

Restaurant Management System

Om Uikey

*Department of Electronics and
Telecommunication*
Vishwakarma Institute of Technology
Pune, India
om.uikey21@vit.edu

Roshan Thorat

*Department of Electronics and
Telecommunication*
Vishwakarma Institute of Technology
Pune, India
roshan.thorat21@vit.edu

Mayur Tidake

*Department of Electronics and
Telecommunication*
Vishwakarma Institute of Technology
Pune, India
mayur.tidake21@vit.edu

Sahil Ukey

Department of Electronics and Telecommunication
Vishwakarma Institute of Technology
Pune, India
sahil.ukey21@vit.edu

Sheela Chinchmalatpure

Department of Electronics and Telecommunication
Vishwakarma Institute of Technology
Pune, India
sheela.chinchmalatpur@vit.edu

Abstract— This paper presents the design and implementation of a computer-based Restaurant Management System (RMS) that automates various aspects of restaurant management, including order-taking, employee management, and employee scheduling. The RMS was developed in response to the challenges faced by restaurant managers in managing their businesses, including the need for improved efficiency, cost savings, and enhanced customer experience. The system consists of several modules, including a user interface for order-taking, a database for storing customer and order information, an inventory management system, and a scheduling system for managing employee shifts.

Keywords— Java , GUI, MySql, Database, Swing, Awt.

I. INTRODUCTION

The introduction of the paper will likely provide an overview of the importance of RMS in the food service industry and the challenges faced by restaurant managers in managing their businesses. It may also discuss the benefits of using an RMS, including improved efficiency, cost savings, and enhanced customer experience.

The paper will likely describe the methodology used to design and implement the RMS and detail the different modules or components of the system. This could include a user interface for order-taking, a database for storing customer and order information, an inventory management system, and a scheduling system for managing employee shifts.

II. Literature Review.

Wesley Williams and Devon M. Simmonds had attempted to make A Case Study in the Design of a Restaurant Management System[1]. This paper reports on the development of a restaurant management system as part of a first course in software engineering.

Carl Abernethy had built the Restaurant Management System [2]. This report documents the process of designing, developing and testing a software system to be used in a restaurant; usually given the name restaurant management system.

Adam Huda Nugraha has made The Cashier Application For The Tongue Wing Meatball Restaurant Using Mysql And Netbeans[3]. In this scientific writing, the author discusses the difficulties in the data collection process, calculating the total price and checking data in a restaurant that still uses the manual method, which is why the author creates this application that aims to simplify and facilitate entrepreneurs help build their business from mistakes or even losses that can be detrimental to both the customer and the restaurant manager.

Shreeja T H and Vigneshwar S have made an Integrated Restaurant Service using POS System[4]. This paper provides a short introduction to the Integrated Restaurant Service System and each of its major modules. Besides, explaining the major modules we are also going to provide some details about the working model which is made using Java Swing and JDBC and then we are going to discuss the benefits of them over the recent system.

[5]There are three main actors in this application: Admin, Restaurant manager and User. The admin will create all menus with their corresponding price lists. Also, the admin will be able to view statistical reports for foods. Restaurant managers will receive orders from customers.

Users/Customers can place orders using this application and they will be able to view available food items. The system is designed as a 3-tier architecture. Each user must register with the system; after logging in successfully, they are able to perform operations. Separate login pages are provided for logging into the system. To develop this system, we use HTML, CSS, JavaScript, and bootstrap for the front end, MySQL database as the back end, and Java, JSP, and JDBC as middleware.

III. METHODOLOGY

Import the necessary packages: The first step in creating a Java Swing GUI is to import the necessary packages. In this case, we import `javax.swing.*` and `java.awt.*` to get access to the various Swing and AWT classes we need to create our GUI.

Create a class for the GUI: Next, we create a class for our GUI called `Miframe`. This class extends the `JFrame` class, which is the main container for our GUI.

Set the frame properties: We set the title of the frame to "Restaurant Management System" using the `setTitle()` method. We also set the size of the frame to 600 pixels by 400 pixels using the `setSize()` method. Finally, we set the layout of the frame to a `BorderLayout` using the `setLayout()` method.

Create a menu bar: We create a `JMenuBar` object called `menuBar` to hold our menu. We then create a `JMenu` object called `fileMenu` to hold our "File" menu. We create a `JMenuItem` object called `exitMenuItem` to hold our "Exit" menu item. We add the "Exit" menu item to the "File" menu, and we add the "File" menu to the menu bar using the `add()` method.

Create a content panel: We create a `JPanel` object called `contentPanel` to hold the main content of our GUI. We set the layout of this panel to a 2x2 grid using the `GridLayout` class. We create a `JLabel` object called `titleLabel` to display the title of our GUI. We create three `JButton` objects called `addButton`, `removeButton`, and `editButton` to provide functionality for our restaurant management system. We add these components to the `contentPanel` using the `add()` method.

