**Project Design Phase**

**Solution Architecture**

|  |  |
| --- | --- |
| Date | 15 April 2025 |
| Team ID | SWTID1743695015 |
| Project Name | Shopez: E-Commerce Application |
| Maximum Marks | 4 Marks |

**Solution Architecture:**

This diagram represents the architecture of a scalable and modular **e-commerce platform**, detailing the interaction between various frontend clients, backend microservices, databases, and third-party services.

**1. Front-End Clients**

* **Web Browser** interacts with the **Web Application** built using React or Angular.
* **Mobile App** interacts with the **Mobile Application** developed for iOS and Android.

Both platforms connect to the backend through an **API Gateway**.

**2. API Gateway**

* **API Gateway (Kong / AWS API Gateway)** is responsible for routing requests from clients to the appropriate backend services. It also handles concerns like rate limiting, authentication, and monitoring.

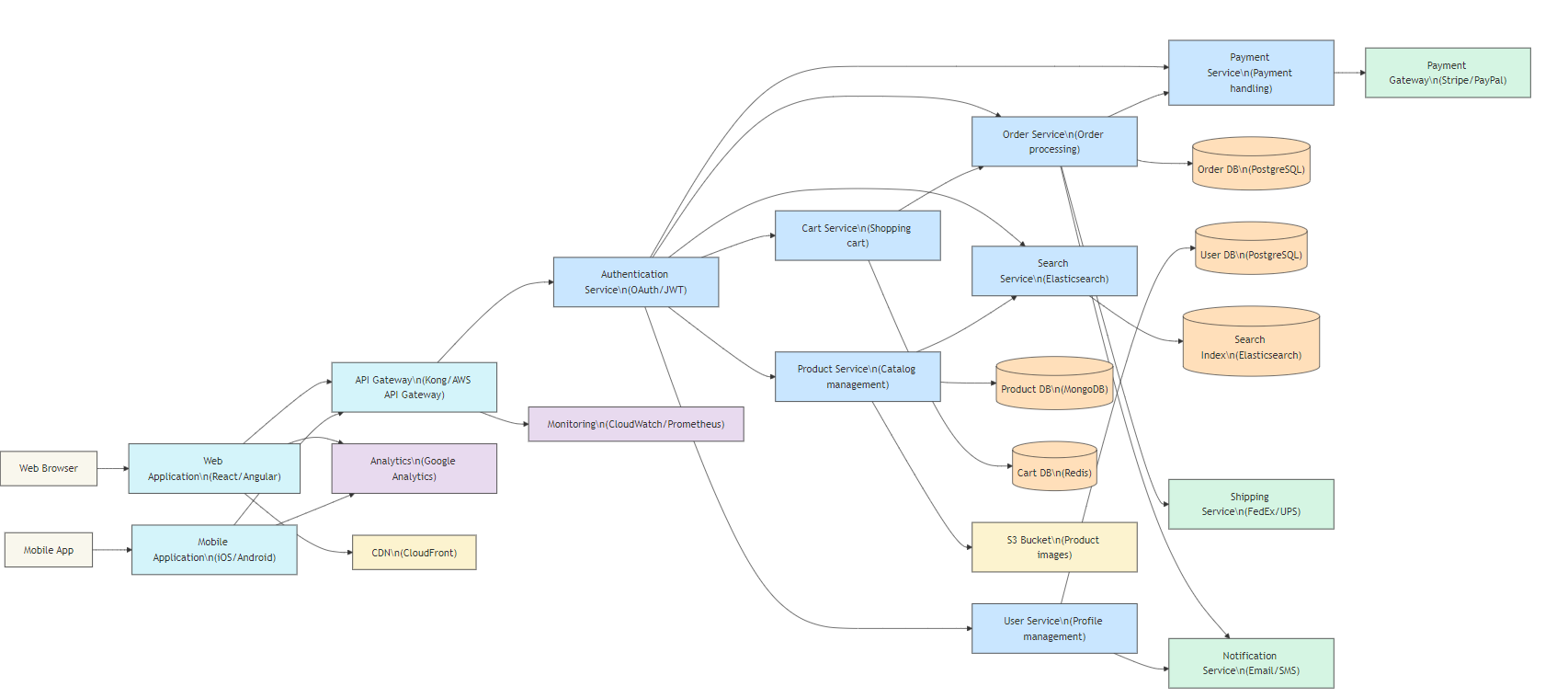
**3. Core Backend Services**

* **Authentication Service (OAuth / JWT)**  
  Manages user authentication and authorization. Issues secure tokens for session management.
* **Cart Service (Shopping Cart)**  
  Manages users’ shopping carts. Connects with the **Cart DB (Redis)** for fast, in-memory access to cart data.
* **Product Service (Catalog Management)**  
  Handles product information and images. Interacts with:
  + **Product DB (MongoDB)** to store product data
  + **S3 Bucket** for product image storage
* **Search Service (Elasticsearch)**  
  Enables product search functionality. Syncs with:
  + **Search Index (Elasticsearch)**
  + Pulls data from **Product Service**
* **Order Service (Order Processing)**  
  Processes customer orders. Communicates with:
  + **Order DB (PostgreSQL)**
  + **Notification Service** for sending order confirmations
* **Payment Service (Payment Handling)**  
  Manages payments and connects to external:
  + **Payment Gateway (Stripe / PayPal)**
* **User Service (Profile Management)**  
  Manages user profiles and account information. Uses:
  + **User DB (PostgreSQL)**
* **Shipping Service (FedEx / UPS)**  
  Handles shipping integration and tracks delivery of orders.
* **Notification Service (Email / SMS)**  
  Sends messages and alerts to users for order status, payment confirmations, etc.

**4. Supporting Components**

* **Monitoring (CloudWatch / Prometheus)**  
  Collects logs, metrics, and alerts for system performance and reliability.
* **Analytics (Google Analytics)**  
  Tracks user behavior and provides insights on site/app usage.
* **CDN (CloudFront)**  
  Distributes static content (like images, scripts, and stylesheets) to enhance front-end performance.

**Solution Architecture Diagram:**

****