

MAJOR PROJECT
ON
ONLINE FOOD ORDERING SYSTEM

Submitted in partial fulfill for the award of B.Tech.

In
Computer Science & Engineering
By

I.K. GUJRAL PUNJAB TECHNICAL UNIVERSITY
JALANDHAR
PUNJAB



SUBMITTED TO:-

Mr. Abhishek Bansal

SUBMITTED BY:-

Veerpal Kaur
Roll No.-2018062

**LUDHIANA GROUP OF COLLEGES,
CHAUKIMANN, LUDHIANA**

Table of Contents

TITLE	PAGE NO.
STUDENT'S DECLARATION	(i)
ACKNOWLEDGEMENT	(ii)
1. Introduction	1
2. System requirement and analysis	2-4
3. Present work	5-12
4. Result and Discussion	13
5. Snapshots	14-18
6. Database Structure	19-20
7. Coding	21-34
8. Testing and Implementation	35-38
9. Conclusion	39
10. Future Scope	40
11. References	41

STUDENT'S DECLARATION

I hereby certify that the work which is being done in the project entitled "**Online food ordering system**" by "**Veerpal Kaur**" as per course requirement of B.Tech. CSE submitted in the Department of Computer Science & Engg. at **Ludhiana Group of Colleges** is record of my/our own work, carried out during 7th Sem. under the supervision of "**Mr. Abhishek Bansal**". The matter presented in this report has not been submitted by me/us in any other institute for the award of B.Tech. Degree.

Veerpal Kaur
Signature of the Student

Abhishek Bansal *Abhishek Bansal*
Name and Signature of the SUPERVISOR

Signature of H.O.D. *[Red Signature]*

(i)

ACKNOWLEDGEMENT

I highly grateful to the **Dr. Rakesh Kumar** (Principal) **Ludhiana Group of Colleges** for providing this opportunity to carry out this project work.

I would also like to express a deep sense of gratitude to **HOD**, who guided during different phases of Project Work . Without the wise counsel and able guidance, it would have been quite difficult to complete this Project Work.

NAME & SIGNATURE OF STUDENT(s)

Veerpal Kaur

2018062

Computer Science & Engg.

7th Sem

CHAPTER 1

INTRODUCTION

1. Introduction about project

The Project is on Online Food ordering System. The project maintains three levels of users one is 'Administrator' and second is 'Customer' and third is 'vendor'. The Website has been designed so that the customer can order food online which is provided by the vendors like some hotels and everything managed by admin.

This Website has following facilities :

- ❖ User can register himself/herself.
- ❖ User can login.
- ❖ The Customer can give feedback online.
- ❖ Customer can Order/Cancel the food.
- ❖ Without authentication user cannot login into the Website.
- ❖ There is separate admin area, where the administration can perform following actions:
 - Add/Delete new Administrator
 - View the List of Administrators
 - Add/Delete a New vendors
 - View the List of vendors
 - View the list of orders.
 - Cancel any order
 - Update the status of order
 - Change Admin login credentials
 - More...
- ❖ The customer after login, has following facilities:
 - Order food.
 - View his/her order
 - Cancel any order he/she has placed
 - Change it's details
- ❖ The Customer can give feedback about the problem faced during transaction or about the any site related complaint.
- ❖ All frequently asked questions are answered on the site, like within how many days a customer will receive the item etc.

CHAPTER 2

SYSTEM REQUIREMENT & ANALYSIS

2.1 Choice of platform used

PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere.

Requirement Analysis: - The aim of the requirements analysis and specification. Phase is to understand the exact requirements of the customer and to document them properly. This phase consists of two distinct activities: Requirement gathering, Requirement specification.

System Design: - The goal of the design phase is to transform the requirements specified in the SRS document into a structure that is suitable for implementation in some programming language.

Coding: - The purpose of the coding of software development is to translate the software design into source code. Each component of the design is implemented as a program module

Implementation: - In this phase modules are integrated in a planned manner. The different modules making up a software product are almost never integrated at shot/integration is normally carried out incrementally over a number of steps . During each integration step the partially integrated system is tested and previously planned modules are added to it.

Testing: - The goal of testing is to ensure that the developed system conforms to its requirements laid out in the SRS document. System testing consists of three different kinds of testing activities: Alpha testing, Beta testing, Acceptance testing.

Maintenance: - Maintenance of a typical software product requires much more than the effort necessary to develop the product itself. Maintenance involves performing any one or more of the following three kinds of activities:

1. Correcting errors that were not discovered during the product development phase. This is called corrective maintenance.
2. Porting the software to work in a new environment. This is called adaptive maintenance.
3. Improving the implementation of the system, and enhancing the functionalities of the system according to the customer's requirements. This is called perfective maintenance.

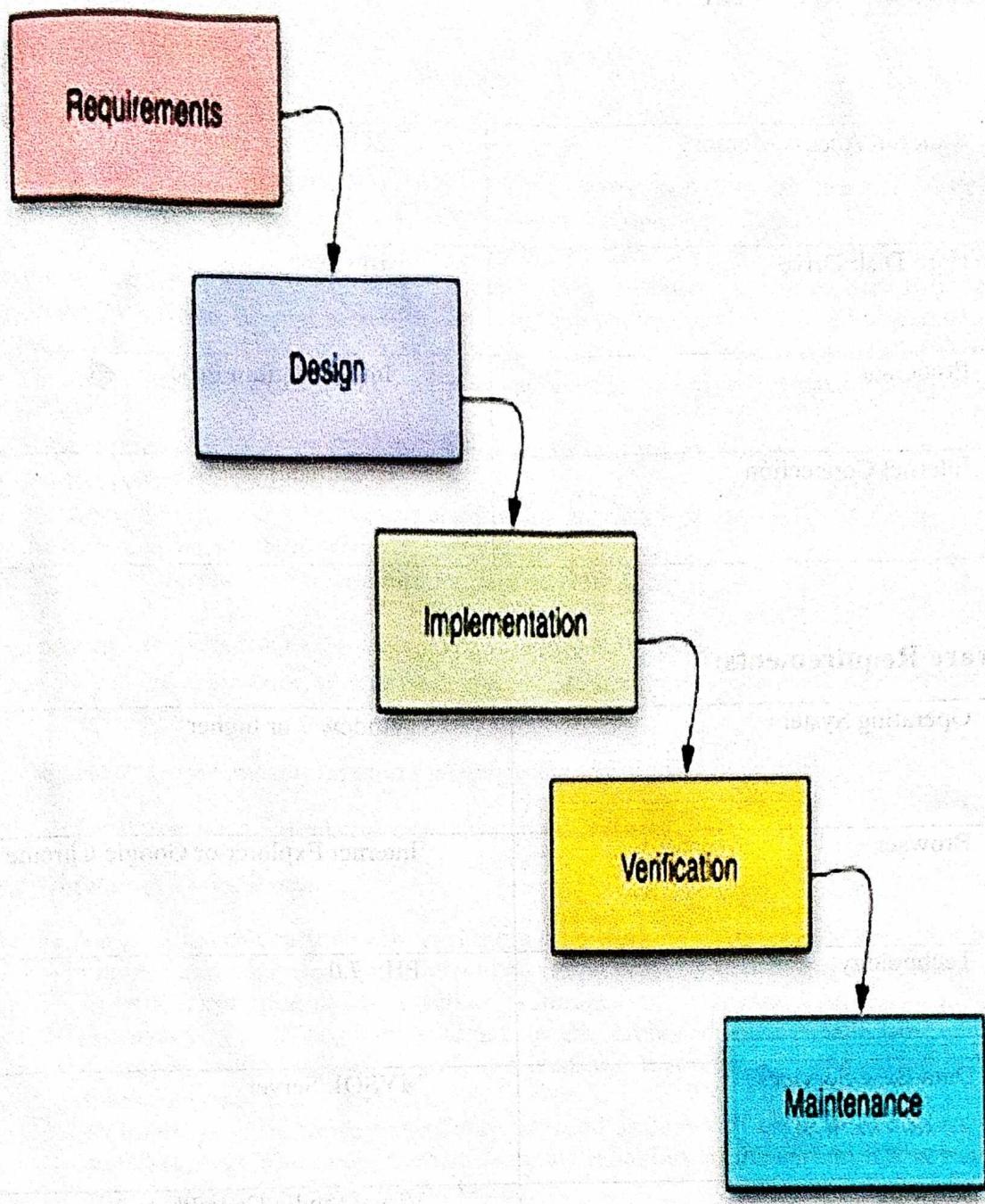


Fig: 2.1. System requirements and analysis

2.2 Hardware Requirements:

Random Access Memory	2GB
Hard Disk Drive	10 GB
Processor	Intel, Pentium Processor 4.
Internet Connection	Broadband

