# MINI PROJECT REPORT ON



**“Food Ordering System”**

BY

# KOMAL SUPEKAR [24]

**SHUBHAM NADHAVALE [57]**

**PREM**

**PAWAR [60]**

**UNDER GUIDANCE OF**

**Mr. RAHUL DIWATE**

**DEPARTMENT OF COMPUTER ENGINEERING ALL INDIA SHRI SHIVAJI MEMORIAL SOCIETY’S INSTITUTE OF INFORMATION TECHNOLOGY, PUNE 411011 SAVITRIBAI PHULE PUNE UNIVERSITY 2020-21**

**All India Shri Shivaji Memorial Society’s Institute of Information Technology Department of Computer Engineering**



## CERTIFICATE

This is to certify that **Miss. Komal Supekar** from **final year** 1st Shift Computer Engineering has successfully completed her mini project work titled “**Food Ordering system**” at All India Shri Shivaji Memorial Society’s Institute of Information Technology, Pune in the partial fulfillment of the mini project.

Mr. Rahul Diwate Internal Guide

Stamp of college

Dr. S. N. Zaware

Place: PUNE Head of the Department

Date: Computer Engineering



**All India Shri Shivaji Memorial Society’s Institute of Information Technology Department of Computer Engineering**

## CERTIFICATE

This is to certify that **Mr. Shubham Nadhavale** from **final year** 1st Shift Computer Engineering has successfully completed her mini project work titled “**Food Ordering system**” at All India Shri Shivaji Memorial Society’s Institute of Information Technology, Pune in the partial fulfillment of the mini project.

Mr. Rahul Diwate Internal Guide

Stamp of college

Dr. S. N. Zaware

Place: PUNE Head of the Department

Date: Computer Engineering

**All India Shri Shivaji Memorial Society’s Institute of Information Technology Department of Computer Engineering**



## CERTIFICATE

This is to certify that **Mr. Prem Pawar** from **final year** 1st Shift Computer Engineering has successfully completed her mini project work titled “**Food Ordering system**” at All India Shri Shivaji Memorial Society’s Institute of Information Technology, Pune in the partial fulfillment of the mini project.

Mr. Rahul Diwate Internal Guide

Stamp of college

Dr. S. N. Zaware

Place: PUNE Head of the Department

Date: Computer Engineering

# ABSTRACT

In a highly competitive restaurant business, giving customers what they want at the click of their fingers is of great advantage. “Food Ordering System” is an online desktop application for delivering food in restaurants.

In this project, we implemented a dynamic food ordering system using Java RMI (Remote Method Invocation), which is a powerful technology for distributed object computing. This system wake to provide service facility to hotel or restaurant. The service which provided is food ordering by customer and also administration. It includes customer management, food information management, chief information management, menu information management and report. Main objective is to build the system which provide ordering system by online also providing the status of order to that customer. with this system ordering become easier and which avoid paper work totally. Proposed system implemented using RMI(Remote Method Invocation) which will access anywhere easily. Distributed models can be contributed by this model.

# ACKNOWLEDGEMENT

We would like to express our deepest appreciation to all those who provided the

possibility to complete this report. A special gratitude we give to our seminar guide, Prof. Rahul Diwate, whose contribution is stimulating suggestions and encouragement, help me to coordinate our project. I would also like to acknowledge with much appreciation to crucial

role of the staff, who gave the permission to use all required equipment and the necessary material to complete the task.

Last but not least, many thanks to Prof. Smita Zambre whose motivation and support is helpful for us. Also, thanks to all supervisors as well as panel.

.

Miss. Komal Supekar

Mr. Shubham Nadhavale

Mr. Prem Pawar

AISSMS IOIT, Pune

# INDEX

|  |  |  |
| --- | --- | --- |
| **Sr Number** | **Name of Content** | **Page Number** |
| 1 | Introduction | 1 |
| 2 | Aim | 1 |
| 3 | Objectives | 1 |
| 4 | Scope of Project | 1 |
| 5 | Motivation | 2 |
| 6 | Need of Project | 2 |
| 7 | Literature Review | 3 |
| 8 | System Requirements | 4 |
| 9 | Data modelling | 5 |
| 10 | Implementation | 10 |
| 11 | Debug and Test case | 17 |
| 12 | Conclusion | 19 |
| 13 | References | 20 |

**List of Figure**

1. RMI module
2. Data flow diagram
3. Activity Diagram
4. ER Diagram
5. Use case diagram

# CHAPTER 1

## INTRODUCTION

1. Typically, in a restaurant food order process involves several steps for ordering the food where firstly customer starting from browsing the paper-based menu and then inform to the waiter for ordering items. Usually, the process require that the customer has to be seated before starting “Food Ordering System” is an online desktop application for delivering food in restaurant. It enables the clients to view menus of different restaurants, query food items, place order for food, and view the orders placed through a single web address. The customer can check menu list, and can place order as for delivery system or pickup system. these selected pre-ordered items shall be shown to chief screen, and when confirmed, order slip shall be printed for further order processing Every process is on a remote system and can be registered in the RMI Registry of the virtual restaurant at runtime. For obtaining references to the remote objects, i.e., the module objects, the ordering system provides a remote object registry interface, which includes methods for storing and retrieving remote object references. In this proposed system restaurant admin receives requests from users and invokes the corresponding services provided by the systems using RMI and displays the information to the clients. Furthermore, the ordering system also provides additional features, for example, the customers can check the status of order by their phone, checks all items with same or different prices, and place an order of their choice. Based on a list food menu given, the user should be able to purchase food by selecting from the list above. Upon confirmation, the user cash balance from his/her account should be deducted from the food price. It should not be possible to purchase the food if the cash is not sufficient from the user account.

