

PREREQUISITES

Before you begin, make sure you have the following software installed:

XAMPP: You can download it from the official Apache Friends website (<https://www.apachefriends.org/index.html>).

INSTALLATION AND SETUP

Follow these steps carefully to set up the project.

Step 1: Download the Project Files from GitHub. Open your computer's Terminal (on Mac) or Git Bash (on Windows).

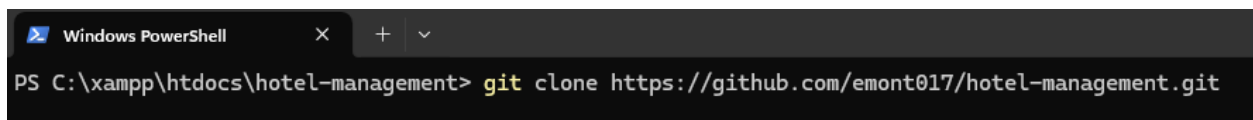
Navigate to a folder:

On Windows, C:\xampp\htdocs\

On macOS, /Applications/XAMPP/xamppfiles/htdocs/

Clone the repository by running this command:

```
git clone https://github.com/emont017/hotel-management.git
```

A screenshot of a Windows PowerShell terminal window. The title bar shows 'Windows PowerShell' with a close button and a dropdown arrow. The command prompt shows the current directory as 'PS C:\xampp\htdocs\hotel-management>' and the command being executed is 'git clone https://github.com/emont017/hotel-management.git'.

This will create a new folder named hotel-management with all the project files inside.

Step 2: Configure the Local Domain- Instructions for Mac & Windows below:

Step 2.1: Configure Apache (Mac)

1. In the XAMPP Control Panel, Stop the Apache service if it is running.
2. Use Finder to go to your Applications folder, open XAMPP, and find the file at the following path:
/Applications/XAMPP/xamppfiles/etc/extra/httpd-vhosts.conf
3. Add this code block to the very end of the file:

```
<VirtualHost *:80>
```

```
    DocumentRoot "/Applications/XAMPP/xamppfiles/htdocs/hotel-management/public"
```

```
    ServerName hotel.local
```

```
<Directory "/Applications/XAMPP/xamppfiles/htdocs/hotel-management/public">
```

AllowOverride All

Require all granted

</Directory>

</VirtualHost>

4. Save and close the file.

Step 2.2: Configure Mac Hosts file

1. Open Terminal

2. Open hosts file with the following command:

```
sudo nano /etc/hosts
```

3. Add the following line at the bottom of the file:

```
127.0.0.1 hotel.local
```

4. Save and exit.

Step 2.3: Configure Apache conf file

1. Open the main Apache configuration file, httpd.conf:

```
/Applications/XAMPP/xamppfiles/etc/httpd.conf
```

2. Find the following line in the file:

```
#Include etc/extra/httpd-vhosts.conf
```

3. Remove the # symbol from the beginning of that line. It should now look like this:

```
Include etc/extra/httpd-vhosts.conf
```

4. Save and close the httpd.conf file.

Step 2.1: Configure Apache (Windows)

1. In the XAMPP Control Panel, Stop the Apache service if it is running.

2. Open the following file in a text editor: C:\xampp\apache\conf\extra\httpd-vhosts.conf

3. Add this code block to the very end of the file:

```
<VirtualHost *:80>
```

DocumentRoot "C:/xampp/htdocs/hotel-management/public"

