

Project Documentation

**Project Title - Online Homemade spices and snacks delivery
system-SpiceLand**

TEAM MEMBERS	EMPLOYEE_ID
AKSHAY KUMAR K	5401
BHARATH H K	5404
DIVYASHREE M D	5407
IRFAN KATARKI	5409
YESHASWINI C	5426

TABLE OF CONTENTS

1 INTRODUCTION	
1.1 Purpose	3
1.2 Scope	3
1.2.1 Overview	3
1.2.2 Benefits	4
1.3 Structure.....	4
2 OVERALL DESCRIPTION.....	5
2.1 Product Perspective	5
2.1.1 User interfaces	5
2.1.2 Hardware interfaces.....	5
2.1.3 Software interfaces	5
3 REQUIREMENTS.....	6
3.1 Functional Requirements	6
3.1.1 Customer	6
3.1.2 Vendor.....	7
3.1.3 Delivery Person	7
3.2 Non-Functional Requirements	8
3.2.1 Safety.....	8
3.2.2 Security.....	8
3.2.3 Human engineering.....	8
3.2.4 Performance requirements.....	9
3.3 Technical Requirements.....	10
4 UML ANALYSIS MODELS	11
4.1 Use Cases.....	11
4.1.1 Use case descriptions	12
4.2 Class Diagram	15
4.2.1 Class descriptions	16

1 INTRODUCTION

The following section provides an overview of the Project Documentation for the subject Online Homemade spices and snacks Delivery System (HSOS). To begin with, the purpose of the document is presented and its intended SME's 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. To conclude, a complete document overview is provided to facilitate increased reader comprehension and navigation.

1.1 Purpose

The purpose of this DOCUMENTATION is to outline both the functional and non-functional requirements of the subject HSOS. In addition to said requirements, the document also provides a detailed profile of the external interfaces, performance considerations imposed on the subsequent implementation. It is the intention that the presented set of requirements possesses the following qualities: correctness, unambiguousness, completeness, consistency, verifiability, modifiability and traceability. Consequently, the document should act as a foundation for efficient and well-managed project completion and further serve as an accurate reference in the future. It will not only provide an extensive capacity for project planning and progress assessment, but it will further assist with senior developers/SME groups interactions.

1.2 Scope

In current formal food purchasing environments, some form of physical static menu is utilized to convey the available food item choices to customers where self-employed women are not able to reach all customers with their sales and marketing activities that are restricted to mouth publicity and experiences. This document specifies the requirements for an innovative online platform to register themselves to alleviate the problems associated with the current archaic method. Three related concepts are encompassed by the general scope of the Online food spices and snacks Ordering and Delivery System. The first pertains to the replacement of mouth publicity towards reaching a large set of consumers online, the second relates to a complementary electronic strategy for the front of house handling of a customer's order and the third surrounds the process of transferring said electronic orders to the kitchen for preparation.

1.2.1 Overview

The Online Homemade Spices and Snacks Delivery System is a software package to facilitate ordering and delivering within a digital platform. The customer can view the menu, place orders, make payments, and organize the final bill through the surface computer interface built into their table. Vendors can include their spices and snacks item into customer dashboard, control table functions remotely to assist customers, confirm orders, send orders to food preparation staff and finalize the customer's bill. The delivery-person interface keeps track of orders and gets the confirmed paid bill to deliver the ordered items to the customer.

Customers, Vendors, Delivery Person are presented with an attractive and easy-to-use surface computer GUI with the clickable's metaphor in their menus.

1.2.2 Benefits

Greater flexibility in menus, an increase in homemade spices and snacks productivity and capacity for extensive business auditing resulting in women empowerment are the primary benefits associated with the HSOS. Menu updates can be rolled out at any time with no extra labor from printing and distributing new menus, allowing for more dynamic pricing and content changes. With the underlying software system taking responsibility for a customer's order throughout its lifecycle, not only is accuracy ensured, but all actions are logged in a database for analysis.

1.3 Structure

The structure of this Software Requirements Specification is as follows. Section 2 presents an overall description of the subject HSOS. Attention is given to putting the resultant software product into perspective and further outlining end-user characteristics, system constraints and assumptions. Section 3 is devoted to the explicit specification of software requirements both functional and non-functional in nature. The functional requirements listed have been demarcated according to the categories of system users. For completeness, Section 4 extends upon Section 3 through the inclusion of UML analysis models and diagrams. To begin with, the identified HSOS use cases are given. In supplement, activity diagrams for each use case are presented along with an overall class diagram and relevant ER diagram.

