



RQF.LEVEL 4

SWDPP401
SOFTWARE
DEVELOPMENT

PHP
Programming

TRAINER'S MANUAL

October, 2024



PHP PROGRAMMING



AUTHOR'S NOTE PAGE (COPYRIGHT)

The competent development body of this manual is Rwanda TVET Board ©, reproduce with permission.

All rights reserved.

- This work has been produced initially with the Rwanda TVET Board with the support from KOICA through TQUM Project.
- This work has copyright, but permission is given to all the Administrative and Academic Staff of the RTB and TVET Schools to make copies by photocopying or other duplicating processes for use at their own workplaces.
- This permission does not extend to making of copies for use outside the immediate environment for which they are made, nor making copies for hire or resale to third parties.
- The views expressed in this version of the work do not necessarily represent the views of RTB. The competent body does not give warranty nor accept any liability.
- RTB owns the copyright to the trainee and trainer's manuals. Training providers may reproduce these training manuals in part or in full for training purposes only. Acknowledgment of RTB copyright must be included on any reproductions. Any other use of the manuals must be referred to the RTB.

© **Rwanda TVET Board**

Copies available from:

- HQs: Rwanda TVET Board-RTB
- Web: www.rtb.gov.rw
- KIGALI-RWANDA

Original published version: October, 2024

ACKNOWLEDGEMENTS

The publisher would like to thank the following for their assistance in the elaboration of this training manual:

Rwanda TVET Board (RTB) extends its appreciation to all parties who contributed to the development of the trainer's and trainee's manuals for the TVET Certificate IV in Software Development, specifically for the module **"SWDPP401 PHP Programming"**

We extend our gratitude to KOICA Rwanda for its contribution to the development of these training manuals and for its ongoing support of the TVET system in Rwanda

We extend our gratitude to the TQUM Project for its financial and technical support in the development of these training manuals.

We would also like to acknowledge the valuable contributions of all TVET trainers and industry practitioners in the development of this training manual.

The management of Rwanda TVET Board extends its appreciation to both its staff and the staff of the TQUM Project for their efforts in coordinating these activities.

This training manual was developed:

Under Rwanda TVET Board (RTB) guiding policies and directives



Under Financial and Technical support of



COORDINATION TEAM

RWAMASIRABO Aimable

MARIA Bernadette M. Ramos

MUTIJIMA Asher Emmanuel

PRODUCTION TEAM

Authoring and Review

MBONYISIMBI Dieudonne

BUKIZI Eric

Validation

NDUNGUTSE Emmanuel

TUYIZERE Emmanuel

Conception, Adaptation and Editorial works

HATEGEKIMANA Olivier

GANZA Jean Francois Regis

HARELIMANA Wilson

NZABIRINDA Aimable

DUKUZIMANA Therese

NIYONKURU Sylvestre

NTEZIMANA Aimable

Formatting, Graphics, Illustrations, and infographics

YEONWOO Choe

SUA Lim

SAEM Lee

SOYEON Kim

WONYEONG Jeong

NYIRANTEZIMANA Odette

Financial and Technical support

KOICA through TQUM Project

TABLE OF CONTENT

AUTHOR'S NOTE PAGE (COPYRIGHT)-----	ii
TABLE OF CONTENT -----	vi
ACRONYMS-----	viii
INTRODUCTION -----	1
MODULE CODE AND TITLE: SWDPP401 PHP PROGRAMMING -----	2
Learning Outcome 1: Apply PHP Fundamental -----	3
Key Competencies for Learning Outcome 1: Apply PHP Fundamentals. -----	4
Indicative content 1.1: Preparation of PHP Programming Environment -----	6
Indicative content 1.2: Application of PHP Concepts -----	9
Indicative content 1.3: Application of PHP Security Concepts -----	14
Indicative content 1.4: Implementation of Object-Oriented Programming (OOP) in PHP -----	17
Learning outcome 1 end assessment -----	20
Further information to the trainer -----	25
Learning Outcome 2: Connect PHP to the Database -----	26
Key Competencies for Learning Outcome 2: Connect PHP to the Database. -----	27
Indicative content 2.1: Application of Database Connection Drives -----	29
Indicative content 2.2: Perform Database CRUD Operations -----	32
Indicative content 2.3: Application of PHP Basic Security Concepts -----	35
Indicative content 2.4: Errors and Exceptions in PHP -----	38
Indicative content 2.5: Implementation of User Authentication -----	40
Learning outcome 2 end assessment -----	43
Further information to the trainer -----	49
Learning Outcome 3: Build a Content Management System (CMS) using PHP -----	50
Key Competencies for Learning Outcome 3: Build a Content Management System (CMS) using PHP. -----	51
Indicative content 3.1: Preparation of Content Management System (CMS) -----	53
Indicative content 3.2: Build Dynamic Content Navigation -----	56
Indicative content 3.3: Management of Cookies and Sessions -----	58

Indicative content 3.4: Application of Context and Options -----	60
Indicative content 3.5: Regulate Page Access -----	62
Indicative content 3.6: CMS Errors Detection -----	65
Indicative content 3.7: Maintain CMS-----	68
Learning outcome 3 end assessment -----	72
Further information to the trainer -----	76
Learning Outcome 4: Build a Web App Using MVC Framework (LARAVEL) -----	77
Key Competencies for Learning Outcome 4: Build a web app using MVC Framework (LARAVEL).-----	78
Indicative content 4.1: Framework Environment Configuration -----	80
Indicative content 4.2: Setup Laravel Custom Routing -----	87
Indicative content 4.3: Perform Form Data Validation -----	90
Indicative content 4.4: Perform CRUD Operations-----	94
Indicative content 4.5: Manage APIs in Laravel Frameworks -----	98
Indicative content 4.6: Authentication and Security-----	101
Indicative content 4.7: API Versioning and Documentation -----	104
Learning outcome 4 end assessment -----	107
Further information to the trainer -----	112

ACRONYMS

ACL: Access Control List

API: Application Programming Interface

CDN: Content Delivery Network

CLI: Command Line Interface

CORS: Cross-Origin Resource Sharing

CRUD: Create, Read, Update, Delete

CSRF: Cross-Site Request Forgery

DBMS: Database Management System

DNS: Domain Name System

Eloquent ORM: Eloquent Object-Relational Mapping

HTTP: Hypertext Transfer Protocol

HTTPS: Hypertext Transfer Protocol Secure

IDE: Integrated Development Environment

JSON: JavaScript Object Notation

JWT: JSON Web Token

KOICA: Korea International Cooperation Agency

LAMP : Linux Apache MySQL/MariaDB PHP/Python/Perl

MAMP: macOS Apache MySQL/MariaDB PHP/Python/Perl

MVC: Model-View-Controller

MySQL: My Structured Query Language

MySQLi: MySQL Improved

ORM: Object-Relational Mapping

PDO: PHP Data Objects

PHP: Hypertext Preprocessor

RDBMS: Relational Database Management System

REST: Representational State Transfer

SPA: Single Page Application

SQL: Structured Query Language

SQLi: SQL Injection

SSH: Secure Shell

SSL: Secure Sockets Layer

TLS: Transport Layer Security

UI: User Interface

URL: Uniform Resource Locator

UX: User Experience

WAMP: Windows Apache MySQL/MariaDB PHP

XAMPP: Cross-Platform Apache MySQL/MariaDB PHP Perl

XSS: Cross-Site Scripting

INTRODUCTION

This trainer's manual includes all the methodologies required to effectively deliver the module titled "**PHP Programming**". Trainees enrolled in this module will engage in practical activities designed to develop and enhance their competencies.

The development of this training manual followed the Competency-Based Training and Assessment (CBT/A) approach, offering ample practical opportunities that mirror real-life situations.

The trainer's manual is organized into Learning Outcomes, which is broken down into indicative content that includes both theoretical and practical activities. It provides detailed information on the key competencies required for each learning outcome, along with the objectives to be achieved.

As a trainer, you will begin by asking questions related to the activities to encourage critical thinking and guide trainees toward real-world applications in the labour market. The manual also outlines essential information such as learning hours, didactic materials, and suggested methodologies.

This manual outlines the procedures and methodologies for guiding trainees through various activities as detailed in their respective trainee manuals. The activities included in this training manual are designed to offer students opportunities for both individual and group work. Upon completing all activities, you will assist trainees in conducting a formative assessment known as the end learning outcome assessment. Ensure that trainees review the key reading and the points to remember section.

MODULE CODE AND TITLE: SWDPP401 PHP PROGRAMMING

Learning Outcome 1: Apply PHP Fundamentals

Learning Outcome 2: Connect PHP to the Database

Learning Outcome 3: Build a Content Management System (CMS) using PHP

Learning Outcome 4: Build a web app using MVC Framework (LARAVEL)

Learning Outcome 1: Apply PHP Fundamental



Indicative contents

1.1. Preparation of PHP Programming environment

1.2. Application of PHP concepts

1.3. Application of PHP Security concepts

1.4. Implementation of Object-oriented programming (OOP) in PHP

Key Competencies for Learning Outcome 1: Apply PHP Fundamentals.

Knowledge	Skills	Attitudes
• Description of PHP programming environment • Description of PHP concepts. • Description of OOP • Description Function • Description of loops, arrays and control structures • Describe PHP security concepts	• Installing XAMPP/WAMPP • Configure the environment XAMPP/WAMPP • Applying PHP concepts • Implementing loop array and control structure • Using PHP functions • Implementing Object-oriented programming (OOP) in PHP.	• Having a teamwork spirit • Having Time management ability • Having curiosity • Being creative • Being adaptive



Duration: 40 hrs



Learning outcome 1 objectives:

By the end of the learning outcome, the trainees will be able to:

1. Describe correctly PHP concepts based on PHP standards.
2. Prepare properly PHP environment based on system requirements
3. Install properly XAMPP/WAMP based on platform requirements
4. Apply effectively PHP concepts based on best practices.
5. Apply conditional statements as it used in PHP programming
6. Apply effectively PHP security concepts based on project requirements.
7. Apply correctly different types of arrays based on project requirements
8. Apply effectively control structure as it used in PHP programming
9. Implement effectively PHP functions based on project requirements
10. Implement correctly OOP concepts based on PHP standards.



Resources

Equipment	Tools	Materials
• Computer	• Text Editor • IDE • Browsers • XAMPP/WAMP	• Internet • Electricity



Advance Preparation:

Before delivering this learning outcome, you are recommended to:

- Ensure that all computers are connected to internet



Indicative content 1.1: Preparation of PHP Programming Environment



Duration: 7 hrs



Theoretical Activity 1.1.1: Description of PHP and related key terms



Notes to the trainer:

- Use small groups to describe PHP environment.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask Trainees to answer to the following questions:

- i. Define the following terms:
 - a) PHP
 - b) Interpreter
 - c) Open source
 - d) Web server
 - e) Database
 - f) DBMS
 - g) MySQL
 - h) Apache
- ii. Differentiate static website from dynamic website.
- iii. What are purposes of PHP?
- iv. List five characteristics of PHP
- v. What are notable tools used in PHP programming

Step 2: Ask any trainees to write answers provided on flipchart/paper.

Step 3: Ask Trainees to present their findings to the whole class

Step 4: Provide expert view and clarifies ideas by using didactic materials.

Step 5: Address any questions or concerns.

Step 6: Ask trainees to read the key reading 1.1.1 in the trainee manual.



Points to Remember

- To be familiar with PHP there are some key terms you have to know such as web browser, webserver, DBMS, and interpreters.

- PHP is server-side scripting, which means it can only run on the web server and generates dynamic web pages. It is often embedded within HTML code to add functionality and interactivity to websites.
- The primary purpose of PHP is to enable the development of dynamic web applications.
- PHP's important characteristics include ease of learning, server-side scripting, cross-platform compatibility, open-source nature, strong database support, a rich ecosystem of frameworks and libraries,
- Some open-source tools are used to build PHP environments such as XAMPP, WAMP, and LAMP



Practical Activity 1.1.2: Setting up the PHP environment

Notes to the trainer

- Avail computer Lab with fully functioning computers
- Avail internet connectivity for all computers



Key steps:

While delivering this activity, pass through the following steps:

- Step 1:** Introduce the activity and ask the trainees to perform the following task
You are requested to go to the computer lab, install and configure PHP environment. This task should be done individually.
- Step 2:** Demonstrate how to install and configure PHP environment. While demonstrating, explain the steps to follow.
- Step 3:** Ask the trainees to go ahead and install and configure PHP environment
- Step 4:** Monitor the activity and Provide expert feedback
- Step 5:** Ask trainees to read key readings 1.1.2 in the trainee manuals for more clarifications.

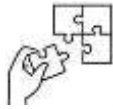


Points to Remember

- There are several steps to set up a PHP environment such as:
 - ✚ Installing development tools platforms (XAMPP, WAMP and others)
 - ✚ Configuring ports and services

 Installation of IDE extensions.

- The modification in ports configuration often causes the environment to fail. Consider reconfiguring.



Application of learning 1.1.

MilleHills Ltd is a company that provides services related to Web Development and it is located in Rubavu District, Gisenyi town. Recently they have started working on a new project that aimed at enlarging their business activities by including web awareness to the general public as a product to sell. They plan to accommodate a large number of people which implicates the need of more computers ready for use. However, MilleHills Ltd does have only few ready computers, others need to be worked on for their readiness. As a trainee who knows what it takes to work in an environment that needs PHP Programming like Web Development, you are hired at MilleHills to set up and configure PHP environment for the rest of their computers.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	PHP environment is correctly set up	1.1 System requirement are checked		
		1.2 XAMPP installer is correctly used		
		1.3 XAMPP is installed		
		1.4 ports are configured		
		1.5 Services are started		
		1.6 Localhost is working		
		1.7 PHPMyAdmin is working		
		1.8 No errors are detected		



Indicative content 1.2: Application of PHP Concepts



Duration: 20 hrs



Theoretical Activity 1.2.1: Description of PHP basic concepts



Notes to the trainer:

- Use small groups for describing PHP basics concepts



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and request Trainees to answer to the following questions:

- i. What do you understand by the following terms?
 - a) Variable
 - b) Operators
 - c) Data types
 - d) Variable scope
 - e) Constants
 - f) Comment
 - g) String concatenation
 - h) Condition statement
 - i) Loop
- ii. Differentiate types of PHP arrays
- iii. Differentiate types of PHP Function.
- iv. What do you understand by PHP file handling?
- v. Explain different categories of Super Global variables found in PHP.
- vi. Explain different PHP security concepts

Step 2: Ask any trainee to write answers provided on flipchart/paper.

Step 3: Ask Trainees to presents their findings to the whole class.

Step 4: Provide expert view and clarifies ideas by using didactic materials.

Step 5: Address any questions or concerns.

Step 6: Ask trainees to read the key reading 1.2.1 in the trainee manual.



Points to Remember

- PHP files can have different file extensions such as .PHP, .phtml, .PHP3, .PHP4,.PHP5, .PHP7 but it is recommended to use only .PHP extension
- Naming variable in PHP you must consider the follow rules:
 - Must start with a dollar sign (\$), followed by a letter or underscore.
 - Cannot start with a number.
 - Can contain letters, numbers, and underscores.
 - Variable names are case sensitive (\$Var and \$var are different variables).
- The difference between Constants and Variables is that constants are automatically global, while variables have varying scopes (global, local, static and Superglobal
- PHP functions are grouped into two main categories such as Built-in functions and user defined functions
- PHP provides several built-in functions like open (), fwrite(), fread(), fclose(), and others to manipulate files in different modes like read, write, append, binary.
- In PHP there are Superglobals variables such as: \$GLOBALS, \$_SERVER, \$_GET, \$_POST, \$_FILES, \$_COOKIE, \$_SESSION, \$_REQUEST, \$_ENV.



