**Project Report**

**On**

**Product Recommendation System**

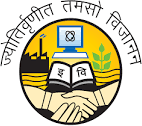
Submitted In the Partial Fulfillment Of The Requirements For The Award Of The Degree Of

**BACHELORS OF TECHNOLOGY**

**Computer Science Engineering**

OF

**GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY**



Under Supervision of: Submitted by:

**Ms. PRATIBHA GAUTAM** **VARUN GANDHI (00355102719)**



**Session 2019 – 2023**

**Mahaveer Swami Institute Of Technology**

Jatheri Rd, Barota Chowki, Jagdishpur, Haryana 131030

(Affiliated To Guru Gobind Singh Indraprastha University)  
(Sector–16 C, Dwarka, Delhi - 110078, India)

**Certificate**

This to certify that **VARUN GANDHI ,** university enrollment number 00355102719has worked under my supervision of to prepare this project report of **“PRODUCT RECOMMENDATION SYSTEM”.** The work embodied in this report is original and is standard expected of B-Tech student and has not been submitted in part or full to this or any university for the award of any degree diploma, He has completed all requirement of guidelines for research project and the work is fit for evaluation.

Date

Signature of HOD(IT)

Signature of Guide:

**DECLARATION**

I hereby declared that work presented in work entitled in partial fulfillment of the requirements for the award of degree of Bachelor of Technology submitted in “**MAHAVEER SWAMI INSTITUTE OF TECHNOLOGY”** affiliated to **GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY** is a authentic record of my own work carried about under the supervision of **Ms Pratibha Gautam.**

**DATE:**

**ACKNOWLEDGEMENT**

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project. Whatever we have done is only due to such guidance and assistance and we would not forget to thank them.

We express our deepest gratitude to **Ms. Pratibha Gautam, MVSIT,** *Mahaveer Swami Institute Of Technology***,** for extending her inevitable and valuable support to us.

I owe my profound gratitude to our **guide** for the project **Ms. Pratibha Gautam mam** who took a keen interest in our project work and guided us all along till the completion of our project work by providing all the necessary information for developing a good system. She has taken immense effort and pain to go through the project and make necessary corrections as and when needed.

I am thankful to and fortunate enough to get constant encouragement, support, and guidance from our faculty members without whom this project would have been a distant reality. We would also like to extend our sincere regards to our families, and well-wishers for their timely support.

VARUN GANDHI (00355102719)

**ABSTRACT**

Recommendation Systems help users to find information and make decisions where they lack the required knowledge to judge a particular product. Also, the information datasets available can be huge and recommendation systems help in filtering this data according to users‟ needs. Recommendation systems can be used in various different ways to facilitate its users with effective information sorting. For a person who loves reading, this paper presents the research and implementation of a Recommendation System for a Newsreader Application using Android Platform. The Newsreaders Application proactively recommends news articles as per the reading habits of the user, recorded over a period of time and also recommends the currently trending articles. Recommendation systems and their implementations using various algorithms is the primary area of study for this project. This research paper compares and details popular recommendation algorithms viz. Content based recommendation systems, Collaborative recommendation systems etc. Moreover, it also presents a more efficient Hybrid approach that absorbs the best aspects from both the algorithms mentioned above, while trying to eliminate all the potential drawbacks observed.

**TABLE OF CONTENT**

|  |
| --- |
| **Declaration** |
| **Acknowledgment** |
| 1. **Introduction & Objectives of the Project** |
| * 1. Introduction |
| * 1. Objectives |
| * 1. Benefits |
| * 1. Drawback of Current Manual-System |
| * 1. Proposed System |
| * 1. Need |
| 1. **System Study** |
| * 1. Preliminary Investigation |
| * 1. System Development Life Cycle |
| 1. **System Analysis** |
| * 1. Importance of Computerized System |
| * 1. Principle of System Analysis |
| **4. System Design** |
| 4.1 System Design |
| 4.2 Data Modeling |
| 4.3 Data Tables and Context level DFD |
| **5. System Development** |
| * 1. Source Code |
| * 1. Code Efficiency |
| 1. **Testing** |
| 6.1 Testing Phase |
| 6.2 Level of Testing |
| 6.3 Verification and Validation (V&V) |
| **7. Security and Validations** |
| **8. Scope of Future Application** |
| **9. Conclusion** |
| **10. Bibliography** |
| **11. Glossary** |

**INTRODUCTION**

Recommendation systems are a very popular and effective paradigm in retail business. With a recommendation system, shoppers can find items they like with less effort. Furthermore, they are presented with items they’ve never thought of buying, but which actually suits their needs. Therefore, as a part of one of our projects for a footwear company, we developed BE-terna’s customer behaviour analysis platform, which provides clients with customer profiling and recommendation system., recommendation systems that use algorithms, to find similar items and similar customers, based on their behaviour, and recommend items which the specific customer should like**.**

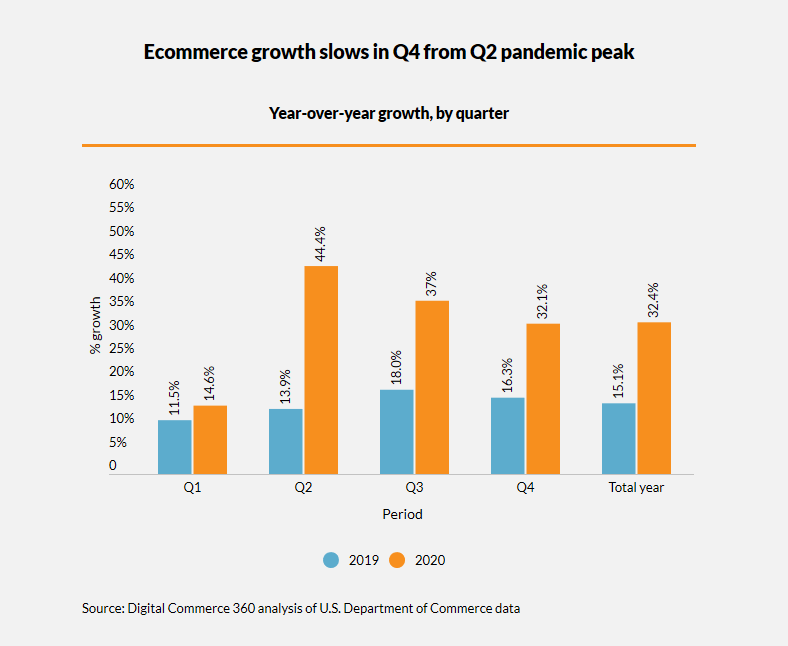
**BENEFITS**

Customer behavior analysis focuses on understanding the type of customers; what do they like, what do they not like, what their pattern of interaction with the items is, customer value, etc.**If we manage to model these aspects of a customer, we can anticipate their future needs.**

The main benefits coming from customer behaviour analysis system are:

1. **Boost in sales,**
2. **Better understanding of customers and**
3. **Long tail strategies.**

**The main benefit can be put in just three words - boost in sales.** According to McKinsey, 35% of all Amazon purchases, and 70% of Netflix purchases, are driven by their recommendation systems, and the start of using recommendations significantly boosted their sales. Furthermore, during COVID-19 pandemic, many retailers went online, digitalised their businesses, and changed their business culture to adapt to the new and everchanging circumstances. According to reports, in 2020 the growth of ecommerce sales in US alone was more than 30% (Figure 1). This provides a huge amount of online data for potential exploration, and use in building machine learning systems.



**The second benefit comes from the better understanding of customers.** This is the part where customer profiling comes in. By profiling customers, we can better understand their behaviour and, consequently, better understand their needs, or in other words, meet their needs, which finally can be rewarded with higher customer satisfaction and loyalty. Other than increased customer satisfaction, we can easily create automated marketing campaigns, and personalise them based on customer analysis.

**The next benefit is a much better strategy for long tail items.**The term long tail item, refers to niche and hard-to-find items that are very specific and unique, and usually only have a small group of people looking for them. From a customer’s perspective, tools such as recommendation systems, allow them to find products outside their immediate area, and items they, otherwise, would not have had access to. From a supplier’s perspective, if they hold items in a warehouse, hidden from the customers that would like them, this strategy could become very profitable.

## Customer profiling

Customer profiling is done, by segmenting customers into clusters, which exemplify similar behaviour based on different parameters, derived from data such as a number of items bought, a value of bought items, a number of items returned, types of items bought, etc. Segmentation is essentially grouping by behavioural similarity. The input to segmentation is parameters for which we would like to find similar groups. The result is the number of similar groups, as well as rules by which a certain property falls into each group. Additionally, we can assign values to each segment per segmentation, and use it as a numerical weight (score) of the segment. Different segmentations can be combined as equals, or according to assigned importance weights, to provide customer value. More segmentations per property will give us more fine-grained profiles of the property, since we will get different views on it, and, consequently, be more insightful about the property (customer, item, brand, store, etc.). The customer profile is therefore, the most important parameter for further recommendations.

**Classification of Recommendation Systems**

(1) **User-Based:** As the name suggests, this approach is based on users rather than the characteristics of items that a user prefers. It assumes that similar users have similar preferences. Similarity between users is based on various attributes like age, sex, and occupation etc.; which normally tend to define an individual‟s preferences and behavior. “K” most similar users (neighbors) to the given user are selected based on the factors as stated above. According to the definition of Collaborative Filtering, the historic information of the preferences of similar neighbors to a given user is studied and the top-N items in the list, which have not yet been purchased or used by the given user, are returned as recommendations. The problems faced by user-based collaborative filtering systems are Scalability, Sparsity, and First-Rater Problem or Cold Start

**(2) Item Based:** Item-based approach considers the similarities between items used by users in the same group. Thus, similar items used by K-neighbors of a given user are used to generate recommendations using cosine similarity, excluding the items already used by the user.

Fig\_1.0



**Data Collection:**

Explicit Data Collection includes the following:

* Collecting user ratings for a product in terms of most liked items to least liked items
* Presenting two items to an user and asking him/her to choose the better of them
* Analyzing the list of products liked by an user, this data can be collected from various
* advertisements of items

**Project Design and Scope**

The Project aims at including the features of both Content-Based and Collaborative-Based Filtering Approaches, considering the advantages of the Hybrid Approach and tries to improvise over the shortcomings of both the approaches to provide proactive recommendations to a newsreader. The main components of the system are:

(1) Parser

(2) Recommendation Engine

(3) Database

(4) All Client

**SYSTEM REQUIREMENT SPECIFICATION**

**HARDWARE**

Processor : Pentium 2.4 GHz or above

Memory : 256 MB RAM or above

Cache Memory : 128 KB or above

Hard Disk : 3 GB or above [at least 3 MB free space required]

Pen Drive : 5 GB

Printer : Laser Printer

**SOFTWARE**

Operating System : Windows 7/8/10/11.

Font-End Tool : HTML CSS, JAVA/JSP, ,Java Script

Back-End : My Sql

Server : Tomcat 7/9

**Methodologies**

The proposed system makes use of Pearson’s correlation to implement User based collaborative filtering, and context, Synonym Finder to implement Context based filtering techniques to generate recommendations for the active user.

Recommendation systems (RS) are the most important factor in ecommerce and several applications; the RS utilizes data mining techniques and tools to predict user’s preference by utilizing their previous shopping information’s and selecting products among the tremendous amount of available items for the users . A recommendation system obtains the interest and preference of consumer and performs recommendations accordingly, so it is broadly used in every ecommerce websites. Recommendation Systems have the potential to help and improve the quality of the decisions consumers make while searching for and selecting products online. With the tremendous growth of ecommerce, due to the huge nature of the information, the data overload problem has created. So, the users are not able to effectively get items on the ecommerce websites or web. In the electronic world, RS has introduced the need for information filtering techniques that are use to help users by filter out information in which they are interested in. Recommendation systems are one of the approaches applied for the ecommerce recommendation system which is based on providing possible items of interest to a user instead of the user to go searching for them. RS changed the way as the websites communicate with their users. Instead of providing a static feel for the users, in product searching, this provides potential suggestions which increases communication to provide a higher experience. This also reduces the problem of stock unavailability. RS recognize recommendations autonomously for individual users based on their previous search, shopping histories, profiles, rating and other reviews given to item and this also considers the other users behavior too. List of recommendations in given in fig 1.0. The branches of RS are described below.

