

**ATMIYA UNIVERSITY**

**RAJKOT**



A

Report On

**HOTEL FOOD ORDER**

Under subject of

**MAJOR PROJECT**

B.TECH, Semester–VII

(Computer Engineering)

Submitted by:

- |                                |           |
|--------------------------------|-----------|
| 1. Varsani Krushali Dineshbhai | 190002119 |
| 2. Patadia Dhara Prashantbhai  | 190002079 |

**Prof. Krina Masharu**

(FacultyGuide)

**Prof. Tosal M. Bhalodia**

(Head of the Department)

Academic Year

**(2022-23)**

## **CANDIDATE'S DECLARATION**

We hereby declare that the work presented in this project entitled “**HOTEL FOOD ORDER**” submitted towards completion of project in **7<sup>th</sup> Semester** of B. Tech. (Computer Engineering) is an authentic record of our original work carried out under the guidance of —**Prof. Krina Masharu**”.

We have not submitted the matter embodied in this project for the award of any other degree.

Semester: 7<sup>th</sup>

Place: Rajkot

**Signature:**

Varsani Krushali (190002119)

Patadia Dhara (190002079)

# ATMIYA UNIVERSITY RAJKOT



## CERTIFICATE

Date:

This is to certify that the HOTEL FOOD ORDER has been carried out by Varsani Krushali (190002119) under my guidance in fulfilment of the subject Major Project in COMPUTER ENGINEERING (7<sup>th</sup>Semester) of Atmiya University, Rajkot during the academic year 2022.

Prof. Krina Masharu  
**(Project Guide)**

Prof. Tosai M. Bhalodia  
**(Head of the Department)**

## **ACKNOWLEDGEMENT**

We have taken many efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

We are highly indebted to **Prof. Krina Masharu** for their guidance and constant supervision as well as for providing necessary information regarding the Major Project titled “**Hotel Food Order**”. We would like to express our gratitude towards staff members of Computer Engineering Department, Atmiya University for their kind co- operation and encouragement which helped us in completion of this project.

We even thank and appreciate to our colleague in developing the project and people who have willingly helped us out with their abilities.

Varsani Krushali (190002119)

Patadia Dhara (190002079)

## **ABSTRACT**

The system is implemented to reduce the manual work and enhances the accuracy of work in a restaurant. This system manages and maintains the record of customers and their order online. This Android App has been made in a user friendly interface. So that Customer can add and delete the food items easily. The menu card of different restaurant consists of various food varieties available in the restaurant. Through the place ordering menu, the customer can simply click and order the food. The messaging module tells the supplier to supply the particular food. Also tracking module track the order. The billing system prepares the bill according to the delivered food. This system entirely reduces the unnecessary time. Every order is associated with an individual seat at the table, and orders are built one customer at a time, just like on paper, but with greater accuracy. Items can also easily be shared by the whole table, moved or modified, and noted and the cost can be calculated in real time.

## INDEX

Sr. No.	TITLES	Page No.
	<b>Acknowledgement</b>	<b>1</b>
	<b>Abstract</b>	<b>2</b>
	<b>List of Figures</b>	<b>3</b>
	<b>List of Tables</b>	<b>4</b>
<b>1.</b>	<b>Introduction</b>	<b>11</b>
1.1.	Purpose	11
1.2.	Scope	11
1.3.	Objective	11
1.4.	Technology and tools	12
<b>2.</b>	<b>Project Management</b>	<b>15</b>
2.1.	Project Planning	15
2.2.	Project Scheduling	16
2.3.	Risk Management	17
2.4.	Risk Identification	17
2.5.	Risk Analysis	18
<b>3.</b>	<b>System Requirements Study</b>	<b>19</b>
3.1.	User characteristics	19
3.2.	Hardware and Software Requirements	19
3.2.1.	Server side hardware requirement	19
3.2.2.	Software requirement	20
3.2.3.	Client Side requirement	20
3.3.	Constraints	20
3.3.1.	Hardware Limitation	20
3.3.2.	Interfacing with other systems	20
3.3.3.	Reliability requirements	21
3.4.	Assumption and dependency	21
<b>4.</b>	<b>System Analysis</b>	<b>22</b>
4.1.	Study of Current System	22
4.2.	Problem and Weaknesses of Current System	22

4.3. Requirements of New System	22
4.3.1. User Requirements	22
4.3.2. System Requirements	22
4.4. Feasibility Study	22
4.4.1. Does the system contribute to overall objectives of the organization?	23
4.4.2. Can the system be implemented using the current technology and within the given cost and schedule constraints	23
4.4.3. Can the system be integrated with other system which are already in place?	23
4.5. Requirement validation	23
4.6. Selection of hardware and software	23
<b>5. System Design</b>	<b>25</b>
5.1. Database Design	25
5.2. Input-Output Design	27
5.3. Interface Design	30
5.3.1. Use Case Diagram	30
5.3.2. Sequence Diagram	31
5.3.3. Data Flow Diagram	32
5.3.4. ER Diagram	32
<b>6. Code Implementation</b>	<b>34</b>
6.1. Implementation Environment	34
6.2. Program/Module Specification	34
6.3. Coding Standards	34
6.3.1. Variable Standards	34
6.3.2. Comment Standards	34
<b>7. Testing</b>	
7.1. Testing	<b>36</b>
7.2. Testing Plan	36
7.2.1. Design Testing	36
7.2.2. Implementation Testing	36
7.2.3. Coding Testing	36
7.3. Testing Strategy	36
7.4. Testing Method	38

7.4.1. Unit Testing	38
7.4.2. Integration Testing	38
7.4.3. Validation Testing	38
7.4.4. Storage Testing	38
7.5. Test Cases	38
7.5.1. Purpose	38
<b>8. Limitations and Future Enhancement</b>	<b>40</b>
8.1. Limitations	40
8.2. Future Enhancement	40
<b>9. Conclusion and Discussion</b>	<b>41</b>
9.1. Self –analysis and Project viabilities	41
9.2. Problem Encountered and Possible Solutions	41
9.2.1. Resource availability	41
9.2.2. To get the knowledge of company	41
9.2.3. Requirement Understanding	41
<b>10. References</b>	<b>41</b>



## **LIST OF FIGURES**

<b>Figure No.</b>	<b>Table Title</b>	<b>Page No.</b>
5.2.1	Main Page	33
5.2.2	Categories Page	34
5.2.3	User Register	34
5.2.4	User Login	35
5.2.5	Admin Login	35
5.3.1	Use case Diagram	36
5.3.2	Sequence Diagram	37
5.3.3	Flowchart	38
5.3.4	Dataflow Diagram	39
5.5.5	ER Diagram	40

## LIST OF TABLES

<b>Table No.</b>	<b>Table Title</b>	<b>Page No.</b>
2.1.2.1	Milestones and Deliverables	18
3.2.1.1	Server side Hardware Requirement	22
3.2.2.3	Software Requirements	23
3.2.3.1	Client side Requirements	23
4.9.1	Hardware Requirements	28
4.9.2	Software Requirements	28
4.9.3	Client side Requirements	29
7.4.1.1	Login Test	47
7.4.1.2	Signup Test	47
7.4.1.3	Admin login Test	48

# CHAPTER-1

## INTRODUCTION

### **1. Introduction**

Online Restaurant Management System is the process of ordering food from a website .The product can be either ready-to-eat food. The aim of developing Online Restaurant Management project is to replace the traditional way of taking orders with computerized system. Another important reason for developing this project is to prepare order summary reports quickly and in correct format at any point of time when required. Online Restaurant Management has a very lot of scope. This PHP project can be used by any restaurants or fast foods for customers for keeping their order records. This project is easy, fast and accurate. It requires less disk space. Online Restaurant Management uses MYSQL Server as backend so there is not any chance of data loss or data security. A customer can choose to have the food delivered or for pick-up. The process consists of a customer choosing the restaurant of their choice, scanning the menu items, choosing an item, and finally choosing for pick-up or delivery. Payment is then administered by paying with a credit card or debit card through the app or website or in cash at the restaurant when going to pick up. The website and app inform the customer of the food quality, duration of food preparation, and when the food is ready for pick-up or the amount of time it will take for delivery.

