Green University

Restaurant Billing System

SUBMITTED BY

Md. Ashiqul Islam
ID : 212002056
Batch : 212
Section : D3
Subject : CSE (202)
Department : CSE

SUBMITTED TO

Ayesha Khatun
Lecturer : Department of CSE
Green University of Bangladesh

RESTAURANT BILLING SYSTEM

Table of Contents

Chapter 1
Introduction

1.1 Introduction

Restaurant management is the profession of managing a restaurant. It includes the major function of planning, organizing, staffing, directing, developing an attitude in food and beverage control systems and to efficiently and effectively plan menus at profitable prices, taking into consideration constraints, preparation and other variables affecting food and beverages outlets. Restaurant management is the profession of managing a restaurant. It includes the major function of planning, organizing, staffing, directing, developing an attitude in food and beverage control systems and to efficiently and effectively plan menus at profitable prices, taking into consideration constraints, preparation and other variables affecting food and beverage outlets. The Restaurant Billing Management Software program is a comprehensive restaurant management tool designed for foodservice management of all types. It is simple to learn and easy to use. This system processes transaction and stores the resulting data. Reports will be generated from these data which help the manager to make appropriate business decisions for the restaurant. For example, knowing the number of customers for a particular time interval, the manager can decide whether more waiters and chefs are required. Restaurant Software Systems are essential to the successful operation of most foodservice establishments because they allow the business to track transactions in real-time. food, they have to pay to confirm the order.

Features:

- It increases operational efficiency.
- It is designed to help you cost your recipes and track inventory saving your Money and Time and maximizing profit.

- It helps the restaurant manager to manage the restaurant more effectively and efficiently by computerizing Meal Ordering, Billing, and Sales Management.
- Accounting.
- It increases the security.
- It avoids paper work.
- It is Simple to learn and easy to use.
- It is portable.

1.2 Design Goals/Objective

Our goal is about make a Restaurant Management System which replace manual system of a restaurant to a digital system.

Chapter 2

Implementation of Restaurant Management System 2.1

The Restaurant Billing Management System project is divided into different modules for better understanding of the project. The modules help us to handle with errors easily and access each and every class properly. Menu management is done by using following

features:

- Add Menu items In this we Add Items in a menu by assigning a unique number to each item. When we add any item we give description as category, full name and price for each item. The access of the item is given to the Owner of the restaurant.

- Billing In this we can see the resultant bill of every costumer & **calculate total cost** of monthly without any mistake
- Get customer full details for next time when they come to our restaurant then we can **define them.**
- We know **how many customers** we get in our restaurant.
- If anyone do any mistake then replace or **upgrade** their Data anytime
- When need any kind of information about our customers then we can **search** easily for their user name & get the full details of the customer.
- Anytime, if we want then ,can **delete** data which are unnecessary .

