



Online Food Portal

Software Engineering Project (CSHP - 410)

Submitted By:

Tanya (19562)
Shivani (19585)

Supervisor:

Mrs. Kavita Rastogi



2021

Department of Computer Science
Shaheed Sukhdev College of Business Studies
University of Delhi

CONTENTS

Problem Statement.....	02
Process Model.....	03
1. Software Requirement Specification.....	04
1.1 Introduction.....	04
1.1.1 Purpose.....	04
1.1.2 Scope.....	04
1.1.3 Overview.....	05
1.1.4 References.....	06
1.2 Overall Description.....	06
1.2.1 Product Functions.....	06
1.2.2 User Characteristics.....	07
1.2.3 General Constraints.....	07
1.2.4 Assumptions & Dependencies.....	07
1.2.5 Functional Requirements.....	08
1.3 External Interface Requirements.....	10
1.3.1 User Interfaces.....	10
1.3.2 Hardware Interfaces.....	13
1.3.3 Software Interfaces.....	13
1.3.4 Communication Interfaces.....	13
1.4 Performance Requirement.....	13
1.5 Design Constraints.....	13
1.6 Other Requirements.....	13
1.7 Data Flow Diagram.....	14
1.8 Data Dictionary.....	19
2. Use Cases.....	21
3. Estimations.....	29
3.1 Function Points.....	29
3.2 Efforts.....	30
4. Scheduling.....	31
5. Risk Management.....	32
6. Data Design.....	36
7. Coding.....	40
8. Testing.....	42

PROBLEM STATEMENT

Online ordering has been around in the restaurant industry for quite some time, in fact, the first online food order was placed in 1994! However, in recent years and especially during the COVID-19 pandemic, online ordering has gone from a desirable convenience to an absolute must for restaurant survival. In addition to helping businesses stay afloat, online ordering offers many advantages to customers. Therefore, we here propose an online restaurant aggregator and food delivery system named FOOD 24 with an easy user interface and an all-time high efficacy. This system would provide information, menus and user-reviews of restaurants as well as food delivery options from partner restaurants in all the cities. While a few months ago, minimizing human contact during restaurant transactions was merely a preference, the COVID-19 pandemic has elevated that preference into a priority. Thus, this online platform even allows the dine out facilities for its users i.e., booking a table in a restaurant as per the timings of their choice. This platform acts as mediator between customers and local restaurants, offering access to various cuisines through a single website or smartphone application. Other seamless advantages of this system for customers are - the ability to easily reorder custom and favourite orders, the elimination of potentially waiting in a long line at the restaurant, and improved order accuracy. Considering the future aspect, even as that spread slows, customers may continue to enjoy the minimal contact that online ordering can offer as it can accommodate busy schedules and allows customers to safely send any kind of meal to friends, loved ones, and colleagues. Having food simply dropped at your doorstep will likely still be desired during customers' busy days.

PROCESS MODEL

We have used the Waterfall Model to represent our project.

Waterfall model is a sequential model that divides software development into pre-defined phases. Each phase must be completed before the next phase can begin with no overlap between the phases. Thus, we have used this Model because of the following reasons: -

- It is relatively simple and easier to understand and approach as compared to other models.
- The requirements of our system are well stated, not changing frequently and understood beforehand.
- Since the requirements are fixed and so work can proceed to completion in a linear manner.
- In this model we have to complete one stage before proceeding to the next. So, we have clearly defined stages and well understood milestones.
- The advancement in process does not need to be checked upon by the customer at every stage. Hence, this model does not create problems.
- The Waterfall Model provides a structured approach.
- Our application is not complicated and big
- Resources are available and trained
- Our project is short

1. SOFTWARE REQUIREMENT SPECIFICATION

1.1 INTRODUCTION

Food 24 is an online restaurant aggregator and food delivery system which provides information, menus and user-reviews of restaurants as well as food delivery options from partner restaurants in all the cities. This online platform even allows the dine out facilities for its users i.e. booking a table in a restaurant as per the timings of their choice. This platform acts as mediator between customers and local restaurants, offering access to various cuisines through a single website or smartphone application.

1.1.1 Purpose:

The purpose of developing this software is to provide a flexible, easier and convenient food delivery experience along with booking facilities. It can connect thousands of people with the restaurants all over the country at one time simultaneously. It helps the customers to order food at their convenience and even book tables hassle free and skip long queues.

1.1.2 Scope:

This system makes food availability and delivery a lot easier for its users. It connects them with everything they need, to enjoy hot and tasty food while sitting at their homes - assessing different food items and cuisines, finding the right outlet for ordering a specific food item on the basis of ratings of different restaurants, booking tables for meals, obtaining discounts and coupons, ordering from different outlets simultaneously, choosing between a variety of payment options and enjoying the door step delivery while having the necessary contact details of the delivery person.

1.1.3 Overview:

The user will first sign-up/login into the system. They will have to fill up the basic information like Name, Email Id, Phone number and Address. Once the sign-up/login is done, the user will be able to –

- Search for restaurant/food and will be able to see the following details about the restaurants - Name, Location, Phone No., Type of food (e.g., fast food), distance from house, approximately. time for delivering, ratings, items available, delivery charges. For food items, customers will be able to see details like name of items, rate, quantity, type (veg/non-veg).
- Buy the item(s) from outlets or customers can book a table in the restaurant for appropriate timing.
- Add items to cart or buy directly. If a customer adds the item to the card, then he/she can see the details like name of item, amount of item, quantity, outlet from which it is purchased, type of item(veg/non-veg), applied coupons, address to which it is being delivered. If a customer opts to buy directly, he/she will be redirected to the page containing order details.
- Cancel the table booking anytime and will receive a refund in their wallet.
- Pay the amount via various modes (Paytm, online banking, debit card, credit card, COD, software wallet).
- Apply any offers or coupons on their bookings and will be able to see the following details related to it like the type of offer (cashback, discount), name, id, outlets where offer is valid, description about the offer.
- Track their order and know the name of the delivery guy and his/her phone number, the current location of the order,

whether the order is picked by the delivery guy or not and approximate time in which the order will be delivered.

- See the table number if they've booked a table.
- Give ratings and write feedback about the food/restaurants.
- See their order history and the following details like Order no., Outlet name, amount, date and time of order, any offers applied, quantity, address at which it was delivered.
- See wallet section where he/she will be able to get details about the money available in their wallet, add money to the wallet and see details about previous refunds.

1.1.4 References:

- SOFTWARE ENGINEERING, A PRACTITIONER'S APPROACH - **By Roger S. Pressman**
- SOFTWARE ENGINEERING - **By Yogesh Singh**

1.2 OVERALL DESCRIPTION

1.2.1 Product Function:

- To provide a platform to all the people who have a busy schedule or those who don't want to get out of their beds, to order their favourite food at home or book a table in advance for their special occasions.
- To provide a platform to all the food outlets/restaurants to increase their sales and promote their businesses by providing food

availability options to their non-regular customers.

- To provide an employment opportunity to those who don't have proper educational qualifications by offering the job of delivery person.

1.2.2 User Characteristics:

- **Customers:** They are the ones who will login to the portal and fill up a signup page after which they can book a table in a restaurant, place an order for any food item and cuisine from any restaurant of their choice.
- **Food Outlets:** They are the ones who will login to the portal and take up online orders from customers. They can also book tables on the name of a particular customer and allot them a table number.
- **Delivery Person:** He is the one who will receive requests for delivering orders and will thus take up any one request at a time to deliver the food on time.
- **Admin:** The group of people who will maintain and handle the software.

1.2.3 General Constraints:

- This portal will only be available in English.
- This portal may behave inappropriately in Internet Explorer 9 or previous versions.

1.2.4 Assumptions & Dependencies:

- All facilities will be available 24*7 online.
- Users should be connected to the internet to access the system.

- **Budget Availability:** The determined budget is accurate and covers all project expenses.
- **Human Resource Availability:** All key project team members are available and have necessary skills and knowledge to work on the project.
- **Performance of Contractors, Suppliers & Vendors:** All necessary equipment and goods are available whenever you need them.
- Minimum android version required is Android 7(Nougat).

