variables that can be predicted by it. Based on those number of categories, Logistic regression can be divided into following types

Binary or Binomial

In such a kind of classification, a dependent variable will have only two possible types either 1 and 0. For example, these variables may represent success or failure, yes or no, win or loss etc.

Multinomial

In such a kind of classification, dependent variable can have 3 or more possible **unordered** types or the types having no quantitative significance. For example, these variables may represent “Type A” or “Type B” or “Type C”.

Ordinal

In such a kind of classification, dependent variable can have 3 or more possible **ordered** types or the types having a quantitative significance. For example, these variables may represent “poor” or “good”, “very good”, “Excellent” and each category can have the scores like 0,1,2,3.

We can call a Logistic Regression a Linear Regression model but the Logistic Regression uses a more complex cost function, this cost function can be defined as the ‘**Sigmoid function**’ or also known as the ‘logistic function’ instead of a linear function.

The hypothesis of logistic regression tends it to limit the cost function between 0 and 1. Therefore linear functions fail to represent it as it can have a value greater than 1 or less than 0 which is not possible as per the hypothesis of logistic regression.

$$0 \leq h_\theta(x) \leq 1$$

Sigmoid Function:

In order to map predicted values to probabilities, we use the Sigmoid function. Thefunction maps any real value into another value between 0 and 1. In machine learning, we use sigmoid to map predictions to probabilities.

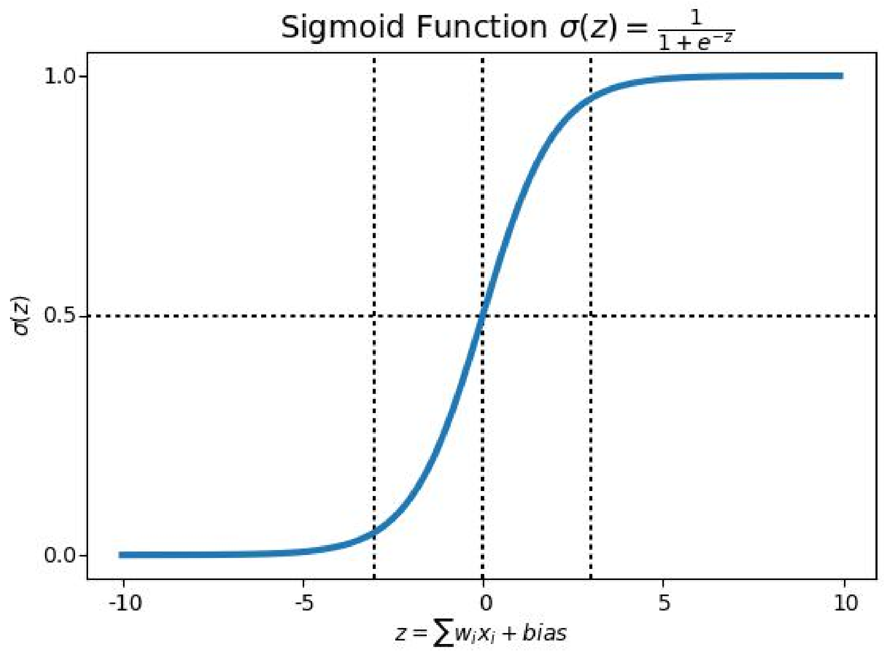


Fig.3.1: Represent Sigmoid function

1.Naive Baye's:

Naive Bayes classifiers are a collection of classification algorithms based on **Bayes' Theorem**. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

They provide aneasy way to build accurate models with very good performance given their simplicity.

They do this by providing a way to calculate the 'posterior' probability of a certain event *A* to occur, given some probabilities of 'prior' events.

$$\boxed{P(A|R) = \frac{P(R|A)P(A)}{P(R)}}$$

P(A): Prior probability of class

P(R|A): Likelihood, the probability of predictor given class

P(R): Prior probability of predictor

P(A|R): posterior probability of class A (target) given the predictor R

Supervised Naive Bayes Algorithm

The steps to perform in order to be able to use the Naive Bayes Algorithm to solve classification problems like the previous problem is:

1. Convert dataset into a frequency table
2. Creates a likelihood table by finding the probabilities of the events to occur.
3. The Naive Bayes equation is used to compute the posterior probability of each class.
4. The class with the higher posterior probability is the outcome of the prediction.

Strengths and Weaknesses of Naive Bayes

The main strengths are:

- Easy and quick way to predict classes, both in binary and multiclass classification problems.
- In the cases that the independence assumption fits, the algorithm performs better compared to other classification models, even with less training data.
- The decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one dimensional

distribution. This helps with problems derived from the curse of dimensionality and improve the performance.

Whereas **the main disadvantages** of using this method **are:**

- Although they are pretty good classifiers, naive bayes are know to be poor estimators. So the probability that outputs from it shouldn't be taken very seriously.
- The naive assumption of independence is very unlikely to match real-world data.
- When the test data set has a feature that has not been observed in the training se, the model will assign a 0 probability to it and will be useless to make predictions.

1.Random Forest:

Random forest is a supervised learning algorithm which is used for both classification as well as regression. But however, it is mainly used for classification problems. As we know that a forest is made up of trees and more trees means more robust forest. Similarly, random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result.

Working of Random Forest Algorithm

We can understand the working of Random Forest algorithm with the help offollowing steps −

Step 1 − First, start with the selection of random samples from a given dataset.

Step 2 − Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.

Step 3 − In this step, voting will be performed for every predicted result.

Step 4 − At last, select the most voted prediction result as the final prediction result.

The following diagram will illustrate its working −

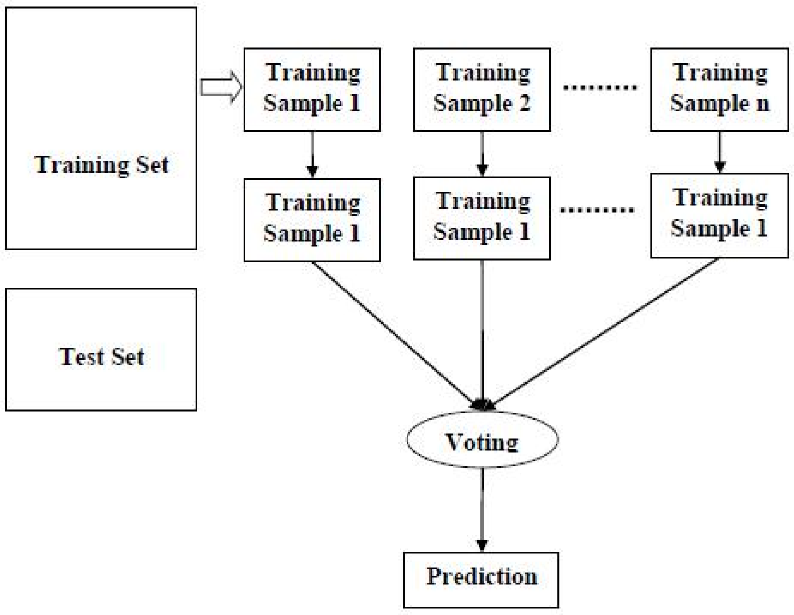


Fig.3.2: Working of Random forest algorithm

Pros and Cons of Random Forest:

Pros

The following are the advantages of Random Forest algorithm −

It overcomes the problem of overfitting by averaging or combining the results of different decision trees.

Random forests work well for a large range of data items than a single decision tree does.

Random forest has less variance then single decision tree.

Random forests are very flexible and possess very high accuracy.

Scaling of data does not require in random forest algorithm. It maintains good accuracy even after providing data without scaling.

Random Forest algorithms maintains good accuracy even a large proportion of the data is missing.

Cons

The following are the disadvantages of Random Forest algorithm −

Complexity is the main disadvantage of Random forest algorithms.

Construction of Random forests are much harder and time-consuming than decision trees.

More computational resources are required to implement Random Forest algorithm.

It is less intuitive in case when we have a large collection of decision trees.

The prediction process using random forests is very time-consuming in comparison with other algorithms.

5.Support Vector Machine(SVM):

In machine learning, support-vector machines (SVMs, also support-vector networks) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on the side of the gap on which they fall.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.When data are unlabelled, supervised learning is not possible, and an unsupervised learning approach is required, which attempts to find natural clustering of the data to groups, and then map new data to these formed groups. The support-vector clustering algorithm, created by Hava Siegelmann and Vladimir Vapnik, applies the statistics of support vectors, developed in the support vector machines algorithm, to categorize unlabeled data, and is one of the most widely used clustering algorithms in industrial applications.More formally, a support-vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like outliers detection. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin, the lower the generalization error of the classifier.

Applications:

1.SVMs are helpful in text and hypertext categorization, as their application can significantly reduce the need for labeled training instances in both the standard inductive and transductive settings. Some methods for shallow semantic parsing are based on support vector machines.

- Classification of images can also be performed using SVMs. Experimental results show that SVMs achieve significantly higher search accuracy than traditional query refinement schemes after just three to four rounds of relevance feedback. This is also true for image segmentation systems, including those using a modified version SVM that uses the privileged approach as suggested by Vapnik.
- Classification of satellite data like SAR data using supervised SVM.
- Hand-written characters can be recognized using SVM.
- The SVM algorithm has been widely applied in the biological and other sciences. They have been used to classify proteins with up to 90% of the compounds classified correctly. Permutation tests based on SVM weights have been suggested as a mechanism for interpretation of SVM models.Support-vector machine weights have also been used to interpret SVM models in the past. Posthoc interpretation of support-vector machine models in order to identify features used by the model to make predictions is a relatively new area of research with special significance in the biological sciences.

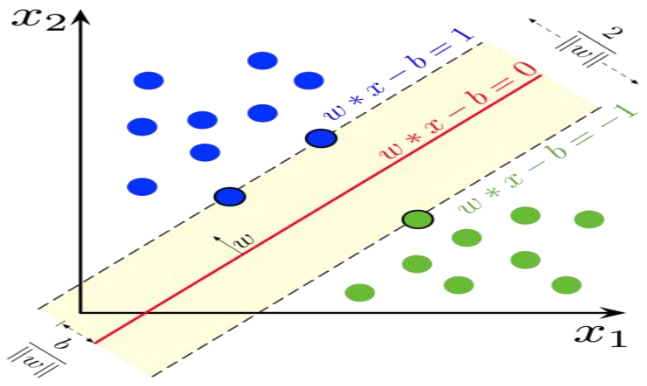


Fig.3.3: Support vector machine graph

Implementation:

The parameters of the maximum-margin hyperplane are derived by solving the optimization. There exist several specialized algorithms for quickly solving the quadratic programming (QP) problem that arises from SVMs, mostly relying on heuristics for breaking the problem down into smaller, more manageable chunks.

Another approach is to use an interior-point method that uses Newton-like iterations to find a solution of the Karush–Kuhn–Tucker conditions of the primal and dual problems. Instead of solving a sequence of broken-down problems, this approach directly solves the problem altogether. To avoid solving a linear system involving the large kernel matrix, a low-rank approximation to the matrix is often used in the kernel trick.

