**Introduction**

In my report I will be showing how I was able to create an automated plant watering system using arduino and java. My report will consist of an explanation relating to all the tasks completed in my minor project video. along with answering all the given questions within the minor project report guideline.In this minor project, I have successfully been able to present an automated watering system using Arduino and java. The code presented is also capable of demonstrating the connections that have been made between the java program and the Arduino hardware, this is shown through interactions via buttons and sliders, as well as a graph.

**Context**

* What: Arduino + Java programming to achieve the goal of making an automated watering system for my plant
* Why: This project can allow one to save time from watering plants, as well as keeping the plant healthy since a machine will not forget to water a plant. This project is also great for beginners in terms of learning coding and operating simple equipment.

**Technical Requirements/Specifications**

The technical requirements for the minor project consist of the following: the arduino board should be able to detect dry soil and water the plant without the use of a java program and/or have a java program which is able to detect dry soil and further alert the arduino to water the plant (one of the two is required, or you may do both). The next three technical requirements for the minor projects are the ones that are all mandatory in which there no choices to do one or the other (all must be completed), they are as follows: a button/command line created on java which tells the arduino board to turn the water pump on or off which is done manually, the next requirement is to create a javafx slider function which sends a command to the arduino which then creates a buzzer or LED, the final technical requirement is to be able to graph the active moisture value from the arduino on a javafx graph.

**Components List**

* A laptop -> which has the correct java program installed along with the additional javafx libraries
* The arduino board with all additional components such as;
  + Mosfet (allows us to connect the water pump to the arduino)
  + Moisture sensor (which will give all necessary moisture reading and allow us to know the soil moisture) - Water pump
  + Water pipe
  + Connectivity wires to connect mosfet/moisture sensor to arduino
  + 4AAA batteries
  + Battery pack in which the batteries will fit
  + Water supply
  + A plant for testing and growing

**A picture containing houseplant, flowerpot, book, indoor

Description automatically generated**

**Procedure**

The procedure will be broken down into two separate categories, the first category will highlight the procedure to assemble the hardware portion of the system, and the second category will be dedicated to explaining the software part of the system.

* Hardware Setup:
  + Setting up a plant
  + Assembling the batteries into the battery holder and then connecting the battery pack to the MOSFET which is then connected to the Grove board with the provided wire (slot D2)
  + Connecting the moisture sensor to the grove board with the wire provided (slot A0) - Connecting the pump to the MOSFET
  + Connecting the water pipe to the water pump
  + Setting up the water reservoir for the water pump to send the water int the plant
  + Connect the grove board to the computer.
* Software Setup:
  + Prepare Arduino code for the automated water system.
  + Prepare the Java code to show interactions between the grove board and Java (graphs and slider).
  + Run the Arduino code first and then the Java code.

**Test**

Making sure that I was attending and understanding labs throughout the semester was a high priority in terms of preparation for this minor project. The online videos were all a great tool but the hand on labs had a greater correlation to this type of hands on project. Also while troubleshooting and trying to get my code to work and everything to run smoothly, I subconsciously flood my plant by doing reruns and constant testing. Shown below is an example of one of the times where my javafx graph was not showing any changes and was staying constant regardless of the soil moisture sensor being in a dry environment or in a wet environment. This was seen during the testing phase and is a situation I was able to figure out and make changes so the final product comes out to be how it is supposed to.

A picture containing computer, text, output device, personal computer

Description automatically generated

**Contingency**

One thing that I will definitely change for future projects and courses such as ENG 4000 is my time devotion to such a project. I feel like I did not give myself enough time to do this project and made it harder on myself. So any ideas I had in mind were not always feasible due to the time constraint I put myself in. An example of this would be part a, where it says showcase either one part or the other or showcase both parts, I would have loved to have been able to showcase my knowledge and ability to complete both parts, however I was only able to include one part for this project. I also wanted to give myself a nice setup and a cleaner workplace dedicated to my EECS hands on assignments. I was not able to do so this semester but I definitely hope to achieve that next semester.

**Conclusion**

In conclusion I feel like that this project was filled with nothing but pros, I was able to learn a ton of new things in terms of programming and the overall java language, along with being able to apply that knowledge and see physical achievements in terms of a working system. I feel that this project gave me a great infrastructure and moving forward I will be able to become better and better as each day passes.