

---

**Design Report**  
**for**  
**Kitchen Zealot**

**Written by:**  
**Abtahi Chowdhury**  
**Abusaleh Masud**  
**Safwan Shahid**  
**Arman Uddin**  
**Farhan Zaman**

**Version 2.0**

**November 23, 2019**

---

<b>Date</b>	<b>Version</b>	<b>Description</b>	<b>Author</b>
11/2/2019	1.0.0	Initial version of the online restaurant system	Abtahi Chowdhury Abusaleh Masud Safwan Shahid Arman Uddin Farhan Zaman
11/23/2019	2.0.0	This report is meant to provide the data structure and logic to carry out the functionalities dictated by the specification.	Abtahi Chowdhury Abusaleh Masud Safwan Shahid Arman Uddin Farhan Zaman

# Table of Contents

<b>Introduction</b>	<b>5</b>
<b>Design</b>	<b>6</b>
2.1 Use Case Scenarios	6
2.2 Collaboration Diagrams	11
2.3 State Diagrams/ Petri-net	21
<b>E/R Diagram</b>	<b>31</b>
<b>Pseudocode</b>	<b>31</b>
addNewEmployee(formValue)	31
login(email:string, password:string)	32
register(email:string, password:string)	33
logout()	33
guestLogin()	33
addToCart(product:Product)	34
removeFromCart(product:Product)	34
getCustomer()	35
getUser()	35
updateCart(customer:Customer)	35
addToGuestCart(product:Product)	36
removeFromGuestCart(product:Product)	36
addCustomer(customer:Customer)	37
removeCustomer(customer:Customer)	37
updateCustomer(uid:string, customer:Customer)	38
getCustomer(uid:string) : Observable<Customer>	38
getUser(uid:string) : Observable<User>	38
getCurrentUser()	38
getCurrentCustomer()	39
addEmployee(employee:Employee)	39
removeEmployee(employee:Employee)	39
getEmployee(uid:string): Observable<Employee>	39
getUser(uid:string) : Observable<User>	40

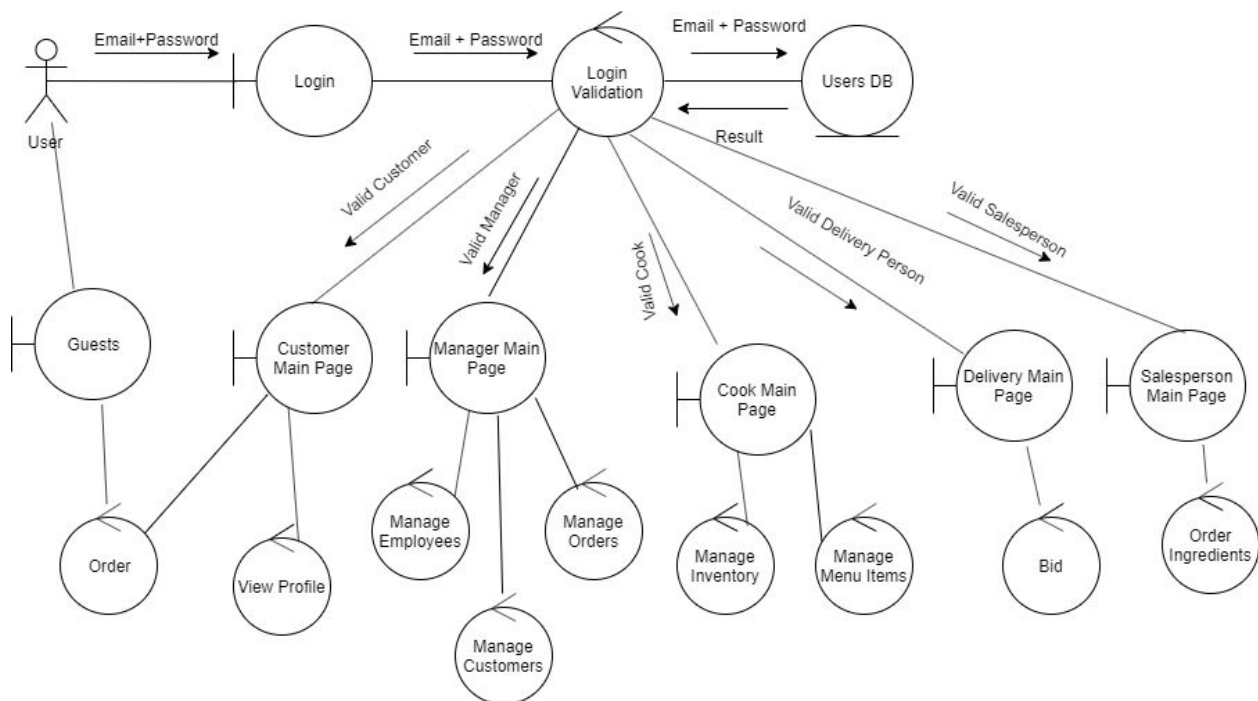
addGuest(guest:Guest)	40
removeGuest(guest:Guest)	40
getGuest(uid:string) : Observable<Guest>	41
addOrder(order:Order)	41
removeOrder(order:Order)	41
updateOrder(uid:string, order:Order)	41
getOrder(uid:string): Observable<Order>	42
create(product:Product)	42
lookUp(uid:string): Observable<Product>	42
update(uid:string, product:Product)	43
delete(uid:string)	43
addUser(user:User)	43
removeUser(user:User)	43
getUser(uid:string): Observable<User>	44
<b>System Screen</b>	<b>44</b>
<b>Minutes</b>	<b>47</b>
<b>Git Repo:</b>	<b>47</b>

# 1.Introduction

Below you will encounter many diagrams, charts, and pseudocode, as they will assist in breaking down the design of our product. As you continue reading the paper, any questions or confusions that arise will be answered and clarified.

Not only does it allow customers (registered users) and guests (unregistered users) to order and receive food, but also gives managers, salespeople, cooks, and delivery people access to their own page to handle services in the company. Delivery people have access to see all the different orders from customers and guests, and bid on them. Salespeople are given comments from cooks, to know and order ingredients that are needed. Cooks are allowed to request more supplies from salespeople, rate salespeople, and change menu items. Managers can approve guest to customers, view order history, view all ratings, start delivery bidding process per order, pay employees, hire/fire employees, and remove warnings.

The collaboration class diagram below gives an overview on the entire Kitchen Zealot system.



DB: Database

## 2.Design

### 2.1 Use Case Scenarios

<b>View Menu</b>
<p><b>Normal Scenario</b></p> <ol style="list-style-type: none"><li>1. Guests, or customers click “Menu” on the homepage to see menu. Customers or Guests who are logged in to the site, can click “Order” to view the menu to order.</li></ol>

<b>Login</b>
<p><b>Normal Scenario</b></p> <ol style="list-style-type: none"><li>1. User clicks “Login” button in the navigation bar</li><li>2. User is asked to enter their email and password</li><li>3. User inputs appropriate information</li><li>4. Information is checked against database for authenticity</li><li>5. User is logged in</li></ol>
<p><b>Exceptional Scenario</b></p> <ol style="list-style-type: none"><li>1. User clicks login and enters credentials</li><li>2. Credentials are not valid</li><li>3. Error message is shown</li><li>4. User is asked to try again</li></ol>

<b>Continue as Guest</b>
<p><b>Normal Scenario</b></p> <ol style="list-style-type: none"><li>1. User clicks on “Login” and then “Continue as Guest”</li><li>2. User inputs their phone number and address when ordering</li></ol>
<p><b>Exceptional Scenario</b></p> <ol style="list-style-type: none"><li>1. User clicks on “Login” and then “Continue as Guest”</li><li>2. User inputs invalid phone number and/or address</li><li>3. Input boxes will highlight red to indicate an error</li></ol>

## Register

### Normal Scenario

1. Guest clicks login
2. Guest clicks "Sign Up"
3. Guest is asked to enter "Name", "Email", "Password", "Phone Number"
4. Guest clicks "Register"
5. A request is sent to the manager to approve
6. Manager approves or disapproves new
7. If approved, then guest info is sent to the database to be added

### Exceptional Scenario

1. Guest clicks login
2. Guest clicks "Sign Up"
3. Guest is asked to enter "Name", "Email", "Password", "Phone Number"
4. Guest enters an email already in use
5. Guest clicks "Register"
6. An error message pops up stating the email is already used

## Add/Remove Products to Cart

### Normal Scenario

1. Customer clicks on order
2. Customer adds an item to the cart with the quantity they want

## Order Food

### Normal Scenario

#### Scenario 1: Customer

1. Customer clicks "Order"
2. Customer views menu and clicks "Add to Cart"
3. Customer adjusts quantity
4. Customer click "Check Out"

#### Scenario 2: Guest

1. Guest clicks "Menu"
2. Guest adds an item to the cart with the quantity they want
3. Guest clicks "Check Out" to place order
4. Guest is prompted to enter their details: Name, Email, Phone Number, Payment Details

