EFFECTIVE RESTAURANT HANDLING SYSTEM

PROJECT REPORT

Submitted to the University of Kerala in partial fulfilment of
the Degree of Bachelor of Science in Computer Science

By

Noble P Biju
(Reg.No:32020806029)

Lijo James
(Reg.No:32020806025)

Mohammed Roshan
(Reg.No:32020806027)

Abraham Mathai
(Reg.No:32020806002)

Under the supervision of

Mr. Sreejith S.R.

DEPARTMENT OF COMPUTER SCIENCE

MAR THOMA COLLEGE OF SCIENCE AND TECHNOLOGY

(Affiliated to the University of Kerala)
Chadayamangalam, Kollam (Dist), Kerala-691534

2023

DECLARATION

We do here by declare that this project entitled ***EFFECTIVE RESTAURANT HANDLING SYSTEM*** is a record of independent project work carried out by us under the supervision of the internal guide **Mr.Sreejith.S.R,** Associate Professor, Department of Computer Science, Mar ThomaCollege of Science & Technology, Ayur and external guide **Mr. DEVA KUMAR S**, Softzane Solutions, Ayur in the partial fulfilment of the award of B.Sc. Computer Science of the University of Kerala during the academic year of 2023. No part of this has previously formed the basis for the award of any Degree Diploma Associateship, Fellowship or other similar titles of this or any other University or Society.

Noble P Biju **(Reg.No:32020806029)**

Abraham Mathai **(Reg.No:32020806002)**

Lijo James **(Reg.No:32020806025)**

Mohammed Roshan **(Reg.No:32020806027)**

Place: Ayur

Date:

Mr.Sreejith.S.R

Associate Professor

Department of Computer Science

Mar Thoma College of Science and Technology,Chadayamangalam, Kollam-691534

Certificate

This is to certify that the project entitiled **EFFECTIVE RESTAURANT HANDLING SYSTEM** is an authentic report of the project done by **Noble P Biju (Reg.No:32020806029), Lijo James (Reg.No:32020806025), Mohammed Roshan (Reg.No:32020806027), Abraham Mathai (Reg.No:32020806002)** in partial fulfilment of the requirement for the award of the Degree in BSc. Computer Science of the University of the Kerala during the academic year of 2022-2023 under my supervision and guidance.

Place: Ayur

Date: 12-03-2023

Mr.Sreejith.S.R

(Internal guide)

ACKNOWLEDGEMENT

We would like to thank the almighty **God** for showering all his heavenly blessings upon us for the successful completion of our project work **EFFECTIVE RESTAURANT HANDLING SYSTEM.**

We are thankful to our Principal **Dr. Joseph Mathai**, Mar Thoma College of Science and Technology, Ayur for providing us all facilities during our course of study.

We are thankful to **Mrs. Shanu Thomas**, Head of the Department of Computer Science, Mar Thoma College of Science and Technology, Ayur for the valuable guidance and support throughout the project work.

We express our affectionate respect and heartful thanks to internal guide **Mr.Sreejith.S.R,** Associate Professor, Department of Computer Science, Mar Thoma College of Science and Technology, Ayur who had guided us with a great deal of patience.

We also express our affectionate respect and sincere thanks to external guide **Mr. Deva Kumar S**, Softzane Solutions, Ayur for his advice, support and involvement were a major source of encouragement and inspiration throughout the project.

Last but not least, we express our deepest gratitude to our beloved parents, classmates and other teachers for their constant encouragement for the completion of this project.

ABSTRACT

Online restaurant management system project in php is computerized application. The main purpose of the restaurant management system project is to reach to large range of customers and to educate them about existing and new packages and discounts offered by restaurants. And another purpose is to allow customer to pay online the bill.

Restaurant management system project documentation we explain the customer's requirement that is menu items with the help of card and mentioned placed. Sometime what happen customer need to wait outside the restaurant until free the table and wait for order. But by using restaurant management system project keeping the need of customers in mind so they can book the table online and save the time.

MODULES

Admin Module

- Hotel Handling
- Account handling
- Complaint handling

Hotel Module

- Registration
- Menu Adding
- Price listing
- Booking details checking
- Amount checking
- Purchasing
- Communication
- Feed back collection

User Module

- Hotel selection
- Items selection
- Seat selection
- Order cancellation
- Payment
- Feed back
- Communication
- Items searching

Staff

- Registration
- Order viewing
- Menu uploading
- New recipes uploading
- Images and video uploading
- Seat management
- Time booking info

CONTENTS

SL.NO	TITLE	PAGE NO
1.	**INTRODUCTION**	**1**
	1.1 About the project	2
	1.2 Module Description	3
2.	**SYSTEM STUDY AND ANALYSIS**	**5**
	2.1 Introduction	6
	2.2 Existing System	6
	2.3 Drawbacks of Existing System	6
	2.4 Proposed System	6
	2.5 Advantages of Proposed System	6
	2.6 Feasibility Study	6
	2.7 Data Flow Diagram	8
3.	**SYSTEM DESIGN**	**15**
	3.1 Input Design	17
	3.2 Output Design	18
	3.3 Database Design	19
	3.4 Table Design	22
	3.5 Language Overview	31
	3.6 Selection of Tools (S/W, H/W requirements)	37
	3.7 Menu Tree	38
4.	**SYSTEM TESTING AND IMPLEMENTATION**	**39**
	4.1 System Testing	40
	4.2 System Implementation	42

4.3 Future Enhancement 43
5. SYSTEM MAINTENANCE 44
6. SECURITY MECHANISMS 46
7. UPGRADABILITY POSSIBILITIES 48
8. CONCLUSION 50
9. APPENDIX 52
9.1 APPENDIX A-MEETING MINUTES 53
9.2 APPENDIX B-SYSTEM CODING 64
9.3 APPENDIX C-SYSTEM CODING 77
10. REFERENCES 90

1.INTRODUCTION

1.1 ABOUT THE PROJECT

Online restaurant management system project in php is computerized application. The main purpose of the restaurant management system project is to reach to large range of customers and to educate them about existing and new packages and discounts offered by restaurants. And another purpose is to allow customer to pay online the bill.

Restaurant management system project documentation we explain the customer's requirement that is menu items with the help of card and mentioned placed. Sometime what happen customer need to wait outside the restaurant until free the table and wait for order. But by using restaurant management system project keeping the need of customers in mind so they can book the table online and save the time.

If customers are feasible with requirement, then booking can be done.

Features:

1.Register/Login System
2.Manage/Create Brands
3.Easy Online Bookings
4.Manage Bookings, Pages, Subscribers
5.Detail Information on menu

In this project, User has to Login through the site for bookings. Searching can be easy. Information is provided of each and every menu with respective of hotel. For bookings, information such as Booking Dates and Feedback should beprovided by the customers. Menu information also includes Feature and Overview. Other features are: user can post their Complaint and update their Profile as well as passwords anytime they want. Staff can Add/Manage menu, manage seat management, bookings, testimonial and many more. It's easy to operate andunderstand by users. The design is pretty simple and the user won't find it difficult to understand, use and navigate.

1.2 MODULE DESCRIPTION

The modules included in this system are:

!.ADMINISTRATOR

Admin have been given access rights and are restricted up to certain functionalities, so that the data is maintained securely and redundant data is prevented

- ➢ Admin can check Hotel validity.
- ➢ Admin can check the complaints.
- ➢ Admin can view the user details
- ➢ Admin can handle Hotel.

2.USER

User can visit the website and check for various Food Menu. If User are feasible with requirement, thenbooking can be done.

- ➢ Hotel selection
- ➢ Items selection
- ➢ Seat selection
- ➢ Order cancellation
- ➢ Payment
- ➢ Feed back
- ➢ Items searching

3.HOTEL

Hotel can register with their details and can approve or reject staff registration.

- ➢ Registration
- ➢ Menu View
- ➢ Booking details checking
- ➢ Amount checking
- ➢ Feed back collection

4.STAFF

- Registration
- Order viewing
- Menu uploading
- New recipes uploading
- Seat management
- Time booking info

2.SYSTEM STUDY AND ANALYSIS

2.1 INTRODUCTION

During our project we went through the different system development life cycle. First ofall, we started with system study which helped us to understand scope of the system. During this phase, we were able to understand the possibilities of a new system. The web application EFFECTIVE RESTAURANT HANDLING SYSTEM will provide the system.

2.2 EXISTING SYSTEM

In the existing system, the services like renting a two-wheeler and other services are still offline. Request for renting is submitted manually.

2.3 DRAWBACKS OF EXISTING SYSTEM

- Wastage of time
- Database integrity
- Requires more time to access the data
- Online payment is not possible

2.4 PROPOSED SYSTEM

In the proposed system, the services become online and manages the users and dealers who register into the system. From this system both user and dealer gets the profit and it saves the time of booking and waiting.

2.5 ADVANTAGES OF PROPOSED SYSTEM

- Online payment is possible.
- User friendly
- Services are online
- Avoid wastage of time

2.6 Feasibility study

The main aim of feasibility study is to determine whether it would be functionally and technically feasible to develop the product. The feasibility study involves the analysis of the collection of relevant information relating to the product such as the different data item which would be the input to the system. The processing required to be carried out on these data, the output data required to be produced by the system, as well as various constraints on the behavior of the system. A feasibility study is rest of the system proposal according to its working, impacton the organization, ability to meet users and effective use of

resources. The objective of feasibility study is acquiring the sense of scope of the system.

The development of a computer-based system is more likely to be projects that are feasible. Three essential factors are involved in the feasibility analysis are:

- Technical feasibility
- Economic feasibility
- Operational feasibility

Technical feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

Economic feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Operational feasibility

One of the main problems faced during development of a new system is getting acceptance from user. Even if a system is technically and economically feasible but the users of the system are resistant to use it then there is no use. In this stage the following issues are considered.

- Is the proposed system is user friendly?
- Is there sufficient support for the project from the management and users?
- Will the proposed system cause harm?
- Will it have proposed poorer result in any area?
- Will loss of control result in any area?

The proposed system is so effective, user friendly and functionally reliable that the users will find it that the new system reduced their effort. Since the users are very much involved in planning and development of the project there cannot be any resistance from the management and the reactions are favorable. The result produced is accurate and optimized. The proposed system will have good control on all parts of the organization and it will take care of current activities.

2. 7 DATA FLOW DIAGRAM (DFD)

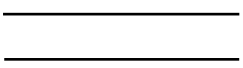
DFD are the most commonly used way of documenting the process of flow and required system. As their name suggests, they are a pictorial way of showing flow of data into, around the system. DFD was introduced by Demacro, Gane and Sarson. Data Flow Diagrams are constructedwith four major components. They are:

Data Flow Diagram Symbols

1. Entities

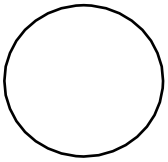
External entities represent the sources of data that enter the system or recipients of data that leave the system.

2. Data Store



Data Store is represented by using two parallel lines. It represents a logical files.

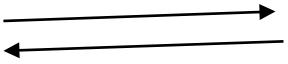
3. Process



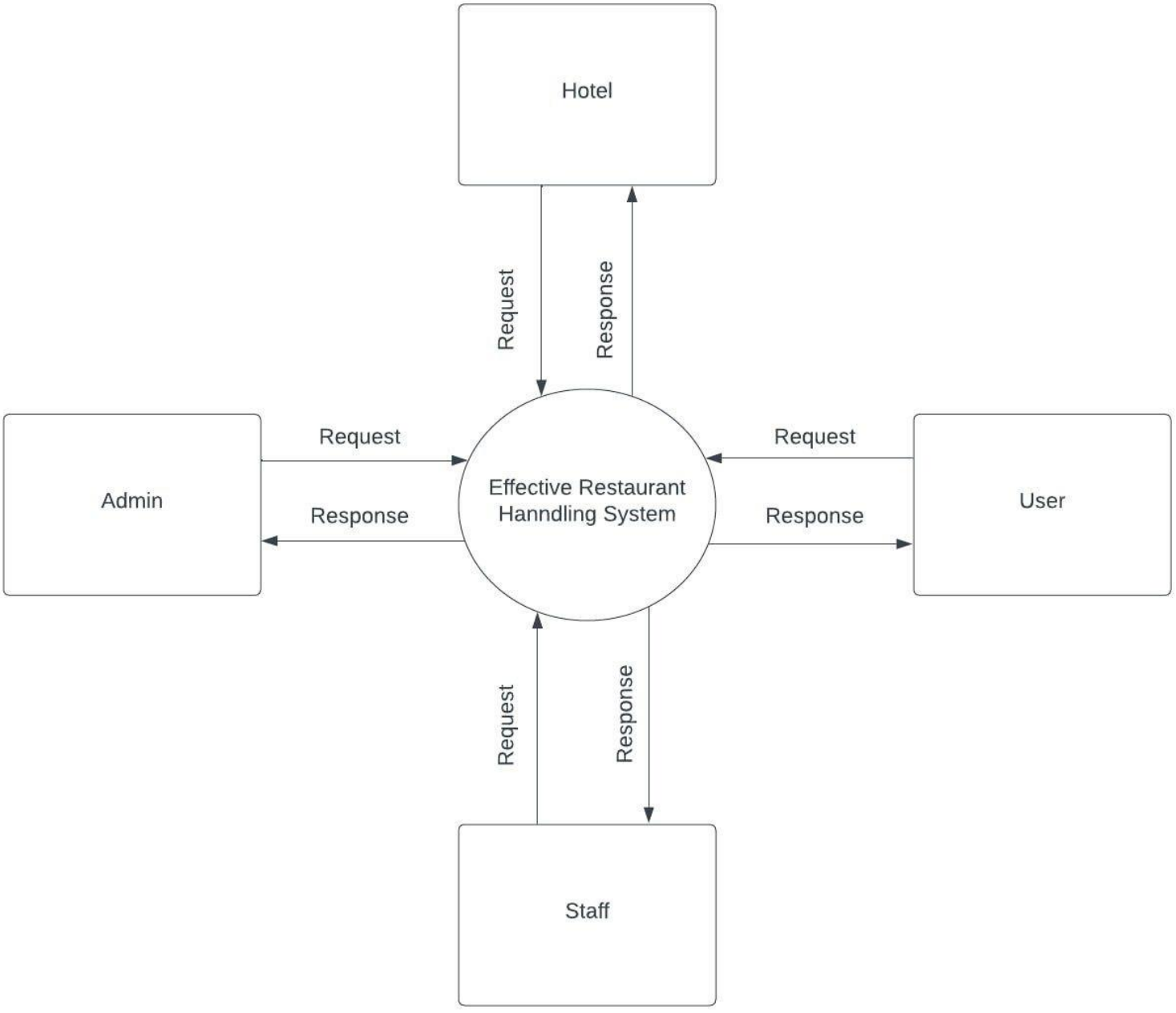
Process represent active in which data is manipulated by being stored or retrieved and transformed in some way. A circle represents it.

