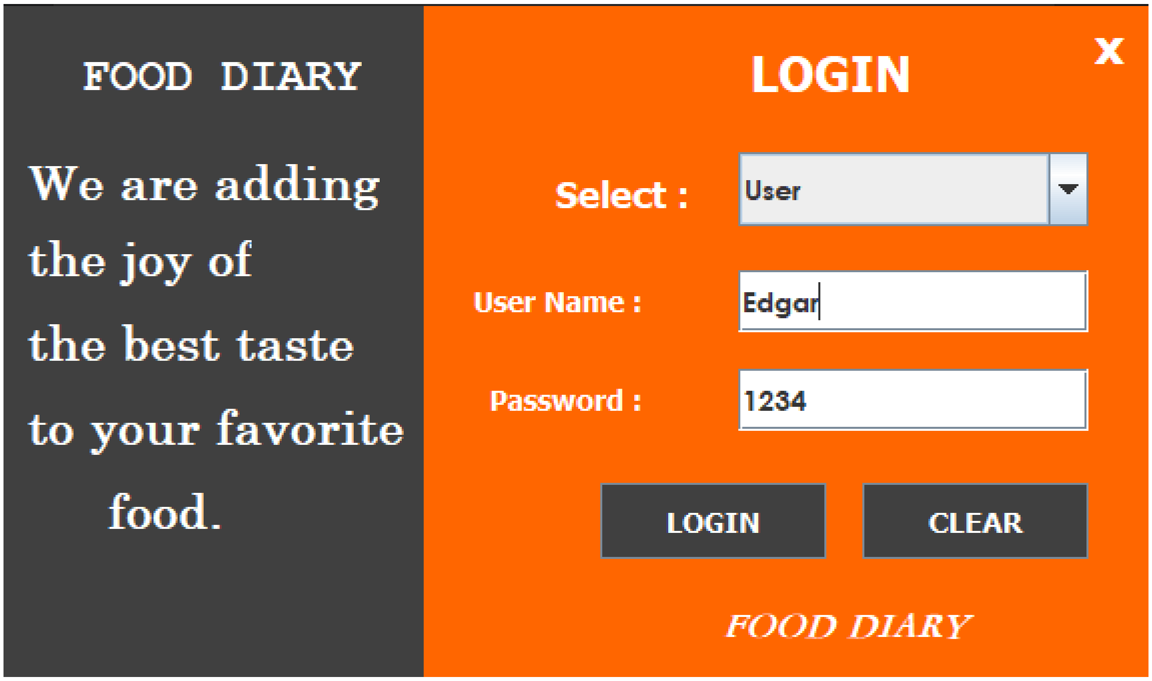
Management System

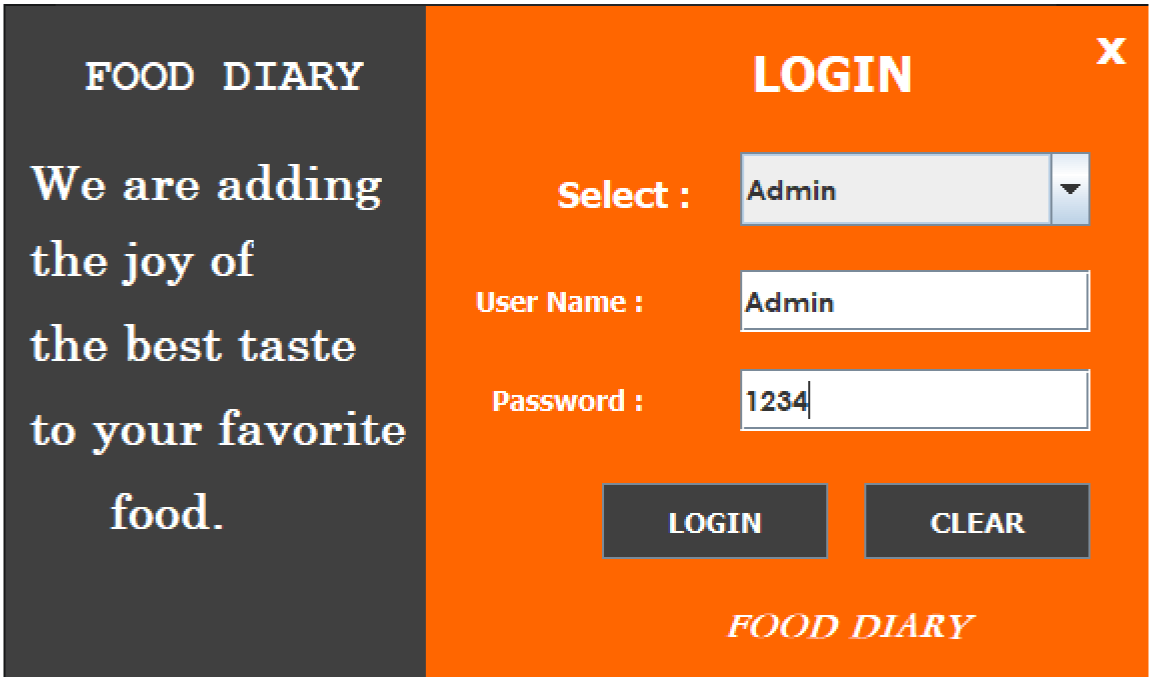


We are adding the joy of the best taste to your favorite food.

