ADMAS UNIVERSITY

Mekanissa Campus

Department of Computer Science

"RESTAURANT RESERVATION SYSTEM"

A PROJECT SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE FOR PARTIAL FULFILLMENT FOR THE REQUIREMENTS OF A BACHELOR'S DEGREE IN COMPUTER SCIENCE.

Group Member

Name Id №

1. Aynalem Lemi.....0164/21

2. Hikma Murad. 0579/21

3. Kisanet Gidey.....0269/21

4. Loza Abdi.....0290/21

5. Rahel Alema.....0533/21

6. Reem Dawd.....0167/21

Advisor: Welde Janfa

Addis Ababa, Ethiopia

May 2024

CERTIFICATION OF RESEARCH APPROVAL

Mekanisa Campus

Admas University

This is to certify that the research prepared by the above listed group members, entitled "ONLINE RESTAURANT RESERVATION FOR AG RESTAURANT " and submitted in partial fulfillment requirement for the Bachelor Degree of Computer Science (BA) complies with the regulation of the University and meets the accepted standards with respect to originality and quality.

Group Members name ..... Id №.....Signature .....Date

1. Aynalem Lemi.....0164/21 _____ _____

2.Hikma Murad. ..... 0579/21 _____ _____

3.Kisanet Gidey.....0269/21 _____ _____

4.Loza Abdi.....0290/21 _____ _____

5.Rahel Alema.....0533/21 _____ _____

6.Reem Dawd.....0167/21 _____ _____

Name of Advisor: ..... Signature.....Date……...

Name of Dep't Head: ..... Signature.....Date…….

Signature of Board of Examiner:

External examiner: ..... Signature.....Date…….

Internal examiner: ..... Signature.....Date…….

Dean: ..... Signature.....Date……...

Declaration

We declare that this project is our original work and has not been presented for a degree in any other university

Group Members name.....Id no.....Signature..... date

1. Aynalem Lemi 0164/21
2. Hikma Murad. 0579/21
3. Kisanet Gidey. 0269/21
4. Loza Abdi. 0290/21
5. Rahel Alema 0533/21
6. Reem Dawd 0167/21

This project documentation has been submitted for examination with my approval as university advisor:

Advisor Name: Welde Janfa

Acknowledgement

First, we would like to thank God for letting us through all the difficulties. We have experienced your guidance day by day. We'd like to thank our adviser, Welde Janfa, for making this job possible. His counsel and direction saw us through all stages of the project's growth and for supporting us with project coordination, particularly in authoring this inspiration. We would also like to thank AG Restaurant for allowing us to do this project and also would like to give much appreciation to all the staff members. We are really grateful to Admas University for their guidance and frequent monitoring, as well as for providing crucial project data and assisting us in completing the project. Finally, we'd want to express our gratitude and appreciation to those who have generously shared their talents with us.

Table of Contents

List of table

List of figure

Abbreviations

P.L.C - Public limited company

PHP - php hypertext preprocessor

HTML- Hypertext markup language

CSS - Cascading stylesheet

SMS - Short message service

UML - unified modeling language

MySQL- structured query language

OOP- Object-oriented programming

Abstract

The "Restaurant Reservation System" project is exclusively designed for the AG restaurant. The project focuses on table reservations and food ordering processes. Through an exploration of the current reservation system at AG Restaurant, the aim is to digitize and automate the manual processes in place. Utilizing object-oriented techniques, the project seeks to create a fast and efficient system that is accessible to both users and customers. The web-based application will provide a functional platform for AG restaurant users, with the ultimate goal of minimizing time and effort. The team is committed to developing a system that aligns with the intended design, ensuring functionality and usability.

CHAPTER ONE: Introduction

A restaurant is a business that prepares and serves food and drinks to customers. Meals are generally served and eaten on the premises, but many restaurants also offer takeout and food delivery services. Restaurants vary greatly in appearance and offerings, including a wide variety of cuisines and service models ranging from inexpensive fast-food restaurants and cafeterias to mid-priced family restaurants to high-priced luxury establishments.[7]

This chapter describes the background of the organization, background of the project, team composition, statement of the problem, objectives of the project, feasibility analysis, scope and significance of the project, target beneficiaries of the system, methodology of the project, and limitations of the project.

1.1 Background information of the organization

According to the proprietors, AG Cafe and Restaurant P.L.C. was established in 1972 E.C. and is situated on Lebu Square. The establishment comprises a distinct cafe as well as a restaurant, with the latter opening from 12:30 p.m. until 2:30 a.m. Three managers oversee the operations of the restaurant, including the owners themselves. Additionally, there are thirty-two employees who serve various roles, such as cashiers, waiters and waitresses, cooks and chefs, and janitors.

The restaurant boasts twenty-three tables that can seat up to four individuals each; however, certain family chairs have been designed to accommodate larger groups ranging from eight to fifteen patrons. Customers may also avail indoor services for special events like birthdays or parties upon request. During peak hours, at most one hundred fifty customers can be served simultaneously, though it should be noted that lunchtime tends to draw large crowds, leading to congestion within the premises.

In the event that all tables are occupied by diners during peak times, patrons may use the cafe area before their designated reservation time slot commences. The establishment operates a manual reservation system whereby customers either call ahead or inform staff directly regarding their desired table arrangement(s). Quality service is provided at reasonable rates depending on booking status (i.e., new versus returning customer). Returning clients receive free reservations, while newcomers incur a fee for this service.

Reservations may be adjusted or cancelled according to customer preferences or term set forth by management regarding restrictions such as ordering food items exclusively from regular tables rather than family seating arrangements, etc.

The establishment conscientiously informs its patrons of the promptness of their reservations, both to maintain the quality and freshness of cuisine and to enhance customer satisfaction. In pursuit of optimizing the customer experience by preserving time and convenience, the restaurant requires an automated system that utilizes online technology to manage reservation processes.

1.2 Background of the Project

Project titled "Restaurant Reservation System," which aims to simplify the process of making reservations at AG restaurant. It is an online platform that allows customers to reserve tables at their desired restaurant and order their preferred dishes in advance. This system saves customers time and effort by eliminating the need to physically visit the restaurant to make a reservation. With the Restaurant Reservation System, customers can easily browse through available tables, select their preferred date and time, and choose their desired dishes from the restaurant's menu. This system benefits both customers and restaurants, as it enhances the dining experience for customers and streamlines the reservation process for restaurants. with restaurant online reservations to help customers save time and eat their desired dishes.

1.3 Team Compositions

This project is prepared by six individuals each with their own role and responsibilities. The Specific responsibilities of each person are listed in the table below.

Project Title: Restaurant Reservation System					
Prepared By	No.	Name	ID No.	Email/Phone Numbers	Responsibilities
	1.	Aynalem Lemi	ADMK/0164/21	0982406949	Technical writer and tester
	2.	Hikma Murad	ADMK/0579/21	0944732562	Web developer and project manager
	3.	Kisanet Gidey	ADMK/0269/21	0936639047	Designer, web developer and tester
	4.	Loza Abdi	ADMK/0290/21	0970416628	Web developer and technical writer
	5.	Rahel Alema	ADMK/0533/21	0953930947	Technical writer and tester
	6.	Reem Dawd	ADMK/0167/21	0940428331	Project architect and web developer
Date	Dec,2023				
Advisor	Wolde J.				

Table 1 ***Team Compositions***

1.4 Statement of the Problem

AG Restaurant currently utilizes a reservation system that is administered manually through traditional phone calls. However, this age-old process has unfortunately caused a plethora of predicaments for the establishment. Sadly, these complications have resulted in both loyal and prospective patrons being deterred by the time-consuming nature of reserving a table and uncertain as to availability due to inadequate tracking mechanisms. This has ultimately led to a significant loss of business for AG Restaurant. One of the primary issues with the current reservation system is its reliance on manual management via phone call. This method is outdated and inefficient, often leading to errors in scheduling and double bookings.

Additionally, customers may find it frustrating to spend valuable time on hold or playing phone tag with restaurant staff just to secure a table. Another issue arising from the antiquated reservation process is the lack of proper tracking mechanisms. Without an effective way to monitor reservations and availability, AG Restaurant cannot accurately gauge their capacity or anticipate busy periods. This results in inadequate staffing levels, which can lead to longer wait times for customers or even turn away potential diners.

Moreover, the current system's shortcomings have led to a loss of both loyal and new patrons who are seeking more convenient alternatives when it comes to reserving tables. With many modern restaurants offering online booking systems or mobile apps that allow customers to reserve tables at their convenience, AG Restaurant risks falling behind its competitors without taking steps towards implementing similar technologies. In conclusion, while AG Restaurant's manual phone-based reservation system may have served them well in the past, it is now causing more harm than good. The establishment must take proactive measures to address these issues if they wish to remain competitive in an increasingly tech-savvy market. Implementing an online booking system or mobile app would not only improve customer satisfaction but also streamline operations for staff, leading to increased efficiency and revenue generation.

The Following statements are the primary issues:

Insufficient Seating: A surge of patrons may inundate the establishment, and inadequate seating could impel them to vacate the premises prematurely.

Inaccurate Communication: Inadequate knowledge regarding dietary limitations or allergies among servers may result in erroneous information being conveyed to customers.

Waiting for an extended duration: Patrons may experience prolonged wait times as the servers are currently occupied with numerous tasks.

1.5 Objective of the Project

General Objectives

The general objective of the project is to develop an online restaurant reservation system for AG restaurant.

specific objectives

The specific objectives are:

To collect data and understand the existing system.
To identify and analyze user and system requirements.
To propose a new system.
To develop new system with efficient tools and techniques.
To test and evaluate the system and see if it meets the objectives.

1.6 Feasibility analysis

Feasibility analysis is a process of assessing the practicality and potential success of a proposed project venture. Our feasibility study involves evaluating various factors of our project such as technical, economical, legal, operational and scheduling aspects to determine whether the project is viable and worth pursuing. It is important for research and development of software. Feasibility study is the main phase for our project or any research by answering what will be done, how will be done and what things we faced in our work or if it is the right way etc, are answered in this stage. [6]

1.6.1 Operational feasibility

Evaluating the workforce needs, production procedures, supply chain management, and effect of our project on daily operations is known as operational feasibility. The project team selects a website platform that facilitates easy user interaction. This enables the system to function.[6]

1.6.2 Technical feasibility

Assessing technical feasibility involves determining if the necessary technology and resources are available for a project. Our team has the resources to develop a web application using PHP, an open-source programming language that allows for dynamic web pages and interaction with databases. The team chose PHP because it is versatile and powerful. The proposed platform is technically feasible. [6]

1.6.3 Economic feasibility

Economic or financial analysis is estimating the initial investment, cash flow and costs on our project. As our project built by open-source tool it minimizes the team cost. And the team is educated on the how the system built, this helps the team investment to save money. Therefore the system is economic feasible.[6]

The given table shows the item we need to work for our project respectively to their cost.

No	Item	Unit	Price	Total cost
1.	Computer	3	free	Free
2.	Transport	-	1000	1000
3.	Paper	1 Pack	500	500
4.	Print	-	600	1200
5.	Internet	-	700	700
6.	Flash disk	2	400	800
7.	Team member fee	-	-	-
8.	Binding	-	-	400
Total cost			**4600**	

Table 2 ***Economic feasibility***

1.6.4 Legal or political feasibility

The legal or political feasibility is examining the legal and regulatory requirements that guide or may impact our project, including permits, licenses and compliance with laws and regulations. So the team gets the license from the organization to work on the given restaurant. There is no any restaurant reservation system for the given restaurant. So, the system does not contradict with any rules established by either the organization or government. Moreover, the system is political feasible. [6]

1.6.5 Schedule feasibility

For any research or project, timeline is basic thing for assuring all about the work flow. As the team believes the importance of scheduling for the starting and ending the project will have a deadline. Based on the given timeline we group the overall work flow. The given table show schedule of the project. [6]

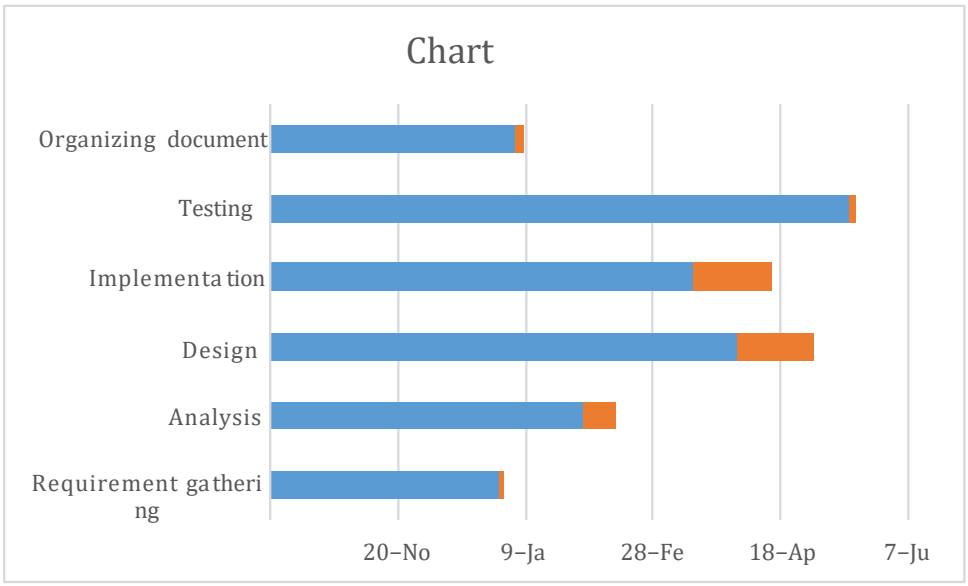


Figure 1 ***Schedule feasibility***

1.7 Scope and significance of the project

1.7.1 Scope

This system offers the capacity to reserve a table for dinner online and access details regarding dining availability, small gatherings, and celebrations. The restaurant proprietor can manage all aspects of their services, including food preparation, seamlessly through this online reservation platform. Moreover, the proposed system operates efficiently and with ease. This project has the following scope:

- Customer Registration
- Registration of tables and their details. Registration of foods and their details.
- The actual reservation process and the hole is managed.
- Any required information and summarization report will be provided.

1.7.2 Significance

A reservation system helps businesses optimize restaurant seating layouts so that they can accommodate more customers and give guests more accurate wait times. Launching a restaurant reservation system for the customer is ideal. Not only can it help boost the restaurant's profit margin, but it can also enhance the customer experience. And to comport people and build attractiveness, many restaurants now offer some special features to customers, like online restaurant reservation software. By having this software, customers can easily reserve their tables and also order online.

1.8 Target beneficiaries of the system

The Project beneficiaries are also known as the target group or target beneficiaries and they are: -

Customers

Customers are the main gainers from the reservation system since it makes it simple for them to make reservations online or via a mobile app without having to call or wait in large lines. In addition, they can choose their desired time and day, see the availability table, and get a reservation confirmation.

Restaurant staff

The reservation system benefits restaurant staff by automating the reservation process, reducing the workload of manually managing reservations, and minimizing errors. The system can also provide staff with real-time information on table availability, customer preferences, and order history, enabling them to provide better service to customers.

Restaurant owners

Restaurant owners benefit from the reservation system by improving their business operations, increasing revenue, and enhancing customer satisfaction. The system provides owners with valuable data on customer behavior, reservation patterns, and menu preferences, which can be used to make informed decisions about menu offerings, staffing levels, and marketing strategies.

Third-party platforms

Third-party platforms such as online travel agencies (OTAs) and food delivery services can integrate with the reservation system to offer customers a seamless booking experience and increase their revenue streams. The integration enables customers to book a table while making a hotel reservation or order food for delivery directly from the restaurant's website or app.

1.9 Methodology for the project

Given that object-oriented programming (OOP) is a prevalent programming paradigm in software development, we will employ it to construct the analysis aspect of our project. OOP simplifies the management of numerous system components and facilitates modular and reusable code creation for restaurant reservation systems. Additionally, OOP enables method and data encapsulation within classes, bolstering data security and guarding against unauthorized access or alteration. Object-oriented design offers several benefits, including:

Reusability

Object-oriented design promotes code reuse by allowing developers to create classes that can be used in other classes.

Modularity

Object-oriented design promotes modularity by breaking down complex systems into smaller, manageable objects.

Encapsulation

Object-oriented design promotes encapsulation by hiding the detail information about the object.

Flexibility

Object-oriented design promotes flexibility by allowing developers to modify and extend existing objects without affecting other part of the system.

Maintainability

Object-oriented design promotes maintainability by making it easier to understand and modify code.

1.9.1 Data Sources

Data comprises discrete or continuous values that communicate information, representing quantity, quality, fact, statistics and other fundamental units of meaning. It may also encompass sequences of symbols that can be formally interpreted. [1]

The group members collected detail information about the restaurant reservation system to know more about our proposed project. Without data, it would be difficult to make decisions. So data is collected from various source.

Primary Data

Primary data is the data that is collected for the first time through personal experiences or evidence, particularly for research. It is also described as raw data or first-hand information.[1]

Primary Data is obtained from Manager, Workers and Customers directly by interviewing them and collecting questionnaires.

Secondary Data

Secondary data are basically second-hand pieces of information. To put it in other words, the secondary data are those that are already collected. [1]

Secondary Data is obtained from different documents by reading other research documents.

1.9.2 Data Collection Techniques

Data collection or data gathering is the process of gathering and measuring information on targeted variables in an established system, which then enables one to answer relevant questions and evaluate outcomes. The group members use suitable data collection techniques to develop system.

