02 – FarmData2 Developer Install
Activities

COMP190 – Tools and Techniques for Software Development
Dickinson College

Name:

As you get deeper into a project and begin to work with the application and make changes to the code it will be necessary to do a *developer installation*. A developer installation includes all of the code and tooling necessary to build, run, modify and test the application. In this set of activities, you will be performing a developer installation of FarmData2. At the completion of the activities in this section, you will have a functional FarmData2 development environment. You will then use that environment to work with FarmData2 to gain an understanding of what it does and how to use it.

Getting Started:

When joining a project the README is almost always the best place to start. In GitHub the contents of the README.md file are displayed at the bottom of the project's home page. Use the information in FarmData2's README (and documents to which it links) to complete the following activities.

1. What file in the FarmData2 repository provides information for those interested in contributing to the development of FarmData2?

2. Skim through the file you identified in question #1.

    a. Which section is most relevant to getting a running version of FarmData2 setup?

    b. To which file in the repository does the section you identified in part a refer you?

FarmData2 Zulip Install Stream:

3. The **Communications** section of the file you found in question #2 directs you to a Zulip stream dedicated to issues related to installing FarmData2. Answer the following questions to confirm that you have found this stream. Also, the information you find might be useful later as you do your own install.

   a. In the install stream, who posted the first message in the "Address in Use Error" topic?

   b. Based on the messages in that topic, what command can be used to resolve this issue?

If you encounter any difficulties or have questions while doing the developer install you should consult the *topics* in the **install** *stream* on Zulip. If you do not find your answer there, create a new *topic* in the *install stream* and ask a question. You should also monitor this channel during this activity and respond to any questions that you can help with. We are all members of the FarmData2 community and building up a record of problems encountered, and their fixes will help everyone.

FarmData2 Developer Install:

The **Development Platform** section of the file you found in question #2 gives the instructions for setting up the development environment for FarmData2. Because prior activities have been building toward this one you will have already completed some of the steps. The activities in the remainder of this section will guide you through the process of following the install documentation.

4. The document that you have found gives some recommendations for setting up a development environment using Linux in Virtual Box. You have already installed Linux in Virtual Box, but let's confirm that your setup is sufficient for working with FarmData2. Ensure that your VM is shut down. Then select your VM in the *VirtualBox Manager* (i.e. the window from which you start your VM) and clicking the using *Settings* icon. This will open the *Settings* dialog for your VM, which provides information about its configuration. The information you need to answer the following questions is provided in the different panes of this *Settings* dialog. You will need to explore those panes a little to find the relevant information.

a. How much *Base Memory* (i.e. RAM) have you allocated to your VM? If what you find does not meet the recommendations for a FarmData2 developer install, adjust the amount of Base Memory allocated to your VM and report the new setting here.

b. How much *Video Memory* have you allocated to your VM? If what you find does not meet the recommendations for a FarmData2 developer install, adjust the amount of Video Memory allocated to your VM and report the new setting here.

c. What is the Virtual Size of the virtual disk allocated to your VM? If what you find does not meet the recommendations for a FarmData2 developer install you can attempt to continue, but you may run into errors if the disk becomes full. Unfortunately, there is no easy way to adjust the size of a virtual disk once it is created. The recommendation here is to create a new Virtual Machine with a virtual disk of at least the recommended size. If you do so, report the size of the virtual disk on your new VM.

5. There are a number of **Prerequisites** that must be installed in order to complete the FarmData2 developer install. You may have some of these already installed, and you may need to install others. Ensure that you have all of the prerequisites installed and complete the table below. Hint: All of the prerequisite programs accept a `--version` flag that will display the version number.

Prerequisite	Installed Version

6. What group does your user need to be a member of in order to use the docker program? Give the output of a `groups` command that confirms that your user is a member of this group. If you are not a member of this group, use the **Linux Docker Configuration** section to fix the issue and give the updated output of a `groups` command here.

7. The section **Getting FarmData2** discusses how to get the FarmData2 code. Fork and clone the FarmData2 GitHub Repository and set an upstream remote to point to the main FarmData2 repository. Note: This should be the *real* FarmData2 upstream, not the fork that you used for earlier exercises in this class.