Software Requirements:

Operating System	Window 7 or higher
Browser	Internet Explorer or Google Chrome
Technology	PHP 7.0
Data Base Services	MYSQL Server
Software	Visual Studio Code/Photoshop

CHAPTER 3

PRESENT WORK

3.1 Problem Formulation

Currently the food Ordering company is doing everything through manual System and maintain various Manual Ledgers. All these works are done on papers. The work is done as follows: - Information about food and vendors is done by just writing the name, Price. Whenever the Customer comes up to buy some, no information is maintained. Bills are also generated by cutting manual receipts. They are calculating total orders done manual in the evening, they are not able to get information about how many rooms are remaining in the list. The Manual records are also destroyed after some time period to decrease the paper load in the office.

The Major problems faced by the food ordering services are as follows:

- ❖ No Record of Customers
- ❖ No Records about the vendors and food which are obsolete.
- ❖ Total No. of orders done so far
- ❖ No facility to search a food vendor on particular subject.

The problem formulation of the project is stated in the following:

- How the application should be developed, so that this can be made more efficient and improved.
- How to improve information for interpretation.
- How to make good relationship between teacher and student services.

3.2 Objectives of the Project

1. **To plan approach towards working:** - The working in the organization will be well planned and organized. The data will be stored properly in data stores, which will help in retrieval of information as well as its storage.
2. **Accuracy:** - The level of accuracy in the proposed system will be higher. All operation would be done correctly and it ensures that whatever information is coming from the centre is accurate.
3. **Reliability:** - The reliability of the proposed system will be high due to the above stated reasons. The reason for the increased reliability of the system is that now there would be proper storage of information.
4. **No Redundancy:** - In the proposed system utmost care would be that no information is repeated anywhere, in storage or otherwise. This would assure economic use of storage space and consistency in the data stored.
5. **Immediate retrieval of information:** - The main objective of proposed system is to provide for a quick and efficient retrieval of information. Any type of information would be available whenever the user requires.
6. **Immediate storage of information:** - In manual system there are many problems to store the largest amount of information.
7. **Easy to Operate:** - The system should be easy to operate and should be such that it can be developed within a short period of time and fit in the limited budget of the user.

3.3 Methodology

The Spiral model can become equivalent to other process model that would be helpful for any situations. When a cycle was finished, developer and customer can review the prototype that gives people who related to project an opportunity to go back earlier stages of cycle to redesign unsatisfied requirements.

For these reasons I am going to employ Spiral model for my project development. I implements Spiral model for my project development. Each cycle of Spiral model is a completed waterfall model and additional risk analysis stage in each cycle, which can reduce many risks during development.

This model incorporates strengths of other existing model and resolves their difficulties and its risk-driven approach accommodates most software project situations. It concentrated on implementing previous project idea to create a simple prototype to prove the idea is workable and feasible.

This project will discuss it briefly in requirement analysis part. This project report is concentrating on the discussion of second round of the model.

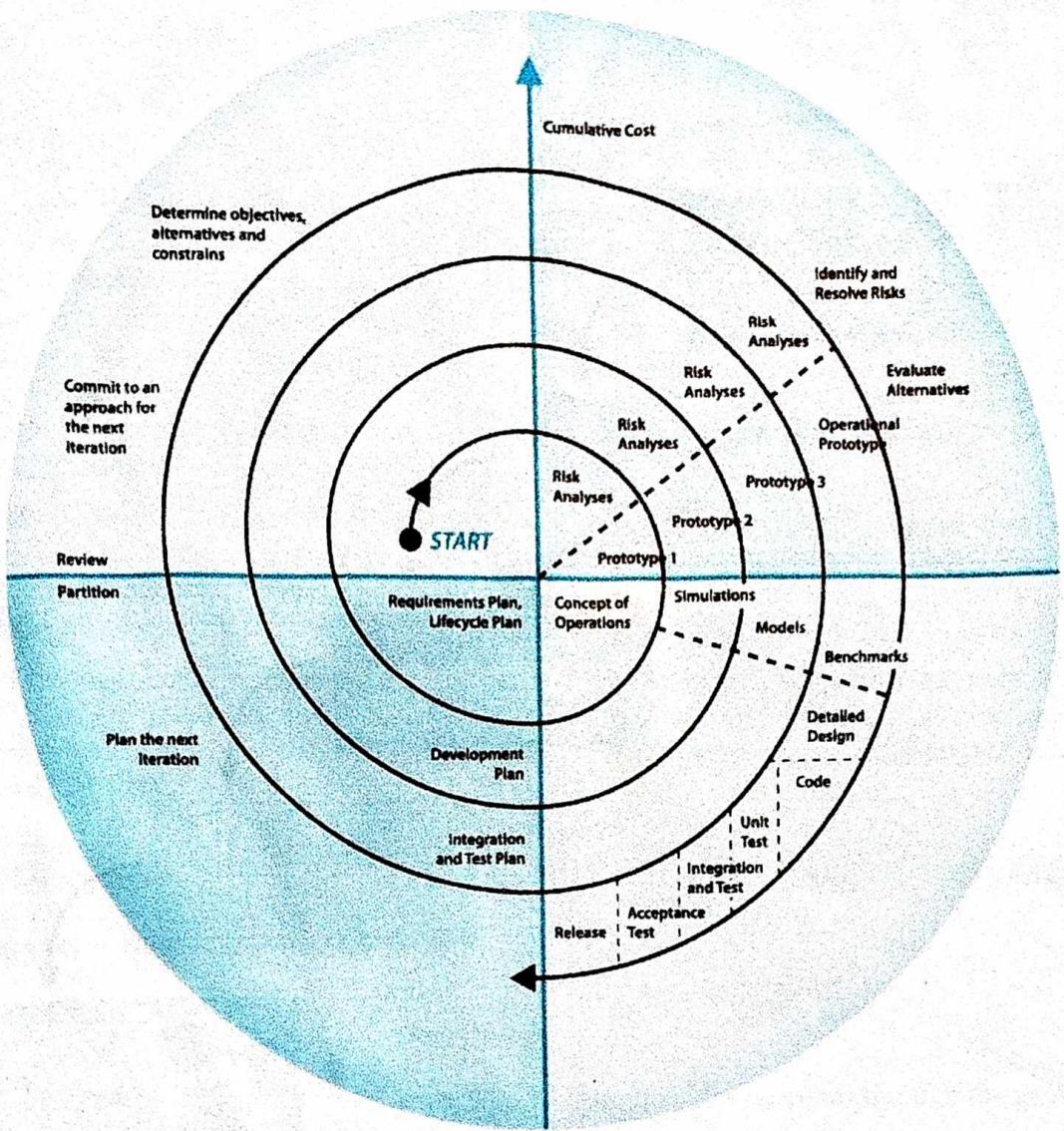


Fig: 3.1. THE SIPRAL MODEL

And said that second round of spiral model can convert to waterfall model in order to design a detailed prototype when many performance and user interface risks have been resolved during previous cycles. Therefore, the project report will follow waterfall model step by step to explain how to develop my second version multiplatform online banking web application prototype.