Practical Activity 1.2.2: Creating simple PHP program



Notes to the trainer

- Avail computer Lab
- Avail computers with PHP environment configured, Text editor and web browser installed



Key steps:

While delivering this activity, pass through the following steps:

- Step 1:** Introduce the activity and ask the trainees to perform the following task
You are asked to go to the computer lab, to Create a simple PHP program that will display the following message **“Hello software Developers”**. This task should be done individually.
- Step 2:** Demonstrate how to create simple PHP program. While demonstrating, explain the steps to follow.
- Step 3:** Ask trainees to create simple PHP Program

- Step 4:** Monitor the activity and assist if necessary
- Step 5:** Provide expert feedback where necessary
- Step 6:** Ask trainees to read key readings 1.1.2 in their manuals.



Points to Remember

- In PHP, there are three type of arrays such as: numeric or indexed, associative and multidimension Arrays
- PHP functions are categorized into two categories like: Built-in function and user defined functions.
- PHP provides a range of built-in functions for file handling, enabling you to perform operations like reading, writing, checking, and manipulating files.



Practical Activity 1.2.3: Applying PHP basic Concepts



Task:

- Avail computer Lab
- Avail computers with PHP environment configured, Text editor and web browser installed



Key steps:

While delivering this activity, pass through the following steps:

Step 1 : Introduce the activity and ask the trainees to perform the following task

You are asked to go to the computer lab, **create PHP programs that will add two numbers and calculate their sum, when the sum is odd number, the program will display message “SUM IS ODD NUMBER” ten times Otherwise it will display the value of sum five times.**

This task should be done individually.

Step 2: Demonstrate how to apply different PHP basic concepts. While demonstrating, explain the different concepts and their working principals such as syntax.

Step 3: Asks trainees to create program and monitor the procedures.

Step 4: Verify whether PHP basic concepts are correctly used in creating program.

Step 5: Provide expert feedback where necessary.

Step 6: Ask trainees to read key readings 1.2.3 in their manuals for more clarifications.



Points to Remember

- To be familiar with PHP programming there are a list of basic concepts that every developer should cover such as: Variables, operators, data types, control structure (like: Conditional statement and iterative statement).



Practical Activity 1.2.4: Applying PHP Functions



Task:

- Avail computer Lab
- Avail computers with PHP environment configured, Text editor and web browser installed



Key steps:

While delivering this activity, pass through the following steps:

Step 1. Introduce the activity and ask the trainees to perform the following task

You are asked to go to the computer lab, **create PHP programs using PHP functions, the program will calculate area of Circle, use Radius which is equal to 5 meters.** This task should be done individually.

Step 2: Demonstrate how to create PHP program using functions. While demonstrating, explain the different steps to follow while creating PHP functions.

Step 3: Ask trainees to create PHP program using functions and monitor the procedures.

Step 4: Verify whether PHP Functions are used to create the program

Step 5: Provide expert feedback where necessary

Step 6: Ask trainees to read key readings 1.2.4 in their manuals for more clarifications.



Points to Remember

- Creating functions in PHP is a fundamental aspect of writing modular, reusable code, Functions allow you to encapsulate blocks of code that perform a specific task, which can then be called multiple times throughout your script.
- There are some basic consideration steps to follow while you are creating function such as: definition of function, add parameters, return value and calling functions.
- PHP provides various functions for handling files, including reading, writing, and manipulating files.



Application of learning 1.2.

BEST FOR FUTURE ACADEMY is a school that provides short course training for Trainees and it is located in Nyarugenge District, Kigali City. after each training session, they give an assessment to the trainees in order to certify them according to their performance. You are asked to develop a simple PHP script that manages student grades for a class. The Script will use variables, arrays, and functions to calculate and display the average grade, as well as to identify students who have passed or failed.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	The PHP script is properly developed	1.1 Variables are used		
		1.2 Data types are used		
		1.3 Operator are used		
		1.4 Arrays are used		
		1.5 Function are used		



Indicative content 1.3: Application of PHP Security Concepts



Duration: 6 hrs



Theoretical Activity 1.3.1: Description of PHP security concepts



Notes to the trainer:

- Use small groups for describing PHP security concepts.
- Avail computers with text editors and PHP environments configured



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask Trainees to answer to the following questions:

- What do you understand by the term form handling?
- Define the following terms:
 - Validation
 - Security
- What is the difference between session and Cookies?
- What are the differences between POST and GET?

Step 2: Asks Trainees to write their answers/findings on flipchart/paper.

Step 3: Involves Trainees to present their findings

Step 4: Provides expert view and clarifies ideas by using didactic materials.

Step 5: Address any questions or concerns.

Step 6: Ask trainees to read the key reading 1.3.1 in the trainee manual.



Points to Remember

- PHP provides two Superglobals `$_GET` and `$_POST` for collecting form data for processing
- Common types of validation in PHP are Form Validation, Input Validation, Data Type Validation, Email Validation, Custom Validation Rules.
- To use cookies in PHP, you can use the `setcookie ()` function while to use sessions in PHP, you first need to start a session using the **`session_start ()`** function



Practical Activity 1.3.2: Applying PHP security for forms handling



Notes to the trainer

- Avail The computer lab with fully functioning computers
- Avail sample forms



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

Create a simple registration form that captures a user's name, email, and password. After form submission, the data will be validated, stored in a session, and a cookie will be set to remember the user's email. This task should be done individually.

Step 2: Ask trainees to outline the steps to create and validate the PHP form.

Step 3: Ask the trainees to go ahead and apply the outlined steps to have a PHP validated form.

Step 4: Verify that forms are validated

Step 5: Provide the expert feedback where necessary.

Step 6: Ask trainees to read key readings 1.3.2 in their manuals for more clarifications.



Points to Remember

- In the <form> tag, we have two attributes, **action** and **method** which are used depending on the purpose.
- Creating and validating PHP forms while handling sessions and cookies involves several key steps such as: Create the HTML Form, Process the Form with PHP, Handle Sessions, Handle Cookies, Test Your Form.



Application of learning 1.3.

Global studies University is a university located at Mideast Africa, with at least one hundred employees who work in different departments, they use papers and notebooks to record information related to employees such as Employees identification, contact information, Qualifications, Positions and salaries. They want someone to digitalize their activities in order to integrate with new technologies, so you are hired to create and validate Employees

Registration form which will help university to record employee's information into the database.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	Form is properly created	1.1 Action attribute is provided		
		1.2 Method attribute is specified		
		1.3 Text box field is created		
		1.4 Submit button is created		
2	Form is correctly validated	2.1 Input is sanitized		
		2.2 All field are settled to be required		
		2.3 Email format is checked		
3	Session and cookies are properly applied	3.1 Session variables are created		
		3.2 Cookie is settled		



Indicative content 1.4: Implementation of Object-Oriented Programming (OOP) in PHP



Duration: 7 hrs



Theoretical Activity 1.4.1: Description of PHP Object-Oriented Programming



Notes to the trainer:

- Use small groups for Describing PHP Object oriented programming



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Involve Trainees for group formulation and introduce the activity by Trainees to answer the following questions:

- What do you understand by Object oriented programming?
- Define the following terms:
 - Inheritance
 - Polymorphism
 - Abstraction
 - Encapsulation
- Differentiate classes from object.

Step 2: Ask trainees to write answers provided on flipchart/paper.

Step 3: involve the Trainees to discuss to the provided answers and choose correct answer.

Step 4: Provide expert view and clarify ideas by using didactic materials.

Step 5: Address any questions or concerns.

Step 6: Ask trainees to read the key reading 1.4.1 in the trainee manual.



Points to Remember

- To be more professional in PHP programming you must be familiar with these concepts: class, object, inheritance, polymorphism, data abstraction and encapsulation.
- PHP provides a special function called **__construct ()** to define a constructor and **__destruct ()** to define a destructor function.



Practical Activity 1.4.2: Applying OOP Concepts



Notes to the trainer

- Avail The computer lab with fully functioning computers.



Key steps:

While delivering this activity, pass through the following steps:

Step 1:Introduce the activity and ask the trainees to perform the following task:

You are asked to go to the computer lab, create a simple system for managing a library by applying classes, objects, inheritance, encapsulation, and polymorphism. This task should be done individually.

Step 2:Demonstrate how to apply OOP in PHP. While demonstrating, explain the different OOP concepts in PHP.

Step 3:Ask the trainees to go ahead and apply class, object, inheritance, abstraction and polymorphism to have a PHP Simple library management system.

Step 4:Verify and monitor the process of activity

Step 5:Provide your expert feedback where necessary.

Step 6:Ask trainees to read key readings 1.4.2 in their manuals for more clarifications.



Points to Remember

- To be more professional in PHP programming you must be familiar with these concepts: class, object, inheritance, polymorphism, data abstraction and encapsulation.
- There several steps to follow while applying PHP OOP Like: Understand OOP Concepts , Design Your Classes, Create Classes and Objects, Apply Encapsulation, Implement Inheritance, Use Polymorphism, Implement Abstraction, Test Your Implementation, Refactor and Optimize, Document Your Code.



Application of learning 1.4.

ABC_MEDIA company needs a system to manage employee records. The system should handle basic functionalities such as adding new employees, viewing employee details, and updating employee information. As web Application developer you are asked to develop a solution system which will help ABC_MEDIA to manage their employees 'records. You are required to use PHP to create this system with OOP concepts to ensure modular and maintainable code.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	OOP Concepts are properly applied	1.1 Class is used		
		1.2. Object are properly created		
		1.3 Function is used		
		1.4 Construct is used		
		1.5 Private variable is used		



Learning outcome 1 end assessment

Theoretical assessment

I. Read the following statement related PHP programming and choose the correct letter that corresponding to the correct

1. Which of the following is a valid PHP variable name?

- A) \$1st_variable
- B) \$variablename
- C) \$variable_name
- D) \$variable name

Answer: C) \$variable_name

2. What is the output of the following PHP code?

```
$number = 5;  
$text = "The number is $number";  
echo $text;
```

- A) The number is 5
- B) The number is \$number
- C) The number is
- D) The number is 5\$

Answer: A) The number is 5

3. Which of the following is not characteristics of static website?

- A) Fixed Content
- B) No Server-Side Processing
- C) Easy to Develop
- D) Dynamic Content

Answer: D) Dynamic Content

4. Among the following important characteristic of PHP which describe that PHP applications can be scaled to handle high traffic and large datasets?

- A) Open Source
- B) Easy to learn
- C) Scalability
- D) Integration Capabilities

Answer: C) SCALABILITY

5. What is the data type of the variable \$array in the following code?

```
$array = array(1, 2, 3);
```

- A) string
- B) integer
- C) array
- D) object

Answer: C) array

Explanation: The array data type is used to store multiple values in a single variable. The variable \$array is of type array.

6. Which of the following PHP constructs is used to execute a block of code repeatedly while a condition is true?

- A) if
- B) switch
- C) while
- D) continue;

Answer: C) while

7. What is the purpose of the break statement in PHP control structures?

- A) To exit the current loop or switch statement
- B) To skip the current iteration of a loop
- C) To terminate a script execution
- D) To pause the execution of the script

Answer: A) To exit the current loop or switch statement

8. Which PHP function is used to check if a key exists in an associative array?

- A) array_key_exists()
- B) key_exists()
- C) has_key()
- D) in_array()

Answer: A) array_key_exists()

9. What will be the output of the following PHP code?

```
$arr = array("a" => 1, "b" => 2, "c" => 3);  
foreach ($arr as $key => $value) {  
    echo "$key: $value\n"; }
```

A)

a: 1

b: 2

c: 3

B)

1: a

2: b

3: c

C)

a: 1

2: b

c: 3

D)

1: a

2: b

c:3

Answer:

A)

a: 1

b: 2

c: 3

10. Which of the following is the correct way to access the value 2 in the following PHP array?

`$numbers = array (1, 2, 3, 4);`

A) `$numbers[2]`

B) `$numbers[1]`

C) `$numbers['2']`

D) `$numbers.get(2)`

Answer: B) \$numbers [1]

II. Complete the following sentences with appropriate terms used to PHP Programming:

1.To define a function in PHP, you use the _____ keyword, followed by the function name and parentheses.

Answer: function

2.To return a value from a PHP function, you use the _____ statement.

Answer: return

3.To call a function and pass an argument by reference, you use the _____ symbol before the parameter in the function definition.

Answer: &

III. Read the following statement related to php programming and write the letter corresponding to the correct answer.

Answer	PHP Concepts	OOP	Explanations
....B....	1.Encapsulation		A. The process of creating a new class based on an existing class, allowing the new class to inherit properties and methods from the parent class.
.....A....	2.Inheritance		B. The practice of hiding the internal implementation details of a class and only exposing the necessary parts of it.
....C....	3.Polymorphism		C. The ability of different classes to be treated as instances of the same class through a common interface, even though they may behave differently.
....E...	4.Abstraction		D. A special method that is automatically called when an object is instantiated.

Practical assessment

DSG Ltd is company which sells smartphone, tablets and other digital devices, and they don't have digital way of managing their store. They want someone who can create web based application which will help them to know available store for each product, but they don't need to use any DBMS simple to use PHP arrays and other related concepts to build the system. so as web application developer, Imagine a web based inventory management system for a retail store and build the system to display a list of products, allow for product updates, and manage stock levels for this company.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	System is correctly created using PHP basic concepts	1.1 List of products are displayed		
		1.2 Product can be updated		
		1.3 Stock level can be managed		



Further information to the trainer

Bierer, D. (2016). *PHP 7 programming book*. Birmingham, B32PB, UK: Pack Publishing Ltd.

Nixon, R. (2014). *Learning PHP, MySQL, JavaScript, CSS & HTML* (3rd ed.). USA: O'Reilly Media.

Popel, D. (2007). *Learning data object*. Luna Park, Sydney: Pack Publisher.

Powers, K. T. (2019). *PHP for the web: Visual quickstart guide* (5th ed.). Peachpit Press.

Ullman, L. (2017). *PHP and MySQL for dynamic web sites: Visual quickpro guide* (5th ed.). Peachpit Press.

FreeCodeCamp. (2024, August 26). *How to use arrays in PHP?* Retrieved from <https://www.freecodecamp.org/news/howtousearraysinPHP/>

Scaler. (2024, August 26). *File handling in PHP*. Retrieved from <https://www.scaler.com/topics/PHPtutorial/filehandlingPHP/>

Simplilearn. (2024, August 26). *Hello world in PHP*. Retrieved from <https://www.simplilearn.com/tutorials/PHPtutorial/helloworldinPHP>

Shiksha. (2024, August 30). *Difference between GET and POST in PHP*. Retrieved from <https://www.shiksha.com/onlinecourses/articles/differencebetweengetandpostinPHPblogId155719#2>

TutorialsPoint. (2024, August 26). *PHP functions*. Retrieved from https://www.tutorialspoint.com/PHP/PHP_functions.htm

Learning Outcome 2: Connect PHP to the Database



Indicative contents

- 2.1. Application of Database Connection drives
- 2.2. Perform database CRUD Operations
- 2.3. Application of PHP Basic security concepts
- 2.4. Errors and exceptions in PHP
- 2.5 Implementation of user authentication

Key Competencies for Learning Outcome 2: Connect PHP to the Database.

