

01 – FarmData2 Developer Install

COMP290 – Large Scale and Open Source 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 in future activities to gain an understanding of what it does and how to use it.

The FD2School Repository:

For all of the FD2School activities you will use the FD2School-FarmData2 repository instead of the actual upstream FarmData2 Repository. The FD2School-FarmData2 repository is a fork of the FarmData2 repository created just for these activities. Using this fork allows you to practice interacting with the project repository as you learn, without mixing the learning activities into the main project history. It also ensures that the repository that you are working on will not change in unexpected ways as the maintainers continue to merge new contributions into the project. When you have completed the FD2School activities you will change over and begin working with the actual FarmData2 upstream repository to make your own contributions.

The FD2School-FarmData2 repository is located here:

<https://github.com/DickinsonCollege/FD2School-FarmData2>

Be sure to use this repository for all of your work on the FD2 School activities!

0. What is the URL of the repository that you should use for the FD2 School activities?

Getting Started:



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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 .md file (and documents to which it links) to complete the following activities.

1. FarmData2 uses three primary communication channels.

a. What are these three communication channels?

b. Follow the link to the *Communications document* to find out more about the communication channels and how they are used. Then answer the following questions.

i. Which channel would be most appropriate to use if you have questions about the installation process?

ii. Which channel would you use when you have specific code or documentation changes that you would like to contribute?

iii. Which channel would be most appropriate to use if you want to report a new bug that you have discovered in the project?

2. As a part of this assignment you will be installing FarmData2. What is the name of the file in the FarmData2 repository that provides information about installing FarmData2?

3. Explore the files linked in the Contributing to FarmData2 section and answer the following questions:



a. What two pledges are a part of the FarmData2 code of conduct?

b. Which open source license applies to the *code* in FarmData2?

c. Which license applies to all other content in FarmData2?

d. List three different types of contributions that can help the FarmData2 project.

e. What file in the repository contains information about the technologies used in the FarmData2 project?

Connect with the FarmData2 Community on Zulip:

4. Use the link in FarmData2's main README .md file to find the FarmData2 Zulip community.

a. If you do not already have a Zulip account, create one. What is your username on Zulip?

b. Join the FarmData2 community on Zulip, if you are not already a member. Then find the "Introduce Yourself" *stream* in the "#general" *topic* and post a short message introducing yourself to the community (if you have not already).

Give a link to your introductory message here. You can copy a link by clicking on the three dots in the upper right of your message near the timestamp.



The Install Stream:

Return to FarmData2's main README.md document and follow the link to the document that gives directions for installing FarmData2.

5. This document points you to a Zulip stream dedicated to questions and issues arising during the installation of FarmData2. Find that stream.

On what date was the *New Installation Process* merged into the main branch of FarmData2?

If you encounter any difficulties or have questions while doing the developer install you should consult the *topics* in the “#install” stream. 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.

The Developer Install:

The **Developer Install** section of the INSTALL.md file gives the instructions for setting up the *development environment* that you will use for working on the FarmData2 project. The development environment will include a running instance of FarmData2 as well as a Linux environment in which you will write and test code.

6. There are a number of **Prerequisites** that must be installed in order to complete the FarmData2 developer install. You may already have some of these installed from prior work. Others you may need to install now.

a. What are the three prerequisites that must be installed?

b. If you do not have git installed, follow the provided link for directions on how to install git. Once you have git installed, test the installation using the `git --version` command given in the directions. Give a screenshot of the command that you used and its output here.



c. If you do not have Docker Desktop installed, follow the directions for your machine (Mac, Windows or Linux). Once you have Docker Desktop installed, test the installation using the `docker run` command given in the directions. Give a screenshot showing the command that you used and its output at least to the point where the “Hello from Docker!” message appears.

d. If you do not have the TigerVNC viewer installed, download the installer for your machine (Mac, Windows or Linux). Once you have the TigerVNC viewer installed on your machine run it, choose the “About” option from the “TigerVNC Viewer” menu and give a screenshot of the dialog box that appears here.

7. Follow the directions in the **Getting FarmData2** to fork, clon and set the upstream remote for the FarmData2 repository.