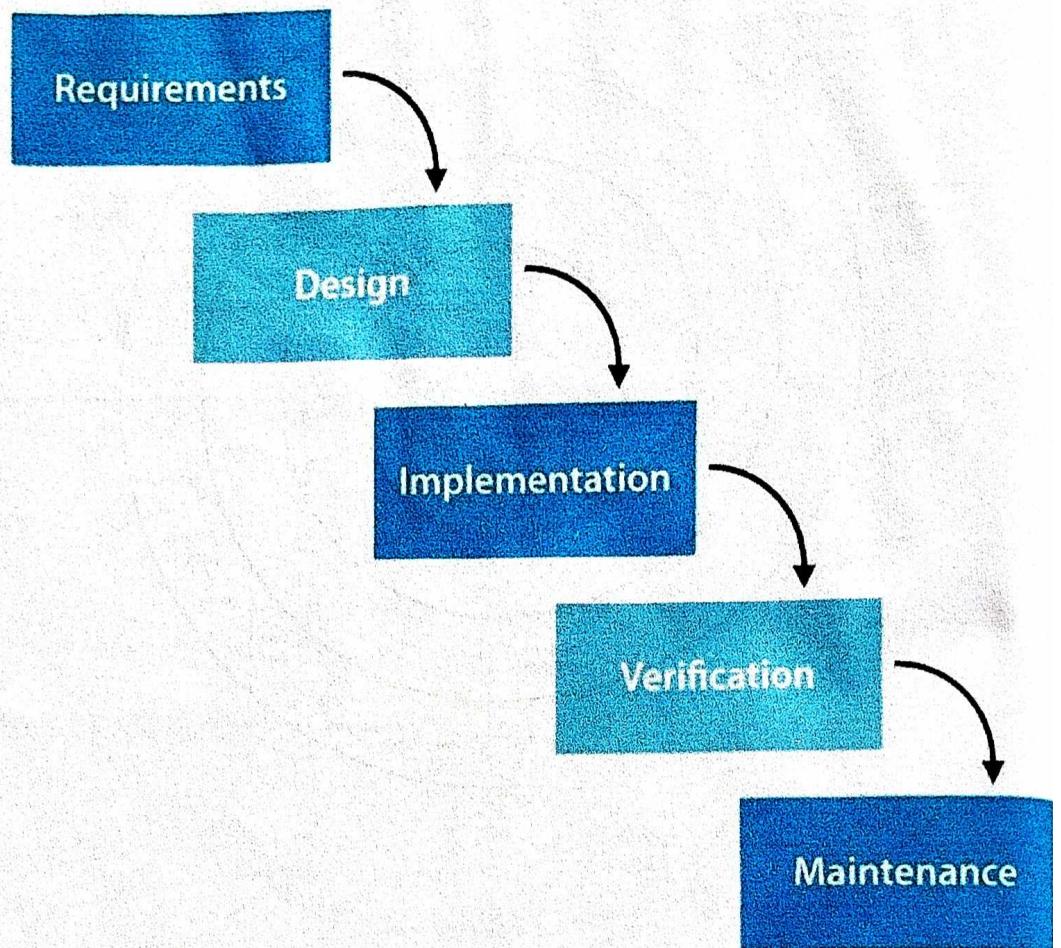


Fig: 3.2. THE WATERFALL MODEL

The project report will evaluate the project at the end. The evaluation includes existing online banking security countermeasures, previous project idea and modified solution against minimum requirements.

3.4 Design of Model

Data Flow Diagram

Data flow diagrams were proposed by Larry Constantine, the original developer of structured design, based on Martin and Estrin's "data flow graph" model of computation.

A two-dimensional diagram that explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output.

Individuals seeking to draft a data flow diagram must identify external inputs and outputs, determine how the inputs and outputs relate to each other, and explain with graphics how these connections relate and what they result in. This type of diagram helps business development and design teams visualize how data is processed and identify or improve certain aspects.

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modeling its process aspects. Often they are a preliminary step used to create an overview of the system which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

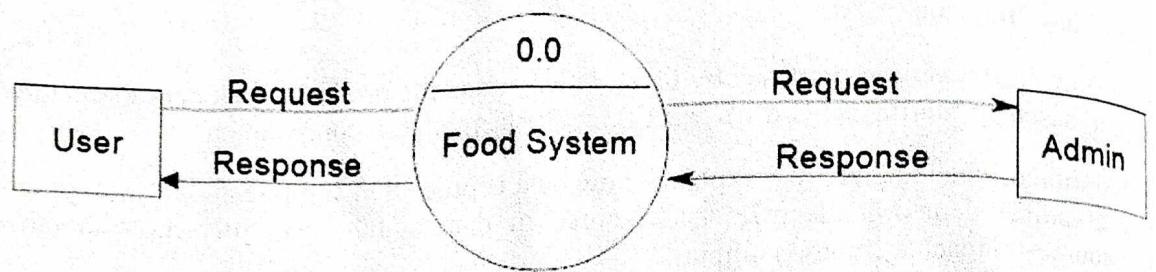
A DFD shows what kinds of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel.

DFD for Food Ordering System

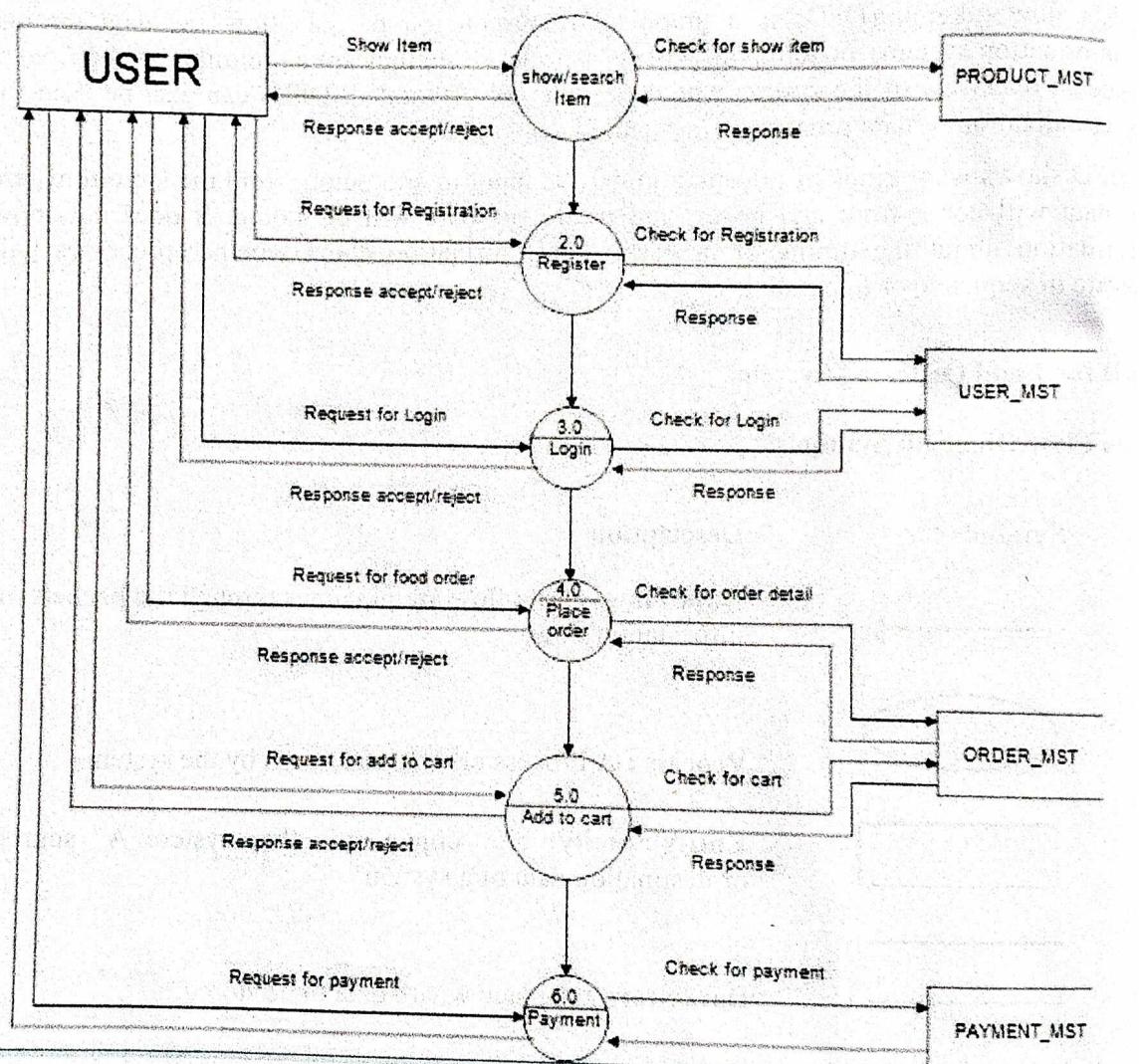
Data Flow Diagram Symbols:

Symbol	Description
	Data Flow – Data flow are pipelines through the packets of information flow.
	Process : A Process or task performed by the system.
	Entity : Entity are object of the system. A source or destination data of a system.
	Data Store : A place where data to be stored.

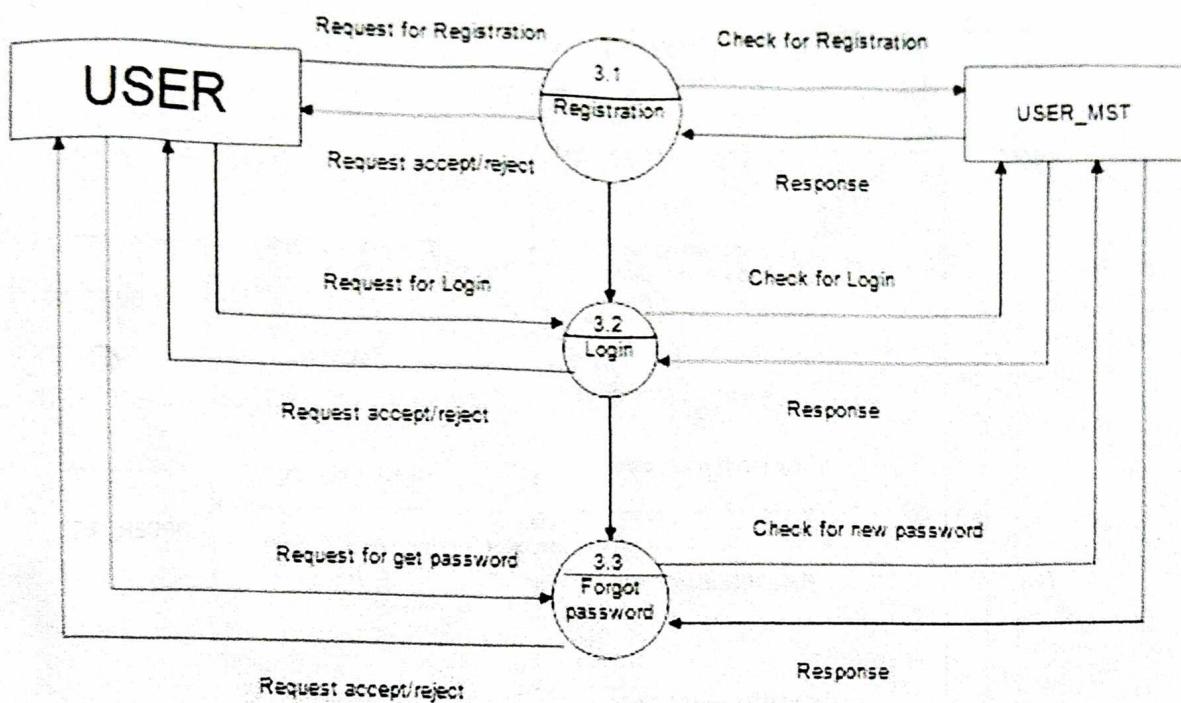
Context level DFD for Online food system



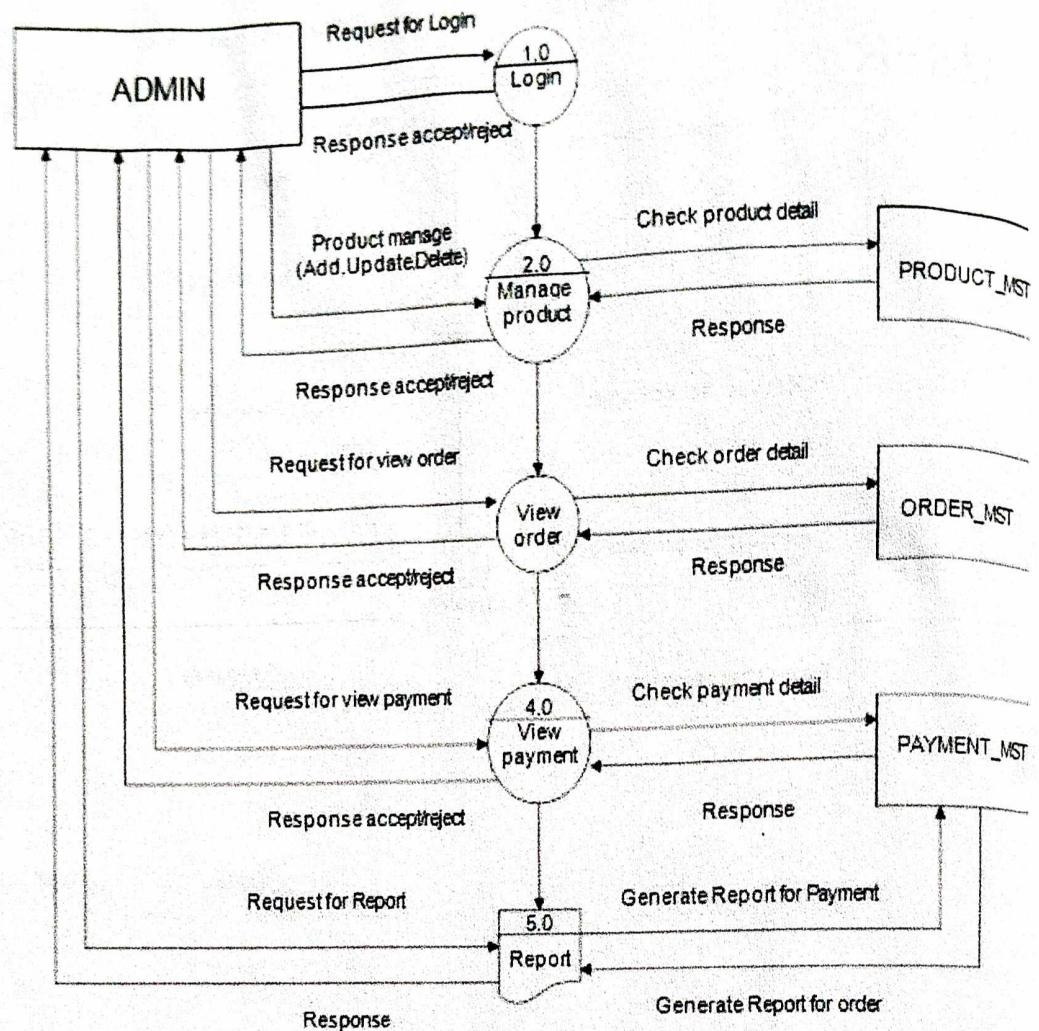
1st level DFD for User



2nd level DFD for user (3.0)



Admin side DFD for online food system



3.5 Applications

There is an area of information technology where automation of each and every activity is gaining importance.