After you have forked, cloned and set the upstream, change into your FarmData2 directory and use the `git remote -v` command. The output from this command should indicate that the origin remote points to the FarmData2 repository in your GitHub space and the upstream remote points to the FarmData2 repository in the DickinsonCollege organization's space. If this is not the case, delete your FarmData2 directory and try again.

Give the full output of a `git remote -v` command showing the correct origin and upstream remotes.

8. Draw a picture showing the relationship between the FarmData2 upstream, your origin and your local repository. Your picture should clearly label each repository and show the fork, clone and remote relationships. Hint: You can refer back to the Version Control activities that had lots of diagrams like this if you need to.

9. Follow the instructions to **Install the Sample Database Image**.

   a. According to the instructions what should have happened when this command completes?

   b. Look at the files in the `docker` directory to confirm, as you described in part a, that you have installed the sample database image. If it has not installed, try again. When you have successfully installed the sample database image, give the full output from the Linux command you used to confirm that it has been installed.

10. Follow the **Starting FarmData2** section to run FarmData2. Then follow the first part of the **Logging Into FarmData2** section to confirm that FarmData2 is up and running. Log in as `manager1` and paste a screen shot of the page you see here.

11. FarmData2 runs in a number of docker containers that are all connected together by docker-compose. Use the docker commands you know to fill in the table below. There should be *one row for every image* with the "Repository" and "Tag" columns filled in. The "Container Name" column should be filled in for the rows that correspond to an image that was used to create one of the currently running containers. Hint: Review the docker commands from A12 to find the ones that will list all of the images or list all of the currently running containers.

Repository	Tag	Container Name

12. When you have finished working with FarmData2 you should shut it down before shutting down your VM. The section on Stopping and Starting FarmData2 describes how to shut down FarmData2 and restart it.

    a. What command is used to stop FarmData2.

    b. Give the output is generated when you run the script for stopping FarmData2?

13. **Optional but encouraged:** If you noticed any typos, missing steps or things that could be clarified in the INSTALL.md file create a feature branch, fix them, push it to your origin and create a pull request so that the upstream can be updated. Your contributions will improve the install process for everyone. Plus, if your changes are merged you'll rack up some GitHub contributions for your resume.

Using the FarmData2 Application:

The activities in this section will give you a little hands-on experience using the FarmData2 application. This will give you just a small feel for how it works and what it does. This will begin to prepare you for contributing to the development of FarmData2.

14. Restart FarmData2. What command did you use?

```

```

15. Log into FarmData2 using the credentials:
    - Username: `worker1` (or 2, or 3, or 4).
    - Password: `farmdata2`

16. The *FieldKit* and the *BarnKit* tabs contain the main features of FarmData2. Thus, far the features on these tabs support *Seeding Inputs* and *Seeing Reports*.

    a. A Seeding Report allows the farmer to retrieve information about the crops that have been planted (i.e. seeded) on the farm. These records can be searched and filtered in a variety of ways. The instance of FarmData2 that you are using has sample data in it for 2019 and the first half of 2020. Use the Seeding Report sub-tab within the BarnKit to answer the following questions.

    i. What crop(s) were planted (i.e. seeded) between April 13th and April 17th in 2020?

```

```

    ii. In which area(s) (i.e. fields) were turnips planted between March 1st and June 30th in 2020?

```

```

iii. What crops were planted in the CHUAU-2 area between March 1st and June 30th in 2020?

```



```

b. When a new crop is planted on the farm a new seedings record is created using the *Seeding Input* sub-tab of the *FieldKit* tab. Use the Seeing Input form to:
- create a new *tray seeding* that occurred on your birthday in 2020.
- Place your name in the comment field.
- You can plant whatever crop you like in whatever area you like.
- Make up values for the tray seeding and labor fields.

When you have successfully created your new seeding, generate a Seeing Report that shows it. Paste a screen capture of the report showing your new seeding here.

```



```

Optional: To help us improve and scope these activities for future semesters please consider providing the following feedback.

a. Approximately how much time did you spend on this activity outside of class time?

```



```

b. Please comment on any particular challenges you faced in completing this activity.

```



```