

SL-1

A Mini Project Report

on

SALES AND INVENTORY

by

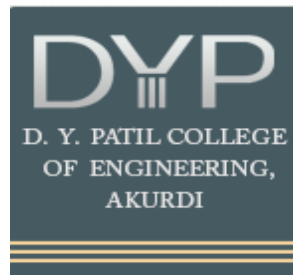
Atharva Chavan (T1851010)(TE A)

Satyam Raj (T1851051)(TEA)

Tejaswa Wadekar (T1851079)(TEB)

Aditya Bhatt (T1851102)(TEB)

[GitHub Link:- github.com/AtharvaChavan/SalesAndInventory/](https://github.com/AtharvaChavan/SalesAndInventory/)



Department of Information Technology

D. Y. PATIL COLLEGE OF ENGINEERING, AKURDI, PUNE

SAVITRIBAI PHULE PUNE UNIVERSITY

2020-2021

Date: 19/12/2020

CERTIFICATE

This is to certify that,

Atharva Chavan (T1851010)
Satyam Raj (T1851051)
Tejaswa Wadekar (T1851079)
Aditya Bhatt (T1851102)

of class T.E IT; have successfully completed their mini project work on “SALES AND INVENTORY” at D. Y. Patil College of Engineering in the partial fulfillment of the Graduate Degree course in T.E at the department of **Information Technology**, in the academic Year 2020-2021 Semester – I as prescribed by the Savitribai Phule Pune University.

Dr. Preeti Patil
Head of the Department
(Department of Information Technology)

Acknowledgement

We take this opportunity to thank our project guide Mrs. Madhuri Gurale and Head of the Department Dr. Preeti Patil for their valuable guidance and for providing all the necessary facilities, which were indispensable in the completion of this project report. We are also thankful to all the staff members of the Department of Information Technology of D. Y. Patil College of Engineering, Akurdi for their valuable time, support, comments, suggestions and persuasion. We would also like to thank the institute for providing the required facilities, Internet access and important books.

Atharva Chavan (T1851010)

Satyam Raj (T1851051)

Aditya Bhatt (T1851102)

Tejaswa Wadekar (T1851079)

Contents

Sr. No.	Topic	Page No.
	Acknowledgement	3
	Contents	4
	List of Tables/Collections	5
	List of Forms	6
	Abstract	8
Chapter-1	Introduction	9
	1.1 Motivation	9
	1.2 Problem Statement	10
	1.3 Framework of the proposed work in project	11
Chapter-2	Software Requirement Specification	12
	3.1 Hardware Requirements	12
	3.2 Software Requirements	13
Chapter-3	Entity-Relationship Diagram	14
Chapter-4	Tables / Collections	16
Chapter-5	Forms/jFrames	26
Chapter-6	Features	45
Chapter-7	Conclusion	46
	References	47

List of Tables

Table No.	Title	Page No.
4.1	Login	15
4.2	Category	17
4.3	Brand	18
4.4	Product	20
4.5	POS	22
4.6	Cashier	24

List of Forms

Figure No.	Title	Page No.
5.1	Login Form	26
5.2	Category Form	27
5.3	Brand Form	30
5.4	Product Form	33
5.5	POS Form	40
5.6	Cashier Form	43

Acronyms

POS	Point Of Sales
JDBC	Java DataBase Connectivity
SQL	Structured Query Language
QTY	Quantity

Abstract

This system helps in tracking records so that past records can be verified through them and one can make decisions based on the past records. This system completes the work in a very less time resulting in less time consumption and high level of efficiency.

This system is developed in such a way that even a naive user can also operate the system easily. The calculations are made very quickly and the records are directly saved into databases and the databases can be maintained for a longer period of time. Also this system provides a high level of security for data. The advantages are,

- Ensure data accuracy
- Proper control of the higher authority
- Minimize manual data entry
- Greater efficiency
- Better service
- User friendliness and interactive
- Minimum time required

Introduction

1.1 Motivation

1.1.1 Reducing Data Redundancy

The file based data management systems contained multiple files that were stored in many different locations in a system or even across multiple systems. Because of this, there were sometimes multiple copies of the same file which lead to data redundancy.

This is prevented in a database as there is a single database and any change in it is reflected immediately. Because of this, there is no chance of encountering duplicate data.

1.1.2 Sharing of Data

In a database, the users of the database can share the data among themselves. There are various levels of authorization to access the data, and consequently the data can only be shared based on the correct authorisation protocols being followed.

Many remote users can also access the database simultaneously and share the data between themselves.

1.1.3 Data Integrity

Data integrity means that the data is accurate and consistent in the database. Data Integrity is very important as there are multiple databases in a DBMS. All of these databases contain data that is visible to multiple users. So it is necessary to ensure that the data is correct and consistent in all the databases and for all the users.

1.1.4 Data Security

Data Security is a vital concept in a database. Only authorised users should be allowed to access the database and their identity should be authenticated using a username and password. Unauthorised users should not be allowed to access the database under any circumstances as it violates the integrity constraints.

1.1.5 Backup and Recovery

Database Management System automatically takes care of backup and recovery. The users don't need to backup data periodically because this is taken care of by the DBMS. Moreover, it also restores the database after a crash or system failure to its previous condition.

1.1.6 Privacy

The privacy rule in a database means only the authorized users can access a database according to its privacy constraints. There are levels of database access and a user can only view the data he is allowed to. For example - In social networking sites, access constraints are different for different accounts a user may want to access.

1.1.7 Data Consistency

Data consistency is ensured in a database because there is no data redundancy. All data appears consistently across the database and the data is the same for all the users viewing the database. Moreover, any changes made to the database are immediately reflected to all the users and there is no data inconsistency.

1.2 Problem Statement

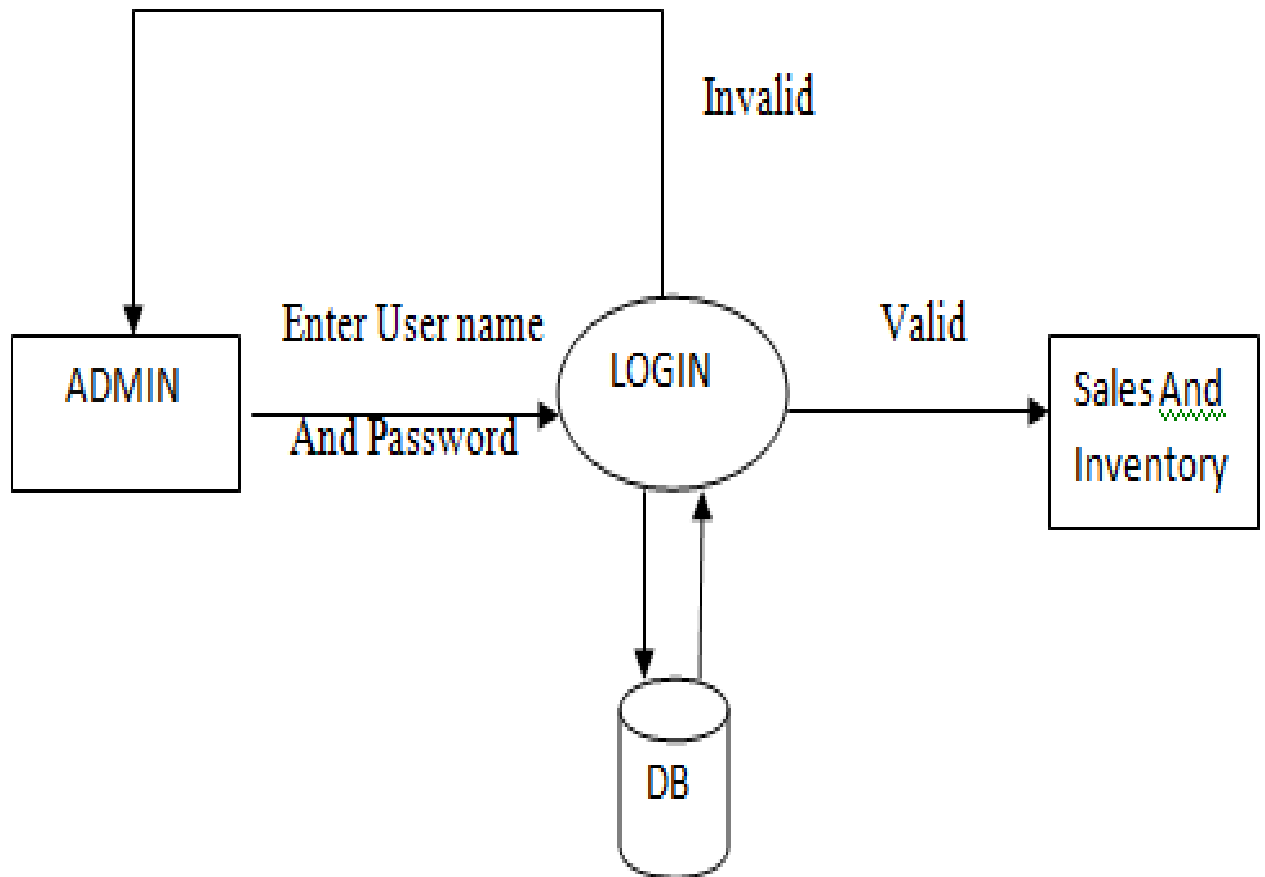
What is a POS (Point of Sale)?

