

ONLINE CATERING MANAGEMENT SYSTEM

BY

NASIR AHMAD

ID: 143-25-423

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Masters of Science in Computer Science and Engineering

Supervised By

Dr. Sheak Rashed Haider Noori
Associate Professor and Associate Head
Department of CSE
Daffodil International University



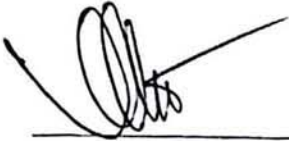
DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH

DECEMBER 2018

APPROVAL

This Thesis titled “Online Catering Management System”, submitted by Nasir Ahmad, ID No: 143-25-423 to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of M.Sc in Computer Science and Engineering and approved as to its style and contents.

BOARD OF EXAMINERS



Dr. Syed Akhter Hossain

Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman



Dr. Md. Ismail Jabiullah

Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Md Tarek Habib

Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Mohammad Shorif Uddin

Professor

Department of Computer Science and Engineering
Jahangirnagar University

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Dr. Sheak Rashed Haider Noori, Associate Professor and Associate Head, Department of CSE Daffodil International University**. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Dr. Sheak Rashed Haider Noori
Associate Professor and Associate Head
Department of CSE
Daffodil International University

Submitted by:

Nasir Ahmad
ID: - 143-25-423
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty Allah for His divine blessing makes us possible to complete this project successfully.

We fell grateful to and wish our profound our indebtedness to **Dr. Sheak Rashed Haider Noori, Associate Professor and Associate Head**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of computer science and engineering to carry out this project .His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Md. Tushar Chowdhury, Wahiduzzaman Rocky, and Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

This project is on “**Catering Management System**” developed for a catering and delivery service provider company called “Lunch BD”. Processes associated with their catering service always been a manual process which requires a lot of paperwork and could sometimes be a cumbersome and tiring task for the personnel in charge. The manual process sometimes can be time wasting, error prone, inefficient and a difficult task. In this particular situation there are times when requests and feedbacks are received from customer regarding the service that has to go through phone conversation and the responsible staff must keep notes about them manually and fulfill accordingly. Sometimes, customers might be in vacation or not available. In such situation the delivered food is wasted and cause financial loss to the company. Also, some customer has personal preferences about the menu which is not easy to handle for them in daily basis.

The aim of this project to considering all of these limitations and to facilitate the process of their service by developing a web application [1] that would allow both customers and the company engage in more productive activities. This application would provide a user interface for individual user to manage their account online that can be accessed from any devices including desktop computer, laptop, tablet, mobile phone. It would help them to keep in touch and stay synced with the latest information about the services even on the way as well as manage everything right from their smart phone or computer. Thus, this application would reduce/eliminate error experienced in manual works, save time, reduce cost as well as keep everything organized.

At first, the requirements and scope of this project is analyzed, designed, implemented and then it’s tested in various stages and found to be work successfully.

TABLE OF CONTENTS

CONTENS	PAGE
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
CHAPTER	
CHAPTER 1: INTRODUCTION	1-5
1.1 Project Context	2
1.2 Project motivation	2
1.3 Project objectives	2
1.4 Expected outcome	4
1.5 Report layout	4
CHAPTER 2: BACKGROUND	6-9
2.1 Introduction	6
2.2 Related works	7
2.3 Scope of the project	8
2.4 Challenges	8

CHAPTER 3: REQUIREMENTS	10-16
3.1 Requirement elicitation	10
3.2 Analysis of stakeholders	11
3.3 Requirement analysis	11
3.4 Requirement specification	12
3.4.1 Functional requirements	12
3.4.2 Nonfunctional requirements	13
3.5 Use Case Model	13
3.6 Logical Data Model	15
3.7 Design Requirements	16
CHAPTER 4: DESIGN SPECIFICATION	17-33
4.1 Frontend Design	17
4.1.1 Home Page Design	18
4.1.2 Login and Registration Forms	19
4.1.3 Static Pages	20
4.2 Backend Design	20
4.2.1 My Account Page	21
4.2.2 Personalization Page	21
4.2.3 Vacation Mode Page	22
4.2.4 Customer Management Page	23
4.2.5 Meals Management Page	23

4.2.6 Ingredients Management Page	25
4.2.7 Schedule Management Page	26
4.2.8 Delivery Management Page	27
4.3 Interaction Design and UX	29
CHAPTER 5: IMPLEMENTATION AND TESTING	34-49
5.1 Implementation of Database	34
5.2 Implementation of the Design	35
5.2.1 CakePHP Framework	35
5.2.3 Utilization of MVC architecture	36
5.2.4 Development process	36
5.2.5 Coding standards	37
5.3 Testing	41
5.3.1 The testing process	41
5.3.2 Requirement Traceability	42
5.3.3 Testing Schedule	42
5.3.4 Testing Strategy	42
5.4 Test Results and Reports	43
CHAPTER 6: CONCLUSION AND FUTURE SCOPES	50
REFERENCES	51

LIST OF FIGURES

FIGURES	PAGE NO
Figure 3.6.1 Use case model of catering management system	14
Figure 3.7.1: Entity Relationship model of the Catering Management System	15
Figure 4.1.1 Web applications	17
Figure 4.1.1.1: Homepage design of the website	18
Figure 4.1.2.1: User login form design	19
Figure 4.1.2.2: User Registration form design	19
Figure 4.1.3.1: Design of the informative pages	20
Figure 4.2.1.1: My account page	21
Figure 4.2.2.1: Meal personalization page	22
Figure 4.2.3.1: Setting up vacation mode	22
Figure 4.2.4.1: Customer management page	23
Figure 4.2.5.1 Meal Management Page	24
Figure 4.2.5.2 Meal management page	24
Figure 4.2.6.1 Manage list of ingredients	25
Figure 4.2.6.2: Add a new ingredient page	26
Figure 4.2.7.1 List of schedules on schedule management page	26
Figure 4.14 Add/update schedule information	27
Figure 4.2.8.1 Delivery management page	28
Figure 4.3.1 Sequence diagram of registration process	29
Figure 4.3.2 Sequence diagram of login process	29
Figure 4.3.3 Sequence diagram of account information updating process.	30
Figure 4.3.4 Sequence diagram of setting up personalized meal.	30
Figure 4.3.5 Sequence diagram of updating vacation status.	31
Figure 4.3.6 Sequence diagram of viewing upcoming delivery	31
Figure 4.3.7 Sequence diagram of viewing delivery history	32
Figure 4.3.8 Sequence diagram to view financial records	32

Figure 4.3.9 Sequence diagram of meals management page.	33
Figure 4.3.10 Sequence diagram of delivery management page.	33
Figure 5.1: E-R diagram of the database	34
Figure 5.2.3.1 MVC architecture and the flow	36
Figure 5.3.1: Testing steps	41

CHAPTER 1: INTRODUCTION

"Online Catering Management System" is a web based application. The general goal of this system is to automate day to day activity of a catering service provider and to facilitate their business management. Catering is a kind of business that serves and delivers meal in daily basis. This system is developed to provide service facility to them and also to the customer. This catering management system can be used by employees in a catering service company to handle the clients, manage requests, deliver meals, manage delivery schedules, view reports and statistics etc.

The main objective is to build a system to provide all these facilities online that can be used managed by the staffs as well as the customers from any devices. Through this system they would be able to manage customer, requests & their feedbacks, meals, deliveries, customer's transactions and reports. Basically, there would be three types of users that can access this portal such as Super Administrator, Administrator and the Customers. Administrator can manage the whole system that includes customer management, payment & transaction management, report management, meal and ingredients management, delivery and delivery schedule management. In customer management interface, the administrator can view and edit the information of the customers that are subscribed to their catering service. Also, he/she would be able to manage the meals and their delivery schedules. Each meal has its name, description, photo, price, special price and delivery schedules associated with it. A meal is composed of one or more ingredients that can be managed from the meal management interface. As well as there is the management of delivery schedules and the actual deliveries that are going to customer's door steps every day. From the transaction management interface, the administrator can manage the debit and credit payments of the customers. He/She can view weekly, monthly and yearly reports of previous deliveries and the revenue. Customers also can leave review about individual meal which he/she can manage from the Review management interface.

1.1 Project Context

This project sets to design, build and test a web-based catering management system. Generally, a web application for catering system aims to solve carting problems with Information Technology (IT). Catering is the business of providing food service at a remote site or a site such as a hotel, public house or other locations. Catering has evolved to become an artisanal affair. Caterers now create an experience that involves the senses. Over decades, the catering management system had evolved to cover more operational aspects in a catering service.

1.2 Project Motivation:

IT has become important tools to support business operations. Especially in the catering business, IT is playing increasingly important roles in resources administration, managing services, and assisting strategic decision making. Several analysis and research works have also suggested that competitive use of IT in a catering business has significant advantages. In term of operational benefits, it can improve process efficiency, reduce possible human errors and maximize use of resources Additionally, it also supports long term business goals, including achieving cost-effectiveness, maximizing profits, and the potential to penetrate wider markets

1.3 Project Objectives

The general goal of this project is to develop, assess and deploy a web-based catering management system. This is supported by researching sufficient knowledge in the SE and HCI domains and then applying this to the development of the system. To achieve this, the project will be underpinned by various stakeholder requirements. It also investigates the possibility of building a real-time information system to allow effective communication.

The following objectives are defined in order to accomplish the project goal:

- a) Developing a multi-tier application
- b) The system required an architecture design that supported separation of concern and highly reusable implementation. It should allow full or part of the functionalities to be accessible by a range of devices. Hence, it should have presentation layer to facilitate an appropriate user interface, a business logic layer that hold common business functions and a data layer for managing database transaction.
- c) Designing a simple and intuitive user interface
- d) In any system, users will need to perform several tasks to achieve a high-level goal. A user interface should guide user through the tasks and help them to attain final goal. Since operations in a catering business involve numerous tasks, extensive analysis should be performed into designing a UI that is simple and intuitive and addresses the functional requirements effectively.
- e) Developing efficient information sharing methodologies
- f) Sharing information should happen instantly since a delay in fulfilling delivery certainly reduces customer satisfaction. The project will need extensive investigation on how real time communication should occur to allow effective collaboration. Besides, data transfer for cross layer communication could also affect system performance and should be addressed by an optimized solution.
- g) Developing a mobile friendly user interface
- h) The system should be easily accessed by different type of devices; so that it is portable and reusable. Considering that each mobile device may have a different screen resolution and size, the UI of the system should provide a responsive mechanism to offset these limitations with a dynamic UI layout and content resizing.
- i) Ensuring quality of the application with through testing

A significant amount of testing such as A/B testing, unit testing and integration testing should be performed in order to ensure that the system is free from bugs and errors. In

addition, the performance should be evaluated to analyze the effectiveness of the proposed methodology.

1.4 Expected Outcome

With this online system, management and delivery will become easier and systematic to replace traditional system where are still using paper. The main point of developing this system is to help catering service providers manage their business and help customers manage their deliveries in real time with the goal of keeping everything organized, more efficient, eliminate errors seen in manual paper works, reduce cost and to help the business grow faster. So, the deliverable of this project is a web based applications for online catering management system.

1.5 Report Layout

This report has been divided into various chapters which documented the concepts and solutions behind the development of the prototype of Online Catering Management System. It is broken down into six major chapters including this one (Chapter 1). This chapter has introduced the project in terms of its context, motivations and objectives. The remainder of the report is structured as following:

- Chapter 2 discusses about discussed background of this project and related works that were conducted by others. Then we analyzed the problem that were faced and scope of the problem that were not considered in their solution. Then it describes approaches that could improve this type of system.
- Chapter 3 performs a requirement elicitation and analysis based on the business workflow of LunchBD by conducting live and phone meeting with the stuffs of this company. Then this chapter put together a set of requirements that the project has to satisfy to fulfill its objective.
- Chapter 4 represents the process of transforming the requirements into conceptual solutions. It familiarize and provide a overview of the approach on both front-end and back-end implementation of the system, interaction design & UX.

- Chapter 5 focuses on the tools and technologies to realize the conceptual design ideas that were depicted in the previous chapter. Finally, it walkthroughs of the actual practice of the developed system is shown. Also, it ensure the systems stability, integrity and reliability by conducting various types of testings.
- Chapter 7 assesses the approaches taken in this project and the accomplishments of the project. It also points out possible future enhancements and a reflection of the skills achieved during this project.

CHAPTER 2: BACKGROUND

This chapter describes the background of catering management system and utilization of ICT in this business. Also it looks at the project's nature in a wider context. It begins by understanding the characteristics of online based catering management system and looking at early attempts at such a system. The chapter then discusses how web technology fits into the development of such system.

2.1 Introduction

The project is developing because; many catering service have a lot difficulties to manage their business manually. If a customer wants to cancel a delivery he must inform the responsive personnel with a phone call, sms or email which is not so convenient both for the customers and the responsive personnel. The customer service provider must take a note with pen and paper and fulfill this request during the packing which is prone to human error. Also, this information may not be stored for future reference, as a result the business owners fails to keep track of their business and may find difficulties to monitor. Sometimes, customers have their own preference about a meal on a particular day that they don't like or looking for an alternative menu. This becomes very difficult for the responsive personnel to keep track with pen and papers.

In the early 1990's, this limitation was removed when web servers were modified to allow communication with server-side custom scripts. No longer were applications just static brochure-ware, edited only by those who knew the arcane mysteries of HTML; with this single change, normal users could interact with the application for the first time. A web based catering management system is the one that facilitates the process and management of customer, delivery, schedule, accounting, report, customer relationship through a web application. Basically, a web application is a computer program that utilizes web browsers and web technology to perform tasks over the Internet. It's often referred as Web App is an application program that is stored on a remote server and delivered over the Internet through a browser interface. It offers the user a broad array of information, arranged in a way that is most convenient for the user to access. When

designed, implemented and maintained correctly a web app becomes the starting or entry point of a web user introducing him into various information, resources. In the early days of the web, web sites consisted of static pages, which severely limited interaction with the user.

2.2 Related Works

Over the years many online based catering management systems and similar online services have emerged. Some of them have come up with very unique idea while others have just implemented ICT in their traditional business workflow.

