Software Requirements Specification

For

Food Ordering System

**Version 1.0 approved**

**Prepared by:**

**Hussain Mustafa (21k-3807)**

**Muhammad Kashif (21k-3818)**

**Ahsan Nadeem (21k-3803)**

**Jamaima Baig (21k-3831)**

**Suleiman Mohiuddin (21k-3887)**

**Zeeshan Mustafa (21k-3919)**

**14/12/2022**

# Table of Content

Table of Content ………………………………………………………………II

List of Figures……………………………………………………. III

List of Tables……………………………………………………………………IV

1. Introduction…………………………………………………………………….1

1.1 Purpose……………………………………………………………………..1

1.2 Product Scope………………………………………………………………

1.3 Feature Trees……………………………………………………………….

1.3.1 Customer Feature Tree…………………………………………..

1.3.2 Seller Feature Tree……………………………………………….

1.3.3 Delivery Feature Tree……………………………………………

1.4 Context Diagram…………………………………………………………..

1.5 Dialogue Map………………………………………………………………

1.6 Stakeholders………………………………………………………………..

1.7 Document Conventions…………………………………………………….

1.8 Intended Audience and Reading Suggestion……………………………….

2. Overall Description………………………………………………………………..

2.1 Product Perspective………………………………………………………….

2.2 Product Function………………………………………………………..

2.3 User Classes and Characteristics………………………………….

2.4 Operating Environment………………………………………………….

2.5 Design and Implementation Constraints………………………………..

2.6 User Documentation……………………………………………………

3. External Interface Requirement…………………………………………………

3.1 User Interfaces…………………………………………………………….

3.2 Hardware Interfaces and Software Interfaces…………………………..

3.3 Communications Interfaces……………………………………………….

4. Requirements……………………………………………………………………

4.1 Functional Requirements…………………………………………………..

4.1.1 Customer……………………………………………………….

4.1.2 Seller…………………………………………………………….

4.1.3 Delivery Module……………………………………………….

4.2 User Requirements…………………………………………………………..

4.2.1 Customer………………………………………………………..

4.2.2 Seller……………………………………………………………..

4.2.3 Delivery Module……………………………………………..

5. User Cases………………………………………………………………………

5.1 Actors……………………………………………………………………….

5.2 Use Case Diagram………………………………………………………….

5.3. User Case Description…………………………………………………

5.3.1 Customer………………………………………………………....

5.3.2 Seller……………………………………………………………..

5.3.3 Delivery Module…………………………………………….

6. Non-Functional Requirements………………………………………………….

7. Requirements Prioritization………………………………………………………

7.1 Prioritized Customer Functional Requirements…………………………..

7.2 Prioritized Seller Functional Requirements………………………………

7.3 Prioritized Delivery Functional Requirements………………………..

7.4 Prioritized Customer User Requirements…………………………….

7.5 Prioritized Seller User Requirements………………………………

7.6 Prioritized Delivery User Requirements…………………………………….

8. Data Requirements………………………………………………………………….

8.1 ERD Diagram……………………………………………………………..

8.2 Data Dictionary………………………………………………………

8.3 Report Specification…………………………………………

9. Business Rules……………………………………………………………

9.1 Facts……………………………………………………………

9.2 Constraints…………………………………………………….

9.3 Action Enabler………………………………………………………..

9.4 Inference……………………………………………………

9.5 Computations………………………………………………..

10. Business Requirements……………………………………………………..

Appendix A: Glossary……………………………………………………………..

Appendix B: I🡪 JRP Script……………………………………………………….

II🡪 Google Form……………………………………………………….

III🡪 Prototype…………………………………………………………..

a. Customer Prototype………………………….

b. Seller Prototype…………………………..

c. Delivery Prototype……………………………….

IV🡪 Swimlane………………………………………………………….

# List of Figures

Figure 1 Customer Feature Tree…………………………………………..

Figure 2 Seller Feature Tree………………………………………………

Figure 3 Delivery Feature Tree……………………………………

Figure 4 Context Diagram……………………………………

Figure 5 Dialogue Map…………………………….

Figure 6 Use Case Diagram…………………………………….

Figure 7 ERD Diagram……………………………………..

Figure 8 Swimlane Diagram………………………………………..

# List of Tables

Table 1 Customer Functional Requirements…..

Table 2 Seller Functional Requirements…..

Table 3 Delivery Module Functional Requirements…..

Table 4 Customer User Requirements…..

Table 5 Seller User Requirements…..

Table 6 Delivery Module User Requirements…..

Table 7 Log in Use Case Description…..

Table 8 Registration User Case Description…..

Table 9 Account Setting Use Case Description…..

Table 10 Customer Care Use Case Description…..

Table 11 Promotions User Case Description…..

Table 12 Track Order Use Case Description…..

Table 13 Place Order Use Case Description…..

Table 14 Add/ Delete Item Use Case Description…..

Table 15 Navigate Map Use Case Description…..

Table 16 Payment Use Case Description…..

Table 17 Cash on Delivery Use Case Description…..

Table 18 Card Use Case Description…..

Table 19 Registration Use Case Description…..

Table 20 Log in Use Case Description…..

Table 21 Sign Up Use Case Description…..

Table 22 Menu Use Case Description…..

Table 23 Menu Deals Use Case Description…..

Table 24 Managing Order Use Case Description…..

Table 25 Availability of Food Use Case Description…..

Table 26 Order History Use Case Description…..

Table 27 Order Status Use Case Description…..

Table 28 Navigate Map Use Case Description…..

Table 29 Track Payment Use Case Description…..

Table 30 Payment Option Use Case Description…..

Table 31 Automated Reporting Use Case Description…..

Table 32 Payment History Use Case Description…..

Table 33 Login Use Case Description…..

Table 34 Contact Customer Use Case Description…..

Table 35 Duty Service Use Case Description…..

Table 36 Order Details Use Case Description…..

Table 37 Payment/ Service History Use Case Description…..

Table 38 Data Dictionary…..

Table 39 Payment Reporting Report Specification…..

Table 40 Order Reporting Report Specification…..

Table 41 Delivered Reporting Report Specification…..

Table 42 Business Rules Fact…..

Table 42 Business Rules Constraints…..

Table 43 Business Rules Action Enabler…..

Table 44 Business Rules Inference…..

Table 45 Business Rules Computations……

# 1. Introduction

## The following section provides an overview of the derived Software Requirements Specification (SRS) for the subject Food Ordering System. To begin with, the purpose of the document is presented and its intended audience outlined. Subsequently, the scope of the project specified by the document is given with a particular focus on what the resultant software will do and the relevant benefits associated with it.

## 1.1. Purpose

For people of Pakistan who wants to easily order food from a wide range of restaurants from the comfort of their homes. **Food Swipe** is an internet-based application for smartphones and web-browsers, which will enable customer to order food from a wide range of restaurants from a single application, and allow restaurants to sell on a widely used platform. Unlike other companies like Food Panda there will be low service charges for seller, will allow customers to cancel order within 5 minutes and will provide exceptional customer service.

**1.2. Product Scope**

**Project Justification**

Unlike other apps, it is necessary to create a user operative app with largest market place through which they can easily access and order from multiple categories (Food Meal and Food Mart) at a pocket friendly rate with multiple payment option to suit them for their required needs from their comfort zone which will not save their time but money to. Earn points with every order the user places and redeem them for amazing rewards. Online shopping in Pakistan is risky but when it comes to our application you can choose in between different verified vendors. It also offers vendors to sell the products through our app with low commission charges.

**Summary of project deliverables**

* User Accessibility
* Simple and Hassle free Registration and Verification Process for Customers and Vendors
* Multiple choice of Sign-up Methods
* Catalogues of HORECA, Stores, and products
* Standardized Product Concept
* Promotions (Coupons, Sign-up Bonus, In-App Redeem Points)
* Quick Delivery at an Affordable Rate
* Powerful Admin Panel to manage all the user data
* Addictive User Interface (UI/UX Design)
* Scheduling of Order
* GPS Tracking in Real-Time
* Various Secure Payment Options
* Amazing Customer Care Service
* Rider Support (Chat and Contact Information)
* Push Notifications
* Personal Favorites
* Ratings and Reviews
* Star Reviewers Program
* Search Filters
* Assist with Virtual Work

**Statement or success Criteria**

The application will be considered a success if within a year after the initial launch it will have over 1 million downloads across all smartphone devices. The application should be used to order food 200,000 times across a week on average after 8 months of launch, and should amass a collection of at least 2000 sellers over the aforementioned period. The application should also have a rating above 4.5 stars on Google Play store and Apple store.

**Project Exclusion**

* Not every restaurant Open will be available in our app.
* Only accept card payment form partner banks.
* Can only deliver items that are available in the season.
* Can not directly chat with food delivery riders.
* Once a person is charged the payment can not be refunded.
* Will be using Google map services so cant take its guarantee.
* We do not directly control the prices of items

**Constraints**

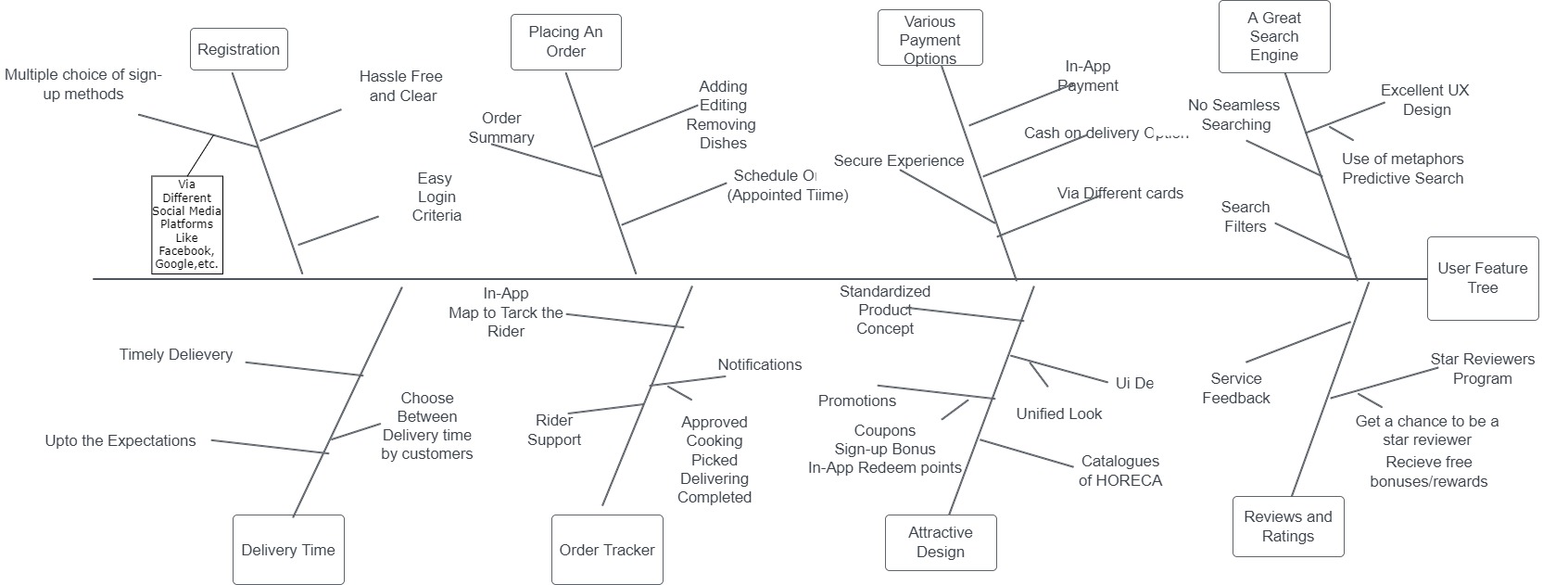
* Only a registered user can order
* Non-Registered users can view only the Restaurant and their menu
* Within 4-6 km our rider will be delivering the food
* The memory usage of the app will have to be constrained by the devices it is intended to run. Since most Tablets and mobile may have limited apps
* Internet enable is required
* The system would require permission for data storage
* The primary candidate tool chains are Java/Swing, C++/SQL
* The system must provide a capacity for parallel operation and system design should not introduce scalability issues about the number of surface computers, tablets, or displays connected at any one time
* If the system is down, then customers must not notice, or notice that the system recovers quickly (seconds)

**Assumptions**

* Food delivery is quite convenient and all raw materials are available
* Quality of food may not suffer
* Every order from a customer must be satisfied by exactly one delivery vehicle
* The transportation time between delivery centers and customers are fixed based upon distance (It is assumed that traffic conditions are always fixed)
* All the cars are maintained and are unlikely to breakdown
* No customer demand will exceed the capacity of a transportation vehicle
* The operating hours for the online system are as the business operation hours which is from 8:00 AM to 12:00 PM everyday
* The operation depends on the changed being made by the restaurant regarding available food and delivery.