Another common method is Platt's sequential minimal optimization (SMO) algorithm, which breaks the problem down into 2-dimensional sub-problems that are solved analytically, eliminating the need for a numerical optimization algorithm and matrix storage. This algorithm is conceptually simple, easy to implement, generally faster, and has better scaling properties for difficult SVM problems.

The special case of linear support-vector machines can be solved more efficiently by the same kind of algorithms used to optimize its close cousin, logistic regression; this class of algorithms includes sub-gradient descentand coordinate descent.LIBLINEAR has some attractive training-time properties. Each convergence iteration takes time linear in the time taken to read the train data, and the iterations also have a Q-linear convergence property, making the algorithm extremely fast.

The general kernel SVMs can also be solved more efficiently using sub-gradient descent (e.g. P-packSVM), especially when parallelization is allowed.

Structured SVM:

SVMs have been generalized to structured SVMs, where the label space is structured and of possibly infinite size.

Regression:

A version of SVM for regression was proposed in 1996 by Vladimir N. Vapnik, Harris Drucker, Christopher J. C. Burges, Linda Kaufman and Alexander J. Smola. This method is called support-vector regression (SVR). The model produced by support-vector classification (as described above) depends only on a subset of the training data, because the cost function for building the model does not care about training points that lie beyond the margin. Analogously, the model produced by SVR depends only on a subset of the training data, because the cost function for building the model ignores any training data close to the model prediction. Another SVM version known as least-squares support-vector machine (LS-SVM) has been proposed by Suykens and Vandewalle.

Applications of SVM:

1. Face Detection
2. Classification of images
3. Bioinformatics
4. Handwriting Recognition
5. Generalised predictive control
6. Text and Hypertext Categorization
7. Protein fold and remote homology Detection

Extensions of SVM:

Support-vector clustering (SVC):

SVC is a similar method that also builds on kernel functions but is appropriate for unsupervised learning. It is considered a fundamental method indata science.

Multiclass SVM:

Multiclass SVM aims to assign labels to instances by using support-vector machines, where the labels are drawn from a finite set of several elements.

The dominant approach for doing so is to reduce the single multiclass problem into multiple binary classification problems. Common methods for such reduction include:

Building binary classifiers that distinguish between one of the labels and the rest (*one-versus-all*) or between every pair of classes (*one-versus-one*). Classification of new instances for the one-versus-all case is done by a winner-takes-all strategy, in which the classifier with the highest-output function assigns the class (it is important that the output functions be calibrated to produce comparable scores). For the one-versus-one approach, classification is done by a max-wins voting strategy, in which every classifierassigns the instance to one of the two classes, then the vote for the assigned class is increased by one vote, and finally the class with the most votes determines the instance classification.

3.2.2 Algorithms Comparison:

Table 3.1 : Algorithms comparison

Learning Method	Generative or Descriminative?	Loss Function	Decision Boundary	Parameter Estimation Algorithm
Logistic Regression	Discriminative	− log P(Y |X)	Linear	No closed form estimate. Opti-mize objective function using gradient descent.
K-Nearest Neighbors	Discriminative	zero-one loss	Arbitrarily complicated	Must store all training data to classify new points. Choose K using cross validation
Support Vector Machines (with slack variables, no kernel)	Discriminative	hinge loss: |1−y(wT x)|+	linear (depends on kernel)	Solve quadratic program to find boundary that maximizes margin
Naive Baye's	Generative	$P(y\|X) = \frac{P(X\|y)P(y)}{P(X)}$	linear decision boundary	EM algoritmfo r parameter estimation
Random Forest	Discriminative	$G(t) = 1 - \sum_{k=1}^{Q} p^2(k\|t)$	Typically better than single decision trees	Meta Estimator.

3.3 SYSTEM ANALYSIS METHODS :

3.3.1 Use case diagram:

To model a system, the most important aspect is to capture the dynamic behavior. Dynamic behavior means the behavior of the system when it is running/operating. Only static behaviour is not sufficient to model a system rather dynamic behavior is more important than static behavior. There should be some internal or external factors for making the interaction.Use case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements.

Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems. System objectives can include planning overall requirements, validating a hardware design, testing and debugging a software product under development, creating an online help reference, or performing a consumer service-oriented task.

A use case diagram contains four components:

- ✧ The boundary, which defines the system of interest in relation to the world around it.
- ✧ The boundary, which defines the system of interest in relation to the world around it.
- ✧ The use cases, which are the specific roles played by the actors within and around the system.
- ✧ The relationships between and among the actors and the use cases.

A Use Case model can be developed by following the steps below:

1.Identify the Actors (role of users) of the system.

2. For each category of users, identify all roles played by the users relevant to the system.

3. Identify what are the users required the system to be performed to achieve these goals.

4. Create use cases for every goal.

5. Structure the use cases.

6. Prioritize, review, estimate and validate the users.

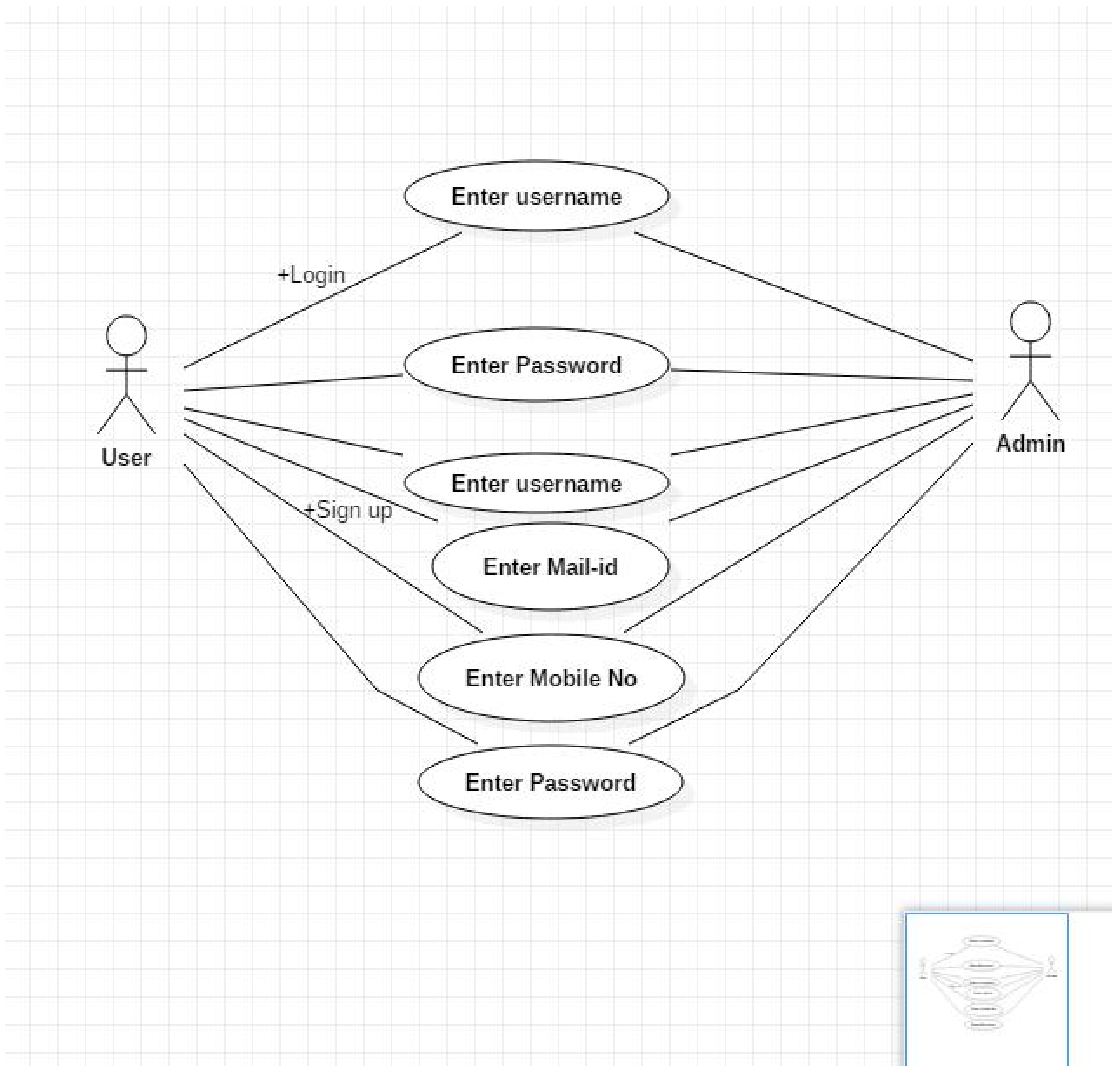


Fig.3.4: Use case Diagram for Hotel Recommendation System

3.3.2 Activity Diagram:

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join,etc.

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

Purpose of Activity Diagrams:

The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.

Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

It does not show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not. It shows different flows such as parallel, branched, concurrent, and single.

The purpose of an activity diagram can be described as -

- ✧ Draw the activity flow of a system.
- ✧ Describe the sequence from one activity to another.
- ✧ Describe the parallel, branched and concurrent flow of the system.

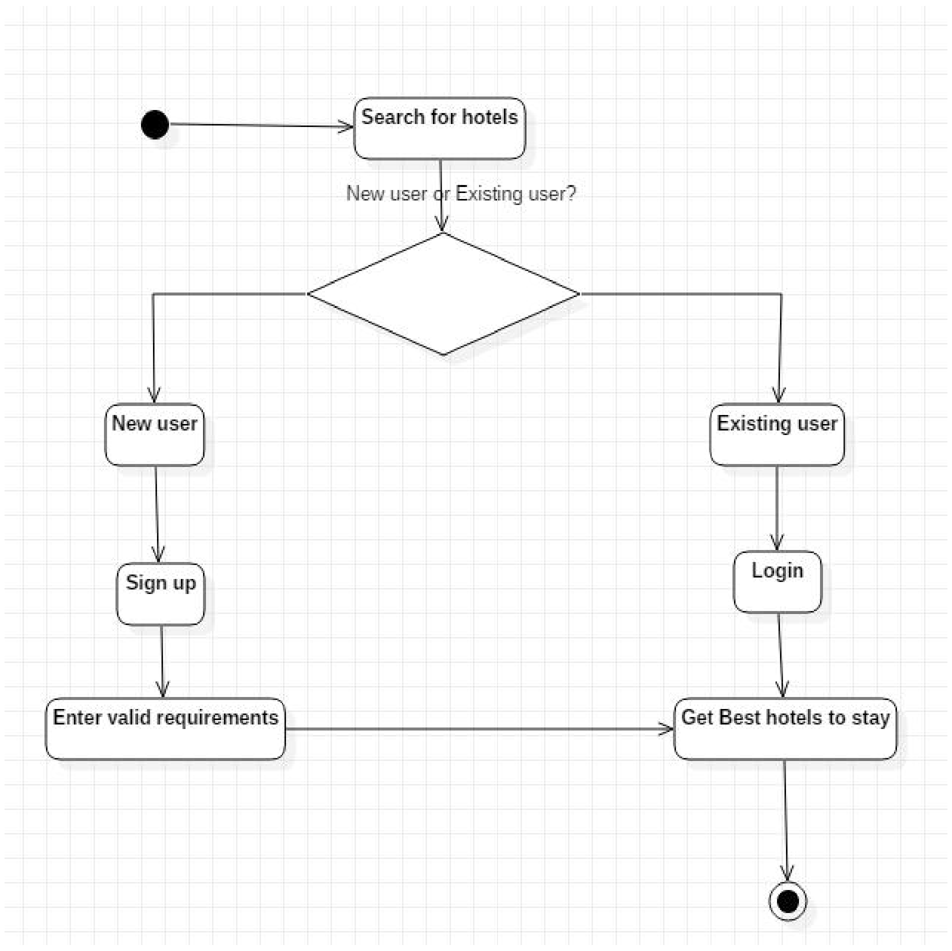


Fig.3.5: Activity Diagram for Hotel Recommendation System

3.4 SYSTEM REQUIREMENTS:

Software Requirements:

Operating system : Windows

Coding Language : PYTHON

Tool : Python Flask.