Interview

We conducted interviews with the managers, workers, and customers of AG Restaurant to gather further information regarding their reservation system. In preparation for these interviews, we formulated open-ended questions, which are included in the annex as questionnaires.

Questionnaires

In situations where some individuals may not feel comfortable or confident participating in interviews, we have taken the necessary steps to accommodate their preferences by providing them with questionnaires for the purpose of data collection. These questionnaires will be included as an annex to this project, allowing those who prefer this method of information gathering to still actively contribute to our research efforts.

Observation

Direct observation is a straightforward method of data collection that involves gathering information through means other than interviews. Through observation and careful consideration of various communication styles and preferences, we ensured that all individuals were able to participate in a way that feels most comfortable and effective for them.

1.9.3 System analysis and design

The selected approach is Object Oriented Writing. These Object-oriented uses object-oriented analysis approach. This analysis approach is used to design software systems. it involves identifying the objects in a system, their properties, and their relationships with other objects. Also, it helps us to divide the system into subsystem so that we can manage each and every project step in a better way.

Development tools

Our system will be entirely web based. So, we will use the following programming language.

Frontend Tools

HTML –client-side programming language

CSS –describes the presentation of the look and formatting of the document written in a markup language.

JAVASCRIPT- used to add functionality, validate forms.

Backend Tools

The choice of programming language depends on the developer's preference and the requirements of the project. Our team decided to use PHP as server-side programming language.

PHP for server-side programming language

MySQL database system for database storage

Apache server for server-side software

1.9.4 Testing procedures

Testing procedures are used to verify the correctness, functionality, and performance of the soft system. Our team uses the following testing types like unit test, system test, integration test and acceptance test. the goal of the testing procedure is to ensure that the software system meets the quality standards and requirements specified by the stakeholders and end-users.

Unit test: this test helps to know how different units or components of a software application are

tested in isolation from rest of the application. We will use unit test in each stage of our project.

Integration testing: this involves testing how different units or components of a software work together. We will test our project to know if the components are working together correctly.

System testing: this involves testing the entire software system as a whole to ensure it meets the requirements and specifications. We will test the whole project using system testing.

Acceptance testing: this involves testing the software system with real-world scenarios to ensure it meets the needs of the end-users. The client representative will test our project to know if our project has fulfilled the customer needs.

1.9.5 Implementation

Implementation is the execution of plan, design, idea, model or standard for doing something. System implementation is installation and delivery of the entire system. During implementation the model will be translated into source code. Our team needs to implement the new system in some specific customers or end-users of the organization.

1.10 Limitation of the project

1. Technical limitations: The document may not consider bandwidth availability or network availability.

2. Resource limitations: The document may not consider resource limitations such as staff availability, budget constraints, and equipment availability

3. Environmental limitations: The document may not consider environmental limitations such as noise restrictions, waste management regulations, and energy efficiency requirements.

CHAPTER TWO: Description of the existing system

This chapter describes the existing system, players in the existing system, the general work flow of the AG Restaurant Reservation System, major functions in the existing system, Reports generated, Forms and documents, business rules, problems, practices to be preserved, a proposed solution for the new system, and requirements of the proposed system. Studying the existing system brings about an important contribution to the entire development process.

2.1 Introduction to Existing System

AG Restaurant employs a manual reservation system wherein the customer contacts through phone call to secure a table. The receptionist gives comprehensive information regarding the reservation process, including seat availability, food options, and a time schedule. Upon mutual agreement between the customer and receptionist, AG Restaurant finalizes the booking for the customer.

2.2 Players in the Existing System

Administrator: They are the ones who control the organizations mainly.
Restaurant Employees: They are the primary actors in making reservations. They primarily manually register it using a paper-based system.

Customers: They are those who use the services.

2.3 Major functions/activities in the existing system

Input:

Customer will be registered with basic information (name, contact details)
Customer will register the date and time.
Customer will register the number of guests.
Customer will register table preferences (e.g. booth, window seat, etc.)
Customer will register about Special requests (e.g. dietary restrictions, birthday celebration, etc.)

Processes:

Customer makes a reservation through the online booking platform or by calling the restaurant. The system checks table availability and confirms the reservation based on the input details.

Staff assigns tables to reservations and updates the table status in real-time.
Confirmation emails or messages are sent to customers with booking details.
Waitlist management is handled by the system to notify customers of available tables.
Customers can modify or cancel their reservations through the system, updating availability.
Customer feedback is collected post-dining experience to gather insights and improve service quality.

Staff scheduling is optimized based on reservation bookings to ensure adequate staffing levels

during peak hours.

Special promotions or offers are communicated to customers through the system for marketing purposes.

Outputs:
Reservation confirmation emails or messages to customers.
Reservation book or system updated with new reservation details Reminders sent to customers prior to their reservation
Reports on reservation trends and patterns for the restaurant's management team.

2.4 Business rule

1. Reservations need to be made at least one day in advance, and the restaurant must confirm them.
2. Guests must arrive on time for their reservation, or the table may be given away to other guests.
3. The restaurant requires a deposit or pre-payment for large groups or special events.
4. Guests are limited to a specific time slot for their reservation, especially during peak hours.
5. The restaurant has a cancellation policy that stipulates that reservations must be canceled with a minimum of 24 hours' notice.
6. The restaurant has different pricing or menu options referrals for special events or holidays.
7. The restaurant offers rewards or incentives for repeat reservations or referrals.

2.5 Reports generated in the existing system

Manual Order Taking and Billing: still rely on manual order taking and billing systems. The waiter takes the customer's order on a notepad, which is then transferred to the kitchen staff. At the end of the meal, the bill is calculated manually based on the ordered items.

Traditional Cash Register: AG restaurant may use a traditional cash register to calculate and record transactions. The cashier manually inputs the prices of the ordered items, and the cash register provides a printed receipt for the customer.

Excel Spreadsheets: In some cases, AG restaurants may use Excel spreadsheets to track inventory, daily sales, and expenses. This system requires manual data entry and calculations.

Handwritten Receipts: After the bill is settled, the cashier typically provides a handwritten receipt to the customer, which includes the details of the items ordered, their prices, and the total amount paid.

2.6 Forms and documents used in the existing system

The existing system doesn't have any forms for AG restaurant.

2.7 Bottlenecks of the existing system

2.7.1 Response time

The existing system has a lot of performance issues. In the current system, people must make a phone call to make a reservation at a restaurant. This process takes a lot of time. Therefore, the response time is high.

2.7.2 Input and output

The process of entering data or information is known as an input.

Inaccurate- As the current system is paper-based, incorrect input may be recorded. When customers order what they want, the clerk may write something wrong.

Redundant- Data redundancy may occur during reservation. Since the current system is a manual system, a table can be reserved more than once.

Inflexible- Because of manual system it is difficult to change or replace input.
The information generated by the system is known as an output.
Inaccurate- In the current manual system, incorrect table reservations can be made by receiving incorrect information or not understanding the customer's needs.

2.7.3 Security and controls

As the current system is a manual one, we cannot protect the written data or information from physical threats. As a result, if the restaurant loses the data that people have given them when they make a reservation, they may lose their customers.

2.7.4 Efficiency

It cannot be said that the existing system is efficient because it is a manual system. Also, it is time-consuming.

2.8 Practice to Be Preserved

- Food safety and hygiene: Maintaining high standards of cleanliness and food safety is crucial to prevent foodborne illnesses and ensure customer satisfaction.
- Quality ingredients: Using fresh, high-quality ingredients is essential for creating delicious dishes that keep customers coming back.
- Customer service: Providing excellent customer service is key to building a loyal customer base and generating positive word-of-mouth.
- Menu diversity: Offering a diverse menu with options for different dietary preferences and restrictions can attract a wider range of customers.

- Sustainability: Implementing sustainable practices, such as reducing food waste, sourcing locally, and using eco-friendly packaging, can help restaurants reduce their environmental impact and appeal to socially conscious consumers.
- Staff training and development: Investing in staff training and development can improve the overall dining experience for customers and help retain talented employees.
- Innovation: Staying up-to-date with food trends and experimenting with new dishes and concepts can keep restaurants competitive in a constantly evolving industry.

2.9 Proposed solution for new system

The purpose of implementing a new system for restaurant reservations is to address the problems of the existing system and provide a more efficient and effective solution. The new system should aim to:

- Improve customer experience by providing clear and concise communication regarding availability, wait times, and reservation policies.
- Streamline the reservation process with accurate and up-to-date record-keeping, efficient table assignments, and flexible options for changes or cancellations.
- Enhance safety measures by adhering to health and safety guidelines, including proper cleaning and sanitization of tables and common areas, as well as enforcing social distancing measures.

  Provide training and support for staff members who manage the reservation system, including regular updates on policies and procedures.
- Optimize seating capacity and minimize wait times through data analysis and trend identification.

  Offer additional features such as online reservations, mobile notifications, and loyalty programs to enhance customer satisfaction and loyalty.
- Integrate with other restaurant systems such as POS, inventory management, and marketing to provide a comprehensive solution for restaurant management.
- By addressing these problems and providing a comprehensive solution, the new reservation system can improve restaurant operations, enhance customer satisfaction, and increase revenue.

2.10 Requirements of the proposed system

2.10.1 Functional Requirements

Functional Requirements are product features or functions that developers must implement to enable users to accomplish their tasks. Functional Requirements explain how the system must work.[2]

Functional Requirements for the system are below:

- Customers should be able to register to make a reservation.
- Customers should be able to login.
- Customers should be able to make reservations online through the restaurant's website or mobile app.
- Once a reservation is made, the system should automatically assign a table based on the number of guests and seating preferences.
- Staff should have access to a dashboard where they can view and manage all reservations, including adding, modifying, or canceling bookings.
- If all tables are booked, the system should allow customers to join a waitlist and receive notifications when a table becomes available.
- The system should provide reports on reservation trends, no-show rates, peak hours, and other key metrics to optimize table management and staffing.
- The system should support various payment methods, such as cash, credit/debit cards, and mobile payments.
- The system should notify the staff the time of reservation.

2.10.2 Non-Functional Requirements

Non-Functional Requirement is a requirement that does not relate to functionality. Non-Functional Requirement explains how the system should perform. [2]

Non-Functional Requirements for the system are below:

- **Performance**: The system should be able to handle a high volume of reservation requests without any lag or delays, ensuring a smooth and efficient booking process for customers.
- **Reliability**: The reservation system should be reliable and available 24/7, with minimal downtime to prevent any disruption in the booking process.
- **Security**: The system should have robust security measures in place to protect customer

data, such as encryption of sensitive information and compliance with data protection regulations.

- **Scalability**: The system should be scalable to accommodate an increasing number of reservations as the restaurant grows, without compromising performance or user experience.
- **Compatibility**: The reservation system should be compatible with different devices and web browsers to ensure a seamless booking experience for customers across various platforms.
- **Accessibility**: The system should be accessible to users with disabilities, following web accessibility guidelines to ensure that all customers can make reservations easily.

- **Data backup and recovery**: The system should have regular data backups and a reliable recovery plan in place to prevent any loss of reservation information in case of system failure or data corruption.
- **Integration capabilities**: The reservation system should have the ability to integrate with other systems used by the restaurant, such as the POS system, to streamline operations and ensure accurate table assignments.
- **Maintenance and support**: The system should have regular maintenance schedules and dedicated support services to address any technical issues or customer queries promptly.
- **User experience**: The reservation system should provide a user-friendly interface with clear instructions and intuitive navigation to enhance the overall customer experience and encourage repeat bookings.
- **Portability:** A server that supports MYSQL and PHP should be able to run the system.
- **Usability:** The system should be intuitive and easy to use for both customers making reservations and restaurant staff managing the system.

CHAPTER THREE: System Analysis

3.1 Introduction

We saw both functional requirement and non-functional requirement in previous chapter. In this chapter, our team use UML modeling tools to analyze the proposed system. this chapter includes use case diagram, sequence diagram, activity diagram and analysis level class diagram.

3.2 System requirement specification (SRS)

This section main focus is to create the system requirement specification document. Specification requirement document includes the description of functionality and behavior of the new system which includes actor and system interactions which will be describe in the use case, object interactions which will be describe by the sequence diagram, and the flow of work which will be also describe by activity diagram. The project team will do these in the subsequent section.

3.2.1 Use case diagram

Use case diagram is a way of describing the interaction that exists between the actor and the system. Use case diagram is helpful for understanding the needs of a system by modeling its behavior. the relationships between the actors in the system are also depicted in these diagrams. [3][4]

- **Use case**: describes a sequence of actions that provides a measurable value to an actor and is drawn as a horizontal ellipse. [3][4]
- **Actors**: Is a user that interacts with the system you are modeling is represented by an actor. a machine, an organization, or another external system can be all considered as users. [3][4]
- **Boundary system:** A rectangle which is drawn to separate the use cases that are internal to the system form the actors that is external to the system. [3][4]

Actor Identification

Roles for "users" of a system, such as human users and other systems are modeled and represented by actor classes. We determine the following users of our system:

1. customers
2. Admin
3. Waiter

4. cashier
5. cooker

Figure 2 use case diagram

3.2.2 Use case description

Use case documentation for login

Use case Name:	Login
ID:	UC-01
Description:	The system allows the customer and the admin to login to the system.
Actors:	Admin, Users (Customer, Receptionist, Cashier, Cooker).
Pre-condition:	The Users and admin must have username and password to login.
Basic flow:	The admin or user access login. The system displays login page. The admin or user type username, password and click login.(A) The system validates information.(A) The system loads homepage. The use case ends.
Alternative flow:	If the admin or users enters incorrect username or password. A.3. The system displays “incorrect username or password” message. And prompt the admin or user to enter valid data. A.4. Use case resume at step 2 of basic flow.
Post-condition:	The admin or the user uses their own pages.

Table 3 Use case description for login

Use case documentation for register

Use case name:	Register
ID:	UC-02
Description:	The system allows customer to registration.
Actors:	Customers.
Pre-condition:	
Basic flow:	The user accesses the login page. The user access register button. The user inputs his/her information. (A) The system validates the entered information. (A) The user is registered. The use case ends.
Alternative flow:	If the user enters invalid information. A.3. The system displays “Invalid entry” message. A.4. The system displays the register form page.
Post-condition:	The user will use the website.

Table 4 use case description for register

Use case documentation for make reservation

Use case name:	Make reservation
ID:	UC-03
Description:	The system allows the customer to make reservation.
Actors:	Customers
Pre-condition:	The customer must click “make a reserve” button.
Basic flow:	The Customer clicks “make a reserve” button. The system displays reservation form page. The Customer inputs necessary information. (A) The Customer clicks “reserve” button. The system prompts “you have made a reservation” message. Use case ends.
Alternative flow:	If the Customer does not fulfill the required information. A.3. The system displays “please fill in required information” message.
Post-condition:	The system will generate reserve.

Table 5 use case description for making reservation

Use case documentation for make reservation with food

Use case name:	Make reservation (With food)
ID:	UC-04
Description:	The system allows the customer to make reservation.
Actor:	Customer
Pre-condition:	The customer must click “make a reserve” button.
Basic flow:	The Customer clicks “make a reserve” button. The system displays reservation form page. The Customer inputs necessary information. (A) The Customer selects food option. The system displays food preferences. The Customer chooses his/her preferred food. The Customer clicks “reserve” button. The system prompts “you have made a reservation” message. Use case ends.
Alternative flow:	If the Customer does not fulfill the required information. A.3. The system displays “please fill in required information” message.

Post-condition:	The system will generate reserve.

Table 6 use case description for making reservation (with food)

Use case documentation for make reservation without food

Use case name:	Make reservation (Without food)
ID:	UC-05
Description:	The system allows the customer to make reservation.
Actor:	Customer
Pre-condition:	The Customer must click “make a reserve” button.
Basic flow:	The Customer clicks “make a reserve” button. The system displays reservation form page. The Customer inputs necessary information. (A) The Customer clicks “reserve” button. The system prompts “you have made a reservation” message. Use case ends.
Alternative flow:	If the Customer does not fulfill the required information. A.3. The system displays “please fill in required information” message.
Post-condition:	The system will generate reserve.

Table 7 use case description for making reservation (without food)

Use case documentation for waitlist management

Use case name:	Waitlist management
ID:	UC-06
Description:	The system puts the user on waitlist.
Actor:	Admin, customer
Pre-condition:	All available tables are reserved.
Basic flow:	The admin checks available tables in the system. The system prompts "There is no available table at the moment; you will be put on waitlist" message. (A) The Customer clicks "ok" button.
Alternative flow:	If the user does not want to be on waitlist. A.2. The user clicks "cancel" button.
Post-condition:	The system generates waitlist.

Table 8 use case description for waitlist management

Use case documentation for view reservation

Use case name:	View reservation
ID:	UC-07
Description:	The System allows Admin to view reservation.
Actor:	Admin, Users (Receptionist, Cashier, Cooker).
Pre-condition:	The Admin must login.
Basic flow:	The Admin and Users enter username and password. The System validates input information. (A) The admin and user login. The admin and users click “view” button. Use case ends.
Alternative flow:	If the Admin enters invalid input A.2. The System prompts “Invalid input” message.
Post-condition:	The reservation viewed.

Table 9 use case description for view reservation

Use case documentation for assign table

Use case name:	Assign Table
ID:	UC-08
Description:	The System allows Admin to assign table.
Actor:	Admin, Receptionist
Pre-condition:	The Admin and Receptionist must login.
Basic flow:	The Admin and Receptionist enter username and password. The System validates input information. (A) The Admin and Receptionist clicks “assign table” button. The System prompts “Assigning table success” message.
	Use case ends.
Alternative flow:	If the Admin and Receptionist enters invalid input A.2. The System prompts “Invalid input” message.
Post-condition:	Table assigned.

