Lab 1: Setting Up a Development Environment

Lab Overview

In this lab, students will set up a complete development environment for PHP web application development. The setup will include the installation and configuration of the following tools:

- Visual Studio Code (VS Code) with a plugin configured to use PHP from XAMPP.
- XAMPP with the latest version of PHP.
- Podman, Podman Desktop, and Podman Compose for containerized development.
- Git and GitHub Desktop for version control.

The lab focuses on ensuring students can independently configure their environment, which will be used for future assignments and projects in this course.

Lab Objectives

1. Understand the importance of a development environment.
2. Install and configure Visual Studio Code to use PHP from XAMPP.
3. Install XAMPP and verify the PHP installation.
4. Install and set up Podman, Podman Desktop, and Podman Compose for containerized development.
5. Install Git and GitHub Desktop for version control.
6. Write and run a "Hello World" PHP program.
7. Commit all codes, screenshots, and a report to a private GitHub repository.
8. Submit the GitHub repository link on Blackboard.

Lab Tasks and Deliverables

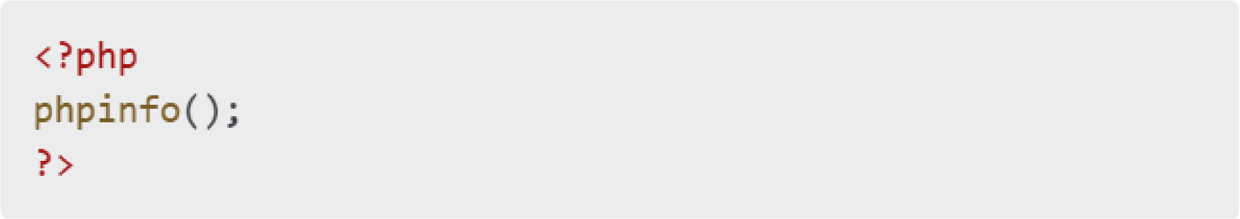
Task 1: Install Visual Studio Code

1. Download the Visual Studio Code installer from the official website.
2. Install Visual Studio Code by following the installation wizard.

Task 2: Install XAMPP with the Latest Version of PHP

1. Download the latest version of XAMPP from the official website.
2. Run the installer and follow the steps to install XAMPP.
3. Start the Apache and MySQL modules using the XAMPP Control Panel.

4. Verify the PHP installation:
   - Open the XAMPP htdocs directory.
   - Create a new file named info.php with the following content:

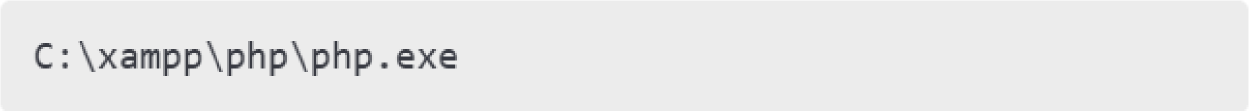


```
<?php
phpinfo();
?>
```

   - Open a web browser and navigate to http://localhost/info.php.
   - Confirm the PHP information page is displayed.
   - Take a screenshot of the PHP version displayed.

Task 3: Configure Visual Studio Code to Use PHP from XAMPP

1. Open Visual Studio Code and navigate to the Extensions Marketplace (Ctrl+Shift+X).
2. Search for **PHP Debug** and click **Install**.
3. Configure VS Code to use PHP from XAMPP:
   - Open the settings in VS Code.
   - Search for php.executablePath and set it to the path of the PHP executable in your XAMPP installation.



```
C:\xampp\php\php.exe
```

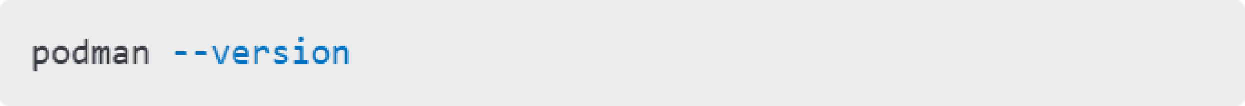
   - Save the settings.
4. Verify the configuration by creating a new PHP file (e.g., test.php) and ensuring the PHP interpreter is detected.

Task 4: Install Podman, Podman Desktop, and Podman Compose

Step 1: Install Podman

1. Download Podman from the official website.
2. Follow the installation instructions specific to your operating system (Windows, macOS, or Linux).