### **View All Placed Orders**

#### **Normal Scenario**

1. Customer clicks on “My Orders” to view all placed orders

### **Rate Food/Delivery**

#### **Normal Scenario**

1. Customers can rate the food(cooks) and the delivery people individually
2. Customer Views order
3. Rate Food
4. Rate Delivery

### **Bid on Deliveries**

#### **Normal Scenario**

1. Delivery person can access the pending bid page via a button on the post-login page
2. Delivery person chooses from a list of pending bids
3. Delivery person places bids on that selected order
4. After 15 minutes time runs out, bid is over

#### **Exceptional Scenario**

1. Two users input the same bid at the same time

### **Give Customers Rating**

#### **Normal Scenario**

1. Delivery person can access the past delivered ordered in the “Delivered Orders” tab
2. Delivery person can view orders by clicking “View” next to the respective order
3. Delivery person can give customers ratings from 1 to 5 and an optional comment

#### **Exceptional Scenario**

1. Can't give a rating if they view the rating they received from said customer



## **View Delivered Orders**

### **Normal Scenario**

1. Delivery person can access the past delivered ordered in the “Delivered Orders” tab

## **Modify Menu Items**

### **Normal Scenario**

1. Cooks can access the “Modify Menu Page” by clicking a button in the navbar
2. Cook choose a menu item they wish to change or add a new menu item

### **Exceptional Scenario**

1. Two cooks may try to delete the same item at the same time

## **Request Supplies**

### **Normal Scenario**

1. Cooks can click a “Request Supplies” button on the navigation bar
2. On the Request Supplies Page, the cook can order the desired supplies by clicking “Order”

## **Rate Supplies**

### **Normal Scenario**

1. Cooks can click “Rate Past Orders’
2. Cooks can click “View” on a past order
3. Cooks can rate from 1 to 5, and write an optional comment

## **Purchase Supplies**

### **Normal Scenario**

1. Salesperson click on “Purchase Supplies”
2. Salesperson can open a list of items and select which items and in what quantity to purchase

## **Start Bids**

### **Normal Scenario**

1. Managers can click “Manage Orders” to view all orders
2. Managers can then click an order to view order details and start bids if appropriate

### **Exceptional Scenario**

1. No current orders

## **Manage Employees**

### **Normal Scenario**

1. Managers can click the “Employees” tab
2. Managers can click “View” next to any Employee to view employee details
3. Managers can hire new employees by manually adding the required information

### **Exceptional Scenario**

1. May go below minimum number of employees

## **Approve Registration**

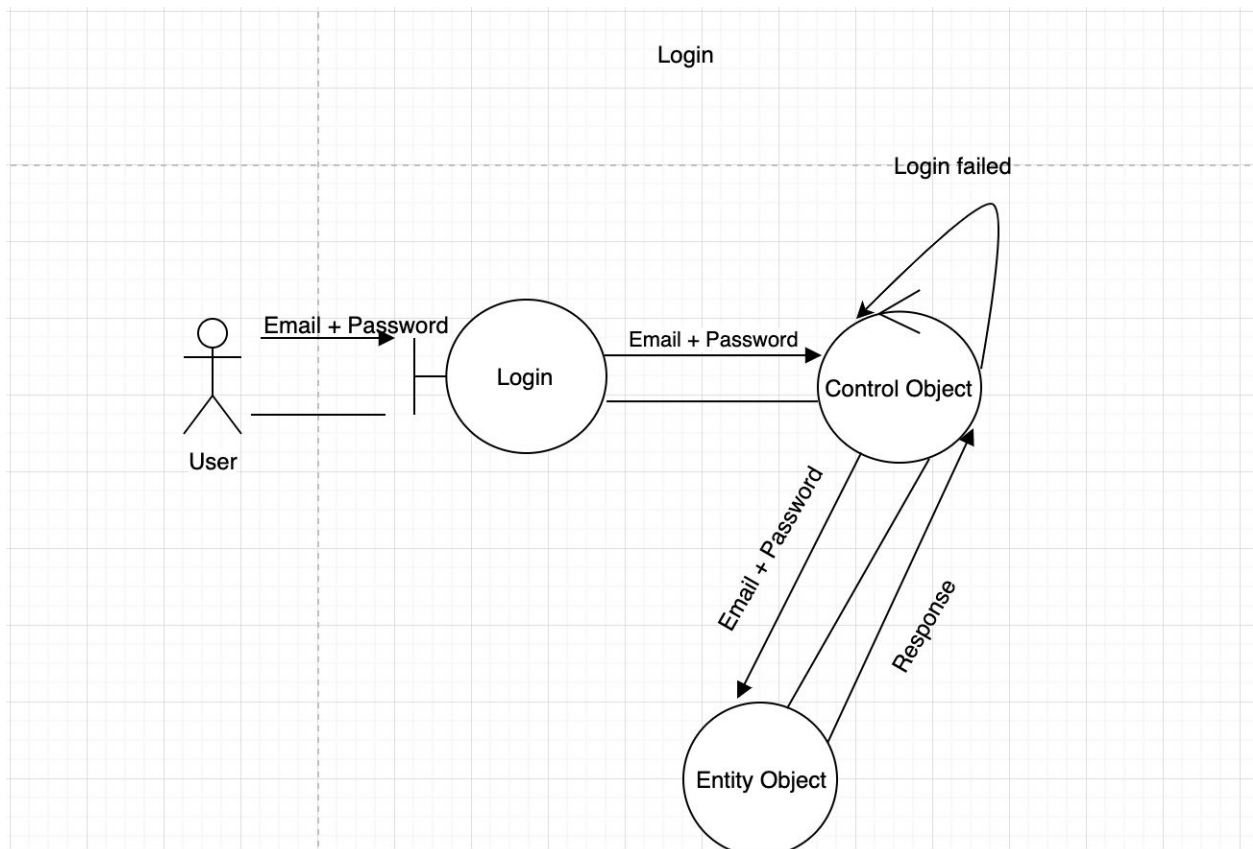
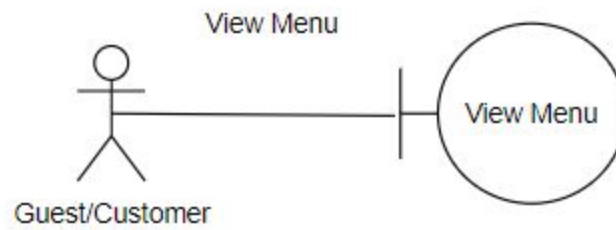
### **Normal Scenario**

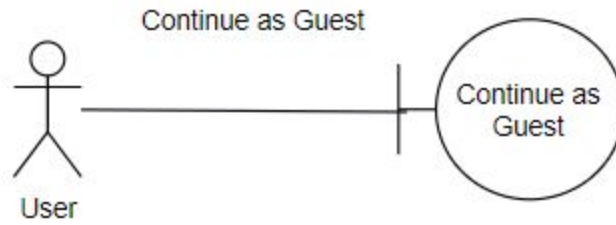
1. Manager clicks “Pending Registrations”
2. Manager can click “View” next to applicants name
3. Manager clicks “Yes” / “No” next to each new guest

