

# ECOMMERCE PROJECT

**"AN INNOVATIVE, SCALABLE, AND SECURE ECOMMERCE PLATFORM"**

# TEAM MEMBERS



**S.m.Refat**  
**FRONTEND**



**Monowar Hossen Sourav**  
**BACKEND & DEVOPS**



**Mohammad Masum Jia**  
**FRONTEND**

# ABOUT US

## Who We Are:

- ❖ A team of developers focused on creating innovative and scalable eCommerce solutions.
- ❖ Driven by technology, user experience, and security.

## Why This Project?

- ❖ The rise of eCommerce as a major industry.
- ❖ Opportunity to address real-world challenges with modern technology.



# WHAT WE BELIEVE



## Vision

- ☐ To create a seamless, user-friendly, and scalable eCommerce platform that caters to both small businesses and large enterprises.



## Mission

- ☐ Enable businesses to reach a broader audience through a robust digital platform.
- ☐ Prioritize customer satisfaction with smooth functionality and personalized experiences.

# FEATURES

## Core Features:

- User Authentication & Authorization (JWT).
- Product search with advanced filtering.
- Wishlist and cart management.
- Order tracking and status updates.
- Role-based admin panel for businesses.
- Notifications via email (SMTP integration)

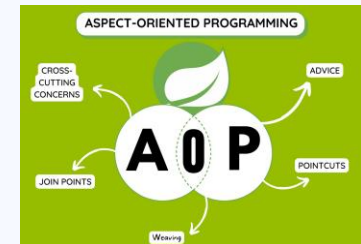
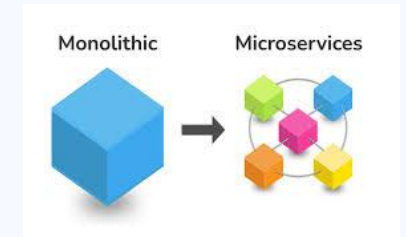
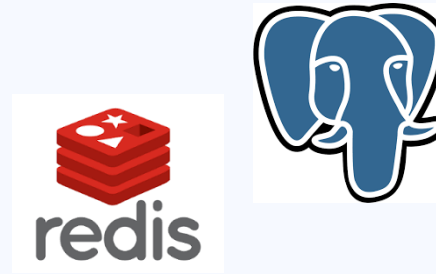
## Technical Features:

- High performance using Redis for caching.
- Real-time updates with Kafka message brokers.
- Scalable microservices for future growth.

# BACKEND TECH STACK

## Core Technologies :

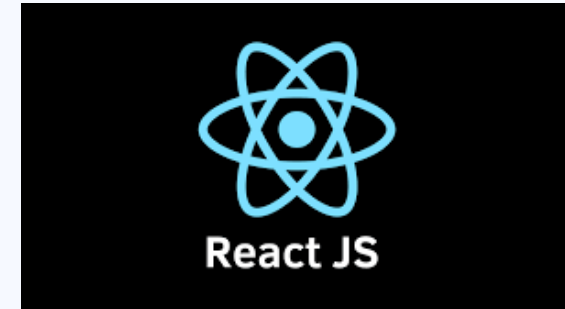
- ❑ **Spring Boot:** Rapid development and robust backend framework.
- ❑ **Microservices:** Scalability and modularity.
- ❑ **Redis Cache:** Enhanced performance with caching.
- ❑ **PostgreSQL/MySQL:** Reliable and robust database solutions.
- ❑ **JWT:** Secure Authentication and Authorization.
- ❑ **AOP:** for Authorization and Logging
- ❑ **Kafka:** Asynchronous communication using message brokers.
- ❑ **SMTP Setup:** Email notifications.



# FRONTEND TECH STACK

## React / Angular

- ❑ Component-based architecture.
- ❑ State management (e.g., Redux or Context API).
- ❑ Responsive and interactive UI design.



# DEVOPS OVERVIEW

- ❑ **Kubernetes:** Orchestrates containers for scalability and availability.
- ❑ **Jenkins:** CI/CD pipelines for automated builds and deployments.
- ❑ **Docker:** Containerization ensures consistency across environments.





# KEY FEATURES

- ❑ User Authentication & Authorization (JWT).
- ❑ Caching with Redis for improved performance.
- ❑ Asynchronous messaging with Kafka.
- ❑ SMTP for email services (e.g., order confirmation, password reset).
- ❑ Scalable microservices architecture.

# HIGH-LEVEL ARCHITECTURE

- ❑ Backend services (Spring Boot microservices)
- ❑ Frontend (React/Angular)
- ❑ Database
- ❑ Cache (Redis)
- ❑ Message Broker (Kafka)
- ❑ DevOps components (Kubernetes, Jenkins, Docker)

# SOURCE CODE MANAGEMENT

## GitHub:

- Version control and collaboration.
- Centralized repository for all codebases (frontend, backend, and DevOps scripts).
- Enables seamless teamwork with pull requests, code reviews, and issue tracking.
- CI/CD integration with Jenkins for automated builds and deployments.



**Thank You**