1.2.5 Functional Requirements:

- Ensuring all the users must have **signed up** to access the portal. Either customers, food outlets or the delivery persons will have to first **sign up** in the system and then again **login** to make use of the portal.
- Every customer has some **necessary data** like their Name, Phone Number, Email Id, Address which needs to be filled during signing up on the portal.
- Every food outlet has to fill up data like name of the outlet, owner name, food quality certificate, location, phone number, email address, list of the food items they provide along with their quantity and prices and their discount policy while signing up on the portal.
- Every delivery person has to fill data like name, phone number, email id, address, id proof and agree to give their location access while they are doing the job of delivering food.
- A customer will see the details like Name of the restaurant, Location, Phone No., Type of food (e.g., fast food), distance from house, approximate time for delivering, ratings, items available, delivery charges while **searching** for a restaurant & for food items customer will see details like name of items, rate, quantity, type (veg/non-veg)

- Customers will only be able to **buy** food from one restaurant in a single order.
- While **booking** a table customer has to wait for the confirmation from the restaurants.
- **Payment** mode has to be selected before confirming the order.
- Outlets will **receive orders** from the customers with following details like the name of the food and its quantity. Once the outlets **accept the order** for takeout then they will receive the details about the customer like name, address, phone number.
- Outlets can **accept or reject orders** for booking a table, they will receive details about the time at which the table needs to be booked and the number of persons. If the table is not available then outlets can reject the order. If outlets accept the order for the table booking, they will get details like the name of the person for whom it is booked and their phone number.
- Delivery persons will **receive requests** for the orders on the basis of their distance from the outlets and will receive the details like the outlet name, its location, address at which it needs to be delivered. They can **accept or reject** the requests based on their convenience. If they accept the order, they will receive information like order quantity, price, name of the customer and their phone number.
- Customer will **receive order/booking confirmation** and will be **redirected** to the page where he/she can see table details/order details like item name, outlet name, payment amount for the order and name & phone number of the delivery person.
- Delivery person will have to **provide confirmation** after picking up the food from the outlet and will have to grant access to their location.
- Once the **order is picked** by the delivery person, the customer **can track** their order and see the location of the delivery person and

approximate time in which the food will be delivered.

- Software will **use cookies** to store temporary progress.
- Delivery person will **provide confirmation** to the outlets after delivering the food and will transfer the amount to the outlet if the mode of payment is COD.
- Customers can **give ratings** to the food/restaurants and **write feedback** after receiving the order.
- Customers will only **receive a refund** in their wallet if they cancel the order/table booking.
- Customers can **see their order history** and will see the following details like Order no., Outlet name, amount, date and time of order, any offers applied, quantity, address at which it was delivered

1.3 EXTERNAL INTERFACE REQUIREMENTS

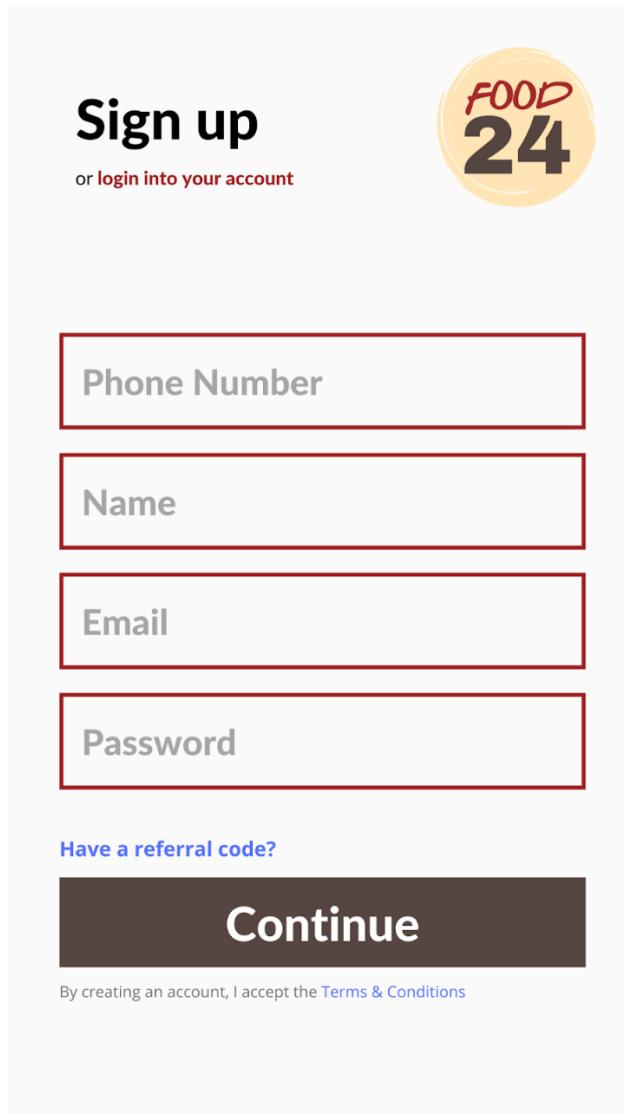
1.3.1 User Interfaces:

1.3.1.1 UI Screens:

- **LOGIN/SIGN-UP:** Screen for login and signup for users.
- **HOME PAGE:** After login, users will see a homepage where top rated food/restaurants will be mentioned and a list of offers and coupons, paid ads. User will also see a navigation bar from where he/she can access the search section, visit the cart, see the offers going on and the user account section.
- **SEARCH SECTION:** User will see the recent searches and popular cuisines. Users can search for desired food/restaurant.
- **CART:** Users will see the items he/she adds to the cart with item details and will get an option to get redirected to the payment gateway. If the cart is empty, the user will get an option to get redirected to the search section to browse for food/restaurants.
- **ACCOUNT:** User will see their personal details and will get an option to edit them, a list of their favourites, option to get

redirected to Payments and Refunds portal, option to get redirected to Wallet and Order History.

- **PAYMENTS & REFUNDS:** Users will see payment modes and will get an option to link their account to other apps to be able to access various payment modes. Users will also see any active refunds.
- **WALLET:** Users will see the amount available in their wallet and transaction history.
- **ORDER HISTORY:** Users will be able to see their past order and details related to every order. Users can rate or write a feedback or reorder the same.



❖ **User Sign-up Interface**

Moti Mahal Delux
Sector 18

Paneer Tikka Masala ₹243

Soya Malai Chaap ₹222

Tip your hunger saviour ⓘ
Thank your delivery partner, support them with a tip.

₹30 ₹50 ₹70 Other

Delivery to Home
Sector 46
28 mins.

₹526.00 View Detailed Bill PROCEED TO PAY

HOME SEARCH CART ACCOUNT

❖ Cart Interface

Bill Details

Item Total	₹565.00
Delivery Partner Fee	₹28.00
Total Discount	- ₹99.99
Delivery Tip	Add Tip
Taxes and Charges	₹33.25
To Pay	₹526.00

You have saved ₹99.99 on the bill.

HOME SEARCH CART ACCOUNT

❖ Payment Details/Bill

1.3.2 Hardware Interface:

- OPERATING SYSTEM: Windows and Unix based/Android
- PROCESSOR: Intel i5 or above.
- RAM: 512MB or above.

1.3.3 Software Interface:

- HTML, PHP, MYSQL, JAVASCRIPT, CSS

1.3.4 Communication Interface:

- E-mail, Skype
- NIC (Network Interface Card) - It is a computer hardware component that allows computers to connect to a network.

1.4 Performance Requirements:

System must generate output and respond within 3 seconds when any interaction takes place, else an error message should pop-up.

1.5 Design Constraint:

System will run on windows/Unix based platforms as well as android platforms.