1. Improved Manual System
2. Online System
3. It is fast, efficient and reliable.
4. It avoids data redundancy and inconsistency.
5. It is very user-friendly system.

CHAPTER 4

RESULT AND DISCUSSION

From a proper analysis of positive points and constraints on the component, it can be safely concluded that the product is a highly efficient component. This application is working properly and meeting to all user requirements. This component can be easily plugged in many other systems.

The online food ordering website is developed using PHP and MYSQL Server fully meets the objectives of the system for which it has been developed. The system has reached a steady state where all bugs have been eliminated. The system is operated at a high level of efficiency and all the teachers and user associated with the system understands its advantage. The system solves the problem. It was intended to solve as requirement specification.

CHAPTER 5

SNAPSHOTS

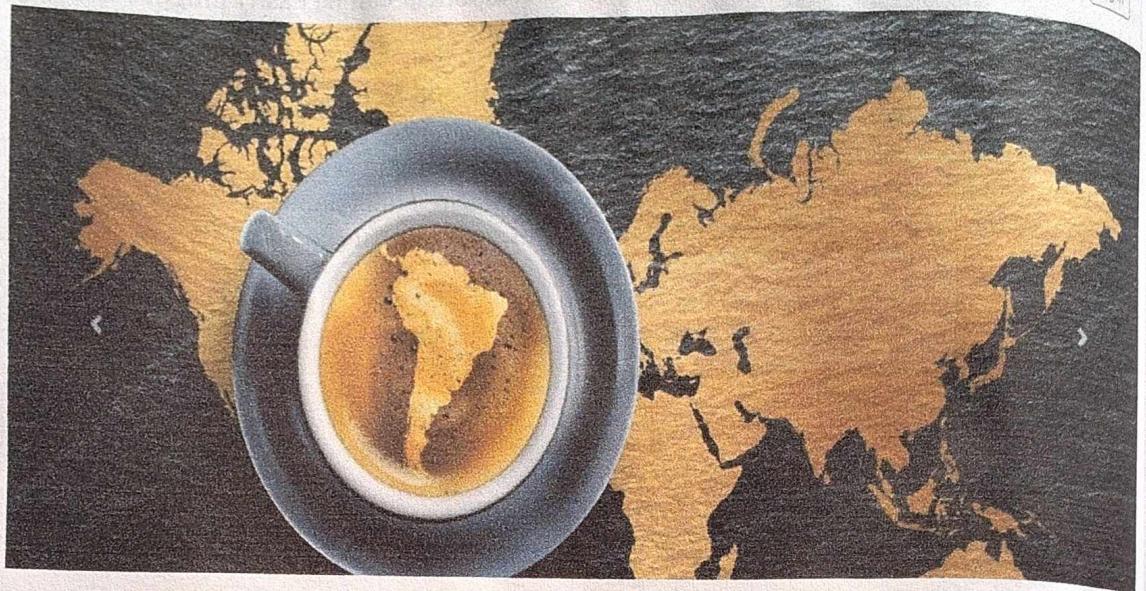
Index

FOOD HUNT

[Home](#) [About](#) [Services](#) [Contact](#)



Login



About Page

ABOUT US

About Us

Fresh food Ordering launched in 5 years ago. It has grown from a home project to one of the largest food aggregators in the world. We are present in 24 countries and 10000+ cities globally, enabling our vision of better food for more people. We not only connect people to food in every context but work closely with restaurants to enable a sustainable ecosystem.

Our Journey

Back in 2015, all it took was an idea to enable digital access to thousands of restaurant menus. Three passionate foodies who hated waiting in lines, drove around Delhi to collect menus from restaurants, scan them and put them online. Their idea has now grown into the vision that drives our team of 5000+ people everyday.

Contact



Name*

email*

Phone(optional)

Message*

00000-00000

Near Main Bazaar, Model Town, Ludhiana
(24*7 Days)

Services

Food Hunt

Home About Services Contact 0

SERVICES

There are many variations of passages of Lorem Ipsum available but the majority have suffered alteration in some injected humour.



Login Pages

Food Hunt

Home About Services Contact

Create New Account

Email address:

Password:

Login In

ADMIN SECTION

[Admin Login](#)

HOTEL LINKS

[Register On Food Hunt](#)

SignUp Page

Food Hunt

Home About Services Contact

[Register](#) [Log In](#)

Name:

Email Id:

Password:

Mobile:

Phone:

Address:

Cart

Food Hunt @Gurjot

Home About Services Contact  1 Log Out

[View Cart](#) [Account Settings](#) [Orders](#)

	shahi panner	RS 20	vegetable	Hotel Radison	Delete
Grand total			₹ 20	PROCEED TO CHECKOUT	

ADMIN SECTION

[Admin Login](#)

localhost/onlinefood/form/cart.php#viewitem

HOTEL LINKS

[Register On Food Hunt](#)
[Hotel Account Login](#)
[Add Foods](#)

Account Settings

Food Hunt @Gurjot

Home About Services Contact  1 Log Out

[View Cart](#) [Account Settings](#) [Orders](#)

Name

Gurjot

Email

gurjot2912@gmail.com

Mobile

9878375478

Password:

[Update](#)

Vendor Dashboard

Food Hunt & Hotel Radison

[Home](#) [About](#) [Services](#) [Contact](#)

[Log Out](#)

[Manage Products](#) [Add Products](#) [Account Setting](#) [Update Code](#) [Order Status](#)

food Image	food name	food Price	food cuisines	Payment Mode	Delete Item	Update Item Details
	malai Kofta	50	vegetable,veg	COD	Delete	Update
	shahi panner	20	vegetable	COD	Delete	Update
	chola kulcha	100	lunch	COD	Delete	Update

Admin dashboard

Food Hunt & Admin

[Home](#) [About](#) [Services](#)

[View All Products](#) [New Product](#) [Manage Settings](#) [Manage Vendors](#) [Order Status](#)

Hotel_Id	Food View	Food Cuisines	Hotel Name	Food Id	Remove Vendor
22		vegetable,veg	Hotel Radison	2	Remove
22		vegetable	Hotel Radison	3	Remove
22		lunch	Hotel Radison	4	Remove
23		Medium Size, fast food	Hotel Peccad	5	Remove
23		Fast food full size	Hotel Peccad	6	Remove
23		Fast food	Hotel Peccad	7	Remove

ADMIN SECTION

[Admin Login](#)

HOTEL LINKS

[Register On Food Hunt](#)

CHAPTER 6

DATABASE STRUCTURE

Database Name:dbfood

Table :tbadmin

+ Options		fld_id	fld_username	fld_password
<input type="checkbox"/>	Edit Copy Delete	1	admin	admin@123
<input checked="" type="checkbox"/>	Check all	With selected:	Edit Copy Delete	Export
<input type="checkbox"/>	Show all	Number of rows:	25	Filter rows: Search this table

