Automated Order Management System for … “City Mart”

T.M.Savindu Pasingtha NG/HNDCSE/24/11

Assignment Cover Sheet

Qualification		Module Number and Title
Higher National Diploma in Computing & Software Engineering		Introduction to OOP- SEC4207
Student Name & No.		**Assessor**
T.M.Savindu Pasingtha NG/HNDCSE/24/11		Mr. Sanaka Perera
Hand out date		**Submission Date**
21/11/2020		17/10/2020
Assessment type	**Duration/Length of Assessment Type**	**Weighting of Assessment**
Coursework	Report and Demo	100%

Learner declaration
I am T.M.Savindu Pasingtha NG/HNDCSE/24/11, certify that the work submitted for this assignment is my own and research sources are fully acknowledged.

Marks Awarded			
First assessor			
IV marks			
Agreed grade			
Signature of the assessor		Date	

FEEDBACK FORM

INTERNATIONAL COLLEGE OF BUSINESS & TECHNOLOGY

Module/Title:

Student: T.M. Savindu Pasingtha

Assessor:

Assignment:

Strong features of your work:

Areas for improvement:

Marks Awarded:

ACKNOLEDGMENT

In preparation of my assignment, I had to take the help and guidance of some respected persons, who deserve my deepest gratitude. As the completion of this assignment gave me much pleasure, I would like to show my gratitude **Mr. Sanaka Perera**, our lecturer, on ICBT Campus for giving me good guidelines for assignment throughout numerous consultations. I would also like to expand my gratitude to all those who have directly and indirectly guided me in making this assignment.

ABSTRACT

This assignment comes under Object Oriented Programming skill for module at HNDCSE program. This assignment mainly focus, “CITY MART” wholesaler to develop a new Automated Order Management System. This system very help full to improve their business.

To completely this assignment firstly I discussed about what data is and what the java programming language object oriented concepts and it means explain with proper examples. So I clearly identify where to we can use OOP concept in this system project. In the second task I provide proper UML diagram for the above mentioned requirements. Its include User case diagram, Class diagram and Sequence diagram. Third task I develop the system for “CITY MART” based on the UML designs. Not only that I provide source code and interface screenshots with proper explanation.

In fourth task I mentioned proper testing technics, and after I tested this system. So below I included proper test plan and proper test cases. Finally I discussed what is the weakness and strength in this system. Not only that I included proper references and GANTT chats to this assignment.

INTRODUCTION

This assignment mainly focused to create Automated Order Management System for "CITY MART" using Java programing language. So, object oriented concepts use to build this system. Not only has that how to UML diagrammed techniques using build proper system discuss in this assignment.

So I included this system to, multi user login system and bellow I mention main three **actors** activities in the system.

- **Manager** – Creating and managing user accounts, view stock details, view sales details, view suppliers, view request and approve requests.
- **Stock Keeper** – Create and manage items, view stock details and create purchase requests.
- **Cashier/Accountant** – Generate purchase order.

That all functions are properly work in the system. I used attractive user interfaces to implement this system. MYSQL Database provided the all backend database tasks in XAMPP server. There for any one can quickly data management task in the system such as logging, registering, purchases order and others.

I think this system can easy to use and improve their business every day.

Table of Contents

TABLE OF FIGURE

TASK 1

What is the OOP Concept?

Object oriented programming is a computer programming model that organizes software design around data or objects rather than functions and logic. (Rouse, 2020) In this part we can identify major topics such as object, classes, Inheritance, Polymorphism, Abstract and Encapsulation. When we learn OOP concepts, these concepts depend on below terms. Such as coupling, cohesion, association and composition. Now we consider what the OOP concepts is one by one by one and describe bellow.

Object

Any entity that has state and behavior is known as object. Such chair, car, pen, keyboard and others. Object can be create as instance of class.

```
13  import oop.LoginClass;
14
    public class Loginpage extends javax.swing.JFrame{
```

```
// LoginClass Instance create
LoginClass log=new LoginClass();
            //instance using access login class variables /methods
            log.setUname(loginusenameTextField.getText());
            log.setPswd(loginPasswordField1.getText());
            log.ManagerCashierStoreLog(logtypecombo);
```

Java data types Data types

We can identify main two type of Data types.

Primitive data types specifies the size and type of variables values, and it has no additional methods. Such as, Boolean, char, byte, short, Int, long, float and double. Not only that, class, interface, arrays, strings and other belong to **Non-primitive** data types. (Anon., 2020)

Access modifiers

Access modifiers in java specific the accessibility or scope of a field, method, constructor or class.so, we can change the access level of fields, constructors, methods and class by applying the

modifier on it. We can see main four type of access modifiers. Such as, **public**, **private**, **protected** and **default**. (Anon., 2011)

Let's understand the access modifiers in Java by a simple table.

Access Modifier	within class	within package	outside package by subclass only	outside package
Private	Y	N	N	N
Default	Y	Y	N	N
Protected	Y	Y	Y	N
Public	Y	Y	Y	Y

Figure 1-Access modifiers

Not only that we can identify Non-access-Modifier as a **static** key word. (Anon., 2011)

Return data type

Most time return data type use when we implement the method or function. Return data type used to identify the returning value data type (void mean is no return value and other data type's int-string-double-float and others or some class object -function.) in the method.

Methods or functions

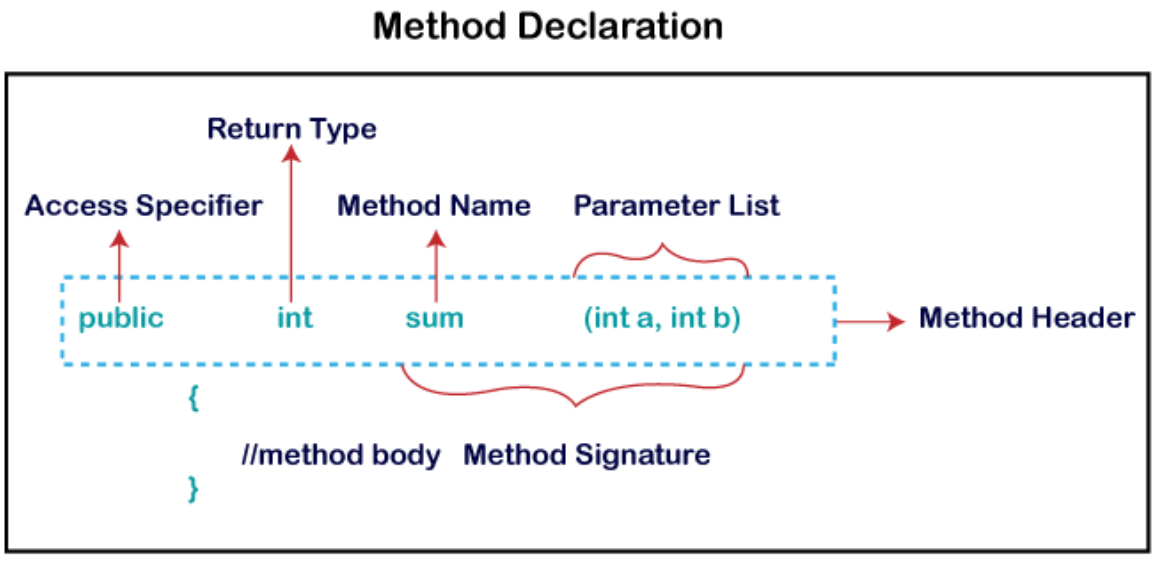


Figure 2-Access Modifires

Especially the method haven't a return value, so we should use **Void** key word as a return data type.

Class

A class is an extensible program-code-template or blueprint or prototype for creating objects. Class have object's behaviors or states and Methods or fields.

Access Modifier class name of the class {// your body of code here}

```
13  import oop.LoginClass;
14
    public class Loginpage extends javax.swing.JFrame{
```

Constructor method

Constructor method is a same class name to same method name. Constructor method to have no return types. Constructor methods run time compiling. Not only that java constructor cannot be abstract, static, final and synchronized. (Anon., 2020)

Class constructor { constructor ( ) {} }

```
public class DatabaseConnect {
        //constructor method same class name
    public void DatabaseConnect() throws SQLException{
        try {
            conn = (Connection) DriverManager.getConnection(urldb, usernamedb, passworddb);
            System.out.println(" XAMPP SERVER MYSQL DATABASE CONNCTED. ");

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

Inheritance

Inheritance in java is a mechanism in which one object requires all the properties and behaviors of a parent object. So, when it's using code reusability and when we use run time polymorphism we can use methods overriding. In the OOP have main five type of inheritance, Such as single, Multilevel, hierarchical, hybrid, and multiple inheritance. But in the java programming not supported to multilevel inheritance. So its problem avoided using the **interfaces** concept. It's like as a java class. "**Extends"** key word use to inherit the some sub class to parent class. When we use interfaces inheritance to use **Implemen**t key word.

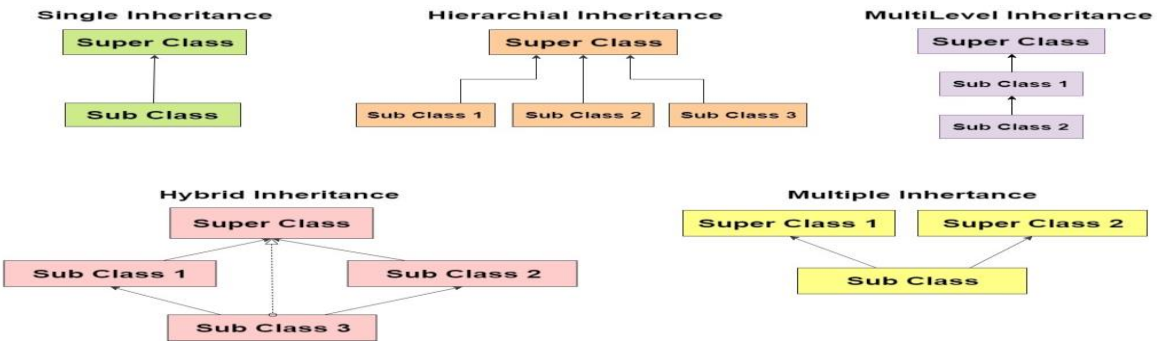


Figure 3-Inheritance types

```
21
22      public class User_Accounts extends javax.swing.JFrame {
23
```

```
package oop;

public class Manager extends DatabaseConnect{
```

Polymorphism

Polymorphism in java is a concept by which we can perform a single action in different ways. (Many forms). We can see two types of polymorphism such as statics and dynamic polymorphism. Compile time polymorphism to tell **static**. So, this concept using **Method Overloading**.

In the run time polymorphism to tell **dynamic**, in this concept use **Method Overriding**.

❖ Method Overloading

Create a same name with multiple methods in same class body. So, do this should all the methods return data type should same and all methods should have difference parameters. This concept to called method overloading. (Anon., 2020)

❖ Method Overriding

When inherited class methods change in the child class body using creating new parent class method name to same method. Not only should that it be same parameters in the method. (Anon., 2020)

- You should know private, statics, final and final class methods cannot be an override.

```java
import com.mysql.jdbc.Connection;
import com.mysql.jdbc.PreparedStatement;
import com.mysql.jdbc.ResultSetMetaData;
import java.sql.ResultSet;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.util.Vector;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.table.DefaultTableModel;

public class DatabaseConnect {
    //DB Connection path/username/password
    public static String urldb="jdbc:mysql://localhost:3306/city_mart";
    public static String usernamedb="root";
    public static String passworddb="";
    //DB Connection classes
    public Connection conn=null;
    public PreparedStatement insert;
    public ResultSet rs= null;
    public DefaultTableModel model;
    public ResultSetMetaData rss;
     //sql query variables
    public String selectquery;
    public String deletequery;
    public String updatequery;
    public String addquery;
```

```java
import java.awt.HeadlessException;
import java.sql.SQLException;
import java.util.Vector;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;



public class SupplierClass extends DatabaseConnect{
        public String s_id,s_name,s_food,s_add,s_contact;
        public String dq="DELETE FROM suplieraccounttable  WHERE s_id=?";
```

```
    public void delete(){
131     try{
            int dialogBoxResult=JOptionPane.showConfirmDialog(null,"Do you want DELETE this Row-Record.. ?",
                    "Warning",JOptionPane.YES_NO_OPTION);
134         if(dialogBoxResult == JOptionPane.YES_OPTION){
135             /*  1-  connect database connection */
136             super.DatabaseConnect();
137             super.deletequery=dq;
138            // deletequery="DELETE FROM suplieraccounttable  WHERE s_id=?";
139             super.delete();  /*insert=(PreparedStatement) conn.prepareStatement(deletequery);*/
140             int id = Integer.parseInt(s_id);
141             insert.setInt(1,id);
142             /* 3-  Executequery Now  */
143             super.queryexeute();
144             JOptionPane.showMessageDialog(null,"ID "+id+" Deleted" );
145         } else {System.out.println("Thank you");}
146     }catch (SQLException ex) {
147          JOptionPane.showMessageDialog(null,ex);
148        Logger.getLogger(Cashier.class.getName()).log(Level.SEVERE, null, ex);
149        System.out.println(ex.getMessage()+"THis is ERROR -- ");
150     }catch (HeadlessException x){
151        JOptionPane.showMessageDialog(null,x);
152        System.out.println("THis is ERROR -- "+x);
153     }finally{
154       /* 4-  close the connection  */
155       super.Connectionclose();
156      }
```

```
public class Storemanager extends javax.swing.JFrame {
```

```
private void deleteitemTextField5MouseClicked(java.awt.event.MouseEvent evt) {
   try{
   //okedi kale Cashier ge delete query ek pass krl Supplier object ekt ethnin Cashier purchase order ek execute kala
   SupplierClass s=new SupplierClass();
   s.s_id=itemidtf.getText();
   s.dq="DELETE FROM store  WHERE id=?";
  // int storedid = Integer.parseInt(itemidtf.getText());
   s.delete();
   }catch(NumberFormatException e){ JOptionPane.showMessageDialog(this, e); }finally{cleartxt();}
}
```

```
21
22      public class User_Accounts extends javax.swing.JFrame {
23
```

```
904     private void r_deleteButtonActionPerformed(java.awt.event.ActionEvent evt) {
905         try{
906         //okedi kale Cashier ge delete query ek pass krl Supplier object ekt ethnin Cashier p
907         SupplierClass s=new SupplierClass();
908         s.s_id=r_idTextField.getText();
909         s.dq="DELETE FROM user_accountstable  WHERE id=?";
910        // int storedid = Integer.parseInt(itemidtf.getText());
911         s.delete();
912         }catch(NumberFormatException e){ JOptionPane.showMessageDialog(this, e);
913         }finally{clearedText();}
914
915     }
```

Casting

- ***Up casting***

When there are inherited parent class object put to sub class object, this concept to call up casting. Therefore when we can access the sup class methods and variable using calling through the parent class object.

- ***Down casting***

When there are inherited sub/child class object put to the parent class object, this concept to call down casting. Therefore when we can access the parent class methods and variable using calling through the sub class object.

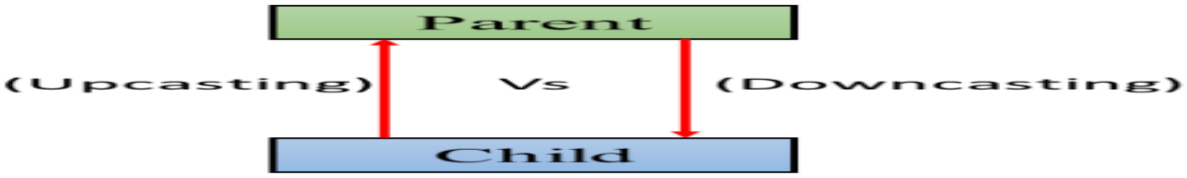


Figure 4-Casting

Abstract

Abstract mean is we cannot write codes. So, some time we can will be identify some methods and class we cannot write codes. So, its class and methods to us call as abstract class or abstract methods. So, when we implement the abstract class or method firstly we should the put "**abstract**"

Keyword before the class keyword or method name.

- You should know, abstract class cannot generate an objects in this class using. Because we cannot generate constructor method in the abstract class. But we can inherited that abstract class and after we can access the sub class through the parent (abstract) class methods and variables. Not only that we can override the other methods in the abstract class.
- Abstract methods access to have another way is that all inherited class should be an abstract. So this task to do have another way. It is an **interfaces** concept. Interface like as a classes. (Anon., 2020)

Encapsulation

Encapsulation in java is a mechanism of wrapping the data variables and code acting on the data (method) together as a single unit. In encapsulation, the variables of a class will be hidden from

other classes, and can be accessed only through the methods of their current class. Therefore, it is also known as **data hiding**.

- Firstly we declare the **variables** with **private access modifier**. After generate **getters** and **setters** with **public** access modifier to, put values in to variable and get the value in inside the variable.

