

**S**oftware **R**equirements **S**pecification

Project – THUNDER

Prepared by: Submitted To:

Team – **Action Kamen Dr. Ashima Singh**

1. Prateek Bansal (101917080)
2. Akshat Gupta (101917081)
3. Bhavesh Sareen (101917085)
4. Harsh Kashyap (101917088)



**Table of Contents**

1. [Introduction 4](#_heading=h.gjdgxs)
   1. Purpose of the document 4
   2. Scope of the Development Project
   3. Definitions, abbreviation[s and acronyms 4](#_heading=h.3znysh7)
   4. Intended Audience and Reading Suggestions
   5. [References](#_heading=h.tyjcwt) 4
   6. Overview
2. [Overall Description 6](#_heading=h.3dy6vkm)
   1. [Product perspective 6](#_heading=h.1t3h5sf)
   2. Product features/ functions 7
   3. [User Classes and Characteristics 7](#_heading=h.2s8eyo1)
   4. [Operating Environment 8](#_heading=h.17dp8vu)
   5. [User Documentation 8](#_heading=h.3rdcrjn)
   6. General Constraints, [Assumptions and Dependencies 8](#_heading=h.26in1rg)
3. System Features/ Specific requirements 9
   1. [Login 9](#_heading=h.35nkun2)
      1. [Description and Priority 9](#_heading=h.1ksv4uv)
      2. [Stimulus/Response Sequences 9](#_heading=h.44sinio)
      3. [Functional Requirements 9](#_heading=h.2jxsxqh)
   2. View Doctor Profile 9
      1. [Description and Priority 9](#_heading=h.3j2qqm3)
      2. [Stimulus/Response Sequences 9](#_heading=h.1y810tw)
      3. [Functional Requirements 10](#_heading=h.4i7ojhp)
   3. [Schedule Add/Update 10](#_heading=h.2xcytpi)
      1. [Description and Priority 10](#_heading=h.1ci93xb)
      2. [Stimulus/Response Sequences 10](#_heading=h.3whwml4)
      3. [Functional Requirements 10](#_heading=h.2bn6wsx)
   4. [Search on Homepage 10](#_heading=h.qsh70q)
      1. Description and Priority 10



|  |  |  |
| --- | --- | --- |
| 3.4.2 | Stimulus/Response Sequences | [11](#_heading=h.1pxezwc) |
| 3.4.3 | Functional Requirements | [11](#_heading=h.49x2ik5) |
| [3.5](#_heading=h.2p2csry) | [Register Screen](#_heading=h.2p2csry) | [11](#_heading=h.2p2csry) |
| [3.5.1](#_heading=h.147n2zr) | [Description and Priority](#_heading=h.147n2zr) | [11](#_heading=h.147n2zr) |
| [3.5.2](#_heading=h.3o7alnk) | [Stimulus/Response Sequences](#_heading=h.3o7alnk) | [11](#_heading=h.3o7alnk) |
| [3.5.3](#_heading=h.23ckvvd) | [Functional Requirement](#_heading=h.23ckvvd)s | 11 |
| 4. External Interface Requirements |  |  |
| [4.1](#_heading=h.32hioqz) | [User Interfaces](#_heading=h.32hioqz) | [12](#_heading=h.32hioqz) |
| [4.2](#_heading=h.1hmsyys) | [Hardware Interfaces](#_heading=h.1hmsyys) | [12](#_heading=h.1hmsyys) |
| [4.3](#_heading=h.41mghml) | [Software Interfaces](#_heading=h.41mghml) | [12](#_heading=h.41mghml) |
| [4.4](#_heading=h.2grqrue) | [Communication Interfaces](#_heading=h.2grqrue) | [12](#_heading=h.2grqrue) |

1. [Non-Functional requirements 13](#_heading=h.vx1227)
   1. [Performance Requirements 13](#_heading=h.3fwokq0)
   2. [Safety Requirements 13](#_heading=h.1v1yuxt)
   3. [Security Requirements 13](#_heading=h.4f1mdlm)
   4. [Software Quality Attributes 13](#_heading=h.2u6wntf)
2. [Other requirements 13](#_heading=h.19c6y18)
3. Change History
4. Document Approvers

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for Changes** | **Version** |
| THUNDER FOOD DELIVERY | 20-06-2021 | Initial Release | V1.0 |

# **Introduction**

## **Purpose**

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality.

This SRS describes the software functional and nonfunctional requirements for release 1.0 of the Thunder Food Delivery Application. This document is intended for users of the software and also potential developers. To conclude, a complete document overview is provided to facilitate increased reader comprehension and navigation.

* 1. **Scope of the Development Project**

In current formal dining environments, some form of physical static menu is utilised to convey the available food and beverage choices to customers. Three related concepts are encompassed by the general scope of the Restaurant Menu and Ordering System. The first pertains to the replacement of paper-based menus using an electronic format, 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.

During the severe times of corona and its after effects, human contact and public gathering should be as less as possible. It is very evident that

minimizing the gathering of people at restaurants and recreational places is highly unlikely to keep in check.

To adhere to the above precautions, we suggest that people especially on

campus students restrict their movements in entertainment places.

Therefore, for fulfilment of this purpose we are working on bringing an

application that could cater to your market needs and especially food

available in the COS market.

