Project Report

Cse327

Software Engineering

North South University

Name	ID
Md Abul Hashim Rafsan Khan	1812403042
Habibur Rahman Habib	1731050642
Abdul Aziz Chowdhury	1812164042

Project Name:

Online Food Ordering System

Github link: https://github.com/Rafsan25/online_food_order

Overview:

This is a user-friendly webpage for a restaurant where people can order food of their will through online. Here user/customer will register and login by giving required information. Necessary details of food items, order details, food related information will be stored in database. Customers can select items from food menu and place order according to their requirements. The administrator is the primary user of the website and he can manage total order procedure, keep track of food availability, total transaction, manage employees.

Features:

1. **User/Customer: -** A person will register via login id and password, fill up a form with necessary details/contacts, and then he or she would be able to place his or her order.

2. **Food items: -** Each food item will have a distinct food id, which will distinguish it from the food of some other restaurant.

3. **Order details: -** This will be various queries related to the food item. For example, quantity, date, added on, etc.

4. **Orders: -** These are the details obtained from the Customer and given to the person delivering the order. At every point in this process, the user/customer can cancel the order.

Functionality:

1. Admin Login/Registration.

2. Manage food items by admin. Add new food items, update existing items, view details of the item.

3. Manage food order. Add new order, update existing order, view details of the order.

4. Manage information of category. Add new items to category, delete items from category.

5. Select items from category. Add new items to cart.

6. Delete items from cart.

7. Manage delivery boy information by admin. Add/delete delivery boy information from admin dashboard.

Requirement Specification:

Functional Requirements:

1.Customer Registration and Login: Customers will register an account with required information and login into account with proper user id and password.

2.View Menu: All users of the system can view food menu.

3.Select Items: Customers will select their desired item from menu.

4.Show Order details: Customers will add their desired food items to cart and view full order details in the cart.

5.Make changes to Order: Customers can update cart before placing order. They can add or delete item from cart.

6.Cancel Order: Customers can cancel order at any time.

7.Add Items to Menu: Restaurant manager can add new items to menu.

8.Delete Items from Menu: Restaurant manager can delete items from menu.

9.Update Price: Restaurant manager can update price of food items at any time.

10.View Orders: Restaurant employees can view the order that has been placed and confirm order, then start preparing the food as well.

11.Select Payment Option: Customers will select payment method before placing order.

12.Manage Delivery Person: Restaurant manager can add or delete delivery person.

Non-functional Requirements:

1.User-friendly: This is a user-friendly system where customers can use this very easily.

2.Appearance: People can access this system at all time except when the server is down.

3.Privacy: Any kind of customer information shall not disclose to anyone.

4.Maintenance: A database is used to maintain all the data of the system.

5.Data Security: Any kind of data in the database shall not lose or disappear.

6.Performance Efficiency: The system is fast. Also, this system is efficient that it doesn't get hang due to the heavy traffic of user.

Testing:

Testing Methods:

Static Testing:

Verification is a static method of checking documents and files. Verification is the process, to ensure that whether we are building the product right, to verify the requirements which we have and to verify whether we are developing the product accordingly or not.
Activities involved here are Inspections, Reviews, Walkthroughs

Dynamic Testing:

It is also known as Validation in Software Testing.
Validation is a dynamic process of testing the real product. Validation is the process, whether we are building the right product i.e., to validate the product which we have developed is right or not.

1) Find problems early: Unit testing finds problems early in the development cycle. In test-driven development (TDD), which is frequently used in both extreme programming and scrum, unit tests are created before the code itself is written. When the tests pass, that code is considered complete. The same unit tests are run against that function frequently as the larger code base is developed either as the code is changed or via an automated process with the build. If the unit tests fail, it is considered to be a bug either in the changed code or the tests themselves. The unit tests then allow the location of the fault or failure to be easily traced. Since the unit tests alert the development team of the problem before handing the code off to testers or clients, it is still early in the development process.

2) Facilitates Change : Unit testing allows the programmer to refactor code or upgrade system libraries at a later date, and make sure the module still works correctly (e.g., in regression testing). The procedure is to write test cases for all functions and methods so that whenever a change causes a fault, it can be quickly identified. Unit tests detect changes which may break a design contract.

3) Simplifies Integration : Unit testing may reduce uncertainty in the units themselves and can be used in a bottom-up testing style approach. By testing the parts of a program first and then testing the sum of its parts, integration testing becomes much easier.

4) Documentation : Unit testing provides a sort of living documentation of the system. Developers looking to learn what functionality is provided by a unit, and how to use it, can look at the unit tests to gain a basic understanding of the unit's interface (API). Unit test cases embody characteristics that are critical to the success of the unit. These characteristics can indicate appropriate/inappropriate use of a unit as well as negative behaviors that are to be trapped by the unit. A unit test case, in and of itself, documents these critical characteristics, although many

software development environments do not rely solely upon code to document the product in development.

Unit Testing:

The steps involved during Unit testing are as follows:

1.Preparation of the test cases.

2.Preparation of the possible test data with all the validation checks.

3.Complete code review of the module.

4.Actual testing done manually.
.
5.Modifications done for the errors found during testing.

6.Prepared the test result scripts.

The unit testing done included the testing of the following items:

1.Functionality of the entire module/forms.

2.Validations for user input.

3.Checking of the Coding standards to be maintained during coding.

4.Testing the module with all the possible test data.

5.Testing of the functionality involving all type of calculations etc.

6.Commenting standard in the source files.

System Testing:

The System Testing done included the testing of the following items:

1.Functionality of the entire system as a whole.