```
import All_Frames.Loginpage;
import All_Frames.Managerhome;
import All_Frames.Storekeep;
import All_Frames.cashier;
import java.awt.HeadlessException;
import java.sql.SQLException;
public class LoginClass extends DatabaseConnect{
    private String uname, pswd,w;

    //ENCAPSULATION - getters/setters create
    public String getUname() {
        return uname;
    }
    public void setUname(String uname) {
        this.uname = uname;
    }
    public String getPswd() {
        return pswd;
    }
    public void setPswd(String pswd) {
        this.pswd = pswd;
    }
    public String getW() {
        return w;
    }
    public void setW(String w) {
        this.w = w;
    }
```

```
import oop.LoginClass;

public class Loginpage extends javax.swing.JFrame{
```

```
    private void logbtnActionPerformed(java.awt.event.ActionEvent evt) {
        String logtypecombo=logComboBox.getSelectedItem().toString();
        LoginClass log=new LoginClass();
                    log.setUname(loginusenameTextField.getText());
                    log.setPswd(loginPasswordField1.getText());
                    log.ManagerCashierStoreLog(logtypecombo);
```

“This” Key word

This key word use to access current class variables and methods.

Ex- this. Variable; this. Method ();

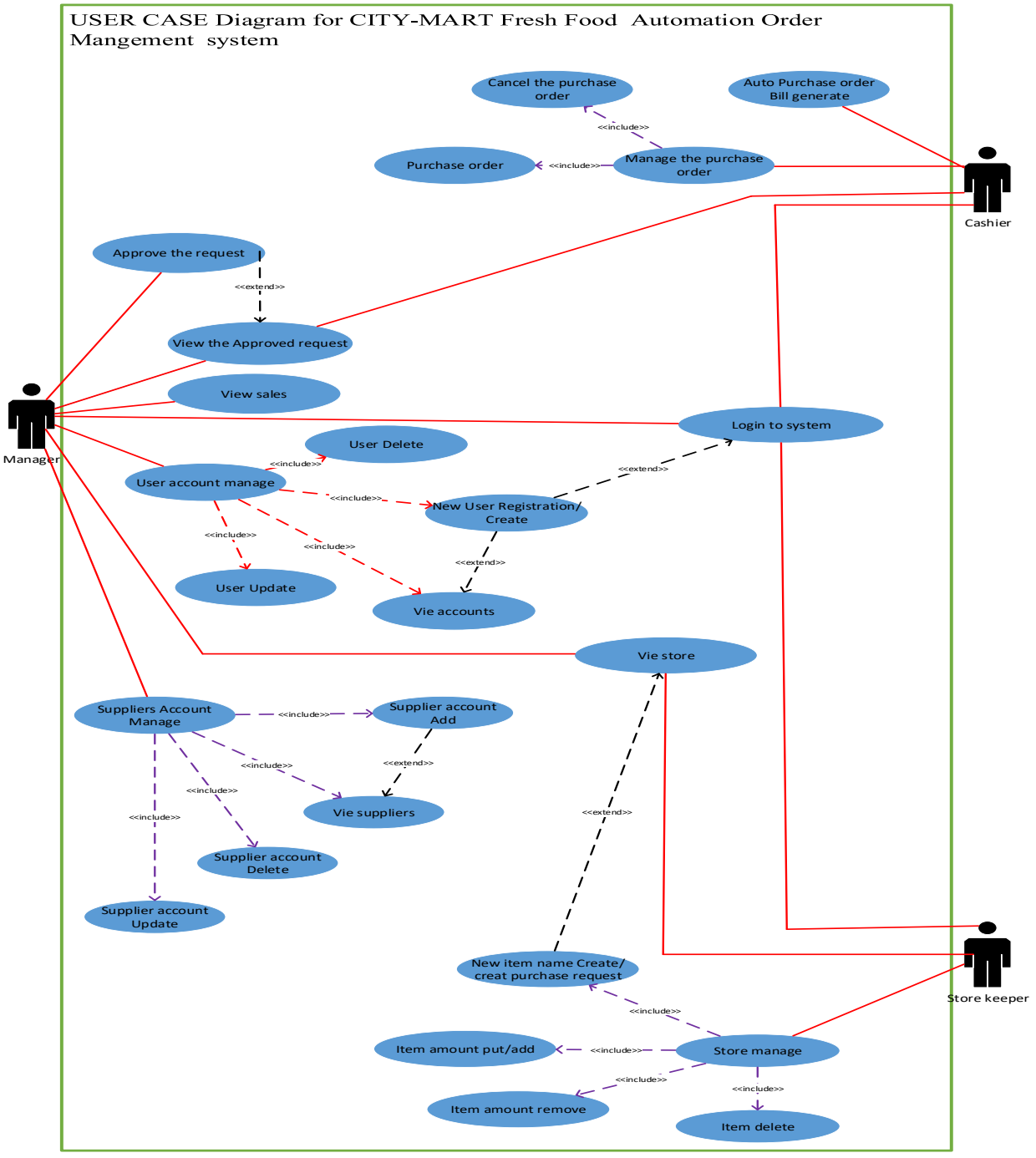
“Super” Key word

Super key word use to access Parent or super class variables and methods.

Ex- super.variable; super.Methods ();

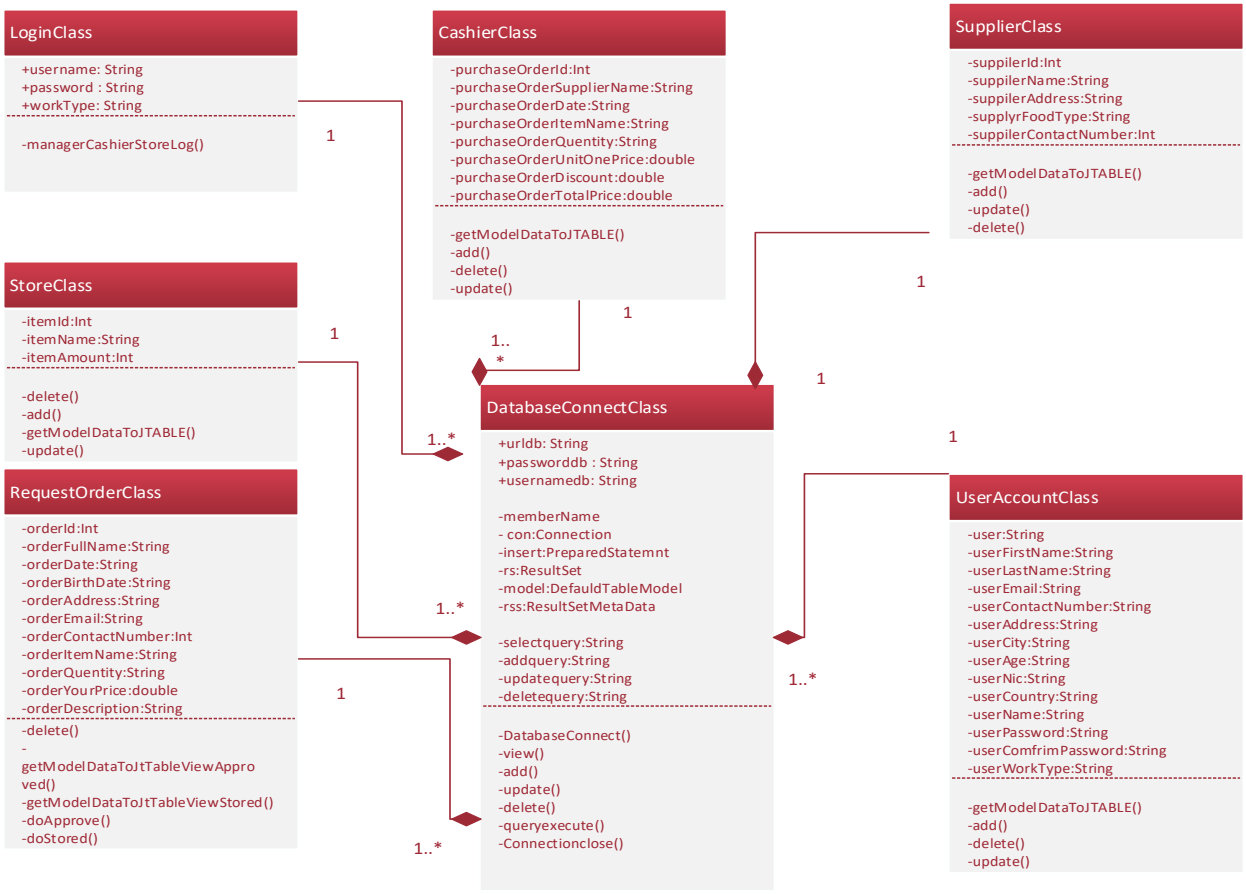
TASK 2

User Case Diagram



In this diagram I show all the this automation order system functions and who is the it's do. So I identified main three actors as "manager, cashier and supplier" and their tasks.

Class Diagram for City Mart Automation Order System



Assumptions

- I thought all DatabseConnectionClass with other all the class have “COMPOSITION” relationship.
- Not only that I thought DatabseConnectionClass multiplicity have “one to many” with other classes.
- DatabseConnectionClass have Add, delete, update methods inherit and override to other all the classes. Because I could identify that methods common to all the classes. Not only have that in the Java between normal classes can’t multiple inheritance.
- GetModelDataToJTable methods to use return the DefaultTableModel class object. So this objects after the cached in design pages “jTable” to send all the model data to view rows and Colums in the Database Table.

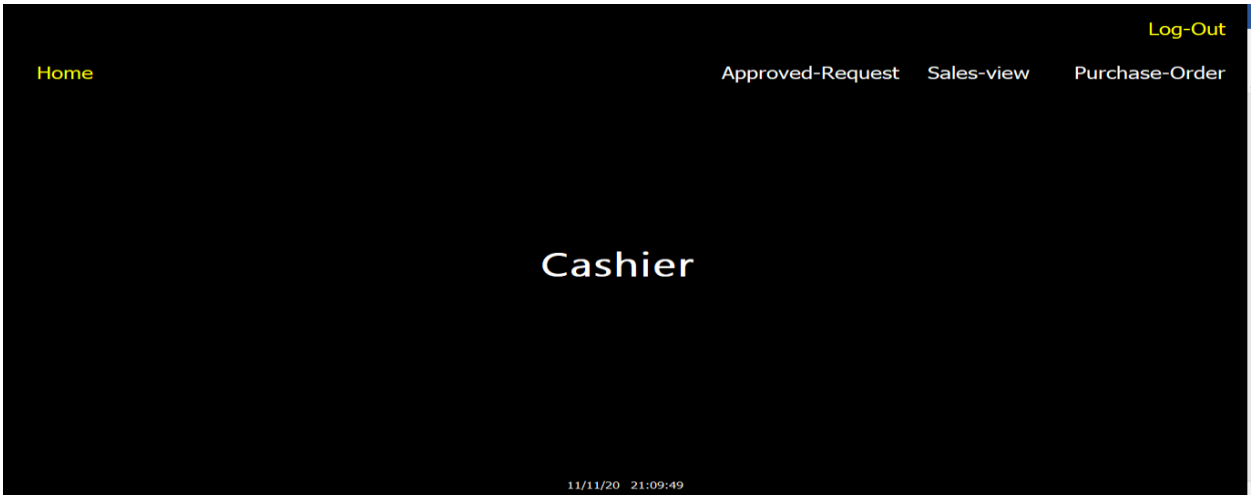
TASK 3

Home Page

This is a Home page or Landing window page of System. This page have Home, Manager, Cashier, Stock-keeper buttons. It's to use me all the label to set mouse click events to like as to work buttons.

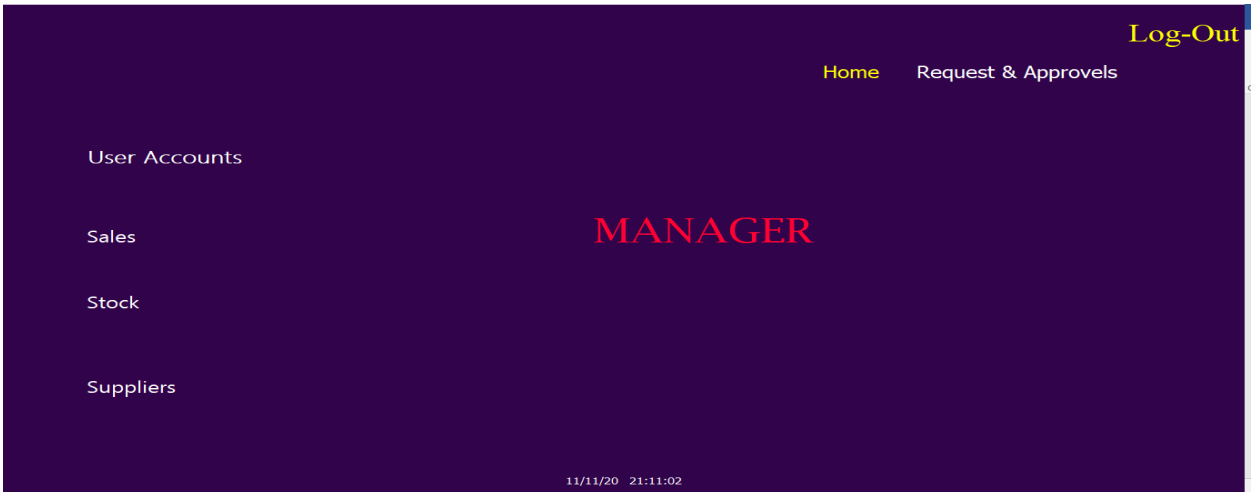
When clicked the buttons we carry to the login page. If login success open new windows page as a Manager, Cashier, Stock-keeper home pages. Middle of the page have label "city mart fresh food", so we can mouse move through it then change it font color.

Cashier Page



Cashier can view the Approve-request, sales view and purchase-order. I think when purchasing order cashier to helpful to check this order is approve or not. Then he or she can purchase the order within bill.

Manager page



Manager can approve the order, user accounts manage, Suppliers mange, sales and stock view access. .

Store-Keeper page

Stock–keeper can create request and he can stack manage.

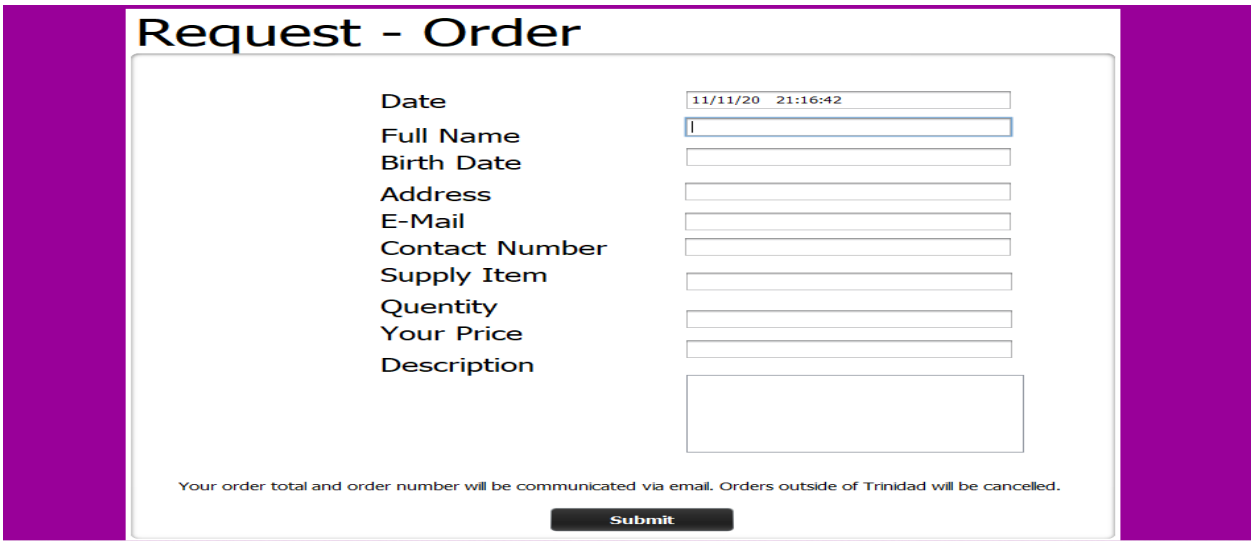
Login page

Login page have multi user login facility. When some user want to login to the system first he should enter he or she username, password and about what is her work position. When click radio button we can hide user name and password. Finally, clicked the "ok" button, its true the login details open the new work space windows. (Go to cashier/manager/ stock-keeper home pages). Else, refresh the login page.

User Account Registration Page

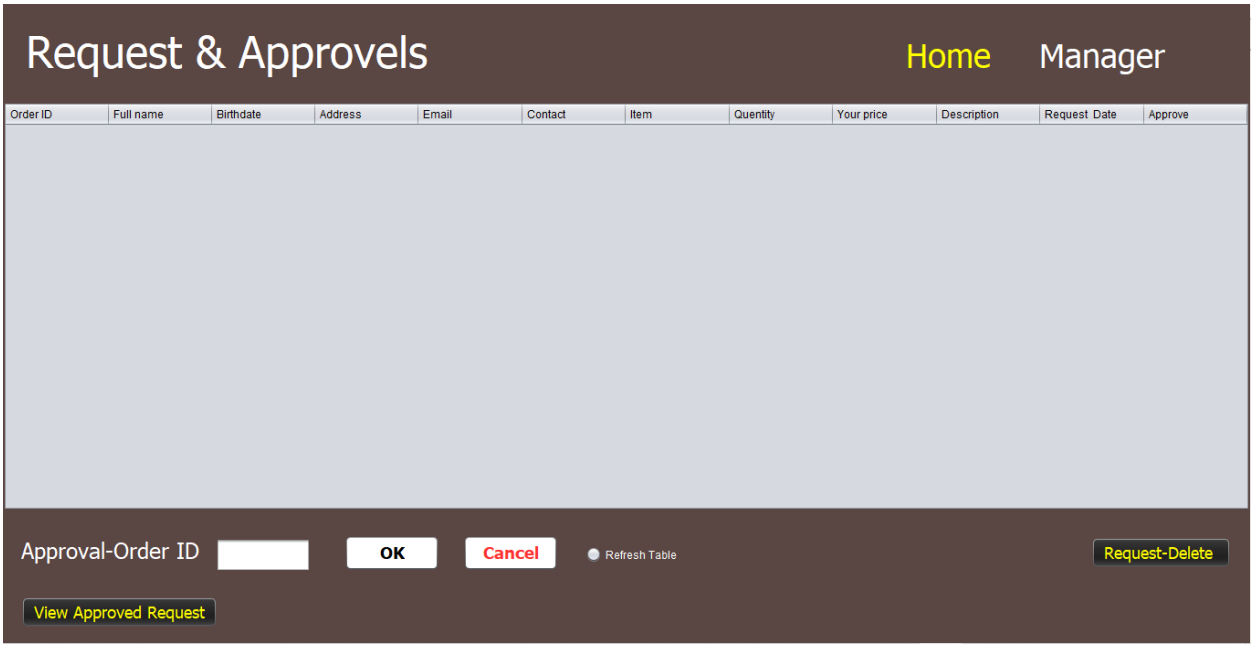
