Republic of Rwanda
Ministry of Education

RTB | RWANDA TVET BOARD

PHP PROGRAMMING

SWDPP401

Apply PHP Programming

Competence

RQF Level: 4

Learning Hours

130

Credits: 13

Sector: ICT and Multimedia

Trade: Software Development

Module Type: Specific

Curriculum: ICTSWD4002: TVET Certificate IV in Software Development

Issue Date: September 2023

<table>
<tr><td>Purpose statement</td><td colspan="5">This module describes the skills, knowledge, and attitude required to apply php programming. This module is intended to prepare students pursuing TVET Level 4 in software development. At the end of this module, the students will be able to Apply PHP Fundamentals, Connect PHP to the Database, and Build a Content Management System (CMS) using PHP, Build a web app using MVC Framework (LARAVEL)</td></tr>
<tr><td>Learning assumed to be in place</td><td colspan="5"></td></tr>
<tr><td rowspan="6">Delivery modality</td><td colspan="2">Training delivery</td><td>100%</td><td colspan="2">Assessment</td><td>Total 100%</td></tr>
<tr><td colspan="2">Theoretical content</td><td>30%</td><td rowspan="4">Formative assessment</td><td>30%</td><td rowspan="4">50%</td></tr>
<tr><td colspan="2">Practical work:</td><td rowspan="3">70%</td><td rowspan="3">70%</td></tr>
<tr><td>Group project and presentation</td><td>20%</td></tr>
<tr><td>Individual project /Work</td><td>50%</td></tr>
<tr><td colspan="2"></td><td colspan="2">Summative Assessment</td><td>50%</td></tr>
</table>

Elements of Competence and Performance Criteria

Elements of competence	Performance criteria
1.Apply PHP Fundamentals	1.1 Environment is properly prepared based on PHP Requirements.
	1.2 PHP Concepts are effectively applied based on PHP standards.
	1.3 PHP Security concepts are effectively applied based on security best practices
	1.4 OOP concepts are effectively applied based on PHP Standards
2. Connect PHP to the Database	2.1 Connection is properly established based on specified database environment
	2.2 CRUD Operations are effectively performed based on PHP standards
	2.3 Security is effectively enforced based PHP data protection standards
	2.4 Errors and exceptions are correctly handled based on error handling standards
	2.5 User authentication are effectively performed based on Web Page Control
3. Build a Content Management System (CMS) using PHP	3.1 Dynamic Content Navigation is properly built based on PHP standards
	3.2 Cookies and Sessions are properly managed based on PHP standards
	3.3 Context options are effectively applied on PHP standards
	3.4 Pages access are effectively regulated based on PHP standards

	3.5 CMS Errors and Logging is properly detected based on CMS security
	3.6. CMS is effectively maintained based on PHP Standards
4. Build a web app using MVC Framework (LARAVEL)	4.1 Environment is properly configured based on Laravel Framework Standard
	4.2 Custom routing is effectively set based on Laravel Framework Standard
	4.3 Form Data are properly validated based on Laravel Framework standard
	4.4 CRUD operations are effectively performed based on Laravel framework standard
	4.5 APIs are properly managed based on web app requirements
	4.6 Web Application is effectively secured based on framework security standard
	4.7 Source code changes are properly managed according to the version control standard

Course content

Learning outcomes	At the end of the module the learner will be able to:
	1. Apply PHP Fundamentals 2. Connect PHP to the Database 3. Build a Content Management System (CMS) using PHP 4. Build a web app using MVC Framework (LARAVEL)

Learning outcome 1: Apply PHP Fundamentals.	Learning hours: 40
Indicative content	

- **Preparation of PHP Programming environment**
  - ✓ Definition of key terms
    - PHP
    - Interpreter
    - Open Source
    - Web Server
    - Apache
    - Database
    - DBMS
    - MySQL
    - Static website
    - Dynamic website
  - ✓ Purpose of PHP
  - ✓ Important characteristics of PHP
  - ✓ PHP Development Tools