Table 10 use case description for assign table

Use case documentation for cancel reservation

Use case name:	Cancel Reservation
ID:	UC-09
Description:	The customer Cancel Reservation.
Actor:	Customer
Pre-condition:	The Customer must login.
Basic flow:	The Customer enters username and password. The System validates input information. (A) The Customer clicks “Cancel Reservation” button. The System prompts “The Reservation Cancel please visit again” message. Use case ends.
Alternative flow:	If the Customer does not fulfill the required information. A.2. The system displays “please fill in required information” message.
Post-condition:	The system automatically canceled the reservation.

Table 11 use case description for cancel reservation

Use case documentation for add or remove food item

Use case name:	Add or Remove Food Item
ID:	UC-10
Description:	The Admin add or remove Food Item.
Actor:	Admin
Pre-condition:	The Admin must login.
Basic flow:	The Admin enters username and password. The System validates input information. (A) The Admin clicks “Add” or “Remove” button. The System prompts “Food Item adds” or “Food Item remove” message. Use case ends.
Alternative flow:	If the Admin enters invalid input A.2. The System prompts “invalid input” message.
Post-condition:	New Food Item added or Food Item removed.

Table 12 use case description for Add/Remove Food Item

Use case documentation for report

Use case name:	Report
ID:	UC-11
Description:	Generate report
Actor:	Admin, Receptionist
Pre-condition:	The Admin and Receptionist must login.
Basic flow:	The Admin and Receptionist enter username and password. The System validates input information. (A) The Admin and Receptionist clicks “report” button. Use case ends.
Alternative flow:	If the Admin and Receptionist enters invalid input A.2. The System prompts “invalid input” message.
Post-condition:	Report generated.

Table 13 use case description for report

Use case documentation for print

Use case name:	Print
ID:	UC-12
Description:	Print report
Actor:	Receptionist
Pre-condition:	The Receptionist must login.
Basic flow:	The Receptionist enters username and password. The System validates input information. (A)
	The Receptionist clicks “print” button. Use case ends.
Alternative flow:	If the Receptionist enters invalid input A.2. The System prompts “invalid input” message.
Post-condition:	The report printed.

Table 14 use case description for print

Use case documentation for payment

Use case name:	Payment
ID:	UC-13
Description:	The Customer will pay for reservation.
Actor:	Customer, Cashier
Pre-condition:	The Customer and Cashier must login.
Basic flow:	The Customer enters username and password. The System validates input information. (A) The Customer clicks “pay” button. The Cashier validates payment. Use case ends.
Alternative flow:	If the Customer does not fulfill the required information. A.2. The system displays “please fill in required information” message.
Post-condition:	The Customer paid for reservation.

Table 15 use case description for payment

Use case documentation for notification

Use case name:	Notification
ID:	UC-14
Description:	The system notifies users for reservation
Actor:	Customer, Cooker and Receptionist
Pre-condition:	The Customer makes reservation
Basic flow:	The system prompts" there is reservation by this time" message.
Alternative flow:	
Post-condition:	The system notified Customer, Cooker and Receptionist.

Table 16 use case description for notification

Use case description for make payment

Use case name:	Handle payment
ID:	UC-15
Description:	External payment system integrated with the payment of the given system.
Actor:	External System
Pre-condition:	
Basic flow:	External system controls the payment of the customer and return necessarily information for cashier.

*Table 17**Use case description for make payment***

3.2.3 Sequence diagram

Sequence diagram is a type of interaction diagram that describes how and in what order a group of objects works together.[5]