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 and accountability of staff.

Our objective is:

● To minimize social interactions and public gatherings.

● To dispense a source of livelihood to various workers (e.g.

E-rickshaws).

● To save time for students during examinations.

● To provide teachers and staff, the various facilities available in COS

at their doorsteps.

● To boost the income of various restaurants and workers.

Therefore, we believe that such a platform will be in favour of every section of the institute.

* 1. **Definitions, abbreviation**[**s and acronyms**](#_heading=h.3znysh7)

|  |  |
| --- | --- |
| **TERM** | **DESCRIPTION** |
| Item | Single serving of food/beverage |
| Order | Comprises one or more items |
| Meal | Comprises one or more orders (associated with one customer) |
| Customer  Staff | Restaurant patron that orders/pays for a meal  General restaurant employee |
| Waiter | Staff member whose primary job is to take orders/serve meals to customers |
| Chef | Staff member whose primary job is to prepare items |
| Supervisor | Staff member whose primary job is to manage restaurant operations |
| Table | Comprises one or more seats at which customers sit and place orders from |
| Account | Comprises all the meals from a table |
| Payment | Comprises the total cost of zero or more meals and zero or more tips |
| Server | Backend computer that hosts the restaurant menu and ordering system |
| Surface Computer | Built into tables to provide customers with menu/ordering functionality |
| Tablet | Wireless mobile computer to provide staff with customer serving functionality |
| Display | Touch screen to provide a means for chefs to interact with the system |
| Register System | Point of sale terminal for handling bill payments |
| Bankcard | Customer debit/credit card |
| Menu | Surface computer representation of the available items and other option |

**Table 1.3.1 System Terminology**

|  |  |
| --- | --- |
| **Acronym** | **Description** |
| SRS | Software Requirement Specification |
| RMOS | Restaurant Menu and Ordering System |
| DBMS | Database Management System |
| LAN | Local Area Network |
| IP | Internet Protocol |
| API | Application programming interface |
| CFD | Context flow diagram |
| DFD | Data flow diagram |
| PDF | Portable document format |
| DB | Database |
| ML | Machine Learning |
| NLP | Natural language processing |
| MongoDB | Mongo DataBase |

**Table 1.3.2 System Acronyms**

**1.4** **Intended Audience and Reading Suggestions**

This document is intended for developers, testers, project managers, users, and the customer to understand the purpose, operation, and requirements of the project. The specific intentions for the various readers and the suggested reading sequences are the following:

· Product Manager - Understand use and requirements of the product to help guide the development of said product.

·Developers – Understand the requirement of the product and guidance to the implementation of the requirements.

·Testers - To understand the purpose, requirements and find bugs of the product to guarantee proper functioning of the product.

·Programmers who are interested in working on the project by further developing it or fixing existing bugs.

The Overall Description section of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification.

Requirements Specification sections of this document are written primarily for the developers and describe in technical terms the details of the functionality of the product.

**1.5 References:**

* <https://en.wikipedia.org/wiki/Online_food_ordering>
* <https://en.wikipedia.org/wiki/Swiggy>

## **1.6 Overview:**

The remaining sections of this document provide a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product. Section 2 gives the functional requirements, data requirements and constraints and assumptions made while designing the multi-utility system. It also gives the user a viewpoint of product use. Section 3 gives the specific requirements of the product. Section 3.0 also discusses the external interface requirements and gives detailed description of functional requirements.

In particular, the product will be put into perspective through a detailed assessment of the system, user, hardware, software and communication interfaces, memory considerations, operational modes and site adaptation requirements. Further, characteristics of the system’s end-users are discussed along with the identified system constraints and assumptions. To conclude the section, an apportioning of requirements has been outlined.



# **Overall Description**

The following section presents an overall description of the subject RMOS. In particular, the product will be put into perspective through a detailed assessment of the system, user, hardware, software and communication interfaces, memory considerations, operational modes and site adaptation requirements. Further, characteristics of the system’s end-users are discussed along with the identified system constraints and assumptions.

## **Product perspective**

Thunder is an online system which provides an ease in establishment of connection between the delivery agents , the customer and the hotel owners. The Thunder Food Delivery System is a new system that replaces the current manual and telephone processes for ordering and picking up lunches in the dining restaurants and COS of Thapar Institute of Engineering and Technology. The context diagram in Figure 1 illustrates the external entities and system interfaces for release 1.0. The system is expected to evolve over several releases, ultimately connecting to the Internet ordering services for several local restaurants and to credit and debit card authorization services.

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. 



## **Product Features/ Functions:**

Our project has its own unique features. Apart from all the aims and objectives, as well as various issues that it intends to eliminate, we are minutely working on various salient features that we wish to provide in this application.