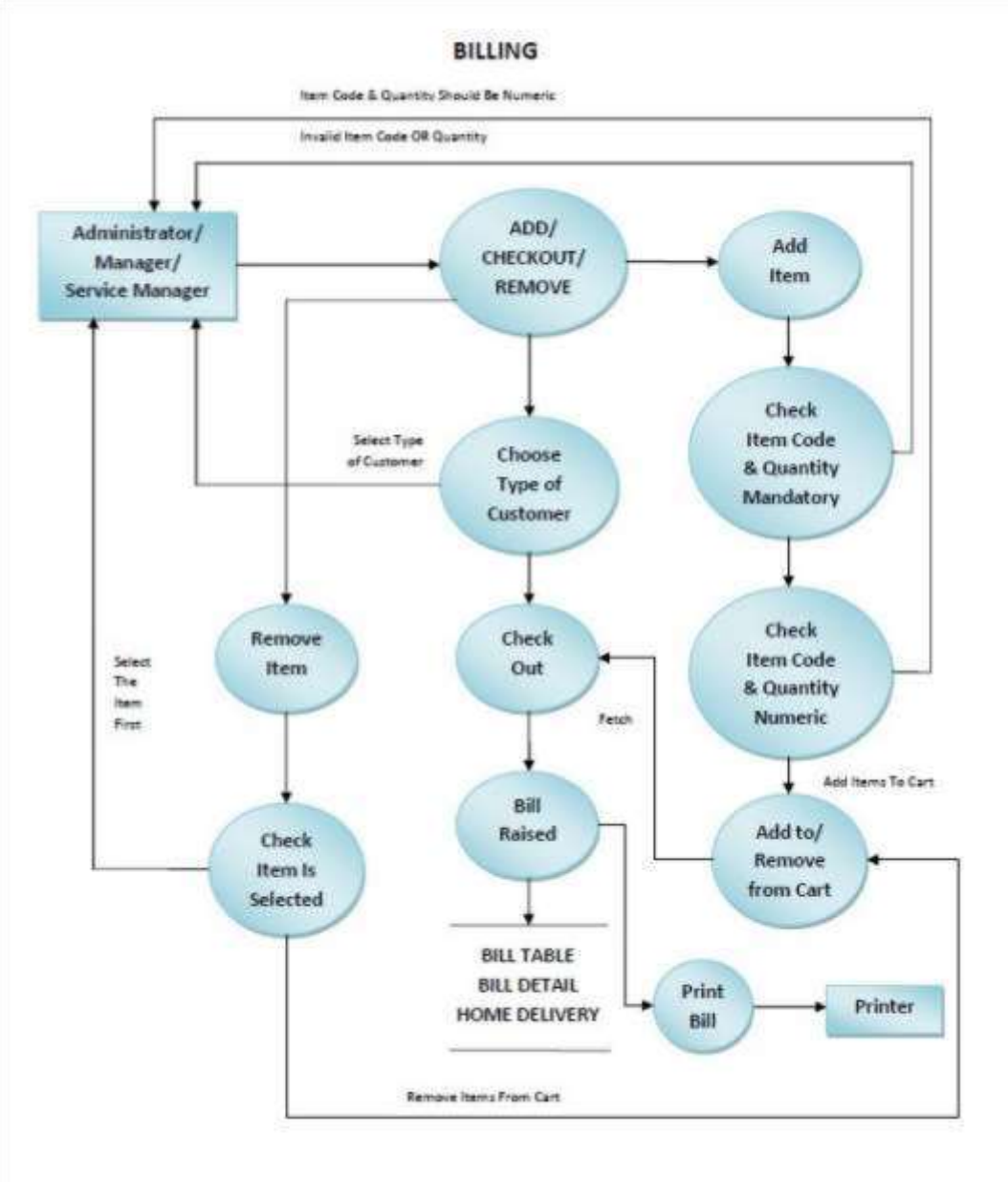

Figure 1.2

Requirement 2.2

- Computer
- OS (window, Linux, etc.)
- JFrame in NetBeans
- Knowledge about JAVA programming language

Figure 1.3

Instruction JFrame:

JFrame is a top-level container that provides a window on the screen. A frame is actually a base window on which other components rely, namely the menu bar, panels, labels, text fields, buttons, etc. Almost every other Swing application starts with the JFrame window. Unlike a frame, JFrame has the option to hide or close the window with the help of the method setDefaultCloseOperation(int).

History: NetBeans began in 1996 as Xelfi (word play on Delphi),[5][6] a Java IDE student project under the guidance of the Faculty of Mathematics and Physics at Charles University in Prague. In 1997, Roman Staněk formed a company around the project and produced commercial versions of the NetBeans IDE until it was bought by Sun Microsystems in 1999. Sun open-sourced the NetBeans IDE in June of the following year. Since then, the NetBeans community has continued to grow.[7] In 2010, Sun (and thus NetBeans) was acquired by Oracle Corporation. Under Oracle, NetBeans had to find some synergy with JDeveloper, a freeware IDE that has historically been a product of the company, by 2012 both IDEs were rebuilt around a shared codebase - the NetBeans Platform. In September 2016, Oracle submitted a proposal to donate the NetBeans project to the Apache Software Foundation, stating that it was "opening up the NetBeans governance model to give NetBeans constituents a greater voice in the project's direction and future success through the upcoming release of Java 9 and NetBeans 9 and beyond". The move was endorsed by Java creator James Gosling.[8] The project entered the Apache Incubator in October 2016.[9]