In the user account page, manager can add, delete, updates the users to system. Such ass manager, cashier or store-keeper. In this time all the users' username and password generated to login the system. Additionally "show-Table and Hide-Table" button using manager can control the table data.

Request-Order page



This part doing stock keeper. In this page some supplier can requites their orders after filling the form.

Request View Page

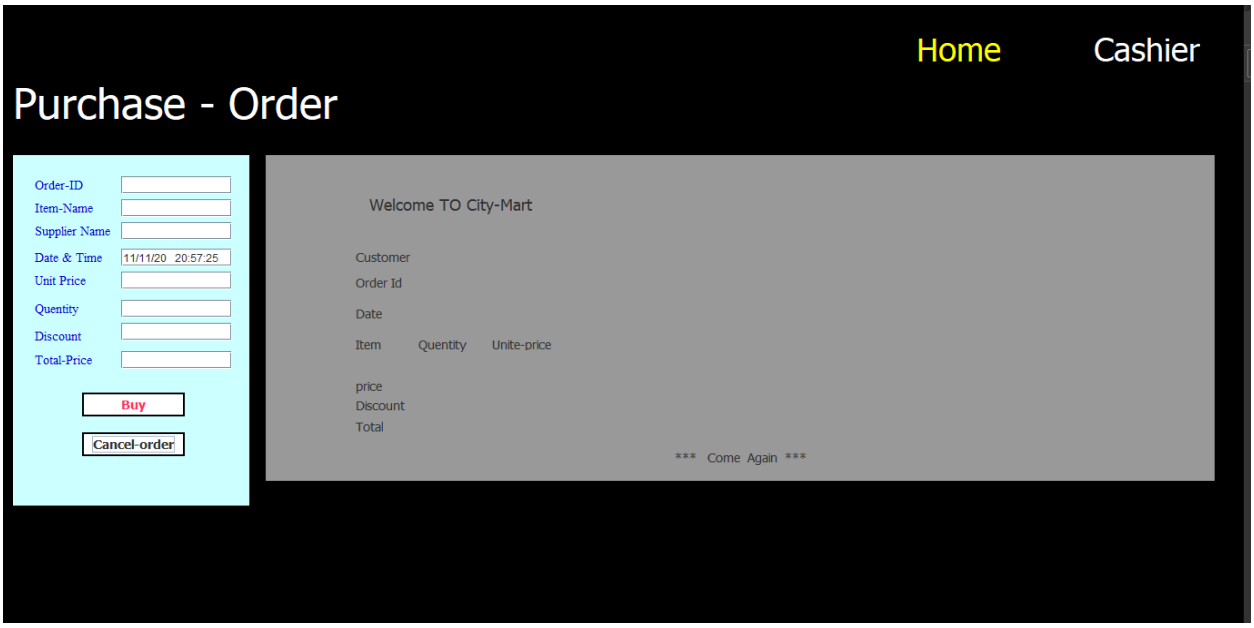


This part doing in Manger. In this page he can show requests and then he or she can Approved or not and delete the requests. Not only can that manager view the only approved request clicking “view approved request” button.

Approved-Request View Page

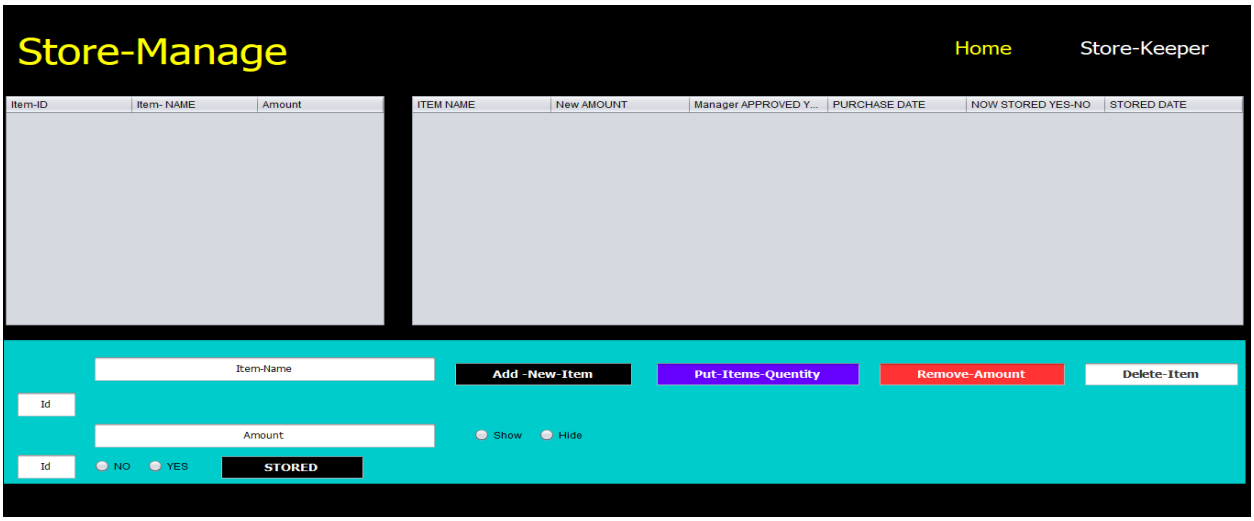
In this page provide some specific details with view the only approved request to stock keeper and cashier. So this details want to when purchasing order and when store manage.

Purchase-Order Page



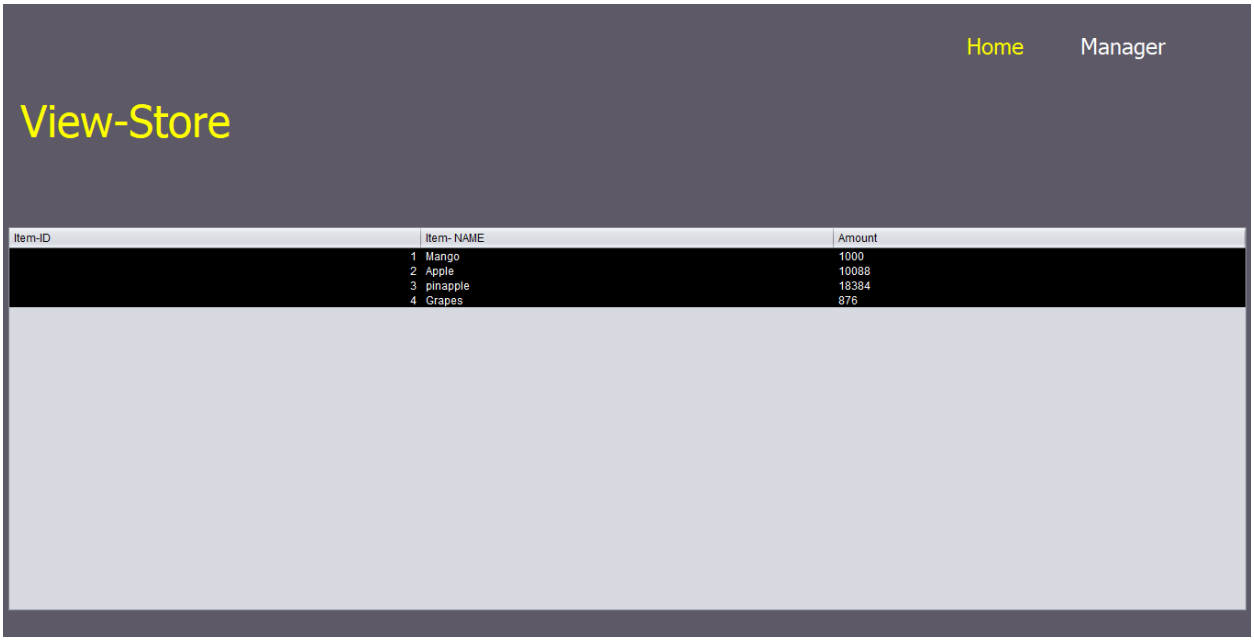
This page only can access cashier. He or she can purchased the approved orders and issue the bill. So cashier should enter above information.

Store-Manager Page



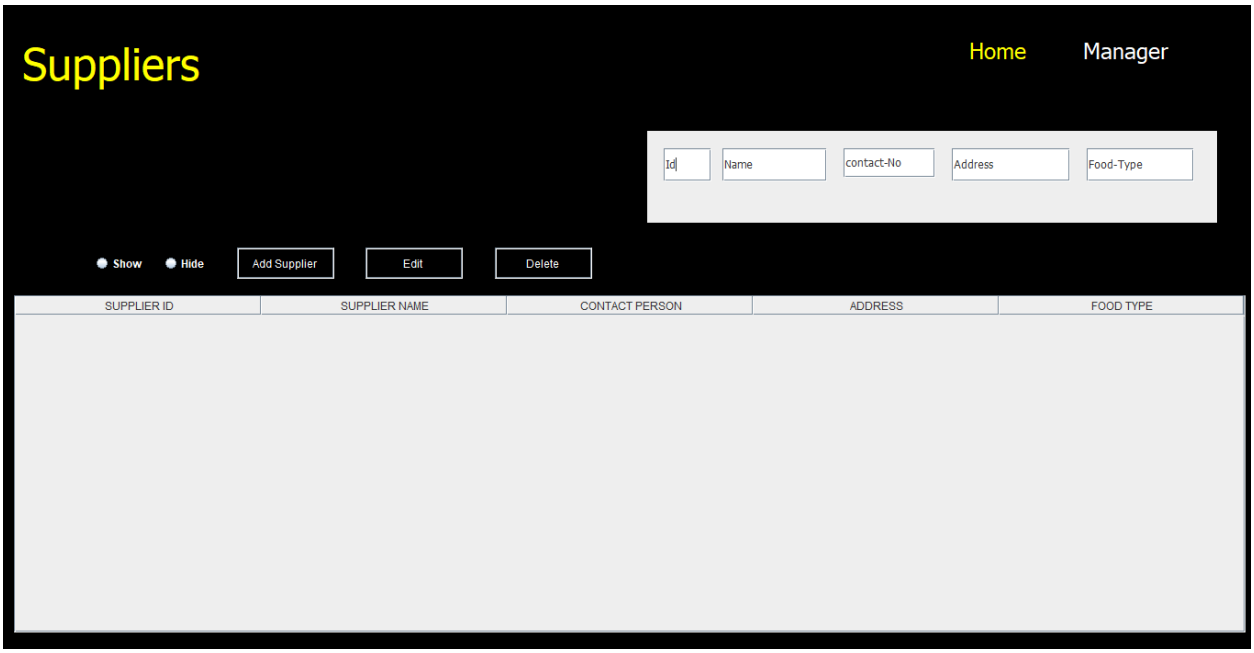
All the approved orders can storing to the data base. Right side table view the app the approved type yes orders details and left side table using view all the inserted store item data. This page provide new item add, that item update or delete , that item amount only add or remove facility and store keeper can order item after the store that order update as a stored “yes or no”.

Store-View



This page can access manager to view how about the these days item store.

Supplier-Mange Page



This page provide to system to add about who is the over company suppliers details. Not only that manager can add, delete and update facilities to handle the supplier database table.

TASK 4

TEST PLAN

Test case number		Test case name	Steps	Expected result
TC 2		Login Testing		
	TC2.1	Check Login with Valid Data	Input valid work type, Username and password	"..Login Successfully.." message view
	TC2.2	Check Login with invalid data	Input invalid work type, Username and password	"Incorrect User name or Password" message view and then sql exception error message view
TC1		User registration Testing		
	TC1.1	Check Data Add to user registration	Entering the data in all the text fields then add button click.	"Added successfully" message view
	TC1.2	Check User account registration data update	Click the table row or then enter new values to text fields.	"Update successfully" message view
	TC1.3	Check User account registration data Delete	Click the table row or then enter ID value to text fields.	View Message Dialog box with Yes No option and then view "Deleted successfully" message view
TC6		Store-Manage testing		
	TC6.1	Check store-item entry.	Entering the data in all the text fields then add button click.	"ID 4 to Add Grapes 876" like as message view.

	TC6.2	Check store-item update	New data input to change old data in the record	"ID 3 Pineapple Updated" like as message view
	TC6.3	Check store-item id delete	Click the table row or then enter ID value to text field.	"Do you want to delete this row record?" message and view "ID 4 Deleted" like as message
	TC6.4	Check store-item View page	Click view button	Open new window and Show the table data
TC4		Supplier Data Manage		
	TC4.1	Check the create new supplier	Entering the data in all the text fields then add button click.	"Added successfully" message view
	TC4.2	Check the supplier data update	Table row click or after new data insert to text fields.	View "Updated Successfully" message
	TC4.3	Check the supplier data delete	Click the table row or then enter ID value to text field.	"Do you want to delete this row record?" message and view "ID 4 Deleted" like as message
	TC4.4	Check the supplier data view.	Clicked the show button	Table data view.
TC3		Create purchase requisitions		
	TC3.1	Check the request-order data entering.	Entering the data in all the text fields then Submit button click.	"Added successfully" message view.
TC5	TC5.1	Generate purchase order checking	Entering the data in all the text fields then Submit button click	View the with entered data.