**PRELIMINARY INVESTIGATION**

System development, a process consisting of two major steps of system analysis and design, start when management or sometimes system development personnel feel that a new system or an improvement in the existing system is required. The system development life cycle is classically thought of as the set of activities that analysts, designers and users carry out to develop and implement an information system. The system development life cycle consists of the following activities:

* Preliminary investigation
* Determination of system requirements
* Design of system
* Development of software
* System testing
* Implementation, evaluation, and maintenance

A request to take assistance from information system can be made for many reasons, but in each case someone in the organization initiates the request is made, the first system activity the preliminary investigation begins. This activity has three parts:

1. Request clarification
2. Feasibility study
3. Request approval

Request clarification: Many requests from employees and users in the organizations are not clearly defined, therefore it becomes necessary that project request must be examined and clarified properly before considering systems investigation.

**SYSTEM DEVELOPMENT LIFE CYCLE**

Systems are created to solve problems. One can think of the systems approach as an organized way of dealing with a problem. In this dynamic world, the subject System Analysis and Design (SAD), mainly deals with the software development activities.

**DEFINING A SYSTEM**

A collection of components that work together to realize some objective forms a system. Basically there are three major components in every system, namely input, processing and output.In a system the different components are connected with each other and they are interdependent. For example, human body represents a complete natural system. We are also bound by many national systems such as political system, economic system, educational system and so forth. The objective of the system demands that some output is produced as a result of processing the suitable inputs.

**SYSTEM LIFE CYCLE**

System life cycle is an organizational process of developing and maintaining systems. It helps in establishing a system project plan, because it gives overall list of processes and sub-processes required for developing a system.

System development life cycle means combination of various activities. In other words we can say that various activities put together are referred as system development life cycle. In the System Analysis and Design terminology, the system development life cycle means software development life cycle.

**Following are the different phases of software development cycle:**

* System study
* Feasibility study
* System analysis
* System design
* Coding
* Testing
* Implementation
* Maintenance

**The Different Phases Of Software Development Life Cycle Are Shown Below.**

**PRELIMINARY INVESTIGATION**

**DETERMINATION OF REQUIREMENTS**

**REVIEW RUNNING SYSTEM AND SYSTEM MAINTENANCE**

**SYSTEM IMPLEMENTATION**

**SYSTEM TESTING**

**DEVELOPMENT OF SOFTWARE AND CODING**

**DESIGN OF SYSTEM**

**DEVELOPMENT OF PROTOTYPE SYSTEM**

**FIG: SHOWING GENERAL LIFE CYCLE PROCESS AND PERCENTAGE OF TIME DEVOTED**

A system analysis is a separation of a substance into parts for study and their implementation and detailed examination.

Before designing any system it is important that the nature of the business and the way it currently operates are clearly understood. The detailed examination provides the specific data required during designing in order to ensure that all the client's requirements are fulfilled. The investigation or the study conducted during the analysis phase is largely based on the feasibility study. Rather it would not be wrong to say that the analysis and feasibility phases overlap. High-level analysis begins during the feasibility study. Though analysis is represented as one phase of the system development life cycle (SDLC), this is not true. Analysis begins with system initialization and continues until its maintenance. Even after successful implementation of the system, analysis may play its role for periodic maintenance and up gradation of the system.

One of the main causes of project failures is inadequate understanding, and one of the main causes of inadequate understanding of the requirements is the poor planning of system analysis.

Analysis requires us to recall the objectives of the project and consider following three questions:

* What type of information is required?
* What are the constraints on the investigation?
* What are the potential problems that may make the task more difficult?

4.1 SYSTEM DESIGN

The design document that we will develop during this phase is the blueprint of the software. It describes how the solution to the customer problem is to be built. Since solution to complex problems isn’t usually found in the first try, iterations are most likely required. This is true for software design as well. For this reason, any design strategy, design method, or design language must be flexible and must easily accommodate changes due to iterations in the design. Any technique or design needs to support and guide the partitioning process in such a way that the resulting sub-problems are as independent as possible from each other and can be combined easily for the solution to the overall problem. Sub-problem independence and easy combination of their solutions reduces the complexity of the problem. This is the objective of the partitioning process. Partitioning or decomposition during design involves three types of decisions: -

Define the boundaries along which to break;

Determine into how money pieces to break; and

Identify the proper level of detail when design should stop and implementation should start.

Basic design principles that enable the software engineer to navigate the design process suggest a set of principles for software design, which have been adapted and extended in the following list:

Free from the suffer from "tunnel vision." A good designer should consider alternative approaches, judging each based on the requirements of the problem, the resources available to do the job.

The design should be traceable to the analysis model. Because a single element of the design model often traces to multiple requirements, it is necessary to have a means for tracking how requirements have been satisfied by the design model.

The design should not repeat the same thing. Systems are constructed using a set of design patterns, many of which have likely been encountered before. These patterns should always be chosen as an alternative to reinvention. Time is short and resources are limited! Design time should be invested in representing truly new ideas and integrating those patterns that already exist.

The design should "minimize the intellectual distance" between the software and the problem as it exists in the real world. That is, the structure of the software design should (whenever possible) mimic the structure of the problem domain.

The design should exhibit uniformity and integration. A design is uniform if it appears that one person developed the entire thing. Rules of style and format should be defined for a design team before design work begins. A design is integrated if care is taken in defining interfaces between design components.

The design activity begins when the requirements document for the software to be developed is available. This may be the SRS for the complete system, as is the case if the waterfall model is being followed or the requirements for the next "iteration" if the iterative enhancement is being followed or the requirements for the prototype if the prototyping is being followed. While the requirements specification activity is entirely in the problem domain, design is the first step in moving from the problem domain toward the solution domain. Design is essentially the bridge between requirements specification and the final solution for satisfying the requirements.

The design of a system is essentially a blueprint or a plan for a solution for the system. We consider a system to be a set of components with clearly defined behavior that interacts with each other in a fixed defined manner to produce some behavior or services for its environment. A component of a system can be considered a system, with its own components. In a software system, a component is a software module.

The design process for software systems, often, has two levels. At the first level, the focus is on deciding which modules are needed for the system, the specifications of these modules, and how the modules should be interconnected. This is what is called the system design or top-level design. In the second level, the internal design of the modules, or how the specifications of the module can be satisfied, is decided. This design level is often called detailed design or logic design. Detailed design essentially expands the system design to contain a more detailed description of the processing logic and data structures so that the design is sufficiently complete for coding.

Because the detailed design is an extension of system design, the system design controls the major structural characteristics of the system. The system design has a major impact on the testability and modifiability of a system, and it impacts its efficiency. Much of the design effort for designing software is spent creating the system design.

The input to the design phase is the specifications for the system to be designed. Hence, reasonable entry criteria can be that the specifications are stable and have been approved, hoping that the approval mechanism will ensure that the specifications are complete, consistent, unambiguous, etc. The output of the top-level design phase is the architectural design or the system design for the software system to be built. This can be produced with or without using a design methodology. Reasonable exit criteria for the phase could be that the design has been verified against the input specifications and has been evaluated and approved for quality.

A design can be object-oriented or function-oriented. In function-oriented design, the design consists of module definitions, with each module supporting a functional abstraction. In object-oriented design, the modules in the design represent data abstraction (these abstractions are discussed in more detail later). In the function-oriented methods for design and describe one particular methodology the structured design methodology in some detail. In a function- oriented design approach, a system is viewed as a transformation function, transforming the inputs to the desired outputs. The purpose of the design phase is to specify the components for this transformation function, so that each component is also a transformation function. Hence, the basic output of the system design phase, when a function oriented design approach is being followed, is the definition of all the major data structures in the system, all the major modules of the system, and how the modules interact with each other. Once the designer is satisfied with the design he has produced, the design is to be precisely specified in the form of a document. To specify the design, specification languages are used. Producing the design specification is the ultimate objective of the design phase. The purpose of this design document is quite different from that of the design notation. Whereas a design represented using the design notation is largely to be used by the designer, a design specification has to be so precise and complete that it can be used as a basis of further development by other programmers. Generally, design specification uses textual structures, with design notation helping in understanding.

**DATA MODELING**

**Introduction to data dictionary:**

Data dictionaries are an integral component of structured analysis, since data flow diagrams by themselves do not fully describe the subject of the investigation . The data flow diagrams provide the additional details about the project/system.

**Data Dictionary (Definition**):

A data dictionary is a catalog- a repository- of the elements in a system. These elements center on the data and the way they are structured to meet user requirements and organization needs. A data dictionary consists of a list of all the elements composing the data flowing through a system . The major elements are data flows , data stores , and processes. The data dictionary stores details and descriptions of these elements.

**DATA TABLES**

**Login Table**

|  |  |  |
| --- | --- | --- |
| Field | Type | constraint |
| Username | Char(30) | Not Null |
| Password | Char(30) | Not Null |

**Admin Table**

|  |  |  |
| --- | --- | --- |
| Field | Type | constraint |
| Username | Char(30) | Not Null |
| Password | Char(30) | Not Null |

**Rating table**

|  |  |  |
| --- | --- | --- |
| Field | Type | constraint |
| Product Name | Char(30) | Not Null |
| Product Rating | Number | Not Null |

**Customer Table**

|  |  |  |
| --- | --- | --- |
| Field | Type | constraint |
| Name | Char (30) | Not Null |
| Address | Char (30) | Not Null |
| Email | Char (50) | Not Null |
| Phone | Number | Not Null |
| Password | Number | Not Null |

**Booking Product table**

|  |  |  |
| --- | --- | --- |
| Field | Type | constraint |
| Customer Name | Char (30) | Not Null |
| Address | Char (30) | Not Null |
| Card Details | Number | Not Null |
| Pin code | Number | Not Null |
| Email | Char (50) | Not Null |
| Phone | Number | Not Null |
| Quantity | Number | Not Null |
| Product Name | Char(50) | Not Null |
| Security Code | Number | Not Null |
| Total Cost | Number | Not Null |

**Product Upload table**

|  |  |  |
| --- | --- | --- |
| Field | Type | constraint |
| P Name | Char (30) | Not Null |
| P code | Number | Not Null |
| P company | Char (30) | Not Null |
| Cost | Char (30) | Not Null |
| P description | Char (50) | Not Null |

## CONTEXT LEVEL DFD

Product Recommendation

\

System

1. **Rating**
2. **Product**

**2. User**

**5. feedback**

**4. Admin**

## SCHEDULING

Scheduling of a software project does not differ greatly from scheduling of any multi- task engineering effort. Therefore, generalized project scheduling tools and techniques can be applied with little modification to software projects.

Program evaluation and review technique (PERT) and critical path method (CPM) are two project scheduling methods that can be applied to software development. Both techniques are driven by information already developed in earlier project planning activities.

# Estimates of Effort

* A decomposition of the product function.
* The selection of the appropriate process model and task set.
* Decomposition of tasks.

Interdependencies among tasks may be defined using a task network. Tasks, sometimes called the project Work Breakdown Structure (WBS) are defined for the product as a whole or for individual functions.

Both PERT and CPM provide quantitative tools that allow the software planner to (1) determine the critical path-the chain of tasks that determines the duration of the project; (2) establish "most likely" time estimates for individual tasks by applying statistical models; and (3) calculate "boundary times" that define a time window" for a particular task.

Boundary time calculations can be very useful in software project scheduling. Slippage in the design of one function, for example, can retard further development of other functions. It describes important boundary times that may be discerned from a PERT or CPM network: (I) the earliest time that a task can begin when preceding tasks are completed in the shortest possible time, (2) the latest time for task initiation before the minimum project completion time is delayed, (3) the earliest finish-the sum of the earliest start and the task duration, (4) the latest finish- the latest start time added to task duration, and (5) the total float-the amount of surplus time or leeway allowed in scheduling tasks so that the network critical path maintained on schedule. Boundary time calculations lead to a determination of critical path and provide the manager with a quantitative method for evaluating progress as tasks are completed.

Both PERT and CPM have been implemented in a wide variety of automated tools that are available for the personal computer. Such tools are easy to use and take the scheduling methods described previously available to every software project manager.