1. Providing users with an inbuilt wallet for quick purchase.

2. Quick and efficient contact between user and shops.

3. Quick and efficient contact between user and delivery agents.

4. Easy to use interface for people of all ages.

5. Light and smooth ordering features.

6. Software confirms meal order or requests to modify meal order.

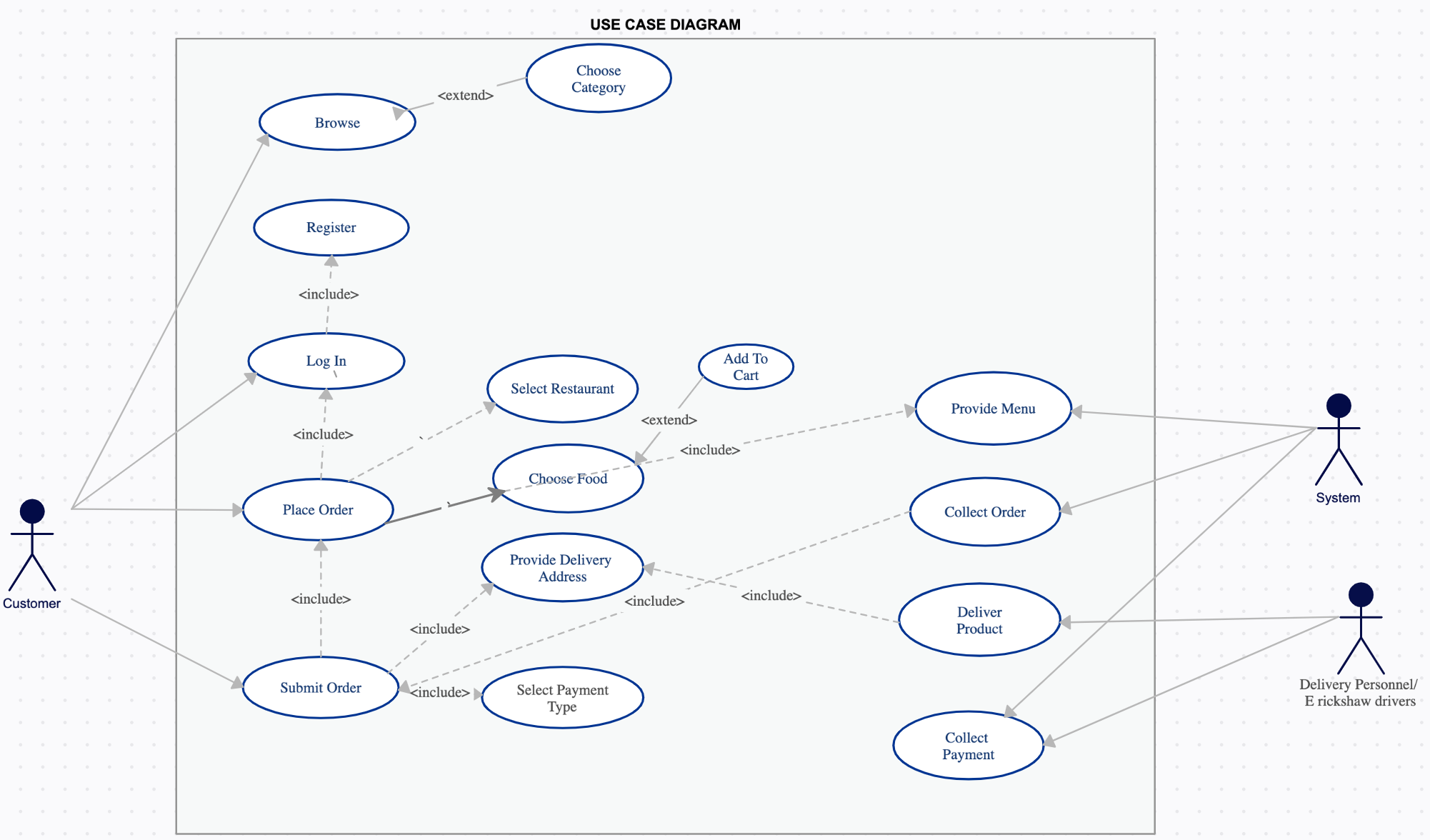
8. Software selects a delivery time and specifies the delivery location.

9. Software specifies payment method.

10. System confirms acceptance of the order.

12. System stores orders in the database and sends messages to notify COS, sends food item information to COS, and updates available delivery times.

Software accesses the System from the corporate intranet or from home, optionally views the menu for a specific date, selects food items, and places an order for a meal to be delivered to a specified location within a specified time window.



## **User Classes and Characteristics**

In our system we have only three user classes which are as:

* Admins (Restaurants)
* Users (Students/Faculty)
* Delivery Agents

These user types are listed below as follows:

**1. *Student and faculty Staffs***

A Customer is someone who wishes to order meals to be delivered from the company cafeteria. There are a lot of potential customers who are expected to use the COS an average of 4 times per week each.Their will surge in delivery during birthday parties and examinations.

***2. Delivery Agents***

As the restaurant prepares orders for delivery, they will print delivery instructions and issue delivery requests to the Meal Deliverer, who is either another cafeteria employee or a contractor. The Meal Deliverer will pick up the food and delivery instructions for each meal and deliver it to the customer.

***3. Restaurant workers.***

The Process Impact cafeteria currently employs about 20 Cafeteria Staff, who will receive orders from the Cafeteria Ordering System, prepare meals, package them for delivery, print delivery instructions, and request delivery.   
The end-users of the Software fall into three primary categories, unskilled, partly skilled and highly skilled.  
   