### **1.1 Purpose**

It helps the restaurant manager to manage the restaurant more effectively and efficiently by computerizing meal ordering, billing and inventory control. This management system is designed for a all type of foods category.

### **1.2 Scope**

Scope of this project is very wide, from small person to society and everyone who want to do and know something new. Also who want to invest money and monetary support to the persons who have great skills.

### **1.3 Objective**

The main objective of this system is to manage the details of item category, food, delivery address, order, and shopping cart. It manages all the information about item category, customer,

3 shopping cart, item category. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose is to build an application program to reduce the managing of the item category, food customers. It tracks all the delivery addresses ordered.

## 1.4 Technology and tools

**Front End:** For designing the structure of the project following technologies are used.

### 1. HTML:

- Hypertext Markup Language (HTML) is the main markup language for creating web pages and other information that can be displayed in a web browser.
- HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like `<html>`), within the web page content. HTML tags most commonly come in pairs like `<h1>` and `</h1>`, although some tags represent empty elements and so are unpaired, for example `<img>`.
- The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, further tags, comments and other types of text-based content.
- The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.

### 2. CSS:

- Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation semantics (the look and formatting) of a document written in a markup language. Its most common application is to style web pages written in HTML and XHTML, but the language can also be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts.
- This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for tableless web design).
- It depends on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet perhaps one on their own computer, to override the one the author has specified.
- CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities or weights are calculated and assigned to rules, so that the results are predictable.

- The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998), and they also operate a free CSS validation service.

### 3. JavaScript:

- JavaScript (JS) is an interpreted computer programming language. As part of web browsers, implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It has also become common in server-side programming, game development and the creation of desktop applications.
- JavaScript is a prototype-based scripting language with dynamic typing and has first class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages.
- It is a multi- paradigm language, supporting object-oriented, imperative, and functional programming styles.
- JavaScript's use in applications outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and frameworks built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications.
- JavaScript was formalized in the ECMA Script language standard and is primarily used as part of a web browser (client-side JavaScript). This enables programmatic access to computational objects within a host environment.

**Back End:** Back End technologies used in the website are:

#### 1. SQL:

- SQL (Structured Query Language) is a special-purpose programming language designed for managing data held in a relational database management system (RDBMS).
- Originally based upon relational algebra and tuple relational calculus, SQL consists of a data definition language and a data manipulation language. The scope of SQL includes data insert, query, update and delete, schema creation and modification, and data access control. Although SQL is often described as, and to a great extent is, a declarative language (4GL), it also includes procedural elements. SQL was one of the first commercial languages for Edgar F. Codd's relational model, as described in his influential 1970 paper "A Relational Model of Data for Large Shared Data Banks". Despite not entirely adhering to the relational model as described by Codd, it became the most widely used database language.
- SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standards (ISO) in 1987. Since then, the standard has been enhanced several times with added features,

But code is not completely portable among different database systems, which can lead to vendor locking. The different makers do not perfectly follow the standard, they add extensions, and the standard is sometimes ambiguous.

## **2. PHP:**

- PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. 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 Preprocessor, 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 commandline 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 General Public 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.

## CHAPTER-2

### Project Management

## **2. Project Management**

### **2.1 Project Planning**

Project Planning is concerned with identifying and measuring the activities, milestones and deliverables produced by the project. Project planning is undertaken and completed sometimes even before any development activity starts. Project planning consists of following essential activities:

- Scheduling manpower and other resources needed to develop the system.
- Staff organization and staffing plans.
- Risk identification, analysis, and accurate planning.
- Estimating some of the basic attributes of the project like cost, duration and efforts. the effectiveness of the subsequent planning activities is based on the accuracy of these estimations.
- Miscellaneous plans like quality assurance plan, configuration management plan, etc. Project management involves planning, monitoring and control of the people, process, and the events that occurs as the software evolves from a preliminary concept to an operational implementation. Cost estimation is a relative activity that is concerned with the resources required to accomplish the project

#### **2.1.1 Project Development Approach And Justification:**

A Software process model is a simplified abstract representation of a software process, which is presented from a particular perspective. A process model for software engineering is chosen based on the nature of the project and application, the methods and tools to be used, and the controls and deliverables that are required. All software development can be characterized as a problem-solving loop which in four distinct stages is encountered:

Requirement analysis

- Design
- Coding
- Testing
- Deployment

## 2.1.2 Milestones and Deliverables:

Management needs information. As software is tangible, this information can only be provided as documents that describe the state of the software being developed without this information it is impossible to judge progress at different phases and therefore schedules cannot be determined or updated. Milestone is an end point of the software process activity. At each milestone there should be formal output such as report that can be represented to the management. Milestones are the completion of the outputs for each activity. Deliverables are the requirements definition and the requirements specification. Milestone represents the end of the distinct, logical stage in the project. Milestone may be internal project results that are used by the project manager to check progress. Deliverables are usually Milestones but reverse need not be true. We have divided the software process into activities for the following milestone that should be achieved.

Software Process Activity	Milestone
Project Plan	Project schedule
Requirement Collection	User requirements, System Requirements
Data flow analysis	DFD, System flow
Design 1. Database design 2. User Interface design	System Design Document
Implementation 1. Code for giving security 2. Code for reports	Access Rights Reports Generation
Testing	Setting validations and error messages

**Table 2.1.2.1: Milestones and Deliverables**

## 2.1.3 Roles and Responsibilities

This phase defines the role and responsibilities of each and every member involved in developing the system. To develop this system there was only one group with two members working on the whole application. Each member was responsible for each and every part of developing the system. Each of the group members has sufficient knowledge in several programming languages. Our team structure is of mixed control team organization as it consists of both democratic and chief programmer organization.

## 2.2 Project Scheduling

The scheduling is the peak of a planning activity, a primary component of software project management. When combined with estimation methods and risk analysis, scheduling establishes a roadmap for project management. The characteristics of the project are used to adapt an appropriate task set for doing work.



## 2.3 Risk Management

Risk management consists of a series of steps that help a software development team to understand and manage uncertain problems that may arise during the course of software development and can plague a software project.

Risks are the dangerous conditions or potential problems for the system which may damage the system functionalities to very high level which would not be acceptable at any cost. so in order to make our system stable and give its 100% performance we must have identify those risks, analyze their occurrences and effects on our system and must prevent them to occur .

## 2.4 Risk Identification

Risk identification is a first systematic attempt to specify risks to project plan, scheduling resources, project development. It may be carried out as a team process using brainstorming approach.