2 OVERALL DESCRIPTION

The following section presents an overall description of the subject HSOS. In particular, the product has been put into perspective through a detailed assessment of the user, hardware, software and communication interfaces requirements.

2.1 Product Perspective

The software described in this documentation is the software for a complete HSOS system. The system merges various hardware and software elements and further interfaces with external systems. Thus, while the software covers the majority of the system's functionality, it relies on a number of external interfaces for persistence and unhandled tasks, as well as physically interfacing with humans.

2.1.1 User interfaces

There is one separate user interface used by the HSOS software, related to an interfaced physical hardware device.

Surface computer UI

The Surface Computer UI is the interface used by all three users i.e. customers, Vendors, delivery Person. This interface uses the surface computer paradigm - users interact with the system by dragging 'objects' around on the GUI display. Customers will summon their snacks and spices menu, which is combined with a system/command menu, and dismiss it with a similar gesture or by tapping a close button GUI element.

2.1.2 Hardware interfaces

There is one external hardware devices used by the HSOS, each related to a user interface. The device is the surface computers. This device must be physically robust and immune to liquid damage and stains. However, they should be fully capable computers that can use textual data from the server along with local UI/interpretation code to display UI elements and take input. All order and transaction records should be stored on the database server, not these computers. In all cases, the hardware device takes information from the HSOS and processes the information to display. It also provides user input information to the HSOS.

2.1.3 Software interfaces

The HSOS will interface with a Database Management System (DBMS) that stores the information necessary for the HSOS to operate. The DBMS must be able to provide, on request and with low latency, data concerning the spices and snacks menu. Additionally, it should take and archive data provided to it by the HSOS. This data will include records of all orders and transactions (system states and state changes) executed by the HSOS. The DBMS must store all data such that it can be used for accounting, as well as accountability.

3 REQUIREMENTS

The following section presents the complete set of functional and non-functional requirements identified for the subject HSOS. Functional requirements are listed first, according to their relationship to the overall system, customers, Vendors, delivery Person. The functional requirements have been specified using a natural language description and as such, the reader is directed to Section 4 (UML Analysis Models) for further detail.

3.1 Functional Requirements

This subsection presents the identified functional requirements for the subject HSOS. Initially, general requirements that pertain to the whole system are given. Where possible, subsequent requirements have been demarcated based on their relevance to the users of the system, that is, customers, Vendors, delivery Person.

3.1.1 Customer

Table 3.1.1 presents the identified functional customer requirements that directly relate to the customers of the subject HSOS.

Requirement	Description
01	A customer shall be able to register to application using required credentials
02	A customer shall be able to login to application
03	A customer shall be able to engage their menu
04	A customer shall be able to get menu of spices and snacks
05	A customer shall be able to navigate through the available items in the menu.
06	A customer shall be able to add an item to an order.
07	A customer shall be able to remove an item from the placed order.
08	A customer shall be able to place an order.
09	A customer shall be able to cancel an order through their engaged menu if it is pending and not yet placed.
10	A customer shall be able to finalize payment through their menu.
11	A customer shall be able to drag a items into a payment or a cash payment.
12	A customer shall be able to set location and complete payment and place order

Table 3.1.1 Functional Customer Requirements

3.1.2 Vendor

Table 3.1.2 presents the identified functional Vendor requirements that directly relate to the Vendor of the subject RMOS.

Requirement	Description
01	A Vendor should be able to log into the website using their assigned username and password.
02	A Vendor shall be able to log out of the application.
03	A Vendor shall be able to add Menu to system
04	A Vendor assigned to the application shall be alerted after: <ul style="list-style-type: none"> An order is placed from that customer Menu list
05	The application shall allow a Vendor to accept an order placed by a customer through website.

Table 3.1.2 Functional Vendor Requirements

3.1.3 Delivery Person

Table 3.1.3 presents the identified functional delivery Person requirements that directly relate to the delivery Person of the subject HSOS.