2.4 Required Code

This program is coded in JAVA language with swing and JLabels, JTextField, JButton, JComboBox and JTextarea ,The JLabel is used to display a short string or an image icon. JLabel can display text, image or both . JLabel is only a display of text or image and it cannot get focus. JTextField is a lightweight component that allows the editing of a single line of text. For information on and examples of using text fields. JButton class is used to create a labeled button that has platform independent implementation. The application result in some action when the button is pushed. It inherits AbstractButton class. JComboBox shows a popup menu that shows a list and the user can select a option from that specified list . JComboBox can be editable or read. JTextArea is a multi-line area that displays plain text

Implementation

```
import com.itextpdf.text.pdf.PdfWriter;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.print.PageFormat;
import java.awt.print.Printable;
import java.awt.print.PrinterException;
import java.awt.print.PrinterJob;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.JFileChooser;
import javax.swing.JOptionPane;

public class RESTAURANT_BILLING_SYSTEM extends javax.swing.JFrame {


    public RESTAURANT_BILLING_SYSTEM() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    Generated Code

    private void cb1ActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }

    private void cb2ActionPerformed(java.awt.event.ActionEvent evt) {...4 lines }

    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

        String Name=(String)tf1.getText();
        int Num=Integer.parseInt(""+tf2.getText());
        double com=Double.parseDouble(""+tf4.getText());
        double cod=Double.parseDouble(""+tf5.getText());
        double totalamt=Double.parseDouble(""+tf8.getText());
         tal.append("------------------GUB RESTUARENT-----------------------\n");
        tal.append("----------------------RECIEPT-----------------------------\n");
        tal.append("Name Of The Customer: " +Name+"\n");
        tal.append("\n");
        tal.append("Phone Number: "+Num+"\n");
        tal.append("\n");
        tal.append("Cost Of Meal: " +com+"\n");
        tal.append("\n");
        tal.append("Cost Of Drinks: "+cod+"\n");
        tal.append("\n");
        tal.append("Total Amount: " +totalamt+"\n");
        tal.append("\n");
```

```
        ta1.append("\n");
        ta1.append("----------------------RECEIPT-------------------------");
    }

    private void tf5ActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }

    private void tf4ActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }

    private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }

    private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        tf1.setText("");
        tf2.setText("");
        tf4.setText("");
        tf5.setText("");
        tf8.setText("");
        cb1.setSelectedItem("SELECT");
        cb2.setSelectedItem("SELECT");
        cb3.setSelectedItem("SELECT");
        cb4.setSelectedItem("SELECT");
        cb5.setSelectedItem("SELECT");
        cb6.setSelectedItem("SELECT");
        ta1.append("");
        qt1.setText("");
        qt2.setText("");
        qt3.setText("");
        qt4.setText("");
        qt5.setText("");
        qt6.setText("");
    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

        double cost=0.0,cost1=0.0,cost2=0.0,cost3=0.0,cost4=0.0,cost5=0.0;
        int qty1,qty2,qty3,qty4,qty5,qty6;
        String item=(String)cb1.getSelectedItem();
        if(item.equals("SELECT")){
            qty1=0;
            qt1.setText(""+qty1);
            cost=0.0;
        }
        else{
            qty1=Integer.parseInt(qt1.getText());
            if(item.equals("Khichuri")){
                cost=80.0;
            }
            else if(item.equals("Chicken Birany")){
                cost=120.0;
            }
            else if(item.equals("Beef Tehari")){
                cost=110.0;
            }
            else if(item.equals("Mutton kacchi")){
                cost=170.0;
            }
            else if(item.equals("Beaf Kacchi")){
                cost=140.0;
            }
        }
        String item1=(String)cb2.getSelectedItem();
        if(item1.equals("SELECT")){
            qty2=0;
            qt2.setText(""+qty2);
            cost1=0.0;
        }
        else{
            qty2=Integer.parseInt(qt2.getText());
            if(item1.equals("Mixed Chowmin")){
                cost1=180.0;
            }
            else if(item1.equals("Chicken fry with Fried rice")){
                cost1=150.0;
            }
            else if(item1.equals("Backed Pasta")){
                cost1=220.0;
            }
            else if(item1.equals("Nachos")){
                cost1 = 160.0;
            }
            else if(item1.equals("Pasta Basta")){
                cost1=250.0;
            }
        }
        String item3=(String)cb3.getSelectedItem();
        if(item3.equals("SELECT")){
            qty3=0;
            qt3.setText(""+qty3);