TC4		Approval Manage		
	TC4.1	Check the Approve “Yes”	Click the table row or enter request ID value and after click “OK” button.	View as message “ID 5 APPROVED “Yes”
	TC4.2	Check the Approve Cancel as “No”	Click the table row or enter request ID value and after click “CANCEL” button.	View as message “ID 5 APPROVED “No”
TC8		Check the Home page buttons	Clicking the all the buttons	Go to the login page

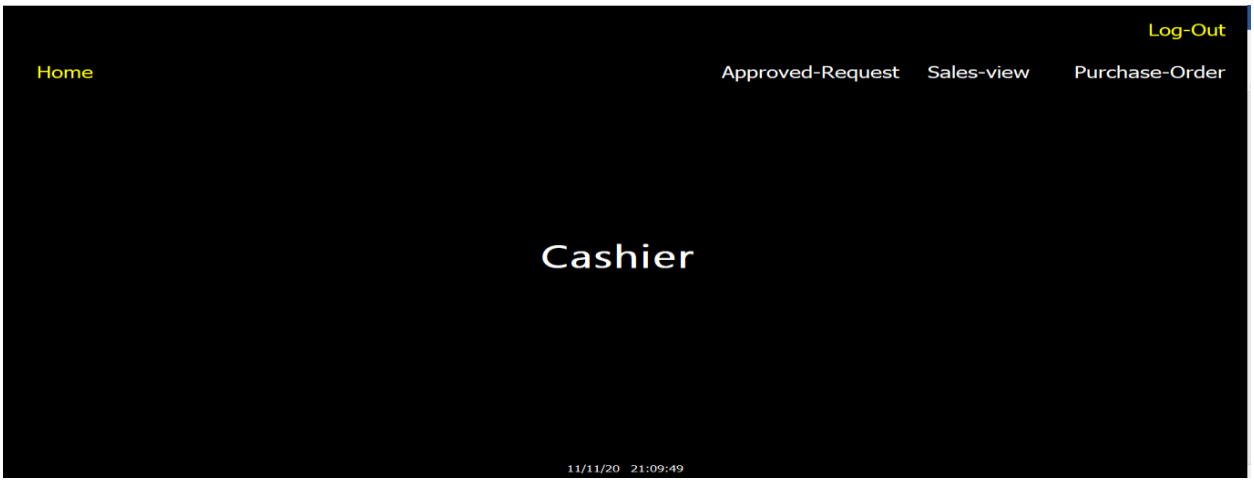
TEST CASES

1. Home Page

Test Case NO	08
TEST DATE	2020/11/11
Test OBJECTIVE	Check Home page Buttons mouse click, move events
OBJECTIVE ID	1
Test Data	i. Clicked on “Home” ii. Clicked on “Manger” iii. Clicked on “Cashier” iv. Clicked on “Stock-keeper” v. Mouse move on “City Mart Organic Fresh Food”
Expected Results	i. Refresh the Home page ii. Go to “Login Page” page iii. Go to “Login Page” page iv. Go to “Login Page” page v. Change the “City Mart Organic Fresh Food” label colour white to green
Actual Results	All Home page items accurately working in the system.
Pass/Fail	pass
Conclusion	Expected actual output.

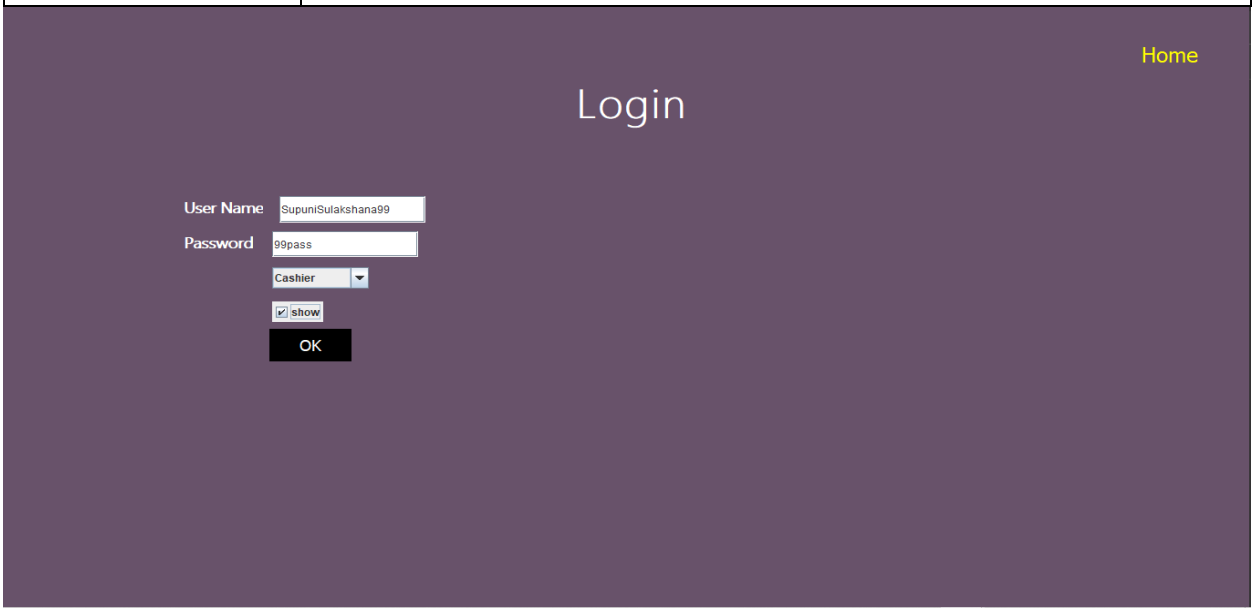

Log-Out
Home
Store-Manage
Create-Request
Store-keeper
11/11/20 21:11:57


Log-Out
Home
Request & Approvels
User Accounts
Sales
Stock
Suppliers
MANAGER
11/11/20 21:11:02


Log-Out
Home
Approved-Request
Sales-view
Purchase-Order
Cashier
11/11/20 21:09:49

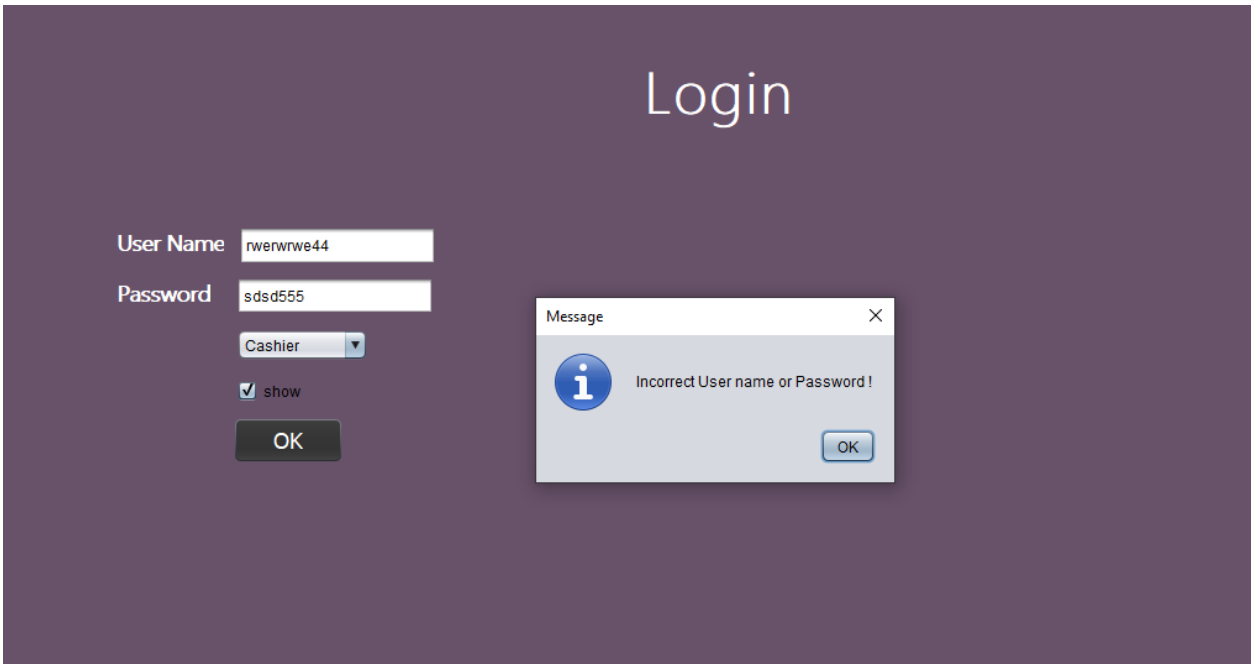
2. Login Authentication

Test Case NO	02
TEST DATE	2020/11/11
Test OBJECTIVE	Check Login Page with valid Data
OBJECTIVE ID	1
Test Data	i. User name= “supunisulakshana99” ii. Password= “99pass” iii. Select “combo box item” = “Cashier” iv. Check box = enabled v. Mouse move the “Login” label
Expected Results	Login Successful message and open “Cashier” Page
Actual Results	Login Successful message and open “Cashier” Page
Pass/Fail	pass
Conclusion	Expected actual output.



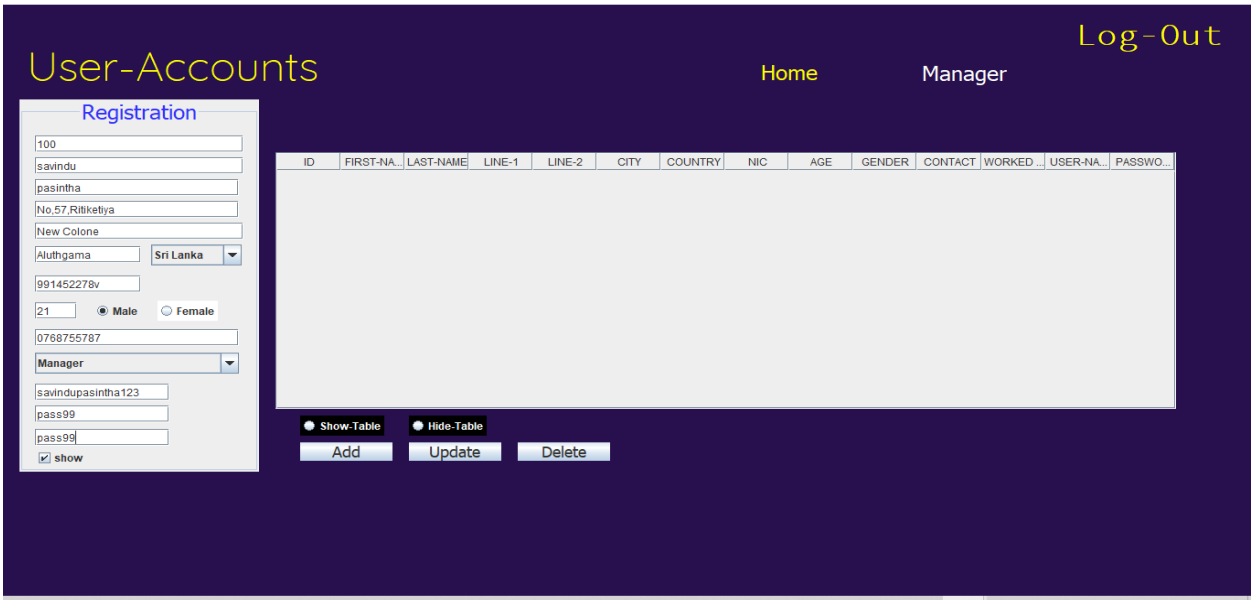
3. Login with invalid data

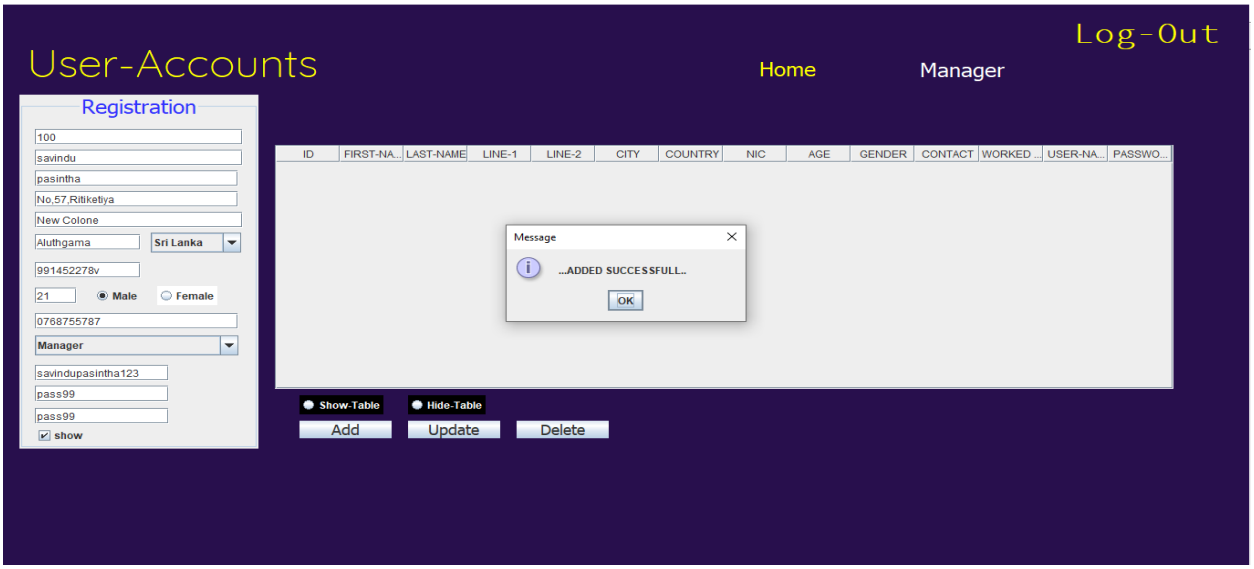
Test Case NO	02
TEST DATE	2020/11/11
Test OBJECTIVE	Check Login Page with invalid Data
OBJECTIVE ID	2
Test Data	i. User name= “rwerwrwe44” ii. Password= “sdsd555” iii. Select “combo box item” = “Cashier”
Expected Results	Invalid username or password message and Refresh “Login” Page
Actual Results	Login fail and direction again login page.
Pass/Fail	Pass
Conclusion	Expected actual output.

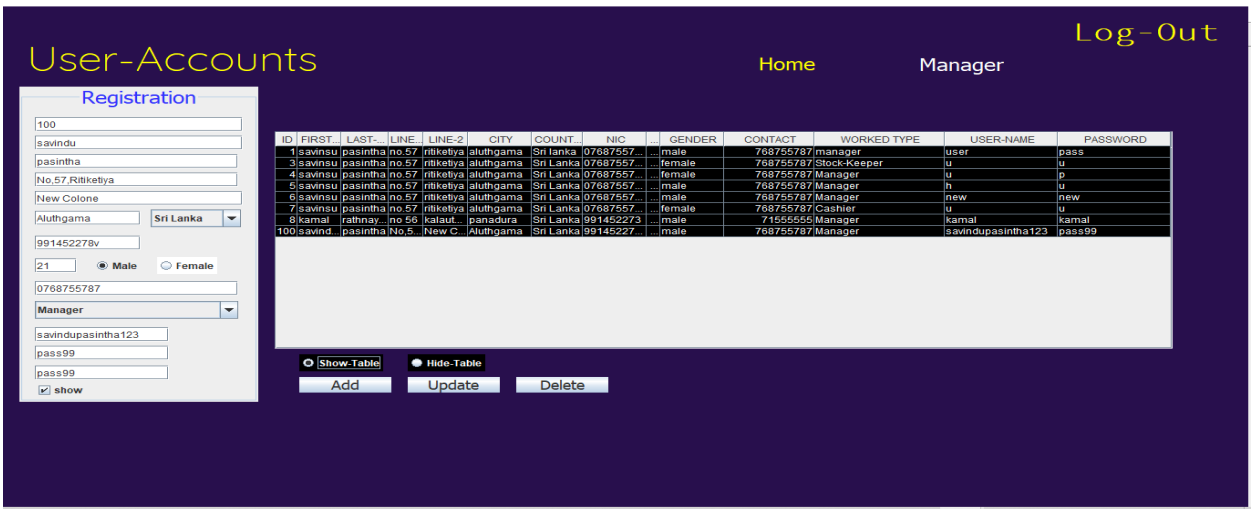


4. User Registration

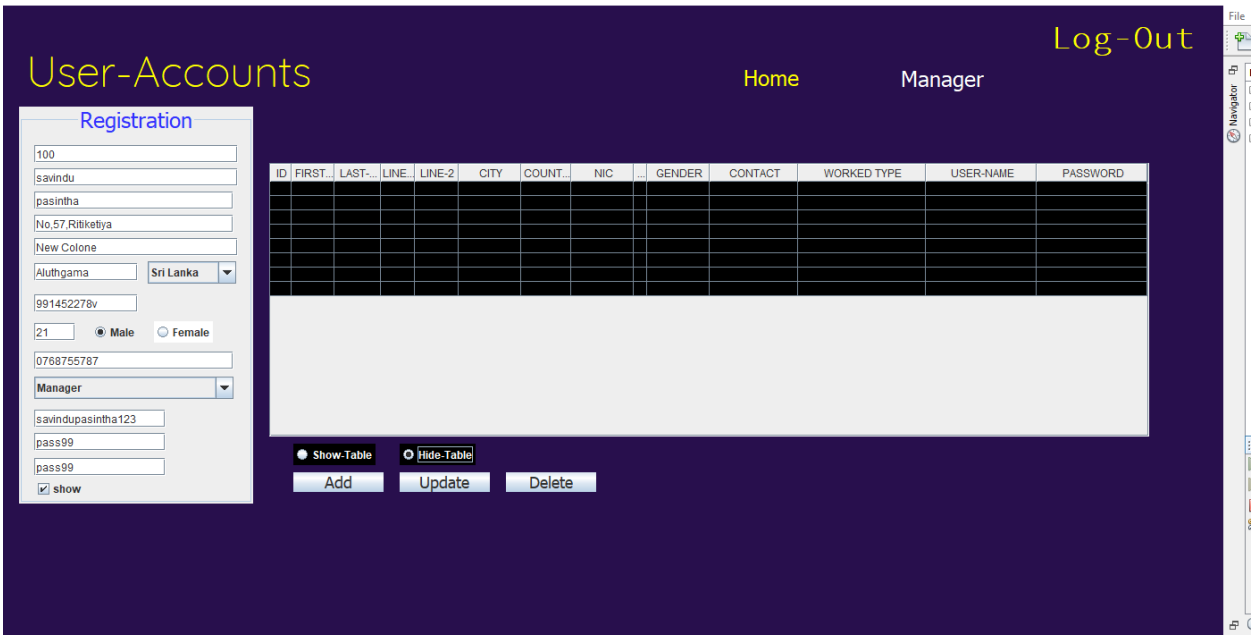
Test Case NO	01
TEST DATE	2020/11/11
Test OBJECTIVE	User Account Registration to data add.
OBJECTIVE ID	1
Test Data	i. ID=“100” ii. First-Name= “savindu” iii. Last-Name= “pasingtha” iv. Address-Line 1= “No.57,Ritiketiya” v. Address-Line 1= “New Colony” vi. City-Name= “Aluthgama” vii. Select country= “Sri Lanka” viii. NIC-Number= “991452273v” ix. Age= “21” x. Contact-No= “0768755787” xi. Select Work type = “Manager” xii. USER-NAME= “savindupasintha123” xiii. Password= “pass99” xiv. Confirm password= “pass99”
Expected Results	Added success full message view and clear the all text fields’ values as an empty. Then “SHOW TABLE” click after view the “Registration data”
Actual Results	Expected result Successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.


Log-Out
User-Accounts
Home
Manager
Registration
100
savindu
pasintha
No,57,Ritiketiya
New Colone
Aluthgama
Sri Lanka
991452278v
21
Male
Female
0768755787
Manager
savindupasintha123
pass99
pass99
show
ID
FIRST-NA...
LAST-NAME
LINE-1
LINE-2
CITY
COUNTRY
NIC
AGE
GENDER
CONTACT
WORKED ...
USER-NA...
PASSWO...
Show-Table
Hide-Table
Add
Update
Delete

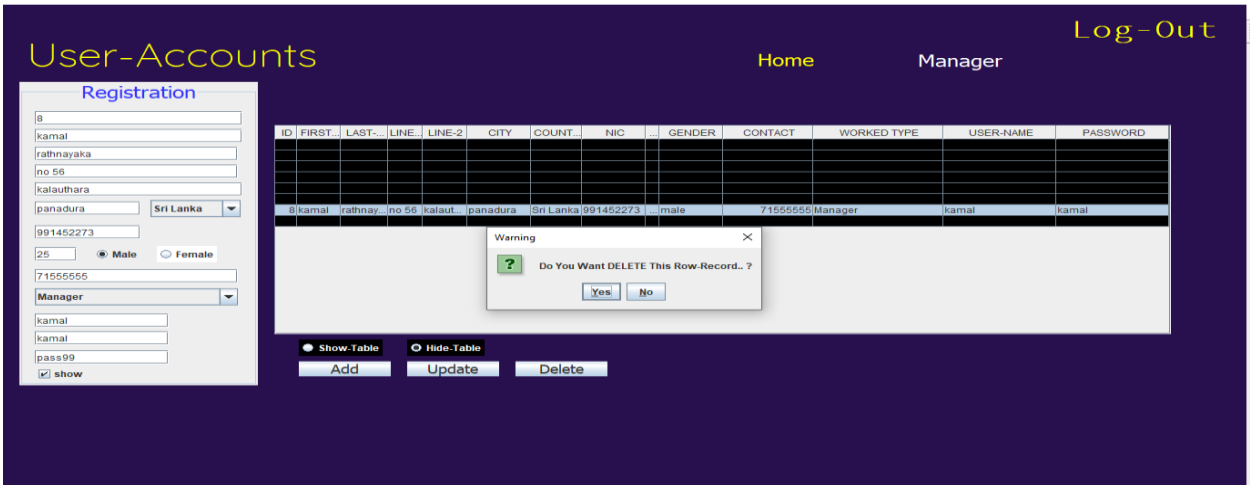

Log-Out
User-Accounts
Home
Manager
Registration
100
savindu
pasintha
No,57,Ritiketiya
New Colone
Aluthgama
Sri Lanka
991452278v
21
Male
Female
0768755787
Manager
savindupasintha123
pass99
pass99
show
ID
FIRST-NA...
LAST-NAME
LINE-1
LINE-2
CITY
COUNTRY
NIC
AGE
GENDER
CONTACT
WORKED ...
USER-NA...
PASSWO...
Message
...ADDED SUCCESSFULL..
OK
Show-Table
Hide-Table
Add
Update
