University of Engineering and Technology, New Campus, Lahore
Department of Computer Science

Complex Computing Problem

Course Code & Title: CSC-414 Enterprise Application Development

1. Course Learning Outcome:

CLO 3: Develop scalable desktop and web applications by integrating modern frameworks, web services, and CRUD operations.

Level of CLO: CLO-3 is mapped to the 6th level, i.e., Create of Bloom's taxonomy.

2. CCP Attributes:

- **A2 Depth of analysis required:** Has no obvious solution and requires conceptual thinking and innovative analysis to formulate suitable abstract models.
- **A3 Depth of knowledge required:** A solution requires the use of in-depth computing or domain knowledge and an analytical approach that is based on well-founded principles.
- **A5 Level of problem:** Is outside problems encompassed by standards and standard practice for professional computing
- **A8 Interdependence:** Is a high-level problem possibly including many component parts or subproblems.

3. Problem Statement:

Automating Attendance Tracking for Academic Institutions in Lahore

Problem Description:

In many schools across Lahore, attendance is still recorded manually on paper. This traditional method often leads to errors, misplaced records, inefficiency, and lack of transparency for both teachers and students. Administrators also face challenges in generating timely attendance reports and monitoring student participation accurately. To overcome these limitations, the institution requires the design and development of a secure, scalable, and user-friendly Attendance Management System**.** The system must streamline attendance tracking by allowing teachers to log in and mark daily attendance for each course, while enabling students to register in courses and view their attendance records in real-time.

Your tasks are:

- Authentication & Authorization
- Attendance Module
- Course Registration & Enrollment
- Reporting & Analytics
- Deployment

4. Instructions:

- This is an individual project so each student will work independently.
- Your project will be graded according to the provided rubrics.
- Implement all CRUD operations using Entity Framework (EF Core). You are not allowed to use simple SQL queries.
- Deadline to submit: **December 31, 2025 (11:59 pm)**

5. Deliverables:

1. **Authentication & Authorization**

- Secure login/logout module with session management.
- Role-based access control (separate dashboards for Teacher and Student).

2. **Attendance Module**

- Fully functional teacher attendance marking interface.
- Student attendance dashboard displaying daily and overall status.

3. **Course Registration & Enrollment**

- Course registration page for students.
- Enrollment module linking students to courses.
- Validation system ensuring teachers can only mark attendance for assigned courses.

4. **Reporting & Analytics**

- Automated attendance percentage reports (per student, per course).
- Report filtering features (course, date range, student).
- Exportable reports (PDF/Excel) for administrative use (optional but good to include).

5. **Deployment**

- Deployed web-based Attendance Management System (AMS) on a server or cloud platform (Azure, AWS, or IIS).
- Accessible URL or server IP for demo/testing.

6. Alignment with Attributes of Complex Computing Problems:

Attribute	Alignment / Justification
Depth of analysis required	• These is no obvious solution. • Requires conceptual thinking to design attendance module, avoid duplicate entries, and create flexible reporting mechanisms.

Depth of knowledge required	• Needs in-depth computing knowledge: ASP.NET Core (MVC, DI, layered design), EF Core (ORM for CRUD and relational DB), Authentication/Authorization principles, Deployment (IIS, Azure, AWS). • Must apply software engineering best practices and database principles.
Level of Problem	• Goes beyond standard classroom exercises. • Involves enterprise-level concerns like scalability, deployment, data integrity, exception handling, and user roles.
Interdependence	• System is composed of many interdependent submodules: Authentication & authorization, Course registration, Attendance marking, Reporting/analytics, Deployment.

Complex Computing Problem Assessment Rubrics

Criteria & CCP Attributes	Excellent (5)	Good (4)	Satisfactory (3)	Needs Improvement (2)	Poor (1)
Authentication & Authorization (A3, A5)	Secure login with session management fully functional; RBAC correctly implemented; strong password & security practices applied.	Mostly secure login with RBAC; minor issues in session handling.	Basic login and role separation; lacks some security features.	Login works but RBAC poorly implemented; vulnerable security.	Login/authorization not implemented or non-functional.
Attendance Module (A2, A3, A8)	Teacher can mark attendance (Present/Absent/Late); duplicate prevention; student dashboard accurate in real time.	Functional attendance marking; dashboard updates correctly but with minor delays/issues.	Attendance marking works but dashboard lacks clarity or partial updates.	Limited attendance features; inaccurate records.	Attendance module missing or unusable.
Course Registration &	Students can register in multiple courses; validation ensures	Registration mostly functional;	Registration implemented but lacks strong	Registration partially functional;	Registration not functional or absent.

Criteria & CCP Attributes	Excellent (5)	Good (4)	Satisfactory (3)	Needs Improvement (2)	Poor (1)
Enrolment (A3, A8)	only assigned teachers can mark attendance; error handling robust.	teacher validation works with small gaps.	validation or error handling.	teacher access not properly restricted.	
Reporting & Analytics (A2)	Accurate reports (attendance % per student/course); filtering by course, student, date range; reports exportable.	Reports accurate but limited filtering; exports partially supported.	Reports generated but with calculation/formatting issues.	Basic summary only; lacks filtering or detailed reports.	No reporting implemented.
Deployment Deliverables (A3, A5)	Fully deployed system on cloud/server (Azure/AWS/IIS); accessible via URL; deployment documentation complete and reproducible.	Deployed system functional; minor documentation gaps.	Deployed locally only; basic instructions provided.	Deployment incomplete; hard to access or poorly documented.	No deployment or inaccessible system.