As you follow the directions for Getting FarmData2 use the FD2School-FarmData2 repository rather than the URL given in the directions:

<https://github.com/DickinsonCollege/FD2School-FarmData2>

To check that you have done this correctly, change into the directory containing your FarmData2 repository and use the `git remote -v` command.

a. Give a screenshot of your `git remote -v` command and its output here.

b. The output you see in part a should indicate that both the origin and upstream remotes point to a repository with the name **FD2School-FarmData2**.



Do the remotes in your output from part a point to a repository with this name? If not, delete the repository you cloned and try question #7 again from the start. You will not be able to successfully go on until you have the proper origin and upstream remotes.

c. The output you see in part a should indicate that the `origin` remote points to the `FD2School-FarmData2` repository in your GitHub space (i.e. the URL contains your GitHub username) and the `upstream` remote points to the `FD2-SchoolFarmData2` repository in the DickinsonCollege organization's space (i.e. the URL contains `DickinsonCollege`).

Do your origin and upstream remotes point to the correct repositories? If not, delete your `FD2School-FarmData2` directory, and try question #7 again from the start. You will not be able to successfully go on until you have the proper origin and upstream remotes.

Starting FarmData2:

8. Follow the directions for starting FarmData2.

a. What command is used to start FarmData2?

b. Give a screen shot of the last 10 or so lines of output that appear when you run the command in part a. Be sure to include the last line that says "FarmData2 started."

9. A FarmData2 developer installation runs in five docker containers. These containers are all connected together by a virtual network so that they can communicate with each other. At times it may be helpful to know the names of these containers and the images that they came from.

With FarmData2 running, use the information in the Docker Desktop application to complete the table below. All of the FarmData2 containers have names that begin with the prefix `fd2_`. Also, recall that docker images names have the format: `repository/image:tag`



Container Name	Repository/Image	Tag

Connecting to the FarmData2 Development Environment:

10. Follow the directions for connecting to the Linux development environment via the TigerVNC Viewer. When you have successfully connected, and see a desktop environment like the one shown in the INSTALL.md directory go on to the next question.

Nothing required here. However, you must be connected to the development environment to continue.

Configuring git:

11. Using the Linux development environment, follow the instructions for configuring git. Once you have successfully configured git, use the `git config --global` command and give a screenshot of your command and its output here. You should see the configuration options that you set, and several others that were automatically set by the development environment.

Install the Sample Database Image:

12. Follow the directions for installing the sample database for FarmData2.

a. What is contained in the sample database? Why might this be useful?

b. Give a screenshot of the command that you used to install the sample database and its output here.



c. The text in this section describes the effect that your command should have had. Use a Linux command to display the files and directories in the docker directory to check that the database has been installed. Give a screenshot showing your command and its output here.

Logging Into FarmData2:

13. Log into FarmData2 as manager1. Give a screen shot of the browser showing the URL that you used and at least some of the contents of the page that is displayed.

Editing Code & Documentation:

Much of your work on FarmData2 will involve editing code and documentation files. The FarmData2 development environment provides a number of different code editors (vi/vim, nano, emacs, VSCodium). The VSCodium (the FOSS version of VSCode) is the most comprehensive IDE that is installed and is the recommended tool for working on FarmData2.

14. Follow the directions for opening VSCodium and the FarmData2 code. Expand the FarmData2 file tree in the “EXPLORER” part of the VSCodium window so that you can see README.md, INSTALL.md, etc. Paste a screenshot here showing the FarmData2 filetree open in the “EXPLORER” part of the VSCodium window.

Stopping and Starting FarmData2:

15. When you have finished working with FarmData2 you should shut it down so that it does not consume resources (e.g. RAM / CPU cycles) on your host machine. The section on Stopping and Starting FarmData2 describes how to shut down FarmData2 and restart it.

a. Should the command to stop FarmData2 be entered in the development environment or on your host machine?



b. What happens if you run the command to stop FarmData2 from within the development environment?

c. Stop FarmData2. Give a screenshot of the command that you used and its output.

16. Optional but encouraged: If you noticed any typos, missing steps or things that could be clarified in the `README.md` or `INSTALL.md` file follow the steps in the `CONTRIBUTING.md` file to 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 into the project 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.

17. Restart FarmData2. What command did you use?

18. Log into FarmData2 using the credentials:

- Username: `worker1` (or 2, or 3, or 4).
- Password: `farmdata2`

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



