Docker and Vagrant are both similar and different in their implementations. To begin with both of them are configuration software, but their integration is different. Vagrant is much more focused and designed to be implemented with web based applications, docker is more suited for desktop applications and is designed as such. Both are mostly implemented and accessed through the command line. Vagrant’s implementation is done through creating VMs (the lectures did it by making it make a Ubuntu VM) that can also be accessed through tools like VirtualBox and acts more like a wrapper, while Docker’s VMs are similar to VMs, but not exactly VMs. They both have integration with tools such as Puppet, Chef and Ansible. They are both open source, but are run by commercial companies and can be run on Windows, Mac OS and Linux. Vagrant (at least according to you guys) also doesn’t work well when run on VMs, Docker does work well on VMs. Though Vagrant used to run on VMs according to the lectures. Vagrant instead is designed to be run in tandem with VirtualBox or other VM/virtualization software. Vagrant also requires that it needs to be connected to the services needed to run it over SSH, whereas Docker does not and after installation, can be run offline and without connecting to the main site. Docker and Vagrant can also interface with version control systems as well.

Now on which of the two pieces of software I prefer. To be honest I prefer Docker over Vagrant, this is for the simple reason as I have little experience with Web development, it is more commonly used for testing and development by most companies. The containers are also much easier to test on than the instances Docker provides as well. It also does provide the connection to other services without the pseudo VMs that Vagrant provides. I also found the documentation provided by Docker and the resources available in general much easier to parse and learn from than Vagrant’s. And lastly, a configuration management tool to upload documents to tools such as wordpress seems a bit overcomplicated and adds an unnecessary middleman in the form of Vagrant itself when publishing documents online, unless said document needs to be pushed at a certain time, even though it would probably need a scheduling tool in addition to Vagrant in order to do it. This is especially that was how it was stupidly difficult to even connect the VM once generated to wordpress, setting up the VM was extremely easy, but I had to go through 3 separate VMs until I found the “right one.” The middleman analogy is extremely apt in this case as vagrant is setting up a series of middlemen software between Virtualbox and the web software that you want to use when it would be easier to just set them all up separately, whereas docker containers are easy to operate and setup, and while they do not have all the features that a normal VM does, which Vagrant does provide, it is much easier to integrate it with other software. My experience with Vagrant was personally the worst I have ever experienced with a piece of software I have ever used, it resisted any attempts to connect it with SSH or any other means and the tutorials that you gave me were useless and even when I did, I feel you will give me a C or lower because it wasn’t the perfect, magical way that you wanted. Therefore, while I do find Vagrant a bit easier to set up, I do find its usage a bit more convoluted and more specialized than what is provided by Docker.