Prototype II and Customer Feedback

By: Ayesha, Adam, Shoaib, Gurinderpal

7/11/19

Introduction:

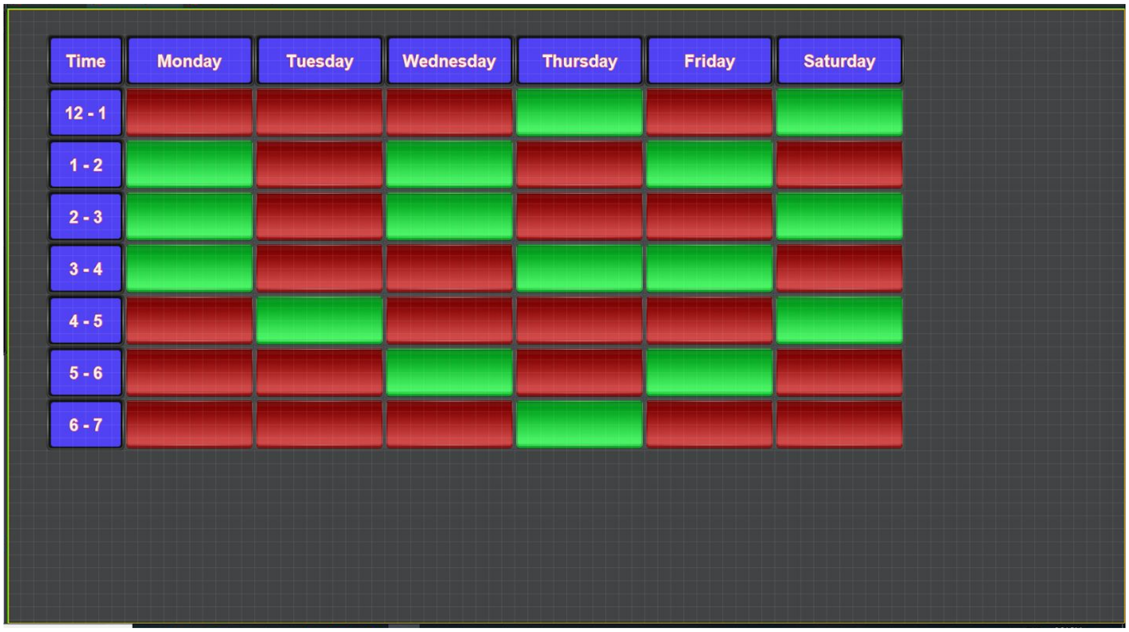
This report outlines prototype II, which consists of a semi functioning dashboard interface and software to accompany each subsystem. This prototype has been analyzed and evaluated by both the team and potential users. These tests provide insight into how the system can be further improved upon. The feedback provided by potential users allowed the team to further refine the user experience and ensure that the system will be as effective and efficient as possible.

Prototype II:

Growth:

Since the last prototype, a display for each subsystem has been created and the Node MCU coding is underway. The team has since implemented a help button on each screen and has decided to use a 5V battery for powering the boards. The board is now able to communicate both ways with the Node MCU. Ayesha is currently figuring out how to send integer values over the connection since the sensors will be taking in integers. She is also learning how to use the http POST function in Dashboard. This prototype brings what was sketched and outlined to life. It will implement the feedback given then as well. For example, the schedule will use the student number to track the user instead of the name. The machines will be displayed in a manner that will allow the user to easily see which is which.

