



Daffodil International University

Department of Software Engineering
Faculty of Science & Information Technology
Final Examination, Summer 2025

Course Code: SE321; Course Title: Software Engineering Web Application

Sections: A,B,C,D,E,F,G,H,I & Teachers: PAS,DB,MI,MMM,AG,AAM,AMR

Time: 2:00 Hrs

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. | a) | Demonstrate various types of event listeners by developing a JavaScript program, providing appropriate examples, and explaining the functionality of each listener along with how it responds to user interactions. | [Marks-5] | CLO-2 <i>Level-3</i> |
| 2. | a) | <pre><!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <title>Image Click Event</title> </head> <body> <button class="btn">First Button</button> <p>Click Count: 0</p> <script> // JavaScript code will go here </script> </body> </html></pre> <p>Construct a JavaScript function that attaches a click event listener to the image with the ID myImage. When the image is clicked:</p> <ol style="list-style-type: none">Change the image source from "image1.jpg" to "image2.jpg"Increment the number shown in the #count span by 1 each time the image is clicked. | [Marks-5] | CLO-3 <i>Level-6</i> |
| | b) | Given two arrays of numbers, write a PHP program to find the smallest number from the first array and the largest number from the second array , then calculate and return the average of these two numbers . | [Marks-5] | |
| | | <pre><?php function averageOfExtremes(\$arr1, \$arr2) { // Solution code will go here } // Example Usage \$arr1 = [4, 1, 2, 1, 2]; \$arr2 = [2, 2, 3, 3, 5]; echo averageOfExtremes(\$arr1, \$arr2); // Output: 3 ?></pre> <p>Prepare the solution code by using PHP.</p> | | |

| | | | | |
|----|----|--|------------|------------------|
| 3. | a) | <p>Design a PHP OOP program that models a number addition system using an interface and a trait.</p> <ol style="list-style-type: none"> Create an interface Input with a method setNumbers(\$a, \$b) to set two numbers. Create a trait Logger with a method log(\$message) to display messages. Create a class Addition that implements the Input interface and uses the Logger trait. The class should calculate the sum of the two numbers. Instantiate the class, set two numbers, perform the addition, and print the result along with logs. | [Marks-10] | CLO-4 Level-6 |
| | b) | <p>In a Laravel application, the following route is defined in web.php:</p> <pre>Route::get('/product', [ProductController::class, 'index'])->name('product.index');</pre> <p>The ProductController is located in app/Http/Controllers/ and contains:</p> <pre>public function index() { \$products = Product::all(); return view('products.index', compact('products')); }</pre> <p>The Product model is located in app/Models/ and the corresponding view file is resources/views/products/index.blade.php.</p> <p>Trace the role and responsibilities of each component in the context of how a browser request to /product is handled.</p> | [Marks-8] | |
| | c) | <p>In a Student Management System, the administrator should be able to manage student records.</p> <p>Develop a PHP-MySQL application to add new students.</p> <p>Requirements:</p> <ul style="list-style-type: none"> Establish a connection to the MySQL database. Create a PHP form to enter a student's name, roll number, and department. Insert the student record into the database upon submission. Display a confirmation message after successful insertion. | [Marks-7] | |

Note: You may add or delete rows as per your requirements. Mention CLO's in the right side of the table as per your course outline.