4. Data Flows

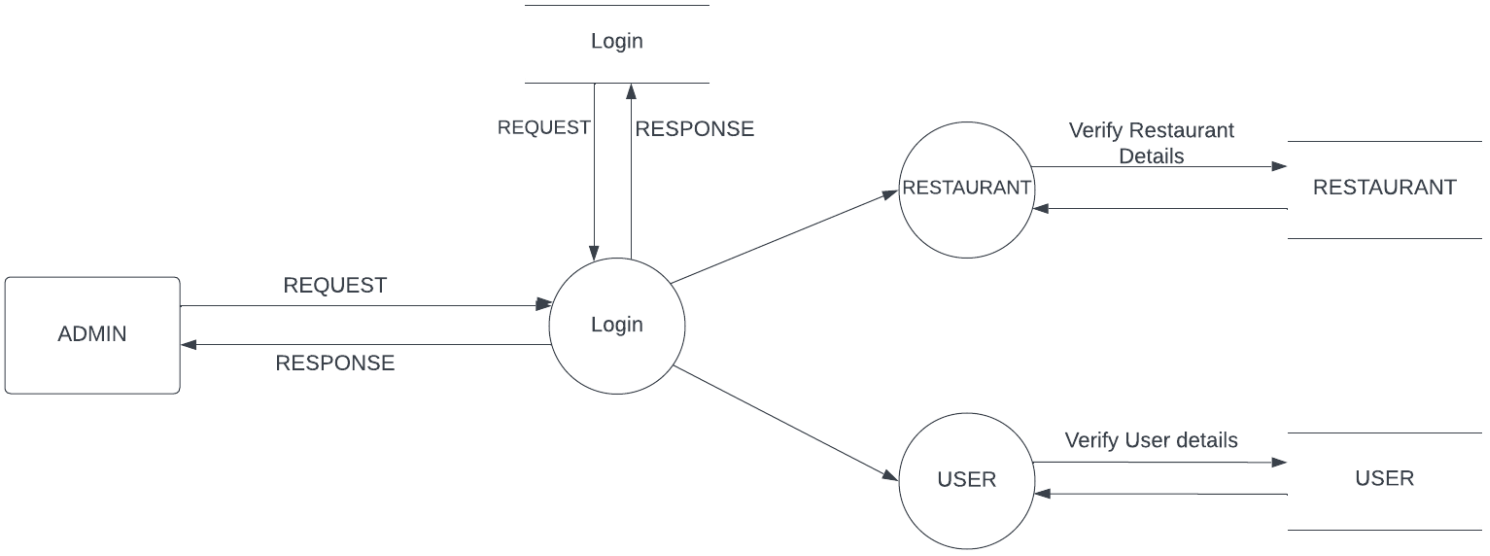
Data flow show the flow of information from its source to its destination's line represents data flow, with arrowheads showing the direction of flow.



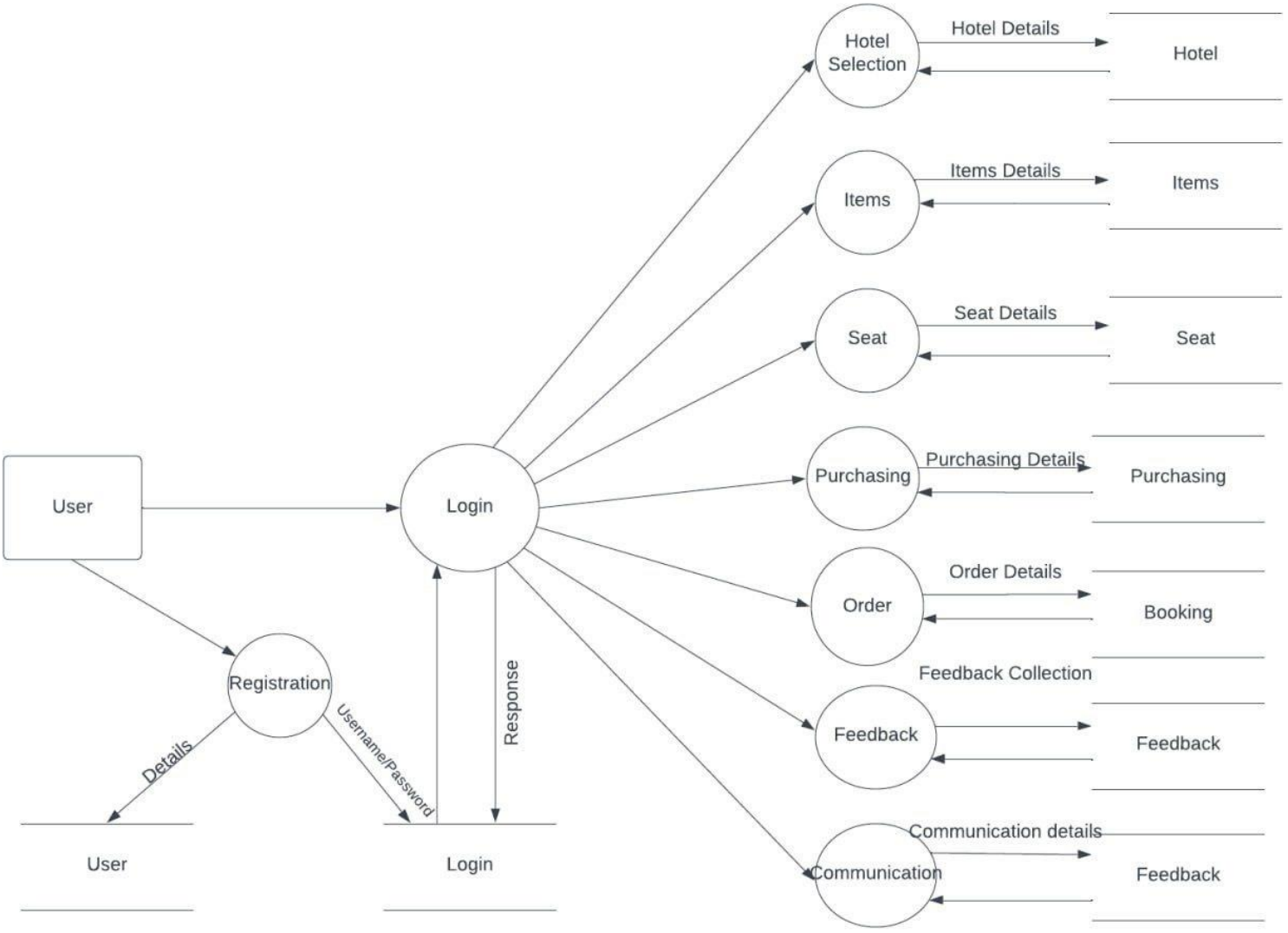
LEVEL 0: EFFECTIVE RESTAURANT HANDLING SYSTEM



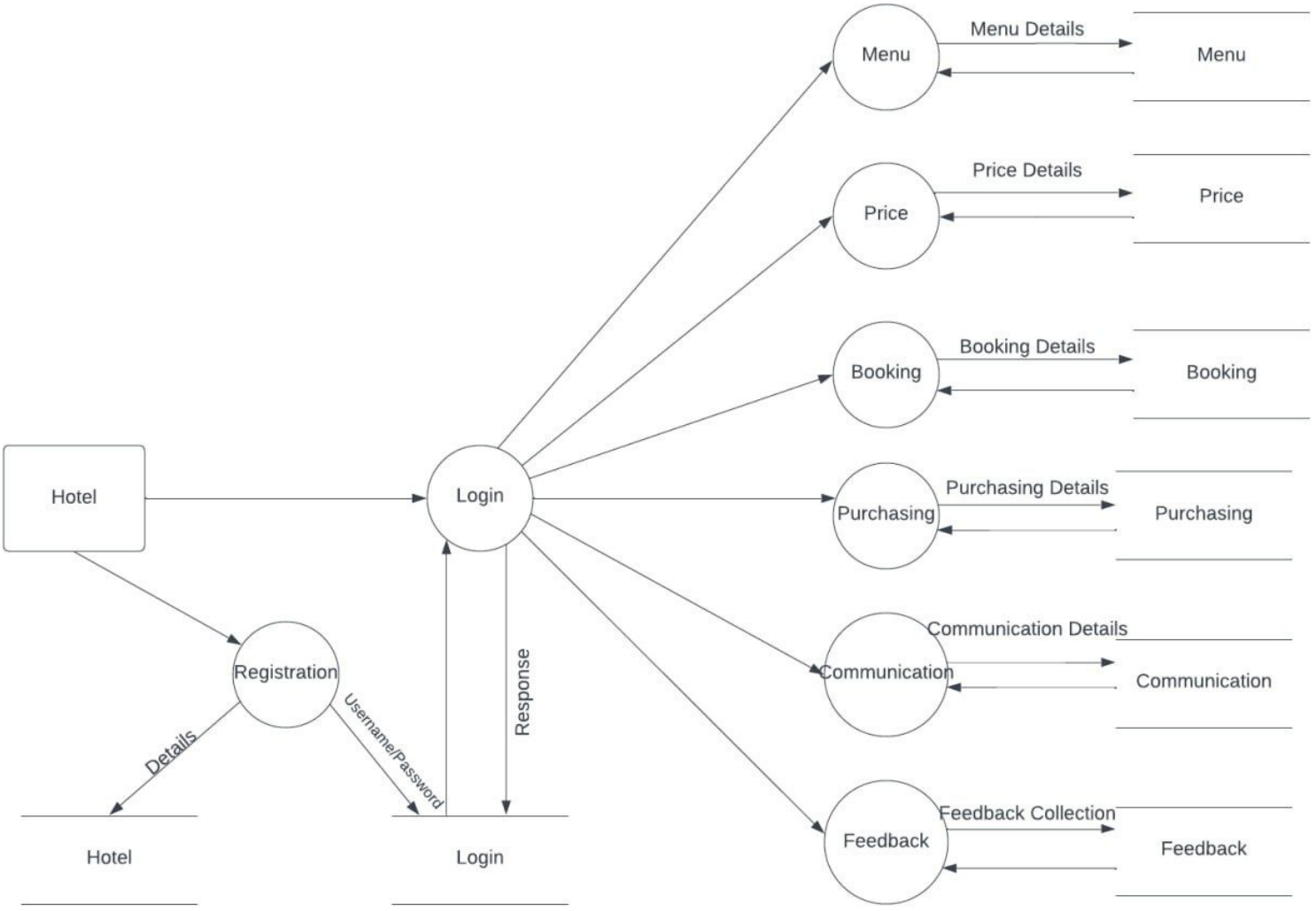
LEVEL 1: ADMIN



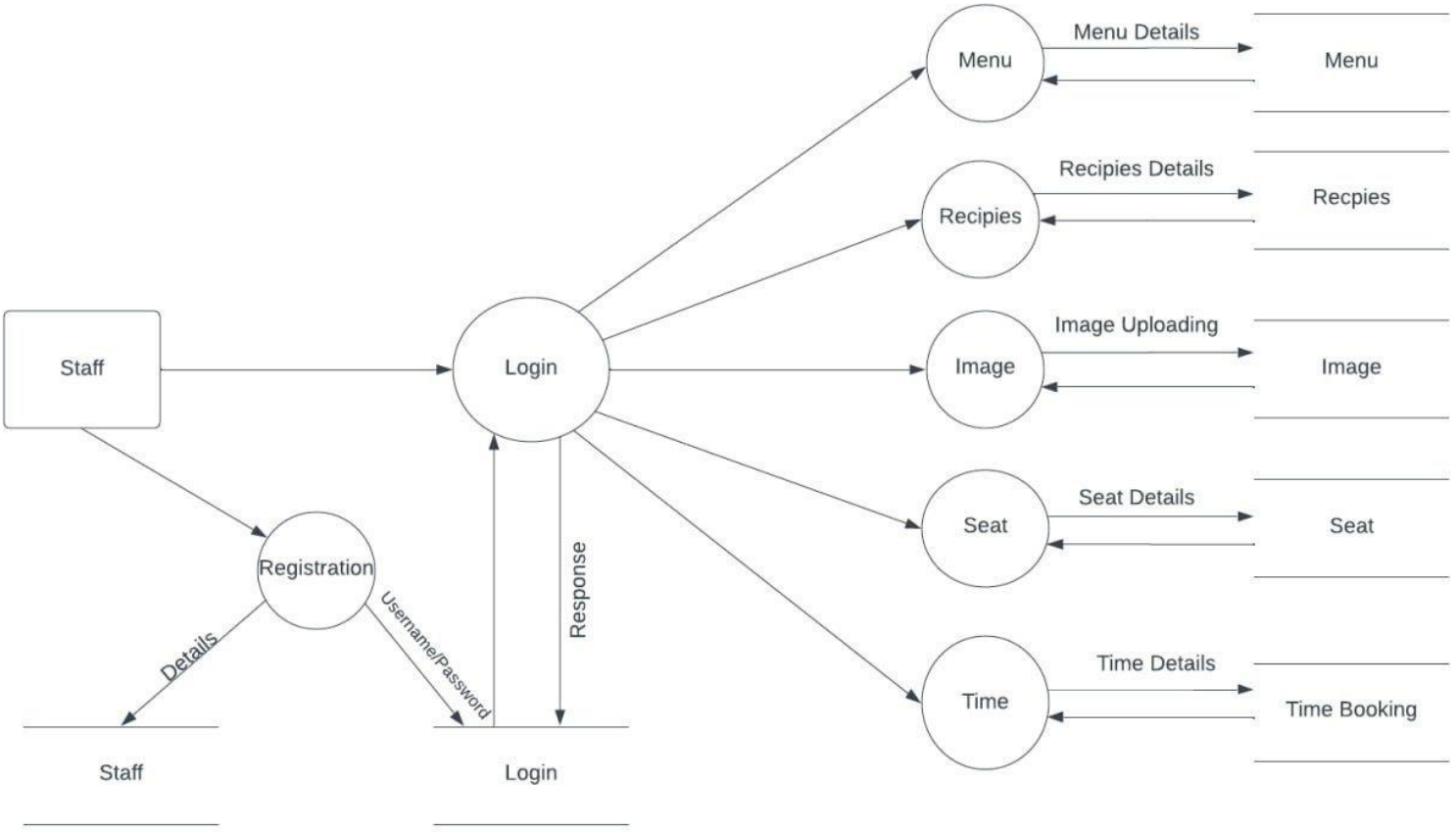
LEVEL 1: USER


User
Login
Registration
Details
Username/Password
Response
User
Login
Hotel Selection
Hotel Details
Hotel
Items
Items Details
Items
Seat
Seat Details
Seat
Purchasing
Purchasing Details
Purchasing
Order
Order Details
Booking
Feedback Collection
Feedback
Feedback
Communication
Communication details
Feedback

LEVEL 1: HOTEL



LEVEL 1: STAFF



3.SYSTEM DESIGN

System design is the first step in the development phase for many engineered product or system. It may be defined as the process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permits its physical realization. This Phase is the first step in moving from the problem domain to the solution domain. It is an iterative Process through which requirements are transmitted into a "blueprint" for constructing the software initially; the blue depicts a holistic view of software. That is design is represented at a high level of abstraction, functional and behavioral requirements.

System design develops the architectural detail required to build a system or product. The system design process encompasses the following activities.

