

Build Instructions for Online Health Management System

First we need to begin with an Ubuntu server. Our team used Ubuntu 14.04, the most recent release with long term support. Our server is a virtual private server running on Microsoft's Azure cloud service, but there are many other providers such as Amazon, Linode, and DigitalOcean that all have attractive offers. If you want to host it yourself, you could also run it inside a virtual machine on your computer.

Here are the steps to setting up the LAMP stack (Linux, Apache, MySQL, PHP) on the server. I used the instructions detailed here <https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-14-04>

I will summarize the actions needed to take here:

1. Log on the server via SSH
2. To update the package list on the server, run

```
sudo apt-get update
```

3. Next, we will install Apache. Run this command:

```
sudo apt-get install apache2
```

4. To install and configure MySQL, Run these commands:

```
sudo apt-get install mysql-server php5-mysql
```

```
sudo mysql_install_db
```

```
sudo mysql_secure_installation
```

5. Now we will install php:

```
sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
```

Now we will set up the database with all of the required tables.

1. Access the MySQL shell as root by using this command

```
mysql -u root -p
```

2. Create a database called *project*:

```
CREATE DATABASE project;
```

3. Now we need to specify that our actions will be for this database. Enter:

```
USE project;
```

4. Lets create the table to store user account information:

```
5. CREATE TABLE IF NOT EXISTS `users` (  
6. `user_id` mediumint(8) unsigned NOT NULL AUTO_INCREMENT,  
7. `email` varchar(30) NOT NULL,  
8. `username` varchar(16) NOT NULL,  
9. `fname` varchar(20) DEFAULT NULL,  
10. `lname` varchar(20) DEFAULT NULL,  
11. `doctorName` varchar(16) DEFAULT NULL,  
12. `isDoctor` char(1) NOT NULL,  
13. `hashPassword` char(60) NOT NULL,  
14. PRIMARY KEY (`user_id`),  
15. UNIQUE KEY `email` (`email`),  
16. UNIQUE KEY `username` (`username`)  
17. ) ;
```

17. Now we will create the table to hold perscription information:

```
18. CREATE TABLE IF NOT EXISTS `medicine` (  
19. `pill_id` mediumint(8) unsigned NOT NULL AUTO_INCREMENT,  
20. `assigned_to` varchar(16) NOT NULL,  
21. `assigned_by` varchar(16) NOT NULL,  
22. `medicine` varchar(30) NOT NULL,
```

```
23. `date` char(10) NOT NULL,  
24. `time` char(5) NOT NULL,  
25. `num_pills` tinyint(1) NOT NULL,  
26. PRIMARY KEY (`pill_id`)  
    ) ;
```

27. This table will hold our appointments information:

```
28. CREATE TABLE IF NOT EXISTS `appointments` (  
29. `appt_id` mediumint(8) unsigned NOT NULL AUTO_INCREMENT,  
30. `patient` varchar(16) NOT NULL,  
31. `doctor` varchar(16) NOT NULL,  
32. `date` char(10) NOT NULL,  
33. `time` char(5) NOT NULL,  
34. PRIMARY KEY (`appt_id`)  
    ) ;
```

35. This table will hold the conversations between our users and doctors

```
36. CREATE TABLE IF NOT EXISTS `chat` (  
37. `chat_id` mediumint(8) unsigned NOT NULL AUTO_INCREMENT,  
38. `date` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
39. `sent_by` varchar(16) NOT NULL,  
40. `sent_to` varchar(16) NOT NULL,  
41. `message` varchar(1000) NOT NULL,  
42. PRIMARY KEY (`chat_id`)  
    ) ;
```

43. This table allows us to turn on and off the SMS alerts

```
44. CREATE TABLE IF NOT EXISTS `alertSwitch` (  
45. `switch` char(1) NOT NULL  
    ) ;
```

We will need to manually add a row to this table for it to be functional. To add this row, enter:

```
INSERT INTO alertSwitch (switch) VALUES ('0');
```

Now we will copy over the PHP sources to the webserver. Use an SFTP client like [FileZilla](#) to connect to the server and navigate to the directory `/var/www/html`

Copy over the source files into this directory. This directory is what Apache uses for their web root by default. This means that you any file will be accessible via HTTP at your server's IP address. If you have the server installed on your local machine, it will be available in your web browser at <http://127.0.0.1/>.

The SMS alerts are off by default. If you would like to turn them on, run this command in the terminal (SSH):

```
php -f /var/www/html/onoff.php
```

To turn back to the off state, just run the same command again.

Now we will create a symlink for the calendar to read our JSON events

```
ln -s /var/www/html/json /var/www/html/partials/json
```

Lastly, we need to make sure the permissions are set for our files

```
sudo chmod -R 777 /var/www/html
```

Now you have successfully installed our Online Health Monitoring System!

Template created by G. Walton (GWalton@mail.ucf.edu) on August 30, 1999 and last modified on August 15, 2000.

This page last modified by James Luke on 11-22-2014