

A
RESTAURANT

2019-
2020

CROSS-
PLATFORM
PROJECT

ACKNOWLEDGEMENT

I undertook this Project work, as the part of my XII-Informatics Practices course. I have tried to apply my best of knowledge and experience, gained during the study and class work experience. However, developing software system is generally a quite complex and time-consuming process. It requires a systematic study, insight vision and professional approach during the design and development. Moreover, the developer always feels the need, the help and good wishes of the people near you, who have considerable experience and idea.

I would like to extend my sincere thanks and gratitude to my teacher **Mr. Kirti Kamal Bhardawaj**, for giving valuable time and moral support to develop this software.

I would like to take the opportunity to extend my sincere thanks and gratitude to my parents for being a source of inspiration and providing time and freedom to develop this software project.

Abstract

The Project “**Restaurant Management System** Project” targeted for the Businessmen and for faculty members integrates the transaction management and inventory management of the shop for better control and timely response.

The Shop is provided with better control over the transactions like entering the customer details, maintaining the item details, and order details This project’s main motto is to reduce the effort of the restaurant administrator and provide better service to customers and Faculty.

Project Introduction

This software project is developed to automate the functionalities of a Restaurant. The purpose of the software project is to develop a Management Information System (MIS) to automate the record keeping of customer entry, order entry and item entry with a view to enhance the decision making of the functionaries.

A MIS mainly consists of a computerized database, a collection of interrelated tables for a particular subject or purpose, capable to produce different reports relevant to the user. An application program is tied with the database for easy access and interface to the database. Using Application program or front-end, we can store, retrieve and manage all information in proper way. This software, being simple in design and working, does not require much training to users, and can be used as a powerful tool for automating a Restaurant System.

During coding and design of the software Project, Java NetBeans IDE, a powerful front-end tool is used for getting Graphical User Interface (GUI) based integrated platform and coding simplicity. As a back-end a powerful, open source RDBMS, MySQL is used as per requirement of the CBSE curriculum of Informatics Practices Course.

Customer and staff automation system project is database management software with front end data updating using java HTML pages.

Proposed System

The proposed system works on the automation process where we use software tools to develop an app using Java and MySQL database.

In this system process of workflow from the application process to deliver process will be online. This process will save time and reduce manual work.

Advantages of the proposed system

1. Data management is easy and effective.
2. Data management will be easy. Admin or user can view data of any records in a short time by just entering in the search field.
3. Time is taken for report generation and record retrieval is very short.
4. Data security is provided for total records.
5. Any department inside an organization can easily access old and new data in a short time.

2. Objective & Scope of the Project

The objective of the software project is to develop a computerized MIS to automate the functions of a Restaurant This software project is also aimed to enhance the current record keeping system, which will help managers to retrieve the up-to-date information at the right time in the right shape.

The proposed software system is expected to do the following functionality-

- To provide a user friendly Graphical User Interface (GUI) based integrated and centralized environment for MIS activities.
- The proposed system should maintain all the records and transactions, and should generate the required reports and information when required.
- To provide graphical and user-friendly interface to interact with a centralized database based on client-server architecture.
- To identify the critical operation procedure and possibilities of simplification using modern IT tools and practices.

In its current scope, the software enables user to retrieve and update the information from centralized database designed with MySQL. This software does not require much training time of the users due to limited functionality and simplicity.

During the development of Restaurant Information System project, Java NetBeans IDE, a powerful, open source event-driven form-based development environment is used for modular design and future expandability of the system.

Despite the best efforts of the developer, the following limitations and functional boundaries are visible, which limits the scope of this application software.

1. This software can store records and produce reports in pre-designed format in soft copy. There is no facility yet to produce customized reports. Only specified reports are covered.
2. There is no provision to calculate fine or penalty etc. for defaulter members; however, it can be developed easily with the help of adding modules.
3. Some application areas like deleting the record.

So far as future scope of the project is concerned, firstly it is open to any modular expansion i.e. other modules or functions can be designed and embedded to handle the user needs in the future. Any part of the software and reports can be modified independently without much effort.

3. Theoretical Background

3.1 What is Database?

Introduction and Concepts:

A database is a collection of information related to a particular subject or purpose, such as tracking customer orders or maintaining a music collection. Using any RDBMS application software like MS SQL Server, MySQL, Oracle, Sybase etc., you can manage all your information from a single database file. Within the file, divide your data into separate storage containers called tables.