- The Foodpanda group: One of the biggest and global mobile food delivery marketplace is known as Foodpanda group. Its headquarter is located in Berlin, Germany and operating in 12 countries and territories. Their service allows users to select from local restaurants and place orders via its mobile applications as well as its websites. They are actively providing their services in many different areas of Bangladesh. Foodpanda doesn't prepare the food themselves. However, they work as a intermediary between their partner restaurants and the customers. They take orders on their behalf and deliver them right into their doorstep. The service is available via its websites and mobile applications. Customers order food by entering their postcodes on the site and browsing for food from a list of restaurants. They can create meals by browsing restaurant menus and selecting items they want to order before entering an address and proceeding to the checkout. Restaurants receive these orders and then deliver to customers. Foodpanda sends out an SMS to confirm orders and their estimated delivery time. Foodpanda also has a review section on restaurant pages, where customers can comment on sales process, delivery, taste and overall impression of restaurants. Foodpanda has entered into a collaboration with Indian online railway system enabling customers to buy meals while travelling, beginning December 2015. Foodpanda is providing their service in Bangladesh since 2014.
- Mealsbangladesh.com: A corporate catering service provider located in Dhaka who provide from boxed lunch to complete event food cater coordination. Their

primary focus is corporate catering service however, they serve social events as well. It was founded in 2016.

- Foodpeon.com: It's a homemade online food delivery service company in Bangladesh. Their platform allows customers to order food and get them delivered to its destination. Cooks around the country subscribe on their website and publish various types of menu that they prepare at home. Foodpeon takes the order on their behalf and deliver them to customer. They are currently operating in Nikunja, Banani, Mohakhali, Gulshan, Bashundhara Baridhara, Badda, Banasree & Uttara. According to them, this company take custom order even for a single person. In most cases, the chefs only cook after you placing an order.
- HungryNaki: Hungrynaki.com is a fully Bangladeshi Online Food Ordering and Delivery Service launched in 2013 to deliver your food.

2.4 Scope of the Project

- The scope of the project includes the followings:
- Any catering business can use this system including LunchBD
- All of the catering delivery services and related works can be managed by this system
- Application Support, Maintenance and Deployment of the system

2.5 Challenges

As this project progress, it seeks to answer several interesting research questions:

1. Could the web based management system fully replace the paper-based system in catering business?
2. How would the stack holder be benefited from this system?
3. How does the project address the requirements of each stakeholder?
4. What interface design technique could ease the tasks that the stakeholders require the system to perform?
5. How could the project ensure that the development process is being properly managed?

6. What software design methodologies and best practices can be utilized to support the desired functionalities and performance?
7. How would the interface support and utilize mobile device as medium to access its services?

These questions are worth investigating because their answer would help us determine and realize the scope of this project, important of SE & HCI principles to support the development of this system.

CHAPTER 3. REQUIREMENTS

This chapter investigates the necessary requirements of the project based on an understanding and analysis of the needs of system users. These requirements are then translated into informative models to assist decision-making and problem understanding in the requirement modeling phase.

3.1 Requirement Elicitation

The requirements of this project was elicited with several phases. Firstly, a initial discussion was arranged with the members and staffs of LunchBD catering management system. We collected as much as information needed to understand to current workflow of their business and the problem they are facing while managing it. The collected data represents that, currently their business is running by pen and paper. The marketing staff is responsible for communicating with the customer directly. Sometimes the communication happens over phone and sometimes face to face. He collect information about the customer and his preference about food and send over the information to their management. The stuff responsible for book keeping stores the information for future reference during delivery. If the customer has to contact Lunchbd for cancelling a meal, register as out of reach or ask for an additional meal for a particular day he/she has to call them back again. Often, the process does not go smoothly as Lunchbd needs to take such requests before the time of their meal preparation. This cause both trouble and resultant customer dissatisfaction. Also, sometimes customers has their own preference for their meal due to health, personal or other issue. Lunchbd must follow this exception during every delivery otherwise it can cause problem with customer as well as financial loss of their company.

3.2 Analysis of Stakeholders

At this moment LunchBD is only providing service in Daffodil University campus. So, it's recognized that, the major stakeholder of their company are the teachers of this institute, staffs and students. Also, the staffs of the company are the stackholders of the proposed system as they are going to manage it.

So, there are two types of stakeholders associated with the system, Clients and the Business Manager that can be expressed as customers and administrator of the system.

3.2.1 Customers

To subscribe to the proposed catering service and to access the functionalities of the web application a customer must register on the site. During the registration process he/she enter their credentials that can be used to login to the system later on. Individual customer has their own dashboard to manage their account where he/she can update their account information such as security credentials, delivery address, request to personalize their menu, cancel delivery, view payment reports, delivery report and expenses etc.

3.2.2 Administrator

Another type of stakeholder of this system is the administrator. Basically, administrator is responsible for the management of the business account such as menu and delivery schedules management, processing delivery, updating customer information, bookkeeping, responding to customer requests and feedbacks etc., generating annual report, management of reviews etc.

3.3 Requirement Analysis

As per the collected data in phase 2.1 "Requirement Elicitation" it's obvious that, there are several issues LunchBD come upon while running their business. The major issues are the communication, reliability of information and their application. So, based on this fact it's concluded that they need a system to manage the internal business process that can be accessed from anywhere and easy to operate by anyone whether they are at home, office or on their way to make the workflow convenient for both the customers and company staffs. So, based on this scenario a web based solution was proposed. With this

system the customer would be able to manage most of the things they are now doing over phone conversation or direct meeting. Similarly, it would save the marketing staffs some times and let him/her focus on other business processes to better serve the customers. Eventually, this will improve quality and service of their business. Mostly with this system the workflow that's done by pen and paper would be replaced by a computerized system. It won't take any major changes in their business workflow. However, all of the data they have not has to be initially entered into the system. Also, the staffs who are responsible for bookkeeping requires training on how to operate the proposed system. And, more importantly they need to inform the customers of this new system and encourage them to exercise that and help them understand the benefit of the proposed system which requires some marketing strategy.

3.4 Requirement Specification

At this stage the requirement specifications are clarified so that we can define requirements in clear and unambiguous language based on requirement identified during requirement elicitation and requirement analysis. The requirements are evolved over time and more accurately reflect the needs of the stakeholders. Based on this fact, followings are the two major categories of requirements that we identified:

3.4.1 Functional Requirement (FR)

- a) Customer management: Subscribe customers on the Online Catering Management system and store required information about them to identify individual customer, provide the ability to change information at any time and allow them to manage their own accounts and information.
- b) Meal management: Store information about the meals and their delivery schedule
- c) Delivery management: Store and manage the information of everyday deliveries and the amount spent for every meal by individual customers
- d) Payment and transactions: Manage payments made by the customers
- e) Ability to cancel a meal or request an additional meal on a particular day
- f) Allow customers to suspend the delivery for a specific number of days such as during their holidays period

- g) Possibility to store custom preference about meals for specific customers
- h) Review Management: Allow customers to review and provide feedback about the quality and value of meals as well as the service.

3.4.2 Non-Functional Requirement (NFR)

- a) Security and privacy: Proper security and privacy practice should be considered and maintained by utilizing the technique of authentication and authorization.
- b) Performance: performance of the system should be optimized
- c) Operational environment: The system should be operable on most devices that supports web browsing such as personal computers, laptops, notebooks, smart phones, tabs, PDA etc. It should be cross browser and cross platform compatible.
- d) Personnel: The development of this project is to be conducted by a single individual (Nasir Ahmad) who is assumed to have the necessary knowledge and skills to complete this project. There's no specific requirements for the end users given its simple functionality which is intended to be usable by most of the users with basic computer literacy skills.

3.6 Use Case Modeling and Description

In order to better understand the proposed system to be built and to identify important ideas and decisions the following use case model was represented:

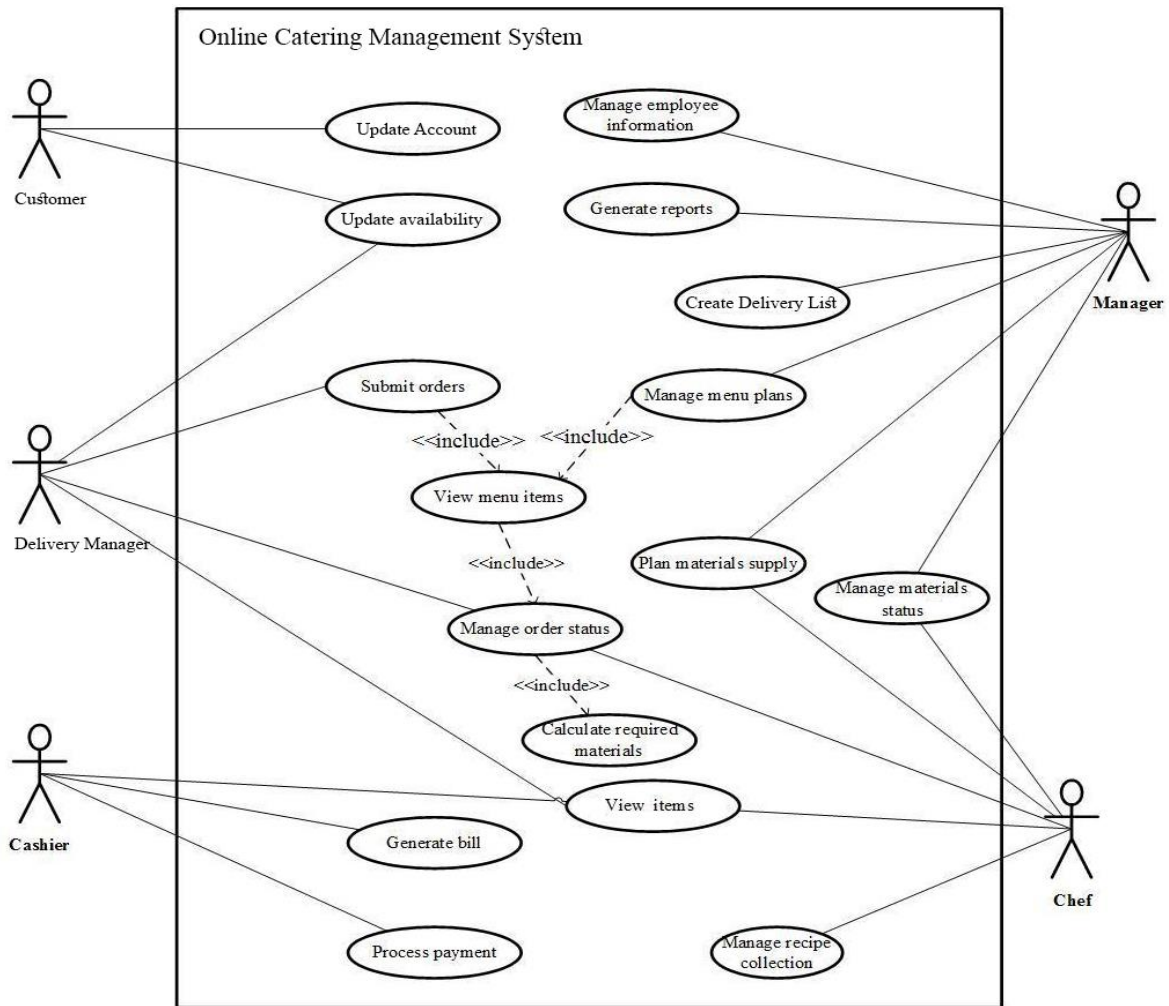


Figure 3.6.1: Use case model of catering management system

This model shows the Online Catering Management System and the interaction of various type of stakeholders with this system that can be identified as actors such as Customer, Manager, Delivery Manager, Cashier and Chef. This scenario shows that, the Customer can access the system and update their account information as well as update their availability to receive delivered meal. Delivery Manager submits the orders/delivery including viewing menu items, updating order status etc. Cashier usually performs the bookkeeping stuffs such as taking/processing payments, generating bill. Manager on the other hand can generate reports, manage menu plans and menu supply. Chef can view menu every day, plan materials/ingredients supply and manage the status about them.

3.7 Logical Data Model

The following E-R diagram illustrates the logical structure of databases and relationship between the entities exist in Online Catering Management system. Customer is the center of all other entities showed in this diagram. A Customer has a User account that stores the credentials for accessing the system based on his/her role. Also, a customer can have one more debit/ credit payments that is related and represented with the Transaction entity. One or more meal Deliveries can be arranged followed by a Reserved Delivery for a customer in a particular Schedule which also initiates a debit Transaction. A delivery can have one or more Delivery Items based on Meal entity. Meals may be composed of one or more Ingredients. If a particular meal is not suitable for a Customer, admin can propose a Replacement Meal. Also, Customer can add a Meal as their Favourite Meal. Customers can also submit Request and Review the meals they have been delivered.