**SOURCE CODES**

**About us**

<%@ page contentType="text/html; charset=iso-8859-1" language="java" import="java.sql.\*" errorPage="" %>

<%@page import="java.io.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title></title>

<style type="text/css">

<!--

.style7 {color: #000000}

.style9 {

font-size: 12px;

color: #FFF;

}

.style10 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-weight: bold;

}

.style11 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-size: 10;

text-align: right;

}

.style12 {

font-size: 10px;

font-family: Verdana, Arial, Helvetica, sans-serif;

font-weight: bold;

}

.style13 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-size: 12px;

}

.style14 {

font-family: "Courier New", Courier, monospace;

color: #000000;

}

em {

color: #FFF;

text-align: left;

}

#form1 table tr td table tr td {

text-align: left;

font-size: small;

color: #00F;

}

-->

</style>

<style type="text/css">

body {

background-image: url();

background-repeat: no-repeat;

}

#form1 table tr td table tr td p a {

color: #FFF;

}

#form1 table tr td p {

}

#form1 table tr td h1 {

color: #FFF;

}

</style>

</head>

<body>

<%@include file="header.jsp" %>

<form id="form1" name="form1" method="post" action="process.jsp">

<table rules="none" width="90%" align="center" cellpadding="4" cellspacing="4">

<tr>

<td height="260" valign="top"><p><a href="index.jsp"></a></p><p>ABOUT</p>

<p>In the under two decades since it was established, the has become distinguished for its focus on 20th century Indian art. Its museum-like collections range from the early-moderns to the moderns, including all the masters and senior artists of the movement all the way to more recent contemporary art.</p>

<p>The DAG collection, therefore, is an important historical as well as chronological time-line of the entire movement of modern art in the country. As one of the largest such repositories of Indian modern art anywhere in the world, it is one of the most critical links for anyone with an interest in Indian art &ndash; whether art-lovers, collectors, academicians, scholars, researchers or investors. At its very inception, the DAG motto was to establish the great art treasures of 20th century India, from its remotest parts to those in its better established art and culture hubs. This meant re-discovering the great artists of the century, many with proven track records who had been allowed to lapse into anonymity. It meant extensive travels across the country, to re-familiarise with artists from all regions and centres, to understand their creative genius and pay homage to the extensive oeuvres that flourished through one of the country's most creatively diverse period.</p>

<p>Entire collections were traced, compiled, restored, archived and made available for posterity. As a result, the extensive inventory is recognised for its quality of artworks. Documentation processes across various genres have resulted in some iconic exhibitions, known for their breadth of scale and depth of research. Extensive exhibition catalogues and books &ndash; part of its ongoing efforts on shedding new light on, of course, the well-established artists and their genres, but also on the lesser-known but equally talented painters and sculptors &ndash; is a measure of the seriousness of DAG's effort of concentrating not just on the names of Indian artists familiar to all art-lovers, but also those artists who deserve their space on the same firmament.With over four hundred artists in its collection, the is committed to bringing together the most seminal works of artists from Bengal's golden period of nationalist art, to the Progressives, all of whom are represented at DAG, to their associates, the important and diverse schools and groups spread throughout India, to artists of more recent vintage with their own, established track records.</p>

<p>, unlike most others, owns its entire, exhaustive collection. It is, simply, the most important institution for 20th century Indian art not just in India but around the world. The was started by Rama Anand in 1993. Ashish Anand, director, DAG, took over the gallery from his mother in 1996 to give it its current shape, volume and credibility.</p>

<br />

<h1>Director's Profile</h1>

<p><strong>Ashish Anand</strong></p>

<p>Ashish Anand, a young entrepreneur and director of the (DAG), has one of the finest collections of 20th century Modern and Contemporary Indian art. With an eye for excellence, he strategized an alternate plan to enhance the receptivity of modern Indian art both nationally and internationally, making it more inclusive, diverse and bringing merit to several significant artists. Refusing to play safe, Anand pioneered the strategy of focusing on the relatively lesser-known artistic talents of the country that awaited proper exposure and visibility.</p>

<p>Under his leadership, DAG spans a spectacular collection which is a visual feast with artworks from all decades of the 20th century- artists from the pre-independence era such as Raja Ravi Varma, Abanindranath Tagore, Nandalal Bose, Rabindranath Tagore, Binode Behari Mukherjee, Chittaprosad, Gopal Ghose to precursors of the post-independent decades, namely M.F.Husain, S.H.Raza, F.N.Souza, Satish Gujral or then the later generation comprising works of Himmat Shah, Ambadas, Gogi Saroj Pal, Sunil Das, Jyoti Bhatt, Amitava Das and several brilliant names.</p>

<p>Over the years, Anand's vision for the gallery has gone beyond its conventional role. His handsome collection of art has in due course triggered the need for a deeper understanding of the context and circumstances in which various genres of art were created. This became a full-fledged pursuit that culminated into the DAG Documentation, Research &amp; Archiving cell, an extension of the art gallery with an integrated team of dedicated art historians and curators. DAG works hard to enrich its archive by documenting every textual and visual material on art and artists and facilitates students, art critics and writers with an extensive information&ndash;database.</p>

<p>Setting up new professional standards in art dealing as well as in showcasing Indian art globally, Ashish Anand is driven with positive energy and committed to well-documented high quality production of art books/ catalogues and meaningful exhibitions that expand the horizon of mainstream Indian Art.</p>

<p>&nbsp;</p></td>

</tr>

<tr>

<td height="14" align="right" valign="top" bgcolor="#70A8D7">&nbsp;</td>

</tr>

</table>

</form>

</body>

</html>

**Admin Panel**

<%@ page contentType="text/html; charset=iso-8859-1" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Untitled Document</title>

<style type="text/css">

<!--

.style6 {color: #000000; font-weight: bold; }

.style7 {color: #000000}

.style9 {font-size: 12px}

.style11 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-size: 10;

}

.style12 {

font-size: 10px;

font-family: Verdana, Arial, Helvetica, sans-serif;

font-weight: bold;

}

.style13 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-size: 12;

}

.style15 {font-size: 10; }

.style16 {color: #000000; font-size: 12px; }

.style17 {color: #000000; font-weight: bold; font-size: 12px; }

.style18 {

font-size: large;

font-weight: bold;

color: #FF0000;

}

.style19 {color: #FF0000}

-->

</style>

<script src="SpryAssets/SpryMenuBar.js" type="text/javascript"></script>

<link href="SpryAssets/SpryMenuBarVertical.css" rel="stylesheet" type="text/css" />

<style type="text/css">

body {

background-image: url();

background-color: #000;

}

</style>

</head>

<body>

<form id="form1" name="form1" method="post" action="">

<table width="90%" rules="none" border="0" align="center" cellpadding="5" cellspacing="5">

<tr>

<td height="92" colspan="2" valign="top" bgcolor="#FFFFFF"><img src="img/hd.jpg" width="1100" height="173" /></td>

</tr>

<tr>

<td colspan="2" valign="top" bgcolor="#00E2DB">&nbsp;</td>

</tr>

<tr>

<td width="170" valign="top" bgcolor="#EEEEEE"><ul id="MenuBar1" class="MenuBarVertical">

<li><a href="ViewUserDetails.jsp">View User Details</a></li>

<li><a href="uploadeproduct.jsp">Upload Product</a></li>

<li><a href="customerbookingdetails.jsp"> Customer Booking Details</a></li>

<li><a href="index.jsp">Logout</a></li>

</ul></td>

<td width="693" height="382" valign="top"><p class="style13"></p>

<p align="center" class="style13">&nbsp;</p>

<p align="center" class="style13"><img src="img/add.jpg" width="576" height="300" /></p>

<p align="center" class="style13"><br />

</p></td>

</tr>

<tr>

<td colspan="2" bgcolor="#00E3DC">&nbsp;</td>

</tr>

</table>

</form>

<script type="text/javascript">

var MenuBar1 = new Spry.Widget.MenuBar("MenuBar1", {imgRight:"SpryAssets/SpryMenuBarRightHover.gif"});

</script>

</body>

</html>

Booking Product

<%@ page contentType="text/html; charset=utf-8" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>Untitled Document</title>

<script src="SpryAssets/SpryValidationTextField.js" type="text/javascript"></script>

<link href="SpryAssets/SpryValidationTextField.css" rel="stylesheet" type="text/css" />

</head>

<body>

<form id="form1" name="form1" method="post" action="bookingproduct\_code.jsp">

<%

String a=request.getParameter("value");

%>

<table width="993" border="0" align="center" cellpadding="4" cellspacing="4">

<tr>

<td align="right"><p><img src="img/hd.jpg" width="1084" height="265" /></p></td>

</tr>

<tr>

<td bgcolor="#304556">&nbsp;</td>

</tr>

<tr>

<td width="991" valign="top"><p><a href="welcome.jsp"></a></p>

<table rules="rows" width="853" border="1" align="center" cellpadding="5" cellspacing="5">

<tr>

<td colspan="2" align="center"><h1><img width="488" height="61" src="bookingproduct\_clip\_image001\_0000.gif" alt="Flowchart: Terminator: CUSTOMER BILLING DETAILS" /></h1></td>

</tr>

<tr>

<td width="255">Customer Name</td>

<td width="557"><span id="sprytextfield1">

<input name="text1" type="text" id="text1" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>Mobile No</td>

<td><span id="sprytextfield2">

<input name="text2" type="text" id="text2" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>Email Id</td>

<td><span id="sprytextfield3">

<input name="text3" type="text" id="text3" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>Full Address</td>

<td><span id="sprytextfield4">

<textarea name="text4" cols="88" rows="11" id="text4"></textarea>

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>Pin Code</td>

<td><span id="sprytextfield7">

<input name="text5" type="text" id="text5" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td height="32">Product Code Number</td>

<td><span id="sprytextfield5">

<input name="text6" type="text"id="text6" value="<%=a%>" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>Qty</td>

<td><span id="sprytextfield6">

<input name="text7" type="text" id="text7" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td colspan="2" align="center"><h1><img src="img/gett.png" width="532" height="169" /></h1></td>

</tr>

<tr>

<td>Card Type</td>

<td><select name="select" id="select">

<option>Master Card</option>

<option selected="selected">Debit Card</option>

<option>Credit Card</option>

<option>Fleet card</option>

<option>ATM card</option>

<option>Debit card</option>

<option>Charge card</option>

</select></td>

</tr>

<tr>

<td>Card Number</td>

<td><span id="sprytextfield8">

<input name="text8" type="text" id="text8" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>Security Code(4 Digit)</td>

<td><span id="sprytextfield9">

<input name="text9" type="text" id="text9" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>Expiry Date</td>

<td><span id="sprytextfield10">

<input name="text10" type="text" id="text10" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>Card Holder Name</td>

<td><span id="sprytextfield11">

<input name="text11" type="text" id="text11" size="44" />

<span class="textfieldRequiredMsg">A value is required.</span></span></td>

</tr>

<tr>

<td>&nbsp;</td>

<td><input type="submit" name="button" id="button" value="Submit" />

<a href="welcome.jsp">Back</a></td>

</tr>

<tr>

<td colspan="2" align="center"><h1><%=session.getAttribute("stock")%>&nbsp;</h1></td>

</tr>

</table>

<p>&nbsp;</p></td>

</tr>

<tr>

<td bgcolor="#B7CBE2">&nbsp;</td>

</tr>

</table>

</form>

<script type="text/javascript">

var sprytextfield1 = new Spry.Widget.ValidationTextField("sprytextfield1");

var sprytextfield2 = new Spry.Widget.ValidationTextField("sprytextfield2");

var sprytextfield3 = new Spry.Widget.ValidationTextField("sprytextfield3");

var sprytextfield4 = new Spry.Widget.ValidationTextField("sprytextfield4");

var sprytextfield5 = new Spry.Widget.ValidationTextField("sprytextfield5");

var sprytextfield6 = new Spry.Widget.ValidationTextField("sprytextfield6");

var sprytextfield7 = new Spry.Widget.ValidationTextField("sprytextfield7");

var sprytextfield8 = new Spry.Widget.ValidationTextField("sprytextfield8");

var sprytextfield9 = new Spry.Widget.ValidationTextField("sprytextfield9", "none");

var sprytextfield10 = new Spry.Widget.ValidationTextField("sprytextfield10");

var sprytextfield11 = new Spry.Widget.ValidationTextField("sprytextfield11");