## Aim

## To provide online food delivery System using RMI

## Objectives

* + - Provide way to place the order at restaurant over internet
    - Avoid paper work
    - Maintain customer flow
    - To ensure easier and faster service.
    - To decrease the waiting time to take an order

## Scope of the Project

* + - In Proposed system there are Customer, chef, stock Manager and cashier Modules.
    - Cashier allows new registration to place the order and handle enquiry section.
    - Chef access the items and gives status of order.
    - Stock Manager maintains flow of whole order system and information of available items.

## Motivation

* + - Food Order system helps to provide pickup as well as parcel delivery service.
    - Maintains customer workflow which helps for social distancing
    - Through this, proposed system can easily maintain the food ordering process by using distributed platform.

## Need of Project

* + - Due to COVID-19 Situation to avoid social gathering proposed system can be implemented.
    - Customers can easily check special item of the day and available items at specific time and there is no need to visit restaurant, as online delivery also available.

# CHAPTER 2

# Literature Review

**Existing Food Order Process**

1. **FULL-SERVICE RESTAURANT**: Tradition food order process used in most full-service restaurants starting when a waiter brought the guests the paper-based menu, and then waiting for the guests to choose items from the menu and inform the waiter the order items. The process typically required the guests to be seated in the restaurant and a waiter to assist the ordering. One of the most widely used food ordering system is the conventional paper-based system. In this system all records are stored on paper. The main drawback of this system is papers can get easily lost or damaged. There is also wastage of money, time and paper. Paper-based systems do not provide any form of dynamicity. Even a small change requires the re-print of entire menu-card. Also, large number of human efforts are required, this system is not work properly because it has some error and from a customer’s point of view it is time consuming.
2. **SELF SERVICE RESTAURANT**: This process required the guests to place order at the service counter in the restaurant by using desktop app. The guests shall have decision in advance, before presented at the counter, of which menu items to order and it will be picked up or has to deliver. Menu catalog is mostly presented as posters placing behind the order counter.
3. **Remote Method Invocation (RMI):**  Remote method invocation is a mechanism that allows an object residing in one system (JVM) to access/invoke an object running on another JVM. **RMI** is used to build distributed applications; it provides remote communication between Java programs. It is provided in the package java.

# Chapter 3

## REQUIREMENTS AND SPECIFICATION

### 3.1System Requirements

To implement this proposed system following software and hardware are required.

### 3.1.1Hardware Requirements (Minimum)

* + - * 2 GB RAM
      * 2.80 G, Core 2duo Processor

### 3.1.2Software Requirements

* + - * IDE: Sublime
      * Browser: Chrome /Firefox
      * Toolkit: JDK
      * Operating System: Windows 7 above, Linux/Ubuntu

### 3.1.3Technology Stack

* + - * RMI (Remote Method Invocation)
      * Front End- Java Swing

### 3.2Specification

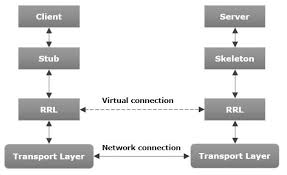
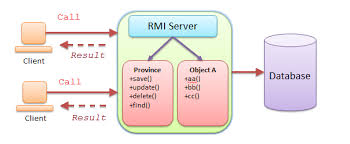
1. **Java Swing**

* Swing is a GUI widget toolkit for Java
* It is part of Oracle's Java Foundation Classes – an API for providing a graphical user interface for Java programs.
* Swing was developed to provide a more sophisticated set of GUI components than the earlier Abstract Window Toolkit
* Swing components are not implemented by platform-specific code
* provides a layer of abstraction between the code structure and graphic presentation of a Swing-based GUI

### RMI

Remote method invocation is a distributed object model for the Java programming language that making distributed objects easy to implement and tousle. Remote method invocation applications are often comprised of two separate programs: a Server and a Client.

A typical server application creates a number of remote objects, makes references to those remote objects accessible, and waits for clients to invoke methods on those remote objects. In view of the needs of the organization in business and cross-platform of the system, the distributed object technology and component technology based on the Java platform is an ideal choice of the large-scale distributed application system of the development of the enterprise. With respect to the current popular Desktop based distributed application system, from the point of the software design, this article puts forward the distributed architecture based on the Java remote method invocation technology, and in the actual design and development have got more satisfactory result.

  **Fig:3.1RMI Modules**

# CHAPTER 4

## Data Modeling

Designing gives us a view and a direction as to how the project would look and what are the steps to be followed during the creating phase. Designing aims at integrating the available resources in a manner that is best suited towards project development. It gives an overview as to the details of the project, the way the resources have to be used and what are the designing complications that can arise and what things are to be taken care of while making the project.

### 4.1Activity Diagram

### It shows the working process of the modules such as customer, waiter, and chef.

### Actual data flow represented by this diagram.

### Data flow diagram, activity diagram, sequence diagram

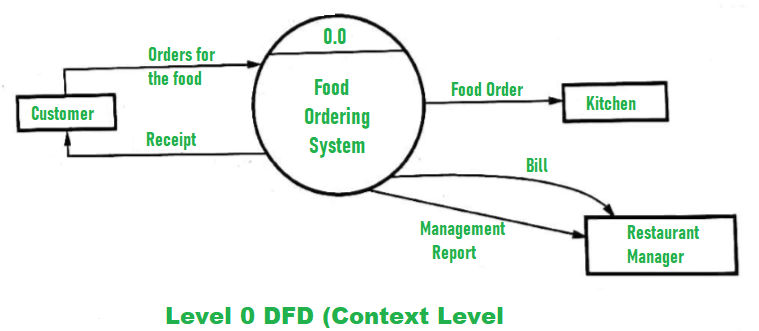
### Figure:4.1. Activity diagram

### 4.2Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel.

### 4.2.1DFD Level 0

DFD for level 0 of proposed system show the actors that is central ordering system, customer, stock manager, chef (food preparation) used to process the food delivery.