2.User Interface of the system.

3.Testing the dependent modules together with all the possible test data scripts.

4.Verification and Validation testing.

5.Testing the reports with all its functionality.

Technologies:

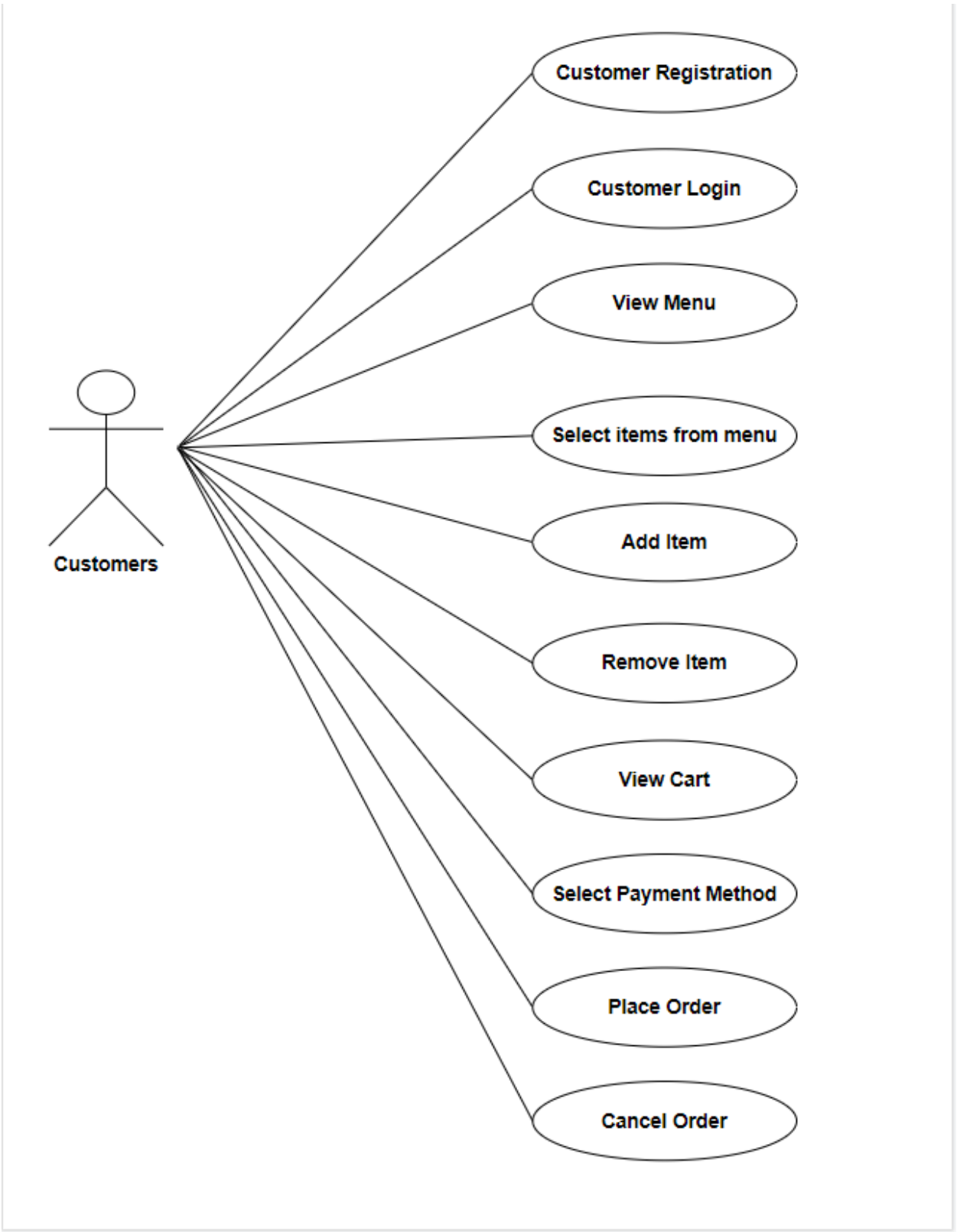
1. Html
2. CSS
3. JavaScript
4. Laravel Framework (PHP)
5. MySQL Database