- Partition the analysis model into subsystem.
- Identify the concurrency that is dictated by the problem.
- Allocate the subsystems to processors and tasks.
- Develop a design for the user interface.
- Choose a basic strategy for implementing data management.
- Identify global resources and control mechanisms required to access them.
- Design an appropriate control mechanism for the system including task management

System design provides an understanding of the procedural details, necessary for implementing the system recommended in the feasibility study. Basically, it is all about thecreation of the new system. This is critical phase since it decides the quality of the system. It has amajor impact on the testing and implementation phase.

System design is the most creative and challenging phase of the system life cycle. The term design describes the final system and the process by which is to be developed. During the system design phase, the designers must design how to produce an efficient and effective system. There are two levels of system design: Logical design and physical design.

In the logical design, the designer produces a specification of the major features of the system which meets the objectives. The delivered product of logical design includes current requirements of the system components: Input design, Output design and Database design.

Physical design takes this logical design blueprint and produces the program specifications. Design specifications instruct programmers about what the system should do.

Structured design is data flow-based methodology that partitions a program into a hierarchy of modules organized top-down manner with details.

Project Lifecycle:

Description:

The waterfall Model is a linear sequential flow. In which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation.

This means that any phase in the development process begins only if the previous phase is complete. The waterfall approach does not define the process to go back to the previous phase to handle changes in requirement. The waterfall approach is the earliest approach that was used for software development.

Problem Statement:

A bike rental is a rented vehicle that can be used temporarily for a fee during a specified period. Getting a rental bike helps people get around despite the fact they do not have access to their own personal vehicle or don't own a vehicle at all. The individual who needs a bike must contact a rental bike company and contract out for a vehicle. This system increases customer retention and simplify vehicle and staff management.

Objectives:

Specific Objectives are

- To produce a web-based system that allow customer to register and reserve bike online and for the company to effectively manage their bike rental business.
- To ease customer’s task whenever they need to rent a bike.
- Availability of vehicle round the clock.
- Wide range of vehicles.
- Maintaining record of each booking history.
- Online Payment option.
- Login and Account generation.

3.1 INPUT DESIGN

The input design is the process of converting the user-oriented inputs into thecomputer-based format. The goal of designing input data is to make automation as easy and free from errors as

possible. The input design requirements such as user friendliness, consistent format and interactive dialogues for giving the right message and help for the user at right time are also considered for the development of the project.

Inaccurate input data is the most common cause of error in processing data. Errors entered by the data entry operators can be controlled by the input design. The arrangement of messages as well as placement of data, headings and titles on display screens or source document is also a part of input design. The design of input also includes specifying the means by which enduser and system operators direct the system what action to take. The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps that are necessary to put transaction data into a usable form for processing data entry.

In this project, all the necessary text boxes are validated. The input forms are designed in Sublime. If any non-empty fields are not filled, it will display an error message and will wait until user types the necessary and correct input. Initially, to access the services of this software, the user has to log on with a login name and password which are validated. Once loggedon, he can access the various services, navigate to different profiles.

3.2 OUTPUT DESIGN

Output generally refers to the results and information that are generated by the system. When designing output, system analyst must accomplish the following.

- Determine what information to present.
- Decide whether to display, print the information and select the output medium
- Arrange the presentation of information in an acceptable format.
- Decide how to distribute the output to intended recipients.

The output design is specified on layout forms, sheets that describe the location characteristics, and formats of the column heading and pagination. In this project, output formsare designed in PHP. Each form has a heading or caption which specifies what services is been given to the users making the software user-friendly. All information is stored in the databaseand when anyone logs on and request for a service, the corresponding page is fetched from the server after validation and is rendered.

3. 3 DATA BASE DESIGN

A relational database is a collection of data items organized as a set of formally described tables from which data can be accessed or reassembled in many different ways without having to recognize database tables. The RDB was invented by E.F Codd at IBM in 1970.

An RDBMS is a program that lets you create, update, and administer a relational database. Most commercial RDBMS use the SQL to access the data base, although SQL was invented after the development of relational model and is not necessary for its use. A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored data in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two-level process. In the first step user requirements are gathered together and a database is designed which will meet these requirements as cleanly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step this information level is transferred into a design for specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concernedwith the characteristics of the specific DBMS that will be used. A DB design runs parallel with the system design. The organization of data in the database is aimed to achieve two major objectives.They are:

- Data independence
- Data integrity

The data base design is made up of three levels

1) Conceptual level (High Level)
2) Physical level (Low Level)

3) View level (Representation level)

3.3. 1 DATA NORMALIZATION

Normalization is the process of efficiently organizing data in a database. Two goals or normalization are: eliminate redundant data and ensure data dependencies make sense. Both these goals reduce the amount of space a database consumes and ensure that data is logically stored. The database commonly has developed a series of guidelines for ensuring that databases are normalized. These are referred to as normal forms and are numbered from one through five (1NF to 5NF).

Data in First Normal Form

A relation r(R) is said to be in First Normal Form (1NF) if and only if every entry of the relation (the intersection of a tuple and a column) has at most a single value. Some authors prefer to say that a relation is in 1NF if and only if all its attributes are based upon a simple domain. These two definitions are equivalent. If all relations of a database are in 1NF we will say that the database is in 1NF

Data in Second Normal Form

A relation r(R) is in Second Normal Form (2NF) if and only if the following two conditions are met simultaneously:

(1) r(R) is already in 1NF.

(2) No nonprime attribute is partially dependent on any key or, equivalently, each nonprime attribute in R is fully dependent upon every key (including candidate keys).

Data in Third Normal Form

A relation r(R) is in Third Normal Form (3NF) if and only if the following conditions are satisfied simultaneously:

(1) r(R) is already in 2NF

(2) No nonprime attribute is transitively depende

(3) nt on the key.

Advantages of Normalization

- Helps in reduction in the complexity of maintaining data integrity by removing the redundant data.
- Reduces inconsistency of data.
- Eliminate repeating fields.
- Create a row for each occurrence of a repeated field. Allows exploitation of column functions.

3.4 TABLE DESIGN

1. login

Description: Used to store username and password of users

Column	Datatype	Size	Constraints	Description
loginid	int	10	Primary Key	Store Login Id
email	varchar	40	Not Null	Store Email id
password	varchar	32	Not Null	Store Password
usertype	enum	‘0’,’1’,’2’,’3’	Not Null	Store User Type
Status	enum	‘0’,’1’,’2’	Not Null	Store Login Status
lkey	char	8	unique	Store login key

2. hotel

Description: Used to store Hotel details

Column	Datatype	Size	Constraints	Description
id	int	11	Primary Key	Store Hotel Id
name	varchar	50	Not Null	Store Hotel Name
address	varchar	100	Not Null	Store Hotel Address
pincode	int	11	Not Null	Store hotel Pincode
district	varchar	50	Not Null	Store hotel District
category	varchar	12	Not Null	Store Hotel category
city	varchar	20	Not Null	Store Hotel City
image	file		Not Null	Store Hotel image
contactno	int	13	Not Null	Store Hotel Contact
seat	varchar	10	Not Null	Store total no of seat
seatavilable	varchar	10	Not Null	Store total no of seat avilable
loginid	int	10	Foreign Key	Store Hotel Login Id

3. menu

Description: Used to store Menu details

Column	Datatype	Size	Constraints	Description
id	int	10	Primary Key	Store Vehicle Id
itemcategory	varchar	15	Not Null	Store Item Category
vegornon	varchar	10	Not Null	Store Veg or Nonveg
item	varchar	20	Not Null	Store Item name
itemimage	file		Not Null	Store Image of the item
otherdetails	varchar	50	Not Null	Store Other details of the item
currentdate	date		Not Null	Store the Date
available	enum('0','1')		Not Null	Store Availability
amount	varchar	30	Not Null	Store the Amount
loginid	int	11	Foreign Key	Store Pensioner Login id

4. menubooking

Description: Used to store booking details

Column	Datatype	Size	Constraints	Description
id	int	10	Primary Key	Store Booking Id
menuid	int	10	Not Null	Store the Menu id
currentdate	date		Not Null	Store Current Date
quantity	varchar	50	Not Null	Store Quantity
totalamount	varchar	30	Not Null	Store Total Amount
noofseat	varchar	30	Not Null	Store No of seat
loginid	int	10	Not Null	Store Login Id
bookingdate	date		Not Null	Store Bookingdate
cstatus	enum	'0','1'	Not Null	Store Cancel status
pstatus	enum	'0','1'	Not Null	Store payment approved status
rstatus	enum	'0','1'	Not Null	Store Refund status
fstatus	enum	'0','1'	Not Null	Store Finished status

5. payment

Description: Used to store payment details

Column	Datatype	Size	Constraints	Description
id	int	10	Primary Key	Store complaint Id
currentdate	date		Not Null	Store Current Date
loginid	int	10	Not Null	Store login id
pkey	char	8	Foreign Key	Store payment key
nameoncard	varchar	30	Not Null	Store Name on card
cardnumber	int	50	Not null	Store Card number
cvv	int	3	Not null	Store Card Cvv
expmonth	int	2	Not null	Store Expiry month
expyear	int	4	Not null	Store Expiry Year
amount	int	40	Not null	Store Amount

6 *feedback*

Description: Used to store feedback details

Column	Datatype	Size	Constraints	Description
id	int	10	Primary Key	Store feedback Id
fkey	char	8	Not Null	Store fkey
feedback	varchar	200	Not Null	Store feedback description
loginid	int	10	Not null	Store feedback login id
vid	int	10	Not null	Store vid
currentdate	date		Not null	Store current date

7.*staff*

Description: Used to store staff details

Column	Datatype	Size	Constraints	Description
id	int	10	Primary Key	Store Staff Id
skey	char	8	Not Null	Store skey
name	varchar	30	Not Null	Store name
address	varchar	70	Not Null	Store address
pincode	int	6	Not Null	Store pincode
contact	bigint	50	Not Null	Store contact
restaurant	varchar	30	Not null	Store restaurant
district	varchar	20	Not null	Store district
city	varchar	15	Not null	Store city
designation	varchar	20	Not null	Store Designation
loginid	int	10	Not null	Store login id

8. bank

Description: Used to store bank details

Column	Datatype	Size	Constraints	Description
id	int	11	Primary Key	Store Bank Id
bkey	char	8	Not Null	Store bkey
bankname	varchar	30	Not Null	Store bank name
branchname	varchar	50	Not Null	Store branch name
ifsc	varchar	30	Not Null	Store ifsc code
accountholdername	varchar	50	Not Null	Store account holder name
accountnumber	int	30	Not Null	Store account number
cardno	int	30	Not Null	Store cardno
cvv	int	3	Not Null	Store cvv
expmonth	date	0	Not Null	Store expiry date
expyear	date	0	Not Null	Store expiry year
contact	varchar	15	Not null	Store contact details
totalamount	varchar	15	Not null	Store total amount

9.*user*

Description: Used to store User Details

Column	Datatype	Size	Constraints	Description
id	int	10	Primary Key	Store Id
ukey	char	8	Not Null	Store ukey
username	varchar	50	Not Null	Store Username
loginid	int	10	Not Null	Store login id
contact	varchar	15	Not Null	Store Contact of User

10.*hotelfeedback*

Description: Used to store Feedback about hotel

Column	Datatype	Size	Constraints	Description
id	int	10	Primary Key	Store Id
fkey	char	8	Not Null	Store fkey
feedback	varchar	50	Not Null	Store feedback details
loginid	int	10	Not Null	Store login id
current date	date	0	Not Null	Store current date

3.5 LANGUAGE OVERVIEW

PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1995. The reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Pre-processor, a recursive acronym.

PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License, which is incompatible with the GNU GeneralPublic License (GPL) due to restrictions on the usage of the term PHP. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform. free of charge.

The PHP interpreter only executes PHP code within its delimiters. Anything outside its delimiters is not processed by PHP. The most common delimiters are <?php to open and"> to close PHP sections. <script language="php"> and </script> delimiters are also available, as are the shortened forms <? or <? = and?> as well as ASP-style short forms <% or <%= and %>. WW1e short delimiters are used, they make script files less portable as support for them can be disabled in the PHP configuration, and so they are discouraged. The purpose of all these delimiters is to separatePHP code from non-PHP code, including HTML.

Much of its syntax is borrowed from C. Java and Perl with some unique features thrown in. The goal of the language is to allow Web developers to write dynamically generated pages quickly.

Advantages of PHP

- Cost is low
- PHP is an open-source software
- PHP is easy to learn
- PHP is embedded within HTML

The HTML-embedding of PHP has many helpful consequences.

- PHP can quickly be added to code produced by WYSIWYG editors.
- PHP lends itself to a division of labour between designers and scripters.
- Every line of HTML does not need to be rewritten in a programming language.
- PHP can reduce labour costs and increase efficiency due to its shallow learning curve and ease of use.
- PHP has Cross-platform compatibility
- PHP is not tag-based
- PHP is much faster for almost every use than CGI scripts.
- PHP makes it easy to communicate with other programs and protocols.
- PHP is fast becoming one of the most popular choices for so-called two-tier development.
- PHP is developed and supported in a collaborative fashion by a worldwide community of users.

Hyper Text Transfer Protocol (HTTP)

HTTP is the protocol "spoken" by web servers. Client programs that can speak I-ITT P. known as browsers, are used by the people on the Internet to connect to HTTP servers. The servers provide access to distributed hyper linked documents, applications and databases. HTTP is a stateless. object oriented application-level protocol that has been in the existence since the early days of the WWW. NSCA HTTP is a HTTP/1.0 compliant web server and is credited with being one of the first HTTP servers available. It supports multiple schemes of authentication.

Html-The Frame Work For Webpages

Hypertext Mark-up Language (HTML) is the text mark-up language on the World Wide Web. The mark-up commands applied to the web-based content tell the browser software the structure of document and, when, how we want the content to be displayed. It has a well-defined syntax and HTML documents have a formal structure. With the introduction of scripting languages such as JavaScript, the concept of dynamic HTML (DHTML) is becoming more and more popular and is used to create highly interactive web pages. When browser reads a document that has html markup in it, it determines how to render it on screen by considering the html elements embedded within the document.

CSS

- CSS stands for Cascading Style Sheets
- Styles define how to display HTML elements
- Styles were added to HTML 5. 0 to solve a problem
- External Style Sheets can save a lot of work External Style Sheets are stored in CSS
- All browsers support CSS today.

SUBLIME TEXT 3

Sublime text is a proprietary cross-platform source code editor with a PythonApplication Programming Interface (API). It natively supports many programming languages and markup languages, and functions can be added by users with plugins, typically community-built and maintained under free-software licenses. Version 3 entered beta on 29 January 2013. At first available only for registered users who had purchased Sublime Text 2, on 28June 2013 it became available to the general public. However, the very latest development builds still required a registration code. Sublime Text 3 was officially released on 13 September 2017.

Two of the main features that Sublime Text 3 adds include symbol indexing and pane management. Symbol Indexing allows Sublime Text to scan files and build an index to facilitatethe features Goto Definition and Got Symbol in Project. Pane Management allows users to move between panesvia hotkeys.

BOOTSTRAP

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 a web framework that focuses on simplifying the development of informative web pages (as opposed to web apps). 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 textwith a highlight.

