BAHRIA UNIVERSITY KARACHI CAMPUS

DEPARTMENT OF ARTIFICIAL INTELLIGENCE

Bahria University
Discovering Knowledge

TOPIC NAME

RESTAURANT MANAGEMENT SYSTEM

COURSE NAME

OBJECT ORIENTED PROGRAMMING

COURSE CODE

CSC 210

NAMES	ENROLLMENT NO
MAIMOONA SABIR	02-136212-047
ADIBA ROHILLA	02-136212-013
HIBA JAMAL	02-136212-049

Acknowledge

First and foremost, I want to express my gratitude to our DLD teacher, *Dr. Asif Aziz*, for his assistance. We'd also like to thank our Lab Engineer *Laila Nadeem* for her invaluable advice and assistance. This project would not have been completed without their assistance. We are also grateful to the University/Faculty for providing us with the opportunity to study *(OOP)* at such a prestigious institution. Thank you for creating a safe environment for all students and for encouraging creativity, interaction, and participation in academic activities. This project has provided us with opportunities to learn and apply our theoretical knowledge in the real world. As a result, I would like to express my gratitude to her for her assistance and guidance on this project.

Abstract

Restaurant Management Systems, or RMS, are critical technology that allow a single outlet or organization to better serve its customers and assist personnel with food and beverage transactions and controls. A restaurant management system is a database application that maintains track of all daily transactions in the restaurant. The Restaurant Management System assists restaurant management in keeping an acceptable record of all transactions carried out and will still be carried out by the business, as well as maintaining the restaurant's database.

Contents

Introduction

This restaurant management system can be used by employees in a restaurant to handle customers, their orders, and to assist them find what they are looking for. The restaurant menu is divided into menu item categories (appetizers, salads, entrees, sides, and beverages). Each item on the menu has a name and a price. A restaurant management system is a system for running a restaurant. Restaurant management might differ across numerous management styles, but when it comes to creating goals, there is always one common denominator: enhancing a restaurant's profitability. To optimize a restaurant's profitability, one must always review and comprehend its operational expenses, as well as how these connect to the restaurant's productivity and efficiency in providing great service to its clients. Management has a critical function in regulating and influencing the cost-profit balance. An effective manager must always be concerned with restaurant concerns such as inventory/stocking, pricing, order-taking, and much more. A restaurant's profitability can often grow or decline depending on how effectively it is handled. Using a well-developed system to run a restaurant reduces the liabilities of mismanagement and production gaps. The use of a Restaurant Management System to manage numerous business operations ensures that your restaurant is competitive, inventive, well-managed, and up to current on the newest management and business trends.