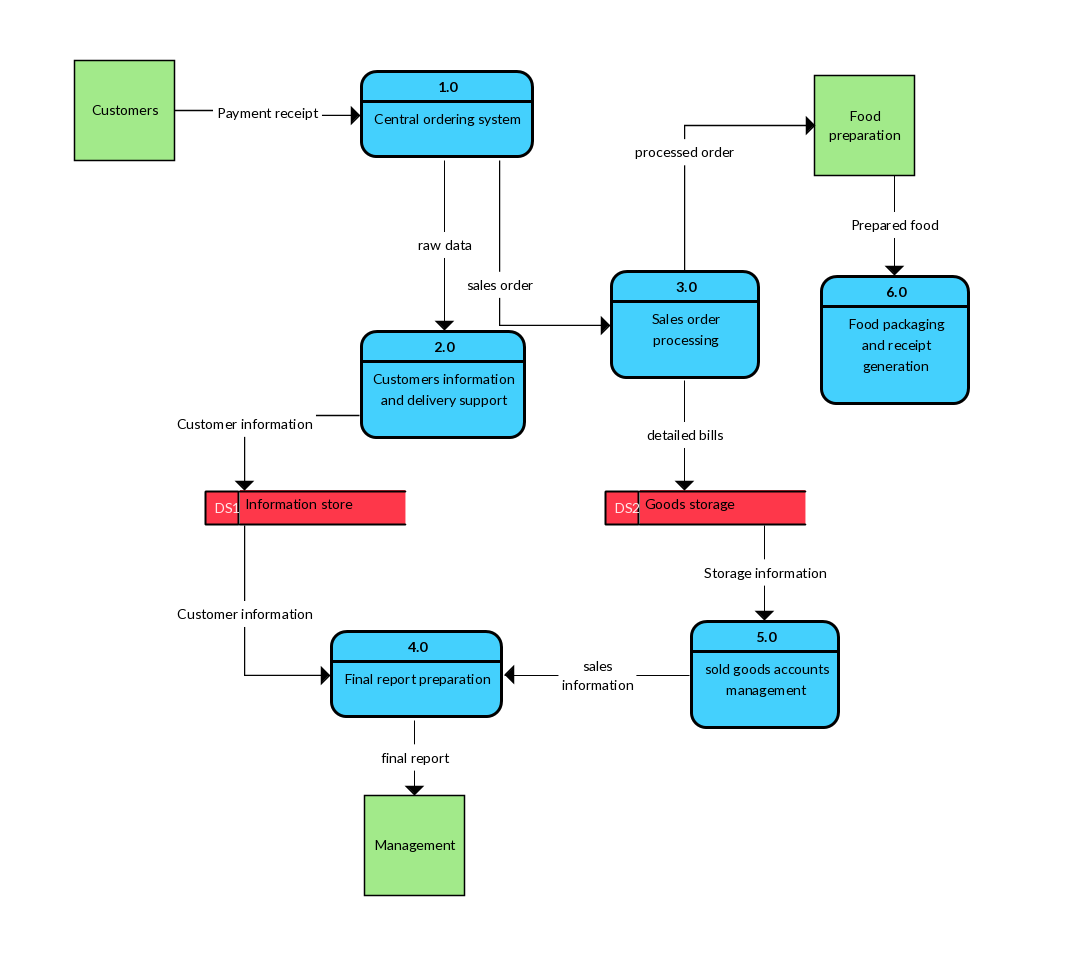


**Figure No.3.2 : DFD Level 0**

As shown in fig 3.2data will flow from customer to the system, for this customer have to register in the system and fill the basic information and the menu which she/he want.

### 4.3.2DFD Level 1:

Level 1 DFD’s aim to give an overview of the full system. They look at the system in more detail. Major processes are broken down into sub-processes.

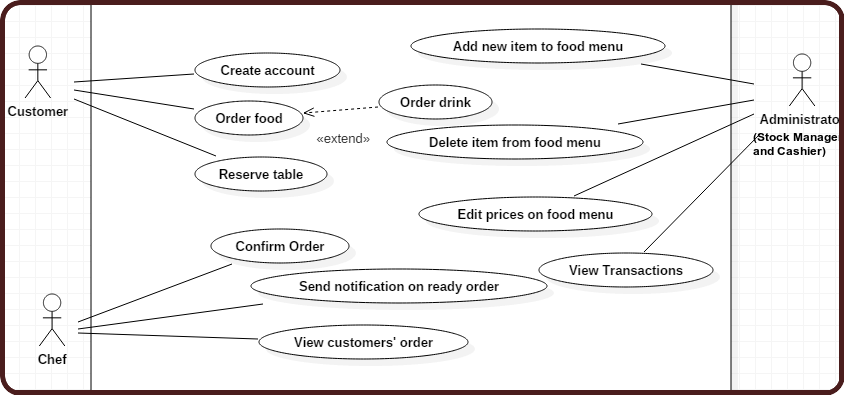


**Figure No.3.3:DFD Level 1**

### 4.4Use Case Diagram:

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. Use case diagrams describe what a system does from the view point of an external observer. The emphasis of use case diagrams is on what a system does rather than how it works. Use case diagram are used to show the interactions between users of the system and the system. A use case diagram represents the several users called actors and the different ways in which they interact with the system.

### Actors: Customer, Chef, Stock manager, Cashier



### Figure No. 4.4: Use Case Diagram of Food Ordering System

**4.1.1Use Case Diagram of Chef:**

Use case diagrams are used to describe behavior of actions that or systems should or can perform in collaboration with one or more external users of the system. This Use Case Diagram is a graphic depiction of the interactions among the elements of Food Ordering System. It represents the methodology used in system analysis to identify, clarify, and organize system requirements of Food Ordering System.

### 4.1.2Customer Module:

The Customer firstly get logging to the application. Once he gets logged in application then customer can see their dashboard. To place the order, customer have to pay the required amount for that item. if the amount is less than the required amount then order will not be processed. Customer has the authority to add any item, delete item from there order list. Customer also can book the table or place order for home delivery. Also, she/he can see item details and status of order.