Knowledge	Skills	Attitudes
● Description of Database Connections ● Explanation of CRUD operations ● Description of PHP errors and exceptions ● Interpretation of user authentication	● Utilizing different Database connections ● Performing CRUD Operations ● Handling errors in PHP programming ● Implementing user authentication	● Having curiosity ● Being creative ● Being adaptive ● Being confident ● Being collaborative ● Having Teamworking spirit



Duration: 37 hrs



Learning outcome 2 objectives:

By the end of the learning outcome, the trainees will be able to:

1. Describe correctly database connections based on their applications context
2. Differentiate correctly MySQLi from PDO based on their key features
3. Connect properly PHP to Database based on the Database environment
4. Perform effectively CRUD Operations based on PHP standards
5. Apply effectively basic security concepts accordingly to PHP standards
6. Handle correctly errors and exceptions based on the error handling standards
7. Implement correctly user authentication in respect to Web Page Controls



Resources

Equipment	Tools	Materials
• Computer	• Text Editor • IDE • Web Browsers • XAMPP/WAMP	• Internet • Electricity •



Advance Preparation:

Before delivering this learning outcome, you are recommended to:

- Have didactic materials
- Avail functioning computers



Indicative content 2.1: Application of Database Connection Drives



Duration: 9 hrs



Theoretical Activity 2.1.1: Description of key terms related to PHP and database connection



Notes to the trainer:

- Use small groups for describing PHP and Database connection.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask Trainees to answer to the following questions:

- i. Define the following terms
 - a) Database
 - b) DBMS
- ii. Differentiate database connection drives from database connection drivers.
- iii. What do you understand by MySQLi
- iv. Differentiate MySQLi from PDO

Step 2: Ask any trainees to write answers provided on flipchart/paper.

Step 3: Ask trainees to present their findings to the whole class

Step 4: Provide expert view and clarifies ideas by using didactic materials.

Step 5: Address any questions or concerns.

Step 6: Ask trainees to read the key reading 2.1.1 in the trainee manual.



Points to Remember

- Database drives are about where the database is physically stored, while database drivers are about how software communicates with the database.
- MySQLi is an improved version of the older mysql extension and offers a more robust and feature rich set of tools for database operations.
- PDO is often favoured for its object-oriented approach and its support for prepared statements, which help prevent SQL injection attacks and Provides a consistent API across different database systems, making it easier to switch between them rather than MySQLi which focus only on MYSQL Database.



Practical Activity 2.1.2: Connecting PHP and database



Notes to the trainer

- Avail The computer lab with fully functioning computers and the same PHP environment correctly configured
- Prepare sample database within the same name for all computers
- Avail same database management system to all computers



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task.

Create PHP scripts that will be used to connect to the localhost server and database called “**My_firstDb**” then display message saying that “**Database is selected successful**”. This task should be done individually.

Step 2: Demonstrate how to connect PHP with database. While demonstrating, explain the connecting procedures.

Step 3: Ask trainees to outline steps to follow while establishing connection between PHP and database.

Step 4: Ask the trainees to apply the outlined steps and connect PHP with my_firstdb database.

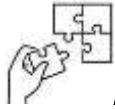
Step 5: Provide expert feedback where necessary.

Step 6: Ask trainees to read key readings 2.1.2 in their manuals for more clarifications.



Points to Remember

- To connect PHP and database requires to follow steps such as: create database, create connection file, test Connection
- Connection file should be included at the top of other files which will need the connection by using include function.



Application of learning 2.1.

TechSolutions Inc. is a mid-sized tech company specializing in custom web applications for clients across various industries. They are currently in the process of developing a new web-based project management tool that will help businesses streamline their project workflows. TechSolutions Inc. has a frontend development team working on the user interface of the project management tool using HTML, CSS, and JavaScript. They are using PHP as the server-side scripting language and have chosen MySQL as their database management system. The backend database design includes tables for users, projects, tasks, and comments. They are seeking someone who is PHP and Database Integration Developer to join their team. You are hired to write scripts which will establish the connection between PHP and their database for smooth integration of their frontend with their database.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	PHP and database connection is properly established	1.1 Connection variables are declared		
		1.2 Functions or object are used		
		1.3 PHP file is correctly saved		
		1.4 Database is properly selected		
		1.5 Project Folder is created		



Indicative content 2.2: Perform Database CRUD Operations



Duration: 10 hrs



Theoretical Activity 2.2.1: Description of CRUD Operations



Notes to the trainer:

- Avail computers and configured PHP environment



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask Trainees to answer to the following questions:

- What do you understand by CRUD Operation in context of web development?
- Define the following terms:
 - SQL
 - Query
- Outline different functions used to perform CRUD Operations.

Step 2: Ask any trainees to write answers provided on flipchart/paper.

Step 3: Ask Trainees to present their findings to the whole class

Step 4: Provide expert view and clarifies ideas by using didactic materials.

Step 5: Address any questions or concerns.

Step 6: Ask trainees to read the key reading 2.2.1 in the trainee manual.



Points to Remember

- SQL, or Structured Query Language, it provides commands to perform various operations on data, such as querying, updating, inserting, and deleting records
- A query is a request for information or data from a database. In SQL, a query is formulated using a structured syntax to retrieve specific data based on certain criteria
- There are three ways you can use by performing CRUD Operations in web-based applications like: MySQLi, MySQLi-OOP and PHP Data object (PDO)



Practical Activity 2.2.2: Performing CRUD Operations



Notes to the trainer

- Avail computers with Apache and MySQL services running properly.
- Avail sample PHP and MySQL project to use as didactic materials



Key steps:

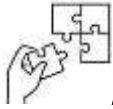
While delivering this activity, pass through the following steps:

- Step 1:** Introduce the activity and ask the trainees to perform the following task:
Create a web-based application using PHP which will allow admin to register user, display user's information, modify the information and delete some information.
- Step 2:** Ask trainees to create Web based application which will be used to perform CRUD Operations.
- Step 3:** Monitor the process of activity and Give assistance where necessary
- Step 4:** Provide expert feedback where necessary.
- Step 5:** Ask trainees to read key readings 2.2.2 in their manuals for more clarifications.



Points to Remember

- Performing CRUD operations requires to follow some steps such as:
- Setup the Development Environment, Design the Database, Connect PHP to MySQL, Implement CRUD Operations, Testing.
- There are three ways you can use by performing CRUD Operations in web-based applications like: MySQLi, MySQLi OOP and PHP Data object (PDO)
- Exporting and importing databases are fundamental operations that help ensure data integrity, support disaster recovery, facilitate migration, and enable efficient data management.



Application of learning 2.2.

RwandaINC Company is cleaning company located at Nyarugenge District, work with different institutions for cleaning their offices and others building. This company has hundreds of employees who help the company to offer cleaning services to the clients, Company want to use digital system to record and manage employee's information. As web application developer, they want you to Develop a web-based Employees Management System that allows users to Register, Modify, and Delete employee information. This application should use PHP and MySQLi to interact with the database.



Indicative content 2.3: Application of PHP Basic Security Concepts



Duration: 8 hrs



Theoretical Activity 2.3.1: Description of PHP basic security concepts



Notes to the trainer:

- Use small groups for describing PHP Security concepts



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask Trainees to answer to the following questions:

- What do you understand by the following terms?
 - Input validation
 - Password Security
 - Session Security
- Differentiate Cross-Site Scripting (XSS) Prevention from Cross-Site Request Forgery (CSRF) Prevention.

Step 2: Ask any trainees to write answers provided on flipchart/paper.

Step 3: Ask Trainees to present their findings to the whole class.

Step 4: Provide expert view and clarifies ideas by using didactic materials.

Step 5: Address any questions or concerns.

Step 6: Ask trainees to read the key reading 2.3.1 in the trainee manual.



Points to Remember

- Input validation is the process of verifying that user inputs (or inputs from other sources) meet certain criteria before they are accepted by the application.
- Effective password security is crucial for safeguarding sensitive information and ensuring the overall security of systems and applications.
- Sessions are used to store user specific data across multiple requests and interactions with a web application
- Proper error reporting helps developers debug issues but can also impact the security of a web application

- CrossSite Scripting (XSS) and CrossSite Request Forgery (CSRF) are both web application vulnerabilities, but they exploit different aspects of the web application's interaction with users



Practical Activity 2.3.2: Implementing PHP security concepts



Notes to the trainer

- Avail the computer Lab



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

You are requested to go to the computer lab, develop an Online Registration system which will help users to register and login using username and password, then apply security concepts such as input validation, session security, password security also try to include the way of reporting all possible errors. This task should be done individually.

Step 2: Ask trainees to develop and secure online registration system as described in the given Task.

Step 3: Monitor the process and give any assistance where necessary

Step 4: Provide your expert feedback where necessary.

Step 5: Ask trainees to read key readings 2.3.2 in their manuals for more clarifications.



Points to Remember

- There are several best practices to implement PHP security concepts like: Secure Database Queries, Validate and Sanitize Input, Implement Secure Session Management, Secure File Uploads, Protect Against CrossSite Request Forgery (CSRF), Use HTTPS.



Application of learning 2.3.

DevTech Ltd company provides services of developing web application system for different institutions but they are still facing security issues and system errors, they decided to create a development team for enhancing the security while developing system. Suppose that you are hired as part of a development team responsible for enhancing the security of a web application. This application must provide functionalities such as user registration, login, data management, and transaction processing. Your goal is to integrate security measures into the application code and ensure that these measures are effective.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	System is effectively secured by applying Security concepts	1.1 Database Queries are secured		
		1.2 Input are sanitized and validated		
		1.3 Session is applied		
		1.4 File upload is secured		
		1.5 HTTPs is used		
		1.6 Protection against CSRF is applied		



Indicative content 2.4: Errors and Exceptions in PHP



Duration: 5 hrs



Practical Activity 2.4.1: Handling PHP Exceptions and Errors



Notes to the trainer

- Avail computers with PHP and database environment configured
- Avail PHP scripts with all types of PHP Errors



Key steps:

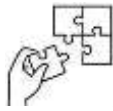
While delivering this activity, pass through the following steps:

- Step 1:** Introduce the activity and ask the trainees to perform the following task
- Examine the given PHP scripts, identify Parse or syntax errors, Fatal errors, Warning errors, notice errors then Correct the errors and perform exceptions handling. This task should be done individually.
- Step 2:** Ask the trainees to fix the identified errors and Perform exception handling.
- Step 3:** Monitor the activity and provide expert feedback where necessary
- Step 4:** Ask any trainee to present the work done to the whole class
- Step 5:** Ask trainees to read key readings 2.4.1 in their manuals for more clarifications.



Points to Remember

- A PHP error comes in 4 different types such as: Parse or syntax errors, Fatal errors, Warning errors, Notice errors.
- Exceptions in PHP are objects representing errors or exceptional conditions that occur during script execution.
- Throwing exceptions is done using the throw keyword, and catching exceptions is done using try and catch blocks.
- Custom exceptions can be created by extending the Exception class to handle specific types of errors.
- The finally block ensures that certain code runs regardless of whether an exception was thrown or not.



Application of learning 2.4.

Tech Solutions Inc. is a mid-sized technology company specializing in software solutions for small to medium-sized businesses. They offer a range of services including custom software development, IT consulting, and technical support. Tech Solutions Inc. is in the process of developing a new Customer Management System (CMS) designed to streamline their client interactions and improve data handling. The system will allow users to manage customer profiles, track interactions, generate reports, and integrate with other business systems. As the development team approaches the final stages of the CMS project, they have encountered several issues that need addressing, so Tech Solutions Inc. is hiring you as a PHP developer to provide the following skills and expertise: Error Reporting and Handling, **Exception Management** to Configure PHP error reporting settings to facilitate debugging during development, Implement custom error handling to catch and manage different types of errors gracefully and Develop a robust exception handling strategy using try-catch blocks to manage exceptions and maintain application stability.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	Errors are correctly identified and fixed	1.1 Syntax errors are identified and fixed		
		1.2 Fatal errors are identified and fixed		
		1.3 Warning errors are identified and fixed		
		1.4 Notice errors are identified and fixed		
2	Exceptions are properly handled	2.1 Throw is used		
		2.2 Try and catch are used		
		2.3 Custom exception is applied		



Indicative content 2.5: Implementation of User Authentication



Duration: 5 hrs



Theoretical Activity 2.5.1: Introduction on user authentication and authorization



Notes to the trainer:

- Avail computers and internet
- Prepare sample system where you access to show the difference between user authentication and user authorization.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and request Trainees to answer to the following questions:

- What do you understand by the following terms?
 - Authentication
 - Authorization
- What are different types of user authentication?
- What is the purpose of user authentication?
- List the types of user authorization

Step 2: Ask any trainees to write answers provided on flipchart/paper.

Step 3: Ask Trainees to presents their findings to the whole class

Step 4: Provide expert view and clarifies ideas by using didactic materials.

Step 5: Address any questions or concerns.

Step 6: Ask trainees to read the key reading 2.5.1 in the trainee manual.



Points to Remember

- The purpose of authentication is to verify that someone or something is who or what they claim to be.
- There are several types of user authentication such as: Password based, two factor/multifactor, biometric, SSO, Token based and certificated based authentication.

- Authorization is the process of determining what actions or resources a user is allowed to access after their identity has been authenticated
- There are also different types of authorization like: Role based Access Control, attribute-based access control, Discretionary access Control and Mandatory access control.



Practical Activity 2.5.2: Performing user authentication



Notes to the trainer

- The computer lab should be available with fully functioning computers



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task

You are requested to go to the computer lab, create a system which will help users to create user Account, login into the system using created user account, display welcome message showing the name of user who logged in and provide logout to exit the system. This task should be done individually.

Step 2: Demonstrate how to create user authentication. While demonstrating, explain the steps and procedures.

Step 3: Ask the trainees to perform user authentication as given in the task

Step 4: Monitor the activities and give assistance where necessary.

Step 5: Provide your expert feedback where necessary.

Step 6: Ask the trainees to perform the task provided in the Application of Learning 2.5



Points to Remember

- Creating user authentication requires several steps such as: setting up database, creating registration scripts, creating login scripts, creating logout scripts and protecting welcome page.



Application of learning 2.5.

A company called **Task_tack** needs a new webbased tasks management system. The system will be used by employees to manage and track their tasks. The company needs an authentication system to ensure that only authorized users can access the application and perform certain actions based on their roles. As web developer you are hired to provide solution to the needs of Task_tack company.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	Authentication is properly created	1.1 Database is created		
		1.2 Table is created		
		1.3 User registration is created and performed		
		1.4 Login is secured		
		1.5 Login is working properly		
		1.6 Logout is working		



Learning outcome 2 end assessment

Theoretical assessment

I. Read the following statement related PHP programming and choose the correct letter that corresponding to the correct

1. What is the primary purpose of user authentication?

- a) To verify user roles
- b) To determine access levels
- c) To confirm the identity of a user
- d) To track user activity

Answer: c) To confirm the identity of a user

2. Which PHP function is used to hash passwords securely?

- a) md5()
- b) crypt()
- c) password_hash()
- d) hash()

Answer: c) password_hash()

3. Which of the following is NOT a common role in a rolebased access control system?

- a) Admin
- b) Moderator
- c) Editor
- d) Viewer

Answer: b) Moderator

4. What is a common method to prevent SQL injection attacks?

- a) Input validation
- b) Use of prepared statements
- c) Encrypting data
- d) Regularly updating PHP

Answer: b) Use of prepared statements

5. Which PHP function is used to verify a password against a hashed value?

- a) password_verify()
- b) password_check()
- c) hash_verify()
- d) verify_password()

Answer: a) password_verify()

6. What is the main difference between authentication and authorization?

- a) Authentication determines what a user can do, while authorization verifies identity.