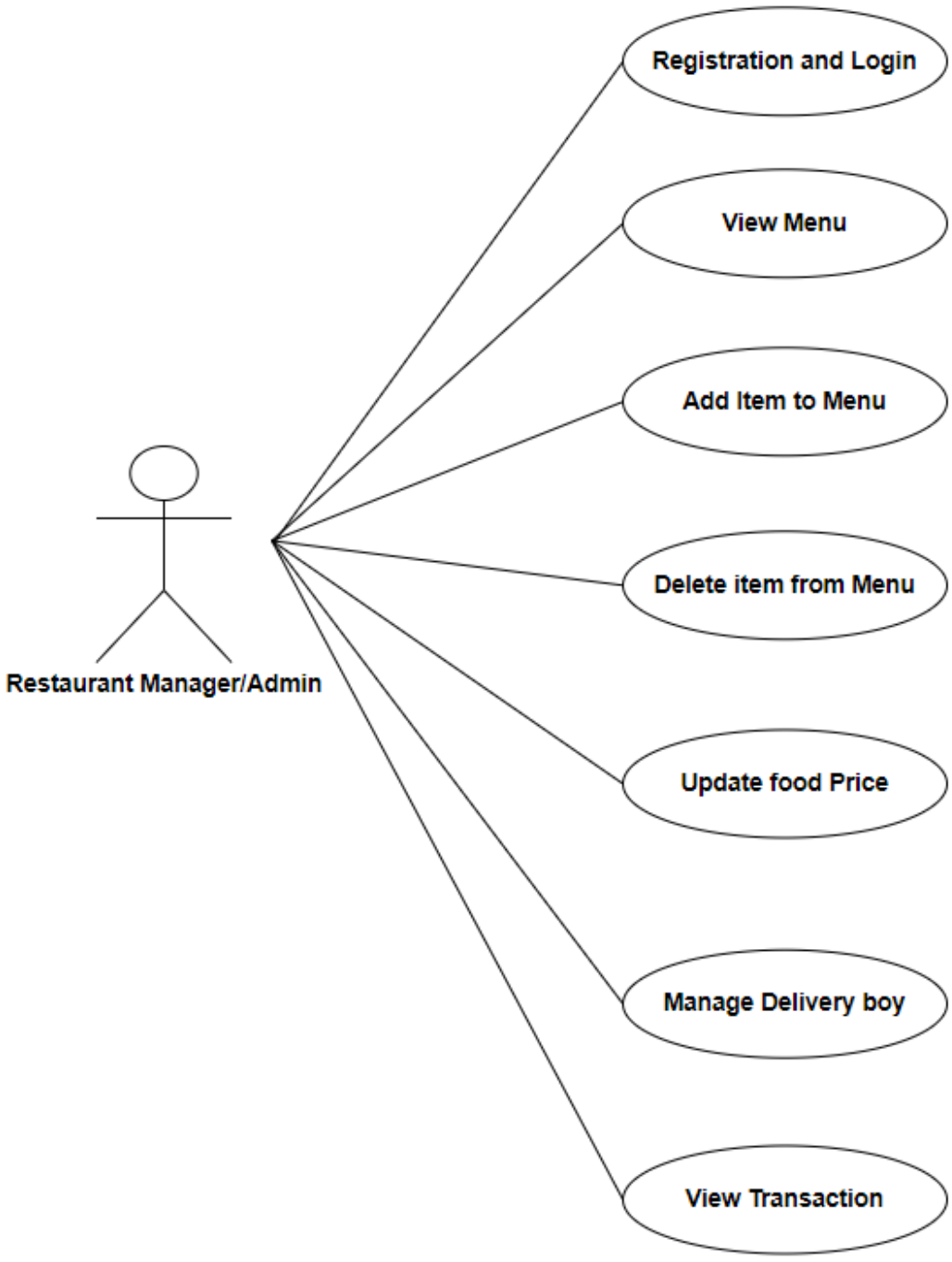
Diagrams:

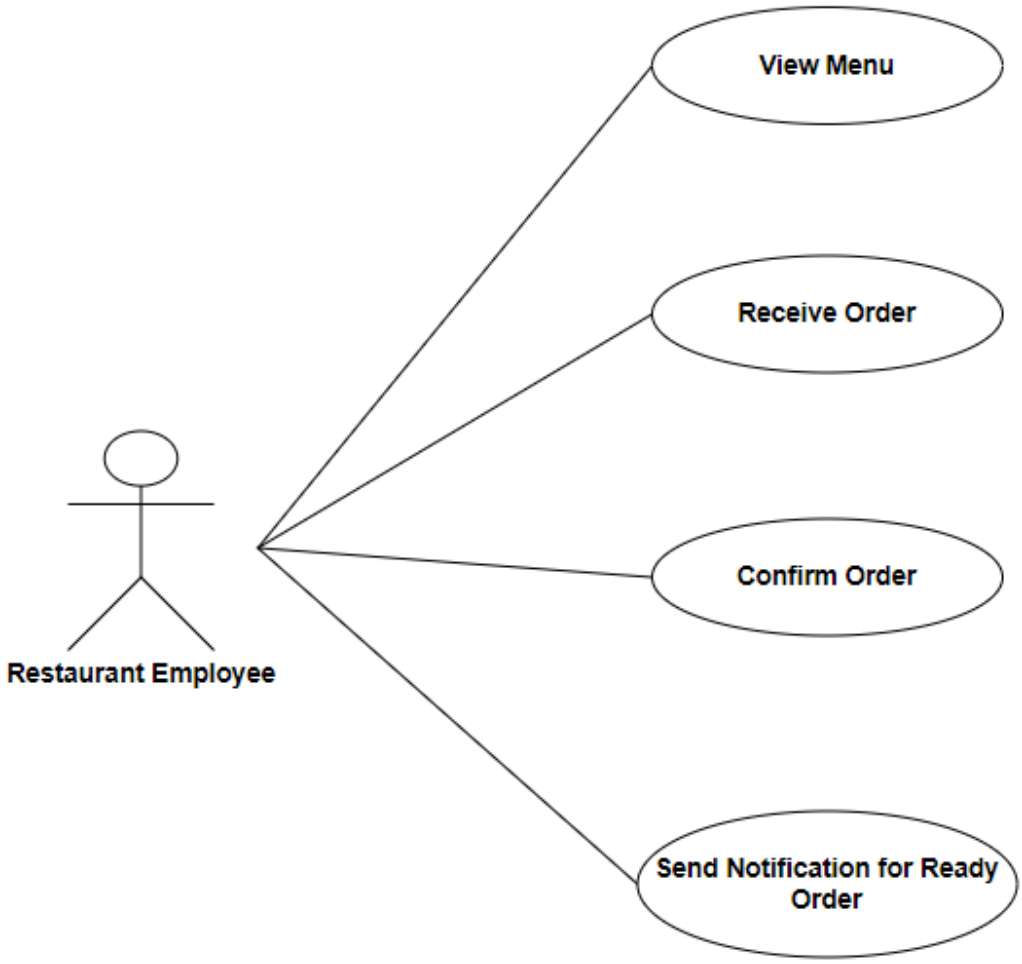
Use Case Diagram

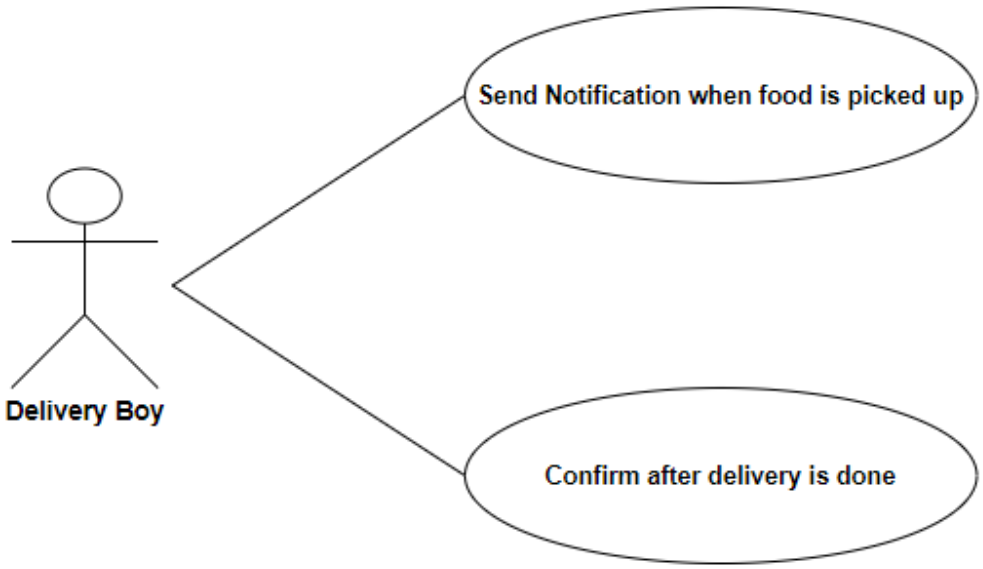
The Stakeholders for “Online Food Order System” project:

1. Customers.
2. Restaurant Manager.
3. Restaurant Employee.
4. Delivery Boy.

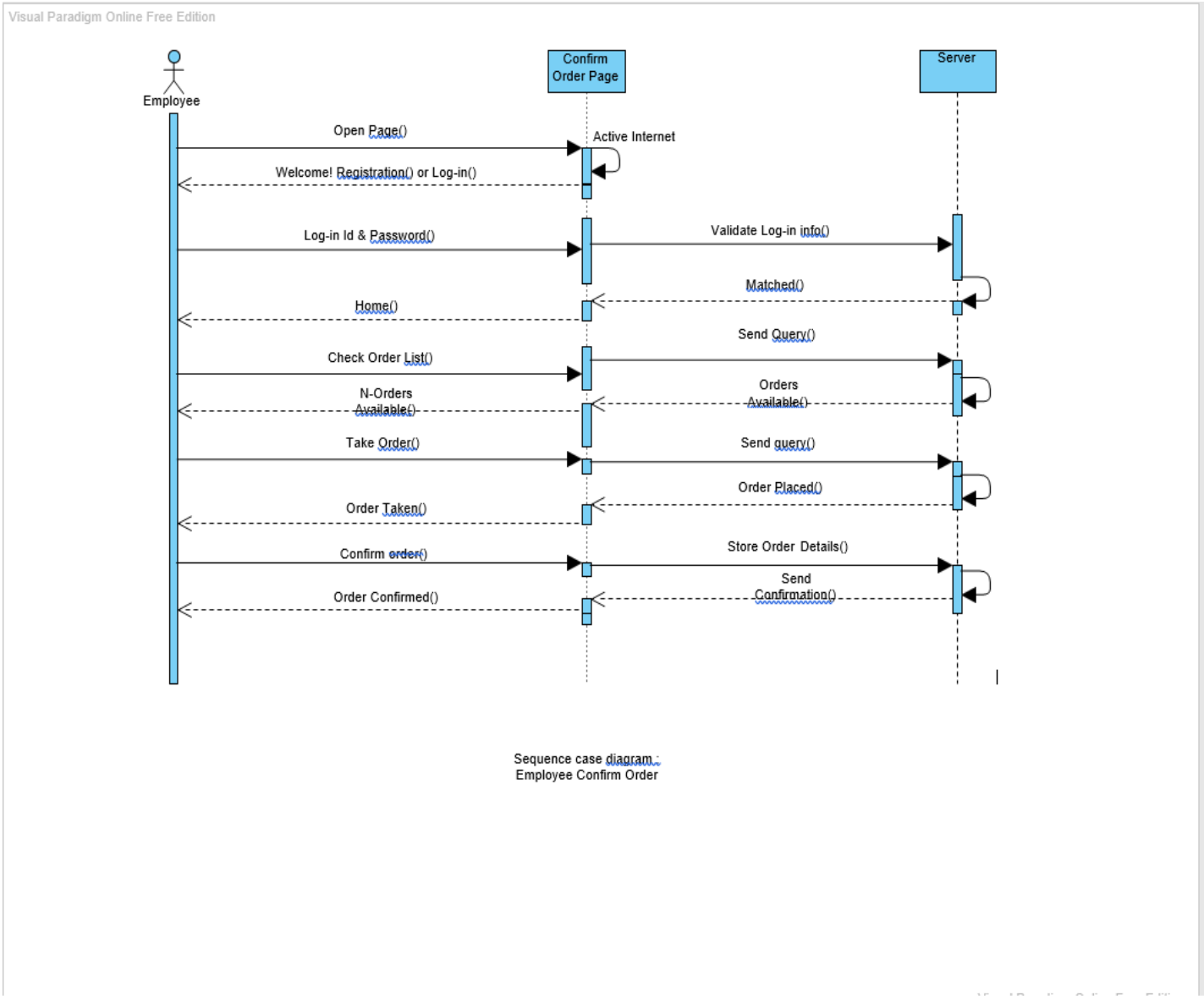

Customer Registration
Customer Login
View Menu
Select items from menu
Add Item
Remove Item
View Cart
Select Payment Method
Place Order
Cancel Order
Customers


Registration and Login
View Menu
Add Item to Menu
Delete item from Menu
Update food Price
Manage Delivery boy
View Transaction
Restaurant Manager/Admin

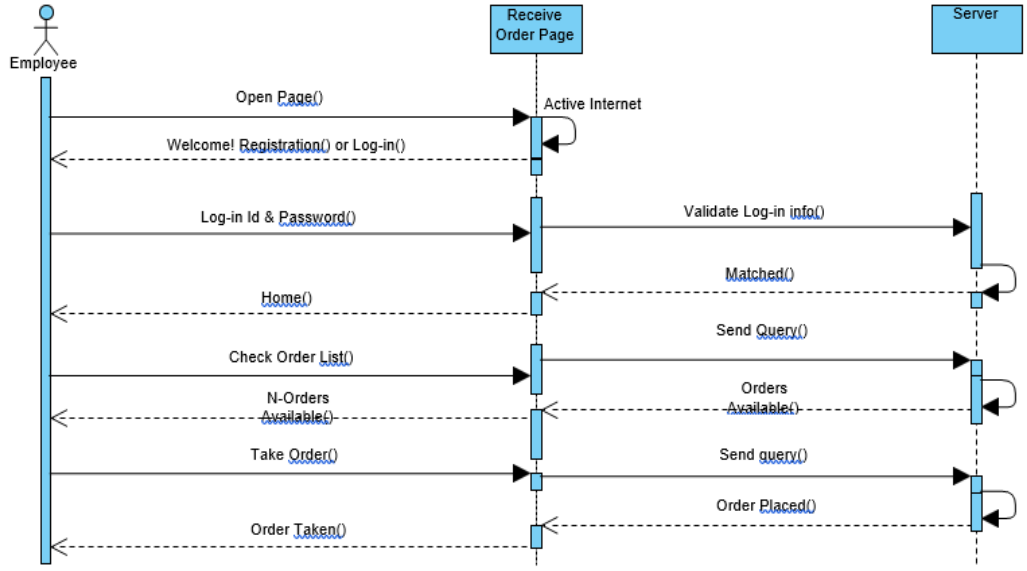

View Menu
Receive Order
Confirm Order
Send Notification for Ready Order
Restaurant Employee


Send Notification when food is picked up
Delivery Boy
Confirm after delivery is done

