

Deployment of a PHP Web Application on a Windows Web Server

Student Name: HAKIZIMANA Emmanuel

Reg No: 24RP03266

Class: Y2 IT B

Module: Develop backend using PHP

1. Introduction

This report documents the process of deploying a previously developed PHP web application onto a Windows Server environment using Internet Information Services (IIS) as the web server. The goal was to create a fully functional, secure, and accessible runtime environment for the application. The process involved setting up IIS, installing PHP, configuring databases, deploying code, and implementing critical security measures.

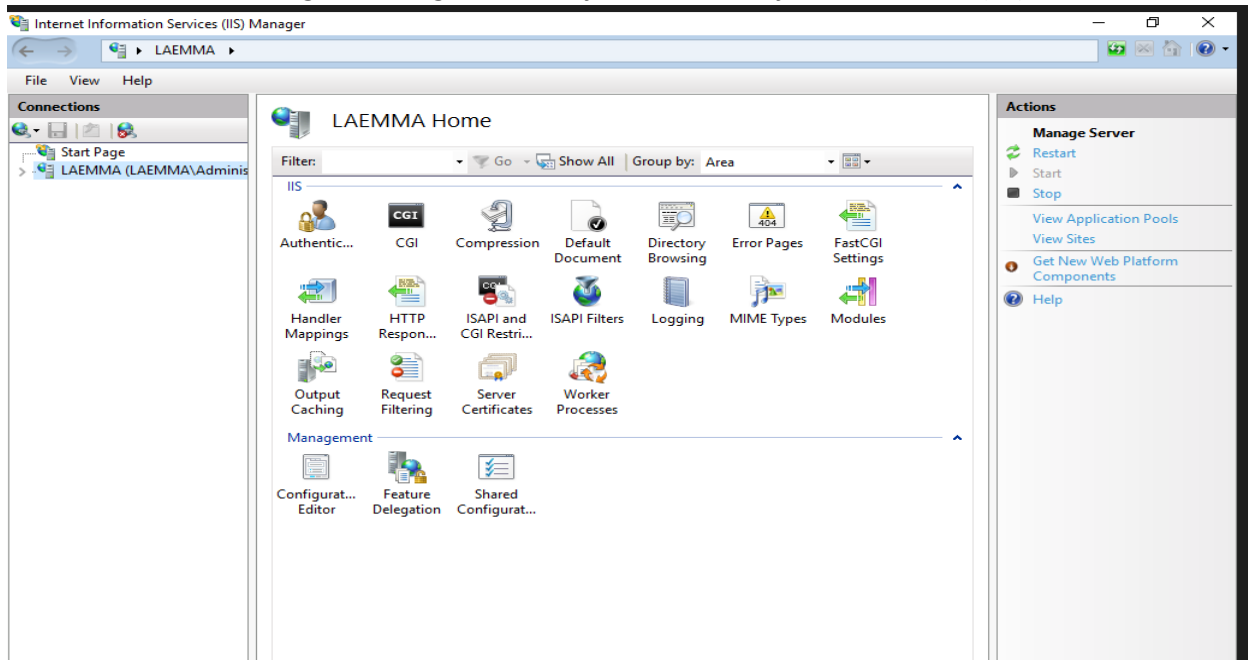
Phase 1: Setting Up the Windows Web Server (IIS)

The first step was to install the Web Server (IIS) role on Windows Server.

Steps:

1. Opened **Server Manager**.
2. Clicked **Add roles and features**.
3. Selected **Role-based or feature-based installation**.
4. Choose the local server.
5. From the roles list, we selected **Web Server (IIS)**.
6. On the **Role Services** page for IIS, I assured you the following were selected:
 - **Web Server > Common HTTP Features > Default Document** (Critical for serving index.php)
 - **Web Server > Application Development > CGI** (This is one way to run PHP, but we will use the better method)
 - **Web Server > Application Development > ISAPI Extensions**
 - **Web Server > Application Development > ISAPI Filters**
7. Clicked **Install**.

Screenshot: Server Manager showing the successful installation of the Web Server (IIS) role.



Phase 2: Installing and Configuring PHP

Instead of the older CGI method, I used the more efficient **FastCGI** protocol for integrating PHP with IIS.

