

# PROJECT GUIDELINES

## 1. Introduction

- **Purpose:**  
The objective of this project is to develop a custom e-commerce platform using the MERN stack (MongoDB, Express.js, React.js, and Node.js). This project aims to provide students with practical experience in full-stack development, focusing on modern web technologies.
  - **Background:**  
E-commerce websites are essential for businesses in the digital era. By creating a custom platform, students will understand how to integrate frontend, backend, and database components seamlessly.
  - **Scope:**  
This project covers user authentication, product catalog management, shopping cart functionality, order processing, and optional payment gateway integration.
- 

## 2. Problem Statement

- **Definition:**  
Small businesses often struggle to find affordable and customizable e-commerce solutions. This project addresses the need for a scalable and customizable e-commerce platform for such businesses.
  - **Importance:**  
Solving this problem enables businesses to have an online presence, improve customer reach, and streamline operations.
- 

## 3. Objectives

- Design a responsive user interface for browsing and managing products.
  - Implement secure user authentication (registration and login).
  - Build a dynamic product catalog with CRUD operations.
  - Integrate a shopping cart and order placement system.
  - Optionally, connect a payment gateway (e.g., Stripe, PayPal).
- 

## 4. Methodology

1. **Data Collection:**  
Gather requirements for user roles, product data, and expected functionalities.
2. **Tools/Technologies:**
  - **Frontend:** React.js, Material-UI or Bootstrap for styling.
  - **Backend:** Node.js, Express.js.

- **Database:** MongoDB (Mongoose for schema management).
- **Version Control:** GitHub/GitLab for source code management.
- **Deployment:** Not yet decided.

### 3. Experimentation/Implementation Process:

- Design the database schema.
  - Set up the backend API endpoints.
  - Create the React frontend and connect it with the API.
  - Test all functionalities end-to-end.
  - Deploy and validate the application in a production-like environment.
- 

## 5. Project Plan

- **Timeline:**
    - Week 1: Requirement gathering, project setup, and initial planning.
    - Week 2-3: Backend API development and database setup.
    - Week 4-5: Frontend design and integration with backend APIs.
    - Week 6: Testing and deployment.
  - **Milestones:**
    - Completion of database and API design.
    - Functional frontend with user authentication.
    - Integrated shopping cart and order placement.
    - Fully tested and deployed application.
- 

## 6. Expected Deliverables

- Fully functional e-commerce website codebase.
  - User and technical documentation.
  - A 10-15 minute presentation demonstrating the project features.
  - A report detailing the project implementation and learnings.
- 

## 7. Evaluation Criteria

- **Quality of literature survey:** 10%.
- **Design and implementation:** 30%.

- **Functionality and results:** Not yet tested.
  - **Report quality:** 10%.
  - **Presentation skills:** 25%.
- 

## 8. Submission Guidelines

- **Format:**
    - Code: Submit via GitHub/GitLab repository.
    - Report: PDF format.
    - Presentation: PowerPoint slides.
  - **Deadline:** By end of February 2025.
  - **Mode of Submission:** Submit via email or upload to an online portal.
- 

## 9. References

- MERN stack official documentation.
- Tutorials and guides on React, Node.js, Express, and MongoDB.
- Articles on best practices for e-commerce design