***Unskilled user***   
The users of the surface computers are walk-in customers and should therefore be assumed to have no relevant prior skills or education other than basic abilities to operate an automated system; no more complex than a parking meter or vending machine.  
   
***Partly skilled user***  
   
The users of the tablets and displays are waiters and chefs respectively and they should be able to use the system and further be able to train others with minimal training themselves. They must be able to explain all elements of the user interfaces except the server.

***Highly skilled user***

Students and staff who are well trained and know to use the software pretty well.

## **Operating Environment**

The project is a web application that can work on any of these devices with internet connections. Devices with the latest hardware are optimal as compared to the old hardware for accurate results. The device must have at least 1GB RAM and should have working internet connection. The mobile device must be compatible with android version 6.0 (Marshmallow) and above.

• Apple Safari 7+

• Google Chrome 44+

• Microsoft Internet Explorer 10+

• Mozilla Firefox 40+

## **User Documentation**

A separate, printed user manual for each user role should be delivered with this software which will document how the software should be used and maintained. Additionally, a tutorial shall be provided for the users describing the major use cases for each role.

## **Assumptions and Dependencies**

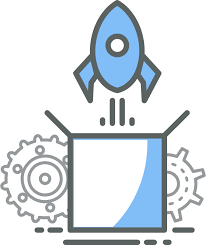
Following are the assumptions and dependencies:

1. The product must have a user friendly interface that is simple enough for all types of users to understand.
2. Response time for loading the software and for processing a transaction should be no longer than five seconds.
3. A general knowledge of basic computer skills and of basic working of an online payment system is required to use the product.
4. The central database server and backup database servers should be updated regularly. This updating and replication of data from the central database server to the backup database server can introduce additional latency in the working of the system.
5. User has a valid email address, phone number and knows basic functioning of mobile phones.
6. User has accepted all the permissions required to use the app
7. Location services may be required.
8. The system must provide a capacity for parallel operation and system design should not introduce scalability issues with regard to the number of surface computers, tablets or mobile displays connected at any one time.
9. The end system should also allow for seamless recovery, without data loss, from individual device failure.
10. The SRS assumes that none of the constituent system components will be implemented as embedded applications.
11. The implication is that the target hardware will provide a capacity for standalone program/application deployment .
12. It is further assumed that tablet PCs / smartphones of sufficient processing capability and battery life will be utilised.

**2.7 Apportioning of Requirements:**

The THUNDER software (including ordering and online payment system) is to be implemented in the following three phases:

* **Pilot Phase:** Here the THUNDER software will be implemented in the one hostel including 2-3 restaurants in COS market with the help of 40 students and 2 delivery agents. Initially we will be providing access privileges for two types of users: student and restaurant staff as they will be ones most involved in this phase.



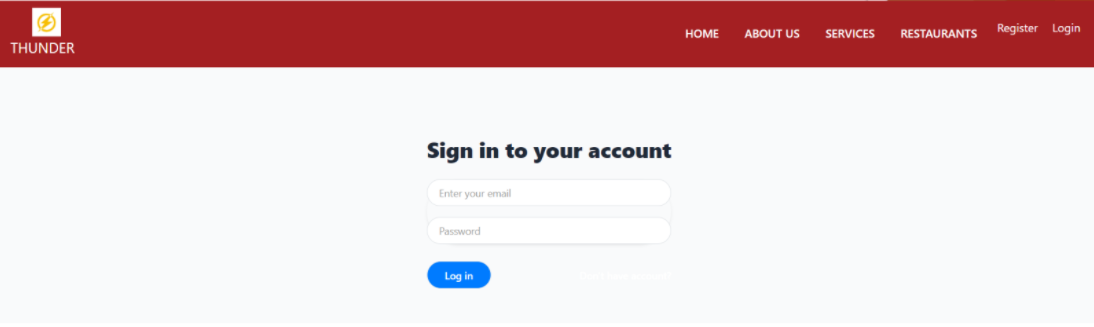
* **Institute wide deployment:** Following the successful completion of the pilot phase, we plan to deploy the same across the whole campus of Thapar University. More security measures will come into phase while deployment across the whole campus.
* **Extension of THUNDER to other Institutes:** In the future we can extend THUNDER software to other colleges/institutes with the help of respective college authorities which will in turn increase the revenue of their market shops.



# **System Features**

This section presents the identified functional requirements. 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, delivery agents and restaurant workers.

## **Login**



### Description and Priority

The login form is used by all the customers and admin. This module has the highest priority when compared to all the other modules. This model allows the customers to enter their respective username and password in order to make use of the software.

