Pokhara University
Faculty of Management Studies

Course Code: CMP 273   Full marks: 100
Course title: **Internet Technology II**   Pass marks: 45
Nature of the course: Theory + Practical   Credit Hrs: 3
Year: Second   Total periods: 48 hours
Level: Bachelor   Program: BCSIT
Semester: III

1. Course Description

This course introduces students to web technologies, focusing on server-side scripting using PHP (Hypertext Preprocessor) for dynamic web content generation. It builds upon the foundational knowledge of web development acquired in Internet Technology I. Covering a wide range of topics, including PHP syntax, control structures, loops, arrays, functions, form handling, file management, database integration, and advanced PHP concepts like object-oriented programming and MVC architecture, students gain practical experience in developing robust and scalable web applications. Through hands-on exercises, projects, and practical assignments, students learn to set up development environments, connect to databases, handle user input, manage sessions, and implement authentication and authorization mechanisms. By the end of the course, students are proficient in PHP programming, capable of designing and developing sophisticated web applications.

2. General Objectives

- To understand the fundamentals of server-side scripting and the role of PHP in web development.
- To develop a strong understanding of core concepts of PHP to build dynamic web applications effectively.
- To gain practical experience integrating PHP with HTML, CSS, and JavaScript to create interactive user interfaces.
- To learn essential techniques for handling user input, processing form submissions, and implementing data validation.
- To gain hands-on experience in using PHP to interact with databases, manipulate files, and handle HTTP requests.
- To learn to implement authentication and authorization mechanisms to secure Web applications.
- To explore advanced PHP concepts such as object-oriented programming, database integration, and error handling.
- To familiarize oneself with popular PHP frameworks and libraries to accelerate development and improve code maintainability.

3. Method of Instructions

General Instructional Technique: Lecture, Discussion, Reading, Question Answer.
Specific Instructional Technique: Practical works, Project Based Learning, Self-Directed Learning, Industry Insights and Case Study.

4. Content in Detail with Specific Objectives

Specific Objectives	Contents [4 Hrs.]

- Gain a foundational understanding of server-side scripting and its relevance to web development. - Learn about the significance of PHP programming language in building dynamic web applications. - Develop the ability to set up a PHP development environment. - Familiarize with the basic syntax of PHP, including statements, variables, and comments. - Understand the functionality and usage of operators for performing different operations in PHP.	**Unit 1: Introduction**
	1.1 Understanding server-side scripting and PHP Programming
	1.2 Installing and setting up PHP development environment (XAMPP, WAMP, or alternatives)
	1.3 Basic syntax and data types
	1.4 Variables and Constants
	1.5 Operators
	   1.5.1 Arithmetic Operators
	   1.5.2 Assignment Operators
	   1.5.3 Logical Operators

Specific Objectives	**Contents [5 Hrs.]**

- Understand the concept of conditional statements and their role in controlling the flow of program execution. - Explore different conditional logic statements. - Learn about the ternary operator as a concise alternative for simple conditional expressions. - Develop the ability to use loops for repetitive tasks. - Control the loop execution flow with continue and break statements.	**Unit 2: Control Structures and Loop**
	2.1 Conditional Statements
	   2.1.1 If, Else, Elseif
	   2.1.2 Ternary Operator
	   2.1.3 Switch Statement
	2.2 Using Loop for Repetitive Tasks
	   2.2.1 While, Do...While, For
	   2.2.2 Continue and Break

Specific Objectives	**Contents [7 Hrs.]**

- Understand the concept of arrays and their significance in storing and organizing data. - Learn how to iterate through arrays using various iteration techniques. - Understand the concept of functions and their role in promoting code reusability by encapsulating reusable logic. - Dealing with strings, numbers, date & time, and arrays with built in function. - Learn about passing arguments to functions and returning values from functions. - Understand variable scoping and how they affect the visibility and accessibility of variables within functions.	**Unit 3: Array and Function**
	3.1 Working with Array
	   3.1.1 Indexed Array
	   3.1.2 Associative Array
	   3.1.3 Array Iteration
	   3.1.4 Multi-dimensional Array
	3.2 PHP's Built in Function
	   3.2.1 String Function
	   3.2.2 Math Function
	   3.2.3 Date and Time Function
	   3.2.4 Array Function
	3.3 User Defined Function
	   3.3.1 Passing arguments and Return
	   3.3.2 Variable Scoping

Specific Objectives	**Contents [6 Hrs.]**
Gain proficiency in processing HTML forms to collect user input from web pages.	**Unit 4: Form Handling and Data Validation**

