Project Report

On

“Movies Recommend System”

*Submitted in partial fulfillment of the requirement for the Award of degree of*

*Master of Computer Applications*

By

Rituraj Bharti (162120042)

Rahul sinha (162120061)

**Under the Guidance of**

Dr. Vishnu Priya



**Session 2016-19**

**Department of Mathematics & Computer Applications**

**Maulana Azad National Institute of Technology, Bhopal (MP) April 2019**

**Declaration**

I, hereby declare that the work presented in this project entitled "**Movies Recommend System**" presented in partial fulfillment for the award of the degree of Master of Computer Applications submitted in the Department Of Mathematics and Computer Applications, Maulana Azad National Institute of Technology, Bhopal is an authentic work carried out from 7th January 2019 to 25th April 2019 under the guidance of **Dr. Vishnu Priya**, MANIT Bhopal.

The matter embodied in this project has not been submitted by me or anybody else to any institution for award of any other degree or diploma.

Rituraj Bharti(162120042)

Rahul Sinha(162120061)

**Counter Signed by:**

Supervisor:

Head, Department Of Mathematics and Computer Applications

MANIT, Bhopal.

Dean (Academic) MANIT, Bhopal.

**Acknowledgement**

Here, I gladly present this project report on **“Movies Recommend System”** as part of the 6th semester MCA

Master in Computer Applications. I take this occasion to thank God, almighty for blessing me with his grace and taking our endeavour to a successful culmination. I extend my sincere and heartfelt thanks to me esteemed guide, **Dr. Vishnu Priya** for providing me with the right guidance and advice at the crucial junctures and for showing me the right way. I extend my sincere thanks to my respected head of the department **Dr. Sujoy Das**, for allowing me to use the facilities available.

I am highly indebted to

**Dr. Ramjivan Singh Thakur**

**Dr. …….**

**Dr………**

**Dr. …….**

for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project .Last but not the least, I would like to express my gratitude towards my parents & my friends for the support and encouragement they have given me during the course of my work.

Submitted By -

Rituraj Bharti (162120042)

Rahul Sinha(162120061)

**Certificate**

**2016-2019**

**DEPARTMENT OF MATHEMATICS AND COMPUTER APPLICATIONS**

****

**MAULANA AZAD NATION INSTITUTE OF TECHNOLOGY BHOPAL**

**(M.P.) – 462003**

This is to certify that Rituraj Bharti **- (162120042)** and Rahul Sinha **- (162120061)** have carried out the project work in this report entitled “Movies Recommend System **”** for the award of Master of Computer Application in Maulana Azad National Institute of Technology Bhopal (M.P) - 462003.

This report is the record of the candidates’ own work carried out by them under our supervision and guidance. This project work is the part of their Master in Computer Applications in Information Technology curriculum.

Their performance was excellent and we wish them good luck for their future endeavors.

**Signature of Project Guide Signature of Head of Department**

|  |
| --- |
|  |

**Abstract**

A movie recommendation is important in our social life due to its strength in providing enhanced entertainment. Such a system can suggest a set of movies to users based on their interest, or the popularities of the movies. Although, a set of movie recommendation systems have been proposed, most of these either cannot recommend a movie to the existing users efficiently or to a new user by any means. In this project we propose a movie recommendation system that has the ability to recommend movies to a new user as well as the others. It mines movie databases to collect all the important information, such as, popularity and attractiveness, required for recommendation. It generates movie swarms not only convenient for movie producer to plan a new movie but also useful for movie recommendation. Experimental studies on the real data reveal the efficiency and effectiveness of the proposed system.

In this project, a movie recommender system is built based on the Movie Lens 1M dataset. We used collaborative filtering method to predict user’s movie rating and we can recommend movies to customers, which they potentially give high ratings according to prediction.

This project is going to develop generic software, which can be applied by any educational organization. More over it provides facility.

|  |
| --- |
|  |
|  |

**TABLE OF CONTENTS**

1. Introduction ………………………………………………12-22

1.1 Algorithm Used…………………………………….13-22

* 1. Technology Used…………………………………....22

1.3 Objectives……………………..……………………..22

1. System Analysis…………………………………………….23-26
   1. 2.1 Requirement Analysis….……………………………..24-25
   2. 2.1.1 Modules……………………………………..25
   3. 2.1.2 Configuration………………………………..25-26
   4. 2.2 Preliminary Investigation…………………………….27-28
   5. 2.3 Feasibility Study……………………………………..29
      1. Operational Feasibility……………………....29
      2. Technical Feasibility…………………………29
      3. Economical Feasibility…………………….....29
2. Software Engineering Paradigms………………………….....30-31
3. Analysis Document ………………………………………….31-32

4.1 Software Requirement Specification………………..31-32

1. Code Efficiency………………………………………………33-34
2. Code Optimization……………………………………………34
3. Validation Checks…………………………………………....35
4. Testing……………………………………………………......36-37
5. Implementation………………………………………………38-40
6. Evaluation…………………………………………………….40
7. Maintenance…………………………………………………..41-42
8. System Security……………………………………………….43
9. Cost Estimation……………………………………………44
10. Future Scope Of Project……………………………………45-46
11. Bibliography……………………………………………….47

**LIST OF FIGURES**

Figure 5.3.1: Home………………………………………………………………..39

Figure 5.3.2: Building Block. …………………………………………………….39

Figure 5.3.3: Confusing Word……………………………………………………..40

Figure 5.3.4: Spell Checker………………………………………………………..40

Figure 5.3.5: Word of the day……………………………………………………..41

Figure 5.3.6: Admin……………………………………………………….............41

Figure 5.3.7: Types………………………………………………………………...42

Figure 5.3.8: Synonyms and Antonyms…………………………………………...42

Figure 5.3.9: Phrasal Verb………………………………………………………...,43

Figure 5.3.10: Add Word….………………………………………………………43

Figure 5.3.11: Difference Word………………………………………………….,.44

Figure 5.3.12: Show History………………………………………………………44

Figure 5.3.13: View for Add Word………………………………………………..45

Figure 5.3.14: View for Admin……………………………………………………45

**LIST OF TABLES**

Table 1: WOEDS

Table 2: TRANSICTION

Table 3: DIFFERENCE

Table 4: VIEW\_WORD

Table 5: PHRASAL

Table 6: HISTORY

1**Introduction:**

Recommender systems are used to provide personalized recommendations according to user profile and previous behaviour. Recommender systems are widely used in the Internet Industry. Services like Amazon, Netflix, and YouTube are typical examples of Recommender System users. Recommender Systems cannot only help the users find their favourite products, but also bring potential profit to online service providers.

In this **project**, we attempt to understand the different kinds of **Movies Recommend System** and compare their performance on the Movie Lens dataset. A **recommendation system** is a type of information filtering **system** which attempts to predict the preferences of a user, and make suggests based on these preferences.

**Movies Recommendation System** produced a ranked list of items on which a user might be interested, in the context of his current choice of an item. Subclass of Information filtering **system** that seek to predict the 'rating' or 'preference' that a user would give to them.