            cost2=0.0;
        }
        else{
            qty3=Integer.parseInt(qt3.getText());
            if(item3.equals(" Sandwich")){
                cost2=30.0;
            }
            else if(item3.equals("Chicken Burger")){
                cost2=80.0;
            }
            else if(item3.equals("Koliza Shingara")){
                cost2=20.0;
            }
            else if(item3.equals("Chicken Chomocha")){
                cost2 = 20.0;
            }
            else if(item3.equals("Chicken Roll")){
                cost2=25.0;
            }
        }
        //CODE FOR DRINKS----------
        String item4=(String)cb4.getSelectedItem();

        if(item4.equals("SELECT")){
            qty4=0;
            qt4.setText(""+qty4);
            cost3=0.0;
        }
        else{
            qty4=Integer.parseInt(qt4.getText());
            if(item4.equals("Pepsi")){
                cost3=25.0;
            }
            else if(item4.equals("Coca Cola")){
                cost3=25.0;
            }
            else if(item4.equals("Sprite")){
                cost3=25.0;
            }
            else if(item4.equals("Merinda")){
                cost3= 25.0;
            }
            else if(item4.equals("Borhani")){
                cost3=40.0;
            }
            else if(item4.equals("Mineral Water")){
                cost3=40.0;
            }
        }
        String item5=(String)cb5.getSelectedItem();
        if(item5.equals("SELECT")){
            qty5=0;
            qt5.setText(""+qty5);
            cost4=0.0;
        }
        else{
            qty5=Integer.parseInt(qt5.getText());
            if(item5.equals("Coffee")){
                cost4=20.0;
            }
            else if(item5.equals("Hot Chocolate")){
                cost4=100.0;
            }
            else if(item5.equals("Green Tea")){
                cost4=15.0;
            }
            else if(item5.equals("Tea")){
                cost4= 10.0;
            }
        }
        String item6=(String)cb6.getSelectedItem();

        if(item6.equals("SELECT")){
            qty6=0;
            qt6.setText(""+qty6);
            cost5=0.0;
        }
        else{
            qty6=Integer.parseInt(qt6.getText());
            if(item6.equals("Mango Juice")){
                cost5=50.0;
            }
            else if(item6.equals("Lemon Juice")){
                cost5=30.0;
            }
            else if(item6.equals("Orange Juice")){
                cost5=50.0;
            }
            else if(item6.equals("Red Grape Juice")){
                cost5= 70.0;
            }
            else if(item6.equals("Pineapple Juice")){
                cost5= 50.0;
```

```
            else if(item6.equals("Pineapple Juice")){
                cost5= 50.0;
            }
        }
        double totalmeal=cost*qty1 + cost1*qty2 + cost2*qty3;
        double totaldrinks=cost3*qty4 + cost4*qty5 + cost5*qty6;
        tf4.setText(""+totalmeal);
        tf5.setText(""+totaldrinks);

    }

    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        double amt1=Double.parseDouble(""+tf4.getText());
        double amt2=Double.parseDouble(""+tf5.getText());
        double totalamount=amt1+amt2;
        int totalamount1=(int)(totalamount);
        tf8.setText(""+totalamount1);

    }

    private void printMouseClicked(java.awt.event.MouseEvent evt) {


PrinterJob job = PrinterJob.getPrinterJob();
            job.setJobName("Print Data");

            job.setPrintable(new Printable() {
    @Override
    public int print(Graphics pg, PageFormat pf, int pageNum) throws PrinterException {
        pf.setOrientation(PageFormat.LANDSCAPE);
        if(pageNum > 0){
            return Printable.NO_SUCH_PAGE;
        }

        Graphics2D g2 = (Graphics2D)pg;
        g2.translate(pf.getImageableX(), pf.getImageableY());
        g2.scale(0.47,0.47);

        tal.print(g2);


        return Printable.PAGE_EXISTS;
    }
});
            boolean ok = job.printDialog();
```

```
PrinterJob job = PrinterJob.getPrinterJob();
            job.setJobName("Print Data");

            job.setPrintable(new Printable() {
    @Override
    public int print(Graphics pg, PageFormat pf, int pageNum) throws PrinterException {
        pf.setOrientation(PageFormat.LANDSCAPE);
        if(pageNum > 0){
            return Printable.NO_SUCH_PAGE;
        }

        Graphics2D g2 = (Graphics2D)pg;
        g2.translate(pf.getImageableX(), pf.getImageableY());
        g2.scale(0.47,0.47);

        tal.print(g2);


        return Printable.PAGE_EXISTS;
    }
});
            boolean ok = job.printDialog();
        if(ok){
        try{

        job.print();
        }
        catch (PrinterException ex){
        ex.printStackTrace();
}
        }

        // TODO add your handling code here:
    }

    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        Look and feel setting code (optional)

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new RESTAURANT_BILLING_SYSTEM().setVisible(true);
            }
        });
    }
```