- b) Authentication verifies identity, while authorization determines what a user can do.
- c) Authentication is used for encryption, while authorization is for hashing.
- d) Authentication involves login, while authorization involves logout.

Answer: a) Authentication determines what a user can do, while authorization verifies identity

7. What is a session fixation attack?

- a) An attack that involves stealing a session cookie.
- b) An attack that involves creating a fake session to impersonate a user.
- c) An attack that modifies session data to gain unauthorized access.
- d) An attack that deletes user sessions to disrupt service.

Answer: b) An attack that involves creating a fake session to impersonate a user.

8. Which HTTP method is typically used for user login forms?

- a) GET
- b) POST
- c) PUT
- d) DELETE

Answer: b) POST

9. What should be done to enhance session security in PHP?

- a) Use session cookies with the Secure and HttpOnly flags.
- b) Store sensitive data in session variables.
- c) Use a simple session ID without regeneration.
- d) Disable session expiration.

Answer: a) Use session cookies with the Secure and HttpOnly flags

10. Which of the following is a common technique to protect against CrossSite Request Forgery (CSRF)?

- a) Using HTTPS
- b) Implementing CSRF tokens
- c) Validating user input
- d) Hashing passwords

Answer: b) Implementing CSRF tokens

11. Which of the following is a purpose of using a database connection driver in PHP?

- a) To manage the graphical user interface of a web application.
- b) To enable PHP scripts to communicate with a database server.
- c) To handle user authentication and session management.
- d) To generate dynamic web content without accessing a database.

Answer: To enable PHP scripts to communicate with a database server.

12. When using the PDO (PHP Data Objects) extension for database connections, which of the following statements is true?

- a) PDO supports only MySQL databases.
- b) PDO provides a unified API for accessing various database management systems.
- c) PDO is a procedural extension only. PDO does not support prepared statements.
- d) PDO does not support prepared statement

Answer: PDO provides a unified API for accessing various database management systems.

13. Which PHP function is used to execute an SQL query that inserts a new record into a MySQL database using the MySQLi extension?

- A) MySQLi_query()
- B) MySQLi_fetch_assoc()
- C) MySQLi_stmt_bind_param()
- D) MySQLi_prepare()

Answer: A) MySQLi_query()

14. Which method is used to update an existing record in a database using PDO in PHP?

- A) PDO::query()
- B) PDO::exec()
- C) PDO::prepare()
- D) PDO::fetch()

Answer: B) PDO::exec()

15. Which of the following PHP functions is commonly used to prevent SQL Injection by escaping special characters in a query string?

- A) htmlspecialchars()
- B) MySQLi_real_escape_string()
- C) strip_tags()
- D) addslashes()

Answer: B) MySQLi_real_escape_string()

16. To protect against Cross-Site Scripting (XSS) attacks, which function should be used to convert special characters to HTML entities in user-generated content before outputting it to the browser?

- A) htmlspecialchars()
- B) base64_encode()
- C) md5()
- D) preg_replace()

Answer: A) htmlspecialchars()

17. Which PHP function can be used to set a custom error handler that replaces the default error handling mechanism?

- A) set_exception_handler()
- B) set_error_handler()
- C) trigger_error()
- D) error_reporting()

Answer: B) set_error_handler ()

18. When an exception is thrown in PHP, which block of code is used to catch and handle the exception?

- A) catch
- B) finally
- C) try
- D) throw

Answer: A) catch

II. Read the following statement related to php programming and complete the statement with appropriate word.

1. In a role-based access control system, the _____ defines what actions a user can perform within an application.

Answer: authorization

2. To securely store user passwords in a database, you should use _____ to hash the passwords before storing them.

Answer: password_hash()

3. The PHP function _____ is used to compare a plaintext password with a hashed password.

Answer: password_verify()

4. When a user logs in, their authentication state is typically managed using a _____.

Answer: session

5. To prevent unauthorized access to certain pages, you should check the user's _____ at the beginning of each protected page.

Answer: role or session

III. Read the following statement related to php programming and write the letter corresponding to the correct answer

Answer	Terms:	Definitions:
...b....	1. Authentication	a. A security measure to prevent attacks where an attacker tricks a user into making unwanted requests.
...d....	2. Authorization	b. The process of verifying a user's identity.
...c....	3. Session Hijacking	c. The act of taking over a user's active session.
...a...	4. CSRF Token	d. The process of determining what actions an authenticated user can perform.

Practical assessment

TWIGIRE MUHINZI COOPERATIVE is a Local cooperative located in Northern Province, Rulindo District. It has many members who cultivate Maize and sell it as cooperative, after harvesting and accurately prepare the crops, every member brought his/her harvest to the cooperative office and store it there, then cooperative looks for clients and sell Maizes then payback to the members relatively to his/her harvest. This Activities are done manually using papers and pens which delay transactions and can lead to the potential loss of data. To fix this issue, they want a web application to manage their members 'information about members, product, client and sales then they can easily generate sales and purchase reports

They hired you as web application developer to develop a web application that enables Administrator of cooperative to create a user account and login with user account information, the login is session based while development. After successful login, he/she

has also to record the Members 'information, product, client and Sales with ability to view, modify and delete records in tables. The application gives a way of displaying a report of sales and product from Members

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	System is created and working properly	1.1 Database is created		
		1.2 Tables is created		
		1.3 User account is created successful		
		1.4 Login is created with session-based settings		
		1.5 Logout is created		
		1.6 Member's information is retrieved		
		1.7 Member's information is updated		
		1.8 Member's information is can be deleted		



Further information to the trainer

Bierer, D. (2016). *PHP 7 programming book*. Birmingham, B32PB, UK: Pack Publishing Ltd.

Nixon, R. (2014). *Learning PHP, MySQL, JavaScript, CSS & HTML* (3rd ed.). USA: O'Reilly Media.

Nixon, R. (2018). *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5* (5th ed.). O'Reilly Media.

Popel, D. (2007). *Learning data object*. Luna Park, Sydney: Pack Publisher.

Powers, K. T. (2019). *PHP for the web: Visual quickstart guide* (5th ed.). Peachpit Press.

Ullman, L. (2017). *PHP and MySQL for dynamic web sites: Visual quickpro guide* (5th ed.). Peachpit Press.

Welling, L., & Thomson, L. (2016). *PHP and MySQL web development* (5th ed.). Addison-Wesley.

FreeCodeCamp. (2024, August 26). *How to use arrays in PHP?* Retrieved from <https://www.freecodecamp.org/news/howtousearraysinPHP/>

Leeds Beckett University Library. *What is the difference between authentication and authorization?* Retrieved from <https://libanswers.leedsbeckett.ac.uk/faq/189738>

OneLogin. *Authentication vs. Authorization*. Retrieved from <https://www.onelogin.com/learn/authentication-vs-authorization>

Raygun. *PHP error reporting*. Retrieved from <https://raygun.com/blog/PHP-error-reporting/>

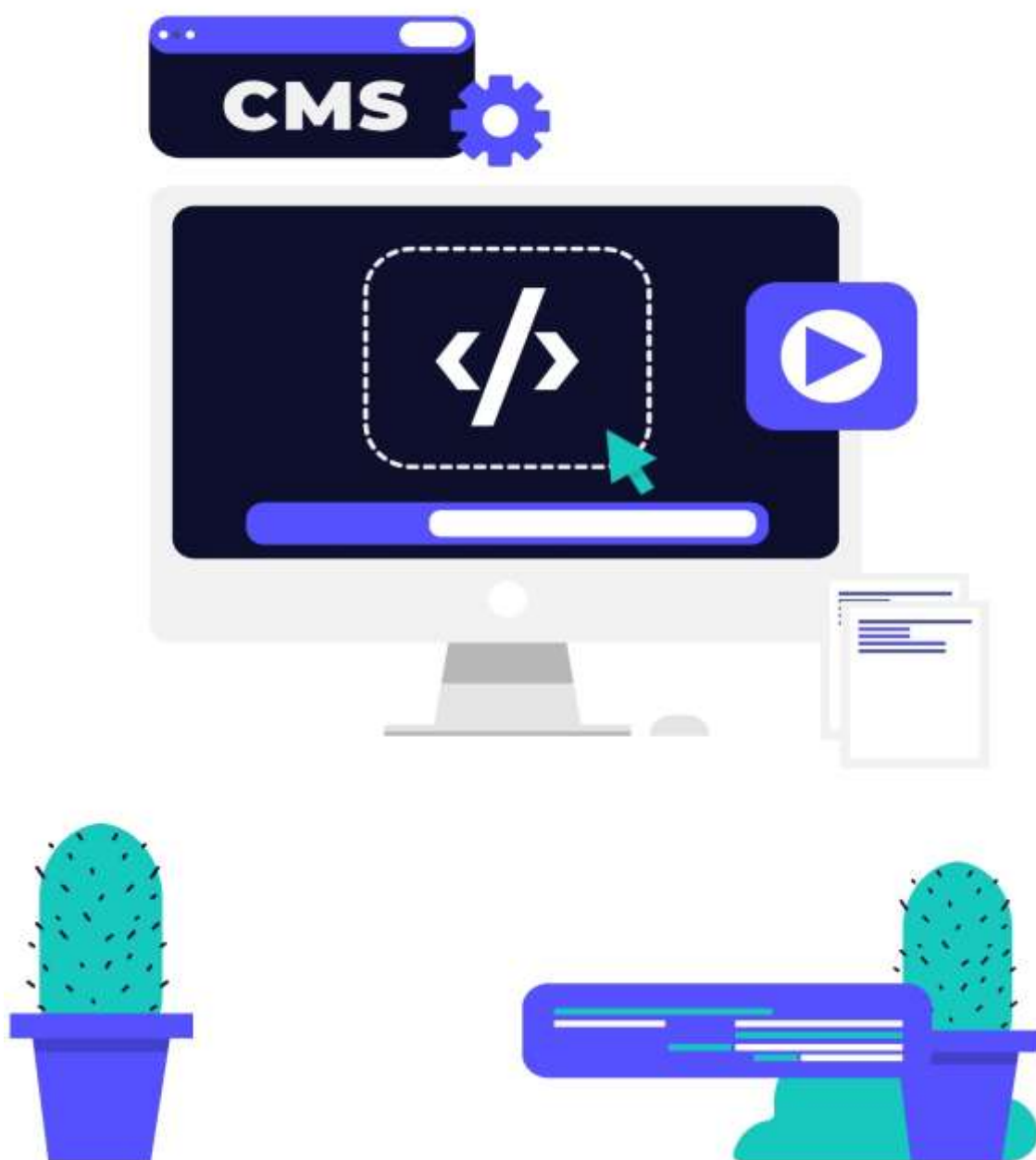
Scaler. (2024, August 26). *File handling in PHP*. Retrieved from <https://www.scaler.com/topics/PHPtutorial/filehandlingPHP/>

Shiksha. (2024, August 30). *Difference between GET and POST in PHP*. Retrieved from <https://www.shiksha.com/onlinecourses/articles/differencebetweengetandpostinPHPblogId155719#2>

Simplilearn. (2024, August 26). *Hello world in PHP*. Retrieved from <https://www.simplilearn.com/tutorials/PHPtutorial/helloworldinPHP>

TutorialsPoint. (2024, August 26). *PHP functions*. Retrieved from https://www.tutorialspoint.com/PHP/PHP_functions.htm

Learning Outcome 3: Build a Content Management System (CMS) using PHP



Indicative contents

3.1. Preparation of Content Management System (CMS)

3.2. Build dynamic content navigation

3.3. Management of cookies and sessions

3.4. Application of Context and Options

3.5. Regulate page access

3.6. CMS Errors Detection

3.7. Maintain CMS

Key Competencies for Learning Outcome 3: Build a Content Management System (CMS) using PHP.

Knowledge	Skills	Attitudes
● Description of Content Management System ● Distinction between Sessions and Cookies ● Description of CMS errors ● Description of CMS context and options	● Preparing a Content Management System environment ● Building navigations for dynamic content ● Managing sessions and cookies ● Regulating content access ● Debugging CMS errors ● Maintaining CMS	● Having teamwork spirit ● Having curiosity ● Being creative ● Being adaptive ● Having patience ● Being flexible ● Being confident ● Being collaborative



Duration: 25 hrs



Learning outcome 3 objectives:

By the end of the learning outcome, the trainees will be able to:

1. Prepare effectively a Content Management System environment in accordance to its requirements.
2. Build rightly a Dynamic Content Navigation based on PHP standards.
3. Manage properly sessions based on PHP standards.
4. Manage properly cookies based on PHP standards.
5. Apply effectively PHP context options in accordance with their standards.
6. Regulated effectively pages access based on PHP standards.
7. Detect correctly CMS errors and logging based on CMS security concepts
8. Maintain effectively CMS based on PHP standards.



Resources

Equipment	Tools	Materials
Computer	- Text Editor - IDE - Browsers - XAMPP/WAMP - Scratch	- Internet - Electricity



Advance Preparation:

Before delivering this learning outcome, you are recommended to:

- Have didactic materials
- Avail functioning computers with XAMPP installed



Indicative content 3.1: Preparation of Content Management System (CMS)



Duration: 5 hrs



Theoretical Activity 3.1.1: Description of a Content Management System environment



Notes to the trainer:

- Use small groups for describing CMS environment.



Key steps:

While delivering this activity, pass through the following steps:

Step 1. Introduce the activity and request trainees to respond to the following questions:

- Define the following terms:
 - CMS
 - Content Management
 - User Roles and Permissions
 - Templates
 - Plugins/Extensions
- Describe the importance of each key concept in a CMS environment.
- Identify the core features to include when blueprinting a CMS application.
- Explain the steps to set up the database for a CMS, including essential tables and relationships.
- Discuss the significance of organizing project files and folders in a CMS, focusing on maintainability and efficiency.

Step 2. Ask trainees to discuss the provided tasks and select the correct answers within their groups.

Step 3. Invite trainees to write the group's answers on the flipchart/paper for review.

Step 4. Encourage the trainees to present their findings and facilitate a discussion around their responses.

Step 5. Provide an expert view and clarify the ideas by using relevant didactic materials to ensure thorough understanding.

Step 6. Ask trainees to read Key Reading 3.1.1 in the trainee manual for additional clarification and encourage them to ask questions where necessary.



Points to Remember

- Align your PHP version with the CMS requirements to avoid compatibility issues.
- Use a dedicated database user with limited permissions and a strong, unique password.
- Follow the CMS's guidelines for server configuration and set appropriate file permissions.
- Schedule regular backups for both CMS files and the database, and test the restoration process.
- Regularly update your CMS, server software, and plugins to protect against vulnerabilities and use strong, unique passwords for all accounts.