- XAMPP
- WAMP/MAMP/LAMP
- IDEs /Text Editors
- Browser

✓ Installation of XAMPP/WAMP or LAMP

✓ Configuration of environment

- Ports
- Browser
- Services
- IDEs Extensions

- Application of PHP concepts
  - ✓ PHP file extension
  - ✓ Syntax
  - ✓ Variable
  - ✓ Operators
  - ✓ Data types
  - ✓ Variable scope
  - ✓ Constants
  - ✓ Comment
  - ✓ Date and time
  - ✓ String concatenation
  - ✓ Condition statement
  - ✓ Arrays
  - ✓ Loop
  - ✓ Function
    - Introduction to function
    - Built-in functions
    - User-defined functions
    - calling function

- Function recursion

✓ Super Global variables

✓ PHP file handling

- Opening a file
- Reading a file
- Writing a file
- Closing a file
- Deleting a file

- Application of PHP Security concepts

✓ PHP form handling

- Post Method
- Get Method
- Validation

✓ Cookies and Session

- **Implementation of Object-oriented programming (OOP) in PHP**

✓ Definition

✓ Classes

✓ Objects

✓ Inheritance

✓ Access modifiers

✓ Encapsulation

✓ Abstraction

✓ Polymorphism

Resources required for the learning outcome	
Equipment	▪ Computer
Materials	▪ Books ▪ Tutorials ▪ Internet

Tools	▪ textditor ▪ IDE ▪ Browser ▪ Xampp/Wampp
Facilitation techniques	▪ Demonstration and simulation ▪ Individual and group work ▪ Practical exercise ▪ Group discussion
Formative assessment methods /(CAT)	▪ Written assessment ▪ Oral presentation ▪ Practical Assessments

Learning outcome 2: Connect PHP to the Database	**Learning hours: 37**
Indicative content	

- Application of Database Connection drives
  - ✓ Mysqli
  - ✓ Mysqli - OOP
  - ✓ PDO
- Perform database CRUD Operations
  - ✓ CRUD with Mysqli
  - ✓ CRUD with Mysqli – OOP
  - ✓ CRUD with PDO
  - ✓ Import and export database
- Application of PHP Basic security concepts
  - ✓ Input Validation
  - ✓ Password Security

- ✓ Cross-Site Scripting (XSS) Prevention
- ✓ Cross-Site Request Forgery (CSRF) Prevention
- ✓ Session Security
- ✓ File Uploads
- ✓ Error Reporting

- Errors and exceptions in PHP
  - ✓ Introduction
  - ✓ Types of errors
  - ✓ Exception Handling
    - Simple “die ()” statements
    - Custom error and error triggers
    - Error reporting

- Implementation of user authentication
  - ✓ Introduction
  - ✓ Types of user authentication
  - ✓ User authorization
  - ✓ Create User authentication
    - Start a session
    - Authenticate the user
    - Protect pages

Resources required for the indicative content	
Equipment	▪ Computer
Materials	▪ Books ▪ Tutorials ▪ Internet
Tools	▪ textditor ▪ IDE ▪ Browser

	▪ Xampp/Wampp
Facilitation techniques	▪ Demonstration and simulation ▪ Individual and group work ▪ Practical exercise ▪ Group discussion
Formative assessment methods /(CAT)	▪ Written assessment ▪ Oral presentation ▪ Practical Assessments

Learning outcome 3: Build a Content Management System (CMS) using PHP	**Learning hours: 25**
Indicative content	

- **Preparation of Content Management System (CMS)**
  - ✓ Introduction to CMS
  - ✓ Prepare CMS Environment
    - Blueprint the application
    - Set up the database
    - Set up project files and folders
- **Build dynamic content navigation**
  - ✓ List subjects
  - ✓ Add pages for each subject
  - ✓ Add page content
  - ✓ Use the navigation to select pages
- **Management of cookies and sessions**
  - ✓ Work with cookies
  - ✓ Set and read cookies values
  - ✓ Unset cookie values

