**PROCEDURAL PHP**

**Learning Outcome**:

At the end of the training, participants would be able to develop relatively secure database-driven website with role-based access.

Projects List:

* Simple Website
* Simple Database-driven website (Photo Gallery)
* Multi-user, multi-role website with login and forgot password utility

Topic List

**Day 1: Create a Simple, Multi-page Website with Navigation Bar**

Session Coverage:

1. XAMPP Environment
   1. Components of XAMPP
   2. PHP Version
   3. PHP extensions
   4. Run Visual Studio Code
2. Procedural PHP Programming
3. **PHP File construction**
4. **PHP Statement format**
5. Logic flow (do not use a variable or function not yet defined)
6. **Comment**
7. Variable
8. Variable typecasting
9. Operators
   1. Arithmetic
   2. Assignment
   3. Comparison
   4. Logic
10. Multiline command
11. Concatenation
12. Constants
13. Superglobal variables
14. $\_GET
15. Conditions
    1. SWITCH
    2. IF-THEN
    3. ELSE
    4. ELSEIF

Exercise: Write code on conditional statements

1. Loops
2. While
3. Do While
4. FOR
5. FOREACH
6. Arrays
   1. Creating Arrays
   2. Identifying elements in an array
   3. Numerically indexed Array
   4. Associative Array
   5. Using arrays in statements
   6. Arrays in list statements
7. Functions
   1. Built-in
   2. User-defined
8. Include vs Require file
9. HTML, CSS and Bootstrap

Project Assignment: ***Create a simple, mobile-responsive LGU website using Bootstrap***

Requirements:

* The website must have the usual minimum pages and parts: navigation bar, banner title, footer, main section and minor section / sidebar.
* The website must have at least 5 main section-pages, with at least two sections having at least two sub-pages.
* There must be a navigation bar that reflects the organization of the sections.
* The homepage must have a main section with a picture that spans the width of the main section, and a sidebar for text or links.
* The pages must reflect different layout options.
* The website layout must adjust when viewed from a smartphone.
* During presentation, present the source code with comments explaining major functions.

**Day 3: Create an Interactive and Database-driven Website**

Session Coverage:

1. Review of Relational Database: Power Plants
   1. What would the database structure look like? Review relational database concept
   2. Who is the user of the system?
   3. What reports are needed by management and other system users?
   4. Access MySQL from command line (XAMPP🡪Run MySQL🡪 Shell🡪 mysql -h 127.0.0.1 -u root
2. Introduction to MySQL commands:
   1. Database
      1. Create
      2. Alter
      3. Drop
      4. Use
   2. Table
      1. Create
      2. Alter
      3. Rename
      4. Drop
      5. Describe
   3. Record
      1. Insert
      2. Select
      3. Order by
      4. Group by
      5. Where
      6. Update
      7. Delete
   4. Join
   5. Create View

Exercise: Write SQL statements

1. Introduction to PHPMyAdmin
2. Accessing the database with PHP
   1. Create a database connection
   2. Create a query
   3. Display results of a single-value query
   4. Display a results array
   5. Insert data
   6. Update data
   7. Delete data
3. Create a form to submit website comments
4. Upload file
5. Use the uploaded file (image)
6. Introduction to Git and Github
   1. Install Git
   2. Configure VS Code to use Git Bash
   3. Add source code
   4. Add to commit
   5. Remove a file from repository
   6. Create .gitignore file
   7. Create Github Account
   8. Create online repository
   9. Add repository to computer
   10. Push repository to Github
   11. Create README.md file
   12. Re-push new files to Github

Project 2 (by partner): ***Online Photo Gallery with information of the product***

Requirements:

* Improve the previous website by adding a photo gallery
* Create a page for submitting user comments. The user comment must display in one section, with maximum of 10 comments per page. There must be links for the remaining posts, with each page displaying 10 comments.

**Day 4: Build and Deploy Multi-user, multi-role website**

**This session will be a full-day follow-my-lead demonstration building a blog site.**

Before the beginning of the demonstration, participants shall identify their project, what it is supposed to achieve, and what minimum functions it should have. From the technical training side, it should have the following functionalities:

* Login form
* Password forgot form
* Session-based security
* Multi-role access
* Email Notification
* Image upload
* Database CRUD
* HTML5 and Bootstrap
  + Navigation Bar
  + Mobile-responsive

Session Coverage:

1. Session
2. Password
3. Hashing
4. Security
   1. filter\_var
   2. FILTER\_SANITIZE\_EMAIL
   3. FILTER\_SANITIZE\_NUMBER\_INPUT
   4. FILTER\_SANITIZE\_SPECIAL\_CHARS
   5. FILTER\_CALLBACK
   6. FILTER\_VALIDATE\_INT
5. Prepared statements
6. Uploading PHP files to webhost using Filezilla

Project 3: Blog Site

Requirements:

* Blog posts can be seen by public users.
* Creating post requires registration using active email address and password.
* Blog posts can only be edited by the user that created the blog post.
* Users can create tags.
* A post may have one or more tags.
* Implement minimum acceptable security practice upon account creation.
* When signed in, there is a page that displays the list of posts created by the signed-in user.
* The page can allow the user to edit or delete the created posts.
* The URL must not show that PHP is used.
* Create page listing registered users with number of published posts.
* User profile page has a photo, which is uploaded by the user.
* Create page for admin to deactivate account.
* Public can comment on posts without registering. Full name will be asked upon submitting a comment.
* The main section should be 2/3 of the horizontal width of the screen, while the remaining part will be for links to similar next post, according to date of post.
* For mobile view, the sidebar is expected to move to the bottom of the page.
* The navigation bar is also expected to be mobile responsive.