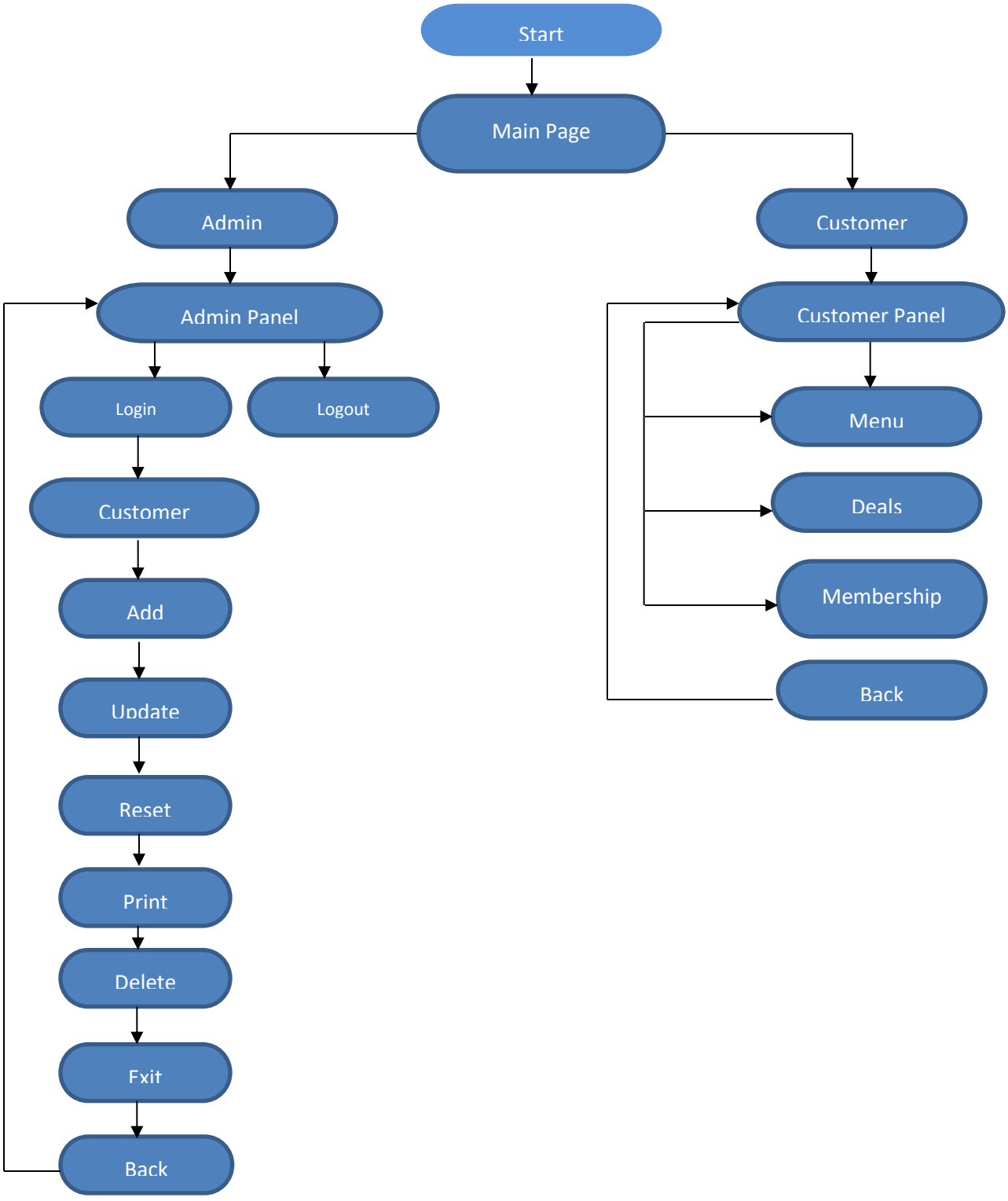
Problem and solution

The major goal is to increase profit by boosting efficiency and eliminating errors in the kitchen, all while maintaining customer pleasure. At the moment, many restaurants still utilize a paper-based system to communicate between the restaurant and the kitchen; this means of communication is one of the least efficient. However, while this strategy may be adopted and constructed in a successful profitable restaurant, there are a number of issues that may be viewed as diminishing the restaurant's efficiency, which are as follows:

- Handwriting causes a breakdown in communication.
- Unrestricted order logging (poor order taking).
- Ineffective communication between the restaurant and the kitchen.
- Order taking flaws and a lack of time management

So, in order to address these issues, we created a restaurant management system that focuses on the aforementioned issues. We attempted to create a user-friendly interface.

Workflow


Start
Main Page
Admin
Customer
Admin Panel
Customer Panel
Login
Logout
Menu
Customer
Deals
Add
Membership
Update
Back
Reset
Print
Delete
Exit
Back

Overview of project

The project plan in developing a computerized gui based system for silk route to handle billing, restaurant records was to include personnel participating in the billing process of a customer to storage of restaurant records and the ability to examine the information as required. Employees are given restricted access to protect the privacy and security of the documents. The database is kept up to date during the project.