### 4.1.3 Cashier and Stock Manager Module:

Cashier has to handle whole system of food ordering system. It mainly include**s** adding the receipts of user separately, adding and deleting of food items, view transaction details, see the profile of customer, can see cookbook,availability of food items. Also, can see the all chefs and their work according to their skillset cashier can categories chef and their priority.Stock manager handles whole food ordering system. Which also included adding and deleting of food items, editing menu details and decides the specials menu of the day, availability of material, availability of inventory, availability of food. Also, it manages the chef.

### 

# CHAPTER 5

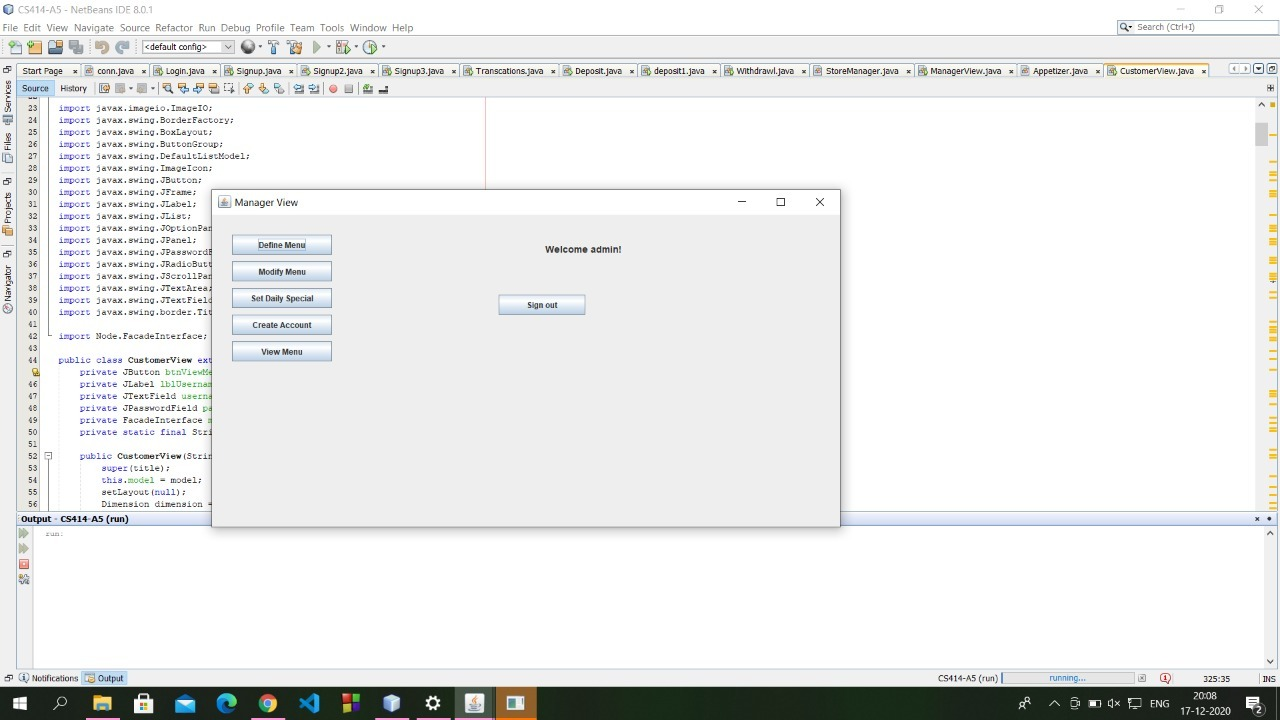
## Module Implementation

In this ‘Food Ordering system’ project consist of four important modules which has been explained below-

* + - 1. Stock Manager Module
      2. Customer Module
      3. Chef Manager
      4. 4.Cashier