Scheduling/Reservation Subsystem:



Time	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
12 - 1	🟥	🟥	🟥	🟩	🟥	🟩
1 - 2	🟩	🟥	🟩	🟥	🟩	🟥
2 - 3	🟩	🟥	🟩	🟥	🟥	🟩
3 - 4	🟩	🟥	🟥	🟩	🟩	🟥
4 - 5	🟥	🟩	🟥	🟥	🟥	🟩
5 - 6	🟥	🟥	🟩	🟥	🟩	🟥
6 - 7	🟥	🟥	🟥	🟩	🟥	🟥

Figure 1: Scheduling map

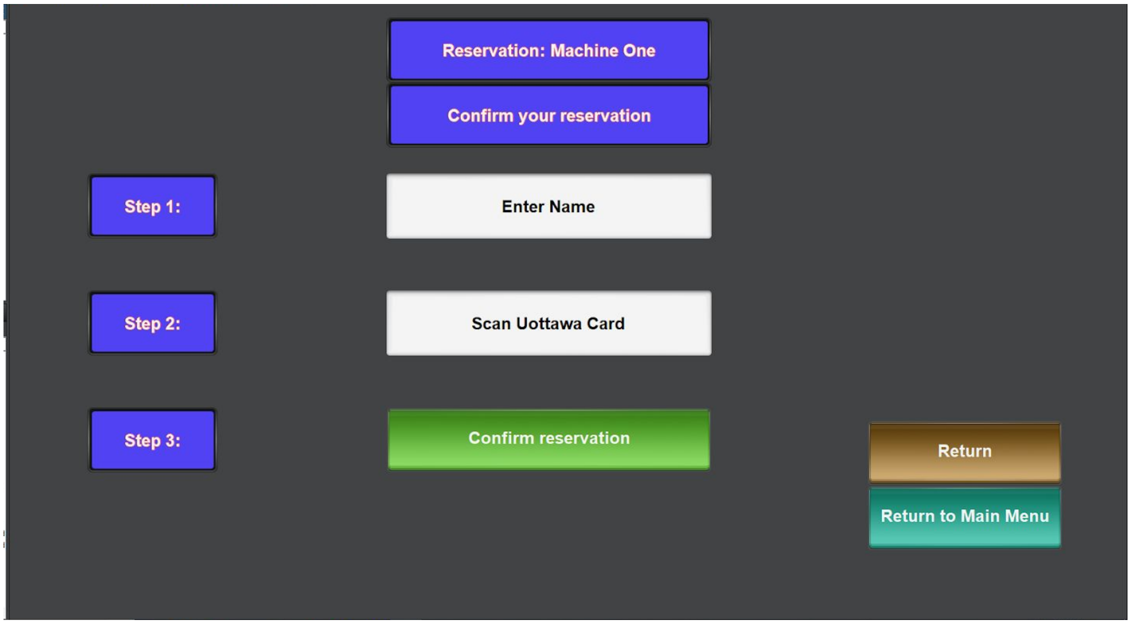
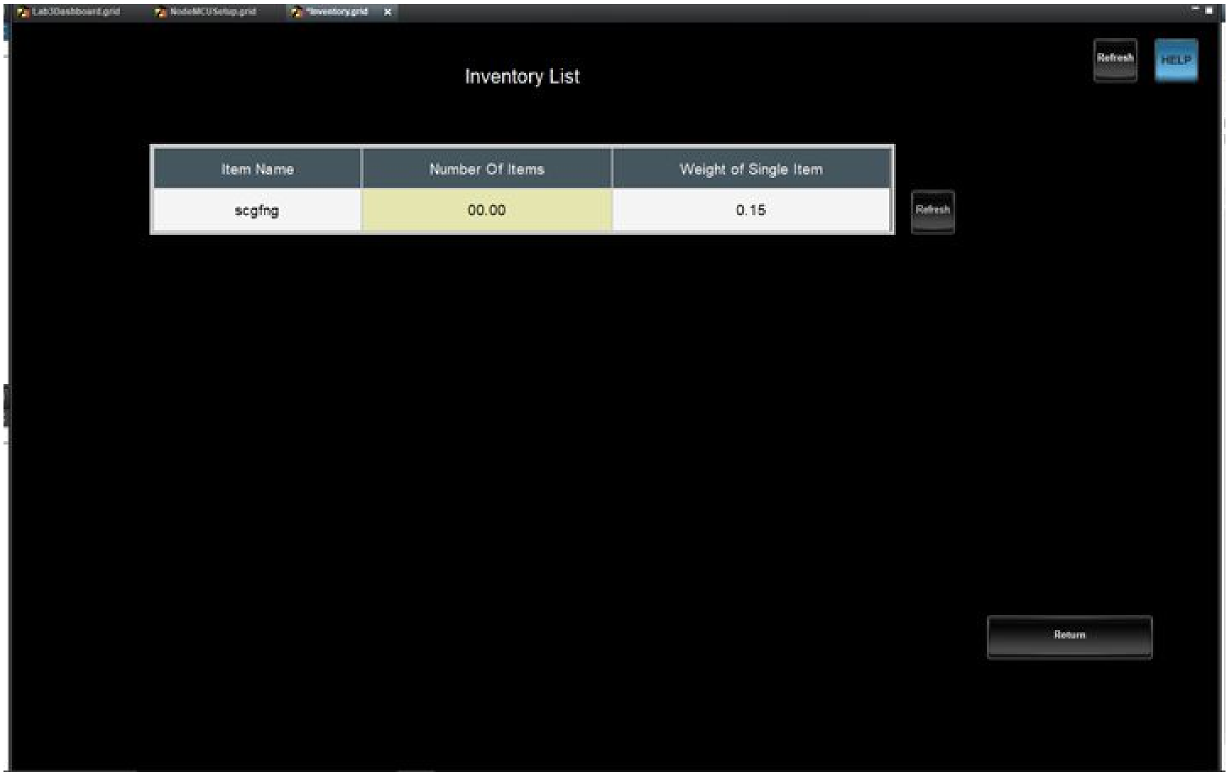