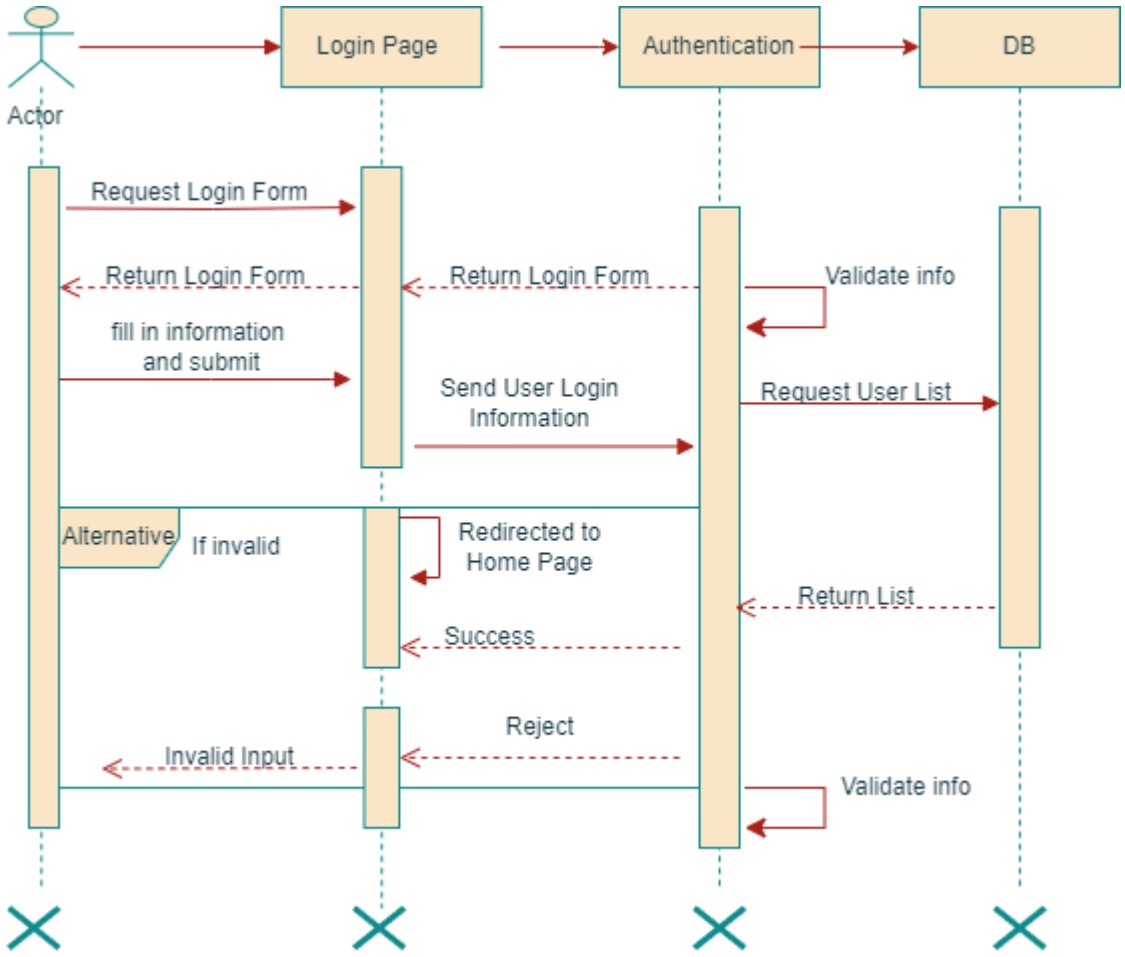


Figure 3 ***sequence diagram for login***

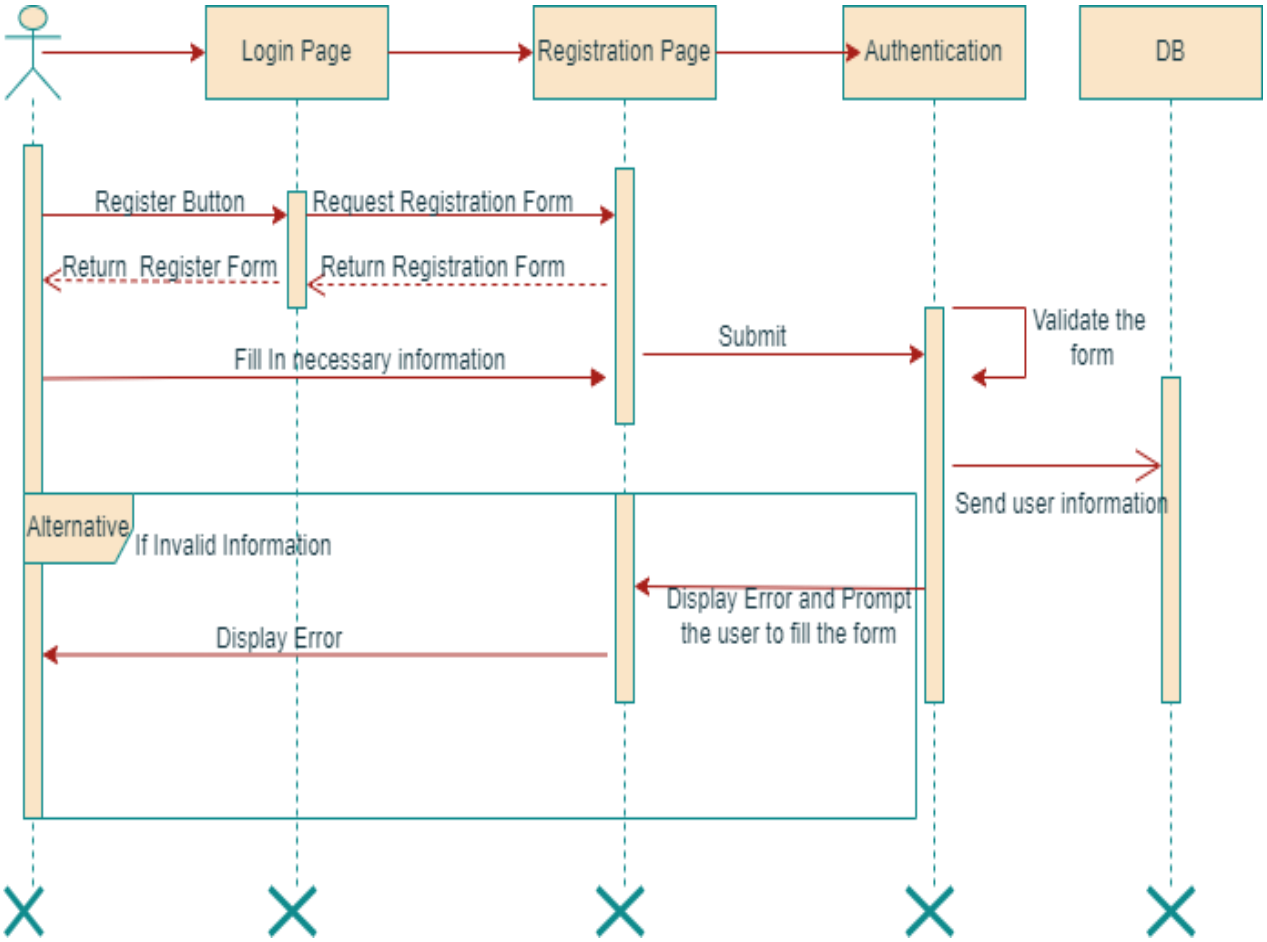


Figure 4 ***Sequence diagram for register***

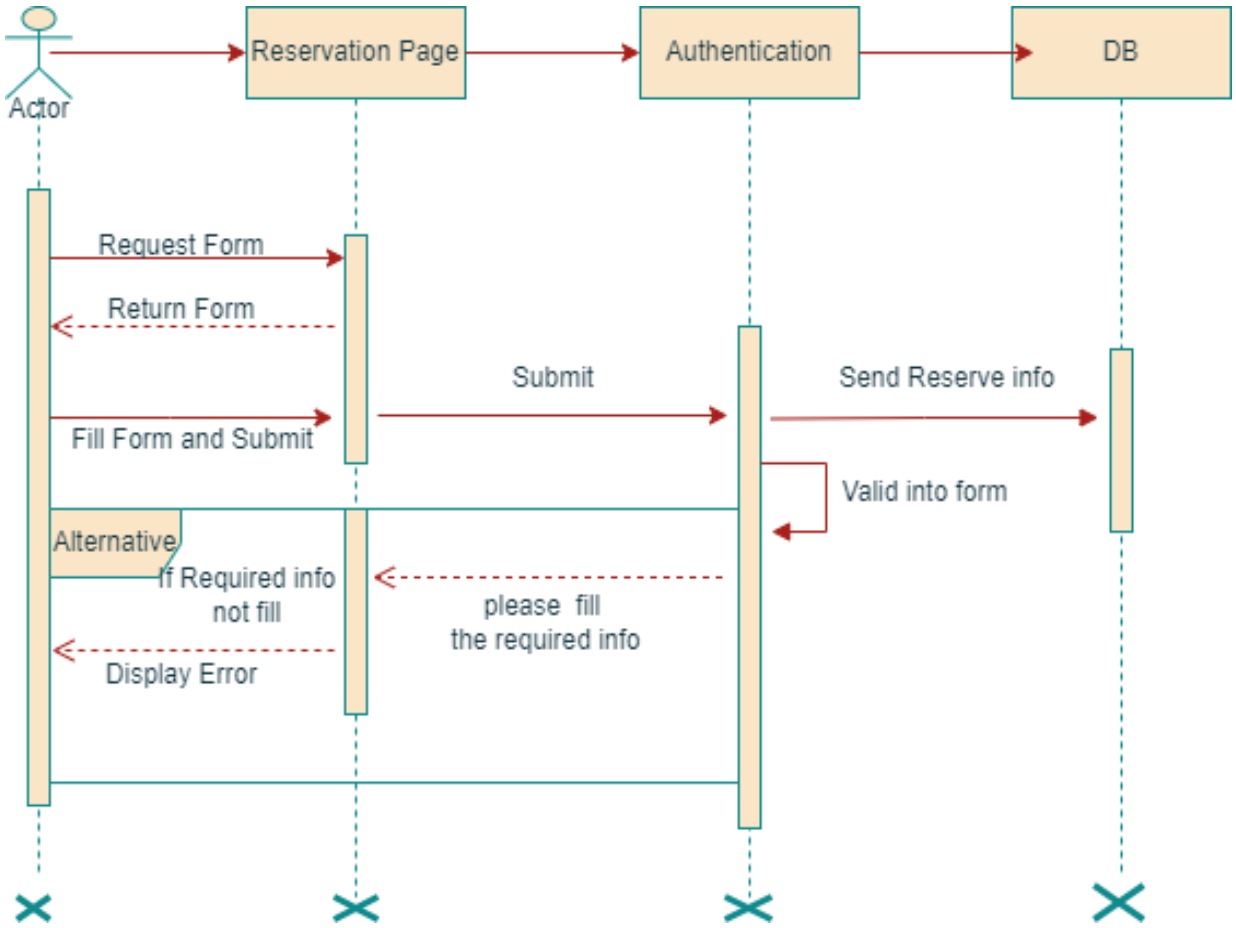


Figure 5 ***Sequence diagram for make reservation***

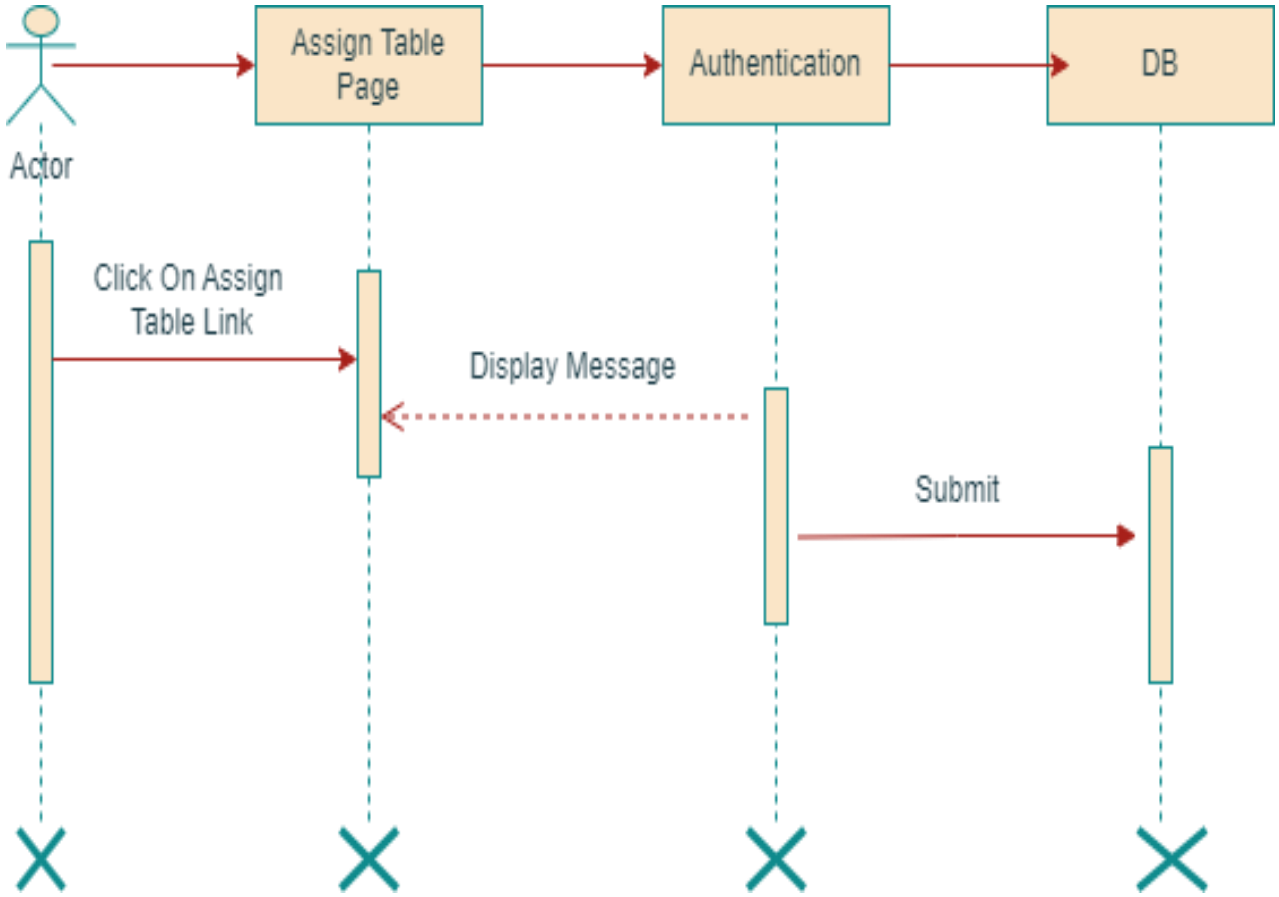


Figure 6 ***Sequence diagram for assign table***

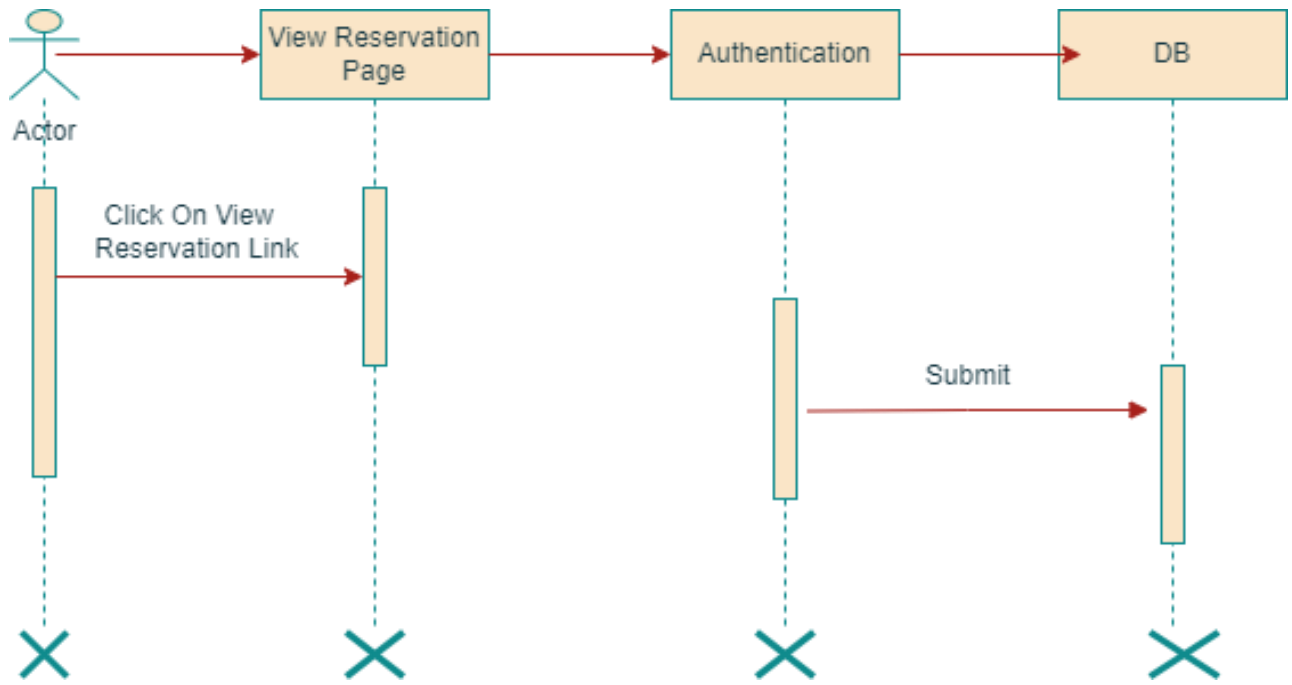


Figure 7 ***Sequence diagram for view reservation***

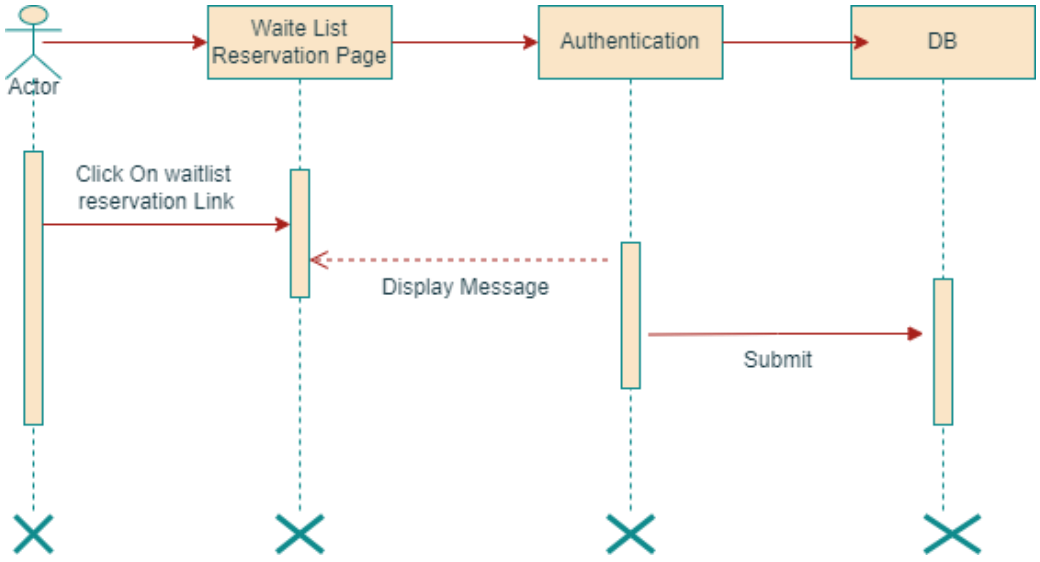


Figure 8 ***Sequence diagram for waitlist management***

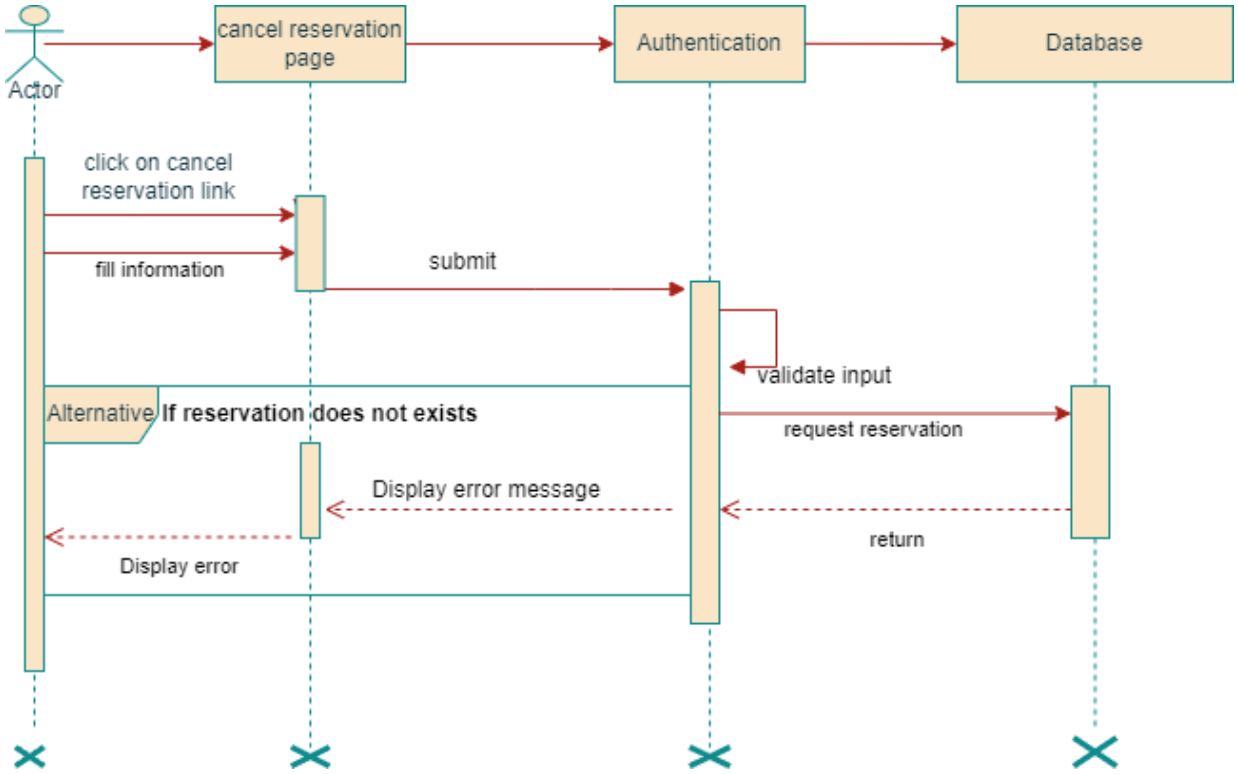


Figure 9 ***Sequence diagram for cancel reservation***

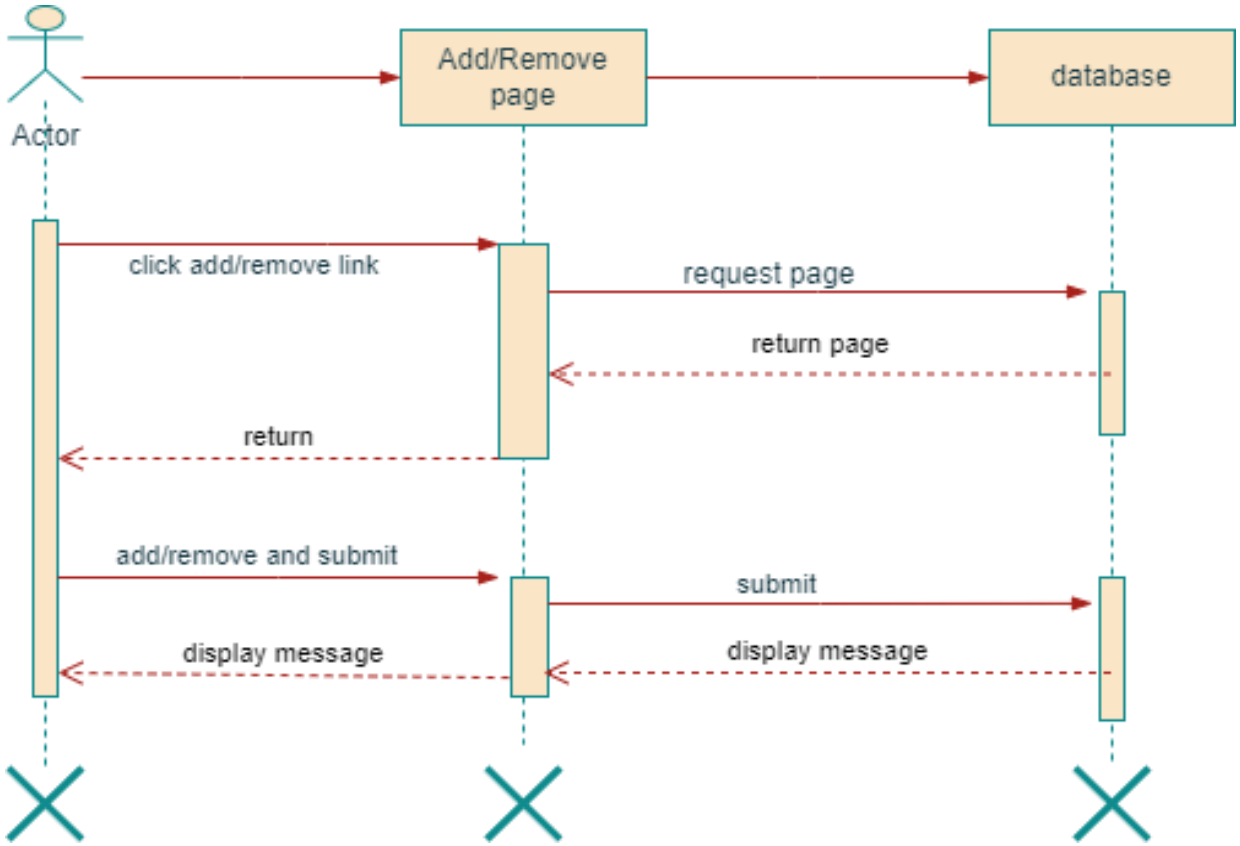


Figure 10 ***Sequence diagram for add or remove food item***

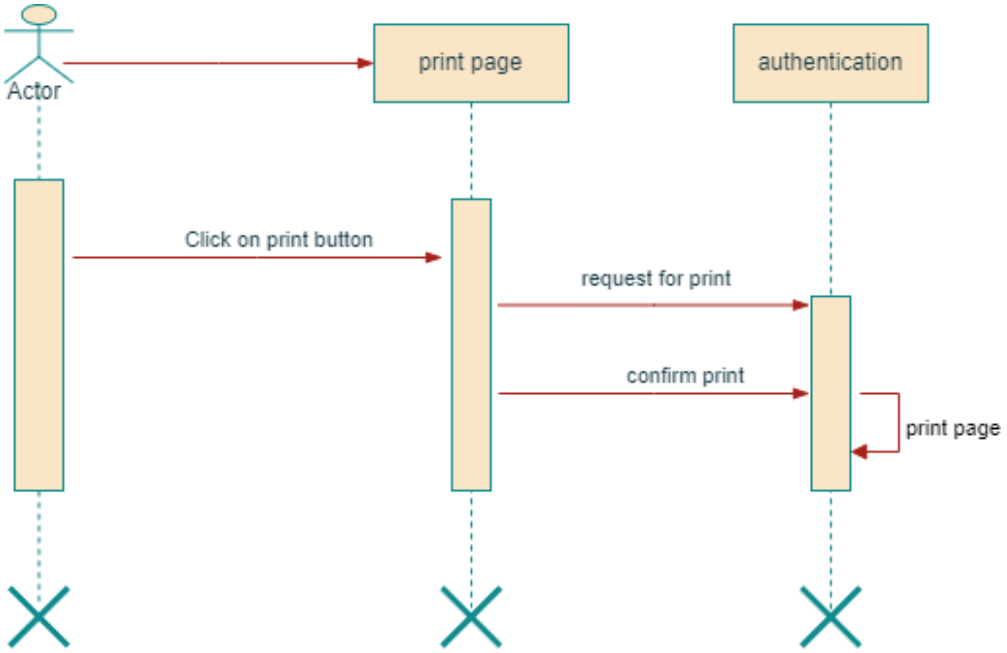


Figure 11 ***Sequence diagram for print***

3.2.4 **Activity diagram**

Activity diagram is another important behavioral diagram in the UML diagram to describe dynamic aspects of the system. Activity diagram is a more complex dorm of flow chart that illustrates how an activity flows from one to next. [12]

The team identified and prepared the following activity diagram for the proposed system.