SOFTWARE ENVIRONMENT :

PYTHON:

Python is a interpreted, high-level, geberal purpose programming language. Created by Guido Van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significantwhitespace. 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 procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard Library.

Features of Python:

1.Easy to code:

Python is high level programming language.Python is very easy to learn language as compared to other language like c, c#, java script, java etc.It is very easy to code in python language and anybody can learn python basic in few hours or days.It is also developer-friendly language.

2.Free and Open Source:

Python language is freely available at official website and you can download it from the given download link below click on the **Download Python** kyeword.

Since, it is open-source, this means that source code is also available to the public.So you can download it as, use it as well as share it.

3.Object-Oriented Language:

One of the key features of python is Object-Oriented programming.Python supportsobject oriented language and concepts of classes, objects encapsulation etc

4.GUI Programming Support:

Graphical Users interfaces can be made using a module such as PyQt5, PyQt4, wxPython or Tk in python.
PyQt5 is the most popular option for creating graphical apps with Python

5.High-Level Language:

Python is a high-level language.When we write programs in python, we do not need to remember the system architecture, nor do we need to manage the memory.

6.Extensible feature:

Python is a **Extensible** language.we can write our some python code into c or c++ language and also we can compile that code in c/c++ language.

7.Python is Portable language:

Python language is also a portable language.for example, if we have python code for windows and if we want to run this code on other platform such as Linux,Unix and Mac then we do not need to change it, we can run this code on any platform.

8.Python is Integrated language:

Python is also an Integrated language because we can easily integrated python with other language like c, c++ etc.

9.Interpreted Language:

Python is an Interpreted Language. because python code is executed line by line at a time. like other language c, c++, java etc there is no need to compile python code this makes it easier to debug our code.The source code of python is converted into an immediate form called **bytecode**.

10.Large Standard Library

Python has a large standard library which provides rich set of module and functions so you do not have to write your own code for every single thing.There aremany libraries present in python for such as regular expressions, unit-testing, web browsers etc.

11.Dynamically Typed Language:

Python is dynamically-typed language. That means the type (for example- int, double, long etc) for a variable is decided at run time not in advance.because of this feature we don't need to specify the type of variable.

3.5 SYSTEM DESIGN :

3.5.1 Class Diagram:

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with objectoriented languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

The purpose of the class diagram can be

● Analysis and design of the static view of an application.

● Describe responsibilities of a system.

● Base for component and deployment diagrams.

● Forward and reverse engineering.

Class Diagram For Hotel Recommendation System:

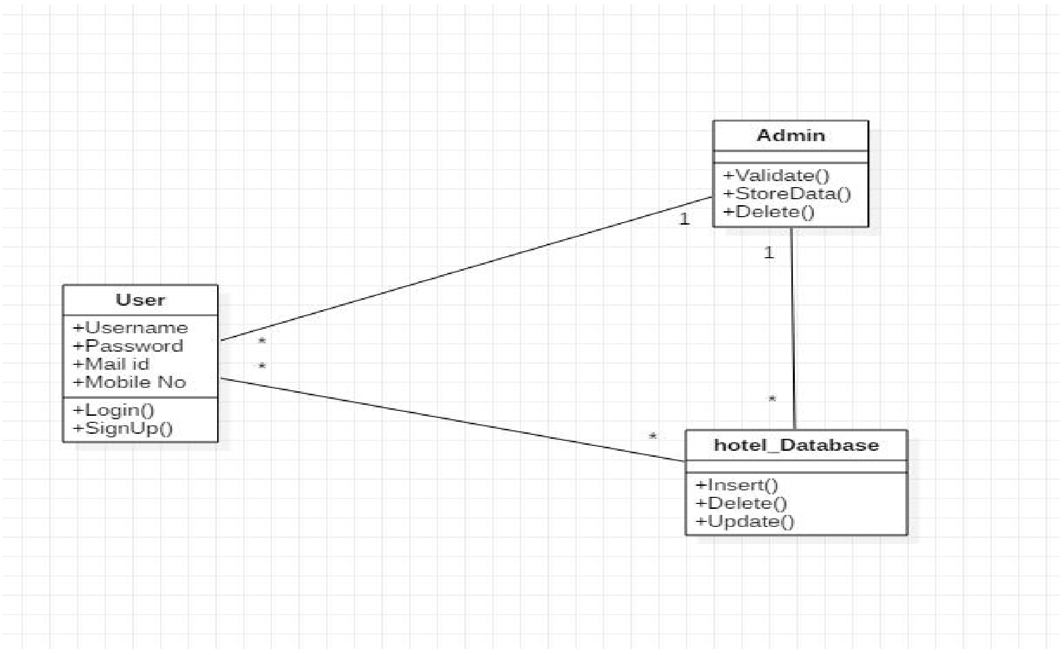


Fig.3.6: Class Diagram of Hotel Recommendation System

3.5.2 Sequence Diagram :

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

Sequence Diagram Notations –

Actors – An actor in a UML diagram represents a type of role where it interacts with the system and its objects. It is important to note here that an actor is always outside the scope of the system we aim to model using the UML diagram.

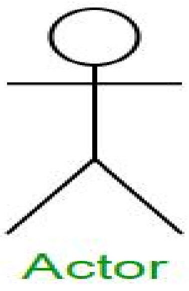


Fig 3.7: Notation symbol for actor

We use actors to depict various roles including human users and other external subjects. We represent an actor in a UML diagram using a stick person notation. We can have multiple actors in a sequence diagram.

Lifelines – A lifeline is a named element which depicts an individual participant in a sequence diagram. So basically each instance in a sequence diagram is represented by a lifeline. Lifeline elements are located at the top in a sequence diagram. The standard in UML for naming a lifeline follows the following format – Instance Name : Class Name.

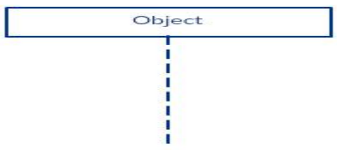


Fig 3.8 : Lifeline

We display a lifeline in a rectangle called head with its name and type. The head is located on top of a vertical dashed line (referred to as the stem) as shown above. If we want to model an unnamed instance, we follow the same pattern except now the portion of lifeline's name is left blank.

Difference between a lifeline and an actor – A lifeline always portrays an object internal to the system whereas actors are used to depict objects external to the system. The following is an example of a sequence diagram

Messages – Communication between objects is depicted using messages. The messages appear in a sequential order on the lifeline. We represent messages using arrows. Lifelines and messages from the core of a sequence diagram.

Messages can be broadly classified into the following categories :
A sequence diagram with different types of messages

Synchronous messages – A synchronous message waits for a reply before the interaction can move forward. The sender waits until the receiver has completed the processing of the message. The caller continues only when it knows that the receiver has processed the previous message i.e. it receives a reply message. A large number of calls in object oriented programming are synchronous.

We use a solid arrow head to represent a synchronous message.

Asynchronous Messages – An asynchronous message does not wait for a reply from the receiver. The interaction moves forward irrespective of the receiver processing the previous message or not.

We use a lined arrow head to represent an asynchronous message.

Create message – We use a Create message to instantiate a new object in the sequence diagram. There are situations when a particular message call requires the creation of an object. It is represented with a dotted arrow and create word labelled on it to specify that it is the create Message symbol.

For example – The creation of a new order on a e-commerce website would require a new object of Order class to be created.

Delete Message – We use a Delete Message to delete an object. When an object is deallocated memory or is destroyed within the system we use the Delete Message symbol. It destroys the occurrence of the object in the system. It is represented by an arrow terminating with a x.

For example – In the scenario below when the order is received by the user, the object of order class can be destroyed.

Self Message – Certain scenarios might arise where the object needs to send a message to itself.

Such messages are called Self Messages and are represented with a U shaped arrow.

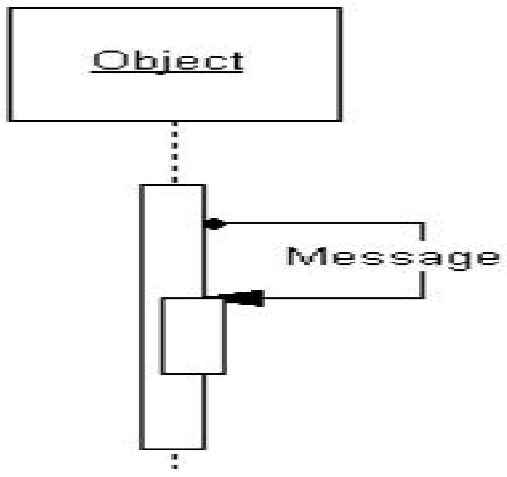


Fig 3.9: self message

Reply Message – Reply messages are used to show the message being sent from the receiver to the sender. We represent a return/reply message using an open arrowhead with a dotted line. The interaction moves forward only when a reply message is sent by the receiver.

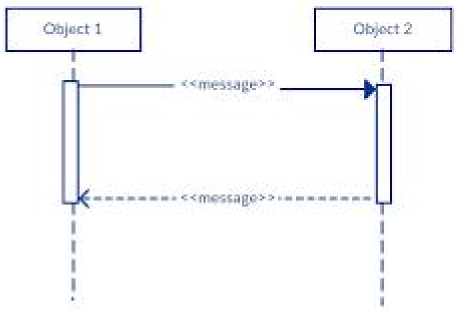


Fig 3.10: Reply message

Found Message – A Found message is used to represent a scenario where unknown source sends the message. It is represented using an arrow direct towards a lifeline from an endpoint. For example: Consider the scenario of a hardware failure.

