Laboratory Activity
Web Systems and Technology
Midterm

OFILANDA, Marc Dave U.
C334_IT
WSTL31C
October 30,2003
Activity 3

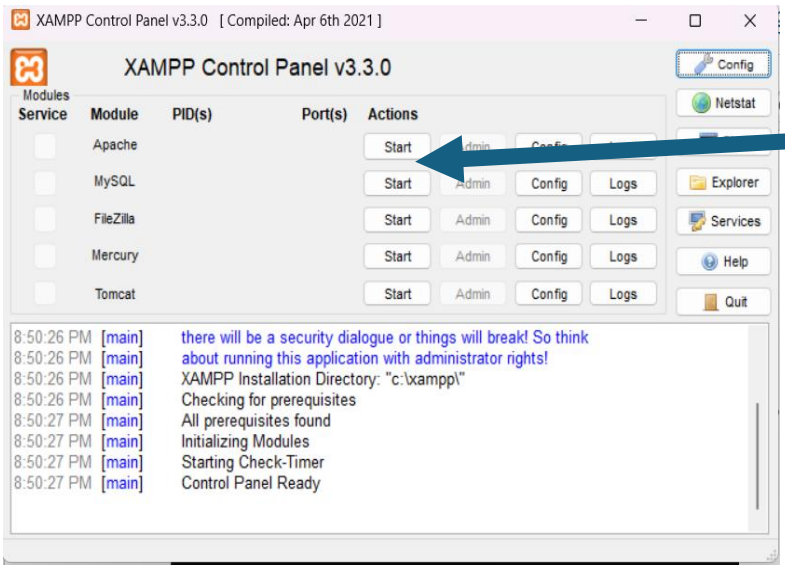
Procedure:

Step 1: Install a local server environment
Since phpMyAdmin requires a web server (Apache), a database server (MariaDB/MySQL), and PHP to run, you will need to install a bundled software package.
Popular options include:

- **XAMPP:** Works on Windows, macOS, and Linux.
- **WAMP:** For Windows users.
- **MAMP:** For macOS users.

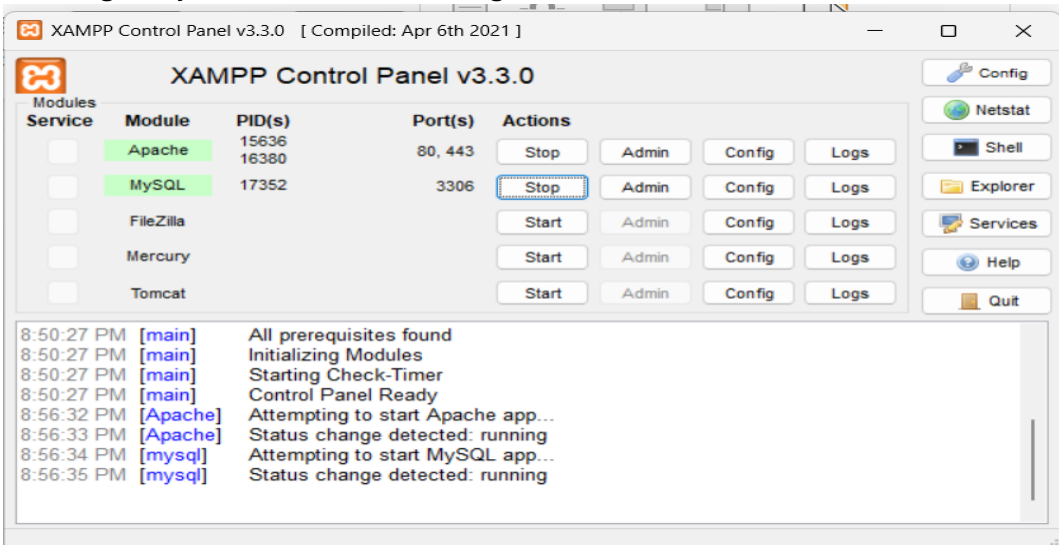
Step 2: Start the required services
After installing your server environment, you need to open its control panel and manually start the services.
For XAMPP, this means:

1. Open the **XAMPP Control Panel**.
2. Click the **Start** button next to **Apache**.
3. Click the **Start** button next to **MySQL**.
4. Ensure that both modules are running successfully.