Chapter 3

Performance Evaluation

Emulation Procedure

If we emulate our project, it will show Graphical window which are show some button and option choose option . Then we can choose any option which are need to us. Suppose a customer Ashiq who ate 1 chicken birany and a burhani then we press the option that window and save this information for future which are suitable for our restuarent.

Figure 1.1

Shown the graphical interface and their option.

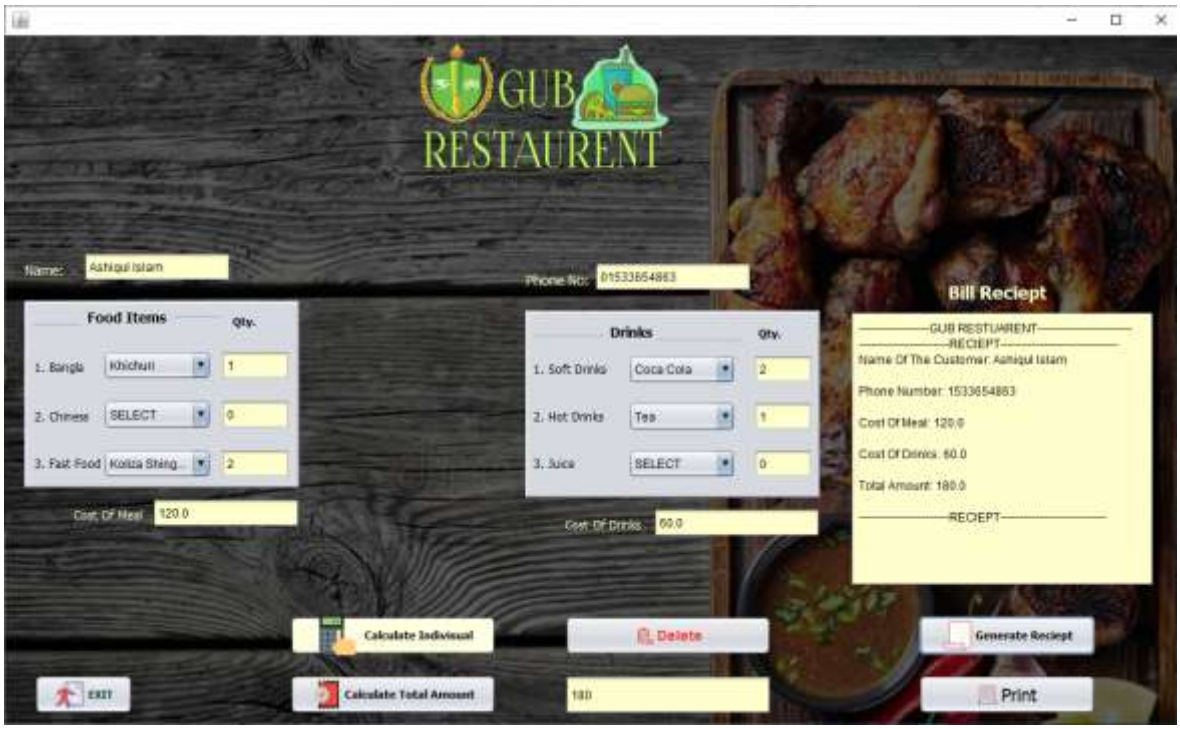

Figure 1.2

When we give the information and selection which are need to us then the interface creat a bill reciept