- ✓ Work with sessions
- ✓ Set and read session values
- ✓ Unset Session

- **Application of Context and Options**
  - ✓ The public content
  - ✓ Skip hidden subjects and pages
  - ✓ Use an option for conditional code
  - ✓ Insecure direct object reference
  - ✓ Project page visibility
  - ✓ Allow html in dynamic contents
- **Regulate page access**
  - ✓ User authentication overview
  - ✓ Create admins table
  - ✓ Build admin Dashboard
  - ✓ PHP password functions
  - ✓ Authentication user access
  - ✓ Require authorization
  - ✓ Log out user
  - ✓ Optional password updating
  - ✓ Authorized previewing
- **CMS Errors Detection**
  - ✓ Description of CMS Errors
  - ✓ Application of Errors testing
- **Maintain CMS**
  - ✓ Regular updates
  - ✓ Plugin and module updates
  - ✓ Regular backups
  - ✓ Database optimization
  - ✓ Security measures

✓ Performance monitoring

Resources required for the indicative content	
Equipment	▪ Computer
Materials	▪ Books ▪ Tutorials ▪ Internet
Tools	▪ Textditor ▪ IDE ▪ Browser ▪ Xampp/Wampp ▪ Scratch
Facilitation techniques	▪ Demonstration and simulation ▪ Individual and group work ▪ Practical exercise ▪ Group discussion
Formative assessment methods /(CAT)	▪ Written assessment ▪ Oral presentation ▪ Practical Assessments

Learning outcome 4: Build a web app using MVC Framework (LARAVEL)	Learning hours: 28

Indicative content

- **Framework environment configuration**
  - ✓ Introduction to PHP framework
  - ✓ Most popular PHP frameworks
    - Characteristics of each
    - Advantages and disadvantages/limitations of each
  - ✓ Laravel MVC Architecture (Model, View, Controller)
  - ✓ Installation of Laravel framework
  - ✓ Laravel .env file configuration
  - ✓ Use blade template for Laravel
- **Setup Laravel custom routing**
  - ✓ Web and API routing
    - Laravel basic routing
    - Routing parameters
    - Laravel named routes
    - Laravel middleware
    - Laravel route groups
- **Perform form data validation**
  - ✓ CSRF Token
  - ✓ Form Elements
- **Perform CRUD Operations**
  - ✓ Configure database file
  - ✓ Create Controllers for Laravel CRUD
  - ✓ Create Models for Laravel CRUD
  - ✓ Creation of migration

- ✓ Perform Seeding
- ✓ Create Views for Laravel CRUD
- ✓ Laravel CRUD operation routes

- **Manage APIs In Laravel frameworks**
  - ✓ Introduction to API development
  - ✓ RESTful APIs
    - Understanding RESTful architecture
    - Building RESTful APIs with Laravel
    - Test APIs with Postman
    - Handling HTTP requests and responses
  - ✓ API Resources
    - Creating API resources
    - Returning resources as JSON
    - Encoding API data
- **Authentication and Security**
  - ✓ Implementing API authentication
  - ✓ Best practices for API security
  - ✓ Managing API security
- **API Versioning and Documentation**
  - ✓ Versioning your API
  - ✓ Documenting your API with Swagger/Postman
  - ✓ Best practices for API documentation

Resources required for the indicative content	
Equipment	▪ Computer
Materials	▪ Books ▪ Tutorials ▪ Internet
Tools	▪ textditor

	▪ IDE ▪ Browser ▪ Xampp/Wampp ▪ Laravel Frameworks
Facilitation techniques	▪ Demonstration and simulation ▪ Individual and group work ▪ Practical exercise ▪ Group discussion
Formative assessment methods /(CAT)	▪ Written assessment ▪ Oral presentation ▪ Practical Assessments

Integrated/Summative assessment

Integrated situation

XY Shop is located in Kigali City, Kicukiro District Sales shoes and clothes.

The Shopkeeper (someone who owns or manages a shop) uses a file system (books) to record stock information. This filling system has a problem with non-efficient security, accessibility, and integrity of information about the stock-in and stock-out, not only that but also lacks an easy way to produce a daily/weekly report of the stock.

