

**A
MINI PROJECT
ON
WEBSITE PUBLISHING SYSTEM**

A project report submitted to The Jawaharlal Nehru Technological University
in partial fulfillment for the award of the

**Bachelor of Technology
In
COMPUTER SCIENCE AND ENGINEERING**

Submitted by
B. RAM PRASAD (11RJ1A0514)
D. SAINATH (11RJ1A0527)
A. ABHINEETH REDDY (11RJ1A0502)

UNDER THE GUIDANCE OF
Mr. N. SATEESH
(ASSISTANT PROFESSOR)
(CSE DEPARTMENT)



MALLA REDDY INSTITUTE OF TECHNOLOGY
(Affiliated to JNTU, Hyderabad)
Maisammaguda, Secunderabad- 500 100



MALLA REDDY INSTITUTE OF TECHNOLOGY

Maisammaguda, Dhulapally Post (Via Hakimpet), Secunderabad – 500014

(Approved by AICTE, Affiliated to JNTU, Hyderabad)

CERTIFICATE

This is to certify that this project work entitled "**Website Publishing System**" is a bonafide work carried out by **B. Ram Prasad** bearing Hall Ticket Number: **11RJ1A0514**, **D. Sainath** bearing Hall Ticket Number: **11RJ1A0527**, **A. Abhineeth Reddy** bearing Hall Ticket Number: **11RJ1A0502** of **COMPUTER SCIENCE AND ENGINEERING DEPARTMENT** at **MALLA REDDY INSTITUTE OF TECHNOLOGY** and submitted to **JNT UNIVERSITY, HYDERABAD** in the partial fulfillment of the requirements for the award of **BACHELOR OF TECHNOLOGY**.

Guide

Mr. N. Sateesh M.Tech
Assistant Professor,
CSE Department

Head of the Department

Mr. N. Sateesh M.Tech
Assistant Professor,
CSE Department

External Examiner

ACKNOWLEDGEMENT

We hereby take the opportunity to thank our beloved **Principal Dr. M. Murali Krishna** for his gratitude and kindness by giving us all the facilities required for the completion of the project.

We would like to extend our sincere thanks to **Mr. N. Sateesh, Head of the Department, Computer Science and Engineering**, for his encouragement and support at all the stages of the project.

We would like to take this opportunity to express our heartfelt gratitude and our sincere thanks to **Mr. T. Srikanth, Asst. Professor, CSE Dept.**, our project coordinator at **Malla Reddy Institute Of Technology** who helped us a lot for the successful completion of our project "**Website Publishing System**".

We would also like to extend our enormous gratitude and sincere thanks to our project internal guide **Mr. N. Sateesh, Asst. Professor, CSE Dept.**, for his efficient, able advice and helping hand for developing the project.

We also thank the entire teaching faculty who were instrumental in making this project a successful one.

BY
B. Ram Prasad (11RJ1A0514)
D. Sainath (11RJ1A0527)
A. Abhineeth Reddy (11RJ1A0502)

DECLARATION

We hereby declare that the project entitled "**Website Publishing System**" submitted to **Malla Reddy Institute Of Technology, affiliated to Jawaharlal Nehru Technological University Hyderabad (JNTUH)** for the award of the degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING** is a result of work done by us. It is further declared that the project report or any part of it has not been previously submitted to any university or institute for the award of degree or diploma.

B. Ram Prasad (11RJ1A0514)
D.Sainath (11RJ1A0527)
A. Abhineeth Reddy (11RJ1A0502)

ABSTRACT

The website publishing system aim is to update the Web site dynamically by giving access to its business groups. Each business group has its products, marketing, tenders, news, events, and advertisements of its own. To enable dynamic updating into the website it has been decided by the management to decentralize this activity.

Static websites can only really be updated by someone with knowledge of website development. Static html web pages requires you to edit your content directly in the code when making updates for your website like uploading a new content, updating pages, upload or update new images of your products.

Dynamic sites on the other hand can be more expensive to develop initially, but the advantages are numerous. At a basic level, a dynamic website can give the website owner the ability to simply update and add new content to the site. For example, news and events could be posted to the site through a simple browser interface.

The purpose of Website Publishing System (WPS), controlled and organized by Administrator of an Organization, is to update the website dynamically by giving access to its business groups. Each business group has its Products, Tenders, Events, News and Advertisement Details of its own. To enable the dynamic updating into the website it has been decided by the management to decentralize these activities.

This website is intended for the use of an Administrator, Employees working at organization and customers. The administrator deals with managing employee work force, creating and modifying the user privileges, managing products, events, organizational news, and tenders.

Data storage and editing, data exchange, publishing details in website are the key aspects of Administering. Data storage usually includes business records like company products, tenders, events, news records and other primary company advertisement details and getting the feedback from the customers. Data applications involve the capture and editing of a products details, image, or spreadsheet. The exchange of stored and manipulated information is an equally important component of a website publishing system.

INDEX

TABLE OF CONTENTS

1. Introduction	1
2. Problem definition	2
3. System analysis	3
3.1 Existing system	3
3.2 Proposed system	3
4. Feasibility study	4
4.1 Types of feasibility	4
5. Project overview	6
6. Definition, acronyms and abbreviations	9
7. System requirements	10
7.1 Software requirements	10
7.2 Hardware requirements	10
8. Technologies	11
9. System design	27
10. Database design	35
11.1 Tables	35
11. Test report, Test plan	40
12. System Implementation	45
13. Conclusion	89
14. Bibliography	90

LIST OF FIGURES

1. Figure 1 – Type 1 Driver Architecture	19
2. Figure 2 – Type 2 Driver Architecture	20
3. Figure 3 – Type 3 Driver Architecture	22
4. Figure 4 – Type 4 Driver Architecture	24
5. Figure 5 – Admin Use Case Diagram	28
6. Figure 6 – Employee Use Case Diagram	29
7. Figure 7 –Client Use Case Diagram	29
8. Figure 8 – Class Diagram	30
9. Figure 9 –Sequence Diagram	31
10. Figure 10 – Activity Diagram for Admin	32
11. Figure 11 – Activity Diagram for Employee	33
12. Figure 12 –Activity Diagram for Client	34

CHAPTER 1

INTRODUCTION

The purpose of Website Publishing System (WPS), controlled and organized by Administrator of an Organization, is to update the website dynamically by giving access to its business groups. Each business group has its Products, Tenders, Events, News and Advertisement Details of its own. To enable the dynamic updating into the website it has been decided by the management to decentralize these activities.

This website is intended for the use of an Administrator, Employees working at organization and customers. The administrator deals with managing employee work force, creating and modifying the user privileges, managing products, events, organizational news, and tenders.

Data storage and editing, data exchange, publishing details in website are the key aspects of Administering. Data storage usually includes business records like company products, tenders, events, news records and other primary company advertisement details and getting the feedback from the customers. Data applications involve the capture and editing of a products details, image, or spreadsheet. The exchange of stored and manipulated information is an equally important component of a website publishing system.

CHAPTER 2

PROBLEM DEFINITION

Previously all the data in the website is updated statically, which would consume a lot of time and also more number of people will be involved in doing that work, which instead of simplifying the task will make it more complex and complicated. Because in this all the details are maintained manually by the respective members, so in order to get any detail of the project the visitor has to go all the way to the respective member of that organization. Maintenance of large amount of data is a complex process.

In the static webpages the content will never change unless the change applied manually, the changes made in the database are not reflected directly in the website.

To overcome all the disadvantages mentioned above the data is automated. By doing so the time and people required to complete a task will be reduced and also the entire process will become simpler to understand. Here dynamic updating is possible and because all the data are represented in a simple and understandable manner, searching the required data will consume less time here. This application runs on client server architecture, and is used to publish the various content to make the system dynamic in nature.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Static websites can only really be updated by someone with knowledge of website development. Static html web pages requires you to edit your content directly in the code when making updates for your website like uploading a new content, updating pages, upload or update new images of your products.

In the case of static Web sites, all the pages are fully constructed by Web designers; any subsequent changes also need to be made by them. Such pages are called "static" because their content doesn't change dynamically - the only way to change them is to have a Web designer modify them explicitly.

DISADVANTAGES OF STATIC WEBPAGES:

- Requires web development expertise to update website.
- Difficult to maintain and update website.
- Website does not interact with customer or website visitors.

Delay in updating web pages since you have to contact a web developer to do it for you.

4.2 PROPOSED SYSTEM:

Dynamic sites on the other hand can be more expensive to develop initially, but the advantages are numerous. At a basic level, a dynamic website can give the website owner the ability to simply update and add new content to the site. For example, news and events could be posted to the site through a simple browser interface.

ADVANTAGES OF DYNAMIC WEBPAGES:

- Much easier to update website.
- Capability to interact with the user.
- You can add new functionalities to your website
- Website is always up-to-date.

CHAPTER 4

FEASIBILITY STUDY

Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of an existing business or proposed venture, opportunities and threats as presented by the environment, the resources required to carry through, and ultimately the prospects for success. In its simplest terms, the two criteria to judge feasibility are cost required and value to be attained. As such, a well-designed feasibility study should provide a historical background of the business or project, description of the product or service accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirements and tax obligations. Generally, feasibility studies precede technical development and project implementation.

4.1 Five common factors

The acronym TELOS refers to the five areas of feasibility - Technical, Economic, Legal, Operational, and Scheduling.

4.1.1 Technology Feasibility

Technological feasibility is carried out to determine whether the company has the capability, in terms of software, hardware, personnel and expertise, to handle the completion of the project.

4.1.2 Economic feasibility

Economic analysis is the most frequently used method for evaluating the effectiveness of a new system. More commonly known as cost/benefit analysis, the procedure is to maintain the record of cost of the software and hardware needed for the proposed system. Comparing the cost of the proposed system and the given budget and see if it profitable or not.

4.1.3 Legal feasibility

Determines whether the proposed system conflicts with legal requirements, e. g. a data processing system must comply with the local Data Protection Acts .

4.1.4 Operational feasibility

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

4.1.5 Schedule feasibility

A project will fail if it takes too long to be completed before it is useful. Typically this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is.

CHAPTER 5

PROJECT OVERVIEW

INTRODUCTION:

The website publishing system aim is to update the website dynamically by giving access to its business groups. Each business group has its products, employees, events, news and divisions of its own. To enable dynamic updating into the website it has been decided by the management to decentralize this activity.

The project consists of following modules:

1. Product Publishing Module
2. Event / Visitor Publishing Module
3. News Publishing Module
4. Advertisement Publishing Module
5. Tender Publishing Module

5.1. PROJECT MODULES

ADMINISTRATOR:

In website publishing system the main role of administrator is publishing the details in the website of an organization specifying access rights to the employees of the organization belonging to different departments

Access Rights:

The access rights for the administrator are insert, update, delete, view. The access rights assigned by the administrator for the employees are insert, update and view and for visitors only view based on their roles.

PRODUCT PUBLISHING SYSTEM:

This module contains all the details of the products launched in the organization. Product details include product id, name, image, cost, description, manufactured date, and the id and name of the division in which the product is manufactured. Product publishing module consists of sub module namely:

Division wise product details:

The employee who wants to publish the products in the website will insert the details in the database and finally those details are published in the website by the admin division wise.

Product photo publishing:

In addition to the details mentioned above division wise products photographs are being published in the website. After publishing the details in the website, viewer can see the complete product details.

Removal of old products description (Archive):

After certain period of time from the list of products, some products move to archive

Feedback or Requirement form from customer:

In an organization, there is always a need of getting a feedback opinion from the customers who viewed the products from various departments, thus the webpage is created to obtain their feedback.

EVENT PUBLISHING SYSTEM:

In this module all the details of the events conducted in the organization are placed. Event details include event name, event image, event description, event date and the division id and name in which the event occurred.

This module consists of sub modules namely:

Division wise Event/Visitor's photographs publishing:

Organization has many departments in which events going to occur. These event/visitors division wise photographs are being published in the website.

Feedback or requirement form from customer:

After publishing the event/visitors details in website, feedback from the customers are taken into consideration.

NEWS PUBLISHING SYSTEM:

In this module all the latest happenings of the organization, which includes all the recent activities, events, and date of launch of new products will be maintained and published. News publishing module consists of sub module namely:

Record the news, enquire, etc.:

The details which are maintained in this module are news id, news title, news description, and news date.

Places the news in organization website:

Finally administrator publishes the above news details in the organization website

ADVERTISEMENT PUBLISHING SYSTEM:

In this module, the advertisements are used for the promotion of the products. It also provides the job details of the organization. Advertisement publishing module consists of sub module namely:

Record the requirement:

The details maintained in this module are advertisements id, advertisements title, advertisements description, advertisements date, starting date of tender and last date of tender.

Places Ads in the website:

Finally administrator publishes advertisements in the organization website.

TENDER PUBLISHING SYSTEM

In this module, customers apply for tenders of products through online (E-Tendering). It consists of sub modules namely:

Division wise Tender Description

This module includes the tender id, company id, company name, quotation, product id, product name, product description.

Tender Quotation:

The customer when applying for tender will mention the tender quotation.

Compares Quotation:

Employee views all the tenders applied and compares the quotations of the tenders.

Generate report of each Tender

If the applied tender's description and quotation are satisfying the rules of the organization. Employee generates a report accepting the tender otherwise that tender is rejected.

CHAPTER 6

DEFINITIONS, ACRONYMS AND ABBREVIATIONS

SRS	:	Software Requirement Specification.
SDD	:	Software Design Description.
WPS	:	Website Publishing System
CED	:	Computer Education Department
UML	:	Unified Modeling Language

WEBSITE:

It is a collection of Web pages which are dynamically interlinked.

QUOTATION:

A statement of the current market price of a commodity.

CHAPTER 7

SYSTEM REQUIREMENTS

7.1 SOFTWARE REQUIREMENTS

Operating System	:	Windows XP Service Pack2
Technologies	:	Java/J2EE (Servlets, JSP, JDBC)
Web Server	:	Tomcat 6.0
Database	:	Oracle Database 10g Express Edition
Software's	:	JDK 1.6

7.2 HARDWARE REQUIREMENTS

Hardware	:	Pentium based systems with a minimum of p4
RAM	:	256MB (Minimum)

CHAPTER 8

TECHNOLOGIES

Languages Used:

In this project, we chose Java language for developing the code.

JAVA:

Java is a programming language originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995 as a core component Sun Microsystems' Java platform. The language derives much of its syntax from C & C++ but has a simpler object model and fewer low –level facilities. Java applications are typically compiled to byte code (class file) that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Java is general - purpose, concurrent, class-based and object-oriented and is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers “write once run anywhere”. Java is considered by many as one of the most influential programming languages of the 20th century, and is widely used from application software to web applications.

The original and reference implementation java compilers, virtual machines and class libraries were developed by Sun from 1995. As of May 2007 in compliance with the specifications of the Java Community Process, Sun relicensed most of its Java technologies under the GNU General Public License. Others have also developed alternative implementations of these Sun technologies, such as GNU compiler for Java and GNU class path.

Principles:

There were five primary goals in the creation of the Java language:

- It should be “simple, object-oriented and familiar”.
- It should be “robust and secure”.
- It should be “architecture neutral and portable”.
- It should execute with “high performance”.
- It should be “interpreted, threaded, and dynamic”.

FEATURES OF JAVA:-

PLATFORM INDEPENDENT :-

The concept of Write-once-run-anywhere (known as platform independent) is one of the important key feature of Java language that makes java as the most powerful language. Unlike many other programming languages including C and C++ when Java is compiled, it is not compiled into platform specific machine, rather into independent byte code. This byte code is distributed over the web and interpreted by Java virtual Machine (JVM) on whichever platform it is being run.

SIMPLE:

Java is designed to be easy to learn. If you understand the basic concept of OOPS, java would be easy to master.

OBJECT ORIENTED:-

In java everything is an object. To be an object-oriented language, any language must follow at least the four characteristics.

Inheritance:

Inheritance is the process by which one object acquires the properties of another object

Encapsulation:

It is the mechanism of combining information & providing the abstraction.

Polymorphism:

Polymorphism is the way of providing different functionality by the functions having the same name based on the signatures of the methods.

Distributed:

The widely used protocols like HTTP and FTP are developed in java. Internet programmers can call functions on these protocols and can access the files from any remote machine on the internet rather than writing codes on their local system.

Portable:

The feature Write-once –run-anywhere makes the java language portable provided that the system must have interpreter for the JVM. Java also has the standard data size irrespective of operating system or the processor. These features makes the java as a portable language.

Dynamic:

While executing the java program the user can get the required files dynamically from a local drive or from a computer thousands of miles away from the user just by connecting with the internet.

Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry extensive amount of run-time information that can be used to verify and resolve accesses to object on run-time

Secure:

With Java secure feature it enabled to develop virus-free, tamper-free systems.

Authentication techniques are based on public-key encryption.

Performance:

Java uses native code usage, and light weight process called threads. In the beginning interpretation of byte code resulted the performance slow but the advance version of JVM uses the adaptive and just in time compilation technique that improves the performance.

Multithreaded:

Java is also Multithreaded programming language. Multithreading means a single program has multiple threads executing independently at the same time. Multiple threads execute instructions according to the program code in a process or program.

Multithreading works the similar way as multi processes run on one computer.

Multithreading programming is very interesting concept in java. In multithreaded programs not even a single thread disturbs the execution of other thread. Threads are obtained from the pool of available ready to run threads and they run on the system CPUs.

Advantages of JAVA:-

It is an open source, so users do not have to struggle with heavy license fees each year

1.Platform independent

2.Java API's can easily be accessed by developers

3.Java supports garbage collection, so memory management is automatic

4.Java always allocates objects on the stack

5.Java embraced the concept of exception specifications

6.Multi-platform support language and support for web-services

7.Using Java we can develop dynamic web applications

8.It allows you to create modular programs and reusable codes

JAVA SERVER PAGES:

Java server pages is a simple but powerful technology used to generate dynamic web pages on the server side. JSP's are direct extension of java servlets and provide a way to separate content generation from content presentation.

FEATURES OF JSP:

1.Portability

Java Server Pages files can be run on any web server or web-enabled application server that provides support for them. Dubbed the JSP engine, this support involves recognition, translation and management of the Java Server Page lifecycle and its interaction components.

2.Components

It was mentioned that the Java Server Pages architecture can include reusable Java components. The architecture also allows for the embedding of a scripting language directly into the Java Server Pages file. The components current supported include Java Beans and Servlets.

3.Processing

A Java Server Pages file is essentially an HTML document with JSP scripting or tags. The Java Server Pages file has extension to the server as a Java Server Pages file. Before the page is served, the Java Server Pages syntax is parsed and processed into a Servlet on the server side. The Servlet that is generated outputs real content in straight HTML for responding to the client.