### **Exceptional Scenario**

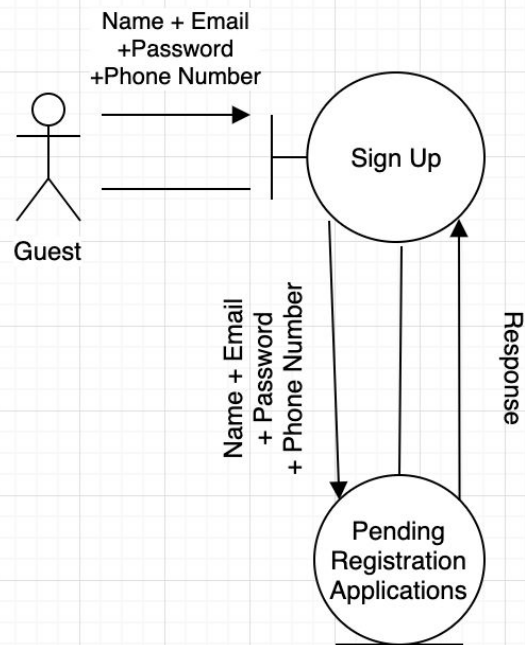
1. Guest may not have order history

## 2.2 Collaboration Diagrams

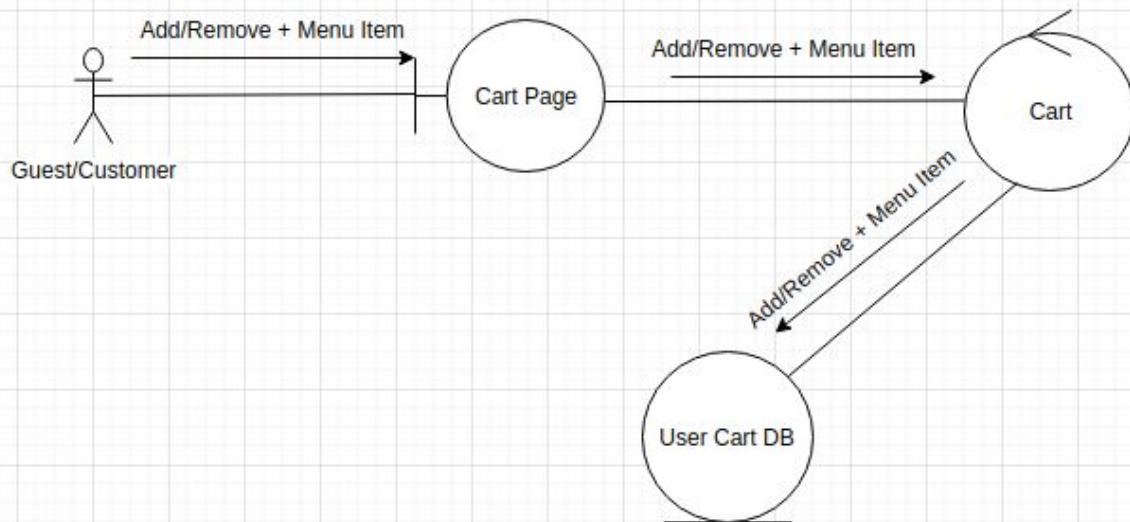




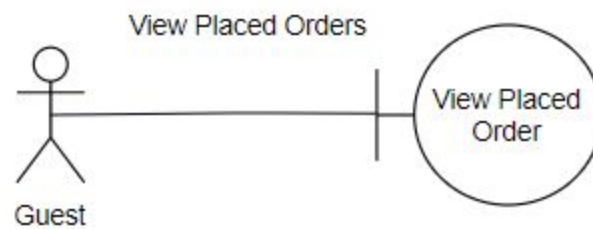
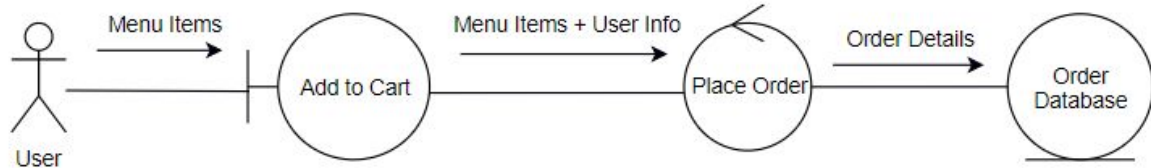
## Register