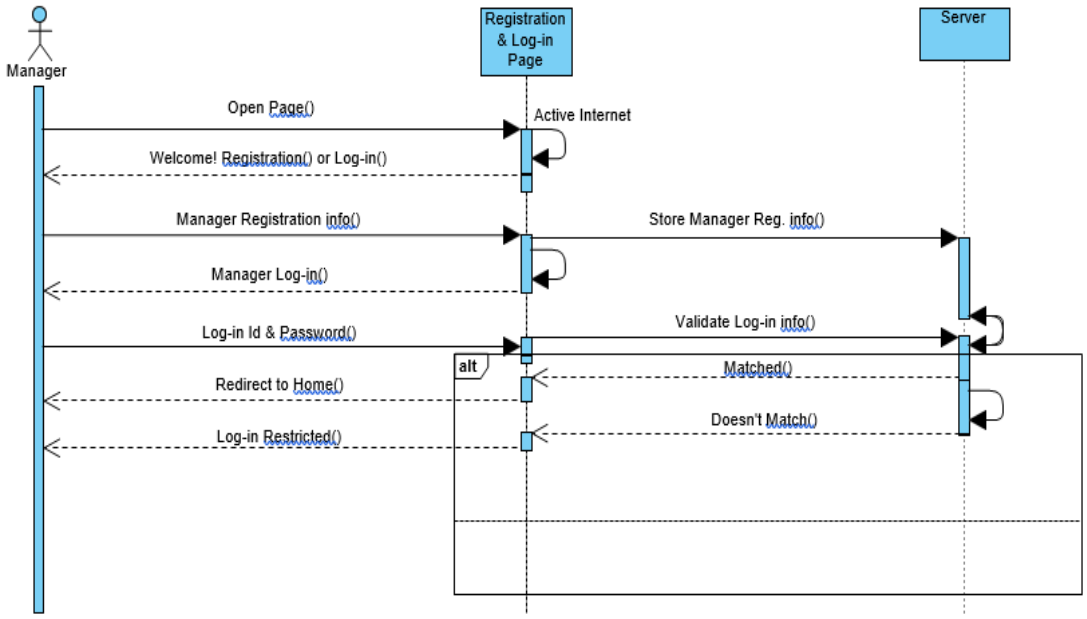
Sequence Diagram



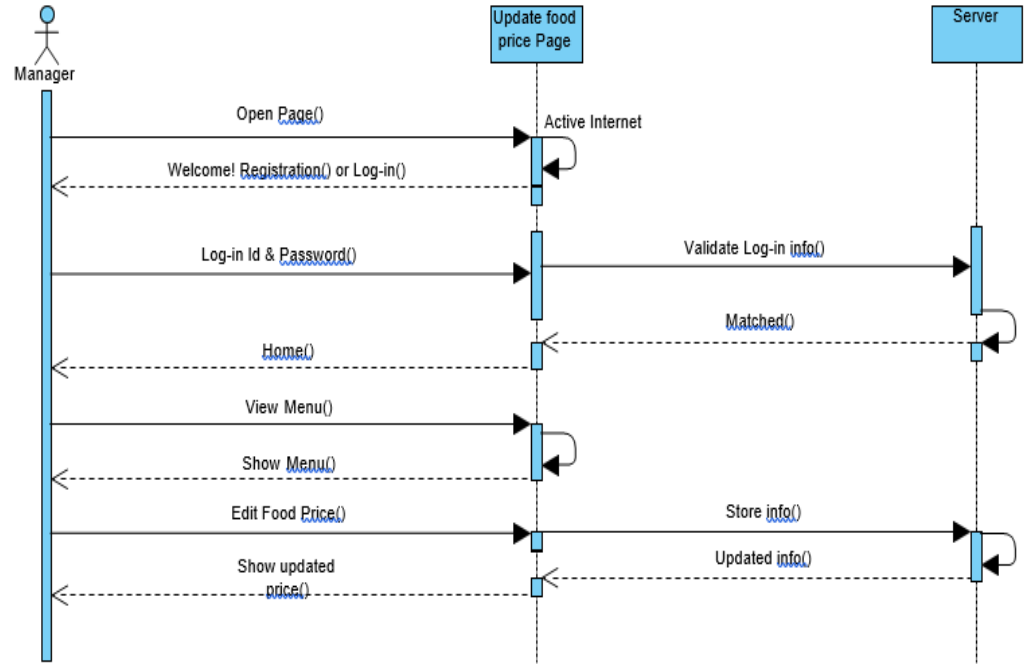
Sequence case diagram :
Employee Confirm Order



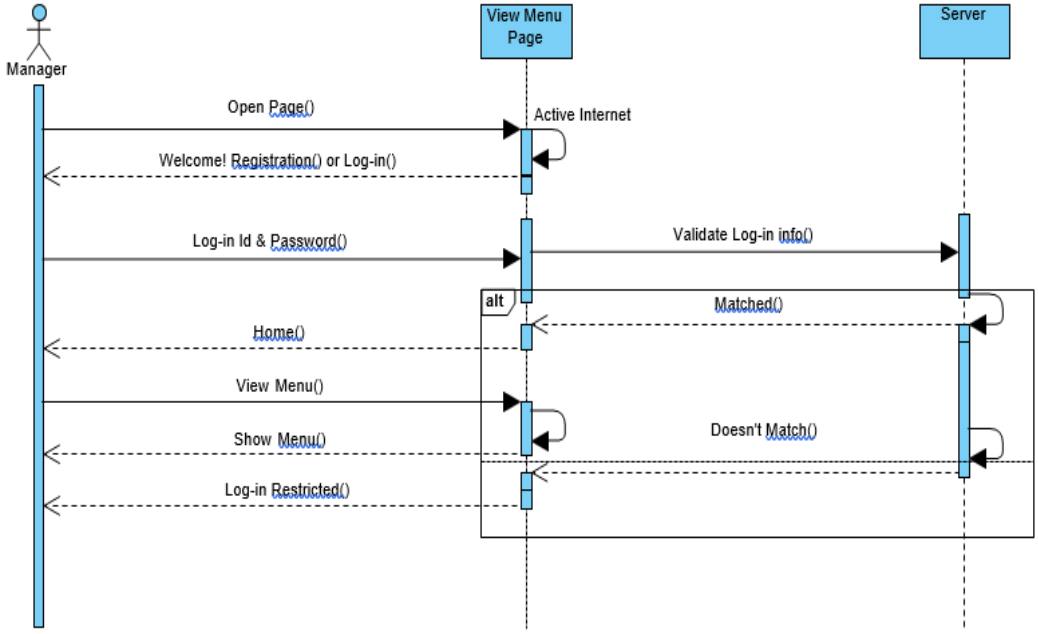
Sequence case diagram :
Employee Receive Order