</script>

</body>

</html>

Contact us

<%@ page contentType="text/html; charset=iso-8859-1" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Untitled Document</title>

<style type="text/css">

<!--

.style6 {color: #000000; font-weight: bold; }

.style7 {color: #000000}

.style9 {font-size: 12px}

body {

text-align: center;

}

.style10 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-weight: bold;

}

.style11 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-size: 10;

}

.style12 {

font-size: 10px;

font-family: Verdana, Arial, Helvetica, sans-serif;

font-weight: bold;

}

.style15 {

font-family: Verdana, Arial, Helvetica, sans-serif;

text-align: center;

}

-->

</style>

</head>

<%@include file="header.jsp" %>

<body>

<table width="90%" border="1" align="center" cellpadding="0" cellspacing="0">

<tr>

<td height="336" align="center" valign="top" bgcolor="#FDFDFD"><p class="style10">&nbsp;</p>

<p class="style10">&nbsp;</p>

<h1>Contact Us</h1>

<p><a href="https://www.google.com/maps?q=Nowe%20Garbary%20Office%20Center%20ul.%20Ma%C5%82e%20Garbary%209%2061-756%20Pozna%C5%84,%20Poland" target="\_blank">Nowe Garbary Office Center<br />

ul. Ma&#322;e Garbary 9<br />

61-756 Pozna&#324;, Poland</a></p>

<p class="style10">VAT-ID: PL7781454968<br />

REGON: 300826280<br />

KRS: 0000745671</p></td>

</tr>

</table>

</body>

</html>

Customer booking details

<%@ page contentType="text/html; charset=iso-8859-1" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Untitled Document</title>

<style type="text/css">

<!--

body,td,th {

font-weight: normal;

}

.style1 {color: #FFFFFF}

.style2 {color: #00FFFF}

.style10 {font-size: 12px}

.style12 {color: #990000}

.style16 {color: #FF0000}

.style17 {font-size: 14px}

.style18 {font-size: 14}

.style19 {color: #FFFFFF; font-size: 12px; }

#form1 table tr td a {

font-size: 12px;

}

-->

</style>

<script src="SpryAssets/SpryMenuBar.js" type="text/javascript"></script>

<link href="SpryAssets/SpryMenuBarVertical.css" rel="stylesheet" type="text/css" />

<style type="text/css">

.style15 {

font-size: 16px;

}

.style191 {

font-size: 11px;

font-weight: normal;

}

.style20 {

font-size: 14px;

font-weight: bold;

}

.style201 {

font-size: 14px;

font-weight: bold;

}

.style202 {

font-size: 14px;

font-weight: bold;

}

</style>

</head>

<%@ include file="conn.jsp"%>

<body>

</form>

<form id="form1" name="form1" method="post" action="">

<table rules="none" width="90%" align="center" cellpadding="5" cellspacing="5">

<tr>

<td height="133"><h1><img src="img/www2.jpg" width="628" height="182" /></h1></td>

</tr>

<tr>

<td valign="top" bgcolor="#2E6297">&nbsp;</td>

</tr>

<tr>

<td width="996" valign="top"><p>&nbsp;</p>

<p>&nbsp;</p>

<table width="1300" border="1" align="center" cellpadding="2" cellspacing="2" rules="none">

<tr>

<td width="105" align="center"><span class="style20"> Name</span></td>

<td width="98" align="center"><span class="style201">Phone</span></td>

<td width="92" align="center"><span class="style20">Email</span></td>

<td width="109" align="center">Address</td>

<td width="126" align="center">Pin Number</td>

<td width="109" align="center">Qty</td>

<td width="109" align="center">Card Type</td>

<td width="109">C.Holder Name</td>

<td width="103">Product Code</td>

<td width="106">Product Catogry</td>

<td width="106">Product Name</td>

<td width="106">Total Cost</td>

</tr>

<%

String query="select \* from bookingproduct";

ResultSet rs=st.executeQuery(query);

while(rs.next())

{

%>

<tr>

<td align="center"><span style="font-size:12px"><%=rs.getString(1)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(2)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(3)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(4)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(5)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(6)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(7)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(11)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(12)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(13)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(14)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(15)%></span></td>

<%

}

%>

</tr>

</table>

<p>&nbsp;</p>

<p>&nbsp;</p></td>

</tr>

<tr>

<td bgcolor="#2E6297">&nbsp;</td>

</tr>

</table>

</form>

<script type="text/javascript">

var MenuBar1 = new Spry.Widget.MenuBar("MenuBar1", {imgRight:"SpryAssets/SpryMenuBarRightHover.gif"});

</script>

</body>

</html>

Header

<%@ page contentType="text/html; charset=iso-8859-1" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Untitled Document</title>

<style type="text/css">

<!--

body,td,th {

}

.style1 {

color: #460A0A

}

#Layer1 {

position:absolute;

left:428px;

top:211px;

width:492px;

height:228px;

z-index:1;

}

.style13 {font-size: 24px}

body {

background-image: url();

}

#form4 td table tr td table tr td h1 {

text-align: center;

color: #55080E;

}

#form4 td table tr td .style1 {

color: #400809;

}

#form4 table tr td table tr td h1 {

text-align: center;

}

#form4 table tr td table tr td {

color: #400809;

}

#form4 table tr td table tr td a {

font-size: 12px;

}

td {

}

-->

</style>

<link href="SpryAssets/SpryMenuBarVertical.css" rel="stylesheet" type="text/css" />

</head>

<body>

<table rules="none" width="90%" border="1" align="center" cellpadding="0" cellspacing="0">

<tr>

<td height="87" valign="bottom" bgcolor="#000000"><p><img src="img/he.png" width="1200" height="260" /></p>

<p>&nbsp;</p></td>

</tr>

<tr>

<td align="right"><table rules="none" width="100%" border="1" align="center" cellpadding="3" cellspacing="3">

<tr>

<td width="9%" height="58" align="center"><a href="index.jsp"><img src="icon/hm.png" width="45" height="45" /></a></td>

<td width="8%" align="center">HOME</td>

<td width="8%" align="center"><a href="AboutUs.jsp"><img src="icon/aa.png" width="51" height="46" /></a></td>

<td width="16%" align="center"><p>ABOUT US</p></td>

<td width="7%" align="center"><a href="Login.jsp"><img src="icon/logiin.png" width="51" height="46" /></a></td>

<td width="12%" align="center"> LOGIN</td>

<td width="8%" align="center"><a href="NewUserRegister.jsp"><img src="icon/images.jpg" width="51" height="46" /></a></td>

<td width="15%" align="center">REGISTER</td>

<td width="17%" align="center"><a href="contactus.jsp"><img src="icon/ww.png" width="77" height="44" /></a></td>

</tr>

</table></td>

</tr>

</table>

</body>

</html>

New user Register

<%@ page contentType="text/html; charset=iso-8859-1" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Untitled Document</title>

</head>

<%@ include file="conn.jsp"%>

<body>

<%

String a=request.getParameter("t1");

String b=request.getParameter("t2");

String c=request.getParameter("t3");

String d=request.getParameter("t4");

String e=request.getParameter("t5");

st.executeUpdate("insert into reguser values('"+a+"','"+b+"','"+c+"','"+d+"','"+e+"')");

response.sendRedirect("index.jsp");

%>

</body>

</html>

Product Recommendation

/\*

SQLyog Community v11.31 (64 bit)

MySQL - 5.5.30 : Database - productrecomdation

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

/\*!40101 SET NAMES utf8 \*/;

/\*!40101 SET SQL\_MODE=''\*/;

/\*!40014 SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0 \*/;

/\*!40014 SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0 \*/;

/\*!40101 SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='NO\_AUTO\_VALUE\_ON\_ZERO' \*/;

/\*!40111 SET @OLD\_SQL\_NOTES=@@SQL\_NOTES, SQL\_NOTES=0 \*/;

CREATE DATABASE /\*!32312 IF NOT EXISTS\*/`productrecomdation` /\*!40100 DEFAULT CHARACTER SET latin1 \*/;

USE `productrecomdation`;

/\*Table structure for table `admin` \*/

DROP TABLE IF EXISTS `admin`;

CREATE TABLE `admin` (

`usern` varchar(55) DEFAULT NULL,

`password` varchar(55) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/\*Data for the table `admin` \*/

insert into `admin`(`usern`,`password`) values ('a','a'),('admin','admin');

/\*Table structure for table `bookingproduct` \*/

DROP TABLE IF EXISTS `bookingproduct`;

CREATE TABLE `bookingproduct` (

`custnm` varchar(44) DEFAULT NULL,

`mobno` varchar(44) DEFAULT NULL,

`emilid` varchar(44) DEFAULT NULL,

`addrs` varchar(44) DEFAULT NULL,

`pinn` varchar(6) DEFAULT NULL,

`qty` int(9) DEFAULT NULL,

`cardtype` varchar(44) DEFAULT NULL,

`cardnumber` varchar(99) DEFAULT NULL,

`SecurityCode` varchar(4) DEFAULT NULL,

`ExpiryDate` varchar(99) DEFAULT NULL,

`CardHolderName` varchar(99) DEFAULT NULL,

`pcode` int(9) DEFAULT NULL,

`pcatogry` varchar(55) DEFAULT NULL,

`pname` varchar(99) DEFAULT NULL,

`totalcost` int(9) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/\*Data for the table `bookingproduct` \*/

insert into `bookingproduct`(`custnm`,`mobno`,`emilid`,`addrs`,`pinn`,`qty`,`cardtype`,`cardnumber`,`SecurityCode`,`ExpiryDate`,`CardHolderName`,`pcode`,`pcatogry`,`pname`,`totalcost`) values ('a','a','ravijee1982@gmail.com','a','a',2,'Debit Card','222','2222','22','22',103,'Samsung','Samsung XY34',110000);

/\*Table structure for table `productupload` \*/

DROP TABLE IF EXISTS `productupload`;

CREATE TABLE `productupload` (

`pcode` varchar(88) DEFAULT NULL,

`classfication` varchar(99) DEFAULT NULL,

`photoupload` varchar(99) DEFAULT NULL,

`cost` int(9) DEFAULT NULL,

`pname` varchar(99) DEFAULT NULL,

`pcopany` varchar(99) DEFAULT NULL,

`productdescription` varchar(999) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/\*Data for the table `productupload` \*/

insert into `productupload`(`pcode`,`classfication`,`photoupload`,`cost`,`pname`,`pcopany`,`productdescription`) values ('101','Mobile','1.jpg',19000,'Sony Xperia E4','Sony','Sony Xperia E4 sdfs fsad fa df df sdf'),('102','Mobile','1.jpg',15000,'Nokia C2','Nokia',' sdf asdf'),('103','Mobile','1.jpg',21000,'Samsung Z3','Samsung','sdfa sd f'),('103','TV','4.jpg',55000,'Samsung XY34','Samsung','sdfa ds'),('104','TV','4.jpg',120000,'LG GXY3','LG','ds f adsfa '),('101','TV','4.jpg',38900,'Sony XY34','Sony','s adfa sdf sd'),('105','Computer','2.jpg',345345,'DEll I5','DEll','f a'),('106','Computer','2.jpg',35000,'HCL I2','HCL','s adf af'),('107','Computer','2.jpg',15000,'HP I5','HP','sdfasd '),('105','TV','4.jpg',444,'TVDY','Samsung','sdfa df a'),('105','TV','4.jpg',23423,'TVDY','Samsung','sdfa');

/\*Table structure for table `rating` \*/

DROP TABLE IF EXISTS `rating`;

CREATE TABLE `rating` (

`productnm` varchar(99) DEFAULT NULL,

`ratingg` int(9) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/\*Data for the table `rating` \*/

insert into `rating`(`productnm`,`ratingg`) values ('Nokia C2',24),('Sony Xperia E4',27),('Samsung Z3',2);

/\*Table structure for table `reguser` \*/

DROP TABLE IF EXISTS `reguser`;