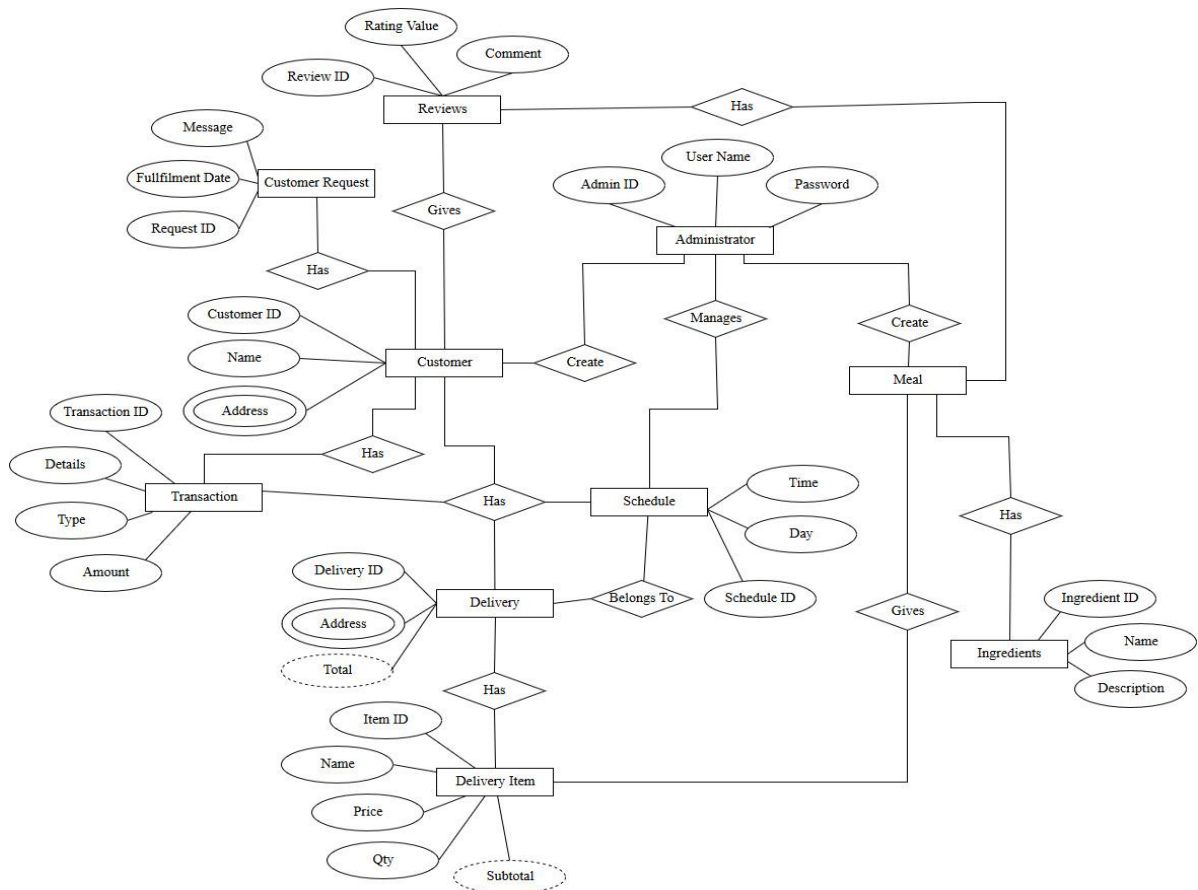


Figure 3.7.1: Entity Relationship model of the Catering Management System

3.8 Design Requirements

The proposed Online Catering Management system is a web application that's to be developed based on several web technologies. Based on the analysis of the requirements the following tools and technologies were selected.

Design Tools:

- MySQL Workbench for design the database
- StarUML to design the logical diagrams

Development tools:

- PHP Programming language as server side language
- HTML, CSS, Javascript for frontend design and development
- MySQL for the database of the system
- CakePHP is a PHP framework that was used as a foundation of the whole system which would provide us several benefits
- XDebug for debugging
- Netbeans as integrated development environment
- Mozilla Firefox as web browser with Firebug and Web Developer plugins

Testing tools:

- Apache JMeter to load test functional behaviour and measure performance
- PHP Unity for unit testing

CHAPTER 4: DESIGN SPECIFICATION

This chapter will describe the each stage of designing, identifying the associated problems. It discusses how web technology can be utilized to develop such system. Next, it investigates the software development and design methodology, processes and best practices to implement them. It then focuses on UI design technique that could enhance user experience and accessibility of mobile devices.

4.1 Frontend Design

The Online Catering Management System is a web applications. The frontend of the system was designed and coded in browser-supported language such as JavaScript[18] and HTML [17] as these languages rely on the browser to render the program executable [1]. Some of the applications are dynamic, requiring server-side processing. Others are completely static with no processing required at the server such as images, CSS style sheet, JavaScript files.

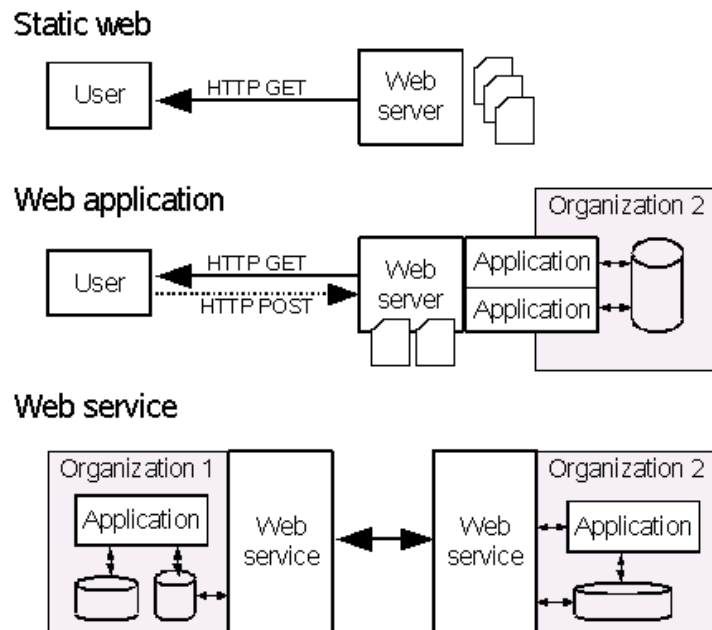


Figure 4.1.1: Web applications

This applications use a combination of server-side script PHP [5] to handle the storage and retrieval of the information, and client-side scripts (JavaScript and HTML) to present information to users. This allows users to interact with the system using online forms, content management systems and more.

In this section we will dive into design phase of the system as well as discuss about the functionalities of the system that is going to be developed and how to use them.

4.1.1 Home Page

It's the main entry point of the website. When a user first visits the URL of the web application he/she is introduced to the system with this page. So, it must be elegant, appealing, user friendly and comprehensible UI so that the visitor appreciate this and feel further interests to proceed. As we already discussed this is going to be a responsive website so, the visitor would experience different user interface depending on the device he/she is browsing from. However, the actual functionalities of the pages would be consistent so that they can easily find it no matter what devices they use.

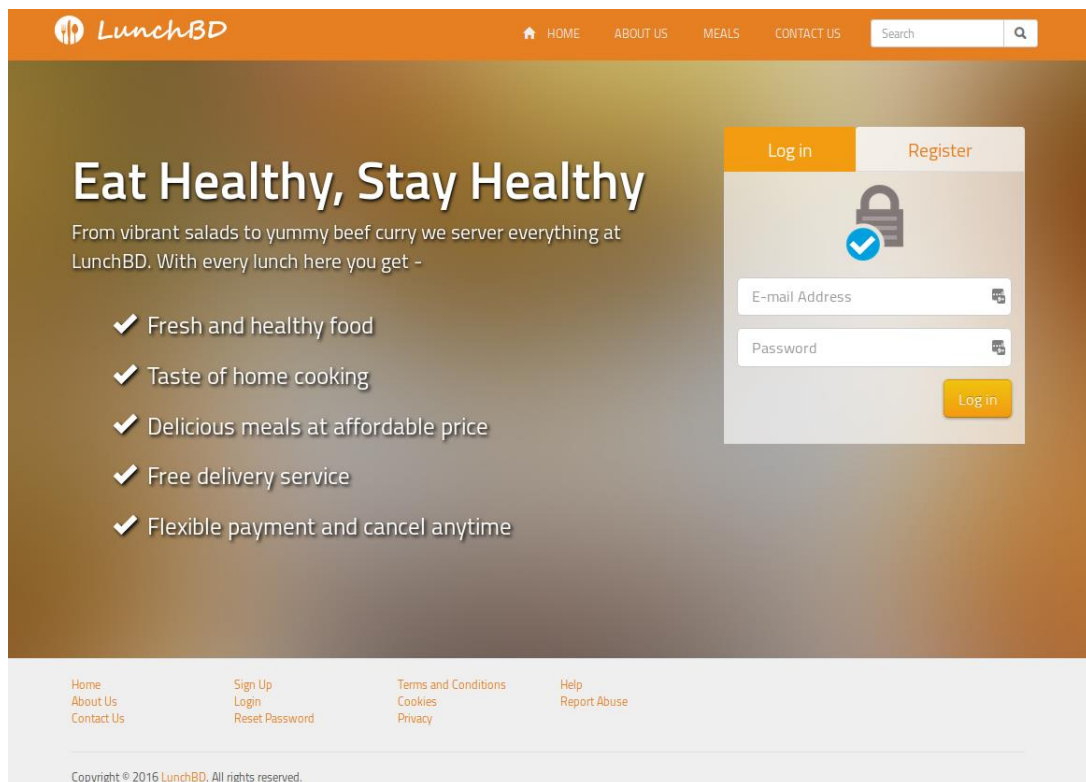
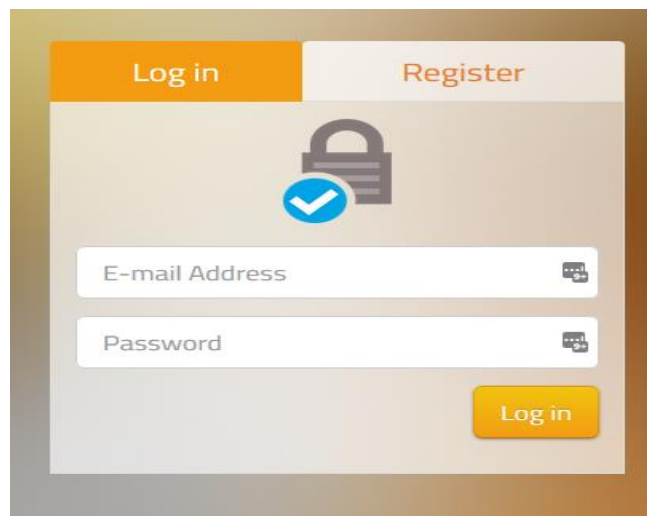


Figure 4.1.1.1: Homepage design of the website

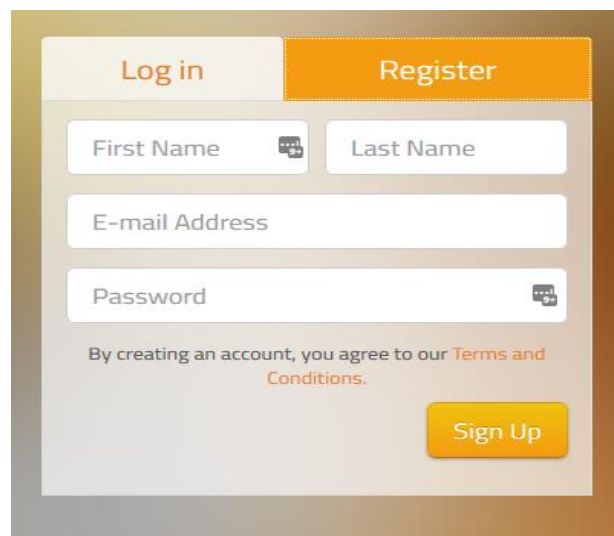
4.1.2 Login and Registration Forms

The homepage is consists of the Registration and Login form that allows the users to register on the system and a registered user to login. Also, search bar at the top can help them find information easily such as information about the services, Meal, etc. Additionally there's About Us, Meals and Contact Us links at the top to easily navigate to the information they are interested about. Once the users logs in he/she is redirected to his account dashboard where various functionalities of the website can be accessed and the users can manage their account. Depending on the user role that's either a customer or the administrator different user interface is represented.



The login form features a header with two tabs: 'Log in' (active) and 'Register'. Below the tabs is a central icon of a padlock with a blue checkmark. The form contains two input fields: 'E-mail Address' and 'Password', each with a small eye icon for visibility toggling. A yellow 'Log in' button is positioned at the bottom right.

Figure 4.1.2.1: User login form design



The registration form features a header with two tabs: 'Log in' and 'Register' (active). Below the tabs are four input fields: 'First Name' and 'Last Name' (side-by-side), 'E-mail Address', and 'Password' (with an eye icon). Below the fields is a line of text: 'By creating an account, you agree to our [Terms and Conditions](#).' A yellow 'Sign Up' button is located at the bottom right.

Figure 4.1.2.2: User Registration form design

4.1.3 Static Pages

The frontend also contains several static pages that would represent the information about the company and use of the various functionalities of the web site.

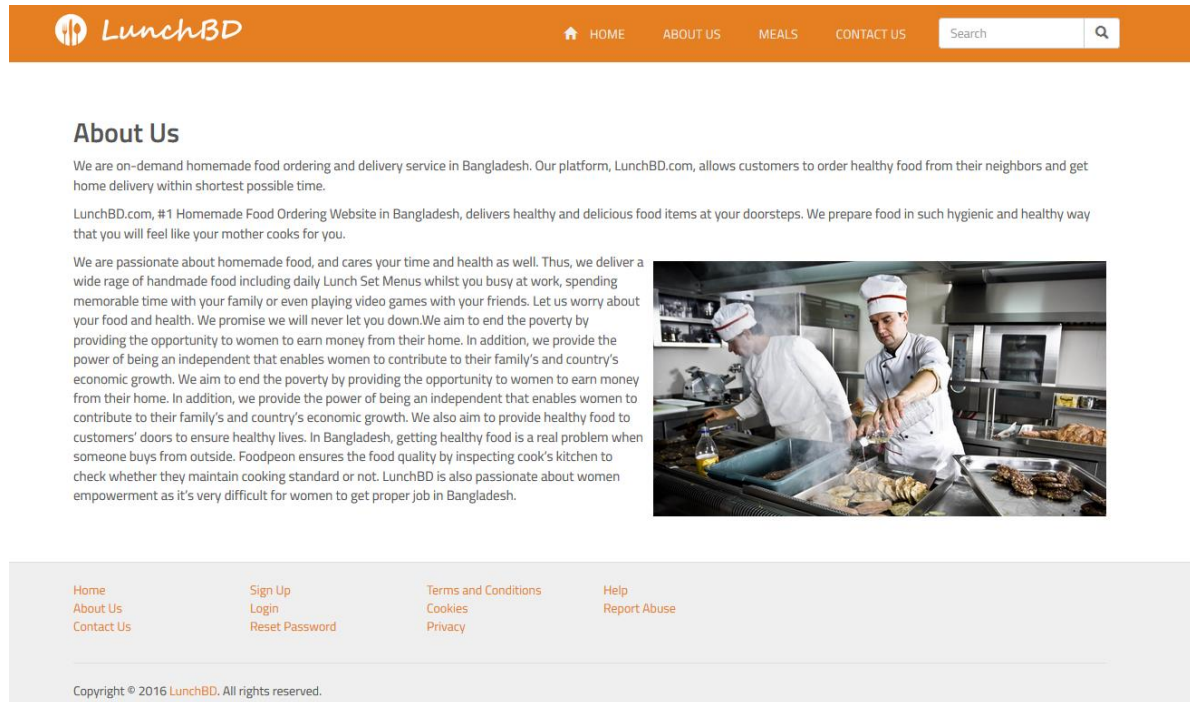


Figure 4.1.3.1: Design of the informative pages

4.2 Backend Design

Once a customer is logged in, he/she is redirected to the backend of the system. The backend page is composed of various modules such as Customer, Delivery, Meals, Schedule and Review modules. These modules represents pages and forms where a user can interact with the system. There are two types of user available on the system such as User and Administrator. Based on the user role different backend is introduced to the user.

