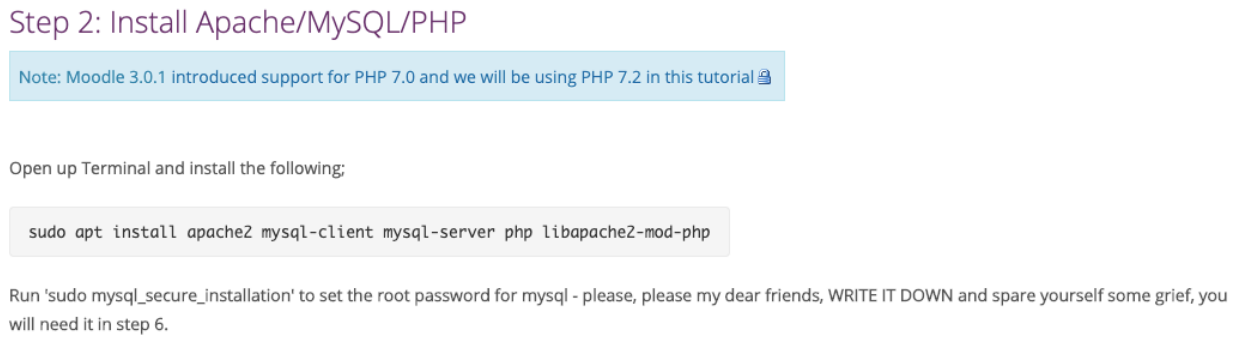
Deliverable 1

Our task was to get our project: Moodle, Up and running on localhost.
Thankfully moodle is very well documented, and we were able to find a step-by-step installation for moodle here.

Step 1: Install Ubuntu
Easy enough. We installed Ubuntu and we now have it running as a virtual machine. This allows us as a team to share the exact same distribution of Linux. Ultimately this will allow us to share files more easily when working on our moodle project.

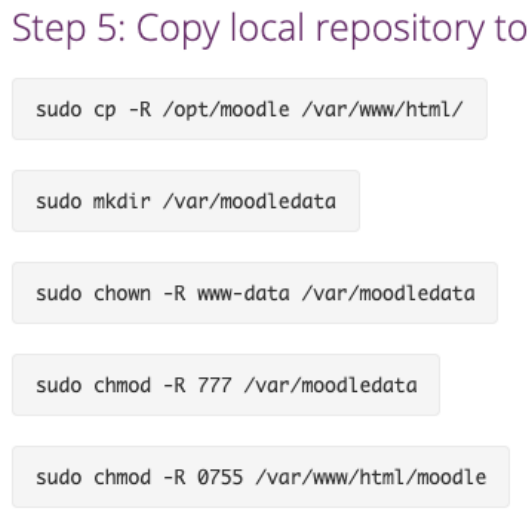
Step 2: Install Apache/MySQL/PHP
Moodle is written using Apache, MySQL, and PHP. Apache is a web server software that will allow us to run moodle on our localhost. Or if we wanted to, we have the ability to run moodle on a server, however, for testing, we will just stick with the localhost. We need MySQL as a database language so that moodle can keep track of the information we send to the site like class information, users, etc. And finally, we need PHP installed. Moodle was written mostly using PHP.



Step 3: Install Additional Software
This part of the installation is installing mostly PHP related tools. We also install Git here if we haven't done so already. We use git in order to download the most recent version of the moodle project.

Step 4: Download Moodle
This is where we download moodle. We first start by finding the most recent update of the moodle project, which is the git branch: MOODLE_37_STABLE

Step 5: Copy local repository to /var/www/html/
This is the most critical part of the installation process. Basically what we did is copy the Moodle project and clone it on to our virtual machine. From there we give the machine the proper permissions to be able to write into the files once the server (or localhost) is running. Also, Since we setup a local repository in the previous step, Having your local repository outside of the webroot, we will be able to prepare and stage upgrades in a more efficient manner.



Step 6: Setup MySQL Server
In this step, we need to change some of the settings in our MySQL database. In the new version of Moodle we need to set the default storage engine to innodb and change the default file format to Barracuda, this is a new setting compared to previous versions. This caused us many problems and we ended up having to delete our entire Moodle project and start over since we had the wrong version of Moodle which did not use Barracuda or innodb. In this step, we also need to create the Moodle database and the Moodle MySQL User with the correct permissions. After setting up our user account with the correct permissions, our Moodle MySQL database was ready to go.

Step 7: Complete Setup
In this step, we were pretty much finished. All we needed to do was go to our localhost/moodle in a web browser and complete the initialization.

Conclusion: Working with Moodle was initially frustrating. We sunk in many hours just trying to get moodle up and running. Fortunately, we were able to figure it out and get it to come up as website on our localhost. Our plan of attack for next time is figuring out how to run tests on our Moodle site. Since Moodle is used on very large servers, there are protections put into place so that a Admin can't accidently crash the site. Even though we have created a test environment for our moodle, we need to create a test environment inside of our moodle project so that it gives us the permission to run the tests.