



Project Name: Restaurant Management System

By

Name: Md. Riaz Ahmed

Id:161-35-141

Supervised by

Md. Shariful Islam

Lecturer(Senior Scale)

Department Of Software Engineering

Daffodil International University

A Project Submitted in Partial Fulfillment of the Requirement for the Degree of
Bachelor of Science in Software Engineering.

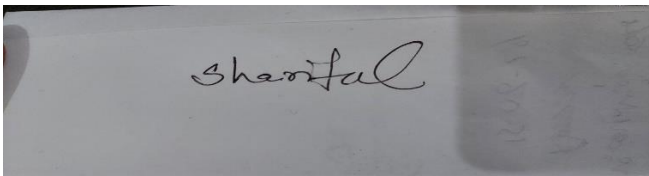
Spring-2021

DEPARTMENT OF SOFTWARE ENGINEERING

DAFFODIL INTERNATIONAL UNIVERSITY

Approval

This project titled on “**Restaurant Management System**”, submitted by Md. Riaz Ahmed Id:161-35-141 to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

A rectangular box containing a handwritten signature in cursive script that reads "Shariful".

Mr. Shariful Islam

Lecturer (senior scale)

Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

**Internal
Examiner 1**

Mr. SK. Fazlee Rabby

Lecturer

Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

**Internal
Examiner 2**

Nayeem Hasan

Lecturer (senior scale)

Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

**Internal
Examiner 3**

DECLARATION

It hereby declares that this project has been done by us under the supervision of MR. Shariful Islam, Lecturer (senior Scale), Department of Software Engineering, Daffodil International University. It also declares that neither this project nor any part of this has been submitted anywhere else for award of any degree.

Riaz

Name: Riaz Ahmed

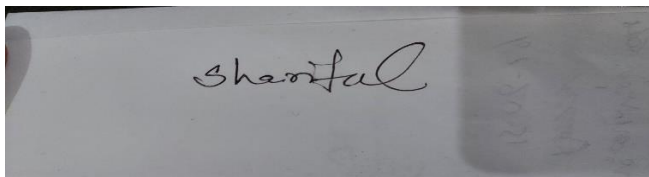
Id: 161-35-141

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

Certified by:

A rectangular box containing a handwritten signature in black ink. The signature is written in a cursive style and reads "Shariful".

Md. Shariful Islam

Lecturer

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

ACKNOWLEDGEMENT

I have taken endeavors in this project. Be that as it may, it would not have been conceivable without the kind help of numerous people. I might want to stretch out my earnest because of every one of them. I am exceptionally obligated to Daffodil International University for their direction and steady supervision by Md. Shariful Islam and in addition for giving necessary information with respect to the venture and additionally for their help in finishing the task. I would like to express my gratitude towards my parents, my batch mate, member of DIU for their kind co-operation and consolation which help me in finishing of this task. My thanks and thanks likewise go to my supervisor in building up the energetically bailed me out with him capacities.

EXCLUSIVE SUMMERY

Restaurant Management System this application will help people who wants order food. This application can be used by anyone. In this application user can login to our system, they can search for a recipe. User will get food at a lower cost from this application. User can also check food which is available on user location. This application provides food category where can user check for their respective menus. This application provides easy checkout for a food. User can message to the admin for any problem.

Appendix

APPROVAL	i
INTERNSHIP DECLARATION	ii
ACKNOWLEDGEMENT	iii
EXECLUSIVE SUMMARY	iv
LIST OF TABLES	viii
LIST OF FIGURES	ix

CHAPTER 1: INTRODUCTION.....	1
1.1 Project Overview.....	1
1.2 Project Purpose.....	1
1.2.1 Background.....	1
1.2.2 Benefits & Beneficiaries.....	2
1.2.3 Goals.....	2
1.3 Stakeholders.....	2
1.4 Project Schedule.....	2
1.4.1 Gantt chart.....	3
1.4.2 Release Plan/Milestone.....	3
CHAPTER 2: SOFTWARE REQUIREMENT SPECIFICATION.....	4
2.1 Functional Requirements.....	4
2.2 Data Requirements.....	4
2.3 Performance Requirements.....	4
2.3.1 Speed and Latency Requirements.....	4
2.3.2 Precision or Accuracy Requirements.....	4
2.3.3 Capacity Requirements.....	4
2.4 Dependability Requirements.....	5
2.4.1 Reliability Requirements.....	5
2.4.2 Availability Requirements.....	5
2.4.3 Robustness or Fault-Tolerance Requirements.....	5
2.4.4 Safety-Critical Requirements.....	5
2.5 Maintainability and Supportability Requirements.....	5

2.5.1 Maintenance Requirements.....	6
2.5.2 Supportability Requirements.....	6
2.5.3 Adaptability Requirements.....	7
2.5.4 Scalability or Extensibility Requirements.....	7
2.6 Security Requirements.....	7
2.6.1 Access Requirements.....	8
2.6.2 Integrity Requirements.....	8
2.6.3 Privacy Requirements.....	8
2.7 Usability and Human-Interaction Requirements.....	8
2.7.1 Ease of Use Requirements.....	8
2.7.2 Personalization and Internationalization Requirements.....	9
2.7.3 Understandability and Politeness Requirements.....	9
2.7.4 Accessibility Requirements.....	9
2.7.5 User Documentation Requirements.....	9
2.7.6 Training Requirements.....	10
2.8 Look and Feel Requirements.....	10
2.8.1 Appearance Requirements.....	10
2.8.2 Style Requirements.....	12
2.9 Operational and Environmental Requirements.....	13
2.9.1 Expected Physical Environment.....	13
2.9.2 Requirements for Interfacing with Adjacent Systems.....	14
2.9.3 Projectization Requirements.....	14
2.9.4 Release Requirements.....	14
2.10. Legal Requirements.....	14
2.10.1 Compliance Requirements.....	15
2.10.2 Standards Requirements.....	15
CHAPTER 3: SYSTEM ANALYSIS.....	16
3.1 Use Case Diagram.....	16
3.2 Use Case Description (for each use case).....	17
3.3 Activity Diagram (for each use case)	29
3.4 System Sequence Diagram (for each use case).....	30
Chapter 4: System Design Specification.....	31
4.1 Class Responsibilities Collaboration (CRC) Cards.....	31
4.2 Sequence Diagram (for each use case)	31
4.3 Class Diagram.....	36
4.4 Entity-Relationship Diagram.....	37

4.5 Development Tools & Technology.....	37
4.5.1 User Interface Technology.....	38
4.5.1.1 ASP.NET MVC4 Framework or Laravel Framework....	38
4.5.1.2 JQuery UI or Angular JS or Other jQuery Framework....	39
4.5.1.3 CSS Framework or Twitter Bootstrap.....	39
4.5.1.4 Font Awesome or Others.....	39
4.5.2 Implementation Tools & Platforms.....	39
4.5.2.1 Microsoft Visual Studio 2010 or PHP Storm.....	39
4.5.2.2 MySQL Community Server 5.5.....	40
4.5.2.3 .NET Runtime or Apache HTTP Server.....	40
CHAPTER 5: SYSTEM TESTING.....	41
5.1 Testing Features.....	41
5.1.1 Features to be tested.....	41
5.1.2 Features not to be tested.....	41
5.2 Testing Strategies.....	41
5.2.1 Test Approach.....	41
5.2.2 Pass/Fail Criteria.....	41
5.2.3 Suspension and Resumption.....	42
5.2.4 Testing Schedule.....	42
5.2.5 Traceability Matrix.....	42
5.3 Testing Environment (hardware/software requirements).....	42
5.4 Test Cases.....	43
CHAPTER 6: USER MANUAL.....	46
6.1 User Manual (type A user).....	46
6.2 User Manual (type B user)	46
6.3 User Manual (type C user)	48
CHAPTER 7: PROJECT SUMMARY.....	50
7.1 GitHub Link.....	50
7.2 Critical Evolution.....	50
7.3 Limitations.....	50
7.4 Obstacles & Achievements.....	50
7.5 Future Scope.....	50

List of Tables

Table01: Maintenance requirement.....	5
Table02: Supportability Requirements	6
Table03: Adaptability Requirements.....	6
Table04: Scalability or Extensibility Requirements.....	7
Table05: Use case Description.....	17

List of Figures

Figure 01: Gantt chart.....	3
Figure 02: Use Case Diagram.....	15
Figure 03: Activity Diagram.....	33
Figure 04: System Sequence Diagram.....	34
Figure 05: Sequence Diagram	35
Figure 06: Class Diagram.....	45
Figure 07: Entity-Relationship Diagram.....	46
Figure 08: User Manual.....	54

CHAPTER 1: INTRODUCTION

1.1 Project Overview

It is a web program called "Restaurant Management System." The goal of this system is to automate a restaurant's daily operations. Restaurants are businesses that provide ready-to-eat meals to customers all around the world. This method was created to give both the restaurant and the customer with a service facility. Employees in a restaurant can use this restaurant management system to manage orders. Employees at a restaurant can use this restaurant management system to handle customers, their orders, and to help them discover free tables or place orders. Food ordering and reservation table management by the customer via the web system, customer and waiter information management, menu information management, and report are among the services provided.

1.2 Project Purpose

Many restaurants nowadays run their operations entirely by hand, especially when it comes to customer ordering. Today, restaurant waiters use a paper-based manual approach to accept customer orders. This is an issue for restaurant waiters since there is a chance that client information will be lost or duplicated. It would also have an impact on the restaurant's reputation in terms of order management. Furthermore, the restaurant waiter information is held on paper by a manual system, making it difficult for restaurant administrators to locate waiter information, as well as missing the paper and arranging the schedule. Information on waiters and customers is sometimes useful to restaurant managers for future reference.

1.2.1 Background

The system for managing the restaurant business is the restaurant management system. The major goal of creating this system is to assist restaurant managers in running their businesses and to assist customers with online ordering and table reservations. Because many restaurants find it difficult to run their operations, such as client ordering and table reservations, the project is evolving. It is difficult for waiters to preserve accurate customer information while employing manual customer ordering, and customer information may be lost.

1.2.2 Benefits & Beneficiaries

Restaurant Management System approaches differ based on the type of operation, but all restaurants have one thing in common: they exist to make money. Restaurant management software assists managers in increasing earnings, lowering costs, and providing a superior customer experience. Increased efficiency and reduced mismanagement are two advantages of using the right restaurant management software.

1.2.3 Goals

The main goal of the system is to give customers with online purchasing and reservation services. The major goal of creating this system is to assist restaurant managers in running their businesses and to assist customers with online ordering and table reservations.