ServerName hotel.local

<Directory "C:/xampp/htdocs/hotel-management/public">

AllowOverride All

Require all granted

</Directory>

</VirtualHost>

```
# Virtual Hosts
#
# Required modules: mod_log_config
#
# If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them. Most configurations
# use only name-based virtual hosts so the server doesn't need to worry about
# IP addresses. This is indicated by the asterisks in the directives below.
#
# Please see the documentation at
# <URL:http://httpd.apache.org/docs/2.4/vhosts/>
# for further details before you try to setup virtual hosts.
#
# You may use the command line option '-S' to verify your virtual host
# configuration.
#
# Use name-based virtual hosting.
#
##NameVirtualHost *:80
#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
# The first VirtualHost section is used for all requests that do not
# match a ##ServerName or ##ServerAlias in any <VirtualHost> block.
#
##<VirtualHost *:80>
#    ##ServerAdmin webmaster@dummy-host.example.com
#    ##DocumentRoot "C:/xampp/htdocs/dummy-host.example.com"
#    ##ServerName dummy-host.example.com
#    ##ServerAlias www.dummy-host.example.com
#    ##ErrorLog "logs/dummy-host.example.com-error.log"
#    ##CustomLog "logs/dummy-host.example.com-access.log" common
##</VirtualHost>
#
##<VirtualHost *:80>
#    ##ServerAdmin webmaster@dummy-host2.example.com
#    ##DocumentRoot "C:/xampp/htdocs/dummy-host2.example.com"
#    ##ServerName dummy-host2.example.com
#    ##ErrorLog "logs/dummy-host2.example.com-error.log"
#    ##CustomLog "logs/dummy-host2.example.com-access.log" common
##</VirtualHost>
#
<VirtualHost *:80>
    DocumentRoot "C:/xampp/htdocs/hotel-management/public"
    ServerName hotel.local
    <Directory "C:/xampp/htdocs/hotel-management/public">
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

4. Save and close the file.

Step 2.2: Configure Windows 'hosts' File

1. Open Notepad as an Administrator
2. In Notepad, go to File > Open and navigate to this exact path:

C:\Windows\System32\drivers\etc\hosts

3. Go to the last line of the file and add this new line:

