

## **CONTACTLESS DINING BASED RESTAURANT MANAGEMENT APPLICATION**

### **A PROJECT REPORT**

*Submitted by*

**AMITH A** **(211517104010)**

**ASWIN KUMAR G R** **(211517104020)**

**LOGESH KARTHIK S** **(211517104088)**

*in partial fulfillment for the award of the degree*

*of*

**BACHELOR OF ENGINEERING**

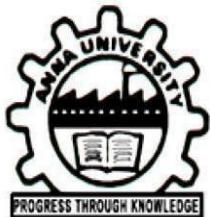
**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**PANIMALAR INSTITUTE OF TECHNOLOGY, POONAMALLEE**

**ANNA UNIVERSITY: CHENNAI -600025**

**AUGUST 2021**



**ANNA UNIVERSITY: CHENNAI -600025**

**BONAFIDE CERTIFICATE**

Certified that this project report "**CONTACTLESS DINING BASED RESTAURANT MANAGEMENT APPLICATION**" is the bonafide work of "**AMITH A(211517104010), ASWIN KUMAR G R(211517104020) and LOGESH KARTHIK S(211517104088)**" who carried out the project work under my supervision.

**SIGNATURE**

**Dr. V. SUBEDHA, M.Tech, Ph.D**

**Professor and Head,**

**Department of CSE,**

**Panimalar Institute of Technology,  
Poonamallee, Chennai - 600123.**

**SIGNATURE**

**Mr. S.RAJA, M.Tech,**

**Assistant Professor,**

**Department of CSE,**

**Panimalar Institute of Technology,  
Poonamallee, Chennai - 600123.**

Certified that the above candidates were examined in the university project work viva voce examination held on **06-08-2021** at Panimalar Institute of Technology, Chennai – 600 123.

**INTERNAL EXAMINER**

**EXTERNAL EXAMINER**

## **ACKNOWLEDGEMENT**

A project of this magnitude and nature requires kind cooperation and support from many, for successful completion. We wish to express our sincere thanks to all those who were involved in the completion of this project.

We would like to express our deep gratitude to **Our Beloved Secretary and Correspondent, Dr. P. CHINNADURAI, M.A., Ph.D.**, for his kind words and enthusiastic motivation which inspired us a lot in completing this project.

We also express our sincere thanks to **Our Dynamic Directors Mrs. C. VIJAYA RAJESWARI, Mr. C. SAKTHI KUMAR, M.E., and Mrs. SARANYASREE SAKTHIKUMAR, B.E.**, for providing us with the necessary facilities for completion of this project.

We also express our gratefulness to **Our Principal Dr. T. JAYANTHI M.E., Ph.D.**, who very much helped us in the completion of the Project.

We wish to convey our thanks and gratitude to our head of the Department, **Dr. V. SUBEDHA, M.Tech., Ph.D.**, Department of Computer Science & Engineering, for her support and by providing us ample time to complete our project.

We express our indebtedness and gratitude to our Project Guide, **Mr. S. RAJA, M.Tech.**, Professor and Head, Department of Computer Science & Engineering for her guidance throughout the Course of our project.

## TABLE OF CONTENTS

<b>CHAPTER</b>	<b>TOPICS</b>	<b>PAGE NO</b>
	<b>ABSTRACT</b>	i
	<b>LIST OF SYMBOLS</b>	ii
	<b>LIST OF FIGURES</b>	iii
<b>1.0</b>	<b>INTRODUCTION</b>	<b>1-2</b>
1.1	General	1
1.2	Objective	2
1.3	Motivation	2
<b>2.0</b>	<b>RELATED WORK</b>	<b>3-4</b>
2.1	Literature Review	3
2.2	Interface of Literature review	4
<b>3.0</b>	<b>SYSTEM ANALYSIS</b>	<b>5-8</b>
3.1	Problem Definition	5
3.2	Proposed System	5
3.3	Feasibility Study	5
3.3.1	Technical Feasibility	6
3.3.2	Economic Feasibility	6
3.4	Requirement Analysis	6
3.5	Hardware Requirements	6
3.6	Software Requirements	7
3.7	Software Description	8

<b>CHAPTER</b>	<b>TOPICS</b>	<b>PAGE NO</b>
<b>4.0</b>	<b>SYSTEM DESIGN</b>	<b>12-16</b>
4.1	Architecture Diagram	12
4.2	Class Diagram	13
4.3	Data Flow Diagram	14
4.4	Activity Diagram	15
<b>5.0</b>	<b>MODULE DESCRIPTION</b>	<b>17-20</b>
5.1	Module classification	17
5.2	Data Processing	20
<b>6.0</b>	<b>TESTING</b>	<b>21-26</b>
6.1	Taxonomy of Testing	21
6.1.1	White box Testing	21
6.1.2	Black box Testing	22
6.2	Testing in Particular	24
6.2.1	Unit Testing	24
6.2.2	Integration Testing	24
6.2.3	Acceptance Testing	25
6.3	Test Cases	26
<b>7.0</b>	<b>RESULT</b>	<b>27-32</b>
<b>8.0</b>	<b>CONCLUSION</b>	<b>33</b>
8.1	Conclusion	33
8.2	Future Enhancement	33
<b>9.0</b>	<b>REFERENCES</b>	<b>34</b>
<b>10.0</b>	<b>APPENDICES</b>	<b>35-129</b>
	APPENDIX A – Base Paper	36
	APPENDIX B – Source Code Structure	44
	APPENDIX C – Source Code	46
	APPENDIX D – Published Paper	131
	APPENDIX E – Certificates	136

## **ABSTRACT**

This project aims at automating the entire restaurant management process for multiple restaurants based on individual login and implementing contactless dining experience to the customers. The Graphical User Interface enables restaurant owners and staff members to view the real time data of the orders ,sales and manage them. The overall Application has various features to manage operations including various departments and also provides separate logins for each department. Various reports on day basis and monthly basis are generated and has the feature to be downloaded also. Customers can have their own login to order food and track the status of their order in real time thereby implementing contactless dining norms. This project thereby reduces man power and also provides real time data and has efficient collection of past data.

## LIST OF SYMBOLS

S.NO	SYMBOL NAME	NOTATION	DESCRIPTION
1.	Initial Activity		This shows the Starting point or first activity of flow.
2.	Final Activity		The end of the Activity diagram is shown by a bull's eye symbol.
3.	Activity		Represented by a rectangle with rounded edges.
4.	Decision		A logic where a decision is to be made.
5.	Use Case		Describe the interaction between a user and a system.
6.	Actor		A role that a user plays with respect to system.
7.	Object		A Real time Entity.
8.	Message		To send message between the life of an object.

## **LIST OF FIGURES**

<b>FIG NO</b>	<b>NAME OF THE FIGURE</b>	<b>PAGE NO</b>
4.1	Architecture Diagram	12
4.2	Class Diagram	13
4.3	Data Flow Diagram	14
4.4	Activity Diagram	16
5.1	Admin module classification	17
5.2	Customer module classification	19
7	Result Screenshots	27
10.1	Source code Structure	44
10.2	Database Structure	45

# **CHAPTER 1**

## **INTRODUCTION**

### **1.1 GENERAL**

Restaurant plays a major role in the food and beverages industry which is actually a major part of daily life of people. This project aims at automating the entire restaurant management process for multiple restaurants based on individual login and implementing contactless dining experience to the customers. The Graphical User Interface enables restaurant owners and staff members to view the real time data of the orders ,sales and manage them. The overall Application has various features to manage operations including various departments and also provides separate logins for each departments. Various reports on day basis and monthly basis is generated and has the feature to be downloaded also. Customers can have their own login to order food and track the status of their order in real time thereby implementing contactless dining norms. This project thereby reduces man power and also provides real time data and has efficient collection of past data. This paper is organized as follows. In Section 2, we briefly introduce several modules present in our application and their features. In Section 3, we describe eachmodule in detail and its use cases and activity are clearly depicted which makes the application more understandable. In Section 4, we test it using various test cases and verify its performance with various testing strategies. The final section concludes the paper and points out the future work.

## **1.2 OBJECTIVE**

The objective of our project is to develop a High performance operation tool that is user friendly and computationally efficient. The application consists of a Visually understandable user interface to enable a friendly environment for end users. HTML , php, javascript are used for efficient development. To store the data My SQL database is used. Accurate data capturing is done using efficient coding and efficient reports on data are generated based on collected data.

## **1.3 MOTIVATION**

The motivation of the project is, Nowadays different types of management systems are available, at the same time they are not following contactless dining norms which is the recent need. Food industry is spread worldwide and restaurants are our focus. One of the main challenges in this area is safety which is fulfilled by contactless practices. Unless we follow contactless dining practices it is considered totally unsafe and unhealthy. Keeping in mind the covid situation, the aim of having in-contact dining practice is a fear and threat to the current situation in the society. Our project takes measures to provide healthier and safe digital restaurant practices to the upcoming days of a better world. One of the solutions to deal with the problem is to build a High performance automated operation tool that consists of a Visually understandable user interface that enables users to get used to contactless dining practices. The idea is to combine all the modules required for management of the restaurant under one roof in a highly efficient digital platform with a Visually understandable user interface and application which captures accurate data and provides Efficient reports on that data.

## **CHAPTER 2**

### **RELATED WORK**

#### **2.1 LITERATURE SURVEY**

- [1] **Bahattin Ozdemir and Osman-Caliskan, proposed a paper “A REVIEW OF LITERATURE ON RESTAURANT MENUS: SPECIFYING THE MANAGERIAL ISSUES ”** in which, This study specifically aims at reviewing the critical managerial issues of the menu, and demonstrating the conceptual structure of menu management. Based on the conceptual and empirical findings of menu literature, the major menu management issues are menu planning, menu pricing, menu designing, menu operating and menu development. Additionally, the paper makes a discussion on the conceptual relations between menu and meal experience. Given the scarcity of research that incorporates evidence and concepts of previous studies in one single study, the conceptual structure of menu management presented in this paper allows a comprehensive understanding of the menu.
- [2] **Soundarya H K, Abhinaya R and Prathiba B S, proposed a paper “SURVEY ON INTELLIGENT FOOD MENU ORDERING SYSTEM ”** in which, The increasing growth of wireless technology and mobile services in this era is creating a great impact on our lifestyle. Some efforts have already been taken to carry the process of ordering in hotels by using hardware components like Avr16 Microcontroller, LCD display module and Zigbee module. The Existing system is fully dependent on hardware and it is very difficult to club all the components to make a system. In addition to that understanding and operating the system is very difficult for some users and this system is not going to manage the business model properly. In this field, touch screen based advanced menu display and ordering system concept is a new innovative idea.

## 2.2 INTERFACE OF LITERATURE REVIEW

<b>TITLE</b>	<b>AUTHOR NAME</b>	<b>INFERENCE</b>
A review of literature on restaurant menus: Specifying the managerial issues	Bahattin Ozdemir, Osman-Caliskan	This study specifically aims at reviewing the critical managerial issues of menu, and demonstrating the conceptual structure of menu management. Based on the conceptual and empirical findings of menu literature, the major menu management issues are menu planning, menu pricing, menu designing, menu operating and menu development. Additionally, the paper makes a discussion on the conceptual relations between menu and meal experience. Given the scarcity of research that incorporates evidence and concepts of previous studies in one single study, the conceptual structure of menu management presented in this paper allows a comprehensive understanding of the menu.
Survey On Intelligent Food Menu Ordering System	Soundarya H K, Abhinaya R, Prathiba B S	The increasing growth of wireless technology and mobile services in this era are creating a great impact on our lifestyle. Some efforts have already been taken to carry the process of ordering in hotels by using hardware components like Avr16 Microcontroller, LCD display module and Zigbee module. The Existing system is fully dependent on hardware and it is very difficult to club all the components to make a system. In addition to that understanding and operating the system is very difficult for some users and this system is not going to manage the business model properly. In this field, touch screen based advanced menu display and ordering system concept is a new innovative idea.

## **CHAPTER 3**

### **SYSTEM ANALYSIS**

#### **3.1 PROBLEM DEFINITION**

The problem statement of the project is that the management tool for any restaurant is mostly semi automatic with manual intervention in many tasks. most modules in restaurants aren't automated yet. Contactless dining practices aren't followed as they are not in practice while in the past but now it is a necessary part of dining culture. This project aims to overcome the above with recent stable technologies in an efficient way possible.

#### **3.2 PROPOSED SYSTEM**

In the proposed system the entire restaurant management process is automated. Multiple restaurants have their own individual login. Also contactless dining norms are followed by reducing human intervention to a great extent from the customers. The Graphical User Interface enables restaurant owners and staff members to view the real time data of the orders ,sales and manage them. Each department had separate logins for them. various departments in restaurants are covered (departments : kitchen , manager, cashier , waiter ) .Generates Daily and monthly Reports .Provides real time data and has efficient collection of past data.

#### **3.3 FEASIBILITY STUDY**

A feasibility study is carried out to select the best system that meets performance requirements. The main aim of the feasibility study activity is to determine that it would be financially and technically feasible to develop the product.

### **3.3.1 TECHNICAL FEASIBILITY**

This is concerned with specifying the software will successfully satisfy the user requirement. Open source and business-friendly and it is truly cross platform, easily deployed and highly extensible.

### **3.3.2 ECONOMIC FEASIBILITY**

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. The enhancement of the existing system doesn't incur any kind of drastic increase in the expenses. PHP is open source and ready available for all users. Since the project is runned in web enhancement and updation is easy hence is cost efficient.

## **3.4 REQUIREMENT ANALYSIS**

Requirement analysis determines the requirements of a new system. This project analyses on product and resource requirement, which is required for this successful system. The product requirement includes input and output requirements it gives the wants in term of input to produce the required output. The resource requirements give in brief about the software and hardware that are needed to achieve the required functionality.

## **3.5 HARDWARE REQUIREMENTS**

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the systems do and not how it should be

implemented.

- Desktop / Mobile
- Active internet connection
- Desktop - ( Either windows /Mac)
- Mobile - (preferably Android)

### **3.6 SOFTWARE REQUIREMENTS**

The software requirements are the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team's and tracking the team's progress throughout the development activity.

- Operating system : Windows
- Coding Language : PHP, JavaScript, AJAX, jQuery
- Front End Tool : HTML, Bootstrap4, JS
- Database : MySQL
- Server : Apache
- Tool : xampp
- Development environment : VScode (visual studio code)

### **3.7 SOFTWARE DESCRIPTION**

#### **PHP: Hypertext Preprocessor**

PHP ( Hypertext Preprocessor) .PHP is an intuitive, server side scripting language. Like any other scripting language it allows developers to build logic into the creation of web page content and handle data returned from a web browser. PHP also contains a number of extensions that make it easy to interact with databases, extracting data to be displayed on a web page and storing information entered by a web site visitor back into the database. PHP consists of a scripting language and an interpreter. Like other scripting languages, PHP enables web developers to define the behavior and logic they need in a web page. These scripts are embedded into the HTML documents that are served by the web server. The interpreter takes the form of a module that integrates into the web server, converting the scripts into commands the computer then executes to achieve the results defined in the script by the web developer.

#### **JAVA**

Java is Object Oriented, Platform Independent, Simple, Secure, Architecture-neutral, Portable, Robust, Multithreaded, Interpreted, High Performance, Distributed, Dynamic. Java is platform independent programming language. Java creates a byte code that can be run on any type of machine with the help of Java Virtual Machine (JVM). Java is secure language because java is not using pointers. Java is also called "write once and run anywhere" programming language. The latest release of the Java Standard Edition is Java SE 8. With the advancement of Java and its widespread popularity, multiple configurations were built to suit various types of platforms. For example: J2EE for Enterprise Applications, J2ME for Mobile Applications. The new J2 versions were renamed as Java SE, Java EE, and Java ME respectively. Java is guaranteed to be Write Once, Run Anywhere.

The Java Development Kit (JDK) is an implementation of either one of the Java

Platform, Standard Edition, Java Platform, Enterprise Edition, or Java Platform, Micro Edition platform released by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, macOS or Windows.

The JDK includes a private JVM and a few other resources to finish the development of a Java Application. Since the introduction of the Java platform, it has been by far the most widely used Software Development Kit (SDK).[citation needed] On 17 November 2006, Sun announced that they would release it under the GNU General Public License (GPL), thus making it free software. This happened in large part on 8 May 2007, when Sun contributed the source code to the OpenJDK. The Java Development Kit (JDK) is a software development environment used for developing Java applications and applets. It includes the Java Runtime Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools needed in Java development.

The Java Runtime Environment (JRE) is a set of software tools for development of Java applications. It combines the Java Virtual Machine (JVM), platform core classes and supporting libraries. JRE is part of the Java Development Kit (JDK), but can be downloaded separately. JRE was originally developed by Sun Microsystems Inc., a wholly-owned subsidiary of Oracle Corporation. Also known as Java runtime.

JRE 1.0 has evolved with a variety of class and package additions to the core libraries, and it has grown from a few hundred classes to several thousand in Java 2 Platform, Standard Edition (J2SE). Entirely new APIs have been introduced, and many of the original version 1.0 APIs have been deprecated. There has been a significant degree of criticism because of these massive additions.

## **Bootstrap 4**

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Bootstrap is among the most starred projects on GitHub, with more than 142,000 stars, behind freeCodeCamp (almost 312,000 stars) and marginally behind Vue.js framework. The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight. Bootstrap also comes with several JavaScript components in the form of jQuery plugins. They provide additional user interface elements such as dialog boxes, tooltips, and carousels. Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields. The most prominent components of Bootstrap are its layout components, as they affect an entire web page. The basic layout component is called "Container", as every other element in the page is placed in it. Developers can choose between a fixed-width container and a fluid-width container.

## **MySQL**

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB.

MySQL is a component of the LAMP web application software stack (and others), which is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database- driven web applications, including Drupal, Joomla, phpBB, and

WordPress.

MySQL is written in C and C++. Its SQL parser is written in yacc, but it uses a home-brewed lexical analyzer. MySQL works on many system platforms, including AIX, BSDi, FreeBSD, HP-UX, eComStation, i5/OS, IRIX, Linux, macOS, Microsoft Windows, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, OS/2 Warp, QNX, Oracle Solaris, Symbian, SunOS, SCO OpenServer, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists. The MySQL server software itself and the client libraries use dual-licensing distribution. They are offered under GPL version 2, or a proprietary license.

## **Visual Studio Code**

Visual Studio Code is a freeware source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

Microsoft has released Visual Studio Code's source code on the microsoft/vscode (Code - OSS) repository of GitHub, under the permissive MIT License, while the releases by Microsoft are freeware. the Stack Overflow 2019 Developer Survey, VisualStudio Code was ranked the most popular developer environment tool, with 50.7% of 87,317 respondents reporting that they use it.

Visual Studio Code was first announced on April 29, 2015, by Microsoft at the 2015 Build conference. A Preview build was released shortly thereafter.

On November 18, 2015, Visual Studio Code was released under the MIT License, having its source code available on GitHub. Extension support was also announced. On April 14, 2016, Visual Studio Code graduated from the public preview stage and was released to the Web.

## CHAPTER 4

### SYSTEM DESIGN

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering.

#### 4.1 ARCHITECTURE DIAGRAM

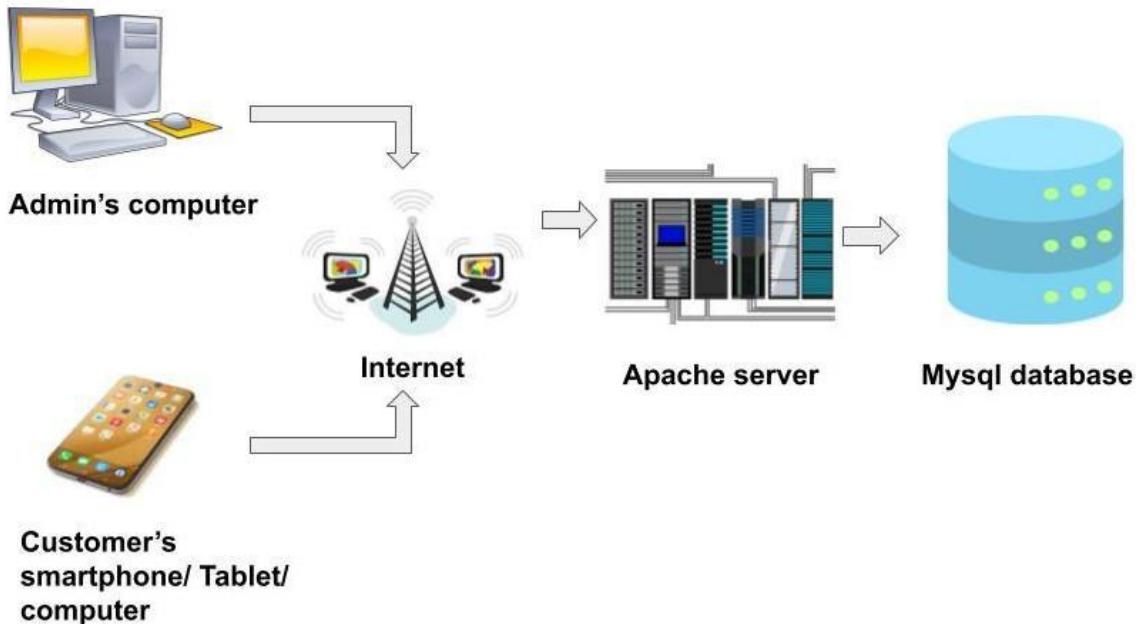


Fig No: 4.1 Architecture Diagram

## 4.2 CLASS DIAGRAM

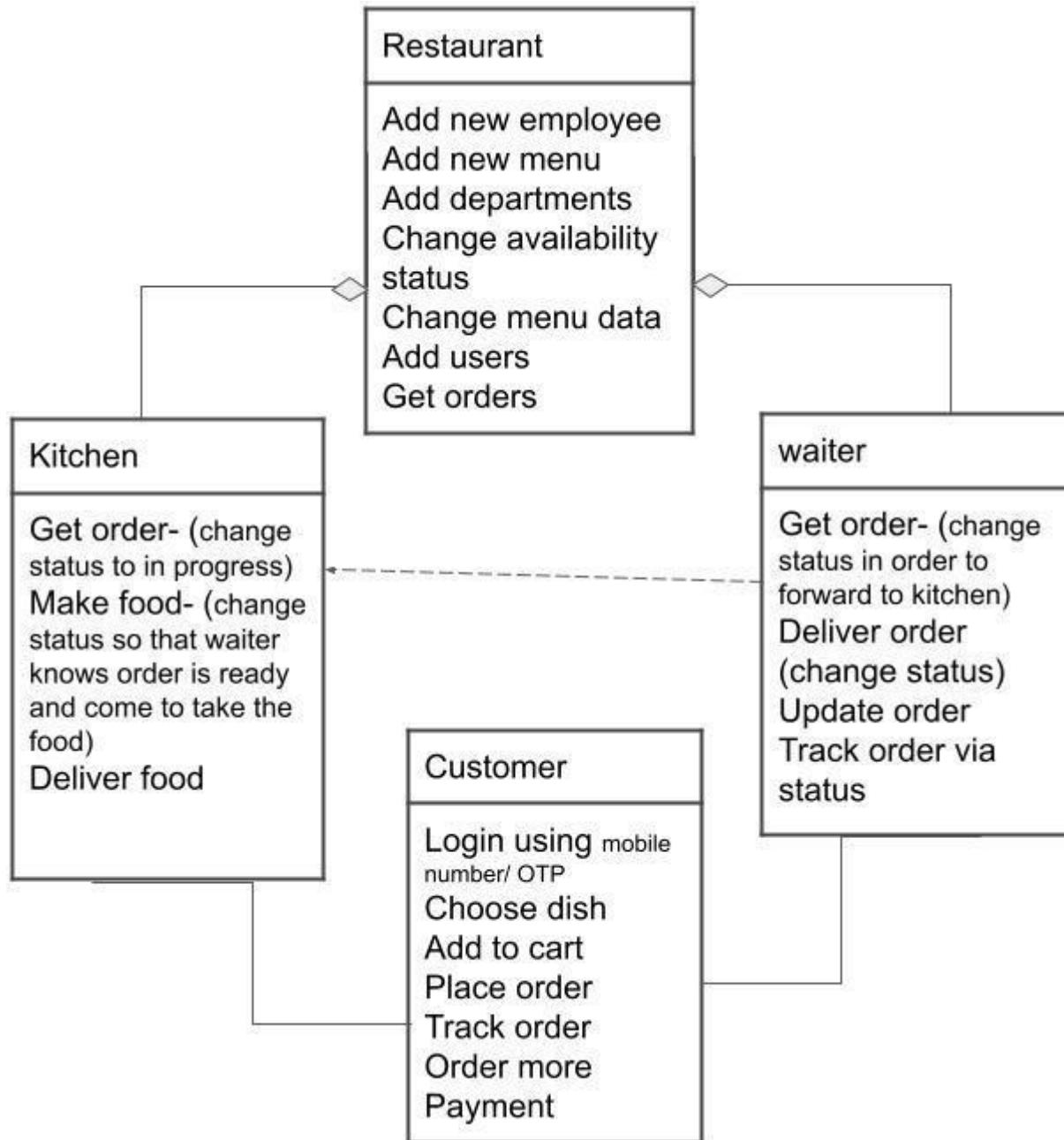


Fig No: 4.2 Class Diagram

### 4.3 DATA FLOW DIAGRAM

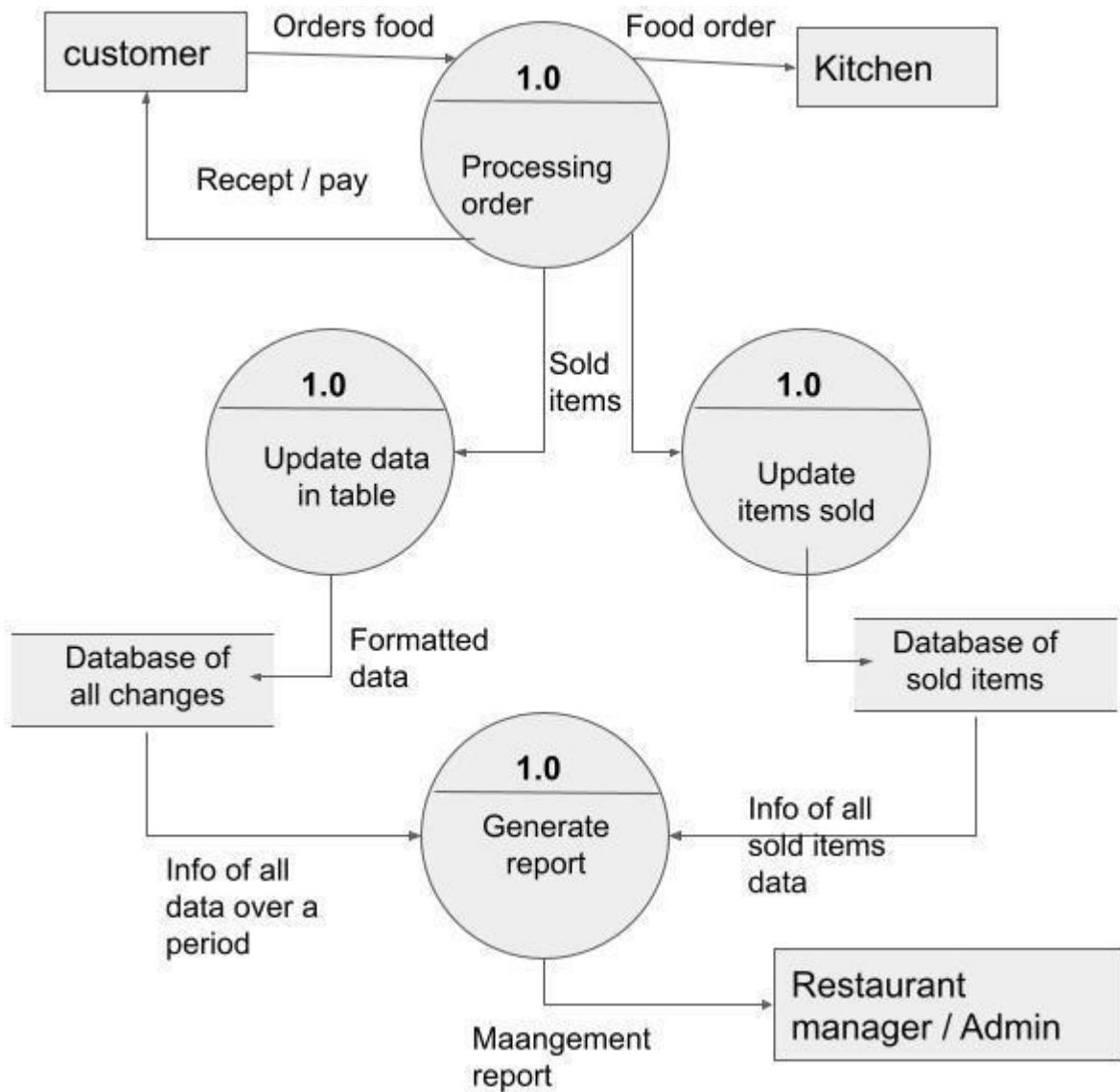
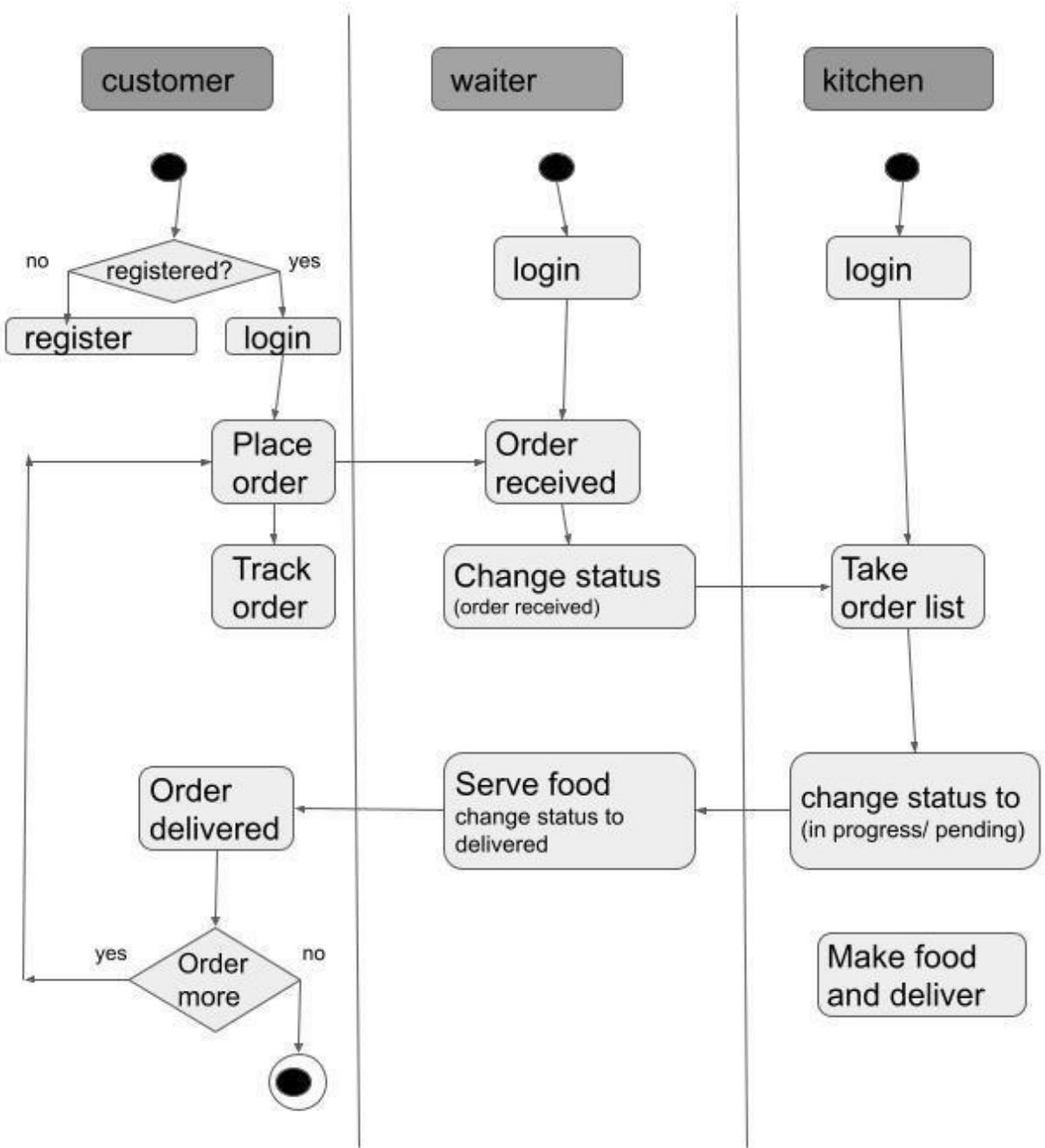