Delete

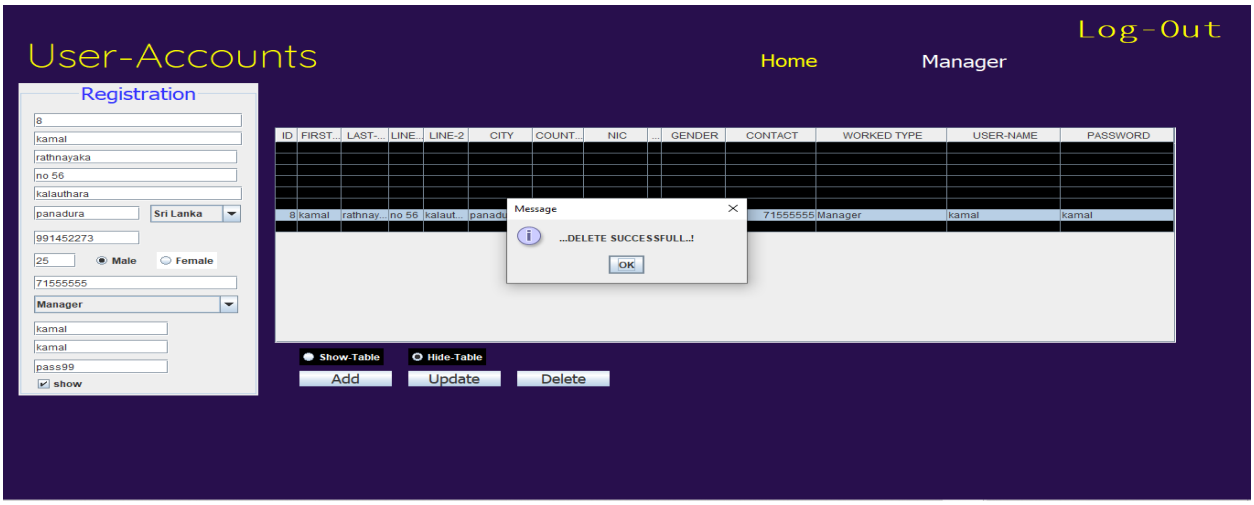


5. User Registration Delete



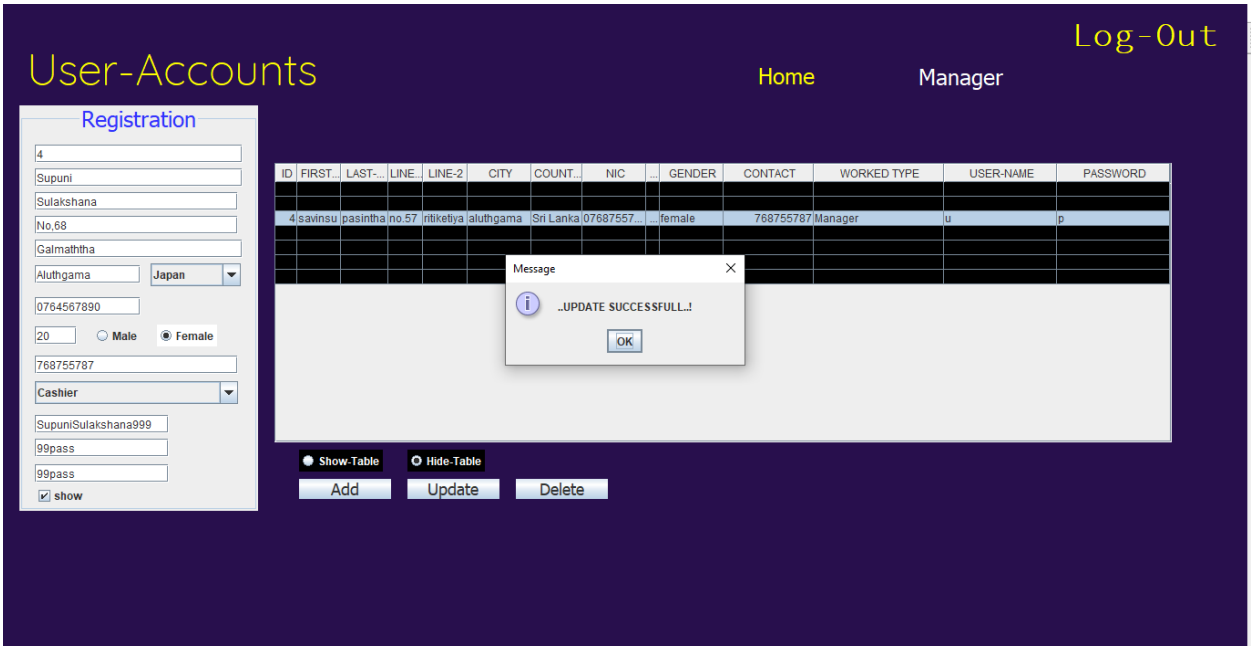
Test Case NO	01
TEST DATE	2020/11/11
Test OBJECTIVE	Check Delete the User Registration data
OBJECTIVE ID	3
Test Data	Click Table Row and Then Click Delete Button.
Expected Results	"Do you want to delete this row record?" message coming and after clicking ok then view "deleted successful" message and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.

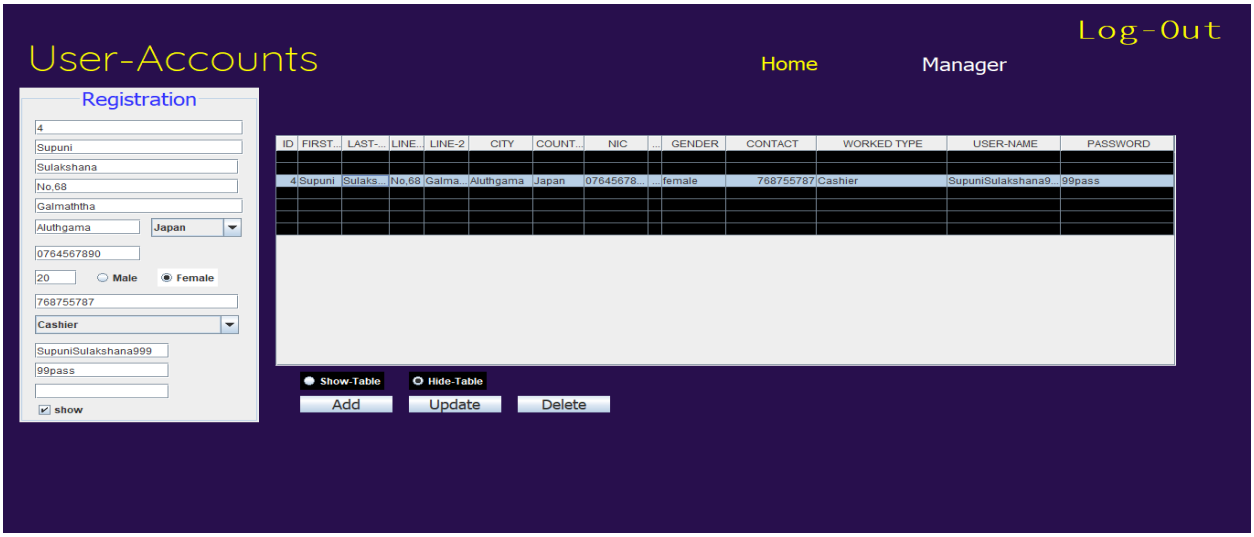



Log-Out
User-Accounts
Home
Manager
Registration
8
kamal
rathnayaka
no 56
kalauthara
panadura
Sri Lanka
991452273
25
Male
Female
71555555
Manager
kamal
kamal
pass99
show
ID
FIRST...
LAST-...
LINE...
LINE-2
CITY
COUNT...
NIC
...
GENDER
CONTACT
WORKED TYPE
USER-NAME
PASSWORD
8 kamal rathnay... no 56 kalaut... panadu 71555555 Manager kamal kamal
Message
...DELETE SUCCESSFULL..!
OK
Show-Table
Hide-Table
Add
Update
Delete

6. User Registration Update

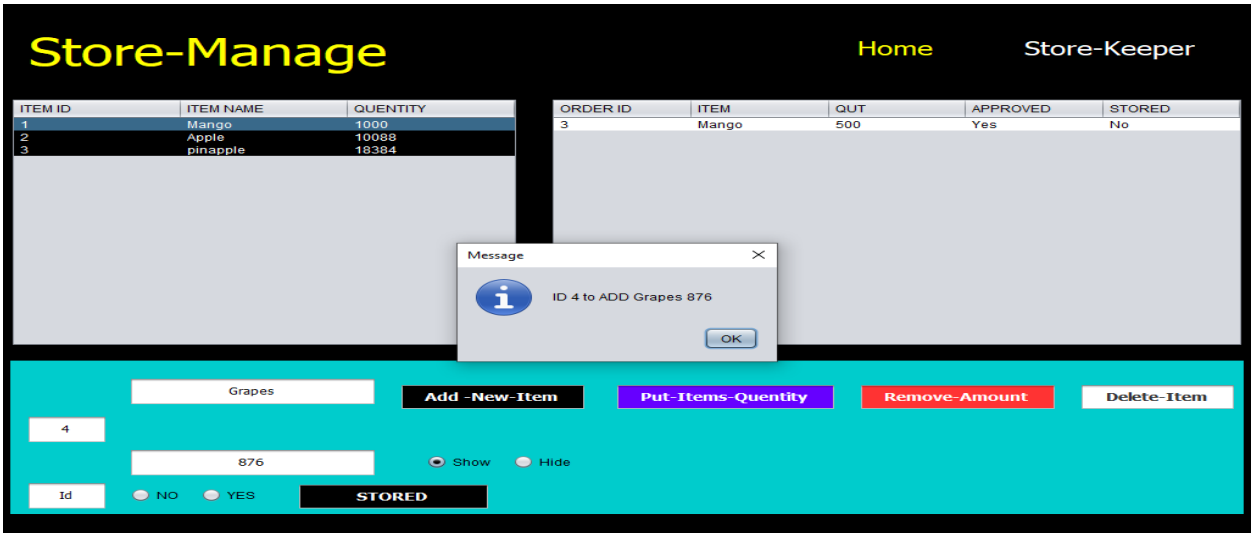
Test Case NO	01
TEST DATE	2020/11/11
Test OBJECTIVE	Check Update the User Registration data
OBJECTIVE ID	2
Test Data	Click Table Row. Enter Data below, i. ID="4" ii. First-Name= "Supuni" iii. Last-Name= "Sulakshana" iv. Address-Line 1= "No.68" v. Address-Line 1= "Galmaththa" vi. City-Name= "Aluthgama" vii. Select country= "Japan" viii. NIC-Number= "0764567890" ix. Age= "20" x. Contact-No= "0768755787" xi. Select Work type = "Cashier" xii. USER-NAME= "SupuniSulakshana999" xiii. Password= "99pass" xiv. Confirm password= "99pass" Then Click Update Button
Expected Results	"Update successfully" message coming and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.


Log-Out
User-Accounts
Home
Manager
Registration
4
Supuni
Sulakshana
No,68
Galmaththa
Aluthgama
Japan
0764567890
20
Male
Female
768755787
Cashier
SupuniSulakshana999
99pass
99pass
show
ID FIRST... LAST-... LINE... LINE-2 CITY COUNT... NIC ... GENDER CONTACT WORKED TYPE USER-NAME PASSWORD
4 savinsu pasintha no.57 ritiketiya aluthgama Sri Lanka 07687557... ... female 768755787 Manager u p
Message
..UPDATE SUCCESSFULL..!
OK
Show-Table
Hide-Table
Add
Update
Delete


Log-Out
User-Accounts
Home
Manager
Registration
4
Supuni
Sulakshana
No,68
Galmaththa
Aluthgama
Japan
0764567890
20
Male
Female
768755787
Cashier
SupuniSulakshana999
99pass
show
ID FIRST... LAST-... LINE... LINE-2 CITY COUNT... NIC ... GENDER CONTACT WORKED TYPE USER-NAME PASSWORD
4 Supuni Sulaks... No,68 Galma... Aluthgama Japan 07645678... ... female 768755787 Cashier SupuniSulakshana9... 99pass
Show-Table
Hide-Table
Add
Update
Delete

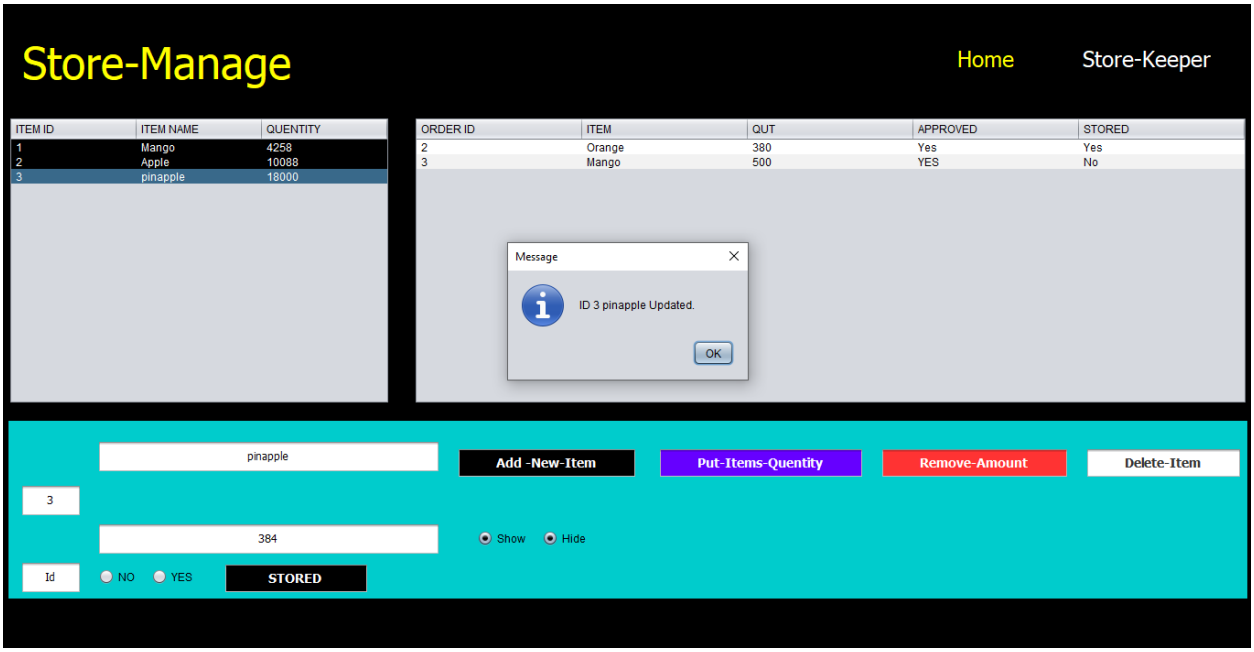
7. Store-Item Entry

Test Case NO	06
TEST DATE	2020/11/11
Test OBJECTIVE	Check ADD Items in the Store Manger
OBJECTIVE ID	1
Test Data	Enter Data below, i. ID=“4” ii. Item-Name= “Grapes” iii. Quantity= “878” Then Click ADD Button
Expected Results	“ID 4 to Add Grapes 876” message coming and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.



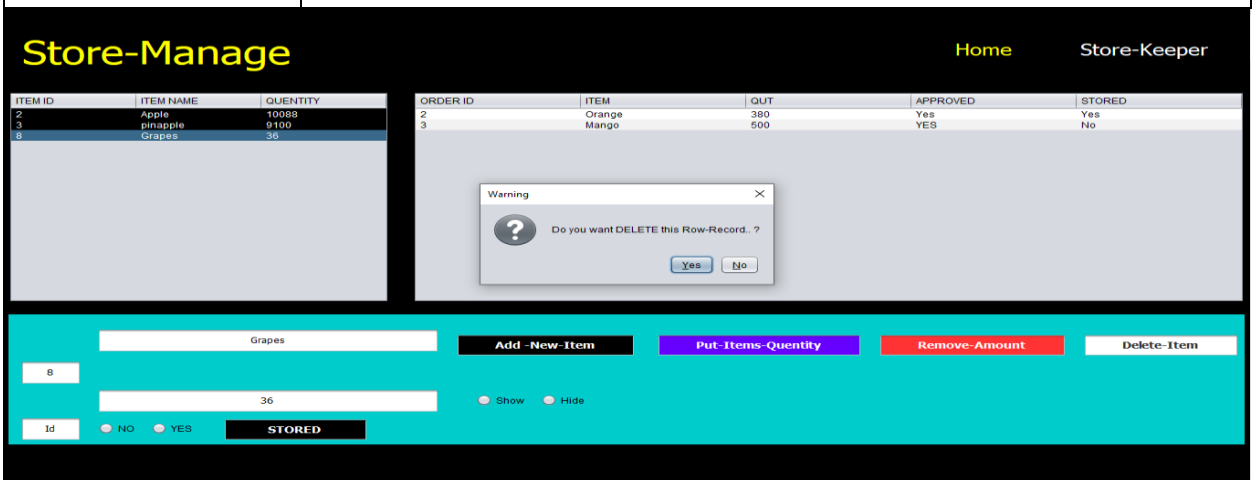
8. Store-Item Update

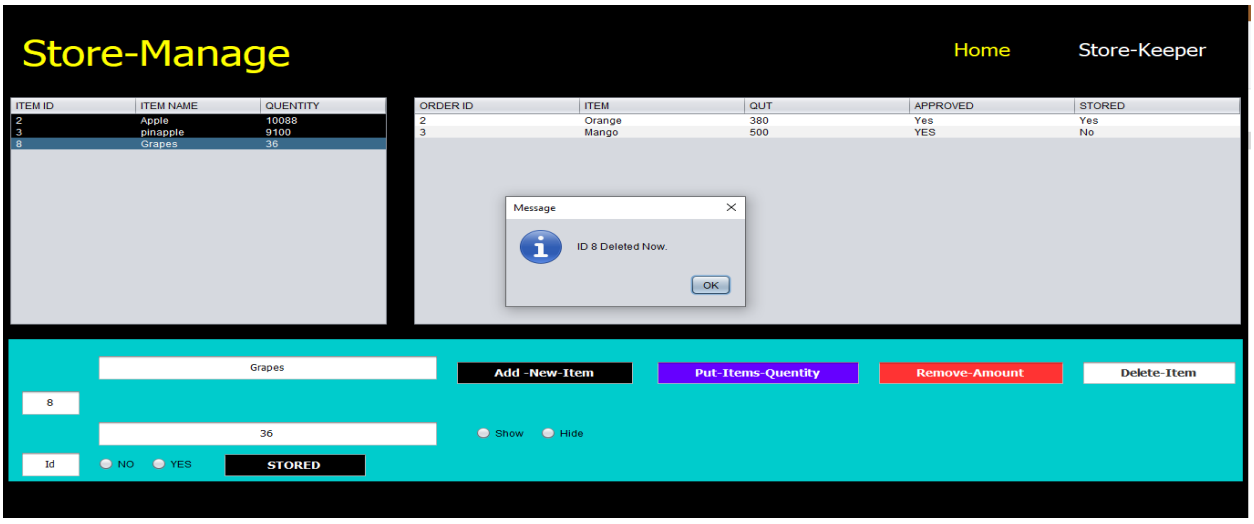
Test Case NO	06
TEST DATE	2020/11/11
Test OBJECTIVE	Check Update Items in the Store Manger
OBJECTIVE ID	2
Test Data	Enter Data below, i. ID=“3” ii. Item-Name= “Pineapple” iii. Quantity= “384” Then Click Put-Item-Quantity Button
Expected Results	“ID 3 Pineapple Updated” message coming and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.

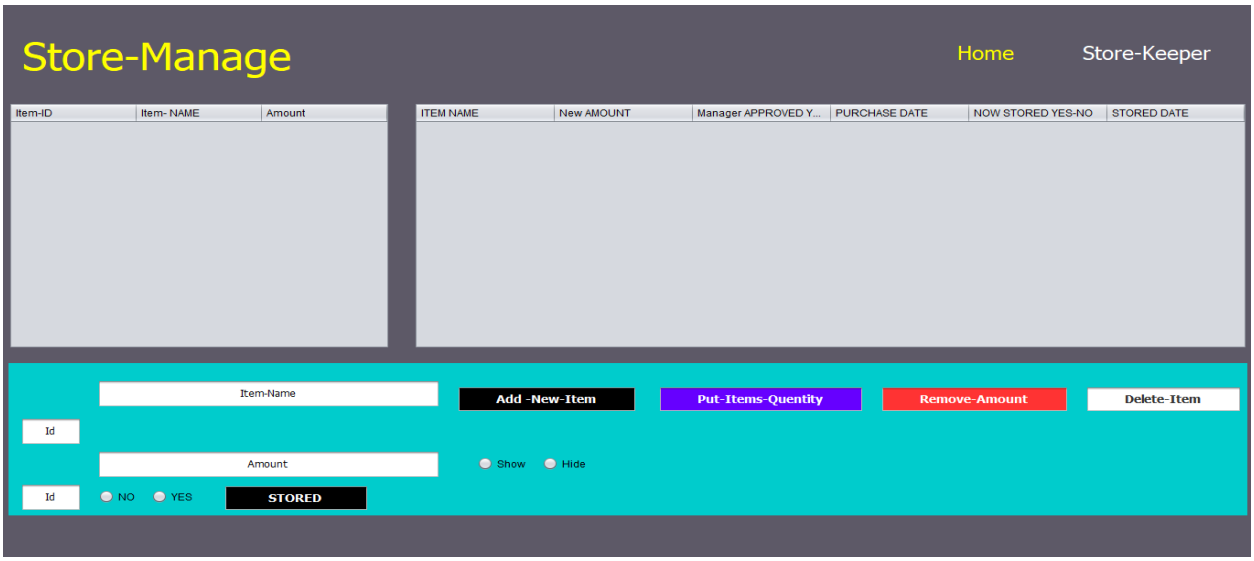


9. Store-Item Delete

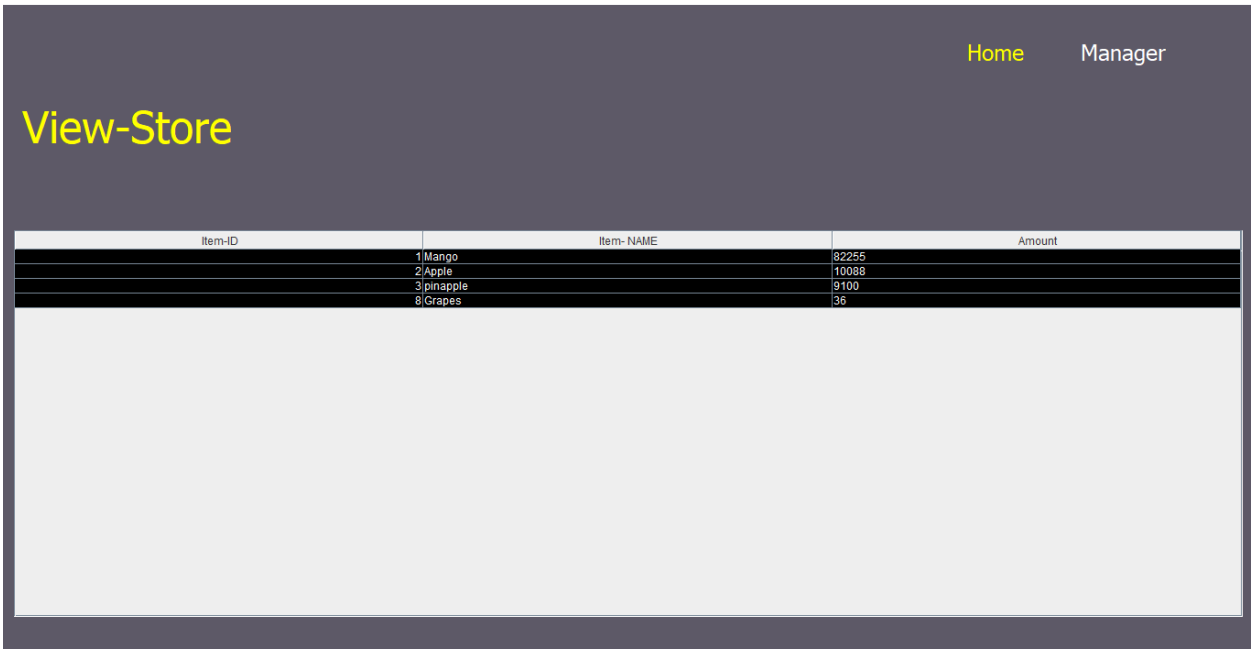
Test Case NO	06
TEST DATE	2020/11/11
Test OBJECTIVE	Check Delete the User Store Item
OBJECTIVE ID	3
Test Data	Click Table Row or ID = "8" enter Then Click Delete Button.
Expected Results	"Do you want to delete this row record?" message and view "deleted successful" message and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.





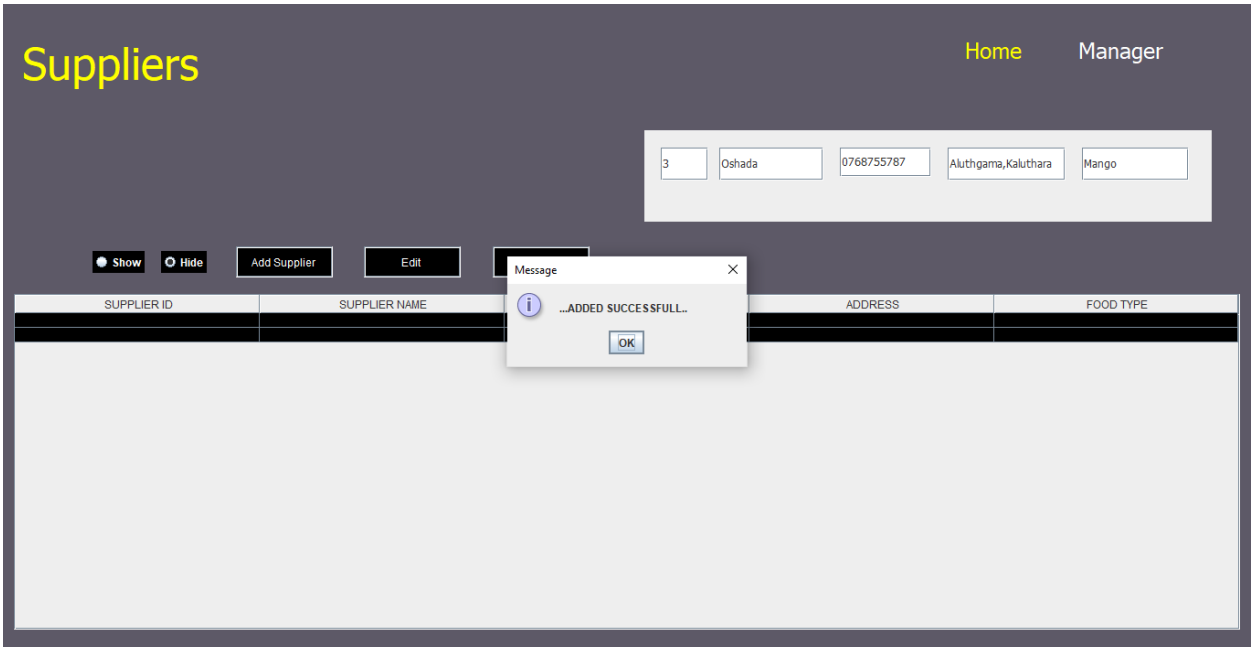


10.Store view data



11.Supplier Entry

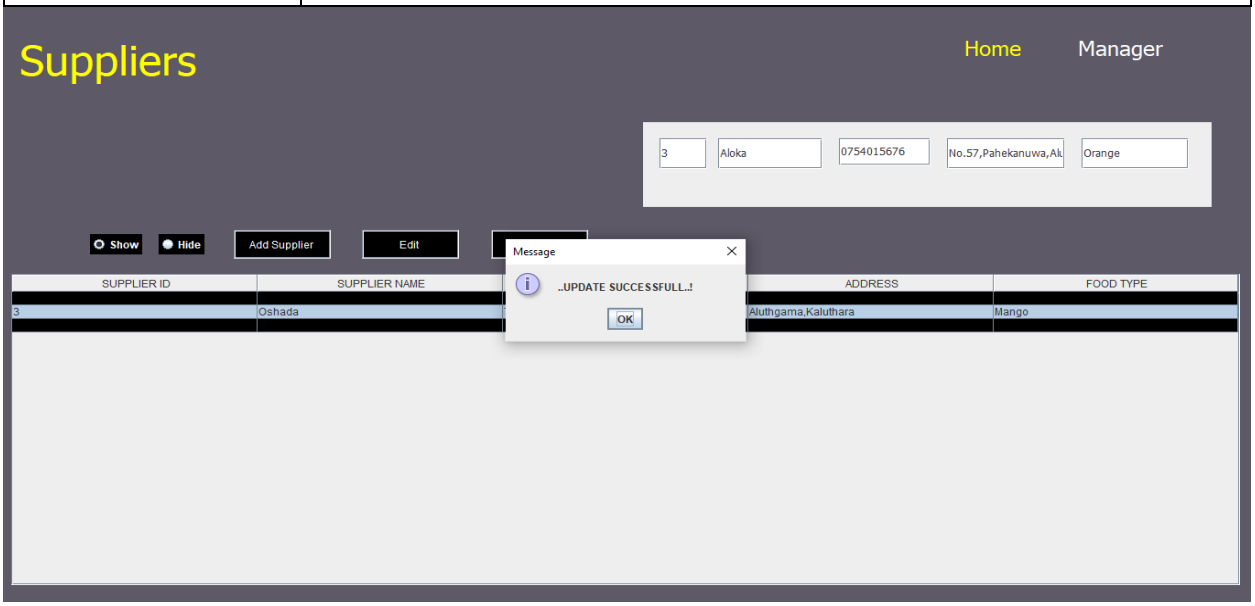
Test Case NO	04
TEST DATE	2020/11/11
Test OBJECTIVE	Check Delete the User Store Item