Requirement	Description
01	A delivery person shall be able to log into the website using their assigned username and password.
02	A delivery person shall be able to log out of the application.
03	A delivery person shall be able to view the current orders by customer.
04	A delivery person shall be able to get information about order, distance and location of customer
05	A delivery person shall be able to get information about order, distance and location of Vendor
06	The application shall allow a delivery person to accept an order placed by a customer
07	The application shall allow a delivery person to update the delivery of an item initiated to customer.
08	The application shall allow a delivery person to take ratings

Table 3.1.3 Functional Delivery Person Requirements

3.2 Non-Functional Requirements

This subsection presents the identified non-functional requirements for the subject HSOS. The subcategories of non-functional requirements given are safety, security, interface, human engineering, qualification, operational and maintenance.

3.2.1 Safety

Table 3.2.1 presents the identified non-functional safety requirements that directly relate to the entire subject RMOS.

Requirement	Description
F01	The system shall log every state and state change of every surface computer, tablet and display to provision recovery from system failure.
F02	The system shall be capable of restoring itself to its previous state in the event of failure (e.g. a system crash or power loss).
F03	The system shall be able to always display a menu to facilitate manual order taking should the need arise.

Table 3.2.1 Non-Functional Safety Requirements

3.2.2 Security

Table 3.2.2 presents the identified non-functional security requirements that directly relate to the entire subject RMOS.

Requirement	Description
Y01	A surface computer requires a user to log in.
Y02	The application shall require a user to log in using a username and password.
Y03	A display shall require a user to log in.

Table 3.2.2 Non-Functional Security Requirements

3.2.3 Human engineering

Table 3.2.3 presents the identified non-functional human engineering requirements that directly relate to the entire subject RMOS.

Requirement	Description
H01	Any element of the system will take no longer than 10-seconds to restart.
H02	A surface computer must not dismiss an engaged menu unless the customer requests it.

Table 3.2.3 Non-Functional Human Engineering Requirements

3.2.4 Performance requirements

Table 3.2.4 presents the identified non-functional performance requirements that directly relate to the entire subject HSOS.

Requirement	Description
P01	The server shall be capable of supporting no less than 200 concurrent connections from any combination of surface computers, tablets and displays.
P02	The server shall be capable of supporting an arbitrary number of surface computers, that is, it shall provide no limit on how many devices are in the system.
P03	The server shall be capable of supporting an arbitrary number of active meals/orders, that is, no meals/orders shall be lost under any circumstances.
P04	The server shall be capable of supporting an arbitrary number of active customer payments, that is, no payments shall be lost under any circumstances.

Table 3.2.4 Non-Functional Performance Requirements

3.3 Technical Requirements

This subsection presents the identified technical requirements for the subject HSOS.

- Spring Tool suite
- Java,
- Oracle Database
- React
- Tom cat server
- Maven
- JIRA

Spring Tool suite:

Spring Tool Suite (STS) is **a java IDE tailored for developing Spring-based enterprise applications**. It is easier, faster, and more convenient. And most importantly it is based on Eclipse IDE.

Java:

Java is **the official language for Android mobile app development**.

ORACLE Data base:

ORACLE is **an open-source lightweight Java database**. It can be embedded in Java applications or run in the client-server mode.

React:

- HTML is the code that is used **to structure a web page and its content**
- CSS is **the language for describing the presentation of Web pages, including colors, layout, and fonts**
- JS is used by programmers across the world **to create dynamic and interactive web content like applications and browsers**.

Tom Cat server:

Apache Tomcat is a web container. It **allows the users to run Servlet and JAVA Server Pages that are based on the web-applications**.

Maven:

Maven is used **for building, publishing, and deploying several projects**.

JIRA:

Jira **helps teams plan, assign, track, report, and manage work and brings teams together for everything from agile software development and customer support to start-ups and enterprises**.

4 UML ANALYSIS MODELS

4.1 Use Cases

This subsection extends upon the functional requirements given in Section 3.1 through the presentation of detailed use cases. To facilitate an unambiguous and clear view of how the end-users interact with the subject HSOS, the actors (end-users) involved in the use cases, a use case diagram and detailed use case descriptions are provided. The use cases that find representation are Log In, Log Out, Accept Order, Deliver Item, Process Bankcard Payment, Process Cash Payment, Place Order,, Pay Bill, Accept/Reject Item and Indicate Item Ready.

4.1.1 Use Case Descriptions

Table 4.1.1 presents the Log In use case description to show the interaction of the actors when logging into the system.