Fig No.: 4.3 Data flow Diagram

#### 4.4 ACTIVITY DIAGRAM



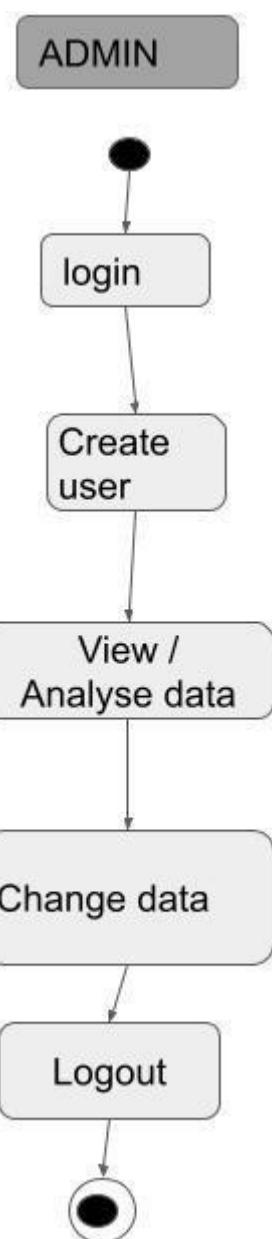


Fig No: 4.4 Activity Diagram

## CHAPTER 5

### MODULE DESCRIPTION

#### 5.1 MODULE CLASSIFICATION

##### Admin Module

- Administration module
- Menu management
- Employees module
- Additional settings module
- Reports module
- Table and order management module

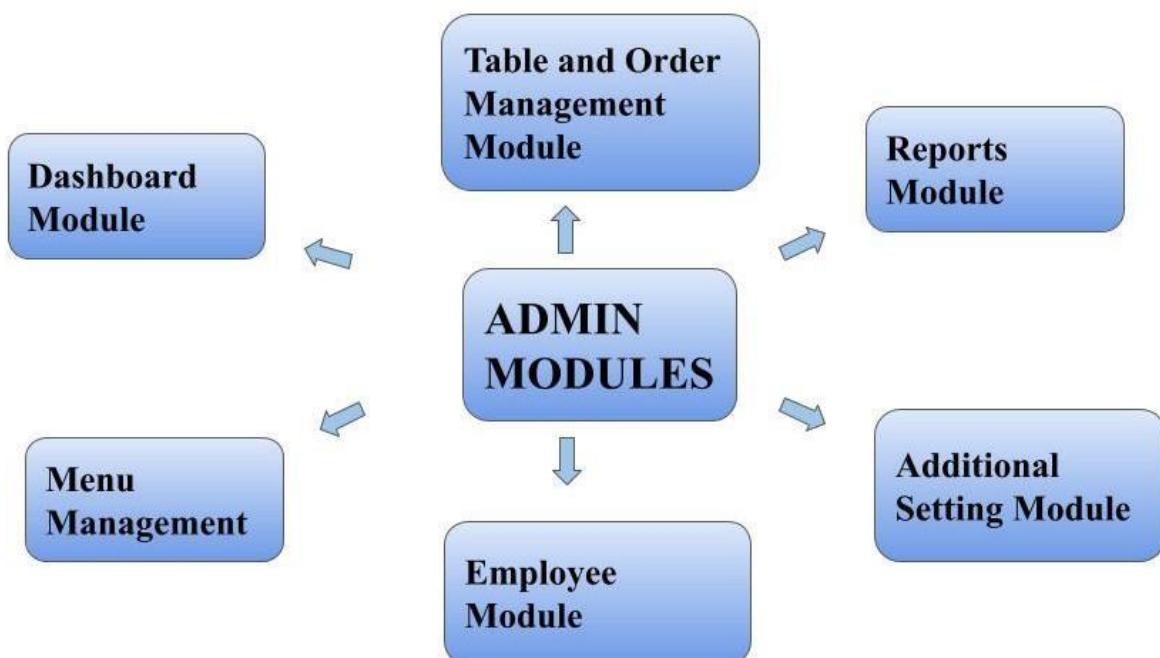


Fig No: 5.1 Admin Module Classification

## **Modules Definition**

### Administration module

Consists of dashboard to view the overall data such as sales , products available, orders, pending orders, total sales etc.

### Menu management

Contains all the features to add delete or edit an item in the digital menu and also provide availability status of the item

### Employees module

Features such as add employee provide separate login , page role mapping and others

### Additional settings module

Consists features to reset pin , password etc.

### Reports module

Generates and displays reports based on selection criteria such as monthly report, daily report etc

### Table and order management module

Has features to create table, change table availability status, create order order status and tracking and edit features

## **Customer Module**

- Menu module
- Cart module
- Bill module
- Track order status module
- Digital menu

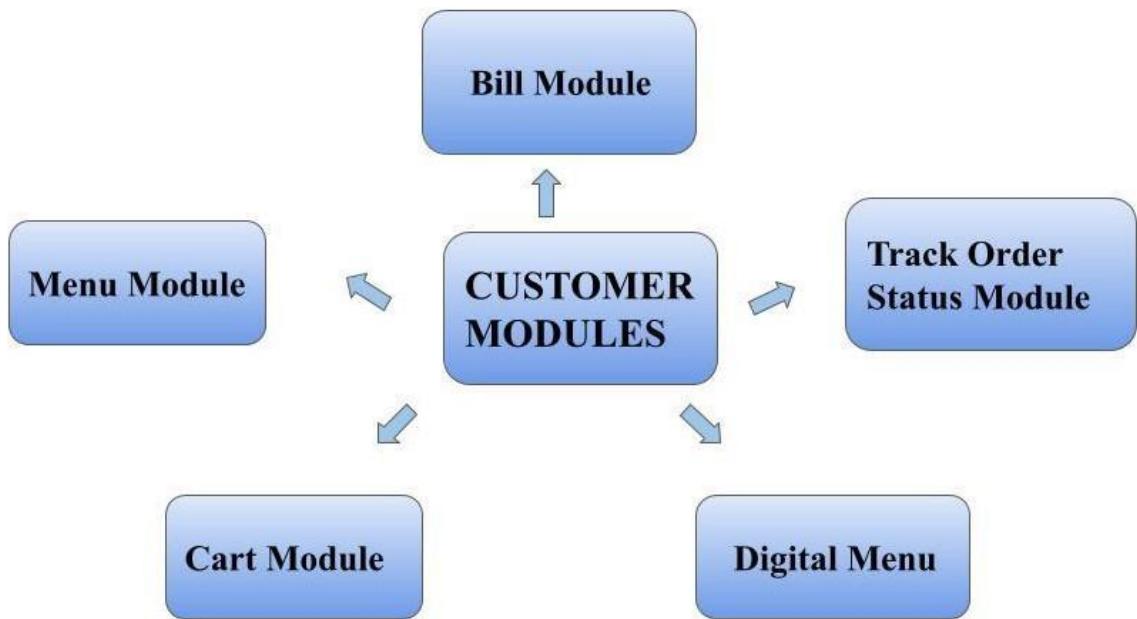


Fig No: 5.2 Customer Module Classification

## **Modules Definition**

### **Menu module**

This module enables customer to view menu and make selections to add in the cart

### **Cart module**

This consists of a list of items based on customer selection in order to provide an overall view of the total selection to proceed to the next process

### **Bill module**

Here comes the final step which enables customer to pay for the food

### **Track order status module**

After making the order this module has features to track the status of the order such as pending, in progress and delivered.

### **Digital menu**

This is the digital food menu provided by the particular restaurant which is made available for the customer to make selection and process further with the selection

## **5.2 DATA PROCESSING**

Collecting the data is one task and making that data useful is another vital task. Data collected from various means will be in an unorganized format and there may be a lot of null values, invalid data values and unwanted data.

Cleaning all these data and replacing them with appropriate or approximate data and removing null and missing data and replacing them with some fixed alternate values are the basic steps in pre processing of data. Even data collected may contain completely garbage values.

It may not be in the exact format or way that is meant to be. All such cases must be verified and replaced with alternate values to make data meaningful and useful for further processing. Data must be kept in an organised format.

This is taken care and data fetching is done in such a way that eliminates the above ills

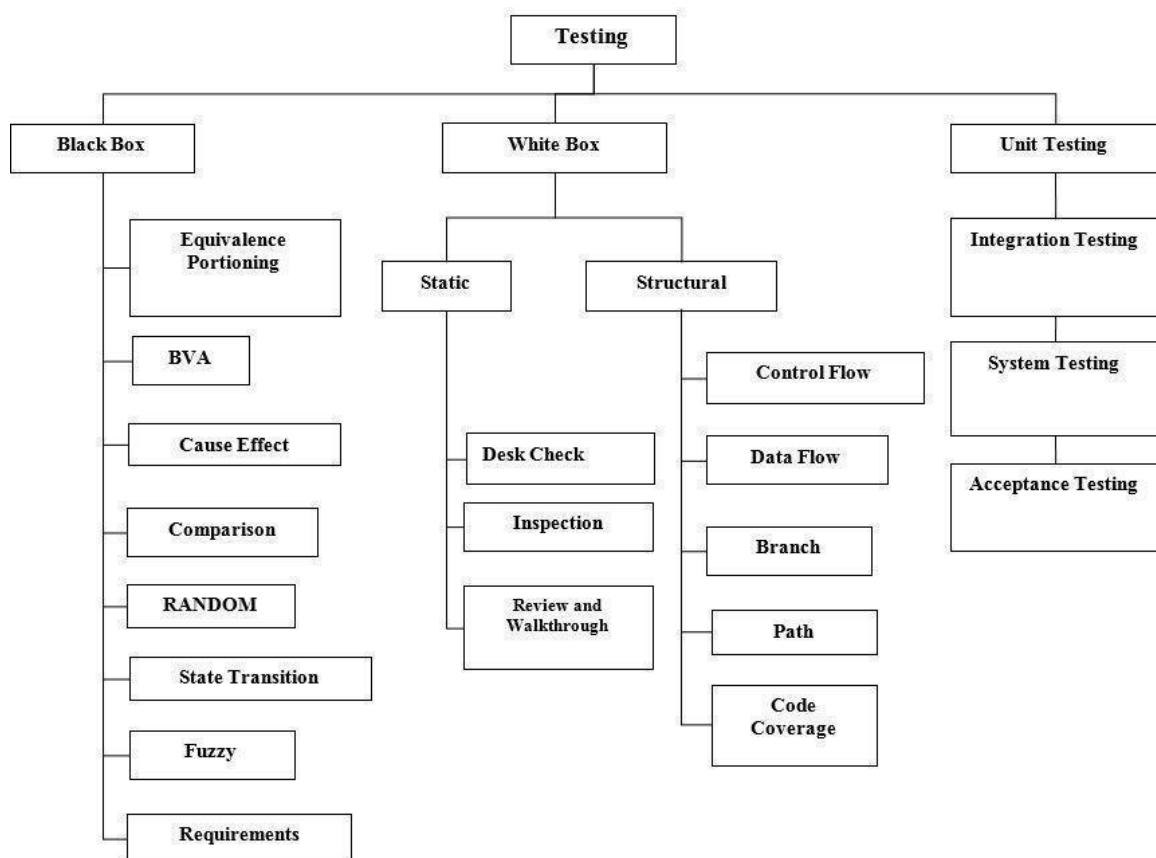
# CHAPTER 6

## TESTING

### TESTING

Generally testing process is carried out to ensure that the system has been developed according to the required specifications and the expected output is properly obtained. There are two main categories of testing namely, the White Box Testing and the Black Box Testing. Each of this testing in turn consists of many types of testing.

#### 6.1 TAXONOMY OF TESTING



**Fig 6.1 Taxonomy of Testing**

##### 6.1.1 WHITE BOX TESTING

White Box testing is also known as glass box testing. This type of testing, tests the internal structure of the program. This can be applied at the unit, integration and

system levels of testing. Mostly, it is used in the unit level of the software testing process. Sometimes it may not reveal defects in areas which have not been implemented. It has its own advantages and its own disadvantages. The advantage is that knowing the programming language code and familiarizing with them may prove vital and help in identifying the errors quickly and at times may help in avoiding them at the earliest.

### **6.1.2 BLACK BOX TESTING**

It is amongst the two methods of mostly used testing methods. This tests the main functionality of the program. It can be applied to every level of testing such as Unit, Integration, System and Acceptance levels of testing. Exhaustive input testing is required to find all errors. For doing this type of testing knowing the internal code and how it works is not needed but what it is supposed to do is known by the person who is performing the test. The test cases are developed based on the specific requirements according to the goals. There are Boundary Value Analysis, Class Partitioning, Cause Effect Graph etc.,

#### **> UNIT TESTING:**

Unit testing is also known as Module Testing which focuses on verification efforts on the module. The module is tested separately and this is carried out at the programming stage itself. Unit test comprises the set of tests performed by an individual programmer before integration of the unit into the system. This will help to test each and every single part or we can say as each and every module completely.

These may even be small parts of the code and test cases will be developed which are independent of each other.

### **>FUNCTIONAL TESTING:**

Functional testing is mainly used as a Quality Assurance process. This is a very simple process where each function is provided with an appropriate input and is verified against an expected output and with boundary values. This would help in ensuring that the output is as acquired according to the expectations and would help assuring the quality.

### **>INTEGRATION TESTING:**

It is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated within the interface. It takes the unit tested modules and builds a program structure. Integration of all the components to form the entire system and an overall testing is executed. Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design.

### **>VALIDATION TESTING:**

Validation test succeeds when the software functions in a manner that can be reasonably expected by the client. Software validation is achieved through a series of black box testing which confirms the requirements. The software is validated based on the series of tests that it passes through according to the condition posed by the customer. Mostly the customer's main requirements would be to make every process as simple as possible and to reduce the complexity of the usage of the final product. Taking all these conditions into mind the validation testing is done and the various test cases are designed.

### **>SYSTEM TESTING:**

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. Once all of the modules have been completely developed and both unit testing and integration testing is done on the various parts of the modules later the system testing is done so as to ensure that the requirements are fulfilled properly. All it basically does is it performs

tests to find the discrepancies between the system and its original objective, current specifications and system documentation. If any discrepancy is to be found the respective errors will be rectified and again system testing will be performed to make sure the rectification does not introduce a new error into the system.

### **>STRUCTURE TESTING:**

It is concerned with exercising the internal logic of a program and traversing certain execution paths. Structure testing takes into account all processes that work internally to make the entire system to work properly. The basic structure and the background codes and fragments that help in upholding the system are used.

This goes layer by layer as in till reaching the core process. The various layers or levels depends on the type of the project and the domain it comes under. This is considered to be a part of the White Box Testing Process. This is done so as to make sure all the internal processes work properly. If They don't then the probability that the entire process may collapse is a possibility and this causes grave danger to the project leading to failure.

## **6.2 TESTING IN PARTICULAR**

### **6.2.1 UNIT TESTING**

Unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. In this project, all statements are executed properly. All units of program programs are tested in different computers. And the result of the project is the same in all systems.

### **6.2.2 INTEGRATION TESTING**

**Integration testing** (sometimes called **integration and testing**, abbreviated **I&T**) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger

aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

### **6.2.3 ACCEPTANCE TESTING**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

### **TEST OBJECTIVES**

- All field entries must work properly
- Pages must be activated from the identified link
- The entry screen, messages and responses must not be delayed

### **FEATURES TO BE USED**

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

### 6.3 TEST CASES

S.NO	Function	Description	Expected output	Actual output	Status
1.	Register	Register user details	Account registration in application	Registration accepted	success
2.	Login/Logout	Login using registered data	login into application	login details accepted	success
3.	Make order	Making an order from menu	To place order	Order placed	success
4.	Track order	view status of the order	show status	status is viewed	success
5.	Data fetching	Data in each table is fully fetched	Do fetch all data correctly	Data fetched	success
6.	Menu Management	Menu in the restaurant	Show digital menu	Digital menu shown	success
7.	Report Table	All the collected data are organised into report	Generate report	Report generated	success

# CHAPTER 7

## RESULTS

### Admin module screenshots

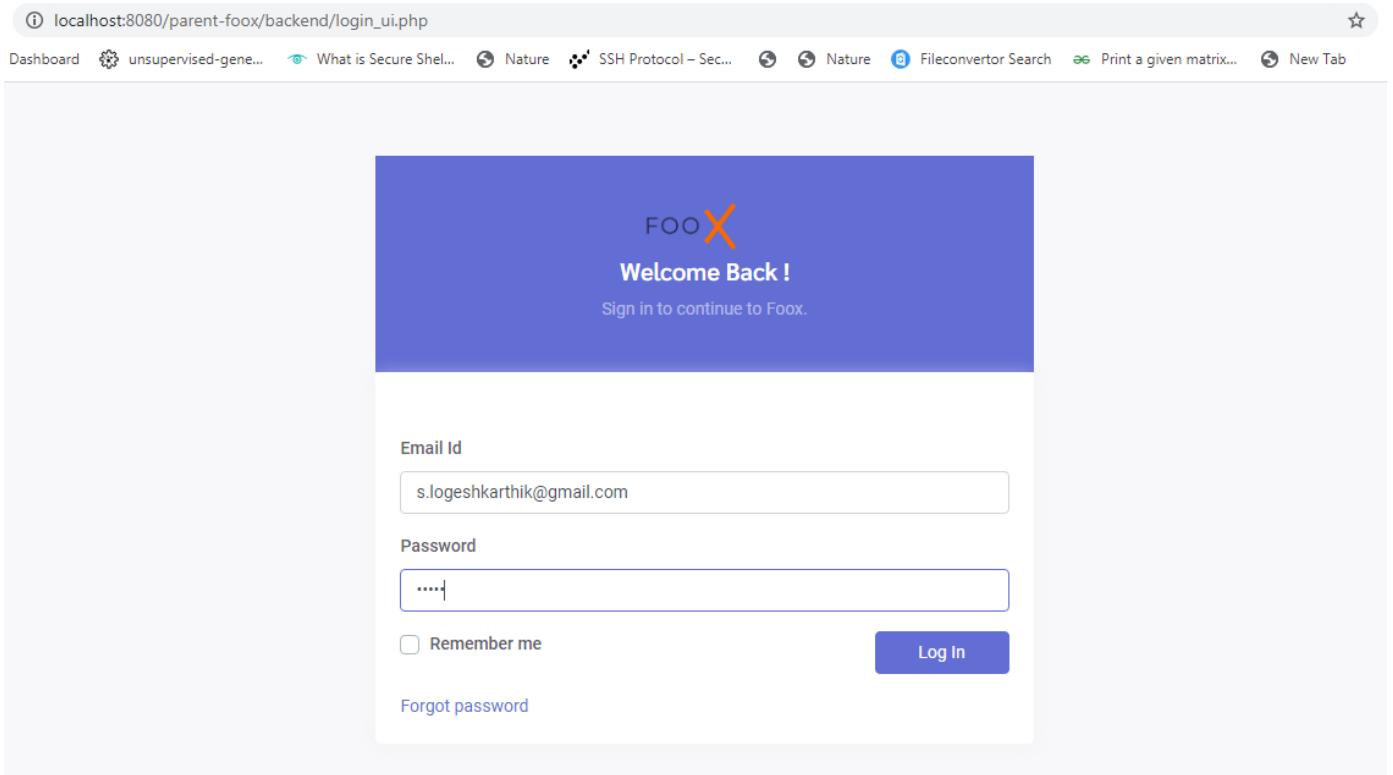


Fig. 7.1

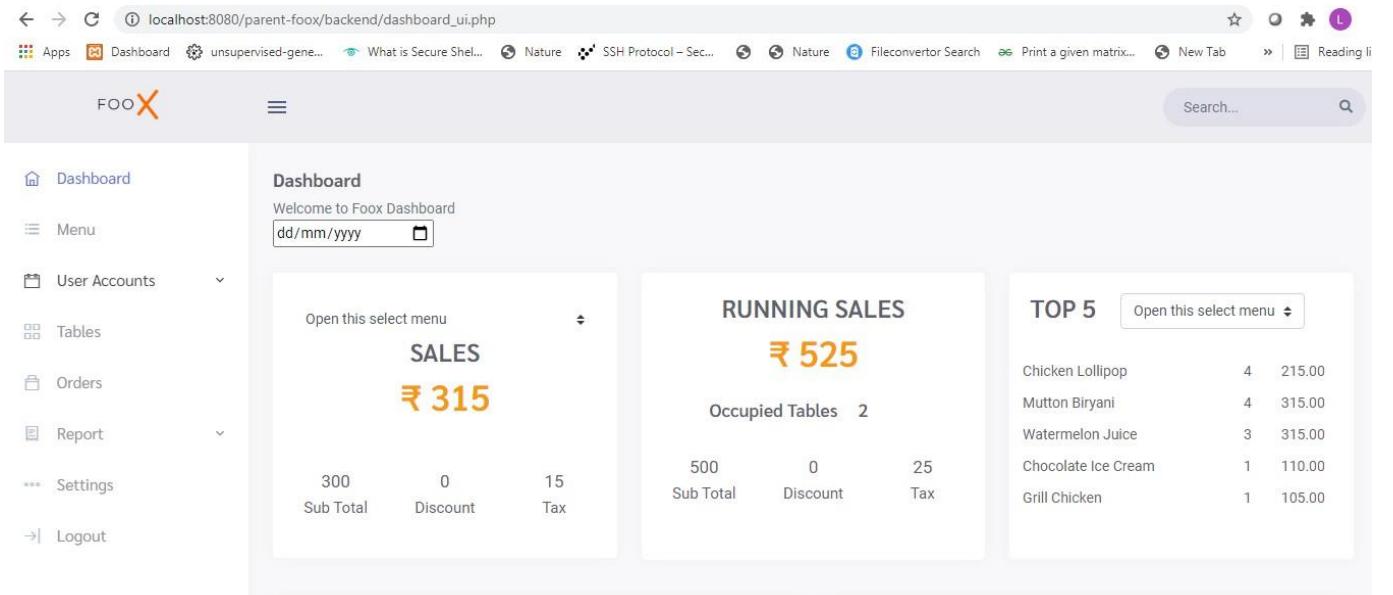


Fig. 7.2

Category	Type	Price	Availability	Action
Starter(2)	Non-Veg	105	Yes	<a href="#">Delete</a>
Dessert(2)	Non-Veg	105	Yes	<a href="#">Delete</a>
Soft Drinks(1)	Non-Veg	105	Yes	<a href="#">Delete</a>
Biriyani(1)	No			

Items	Type	Price	Availability	Delete	Actions
Chicken Lollipop	Non-Veg	105	Yes	<a href="#">Delete</a>	<a href="#">Edit</a>
Grill Chicken	Non-Veg	105	Yes	<a href="#">Delete</a>	<a href="#">Edit</a>

Fig 7.3

S.no	Department name	Department code	Employees	Resources	Edit	Delete
1	test1	TES-44202112341991	2	1000	<a href="#">Edit</a>	<a href="#">Delete</a>
2	waiter	WAI-792020184234603	5	10000	<a href="#">Edit</a>	<a href="#">Delete</a>
3	cashier	CAS-792020184218348	1	1000	<a href="#">Edit</a>	<a href="#">Delete</a>
4	kitchen	KIT-792020184155273	4	10000	<a href="#">Edit</a>	<a href="#">Delete</a>
5	Manager	MAN-792020184017643	2	1000	<a href="#">Edit</a>	<a href="#">Delete</a>

Fig 7.4

The screenshot shows the FooX application interface with the URL `localhost:8080/parent-foox/backend/roles.php`. The left sidebar has a 'User Accounts' section expanded, showing 'Departments', 'Roles', 'Employees', and 'Page Role Mapping'. The main content area is titled 'Manage Roles' with a sub-header 'Home > Roles'. It contains a table with columns: S.no, Department, Role, Edit, and Delete. The table data is as follows:

S.no	Department	Role	Edit	Delete
1	test1	test role	<button>Edit</button>	<button>Delete</button>
2	cashier1	senior cashier	<button>Edit</button>	<button>Delete</button>
3	waiter	waiter2	<button>Edit</button>	<button>Delete</button>
4	waiter	waiter1	<button>Edit</button>	<button>Delete</button>
5	cashier	cashier1	<button>Edit</button>	<button>Delete</button>

Fig 7.5

The screenshot shows the FooX application interface with the URL `localhost:8080/parent-foox/backend/employees.php`. The left sidebar has a 'User Accounts' section expanded, showing 'Departments', 'Roles', 'Employees', and 'Page Role Mapping'. The main content area is titled 'Manage Employees' with a sub-header 'Home > Employees'. It contains a table with columns: ID, Code, Name, Username, DOJ, Mail, Phone, Department, and Role. The table data is as follows:

ID	Code	Name	Username	DOJ	Mail	Phone	Department	Role
1	LST-19990708	logesh karthik	logesh	2021-04-08	s.logeshkarthik@gmail.com	9551783347	test role	test1

Showing 1 to 1 of 1 entries

Fig 7.6

**Manage Tables**

S.no	Table name	Short name	Table Capacity	isActive	Edit	Delete
1	Table 31	T3	4	Yes	<button>Edit</button>	<button>Delete</button>
2	Table 2	T2	4	Yes	<button>Edit</button>	<button>Delete</button>
3	Table 1	T1	4	Yes	<button>Edit</button>	<button>Delete</button>

Showing 1 to 3 of 3 entries

Fig 7.7

**Recent Orders**

Order ID - 18				
S. No.	Item Name	Qty	Status	Action
1	Mutton Biryani	1	<span>Delivered</span>	

Order ID - 16				
S. No.	Item Name	Qty	Status	Action
1	Watermelon Juice	1	<span>Delivered</span>	
2	Grill Chicken	1	<span>Delivered</span>	

Order ID - 14				
S. No.	Item Name	Qty	Status	Action

**Live Table Status**

T1    T2    T3

Fig 7.8

localhost:8080/parent-foox/backend/daily\_report.php

The screenshot shows a web application interface for a food ordering system. On the left is a sidebar with navigation links: Dashboard, Menu, User Accounts, Tables, Orders, Report (with sub-links: Daily Report, Monthly Report, Items Report, Tables Report, Category Report), and Settings. The main content area has a header "Daily Report" with a breadcrumb trail: Home > Report > Daily Report. Below this is a search bar with fields for "Start Date" and "End Date", and buttons for "Search" and "Clear". The central part of the screen is titled "Daily Summary" and contains a table with the following data:

