Assignment 2: Develop a Java Web Application

This is a **group-based** assignment where you will be working in a group of maximum **THREE (3)** students. You may discuss general points of view about the assignment with the other groups, but you are not permitted to work with them in completing this assignment.

1. **Grouping**
   Upon the **release of this assignment**, each student is required to register themselves into groups **within five (5) working days** through this module GitHub classroom assignment, https://classroom.github.com/a/Ebf78v48

2. **Assignment Solution**
   Upon submission, each group must submit their source code to their respective group GitHub repository through this module GitHub classroom assignment, https://classroom.github.com/a/Ebf78v48

3. **Video Demonstration**
   Each group must also submit video demonstration to explain the solution they made for the assignment.
   - It is required for you to hand in the video demonstration **[Maximum file size limit: 1GB]** through the Microsoft Form link: https://forms.office.com/r/TmVpyKDXEe

Weightage
25%

Due date
Saturday, 27th April 2023, 11:59PM

Late Submission
Penalties of 10% deduction of marks for assignment received for each working day after the due date. After one week of submission from due date consider as Fail.

Module Lecturer
Mohammad Jailani bin Haji Abdul Rahman
jailani.rahman@pb.edu.bn

Resources
Student own research, lecture and tutorial covered.

Equipment/Software
The source code should be done using Java Programming Language but you are free to use which IDEs that supports Java Development.

Scenario:
For this assignment, your group are required to develop a Java web application. The application should manage the data in a database management system.

The themes for this assignment are as follows:

1. Books catalogue
2. Contacts listing
3. Expenses tracking
4. Fitness routines
5. Jobs listing
6. Medical records
7. Movies review
8. Music listing
9. Recipes listing
10. Sport tournaments

No **two (2)-**module groups should choose the same topic. You are not allowed to use the practical session example as your topic.

Requirements

1. The implementation of the application should use the Model View Controller (MVC) design pattern.
2. Students are allowed to use any type of database management systems.
3. The application should be able to connect to the database implemented and do Create, Read, Update and Delete (CRUD) operations.
4. All interaction with the application should be through the Web. No console interactions are allowed.
5. The application will be a single server web application.
6. There should be at least 5 pages that each can be navigated through.
7. At least 3 out of the 5 pages display dynamic data from the database or from previous page.
8. Each page should contain at least 5 useful web components where it is organised accordingly. (Hint: Useful web components typically components that the user or the application can interact with)

Grading Criteria

Marks will be assessed based on the criteria given below:

Criteria	Percentage
Implementation of Web Application (35%)	
Implemented all components in single server web application. • All CRUD operations are handled by a single server web application. • The same server web application to handle all Uniform Resource Identifier (URI) requests. • Host all pages in the same server web application.	10%
Implemented at least 5 pages (HTML pages). • The design of all the pages is consistent. • At least 3 out of 5 pages are dynamic pages. (Marks are allocated based on the complexity of how the data are passed to the pages and how it is represented in the page) • It is possible to navigate through all pages through URI requests in the same server web application. • Each page contains at least 5 useful web components organised accordingly.	25%
Integrate Database in Web Application (35%)	
Integrate database in web application. (Marks are allocated based on how complex the database is structured, no marks given if no database structure is provided [Typically a Plain Old Java Object, POJO, classes in Hibernate])	10%
Implementation of database CRUD in web application. • Add data to database • Retrieve data from database • Update data from database • Remove data from database (Marks are allocated based on how complex the data manipulation and validation need to be done to maintain the data integrity and consistency)	25%
Code Structure (10%)	
Consistent implementations of MVC design pattern. • Separate Model components. • Separate View components. • Separate Controller components.	5%
Basic code structure • Consistent naming scheme (for classes, variables, methods). • Consistent code structured and easy to read. • The codes are properly commented.	5%

Group Collaboration in GitHub (10%)	
• The respective group's GitHub repository are consistently committed meaningful implementation to their source code. • Each member committed at least five (5) meaningful implementations to their respective group GitHub repository's source code.	10%
Video Demonstration (10%)	
• Demonstrated and explain how to navigate through the pages. • Demonstrated and explain how to interact with the pages. • Demonstrated within 5 to 10 minutes with each member explains their respective implementations	10%
TOTAL PERCENTAGE	**100%**

Assignment Rules

1. Any problems with your own group should be settled internally first. If it cannot be resolved then you may involve the module lecturer. Module lecturer has the rights to not entertain if the complaint is highlighted within five (5) working days before the assignment's deadline.

2. You may consult with the module lecturer if you are not clear with the assignment. But the module lecturer has the rights not to entertain any questions except for questions in regards to submission on the day of the assignment's deadline.

3. Any deliverables your group hand-in are considered as the group's own work unless it was stated otherwise (using proper referencing). No marks will be given on the work if no reasonable effort being done on top of the referenced work.

4. Any deliverables that are under suspicion of **Plagiarism**, **Ghost Writing**, **Collusion** and **Purloining**, the student(s) will be asked for further explanation to prove their understanding and originality of their work.

5. Failure to prove the originality or understanding of their work will result in **FAILURE (0 marks)** of the assignment.

6. Students are encouraged to use Git and GitHub private repository to host their assignment which can be used as their evidence for their effort and work.

7. Students are to make sure the submitted softcopy are not corrupted. Corrupted submission will be penalized accordingly.

8. Penalties of 10% deduction of marks for assignment received for each working day after the due date. Marks will be capped to 50% if the marks are less than 50% after deduction.

9. Rules may be updated from time to time. Any updates will be posted on the LMS.

- END OF FILE -