Point Of Sale Definition: A Point of Sale (POS) is technically a system in a retail store from which you conduct the sale of physical goods. In a store, a POS is where the checkout happens, orders are processed and bills are paid.

Sales is an integral part of most companies. It is one of the most critical sections which have a direct impact on the profitability of the company.

Sales staff has to rely mostly on the old systems or paperwork to access information about the potential customer. The main perspective of this system is to solve all these problems. The system will help the Persons and Managers to manage to Sales and System in a better way.

1.3 Framework of the proposed work in project



Software Requirement Specification

1. Hardware Requirements

1.1. **Hard Disk:** SQL Server requires a minimum of 6 GB of available hard-disk space. Disk space requirements will vary with the SQL Server components you install. For more information, see **Hard Disk Space Requirements** later in this article. For information on supported storage types for data files, see **Storage Types for Data Files**. Installing SQL Server on computers with the NTFS or ReFS file formats is recommended. The FAT32 file system is supported but not recommended as it is less secure than the NTFS or ReFS file systems. Read-only, mapped, or compressed drives are blocked during installation.

1.2. **Drive:** A DVD drive, as appropriate, is required for installation from disc.

1.3. **Monitor:** SQL Server requires Super-VGA (800x600) or higher resolution monitor.

1.4. **Internet:** Internet functionality requires Internet access (fees may apply).

1.5. **Memory *:**

Minimum:

Express Editions: 512 MB

All other editions: 1 GB

Recommended:

Express Editions: 1 GB

All other editions: At least 4 GB and should be increased as database size increases to ensure optimal performance.

1.6. **Processor Speed:**

Minimum: x64 Processor: 1.4 GHz

Recommended: 2.0 GHz or faster

- 1.7. **Processor Type**x64 Processor: AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support

2. Software Requirements

- 2.1. **.NET Framework:** SQL Server 2016 (13.x) and later require .NET Framework 4.6 for the Database Engine, Master Data Services, or Replication. SQL Server setup automatically installs .NET Framework. You can also manually install .NET Framework from Microsoft .NET Framework 4.6 (Web Installer) for Windows. For more information, recommendations, and guidance about .NET Framework 4.6 see .NET Framework Deployment Guide for Developers.

Windows 8.1, and Windows Server 2012 R2 require KB2919355 before installing .NET Framework 4.6.

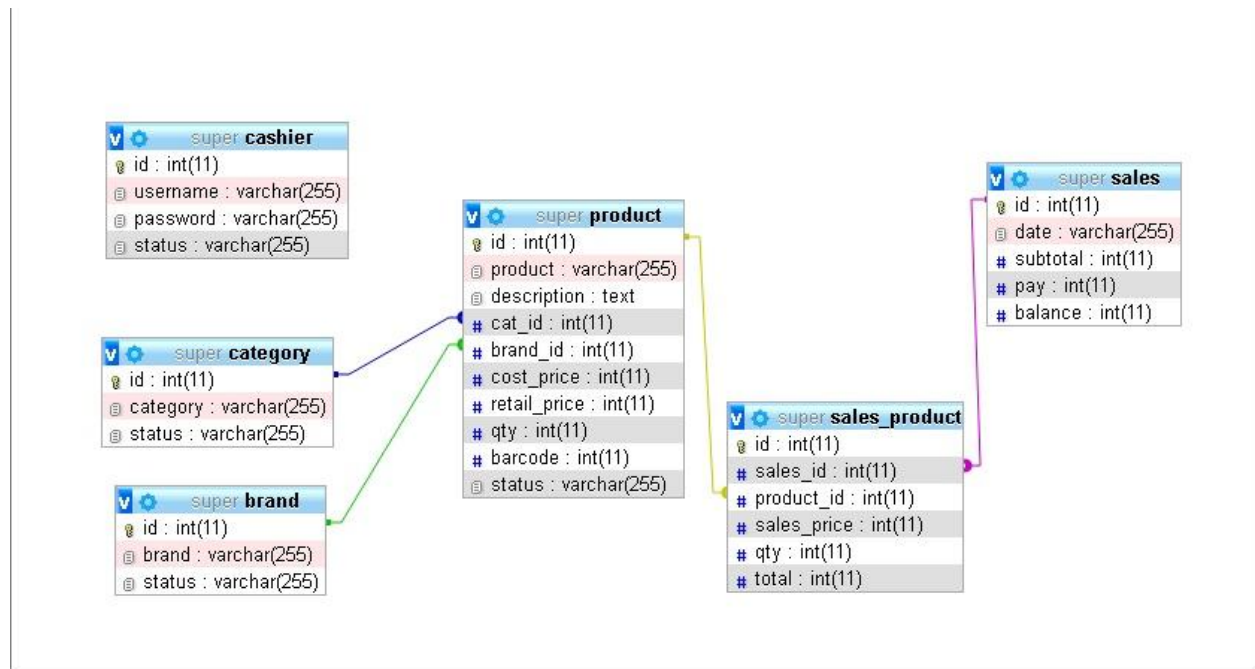
- 2.2. **Network Software:** Supported operating systems for SQL Server have built-in network software. Named and default instances of a stand-alone installation support the following network protocols: Shared memory, Named Pipes, TCP/IP, and VIA.

Note: VIA protocol is not supported on failover clusters. Clients or applications running on the same node of the failover cluster as the SQL Server instance, can use Shared Memory protocol to connect to SQL Server using its local pipe address. However this type of connection is not cluster-aware and will fail after an instance failover. It is therefore not recommended and should only be used in very specific scenarios.

Important: The VIA protocol is deprecated. This feature will be removed in a future version of Microsoft SQL Server. Avoid using this feature in new development work, and plan to modify applications that currently use this feature.

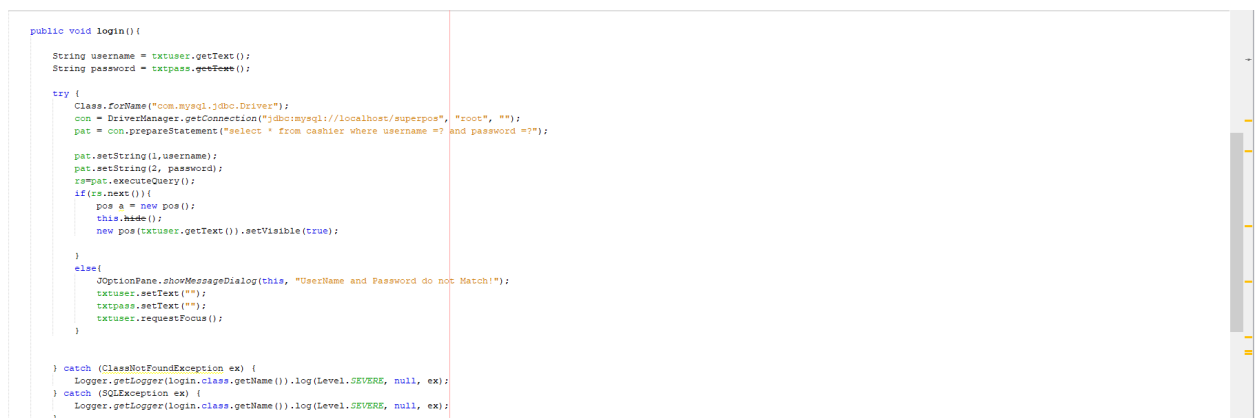
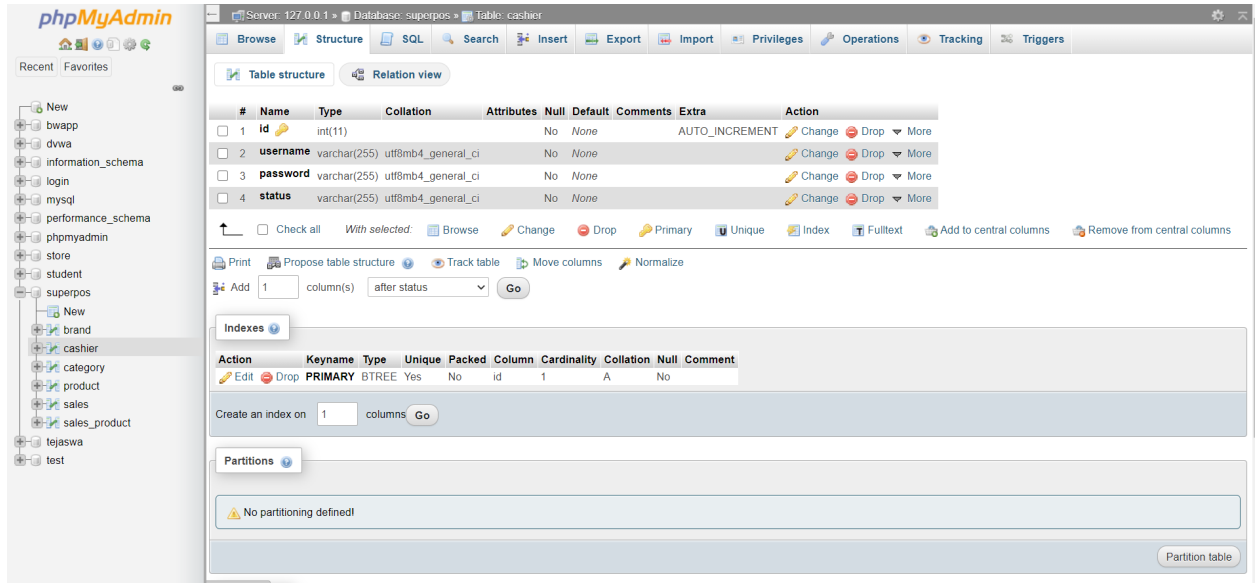
For more information about Network Protocols and Network Libraries, see Network Protocols and Network Libraries.

