**Final Presentation and Report assignment FP – Polish / Document / Present**

Assignment authored by Prof. Grant Braught,
with edits by John MacCormick

**Introduction:**

The goal of this assignment is to polish your current work, document the current status of the issue you are working on, and give a presentation summarizing your Epic project.

**Polish:**

Your goal should be to have a cohesive fully working polished feature or set of features that are ready to be merged into the upstream main. You are not required to fully complete your epic. Rather, you should have a collection of completed user stories that provide something useful to the end user. You should aim to have no half-finished stories or features that don’t quite work. If you have unfinished stories or features that don’t work, make sure they are in a branch that is separate from your PR.

You should also be sure to refine and revise your code.

* Make sure that it is neatly formatted (e.g. indented, etc). Use the autoformat feature of your IDE to make this easy.
* Check that your variables have well-chosen and consistent names that help the reader.
* Remove any commented-out or non-functional code. It may make sense to you now, but it will confuse a future programmer.
* Add useful comments. Use high-level comments to describe how you are doing things and possibly why. Do not over-comment. Assume the reader knows the programming languages and the framework. Don’t explain language/framework features.

**Document:**

You should fully document the status of your issue on GitHub. Add comments on your issue that explain the current state of the implementation. Explain what has been completed, what is in progress and what is yet to be done. Your Kanban board should help with this. Identify any significant challenges you have encountered and explain what you have tried that did not work. Link to your PRs as appropriate. For example, when describing what has been completed, link to the PR that is to be merged into the upstream main. When describing things that are in progress, link to a draft PR.

When creating this documentation on GitHub, it might be helpful to think about writing to your former self. Imagine you at the half-way point of this semester being assigned to pick up your issue in its current state. Ask yourself: What would be helpful for you to know? Have you written it in a way that you, at that point in time, would understand? That last one can be difficult. You are now very very familiar with the work. In fact, you know more about your part of FarmData2 than anyone in the world. So, be on the lookout for any assumptions you make in your writing that would not be clear to your former self.

**Present:**

Each team will have a 15-minute presentation slot during the final exam slot. Target your presentation for 10-12 minutes with 3-5 minutes for discussion and questions. All team members should participate approximately equally in the presentation. You may use any combination of presentation tools you choose (PowerPoint, screen shares, videos, live demos etc.).

The presentation should:

* Clearly define the issue that you were working on. Do not assume everyone else knows what your feature is or what it is supposed to do.
* Demonstrate what you have done. Indicate as appropriate what is complete and ready to be merged into the upstream main, and what is still work in progress. It is acceptable to demonstrate work in progress as a part of the presentation.
* Show an interesting part of your implementation and discuss the challenges that it presented and the solution you found.
* Describe what remains to be done and any significant challenges that you anticipate.

**Moodle submission:**

Before the start of the final exam slot, submit to Moodle a progress report of 100-200 words, describing your personal contribution to the team’s progress on its Epic since EPR2 was completed. If you were continuing to work on new Tasks, list them here as in your previous progress reports. But this report should focus more on your activities to polish the documentation and create the presentation. Briefly describe which parts of the documentation and code you polished, and which parts of the presentation you contributed to. In addition, feel free to include any other information that provides convincing evidence you have been devoting the required amount of effort to the Epic project (about four hours per week).