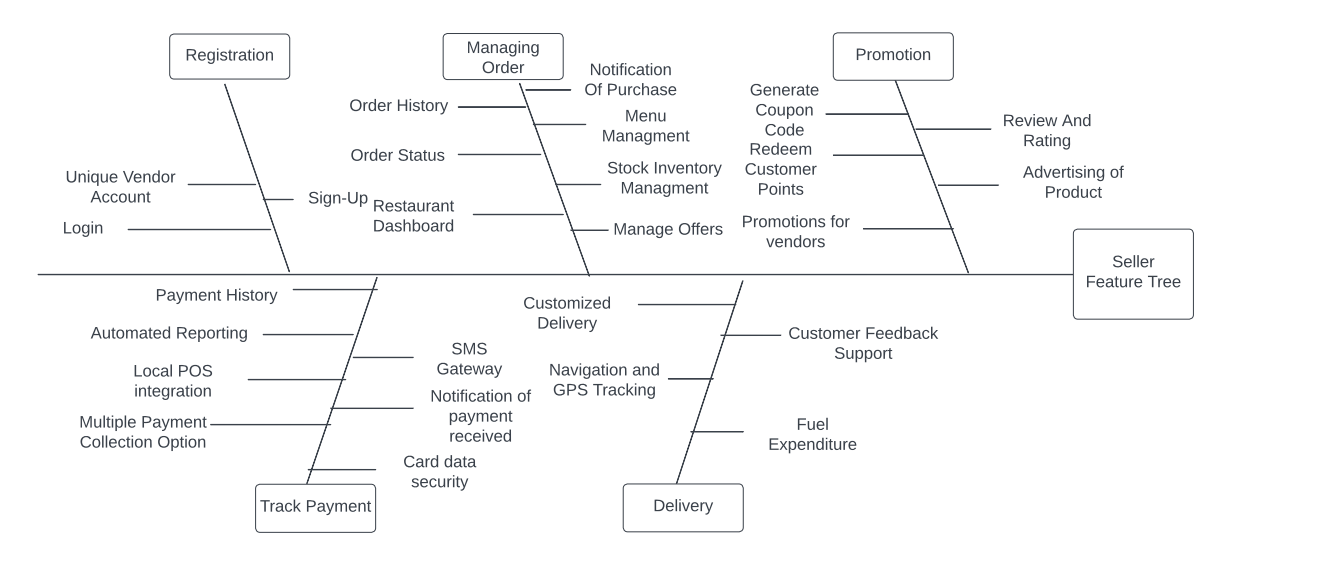
**1.3 Feature Trees**

1. **Customer Feature Tree:**



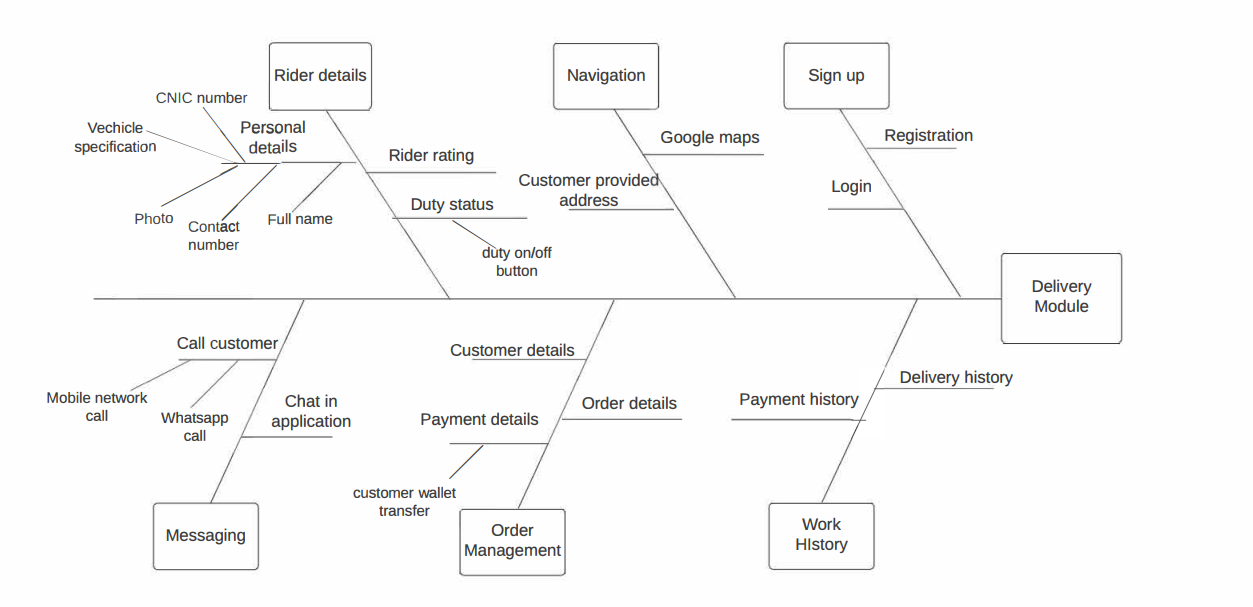
**Figure 1**

1. **Seller Feature Tree:**

****

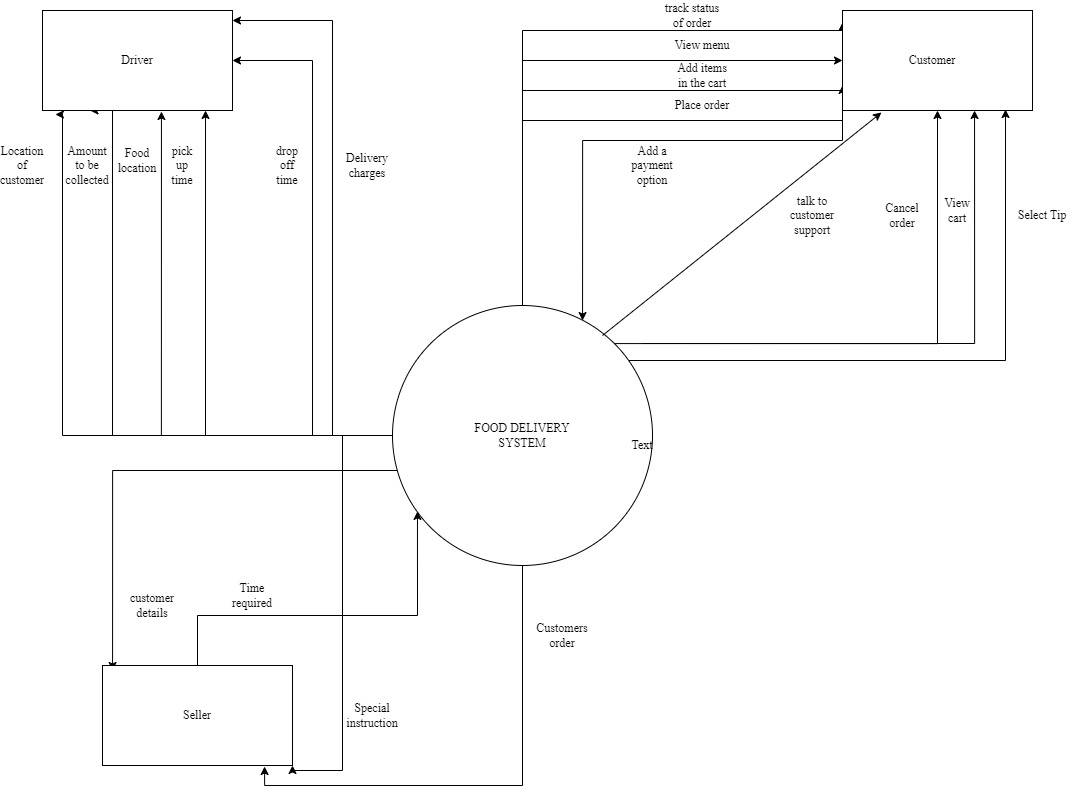
**Figure 2**

1. **Delivery Feature Tree:**

****

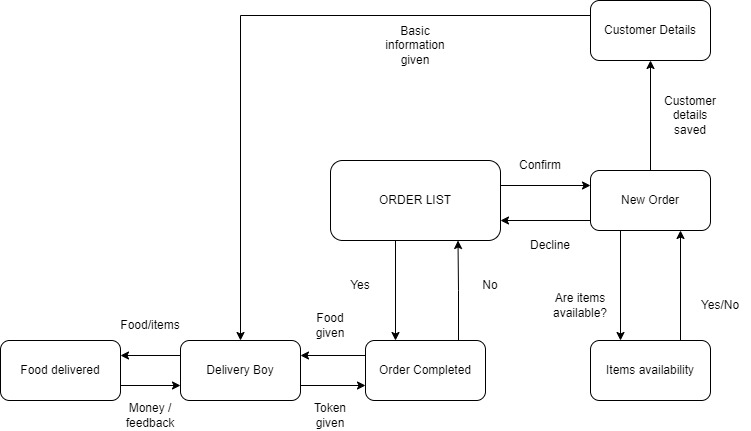
**Figure 3**

**1.4 Context Diagram**

****

**Figure 4**

**1.5 Dialogue MAP**

****

**Figure 5**

**1.6. Stakeholders**

|  |
| --- |
| Customer |
| Project Manager |
| Business Analyst |
| Application Architect |
| Designer |
| Developer |
| Product Owner |
| Data Modeler |
| Process Analyst |
| Tester |
| Product Manager |
| Quality Assurance Staff |
| Documentation Writer |
| Database Administrator |
| Hardware Engineer |
| Infrastructure Analyst |
| Business Solution Architect |
| Seller |
| Delivery Man |
| Banking Companies( Visa, Easypaisa etc) |
| FBR |
| Government Agency |
| Share Holders |
| Hosting Servers |
| Development Manager |

|  |
| --- |
| Marketing |
| Operational Support Staff |
| Legal Staff |
| Information Architect |
| Company Owner |
| Sales Staff |
| Installer |
| Maintainer |
| Program Manager |
| Usability Expert |
| Subject Matter Expert |
| Executive Sponsor |
| Project Management Office |
| Manufacturing |
| Training Staff |
| Portfolio Architect |
| Infrastructure Support Staff |
| Acquirer |
| Procurement Staff |
| Contractor |
| Subcontractor |
| Business Management |
| Program Manager |
| Beta Tester |
| General Public |

|  |
| --- |
| Consultant |
| Compliance Auditor |
| Certifier |
| Regulatory Body |
| Software Supplier |
| Material Supplier |
| Venture Capitalist |
| Market Research Specialist |
| Lawyers |
| Product Champion |
| Session Leader |
| Analyst |
| Scribe |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

**1.7. Document Conventions**

The requirements contained with this document will be arranged according to a timeline order. This document was created based on IEEE template for system requirements specification documents.

**1.8. Intended Audience and Reading Suggestion**

The primary audience of this SRS document will be the development team employed to implement the specified food ordering system. It will not only provide an extensive capacity for project planning and progress assessment but it will further assist with stakeholder interactions. The secondary document audience comprises the stakeholders of the project, that is, sellers and associated staff. To this audience group, this SRS should convey and confirm the required functionality and represent a contractual agreement between the involved parties

**2. Overall Description**

**2.1. Product Perspective**

The software described in this SRS is the software for a complete food ordering system. The system merges various hardware and software elements and further interfaces with external systems. it relies on a number of external interfaces for persistence and unhandled tasks, as well as physically interfacing with humans.

**2.2. Product Functions**

CUSTOMER:

1. Registration

* Log in
* Sign up

1. Browse through different menus and deals along rates
2. Adjust account settings
3. Customer care helpline
4. Place an order
5. Track order by navigating through Google map
6. Payment

* Credit card
* Cash on delivery

1. Report a problem

SELLER:

1. Registration

* Log in
* Sign up

1. Managing orders

* Check availability of food
* Order history
* Order status

1. Menu management

* Manage offers

1. Track payment

* Payment history
* Automated reporting
* Payment option

1. Track dispatched order by navigating through Google maps
2. Promotions and deals management

* Generate coupon code
* Redeem customer points
* Review and rating

DELIVERY MAN:

1. Login through user profile
2. Contact customer
3. View order details
4. View drop off location
5. Work and payment history
6. Turn duty status on and off
7. Transfer money to customer wallet

**2.3. User Classes and Characteristics**

**2.4. Operating Environment**

Hardware

* Mobile with wifi and Gps
* Dual-core processor.
* Touch screen.

Software

* Android 9 + /windows 10 +11.

Any browser

**2.5. Design and Implementation Constraints**

* Will only work for people with a compatible smart phone with Wi-Fi and GPS services available.
* Person must be living in a Urban area.
* Can only deliver if the restaurant owner agrees.
* Must invest in high quality servers to keep app always active and smooth running.
* Must agree to terms and conditions.

**2.6. User Documentation**

An installation guide will be provided. Any user of the Food Swipe System is the target audience for user documentation generated about the software system. A range of short document types (e.g., guidelines, tutorials, frequently asked questions) in Hyper Text Markup Language (HTML) and/or Portable Document Format (PDF) format must describe the use of the software system.

**3. External Interface Requirements**

**3.1. User Interfaces**

However, the app would maintain consistency and follow some standards:

* There shall be a fixed menu bar at the top with following buttons (All, Budget, Deals/discounts/Rating)
* There should be fixed drop down menu pointer at top left with following options (Profile, Help, Settings, Logout)
* On clicking the logo, the system shall return to Home Page
* There should be a “Contact us” and “Logout” buttons at bottom
* The layout should be simple and easy to use
* The layout should provide maximum user experience
* In case of any errors, it should provide proper guidelines to resolve

**3.2. Hardware Interfaces and Software Interfaces**

We would require following technology for development of the app:

* GPS receiver: indicates user location
* MySql database
* JavaScript and React for front end
* PHP for backend
* Any framework such as Angular and server such as Apache.
* A cache such as Memcached would be required to store data.
* Cloud can be used for backup and retrieval of data

**3.3. Communications Interfaces**

The system will use following communication interfaces:

* Emails
* Social media
* Text communication
* Protocols would be required for secure communication and message encryption

**4. Requirements**

The following section presents the complete set of functional and user requirements identified for the subject Food Delivery System. Functional requirements and user requirements are listed, according to their relationship to the overall system, customer, and seller and delivery man.

**4.1. Functional Requirements**

This subsection presents the identified functional requirements for the subject Food Ordering System. Requirements have been demarcated based on their relevance to the users of the system that is, customer, and seller and delivery man.

**4.1.1. Customer**

Table presents the identified functional customer requirements that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirement** | **Description** |
| FRC01 | The customer shall be able to Sign Up into the system by entering the user name, password, email-id, phone number and CNIC number. |
| FRC02 | User name and password should be mailed to the respective email-id by the system. |
| FRC03 | The customer shall be able to Login in into the system by entering the email-id and password. |
| FRC04 | While retrieving the password (if forget) system should provide the username and email-id. |
| FRC05 | If the customer enters wrong email-id and password into the system the system provides 3 opportunities for correct email-id and password otherwise the system terminates. |
| FRC06 | The customer can search the restaurants which are registered through the application. |
| FRC07 | The customer cannot order if the restaurant is closed, the system terminates the order and the user cancels the order. |
| FRC08 | System gives customer 3 opportunities for correct OTP code otherwise the system terminates the order. |
| FRC09 | If the customer need to pay avail the online payment through banking system, the customer can select the most convenient method to pay through. |
| FRC10 | The customer shall be able to engage bill mode to finalize payment through their engaged menu |
| FRC11 | If the customer enters the wrong card number the system will generate an error message of invalid card number and it terminates the order. |
| FRC12 | System gives customer 3 opportunities for correct CNIC otherwise the system generates an error message of invalid CNIC number and it terminates the order. |
| FRC13 | The Customer can schedule the order through the timer given in the application |
| FRC14 | The customer can modify the order through the system. |
| FRC15 | The customer can cancel the order before the time allocated by the system. |
| FRC16 | The customer shall be able to navigate the rider through the GPS system. |
| FRC17 | The customer shall be able to add and remove the item from the basket. |
| FRC18 | The customer can avail the discount coupons after completing the multiple milestones which need to be accomplished |

**Table 1**

**4.1.2. Seller**

Table presents the identified functional seller requirements that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| FRS01 | The seller shall be able to Sign Up into the system by entering the user name, password, email-id, phone number and CNIC number. |
| FRS02 | User name and password should be mailed to the respective email-id by the system. |
| FRS03 | The seller shall be able to Login in into the system by entering the email-id and password. |
| FRS04 | While retrieving the password (if forget) system should provide the username and email-id. |
| FRS05 | Food brand requests to create menu for specified date, if the food brand requests to cancel menu for the specified date the system terminates the use case. |
| FRS06 | The seller can check the history and order status in the previous order history. |
| FRS07 | The seller can generate the coupon code after seeing the frequency of a particular customer ordering from his store. |
| FRS08 | The seller will get the notification from the customer about the order and then seller will notify the order status to the rider. |

**Table 2**

**4.1.3. Delivery Module**

Table presents the identified functional delivery module requirements that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| FRD01 | The rider shall be able to Sign Up into the system by entering the user name, password, email-id, phone number and CNIC number. |
| FRD02 | User name and password should be mailed to the respective email-id by the system. |
| FRD03 | The rider shall be able to Login in into the system by entering the email-id and password. |
| FRD04 | While retrieving the password (if forget) system should provide the username and email-id. |
| FRD05 | The rider gets the notification via SMS from the seller about the order status. |
| FRD06 | The rider shall be able to access the location of the customer and the seller location. |
| FRD07 | The rider shall be able to message or call the customer or the seller through the messaging system. |

**Table 3**

**4.2. User Requirements**

This subsection presents the identified user requirements for the subject Food Ordering System. Requirements have been demarcated based on their relevance to the users of the system that is, customer, and seller and delivery man.

**4.2.1. Customer**

Table presents the identified user customer requirements that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| URC01 | I should be able to register to the food ordering system application. |
| URC02 | I should be able to login to my profile in the application. |
| URC03 | I should be able to select the location I want my food to be delivered to |
| URC04 | I should be able to browse the restaurant which delivers to my location through a list. |
| URC05 | I should be able to search restaurant in registered on the application |
| URC06 | I should be able to view the menu of a selected restaurant by clicking on the restaurant’s entry from the list. |
| URC07 | I should be able to filter the restaurants list according to the type of food I want to order through the application. |
| URC08 | I should be able to filter the restaurants according to their time of delivery to my specified address |
| URC09 | I should be able to select items from the menu of a selected restaurant |
| URC10 | I should be able to specify the quantity of the item I am ordering. |
| URC11 | I should be able to navigate to the cart window and view my entire order |
| URC12 | I should be able to able to order the items to be delivered as soon as possible to my specified address. |
| URC13 | I should be able to schedule the time I want the order to reach my address |
| URC14 | I should be able to pay using any credit or debit card. |
| URC15 | I should be able to pay using Easypaisa |
| URC16 | I should be able to pay using JazzCash. |
| URC17 | I should be able to pay when the delivery person arrives at my address. (Cash on delivery). |
| URC18 | I should be able to track the delivery person carrying my order on Google maps. |
| URC19 | I should be able to contact the delivery person carrying my order through a chat section in the application. |
| URC20 | I should be able to select contactless delivery |
| URC21 | I should confirm my delivery address before confirming order |
| URC22 | I should be able to save multiple addresses in the application |
| URC23 | I should be able to leave a review from the restaurant I have ordered |
| URC24 | I should be able to leave a review to the delivery person who has delivered the order to my specified address. |
| URC25 | I should be able to seek help concerning my order through the application. |
| URC26 | I should be able to lodge a complaint about the food I have received through the application |
| URC27 | I should be able to lodge a complaint about the delivery person who has delivered the food to my specified address. |
| URC28 | I should be able to update my contact details in my profile in the application. |
| URC29 | I should be able to update credit/debit card number in the application |
| URC30 | I should be able to delete credit/debit card information from the application if I choose to do so. |

**Table 4**

**4.2.2. Seller**

Table presents the identified user seller requirements that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| URS01 | I should be able to register as a vendor on the application. |
| URS02 | I should be able to login to my vendor account on the application |
| URS03 | I should be able to set the menu:   * Set names of different food * Add pictures for every item * Add description for every item * Add options for different variants of a cuisine |
| URS04 | I should be able to manage offers and discounts   * + Generate discount codes   + Generate deals   + Generate sales on the whole menu |
| URS05 | I should be able to view my order history |
| URS06 | I should be able to view a list of all active orders |
| URS07 | I should be able to view the reviews given to me by the customers |
| URS08 | I should be able to track the delivery person of a particular order as he is delivering it. |
| URS09 | I should be able to view my payment history |
| URS10 | I should be able to food preparation time for my restaurants |
| URS11 | I should be able to add branches of my restaurants   * + I should be able to manage the menu of each brand individually   + I should be able to manage the food preparation time of each branch individually |
| URS12 | I should be able to receive live complaints about the food if any |
| URS13 | I should be able to provide special delivery instructions to the delivery person through the application. |