Select : User

User Name : Edgar

Password : 1234

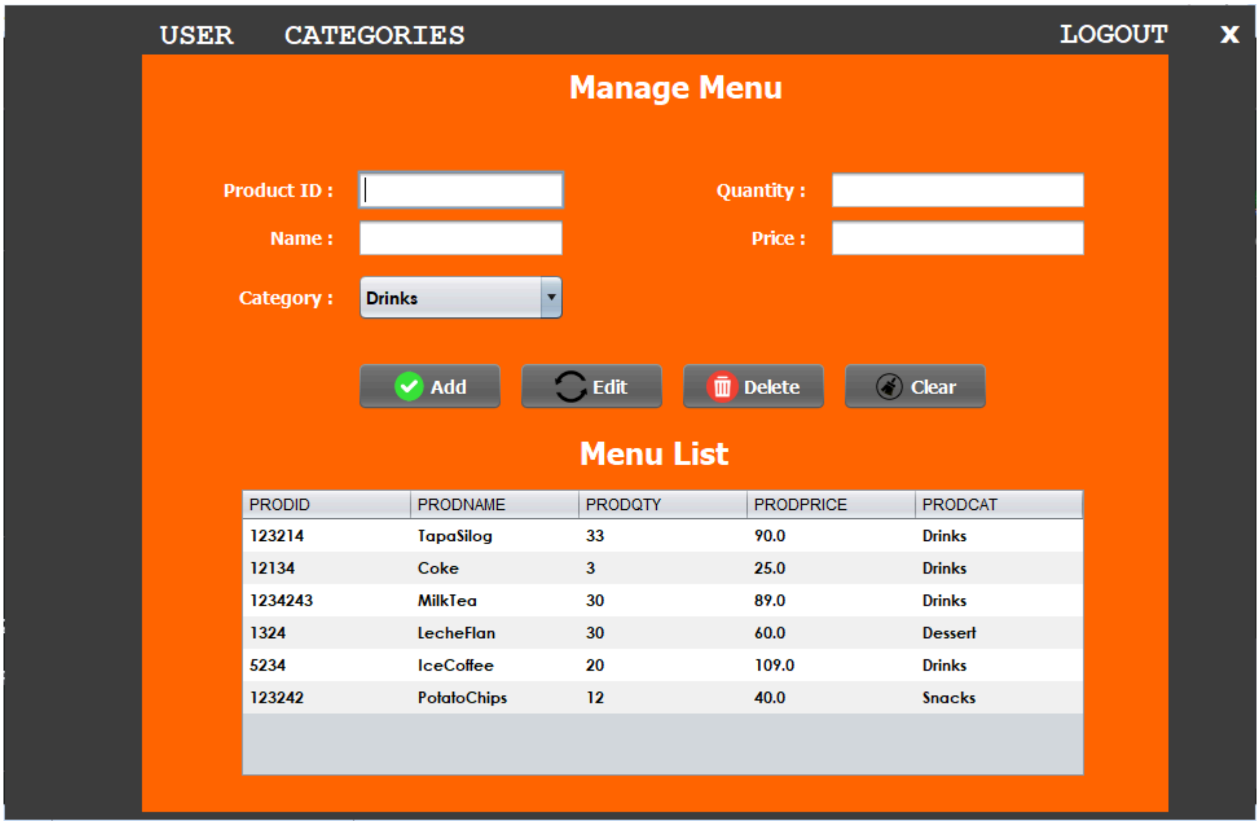


We are adding the joy of the best taste to your favorite food.