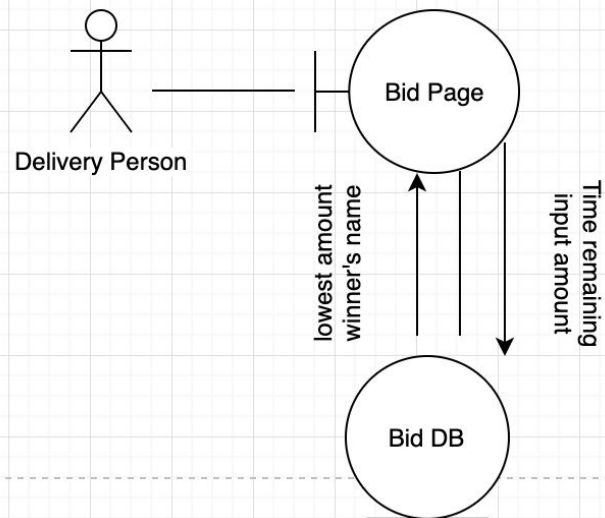
# Add/Remove Item



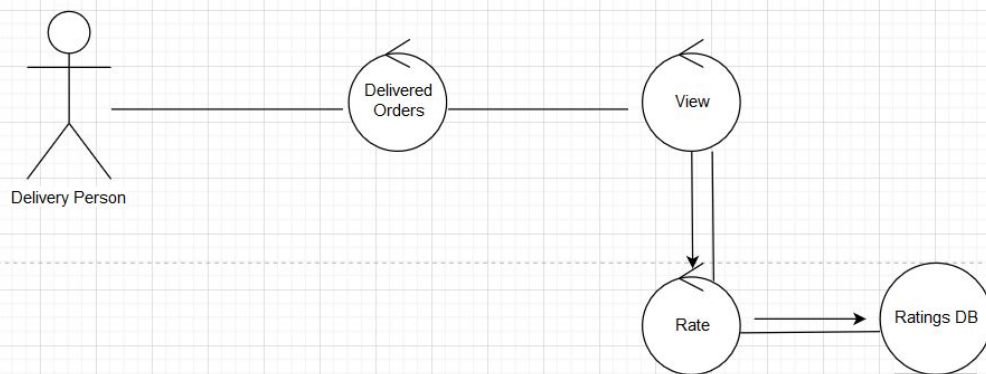
## ORDER



## Bid on Deliveries



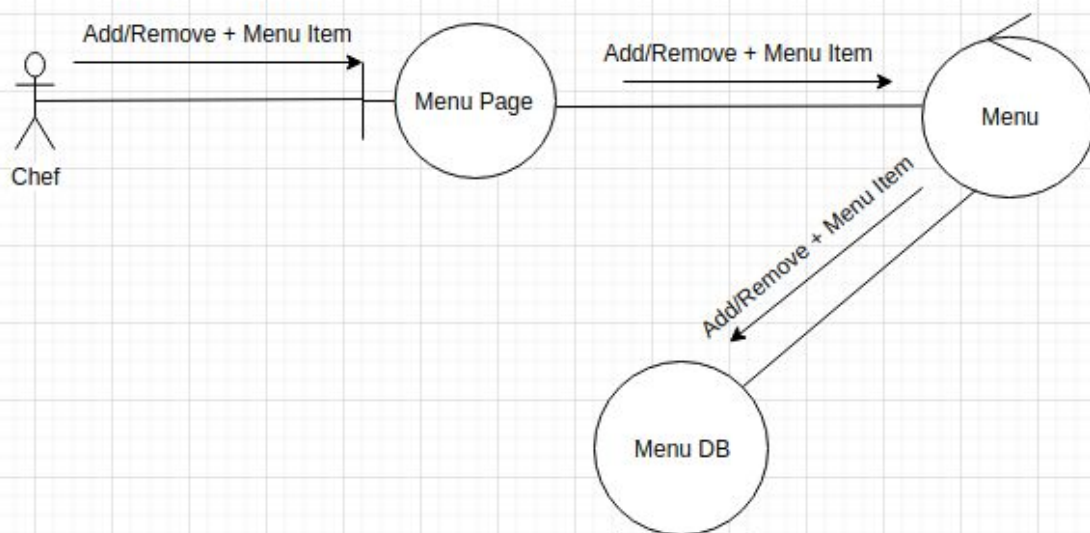
## Give Customer Ratings



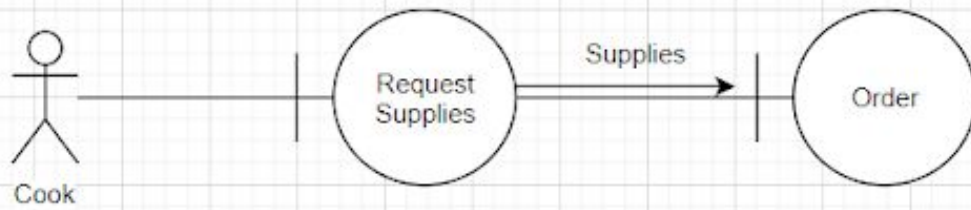
## View Delivered Orders



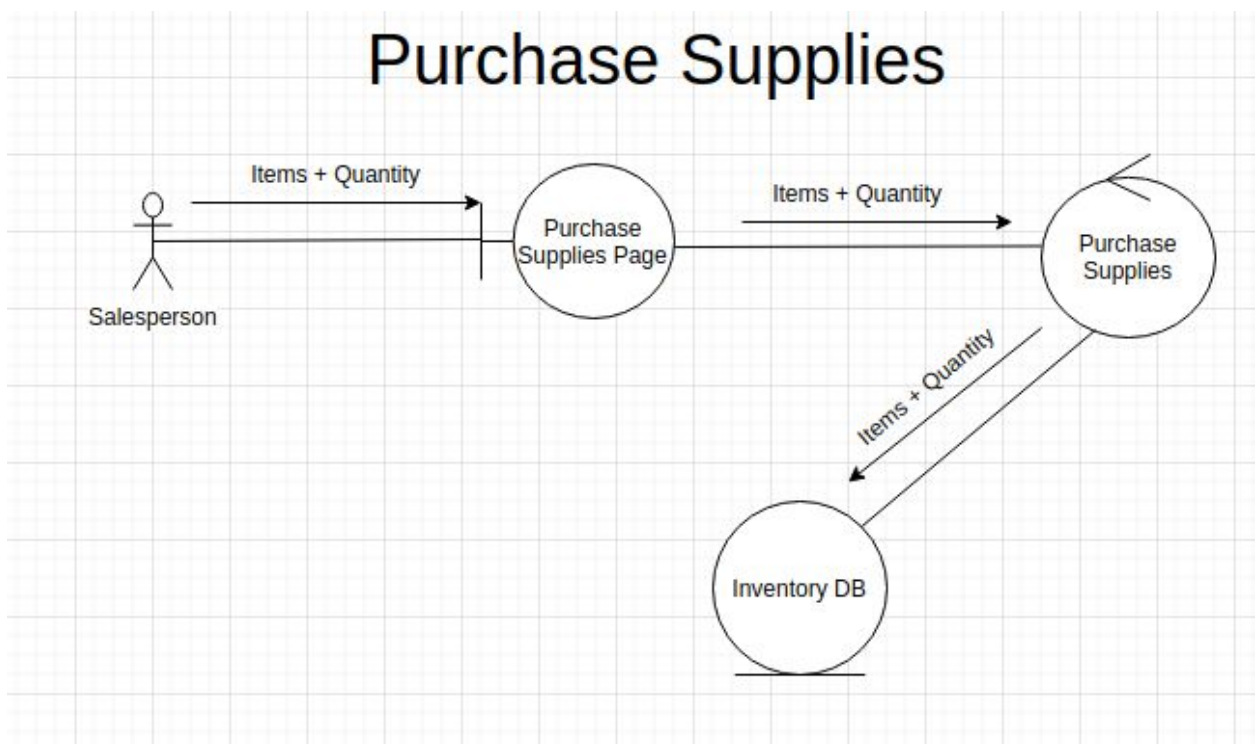
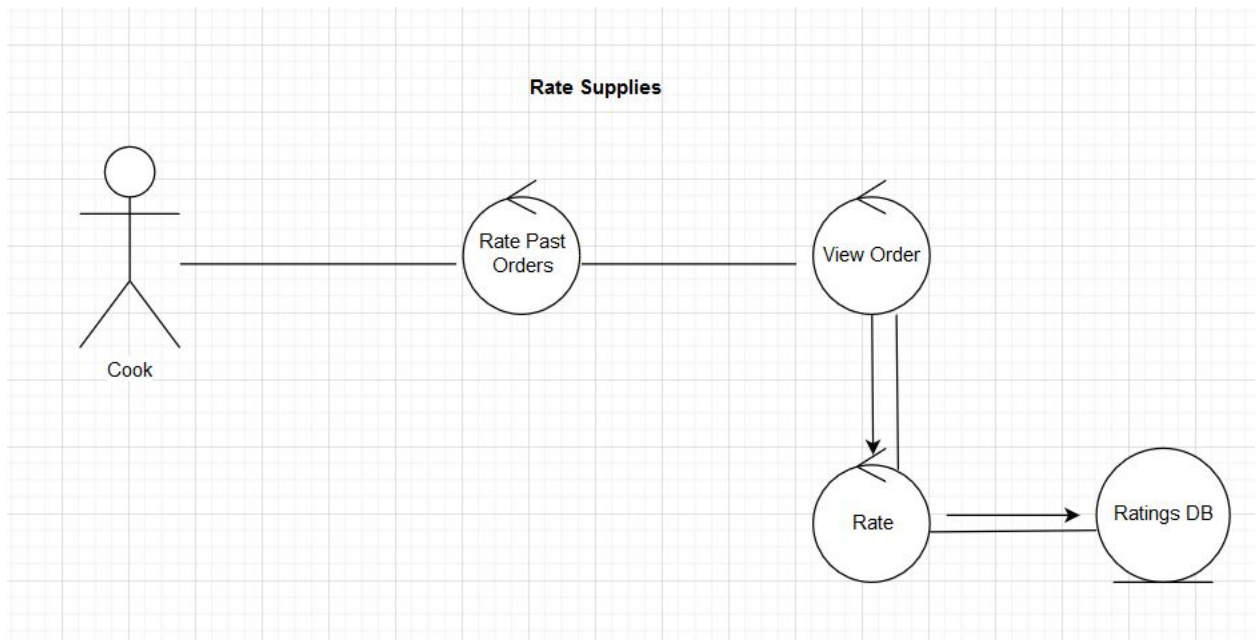
## Modify Menu Items