Table:tbfood

+ Options		food_id	fldvendor_id	foodname	cost	cuisines	paymentmode	fldimage
<input type="checkbox"/>	Edit Copy Delete	2	22	malai Kofta	50	vegetable,veg	COD	1469258122-malai-kofta.jpg
<input type="checkbox"/>	Edit Copy Delete	3	22	shahi panner	20	vegetable	COD	Shahi-Panner-Recipe.jpg
<input type="checkbox"/>	Edit Copy Delete	4	22	chola kulcha	100	lunch	COD	maxresdefault.jpg
<input type="checkbox"/>	Edit Copy Delete	5	23	Pizza	100	Medium Size, fast food	COD	phut_0.jpg
<input type="checkbox"/>	Edit Copy Delete	6	23	Pizza Full	300	Fast food full size	COD	phut_0.jpg
<input type="checkbox"/>	Edit Copy Delete	7	23	burger	50	Fast food	COD	photo-1534790566855-4cb788d389ec.jpg

Table:tblcart

+ Options		fld_cart_id	fld_product_id	fld_customer_id
<input type="checkbox"/>	Edit Copy Delete	3	3	gurjot2912@gmail.com
<input checked="" type="checkbox"/>	Check all	With selected:	Edit Copy Delete	Export
<input type="checkbox"/>	Show all	Number of rows:	25	Filter rows: Search this table

Table:tblcustomer

	fld_cust_id	fld_name	fld_email	fld_mobile	password
<input type="checkbox"/>	4	Gurjot	gurjot2912@gmail.com	9878375478	12345
<input type="checkbox"/>	1	Harjit	customer1@gmail.com	7503515382	customer1
<input type="checkbox"/>	2	John	customer2@gmail.com	7503515386	customer2
<input type="checkbox"/>	3	Mohan	customer3@gmail.com	7503515383	customer3

Check all With selected:

Table:tblorder

	fld_order_id	fld_cart_id	fldvendor_id	fld_food_id	fld_email_id	fld_payment	fldstatus
<input type="checkbox"/>	1	1	21	1	customer3@gmail.com	50	Delivered
<input type="checkbox"/>	2	2	22	3	customer3@gmail.com	20	Out Of Stock

Table:tblvendor

	fldvendor_id	fld_name	fld_email	fld_password	fld_mob	fld_phone	fld_address	fld_logo
<input type="checkbox"/>	22	Hotel Radison	vendor1@gmail.com	vendor1	7503515386	114565457	Ludhiana	45386559
<input type="checkbox"/>	23	Hotel Picasso	vendor2@gmail.com	vendor2	7503515385	114565457	Chandigarh	45386559

Check all With selected:

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

CHAPTER 7

CODING

```
<?php  
session_start();  
  
include("connection.php");  
extract($_REQUEST);  
$arr=array();  
if(isset($_GET['msg']))  
{  
    $loginmsg=$_GET['msg'];  
}  
else  
{  
    $loginmsg="";  
}  
if(isset($_SESSION['cust_id']))  
{  
    $cust_id=$_SESSION['cust_id'];  
    $cquery=mysqli_query($con,"select * from tblcustomer where  
fld_email='$cust_id'");  
    $cresult=mysqli_fetch_array($cquery);  
}  
else  
{  
    $cust_id="";  
}
```

```

$query=mysqli_query($con,"select
tblvendor.fld_name,tblvendor.fldvendor_id,tblvendor.fld_email,
tblvendor.fld_mob,tblvendor.fld_address,tblvendor.fld_logo,tbfood.food_id,tbfood.foodname,tbfood.cost,
tbfood.cuisines,tbfood.paymentmode
from      tblvendor      inner      join      tbfood      on
tblvendor.fldvendor_id=tbfood.fldvendor_id;");
while($row=mysqli_fetch_array($query))
{
    $arr[]=$row['food_id'];
    shuffle($arr);
}

//print_r($arr);

if(isset($addtocart))
{
    if(!empty($_SESSION['cust_id']))
    {
        header("location:form/cart.php?product=$addtocart");
    }
    else
    {
        header("location:form/?product=$addtocart");
    }
}

if(isset($login))
{
    header("location:form/index.php");
}
if(isset($logout))
{
    session_destroy();
    header("location:index.php");
}

```

```

}

$query=mysqli_query($con,"select
tbfood.foodname,tbfood fldvendor_id,tbfood.cost,tbfood.cuisines,tbfo
od fldimage,tblcart fld_cart_id,tblcart fld_product_id,tblcart fld_custo
mer_id from tbfood inner join tblcart on
tbfood.food_id=tblcart fld_product_id
tblcart fld_customer_id='$cust_id'");

$re=mysqli_num_rows($query);
if(isset($message))
{
    if(mysqli_query($con,"insert
tblmessage(fld_name,fld_email,fld_phone,fld_msg)
values
('$nm','$em','$ph','$txt')"))
    {
        echo "<script> alert('We will be Connecting You
shortly')</script>";
    }
    else
    {
        echo "failed";
    }
}
?>
<html>
<head>
<title>Home</title>
<!--bootstrap files-->
<link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
      integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUhcWr7x
      9JvoRxT2MZw1T" crossorigin="anonymous">
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"

```

```
        integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzbzo5smXKp4YfRvH+8a
btTE1Pi6jizo" crossorigin="anonymous">
    </script>
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popp
er.min.js"
        integrity="sha384-
UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq
4sF86dIHNDz0W1" crossorigin="anonymous">
    </script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.
min.js"
        integrity="sha384-
JjSmVgyd0p3pXB1rRibZUAYoIlly6OrQ6VrjIEaFf/nJGzIxFDsf4x0x
IM+B07jRM" crossorigin="anonymous">
    </script>
<!--bootstrap files-->

<link      href="https://fonts.googleapis.com/css?family=Lobster"
rel="stylesheet">
    <link      rel="stylesheet"      href="path/to/font-awesome/css/font-
awesome.min.css">
    <link
            href="https://use.fontawesome.com/releases/v5.8.1/css/all.css"
            integrity="sha384-
50oBUHEmpQ+1IW4y57PTFmhCaXp0ML5d60M1M7uH2+nqUiv
zIebhndOJK28anvf" crossorigin="anonymous">
    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js"
></script>
    <link
            href="https://fonts.googleapis.com/css?family=Great+Vibes|Permane
nt+Marker" rel="stylesheet">
```

```
<script>
//search product function
$(document).ready(function() {

    $("#search_text").keypress(function() {
        load_data();

        function load_data(query) {
            $.ajax({
                url: "fetch2.php",
                method: "post",
                data: {
                    query: query
                },
                success: function(data) {
                    $('#result').html(data);
                }
            });
        }

        $('#search_text').keyup(function() {
            var search = $(this).val();
            if (search != "") {
                load_data(search);
            } else {
                $('#result').html(data);
            }
        });
    });

    //hotel search
    $(document).ready(function() {
        $("#search_hotel").keypress(function() {
```

```
load_data();

function load_data(query) {
    $.ajax({
        url: "fetch.php",
        method: "post",
        data: {
            query: query
        },
        success: function(data) {
            $('#resulthotel').html(data);
        }
    });
}

$('#search_hotel').keyup(function() {
    var search = $(this).val();
    if (search != "") {
        load_data(search);
    } else {
        load_data();
    }
});
});

});

</script>
<style>
</style>
</body>
background-image:url("img/main_spice2.jpg");
background-repeat: no-repeat;
background-attachment: fixed;
background-position: center;
}

ul li {
    list-style: none;
}
```

```

ul li a {
    color: black;
    font-weight: bold;
}

ul li a:hover {
    text-decoration: none;
}
</style>
</head>

<body>

<div id="result" style="position:fixed;top:300; right:500;z-index:3000;width:350px;background:white;"></div>
<div id="resulthotel" style=" margin:0px auto; position:fixed; top:150px;right:750px; background:white; z-index: 3000;"></div>
<nav class="navbar navbar-expand-lg navbar-light bg-light fixed-top">
    <a class="navbar-brand" href="index.php"><span style="color:green;font-family: 'Permanent Marker', cursive;">Food Hunt</span></a>
    <?php
        if(!empty($cust_id))
        {
        ?>
        <a class="navbar-brand" style="color:black; text-decoration:none;">i