Add components to the frame: We add the `menuBar` to the top of the frame using the `add()` method with the `BorderLayout.NORTH` argument. We add the `contentPanel` to the center of the frame using the `add()` method with the `BorderLayout.CENTER` argument.

Set the frame to be visible: We set the visibility of the frame to true using the `setVisible()` method.

Set the close operation of the frame: We set the close operation of the frame to `JFrame.EXIT_ON_CLOSE` using the `setDefaultCloseOperation()` method.

Create a main method: We create a `main()` method to create an instance of the `RestaurantManagementSystemGUI` class.

RESULTS AND DISCUSSIONS

The Restaurant Management System project starts from the register page and if users have already registered then the 2nd GUI is open which is Log in page. After logging-in the home page is opened which contains three blocks which are Menu which shows menu list, Order which is used to place order and Reservation to reserve table in restaurant. This project also contains one GUI for Managing menu items which can be accessed by admin only. Here are the results of this project.

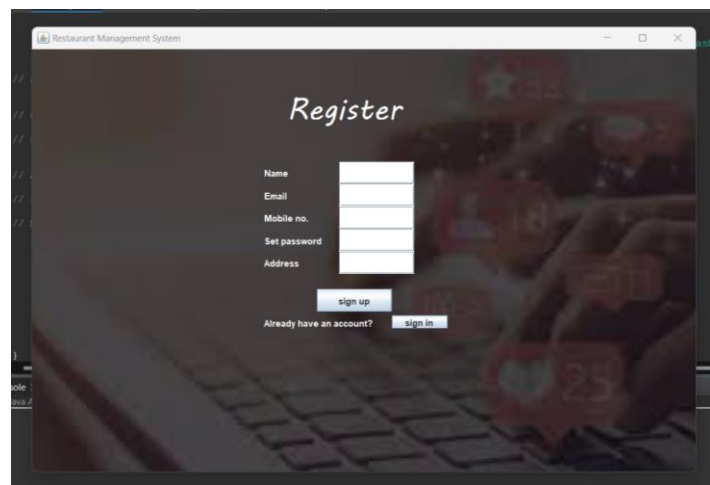


Fig.no.1 Registration page

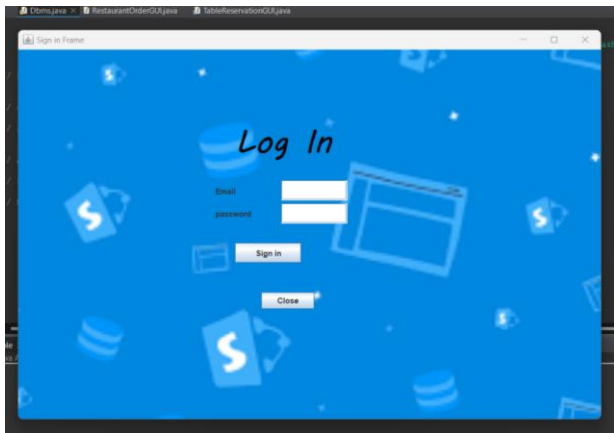


Fig.no.2 Login page if user have already registered

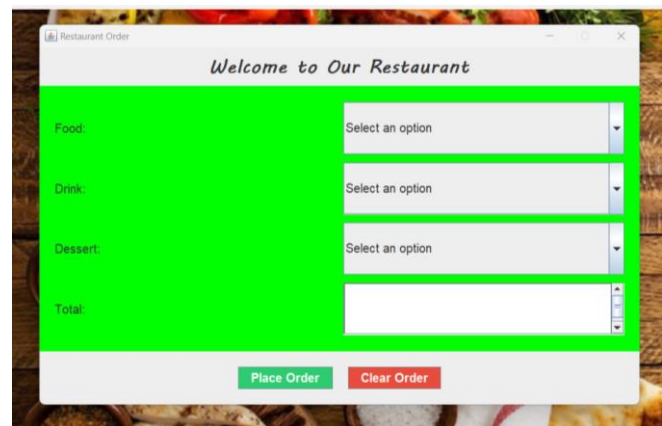


Fig.no.5 Order page opens after clicking on Order block

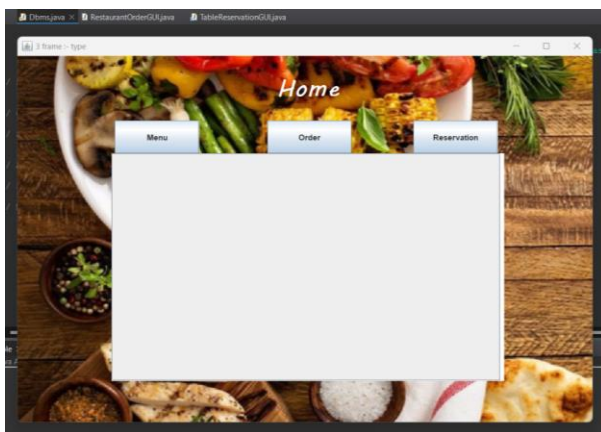


Fig.no.3 Home page which opens after registration or log in

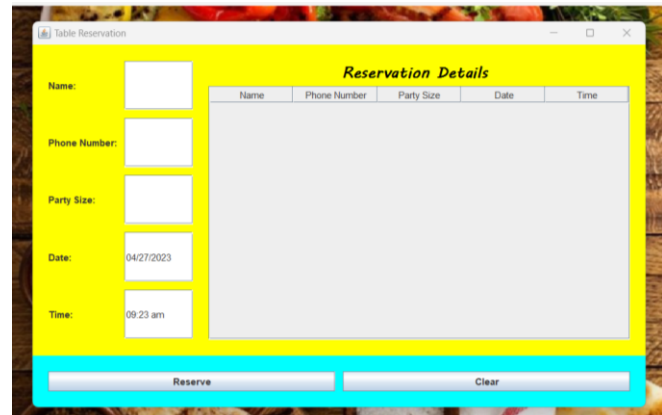


Fig.no.6 Table reservation page opens after clicking on Reservation block

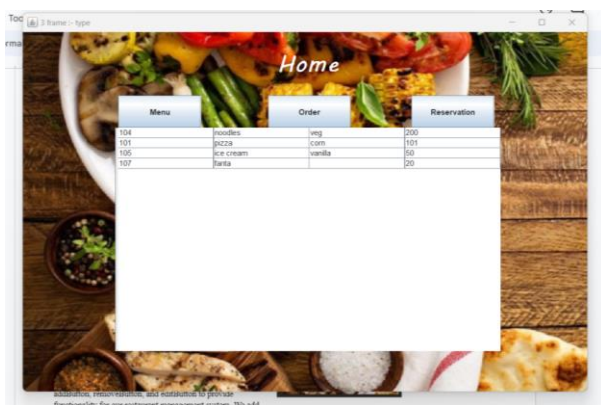