A table is a collection of data about a specific topic, such as products or suppliers. Using a separate table for each topic means you can store that data only once, which makes your database more efficient and reduces data-entry errors. Table organizes data into columns (called fields) and rows (called records).

A Primary key is one or more fields whose value or values uniquely identify each record in a table. In a relationship, a primary key is used to refer to specific record in one table from another table. A primary key is called foreign key when it is referred to from another table.

To find and retrieve just the data that meets conditions you specify, including data from multiple tables, create a query. A query can also update or delete multiple records at the same time, and perform built-in or custom calculations on your data.

RDBMS Application Program

A computer database works as an electronic filing system, which has a large number of ways of cross-referencing, and this allows the user many different ways in which to re-organize and retrieve data. A database can handle business inventory, accounting and filing and use the information in its files to prepare summaries, estimates and other reports. The management of data in a database system is done by means of a general-purpose software package called a Database Management System (DBMS). Some commercially available DBMS are MS SQL Server, MS ACCESS, INGRES, ORACLE, and Sybase. A database management system, therefore, is a combination of hardware and software that can be used to set up and monitor a database, and can manage the updating and retrieval of database that has been stored in it. Most of the database management systems have the following capabilities:

- Creating of a table, addition, deletion, modification of records.
- Retrieving data collectively or selectively.
- The data stored can be sorted or indexed at the user's discretion and direction.
- Various reports can be produced from the system. These may be either standardized report or that may be specifically generated according to specific user definition.
- Mathematical functions can be performed and the data stored in the database can be manipulated with these functions to perform the desired calculations.
- To maintain data integrity and database use.

The DBMS interprets and processes users' requests to retrieve information from a database. In most cases, a query request will have to penetrate several layers of software in the DBMS and operating system before the physical database can be accessed. The DBMS responds to a query by invoking the appropriate subprograms, each of which performs its special function to interpret the query, or to locate the desired data in the database and present it in the desired order.

3.2 What is MySQL?

The management of data in a database system is done by means of a general-purpose software package called a Database Management System (DBMS). Some commercially available RDBMS are MS SQL Server, MS ACCESS, INGRES, ORACLE, and Sybase.

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. MySQL is named after co-founder Monty Widenius's daughter, My. The name of the MySQL Dolphin (our logo) is "Sakila,".

- MySQL is a database management system.
A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large

amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- MySQL is based on SQL.

A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of “MySQL” stands for “Structured Query Language.” SQL is the most common standardized language used to access databases and is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, “SQL-92” refers to the standard released in 1992, “SQL:1999” refers to the standard released in 1999, and “SQL:2003” refers to the current version of the standard.

- MySQL software is Open Source.

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License),

- The MySQL Database Server is very fast, reliable, and easy to use.

If that is what you are looking for, you should give it a try. MySQL Server also has a practical set of features developed in close cooperation with our users. You can find a performance comparison of MySQL Server with other database managers on our benchmark page. MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

- MySQL Server works in client/server or embedded systems.

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backend, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

The Main Features of MySQL

- Written in C and C++.
- Works on many different platforms.
- Uses multi-layered server design with independent modules.
- Provides transactional and non-transactional storage engines.
- Uses a very fast thread-based memory allocation system.
- Executes very fast joins using an optimized nested-loop join.
- Implements SQL functions using a highly optimized class library that should be as fast as possible. Usually there is no memory allocation at all after query initialization.
- Provides the server as a separate program for use in a client/server networked environment, and as a library that can be embedded (linked) into standalone applications. Such applications can be used in isolation or in environments where no network is available.
- Password security by encryption of all password traffic when you connect to a server.
- Support for large databases. We use MySQL Server with databases that contain 50 million records. We also know of users who use MySQL Server with 200,000 tables and about 5,000,000,000 rows.
- MySQL client programs can be written in many languages. A client library written in C is available for clients written in C or C++, or for any language that provides C bindings.
- APIs for C, C++, Eiffel, Java, Perl, PHP, Python, Ruby are available, enabling MySQL clients to be written in many languages.
- The Connector/ODBC (MyODBC) interface provides MySQL support for client programs that use ODBC (Open Database Connectivity) connections.
- The Connector/J interface provides MySQL support for Java client programs that use JDBC connections. Clients can be run on Windows or Unix. Connector/J source is available.