Practical Activity 3.1.2: Setting up a working CMS environment with PHP

Programming

Notes to the trainer

- The computer lab should be available with fully functioning computers and XAMPP installed.
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

You are requested to go to the computer lab, prepare and configure your CMS environment by creating the database and tables, setting up project files and directories, and implementing core features such as user registration, login, and CRUD operations for posts. This task should be done individually.

Step 2: Demonstrate how to set up the CMS environment, including creating the database, configuring project files, and implementing core features

Step 3: Ask the trainees to set up and configure CMS environment and monitor the process.

Step 4: Verify the completeness of the CMS environment setup and core features.

Step 5: Provide Expert feedback where necessary.

Step 6: Ask trainees to read Key Reading 3.1.2 in the trainee manual.



Points to Remember

- Ensure PHP, MySQL, and Apache/Nginx are correctly installed and configured before starting the CMS setup.
- Create a dedicated MySQL database and user for the CMS with appropriate permissions and secure credentials.
- Follow the CMS installation instructions carefully to avoid misconfigurations that can lead to security vulnerabilities.
- Set correct file permissions for CMS files and directories to prevent unauthorized access and ensure smooth operation.
- Test the CMS setup thoroughly to verify that all features work as expected and fix any issues before going live.



Application of learning 3.1.

IremboSoft Ltd, a digital solutions provider located in Kigali City, Nyarugenge District, specializes in developing custom software for educational institutions. They are planning to introduce a new Content Management System (CMS) for managing online courses and student resources. However, they need assistance in setting up the CMS environment to ensure it meets the needs of the institutions they serve. You have been hired by IremboSoft Ltd to prepare and configure the CMS environment, including setting up the necessary server requirements, installing the CMS software, and ensuring that it is ready for content creation and management.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	CMS environment is properly settled up	1.1 CMS environment is prepared		
		1.2 CMS environment is configured		
		1.3 Server requirements are checked		
		1.4 CMS software is installed		
		1.5 CMS environment is operational		



Indicative content 3.2: Build Dynamic Content Navigation



Duration: 3hrs



Practical Activity 3.2.1: Building a dynamic CMS navigation



Notes to the trainer

- The computer lab should be available with fully functioning computers and XAMPP installed.
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

Go to the computer lab and build a dynamic CMS navigation. Create the necessary database and PHP scripts to list subjects, display pages, and show content. Implement and test the navigation functionality to ensure it works seamlessly. This task should be done individually.

Step 2: Explain the task and provide clear work instructions.

Step 3: Demonstrate the steps to build a dynamic Content Management System navigation.

Step 4: Ask trainees to build dynamic CMS and Monitor the activity.

Step 5: Verify the completeness and functionality of their work, offering feedback and assistance as needed.

Step 6: Provide feedback and support where necessary.

Step 7: Encourage trainees to refer to Key Reading 3.2.1 for any clarifications or guidance.



Points to Remember

- Design a database schema that supports hierarchical relationships for menu items to ensure flexible and dynamic navigation.

- Implement functions to fetch and display menu items from the database, ensuring that the navigation structure updates automatically as content changes.
- Use CSS and JavaScript to enhance the navigation's appearance and interactivity for a better user experience.
- Ensure the navigation system is tested on various devices and screen sizes to guarantee responsiveness and usability.
- Implement caching strategies if necessary to optimize performance, especially for large or complex navigation structures.



Application of learning 3.2.

NyundoTech, a growing tech company based in Rubavu District, Nyundo Sector, specializes in creating interactive websites for local businesses. They are currently developing a website for a regional tourism board that requires a dynamic content navigation system to showcase various tourist attractions. The navigation must be intuitive and dynamically update based on the content added by the administrators. You have been contracted by NyundoTech to build and implement this dynamic content navigation system, ensuring that it is user-friendly and capable of handling a large and varied content database.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	Dynamic Content Navigation is properly built	1.1 Database is created		
		1.2 Tables is created		
		1.3 Content are inserted		
		1.4 Content are updated		
		1.5 Admin panel is created		
		1.6 The navigation is user friendly		
		1.7 The system can handle large content		



Indicative content 3.3: Management of Cookies and Sessions



Duration: 3 hrs



Practical Activity 3.3.1: Managing Cookies and Sessions



Notes to the trainer

- Computers must be well functioning and XAMPP installed.
- The trainees can resume from previously built CMS projects
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

Go to the computer lab and use the previous built CMS, then implement user authentication and session management, including testing login, cookies, and logout functionality as outlined in Key Reading 3.3.1.

Step 3: Clearly explain the task and provide comprehensive work instructions.

Step 4: Demonstrate how to manage cookies and sessions in CMS. Explain each step of the process during the demonstration.

Step 5: Ask trainees to implement user authentication and apply cookies and session management

Step 6: Monitor trainees as they implement the task, providing feedback and assistance where needed.

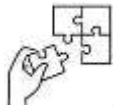
Step 7: Verify the completeness and functionality of their work, ensuring it aligns with Key Reading 3.3.1.

Step 8: Encourage trainees to refer to Key Reading 3.3.1 for additional guidance.



Points to Remember

- Ensure that cookies are securely handled to protect user information.
- Always test session management to confirm that user data is properly maintained and cleared upon logout.
- Implement robust authentication mechanisms to prevent unauthorized access to the system.



Application of learning 3.3.

KigaliSmart Solutions, a tech company located in Kicukiro District, Kigali City, specializes in developing secure e-commerce platforms. They are currently working on a new project that involves user-specific experiences, requiring the effective management of cookies and sessions to maintain user preferences, manage logins, and secure sensitive data. As an expert in PHP development, you have been hired by KigaliSmart Solutions to implement robust cookie and session management features that ensure seamless user interactions while maintaining high security standards across their e-commerce platform.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	User Authentication is correctly performed	1.1 User accounts are created		
		1.2 Users are logged in		
2	Cookies and Sessions are properly managed	2.1 Sessions are set		
		2.2 Sessions are destroyed		
		2.3 Users are logged out		
		2.4 Logged out account content can't be accessed		
		2.5 Cookies are created		
		2.6 Cookies are unset		



Indicative content 3.4: Application of Context and Options



Duration: 3hrs



Practical Activity 3.4.1: Applying the CMS concepts of Context and Options



Notes to the trainer

- The computer lab should be available with fully functioning computers and XAMPP installed.
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

Go to the computer lab and update the database schema to include visibility and hidden status fields. Implement PHP scripts to manage content visibility, hidden subjects, and user options. Test all features to ensure functionality and security as outlined in Key Reading 3.4.1.

Step 2: Clearly explain the task and provide adequate work instructions.

Step 3: Demonstrate how to apply CMS concepts of context and options. Explain each step during the demonstration.

Step 4: Ask trainees to apply CMS concepts of context and options.

Step 5: Monitor trainees as they implement the task. Provide feedback and assistance where needed.

Step 6: Verify the completeness and functionality of their work, ensuring it aligns with Key Reading 3.4.1.

Step 7: Encourage trainees to refer to Key Reading 3.4.1 for additional guidance.



Points to Remember

- Clearly define the context of your CMS by specifying the user roles, content types, and workflows to ensure the system meets the intended requirements.
- Configure the CMS options effectively to align with the context, such as setting up permissions, content visibility, and customization options.

- Regularly review and update context and options settings as your CMS evolves to accommodate changes in user needs and business goals.
- Test all context and options configurations thoroughly to ensure they function as expected and do not introduce any security or usability issues.
- Document the context and options settings clearly for future reference and maintenance, making it easier for team members to understand and manage the CMS.



Application of learning 3.4.

NyungweTech, a software firm located in Rusizi District, Western Province, focuses on developing educational platforms for schools across Rwanda. They are currently upgrading their Content Management System (CMS) to provide schools with customizable options for content visibility and user roles. This upgrade includes managing public content, hidden pages, and user-specific options while ensuring the platform's security. As a developer with expertise in CMS, you have been brought on board to implement features that allow administrators to control content visibility, manage hidden subjects, apply conditional logic based on user roles, and enhance the platform's security measures against unauthorized access.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	Context and Options are properly applied	1.1 User accounts are created		
		1.2 Admin panel is created		
		1.3 Content is created		
		1.4 Content can be hidden		
		1.5 User roles are applied		
		1.6 No unauthorised access of user's content		



Indicative content 3.5: Regulate Page Access



Duration: 4 hrs



Practical Activity 3.5.1: Regulating page access within CMS



Notes to the trainer

- The computer lab should be available with fully functioning computers and XAMPP installed.
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

You are requested to go to the computer lab, set up the admins table, create a login form and admin dashboard, and implement user authentication, secure passwords, access control, and optional updates as detailed in Key Reading 3.5.1. This task should be done individually.

Step 2: Demonstrate how to regulate page access within the CMS, including setting up the admins table, creating the login form and admin dashboard, and implementing user authentication and access control.

Step 3: Ask the trainees to implement the regulation of page access within the CMS and monitor the process.

Step 4: Verify the completeness of the page access regulation, including the correct setup of user authentication, secure passwords, and access control.

Step 5: Provide Expert feedback where necessary.

Step 6: Ask trainees to read Key Reading 3.5.1 in the trainee manual.



Points to Remember

- Implement user authentication mechanisms to ensure that only authorized users can access or modify certain pages.
- Define and manage user roles and permissions accurately to control access levels and functionalities based on user needs and responsibilities.
- Use access control lists (ACLs) or similar methods to enforce page-specific access rules and prevent unauthorized access.
- Regularly audit and review access controls to ensure they align with current security policies and organizational requirements.
- Document and communicate access control policies clearly to all stakeholders to ensure proper understanding and adherence.



Application of learning 3.5.

TechNet Solutions, based in Kigali City, specializes in providing IT solutions and support for local businesses. They are enhancing their CMS platform to include more robust user management and access control features. As a developer hired by TechNet Solutions, you are tasked with implementing a system that regulates page access based on user roles. This involves setting up user authentication, creating an admin management table, building an admin dashboard, and ensuring secure password management. Your role includes configuring access control to differentiate between user and admin functionalities, and integrating features for content preview, user role-based permissions, and logout capabilities to maintain security and efficiency across their CMS platform.

Checklist:

S N	Criteria	Indicators	Observation	
			Yes	No
1	Page access is correctly regulated	1.1Admin panel is created		
		1.2Users are authenticated		
		1.3Roles are assigned to Users		
		1.4Admin dashboard is built		
		1.5Access controls are configured		
		1.6 Features are integrated		
		1.7 Logout is working		
		1.8 Content of logged out account cannot be accessed		



Indicative content 3.6: CMS Errors Detection



Duration: 3 hrs



Theoretical Activity 3.6.1: Description of errors in CMS



Notes to the trainer:

- Use small groups to describe the CMS errors.



Key steps:

While delivering this activity, pass through the following steps:

Step 1. Introduce the activity and request trainees to answer the following questions related to detecting and testing errors in a CMS:

- i. Define the following types of CMS errors:
 - a. Syntax Errors
 - b. Runtime Errors
 - c. Validation Errors
 - d. Database Errors
 - e. Security Errors
 - f. Permission Errors
 - g. 404 Errors
 - h. 500 Internal Server Errors
- ii. Describe the implications of each error type on CMS functionality and user experience.
- iii. Explain practical methods for detecting and handling each type of error in a CMS.
- iv. Discuss various testing approaches for ensuring CMS stability and performance, including:
 - a. Unit Testing
 - b. Integration Testing
 - c. Functional Testing
 - d. Usability Testing
 - e. Security Testing
 - f. Performance Testing
 - g. Load Testing
 - h. Regression Testing

Step 2. Ask trainees to provide the answers for the questions and write them on papers.

Step 3. Invite trainees to present their findings/answers to the whole class.

Step 4. Provide expert views and clarifications based on the presentations, using didactic materials to support explanations.

Step 5. Address any questions or concerns raised during the presentations and discussions.

Step 6. Ask trainees to read Key Reading 3.6.1 in their manuals for additional clarification and guidance.



Points to Remember

- Regularly perform error detection and testing using debugging tools and error logs to identify and resolve issues promptly.
- Implement thorough testing procedures, including unit tests and integration tests, to ensure that all components of the CMS function correctly.
- Keep error handling mechanisms strong by implementing proper error messages and logging to facilitate easier troubleshooting.
- Ensure that error detection and testing are part of the development lifecycle to catch issues early and maintain system stability.
- Document common errors and their resolutions to create a reference guide for future troubleshooting and to improve system reliability.



Application of learning 3.6.

KigaliTech Innovations, located in Kigali, Rwanda, is a company specializing in custom software solutions for local businesses. They are in the process of refining their CMS platform to improve reliability and user experience. As a newly hired software developer at KigaliTech Innovations, you are responsible for implementing a comprehensive error detection and testing strategy for their CMS. This involves identifying and addressing various types of CMS errors, such as syntax, runtime, validation, and database errors. You will also set up systematic testing approaches, including unit testing, integration testing, and performance testing. Your task is to ensure that the CMS operates smoothly by detecting potential errors, applying fixes, and documenting the testing process to enhance the system's stability and performance.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	CMS errors are correctly detected and fixed	1.1 Testing strategies are applied		
		1.2 Errors are identified		
		1.3 Systematic error testing approaches are set		
		1.4 Fixes are applied		
		1.5 Testing process is documented		



Indicative content 3.7: Maintain CMS



Duration: 4 hrs



Theoretical Activity 3.7.1: Explanation on CMS Maintenance



Notes to the trainer:

- Use small groups to explain CMS Maintenance



Key steps:

While delivering this activity, pass through the following steps:

Step 1. Introduce the activity and request trainees to address the following aspects related to maintaining a CMS:

- Define the importance of regular CMS updates, including core, plugins, and themes.
- Discuss best practices for performing regular updates, including using a staging environment before applying updates to the live system.
- Explain the significance of regularly updating plugins and modules, and describe how to ensure their compatibility and functionality.
- Describe the role of regular backups in CMS maintenance and outline the best practices for implementing automated backups, including testing the restoration process.
- Discuss the importance of database optimization and detail the tasks involved in optimizing a CMS database, such as cleaning up unused data and indexing frequently queried fields.
- Explain the importance of security measures in CMS maintenance, including updating software, implementing strong authentication methods, and regularly scanning for vulnerabilities.
- Discuss the significance of performance monitoring in CMS maintenance and describe the tools and methods used to track key metrics and optimize system resources.

Step 2. Ask trainees to write down their group's answers and findings on paper.

Step 3. Invite trainees to present their findings to the class.

Step 4. Provide expert views and clarifications based on the presentations, using didactic materials to support explanations.

Step 5. Address any questions or concerns raised during the presentations and discussions.

Step 6. Ask trainees to read Key Reading 3.7.1 for additional clarification and guidance.



Points to Remember

- Regularly update your CMS to the latest version to ensure security patches and new features are applied.
- Back up your CMS database and files before making any significant changes or updates.
- Monitor the performance of your CMS regularly to identify and address any slowdowns or errors early.
- Keep your CMS plugins and themes updated to prevent compatibility issues and security vulnerabilities.
- Document all maintenance activities, including updates, backups, and troubleshooting steps, for future reference.