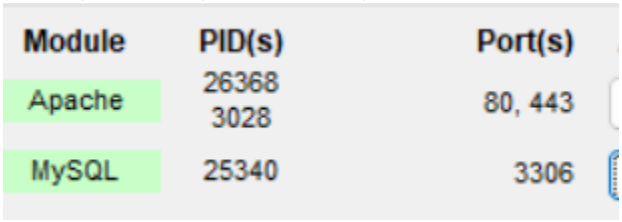


Make sure to click the Start button to open the Apache and MySQL

- Once it green your server is now running



- Make sure to check the ports of Apache and MySQL as follows

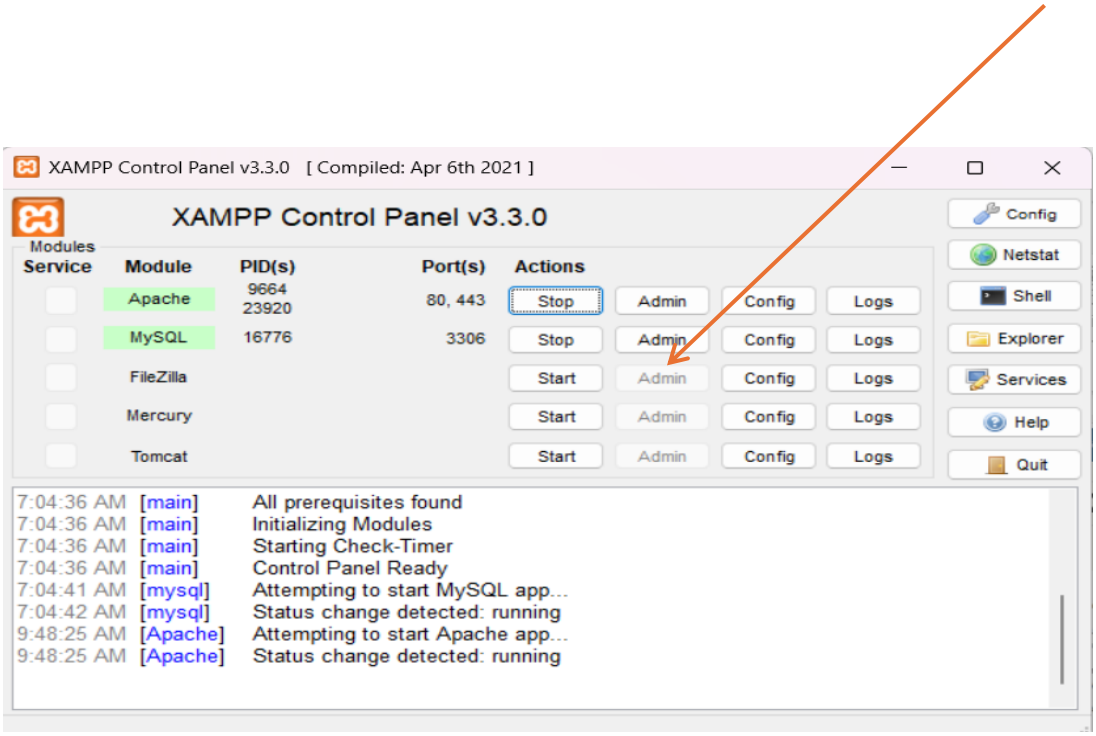


- Your now ready to test your server

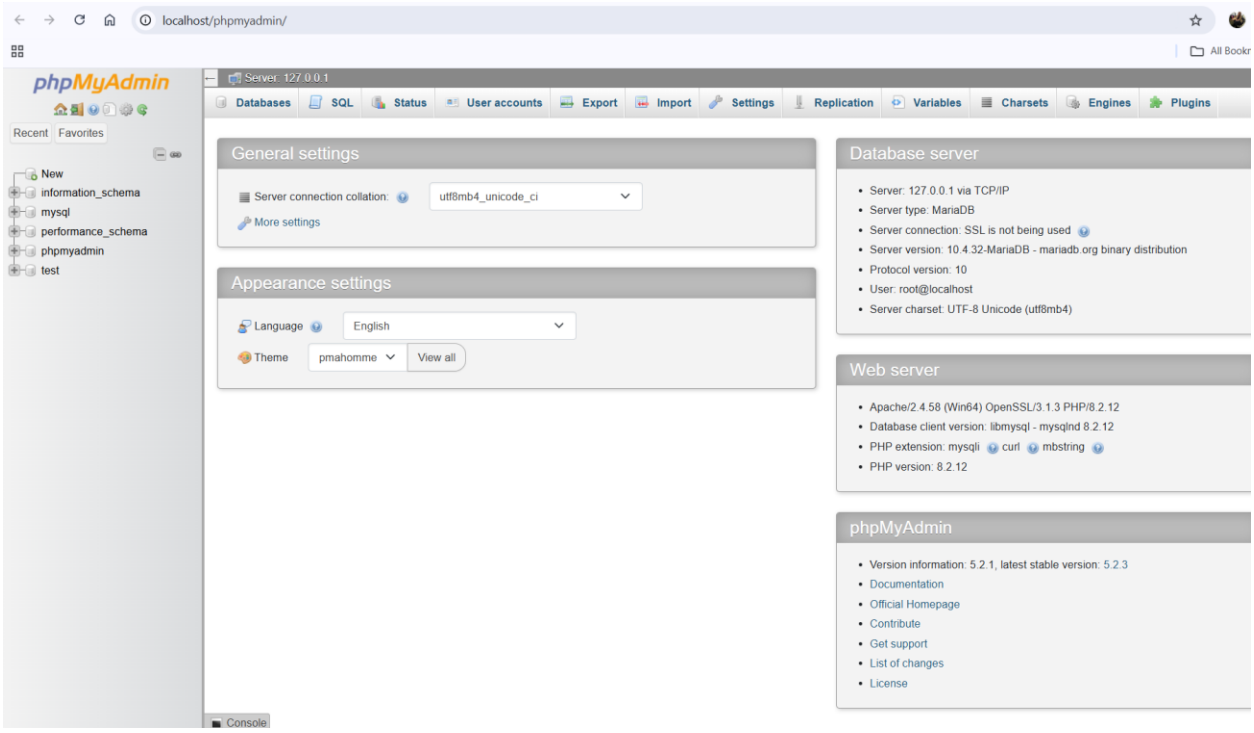
Step 3: Access phpMyAdmin

Once your servers are running, you can access the phpMyAdmin interface through your web browser.

1. Open your web browser.
2. Type http://localhost/phpmyadmin in the address bar and press Enter.
3. Log in with your credentials. If you are using a new installation, the default username is often **root** with **no password**.



Then it will redirect you to the admin panel



Step 4: Plan your database schema

Before creating your database, consider what information you need to store and how it should be organized. Proper planning helps avoid structural problems later on.

A simple planning process involves:

- **Identifying entities:** Figure out the main "things" your database needs to track, like Users, Products, or Orders.
- **Defining attributes:** Decide on the specific data points for each entity. For a Users table, this might include first_name, last_name, and email.
- **Specifying primary keys:** Choose a unique identifier for each table, such as a user ID. This is typically an auto-incrementing integer.
- **Establishing relationships:** Determine how your tables will connect to each other. For example, the Orders table might link to a Users table.
  This time:
    - I want you to explore or try to create a sample database as your activity for today without connecting it to php (or to your website)
    - Name it on the desired table name.
    - Screenshot all your works and output in this document and sent it through your github (filename: Activity 3 -10/30) this is a different filename from your Activity 2.

Answer:

users table contains the columns **id, name, email, password_hash, role, and registration_date**. It is located inside the fridget_db database, which is the project I created before**.**