S.No.	Order Date	Total Orders	Total Customers	Amount without GST	Total GST	Total Amount
1	2020-09-23	1	1	300	15	315
2	2020-12-31	1	1	200	10	210
3	2021-01-03	2	2	300	10	310
4	2021-01-04	2	2	200	10	210
5	2021-01-08	1	1	300	5	305
	2021-02-28	1	1	100	5	105

Fig 7.9

## Customer module screenshots

The three screenshots illustrate the customer-facing features of the system:

- Login Screen:** Shows fields for Name (LOGESH KARTHIK), Email address (s.logeshkarthik@gmail.com), Number of Guests, and Phone number (9551783347). A large orange "GET OTP →" button is at the bottom.
- OTP Verification Screen:** Displays a text input field with the value "5367" and an orange "CONFIRM →" button below it.
- Food Items Screen:** Shows a list of items with their details and "ADD +" buttons. The items listed are:
  - Mutton Biryani (Category: , Rs 100.00)
  - Watermelon Juice (Category: , Rs 100.00)
  - Grill Chicken (Category: Starter, Rs 100.00)
  - Half Pkg (Category: , Rs 100.00)

Fig 7.10

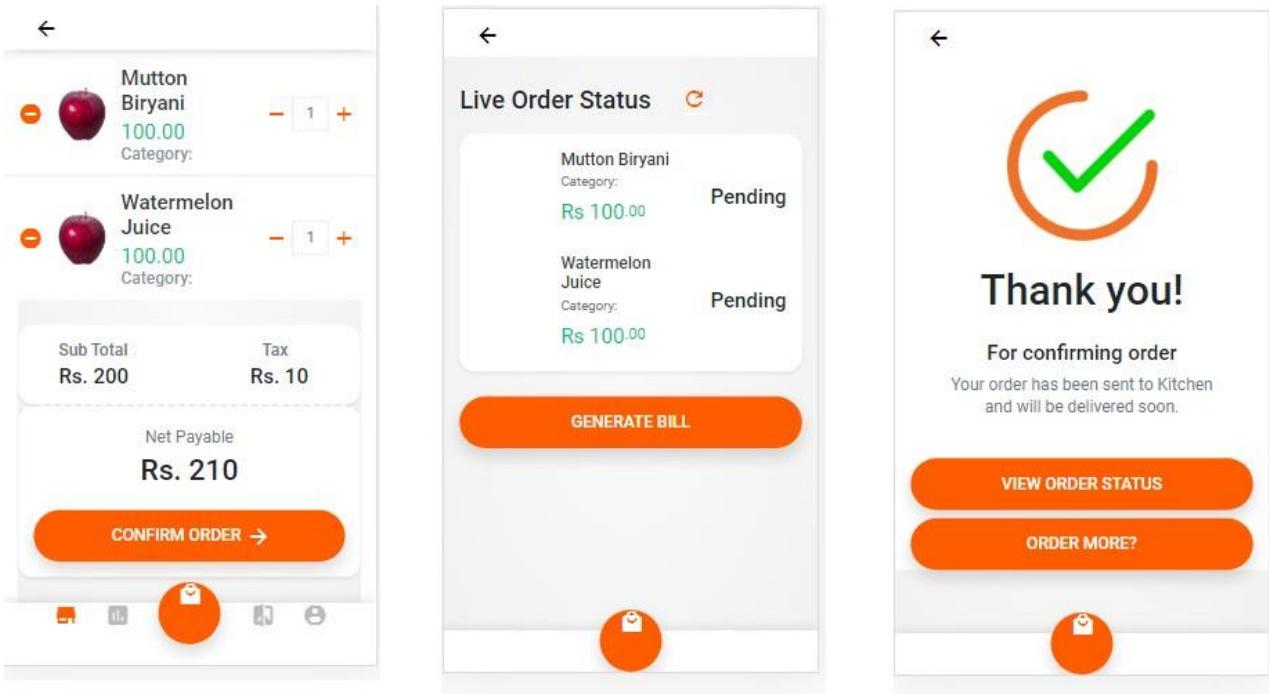


Fig 7.11

## **CHAPTER 8**

### **CONCLUSION AND FUTURE ENHANCEMENT**

#### **8.1 CONCLUSION**

Nowadays different types of management systems are available, at the same time they are not following contactless dining norms which is the recent need. Food industry is spread worldwide and restaurants are our focus. One of the main challenges in this area is safety which is fulfilled by contactless practices. Hereby we conclude that our project will provide a better futuristic solution for advanced management of restaurants considering current needs and efficient data processing and manipulation.

#### **8.2 FUTURE ENHANCEMENT**

This paper has focused on an important aspect of High performance restaurant management and operational tools. In future even more upcoming features and needs for the time are to be focussed and implemented in order to make the application more efficient to the need of the time.

## **CHAPTER 9**

### **REFERENCES**

- [1] Noor Azah Samsudin, Shamsul Kamal Ahmad Khalid, Mohd Fikry Akmal Mohd Kohar, Zulkifli Senin, Mohd Nor Ihkasan; “A Customizable Wireless Food Ordering System With Real-Time Customer Feedback.”; 2011 IEEE Symposium on Wireless Technology and Applications (ISWTA), September 25-28, 2011, Langkawi, Malaysia
- [2] Sakari Pieskä, Markus Liuska, Juhana Jauhainen, and Antti Auno Of Centria University Of Applied Sciences Ylivieska; “Intelligent Restaurant System Smart Menu That Digital Technology”; coginfocom 2013 4th IEEE International Conference on Cognitive Infocommunications December 2–5, 2013 , Budapest, Hungary.
- [3] Ching-suchang, Che-chen Kung, Tan-hsu Tan,”Development And Implementation Of An E-Restaurant For Customer-Centric Service Using Wlan And Rfid Technology”,proceedings of the Seventh International Conference On Machine Learning And Cybernetics, Kunming, 12-15 July 2008
- [4] Soon Nyeon Cheong, Wei Wing Chiew, And Wen Jiun Yap; “Design And Development Of Multi-Touchable E-Restaurant Management System “; 2010 International Conference On Science And Social Research (Cssr 2010),December 5 - 7, 2010, Kuala Lumpur, Malaysia
- [5] Nilam Kadale , Pranjali Bansod, Reshma Pillai, Shivangi Sane, Snehal Pratape, Swati Pawar ; “ Wireless Customizable Food Recommendation System Using Apriori And K-Means Algorithm .”; IJLTEMAS , Volume IV, Issue X, October 2015
- [6] Varsha Chavan, Priya Jadhav, Snehal Korade and Priyanka Teli of Computer Department,Pune University Indapur, Maharashtra,India; “Implementing Customizable Online Food Ordering System Using Web Based Application”; IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 4, April 2015.
- [7] Jiawei Han and Micheline Kamber, “Data Mining: Concept And Techniques”, University of Illinois.
- [8] M. Z. H. Noor, A. A. A. Rahman, M. F. Saaid, M. S. A. M. Ali, M. Zolkapli —The Development of Self-service Restaurant Ordering System (SROS)॥ 2012IEEE Control and System Graduate Research Colloquium (ICSGRC 2012)

## **CHAPTER 10**

## **APPENDICES**

### **APPENDIX A- BASE PAPER**

## CORMS: AN AUTOMATED RESTAURENT MANAGEMENT SYSTEM

Roy Davis<sup>1</sup>, Ninu Francis<sup>2</sup>, Swathi K.Sukumaran<sup>3</sup>, Swetha Jeevan E<sup>4</sup>, Umesh Nair<sup>5</sup>

<sup>1</sup>Dept. Of CSE, Jyothi Engineering College, Kerala, India

<sup>2</sup> Professor, Dept. of Computer Science Engineering, Jyothi Engineering college, Kerala, India

<sup>3</sup>Dept. Of CSE, Jyothi Engineering College, Kerala, India

<sup>4</sup>Dept. Of CSE, Jyothi Engineering College, Kerala, India

<sup>5</sup>Dept. Of CSE, Jyothi Engineering College, Kerala, India

\*\*\*

**Abstract** - Customer Oriented Restaurant Management System (CORMS) is a web application and an android version to restaurant management. It has all the features of the rapid involving science and its different attributes. Through a strategic design and customer orientation, CORMS technology is integrated and has been created to optimize the work force and streamline restaurant work flow. It can run on a tablet, and is both scalable and modular to meet the needs of any establishment. CORMS is an effort to bring technology into the dining menu of customers. CORMS offer robust features that not only help your restaurant to update the menu any time but also improve the overall dining experience. The tablet menu is to provide a user-friendly interface by offering smooth navigation and browsing through digital menu thus providing a delightful experience. The customers can order the food, through that tablet interface. Our project aims to not only improve the business of restaurants but also to incorporate the essence of science in dining menu. Our future-ready restaurant management software is designed to keep track of everything that goes inside the restaurant, and everything is permission based to avoid theft.

**Key Words:** CORMS; Restaurant Management System; android; digital

### 1. INTRODUCTION

Visiting a restaurant traditionally involves selecting a meal from a paper based menu and being waited on by the restaurant's waiter staffs. A busy restaurant or inattentive staff can leave customers waiting to have their orders taken, to refill their drinks or to receive their bill for a long time. If the restaurant is busy the customer is left there, where he occupy a table longer than they need. Any unnecessary waiting can reduce customer satisfaction and reliability and ultimately result in lost business. To reduce customer wait, prior management of time must be ensured. Sufficient staffs should be present during peak hours and that they are properly trained to provide excellent customer service. These staffing issues can lead to substantial costs for the business. Paper based menus are

problematic. The restaurant may have a large number of menu items which can make the menu appear overwhelming to go through it. As a result, customers may not see all the items they would have been interested in. When changes to the menu are required, such as price adjustments or quantity change or item updates, the costs and environmental concerns associated with reprinting and all need to be considered. Menu changes are often left to accumulate until enough are required to justify the costs of reprinting. Changes may be required frequently and a paper menu would quickly become outdated. Waiting until a reprint is done before implementing the changes in the restaurant may not be a sound business practice. Manually updating the menus instead of reprinting can lead to inconsistencies and this can give a bad impression to the customers. This may make the restaurant appear cheap and low quality. The project is designed and is building a restaurant management system that provides an interactive tablet based menu which replaces the paper menu entirely and removes much of the need to be waited on by the restaurant's waiter staff. This tablet based menu app also provides additional features designed to enhance the customer's overall experience. In the management side, it allows the restaurant's management to quickly make changes to the menu and provide a larger view of the restaurant at any given time. The restaurant menu and management system consists of the menu app, the management app, the web based site, the server and a database. Other apps, intended to be used by the restaurant's kitchen and wait staff were not developed for this project.

### 2. MOTIVATION AND TECHNICAL RELAVANCE

The mobile market is growing in each year replacing the demand for traditional desktop applications. This makes software development for mobile devices an interesting and attractive industry to work in. The primary motivation for this project stems from the desire to learn and gain experience in android apps and web sites development as well as an interest in the design and development of distributed systems. The paper based menu system is very tiresome and need lots to wait by the

customers to get noticeable by a waiter in a busy restaurant. And hence such a system will surely be a blessing for both customer and restaurant owners, which motivated us to work on our project more.

### 3. RELATED WORKS

The world has contracted with technology. Technology had affected the restaurants with greater impact. RFID technology, digital menus, service robots and others are some examples of advanced technology are coming to the future restaurants. However, the restaurant service process has to stay customer-centred and it will mainly include human service also in the future

Customizable Wireless Food Ordering System with Real-Time Customer Feedback [1] is discussing, the design and implementation of a customizable wireless food ordering system with the help of a real-time customer feedback for a restaurant (CWOS-RTF). The CWOS-RTF enables restaurant owners to set-up the system in wireless environment and update menu presentations easily. Smart phone has been integrated in the CWOS-RTF. Instead of using PDA's to interface with customers, they leverage smart phones to provide necessary interfaces for customer to view and order menu. With private login system, customers can view and make order and receive updates in real-time and collect receipts right from the smart phone itself. It allows restaurant owners to manage orders from customers instantaneously whenever he or she logged in into the system. The existence of wireless technology and the emergence of mobile devices enable a simple yet powerful infrastructure for business application. Some early efforts have been made to utilize both technologies in food ordering system implementations. However, the food ordering systems that have been proposed earlier exhibit limitations, primarily in cost effectiveness, allowing customizations and supporting real-time feedback to customer's implementation to facilitate real-time communication between restaurant owners and customers. A preliminary testing suggested that the CWOS-RTF has the potential to eliminate the limitations of existing food ordering systems.

The objectives of the proposed system are:

- To automate food ordering system at Restaurants so that it can eliminate or at least minimize the current problems in conventional system.
- To utilize wireless communication and smart phone technology in implementing the automated system.
- To facilitate more intuitive interfaces and customization for the restaurant owner to update the menu content on the customer devices.
- enable real-time feedback between the restaurant owner and customers on the order

status.

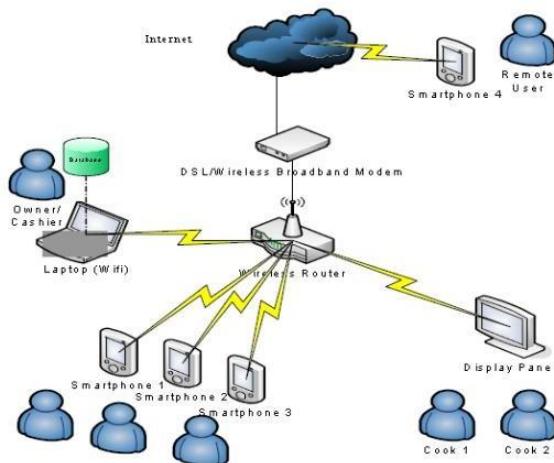


Fig I: Architectural diagram of CWOS-RTF with internet access

The CWOS-RTF is built on four main components:

- The mobile application on the smart phones for customers to make order.
- The web based application and server on the laptop for restaurant owner to keep track and respond to received customer's orders, and customize menu information.
- The database for restaurant owner to store order details, and updated menu information.
- The wireless infrastructure to support networked communications. This system can be extended as running CWOS-RTF on more restaurants and customers to report on their acceptance. Although current interface (New Order List) can be used by the staff in the kitchen, the system can be further enhanced by adding inventory management module for the kitchen staff. Besides this, a module for remote delivery can be added for bigger customer coverage. Finally, the system can be extended to register and link multiple restaurants for more food and beverage varieties to the customers.

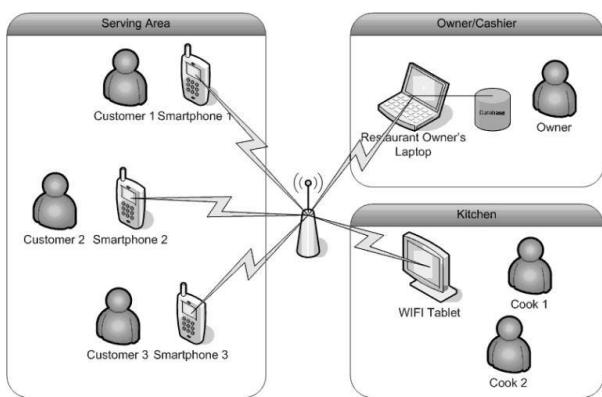


Fig II: Architectural diagram of CWOS-RTF without internet access

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 requires that the customer has to be seated before starting. An alternative method for the customers is described through Implementation Customizable Ordering Food Ordering System Using Web Based Application [2] "A Food Pre-Order System using Web Based Application "in which customer can be able to create the order before they approach the restaurant using Smart phone. When the customer approaches the restaurant, the saved order can be confirmed by touching the Smart phone. The list of selected pre-ordered items shall be shown on the kitchen screen, and when confirmed, order slip shall be printed for further order processing. The solution provides easy and convenient way to select pre-order transaction form customers. The Objectives of this system is:

- To combine Wireless technology and Android OS to automate food ordering process.
- To minimize the imperfection in conventional system by reducing the working of a restaurant.
- To make provisions for obtaining feed-back from the customers and provide the restaurant a means of review of their service.
- To utilize wireless communication and smart phone technology in implementing the automated system.
- To make more user interfaces friendly and customization for the restaurant owner to update the menu content on the customer devices.
- To enable real-time feedback between the restaurant owner and customers on the order status.

Conceptually this system is built using following components:

- The android application is used to make orders from tablet.
- The restaurant-owners laptop/tablet will keep track of customer records and also customize menu using server application.
- The central database is used for restaurant-owner to store updated menu information and order details.
- Three main areas of restaurant are connected using wireless technology.
- The Android application is used to find out the location in restaurant according to its latitude and longitude.

The technologies which are used to implement the system are:

1. Visual Studio 2010 for developing web application.
2. Android version 2.2 or more for Tablets is required.
3. SQL 2008 is a light weight Database which is going to be used for database access from the tablet.

The existence of wireless technology and the emergence of mobile devices enable a simple yet powerful infrastructure for business application like restaurant management system. Technology can be deployed efficiently to manage all the day-to-day tasks in restaurants. By using a new software-oriented approach we can eliminate a number of counters leading to a savings in space as well as staff for a restaurant .This new approach is a one-time investment as we do not have to pay salaries to a very large staff. The new system Wireless Customizable Food Recommendation System Using APRIORI and KMEANS Algorithm [3] is useful at places where it is difficult to find employees and where labour rates are sky-rocketing as the day passes. Moreover in the present day environment where numbers are a measure of progress, there is a need for tools to analyse patterns which can be exploited for designing new sales strategies.

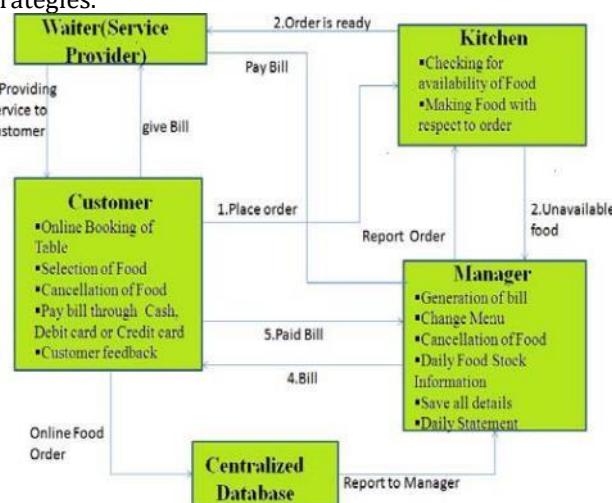


Fig III: Architectural Diagram of Customizable Ordering Food Ordering System

The "Wireless customizable food recommendation system using Apriori and K-means algorithm." is a system where we are using Apriori and k-means algorithms for analysis. The Apriori algorithm forms the core of the recommendation system, prompting customers to try popular dishes. The purpose of using k-means is to classify the customers according to their expenditure. Therefore, the proposed system provides automation along with analysis. The proposed system provides automation for Kitchen order ticket (KOT), billing and Customer Relation Management (CRM) as will be seen in the later sections of the paper.

#### Summary of system functionalities

##### A. Tablet on table

- There will be a tablet on each table.
- This will allow the customers to browse the food items as many times as they wish.
- Customer can view the suggestions for a particular menu item generated by the system.
- Customer can enter his/her details during bill payment. This helps the Restaurant owner to analyse the service and can notify the customer regarding different offers through messages or emails.
- Suggestions for Customer
- The Restaurant owner can post various combinations of menu items on tablet. This will help the customer to place the best order.

##### B. Attractive Presentation

- The Menu is organized in an attractive way.
- There are images of every food item which will make the view of customers more clearly about how the food will look like after delivery.
- There is an attractive use of various themes and colour schemes.

##### C. Modifiable Menu

- The menu can be modified by the Admin manager.
- Admin manager can add, update, delete menu items.

#### Algorithm K MEANS

- 1) Clusters the data into k groups where k is predefined.
- 2) Select k points at random as cluster centres.
- 3) Assign objects to their closest cluster centre according to the Euclidean distance function.
- 4) Calculate the centroid or mean of all objects in each cluster.

- 5) Repeat steps 2, 3 and 4 until the same points are assigned to each cluster in consecutive rounds.

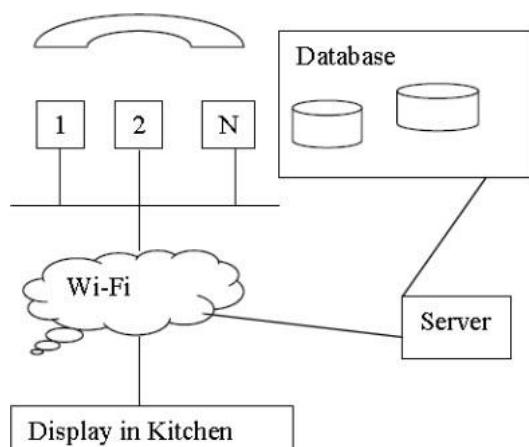


Fig IV: Architectural Diagram of Wireless

Customizable Food Recommendation System  
Using WLAN and RFID Technology Traditional restaurants only provide passive service where waiter can only deal with customers order by asking customers need and then waits for answer. However, a high quality service system should be customer-centred, i.e., customer's identity and therefore his/her favourite meals and expenditure records in past days can be immediately recognized by service system so as to provide customer-centric services. To achieve this goal, by Development and Implementation of an E-Restaurant for Customer-Centric Service [5], this study integrates RFID and wireless local area network (WLAN) technologies to implement an e-restaurant for customer centric service, which enables waiters to immediately identify each customer via his/her own RFID-based membership card and then actively provides customized services. The user interface of the proposed system is built with Visual C 2005 and embedded Visual C++, and the database is built on Microsoft SQL Server 2005 for server management and statistic reporting. WLAN and RFID are used to transmit the real-time information of each dining table. A high quality service system should be customer-centred, i.e., customers identity and therefore his/her favourite meals and expenditure records in past days can be immediately recognized by service system so as to provide customer-centric services. To achieve this goal, we integrate RFID and wireless local area network (WLAN) technologies to implement an e-restaurant for customer-centric service, which enables waiters to immediately identify each customer via his/her own RFID-based membership card and then actively provide customized services. Also, customers can use the RFID-based membership card to pay the bill instead of using cash. Moreover, to facilitate waiters dining table service, this research develops portable service unit on personal digital assistant (PDA). By means of the PDA-based service

unit customers order can be instantly transmitted via WLAN to the kitchen for meal preparation. Also, the expenditure information can be sent to the counter for pre-processing of bill. The restaurant managers can access to the database for mastering the business status any time and making appropriate redeployment for food materials.

#### 4. PROPOSED SYSTEM

CORMS is an integrated system that helps customers to select their favourite with a personal touch that is ,which helps the restaurant to make a personal data file about customer is referred ,And these data is used in the future to refer the same customer to repeat his visit and to use the data once analysed and make the best choice of food. It is a network based central control system that can be used in restaurant, resorts, etc. The network uses three participants, where customers and kitchen manager becomes the client participant and the manager become the server participant in the TCP/IP network system .It is a tab oriented application that replaces menu list and waiters which includes new update for restaurant management. Each table comes with a tab where customer makes the cart and sends the information to both main control system (receptionist or manager) and kitchen manager, And finally the whole data is updated and uploaded to the main server of the central database which makes the restaurant to manage the current business of the restaurant data The implementation includes client-server networking, TCP/IP protocols, wireless communication without internet, Database etc. The different modules associated within are

**1. User Registration Module :** Through this module the user creates a simple account, in which billing details, order details is saved and is used for future recommendation.

**2. User Ordering Module:** This module handles ordering of food, which will have an interface, includes dish recommendation, dish comment, top dish list, dish diet list, dish cancellation, etc. The ordered dish status is automatically added to a web server and sends to both kitchen manager and Manager

**3. Kitchen Module:** In this module, the chiefs will get the order details from the customer, where he will display the current status of cooking; it will also display the current table of the customer.

**4. Manager Module:** Manager Module will do a current updation of customer's food, billing, where he connects the data to the server. Manager manages the order cancellations of the customer and helps the chef to cook according to that.

Advantages of proposed system:

The following are the advantages of CORMS

- Software with real time customer interaction.

- It can replace both waiter and paper based menu.
- Customers feel free to express their opinion about food.
- Restaurant owners will be able to get a chance to improve through the feedback of customers.
- Time saving process is that the cook and all other staffs works in fixed time limit.
- Up-to-date information about the ordered dish can be made available to customer.
- Dish cancellation can be done with in a specific time.
- Diet based life style can be followed with such a system.
- Restaurants become friendlier to customer.

The figure shows the architectural diagram of CORMS. It shows how each module is related to each other. Each module is connected to a database where every data is stored. The central server controls the overall working of the system. The details about the food ordering, food comments etc. are stored in the database. The user register themselves in the user registration module with their name ,contact details food diet details etc. and order their food . The ordering is a part of ordering module where the application compares the previous history and food recommended according to user's profile. Next step is to prepare the food according to customers wish. For this the data entered by the user is send to the chef via a central server and he prepares the food. All the details about cooking is visible to the customer. The customer may cancel some dishes and the details of cancellation are updated in the server, further send to the cook. At last these details are send to the manager section to bill the expense and a hard copy is delivered to the customer at the same time soft copy is provided via a tablet application .This comes under the manager module.

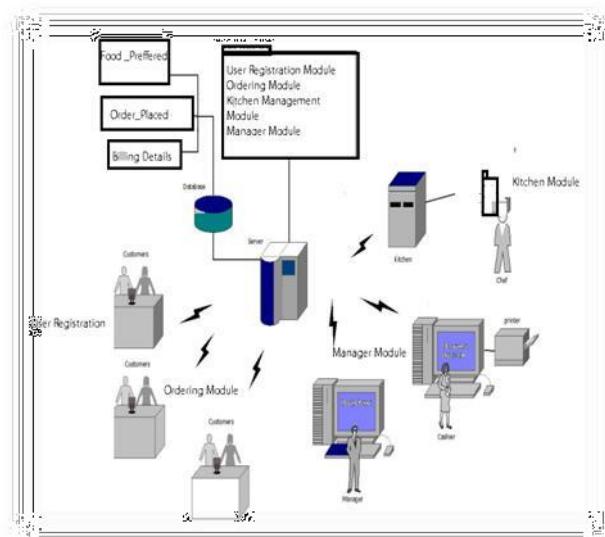


Fig V: Architectural Diagram of CORMS

## 5. SYSTEM REQUIREMENTS & SPECIFICATION

Here we are including the software's and hardware's used for developing the project and implementing the project.

### A. Software Requirements

1. Android SDK (Software Development Kit)
2. JDK (Java Development Kit)
3. Android Studio
4. Expression Web 4.0
5. WAMP Server

### B. Hardware Requirements

Android Tablet

- Version : Minimum 2
- Storage : 2 GB
- CPU : 1 MB

PC with Windows OS

- Storage : 2 GB
- CPU : 1 MB
- Android SDK is used to create an android platform in the system. JDK is the Java
- Supporting package.

ADT plug in or AVD managers are using to create emulator within the IDE.

### C. Technologies Utilized

In this section we are including the technologies and software used for our project development phase.

#### 1. Android Platform

Android is an operation system based on the Linux with Java programming interface. It provides tools, example a compiler, debugger and a device emulator as well as its on Java Virtual Machine (Dalvik Virtual Machine- DVM). Android is officially guided by the Open Handset Alliance but in reality Google leads the project. Android supports 2-D and 3-D graphics using the Open GL libraries and supports data storage in a SQLite database. Every Android applications runs in its own process and under its own user id which is generated automatically by the Android system during deployment. Therefore the application is isolated from other running applications and misbehaving applications cannot easily harm other Android applications. The Android SDK provides the tools and API's necessary to begin developing applications on the Android platform using Java programming language.

### 2. GitHub

GitHub is a web-based hosting service for software development projects that uses the Git revision control system. GitHub offers paid plans for private repositories, and free accounts for open source projects. As of May 2011, GitHub was most popular open source code repository site. The site provides social networking functionality such as feeds, followers and the network graph to display how developers work on their versions of a repository. GitHub also operates a paste bin style site called Gist, wikis for individual repositories, and web pages that can be edited through a Git repository.

## 6. SYSTEM IMPLEMENTATION

The system is implemented in two phases. The first phase of implementation includes the development of the web application, which is the interface for restaurant manager and staffs. The web page is developed using Php and Mysql is the database used here for processing the data.

Through this interface, the restaurant manager, kitchen manager, admin can login through his account and do the neediest to the customers. The various services offered to the customer by the reception manager includes reservation of table for the customer, making of bill for each customer, updating the list of food item, maintaining the user profile etc.. The admin has the power to update the food table, which stores the food related data and the user table, which stores the user data. He can also update the offer list. All these data is updated to the central server. The kitchen manager is responsible for preparing the food items that the customer order through the tab interface and he also update and return the time of progress of food to the customers.

The second phase of implementation includes the development of the tab interface, which is purely android version of the restaurant management system CORMS. This tab interface is through which the user creates his personnel account, orders his food by making his cart, displays his food etc. during the creation of the personnel account, the user can add his personal details regarding his age, address, diseases he/she have and so on. So using these details CORMS present the customer his recommendation list, which contains the food items good for his health or something according to the customer's age group.