Sequence case diagram :
Manager Registration & Log-in

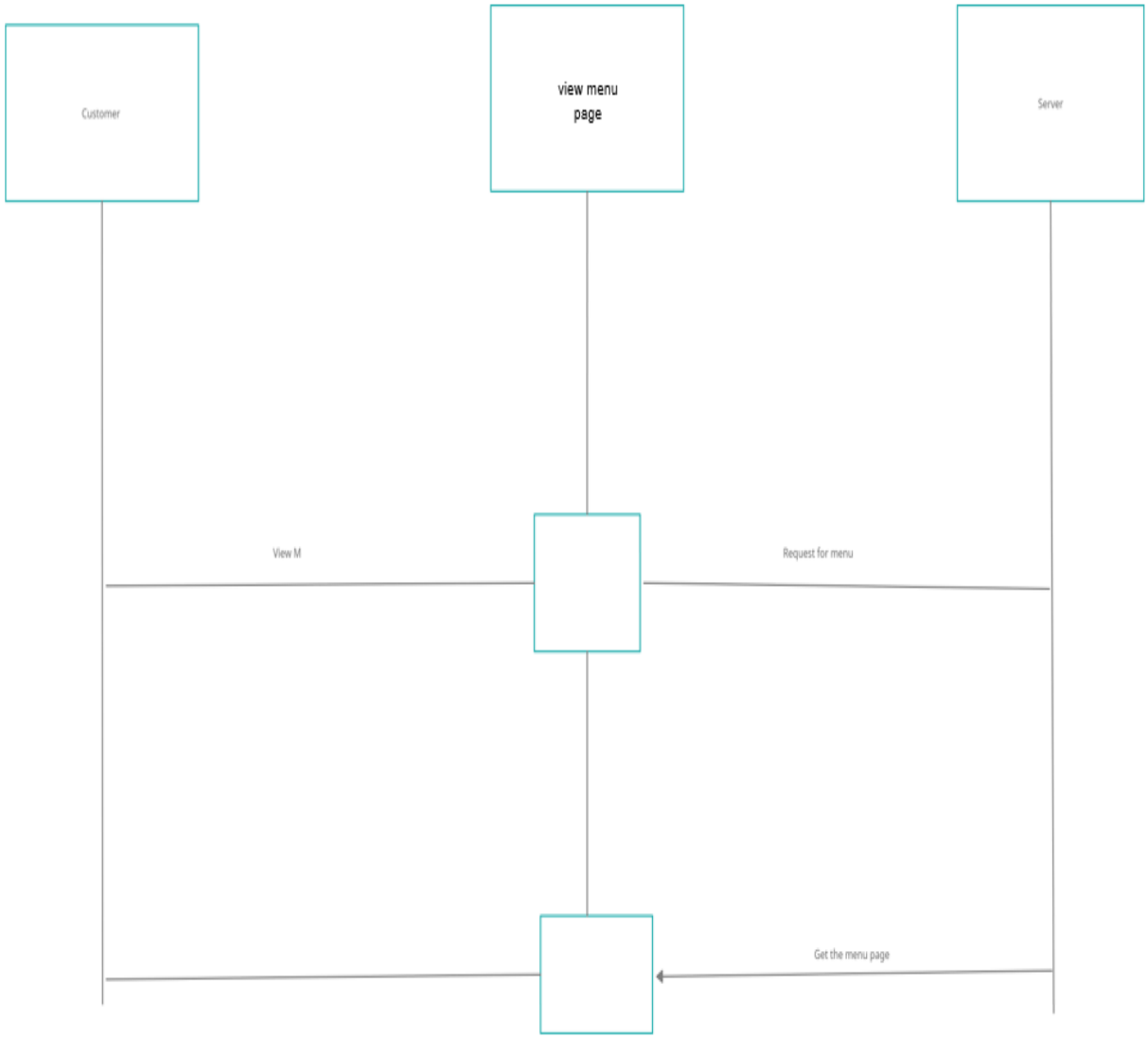


Sequence case diagram :
Manager-Update Food price

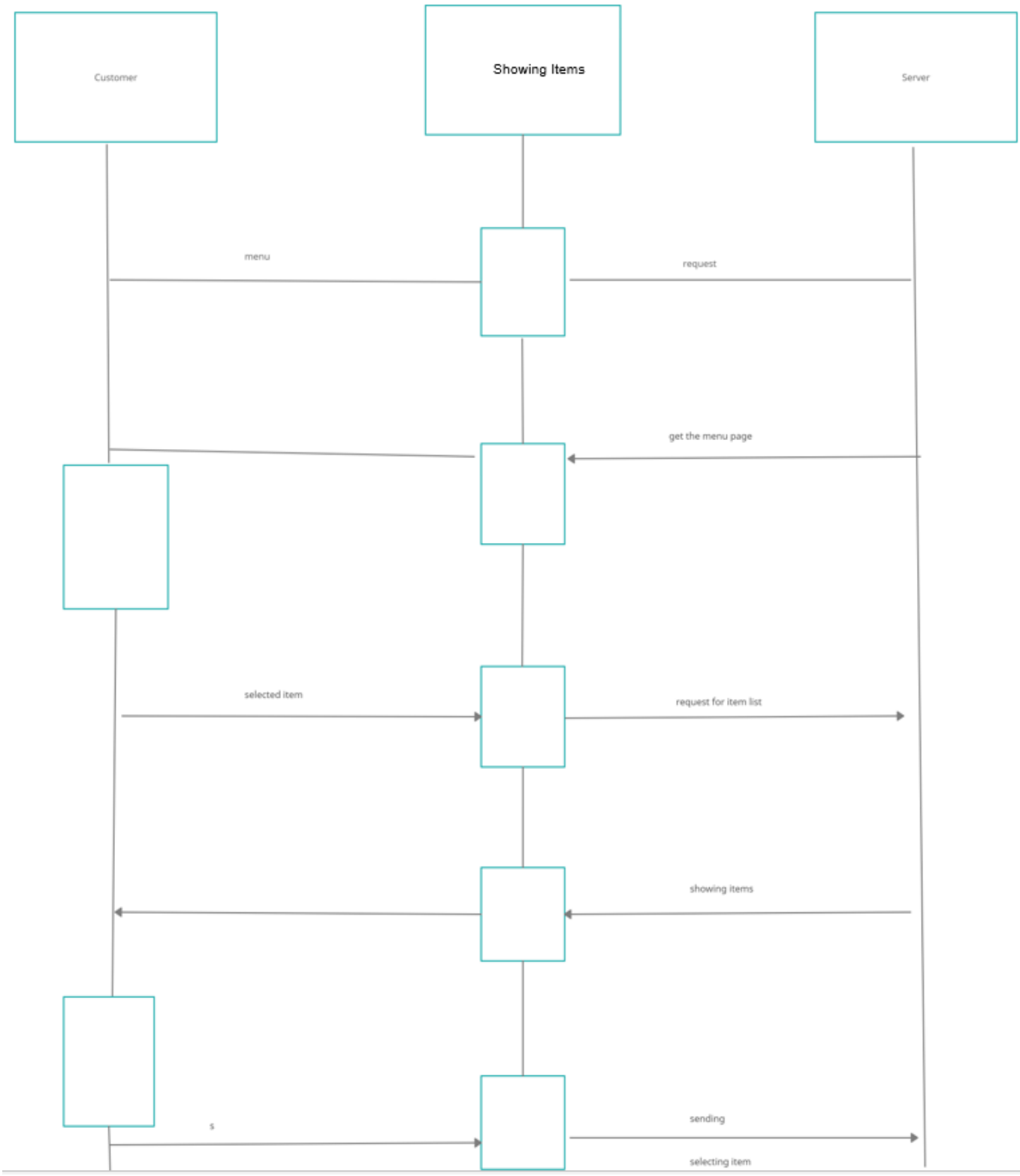


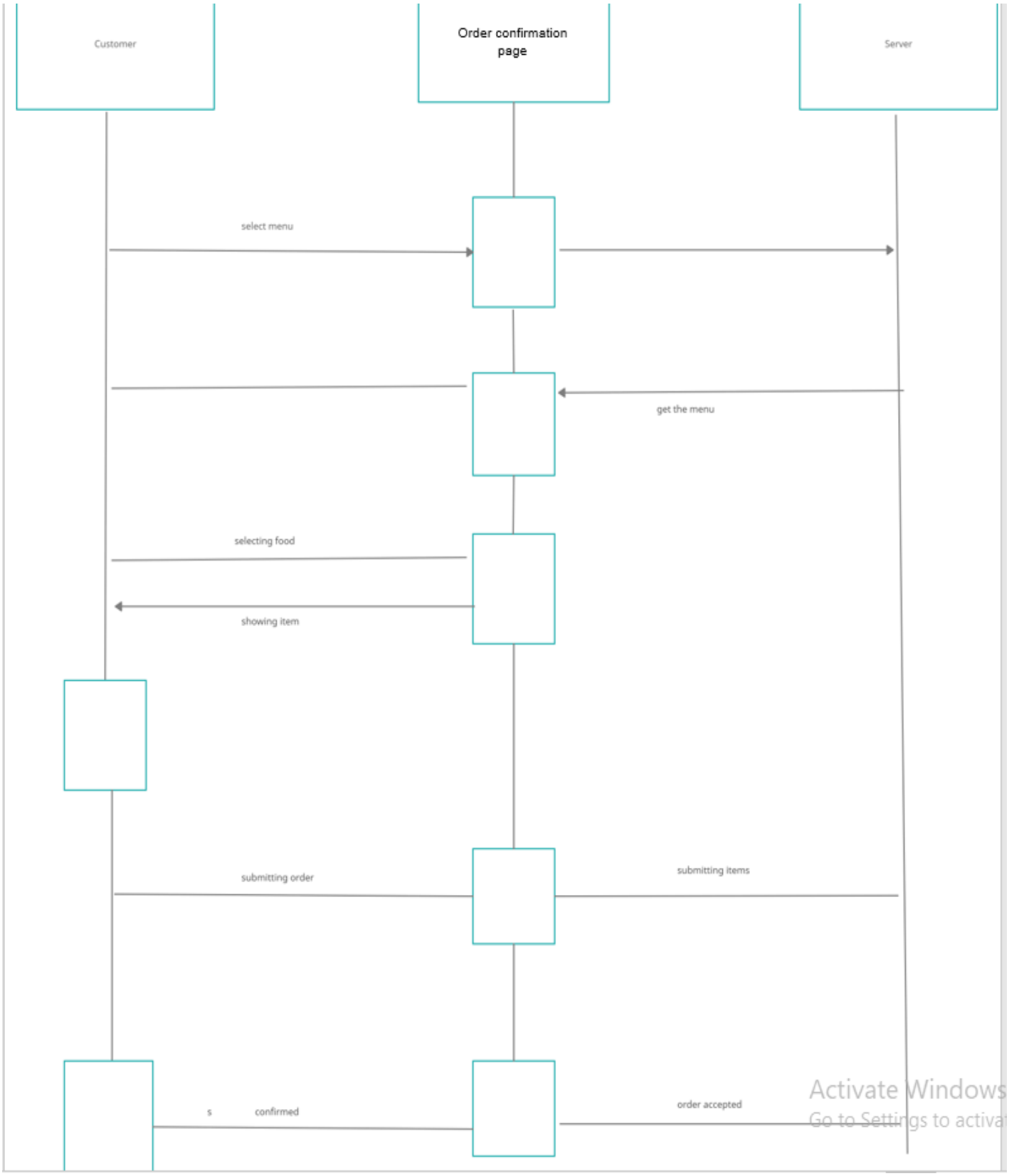
Sequence case diagram :
Manager View Menu

After log in



Sequence diagram for view menu


Customer
Showing Items
Server
menu
request
get the menu page
selected item
request for item list
showing items
s
sending
selecting item



Class Diagram