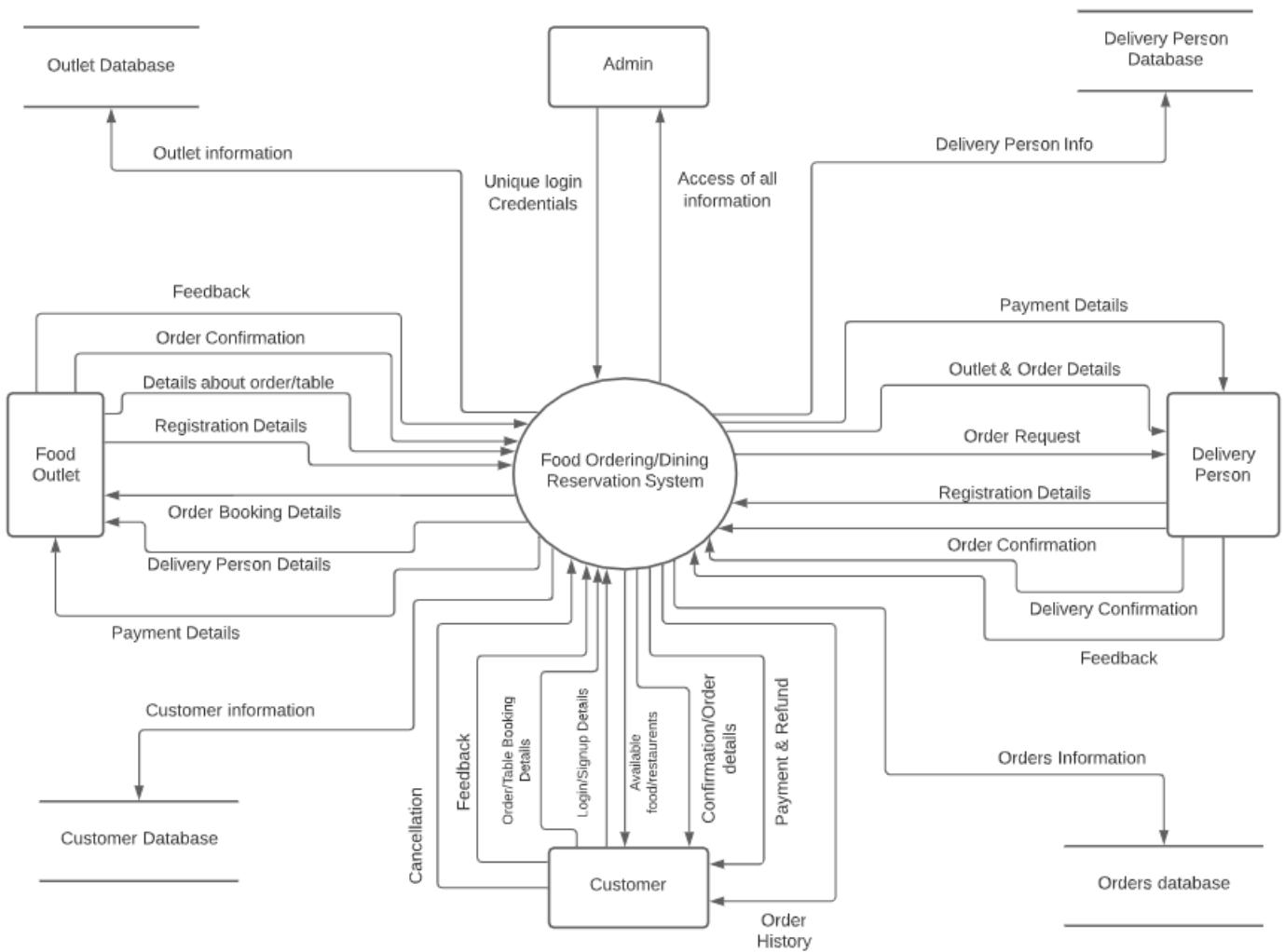
1.6 Other Requirements:

1.6.1 Non-functional requirements:

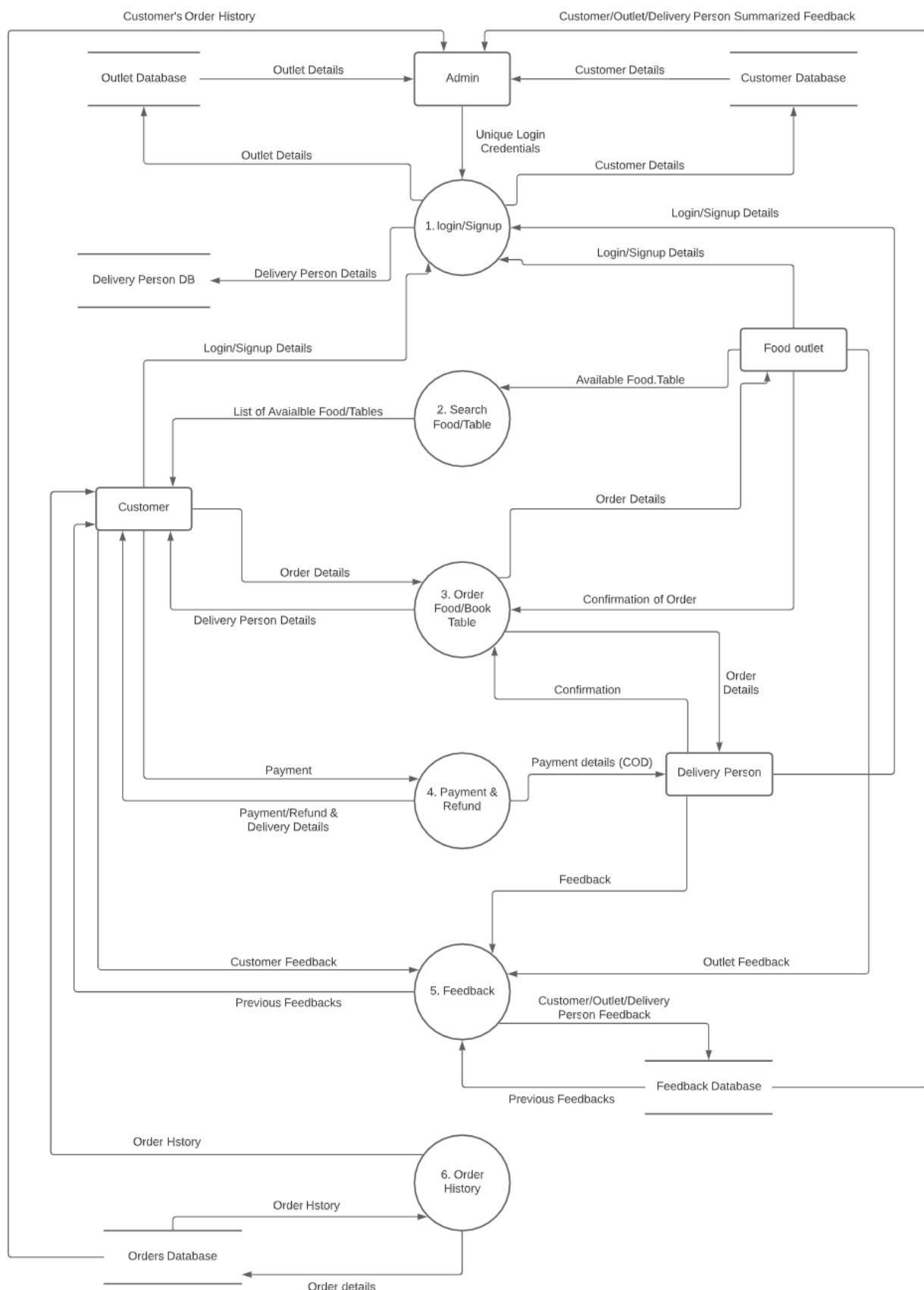
- User interface - User interface screen will respond within 5 seconds.
- Capacity -The system must support 1000 people at a time.
- Administrators and many other users can access the system but the access level is controlled for each user according to their work scope.