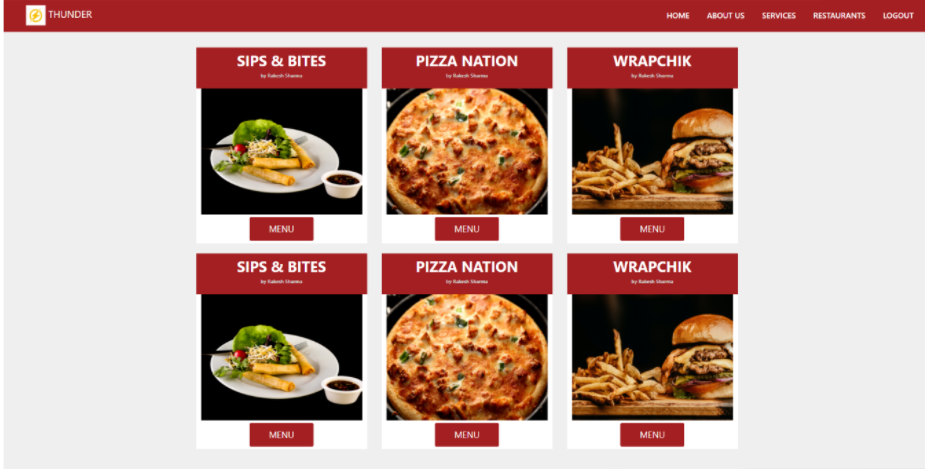
### Stimulus/Response Sequences

This module has text boxes where the customers can enter their username and password. If the necessary information is not provided or if invalid inputs are given by the customer, then the system will generate an error.

### Functional Requirements

Only authorized users are allowed to login. The authorized users are the registered customers and the admin. If invalid user name or password is given the system should inform the user. If unauthorized users try to access then it should not allow the user to enter in the software and make changes.

## **View Restaurant Profile**



### Description and Priority

All customers, admin will be able to view the information added or updated by the restaurant admin. This feature has higher priority because this is one of the most essential features of the product to provide information about the restaurant availability. A customer shall be able to navigate through the available items in their engaged menu.

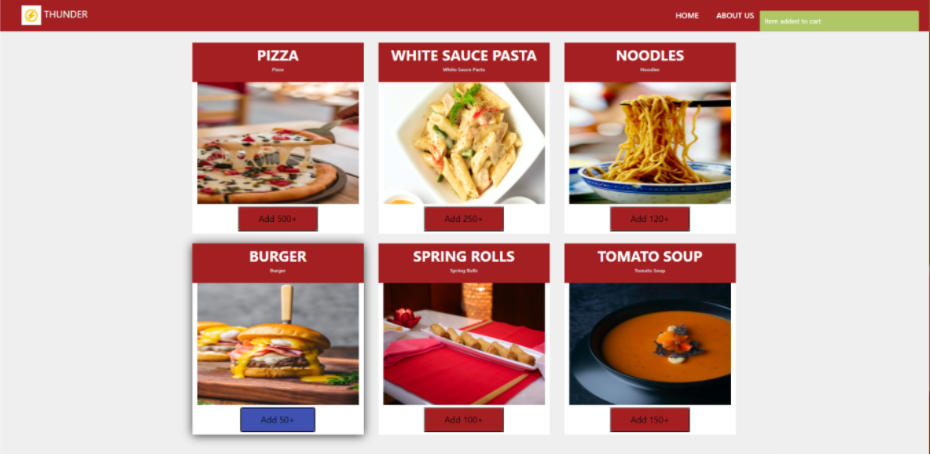
### Stimulus/Response Sequences

Once the customer searches a new restaurant page is opened, and further customers will click on respective restaurants. Based on the click, the portal will fetch the complete dishes of the respective restaurant from the database.

### Functional Requirements

* + - * In the profile page, the user will be able to view all the information added by the admin.
      * If the Restaurant is not currently available the order will be not placed and all details related to the restaurant will be available.

## **Schedule Add/Update**



### Description and Priority

This module is used by the restaurant admin who is responsible for adding new food items, updating the existing and deleting the old ones from the system. This feature has higher priority because this is one of the most essential features of the product to add information about the availability of food items.

### Stimulus/Response Sequences

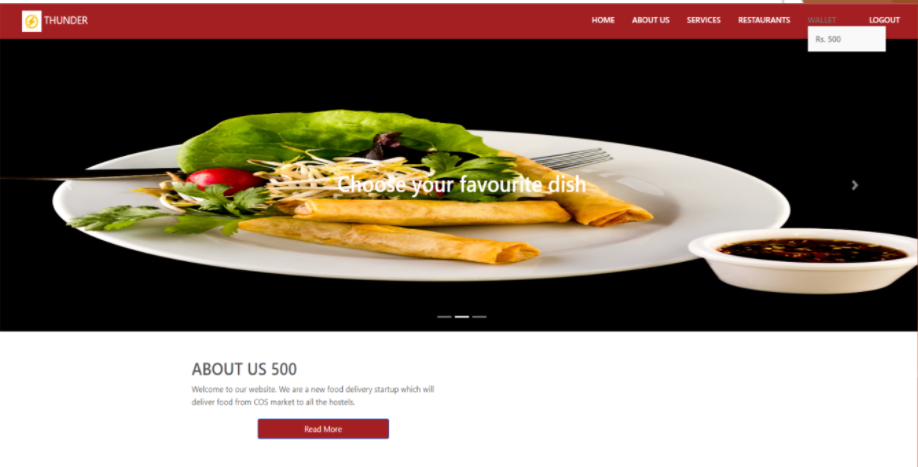
Once the required admin login into the system, one of the two sub-flows is executed.

* + - * Add new food items
      * Update and delete the existing food item.

### Functional Requirements

* + - * In “Add new schedule”, restaurant admin can enter information about new food items.
      * In “Update and delete food items”, admin can update the existing information about the food items.