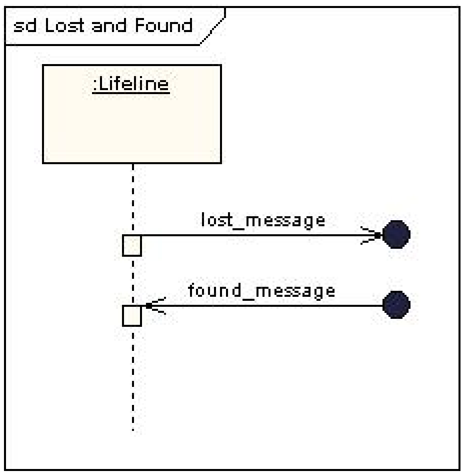


Fig 3.11 : Found message

Lost Message – A Lost message is used to represent a scenario where the recipient is not known to the system. It is represented using an arrow directed towards an end point from a lifeline. For example: Consider a scenario where a warning is generated.

Sequence Diagram for Hotel Recommended System:

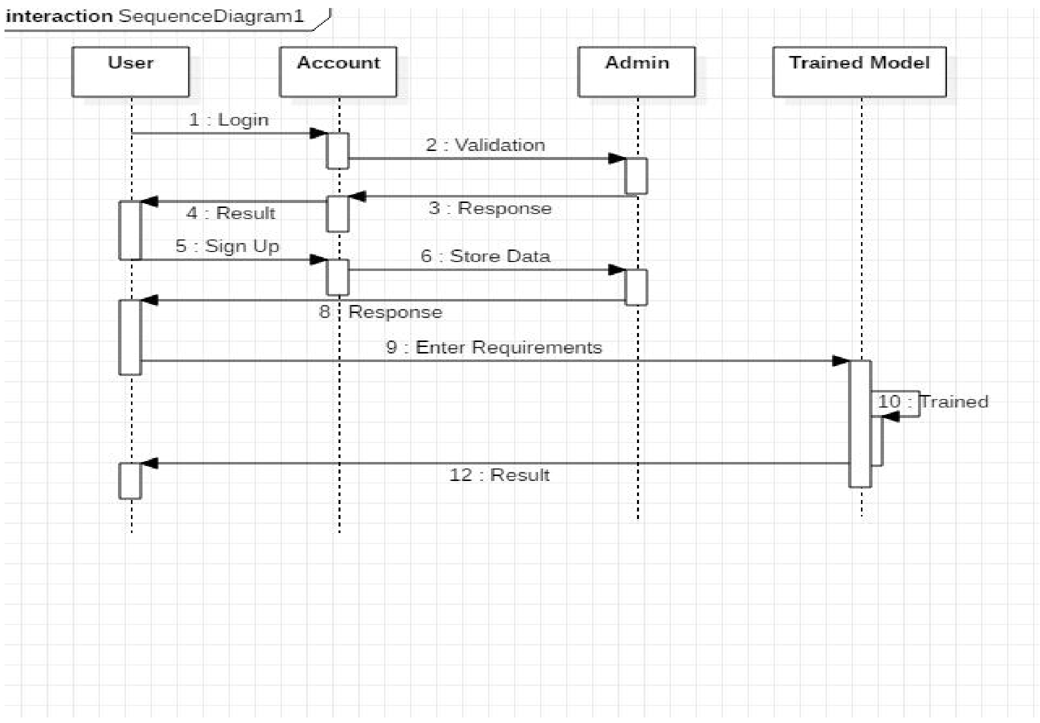


Fig.3.12: Sequence Diagram for Hotel Recommendation System

CHAPTER-4

IMPLEMENTATION

4.1 TOOLS USED :

Bootstrap:

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-firstfront-end web development. It contains CSS and (optionally) JavaScript

based design template for typography, forms, buttons, navigation, and other interface components.Bootstrap is the sixth-most-starred project on GitHub, with more than 135,000 stars, behind freeCodeCamp (almost 307,000 stars) and marginally behind Vue.js framework. According to Alexa Rank, Bootstrap getbootstrap.com is in the top-2000 in US while vuejs.org is in top-7000 in US.Early beginningsBootstrap,originally named Twitter Blueprint, was developed by Mark Otto andJacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden. According to Twitter developer Mark Otto:

A super small group of developers and I got together to design and build a new internal tool and saw an opportunity to do something more. Through that process, we saw ourselves build something much more substantial than another internal tool. Months later, we ended up with an early version of Bootstrap as a way to document and share common design patterns and assets within the company. After a few months of development by a small group, many developers at Twitter began to contribute to the project as a part of Hack Week, a hackathon-style week for the Twitter development team. It was renamed from Twitter Blueprint to Bootstrap, and released as an open source project on August 19, 2011. It has continued to be maintained by Mark Otto, Jacob Thornton, and a small group of core developers, as well as a large community of contributors.

Bootstrap 2 and 3

On January 31, 2012, Bootstrap 2 was released, which added built-in support for Glyphicons, several new components, as well as changes to many of the existing components. This version supports responsive web design. This means the layout ofweb pages adjusts dynamically, taking into account the characteristics of the device used (desktop, tablet, mobile phone). The next major version, Bootstrap 3, was released on August 19, 2013. It redesigned components to use flat design, and a mobile first approach.

Bootstrap 4

Mark Otto announced Bootstrap 4 on October 29, 2014. The first alpha version of Bootstrap 4 was released on August 19, 2015.The first beta version was released on

10 August 2017. Mark suspended work on Bootstrap 3 on September 6, 2016, to free up time to work on Bootstrap 4. Bootstrap 4 was finalized on January 18, 2018.

Significant changes include:

- ✧ Major rewrite of the code
- ✧ Replacing Less with Sass
- ✧ Addition of Reboot, a collection of element-specific CSS changes in a single file, based on Normalize
- ✧ Dropping support for IE8, IE9, and iOS 6
- ✧ CSS Flexible Box support
- ✧ Adding navigation customization options
- ✧ Adding responsive spacing and sizing utilities
- ✧ Switching from the pixels unit in CSS to root ems
- ✧ Increasing global font size from 14px to 16px
- ✧ Dropping the panel, thumbnail, pager, and well components
- ✧ Dropping the Glyphicons icon font
- ✧ Huge number[quantify] of utility classes

Improved form styling, buttons, drop-down menus, media objects and image classes

Bootstrap 4 supports the latest versions of the Google Chrome, Firefox, Internet Explorer, Opera, and Safari (except on Windows). It additionally supports back to IE9 and the latest Firefox Extended Support Release (ESR).

Bootstrap 5

Bootstrap 5 is supposed to be the upcoming version of the most popular open source front-end framework in the world. Major changes include:

- ✧ Dropping jQuery in favor of vanilla JavaScript
- ✧ Rewriting the grid to support columns placed outside of rows and responsive gutters
- ✧ Migrating the documentation from Jekyll to Hugo
- ✧ Dropping support for IE10
- ✧ Moving testing infrastructure from QUnit to Jasmine
- ✧ Adding custom set of SVG icons

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.Bootstrap also comes with several JavaScript components in the form of jQuery plugins. They provide additional user interface elements such as dialog boxes, tooltips, and carousels. Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields.

The most prominent components of Bootstrap are its layout components, as they affect an entire web page. The basic layout component is called "Container", as every other element in the page is placed in it. Developers can choose between a fixed-width container and a fluid-width container.While the latter always fills the width of the web page, the former uses one of the four predefined fixed widths, depending on the size of the screen showing the page:

- Smaller than 576 pixels
- 576–768 pixels
- 768–992 pixels
- 992–1200 pixels
- Larger than 1200 pixels

Once a container is in place, other Bootstrap layout components implement a CSS grid layout through defining rows and columns.

A precompiled version of Bootstrap is available in the form of one CSS file and three JavaScript files that can be readily added to any project. The raw form of Bootstrap,however, enables developers to implement further customization and size optimizations. This raw form is modular, meaning that the developer can remove unneeded components, apply a theme and modify the uncompiled Sass files.

HTML(Hyper Text Markup Language):

Hypertext Markup Language (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 elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img /> and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.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. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

In 1980, physicist Tim Berners-Lee, a contractor at CERN, proposed and prototyped ENQUIRE, a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo an Internet-based hypertext system. Berners-Lee specified HTML and wrote the browser and server software in late 1990. That year, Berners-Lee and CERN data systems engineer Robert Cailliau collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes from 1990 he listed"some of the many areas in which hypertext is used" and put an encyclopedia first.The first publicly available description of HTML was a document called "HTML Tags", first mentioned on the Internet by Tim Berners-Lee in late 1991.It describes 18 elements comprising the initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by SGMLguid, an in-house Standard Generalized Markup Language (SGML)-based documentation format at CERN. Eleven of these elements still exist in HTML 4.

HTML is a markup language that web browsers use to interpret and compose text, images, and other material into visual or audible web pages. Default characteristics for every item of HTML markup are defined in the browser, and these characteristics can be altered or enhanced by the web page designer's additional use of CSS. Many of the text elements are found in the 1988 ISO technical report TR 9537 Techniques for using SGML, which in turn covers the features of early text formatting languages such as that used by the RUNOFF command developed in the early 1960s for the CTSS (Compatible Time-Sharing System) operating system: these formatting commands were derived from the commands used by typesetters to manually format documents. However, the SGML concept of generalized markup is based on elements (nested annotated ranges with attributes) rather than merely print effects, with also the separation of structure and markup; HTML has been progressively moved in this direction with CSS.

Berners-Lee considered HTML to be an application of SGML. It was formally defined as such by the Internet Engineering Task Force (IETF) with the mid-1993 publication of the first proposal for an HTML specification, the "Hypertext Markup Language (HTML)" Internet Draft by Berners-Lee and Dan Connolly, which included an SGML Document type definition to define the grammar.[9][10] The draft expired after six months, but was notable for its acknowledgment of the NCSA Mosaic browser's custom tag for embedding in-line images, reflecting the IETF's philosophy of basing standards on successful prototypes. Similarly, Dave Raggett's competing Internet-Draft, "HTML+ (Hypertext Markup Format)", from late 1993, suggested standardizing already-implemented features like tables and fill-out forms.

After the HTML and HTML+ drafts expired in early 1994, the IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based.

Further development under the auspices of the IETF was stalled by competing interests. Since 1996, the HTML specifications have been maintained, with input from commercial software vendors, by the World Wide Web Consortium (W3C). However, in 2000, HTML also became an international standard (ISO/IEC 15445:2000). HTML 4.01 was published in late 1999, with further errata published through 2001. In 2004, development began on HTML5 in the Web Hypertext Application Technology Working Group (WHATWG), which became a joint deliverable with the W3C in 2008, and completed and standardized on 28 October 2014.

