**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59438 |
| Project Name | ShopSmart: Your Digital Grocery Store Experience |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Example: ShopSmart: Your Digital Grocery Store Experience**

Guidelines:

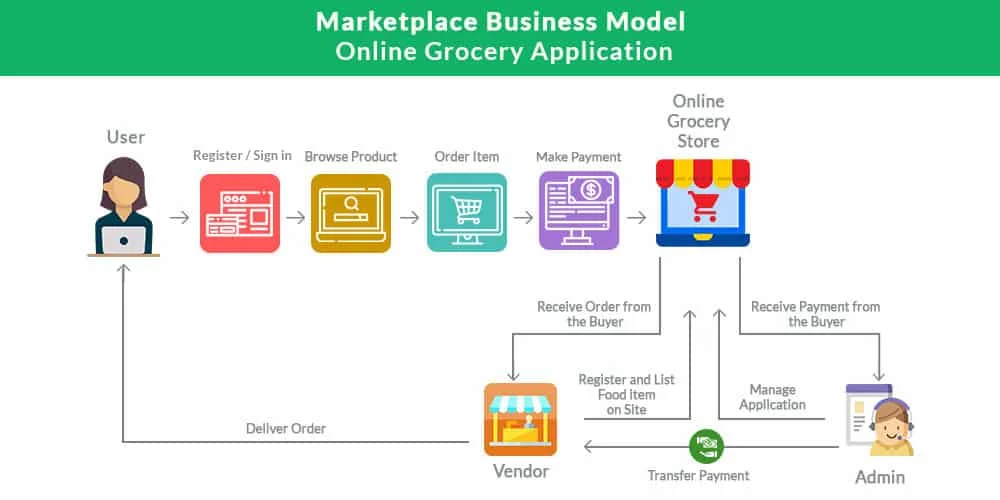
Include all the processes (As an application logic / Technology Block)

Provide infrastructural demarcation (Local / Cloud)

Indicate external interfaces (third party API’s etc.)

Indicate Data Storage components / services

Indicate interface to machine learning models (if applicable)



**Table 1: Components & Technologies**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| **1.** | **User Interface** | **How user interacts with application** | **HTML, CSS, JavaScript, React.js** |
| **2.** | **Application Logic-1** | **Core backend logic** | **Node.js (Express.js)** |
| **3.** | **Application Logic-2** | **Core Frontend logic** | **React.js** |
| **5.** | **Database** | **Data storage and structure** | **MongoDB (NoSQL)** |
| **6.** | **Cloud Database** | **Cloud DB Service** | **Planned: MongoDB Atlas** |
| **7.** | **File Storage** | **Static file storage** | **Local Filesystem (option: AWS S3)** |
| **8.** | **External API-1** | **External APIs used** | **Not applicable currently** |
| **9.** | **External API-2** | **—** | **Not applicable** |
| **10.** | **Machine Learning Model** | **Machine learning integration** | **Not used in this application** |
| **11.** | **Infrastructure (Server / Cloud)** | **Deployment platform** | **Local (Dev), Cloud-ready (Heroku, AWS, Render)** |

**Table-2:**

**Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Characteristics | Description | Technology |
| 1. | Open-Source Frameworks | List the open-source frameworks used | React.js, Express.js, Mongoose, bcryptjs, JWT, Bootstrap |
| 2. | Security Implementations | Security/authentication/access controls | JWT auth, bcrypt password hashing, HTTPS (prod), input validation |
| 3. | Scalable Architecture | Justify scalability | 3-tier (Frontend-Backend-DB), load balancer & MongoDB sharding support |
| 4. | Availability | Availability features | Deployable to cloud with auto-scaling + MongoDB replica sets |
| 5. | Performance | Optimizations for performance | Optimized REST APIs, efficient Mongoose queries, optional CDN |