CREATE TABLE `reguser` (

`nm` varchar(99) DEFAULT NULL,

`emaild` varchar(99) DEFAULT NULL,

`password` varbinary(99) DEFAULT NULL,

`mobiel` varchar(99) DEFAULT NULL,

`addres` varchar(99) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

/\*Data for the table `reguser` \*/

insert into `reguser`(`nm`,`emaild`,`password`,`mobiel`,`addres`) values ('ravi','ri66kumar@gmail.com','p','p','p'),('suraj','suraj@gmail.com','u','u','u'),('a','a','66','a','s');

/\*!40101 SET SQL\_MODE=@OLD\_SQL\_MODE \*/;

/\*!40014 SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS \*/;

/\*!40014 SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS \*/;

/\*!40111 SET SQL\_NOTES=@OLD\_SQL\_NOTES \*/;

Welcome code

<%@ page contentType="text/html; charset=iso-8859-1" language="java" import="java.sql.\*" errorPage="" %>

<%@page import="java.io.\*"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title></title>

<style type="text/css">

<!--

.style7 {color: #000000}

.style9 {

font-size: 12px;

color: #FFF;

}

.style10 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-weight: bold;

}

.style11 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-size: 10;

text-align: right;

}

.style12 {

font-size: 10px;

font-family: Verdana, Arial, Helvetica, sans-serif;

font-weight: bold;

}

.style13 {

font-family: Verdana, Arial, Helvetica, sans-serif;

font-size: 12px;

}

.style14 {

font-family: "Courier New", Courier, monospace;

color: #000000;

}

em {

color: #B83044;

}

-->

</style>

<script src="SpryAssets/SpryValidationTextField.js" type="text/javascript"></script>

<link href="SpryAssets/SpryValidationTextField.css" rel="stylesheet" type="text/css" />

<style type="text/css">

.style141 { font-family: "Courier New", Courier, monospace;

font-size: 18px;

color: #FFF;

}

#form1 table tr td table tr td h1 {

color: #FFF;

}

#form1 table tr td a {

}

body {

background-image: url();

}

</style>

</head>

<%@ include file="conn.jsp"%>

<body>

<form id="form1" name="form1" method="post" action="IconSearch.jsp">

<%

session.setAttribute("stock","");

%>

<table rules="none" width="974" border="0" align="center" cellpadding="4" cellspacing="4">

<tr>

<td height="94" colspan="2" align="right" valign="top"><h1><img src="img/hd.jpg" width="1084" height="224" /><br />

</h1></td>

</tr>

<tr>

<td height="21" colspan="2" valign="top"><a href="index.jsp">Logout</a></td>

</tr>

<tr>

<td height="21" colspan="2" align="right" valign="top" bgcolor="#70A8D7"><span id="sprytextfield1">

<input name="value" type="text" id="value" value="Classification" size="55" maxlength="22" />

<span class="textfieldRequiredMsg">A value is required.</span></span><span id="sprytextfield2">

<input name="text2" type="text" id="text2" value="Product Name" size="55" />

<span class="textfieldRequiredMsg">A value is required.</span></span>

<input type="submit" name="button" id="button" value="Search" /></td>

</tr>

<tr>

<td width="413" height="260" valign="top">

<p><a href="index.jsp"></a></p>

<table width="466" align="center" rules="rows">

<tr>

<td width="142" align="center"><a href="IconSearch.jsp?value=Mobile"><img src="img/mm.jpeg" width="63" height="125" /></a></td>

<td width="138"><a href="IconSearch.jsp?value=Computer"></a></td>

<td width="170" align="center"><a href="IconSearch.jsp?value=Computer"><img src="img/4.jpg" width="200" height="200" /></a></td>

</tr>

<tr>

<td colspan="3" align="center"><a href="IconSearch.jsp?value=TV"><img src="img/5.jpg" width="256" height="197" /></a></td>

</tr>

</table><p>&nbsp;</p></td>

<td width="413" align="center" valign="top">

<p>&nbsp;</p>

<p><a href="productrating.jsp"><img src="img/qq.png" width="298" height="222" /></a></p></td>

</tr>

<tr>

<td height="14" colspan="2" align="right" valign="top">

<%

String un=session.getAttribute("un").toString();

int popular=0;

try

{

ResultSet rs=st.executeQuery("select \* from bookingproduct where custnm='"+un+"'");

//--------------------------------------------------------------

int count = 1, tempCount;

int temp = 0,i=0,dbcount=0,s=0;

//-----------------------------------------------------------------

int a [] = new int[100];

while(rs.next())

{

a[i]=Integer.parseInt(rs.getString(12));

s=s+a[i];

i++;

}

// out.print(s);

popular = a[0];

dbcount=i;

for ( i = 0; i < dbcount; i++)

{

temp = a[i];

tempCount = 0;

for (int j = 1; j < dbcount; j++)

{

if (temp == a[j])

tempCount++;

}

if (tempCount > count)

{

popular = temp;

count = tempCount;

}

}

//out.print(popular);

}

catch(Exception ee)

{

out.print(ee);

}

%>

</td>

</tr>

<tr>

<td height="14" colspan="2" align="right" valign="top"><table rules="none" width="100%" align="center" cellpadding="5" cellspacing="5">

<tr>

<td width="126" bgcolor="#804040"><div align="center" class="style141">Product Code</div></td>

<td width="176" bgcolor="#804040"><div align="center" class="style141">Classfication </div></td>

<td width="155" bgcolor="#804040"><div align="center" class="style141">PHOto</div></td>

<td width="99" bgcolor="#804040"><span class="style141">Cost</span></td>

<td width="135" bgcolor="#804040"><span class="style141">P.Name</span></td>

<td width="190" bgcolor="#804040"><span class="style141">Company Name</span></td>

<td width="91" bgcolor="#804040"><div align="center"><span class="style141">Status</span></div></td>

</tr>

<tr>

<%

String a=request.getParameter("value");

String b=request.getParameter("text2");

String query1="select \* from productupload where pcode='"+popular+"'";

ResultSet rs=st.executeQuery(query1);

while(rs.next())

{

%>

<td><div align="center"><%= rs.getString(1) %></div></td>

<td><div align="center"><%= rs.getString(2) %></div></td>

<td><div align="center"><img src="productimage/<%= rs.getString(3)%>" width="100" height="100" border="2" /></div></td>

<td align="center"><%= rs.getString(4) %></td>

<td align="center"><%= rs.getString(5) %></td>

<td align="center"><%= rs.getString(6) %></td>

<td width="91" align="center"><span style="font-size:12px"><a href="bookingproduct.jsp?value=<%=rs.getString(1)%>">Buy</a></span></td>

</tr>

<%

}

%>

</table></td>

</tr>

<tr>

<td height="14" colspan="2" align="right" valign="top"><table rules="none" width="100%" align="center" cellpadding="5" cellspacing="5">

<tr>

<%

ResultSet r=st.executeQuery("SELECT \* FROM rating ORDER BY ratingg DESC");

String top=null;

if(r.next())

{

top=r.getString(1);

}

ResultSet sr=st.executeQuery("select \* from productupload where pname='"+top+"'");

while(sr.next())

{

%>

<td width="89"><div align="center"><%=sr.getString(1) %></div></td>

<td width="213"><div align="center"><%= sr.getString(2) %></div></td>

<td width="153"><div align="center"><img src="productimage/<%= sr.getString(3)%>" width="100" height="100" border="2" /></div></td>

<td width="85" align="center"><%= sr.getString(4) %></td>

<td width="145" align="center"><%= sr.getString(5) %></td>

<td width="199" align="center"><%= sr.getString(6) %></td>

<td width="88" align="center"><span style="font-size:12px"><a href="bookingproduct.jsp?value=<%=sr.getString(1)%>">Buy</a></span></td>

</tr>

<%

}

%>

</table></td>

</tr>

<tr>

<td height="14" colspan="2" align="right" valign="top" bgcolor="#804040"><a href="index.jsp"></a></td>

</tr>

</table>

</form>

<script type="text/javascript">

var sprytextfield1 = new Spry.Widget.ValidationTextField("sprytextfield1", "none", {validateOn:["blur"]});

var sprytextfield2 = new Spry.Widget.ValidationTextField("sprytextfield2", "none", {validateOn:["blur"]});

</script>

</body>

</html>

User Details

<%@ page contentType="text/html; charset=iso-8859-1" language="java" import="java.sql.\*" errorPage="" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Untitled Document</title>

<style type="text/css">

<!--

body,td,th {

font-weight: normal;

}

.style1 {color: #FFFFFF}

.style2 {color: #00FFFF}

.style10 {font-size: 12px}

.style12 {color: #990000}

.style16 {color: #FF0000}

.style17 {font-size: 14px}

.style18 {font-size: 14}

.style19 {color: #FFFFFF; font-size: 12px; }

#form1 table tr td a {

font-size: 12px;

}

-->

</style>

<script src="SpryAssets/SpryMenuBar.js" type="text/javascript"></script>

<link href="SpryAssets/SpryMenuBarVertical.css" rel="stylesheet" type="text/css" />

<style type="text/css">

.style15 {

font-size: 16px;

}

.style191 {

font-size: 11px;

font-weight: normal;

}

.style20 {

font-size: 14px;

font-weight: bold;

}

.style201 {

font-size: 14px;

font-weight: bold;

}

.style202 {

font-size: 14px;

font-weight: bold;

}

body {

background-image: url(img/174331.jpg);

}

</style>

</head>

<%@ include file="conn.jsp"%>

<body>

</form>

<form id="form1" name="form1" method="post" action="">

<table rules="none" width="90%" border="0" align="center" cellpadding="5" cellspacing="5">

<tr>

<td height="133" align="right"><h1><img src="img/dag-logo.gif" width="398" height="34" align="middle" /><br />

<em>India's Largest Marketplace</em></h1></td>

</tr>

<tr>

<td valign="top" bgcolor="#2E6297">&nbsp;</td>

</tr>

<tr>

<td width="996" valign="top"><p>&nbsp;</p>

<p>&nbsp;</p>

<table rules="none" width="900" border="1" align="center">

<tr>

<td width="115" align="center"><span class="style20"> Name</span></td>

<td width="94" align="center"><span class="style20">Email</span></td>

<td width="112" align="center">Password</td>

<td width="110" align="center"><span class="style201">Phone</span></td>

<td width="109" align="center">Address</td>

</tr>

<%

String query="select \* from reguser";

ResultSet rs=st.executeQuery(query);

while(rs.next())

{

%><tr>

<td align="center"><span style="font-size:12px"><%=rs.getString(1)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(2)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(3)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(4)%></span></td>

<td align="center"><span style="font-size:12px"><%=rs.getString(5)%></span></td>

<%

}%>

</tr>

</table>

<p>&nbsp;</p>

<p>&nbsp;</p></td>

</tr>

<tr>

<td bgcolor="#2E6297">&nbsp;</td>

</tr>

</table>

</form><script type="text/javascript">

var MenuBar1 = new Spry.Widget.MenuBar("MenuBar1", {imgRight:"SpryAssets/SpryMenuBarRightHover.gif"});

</script>

</body>

</html>

**5.2 CODE EFFICIENCY**

Reviewing of Code efficiency for a module is carried out after the module is successfully compiled and all the syntax errors eliminated. Code efficiency review is extremely cost-effective strategies for reduction in coding errors in order to produce high quality code. Normally, two types of efficiency are carried out on the code of a module - code optimization and code inspection. The procedure and final objective of these two efficiency techniques are very different as discussed below.