## **Search on Homepage**



### Description and Priority

This module is used by the customers to categories and select the restaurants of their interests. Customers can enter their specific dish in the search menu provided to find food items of their interest. This module has the medium priority when compared to all the other modules as it may not be able to satisfy each customer’s needs.

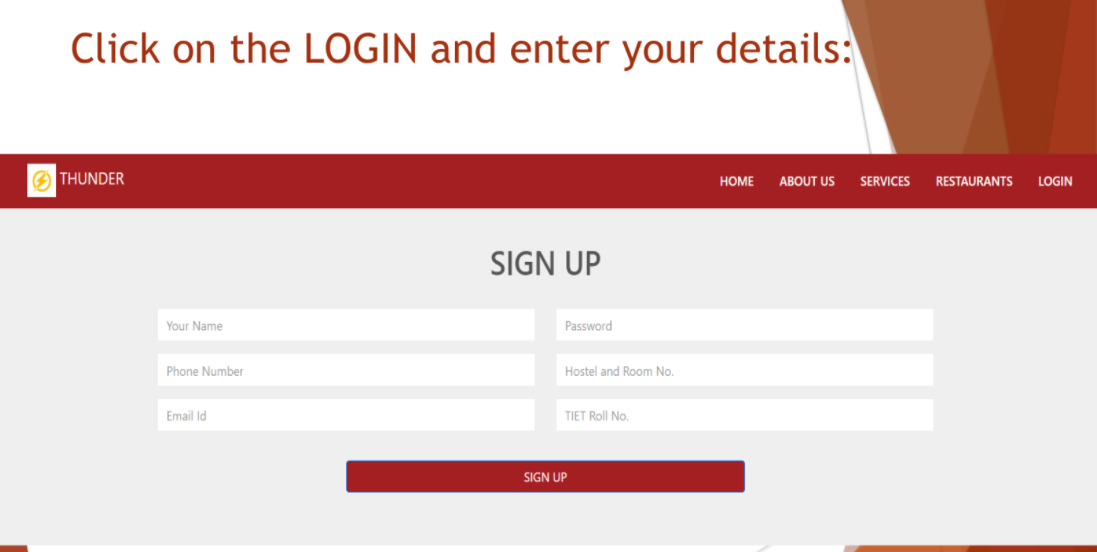
### Stimulus/Response Sequences

This module has a search box where the customers can enter their query, also single-click search is also available.

* + 1. **Functional Requirements**

If the query entered by the user(customer) matches the specific food items of the restaurant then the customer names will be displayed on the restaurant's order page. If users enter invalid food items then an empty list will appear.

* 1. **Register Screen**

****

* + 1. **Description and Priority**

This module is used by customers to register themselves into Thunder. The aim for this screen is to make an individual register on our database. The customer has to feed authentic info into the form and make himself register into the website. This module has high priority from the customer's point of view as without this they won't be able to order from our software.

* + 1. **Stimulus/Response Sequences**

This module has several input boxes or text fields where the customers have to enter their details that too the details should be officially i.e all details belong to the Thapar students. The customers should fill the info one by one in a sequence. If all input conditions are satisfied while registering then a message of successful registration will be shown else error will be thrown.

* + 1. **Functional Requirements**

Inputs should be concise, complete and unambiguous. If any field is left blank, the error will be highlighted while registering and it has to be filled. If no error/no field left blank then the customer will be registered into our database.



1. **External Interface Requirements**
   1. **User Interfaces**

* Anyone having this web application on his/her phone/laptop would be able to access the services.
* Our web application will provide a single click search feature for easy and fast search. A new user has no difficulty in searching a restaurant / food / cuisine of his/her own choice.
* The menu and offers displayed in the web app is only done by registered restaurant representatives / supervisors.
* Searches should be viewed as big tiles images having all major info about the restaurant / dish such as image of search, rating, cuisine and availability along with the cost per person
* Click on tile should lead to the respective restaurant page.
  1. **Hardware Interfaces**
* This web application should be viewed/accessed on all latest web browsers irrespective of the machine.
  1. **Software Interfaces**
* This application will be available for both android and iOS ecosystems. Anyone using an iPhone or an android device or a laptop or desktop will be able to use it.
* NodeJS is used as a backend. And JS / FLUTTER li ke interface is used for frontend.
  1. **Communication Interfaces**
* Basic HTTPS requests are used to communicate with the server.
* Basic CRUD Operations



1. **Non-Functional requirements**
   1. **Performance Requirements**

* The system should be available all the time and performance should not be degraded with the increase in number of users.
* If any maintenance needs to be done or unable to process the requests, then a proper message should be shown.
* Webpages should load in less than 5 seconds.
  1. **Safety Requirements**
* The details must be maintained and users must be authenticated.
* The customers shall be able to change his password of the account with the help of email address provided.
* The application shall have 2 step verification if the restaurant's admin wants additional security of the payment option.
  1. **Security Requirements**
* The login system of the application shall use robust firewalls which can protect against malicious security threats.
* The login system shall also not use default credentials and shall allow restaurants admin to change the food items as and when needed.
  1. **Software Quality Attributes**
* The application’s interface will be very simple and easy to use so that customers would be able to use it with ease and enjoy its services.
* The application shall be linked to the admin's email address so that he may be contacted as and when required.
* The customers shall be able to access his account from any device by providing valid credentials.