1.7 DATA FLOW DIAGRAMS

1.7.1 Level 0/ Context Level Diagram

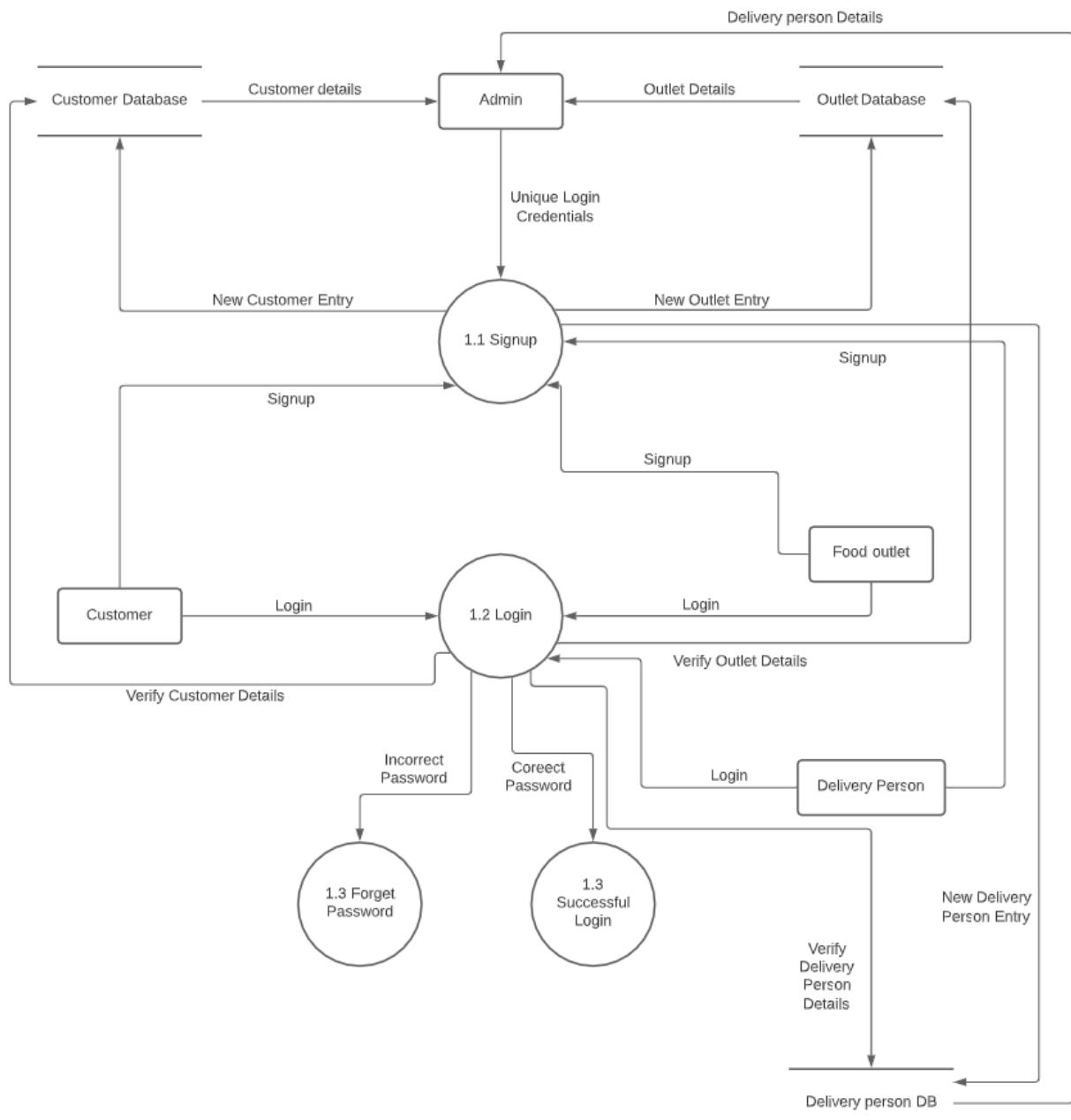


1.7.2 Level 1

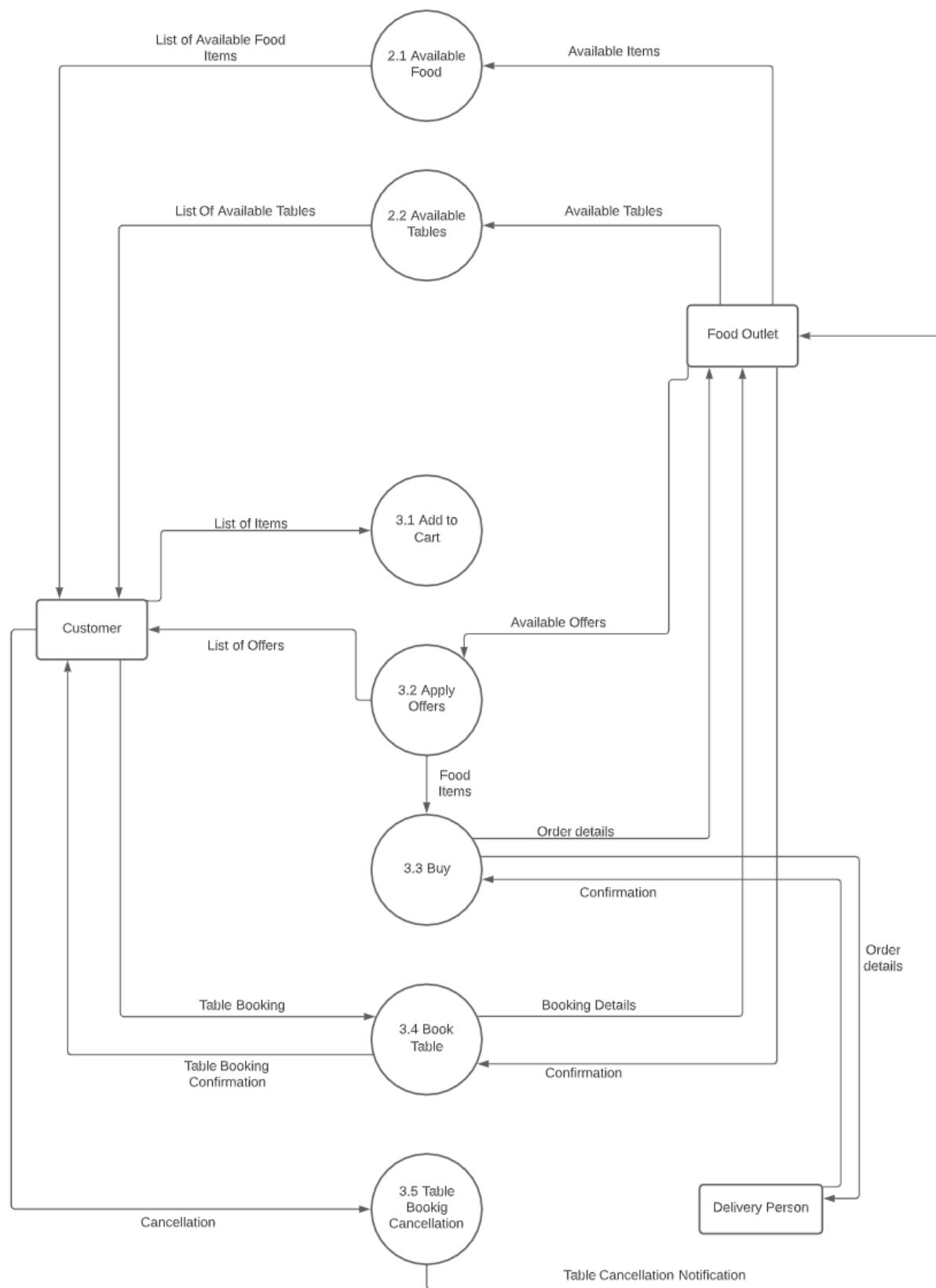


1.7.3 Level 2

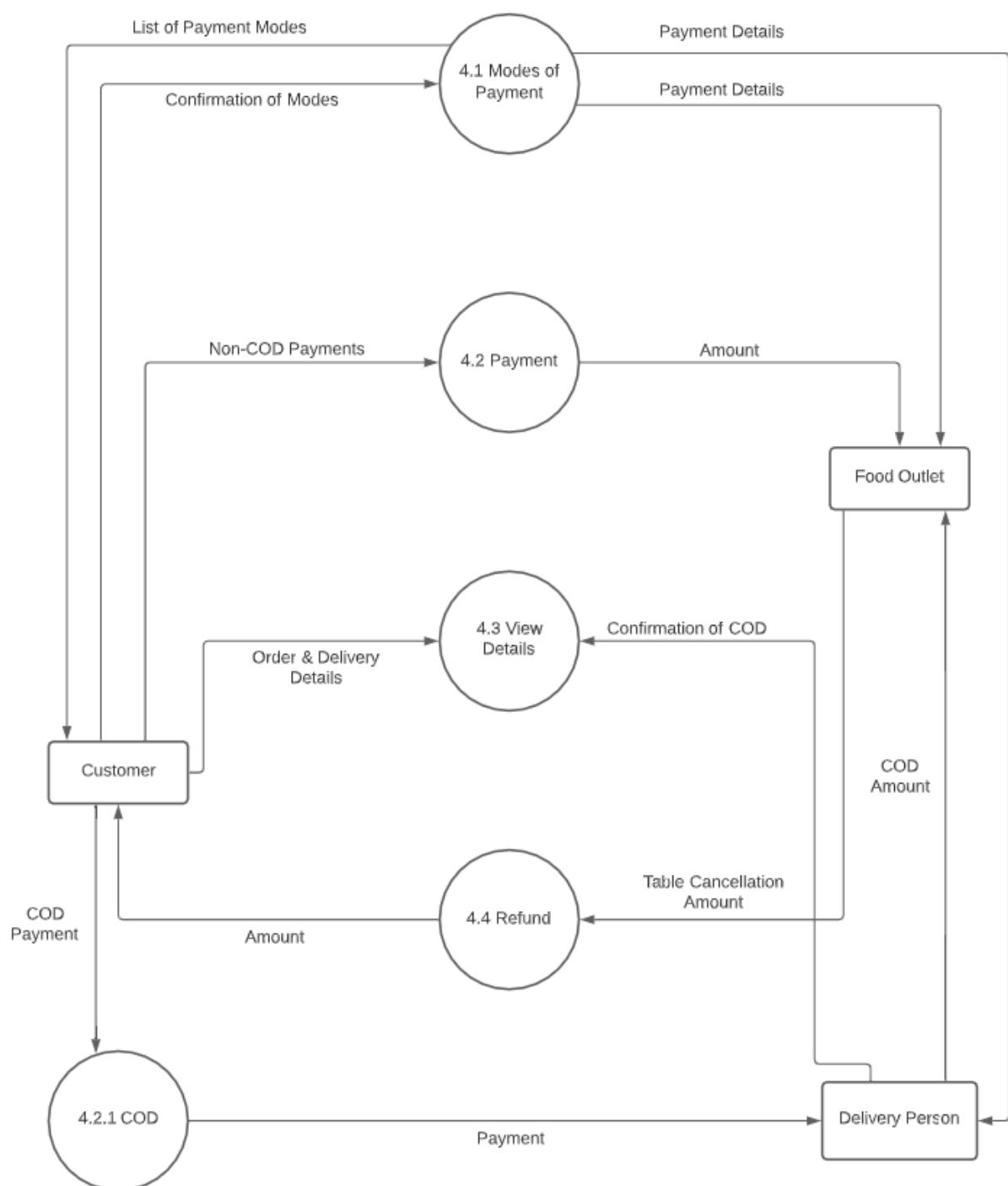
Level 2 - 1. Login/Signup



Level 2 - 2. Search & 3. Order



Level 2 - 4. Payment & Refund



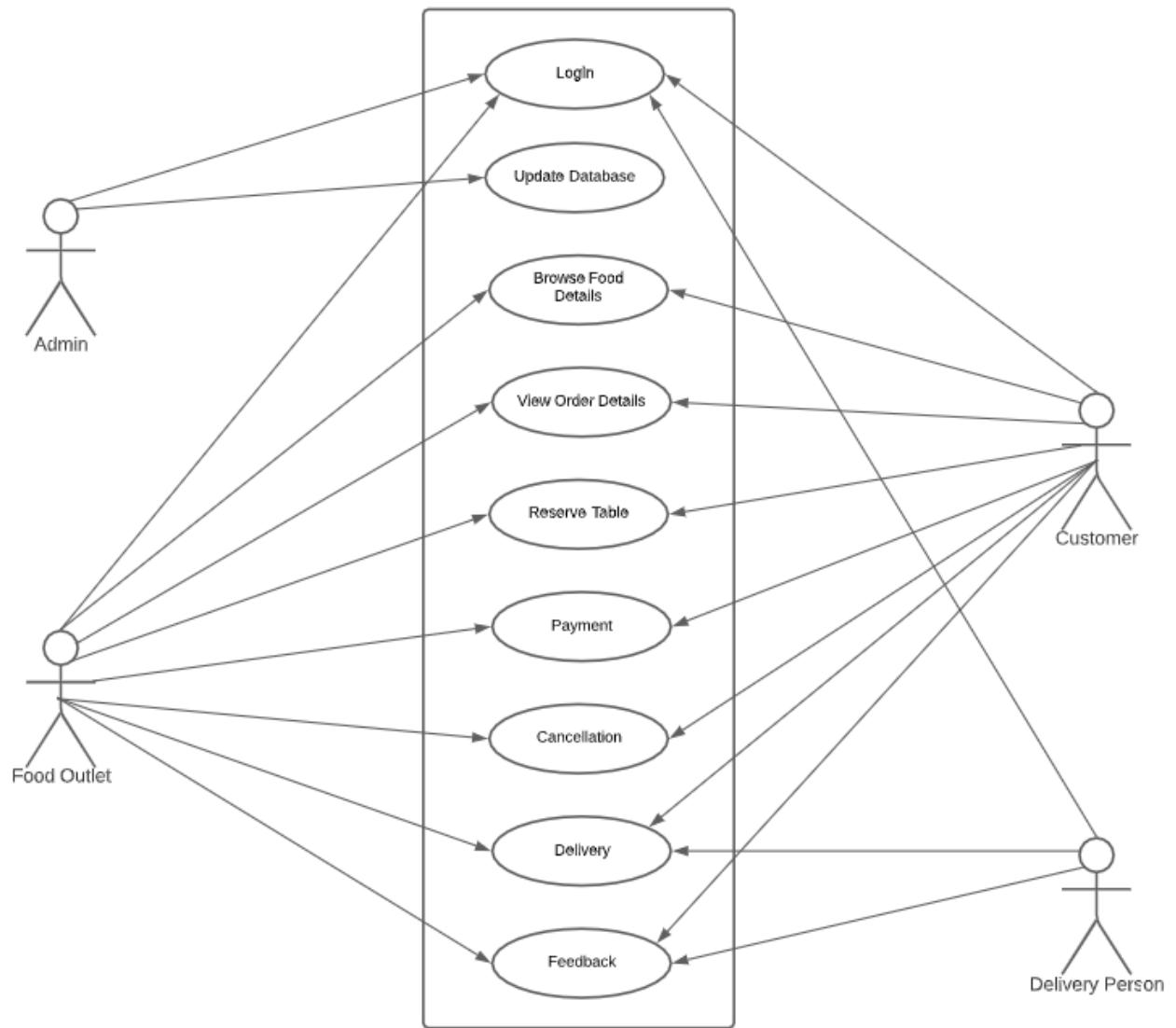
1.8 Data Dictionary

- ❖ **Customer Signup Details** - Fname+Lname+Phone_no+Email_id+House_no+Block+Sector+City+State+Pin_code+Password
- ❖ **Outlet Sign Up Details** - Name+Phone_no+Email_id+Outlet_no+Address+Owner_name+Food_quality_certificate+Website_link+Password
- ❖ **Delivery Person Signup Details** - Fname+Lname+Phone_no+Email_id+Id_proof+Password
- ❖ **Login** - Email_id+Password
- ❖ **Phone No** - digit+digit+digit+digit+digit+digit+digit+digit+digit
- ❖ **Outlet Details** - Name_of_outlet+Address+Phone_no+Type_of_food+Distance_from_the_house+Delivery_time+Ratings+Menu
- ❖ **Customer Details** - Fname+Lname+Address+Phone_no
- ❖ **Item Details** - Name+Price+Quantity+Type (veg/non-veg)
- ❖ **Confirmation** - Yes|No
- ❖ **Order Details** - Outlet_name+Item_details+Order_no
- ❖ **Cart Details** - Item_details+Outlet_name+Coupon_code+Customer_address+Total_amount