Use Case	Log In
Primary Actor	Customer, Vendor and Delivery person
Goal In Context	Enable Actors access to the system through the website.
Preconditions	The Actors has a valid username and password and is not already logged in
Scenario	1) The Actors selects 'Log In' from the website 2) The application prompts the user for their username and password 3) The Actors enters their username and password 4) The application enables access to the system according to access control
Exceptions	The Actors enters an invalid username or password

Table 4.1.1 Log In Use Case Description

Table 4.1.2 presents the Log Out use case description to show the interaction of the actors table when logging out of the system.

Use Case	Log Out
Primary Actor	Customer, Vendor and Deliver person
Goal In Context	Disable Actors access to the system through an application
Preconditions	The actors is already logged in
Scenario	1) The actors select 'Log Out' from the application 2) The application disables access to the system

Table 4.1.2 Log Out Use Case Description

Table 4.1.3 presents the place Order use case description to show the interaction between a customer and a surface computer when placing an order through the cart page.

Use Case	Place Order
Primary Actor	Customer
Goal In Context	Place an order from product items available
Preconditions	The customer has been set up the applications
Trigger	The customer wants to order menu items
Scenario	1) The customer brings up the list 2) The customer selects 'Order' from the product list menu 3) The customer navigates through the available items and adds the ones they want from the product list menu onto their order 4) The customer selects 'Place Order' from the menu and payment is done

Table 4.1.3 Place Order Use Case Description

Table 4.1.4 presents the Accept Order use case description to show the interaction between a Vendor and a delivery Person when accepting a new order placed by a customer.

Use Case	Accept Order
Primary Actor	Vendor and Deliver person
Goal In Context	Accept an order that has been placed by a customer
Preconditions	A customer has placed an order
Trigger	The Actors chooses to serve the customer
Scenario	1) The Actors selects 'Take Order' from the menu 2) The delivery Person confirms the order and tracks the order placed by the customer 3) The items in the order are sent to the packing 4) The order is accepted in customer's tracking

Table 4.1.4 Accept Order Use Case Description

Table 4.1.5 presents the Deliver Item use case description to show the interaction between a Delivery Person and the Vendor when delivering an order item to a customer.

Use Case	Deliver Item
Primary Actor	Delivery person
Goal In Context	Deliver the item order to its customer
Preconditions	An item ordered assigned to the Delivery person is packed and assigned
Scenario	1) The Deliver person reads address and go to location 2) The Deliver person delivers the item to the customer who ordered it

Table 4.1.5 Deliver Item Use Case Description

Table 4.1.6 presents the Process Bankcard Payment use case description to show the interaction between a delivery Person and customer when processing the bankcard payment of a customer bill.

Use Case	Process Bankcard Payment[MOCK PAYMENT]
Primary Actor	Delivery Person
Goal In Context	Charge a customer for their items ordered taking a bankcard payment
Preconditions	Items which are ordered have been assigned to the customer's bankcard payment bill
Trigger	The system sends an alert to the delivery Person notifying them the item is ready
Scenario	1) The waiter delivers the item to the customer who ordered it.

Table 4.1.6 Deliver Item Use Case Description

Table 4.1.7 presents the Pay Bill use case description to show the interaction between a customer and a surface computer when paying for their orders.

Use Case	Pay Bill
Primary Actor	Customer
Goal In Context	A customer decides to pay for their orders when they are ready to leave
Preconditions	Every item ordered by the customer has been delivered or cancelled
Trigger	The customer asks to finalize their bill
Scenario	<ol style="list-style-type: none"> 1) The customer double-taps the table to bring up a menu 2) The customer selects 'Ask for Bill' from the menu 3) The surface computer enters billing mode and displays a representation of each customer's bankcard payment. 4) The customer finalizes the assignment process and an alert is sent to the waiter assigned to the table 5) The customer double-taps the dismiss menu option on the table to close the menu
Exceptions	The customer cancels billing mode

Table 4.1.7 Pay Bills Use Case Description

Table 4.1.8 presents the Accept/Reject Item use case description to show the interaction between a chef and a display when accepting or rejecting an item requested by a customer.

Use Case	Increase/Decrease the quantity of Item in cart page
Primary Actor	Customer
Goal In Context	Functionality to Increase/Decrease the quantity of Item in cart page
Preconditions	The item has not already been accepted or rejected
Scenario	<ol style="list-style-type: none"> 1) The customer is willing to increase the number of items he/she wanted to purchase.