**6. Other requirements**

There are no other requirements for this project.

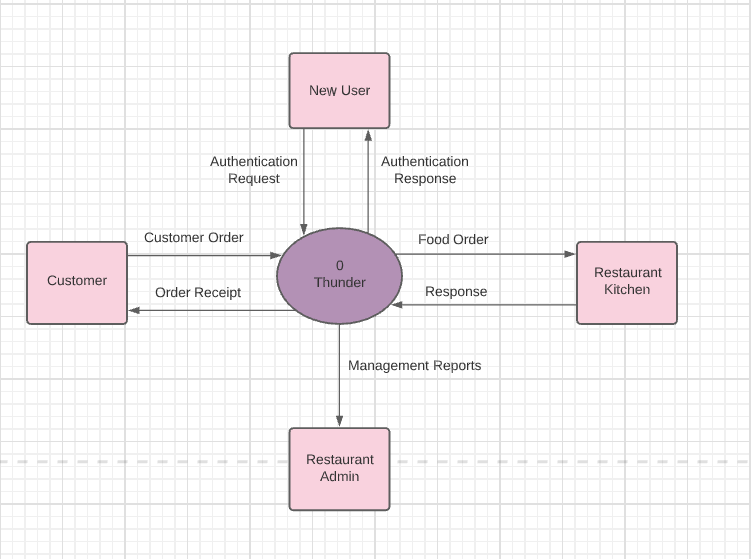
**7. Change History**

This will be the initial release of the project and this will be Version 1.0.

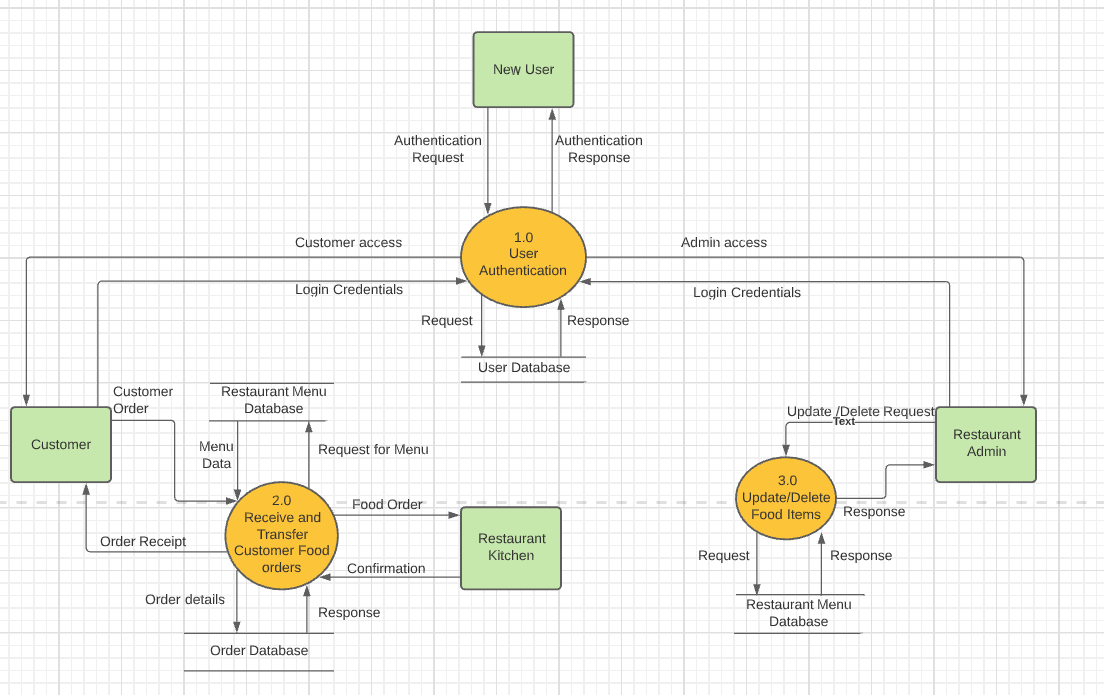
No further release as of now is scheduled yet.