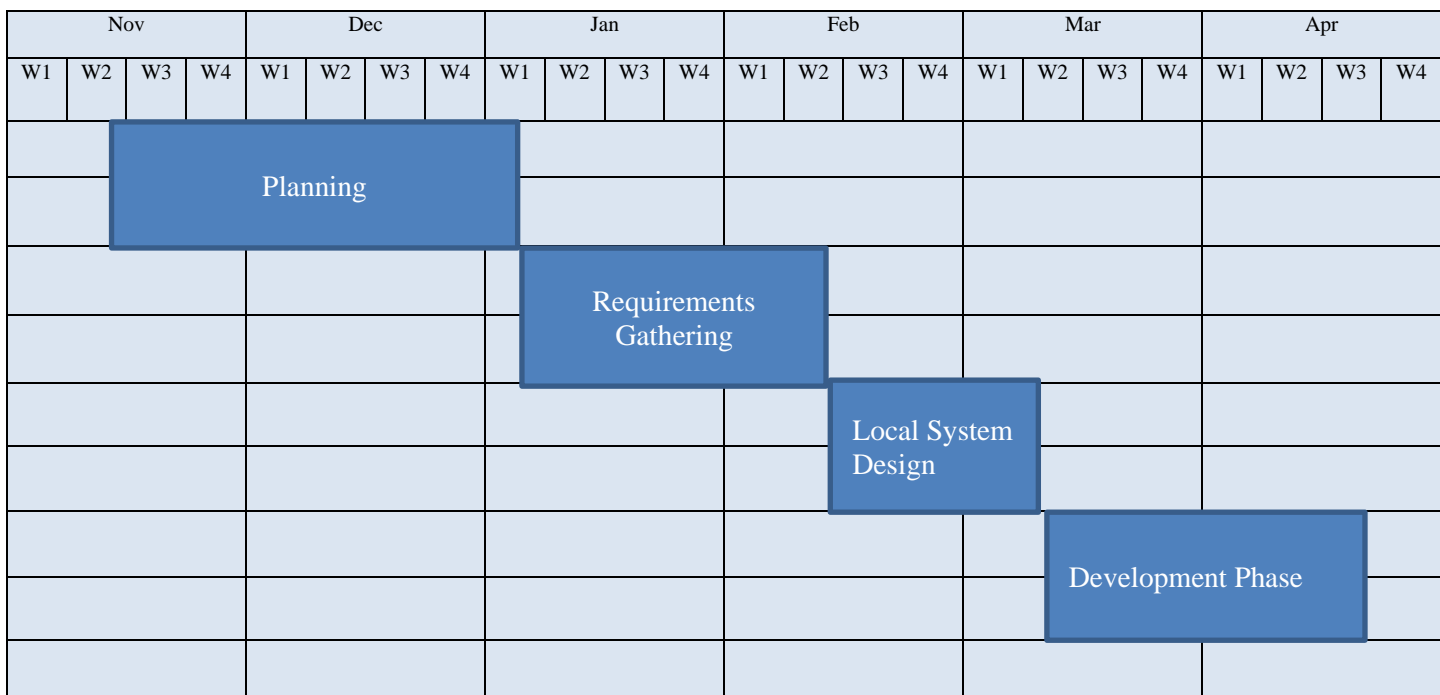
1.3 Stakeholders

Those who are using my web system and also android apps are my stakeholders. The user is stakeholders.

1.4 Project Schedule

Though our project is big so this project will take time. But I am trying to finish the web and mobile application part from the whole system for our Final Defense. So, I used (5) months to make our system. Here's the Gantt Chart. So that I can easily find out when I have finished our work

1.4.1 Grant Chart



1.5.2 Release Plan/Milestone

Release plan-1: I will try to release our system on August 20, 2021.

Because our system is not a small project. I can't finish it within a short time. So, I will update it & release the updated version next time. On our next release date, I will try to use an Appointment system & will try to make the system more reliable

CHAPTER 2: SOFTWARE REQUIREMENT SPECIFICATION

2.1 Functional Requirements

Restaurant Management is a smart web application, this provides a registration and login for User. Users can register by giving their necessary details. After successful registration, the user can log in by giving a username and password. Then the user has to fill up their profile. A user can see other user-profiles and also post.

2.2 Data Requirements

Data requirements refer to those data which are needed to build the system model. For my project, I need to focus on some points such as:

- 1) User Information:
 - User Identification
 - Signup/Login
 - Authentication
 - View another user information

- 2) Recipe:
 - Recipe Creation
 - Modify Recipe info

2.3 Performance Requirements

Performance requirements are one of the most important things for an application. If the performance is well, then the Application will be more useable.

2.3.1 Speed and Latency Requirements

Speed: The system will run at a high speed.

Latency: The latency will be also user-friendly.

2.3.2 Precision or Accuracy Requirements

The precision and the accuracy is for the research purpose however we know And implemented the project. This project is fully functional in localhost access.

2.3.3 Capacity Requirements

User capacity: About 10000 people can access the website at a time. The capacity will be increased according to the users

2.4 Dependability Requirements

However, this project has dependability because it depends on the frameworks and users.

2.4.1 Reliability Requirements

Reliability is an important non-functional requirement for most software products so a software requirements specification (SRS) should contain a reliability requirement, and most do. A software product will fail under certain conditions, with certain inputs, and given the same inputs and conditions will fail every time until the cause of the failure is corrected. So, I tried to make our software more reliable so that users can easily use our system & get better service from us.

2.4.2 Availability Requirements

This project idea is unique but the numbers of availability of the food will increase based on the increments of the users. We plan to publicly launch the sites and we are capable to manage large datasets.

2.4.3 Robustness or Fault-Tolerance Requirements

This project has the tolerance of the least user right now as now it is in the localhost so based on the computer requirement and the configuration it can manage the datasets.

2.4.4 Safety-Critical Requirements

Safety and privacy are maintained in the project. In our privacy policy, this is mentioned that no data will be disclosed or shared publicly.

2.5 Maintainability and Supportability Requirements

At least one backup server with the same configuration as in the main server is also recommended for fault tolerance and better performance. Separate storage (with backup) for the database, electronic documents, and manuscripts is also recommended. Multiple computing nodes with storage are required for

high availability and to enhance the performance of the application. Again, after a certain period, the preliminary manuscript files and other files related to that can be deleted manually from the database to increase the performance.

2.5.1 Maintenance Requirements

Requirement No.	Requirement
L-1	While login matches the username with user type
L-2	User will get the functionalities of his/her type
L-3	Login time should be stored in the log file

2.5.2 Supportability Requirements

Requirement No.	Requirement
A-1	Admin will log in using email & password
A-2	Admin will add recipe name
A-3	Admin can add international food name of the recipe.

2.5.3 Adaptability Requirements

Requirement No.	Requirement
AI-1	Admin & user both can add their personal information.
AI-2	Users may need to add their information.
AI-3	Admin needs to add their specialty information.

2.5.4 Scalability or Extensibility Requirements

Requirement No.	Requirement
CI-1	customer can check their order
CI-2	Admin can check their information & can add edit.
CI-3	Then all update is saving in the database

2.6 Security Requirements

Each time there is a security violation. For this reason, when the user will register, the user will receive a mail & have to verify this for login. Otherwise, a user can't log in to the main option.

2.6.1 Access Requirements

To access the website a user must need the credentials to access the website, otherwise, the user only can view the website can't order or purchase the recipe or the other access.

2.6.2 Integrity Requirements

The operational demand definition method includes the subsequent activities:

1. Identify stakeholders who will or have an interest in our system. So mainly our stakeholders are Seller & customer. They should have Register & Login for their other activities.
2. Establish measures of effectiveness and suitability, so that users can easily access & can use our system.

A system of dependableness relies upon a stable atmosphere. The look of the environmental system for your information center should make sure that every system will operate faithfully whereas, remaining at intervals the vary of its in operation specifications. therefore, I attempted to form our systems atmosphere additional reliable so that users will simply access the system with no delay.

2.6.3 Privacy Requirements

Privacy is an important part of the business model or any website. We ensure the privacy and safety of the database that is going to be stored in the restaurant management database. There will be no pirates with the data.

2.7 Usability and Human-Interaction Requirements

It's very easy to use the website. It's very user-friendly so to interact with the website a user doesn't need anything besides the internet connection to access the website. To order any food or service a user must need the credential user name and password.

2.7.1 Ease of Use Requirements

The user interface must be familiar to users, and so may need to follow a

single set of rules consistent with those of the operating system, or other mainstream applications. These days, most vendors do follow this good practice, and it is much less an issue than it used to be.

2.7.2 Personalization and Internationalization Requirements

The structure of your e-commerce system, including the software itself, the externally visible properties of the user interface, and the relationships between them.

Consider your new requirements for international markets, finding the balance of what is not in your e-commerce site that needs to be added

Likewise, examine what is in your site's code that needs to be changed to support the markets.

2.7.3 Understandability and Politeness Requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behavior under specific conditions. For instance:

A search feature allows a user to hunt among various invoices if they want to credit an issued invoice.

2.7.4 Accessibility Requirements

Accessibility focuses on how a disabled person accesses or benefits from a site, system, or application. Accessibility is an important part of designing our site and should be considered throughout the development process.

2.7.5 User Documentation Requirements

The user requirement(s) document (URD) or user requirement(s) specification (URS) is a document usually used in software engineering that specifies what the user expects the software to be able to do.

Once the required information is completely gathered it is documented in a URD, which is meant to spell out exactly what the software must do and becomes part of the contractual agreement. A customer cannot demand features not in the URD, while the developer cannot claim the product is ready if it does not meet an item of the URD.

The URD we used as a guide for planning cost, timetables, milestones, testing, etc. The explicit nature of the URD allows customers to show it to various stakeholders to make sure all necessary features are described.

2.7.6 Training Requirements

Identify and document the types of training required. More than one type of training may be required for a training group. The following suggests training appropriate for:

1. application staff,
2. operations staff,
3. technical staff.
4. Tips and Hints

The project delivery team may not conduct every training course but may advise the customer as to the training required and recommend applicable vendor training courses.

2.8 Look and Feel Requirements

The look and feel requirements describe the intended spirit, the mood, or the style of the product's appearance. These requirements specify the intention of the appearance and are not a detailed design of an interface

2.8.1 Appearance Requirements

Appearance:

A site must be visually appealing, polished, and professional. Remember, it's reflecting your company, your products, and your services. Our website may be the first, and only, impression a potential customer receives of your company.

An attractive site is far more likely to generate a positive impression and keep visitors on your site once they arrive. As businesses large and small continue to populate the web, your challenge is to attract and keep users' attention. Ideas like this are what PR professionals pay attention to keep our businesses successful.

Functionality:

Every component of your site should work quickly and correctly. Broken or poorly constructed components will only leave your visitors frustrated and disillusioned with your company. Across the spectrum, everything should work as expected, including hyperlinks, contact forms, site search, event

registration, and so on.

Error-free copy:

Remember the exposure your website will get. Double-check your facts and figures, as you don't know who may be quoting you tomorrow. Nor do you want to be recognized or remembered for typos, incorrect grammar and punctuation, or misspellings. Spelling mistakes and bad grammar are as unforgivable on a website as they are in other company materials.

Usability:

A critical, but often overlooked component of a successful website is its degree of usability. Your site must be easy to read, navigate, and understand. Some key usability elements include:

Simplicity:

The best way to keep visitors glued to your site is through valuable content, good organization, and attractive design. Keep your site simple and well organized.