**Technology risk:** Technical risks concern implementation, potential design, interfacing, testing, and maintenance problems

- Database Corruptness
- Garbage Collection

**People Risks:** These risks are concerns with the team and its members who are taking part in developing the system. Leaking an important data

- Lack of clear product vision
- Technical staff conflict
- Poor communication between people.

**Tools Risks:** These are more concerned with tools used to develop the system Tools containing virus.

**General Risks:** General Risks are the risks, which are concerned with the mentality and resources.

- Rapidly changing requirements.
- Lack of resources can cause great harm to efficiency and timely productivity.
- Changes in requirements can cause a great harm to implementation, designing and schedule of developing the system.
- Insufficient planning and task identification.

## **2.5 Risk Analysis**

**“Risk analysis = risk assessment + risk management + risk communication.”**

Risk analysis is employed in its broadest sense to include:

### **Risk assessment**

Involves identifying sources of potential harm, assessing the likelihood that harm will occur and the consequences if harm does occur. For this project It might be :- System Crash.

### **Risk management**

Evaluates which risks identified in the risk assessment process require management and selects and implements the plans or actions that are required to ensure that those risks are controlled.

Precautions taken to make risks minimal are as under:- Periodical backups are taken to avoid major loss in case of system crash.

### **Risk communication**

Involves an interactive dialogue between stakeholders and risk assessors and risk managers which actively informs the other processes.

Steps taken for risk communication is as under:-

Probability of certain risks is negotiated with client.

All the possible risks are listed out during communication and project is developed taking care of that risks.

## **CHAPTER 3**

### **System Requirements Study**

#### **3.1 User Characteristics Admin**

- Login
- Show project and Dashboard
- Show project and user full detail
- Manage user
- Manage category (n level category)
- Manage Eligible country
- Manage static page
- Mange project
- Manage FAQ

#### **3.2 Hardware And Software Requirement Specification**

This shows minimum requirements to carry on to run this system efficiently.

##### **3.2.1 Hardware Requirements**

###### **Server side Hardware Requirement:**

Devices	Description
Processor	Intel Core Duo 2.0 GHz or more
RAM	512 MB or more
Hard Disk	10 GB or more

**Table 3.2.1.1: Server side Hardware Requirement**

## 3.2.2 Software Requirements

For Which	Software
Operating System	Windows7/8/10, Linux
Front End	HTML, CSS
Back End	MySQL Database
Scripting Language	Javascript

**Table 3.2.2.1: Software Requirements**

## 3.2.3 Client side Requirements

For which	Requirement
Browser	Any Compatible browser device

**Table 3.2.3.1 : client side Requirements**

## 3.3 Constraints

### 3.3.1 Hardware Limitations

The major hardware limitations faced by the system are as follows: If the appropriate hardware is not there like processor, RAM, hard disks -The problem in processing requests of client -If appropriate storage is not there our whole database will crash due to less storage because our main requirement is large storage.

### 3.3.2 Interfacing with other systems

There should be the compatible browser to perfectly run our portal.

### **3.3.3 Reliability Requirements**

Since many users can access the server simultaneously, load on the server becomes very high. Hence, the server should be of enough high configurations. There should be high back up storage and management of huge data for overall ideas, videos , images, multiple countries, multiple user profile.

The Reliability requirements are the validations used to protect the system against one or more incorrect activities. Without proper validation of the system, the failure possibilities of it grow higher so it is must to understand the proper validation of the system and must implement them. All the required validator controls spend very good role to keep the system secure from any unauthorized or incorrect information. In all these validation actions if system found one or more entries violating validation rules then user will be warned by proper error messages and the details or the record is not going to be saved until corrections are made to them.

## **3.4 Assumptions And Dependencies**

### **Dependencies**

The entire project depends on end-users operation. They should possess enough knowledge to work with the system.

### **Assumptions**

- End-User is the person having enough knowledge for the project operation.
- The PC on which this software is used must meet its minimum requirements in terms of hardware and software.
- Only Administrator person has all the privileges.
- The database is correct and up-to date every Time.
- The availability and reliability of the system is at the level what user wants in.

The user of the system must beware about the various functionalities of the system and all the operations of it.

## **CHAPTER- 4**

### **System Analysis**

#### **4.1 Study Of Current System**

My current system is dividing into two parts one for admin and one for user side.

#### **4.2 Problems And Weakness Of Current System**

The current system is undoubtedly well-designed for crowd funding portal expenses but it has some following limitations:

- Lack of awareness of this system.
- Some Security related issues may be created □ Idea stealing problem is there.

#### **4.3 Requirements Specification**

Requirements specification adds further information to the requirements definition.

##### **4.3.1 User Requirements**

Here two types of user can use this project

###### **Admin:**

Admin users manage the entire project like login, management of Project ,Patients ,Doctors, Updates, FAQ

##### **4.3.2 System Requirements**

###### **Usability:**

The interface should use terms and concepts, which are drawn from the experience of the people who will make most of the system. For example, basic social networking concepts are followed.

###### **Efficiency:**

The system should provide easy and fast access.

#### **4.4 Feasibility Study**

An important outcome of the preliminary investigation is the determination that the system is feasible or not. The main aim of the feasibility study activity is to determine whether it would be financially and technically feasible to develop a project .The feasibility study activity involves the analysis of the problem and collection of all

relevant information relating to the product such as the different data items which would be input to the system, the processing required to be carried out on these data, the output required to be produced by the system as well as the various constraints on the behaviour of the system.

#### **4.4.1. Does the system contribute to the overall objectives of the organization?**

The main aim of this website is to provide easily search information which user required. And also they got best food of good quality.

#### **4.4.2. Can the system be implemented using the current technology and within the given cost and schedule constraints?**

The system can be easily implemented using existing technology. The technology used is PHP which is user friendly and freeware. After seeing the functionality that system provides the cost of developing the application does not matter.

#### **4.4.3. Can the system be integrated with other system which is already in place?**

Yes, the system can be integrated with other system which is already in place. If other system wants to use our functionality it can be easily integrate.

## **4.5 Requirement Validation**

A requirements validation is concerned to check whether the requirements actually define the system, which the customer wants? Requirements validation is important because errors in requirements document can lead to extensive rework costs when they are subsequently discovered. We have performed the following validation checks :

### **Validity checks**

Check whether the information entered is in valid format.

### **Existence check**

Check whether search data is present or not in database.

## **4.6 Selection Of Hardware And Software**

The Tables 4.6.1 and 4.6.2, 4.6.3 below give idea of the hardware and software required for the system and client side requirements.

**Hardware Selection :**

Devices	Description
Processor	Intel Core Duo 2.0 GHz or more
RAM	512 MB or more
Hard Disk	10 GB or more

**Table 4.6.1: Hardware Requirements****Software Selection :**

For which	Software
Operating System	Windows7/8/10,Linux
Front End	HTML
Back End	MySQL Database
Scripting Language	, JavaScript

**Table 4.6.2: Software Requirements****Client side requirements:**

For which	Requirement
Browser	Any Compatible browser device

**Table 4.6.3: Client Side Requirements**



## **CHAPTER- 5**

### **System Design**

#### **5.1 Database Design**

As a system data model is derived, many named entities, relationships and so forth will be identified. The names given to the entities should be chosen to give the reader some clues to their meaning. However, further description of the named entities is usually needed to make the model understandable. The description can be informal or the formal, whether which approach is used. It is always worth collecting all the descriptions in a single repository or data dictionary.