Table 4.1.8 Increase/Decrease Quantity of Item Use Case Description

4.2 Class Diagram

4.2.1 Class descriptions

The following subsection presents descriptions for the classes identified for the subject HSOS.

Products

This class represents the items from Vendor. A Product that is part of an order may contain a requirement. The product contains the menu product's name, price, description, status and an image or model to depict the product on a surface computer or display.

Cart

This class represents a collection of products. An Order can be in either placed, cancelled or the approved state. An Order will typically be deleted shortly after it has been approved.

Customer

This class represents the user who buys products and places orders.

Vendor This class represents the user who displays the products and adds into the list of products

Orders

This class represents the status of selected items and proceeds towards the payment page leading towards successful delivery of products to customers

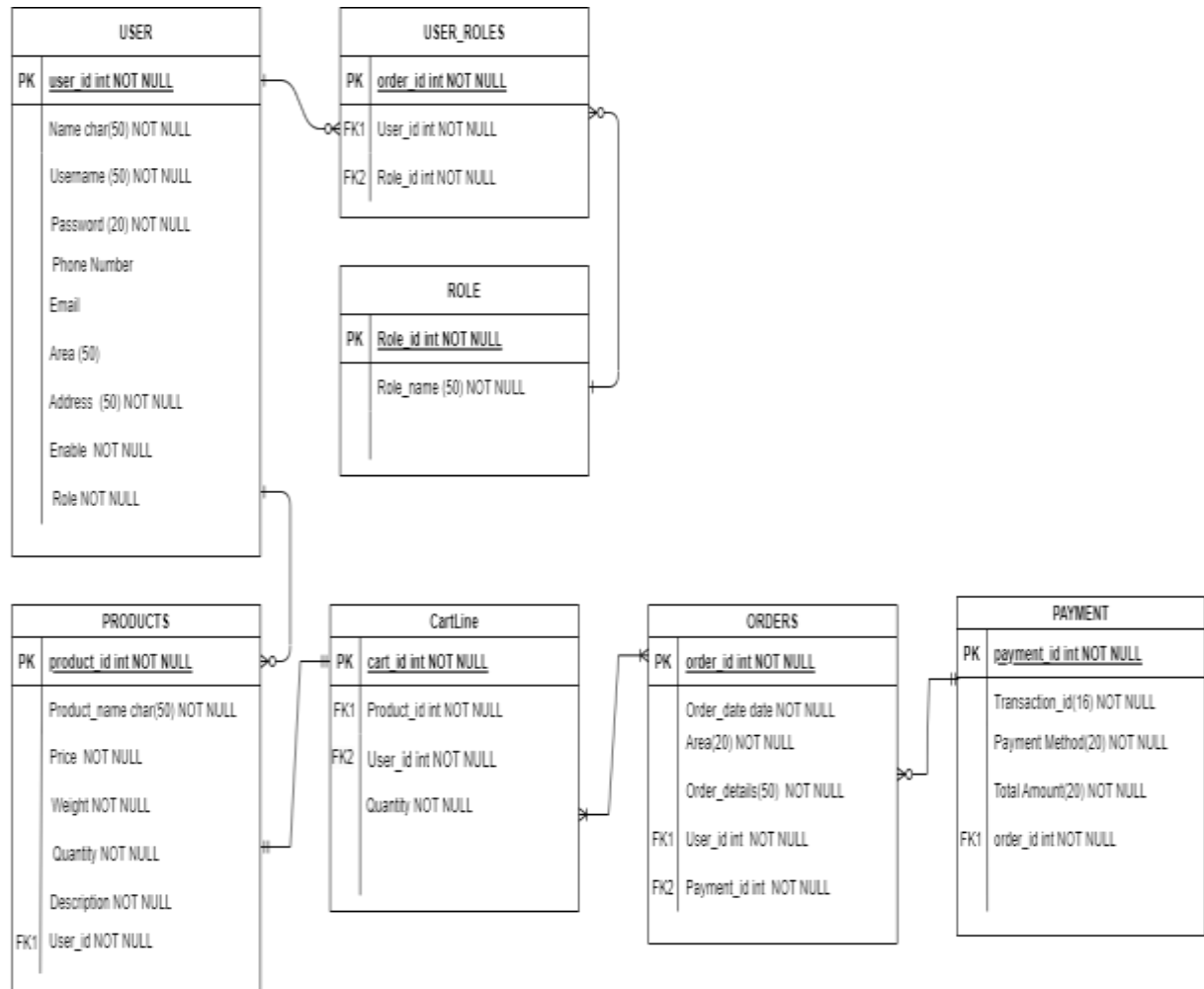


Figure 4.2.1 Online Food Spices and Snacks Ordering and Delivery Class Diagram

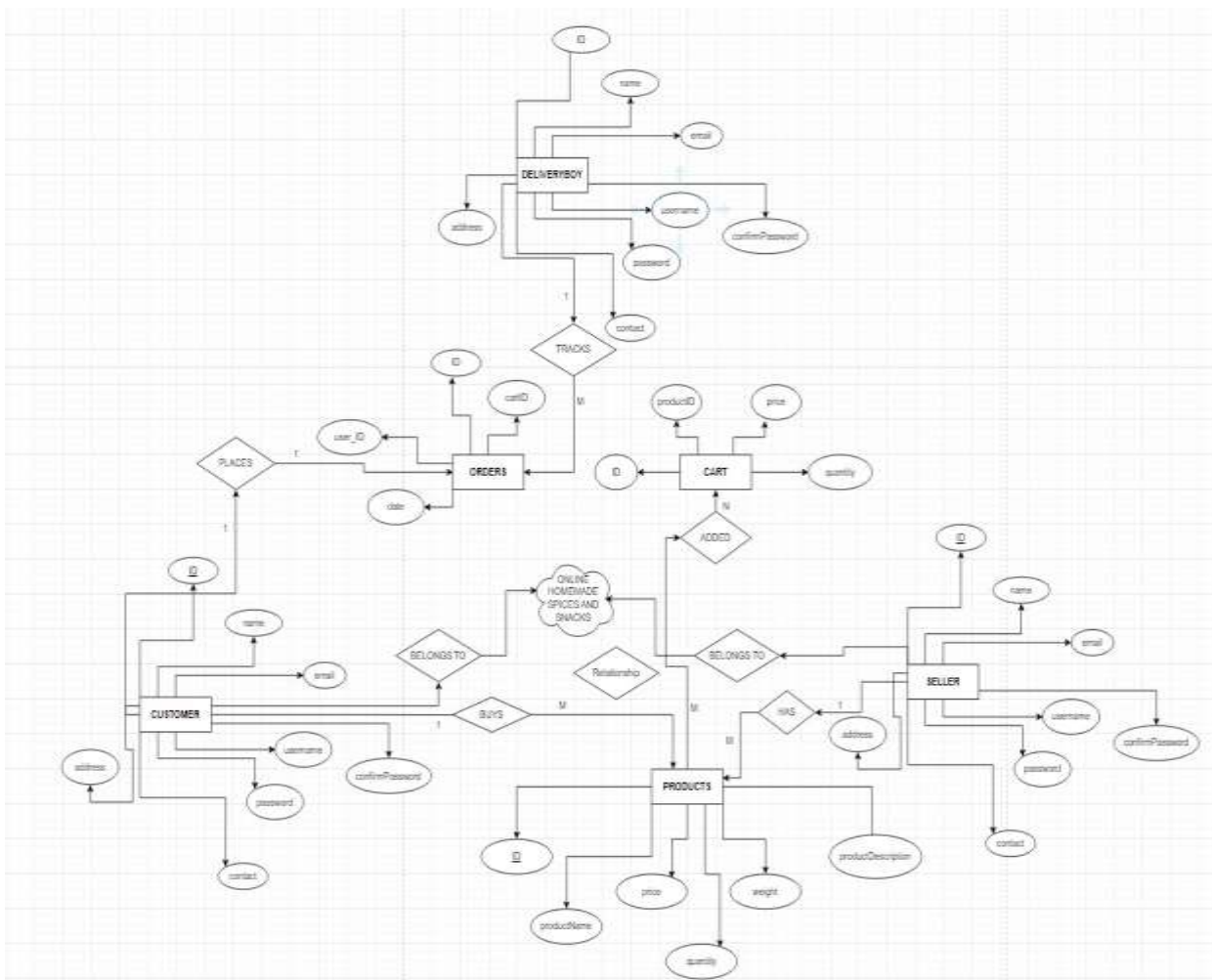


Figure 4.2.2 Entity-Relationship Diagram
