**Project Design Phase**

**Proposed Solution Template**

|  |  |
| --- | --- |
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59438 |
| Project Name | ShopSmart: Your Digital Grocery Store Experience |
| Maximum Marks | 2 Marks |

**Proposed Solution Template:**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Description** |
| 1. | Problem Statement (Problem to be solved) | The proposed solution addresses the problem of cluttered, overwhelming online shopping experiences. Many users, especially young professionals, seek a streamlined and responsive way to browse and purchase products without unnecessary distractions. Shopsmart aims to simplify product browsing and cart management for a better user experience. |
| 2. | Idea / Solution description | Shopsmart is a full-stack web application that allows users to browse a curated list of products, view item details, and manage their cart seamlessly. The system consists of a React.js frontend and a Node.js backend. Users can view available stock, add items to the cart, and get a real-time summary of their selections. The backend handles API requests and data storage with a focus on simplicity and performance. The admin can manage product listings via the backend, and users interact with a clean, responsive UI. |
| 3. | Novelty / Uniqueness | Shopsmart stands out with its minimal and focused design, contrasting with the feature-heavy interfaces of major platforms. It prioritizes ease of use, speed, and clarity. It eliminates distractions to offer a clutter-free, responsive shopping experience for users who value simplicity. |
| 4. | Social Impact / Customer Satisfaction | Shopsmart improves digital accessibility for users who may find large marketplaces overwhelming. It empowers users to make quick, informed purchase decisions, enhancing customer satisfaction through speed, simplicity, and clarity. Small business owners can also use the platform to showcase products efficiently. |
| 5. | Business Model (Revenue Model) | The primary revenue model is product listing fees for vendors who want to showcase their products. Additional monetization avenues include featured placements and targeted ads for specific product categories or vendors. |
| 6. | Scalability of the Solution | The solution is built using a modular architecture with Node.js and MongoDB for backend scalability and React.js for frontend performance. It supports containerization, API scaling, and distributed deployment, making it ready for high user traffic and rapid growth. |