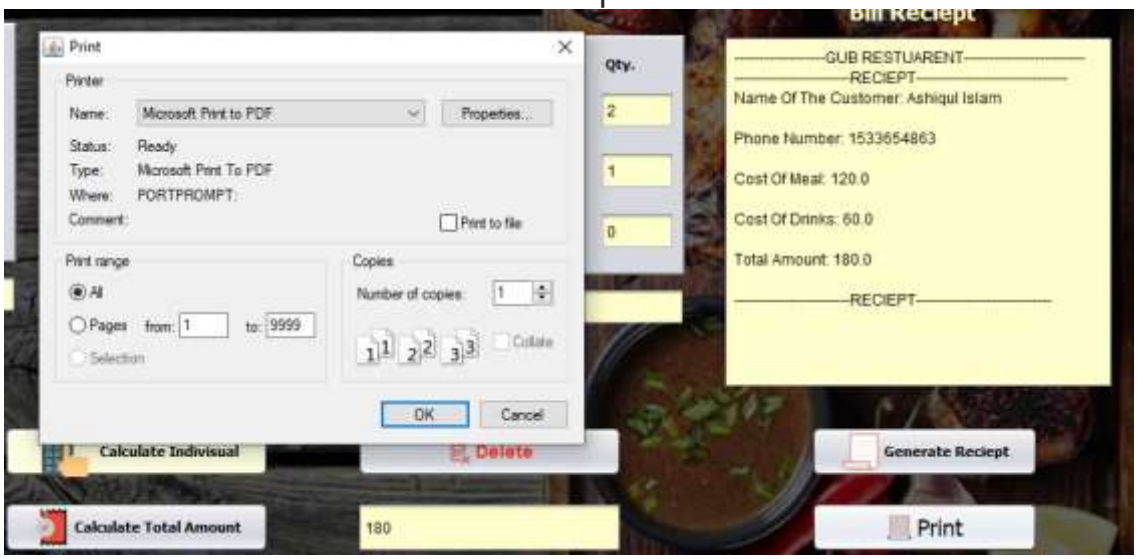

Figure 1.3

When we want to the bill print or save in my space then we press **Print** option and show the interface for convert the bill receipt in to pdf.

