ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC COOPERATION (OIC)
Department of Computer Science and Engineering (CSE)

SEMESTER FINAL EXAMINATION	SUMMER SEMESTER, 2022-2023
DURATION: 3 HOURS	FULL MARKS: 150

CSE 4635: Web Architecture

Programmable calculators are not allowed. Do not write anything on the question paper.
Answer all **6 (six)** questions. Figures in the right margin indicate full marks of questions with
corresponding COs and POs in parentheses.

The following scenario will be used across multiple questions in this paper.

Scenario 1: ABC University wants to develop a comprehensive web application to streamline its campus operations and enhance the educational experience. The application will feature three main portals: Student, Faculty, and Administration.

Student Portal: Students can enroll in courses, access their course materials, submit assignments, view their grades, and receive notifications about university events. They can also apply for scholarships and track their application status.

Faculty Portal: Faculty members can manage their course schedules, upload lecture materials, grade assignments, publish results, and communicate with students. They can also keep records of their research and apply for grants.

Administration Portal: The university administration can oversee admissions, manage student records, handle faculty payroll, and maintain the university's infrastructure. They can also generate reports on university performance metrics.

1. a) For the scenario mentioned in Scenario 1, assume you have a JavaBean called FacultyBean, which has the following properties: FacultyID, Name, Adress, Position, DeptID, and Salary. All with necessary getter and setter methods. Also, assume that you have obtained an ArrayList object FacultyList of 10 FacultyBean objects from your database. Now, create a FacultyRecords.jsp page (with code) to dynamically load the information (FacultyID, Name, Adress, Position, DeptID) about all the faculties with a salary greater than $10,000. Use JSP standard action tags to access the JavaBean objects and JSTL tags to show their information inside the JSP page.   **10** (CO3) (PO3)

   b) Draw and explain the lifecycle of a servlet. When do the lifecycle methods get called?   **10** (CO2) (PO2)

   c) Explain ServletContext and ServletConfig parameters and attributes.   **5** (CO1) (PO1)

2. a) Discuss the steps involved in creating an RMI application.   **5** (CO1) (PO1)

   b) Suppose you have to implement an RMI application for restaurant table reservations. Suppose a restaurant has a reservation system that allows customers to check table availability for a specific date and time. Customers can use a client application to send a request to the central reservation server via RMI. The server then checks its reservation database to see if a table is available for the requested date and time. If a table is available, the server confirms the reservation and assigns a table number to the customer. Otherwise, it notifies the customer that no tables are available for the requested time slot. Write a Java RMI application for the scenario.   **20** (CO3) (PO3)

3. a) For the application mentioned in Scenario 1, suppose ABC University has a static list of ArrayList objects that store information about the courses offered by each department. The name of the static list is departmentCourseList. Information related to each course is as follows: (courseId, courseName, creditHours, deptId, semester, and courseType). Using JAX-RS, create a RESTful endpoint (with code) that takes a department ID as a query parameter and returns the list of courses for that department as an array list of JSON objects.

b)

```html
<form id="registrationForm" action="#" method="post">
    <div>
        <input type="text" id="name" name="name"/>
        <span class="error" id="nameError"></span>
    </div>
    <div>
        <input type="email" id="email" name="email"/>
        <span class="error" id="emailError"></span>
    </div>
    <div>
        <input type="password" id="password" name="password"/>
        <span class="error" id="passwordError"></span>
    </div>
    <div>
        <input type="password" id="confirmPass" name="confirmPass"/>
        <span class="error" id="confirmPassError"></span>
    </div>
    <div>
        <button type="submit">Register</button>
    </div>
</form>
```

Code Snippet 1: Code snippet for question Question 3.b

Write a javascript function, formValidate(), that will be invoked whenever a user clicks on the Register button of the aforementioned HTML form in Code Snippet 1. The function should prevent the default form submission behavior. Form validation should be performed based on the following:

- Password must be at least 8 characters long.
- Password must contain a capital letter and a special character.
- The name must not contain any special characters (?, $, #, @)
- The two passwords should match.

If any of the above requirements are not met, a corresponding error should be displayed using the corresponding <span> in the form.

4. Consider the EJB local interface below and answer the following questions

```java
import java.util.List;
import javax.ejb.Local;

@Local
public interface AdmissionCandidateInterface {

    void addCandidate(String name);

    List getCandidates();
}
```

a) What type of EJB will you choose to implement the functions in the Local interface and why? Write the corresponding EJB code.

<div align="right">10
(CO2)
(PO2)</div>

b) Write a servlet including the dependency injection of the EJB Local interface. The servlet should call the business method of the local interface for the input taken through a JSP page.

<div align="right">10
(CO3)
(PO3)</div>

c) Write the JSP page to take the input from the user for adding and displaying the list of candidates.

<div align="right">5
(CO3)
(PO3)</div>

5. a) What is Object Relational Mapping (ORM)?

<div align="right">5
(CO1)
(PO2)</div>

b) Assume you are currently working with the following two entities for the database of the application mentioned in scenario 1.

<div align="right">10
(CO5)
(PO3)</div>

Student	Department
Student_ID	Department_ID
Name	Name
Admission_date	Department_code
Email	HoD
Address	Address

Figure 1: Entities for Question Question 5.b

Student_ID and Department_ID are the primary keys. The are generated using the 'IDEN-TITY'. Assume the relationship is unidirectional, and create the EJB 3.0 entity classes for these two tables. Your code should demonstrate the proper relationship between them.

c) Write the corresponding DAOs for the Student Entity beans you created in question 5 (b). The DAO should support the CRUD operations on the Entity beans.

<div align="right">10
(CO3)
(PO3)</div>

6. a) Design and draw a detailed web architectural model for the application mentioned in scenario 1, showing relationships between different components of the application mentioned in scenario 1. The chosen design patterns and technologies should be indicated.

<div align="right">10
(CO3)
(PO2)</div>

b) Write short notes on the following:

<div align="right">5 × 3
(CO1)
(PO1)</div>

1. Session Bean Lifecycle Callback Methods.
2. Façade Design Pattern.
3. Differentiate between REST and SOAP.