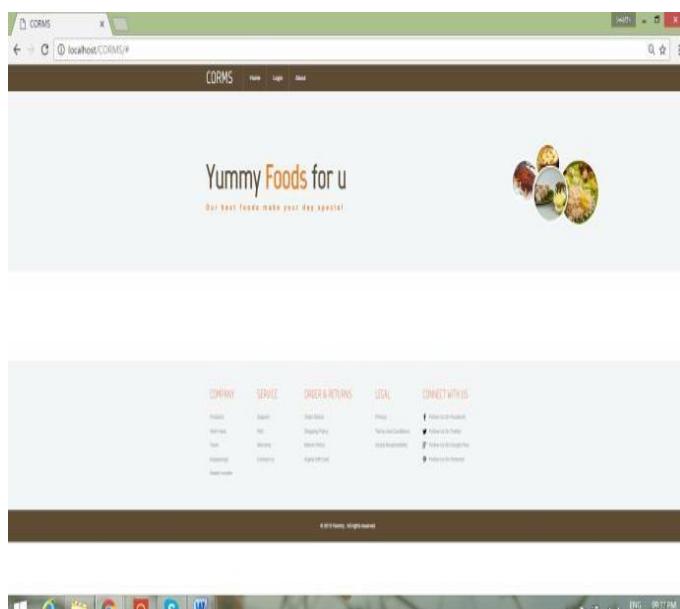


Fig VI: Screenshot of Home page of CORMS

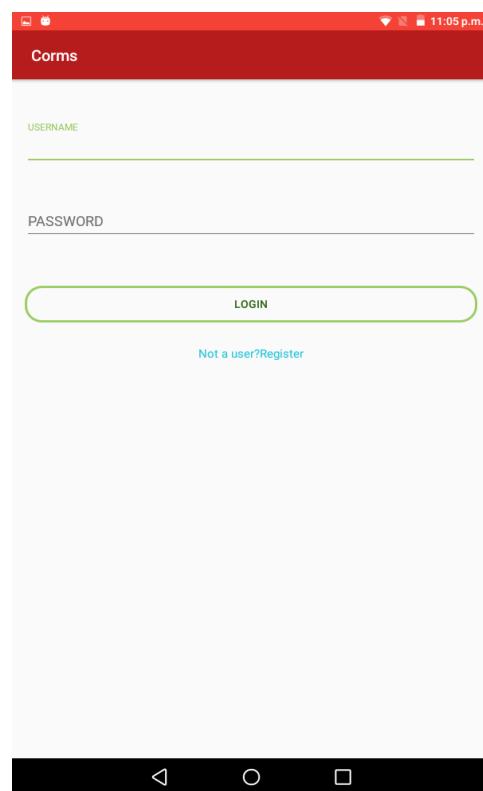


Fig VIII: Screenshot of Login page of CORMS App

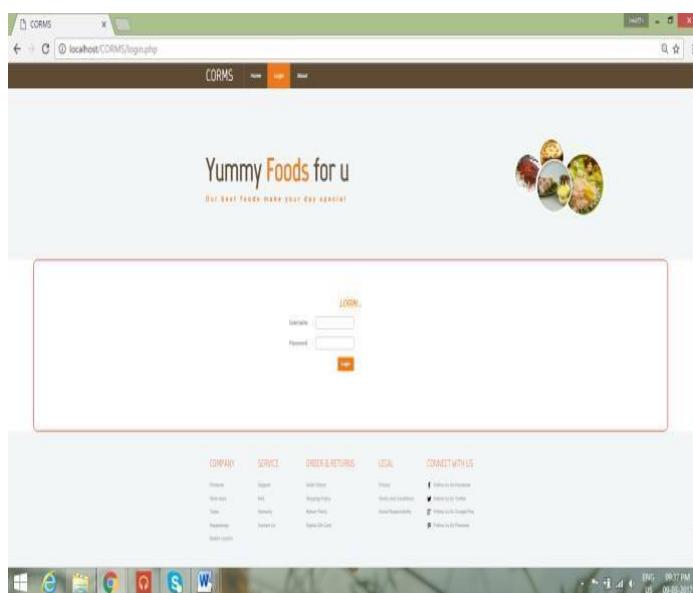


Fig VII: Screenshot of Login page of CORMS



Fig IX: Screenshot of Menu List of CORMS App

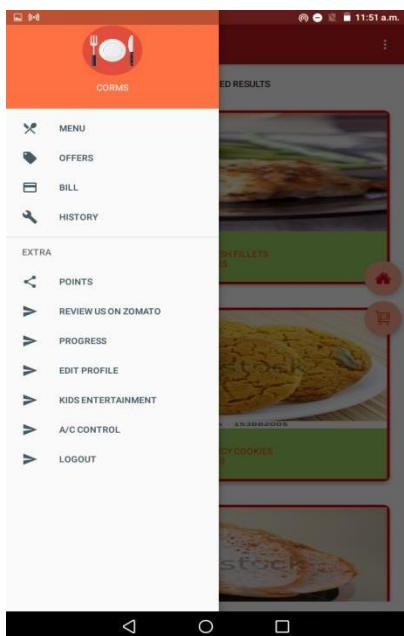


Fig X: Screenshot of Navigation bar of CORMS App

## 7. CONCLUSION

The urge for the digital restaurant management systems is increasing day by date. 'Customer Oriented Restaurant Management System 'named as CORMS is a perfect solution for this. Through this the ease of access and flexibility of the day to day works in the restaurant is made

[4] Soon Nyean Cheong, Wei Wing Chiew, And Wen Jiun Yap; "Design And Development Of Multi-Touchable E-Restaurant Management System "; 2010 International Conference On Science And Social Research (Cssl 2010), December 5 - 7, 2010, Kuala Lumpur, Malaysia

[5] Nilam Kadale , Pranjali Bansod, Reshma Pillai, Shivangi Sane, Snehal Pratape, Swati Pawar ; "Wireless Customizable Food Recommendation System Using Apriori And K-Means Algorithm ."; IJLTEMAS , Volume IV, Issue X, October 2015

[6] Varsha Chavan, Priya Jadhav, Snehal Korade and Priyanka Teli of Computer Department,Pune University Indapur, Maharashtra,India; "Implementing Customizable Online Food Ordering System Using Web Based Application"; IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 4, April 2015.

[7] K. J. Patel, et al., "Pda-Based Wireless Food Ordering System For Hospitality Industry - A Case Study Of Box

simpler. The features such as dish recommendation and rating make this software user friendly. Both the management side and worker site can manage the data easily using such a system. It is very good and reliable system which can be incorporated to the chain of hotels so can easily maintained and addressed.

## REFERENCES

- [1] Noor Azah Samsudin, Shamsul Kamal Ahmad Khalid, Mohd Fikry Akmal Mohd Kohar, Zulkifli Senin, Mohd Nor Ihkasan; "A Customizable Wireless Food Ordering System With Real-Time Customer Feedback."; 2011 IEEE Symposium on Wireless Technology and Applications (ISWTA), September 25-28, 2011, Langkawi, Malaysia
- [2] Sakari Pieskä, Markus Liuska, Juhana Jauhainen, and Antti Auno Of Centria University Of Applied Sciences Ylivieska; "Intelligent Restaurant System Smart Menu That Digital Technology"; coginfocom 2013 4th IEEE International Conference on Cognitive Infocommunications December 2-5, 2013 , Budapest, Hungary.
- [3] Ching-suchang, Che-chen Kung, Tan-hsu Tan,"Development And Implementation Of An E-Restaurant For Customer-Centric Service Using Wlan And Rfid Technology",proceedings of the Seventh International Conference On Machine Learning And Cybernetics, Kunming, 12-15 July 2008
- Hill Institute," in Wireless Telecommunications Symposium 2007, Pomona, CA, 2007, pp. 1-8.
- [8] Jiawei Han and Micheline Kamber, "Data Mining: Concept And Techniques", University of Illinois.
- [9] M. Z. H. Noor, A. A. A. Rahman, M. F. Saaid, M. S. A. M. Ali, M. Zolkapli –The Development of Self-service Restaurant Ordering System (SROS)|| 2012 IEEE Control and System Graduate Research Colloquium (ICSGRC 2012)
- [10] C. Chen, and Y, Guan, "Experience design of the theme restaurant, make the dining be a memorable experience," in Proceedings of 9<sup>th</sup> International Conference on Computer-Aided Industrial Design and Conceptual Design, Shanghai, China, pp. 982–985, November 2008.
- [11] <https://github.com/>
- [12] <https://www.android.com>

## APPENDIX B – SOURCE CODE STRUCTURE

### Admin and Customer Module Source Code Structure

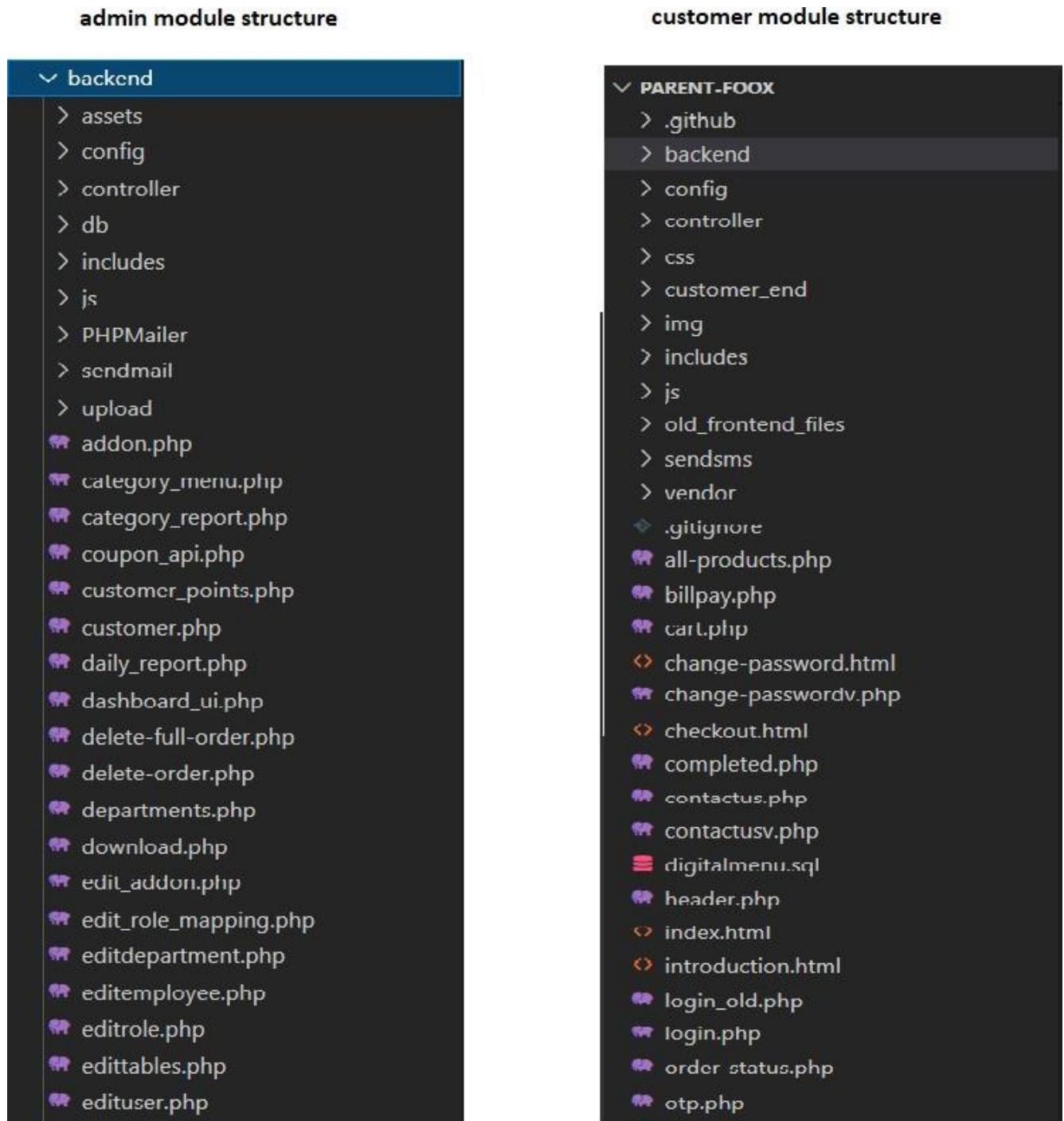


Fig 10.1

## Database Structure

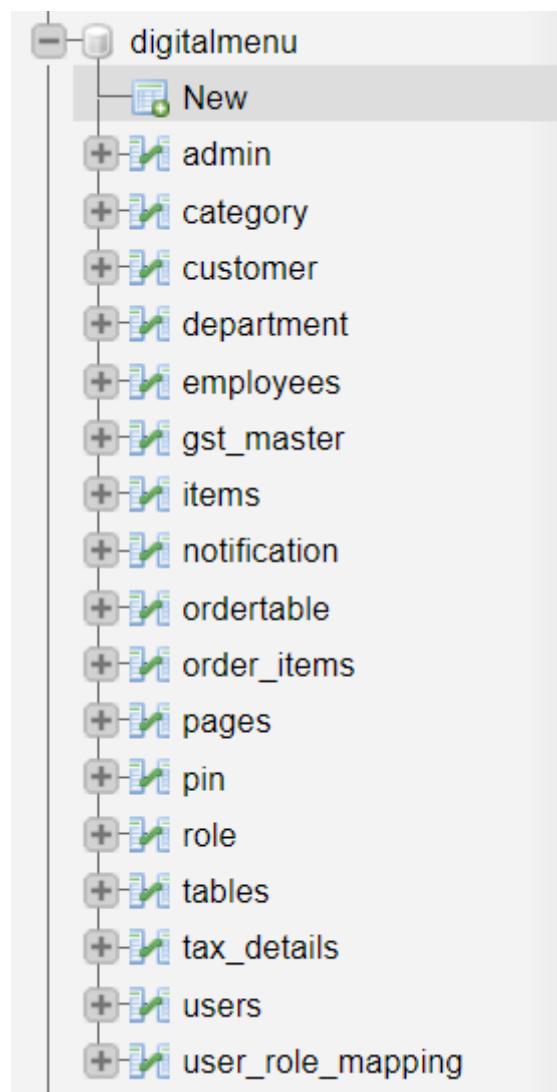


Fig 10.2

## **APPENDIX C- SOURCE CODE**

## SOURCE CODE

**FILE NAME:** common\_controller.php ( MAIN FILE FOR ADMIN MODULE)

```
<?php
ini_set('display_errors', 1);
ini_set('display_startup_errors', 1);
error_reporting(E_ALL);
include_once('../config/dbconn.php');
include_once('../config/constant.php');
include_once('config/de-encrypt.php');
//include_once('../sendmail/trigger.php');
include_once('../PHPMailer/PHPMailerAutoload.php');
// ini_set('display_errors', '1');

session_start();
$type = $_REQUEST['Type'];

if($type == 'add_category'){
    if($_REQUEST["cat_id"] == ""){
        $add_category = gosql("INSERT INTO category
(restaurant_id,name,description,is_available,createdon,createdby) VALUES
(\".$_REQUEST["catname"].\"",".$_REQUEST["catdesc"].\"",".$_REQUEST["is_available"].\"",now(),'1')");
    }else{
        $update_category = gosql("UPDATE category SET name =
\".$_REQUEST["catname"].\"",description = \".$_REQUEST["catdesc"].\"",is_available =
\".$_REQUEST["is_available"].\" WHERE id = \".$_REQUEST["cat_id"]."\"");
    }
}
else if($type =='fetch_category_details')
{
    $cat_id = $_REQUEST['cat_id'];
    $fetch_details = return_single("SELECT * from category where id = \"$cat_id.\"");
    echo json_encode($fetch_details);

}
else if($type =='fetch_item_details')
{
    $item_id = $_REQUEST['edit_item_id'];
    $fetch_details = return_single("SELECT * from items where id = \"$item_id.\"");
    echo json_encode($fetch_details);
```

```

}

else if($type=='check_cat_name'){
    if(!empty($_POST["category"])){
        $query = gosql("SELECT * FROM category WHERE name="" . $_POST["category"]
. """);
        $user_count=mysqli_num_rows($query);
        if($user_count>0) {

            echo "<span id='name_status' class='r cat-name-not-available'> Category Name
Already Exists</span>";
        }else{
            echo "<span id='name_status' class='b cat-name-available'>Category Name
Available</span>";
        }
    }
}

else if($type=='check_item_name'){
    if(!empty($_POST["item"])){
        $query = gosql("SELECT * FROM items WHERE name="" . $_POST["item"] . """);
        $user_count=mysqli_num_rows($query);
        if($user_count>0) {

            echo "<span id='item_name_status' class='item-name-not-available'><span
class='r'> Item Name Already Exists</span></span>";
        }else{
            echo "<span id='item_name_status' class='item-name-available'><span
class='b'>Item Name Available</span></span>";
        }
    }
}

else if($type == 'delete_department'){

$pin_validation=$_REQUEST["pin"];
$confirm_pin = return_single("SELECT pin_number FROM pin");
$confirm_pin_validation=$confirm_pin['pin_number'];
if($confirm_pin_validation==$pin_validation){

    echo 1;
    $delete_tables = gosql("DELETE FROM department WHERE id =
"" .$_REQUEST["delete_department_id"]. """);
}
else{

```

```

        echo 0;
    }

}

else if($type == 'delete_category'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
        $delete_category = gosql("DELETE FROM category WHERE id =
".$REQUEST["delete_category_id"]."");
        $delete_item = gosql("DELETE FROM items WHERE category_id =
".$REQUEST["delete_category_id"]."");
    }
    else{
        echo 0;
    }
}

else if($type == 'delete_item'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
        $delete_category = gosql("DELETE FROM items WHERE id =
".$REQUEST["delete_item_id"]."");
    }
    else{
        echo 0;
    }
}

else if($type == 'delete_order'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
        $delete_tables = gosql("DELETE FROM order_items WHERE order_id =

```

```

"".$_REQUEST["delete_order_id"]." AND item_id=""$_REQUEST["item_id"]." AND
status=""$_REQUEST["status"]."");
}
else{
    echo 0;
}
}

else if($type=='delete_live_order'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
        $delete_tables = gosql("DELETE FROM order_items WHERE id =
""$_REQUEST["order_id"]."");
    }
    else{
        echo 0;
    }
}

else if($type == 'delete_full_order'){

$pin_validation=$_REQUEST["pin"];
$confirm_pin = return_single("SELECT pin_number FROM pin");
$confirm_pin_validation=$confirm_pin['pin_number'];
if($confirm_pin_validation==$pin_validation){
    echo 1;
    $delete_tables = gosql("DELETE FROM ordertable WHERE id =
""$_REQUEST["delete_full_order_id"]."");
    $delete_items = gosql("DELETE FROM order_items WHERE order_id =
""$_REQUEST["delete_full_order_id"]."");
}
else{
    echo 0;
}

}

else if($type == 'add_department'){
    if($_REQUEST["deptid"]==""){

```

```

    $add_department = gosql("INSERT INTO department
(deptname,deptcode,noemp,resource) VALUES
('".$_REQUEST["userdept"]."', '".$_REQUEST["deptcode"]."', '".$_REQUEST["noemp"]."',
'".$_REQUEST["resource"]."')");
}
else{
    $update_department=gosql("UPDATE department SET deptname =
'".$_REQUEST["userdept"]."', deptcode= '".$_REQUEST["deptcode"]."',
noemp='".$_REQUEST["noemp"]."', resource='".$_REQUEST["resource"]."'" WHERE id
= '".$_REQUEST["deptid"]."';");
}
}

// Checking Employee name already exists or not
else if($type=='check_emp_username'){
    if(!empty($_POST["emp_username"])){
        $query = gosql("SELECT * FROM employees WHERE username='".
$_POST["emp_username"] . "'");
        $user_count=mysqli_num_rows($query);
        if($user_count>0) {
            echo "<span id='name_status' class='status-name-not-available'><span class='r'>
Userame Already Exists</span></span>";
        }else{
            echo "<span id='name_status' class='status-name-available'><span
class='b'>Username Available</span></span>";
        }
    }
}

//Checking Department name already exists or not
else if($type=='check_dept_name'){
    if(!empty($_POST["department_name"])){
        $query = gosql("SELECT * FROM department WHERE deptname='".
$_POST["department_name"] . "'");
        $user_count=mysqli_num_rows($query);
        if($user_count>0) {

            echo "<span id='name_status' class='status-name-not-available'><span class='r'>
Department Name Already Exists</span></span>";
        }else{
            echo "<span id='name_status' class='status-name-available'><span
class='b'>Department Name Available</span></span>";
        }
    }
}

```

```

//Checking department code already exists or not
if(!empty($_POST["department_code"])) {
    $query = gosql("SELECT * FROM department WHERE deptcode="" .
$_POST["department_code"] . """);
    $user_count=mysqli_num_rows($query);
    if($user_count>0) {
        echo "<span id='code_status' class='r status-code-not-available'> Department Code
Already Exists</span>";
    }else{
        echo "<span id='code_status' class='b status-code-available'>Department Code
Available</span>";
    }
}
}

else if($type=='select_dept_name'){
    if(!empty($_POST["department_name"])) {
        $query = gosql("SELECT * FROM department WHERE deptname="" .
$_POST["department_name"] . """);
    }
}

//check delete table
else if($type == 'delete_tables'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
        $delete_tables = gosql("DELETE FROM tables WHERE id =
".$_REQUEST["table_id"]."");
    }
    else{
        echo 0;
    }
}

//check table name and short name availability
else if($type=='check_table_name'){
    if(!empty($_POST["table_name"])){

```

```

$query = gosql("SELECT * FROM tables WHERE table_name="" .
$_POST["table_name"] . "");
$user_count=mysqli_num_rows($query);
if($user_count>0) {

    echo "<span id='table_name_status' class='status-table_name-not-available r '>
Table Name Already Exists</span>";
} else{
    echo "<span id='table_name_status' class='status-table_name-available b '>
Available </span>";
}
if(!empty($_POST["short_name"])){
    $query = gosql("SELECT * FROM tables WHERE short_name="" .
$_POST["short_name"] . "");
$user_count=mysqli_num_rows($query);
if($user_count>0) {
    echo "<span id='short_name_status' class='status-short_name-not-available r '>
Short name Already Exists</span>";
} else{
    echo "<span id='short_name_status' class='status-short_name-available b '>Short
name Available</span>";
}
}
}

//add table
else if($type == 'add_table'){
if($_REQUEST["table_id"]==""){
    $add_table = gosql("INSERT INTO tables
(table_name,short_name,table_capacity,is_active,createdon) VALUES
(\"".$_REQUEST["tablename"]."\",\"".$_REQUEST["shortname"]."\",\"".$_REQUEST["tableca
pacity"]."\",\"".$_REQUEST["is_available"]."\",now())");
} else{
    $update_table = gosql("UPDATE tables SET table_name =
\"".$_REQUEST["tablename"]."\",short_name =
\"".$_REQUEST["shortname"]."\",table_capacity =
\"".$_REQUEST["tablecapacity"]."\",is_active = '".$_REQUEST["is_available"]."'" WHERE
id = '".$_REQUEST["table_id"]."');");
}
}

// page: roles.php function: add role /edit role

```

```

else if($type == 'add_role'){
    if($_REQUEST["roleid"]==""){
        $add_role = gosql("INSERT INTO role (role,dept) VALUES
        (".$_REQUEST["userrole"].",".$_REQUEST["department"].");");
    }else{
        $update_role=gosql("UPDATE role SET role = ".$_REQUEST["userrole"].", dept =
        ".$_REQUEST["department"]." WHERE id = ".$_REQUEST["roleid"].";");
    }
}

// page: roles.php function: delete role
else if($type == 'delete_role'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
        $delete_tables = gosql("DELETE FROM role WHERE id =
        ".$_REQUEST["role_id"]."");
    }
    else{
        echo 0;
    }
}

// check roles availability
else if($type=='check_role_name'){
    if(!empty($_POST["role"])&&!empty($_POST["department"])){
        $query = gosql("SELECT * FROM role WHERE role='$_POST[\"role\"]' and
        dept='$_POST[\"department\"]'");
        $user_count=mysqli_num_rows($query);
        if($user_count>0) {
            echo "<span id='name_status' class='role-name-not-available'><span class='r'> Role
Name Already Exists</span></span>";
        }else{
            echo "<span id='name_status' class='role-name-available'><span class='b'>Role
Name Available</span></span>";
        }
    }
}

```

```

// delete payment-method
else if($type == 'delete_method'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
        $delete_method = gosql("DELETE FROM paymethod WHERE id =
        """. $_REQUEST["method_id"]."");
    }
    else{
        echo 0;
    }
}

//fetch data for edit
else if($type =='fetch_payment_details')
{
    $fetch_details = return_single("SELECT * from paymethod where id =
    """. $_REQUEST["methodid"]."");
    echo json_encode($fetch_details);
}

//check method already available
else if($type=='check_method_name'){
    if(!empty($_POST["method_name"])){
        $query = gosql("SELECT * FROM paymethod WHERE method_name=". .
        $_POST["method_name"]);
        $user_count=mysqli_num_rows($query);
        if($user_count>0) {

            echo "<span id='method_name-availability-status'
            class='status-method_name-not-available r '> Method Already Exists</span>";
        }else{
            echo "<span id='method_name-availability-status'
            class='status-method_name-available b '> Available </span>";
        }
    }
}

```

```

//add payment-method
else if($type == "add_method"){
    if($_REQUEST["methodid"]==""){
        $add_method = gosql("INSERT INTO paymethod (id,method_name) VALUES
        (".$_REQUEST["methodid"].",".$_REQUEST["method_name"].")");
    }
    else{
        $update_method = gosql("UPDATE paymethod SET method_name =
        ".$_REQUEST["method_name"]." WHERE id = ".$_REQUEST["methodid"]."");
    }
}

//edit method
// else if($type == "edit_method"){
//     if(!empty($_POST["methodid"])){
//         $update_table = gosql("UPDATE paymethod SET method_name =
//         ".$_REQUEST["method_name"]." WHERE id = ".$_REQUEST["methodid"]."");
//     }
// }

// page: daily_report.php function: report display/ export csv / search
else if($type=="viewdailyreport"){

$cond = "";
if($_POST["is_date_search"] == "yes")
{
    $cond = 'AND createdon BETWEEN "'.$_POST["start_date"].'" AND
    "'.$_POST["end_date"].'";
}

$daily_report = return_array("SELECT date(createdon) as createdonval, count(1) as
ordercount, count(customerid) as
customerid, sum(price) as price, sum(gst_value) as gst_value, sum(total_price) as
total_price FROM ordertable where 1=1
".$cond." GROUP BY date(createdon);");
?>

<form method='post' action='download.php'>
<div class="pl-4">
<input type='submit' class=" btn-md btn btn-primary " value='Export' name='Export'>

```

```

</div>
<br><br>
<div class="card">
  <div class="card-body">

    <h4 class="card-title">Daily Summary</h4>

    <table id="datatable" class="table table-bordered dt-responsive nowrap">
      <thead>
        <tr>
          <th>S.No.</th>
          <th>Order Date</th>
          <th>Total Orders</th>
          <th>Total Customers</th>
          <th>Amount without GST</th>
          <th>Total GST</th>
          <th>Total Amount</th>
        </tr>
      </thead>

      <tbody>
        <?php
$ i = 1;
$user_arr = array();
$user_arr[] = array("S. No.", "Order Date", "Total Orders", "Total Customers", "Amount
without GST", "Total GST", "Total Amount");
foreach ($daily_report as $key => $value) { ?>
        <tr>
          <td><?php echo $i; ?></td>
          <?php $id = $i;?>
          <td><?php echo $value["createdonval"];
$createdonval=$value["createdonval"]; ?></td>
          <td><?php echo $value["ordercount"];
$ordercount=$value["ordercount"]; ?></td>
          <td><?php echo $value["customerid"];
$cusid=$value["customerid"]; ?></td>
          <td><?php echo $value["price"];
$price=$value["price"]; ?></td>
          <td><?php echo $value["gst_value"];
$gst=$value["gst_value"];?></td>

```

```

        <td><?php echo $value["total_price"];
$total=$value["total_price"];?></td>
      </tr>
      <?php $i++;
$user_arr[] = array($id,$createdonval,$ordercount,$cusid,$price,$gst,$total);
}
?>
      </tbody>
    </table>

</div>
</div>
<?php
    echo "
<pre>";
$serialize_user_arr = serialize($user_arr);
?>
<textarea name='export_data' style='display: none;'><?php echo $serialize_user_arr;
?></textarea>
<textarea name='page_name' style='display: none;'>Daily Report</textarea>
</form>
<script src="assets/js/pages/datatables.init.js"></script>
<?php
}
//Delete user

else if($type=='forgot_email'){
    // $email_entered= $_POST["email"];
    if(!empty($_REQUEST["email"])){
        $query = gosql("SELECT * FROM users WHERE email=" . $_REQUEST["email"] . "'");
        $email_count=mysqli_num_rows($query);
        if($email_count>0) {
            $mail = new PHPMailer;

            $mail->isSMTP();                      // Set mailer to use SMTP
            $mail->Host = 'smtp.gmail.com';        // Specify main and backup SMTP servers
            $mail->SMTPAuth = true;                // Enable SMTP authentication
            $mail->Username = 'malayika2610@gmail.com'; // SMTP username
            $mail->Password = 'malu26121999'; // SMTP password
            $mail->SMTPSecure = 'tls';           // Enable TLS encryption, `ssl` also accepted

