Deliverable G

Group 8

Thomas Alkhoury, Laura Godfrey, James Hight, Cecilia Lou & Julian Ward

GNG 1103

November 8, 2019

Contents

1. Introduction 3
2. Reflection 3
3. Objective 3
4. Stopping Criteria 3
5. Plans/Procedure 4
6. Results and Analysis 4
7. User Feedback 5
8. Proof of Concept 5
9. Conclusion 5

1. Introduction

Within this deliverable we will be going into more depth to achieve a better prototype. This will be accomplished by reflecting on our previous prototype and deliverable to improve our second design. We will become more specific with our objective, stopping criteria and plans/procedure. We will also include the results and an analysis of our current prototype (prototype 2), as well as user feedback on our second prototype in an attempt to make our prototype as user friendly as possible.

2. Reflection

After looking over the results and analysis for Prototype I as well as the user feedback. It can be concluded that our Prototype met the objective and stopping criteria, it takes a manual input and displays whether the machine is in use. In order to improve the prototype, in Prototype II, the team plans on finishing the Raspberry Pi (RPi) aspect of the project, allowing flow of electricity based on the user’s qualifications. This will improve on the previous prototype, while allowing us to test a different aspect of the product.

3. Objective

The objective of Prototype II is to control the flow of electricity through the use of the RPi. It should recognise an input and allow or deny the flow of electricity based on the input.

4. Stopping Criteria

The stopping criteria is the criteria which needs to be met so that we can stop testing the prototype and deem it acceptable. For this deliverable, the stopping criteria must be more specific than the one described in Prototype I (Deliverable F). It must therefore do the following;

- Register arbitrary input in Ross Video representing barcode. This will be done either by manually inputting it into the system or by having the Ross Video interface.
- Access or store a list of student IDs and their corresponding training for each student ID. This will be done either by saving and editing it locally in the Ross Video interface or by communicating to a machine containing this list.

- Relay must activate (shut power off) when the user's student ID does not match the required field for the machine in question. This will be done by referring to the list of student IDs and their corresponding training in Ross Video (by communicating with live feed).

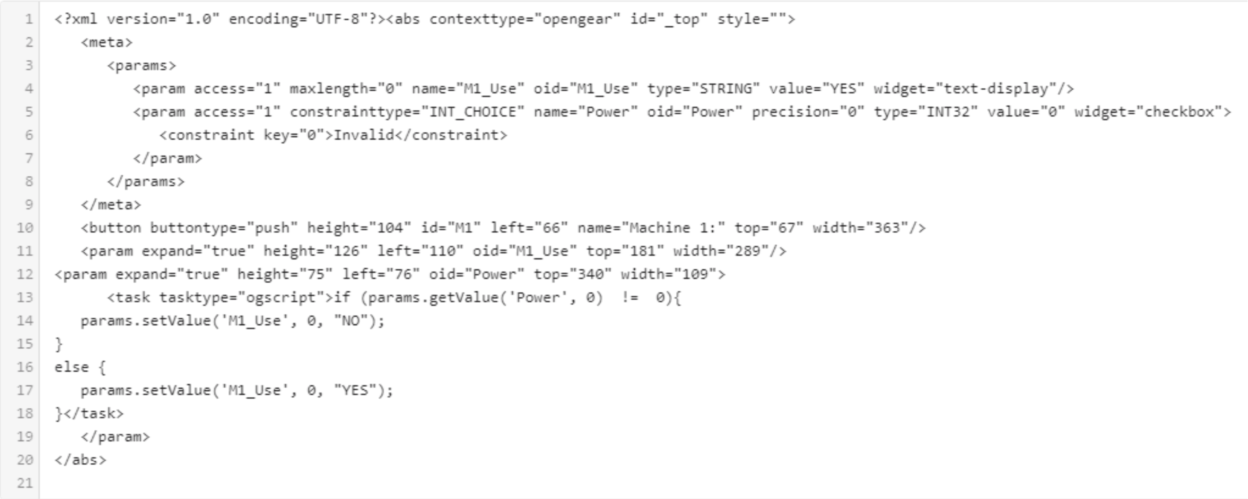
5. Plans/Procedure

For this prototype we planned on making the code that integrates Raspberry Pi with dashboard so that machines in CEED don't turn on unless your MakerRepo account shows that the user has the training for that machine. To make this we wanted to write a code that gets input from the RPi and then depending on whether the person has the training, the relay will let the machine be turned on. This prototype is mostly focused on the code used on the RPi.