Select : Admin

User Name : Admin

Password : 1234

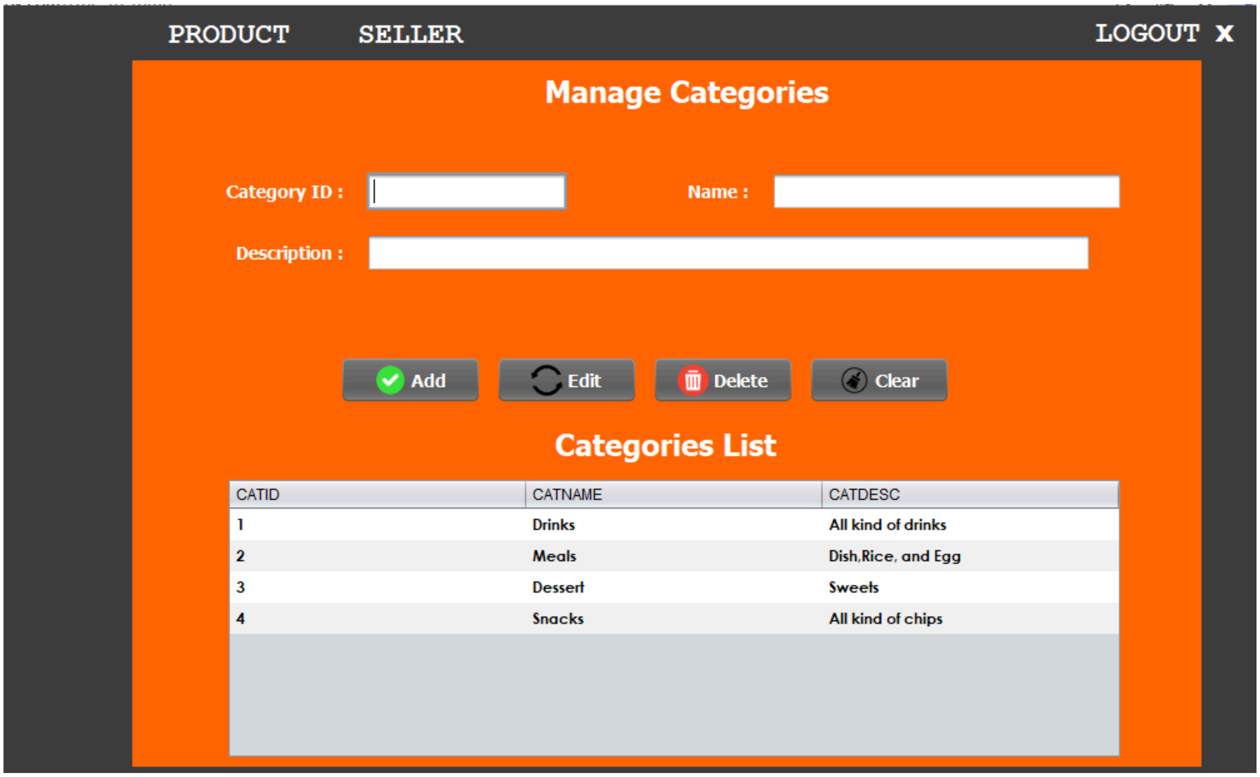


Manage Menu

Category : Drinks

Menu List

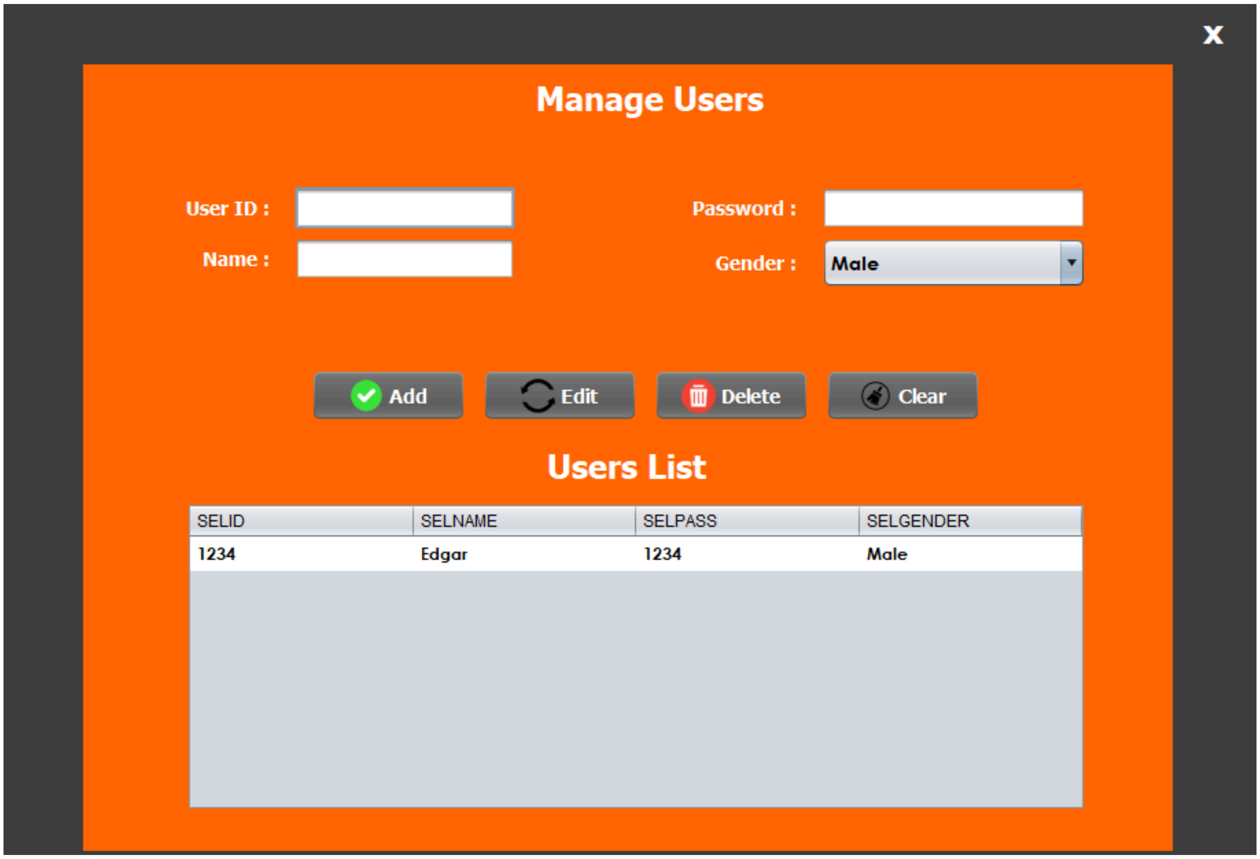
PRODID	PRODNAME	PRODQTY	PRODPRICE	PRODCAT
123214	TapaSilog	33	90.0	Drinks
12134	Coke	3	25.0	Drinks
1234243	MilkTea	30	89.0	Drinks