Steps:

1. **Download PHP:** I went to windows.php.net/download and downloaded the **Non-Thread Safe (NTS) x64** version of the latest PHP 8.x ZIP package. NTS is recommended for IIS with FastCGI.
2. **Extract PHP:** I created a folder C:\PHP and extracted the contents of the ZIP file into it.

Phase 3: Verifying the Environment with phpinfo()

To confirm that PHP was installed and configured correctly, I created a phpinfo.php file.

Steps:

1. I navigated to the default web directory, which is typically C:\inetpub\wwwroot.
2. I created a new file named phpinfo.php.
3. I added the following single line of code to the file and saved it:

```
<?php
echo phpinfo();
?>
```

4. I opened a web browser and went to <http://localhost/phpinfo.php>.

Screenshot: The phpinfo() page displayed in a web browser, showing detailed information about the PHP configuration, loaded extensions, and server environment.

PHP Version 8.2.29	
System	Windows NT WIN-G0V8DQKUVFC 10.0 build 14393 (Windows Server 2016) AMD64
Build Date	Jul 1 2025 20:17:36
Build System	Microsoft Windows Server 2022 Datacenter [10.0.20348]
Compiler	Visual C++ 2019
Architecture	x64
Configure Command	cscript /nologo /e:js cscript configure.js "--enable-snapshot-build" "--enable-debug-pack" "--disable-zts" "--with-oci=.\\..\\..\\instantclient\\sdk,shared" "--with-oci8-19=.\\..\\..\\instantclient\\sdk,shared" "--enable-object-out-dir=.\\obj\\" "--enable-com-dotnet=shared" "--without-analyzer" "--with-pgo"
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	no value
Loaded Configuration File	C:\php\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20220829
PHP Extension	20220829
Zend Extension	420220829
Zend Extension Build	API420220829,NTS,VS16
PHP Extension Build	API20220829,NTS,VS16
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
Zend Max Execution Timers	disabled
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	php, file, glob, data, http, ftp, compress.zlib, phar
Registered Stream Socket Transports	tcp, udp
Registered Stream Filters	convert.iconv.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, zlib.*