Recommender systems usually make use of either or both collaborative filtering and content-based filtering (also known as the personality-based approach), as well as other systems such as knowledge-based systems. Collaborative filtering approaches build a model from a user's past behavior (items previously purchased or selected and/or numerical ratings given to those items) as well as similar decisions made by other users. This model is then used to predict items (or ratings for items) that the user may have an interest in.Content-based filtering approaches utilize a series of discrete, pre-tagged characteristics of an item in order to recommend additional items with similar properties. Current recommender systems typically combine one or more approaches into a hybrid system.

**1.1Algorithm Used:**

**1. Collaborative Filtering**

* Nearest-neighbors
* Matrix Factorization

**2. Content-based Recommendations**

* Classification (e.g. KNN or Naive Bayes)

1. Classifiers (e.g. ANN or Naive Bayes)

**What is Artificial intelligence?**

Artificial intelligence (AI) is an area of computer science that emphasizes the creation of intelligent machines that work and reacts like humans. Some of the activities computers with artificial intelligence are designed for include:

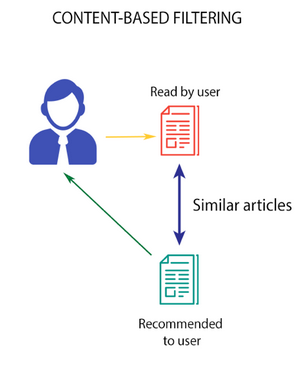
* Speech recognition.
* Learning.
* Planning.
* Problem solving.

**What is Collaborative Filtering?**

Like many machine learning techniques, a recommender system makes prediction based on users’ historical behaviors. Specifically, it’s to predict user preference for a set of items based on past experience. To build a recommender system, the most two popular approaches are Content-based and Collaborative Filtering.

**Content-based-** approach requires a good amount of information of items’ own features, rather than using users’ interactions and feedbacks. For example, it can be movie attributes such as genre, year, director, actor etc., or textual content of articles that can extracted by applying Natural Language Processing.

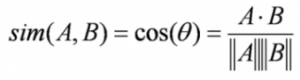
This algorithm recommends products which are similar to the ones that a user has liked in the past.



For example, if a person has liked the movie “Inception”, then this algorithm will recommend movies that fall under the same genre. But how does the algorithm understand which genre to pick and recommend movies from?

Consider the example of Netflix. They save all the information related to each user in a vector form. This vector contains the past behavior of the user, i.e. the movies liked/disliked by the user and the ratings given by them. This vector is known as the profile vector. All the information related to movies is stored in another vector called the item vector. Item vector contains the details of each movie, like genre, cast, director, etc.

The content-based filtering algorithm finds the cosine of the angle between the profile vector and item vector, i.e. cosine similarity. Suppose A is the profile vector and B is the item vector, then the similarity between them can be calculated as:



Based on the cosine value, which ranges between -1 to 1, the movies are arranged in descending order and one of the two below approaches is used for recommendations:

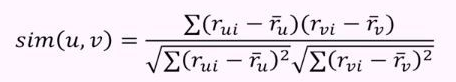
* **Top-n approach**: where the top n movies are recommended (Here n can be decided by the business)
* **Rating scale approach**: Where a threshold is set and all the movies above that threshold are recommended

Other methods that can be used to calculate the similarity are:

* **Euclidean Distance**: Similar items will lie in close proximity to each other if plotted in n-dimensional space. So, we can calculate the distance between items and based on that distance, recommend items to the user. The formula for the euclidean distance is given by:

2zjgw1x1.png

**Pearson’s Correlation**: It tells us how much two items are correlated. Higher the correlation, more will be the similarity. Pearson’s correlation can be calculated using the following formula:

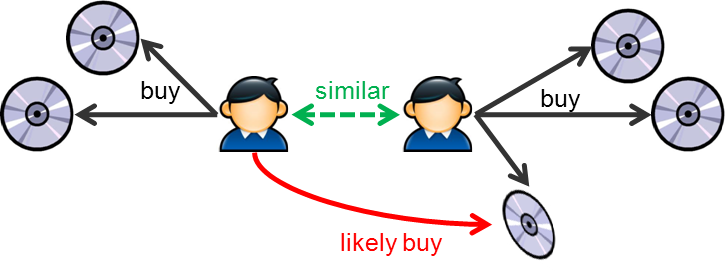


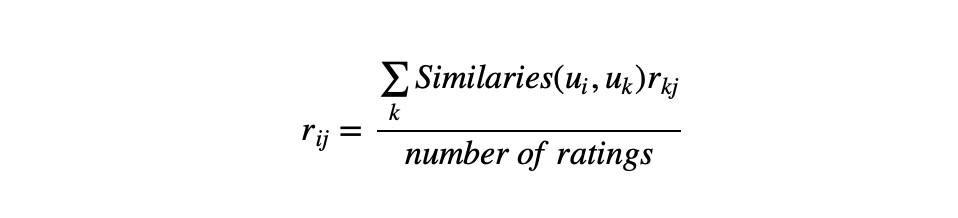
A major drawback of this algorithm is that it is limited to recommending items that are of the same type. It will never recommend products which the user has not bought or liked in the past. So if a user has watched or liked only action movies in the past, the system will recommend only action movies. It’s a very narrow way of building an engine.

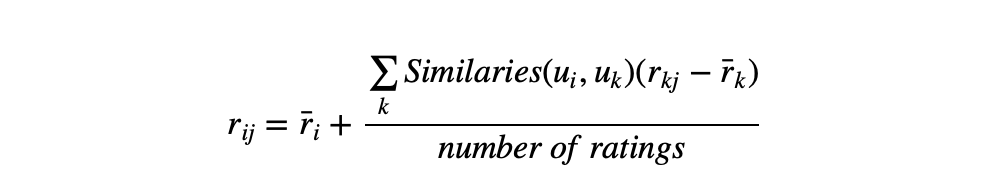
To improve on this type of system, we need an algorithm that can recommend items not just based on the content, but the behavior of users as well.

**Collaborative Filtering**, on the other hand, doesn’t need anything else except users’ historical preference on a set of items. Because it’s based on historical data, the core assumption here is that the users who have agreed in the past tend to also agree in the future.

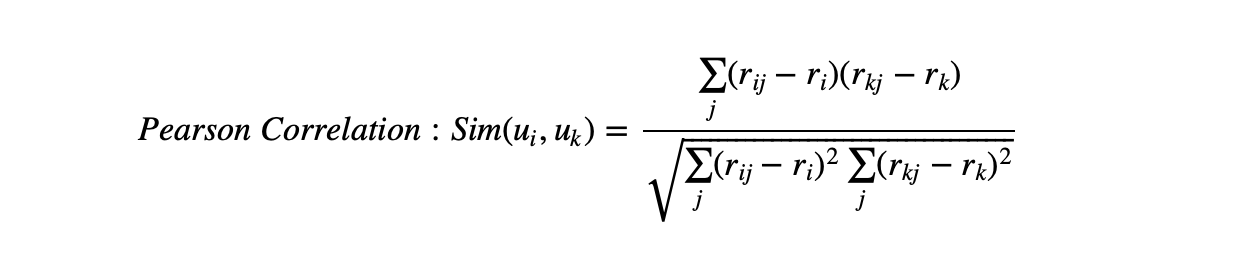
Nearest Neighborhood The standard method of Collaborative Filtering is known as **Nearest Neighborhood**algorithm. There are user-based CF and item-based CF. Let’s first look at **User-based CF**. We have an n × m matrix of ratings, with user uᵢ, i = 1, ...n and item pⱼ, j=1, …m. Now we want to predict the rating rᵢⱼ if target user i did not watch/rate an item j. The process is to calculate the similarities between target user i and all other users, select the top X similar users, and take the weighted average of ratings from these X users with similarities as weights.