1324	LecheFlan	30	60.0	Dessert
5234	IceCoffee	20	109.0	Drinks
123242	PotatoChips	12	40.0	Snacks



Manage Categories

Categories List

CATID	CATNAME	CATDESC
1	Drinks	All kind of drinks
2	Meals	Dish,Rice, and Egg
3	Dessert	Sweets
4	Snacks	All kind of chips



Manage Users

User ID :

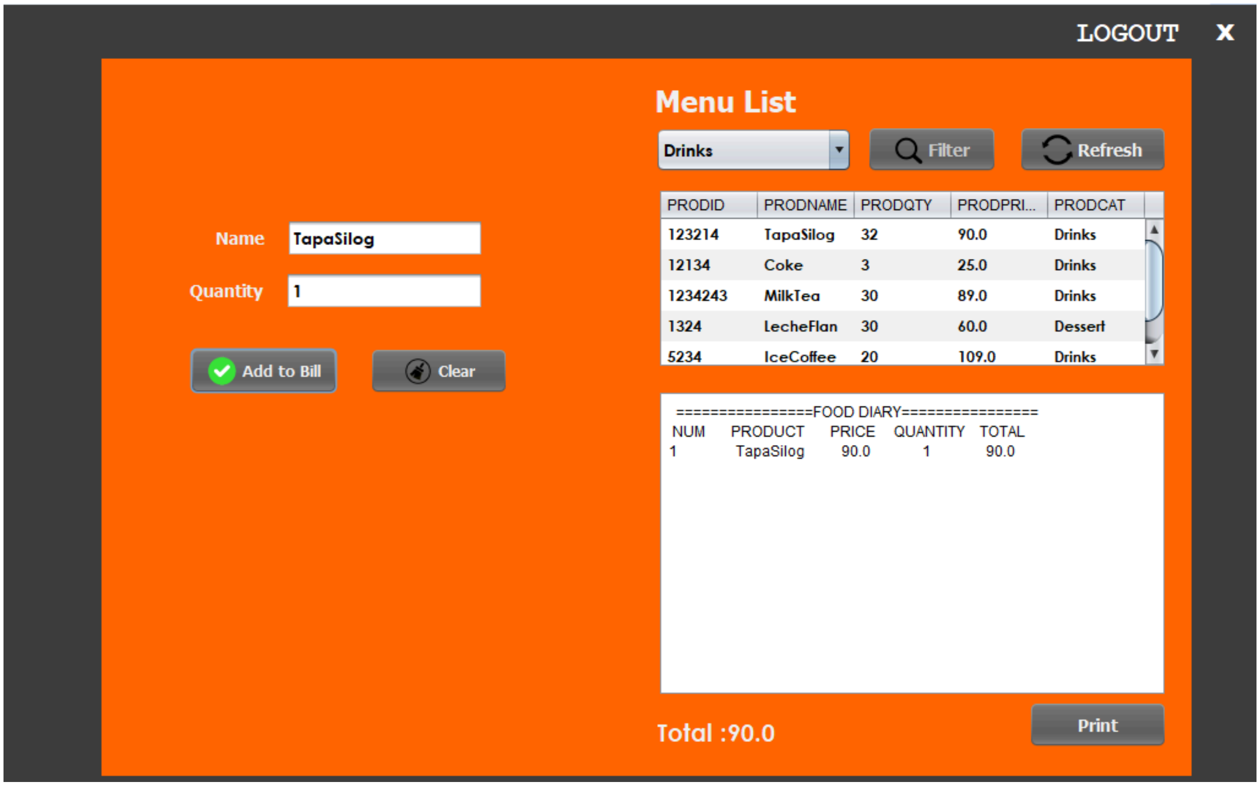
Password :