Entity-Relationship Diagram



Tables/Entities

1. Login



2. Category

The screenshot shows the phpMyAdmin interface with the 'category' table selected. The SQL query results are displayed, showing 0 rows. The query is: `SELECT * FROM `category``. The table structure is visible, showing columns: `id` (int(11)), `category` (varchar(255)), and `status` (varchar(255)).

Showing rows 0 - 0 (1 total, Query took 0.0011 seconds)

```
SELECT * FROM `category`
```

Options: ☐ Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations: [Print](#) [Copy to clipboard](#) [Export](#) [Display chart](#) [Create view](#)

Bookmark this SQL query: Label: ☐ Let every user access this bookmark [Bookmark this SQL query](#)

The screenshot shows the phpMyAdmin interface with the 'category' table selected. The 'Table structure' tab is active, displaying the table structure. The table has three columns: `id` (int(11)), `category` (varchar(255)), and `status` (varchar(255)). The `id` column is the primary key.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	category	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
3	status	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More

Indexes: [Add](#) 1 column(s) after status [Go](#)

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	1	A	No	

Partitions: No partitioning defined! [Partition table](#)


```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String category = txtcat.getText();
    String status = txtstatus.getSelectedItem().toString();

    try {
        Class.forName("com.mysql.jdbc.Driver");
        con1 = DriverManager.getConnection("jdbc:mysql://localhost/superpos", "root", "");
        pst = con1.prepareStatement("insert into category(category,status)values(?,?)");
        pst.setString(1, category);
        pst.setString(2, status);
        pst.executeUpdate();
        JOptionPane.showMessageDialog(null, "Category Addeddd");
        table_update();

        txtcat.setText("");
        txtstatus.setSelectedIndex(-1);
        txtcat.requestFocus();

    } catch (ClassNotFoundException ex) {
        Logger.getLogger(category.class.getName()).log(Level.SEVERE, null, ex);
    } catch (SQLException ex) {
        Logger.getLogger(category.class.getName()).log(Level.SEVERE, null, ex);
    }
}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    DefaultTableModel dl = (DefaultTableModel)jTable1.getModel();
    int selectIndex = jTable1.getSelectedRow();

    int id = Integer.parseInt(dl.getValueAt(selectIndex, 0).toString());
    String category = txtcat.getText();
    String status = txtstatus.getSelectedItem().toString();

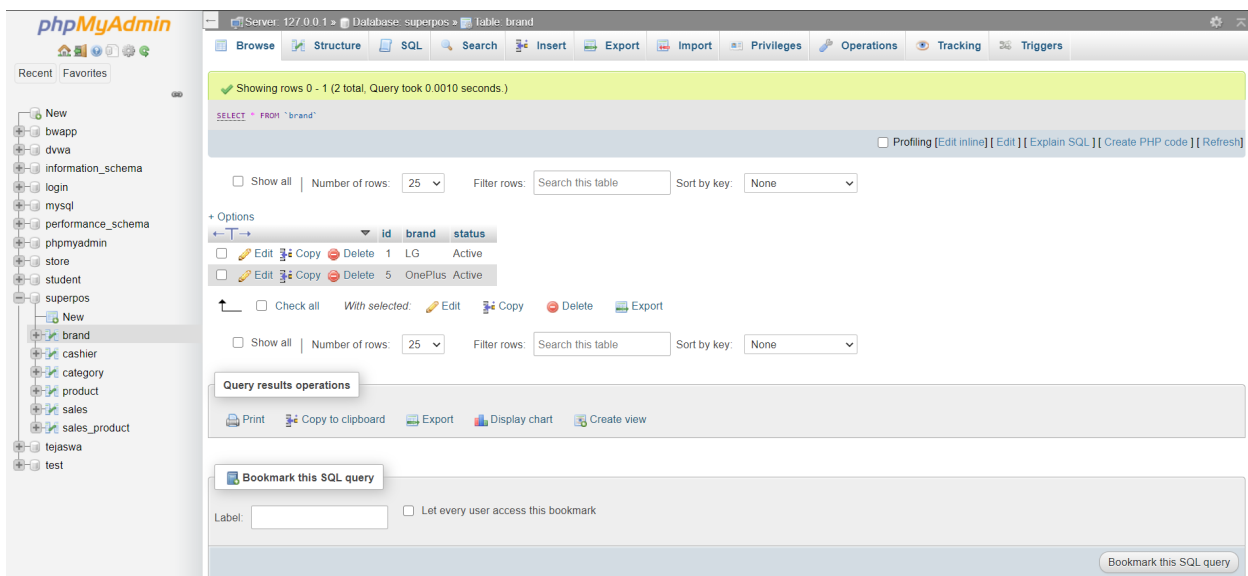
    try {
        Class.forName("com.mysql.jdbc.Driver");

        con1 = DriverManager.getConnection("jdbc:mysql://localhost/superpos", "root", "");
        pst = con1.prepareStatement("update category set category=?, status=? where id=?");
    }
}

```

3. Brand

```
private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
    DefaultTableModel d1 = (DefaultTableModel)jTable1.getModel();  
    int selectIndex = jTable1.getSelectedRow();  
    txtbrand.setText(d1.getValueAt(selectIndex, 1).toString());  
    txtstatus.setSelectedItem(d1.getValueAt(selectIndex, 2).toString());  
}  
  
private void jButtonActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    DefaultTableModel d1 = (DefaultTableModel)jTable1.getModel();  
    int selectIndex = jTable1.getSelectedRow();  
    int id = Integer.parseInt(d1.getValueAt(selectIndex, 0).toString());  
  
    int dialogResult = JOptionPane.showConfirmDialog(null, "Do you want to Delete the Record","Warning",JOptionPane.YES_NO_OPTION);  
    if(dialogResult== JOptionPane.YES_OPTION){  
        try {  
            Class.forName("com.mysql.jdbc.Driver");  
            con1 = DriverManager.getConnection("jdbc:mysql://localhost/superpos", "root", "");  
            pst = con1.prepareStatement("Delete from brand where id =?");  
            pst.setInt(1, id);  
            pst.executeUpdate();  
            JOptionPane.showMessageDialog(null,"Brand Deleted");  
            table.update();  
            txtbrand.setText("");  
            txtstatus.setSelectedIndex(-1);  
            txtbrand.requestFocus();  
        } catch (ClassNotFoundException ex) {  
            Logger.getLogger(brand.class.getName()).log(Level.SEVERE, null, ex);  
        } catch (SQLException ex) {  
            Logger.getLogger(brand.class.getName()).log(Level.SEVERE, null, ex);  
        }  
    }  
}
```



phpMyAdmin

Recent Favorites

- New
- bwapp
- dwva
- information_schema
- login
- mysql
- performance_schema
- phpmyadmin
- store
- student
- superpos
- brand
- cashier
- category
- product
- sales
- sales_product
- tejaswa
- test

Server: 127.0.0.1 » Database: superpos » Table: brand

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)		No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	brand	varchar(255)	utf8mb4_general_ci	No	None			Change Drop More
<input type="checkbox"/>	3	status	varchar(255)	utf8mb4_general_ci	No	None			Change Drop More

☐ Check all With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Fulltext](#) [Add to central columns](#) [Remove from central columns](#)

[Print](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Normalize](#)

[Add](#) 1 column(s) after status [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	2	A	No	

Create an index on 1 columns [Go](#)

Partitions

No partitioning defined!

[Partition table](#)

Information

Console

4. Product

```
private void jButtonActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    DefaultTableModel d1 = (DefaultTableModel)jTable1.getModel();  
    int selectIndex = jTable1.getSelectedRow();  
  
    int id = Integer.parseInt(d1.getValueAt(selectIndex, 0).toString());  
    String product = txtpro.getText();  
    String desc = txtDESC.getText();  
    CategoryItem citem = (CategoryItem)txtcat.getSelectedItems();  
    BrandItem bitem = (BrandItem)txtbrand.getSelectedItems();  
    String cprice = txtcosp.getText();  
    String rprice = txtretailp.getText();  
    String qty = txtqty.getText();  
    String barcode = txtbarcode.getText();  
  
    String status = txtstatus.getSelectedItems().toString();  
  
    try {  
        Class.forName("com.mysql.jdbc.Driver");  
  
        con1 = DriverManager.getConnection("jdbc:mysql://localhost/superpos", "root", "");  
        pat = con1.prepareStatement("update product set product=?,description=?,cat_id=?,brand_id=?,cost_price=?,retail_price=?,qty=?,barcode=?,status=? where id=?");  
  
        pat.setString(1, product);  
        pat.setString(2, desc);  
        pat.setInt(3, citem.id);  
        pat.setInt(4, bitem.id);  
        pat.setString(5, cprice);  
        pat.setString(6, rprice);  
        pat.setString(7, qty);  
        pat.setString(8, barcode);  
        pat.setString(9, status);  
        pat.setInt(10, id);  
  
        pat.executeUpdate();  
        JOptionPane.showMessageDialog(null, "Product Updated!!!");  
        table_update();  
    }  
}
```

Server: 127.0.0.1 » Database: superpos » Table: product

Showing rows 0 - 2 (3 total, Query took 0.0007 seconds)

SELECT * FROM `product`

Options: Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

	id	product	description	cat_id	brand_id	cost_price	retail_price	qty	barcode	status
<input type="checkbox"/>	1	LG S2	LG S2 X	2	1	12000	13000	2	144422	DeActive
<input type="checkbox"/>	3	LS2	LS2	2	1	140000	150000	10	123	Active
<input type="checkbox"/>	4	OnePlus 8T 5G	Aquamarine Green, 12GB RAM, 256GB Storage	6	5	45999	53000	15	321	Active

Query results operations: Print, Copy to clipboard, Export, Display chart, Create view

Bookmark this SQL query: Label: ☐ Let every user access this bookmark

Console: Bookmark this SQL query

phpMyAdmin

Server: 127.0.0.1 » Database: superpos » Table: product

Recent Favorites

Table structure Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	product	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
3	description	text	utf8mb4_general_ci		No	None			Change Drop More
4	cat_id	int(11)			No	None			Change Drop More
5	brand_id	int(11)			No	None			Change Drop More
6	cost_price	int(11)			No	None			Change Drop More
7	retail_price	int(11)			No	None			Change Drop More
8	qty	int(11)			No	None			Change Drop More
9	barcode	int(11)			No	None			Change Drop More
10	status	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More

☐ Check all
 With selected: ☐ Browse ☐ Change ☐ Drop ☐ Primary ☐ Unique ☐ Index ☐ Fulltext ☐ Add to central columns ☐ Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after status Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	2	A	No	

Create an index on 1 columns Go

5. POS

```
private void sales() {
    DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy/MM/dd");
    LocalDateTime now = LocalDateTime.now();
    String date = dtf.format(now);

    String subtot = txtsub.getText();
    String pay = txtpay.getText();
    String bal = txtbal.getText();
    int lastinsertid = 0;

    try {
        Class.forName("com.mysql.jdbc.Driver");

        con1 = DriverManager.getConnection("jdbc:mysql://localhost/superpos", "root", "");
        String query = "insert into sales (date, subtotal, pay, balance) values (?, ?, ?, ?)";
        insert = con1.prepareStatement(query, Statement.RETURN_GENERATED_KEYS);
        insert.setString(1, date);
        insert.setString(2, subtot);
        insert.setString(3, pay);
        insert.setString(4, bal);
        insert.executeUpdate();

        ResultSet generatedKeyResult = insert.getGeneratedKeys();

        if (generatedKeyResult.next()) {
            lastinsertid = generatedKeyResult.getInt(1);
        }
        JOptionPane.showMessageDialog(this, lastinsertid);

    } catch (ClassNotFoundException ex) {
        Logger.getLogger(pos.class.getName()).log(Level.SEVERE, null, ex);
    } catch (SQLException ex) {
        Logger.getLogger(pos.class.getName()).log(Level.SEVERE, null, ex);
    }
}
```

The screenshot shows the phpMyAdmin web interface. On the left is a sidebar with a tree view of databases and tables, including 'superpos' and its sub-tables like 'brand', 'cashier', 'category', 'product', 'sales', and 'sales_product'. The main area displays the 'Table: sales' view. A green banner at the top indicates 'Showing rows 0 - 4 (5 total, Query took 0.0018 seconds)'. Below this, a SQL query 'SELECT * FROM `sales`' is shown. The table data is presented in a grid with columns: id, date, subtotal, pay, and balance. There are five rows of data. Below the table, there are options for 'Check all', 'With selected', 'Edit', 'Copy', 'Delete', and 'Export'. At the bottom, there is a section for 'Query results operations' with buttons for 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'. A 'Bookmark this SQL query' section is also visible with a label input field and a checkbox 'Let every user access this bookmark'.

	id	date	subtotal	pay	balance
<input type="checkbox"/>	1	2020/12/12	1050000	1500000	13850000
<input type="checkbox"/>	2	2020/12/12	750000	1500000	750000
<input type="checkbox"/>	4	2020/12/13	750000	1000000	9250000
<input type="checkbox"/>	5	2020/12/13	750000	1200000	450000

phpMyAdmin

Server: 127.0.0.1 • Database: superpos • Table: sales

Recent Favorites

- New
- bwapp
- dwva
- information_schema
- login
- mysql
- performance_schema
- phpmyadmin
- store
- student
- superpos
 - New
 - brand
 - cashier
 - category
 - product
 - sales
 - sales_product
 - tejaswa
 - test

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)		No	None			AUTO_INCREMENT	Change Drop More
2	date	varchar(255)	utf8mb4_general_ci	No	None				Change Drop More
3	subtotal	int(11)		No	None				Change Drop More
4	pay	int(11)		No	None				Change Drop More
5	balance	int(11)		No	None				Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after balance Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	4	A	No	

Create an index on 1 columns Go

Partitions

No partitioning defined!

Console

phpMyAdmin

Server: 127.0.0.1 • Database: superpos • Table: sales_product

Recent Favorites

- New
- bwapp
- dwva
- information_schema
- login
- mysql
- performance_schema
- phpmyadmin
- store
- student
- superpos
 - New
 - brand
 - cashier
 - category
 - product
 - sales
 - sales_product
 - tejaswa
 - test

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)		No	None			AUTO_INCREMENT	Change Drop More
2	sales_id	int(11)		No	None				Change Drop More
3	product_id	int(11)		No	None				Change Drop More
4	sales_price	int(11)		No	None				Change Drop More
5	qty	int(11)		No	None				Change Drop More
6	total	int(11)		No	None				Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after total Go

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	0	A	No	

Create an index on 1 columns Go

Partitions

No partitioning defined!

Console

6. Cashier

```
private void jTableMouseClicked(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
    DefaultTableModel dt = (DefaultTableModel) jTable1.getModel();  
    int selectIndex = jTable1.getSelectedRow();  
  
    txtuser.setText(dt.getValueAt(selectIndex, 1).toString());  
    txtstatus.setSelectedItem(dt.getValueAt(selectIndex, 2).toString());  
}  
  
private void jButtonActionPerformed(java.awt.event.ActionEvent evt) {  
    DefaultTableModel dt = (DefaultTableModel) jTable1.getModel();  
    int selectIndex = jTable1.getSelectedRow();  
    int id = Integer.parseInt(dt.getValueAt(selectIndex, 0).toString());  
  
    int dialogResult = JOptionPane.showConfirmDialog(null, "Do you want to Delete the Record", "Warning", JOptionPane.YES_NO_OPTION);  
  
    if (dialogResult == JOptionPane.YES_OPTION) {  
        try {  
            Class.forName("com.mysql.jdbc.Driver");  
            con1 = DriverManager.getConnection("jdbc:mysql://localhost/superpos", "root", "");  
            pst = con1.prepareStatement("Delete from cashier Where id =?");  
            pst.setInt(1, id);  
            pst.executeUpdate();  
            JOptionPane.showMessageDialog(null, "Brand Deleted");  
            table_update();  
            txtuser.setText("");  
            txtstatus.setSelectedIndex(-1);  
            txtuser.requestFocus();  
        } catch (ClassNotFoundException ex) {  
            Logger.getLogger(cashier.class.getName()).log(Level.SEVERE, null, ex);  
        } catch (SQLException ex) {  
            Logger.getLogger(cashier.class.getName()).log(Level.SEVERE, null, ex);  
        }  
    }  
}
```

phpMyAdmin

Recent Favorites

- New
- bwapp
- dwva
- information_schema
- login
- mysql
- performance_schema
- phpmyadmin
- store
- student
- superpos
 - New
 - brand
 - cashier**
 - category
 - product
 - sales
 - sales_product
- tejaswa
- test

Server: 127.0.0.1 » Database: superpos » Table: cashier

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Showing rows 0 - 1 (2 total, Query took 0.0024 seconds)

SELECT * FROM `cashier`

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	id	username	password	status
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	1	Tejaswa	Tejas@123	Active
<input type="checkbox"/> Edit <input type="button" value="Copy"/> <input type="button" value="Delete"/>	2	Atharva	Atharva@123	Active

☐ Check all | With selected:

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Query results operations

Bookmark this SQL query

Label: ☐ Let every user access this bookmark

phpMyAdmin

Server: 127.0.0.1 • Database: superpos • Table: cashier

Recent Favorites

- New
- bwapp
- dwva
- information_schema
- login
- mysql
- performance_schema
- phpmyadmin
- store
- student
- superpos
 - New
 - brand
 - cashier
 - category
 - product
 - sales
 - sales_product
- tejaswa
- test

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)		No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	username	varchar(255) utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/>	3	password	varchar(255) utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/>	4	status	varchar(255) utf8mb4_general_ci		No	None			Change Drop More