HTML documents imply a structure of nested HTML elements. These are indicated in the document by HTML tags, enclosed in angle brackets thus: <p>

In the simple, general case, the extent of an element is indicated by a pair of tags: a "start tag" <p> and "end tag" </p>. The text content of the element, if any, is placed between these tags.Tags may also enclose further tag markup between the start and end, including a mixture of tags and text. This indicates further (nested) elements, as children of the parent element.The start tag may also include attributes within the tag. These indicate other information, such as identifiers for sections within the document, identifiers used to bind style information to the presentation of the document, and for some tags such as the <img> used to embed images, the reference to the image resource.Some elements, such as the line break <br>, do not permit any embedded content, either text or further tags. These require only a single empty tag (akin to a start tag) and do not use an end tag.

- ✧ Many tags, particularly the closing end tag for the very commonly used paragraph element <p>, are optional. An HTML browser or other agent can infer the closure for the end of an element from the context and the structural rules defined by the HTML standard. These rules are complex and not widely understood by most HTML coders.The general form of an HTML element is <tag attribute1="value1" attribute2="value2">"content"</tag>.
- ✧ Some HTML elements are defined as empty element take the form <tag attribute1="value1" attribute2="value2">.
- ✧ Empty elements may enclose no content, for instance, the <br> tag or the inline <img> tag. The name of an HTML element is the name used in the tags. Note that the end tag's name is preceded by a slash character, /, and that in empty elements the end tag is neither required nor allowed. If attributes are not mentioned, default values are used in each case.
- ✧ Most of the attributes of an element are name-value pairs, separated by = and written within the start tag of an element after the element's name. The value may be enclosed in single or double quotes, although values consisting of certain characters can be left unquoted in HTML (but not XHTML).[73][74] Leaving attribute values unquoted is considered unsafe. In contrast with name-value pair attributes, there are some attributes that affect the element simply by their presence in the start tag of the element, like the ismap attribute for the img element.
- ✧ There are several common attributes that may appear in many elements

- The id attribute provides a document-wide unique identifier for an element. This is used to identify the element so that stylesheets can alter its presentational properties, and scripts may alter, animate or delete its contents or presentation. Appended to the URL of the page, it provides a globally unique identifier for the element, typically a sub-section of the page.
- The class attribute provides a way of classifying similar elements. This can be used for semantic or presentation purposes. For example, an HTML document might semantically use the designation <class="notation"> to indicate that all elements with this class value are subordinate to the maintext of the document. In presentation, such elements might be gathered together and presented as footnotes on a page instead of appearing in the place where they occur in the HTML source. Class attributes are used semantically in microformats. Multiple class values may be specified; for example <class="notation important"> puts the element into both the notation and the important classes.
- An author may use the style attribute to assign presentational properties to a particular element. It is considered better practice to use an element's id or class attributes to select the element from within a stylesheet, though sometimes this can be too cumbersome for a simple, specific, or ad hoc styling.
- The title attribute is used to attach subtextual explanation to an element. In most browsers this attribute is displayed as a tooltip.
- The lang attribute identifies the natural language of the element's contents, which may be different from that of the rest of the document. For example, in an English-language document:
- <p>Oh well, <span lang="fr">c'est la vie</span>, as they say in France.</p>
- The abbreviation element, abbr, can be used to demonstrate some of these **attributes:**
- <abbr id="anId" class="jargon" style="color:purple;" title="Hypertext Markup Language">HTML</abbr>
- This example displays as HTML; in most browsers, pointing the cursor at the abbreviation should display the title text "Hypertext Markup Language."
- Most elements take the language-related attribute dir to specify text direction, such as with "rtl" for right-to-left text in.

Document type declaration:

HTML documents are required to start with a Document Type Declaration (informally, a "doctype"). In browsers, the doctype helps to define the rendering mode—particularly whether to use quirks mode**.**

The original purpose of the doctype was to enable parsing and validation of HTML documents by SGML tools based on the Document Type Definition (DTD). The DTD to which the DOCTYPE refers contains a machine-readable grammar specifying the permitted and prohibited content for a document conforming to such a DTD. Browsers, on the other hand, do not implement HTML as an application of SGML and by consequence do not read the DTD.

HTML5 does not define a DTD; therefore, in HTML5 the doctype declaration is simpler and shorter:

<!DOCTYPE html>

An example of an HTML 4 doctype

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN""https://www.w3.org/TR/html4/strict.dtd">
```

This declaration references the DTD for the "strict" version of HTML 4.01. SGML-based validators read the DTD in order to properly parse the document and to perform validation. In modern browsers, a valid doctype activates standards mode as opposed to quirks mode.

In addition, HTML 4.01 provides Transitional and Frameset DTDs, as explained below. Transitional type is the most inclusive, incorporating current tags as well as older or "deprecated" tags, with the Strict DTD excluding deprecated tags. Frameset has all tags necessary to make frames on a page along with the tags included in transitional type.

CSS(Cascading Style Sheets):

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like 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, and reduce complexity and repetition in the structural content.Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable. The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents. In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

Selector:

In CSS, selectors declare which part of the markup a style applies to by matching tags and attributes in the markup itself.

Selectors may apply to the following:

- all elements of a specific type, e.g. the second-level headers h2
- elements specified by attribute, in particular:
- id: an identifier unique within the document
- class: an identifier that can annotate multiple elements in a document
- elements depending on how they are placed relative to others in the document

Classes and IDs are case-sensitive, start with letters, and can include alphanumeric characters, hyphens and underscores. A class may apply to any number of instances of any elements. An ID may only be applied to a single element.

Pseudo-classes are used in CSS selectors to permit formatting based on information that is not contained in the document tree. One example of a widely used pseudo-class is :hover, which identifies content only when the user "points to" the visible element, usually by holding the mouse cursor over it. It is appended to a selector as in a:hover or #elementid:hover. A pseudo-class classifies document elements, such as :link or :visited, whereas a pseudo-element makes a selection that may consist of partial elements, such as ::first-line or ::first-letter.Selectors may be combined in many ways to achieve great specificity and flexibility. Multiple selectors may be joined in a spaced list to specify elements by location, element type, id, class, or any combination thereof. The order of the selectors is important. For example, div .myClass {color: red;} applies to all elements of class myClass that are inside div elements, whereas .myClass div {color: red;} applies to all div elements that are in elements of class myClass.

The following table provides a summary of selector syntax indicating usage and the version of CSS that introduced it.

Sources:

CSS information can be provided from various sources. These sources can be the web browser, the user and the author. The information from the author can be further classified into inline, media type, importance, selector specificity, rule order, inheritance and property definition. CSS style information can be in a separate document or it can be embedded into an HTML document. Multiple style sheets can be imported. Different styles can be applied depending on the output device being used. for example, the screen version can be quite different from the printed version, so that authors can tailor the presentation appropriately for each medium.The style sheet with the highest priority controls the content display. Declarations not set in the highest priority source are passed on to a source of lower priority, such as the user agent style. The process is called cascading.One of the goals of CSS is to allow users greater control over presentation. Someone who finds red italic headings difficult to read may apply a different style sheet. Depending on the browser and the web site, a user may choose from various style sheets provided by the designers, or may remove all added styles and view the site using the browser's default styling, or may override just the red italic heading style without altering other attributes.CSS was first proposed by Håkon Wium Lie on October 10, 1994.At the time, Lie was working with Tim Berners-Lee at CERN. Several other style sheet languages for the web were proposed around the same time, and discussions on public mailing lists and inside World Wide Web Consortium resulted in the first W3C CSS Recommendation (CSS1)being released in 1996. In particular, a proposal by Bert Bos was influentialhe became co-author of CSS1, and is regarded as co-creator of CSS.Style sheets have existed in one form or another since the beginnings of Standard Generalized Markup Language (SGML) in the 1980s, and CSS was developed to provide style sheets for the web.One requirement for a web style sheet language was for style sheets to come from different sources on the web. Therefore, existing style sheet languages like DSSSL and FOSI were not suitable. CSS, on the other hand, let a document's style be influenced by multiple style sheets by way of "cascading" styles.As HTML grew, it came to encompass a wider variety of stylistic capabilities to meet the demands of web developers. This evolution gave the designer more control over site appearance, at the cost of more complex HTML. Variations in web browser implementations, suchas ViolaWWW and WorldWideWeb, made consistent site appearance difficult, and users had less control over how web content was displayed. The browser/editor developed by Tim Berners-Lee had style sheets that were hard-coded into the program. The style sheets could therefore not be linked to documents on the web. Robert Cailliau, also of CERN, wanted to separate the structure from the presentation so that different style sheets could describe different presentation for printing, screen-based presentations, and editors.

Improving web presentation capabilities was a topic of interest to many in the web community and nine different style sheet languages were proposed on the www-style mailing list. Of these nine proposals, two were especially influential on what became CSS: Cascading HTML Style Sheets and Stream-based Style Sheet Proposal (SSP). Two browsers served as testbeds for the initial proposals; Lie worked with Yves Lafon to implement CSS in Dave Raggett's Arena browser. Bert Bos implemented his own SSP proposal in the Argo browser. Thereafter, Lie and Bos worked together to develop the CSS standard (the 'H' was removed from the name because these style sheets could also be applied to other markup languages besides HTML).

Lie's proposal was presented at the "Mosaic and the Web" conference (later called WWW2) in Chicago, Illinois in 1994, and again with Bert Bos in 1995. Around this time the W3C was already being established, and took an interest in the development of CSS. It organized a workshop toward that end chaired by Steven Pemberton. This resulted in W3C adding work on CSS to the deliverables of the HTML editorial review board (ERB). Lie and Bos were the primary technical staff on this aspect of the project, with additional members, including Thomas Reardon of Microsoft, participatingas well. In August 1996, Netscape CommunicationCorporation presented an alternative style sheet language called JavaScript Style Sheets (JSSS). The spec was never finished, and is deprecated. By the end of 1996, CSS was ready to become official, and the CSS level 1 Recommendation was published in December.

Development of HTML, CSS, and the DOM had all been taking place in one group, the HTML Editorial Review Board (ERB). Early in 1997, the ERB was split into three working groups: HTML Working group, chaired by Dan Connolly of W3C; DOM Working group, chaired by Lauren Wood of SoftQuad; and CSS Working group, chaired by Chris Lilley of W3C.

The CSS Working Group began tackling issues that had not been addressed with CSS level 1, resulting in the creation of CSS level 2 on November 4, 1997. It was published as a W3C Recommendation on May 12, 1998. CSS level 3, which was started in 1998, is still under development as of 2014.

In 2005, the CSS Working Groups decided to enforce the requirements for standards more strictly. This meant that already published standards like CSS 2.1, CSS 3 Selectors, and CSS 3 Text were pulled back from Candidate Recommendation to Working Draft level.

Some noted limitations of the current capabilities of CSS include:

Selectors are unable to ascend

CSS currently offers no way to select a parent or ancestor of an element that satisfies certain criteria. CSS Selectors Level 4, which is still in Working Draft status, proposes such a selector, but only as part of the "complete" selector profile, not the "fast" profile used in dynamic CSS styling. A more advanced selector scheme (such as XPath) would enable more sophisticated style sheets. The major reasons for the CSS Working Group previously rejecting proposals for parent selectors are related to browser performance and incremental rendering issues.

Cannot explicitly declare new scope independently of position

Scoping rules for properties such as z-index look for the closest parent element with a position:absolute or position:relative attribute. This odd coupling has undesired effects. For example, it is impossible to avoid declaring a new scope when one is forced to adjust an element's position, preventing one from using the desired scope of a parent element.

Pseudo-class dynamic behavior not controllable

CSS implements pseudo-classes that allow a degree of user feedback by conditional application of alternate styles. One CSS pseudo-class, ":hover", is dynamic (equivalent of JavaScript "onmouseover") and has potential for abuse (e.g., implementing cursor-proximity popups), but CSS has no ability for a client to disable it (no "disable"-like property) or limit its effects (no "nochange"-like values for each property).

Cannot name rules

There is no way to name a CSS rule, which would allow (for example) client-side scripts to refer to the rule even if its selector changes.

Cannot include styles from a rule into another rule

CSS styles often must be duplicated in several rules to achieve a desired effect, causing additional maintenance and requiring more thorough testing. Some new CSS features were proposed to solve this, but (as of February, 2016) are not yet implemented anywhere.

Cannot target specific text without altering markup

Besides the :first-letter pseudo-element, one cannot target specific ranges of text without needing to utilize place-holder elements.

JavaScript:.

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behavior, and all major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports 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). However, the language itself does not include any input/output (I/O), such as networking, storage, or graphics facilities, as the host environment (usually a web browser) provides those APIs.

Originally used only in web browsers, JavaScript engines are also now embedded in server-side website deployments and non-browser applications.

Although there are similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

Examples of scripted behavior:

- Loading new page content without reloading the page. For example, social media websites use Ajax so that users can post new messages without leaving the page.
- Animation of page elements, such as fading them in and out, resizing, and moving them.
- Interactive content, such as games and video.
- Validating input values of a web form to make sure that they are acceptable before being submitted to the server.
- Transmitting information about the user's behavior for analytics, ad tracking, and personalization.

Libraries and frameworks:

The majority of websites use a third-party JavaScript library or web application framework as part of their client-side page scripting.

jQuery is the most popular library, used by over 70% of websites.

The Angular framework was created by Google for its web services; it is now open source and used by other websites. Likewise, Facebook created the React framework for its website and later released it as open source; other sites, including Twitter, now use it. There are other open source frameworks in use, such as Backbone.js and Vue.js.

Features:

Imperative and structured

JavaScriptsupports much of the structured programming syntax from C (e.g., if statements, while loops, switch statements, do while loops, etc.). One partial exception is scoping: JavaScript originally had only function scoping with var. ECMAScript 2015 added keywords let and const for block scoping, meaning JavaScript now has both function and block scoping. Like C, JavaScript makes a distinction between expressions and statements. One syntactic difference from C is automatic semicolon insertion, which allows the semicolons that would normally terminate statements to be omitted.

Weakly typed

JavaScript is weakly typed, which means certain types are implicitly cast depending on the operation used. JavaScript has received criticism for the way it implements these conversions as well as the inconsistency between them. For example, when adding a number to a string, the number will be cast to a string before performing concatenation, but when subtracting a number from a string, the string is cast to a number before performing subtraction.

Python Flask:

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Flask is a micro web framework written in Python.It isclassified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.Extensionsare updated far more frequently than the core Flask program.Applications that use Flask framework include Pinterest and LinkedIn.Flask was created by Armin Ronacher of Pocoo, an international group of Python enthusiasts formed in 2004. According to Ronacher, the idea was originally an April Fool's joke that was popular enough to make into a serious application.When Ronacher and Georg Brandl created[when?] a bulletin board system written in Python, the Pocoo projects Werkzeug and Jinja were developed.Flask has become popular among Python enthusiasts. As of January 2020, it has more stars on GitHub than any other Python web-development framework, and was voted the most popular web framework in the Python Developers Survey 2018.

The microframework Flask is based on the Pocoo projects Werkzeug and Jinja2.

Werkzeug

Werkzeug is a utility library for the Python programming language, in other words a toolkit for Web Server Gateway Interface (WSGI) applications, and is licensed under a BSD License. Werkzeug can realize software objects for request, response, and utility functions. It can be used to build a custom software framework on top of it and supports Python 2.6, 2.7 and 3.3.

Jinja

Jinja, also by Ronacher, is a template engine for the Python programming language and is licensed under a BSD License. Similar to the Django web framework, it handles templates in a sandbox.

- Development server and debugger
- Integrated support for unit testing
- RESTful request dispatching
- Uses Jinja templating
- Support for secure cookies (client side sessions)
- 100% WSGI 1.0 compliant
- Unicode-based
- Extensive documentation
- Google App Engine compatibility
- Extensions available to enhance features desired

4.2 COMPONENT DIAGRAM :

The purpose of a component diagram is to show the relationship between different components in a system. For the purpose of UML , the term "component" refers to a module of classes that represent independent systems or subsystems with the ability to interface with the rest of the system.

There exists a whole development approach that revolves around components: componentbased development (CBD). In this approach, component diagrams allow the planner to identify the different components so the whole system does what it's supposed to do.

More commonly, in an object oriented programming approach, the component diagram allows a senior developer to group classes together based on common purpose so that the developer and others can look at a software development project at a high level.

Benefits of component diagrams

Though component diagrams may seem complex at first glance, they are invaluable when it comes to building your system. Component diagrams can help your team:

- Imagine the system's physical structure.
- Pay attention to the system's components and how they relate.
- Emphasize the service behavior as it relates to the interface.

How to use component diagrams:

A component diagram in UML gives a bird's-eye view of your software system. Understanding the exact service behavior that each piece of your software provides will make you a better developer. Component diagrams can describe software systems that are implemented in any programming language or style.

UML is a set of conventions for object-oriented diagrams that has a wide variety of applications. In component diagrams, the Unified Modeling Language dictates that components and packages are wired together with lines representing assembly connectors and delegation connectors.

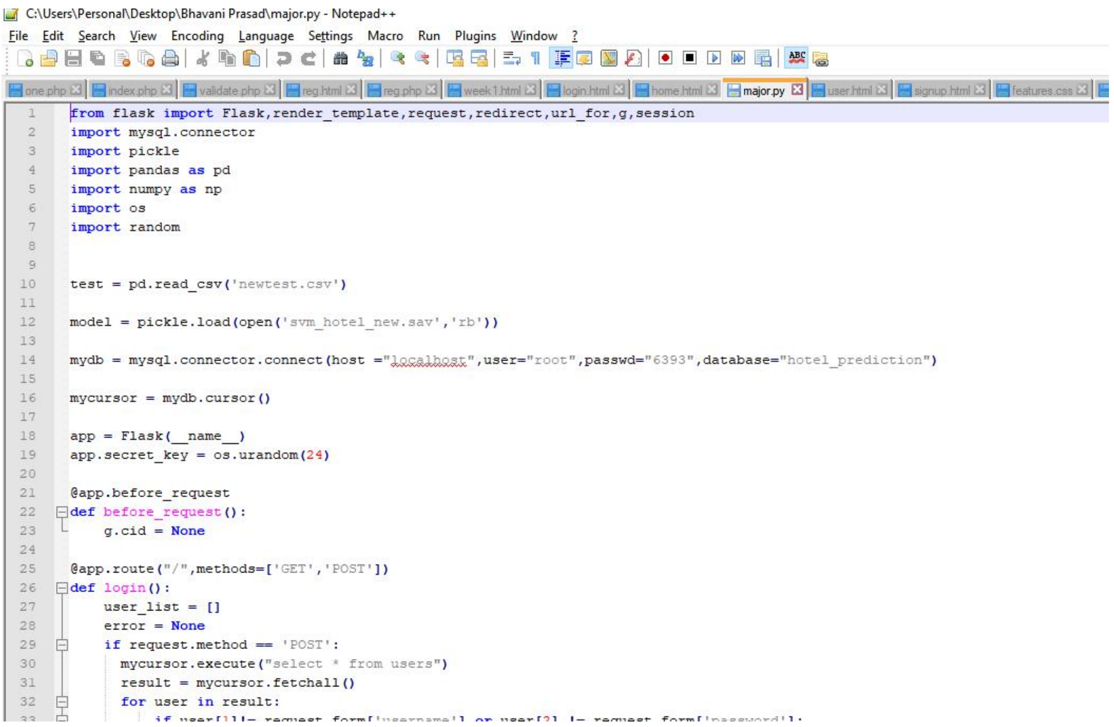
How to use component shapes and symbols

There are three popular ways to create a component's name compartment. You always need to include the component text inside the double angle brackets and/or the component logo. The distinction is important because a rectangle with just a name inside of it is reserved for classifiers (class elements).

Diagram Elements:

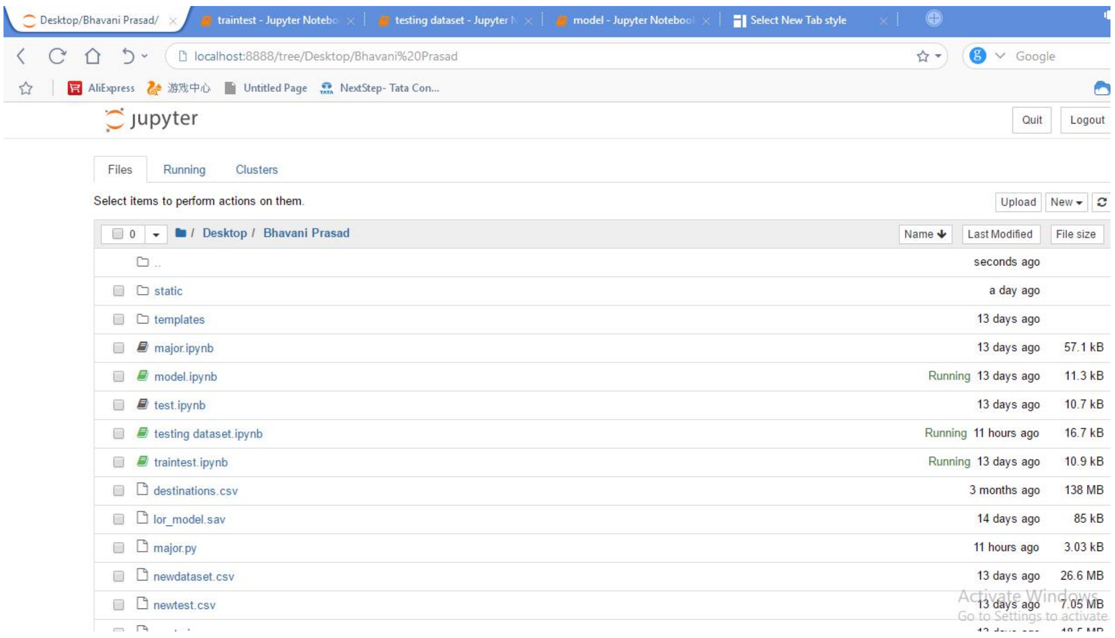
The component diagram extends the information given in a component notation element. One way of illustrating the provided and required interfaces by the specified component is in the form of a rectangular compartment attached to the component element.Another accepted way of presenting the interfaces is to use the ball and socket graphic notation. A provided dependency from a component to an interface is illustrated with a solid line to the component using the interface from a "lollipop", or ball, labelled with the name of the interface. A required usage dependency from a component to an interface is illustrated by a half-circle, or socket, labelled with the name of the interface, attached by a solid line to the component that requires this interface. Inherited interfaces may be shown with a lollipop, preceding the name label with a caret symbol. To illustrate dependencies between the two, use a solid line with a plain arrowhead joining the socket to the lollipop.

4.1 IMPLEMENTATION:


C:\Users\Personal\Desktop\Bhavani Prasad\major.py - Notepad++
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
one.php index.php validate.php reg.html reg.php week1.html login.html home.html major.py user.html signup.html features.css