HTML:-

HTML, which stands for Hypertext Markup Language, is the predominant markup language for web pages. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It allows images and object to be embedded and can be used to create interactive forms. It is written in the form of HTML elements consisting of “tags” surrounded by angle brackets within the web page content. It can embed scripts in languages such as JavaScript which affect the behavior of HTML web pages. HTML can also be used to include Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both HTML and CSS standards, encourages the use of CSS over explicit presentational markup.

HTML TAGS:-

Basic HTML Tags:

<!-- -->	Specifies comments
<A>.....	Creates hypertext links
.....	Formats text as bold
<BIG>.....</BIG>	Formats text in large font

<BODY>.....</BODY>	Contains all tags and text in the HTML documents
<CENTER>.....</CENTER>	Aligns text center
<DD>.....</DD>	Definition of a term
<DL>.....</DL>	Creates definition list
.....	Formats text with a particular font
<FORM>.....</FORM>	Encloses a fill-out form
<FRAME>.....</FRAME>	Defines a particular frame in a set of frames
<H#>.....</H#>	Creates headings of different levels
<HEAD>.....</HEAD>	Contains tags that specify information about a document
<HR>.....</HR>	Creates a horizontal rule
<HTML>.....</HTML>	Contains all other HTML tags
<META>.....</META>	Provides meta-information about a document
<SCRIPT>.....</SCRIPT>	Contains client-side or server-side script
<TABLE>.....</TABLE>	Creates a table
<TR>.....</TR>	Designates a table row
<TH>.....</TH>	Creates a heading in a table

Advantages:

- 1.A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.
- 2.HTML is platform independent.
- 3.HTML tags are not case-sensitive.

JDBC:-

JDBC is an API for the java programming language that defines how a client may access a database. It provides methods for querying and updating data in a database. JDBC is oriented towards relational databases.

JDBC was first introduced in the Java2 Platform, Standard Edition, version 1.1(J2SE), together with a reference implementation JDBC-to-ODBC bridge, enabling connections to any ODBC-accessible data source in the JVM host environment.

Functionality:

JDBC allows multiple implementations to exist and be used by the same application. The API provides a mechanism for dynamically loading the correct java packages and registering them with the JDBC Driver Manager. The Driver Manager is used as a connection factory for creating JDBC connections.

JDBC connections support creating and executing statements. These may be update statements such as SQL's CREATE, INSERT, UPDATE and DELETE or they may be query statements such as SELECT. Additionally, stored procedures may be invoked through a JDBC connection. JDBC represents statements using one of the following classes:

- 1.Statement** – the statement is sent to the database server each and every time.
- 2.Prepared Statement**- the statement is cached and then the execution path is pre-determined on the database server allowing it to be executed multiple times in an efficient manner.
- 3.Callable Statement**-used for executing stored procedures on the database.

Update statements such as INSERT, UPDATE and DELETE return an update count that indicates how many rows were affected in the database. These statements do not return any other information.

Query statements return a JDBC row result set. The row result set is used to walk over the result set. Individual columns in a row are retrieved either by name or by column number. There may be any number of rows in the result set. The row result set has metadata that describes the names of the columns and their types.

JDBC Drivers:

JDBC drivers are client-side adapters (installed on the client machine, not on the server) that convert requests from java programs to a protocol that the DBMS can understand.

JDBC drivers fall under four categories:

- 1.TYPE-1
- 2.TYPE-2
- 3.TYPE-3
- 4.TYPE-4

Type 1 driver- JDBC-ODBC bridge

The JDBC type 1 driver, also known as the JDBC-ODBC bridge, is a database driver implementation that employs the ODBC driver to connect the database. The driver converts JDBC method calls into ODBC function calls.

This driver leads to other installation dependencies; for example, ODBC must be installed on the computer having the driver and the database must support an ODBC driver. The use of this driver is discouraged if the alternative of a pure-Java driver is available.

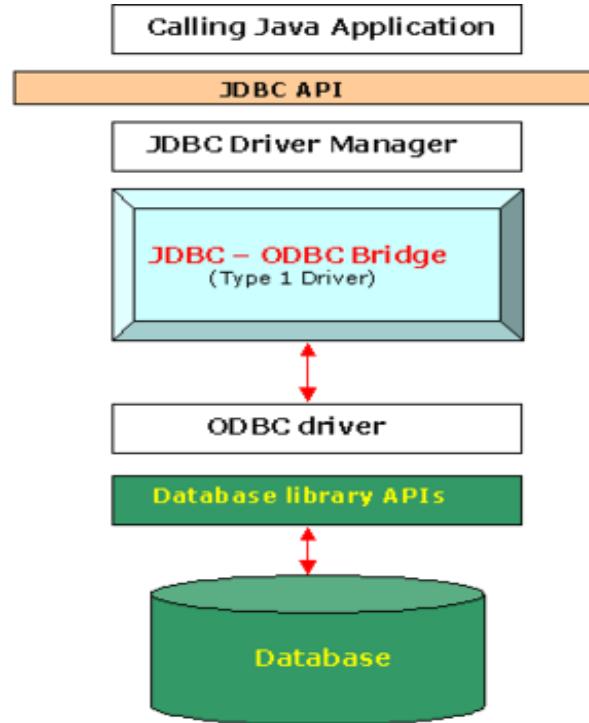


Figure 1 Type 1 Driver Architecture

Functions:-

1. Translates query obtained by JDBC into corresponding ODBC query, which is then handled by the ODBC driver.
2. Sun provides a JDBC-ODBC Bridge driver. `Sun.jdbc.odbc.JdbcDriver`. This driver is native code and not Java, and is closed source.
3. Client -> JDBC Driver -> ODBC Driver -> Database
4. There is some overhead with the translation work to go from JDBC to ODBC.

Advantages:

1. Almost any database for which ODBC driver is installed, can be accessed.
2. A type 1 driver is easy to install.

Disadvantages:

1. The driver is platform-dependent as it makes use of ODBC which is in turn depends on native libraries of the underlying operating system the JVM is running upon.
2. This technology isn't suitable for a high-transaction environment.

- 3.This driver is non-portable given the binding between the driver and platform.
- 4.This drivers also don't support the complete Java command set and are limited by the functionality of the ODBC driver.

Type 2 Driver – Native – API Driver specification:

The JDBC type 2 driver, also known as the Native –API driver, is a database driver implementation that uses the client-side libraries of the database. The driver converts JDBC method calls into native calls of the database API.

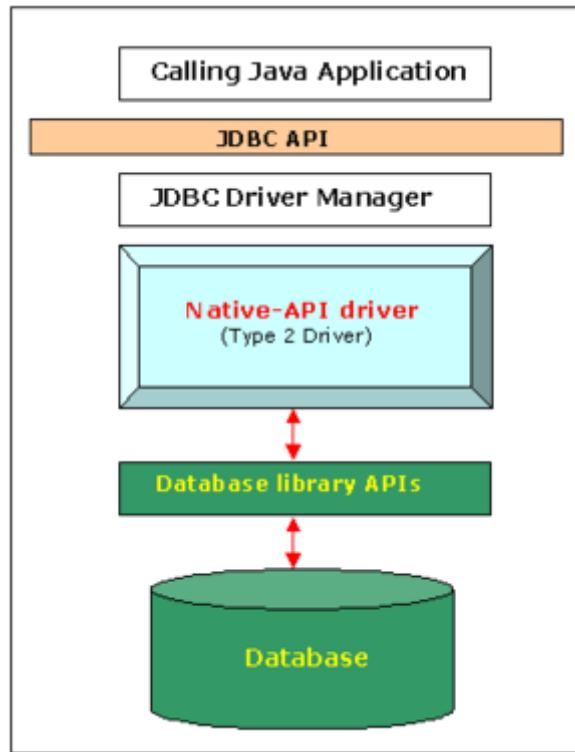


Figure 2Type 2 Driver Architecture

The type 2 driver is not written entirely in **java** as it interfaces with non-Java code that makes the final database calls. The driver is compiled for use with the particular operating system. For platform interoperability, the Type 4 driver, being a full-Java implementation, is preferred over this driver.

However the type 2 driver provides more functionality and better performance than the type 1 driver as it does not have the overhead of additional ODBC function calls.

Advantages

- 1.Better performance than Type 1 Driver(JDBC-ODBC Bridge)
- 2.Provides Fastest performance than all 3 drivers as it calls native APIs (MySQL, Oracle etc.)

Disadvantages

- 1.The vendor client library has to be installed on the client machine.
- 2.Cannot be used in web-based application due to the client side software needed.
- 3.Not all databases have client side library.
- 4.This server is platform dependent.
- 5.This driver supports all java applications except Applets.

Type 3 Driver – Network – Protocol Driver

The JDBC type 3 driver, also known as the Pure Java Driver for Database Middleware, is a database driver implementation which makes use of a middle tier between the calling program and the database.

The middle-tier (application server) converts JDBC calls directly or indirectly into the vendor-specific database protocol.

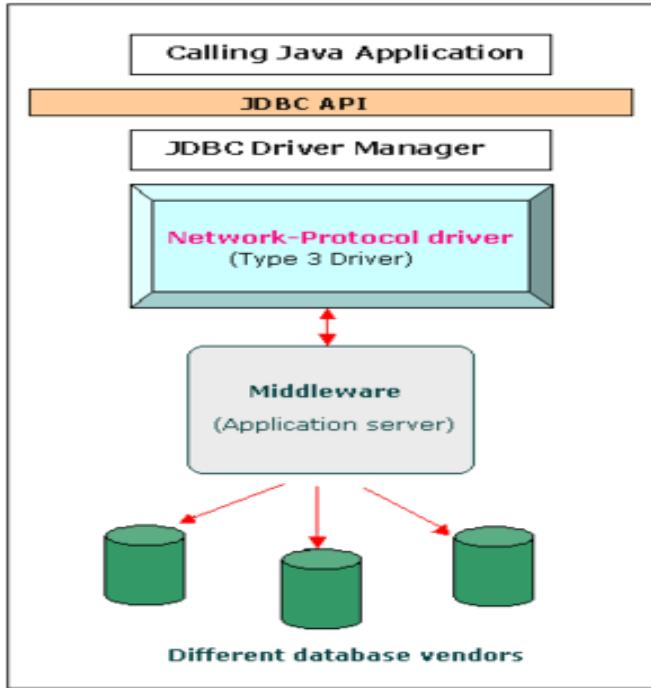


Figure 3 Type 3 Driver Architecture

This differs from the type 4 driver in that the protocol conversion logic resides not at the client, but in the middle-tier. Like type 4 drivers, the type 3 driver is written entirely in Java. The same driver can be used for multiple databases. It depends on the number of databases the middleware has been configured to support. The type 3 driver is platform –independent as the platform-related differences are taken care by the middleware. Also, making use of the middleware provides additional advantages of security and firewall access.

Functions:-

- 1.Follows a three tier communication approach.
- 2.Can interface to multiple databases –Not vendor specific.
- 3.The JDBC Client driver written in Java, communicates with a middleware-net-server using a database independent protocol, and then this net server translates this request into database commands for that database.
- 4.Thus the client driver to middleware communication is database independent.
- 5.Client -> JDBC Driver -> Middleware-Net-Server -> Any Database.

Advantages

- 1.Since the communication between client and the middleware server is database independent, need not be changed for a new database.

2.The Middleware Server (which can be a full fledged J2EE Application server) can provide typical middleware services like caching(connections, query results, and so on), load balancing, logging, auditing etc.

3.Eg. For the above include jdbc driver features in Weblogic.

- Can be used in internet since there is no client side software needed.
- At client side a single driver can handle any database. (It works provided the middleware supports that database)

Disadvantages

- 1.Requires database-specific coding to be done in the middle tier.
- 2.An extra layer added may result in a time-bottleneck. But typically this is overcome by providing efficient middleware services.

Type 4 Driver - Native-Protocol Driver:

The JDBC type 4 driver, also known as the Direct to Database Pure Java Driver, is a database driver implementation that converts JDBC calls directly into the vendor-specific database protocol.

The type 4 driver is written completely in Java and hence platform independent. It is installed inside the Java Virtual Machine of the client. It provides better performance over the type 1 and 2 drivers as it does not have the overhead of conversion of calls into ODBC or database API calls. Unlike the type 3 drivers, it does not need associated software to work.

As the database protocol is vendor- specific, separate drivers, usually vendor-supplied, need to be used to connect to different databases.

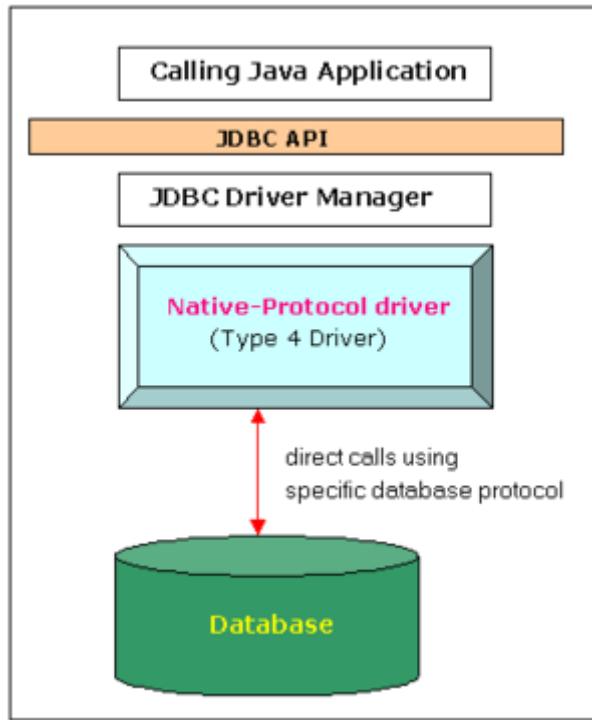


Figure 4Type 4 Driver Architecture

FUNCTIONS:-

- 1.Type 4 drivers are entirely written in Java that communicates directly with a vendor's database, usually through socket connections. No translation or middleware layers are required, improving performance.
- 2.The driver converts JDBC calls into the vendor-specific database protocol so that client application can communicate directly with the database server.
- 3.Completely implemented in Java to achieve platform independence.
- 4.Eg: include the widely used Oracle thin driver – `oracle.jdbc.driver.OracleDriver` which connect to `jdbc:oracle:thin` URL format.
- 5.Client -> Native Protocol JDBC Driver -> Database server

Advantages

- 1.These drivers don't translate the requests into an intermediary format (such as ODBC), nor do they need a middleware layer to service requests. Thus the performance may be considerably improved.

2.All aspects of the application to database connection can be managed within the JVM; this can facilitate easier debugging.

Disadvantages

- 1.At client side, a separate driver is needed for each database.

STAGES IN A JDBC PROGRAM:

1.REGISTERING THE DRIVER:

A database driver is software containing the classes and interfaces written according to the JDBC API. Since there are several drivers available in the market, we should first declare the driver which is going to be used for communication with the database server in a java program.

2.CONNECTING TO A DATABASE:

In this stage we establish the connection with a specific database through the driver which is already registered in the previous step.

3.PREPARING SQL STATEMENT:

We should create SQL statements in our java program using any one of the interfaces like Statement, Prepared Statement, Callable Statement which are available in java.sql package.

4.EXECUTING THE SQL STATEMENTS ON THE DATABASE:

For this purpose, we can use executeUpdate(), executeQuery(), methods of statement Interface.

5.RETRIEVING THE RESULTS:

The Results obtained by executing the SQL statements can be stored in an object with the help of interfaces like Result Set.

6.CLOSING THE CONNECTION:

We should close the connection between the Java program and the database using close() method of connection Interface.

DATABASE

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.

Other kinds of data stores can be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those type of systems.

So now-a-days we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as foreign keys.

A Relational Database Management System (RDBMS) is a software that: Enables you to implement a database with tables, columns, and indexes. Guarantees the Referential integrity between rows of various tables. Updates the indexes automatically. Interprets an SQL query and combines information from various tables.

RDBMS Terminology:

Database: A database is a collection of tables, with related data.

Table: A table is a matrix with data. A table in a database looks like a simple spreadsheet.

Column: One column (data element) contains data of one and the same kind, for example the column postcode.

Row: A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.

Redundancy: Storing data twice, redundancy to make the system faster.

Primary Key: A primary key is unique. A key value cannot occur twice in one table. With a key you can find at most one row.

Foreign Key: A foreign key is the linking pin between two tables.

Compound Key: A compound key (composite key) is a key that consist of multiple columns, because one column is not sufficiently unique.

Index: An index in a database resembles an index at the back of a book.

Referential Integrity: Referential Integrity makes sure that a foreign key value always points to existing row.

CHAPTER 9

SYSTEM DESIGN

UML DIAGRAMS

A diagram is a graphical presentation of a set of elements, most often rendered as a connected graph. UML includes nine such diagrams. Unified Modeling Language (UML) is probably the most widely known and used notation for object-oriented methods. The Unified Modeling Language (UML) is a standard language for writing software blueprints.

The UML may be used to visualize, specify, construct and document the artifacts. A Modeling Language is a language whose vocabulary and rules focus on the conceptual and the physical and the physical representation of a system. Modeling is the designing of the software applications before coding. It is an essential part of large software projects, and helpful to medium and even small projects as well. A model plays an analogous role in software development project's success can assure themselves that business functionality is complete and correct. Care should be taken that end-user's needs are met.

The underlying premise of UML is that no one diagram can capture the different elements of a system at different points of time in the software life cycle of a system.

The UML diagrams are as follows:

- ❖ Use case Diagram
- ❖ Class Diagram
- ❖ Sequence Diagram
- ❖ Activity Diagram

Use Case Diagram: The use case diagram is used to identify the primary elements and processes that form the system. The primary elements are termed as “actors” and the processes are called as “use cases”. The use case diagram shows which actors interact with each use case.