A data dictionary is simplistically a list of names used by the system, arranged alphabetically. As well as the name, the dictionary should include a description of the named entity and if the name represents a composite object, there may be a description of the composition. Other information such as the date of creation, the creator, and the representation of the entity may also be included depending on the type of the model which is being developed.

#### **Advantages of using a data dictionary are:**

It is a mechanism for name management. Many different people who have to invent names for entities and relationships may develop a large system model. These names should be used consistently and should not clash with their meanings. The data dictionary software can check for the name uniqueness and tell requirements analyst of the name duplications.

It serves as a store of organizational information which can link analysis, design, implementation and evolution. As the system is developed, information is taken to inform the development. New information is added to it. All information about an entity is in one place.

All system names, whether they are names of entities, types, relations, attributes or services should be entered in the dictionary. Support software should be available to relate, maintain and interrogate the dictionary. This software might be integrated with other tools so that dictionary creation is partially automated.

## Database Table:

### admin

Column	Type	Null	Default	Links to	Comments	MIME
id (Primary)	int(11)	No				
name	varchar(250)	No				
email	varchar(50)	No				
password	varchar(250)	No				

### Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	

### categories

Column	Type	Null	Default	Links to	Comments	MIME
id (Primary)	int(11)	No				
name	varchar(250)	No				
short_desc	varchar(250)	No				
long_desc	varchar(500)	No				