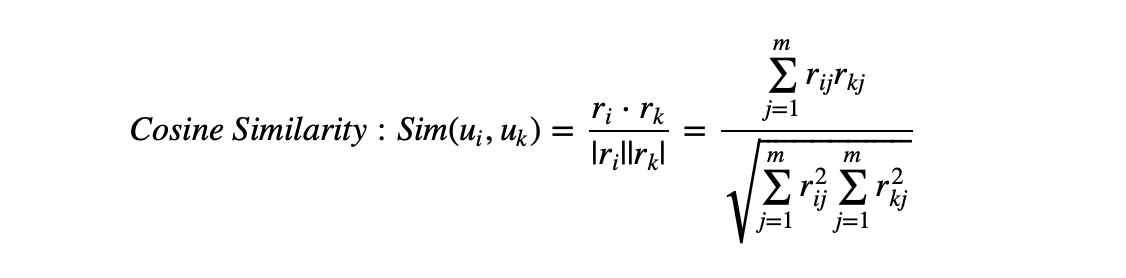
****

****While different people may have different baselines when giving ratings, some people tend to give high scores generally, some are pretty strict even though they are satisfied with items. To avoid this bias, we can subtract each user’s average rating of all items when computing weighted average, and add it back for target user, shown as below.

****

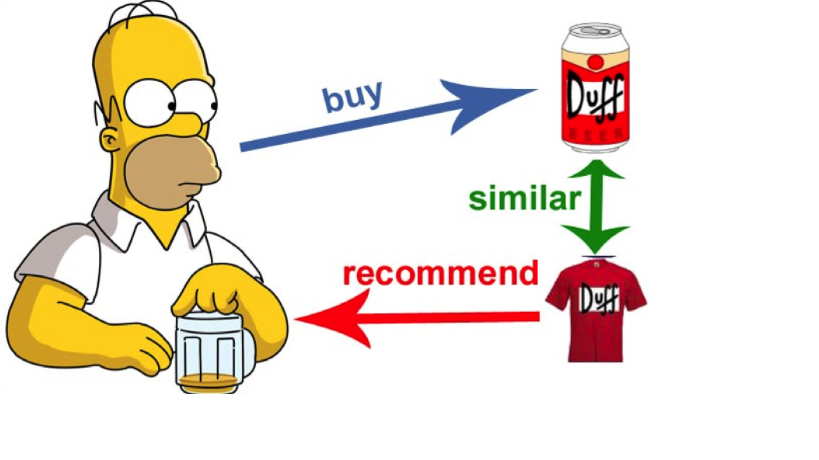
Two ways to calculate similarity are **Pearson Correlation** and **Cosine Similarity**.

****

****

Basically, the idea is to find the most similar users to your target user (nearest neighbors) and weight their ratings of an item as the prediction of the rating of this item for target user.

Without knowing anything about items and users themselves, we think two users are similar when they give the same item similar ratings . Analogously, for **Item-based CF**, we say two items are similar when they received similar ratings from a same user. Then, we will make prediction for a target user on an item by calculating weighted average of ratings on most X similar items from this user. One key advantage of Item-based CF is the stability which is that the ratings on a given item will not change significantly overtime, unlike the tastes of human beings.

****

There are quite a few limitations of this method. It doesn’t handle sparsity well when no one in the neighborhood rated an item that is what you are trying to predict for target user. Also, it’s not computational efficient as the growth of the number of users and products.

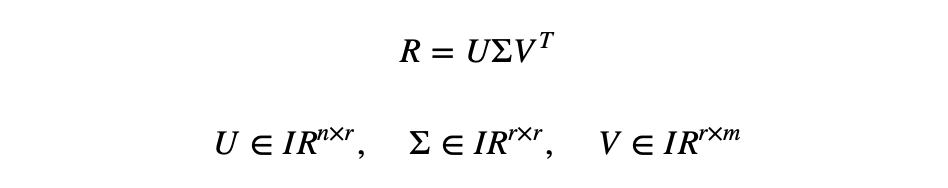
Matrix Factorization

Since sparsity and scalability are the two biggest challenges for standard CF method, it comes a more advanced method that decompose the original sparse matrix to low-dimensional matrices with latent factors/features and less sparsity. That is Matrix Factorization.

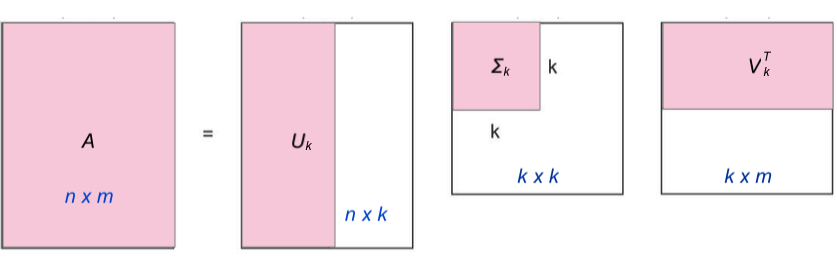
Beside solving the issues of sparsity and scalability, there’s an intuitive explanation of why we need low-dimensional matrices to represent users’ preference. A user gave good ratings to movie Avatar, Gravity, and Inception. They are not necessarily 3 separate opinions but showing that this users might be in favor of Sci-Fi movies and there may be many more Sci-Fi movies that this user would like. Unlike specific movies, latent features is expressed by higher-level attributes, and Sci-Fi category is one of latent features in this case. What matrix factorization eventually gives us is how much a user is aligned with a set of latent features, and how much a movie fits into this set of latent features. The advantage of it over standard nearest neighborhood is that even though two users haven’t rated any same movies, it’s still possible to find the similarity between them if they share the similar underlying tastes,

again latent features.

To see how a matrix being factorized, first thing to understand is **Singular Value Decomposition(SVD)**. Based on Linear Algebra, any real matrix R can be decomposed into 3 matrices U, Σ, and V. Continuing using movie example, U is an n × r user-latent feature matrix, V is an m × r movie-latent feature matrix. Σ is an r × r diagonal matrix containing the singular values of original matrix, simply representing how important a specific feature is to predict user preference.