**8. DFD (DATA FLOW DIAGRAM):**

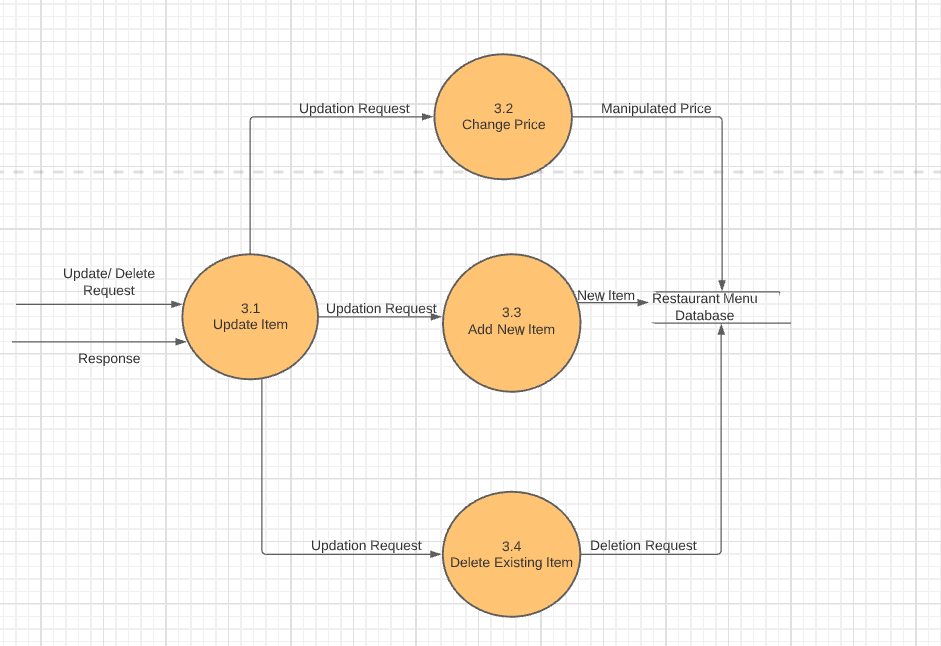
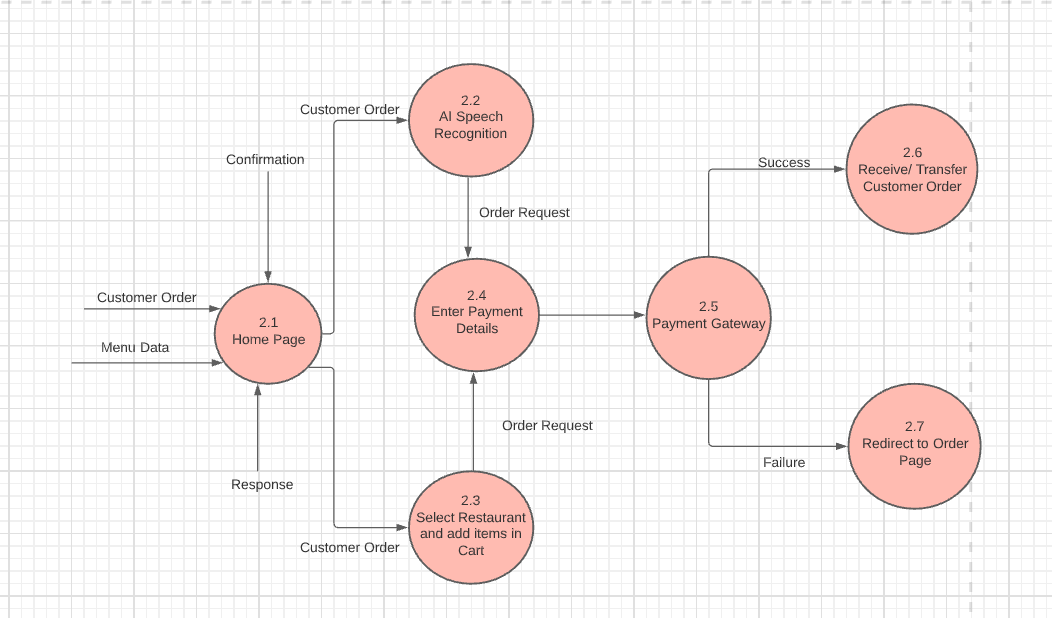
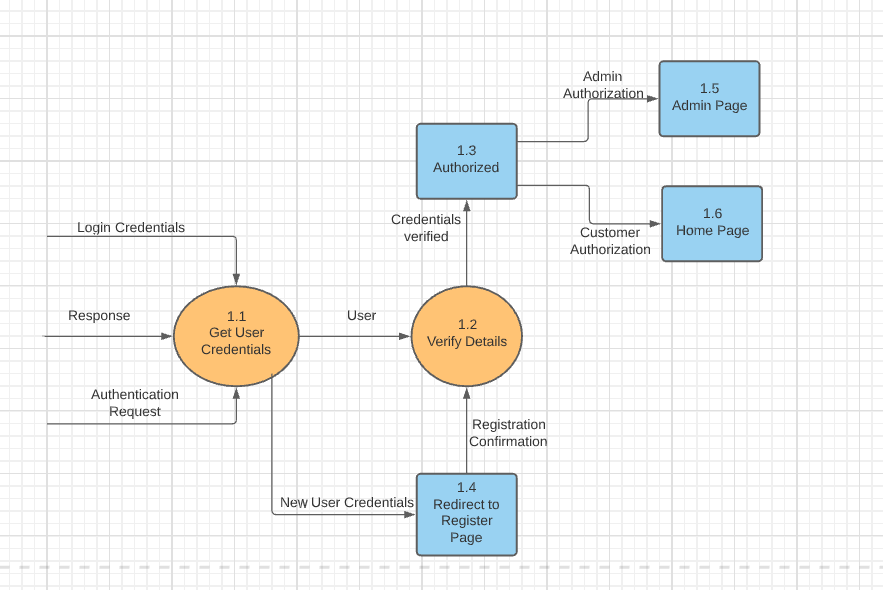
*LEVEL 0: CFD (Context Flow Diagram)*

****

*LEVEL 1:*

****

*Level 2:*

**

**9. Document Approvers**

SRS forThunder Food Delivery Application System approved by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(name)

Designation Date: