Shop Management System

**Overview**

Shop Management system is a simplified console system that includes the functions of goods management, Goods management, and forefront payment.

1. Forefront payment includes: System Login, Sales Cart and Payment functions
2. Backend management display the sales today by Clerk or Goods categories.
3. Goods maintenance includes Add, Change, Delete, Search, Display List functions
4. Cashier management includes: adding, changing, deleting, searching, display functions.

**System environment**

1. Windows 7
2. Eclipse IDE fNeon.2 Release (4.6.2) with Java SE 1.8
3. MySQL Sever 5.7
4. mysql-connector-java-5.1.41-bin.jar (JDBC 5.1.40 connector)
5. rs2xml.jar (third party library, which transfer from ResultSet to TableModel)
6. resource folder contains mysql-connector-java-5.1.41-bin.jar and rs2xml.jar

**Setup to run the application**

1. Install Java Swing from <http://download.eclipse.org/windowbuilder/WB/release/4.6/>
2. Set Classpath Variables and define user libraries for mysql-connector-java-5.1.41-bin.jar and rs2xml.jar under Window/Preferences.
3. Add the above libraries to the Shop Management System Project.
4. Open mysql –uroot –p console and create database and tables for the system, see the SQL script file under resource folder. Sample testing data also included in the script file.
5. Open src/model/DBConnection.java and hard code the value of ***JDBC\_DRIVER, DB\_URL, USER*** and ***PASS*** according your MySQL database.
6. Run the application.

**Operate the application**

Login Screen displays when system starts. The system will prompt you to create an account if there isn’t any account in the database. If there is an account in the database, the system gives the user 3 chances to login if it is failed.

The main application consists of Sale, Checkout, Sales Today, Goods Admin and Clerk Admin tabs.

Sale tab let Clerk to select the good to buy according the customer’s needs. It also allow the user to logout or exit the system.

Checkout tab let Clerk to proceed to payment for the goods in the Sales Cart.

Sales Today tab shows the sales report based on category of Clerk or Goods. It also gives the user to see the report based on Clerk or Goods name. The report also shows the total number and amount of money based inputs.

Goods Admin tab can Add, Change, Delete, Search, Display Goods in the Database.

Clerk Admin tab can Add, Change, Delete, Search, Display Clerk account in the Database.

**Architecture of the application**

The Application apply MVC architectural pattern and Java Swing to the development.

1. The idea is the viewer registers as a listener

public void addMainViewListener(ActionListener listenForClick)

{

btnSearchClerk.addActionListener(listenForClick);

}

1. User actions invoke a registered listener method in the controller class.

this.theMainView.addMainViewListener(new MainViewListener());

1. The controller accesses the model and updating it in a way appropriate to the user's action.

class MainViewListener implements ActionListener{

public void actionPerformed(ActionEvent e)

{

if(e.getActionCommand().contains("Search"))

{

theDBAccess.dailySales(theMainView);

}

}

}

**Issues needs to address**

1. Detect the changes in the Model and broadcast them to the Controller. The controller pass the changes to the View.
2. Add the transaction concepts for accessing database.
3. Use thread safe codes for 7accessing database.

**Enhancements to implement in the future.**

1. Migrate local mysql database to RDS mysql database from Amazon Web Service remotely.

Create an Amazon Relational Database Service MySQL DB instance at Amazon Web Service. You can type the command at a command prompt on a client computer to connect to the shopsystem database using the MySQL monitor, in my case:

mysql -h sms.ccsde8goiljn.ap-southeast-2.rds.amazonaws.com -P 3306 -u root -p

Here is the usefully commands to access the database, please note table name is case sensitive:

USE SHOPSYSTEM;

SHOW TABLES;

SELECT \* FROM SALESMAN;

SELECT \* FROM GOODS;

SELECT \* FROM GSALES;

1. Create an instance in the AWS cloud and migrate my application to the AWS and deploy it on Elastic Beanstalk.

Create a Window’s Amazon Elastic Compute Cloud (EC2) instance at Amazon Web Service. You can connect to the Windows instance using a remote desktop client of your choice, such as a remote desktop connection from local window’s OS machine. Here are connection details in my case:

Public DNS:    ec2-54-206-125-207.ap-southeast-2.compute.amazonaws.com

Username:    Administrator

Password:    P.yKlTQvDP8W&$6IFIta7yznugoUfm?8

Copy sms.jar executable file to the Window’s Amazon Elastic Compute Cloud (EC2)instance desktop. You need to installed jdk-8u121-windows-x64.exe so that you can double click sms.jar to run the shop management system. The application will access the Amazon Relational Database ServiceMySQL DB instance as its database.