OBJECTIVE ID	1
Test Data	i. ID = "3" ii. Name = "Oshada" iii. Contact-No = "0768755787" iv. Address = "Aluthgama, Kalauthara" v. Food-Type = "Mango" Clicked added-supplier Button.
Expected Results	"Added Successfully" message and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.



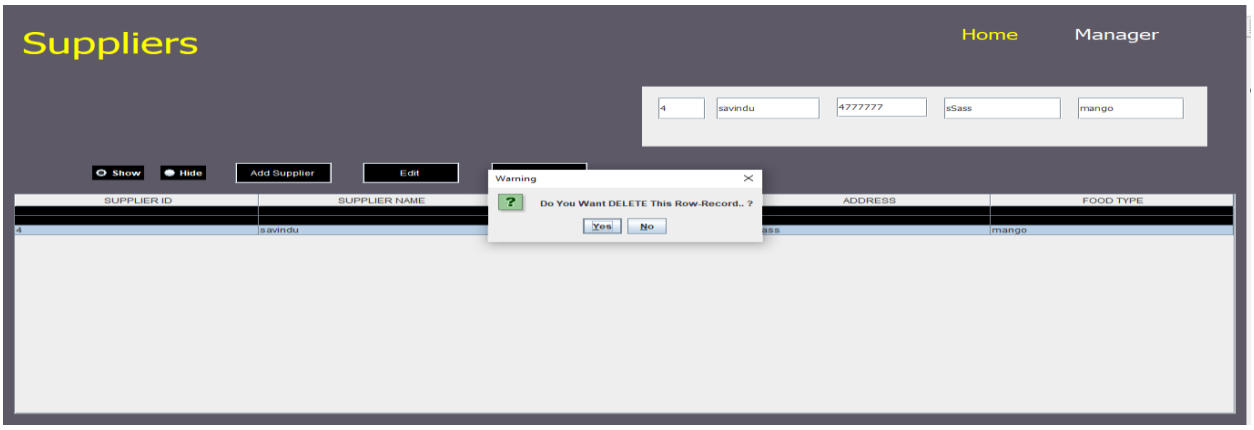
Test Case NO	04
TEST DATE	2020/11/11
Test OBJECTIVE	Check Delete the User Store Item
OBJECTIVE ID	2
Test Data	i. ID = "3" ii. Name = "Aloka" iii. Contact-No = "0754015676" iv. Address = "No.57, Pahekanuwa,Aluthgama" v. Food-Type = "Orange" Clicked Edit button.
Expected Results	"Updated Successfully" message and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.

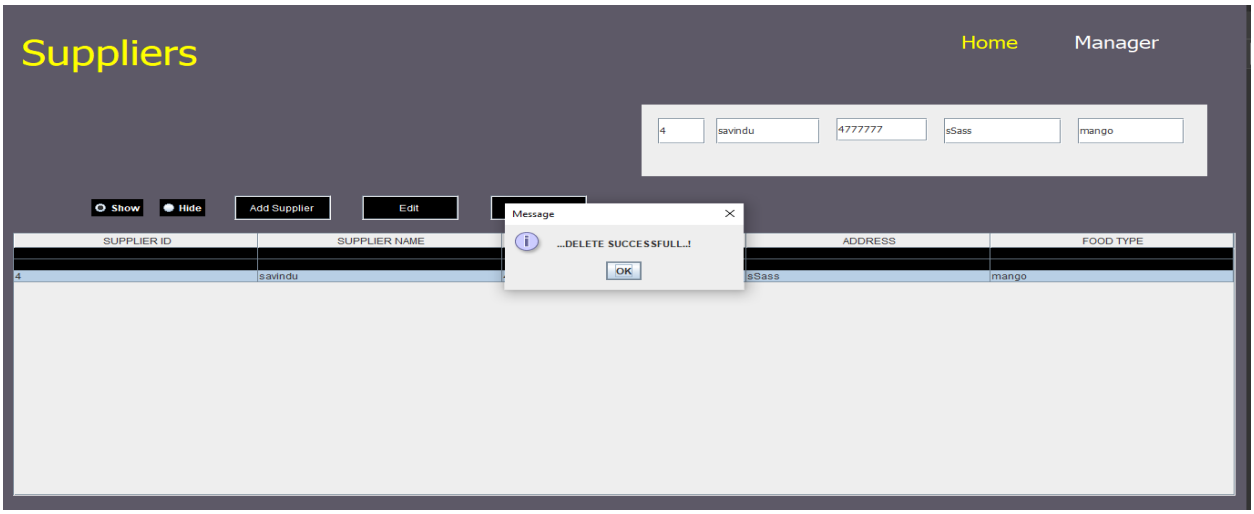
12.Supplier Update

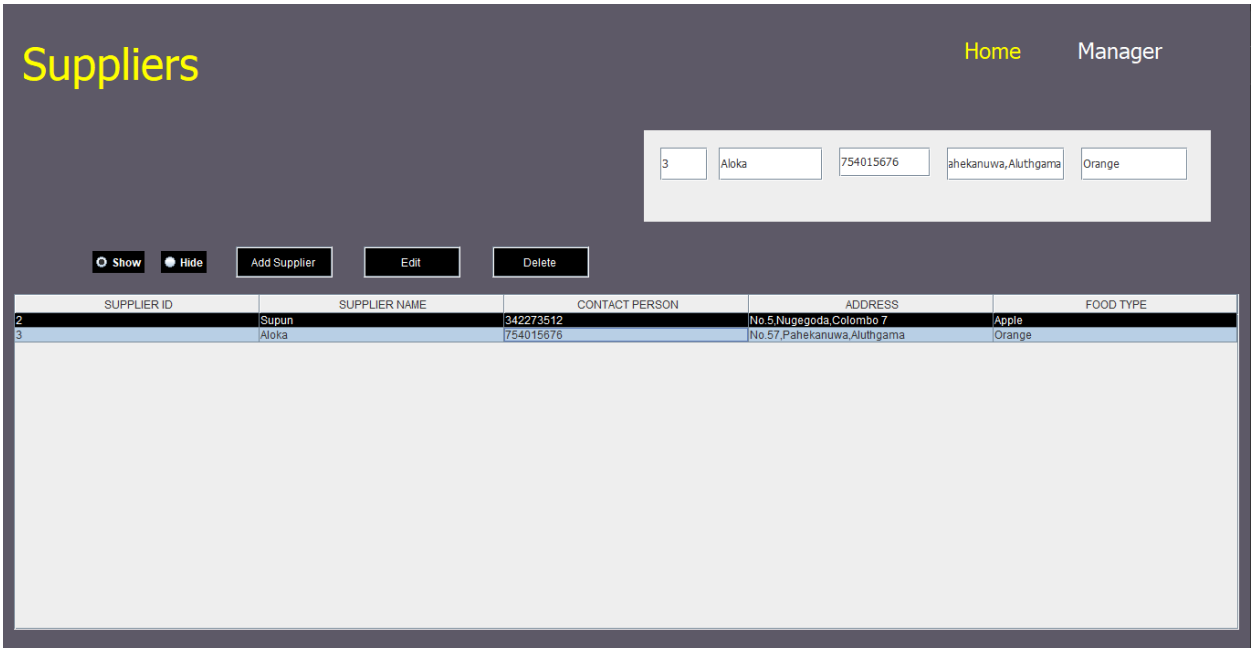
Test Case NO	04
TEST DATE	2020/11/11
Test OBJECTIVE	Check Delete the User Store Item
OBJECTIVE ID	3
Test Data	Click Table Row or ID = "4" enter Then Click Delete Button.
Expected Results	"Do you want to delete this row record?" message and view "deleted successful" message and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.



13.Supplier Delete

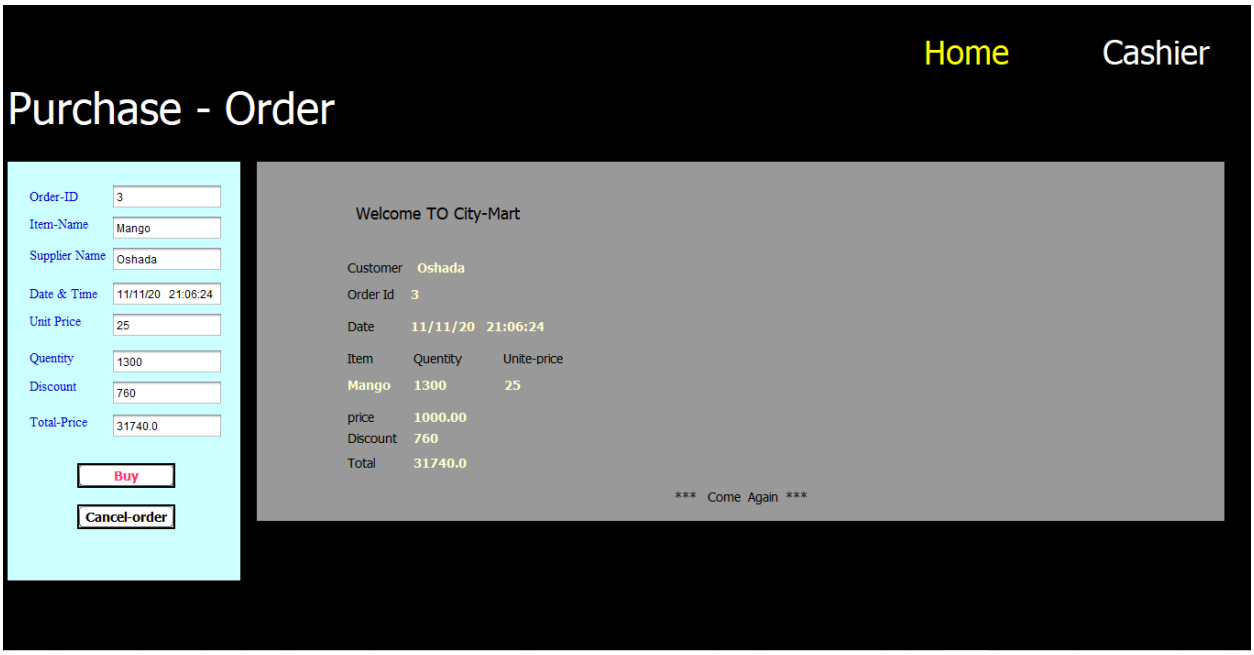

Suppliers
Home
Manager
4
savindu
4777777
sSass
mango
Show
Hide
Add Supplier
Edit
Warning
Do You Want DELETE This Row-Record.. ?
Yes
No
SUPPLIER ID
SUPPLIER NAME
ADDRESS
FOOD TYPE
4
savindu
mango


Suppliers
Home
Manager
4
savindu
4777777
sSass
mango
Show
Hide
Add Supplier
Edit
Message
...DELETE SUCCESSFULL..!
OK
SUPPLIER ID
SUPPLIER NAME
ADDRESS
FOOD TYPE
4
savindu
sSass
mango


Suppliers
Home
Manager
3
Aloka
754015676
ahekanuwa,Aluthgama
Orange
Show
Hide
Add Supplier
Edit
Delete
SUPPLIER ID
SUPPLIER NAME
CONTACT PERSON
ADDRESS
FOOD TYPE
2
Supun
342273512
No.5,Nugegoda,Colombo 7
Apple
3
Aloka
754015676
No.57,Pahekanuwa,Aluthgama
Orange

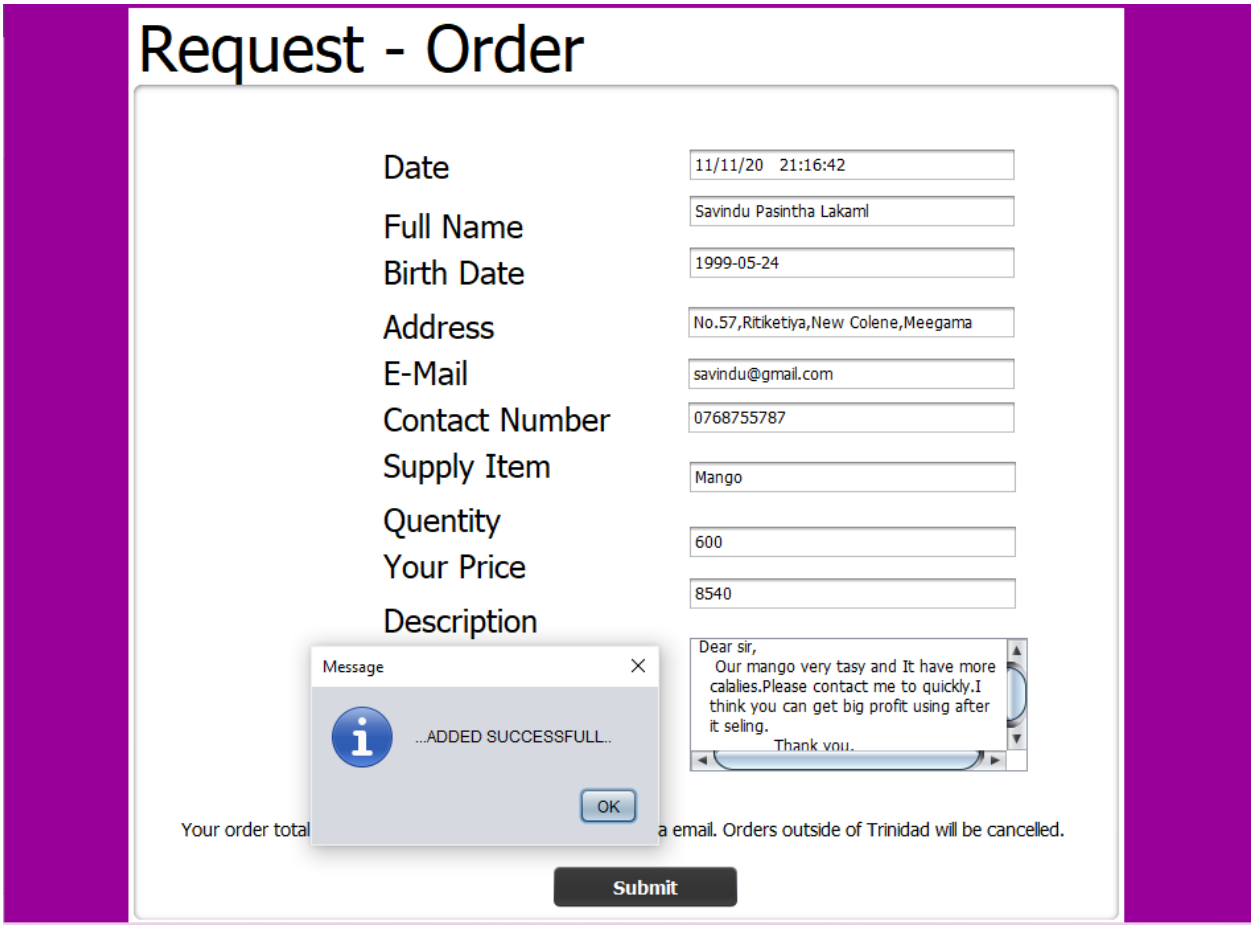
14.Generate Purchase Order (PO)

Test Case NO	05
TEST DATE	2020/11/11
Test OBJECTIVE	Generate bill and Purchase order
OBJECTIVE ID	1
Test Data	i. Order-ID="3" ii. Item-Name= "Mango" iii. Supplier-Name= "Oshada" iv. Date and Time = "" v. Unit-Price = "25" vi. Quantity = "1300" vii. Discount = "760" viii. Total = "" Clicked Buy Button.
Expected Results	Automatically Added "Total" and Bill print and clear the all text field's values.
Actual Results	Expected result Successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.

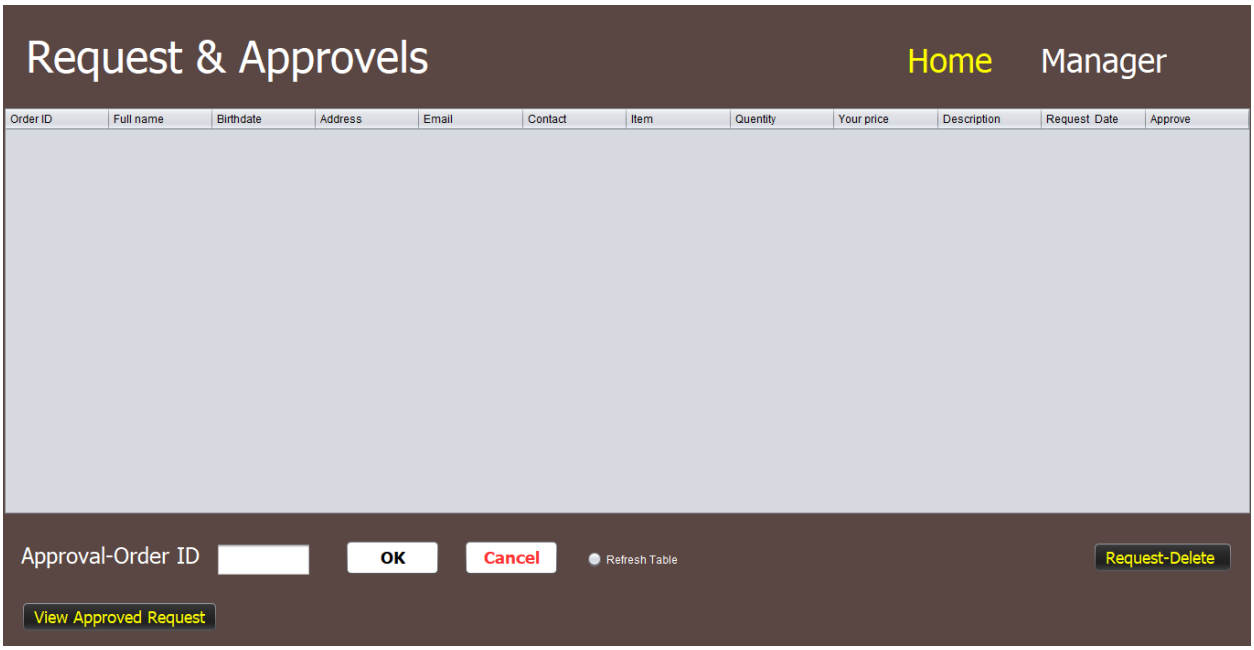

Home
Cashier
Purchase - Order
Order-ID
3
Item-Name
Mango
Supplier Name
Oshada
Date & Time
11/11/20 21:06:24
Unit Price
25
Quentity
1300
Discount
760
Total-Price
31740.0
Buy
Cancel-order
Welcome TO City-Mart
Customer Oshada
Order Id 3
Date 11/11/20 21:06:24
Item Quentity Unite-price
Mango 1300 25
price 1000.00
Discount 760
Total 31740.0
*** Come Again ***

15.Create a Purchase Requisition (PR)

Test Case NO	03
TEST DATE	2020/11/11
Test OBJECTIVE	User Account Registration
OBJECTIVE ID	1
Test Data	i. Date="11/11/20 21:16:42" ii. Full Name= "Savindu Pasingtha Lakml" iii. Birth Date = "" iv. Address= "pasingtha" v. E-Mail = "savindu@gmail.com" vi. Contact No = "0768755787" vii. Supply Item = "Mango" viii. Quantity = "600" ix. Your Price = "8540" x. Description = "Dear sir, our mango is very tasty and it have more calories. Please contact me quickly and I think you can get big profit after selling. Thank you." Enter submit button.
Expected Results	Added success full message view and clear the all text fields' values as an empty.
Actual Results	Expected result Successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.

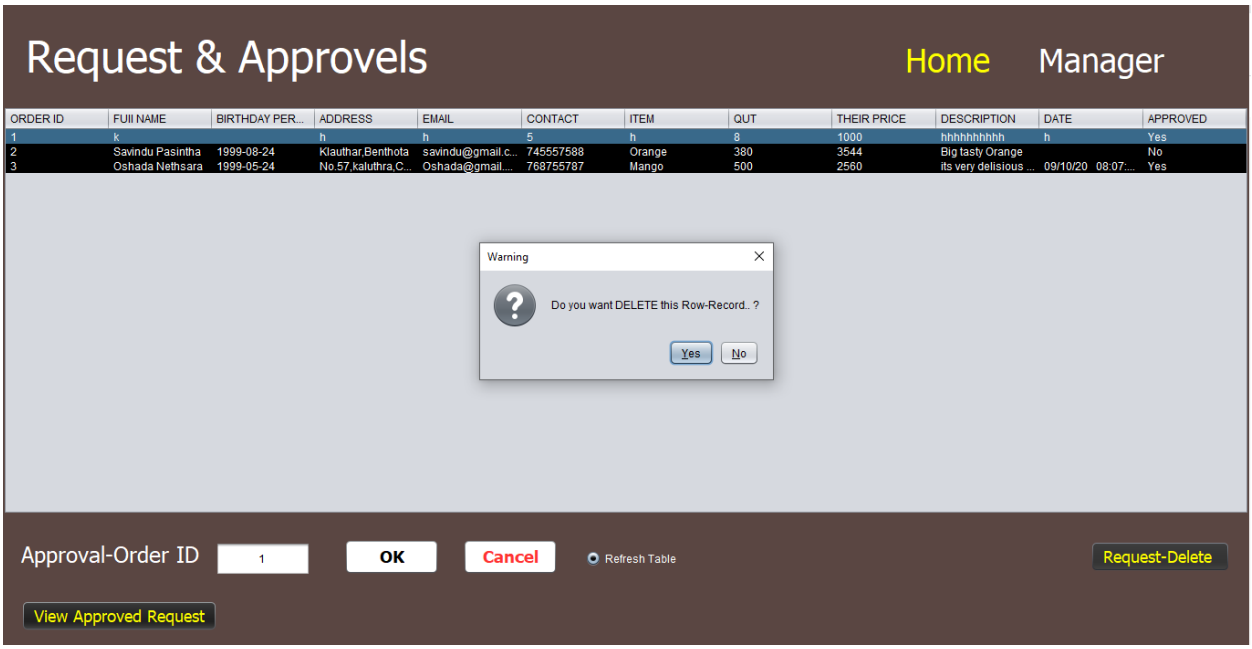

Request - Order
Date
11/11/20 21:16:42
Full Name
Savindu Pasintha Lakaml
Birth Date
1999-05-24
Address
No.57,Ritiketiya,New Colene,Meegama
E-Mail
savindu@gmail.com
Contact Number
0768755787
Supply Item
Mango
Quentity
600
Your Price
8540
Description
Dear sir,
Our mango very tasy and It have more calalies.Please contact me to quickly.I think you can get big profit using after it seling.
Thank you.
Message
...ADDED SUCCESSFULL..
OK
Your order total