**Table 5**

**4.2.3. Delivery Module**

Table presents the identified user delivery requirements that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| URD01 | I should be able to register myself as a user after receiving confirmation from an opportunity center of my registration |
| URD02 | I should be able to login to the application’s delivery module by using my username and password. |
| URD03 | I should be able to see the address of the customer to whom I have to deliver food in text format as provided by the customer |
| URD04 | I should be able to see a location pin at the location of delivery on Google maps. |
| URD05 | I should be able to see the name and phone number of the customer to whom I have to deliver food |
| URD06 | I should be able to view the details of the order I have to deliver. |
| URD07 | I should be able to view the payment method the customer to whom I have selected when ordering food. |
| URD08 | I should be able to add money to the virtual wallet of the customer through the application. |
| URD09 | I should be able to chat with a customer using the application. |
| URD10 | I should be able to call the customer through mobile networks by redirection from the application. |
| URD11 | I should be able to call the customer through WhatsApp by redirection from the application |
| URD12 | I should be able to view the rating customers leave for me in the application. |
| URD13 | I should be able to view and update my personal details   * + Should be able to view CNIC number   + Should be able to view and update contact number   + Should be able to view and update profile picture   + Should be able to view and update vehicle registration number   + Should be able to view full name |
| URD14 | I should be able to toggle my duty status between on and off. |
| URD15 | I should be able to view to my work history |
| URD16 | I should be able to view my work frequency |
| URD17 | I should be able to view my payment history |

**Table 6**

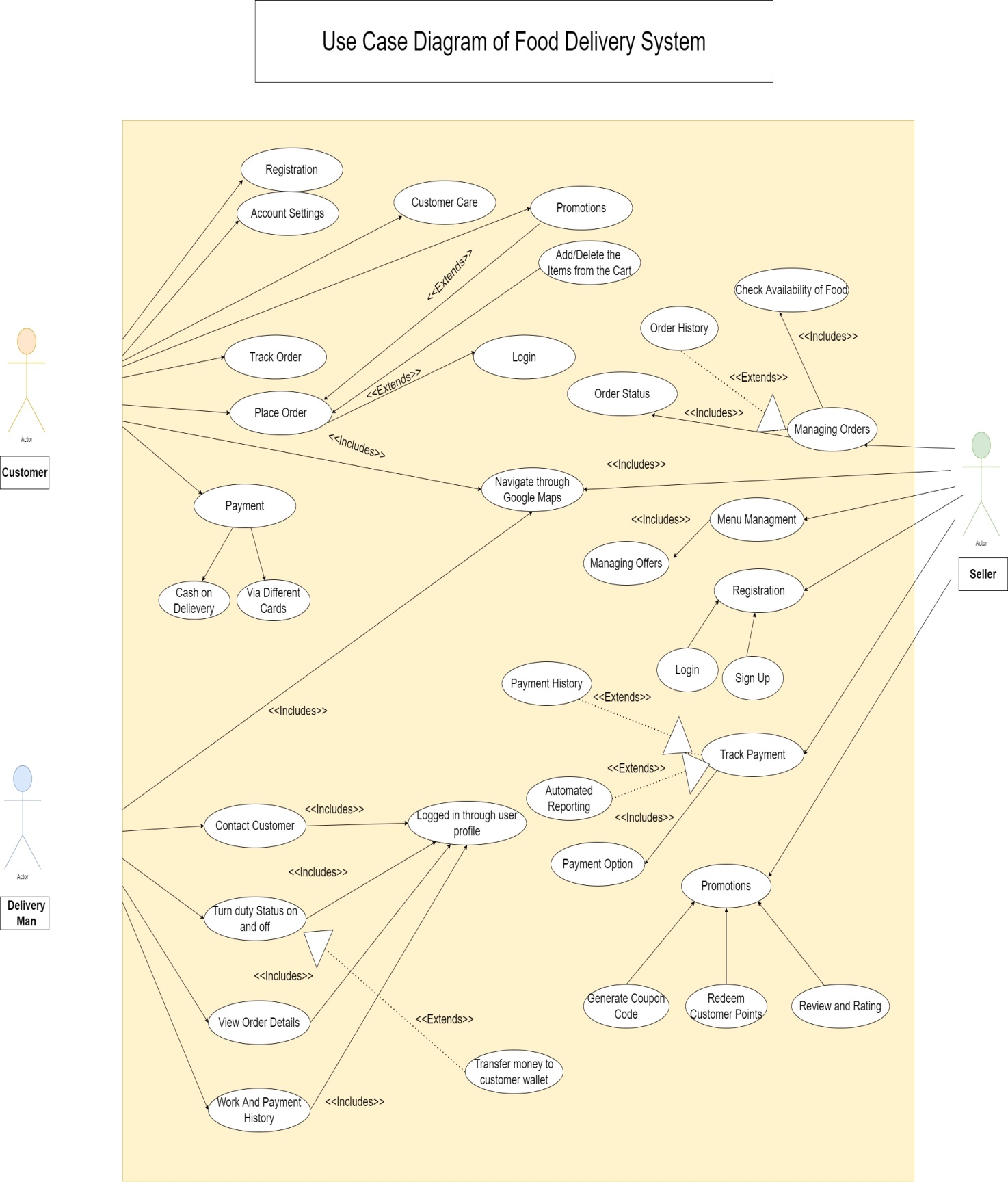
**5. Use Cases**

This subsection extends upon the functional requirements given in Section 4 through the presentation of detailed use cases. To facilitate an unambiguous and clear view of how the end-users interact with the subject Food Ordering System, the actors (end-users) involved in the use cases, a use case diagram and detailed use case descriptions are provided.

**5.1. Actors**

There are three actors in the food ordering system, Customer, Seller, Delivery Man. The primary actors are Customer and Delivery Man where as the secondary actors is Seller.

**5.2. Use Case Diagram**

****

**Figure 6**

**5.3. Use Case Description**

**5.3.1. Customer**

Table presents the Log In use case description to show the interaction between a customer and a application when logging into the system.

|  |  |
| --- | --- |
| Use Case | Log In |
| Primary Actor | Customer |
| Goals In Context | Enable customers to order the item from our system |
| Pre-Conditions | The user has a valid name and password to login the system |
| Trigger | It came when the user opens the system the first time and also when he wanted to order. |
| Description | This use case describes the event when the customer wanted to order something or he/she had to access the system for the first time. Then customer will enter their name and password if he/she is already registered. Then customers are enabled access to order items. |
| Exceptions | A user id and Password is invalid |

**Table 7**

Table presents the Registration use case description to show the interaction between a customer and a application when to enable customer to order the item from our system.

|  |  |
| --- | --- |
| Use Case | Registration |
| Primary Actor | Customer |
| Goals In Context | Enable customers to order the item from our system as well as to get all the necessary information about the customer |
| Pre-Conditions | The user is using the system for the first system |
| Trigger | It came when the user opens the system for the first time |
| Description | This use case describes the event when the customer is accessing the system for the first time. It will ask for all the details about the customer i.e.   1. Name\* 2. Number\* 3. Address 4. Email 5. New Password and Confirm Password\* |
| Exceptions | A user does not enter the steric detail |
|  |  |
|  |  |

**Table 8**

Table presents the Account Setting use case description to show the interaction between a customer and a application when enable customer to update the details of their account.

|  |  |
| --- | --- |
| Use Case | Account Setting |
| Primary Actor | Customer |
| Goals In Context | Enable customers to update details about his/her account |
| Pre-Conditions | The user is login to the system |
| Trigger | It came when the user wanted to update any of his/her details. |
| Description | This use case describes the event when the customer wanted to update his /her details which are given to the system when he is registering on the system. The following options will came   1. Address 2. Number 3. Email 4. Card Detail |
| Exceptions | If the customer leaves it blank |

**Table 9**

Table presents the Customer Care use case description to show the interaction between a customer and a application when enable customers to get assistance from care person.

|  |  |
| --- | --- |
| Use Case | Customer Care |
| Primary Actor | Customer |
| Goals In Context | Enable customers to get any assistance from the care person |
| Pre-Conditions | The customer is logged in to the system |
| Trigger | It came when the user goes to the Customer Care option and taps on it |
| Description | This use case describes the event when the customer wanted to get help from the Care Person. It is a chatbot as well as a caring person if a customer’s detail is not answered by the bot he/she has the option to get assistance from the care person. This care option is used by the customer for many reasons like   1. To get help 2. Having Difficulty 3. Report the Problem 4. About Delivery / Order |
| Exceptions | A Customer Care person is not available |

**Table 10**

Table presents the Promotions use case description to show the interaction between a customer and a application when enable customer to view Promo Code.

|  |  |
| --- | --- |
| Use Case | Promotions |
| Primary Actor | Customer |
| Goals In Context | Enable Customers to watch all exciting Deals as well as Promo Code for heavy discounts |
| Pre-Conditions | The user is logged in to the system |
| Trigger | It came when the user goes to the Promotion feature and taps on it |
| Description | This use case describes the event when the customer wanted to see all the deals which are offering by the Restaurant as well as to get Promo Code that is given by the system to their loyal customers. They can redeem the voucher as well as they can also redeem the deals by just tapping on it. |
| Exceptions | A user id and Password is invalid |

**Table 11**

Table presents the Track Order use case description to show the interaction between a customer and a application when enable customer to track order.

|  |  |
| --- | --- |
| Use Case | Track Order |
| Primary Actor | Customer |
| Goals In Context | Enable customers to track the order |
| Pre-Conditions | The user had ordered an item |
| Trigger | It came when the user ordered the item and it is confirmed by the system |
| Description | This use case describes the event when the customer wanted to track his/her order which they had ordered by our system. It has 3 options which were shown on the tracking   1. Confirming 2. Preparing 3. On the way |
| Exceptions | If there is no order made by the customer |

**Table 12**

Table presents the Place Order use case description to show the interaction between a customer and a application when enable customer to place order.

|  |  |
| --- | --- |
| Use Case | Place order |
| Primary Actor | Customer |
| Goals In Context | Enable customers to order the item from our system |
| Pre-Conditions | The user is logged in to the system |
| Trigger | It came when the user had added to the cart all the items |
| Description | This use case describes the event when the customer wanted to order something from the customer. And he/she had added items to the cart and given all the necessary information |
| Reference | Taking Information from the following use case  Log In  Add/Delete Item  Promotions |
| Exceptions | If there is nothing in the cart |

**Table 13**

Table presents the Add/Delete Item use case description to show the interaction between a customer and a application when enable customer to update their cart.

|  |  |
| --- | --- |
| Use Case | Add/Delete Item |
| Primary Actor | Customer |
| Goals In Context | Enable customers to update their Cart |
| Pre-Conditions | The user is logged in to the system  For deletion, the user has added an item. |
| Trigger | It came when the user added an item or wanted it o add to their cart |
| Description | This use case describes the event when the customer wanted to order food from the restaurant. So they must have added something to their cart. If he/she had added an item but wanted to remove the item so there is an option for it as well. |
| Exceptions | The user is not logged in |

**Table 14**

Table presents the Navigate Map use case description to show the interaction between a customer and a application when enable customer to view the location of delivery man.

|  |  |
| --- | --- |
| Use Case | Navigate Map |
| Primary Actor | Customer |
| Goals In Context | Enable customers to see the delivery guy |
| Pre-Conditions | The user has ordered food |
| Trigger | It came when the delivery guy picked up the food |
| Description | This use case describes the event when the customer wanted to see the delivery guy that when it will reach him and how much distance he is from him. |
| Exceptions | If a Delivery guy has not picked up the food |

**Table 15**

Table presents the Payment use case description to show the interaction between a customer and a application when enable customer to select payment option.

|  |  |
| --- | --- |
| Use Case | Payment |
| Primary Actor | Customer |
| Goals In Context | Enable customers to select payment option |
| Pre-Conditions | The user has added the items to the cart |
| Trigger | It came when the user is confirming the items |
| Description | This use case describes the event when the customer wanted to order something and when he/she is just confirming the order and the payment options come to them to select any one of them |
| Reference | Ref to the following  COD  CARD |
| Exceptions | If the user has not confirmed the items |

**Table 16**

Table presents the COD ( Cash On Delivery) use case description to show the interaction between a customer and a application when enable customer to pay through cash on delivery.

|  |  |
| --- | --- |
| Use Case | COD(Cash On Delivery) |
| Primary Actor | Customer |
| Goals In Context | Enable customer to select the suitable payment option |
| Pre-Conditions | The user confirmed the items |
| Trigger | It came when the user select the payment mode |
| Description | This use case describes the event when the customer wanted to order something from the system and select the payment option and when the mode of payment came he/she select the COD option which means they will pay for the food via cash to the delivery guy when he arrives to their place. |
| Exceptions | The payment option is not selected |

**Table 17**

Table presents the Card use case description to show the interaction between a customer and a application when enable customer to pay through card.

|  |  |
| --- | --- |
| Use Case | Card |
| Primary Actor | Customer |
| Goals In Context | Enable customer to select the suitable payment option |
| Pre-Conditions | The user confirmed the items |
| Trigger | It came when the user select the payment mode |
| Description | This use case describes the event when the customer wanted to order something from the system and select the payment option and when the mode of payment came he/she select the card option which means that they will pay for their food in advance via account card. |
| Exceptions | The payment option is not selected |

**Table 18**

**5.3.2. Seller**

Table presents the Registration use case description to show the interaction between a Manager and a application when enable to get order from the system.

|  |  |
| --- | --- |
| Use Case | Registration |
| Primary Actor | Manager |
| Goals In Context | Enable manager to get orders from the system once the user ordered from their Restaurant |
| Pre-Conditions | The Manager wanted to register their Restaurant |
| Trigger | It came when the manager is registering their restaurant for the first time |
| Description | This use case describes the event when the manager is accessing the system for the first time. It will ask for all the details about the customer i.e.   1. Name\* 2. Number\* 3. Address\* 4. Type of Food Offering\* 5. Price Range\* 6. Health Certificate\* 7. Tax Payer\* 8. Commission Contract Acceptance\* |
| Reference | Ref to following  Log In  Sign up |
| Exceptions | A user does not enter the steric detail |

**Table 19**

Table presents the Log In use case description to show the interaction between a Manager and a application when enable the user to log in.

|  |  |
| --- | --- |
| Use Case | Log In |
| Primary Actor | Manager |
| Goals In Context | Enable manager to get orders from the system once the user ordered from their Restaurant |
| Pre-Conditions | The manager has a valid name and password to login the system |
| Trigger | It came when the manager wanted to open his/her shop on the system |
| Description | This use case describes the event when the manager wanted to accept the order from their customers via our system so he/she will log in to our system. |
| Exceptions | A user id and Password is invalid |

**Table 20**

Table presents the Sign Up use case description to show the interaction between a Manager and a application when enable the user to sign up.

|  |  |
| --- | --- |
| Use Case | Sign Up |
| Primary Actor | Manager |
| Goals In Context | Enable manager to get orders from the system once the user ordered from their Restaurant |
| Pre-Conditions | The user is using the system for the first system |
| Trigger | It came when the manager once register their Restaurant to our system |
| Description | This use case describes the event when the manager is accessing the system for the first time. It will ask the following so that they can log in to the system i.e.   1. Name\* 2. Number\* 3. Email 4. New Password and Confirm Password\* |
| Exceptions | A user does not enter the steric detail |

**Table 21**

Table presents the Menu use case description to show the interaction between a Manager and a application when enable to manage all the items offered in their menu.

|  |  |
| --- | --- |
| Use Case | Menu |
| Primary Actor | Manager |
| Goals In Context | Enable managers to manage all the items offered in their menu |
| Pre-Conditions | The manager is logged in to the system |
| Trigger | It came when the manager wanted to update the menu |
| Description | This use case describes the event when the manager wanted to manage the menu for example if any item is out of stock or finished so they remove it from the menu or if any new item is introduced by the system they can add it to their menu. Also if there is no menu they can enter their menu for the first time. |
| Reference | Ref to  Menu Deals |
| Exceptions | A manager is not logged in to the system |

**Table 22**

Table presents the Menu Deals use case description to show the interaction between a Manager and a application when enable to manage deals in the menu.

|  |  |
| --- | --- |
| Use Case | Menu Deals |
| Primary Actor | Manager |
| Goals In Context | Enable managers to manage deals in the menu. |
| Pre-Conditions | A menu already exists |
| Trigger | It came when the manager tap on the offer |
| Description | This use case describes the event when the manager wanted to give promotion deals to their customers. For example, giving an offer of Buy 1 Get 1 Free or Giving a free sample on Purchasing an Item |
| Exceptions | A menu should exist |

**Table 23**

Table presents the Managing Order use case description to show the interaction between a Manager and a application when enable to manage the orders received by the customer.