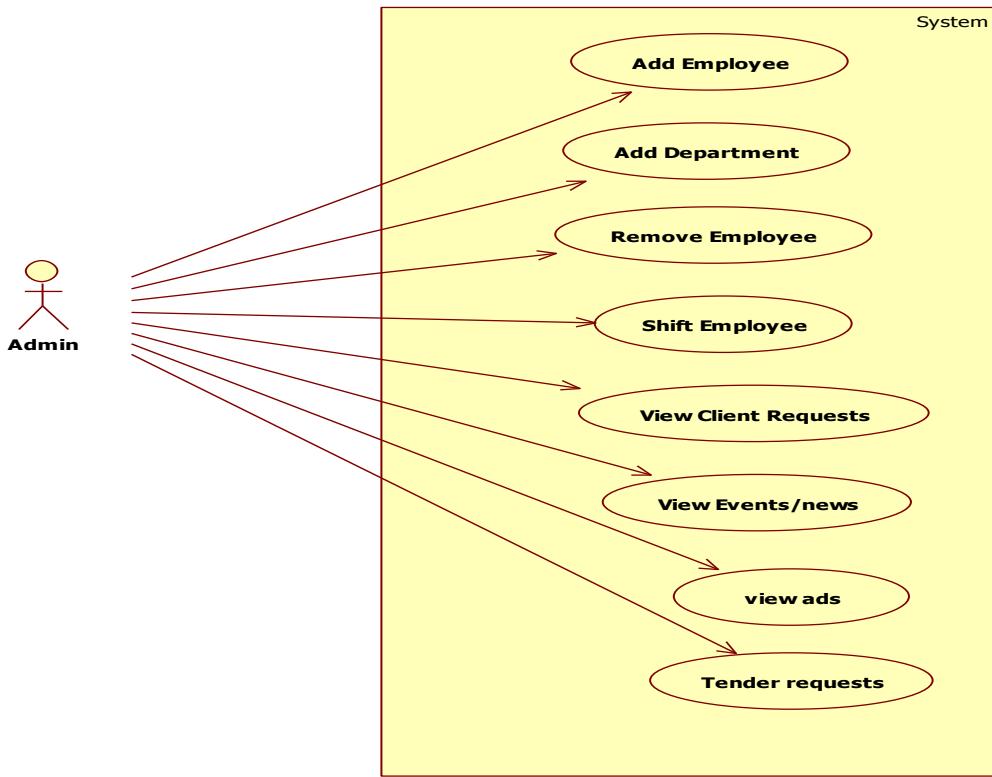


Figure 5 Admin Use Case Diagram

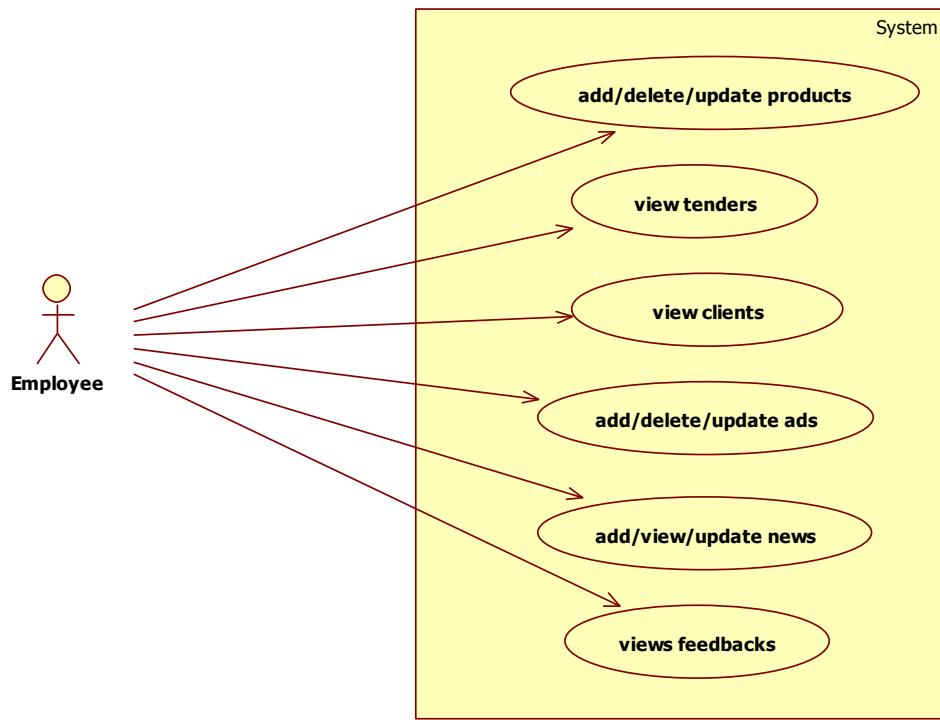


Figure 6 Employee Use Case Diagram

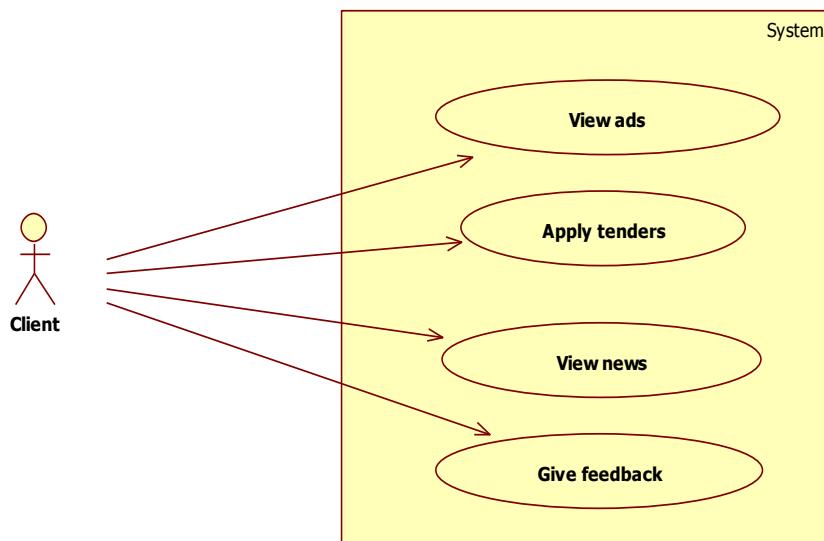


Figure 7 Client Use Case Diagram

Class Diagram: The class diagram is used to define a detailed design of the system. It classifies the actors into a set of interrelated classes. The relationship or association between the classes can be either an “is-a” or “has-a” relationship. Each class in the class diagram may be capable of providing certain functionalities. These functionalities provide by the class are termed “methods” of the class. Apart from this, each class may have certain “attributes” that uniquely identify the class.

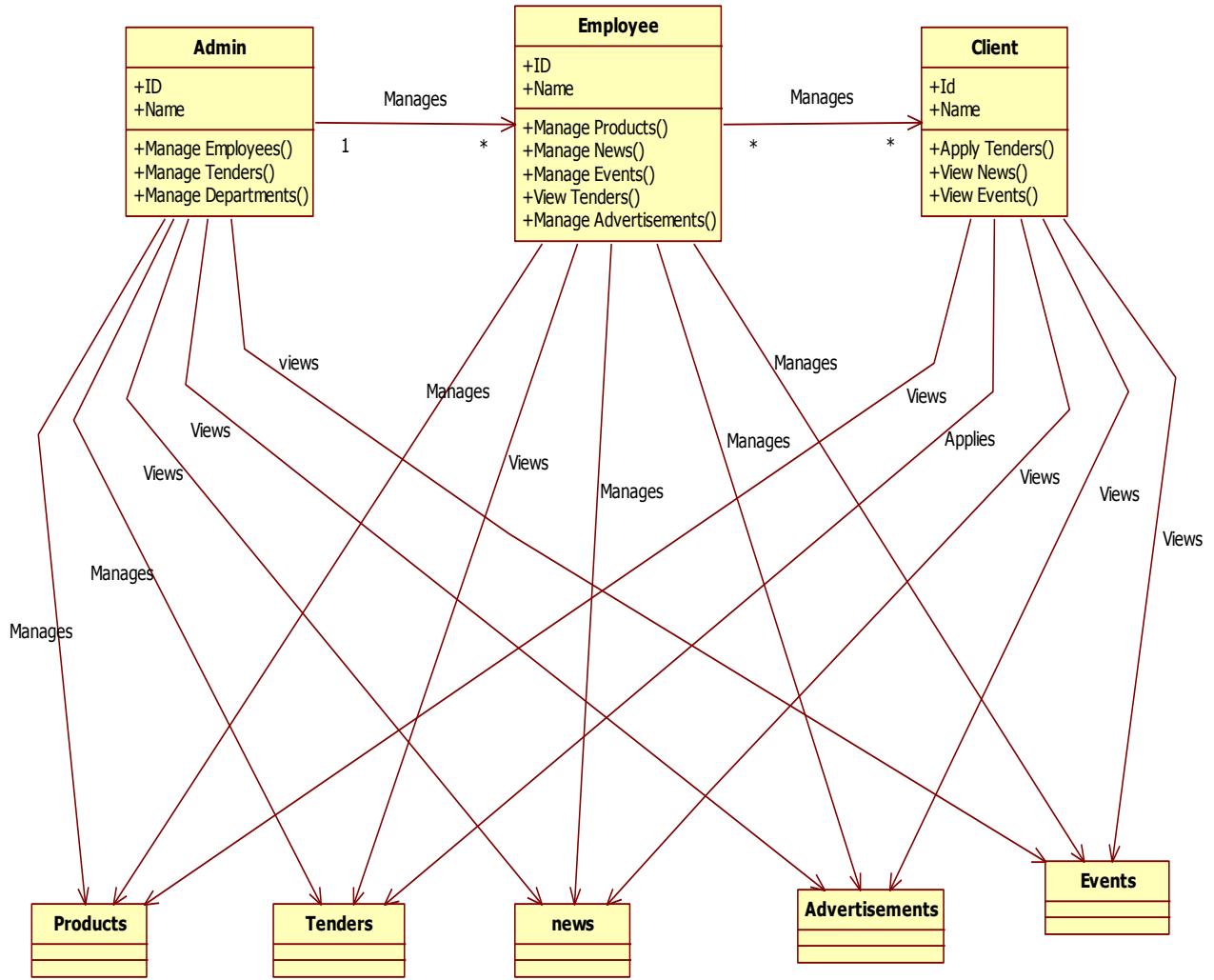


Figure 8 Class Diagram

Sequence Diagram: A sequence diagram represents the interaction between different objects in the system. The important aspect of a sequence diagram is that it is time-ordered. This means that the exact sequence of the interactions between the objects is represented step by step. Different objects in the sequence diagram interact with each other by passing “messages”.

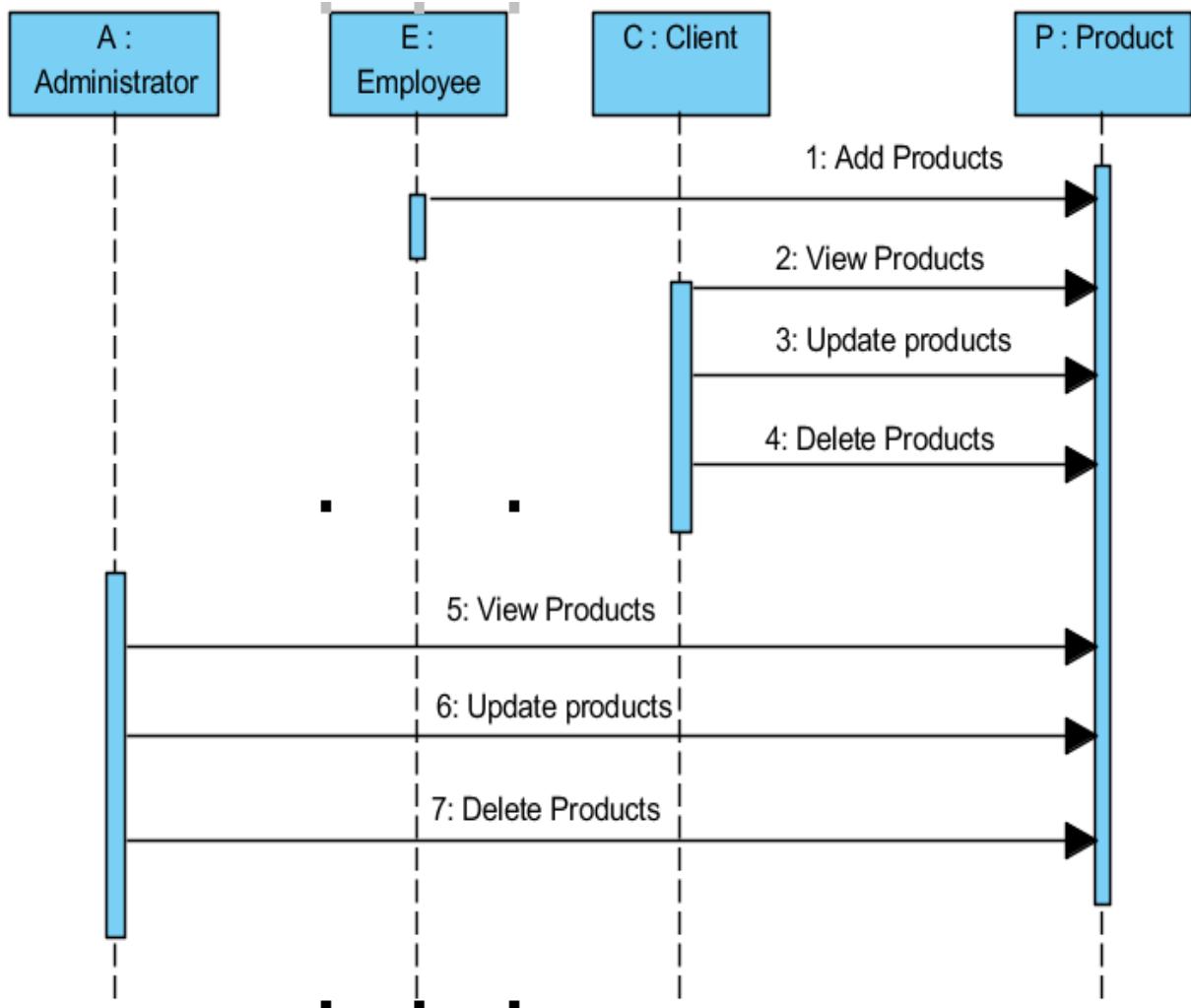


Figure 9 Sequence Diagram

Activity Diagram: The process flows in the system are captured in the activity diagram. Similar to a state diagram, an activity diagram also consists of activities, actions, transitions, initial and final states, and guard conditions.