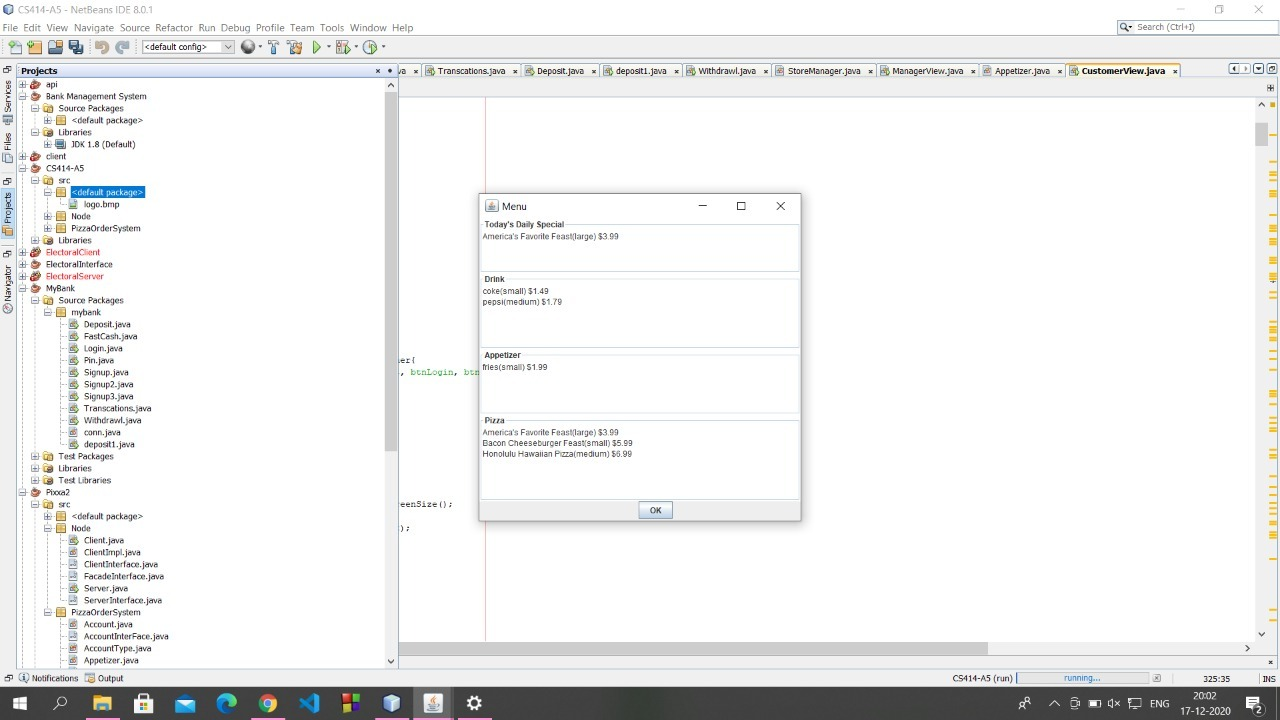
## 5.1Stock Manager

### Output Screenshot:



**Figure No.5.1: Output screenshot of Stock Manager Page**

Stock manager have to handle all the ordering details. Dashboard of stock manager shows the menu details, modified menu, account created by user and view of the menu. Shown in figure 5.1.



**Figure No.5.2.Stock manager Working**

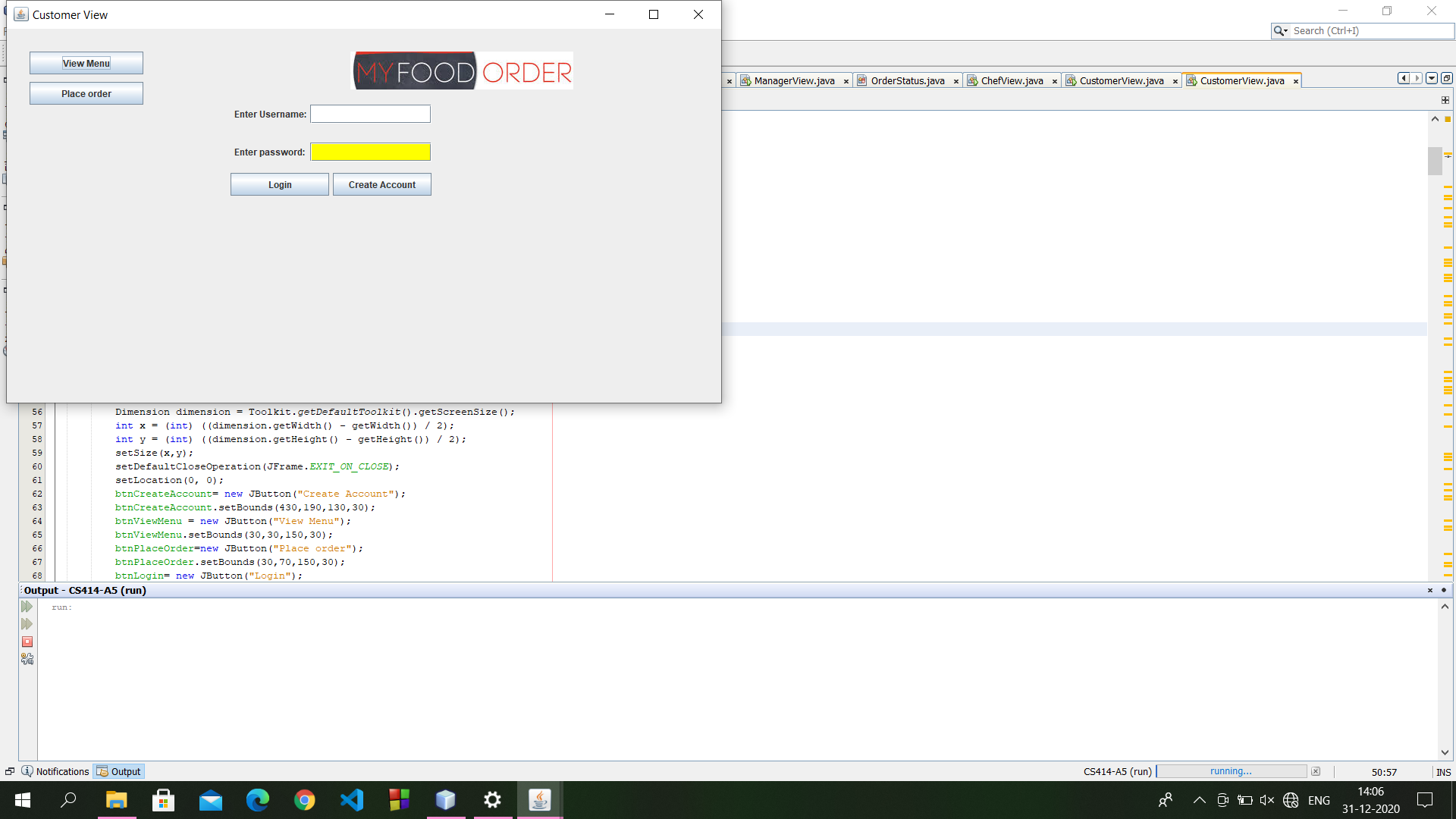
The Menu details shown by Figure 5.2.The working of the system is mainly the ordering of menu and its details.