3.3 What is NetBeans IDE?

NetBeans started as a student project (originally called Xelfi) in the Czech Republic in 1996. The goal was to write a Delphi-like Java IDE in Java. Xelfi was the first Java IDE (Integrated Development Environment) written in Java, with its first pre-releases in 1997. Xelfi was a fun project to work on, especially since Java IDE space was uncharted territory at that time. The project attracted enough interest that these students, once they graduated, decided that they could market it as a commercial product. Soliciting resources from friends and relatives for a web space, they formed a company around it.

Soon after, they were contacted by [Roman Stanek](#), an entrepreneur who had already been involved in several startups in the Czech Republic. He was looking for a good idea to invest in, and discovered Xelfi. He met with the founders; they hit it off, and a business was born.

In the spring of 1999, [NetBeans DeveloperX2](#) was released, supporting Swing. The performance improvements that came in JDK 1.3, released in the fall of 1999, made NetBeans a viable choice for development tools. By the summer of 1999, the team was hard at work re-architecting DeveloperX2 into the more modular NetBeans that forms the basis of the software today.

Something else was afoot in the summer of 1999: [Sun Microsystems](#) wanted better Java development tools, and had become interested in NetBeans. It was a dream come true for the NetBeans team: NetBeans would become the flagship tool set of the maker of Java itself! By the Fall, with the next generation of NetBeans Developer in beta, a deal was struck. Sun Microsystems had also acquired another tools company, During the acquisition, the young developers who had been involved in open-source projects for most of their programming careers, mentioned the idea of open-sourcing NetBeans. Fast forward to less than six months later, the decision was made that NetBeans would be open sourced. While Sun had contributed considerable amounts

of code to open source projects over the years, this was Sun's first *sponsored* open source project, one in which Sun would be paying for the site and handling the infrastructure.

Features of NetBeans

A free, open-source Integrated Development Environment for software developers. You get all the tools you need to create professional desktop, enterprise, web, and mobile applications with the Java platform, as well as C/C++, PHP, JavaScript, Groovy, and Ruby.

NetBeans IDE 6.9 introduces the JavaFX Composer, support for JavaFX SDK 1.3, OSGi interoperability, support for the PHP Zend framework and Ruby on Rails 3.0, and more.

4. Problem Definition & Analysis

The hardest part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements. Defining and applying good, complete requirements are hard to work, and success in this endeavor has eluded many of us. Yet, we continue to make progress.

Problem definition describes the *What* of a system, not *How*. The quality of a software product is only as good as the process that creates it. Problem definition is one of the most crucial steps in this creation process. Without defining a problem, developers do not know what to build, customers do not know what to expect, and there is no way to validate that the built system satisfies the requirement.

Problem definition and Analysis is the activity that encompasses learning about the problem to be solved, understanding the needs of customers and users, trying to find out who the user really is, and understanding all the constraints on the solution. It includes all activities related to the following:

- Identification and documentation of customer's or user's needs.
- Creation of a document that describes the external behavior and the association constraints that will satisfy those needs.
- Analysis and validation of the requirements documents to ensure consistency, completeness, and feasibility
- Evolution of needs.

After the analysis of the functioning of a Public Library system, the proposed System is expected to do the following: -

- To provide a user friendly Graphical User Interface (GUI) based integrated and centralized environment for computerized Public Library System.
 - The proposed system should maintain all the records and transactions, and should generate the required reports and information when required.
 - To provide efficient and secure Information storage, flow and retrieval system, ensuring the integrity and validity of records.
 - To provide graphical and user-friendly interface to interact with a centralized database based on client-server architecture.
 - To identify the critical operation procedure and possibilities of simplification using modern IT tools and practices.
-

5. System Implementation

5.1 The Hardware used:

While developing the system, the used hardware are:

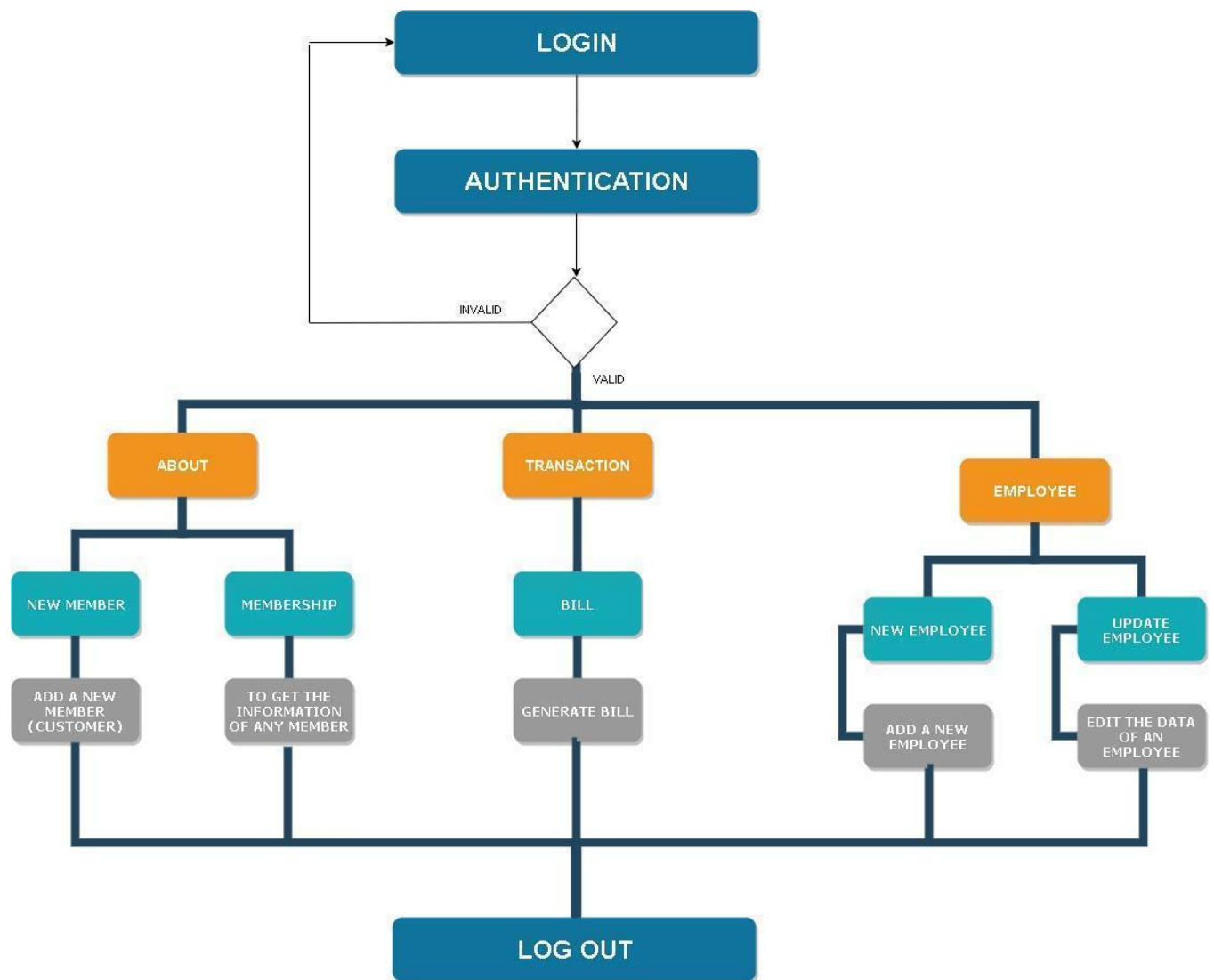
- PC with Pentium IV processor or sometimes, PC with Celeron (1.7 GHz) processor having 256 MB RAM, SVGA and other required devices.

5.2 The Softwares used:

- Microsoft Windows® 7 as Operating System.
- Java NetBeans 6.9 as Front-end Development environment.
- MySQL as Back-end Server with Database for Testing.
- MS-Word 2000 for documentation.

