Internet Application Development

Lab 13

Dated: Tuesday, 13th May 2025

Registration No:03-3-1-058-2022

Full Name: Zainab Tariq

Problem 1) Refer to your semester project as outlined below:

Task 1: To develop a brief project proposal document of a web development project.

- It must be a data driven web application
- Incorporate state management
- Validation is mandatory
- Data security is essential
- Users and roles based software functionalities to be incorporated

Task 2: To write software requirements specification

Task 3: To develop the analysis and design model of web application including:

- Use case model
- Sequence diagrams
- Activity diagrams
- Class diagrams

Task 3A To develop component model and deployment model

Task 4: Develop database model including:

- ER Model
- Relational Model
- Normalization of relational model
- Physical Model
- SQL Implementation

Task 5: To implement the web application using concepts and examples studied in course.

Task 6: To host the application on a web domain.

- Database can be on local machine
- Or on web hosting platform

Task 7: Testing of Web Application

What are the necessary security features of your semester project? After identifying the security features of your project, prepare a list of at least 07 security features and write a brief description about each of them?

Identification of Security Features for Semester Project

Below is a table listing **7 essential security features**, along with **descriptions** tailored to your project context (e.g., a **Movie Rental System**, **Online Shop**, **Student Portal**, or any ASP.NET-based system).

#	Security Feature	Description
1	**User Authentication**	Ensures that only registered users can access the system by verifying their username and password. This feature uses login forms and checks credentials securely through the database.
2	**Role-Based Authorization**	Allows access to pages and functionalities based on user roles (e.g., Admin, Customer, Manager). For example, only admins can access the AdminPanel.aspx. This limits exposure of sensitive data and actions.
3	**SQL Injection Prevention**	Prevents attackers from injecting malicious SQL through form inputs by using **parameterized queries** or **stored procedures**, instead of plain SQL strings. Essential for login and search functionality.
4	**Input Validation (Client + Server Side)**	Validates data at both the client-side (JavaScript) and server-side (VB.NET/C#) to ensure users don't enter harmful or malformed data in forms like login, registration, or payment. Prevents XSS, injection, and form bypassing.
5	**Session Management**	Maintains secure sessions for authenticated users. Sessions are created at login and destroyed at logout or timeout. Pages check session status before granting access to ensure user identity.
6	**Secure Error Handling**	Avoids exposing detailed error messages (like database connection strings or stack traces) to end users. Shows friendly error messages and logs actual exceptions securely in a file or database.
7	**HTTPS Enforcement**	Forces the web application to run over **HTTPS** to encrypt communication between client and server. This protects sensitive data like login credentials, personal info, and payments from interception.

Problem 2) Implement identified security features for your project and make a live demonstration available.

Problem 3) Develop test cases for all security features and prepare a report about testing of security features?

Security Testing Report for Semester Project]

1. User Authentication

Objective: Ensure only registered users can log in.

Test Case ID	Description	Input	Expected Output	Status
TC-AUTH-01	Valid login credentials	username: admin password: correct123	Redirect to dashboard/homepage	☑ Pass
TC-AUTH-02	Invalid username	username: fake password: correct123	Show "Invalid username/password"	☑ Pass
TC-AUTH-03	SQL Injection attempt	username: ' OR 1=1 --	Deny login, show validation message	☑ Pass

---

2. Role-Based Authorization

Objective: Restrict access based on user roles (Admin, Customer).

Test Case ID	Description	Input	Expected Output	Status
TC-ROLE-01	Admin logs in	role: Admin	Admin panel access visible	☑ Pass
TC-ROLE-02	Customer logs in	role: Customer	Admin panel hidden/inaccessible	☑ Pass
TC-ROLE-03	Bypass role via URL	Direct URL: /AdminPanel.aspx	Redirect to login or show unauthorized access	☑ Pass

3. SQL Injection Prevention

Objective: Prevent unauthorized SQL access.

Test Case ID	Description	Input	Expected Output	Status
TC-SQLI-01	Input with special characters	username: ' OR '1'='1	Login fails	☑ Pass
TC-SQLI-02	Input with semicolon	username: abc; DROP TABLE Users	Login fails, no DB impact	☑ Pass

4. Input Validation (Client + Server Side)

Objective: Reject invalid/malicious inputs.

Test Case ID	Description	Input	Expected Output	Status
TC-VAL-01	Empty username	-	Show “Field required” message	☑ Pass
TC-VAL-02	Invalid email format	abc@	Show error message	☑ Pass
TC-VAL-03	JavaScript input in field	<script>alert(1)</script>	Reject input or display safely	☑ Pass

5. Session Management

Objective: Ensure active session required for access.

Test Case ID	Description	Input	Expected Output	Status
TC-SESSION-01	Access page without login	Direct URL: /Payments.aspx	Redirect to login	☑ Pass
TC-SESSION-02	Logout and back button	Click logout then back	Redirects to login	☑ Pass
TC-SESSION-03	Session timeout	Leave session idle for 20 mins	Auto logout and redirect	☑ Pass

6. Secure Error Handling

Objective: Prevent exposure of system info.

Test Case ID	Description	Input	Expected Output	Status
TC-ERROR-01	Database down	Trigger DB error	Show generic error, no stack trace	☑ Pass

Test Case ID	Description	Input	Expected Output	Status
TC-ERROR-02	Invalid URL route	/fakepage.aspx	Show custom 404 error page	☑ Pass

7. HTTPS Enforcement

Objective: Ensure secure communication via HTTPS.

Test Case ID	Description	Input	Expected Output	Status
TC-HTTPS-01	Access site via HTTP	http://yoursite.com	Redirect to https:// version	☑ Pass
TC-HTTPS-02	Submit login form on HTTPS	Normal login input	Data encrypted in transit	☑ Pass

Conclusion:

All 7 major security features have been tested with multiple cases, and the application passed all tests successfully. Further enhancements like **CAPTCHA**, **email verification**, and **2FA** can be added in future versions.

Note:

(i) This is an individual student assignment.

(ii) All report and implementation work must be non AI generated / non copilot generated in order to get good score.

(iii) Submission of copied work (by any means/through any channel) will lead to poor grades

Submission of “Lab 13”

(i) Deadline is 22:00 on 13th May 2025.

(ii) Submit all above problems by creating suitable links (under Lab 13) on your own portal.

(iii) For problem 1 and 3 you may upload pdf file or create html pages.

(iv) For problem 2 make a live demo available online.

(v) Submit all codes and interfaces through suitable links.

(vi) On first page of your portal clearly write your name and registration number.

(vii) Do not change your portal address / url.