```

```

$mail->Port = 587; // TCP port to connect to

$mail->setFrom("malayika2610@gmail.com", "malavika");
$mail->addReplyTo("malayika2610@gmail.com", "malavika");
$mail->addAddress("s.logeshkarthik@gmail.com"); // Add a recipient
$mail->isHTML(true); // Set email format to HTML

$mail->Subject = "Link to reset password";
$query1 = return_single("SELECT * FROM users WHERE email='". $_REQUEST["email"] . "'");
// $encrypt= $query1["username"];
$encrypt= str_rot13($query1["username"]);
$mail->Body = "<pre>Click the below link and reset your password  

http://localhost/parent-foox/backend/mail.php?username=".$encrypt."</pre>";

$mail->send();
//header("Location:mail_sent.php?email=".$_REQUEST["email"].");
echo 1;
}else{
    $message="Invalid email Id";
    echo ($message);
}
}

//reset_password
else if($type=='reset_password'){
$hashed_password=md5($_REQUEST["password1"]);
$reset_query= gosql("UPDATE users SET `password`='".$hashed_password."' WHERE
username='". $_REQUEST["username"]."';");
//echo 1;
}
else if($type == 'delete_user'){
$pin_validation=$_REQUEST["pin"];
$confirm_pin = return_single("SELECT pin_number FROM pin");
$confirm_pin_validation=$confirm_pin['pin_number'];
if($confirm_pin_validation==$pin_validation){
    echo 1;
    $delete_users = gosql("DELETE FROM users WHERE id =
".$_REQUEST["user_id"].");
}
else{
    echo 0;
}
}

```

```

    }

}

//Checking Email Id already exists or not
else if($type=='check_user_email'){
    if(!empty($_POST["email_id"])){
        $query = gosql("SELECT * FROM users WHERE email='' . $_POST["email_id"] .
        """);
        $email_count=mysqli_num_rows($query);
        if($email_count>0) {

            echo "<span id='email_status' class='r status-email-not-available'>Email Id Already
Exists</span>";
        }else{
            echo "<span id='email_status' class='b status-email-available'>email
accepted</span>";
        }
    }
    //Checking Phone number already exists or not
    if(!empty($_POST["phone_number"])){
        $query = gosql("SELECT * FROM users WHERE phone='' .
$_POST["phone_number"] . """);
        $phone_count=mysqli_num_rows($query);
        if($phone_count>0) {

            echo "<span id='phone_status' class='r status-phone-not-available'> Phone number
Already Exists</span>";
        }else{
            echo "<span id='phone_status' class='b status-phone-available'>Phone
Available</span>";
        }
    }
    //Add_user
}else if($type == 'add_user'){
    if($_REQUEST["userid"]==""){
        $add_user = gosql("INSERT INTO users
(emp_name,email,phone,dept,role,role_id,username,password) VALUES
('".$_REQUEST["empname"]."','".$_.REQUEST["useremail"]."','".$_.REQUEST["userphon
e"]."."','".$_.REQUEST["userdept"]."','".$_.REQUEST["userrole"]."','".$_.REQUEST["role_id
"]."','".$_.REQUEST["username"]."','".$_.md5($_REQUEST["userpassword"]."")"");
    }else{
        $update_user = gosql("UPDATE users SET

```

```

emp_name="" .$_REQUEST["empname"].",email = "" .$_REQUEST["useremail"].",phone
= "" .$_REQUEST["userphone"].",dept = "" .$_REQUEST["userdept"].",role =
"" .$_REQUEST["userrole"].",role_id = "" .$_REQUEST["role_id"].",username =
"" .$_REQUEST["username"]." WHERE id = "" .$_REQUEST["userid"]."");
}

//page: monthly_report.php function: report display/ export csv / search
else if($type=="viewmonthlyreport"){
    $cond = "";
    if($_POST["is_month_search"] == "yes")
    {
        $cond = 'AND date(createdon) BETWEEN concat("".$_POST["start_month"].",-01)
AND LAST_DAY(concat("".$_POST["end_month"].",-01));
    }

    $monthly_report = return_array("SELECT MONTHNAME(createdon) as
createdonval,year(createdon) as createdonyear,count(1) as ordercount,count(customerid) as
customerid, sum(price)as price, sum(gst_value)as gst_value, sum(total_price)as total_price
FROM ordertable where 1=1 ".$cond." GROUP BY month(ordertable.createdon);");

    ?>
<form method='post' action='download.php'>
    <div class="pl-4">
        <input type='submit' class="btn btn-primary " value='Export' name='Export'>
        <!-- <button type="button" class="btn btn-primary ">Export</button> -->
    </div>
    <br><br>

    <div class="card">
        <div class="card-body">

            <h4 class="card-title">Monthly Report</h4>

            <table id="datatable" class="table table-bordered dt-responsive nowrap"
                style="border-collapse: collapse; border-spacing: 0; width: 100%;">
                <thead>
                    <tr>
                        <th>S. No.</th>
                        <th>Order Month</th>
                        <th>Order Year</th>
                        <th>Total Orders</th>
                        <th>Total Customers</th>

```

```

<th>Amount without GST</th>
<th>Total GST</th>
<th>Total Amount</th>
</tr>
</thead>
<tbody>
<?php
$i = 1;
$user_arr = array();
$user_arr[] = array("S. No.", "Order Month", "Order Year", "Total Orders", "Total
Customers", "Amount without GST", "Total GST", "Total Amount");
foreach ($monthly_report as $key => $value) { ?>
<tr>
<td><?php echo $i; ?></td>
<?php $id = $i;?>
<td><?php echo $value["createdonval"];
$createdonval=$value["createdonval"]; ?></td>
<td><?php echo $value["createdonyear"];
$createdonyear=$value["createdonyear"]; ?></td>
<td><?php echo $value["ordercount"];
$ordercount=$value["ordercount"]; ?></td>
<td><?php echo $value["customerid"];
$cusid=$value["customerid"]; ?></td>
<td><?php echo $value["price"];
$price=$value["price"]; ?></td>
<td><?php echo $value["gst_value"];
$gst=$value["gst_value"];?></td>
<td><?php echo $value["total_price"];
$total=$value["total_price"];?></td>

</tr>
<?php $i++;
$user_arr[] = array($id, $createdonval, $createdonyear, $ordercount, $cusid, $price, $gst, $total);
}
?>

</tbody>
</table>
</div>
</div>
<?php

```

```

echo "
<pre>";
$serialize_user_arr = serialize($user_arr);
?>
<textarea name='export_data' style='display: none;'><?php echo $serialize_user_arr;
?></textarea>
<textarea name='page_name' style='display: none;'>Monthly Report</textarea>
</form>
<script src="assets/js/pages/datatables.init.js"></script>
<?php
}

else if($type=="viewitemsreport"){
    $cond = "";
    if($_POST["is_date_search"] == "yes")
    {
        $cond = 'AND ordertable.createdon BETWEEN "'.$_POST["start_date"].'" AND
"'.$_POST["end_date"].'";
    }
    else{
        $cond = 'AND month(ordertable.createdon)=MONTH(CURDATE()) and
year(ordertable.createdon)=YEAR(CURDATE())';
    }
    $daily_report = return_array("SELECT items.name as item_name,category.name as
category_name,sum(order_items.qty) as item_qty,sum(order_items.price) as
item_price,sum(order_items.gst_value) as gst_value,sum(order_items.total_price) as
total_price,order_items.item_id as item_id, count(*) as cou from order_items inner join
ordertable on order_items.order_id=ordertable.id inner join items on
order_items.item_id=items.id INNER JOIN category on category.id=items.category_id
where 1=1 ".$cond." group by order_items.item_id");
    ?>
<form method='post' action='download.php'>
    <div class="pl-4">
        <input type='submit' class="btn btn-primary" value='Export' name='Export'>
    </div>
    <br><br>
    <div class="card">
        <div class="card-body">
            <h4 class="card-title">Items Report</h4>
            <table id="datatable" class="table table-bordered dt-responsive nowrap" data-bbox="150 870 840 900">

```

```

style="border-collapse: collapse; border-spacing: 0; width: 100%;"
<thead>
  <tr>
    <th>S.No</th>
    <th>Item Name</th>
    <!-- <th>Item ID</th> -->
    <th>Quantity</th>
    <th>Price</th>
    <th>GST</th>
    <th>Total Price</th>

  </tr>
</thead>
<tbody>
  <?php
$id = 1;
$c="";
$c1="";
$no=1;
$user_arr = array();
$user_arr[] = array("S.No","Category Name","Item Name","Quantity","Price");
foreach ($daily_report as $key => $value) {

if($c=="" || $c != $value["category_name"] ){
  $no=1;
  $c=$value["category_name"];
  ?>

  <tr>
    <td style="text-align:center;" colspan="5"><?php echo
$value["category_name"]; ?></td>
      <?php $category_name = $value["category_name"];?>
    </tr>
    <br />
    <?php
}
?>
  <tr>
    <td><?php echo $no;?></td>
    <?php $id= $no;$no++;?>
    <td><?php echo $value["item_name"];
```

```

$item_name=$value["item_name"]; ?></td>

    <td><?php echo $value["item_qty"];
$item_qty=$value["item_qty"]; ?></td>
    <td><?php echo $value["item_price"];
$item_price=$value["item_price"]; ?></td>
    <td><?php echo $value["gst_value"];
$item_price=$value["gst_value"]; ?></td>
    <td><?php echo $value["total_price"];
$item_price=$value["total_price"]; ?></td>
</tr>
<?php

$user_arr[] = array($id,$category_name,$item_name,$item_qty,$item_price);

}

?>
</tbody>
</table>
</div>
</div>
<?php
    echo "
<pre>";
$serialize_user_arr = serialize($user_arr);
?>
<textarea name='export_data' style='display: none;'><?php echo $serialize_user_arr;
?></textarea>
<textarea name='page_name' style='display: none;'>Items Report</textarea>
</form>
<script src="assets/js/pages/datatables.init.js"></script>
<?php
    }
else if($type == 'A'){
    $Pageid = $_REQUEST['Pageid'];
    $RoleID = $_REQUEST['RoleID'];
    $SideType = $_REQUEST['SideType'];
    if ($SideType == 'LSS') {
        $tsql = "INSERT INTO rolespages (role_id,page_id,Status) VALUES ('".
$RoleID . "','" . $Pageid . "','" . '0' . "');";
    } else if ($SideType == 'RSS') {
        $tsql = "DELETE FROM rolespages WHERE page_id=" . $Pageid . " AND

```

```

role_id="" . $RoleID . "";
    }
    $getResults = gosql($tsql);
}
else if($type=='P')
{
    $RoleID = $_REQUEST['RoleID'];
    $SideType = $_REQUEST['SideType'];
    if ($SideType == 'LSS') {
        $tsql = "SELECT id as PageID, page_name FROM pages WHERE id NOT
IN(SELECT page_id from rolespages WHERE role_id="" . $RoleID . "") AND status='0'";
    } else {
        $tsql = "SELECT page_id as PageID, (SELECT page_name FROM pages
WHERE Id=page_id) as page_name FROM rolespages WHERE status='0' AND
role_id="" . $RoleID . "";
    }
    $getResults = return_array($tsql);
    foreach($getResults as $pf){
        echo '<option value='.$pf["PageID"].'>'.$pf['page_name'].'</option>';
    }
}
else if($type=="edit_qty"){
    $update_table = gosql("UPDATE order_items SET qty = ".$_REQUEST["qty"].""
WHERE id = ".$_REQUEST["itemid"]."");
    $id=$_REQUEST["itemid"];
    $sel_query=return_single("SELECT * FROM order_items where id="" . $id . """);
    $itemid=$sel_query['item_id'];
    $quantity_no=$sel_query['qty'];
    $single_price=return_single("SELECT * FROM items where id="" . $itemid . """);
    $total_price=($single_price['price'])*($quantity_no);
    $gst_value=($single_price['gst_value'])*($quantity_no);
    $total_sum_price=($single_price['total_price'])*($quantity_no);
    $update_price = gosql("UPDATE order_items SET price = ".$total_price.", gst_value =
".$gst_value.", total_price = ".$total_sum_price." WHERE id = ".$id.");
    $result = return_single("SELECT SUM(price) AS value_sum, SUM(gst_value) AS
table_gst_value, SUM(total_price) AS table_total_value FROM order_items where
order_id="" . $sel_query['order_id']. "");
    $update_order_price=gosql("UPDATE ordertable SET
price=".$_result['value_sum'].",gst_value=".$_result['table_gst_value'].",
total_price=".$_result['table_total_value']."' WHERE id="" . $sel_query['order_id']. ")");
    $cal_order_total_price=return_single("SELECT SUM(price + gst_value) as total from
ordertable where id="" . $sel_query['order_id']. """);
}

```

```

echo($cal_order_total_price);
$update_order_total_price=gosql("UPDATE ordertable SET
total_price='".$cal_order_total_price['total']."' where id='".$sel_query['order_id']."'');");
}

else if($type =='fetch_venture_details'){
$venture_id = $_REQUEST['venture_id'];
$fetch_details = return_single("SELECT * from venture where id =
".$REQUEST["venture_id"]."");
echo json_encode($fetch_details);

}

else if($type == 'delete_venture'){
$pin_validation=$_REQUEST["pin"];
$confirm_pin = return_single("SELECT pin_number FROM pin");
$confirm_pin_validation=$confirm_pin['pin_number'];
if($confirm_pin_validation==$pin_validation){
echo 1;
$delete_category = gosql("DELETE FROM venture WHERE id =
".$REQUEST["delete_venture_id"]."");
}
else{
echo 0;
}

}

else if($type == 'new_customer'){

if($_REQUEST["id"]==""){
$hashed_password = md5($_REQUEST["password"]);
$add_details = gosql("INSERT INTO customer (name,email,mobile) VALUES
('".$_REQUEST["name"]."', '".$_REQUEST["email"]."', '".$_REQUEST["number"]."');");
}
else{
$update_item = gosql("UPDATE customer SET name =
".$_REQUEST["name"].", email =
".$_REQUEST["email"].", mobile = '".$_REQUEST["number"]."'" WHERE
`customer`.`id` = '".$_REQUEST["id"]."');
echo 0;
}

}

```

```

else if($type == 'delete_customer'){

    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){

        $delete_tables = gosql("DELETE FROM customer WHERE id =
"".$_REQUEST["delete_customer_id"]."");
        echo 1;
    }
    else{
        echo 0;
    }
}

else if($type =='fetch_customer_details')
{
$customer_id = $_REQUEST['customer_id'];
// echo "SELECT * from customer where id = ".$_REQUEST["customer_id"]."";
$fetch_details = return_single("SELECT * from customer where id =
".$_REQUEST["customer_id"]."");
echo json_encode($fetch_details);
}

else if($type == 'new_venture'){
    if($_REQUEST["id"]==""){
        $add_det = gosql("INSERT INTO venture (venture_name, venture_address,
brand_name, brand_address, owner_name,owner_mail,
manager_name,manager_mail,phone_number,bank_details,product_list_button,product_pe
rcentage)

VALUES".$_REQUEST["venture_name"].","."$_REQUEST["venture_address"].","."$_R
EQUEST["brand_name"].","."$_REQUEST["brand_address"].","."$_REQUEST["owner_n
ame"].","."$_REQUEST["owner_mail"].","."$_REQUEST["manager_name"].","."$_REQ
UEST["manager_mail"].","."$_REQUEST["phone_number"].","."$_REQUEST["bank_det
ails"].","."$_REQUEST["product_list_button"].","."$_REQUEST["product_percentage"]."
);");

        echo 1;
    }
    else{
        $update_item = gosql("UPDATE venture SET venture_name =
".$_REQUEST["venture_name"].",venture_address =

```

```

    """.REQUEST["venture_address"].",brand_name =
    """.REQUEST["brand_name"].",brand_address =
    """.REQUEST["brand_address"].",owner_name =
    """.REQUEST["owner_name"].",owner_mail =
    """.REQUEST["owner_mail"].",manager_name =
    """.REQUEST["manager_name"].",manager_mail =
    """.REQUEST["manager_mail"].",phone_number =
    """.REQUEST["phone_number"].", bank_details =
    """.REQUEST["bank_details"].",product_list_button =
    """.REQUEST["product_list_button"].", product_percentage =
    """.REQUEST["product_percentage"]." WHERE `venture`.`id` =
    """.REQUEST["id"]."");
    echo 0;
}

else if($type == 'delete_tables'){
    $delete_tables = gosql("DELETE FROM tables WHERE id =
    """.REQUEST["table_id"]."");
}
else if($type=='check_pin')
{
    $pin_validation=$REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
    }
    else{
        echo 0;
    }
}
else if($type=='update_pin')
{
    $update_pin = gosql("UPDATE pin SET
    pin_number=".REQUEST["confirm_pwd"].";");
}
else if($type == 'delete_emp'){
    $pin_validation=$REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
}

```

```

if($confirm_pin_validation==$pin_validation){
    echo 1;
    $delete_emp = gosql("DELETE FROM employees WHERE id =
".$_REQUEST["emp_id"]."");
}
else{
    echo 0;
}
}

// page: addon.php function: delete addon
else if($type == 'delete_addons'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
        $delete_tables = gosql("DELETE FROM addons WHERE id =
".$_REQUEST["role_id"]."");
    }
    else{
        echo 0;
    }
}

// page: addon.php function: add addon /edit addon
else if($type == 'add-addons'){
    if($_REQUEST["addonid"]==""){
        $addAddon = gosql("INSERT INTO addons(addonname,addonprice) VALUES
(\"".$_REQUEST["addonname"]."\",\"".$_REQUEST["addonprice"]."');");
    }else{
        $updateAddon = gosql("UPDATE addons SET addonname =
".$_REQUEST["addonname"]." , addonprice = '".$_REQUEST["addonprice"]."'" WHERE
id = '".$_REQUEST["addonid"]."');");
    }
}

// check addon availability
// else if($type=='check_addonname'){
//     if(!empty($_POST["addonname"])&&!empty($_POST["addonprice"])){
//         $query = gosql("SELECT * FROM addons WHERE addonname='".
$_POST["addonname"]." and addonprice='". $_POST["addonprice"]."'");
//         $user_count=mysqli_num_rows($query);
//         if($user_count>0) {

```

```

//      echo "<span id='name_status' class='role-name-not-available'><span class='r'>
Role Name Already Exists</span></span>";
//    }else{
//      echo "<span id='name_status' class='role-name-available'><span class='b'>Role
Name Available</span></span>";
//    }
//  }

// }

else if($type == 'check_addonname'){
  if(!empty($_POST["addonname"])){
    $query = gosql("SELECT * FROM addons WHERE
addonname='". $_POST["addonname"]. "'");
    $user_count=mysqli_num_rows($query);
    // echo ($name);
    if($user_count>0) {
      echo "<span id='name_status' class='role-name-not-available'><span class='r'> Role
Name Already Exists</span></span>";
    }else{
      echo "<span id='name_status' class='role-name-available'><span class='b'>Role
Name Available</span></span>";
    }
  }
}

else if($type == 'insert_page_rolling'){
  if($_REQUEST["page_id"]!="") && ($_REQUEST["role_id"]!="") &&
($_REQUEST["status"]!="")){
    $add_query=gosql("INSERT IGNORE INTO `user_role_mapping`(`Role_id`,
`Page_id`, `Status`) VALUES ('".$_REQUEST["role_id"]."','".$_.REQUEST["page_id"]."',
".$_REQUEST["status"]."');");
  }
}

else if($type=='delete_page_rolling'){
  if($_REQUEST["role_id"]!=""){
    $del_page_query=gosql("DELETE FROM `user_role_mapping` WHERE
`Role_id`='".$_REQUEST["role_id"]."';");
  }
}

else if($type == 'add_employee'){

```

```

if($_REQUEST["table_id"]==""){
    $add_employee = gosql("INSERT INTO `employees`(`code`, `username`, `password`,
    `name`, `faname`, `dob`, `aadhar`, `pan`, `doj`, `mail`, `phone`, `role`, `depart`, `accno`,
    `ifsc`) VALUES
    (".$_REQUEST["empId"].","","$_REQUEST["emp_username"].","",md5($_REQUEST["p
    assword"]).","","$_REQUEST["empname"].","",$_REQUEST["fathername"].","",$_REQUEST[
    "dob"].","","$_REQUEST["aadhar"].","",$_REQUEST["pannum"].","",$_REQUEST[
    "doj"].","",$_REQUEST["email"].","",$_REQUEST["phone"].","",$_REQUEST["departme
    nt"].","",$_REQUEST["role"].","",$_REQUEST["acountnum"].","",$_REQUEST["ifsc"].""
    );");
}else{
    $update_employee=gosql("UPDATE `employees` SET
    `username`=".$_REQUEST["emp_username"].","`name`=".$_REQUEST["empname"].",
    `faname`=".$_REQUEST["fathername"].","`dob`=".$_REQUEST["dob"].","`aadhar`=".$__
    REQUEST["aadhar"].","`pan`=".$_REQUEST["pannum"].","`doj`=".$_REQUEST["doj"].",
    `mail`=".$_REQUEST["email"].","`phone`=".$_REQUEST["phone"].","`role`=".$_REQ
    UEST["role"].","`depart`=".$_REQUEST["department"].","`accno`=".$_REQUEST["acou
    ntnum"].","`ifsc`=".$_REQUEST["ifsc"]." WHERE id = ".$_REQUEST["table_id"].";");
}

}
else if($type == 'delete_emp'){
    $delete_emp = gosql("DELETE FROM employees WHERE id =
    ".$_REQUEST["emp_id"]."");
}

else if($type == 'get_emp_role'){
    $get_role=return_array("SELECT * FROM role WHERE
    dept=".$_REQUEST["get_department_name"]."");

    if(!empty($get_role)){
        foreach($get_role as $row) {
            echo '<option value="'.$row["role"].'" id="'.$row["id"].'">'.$row["role"].'</option>';
        }
    }else{
        echo '<option disabled>No roles assigned to this department</option>';
    }
}

else if($type == 'get_top5_by_selection'){
    // $_REQUEST["deptid"]==""
    if($_REQUEST["get_top5_selection_id"]=="category5"){
        $get_top5_category=return_array("SELECT order_items.category_id,category.name
        as category_Name, SUM(order_items.qty), SUM(order_items.total_price) FROM

```

```

order_items,category WHERE order_items.item_id=category.id GROUP BY
order_items.category_id ORDER BY `SUM(order_items.qty)` DESC LIMIT 5");
if(!empty($get_top5_category)){
    foreach($get_top5_category as $row) {
        // echo '<option value="'. $row["role"].'"'
        id="'. $row["id"].'"'.'$row["role"].'</option>';
        echo '<tr class="p-0">
            <td class="float-left">' . $row["category_Name"] . '</td>
            <td>' . $row["quantity"] . '</td>
            <td>' . $row["total_price"] . '</td>
        </tr>';
        // echo '<tr class="p-0">
        //     <td class="float-left">1</td>
        //     <td>2</td>
        //     <td>3</td>
        // </tr>';
    }
} else{
    echo '<option disabled>No roles assigned to this department</option>';
}
}

} else{
    $get_top5_items=return_array("SELECT order_items.item_id,items.name as
item_Name, SUM(order_items.qty), SUM(order_items.total_price) FROM
order_items,items WHERE order_items.item_id=items.id GROUP BY order_items.item_id
ORDER BY `SUM(order_items.qty)` DESC LIMIT 5 ;");
if(!empty($get_top5_items)){
    foreach($get_top5_items as $row) {
        // echo '<option value="'. $row["role"].'"'
        id="'. $row["id"].'"'.'$row["role"].'</option>';
        echo '<tr class="p-0">
            <td class="float-left">' . $row["item_Name"] . '</td>
            <td>' . $row["order_items.qty"] . '</td>
            <td>' . $row["total_price"] . '</td>
        </tr>';
        // echo '<tr class="p-0">
        //     <td class="float-left">1</td>
        //     <td>2</td>
        //     <td>3</td>
    }
}
}

```

```

// </tr>';

}

}else{
    echo '<option disabled>No roles assigned to this department</option>';
}

}

}

else if($type == 'add_dept'){
    if($_REQUEST["deptid"]==""){
        $add_dept = gosql("INSERT INTO department (deptname,deptcode,noemp,resource)
VALUES
('".$_REQUEST["userdept"]."','".$_REQUEST["deptcode"]."','".$_REQUEST["noemp"]."",
'".$_REQUEST["resource"]."')");
    }
    else{
        $update_dept=gosql("UPDATE department SET deptname =
'".$_REQUEST["userdept"]."', deptcode= '".$_REQUEST["deptcode"]."',
noemp='".$_REQUEST["noemp"]."', resource='".$_REQUEST["resource"]."'" WHERE id
= '".$_REQUEST["deptid"]."');
    }
}
else if($type == 'add_item'){
    $get_gst_value = return_single("SELECT percent FROM gst_master WHERE id =
'".$_REQUEST["gst"]."')");
    $gst_value = ($_REQUEST["price"] * $get_gst_value["percent"])/100;
    $totalprice = $gst_value + $_REQUEST["price"];
    $total_price_parcel = $totalprice + $_REQUEST["packing"];
    if($_REQUEST["item_id"] == ""){
        $add_item = gosql("INSERT INTO items
(name,description,restaurant_id,category_id,is_available,foodtype,price,parcel_charges,gst,
gst_value,total_price,total_price_parcel,url,item_image,createdon,createdby) VALUES
('".$_REQUEST["itemname"]."','".$_REQUEST["itemdesc"]."','".$_REQUEST["RESTAURANT_ID"]."',
'".$_REQUEST["category"]."','".$_REQUEST["is_available"]."','".$_REQUEST["foodtype"]."',
'".$_REQUEST["price"]."','".$_REQUEST["packing"]."','".$_REQUEST["gst"]."','".$_REQUEST["gst
_value"]."','".$_REQUEST["totalprice"]."','".$_REQUEST["total_price_parcel"]."','".$_REQUEST["url"]."',
'".$_REQUEST["item_image"]."'.now(),'1')");
        echo 1;
    }
}

```

```

$update_item = gosql("UPDATE items SET name =
".$REQUEST["itemname"].",description = ".$REQUEST["itemdesc"].",restaurant_id =
".$REQUEST["category_id"].",category_id = ".$REQUEST["category"].",is_available =
".$REQUEST["is_available"].",foodtype = ".$REQUEST["foodtype"].",price =
".$REQUEST["price"].",parcel_charges = ".$REQUEST["packing"].",gst =
".$REQUEST["gst"].",gst_value = ".$gst_value.",total_price =
".$totalprice.",total_price_parcel = ".$total_price_parcel.",url =
".$REQUEST["url"].",item_image = ".$REQUEST["item_image"]." WHERE id =
".$REQUEST["item_id"].");;
echo 0;
}
}

else if($type == 'open_item_category'){
    $get_cat_det = return_single("SELECT * FROM category WHERE id =
".$REQUEST["cat_id"]." AND restaurant_id = ".RESTAURANT_ID."");
    $get_items = return_array("SELECT * FROM items WHERE category_id =
".$REQUEST["cat_id"]." AND restaurant_id = ".RESTAURANT_ID."");
    $foodtype_arr = array("1"=>"Veg","2"=>"Non-Veg","3"=>"Egg")
    ?>
<div id="wrap_item_list">
    <div class="item-list">
        <table id="datatable" class="table table-bordered dt-responsive nowrap"
            style="border-collapse: collapse; border-spacing: 0; width: 100%;">
            <?php if(count($get_items) > 0){ ?>
                <thead>
                    <tr>
                        <th> Items </th>
                        <th> Type </th>
                        <th> Price</th>
                        <th> Availability </th>
                        <th> Delete </th>
                        <th> Actions</th>
                    </tr>
                </thead>
                <?php } ?>
                <tbody>
                    <?php
                    if(count($get_items) > 0){
                        foreach ($get_items as $key => $value) {

```

```

$checked = "";
if($value["is_available"] == '1'){
    $checked = "checked";
}

?>

<tr scope="row" class="item-row" data-id=<?php echo $key;?>>

    <td><?php echo $value["name"] ?></td>
    <td><?php if(($value["foodtype"])==1)
{ ?>
        
        </span><?php echo $foodtype_arr[$value["foodtype"]]; ?>
        <?php }
else{ ?>
        
        </span><?php echo $foodtype_arr[$value["foodtype"]]; ?>
        <?php } ?>
    </td>
    <td><?php echo $value["total_price"]; ?></td>

    <td><span class="badge badge-pill list-group-flush"><input type="checkbox"
class="toggle_item_available" id=<?php echo $value["id"] ?>" switch="primary"
name="toggle" <?php echo $checked; ?>>
        <label for=<?php echo $value["id"] ?>" data-on-label="Yes"
data-off-label="No"></label></span>
    </td>
    <td><a href="#" data-toggle="modal" class="delete_item"
data-target="#delete_category"
data-id=<?php echo $key;?>" id=<?php echo $value["id"] ?>">Delete</a></td>
    <td><a href="#" id=<?php echo $value["id"] ?>" class="edit_item">Edit
</a></td>

    <?php }
}else{ ?>
        <li class="list-group-item d-flex justify-content-between
align-items-center">NO ITEMS FOUND!</li>
        <?php } ?>

```

```

        </tbody>
    </table>
</div>
</div>

<script src="assets/js/pages/datatables.init.js"></script>
<?php
}

else if($type == 'toggle_available'){
    $toggle_available = gosql("UPDATE ".$_REQUEST["table"]." SET is_available =
".$_REQUEST["value"]." WHERE id= '".$_REQUEST["id"]."');");
}

else if($type == 'login_pin'){
    $pin_validation=$_REQUEST["pin"];
    $confirm_pin = return_single("SELECT pin_number FROM pin");
    $confirm_pin_validation=$confirm_pin['pin_number'];
    if($confirm_pin_validation==$pin_validation){
        echo 1;
    }else{
        echo 0;
    }
}

else if($type == "login"){
    $username = $_REQUEST['email'];
    $password = $_REQUEST['password'];
    $check_cnt = return_single("SELECT COUNT(1) as cid FROM users WHERE email =
".$username."");
    if($check_cnt['cid'] > 0){
        $sfqry = "SELECT * FROM users WHERE email='".$username."' and
password='".md5($password)."'";
        $row1 = return_single($sfqry);
        $res = array();
        $res["status"]="failure";
        if($row1)
        {
            $_SESSION["users"]["Id"] = $row1["id"];
            $_SESSION["users"]["emp_name"] = $row1["emp_name"];
            $_SESSION["users"]["email"] = $row1["email"];
            $_SESSION["users"]["username"] = $row1["username"];
        }
    }
}

