**Tasty-Treat Online Food Delivery System**

**Software Requirement Specification Document**

1. **Introduction**

The Online Food Delivery project aims to digitalize the process of various restaurant management operations including ordering. This document aims to capture the system requirements and features particularly related to ordering to be implemented.

* 1. Purpose of the project

Purpose of the project is digitalize the process of restaurant management operations including ordering. The system also allows to quickly and easily manage their food orders in an online menu where customers can browse on the internet and use it to make the orders with just a few clicks by finger. Restaurant manager will use these orders list which comes from customers through an easy to navigate graphical interface for efficient processing all the orders.

* 1. Scope

The software system being produced is called Online Food Delivery System. It is been produced for a customer interested in Buying Food items via the Internet. This system is largely available for anyone who have the Internet connection. The system will be run on a central server and distribute the remote user interface through a web browser to all the devices and also useful to various restaurant management operations.

# System Requirements

* 1. **Functional Requirements**
  + Access Website :

User should be able to access web-application through either an application browser or similar service on the mobile phone or computer.

* + Registration :

Given that user and admin has accessed web-application, then both should be able to register through the web-application. The user must provide first name, last name, email, contact number, username and password.

* + View Profile :

In which user and admin can see their current profile status.

* + View Product Details & add to cart:

Users can see the product details by login to the web application. e.g., Product category, Product items, Quantity, etc.

* + User log-in:

Given that the user has registered, then the user should be able to login to the web-application. The login information will be stored on the database for future use.

* + Search product in product list:

Specific product can be viewed in the product list. Users can search product by using product category.

* + Update product in product list:

Restaurant Manager can update their food products as well as add new Products. Also can give discount and Offers to Products. Specific product can be viewed in the product list.

* + Assign Delivery:

Restaurant Manager can assign delivery to delivery person as per customer order.

* + Deliver Food order to Customer:

Delivery person delivers order to particular customer as assigned by restaurant manager

# Non-functional Requirements

1. Security

The system’s back-end servers shall only be accessible to authenticated administrators. Sensitive data will be encrypted before being sent over insecure connections like the internet.

1. Availability

The system should be available at all times, meaning the user can access it using a web browser, only restricted by the downtime of the server on which the system runs. In case of an of a hardware failure or database corruption, a replacement page will be shown. Also, in case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the administrator. Then the service will be restarted. It means 24 X 7 availability.

1. Reliability

The reliability of the overall program depends on the reliability of the separate components. The main pillar of the reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes. Thus, the overall stability of the system depends on the stability of container and its underlying operating system.

1. Maintainability

A commercial database is used for maintaining the database and the application server takes care of the site. In case of a failure, a re-initialization of the program will be done. Also, the software design is being done with modularity in mind so that maintainability can be done efficiently.

1. Accessibility

The system will be a web-based application it is going to be accessible on the web browser.

1. Back up

We will take a backup in our system database. In order to enable the administrator and the user to access the data from our system.

1. Performance

The product shall be based on web and has to be run from a web server. The product shall take initial load time depending on internet connection strength which also depends on the media from which the product is run. The performance shall depend upon hardware components of the client/customer.

8. Supportability

The source code developed for this system shall be maintained in configuration management tool.

1. **Features** 
   1. **Password Encryption**

Encrypted password saved in database after successful sign up by customer (Serialization) and automatically decrypted while sign in (De-Serialization)

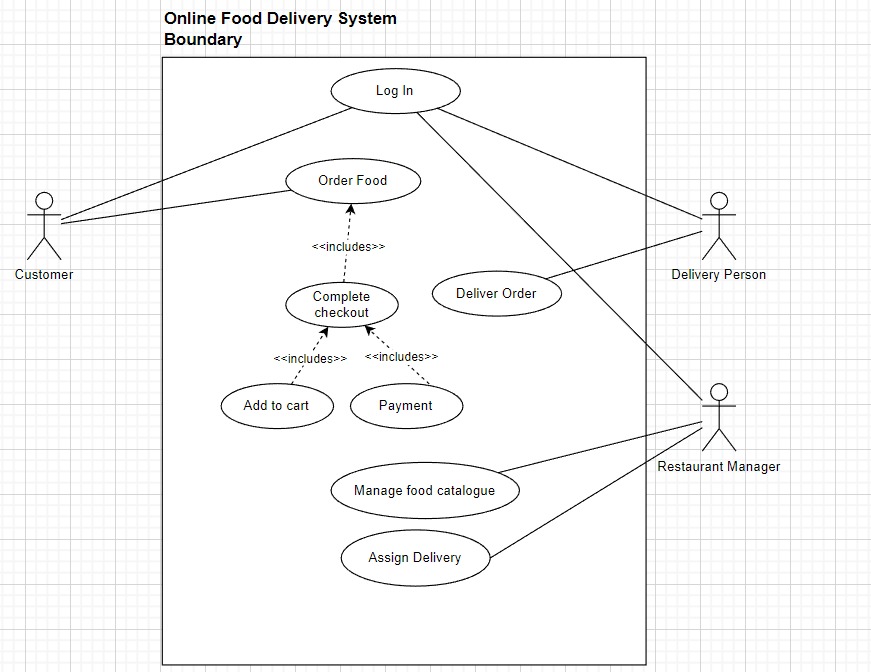
* 1. **Email Notification**

Customer receives welcome email on sign up.

* 1. **Role Based User Functionality**

Restaurant Manager, Customer and Delivery Person have different functionalities in the same application.

1. **Use Case Diagram**



1. **Hardware Requirements**

|  |  |
| --- | --- |
| **RAM** | 2 GB |
| **Hard disk** | 320 GB |
| **Processor** | Dual Core |

1. **Software Requirements**

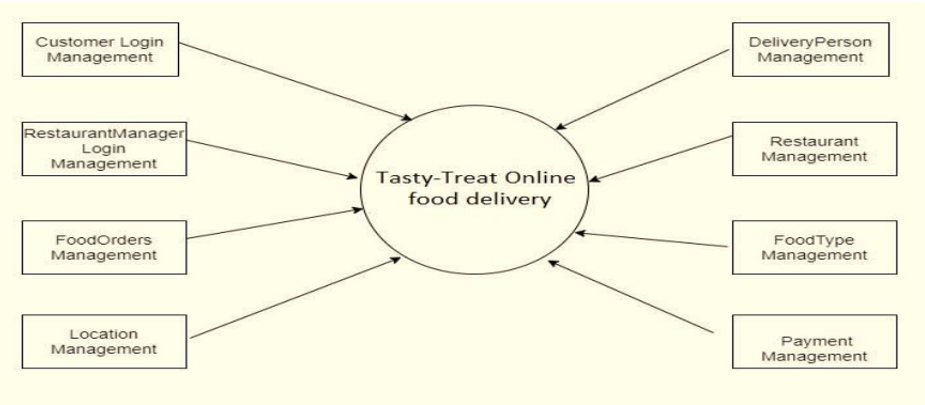
**Client side:**

|  |  |
| --- | --- |
| **Web Browser** | Google Chrome or any  compatible browser |
| **Operating System** | Windows or any equivalent OS |
| **RAM** | 4 GB and above |
| **Stable Internet Connection** |  |

**Server side:**

|  |  |
| --- | --- |
| **Processor** | Intel i3 8th gen & above |
| **HDD** | 500gb & above |
| **OS** | Windows 10 & above |
| **RAM** | 4 GB and above |
| **Database** | MySQL |
| **JDK 11 & above** |  |
| **Eclipse IDE** |  |
| **Visual Studio Code** |  |

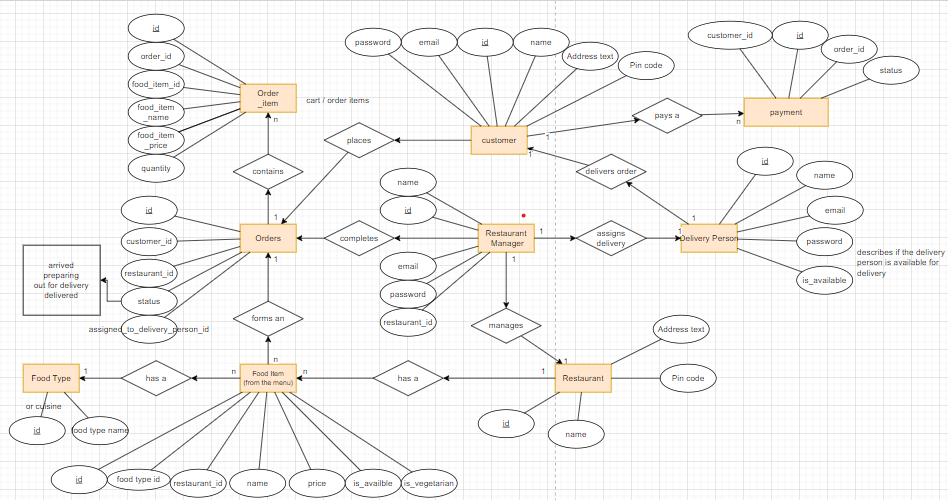
**6. System Design Specification:**

****

**ER DIAGRAM**

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design for the database designer, the utility of the ER model is:

* It maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
* It is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
* In addition, the model can be used as a design plan by the database developer to implement a data model in specific database management software.

****