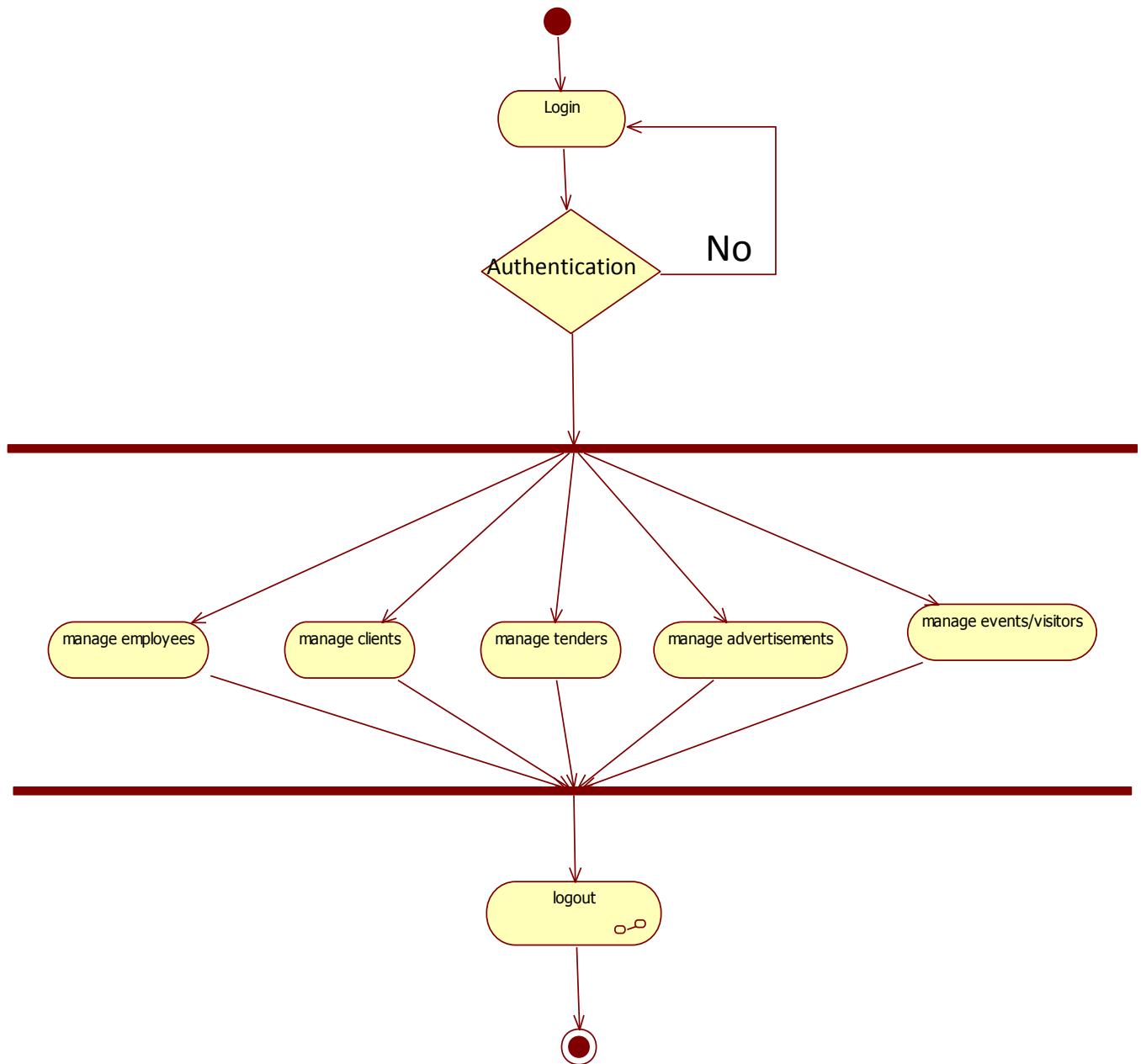


Figure 10 Activity Diagram for Admin

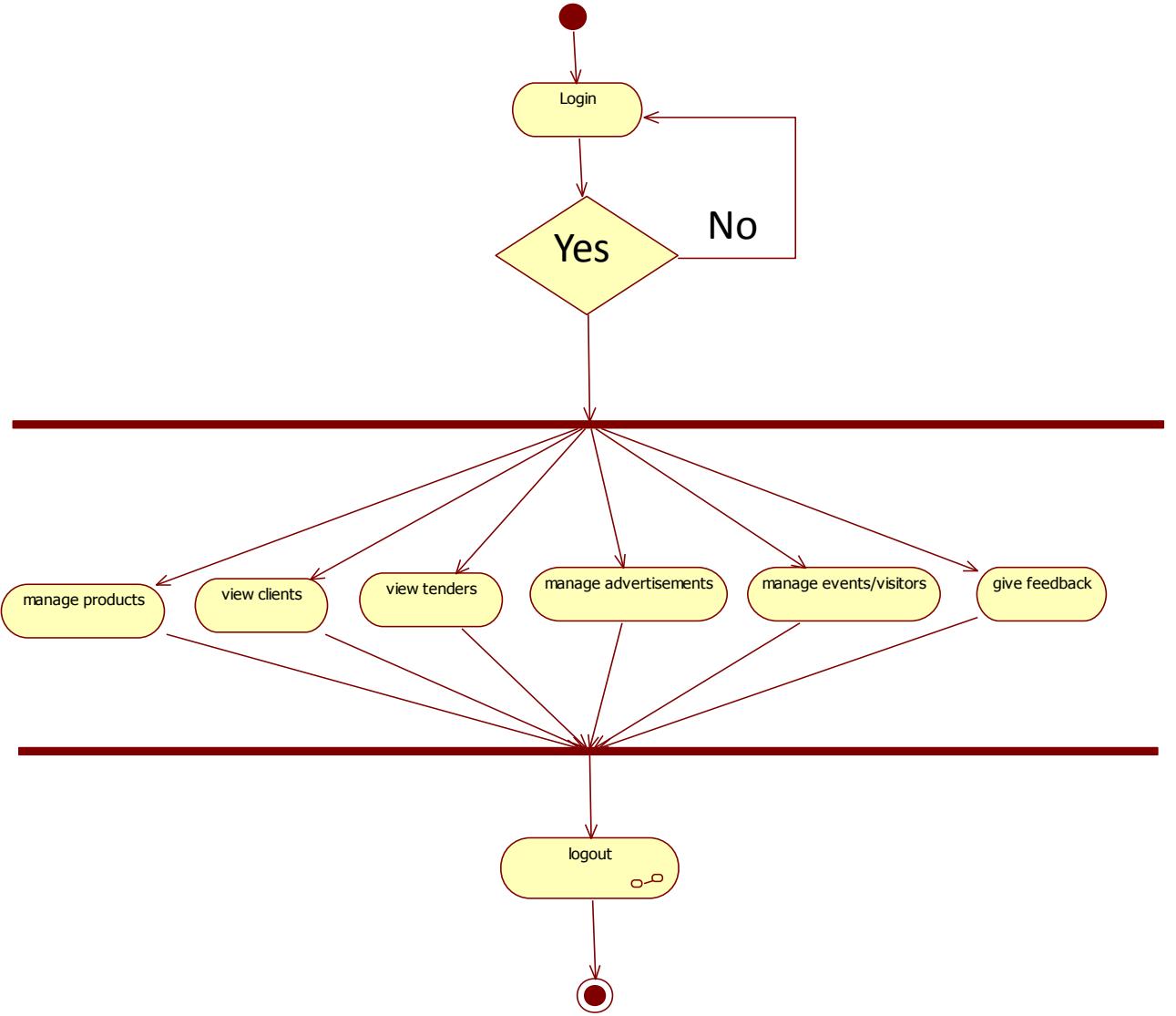


Figure 11 Activity Diagram for Employee

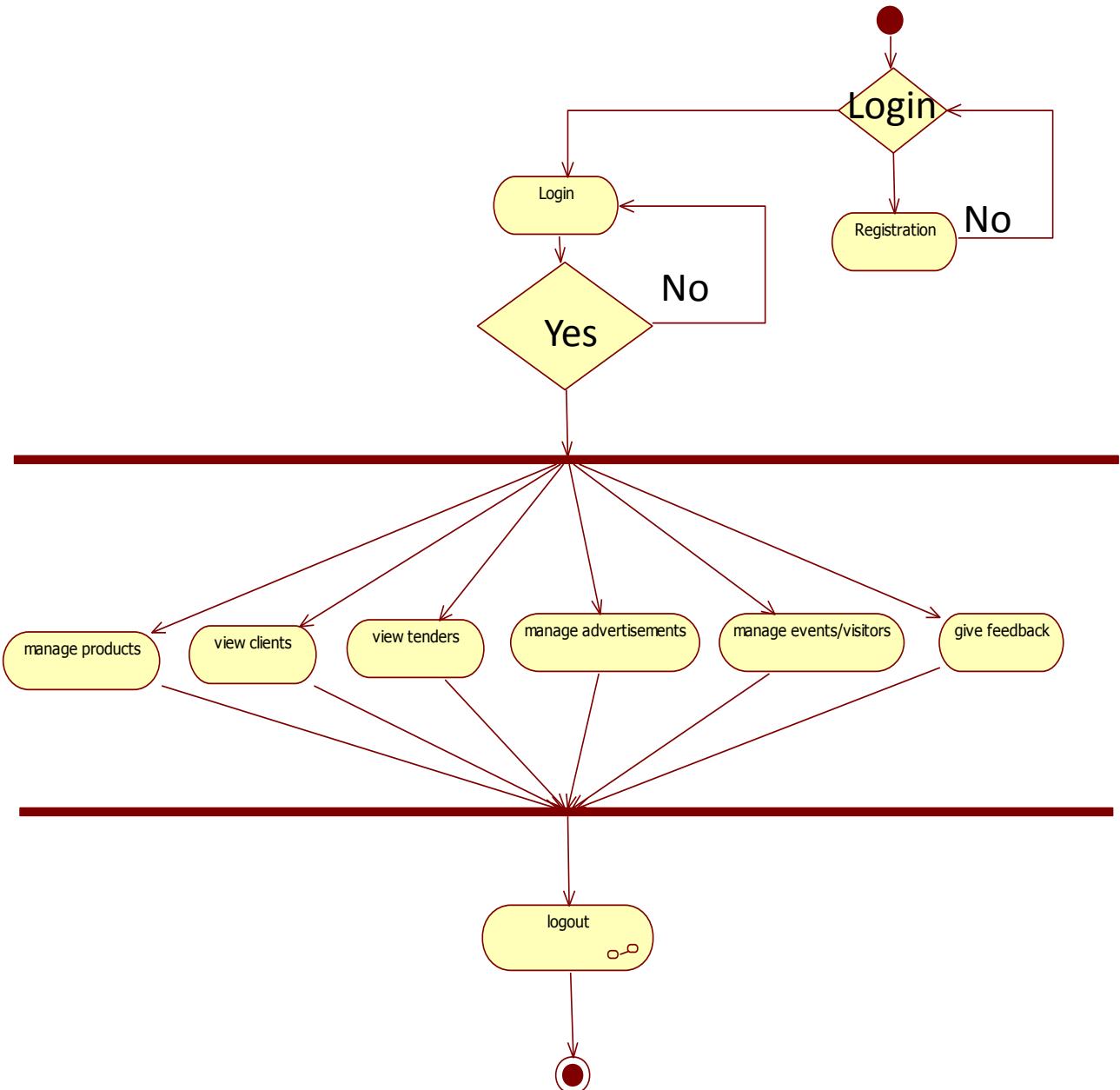


Figure 12 Activity Diagram for Client

CHAPTER 10

DATABASE DESIGN

11.1 TABLES

- ❖ Tables are logical structures maintained by the database manager. Tables are made up of columns and rows.
- ❖ At the intersection of every column and row is a specific data item called a value. A column is a set of values of the same type or one of its subtypes. A row is a sequence of values arranged so that the nth value is a value of the nth column of the table.
- ❖ When designing tables, you need to be familiar with certain concepts, determine the space requirements for tables and user data, and determine whether you will take advantage of certain features, such as space compression and optimistic locking.

EMPLOYEE TABLE:

Column Name	Data Type	Nullable	Default	Primary Key
EID	NUMBER	No	-	1
ENAME	VARCHAR2(4000)	No	-	-
EDOB	VARCHAR2(40)	No	-	-
EADDRESS	VARCHAR2(4000)	No	-	-
EPH	NUMBER(10,0)	No	-	-
EGENDER	VARCHAR2(4000)	No	-	-
EDEPT	VARCHAR2(4000)	No	-	-
EEMAIL	VARCHAR2(4000)	No	-	-
PASSWORD	VARCHAR2(4000)	No	-	-
CONFIRM_PASSWORD	VARCHAR2(4000)	No	-	-
EMP_PHOTO	BLOB	Yes	-	-
EDEPT_ID	NUMBER(10,0)	Yes	-	-

1 - 12

PRODUCTS TABLE:

Column Name	Data Type	Nullable	Default	Primary Key
PID	NUMBER	No	-	1
PNAME	VARCHAR2(4000)	No	-	-
P_PRICE	NUMBER	No	-	-
EID	NUMBER	No	-	-
DATE_TIME	VARCHAR2(100)	No	-	-
PRODUCT_IMAGE	BLOB	Yes	-	-
1 - 6				

TABLE FOR TENDER:

Column Name	Data Type	Nullable	Default	Primary Key
TNAME	VARCHAR2(4000)	No	-	-
TCOMPANY_NAME	VARCHAR2(4000)	No	-	-
TCOMPANY_PH	NUMBER(10,0)	No	-	-
TCOMPANY_EMAIL	VARCHAR2(4000)	No	-	-
T_DESC	VARCHAR2(4000)	No	-	-
TID	NUMBER	No	-	-
TDATE	VARCHAR2(4000)	No	-	-
TENDER_STATUS	VARCHAR2(4000)	Yes	-	-
P_ID	NUMBER	No	-	-
CLIENT_ID	NUMBER	No	-	-
PRODUCT_NAME	VARCHAR2(4000)	Yes	-	-
TENDER_DOC	BLOB	Yes	-	-
1 - 12				

TABLE FOR ADVERTISEMENTS:

Column Name	Data Type	Nullable	Default	Primary Key
AD_ID	NUMBER	No	-	1
ADNAME	VARCHAR2(4000)	No	-	-
PNAME	VARCHAR2(4000)	No	-	-
EID	NUMBER	No	-	-
AD_LAST_DATE	VARCHAR2(4000)	No	-	-
P_ID	NUMBER	No	-	-
AD_IMAGE	BLOB	Yes	-	-
1 - 7				

TABLE FOR NEWS:

Column Name	Data Type	Nullable	Default	Primary Key
NID	VARCHAR2(100)	No	-	1
NTITLE	VARCHAR2(4000)	No	-	-
NCONTENT	VARCHAR2(4000)	No	-	-
EID	NUMBER	No	-	-
NDATE_TIME	VARCHAR2(100)	No	-	-
NEWS_IMAGE	BLOB	Yes	-	-
1 - 6				

TABLE FOR EVENTS:

Column Name	Data Type	Nullable	Default	Primary Key
EVENT_ID	NUMBER	No	-	1
EVENT_NAME	VARCHAR2(4000)	No	-	-
EVENT_DESC	VARCHAR2(4000)	Yes	-	-
EVENT_DATE	VARCHAR2(4000)	No	-	-
EID	NUMBER	Yes	-	-
				1 - 5

FEEDBACK TABLE:

Column Name	Data Type	Nullable	Default	Primary Key
CID	NUMBER	No	-	1
CLIENT_NAME	VARCHAR2(4000)	No	-	-
REMARK	VARCHAR2(4000)	No	-	-
				1 - 3

DEPARTMENT TABLE:

Column Name	Data Type	Nullable	Default	Primary Key
DID	NUMBER	No	-	1
DNAME	VARCHAR2(4000)	No	-	2
DNO_OF_EMP	NUMBER	No	-	-
D_ADDED_DATE	VARCHAR2(100)	No	-	-
1 - 4				

TABLE FOR CLIENT:

Column Name	Data Type	Nullable	Default	Primary Key
CID	NUMBER	No	-	1
CNAME	VARCHAR2(100)	No	-	-
CDESIG	VARCHAR2(100)	No	-	-
C_COMPANY_NAME	VARCHAR2(100)	No	-	-
C_ADDRESS	VARCHAR2(100)	No	-	-
C_PH	NUMBER(10,0)	No	-	-
C_DATE_AND_TIME	VARCHAR2(40)	No	-	-
CPASS	VARCHAR2(100)	No	-	-
C_CONFIRM_PASS	VARCHAR2(100)	No	-	-
C_EMAIL	VARCHAR2(4000)	No	-	-
CLIENT_STATUS	VARCHAR2(4000)	Yes	-	-
CLIENT_PHOTO	BLOB	Yes	-	-
1 - 12				

CHAPTER 11

TEST REPORT, TEST PLAN

Test Reports, Test Plan:

Software testing is a process of validating and verifying that a software program /application/product meets:

1. The business and technical requirements that guided its design and development
2. Works as expected
3. Can be implemented with the same characteristics.

Purpose:

A primary purpose for testing is to detect software failures so that defects may be uncovered and corrected. Software testing includes examination of code as well as execution of that code in various environments and conditions.

It also includes examining the aspects of code: does it do what it is supposed to do and do what it needs to do. Information derived from software testing may be used to correct the process by which software is developed.

Testing Methods:

Software testing methods are divided into White, Black and Grey box testing. These approaches are used to describe the point of view that a test engineer takes when designing test cases.

1. White Box Testing:

White box testing is done when the tester has access to the internal data structures and algorithms including the code that implements it.

2. Black Box Testing:

Black box testing treats the software as a “black box” without any knowledge of internal implementation.

3. Grey Box Testing:

Grey box testing involves having knowledge of internal data structures and algorithms for purpose of designing the test cases, but testing at the user, or black-box level. Grey box testing may also include reverse engineering to determine, for instance, boundary values or error messages.

Testing Levels:

Tests are frequently grouped by where they are added in the software development process, or by the level of specificity of the test.

1. Unit Testing:

Unit testing refers to tests that verify the functionality of a specific section of code, usually at the function level. In the object-oriented environment, this usually at class level, and the minimal unit tests include the constructors and destructors.

2.Integration testing:

Integration testing is any type of software testing that seeks to verify the interfaces between components and expose defects in it.

3.System testing:

System testing tests a completely integrated system to verify that it meets its requirements.

4.System Integration Testing:

System integration testing verifies that a system is integrated to any external or third party systems defined in the system requirements.

Test Report 1:

1.Project Name: WEBSITE PUBLISHING SYSTEM

2.Form Name: Login

3.Unit Name: User-id, Password

4.Test Result: After entering two fields the user successfully logs into the system

Test Plan 1:

Unit id : Login

Test Case id : User-id

Test Type : Unit Testing

Form Name : Login

Base Table : Admin/Employee/Customer

Purpose : To give access to the user after he/she enters valid user-id and password.

Description:

User id : varchar (50)

Password : varchar (50)

Test data:

Serial No.	Input Specification	Output Specification
1	User id: Valid Input	Navigates to the respective home pages.
	Invalid input	It will ask to enter correct values again.
2	Password: Valid Input	Navigates to the respective home page.
	Invalid input	It will ask to enter correct values again.

Test Process:

Login form will be used for allowing the correct user to use the software. Every person will be given a user id and password. After successful login the user can use the software as per the privileges given to him. The user id will be entered in the textbox given for user id. The password will be entered in the textbox given for password.

Test completion criteria:

When expected results match the actual results after performing the test, the test is considered to be complete.

Test Report 2:

1. Project name: Website Publishing System

2. Module name: Admin module

3. Unit name: Emp Id

Test Plan 2:

Unit id : Admin Home page

Test case Id : Emp Id

Test type : Unit Testing

Form name : Employee form

Base table : Employee

Purpose:

Employee Authorization table will be used to view, update and other privileges provided to the employee by administrator

By using the details the user will perform his/her operations which are allotted to that person

Test case description:

Emp_Id (50) primary key

Test Data:

S no	Input Specification	Output Specification
1.	Column name: Emp_Id Valid input	Navigates to allocated page
	Invalid input Column name: Password Valid input	Error message Navigates to allocated page
	Invalid input	Error Message

Test Process:

The Emp id will be entered in the textbox given for Employee id. If the given id matches with the emp id in database, that particular employees details are shown otherwise an error message is displayed.

Test completion criteria:

When expected results match the actual results after performing the test, the test is considered to be complete.

CHAPTER 12

SYSTEM IMPLEMENTATION

SAMPLE CODE:**After Admin Login:**

```
<%--  
Document : floattest  
Created on : Jun 26, 2014, 12:37:36 PM  
Author : Sai  
--%>
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>  
<!DOCTYPE html>  
<html>  
<head>  
  
<LINK REL=StyleSheet HREF="panel_style.css" TYPE="text/css" MEDIA=screen>
```

```

<LINK REL=StyleSheet HREF="panel_buttons_style.css" TYPE="text/css"
MEDIA=screen>

<title>ADMIN HOME PAGE</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<link rel="stylesheet" type="text/css" href="template.css">
<script type="text/javascript" src="js/jquery-2.1.1.js"></script>
<script type="text/javascript">
$(document).ready(function(){
    $(".trigger").click(function(){
        $(".panel").toggle("fast");
        $(this).toggleClass("active");
        return false;
    });
});

</script>

<script type="text/javascript">
window.history.forward();
function noBack()

```

```

{
    window.history.backward();
}

</SCRIPT>

</head>

<style>
body
{
    background: url(white.jpg);
    background-size: 1350px 1400px;
background-repeat: no-repeat;
}
</style>

<%
    String session_id= (String) session.getAttribute("id");
%>
<body onload="noBack();" onPageShow= "if(event.persisted) noBack();" onunload="">
<center>
<div id="">
    <h1 class="">&ampnbsp&ampnbsp&ampnbsp&ampnbsp&ampnbsp&ampnbsp&ampnbsp
    
    </h1>
</div></center>

```


© M@vericks

```
</h3>
</div>

<div class="panel" >
    <h3 style="color:white;"></h3>
    <p class="x"><b>Hello <%=session_id%>!</b></p>
    
    <div style="clear:both;"></div>
    <div class="columns">
        <div class="colleft">
            <ul>
                <li><a href="After Admin Login.jsp" class="push_button red">Home</a></li>
            </ul>
        </div>
        <div class="colright">
            <ul>
                <li><a href="logout_action.jsp" class="push_button blue">Logout</a></li>
            </ul>
        </div>
    </div>
```

```

</div>
<div style="clear:both;"></div>

</div>
<a class="trigger" style="font-family:Berlin Sans FB Demi;FONT-SIZE:20px" href="#" title="Click here for Navigation!">NAVIGATE</a>

<% if(session_id==null)
{
    response.sendRedirect("home_2.jsp");
} %>
</body>
</html>

```

After Employee Login:

```

<html>
<head>
    <title>EMPLOYEE HOME PAGE</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <link rel="stylesheet" type="text/css" href="template.css">
    <LINK REL=StyleSheet HREF="panel_style.css" TYPE="text/css" MEDIA=screen>
<LINK REL=StyleSheet HREF="panel_buttons_style.css" TYPE="text/css"
MEDIA=screen>
    <script type="text/javascript" src="js\jquery-2.1.1.js"></script>
    <script type="text/javascript">

```

```
$(document).ready(function(){
    $(".trigger").click(function(){
        $(".panel").toggle("fast");
        $(this).toggleClass("active");
        return false;
    });
});

</script>
<style>
body
{
    background: url(white.jpg);
    background-size: 1350px 1390px;
    background-repeat: no-repeat;
}
</style>
<script type="text/javascript">

    window.history.forward();
    function noBack()
    {
        window.history.backward();
    }

```



```
</h3>
</div>

<div class="panel" >
    <h3 style="color:white;"></h3>
    <p>
        class="x"><b>Hello <%=session_id%>!</b></p>
        
        <div style="clear:both;"></div>
        <div class="columns">
```

```

<div class="colleft">
    <ul>
        <li><a style="font-family:Gloucester MT Extra Condensed;font-size:30px"
            href="After Employee login .jsp" class="push_button red">Home</a></li>
    </ul>
</div>

<div class="colright">
    <ul>
        <li><a style="font-family:Gloucester MT Extra Condensed;font-size:30px"
            href="logout_action.jsp" class="push_button blue">Logout</a></li>
    </ul>
</div>

</div>
<div style="clear:both;"></div>

</div>
<a class="trigger" style="font-family:Berlin Sans FB Demi;FONT-SIZE:20px" href="#"
    title="Click here for Navigation!">NAVIGATE</a>

<% if(session_id==null)
{
    response.sendRedirect("home_2.jsp");
} %>

</body>

```

```
</html>
```

After Client Registration:

```
<html>
  <head>
    <title>CLIENT HOME PAGE</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <link rel="stylesheet" type="text/css" href="template.css">

    <LINK REL=StyleSheet HREF="panel_style.css" TYPE="text/css" MEDIA=screen>
    <LINK REL=StyleSheet HREF="panel_buttons_style.css" TYPE="text/css"
          MEDIA=screen>
    <script type="text/javascript" src="js\jquery-2.1.1.js"></script>
    <script type="text/javascript">
$(document).ready(function(){
  $(".trigger").click(function(){
    $(".panel").toggle("fast");
  });
  $(this).toggleClass("active");
  return false;
});
    </script>
    <style>
      body
```

```

{
    background: url(white.jpg);
    background-size: 1350px 1260px;
background-repeat: no-repeat;
}

</style>
<%
String session_id= (String)session.getAttribute("cid");
String cl_id= (String)session.getAttribute("client_id");
%>
<script type="text/javascript">

window.history.forward();
function noBack()

{
    window.history.backward();
}

</SCRIPT>
</head>
<body onload="noBack(); onPageShow= "if(event.persisted) noBack();"
onunload=""><center>

<div id="">
    <h1 class="gbts">&nbsp;&nbsp;&nbsp;&nbsp; 
</div></center>

```



```

</h3>
</div>

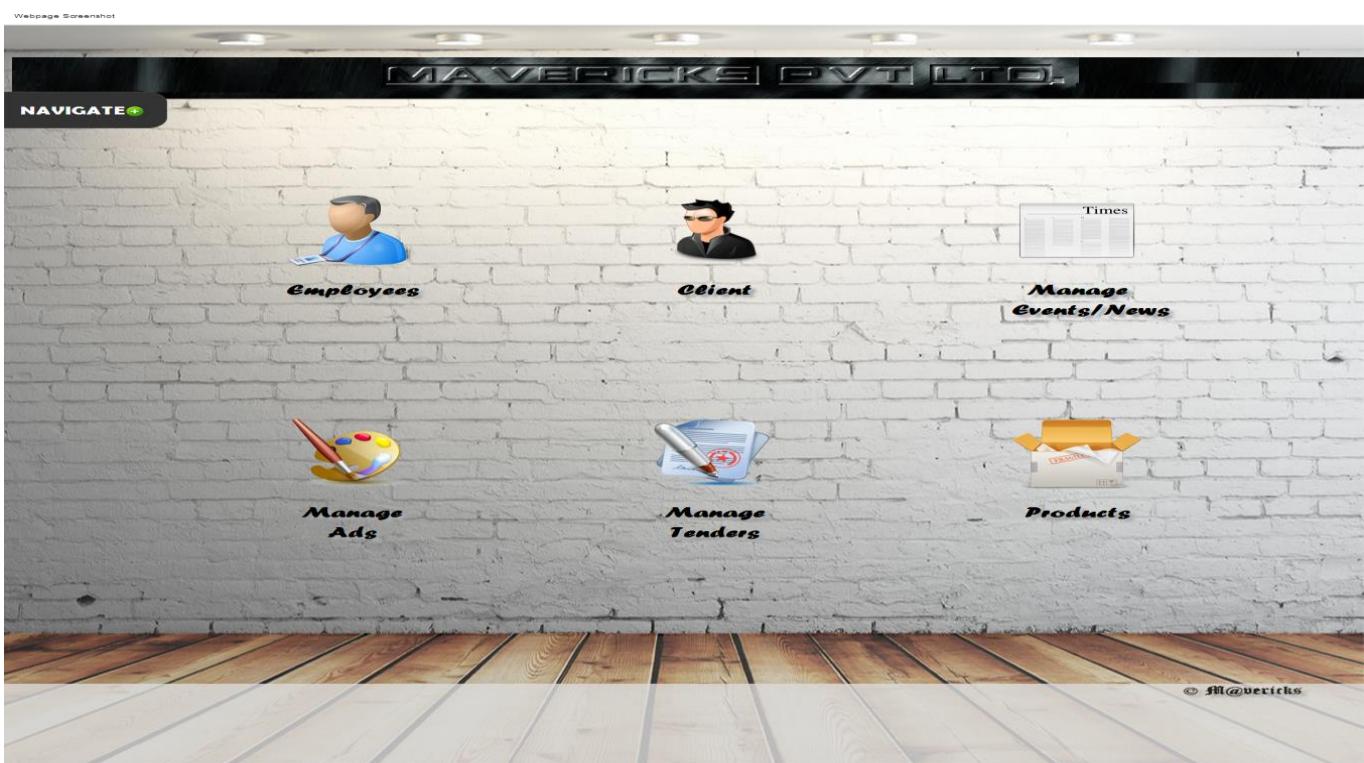
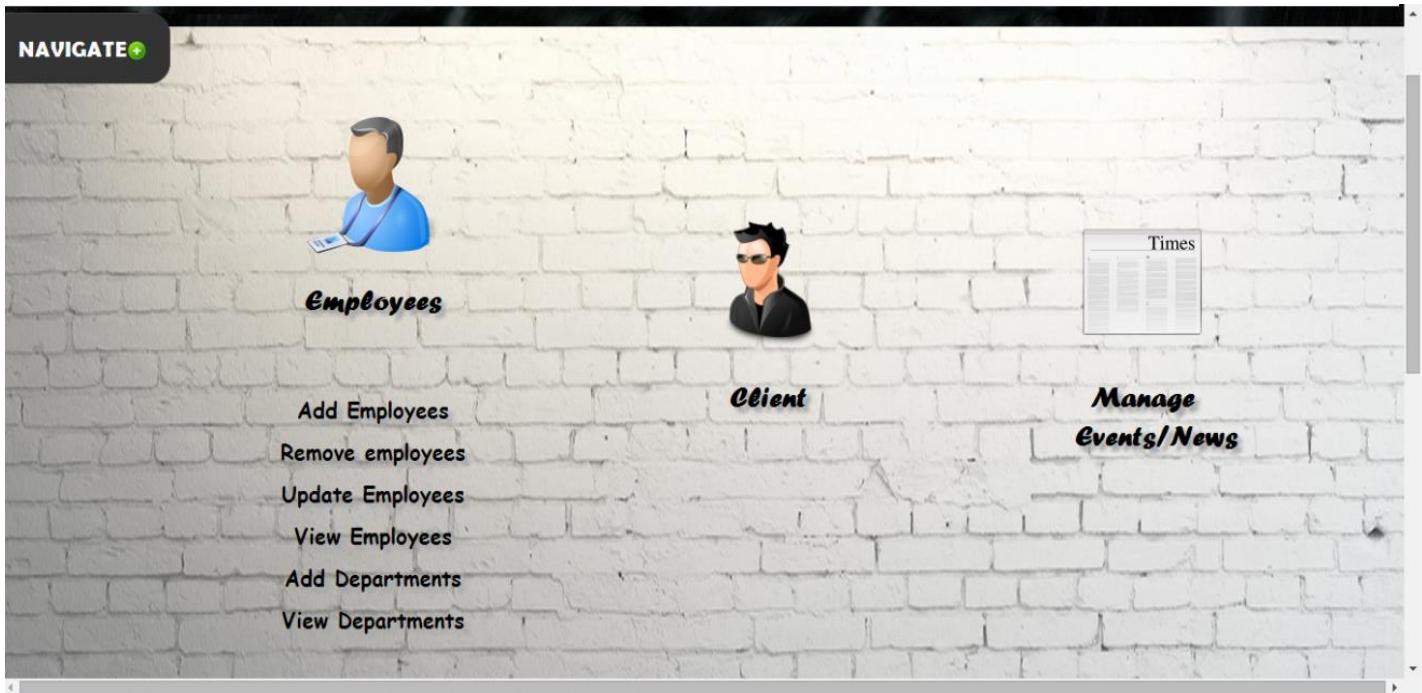
<div class="panel" >
    <h3
        style="color:white;"></h3>
    <p
        class="x"><b>Hello <%=session_id%>!</b></p>
    
    <div style="clear:both;"></div>
    <div class="columns">
        <div class="colleft">
            <ul>
                <li><a style="font-family:Gloucester MT Extra Condensed;font-size:30px"
                    href="After Admin Login.jsp" class="push_button red">Home</a></li>
            </ul>
        </div>
        <div class="colright">
            <ul>
                <li><a style="font-family:Gloucester MT Extra Condensed;font-size:30px"
                    href="logout_action.jsp" class="push_button blue">Logout</a></li>
            </ul>
        </div>
    </div>
    <div style="clear:both;"></div>

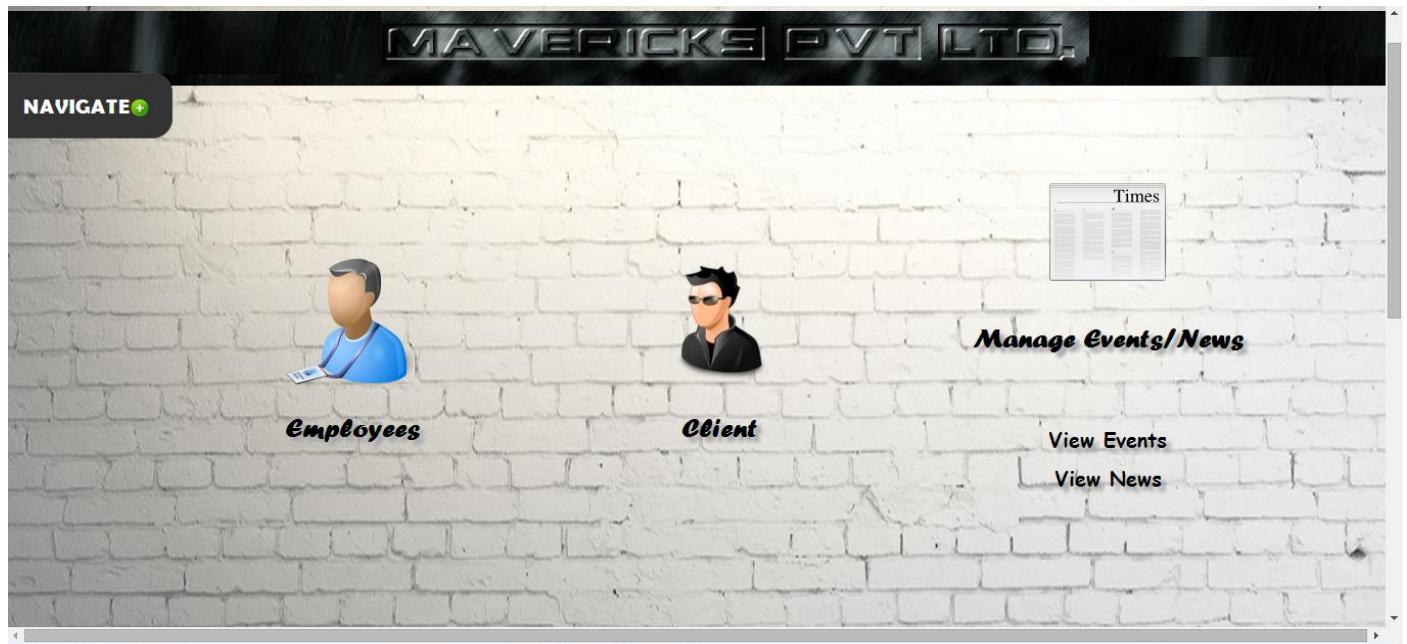
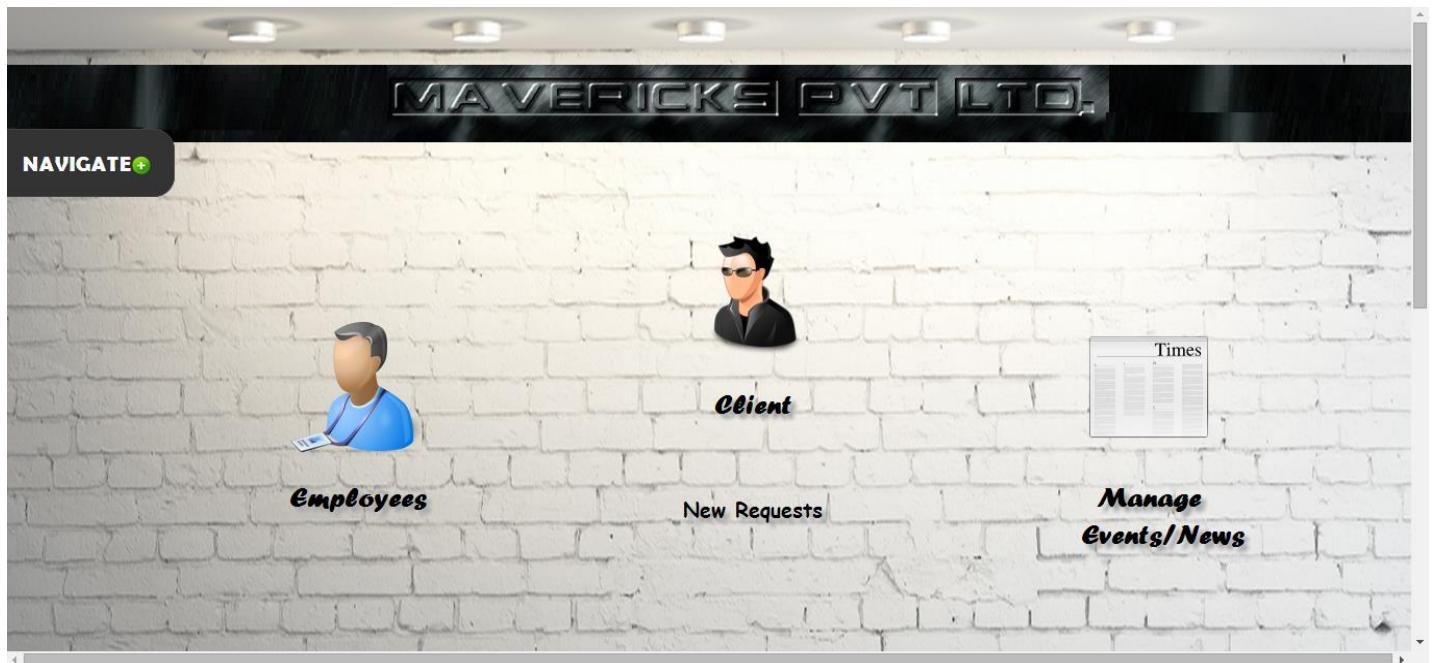
```