```

```

        class="far fa-user">><?php echo $cresult['fld_name'];
?></i></a>
<?php
}
?>
<button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarResponsive"
aria-controls="navbarResponsive" aria-expanded="false" aria-
label="Toggle navigation">
<span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">

<ul class="navbar-nav ml-auto">

<li class="nav-item active">
<a class="nav-link" href="index.php">Home
</a>
</li>
<li class="nav-item">
<a class="nav-link" href="aboutus.php">About</a>
</li>
<li class="nav-item">
<a class="nav-link" href="services.php">Services</a>
</li>
<li class="nav-item">
<a class="nav-link" href="contact.php">Contact</a>
</li>
<li class="nav-item">
<form method="post">
<?php
if(empty($cust_id))
{
?>
<a href="form/index.php?msg=you must be log in
first"><span style="color:red; font-size:30px;"><i

```

```
        class="fa fa-shopping-cart" aria-
hidden="true">><span style="color:red;" id="cart"
            class="badge badge-
light">0</span></i></span></a>

        &nbsp;&nbsp;&nbsp;
        <button class="btn btn-outline-danger my-2 my-sm-0"
name="login" type="submit">Login
        </button>&nbsp;&nbsp;&nbsp;
        <?php
        }
        else
        {
        ?>
        <a href="form/cart.php"><span style=" color:green;
font-size:30px;"><i
            class="fa fa-shopping-cart" aria-
hidden="true"><span style="color:green;" id="cart"
            class="badge badge-light"><?php
if(isset($re)) { echo $re; }?></span></i></span></a>
        <button class="btn btn-outline-success my-2 my-sm-0"
name="logout" type="submit">Log
        Out</button>&nbsp;&nbsp;&nbsp;
        <?php
        }
        ?>
        </form>
    </li>

</ul>
</div>
</nav>
<!--menu ends-->
<div id="demo" class="carousel slide" data-ride="carousel">
    <ul class="carousel-indicators">
```

```
<li data-target="#demo" data-slide-to="0" class="active"></li>
    <li data-target="#demo" data-slide-to="1"></li>
    <li data-target="#demo" data-slide-to="2"></li>
</ul>
<div class="carousel-inner">
    <div class="carousel-item active">
        
        <div class="carousel-caption">
            <h3>Los Angeles</h3>
            <p>We had such a great time in LA!</p>
        </div>
    </div>
    <div class="carousel-item">
        
        <div class="carousel-caption">
            <h3>Chicago</h3>
            <p>Thank you, Chicago!</p>
        </div>
    </div>
    <div class="carousel-item">
        
        <div class="carousel-caption">
            <h3>New York</h3>
            <p>We love the Big Apple!</p>
        </div>
    </div>
</div>
```

```

        <a class="carousel-control-prev" href="#demo" data-
slide="prev">
            <span class="carousel-control-prev-icon"></span>
        </a>
        <a class="carousel-control-next" href="#demo" data-
slide="next">
            <span class="carousel-control-next-icon"></span>
        </a>
    </div>

<!--slider ends-->

<!--container 1 starts-->

<br><br>
<div class="container">
    <div class="row">
        <?php
            $food_id=$arr[0];
            $query=mysqli_query($con,"select
tblvendor.fld_email,tblvendor.fld_name,tblvendor.fld_mob,
tblvendor.fld_phone,tblvendor.fld_address,tblvendor fldvendor_
id,tblvendor.fld_logo,tbfood.food_id,tbfood.foodname,tbfood.cost,
tbfood.cuisines,tbfood.paymentmode,tbfood.fldimage      from
tblvendor inner join
tbfood
on
tblvendor.fldvendor_id=tbfood.fldvendor_id ");
            while($res=mysqli_fetch_assoc($query))
            {
                $hotel_logo=
"image/restaurant/".$res['fld_email']."/".$res['fld_logo'];
                $food_pic=
"image/restaurant/".$res['fld_email']."/foodimages/".$res['fldimage'];
            }
        </?php>
    </div>
</div>

```

```

?>
<div class="col-sm-6" style="margin-bottom:20px;">
    <div class="row">
        <div class="col-sm-2">
            <img src=<?php echo $hotel_logo; ?>
                class="rounded-circle" height="50px" width="50px"
                alt="Cinque Terre">
        </div>
        <div class="col-sm-5">
            <a href="search.php?vendor_id=<?php echo
                $res['fldvendor_id']; ?>"><span
                    style="font-family: 'Miriam Libre', sans-serif;
                    font-size:28px;color:#CB202D;">
                <?php echo $res['fld_name']; ?></span></a>
            </div>
            <div class="col-sm-3">
                <i style="font-size:20px;" class="fas fa-rupee-
sign"></i>&ampnbsp<span
                    style="color:green; font-size:25px;"><?php echo
                $res['cost']; ?></span>
            </div>
            <div class="col-sm-2">
                <form method="post">
                    <div class="col-sm-2" style="text-
align:left;padding:10px; font-size:25px;"><button
                        type="submit" name="addtocart"
                        value=<?php echo $res['food_id']; ?>" )"><span
                            style="color:green;"><i class="fa fa-
shopping-cart"
                                aria-
                            hidden="true"></i></span></button></div>
                    <form>
                        </div>
                </div>
                <div class="row">
                    <div class="col-sm-12">

```

```
  
</div>  
</div>  
  
<div class="row">  
    <div class="col-sm-6">  
        <span>  
            <li><?php echo $res['cuisines']; ?></li>  
        </span>  
        <span>  
            <li><?php echo "Rs ". $res['cost']; ?>&nbsp;for  
1</li>  
        </span>  
        <span>  
            <li>Up To 60 Minutes</li>  
        </span>  
    </div>  
    <div class="col-sm-6">  
        <h3><?php echo "(" . $res['foodname']. ")" ?></h3>  
    </div>  
    </div>  
  
</div>  
<?php  
}  
?>  
  
</div>  
</div>  
  
<!--footer primary-->  
  
<?php  
    include("footer.php");
```

?>

</body>

</html>

CHAPTER 8

TESTING AND IMPLEMENTATION

7.1 Testing

The goal of testing is to ensure that the developed system conforms to its requirements laid out in the SRS document. This phase is the important part of System Development Life Cycle. Testing is the process of executing the programs with the intention of finding out errors. During the process, the program to be tested is executed with set of cases and the output of program and the test cases is evaluated to determine if the program is performing as expected. Testing makes a logical assumption that if all the parts of the module are correct the goals will be successfully achieved.

Testing includes:

1. Positive Testing: Positive testing is making sure that new program do intact process. Certain transactions according to specifications.
2. Negative Testing: Negative testing is to test those transactions, which are not according to the specifications.

Testing should begin "in the small" and progress towards testing "in the large". The first tests planned and executed generally focus on individual components as testing progress focus shift in an attempt to find errors in integrated cluster of components and ultimately in the entire system. The different levels of testing attempt to detect different types of faults. The basic levels are:

Unit Testing

The testing of individual modules is yet to be completed during the design phase itself. Creating test data at the time of coding and necessary change tests each module and there on to make sure the module is working satisfactorily. Modules testing have a goal of discovering errors in the individuals' modules of the system. Modules are tested in the isolation from each other in an artificial environment. Keeping this in mind various modules and sub-modules will be tested as a single unit by checking the outputs and reports, menus etc will be carried out separately.