Output (screen shots with details)

Figure 1

WELCOME PAGE

DETAILS: Welcome page is mainly for giving a better view to the viewer of our restaurant and from here we can access to admin and customer

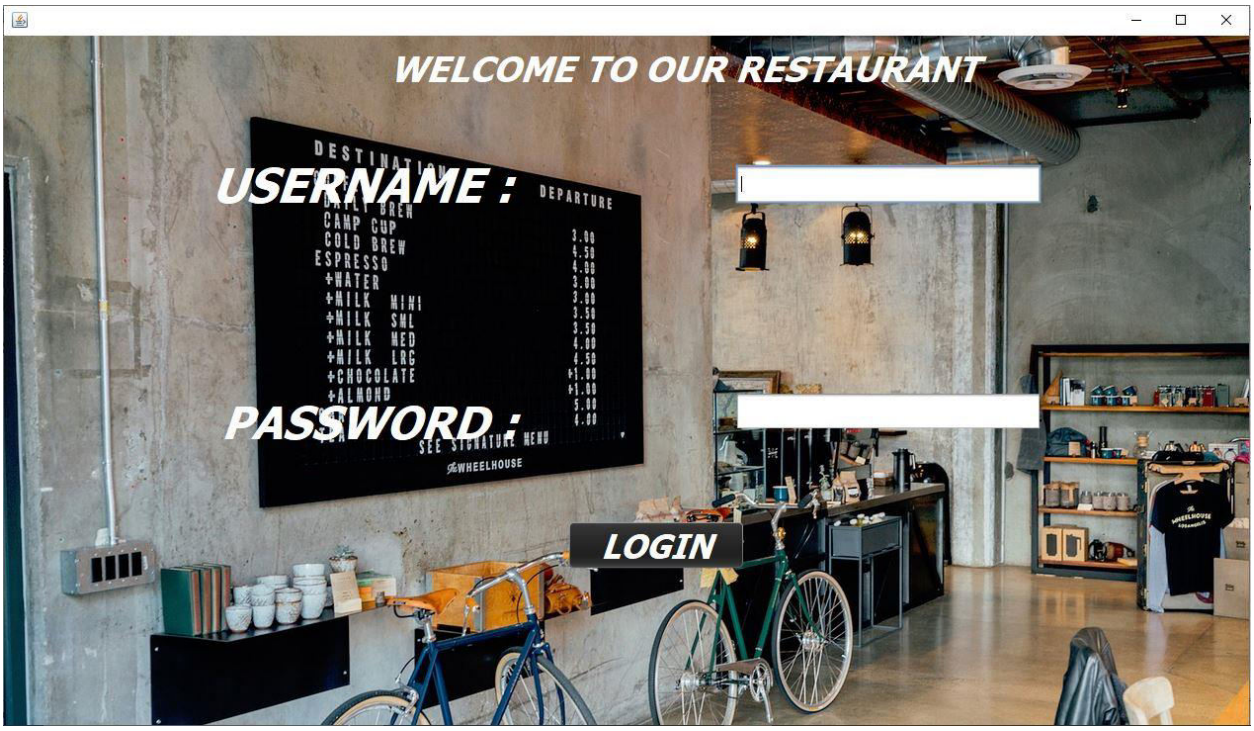


Figure 2

DETAILS: From this page we can login by entering right username and password.

Figure 3

DETAILS: From this page we can access in customer page and use it functions or if we want so we can logout as well.

Figure 4

DETAILS: This is a customer page from where can add new record of the customer and can update the previous one or can use buttons like reset, print, delete and exit to come out from this page.