****

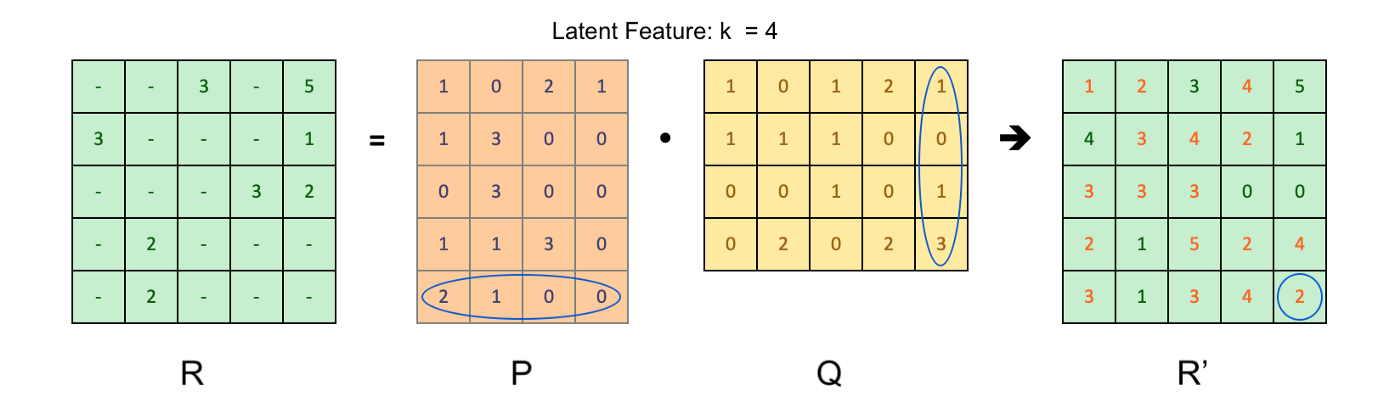
To sort the values of Σ by decreasing absolute value and truncate matrix Σ to first k dimensions ( k singular values), we can reconstruct the matrix as matrix A. The selection of k should make sure that A is able to capture the most of variance within the original matrix R, so that A is the approximation of R, A ≈ R. The difference between A and R is the error that is expected to be minimized. This is exactly the thought of Principle Component Analysis.

****

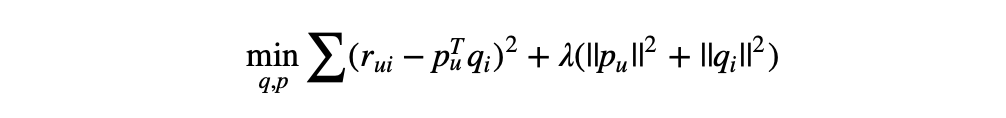
When matrix R is dense, U and V could be easily factorized analytically. However, a matrix of movie ratings is super sparse. Although there are some imputation methods to fill in missing values , we will turn to a programming approach to just live with those missing values and find factor matrices U and V. Instead of factorizing R via SVD, we are trying find U and V directly with the goal that when U and V multiplied back together the output matrix R’ is the closest approximation of R and no more a sparse matrix. This numerical approximation is usually achieved with **Non-Negative Matrix Factorization** for recommender systems since there is no negative values in ratings.

See the formula below. Looking at the predicted rating for specific user and item, item i is noted as a vector qᵢ, and user u is noted as a vector pᵤ such that the dot product of these two vectors is the predicted rating for user u on item i. This value is presented in the matrix R’ at row u and column i.

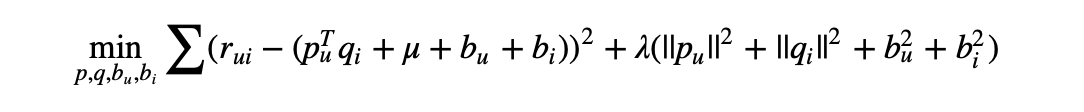
****

****

How do we find optimal qᵢ and pᵤ? Like most of machine learning task, a loss function is defined to minimize the cost of errors.

****

rᵤᵢ is the true ratings from original user-item matrix. Optimization process is to find the optimal matrix P composed by vector pᵤ and matrix Q composed by vector qᵢ in order to minimize the sum square error between predicted ratings rᵤᵢ’ and the true ratings rᵤᵢ. Also, L2 regularization has been added to prevent over fitting of user and item vectors. It’s also quite common to add bias term which usually has 3 major components: average rating of all items μ, average rating of item i minus μ(noted as bᵤ), average rating given by user u minus u(noted as bᵢ).

****

### ****Content-Based****

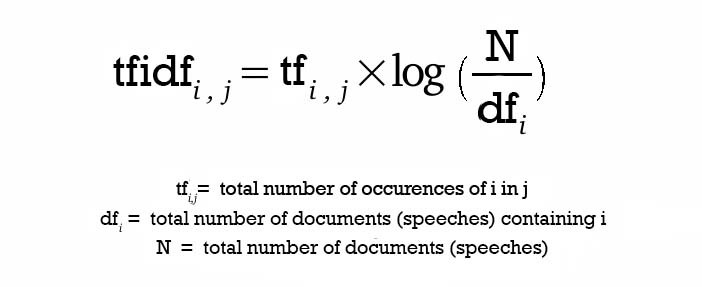
The Content-Based Recommender relies on the similarity of the items being recommended. The basic idea is that if you like an item, then you will also like a “similar” item. It generally works well when it’s easy to determine the context/properties of each item.

A content based recommender works with data that the user provides, either explicitly movie ratings for the MovieLens dataset. Based on that data, a user profile is generated, which is then used to make suggestions to the user. As the user provides more inputs or takes actions on the recommendations, the engine becomes more and more accurate.

The concepts of **Term Frequency (TF)** and **Inverse Document Frequency(IDF)** are used in information retrieval systems and also content based filtering mechanisms (such as a content based recommender). They are used to determine the relative importance of a document / article / news item / movie etc.

TF is simply the frequency of a word in a document. IDF is the inverse of the document frequency among the whole corpus of documents. TF-IDF is used mainly because of two reasons: Suppose we search for “**the results of latest European Soccer games**” on Google. It is certain that “**the**” will occur more frequently than “**soccer games**” but the relative importance of **soccer games** is higher than the search query point of view. In such cases, TF-IDF weighting negates the effect of high frequency words in determining the importance of an item (document).

Below is the equation to calculate the TF-IDF score:

****

After calculating TF-IDF scores, how do we determine which items are closer to each other, rather closer to the user profile? This is accomplished using the **Vector Space Model** which computes the proximity based on the angle between the vectors. In this model, each item is stored as a vector of its attributes (which are also vectors) in an **n-dimensional space** and the angles between the vectors are calculated to **determine the similarity between the vectors**. Next, the user profile vectors are also created based on his actions on previous attributes of items and the similarity between an item and a user is also determined in a similar way.

**KNN Algorithm**

K nearest neighbors or KNN Algorithm is a simple algorithm which uses the entire dataset in its training phase. Whenever a prediction is required for an unseen data instance, it searches through the entire training dataset for k-most similar instances and the data with the most similar instance is finally returned as the prediction.