Fast-loading pages:

A page should load in 20 seconds or less via dial-up; at more than that, you'll lose more than half of your potential visitors.

Minimal scroll:

This is particularly important on the first page. Create links from the main page to read more about a particular topic. Even the Search Engines will reward you for this behavior.

Consistent layout:

Site layout is extremely important for usability. Use a consistent layout and repeat certain elements throughout the site.

Prominent, logical navigation:

Place your menu items at the top of your site, or above the fold on either side. Limit your menu items to 10 or fewer. Remember, your visitors are in a hurry -- don't make them hunt for information.

Descriptive link text:

Usability testing shows that long link text makes it much easier for visitors to find their way around a site. Long, descriptive link text is favored by Search Engines, too. Backlinks are important to give users a sense of direction and to keep them from feeling lost. Use a site map, and breadcrumbs, if necessary.

Cross-platform/browser compatibility:

Different browsers often have different rules for displaying content. At a minimum, you should test your site in the latest versions of Internet Explorer (currently, versions 8 and 9), as well as Firefox and Safari.

Screen Resolution:

Screen resolution for the typical computer monitor continues to increase. Today, the average web surfer uses a resolution of 1024 x 768 pixels. However, you need to make sure that what looks good in this setting will also work nicely for other resolutions.

Search Engine Optimized (SEO)

There are hundreds of rules and guidelines for effective search engine optimization, and this isn't the place to cover them all. For starters, follow these simple rules:

Include plenty of written content in HTML format. Don't use Flash, JavaScript, or image-only objects for your navigational items.

Use your important keywords frequently and appropriately in your copy.

Minimize the use of tables and use Cascading Style Sheets for layout and positioning; keep your HTML code clutter-free.

We've barely scratched the surface of what makes a website most effective. However, by following these simple guidelines, you will be headed in the right direction. Visit our section on website design and development for more information, or send us an e-mail. We'd be happy to answer your questions!

2.8.2 Style Requirements:

we are always looking for ways to improve how they manage the look, feel, and complexity when building a website. This can be difficult when we have multiple team members and we are two people in our team so we maintain the below guideline for our project

Style guide:

A style guide is a document of code standards that details the various elements and patterns of a site or application. It is a one-stop place to see all visual styles of the site such as headers, links, buttons, color pallets, and any visual language that is used on the site.

2.9 Operational and Environmental Requirements:

The operational demand definition method includes the subsequent activities :

1) Identify stakeholders who will or have an interest in our system. So mainly our stakeholders are seller & customer. They should have Register & Login for their other activities.

2) Establish a measure of effectiveness and suitability, so that users can easily access & can use our system.

A system of dependableness relies upon a stable atmosphere. The look of the environmental system for your information center should make sure that every system will operate faithfully whereas, remaining at intervals the vary of its in operation specifications. therefore, attempted to form our systems atmosphere additional reliable so that users will simply access the system with no delay.

2.9.1 Expected Physical Environment

Design is fundamentally an innovative process. The methods discussed in this chapter are intended to support the identification and exploration of design alternatives to meet the requirements revealed by analyses of opportunity space and context of use. The methods are not a substitute for creativity or inventiveness. Rather they provide a structure and context in which innovation can take place. We begin with a discussion of the need for and the methods used to establish requirements based on the concept of user-centered design. The types of methods included here are work domain analysis, workload assessment, situation awareness assessment, participatory design; contextual design; physical ergonomics; methods for analyzing and mitigating fatigue, and the use of prototyping, scenarios, persona, and models, and simulations. As with the descriptions in Chapter 6, each type of method is described in terms of uses, shared representations, contributions to the system design phases, and strengths, limitations, and gaps. These methods are grouped under design because their major contributions are made in the design phase; however, it is important to note that they are also used in defining the context of use and in evaluating design outcomes as part of system operation.

2.9.2 Requirements for Interfacing with Adjacent Systems

Motivation:

Requirements for the interfaces to other applications often remain undiscovered until implementation time. We avoid a high degree of rework by discovering these requirements early.

Fit Criterion

For each inter-application interface, specify the following elements:

- The data content
- The physical material contents
- The medium that carries the interface
- The frequency
- The volume

2.9.3 Projectization Requirements

In the projection requirements, we have followed the following guideline for our project work.

Content

Any requirements that are necessary to make the product into a distributable or salable item.

Considerations

Some products have special needs to turn them into salable or usable products. might consider that the product has to be protected such that only paid-up customers can access it. Ask questions of our marketing department to discover unstated assumptions that have been made about the specified environment and the customers' expectations of how long installation will take and how much it will cost.

2.9.4 Release Requirements

To publish a website there are so many checks list need to check the UI design, any kind of error all the functionality is working properly or not. The database connection and stored process all the valid and legal information.

2.10. Legal Requirements

The Act requires you to disclose certain information about the identity of your company on your website. This information doesn't need to be on every page, but it does need to be easily found so it will typically go on our Contact Us page. The footer section of your pages will be useful to both users, and for your search engine optimization:

2.10.1 Compliance Requirements

The terms and conditions agreement is not only for customers but also highly important for protecting your business by making sure customers know their rights and responsibilities.

It helps you avoid customer uncertainty and misunderstandings, so it should be written with absolute clarity about what should be done in any given situation. Consumers should be informed about all the issues that come with shipping and delivery, the return policy, and the privacy policy as well as information about pricing, payment, taxes, and so on.

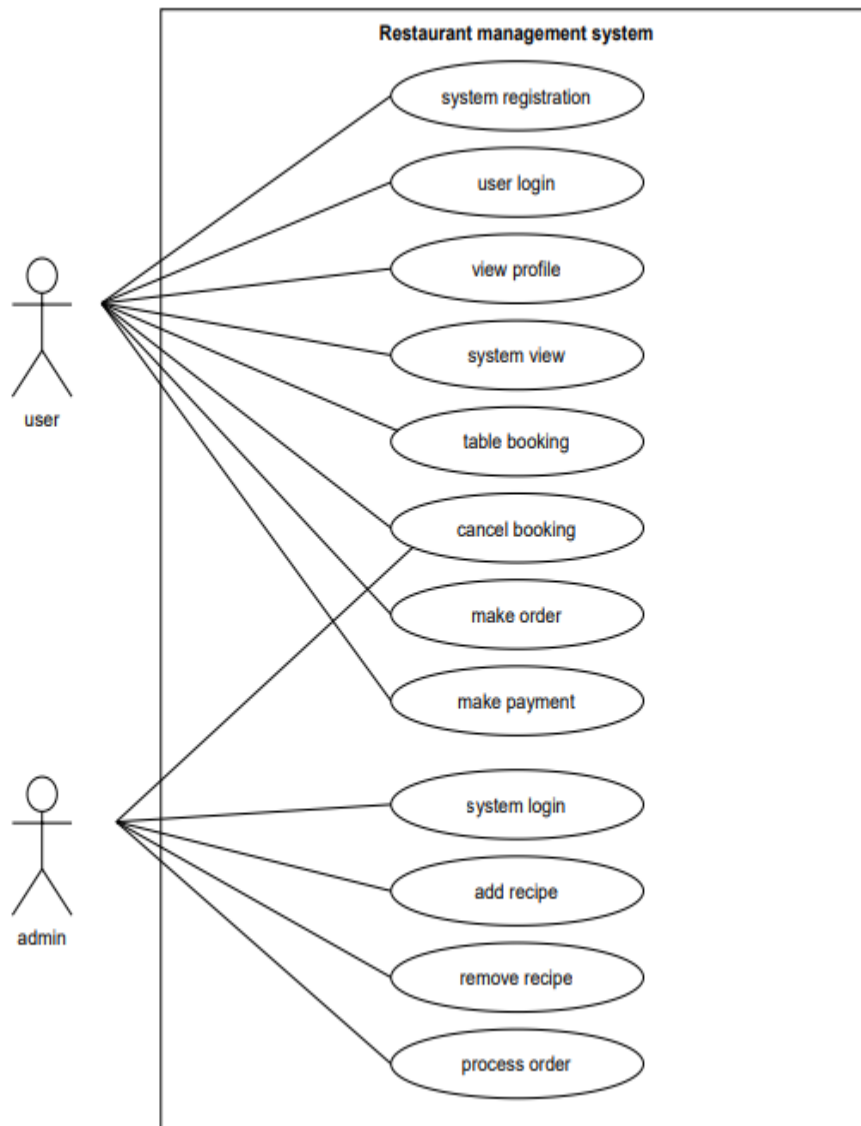
2.10.2 Standards Requirements

Tech Requirements & Getting Setup The “Right Way”

1. Domain Name
2. Hosting
3. The 2 most common types of hosting you will come across are
4. Website Platform
5. Website Speed
6. SSL Security
7. SEO Friendly
8. Tracking and Analytics

CHAPTER 3: SYSTEM ANALYSIS

3.1 Use Case Diagram



3.2 Use case description for each use case

Use Case Name:	Complete registration	
Scenario:	Complete a registration.	
Triggering Event:	Customer wants to login into the system	
Brief Description:	A new customer needs to first register into the system before performing any transaction.	
Actors:	Customer	
Related Use Case:	none	
Stakeholders:	System management	
Preconditions:	An unregistered customer	
Post conditions:	Registered customer	
	<ol style="list-style-type: none"> 1. The customer clicks the register button on the Home Page. 2. The system displays the register page. 3. The customer enters all of the required information. 4. The customer clicks the send button. 	<ol style="list-style-type: none"> 4.1 The system checks that all of the required information were entered. <ol style="list-style-type: none"> a. If yes, the system updates the customer's record in the customer and account tables in the database. b. System displays OK message.
Exception Conditions:	1.if customer don't fill up any block of the form, then customer cannot complete registration	

Use Case Name:	User login	
Scenario:	Customer have to login to perform a transaction	
Triggering Event:	When customer want to do a transaction	
Brief Description:	A customer needs to log-in into the system before performing any transaction.	
Actors:	Customer	
Related Use Case:	none	
Stakeholders:	System management	
Preconditions:	A registered user	
Post conditions:	Successful login	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> 1. The customer clicks the log-in button on the home page. 2. The customer enters his/her user ID and password. 3. The customer clicks the OK button. 4. customer is an authorized user; the system displays the Personal Home Page to the customer 	<ol style="list-style-type: none"> 1.1. The system displays the Log-in Page 3.1. The system validates the log-in information against the account table in the database. 4.1 The system displays the personal home page to the customer.
Exception Conditions:	1.if customer cannot login, then the customer cannot perform any transaction	

Use Case Name:	View Profile	
Scenario:	A customer can her view profile.	
Triggering Event:	View profile	
Brief Description:	When user click the profile button then website show the her profile.	
Actors:	Customer	
Related Use Case:	System login, view profile	
Stakeholders:	System management	
Preconditions:	User have logged-in.	
Post conditions:	User change profile	
Flow Of Events:	Actor	System
	1. The user can her profile and details information her dashboard.	1.1. The system displays her profile on the web Page.
Exception Conditions:	1. If user do not register then user cannot get her profile which they want.	