4.2.1 My Account Page

It's a common page for both user and the administrator and most of the functionalities and UI elements are similar. On "My Account" page a user can enter information about them in the system such as photo, name, address, mobile, telephone, email and also can change their password. This UI is same for both customer and the administrator. The address client enters here is used during the delivery and that's where they receive their meal. However, for administrator this area is just to keep record of their information for the business owner to keep track of.

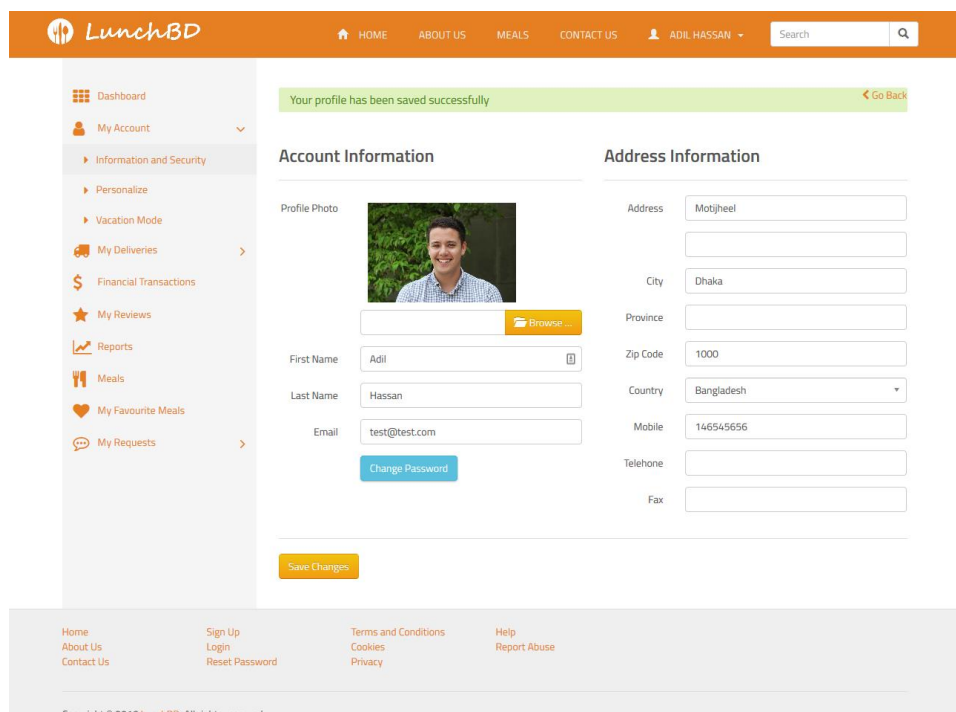


Figure 4.2.1.1: My account page

4.2.2 Personalization Page

This area is only accessible by the customers where they can input the information regarding their personal preference. This is particularly important and must be filled for not only to work the system properly but also for his/her own benefit. So that, the system can decide whether a particular meal is good for them or not. Customers can update this information anytime they want from this page.

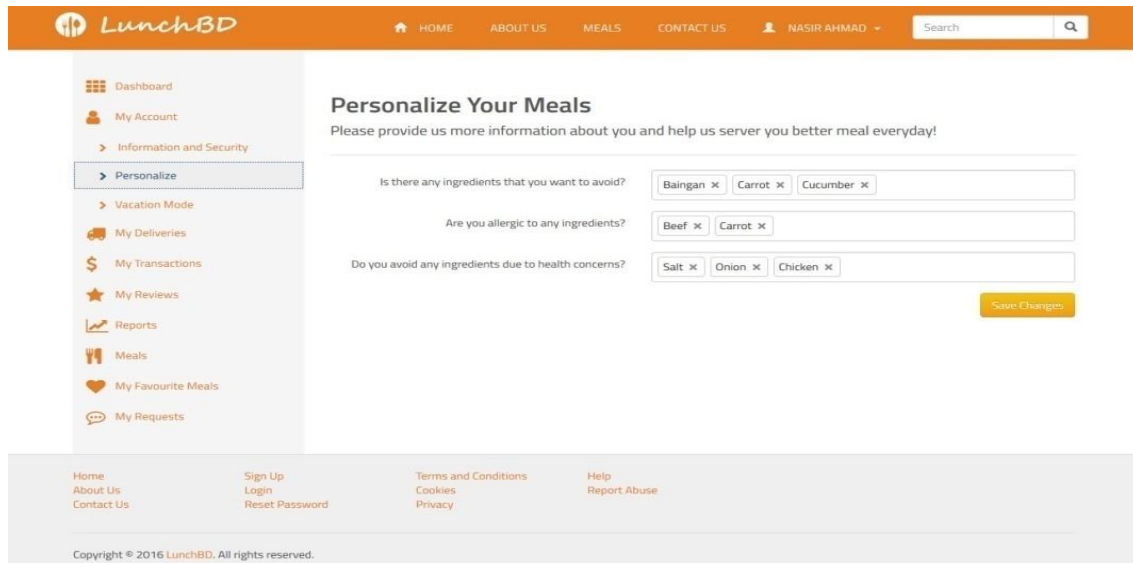


Figure 4.2.2.1: Meal personalization page

4.2.3 Vacation mode

If the customer is on holidays and not available to collect food delivery, he/she can turn vacation mode and specify the range of date they are not available. The system will automatically take this into account and help the administrator not to process this delivery and save both of their trouble. This is automatically turned off once the date is expired and customer would continue to receive delivery after that. This page is only available to the customers which can be accessed upon login.

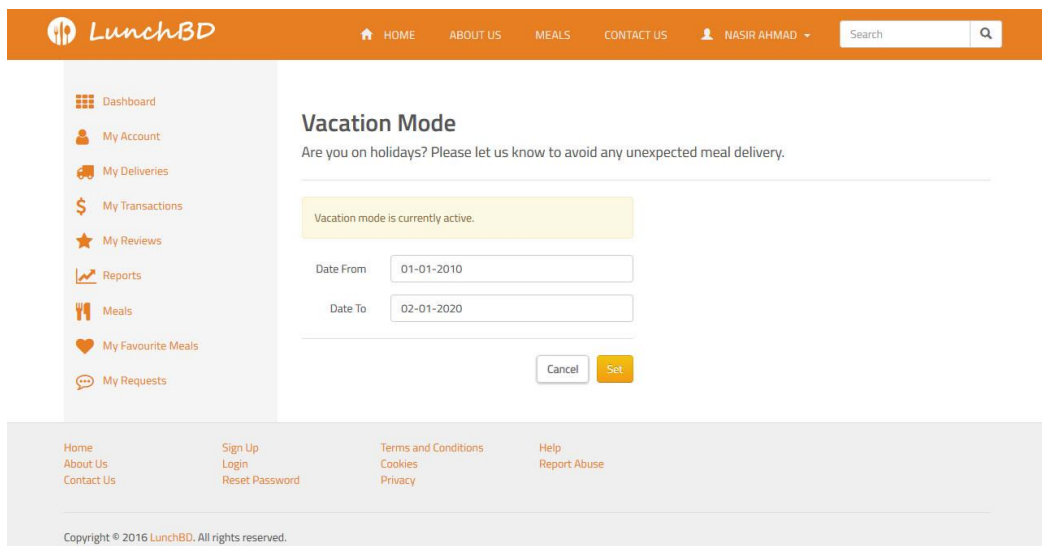


Figure 4.2.3.1: Setting up vacation mode

4.2.4 Customer Management Page

The administrator can view the list of existing customers from the customer management page as well as update the information regarding the customer. If required he/she can also add new customer by clicking on the “Add” button. Then the administrator is redirected to the page described in 4.2.1 where he/she can fill all required information and save it in the system. On this page the administrator can search a customer based on their ID, name and email address. Also, the administrator can edit customer information and remove them.

Customer ID	Name	Email	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
2	Nasir Ahmad	nasir423@idui.edu.bd	Q ✎ ✖
3	Demo Admin	admin@test.com	Q ✎ ✖
4	Abdul Kadir	test2@test.com	Q ✎ ✖
5	Adil Hassan	adil@test.com	Q ✎ ✖
6	Fizul Haque	test3@test.com	Q ✎ ✖
7	Riaz Uddin	test4@test.com	Q ✎ ✖
8	Jashim Uddin	test5@test.com	Q ✎ ✖
9	Kamrul Islam	test6@test.com	Q ✎ ✖

Figure 4.2.4.1: Customer management page

4.2.5 Meals Management Page

On meal management page administrator can view the list of existing meals on the website. Also, he/she can update their information and remove a meal by clicking on the “Edit” and “Remove” button respectively. On this page the basic information of the meals are displayed. such as Name, SKU, Price, Special Price, Status etc. To view more detailed information he has to click on name of that meal. Also, there’s a “Add new Meal” button that can used to add a new meal on the website.

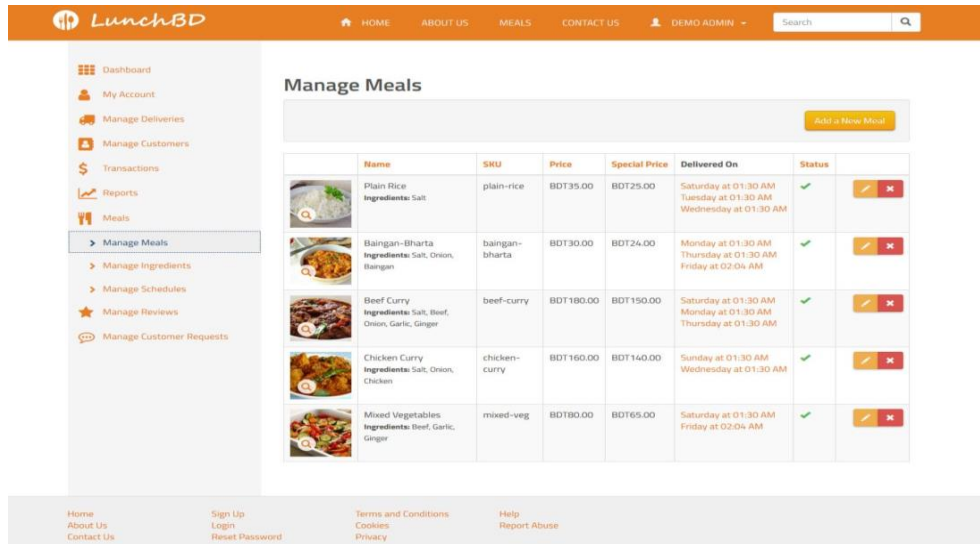


Figure 4.2.5.1: Meal Management Page

Upon clicking on “Edit” or “Add” button, the administrator is redirected to the following page where he/she can update the information regarding meal such as Name, SKU, Price, Special Price, Description, select the list of ingredients the meal is composed of and the delivery schedule of this menu. The ingredients and schedules can be managed from their own management pages. Only the available ingredients and schedules can be entered here. So, the administrator must setup these prior creating the meal. Also, these information can be updated later on whenever required. It’s also important to upload the photo of the meal that can be done from this page.

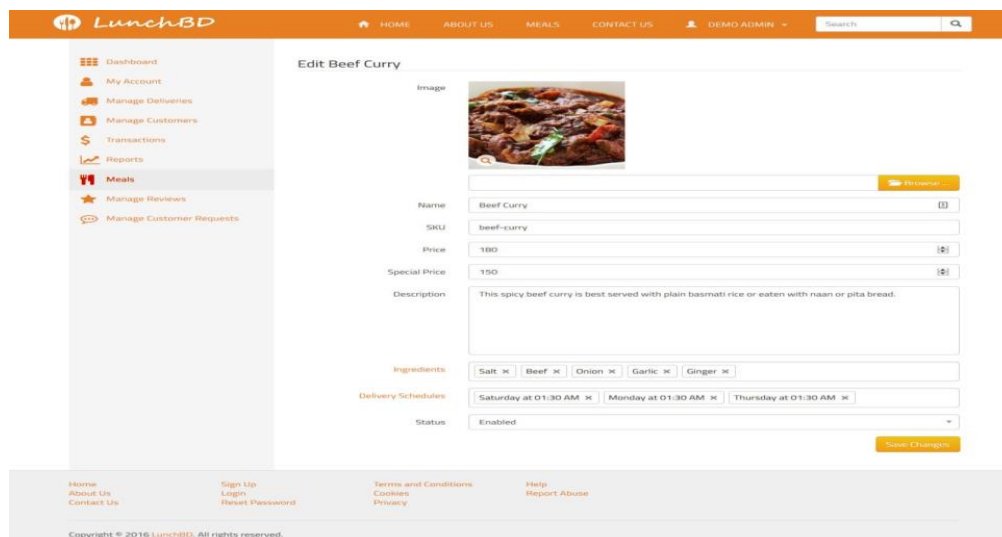


Figure 4.2.5.2: Meal add/edit page

4.2.6 Ingredients Management Page

Administrator can view the list of existing ingredients on the website as well as update and add new ingredients. Ingredients that are created on the website can be associated with multiple meals. This information is important for the system to work properly. Because, this particular information is used to determine whether a customer is sensitive to any ingredients. And necessary notification is displayed during the delivery processing time.