|  |  |
| --- | --- |
| Use Case | Managing Order |
| Primary Actor | Manager |
| Goals In Context | Enable managers to manage all the orders received by the customer |
| Pre-Conditions | There should be an order received |
| Trigger | When an order received |
| Description | This use case describes the event when the manager wanted to manage the order which is received by the customer for the reasons to accept or decline the order or to see the previous order history and the status of the order whether it is delivered or not. |
| Reference | Ref to the following use case   1. Order Status 2. Order History 3. Availability of Food |
| Exceptions | If there is no order placed by a single customer |

**Table 24**

Table presents the Availability Of Food use case description to show the interaction between a Manager and a application when enable to see whether the received order is available.

|  |  |
| --- | --- |
| Use Case | Availability of Food |
| Primary Actor | Manager |
| Goals In Context | Enable managers to see whether the received order food is available or not |
| Pre-Conditions | There should be an order received |
| Trigger | When an order received |
| Description | This use case describes the event when the manager wanted to see the availability of food that is received by the order of the customer for the reasons to accept or decline the order. If the received order is not available which is asked by the customer, then you will reject the order. But this act will affect the capability of the restaurant as the customer will not be satisfied due to the non-availability of food |
| Exceptions | If there is no order placed by a single customer |

**Table 25**

Table presents the Order History use case description to show the interaction between a Manager and a application when enable to see the previous orders.

|  |  |
| --- | --- |
| Use Case | Order History |
| Primary Actor | Manager |
| Goals In Context | Enable managers to see all the previous orders received by the customer |
| Pre-Conditions | There should be an order received |
| Trigger | When the manager taps on the manage history |
| Description | This use case describes the event when the manager wanted to manage the order history which is received by the customer the reason to just wanted to see the rating of the food given by the customer as well as to see the most ordered dishes so that they can come with a strategy for which our customer care would also help with them so that they get most of it and there will be multiple more reasons for the manager to see the history. |
| Exceptions | If there is no order placed by a single customer |

**Table 26**

Table presents the Order Status use case description to show the interaction between a Manager and a application when enable to check the status of food.

|  |  |
| --- | --- |
| Use Case | Order Status |
| Primary Actor | Manager |
| Goals In Context | Enable managers to check the status of the food |
| Pre-Conditions | There should be an order received |
| Trigger | When an order received |
| Description | This use case describes the event when the manager wanted to check the status of the order whether is prepared or not as well if it was prepared by the restaurant and whether it is picked up by the delivery guy or not as well to see if it is delivered or not |
| Exceptions | If there is no order placed |

**Table 27**

Table presents the Navigate Map use case description to show the interaction between a Manager and a application when enable to track status of delivery.

|  |  |
| --- | --- |
| Use Case | Navigate Map |
| Primary Actor | Manager |
| Goals In Context | Enable manager to track the status of delivery |
| Pre-Conditions | The order is prepared by the restaurant |
| Trigger | When the delivery guy picked up the food |
| Description | This use case describes the event when the manager wanted to see the delivery guy when it will reach the destination and how far is he from the destination. |
| Exceptions | If the food is not ready |

**Table 28**

Table presents the Track Payment use case description to show the interaction between a Manager and a application when enable to track payment.

|  |  |
| --- | --- |
| Use Case | Track Payment |
| Primary Actor | Manager |
| Goals In Context | Enable managers to track the payment whether received or not. |
| Pre-Conditions | An order is received at the restaurant |
| Trigger | When the food is successfully delivered to the destination |
| Description | This use case describes the event when the manager wanted to check the payment tracking of the restaurant as the restaurant will not directly receive any payment by the customer because if it is a card payment then it will be received to the account of our system than at the end of the month we will deduct our service charges and then we will give to the restaurant same with the cash option that the delivery guy will receive the payment it will submit to us then we will deduct and give back to them. |
| Reference | Ref to the following use cases   1. Payment Option 2. Payment History 3. Automated Reporting |
| Exceptions | If there is no order receive |

**Table 29**

Table presents the Payment Option use case description to show the interaction between a Manager and a application when enable to see the payment details.

|  |  |
| --- | --- |
| Use Case | Payment Option |
| Primary Actor | Manager |
| Goals In Context | Enable managers to see the details of the payment which is to be paid by the customer |
| Pre-Conditions | An order is received at the restaurant |
| Trigger | When the manager goes through the payment option |
| Description | This use case describes the event when the manager wanted to see the details of the payment option to track the details. |
| Exceptions | If there is no order received |

**Table 30**

Table presents the Automated Reporting use case description to show the interaction between a Manager and a application when enable to get report of the total amount of sales.

|  |  |
| --- | --- |
| Use Case | Automated Reporting |
| Primary Actor | Manager |
| Goals In Context | Enable managers to get the Report of the total amount of sales. |
| Pre-Conditions | When the number of orders is received at the restaurant |
| Trigger | When the number of food is successfully delivered to the destination |
| Description | This use case describes the event when the manager wanted to generate the total payment report in which he will get all the details that how much order he received, for many rupees of order, gross profit, gross sales, and Deduction of amount. There will be 3 types of the generated report given in the system for the restaurant manager i.e.   1. Annually 2. Monthly 3. Weekly |
| Exceptions | If there is no order receive |

**Table 31**

Table presents the Payment History use case description to show the interaction between a Manager and a application when enable to see details of payment which is paid by the customer.

|  |  |
| --- | --- |
| Use Case | Payment History |
| Primary Actor | Manager |
| Goals In Context | Enable managers to see the details of the payment which is paid by the customer |
| Pre-Conditions | An order is received at the restaurant |
| Trigger | When the manager taps into the history of payment option |
| Description | This use case describes the event when the manager wanted to see the details of the payment which has been paid by the customer. |
| Exceptions | If there is no order received |

**Table 32**

**5.3.3. Delivery Module**

Table presents the Log In use case description to show the interaction between a Delivery Man and a application when enable to accept orders from the system.

|  |  |
| --- | --- |
| Use Case | Log In |
| Primary Actor | Delivery Man |
| Goals In Context | Enable the delivery man to accept orders from the system for delivery |
| Pre-Conditions | The delivery man has a valid name and password to login the system |
| Trigger | It came when the delivery man wanted to open his/her account to receive deliveries from the system |
| Description | This use case describes the event when the delivery man wanted to accept the deliveries from the customers via our system so he/she will log in to our system. |
| Exceptions | A user id and Password is invalid |

**Table 33**

Table presents the Contact Customer use case description to show the interaction between a Delivery Man and a application when enable to contact the customer.

|  |  |
| --- | --- |
| Use Case | Contact Customer |
| Primary Actor | Delivery Man |
| Goals In Context | The delivery man contacts the person who orders the food so that he can reach his/her destination successfully |
| Pre-Conditions | The delivery man accepts the ride for delivery |
| Trigger | It came when the delivery man picks up the food then will contact the person |
| Description | This use case describes the event when the delivery guy accepts deliveries from customers via our system so and wanted to reach the person’s destination as soon as possible and wanted to know the perfect location of their house. |
| Reference | Ref to use case  Log In |
| Exceptions | The user has not accepted any ride for delivery |

**Table 34**

Table presents the Duty Service use case description to show the interaction between a Delivery Man and a application when enable to select whether to deliver the food or not.

|  |  |
| --- | --- |
| Use Case | Duty Service |
| Primary Actor | Delivery Man |
| Goals In Context | The delivery man has full control over his job whether wanted to deliver the food or not with just one click |
| Pre-Conditions | The delivery man logs in to the system |
| Trigger | It came when the delivery man open the app and log in to the system |
| Description | This use case describes the event when the delivery guy logs in to our system he has an option for ON/OFF whether he wanted to deliver the food or not. If he taps on the ON button, then all nearby deliveries will pop up. There has been a restriction made by the company that a delivery man has to be on job for some particular hours per week. |
| Reference | Ref to use case  Log In |
| Exceptions | The user is not logged in to the system |

**Table 35**

Table presents the Order Details use case description to show the interaction between a Delivery Man and a application when enable to see the details of order.

|  |  |
| --- | --- |
| Use Case | Order Details |
| Primary Actor | Delivery Man |
| Goals In Context | The Delivery man has a tab where he sees all the details of the order |
| Pre-Conditions | The delivery man is on the service |
| Trigger | It came when the delivery man accepts the ride for delivery |
| Description | This use case describes the event when the delivery man accepts a delivery he has a tab where he will be shown what the person has ordered is there any tip for the delivery man to check before accepting the from the Restaurant? For example, if the customer gives tip to check the cold drink if it is chilled then accepts it otherwise not. |
| Reference | Ref to use case  Log In |
| Exceptions | The user has not accepted any ride for delivery |

**Table 36**

Table presents the Payment/Service History use case description to show the interaction between a Delivery Man and a application when enable to see the details of the payment.

|  |  |
| --- | --- |
| Use Case | Payment/Service History |
| Primary Actor | Delivery Man |
| Goals In Context | Enable the delivery man to see the details of the Payment and the working hours service |
| Pre-Conditions | An order is accepted by the delivery man |
| Trigger | When the delivery man taps into the history of payment and service option |
| Description | This use case describes the event when the manager wanted to see the details of the payment which he has received from the customer so that he can give it to our system manager and he can also see the service hour which he has provided to our system on that bases he also gets a reward of appreciation from our system managers. |
| Exceptions | No service Hour duty provided |

**Table 37**

**6. Non- Functional Requirements**

**7. Requirements Prioritization**

Prioritization Criteria

For prioritizing we will use the 1 to 9 scale with a restricted range of 1, 3, 6, and 9. For each of the criteria, the following are guides to stakeholders for selecting the correct rating value.

Relative Benefit

1 = No users would find this feature useful

3 = some users would find this feature useful

6 = some users would find this feature very useful

9 = Many users would this feature very useful

Relative Penalty

1 = No users would be upset if this feature was absent

3 = some users would be upset if this feature was absent

6 = some users would be very upset if this feature was absent

9 = Many users would be very upset if this feature was absent

Relative Cost

1 = It would be quick, easy, and low cost to implement this feature

3 = It would be moderately easy and middling cost to implement this feature

6 = It would moderately difficult and expensive to implement this feature

9 = It would be very difficult and very expensive to implement this feature

Relative Risk

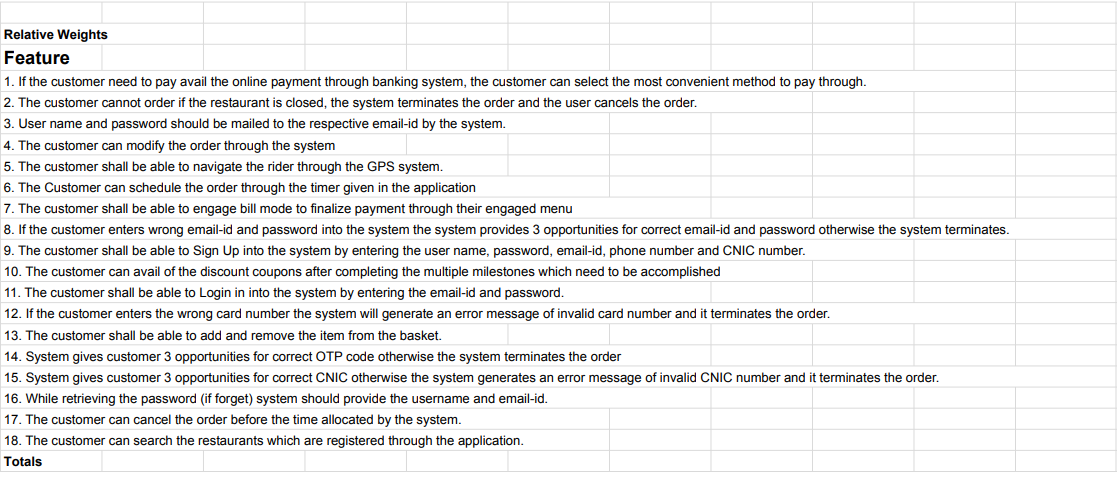
1 = This feature could be implemented as requested with almost no risk of change or delay

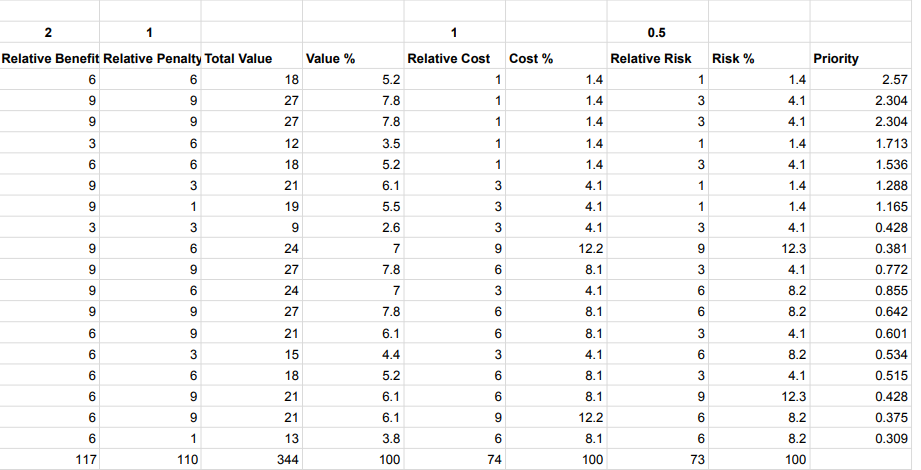
3 = There are minor concerns that this feature could be feasibly implemented as requested within the project timeline

6 = There are significant concerns that this feature could be feasibly implemented as requested within the project timeline

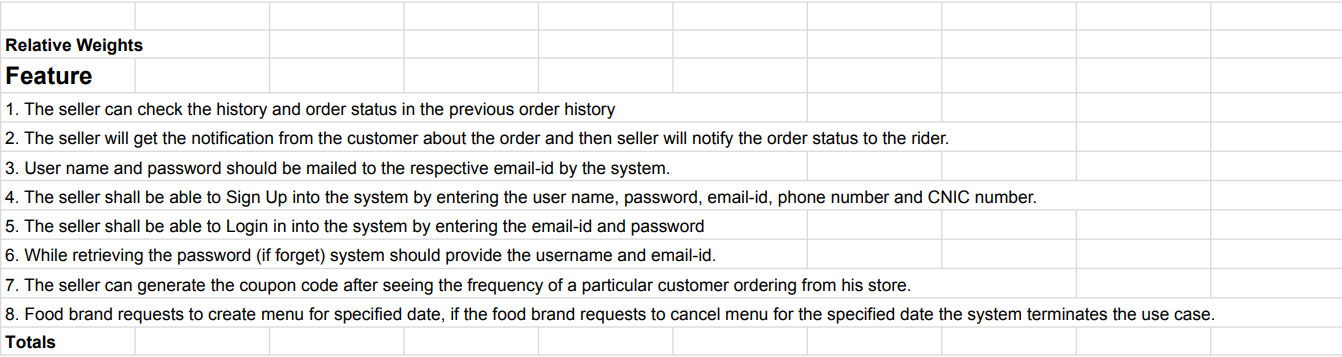
9 = There are very serious concerns that this feature could be feasibly implemented as requested within the project timeline

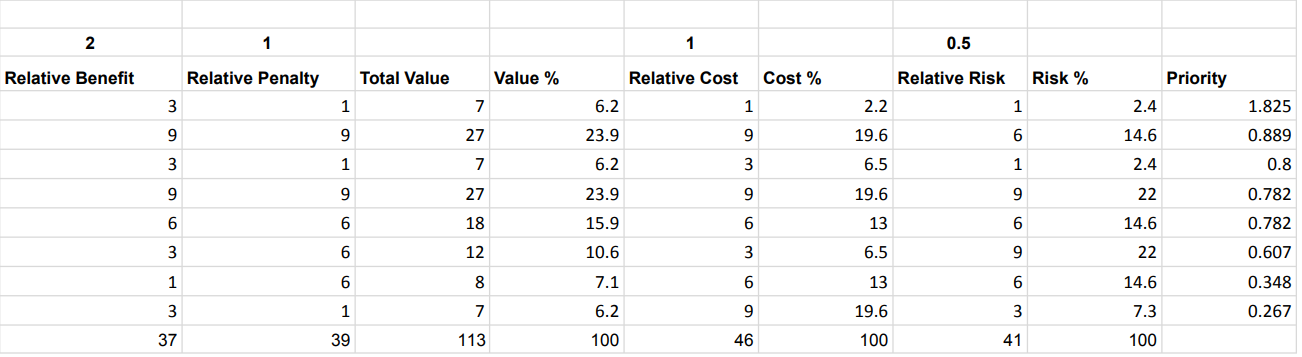
**7.1. Prioritized Customer Functional Requirements**