☐ Check all With selected: [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Fulltext](#) [Add to central columns](#) [Remove from central columns](#)

[Print](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Normalize](#)

[Add](#) 1 column(s) after status [Go](#)

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	1	A	No	

Create an index on 1 column(s) [Go](#)

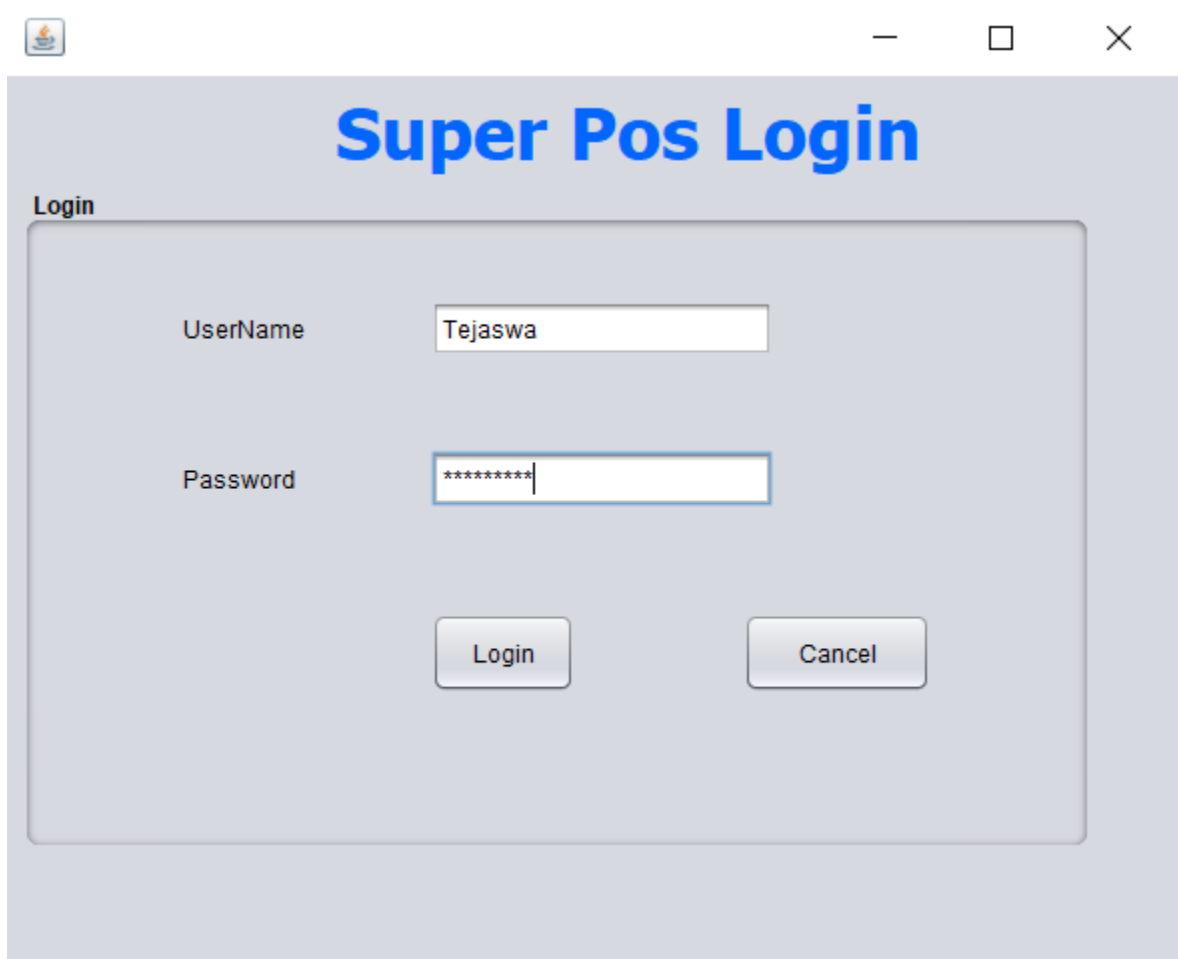
Partitions

No partitioning defined

[Partition table](#)

jForms

1.Login Form



The image shows a screenshot of a Java Swing window titled "Super Pos Login". The window has a standard Mac OS X-style title bar with a red, yellow, and green button on the left, and minimize, maximize, and close buttons on the right. The main content area has a light gray background. At the top, the text "Super Pos Login" is displayed in a large, bold, blue font. Below this, the word "Login" is written in a smaller, black font. The login form itself is a rounded rectangle with a light gray border. It contains two labels: "UserName" and "Password", both in a black font. The "UserName" label is positioned to the left of a text input field that contains the text "Tejaswa". The "Password" label is positioned to the left of a text input field that contains seven asterisks "*****". Below the input fields, there are two buttons: "Login" and "Cancel", both with a light gray background and a thin black border. The "Login" button is on the left and the "Cancel" button is on the right.

2) Category Form

Category
Brand
Product
Pos
Exit
Cashier

Super POS

Category

Category

Mobile

Status

Active

Add

Edit

Delete

id	Category	Status
----	----------	--------

Category
Brand
Product
Pos
Exit
Cashier

Super POS

Category

Category

Mobile

Status


Active

Add

Edit

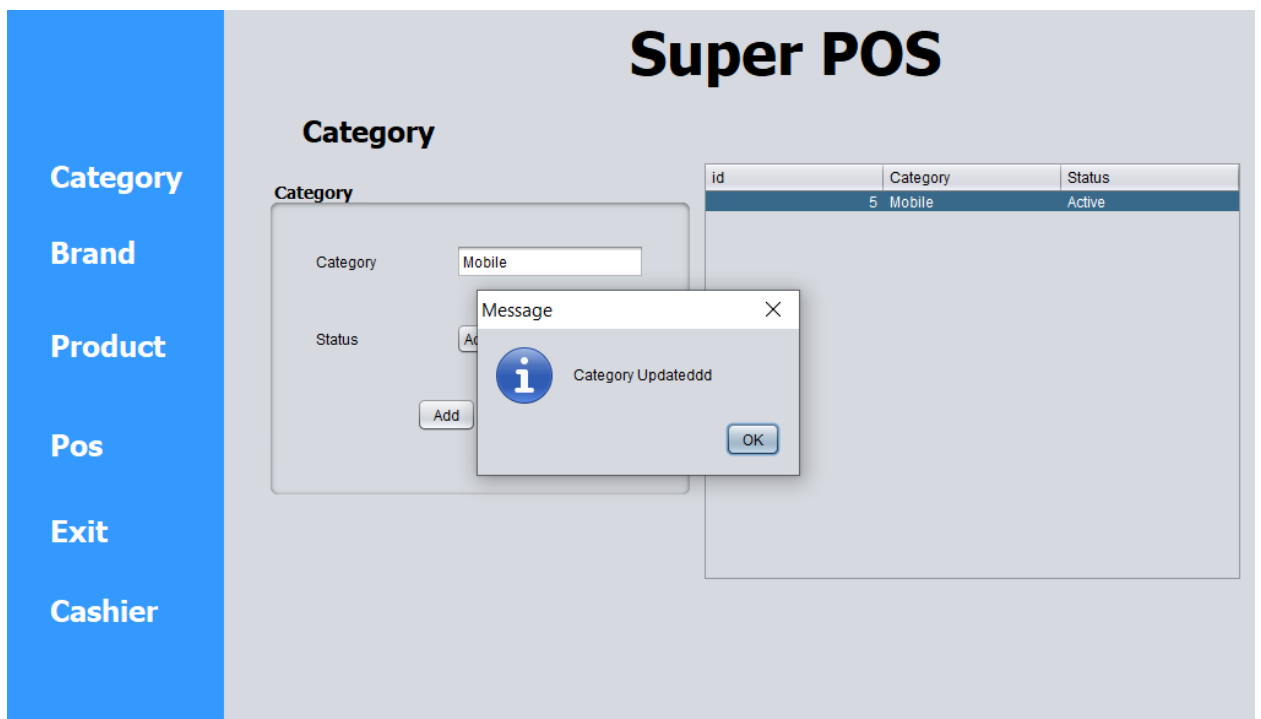
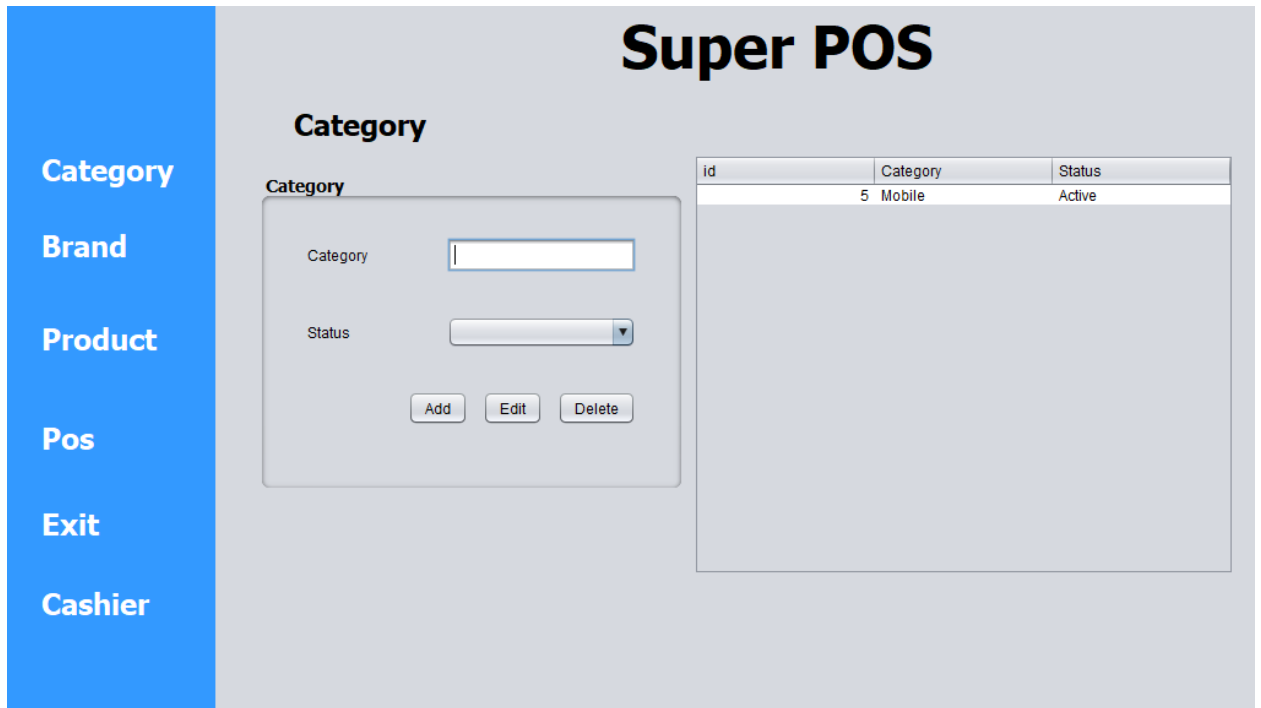
Delete