**4.2Customer Module:**

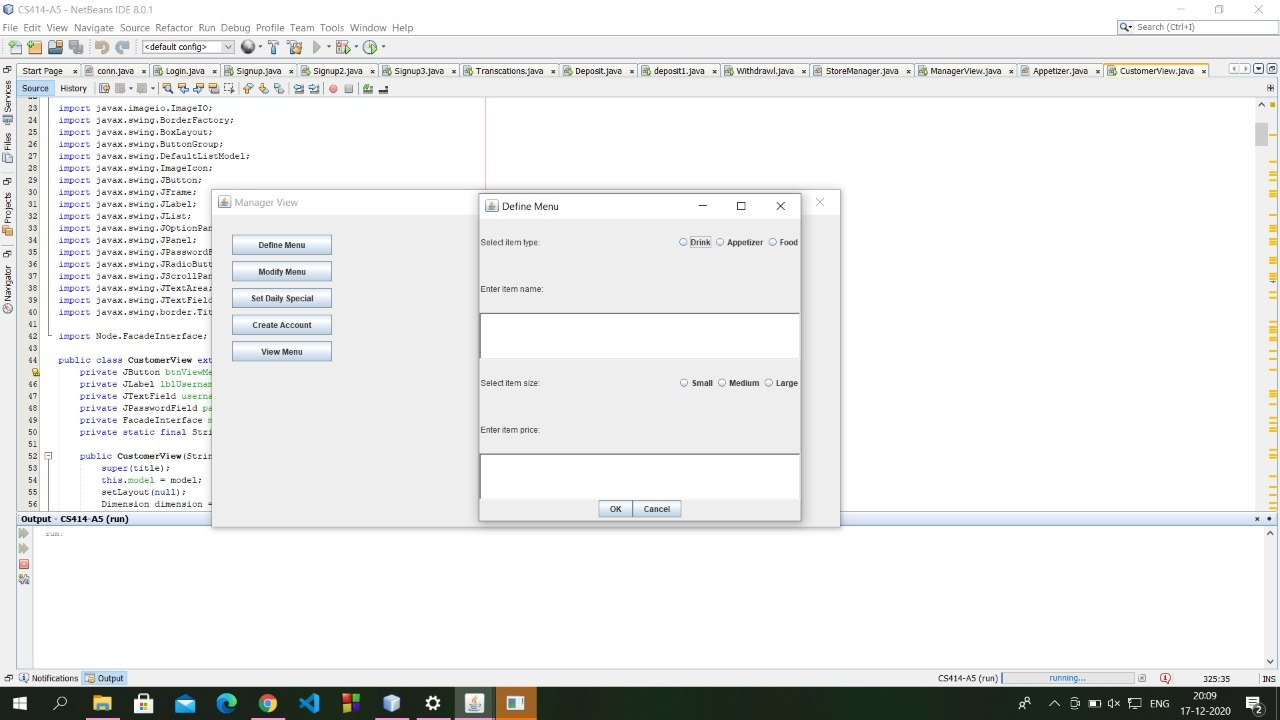
User authentication is very common in modern web application. It is a security mechanism that is used to restrict unauthorized access to member-only areas. In this module have create a login form where user can enter their username and password.

When user submit the form these inputs will be verified against the credentials stored in the database, if the username and password match, the user is authorized and granted access to the site, otherwise the login attempt will be rejected. Also, user has to fill the details of item which he/she required. They have provided the list of items with their price. Then have to place the order and pay the bill during this stage. If money is less than required amount then process will stop and give alerted message.

**Output Screenshot:**



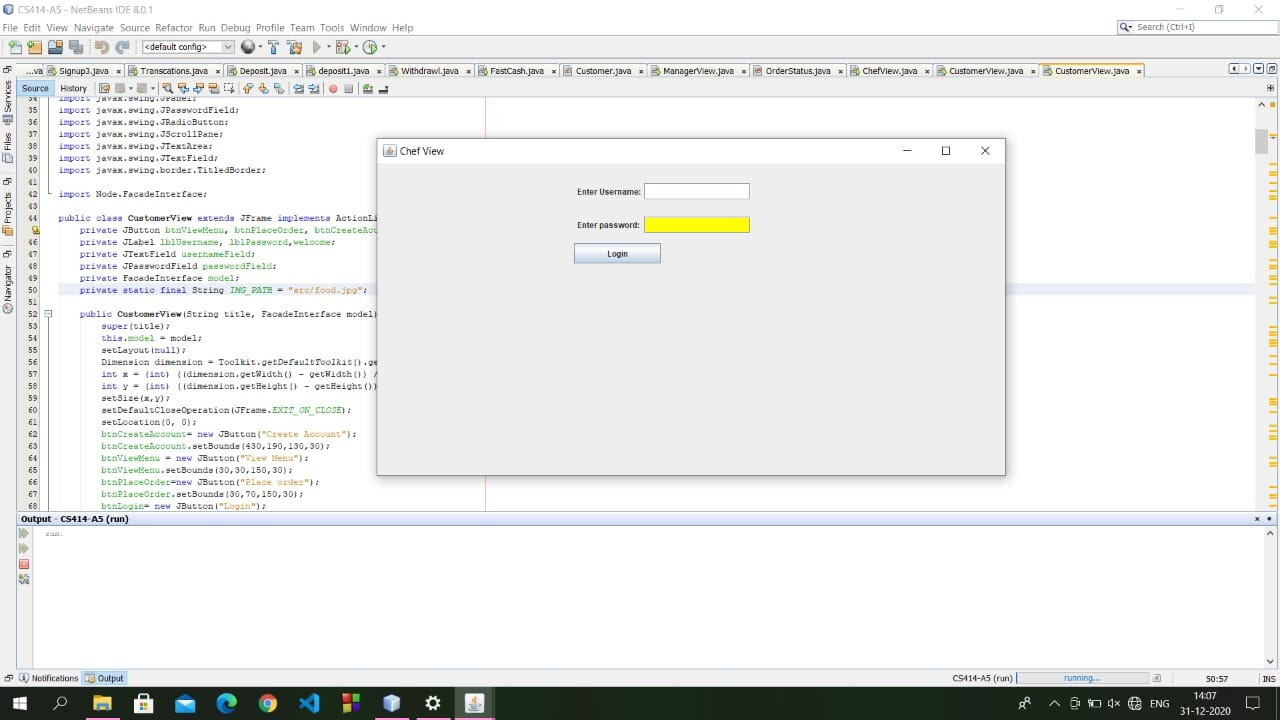
**Figure No.5.3.Login page of Customer Module**