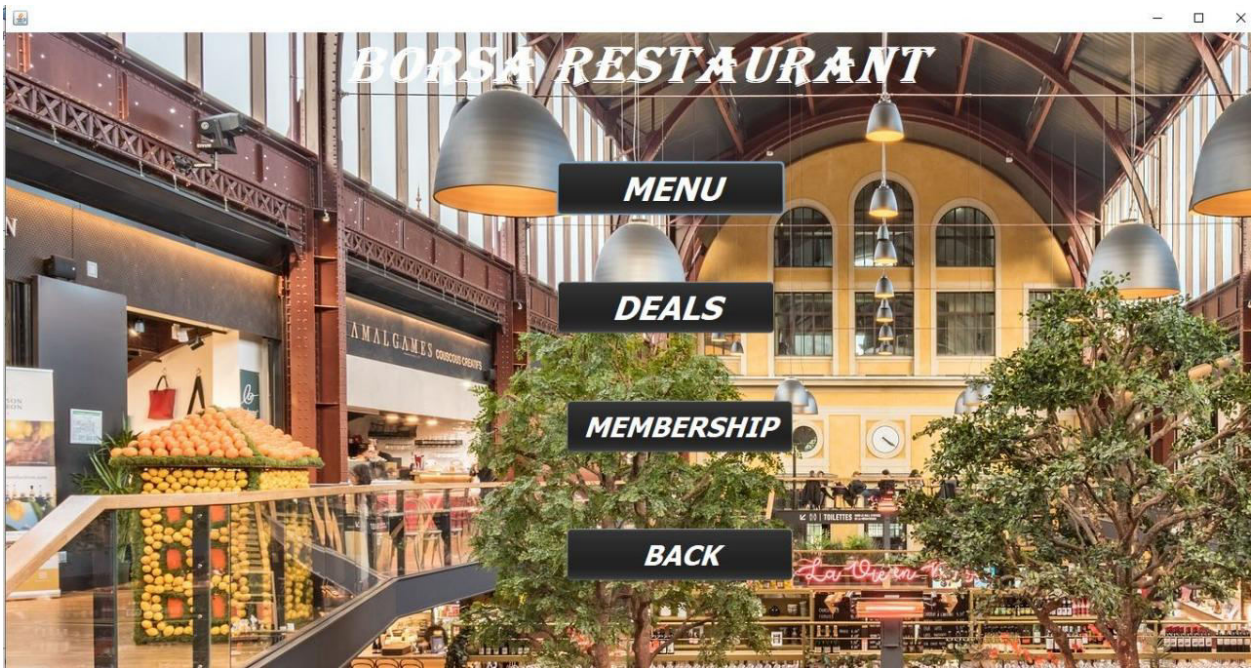


Figure 5

DETAILS: From this page we can access to customers friendly interface which includes menu, deals, membership and also a back button to return to the main page.

Figure 6

DETAILS: This page is showing us the deals which we are offered in our restaurant

SOURCE CODE:

```
private void jBut_ReceiptActionPerformed(java.awt.event.ActionEvent evt) {
//=================================================================
==
      int refs = 1325 + (int)(Math.random()*4328);

//=================================================================
==
      Calendar timer = Calendar.getInstance();
      timer.getTime();
      SimpleDateFormat tTime = new SimpleDateFormat("HH:mm:ss");
      tTime.format(timer.getTime());
      SimpleDateFormat Tdate = new SimpleDateFormat("dd-MMM-yyyy");
      Tdate.format(timer.getTime());
      Receipt_Panel.append("\n"+"\t       BORSA RESTAURANT \n" +
            "\n"+
            " References:\t\t\t" + refs +
            "\n=================================================\n" +
               " Meals:\t\t" + CostMealShow.getText()+ "\n\n"+
               " Drinks:\t\t" + CostDrinkShow.getText()+ "\n\n"+
             " Total Cost:\t\t" + TotalCostShow.getText()+ "\n\n"+
          "\n=================================================\n" +"\n\n"+
            " Tax:\t\t" + TaxShow.getText() + "\n\n" +
            " SubTotal:\t\t" + SubTotalShow.getText() + "\n\n" +
            " TotalShow:\t\t" + TotalShow.getText() + "\n\n" +
            "================================================="+
            "\n\n" +
```

```
            "Order Date : " + Tdate.format(timer.getTime())+"\n\n" +
            "Order Time : " + tTime.format(timer.getTime())+
            "\n\n\n\t            THANK YOU ");  }
```

SOURCE CODE:

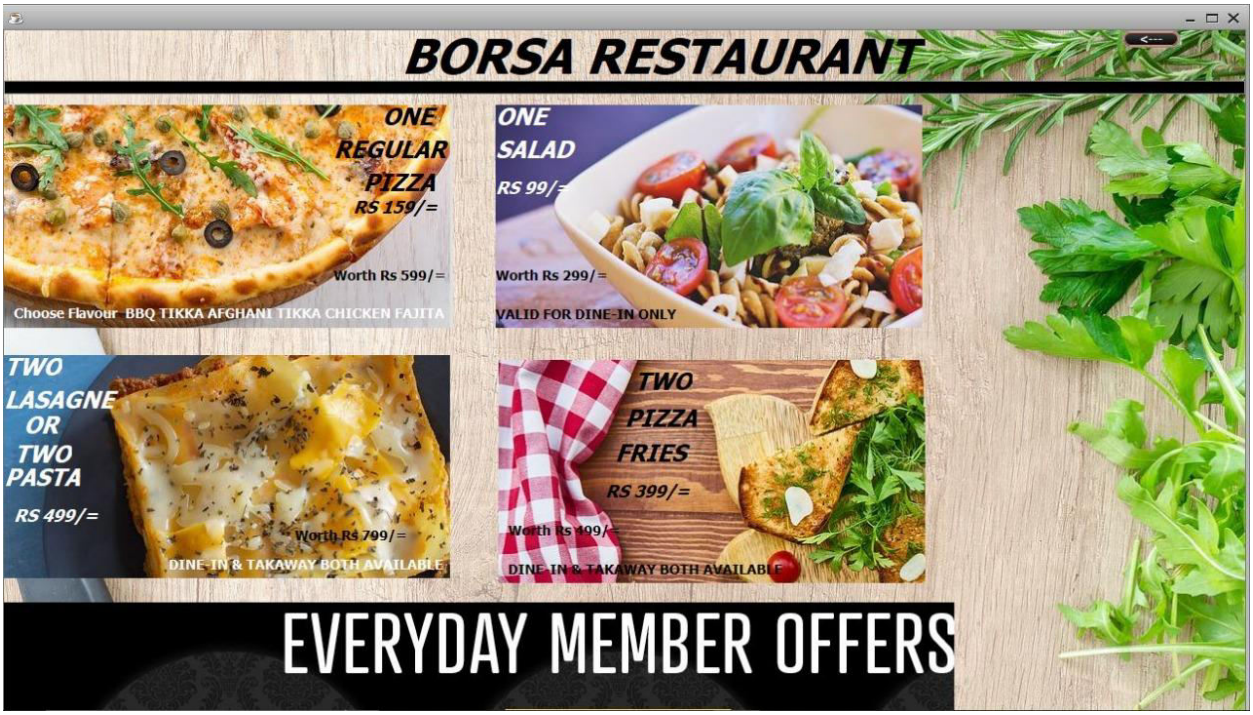
```
private void jBut_ResetActionPerformed(java.awt.event.ActionEvent evt) {
        //For Payment Receipt Reset Function//
        Receipt_Panel.setText(null);
        //For Payment Panel Reset
        JTextField clearText = (null);
        for (Component c: Payment_panel.getComponents()) {
           if (c.getClass().toString().contains("javax.swing.JTextField")) {
              clearText = (JTextField)c;
              clearText.setText("");
           }   }
        //For Cost Panel Reset Function//
        for (Component c: Cost_panel.getComponents()) {
           if (c.getClass().toString().contains("javax.swing.JTextField")) {
              clearText = (JTextField)c;
              clearText.setText("");
           }
            }
        //For Meal Panel Reset Function//
        for (Component c: Meal_Panel.getComponents()) {
           if (c.getClass().toString().contains("javax.swing.JTextField")) {
              clearText = (JTextField)c;
```

```
            clearText.setText("0");
        }
      }
      //For Drink Panel Reset Function//
      for (Component c: Drink_Panel.getComponents()) {
        if (c.getClass().toString().contains("javax.swing.JTextField")) {
            clearText = (JTextField)c;
            clearText.setText("0");
        }
      }
       }
   private void jBut_TotalActionPerformed(java.awt.event.ActionEvent evt) {
      double iTax, iSubTotal , iTotal;
      SubClass_Child ItemCost = new SubClass_Child();
      ItemCost.ZingerBurger = ItemCost.pZingerBurger * Double
.parseDouble(ZingerQty.getText());
      ItemCost.ZingerCheeseBurger = ItemCost.pZingerCheeseBurger * Double
.parseDouble(BurgerQty.getText());
      ItemCost.ChickenCheeseBurger= ItemCost.pChickenCheeseBurger * Double
.parseDouble(BurgMealQty.getText());
      ItemCost.ChickenLegend = ItemCost.pChickenLegend * Double
.parseDouble(ChLegQty.getText());
      ItemCost.BaconAndCheeseBurger = ItemCost.pBaconAndCheeseBurger * Double
.parseDouble(BacCheQty.getText());
      ItemCost.MilkShake = ItemCost.pMilkShake * Double .parseDouble(MilkQty.getText());
      ItemCost.SpecialOreoShake = ItemCost.pSpecialOreoShake * Double
.parseDouble(VanQty.getText());
```

```
        ItemCost.VanMilkShake = ItemCost.pVanMilkShake * Double
.parseDouble(VanMilkQty.getText());

        ItemCost.SpeciakShake = ItemCost.pSpeciakShake * Double
.parseDouble(ClasVan.getText());

        iSubTotal = ItemCost.getAmount();

        iTax = ItemCost.cFindTax(iSubTotal);

        iTotal = iSubTotal + iTax;

        String SubTotal = String.format("%.2f Rs", iSubTotal);

        SubTotalShow.setText(SubTotal);

        String Tax = String.format("%.2f Rs", iTax);

        TaxShow.setText(Tax);

        String Total = String.format("%.2f Rs", iTotal);

        TotalShow.setText(Total);

        String Meal = String.format("%.2f Rs", ItemCost.Meals);

        CostMealShow.setText(Meal);

        String Drink = String.format("%.2f Rs", ItemCost.Drinks);

        CostDrinkShow.setText(Drink);

        String TotalS = String.format("%.2f Rs", ItemCost.TotalofMD);

       TotalCostShow.setText(TotalS);    }

    private void jBut_BackActionPerformed(java.awt.event.ActionEvent evt) {

        Second_page sp = new Second_page();

        sp.show();

        this.dispose();

//   back on main page

//        TODO add your handling code here:

    }
```

```
    private void jBut_SaveActionPerformed(java.awt.event.ActionEvent evt) {
        try {
            // TODO add your handling code here:
            Receipt_Panel.print();
        } catch (PrinterException ex) {
}
}
```