a email. Orders outside of Trinidad will be cancelled.
Submit

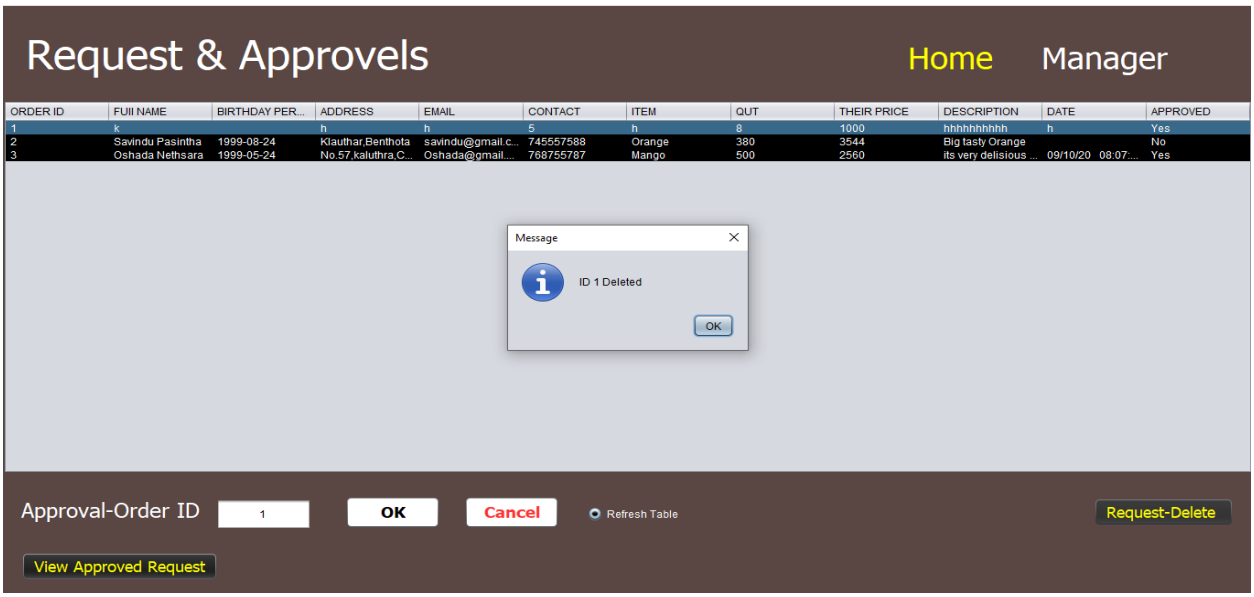
16.Approve


Request & Approvels
Home
Manager
Order ID
Full name
Birthdate
Address
Email
Contact
Item
Quentity
Your price
Description
Request Date
Approve
Approval-Order ID
OK
Cancel
Refresh Table
Request-Delete
View Approved Request

17.Request Delete

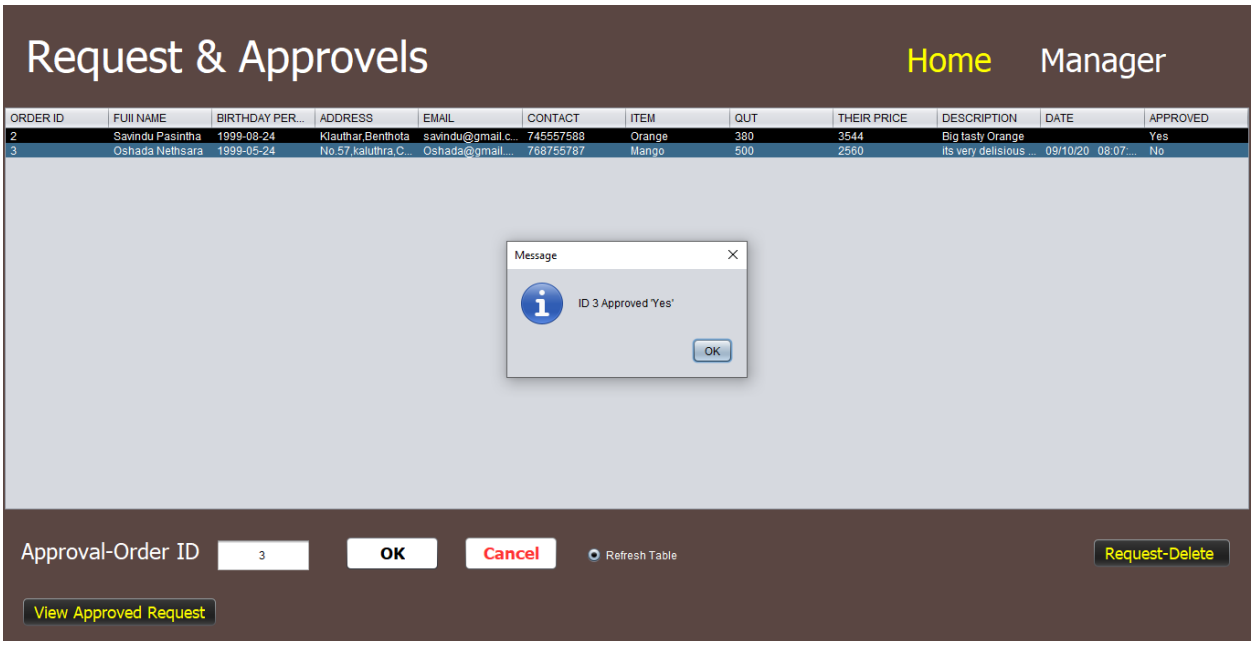
Test Case NO	02
TEST DATE	2020/11/11
Test OBJECTIVE	Check Delete the Request
OBJECTIVE ID	1
Test Data	Click Table Row or ID = "1" enter Then Click Request-Delete Button.
Expected Results	"Do you want to delete this row record?" message and view "deleted successful" message and after table Refresh.
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.

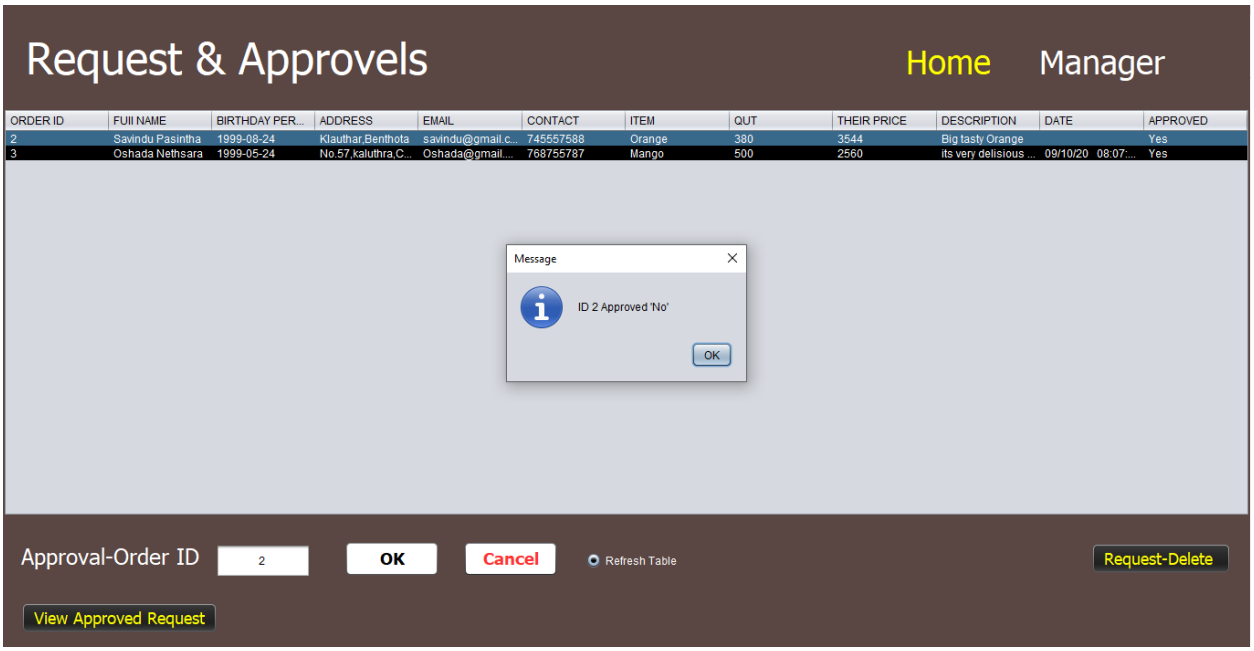


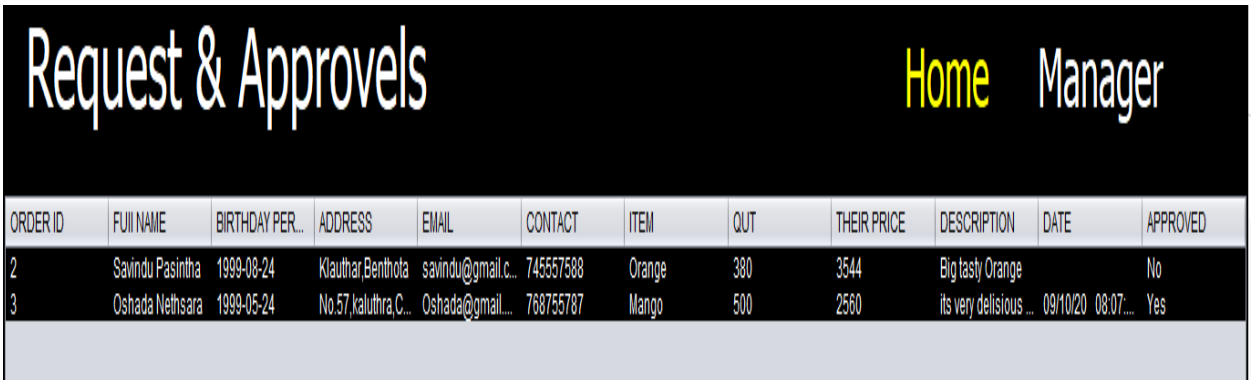

Request & Approvels
Home
Manager
ORDER ID
FUII NAME
BIRTHDAY PER...
ADDRESS
EMAIL
CONTACT
ITEM
QUT
THEIR PRICE
DESCRIPTION
DATE
APPROVED
1 k h h 5 h 8 1000 hhhhhhhhhh h Yes
2 Savindu Pasintha 1999-08-24 Klauthar,Benthota savindu@gmail.c... 745557588 Orange 380 3544 Big tasty Orange No
3 Oshada Nethsara 1999-05-24 No.57,kaluthra,C... Oshada@gmail.... 768755787 Mango 500 2560 its very delisious ... 09/10/20 08:07:... Yes
Message
ID 1 Deleted
OK
Approval-Order ID
1
OK
Cancel
Refresh Table
Request-Delete
View Approved Request

18.Approve the Request

Test Case NO	04
TEST DATE	2020/11/11
Test OBJECTIVE	Check Delete the User Store Item
OBJECTIVE ID	1
Test Data	Click Table Row or ID = "3" enter Then Click "ok" and "cancel" Button.
Expected Results	When "ok" clicked updated as "YES" When "cancel" clicked updated as "NO"
Actual Results	Expected result successfully.
Pass/Fail	Pass
Conclusion	Expected actual output.




Request & Approvels
Home
Manager
ORDER ID
FUII NAME
BIRTHDAY PER...
ADDRESS
EMAIL
CONTACT
ITEM
QUT
THEIR PRICE
DESCRIPTION
DATE
APPROVED
2
Savindu Pasintha
1999-08-24
Klauthar,Benthota
savindu@gmail.c...
745557588
Orange
380
3544
Big tasty Orange
Yes
3
Oshada Nethsara
1999-05-24
No.57,kaluthra,C...
Oshada@gmail....
768755787
Mango
500
2560
its very delisious ...
09/10/20 08:07:...
Yes
Message
ID 2 Approved 'No'
OK
Approval-Order ID
2
OK
Cancel
Refresh Table
Request-Delete
View Approved Request

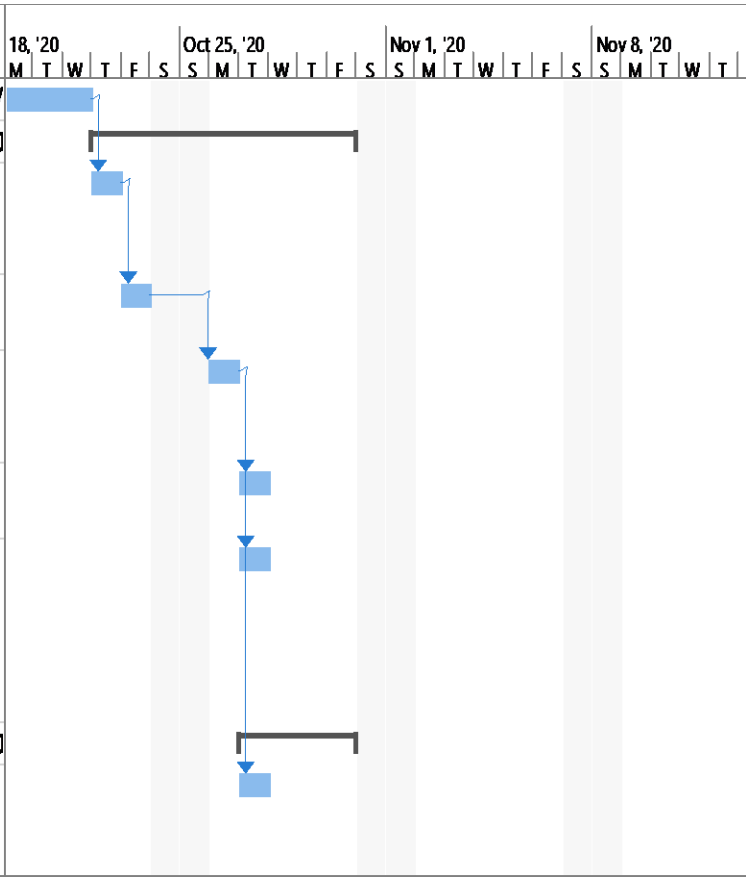

Request & Approvels
Home
Manager
ORDER ID
FUII NAME
BIRTHDAY PER...
ADDRESS
EMAIL
CONTACT
ITEM
QUT
THEIR PRICE
DESCRIPTION
DATE
APPROVED
2
Savindu Pasintha
1999-08-24
Klauthar,Benthota
savindu@gmail.c...
745557588
Orange
380
3544
Big tasty Orange
No
3
Oshada Nethsara
1999-05-24
No.57,kaluthra,C...
Oshada@gmail....
768755787
Mango
500
2560
its very delisious ...
09/10/20 08:07:...
Yes

CONCLUTION

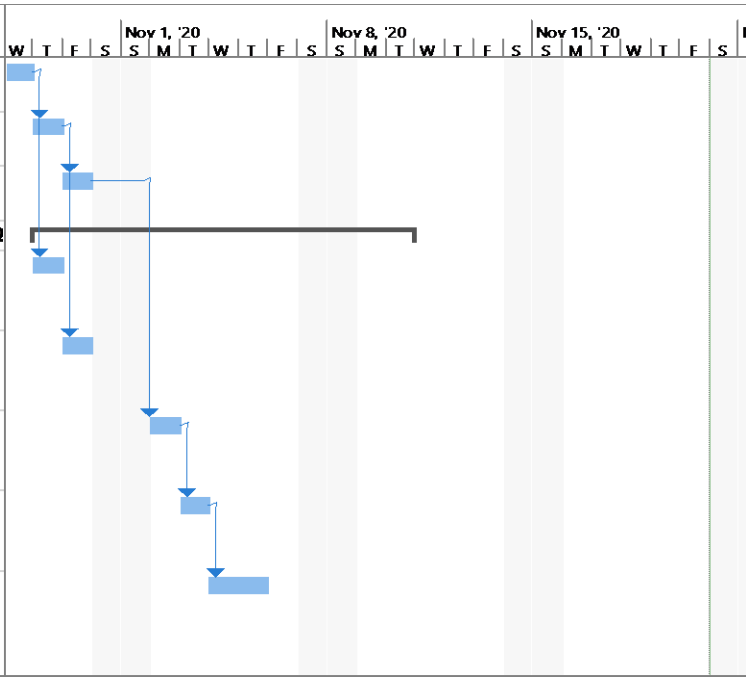
This automation order management system to created “city mart”. We can identify main actors as manager, cashier, and store keeper in the system. This system allows difference functions as automation doing to this actors. Manager can view the sales, suppliers, customer requests, stock and he can delete requests and approve the requests. Not only has that he can creating and managing system user accounts and suppliers. Cashier allows generate orders and receive bill. Not only that I added new function as a show the approved request view page to when generating the order to sometime check request details. Stock keeper can manage stock and create purchase request to customers. This system have more facilities to user to easy. Tables show, hide and refresh functions allows some security of the system. It can hide what is the data in the Database. I used user like bright colors to decorate the system. Some labels and background clicked the mouse, I it to added change the background colors functions. This system using all the data automatically store to database in ‘MySQL’. I think this system very simple to handle any one. So “city mart” can increase their sales every day with handle the organic fresh foods. Any one enter the invalid data to system, then quickly popup the error messages. When create this system I used java programming language object oriented concepts with the “APACHE NETBEANS 12.0 IDE” to coding. I think this system can use any other order automation system, after little changes in the system. Not only that I think this system will be very useful to success their business.