```
from flask import Flask,render_template,request,redirect,url_for,g,session
import mysql.connector
import pickle
import pandas as pd
import numpy as np
import os
import random


test = pd.read_csv('newtest.csv')

model = pickle.load(open('svm_hotel_new.sav','rb'))

mydb = mysql.connector.connect(host ="localhost",user="root",passwd="6393",database="hotel_prediction")

mycursor = mydb.cursor()

app = Flask(__name__)
app.secret_key = os.urandom(24)

@app.before_request
def before_request():
    g.cid = None

@app.route("/",methods=['GET','POST'])
def login():
    user_list = []
    error = None
    if request.method == 'POST':
      mycursor.execute("select * from users")
      result = mycursor.fetchall()
      for user in result:
          if user[1]!= request.form['username'] or user[2] != request.form['password']:
```

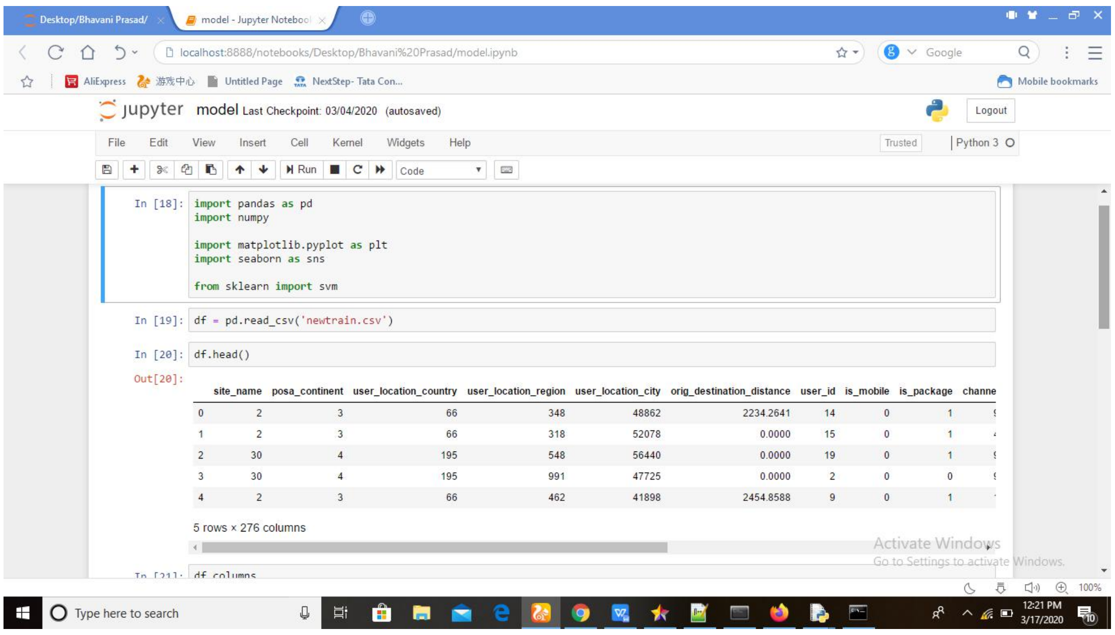
Fig.4.1: Server Side Script



Name	Last Modified	File size
..	seconds ago	
static	a day ago	
templates	13 days ago	
major.ipynb	13 days ago	57.1 kB
model.ipynb (Running)	13 days ago	11.3 kB
test.ipynb	13 days ago	10.7 kB
testing dataset.ipynb (Running)	11 hours ago	16.7 kB
traintest.ipynb (Running)	13 days ago	10.9 kB
destinations.csv	3 months ago	138 MB
lor_model.sav	14 days ago	85 kB
major.py	11 hours ago	3.03 kB
newdataset.csv	13 days ago	26.6 MB
newtest.csv	13 days ago	7.05 MB

localhost:8888/tree/Desktop/Bhavani%20Prasad
jupyter
Quit
Logout
Files
Running
Clusters
Select items to perform actions on them.
Upload
New
Desktop / Bhavani Prasad

Fig.4.2:Jupyter Notebook Dashboard



```
In [18]: import pandas as pd
         import numpy

         import matplotlib.pyplot as plt
         import seaborn as sns

         from sklearn import svm

In [19]: df = pd.read_csv('newtrain.csv')

In [20]: df.head()
Out[20]:
   site_name  posa_continent  user_location_country  user_location_region  user_location_city  orig_destination_distance  user_id  is_mobile  is_package  channe
0          2               3                     66                   348               48862                  2234.2641       14          0           1
1          2               3                     66                   318               52078                     0.0000       15          0           1
2         30               4                    195                   548               56440                     0.0000       19          0           1
3         30               4                    195                   991               47725                     0.0000        2          0           0
4          2               3                     66                   462               41898                  2454.8588        9          0           1