KNN is often used in search applications where you are looking for **similar** items, like **find items similar to this one.**

Algorithm suggests that if **you’re similar to your neighoubers, then you are one of them**. For example, if apple looks more similar to peach, pear, and cherry (fruits) than monkey, cat or a rat (animals), then most likely apple is a fruit.

The k-nearest neighbors algorithm uses a very simple approach to perform classification. When tested with a new example, it looks through the training data and finds the k training examples that are closest to the new example. It then assigns the most common class label (among those k-training examples) to the test example.

****

k in KNN algorithm represents the number of nearest neighbor points which are voting for the new test data’s class.

If k=1, then test examples are given the same label as the closest example in the training set.

If k=3, the labels of the three closest classes are checked and the most common (i.e., occurring at least twice) label is assigned, and so on for larger ks.

## Naive Bayes algorithm

It is a [classification technique](https://courses.analyticsvidhya.com/courses/introduction-to-data-science-2/?utm_source=blog&utm_medium=6stepsnaivebayesarticle) based on [Bayes’ Theorem](https://en.wikipedia.org/wiki/Bayes%27_theorem" \t "_blank) 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:



Above,

* *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*.

**1.2Technology Used:**

Language: Python

Server : Apache HTTPD

IDE : Spider

Library : Numpy,Pandas,Scikit-Learn

**1.3Objective:**

The **purpose** was to evaluate how well an existing algorithm in a **recommender system** predicts **movie** ratings and get an indication of how the users perceive the **recommendations** given by the **system**.

Recommendation systems provide content for us by taking what other people recommend as well as our selections into account. Collaborative Filtering is a widely used solution for this problem which we make use of in our project.

1. **System Analysis**

System Analysis is the main feature of the software field. It can be said that it is a sole of any of the system. The analysis needs an expert supervision & the person should be minimum System Analyst with the experience of 5 to 7 yrs. One can use some of the different tools to analyze the system.

System Analysis refers to the process of examining a business situation analytically with the intent of improving it through alternate or new better procedures and methods. More precisely, it is a management technique that help us in designing a new system or improves an existing system. It was done keeping the following objectives in mind

Some of the tools we have used to analysis the problem is given below:

* Identify the Organization’s need
* Evaluate System Concept for Feasibility
* Perform Economic and Technical Feasibility
* Allocate functions to Hardware, Software, database and other System Elements

**ANALYSIS TOOLS:**

**Data Collection Tools**: We have collected the information regarding the current system & procedure which are going on. This collection helps us to understand the old systems know how.

**Charting Tools:** The graphical representation of system & activities help us to understand the minor problems and the flow procedures, which can become very helpful in analysis as well as to generate the new system requirements.

**Dictionary Tools:** This tool help us to maintain & record the data & description of system element through Data items, processes & data stores. After the use of tools we have to do the analysis of a system through main factors:

**Current system**: We study the working of current system, which help us to collect the benefits & limitations of the current system.

**Equipment’s**: Then we collect the information about the hardware requirement.

**Space**: In this we collect the information about the storage space needed in the new system.

**Procedure:** Which type of procedures are going on, this study is being useful to analyze that if all procedures are doing well or if there is a need for change.

**2.1 Requirement Analysis**

Requirement Analysis means studying or observing the current Business System to find how it works and where improvement can be made. It may include ways of capturing or processing data, producing information, or supporting management.

The First step of System Analysis is the identification of need. In this regard a series of interviews were conducted with authorized person of the institution. This was essential so as to know the processes that were being followed.

Requirements analysis is a software engineering task that bridges the gap between system level software allocation and software design.

**Figure 2.1 Requirement analysis**

**Analysis and a bridge between system engineering and software design**

Software requirements analysis may be divided into five areas of effort:-

1. Problem recognition.
2. Evaluation and synthesis.
3. Modelling.
4. Specification.
5. Review.

Initially, the analyst studies the system specification and the software project plan. Problem evaluation and solution synthesis is the next major area of effort for analysis.

Upon evaluating current problems and desired information (input and output) the analyst begins to synthesize one or more solutions.

During the evaluation and solution synthesis activity, the analyst creates models of the system in an effort to better understand data and control flow, functional processing and behavioral operation, and information content. The model serves as a foundation for software design and as the basis for the creation of a specification for the software.

**2.1.1 Modules:**

**Admin Module:**

**User Module:**

**2.1.2 Configuration:**

**Hardware Configuration:**

The section of hardware configuration is an important task related to the software development insufficient random access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application.

Processor: I3 and above

Processor speed: 1.4 GHz Onwards

System memory: 2 GB minimum 8 GB recommended

Cache size: 512 KB

RAM: 2 GB(Minimum)

Network card: Any card can provide a 100mbps speed

Network connection: UTP or Coaxile cable connection

Printer: Inkjet/Laser Color printer provides at least 1000 Dpi

Hard disk: 500Gb

Monitor: SVGA Color 15” 5

**Software Configuration:**

A major element in building a system is the section of compatible software since the software in the market is experiencing in geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system. This document gives a detailed description of the software requirement specification. The study of requirement specification is focused specially on the functioning of the system. It allow the developer or analyst to understand the system, function to be carried out the performance level to be obtained and corresponding interfaces to be established.

Operating system: Windows 10

Client Side: HTML, CSS, JavaScript.

##### 2.2 Preliminary Investigation

After clearly defining the requirements of the firm during Requirement Analysis then next step is the Preliminary Investigation that is done to determine the feasibility of the System. The purpose of the Preliminary Investigation is to evaluate Project requests.

The Data during Preliminary investigation was gathered through three Primary methods

###### Reviewing Organization documents -

I reviewed the Organization Documents such as forms, records, Reports, Manuals, Transaction detail etc. These Documents clearly defined the important steps involved in the working of the Firm. I also had a talk with concerned Manager and other related People to get information about the Firm.

###### On – Site Observation -

I observed the activities of the system directly. One purpose of On-Site observation is to get as close as possible to the Real System. I closely observed the Firm’s Environment, Workload on the System and the Users, the method of Work and the facilities provided by the Firm to the customer. The Physical layout of the Current System, its location and the Workflow was also observed.

###### Questionnaires –

Questionnaires were prepared and the employees were asked to answer them. The Questions were in the form of Multiple-Choice Questions and Open-ended Questions. This helped in getting an overview of the current mode of working.

The Preliminary Investigation helped me in achieving the following –

1. To obtain a detailed description of all the Records under Processing.
2. To obtain a detailed description of stored Data, their uses and inter-relationship between the data.
3. To differentiate each function within the System and identify Output Data Elements, Input Data Elements and Processing steps.
4. To identify the system processes and other functions which could be automated.

**Limitation of Existing System:**

The following are the limitation of the current working system:

1. Manual work consumes time and energy.
2. Large Number of calculation and human errors are generated.
3. Scope of limited.
4. No protection of data.

5. More Stationery is required

**Proposed System Description:**

The following points describe the requirement for the new system:

The working of the system will be fully Automated and online.

System should provide concrete security features by assigning login password, so that only the authorized person should access the software.

Various types of reports associated with different modules should be generated.

The system should have daily backup and restore facility to allow complete protection of data.