**OPTIMIZATION OF CODE**

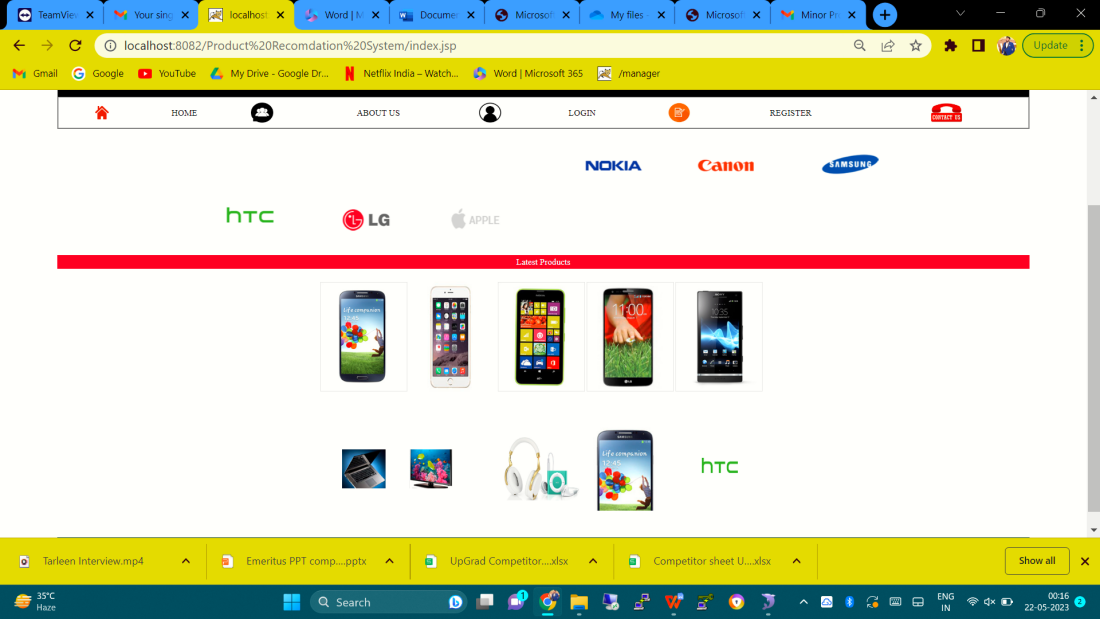
Code optimization is an informal code analysis technique. In this technique, after a module has been coded, it is successfully compiled and all syntax errors are eliminated. Some members of the development team are given the code a few days before the optimization meeting to read and understand the code. Each member selects some test cases and simulates execution of the code by hand (i.e. trace execution through each statement and function execution). The main objectives of the optimization are to discover the algorithmic and logical errors in the code. The members note down their findings to discuss these in a optimization meeting where the coder of the module is also present.

Even though a code optimization is an informal analysis technique, several guidelines have evolved over the years for making this naïve technique more effective and useful. Of course, these guidelines are based on personal experience, common sense, and several subjective factors. Therefore are based on personal experience, common sense, and several subjective factors. Therefore, guidelines should be considered as examples rather than as rules to be applied dogmatically. Some of these guidelines are the following:

The team performing the code optimization should not be either too big or too small. Ideally, it should consist of three to seven members.

**OUTPUTS**

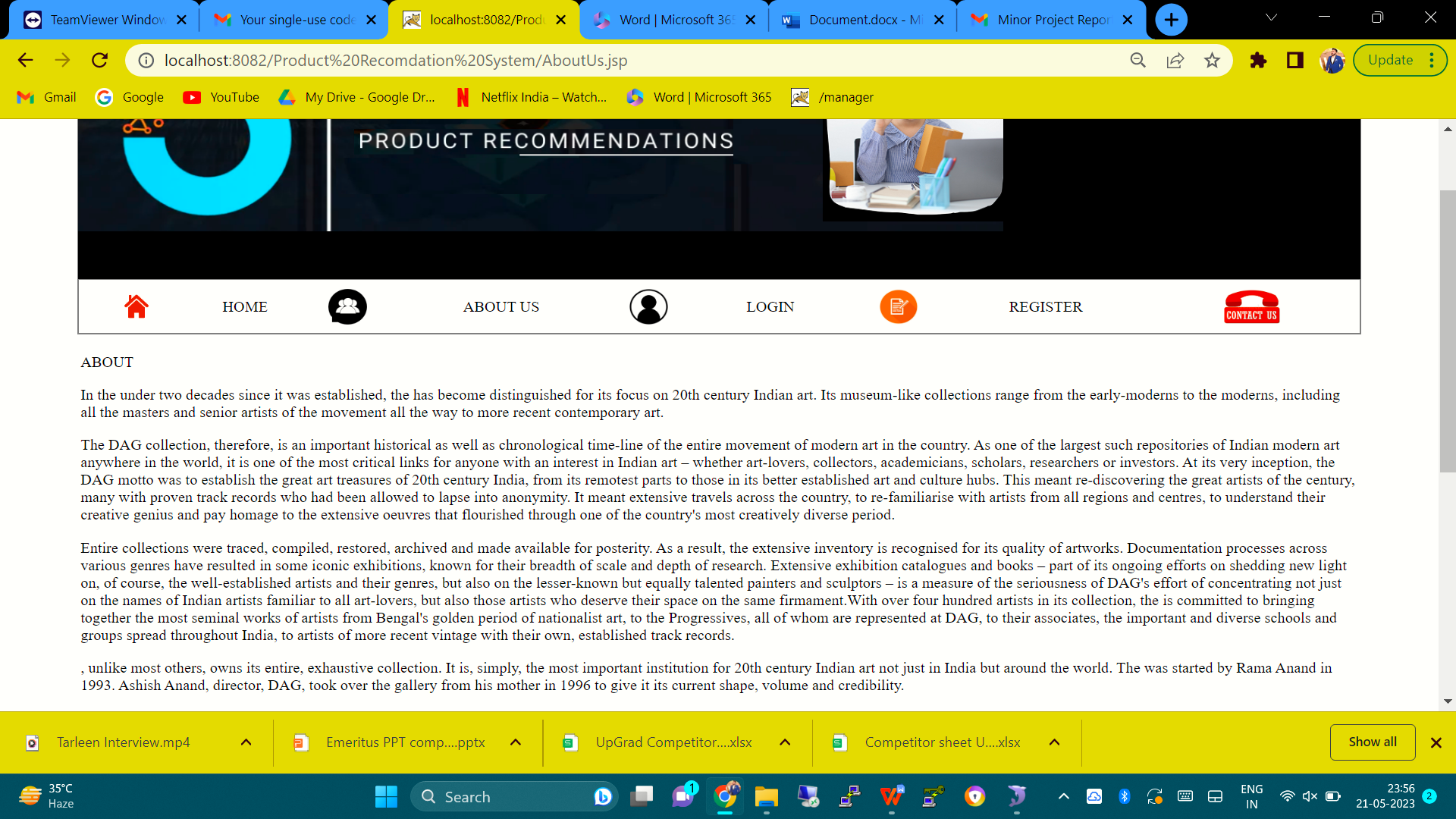
**Home Screen**



**Login Page**



**ABOUT US**

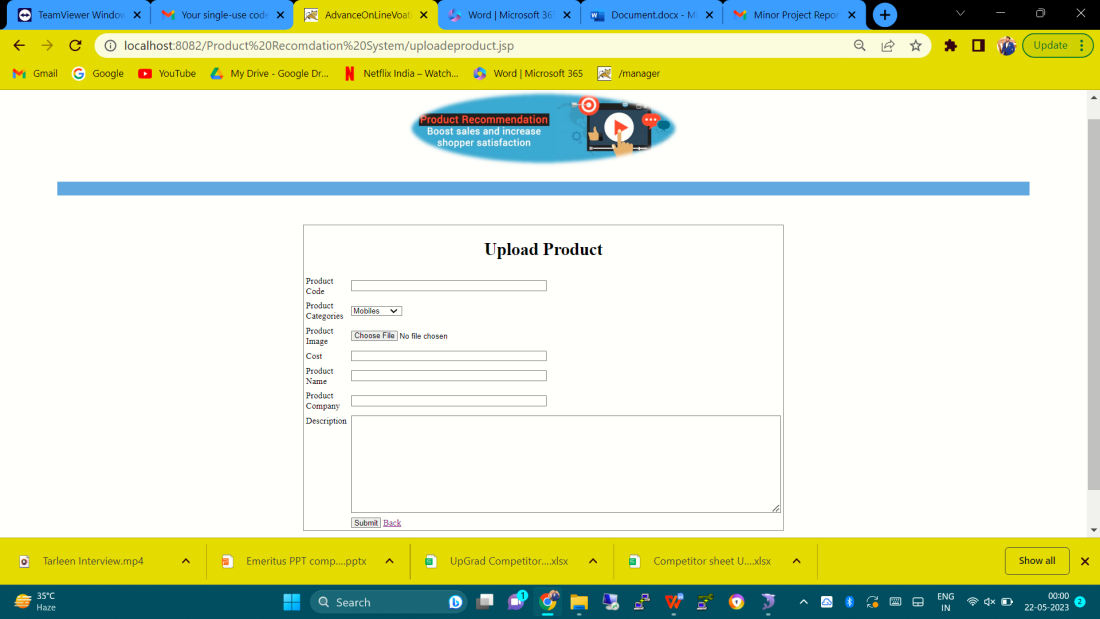


**ADMIN**

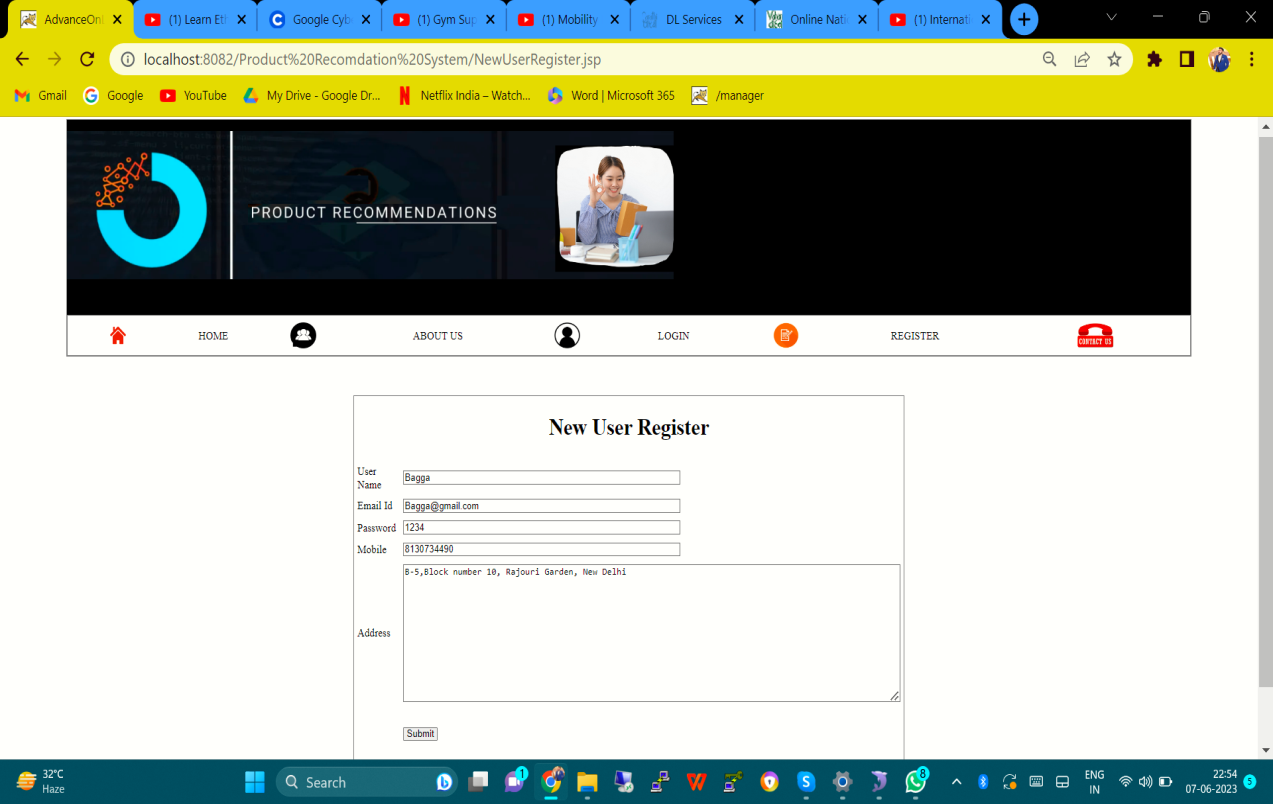
USER DETAILS (Users Logged in till now)