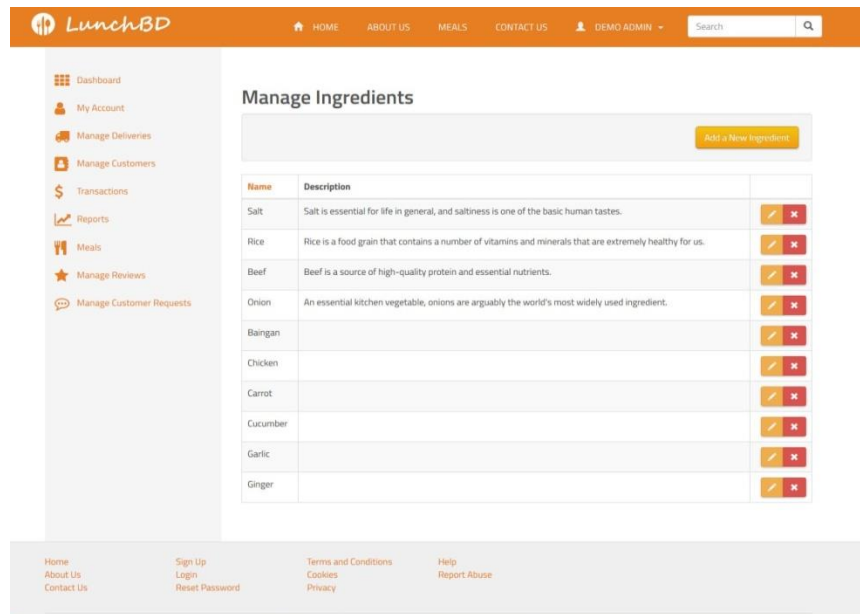


Figure 4.2.6.1: Manage list of ingredients

On ingredients list page the basic information of the ingredients are displayed as well as the required buttons to take action on them such as editing or deleting an ingredient. Also, there's a button to add new ingredient in the system.

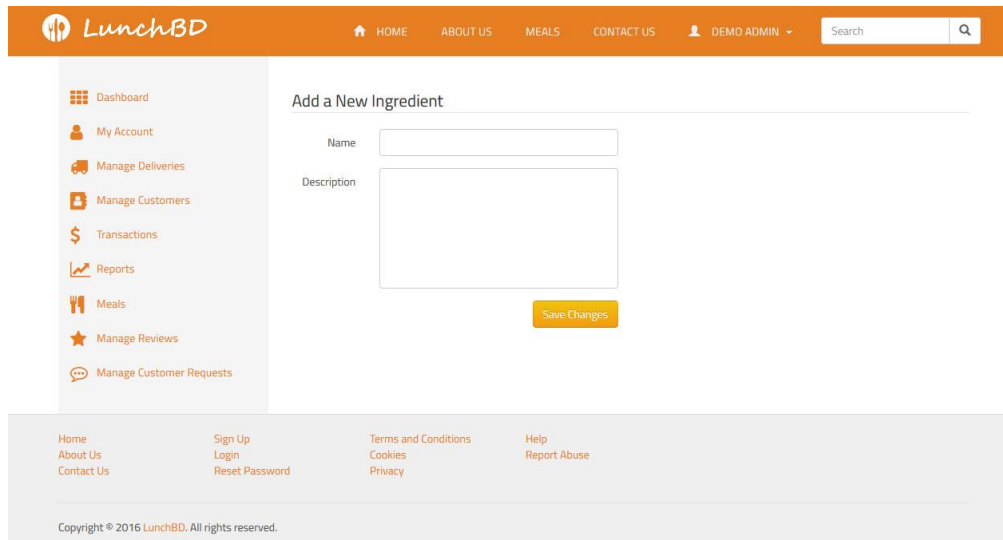


Figure 4.2.6.2: Add a new ingredient page

Clicking on the “Add” button the administrator is redirect to a new page where he/she can enter all the required information to create an ingredient. Then the “Save” button must be clicked to save the changes in the system.

4.2.7 Schedule Management Page

On schedule management page the administrator can view the list of existing schedules on the website when the delivery is sent.

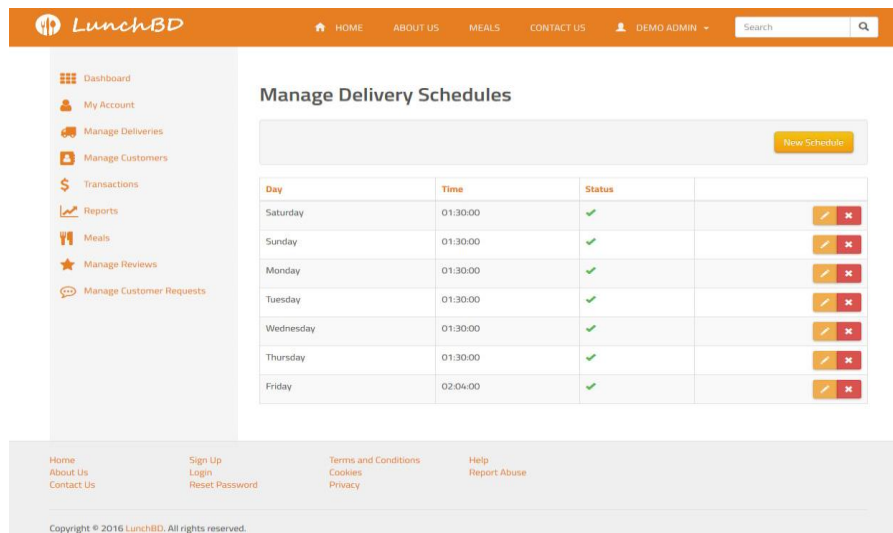


Figure 4.2.7.1: List of schedules on schedule management page

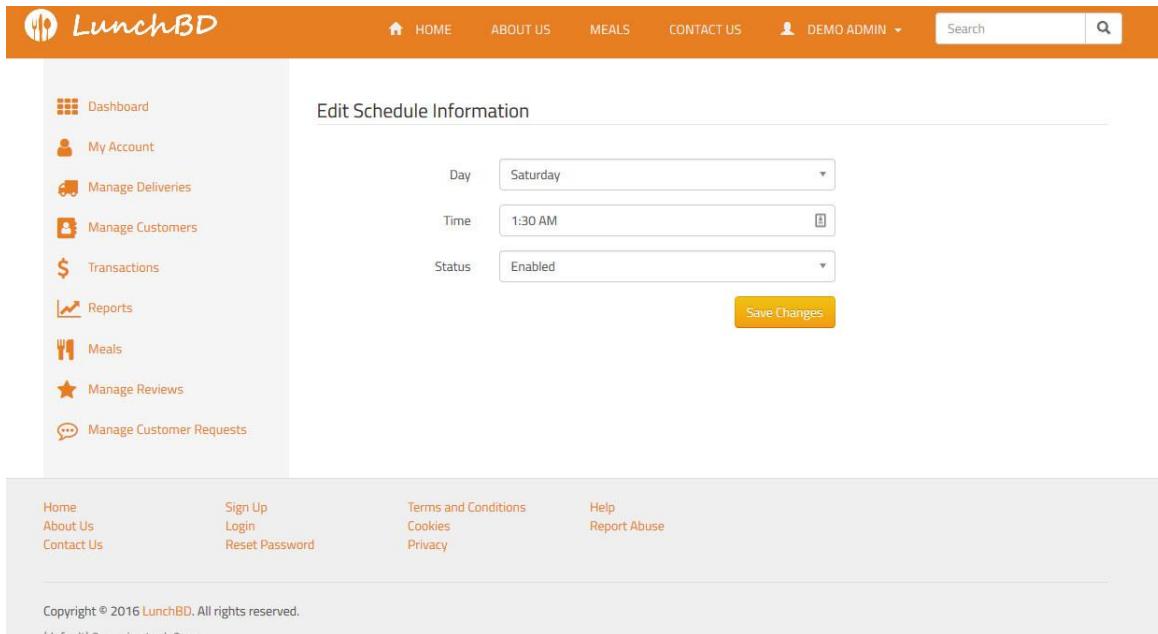


Figure 4.14: Add/update schedule information

Also, he/she can update the information of the existing schedules and add new schedules. These schedules are then associated with individual meal to specify the time it is delivered. This information is used during the delivery processing.

4.2.8 Delivery Management Page

It's one of the most important part of the system that is used by the administrator regularly. Basically, this page is where all of the data entered in the previous sections come to action. The delivery process is started by click on the “Generate” button that creates the list of deliveries that are to be processed on a particular day. The delivery list page shows the number of deliverables as well as the basic information about them such as the name of customer and his/her the menu will be delivered to, delivery related information, status of the delivery etc. If there's any issue with a delivery such as a ingredients is found to be unhealthy for the customer is displayed in delivery info section. The administrator then has to contact the customer regarding this and find an alternative meal for him/her. The alternative meal for that particular customer then can be entered into the system by clicking on “Resolve” button. If any of the customer is unavailable that information is also displayed on this page as well as the canceled delivery list are

also displayed. Additionally, individual delivery can be reviewed before processing by clicking on the “Process” button or the administrator can process all orders at once by clicking on “Process All” button located on this page.

Delivery Management

Total 7 record(s) have been generated successfully.

Date: 22-08-2017

Generate Delivery List

Customer Name	Address	Delivery Info	Delivery ID	Status	Actions
<input type="checkbox"/> Riaz Uddin	No address was found Email: test4@test.com Mobile: Telephone:	Ready for packing		pending	Q +
<input type="checkbox"/> Nasir Ahmad	12 Nayapaltan, Dhaka, 1000, BD Email: nasir423@diu.edu.bd Mobile: +8801915952898 Telephone:	Requires attention about ingredients: The customer has health concern about the ingredient(s) "Salt" found in this meal. Resolve		on_vacation	Q +
<input type="checkbox"/> Jashim Uddin	No address was found Email: test5@test.com Mobile: Telephone:	Ready for packing		pending	Q +
<input type="checkbox"/> Abdul Kadir	No address was found Email: test2@test.com Mobile: Telephone:	Ready for packing		pending	Q +
<input type="checkbox"/> Kamrul Islam	No address was found Email: test6@test.com Mobile: Telephone:	Ready for packing		pending	Q +
<input type="checkbox"/> Adil Hassan	T&T Colony, Motizheel, Dhaka, 1000, BD Email: adil@test.com Mobile: 017145656546 Telephone: 146545656	Requires attention about ingredients: The customer has health concern about the ingredient(s) "Salt" found in this meal. Resolve		pending	Q +
<input type="checkbox"/> Fizul Haque	No address was found Email: test3@test.com Mobile: Telephone:	Ready for packing		pending	Q +

Previous Next

Home Sign Up Terms and Conditions Help
About Us Login Cookies Report Abuse
Contact Us Reset Password Privacy

Copyright © 2016 LunchBD. All rights reserved.

Figure 4.2.8.1: Delivery management page

4.3 Interaction Design and UX

This section will familiarize with possible user interactions with the system and user experience with sequence diagram.