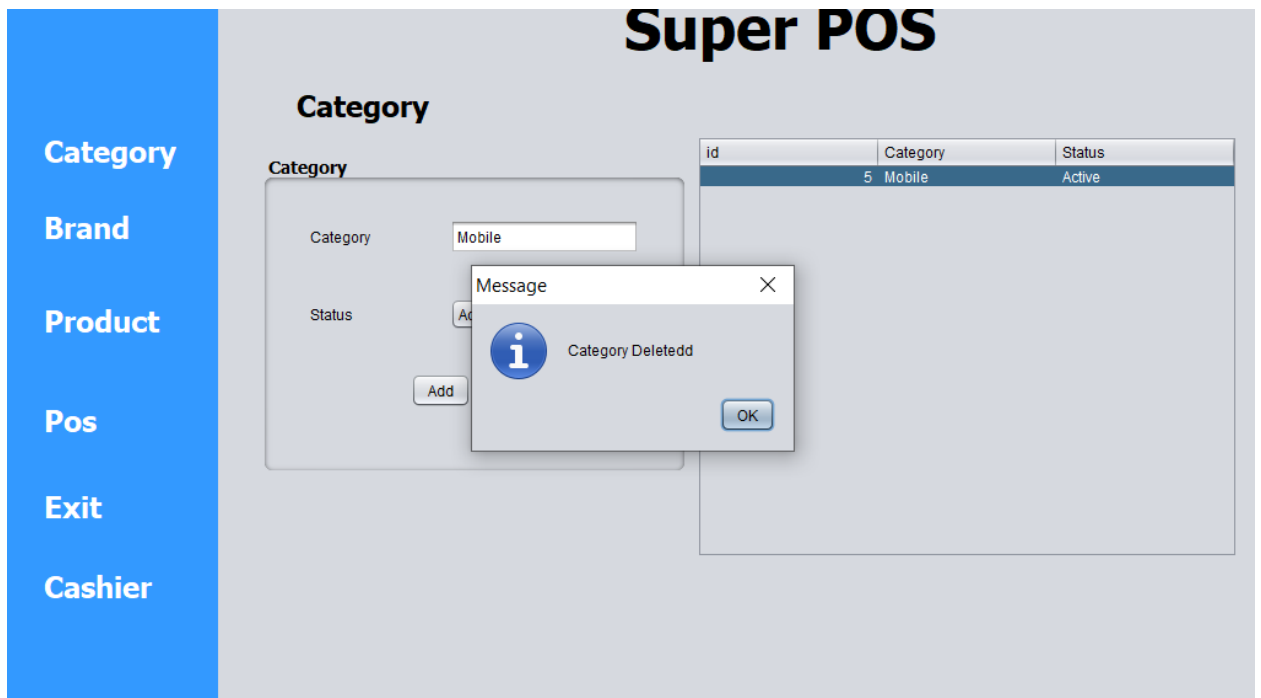
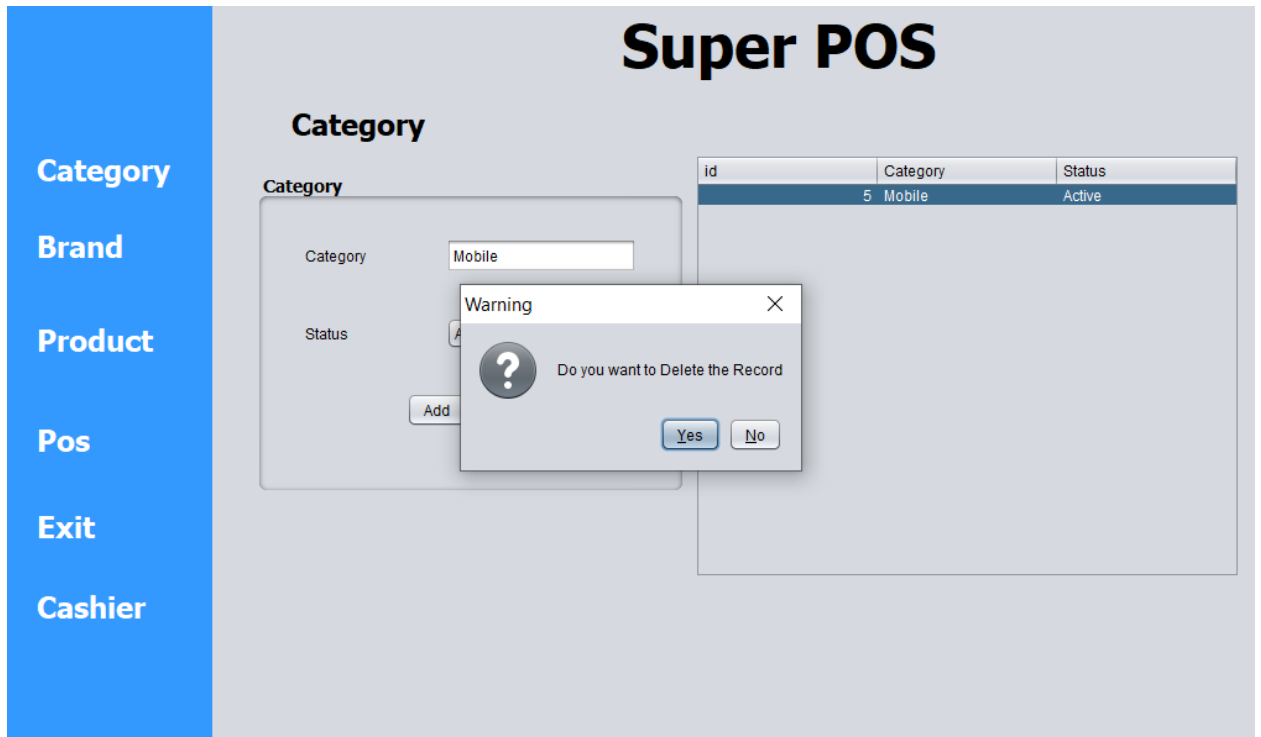
Message

