

Assignment 1 COSC349
‘World Time Zone Converter’

Ben Highsted (5536340)
Matthew Neil (7388248)

For our multiple VM assignment, we decided to create a world time zone converter, where you can convert your current time to see what time it currently is for different places around the world. We used Apache and MYSQL for our project, and three different VM's all running Ubuntu. Because there is such a large amount of time zones around the world, we decided to pick a handful of different time zones and just work with those. Also to save time, we have assumed the users will be running the website in New Zealand, although this is a very easy thing to change for other developers in other countries.

Our first virtual machine is a web server (Apache) which runs a simple PHP page to allow a user to interact and choose a time zone they wish to convert their time to. It will then display the converted time zone on the screen.

The second virtual machine is a SQL server, which is used to store the users preferences for which timezone they picked last. This is for the convenience of the user, as if they only want to convert to one time zone each time they come back the first virtual machine will query the database, and find out which one they used last.

Our third VM is another Apache web server, which runs a simple PHP script. It takes a query in the website link, and will then produce the output onto a web page, which our first VM can easily obtain to display to the users.

The download volume for the first build is approximately 270MB for the Ubuntu xenial box install and 184 MB for the MYSQL database and Apache installs. Sequential builds will be approximately 180-200MB for MySQL database and Apache as well as updates. Depending on your computer, the build time can be anywhere from 5 minutes to 8 minutes long.

The users can simply download/clone the files from github, and the only command necessary for the build is 'vagrant up'. No other manual interaction is required for the build process. Our program is built around very basic user interaction, so we have not included any test data.

To use the application, the user will need to navigate to localhost:8080 on the machine running the virtual machines, which will bring up the web interface. It will then display your current time, and give you a selection of options to convert your time to. You select the option, and click the 'convert' button, and it will refresh the page now displaying the converted time. Here is a video showing the basic function of our VM's:

<https://www.youtube.com/watch?v=8-Jnc5XlpmM&feature=youtu.be> (45 seconds long).

A developer may want to modify the way our virtual machines currently work. They can do this by either editing 'Vagrantfile', 'www/index.php' for the webserver default web page and/or database-tz-setup.sql for the initial database table setup. They may edit the database to hold more information and then edit 'index.php' to display that new information on the website.

A developer may also want to add more time zones than we currently have, or change which ones are being used, which to do this they would simply need to go to /www/converter/index.php and edit the index.php script to add more timezones.

They might also want to change the port forwarding on 'webserver' and/or 'tzconverter' incase their current machine already has the ports 8080 (webserver) and 8081 (tzconverter) in use. They can change this in "Vagrantfile" and change the "forwarded_port" line where it says "host: 8080" to a different port number that isn't in use. If they change the port number to '8082' for example, they will have to access the webserver website using "localhost:8082".

When a developer has implemented their modification they can ether type into the terminal 'vagrant destroy' and press 'y' to confirm destroying each current VM and then type 'vagrant up' to start the virtual machines again. When changing the PHP files, they will update automatically so they won't need to restart the virtual machines.