**Software Requirement Specification (SRS) Template**

Title: FreshToHome

Document Version: 1.0

Date: 15 November 2023

Prepared by: Shilpa Chandrashekar

1. Introduction

* **Purpose**: The goal is to develop an online meat ordering system that will simplify life and provide minutes-long, personalized delivery to the doorstep.
* **Scope**: This meat store that operates online, maintains a wide inventory of fresh meat & cuts our catalogue has everything one could possibly need.
* **Background**: Select from a large selection of options in each category that have been carefully chosen to find the best quality products at the lowest prices.

2. Functional Requirements

* **Requirement 1**: Search for products.
  + - Provide input box to search on products and button to perform search.
    - When a user searches for a product and click enter/click on search button the products page with all the relevant products that match the search should appear.
* **Requirement 2**: Order Products.
  + - If the user wants to check for location proximity by providing their location.
    - The products should be listed on the home page according to categories and active promotions.
* **Requirement 3**: Modify order from cart.
  + - The user ought to have the ability to add items to their cart and edit it.
    - Items can be added to the cart by all users.
    - Cart and checkout should be accessible to all users.
    - A login button should be available when an unregistered user tries to checkout.
* **Requirement 4**: Payment Methods.
  + - The user can choose to pay with cash, UPI, or Sodexo upon delivery, or online with a debit or credit card.
* **Requirement 6**: Login Credentials.
  + - The user should be able to use their email address or phone number to sign up or login with otp / with password.

3. Non-Functional Requirements

* **Requirement 1**: Page should render <1ns.
* **Requirement 2**: UI/UX design ought to be visually appealing.

4. Use Cases

* **Use Case 1**: The user can search any product from any of the listed category.
* **Use Case 2:** The user should be able to order product.
* **Use Case 2**: The user should be able to modify cart.
* **Use Case 4**: The user can choose for convenient payment methods.
* **Use Case 5**: The user should be able to login/sign-up with proper credentials.

5. System Architecture

* **Architecture Overview**: 3 tier architecture. UI, API to receive the request.
* **Data Model**: Relational Model.

6. Constraints

* **Constraint 1**: Geo-Location.
* **Constraint 2**: User delivery address.

7. Assumptions and Dependencies

* **Dependency 1**: Payment gateway.
* **Assumption 1**: Null.