Name :

Gender : Male

Users List

SELID	SELNAME	SELPASS	SELGENDER
1234	Edgar	1234	Male



Menu List

Drinks

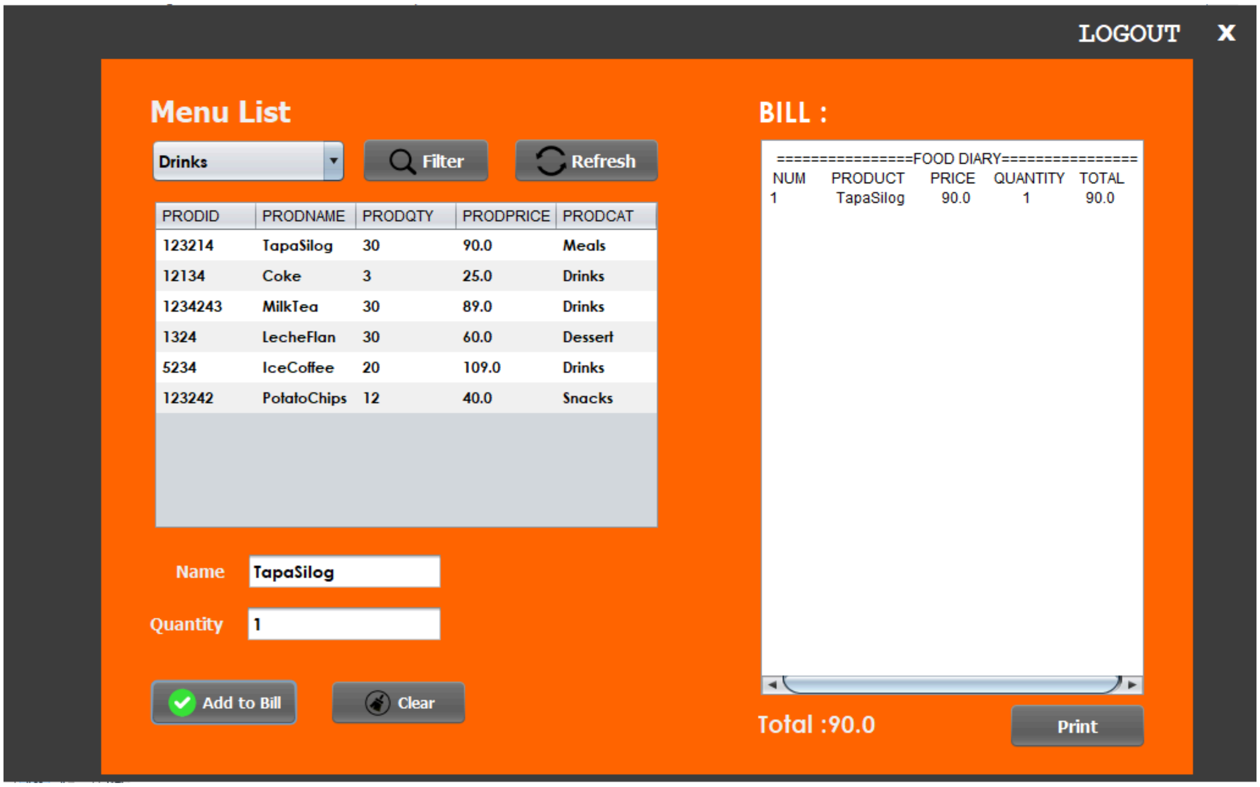
PRODID	PRODNAME	PRODQTY	PRODPRI...	PRODCAT
123214	TapaSilog	32	90.0	Drinks
12134	Coke	3	25.0	Drinks
1234243	MilkTea	30	89.0	Drinks
1324	LecheFlan	30	60.0	Dessert
5234	IceCoffee	20	109.0	Drinks

