
EE/CprE/SE 491 WEEKLY REPORT 5

10/10/2024 - 10/17/2024

Group number: ***sdmay25-33***

Project title: ***Interactive Embedded Systems Learning using the Prairie Learn Framework***

Client &/Advisor: ***Phillip Jones***

Team Members/Role:
Rachel Druce-Hoffman — Notetaker
Justin Cano — Technical Lead
Joey Krejchi — Quality Assurance
Caden Otis — Project Manager
Devin Alamsya — Consultant

(All the above information should be there in each weekly report. The format/color scheme etc need not be the same. However, please remove everything that is in a bracket from your final submission. These are just part of the template and need not be a part of the report.)

- **Weekly Summary**

  Our objective for this week was to continue improving our skills with and understanding of PrairieLearn. We continued work on the server, reading documentation, and re-learning 288 material. Now that we have created initial test questions, this week we were able to expand into autograding, question elements, and other more complex aspects of this project. We continue to gain a better understanding of PrairieLearn, past teams' successes and shortcomings, and our own vision for the project.

- **Past week accomplishments**

  - Caden: I recreated problem 2a from a past version of HW 5, which involved students creating a method that initialized GPIO port registers based on a diagram. From this, I learned a new way of making programs interactable, which is to have students write and submit their own code. I also learned how to take the student's answers, format them, put them into a txt file, and compare that txt file to the correct answers for autograding C code.
  - Rachel: First I fixed my VM- my issue was that I had been using the server Ubuntu image instead of the desktop image. This included cloning my old machine, installing the desktop ISO over it, and re-following the setup instructions. From there, I

decided to look into the student code autograding, as there are quite a few code snippet questions in the original homeworks. I learned about the required file structure to run tests on student code, and that although the C and assembly autograders are similar, they have distinct differences. I copied all the files from sdmay24's repo into our own, and I made some directory structure changes to make things easier to find. Then I created a new assessment/homework locally to test question types. I looked into the question elements described on readthedocs, and I was able to experiment with checkbox elements, number input with tolerance, and number input with units.

- Joey: I found out the fix to my original problem from last week. I started to write a question that uses the C autograder but ran into issues I think related to the autograder container. I will work with Caden on making this work in the upcoming week. I did not get as much done this week as I would have liked because I got sick on my planned work days.
- Justin: I created a group google account called cpre288.pl.f2024@gmail.com which I used to set up Google OAuth. Got my own VM set up with Prairie Learn and started looking at building my own questions from the given homeworks. We got the certificate signed by ASW and setup SSL for the server. Now the website is secured with HTTPS meaning we can now set the server up for ISU account integration.
- Devin: This week I looked into some more question creation and looked over some of the past CPRE 2880 Homeworks. I noticed that H1_Q5 looked like it was one that could be easily fully autogradeable. Changed the format of the question to be multiple choice dropdown instead of short answer. This made it so that it was autogradeable instead of having a TA or Professor grade the students short answer. Played around with some of the formatting attributes to make the question look more presentable.

- **Pending issues** *(If applicable: Were there any unexpected complications? Please elaborate.)*

🎬 Team Member 1: Worked on… 🎬 Team Member 2:

- Rachel: I moved contents of the folder /HW1 into /questions/HW1/rss to declutter our root folder and because these are resources only used in hw 1. PrairieLearn tries to interpret this rss folder as an additional question requiring an info.json file. I will need to work on having it ignore this as a question or a way for it to not interfere/confuse people with HW1.

- **Individual contributions**

NAME	Individual Contributions	Hours this week	HOURS cumulative
Caden Otis	I recreated a problem from a previous HW (HW 5, 2a) that focused on having students create a method in C and submit their program. From this, I learned how to format a question that takes in a C programming input and autogrades it.	5.5	16

Rachel D-H	Fixed VM. Read C autograding documentation. Copied sdmay24 repo into sdmay25 repo. Directory restructures and troubleshooting errors that that caused. Created 2 test questions to experiment with interactive question elements.	9	21.5
Justin Cano	I got google OAuth set up for our Prairie Learn App. Also finished getting SSL setup on the VM. I got Prairie Learn running on my own VM and I started creating questions from the homeworks starting with homework 6.	5	14
Joey Krejchi	Fixed my question from last week and attempted to write one that uses the C autograder. Got sick and didn't accomplish much.	3	14.5
Devin Alamsya	Looked at some of the documentation about question grading and creation. This helped me develop a better understanding of how to attack problems. Worked to make it so that H1_Q5 is fully autogradeable.	5	12.5

- **Plans for the upcoming week**
  - Caden: I first want to continue to work on my recreation of HW 5, 2a that I made. I didn't have enough time to randomize/create variants of the question. I looked a little bit into how the previous team did this, and there are quite a few components that I need to dive through and understand. Next, I want to continue to review more CPRE 288 concepts that I've forgotten about so that I can create more questions for any and all concepts.
  - Joey: I will fix the autograder so that it works with my question and then look into the emulator functionality that was written by the previous team.
  - Justin: This next week I plan on getting our OAuth app setup with ISU integration to allow students to login to Prairielearn with their student accounts. Then I hope to start getting the questions our team has finished uploaded and put onto the server.
  - Rachel: I would like to understand the C and assembly autograders better so that I can improve on the questions that utilize them. I also would like to work more with the different question elements and parameter randomization on more than just integers (ie strings, floats).
  - Devin: In the coming week I'd like to continue looking at H1_Q5 and see if I can continue to make the question more presentable and also more randomizable with the options that students are given for answers. Looking into being able to replace different verbs or nouns in a sentence to give a larger answer pool. I plan to look at the rest of the homeworks and the homework questions that the previous years team have implemented to figure out if the formatting or the implementation can be improved. I also want to look at the CPRE 2880 concepts to see if there's any inspiration I can find for new questions.

- **Summary of weekly advisor meeting**

For this week's advisor meeting, we mainly discussed with our advisor what everyone has been learning and experimenting with their local version of Prairie Learn. Since mostly everyone is now starting to learn how to autograde different types of questions, we spent some time talking about autograding techniques that the previous team from last year used, and an approach that Dr. Jones used to autograde C++ solutions. We also got Dr. Jones to help us reach the next steps of finalizing our server that will host the team's instance of Prairie Learn. Finally, we finalized a broad idea for our timeline of the project, where we intend to have a working prototype (beta version) ready for next semester's CPRE 2880 class to use.