

## Exercise 7

---

 [coursys.sfu.ca/2021fa-cmpt-470-d1/pages/Exercise7](https://coursys.sfu.ca/2021fa-cmpt-470-d1/pages/Exercise7)

For this exercise, you will “deploy” the tutorial code you wrote in [Exercise6](#) with either Vagrant + Virtualbox or with Docker + Docker Compose. You should work with the same GitLab repository.

Once again, I will point out: it's usually much easier to install the Ubuntu packages or packages with your language's tools (Gems, PyPi/pip, etc) for the software you need (not manual installations, getting Git repository code, etc.). That way, the package manager will get all of the required libraries for you and everything will mostly just work.

If you haven't already, follow the [ProjectRepo](#) instructions to add the template repository remote and merge it into your project. (Even if you have been working in a repository, you can follow the instructions from `git remote add template .`)

### Option 1: Vagrant VMs

---

Create configuration management recipes (with Chef, Puppet, or Fabric) that get your tutorial code running in a VM. This might require forwarding another port in the `Vagrantfile`, depending how you deploy. Feel free to work with others to overcome technical hurdles: it's not possible to give instructions for (or even for us to be a lot of help with) every possible combination of technologies.

For some instructions on how to get common frameworks running, see the [Development-Mode Configuration](#) instructions. Video: [Development mode deployment of a web app](#)

Test your site as it is deployed, and make sure everything is working.

If your configuration is working, you should be able to...

```
vagrant destroy  
vagrant up
```

... and access the URL on localhost to see the project. (That's basically what we will be doing, so it had better work.)

### Option 2: Docker Containers

---

Create Docker Compose configuration to get your tutorial code running in a container (or multiple containers). Feel free to work with others to overcome technical hurdles: it's not possible to give instructions for (or even for us to be a lot of help with) every possible combination of technologies.

For some instructions on how to get common frameworks running, see the [tech instructions](#) for the course. For this

Test your site as it is deployed, and make sure everything is working. If your configuration is working, you should be able to...

```
docker-compose down && docker system prune -f  
docker-compose build && docker-compose up
```

... and access the URL on localhost to see the project. (That's basically what we will be doing, so it had better work.)

## Submission

---

In your project's `README` (or `README.md` or whatever) include the URL where we can find your project after starting the VM/containers, probably something like <http://localhost:8000/>. Also include any usernames/passwords required to access the demo site.

Commit everything and create a Git tag, as you have before. Submit your Git tag through the CourSys activity [Exercise 7](#).

Updated Mon Aug. 30 2021, 07:36 by ggbaker.