Figure 2: Confirming your reservation

Inventory Management Subsystem:

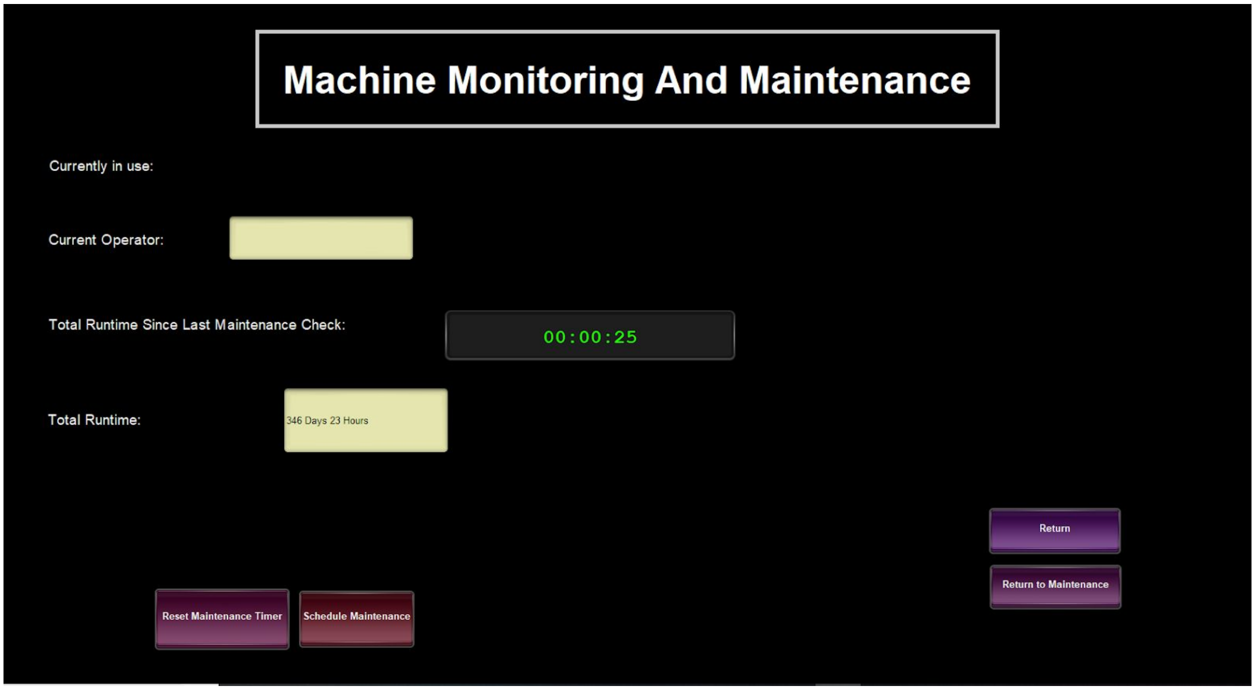


Inventory List

Item Name	Number Of Items	Weight of Single Item
scgfng	00.00	0.15

Figure 3: Inventory Screen

Machine Monitoring Subsystem:



Machine Monitoring And Maintenance

Currently in use:

Current Operator:

Total Runtime Since Last Maintenance Check: 00:00:25

Total Runtime: 346 Days 23 Hours

Figure 4: Machine Monitoring for Maintenance

Dashboard UI:

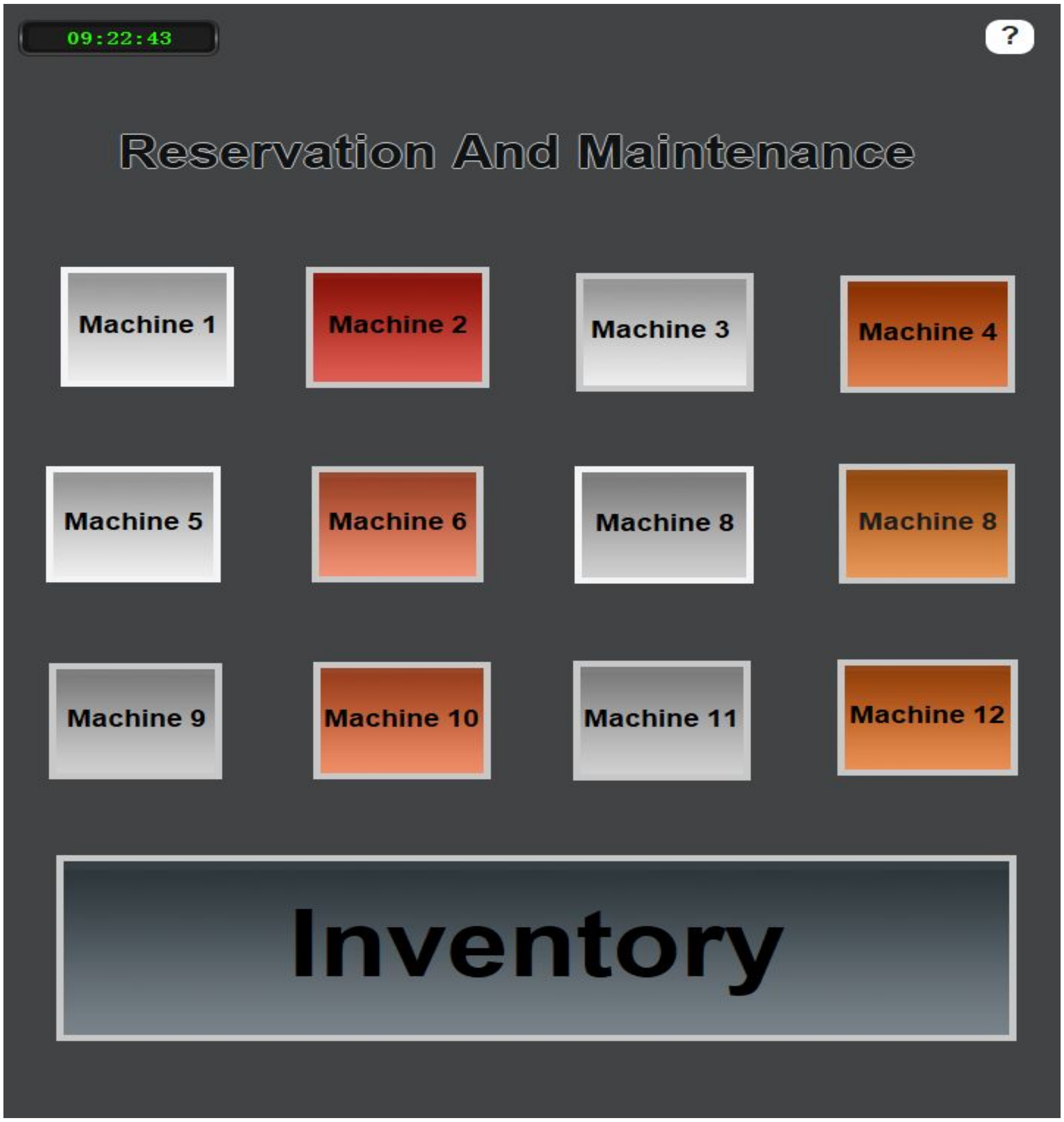


Figure 5: Home Screen for Orwell

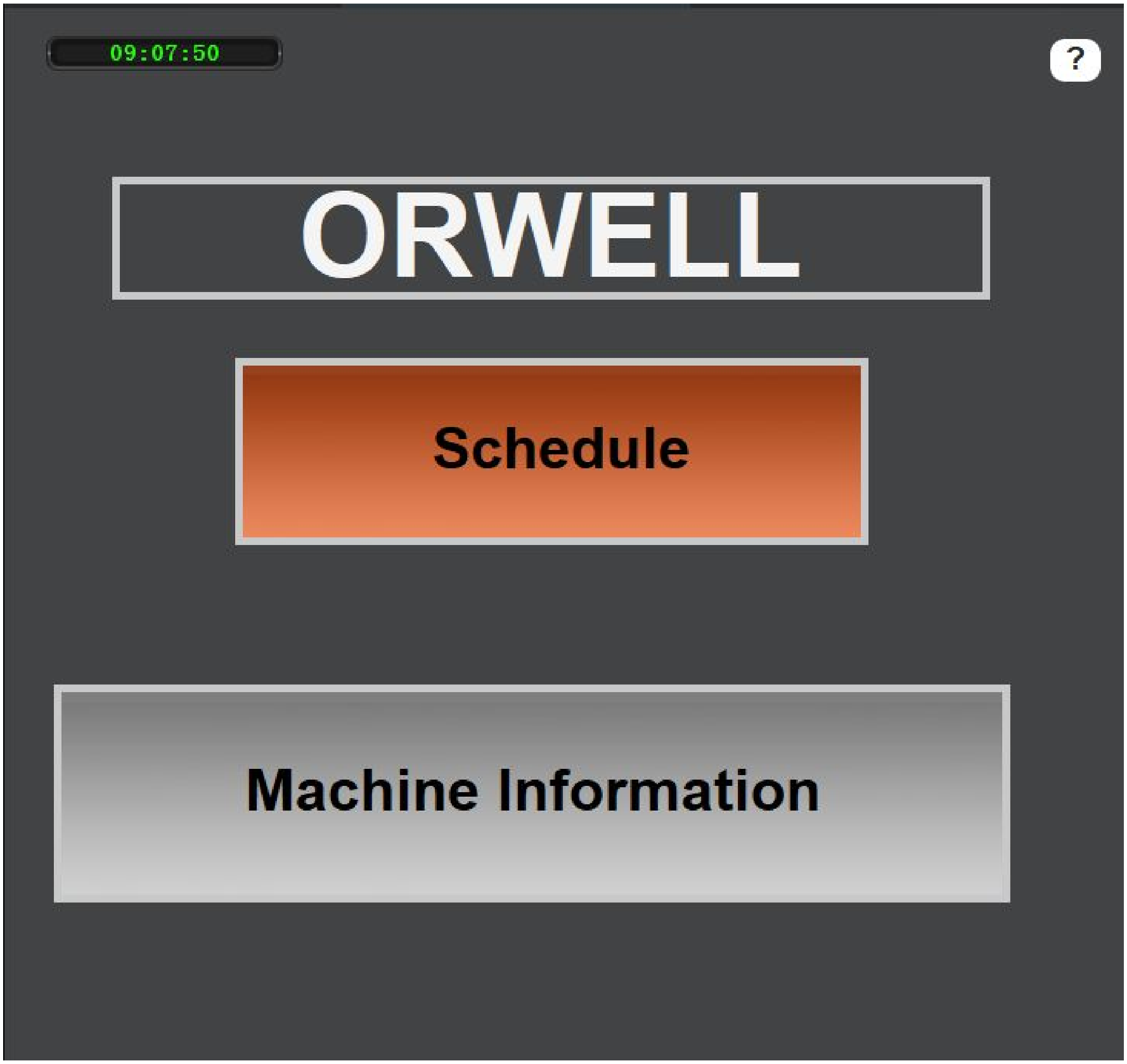


Figure 6: Screen 2 for When a Machine is Selected

Prototype Description:

Scheduling/Reservation Subsystem:

The reservation subsystem prototype is an almost fully functional dashboard interface. Users can select which machine and the time at which they want to reserve it. The current prototype will allow the user to create and remove their reservations. The prototype will automatically remove reservations that have been completed.

Inventory Management Subsystem:

In this prototype, the interface has been created with all the necessary parameters. As seen in figure 3, the user will be able to both view and edit the inventory list on the same page. A refresh button will be supplied in order to give the user more control should there be any problems in the future while it's running. The Node MCU is connected to Dashboard and the internet. Once a method for sending integers has been found the system shouldn't take long to complete.

Machine Monitoring Subsystem:

In this prototype a Dashboard component for the machine monitoring was created. It displays if the machine is currently in use, who is using the machine. It displays its lifetime use, and how long its been used since it last received maintenance. It also has the option to reset the maintenance timer, and schedule maintenance.