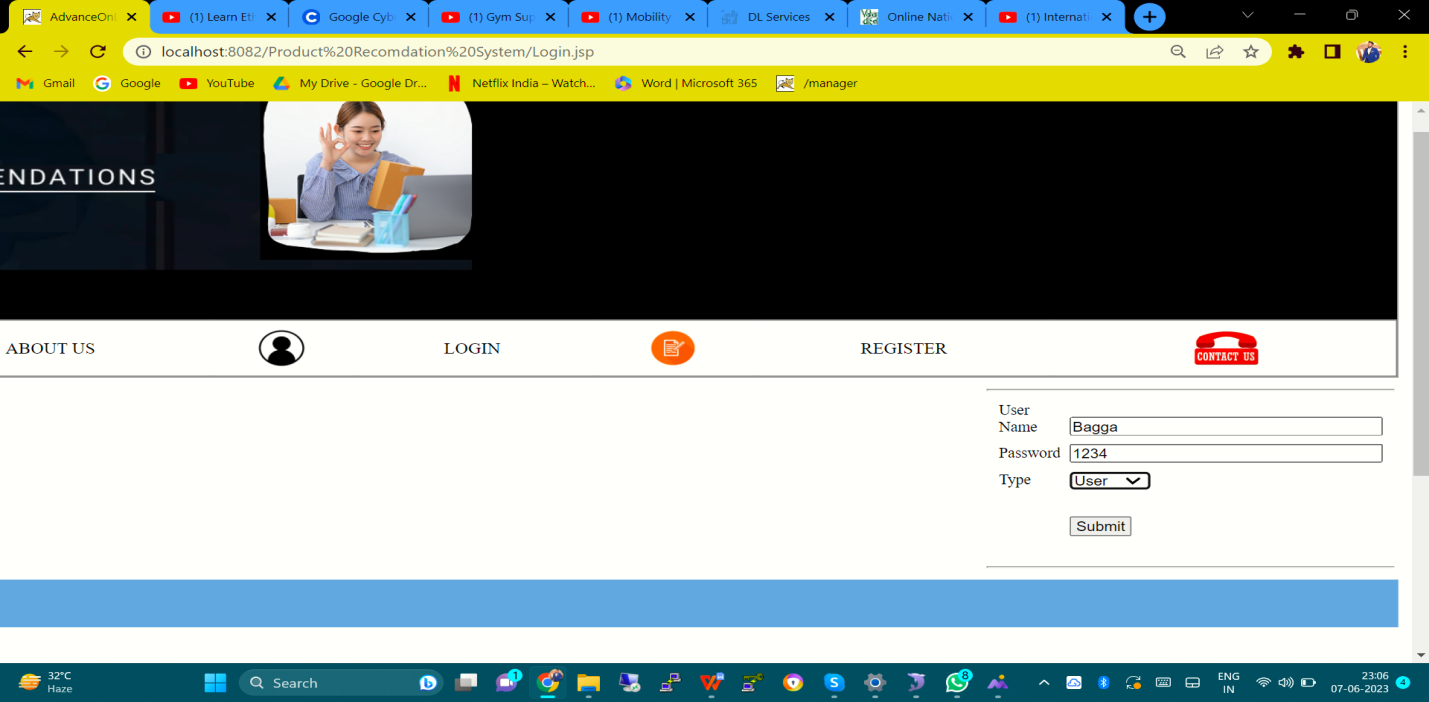
**PRODUCT UPLOAD DETAILS**



**New User Registration**



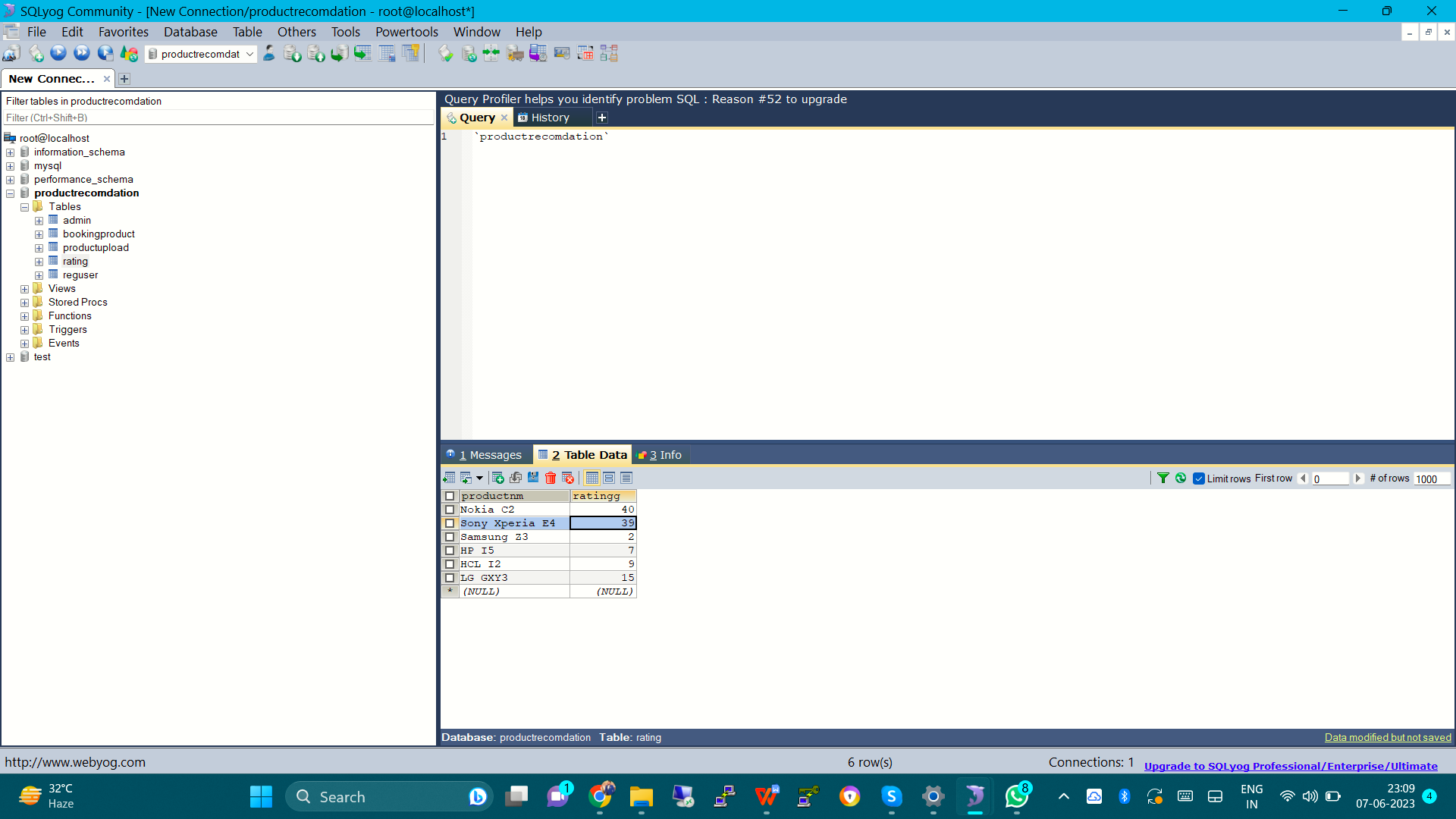
**New User Login**



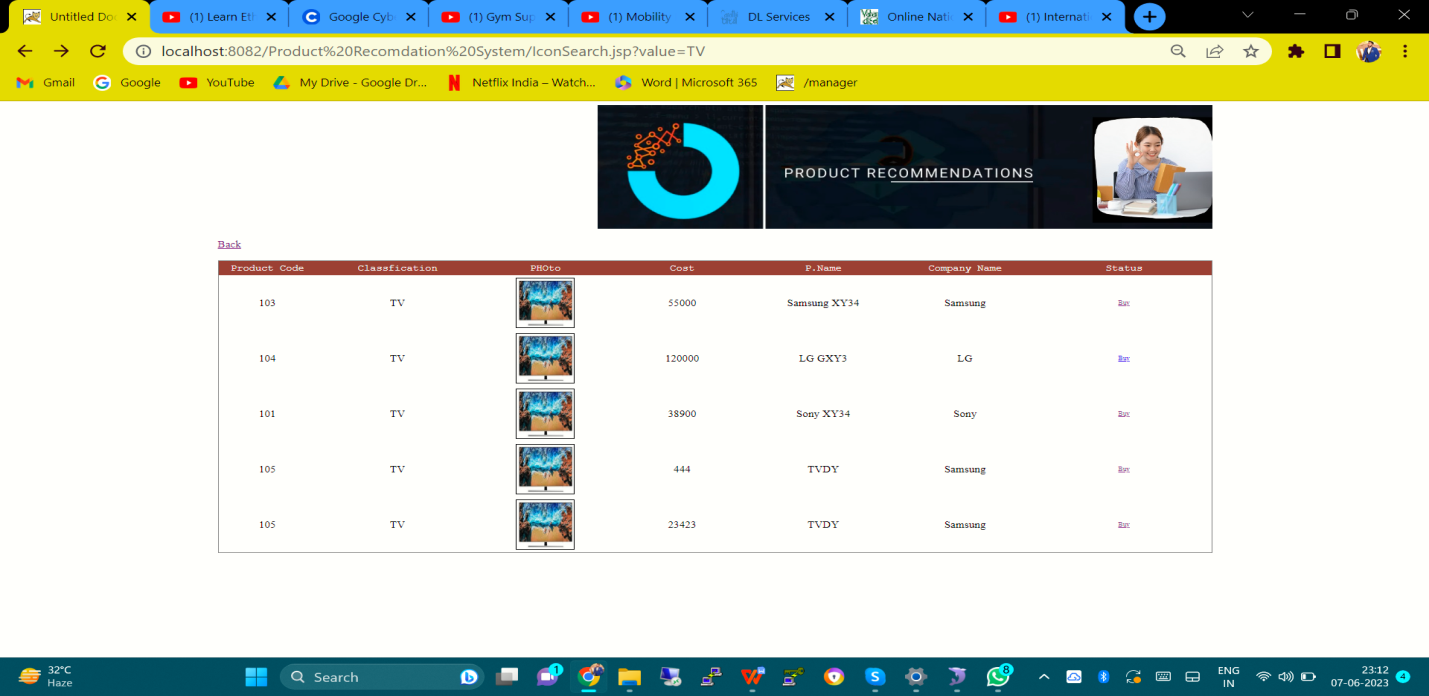
**New User home page with first recommendation according to Products ratings by previous users**



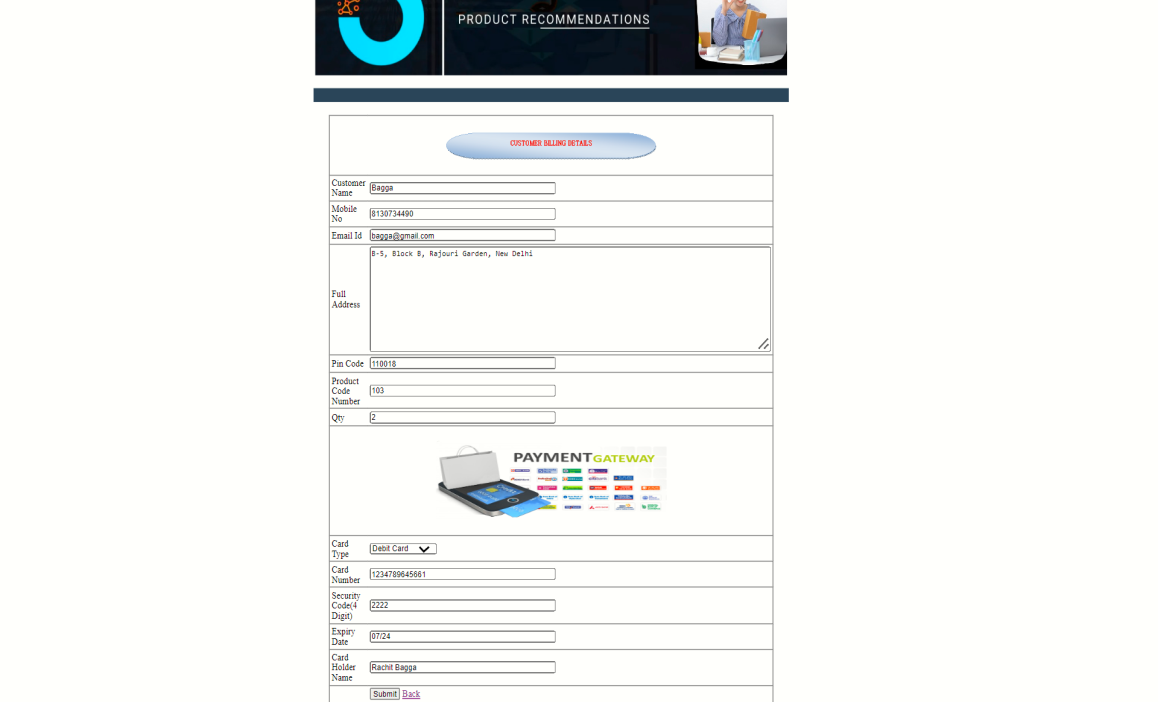
**Product Rating List( Nokia C2 has the highest rating )**



**First purchase of new user**



**Billing of Samsung XY34 ( PC: 103)**



**Recommendation of products after first purchase**



**Product Rating**



**Result**

**Registering as a new user will recommend the product which is rated the most by previous users but as soon as a product of a company is purchased by the new user, home screen will recommend the list of products related to the same company of the product.**

**Just like that, Bagga (new user) purchased Samsung Television and after the purchase all the samsung products are being listed as a recommendation.**

**TESTING PHASE**

One of the purposes of the testing is to validate and verify the system. Verification means checking the system to ensure that it is doing what the function is supposed to do and Validation means checking to ensure that system is doing what the user wants it to do.