5 rows × 276 columns
```

localhost:8888/notebooks/Desktop/Bhavani%20Prasad/model.ipynb
jupyter model Last Checkpoint: 03/04/2020 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code

Fig.4.3: Train the model

4.3 DEPLOYMENT DIAGRAM:

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships. Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

Purpose of Deployment Diagrams:

The term Deployment itself describes the purpose of the diagram. Deployment diagrams are used for describing the hardware components, where software components are deployed.

Component diagrams and deployment diagrams are closely related.Component diagrams are used to describe the components and deployment diagrams shows how they are deployed in hardware.

Deployment Diagram for Hotel Recommendation System :

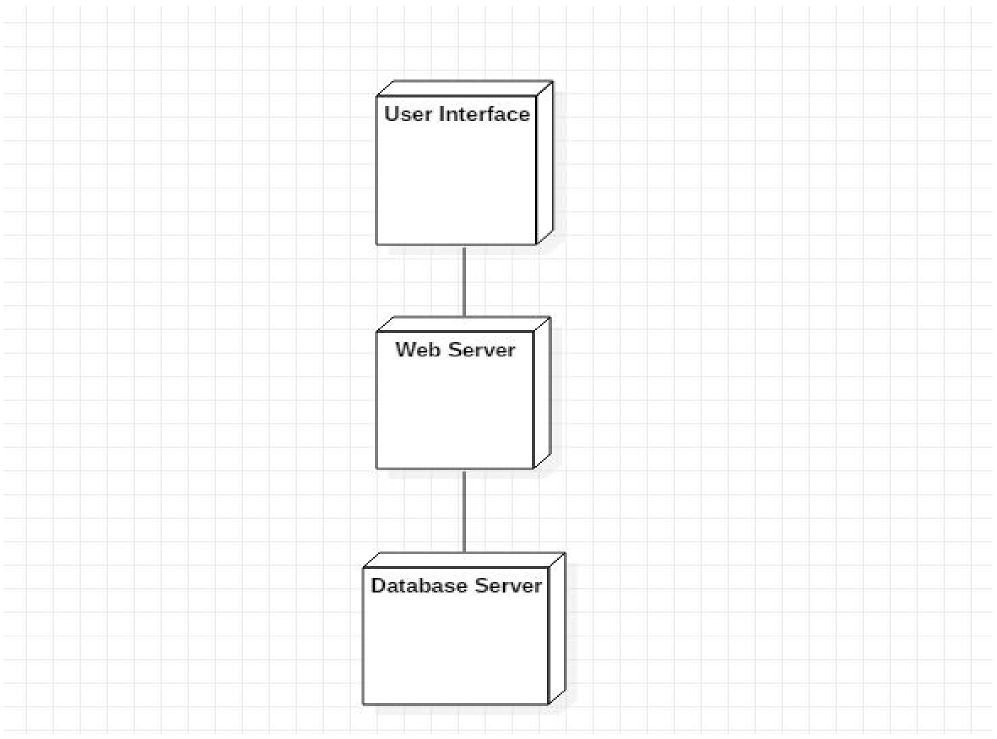


Fig.4.4 :Deployment diagram for Hotel Recommendation System

4.5 SCREENSHOTS

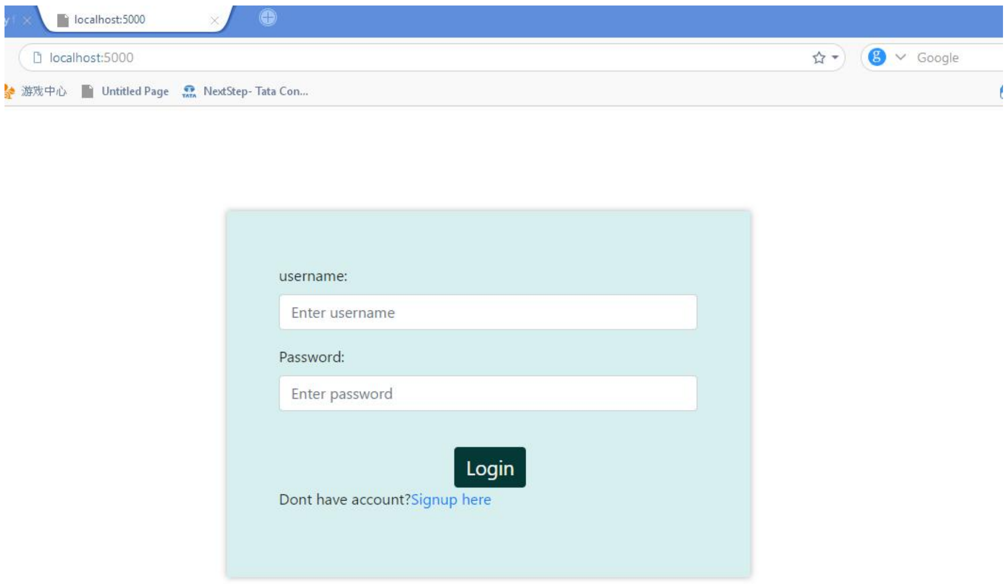


Fig.4.5: Login page of user

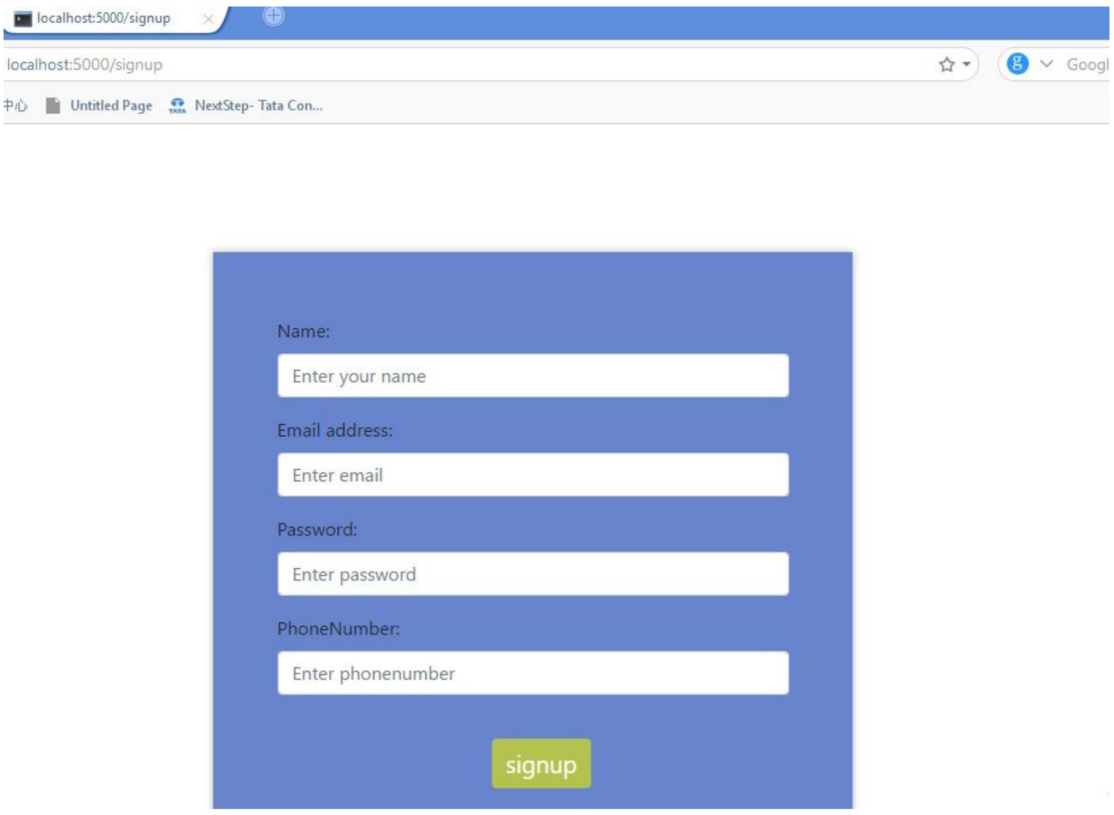
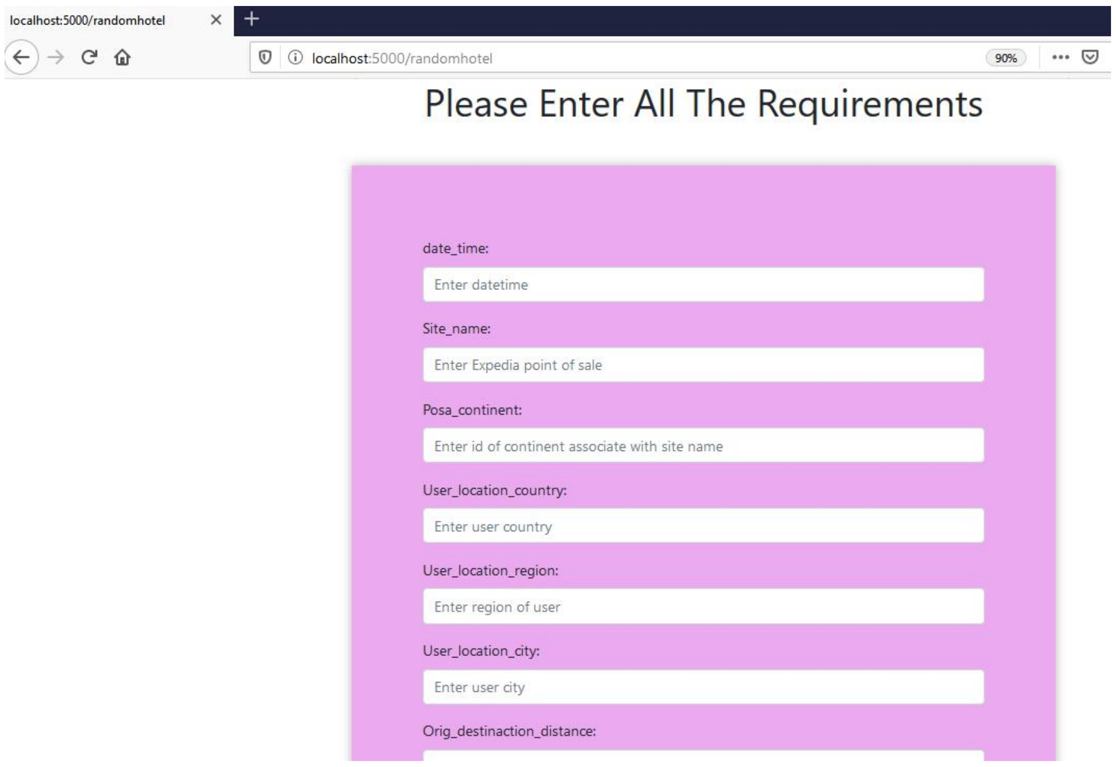


Fig.4.6: SignUp page for user


localhost:5000/randomhotel

Please Enter All The Requirements

date_time:
Enter datetime

Site_name:
Enter Expedia point of sale

Posa_continent:
Enter id of continent associate with site name

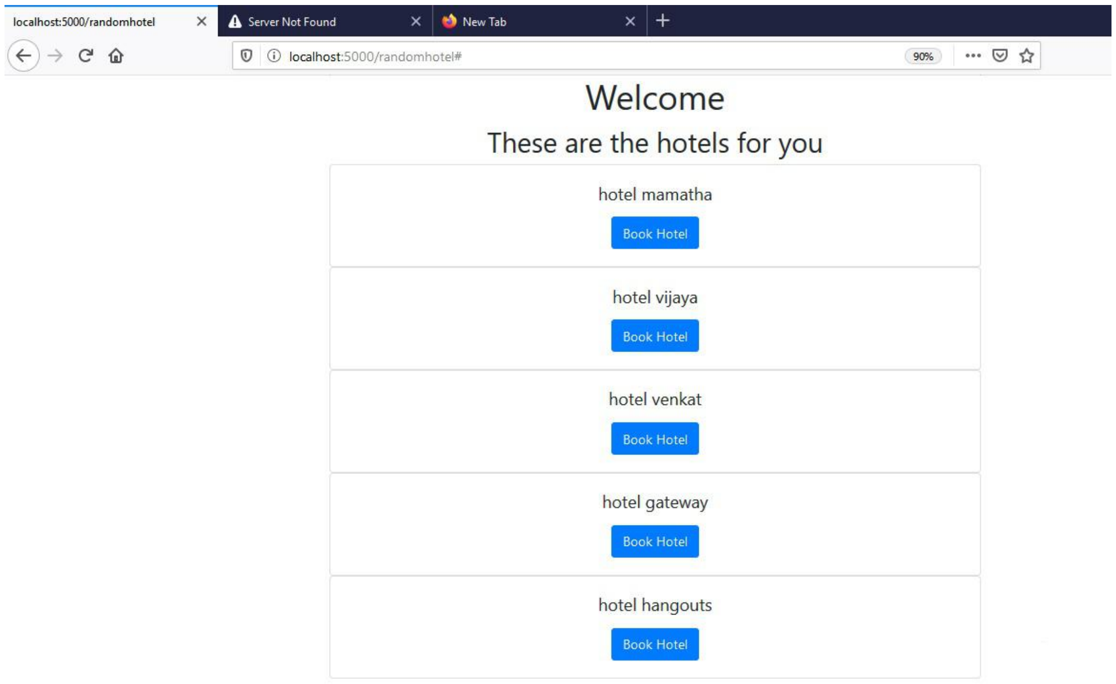
User_location_country:
Enter user country

User_location_region:
Enter region of user

User_location_city:
Enter user city

Orig_destinaction_distance:

Fig.4.7: Requirements Page for user


localhost:5000/randomhotel#

Welcome

These are the hotels for you

hotel mamatha

hotel vijaya

hotel venkat

hotel gateway

hotel hangouts

Fig.4.8:Displaying hotels to user

CHAPTER-5

TESTING

A test case is a specification of the inputs, execution conditions, testing procedure, and expected results that define a single test to be executed to achieve a particular software testing objective, such as to exercise a particular program path or to verify compliance with a specific requirement.

Test cases underlie testing that is methodical rather than haphazard. A battery of test cases can be built to produce the desired coverage of the software being tested. Formally defined test cases allow the same tests to be run repeatedly against successive versions of the software, allowing for effective and consistent regression testing.

A test case is also defined as a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application.

5.1 Test cases :

Table 5.1 Test cases

Input	Actual output	Expected output	Result
Valid Username,password,Requirements	Display hotel names	Display hotel names	Accepted
Invalid Username, Password	Display Invalid username and password	Display Invalid username and password	Accepted
Valid Username,Invalid Password	Displays error message	Displays error message	Accepted
Invalid username,validpassword	Displays error message	Displays error message	Accepted
Enter Invalid Requirements	Please enter valid details	Please enter valid details	Accepted

Enter Invalid requirements	Display error message	Expects the output for query	Rejected

CHAPTER-6

CONCLUSION

This project provides an excellent Recommendation System for applying machine learning algorithms to large data sets. It also giving recommending items about which we have entered features. In addressing these challenges, we demonstrated that a creative combination of user similarity matrices and we successfully implemented a hotel recommendation system using Expedia's dataset even though most of the data anonymized which restricted the amount of feature engineering we could do.

BIBILOGRAPHY

[1] G.Huming and L.weili. A hotel recommendation System based on collobarative filtering and rankboost algorithm. In 2010 second International conference on Multimedia and Information Technology, April 2010.

[2] R. Saga, Yoshilhiro Hayashi, and H. Tsuji. Hotel recommender System based on User's preference transition. In 2008 IEEE International Conference on Systems, Man and Cybernatics, Pages 2437-2442, Oct 2008.

WebLink:
https://towardsdatascience.com/a-machine-learning-approach-building-a-hotel-recommendation-engine-6812bfd53f50?gi=e655fcc89b82.