**Project Design Phase**

**Solution Architecture**

|  |  |
| --- | --- |
| Date | 15 February 2025 |
| Team ID |  |
| Project Name |  |
| Maximum Marks | 4 Marks |

**Solution Architecture:**

**1. Introduction**

**ShopEZ follows a structured 3-Tier Architecture to ensure scalability, maintainability, security, and performance. The architecture separates the system into Presentation Layer, Application Layer, and Data Layer, enabling clear division of responsibilities and efficient system management.**

**2. Architecture Overview**

**ShopEZ is designed using the MERN Stack (MongoDB, Express.js, React.js, Node.js). The system supports both Customer and Admin workflows with secure communication between frontend, backend, and database layers.**

**3. Architecture Layers**

**3.1 Presentation Layer (Frontend)**

**Technologies Used:  
React.js, HTML, CSS, Bootstrap**

**Responsibilities:**

* **Provides User Interface for Customers and Admin**
* **Product browsing and filtering**
* **Cart management functionality**
* **Checkout and payment interface**
* **Admin dashboard for product and order management**
* **Sending API requests to backend services**

**This layer ensures responsive design and seamless user interaction.**

**3.2 Application Layer (Backend)**

**Technologies Used:  
Node.js, Express.js**

**Responsibilities:**

* **REST API development**
* **Authentication and Authorization using JWT**
* **Business logic implementation**
* **Order processing workflow**
* **Payment gateway integration (Razorpay/Stripe)**
* **Email notification handling**
* **Role-based access control (Admin/User)**

**The backend acts as the core processing unit of the system.**

**3.3 Data Layer (Database)**

**Technology Used:  
MongoDB (MongoDB Atlas – Cloud)**

**Collections:**

* **Users**
* **Products**
* **Cart**
* **Orders**
* **Categories**
* **Admin**

**The database securely stores all persistent data required for system operations.**

**4. Solution Architecture Flow (Data Flow)**

**4.1 Customer Flow**

**User (Browser)  
⬇  
React Frontend  
⬇ REST API Calls  
Node.js + Express Server  
⬇  
MongoDB Atlas  
⬇  
Response Returned to User**

**This flow ensures smooth interaction between user interface and database through secure backend APIs.**

**4.2 Admin Flow**

**Admin Login  
⬇  
Admin Dashboard (React)  
⬇  
Backend API  
⬇  
MongoDB (Update Products / Orders / Categories)**

**Admins manage products, orders, and categories through secure backend APIs.**

**4.3 Payment & Notification Flow**

**Customer → Checkout  
⬇  
Backend Payment Gateway API (Razorpay/Stripe)  
⬇  
Payment Confirmation  
⬇  
Order Stored in Database  
⬇  
Email Notification Sent**

**This ensures secure payment processing and real-time order confirmation.**

**5. Security Architecture**

**ShopEZ implements multiple security mechanisms:**

* **Password hashing using bcrypt**
* **JWT-based authentication**
* **Role-based access control (User/Admin)**
* **HTTPS secure communication**
* **Environment variables for storing sensitive credentials**
* **Secure API validation and error handling**

**These measures ensure data privacy, integrity, and system protection.**

**6. Conclusion**

**The ShopEZ 3-Tier Architecture ensures:**

* **Clear separation of concerns**
* **Scalability and cloud readiness**
* **High performance and maintainability**
* **Secure user authentication and transactions**
* **Easy future enhancements (AI recommendations, mobile app integration)**

**This architecture effectively bridges business requirements with technical implementation.**