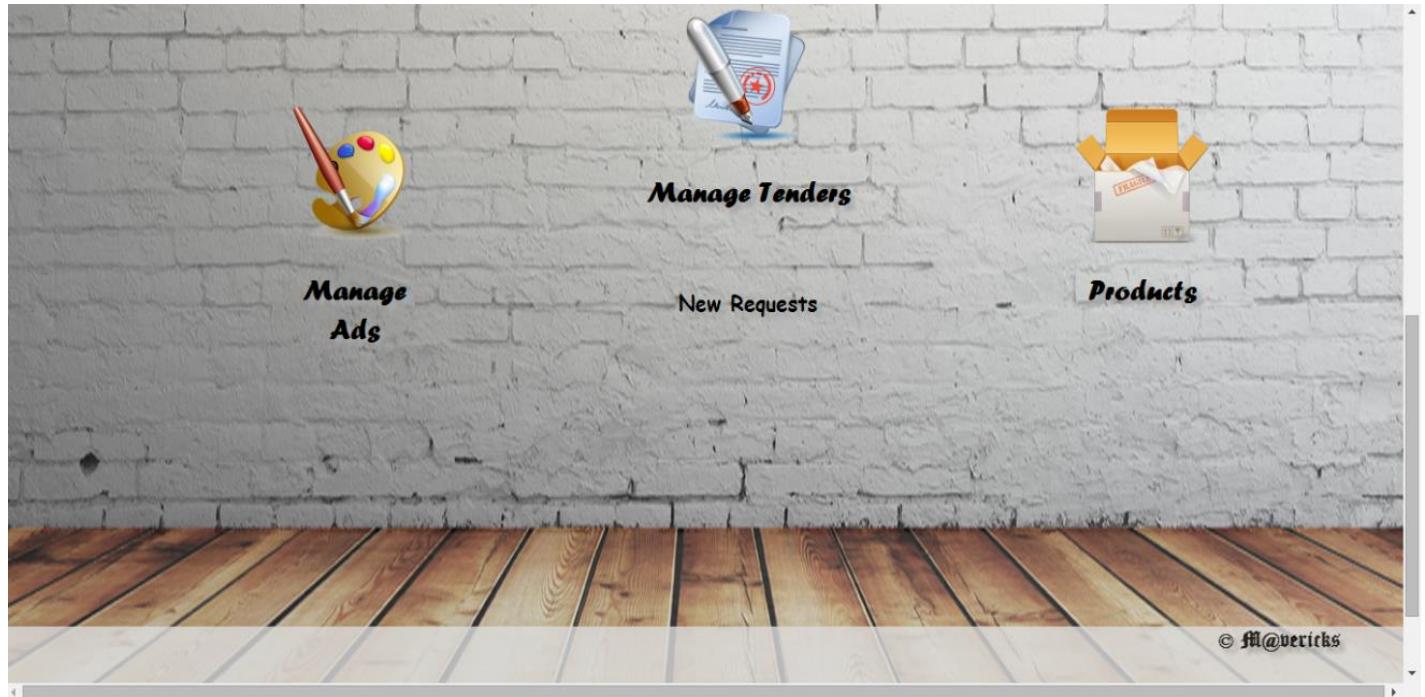
```
</div>
<a class="trigger" style="font-family:Berlin Sans FB Demi;FONT-SIZE:20px" href="#" title="Click here for Navigation!">NAVIGATE</a>

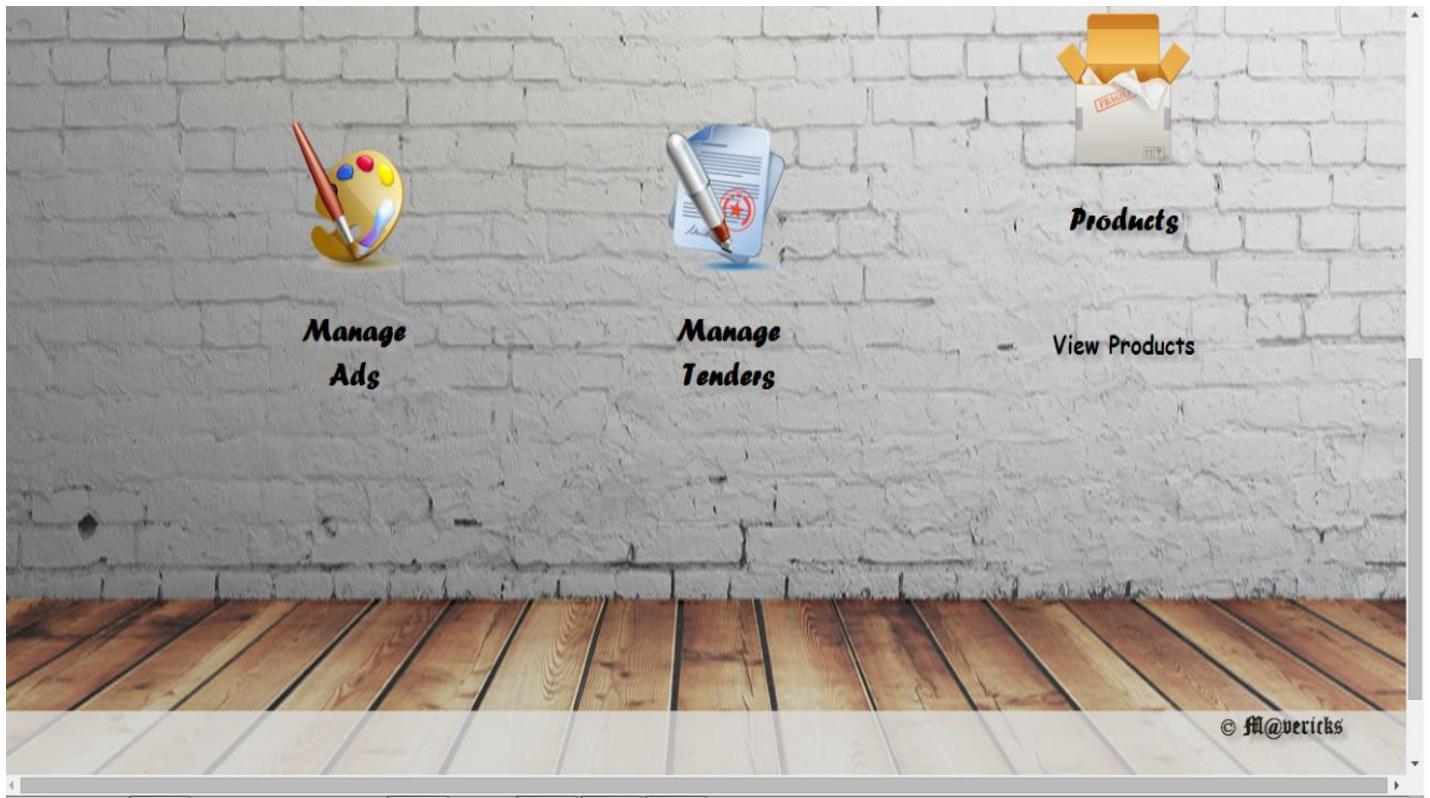
<% if(session_id==null)
{
    response.sendRedirect("home_2.jsp");
} %>
</body>
</html>
```

SCREENSHOTS:









EMPLOYEE DETAILS

Name	ID	D.O.B	Address	Phone Number	Gender	Department	Email ID	Photo	Delete
X	34537	2014-07-10	sdvsdvd	56756	MALE	cs	tsak@gmail.com		Delete
Zara	9998890	2014-07-01	sdvsdvd	656475747	FEMALE	clgclgclg	asdk@ghhh		Delete
employee11	979744999	2014-07-08	dsbfbfdov wwoe	123	FEMALE	sdhsdhsd	sak@gmail.com		Delete
Sai	223	2014-07-01	askdkdkdkdkdk	9492903619	MALE	Admin	tsak@gmail.com		Delete
employee11	979888633	2014-07-08	dsbfbfdov wwoe	123456789	FEMALE	sdhsdhsd	sak@gmail.com		Delete
Ram	8778	2014-07-03	sdkjhshdhdhdhd	7032160333	MALE	df	asdk@ghhh		Delete
X	54766666	2014-07-01	DSGDFGDFG	56876	FEMALE	DGDFG	tsak@gmail.com		Delete
dan	12345	2014-07-01	asdasda	9492903619	on	ashish	ashokcm227@gmail.com		Delete
employee11	979744	2014-07-08	dsbfbfdov wwoe	123	FEMALE	sdhsdhsd	sak@gmail.com		Delete
X	223777788	2014-07-09	VRF	9492903619	FEMALE	df	tsak@gmail.com		Delete

http://localhost:8084/home?action.jsp?Thu Jul 17 2014 20:34:51 GMT+0530 (India Standard Time)

MAVERICKS PVT LTD.**Navigate****Employee Registration****NAME****EMPLOYEE ID****DATE OF BIRTH****ADDRESS****PHONE NUMBER****GENDER** Male Female**DEPARTMENT****DEPARTMENT ID****UPLOAD PHOTO** No file chosen**EMAIL****SET PASSWORD****CONFIRM PASSWORD****Submit****Reset**

http://localhost:8084/ADD%20EMPLOYEE.jsp Thu Jul 17 2014 20:34:51 GMT+0530 (India Standard Time)

EMPLOYEE DETAILS

<u>Employee Name</u>	<u>Employee ID</u>	<u>Employee DOB</u>	<u>Employee Address</u>	<u>Employee Phone Number</u>	<u>Employee gender</u>	<u>Department</u>	<u>Email ID</u>	<u>Delete</u>	<u>Update</u>
X	34537	2014-07-10	sdfsdfsd	56756	MALE	cs	1sai@gmail.com	Delete	Update
Zara	9998890	2014-07-01	sdvsdvsd	656475747	FEMALE	dgdgfdgfdg	asd@ghhh	Delete	Update
employee11	979744999	2014-07-08	dnbfdsdov wioe	121	FEMALE	sdfhsdofl	sai@gmail.com	Delete	Update
Sai	223	2014-07-01	askdfsdjkjbjksjd	9492903619	MALE	Admin	1sai@gmail.com	Delete	Update
employee11	97988633	2014-07-08	dnbfdsdov wioe	123348888	FEMALE	sdfhsdofl	sai@gmail.com	Delete	Update
Ram	8778	2014-07-03	sdkfjhshdlohsdf	7032160333	MALE	df	asd@ghhh	Delete	Update
X	54766666	2014-07-01	DSGDFGDFG	56876	FEMALE	DF6DFG	1sai@gmail.com	Delete	Update
dan	12345	2014-07-01	asdasda	9492903619	on	asddsa	ashokcm227@gmail.com	Delete	Update
employee11	979744	2014-07-08	dnbfdsdov wioe	121	FEMALE	sdfhsdofl	sai@gmail.com	Delete	Update
X	223777788	2014-07-09	vif	9492903619	FEMALE	df	1sai@gmail.com	Delete	Update

http://localhost:8084/Update_employees.jsp Fri Jul 18 2014 20:51:06 GMT+0530 (India Standard Time)

EMPLOYEE DETAILS

<u>Name</u>	<u>ID</u>	<u>D.O.B</u>	<u>Address</u>	<u>Phone Number</u>	<u>Gender</u>	<u>Department</u>	<u>Email ID</u>	<u>Photo</u>	<u>Delete</u>
X	34537	2014-07-10	sdfsdfsd	56756	MALE	cs	1sai@gmail.com		Delete
Zara	9998890	2014-07-01	sdvsdvsd	656475747	FEMALE	dgdgfdgfdg	asd@ghhh		Delete
employee11	979744999	2014-07-08	dnbfdsdov wioe	121	FEMALE	sdfhsdofl	sai@gmail.com		Delete
Sai	223	2014-07-01	askdfsdjkjbjksjd	9492903619	MALE	Admin	1sai@gmail.com		Delete
employee11	97988633	2014-07-08	dnbfdsdov wioe	123348888	FEMALE	sdfhsdofl	sai@gmail.com		Delete
Ram	8778	2014-07-03	sdkfjhshdlohsdf	7032160333	MALE	df	asd@ghhh		Delete
X	54766666	2014-07-01	DSGDFGDFG	56876	FEMALE	DF6DFG	1sai@gmail.com		Delete
dan	12345	2014-07-01	asdasda	9492903619	on	asddsa	ashokcm227@gmail.com		Delete
employee11	979744	2014-07-08	dnbfdsdov wioe	121	FEMALE	sdfhsdofl	sai@gmail.com		Delete
X	223777788	2014-07-09	vif	9492903619	FEMALE	df	1sai@gmail.com		Delete

http://localhost:8084/Viewemployees.jsp Fri Jul 18 2014 20:46:25 GMT+0530 (India Standard Time)

Webpage Screenshot

CLIENT DETAILS

<u>Client Name</u>	<u>Client Designation</u>	<u>Company Name</u>	<u>Client Address</u>	<u>Client Phone No</u>	<u>Date and Time</u>	<u>Client ID</u>	<u>Client Email</u>	<u>Client Image</u>	<u>Client Status</u>
Sai	Manager	sdgdg	SDKJFBSDJSD.HKVJSKB	9492909619	2014-07-01	11001	ashokcm227@gmail.com		Accept

http://localhost:8084/viewclient.jsp Fri Jul 18 2014 21:00:23 GMT+0530 (India Standard Time)

Webpage Screenshot

DEPARTMENT DETAILS

<u>Department Name</u>	<u>Department ID</u>	<u>No of Employees</u>	<u>Department date</u>	<u>Delete</u>
Accounts	4324343	22	2014-07-02	Delete

http://localhost:8084/viewdept.jsp Fri Jul 18 2014 20:58:38 GMT+0530 (India Standard Time)

EVENT DETAILS

<u>Event Name</u>	<u>Event ID</u>	<u>Event Description</u>	<u>Event Date</u>	<u>Delete events</u>
-------------------	-----------------	--------------------------	-------------------	----------------------

shi	2232332	sdghj	2014-07-08	Delete
-----	---------	-------	------------	------------------------

http://localhost:8084/viewevents.jsp Fri Jul 18 2014 21:01:56 GMT+0530 (India Standard Time)

MAVERICKS PVT LTD.

NAVIGATE +

Add Department

DEPARTMENT NAME

DEPARTMENT ID

NUMBER OF EMPLOYEES

DATE

 dd----yyyy

http://localhost:8084/Add%20department.jsp Thu Jul 17 2014 20:36:33 GMT+0530 (India Standard Time)

NEWS DETAILS

<u>News Title</u>	<u>News Content</u>	<u>News Id</u>	<u>Employee Id</u>	<u>News Photo</u>	<u>News date</u>	<u>Delete news</u>
Company retreat	The company is planning a retreat on this Sunday.	32456	12345		2014-02-02	Delete

http://localhost:8084/viewnews.jsp Fri Jul 18 2014 21:07:59 GMT+0530 (India Standard Time)

TENDER DETAILS

<u>Tender Name</u>	<u>Company Name</u>	<u>Phone Number</u>	<u>Email ID</u>	<u>Tender Description</u>	<u>Tender Id</u>	<u>Tender date</u>	<u>Status</u>	<u>Quotation</u>	<u>Generate Status</u>
QWEEE	SDSDF	123456789	SFSDF@SDFSD	SDFSDFSDFSDV	11014	12-2-2014	Accept	QWEEE.dock	Generate status
A	B	9550702460	skf@sdf	ASDSDSDFSD	11012	2014-07-03	Pending	A.dock	Generate status
X	Y	676767676	xvkjnxvc@dvkhxdv	skdjchsdoivnxckv	11013	2014-07-02	Pending	X.dock	Generate status
Q	P	3445678990	SDFSD@SDFSD	SDFSDKJHGFSIDV SD	11011	2014-07-03	Reject	Q.dock	delete
Tender1	sdfsdf	565656566	asdasd@ddfgdf	sdfsdsdkfsdifi	11015	2014-07-04	Pending	Tender1.dock	Generate status

http://localhost:8084/viewtender.jsp Fri Jul 18 2014 21:35:44 GMT+0530 (India Standard Time)

PRODUCT DETAILS

<u>Product Name</u>	<u>Product ID</u>	<u>Product price</u>	<u>Employee ID</u>	<u>Date</u>	<u>Product Image</u>	<u>Delete</u>
-------------------------------------	-----------------------------------	--------------------------------------	------------------------------------	-----------------------------	--------------------------------------	-------------------------------

Kit	3423	2323	12345	2014-07-08		Delete
-----	------	------	-------	------------	---	------------------------

http://localhost:8084/viewproducts.jsp Fri Jul 18 2014 21:37:28 GMT+0530 (India Standard Time)

>

ADS

<u>Advertisement Name</u>	<u>Product Name</u>	<u>Advertisement Id</u>	<u>Employee Id</u>	<u>Last Date</u>	<u>Product Id</u>	<u>Image</u>	<u>Delete Ads</u>
---	-------------------------------------	---	------------------------------------	----------------------------------	-----------------------------------	------------------------------	-----------------------------------

gak	gom	12345	12345	2014-07-03	3423		Delete
-----	-----	-------	-------	------------	------	---	------------------------

http://localhost:8084/viewads.jsp Fri Jul 18 2014 21:33:51 GMT+0530 (India Standard Time)

PRODUCT DETAILS

<u>Product Name</u>	<u>Product ID</u>	<u>Product price</u>	<u>Employee ID</u>	<u>Date</u>	<u>Product Image</u>	<u>Delete</u>
Kit	3423	2323	12345	2014-07-08		Delete

PRODUCT DETAILS

<u>Product Name</u>	<u>Product ID</u>	<u>Product price</u>	<u>Employee ID</u>	<u>Date</u>	<u>Update</u>
Kit	3423	2323	12345	2014-07-08	Update

TENDER DETAILS

<u>Tender Name</u>	<u>Company Name</u>	<u>Phone Number</u>	<u>Email ID</u>	<u>Tender Description</u>	<u>Tender Id</u>	<u>Tender date</u>	<u>Status</u>	<u>Onotiation</u>	<u>Generate Status</u>
QWEEE	SDSDF	123456789	SFSDF@SDFSD	SDFSDFSDFSDF	11014	12-2-2014	Accept	QWEEE.docx	
A	B	9550702460	skf@sdf	ASDSDSDFSD	11012	2014-07-03	Pending	A.docx	Generate status
X	Y	676767676	xvkjnxvc@dvkhxdv	skdjchsdovnxckv	11013	2014-07-02	Pending	X.docx	Generate status
Q	P	3445678990	SDFSD@SDFSD	SDFSDKJHGFSIDV SD	11011	2014-07-03	Reject	Q.docx	delete
Tender1	sdfsdf	565656566	asdasd@ddfgdf	sdfsldskfsdhi	11015	2014-07-04	Pending	Tender1.docx	Generate status

http://localhost:8084/viewtender.jsp Fri Jul 18 2014 21:52:39 GMT+0530 (India Standard Time)

CLIENT DETAILS

<u>Client Name</u>	<u>Client Designation</u>	<u>Company Name</u>	<u>Client Address</u>	<u>Client Phone No</u>	<u>Date and Time</u>	<u>Client ID</u>	<u>Client Email</u>	<u>Client Image</u>	<u>Client Status</u>
Sai	Manager	sdgdg	SDKJFBSDJSJKVJSKB	9492903619	2014-07-01	11001	ashokcm227@gmail.com		Accept

http://localhost:8084/viewclient.jsp Fri Jul 18 2014 21:54:28 GMT+0530 (India Standard Time)

Webpage Screenshot

MAVERICKS PVT LTD.

NAVIGATE +



Add Ads

ADS NAME:

PRODUCT NAME:

ADS ID:

EMPLOYEE ID:

LAST DATE FOR ADS:

UPLOAD IMAGE: No file chosen

Submit **Reset**

http://localhost:8084/ADD%20ADS.jsp Thu Jul 17 2014 20:44:05 GMT+0530 (India Standard Time)

Webpage Screenshot

MAVERICKS PVT LTD.

NAVIGATE +



Add Products

PRODUCT NAME:

PRODUCT ID:

PRICE OF PRODUCT:

EMPLOYEE ID:

DATE:

UPLOAD IMAGE: No file chosen

Submit **Reset**

http://localhost:8084/ADD%20PRODUCTS.jsp Thu Jul 17 2014 20:39:38 GMT+0530 (India Standard Time)

Webpage Screenshot

MAVERICKS PVT LTD.

NAVIGATE +

Add News

OXYGENTIMES



NEWS TITLE:

NEWS CONTENT:

NEWS ITEM ID:

EMPLOYEE ID:

DATE:

UPLOAD Photo: No file chosen

Submit **Reset**

http://localhost:8084/ADD%20NEWS.jsp Thu Jul 17 2014 20:48:35 GMT+0530 (India Standard Time)

Webpage Screenshot

NEWS DETAILS						
<u>News Title</u>	<u>News Content</u>	<u>News ID</u>	<u>Employee Id</u>	<u>News Photo</u>	<u>News date</u>	<u>Delete news</u>
Company retreat	The company is planning a retreat on this Sunday.	32456	12345		2014-02-02	Delete

http://localhost:8084/viewnews.jsp Fri Jul 18 2014 21:07:59 GMT+0530 (India Standard Time)

Webpage Screenshot

ADVERTISEMENTS DETAILS

<u>Advertisement Name</u>	<u>Product Name</u>	<u>Advertisement ID</u>	<u>Employee Id</u>	<u>Advertisement Last date</u>	<u>Delete ads</u>	<u>Updates ads</u>
gak	gom	12345	12345	2014-07-03	Delete	Update

http://localhost:8084/updateads.jsp Fri Jul 18 2014 21:12:45 GMT+0530 (India Standard Time)

Webpage Screenshot

%>

ADS

<u>Advertisement Name</u>	<u>Product Name</u>	<u>Advertisement Id</u>	<u>Employee Id</u>	<u>Last Date</u>	<u>Product Id</u>	<u>Image</u>	<u>Delete Ads</u>
gak	gom	12345	12345	2014-07-03	3423		Delete

http://localhost:8084/viewads.jsp Fri Jul 18 2014 21:11:20 GMT+0530 (India Standard Time)

Webpage Screenshot

MAVERICKS PVT LTD.