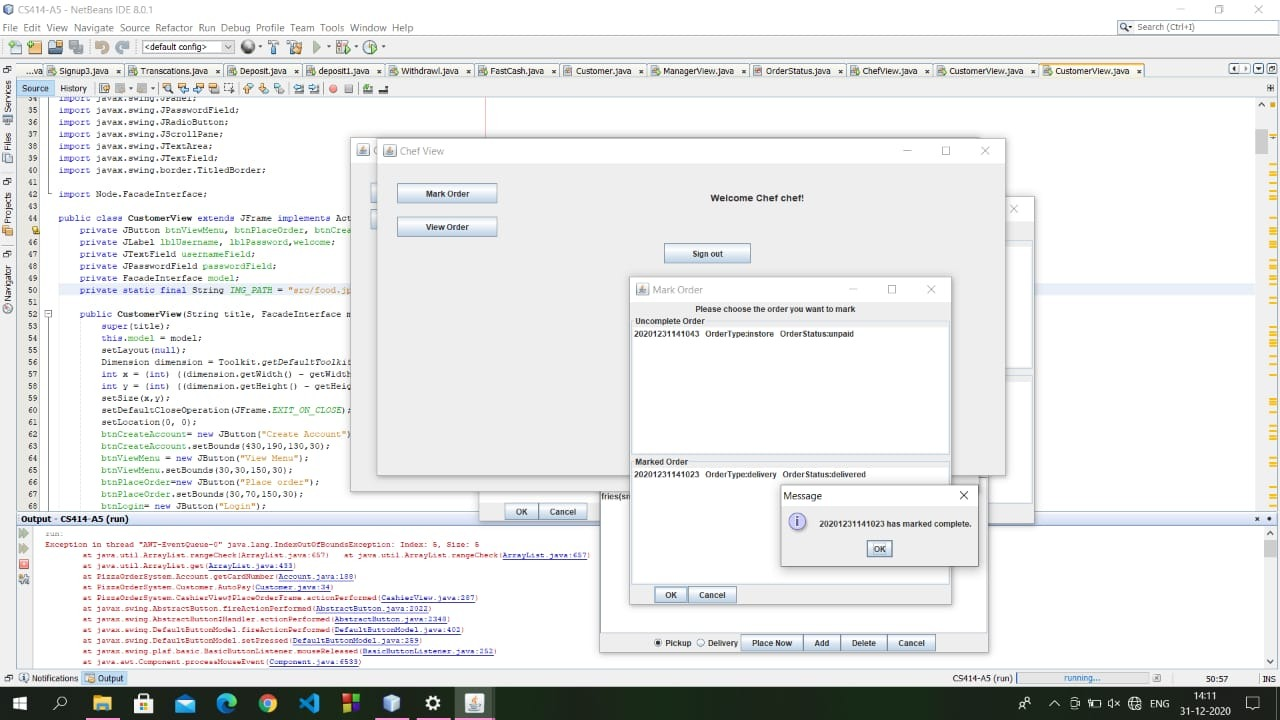
**Figure 5.4. Output of customer Module**

**5.3Chef Module:**

This Module is mainly to handle whole kitchen stuff. Which item has to make by which chef this whole system handled by this module. For Chef also there is authentication system. According the skillset and work of chef best chef decided. Also, there are each chef for specified food and also they can be managed and coordinated to provide order in specific time and give the status to the customer.



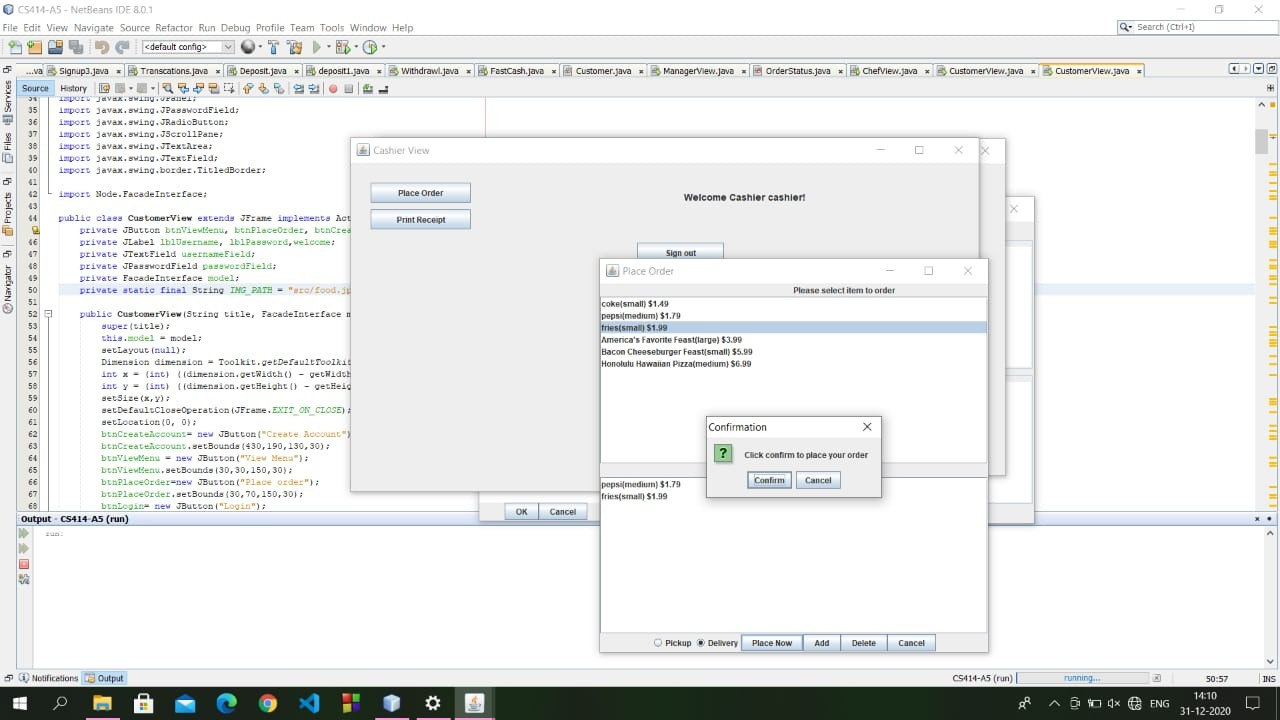
**Figure 5.5.Login page of Chef Module**