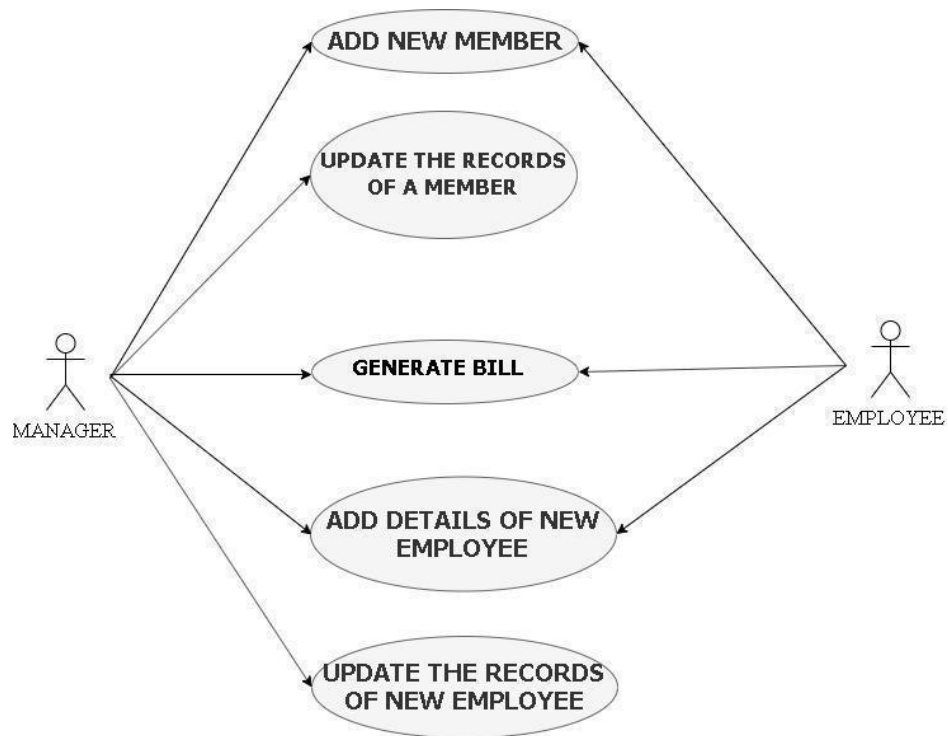
Activity Diagram for the Restaurant Management System

Activity diagram describes the flow of activity through a series of actions. An activity is an action or an operation which is performed in the system.



Use Case Diagram of the Restaurant Management System

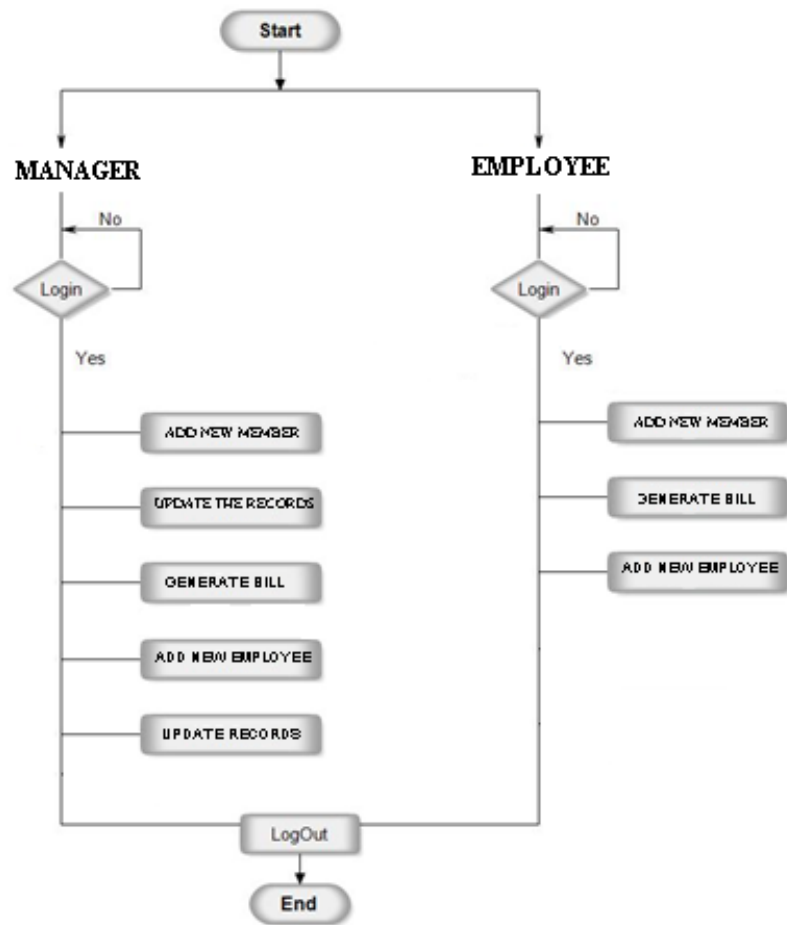
Use Case Diagram or Behavior Diagram is used to describe the actions of all the users in a system. All the users described in a Use Case are actors. Functionality is described as an action of the system.