System should be able to handle extremely large volumes of data.

**2.3 Feasibility Study**

Feasibility of a project is being analyzed with in some framework. The most important factor is that if a project is feasible & desirable then it include in the schedule of the management that if the clearance should be give to it or not. Normally the assessment of feasibility depends upon some main factors like Technical, Economical and Operational.

The assessment of the “Database Management System” has the following facts.

**2.3.1 Operational Feasibility:**

* + 1. It seems that management is very much interested in the new system.
    2. The management & the user of the system are normally the same so there is no problem about confliction the management & users.
    3. Since all the work is going manually that is why there is no liking for the old system.
    4. The user and management both are ready to accept the new system, so they both show keen interest in giving the facts.

**2.3.2 Technical Feasibility:**

The technology seems sufficient to run the new system. The data holding facility is also seems sufficient because there is not enough transaction and it can be managed easily. It is also analyzed the hardware technology can be expanded if the data increases.

**2.3.3 Economical Feasibility:**

It is analyzed that it will benefit the company in terms of time saving as well as cost. It is also analyzed that cost of full system investigation is not much as it is not very much complex system.

**3 Software Engineering Paradigms**

**Software Process Models**

To solve actual problems in an industry setting, a software engineer or a team of engineers must incorporate a development strategy that encompasses the process, method and tools.

A process model for software engineering is chosen based on the nature of the project and application, the methods and tools to be used, and the control and deliverables that are required.

Among the various available Software Process Models (Waterfall models, spiral model Prototyping model, etc) the best suited for this project is the Prototyping Model.

To develop the system Proto Type Modal is applied as the company wanted to see the software development process and appearance of the software so that the idea of the functionally of the system can be understood. The software developed in three phases in first phase the dry proto type is developed in which after the analysis the screens are designed and no validation are performed and also no database functionality is incorporated. After the demonstration of dry proto type suggestions from the client are noted and the development process is moved to the second phase i.e. wet prototype the actual designed is incorporated, validations are performed and the software is submitted to the user for acceptance and testing and then after final submission of the software is produced with user manual.

**Prototyping Model**

Often, a customer defines a set of general objectives for software but does not identify detailed input, processing or output requirements. In other cases the developer may be unsure of the efficiency of an algorithm, the adaptability of an operating system, or the form that human –machine interaction should take, in these and many other situations, prototyping paradigm may offer the best approach.

The prototype can serve as “the first system’’. It is true that both the customer and developers like the prototyping paradigm. Users get a feel for the actual system and developers get to build something immediately. Yet prototyping can also be problematic for the following reasons:

1. The customer sees what appears to be a working version of the software that the prototype is held together “with chewing gum and a baling wire’’ unaware that in the rush to get it working we haven`t considered overall software quality or long term maintainability. When informed that the product must be rebuild so high level of quality can be maintained. Too often, software development management relents.
2. The developer often makes implementation compromises in order to get a prototype working quickly. An inappropriate operating system or programming language may be used simply because it is available and known, an efficient algorithm may become familiar with these choices and forget all the reasons why they were inappropriate. The less than ideal choice has now become an integral part of the system.

**4 Analysis Documents**

Document Analysis is a technique used to gather requirements during the requirements elicitation phase of a project.  It describes the act of reviewing the existing documentation of comparable business processes or systems in order to extract pieces of information that are relevant to the current project, and therefore should be consider projects requirements.

**4.1 Software Requirement Specification:**

The software requirements specification is produced at the culmination of the analysis task. The function and performance allocated to software as a part of system engineering are refined by establishing a complete information description a detailed functional and behavioral description, a indications of performance requirements and design constraints, appropriate validation criteria and other data pertinent to requirements. The National Bureau of standard IEEE and the US Department of Defense have all proposed candidate formats for software requirements specifications. For our purpose however the simplified outline presented may be used as a framework for the specification.

S. R.S. Of Our Project:

Introduction: It states the goals and objectives of the software descriptions in the context of the computer – based system actually, the introduction may be nothing more than the software scope of the planning document.

Information description

It provides a detailed description of the problem that the software must solve information contest and relationships, flow and structure are documented. Hardware, software and human interfaces are described for external system elements and internal software function. The Department of Administrative Reforms and Public Grievances aims to facilitate the pursuit of

excellence in governance. It gives the information about the citizen with grievances against public a institution and government organization viz. central Government Ministries / departments / Organizations / State Governments /UT Administrations and services to them. It forward all the information to citizens with grievance to the concerned Government organization within 15days under intimation to the petition.

Functional Description

It present description of each function required to solve the problem. A processing narrative is provided for each function, design constraints are stated and justified, performance characteristics are stated, and one or more diagrams are included to graphically represent the overall structure of the software and interplay among software function and other system elements under this topic, the functional working of the various modules where give the complete picture for the data flow from one location to other. There proper linking between them. The department of Administrative Reforms and Public Grievance aims to facilitate the pursuit of excellence in governance through promotion of:-

1. Improvements in Auction structures and processes.
2. Consumer friendly initiatives including redressal of grievances.
3. Documentation, incubation and discrimination of best practices.
4. Codification and simplification of procedures.
5. Internet Based.

Behavioral Description

This section of the specification examiners the operation of the software as a consequences of external events and internally generated control characteristics. In this topic may project shows the event flow diagram which tells us at what step what will be the result.

# 5 Code Efficiency

In software project we must keep a balance between the language used & the area of the application as well as database management system were actual data is being stores. There must be provision for those situations which are not meeting at the time of making the project for example the codes should be written keeping the aspect that software should compatible with others database or data can be migrated from outside. The information systems are designed with space & time complexity and cost saving in mind. The coding systems are methods in which conditions, words, ideas or relationship are expressed by a code. A code is an ordered collection of symbols designed to provide unique identification of an entity or attribute. It may be brief number, title or symbol. The main purpose of codes is to facilitate the identification and retrieval of times of information from the system.

There are many possible-coding structures. The main types of codes are described below:

**Classification Code**: Classification is the best described as the establishment of categories of entities, types and attributes in away that brings like or similar items together according to pre-determined relationships. A classification is by nature an order systematic structure. So the classification code places separate entities like events, people, or object into distinct groups called classes. A code is used to identify one class from another. Using the code the user classifies the event into one of several possible categories and records the code.

Classification codes vastly simplify the input process because only a single-digit code is required. The need for writing lengthy description or making judgments is eliminated.

**Function Code**: Function codes states the activities or work to be performed without spelling out all of the details in narrative statement.

It has been recognized that good coding style can overcome many of the deficiencies of a primitive programming language, while poor styles can defeat the intent of an excellent language. The goal of a good coding style is to provide easily understood, straight-forward and elegant code. The guidelines for coding includes:

* Use of a few standard control constants.
* Use of unconditional branching (go to) in a disciplined manner.
* Introduction of user-defined data types to model entities in the problem domain.
* Hiding of data structures behind access functions.
* Providing standard documentation prologues for each subprogram and compilation unit.
* Use of indentation, parenthesis, and blank spaces, blank lines, borders and comments to enhance readability.