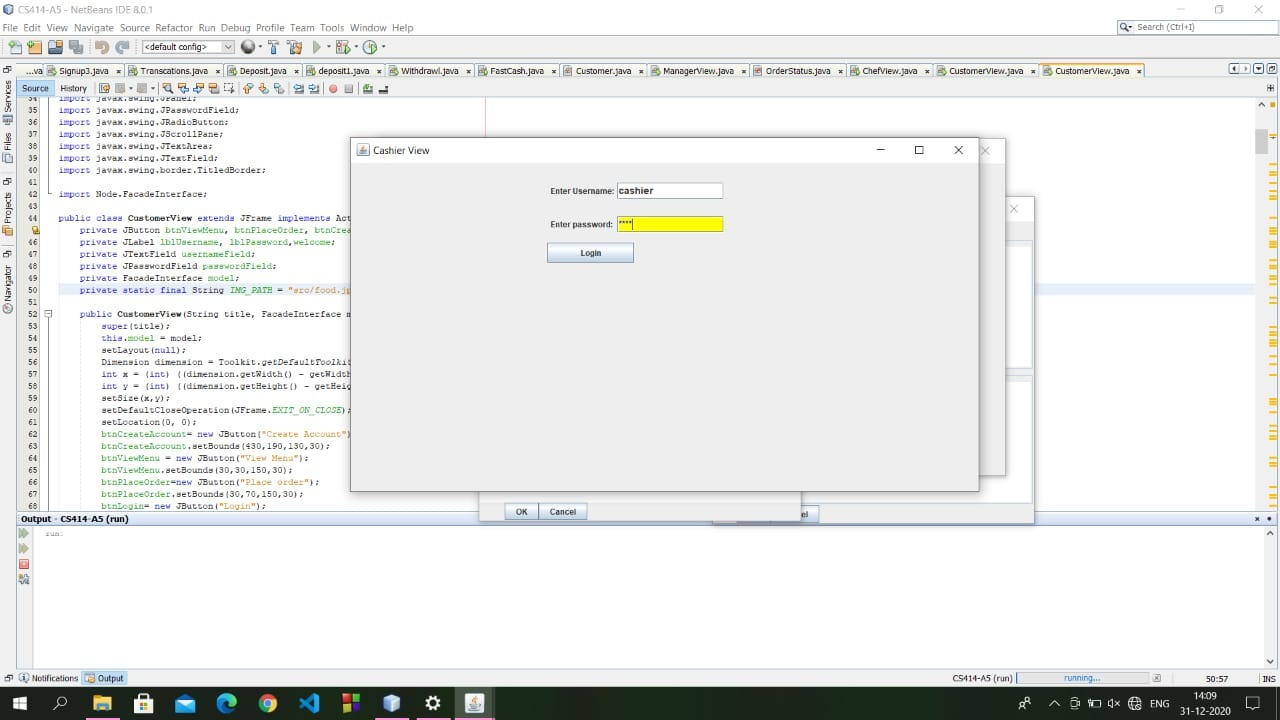
**Figure:5.6: Working of Chef Module**

**4.3Cashier Module:**

The Cashier Module mainly handles the details of order. Cashier can edit item details, she/he can add or delete the item, helps to customer to place the order. It also handles enquiry section. Cashier can view all details of customer including transaction process.



**Figure:5.7.Working of Chashier Module**



**Figure:5.8.Login page of Chashier Module**

## 6.1.Debug

**CHAPTER 6**

**Debug and Test Case**

Debugging is process of finding errors in developed system. At the time of first debugging the system there was an error of login failure this was due to method declared in the code. Later method was replaced by “POST” and the login was successful. Further there was an error in booking appointment, after clicking on book appointment button the page would not return the desired page, so this problem was resolved after checking the URL in the code. Correction was made in URL and correct URL of desired page was entered. Later on, an error was encountered in doctor model, so tried to add department but the input was not accepted, this error was due to syntax error in insert query of MySQL.

## 6.2. Manual Test Scenarios and Test Cases

### Test Scenarios

**Test Scenario 1: Check the Login Functionality**

1. Check system behavior when valid username and password is entered.
2. Check system behavior when invalid username and password is entered.

### Test Scenario 2: Check the chef functionality

1. Check system behavior when placed order are available or not and also see the status of the order and provide it to customer

**Test Scenario 3: Check the Stock Manager Functionality** 1. Check system behavior when add or remove item from list

1. Check system behavior

### Test Scenario 4: Check the Cashier Functionality

1.Check system behavior when order placed

2.Check food delivered or not

**CHAPTER 6**

**Conclusion**

We have implemented Food ordering system using distributed architecture based on the java RMI technology. This architecture of the system is definite and has reduced the complexity of development, which has good scalability and flexibility making the system easier.

Compared to the architecture Servlet/JSP and JavaBeans, it can adapt to a more large-scale distributed application. We have implemented cashier, customer, store manager, chief modules by calling them remotely.

# References

* + 1. h<https://ieeexplore.ieee.org/document/5199968>
    2. M. Firdouse Ali Khan, Swapna, “Design and Implementation of Ordering System for Restaurants”, in International Journal of Engineering Research & Technology (IJERT), Vol. 1, Issue 10, December- 2012.
    3. Medium.com/basics of RMI