****

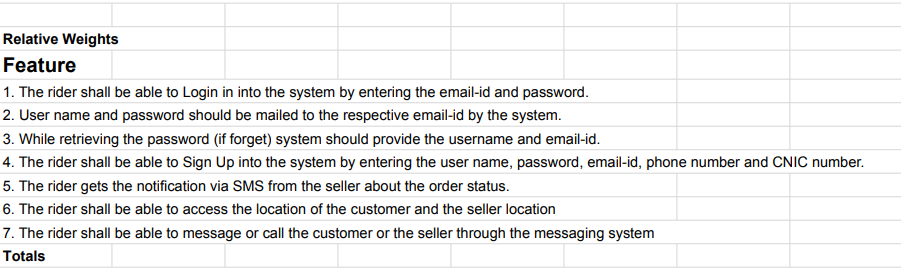
****

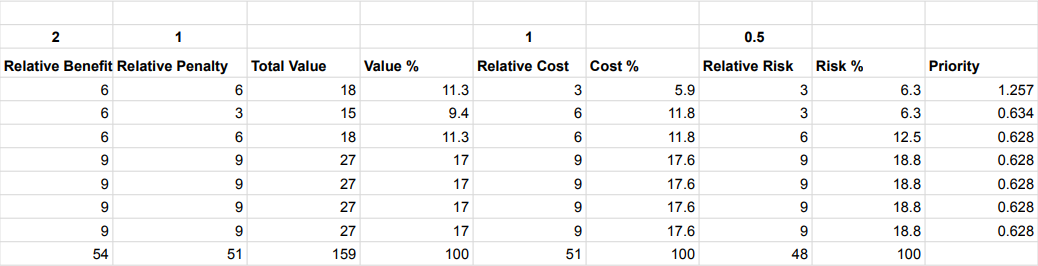
**7.2. Prioritized Seller Functional Requirements**

****

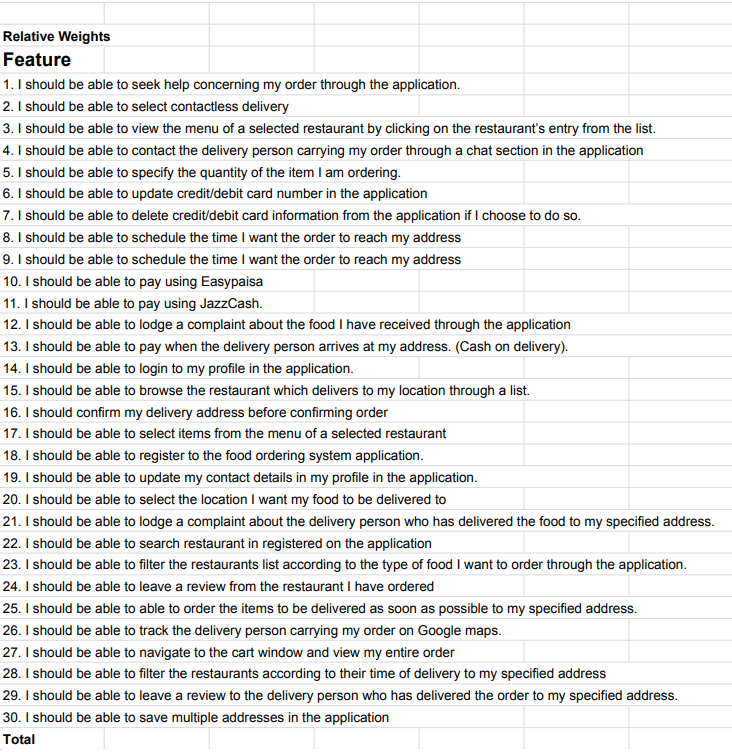
****

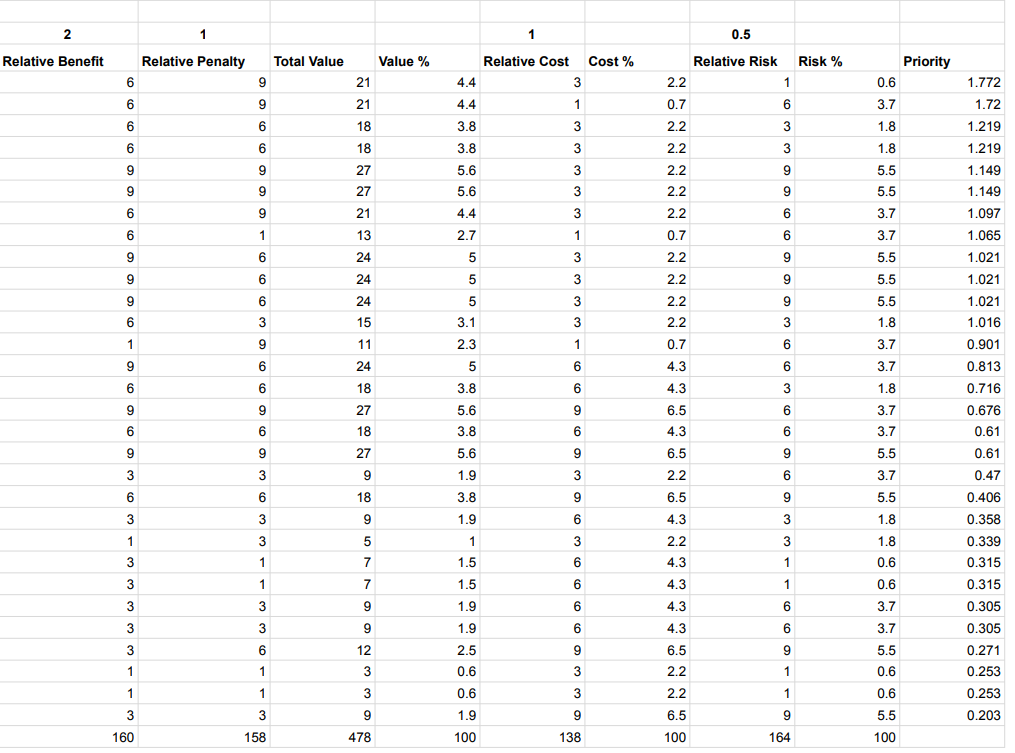
**7.3. Prioritized Delivery Functional Requirements**

****

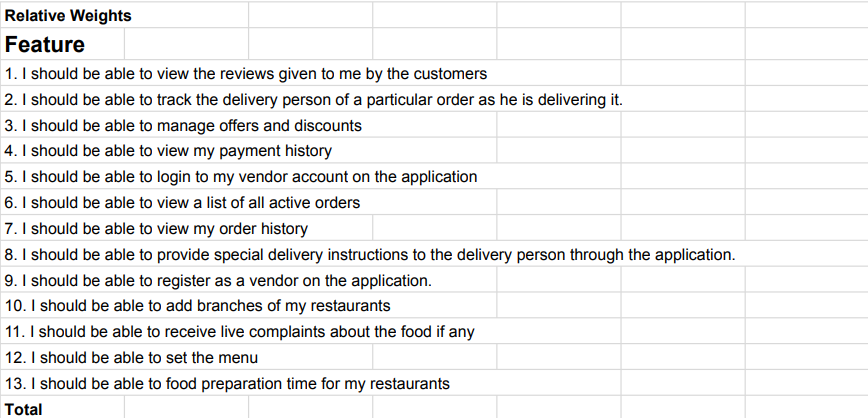
****

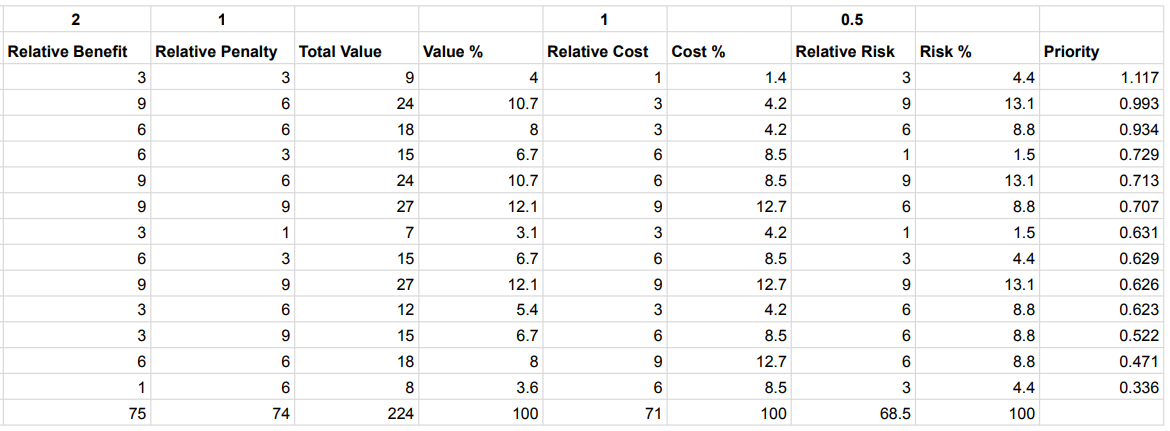
**7.4. Prioritized Customer User Requirements**

****

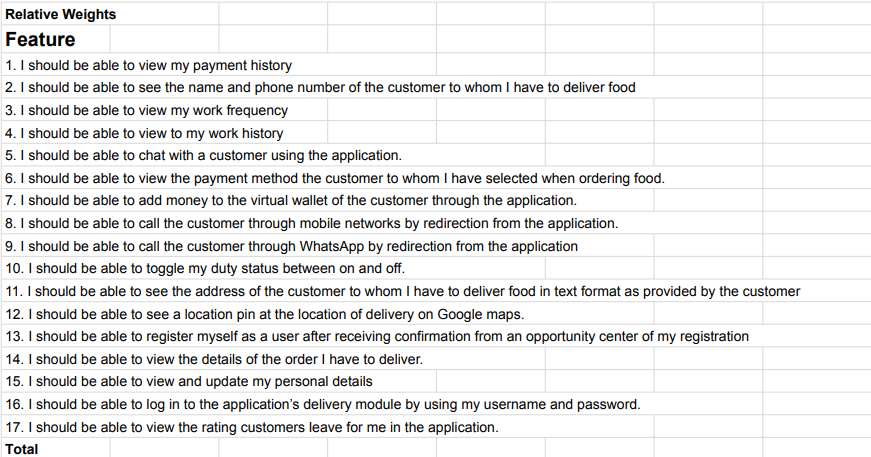
****

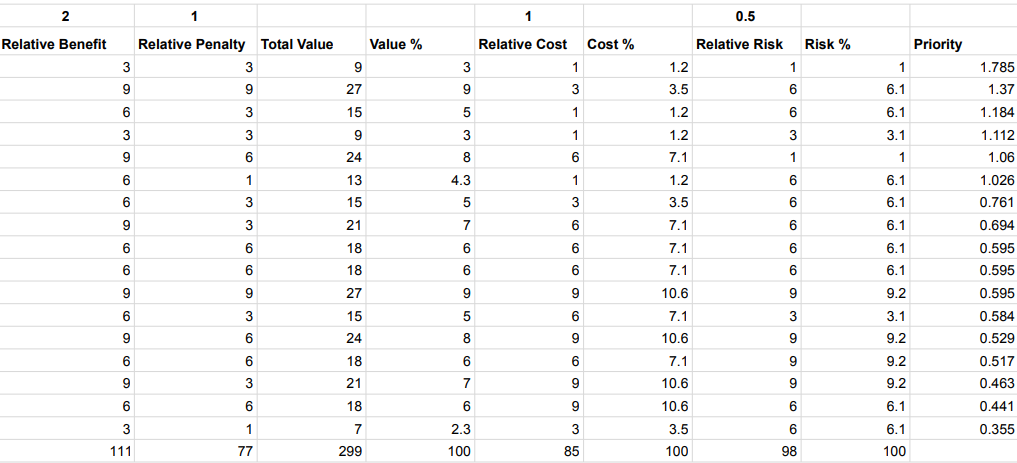
**7.5. Prioritized Seller User Requirements**

****

****

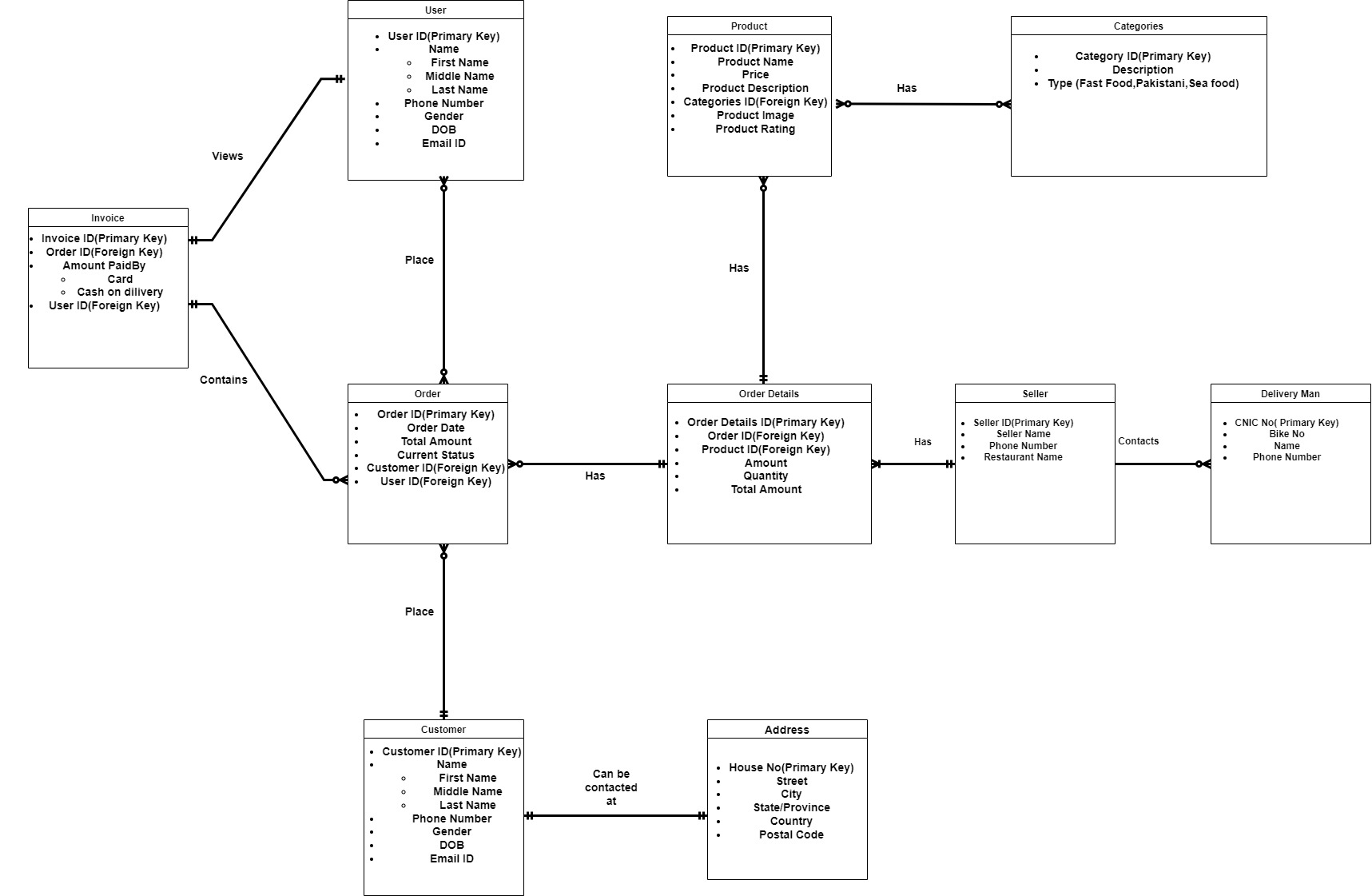
**7.6. Prioritized Delivery User Requirements**