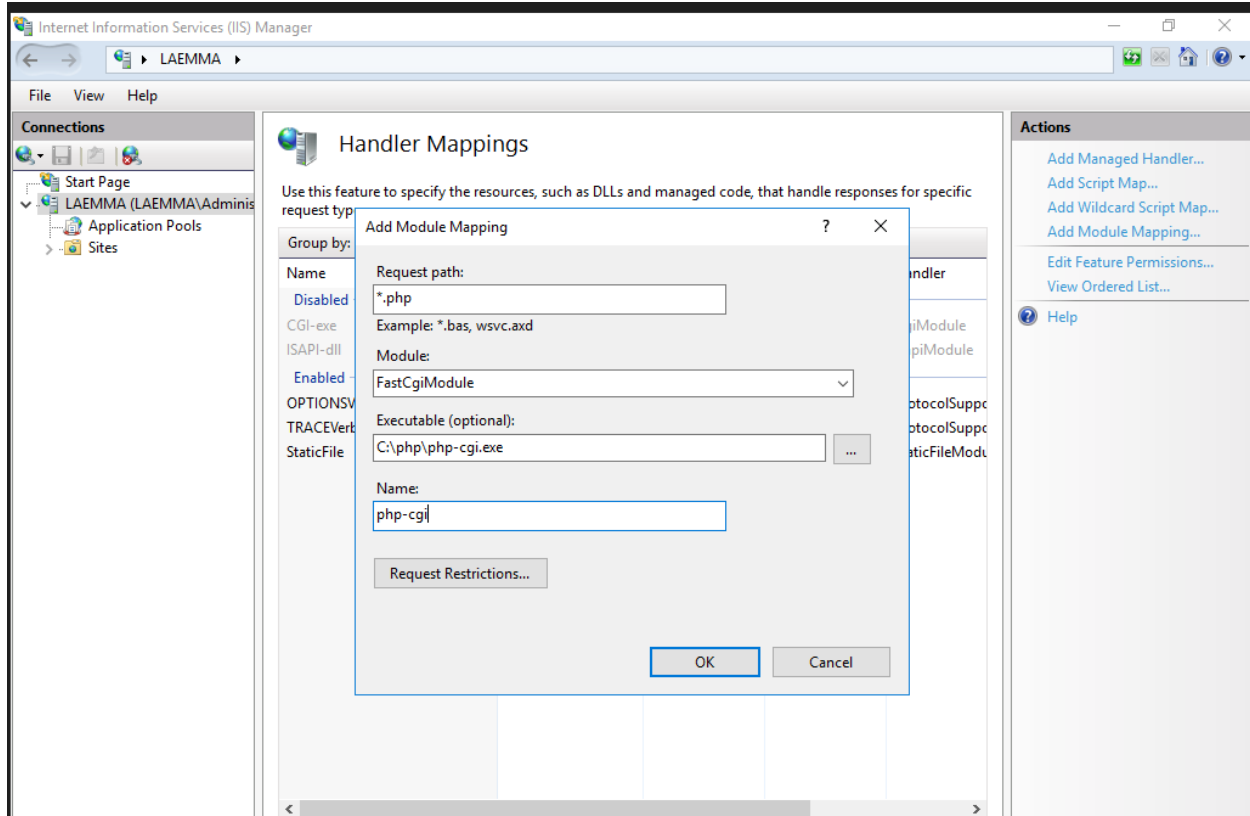
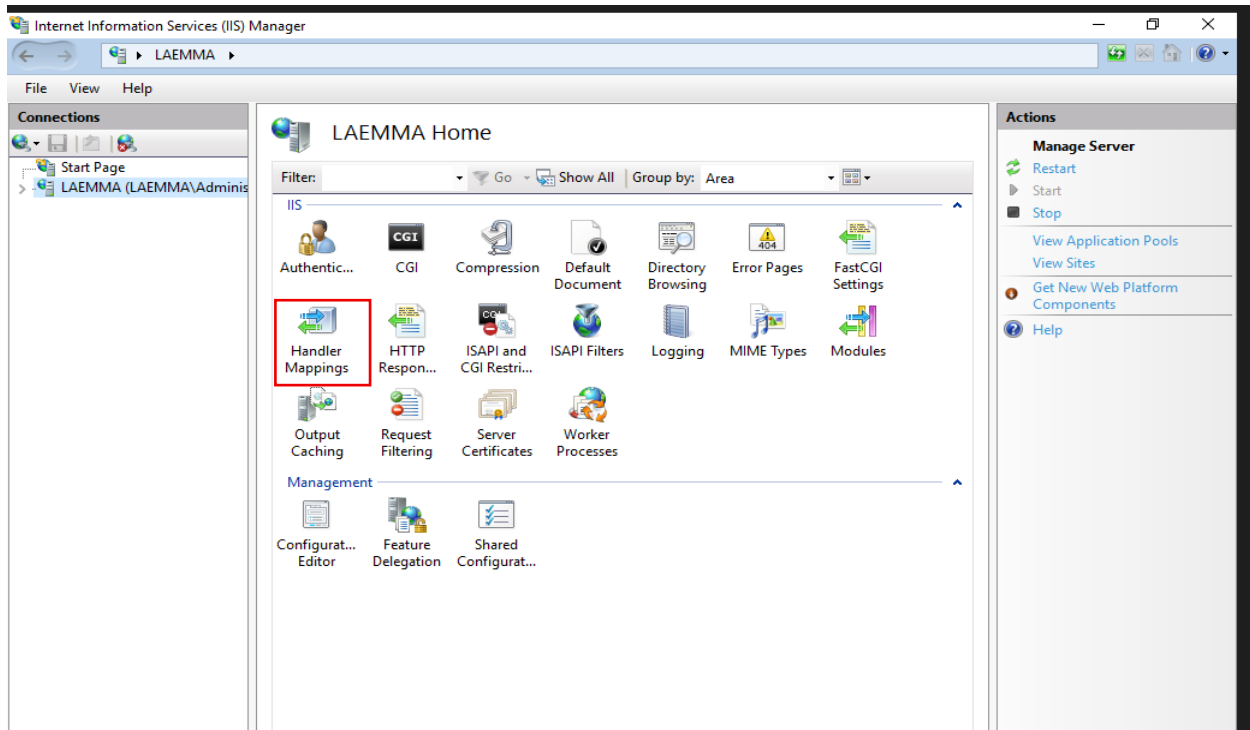
Phase 4: Deploying the PHP Application

For this deployment, I decided to use the Default Web Site, which is a better practice for hosting multiple applications. And make handler mapping

Steps:

1. **Copy Application Files:** I placed all the files from my previous PHP project into a new folder, e.g., C:\inetpub\ig.
2. **I used default Site in IIS:**

Screenshot:



Phase 5: Database Configuration (MySQL Example)

My application uses a MySQL database.