In this RESTAURANT system project there are two users - Manager and Employee. Both of them can perform various activities after login in to the restaurant management system.

Flowchart for Restaurant Management System

System flow diagram is a visual representation of all processes in sequential order. Flowchart diagram is a graphical representation of the relation between all the major parts or steps of the system. Flow chart diagram cannot include minor parts of the system.



Screenshots of the Restaurant Management System



FOODIES

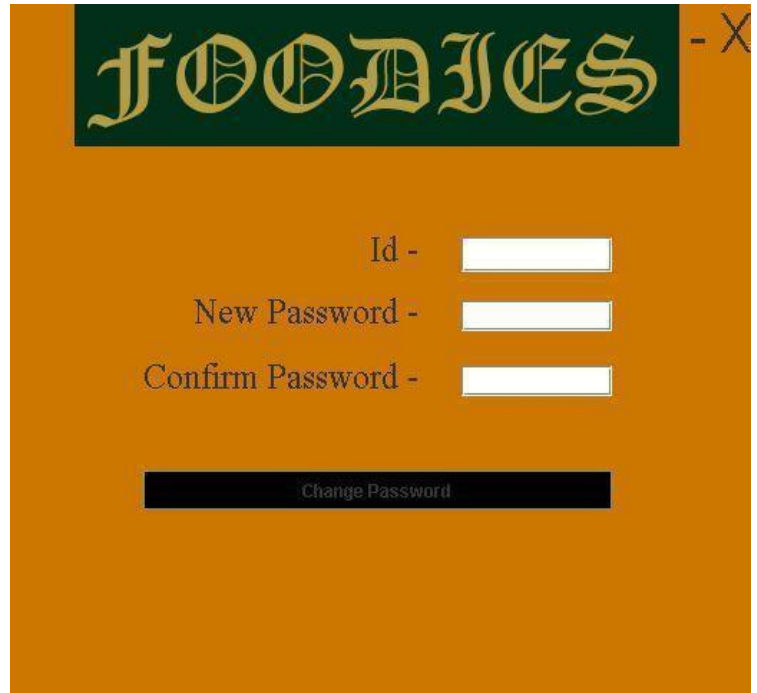
BEVERAGES Coffee req... 80

yummy!!

ITEM PURCHASED

TOTAL COST 0

CLEAR



FOODIES

Id -

New Password -

Confirm Password -

Change Password



FOODIES

USERNAME

PASSWORD

RESET PASSWORD

LOGIN SIGN UP



About Transaction Employee Login

FOODIES

FOODIES - X

username

Name

Phone no.

Address

Password

Re-Enter Password

FOODIES - X

NAME

WORK TYPE

PHONE NO.

QUALIFICATION

ADDRESS

FOODIES - X

NAME	SUBSCRIPTION N...	MOBILE NO.
1	y	12346
2	Yash Mishra	1234567891
3	Pranjal	0000000000
4	Yash Mishra	1234567890
5	1	1234567890

ID-

NAME-

MOBILE NO.-

FOODIES X

Name

Mobile no.

FOODIES

- X

ITEM NO.	CATEGO...	ITEM	PRICE
1	INDIAN D...	Veg. Kor...	250
2	INDIAN D...	Green Sa...	100
3	INDIAN D...	Navratan ...	300
4	INDIAN D...	Paneer M...	280
5	INDIAN D...	Paneer D...	260
6	INDIAN D...	Paneer L...	270
7	INDIAN D...	Paneer M...	270
8	INDIAN D...	Paneer T...	270
9	INDIAN D...	Paneer S...	270
10	INDIAN D...	Shahi Pa...	260
11	INDIAN D...	Kadhail ...	260
12	INDIAN D...	Mutter Pa...	250
13	INDIAN D...	Paneer B...	260
14	INDIAN D...	Paneer K...	270
15	INDIAN D...	Paneer T...	280
16	INDIAN D...	Paneer B...	300
17	INDIAN D...	Paneer A...	290
18	INDIAN D...	Veg. Kofta	260
19	INDIAN D...	Malai Kofta	280
20	INDIAN D...	Cheese ...	300
21	INDIAN D...	Mushroo...	290
22	INDIAN D...	Mushroo...	320
23	INDIAN D...	Mushroo...	320
24	INDIAN D...	Aloo Pudi...	240
25	INDIAN D...	Aloo Mutter	250

UPDATE

ItemNo.