- ❖ **Offers** - Type (cashback, discount)+Outlet_name+Description
- ❖ **Coupon Details** - Coupon_code+Description+T&C
- ❖ **Payment** - Payment_mode+Bill+Date&time_of_payment
- ❖ **Bill** - Outlet_name+Item_name+Quantity+Delivery_charge+Total_discount+Delivery_tip+Taxes+Items_total_amount

- ❖ **Refund** - Order_details+Refund_amount
- ❖ **Delivery** – Name_of_Delivery_Person+Phone_no+Delivery_Time+Location
- ❖ **Feedback** - Rating+Feedback
- ❖ **Order History** - Order_details+Amount+Date_of_order+time_of_order+Delivery_address+Feedback
- ❖ **Address Details** - House_no+Block+Sector+City+State+Pin_code

2.USE CASE

❖ Use Case Diagram



❖ USE CASES

1. Login

1.1 Brief description

This use case describes how the users Login into our Food 24 Portal.

1.2 Actor

Admin, Customer, Food Outlet and Delivery Person actors interact in this use case.

1.3 Flow of events

1.3.1 Basic Flow

- The system will ask the user to enter Name, E-mail id, Phone number and generate a new password and verify the entered information using OTP on the registered/entered Email ID/Phone Number.
- The actor will enter the above details asked by system.
- The system will store the data into login details user database.