Name TapaSilog

Quantity 1

```
 ===============FOOD DIARY===============
NUM     PRODUCT     PRICE    QUANTITY   TOTAL
1         TapaSilog      90.0          1           90.0
```

Total :90.0



PRODID	PRODNAME	PRODQTY	PRODPRICE	PRODCAT
123214	TapaSilog	30	90.0	Meals
12134	Coke	3	25.0	Drinks
1234243	MilkTea	30	89.0	Drinks
1324	LecheFlan	30	60.0	Dessert
5234	IceCoffee	20	109.0	Drinks
123242	PotatoChips	12	40.0	Snacks

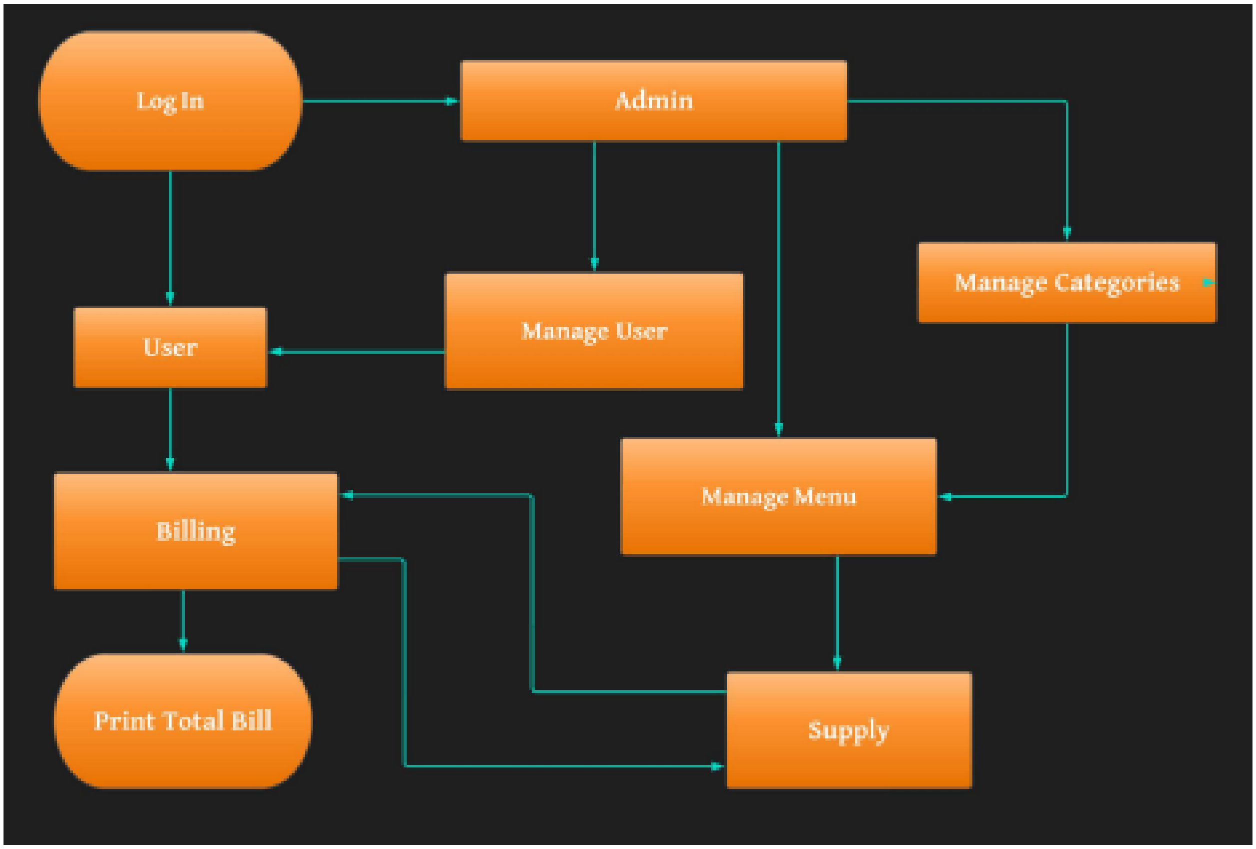
Name TapaSilog

Quantity 1

```
===============FOOD DIARY================
NUM      PRODUCT     PRICE   QUANTITY   TOTAL
1         TapaSilog    90.0       1         90.0
```

Total :90.0

Code Documentation



Login Page:

```
function public Login()
    call initComponents()

    // Declare variables
    Connection Con
    Statement St
    ResultSet Rs

    // Define initComponents function
    function initComponents()
        // Create GUI components
        jPanel1 = new Panel
        jLabel5 = new Label
        jLabel7 = new Label
        jLabel8 = new Label
        jLabel9 = new Label
        jLabel10 = new Label
        jLabel11 = new Label

        jPanel2 = new Panel
        jLabel1 = new Label
        Uid = new TextField
        password = new TextField
        RoleCb = new ComboBox
        jLabel2 = new Label
        jLabel3 = new Label
        jLabel4 = new Label
        jButton1 = new Button
        jButton2 = new Button
        jLabel6 = new Label
        jLabel12 = new Label

    // Define jLabel12MouseClicked function
    function jLabel12MouseClicked()
        exit the application

    // Define jButton2MouseClicked function
    function jButton2MouseClicked()
        if RoleCb.getSelectedItem() is "User" then
            Query = "select * from User1.SELERTBL where SELNAME='" + Uid.getText() + "' and SELPASS='" + password.getText() + "'" try
                Con = DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb","User1","1234")
                St = Con.createStatement()
                Rs = St.executeQuery(Query)
                if Rs.next() then
                    create new Selling window
                    close current window
                else
                    display "Wrong User ID or Password" message
                end if