SMARTY

Smarty is a web template system written in PHP. Smarty is primarily promoted as a tool for separation of concerns. Smarty is intended to simplify compartmentalization, allowing the presentation of a web page to change separately from the back-end. Ideally, this eases the costs and efforts associated with software maintenance.

Smarty is a template engine for PHP. More specifically, it facilitates a manageable way to separate application logic and content from its presentation. This is best described in a situation where the application programmer and the template designer play different roles, or in most cases is not the same person.

ADVANTAGES

➢ It is extremely fast.

➢ It is efficient since the PHP parser does the dirty work.

➢ No template parsing overhead, only compiles once.

➢ It is smart about recompiling only the template files that have changed.

➢ You can make custom functions and custom variable modifiers, so the template language is extremely extensible.

➢ Configurable template delimiter tag syntax, so you can use {}, {{}}, <!--{}-->, etc.

➢ The if/else if/else/end if constructs are passed to the PHP parser, so the {if ...} expression syntax can be as simple or as complex as you like.

➢ Unlimited nesting of sections, ifs, etc. allowed.

DATABASE

A database is a separate application that stores a collection of data. Each database has one or more distinct AP is for creating, accessing, managing, searching, and replicating the data it holds. Other kinds of data stores can be used, such as files on the file system or large hash tablesin memory. A database system must provide following features

- A variety of user interfaces.
- Physical data independence.
- Logical data independence.
- Query optimization.
- Data integrity.
- Concurrency control.
- Backup and recovery.
- Security and authorization.

MYSQL

MySQL is an open-source Relational Database Management System that uses Structured Query Language. Information is stored in "Tables" which can be thought of as the equivalent of Excel spreadsheets. A single MySQL database can contain many tables at once and store thousands of individual records. It's fast, reliable and flexible. We can copy MySQL to our PCs and Laptops. MySQL4 version has introduced some innovation and changes both on the database side and PHP side. Earlier versions of the MYSQL lacked some core SQL constructs such as sub selects and foreign keys. But because of simple licensing nature it become popular among users. MYSQL tables are of type called MyISAM. The new version of the MYSQL created new types oftransaction safe table InnoDB and BDB. They impose little over head and slower in action. MySQL allows you to grant quite fine-grained permission to different users from different client locations. There are four descending levels of privileges global database, table and column.

The features of MySQL server are:

➢ A very fast thread-based memory allocation system.

➢ fast joins using an optimized one-sweep multi-join.

➢ In-memory hash tables, which are used as temporary tables.

➢ SQL functions are implemented using a highly optimized class library and should be as fast as possible. Usually there is no memory allocation at all after query initialization.

➢ The best and the most-used database in the world for online applications.

➢ Available and affordable for all.

➢ Easy to use.

➢ Continuously improved while remaining fast, secure and reliable.

➢ Fun to use and improve.

➢ Free from bugs.

jQuery

jQuery is a library of JavaScript Functions. jQuery is a lightweight "write less, do more" JavaScript library. The jQuery library is stored as a single JavaScript file, containing all the jQuery methods.

The jQuery library contains the following features:

➢ HTML element selections

➢ HTML element manipulation

➢ CSS manipulation

- ➢ HTML event functions
- ➢ JavaScript Effects and animations
- ➢ HTML DOM traversal and modification
- ➢ AJAX
- ➢ Utilities.

3.6 SELECTION OF TOOLS (S/W, H/W REQUIREMENTS)

Minimum Hardware Requirements

Processor	:	Intel(R) CPU
RAM	:	4GB
Hard Disk	:	500GB
CD-ROM	:	700 MB
Keyboard	:	Standard 101/102 key
Mouse	:	Optical mouse
Monitor	:	Plug and Play monitor
Printer	:	Ink jet

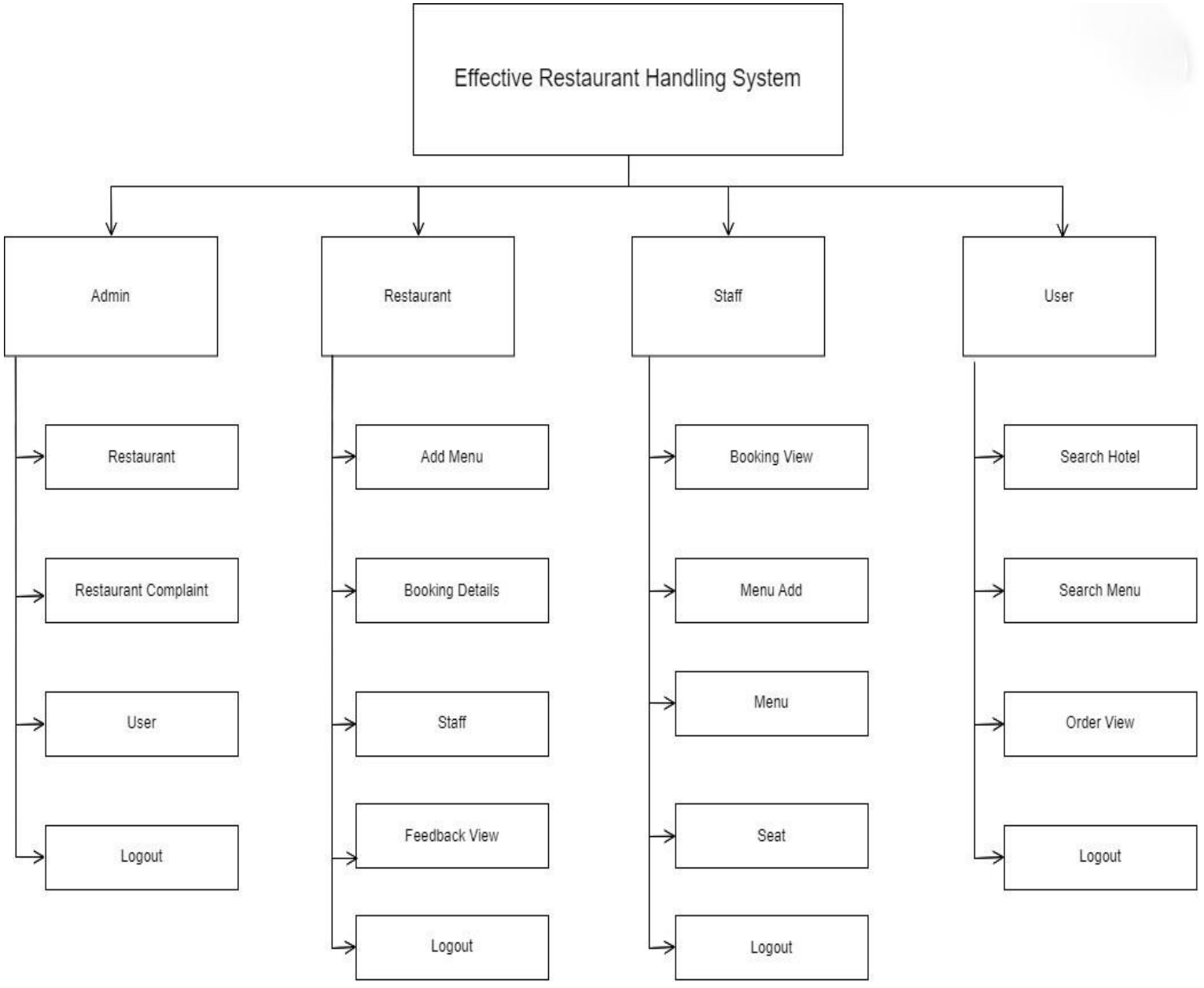
Software Specification

Operating System	:	Windows 10/11
Front End	:	PHP
Back End	:	MySQL
Browser	:	Google Chrome

3.4 MENU TREE

Menu Tree:

A menu tree is a menu type that follows a hierarchical structure of data presentation. It enables you to categorize data items based on the kind of tasks, products, functions etc. In the apps you create with a low-code platform, you can add, modify and customize these categories and subcategories by editing records and pages.



3.SYSTEM TESTING AND IMPLEMENTATION

4.1 SYSTEM TESTING

Testing is the process of executing a program with the intent of finding any errors. A good test of course has the high probability of finding a yet undiscovered error. A successful testing is the one that uncovers a yet undiscovered error. A test is vital to the success of the system. System that makes logical assumptions that if all parts of this system are correct, then foal will be successfully achieved. The candidate system is subjected to variety of tests online like responsiveness, its value, stress and security. System testing can be broadly classified into:

- Black box testing
- White box testing
- Unit testing
- Integration testing
- Validation testing

Black Box Testing

When computer software is considered b1ack box testing alludes to tests that are conducted at the software interface. A black box test examines some fundamental aspects of a system with little regard for the internal logical structure of the software. Black box testing attempts to find errors in the following categories:

- Incorrect or missing function
- Interface errors
- Performance errors
- Errors in data structures or external database access
- Initialization and termination errors

In our application, we use a number of functions to perform operations. Using the black box testing we made sure that all functions are executing correctly by giving the required result.

White Box Testing

It is a testing method that uses control structure of procedural design to derive testing. Knowing the internal working of a product tests can be conducted to ensure that the internal operations perform according to specification and all internal components have been adequately exercised. White box testing of software is predicated on close examination of conditions and/or loops test logical paths through the software. Using this testing method, the software engineer can do tests that:

Guarantee that all independent paths within a module have been exercised at least Once.

- Exercise all logical decisions on their true and false values.
- Execute all loops at their boundaries and within their operational bounds.
- Exercise internal data structures to ensure their validity.

Here all logical structures are tested in their true and false conditions. We also made sure that all loops are performing well at their boundaries. For the checking appropriate data inputs are given and they are processed correctly. Individual functions are tested separately for each of the above conditions.

Unit Testing

This is the first level of testing. Here different functions used in the software development are split into different modules and tested to see whether they satisfy our needs. Code produced during the coding phase of the software development process and the internal logic of the module is tested here. After coding each function was tested individually. The logical errors found were corrected.

Integration Testing

This is systematic technique for constructing the structure while conducting tests touncover errors with interfacing. Here the different functions of software are combined into sub system, which are again tested. The various unit tested functions of the software were integrated and rigorous integration testing was conducted to make the application free of any interface errors that may occur. In this phase various functions are combined. Once the individual functions were tested,we tested the control hierarchy in a top-down integration manner.

Validation Testing

It provides the final assurance that the software meets all functional, behavioral and performance requirements. Then software changed for the better performance. When the application was made free of all logical and interface errors, validation testing was conducted by inputting dummy data to ensure that the software developed satisfied all the requirements of the user. This includes providing various valid and invalid inputs.

System tests carried out to validate dully developed system with a view assuring that it meets its requirements. There are essentially three kinds of system testing:

1. Alpha Testing

It refers to the system testing that is carried out by the test team within the organization.

2. Beta Testing

Beta testing is the system testing performed by a selected group of friendly customers.

3. Acceptance Testing

Acceptance testing is the system testing performed by the customer to determine whether or not to accept the delivery of the system. The application is tested to ensure the requirements. Different sets of input data are entered to validate the system. In all cases the system produces the reasonable output.

4.2 SYSTEM IMPLEMENTATION

Implementation is the process of converting a new or revised system design into operation. It is the key stage in achieving a successful new system because, usually it reveals a lot of up heal. It must therefore be carefully planned and controlled. Apart from planning the two major tasks of preparing for implementation are education and training of users and testing of the system. Education of users should really take place much earlier in the project, Training has to be given to the web masters regarding the new system. Implementation is the stage of project where the theoretical design is turned into working system or it is the key stage in achieving a successful new system. Therefore, it must be carefully planned and controlled. It can also be considered to be the most crucial stage in achieving a successful new system and in giving the user confidence that the new system and in giving the user confidence that the new system will work and be effective.

Implementation is the final and important phase. It is the phase where theoretical design is turned into working system, which works for the user in the most effective manner. It involves careful planning, investigation of the present system and the constraints involved, user training, system testing and successful running of developed proposed system. The implementation process begins with preparing a plan for the implementation of the system. According to this plan the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system, The user tests the developed system and changes are made according to their needs. The testing phase involves the testing of a system using various kinds of data. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

4.3 3 FUTURE ENHANCEMENTS

Enhancement means adding, modifying or developing the code to support the changes in the specification. It is the process of adding new capabilities such as report, new interface without other systems and new features such as better screen or report layout. Every module in the system is being developed carefully stich that the future enhancements do not affect the basic performance of the system. In future we can add any links or services to the System very easily. Moreover, due to limited time allotted for the project, there are features, which we couldn't implement. Thus, the system offers the scope of future enhancement. As this software is reliable touse, any modification in accordance with the necessity of the user can be done for the future use. Any additional feature can be implemented very easily. So, what we call this software also a user friendly. Some of the future developments that can be incorporated in this software are

- In current system tender verification process is semi computerized we can implement it total computerized.
- This application is implementing all over the world.

5. SYSTEM MAINTENANCE

5. SYSTEM MAINTENANCE

It is possible to produce systems of any size which do not need to be changed. Over the lifetime of a system, its original requirements will be modified to reflect the changing user.

After implementation, maintenance is the important process. Usually once the system is implemented, the software developers and customer would sign a contract. According to the time mentioned in the contract all errors and requirements would be charged. During the contract period we would frequently visit the site where the system is implemented and check the system performance such as response time and also how it works at peak hours. If any problem is found, it is corrected.

The four types of maintenance activities are listed below.

Corrective Maintenance

This is concerned with fixing reported errors in the software. Coding errors are relatively cheap to correct; design errors are more expensive as they may involve the rewriting of several program components. Requirements errors are the most expensive to repair because of the extensive system redesign which may be necessary.

Adaptive Maintenance

Adaptive maintenance means changing the software to some new environment such as different hardware platform or for use with different operating system. The software functionality does not radically change.

Perfective Maintenance

This involves implementing a functional or non-functional system requirement. These are generated by software consumers as their organization or business changes.

Preventive Maintenance

This occurs when software is changed to improve future maintainability or reliability or to provide a better basis for future enhancement. In this project, all the above maintenance were implemented.

6.SECURITY MECHANISMS

6.SECURITY MECHANISMS

This project provides some security features. We can implement application security in the developing system. After registering details, they have to authenticate by providing the appropriate username and password. Securities are provided in this project that the data remains confidential. We can implement security in two ways, through passwords and through the limitations in the access rights. Password facility is implemented to avoid unauthorized access of information. For providing more security they have to change their passwords with their wishes. And it is more secure to change password periodically and it must be kept confidentially. A combination of alphabets, numbers and special characters make a password strong.

7.UPGRADABILITY POSSIBILITIES

7. UPGRADABILITY POSSIBILITIES

The technology is changing day to day. The efficiency of developed system can be improved by modifications. The quality of this online site can be improved by keeping wise list ofreports and other documents effectively. So, it is easy to add or remove modules. Software development in PHP is very flexible and all application was tested with live data and has proved respond successful. So, it is quite and helps in smooth migration from manual system to computerized system.

Our project *EFFECTIVE RESTAURANT HANDLING SYSTEM is* a project that contain only four modules such as Administrator, User, Restaurant, Staff. It is easy to develop and upgrade. It is compatible with any future developments.

8. CONCLUSION

8. CONCLUSION

The new system has overcome most of the limitations of the existing system and works according to the design specification given. The developed systems dispense theproblem and meet the needs of by providing reliable and comprehensive information. All the requirements projected by the user have been met by the system. The newly developedsystem consumes less processing time and all the details are updated and processed immediately. since the screen provides online help messages and is, very user-friendly, any user will get familiarized with its usage. Modules are designed to be highly flexible so thatany failure requirements can be easily added to the modules without facing many problems.

9.APPENDIX

APPENDIX A -MEETING MINUTES

MEETING MINUTES

Date:26-11-2022

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions, Ayur

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Individual Progress Report

We learned the fundamentals of PHP language and analyzed our project. There are five modules in the project.

Discussion of the Problem to Be Solved by The Software

Our Project is **EFFECTIVE RESTAURANT HANDLING SYSTEM** . Effective Restaurant Handling System is to reach to large range of customers and to educate them about existing and new packages and discounts offered by restaurants And another purpose is to allow customer to pay online the bill.

Discussion of Software Requirements

Operating System : Windows 10
Front End : PHP
Back End : MYSQL

Schedule Next Meeting

The team meeting will be on 30-12-2022

MEETING MINUTES

Date: 30-12-2022

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions, Ayur

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Discussion of Management Plan

The admin module is assigned to Abraham Mathai. Staff is assigned to Lijo James. The Restaurant module is assigned to Mohammad Roshan and User module is assigned to Noble P Biju.

Specific Task Assigned with Dead Line

Abraham Mathai: learned the fundamentals of PHP language

Lijo James : learned the fundamentals of PHP language

Mohammad Roshan : learned the fundamentals of PHP language

Noble P Biju: learned the fundamentals of PHP language

Schedule Next Meeting

The team meeting will be on 10-01-2023

External Guide

MEETING MINUTES

Date: 10-01-2023

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions, Ayur

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Individual Progress Report

We learned the fundamentals of PHP language and analyzed our project.

Discussion of Software Planning

We have downloaded templates that is required for the project. (Admin page, Home page, Login page,Restaurant Home, User Home, Staff Home).

Discussion of Software Requirements

Operating System	: Windows xp-11
Front End	: PHP
Back End	: MYSQL

Schedule Next Meeting

The team meeting will be on 19-01-2023

External Guide

MEETING MINUTES

Date: 19-01-2023

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Discussion Of Management Plan

Each member is assigned to create the view forms for their module

Schedule Next Meeting

The team meeting will be on 26-01-2023

External Guide

MEETING MINUTES

Date: 26-01-2023

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Discussion of Management Plan

Abraham Mathai : To design the login page and home page for Admin.

Lijo James : To design home page for Staff.

Mohammad Roshan : To design home page for Restaurant.

Noble P Biju : To design User home page

Schedule Next Meeting

The team meeting will be on 04-02-2023

External Guide

MEETING MINUTES

Date: 04-02-2023

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions, Ayur

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Discussion of Management Plan

Abraham Mathai : To set the profile of the Restaurant.

Lijo James : To set the profile edit of the Restaurant.

Mohammad Roshan : To set the profile edit of Staff.

Noble P Biju:To set profile of USer.

Schedule Next Meeting

The team meeting will be on 17-02-2023

External Guide

MEETING MINUTES

Date: 17-02-2023

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions, Ayur

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Individual Progress Report

Database connectivity has been done and sample data has been provided

Discussion Of Management Plan

Decided to start the coding for each module

Schedule Next Meeting

The team meeting will be on 25-02-2023

External Guide

MEETING MINUTES

Date: 25-02-2023

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions, Ayur

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Discussion Of Management Plan

We completed admin module (Registration and Home page).and we completed Registration and contact us page.

Schedule Next Meeting

The team meeting will be on 03-03-2023

External Guide

MEETING MINUTES

Date: 03-03-2023

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions, Ayur

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

Discussion Of Management Plan

User Handling and Bank Module.

Schedule Next Meeting

The team meeting will be on 09-03-2023

External Guide

MEETING MINUTES

Date: 09-03-2023

Time: 10:00 am to 4:00 pm

Location: Softzane Solutions, Ayur

Present:

1. Abraham Mathai
2. Lijo James
3. Mohammed Roshan
4. Noble P Biju

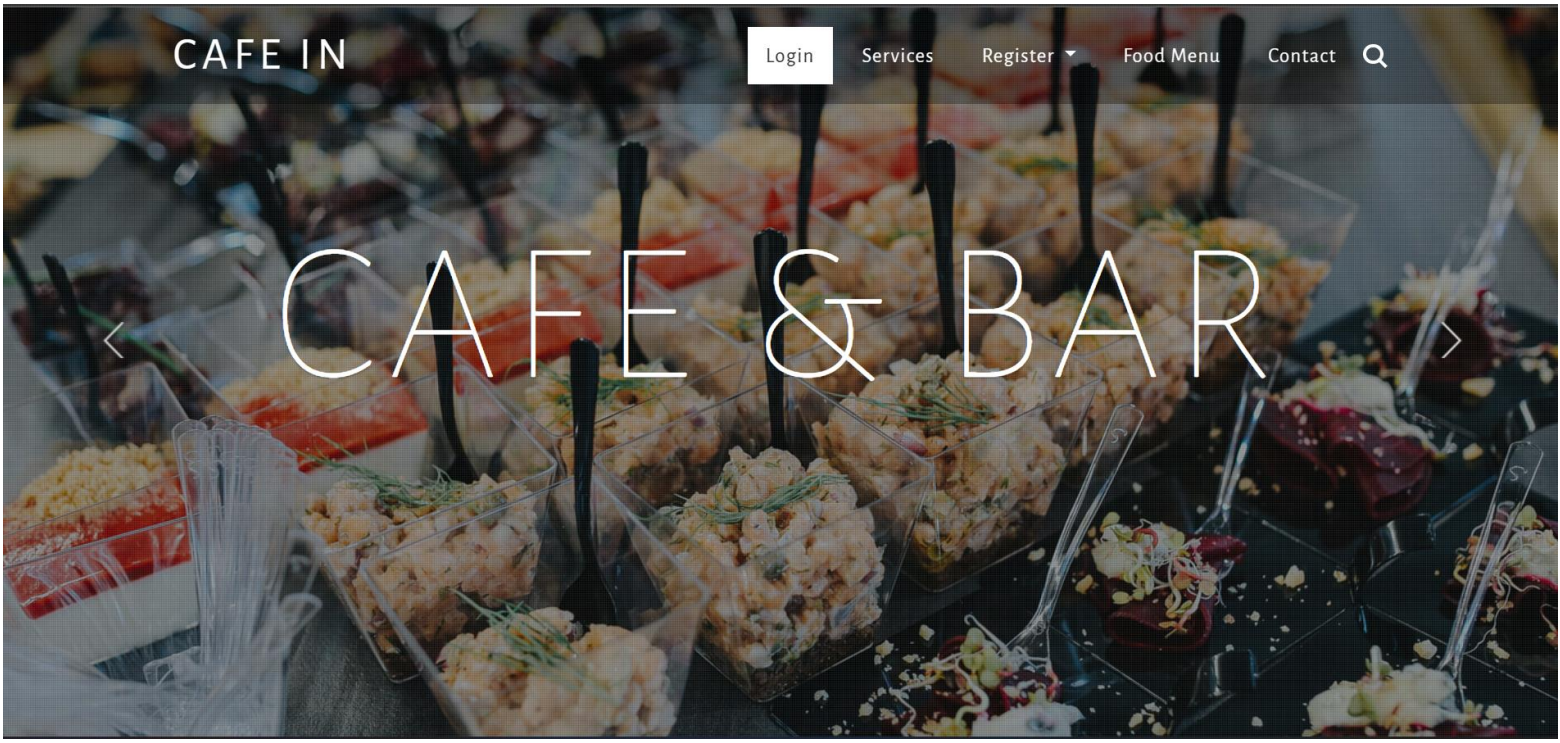
Discussion Of Management Plan

Booking Management

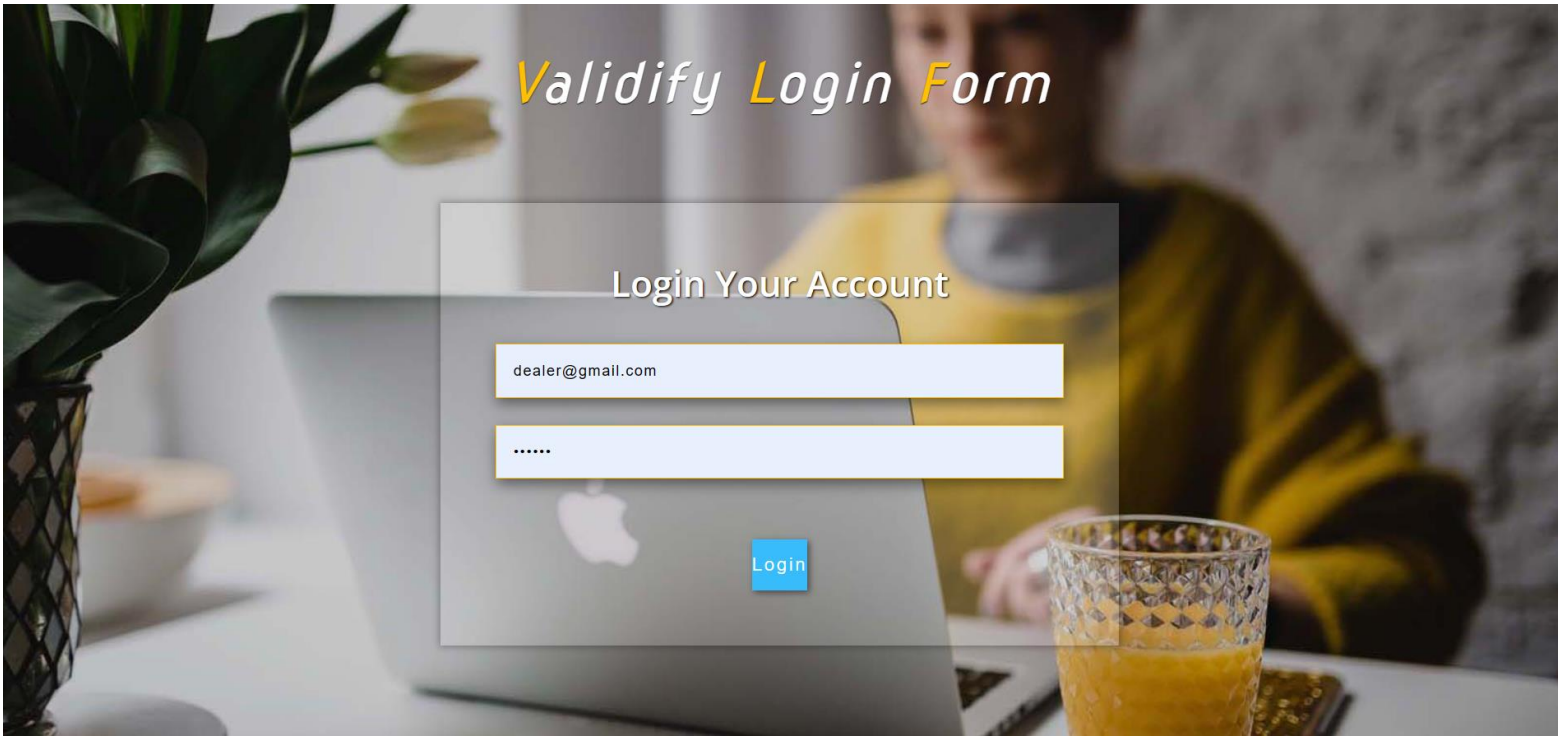
External Guide

APPENDIX B -FORM LAYOUTS

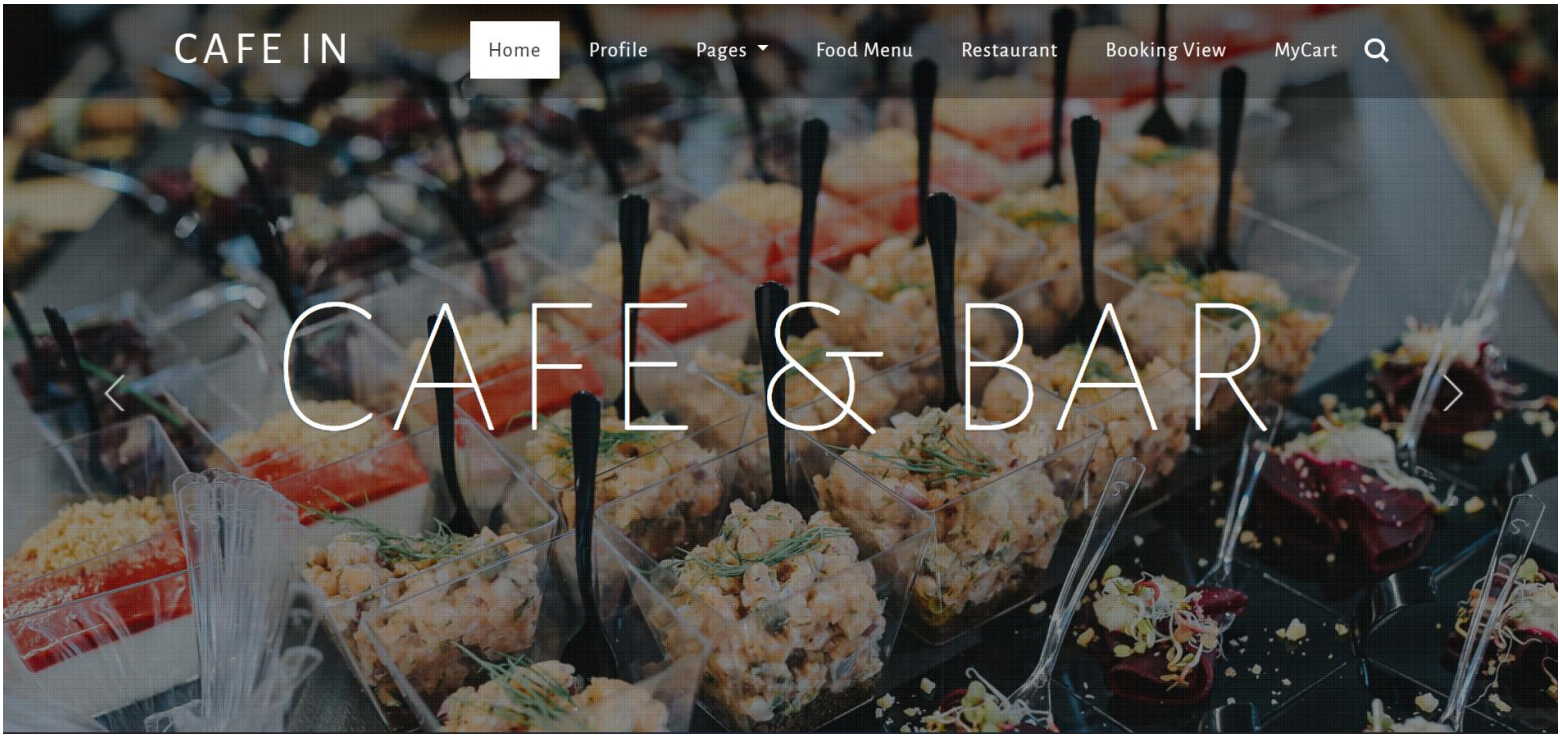
INDEX PAGE



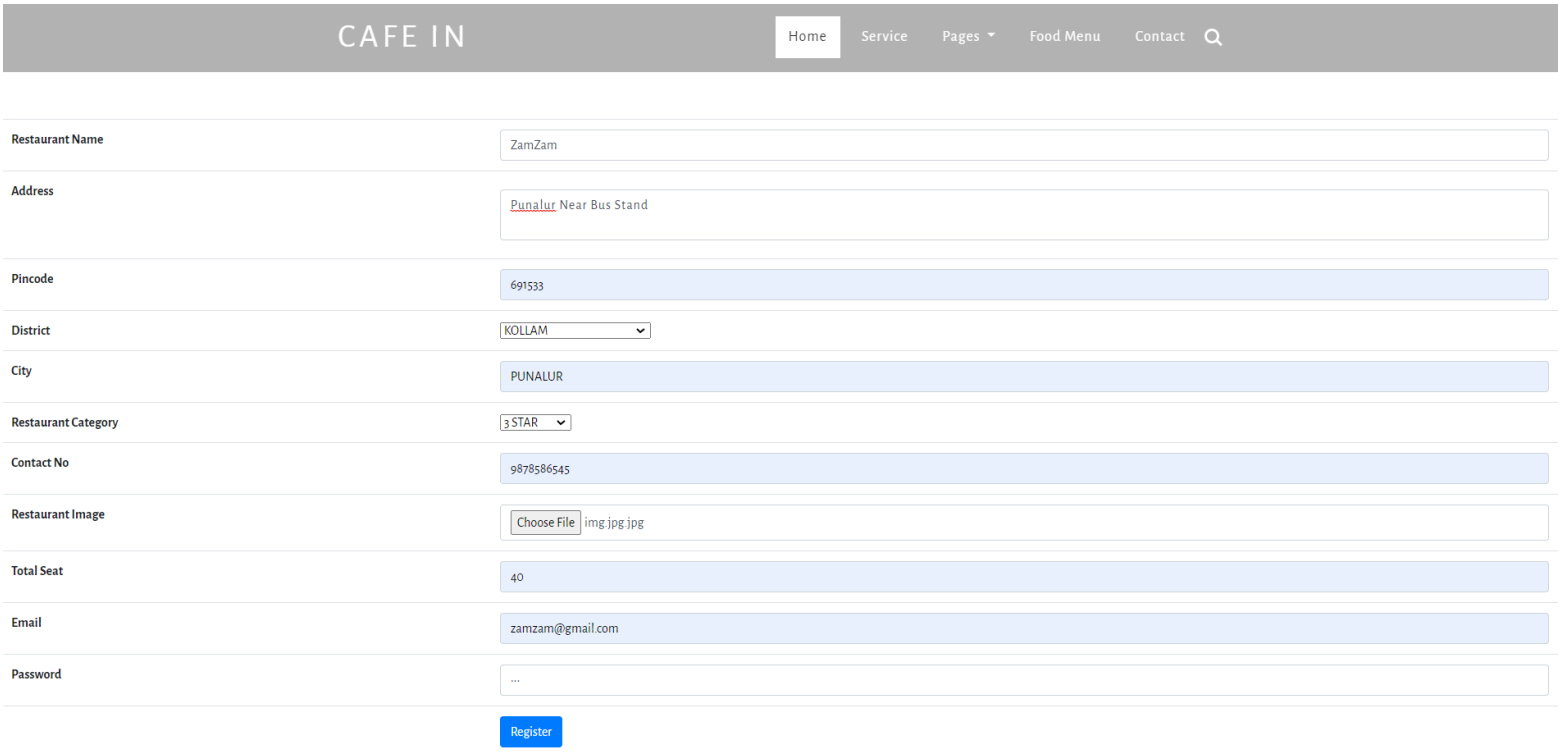
LOGIN PAGE


Validify Login Form
Login Your Account
dealer@gmail.com
.....
Login

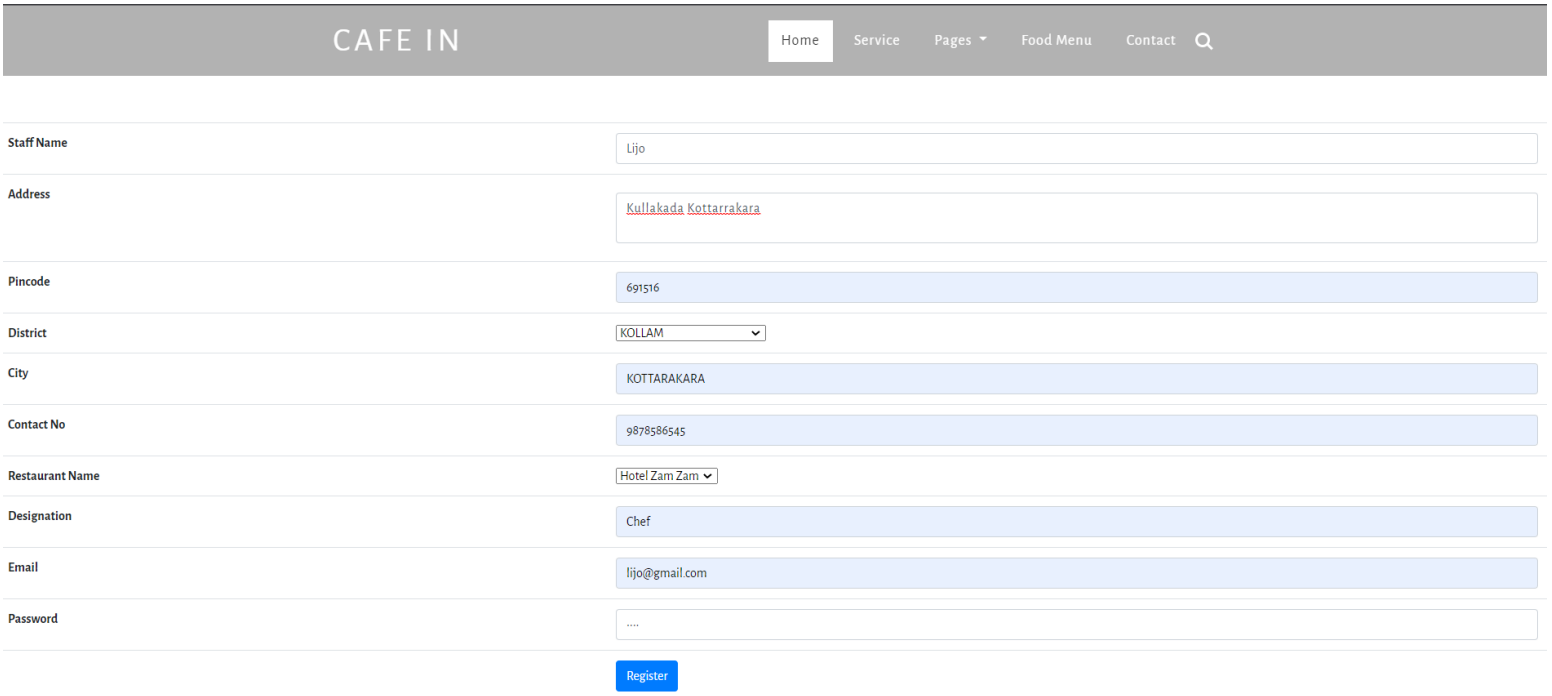
USER HOME


CAFE IN
Home
Profile
Pages
Food Menu
Restaurant
Booking View
MyCart
CAFE & BAR

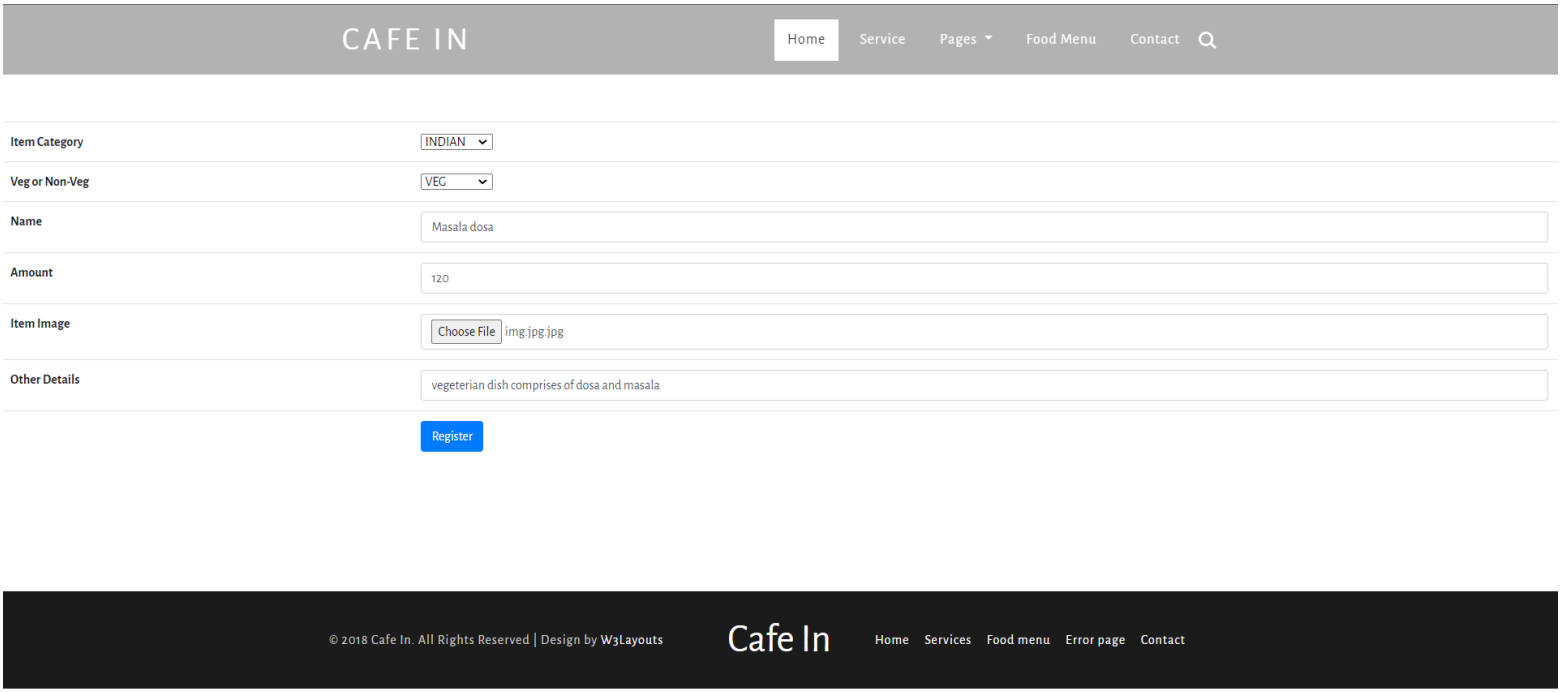
HOTEL REGISTRATION



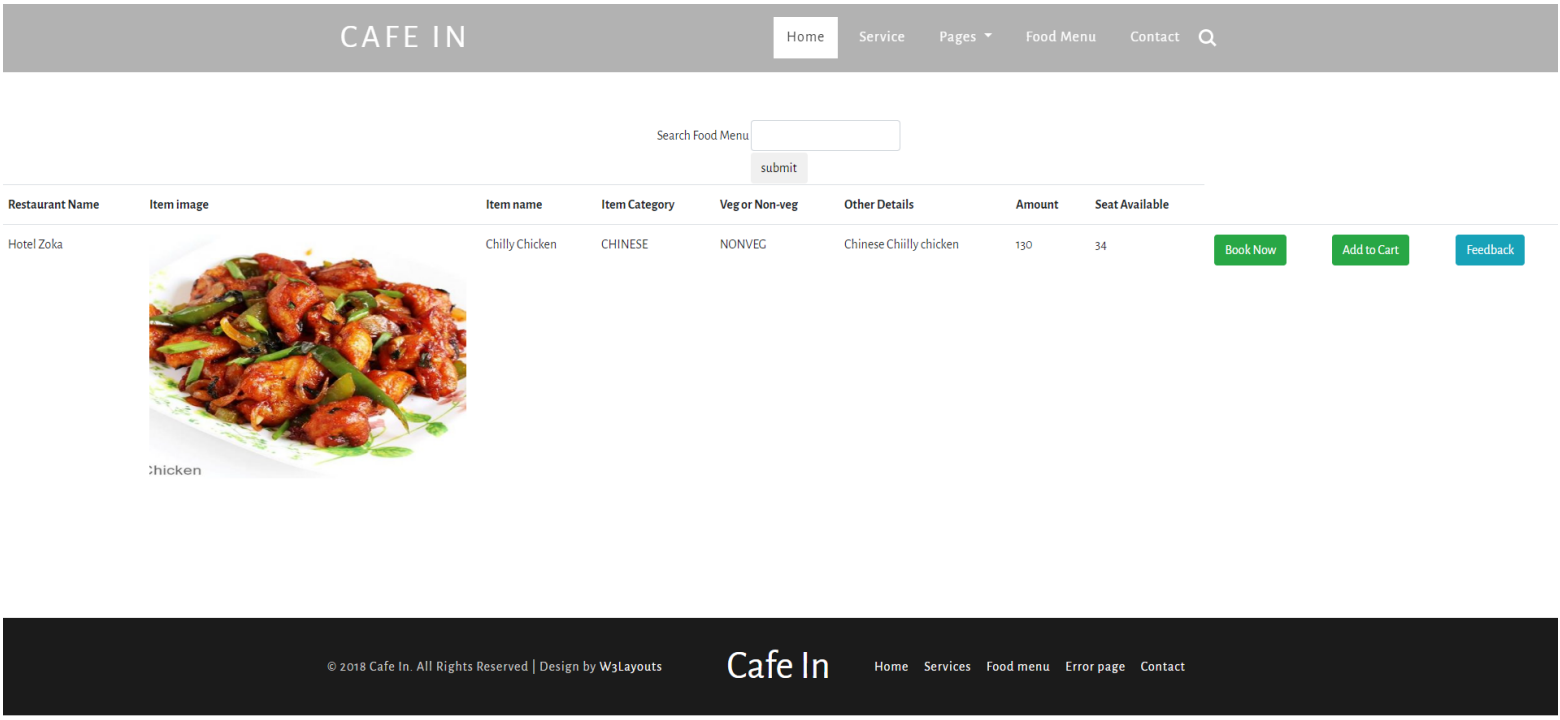
STAFF REGISTRATION



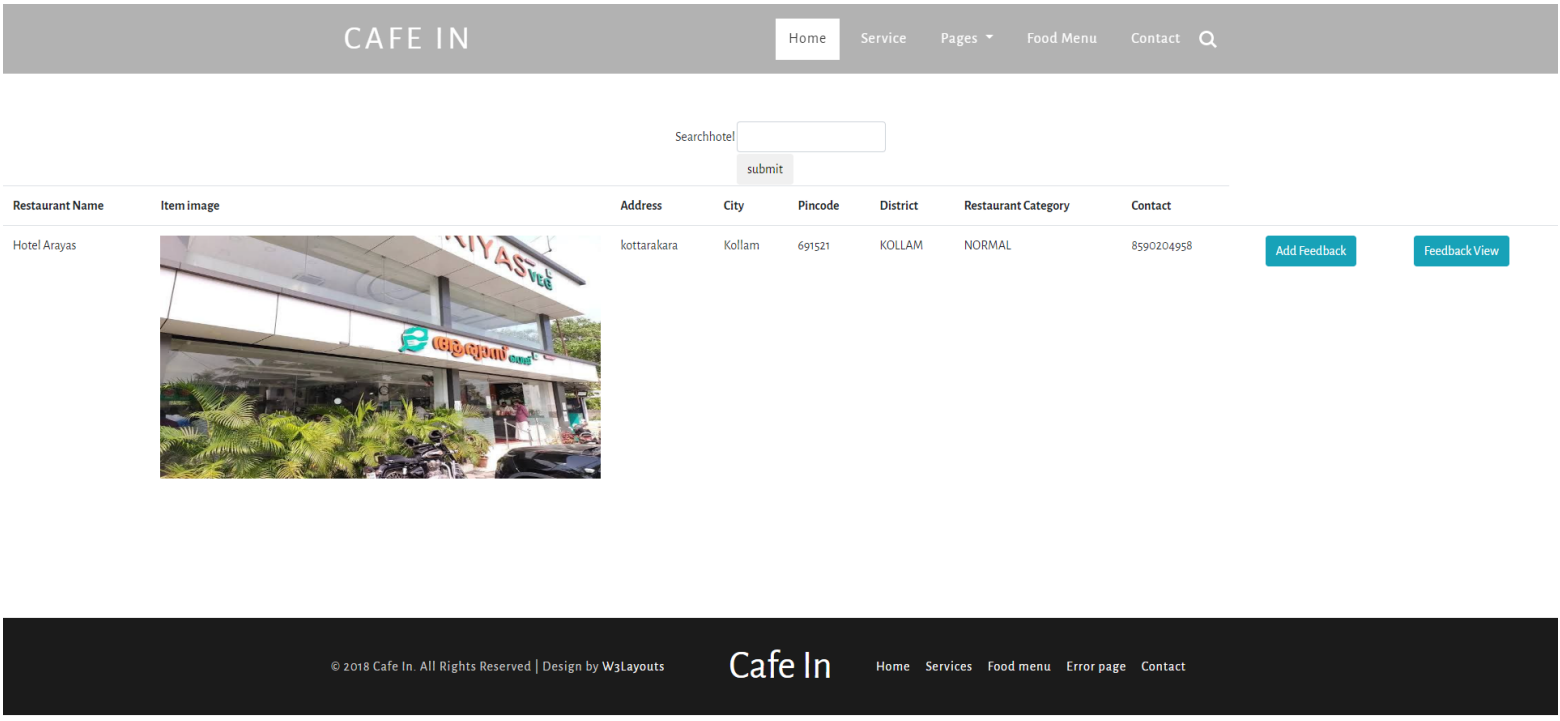
ADD MENU


CAFE IN
Home
Service
Pages
Food Menu
Contact
Item Category
INDIAN
Veg or Non-Veg
VEG
Name
Masala dosa
Amount
120
Item Image
Choose File
img.jpg.jpg
Other Details
vegeterian dish comprises of dosa and masala
Register
© 2018 Cafe In. All Rights Reserved | Design by W3Layouts
Cafe In
Home
Services
Food menu
Error page
Contact

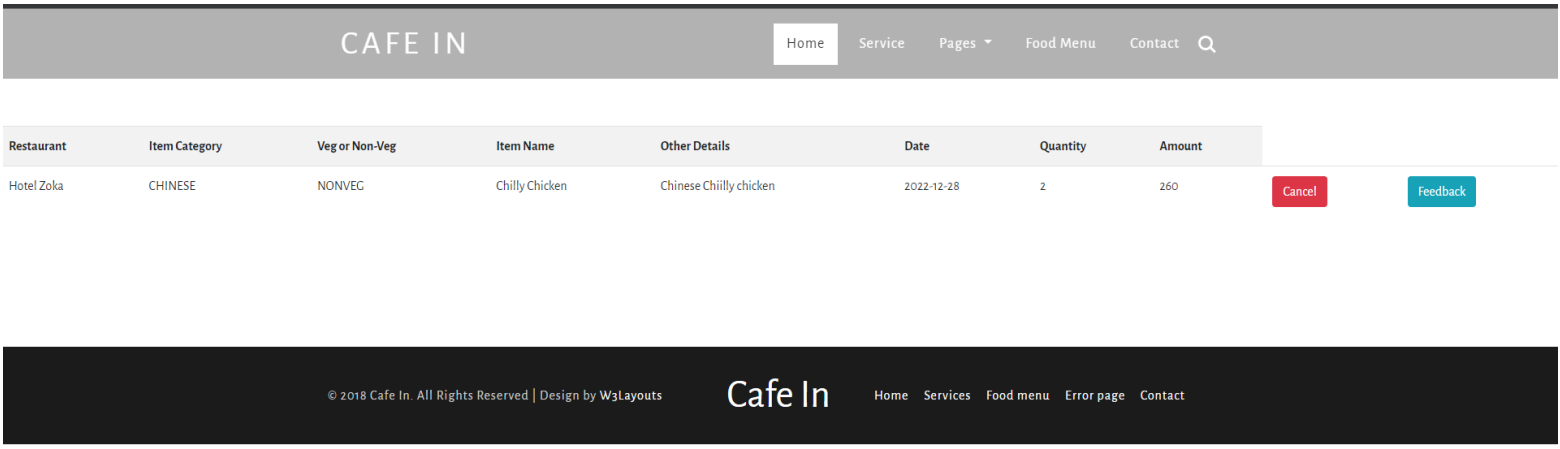
MENU BOOKING



SEARCH HOTEL



BOOKING VIEW



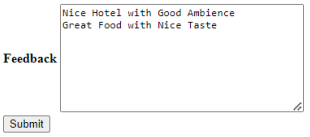
MY CART

CAFE IN

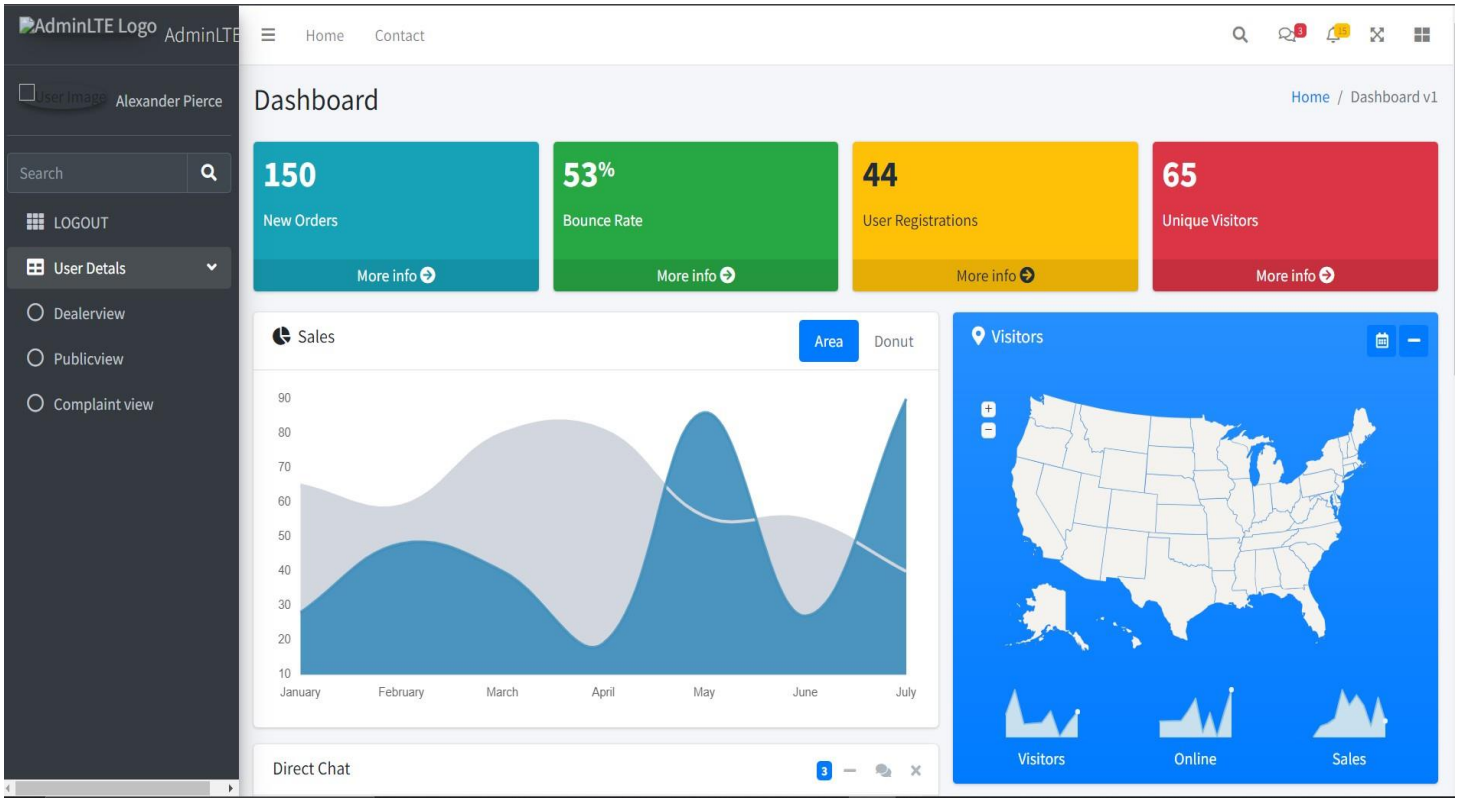
Home | Service | Pages | Food Menu | Contact

Item image	Item name	Item Category	Veg or Non-veg	Other Details	Quantity	Amount		
	Shawayi	ARABIAN	NONVEG	Grilled Chicken	5	1750	Buy	Delete
	Chilly Chicken	CHINESE	NONVEG	Chinese Chiilly chicken	2	260	Buy	Delete
	Chilly Chicken	CHINESE	NONVEG	Chinese Chiilly chicken	1	130	Buy	Delete

FEEDBACK


Nice Hotel with Good Ambience
Great Food with Nice Taste
Feedback
Submit

ADMIN


AdminLTE Logo
AdminLTE
Home
Contact
Alexander Pierce
Dashboard
Home / Dashboard v1
Search
LOGOUT
User Detals
Dealerview
Publicview
Complaint view
150
New Orders
More info
53%
Bounce Rate
More info
44
User Registrations
More info
65
Unique Visitors
More info
Sales
Area
Donut
90
80
70
60
50
40
30
20
10
January
February
March
April
May
June
July
Visitors
Visitors
Online
Sales
Direct Chat

APPENDIX C-SYSTEM CODING

restaurant.tpl

```
<!DOCTYPE html>
<html>
<head>
	<title>Restaurant</title>
</head>
<body>
	<table  class="table table-stripped">
		<form method="post" action="" enctype="multipart/form-data">
			<input type="hidden" name="hide" value="h">
			<tr>
				<th>Restaurant Name</th>
				<td>
					<input type="text" name="name"class="form-control">
				</td>
			</tr>
	<tr>
	<th>Address</th>
				<td>
					<textarea name="address"class="form-control"></textarea>
				</td>
			</tr>
			<tr>
				<th>Pincode</th>
				<td>
					<input type="text" name="pincode"class="form-control">
				</td>
			</tr>
			<tr><th>District</th><td><select name="district">
			<option value="SELECT">SELECT</option>
	<option value="ALAPUZHA">ALAPUZHA</option>
	<option value="ERNAKULAM">ERNAKULAM</option>
		<option value="IDUKI">IDUKI</option>
		<option value="KANNUR">KANNUR</option>
		<option value="KASARGOD">KASARGOD</option>
		<option value="KOLLAM">KOLLAM</option>
		<option value="KOTTYAM">KOTTAYAM</option>
		<option value="KOZHIKODE">KOZHIKODE</option>
		<option value="MALAPPURAM">MALAPPURAM</option>
		<option value="PALAKKAD">PALAKKAD</option>
		<option value="PATHANAMTHITTA">PATHANAMTHITTA</option>
		<option value="THIRUVANATHAPURAM">THIRUVANATHAPURAM</option>
		<option value="THRISSUR">THRISSUR</option>
		<option value="WAYANAD">WAYANAD</option>
	</select></td></tr>
			<tr>
				<th>City</th>
				<td>
					<input type="text" name="city"class="form-control">
				</td>
			</tr>
			<tr>
				<th>Restaurant Category</th>
				<td>
				<select name="rcategory">
```

```
                                <option value="SELECT">SELECT</option>
                                <option value="NORMAL">NORMAL</option>
        <option value="3 STAR">3 STAR</option>
            <option value="4 STAR">4 STAR</option>
            <option value="5 STAR">5 STAR</option>
                        </select>
                    </td>
                    </tr>
<tr>
                        <th>Contact No</th>
                        <td>
                            <input type="text" name="contact"class="form-control">
                        </td>
                    </tr>
                    <tr>
                        <th>Restaurant Image</th>
                        <td>
                            <input type="file" name="rimage"class="form-control">
                        </td>
                    </tr>
                    <tr>
                        <th>Total Seat</th>
                        <td>
                            <input type="text" name="seat"class="form-control">
                        </td>
                    </tr>
                    <tr>
                        <th>Email</th>
                        <td>
                            <input type="email" name="email"class="form-control">
                        </td>
                    </tr>
                    <tr>
                        <th>Password</th>
                        <td>
                            <input type="password" name="password"class="form-control">
                        </td>
                    </tr>
                    <tr>
                        <th>
                            <td>
                                <input type="Submit" name="submit" value="Register" class="btn btn-primary">
                            </td>
                        </th>
                    </tr>
            </form>
        </table>

</body>
</html>
```

menubooking.tpl

```
<!DOCTYPE html>
<html>
<head>
```

```
 <title>Menu Booking</title>
</head>
<body>
 <table  class="table table-stripped">
<form method="post" action="" enctype="multipart/form-data">
 <input type="hidden" name="hide" value="h">
                        <tr>
                                <th>Quantity</th>
                                <td>
                                        <input type="text" name="quantity"class="form-control">
                                </td>
                        </tr>
                        <tr>
                                <th>No of Seat</th>
                                <td>
                                        <input type="text" name="noofseat"class="form-control">
                                </td>
                        </tr>
                        <tr>
                                <th>Booking Time</th>
                                <td>
                                        <input type="time" name="time"class="form-control">
                                </td>
                        </tr>
                        <tr>
                                <th>
                                        <td>
                                                <input type="submit" value="submit">
                                        </td>

            <!-- <td> <a href="payment.php?key={$mkey}&&amount={$amount}" class="btn btn-success">Pay</a></td> -->

                                </th>

                        </tr>

                        </form>

        </table>

</body>
</html>
```

mycart.tpl

```
<!DOCTYPE html>
<html>
<head>
        <title>My Cart</title>
</head>
<body>

        <table class="table table-striped">
                <tr>

            <th>Item image</th>
            <th>Item name</th>
```

```
            <th>Item Category</th>
            <th>Veg or Non-veg</th>
            <th>Other Details</th>
            <th>Quantity</th>
            <th>Amount</th>
    </tr>


        {foreach from=$data item="d"}

            <tr>
            <td><img src="uploads/{$d.mkey}/{$d.itemimage}" height="300px" width="500px" ></td>
            <td>{$d.item}</td>
            <td>{$d.itemcategory}</td>
            <td>{$d.vegornon}</td>
            <td>{$d.otherdetails}</td>
            <td>{$d.quantity}</td>
            <td>{$d.totalamount}</td>
            <td><a href="cartpayment.php?key={$d.mkey}&&am={$d.amount}" class="btn btn-success">Buy</td>
            <td><a href="deletecart.php?key={$d.mbkey}" class="btn btn-danger">Delete</td>

    </tr>
    {/foreach}
    <tr>
       <th></th>
       <td></td>
       <td></td>
       <td></td>
       <td>Total Amount</td>
       <td>{$data1}</td>
       <td></td>
       <td><a href="buyall.php?key={$a.pkey}&&am={$data1}" class="btn btn-success">Buy All</td>
    </tr>


</table>
</body>
</html>
```

menu.tpl

```
<!DOCTYPE html>
<html>
<head>
        <title>Menu</title>
</head>
<body>
        <table  class="table table-stripped">
                <form method="post" action="" enctype="multipart/form-data">
                        <input type="hidden" name="hide" value="h">
                        <tr><th>Item Category</th><td><select name="itemcategory">
                        <option value="SELECT">SELECT</option>
        <option value="CHINESE">CHINESE</option>
        <option value="ARABIAN">ARABIAN</option>
            <option value="INDIAN">INDIAN</option>
            <option value="SPANISH">SPANISH</option>
```

```
                    </select></td></tr>
                    <tr><th>Veg or Non-Veg</th><td><select name="vegornon">
                    <option value="VEG">VEG</option>
     <option value="NONVEG">NON-VEG</option>
    <tr>
                        <th>Name</th>
                        <td>
                            <input type="text" name="item"class="form-control">
                        </td>
                    </tr>
                    <tr>
                        <th>Amount</th>
                        <td>
                        <input type="text" name="amount"class="form-control">
                        </td>
                    </tr>
                    <tr>
                        <th>Item Image</th>
                        <td>
                        <input type="file" name="itemimage"class="form-control">
                        </td>
                    </tr>
                     <tr>
                        <th>Other Details</th>
                        <td>
                        <input type="text" name="otherdetails"class="form-control">
                        </td>
                    </tr>
                    <tr>
                        <th>
                            <td>
                            <input type="Submit" name="submit" value="Register" class="btn btn-primary">
                            </td>
                        </th>
                    </tr>

            </form>
    </table>

</body>
</html>
```

menubooking.php

```
   <?php
    include_once"settings/settings.php";
    include_once"classes/userclass.php";
    $obj=new userclass();
    $lkey=$_COOKIE['lkey'];
    $mkey=$_GET['key'];
  // $mbkey=$_GET['mbkey'];
     $amount=$_GET['amount'];
    $smartyObj->assign("amount",$amount);
    $smartyObj->assign("mkey",$mkey);
   if(isset($_POST['hide'])AND($_POST['hide']) =="h")
{
        if(isset($_POST['quantity'])AND($_POST['quantity'])!=NULL)
        {
```

```
        if(isset($_POST['noofseat'])AND($_POST['noofseat'])!=NULL)
{
        if(isset($_POST['time'])AND($_POST['time'])!=NULL)
{
                                $quantity=trim($_POST['quantity']);
                                $noofseat=trim($_POST['noofseat']);
                                $time=trim($_POST['time']);

    $obj->menubooking($quantity,$noofseat,$time,$lkey,$mkey,$amount);

                                                                }
                                                                else
                                                        echo "<script>alert('time is empty')</script>";
                                                                }
                                                                else
                                                        echo "<script>alert('nooseat is empty')</script>";
                                                          }
                                                                else
                                                        echo "<script>alert('Quantity is empty')</script>";
                                                                }
$smartyObj->display('subheader.tpl');
$smartyObj->display('menubooking.tpl');
$smartyObj->display('footer.tpl');
?>
```

hotel.php

```
<?php
include_once"settings/settings.php";
include_once"classes/userclass.php";
$obj=new userclass();
if(isset($_POST['hide'])AND($_POST['hide']) =="h")
{
                if(isset($_POST['name'])AND($_POST['name'])!=NULL)
                {
                    if(isset($_POST['address'])AND($_POST['address'])!=NULL)
                    {
                            if(isset($_POST['pincode'])AND($_POST['pincode'])!=NULL)
                            {
                                    if(isset($_POST['district'])AND($_POST['district'])!=NULL)
                                    {
                                            if(isset($_POST['city'])AND($_POST['city'])!=NULL)
                                            {
                                                    if(isset($_POST['rcategory'])AND($_POST['rcategory'])!=NULL)
                                                    {
                                                            if(isset($_POST['contact'])AND($_POST['contact'])!=NULL)
                                                            {
                                                                    //
if(isset($_POST['rimage'])AND($_POST['rimage'])!=NULL)
                                                                    // {
```

```
                if(isset($_POST['seat'])AND($_POST['seat'])!=NULL)
                                                        {

                if(isset($_POST['email'])AND($_POST['email'])!=NULL)
                                                                {

                if(isset($_POST['password'])AND($_POST['password'])!=NULL)
                                                                        {
                $name=trim($_POST['name']);
                $address=trim($_POST['address']);
                $pincode=trim($_POST['pincode']);
                $district=trim($_POST['district']);
                $city=trim($_POST['city']);

                $rcategory=trim($_POST['rcategory']);
                $contact=trim($_POST['contact']);
                // $rimage=trim($_POST['rimage']);
                $seat=trim($_POST['seat']);
                $email=trim($_POST['email']);

                $password=trim($_POST['password']);
              $obj-
>restaurant($name,$address,$pincode,$district,$city,$rcategory,$contact,$seat,$email,$password,$_FILES['rimage']);
                                                                        }
                                                                        else
                                 echo "<script>alert('Password is empty')</script>";
                                                                }
                                                                else
                                echo "<script>alert('Email is empty')</script>";
                                                                }
                                        else
                                 echo "<script>alert('seat is empty')</script>";
                                                        }
                                                //          else
                                 echo "<script>alert('Image is empty')</script>";
                                                // }
                                                else
                                 echo "<script>alert('Contact No is empty')</script>";
                                        }
                                        else
                                 echo "<script>alert('Category is empty')</script>";
                                    }
                                    else
                                 echo "<script>alert('City is empty')</script>";
                                 }
                                 else
                                 echo "<script>alert('District is empty')</script>";
                        }
                        else
                                 echo "<script>alert('Pincode is empty')</script>";
                    }
                    else
                                echo "<script>alert('Address is empty')</script>";
                }
                else
                                 echo "<script>alert('Name is empty')</script>";
}
```

```
$smartyObj->display('subheader.tpl');
$smartyObj->display('restaurant.tpl');
 $smartyObj->display('footer.tpl');
 ?>
```

feedback.php

```
 <?php
 include_once "settings/settings.php";
 include_once "classes/userclass.php";
$obj=new userclass();
$key=$_COOKIE['lkey'];
$rkey=$_GET['rkey'];
$smartyObj->assign("rkey",$rkey);
 if (isset($_POST['hide'])and($_POST['hide'])=='h')
  {
                if(isset($_POST ['feedback'])and($_POST['feedback'])!=null)
                {

                    $feedback=trim($_POST['feedback']);


                            $obj->hotelfeedback($feedback,$key,$rkey);


   }
   else
                    echo "<script>alert('enter your feedback')</script>";
   }



  $smartyObj->display("hotelfeedback.tpl");


    ?>
```

payment.php

```
 <?php
 include_once "settings/settings.php";
 include_once "classes/userclass.php";
$obj=new userclass();
$key=$_COOKIE['lkey'];
$amount=$_GET['amount'];
 $key1=$_GET['mkey'];
$mbkey=$_GET['mbkey'];

        if(isset($_POST['hide']) AND ($_POST['hide'])=='h')
            {
                    if(isset($_POST['nameoncard']) AND ($_POST['nameoncard'])!=null)
                                {

                                        // if(isset($_POST['email']) AND ($_POST['email'])!=null)
                                //          {
                                                if(isset($_POST['cardno']) AND ($_POST['cardno'])!=null)
```

```
                        {
                                if(isset($_POST['cvv']) AND ($_POST['cvv'])!=null)
                        {
                                if(isset($_POST['expmonth']) AND ($_POST['expmonth'])!=null)
                        {
                                if(isset($_POST['expyear']) AND ($_POST['expyear'])!=null)
                        {



                                $nameoncard=trim($_POST['nameoncard']);
                                // $email=trim($_POST['email']);
                                $cardno=trim($_POST['cardno']);
                                $cvv=trim($_POST['cvv']);
                                $expmonth=trim($_POST['expmonth']);
                                $expyear=trim($_POST['expyear']);

        $obj->payment($nameoncard,$cardno,$cvv,$expmonth,$expyear,$amount,$key,$key1,$mbkey);
                                                                }

                                        else
                                                echo "<script>alert('expyear is empty')</script>";
                                }
                                else
                                                echo "<script>alert('expmonth is empty')</script>";
                                                                }
                                else
                                                echo "<script>alert('cvv is empty')</script>";

                        }
                        else
                                                echo "<script>alert('cardno is empty')</script>";
                    // }
                    // else
                    //                         echo "<script>alert('email is empty')</script>";
                    }
                    else
                                                echo "<script>alert('nameoncard is empty')</script>";
}

    $smartyObj->display("payment.tpl");


?>
```

searchhotel.php

```
    <?php
     include_once"settings/settings.php";
     include_once"classes/userclass.php";
     $obj=new userclass();

      $key=$_COOKIE['lkey'];

     if(isset($_POST['hide'])AND($_POST['hide'])=='h')
       {
                        $a=trim($_POST['search']);
```

```
                $s=$obj->searchhotel($a);
                $smartyObj->assign("view",$s);

 }
 $smartyObj->display('subheader.tpl');
 $smartyObj->display('searchhotel.tpl');
 $smartyObj->display('footer.tpl');


?>
```

menuedit.php

```
 <?php
 include_once"settings/settings.php";
include_once"classes/userclass.php";
$obj=new userclass();
$key1=$_COOKIE['lkey'];
  $d=$_GET['key'];
 $k=$obj->menuview($key1);
 $smartyObj->assign("data",$k);

  if(isset($_POST['hide'])AND($_POST['hide']) =="h")
        {
                if(isset($_POST['itemcategory'])AND($_POST['itemcategory'])!=NULL)
                {
                      if(isset($_POST['vegornon'])AND($_POST['vegornon'])!=NULL)
                      {
                              if(isset($_POST['item'])AND($_POST['item'])!=NULL)
                              {
                                      if(isset($_POST['amount'])AND($_POST['amount'])!=NULL)
                                      {


                if(isset($_POST['otherdetails'])AND($_POST['otherdetails'])!=NULL)
                                                                                {


                $itemcategory=trim($_POST['itemcategory']);

                $vegornon=trim($_POST['vegornon']);
                  $item=trim($_POST['item']);

                $amount=trim($_POST['amount']);
               $rimage=trim($_POST['rimage']);

                $otherdetails=trim($_POST['otherdetails']);

    $obj->menuedit($itemcategory,$vegornon,$item,$amount,$_FILES['itemimage'],$otherdetails,$key1,$d);

                      }
                      else
                      echo "<script>alert('otherdetails is empty')</script>";
                              }
                //     else
                // echo "<script>alert('Image is empty')</script>";
                // }
```

```
                                        else
                                                                        echo "<script>alert('amount is empty')</script>";
                                }
                                else
                                                                        echo "<script>alert('name is empty')</script>";
                        }
                        else
                                                                        echo "<script>alert('vegornon is empty')</script>";
                }
                else
                                                                        echo "<script>alert('itemcategory is empty')</script>";
  }
 $smartyObj->display('subheader.tpl');
 $smartyObj->display('menuedit.tpl');
 $smartyObj->display('footer.tpl');
?>
```

userupdate.php

```
 <?php
 include_once"settings/settings.php";
include_once"classes/userclass.php";
$obj=new userclass();
 $key=$_COOKIE['lkey'];
$k=$obj->userview($key);
 $smartyObj->assign("data",$k);
if(isset($_POST['hide'])AND($_POST['hide']) =="h")
    {
                if(isset($_POST['username'])AND($_POST['username'])!=NULL)
                {

                                if(isset($_POST['contact'])AND($_POST['contact'])!=NULL)
                                        {
        if(isset($_POST['email'])AND($_POST['email'])!=NULL)
                                                                        {



                $name=trim($_POST['username']);

                $contact=trim($_POST['contact']);
                $email=trim($_POST['email']);

                $obj->useredit($username,$contact,$email,$key);

                }
                else
                echo "<script>alert('Email is empty')</script>";
                }
                else
                echo "<script>alert('Contact No is empty')</script>";

                }
                else
                echo "<script>alert('Name is empty')</script>";
```

```
}
 $smartyObj->display('usersubheader.tpl');
 $smartyObj->display('userupdate.tpl');
  $smartyObj->display('footer.tpl');
?>
```

userfeedbackview.php

```
 <?php
 include_once"settings/settings.php";
include_once"classes/userclass.php";
 $obj=new userclass();
 $key=$_COOKIE['lkey'];
 $rkey=$_GET['rkey'];
 $s=$obj->userhotelfeedbackview($key,$rkey);
 $smartyObj->display("subheader.tpl");

 $smartyObj->display("userhotelfeedbackview.tpl");
 $smartyObj->display("footer.tpl");
 ?>
```

menufeedback.php

```
 <?php
 include_once"settings/settings.php";
 include_once"classes/userclass.php";
 $obj=new userclass();
 $key=$_COOKIE['lkey'];
 $s=$obj->menufeedback($key1);
 $smartyObj->assign("data",$s);
 $smartyObj->display("subheader.tpl");

 $smartyObj->display("menufeedback.tpl");
 $smartyObj->display("footer.tpl");
 ?>
```

10.REFERENCES

I. Books

1. Julie C Meloni," PHP, MySQL and Appache", Pearson Education
2. Rajib Mall, "Fundamentals of Software Engineering", PHI
3. V.K Jain, "Advanced Programming in Web Design", Cyber Tech Publications
4. Ramon A.Mata-toledo and Pauline Cushman," Fundamentals of Relational Data Bases", SchaumOutlines, MaGraw Hills
5. Idolia Cox Collier, "Medical Surgical Nursing", Third Edition

II. Websites

1. localhost/two wheeler rental service/index.php
2. https://www.w3schools.com/php/default.asp
3. https://www.php.net
4. https://www.onnbikes.com
5. https://www.royalbrothers.com