Category

Item

Price

ADD

Category

Item

Price

FOODIES

- X

EMPL NO.	NAME	WORK	PHONE NO.	QUALIFICATION	ADDRESS
1	Pranjal	Sweeper	0000000000	Metric Pass	Jagdishpur, Ballia
2	12th pass				France

EmpNo.

Name

WorkType

Phone No.

Qualification

Address

DATABASE TABLES AND THEIR STRUCTURES

For the managers

	Field	Type	Null	Key	Default	Extra
►	id	int(11)	NO	PRI		auto_increment
	name	varchar(30)	YES			
	phone	varchar(30)	YES			
	address	varchar(30)	YES			
	password	varchar(30)	YES			
	username	varchar(30)	YES			

For the members

	Field	Type	Null	Key	Default	Extra
►	SubID	int(11)	NO	PRI		auto_increment
	Name	varchar(40)	YES			
	Phone	varchar(10)	YES			

For Menu and Bill

	Field	Type	Null	Key	Default	Extra
►	ino	int(11)	NO	PRI		auto_increment
	category	varchar(30)	YES			
	item	varchar(30)	YES			
	price	int(11)	YES			

SOURCE CODE

FOR LOGIN BUTTON

```
String p=new String(pPF.getPassword());    String id= uTF.getText();
try{    rs=stmt.executeQuery("select*from managers where username="
    +uTF.getText()+" AND PASSWORD='"+p+"'");
    if(rs.next()){    FirstPage mb=new FirstPage();    mb.show();    this.dispose();    }
    else {    JOptionPane.showMessageDialog(null,"USERNAME OR PASSWORD IS INCORRECT!!!");    }
catch(Exception ex){    ex.printStackTrace();    }
```

FOR RESET PASSWORD

```
String p=new String(pTF.getPassword());        String p1=new String(P1TF.getPassword());
String rp=new String(rpTF.getPassword());
if(p.equals(p1)){
    try{    stmt.executeUpdate("update cont set password='"+p+"' where folio= '"+rp+"'
        JOptionPane.showMessageDialog(null,"Your Password Was Changed!!");    }
catch(Exception ex){    ex.printStackTrace();    }
    Login sn = new Login();    sn.show();    this.dispose();    }
else {    JOptionPane.showMessageDialog(null,"TYPE THE CORRECT RECOVERY PIN");    }
```

THE BILL PAGE (AFTER initComponents())

```
totalTF.setText(""+0);
stmt=cn.createStatement(); stmt1=cn.createStatement(); stmt2 =cn.createStatement();
catCBO.setModel(dcm);    itemCBO.setModel(dcm1);
rs=stmt.executeQuery("select distinct(category) from newmenu ORDER BY CATEGORY");
while(rs.next()){    dcm.addElement(rs.getString("category"));    }    }
catch (Exception ex) {    ex.printStackTrace();    }
```

(for category CBO)

```
try{    String sql = "select * from newmenu where category='"+catCBO.getSelectedItem().toString()+"ORDER BY TEM";
rs1=stmt1.executeQuery(sql);    dcm1.removeAllElements();
    while(rs1.next()){    dcm1.addElement(rs1.getString("item"));    }    }
catch(Exception ex) {    ex.printStackTrace();    }
```

(for item CBO)

```
try{    String sql1 = "select * from newmenu where item='"+itemCBO.getSelectedItem().toString()+"";
rs2=stmt2.executeQuery(sql1);
    while(rs2.next()){    amtTF.setText(rs2.getString("price")+"" );    }    }
catch(Exception ex) {    ex.printStackTrace();    }
```

To navigate to any page

```
Membership mb=new Membership();    mb.show();
```

To update any record

```
try{    stmt.executeUpdate("update empl set name='"+nTF.getText()+"' where emplno='"+eTF.getText();    }
catch(Exception ex){    ex.printStackTrace();    }    JOptionPane.showMessageDialog(null,"SUCCESSFUL!!!");
```

To create the connection between MySql and JAVA

```
cn= DriverManager.getConnection("jdbc:mysql://localhost/d2155329","root","");
```

Restaurant Management System DFD Template

A library management system data flow diagram (DFD) shows how the information within a library system needs to flow to meet the needs of authorities, students, library administrators. Adapt the following restaurant management system DFD template to match the circumstances of your restaurant.