            catch SQLException ex
                print stack trace
            end try
        else
            Query = "select * from User1.ADMINTBL where ADMINNAME='" +
Uid.getText() + "' and ADMINPASS='" + password.getText() + "'" try
                Con =
DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb","U
ser1","1234")
                St = Con.createStatement()
                Rs = St.executeQuery(Query)
                if Rs.next() then
                    create new Products window
                    close current window
                else
                    display "Wrong Admin ID or Password" message
                end if
            catch SQLException ex
                print stack trace
            end try
        end if

    // Define jButton1MouseClicked function
    function jButton1MouseClicked()
        // Clear input fields
        Uid.setText("")
        password.setText("")

    // Main method
    function main(args)
        // Set look and feel
        setLookAndFeel("Nimbus")

        // Create and display the login window
        create new Login window
        set location to center of the screen
```

Manage Category (Admin Page)

```
class Category extends javax.swing.JFrame {

    Category() {
        initComponents();
        SelectSeller();
    }

    Connection Con = null;
    Statement St = null;
    ResultSet Rs = null;

    SelectSeller() {
        try {
            // Establish database connection
            Con =
DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb",
"User1", "1234");
            St = Con.createStatement();

            // Execute SQL query to retrieve data from CATEGORYTBL Rs
            = St.executeQuery("Select * from User1.CATEGORYTBL");

            // Display the query results in the CategoryTable
            CategoryTable.setModel(DbUtils.resultSetToTableModel(Rs));
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    // Other methods and variables...

    public static void main(String[] args) {
        // Create an instance of the Category class
        Category category = new Category();

        // Set the Category window visible
        category.setVisible(true);
    }

    // Other GUI-related methods and components...
}
```