No program or system design is perfect; communication between the user and the designer is not always complete or clear, and time is usually short. The result is errors and more errors. Theoretically, a newly designed system should have all the pieces in working order, but in reality, each piece works independently. Now is the time to put all the pieces into one system and test it to determine whether it meets the user's requirements. This is the best chance to detect and correct errors before the system is implemented. The purpose of system testing is to consider all the likely variations to which it will be subjected and then push the system to its limits. If we implement the system without proper testing then it might cause the problems.

1. Communication between the user and the designer.

2. The programmer's ability to generate a code that reflects

exactly the system specification.

3. The time frame for the design.

Theoretically, a new designed system should have all the pieces in working order, but in reality, each piece works independently. Now is the time to put all the pieces into one system and test it to determine whether it meets the requirements of the user. The process of system testing and the steps taken to validate and prepare a system for final implementation are:

**LEVELS OF TESTING**

**The different types of testing are as follows:**

**UNIT TESTING**

**VALIDATION**

**INTEGRATION**

**SYSTEM**

1. **UNIT TESTING:** This is the smallest testable unit of a computer system and is normally tested using the white box testing. The author of the programs usually carries out unit tests.
2. **INTEGRATION TESTING:**In integration testing, the different units of the system are integrated together to form the complete system and this type of testing checks the system as whole to ensure that it is doing what is supposed to do. The testing of an integrated system can be carried out top-down, bottom-up, or big-bang. In this type of testing, some parts will be tested with white box testing and some with black box testing techniques. This type of testing plays very important role in increasing the systems productivity. We have checked our system by using the integration testing techniques.
3. **SYSTEM TESTING:** A part from testing the system to validate the functionality of software against the requirements, it is also necessary to test the non-functional aspect of the system. Some examples of non-functional tools include tests to check performance, data security, usability/user friendliness, volume, load/stress that we have used in our project to test the various modules.

**System testing consists of the following steps:**

1. Program(s) testing.

2. String testing.

3. System testing.

4. System documentation.

5. User acceptance testing.

**4. FIELD TESTING:**

This is a special type of testing that may be very important in some projects. Here the system is tested in the actual operational surroundings. The interfaces with other systems and the real world are checked. This type of testing is very rarely used. So far our project is concerned; we haven't tested our project using the field testing.

**ACCEPTANCE TESTING:** After the developer has completed all rounds of testing and he is satisfied with the system, then the user takes over and re-tests the system from his point of view to judge whether it is acceptable according to some previously identified criteria. This is almost always a tricky situation in the project because of the inherent conflict between the developer and the user. In this project, it is the job of the bookstores to check the system that whether the made system fulfills the goals or not.

**WHY SYSTEM TESTING?**

Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, the goal will be successfully achieved. Inadequate testing results in two types of problems:

1. The time lag between the cause and the appearance of the problem.

2. The effect of system errors on the files and records within the system.

**ACTIVITY NETWORK FOR SYSTEM TESTING**

**The test plan entails the following activities:**

1. Prepare test plan.

2. Specify conditions for user acceptance testing.

3. Prepare test data for program testing.

4. Prepare test data for transaction path testing.

5. Plan user training.

6. Compile/assemble programs.

7. Prepare job performance aids.

8. Prepare operational documents.

**PREPARE TEST:** A workable test plan must be prepared in accordance with established design specifications. It includes the following items:

* Outputs expected from the system.
* Criteria for evaluating outputs.
* A volume of test data.
* Procedure for using test data.
* Personnel and training requirements.

**SPECIFY CONDITIONS FOR USER ACCEPTANCE TESTING**

Planning for user acceptance testing calls for the analyst and the user to agree on conditions for the test. **PREPARE TEST DATA FOR PROGRAM TESTING**

As each program is coded, test data are prepared and documented to ensure that all aspects of the program are properly tested.

**PREPARE TEST DATA FOR TRANSACTION PATH TESTING**

This activity develops the data required for testing every condition and transactions to be introduced into the system. The path of each transaction from origin to destination is carefully tested reliable results.

**PLAN USER TRAINING**

User training is designed to prepare the user for testing and converting the system. User involvement and training take place parallel with programming for three reasons:

• The system group has time available to spend on training while the programs are being written.

• Initiating a user-training program gives the systems group a clearer image of the user's interest in the new system.

• A trained user participates more effectively in system testing.

The training plan is followed by preparation of the user training manual and other text materials.

**COMPILE / ASSEMBLE PROGRAMS**

All programs have to be compiled / assembled for testing.

**PREPARE JOB PERFORMANCE AIDS**

In this activity the materials to be used by personnel to run the system are specified and scheduled. This includes a display of materials.

**PREPARE OPERATIONAL DOCUMENTS**

During the test plan stage, all operational documents are finalized including copies of the operational formats required by the candidate system.

**SYSTEMS TESTING**

The computer department to ensure that the system functions as specified does this testing. This testing is important to ensure that a working system is handed over to the user for acceptance testing.

**ACCEPTANCE TESTING**

The user to ensure that the system functions, as the user actually wanted performs this testing. With prototyping techniques, this stage becomes very much a formality to check the accuracy and completeness of processing. The screen layouts and output should already have been tested during the prototyping phase.

An error in the program code can remain undetected indefinitely. To prevent this from happening the code was tested at various levels. To successfully test a system, each condition, and combinations of conditions had to be tested. Each program was tested and linked to other programs. This unit of program is tested and linked to other units and so on until the complete system has been tested.

The purpose of testing is to ensure that each program is fully tested. To do so a test plan had to be created. The test plan consists of a number of test runs such as the valid paths through the code, and the exception and error handling paths. For each test run there is a list of conditions tested, the test data used and the result expected. The test plan was then reviewed to check that each path through the code is tested correctly. It is the responsibility of the programmer to collect the data that will produce the required test condition.

**VERIFICATION AND VALIDATION (V&V)**

The objectives of verification, validity activities are to assess and improve the quality of the work products generated during development and modification of the software. Quality depends upon the various attributes like correctness, completeness, consistency, reliability, usefulness, usability, efficiency and conformance to standards.

The terms verification and validation are used synonymously. These are defined as under: -

**Verification:** “Are we building the product right?”

**Validation:** “Are we building the right product?”

Verification activities include proving, testing, and reviews. Validation is the process of evaluating software at the end of the software development to ensure compliance with the software requirements. Testing is a common method of validation. Clearly, for high reliability we need to perform both activities. Together, they are often called V&V activities.

The major V&V activities for software development are inspection, reviews, and testing (both static and dynamic). The V&V plan identifies the different V&V tasks for the different phases and specifies how these tasks contribute to the project V&V goals. The methods to be used for performing these V&V activities, the responsibilities and milestones for each of these activities, inputs and outputs for each V&V task, and criteria for evaluating the outputs are also specified.

The two major V&V approaches are testing and inspections. Testing is an activity that can be generally performed only on code. It is an important activity and is discussed in detail in a later chapter. Inspection is a more general activity that can be applied to any work product, including code. Many of the V&V tasks are such that for them, an inspection type of activity is the only possible way to perform the tasks (e.g. trace ability and document evaluation). Due to this, inspections play a significant role in verification.

**FUTURE SCOPE**

**One of the benefits of Deep Learning is similar to matrix factorization, in that there is an ability to derive latent attributes. Deep Learning, however, can make up for some of the weaknesses of matrix factorization such as the inability to include time in the model — which standard matrix factorization isn’t designed for. Deep Learning, however, can utilize Recurrent Neural Networks which are specifically designed for time and sequence data.Incorporating time into a recommender system is important, because there are often preference seasonal effects. For example, it is likely that in December, more people are going to be watching holiday-themed movies and buying home decorations.**

**Another point is the need to see what would happen if a customer was shown a sub-optimal recommendation. This is taking a reinforcement learning approach, since the goal in this case would be to show customers a recommendation, and then record what the customer does. At times, customers can be recommended something that does not seem like the best option, just to see how the customer reacts which will improve the learning in the long-term.Recommendation systems can be a very powerful tool in a company’s arsenal, and future developments are going to increase business value even further. Some of the applications include being able to anticipate seasonal purchases based on recommendations, determine important purchases, and give better recommendations to customers which can increase retention and brand loyalty.**

**Most businesses will have some use for recommendation systems, and I encourage everyone to learn more about this fascinating area.**

**BIBLIOGRAPHY**

1. Senn James A : Analysis and Design of Information Systems, McGraw Hill, International Ed, 1989.

2. O'Brien Jamea A : Management Information Systems, Galgotia Publications Pvt Ltd, New Delhi, 1998.

3. Laudon & Laudon : Management Information Systems, Organisation and Technology, PHI, New Delhi, 1998.

4. Wetherebe James : Systems Analysis and Design, Galgotia Publications Pvt Ltd, New Delhi, 1990

**GLOSSARY**

**TEXT BOX**

A text Box control, sometimes called an edit field or edit control, displays information entered at design time, entered by the user, or assigned to the control in code at run time.

**BUTTON**

Use a Command Button control to begin, interrupt, or end a process. When chosen, a Command Button appears pushed in and so is sometimes called a push button.

**LIST BOX**

A List Box control displays a list of items from which the user can select one or more. If the number of items exceeds the number that can be displayed, a scroll bar is automatically added to the List Box control.

**LABEL**

A Label control is a graphical control you can use to display text that a user can’t change directly.

**GROUPBOX**

A GROUPBOX control provides an identifiable grouping for controls. You can also use a Frame to subdivide a form functionally – for example, to separate groups of Option Button controls.

**COMBO BOX**

A Combo Box control combines the features of a text box and a list box. This control allows the user to select an item either by typing text into the combo box, or by selecting it from the list.

**TIMER**

A Timer control can execute code at regular intervals by causing a Timer event to occur. The Timer control, invisible to the user, is useful for background processing.

**PICTURE BOX**

The primary use for the Picture Box control is to display a picture to the user. The actual picture that is displayed is determined by the picture property. The picture property contains the file name (and optional path) for the picture file that you wish to display.

**DATA GRID CONTROL**

The DATA Grid control displays and operates on tabular data. It allows complete flexibility to sort, merge, and format tables containing strings and pictures. When bound to a Data control, MSFlexGrid displays read-only data.

**DATE AND TIME PICKER CONTROL**

A Date and Time Picker (DTP) Control provides a simple and intuitive interface through which to exchange data and time information with a user. For example, with a DTP control you can ask the user to enter a data and then retrieve his or her selection with ease.

**OPTION BUTTON**

An Option Button control displays an option that can be turned on or off.

**IMAGE CONTROL**

Use the Image control to display a graphic. An Image control can display a graphic from an icon, bitmap or metafile, as well as enhanced metafile, JPEG, or GIF files.

**CHECK BOX CONTROL**

A Check Box indicates whether a particular condition is on or off. We use check boxes in an application to give users true/false or yes/no options. Because check boxes work independently of each other, a user can select any number of check boxes at the same time.