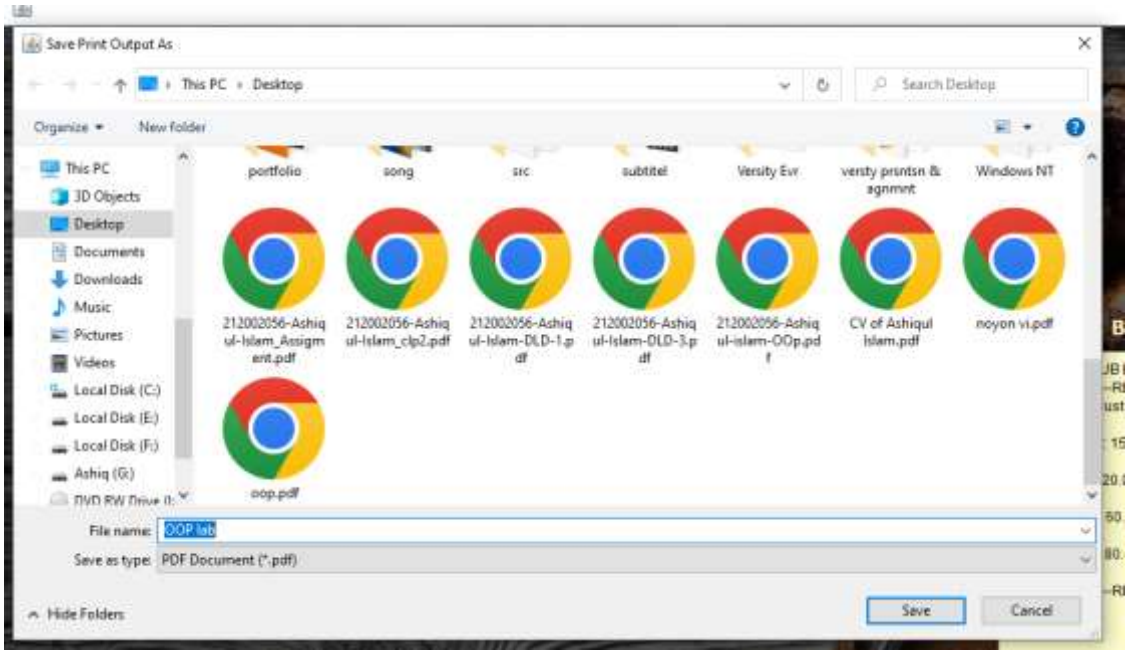

Figure 1.4

And then save the pdf in my desktop as a future documentary

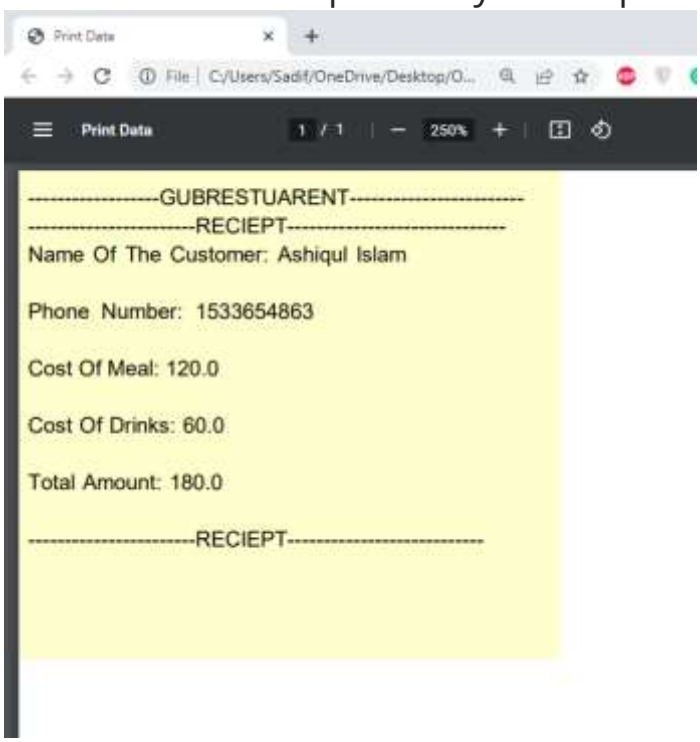

Figure 1.5

The pdf perfectly work and save my history.

Chapter 4

Conclusion

4.1 Introduction

In our Project entitled “RESTAURANT BILLING SYSTEM” we have tried our best to fulfill all the requirements of restaurant. The project being simple and flexible is running successfully. The main advantage of our project is that its simplicity.

The Restaurant Billing Management System helps the restaurant manager to manage the restaurant more effectively and efficiently by computerizing meal ordering, billing , remind data, upgrade data, and inventory control. The system processes transaction and stores the resulting data. Reports will be generated from these data which help the manager to make appropriate business decisions for the restaurant. For example, knowing the number of customers for a particular time interval, the manager can decide whether more waiters and chefs are required. This project when implemented it will remove all the security issues. Also, there will be speedy and secured authentication procedure for the maintenance of records. Data entry is fast and simple. Therefore, our software will definitely prove to be a successful stepping stone in replacing the outdated manual method of maintaining secure records. The work plan also includes the detailed features of the technology used in the project defining the front end and back end. The objectives and scope of the project in future have been elaborated

4.2 Limitation

Our project with so many facilities but our system has some lacking in it. it cannot connected any online e-pay services. It cannot back the option which are chosen The system takes 25-28 second for run this a major fact.

4.3 scope of Future Work

We try to in future remove our project lack, we try add 2 main interface . 1 are for restaurant admin & 2nd interface for customer . Where any of customer chose & order there food at home or restaurant at anytime. And they can payment their bill by this app.

References

www.google.co.in

www.yahoo.com

www.wikipedia.co.in www.ebooks.in