Practical Activity 3.7.2: Performing CMS Maintenance



Notes to the trainer

- The computer lab should be available with fully functioning computers with a working XAMPP
- Make sure that the computers in the computer lab has internet access
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

You are requested to go to the computer lab and perform CMS maintenance tasks, including updates, backups, optimization, security checks, and performance monitoring. This task should be done individually.

Step 2: Demonstrate how to perform CMS maintenance, including updating, backing up, optimizing, securing, and monitoring the CMS.

Step 3: Ask the trainees to perform the CMS maintenance tasks and monitor the process.

Step 4: Verify that the CMS is properly maintained, including successful updates, backups, and optimizations.

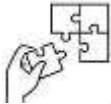
Step 5: Provide Expert feedback where necessary.

Step 6: Ask trainees to read Key Reading 3.7.2 in the trainee manual.



Points to Remember

- Schedule regular maintenance to prevent unexpected downtimes and ensure smooth operation of your CMS.
- Check for and install updates for the CMS core, plugins, and themes to keep everything secure and functional.
- Regularly back up your CMS and database to avoid data loss in case of failures or attacks.
- Optimize the database by cleaning up unused data and running maintenance tasks to improve performance.
- Monitor security logs and system performance to quickly detect and respond to potential issues.



Application of learning 3.7.

TechGuru Solutions, based in Butare, Rwanda, is a rapidly growing company that provides digital solutions and CMS development services. They are committed to maintaining their CMS platform to ensure it remains secure, efficient, and up-to-date. As a newly appointed CMS maintenance specialist at TechGuru Solutions, your role involves implementing a robust maintenance plan. This includes performing regular updates to the CMS core, themes, and plugins, setting up automated backup solutions, optimizing database performance, and enhancing security measures. You will also monitor the system's performance and keep detailed logs of maintenance activities. Your objective is to ensure that the CMS operates at peak performance, with minimal downtime and maximum reliability, while documenting and addressing any issues encountered during maintenance.

Checklist

SN	Elements	Indicators	Observation	
			Yes	No
1	CMS is properly maintained	1.1 Maintenance plan is developed		
		1.2 Regular update are planned		
		1.3 Automated backup are set		
		1.4 Database performance is optimized		
		1.5 Security measures are available		
		1.6 System performance is monitored		
		1.7 Maintenance activities logs are available		



Learning outcome 3 end assessment

Theoretical assessment

I. Read the following statement related PHP programming and choose the correct letter that corresponding to the correct

1. What should be the first step when implementing a logout feature in PHP?

- a) Redirect the user to the login page
- b) Destroy the session
- c) Unset all cookies
- d) Display a logout confirmation message

Answer: b) Destroy the session

2. What is the function of the **require_login()** function in a CMS?

- a) To check if a user is logged in
- b) To update user passwords
- c) To log out a user
- d) To delete a user account

Answer: a) To check if a user is logged in

II. Read the following statement related to php programming and complete with the appropriate terminology among: _start, password_hash

1. In PHP, the function used to start a session is _____.

Answer: **session_start()**

2. The PHP function _____ is used to create a secure hash of a password.

Answer: **password_hash()**

III. Read the following statement related PHP programming and answer by True if the statement is correct and False if the statement is wrong

1. Sessions in a CMS can be used to maintain the logged-in state of a user across multiple pages.

Answer: **True**

2. Sanitizing user inputs is unnecessary if your CMS has user authentication.

Answer: **False**

3. The **password_verify()** function in PHP is used to compare a user's entered password with the hashed password stored in the database.

Answer: **True**

4. Cookies can be used to store user preferences like themes or login details, enhancing the user experience in a CMS.

Answer: **True**

5. To create a new admin account in a CMS, you should store the password in plain text for easy retrieval.

Answer: **False**

IV. Read the following statement related PHP programming and write the letter related to the correct significance

Answer	Error testing approaches	Significances
.....h.....	1.Regression Testing	a. Testing individual components of the CMS to ensure they function as expected. This includes testing functions, classes, and methods.
.....g....	2.Load Testing	b. Ensuring that different components of the CMS work together seamlessly. This includes testing interactions between the CMS and the database, external APIs, and other systems.
....c...	3.Functional Testing	c. Verifying that the CMS meets the specified requirements by testing its features and functionalities.
....f....	4.Performance Testing	d. Ensuring that the CMS is user-friendly and that users can interact with it as intended without confusion or difficulty.
....b....	5.Integration Testing	e. Identifying vulnerabilities in the CMS that could be exploited by attackers. This includes testing for common security issues like SQL injection, XSS, and CSRF.
.....e....	6.Security Testing	f. Assessing how the CMS performs under various conditions, including high traffic or large data sets.
....a....	7.Unit Testing	g. Evaluating how the CMS handles multiple users accessing it simultaneously.
.....d....	8. Usability Testing	h. Re-running previous tests after updates or changes to the CMS to ensure that new code has not introduced new errors.

Practical assessment

Visionary Web Solutions, a Kigali-based web development agency, is focused on creating tailored digital solutions for various clients. They have been tasked with building a custom Content Management System (CMS) for a local educational institution that wants to manage and update its website content independently. As a newly recruited PHP developer at Visionary Web Solutions, your role is to design and develop this CMS from scratch using PHP.

This includes setting up the CMS architecture, developing features to add, edit, and delete content, and creating a user authentication system to control access to the CMS. You'll need to ensure the system is user-friendly, secure, and scalable. Implement dynamic content navigation to allow administrators to easily manage subjects and pages. Additionally, you must integrate a database using MySQL, develop the backend logic for managing content, and design the frontend interface for user interactions. After building the CMS, thoroughly test it for functionality, security, and usability, documenting your development process and preparing a detailed report for client review. Your objective is to deliver a fully functional CMS that meets the client's requirements and allows for seamless content management.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	CMS environment is properly set up	1.1 CMS environment is prepared		
		1.2 CMS environment is configured		
		1.3 Server requirements are checked		
		1.4 CMS software is installed		
		1.5 CMS environment is operational		
2	Dynamic Content Navigation is correctly built	2.1 Database is created		
		2.2 Tables is created		
		2.3 Content are inserted		
		2.4 Content are updated		
		2.5 Admin panel is created		
		2.6 The navigation is user friendly		
		2.7 The system can handle large content		
3	Content, Cookies and Sessions are effectively managed	3.1 User accounts are created		
		3.2 Users are logged in		
		3.3.Sessions are set		
		3.4 Sessions are destroyed		
		3.5 Users are logged out		
		3.6 Logged out account content can't be accessed		
		3.7 Cookies are created		
		3.8 Cookies are unset		
		3.9 Admin panel is created		
		3.10 Content is created		
		3.11 Content can be hidden		
		3.12 User roles are applied		
		3.13 No unauthorised access of user's content		
4	CMS is properly maintained	4.1 Maintenance plan is developed		

		4.2 Regular update are planned		
		4.3 Automated backup are set		
		4.4 Database performance is optimized		
		4.5 Security measures are available		
		4.6 System performance is monitored		
		4.7 Maintenance activities logs are available		

END



Further information to the trainer

Bierer, D. (2016). *PHP 7 programming book*. Birmingham, B32PB, UK: Pack Publishing Ltd.

Bradley, M. (2015). *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5*. O'Reilly Media.

Castledine, E., & Zandstra, M. (2020). *PHP & MySQL: Novice to Ninja* (7th ed.). SitePoint.

Castro, E. (2016). *PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide*. Peachpit Press.

Jesus Castagnetto, H. R. (1999). *Professional PHP Programming*. Birmingham: Wrox Press Ltd.

Lavender, M., & Petty, R. (2021). *PHP & MySQL: Server-side Web Development*. Pearson.

Nixon, R. (2014). *Learning PHP, MySQL, JavaScript, CSS & HTML* (3rd ed.). USA: O'Reilly Media.

Nixon, R. (2018). *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5* (5th ed.). O'Reilly Media.

Popel, D. (2007). *Learning data object*. Luna Park, Sydney: Pack Publisher.

Powers, D. (2018). *PHP 7 Solutions: Dynamic Web Design Made Easy* (4th ed.). Apress.

Robinson, K. (2017). *PHP and MySQL Web Development*. Addison-Wesley.

Schlossnagle, G. (2017). *Advanced PHP Programming* (2nd ed.). Addison-Wesley Professional.

Welling, L., & Thomson, L. (2016). *PHP and MySQL web development* (5th ed.). Addison-Wesley.

PHP.net. (2023). *PHP Manual*. Retrieved from <https://www.PHP.net/manual/en/>

W3Schools. (2023). *PHP Tutorial*. Retrieved from <https://www.w3schools.com/PHP>

Javatpoint. (2023, May 4). *PHP Tutorial*. Retrieved from <https://www.javatpoint.com>

Learning Outcome 4: Build a Web App Using MVC Framework (LARAVEL)



Indicative contents

- 4.1. Framework environment configuration
- 4.2. Setup Laravel custom routing
- 4.3. Perform form data validation
- 4.4. Perform CRUD Operations
- 4.5. Manage APIs in Laravel frameworks
- 4.6. Authentication and Security
- 4.7. API Versioning and Documentation

Key Competencies for Learning Outcome 4: Build a web app using MVC Framework (LARAVEL).

Knowledge	Skills	Attitudes
● Description of PHP frameworks ● Description of MVC Architecture ● Description of Laravel environment ● Explanation of APIs in Laravel Framework.	● Setting up Laravel framework environment ● Setting Laravel custom routing ● Performing Form Data validation ● Managing APIs in Laravel framework ● Performing authentication and security ● Managing APIs source codes with version control systems	● Having teamwork spirit ● Being flexible ● Having Time management ability ● Having curiosity ● Being creative ● Being innovative ● Being adaptive



Duration: 28 hrs



Learning outcome 4 objectives:

By the end of the learning outcome, the trainees will be able to:

1. Configure properly Laravel Framework environment based on its standards.
2. Set effectively custom routing in accordance with Laravel Framework standards.
3. Validate properly Form Data based on Laravel Framework standards.
4. Perform effectively CRUD Operations in accordance with Laravel Framework standards.
5. Manage correctly APIs in regards to Web Application standards.
6. Secure effectively Web Application based on framework security standards.
7. Manage properly source code changes according to the version control system standards.



Resources

Equipment	Tools	Materials
• Computer	• Text Editor • IDE • Browsers • XAMPP/WAMP • Laravel framework	• Internet • Electricity



Advance Preparation:

Before delivering this learning outcome, you are recommended to:

- Have didactic materials
- Avail functioning computers connected to the internet



Indicative content 4.1: Framework Environment Configuration



Duration: 4 hrs



Theoretical Activity 4.1.1: Introduction on PHP Frameworks



Notes to the trainer:

- Use small groups for trainees to discuss PHP frameworks.



Key steps:

While delivering this activity, pass through the following steps:

Step 1. Introduce the activity and request trainees to answer to the following questions:

- What do you understand by a PHP Framework?
- Briefly explain how PHP Frameworks work.
- State the reason why we would need to use a PHP framework in our Web Application development process.
- Outline the benefits of using a PHP Framework.
- After giving different examples of PHP Frameworks, differentiate them according to their characteristics.
- What are the advantages and disadvantages of the following PHP Frameworks?
 - Laravel
 - Symfony
 - Yii
 - CakePHP
 - CodeIgniter
 - Lumen
- Describe the Laravel MVC Architecture (Model, View, Controller).

Step 2. Ask trainees to discuss the provided tasks and select the correct answers within their groups.

Step 3. Ask a trainee to write the group's answers on the flipchart/paper.

Step 4. Ask the trainees to present their findings to the whole class

Step 5. Provide an expert view and clarify the ideas by using relevant didactic materials to ensure thorough understanding.

Step 6. Ask trainees to read Key Reading 4.1.1 in the trainee manual for additional clarification and encourage them to ask questions where necessary.



Points to Remember

- Understand the core purpose of PHP frameworks: they streamline development by providing a structured way to build applications efficiently.
- Familiarize yourself with popular PHP frameworks and their key features to choose the one that best fits your project needs.
- Learn the fundamental components of PHP frameworks, such as MVC architecture, routing, and ORM, which help in organizing and managing code effectively.
- Keep in mind that using a framework can significantly reduce development time and effort by offering reusable components and built-in functionality.
- Ensure that you understand the framework's documentation and community support, as they are crucial for troubleshooting and leveraging advanced features.



Theoretical Activity 4.1.2: Description of Laravel MVC Architecture (Model, View, Controller)



Notes to the trainer:

- Use small groups for describing Laravel MVC architecture.



Key steps:

While delivering this activity, pass through the following steps:

Step 1. Introduce the activity and request trainees to answer to the following questions:

- i. Why is the use of the MVC architectural pattern important in the Laravel framework?
- ii. Differentiate the three components of the MVC architectural pattern as used within the Laravel framework.
- iii. Briefly describe the flow of an MVC application in the Laravel framework.
- iv. What are the advantages of using MVC in Laravel?

Step 2. Ask trainees to discuss the provided tasks and select the correct answers within their groups.

Step 3. Ask trainees to write the group's answers on the flipchart/paper.

Step 4. Invite the trainees to present their findings and facilitate a discussion around their responses.

Step 5. Provide your expert view and clarify the ideas by using relevant didactic materials.

Step 6. Ask trainees to read Key Reading 4.1.2 in the trainee manual for additional clarification and encourage them to ask questions where necessary.



Points to Remember

- The Model represents the data and business logic of the application; ensure models handle data interactions and validation properly.
- The View is responsible for presenting data to the user; make sure views are designed to be clean and separate from the application logic.
- The Controller manages the flow of the application; it should process user input, interact with models, and return responses to the view.
- Adhere to the MVC pattern to maintain a clear separation of concerns, which improves code maintainability and scalability.
- Utilize Laravel's built-in features to facilitate the MVC pattern, such as Eloquent ORM for models and Blade templating for views.



Practical Activity 4.1.3: Installing Laravel Framework



Notes to the trainer

- The computer lab should be available with fully functioning computers
- Avail internet connectivity to all computers
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:
Go to the computer lab, install and configure the Laravel Framework using Composer.
This task should be done individually.

Step 2: Demonstrate the steps to install the Laravel framework on a computer using Composer.

Step 3: ask trainees to install and configure the Laravel Framework using Composer.

Step 4: Verify the installation process and results to ensure completeness and functionality.

Step 5: Provide expert guidance and assistance where necessary.

Step 6: Ask trainees to read Key Reading 4.1.3 in their manuals for additional clarifications.



Points to Remember

- Ensure your development environment meets Laravel's requirements, including PHP version and necessary extensions, before starting the installation.
- Use Composer to install Laravel by running **composer create-project -prefer-dist laravel/laravel projectName** to set up a new Laravel project.
- Configure your **.env** file with appropriate database and environment settings immediately after installation to ensure smooth operation.
- Run **PHP artisan key:generate** to generate a unique application key, which is crucial for securing session and encryption data.
- Verify the installation by running **PHP artisan serve** and accessing the application in a web browser to confirm that Laravel is set up correctly.
- Most PHP frameworks follow a design pattern called MVC (Model-View-Controller). This pattern helps organize your code in a way that separates different parts of your application.
- Laravel utilizes Composer to manage its dependencies. It must be installed before setting up Laravel.
- The **.env** file is a powerful tool in Laravel that allows you to manage your application's configuration in a flexible and secure manner.
- Blade allows you to echo data, use directives, and create clean and readable templates.