Fig. 5.1.1: Database Design

## 5.2 Input output design

Home page

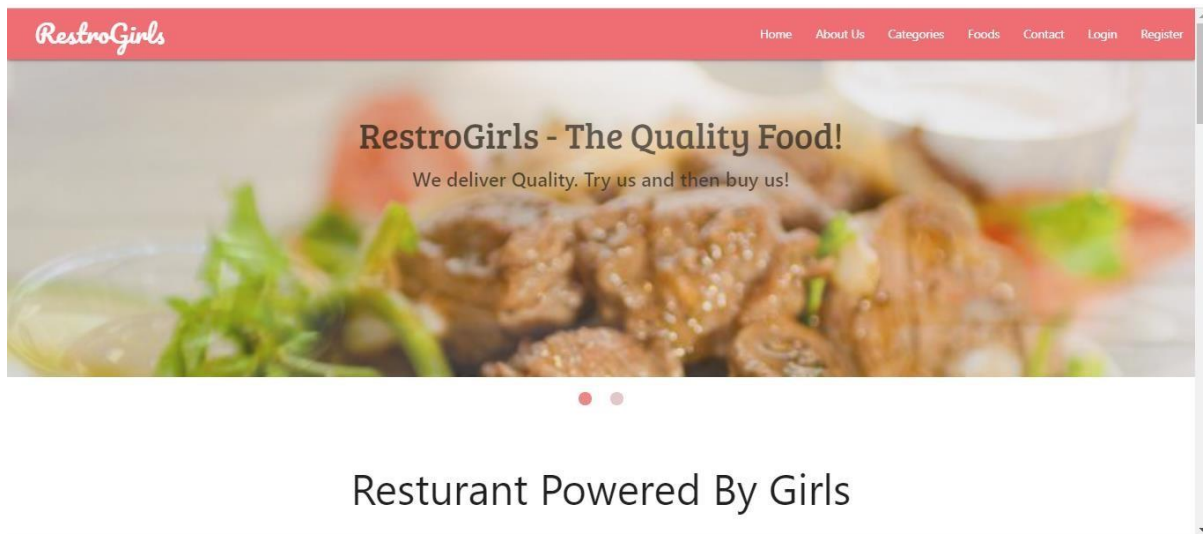


Fig. 5.2.1: Home Page

Categories page

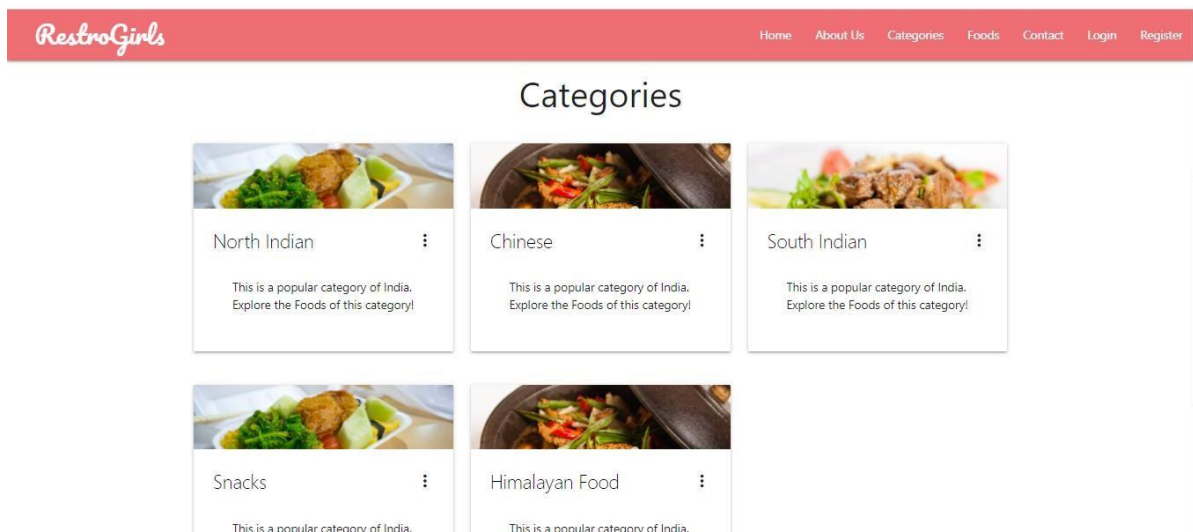


Fig. 5.2.2: Categories page

## User Register

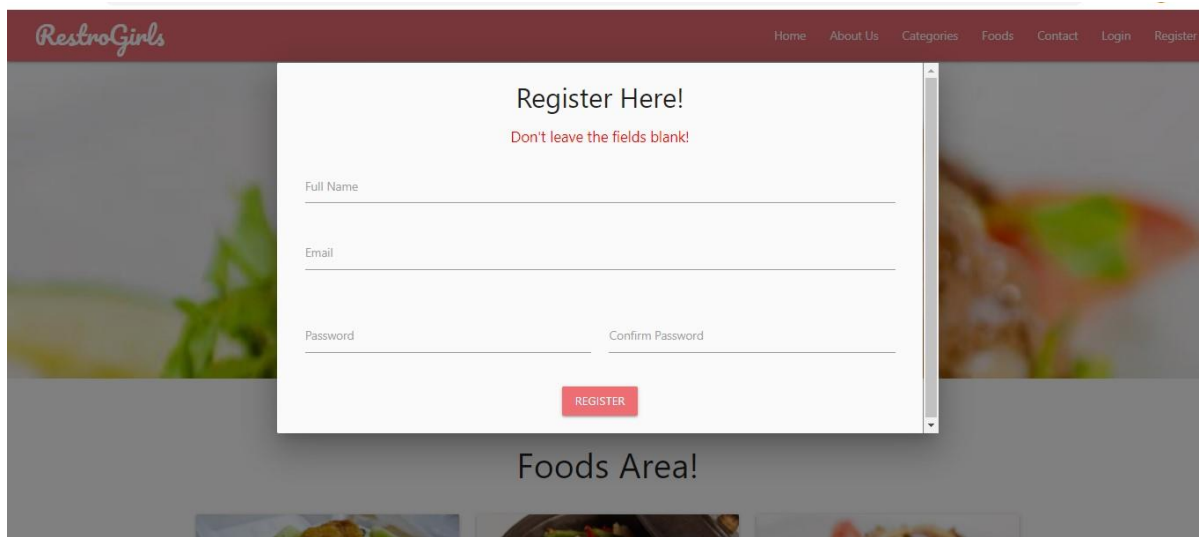


Fig. 5.2.3: User Register

## User Login

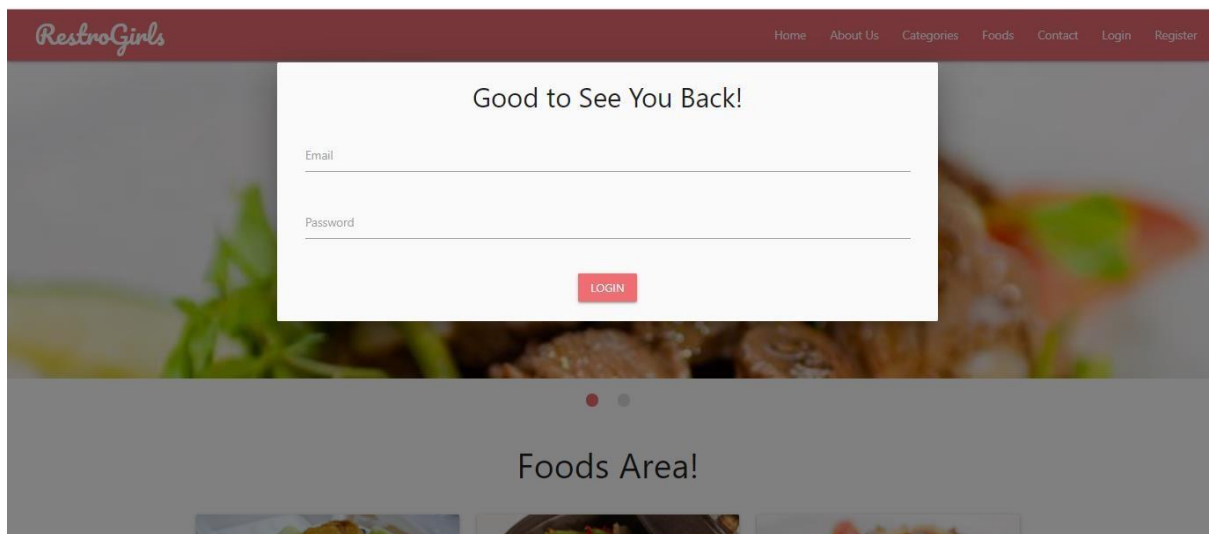


Fig. 5.2.4: User Login

Admin Login

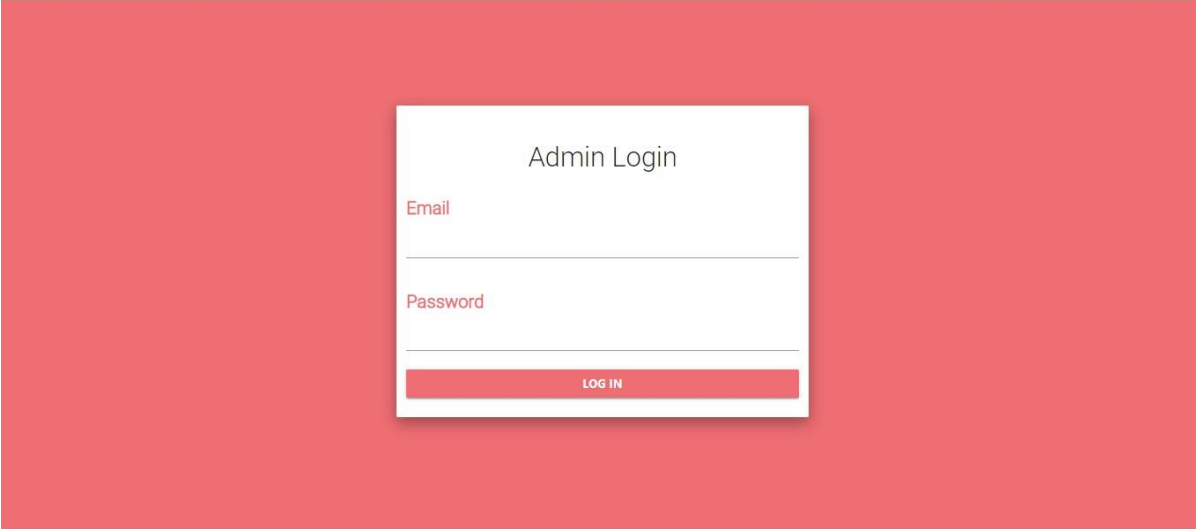


Fig. 5.2.5: Admin Login

## 5.3 Interface Design

### Use case diagram

Use Case Diagram A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse an actor is a person, organization, or external system that plays a role in one or more interactions with your system.