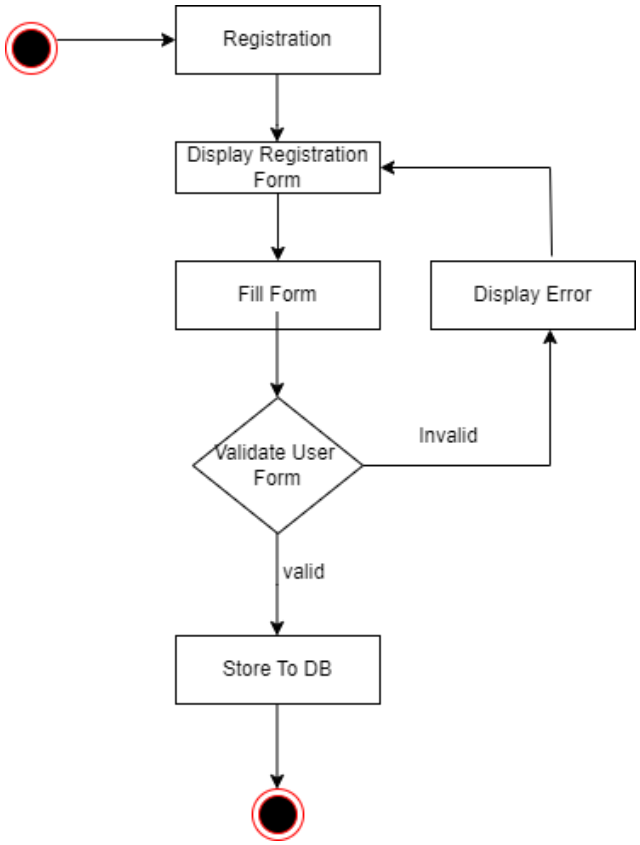


Figure 12 ***Activity diagram for register***

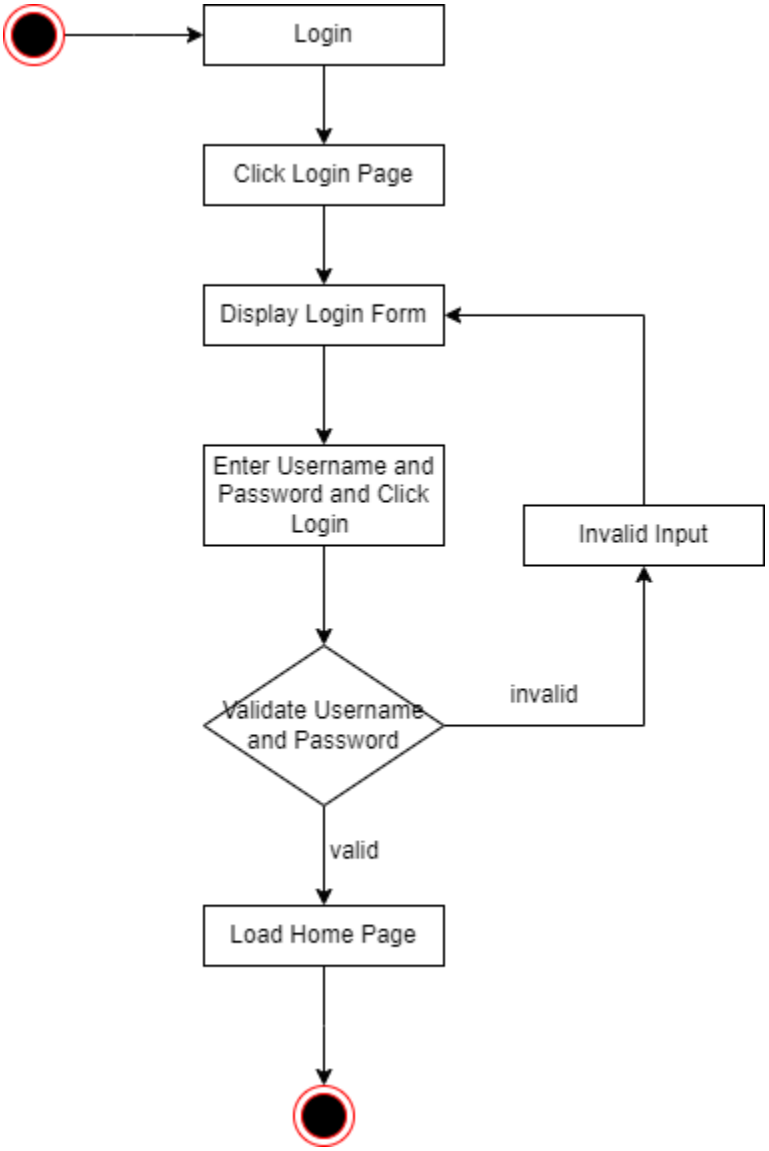


Figure 13 ***Activity diagram for login***

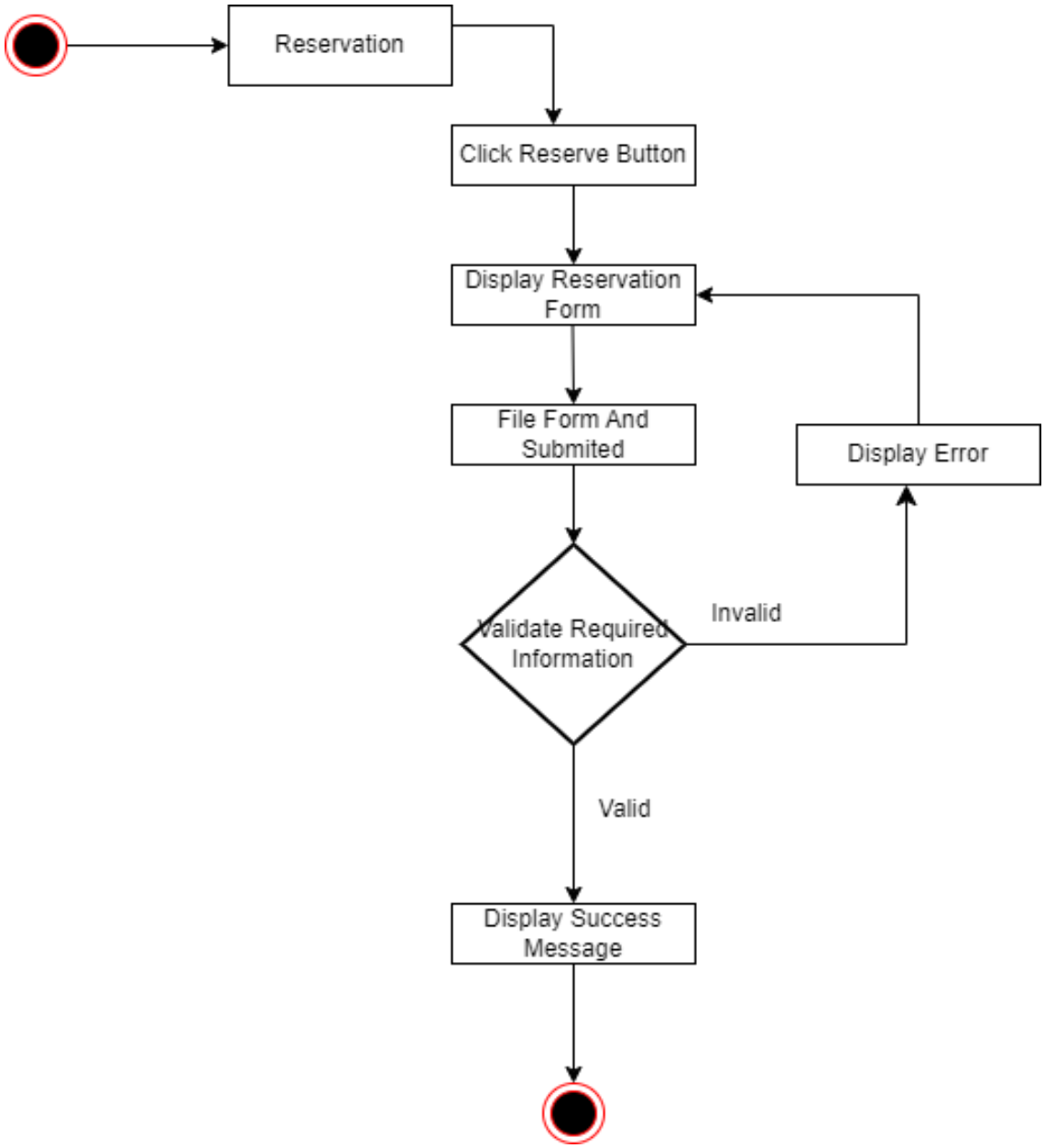


Figure 14 ***Activity diagram for make reservation***

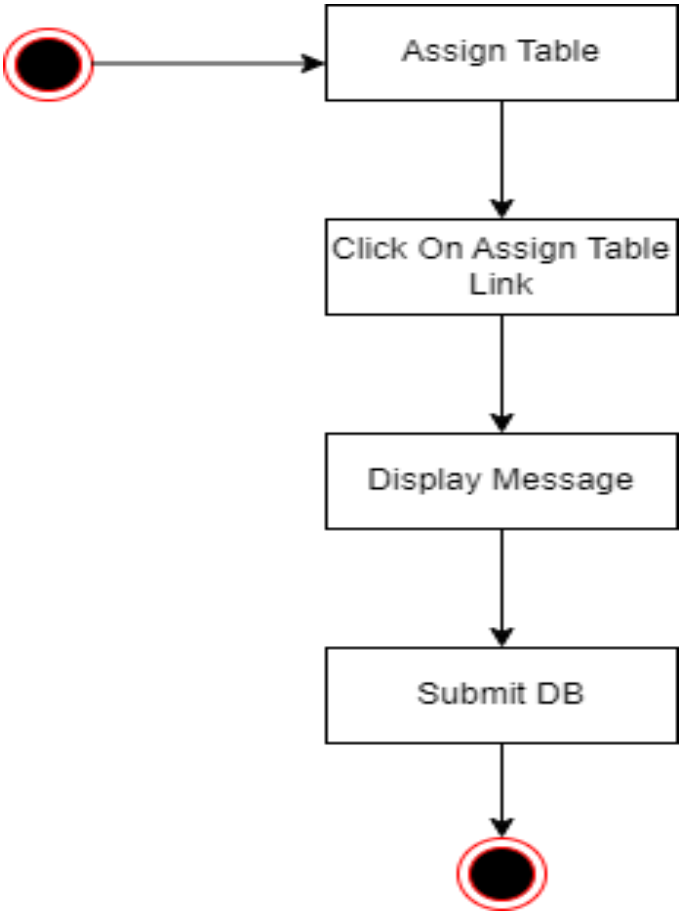


Figure 15 ***Activity diagram for assign table***

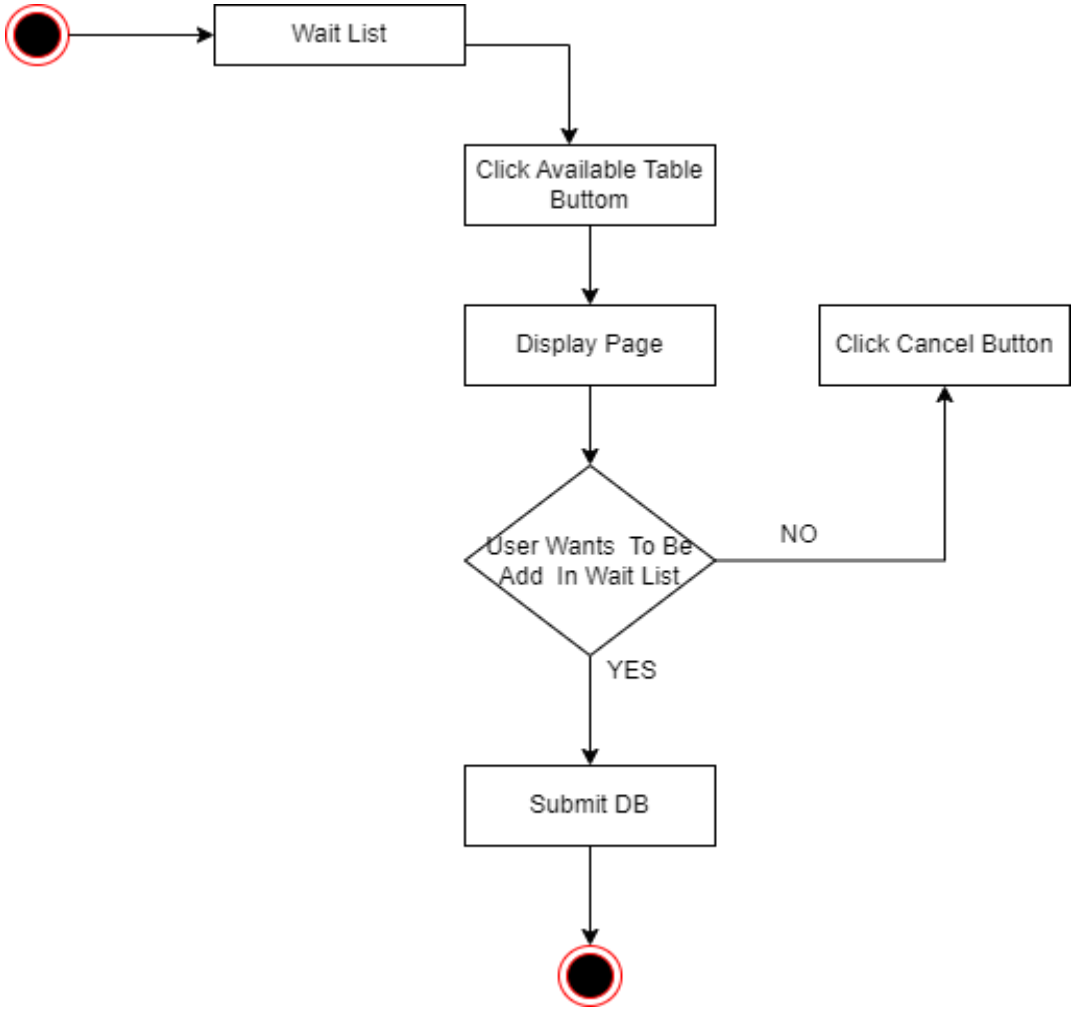


Figure 16 ***Activity diagram for waitlist management***

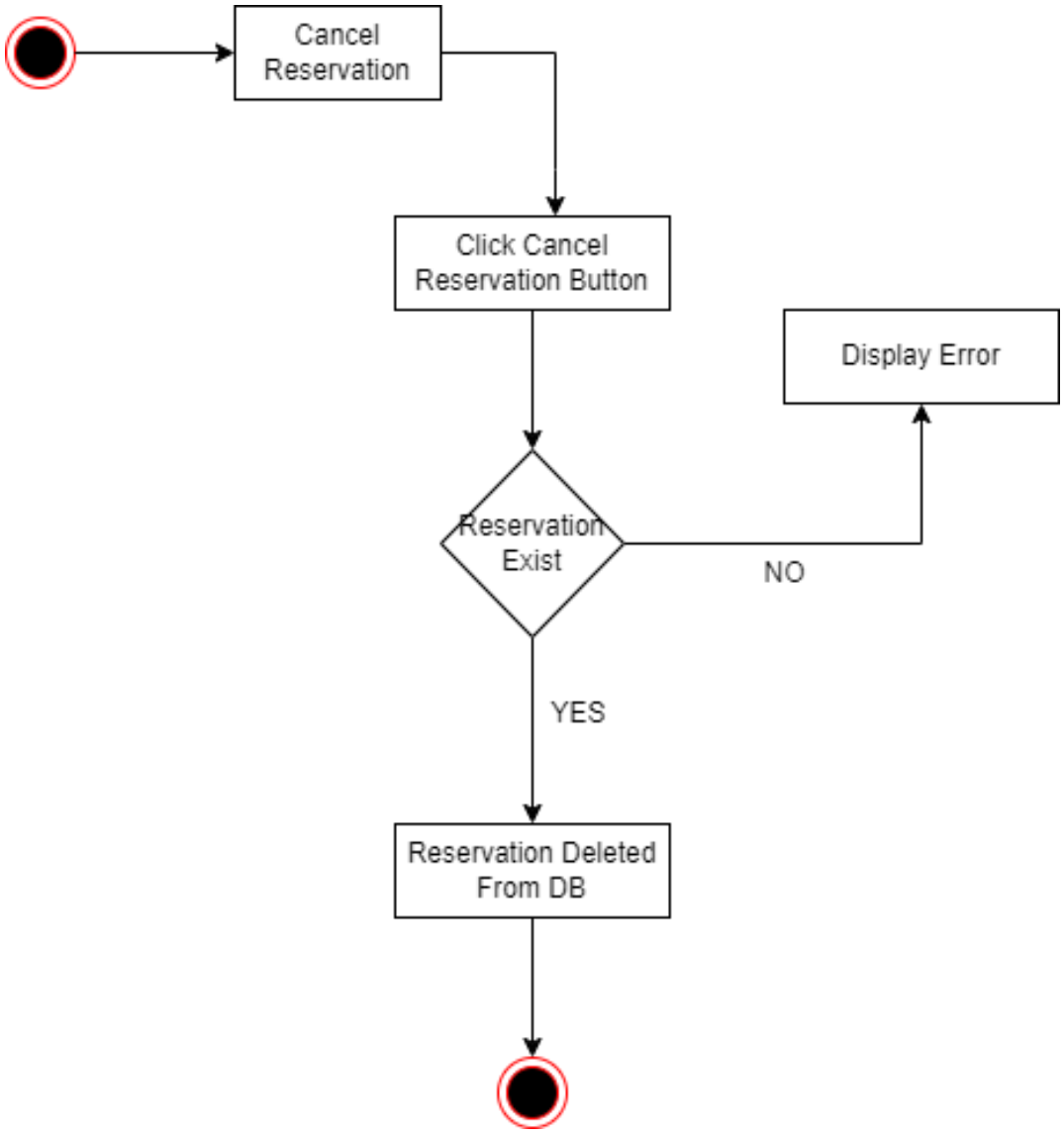


Figure 17 ***Activity diagram for cancel reservation***

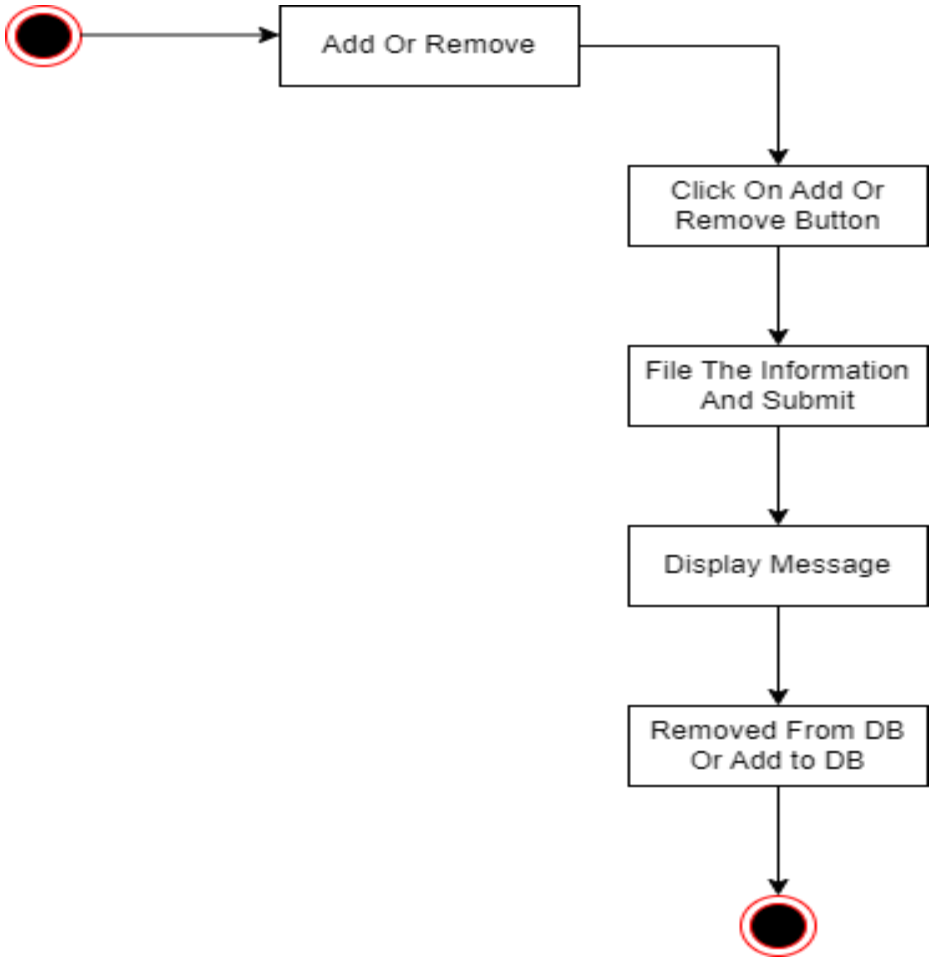


Figure 18 ***Activity diagram for add or remove food item***

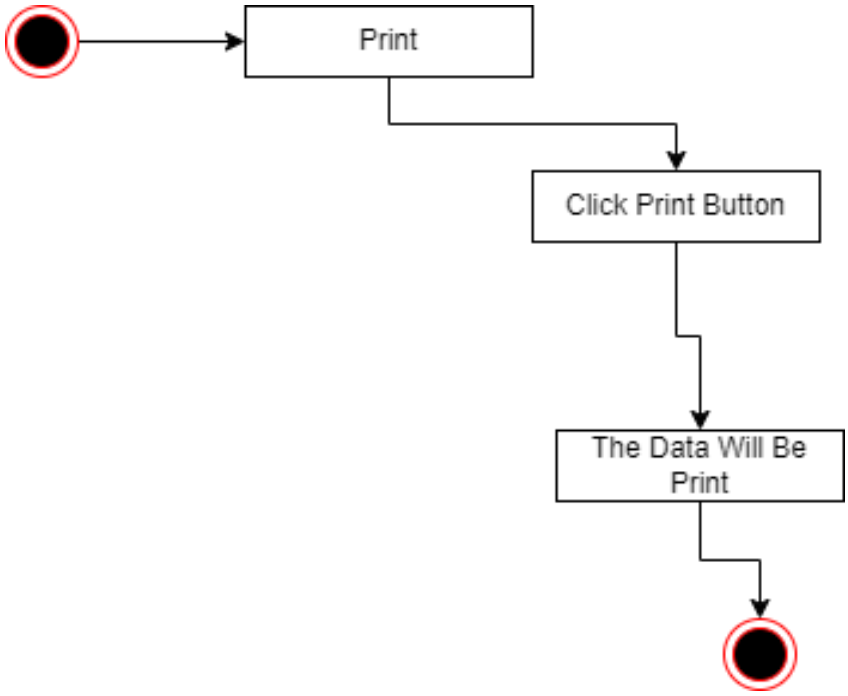


Figure 19 ***Activity diagram for print***

analysis level class diagram

the class diagram is the main building block of object-oriented modeling. it is used for static diagrams. it represents the static view of an application. the class diagram figure below shows various objects in our system, their attributes, operations, and relationships among them.[13]

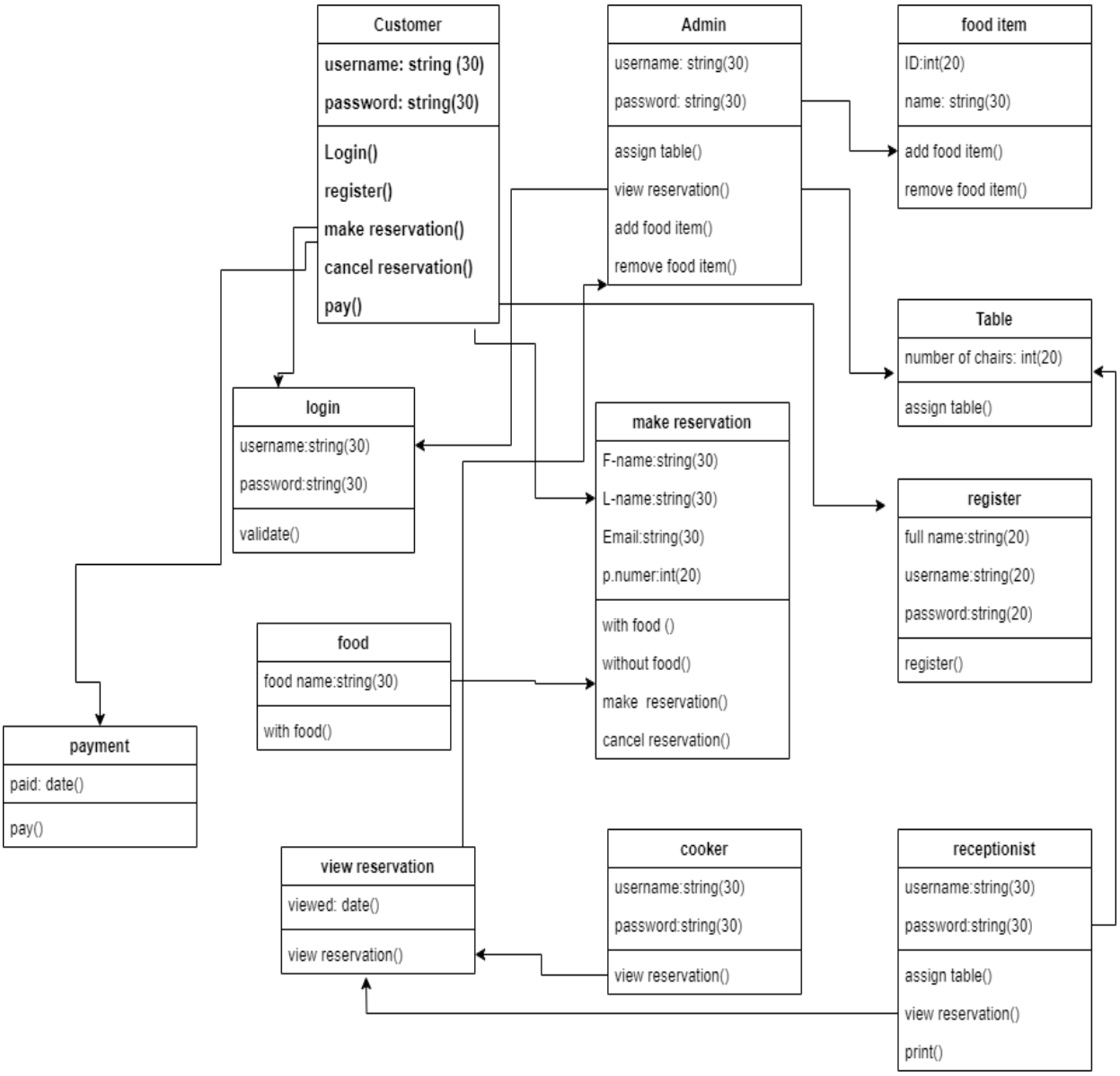


Figure 20 Conceptual level class diagram

User interface prototyping

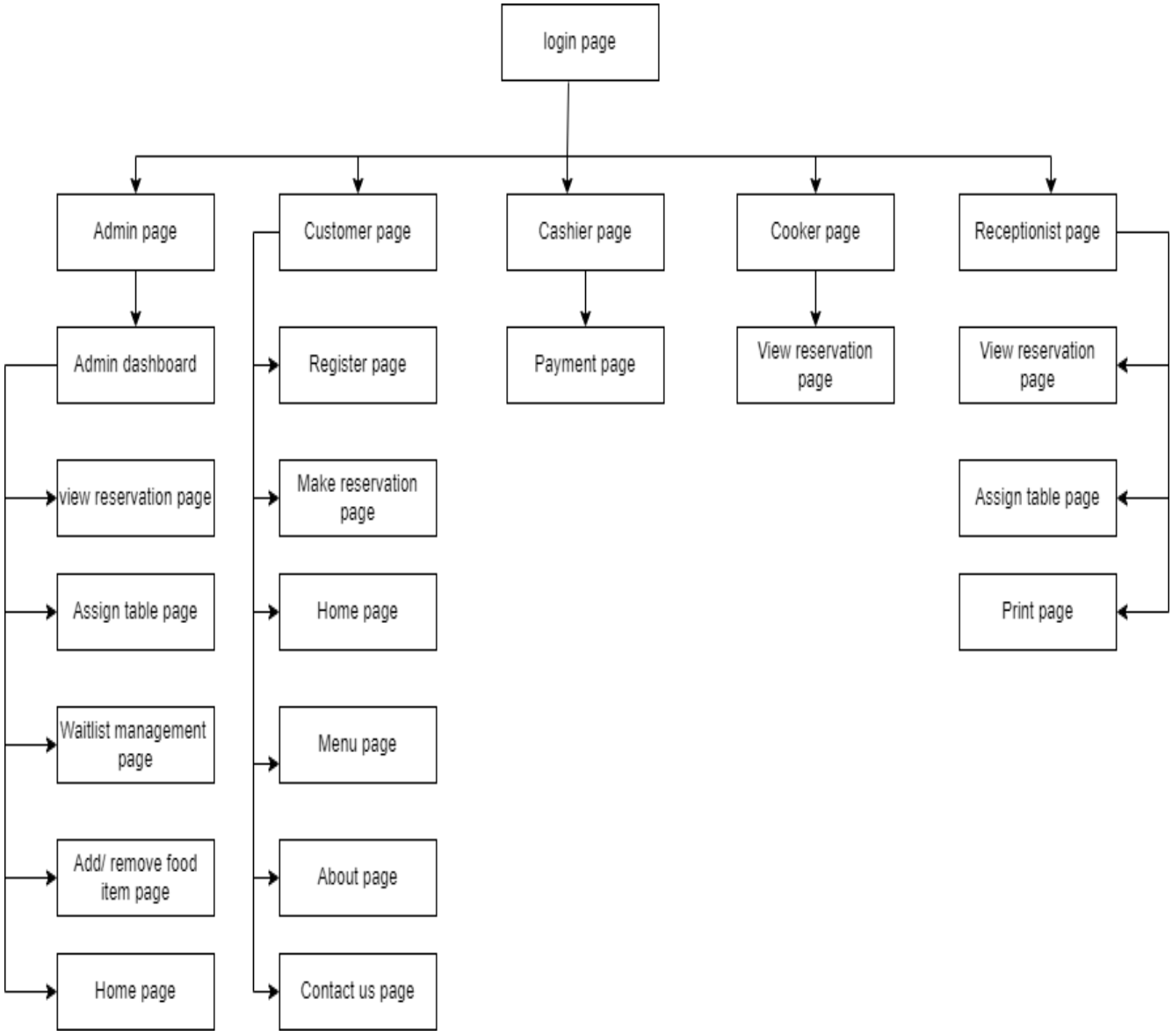


Figure 21 User interface prototype of the new system

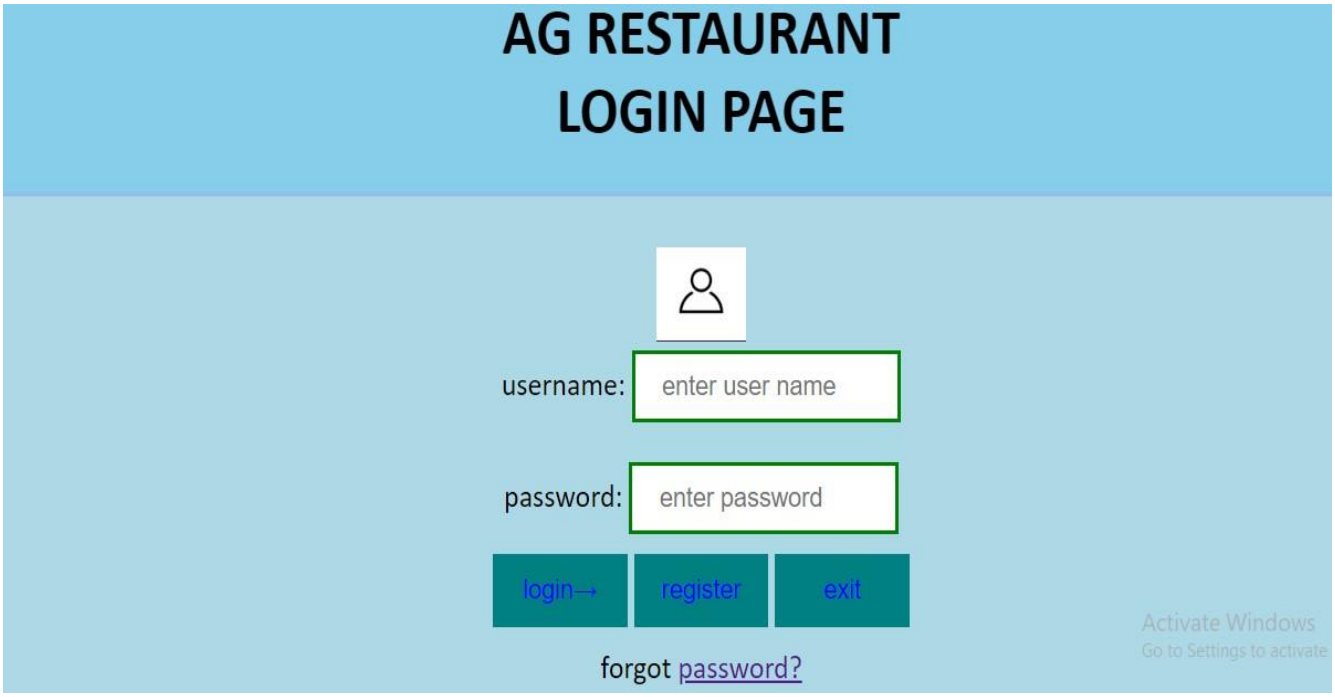


Figure 22 User interface prototype for login

CHAPTER FOUR: System Design

4.1 Introduction

In this chapter, our team will do the class type architecture, class modelling, deployment modelling, persistence modelling and user interface design of our system. Converting the analytical model into a system design model is known as system design.

4.2 Class Type Architecture

The class type architecture is a variant of the standard layered architecture that organizes classes and objects based on their responsibilities.[8]

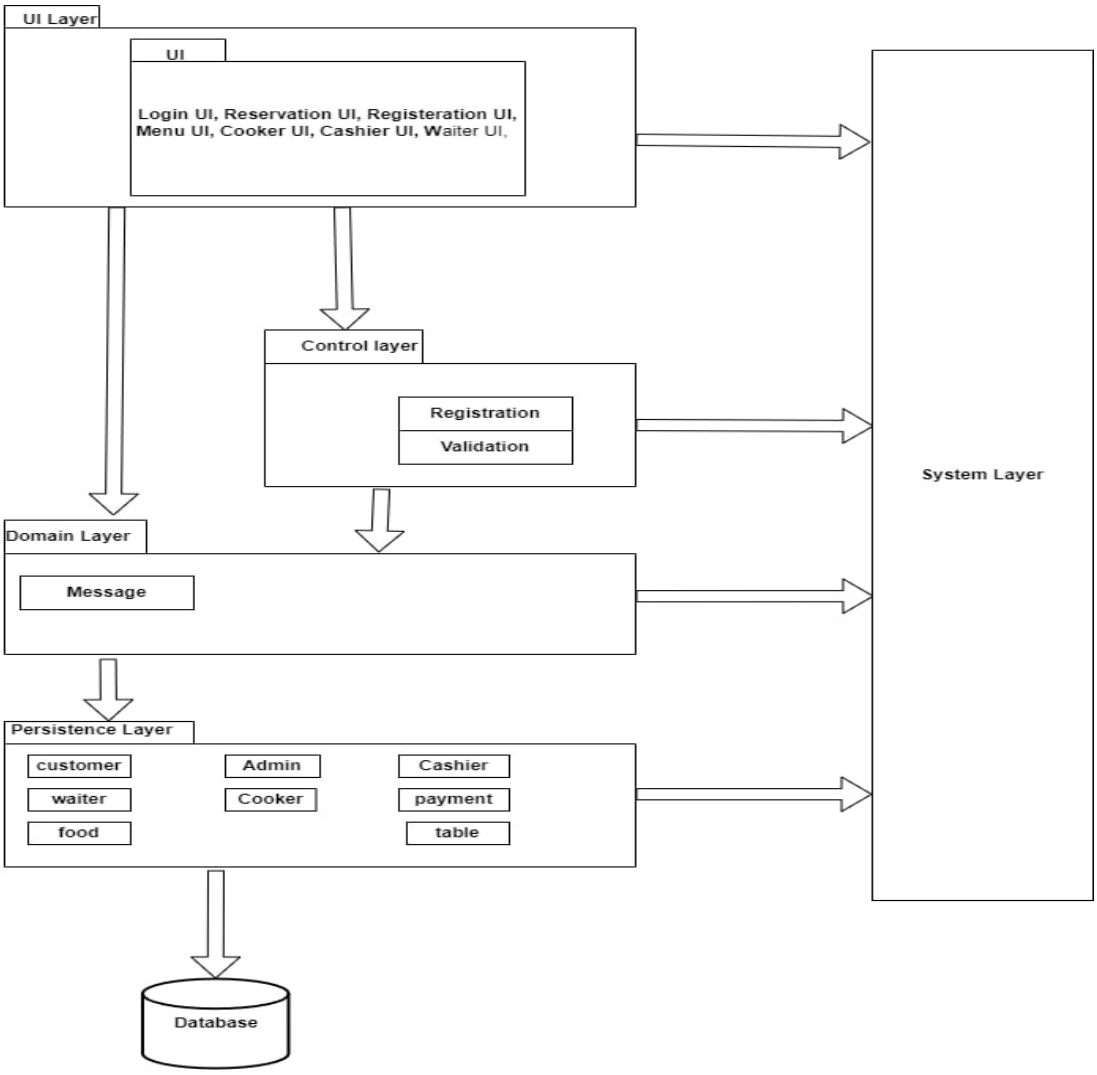
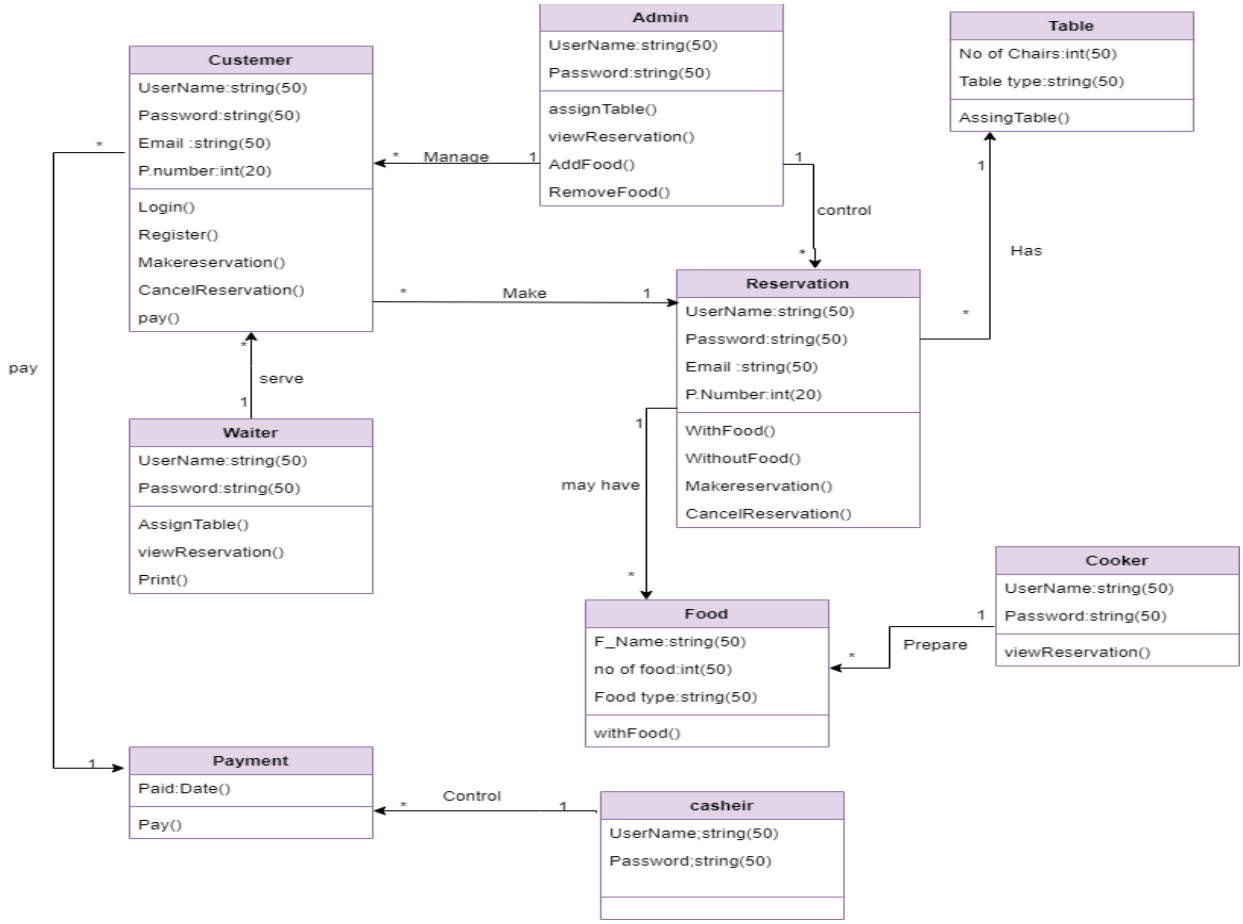


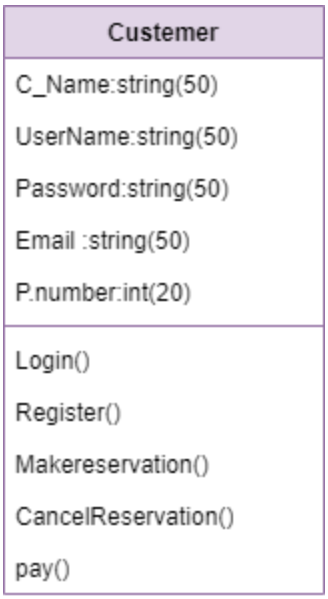
Figure 23 The above class type architecture shows the interactions between the entire classes.

4.3 Class Modeling

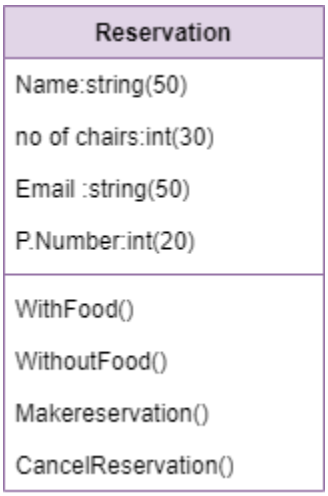
There is no movement in the class diagram. It depicts an application's static view. A class diagram is utilized not only to build the executable code of a software application but also to

visualize, describe, and record various features of a system. The class diagram explains the characteristics and functions of a class as well as the limitations placed on the system.[9]
A group of classes, interfaces, associations, collaborations, and constraints are displayed in the class diagram. Another name for it is a structural diagram. A class diagram illustrates the class, properties, actions (or methods), and relationships between objects in the system to define its structure.[9]

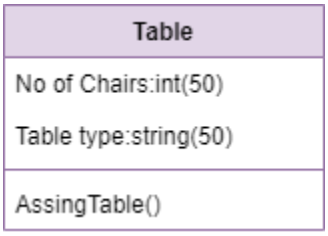




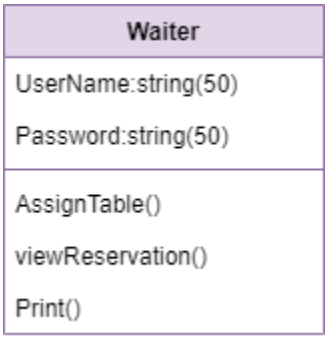
- **C_Name** - Is the name of the customer with string data type.
- **Username**-Is the name of the account used for verification with string data type.
- **Password**-Is the customer's pin used for verification with sting data type.
- **Email**-Is the customer's email account with string data type.
- **Phone number**-Is the customer's phone number with int data type.



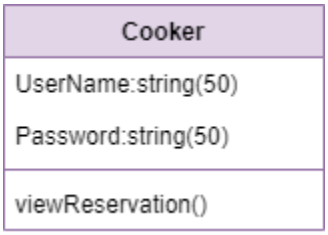
- **Name** - Is the name of the customer with string data type.
- **No of chairs**-is the number of chairs in a reservation with int data type
- **Email**-Is the customer's email account with string data type.
- **Phone number**-Is the customer's phone number with int data type.



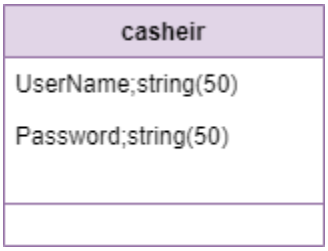
- **No of chairs**-Is the number of chairs in a single table with int data type.
- **Table type**- is the table type which is available for reservation.



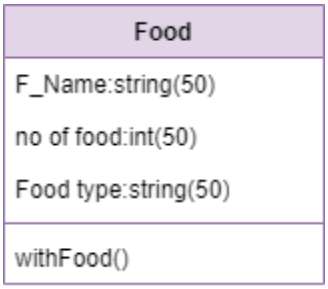
- **Username**-Is the name of the account used for verification with string data type.
- **Password-**Is the waiter's pin used for verification with string data type.



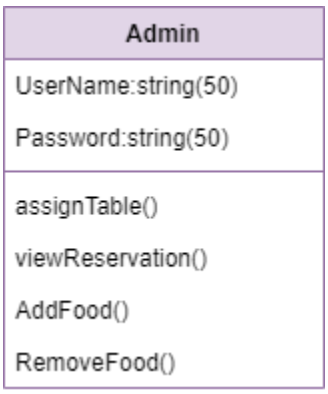
- **Username**-Is the name of the account used for verification with string data type.
- **Password-**Is the cooker's pin used for verification with string data type.



- **Username**-Is the name of the account used for verification with string data type.
- **Password**-Is the cashier's pin used for verification with string data type.



- **F_Name**-Is the name of the food with string data type.
- **Food type**-Is the food type which is exists in the menu with string data type.
- **No of food**-Is the number of foods in one reservation with int data type.



- **Username**-Is the name of the account used for verification with string data type.
- **Password**-Is the admin's pin used for verification with string data type.

4.4 Deployment Modeling

One type of structure diagram used to represent the physical components of an object-oriented system is the deployment diagram. The setup of run-time processing nodes and the components stored on them are shown in a UML deployment diagram. They are frequently used to simulate a system's static deployment view, or hardware topology.[10]

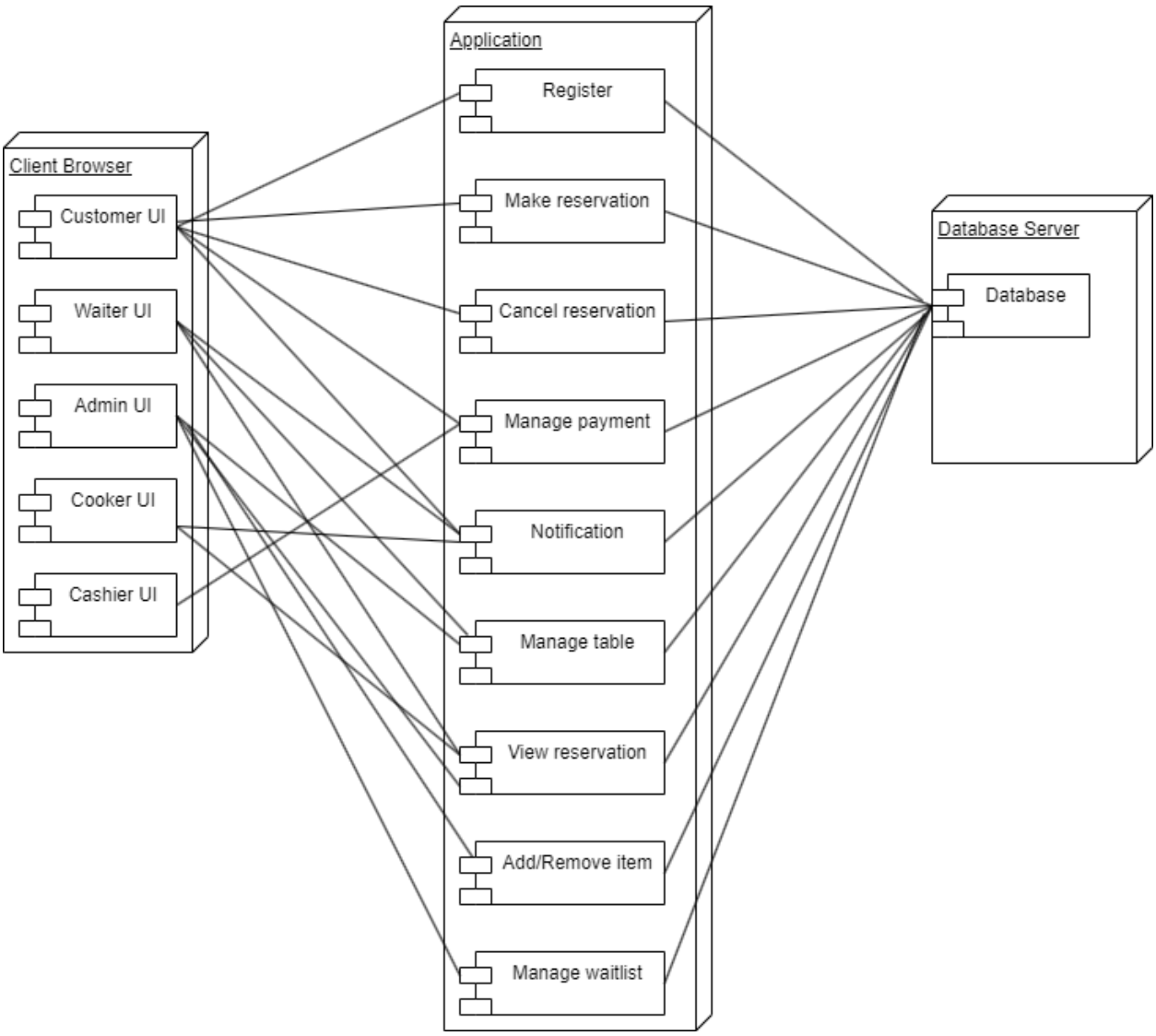


Figure 24 Deployment Diagram

4.5 Persistence Modeling

Persistence modeling is used to depict the design of the database. The persistent classes are used to store the most important and permanent information in the system. [11]

In persistence modeling the team member will perform the following activities:

- Identifying entities
- Identifying keys
- The methods of classes
- Data type

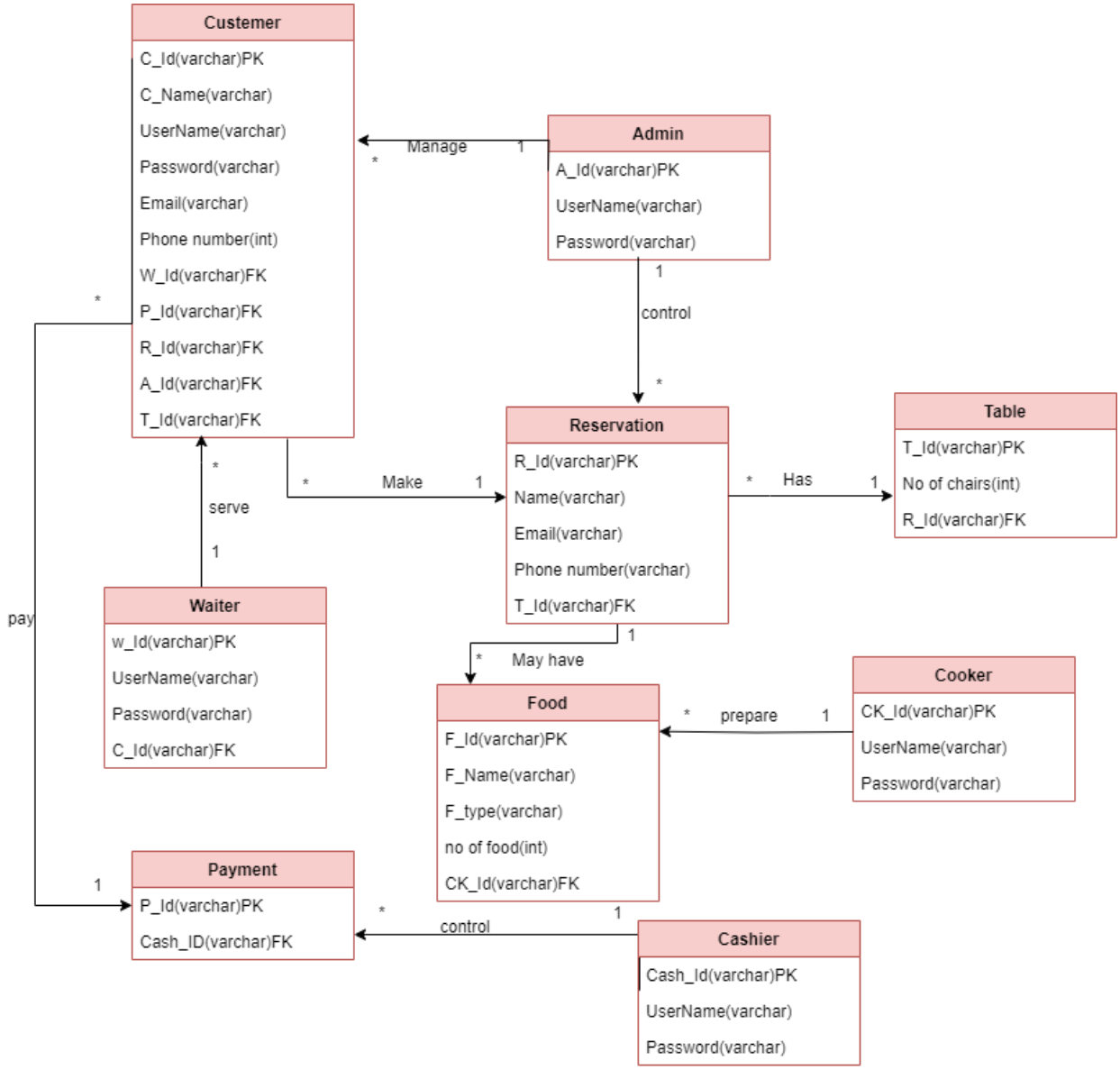


Figure 25 Persistence diagram

Customer

C_Id – primary key

- Varchar (50)
- Is a unique id that used to distinguish the customers.

C_Name

- Varchar (50)
- Is the name of the customer.

Username

- Varchar (50)
- Is the name of the account used for verification.

Password

- Varchar (50)
- Is the customer's pin used for verification.

Email
- Varchar (50)
- Is the customer's email account.

Phone number
- Int (30)
- Is the customer's phone number.

Admin

A_id – primary key
- Varchar (50)
- Is a unique id.

Username
- Varchar (50)
- Is the name of the account used for verification.

Password
- Varchar (50)
- Is the admin's pin used for verification.

Reservation

R_Id – primary key
- Varchar (50)
- Is a unique id that used to distinguish the reservations.

Waiter

W_Id – primary key
- Varchar (50)
- Is a unique id that used to distinguish the waiters.

Username
- Varchar (50)
- Is the name of the account used for verification.

Password
- Varchar (50)
- Is the waiter's pin used for verification.

Cooker

CK_Id – primary key
- Varchar (50)
- Is a unique id that used to distinguish the cooker.

Username
- Varchar (50)
- Is the name of the account used for verification.

Password
- Varchar (50)
- Is the cooker's pin used for verification.

Cashier

Cash_Id – primary key
- Varchar (50)
- Is a unique id that used to distinguish the cashier.

Username
- Varchar (50)
- Is the name of the account used for verification.

Password
- Varchar (50)
- Is the cashier's pin used for verification.

Table

T_Id – primary key

- Varchar (50)
- Is a unique id that used to distinguish the tables.

No of chairs

- Varchar (50)
- Is the number of chairs in a single table.

Food

F_Id – primary key

- Varchar (50)
- Is a unique id that used to distinguish the tables.

F_Name

- Varchar (50)
- Is the name of the food.

F_type

- Varchar (50)
- Is the type of the food which is exists in the menu.

No of food

- Varchar (50)
- Is the number of foods in one reservation.

Payment

P_Id – primary key

- Varchar (50)
- Is a unique id that used to distinguish the payment.

4.7 user interface design

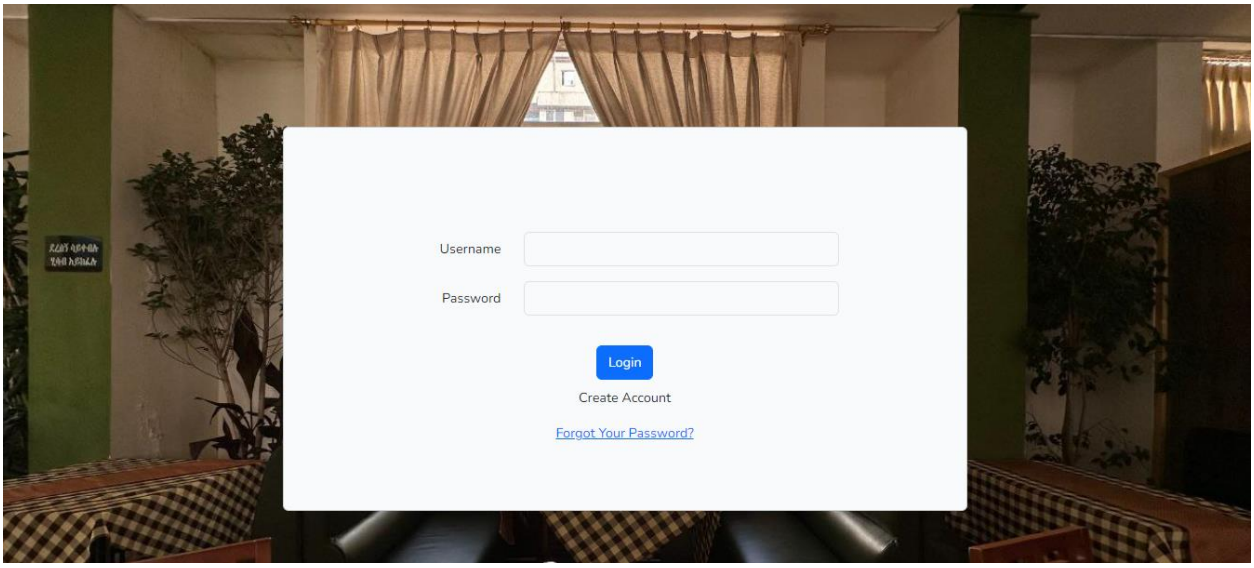


Figure 26 User interface design for login

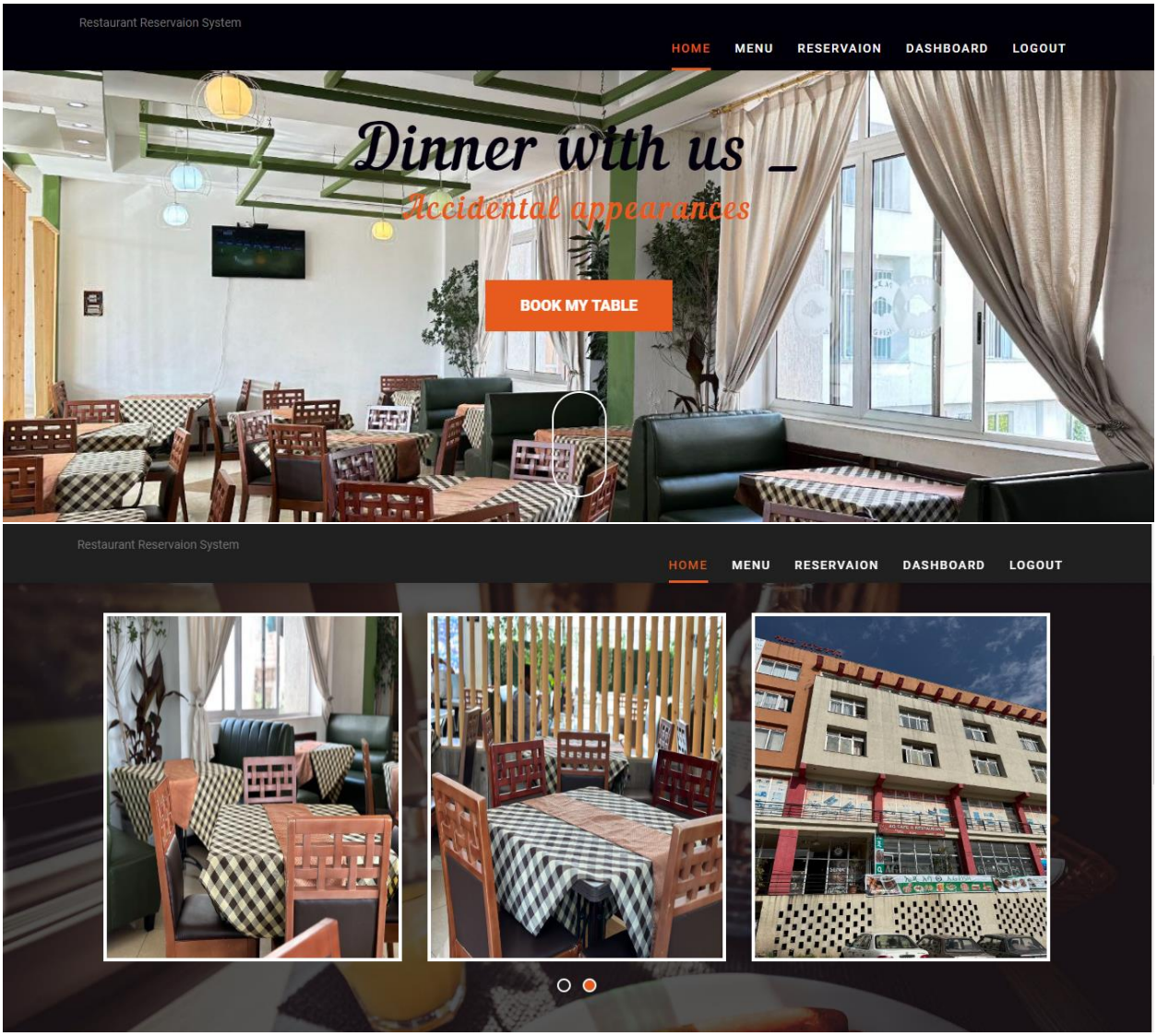


Figure 27 User interface design for homepage

CHAPTER FIVE: IMPLEMENTATION AND TESTING

5.1. Introduction

During this phase physical design specification must be turned into working computer code, and provide help for current and future users and take care of the system. And then the code is tested until most of the errors have been detected and corrected. The purpose of this activity is to convert the final physical system specification into working model with reliable software and hardware.

5.2 Final Testing of the system

Testing is a process to show the correctness of the program and designed to analyses the logic used in the implementation of the System. [14]

In the case of our project, we use unit, user acceptance, and integration testing method, which is briefly stated in proposal.

5.2.1. Unit testing

Each module is tested alone in an attempt to discover any errors in its code. In unit testing, each module (roughly a section of code that performs a single function) is tested alone in an attempt to discover any errors that may exist in the module's code. [15]

We tested the system as shown below in the appendix.

Unit testing, also known as component testing refers to tests that verify the functionality of a specific section of code, usually at the function level. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. The system has several Functionalities, forms or pages codes and links and all of them are tested individually.[15]

Test case 01	Register
Purpose	Testing registered users
Pre-request	Visit the site
Test data	Confirmation status------YES/NO YES->if the user fills the form correctly NO-> if the user fills the form incorrectly
Steps	1. Activate the system 2. Click registration button 3. The user fills the registration form 4. Click register button If YES—the system will display successful message
Expected result	The page will invite the user to login page.

Table 18 unit testing for registration

Test case 02	Login
Purpose	Testing logged users
Pre-request	Username and password
Test data	Confirmation status------YES/NO YES->if the user enters the correct username and password NO-> if the user enters the incorrect or **null** value
Steps	1. Activate the system 2. Click login button 3. The user enters username and password 4. Click login button If YES—the system will display successful message
Expected result	The page will invite the user to home page, admin page, cooker page, receptionist page.

Table 19 unit testing for login

5.2.2. Integration testing

The process of bringing together for testing purposes all of the modules that a program comprises. Modules are typically integrated in a top-down, incremental fashion. If an error occurs, the process stops, the error is identified and corrected, and the test is redone. The process repeats until the entire program-all modules at all level is successfully integrated and tested with no errors. After the team tested using unit testing the next step is integrating testing. In integrated testing the team tested the system all modules that a program contains.[16]

5.2.3. Acceptance testing

It is the process whereby actual users test a completed information system, the end result of which is the users' acceptance of it once they are satisfied with it. Acceptance refers to the fact that users typically sign off on the system and "accept" it once they are satisfied with it. Testing the system in the environment where it will be used. The purpose of acceptance testing is for users to determine whether the system meets their requirements. In acceptance test the team tests the system by different user that satisfies the requirements. The user tested by inserting real data.[17]

5.2.4. System testing

System testing is simply expanded integration testing, where you are testing the interfaces between programs in a system rather than testing the interfaces between modules in a program. System testing is also intended to demonstrate whether a system meets its objectives. It is the final step of testing. In this step the team members tests the entire system as a whole with all forms, code, modules and tests all the functionalities in the System. This form of testing is popularly known as Black Box testing or System tests. All errors in the forms, functions, modules are tested.[18]

5.3 Hardware software acquisitions

Hardware acquisition: Before actually continuing, it is necessary to have a web server to which the web application will be deployed; also, there are a few considerations that need to be made. To

name a few, the web server must have the XAMPP server installed. The database is connected to the server once it has been setup.

On the other hand, we will only attach the database if the server is already installed. After then, anybody can visit the website. Since the system will be used in a networked environment, the server and client hardware components must exist.

Server computer:

The server computer can effectively manage as many requests as the clients requesting access can make. As a result, we determined the following for our system's server computer, albeit it really depends on how many users there are in the business.

Client computer:

Within the client-server model of computer networks, a client is a piece of hardware or software that accesses a service provided by a server. The client uses a network to access the service if the server is located on a different computer system, which happens frequently but not always.[19]

Software acquisition:

The Windows environment is compatible with the client-side application. Our program's database was made using XAMPP server, while the application portion was created using draw.io.

5.4 User Manual Preparation

Anyone familiar with computers and other social media platforms can easily use online reservations. Additionally, our system includes a user manual that users can easily read, comprehend, and apply to get the most out of the system.

5.5 User training

The intention of this testing section is to present how to use the online reservation for each user of the system. It shows some practices, such as registration, how the reservation is made and also how the payment is done.

Training methodology:

It is necessary to develop the appropriate training techniques for the various trainee groups in order for online reservation to be successfully implemented.

The following are the different groups targeted for training:

- System Admin
- Admin
- Cooker
- Cashier
- Waiter

This training program is used to tutor users about the new system and how to do their work properly on it. This training session will be held in AG restaurant.

System admin

The system admin training program includes an overview of the system as well as explanations of the system's objectives and functionality. This training program prepares system admin to effectively understand the system.

Admin

This training program is used to teach administrators about how to log into the system and control it.

Cooker

This training session is used to teach cookers how to check the menu made by the customers while making the reservation.

Cashier

This training segment is meant to teach Cashier how to check the payment made by the customer.

waiter

This training segment is meant to teach receptionist how to check reservations made by the customers.

5.6 Installation process

- Server-side installation
  - Pre required software's
    - ✓ MYSQL is an open-source relational database management system.
    - ✓ APACHE is open-source web server that delivers web content over the internet.
    - ✓ PHPMyAdmin is a free software tool intended to handle the administration of MySQL over the Web.
  - Configuration steps
    - ✓ Import a database file from the project directory.
    - ✓ Login via admin account (username: 'admin', Password: 'admin pass') Change the admin password and username.
- Client-side installation
  - Pre required software's
    - ✓ Chrome browser
    - ✓ Firefox browser
    - ✓ Edge browser

5.7 Start-up Strategy

The start-up strategy will be:
Buying a domain name
Hosting the system
Finally looking available Internet connection

CHAPTER SIX: CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

The aim of this project was to create a website that facilitates online reservations, thereby reducing waiting times and increasing productivity. The development process was divided into three stages: the first stage involved gathering requirements and doing analysis; the second stage dealt with design; and the third stage involved implementation.

The team began the first phase by interviewing managers and customers in order to determine the weaknesses of the current system. Object-oriented system analysis was used due to its greater flexibility. UML diagrams are designed in the second step, with the functional requirements serving as the basis. Using the Apache web server, MySQL database, and PHP back-end programming language, the system was put into operation in the third phase.

As a result of implementation, the project is accomplishing its goals. The objective of the project is achieved.

6.2 Recommendations

The system that we tried to develop has achieved its purpose and also met our objectives, and since the system is very good, we recommend AG Restaurant to use it.

In order to improve the system's usefulness and performance in the future, we seek to include the following:

- Develop better security mechanisms.
- Developing a mobile application for better accessibility.
- Full performance of the transaction.
- System users should be reading skills to view different services and can use different languages.

References

[1] https://www.library.unsw.edu.au/using-the-library/information-resources/primary-and-secondary-sources

[2] https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types

[3] https://docs.staruml.io/working-with-uml-diagrams/use-case-diagram

[4] https://en.m.wikipedia.org/wiki/Use_case_diagram

[5] https://creately.com/diagram/example/hk2rjxfl1/sequence-diagram-for-online-booking- system

[6] https://en.m.wikipedia.org/wiki/Feasibility_study

[7] https://en.m.wikipedia.org/wiki/Restaurant#

[8] https://www.researchgate.net/figure/The-class-type-architecture-is-a-variant-of-the-standard-layered-architecture-that_fig2_220156007

[9] https://support.ptc.com/help/modeler/r9.0/en/index.html#page/Integrity_Modeler/rtsme/class_modeling_overview.html

[10] https://online.visual-paradigm.com/diagrams/tutorials/deployment-diagram-tutorial/#:~:text=Deployment%20diagrams%20are%20one%20of,deployment%20view%20of%20an%20architecture.

[11] https://support.unicomsi.com/manuals/systemarchitect/114100/starthelp.html#page/Architecting_and_designing/UML1007445.html

[12] https://www.mindmanager.com/en/features/activity-diagram/#:~:text=An%20activity%20diagram%20is%20a,happen%20in%20the%20modeled%20system.

[13] https://www.researchgate.net/figure/Analysis-class-diagram-of-the-functionality-of-the-system_fig2_228998373#:~:text=This%20diagram%20gives%20a%20conceptual,help%20for%20understanding%20its%20architecture.

[14] https://www.vocabulary.com/dictionary/testing#:~:text=the%20act%20of%20conducting%20a,synonyms%3A%20examination

[15] https://aws.amazon.com/what-is/unit-testing/#:~:text=Unit%20testing%20is%20the%20process,test%20for%20each%20code%20unit.

[16] https://katalon.com/resources-center/blog/integration-

testing#:~:text=Integration%20testing%20is%20a%20type,when%20integrated%20with%20other%20components.

[17] https://testsigma.com/guides/acceptance-testing/

[18] https://www.techtarget.com/searchsoftwarequality/definition/system-testing#:~:text=System%20testing%2C%20also%20referred%20to,full%2C%20integrated%20system%20or%20application.

[19] https://en.m.wikipedia.org/wiki/Client_(computing)#:~:text=A%20client%20is%20a%20computer,be%20located%20on%20another%20computer).