****

****

**8. Data Requirements**

**8.1. ERD Diagram**

****

**Figure 7**

**8.2. Data Dictionary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DATA ELEMENT** | **DESCRIPTION** | **COMPOSITION** | **DATA TYPE** | **LENGTH** |
| INVOICE | The list of food delivered with a statement of the sum due for these | Invoice ID | Integer | 10 |
| Order ID  Amount paid by | Integer | 10 |
| +Cash on delivery  +Card |
| User ID | Integer | 10 |
|  | | | | |
| USER | Unique identifier for a user | USER ID | Integer | 10 |
| +Name  First Name  Middle Name  Last Name | String |  |
| +Phone Number | Double | 12 |
| +Gender | String |  |
| +DOB | String |  |
| +Email ID | Varchar |  |
|  | | | | |
| ORDER | The details of order that is being placed using food delivery application | ORDER ID | Integer | 10 |
| +Order Date | String |  |
| +Total Amount | Integer | 10 |
| +Current Status | Timestamp |  |
| CUSTOMER ID | Integer | 10 |
| USER ID | Integer | 10 |
|  | | | | |
| CUSTOMER | The details of customer who is registered to our food deliver application | CUSTOMER ID | Integer | 10 |
| +Name  First Name  Middle Name  Last Name | String |  |
| +Phone Number | Double | 12 |
| +Gender | String |  |
| +DOB | String |  |
| +Email ID | Varchar | 30 |
|  | | | | |
| PRODUCT | The details of product that is available on food delivery application | PRODUCT ID | Integer | 10 |
| +Product Name | String |  |
| +Price | Integer | 6 |
| +Product Description | String |  |
| +Product Image | JPEG |  |
| +Product Rating | Numeric |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| ORDER DETAILS | The details of the order that is being placed by the customer | ORDER DETAILS ID | Varchar | 30 |
| ORDER ID | Integer | 10 |
| PRODUCT ID | Integer | 10 |
| +Amount | Integer | 6 |
| +Quantity | Numeric | 3 |
| +Total Amount | Integer | 6 |
|  | | | | |
| ADDRESS | The location details of the customer whose order has to be placed | +House Number | String |  |
| +Street | Varchar | 30 |
| +City | String |  |
| +State/Province | String |  |
| +Country | String |  |
| +Postal Code | Integer | 6 |
|  | | | | |
| SELLER | The details of seller who is a third party partner | SELLER ID | Integer | 10 |
| +Seller Name | String |  |
| +Phone Number | Double | 12 |
| +Restaurant Name | String |  |
|  | | | | |
| CATEGORIES | The category description of food that is available on the food delivery application | CATEGORY ID | Integer | 10 |
| +Description | String |  |
| TYPE |  |  |
| +Fast Food | String |  |
| +Sea Food |  |
| +Desi Food |  |
|  | | | | |
| DELIVERY MAN | The details of the rider | DELIVERY MAN ID | Double | 14 |
| +Bike Number | Integer | 7 |
| +Name | String |  |
| +Phone Number | Double | 12 |

**Table 38**

**8.3. Report Specification**

|  |  |
| --- | --- |
| Report Element | Element Description |
| Title | Payment Reporting |
| Purpose | The purpose of this report is for the seller can look into how many orders he received and date how many amounts are in his account and what is the review about his restaurant’s food |
| Decision | The decision made from this report on how the seller is doing and whether his restaurant is getting any orders or not. |
| User | It can only be accessible by the seller system owner |
| Data Source | The data will extract from the managing order, payment, and lastly customer review |
| Frequency | The report data will update automatically by the system every week and can be accessible by the seller at any time and at any place. The report is only generated if the seller has delivered any food item to the customer’s place. It will |
| Layout | It’s in the pdf form |
| Look | Restaurant Daily Sales Report Template Check more at  https://www.pruneyardinn.com/restaurant-... | Sales report template,  Restaurant management, Report template |
| Security | It can only be generated by the seller itself |

**Table 39**

|  |  |
| --- | --- |
| Report Element | Element Description |
| Title | Order Reporting |
| Purpose | The purpose of this report is for the customer to know how many orders to date he/she had ordered from our system of how many rupees. |
| Decision | The decision made from this report is how frequently the customer ordering from our system/ |
| User | It can only be accessible by the customer and System owner |
| Data Source | The data will extract from the Order Placed, Received order, and lastly customer review |
| Frequency | The report data will update automatically by the system every day and can be accessible by the seller at any time and at any place. The report is only generated if the customer itself. |
| Layout | It’s in the print lay out form. |
| Look | It has following attributes   1. Total orders 2. Total Payment 3. Date and Time 4. By which Restaurants |
| Security | It can only be generated by the user itself |

**Table 40**

|  |  |
| --- | --- |
| Report Element | Element Description |
| Title | Delivered Reporting |
| Purpose | The purpose of this report is for the delivery guy so he can look into that how many orders he has accepted to deliver and successfully delivered with reviews by the customer. |
| Decision | The decision made from this report that to know how the delivery guy is dedicated to his work and how many orders he delivered. |
| User | It can only be accessible by the Delivery Guy and System owner |
| Data Source | The data will extract from the order accepted, payment, and lastly customer review |
| Frequency | The report data will update automatically by the system every 3 to 4 days and the delivery guy can look into that what is the review of the customer about his gesture and performance and how much time he gives to this service company. |
| Layout | It’s in the pdf form |
| Look | Restaurant Daily Sales Report Template Check more at  https://www.pruneyardinn.com/restaurant-... | Sales report template,  Restaurant management, Report template |
| Security | It can only be generated by the seller itself |

**Table 41**

**9. Business Rules**

**9.1. Facts**

Table presents the identified Business requirements facts that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| BF1 | All orders received online are pushed in the system directly |
| BF2 | The users can filter the restaurants according to their budge, or nearby location. |
| BF3 | Delivery time is 45 mins for every order. |
| BF4 | Any changes made in the menu of any restaurant will automatically reflect on the mobile application |
| BF5 | Users can view their order history. |
| BF6 | Users can give feedback according to their experience |
| BF7 | The delivery riders can be tracked through GPS location. |
| BF8 | The food delivery application can be operated through android/ios or any other system |
| BF9 | The users can check their digital wallet. |
| BF10 | The application has in-app chat feature for the user among other features. |

**Table 42**

**9.2. Constraints**

Table presents the identified Business requirements constraints that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| BC1 | The customer must supply a local telephone number for online order. |
| BC2 | A valid credit/debit card number must be available for order confirmation |
| BC3 | The online order must be confirmed by head server via email. |
| BC4 | Deliveries must be completed between 8:00 am – 12:00am |
| BC5 | Network transmissions that involve financial information or personally identifiable information require 128 bit encryption. |
| BC6 | The loading spaces of the vehicle and all boxes have to be clean and hygienic. |
| BC7 | The primary candidate tool chains are Java, Swing, C++ and MySql. |
| BC8 | Only a registered user can make an online order. |
| BC9 | Non registered users can view only the restaurant and the menu. |
| BC10 | Every order from a customer must be satisfied by exact one delivery vehicle. |
| BC11 | Within 4-6 km our rider will be delivering the food |
| BC12 | The memory usage of the app will have to be constrained by the devices it is intended to run. Since most Tablets and mobile may have limited apps |
| BC13 | 1. The system must provide a capacity for parallel operation and system design should not introduce scalability issues about the number of surface computers, tablets, or displays connected at any one time |
| BC14 | If the system is down, then customers must not notice, or notice that the system recovers quickly. |

**Table 43**

**9.3. Action Enabler**

Table presents the identified Business requirements action enabler that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| BA1 | If some item is not available then notify the user. |
| BA2 | If user ordered a meal from a restaurant then recommend “beverages” and “desserts” items to user before completing the order. |
| BA3 | If the restaurant is closed then the order will not be confirmed and the user will be notified via email |
| BA4 | If user submits unsatisfactory feedback about delivery service then the server will call the user for clarity |
| BA5 | If the user orders from one restaurant then the order will be directed to the user history. |
| BA6 | If the user uses credit/debit or any other card method for online payment, then he/she will be notified about the amount deduction by our delivery app. |

**Table 44**

**9.4. Inference**

Table presents the identified Business requirements inference that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| BI1 | If user does not get the confirmation email or sms after ordering then the order is considered incomplete |
| BI2 | If the restaurant is closed then the restaurant is not supposed to be available on the delivery application at that time. |
| BI3 | If a user orders for more than 7 times in a month then that user will be considered as our golden customer |
| BI4 | If the user haven’t received their order in specified time then that order will be considered late |
| BI5 | If there are discounts on using certain cards for online payment then the codes will be available on discounts section so the user can apply them. |

**Table 45**

**9.5. Computations**

Table presents the identified Business requirements computations that directly relate to the entire subject Food Ordering System.

|  |  |
| --- | --- |
| **Requirements** | **Description** |
| BCO1 | Sales tax of 8% shall be added to each order |
| BCO2 | A state income tax of 5% is to be withheld from the total pay |
| BCO3 | The food delivery application will pay income tax of 20% of their profits to the government |
| BCO4 | The restaurants will pay 25% of their profits they make by our food delivery application |
| BCO5 | A generic formula for calculating ETAs (estimated time of arrival) before an order is placed would look like this  **Pre order ETA** = Max [{(Order Preparation Time) \* (Peak Hour Multiplier)} + (Delivery on Road Transit Time)] |

**Table 46**

**10. Business Requirements**

Following are the Business Requirements of our system:

1) Opportunity: - As Food Market is always been a major growing and expanding market. We have also seen that any crisis comes it is still the most sustainable business. Identifying the people’s problems while facing dine-in, as well as pickup or delivery, are as follows

* + - * 1. Limited options
        2. Waiting in Queues
        3. Time Consuming
        4. No track of delivery
        5. Communication Problems
        6. Payment
        7. Finding The Details of the Restaurant
        8. Customer Service

2) Business Objective& Success Criteria: - Following are the Business Objectives of our system

1. To reduce the time of delivering
2. To offer at least 120+ options of Restaurants nearby
3. Used by 50k registered members in the first 3 months
4. Expand our system to 40+ cities after 6 months
5. To generate of revenue

**Appendix A**

**GLOSSARY**

**A**

**Acceptancecriteria**:Theexitcriteriathatacomponentorsystemmustsatisfyinordertobe acceptedby a user, customer, or other authorized entity.

**Accuracy**: The capability of the software product to provide the right or agreed results or effects with the needed degree of precision.

**Activity diagram**: A graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

**Adaptability:**Thecapabilityofthesoftwareproducttobeadaptedfordifferentspecifiedenvironments.

**Availability:** The degree to which a component or system is operational and accessible when required for use. Often expressed as a percentage.

**B**

**BA:** see *Business Analysis, Business Analyst.*

**Baseline:** A specification or software product that has been formally reviewed or agreed upon, that there after serves as the basis for further development, and that can be changed only through a formal change control process.

**Benefit**: Value delivered to stakeholders

**Bestpractice:**Asuperiormethodorinnovativepracticethatcontributestotheimprovedperformance of an organization under given context, usually recognized as ‘best’ by other peer organizations.

**bug**: See *defect.*

**Business Analysis:** The set of tasks, knowledge, tools and techniques required to identify business needs and determine solutions to business problems [BABOK]. See also: *System Analysis*

**Business Analyst:** A person responsible for identifying the business needs of their clients and stakeholders, to determine solutions to business problems[BABOK]. See also: *System Analyst*

**Business Requirement:** Business Requirement captures the reasoning for initiating a project or task. It describes a justification for the project in terms of the value added to the business as a result of the project outcomes in comparison to the cost of developing the new solution.

**Business Goal:** Short-or long-term objective of an organization.

**Business Need:** Defines the business problem or opportunity, which BAs have to understand in order to recommend appropriate solutions.

**BusinessProcess:**Acollectionofactivitiesdesignedtoproduceaspecificoutputforaparticularcustomerormarket.

**BusinessProcessModelingNotation(BPMN):**Agraphicalnotationthatdepictsthestepsinabusiness process. BPMN depicts the end to end flow of a business process. The notation has been specifically designed to coordinate the sequence of processes and the messages that flow between different process participants in a related set of activities [BPMN.ORG].

**BusinessSponsor:**Apersonwhoproposestheproposednewprojecttothegovernancegroupforthemtoselectandprioritizethe portfolio of projects for the enterprise [BABOK].

# C

**Class:**Aclassdescribesasetofobjectsthatsharethesamespecificationsoffeatures,constraints,andsemantics. Classisakindofclassifierwhosefeaturesareattributesandoperations.

**Class diagram**: A type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes

**Client**: see *Customer*

**Commitment**: The degree of obligation of meeting the requirement.

**Completenessofarequirement**:Thedegreetowhicharequirementcontainsallnecessaryinformation.

**Complexity:** The degree to which a component or system has a design and/or internal structure that is difficult to understand, maintain and verify.

**Compliance:**Thecapabilityofthesoftwareproducttoadheretostandards,conventionsorregulationsin laws and similar prescriptions

**Configuration:** The composition of a component or system as defined by the number, nature, and interconnections of its constituent parts.

**Consistency:** The degree of uniformity, standardization, and freedom from contradiction among the documents or parts of a component or system[IEEE610].

**Constraint**: A statement of restriction that modifies a requirement or set of requirements by limiting the range of acceptable solutions.

**Context**: System view from any useful perspective.

**Contractor**: see *Vendor*

**Context diagram**: A diagram that represents the actors outside a system that could interact with that system.

**Customer:**Currentorpotentialbuyeroruseroftheproductsorserviceofanindividualororganization,called the supplier, seller, or vendor.

# D

**Data definition:** An executable statement where a variable is assigned a value.

**Data flow:** An abstract representation of the sequence and possible changes of the state of data objects, where the state of an object is any of: creation, usage, or destruction.

**Data flow diagram:** A graphical representation of the sequence and possible changes of the state of data objects, where the state of an object is any of: creation, usage, ordestruction.

**Deliverable:** Any (work) product that must be delivered to someone other than the (work) product’s author.

**Dependency**: A reliance of some kind, of one set of components on another set of components, or one set of requirements or other artifacts on another set.

**Domain:** The set from which valid input and/or output values can be selected.

**E**

**Efficiency:** The capability of the software product to provide appropriate performance, relative to the amount of resources used under stated conditions [ISO/IEC25000].

**Elicitation:** The act of obtaining information from other people. In the context of Requirements Engineering, elicitation is the process of gathering requirements from stakeholders.

**End user**: see *User*

**Entity:** (1) An element or set of elements that has a distinct, separate existence, although it need not be a material existence.

**ERD**: see *Entity-relationship diagram*.

**Entity-relationship model**: An abstract and conceptual representation of data. Entity-relationship model consists of a set of entities, characterized by attributes and linked by relationships.

**Error*:***A human action that produces an incorrect result[AfterIEEE610].

**Estimate**: A numeric judgment about a future, present or past level of a scalar system attribute. This includes all performance and cost attributes. Estimates are usually made where direct measurement is: impossible (future),or impractical(past),or uneconomic(currentlevels).

**Evaluation:** See *testing*.

**Exceptionhandling:**Behaviorofacomponentorsysteminresponsetoerroneousinput,fromeitherahumanuserorfromanothercomponentor system, or to an internal failure.

**F**

**Facilitator**: A person or group who assists others in carrying out a work process, such as quality control or setting objectives; by virtue of their being especially trained, qualified, and knowledgeable in that process .

**Failure:** Deviation of the component or system from its expected delivery, service or result[Fenton].

**Fault:** See *defect*.

**Feature:** An attribute of a component or system specified or implied by requirements documentation (for example reliability, usability or design constraints)[IEEE1008].

**Function:** A description of “what “a system does. A function has a corresponding impliedpurposeandisafundamentalpartofasystemdescription:asystemconsistsoffunctionattributes,performance attributes, resource (cost) attributes and design attributes. All attributes exist with respect to defined specified conditions. A function can often be decomposed into a hierarchical set of sub-functions .

**Functional requirement:** A requirement that specifies a function that a component or system must perform [IEEE 610].

**Functionality:** The capability of the software product to provide functions which meet stated and implied needs when the software is used under specified conditions [ISO/IEC25000].

# G

**Goal:** A desired state or result of an undertaken. Goals should be measurable and defined in time so that the progress can be monitored.

# I

**Impact:** Estimated or actual numeric effect of a design idea on a requirement attribute under given conditions.

**Installability:**Thecapabilityofthesoftwareproducttobeinstalledinaspecifiedenvironment.Seealso*Portability*.

**Integration:**Theprocessofcombiningcomponentsorsystemsintolargerassemblies.

**Interview**:Aconversationaltechniquewheretheinterviewerisaskingtherespondertoobtaininformationonspecifiedtopic.

# L

**Learnability:**Thecapabilityofthesoftwareproducttoenabletheusertolearnitsapplication.Seealso*Usability*.

**Life cycle model:** A partitioning of the life of a product or project into phases.

# M

**Maintainability:**Theeasewithwhichasoftwareproductcanbemodifiedtocorrectdefects,modified to meet new requirements, modified to make future maintenance easier, or adapted to a changed environment.

**Maintenance:**Modificationofasoftwareproductafterdeliverytocorrectdefects,toimproveperformanceorother attributes, or to adapt the product to a modified environment.

**Measure:** The number or category assigned to an attribute of an entity by making a measurement.

**Measurement:** The process of assigning a number or category to an entity to describe an attribute of that entity.

**Measurementscale:**Ascalethatconstrainsthetypeofdataanalysisthatcanbeperformedonit.

**Metric:** A measurement scale and the method used for measurement [ISO14598]

**Module:** See *component*.

# 

# N

**Need:** Something desired by a defined stakeholder. Satisfying that need would have some value forsomestakeholder.Aneedmightnotbeagreedasaformalrequirement,anditmightnotbeprioritized such that it is actually acted upon (designed and implemented).Need is a term often used as a stake holder view of a problem before requirements specification is carried out.

**New Product Development**: A complete process of bringing a new product or service to market.

**Non-functional requirement:** A requirement that does not relate to functionality, but to attributes such as reliability, efficiency, usability, maintainability and portability.