They hired a database designer who designed the database model that can be used to manage the stock. The designed database is below:

Database Name: XY_Shop

Tables:

- Shopkeeper (ShopkeeperId (PK), UserName, Password)
- Product (ProductCode (PK), ProductName)
- ProductIn (ProductCode (FK), DateTime, Quantity, UnitPrice, TotalPrice)
- ProductOut (ProductCode (FK), DateTime, Quantity, UnitPrice, TotalPrice)

The XY Shop has hired you as a web application developer to develop a web application using Laravel that enables the Shopkeeper to :

- Record/insert the products with the ability to view, modify and delete products.
- Generate a report of stock status and the total price of stock-in or stock-out.

Instruction

- The shopkeeper must have an account in order to login into the system before starting all other activities
- Create an application directory on your desktop and rename it with your FirstName_LastName and save your work.
- The application is needed in four (4) hours.

Resources

Tools	▪ TextEdit, VS code , PHPStorm, Browser, Xampp and Laravel Frameworks
Equipment	▪ Computer
Materials/ Consumables	▪ Electricity

Assessable outcomes	Assessment criteria (Based on performance criteria)	Indicator	Observation		Marks allocation
			Yes	No	
Learning outcome 1:Apply PHP Fundamentals	1.1 Environment is properly prepared based on PHP Requirements	Ind.1 Required tools are Selected			5
		Ind.2 PHP Development Environment is configured			5
	1.2 PHP Concepts are effectively applied based on PHP standards.	Ind.1 Variables are Applied			**5**
		Ind.2 Function are used			**6**
		Ind.3 Control statement is Implemented			**6**
Learning outcome 2:	2.1. Connection is properly established based on specified database environment	Ind.1 Database is connected			6

Connect PHP to the Database	2.3 PHP Security concepts are effectively applied based on security best practices	Ind.1 Cookies and Session are applied			6
	2.5 User authentication are effectively performed based on Web Page Control	Ind.1 User authentication are performed			5
Learning outcome 3: **Build a web app using MVC Framework (LARAVEL)**	4.1. Environment is properly configured based on Laravel Framework Standard	Ind.1 Laravel Framework is Installed			5
		Ind.2 Laravel. env file is Configured			6
		Ind.3. Blade template is used			2
	4.2. Custom routing is effectively set based on Laravel Framework Standard	Ind.1 Custom routing is applied			6
	4.3 Form Data are properly validated based on Laravel Framework standard	Ind.1 data validation is performed			8
		Ind.2 Form handling are applied			6

	4.4 CRUD operations are effectively performed based on Laravel framework standard	Ind.1. Database is Configured			5
		Ind.2. Models are created			4
		Ind.3. Migration is created			4
		Ind.4 CRUD operations are performed			10
Total marks					**100**
Percentage Weightage					**100%**
Minimum Passing line % (Aggregate): 70%					

References

1. Bierer, D. (2016). *PHP 7 programming book.* Birmingham B32PB,UK: Pack publishing LTD.

2. javatpoint. (2023, may 6th saturday). *https://www.javatpoint.com/laravel*. Retrieved from javatpoint: https://www.javatpoint.com/

3. javatpoint. (2023, may 4th thursday). *https://www.javatpoint.com/php-tutorial*. Retrieved from javatpoint: https://www.javatpoint.com

4. JavaTpoint. (2023, may 6th saturday). *javatpoint*. Retrieved from javatpoint tutorial: https://www.javatpoint.com/

5. Jesus Castagnetto, H. R. (1999). *professional PHP Programming.* Birminghan: Wrox press Lmd.

6. Nixon, R. (2014). *Learning php,mysql,java script ,css&HTML,third edition.* USA: Oreilly media.

7. Otwell., a. (2023, may saturday on 6th). *The PHP Framework for web artisan*. Retrieved from laravel official website: https://laravel.com/docs/10.x

8. popel, D. (2007). *Learning data object.* Luna park sydney: pack publisher.