ourdiu.com | Shared by Rafsan Bin Habib

Daffodil International University

Faculty of Science & Information Technology

Department of Computing and Information System

Final Examination, Spring-2026

Course Code: CIS324, Course Title: Web Engineering

Level: 3 Term: 3

Exam Duration: 2 Hours **Marks: 40**

Answer ALL Questions

[The figures in the right margin indicate the full marks and corresponding course outcomes. All portions of each question must be answered sequentially.]

1.	The Department of Computing and Information System (CIS) is organizing the annual "AI Fest." As the lead developer, you are tasked with building a secure registration page. The frontend HTML form must collect a participant's Name, Event Date, Student ID, Academic Email, Project Code, and Mobile Number. This data must be securely submitted and saved into the *aifest_students* table within the *diu_db* database.			CO1
	a)	**Explain** how you would securely capture these six fields using an HTML form. Subsequently, **describe** the exact PHP backend steps required to establish a secure database connection and safely insert this data.	[7]	
	b)	Since direct user input poses security risks, **demonstrate** how you would use PHP Regular Expressions for strict data validation by **explaining** the pattern for each of the following rules: i. *Name*: Alphabetic characters and spaces only. ii. *Event Date*: YYYY-MM-DD format. iii. *Student ID*: Exactly 9 numeric digits. iv. *Academic Email*: Standard prefix rules, strictly ending with @diu.edu.bd. v. *Project Code*: The exact string "CIS" immediately followed by 3 digits. vi. *Mobile*: Exactly 11 digits, strictly beginning with '0'.	[3]	
2.	A newly established IT company has deployed a PHP-based API for an online food ordering system. The API connects to a MySQL database and utilizes HTTP GET requests to fetch menus and HTTP POST requests to submit new orders. However, a critical issue has emerged: users are clicking the "Place Order" button multiple times, or network delays are causing the identical request to be transmitted twice. This results in duplicate orders being saved in the database, disrupting operations.			CO2
	a)	**Apply** your understanding of web APIs to explain the benefits of using an API in this system and describe its working principle in handling GET and POST requests.	[5]	
	b)	**Apply** an appropriate design strategy to prevent duplicate order submissions. Illustrate your solution with a clear diagram and explain how it ensures that identical POST requests are safely detected and rejected.	[5]	

ourdiu.com | Shared by Rafsan Bin Habib

3.	A university is in the process of developing a secure web-based student portal that requires user authentication to access personalized academic resources. The system incorporates a "Remember Me" functionality to enable persistent login across browser sessions, as well as user interface customization options such as Light Mode and Dark Mode. The university places a strong emphasis on maintaining both robust security and a seamless user experience, while mitigating risks including unauthorized access and session hijacking.		CO3
	a)	**Analyze** how state management mechanisms interact to implement this "Remember Me" functionality across multiple sessions. Illustrate your explanation with a clear diagram.	[5]
	b)	**Examine** why relying *exclusively* on cookies for user authentication compromises the portal's security. Detail the specific vulnerabilities associated with this approach in your analysis.	[5]
4.	An e-commerce company recently launched an online retail platform but bypassed a structured Web Development Life Cycle (WDLC) to meet a tight deadline. Consequently, the system is difficult to maintain because the PHP database logic is heavily mixed with the HTML/CSS presentation layers. The company now plans to build a mobile application that fetches existing user data in JSON format, but developers estimate they must rewrite the backend entirely due to the poor code structure.		CO3
	a)	**Evaluate** the negative impacts of skipping a structured Web Development Life Cycle (WDLC) in this scenario, and **justify** how applying it could have improved the overall development process of the website.	[5]
	b)	**Assess** the structural flaws caused by mixing database logic directly with HTML presentation. **Defend** how adopting the Model-View-Controller (MVC) architectural pattern would resolve these issues and facilitate the development of the new mobile app.	[5]