**6 Code Optimization**

The main purpose of the optimization of code is to reduce redundancy of code, for this user has to write procedure function for repetitive. If in two situation the same type of interface needed then only one interface should be created and it may be called by passing different parameter.

The basic design model uses the analysis model as the framework for implementation. The analysis model captures the logical information about the system, while the design model must add details to support efficient information access.

During design optimization we must:

* Add redundant association to minimize access cost and maximize convenience.
* Rearrange the computation for greater efficiency.
* Saved derived attributes to avoid re-computation of complicated expression.
* During report generation we used an optimization technique of rearranging execution order for greater efficiency.
* Redundancy of data in the tables has been removed through Normalization technique. All the tables are designed in normalized form.
* Explicit and Implicit locks are used where needed.
* Explicit update: In this each derived attribute is defined in terms of one or more fundamental base object. We determined that which derived attributes are affected by each change to a fundamental attribute and inserted code into the update operation on the base object to explicitly update the derived attribute that depends on it.
* Optimization of Resource: During coding we have kept the strict vigil to use the same function again by writing the code in a module, so it keeps our coding minimize.

**7 Validation Checks**

In the complete project there are many validation and have been used in different forms in different ways.

## Following validation checks are there in this project which are to be checked in different modules.

* User ID should be unique , its entry can not be repeated.
* Various fields in various forms should not be left blank.

**8 Testing:**

In this section we have tested our system in different modes.

Basically the testing is done on different level like Unit testing, Integration testing , System testing and Acceptance testing. These different testing tests different types of faults.

We have tested in the following manner and whatever the fault we have got, fixed it immediately.

Clients Need Acceptance Testing

Requirements System Testing

Design Integration Testing

Code Unit Testing

Testing levels

**SYSTEM TESTING**

As the part of system testing we execute the program with the intent of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. The ultimate aim is quality assurance.

Tests are carried out and the results are compared with the expected document. In the case of

errorneous results, debugging is done. Using detailed testing strategies a test plan is carried out on each module. The various tests performed are unit testing, integration testing and user acceptance testing.

**Unit Testing:**

The software units in the system is are modules and routines that are assembled and integrated to perform a specific function. As a part of unit testing we executed the program for individual modules independently. This enables, to detect errors in coding and logic that are contained within each of the three module. This testing includes entering data that is filling forms and ascertaining if the value matches to the type and entered into the database. The various controls are tested to ensure that each performs its action as required.

**Integration Testing:**

Data can be lost across any interface, one module can have an adverse effect on another, sub functions when combined, may not produce the desired major functions. Integration testing is a systematic testing to discover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here the admin module, sec module and student module options are integrated and tested. This testing provides the assurance that the application is well integrated functional unit with smooth transition of data.

**User Acceptance Testing:**

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keep the records of applicants and

making changes to the details and password whenever required.

9 Implementation

A crucial phase in then systems life cycle is the successful implementation of the new system design. Implementation means converting a new system design into operation. This creating computer - compatible files, training the operating staff before the system is up and running. A major factor in conversion is not disrupting the functioning of the organization.

The implementation phase of software development is also concerned with translating design specification into source code. it is necessary to write source code and internal and documentation so that conformance of the code to its specification can be easily verified, and so that debugging, testing and modification are eased. This can be achieved by making the source code as clear and straight forward as possible. The implementation team should therefore be provided with-defined set of software requirements. An architectural design specification and a detailed design description.

User training is another important area, which is responsible for minimizing resistance to change and giving the new system a chance to provide it worth. Training aids, such as user-friendly manuals, data dictionary, and job performance aids that communication information about the new system; help screens provide the user with a good start on the new system.

Following conversion, it is desirable to review the performance of the system and to evaluate it against established criteria. Software maintenance follows conversion, which includes minor enhancements or corrections to problems that surface late in the system’s operating.

Conversion:

The objective of conversion is to put the system into operation while holding costs, risks and personal problems to a minimum. It involves three major steps:

* Creating computer-compatible files
* Training the operating staff
* Installing hardware/software

Procedures and documents are unique to the conversion phase. These are shown below:

* Conversion begins with a review of the project plan, the system test documentation, and the implementation plan. The parties involve are the user, the project team, programmers and operators.
* The conversion portion of the implementation plan is finalized and approved.
* Files are converted.
* Parallel processing between the existing and the new system is initiated.
* Results of computer runs and operators for the new system are logged on a special form.
* Assuming no problems, parallel processing is discounted. Implementation results are documented for reference.
* Conversion is completed. Plans for the post-implementation review are prepared. Following the review, the new subsystem is officially operational.

File Conversion:

File conversion involves capturing data and creating a computer file from existing files. There can be problems like staff shortage for loading data. Also, specialized training necessary tom prepares records in accordance with the new system specifications. In most cases, an outside agency performs this function for a flat rate. If a computerized system already exists, then copying existing files for a new system is a concern area. Program can be written to copy files intact for the new system and test programs on both systems. A file comparison program can be used for this purpose.

Many systems are prone to errors because of insufficient attention given to data entry control or protective features like audit control trails. These items must be part of the overall plan of for conversion. Before a data entry operator starts working, a data entry validation program can be written to keep track of wrong data entered.

At the time installation of the software it is required that oracle must be installed on the server of the company and it must contain the data files related to the software before implementing the developed system on all machines

Connected to the server . For this the exported data is imported to the client machine and the database is checked that all tables are properly imported to the client machine or not as the software is developed in oracle database and also going to be implemented in oracle is no other conversion is required.

A good audit control trail is the key to detecting fraud and errors in data entry.

User Training:

Analysis of user training focuses on two factors-user capabilities and the nature of the system being installed. The requirements of the system also range from very simple tasks like using a pocket calculator to complex tasks like learning to program a database system. Tasks that require the user to follow a well-defined, concrete, step-by-step procedure-require limited problem solving. For this, the training level and duration is basic and brief.

Hardware/Software Installation:

Adequate time and resources for installation of software and hardware must be allotted in the development schedule. User can be trained on the installation procedure. The detailed instructions can be a part of the user manual.

**10Evaluation:**

Evaluation of the system is performed to identify its strength and weaknesses.

The actual evaluation can occur along any of the following dimensions:

* **Operational Evaluation**: Assessment of the manner in which the system functions, including ease of use, overall reliability and level of utilization.
* **Organizational Impact**: Identification and measurement of benefit to the organization in such areas as financial concerns, operational efficiency and competitive impact.
* **User Manager Assessment**: Evaluation of attitudes of senior and user manager within the organization as well as end users.
* Development Performance: Evaluation of the development process in accordance with such yardsticks as overall development time and effort, conformance to budgets and standards and other project management criteria.

11 Maintenance

Software maintenance is used to describe the software engineering activities that occur following delivery of a software product to the customer.

The maintenance of existing software can account for over 60% of all effort expanded by a development organization, and the percentage continues to rise as more software is produced.