Integration Testing:

The next level of testing is often called integration testing. In this, many unit testing modules are compiled into sub-systems, which are then compiled. The goal here is to see if the module can be integrated properly. Hence, the emphasis on the interface between the modules.

System Testing:

The system testing is executing a program to check logic changes made in it and within the intention of finding errors making the program fail. Effective testing does not guarantee reliability. Reliability is designed considerations.

Acceptance Testing:

Acceptance testing is running the system by the live data by the actual user . Testing here focus on the external behavior of the system, the internal logic of the system is not emphasized.

Error Guessing: The concept of error guessing is an informed, heuristic techniques utilized to develop test cases for exposing errors. Areas of program that are likely to contain errors are identified and test cases are specified to test the condition.

System testing consists of three different kinds of testing activities:

1. Alpha testing
2. Beta testing
3. Acceptance testing

Beta Testing: - It is conducted at the end users site. If the software is product then Beta testing will be done. If the testers and model customers combined test the software in customer site, then it is called Beta testing. Beta Testing: Whole application is tested by end-user.

Beta Testing conducted at one or more customer sites by end user of software. Live application environment cannot be controlled by developer. Customer records all problems encountered and reports to developer at regular intervals. Beta testing is where the test is to be conducted in customer presence.

In a Product based company:

Alpha Testing defines as the testing which is done at the developer's site by the testers.

Beta testing is the testing done by the tester by generating the customer's environment.

In a Service based company:

Alpha testing is the testing conducted by customers at developer's site.

Beta testing is the process of giving the product to customers and let them do the testing at their environment.

Black Box Testing:

Black box testing is essentially "knowing the internal working of the product, test can be conducted to ensure all components perform according to the specification and all the components have been adequately exercised".

White Box Testing:

White box testing is essentially "knowing the internal working of the product, test can be conducted to ensure all components perform according to the specification and all the components have been adequately exercised".

Quality and reliability:

The quality and reliability of the software depends on the result of the phases in the system development. The software quality includes the modularity, good documentation, user friendliness, and maintainability etc. the software reliability depends on the less chances of the failures and enhancements in the system.

As far as this system is concerned, it tries to make the system user friendly as possible where a user can handle a system very comfortably. The system was designed in various modules and used menu driven, so it is easy to understand the system during the use.

The goals accomplished in designed/ developing of this system are: User Friendly Interfacing: Since main interaction of the system has to be with the user, the user interface should be attractive and meaningful.

Minimum Effort:

It ensures that very effort will be required to start the data entry and the generation and modifying the information about various users of the system.

Flexibility:

It provides maximum flexibility to the administrator in maintaining and modifying the information about various users of the system.

Security:

Since the information entered is of vital importance to the organization and to be entered only once, provision will be made to allow only the authorized user to manipulate the data.

Speed:

The system will be fast enough to give the user of the system the feel of using online web system.

7.2 Implementation

In this phase modules are integrated in a planned manner. The different modules making up a software product are almost never integrated at shot/integration is normally carried out incrementally over a number of steps. During each integration step the partially integrated system is tested and previously planned modules are added to it.

The Test Implementation stage is the process of implementing the test in an automated executable that can be hosted on a production test station and drives the test sequences on the DUT and collects the resulting test data. This stage traditionally involves software development done by the Test Engineer or by a separate software team that may be in-house or out-sourced.

The system or system modifications are installed and made operation in a production Environment. The phase is initiated after the system has been tested and accepted by the user. This phase continues until the system is operating in production in accordance with the defined user requirements.

Implementation of a new computer system to replace an existing one, this is usually a difficult conversion. The problems encountered are: converting files, training user, creating accurate files and verifying printouts for integrity. If not properly planned, there can be many problems. Some large computer systems have taken as long as a year to convert.

The implementation of a modified application to replace an existing one using the same computer. This type of conversion is relatively easy to handle, provided there are no major changes in this file.

7.3 Post Implementation

A Post-Implementation is an assessment and review of the completed working solution. It will be performed after a period of live running; some time after the project is completed.

There are three purposes for a Post-Implementation:

- To ascertain the degree of success from the project, in particular, the extent to which it met its objectives, delivered planned levels of benefit, and addressed the specific requirements as originally defined.
- To examine the efficacy of all elements of the working business solution to see if further improvements can be made to optimize the benefit delivered.
- To learn lessons from this project, lessons which can be used by the team members and by the organization to improve future project work and solutions.

The problems encountered are converting files, training users, creating accurate files and verifying printouts for integrity. Implementation of a new computer system to replace an existing one. This is usually a difficult conversion. If not properly planned, there can be many problems. Some large computer systems have taken as long as a year to convert.

A Post-Implementation should be scheduled some time after the solution has been deployed. There is often a difference of opinion as to who should perform the Post-Implementation. Usually, members of the project team will want to complete the review as a natural extension of their responsibility to deliver optimum benefit from the solution. They understand what was required, what was changed, how it was achieved, how things are supposed to work, how to fix problems, etc.

The project team might then reconvene to consider that input and also to examine how to generate further value from the solution. This is intended to be an assessment and review of the final working solution. There should have been at least one full processing and reporting cycle completed. It should not be performed while the initial snags are still being dealt with or while users are still being trained, coached and generally getting used to its operation.

CHAPTER 9

CONCLUSION

8.1 Conclusion

The Online food ordering system is a great improvement over the manual system using case fields and paper. The computerization of the system has sped up the process. In the current system, the front office managing is very slow. This website was thoroughly checked and tested with dummy data and thus is found to be very reliable. The Website takes care of all the requirements of an average food ordering system and is capable to provide easy and effective storage of information related to people come to order food.

It generates test reports like list of food ordered, orders which are pending etc. The system also provides the facility of backup as per the requirement.

CHAPTER 10

FUTURE SCOPE

The food industry is the one industry which will never become outdated. Human beings will always crave for food, looking for new food trends and tasty cuisines. The most significant disruption in the food industry has been the introduction of food delivery and ordering apps. Future of Online Food Ordering is Huge!

Online food ordering platforms have already disrupted the food ordering and delivery process in restaurants. People no more have to shout for their orders in a crowded restaurant, or especially go out to eat at a restaurant.

Online food ordering platforms also prevent missed orders due to busy phone lines or a lack of resources to monitor the phone. The face of the restaurant industry has shifted from the traditional dine-in culture to takeaways, online ordering, and home deliveries. Restaurants are quickly incorporating mobile food ordering apps in their restaurant management systems to streamline the entire order taking process.

Furthermore, online food ordering system for restaurants is a trend which is growing at a fast pace. Major Restaurants today just for the convenience factor and increasing revenue are listing their services even online.

Choosing the best food ordering platform for your restaurant business is a tough task! This is due to the fact that adoption for mobile technology has increased in leaps and bounds in the world. Phone orders and booking rates have declined these days extraordinarily.

Mobile apps developed to provide food ordering dedicatedly is now a raging trend for all mobile users. With more than 2 billion mobile users, an app is today the best way to promote the restaurant.

REFERENCES

1. The complete reference PHP
2. PHP and MYSQL Create-Modify-Reuse
3. PHP Projects for Beginners
4. PHP and MYSQL for Begineers
5. www.w3schools.com
6. www.php.net
7. www.tutorialspoint.com/php/
8. www.codeacademy.com
9. <https://www.mysql.com/>
10. www.meeraacademy.comdfd-diagram-for-online-food-ordering-system/