6. Results and Analysis

Initially, using the Raspberry Pi was successful in controlling an LED and turning it on and off. However, after subsequent use the RPi stopped being able to boot. After some troubleshooting, the most likely cause was determined to be that the Pi was shorted in between uses. If this is what happened, then it means the console is no longer usable and a replacement RPi will be needed.

The initial steps were taken in integrating the RPi and Dashboard, but the next steps could not be taken without access to a RPi unit. Additionally, the code on Dashboard to receive an input and give an output was successfully completed, as shown below.



```
<?xml version="1.0" encoding="UTF-8"?><abs contexttype="opengear" id="_top" style="">
   <meta>
      <params>
         <param access="1" maxlength="0" name="M1_Use" oid="M1_Use" type="STRING" value="YES" widget="text-display"/>
         <param access="1" constrainttype="INT_CHOICE" name="Power" oid="Power" precision="0" type="INT32" value="0" widget="checkbox">
            <constraint key="0">Invalid</constraint>
         </param>
      </params>
   </meta>
   <button buttontype="push" height="104" id="M1" left="66" name="Machine 1:" top="67" width="363"/>
   <param expand="true" height="126" left="110" oid="M1_Use" top="181" width="289"/>
<param expand="true" height="75" left="76" oid="Power" top="340" width="109">
      <task tasktype="ogscript">if (params.getValue('Power', 0)  !=  0){
   params.setValue('M1_Use', 0, "NO");
}
else {
   params.setValue('M1_Use', 0, "YES");
}</task>
   </param>
</abs>

```

Figure 1: Raspberry Pi Code

7. User Feedback

For this deliverable, since the prototype had major issues, users were instead asked a potential user to review the overall concept of the project. He recommended having a central monitor that would display all the machines and their current status (in use or not). He suggested using a spreadsheet-style setup, similar to a departures board at an airport that would display the machine, the elapsed run time, occupancy, and any error messages (such as because the user does not have the training).

He also recommended having an aural or visual indication of the machine turning on or not, having instructions on or near the Raspberry Pi to explain to the user what could happen when they tap, and the possibility of the occupancy display being accessed remotely (such as from a phone online).

8. Proof of Concept

While we are having issues with the Raspberry Pi we were able to improve upon our Dashboard code to display if the machine is receiving current or not by using a manual input of a check box.

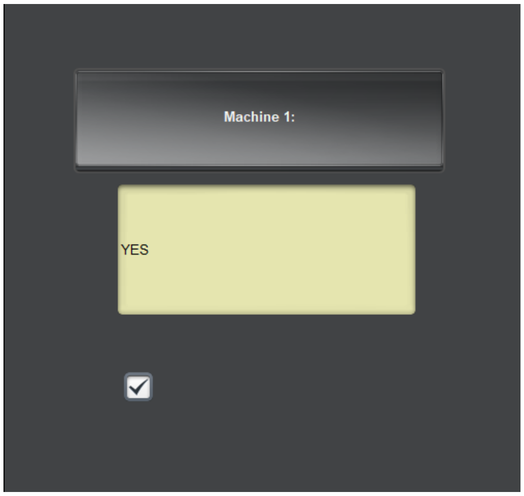


Figure 2: Dashboard Display

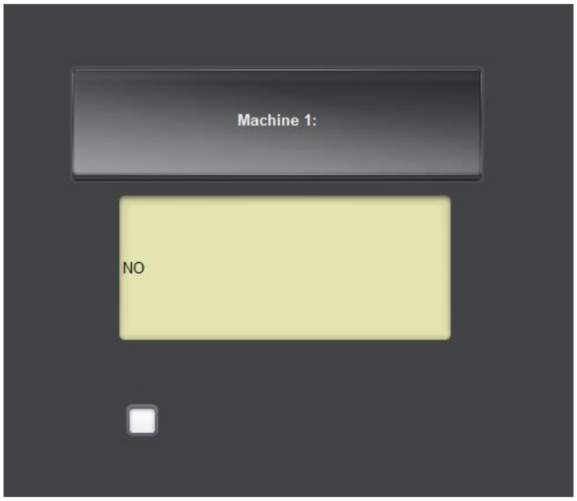


Figure 3: Dashboard Display

9. Conclusion

This deliverable served as a learning experience for the group to introduce itself with Raspberry Pi and trying to integrate it with Dashboard. Unfortunately, the RPi unit was rendered unusable, so a contingency plan needs to be drafted and adopted. This may include buying a new RPi console, adapting the project to use an Arduino Uno instead, or figuring out a way to use neither Arduino nor RPi in the final project. The team will be meeting to discuss possibilities. The overall project was met with positive reviews and given some suggestions to make it more accessible to potential users.