NAVIGATE +



Add Events

EVENT NAME:

EMPLOYEE ID:

EVENT DESCRIPTION:

EVENT DATE:

Submit **Reset**

http://localhost:8084/ADD%20EVENTS.jsp Thu Jul 17 2014 20:46:08 GMT+0530 (India Standard Time)

Webpage Screenshot

MAVERICKS PVT LTD.

Daily News

Lorum ipsum lorem ipsum
Lorum ipsum is simply dummy text of the printing and typesetting industry. Lorum ipsum has been the industry's standard dummy text ever since the 1980s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorum ipsum, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorum ipsum.



Update News

CURRENT NEWS TITLE:

NEW NEWS TITLE:

EDIT CONTENT:

EDIT IMAGE:

DATE AND TIME:

Submit **Cancel**

http://localhost:8084/UPDATE%20NEWS.jsp Thu Jul 17 2014 20:45:39 GMT+0530 (India Standard Time)

EVENT DETAILS

<u>Event Name</u>	<u>Event ID</u>	<u>Event Description</u>	<u>Event Date</u>	<u>Delete events</u>
shi	2232332	sdgħlj	2014-07-08	Delete

http://localhost:8084/viewevents.jsp Fri Jul 18 2014 21:18:36 GMT+0630 (India Standard Time)

MAVERICKS PVT LTD.

Update News

Daily News

Lorem ipsum lorem ipsum

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s,



When an unknown printer took a galley of type and scrambled it to make

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the

industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make

CURRENT NEWS TITLE:

NEW NEWS TITLE:

EDIT CONTENT:

EDIT IMAGE:

DATE AND TIME:

Submit

Cancel

http://localhost:8084/UPDATE%20NEWS.jsp Thu Jul 17 2014 20:45:39 GMT+0630 (India Standard Time)

EVENTS DETAILS

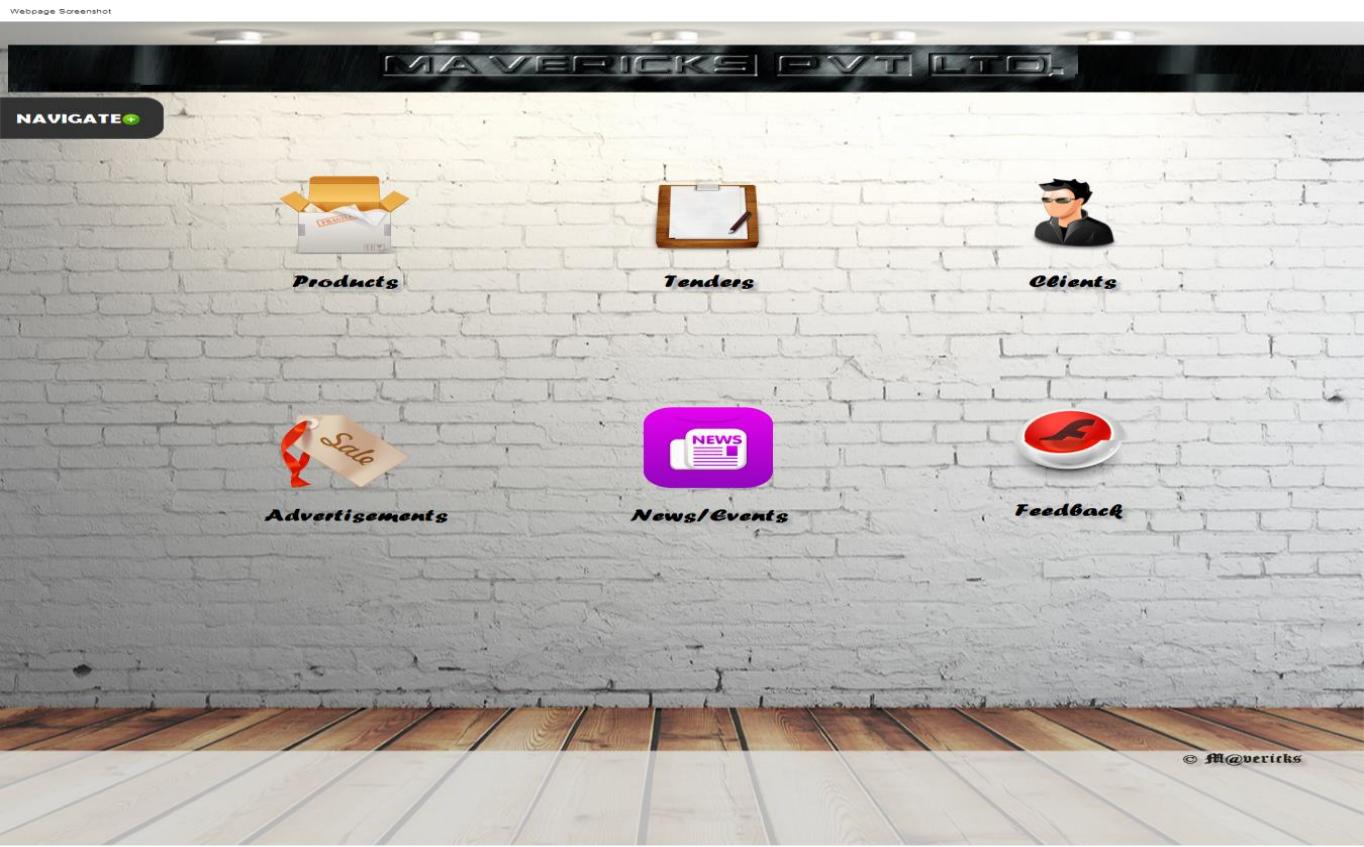
Event Name	Event ID	Event description	Event date	Update Event
------------	----------	-------------------	------------	--------------

shi	2232332	sdgħlji	2014-07-08	Update
-----	---------	---------	------------	------------------------

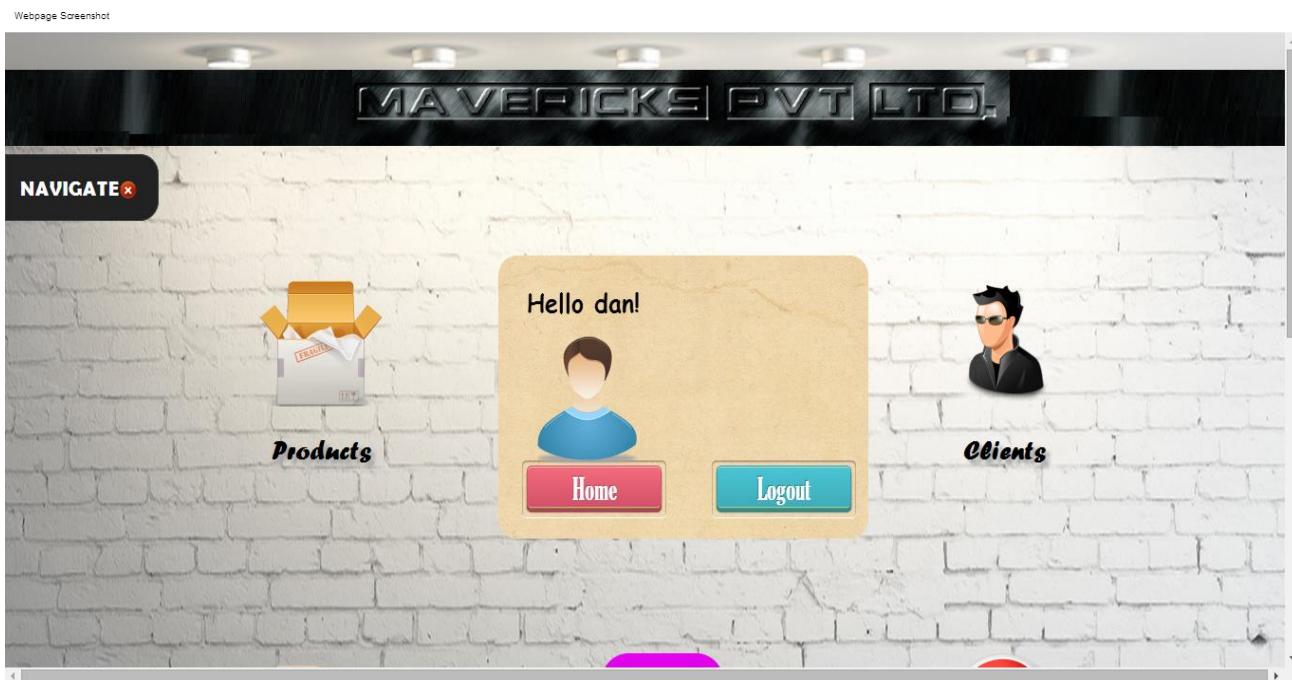
FEEDBACK DETAILS

Client Name	Client ID	Remark
-------------	-----------	--------

Shyam	11001	The service has been really good.
-------	-------	-----------------------------------



http://localhost:8084/home_action.jsp Thu Jul 17 2014 20:19:59 GMT+0530 (India Standard Time)



http://localhost:8084/home_action.jsp Thu Jul 17 2014 20:25:30 GMT+0530 (India Standard Time)

MAVERICKS PVT LTD.

NAVIGATE



Products

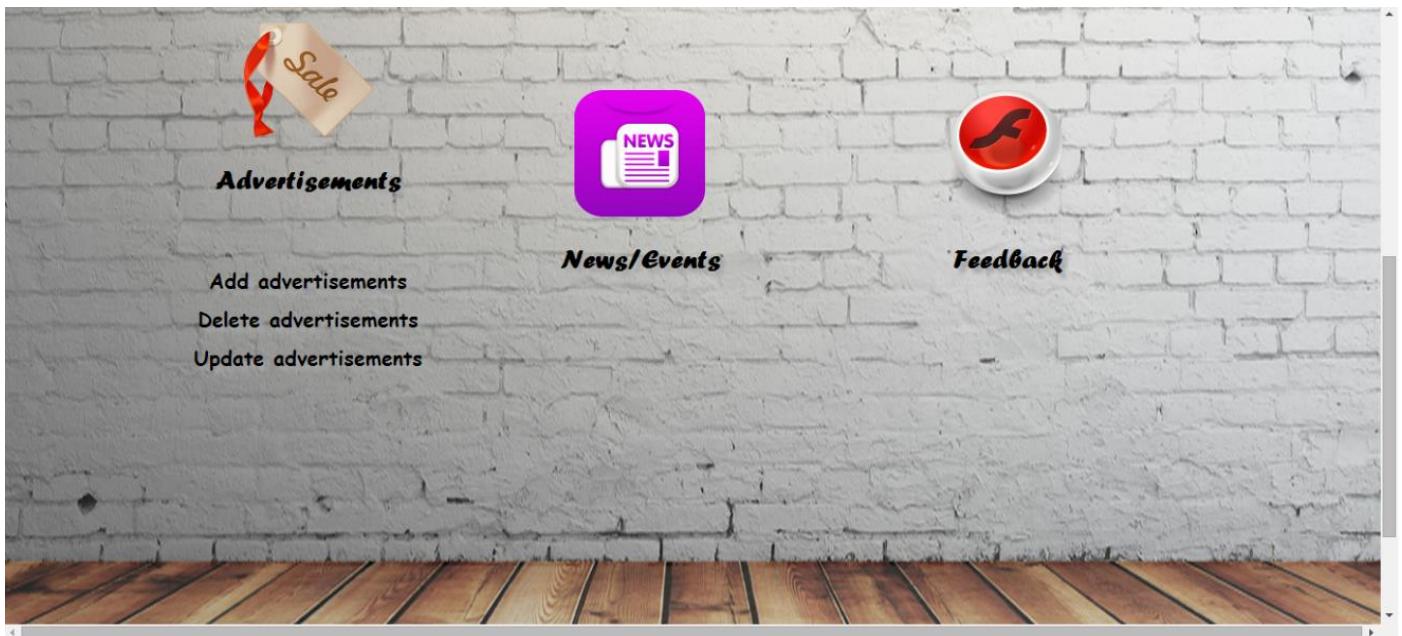
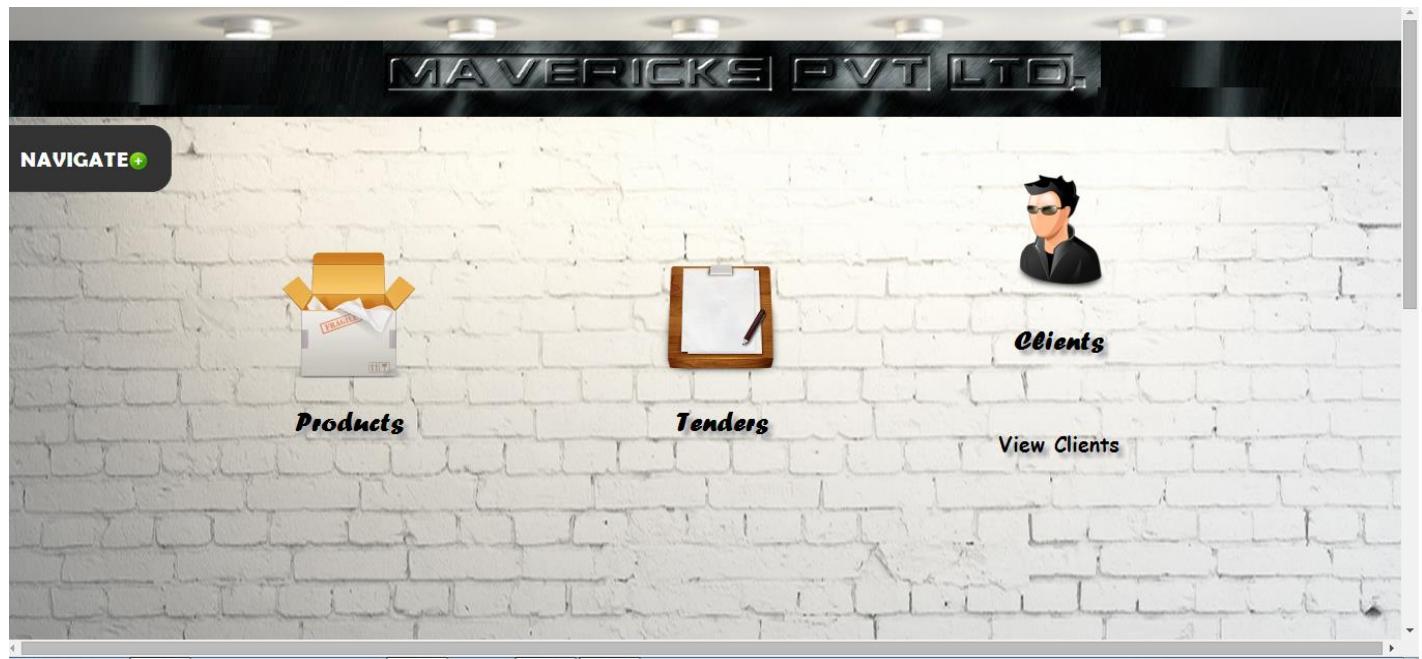