Use Case Name:	Make Order	
Scenario:	A customer can order product to purchase.	
Triggering Event:	After selecting the Product	
Brief Description:	When user add to cart button click then cart store, and cart details menu show	
Actors:	Customer	
Related Use Case:	System login, checkout, product, cart, order	
Stakeholders:	System management	
Preconditions:	User have logged-in.	
Post conditions:	User make a choice for product	
Flow Of Events:	Actor	System
	2. The customer enters the keyword for a product and clicks the search button on the personal home Page. 3. The customer chooses the desired products and clicks add to cart button on the web page.	1.2. The system displays the matching products on the web Page. 2.1. The system adds the products into the customer's order table in the database.
Exception Conditions:	2. If user do not order product, then user cannot get product.	

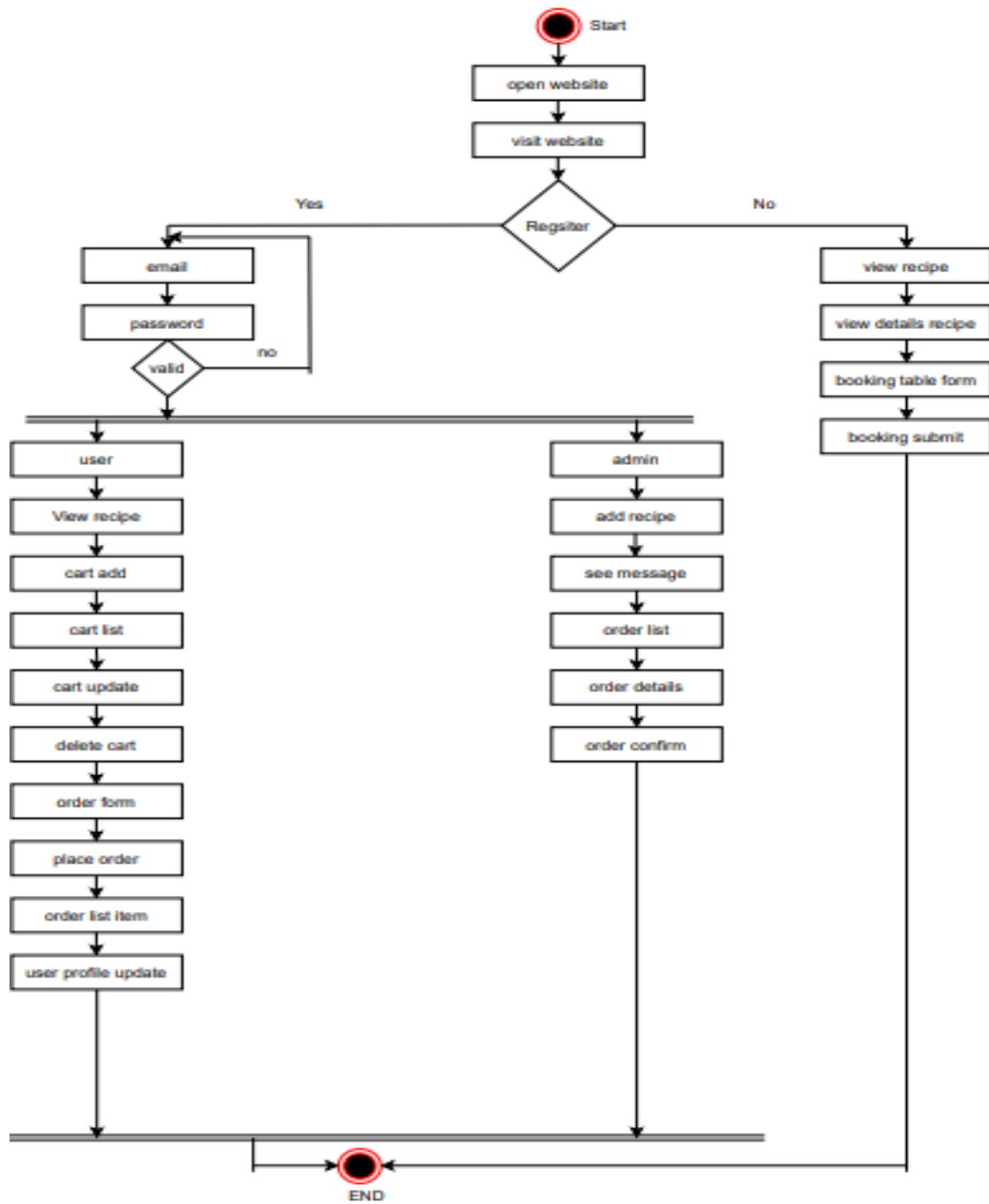
Use Case Name:	make payment	
Scenario:	A customer can purchase the product in his/her shopping cart	
Triggering Event:	When a customer wants to buy a product	
Brief Description:	User can check out the Product when checkout success then user can see her order status, after ordering the required product, a customer can pay the money	
Actors:	Customer	
Related Use Case:	none	
Stakeholders:	System management,	
Preconditions:	The user has logged in and has at least have one Product in the cart	
Post conditions:	Customer checked out the Product	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> 1. The customer clicks the checkout button on the web page. 2. The customer checks the order list for any inconsistency. If nothing found, customer clicks the proceed button. 3. The customer enters the relevant payment method information and clicks the ok button. 4. The customer checks that all information is correct and then check out the Product. 	<ol style="list-style-type: none"> 1.1. The system displays the products in the order table of the customer on the web Page. 3.1. The system displays the Invoice page. 3.1 The system checks that the payment method is valid. Then, the system displays the delivery details page. 3.2 The system checks that the payment system is valid. Then, the system displays the delivery details page. 4.1 The system will display the check-out information for confirmation.
Exception Conditions:	<ol style="list-style-type: none"> 1. If Customer don't login to the system, then the customer cannot check out the Product. 2. If customer don't give information about payment method, then the customer cannot check out the Product. 3. If customer don't give information about delivery location, then the customer cannot check out the Product. 	

Use Case Name:	System View	
Scenario:	A customer can book table, pre-order, view user interface.	
Triggering Event:	Customer can all menus button access in frontend	
Brief Description:	A Customer Can see the service and get the service when he/she contact us. A Customer Can see the whole recipe in website.	
Actors:	Customer	
Related Use Case:	System login, System view	
Stakeholders:	System management	
Preconditions:	The user has logged-in	
Post conditions:	The Customer Can see the all-product view log in.	
Flow Of Events:	Actor	System
	1. The Customer can see the recipes when click the recipes.	1.1 The system displays the recipes this website
Exception Conditions:	1. If a user does not login to the system, then user cannot buy the Product. 2. If a user does not input correct about the products, he going to buy then he cannot buy the Product.	

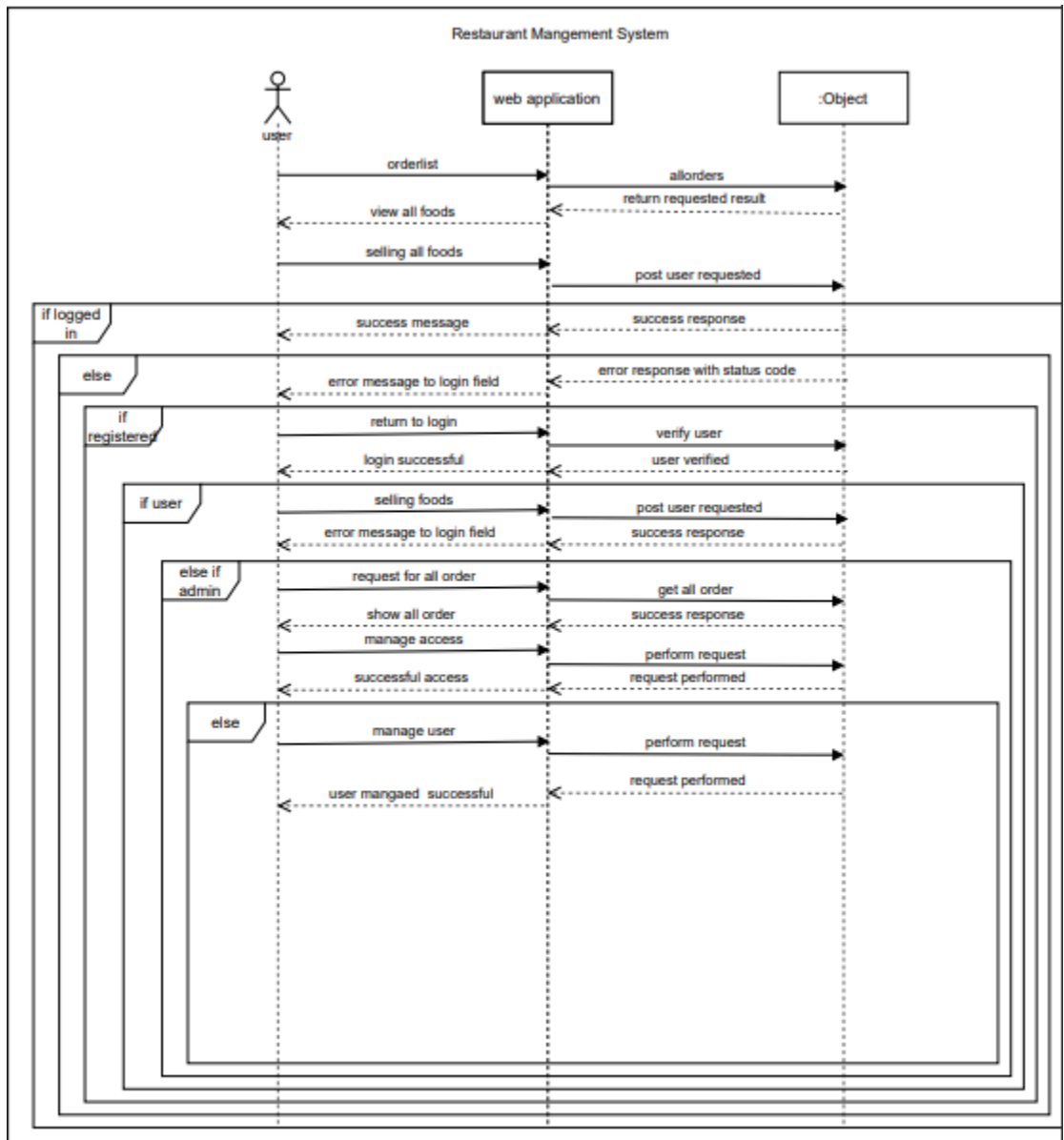
Use Case Name:	Table booking	
Scenario:	A customer can table booking.	
Triggering Event:	Table booking	
Brief Description:	When user click the booking button then website show booking form.	
Actors:	Customer	
Related Use Case:	System login, table booking	
Stakeholders:	System management	
Preconditions:	Non register.	
Post conditions:	User show table booking	
Flow Of Events:	Actor	System
	4. The user can book table	1.3. The system displays table booking details
Exception Conditions:	3. If user non register can book table.	