Dashboard UI:

In this prototype, the different subsystem panels were created in Dashboard. A home screen to integrate all the screens together can be found in figure 5, and figure 6 represents the secondary homescreen for machine monitoring and scheduling/reservation.

System Analysis:

The interface will be relatively straightforward, minimizing any possible confusion when navigating it. Each page/panel will have a help/troubleshooting button available to ensure that any possible problems can also be minimized. Furthermore, the user will also have the ability to refresh most if not all of the pages. This system will be key to the user experience and functionality of all the subsystems.

Test Plan:

This prototype will be tested to see if the improvements implemented from the previous round of prototyping and customer feedback work and if they are actually beneficial. The test will also give insight to the user experience. This prototype will be far more focused than the previous prototype. This prototype will be a critical analysis of the scheduling/reservation subsystem. The tests will see if the dashboard is effective in fulfilling its purpose and/or if any further improvements must be made when creating the final prototype.

Test Objectives Description:

The specific test objectives are the following:
- Is the program Interface simple to navigate?
- Is the help button working?
- Is the Node MCU connected to Dashboard?
- Can you send Integers over the Connection?

This prototype and test will determine if the UI is effective and if the Node MCU connection is working and useable. This test will help the team learn about any problems with the interface and if there are any shortcomings in the user experience.

The result will be in the form of notes and will most likely not have any metric results. When testing the Node MCU connection, the result will be a confirmation message in the serial monitor and in Dashboard.

The results from this test will help the team determine if the UI is effective and efficient. It will help the team further sort out what to keep and what to get rid off for the final prototype. By testing the NodeMCU wireless connection the team will be able to gauge how long it will take to get the machine monitoring and inventory system up and running. These results will aid in maintaining an effective and accurate timeline for creating the final iteration of Orwell. A successful test will simply be a test that fulfills each criterion (that applies) in its entirety. If a criterion is only partially or not at all fulfilled then the test will be considered a fail. By having a binary system for testing the system, the team will be able to gauge how where they stand in the manufacturing process and what needs to be completed.