 Category Addedddd

OK

id	Category	Status
----	----------	--------





2) Brand Form

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Brand

Brand

OnePlus

Status

Active

Add

Edit

Delete

id	Brand	Status
1	LG	Active
4	Sony	Active

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Brand

Brand

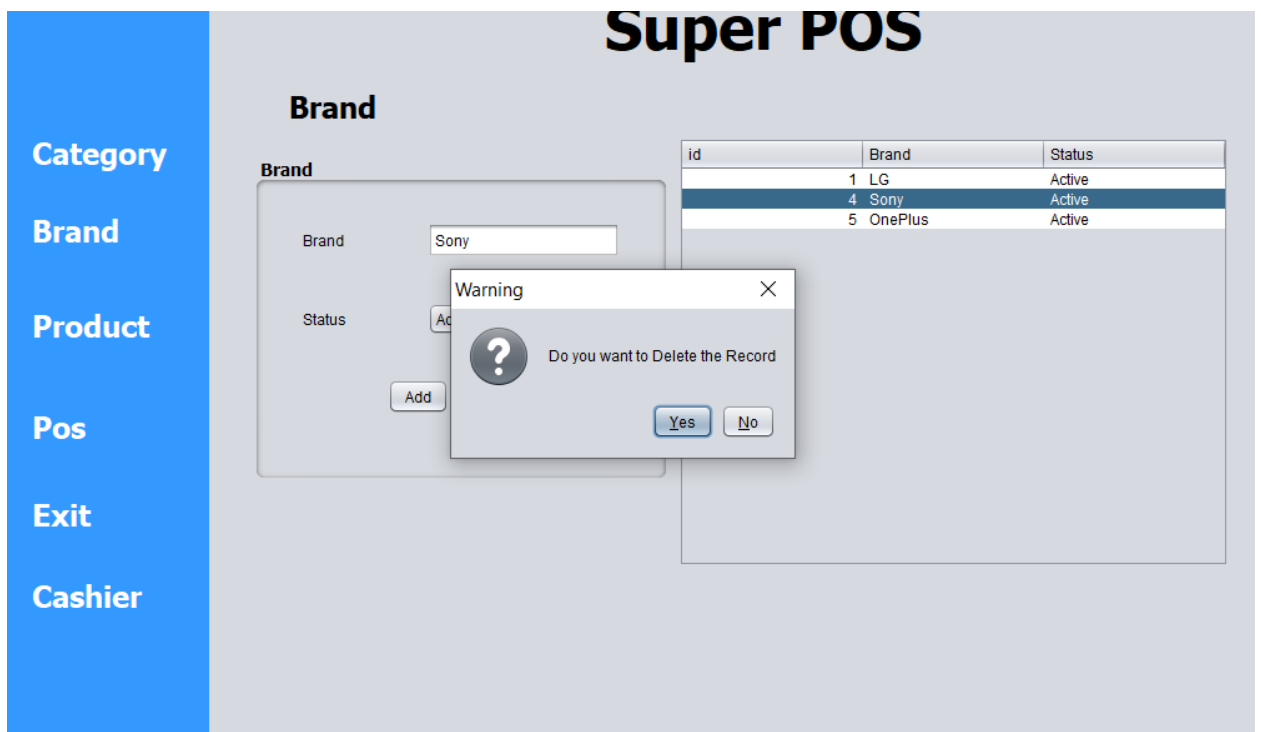
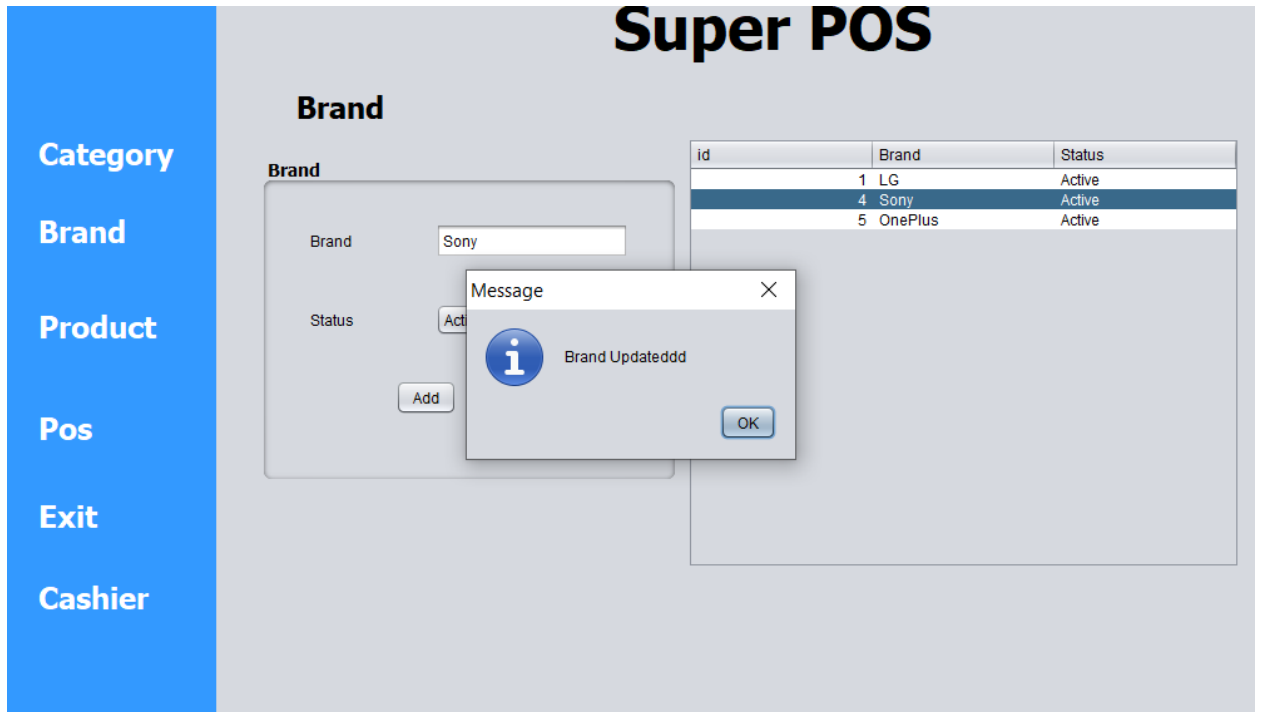
Status

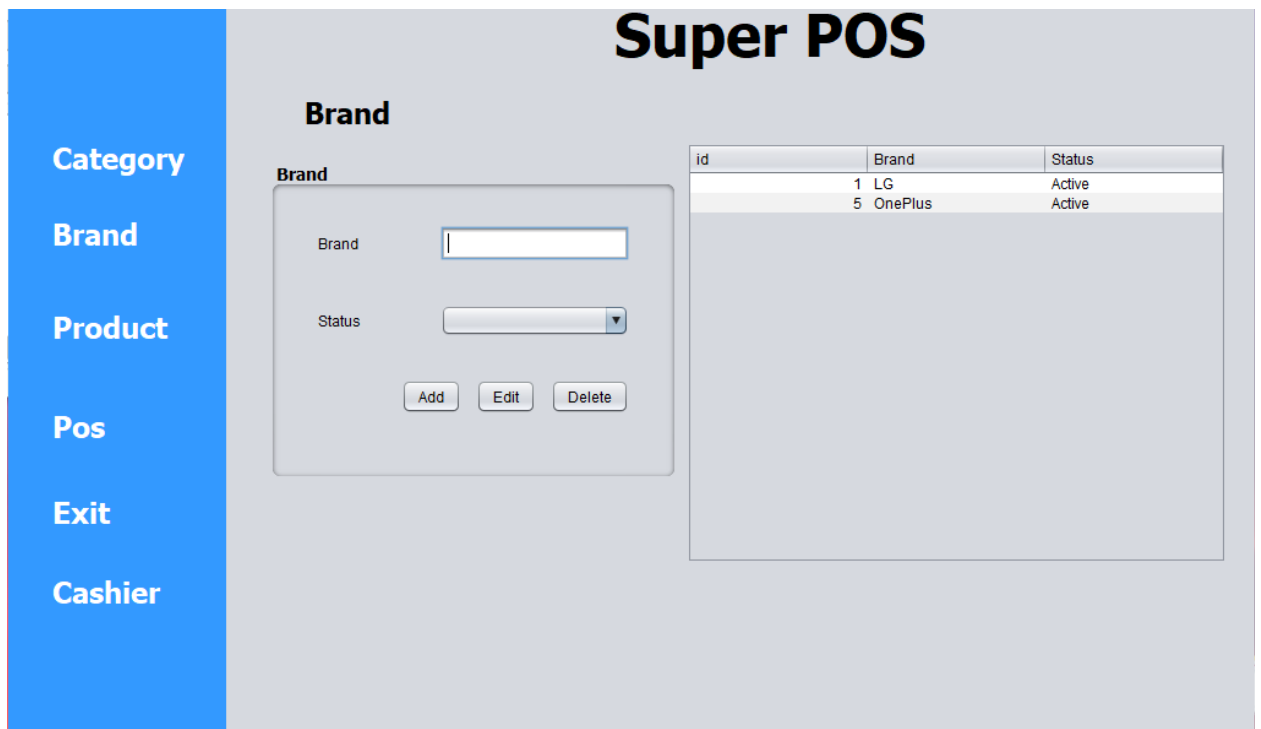
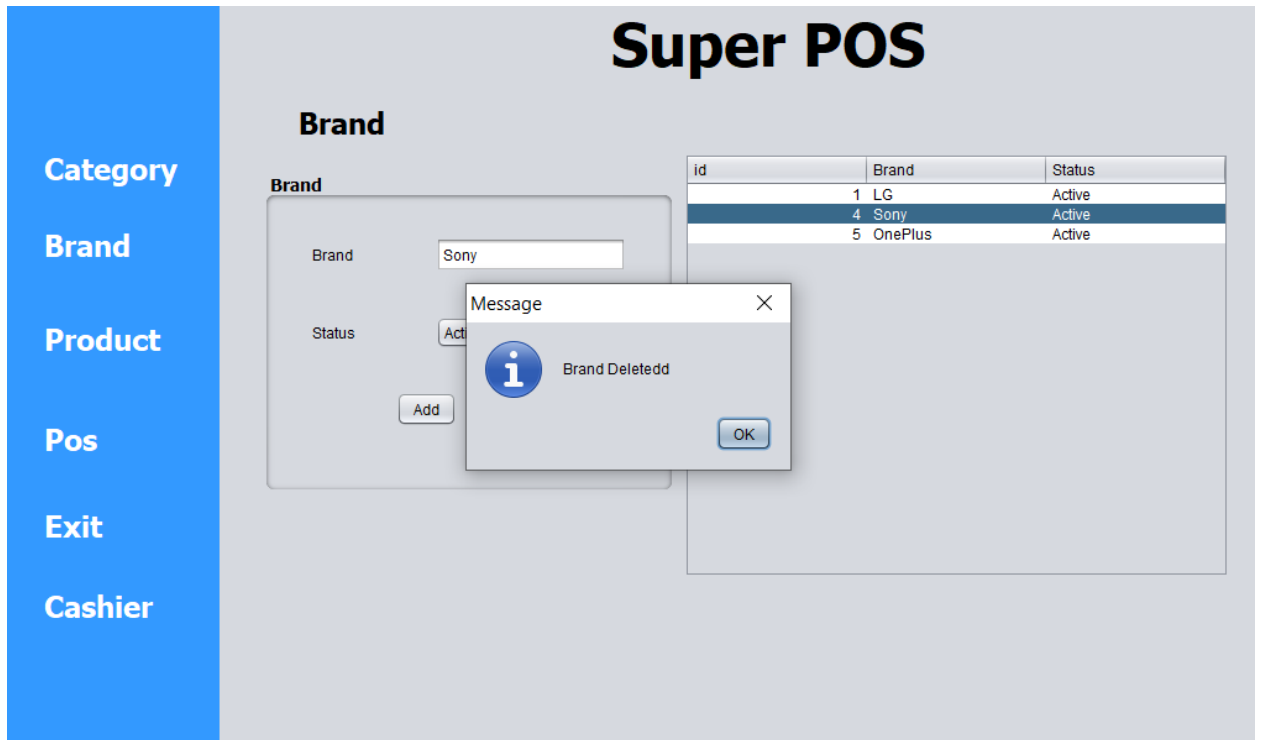
Add