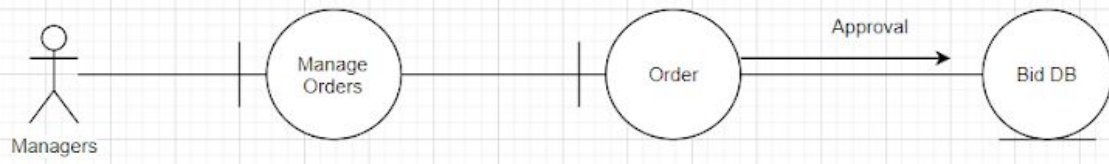
## Request Supplies



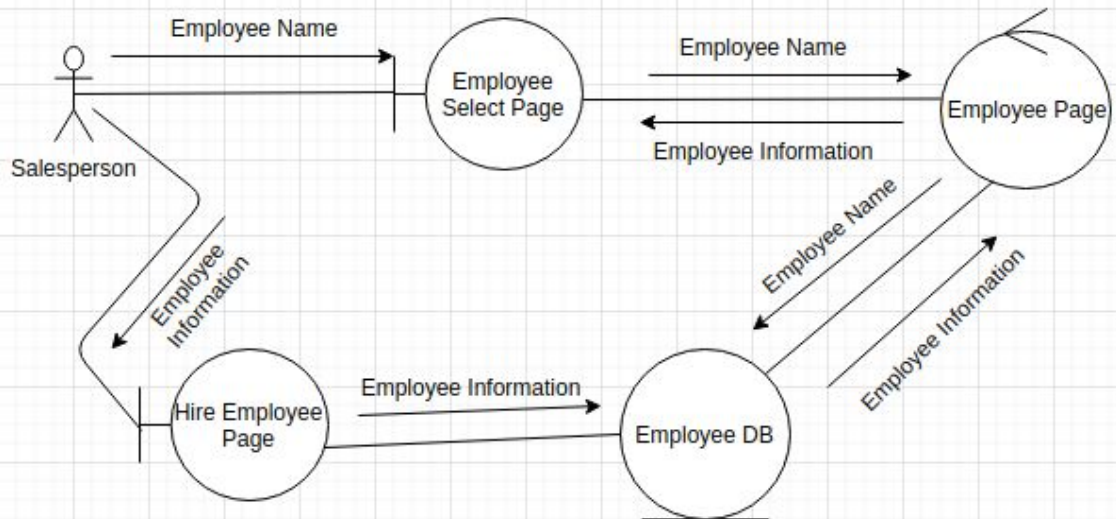




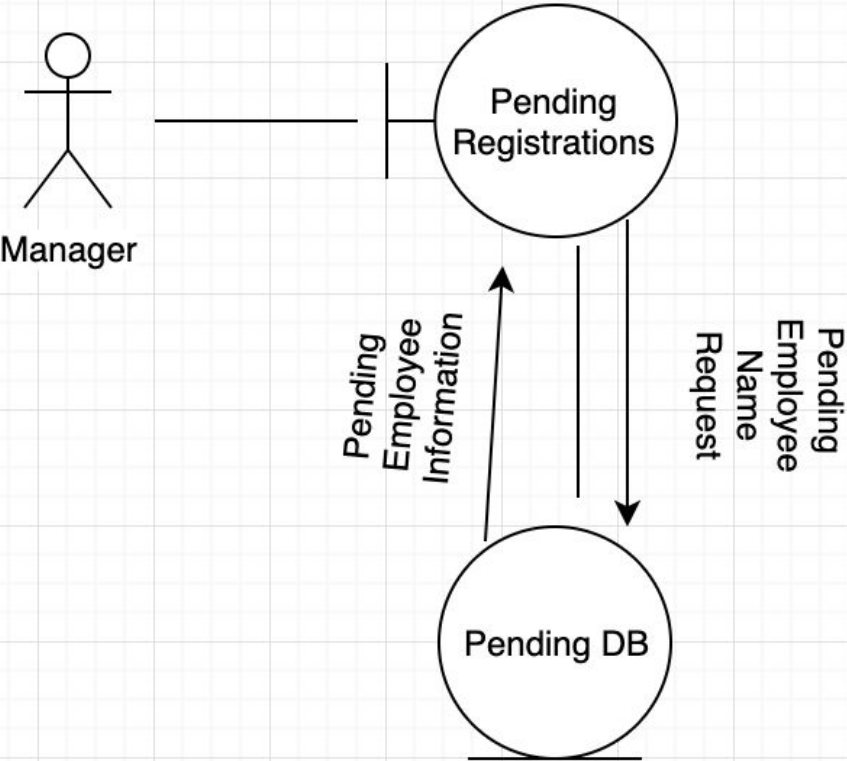
## Start Bids



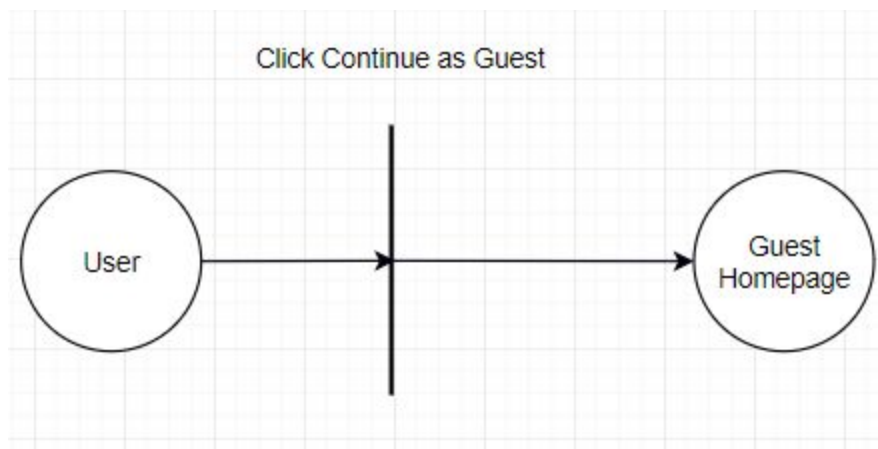
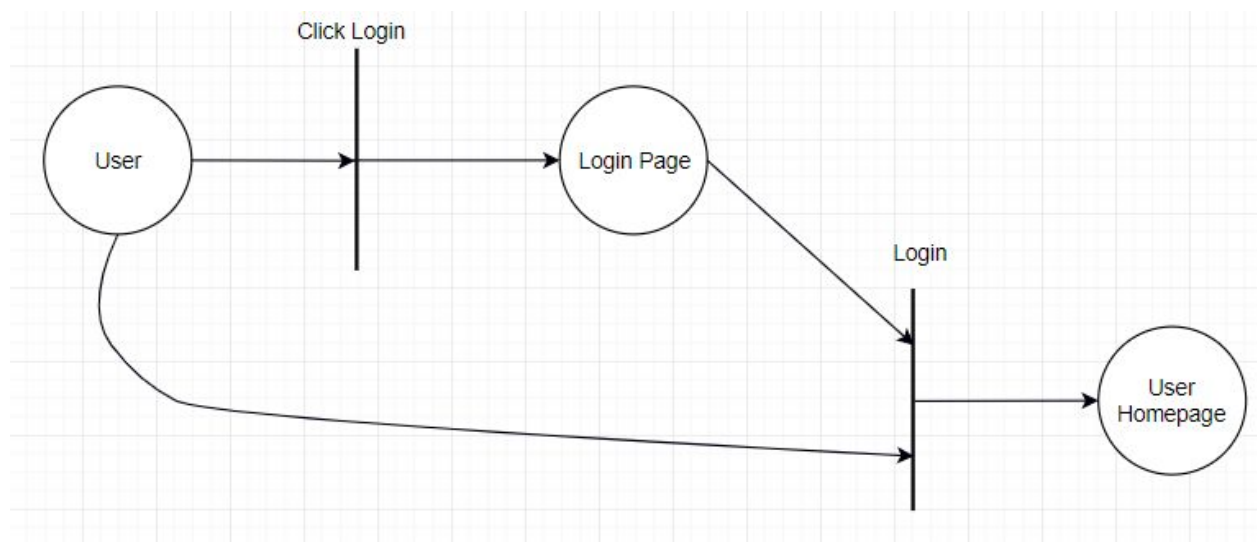
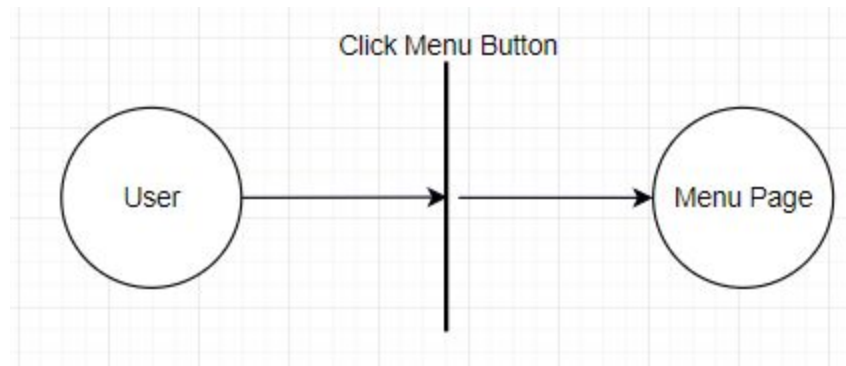
# Manage Employees