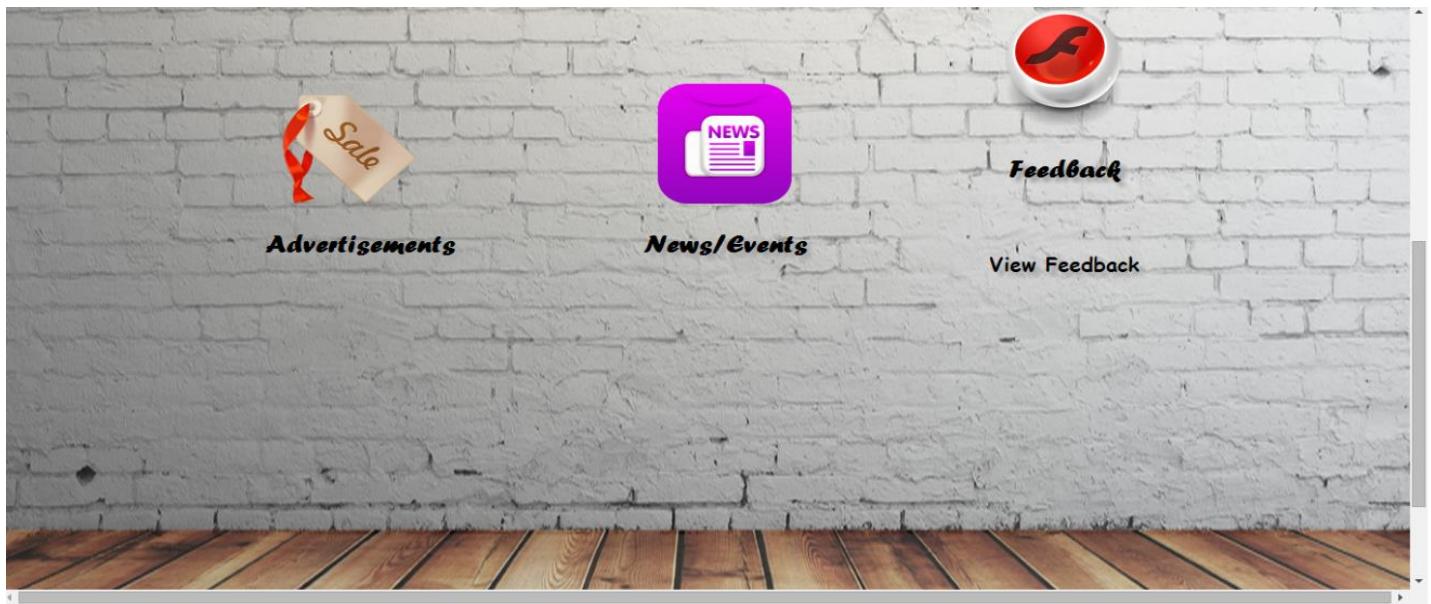
Tenders

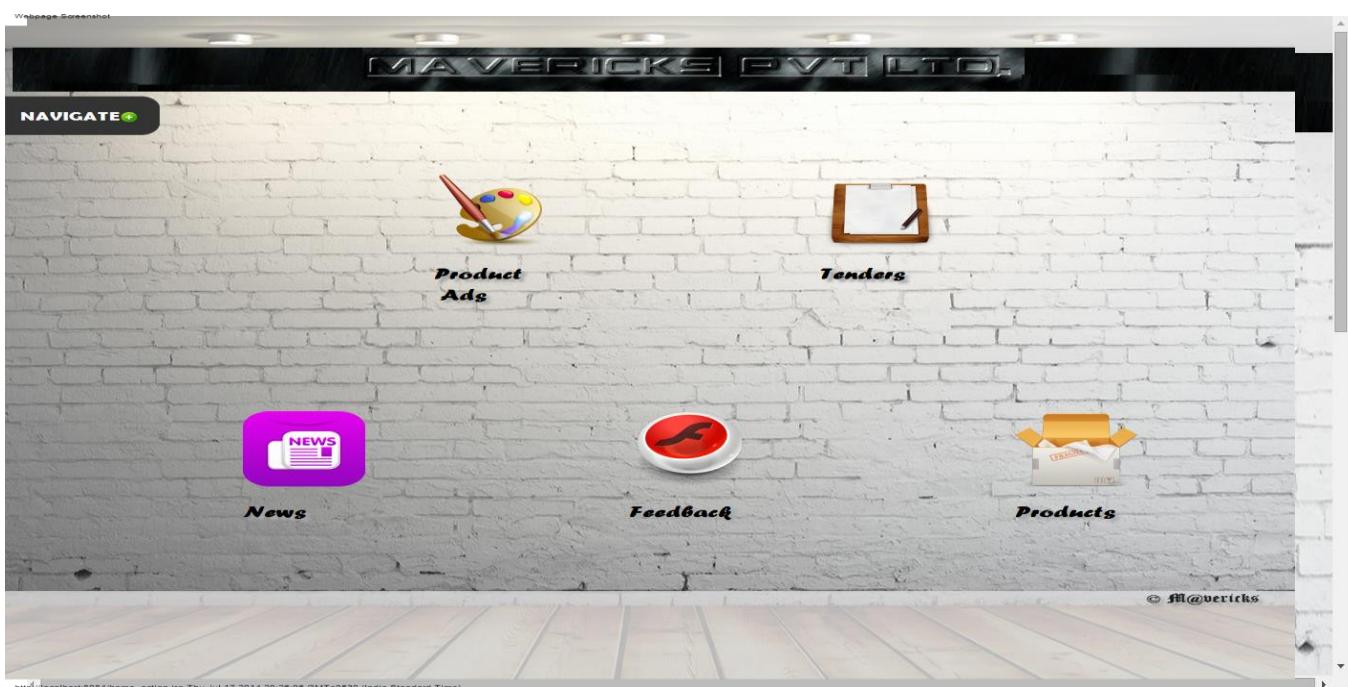
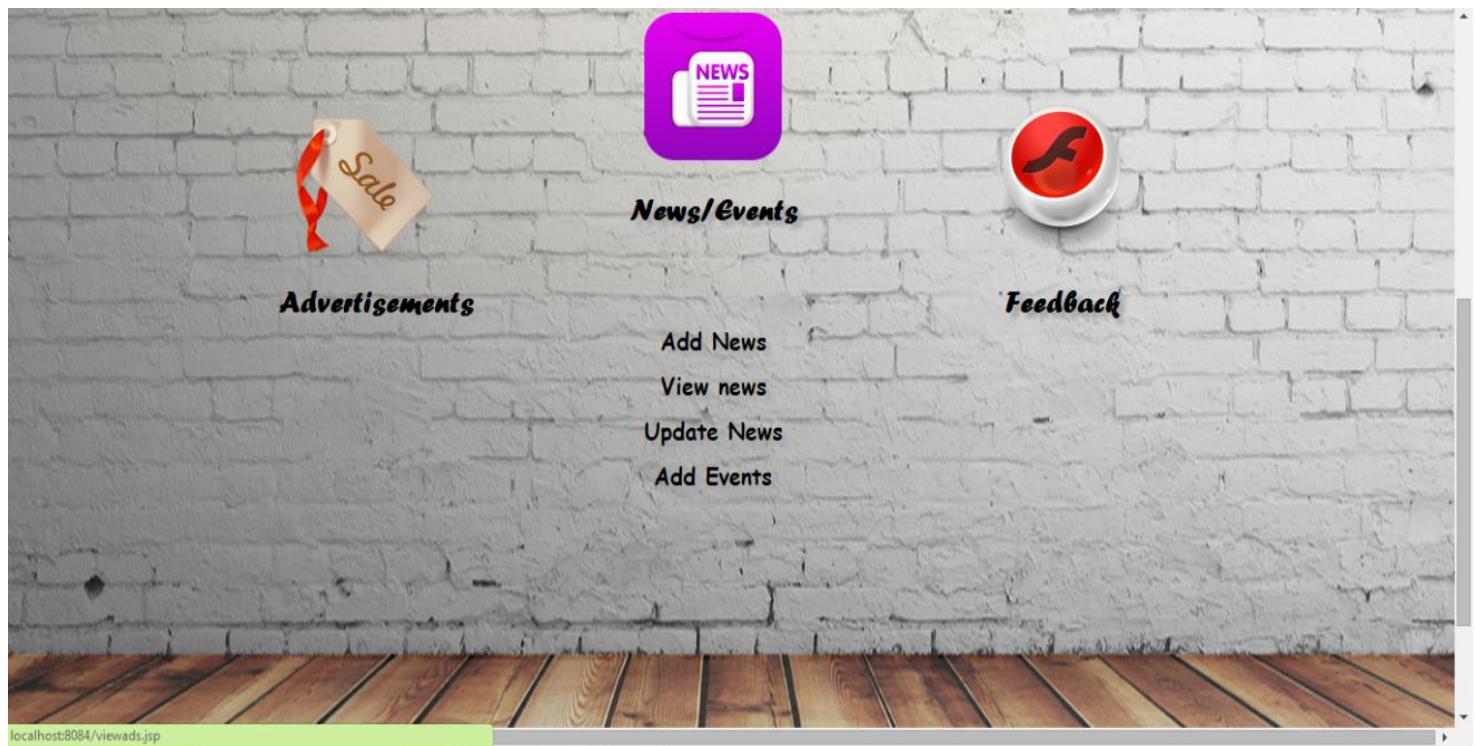
View



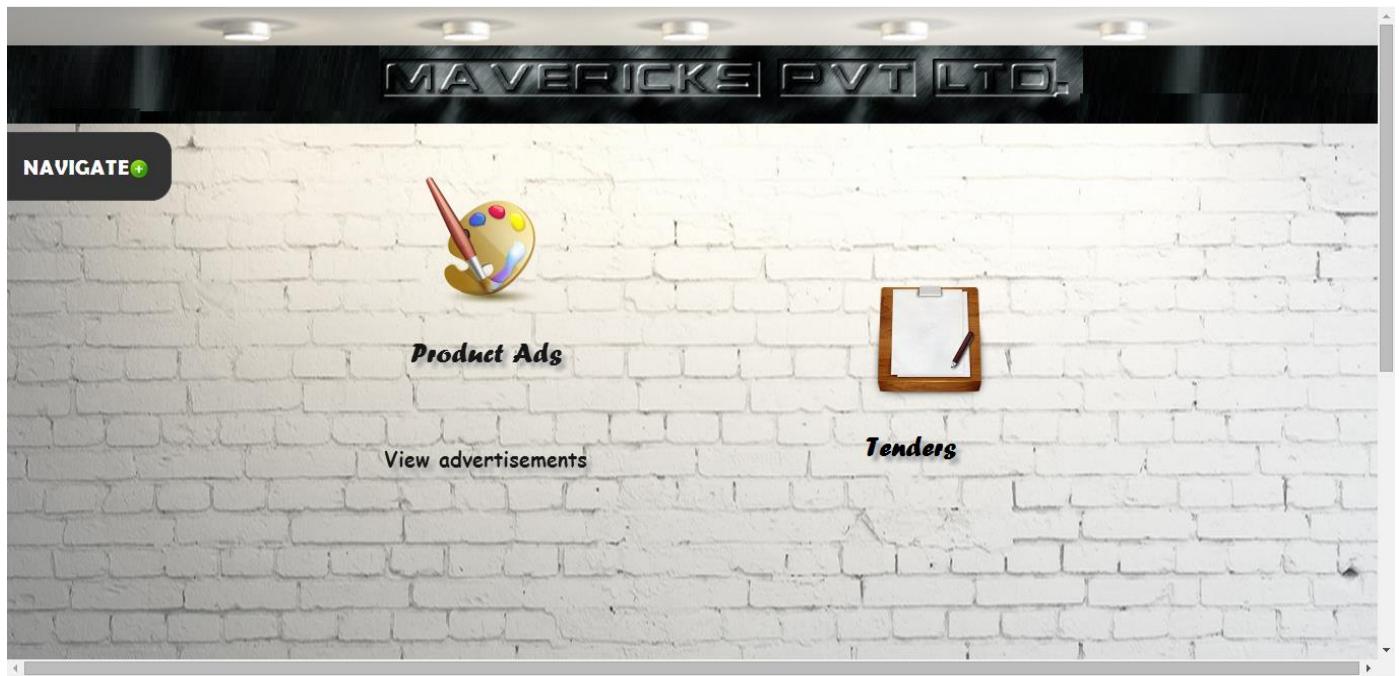
Clients

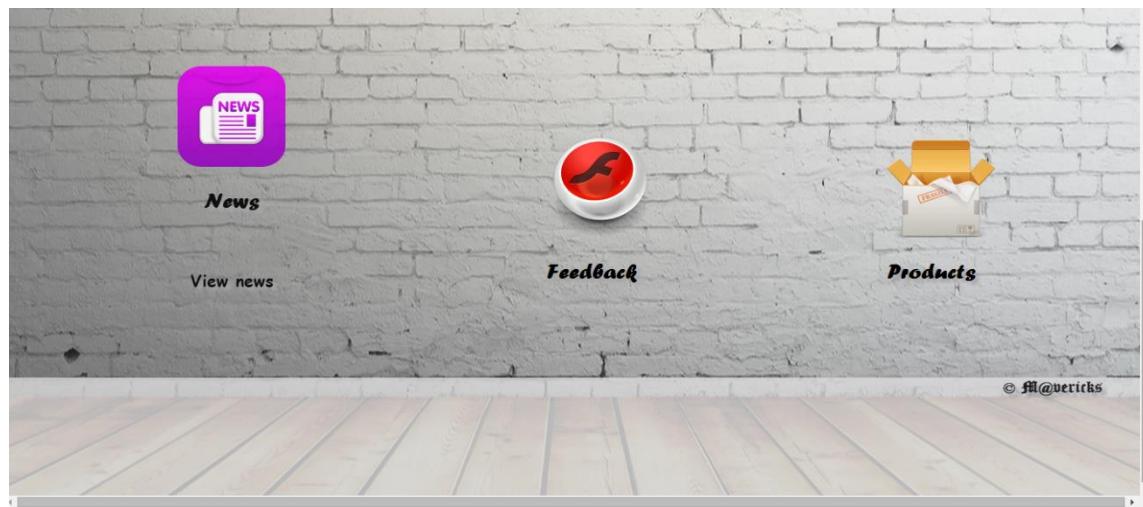


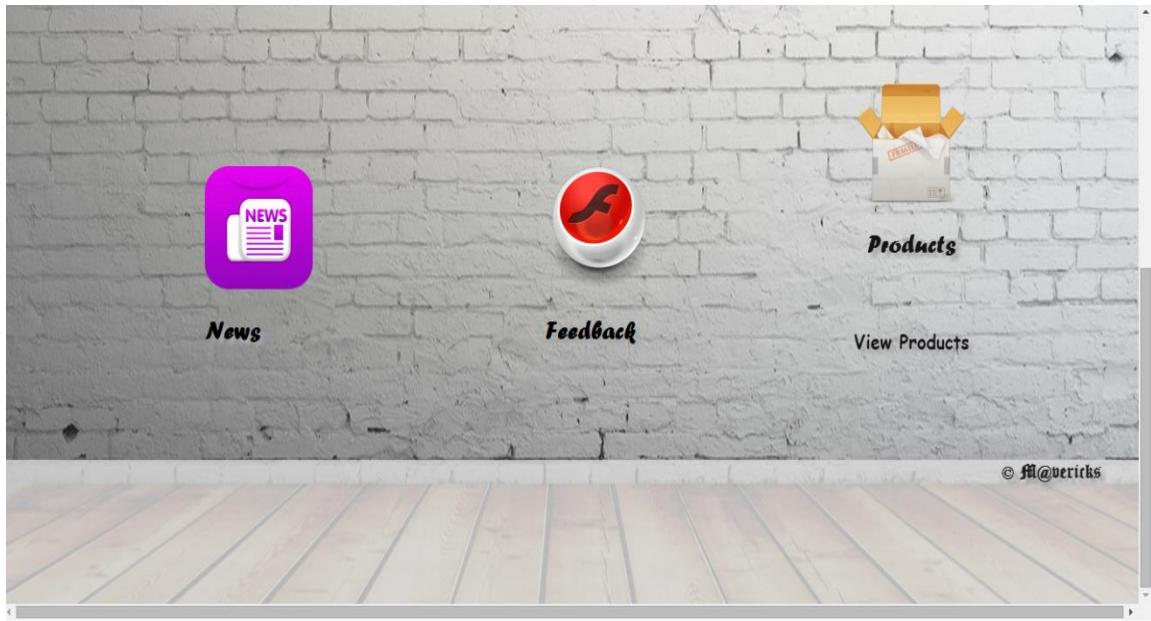




Webpage Screenshot







ADVERTISEMENTS DETAILS

<u>Advertisement Name</u>	<u>Product Name</u>	<u>Advertisement ID</u>	<u>Employee Id</u>	<u>Advertisement Last date</u>
---------------------------	---------------------	-------------------------	--------------------	--------------------------------

gak	gom	12345	12345	2014-07-03
-----	-----	-------	-------	------------

http://localhost:8084/viewads_for_client.jsp Fri Jul 18 2014 22:50:00 GMT+0530 (India Standard Time)

MAVERICKS PVT LTD.



Feedback

CLIENT NAME:

CLIENT ID:

 11001

REMARK:

http://localhost:8084/FEEDBACK.jsp Fri Jul 18 2014 23:10:00 GMT+0530 (India Standard Time)



MAVERICKS PVT LTD.

Add Tenders

TENDER NAME: COMPANY NAME: COMPANY PHONE NUMBER: COMPANY EMAIL: TENDER DESCRIPTION: TENDER ID: DATE: CLIENT ID: PRODUCTS: UPLOAD QUOTATION:	<input type="text" value="Enter Tender name"/> <input type="text" value="Enter Company Name"/> <input type="text" value="Enter phone number"/> <input type="text" value="Enter email address"/> <input type="text" value="Enter Tender Description"/> <input type="text" value="11016"/> <input type="text" value="dd----yyyy"/> <input type="text" value="11001"/> <input type="text" value="Kit"/> <input type="file" value="Choose File"/> No file chosen <div style="text-align: right; margin-top: -10px;"> <input type="button" value="Submit"/> <input type="button" value="Reset"/> </div>
--	--

http://localhost:8084/Add%20Tenders.jsp Thu Jul 17 2014 20:52:00 GMT+0530 (India Standard Time)

NEWS DETAILS					
News Title	News Content	News ID	Employee Id	News date	
Company retreat	The company is planning a retreat on this Sunday.	32456	12345	2014-02-02	

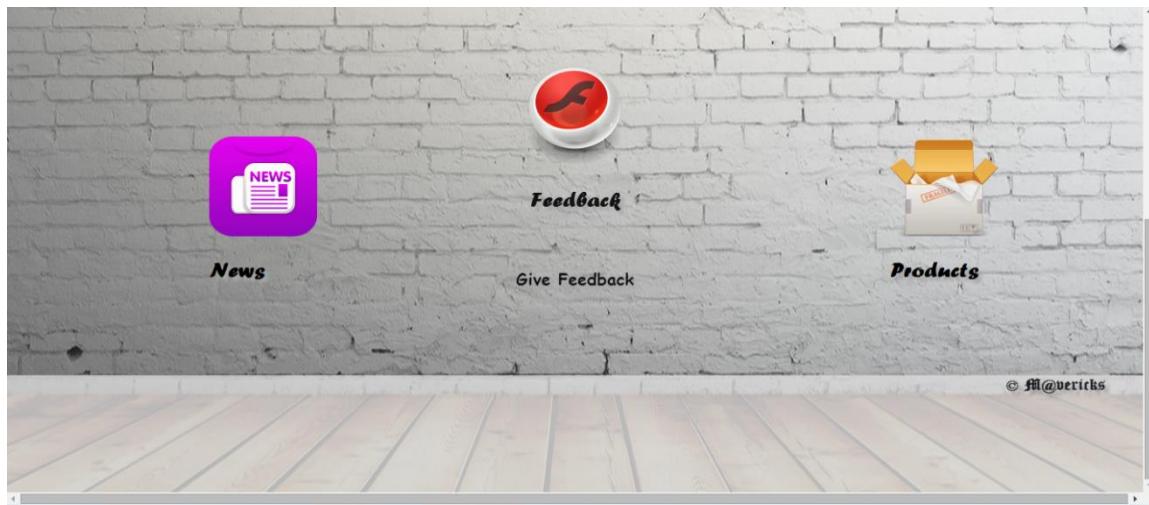
http://localhost:8084/viewnews_for_client.jsp Fri Jul 18 2014 23:18:04 GMT+0530 (India Standard Time)

PRODUCT DETAILS

Product Name	Product ID	Product price	Date
--------------	------------	---------------	------

Kit	3423	2323	2014-07-08
-----	------	------	------------

http://localhost:8084/viewproducts_client.jsp Fri Jul 18 2014 23:17:50 GMT+0530 (India Standard Time)



CHAPTER 13

CONCLUSION

This project is able to successfully incorporate all the requirements specified for creating a Website. Proper care has been taken during database design to maintain data integrity and to avoid data redundancy. Design procedures are included in the project to have the Document management better understand the system. The project is designed and coded in such a way that any further modifications that are needed in the future can be easily implemented without affecting the functionality of the system. The documentation provided in the project report helps the application developers understand the internal architecture of the system and thus assist them in enhancing the system

Web Publishing System is a software application which is user friendly and can be accessed by admin, employees and clients of an organization.

BIBLIOGRAPHY

www.java.com

www.oracle10g.com

www.w3schools.com

Java: The Complete Reference, Seventh Edition by Herbert Schildt

Java: How to Program, Paul Dietel, Harvey Deitel

<https://netbeans.org/>

<http://tomcat.apache.org/>

<http://www.oracle.com/technetwork/java/javase/documentation/8u-relnotes-2225394.html>