UOW MALAYSIA

PART OF THE UNIVERSITY
OF WOLLONGONG AUSTRALIA
GLOBAL NETWORK

Bachelor of Computer Science (Hons)
Bachelor of Software Engineering (Hons)

Internet & Web Development

XBMC3014N

Prepared by Wong Choon Yee
Semester June 2025

SCHOOL OF COMPUTING & CREATIVE MEDIA

ASSIGNMENT: Group Project

Course Title : Bachelor of Computer Science (Hons)
Bachelor of Software Engineering (Hons)
Course Code : XBMC3014N
Course Lecturer : Wong Choon Yee
: cywong@uow.edu.my

LEARNING OUTCOMES

On completion of this module, the student should be able to:

1. Design a web site or web application and identify the different components required to execute it . (PLO5,C4)
2. Develop a web site or web application and identify the different components required to execute it . (PLO3,P4)
3. Work in groups to present web application developed to show it meets specified requirements (PLO4,A3)

BRIEF

Scenario

Managing student performance is a critical part of academic leadership, especially in higher education institutions. Manual tracking methods are time-consuming, error-prone, and lack real-time analytics. As part of a university's digital transformation initiative, your development team is commissioned to design and build a web-based Student Performance Monitoring System, focused specifically on the Programme Leader (PL) role.

Your Task

As a team of professional web developers in an education technology consultancy, your job is to design, develop, and deploy a responsive web application that allows the Programme Leader to:

- Monitor individual student performance (CGPA, credits earned, progression risk)
- View academic planning indicators such as:
    - Modules This Semester
    - Modules Next Semester
    - Exempted without Credit Transfer
    - Credit Transferred
    - Remaining Subjects
    - Semesters Left

The system should support data analysis, storytelling features, sorting, and filtering to assist in academic decision-making.

Note: Sample data is provided for illustration purposes.

SYSTEM REQUIREMENTS

1. Student & Course Management
    - Add, edit, and delete student records.
    - Create and manage courses and modules.
    - Assign students to specific courses or programme tracks.
    - Manage module status (e.g., completed, in-progress, exempted).
    - Bulk import students via CSV.

2. Database & Data Persistence
    - Use MySQL or another secure relational database.
    - Implement foreign key constraints and referential integrity.
    - Normalize tables to reduce redundancy.
    - Support data backup and recovery.
    - Detect and handle data duplication.

3. Student's profile dashboard.
    - Display personal and academic information (name, ID, programme).
    - Show all completed subjects (subject code, subject name, grade, credit, and CGPA) organized by semester.
    - List exempted and credit-transferred modules.
    - Highlight subjects planned for the upcoming semester.
    - Provide data visualizations.

4. PL Dashboard
    - View overall student performance indicators (e.g., Dean's List and Probation).
        - Dean's List: CGPA above 3.75 with a minimum of 12 credits.
        - Probation List: CGPA below 2.00 for two consecutive semesters.
    - View module breakdown: current semester, next semester, and remaining subjects.
    - Generate summary reports (e.g., unregistered subjects, at-risk students).
    - Filtering and sorting to improve decision making.
    - View semester-wise progression overviews.

5. Interactive Dashboards
    - Combine filter criteria by student, semester, course, grade, etc.
    - Display charts such as CGPA trends, pass/fail ratios, and subject averages.
    - Support drill-down into individual student or subject performance.
    - Ensure the interface is intuitive and easy to use.
    - Export dashboard views to CSV or PDF.

6. Storytelling with Data
    - Automatically generate narrative summaries from raw data.
    - Highlight positive trends (e.g., consistent CGPA improvement).
    - Identify and describe warnings (e.g., repeated failures).
    - Present student or module-level highlights in plain language.
    - Display all insights in a clearly formatted storytelling panel.

7. Innovative Features
    - Propose and implement three innovative ideas using modern web technologies.

ASSESSMENT CRITERIA

1. If you do not attend the proposal or Presentation, the maximum mark you can achieve for this assignment is **40%**.
2. Your submission should include the following items:
    - Proposal Presentation (20%)
    - Group Project (60%)
    - Project Presentation (10%)
    - Progress Report (10%)

Please refer to the marking rubric for further details

RESOURCES

Credit all sources you use, including photos, graphics, logos, widgets, and text. If you created your own graphics or took photos, credit yourself. Do not submit prototype from other sources as your own, as this constitutes plagiarism and will result in a grade of 0 for the assignment. The module team is aware of internet sources and recognizes the sharing and reuse of code in the coding community. The goal of this assignment is to demonstrate your skills, not your ability to copy others' code.