```

```

$_SESSION["users"]["role_id"] = $row1["role_id"];
// echo 1;

// echo "<span id='email_forgot1 class='r email-name-available">Email
Exists</span>";
// $count=mysql_num_rows($row1);

// if($count==1){
$_SESSION["logged_in"] = true;
$qry="SELECT page_id from user_role_mapping where
Role_id="" . $row1["role_id"] . "";
$getResults = return_array($qry);
$accesablePages = array();
foreach($getResults as $row){
    $accesablePages[] = $row["page_id"];
}
$_SESSION["users"]["pages"] = implode(",",$accesablePages);
$pageIds = array(
    1 => "dashboard_ui.php",
    2 => "users.php",
    3 => "departments.php",
    4 => "tables.php",
    5 => "role_mapping.php",
    6 => "orders.php",
    7 => "daily_report.php",
    8 => "monthly_report.php",
    9 => "items_report.php",
    10 => "table_report.php",
    11 => "category_report.php",
    12 => "employees.php",
    13 => "profiles.php",
    14 => "settings.php",
    15 => "roles.php",
    16 => "category_menu.php",
    17 => "index.php",
);
}

$res["status"] = "success";
if(empty($getResults)){
    $res["page"]="index.php";
}else{

```

```

if(in_array(1,$accesablePages)){
    $res["page"] ="dashboard_ui.php";
    // header("location:dashboard_ui.php");
}
else{
    $res["page"] = $pageIds[$accesablePages[0]];
}
}

else{
    $res["message"] = "Invalid Username or Password!";
    //echo "<span id='email_forgot1'>Email Id Already Exists</span>";
}

}

else{
    $res["message"] = "404";
    //echo "<span id='email_forgot1'>Email Id Already Exists</span>";
}
echo json_encode($res);
}

else if($type == "change_status_order"){
    $status = $_REQUEST["status"];
    $order_id = $_REQUEST["order_id"];
    $customer_id = $_REQUEST["customer_id"];
    $order_item_id = $_REQUEST["order_item_id"];
    if($status <= 2){
        $next_status = $status + 1;
    }
    $change_status_order = gosql("UPDATE order_items SET status = '".$next_status."' WHERE id = ".$order_item_id." AND order_id = ".$order_id." AND customer_id = ".$customer_id."");
    $check_cnt_completed = return_single("SELECT count(1) AS cid FROM order_items WHERE status = '3' AND order_id = ".$order_id." AND customer_id = ".$customer_id."");
    $get_total_item_order = return_single("SELECT count(1) AS cid FROM order_items WHERE order_id = ".$order_id." AND customer_id = ".$customer_id."");
}

```

```

if($check_cnt_completed["cid"] == $get_total_item_order["cid"]){
    $update_status_order = gosql("UPDATE ordertable SET completed = '1' WHERE id =
".$order_id." AND customerid = ".$customer_id."");
}
}

else if($type=="insert_tax_details1"){

    $tax_details_insert=gosql("INSERT INTO `tax_details`(`tax_name`,
`tax_percentage`) VALUES ('".$_REQUEST["tax_name"]."',
".$_REQUEST["tax_percentage"]."');");
}

else if($type=="insert_tax_details"){

    $tax_details=$_REQUEST["tax_details"];
    foreach($tax_details as $row){

        $tax_details_insert=gosql("INSERT INTO `tax_details`(`tax_name`,
`tax_percentage`) VALUES ('".$row["tax_name"]."', '".$row["tax_percentage"]."');");
    }
    echo 1;
}

else if($type=="show_live_orders"){

    $status_common = array("0"=>"Pending","1"=>"In Progress","2"=>"Ready to
deliver","3"=>"Delivered","4"=>"Completed");?>
<table id="order_datatable" class="table mb-0 table-bordered dt-responsive nowrap">
    <thead>
        <tr>
            <th>S. No.</th>
            <th>Table ID</th>
            <th>Order ID</th>
            <th>Category Name</th>
            <th>Item Name</th>
            <th>Time</th>
            <th>Qty</th>
            <th>Price</th>
            <th>GST Value</th>
            <th>Total Value</th>
            <th>Status</th>
            <th>Action</th>
        </tr>
    </thead>
    <tbody>
        <?php

```

```

$count=1;
$sel_query="SELECT order_items.id as id,order_items.order_id as
order_id,order_items.table_id as table_id,order_items.customer_id as customer_id,
order_items.item_id as item_id,items.name as item_name,category.name as cat_name,
order_items.category_id as category_id, order_items.createdon as timecreatedon,
order_items.qty as qty, order_items.price as price, order_items.gst_value as gst_value,
order_items.total_price as total_price, order_items.status as status FROM
`order_items`, `category`, `items` WHERE order_items.category_id=category.id AND
order_items.item_id=items.id and date(order_items.createdon)=date(now()) ORDER BY
date(order_items.createdon) ASC";
$result = return_array($sel_query);
foreach($result as $row) {
    //extra
    if($row['status']==0){
        $status_color='#ffc107';
    }else if($row['status']==1){
        $status_color='#F7344C';
    }else if($row['status']==2){
        $status_color='skyblue';
    }
    else if($row['status']==3){
        $status_color='darkyellow';
    }
//end extra
    ?>
<tr>
    <td align="center"><?php echo $count; ?></td>
    <td align="center"><?php echo $row["table_id"]; ?></td>
    <td align="center"><?php echo $row["order_id"]; ?></td>
    <td align="center"><?php echo $row["cat_name"]; ?></td>
    <td align="center"><?php echo $row["item_name"]; ?></td>
    <td align="center"><?php
date_default_timezone_set("Asia/Kolkata");
$time_ago      = strtotime($row["timecreatedon"]);
$current_time  = time();
$time_difference = $current_time - $time_ago;
$seconds       = $time_difference;

$minutes = round($seconds / 60); // value 60 is seconds
$hours   = round($seconds / 3600); //value 3600 is 60 minutes * 60 sec
$days    = round($seconds / 86400); //86400 = 24 * 60 * 60;
$weeks   = round($seconds / 604800); // 7*24*60*60;

```

```

$months = round($seconds / 2629440);
//((365+365+365+365+366)/5/12)*24*60*60
$years = round($seconds / 31553280); //((365+365+365+365+366)/5 * 24 * 60 *
60

if ($seconds <= 60){

echo "Just Now";

} else if ($minutes <= 60){
if ($minutes == 1){
echo "one minute ago";
} else {

echo "$minutes minutes ago";

}

} else if ($hours <= 24){

if ($hours == 1){

echo "an hour ago";

} else {

echo "$hours hrs ago";

}

} else if ($days <= 7){

if ($days == 1){

echo "yesterday";

} else {

echo "$days days ago";

}

}

```

```

} else if ($weeks <= 4.3){

if ($weeks == 1){

echo "a week ago";

} else {

echo "$weeks weeks ago";

}

} else if ($months <= 12){

if ($months == 1){

echo "a month ago";

} else {

echo "$months months ago";

}

} else {

if ($years == 1){

echo "one year ago";

} else {

echo "$years years ago";

}

}

?></td>
<!-- <td align="center"><?php echo $row["qty"]; ?></td> -->

<?php
if($row["status"]==0)
{

```

```

?>

<td style="text-align:center;"><a class="btn btn-light extraqty" href="javascript:;" id="<?php echo $row["id"]?>" data-toggle="modal" data-target="#view_order_qty"><?php echo $row["qty"]; ?></a></td>
<?php
}
else{
?>
<td style="text-align:center;"><a class="btn btn-light extraqty disabled" href="javascript:;" id="<?php echo $row["id"]?>" data-toggle="modal" data-target="#view_order_qty"><?php echo $row["qty"]; ?></a></td>

<?php
}
?>

<td align="center"><?php echo $row["price"]; ?></td>
<td align="center"><?php echo $row["gst_value"]; ?></td>
<td align="center"><?php echo $row["total_price"]; ?></td>
<td id="extraColumn"><a style="background:<?php echo $status_color;?>;" class="success btn text-white change_live_status" href="javascript:;" custid="<?php echo $row["customer_id"]; ?>" orderid="<?php echo $row["order_id"]; ?>" id="<?php echo $row["id"]; ?>" status="<?php echo $row["status"]; ?>"><?php echo $status_common[$row["status"]]; ?></a></td>

<?php
if($row["status"]==0)
{
?>

<td>
<a type="button" href="javascript:;" id="<?php echo $row["id"]; ?>" class="delete_live_order btn" data-toggle="modal" data-target="#delete_live_order">

</a>

```

```

</td>

<?php }else{ ?>

<td></td>

    <?php } ?>
</tr>
<?php $count++; } ?>
</tbody>
</table>
<script src="assets/js/pages/datatables.init.js"></script>
<?php
}

else if($type == "show_orders"){
    $status_common = array("0"=>"Pending","1"=>"In Progress","2"=>"Ready to
deliver","3"=>"Delivered","4"=>"Completed");
    $get_orders = return_array("SELECT * FROM ordertable AS ot WHERE restaurant_id
= '".RESTAURANT_ID."' AND confirmed = '1' AND (completed = '0' OR generate_bill =
'0') HAVING (SELECT COUNT(1) FROM order_items WHERE order_id = ot.id) > 0
ORDER BY createdon DESC");
    foreach ($get_orders as $key => $value) {
        $get_order_items = return_array("SELECT * FROM order_items WHERE order_id =
\".$value["id"]."");
        ?>
<table id="order_datatable" class="table mb-0 table-bordered dt-responsive nowrap">
    <thead>
        <tr>
            <td colspan="5">Order ID - <?php echo $value["id"]; ?>
                <?php if($value["notes"]!="){ ?>
                    <a href="javascript:;" id="<?php echo $value["id"]; ?>">
                        class="btn btn-md btn-primary ml-4 pl-3 viewnotes" data-toggle="modal"
                        data-target="#view_notes">View Notes</a>
                    <a type="button" href="javascript:;" id="<?php echo $value["id"]; ?>">
                        class="delete_full_order float-right btn btn-danger" data-toggle="modal"
                        data-target="#delete_full_order">
                            Delete
                        </a>
                    <?php }?>
            </td>
        </tr>

```

```

<tr>
    <th>S. No.</th>
    <th>Item Name</th>
    <th>Qty</th>
    <th>Status</th>
    <th>Action</th>
</tr>
</thead>
<tbody>
    <?php
        $i = 1;
        foreach ($get_order_items as $key1 => $value1) {
            //extra
            if($value1['status']==0){
                $status_color='#ffc107';
            }else if($value1['status']==1){
                $status_color='#F7344C';
            }else if($value1['status']==2){
                $status_color='skyblue';
            }
            else if($value1['status']==3){
                $status_color='darkyellow';
            }
        //end extra
        $get_item_det = return_single("SELECT * FROM items WHERE id =
        ".$value1["item_id"]."");
    ?>
    <tr>
        <td style="text-align:center;"><?php echo $i; ?></td>
        <td><?php echo $get_item_det["name"]; ?></td>

        <?php
            if($value1["status"]==0)
            {
        ?>

        <td style="text-align:center;"><a class="btn btn-light extraqty" href="javascript:;"
            id=<?php echo $value1["id"]?>" data-toggle="modal"
            data-target="#view_order_qty"><?php echo $value1["qty"]; ?></a></td>
    <?php
        }

```

```

else{
?
<td style="text-align:center;"><a class="btn btn-light extraqty disabled"
href="javascript;" id="<?php echo $value1["id"]?>" data-toggle="modal"
data-target="#view_order_qty"><?php echo $value1["qty"]; ?></a></td>

<?php
}
?

<td id="extraColumn"><a style="background:<?php echo $status_color;?>;"
class="success btn text-white change_status" href="javascript;""
custid="<?php echo $value1["customer_id"]; ?>" orderid="<?php echo
$value["id"]; ?>"
id="<?php echo $value1["id"]; ?>""
status="<?php echo $value1["status"]; ?>"><?php echo
$status_common[$value1["status"]]; ?></a></td>

<?php
if($value1["status"]==0)
{
?

<td id="extraRow" style="font-size:20px;text-align:center;text-decoration:none">
<a type="button" class="delete_order" href="javascript;" id="<?php echo
$value1["item_id"]; ?>""
itemid="<?php echo $value1["id"]; ?>" status="<?php echo $value1["status"];
?>" data-toggle="modal"
data-target="#delete_order">

</a>
</td>
<?php
}
else{
?
<td></td>

<?php
}
?

```

```

</tr>
<?php
    $i++;
} ?>
</tbody>
</table>
<script src="assets/js/pages/datatables.init.js"></script>
<?php }
}
//.....list table.....
```

else if(\$type == "show\_table") {  
 \$status\_common = array("0"=>"Pending","1"=>"In Progress","2"=>"Ready to deliver","3"=>"Delivered","4"=>"Completed");  
 \$get\_orders = return\_array("SELECT \* FROM ordertable AS ot WHERE restaurant\_id = ".\$RESTAURANT\_ID." AND confirmed = '1' AND (completed = '0' OR generate\_bill = '0') HAVING (SELECT COUNT(1) FROM order\_items WHERE order\_id = ot.id) > 0 ORDER BY createdon DESC");  
 foreach (\$get\_orders as \$key => \$value) {  
 \$get\_order\_items = return\_array("SELECT \* FROM order\_items WHERE order\_id = ".\$value["id"]."");  
 }?  
<div class="w-100">  
<table border="1" style="border-collapse: collapse; border-spacing: 0; width: 100%;">  
<thead>  
<tr>  
<td colspan="4">Order ID - #<?php echo \$value["id"]; ?></td>  
</tr>  
<tr>  
<th>S. No.</th>  
<th>Table Name</th>  
<th>capacity</th>  
<th>action</th>  
</tr>  
</thead>  
<tbody>  
<?php  
\$i = 1;  
foreach (\$get\_order\_items as \$key1 => \$value1) {  
 \$get\_item\_det = return\_single("SELECT \* FROM items WHERE id = ".\$value1["item\_id"]."");  
?>

```

<tr>
<td><?php echo $i; ?></td>
<td><?php echo $get_item_det["name"]; ?></td>
<td><?php echo $value1["qty"]; ?></td>
<td><a href="javascript:;" custid=<?php echo $value1["customer_id"]; ?>" 
    orderid=<?php echo $value1["id"]; ?>" id=<?php echo $value1["id"]; ?>" 
    status=<?php echo $value1["status"]; ?>" 
    class="change_status"><?php echo $status_common[$value1["status"]]; 
?></a></td>
</tr>
<?php
$i++;
} ?>
</tbody>
</table>
</div>
<?php }
}

//.....end of list table.....
else if($type == 'tablestatus'){

$update_notification_table = gosql("UPDATE `notification` SET
`status`=1,`updated_date`=NOW() WHERE id=".$_REQUEST["id"]."");
}

else if($type == "live_table_status"){

$all_tables = return_array("SELECT * FROM tables WHERE status = 0");

foreach ($all_tables as $key => $value) {
    $live_table = return_single("SELECT * FROM ordertable WHERE table_id =
".$_value["id"]." AND confirmed = 1 AND completed = 0");
    if($live_table["id"] != ""){
        $add_class = "active view_add_order";
    }else{
        $add_class = "";
    }
?>

<a href="javascript:;" id=<?php echo $live_table["id"]; ?>" class=<?php echo
$add_class; ?>"><span
    class="table_span rounded-circle border-0 <?php echo $add_class; ?>"><?php echo

```

```

$value["short_name"]; ?></span></a>
<?php }
}

else if($type=="viewtimelineorder"){
    $dt1=$_REQUEST["dt1"];
    $dt2=$_REQUEST["dt2"];
    $ono=$_REQUEST["ono"];
?
>
<form method='post' action='download.php'>
    <input type='submit' value='Export' name='Export'>
    <table width="100%" border="1" style="border-collapse:collapse;float:right;">

        <thead>
            <tr>
                <th><strong>ID</strong></th>
                <th><strong>Restaurent </strong></th>
                <th><strong>Customer id</strong></th>
                <th><strong>Price</strong></th>
                <th><strong>Notes</strong></th>
                <th><strong>Created on </strong></th>
                <th><strong>Action </strong></th>
            </tr>
        </thead>
        <tbody>
            <?php
$count=1;
$sel_query="SELECT * FROM `ordertable` WHERE id=$ono AND createdon
BETWEEN '$dt1-INTERVAL 1 DAY' AND '$dt2-INTERVAL 1 DAY';";
$user_arr = array();
$user_arr[] = array("ID","Restaurent","Customer ID","Price","Notes","Created on");
$result = return_array($sel_query);
foreach($result as $row) { ?>
            <tr>
                <td align="center"><?php echo $count; ?></td>
                <?php $id = $count;?>
                <td align="center"><?php echo $row["restaurant_id"]; ?></td>
                <?php $resid=$row["restaurant_id"];?>
                <td align="center"><?php echo $row["customerid"]; ?></td>
                <?php $cusid=$row["customerid"];?>
                <td align="center"><?php echo $row["total_price"]; ?></td>

```

```

<?php $totalprice=$row["total_price"];?>
<td align="center"><?php echo $row["notes"]; ?></td>
<?php $notes=$row["notes"];?>
<td align="center"><?php echo $row["createdon"]; ?></td>
<?php $created=$row["createdon"];?>
<td align="center"><a href="javascript: ;" id=<?php echo $row["id"]; ?>">
class="viewpreviousorder">view
order</a></td>

</tr>
<?php $count++;
$user_arr[] = array($id,$resid,$cusid,$totalprice,$notes,$created);
} ?>
</tbody>
</table>
<?php
echo "<pre>";
$serialize_user_arr = serialize($user_arr);
?>
<textarea name='export_data' style='display: none;'><?php echo $serialize_user_arr;
?></textarea>
</form>

<?php
}

else if($type=="viewtablereport"){
$cond = "";
if($_POST["is_date_search"] == "yes")
{
    $cond = 'AND ordertable.createdon BETWEEN "'.$_POST["start_date"].'" AND
"'.$_POST["end_date"].'";
}
else{
    $cond = 'AND month(ordertable.createdon)=MONTH(CURDATE()) and
year(ordertable.createdon)=YEAR(CURDATE())';
}

```

```

$daily_report = return_array("SELECT tables.table_name as
tablename,table_id,SUM(numberofguests) as numberofguests,SUM(price) as
price,SUM(gst_value) as gst_value,SUM(total_price)as total_price FROM ordertable inner
join tables on (ordertable.table_id = tables.id) where 1=1 ".$cond." group by table_id");
?>
<form method='post' action='download.php'>
<div class=" pl-4">
    <input type='submit' class='btn-md btn btn-primary' value='Export' name='Export'>
</div>
<br><br>
<div class="card">
    <div class="card-body">

        <h4 class="card-title">Table Report</h4>
        <table id="datatable" class="table table-bordered dt-responsive nowrap"
            style="border-collapse: collapse; border-spacing: 0; width: 100%;">
            <thead>
                <tr>
                    <th>S.No</th>
                    <th>Table Id</th>
                    <th>table Name</th>
                    <th>Number of guest</th>
                    <th>Price</th>
                    <th>GST value</th>
                    <th>Total Price</th>
                </tr>
            </thead>
            <tbody>
                <?php
$id = 1;
$c="";
$c1="";
$no=1;
$user_arr = array();
$i=1;
$user_arr[] = array("S.No","Table Id","Table Name","Number of guest","Price","GST
value","Total Price");
foreach ($daily_report as $key => $value) {
?>
                <tr>
                    <td><?php echo $no;?></td>
                    <?php $id= $no;$no++;?>

```

```

        <?php ?>
        <td><?php echo $value["table_id"];
$table_id=$value["table_id"]; ?></td>
        <td><?php echo $value["tablename"];
$table_name=$value["tablename"]; ?></td>
        <td><?php echo $value["numberofguests"];
$numberofguests=$value["numberofguests"]; ?></td>

        <td><?php echo $value["price"];
$price=$value["price"]; ?></td>
        <td><?php echo $value["gst_value"];
$gst_value=$value["gst_value"]; ?></td>
        <td><?php echo $value["total_price"];
$total_price=$value["total_price"]; ?></td>
        </tr>
        <?php $i++;
$user_arr[] = array($id,$table_id,$table_name,$numberofguests,$price,$gst_value,$total_price);
}

?>
</tbody>
</table>
</div>
</div>
<?php
    echo "<pre>";
    $serialize_user_arr = serialize($user_arr);
?
<textarea name='export_data' style='display: none;'><?php echo $serialize_user_arr;
?></textarea>
</form>
<script src="assets/js/pages/datatables.init.js"></script>
<?php
    }
else if($type=="viewcategoryreport"){

$cond = "";
if($_POST["is_date_search"] == "yes")
{
    $cond = 'AND order_items.createdon BETWEEN "'.$_POST["start_date"].'" AND
"'.$_POST["end_date"].'";
}

```

```

    }
else{
    $cond = 'AND month(order_items.createdon)=MONTH(CURDATE()) and
year(order_items.createdon)=YEAR(CURDATE())';
}
$daily_report = return_array("SELECT date(order_items.createdon) as
order_date,category.name as category_name, order_items.category_id as category_id,
sum(order_items.qty) as category_items_quantity , sum(order_items.price) as
category_items_price,sum(order_items.gst_value) as
category_items_gst,sum(order_items.total_price) as category_items_total_price from
order_items inner join category on (order_items.category_id = category.id) where 1=1
".$cond." group by(order_items.category_id)");
?>

<form method='post' action='download.php'>
<div class="pl-4">
    <input type='submit' class=" btn-md btn btn-primary " value='Export' name='Export'>
</div>
<br><br>
<div class="card">
    <div class="card-body">
        <h4 class="card-title">Category Report</h4>
        <table id="datatable" class="table table-bordered dt-responsive nowrap"
style="border-collapse: collapse; border-spacing: 0; width: 100%;">
            <thead>
                <tr>
                    <th>S. No.</th>
                    <th>Order Date</th>
                    <th>Category Name</th>
                    <th>Quantity of Items</th>
                    <th>Category Price</th>
                    <th>GST</th>
                    <th>Total Price</th>
                </tr>
            </thead>
            <tbody>
                <?php
$i = 1;
$user_arr = array();
$user_arr[] = array("S. No.", "Order Date", "Category Name", "Quantity of Items", "Category
Price", "Category GST", "Category Total Price");
foreach ($daily_report as $key => $value) { ?>

```

```

<tr>
    <td><?php echo $i; ?></td>
    <?php $id = $i;?>
    <td><?php echo $value["order_date"];
$createdonval=$value["order_date"]; ?></td>
    <td><?php echo $value["category_name"];
$ordercount=$value["category_name"]; ?></td>
    <td><?php echo $value["category_items_quantity"];
$cusid=$value["category_items_quantity"]; ?></td>
    <td><?php echo $value["category_items_price"];
$price=$value["category_items_price"]; ?></td>
    <td><?php echo $value["category_items_gst"];
$gst_price=$value["category_items_gst"]; ?></td>
    <td><?php echo $value["category_items_total_price"];
$total_price=$value["category_items_total_price"]; ?></td>
</tr>
<?php $i++;
$user_arr[] = array($id,$createdonval,$ordercount,$cusid,$price,$gst,$total);
}
?>
</tbody>
</table>

</div>
</div>
<?php
    echo "
<pre>;
$serialize_user_arr = serialize($user_arr);
?>
<textarea name='export_data' style='display: none;'><?php echo $serialize_user_arr;
?></textarea>
<!-- <textarea name='page_name' style='display: none;'>Category Report</textarea> -->
</form>
<script src='assets/js/pages/datatables.init.js'></script>
<?php
}

else if($type=="viewpreviousorder"){
    $id=$_REQUEST["id"];

```

```

if($id!=""){
    $orderdetails=return_single("SELECT * FROM order_items WHERE
order_id='".$$_REQUEST["id"].".'; ");
    $value = return_single("SELECT * FROM ordertable where
id='".$$_REQUEST["id"].".'");
    $get_order_items = return_array("SELECT * FROM order_items WHERE
order_id = $id");
    ?>
<table style="margin-bottom: 20px;">
    <thead>
        <tr>
            <td colspan="4">Order ID - #<?php echo $id; ?></td>
        </tr>
        <tr>
            <th>S. No.</th>
            <th>Item Name</th>
            <th>Quantity</th>
            <th>Price</th>
        </tr>
    </thead>
    <tbody>
        <?php
            $i = 1;
            foreach ($get_order_items as $key1 => $value1) {
                $get_item_det = return_single("SELECT * FROM items WHERE id =
".$value1["item_id"]."");
                ?>
                <tr>
                    <td style="text-align:center;"><?php echo $i; ?></td>
                    <td><?php echo $get_item_det["name"]; ?></td>
                    <td style="text-align:center;"><?php echo $value1["qty"]; ?></td>
                    <td><?php echo $value1['price']?></td>
                </tr>

            <?php
                $i++;
            } ?>
            <tr>
                <td colspan="3" style="text-align:center;">Price</td>
                <td style="text-align:center;"><?php echo $value1["price"]?></td>
            </tr>
            <tr>

```

```

<td colspan="3" style="text-align:center;">GST 18%</td>
<td style="text-align:center;"><?php echo $value["gst_value"]?></td>
</tr>
<tr>
    <td colspan="3" style="text-align:center;">Total Value</td>
    <td style="text-align:center;"><?php echo $value["total_price"]?></td>
</tr>
</tbody>
</table>
<?php
    }
}
else if($type=="viewnotes"){
    $id=$_REQUEST["id"];

    if($id!="){

        $note=return_single("SELECT notes FROM ordertable WHERE
id=""$. $_REQUEST["id"].""); ?>

<!-- Notes: -->
<?php echo $note["notes"]; ?>

<?php

    }
}
else if($type=="extraqty"){
$item_id=$_REQUEST["item_id"];
if($item_id!="){
    $item_det=return_single("SELECT * FROM order_items WHERE id=
""$. $_REQUEST["item_id"]." ");
    if($item_det!="{
        $value1=return_single("SELECT qty FROM order_items WHERE
id=""$. $_REQUEST["item_id"]."");
        ?>
<input type="hidden" name="hiditemid" id="hiditemid" class="hiditemid" value="<?php
echo $_REQUEST["item_id"]; ?>" />
Quantity:
<input type="number" name="quantity" id="quantity" min="1" value="<?php echo
$value1["qty"]; ?>">

```

```

<?php
    }
}
}
else if($type =='fetch_vendor')
{
    $vendor = $_REQUEST["vendor"];
    $fetch_details = return_array("SELECT name FROM items WHERE vendor=
".$vendor."");
?