Manage Product (Admin Page)

```
import java.sql.*;

public class Products {
    public Products() {
        initComponents();
        selectSeller();
        getCat();
    }

    private void initComponents() {
        // Code for initializing components and setting up the UI
        // ...
    }

    private void selectSeller() {
        try {
            Connection con =
DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb",
"User1", "1234");
            Statement st = con.createStatement();
            ResultSet rs = st.executeQuery("SELECT * FROM
User1.PRODUCTTBL");
            ProductTable.setModel(DbUtils.resultSetToTableModel(rs));
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private void getCat() {
        try {
            Connection con =
DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb",
"User1", "1234");
            Statement st = con.createStatement();
            String query = "SELECT * FROM User1.CATEGORYTBL";
            ResultSet rs = st.executeQuery(query);
            while (rs.next()) {
                String myCat = rs.getString("CATNAME");
                CatCb.addItem(myCat);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private void EditBtnMouseClicked(java.awt.event.MouseEvent evt) {
        // Code for handling Edit button click event
        // ...
    }

    private void AddBtnMouseClicked(java.awt.event.MouseEvent evt) {
        // Code for handling Add button click event
        // ...
    }

    private void DeleteBtnMouseClicked(java.awt.event.MouseEvent evt) {
        // Code for handling Delete button click event
        // ...
    }

    private void ClearBtnMouseClicked(java.awt.event.MouseEvent evt) {
        // Code for handling Clear button click event
        // ...
    }

    private void ProductTableMouseClicked(java.awt.event.MouseEvent evt) {
        // Code for handling ProductTable mouse click event
        // ...
    }

    public static void main(String[] args) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Products().setVisible(true);
            }
        });
    }
}
```

Manage Seller (Admin Page)

```
class Seller extends javax.swing.JFrame {

    Seller() {
        initComponents();
        SelectSeller();
    }

    Connection Con = null;
    Statement St = null;
    ResultSet Rs = null;

    SelectSeller() {
        try {
            Con =
DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb",
"User1", "1234");
            St = Con.createStatement();
            Rs = St.executeQuery("Select * from User1.SELERTBL");
        SelerTable.setModel(DbUtils.resultSetToTableModel(Rs)); }
        catch (Exception e) {
            e.printStackTrace();
        }
    }

    initComponents() {
        // GUI component initialization
        // Add action listeners to buttons and table mouse click event
    }

    EditBtnMouseClicked(evt) {
        // Handle Edit button click event
    }

    AddBtnMouseClicked(evt) {
        // Handle Add button click event
    }

    DeleteBtnMouseClicked(evt) {
        // Handle Delete button click event
    }

    ClearBtnMouseClicked(evt) {
        // Handle Clear button click event
    }

    SelerTableMouseClicked(evt) {
        // Handle Seller table mouse click event
    }
}

// Create an instance of the Seller class
Seller seller = new Seller();
```

Menu List And Billing (User Page)

```
function Selling()
    initComponents()
    SelectSeller()
    GetCat()

    Connection Con = null
    Statement St = null
    ResultSet Rs = null
    SelectSeller()
    int Prid, newQty

    update()

    GetCat()

function SelectSeller()
    try
        Con = DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb", "User1", "1234")
        St = Con.createStatement()
        Rs = St.executeQuery("Select * from User1.PRODUCTTBL")
    ProductTable.setModel(DbUtils.resultSetToTableModel(Rs))
    catch(Exception e)
        e.printStackTrace()

function update()
    newQty = AvailQty - Integer.valueOf(ProdQty.getText())
    try
        Con = DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb", "User1", "1234")
        String Query = "Update User1.PRODUCTTBL set PRODQTY=" + newQty + " where PRODID=" + Prid
        Statement Add = Con.createStatement()
        Add.executeUpdate(Query)
        SelectSeller()
    catch(SQLException e)
        e.printStackTrace()

function GetCat()
    try
        Con = DriverManager.getConnection("jdbc:derby://localhost:1527/restaurantdb", "User1", "1234")
        St = Con.createStatement()
        String query = "Select * from User1.CATEGORYTBL"
        Rs = St.executeQuery(query)
        while(Rs.next())
            String Mycat = Rs.getString("CATNAME")
            CatCb.addItem(Mycat)
    catch(Exception e)
        e.printStackTrace()

// Generated Code
function initComponents()
    // Code for initializing components and UI layout

function CatCbItemStateChanged(evt)
    // Code for handling item state change event of CatCb

function CatCbMouseClicked(evt)
    // Code for handling mouse click event of CatCb

function EditBtnMouseClicked(evt)
    // Code for handling mouse click event of EditBtn

function AddBtnMouseClicked(evt)
    // Code for handling mouse click event of AddBtn

function DeleteBtnMouseClicked(evt)
    // Code for handling mouse click event of DeleteBtn

function ClearBtnMouseClicked(evt)
    // Code for handling mouse click event of ClearBtn

function ProductTableMouseClicked(evt)
    // Code for handling mouse click event of ProductTable

function jButton1MouseClicked(evt)
    // Code for handling mouse click event of jButton1
```