3. Verify the installation by running:



```
podman --version
```

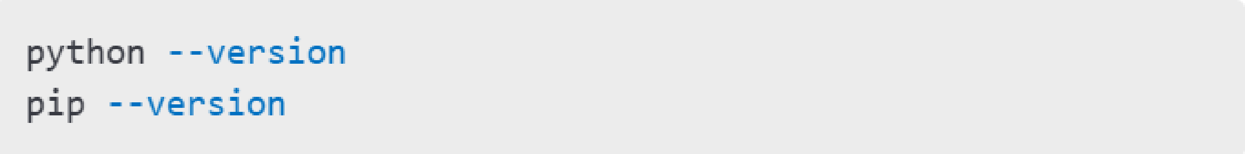
   - Take a screenshot of the Podman version.

Step 2: Install Podman Desktop

1. Download and install Podman Desktop from the official website.
2. Open Podman Desktop and ensure it is correctly configured by checking the connection status.

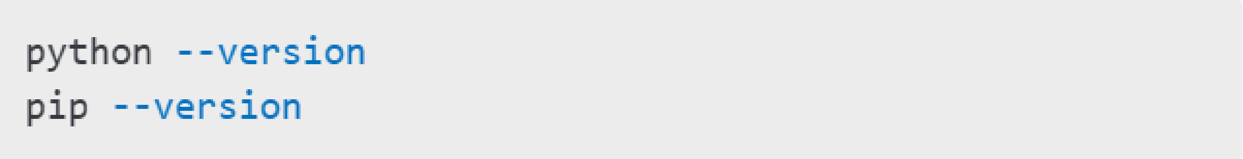
Step 3: Install Podman Compose

1. Check if Python and pip are installed by running:



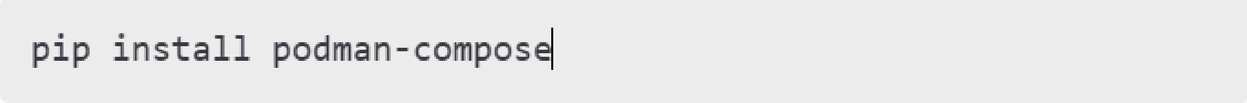
```
python --version
pip --version
```

2. If Python and pip are not installed:
   - Download Python from the official Python website.
   - Install Python and ensure the option to "Add Python to PATH" is selected.
   - Verify the installation by running:



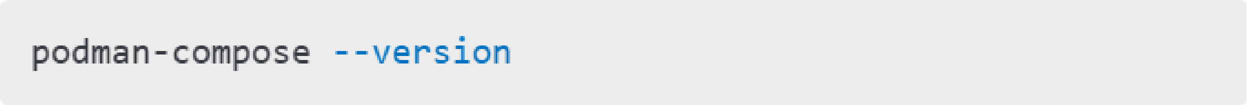
```
python --version
pip --version
```

3. Install Podman Compose using pip:



```
pip install podman-compose
```

4. Verify the installation by running:

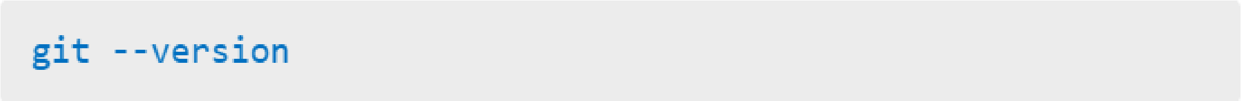


```
podman-compose --version
```

   - Take a screenshot of the Podman Compose version.

Task 5: Install Git and GitHub Desktop

1. Download and install Git from the official website.
2. Verify the installation by running the following command in a terminal:



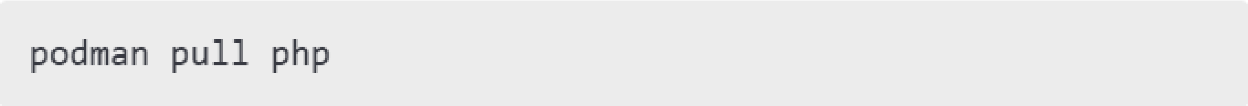
```
git --version
```

   - Take a screenshot of the Git version.

3. Download and install GitHub Desktop from the official website.
4. Log in to GitHub Desktop using your GitHub account credentials.
5. Clone a test repository or create a new one to confirm that GitHub Desktop is working correctly.