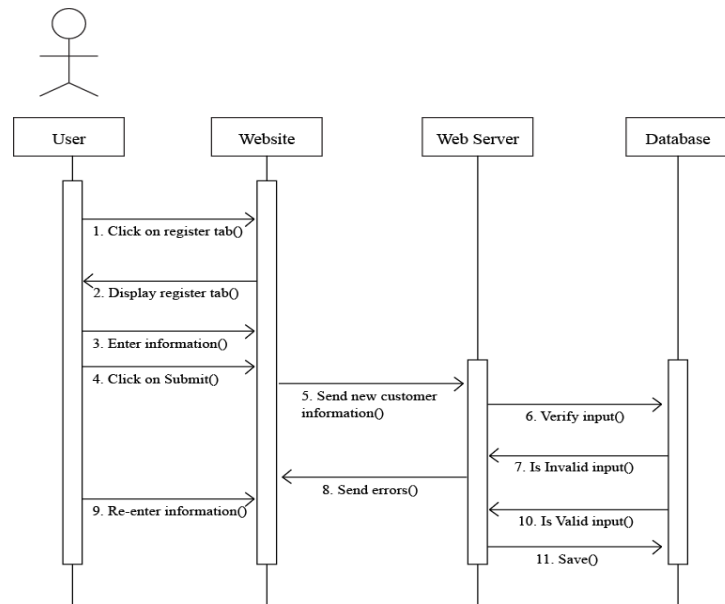


Figure 4.3.1: Sequence diagram of registration process

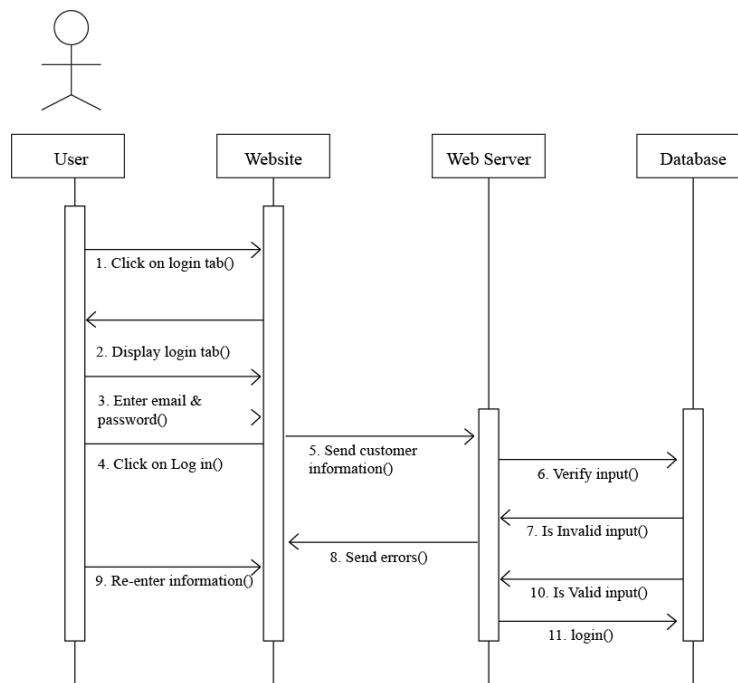


Figure 4.3.2: Sequence diagram of login process

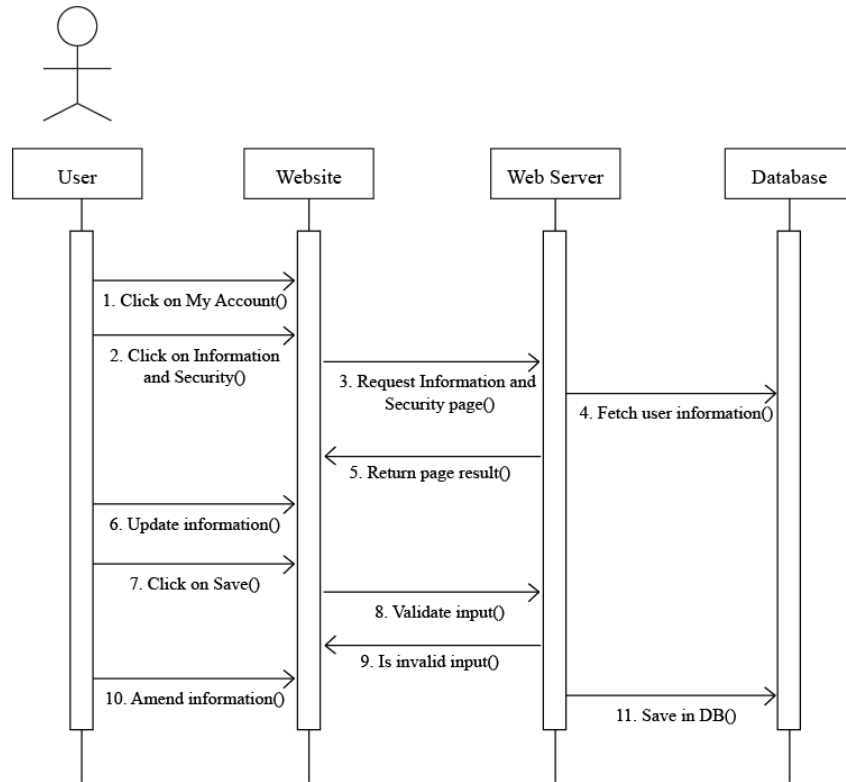


Figure 4.3.3: Sequence diagram of account information updating process.

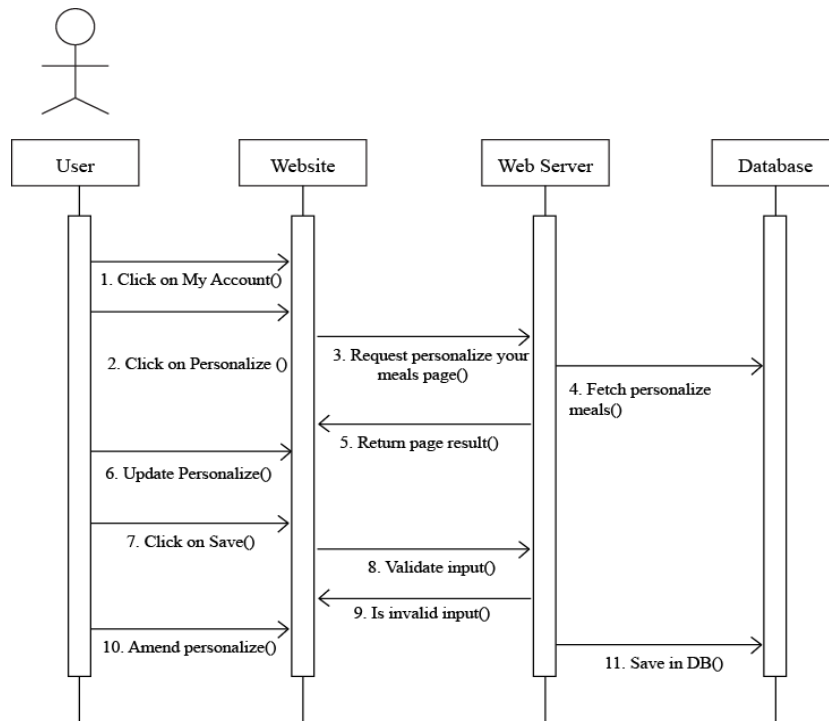


Figure 4.3.4: Sequence diagram of setting up personalized meal.

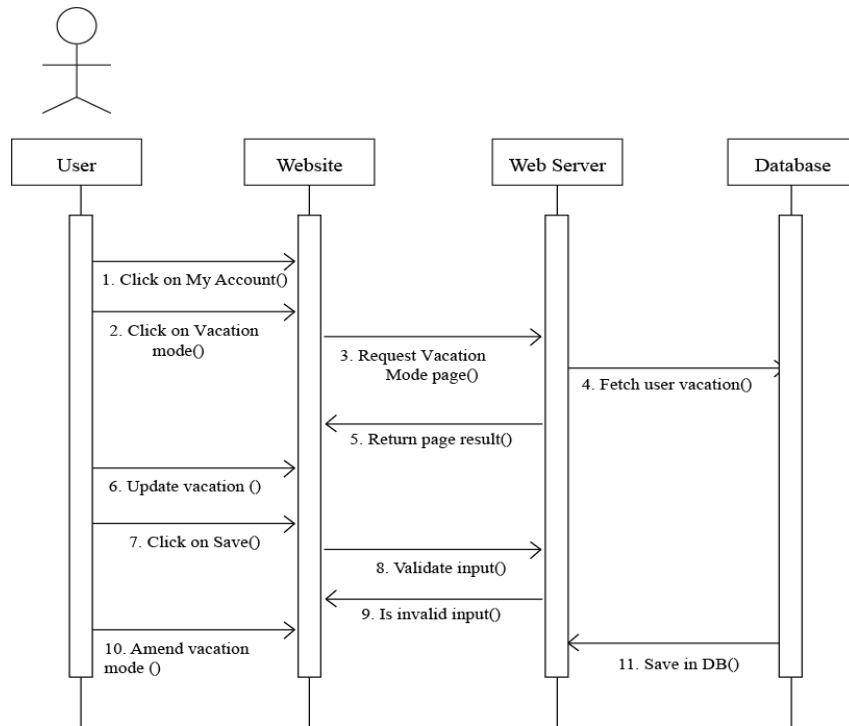


Figure 4.3.5: Sequence diagram of updating vacation status.

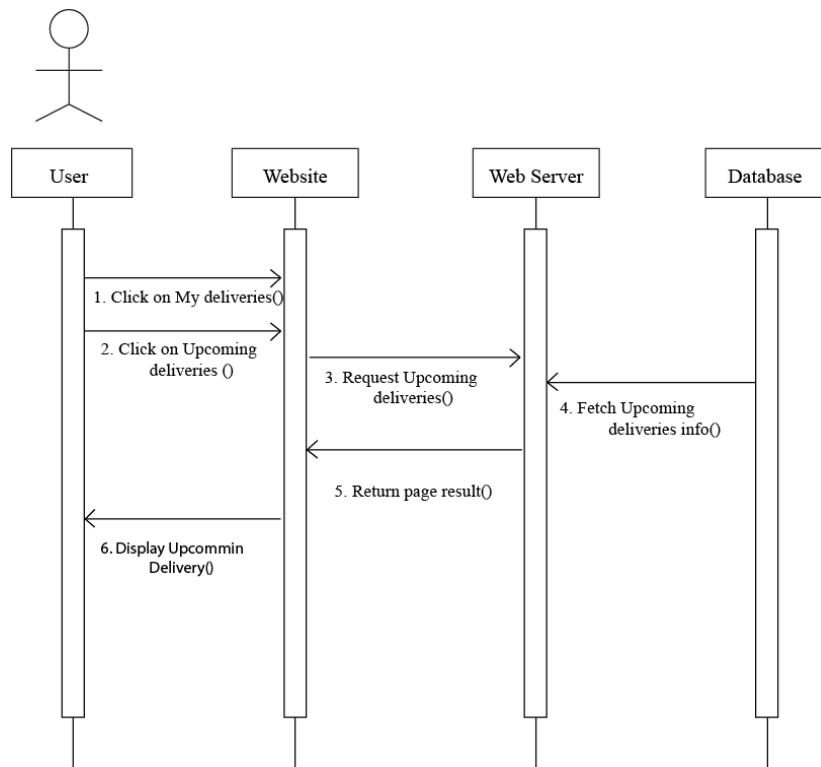


Figure 4.3.6: Sequence diagram of viewing upcoming delivery

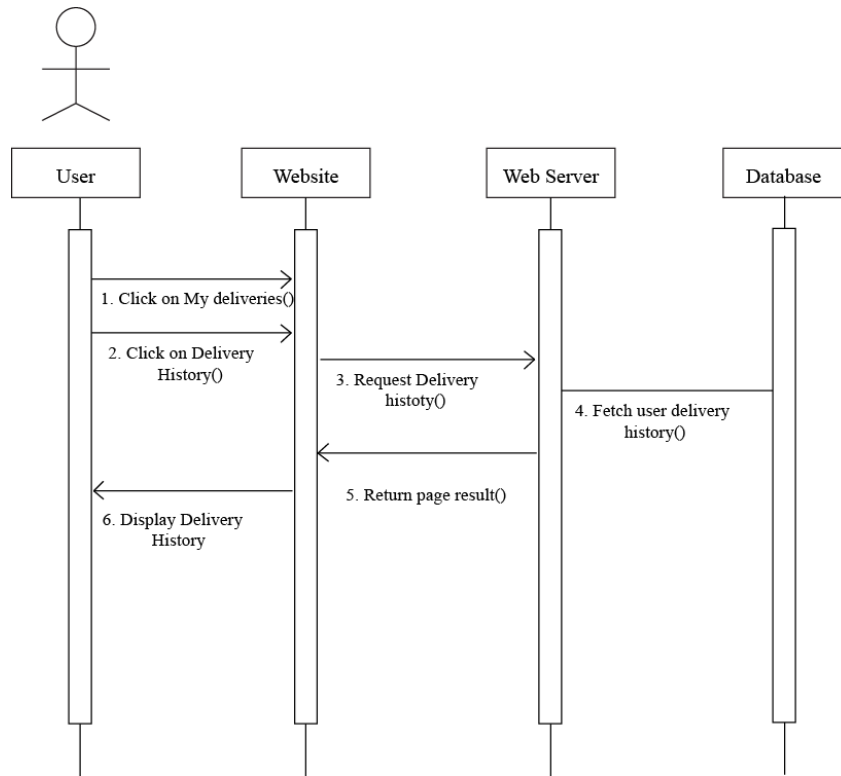


Figure 4.3.7: Sequence diagram of viewing delivery history

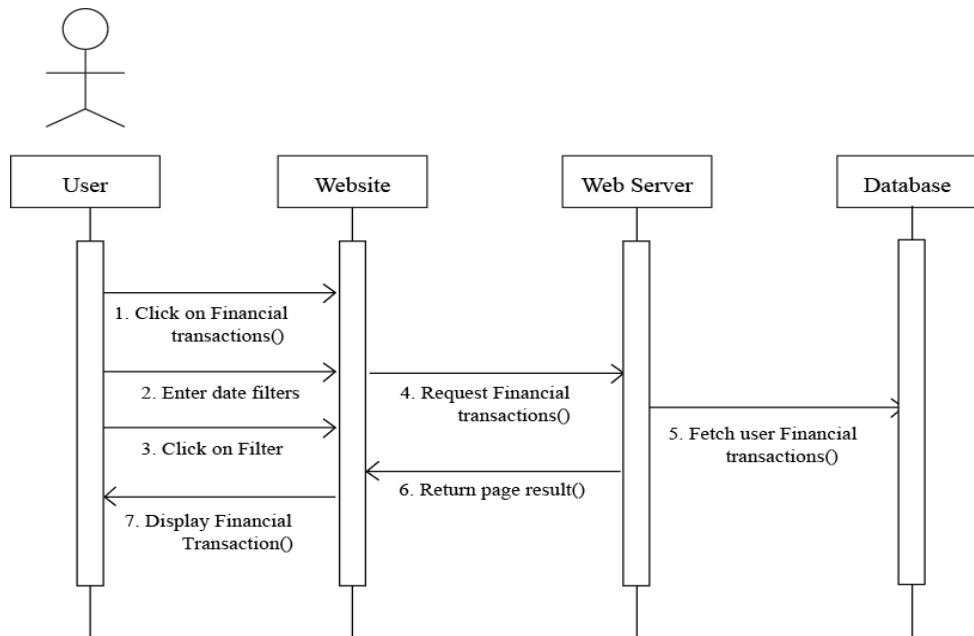


Figure 4.3.8: Sequence diagram to view financial records

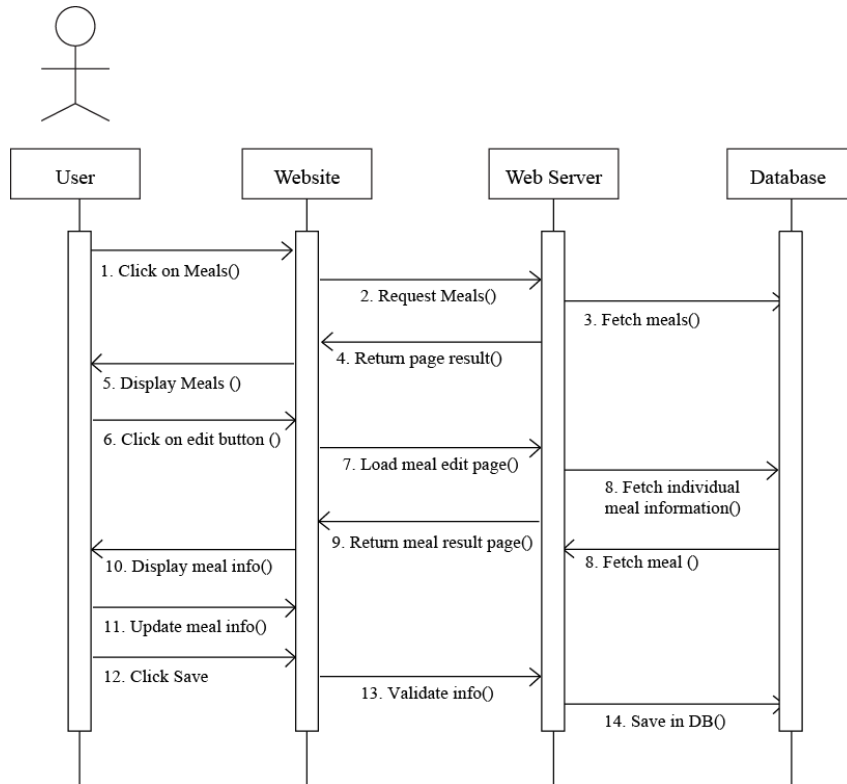


Figure 4.3.9: Sequence diagram of meals management page.