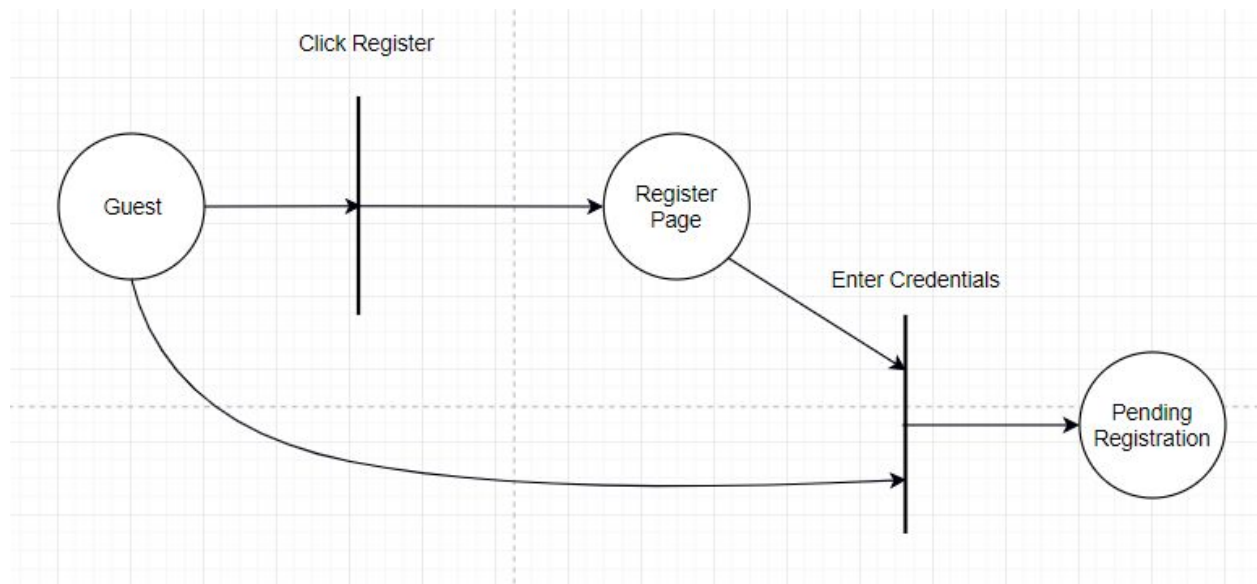


Approve Registration

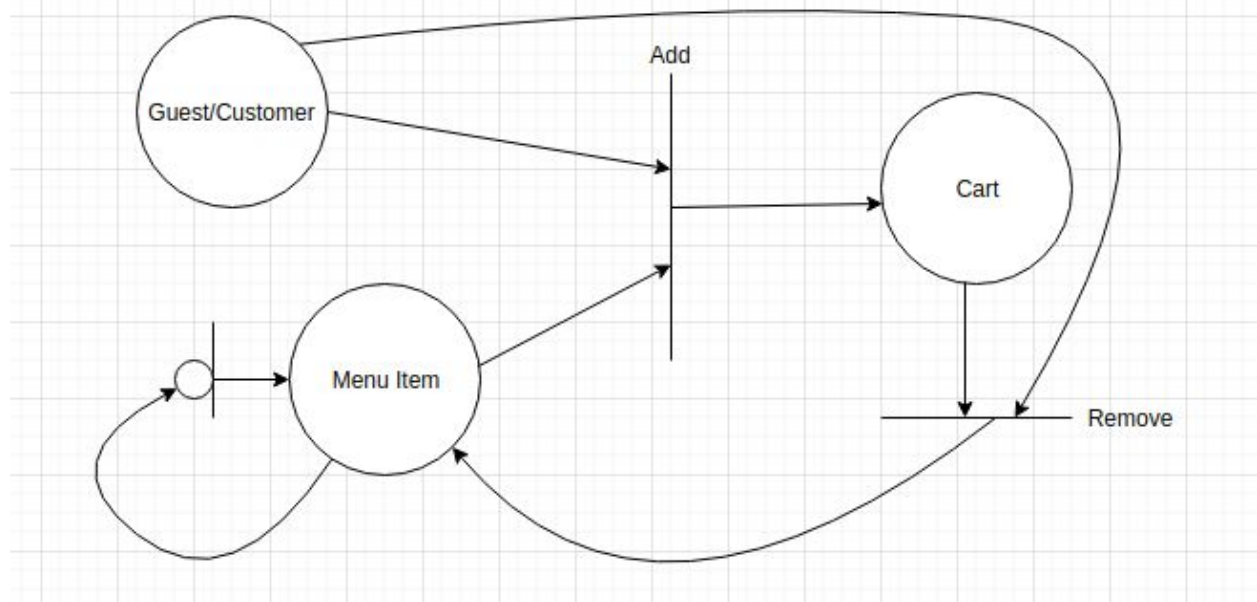


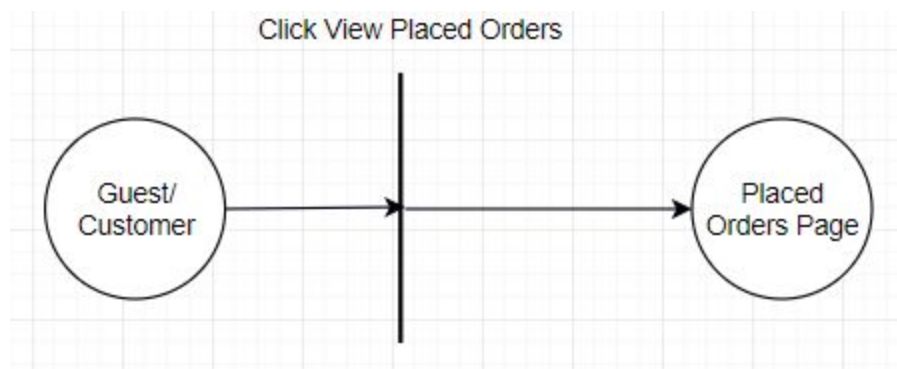
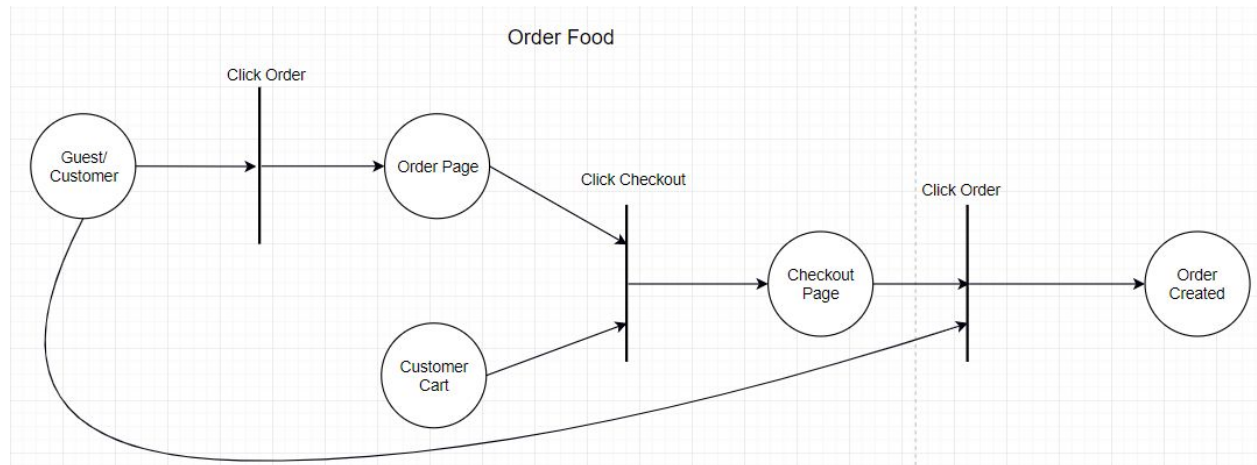
## 2.3 State Diagrams/ Petri-net

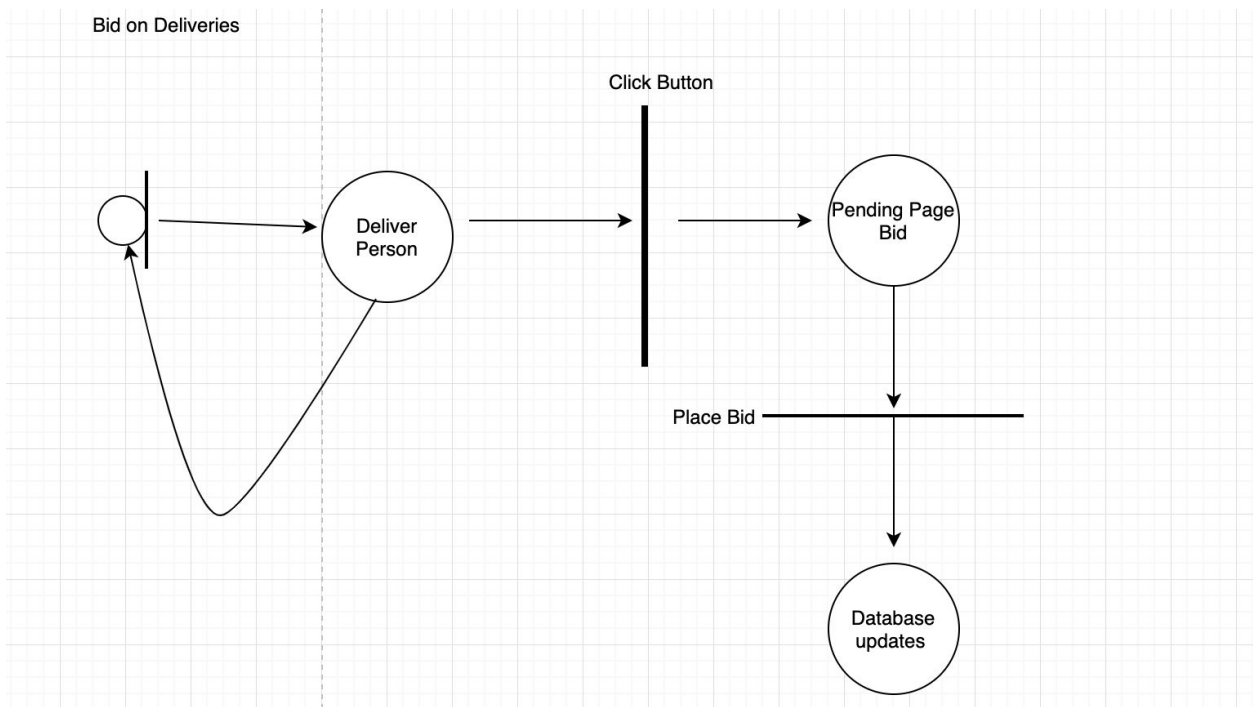
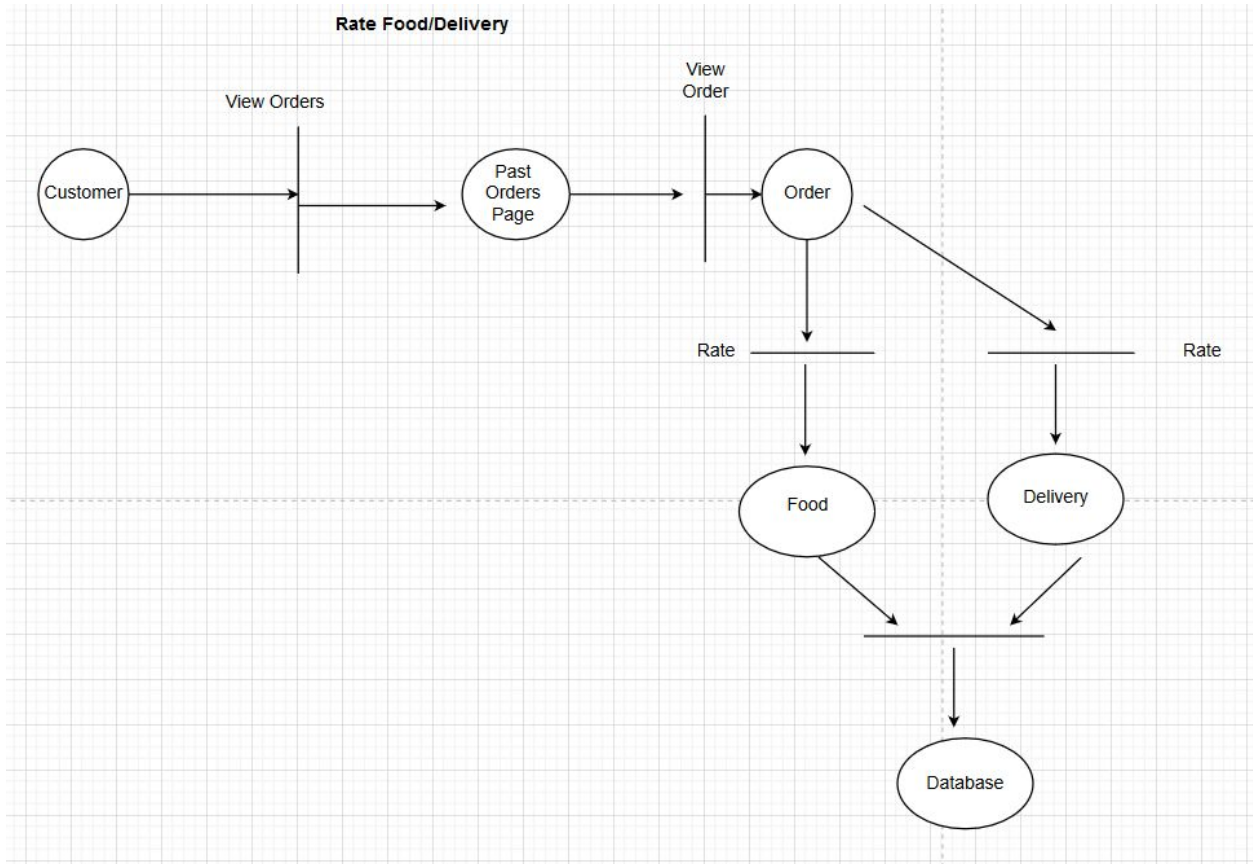