```
127.0.0.1 hotel.local
```

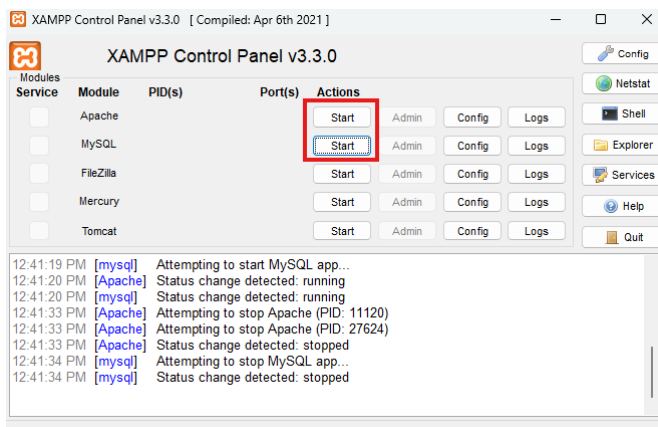
```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#     102.54.94.97     rhino.acme.com   # source server
#     38.25.63.10     x.acme.com      # x client host
#
# localhost name resolution is handled within DNS itself.
#
#     127.0.0.1       localhost
#     ::1             localhost
127.0.0.1 hotel.local
```

4. Save and close the file.

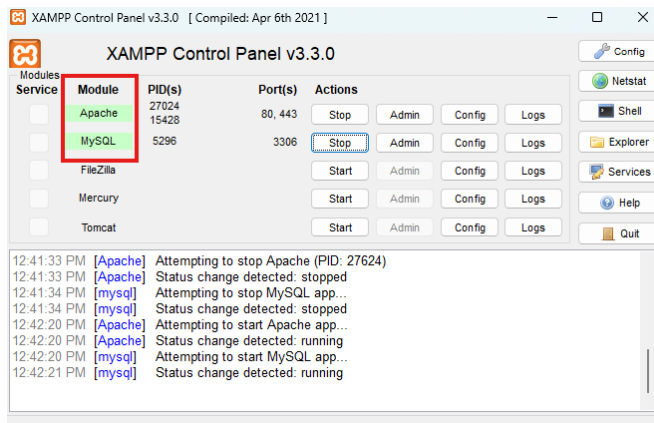
Step 3: Start Your Local Server with XAMPP

Open the XAMPP Control Panel.

Click the "Start" button next to both Apache and MySQL.



Both should turn green, indicating they are running correctly.



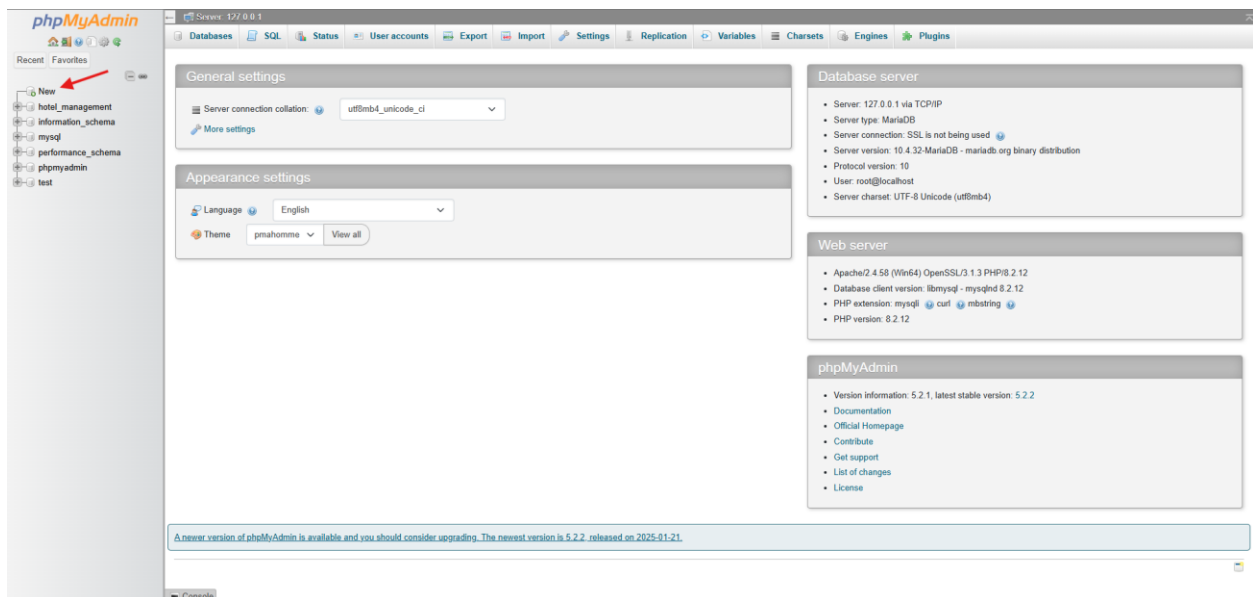
Step 4: Set Up the Database with phpMyAdmin

This is the most important step. The website will not work without the database.

Open your web browser and go to this address: <http://localhost/phpmyadmin>

Step 4.1: Create the Database:

On the left sidebar, click on "New".



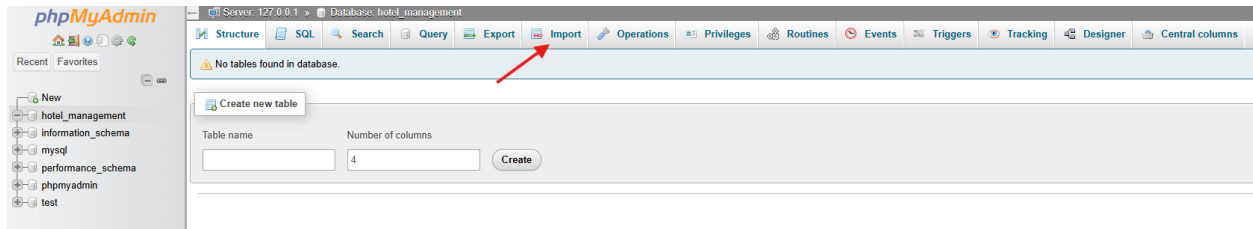
In the "Database name" field, type exactly: `hotel_management`

Click the "Create" button.

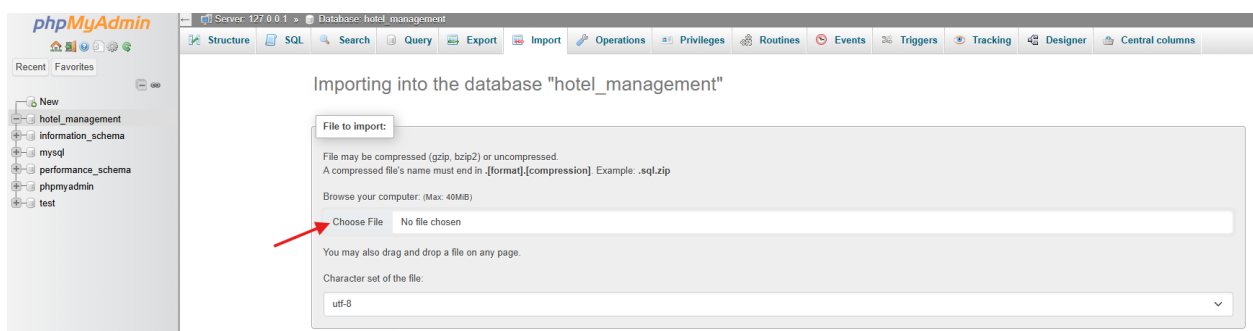
Step 4.2: Import the Database Structure:

In the left sidebar, click on the hotel_management database you just created.

Click on the "Import" tab at the top of the page.

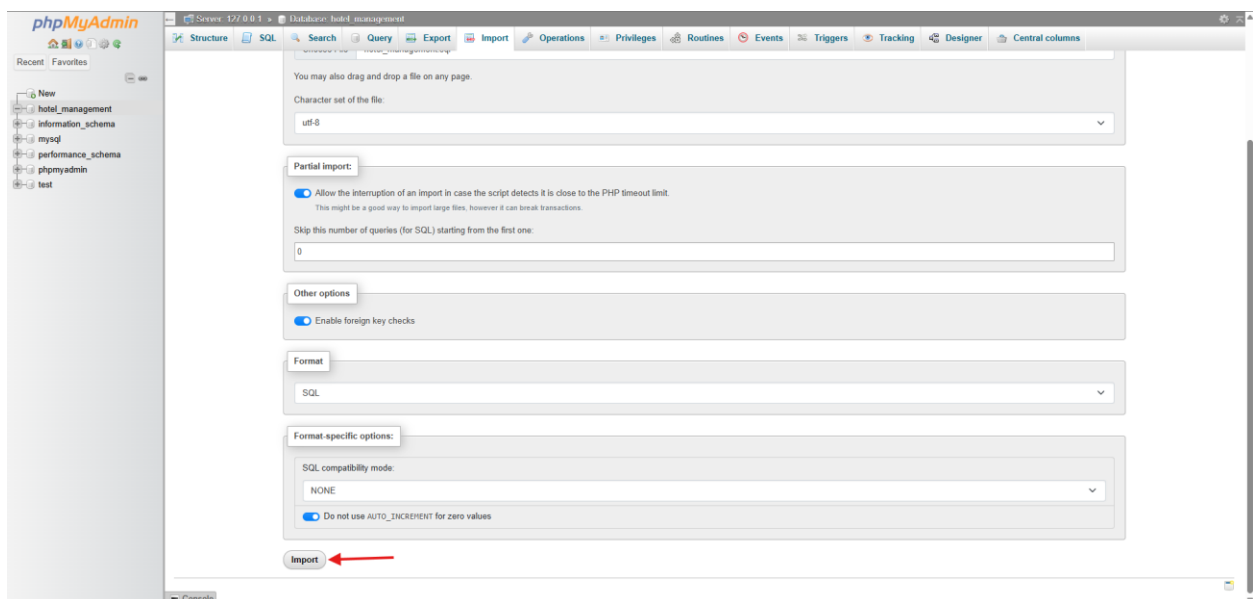


Under the "File to import" section, click the "Choose File" button.



Navigate to the project folder you downloaded from GitHub (hotel-management). Inside that folder, go into the db_backup folder and select the .sql file.

Scroll to the bottom of the page and click the "Import" button.