1.3.2 Alternative Flow

- If in the basic Flow, the actor enters an invalid email, phone number, or do not verify the entered details vis OTP the system displays an error message and the actor will need to enter those values again.

1.4 Pre-condition

None

1.5 Post-condition

The user shall be able to login into the system.

2. Update Database

2.1 Brief description

This use case describes how the Admin will update different databases through Food 24 Portal.

2.2 Actor

Admin can interact in this use case.

2.3 Flow of Events

2.3.1 Basic Flow

- User will need to select the desired database
- After selecting Plan user will update the data in the chosen database
- After a successful updation, the changes will be saved in the database

2.3.2 Alternative Flows

- If in the basic Flow, the details updated by the user are not valid then an error message will be shown to the user and the user can enter the details again.

2.4 Pre-Condition

None

2.5 Post-Condition

The new data will be updated in the database and the user will be able to see the changes if the updation is done successfully.

3. Browse Food Details

3.1 Brief description

This use case will show the available food item details of various restaurant aggregators registered on Food 24 Portal.

3.2 Actor

Customer and Food Outlet users can interact in this use case

3.3 Flow of events

3.3.1 Basic Flow

- The user will need to enter the food item to see the details available linked to it.
- Now the user can view the desired food details.

3.3.2 Alternative Flow

- If the food items searched by the user are not valid then an error message will be shown to the user and the user can enter the name of food items again.

3.4 Pre-condition

There has to exist a database of restaurants where the desired food item might be available.

3.5 Post Condition

User can see the details of the desired food item.

4. View Order Details

4.1 Brief description

This use case will describe the details of orders placed through the Food 24 Portal.

4.2 Actor

Food Outlet and Customer can interact in this use case

4.3 Flow of events

4.3.1 Basic Flow

- The user will have to enter the Bill No. for that particular order to see the order details.
- Now the portal will display the details of desired order.

4.3.2 Alternative Flow

- If the Bill No. entered by the user is not valid then an error message will be shown to the user and the user can enter the Bill No. again.

4.4 Pre-condition

There has to exist a database keeping the track of orders along with their details.

4.5 Post-condition

User shall be able to see the details of desired order.

5. Reserve Table

5.1 Brief description

This use case will describe how the user can book a table in a restaurant through the Food 24 Portal.

5.2 Actor

Food Outlet and Customer can interact in this use case

5.3 Flow of events

5.3.1 Basic Flow

- The user will have to select a restaurant from the list of restaurants that offer table reservation facilities
- The user will have to select a table for reservation from the list of unreserved tables.
- The user will have to enter the time for which he/she wishes to reserve the table.
- The user will have to pay some advance amount to confirm the reservation.
- Now the portal will make the desired reservation.

5.3.2 Alternative Flow

- If the user is unable to proceed towards the advance payment after entering the time for which the reservation has to be done, then the user will be redirected to the table reservation page to choose another table as it might have got booked meanwhile or to again.

5.4 Pre-condition

There has to exist a list of restaurants that offer table reservation facilities.

5.5 Post-condition

User must reach the restaurant and avail the table booking maximum by 15 minutes late to the selected time.

6. Payment

6.1 Brief description

This use case will describe how the user will pay for the order/table reservation on Food 24 Portal.

6.2 Actor

Food Outlet and Customer can interact in this use case

6.3 Flow of events

6.3.1 Basic Flow

- The customer will need to choose the payment method first, then enter the required details i.e., Credit Card No./Debit Card No./UPI ID.
- Customers can now verify the payment using a registered Email ID or Phone No.
- Customer will now receive a confirmation mail about Payment or transaction.
- Food Outlet will now receive the payment in its account.

6.3.2 Alternative Flow

- If the Payment details entered by the customer are not valid, then an error message will be shown and the customer can enter the payment details again.

6.4 Pre-condition

There has to exist a bank account linked with registered phone no.

6.5 Post-condition

Customer will receive the payment receipt on the registered main ID and the Food Outlet also, will be able to see the transaction done in its account, if payment is done successfully.

7. Cancellation

7.1 Brief description

This use case will describe how the user can cancel the order on Food 24 Portal.

7.2 Actor

Food Outlet and Customer can interact in this use case.

7.3 Flow of events

7.3.1 Basic Flow

- The customer will have to enter the Table No. of the restaurant.
- Customer will then confirm cancellation of the reservation.
- Customer will now receive a confirmation mail about cancellation of reservation.
- Food Outlet will now cancel the reservation and will make the refund within 24 Hrs.

7.3.2 Alternative Flow

- If the Table No. entered by the customer is not valid, then an error message will be shown and the customer can enter the Table No. again.

7.4 Pre-condition

There has to exist a database of reservations to keep track of all the table bookings.

7.5 Post-condition

Customer will receive a confirmation of cancellation of reservation on the registered Email Id and the Food Outlet will cancel the table booking.

8. Delivery

8.1 Brief description

This use case will describe how the delivery of the order will be done through Food 24 Portal.

8.2 Actor

Food Outlet, Customer and Delivery Person can interact in this use case

8.3 Flow of events

8.3.1 Basic Flow

- The autonomous system of this Portal will assign a Delivery Person, the delivery of an order depending upon the distance of the Food Outlet from the live location of the Delivery Person and his availability.
- The Delivery Person may or may not accept the assignment of a particular delivery within 5 minutes of the notification by the system.
- If the Delivery person doesn't accept the assignment, then the system will assign the delivery to some other person. But if the Delivery Person accepts the assignment, then he shall reach the Food Outlet to take the order for delivery and switch on its live location for the Customer too.

- Now the Delivery Person shall deliver the order and accept the payment too, if its COD and send a confirmation to the system and the Food Outlet about the successful delivery of the order.
- The Customer shall also mark the delivery as order received on the portal so as to confirm the delivery and thus the closing of order. Until all this is done, the system shall not close the order.

8.3.2 Alternative Flow

- If the delivery of the order is not received by the customer within 50 minutes (max) from the time of order placement, then the Customer may cancel the order or call the Delivery Person to know about the delivery of order.

8.4 Pre-condition

The order must have been confirmed by the Customer.

8.5 Post-condition

Customer will receive the order.

9. Feedback

9 .1 Brief description

This use case will register feedback of the users through Food 24 Portal.

9.2 Actor

Customer, Food Outlet and Delivery Person can interact in this use case.

9.3 Flow of events

9.3.1 Basic Flow

- Users will register the feedback after the usage of Food 24 Portal.

9.4 Pre-condition

Users will first have to use the portal to give the feedback.

9.5 Post-condition

The concerned users can see the received feedback and improve their services accordingly.

3. ESTIMATION

3.1 Function Points

S.No.	QUESTIONS	GRADE VALUE
1.	Does the system require reliable backup and recovery?	5
2.	Are specialised data communication required to transfer information to or from the application?	4
3.	Are there distributed processing functions?	3
4.	Is performance Critical?	0
5.	Will the system run in an existing, heavily utilized operational environment?	5
6.	Does the system require online data entry?	5
7.	Does the online data entry require the input transaction to be built over multiple screens or operations?	2
8.	Are the ILFs updated online?	5
9.	Are the inputs, outputs, files, or inquiries complex?	2
10.	Is the internal processing complex?	3
11.	Is the code designed to be reusable?	3
12.	Are conversions and installations included in different organizations?	3
13.	Is the system designed for multiple installations in different organizations?	0
14.	Is the application designed to facilitate change and for ease of use by the user?	2

VALUE ADJUSTMENT FACTOR: $\Sigma f_i = 42$

Rate on each Factor on scale of 0 to 5:

0: No Influence

1: Incidental

2: Moderate

3: Average

4: Significant

5: Essential

Information Domain Value	Estimated Count			Weighing Factor			Total
	Simple	Average	Complex	Simple	Average	Complex	
External Input	2	1	1	3	4	6	16
External Output	1	0	2	4	5	7	18
External Inquiries	1	1	2	3	4	6	19
Number of Logical Files	3	0	2	7	10	15	51
External Interface Files	0	2	0	5	7	10	14

UNADJUSTED FUNCTION POINT (UFP): $16 + 18 + 19 + 51 + 14 = 104$

$$\begin{aligned}
 \text{COMPLEXITY ADJUSTMENT FACTOR (CAF)} &: 0.65 + (0.01 * \sum f_i) \\
 &= 0.65 + (0.01 * 42) \\
 &= 1.07
 \end{aligned}$$

$$\begin{aligned}
 \text{FUNCTION POINT METRIC (FP)} &: UFP * CAF \\
 &= 104 * 1.07 \\
 &= 111.28
 \end{aligned}$$

3.2 Cost and Efforts

Average productivity rate in India = 14.00 FP/person-month

Burdened Labour Rate = \$ 8000/month

Cost per FP = Burdened Labour Rate/Productivity = \$8000/14.00 FP = \$571.42/FP

Total Cost = Cost per FP x FP = \$571.42/FP x 111.28 FP = \$ 63,587

Total effort = FP/Productivity = 111.28/14.00 person-month = 7.95 person-month = 8 person-month (approx.)