Practical Activity 4.1.4: Configuring the Laravel .env file



Notes to the trainer

- The computer lab should be available with fully functioning computers
- Trainees can resume from the project of the previous practical activity
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

Go to the computer lab, configure the Laravel `.env` file by setting up your database connection, app key, and environment settings, and then verify that the configuration is working as outlined in Key Reading 4.1.4.

Step 2: Demonstrate how to configure the Laravel `.env` file, explaining each step of the process.

Step 3: Ask trainees to configure the `.env` file and verify their settings using Laravel's `env()` function.

Step 4: Provide expert guidance and support where necessary.

Step 5: Ask trainees to review Key Reading 4.1.4 for further clarification and encourage them to ask questions if needed.



Points to Remember

- Ensure all sensitive information such as database credentials and application keys are stored securely in the `.env` file and not in the codebase.
- Use environment-specific configurations to separate development, testing, and production settings, ensuring that sensitive data does not leak between environments.
- Always run **PHP artisan config: cache** after making changes to the `.env` file to update the configuration cache and ensure changes are applied.
- Avoid committing the `.env` file to version control; use `.gitignore` to keep it excluded from the repository to protect sensitive information.
- Regularly review and update the `.env` settings to match the current environment needs and security practices.



Practical Activity 4.1.5: Using blade templates for Laravel



Notes to the trainer

- The computer lab should be available with fully functioning computers
- Trainees can resume from the project of the previous practical activity
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

- Step 1:** Introduce the activity and ask the trainees to access the computer lab, apply Blade templates to their Laravel projects individually.
- Step 2:** Demonstrate the use of Blade templates in Laravel, clearly explaining each step of the process.
- Step 3:** Guide trainees in applying Blade templates to their Laravel projects, following the demonstrated steps.
- Step 4:** Provide expert feedback and assistance as needed.
- Step 5:** Encourage trainees to review Key Reading 4.1.5 for further details and offer support for any questions or clarifications.



Points to Remember

- Ensure all sensitive information such as database credentials and application keys are stored securely in the **.env** file and not in the codebase.
- Use environment-specific configurations to separate development, testing, and production settings, ensuring that sensitive data does not leak between environments.
- Always run **PHP artisan config:cache** after making changes to the **.env** file to update the configuration cache and ensure changes are applied.
- Avoid committing the **.env** file to version control; use **.gitignore** to keep it excluded from the repository to protect sensitive information.

Regularly review and update the **.env** settings to match the current environment needs and security practices.



Application of learning 4.1.

Takarata is a software development company that focuses its work on Web Development and it is located in Gisenyi, Western Province. They have been using other framework but now are committed to Laravel. They need to hire someone who is skilled in working with Laravel framework.

You are requested by Takarata to install and configure Laravel Framework in their computers so that they can use it in their daily job.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	Laravel framework environment is fully set up	1. 1 Composer is downloaded		
		1.2 Laravel is installed		
		1.3Laravel .env file is configured		
		1.4Laravel is correctly launched		



Indicative content 4.2: Setup Laravel Custom Routing



Duration: 4 hrs



Theoretical Activity 4.2.1: Description of Laravel custom routing



Notes to the trainer:

- Use small groups for describing custom routing.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following tasks:

In small groups, you are requested to answer the following questions:

- i. What is the role of routing in Laravel, and how do web and API routing differ?
- ii. How are routes defined in Laravel, and what are the different HTTP methods you can use?
- iii. What are required and optional parameters in routes, and how can you apply constraints to them?
- iv. What is the purpose of named routes, and how can they be utilized in Laravel applications?
- v. What is middleware, and what are the differences between global and route-specific middleware? Provide examples of common middleware.
- vi. How do route groups simplify route management, and what are some ways to group routes by prefix or middleware?

Step 2: Allow trainees to present their findings and examples to the class.

Step 3: Provide expert views and clarification where necessary.

Step 4: Ask trainees to read Key Reading 4.2.1.



Points to Remember

- Define custom routes in the **routes/web.PHP** or **routes/api.PHP** file, specifying the URL path and corresponding controller method.
- Use route parameters in your URL paths to capture dynamic segments, and validate these parameters using route constraints.
- Group routes with common middleware or prefixes to streamline route management and apply consistent behaviors across multiple routes.
- Test custom routes thoroughly to ensure they resolve correctly and handle requests as expected, including edge cases and error scenarios.



Practical Activity 4.2.2: Setting Up Custom Routing in Laravel



Notes to the trainer

- The computer lab should be available with fully functioning computers with composer installed.
- The computers should have access to the internet
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and Ask the trainees to set up custom routes in Laravel by defining route parameters, names, and middleware, ensuring they align with the application's logic as detailed in Key Reading 4.2.2.

Step 2: Demonstrate how to create a new route in Laravel and define a route that uses a controller. Ensure that both a web route and an API route are set up in the Laravel application.

Step 3: Ask trainees to perform the task provided on step1.

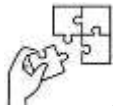
Step 4: Provide expert feedback and assistance as needed.

Step 5: Instruct trainees to review Key Reading 4.2.2.



Points to Remember

- Create custom routes by defining them in **routes/web.PHP** or **routes/api.PHP**, specifying the URL and associated controller or closure.
- Utilize route parameters to capture dynamic data from the URL and validate them using route constraints to ensure proper data format.
- Apply route middleware to enforce authentication, authorization, or other filters on your routes for added security and functionality.
- Group related routes together with common prefixes or middleware to maintain organized and efficient route management.
- Verify that all custom routes function as intended by thoroughly testing them, including edge cases and error handling scenarios



Application of learning 4.2.

RukomaTech Ltd, a software development company located in Muhanga District, Southern Province, specializes in building customized web applications for local businesses. Recently, they decided to enhance their service offerings by creating a web application that requires complex and highly customized routing. To achieve this, they need an expert in Laravel who can set up and configure custom routing that aligns with their specific project requirements. You have been hired by RukomaTech Ltd to implement and configure Laravel custom routing on their development servers to ensure the project meets the desired specifications.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	Custom routing are properly set up and configured	1.1 Laravel environment is set		
		1.2 MVC Laravel project is built		
		1.3 Routes are created		
		1.4 Routes are configured		
		1.5 Routes are grouped		



Indicative content 4.3: Perform Form Data Validation



Duration: 4 hrs



Theoretical Activity 4.3.1. Elaboration of data validation in Laravel



Notes to the trainer:

- Use small groups for describing data validation in Laravel.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to read Key Reading 4.3.1 to understand the essentials of data validation in Laravel and answer the following questions.

- Explain why data validation is crucial for:
 - Security: Preventing malicious input.
 - Data Integrity: Ensuring clean, consistent data.
 - User Experience: Guiding users to submit correct information.
- Describe how Laravel validates CSRF tokens and the importance of this process.
- Discuss validation for different form elements and why specific rules are necessary.
- Outline basic validation in Laravel controllers using `$request->validate()` and provide examples.
- Briefly discuss the following advanced validation concepts:
 - Conditional Rules
 - Custom Validation
 - Array Validation
- Explain how Laravel handles validation errors within these applications:
 - Redirection
 - Displaying Errors
 - Flash Messages

Step 2: Ask the trainees to document their understanding with summaries, examples, and any questions they have about the validation concepts discussed.

Step 3: Invite the trainees to present their findings to the class. Encourage them to revise their work based on feedback and class discussion.

Step 4: Provide an expert view by explaining complex or nuanced aspects of validation and offer additional insights or best practices not covered in the reading.

Step 5: For further clarification, ask the trainees to review Key Reading 4.3.1 again and encourage them to ask questions if they need additional information.



Points to Remember

- Use Laravel's built-in validation rules to ensure data integrity and enforce constraints on user inputs.
- Customize validation messages to provide clear and user-friendly feedback for validation errors.
- Implement server-side validation in controllers to protect against invalid data and potential security issues.
- Use form request classes for complex validation logic to keep your controllers clean and focused.
- Test validation thoroughly to handle various scenarios and edge cases, ensuring that your application remains robust and reliable.



Practical Activity 4.3.2: Validating form data in Laravel



Notes to the trainer

- The computers to use should be fully functioning with composer installed.
- The computers should have access to the internet
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:
You are requested to go to the computer lab and work on validating form data in Laravel. This task should be done individually.

Step 2: Demonstrate how to create a form in Laravel with CSRF protection using `@csrf` and how to implement server-side validation in the controller. Ask the trainees to create and validate a form in Laravel, monitoring their process

Step 3: Ask the trainees to create and validate a form in Laravel, monitoring their process.

Step 4: Verify that the form validation is correctly implemented, including customization of error messages and incorporation of client-side validation if applicable.

Step 5: Provide your feedback where necessary

Step 6: Ask trainees to read Key Reading 4.3.2 in the trainee manual.



Points to Remember

- Create a form with appropriate HTML elements and CSRF protection to prevent cross-site request forgery attacks.
- Implement server-side validation in your controller or form request class by defining validation rules and custom messages as needed.
- Add client-side validation using JavaScript or Laravel's built-in tools to enhance user experience and provide immediate feedback.
- FTest the form thoroughly to ensure that validation rules are applied correctly and that error messages are displayed to users as intended.
- Review and update validation rules periodically to accommodate changes in form requirements and ensure data integrity.



Application of learning 4.3.

Kigali Digital Solutions, based in Kicukiro District, Kigali City, is a company known for developing innovative web platforms for clients across various industries. They are currently working on a new client project that requires secure and accurate form submissions for user registration, contact forms, and feedback. To ensure data integrity and prevent malicious inputs, Kigali Digital Solutions needs a skilled Laravel developer to perform robust form data validation. You have been brought on board to implement and configure form data validation in their Laravel project, ensuring that all user inputs are properly validated before being processed or stored in the database.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	Laravel form data are well validated	1.1 CSRF Token is used on forms		
		1.2 Server-side validation is applied		
		1.3 Error messages are customised		
		1.4 Flash messages are used		



Indicative content 4.4: Perform CRUD Operations



Duration: 4 hrs



Theoretical Activity 4.4.1: Description of CRUD Operations in Laravel



Notes to the trainer:

- Use small groups for describing CRUD operations in Laravel.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following tasks:

In small groups, answer the following questions to understanding CRUD operations in Laravel.

- Describe how Laravel controllers manage CRUD operations, including:
 - MVC architecture
 - Resource controllers
 - Separation of concerns
- Discuss the purpose of the following models in CRUD operations:
 - Eloquent ORM
 - Defining relationships
 - Mass assignment
- Outline the process and importance of migrations in Laravel, including:
 - Schema Builder
 - Database versioning
 - Rollback and reset
- Summarize the concept of seeding in Laravel, including:
 - Using factory classes
 - Database seeding
 - Best practices
- Explain the role of views in CRUD operations, focusing on:
 - Blade templating engine
 - Separation of logic and presentation
 - Components and layouts
- Discuss CRUD operation routes in Laravel, including:
 - Resource routes
 - Route naming
 - Middleware usage

Step 2: Ask trainees to document their answers and write them on flipchart/paper.

Step 3: Facilitate a class presentation where trainees share their findings and revise their work based on feedback.

Step 4: Provide expert views and address any questions or concerns.

Step 5: Ask trainees to review Key Reading 4.4.1 again for any clarifications or additional details needed.



Points to Remember

- Understand that CRUD operations **Create**, **Read**, **Update**, and **Delete** are fundamental to managing data in any Laravel application.
- Learn how Laravel's Eloquent ORM simplifies CRUD operations by providing a fluent, expressive syntax for interacting with the database.
- Explore Laravel's resource controllers that automate CRUD operations, reducing the amount of boilerplate code needed.
- Recognize the importance of database migrations in managing schema changes and ensuring consistency across different environments.
- Implement validation and authorization checks within CRUD operations to ensure data integrity and secure access control.



Practical Activity 4.4.2: Implementing CRUD Operations in Laravel



Notes to the trainer

- The computer lab should be available with fully functioning computers
- The computers to use must have composer installed and internet access
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and explain that the task is to perform the following:
You are requested to go to the computer lab, implement CRUD operations in Laravel by creating controllers, models, and views for managing records, ensuring proper data handling. This task should be done individually.

Step 2: Ask trainees to set up a new Laravel project and configure their database.

Step 3: Instruct trainees to create a model and migration, then generate a resource controller and define CRUD methods.

Step 4: Request trainees to create views for listing, adding, editing, and viewing resources, and define routes for CRUD operations.

Step 5: Allow trainees to test all CRUD functionalities to ensure they work correctly.

Step 6: Ask trainees to document their process and any issues encountered.

Step 7: Ask trainees to review Key Reading 4.4.2 for additional details or clarifications.



Points to Remember

- Create models, controllers, and views to handle the CRUD operations, ensuring that each component follows Laravel's conventions.
- Define routes for CRUD functionality in the **web.PHP** file, linking them to the appropriate controller methods.
- Use Laravel's built-in validation to check data before it's saved or updated, preventing invalid data from entering the database.
- Test all CRUD operations thoroughly to ensure that data is created, read, updated, and deleted correctly without errors.
- Ensure that user permissions are correctly applied to restrict access to CRUD operations based on user roles.



Application of learning 4.4.

MuhaziTech Ltd, a software company located in Rwamagana District, Eastern Province, specializes in developing custom web applications for small businesses in the region. They are currently developing an inventory management system for a local retail chain. The system requires efficient handling of data, including creating, reading, updating, and deleting (CRUD) operations for products, categories, and suppliers. MuhaziTech Ltd has requested your expertise in implementing these CRUD operations using the Laravel framework to ensure smooth and reliable data management across the system.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	CRUD Operations are properly implemented in Laravel	1.1 Laravel project is initiated		
		1.2 Database is created and configured		
		1.3 Models and migrations are created		
		1.4 Resource controller is generated		
		1.5 Views are created		
		1.6 Routes for CRUD are defined		
		1.7 CRUD functionalities are working		



Indicative content 4.5: Manage APIs in Laravel Frameworks



Duration: 4 hrs



Theoretical Activity 4.5.1: introduction to Laravel's APIs



Notes to the trainer:

- Use small groups for describing APIs in Laravel framework.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and request trainees to answer the following aspects of API development with Laravel:

- Define what an API is and its significance.
- Explain how Laravel supports API development and the tools it provides.
- Describe the principles of RESTful architecture and its application in API design.
- Outline the process of creating a RESTful API in Laravel, including routing, controllers, models, and CRUD operations.
- Detail how to test a Laravel API using Postman.
- Explain how Laravel manages HTTP requests and responses, including request handling and response formatting.
- Describe the role of API resources in Laravel and how they format JSON responses.

Step 2: Ask a trainee to document the responses on a flipchart or paper.

Step 3: Ask trainees to presents their findings.

Step 4: Provide expert insights and clarify concepts using didactic materials.

Step 5: Address any questions or concerns from the trainees.

Step 6: Ask trainees to read Key Reading 4.5.1 for further information



Points to Remember

- Understand that APIs (Application Programming Interfaces) facilitate communication between different software systems, allowing them to interact and exchange data seamlessly.

- Learn about RESTful APIs, which use standard HTTP methods (GET, POST, PUT, DELETE) and are designed around resources and stateless communication.
- Recognize the importance of API endpoints, which define the specific paths through which API requests are made and responses are received.
- Familiarize yourself with API documentation and best practices for designing clear, comprehensive, and user-friendly API specifications.