• Understand the HTTP request methods and their usage in form submissions. • Validating and sanitizing user input to prevent security vulnerabilities and ensure data integrity. • Validating file types, sizes, and other attributes to ensure the integrity of uploaded files and prevent potential security risks. • Understand the process of sending emails programmatically.	4.1 Processing HTML forms with PHP 4.2 Working with HTTP Request (GET, POST, SERVER) 4.3 Preserving User Input 4.4 Validating and Sanitizing User's Input 4.5 Dealing with checkbox, radio button and list. 4.6 File Upload 4.7 Send Email
• **Specific Objectives**	**Contents [6 Hrs.]**
• Learn about the importance of file handling in web development, including reading file content and storing user-generated content, logging application data etc. • Understand the various file reading and writing modes and their respective use cases. • Know the differences between file inclusion methods (include and require) and their respective use cases. • Gain practical knowledge in managing sessions and cookies for maintaining user state and tracking user interactions. • Acquire skills in implementing session-based authentication and authorization mechanisms to control access to web applications and resources. • Learn to efficiently manage and gracefully handle unexpected errors or exceptional conditions in the application	**Unit 5: File Handling, Sessions, and Error Handling** 5.1 Reading from and writing to files 5.2 Understanding file permissions and security considerations 5.3 File Inclusion (Include and Require) 5.4 Managing sessions and cookies 5.5 Implementing session-based authentication and authorization 5.6 Error handling in PHP: try-catch blocks, exception handling
• **Specific Objectives**	**Contents [8 Hrs.]**
• Understand the importance of using databases in PHP applications for efficient data storage, retrieval, and management. • Connecting PHP applications to a MySQL database using MySQLi extensions. • Acquire skills in executing SQL queries dynamically in PHP to interact with MySQL databases. • Explore transaction management concepts to ensure data consistency and integrity in multi-step database operations.	**Unit 6: Working with Database** 6.1 Introduction to MySQL database management system 6.2 Connecting PHP with MySQL database 6.3 Performing CRUD operations (Create, Read, Update, Delete) 6.4 Executing SQL queries using PHP 6.5 User Registration and Login 6.6 Error handling and transaction management
• **Specific Objectives**	**Contents [6 Hrs.]**
• Understand the importance and benefits of	**Unit 7: Advanced PHP Concepts**

using object-oriented programming (OOP) in PHP. • Creating and using classes and objects to organize and encapsulate code. • Understand the purpose and usage of static properties and methods, including declaring and accessing static members without the need for class instantiation. • Learn about properties and methods in PHP classes and how they contribute to code organization and reusability. • Explore advanced concepts such as inheritance, encapsulation, and polymorphism. • Learn about PHP magic methods and their role in implementing specific functionality, enhancing code flexibility and extensibility.	7.1 Object-oriented programming (OOP) in PHP 7.2 Creating and using classes and objects 7.3 Properties and Methods 7.4 Inheritance, encapsulation, and polymorphism 7.5 Static Properties and Methods 7.6 Magic Methods 7.7 Practical applications of OOP in PHP development
• **Specific Objectives**	**Contents [6 Hrs.]**
• Understand the concepts of models, views, and controllers, and understand how they interact within the MVC architecture to handle user requests and manage application data. • Learn about the benefits of using a framework for web development, such as code reusability, modularity, and rapid development. • Explore one of the PHP frameworks available, such as **Laravel**, Symfony, CodeIgniter. • Learn about common features provided by PHP frameworks, such as routing, templating, authentication, authorization, and ORM (Object-Relational Mapping).	**Unit 8: PHP Framework** 8.1 MVC Model 8.2 Benefits of using the PHP Framework 8.3 Getting Started with PHP Framework

5. **Laboratory Work**
   It builds the foundation for how to write a program using any high-level language. Hence, this course requires a lot of programming practice so that students will be able to develop good logic building and program developing capability which is essential throughout the course.

   Some important contents that should be included in lab exercises are as follows:
   1. Create a basic PHP web page that displays dynamic content and interacts with the user.
   2. Create an HTML form with various input elements and server-side validation.
   3. Create a form to upload images using PHP.
   4. Implement session-based authentication and authorization.
   5. Implement CRUD operation in MySQL Database using PHP.
   6. General concept of Laravel, Symfony, or CodeIgniter for rapid web development.

Note:
   1. Motivate students to create small project work integrating all the above concepts.
   2. Each of the above lab sessions should cover more than 4 hours of practical work.

6. Evaluation system and Student's Responsibilities

In addition to the formal exam(s), the internal evaluation of a student may consist of quizzes, assignments, lab reports, projects, class participation, etc. The tabular presentation of the internal evaluation is as follows.

External Evaluation	Marks	Internal Evaluation	Weight	Marks
Semester-End Examination	50	**Theory**		
		Attendance & Class Participation	10%	30
		Assignments	20%	
		Presentations/Quizzes	10%	
		Internal Assessment	60%	
		Practical		
		Attendance & Class Participation	10%	20
		Lab Report/Project Report	20%	
		Practical Exam/Project Work	40%	
		Viva	30%	50
Total External	50	Total Internal		
Full Marks: 50 + 50 = 100				

7. Students' Requirements

Each student must secure at least 45% marks separately in both internal assessment and practical evaluation with a minimum of 80% attendance in the class in order to appear in the Semester End Examination. Failing to get such a score will be given NOT QUALIFIED (NQ) to appear in the Semester-End Examinations. Students are advised to attend all the classes, formal exams, tests, etc., and complete all the assignments within the specified time period. ***Students are required to complete all the requirements defined for the completion of the course.***

8. Prescribed Books and References
Prescribed Text books:
1. Larry Ullman (2018). *PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide (5th Edition)* Peachpit Press
Reference Books:
1. Butler, T., & Yank, K. (2017). *PHP & MySQL: Novice to Ninja (6th Edition).* SitePoint Pty. Ltd.
2. Josh Lockhart (2015*). Modern PHP: New Features and Good Practices (1st Edition).* O'Reilly Media
3. Robin Nixon (2015). *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5 (5th Edition).* O'Reilly Media