4. SCHEDULING

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10
1. Identify Customer Requirements										
Meet Customers										
Identify Needs and Constraints										
Establish Problem Statement										
Describe Process Model										
Milestone: Problem Statement Defined					◆					
2. Define Function Behaviour										
Document SRS										
Design Function Module										
Design DFD										
Database Design										
Milestone: System Function Defined							◆			
3. Estimation										
Function Point Estimation										
Estimate Schedule of Project										
Milestone: Project Scheduling Concluded							◆			
4. Perform Risk Analysis										
Developing Risk Management										
Milestone: Risk Management								◆		
5. Design Development										
Formulated System Architecture										
Generate Code										
Milestone: System Design Developed								◆		
6. Testing										
Developing Test Cases										
Calculate Cyclomatic Complexity										
Develop Flow Graph										
Milestone: Testing Complete									◆	

5. RISK MANAGEMENT

RISK TABLE: -

S.NO.	RISK	CATEGORY	PROBABILITY	IMPACT
1.	Size estimate is very low	PS	30%	1
2.	Project team unable to keep up with schedule	PR	25%	2
3.	Staff may be inexperienced	ST	35%	3
4.	Lack of training on project development tools	DE	30%	2
5.	The code built is not flexible to incorporate changes	TE	20%	2
6.	The users of the app may not have technical background	CU	25%	3

IMPACT TYPE: -

- 1- Catastrophic
- 2- Critical
- 3- Marginal
- 4- Negligible

CATEGORY TYPE: -

ABBREVIATION	FULL FORM
PS	Product Size
PD	Product Definition
PR	Project Risk
BU	Business Impact Risk
ST	Staff Risk
TR	Technical Risk
CU	Customer Risk
DE	Development Environment

RISK MITIGATION, MONITORING & MANAGEMENT PLAN (RMMM-PLAN)

RMMM Plan (For risks above cut-off line)

(1) Size estimate is very low

i) Mitigation:

- Estimation should be done using good estimation technique
- More than one method of estimation should be used
- Should not use estimates of past projects as final estimates for the project
- Estimation should not be based on guesswork
- Decomposition should be done to correctly estimate the size

ii) Monitoring:

- Estimates of each decomposed part should be verified again
- Monitor that requirements are not changed frequently
- Check whether decomposition is done correctly

iii) Management:

- Re-estimate the size using different methods
- Adjust budget and cost using new estimates

(2) Project team unable to keep up with schedule

i) Mitigation:

- Make a realistic schedule
- Take the software team's experience and skills into consideration when making schedule

ii) Monitoring:

- Monitor the timeliness of work products
- Monitor the efficiency of project team
- Monitor the frequency of missed deadlines

iii) Management:

- Speak to the software team and get review as to why the deadlines are getting missed
- Adjust the deadlines according to project team skills
- Divide the work if required

(3) Staff may be inexperienced

i) Mitigation:

- Proper training to be given through workshops and presentation
- Ensure at least one experienced member in every team to guide the others

ii) Monitoring:

- Monitor the productivity of the team
- Monitor the quality of the work-products

iii) Management:

- Reassign work and teams based on experience
- Hold more training session to give hands-on experience

(4) Lack of training/knowledge of project development tools

i) Mitigation:

- Encourage software teams to use project development tools
- Regularly hold training sessions to get hands-on experience on project development tools

ii) Monitoring:

- Monitor how efficiently and timely each work product is delivered
- Check the tools used by various teams

iii) Management:

- Encourage the use of open-source project development tools
- Hold training sessions where experienced developers train the inexperienced team members

6. Data Design

TABLE 1 (CUSTOMER DB)

Attribute	Data type	Constraint
Customer_id	NUMBER(10)	PRIMARY KEY
First_name	VARCHAR(20)	NOT NULL
Last_name	VARCHAR(20)	NOT NULL
Phone_no	NUMBER(10)	NOT NULL
Email_id	VARCHAR(20)	NULL
Customer_address	VARCHAR(100)	NOT NULL
Password	VARCHAR(8)	NOT NULL

TABLE 2 (OUTLET DB)

Attribute	Data type	Constraint
Outlet_name	VARCHAR(20)	NOT NULL
Owner_name	VARCHAR(20)	NOT NULL
Outlet_id	NUMBER(10)	PRIMARY KEY
Food_quality_certificate	VARCHAR(30)	NOT NULL
Website_link	VARCHAR(50)	NOT NULL
Phone_no	NUMBER(10)	NOT NULL
Email_id	VARCHAR(20)	NULL
Outlet_address	VARCHAR(100)	NOT NULL
Password	VARCHAR(8)	NOT NULL

TABLE 3 (DELIVERY PERSON DB)

Attribute	Data type	Constraint
First_name	VARCHAR(20)	NOT NULL
Last_name	VARCHAR(20)	NOT NULL
Phone_no	NUMBER(10)	NOT NULL
Email_id	VARCHAR(20)	NULL
Id_proof	VARCHAR(30)	NOT NULL
Password	VARCHAR(8)	NOT NULL
Delivery_person_id	NUMBER(10)	PRIMARY KEY

TABLE 4 (ITEM DB)

Attribute	Data Type	Constraint
Item_id	NUMBER(10)	PRIMARY KEY
Item_name	VARCHAR(20)	NOT NULL
Item_price	NUMBER(10)	NOT NULL
Item_quantity	VARCHAR(20)	NOT NULL
Item_type	VARCHAR(20)	NOT NULL
Cart_id	NUMBER(10)	FOREIGN KEY(References Cart_id in Cart DB)
Order_id	NUMBER(10)	FOREIGN KEY(References Order_id in Order DB)

TABLE 5 (CART DB)

Attribute	Data type	Constraint
Customer_id	NUMBER(10)	FOREIGN KEY(References Customer_id in Customer DB)
Cart_id	NUMBER(10)	PRIMARY KEY
Outlet_name	VARCHAR(20)	FOREIGN KEY(References Outlet_name in Outlet DB) + Unique
Coupon_code	VARCHAR(20)	NULL
Customer_address	VARCHAR(20)	NOT NULL
Total_amount	NUMBER(10)	NOT NULL

TABLE 6 (PAYMENT DETAILS)

Attribute	Data type	Constraint
Bill_no	NUMBER(10)	PRIMARY KEY
Order_id	NUMBER(10)	FOREIGN KEY (References Cart_id in Cart DB)
Customer_id	NUMBER(10)	FOREIGN KEY(References Customer_id in Customer DB)
Payment_mode	VARCHAR(20)	NOT NULL
Date_of_payment	DATE(10)	NOT NULL
Time_of_payment	TIME(10)	NOT NULL
Delivery_charge	NUMBER(10)	NULL
Total_discount	NUMBER(10)	NULL
Delivery_tip	NUMBER(10)	NULL
Taxes	NUMBER(10)	NOT NULL
Items_total_amount	NUMBER(10)	NOT NULL

TABLE 7 (ORDER DB)

Attribute	Data type	Constraint
Customer_id	NUMBER(10)	FOREIGN KEY(References Customer_id in Customer DB)
Outlet_id	NUMBER(10)	FOREIGN KEY(References Customer_id in Outlet DB)
Order_id	NUMBER(10)	PRIMARY KEY
Bill_no	NUMBER(10)	FOREIGN KEY(References Bill_no in Payment DB)
Delivery_person_id	VARCHAR(20)	FOREIGN KEY(References Delivery_person_id in Delivery_person DB)
Delivery_person_pno	NUMBER(10)	NOT NULL
Delivery_time	TIME(10)	NOT NULL
Tracking_location	VARCHAR(20)	NOT NULL

TABLE 8 (TABLE BOOKING DB)