Practical Activity 4.5.2: Using and Managing APIs in Laravel



Notes to the trainer

- The computer lab should be available with fully functioning computers
- The computers to use must have composer installed and internet access
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and request trainees to perform the following task:

You are requested to go to the computer lab and implement and manage APIs in Laravel by creating routes, controllers, and API resources, ensuring proper request handling and data response as outlined in Key Reading 4.5.2.

Step 2: Demonstrate how to implement and manage APIs in Laravel by creating routes, controllers, and API resources.

Step 3: Ask trainees to implement and manage APIs in Laravel by creating routes, controllers, and API resources

Step 4: Monitor the activity and provide expert feedback where necessary.

Step 5: Ask trainees to review Key Reading 4.5.2 for any additional details or clarifications.



Points to Remember

- Set up a basic API in Laravel by creating routes and controllers to handle API requests and responses effectively.
- Implement CRUD operations for your API endpoints to manage data and ensure your API performs all necessary data manipulations.
- Testing your API endpoints thoroughly using tools like Postman to ensure they are functioning correctly and returning the expected results.

- Document your API endpoints and their functionality to provide clear and useful information for users and developers interacting with the API.



Application of learning 4.5.

KigaliSoft Ltd, a leading tech company based in Nyarugenge District, Kigali City, is expanding its services to include a mobile application that integrates seamlessly with their existing web platform. The mobile app will require robust APIs to manage data synchronization between the web platform and the mobile app. KigaliSoft Ltd has requested your expertise to develop and manage APIs within the Laravel framework, ensuring secure and efficient data exchange between the mobile and web applications.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	APIs are properly used and managed in Laravel	1.1 Laravel project is started		
		1.2 .env file is configured		
		1.3 Model and migration are generated		
		1.4 Postman collection is set		
		1.5 Routes are protected		
		1.6 The project is hosted and tested		



Indicative content 4.6: Authentication and Security



Duration: 4 hrs



Theoretical Activity 4.6.1: Description of APIs authentication and security



Notes to the trainer:

- Use small groups for describing API authentication and security.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and request trainees to discuss and answer the following questions in their groups:

- What is API authentication, and how does it work in Laravel with Sanctum and Passport?
- What are the key concepts of Laravel Sanctum and Passport for API authentication?
- Outline best practices for securing an API, including HTTPS, token expiration, rate limiting, input validation, and CSRF protection.
- Explain the importance of logging, monitoring, API gateways, role-based access control, security audits, security headers, and user education in securing APIs.

Step 2: Ask a trainee to document the group's findings.

Step 3: Request trainees to present their findings to the class.

Step 4: Provide your expert view, offering clarifications and additional insights where necessary.

Step 5: Address any questions or concerns from the trainees.

Step 6: Ask trainees to read Key Reading 4.6.1 for further clarification and understanding.



Points to Remember

- Always implement authentication mechanisms like OAuth, API keys, or tokens to ensure only authorized users can access your API.
- Use HTTPS to encrypt API communications, protecting sensitive data from being intercepted during transmission.

- Regularly update and patch security vulnerabilities in your API to prevent unauthorized access and attacks.
- Apply rate limiting and throttling to prevent abuse and ensure your API remains responsive and secure under heavy usage.



Practical Activity 4.6.2: Authenticating and securing APIs in Laravel



Notes to the trainer

- The computer lab should be available with fully functioning computers
- The computers to use must have composer installed and internet access
- The trainees can resume from previous Laravel project.
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

You are requested to go in the computer lab and implement API authentication and security in Laravel by setting up authentication middleware, securing endpoints, and managing API tokens as outlined in Key Reading 4.6.2.

Step 2: Demonstrate how to authenticate and secure Laravel APIs, while demonstrating explain the steps.

Step 3: Ask trainees to perform authentication and securing APIs in Laravel

Step 4: Assist the trainees with your expert views where necessary.

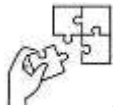
Step 6: Ask trainees to read Key Reading 4.6.2 in their manuals for more clarifications.



Points to Remember

- Integrate Laravel Passport or Sanctum to manage API authentication using tokens, ensuring secure user access.
- Implement middleware to enforce authentication on specific API routes, allowing only authenticated users to access certain endpoints.

- Use Laravel's built-in encryption features to secure sensitive data and protect it from unauthorized access.
- Test your API security by simulating common attacks, such as SQL injection and cross-site scripting, to ensure robust protection.
- Regularly review and update your API security settings to align with the latest best practices and Laravel updates.



Application of learning 4.6.

TechGuard Ltd, a cybersecurity-focused company based in Huye District, Southern Province, is developing a new online platform that will handle sensitive client data. To ensure the highest level of security, TechGuard Ltd requires strong user authentication and security measures. You have been hired by TechGuard Ltd to implement authentication and security features within the Laravel framework, ensuring that only authorized users can access specific areas of the platform and that all data remains secure from potential threats.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	APIs are correctly authenticated and secured	1.1 Laravel sanctum is used		
		1.2 Endpoints are secured		
		1.3 Authentication middleware are set		
		1.4 API tokens are well managed		



Indicative content 4.7: API Versioning and Documentation



Duration: 4 hrs



Theoretical Activity 4.7.1: Describing API versioning and documentation in Laravel



Notes to the trainer:

- Use small groups for describing API versioning and documentation in Laravel.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

In small groups, you are requested to address the key aspects of API versioning and documentation in Laravel and answer the following questions.

- i. Define API versioning and explain its importance in managing API evolution.
- ii. Compare and contrast URL Path Versioning, Query Parameter Versioning, and Header Versioning, discussing their advantages and disadvantages.
- iii. Describe how Laravel implements API versioning, providing examples of route definitions with versioning.
- iv. Explain the role of Swagger and Postman in API documentation, detailing the steps to set up and use these tools for documenting APIs.
- v. Discuss best practices for API documentation, including clarity, consistency, comprehensive coverage, and versioning.

Step 3: Ask any trainee to write the provided answers on a flipchart or paper.

Step 4: Allow the trainees to present their findings to the rest of the class.

Step 5: Provide expert views and clarify ideas using the key concepts.

Step 6: Address any questions or concerns.

Step 7: Ask trainees to read Key Reading 4.7.1 in their manuals for more clarifications.



Points to Remember

- Use versioning in your API URLs (e.g., `/api/v1/`) to manage changes over time without disrupting existing clients.
- Document your API endpoints using tools like Swagger or Postman to ensure developers have clear and accessible references.

- Adopt consistent naming conventions and structure in your API routes to make them intuitive and easier to maintain.
- Provide examples and explanations for each API endpoint in your documentation, including expected inputs and outputs.
- Keep your API documentation up to date with every change or update in your API to prevent discrepancies and confusion.



Practical Activity 4.7.2: Performing API versioning and documentation in Laravel



Notes to the trainer

- The computer lab should be available with fully functioning computers
- The computers to use must have composer installed and internet access
- The trainees can resume from previous Laravel project in order to save time
- Make sure that every trainee will be able to carry out this activity individually.



Key steps:

While delivering this activity, pass through the following steps:

Step 1: Introduce the activity and ask the trainees to perform the following task:

You are requested to go to the computer lab, set up a new Laravel project, define versioned API routes, create controllers for different versions, and document the API using Swagger and Postman as detailed in Key Reading 4.7.2. This task should be done individually.

Step 2: Demonstrate how to perform API versioning in Laravel, including setting up versioned routes, creating controllers, and documenting the API using Swagger and Postman.

Step 3: Ask the trainees to implement API versioning and documentation in their Laravel projects and monitor the process.

Step 4: Verify the completeness of the API versioning and documentation, ensuring the correct setup of versioned routes, controllers, and accurate API documentation

Step 5: Provide your feedback where necessary

Step 6: Ask trainees to read Key Reading 4.7.2 in the trainee manual.



Points to Remember

- When creating new API versions, ensure backward compatibility to avoid breaking existing applications.
- Clearly label each API version in your routes and documentation, specifying the changes introduced in each version.
- Use tools like Laravel's apiResource to maintain consistency in your API routes across different versions.
- Document each version of your API separately, highlighting any differences or updates from previous versions.
- Test your APIs thoroughly with each version to ensure they function as expected across all supported versions.



Application of learning 4.7.

KivuConnect, a tech startup located in Rubavu District, Western Province, specializes in developing web applications that integrate with various online services. As their platform grows, they need to implement API versioning to manage different versions of their APIs and provide clear documentation for their development team and external partners. You have been brought on board by KivuConnect to set up API versioning and create comprehensive API documentation within the Laravel framework, ensuring that their APIs are scalable, maintainable, and easy to use for future development.

Checklist

SN	Criteria	Indicators	Observation	
			Yes	No
1	API is properly versioned and documented	1.1 A Laravel project is created		
		1.2 Versioned routes are added		
		1.3 Versioned routes are tested		
		1.4 Swagger is installed and configured		
		1.5 API endpoints are annotated		
		1.6 Swagger documentation is generated		
		1.7 Swagger documentation can be accessed		



Learning outcome 4 end assessment

Theoretical assessment

I. Read the following statement related PHP programming and choose the correct letter that corresponding to the correct

1. Which of the following is a key advantage of using a PHP framework?

- A) Slower development process
- B) Lack of security features
- C) Streamlined code organization
- D) Increased need for manual coding

Answer: C) Streamlined code organization

2. Which PHP framework is known for its simplicity and speed, often used for building RESTful APIs?

- A) Laravel
- B) Symfony
- C) CodeIgniter
- D) Zend Framework

Answer: C) CodeIgniter

3. Which PHP framework uses the Eloquent ORM for database interactions?

- A) Yii
- B) Laravel
- C) CakePHP
- D) Slim

Answer: B) Laravel

4. In Laravel, which command is used to create a new controller?

- A) php artisan make:controller
- B) php artisan new:controller
- C) php artisan create:controller

D) php artisan generate:controller

Answer: A) php artisan make:controller

5. What does MVC stand for in the context of PHP frameworks?

A) Model-View-Component

B) Module-View-Controller

C) Model-View-Controller

D) Module-View-Component

Answer: C) Model-View-Controller

II. Read the following statement related to php programming and complete with the appropriate terminology among: **swagger, MVC, HttpFoundation, Postman, small, Eloquent ORM, Models**

1. _____ is an open-source tool for designing, building, and documenting APIs. It uses the OpenAPI Specification (OAS) to define the structure and behavior of your API.

Answer: **Swagger**

2. _____ is a popular tool for testing APIs, and it also provides powerful features for API documentation.

Answer: **Postman**

3. The _____ pattern is commonly used in PHP frameworks to separate concerns in web applications.

Answer: **MVC**

4. In Symfony, the _____ component is responsible for handling HTTP requests and responses.

Answer: **HttpFoundation**

5. Laravel's _____ provides a simple and expressive syntax for database queries.

Answer: **Eloquent ORM**

6. CodeIgniter is known for its _____ footprint, making it suitable for shared hosting environments.

Answer: **Small**

7. In CakePHP, _____ are used to validate and handle data before saving it to the database.

Answer: **Models**

III. Read the following statement related PHP programming and answer by True if the statement is correct and False if the statement is wrong

1. PHP frameworks like Laravel and Symfony are designed to help developers build web applications faster and more securely.

Answer: True

2. The Zend Framework is now known as Laminas.

Answer: True

3. CodeIgniter requires the use of the Eloquent ORM for database operations.

Answer: False

4. In Laravel, middleware is used to filter HTTP requests entering your application.

Answer: True

5. Symfony's components are reusable PHP libraries that can be used independently of the full framework.

Answer: True

Practical assessment

InnovateTech, a tech startup in Kigali, Rwanda, has recently secured a contract to develop a comprehensive web application for a local non-profit organization. The organization requires a web app to manage donor information, track fundraising events, and generate reports. As a newly hired Laravel developer at InnovateTech, you are tasked with building this web application using the Laravel MVC framework. Your responsibilities include setting up the Laravel environment, designing the database schema, and implementing models, views, and controllers to handle various functionalities. You will need to create routes and controllers to manage donor information, event schedules, and reporting features. Additionally, integrate authentication and authorization mechanisms to ensure secure access to sensitive data. Develop a responsive and user-friendly interface that allows administrators to easily navigate the application and perform tasks. After implementing the core features, conduct thorough testing to ensure the application works seamlessly across different devices and browsers. Document your development process and provide a detailed user guide to the client. Your goal is to deliver a robust and scalable web application that meets the non-profit organization's needs and enhances their ability to manage their operations effectively.

Checklist

SN	Elements	Indicators	Observation	
			Yes	No
1	Laravel framework environment is fully set up	1.1 Laravel is installed		
		1.2 Laravel .env file is configured		
		1.3 Laravel is correctly launched		
		1.4 Laravel environment is set		
2	Custom routing are set and configured	2.1 MVC Laravel project is built		
		2.2 Routes are created		
		2.3 Routes are configured		
		2.4 Routes are protected		
		2.5 Routes are grouped		
3	Laravel form data are well validated	3.1 CSRF Token is used on forms		
		3.2 Server-side validation is applied		
		3.3 Error messages are customised		
		3.4 Flash messages are used		
4	CRUD Operations are implemented	4.1 Database is created and configured		
		4.2 Models and migrations are created		
		4.3 Resource controller is generated		
		4.4 Views are created		
		4.5 CRUD functionalities are working		
5	APIs are authenticated and secured	5.1 Laravel sanctum is used		
		5.2 Endpoints are secured		
		5.3 Authentication middleware are set		
		5.4 API tokens are well managed		
6	Laravel project and APIs are versioned and documented	6.1 Versioned routes are added		
		6.2 Versioned routes are tested		
		6.3 Swagger is configured and used		
		6.4 API endpoints are annotated		
		6.5 Swagger documentation is generated		

		6.6 Swagger documentation can be accessed		
--	--	---	--	--

END



Further information to the trainer

Bierer, D. (2016). *PHP 7 programming book*. Birmingham, B32PB, UK: Pack Publishing Ltd.

Castagnetto, J., & Rawat, H. (1999). *Professional PHP Programming*. Birmingham: Wrox Press Ltd.

Katz, J., & Sipos, B. (2019). *Laravel: Up & Running: A Framework for Building Modern PHP Apps* (2nd ed.). O'Reilly Media.

Keenan, M. (2019). *Laravel Design Patterns and Best Practices* (2nd ed.). Packt Publishing.

Nixon, R. (2014). *Learning PHP, MySQL, JavaScript, CSS & HTML* (3rd ed.). O'Reilly Media.

Popel, D. (2007). *Learning Data Object*. Luna Park, Sydney: Pack Publisher.

Stauffer, M. (2017). *Laravel: The Definitive Guide*. Leanpub.

Stauffer, M. (2020). *Laravel: A Beginner's Guide*. McGraw-Hill Education.

Williams, R. (2019). *Pro PHP MVC* (2nd ed.). Apress.

Otwell, A. (2023, May 6). *The PHP Framework for Web Artisans*. Retrieved from Laravel Official Website: <https://laravel.com/docs/10.x>

Javatpoint. (2023, May 4). *PHP Tutorial*. Retrieved from <https://www.javatpoint.com>

Javatpoint. (2023, May 6). *Laravel Tutorial*. Retrieved from <https://www.javatpoint.com/laravel>