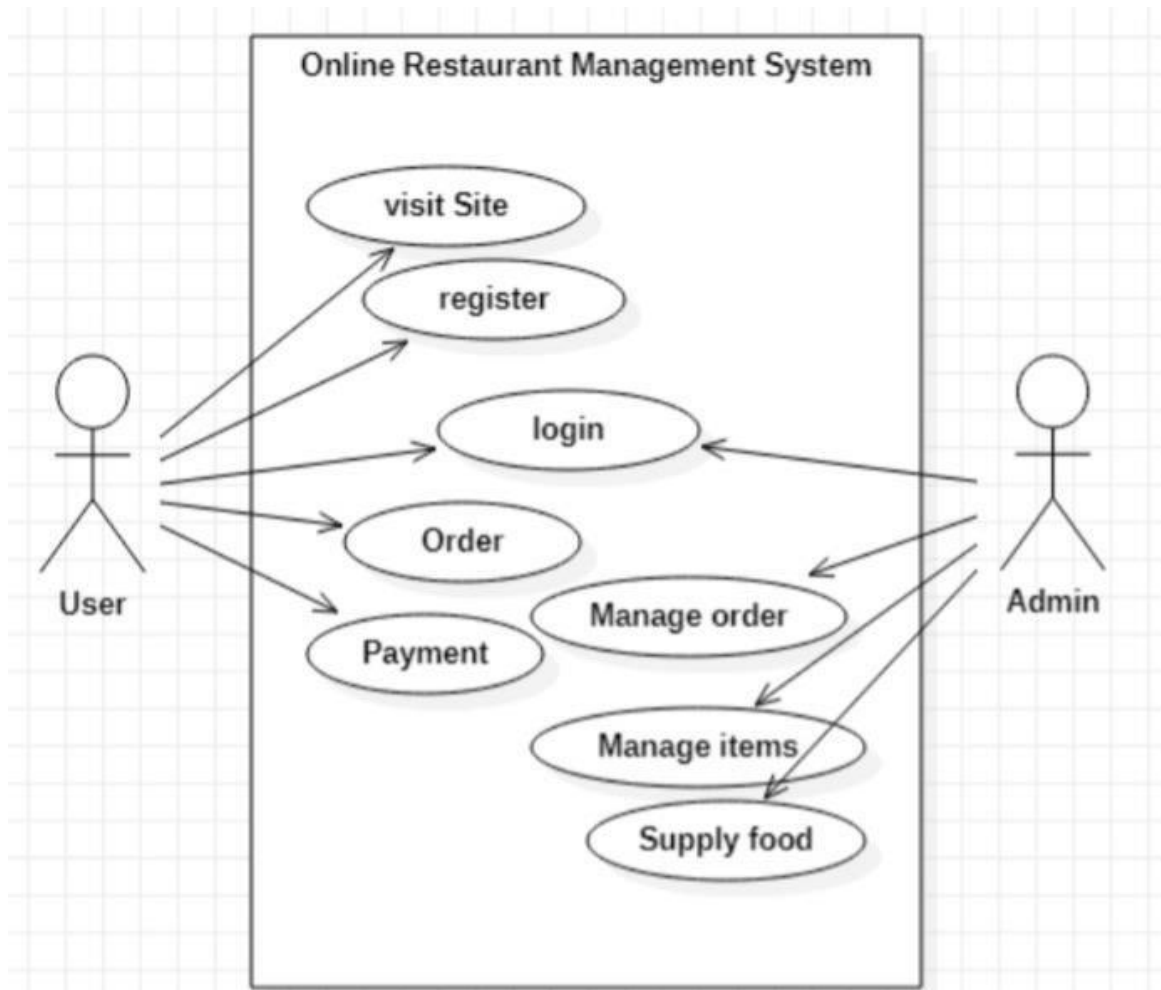


Fig.5.3.1 :Use Case Diagram

## Schema Diagram

UML sequence diagrams model the flow of logic within your system in a visual manner, enabling you both to document and validate your logic, and are commonly used for both analysis and design purposes. Sequence diagrams are the most popular UML artifacts for dynamic modeling, which focuses on identifying the behavior within your system.