Steps:

1. **Install MySQL:** I installed MySQL Server on the same Windows machine.
2. **Create Database & User:** Using MySQL Command Line or Workbench, I created a new database and a dedicated user for the application with appropriate privileges.
3. **Configure Application Connection:** I updated my application's database configuration file (e.g., config.php) with the new connection details:

php

<?php

```
$servername = "localhost";
```

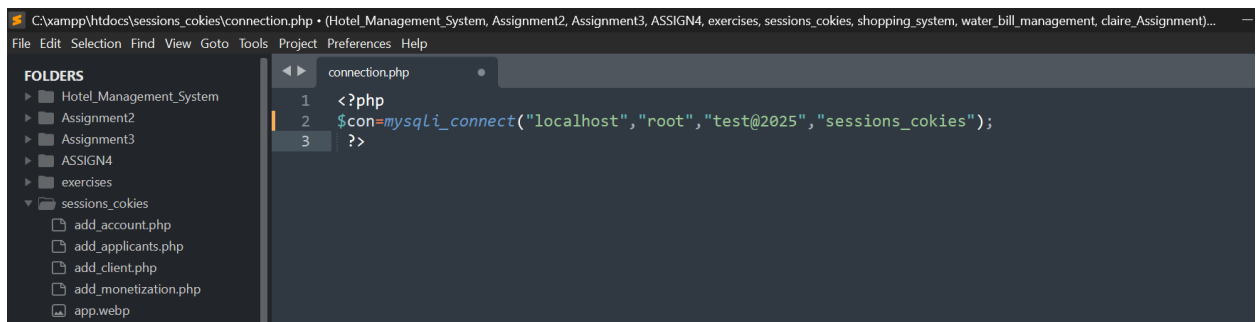
```
$username = "root";
```

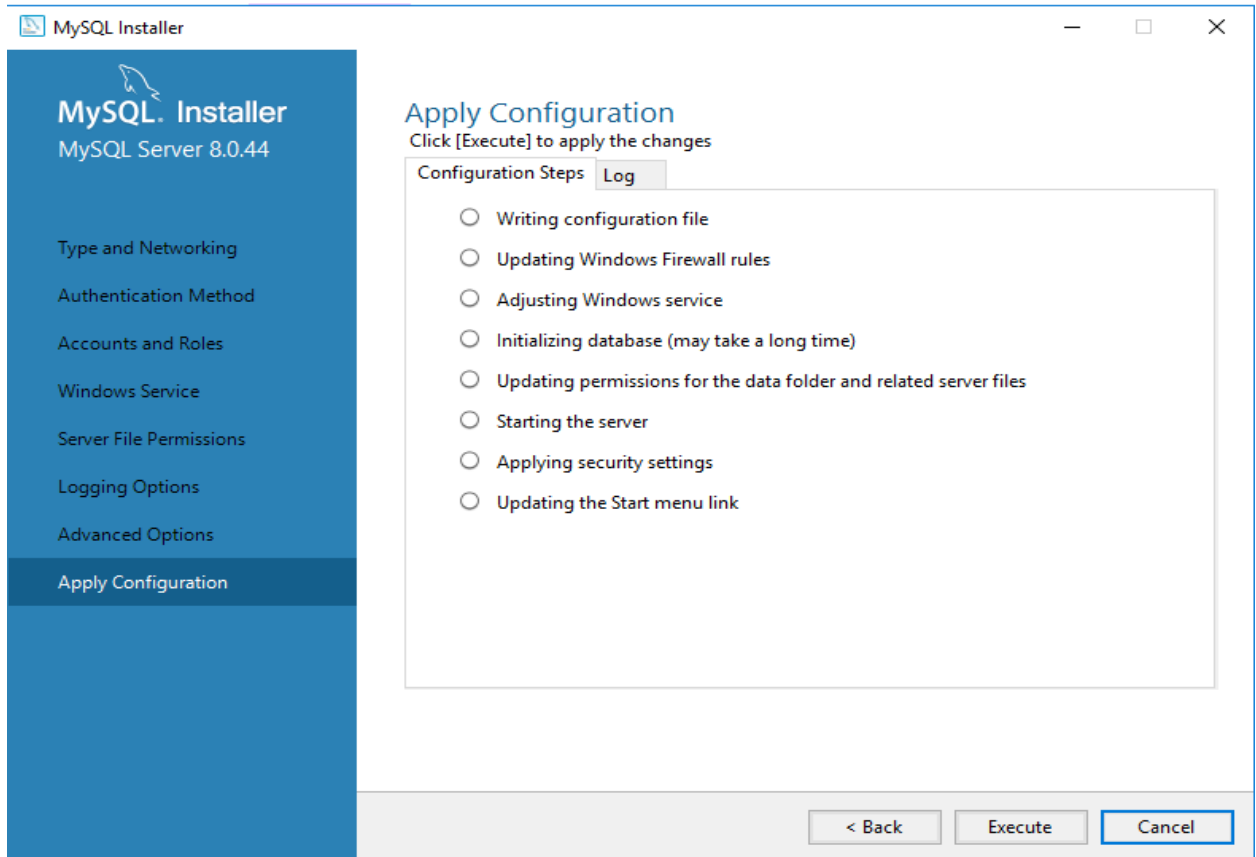
```
$password = "test@2025";
```

```
$dbname = "sessions_cokies";
```