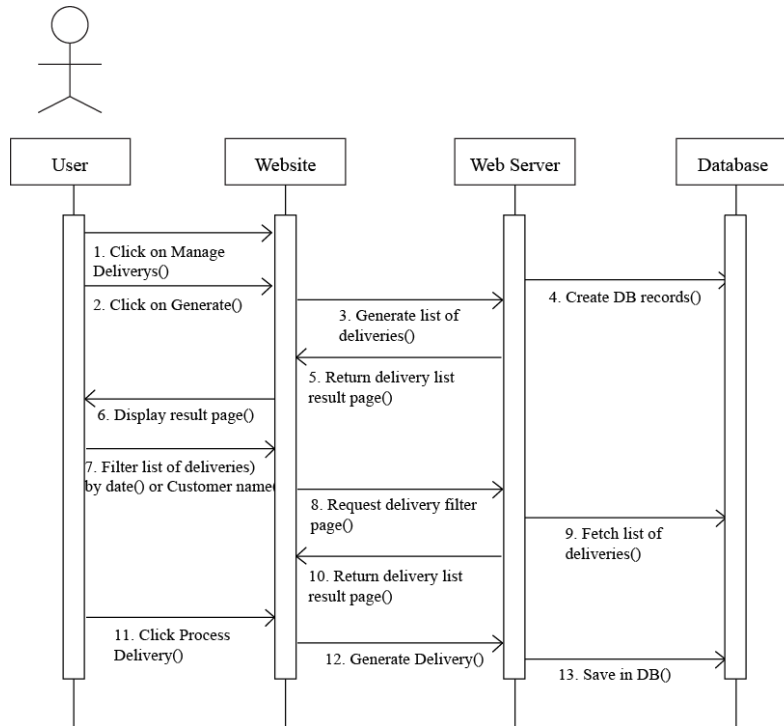


Figure 4.3.10: Sequence diagram of delivery management page.

CHAPTER 5: IMPLEMENTATION AND TESTING

This chapter will cover the theory behind software implementation and complex features that had to be researched and then implemented. It will mainly discuss the development details of the previously discussed design ideas in chapter 4. Then, later all of the functionalities of the system is thoroughly tested.

5.1 Implementation of Database

This project employs RDBMS as its data storage method. The RDBMS selected for the project is MySQL which was utilized during the development stage and the final full version at deployment stage. An interesting note regarding the choice of RDBMS due to selecting CakePHP is that this could be easily switched to other another vendor if necessary as long as it's supported by this framework. This is because the CakePHP ORM was used to construct the database schema and perform queries.

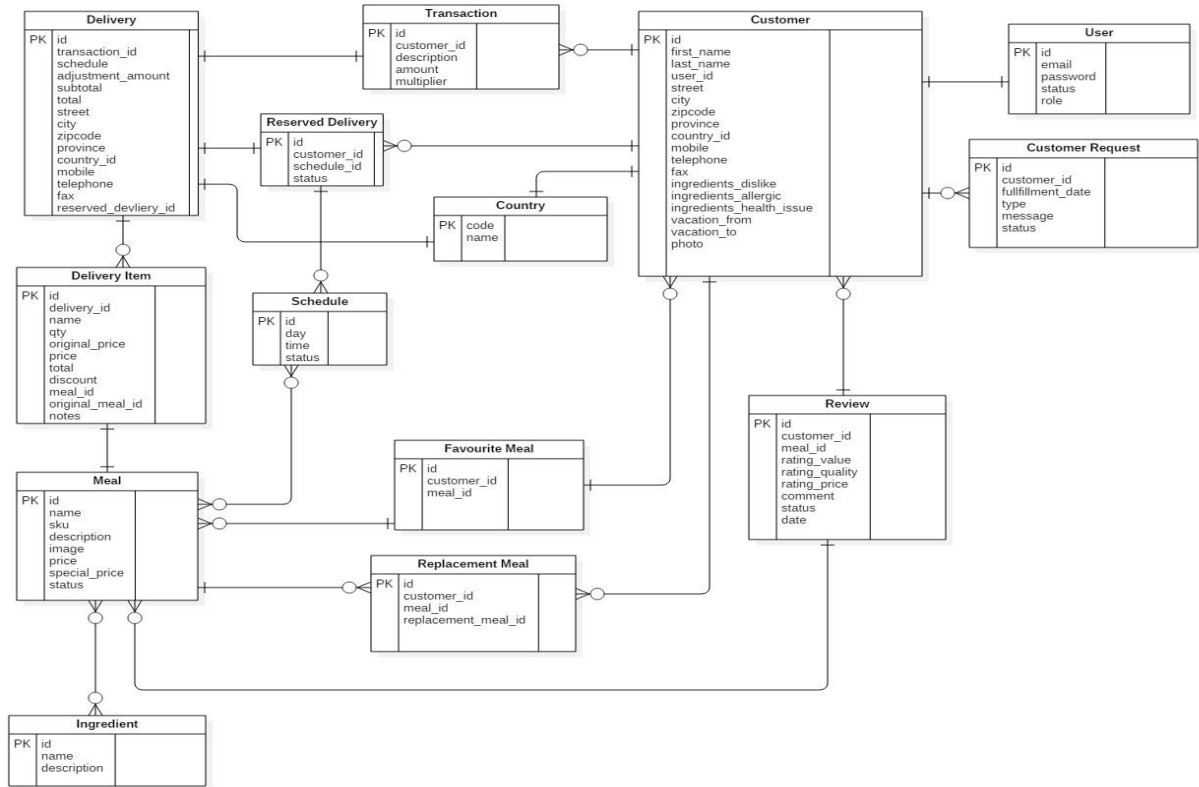


Figure 5.1: E-R diagram of the database

5.2 Implementation of the Design

Implementation of the presentation layer mainly focuses on the UI and presentation logic. The UI is often the main concern of users when using a software product. It usually gives the user the first impression as to whether they can accomplish their goal easily which is referred as Usability.

5.2.2 CakePHP Framework:

Today, there are many open source frameworks accessible for the organizations to build up their web applications. However, CakePHP framework has turned into the prime decision for this particular project, as it is found to be useful for quick and financially savvy web application development. It is an intense and powerful PHP development system that enables developers to fabricate site and web applications rapidly. CakePHP Web Service additionally gives an essential structure to sorting out records and database table names. These are some of the most important advantages of this framework that was considered when selecting this framework for this project:

- Special configuration is not necessarily required
- Backings MVC Architecture
- Database Management has very simple method which enables database administration completely simple with the assistance of CRUD
- Search engine optimization Friendly Website Development
- Simple and Fast Documentation Method
- Expansive Support of Online Community
- Supports Object Relational Mapping
- Built in ACL Functionality and Security
- CRUD scaffolding
- Ability to create tests that enables testing very easy and time efficient
- Re-usability of code

5.2.3 Utilization of MVC architecture:

The concerns of designing and developing web application are generally similar from a SE perspective as specified by. The user will interact with the user interface, often a browser to view and manipulate with the data managed at the server. However the connection between web page contents and logics need maintainability and reusability so, here comes the role of MVC [10]. The Model–view–controller shortly known as MVC is a software architectural design for implementing user interfaces on computers. The MVC pattern is a great architecture no matter whatever the language is used for development [19]. That’s why MVC was used for this particular project. Interestingly, CakePHP Supports MVC software design pattern.

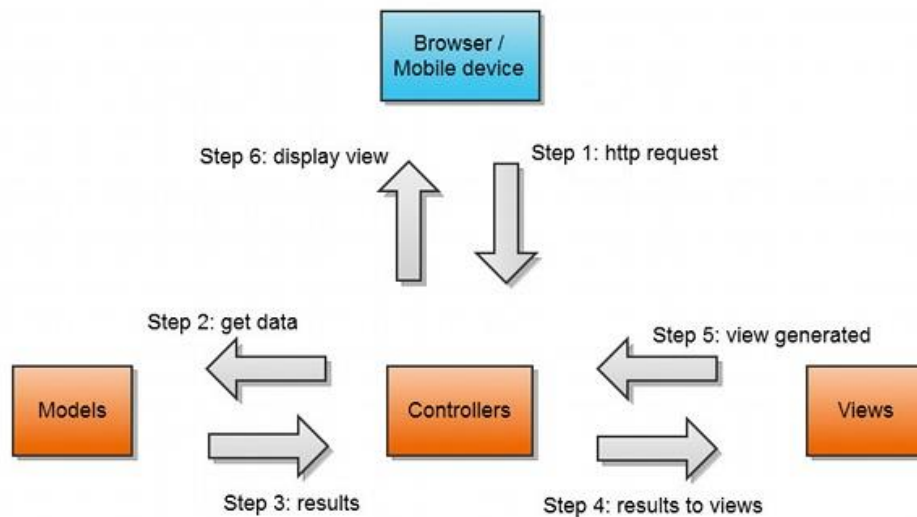


Figure 5.2.3.1: MVC architecture and the flow

5.2.4 Development process:

To realize careful planning of development activities is required and this results in the adoption of software development process model. So, it’s important to choose it very wisely depending on the type of project.

Among all of the models, the Agile software development methodology was used in this project as it represents the features that meet the requirements as follows:

Working software is delivered frequently (weeks rather than months) to ensure the project is progressing according to the requirements

- Face-to-face conversation is the best form of communication which is very important for this project
- Ensures daily cooperation between business people and developers
- Regular adaptation to changing circumstances

5.2.5 Coding Standards

In order to develop the project with CakePHP framework we have to follow certain naming conventions built into it. By following convention, most of the repeated tasks are automatically managed by this framework and save us from the maintenance nightmare of tracking config files etc.. Also, convention makes for a very uniform system development, allowing other developers to jump in and help more easily.

Controller Conventions: Controller class names are plural, CamelCased, and end in Controller. PeopleController and LatestArticlesController are both examples of conventional controller names.

The first method written for a controller might be the index() method. When a request specifies a controller but not an action, the default CakePHP behavior is to execute the index() method of that controller. For example, a request for <http://www.example.com/apples/> maps to a call on the index() method of the ApplesController, whereas <http://www.example.com/apples/view/> maps to a call on the view() method of the ApplesController.

Also visibility of controller methods can be changed in CakePHP by prefixing controller method names with underscores. If a controller method has been prefixed with an underscore, the method will not be accessible directly from the web but is available for internal use.

For example:

```
class NewsController extends ApplicationController {

    public function latest() {
        $this->_findNewArticles();
    }

    protected function _findNewArticles() {
        // Logic to find latest news articles
    }
}
```

While the page <http://www.example.com/news/latest/> would be accessible to the user as usual, someone trying to get to the page http://www.example.com/news/_findArticles/ would get an error, because the method is preceded with an underscore. We can also use PHP's visibility keywords to indicate whether or not a method can be accessed from a URL. Non-public methods cannot be accessed.

URL Considerations for Controller Names: As we've just seen a single word controllers map easily to a simple lower case URL path. For example, ApplesController (which would be defined in the file name 'ApplesController.php') is accessed from <http://example.com/apples>.

Multiple word controllers can be any 'inflected' form which equals the controller name so:

```
/redApples
/RedApples
/Red_apples
/red_apples
```

will all resolve to the index of the RedApples controller. However, the convention is that URLs are lowercase and underscored, therefore [/red_apples/go_pick](#) is the correct form

to access the `RedApplesController::go_pick` action. CakePHP URLs can be further customized with Routes Configuration.

File and Class Name Conventions: In general, filenames match the class names, which are CamelCased. So if a class is named as `MyNiftyClass`, then in CakePHP the file should be named `MyNiftyClass.php`. Below are examples of how to name the file for each of the different types of classes you would typically use in a CakePHP application:

- The Component class `MyHandyComponent` would be found in a file named `MyHandyComponent.php`
- The Model class `OptionValue` would be found in a file named `OptionValue.php`
- The Behavior class `EspeciallyFunkableBehavior` would be found in a file named `EspeciallyFunkableBehavior.php`
- The View class `SuperSimpleView` would be found in a file named `SuperSimpleView.php`
- The Helper class `BestEverHelper` would be found in a file named `BestEverHelper.php`

Each file would be located in the appropriate folder in app folder.

Model and Database Conventions: Model class names are singular and CamelCased. `Person`, `BigPerson`, and `ReallyBigPerson` are all examples of conventional model names. Table names corresponding to CakePHP models are plural and underscored. The underlying tables for the above mentioned models would be `people`, `big_people`, and `really_big_people`, respectively.

We can use the utility library `Inflector` to check the singular/plural of words. Field names with two or more words are underscored: `first_name`.

Foreign keys in `hasMany`, `belongsTo` or `hasOne` relationships are recognized by default as the (singular) name of the related table followed by `_id`. So if a `Baker` `hasMany` `Cake`, the `cakes` table will refer to the `bakers` table via a `baker_id` foreign key. For a table like `category_types` whose name contains multiple words, the foreign key would be `category_type_id`.

Join tables, used in `hasAndBelongsToMany` (HABTM) relationships between models, must be named after the model tables they will join, e.g. `users HABTM groups` would be

joined by `groups_users`, and should be arranged in alphabetical order, e.g. `apes_zoos` is better than `zoos_apes`.