Edit

Delete

id	Brand	Status
1	LG	Active
4	Sony	Active
5	OnePlus	Active





4) Product Form

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Product

Product

OnePlus 8T 5G

Retail price

53000

Description

ne Green, 12GB RAM, 256GB Storage

Qty

15

Category

mobile

Barcode

321

Brand

OnePlus

Status

Active

Cost price

45999

Add

Edit

Delete

Message

i

Product Addedddd

OK

id	Product	Description	Category	Brand	Cost Price	Retail Price	Qty	Barcode	Status
----	---------	-------------	----------	-------	------------	--------------	-----	---------	--------

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Product

Product

OnePlus 8T 5G

Retail price

53000

Description

ne Green, 12GB RAM, 256GB Storage

Qty

15

Category

mobile

Barcode

321

Brand

OnePlus

Status

Active

Cost price

45999

Add

Edit

Delete

Id	Product	Description	Category	Brand	Cost Price	Retail Price	Qty	Barcode	Status
----	---------	-------------	----------	-------	------------	--------------	-----	---------	--------

34

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Product

Product

Retail price

Description

Qty

Category

Barcode

Brand

Status

Cost price

Add

Edit

Delete

Id	Product	Description	Category	Brand	Cost Price	Retail Price	Qty	Barcode	Status
4	OnePlus 8T 5G	Aquamarine Gre...	mobile	OnePlus	45999	53000	15	321	Active

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Product

Product

OnePlus 8T 5G

Description

Aquamarine Green, 12GB RAM, 256GB Storage

Category

mobile

Brand

OnePlus

Cost price

45999

Retail price

53000

Qty

15

Barcode

321

Status

Active

Add

Edit

Delete

Warning

Do you want to Delete the Record

Yes

No

id	Product	Description	Category	Brand	Cost Price	Retail Price	Qty	Barcode	Status
4	OnePlus 8T 5G	Aquamarine Gre...	mobile	OnePlus	45999	53000	15	321	Active

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Product

Product

Product

OnePlus 8T 5G

Retail price

53000

Description

Aquamarine Green, 12GB RAM, 256GB Storage

Qty

15

Category

mobile

Barcode

321

Brand

OnePlus

Status

Active

Cost price

45999

Add

Edit

Delete

Message

i

Product Deleted

OK

id	Product	Description	Category	Brand	Cost Price	Retail Price	Qty	Barcode	Status
4	OnePlus 8T 5G	Aquamarine Gre...	mobile	OnePlus	45999	53000	15	321	Active

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Product

Product

Product

LG M31

Retail price

21000

Description

mirage Black, 8GB RAM, 128GB Storage

Qty

10

Category

mobile

Barcode

456

Brand

LG

Status

Active

Cost price

20000

Add

Edit

Delete

Message

i

Product Updatedddd

OK

id	Product	Description	Category	Brand	Cost Price	Retail Price	Qty	Barcode	Status
5	LG M31	Mirage Black, 8G...	mobile	LG	20000	21000	5	456	Active

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Product

Product

Retail price

Description

Qty

Category

Barcode

Brand

Status

Cost price

Add

Edit

Delete

id	Product	Description	Category	Brand	Cost Price	Retail Price	Qty	Barcode	Status
5	LG M31	Mirage Black, 8G...	mobile	LG	20000	21000	10	456	Active

39

5) POS Form

Category

Brand

Product

Pos

Exit

Cashier

Super POS

jLabel16

POS

Product

Product code

123

Product name

LS2

Price

150000

Qty

10

Add

Product Code	Product Name	Price	Qty	Total
--------------	--------------	-------	-----	-------

Delete

Subtotal

Pay

Balance

Pay Invoice

Category

Brand

Product

Pos

Exit

Cashier

Super POS

jLabel16

POS

Product

Product code

123

Product name

LS2

Price

150000

Qty

10

Add

Product Code	Product Name	Price	Qty	Total
--------------	--------------	-------	-----	-------

Delete

Subtotal

Pay

Balance

Pay Invoice

Message

i

Available Product = 10

OK

40

Category
Brand
Product
Pos
Exit
Cashier

Super POS

POS

jLabel16

Product

Product code

123

Product name

LS2

Price

150000

Qty

10

Add

Product Code

Product Name

Total

Message

i

Qty is not Enough

OK

Delete

Subtotal

Pay

Balance

Pay Invoice

Category
Brand
Product
Pos
Exit
Cashier

Super POS

POS

Tejaswa

Product

Product code

Product name

Price

Qty

Add

Product Code

Product Name

Price

Qty

Total

123

LS2

150000

5

750000

Delete

Subtotal

750000


Pay

1200000

Balance

450000

Pay Invoice

 **Print** ✕

General | Page Setup | Appearance

Print Service

Name: Microsoft Print to PDF ▼ Properties...

Status: Accepting jobs

Type:

Info: ☐ Print To File

Print Range

☒ All


☐ Pages 1 To 1

Copies

Number of copies: 1 ▲ ▼

☐ Collate

Print Cancel

 — □ ✕

*****SuperPos*****

Product	Price	TotalLS2	150000	750000
---------	-------	----------	--------	--------

SubTotal750000
Pay1200000
Balance450000

Thank you come again.....
|

6) Cashier Form

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Cashier

Cashier

Username

Atharva

Password

Status

Active

Add

Edit

Delete

id	Username	Status
1	Tejaswa	Active

Category

Brand

Product

Pos

Exit

Cashier

Super POS

Cashier

Cashier

Username

Atharva

Password

Status

Active

Add

Edit

Delete

id	Username	Status
1	Tejaswa	Active

Message

i

User Addedddd

OK

Super POS

Category

Brand

Product

Pos

Exit

Cashier

Cashier

Cashier

Username

Password

Status

Add

Edit

Delete

id	Username	Status
1	Tejaswa	Active
2	Atharva	Active

2

44

Features

1. TECHNICAL FEASIBILITY

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within the latest technology. Though the technology may become obsolete after some period of time, due to the fact that newer versions of the same software supports older versions, the system may still be used. So there are minimal constraints involved with this project. The system has been developed using Java. The project is technically feasible for development.

2. ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give the best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it gives an indication that the system is economically possible for development.

3. BEHAVIORAL FEASIBILITY

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioural aspects are considered carefully and conclude that the project is behaviourally feasible.

Conclusion

The project titled as “Sales Management System” is a desktop based application. This system provides facilities for assigning targets, add lead, add product, add activity, view previous sales etc. This system is developed with scalability in mind. Additional modules can be easily added when necessary. The system is developed with a modular approach. All modules in the system have been tested with valid data and invalid data and everything works successfully. Thus the system has fulfilled all the objectives identified and is able to replace the existing system.

The project has been completed successfully with the maximum satisfaction of the organization. The constraints are met and overcome successfully. The system is designed as it was decided in the design phase. The project gives a good idea on developing a full fledged application satisfying the user requirements. The system is very flexible and versatile. This system has a user-friendly screen that enables the user to use without any inconvenience. Validation checks induced have greatly reduced errors. Provisions have been made to upgrade the system. The application has been tested with live data and has provided a successful result. Hence the system has proved to work efficiently

References

- github.com/AtharvaChavan/SalesAndInventory/
- <https://www.w3schools.com/sql/>
- <https://docs.oracle.com/en/java/>
- <https://dev.mysql.com/doc/>
- <https://docs.joomla.org/API16:JForm>