?>

4. **Import Database Schema:** I imported the .sql dump file from my development environment into the new myapp_database to create the necessary tables.





7. Phase 6: Applying Security Configurations

Security is a critical part of the deployment.

Configurations Applied:

1. **Folder Permissions:** I assigned the correct permissions to my application folder (C:\inetpub\ig). The IIS_IUSRS group needs **Read & Execute** permissions. For any folders that need write access (e.g., uploads, logs), I granted **Modify** permissions to IIS_IUSRS specifically on those folders, not the entire site.
2. **Disable Directory Browsing:** In IIS Manager, for my site, I double-clicked the **Directory Browsing** feature and clicked **Disable** in the Actions panel.
3. **Request Filtering:** I used the **Request Filtering** feature in IIS to block requests for sensitive files like .env, .git, or config.php if they shouldn't be directly accessed.
4. **Adjust Firewall:** I ensured the Windows Firewall had an inbound rule allowing traffic on port 80 (HTTP) and, later, port 443 (HTTPS).

9. Troubleshooting: Five Issues and Solutions

1. **Issue: HTTP Error 403.14 - Forbidden (The Web server is configured to not list the contents of this directory.)**

- **Cause:** The default document (index.php) was not configured or was missing from the list.
- **Solution:** I went to the "Default Document" feature for my site in IIS and ensured index.php was present and at the top of the list.

2. **Issue: HTTP Error 500.19 - Internal Server Error (Config file is not readable)**

- **Cause:** The web.config file (or application files) had incorrect permissions. The IIS_IUSRS group did not have read access.
- **Solution:** I right-clicked the application folder, went to **Properties > Security**, and added the IIS_IUSRS group with **Read & Execute** permissions.

3. **Issue:** Call to undefined function mysqli_connect()

- **Cause:** The php_mysqli extension was not enabled in the php.ini file.
- **Solution:** I opened C:\PHP\php.ini, searched for ;extension=mysqli, and removed the semicolon (;) to uncomment it. I then **restarted the IIS server** from the command line (iisreset) for the changes to take effect.

10. Accessing my project in browser

Facebook Login

localhost/sessions_cookies/login.php

Summarize

facebook

Log Into Facebook

Email or phone number

Password

Log In

[Forgot account?](#) · [Sign up for Facebook](#)

Login with Google

Facebook Login

localhost/sessions_cookies/add_account.php

Summarize

facebook

Sign up

Firstname

Lastname

yyy@gmail.com

Password

Confirm Password

Sign Up