What is going on and how is it being done?

The second prototype is focused on the Dashboard UI and its relevance with all the different subsystems. This type of prototype was selected because the team is still developing the Node MCUs and they aren’t ready to be tested yet. This way the team will also be able to spot any problems or issues with the user experience, and create a basis for the Node MCUs to be connected to. Additionally, this prototype will also test the connection between the Node MCU and Dashboard ensuring that they are communicating with each other.

The test will consist of the testers (team members) navigating through the different panels and noting whether or not each test objective is being fulfilled. If an objective is not fulfilled the testers will then note down, what is supposed to happen and what is actually happening. There will be three tests in total: machine monitoring, Inventory, scheduling/reservation. Each tester will navigate through the system trying to complete test specific tasks.

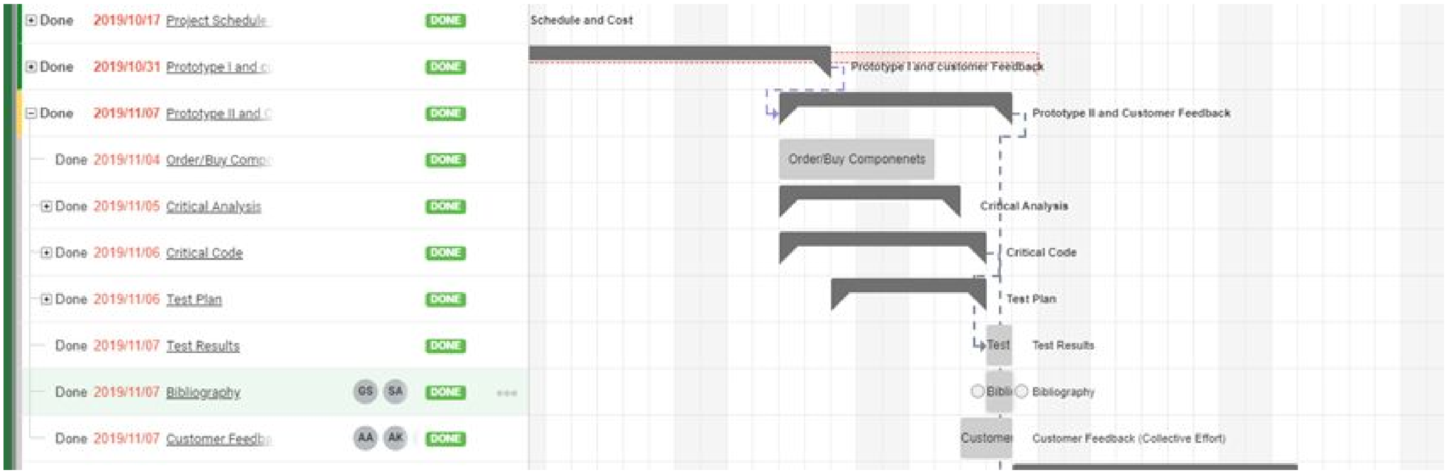
In test 1, the tester will navigate through the system with the intent of creating a reservation and viewing the schedule. In test 2, the tester will navigate through the system will the intent of trying to view and update the machine maintenance available. In test 3, the tester will navigate through the system with the intent of adding an inventory item and viewing the inventory list. Whilst going through the system, any flaws, accomplishments and suggestions will be noted.

The tester's (who in this case will be the user) experience when using the UI. Their satisfaction and frustration will be measured. This will not be a metric value, however, there will be notes taken to outline their thoughts and notes. The team will be observing how effective and efficient the UI is and will be taking notes whilst conducting each test. The notes will serve to record the results and act as a reference point for any future points of improvements. Since Dashboard and Arduino IDE are free services, therefore, there are no costs associated with them and the Node MCUs were purchased early on in the development process. Hence, there will be no additional costs to test this prototype.

