GreenNest E-Commerce Platform Documentation

Team Details:

Redg Number	Name
22PA1A0512	B. Samhitha
22PA1A0530	CH. Eswar Adesh
22PA1A0554	G. Deepika
22PA1A0563	K. Kishore

1. Introduction

1.1 Purpose

This document defines the software requirements for the **GreenNest E-Commerce Platform**, a web application built for gardening enthusiasts to explore, buy, and manage gardening-related products online. It outlines the functional and non-functional requirements, system features, design constraints, and interfaces.

1.2 Scope

GreenNest is a full-stack e-commerce solution allowing users to:

- Register and log in securely
- Browse categorized gardening products
- Add products to a cart and checkout
- Track and manage their orders

Tech Stack: MongoDB, Express.js, React.js, Node.js (MERN)

2. Overall Description

2.1 Product Perspective

GreenNest uses:

- **Frontend**: React 18+
- **Backend**: Express.js on Node.js 18+
- **Database**: MongoDB (managed via MongoDB Compass)
  The frontend communicates with the backend via **RESTful APIs**.

2.2 Product Functions

- User Registration/Login with JWT
- Product Listings & Filtering
- Cart and Checkout Operations
- Admin Order/Product Management

2.3 Operating Environment

- **Client**: Chrome, Firefox, Edge, Safari
- **Server**: Node.js/Express
- **Database**: MongoDB Compass

3. System Features

3.1 User Authentication

- Secure registration with name, email, password
- JWT-based session authentication
- Role-based access control (user/admin)

3.2 Product Browsing

- Categories: Indoor Plants, Outdoor Plants, Pots, Tools, Seeds, Care Kits
- Features: Search, filter, sort, pagination

3.3 Shopping Cart

- Add/remove items
- Update quantities
- Store data using `localStorage`

3.4 Order Management

- Checkout with shipping info
- Order lifecycle: `Pending → Processing → Shipped → Delivered`
- Admin tools to view and update order status

4. External Interface Requirements

4.1 User Interfaces

- Responsive UI built with React.js
- Mobile-friendly layout

4.2 Hardware Interfaces

- Works on desktops, tablets, smartphones with standard browsers

4.3 Software Interfaces

- MongoDB (via Mongoose ODM)
- RESTful APIs with Express
- JWT for auth, bcrypt for password hashing

5. System Architecture

5.1 Component Diagram

```
React Client ──► Express API ──► MongoDB Compass
```

5.2 Modules

- **Backend**: `models/`, `routes/`, `middleware/`, `seed/`
- **Frontend**: `pages/`, `components/`, `context/`, `App.js`

6. Database Design

6.1 User Schema

```
{
  name: String,
  email: String,
  password: String, // hashed
  role: 'customer',
  address: {...},
  phone: String,
  timestamps: true
}
```

6.2 Product Schema

```
{
  name: String,
  description: String,
  price: Number,
  category: String,
  stock: Number,
  image: String,
  featured: Boolean,
  ratings: {...},
  timestamps: true
}
```

6.3 Order Schema

```
{
  user: ObjectId,
  items: [ { product, quantity, price } ],
  shippingInfo: {...},
  subtotal: Number,
  shipping: Number,
  total: Number,
  status: 'pending' | 'shipped' | 'delivered',
  paymentStatus: String,
  timestamps: true
}
```

7. Future Enhancements

- Integration with payment gateways (Stripe/PayPal)
- Email notifications for order events
- Product reviews and ratings
- Wishlist & favorites feature
- React Native-based mobile app

8. GitHub Repository: green-nest-ecommerce