# 

# O

**Output:**Avariable(whetherstoredwithinacomponentoroutside)thatiswrittenbyacomponent.

# P

**Performance:** The degree to which a system or component accomplishes its designated functionswithingivenconstraintsregardingprocessingtimeandthroughputrate[IEEE610].Seealso*efficiency*.

**Persona:** A fictional character, an arche type description, which representsthedifferenttypesofuserswhowillbeusingthefinalproductorsolution.Personashouldrepresentagroupofpeoplewiththesameneeds,attitude,behaviororexpectations towards the product.

**Point of view**: A certain perspective on the system or requirements.

**Output:**Avariable(whetherstoredwithinacomponentoroutside)thatiswrittenbyacomponent.

# Q

**Quality attribute:** A feature or characteristic that affects an item’s quality.

**Quality characteristic**: See *Quality attribute*.

**Quality Management:** Coordinated activities to direct and control an organization with regard to quality. Direction and control with regard to quality generally includes the establishment of the quality policy and quality objectives, quality planning, quality control, quality assurance and quality improvement.

# R

**Recoverability:**Thecapabilityofthesoftwareproducttore-establishaspecifiedlevelofperformanceandrecover the data directly affectedin caseoffailure.

**Redundancy**:Multiple occurrences of the same information in different places.

**Release**: A version of the solution released for installation and use by the customer/end users.

**Reliability:**Theabilityofthesoftwareproducttoperformitsrequiredfunctionsunderstatedconditionsforaspecified period of time, or for a specified number of operations[ISO/IEC25000].

**Replace ability:** The capability of the software product to be used in place of another specifiedsoftwareproductforthesamepurposeinthesameenvironment[ISO/IEC25000].Seealso*Portability*.

**Requirementsacceptance:**Aprocessofformalagreementthatthecontentandscopeoftherequirementsareaccurateandcomplete between all relevant stakeholders [BABOK].

**Requirements analysis**: A set of tasks, activities and tools to determine whether the stated (elicited)requirementsareunclear,incomplete,ambiguous,orcontradictory,andthendocumentingtherequirementsinaformofconsistentmodel.

**Requirement attribute:** Descriptive information about a requirement that enrichesits definition beyond the statement of intended functionality. Examples include origin, rationale, priority, owner, release number, and version number.

**Requirements Development**: Collection of activities, tasks, techniques and tools to identify, analyze and validate requirements. Includes the process of transforming needs into requirements. In CMMI model, Requirements Development isanengineeringprocessareaatMaturityLevel3.

**Requirements elicitation**: see *Elicitation*

**Requirements Engineering:** A sub-discipline of systems engineering and software engineering that is concerned with determining the goals, functions, and constraints of hardware and software systems. Requirements Engineering discipline involves the following sub-processes: requirements elicitation, analysis and negotiation, specification, system modeling, requirements validation and requirements management.

**Requirements model**: A representation of user requirements using text and diagrams. Requirements models can also be called user requirements models or analysis models and can supplement textual requirements specifications.

**Requirementssource**:Thesourcefromwhichrequirementshavebeenderived.Requirementssourcescanbestakeholders,documents,businessprocesses,existingsystems,marketetc.

**Requirementsspecification(customer):**Aspecificationdescribingtheproblemarea.(Customerrequirements specification is usually provided by the customer and contains a description of the required capabilities of a solution from the customer's point of view.)

**Review:** An evaluation of a product or project status to ascertain discrepancies from planned results and to recommend improvements. Examples include management review, informal review, technical review, inspection, and walkthrough [IEEE1028].

**Risk:** (1) The effect of uncertainty on objectives, whether positive or negative[ISO 31000]. (2)A factor that could result in future negative consequences; usually expressed as impact and likelihood.)[ISTQB].

**Risk analysis:** The process of assessing identified risks to estimate their impact and probability of occurrence (likelihood).

**Risk category**: See *risk type*.

**Riskcontrol:**Theprocessthroughwhichdecisionsarereachedandprotectivemeasuresareimplementedforreducingrisks to,ormaintainingrisks within, specified levels.

# S

**Safety:** The capability of the software product to achieve acceptable levels of risk of harm to people, business, software, property or the environmentina specifiedcontextof use[ISO/IEC25000].

**Scalability:** The capability of the software product to be upgraded to accommodate increased loads[Gerrard].

**Scenario**: (1) A projected course of action, events or situations leading to specified result. (2) An ordered sequence of interactions between specified entities (e.g. a system and an actor). (3) In UML:anexecutiontraceofa usecase.

**Scope**: The extent of influence of something. Scope can apply to anything, like a specification, or a specified system orproject [TGilb].

**Security:** Attributes of software products that bear on its ability to prevent unauthorizedaccess,whetheraccidentalordeliberate,to programsanddata[ISO/IEC25000].See also *Functionality.*

**Sequence diagram:**In UMLitisastructured representation of behavior asa series of sequentialstepsovertime.Sequencediagramisakindofinteractiondiagramthatshowshowprocessesoperatewith oneanotherandin whatorder.

**Software quality:** The totality of functionality and features of a software product that bear on its ability to satisfy stated or implied needs[ISO/IEC25000].

**Software quality characteristic**: See *Quality attribute*.

**Solution:**(1)Solutionistheimplementationoftherequirement.(2)Adesignideawhich,ifimplemented, is expected to lead to the partial or full satisfaction of a set of attribute requirements;to solvea(defined)problem[TGilb].

**Solution model**:Amodeldescribingthe solutionareafrom differentviewsonthesystem.

**Solution validation:** Solution validation is the activity of explaining the solution's appropriateness tostakeholdersandsponsor.

**SubjectMatterExpert:**apersonwhoexhibitsthehighestlevelofexpertiseinperformingaspecializedjob, task,or skill [Pyzdek,Thomas andPaulA. Keller].

**SolutionSpecification**:alsocalledFunctionalSpecification,SystemRequirementSpecificationorSoftware Requirements Specification. Describes the solutionarea.

**Specification:** A document that specifies, ideally in a complete, precise and verifiable manner, the requirements, design, behavior, or other characteristics of a component or system, and, often, theproceduresfordeterminingwhetherthese provisionshave beensatisfied[IEEE610].

**Stability:** The capability of the software product to avoid unexpected effects from modifications inthesoftware [ISO/IEC25000]. Seealso*Maintainability*.

**Stakeholder:** Anyperson whohas an interestin an ITproject. Project stakeholdersareindividualsand organizations that are actively involved in the project, or whose interests may be affected as aresult of project execution or project completion. Stakeholders can exercise control over both theimmediatesystemoperationalcharacteristics,aswellasoverlong-termsystemlifecycleconsiderations(suchasportability,lifecyclecosts,environmentalconsiderations,anddecommissioningofthe system) [TGilb].

**Standard:**Formal,possiblymandatory,setofrequirementsdevelopedandusedtoprescribeconsistent approaches to the way of working or to provide guidelines (e.g., ISO/IEC standards, IEEEstandards,andorganizational standards) [CMMI].

**System:** A collection of components organized to accomplish a specific function or set of functions[IEEE610].

**System Analysis:**A set of activities, methods, techniques, tools focused on the translation of thebusiness requirements into systems requirements. It describes a system and its limitations to theenvironmentandprovidesawell-foundedunderstandingoftheenvironmentandthesystemrequirements.

**SystemAnalyst**:Atechnically-orientedperson,whoresearchesgivenbusinessproblem,planssoftwaresolutions,recommendssoftwareandsystems,andcoordinatesdevelopmenttomeetbusiness or other requirements. The task of System Analyst is to develop business requirements intotechnicalspecifications.

# T

**Testability:**Thecapabilityofthesoftwareproducttoenablemodifiedsoftwaretobetested[ISO/IEC25000].Seealso *Maintainability*.

**Traceability:**Theabilitytoidentifyrelateditemsindocumentationandsoftware,suchasrequirementswith associated tests. *See also* horizontal traceability, vertical traceability.

# 

# U

**Understandability:** The capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use [ISO/IEC25000].See also *Usability*.

**Usability:** The capability of the software to be understood, learned, used and attractive to the user when used under specified conditions[ISO/IEC 25000].

**Usecase:**Asequenceoftransactionsinadialoguebetweenanactorandacomponentorsystemwith a tangible result, where an actor can be a user or anything that can exchange information with the system.

**UseCasediagram**:InUMLadiagramthatshowsusecases,actors,andtheirinterrelationships.

**User**: A person who uses a software product.

# V

**Validation:**Confirmationbyexaminationandthroughprovisionofobjectiveevidencethattherequirementsfor aspecificintended useorapplicationhave beenfulfilled[ISO9000].

**Vendor**:Aperson,groupororganizationprovidingthesolution.

**Verification:** Confirmation by examination and through provision of objective evidence that specifiedrequirementshavebeenfulfilled [ISO9000].

**Version:**A specificformorvariationofsomething.

**Vision:** An image of the project's deliverable as the solution/

**Appendix B**

**1. JRP (Script)**

Ahsan Seller representative

Hussain Delivery representative

Jamaima Analyst/Scribe

Kashif IT specialist

Suleman JRP leader/ Analyst

Zeeshan Executive Sponsor/User representative

**Suleman**: Good morning everyone, and welcome to our joint requirements planning session for the food ordering system. Before we get started, let's go around the room and introduce ourselves.

Everyone introduces themselves.

**Suleman**: Great, thank you everyone for introducing yourselves. As you know, the purpose of this session is to discuss and define the requirements for the food ordering system. This will help us ensure that we are building a system that meets the needs of our users and the business. Now I would like Mr Zeeshan to say some Words.

**Zeeshan** : Thank you. First of all i would like to congratulate you all for being one of the first unicorn company of Pakistan, moreover it is my pleasure to announce that the budget for this app is 1m rupees so everyone please work accordingly.

**Suleman**: Now we will start the Requirement Gathering session .

User Requirements

Jamaima and Zeeshan

Jamaima, being the analyst, asks Zeeshan 3 questions about the app.

* What are the ideal requirements for the app?
* What are the Main requirements that can't be overlooked?
* What competitive app feature do you like the most that you would like in this application as well.

Seller Requirements

Suleman and Ahsan

Suleman, being the analyst, asks Ahsan 3 questions about the app.

* What are the ideal requirements for the app?
* What are the Main requirements that can't be overlooked?
* What competitive app feature do you like the most that you would like in this application as well.

Deliver Requirements

Kashif and Hussain

Kashif, being the analyst, asks Hussain 3 questions about the app.

* What are the ideal requirements for the app?
* What are the Main requirements that can't be overlooked?
* What competitive app feature do you like the most that you would like in this application as well.

**Suleman**: Now all the functional Requirements are done I would like Mr Kashif our IT specialist to tell us about the Non Functional requirements.

**Kashif**: Thank you.(he then proceeds to tell the representative about the Non Functional requirements.)

**Ahsan**: As a seller, security is our top priority.

**Kashif**: Indeed but increasing security may lead to performance degradation.

**Zeeshan**: But as a user Performance is key for a new app.

**Kashif**: hmmm let me think about it.

**Jamaima**: How about we keep an equal balance between both?

**Hussain**: Do you mind explaining further ?

**Jamaima**: Let's suppose after 4 wrong passwords the account will be automatically locked and if you are a seller you must visit on of our service centers to unlock the account and if you are a user you must provide some form of identification and then we will send a change password email.

**Kashif**: That's a great idea! Are you people ok with it ?

**Representatives** : yes that's a great solution.

**Suleman**: Great, this concludes today's meeting. We will all meet after 2 days so we can show you a prototype.

**2 DAYS LATER**

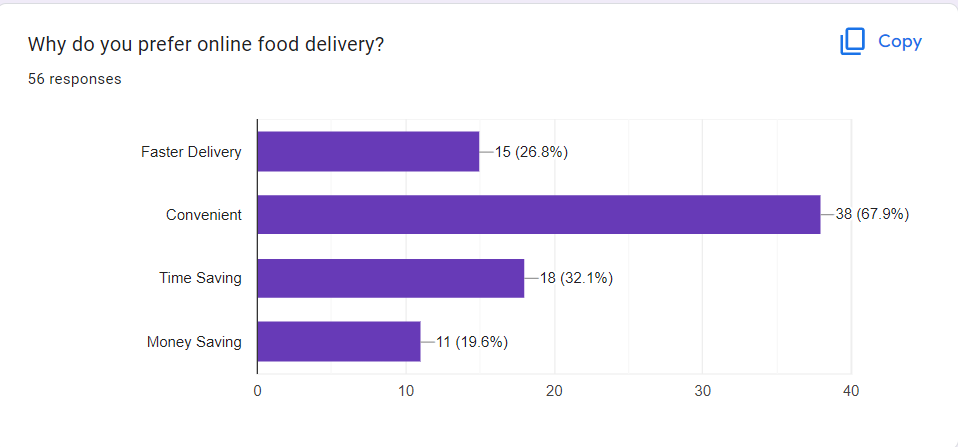
**Suleman**: Welcome back everyone. Our team has made a prototype to show how the app may look like, keep in mind it is only a prototype and is no way a functional app.

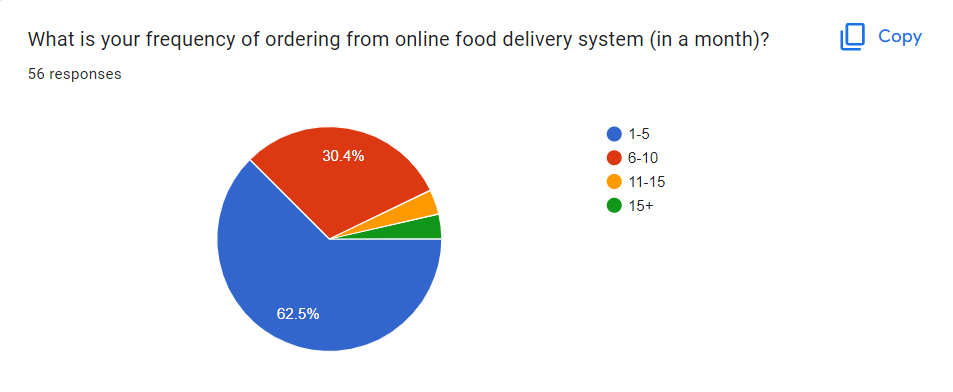
Everyone : okay.

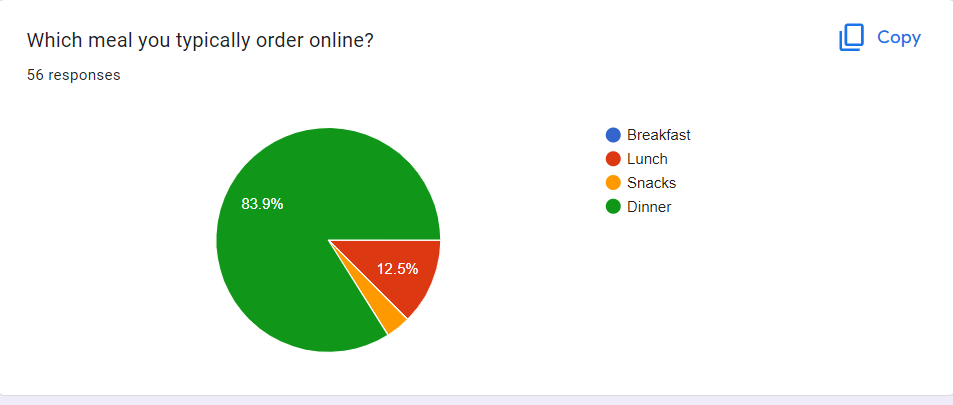
**Suleman** then proceeds to give a demo of the prototype

**Suleman**: I hope everything is clear and you all get an idea of what the app might look like. This session has been a huge success thanks to everyone for coming.