* Activities involved in maintenance of a software product include:
  + Analysis activities
  + Standards and guidelines
  + Design activities
  + Implementation activities
  + Supporting documents
* Configuration management is concerned with tracking and controlling of the work product that constitute a software product.
* Configuration management tools include:
  + Configuration management databases
  + Version control libraries
* Automated tools to support software maintenance include technical support tools and managerial support tools.
* Other automated tools include
* Text editors
* Debugging aids
* Linkage editor

Computer Maintenance covers a wide range of activities. Many activities performed during development of Airlines Reservation System for Global Airways to enhance the maintainability of it are as follows-

* Analysis Activities: The analysis phase of software development is concerned with determining customer requirements and constraints and establishing feasibility of the product. From maintenance view point, the most important activities that occur during analysis are establishing standards and guidelines for the project and the work products to ensure uniformity of the products; setting of milestones to ensure that the work products are produce on schedule; specifying quality assurance procedures to ensure development of high quality documents; identifying product enhancements that will most likely occur following initial delivery of the system; and estimating the response (personnel, equipment, floor space) required to perform the maintenance activities.
* Standards and Guidelines: Various types of standards and guidelines we developed to enhance the maintainability of our software.
* Design Activities: Design is concerned with developing the functional components, conceptual data structures and interconnection in a software system. The most important activity for enhancing maintainability during the design.

Automated tools to support software maintenance include technical support tools and managerial support tools. Tolls to support the technical aspects of software maintenance span the spectrum from analysis and design tools to implementation to debugging the testing tools. Automated tolls include text editors debugging aids, cross-reference generators, linkage editors, comparators, complexity metric calculators, and version control system and configuration management databases. Text editors permit a rapid, efficient modification of source programs, test data and supporting documents. Text editors can be used to insert the replace segments of source code, internal comments, test data and supporting documents; to systematically change the occurrences of an identifier or other textual strings to locate all references to a given identifier or other string of text; and to save both old and new version of a routine, test files or document. A syntax-directed text editor can ensure that all cross-references in the supporting documents are correctly updated.

Debugging aid provide traps, and traces assertion checking and history file aid in locating the causes of known errors. System-level-cross-reference generators provide cross-reference listing of procedure calls, statements usages and data references. Cross-references directories provide the calling structures of who calls whom and for where, and procedures names and statements numbers where formal parameters local variables and global variables are defined, set and used.

**12 System Security**

System must provide built-in feature of security and integrity of data without sage guard against unauthorized development access, fraud, even embezzlements, fire and natural disasters, a system could be so well vulnerable as to threaten the survival of the organization.

To do an educated job on security a system analyst must analyze the risk exposure and cost and specify measures such as password and encryption to provide protection. In addition backup copy of the software and recovery restart procedure must be available when needed. A disaster/recovery that has management support should also be prepared. Then no matter that the disaster, the firm can recover.

The strength behind integrity and success is ethics and professional standards of behavior. When ethics are compromised regardless of technology.

The main objective during the development of this system, which kept in mind, are

1. The various threats to system security and their defenses.
2. How to do risk analysis and specify measures
3. The importance of disaster recovery planning and how such a plan is initiated.
4. The meaning and importance of ethics in system development.

We also followed following methods for the development of this system for the security reasons.

1. Recovery of table structure with detail of full of modification date and time.
2. Password feature support by the software.
3. Detail of nomenclature of variable and acronyms use Assumption of system failure and recovery.
4. No unauthorized user can use the software without providing correct login and password.
5. The system should have daily backup and restore facility and restore facility to complete protection to data.

# 13 Cost Estimation

For a given set of requirement it is desirable to know how much it will cost to develop the software to satisfy the given requirements & how much time development will take.

**NEED:**

For a given set of requirements it is desirable to know how much it will cost to develop the software to satisfy the given requirements, and how much time development will take. These estimates are needed before development is initiated. The primary reason for cost and schedule estimation is to enable the client or developer to perform a cost benefit analysis and for project monitoring and control. A more practical use of these estimates is in bidding for software projects, where the developer must give cost estimates to a potential client for the development contract.

# The Cost of our project has been projected in the following manner-

# As we know that cost estimation for a project is due to the requirements for software, hardware resources and human resources. In hardware it includes computer time, terminal time & memory required for the project. Besides it software development is due to the human resources needed & the most cost estimation procedure focus on this aspect.

# We have found that we should follow the “COCOMO MODEL” will suit the best of calculation of estimation of our project.

# We have also projected our cost on the basis of person /month on for all Cost Drivers and Project Duration in months.

The COCOMO model defined for three classes of software projects are :-

1. Organic Model: - Relatively small, simple projects in which small teams with good application experience work to a set of less than rigid requirements.
2. Semidetached Model: - An intermediate (in size and complexity) software project in which teams with mixed experience levels must meet a mix of rigid and less than rigid requirements.
3. Embedded Model: - A software project that must developed within a set of tight hardware, software and operational constraints.

**14 Future Scope of the Project.**

This software is made according to the requirement of organization but expansion of modules can be easily included in the software.

This package can be used in internet environment that will make handling of different customers easier..

Following features can be added for enhancement:

* This software can be used for B2B and B2C sites.
* A proper revenue model for the system can be defined.
* With slight modifications it can be made to work or to fit into any such organization.

Future expandability and interconnectivity are the features, which are considered for scope in future.

**Software Development:**

The total span of developing the Database Management System that I got was 3 months. During this duration, I was involved in the different aspects of system development viz. System analysis, System design and development.

In this limited period of time, it was impossible for me to devote as much time as I would have wanted, on each of the phases.

Despite this limitation, the project gave me an opportunity to understand and learn about the various phases of the Software development life cycle.

It was a good opportunity for me to learn about the various phases of System development in depth, by means of documents, System Study and interacting with System users and further clarifications of doubts with the guide.

Since it was an individual activity, it helped me even more to strive harder to gain invaluable experience and knowledge to handle a software development project.

**Achievements:**

Working on a live project form its initial stages gave me an advantage of understanding the various needs of the customer and devising a plan to satisfy his requirements. Working on this project has benefited me in a number of ways. It gave me an opportunity to understand the users need, interact with them on personal basis and consequently develop a plan to meet their requirements.

The project gave me an experience of working in a professional set-up enriched with quality systems. I got an opportunity to learn and discover the various powerful features of Python, Machine Learning, Server in application development.

An important aspect of developing an information system is to understand, in depth the entire functionality and features of the system before planning to develop the proposed system.

**16Bibliography**

1. **Software Engineering- A Practitioner’s Approach**

–Roger S. Pressman.(McGraw Hill International Edition).

1. **Core Servlets and JSP : Volume One**

-by Martyn Hall and Larry Browrn(Person Publication)

1. **Java Complete Reference**

- by Herbert Schint. (McGraw Hill International Edition).

1. **An integrated approach to software engineering**

- byPankajJalote.

1. **Web Site’s:** Thesaurus.com, HinKhoj.com.
2. **Algorithms:** geeksforgeeks.com, Youtube.com.
3. **GitHub:** https://github.com/mission-peace/interview/wiki.
4. **Books:**Information Retrieval Systems Theory and Implementation BY Gerald Kowalski.