**Software Requirements Specification (SRS)**

**Project Name:** FreshVege  
 **Version:** 1.0  
 **Date:** June 19 – August 21  
 **Prepared By:** Ilma Hossain Mim

### **1. Introduction**

#### **1.1 Purpose**

The purpose of *FreshVege* is to provide a simple, efficient e-commerce platform for selling fresh produce such as vegetables, grains, and fruits. The application enables farmers to list their products, users to browse and order items, and administrators to manage inventory and users.

#### **1.2 Scope**

FreshVege will:

* Allow users to browse a catalog of fresh produce with prices.
* Provide cart and wishlist features.
* Enable farmers/admins to add new products.
* Allow customers to register, sign in, and view detailed product descriptions.
* Provide admins with access to customer information and reporting features.
* Operate without payment processing—transactions are handled offline.

#### **1.3 Stakeholders**

* **Farmers** – List and manage products.
* **Users/Customers** – Browse products, add to cart/wishlist, place orders.
* **Admin** – Manage products, view user details, monitor orders.

### **2. Overall Description**

#### **2.1 System Environment**

FreshVege will be a **web-based** platform built using HTML, CSS, JavaScript, and PHP with MySQL as the database. It will be accessible via standard web browsers on desktop and mobile devices.

#### **2.2 User Classes**

* **Guest Users** – Can view products but must register to order or create a wishlist.
* **Registered Customers** – Can place orders, manage their cart/wishlist.
* **Admins** – Can add/edit products, manage users, and generate reports.

#### **2.3 Operating Environment**

* **Frontend:** HTML, CSS, JavaScript
* **Backend:** PHP
* **Database:** MySQL
* **Platform:** Web browsers (Chrome, Firefox, Edge, Safari)
* **Hosting:** Shared web hosting

### **3. Functional Requirements**

1. **User Registration/Login**
   * Sign-up page to add new users to the database.
   * Login authentication with username/password.
2. **Product Management**
   * Admin can add, edit, and delete products from a dashboard.
   * Display product list with names, prices, and images.
   * View detailed product pages.
3. **Cart and Wishlist**
   * Customers can add products to cart or wishlist.
   * Cart stores selected products until checkout.
4. **User Management**
   * Admin can view user details from the database.
   * Admin can generate simple reports (e.g., product stock levels, user counts).
5. **Data Management**
   * Store user information in the database.
   * Retrieve and display product and user details.

### **4. Non-Functional Requirements**

* **Performance:** All pages should load within **500 ms** under normal conditions.
* **Security:** Basic authentication for login; input validation to prevent SQL injection.
* **Scalability:** The system should support at least 500 concurrent users.
* **Availability:** Target uptime 99%.
* **Usability:** Simple UI for non-technical users.
* **Maintainability:** Code should be modular to allow future updates.

### **5. Constraints**

* No online payment gateway integration.
* Must use **MySQL** for the database.
* Tech stack limited to **HTML, CSS, JavaScript, PHP**.
* Project timeline: **June 19 – August 21**.

### **6. Assumptions**

* Farmers and admins will use the same dashboard interface.
* All transactions are handled outside the platform.
* Product images and descriptions are provided by farmers.

### **7. Future Enhancements**

* Integration with a payment gateway.
* Mobile app version for Android/iOS.
* Advanced analytics and sales tracking.