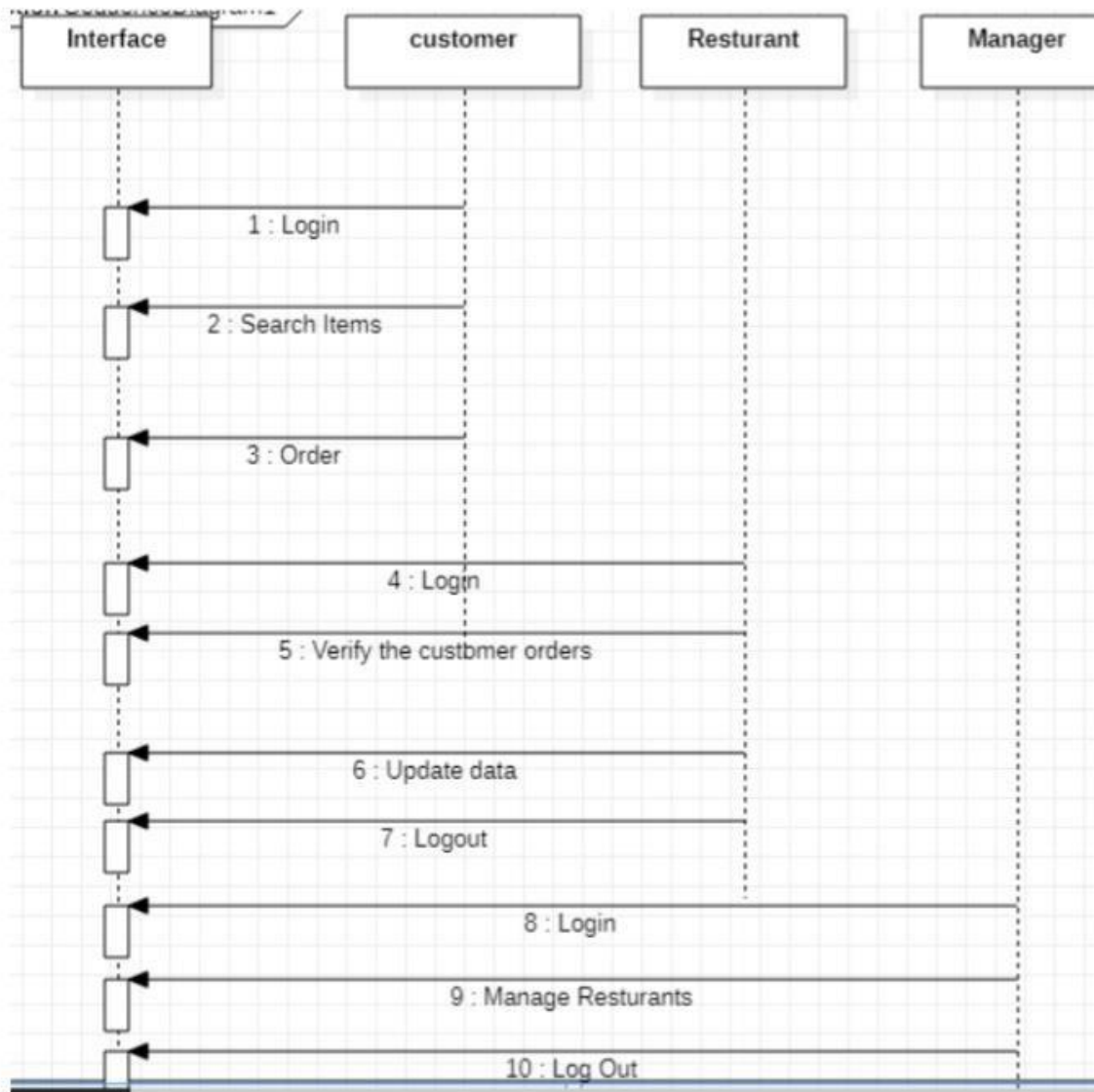


Fig.5.3.2: Sequence Diagram

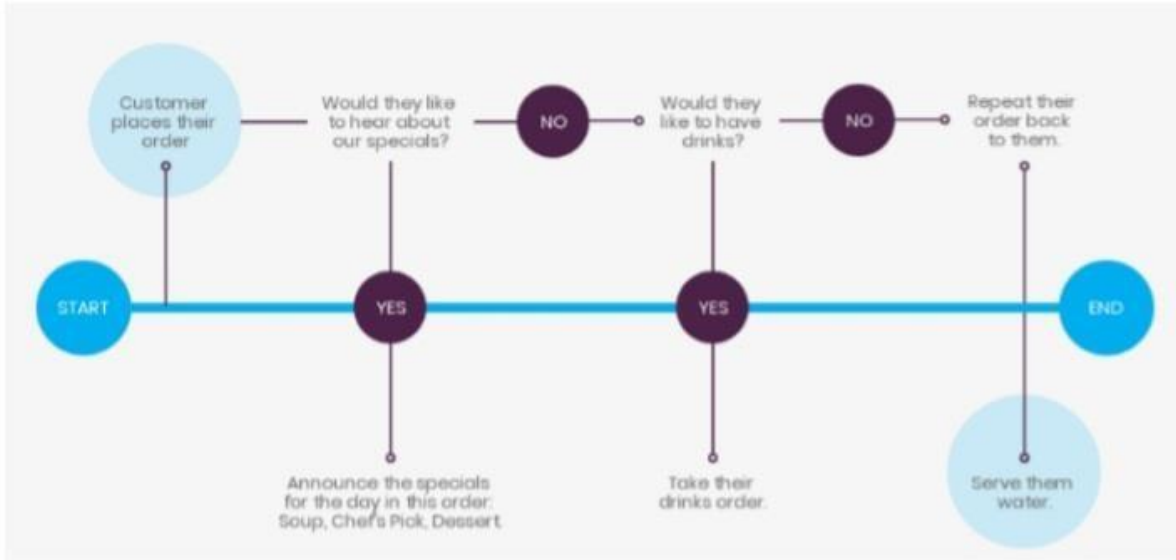


Fig.5.3.3 :Flowchart

## Data Flow Diagram

Level 0

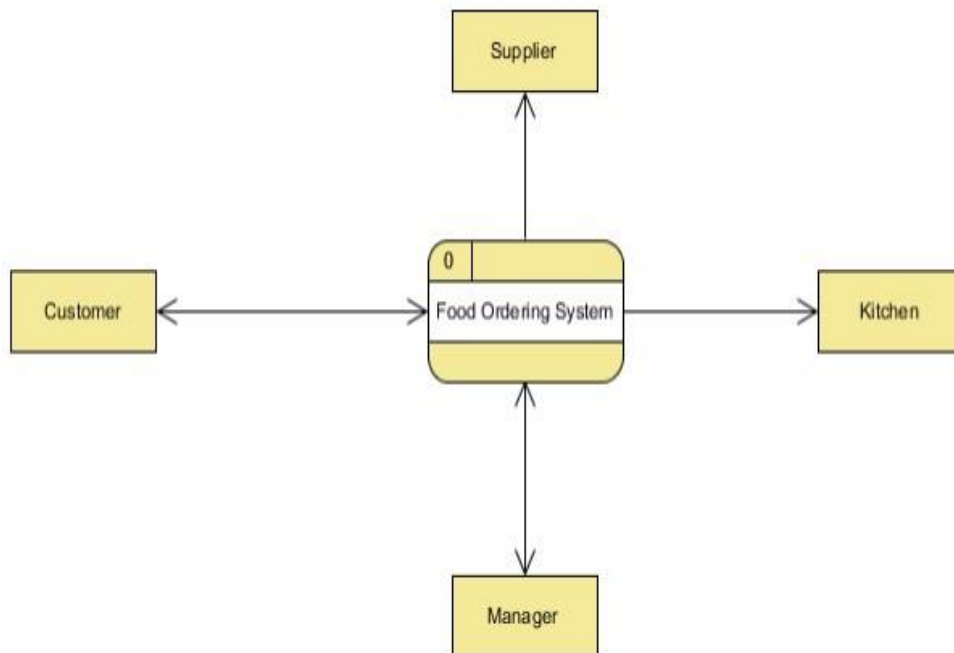


Fig.5.3.4 : DFD



## ER diagram

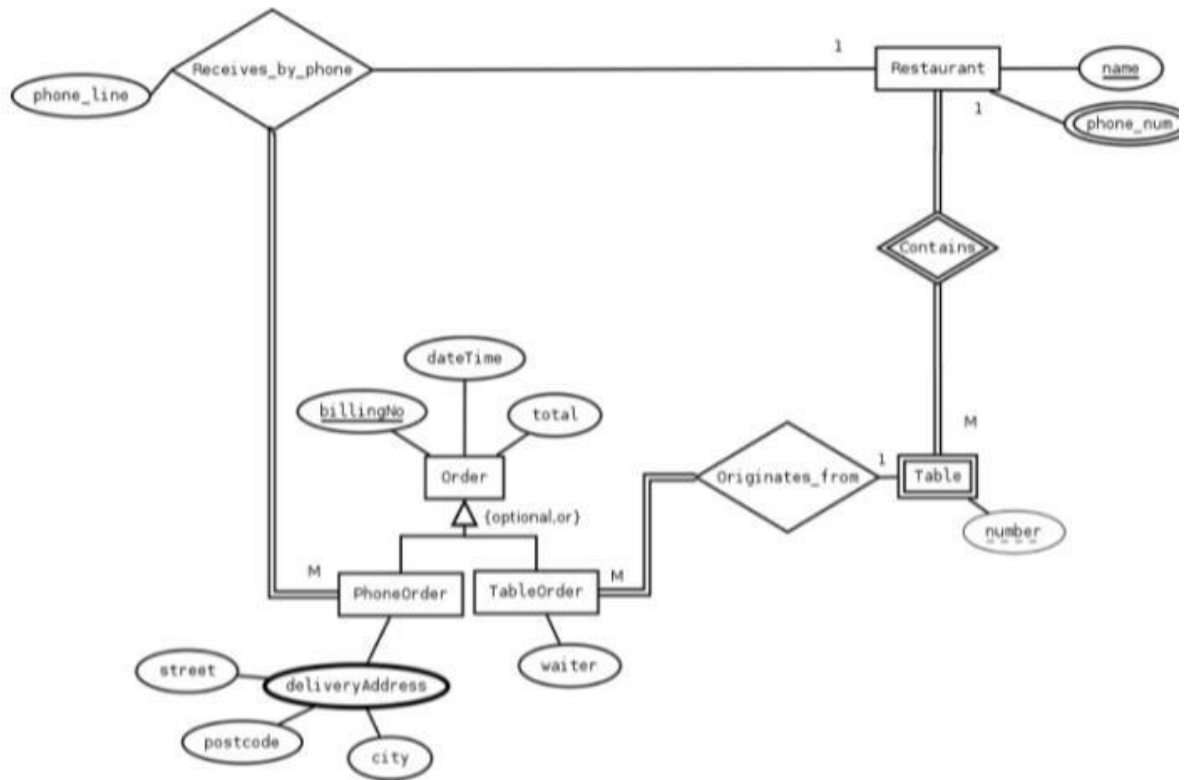


Fig.5.3.5: ER Diagram

## **CHAPTER-6**

### **Code Implementation**

#### **6.1 Implementation Environment**

Challenges identified for successful design and implementation of this project are dominated by: complexity, reliability/availability, transparent data access while respecting security. The project was a result of a group consensus. The team was having two members. The team was guided by project manager. The team structure depends on the management style of the organization, the no. of people in the team, their skill levels and the problem difficulty. After the completion of each module, a module testing was performed on each. When the development was completed, System testing was performed to test the system.

#### **6.2 Program/Module Specification**

- System GUI must be as simple and user friendly as any one can use it. At front side we implemented registration form for access the system
- Authentication is necessary to enter into the system only if one require to start his/her own project. This is required to prevent unauthorized access to the system
- If someone steals the password of the administrator or any regular user then he can able to change the database or misuse the system and can enter in restricted area so for this purpose system will provide encrypted password storage format in the database. Option to change the Password.
- A Session is maintained throughout the system when a particular user enters into the system. The Session is regularly checked whenever it is required.
- Proper validation is placed as and when it is required.

#### **6.3 Coding Standards**

Normally, good software development organization requires their programmers to adhere to some well defined and standard style of coding called coding standard.

##### **6.3.1 Variable Standards:**

We have used meaningful variables name.

##### **6.3.2 Comment Standards:**

- The comment should describe what is happening, how it is being done, what parameters mean, which global are used and which are modified, and any registration or bugs. The standards I have followed are:

- Every script should begin with a comment block, which describes the script's purpose; any argument used (if applicable), and return values (if applicable), inputs-outputs, and name of script.
- Comment may also be used in the body of the script to explain individual sections or lines of codes.
- It is also used to describe variable definition or declarations.
- Inline comments should be made with the `//` comment style and should be indented at the same level as the code described.
- For multiple line comments we write between `/* ...*/`.