Figure 7



DETAILS: From this page we can record the customer's order and can generate bill as well.

Figure 8

DETAILS: From this page customers can view which offers are available and can even avail if they want to.

SOURCE CODE:

```
private void createPopupMenu(JFrame frame){
    menuitem = new JMenuItem(
    "AVAIL ");
    menuitem.getAccessibleContext().setAccessibleDescription("AVAIL");
    menuitem.addActionListener(new ActionListener() {
       private Object JOptionpane;
       @Override
       public void actionPerformed(ActionEvent e) {
         JOptionPane.showMessageDialog(frame,"OFFER AVAILED");
       }
    });
     popupMenu.add(menuitem);
        menuitem = new JMenuItem(
    "CANCEL AVAIL");
    menuitem.getAccessibleContext().setAccessibleDescription("AVAIL");
    menuitem.addActionListener(new ActionListener() {
       private Object JOptionpane;

       @Override
       public void actionPerformed(ActionEvent e) {
         JOptionPane.showMessageDialog(frame,"YOU CANCELLED YOUR MEMBERSHIP");
       } });
     popupMenu.add(menuitem);  }
```

Conclusion

We were successful in developing a computerized gui based system for Silk Route to keep invoicing and restaurant data. This system can securely store billing records and simply retrieve them when needed. Along with the purchase and billing processes, this system includes data entry for customers and workers. Customers, restaurant data, and workers are all linked together to ensure the system's correctness. This system may also be improved by adding many more features and incorporating other systems. Finally, we feel that we were successful in introducing an excellent system to the restaurant, which will enable the restaurant to operate effectively in the future in terms of invoicing and restaurant records.