All tables with which CakePHP models interact (with the exception of join tables) require a singular primary key to uniquely identify each row. If you wish to model a table that does not already have a single-field primary key, CakePHP's convention is that a single-field primary key is added to the table. You must add a single-field primary key if you want to use that table's model.

If primary key's name is not `id`, then we must set the `Model.primaryKey` attribute. CakePHP does not support composite primary keys. If directly manipulation of join table data is required we have to use direct query calls or add a primary key to act on it as a normal model. For example:

```
CREATE TABLE posts_tags (  
    id INT(10) NOT NULL AUTO_INCREMENT,  
    post_id INT(10) NOT NULL,  
    tag_id INT(10) NOT NULL,  
    PRIMARY KEY(id)  
);
```

Rather than using an auto-increment key as the primary key, it's also possible to use `char(36)`. CakePHP will then use a unique 36 character UUID (`String::uuid`) whenever a new record is saved using the `Model::save` method.

View Conventions: View template files are named after the controller functions they display, in an underscored form. The `getReady()` function of the `PeopleController` class will look for a view template in `/app/View/People/get_ready.ctp`.

The basic pattern is `/app/View/Controller/underscored_function_name.ctp`.

By naming the pieces of the application using CakePHP conventions, we can gain functionality without the hassle and maintenance tethers of configuration. Here's a final example that ties the conventions together:

Database table: "people"

Model class: "Person", found at `/app/Model/Person.php`

Controller class: "PeopleController", found at `/app/Controller/PeopleController.php`

View template, found at `/app/View/People/index.ctp`

5.3 Testing

Extensive testing approaches were utilized in this project. Different and all-encompassing testing approaches have been conducted such as unit testing, integration testing, system testing and performance testing to ensure the system is as robust as possible and fault tolerance. The following steps were taken into actions while testing:

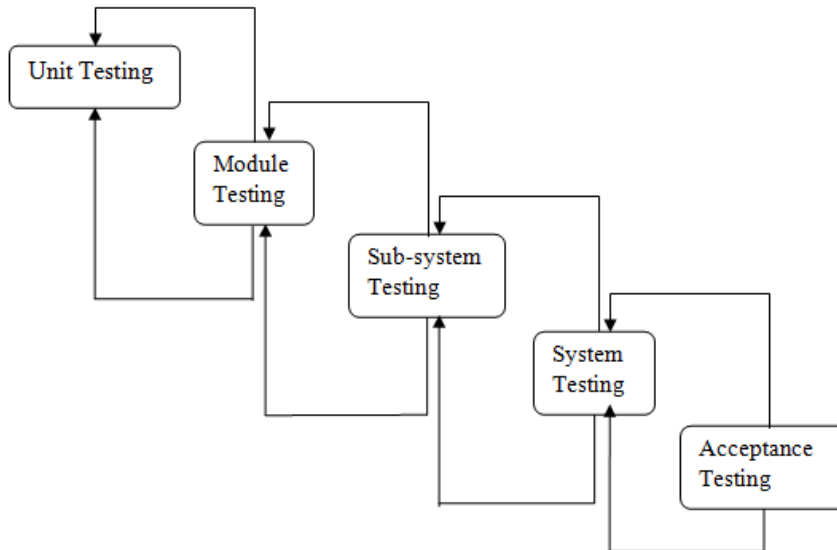


Figure 5.3.1: Testing steps

5.3.1 The testing process

The software process activities such as Design, Implementation, and Requirement Engineering etc. were tested in this phase. Because, design errors are very costly to repair once system has been started to operate, it is quite obvious to repair them at early stage of the system. So analysis is the most important process of any project.

5.3.2 Requirement Traceability

As most interested portion is whether the system is meeting its requirements or not, for that testing should be planned so that all requirements are individually tested. We checked the output of certain combination of inputs so that we can know whether it gives desirable results or not. Strictly sticking to your requirements specifications, give the path to get desirable results from the system.

5.3.3: Testing Schedule

We have tested each procedure back-to-back so that errors and omissions can be found as early as possible. Once the system has been developed fully we tested it on other machines, which differs in configuration to ensure the web application cross platform and cross browser compatible as well as supported among various devices such as PC, Notebook, Smartphone, PDA etc.

5.3.4 Testing Strategy

There are types of testing that we implemented. They are as follows-

- While deciding on the focus of testing activities, study project priorities. For example, for an on-line system, pay more attention to response time. Spend more time on the features used frequently.
- Decide on the effort required for testing based on the usage of the system. If the system is to be used by a large number of users, evaluate the impact on users due to a system failure before deciding on the effort.
- A necessary part of the test case is a definition of the expected result.
- Write test cases for invalid and unexpected as well as valid and expected input conditions.
- Thoroughly inspect the results of each test.

5.4 Test Results are Reports

In this particular project, PHP Unit framework was used to conduct the unit testing and Apache Bench was used for performance testing. The system has proved to be functional and usable after all the testing was done. The chapter has marked the end of the software development process of the project.

Table 5.4.1: Test case for login

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on login tab on homepage	Should display login tab	Shows login tab	Pass
2.	Enter Uername and Password	Should allow entering values to Username and Password input field	Allows entering values in Username and Password field	Pass
3.	Click on Login button	Should login to the system	Logged in to the system	Pass

Table 5.4.2: Test case for registration

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on registration tab on homepage	Should display registration tab	Shows registration tab	Pass
2.	Fill up the required fields	Should allow filling up the input fields	Allows filling up the input fields	Pass
3.	Click on Register button	Register user to the system	Registration was successful	Pass

Table 5.4.3: Test case for updating account information

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on My Account	Display the account	Shows account	Pass

	link	information page	information page	
2.	Update account information	Allow updating information in the input fields	Allows filling up the input fields	Pass
3.	Upload photo and test whether only specified images are supported	Should upload only JPEG, GIF, PNG images	Uploads only JPEG, GIF and PNG image	Pass
4.	Click on Save button	Should save the changes in DB	Saved successfully	Pass
5.	Invalid input validation for email	Display error in case of invalid email address	Shows error	Pass

Table 5.4.4: Test case for Meals Personalization page

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on Personalize Link	Display the Meals Personalization page	Shows Meals Personalization page	Pass
2.	Enter values into the input fields	Should display list of ingredients upon typing and allow autocomplete	Shows list of ingredients upon typing and allows to select	Pass
3.	Click on Save button	Should save the changes	Save the changes	Pass

Table 5.4.5: Test case for Meals Personalization page

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on Vacation Mode link	Display the Vacation Mode page	Shows Vacation Mode page	Pass
2.	Enter invalid input	Should not allow	Doesn't allow	Pass

	values for date range field			
3.	Enter and select valid date from the date selector	Should allow selecting	Allows selecting	Pass
4.	Click on Save buton	Update the vacation mode and display current status	Update was successful and displayed current status	Pass

Table 5.4.6: Test case for Meals Management page

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on Meals link	Display the list of available Meals	Shows the list of available Meals on the system	Pass
2.	Click on Add button	Should display form in a new window to enter meal information	Shows a new window with a form to enter meal information	Pass
3.	Enter meal information	Should only allow valid input values into the input fields	Allows only valid input values to the input fields and shows error when an invalid value is entered	Pass
4.	Upload photo	Should allow only JPEG, GIF and PNG image types and display error otherwise	Allows uploading JPEG, GIF and PNG image types and display error otherwise	Pass
5.	Click on Save button	Should save the new meal on the system	Successfully saved and redirected to the	Pass

			meal list page	
6.	Check whether the new meal is displayed in the list	Display the new meal in the Meal management page	Shows the new meal	Pass
7.	Click on Edit button	Should open a new window with the meal information and allow to edit	Opens a new window and show meal information in a form	Pass
8.	Edit meal information	Allow editing the information	Allows editing	Pass
9.	Change Photo	Should allow uploading a new photo and replace existing one	Successfully uploaded and replaced the existing image	Pass
10.	Enter ingredients	Display list of existing ingredients in a dropdown to choose from	Shows list of ingredients upon typing and allows to select them	Pass
9.	Save the changes	Should allow saving the changes and reflect on the website	Successfully saved and reflected the changes on the website	Pass

Table 5.4.7: Test case for Delivery Management page

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on Manage Deliveries link	Display the list of available Deliveries or empty if nothing was generated	Shows list of empty deliveries	Pass
2.	Click on Generate button	Should generate the list of deliveries for today	Generated the list of deliveries based on today's	Pass

			date	
3.	Click on Process button	Should create the delivery and deduct balance from the customers account that should create a new transaction automatically	New delivery was created and deducted the amount for customer's balance	Pass
4.	Verify customers address before processing order	Should check customer address automatically and display error due to missing information	Shows no error for the customer that have valid address and required information otherwise shows error	Pass
5.	Verify customer preference before placing an order	Should display error notification if there's a issue with customer's meal preference	Shows error if there's an issue with customers meal preference otherwise not	Pass
6.	Click on Delivery button	Should display the delivery page	Shows the delivery page	Pass

Table 5.4.8: Test case for Schedules Management page

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on Manage Schedule link	Display the list of available Schedules on the system	Shows the list of available schedules	Pass
2.	Click on Add button	Should display a new page with form to enter schedule information	Shows a new page with form to enter schedule information	Pass

3.	Enter ingredient information	Allow entering only valid values to the input fields	Allows entering only valid values to the input fields	Pass
4.	Click on Save button	Save the schedule on the database and reflect the changes on schedule list page	Successfully saved and displayed new schedule on the list page	Pass
5.	Click on Edit button	Allows to edit the schedule information	Allows editing schedule information	Pass
6.	Click on Delete button	Delete the selected schedule from the system	Successfully deleted	Pass

Table 5.4.9: Test case for Ingredients Management page

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on Manage Ingredients link	Display the list of available Ingredients on the system	Shows the list of available ingredients	Pass
2.	Click on Add button	Should display a new page with form to enter ingredient information	Shows a new page with form to enter ingredients information	Pass
3.	Enter ingredient information	Allow entering values to the input fields such as Name, Description	Allows entering values to the input fields	Pass
4.	Click on Save button	Save the ingredients on the database and reflect the changes on ingredient list page	Successfully saved and shows new ingredient on the list page	Pass
5.	Click on Edit	Allows to edit the	Allows editing ingredient	Pass

	button	ingredient information	information	
6.	Click on Delete button	Delete the selected ingredient from the system	Successfully deleted	Pass

Table 5.4.10: Test case for Review Management page

#	Test Condition	Expected Output	Actual Output	Pass/Fail
1	Click on Reviews link	Display the list of customer reviews	Shows the list of customer reviews	Pass
2.	Click on a specific review	Display the details of the review	Shows the details of the review	Pass

CHAPTER 7: CONCLUSION AND FUTURE SCOPES

The project has gone through a series of activities to develop such a complex solution for the computerized catering management system. After analyzing the project's requirements and research direction, a set of objectives were established. All activities done during the development of this project were attempts to accomplish these objectives.

Utilizing the agile development method also showed useful in managing the software development process. The software prototype was constantly evolving thanks to the incremental and iterative development cycle. The system was tested during every iteration; hence most defects were addressed early in the project.

The project was ambitious and time-consuming. It was such a challenge to implement as many features as possible within the very short timeframe. It has successfully satisfied the Functional Requirements (FR) and all Non-functional Requirements (NFR) of the system. These requirements have top priority and reflect the most needed features required by stakeholders. At the bottom line, the system is useable in term of the stakeholder's need and operational concerns. It has satisfied the basic goal of replacing the paper-based system in the catering business operations.

The coding pattern is kept as dynamic as possible with minimum amount of static values to make it easier for future extensions. As the current system is expected to add more functionality and dependency according to requirement changes and technology, proper coding standards and working platform have been kept in mind to produce a quality product.

REFERENCES

- [1] Rich Rosen, *Web Application Architecture: Principles, Protocols and Practices, 2nd Edition*, pp 234- 248
- [2] Jamie Munoro, *Rapid Application Development With CakePHP*, pp 172-175
- [3] Craig Larman, *Agile and iterative development*, pp 73-77
- [4] Erich Gamma, John Vlissides, Ralph Johnson, and Richard Helm, *Design Patterns: Elements of Reusable Object-Oriented Software*, pp 123-137
- [5] Andi Gutmans, Derick Rethans, and Stig Saether Bakken, *PHP 5 Power Programming*, pp 180-183
- [6] Chris Schneider and Ronald Bradford, *Effective MySQL Replication Techniques in Depth*, pp 265-267
- [7] Michael McLaughlin, *MySQL Workbench: Data Modeling & Development*, pp 143-145, pp 189-193
- [8] Russell J. T. Dyer, *MySQL in a Nutshell*, pp 221-223, pp 338-341
- [9] Leon Shklar, *Web Application Architecture: Principles, Protocols*, pp 180-183
- [10] Chris Pitt, *Pro PHP MVC*, pp 140-144
- [11] Sam Lightstone, Toby J. Teorey, and Tom Nadeau, *Database Modeling and Design: Logical Design, 1998*, pp 35- 38, 53-57
- [12] Jeffrey O. Grady, *System Requirements Analysis*
- [13] Alan M. Davis, *Software Requirements: Analysis and Specification*, pp 215-218
- [14] Richard H. Thayer, Merlin Dorfman, Sidney C. Bailin, *Software Requirements Engineering*, pp 107-112
- [15] Larry E. Wood, *User Interface Design: Bridging the Gap from User Requirements to Design*, pp 83-87
- [16] Debbie Stone, Caroline Jarrett, Mark Woodroffe, Shailey Minocha, *User Interface Design and Evaluation*, pp 331-369
- [17] Jon Duckett, *HTML & CSS: Design and Build Web Sites, 2011*, pp 428-445, 226-238
- [18] Marijn Haverbeke, *Eloquent JavaScript: A Modern Introduction to Programming, 2011*, pp 93-110
- [19] Ajit Kumar, *Software Architecture and Design Illuminated*, pp 201-234
- [20] Hassan Gomaa, *Software Modeling and Design, 2010*, pp 279-298, 29-43