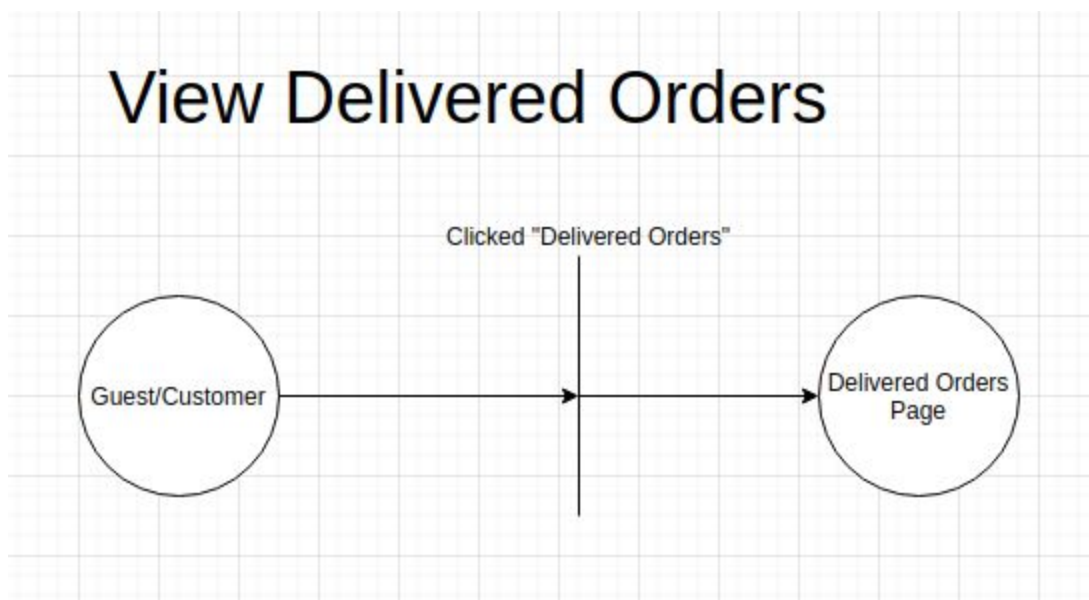
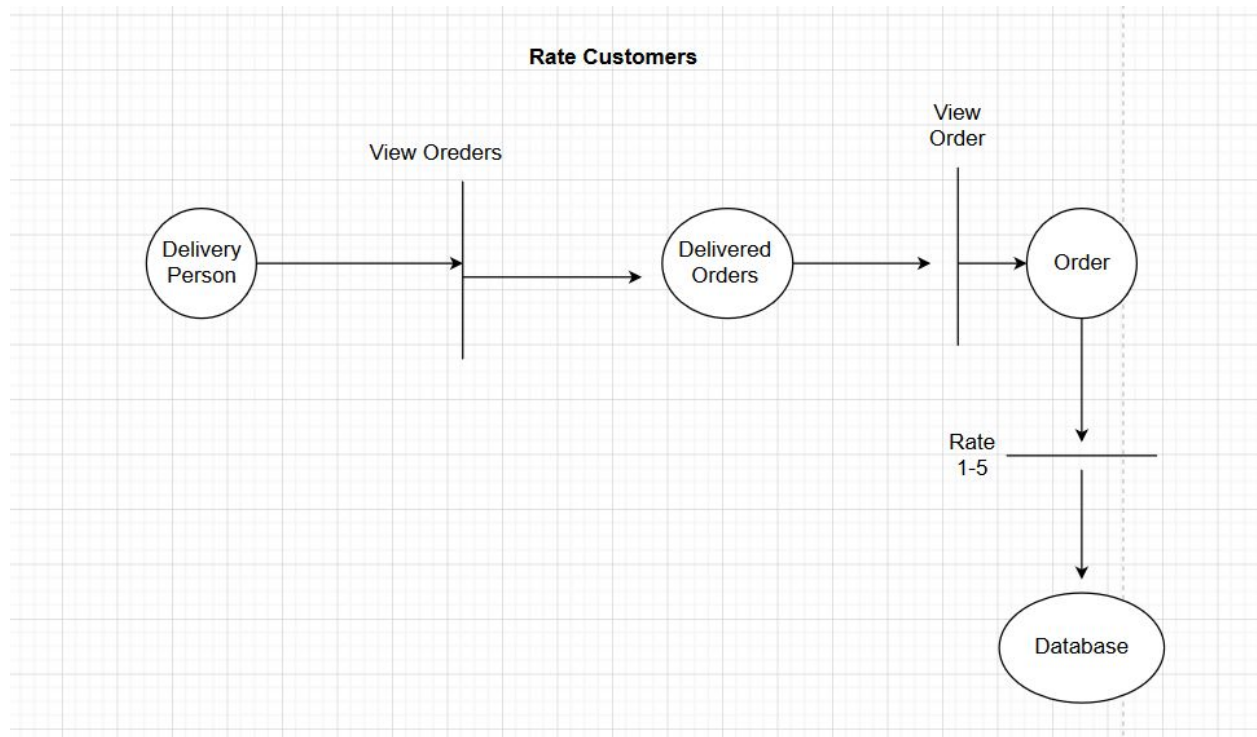
## Add/Remove Products to Cart

