Module: **Web Application Development**

Assignment No. = **Group Assignment/Project**

Total Marks: **40**

This is a group assignment where you are asked to work in a group of 7 or 8 members and build a web application collectively. Groups has been assigned on the Loop and details of each group will be added on the loop. Even though it is a group project, the grading of the project will be done on the overall project submission (50% marks) which will be equally given to all "active" team members and individual role marks (50% marks) which will be individually graded based on quality of tasks assigned to the individual team member. Please note that all group members should strictly adhere to DCU Academic Integrity policy and students are not allowed to discuss or share their group work with other groups.

**Requirements:**

In this assignment, you are required to

- Build a complete web application using the technologies learned during this module.
- Your application for the group assignment is **Smart DCU Data Dashboard** (details are added below)
- You have been already assessed for each of the individual underlying technology, your focus in this assignment/project should be on the overall web application performing a specific task rather than using all technologies within your web application.
- You are allowed to use any IDE and Frameworks for development; however, you have to stick to the core techniques covered in this module e.g. you are **not** allowed to use PHP instead of Java Spring boot or using IBM Web sphere instead of Tomcat is not permitted.

**Minimum Functionalities:**

There must be at least following functionalities in your web application.

- Minimum 5 or more static web pages connected through hyperlinks with reasonable contents in each page, with menu bar and navigation.
- Have some client level javascript functionalities such as events handling, forms validation and dynamic contents creation at the client side.
- Minimum 5 or more dynamic pages/functionalities through server-side programming displaying contents based on the context or user input.

- Have a back-end database running and connected through server-side programming where data can be dynamically queried or updated.
- Have a login functionality showing multiple roles for different kinds of authenticated users.
- Deployment of application on EC2 cloud (link and login details will be provided) and/or a packaged war file submission with scripts for database creation and installation instructions (read me file).
- A comprehensive project documentation, including introduction to the application domain, requirements gathering, technology selection decisions, code management decisions, roles assignments and responsibilities, minutes of group meetings, and installation instructions including readme file.
- Project demonstration using screencast with audio explanation or at least screenshots of different pages/functionalities with text explanation should be submitted.


**Assessment Purpose:**

The main purpose of this assignment is to assess your ability to

- Understand the Web Application Development Life-Cycle.
- Understand the requirements for a web application, your potential audience and provide better end-user experience.
- Provide a right balance between web page design, navigation and back-end functionalities.
- Design Front-end of your application using HTML and CSS.
- Use Java Script to provide basic level browser based interaction for validation, visualisation, event handling, dynamic menu bars and animations etc.
- Design back-end logic for server-side programming.
- Design and create a database tier for your web application.
- Store, query and update data web browser into persistent data storage.
- Create and maintain user sessions with your web application and provide log in functionality.
- Demonstrate your ability to work in a team environment and contribute to build large web applications.
- Quality assurance and testing your web application.
- Deploy and maintain your web application.


**Web Application (Smart DCU Data Dashboard):**

For this group assignment, all groups are asked to build a web application for Smart DCU Data Dashboard. Data dashboards are designed to summarise and visualise different but related type of data for a given domain, topic or infrastructure. A data dashboard for a campus can provide information related to campus occupancy, car parking availability, noise level, air quality and other similar information which can be used for decision making. Imagine a scenario where university campus is equipped with sensors-based infrastructure capturing a

variety of information. You are required to build a web application containing data summary and visualisation over different data sources.

**Data Sources:** A few examples of sensors data can be.

1. Air Quality Monitoring Across the Campus (Indoor or outdoor sensors for air quality)
2. Room Noise Level (noise levels at the different rooms or lecture halls)
3. Room Occupancy (Number of people in the room at a given time stamp)
4. Campus Occupancy (Number of vehicles and pedestrians entering and exiting the main entrance of the campus)
5. Parking Sensors (providing sensors)

These are sample data sources; you do not need to use only these types of datasets. You are free to choose any other type of data or pick a few from the above listed datasets. Please also note that students won't have access to any real dataset, hence you have full freedom to improvise any dataset and its structure before storing it in the database.

**Data Visualisation/Summarisation/Dashboard:** You are required to build a data dashboard for DCU campus which can provide historical data access through visualisation. For example, you can showcase a graph indicating the list of rooms and their occupancy level at different timestamps during the day, week or month. Similarly, you can use noise levels data to show noisy or quite areas on the campus. A few examples for visualisation can be taken from the JavaScript Graphics Section on the W3Schools Website accessible at: https://www.w3schools.com/js/js_graphics.asp The most important java script libraries for graphs and visualisation are Plotly.js, Chart.js, D3.js and Google Chart.

**User Authentication and User Access Roles**: Like your individual assignment, you are also expected to implement authentication mechanism for your group application where some information can be readily available for any users without login and some information can only be available for authenticated users. The authenticated users can be a viewer (can view information, graphs or datasets) and/or admin (can add new or remove existing information sources).

**List of Potential Roles for Group Project:**

Below is the list of potential roles which members of the group can take on during assignment preparation and submission. These are only tentative roles; you are free to create new roles as well. There is also no restriction to have assigned members for each role or on the number of group members taking the same role, e.g., if you believe frontend development or server-side programming requires more efforts for your application multiple members can take on the same role. You can take your individual role depending on your confidence on one of the technologies or skills. A sample list of roles is as below.

- Project management, business requirements analysis and/or Web application structure design.

- Front-end development HTML and CSS
- Client-side interactive features design (e.g. Java Script)
- Server-side development
- Database design and connectivity
- Testing and quality assurance
- Cloud deployment, maintenance, and scalability
- Code repository management, and installation
- Project documentation including minutes of the meetings, and the final report (each group member should contribute to preparing project documentation explaining their role and approach for the implementation of their assigned tasks.)
- Project demonstration (screencast recording or documentation with screenshots)

**Submission Instructions:**

1. Please arrange an initial meeting with your group members and make decisions regarding the application structure, roles, and responsibilities. You should submit a short document containing information on the assigned roles of each member by the end of week 8 (March 8th, 2024). This can be submitted by one member of the group on the behalf of whole team, and it should be submitted in the Group Assignment – Overall Group Marks section of the assignment on module loop page.

2. The following files must be submitted on the loop on or before the submission deadline.

    a. Minutes of the meeting arranged at least once in a week (Kick off meeting and one meeting per week during week 8, 9, 10, 11, 12)

    b. Project source code and if available link to the code repository

    c. Packaged War file ready for deployment

    d. Script files for database creation and data population

    e. A Readme file containing any additional installation instructions for installation.

    f. Project documentation

    g. A screencast with audio for project demonstration and/or screenshots of web pages with explanation of different navigation steps.

You can submit all files as one zip file or multiple zip files depending on the application size. One person can submit the project for the whole group. It must be submitted in the Group Assignment – Overall Group Marks section of the assignment.

3. Each student must submit the following document in the Group Assignment – Individual Marks section of the assignment.

a. A maximum of one-page description of your role and tasks which you have fulfilled in the group assignment. You can take this text from the project documentation, or you can only include pointers to group submission documentation indicating which part of the project was done by you to highlight your individual contributions. Please keep it to the point and don't exceed it beyond the one-page limit.