Task 6: Verify Podman with a Sample Container

1. Pull the latest PHP image from Docker Hub:



```
podman pull php
```

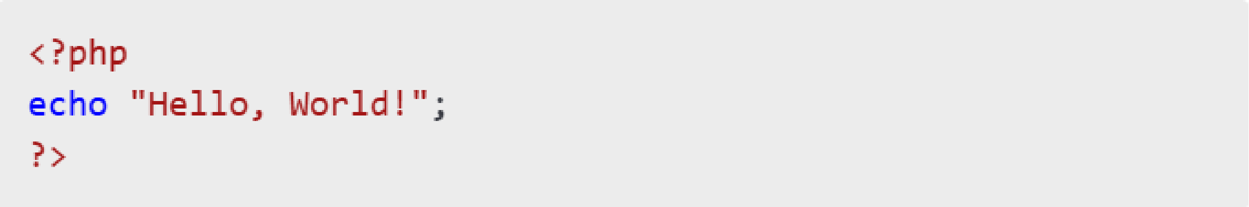
2. Run a PHP container and mount a local directory for testing:

podman run --rm -v $(pwd):/usr/src/myapp -w /usr/src/myapp php php -r "echo 'Podman setup successful';"

3. Confirm the output is displayed as Podman setup successful.
   - Take a screenshot of the successful output.

Task 7: Write and Run a "Hello World" PHP Program

1. Create a new PHP file named hello_world.php in the XAMPP htdocs directory.
2. Add the following code to the file:



```
<?php
echo "Hello, World!";
?>
```

3. Open a web browser and navigate to http://localhost/hello_world.php.
4. Verify that the output "Hello, World!" is displayed on the browser.
   - Take a screenshot of the output.

Task 8: Commit Codes and Screenshots to GitHub

1. Create a new private repository on GitHub named php_dev_env_lab.
2. Add the following to the repository:
   - PHP files (info.php and hello_world.php).
   - Screenshots of:
     - PHP version.
     - Podman version.
     - Podman Compose version.
     - Git version.
     - Successful Podman test output.
     - "Hello World" program output.
   - A report summarizing the steps you followed during the lab.
3. Ensure the repository contains a README.md file that briefly describes the lab.
4. Make sure the GitHub repository is **private**.
5. Add the instructor as a collaborator to the repository. The instructor's GitHub username is **vbogudskyi**.
6. Push all files to the GitHub repository using Git or GitHub Desktop.

Task 9: Submit GitHub Link

1. Copy the repository link from GitHub.
2. Submit the GitHub link on Blackboard as your lab submission.

Submission Guidelines

1. Submit only the GitHub repository link on Blackboard by the specified due date.
2. Ensure the repository contains:
   - All required PHP files.
   - Screenshots as specified.
   - A README.md file with a summary.
   - A PDF report detailing the steps you followed during the lab.

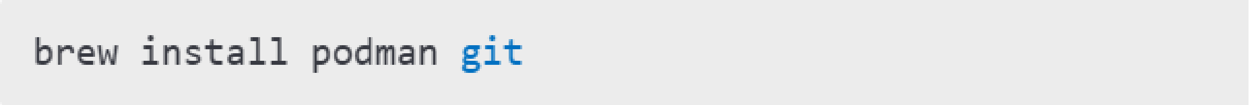
3. Ensure the instructor has access to the repository by adding them as a collaborator.

Evaluation Criteria

1. **Completeness (50%)**: All tasks are completed as per instructions, and files are present in the GitHub repository.
2. **Accuracy (30%)**: PHP programs, screenshots, and report are accurate and functional.
3. **Documentation (20%)**: The README.md file and PDF report are clear and detailed.

Additional Notes

1. For **Mac users**:
    - Use MAMP instead of XAMPP for PHP and MySQL. Download it from MAMP's official website.
    - Use Homebrew to install Podman and Git by running:



```
brew install podman git
```

    - Ensure the PHP executable path in VS Code is set to the PHP path in MAMP.
    - Follow the same steps for GitHub Desktop installation.
2. If you encounter any issues, consult the official documentation or seek help during lab hours.
3. Ensure all software installations are done on your personal computer or development machine.
4. This lab setup will be used in subsequent labs and projects, so ensure everything is correctly configured.
5. Late submissions will be subject to the course's late submission policy.