Attribute	Data type	Constraint
Customer_id	NUMBER(10)	FOREIGN KEY(References Customer_id in Customer DB)
Outlet_name	VARCHAR(20)	FOREIGN KEY(References Outlet_name in Outlet DB)
Table_no	NUMBER(10)	PRIMARY KEY
Status	BOOLEAN	NOT NULL
Date	DATE	NOT NULL
Time	TIME	NOT NULL

TABLE 9 (FEEDBACK DB)

Attribute	Data type	Constraint
Outlet_name	VARCHAR(20)	FOREIGN KEY(References Outlet_name in Outlet DB)
Order_id	NUMBER(10)	PRIMARY KEY
Rating	NUMBER(10)	NULL
Feedback	VARCHAR(20)	NULL

7. CODING

```
#include<iostream>
using namespace std;
1. int main()
{
    int ch1, ch2, ch3;
    int day, month, year;
2. do
{
    cout<<"\t\tWelcome to Food24\n";
    cout<<"\n\tMenu:\n\t1. New Customer? Sign-up\n\t2. Login\n\t3. Search
for food\n\t4. Cart\n\t5. My Account\n\t6. Payments & Refunds\n\t7.
Order History\n";
    cout<<"\nEnter your choice: ";
    cin>>ch2;
3. if(ch2 == 1)
4. {
    cout<<"Order food from your favourite restaurants near you!"<<endl;
    signUp();
}
5. else if(ch2 == 2)
6. {
    cout<<"Delighted to have you back!"<<endl;
    login();
}
7. else if(ch2 == 3)
8. {
    cout<<"Search for Restaurants & Food"<<endl;
    search();
}
9. else if(ch2 == 4)
10. {
    cout<<"Cart"<<endl;
    cart();
}
11. else if(ch2 == 5)
12. {
```

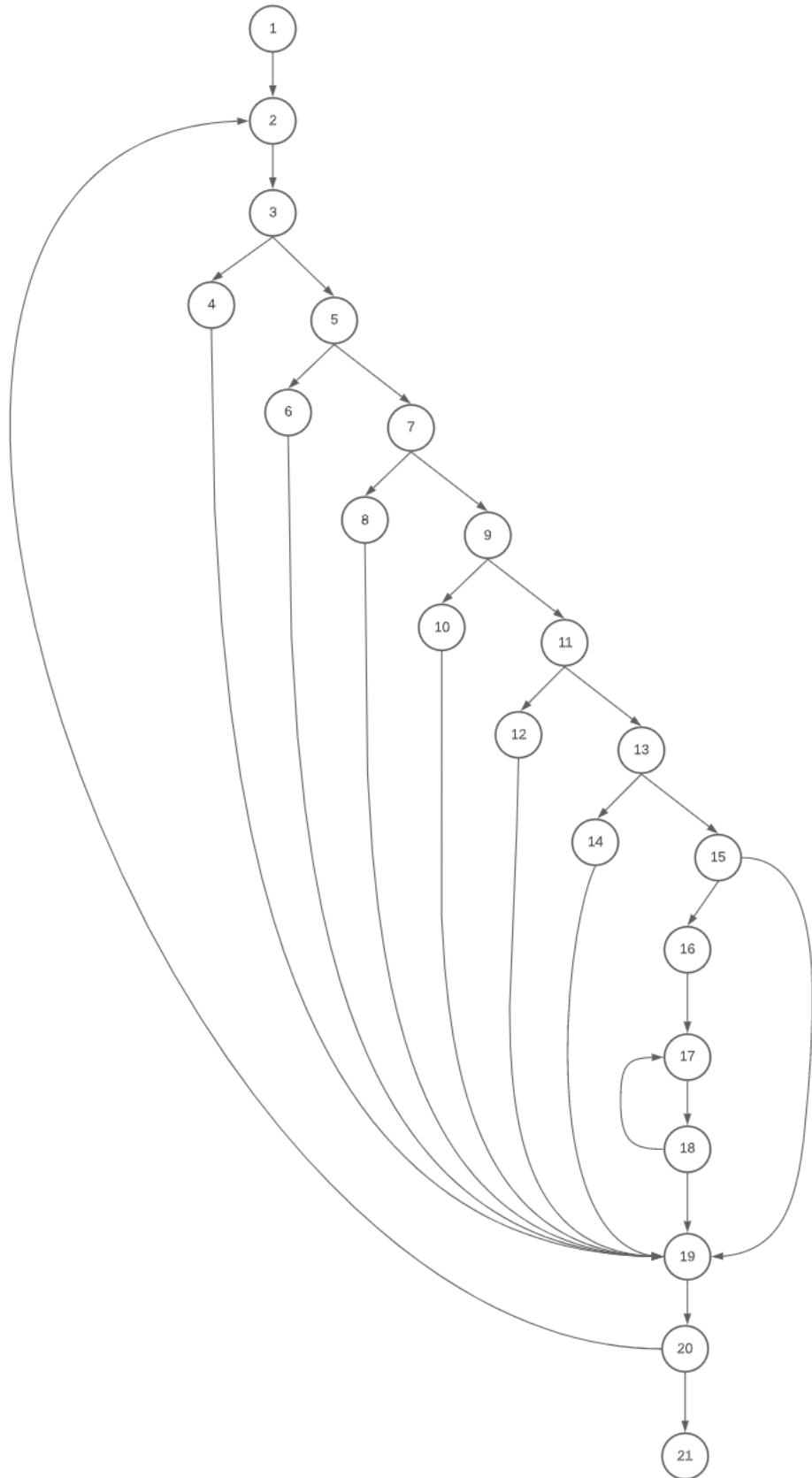
```

        cout<<"My Account"<<endl;
        account();
    }
13. else if(ch2 == 6)
14. {
    cout<<"Payments & Refunds"<<endl;
    payment();
    refurnd();
}
15. else if(ch2 == 7)
16. {
    cout<<"Order History"<<endl;
17. do
{
    cout<<"Please enter the date for which you want to see the order
history"<<endl;
    cout<<"Enter day(dd): ";
    cin>>day;
    cout<<"Enter month(mm): ";
    cin>>month;
    cout<<"Enter year(yyyy): ";
    cin>>year;
    cout<<"Your order history for the date "<<date<<"-"<<month<<-
"<<year<<" is: "<<endl;
    History(date, month, year);
    cout<<"Do you want to continue?(Y/N)"<<endl;
    cin>>ch3;
18. }while(ch3 == 'Y');
}
else
    cout<<"Wrong input!"<<endl;
19. cout<<"Do you want to browse more?(Y/N)"<<endl;
    cin>>ch1;
20. }while(ch1 == 'Y');
21. return 0;
}

```

8. TESTING

❖ Control Flow Graph



❖ Cyclomatic Complexity

1. $E = 30, N = 22$
 $V(G) = E - N + 2P$
 $= 29 - 21 + 2$
 $= 8+2$
 $= 10$
2. $V(G) = \pi + 1$
 $= 9 + 1$
 $= 10$
3. $V(G) = \text{Total Regions} = 10$

❖ Independent Paths

1. 1-2-3-4-19-20-21
2. 1-2-3-5-6-19-20-21
3. 1-2-3-5-7-8-19-20-21
4. 1-2-3-5-7-9-10-19-20-21
5. 1-2-3-5-7-9-11-12-19-20-21
6. 1-2-3-5-7-9-11-13-14-19-20-21
7. 1-2-3-5-7-9-11-13-15-16-17-18-17
8. 1-2-3-5-7-9-11-13-15-19-20-2
9. 1-2-3-5-7-9-11-13-15-19-20-21
10. 1-2-3-5-7-9-11-13-15-16-17-18-19-20-21

❖ Boundary Value Analysis

- The range for the input date:
1 <= day <= 31
1 <= month <= 12
2001 <= year <= 2021
- The expected output if date is correct is the Order History, and “Invalid date” if date entered is incorrect
- With Single fault assumption theory, $4n+1$ test case can be designed, which are equal to 13

- The Boundary Value Test cases are:

Test Cases	Day	Month	Year	Expected output
1	1	6	2010	Order History
2	2	6	2010	Order History
3	15	6	2010	Order History
4	30	6	2010	Order History
5	31	6	2010	Invalid Date
6	15	1	2010	Order History
7	15	2	2010	Order History
8	15	11	2010	Order History
9	15	12	2010	Order History
10	15	6	2001	Order History
11	15	6	2002	Order History
12	15	6	2020	Order History
13	15	6	2021	Order History