Fig.no.4 Menu list when click on Menu block

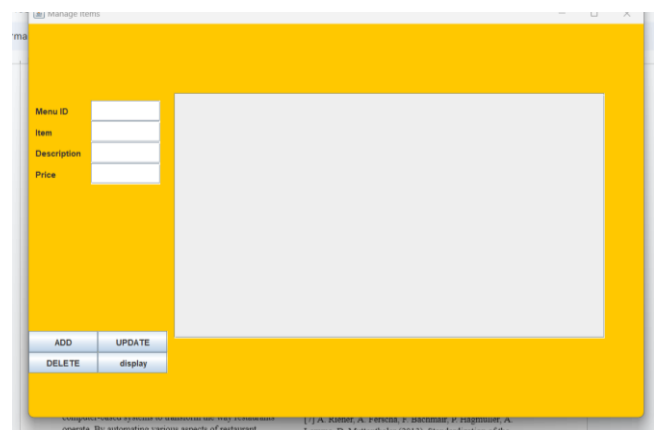


Fig.no.7 Manage Menu item page opens after logging in as admin

CONCLUSION

In conclusion, the Restaurant Management System (RMS) presented in this paper demonstrates the potential for computer-based systems to transform the way restaurants operate. By automating various aspects of restaurant management. This suggests that RMS can play a critical role in improving the profitability and competitiveness of restaurants in today's fast-paced food service industry. However, the success of an RMS depends on factors such as the quality of the software, training and support for users, and the willingness of managers to embrace technology. Future research should focus on evaluating the long-term impact of RMS on restaurant operations and customer satisfaction, as well as exploring opportunities to integrate emerging technologies such as artificial intelligence and blockchain. Overall, the RMS offers a promising solution to the challenges faced by restaurants today and represents a significant opportunity for innovation in the food service industry.

REFERENCES

- [1] Wesley Williams & Devon M (2010). Simmonds. A Case Study in the Design of a Restaurant Management System.
- [2] Carl Abernethy (2010). The Restaurant Management System.
- [3] Adam Huda Nugraha (2021). The Cashier Application For The Tongue Wing Meatball Restaurant Using Mysql And Netbeans.
- [4] Shreeja T H & Vigneshwar S (2019). Integrated Restaurant Service using POS System.
- [5] Prudveer Karne, Bachelor of Science (MECs), Osmania University.(2017) MANAGEMENT SYSTEM FOR A RESTAURANT.