# **CHAPTER-7**

## **Testing**

### **7.1 Testing**

Various parameters like implementation environment, program modules and coding standards are explained in previous chapter while this chapter is aimed to provide brief account of testing the software. There are two principal motives of testing the software

1. To rectify the error in execution
2. To check the viability of software

The testing ensures that the software is according to the required specification standards and performs the task meant for it. The testing is done by our in house employee that act as novice user and test the application with all possible way to find the bugs and error as well as check validation.

### **7.2 Testing Plan**

Testing is carried out at the following three stages:

- Design
- Implementation
- Coding

#### **7.2.1 Design Testing**

The design errors are to be rectified at the initial stage. Such errors are very difficult to repair after the execution of software.

#### **7.2.2 Implementation Testing**

The errors occurred at this stage can't be overlooked because such errors do not allow the further process.

#### **7.2.3 Coding Testing**

The coding procedure plays significant role in software designing. The improper coding of any software can generate inconsistent results. Such errors may occur due to incorrect syntax or false logic. If the errors at coding stage remain unnoticed may give rise to grave failure of the system

### **7.3 Testing Strategy**

A strategy for software testing integrates software test case design method into a well planned series of steps that result in the successful construction of the software. The strategy provides

the roadmap that describes the steps to be conducted as a part of testing, then these steps are planned and then undertaken, and how much effort, time and resource will be required.

We have tested our whole system using bottom up testing strategy.

- Bottom up testing involves integrating and testing the modules to the lower levels in the hierarchy, and then working up hierarchy of modules until the final module is tested.
- Bottom up testing strategy shows how actual testing is to be done with whole system but it does not show any detail about each module testing.
- For each module testing we have decided to test each lower level module with white box testing strategy.
- When all modules are tested successfully then I will move to one step up and continue with white box testing strategy.
- When all modules will be tested successfully then I will integrate those modules and try to test integrated system using black box testing strategy.

### **Why Black Box Testing in my Project?**

In my project whatever I have implemented was going to be tested by external guide Mr. DharmeshBhai Vaja without knowing our code, so there was a black box testing involve directly.

### **Why White Box Testing in my Project?**

During the project we were making the applications, we knew how it should proceed internally; we needed to Debugging also for testing our small functionalities.

### **Why interface Testing in our Project?**

We examined the code to be tested and explicitly list each call to an external component. In the system, standards tests for GUIs have been performed, which are as follows.

- Testing the screen control for its position and side.
- The position and the related labels for all controls were checked.
- Name of the form in system is given appropriately.
- All menu functions and sub functions were verified for correctness.
- Validations for all input were done.
- Whether the system prompts the user with appropriate message as and when invalid information is entered.
- All required fields aren't left blank.

## **7.4 Testing Method**

### **7.4.1 Unit Testing**

The unit testing is meant for testing smallest unit of software. There are two approaches namely bottom-up and top-down.

In bottom up approach the last module is tested and then moving towards the first module while top down approach reverses the action. In present work we opt for the first one.

The bottom up approach for the current project is carried out as shown in.

### **7.4.2 Integration Testing**

The integration testing is meant to test all the modules simultaneously because it is possible that all the modules may function correctly when tested individually. But they may not work altogether and may lead to unexpected outcome.

### **7.4.3 Validation Testing**

After the integration testing software is completely assembled as a package, interfacing error have been uncovered and corrected, and then validation testing may begin. Validation can be defined in many ways but a simple definition is what a validation succeeds when software functions in a manner that can be reasonably accepted by the user.

### **7.4.4 Storage Testing**

The database of the system has to be stored on the hard disk. So the storage capacity of the hard disk should be enough to store all the data required for the efficient running of the system.

## **7.5 Test Cases**

### **7.5.1 Purpose**

The purpose of this application is to reduce overhead in paper work and all the records are maintained such that the user as well as administrator can easily segment them into desired properties so it is easy for any novice user to have access to the application.

Another purpose is to make record of papers in database so it can be referred in future.

• **Login test:**

Test Case	Test Data	Test Result	Test Report
Blank Email Password	Email password	Invalid	Fill given detail
Invalid Email	Email:- user@mail.com	Invalid	Email is incorrect
Invalid Password	Password : user	Invalid	Password is incorrect
Valid Email and password	Email:- valid email id	Valid	Welcome to our site

**Table 7.5.1.1: Login test**

• **Register test:**

Test Case	Test Data	Test Result	Test Report
Blank name , email, password	Name: Email:- Password	Invalid	Fill given detail
Invalid name , email , password	Name:abc Email : user@gmail.com password:1253	Invalid	User alreadyregister
Password Valid name , email	name:-test email :- test@test.com Password:-test	Valid	Welcome message

**Table 7.5.1.2 :Signup test**

**Admin login test:**

Test Case	Test Data	Test Result	Test Report
Blank Email password	Email password	Invalid	Fill given detail
Invalid Email	Email:- user@mail.com	Invalid	Email is incorrect
Invalid Password	Password : user	Invalid	Password is incorrect
Valid Email and password	Email:- admin@gmail.com Password:-admin	Valid	Welcome to our site

**Table 7.5.1.3: Admin login test**

## **CHAPTER- 8**

### **Limitations and Future Enhancement**

#### **8.1 Limitations:**

Though we tried our best in developing this system but as limitations are mere parts of any system so are of our system. Some limitations of — OPD. Today portal are as under:

- Storage Capacity low
- User Cannot Effort

#### **8.2 FUTURE ENHANCEMENT:**

There is always a scope for enhancements in any developed system, especially when our nature of the project is iterative waterfall which allows us to rethink on the method of development to adopt changes in the project. Below mentioned are some of the changes possible in the future to increase the adaptability, and efficiency of the system.

- Improve Storage Capacity
- In Future User also can Effort



## **CHAPTER-9**

### **Conclusion and Discussion**

#### **9.1 Self Analysis And Project Viabilities**

As a trainee, working in Prompt ERP Limited company was a very enrich experience. During the period of training in Prompt ERP Limited, We have great experience of working as professional web developer. They will provide such a friendly environment such that many problems that we are not able to solve before this our mind gives the answer. Our logical power, database interaction power and dealing with the web development has improved largely. Our guide Nirali madam is so supported. Any query related to project, she is always there to guide us any time anywhere. Our college take the time to time update regarding to our project, such a environment make us more punctual

#### **9.2 Problem Encountered And Possible Solutions**

##### **9.2.1 Resource Availability**

It is the most important requirement that you have powerful and useful tools to develop proper and useful product. This is provided by our company.

##### **9.2.2 To get the knowledge of Company**

To work with the organization and fit ourselves in that situation was very much difficult task but with the support of my colleagues and faculty of college we manage to work and enriched our knowledge.

##### **9.2.3 Requirement Understanding**

The most important to build the user required and user friendly portal we have to understand what are the needs of users those should be fulfilled by our portal

## **CHAPTER-10**

### **REFERNCES**

#### **BOOKS:**

- Software Engineering by Roger Pressman

#### **WEB SITES:**

- [www.w3schools.com](http://www.w3schools.com)
- <http://stackoverflow.com/>
- [www.hmis.crmlive.in](http://www.hmis.crmlive.in)