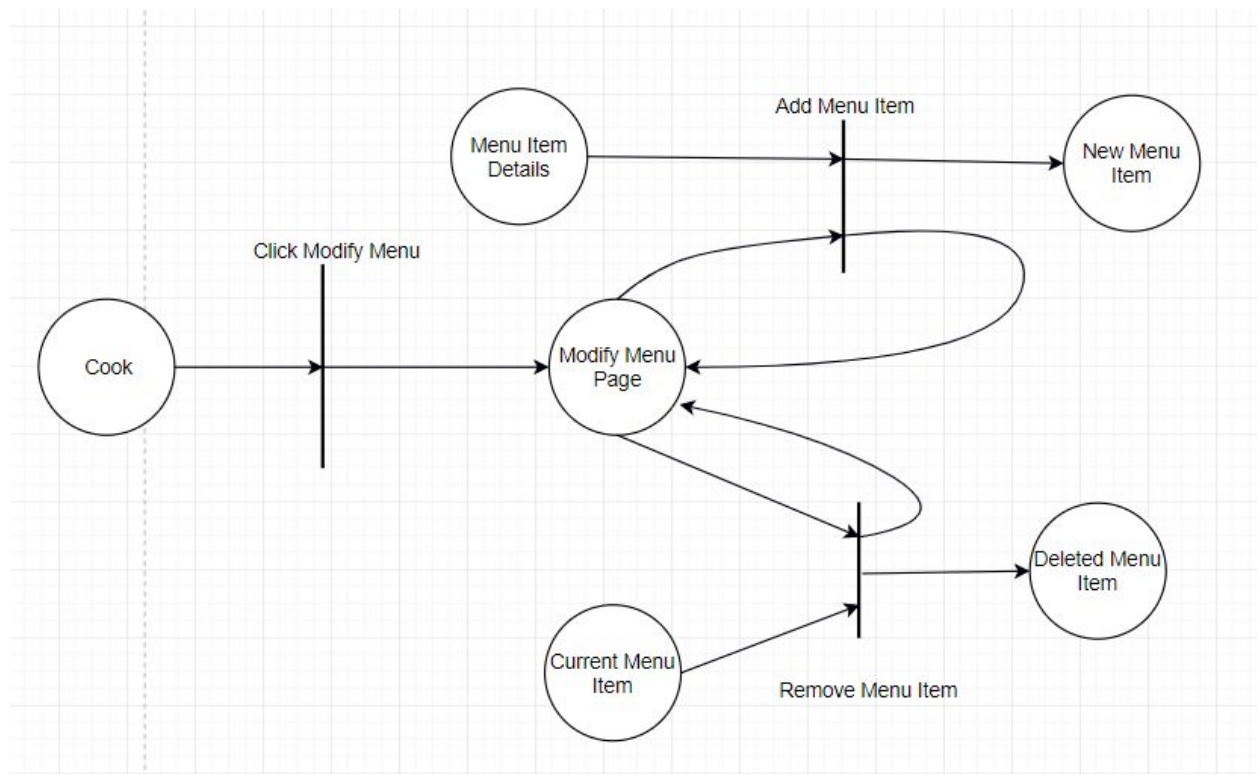






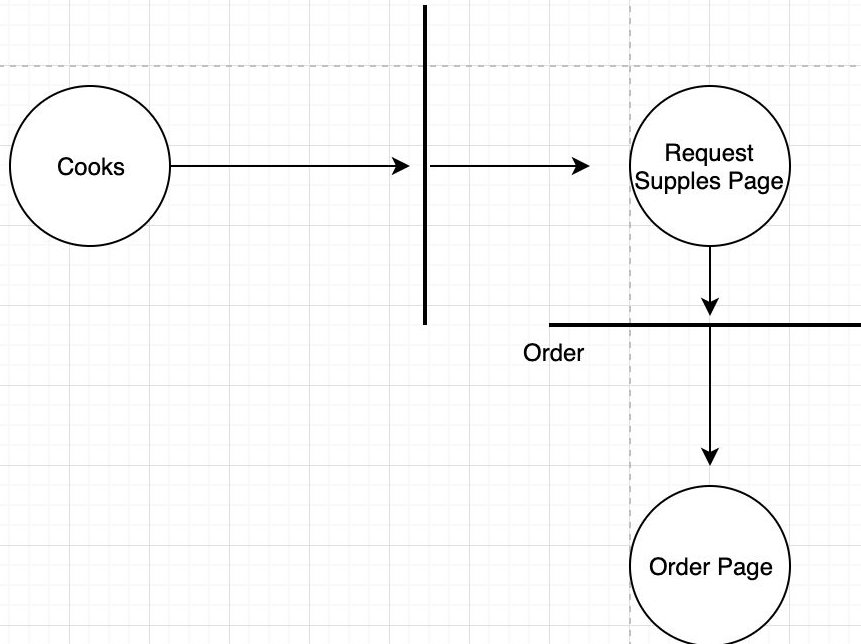




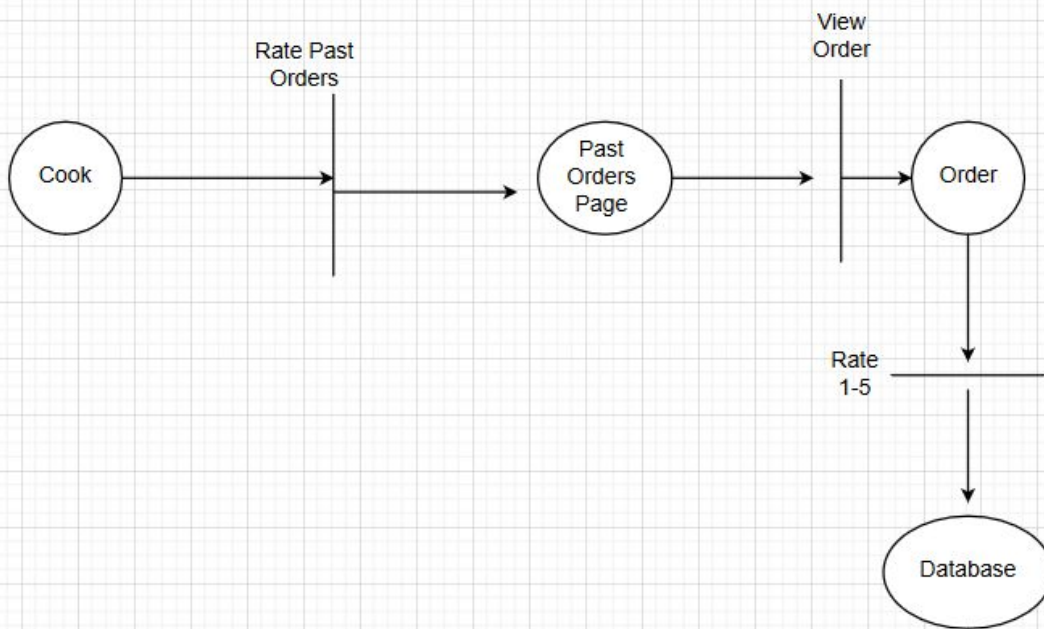


## Request Supplies

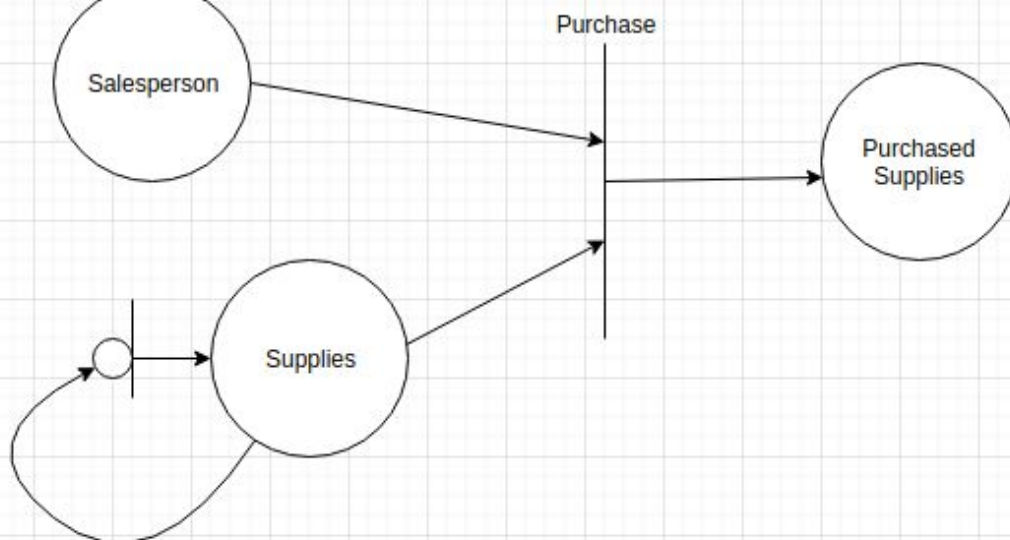
Click Request Supplies



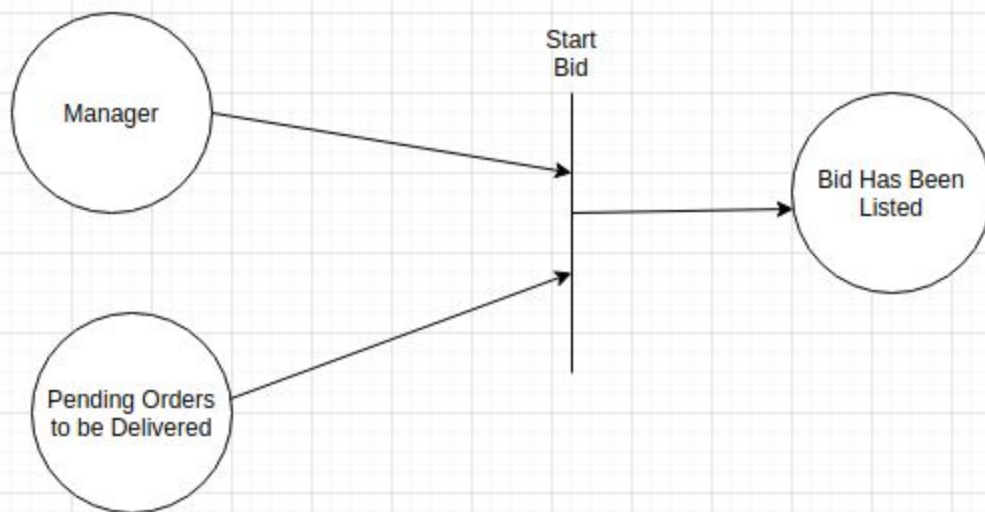
### Rate Supplies



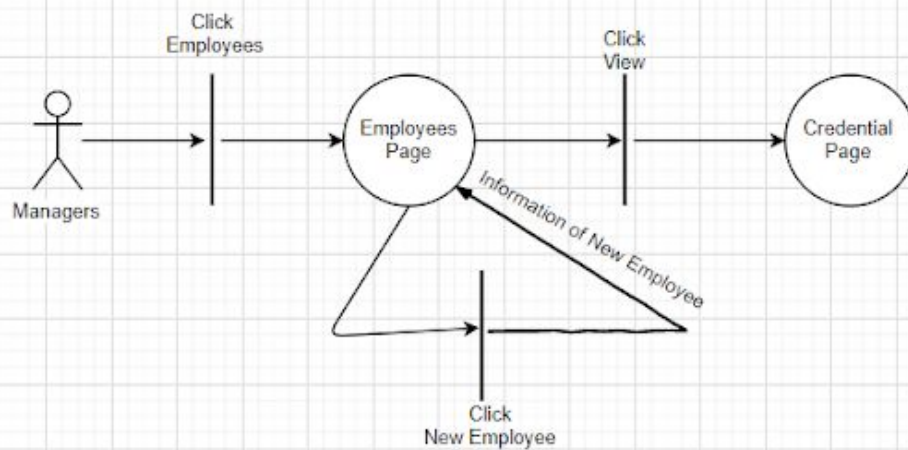
### Purchase Supplies



# Start Bids



# Manage Employees



## Approve Registration