If it's successful, you will see a green message and your new tables will appear in the sidebar under hotel_management.

Step 5: Restart Apache

1. Go back to the XAMPP application.
 2. Restart the Apache server.
 3. You can now access the entire project by navigating to: <http://hotel.local/>
-

Default Admin Login:

User: Test2

Password: Test123

Default Housekeeper Login:

User: Housekeeper1

Password: Test123

Default Accounting Login:

User: Accountant1

Password: Test123

HOW TO SAVE AND PUSH YOUR WORK TO GITHUB

When you make changes to the code, you need to save them back to GitHub so the rest of the team can see them.

Step 1: Add and Commit Your Changes

After you've saved your changes in your code editor, open your Terminal or Git Bash.

Navigate into the project folder:

On Windows, C:\xampp\htdocs\hotel-management

On macOS, /Applications/XAMPP/xamppfiles/htdocs/hotel-management

Step 2: Add all the changed files to be saved:

git add .

Step 3: Commit (save) the changes with a descriptive message:

```
git commit -m "Describe the change you made"
```

Step 4: Push Your Changes to GitHub

Run the push command:

```
git push
```

Authentication: GitHub no longer accepts your regular password here. You must use a Personal Access Token (PAT).

You will be prompted for your username and password.

Username: Enter your GitHub username.

Password: PASTE your Personal Access Token here. Do not type your GitHub password.

If you don't have a token, you can generate one in your GitHub Developer Settings (<https://github.com/settings/tokens/new>). Make sure to give it the "repo" scope.