Use Case Name:	Add Recipe	
Scenario:	Admin want to control a this website recipes	
Triggering Event:	It occurs when admin want to manage a recipes	
Brief Description:	Admin can edit their recipes like adding new recipes, remove a recipes and update recipes status.	
Actors:	Admin	
Related Use Case:	System login	
Stakeholders:	Admin management	
Preconditions:	Admin check a recipes	
Post conditions:	Admin updated the status of the recipes	
Flow Of Events:	Actor	System
	<ol style="list-style-type: none"> 1. Admin enter the system after login 2. Admin add a new recipe. 3. Admin remove a recipe. 4. Admin update the status of a recipes which is unavailable. 	<ol style="list-style-type: none"> 2.1 System added a new recipe 4.1 System shows the recipes. <ol style="list-style-type: none"> a. System updates the recipes status
Exception Conditions:	<ol style="list-style-type: none"> 5. If admin do not login, then admin cannot edit a recipes status 6. If user give wrong information about then admin will not approve the recipes 	

3.1 Activity Diagram (for each use case)



3.2 System Sequence Diagram



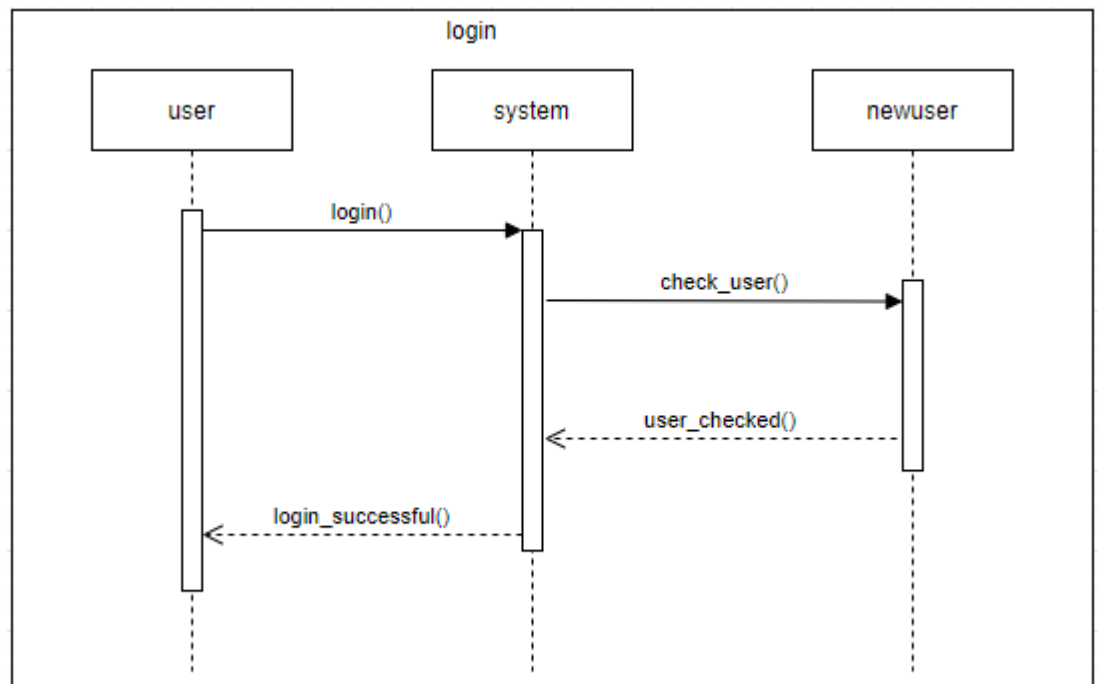
CHAPTER 4: SYSTEM DESIGN SPECIFICATION

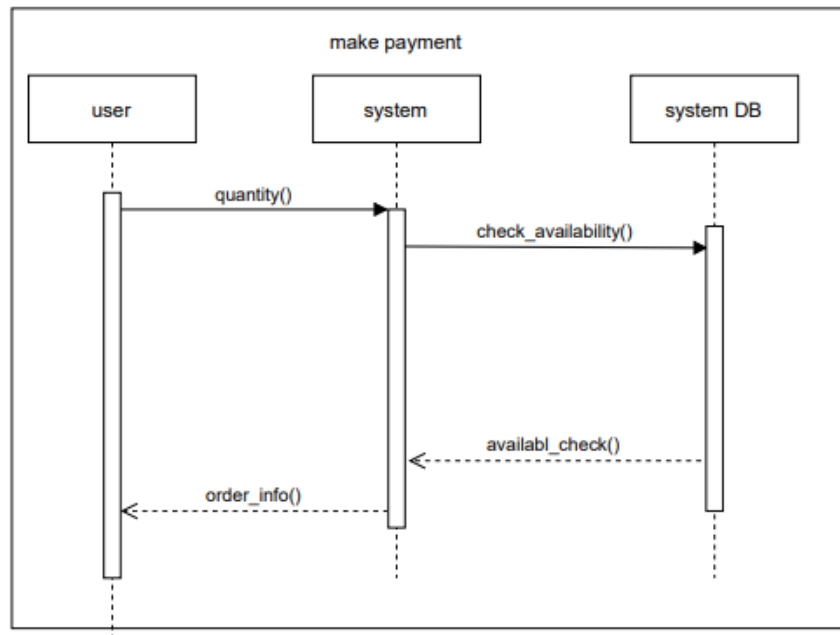
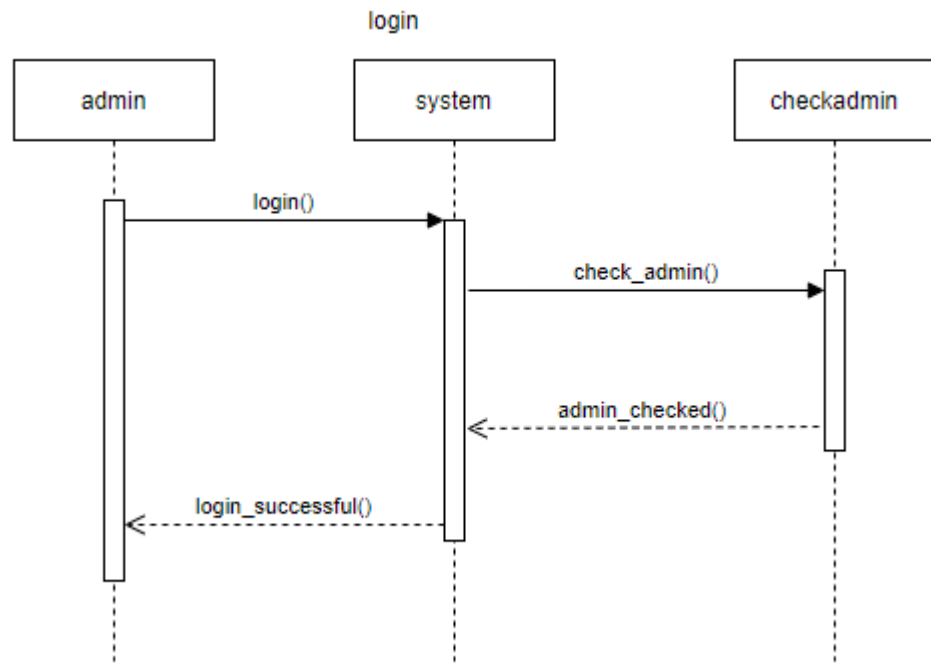
4.1 Class Responsibilities Collaboration (CRC) Cards

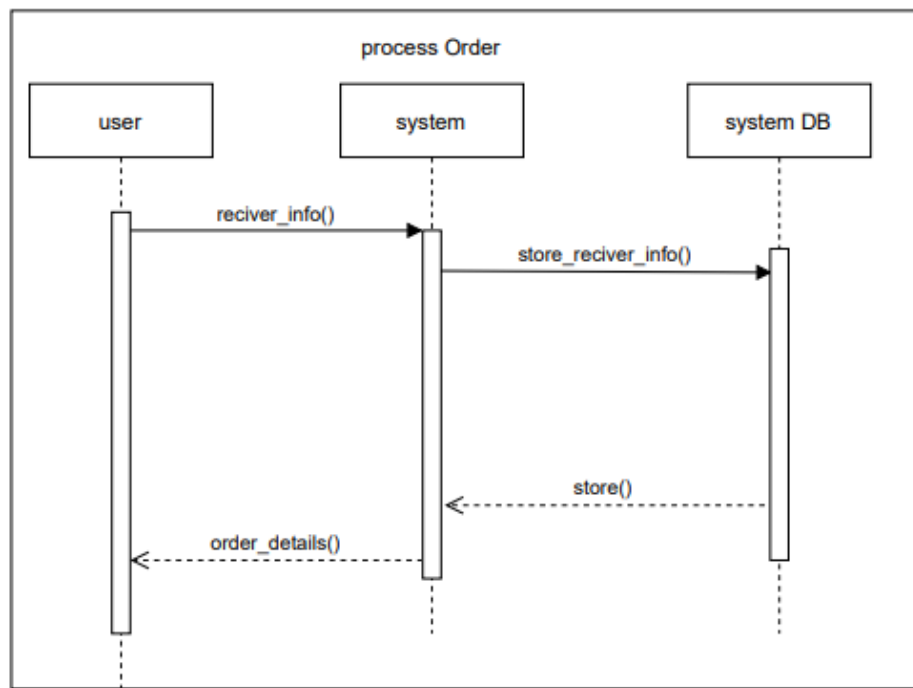
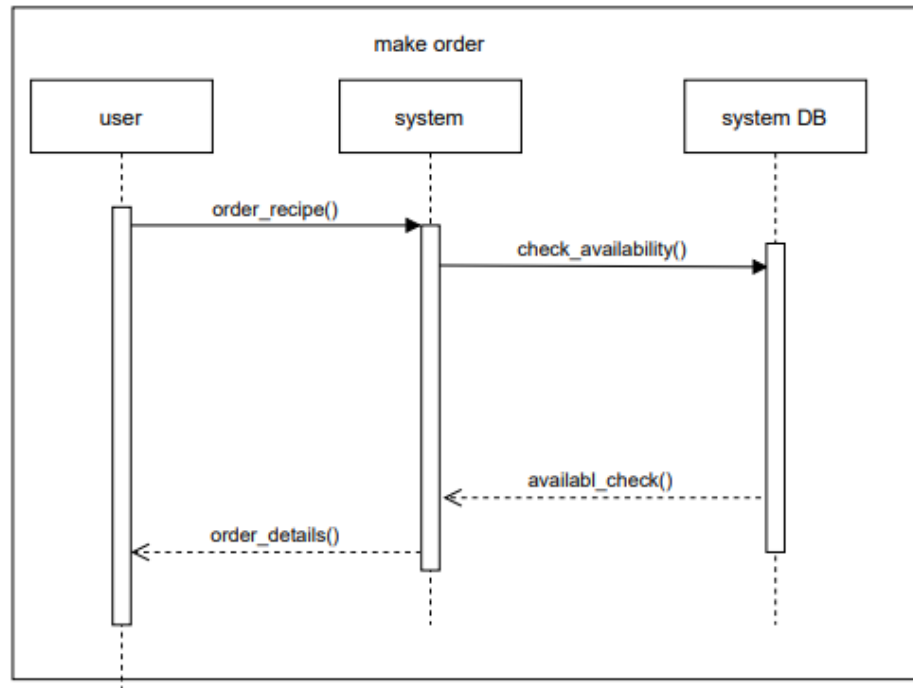
Although CRC cards were originally introduced as a technique for teaching object-oriented concepts, they have also been successfully used as a full-fledged modeling technique. My experience is that CRC models are an incredibly effective tool for conceptual modeling as well as for detailed design. CRC cards feature prominently in extreme Programming (XP) (Beck 2000) as a design technique. My focus here is on applying CRC cards for conceptual modeling with your stakeholders.