GANTT

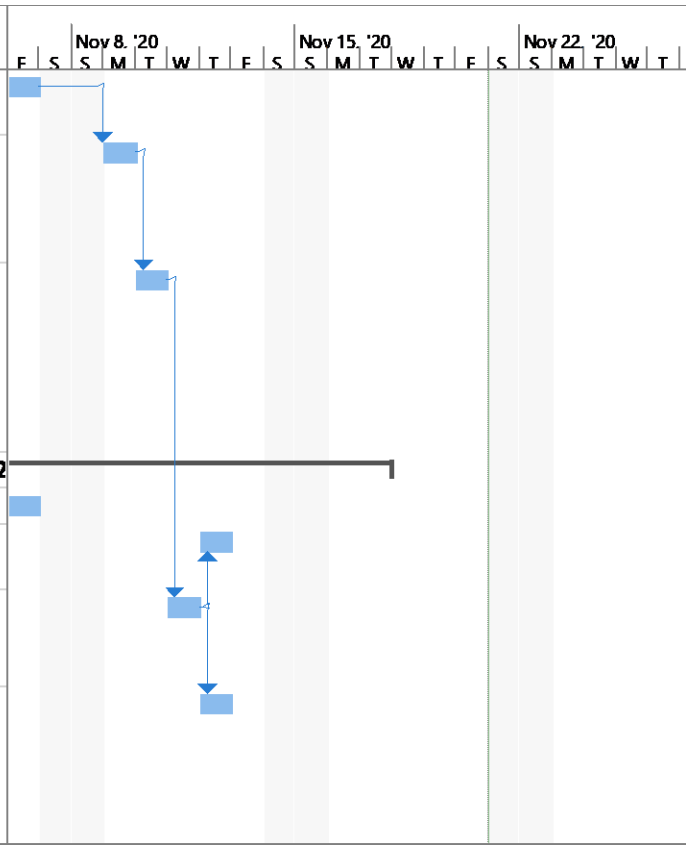
ID	Task Mode	Task Name	Duration	Start	Finish
1		Data Gathering	3 days	Mon 10/19/	Wed 10/21/
2		**TASK 1**	**7 days**	**Thu 10/22/2**	**Fri 10/30/20**
3		Discribe OOP concept of what is object,classes	1 day	Thu 10/22/20	Thu 10/22/20
4		Discribe java data types	1 day	Fri 10/23/20	Fri 10/23/20
5		Describe access modifiers and return types	1 day	Mon 10/26/20	Mon 10/26/20
6		Discribe constructors	1 day	Tue 10/27/20	Tue 10/27/20
7		Discribe inhitance,Polimo overloading,ove upcasting,down casting and	1 day	Tue 10/27/20	Tue 10/27/20
8		**TASK 2**	**4 days**	**Tue 10/27/2**	**Fri 10/30/20**
9		Identify system user/actors	1 day	Tue 10/27/20	Tue 10/27/20



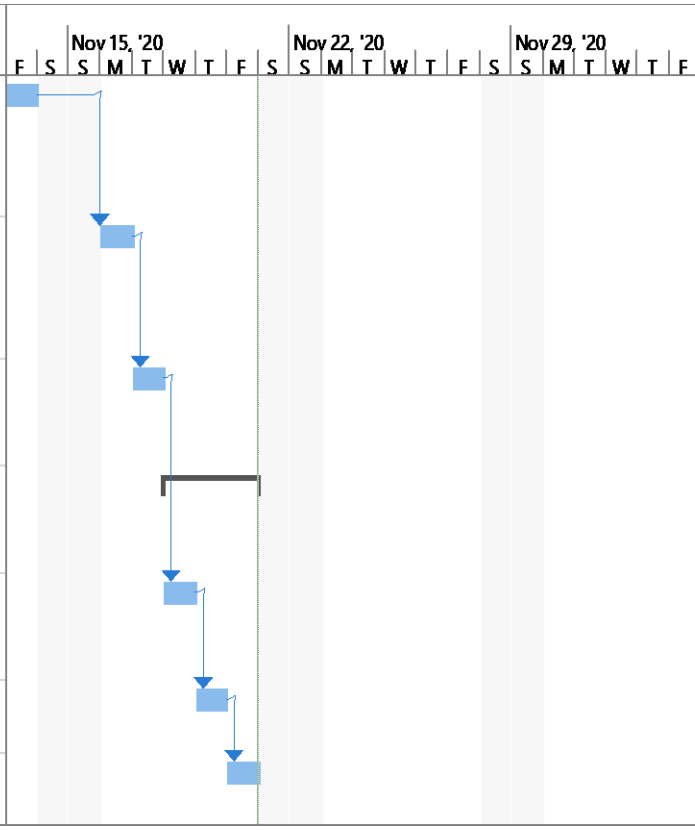
ID	Task Mode	Task Name	Duration	Start	Finish
10		Draw user case diagram	1 day	Wed 10/28/20	Wed 10/28/20
11		Draw class diagram	1 day	Thu 10/29/20	Thu 10/29/20
12		Draw sequence	1 day	Fri 10/30/20	Fri 10/30/20
13		**TASK 3**	**9 days?**	**Thu 10/29/2**	**Tue 11/10/2**
14		Database connection class and Home page	1 day	Thu 10/29/20	Thu 10/29/20
15		Supplier class and supplier pages designs	1 day	Fri 10/30/20	Fri 10/30/20
16		Cashier class and chashier pages create	1 day	Mon 11/2/20	Mon 11/2/20
17		Registration,pur order, and request class	1 day	Tue 11/3/20	Tue 11/3/20
18		User Registration,pur order, and request Design	2 days	Wed 11/4/20	Thu 11/5/20



ID	Task Mode	Task Name	Duration	Start	Finish
19		Login class and login page	1 day	Fri 11/6/20	Fri 11/6/20
20		Request Approve page and store keeper pages	1 day	Mon 11/9/20	Mon 11/9/20
21		Store manager class Create and all Store vie,sale view pages design and create classes	1 day?	Tue 11/10/20	Tue 11/10/20
22		**TASK 4**	**8 days**	**Fri 11/6/20**	**Tue 11/17/2**
23		Creat Test Plan	1 day	Fri 11/6/20	Fri 11/6/20
24		Test login page and create test	1 day	Thu 11/12/20	Thu 11/12/20
25		Test User registration page and test	1 day	Wed 11/11/20	Wed 11/11/20
26		Test Store page,Chasier page,Supplier pages and create test cases	1 day	Thu 11/12/20	Thu 11/12/20



ID	Task Mode	Task Name	Duration	Start	Finish
27		Test stor item manage and store vie pages and create test	1 day	Fri 11/13/20	Fri 11/13/20
28		Test supplier manage and vie pages and create test cases	1 day	Mon 11/16/20	Mon 11/16/20
29		Appvoed page tes ans create test case	1 day	Tue 11/17/20	Tue 11/17/20
30		**Other Documentation completing**	**3 days**	**Wed 11/18/20**	**Fri 11/20/20**
31		Acknowlegment and Abstract writing	1 day	Wed 11/18/20	Wed 11/18/20
32		Introduction writing	1 day	Thu 11/19/20	Thu 11/19/20
33		Conclution writing	1 day	Fri 11/20/20	Fri 11/20/20



References

Anon., 2011. *Access Modifiers in Java.* [Online] Available at: https://www.javatpoint.com/access-modifiers [Accessed 1 November 2020].

Anon., 2011. *static keyword in java.* [Online] Available at: https://www.javatpoint.com/static-keyword-in-java [Accessed 1 November 2020].

Anon., 2020. [Online] Available at: https://www.javatpoint.com/method-overriding-in-java [Accessed 1 November 2020].

Anon., 2020. *abstract-class-in-java.* [Online] Available at: https://www.javatpoint.com/abstract-class-in-java [Accessed 1 November 2020].

Anon., 2020. *Java constructor.* [Online] Available at: https://www.javatpoint.com/java-constructor [Accessed 1 November 2020].

Anon., 2020. *Java Data Types.* [Online] Available at: https://www.w3schools.com/java/java_data_types.asp [Accessed 1 November 2020].

Anon., 2020. *Method Overloading in Java.* [Online] Available at: https://www.javatpoint.com/method-overloading-in-java [Accessed 1 November 2020].

Rouse, M., 2020. *object-oriented programming (OOP).* [Online] Available at: https://searchapparchitecture.techtarget.com/definition/object-oriented-programming-OOP [Accessed 1 November 2020].