```

```

<div class="row">
    <div class="col-12">
        <div class="card">
            <div class="card-body">
                <h4 class="card-title">Products</h4>
                <table id="datatable" class="table table-bordered dt-responsive nowrap"
                    style="border-collapse: collapse; border-spacing: 0; width: 100%;">
                    <thead>
                        <tr>
                            <th>S.no</th>
                            <th>owner_name</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php
                            $count=1;
                            $sel_query="SELECT name FROM items WHERE vendor=
".$vendor."";
                            $result = return_array($sel_query);
                            foreach($result as $row) { ?>
                        <tr>
                            <td><?php echo $count; ?></td>
                            <td><?php echo $row["name"]; ?></td>
                        </tr>
                        <?php $count++; } ?>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
<!--end col-->

```

```

</div>
<?php

}

else if($type == 'open_item_category'){
    $get_cat_det = return_single("SELECT * FROM category WHERE id =
".$_REQUEST["cat_id"]." AND restaurant_id =
".$_RESTAURANT_ID."");
    $get_items = return_array("SELECT * FROM items WHERE category_id =
".$_REQUEST["cat_id"]." AND restaurant_id =
".$_RESTAURANT_ID."");
    $foodtype_arr = array("1"=>"Veg","2"=>"Non-Veg","3"=>"Egg")
?>
<div class="item-list">
    <table id="datatable" class="table table-bordered dt-responsive nowrap"
        style="border-collapse: collapse; border-spacing: 0; width: 100%;">

        <?php if(count($get_items) > 0){ ?>
        <thead>
            <tr>
                <th> Items </th>
                <th> Type </th>
                <th> Base Price</th>
                <th> Margin Price</th>
                <th> Profit</th>
                <th> Total Price</th>
                <th> Availability </th>
                <th> Delete </th>
                <th> Actions</th>
            </tr>
        </thead>
        <?php } ?>

        <tbody>
            <?php
                if(count($get_items) > 0){
                    foreach ($get_items as $key => $value) {
                        $checked = "";
                        if($value["is_available"] == '1'){
                            $checked = "checked";
                        }
                }
            </tbody>

```

```

?>

<tr scope="row">

<td><?php echo $value["name"] ?></td>
<td><?php if(($value["foodtype"])==1)
{ ?>
    
    </span><?php echo $foodtype_arr[$value["foodtype"]]; ?>
    <?php }
else{ ?>
    
    </span><?php echo $foodtype_arr[$value["foodtype"]]; ?>
    <?php } ?>
</td>
<td><?php echo $value["price"]; ?></td>
<td><?php echo $value["margin_price"]; ?></td>
<td><?php echo $value["profit"]; ?></td>
<td><?php echo $value["total_price"]; ?></td>

<td>
    <label class="switch">
        <input type="checkbox" class="toggle_item_available" id="<?php echo
$value["id"] ?> switch1"
            <?php echo $checked; ?>>
        <span class="slider"></span>
    </label>

    <!-- <span class="badge badge-pill ">
        <input type="checkbox" class="toggle_item_available" id="<?php echo
$value["id"] ?> switch1"
            switch="primary" <?php echo $checked; ?>>
        <label for="<?php echo $value["id"] ?> switch1" data-on-label="Yes"
data-off-label="No"></label>
    </span>

    <span class="badge badge-pill list-group-flush">
        <input type="checkbox" class="toggle_item_available" id="<?php echo
$value["id"] ?> switch1"
            switch="primary" name="toggle" <?php echo $checked; ?>>

```

```

        <label for=<?php echo $value["id"] ?> switch1" data-on-label="Yes"
data-off-label="No"></label>
        </span> -->
</td>

<td><a href="#" data-toggle="modal" class="delete_item"
data-target="#delete_category"
        id=<?php echo $value["id"] ?>">Delete</a></td>
<td><a href="#" id=<?php echo $value["id"] ?>" class="edit_item">Edit
</a></td>
<?php }
} else{ ?
<li class="list-group-item d-flex justify-content-between align-items-center">NO
ITEMS FOUND!</li>
<?php } ?>
</tbody>
</table>
</div>
<script src="../assets/js/pages/datatables.init.js"></script>
<?php
}
else if($type == "view_add_order"){
    $order_id = $_REQUEST["order_id"];
    $status_common = array("0"=>"Pending","1"=>"In Progress","2"=>"Ready to
deliver","3"=>"Delivered","4"=>"Completed");
    if($order_id != ""){
        $orders_det = return_single("SELECT * FROM ordertable WHERE id =
"".$_REQUEST["order_id"]." AND confirmed = '1' AND completed = '0'");
        ?><div class="mb-3 font-weight-bold"><?php
        echo("Number of Guests - ".$orders_det['numberofguests'].");
        $numberofguests=$orders_det['numberofguests'];?></div><?php
        if($orders_det["id"] != ""){
            $order_items_det = return_array("SELECT * FROM order_items WHERE order_id
= ".$orders_det["id"].");
            ?>
<table id="datatable" class="table mb-0 table-bordered dt-responsive nowrap w-100">
<thead>
<tr>
<th>S. No.</th>
<th>Item Name</th>
<th>Qty</th>
<th>Status</th>

```

```

        </tr>
    </thead>
    <tbody><?php
$i = 1;
foreach ($order_items_det as $key => $value) {
    $item_det = return_single("SELECT * FROM items WHERE id =
".$value["item_id"]." AND restaurant_id = ".RESTAURANT_ID."");
?>
<tr>
    <td><?php echo $i; ?></td>
    <td><?php echo $item_det["name"]; ?></td>
    <td><?php echo $value["qty"]; ?></td>
    <td><?php echo $status_common[$value["status"]]; ?></td>
</tr>
<?php
    $i++;
} ?>
</tbody>
</table>
<?php
    }
}
else{
    echo 0;
} ?>
<script src="assets/js/pages/datatables.init.js">
</script>
<?php
}
?>

```

**FILE NAME:** common\_controller.php ( **MAIN FILE FOR CUSTOMER MODULE**)

```

<?php
include_once('../config/dbconn.php');
include_once('../config/constant.php');
include_once('../sendsms/trigger.php');
ini_set('display_errors', '0');
session_start();
$type = $_REQUEST['Type'];

```

```

if($type == "login_getstarted"){
    $otpno = rand(1000,9999);
    $check_table_free = return_single("SELECT completed FROM ordertable WHERE table_id = ".$REQUEST["table_id"]." AND (SELECT status FROM tables WHERE id = ".$REQUEST["table_id"]."") = 0");
    if($check_table_free["completed"] != "0"){
        $check_user = return_single("SELECT COUNT(1) as cid FROM customer WHERE email = ".$REQUEST["email"]." OR mobile = ".$REQUEST["mobile"]."");
        if($check_user["cid"] == "0"){
            $login_getstarted = gosql1("INSERT INTO customer (name,email,mobile,otpno,createdon) VALUES ('".$REQUEST["name"]."','".$REQUEST["email"]."','".$REQUEST["mobile"]."','".$otpno."','".$now()."')");
        }else{
            $login_getstarted = gosql("UPDATE customer SET name = '".$REQUEST["name"]."',otpno = '".$otpno."' WHERE email = ".$REQUEST["email"]." OR mobile = ".$REQUEST["mobile"]."");
            $get_customer_id = return_single("SELECT id FROM customer WHERE email = ".$REQUEST["email"]." OR mobile = ".$REQUEST["mobile"]."");
            $login_getstarted = $get_customer_id["id"];
        }
        $insert_order = gosql1("INSERT INTO ordertable (restaurant_id,customerid,table_id,numberofguests,createdon) VALUES ('".$RESTAURANT_ID."','".$login_getstarted."','".$REQUEST["table_id"]."','".$REQUEST["numberofguests"]."','".$now()."')");
        $_SESSION["customer"]['ID'] = $login_getstarted;
        $_SESSION["customer"]['name'] = $REQUEST['name'];
        $_SESSION["customer"]['email'] = $REQUEST['email'];
        $_SESSION["customer"]['mobile'] = $REQUEST['mobile'];
        $_SESSION["customer"]['table_id'] = $REQUEST['table_id'];
        $_SESSION["customer"]['order_id'] = $insert_order;
        $_SESSION["customer"]['restaurant_id'] = RESTAURANT_ID;

        echo $login_getstarted;
    }else{
        echo 0;
    }
}

```

```

else if($type == "confirm_otp"){
    $customerid = $_REQUEST["customer_id"];
    $otp = $_REQUEST["otp"];

    $check_otp = return_single("SELECT COUNT(1) AS cid FROM customer WHERE
otpno = ".$otp." AND id = ".$customerid."");
    if($check_otp["cid"] > 0){
        echo 1;
    }else{
        echo 0;
    }
}

else if($type == "add_to_cart"){
    $customerid = $_SESSION["customer"]["ID"];
    $order_id = $_SESSION["customer"]["order_id"];
    $restaurant_id = $_SESSION["customer"]["restaurant_id"];
    $items_ids = $_REQUEST["items_ids"];

    $items_arr = explode(", ", $items_ids);
    foreach ($items_arr as $key => $value) {
        $items_value = explode('#', $value);
        $price = $items_value[2] * $items_value[0];
        $get_items_det = return_single("SELECT category_id,gst_value,total_price
FROM items WHERE id = ".$items_value[1]."");

        $add_to_cart = gosql("INSERT INTO order_items
(order_id, customer_id, restaurant_id, category_id, item_id, qty, price, gst_value, total_price, createdon) VALUES
('".$order_id."','".$customerid."','".$RESTAURANT_ID."','".$.$get_items_det["category_id"]."',
'".$items_value[1]."','".$items_value[0]."','".$price."','".$.$get_items_det["gst_value"]."','".$.
$get_items_det["total_price"]."','now())");
    }

    $check_cnt_completed = return_single("SELECT count(1) AS cid FROM
order_items WHERE status = '4' AND order_id = ".$order_id." AND customer_id =
".$customerid."");

    $get_total_item_order = return_single("SELECT count(1) AS cid FROM
order_items WHERE order_id = ".$order_id." AND customer_id = ".$customerid."");
    if($check_cnt_completed["cid"] == $get_total_item_order["cid"]){
        $update_status_order = gosql("UPDATE ordertable SET completed = '1'
WHERE id = ".$order_id." AND customer_id = ".$customerid."");
    }
}

```

```

    }else{
        $update_status_order = gosql("UPDATE ordertable SET completed = '0'
WHERE id = ".$order_id." AND customer_id = ".$customerid."");
    }
}

else if($type == "load_order_items"){
    $customerid = $_SESSION["customer"]["ID"];
    $order_id = $_SESSION["customer"]["order_id"];
    $restaurant_id = $_SESSION["customer"]["restaurant_id"];
    $cart_order_items = return_array("SELECT * FROM order_items WHERE order_id
= ".$order_id." AND customer_id = ".$customerid." AND restaurant_id =
".$RESTAURANT_ID."");
    ?>
    <div class="col-12 px-0">
        <ul class="list-group list-group-flush mb-4">
            <?php foreach ($cart_order_items as $key => $value) {
                $get_item_det = return_single("SELECT name,category_id,price,gst_value
FROM items WHERE id = ".$value["item_id"]."");
                $get_cat = return_single("SELECT name FROM category WHERE id =
".$get_item_det["category_id"]."");
                $total_price += $value["price"];
                $total_gst += $get_item_det["gst_value"];
            ?>
            <li class="list-group-item">
                <div class="row">
                    <div class="col-auto align-self-center">
                        <a href="javascript:;" id="<?php echo $value["id"]; ?>" class="btn
btn-sm btn-link p-0 float-right delete_item"><i
class="material-icons">remove_circle</i></a>
                    </div>
                    <div class="col-2 pl-0 align-self-center">
                        <figure class="product-image h-auto"></figure>
                    </div>
                    <div class="col px-0">
                        <a href="#" class="text-dark mb-1 h5 d-block"><?php echo
$get_item_det["name"] ?></a>
                        <h5 class="text-success font-weight-normal mb-0"><?php echo
$value["price"] ?></h5>
                        <p class="text-secondary h6 text-mute mb-0">Category: <?php echo
$get_cat["name"]; ?></p>
                    </div>
                </div>
            </li>
        </ul>
    </div>
}

```

```

        </div>
        <div class="col-auto align-self-center">
            <div class="input-group input-group-sm">
                <div class="input-group-prepend">
                    <a href="javascript:;" class="btn btn-light-orange px-1 sub"><i
class="material-icons">remove</i></a>
                </div>
                <input type="text" class="form-control w-35 txtqty" placeholder
id="<?php echo $value["id"]; ?>" price="<?php echo $get_item_det["price"]; ?>"
value="<?php echo $value["qty"] ?>">
                <div class="input-group-append">
                    <a href="javascript:;" class="btn btn-light-orange px-1 add"><i
class="material-icons">add</i></a>
                </div>
            </div>
        </div>
    </li>
    <?php } ?>
</ul>
</div>
<?php
}
else if($type == "delete_order_item"){
    $order_item_id = $_REQUEST["order_item_id"];
    $delete_order_item = gosql("DELETE FROM order_items WHERE id =
".$order_item_id."");
}
else if($type == "cart_update_pricing"){
    $order_item_id = $_REQUEST["order_item_id"];
    $price = $_REQUEST["price"];
    $updated_qty = $_REQUEST["updated_qty"];
    $final_price = $price * $updated_qty;
    $cart_update_pricing = gosql("UPDATE order_items SET price =
".$final_price.",qty = ".$updated_qty." WHERE id = ".$order_item_id."");
}
else if($type == "cart_update_pricing"){
    $order_item_id = $_REQUEST["order_item_id"];
    $price = $_REQUEST["price"];
}

```

```

$updated_qty = $_REQUEST["updated_qty"];
$final_price = $price * $updated_qty;
$cart_update_pricing = gosql("UPDATE order_items SET price =
".$final_price.",qty = ".$updated_qty." WHERE id = ".$order_item_id."");
}

else if($type == "cart_update_totals"){
    $customerid = $_SESSION["customer"]["ID"];
    $order_id = $_SESSION["customer"]["order_id"];
    $restaurant_id = $_SESSION["customer"]["restaurant_id"];

    $get_prices = return_array("SELECT price,item_id FROM order_items WHERE
order_id = ".$order_id." AND customer_id = ".$customerid." AND restaurant_id =
".$restaurant_id."");

    foreach ($get_prices as $key => $value) {
        $total_price += $value["price"];
        $get_gstval = return_single("SELECT (SELECT percent FROM gst_master
WHERE id = it.gst) AS gstval FROM items AS it WHERE id = ".$value["item_id"].""
AND restaurant_id = ".$restaurant_id."");
        $gst_value = ($value["price"] * $get_gstval["gstval"])/100;
        $total_gst += $gst_value;
    }
    $price_arr = array();
    $price_arr["total_price"] = $total_price;
    $price_arr["total_gst"] = $total_gst;
    $total_price_with_gst = $total_price + $total_gst;
    $price_arr["total_price_with_gst"] = $total_price_with_gst;
    $update_prices_ordertable = gosql("UPDATE ordertable SET price =
".$total_price.",gst_value = ".$total_gst.",total_price = ".$total_price_with_gst."
WHERE id = ".$order_id." AND restaurant_id = ".$restaurant_id."");
    echo json_encode($price_arr);
}

else if($type == "confirm_order"){
    $order_id = $_SESSION["customer"]["order_id"];
    $restaurant_id = $_SESSION["customer"]["restaurant_id"];
    $confirm_order = gosql("UPDATE ordertable SET confirmed = '1' WHERE id =
".$order_id."");
}

else if($type == "create_table"){
    $order_id = $_REQUEST["order_id"];
}

```

```

$customer_id = $_SESSION["customer"]["ID"];
$restaurant_id = $_SESSION["customer"]["restaurant_id"];
$check_order_completed = return_single("SELECT completed FROM ordertable
WHERE id = ".$order_id." AND customerid = ".$customer_id." AND restaurant_id =
".$restaurant_id."");
if($check_order_completed["completed"] == "1"){
    $generate_bill = gosql("UPDATE ordertable SET generate_bill = '1' WHERE
id = ".$order_id." AND customerid = ".$customer_id." AND restaurant_id =
".$restaurant_id."");
    echo 1;
}

}

else if($type == "generate_bill"){
    $order_id = $_REQUEST["order_id"];
    $customer_id = $_SESSION["customer"]["ID"];
    $restaurant_id = $_SESSION["customer"]["restaurant_id"];
    $check_order_completed = return_single("SELECT completed FROM ordertable
WHERE id = ".$order_id." AND customerid = ".$customer_id." AND restaurant_id =
".$restaurant_id."");
    if($check_order_completed["completed"] == "1"){
        $generate_bill = gosql("UPDATE ordertable SET generate_bill = '1' WHERE
id = ".$order_id." AND customerid = ".$customer_id." AND restaurant_id =
".$restaurant_id."");
        echo 1;
    }
    else{
        echo 0;
    }
}

else if($type == "load_order_items_billpay"){
    $customerid = $_SESSION["customer"]["ID"];
    $order_id = $_SESSION["customer"]["order_id"];
    $restaurant_id = $_SESSION["customer"]["restaurant_id"];
    $cart_order_items = return_array("SELECT * FROM order_items WHERE order_id =
".$order_id." AND customer_id = ".$customerid." AND restaurant_id =
".$RESTAURANT_ID."");
    ?>
    <div class="col-12 px-0">
        <ul class="list-group list-group-flush mb-4">

```

```

<?php foreach ($cart_order_items as $key => $value) {
    $get_item_det = return_single("SELECT name,category_id,price,gst_value
FROM items WHERE id = ".$value["item_id"]."");
    $get_cat = return_single("SELECT name FROM category WHERE id =
".$get_item_det["category_id"]."");
    $total_price += $value["price"];
    $total_gst += $get_item_det["gst_value"];
}
<li class="list-group-item">
    <div class="row">
        <div class="col-2 pl-0 align-self-center">
            <figure class="product-image h-auto"></figure>
        </div>
        <div class="col px-0">
            <a href="#" class="text-dark mb-1 h6 d-block"><?php echo
$get_item_det["name"] ?></a>

            <p class="text-secondary small text-mute mb-0">Category: <?php
echo $get_cat["name"]; ?></p>
        </div>
        <div class="col-auto align-self-center">
            <h5 class="text-success font-weight-normal mb-0"><?php echo
$value["price"] ?></h5>
        </div>
        </div>
    </li>
    <?php } ?>
</ul>
</div>
<?php
}
?>

```

**FILE NAME:** digitalmenu.sql ( Database source code for entire project)

```
-- phpMyAdmin SQL Dump
-- version 5.0.1
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Apr 05, 2021 at 05:24 AM
-- Server version: 10.4.11-MariaDB
-- PHP Version: 7.4.3

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET AUTOCOMMIT = 0;
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

-- Database: `digitalmenu`



-----



-- Table structure for table `admin`


--



CREATE TABLE `admin` (
`id` int(11) NOT NULL,
`name` varchar(100) NOT NULL,
`username` varchar(100) NOT NULL,
`email` varchar(250) NOT NULL,
`password` varchar(250) NOT NULL,
`date` datetime NOT NULL,
```

```
`role` varchar(150) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- 
-- Dumping data for table `admin`
-- 

INSERT INTO `admin` (`id`, `name`, `username`, `email`, `password`, `date`, `role`)
VALUES
(1, 'Gopi', 'admin', 'gopinath93@gmail.com', '81dc9bdb52d04dc20036dbd8313ed055',
'2018-10-06 08:57:39', '1');

-----
-- 
-- Table structure for table `category`
-- 

CREATE TABLE `category` (
`id` int(11) NOT NULL,
`restaurant_id` int(11) NOT NULL,
`name` varchar(400) NOT NULL,
`description` text NOT NULL,
`is_available` int(11) NOT NULL DEFAULT 0,
`createdon` datetime NOT NULL,
`createdby` int(11) NOT NULL DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- 
-- Dumping data for table `category`
-- 

INSERT INTO `category` (`id`, `restaurant_id`, `name`, `description`, `is_available`,
`createdon`, `createdby`) VALUES
(1, 1, 'Starter', 'Starter is good', 1, '2020-06-06 19:10:46', 1),
(2, 1, 'Dessert', 'Dessert', 1, '2020-06-06 19:11:48', 1),
(3, 1, 'Soft Drinks', 'Soft Drinks', 1, '2020-06-06 19:13:57', 1),
(4, 1, 'Biryani', 'Biryani', 0, '2020-06-06 19:17:15', 1),
(9, 1, 'mocktail', 'juice', 1, '2021-04-04 12:31:35', 1);
```

```

-- 
-- Table structure for table `customer` 

-- 

CREATE TABLE `customer` (
  `id` int(11) NOT NULL,
  `name` varchar(400) NOT NULL,
  `email` varchar(400) NOT NULL,
  `mobile` varchar(400) NOT NULL,
  `otpno` int(11) NOT NULL DEFAULT 0,
  `createdon` datetime NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- 
-- Dumping data for table `customer` 

-- 

INSERT INTO `customer` (`id`, `name`, `email`, `mobile`, `otpno`, `createdon`) VALUES
(1, 'Gopinath', 'gopinath93@gmail.com', '9944526954', 4226, '2020-06-28 21:33:01'),
(2, 'Rakshna', 'rakshna@gmail.com', '8754557900', 2798, '2020-06-28 22:59:42'),
(3, 'LOGESH KARTHIK', 's.logeshkarthik@gmail.com', '9551783347', 5367, '2020-09-30 18:34:51');

----- 

-- 
-- Table structure for table `department` 

-- 

CREATE TABLE `department` (
  `id` int(11) NOT NULL,
  `deptname` varchar(200) NOT NULL,
  `deptcode` varchar(200) NOT NULL,
  `noemp` varchar(200) NOT NULL,
  `resource` varchar(200) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- 
-- Dumping data for table `department` 

-- 

INSERT INTO `department` (`id`, `deptname`, `deptcode`, `noemp`, `resource`) VALUES

```

```
(21, 'Manager', 'MAN-792020184017643', '2', '1000'),  
(22, 'kitchen', 'KIT-792020184155273', '4', '10000'),  
(23, 'cashier', 'CAS-792020184218348', '1', '1000'),  
(24, 'waiter', 'WAI-792020184234603', '5', '10000'),  
(26, 'test1', 'TES-44202112341991', '2', '1000');
```

---

```
--  
-- Table structure for table `employees`  
--
```

```
CREATE TABLE `employees` (  
  `id` int(11) NOT NULL,  
  `code` varchar(100) NOT NULL,  
  `name` varchar(150) NOT NULL,  
  `username` varchar(60) NOT NULL,  
  `password` varchar(245) NOT NULL,  
  `faname` varchar(150) NOT NULL,  
  `dob` date NOT NULL,  
  `aadhar` varchar(150) NOT NULL,  
  `pan` varchar(150) NOT NULL,  
  `doj` date NOT NULL,  
  `mail` varchar(150) NOT NULL,  
  `phone` varchar(150) NOT NULL,  
  `role` varchar(150) NOT NULL,  
  `depart` varchar(150) NOT NULL,  
  `accno` varchar(160) NOT NULL,  
  `ifsc` varchar(60) NOT NULL,  
  `document` text DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Dumping data for table `employees`  
--
```

```
INSERT INTO `employees` (`id`, `code`, `name`, `username`, `password`, `faname`, `dob`,  
  `aadhar`, `pan`, `doj`, `mail`, `phone`, `role`, `depart`, `accno`, `ifsc`, `document`)  
VALUES  
(10, 'LST-19990708', 'logesh karthik', 'logesh', '37ace2e2e6319a33972412d0e6c96429',  
  'xyz', '1999-07-08', '1111-1111-1111-1111', 'EJAPS0276M', '2021-04-08',  
  's.logeshkarthik@gmail.com', '9551783347', 'test1', 'test role', '10818', "", NULL);
```

```
--  
-- Table structure for table `gst_master`  
  
CREATE TABLE `gst_master` (  
  `id` int(11) NOT NULL,  
  `percent` int(11) NOT NULL,  
  `createdon` datetime NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
--  
-- Dumping data for table `gst_master`  
  
INSERT INTO `gst_master` (`id`, `percent`, `createdon`) VALUES  
(1, 5, '2020-06-06 16:56:43'),  
(2, 12, '2020-06-06 16:57:43'),  
(3, 18, '2020-06-06 16:58:43'),  
(4, 28, '2020-06-06 16:59:43');
```

```
--  
-- Table structure for table `items`  
  
CREATE TABLE `items` (  
  `id` int(11) NOT NULL,  
  `name` varchar(400) NOT NULL,  
  `description` text NOT NULL,  
  `restaurant_id` int(11) NOT NULL,  
  `category_id` int(11) NOT NULL,  
  `is_available` int(11) NOT NULL DEFAULT 0,  
  `id_addon` int(11) NOT NULL DEFAULT 0,  
  `foodtype` int(11) NOT NULL,  
  `price` float(10,2) NOT NULL,  
  `parcel_charges` int(11) NOT NULL,  
  `gst` int(11) NOT NULL,  
  `gst_value` float NOT NULL,
```

```

`total_price` float NOT NULL,
`total_price_parcel` float NOT NULL,
`url` text NOT NULL,
`image` text DEFAULT NULL,
`sortby` int(11) NOT NULL DEFAULT 0,
`item_image` text DEFAULT NULL,
`createdon` datetime NOT NULL,
`createdby` int(11) NOT NULL DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- 
-- Dumping data for table `items`
-- 

INSERT INTO `items` (`id`, `name`, `description`, `restaurant_id`, `category_id`, `is_available`, `id_addon`, `foodtype`, `price`, `parcel_charges`, `gst`, `gst_value`, `total_price`, `total_price_parcel`, `url`, `image`, `sortby`, `item_image`, `createdon`, `createdby`) VALUES
(1, 'Chicken Lollipop', 'Chicken Lollipop', 1, 1, 1, 0, 2, 100.00, 5, 1, 5, 105, 110, 'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 1, '', '2020-06-06 20:11:27', 1),
(2, 'Vanilla Ice Cream', 'Vanilla Ice Cream', 1, 2, 1, 0, 1, 100.00, 5, 1, 5, 105, 110, 'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 1, NULL, '2020-06-06 20:26:24', 1),
(4, 'Chocolate Ice Cream', 'Chocolate Ice Cream', 1, 2, 0, 0, 1, 100.00, 5, 1, 5, 105, 110, 'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 2, '', '2020-06-06 20:29:03', 1),
(5, 'Chicken Biryani', 'Chicken Biryani', 1, 5, 1, 0, 2, 100.00, 5, 1, 5, 105, 110, 'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 1, NULL, '2020-06-06 21:29:20', 1),
(6, 'Mutton Biryani', 'Mutton Biryani', 1, 5, 1, 0, 2, 100.00, 5, 1, 5, 105, 110, 'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 0, NULL, '2020-06-14 22:51:00', 1),
(7, 'Watermelon Juice', 'Watermelon Juice', 1, 7, 1, 0, 1, 100.00, 5, 1, 5, 105, 110, 'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 0, NULL, '2020-06-14 23:21:30', 1),
(8, 'Grill Chicken', 'Grill Chicken', 1, 1, 1, 0, 2, 100.00, 5, 1, 5, 105, 110, 'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 0, '', '2020-06-15 22:55:13', 1),
(9, 'Half Plate', 'Half Plate', 1, 8, 1, 0, 2, 100.00, 5, 1, 5, 105, 110, 'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 0, '', '2020-06-28 22:53:05', 1),

```

```
(10, 'Chocolate', 'Mixes with milk and choco powder
', 1, 3, 0, 0, 1, 100.00, 5, 1, 5, 105, 110,
'http://localhost/foox-digital-restaurant/backend/add-menu.php', NULL, 0, ", '2020-07-27
14:07:38', 1),
(11, 'chiken biriyani', 'chiken biriyani', 1, 4, 1, 0, 2, 100.00, 5, 1, 5, 105, 110, ", NULL, 0, ",
'2020-09-07 21:58:01', 1),
(12, 'apple mocktail', 'juice', 1, 9, 1, 0, 1, 101.00, 5, 1, 5.05, 106.05, 111.05, ", NULL, 0, ",
'2021-04-04 12:32:21', 1);
```

---

```
--  
-- Table structure for table `notification`  
--
```

```
CREATE TABLE `notification` (
`id` int(11) NOT NULL,
`tname` varchar(100) NOT NULL,
`status` int(30) NOT NULL DEFAULT 0,
`created_date` datetime NOT NULL,
`updated_date` datetime NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Dumping data for table `notification`  
--
```

```
INSERT INTO `notification` (`id`, `tname`, `status`, `created_date`, `updated_date`)
VALUES
(1, 'Table 1', 0, '2020-07-25 00:00:00', '2020-07-25 00:00:00'),
(2, 'Table 2', 0, '2020-07-25 00:00:00', '2020-07-25 00:00:00'),
(3, 'Table 3', 0, '2020-07-25 00:00:00', '2020-07-25 00:00:00'),
(4, 'Table 4', 0, '2020-07-25 00:00:00', '2020-07-25 00:00:00');
```

---

```
--  
-- Table structure for table `ordertable`  
--
```

```
CREATE TABLE `ordertable` (
`id` int(11) NOT NULL,
```

```

`restaurant_id` int(11) NOT NULL DEFAULT 1,
`customerid` varchar(40) NOT NULL DEFAULT '',
`table_id` int(11) NOT NULL DEFAULT 0,
`numberofguests` int(11) NOT NULL,
`price` int(50) NOT NULL DEFAULT 0,
`gst_value` int(50) NOT NULL DEFAULT 0,
`total_price` int(50) NOT NULL DEFAULT 0,
`notes` varchar(255) NOT NULL,
`confirmed` int(11) NOT NULL DEFAULT 0,
`completed` int(11) NOT NULL DEFAULT 0,
`generate_bill` int(11) NOT NULL DEFAULT 0,
`customer_completed` int(11) NOT NULL DEFAULT 0,
`createdon` datetime DEFAULT NULL,
`createdby` int(11) NOT NULL DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

-- 
-- Dumping data for table `ordertable`
-- 
```

```

INSERT INTO `ordertable`(`id`, `restaurant_id`, `customerid`, `table_id`,
`numberofguests`, `price`, `gst_value`, `total_price`, `notes`, `confirmed`, `completed`,
`generate_bill`, `customer_completed`, `createdon`, `createdby`) VALUES
(1, 1, '1', 2, 3, 300, 15, 315, 'Nothing to say', 1, 0, 0, 0, '2020-09-23 04:00:00', 0),
(12, 1, '3', 0, 0, 200, 10, 210, ", 1, 0, 0, 0, '2020-12-31 18:55:50', 0),
(13, 1, '3', 0, 0, 0, 0, ", 0, 0, 0, 0, '2021-01-03 02:15:55', 0),
(14, 1, '3', 0, 0, 300, 10, 310, ", 1, 1, 0, 0, '2021-01-03 02:31:40', 0),
(15, 1, '3', 0, 0, 0, 0, ", 0, 0, 0, 0, '2021-01-04 02:17:50', 0),
(16, 1, '3', 0, 0, 200, 10, 210, ", 1, 1, 0, 0, '2021-01-04 02:17:51', 0),
(17, 1, '3', 0, 3, 300, 5, 305, ", 1, 1, 1, 0, '2021-01-08 17:31:22', 0),
(18, 1, '3', 0, 2, 100, 5, 105, ", 1, 1, 0, 0, '2021-02-28 19:17:17', 0),
(19, 1, '3', 0, 4, 101, 5, 106, ", 1, 1, 1, 0, '2021-04-04 12:42:54', 0),
(20, 1, '3', 0, 2, 0, 0, 0, ", 0, 0, 0, 0, '2021-04-04 12:50:55', 0); 
```