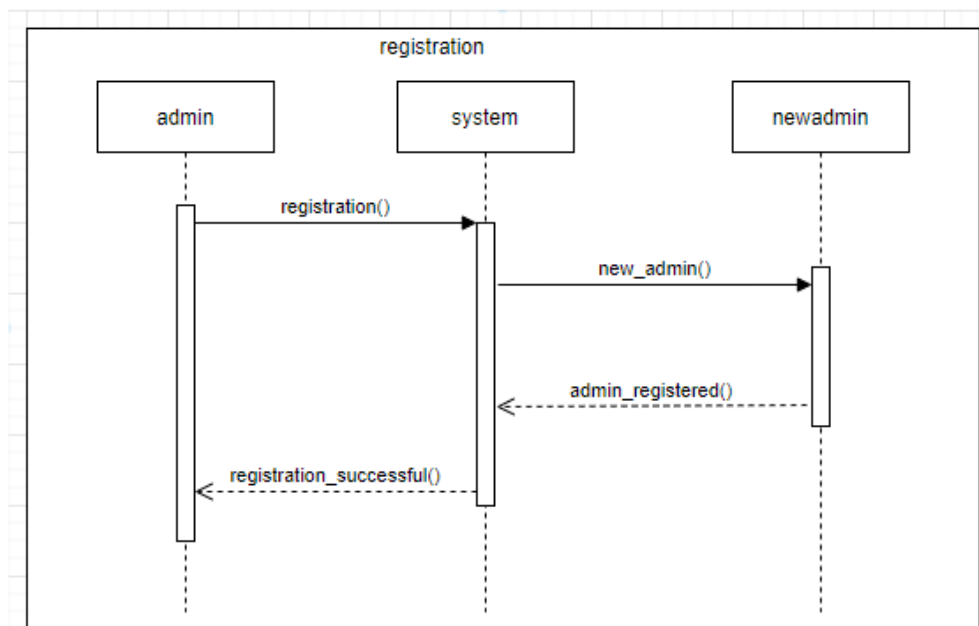
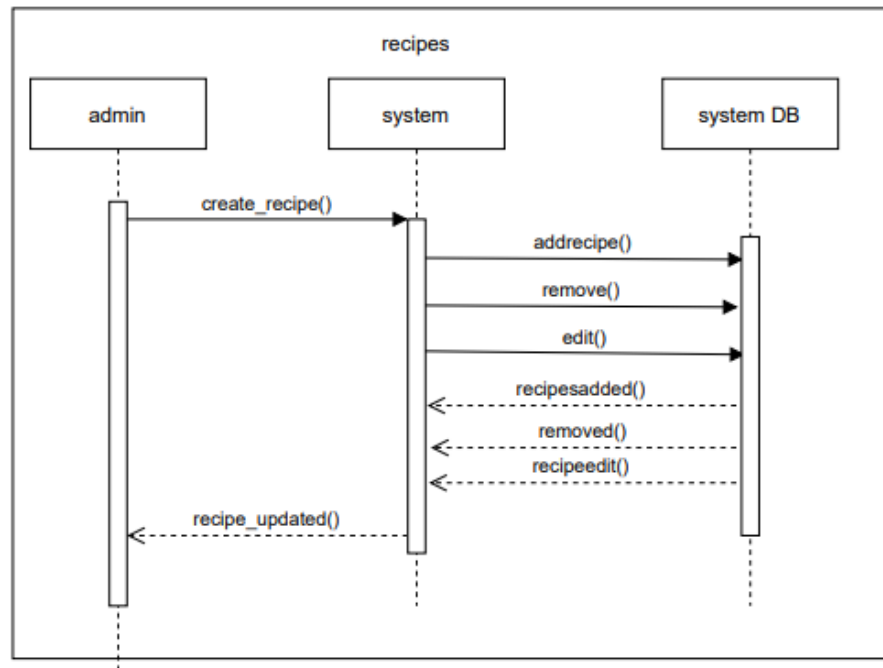
A class represents a collection of similar objects. An object is a person, place, thing, event, or concept that is relevant to the system at hand. For example, in a university system, classes would represent students, tenured professors, and seminars. The name of the class appears across the top of a CRC card and is typically a singular noun or singular noun phrases, such as Student, Professor, and Seminar. You use singular names because each class represents a generalized version of a singular object.

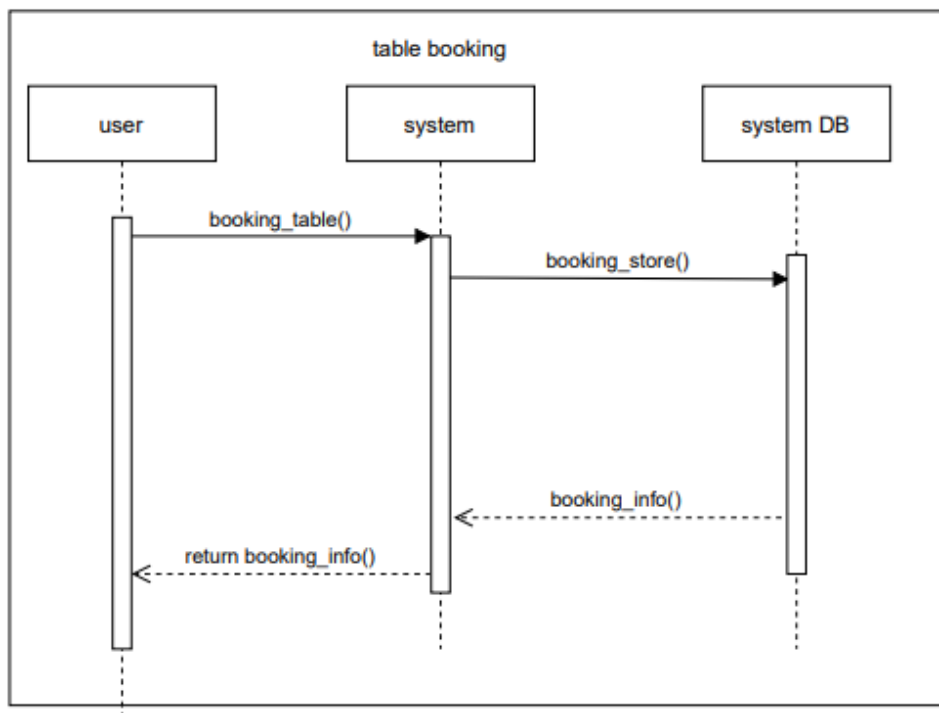
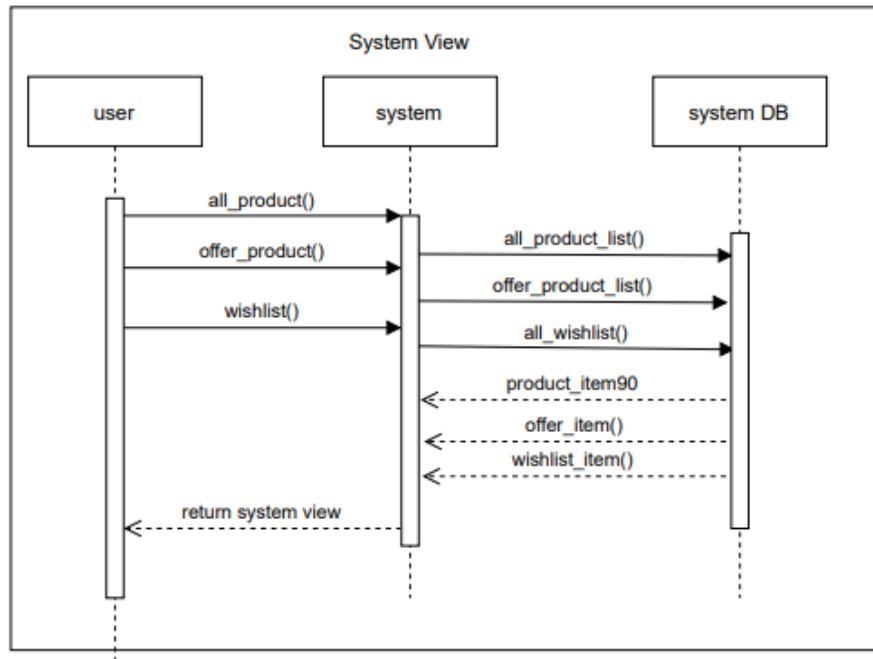
4.2 Sequence Diagram (for each use case)

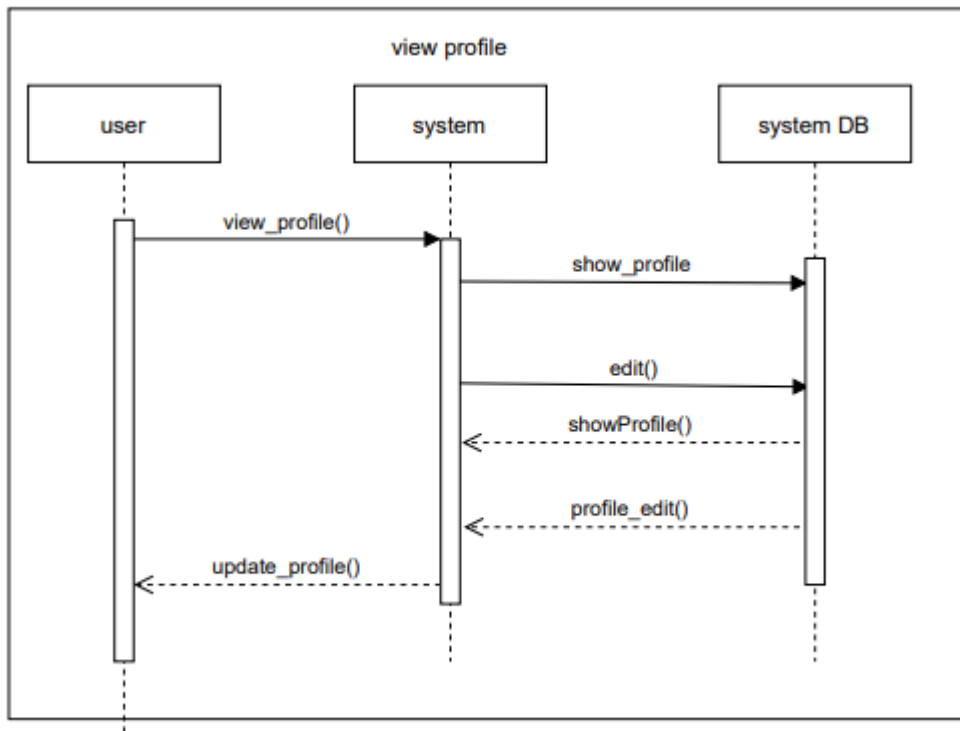




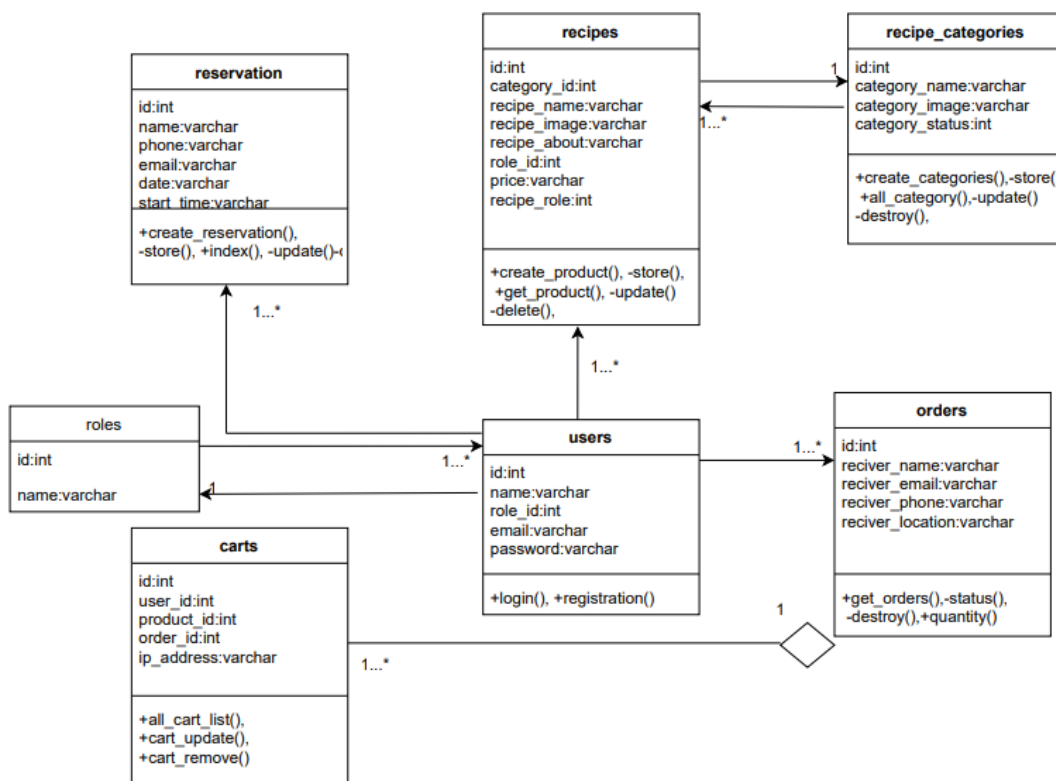




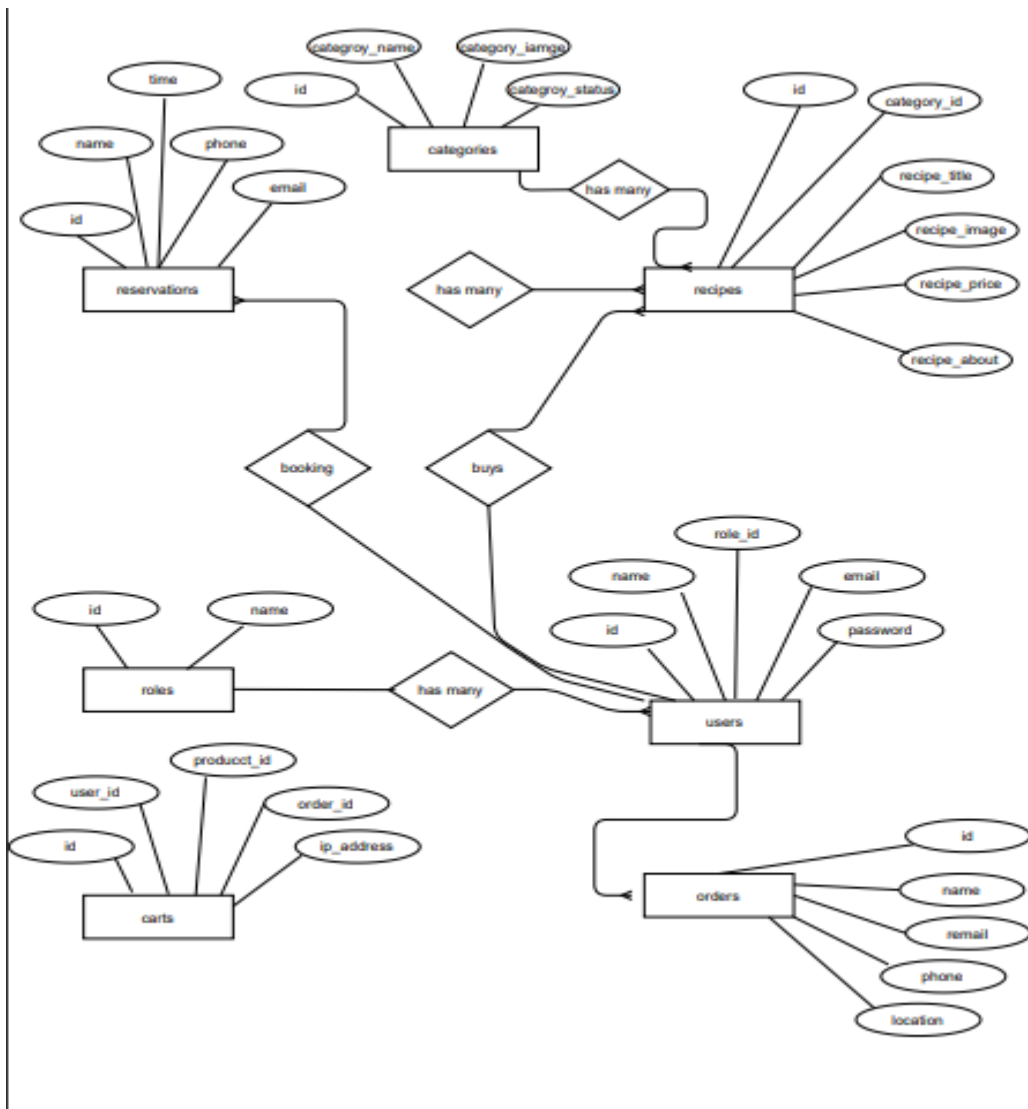




4.3 Class Diagram



4.4 Entity-Relationship Diagram



4.5 Development Tools & Technology:

The web development tools that help the developer to ease the workflow of development are:

1. JavaScript Libraries
2. Front-end Frameworks
3. Databases
4. Programming Languages
5. Icons
6. JavaScript Libraries

7. PHP Framework

JavaScript happens to be one of the most popular programming languages for the developer community. The library contains a set of already prepared libraries – that helps in easy access to web development. Some of the famous Javascript libraries are:

1. Front-end Frameworks
2. Php Frameworks

The front-end frameworks constitute folders and files. JavaScript, CSS, and HTML and Bootstrap are some of the examples to be cited.

Programming Languages

The languages form the backbone of web development tools. The popular ones used are PHP framework(Laravel) ,JavaScript and many more.

4.5.1 User Interface Technology:

User experience marks the top priority for almost every business in the market today. As a beginner in the developer world, you would have come across frequently that even the brief that you were given for your project, mentioned the simplicity of its user interface that it should feature. It doesn't matter how complex are the functions and background operations happening alongside. What the users see and get to experience has to be a seamless and smooth execution.

Some of the most popular and successful companies like Netflix, Facebook, and Instagram operate on the principle of prioritizing their UI over everything. Take any one of them, and you would notice that the platform is powerful yet simple with a brilliant user interface. And that is because of the many versatile frameworks working behind the curtains.

However, there is no denying that the ever-increasing demand of the customers and the market, in general, requires a constant need to put out high-quality functionality and usability. As this responsibility piles up on the developer, many developers find themselves stuck and confused about which framework to pick among so many.

4.5.1.1 ASP.NET MVC4 Framework or PHP Framework

Laravel is a web application framework with expressive, elegant syntax. We

believe development must be an enjoyable, creative experience to be truly fulfilling. Laravel attempts to take the pain out of development by easing common tasks used in the majority of web projects, such as authentication, routing, sessions, and caching.

Laravel aims to make the development process a pleasing one for the developer without sacrificing application functionality. Happy developers make the best code. To this end, we've attempted to combine the very best of what we have seen in other web frameworks, including frameworks implemented in other languages, such as Ruby on Rails, ASP.NET MVC, and Sinatra.

4.5.1.3 CSS Framework or Bootstrap:

Cascade Style Sheet version 3.0 is used in this project. Because of this framework and the bootstrap style the site is responsive. Anyone can use any smart tools to access the website. Bootstrap has consistently been one of the biggest CSS Frameworks there is. It is often a de facto framework, especially for Web Developers. After several years of work, in early 2018, the official 4.0 version was released! Bootstrap 4.0 was a significant update.

4.5.1.4 Font Awesome or Others:

Font Awesome is used in this project. To make a beautiful website there us various front styles needed font awesome is a database of the font where we all collect the font and linked in our project work. Get vector icons and social logos on your website with Font Awesome, the web's most popular icon set, and toolkit.

Font Awesome is a font and icon toolkit based on CSS and LESS. It was made by Dave Gandy for use with Twitter Bootstrap and later was incorporated into the BootstrapCDN. Font Awesome has a 20% market share among those websites which use third-party Font Scripts on their platform, ranking it second place after Google Fonts.

4.5.2 Implementation Tools & Platforms

To implement this project various things needed to be done

1. Research Analysis
2. Notepad++/ Bracket
3. PHP framework(Laravel)
4. Microsoft Visual Studio 2010
5. Database

And so many things that are required to developed a professional eCommerce site.

4.5.2.1 Microsoft Visual Studio 2010 or PHP Storm

Both of the tools are user-friendly however I am confident and used to VMS 2010. It was very easy to figure out the problem and solve it. Also, it helps us to work on the project as a live collaboration.

4.5.2.2 MySQL Community Server 5.5

For the database, we have used MySQL Community Server 5.5 as we have taught in our university. MySQL is very user-friendly and very easy to use. All the database query is run on MySQL server.

4.5.2.3 .NET Runtime or Apache HTTP Server

Apache HTTP Server for the database connection otherwise no performance can be done from the admin to user interface. In the design phase, the login form for user admin is designed to log in and access the database connection is needed. Apache HTTP Server is used for the database connection.

CHAPTER 5: SYSTEM TESTING

5.1 Testing Features

5.1.1 Features to be tested

- 1) At first I need to test Registration & Login Features.
- 2) Then I need to test the part of the order.
- 3) Then I can test the feature of Approve of the order part.
- 4) I can also test the View order.
- 5) Seller features can be tested. (Like Seller information, Seller Education, Seller Specialty).

5.1.2 Features not to be tested

- 1) I need not test the feature of the Home Page.
- 2) I need not test about page, contact page features.

5.2 Testing Strategies:

A testing strategy is an overview that describes the testing approach of the code development cycle. it's created to tell project managers, testers, developers regarding some key problems with the testing method. check methods describe however the merchandise risks of the stakeholder's square measure satisfied at the test level, that varieties of testing square measure to be performed, and that entry and exit criteria apply. they're created supported development style documents. System style documents square measure primarily used, and infrequently abstract style documents could also be stated.

5.2.1 Test Approach:

Test approach is the test strategy implementation of a project, which defines how testing would be carried out. Test approach has two techniques: Proactive - An approach in which the test design process is initiated as early as possible to find and fix the defects before the build is created. This approach is applied in our project.

5.2.2 Pass/Fail Criteria:

The exact pass/fail criteria for load and stress testing are to be determined by ITS. However, in a general sense, if the application performs to an acceptable degree after it has been put under a certain capacity, then the test item will pass. On the contrary, the test item will fail if the application underperforms. Our system is tested with all the test cases and it passed the requirement.

5.2.3 Suspension and Resumption:

It is important to understand that if a defect is detected at a point after which the testing shall resume proves to be of no use, then applying the resources on testing will be futile. One needs to specify the reason for stopping the test activities and define the acceptable level of defects that allows the testing process to surpass those defects. Few defects may arise in resuming the test activity. Those are the hidden defects that were somewhere ignored earlier.:

5.2.4 Testing Schedule

A Test Plan is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product. Test Plan helps us determine the effort needed to validate the quality of the application under test. The test plan serves us as a blueprint to conduct software testing activities as a defined process, which is minutely monitored and controlled by the test manager.

5.2.5 Traceability Matrix

The main agenda of every tester should be to understand the client's requirements and make sure that the output product should be defect-free. To achieve this goal, every QA should understand the requirement thoroughly and create positive and negative test cases.

This would mean that the software requirements provided by the client have to be further split into different scenarios and further to test cases. Each of these cases has to be executed individually. A simple way is to trace the requirement with its corresponding test scenarios and test cases. This merely is termed as 'Requirement Traceability Matrix.'

The traceability matrix is typically a worksheet that contains the requirements with its all-possible test scenarios and cases and their current state, i.e. if they have been passed or failed. This would help the testing team to understand the level of testing activities done for the specific product.

1. Requirement Traceability Matrix
2. Requirement ID
3. Requirement Type and Description

Test Cases with Status

5.3 Testing Environment (hardware/software requirements)

Network

- Documentation required like reference documents/configuration guides/installation guides/ user manuals

Test Case ID:	Test Name:	Test Case Objective:	Pre-Requisite:	Steps to Perform:	Expected Result:	Output:	Status : Pass/Fail.
001	Sign up with empty fields	To verify the none of the fields remain empty	Displaying sign up form	1. Submit a form with several empty fields. 2. Click the signup button	Error message showing indicating empty fields	Error	pass
002	Sign up with completing fields	Completing the fields with data	Displaying sign up form	1. Submit the form by completing all empty fields. 2. Click the signup button	Sign up successful	Input data in the wrong field	fail
003	Sign up with corrected data	To input the correct data incorrect field	Displaying sign up form	1. Submit the form by completing all correct data incorrect fields. 2. Click the signup button	Sign up successful	Sign up successful	pass
004	Login with empty fields	To verify the none of the fields remain empty	Displaying login form	1. Submit the form with several empty fields. 2. Click the login button	Error message showing indicating empty fields	Error	pass
005	login with completing fields	Completing the fields with data	Displaying login form	1. Submit the form by completing all empty fields. 2. Click the login button	Login successful	The user is not registered	fail
006	login with corrected data	To input the correct data incorrect field	Displaying login form	1. Submit the form by completing all correct data incorrect fields. 2. Click the sign-up button	Login successful	Login successful	pass

007	Order food by user	Checking order criteria	Displaying food	1.user select food 2.guest added it to the cart	You need to login	You need to login	pass
-----	--------------------	-------------------------	-----------------	--	-------------------	-------------------	------

Tests are limited to what can be tested and what not should be tested.

Following people are involved in test environment setup

- System Admins,
- Developers
- Testers
- Sometimes users or techies with an affinity for testing

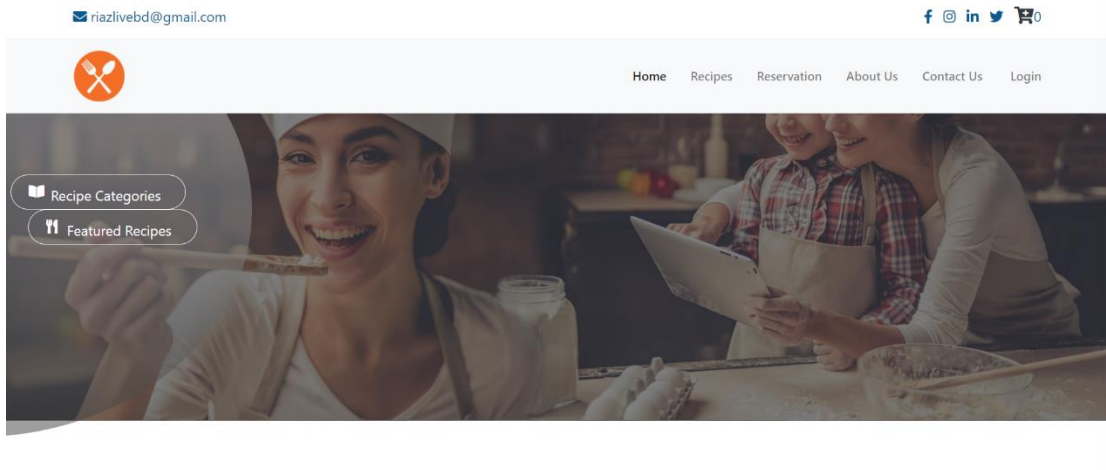
5.4 Test Cases

Test Case ID:	Test case name	Test Case Objective:	Pre-Requisite:	Steps to Perform:	Expected Result:	Output:	Status: Pass/Fail.
008	Order food by user	Checking order criteria	Displaying foods	1.user select food 2.user added it to the cart	Food added to the cart	Food added to the cart	pass
009	Check Out	Check out method	Displaying payment	1.user can use a payment method to check out the food	Check out successful	Check out successful	pass
010	Food status by admin	Add and remove a food	Displaying foods	1. admin add a new food in the system 2. admin remove a food in the system	Foods updated	Foods updated	pass
011	Food status by admin	Admin approve a food from the user	Displaying foods	1. user upload a food 2. admin approve the foods	Foods updated	Foods updated	pass


012	Order approved by admin	Admin approve order	sell order	Display order	1. user orders a food. 2. admin approve the order	Order approved	Order approved	pass
-----	-------------------------	---------------------	------------	---------------	--	----------------	----------------	------

CHAPTER 6: USER MANUAL


6.1 User Manual Home Page)



6.2 User Manual (user)



[Home](#)
[Recipes](#)
[Reservation](#)
[About Us](#)
[Contact Us](#)
[user](#)



user
user@gmail.com

[Dashboard](#)

[Logout](#)

user Profile


First Name: user

Email: user@gmail.com

[About Us](#)

[Need Help](#)

[Follow Us](#)



[Home](#)
[Recipes](#)
[Reservation](#)
[About Us](#)
[Contact Us](#)
[Login](#)


Login

E-Mail Address

Password


Remember Me


[Create an Account](#)
[Sign up](#)





[Home](#)
[Recipes](#)
[Reservation](#)
[About Us](#)
[Contact Us](#)
[Login](#)


Recipe Categories



USA



VAGITABLE



CANADIAN



ASIAN



COOL DRINKS



FOOD


DESERT


JUICE


DRINKS


EGG


FISH



Egg Omlate

Egg

Price: 200 Balance

Add to Cart

View

Hotel Reservation



Enter Your Name

Enter Your Phone

Enter Your Email

Enter Your Date

Enter Your Start Time



Recipes



Cocola

♥0

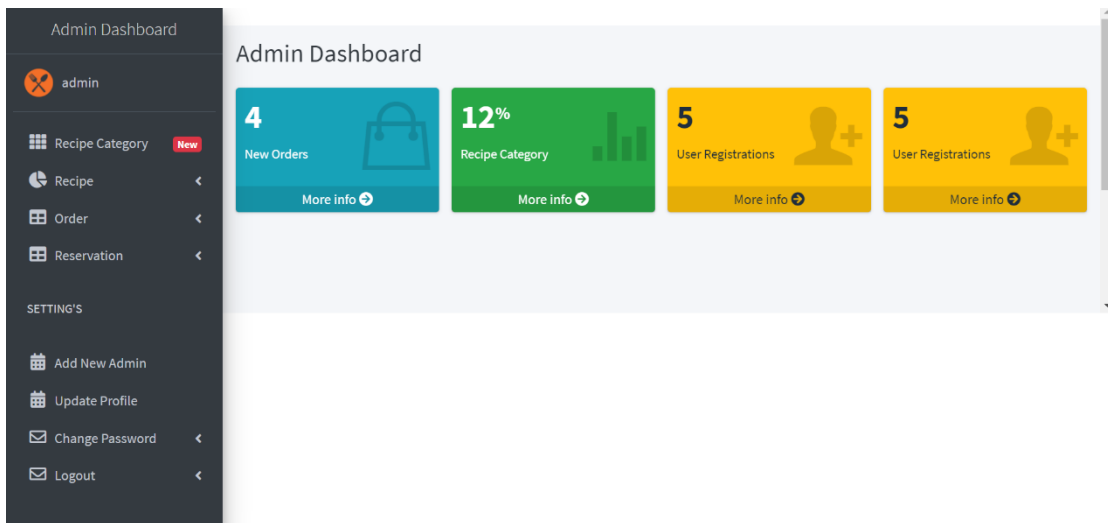


Egg Omlate

♥1



6.3 Admin Panel



CHAPTER 7: PROJECT SUMMARY

7.1 GitHub Link

7.2 Critical Evolution

There was some critical situation for the evolution of this project main was the database connection and then the responsiveness of all the pages and lastly the checkout process.

7.3 Limitations

This project is Internet-based, you won't be able to access it if your Internet connection goes down. With software-based restaurant management system, there are recurring costs. Also upgrade system can be costly.

7.4 Future Scope

In future I will add more category of food, where user can get at lower cost. Most restaurants can calculate their return on investment in terms of cost savings, which is mostly due to enhanced personnel productivity and resource usage. Revenue growth can also be quantified in terms of restaurant management system, which is generally a consequence of greater guest satisfaction and, in certain situations, faster table turn times—and, in some situations, the ability to reach and serve a large number of people.