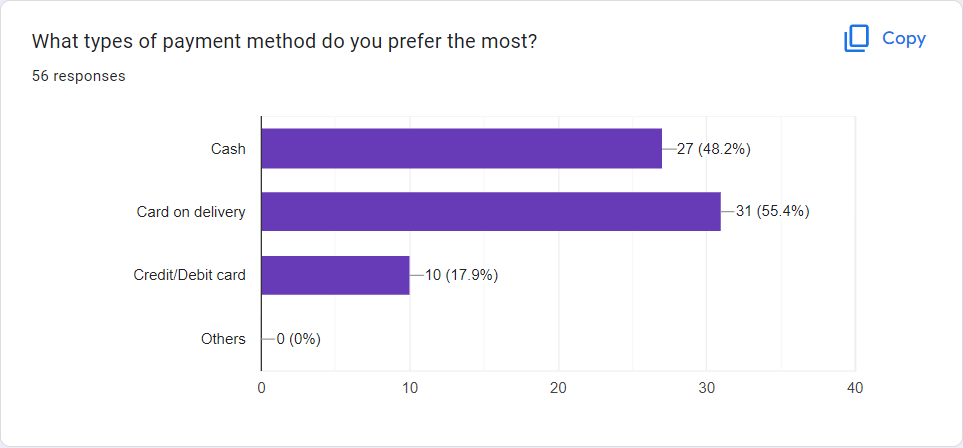
**2. Google Form**

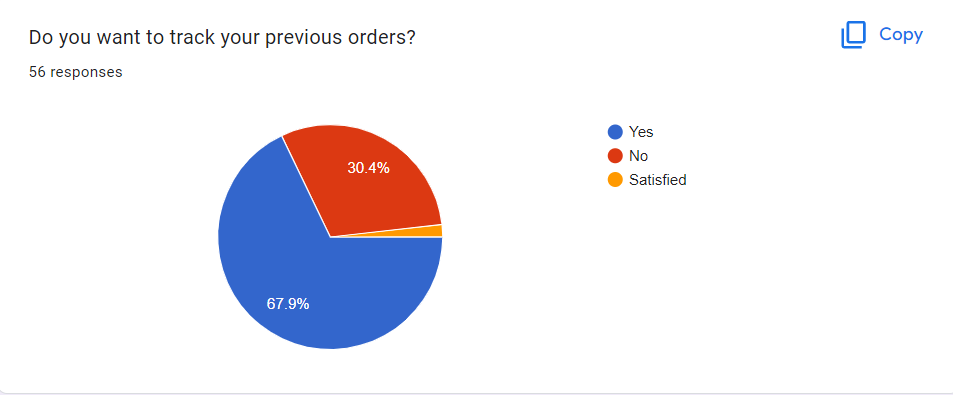
****

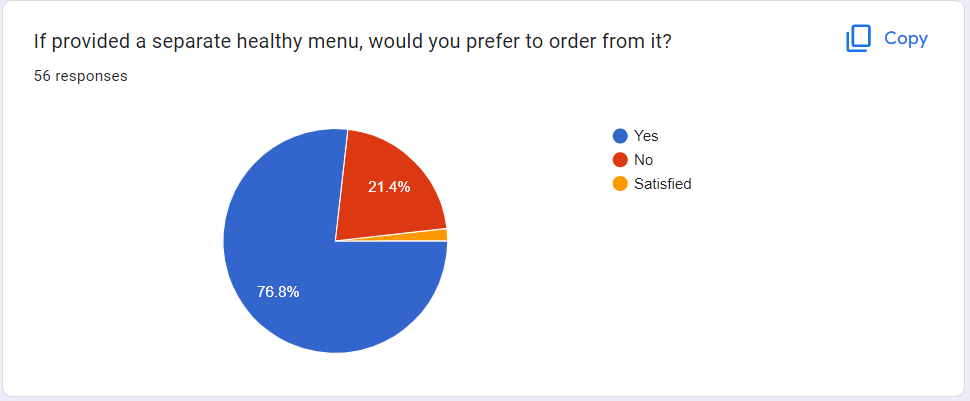
****

****

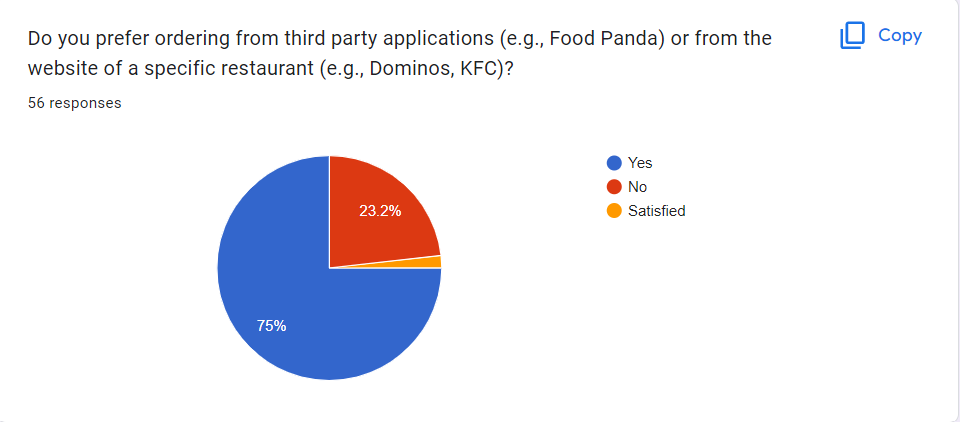
****

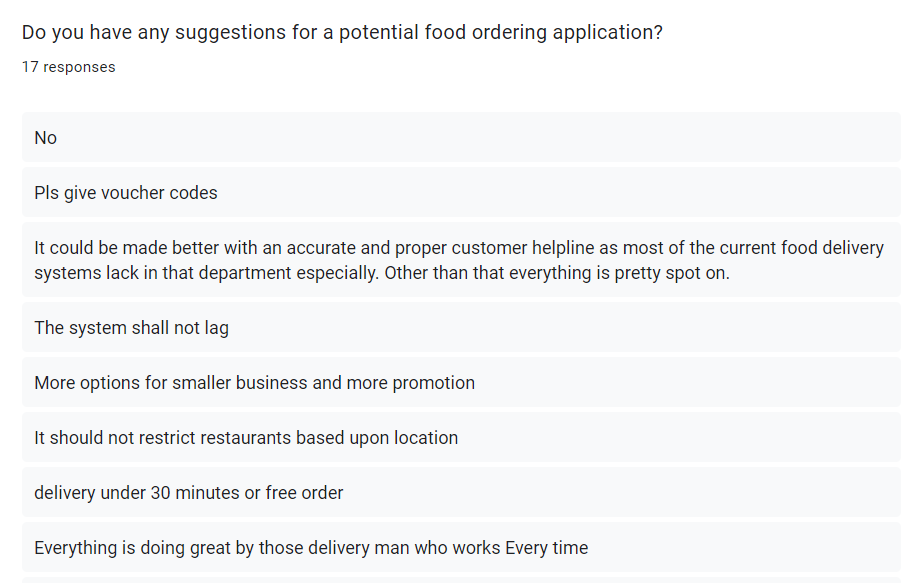
****

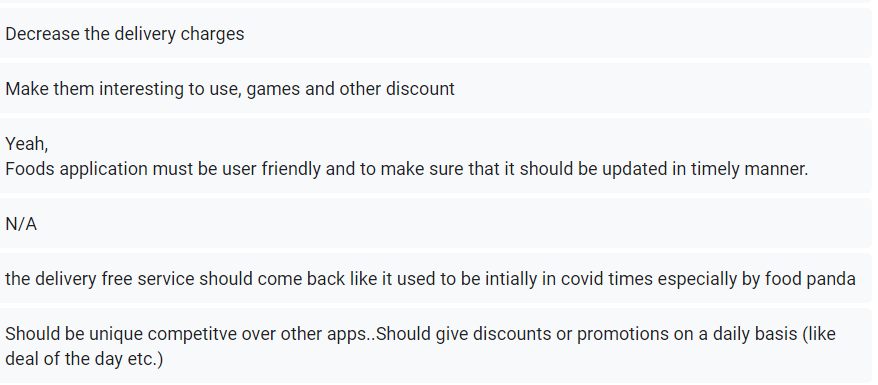
****

****

****

****

****

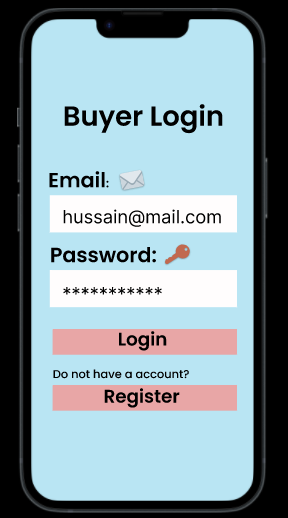
****

**Google Form Link**

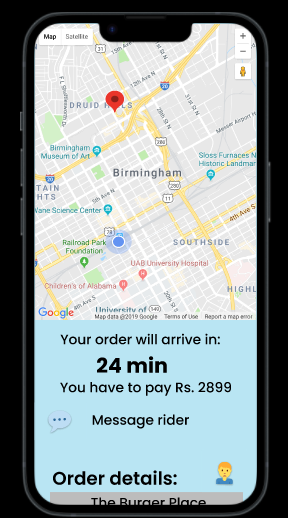
* [**Google Form Questions**](mailto:https://docs.google.com/forms/d/1w9554BC6DrkEMRICCQIHx-KjJWHCthx7KqDb1aXPLuQ/edit)
* [**Google Form Responses**](mailto:https://docs.google.com/forms/d/1w9554BC6DrkEMRICCQIHx-KjJWHCthx7KqDb1aXPLuQ/edit%23responses)

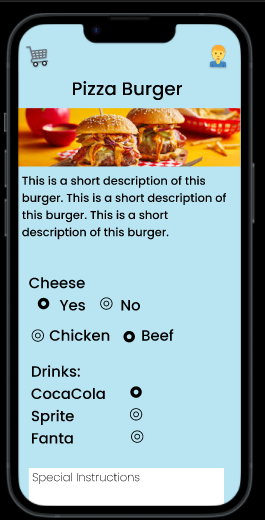
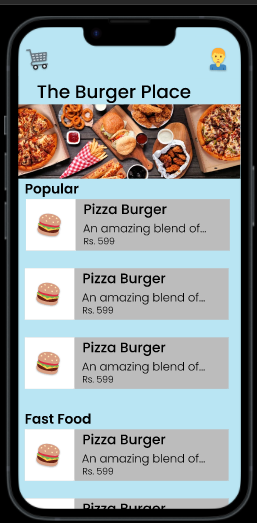
**3. Prototype**

**3.1. Customer Prototypes**

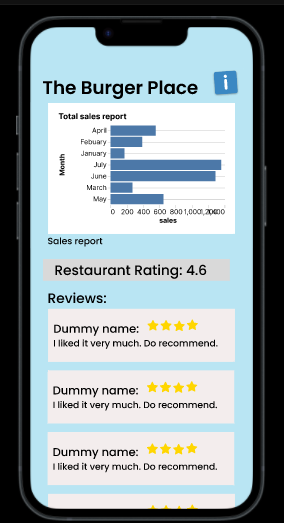
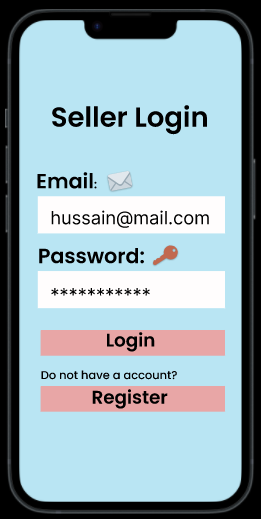
****

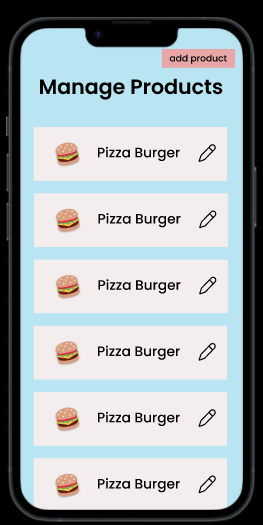
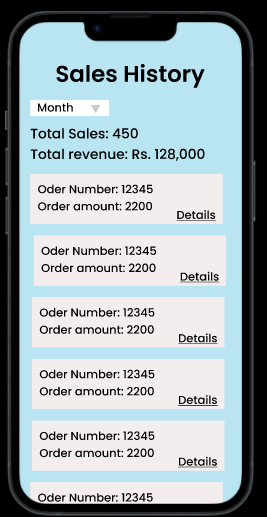
****

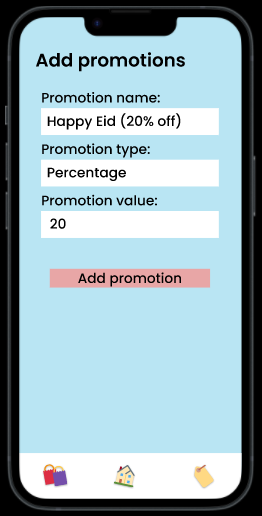
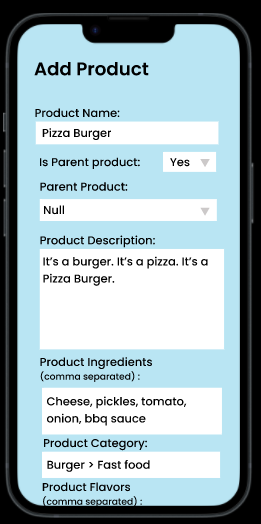
****

****

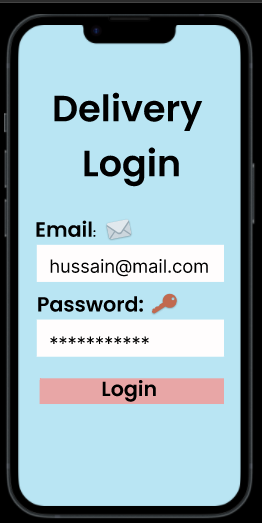
**3.2. Seller Prototype**

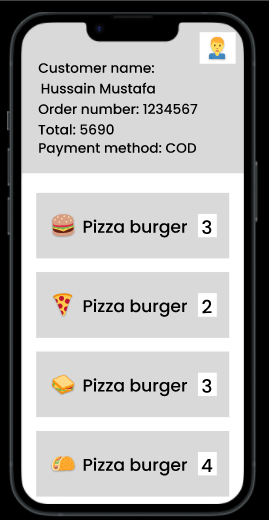
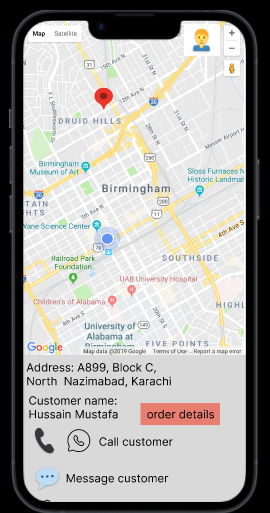
****

****

****

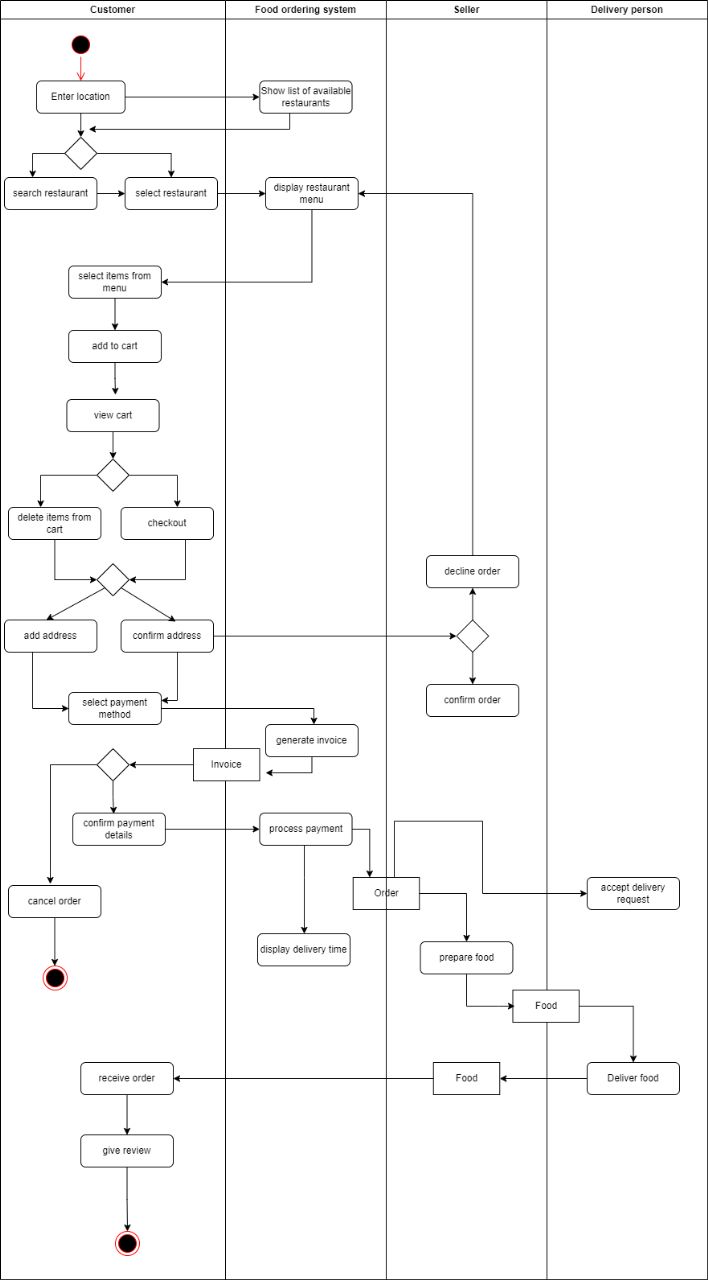
**3.3 Delivery Prototype**

****

****

****

**4. Swimlane**

****

**Figure 8**