---

```

-- 
-- Table structure for table `order_items`
-- 
```

```

CREATE TABLE `order_items` (
`id` int(11) NOT NULL, 
```

```

`order_id` int(11) NOT NULL DEFAULT 0,
`table_id` int(11) NOT NULL DEFAULT 0,
`customer_id` int(11) NOT NULL DEFAULT 0,
`restaurant_id` int(11) NOT NULL DEFAULT 0,
`item_id` int(11) NOT NULL DEFAULT 0,
`category_id` int(30) NOT NULL,
`createdon` datetime NOT NULL,
`qty` int(11) NOT NULL DEFAULT 0,
`price` float(10,2) NOT NULL DEFAULT 0.00,
`gst_value` float(10,2) NOT NULL DEFAULT 0.00,
`total_price` float(10,2) NOT NULL DEFAULT 0.00,
`status` int(11) NOT NULL DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```
--  
-- Dumping data for table `order_items`  
--
```

```

INSERT INTO `order_items` (`id`, `order_id`, `table_id`, `customer_id`, `restaurant_id`,
`item_id`, `category_id`, `createdon`, `qty`, `price`, `gst_value`, `total_price`, `status`)
VALUES
(1, 1, 2, 1, 1, 1, '2020-09-07 22:30:23', 1, 100.00, 10.00, 110.00, 3),
(2, 1, 2, 1, 1, 2, '2020-09-07 22:30:23', 1, 100.00, 10.00, 110.00, 3),
(3, 1, 2, 1, 1, 4, 2, '2020-09-07 22:28:17', 1, 100.00, 10.00, 110.00, 2),
(54, 12, 0, 3, 1, 6, 5, '2020-12-31 18:56:19', 1, 100.00, 5.00, 105.00, 3),
(55, 12, 0, 3, 1, 7, 7, '2020-12-31 18:56:20', 1, 100.00, 5.00, 105.00, 1),
(56, 14, 0, 3, 1, 6, 5, '2021-01-03 02:32:06', 2, 200.00, 5.00, 105.00, 3),
(57, 14, 0, 3, 1, 7, 7, '2021-01-03 02:33:01', 1, 100.00, 5.00, 105.00, 0),
(58, 16, 0, 3, 1, 7, 7, '2021-01-04 02:18:21', 1, 100.00, 5.00, 105.00, 3),
(59, 16, 0, 3, 1, 8, 1, '2021-01-04 02:18:21', 1, 100.00, 5.00, 105.00, 3),
(60, 17, 0, 3, 1, 1, 1, '2021-01-08 17:33:15', 3, 300.00, 5.00, 105.00, 3),
(61, 18, 0, 3, 1, 6, 5, '2021-02-28 19:22:55', 1, 100.00, 5.00, 105.00, 3),
(62, 19, 0, 3, 1, 12, 9, '2021-04-04 12:43:38', 1, 101.00, 5.05, 106.05, 3);

```

```
--  
-- Table structure for table `pages`  
--
```

```

CREATE TABLE `pages` (
`Id` int(11) NOT NULL,

```

```
`Page_Name` varchar(50) DEFAULT NULL,  
 `Status` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
--  
-- Dumping data for table `pages`  
--  
  
INSERT INTO `pages` (`Id`, `Page_Name`, `Status`) VALUES  
(1, 'Dashboard', 0),  
(2, 'Users', 0),  
(3, 'Departments', 0),  
(4, 'Tables', 0),  
(5, 'Role Mapping', 0),  
(6, 'Orders', 0),  
(7, 'Daily Report', 0),  
(8, 'Monthly Report', 0),  
(9, 'Items Report', 0),  
(10, 'Tables Report', 0),  
(11, 'Category Report', 0),  
(12, 'Employees', 0),  
(13, 'Profiles', 0),  
(14, 'Settings', 0),  
(15, 'Roles', 0),  
(16, 'Menu', 0),  
(17, 'index', 0);
```

---

```
--  
-- Table structure for table `pin`  
--  
  
CREATE TABLE `pin` (  
 `pin_number` varchar(30) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Dumping data for table `pin`  
--  
  
INSERT INTO `pin` (`pin_number`) VALUES
```

```
('0000');
```

---

```
--  
-- Table structure for table `role`  
--
```

```
CREATE TABLE `role` (  
  `id` int(11) NOT NULL,  
  `dept` varchar(200) NOT NULL,  
  `role` varchar(255) NOT NULL,  
  `created_by` int(11) NOT NULL DEFAULT 0,  
  `created_on` int(11) NOT NULL DEFAULT 0  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Dumping data for table `role`  
--
```

```
INSERT INTO `role` (`id`, `dept`, `role`, `created_by`, `created_on`) VALUES  
(32, 'Manager', 'manager1', 0, 0),  
(33, 'Manager', 'manager2', 0, 0),  
(34, 'kitchen', 'Head Cook', 0, 0),  
(35, 'kitchen', 'Assistant Cook', 0, 0),  
(36, 'kitchen', 'Dish Washer', 0, 0),  
(37, 'cashier', 'cashier1', 0, 0),  
(38, 'waiter', 'waiter1', 0, 0),  
(39, 'waiter', 'waiter2', 0, 0),  
(41, 'cashier1', 'senior cashier', 0, 0),  
(42, 'test1', 'test role', 0, 0);
```

---

```
--  
-- Table structure for table `tables`  
--
```

```
CREATE TABLE `tables` (  
  `id` int(11) NOT NULL,  
  `restaurant_id` int(11) NOT NULL DEFAULT 0,  
  `table_name` varchar(100) NOT NULL DEFAULT "",
```

```
`short_name` varchar(40) NOT NULL DEFAULT "",  
 `table_capacity` int(11) NOT NULL,  
 `is_active` int(11) NOT NULL DEFAULT 1,  
 `status` int(11) NOT NULL DEFAULT 0,  
 `createdon` datetime NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--  
-- Dumping data for table `tables`  
--
```

```
INSERT INTO `tables`(`id`, `restaurant_id`, `table_name`, `short_name`, `table_capacity`,  
 `is_active`, `status`, `createdon`) VALUES  
(1, 1, 'Table 1', 'T1', 4, 1, 0, '2020-06-26 15:28:19'),  
(2, 1, 'Table 2', 'T2', 4, 1, 0, '2020-06-26 15:28:19'),  
(3, 1, 'Table 31', 'T3', 4, 1, 0, '2020-06-26 15:28:19');
```

```
--  
-- Table structure for table `tax_details`  
--
```

```
CREATE TABLE `tax_details` (  
 `id` int(11) NOT NULL,  
 `tax_name` varchar(10) NOT NULL,  
 `tax_percentage` int(10) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Dumping data for table `tax_details`  
--
```

```
INSERT INTO `tax_details`(`id`, `tax_name`, `tax_percentage`) VALUES  
(1, 'gst', 10),  
(2, 'gst', 10),  
(3, 'sgst', 20),  
(4, 'gst', 10),  
(5, 'sgst', 20),  
(6, 'igst', 30),  
(7, 'logesh', 10),  
(8, 'karthik', 0),
```

```
(9, 'rasna', 5),
(10, 'malavika', 6),
(11, 'malavika', 12),
(12, 'lokesh', 4);
```

---

```
--  
-- Table structure for table `users`  
--
```

```
CREATE TABLE `users` (  
  `id` int(11) NOT NULL,  
  `emp_name` varchar(255) NOT NULL,  
  `email` varchar(255) NOT NULL,  
  `phone` varchar(255) NOT NULL,  
  `dept` varchar(255) NOT NULL,  
  `role` varchar(255) NOT NULL,  
  `role_id` int(50) NOT NULL,  
  `username` varchar(255) NOT NULL,  
  `password` varchar(255) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--  
-- Dumping data for table `users`  
--
```

```
INSERT INTO `users` (`id`, `emp_name`, `email`, `phone`, `dept`, `role`, `role_id`,  
  `username`, `password`) VALUES  
(1, 'emp1', 's.logeshkarthik@gmail.com', '9551783347', 'terace', 'ENG', 27, 'sasi',  
  '81dc9bdb52d04dc20036dbd8313ed055'),  
(2, 'sasi', 's@j.c', '9876543210', 'terace', 'ENG', 27, 'lokesh',  
  '81dc9bdb52d04dc20036dbd8313ed055'),  
(3, 'Administrator', 'admin@gmail.com', '9876543211', 'terace', 'ENG\r\n', 27, 'admin',  
  '81dc9bdb52d04dc20036dbd8313ed055'),  
(4, 'vishal', 'vishal@gmail.com', '9875643210', 'kitchen', 'user', 0, 'vrkvishal',  
  '81dc9bdb52d04dc20036dbd8313ed055');
```

---

```
--  
-- Table structure for table `user_role_mapping`
```

```
--  
CREATE TABLE `user_role_mapping` (  
    `Id` int(11) NOT NULL,  
    `Role_id` int(11) DEFAULT NULL,  
    `Page_id` int(11) DEFAULT NULL,  
    `Status` int(11) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
--  
-- Dumping data for table `user_role_mapping`  
--  
  
INSERT INTO `user_role_mapping` (`Id`, `Role_id`, `Page_id`, `Status`) VALUES  
(38, 27, 11, 0),  
(39, 27, 7, 0),  
(40, 27, 1, 0),  
(41, 27, 9, 0),  
(42, 27, 3, 0),  
(43, 27, 12, 0),  
(44, 27, 8, 0),  
(45, 27, 6, 0),  
(46, 27, 13, 0),  
(47, 27, 5, 0),  
(48, 27, 15, 0),  
(49, 27, 14, 0),  
(50, 27, 4, 0),  
(51, 27, 10, 0),  
(52, 27, 2, 0),  
(53, 27, 16, 0),  
(54, 31, 6, 0),  
(55, 36, 11, 0),  
(56, 36, 1, 0),  
(57, 36, 7, 0),  
(58, 36, 12, 0),  
(59, 36, 3, 0),  
(60, 36, 9, 0),  
(61, 36, 17, 0),  
(62, 36, 16, 0),  
(63, 36, 15, 0),  
(64, 36, 6, 0),  
(65, 36, 14, 0),
```

```
(66, 36, 13, 0),
(67, 36, 8, 0),
(68, 36, 4, 0),
(69, 36, 10, 0),
(70, 36, 5, 0),
(71, 36, 2, 0),
(81, 42, 11, 0),
(82, 42, 7, 0),
(83, 42, 3, 0),
(84, 42, 12, 0),
(85, 42, 9, 0),
(86, 42, 16, 0),
(87, 42, 8, 0),
(88, 42, 6, 0),
(89, 42, 13, 0);

-- 
-- Indexes for dumped tables
--

-- 
-- Indexes for table `admin`
-- 
ALTER TABLE `admin`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `id_2`(`id`),
  ADD KEY `id`(`id`);

-- 
-- Indexes for table `category`
-- 
ALTER TABLE `category`
  ADD PRIMARY KEY (`id`);

-- 
-- Indexes for table `customer`
-- 
ALTER TABLE `customer`
  ADD PRIMARY KEY (`id`);

-- 
-- Indexes for table `department`
```

```
--  
ALTER TABLE `department`  
ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `employees`  
--  
ALTER TABLE `employees`  
ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `gst_master`  
--  
ALTER TABLE `gst_master`  
ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `items`  
--  
ALTER TABLE `items`  
ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `notification`  
--  
ALTER TABLE `notification`  
ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `ordertable`  
--  
ALTER TABLE `ordertable`  
ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `order_items`  
--  
ALTER TABLE `order_items`  
ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `pages`
```

```
--  
ALTER TABLE `pages`  
    ADD UNIQUE KEY `Id_2` (`Id`),  
    ADD KEY `Id` (`Id`);  
  
--  
-- Indexes for table `role`  
--  
ALTER TABLE `role`  
    ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `tables`  
--  
ALTER TABLE `tables`  
    ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `tax_details`  
--  
ALTER TABLE `tax_details`  
    ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `users`  
--  
ALTER TABLE `users`  
    ADD PRIMARY KEY (`id`);  
  
--  
-- Indexes for table `user_role_mapping`  
--  
ALTER TABLE `user_role_mapping`  
    ADD PRIMARY KEY (`Id`),  
    ADD UNIQUE KEY `Id_2` (`Id`),  
    ADD KEY `Id` (`Id`);  
  
--  
-- AUTO_INCREMENT for dumped tables  
--  
--
```

```
-- AUTO_INCREMENT for table `admin`
--
ALTER TABLE `admin`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;

--
-- AUTO_INCREMENT for table `category`
--
ALTER TABLE `category`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=10;

--
-- AUTO_INCREMENT for table `customer`
--
ALTER TABLE `customer`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;

--
-- AUTO_INCREMENT for table `department`
--
ALTER TABLE `department`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=27;

--
-- AUTO_INCREMENT for table `employees`
--
ALTER TABLE `employees`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=11;

--
-- AUTO_INCREMENT for table `gst_master`
--
ALTER TABLE `gst_master`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;

--
-- AUTO_INCREMENT for table `items`
--
ALTER TABLE `items`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=13;
```

```
-- AUTO_INCREMENT for table `notification`  
--  
ALTER TABLE `notification`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;  
  
--  
-- AUTO_INCREMENT for table `ordertable`  
--  
ALTER TABLE `ordertable`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=21;  
  
--  
-- AUTO_INCREMENT for table `order_items`  
--  
ALTER TABLE `order_items`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=63;  
  
--  
-- AUTO_INCREMENT for table `pages`  
--  
ALTER TABLE `pages`  
    MODIFY `Id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=18;  
  
--  
-- AUTO_INCREMENT for table `role`  
--  
ALTER TABLE `role`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=43;  
  
--  
-- AUTO_INCREMENT for table `tables`  
--  
ALTER TABLE `tables`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=17;  
  
--  
-- AUTO_INCREMENT for table `tax_details`  
--  
ALTER TABLE `tax_details`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=13;
```

```
-- AUTO_INCREMENT for table `users`  
--  
ALTER TABLE `users`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;  
  
--  
-- AUTO_INCREMENT for table `user_role_mapping`  
--  
ALTER TABLE `user_role_mapping`  
    MODIFY `Id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=90;  
COMMIT;  
  
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;  
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;  
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

## **APPENDIX D - PUBLISHED PAPER**

## Contactless Dining Based Restaurant Management Application

**Aswin Kumar G R<sup>1</sup>, Logesh Karthik S<sup>2</sup>, Amith A<sup>3</sup>, Raja S<sup>4</sup>**

<sup>1,2,3</sup>Students, Dept of Computer Science and Engineering, Panimalar Institute of Technology.

<sup>4</sup>Assistant professor, Dept of Computer Science and Engineering, Panimalar Institute of Technology.

**Abstract** - This project aims at automating the entire restaurant management process for multiple restaurants based on individual login and implementing contactless dining experience to the customers. The Graphical User Interface enables restaurant owners and staff members to view the real time data of the orders ,sales and manage them. The overall Application has various features to manage operations including various departments and also provides separate logins for each department. Various reports on day basis and monthly basis are generated and have the feature to be downloaded also. Customers can have their own login to order food and track the status of their order in real time thereby implementing contactless dining norms. This project thereby reduces man power and also provides real time data and has efficient collection of past data.

### 1. INTRODUCTION

Restaurant plays a major role in the food and beverages industry which is actually a major part of daily life of people. This project aims at automating the entire restaurant management process for multiple restaurants based on individual login and implementing contactless dining experience to the customers. The Graphical User Interface enables restaurant owners and staff members to view the real time data of the orders ,sales and manage them. The overall Application has various features to manage operations including various departments and also provides separate logins for each department. Various reports on day basis and monthly basis are generated and has the feature to be downloaded also. Customers can have their own login to order food and track the status of their order in real time thereby implementing contactless dining norms. This project thereby reduces man power and also provides real time data and has efficient collection of past data. This paper is organized as follows. In Section 2, we briefly introduce several modules present in our application and their features. In Section 3, we describe each module in detail and its use cases and activity are clearly depicted which makes the application more understandable. In Section 4, we test it using various test cases and verify its performance with various testing strategies. The final section concludes the paper and points out the future work.

### 2. MOTIVATION

The motivation of the project is, nowadays different types

of management systems are available, at the same time they are not following contactless dining norms which is the recent need. Food industry is spread worldwide and restaurants are our focus. One of the main challenges in this area is safety which is fulfilled by contactless practices. Unless we follow contactless dining practices it is considered totally unsafe and unhealthy. Keeping in mind the covid situation, the aim of having in-contact dining practice is a fear and threat to the current situation in the society. Our project takes measures to provide healthier and safe digital restaurant practices to the upcoming days of a better world. One of the solutions to deal with the problem is to build a High-performance automated operation tool that consists of a Visually understandable user interface that enables users to get used to contactless dining practices. The idea is to combine all the modules required for management of the restaurant under one roof in a highly efficient digital platform with a Visually understandable user interface and application which captures accurate data and provides Efficient reports on that data.

### 3. PROBLEM DEFINITION

The problem statement of the project is that the management tool for any restaurant is mostly semi-automatic with manual intervention in many tasks. Most modules in restaurants aren't automated yet. Contactless dining practices aren't followed as they are not in practice while in the past but now it is a necessary part of dining culture. This project aims to overcome the above with recent stable technologies in an efficient way possible.

### 4. PROPOSED SYSTEM

In the proposed system the entire restaurant management process is automated. Multiple restaurants have their own individual login. Also, contactless dining norms are followed by reducing human intervention to a great extent from the customers. The Graphical User Interface enables restaurant owners and staff members to view the real time data of the orders ,sales and manage them. Each department had separate logins for them. various departments in restaurants are covered (departments : kitchen , manager, cashier , waiter ) .Generates Daily and monthly Reports .Provides real time data and has efficient collection of past data.

## 5. FEASIBILITY STUDY

A feasibility study is carried out to select the best system that meets performance requirements. The main aim of the feasibility study activity is to determine that it would be financially and technically feasible to develop the product.

### 5.1 Technical Feasibility

This is concerned with specifying the software will successfully satisfy the user requirement. Open source and business-friendly and it is truly cross platform, easily deployed and highly extensible.

### 5.2 Economic Feasibility

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. The enhancement of the existing system doesn't incur any kind of drastic increase in the expenses. PHP is open source and ready available for all users. Since the project is runned in web enhancement and updation is easy hence is cost efficient.

## 5. SYSTEM REQUIREMENTS

### 6.1 Software Requirements

The software requirements are the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team's and tracking the team's progress throughout the development activity.

Operating system : Windows

Coding Language : PHP, JavaScript, AJAX, jQuery

Front End Tool : HTML, Bootstrap4, JS

Database : MySQL

Server : Apache

Tool : xampp

Development

environment : Visual studio code

### 6.2 Hardware Requirements

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the systems do and not how it should be implemented.

## 7. SYSTEM DESIGN

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering.

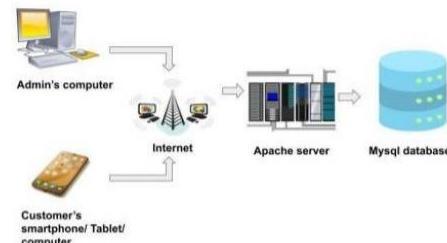


Fig -1: Architecture diagram

### 7.1 Modules Description

#### Admin Module

1. Administration module
2. Menu management
3. Employees module
4. Additional settings module
5. Reports module
6. Table and order management module

#### Customer Module

1. Menu module
2. Cart module
3. Bill module
4. Track order status module
5. Digital menu

### 7.2 Modules Definition

#### Administration module

Consists of dashboard to view the overall data such as sales , products available, orders, pending orders, total sales etc.

#### Menu management

Contains all the features to add delete or edit an item in the digital menu and also provide availability status of the item

#### Employees module

Features such as add employee provide separate login , page role mapping and others

#### Additional settings module

Consists features to reset pin , password etc.

#### Reports module

Generates and displays reports based on selection criteria

such as monthly report,  
daily report etc

#### **Table and order management module**

Has features to create table, change table availability status, create order order status and tracking and edit features

#### **Menu module**

This module enables customer to view menu and make selections to add in the cart

#### **Cart module**

This consists of a list of items based on customer selection in order to provide a over all view of the total selection to proceed to the next process

#### **Bill module**

Here comes the final step which enables customer to pay for the food

#### **Track order status module**

After making the order this module has features to track the status of the order such as pending , in progress and delivered.

#### **Digital menu**

This is the digital food menu provided by the particular restaurant which is made available for the customer to make selection and process further with the selection

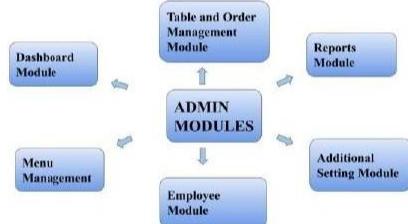
will be in an unorganized format and there may be a lot of null values, in-valid data values and unwanted data. Cleaning all these data and replacing them with appropriate or approximate data and removing null and missing data and replacing them with some fixed alternate values are the basic steps in preprocessing of data. Even data collected may contain completely garbage values. It may not be in the exact format or way that is meant to be. All such cases must be verified and replaced with alternate values to make data meaningful and useful for further processing. Data must be kept in an organised format. This is taken care and data fetching is done in such a way that eliminates the ills.

## **8. TESTING**

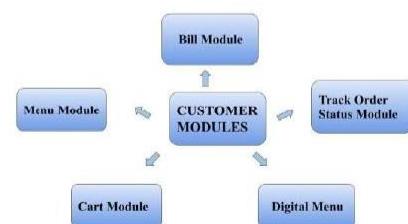
Generally testing process is carried out to ensure that the system has been developed according to the required specifications and the expected output is properly obtained. There are two main categories of testing namely, the White Box Testing and the Black Box Testing. Each of this testing in turn consists of many types of testing.

**Table -1:** Test Case Table

S.NO	Function	Description	Expected output	Actual output	Status
1.	Register	Register user details	Account registration in application	Registration accepted	success
2.	Login/ Logout	Login using registered data	login into application	login details accepted	success
3.	Make order	Making an order from menu	To place order	Order placed	success
4.	Track order	view status of the order	show status	status is viewed	success
5.	Data fetching	Data in each table is fully fetched	Do fetch all data correctly	Data fetched	success
6.	Menu Management	Menu in the restaurant	Show digital menu	Digital menu shown	success
7.	Report Table	All the collected data are organised into report	Generate report	Report generated	success



**Fig -2:** Admin Module Classification



**Fig -3:** Customer Module Classification

### **7.3 Data Processing**

Collecting the data is one task and making that data useful is another vital task. Data collected from various means

## **9. CONCLUSION**

Nowadays different types of management systems are available, at the same time they are not following contactless dining norms which is the recent need. Food industry is spread worldwide and restaurants are our focus. One of the main challenges in this area is safety which is fulfilled by contactless practices. Hereby we conclude that our project will provide a better futuristic solution for advanced management of restaurants considering current needs and efficient data processing and manipulation.

## 10. REFERENCES

- [1] Noor Azah Samsudin, Shamsul Kamal Ahmad Khalid, Mohd Fikry Akmal Mohd Kohar, Zulkifli Senin, Mohd Nor Ihkasan; "A Customizable Wireless Food Ordering System With Real-Time Customer Feedback."; 2011 IEEE Symposium on Wireless Technology and Applications (ISWTA), September 25-28, 2011, Langkawi, Malaysia
- [2] Sakari Pieskä, Markus Liuska, Juhana Jauhainen, and Antti Auno Of Centria University Of Applied Sciences Ylivieska; "Intelligent Restaurant System Smart Menu That Digital Technology"; coginfocom 2013 4th IEEE International Conference on Cognitive Infocommunications December 2-5, 2013 , Budapest, Hungary.
- [3] Ching-suchang, Che-chen Kung, Tan-hsu Tan,"Development And Implementation Of An E-Restaurant For Customer-Centric Service Using Wlan And Rfid Technology",proceedings of the Seventh International Conference On Machine Learning And Cybernetics, Kunming, 12-15 July 2008
- [4] Soon Nyean Cheong, Wei Wing Chiew, And Wen Jiun Yap; "Design And Development Of Multi-Touchable E-Restaurant Management System "; 2010 International Conference On Science And Social Research (Cssr 2010),December 5 - 7, 2010, Kuala Lumpur, Malaysia
- [5] Nilam Kadale , Pranjali Bansod, Reshma Pillai, Shivangi Sane, Snehal Pratape, Swati Pawar ; " Wireless Customizable Food Recommendation System Using Apriori And K-Means Algorithm ."; IJLTEMAS , Volume IV, Issue X, October 2015
- [6] Varsha Chavan, Priya Jadhav, Snehal Korade and Priyanka Teli of Computer Department,Pune University Indapur, Maharashtra,India; "Implementing Customizable Online Food Ordering System Using Web Based Application"; IJSET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 4, April 2015.
- [7] Jiawei Han and Micheline Kamber, "Data Mining: Concept And Techniques", University of Illinois.
- [8] M. Z. H. Noor, A. A. A. Rahman, M. F. Saaid, M. S. A. M. Ali, M. Zolkapli –The Development of Self-service Restaurant Ordering System (SROS)|| 2012 IEEE Control and System Graduate Research Colloquium (ICSGRC 2012)

## BIOGRAPHIES



Mr.G.R.Aswin Kumar is a student of CSE department in Bachelor of Engineering at Panimalar Institute of Technology. He has attended major international conferences and his area of research interest includes Artificial Intelligence and Neural Networks.



Mr.S.Logesh Karthik, is a student of CSE department in Bachelor of Engineering at Panimalar Institute of Technology. He has worked as a part of many organisations and his area of research interest includes Web development, ERP, Artificial Intelligence and Machine Learning.



Mr.A. Amith, is a student of CSE department in Bachelor of Engineering at Panimalar Institute of Technology. His area of research interest includes Digital Marketing, Entrepreneurship, Artificial Intelligence and Machine Learning.



Mr.S.Raja, is working as Assistant professor in Computer Science and Engineering department at Panimalar Institute of Technology. He received his M.Tech degree in CSE from Dr. M.G.R Educational and research institute, Chennai, India in 2013. He has 7 years of teaching experience. His area of research interest includes Data mining, Network security and Cloud computing.

## **APPENDIX E - CERTIFICATES**

# PANIMALAR INSTITUTE OF TECHNOLOGY

(A Christian Minority Institution)



(Jaisakthi Educational Trust)

Approved by AICTE, New Delhi & Affiliated to Anna University

An ISO 9001:2008 Certified Institution

Bangalore Trunk Road, Poonamallee, Chennai 600123

Accredited by National Board of Accreditation (NBA)

## Sixth International Conference on Innovative and Emerging Trends in Engineering and Technology (ICIETET '21)

### *Certificate of Participation*

This is to certify Prof./Dr./Mr./Ms./Mrs RAJA S of PANIMALAR INSTITUTE OF TECHNOLOGY has presented a paper on CONTACTLESS DINING BASED RESTAURANT MANAGEMENT APPLICATION in the Sixth International Conference on Innovative & Emerging Trends in Engineering & Technology organized by Panimalar Institute of Technology, held on 12<sup>th</sup> April, 2021.

CONFERENCE CHAIR

Dr. T. Jayanthi, Principal



SAEINDIA



# PANIMALAR INSTITUTE OF TECHNOLOGY

(A Christian Minority Institution)

(Jaisakthi Educational Trust)

Approved by AICTE, New Delhi & Affiliated to Anna University

An ISO 9001:2008 Certified Institution

Bangalore Trunk Road, Poonamallee, Chennai 600123

Accredited by National Board of Accreditation (NBA)



## Sixth International Conference on Innovative and Emerging Trends in Engineering and Technology (ICIETET '21)

### *Certificate of Participation*

This is to certify Prof./Dr./Mr./Ms./Mrs **AMITH A** of **PANIMALAR INSTITUTE OF TECHNOLOGY** has presented a paper on **CONTACTLESS DINING BASED RESTAURANT MANAGEMENT APPLICATION** in the Sixth International Conference on Innovative & Emerging Trends in Engineering & Technology organized by Panimalar Institute of Technology, held on **12<sup>th</sup> April, 2021**.

CONFERENCE CHAIR

Dr. T. Jayanthi, Principal



**SAEINDIA**



**IEEE**



# PANIMALAR INSTITUTE OF TECHNOLOGY

(A Christian Minority Institution)

(Jaisakhi Educational Trust)

Approved by AICTE, New Delhi & Affiliated to Anna University

An ISO 9001:2008 Certified Institution

Bangalore Trunk Road ,Poonamallee, Chennai 600123

Accredited by National Board of Accreditation (NBA)



## Sixth International Conference on Innovative and Emerging Trends in Engineering and Technology (ICIETET '21)

### *Certificate of Participation*

This is to certify Prof./Dr./Mr./Ms./Mrs **ASWIN KUMAR G R** of **PANIMALAR INSTITUTE OF TECHNOLOGY** has presented a paper on **CONTACTLESS DINING BASED RESTAURANT MANAGEMENT APPLICATION** in the Sixth International Conference on Innovative & Emerging Trends in Engineering & Technology organized by Panimalar Institute of Technology, held on **12<sup>th</sup> April, 2021**.

**CONFERENCE CHAIR**

**Dr. T. Jayanthi, Principal**



**SAEINDIA**



The Institution of  
Engineering and Technology

# PANIMALAR INSTITUTE OF TECHNOLOGY

(A Christian Minority Institution)

(Jaisakthi Educational Trust)

Approved by AICTE, New Delhi & Affiliated to Anna University

An ISO 9001:2008 Certified Institution

Bangalore Trunk Road ,Poonamallee, Chennai 600123

Accredited by National Board of Accreditation (NBA)



## Sixth International Conference on Innovative and Emerging Trends in Engineering and Technology (ICIETET '21)

### *Certificate of Participation*

This is to certify Prof./Dr./Mr./Ms./Mrs **LOGESH KARTHIK S** of **PANIMALAR INSTITUTE OF TECHNOLOGY** has presented a paper on **CONTACTLESS DINING BASED RESTAURANT MANAGEMENT APPLICATION** in the Sixth International Conference on Innovative & Emerging Trends in Engineering & Technology organized by Panimalar Institute of Technology, held on **12<sup>th</sup> April, 2021**.

**CONFERENCE CHAIR**

**Dr. T. Jayanthi, Principal**



**SAEINDIA**