HAND-IN REQUIREMENTS

Hand in all components through Microsoft Teams by the due date listed in this assignment brief. Late submissions will receive full marks, but will be penalized 10% per day after the deadline. NOTE – You can always hand in the work early.

Team leader:

1. Proposal Presentation (20%)
2. Group Project (60%)
3. Project Presentation (10%)
4. Progress Report (10%)

Name all files using your student ID and name, e.g. PeerAssessment0123456MaryJane.doc or IWD-Group1.zip.

REFERENCES

1. Do not include references in your word count.
2. References must be in APA format. Recommended to use Mendeley for auto generation of the citation.
3. Please use ACM or IEEE portal for research journals.
4. Font Times New Roman or Arial, 11 points font, 1.15-line spacing, and justified format.

DUE DATE:

Component	Weightage (%)	Due Date
Proposal	20	14-Jul-25
Final Project	60	18-Aug-25
Project Presentation	10	TBA
Progress Report - Analysis, Design & Task Allocation	2	30-Jun-25
Progress Report - Implementation & Task Allocation	3	21-Jul-25
Progress Report - Critical Review	5	30-Aug-25

ASSESSMENT RUBRIC: Proposal Presentation (20%)

Proposal Presentation (20%)						
Criteria	**Very Poor (0-2)**	**Poor (3-4)**	**Average (5-6)**	**Good (7-8)**	**Excellent (9-10)**	**Marks**
Problem Statement & Objectives	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Introduction: a. Clear articulation of the purpose and goals of the website. b. Identification of the target audience and their needs. c. Identification of the Programme Leader and their needs. 2. Problem Statement and Justification: a. Clearly defined problem statement related to website. b. Justification for why a website is needed.	
Functional specification	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Comparison tables of 3 similar apps. 2. Comprehensive set of comparison criteria for 3 apps. 3. Highlighting the uniqueness of each app. 4. Clearly outlining the main functional specifications. 5. Clearly outlining the Innovative Features.	
Wireframe	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Clarity and Simplicity 2. Navigation and User Flow 3. Content Representation & Information Hierarchy 4. Functionality and Interaction 5. Consistency with Design Principles	

Database Design	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. ERD Diagram – Shows all key entities and relationships clearly. 2. Core Tables – Includes Students, Modules, Results, Semesters, etc. 3. Foreign Keys – Uses proper relationships between tables. 4. Normalization – Avoids redundancy, follows at least 3NF. 5. Schema Explanation – Brief description of each table and fields.	

ASSESSMENT RUBRIC: Group Project (60%)

Group Project (50%)						
Criteria	**Very Poor (0-2)**	**Poor (3-4)**	**Average (5-6)**	**Good (7-8)**	**Excellent (9-10)**	**Marks**
Functional Modules	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Student record management (add, update, delete, view) 2. Module tracking for current, next, and remaining subjects 3. Exemptions and credit transfer implementation 4. Semester-wise CGPA and credit summaries 5. Export functions (PDF/CSV) and filtering features	
Student's profile dashboard.	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Displays student details and academic overview (CGPA, credits) 2. Lists subjects by semester with grades and status 3. Shows exemptions, transferred credits, and completed modules 4. Highlights subjects for upcoming semester 5. intuitive and high usability	
Programme Leader (PL) Dashboard	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Displays Dean's List and Probation List based on defined rules 2. Overview of student progression and semesters left 3. Visual summaries of CGPA, credit load, and filters 4. Highlights subject registration and planning status 5. Includes filtering by semester, student, or academic status	
Storytelling & Data Analysis	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Generates charts and graphs (e.g., CGPA trends) 2. Detects academic patterns and risks automatically 3. Provides narrative summaries of performance in natural language 4. Alerts for GPA drops, repeated failures, or other anomalies 5. Allows breakdown by semester, subject, or student	
Code Quality	Fulfilled 0 - 1 of the	Fulfilled 2 of the	Fulfilled 3 of the	Fulfilled 4 of the	1. Modular, reusable code structure 2. Well-commented and easy-to-read source code	

