আন্তর্জাতিক ইসলামী বিশ্ববিদ্যালয় চট্টগ্রাম

الجامعة الإسلامية العالمية شيتاغونغ

International Islamic University Chittagong

Final Project Description

CSE-3532

Tools and Technologies for Internet Programming

Project Title

Smart E-Commerce & Service Management Platform

Project Description

The **Smart E-Commerce & Service Management Platform** is a responsive, full-stack web application that enables users to browse products or services, view details, place orders, submit reviews, and manage their purchases through a secure authentication system.

The system also provides a **role-based dashboard** where administrators can manage products or services, while users can track, update, or cancel their own orders. The application is engineered using **modern web technologies**, **RESTful backend architecture**, and **secure third-party authentication**, addressing real-world scalability, usability, and stakeholder requirements.

Students must design and implement this system following **professional software engineering practices**, making it suitable for evaluation under **Complex Engineering Problem (CEP) properties: WP1, WP2, and WP7**.

Mandatory Technology Stack

Frontend

React.js

Component-based architecture

CSS Framework (any one or combination):

Tailwind CSS

DaisyUI

Flowbite

Fully responsive UI (mobile, tablet, desktop)

Backend

Node.js

Express.js

RESTful API implementation

MVC (3-layer) architecture – *optional but encouraged*

Database (Anyone)

MongoDB Atlas

Mongoose ODM

Schema validation & relationships

Authentication & Authorization

Firebase Authentication

Email & Password Login / Registration

Google Login

GitHub Login

Role-based access control (Admin, User)

Functional Feature Requirements (Mandatory)

1. Navigation Bar

Logo

Menu links (Home, Products/Services, About, Contact, Dashboard)

Login / Logout buttons

Responsive hamburger menu

2. Sliding Banner (Hero Section)

Auto-sliding banner

Promotional or featured content

Call-to-action buttons

3. Products / Services Section

Grid or card-based layout

Product/service image, title, price

Short description

“View Details” button

4. Product / Service Details Page

Full description

Price

Ratings & reviews

Purchase / Order button

5. Purchasing Features

Order form

Quantity selection

User information auto-filled

Order validation

6. Order Confirmation

Confirmation page after successful order

Order summary

Order ID and status

7. Authentication (Email & Password)

User registration

User login

Password validation and error handling(optional)

8. Social Login

Login with **Google**

Login with **GitHub**

9. Dashboard (Role-Based)

Admin Dashboard

Insert product/service

Update product/service

Delete product/service

Manage all orders

Change order status

User Dashboard

View own orders

Edit own order

Delete/cancel own order

View order status

10. Customer Review Form

Authenticated users can submit reviews

Rating + text feedback

11. Review Preview with Sliding Banner

Auto-sliding customer reviews

Reviewer name, rating, comment

12. Extra Unique Mandatory Feature (Per Group)

Each group must implement **one unique feature**, such as:

AI-based product recommendation

Real-time order tracking

Inventory alert system

Wishlist & comparison feature

But highly encouraged to implement any AI tools such as AI Chatbot,AI Product Recommendation etc.

13. Contact Page

Contact form

Email / phone / address

Form validation

Message storage in database or email notification

14. About Us Page

Organization/project overview

Mission & vision

Team information

15. Footer

Quick links

Social media icons

Copyright

Responsive layout

Backend Engineering Requirements

All APIs must be **RESTful**

CRUD operations using Express

Proper HTTP status codes

Middleware for:

Authentication

Authorization

Error handling

Secure environment variable usage

Database Requirements

MongoDB Atlas cloud database

Multiple collections (Users, Products/Services, Orders, Reviews)

Responsiveness Requirement

Fully responsive on:

Mobile

Tablet

Desktop

Adaptive layouts using Tailwind / DaisyUI / Flowbite

Evaluation Using Complex Engineering Properties

WP1 – Depth of Engineering Knowledge

Students must demonstrate:

Full-stack integration (React + Express + MongoDB)

Firebase authentication & token validation

RESTful API design

Database schema modeling

Secure role-based system design

WP2 – Conflicting Technical Requirements

Students must manage:

Security vs usability (social login + RBAC)

Performance vs scalability

UI complexity vs responsiveness

Data consistency vs flexibility

Feature richness vs maintainability

WP7 – Multiple Stakeholders & Real-World Constraints

The project addresses:

Different user roles (Admin vs Customer)

Data privacy and ethical handling

Submission Requirements

1. GitHub repository with commits
2. Live deployment
3. **A Lab Report (Each Student)**
4. **A Final Project Report (Each Group)**