	excellent elements.	excellent elements.	excellent elements.	excellent elements.	3. Organized file/folder structure and naming conventions 4. Use of version control (e.g., Git) during development 5. Follows best practices for maintainability and clarity	
Non-Functional Design & User Experience	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Responsive and fast-loading design suitable for both desktop and mobile users 2. Clean, consistent, and intuitive interface with easy navigation 3. Effective use of CSS frameworks (e.g., Bootstrap/Tailwind) for layout and styling 4. Includes proper error handling, validation, confirmation alerts, and feedback messages 5. Clear visual hierarchy with readable fonts, appropriate colors, spacing, and tooltips	
Database Design	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Well-structured ERD or schema diagram 2. Normalized tables (at least up to 3NF) 3. Proper use of primary and foreign keys 4. Supports core features like CGPA calculation and filtering 5. Maintains data integrity and relational consistency	

ASSESSMENT RUBRIC: Presentation (10%)

Presentation (10%)						
Criteria	**Very Poor (0-2)**	**Poor (3-4)**	**Average (5-6)**	**Good (7-8)**	**Excellent (9-10)**	**Marks**
A short video of your website.	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Introduction: Team introduction and project purpose 2. Feature Demonstration: Show all key features, including innovative functions. 3. Visual & Audio Quality: Clear visuals and audible narration with minimal background noise. 4. Organization: Logical flow with smooth transitions and clear explanations. 5. Professionalism: Confident, engaging presentation with a strong conclusion.	
Engagement & Delivery	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Audience Engagement: Does the team actively engage the audience through questions or interaction? 2. Confidence & Fluency: Is the presentation delivered confidently and fluently, with minimal pauses or hesitations? 3. Body Language & Eye Contact: Do the presenters use appropriate body language and maintain eye contact? 4. Team Collaboration: Do all members contribute evenly to the presentation, showing good collaboration? 5. Handling of Distractions: How well do the presenters handle interruptions or distractions during the presentation?	

Visual Aids & Demonstration	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Introduction & Conclusion: Is there a clear introduction outlining the project's objectives and purpose? 2. Live Demonstration: Is the dashboard demonstrated live, showing key features and functionality? 3. Visual Quality: Are the visual aids and demo clear, easy to follow, and visually appealing? aids (slides, graphs, or diagrams) 4. Flow of the Demonstration: Does the demo proceed logically, with smooth transitions between features? 5. Relevance of Visual Aids: Do the visual aids and demo align with and enhance the points being made?	
Quality of coding	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Code Structure: Well-organized and consistently formatted. 2. Readability: Clear code with meaningful names and comments. 3. Efficiency: Optimized code with no unnecessary repetition. 4. Modularity: Reusable components and functions. 5. Error Handling: Proper error checks and minimal bugs.	

ASSESSMENT RUBRIC: Progress Report (10%)

Progress Report (10%)						
Criteria	**Very Poor (0-2)**	**Poor (3-4)**	**Average (5-6)**	**Good (7-8)**	**Excellent (9-10)**	**Marks**
Progress report 1: Analysis, Design & Task Allocation (2%)	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Comprehensive Progress Update. 2. Quality of Deliverables: 3. Task Allocation and Timeline. 4. Communication and Collaboration. 5. Adaptability and Problem Resolution.	/2
Progress report 2: Implementation & Task Allocation (3%)	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Comprehensive Progress Update. 2. Quality of Deliverables: 3. Task Allocation and Timeline. 4. Communication and Collaboration. 5. Adaptability and Problem Resolution.	/3
Progress report 3: Critical Review (5%)	Fulfilled 0 - 1 of the excellent elements.	Fulfilled 2 of the excellent elements.	Fulfilled 3 of the excellent elements.	Fulfilled 4 of the excellent elements.	1. Comprehensive Progress Update. 2. Quality of Deliverables: 3. Task Allocation and Timeline. 4. Communication and Collaboration. 5. Adaptability and Problem Resolution.	/5