In order to achieve a complete product, the team must learn to send integers to and from Dashboard. The Node MCUs must be wired and tested, the different panels in Dashboard need to be connected to each other and a housing unit must be manufactured (if possible) for the Node MCUs and protoboard.

When is it happening?

Each test will take 10 - 20 minutes to complete and before they can be conducted the Dashboard panels for each subsystem must be completed. An initial connection must be established between the Dashboard and Node MCU before it can be tested. The tests are required by the 7th of November at the latest as the rest of the project relies on the results.



⊞ Done	2019/10/17	Project Schedul		DONE
⊞ Done	2019/10/31	Prototype I and c		DONE
⊟ Done	2019/11/07	Prototype II and C		DONE
Done	2019/11/04	Order/Buy Comp		DONE
⊞ Done	2019/11/05	Critical Analysis		DONE
⊞ Done	2019/11/06	Critical Code		DONE
⊞ Done	2019/11/06	Test Plan		DONE
Done	2019/11/07	Test Results		DONE
Done	2019/11/07	Bibliography	GS SA	DONE
Done	2019/11/07	Customer Feedb	AA AK	DONE

Figure 7: Gantt Chart for Test Plan

Prototyping Results:

Test 1: Scheduling/Reservation Subsystem

Goal: View List and Create Reservation

Notes:

- The system is easy to navigate but, there is no method of returning to the previous screen from the schedule map. There is no apparent way to choose between the schedule map and "making a reservation" screen.
- The machines aren't labelled correctly at this time
- There is a help button on the reservation screen however no help button on the schedule map screen.

Result: Failure to meet all applicable criteria

Test 2: Machine Monitoring Subsystem

Goal: View when a machine was last maintained, view maintenance data.

Notes:

- Easy to navigate the system. All buttons are labelled and present
- No help or refresh button

Results: Failure to meet all applicable criteria.

Test 3: Inventory Subsystem

Goal: View Inventory and add a new Item

Notes:

- The system is easy to navigate, there is a return button
- There is a working Help, reset and return button.
- The Node MCU can communicate with Dashboard
- Unable to send Integers over the connection at the moment

Results: Failure to meet all applicable criteria.

Customer Feedback:

Client Name: Matt - Post-graduate Student

Notes:

Matt recommended that the design can have a feature of allowing the person to make reservations without actually coming to Brunsfield. He wanted us to link our setup to some website or existing online interface. According to him, it would be a beneficial asset of the system as it could be a great help for the person who lives at a relatively far distance from the building and needs to save time. He said that the UI as it is was understandable and effective.

Client Name: Daniel - Student

Notes:

Daniel suggested that the inventory and machine maintenance systems be password guarded to ensure that only authorized personnel will be able to access it. He also suggested that there be a system available to secure a users spot when they make a reservation as an incentive to use the system. He said the UI was clear and understandable, however, that the colours were a bit jarring and that a help or refresh button would help the user if they get stuck.

Client Name: Jared P - CEED Staff Member

Notes:

Jared suggested that there be a quicker method to sign in. For example, where the user would come in and simply select the machine then tap their card on the RFID to reserve it. This would help the staff in the mornings when people rush in and keep everyone working efficiently. He also suggested that each “machine” button should indicate whether it's in use or not without having to go into the schedule this would, saying that it would make the system more efficient.

For machine monitoring he suggested that each machines maintenance screen be easily customizable since each machine would have very specific needs as far as maintenance goes. For inventory he said to ensure that the list continuously updates automatically.

Improvements:

From the feedback gathered the team has decided to try and make the :machine" buttons color coded, the colour scheme for the different subsystem panels will match more and be less jarring. They will try to implement a function that allows the admin systems (machine maintenance and inventory) to be more secure. The team will also learn how to send integers over the NodeMCU connection.

Conclusion:

Building and testing the second prototype for Orwell helped the team identify certain limitations in our project which might have led them to making a poor and flawed product. The feedback obtained from the CEED staff and students will be used to further refine Orwell's UI. The tests revealed flaws in the Dashboard interface, these flaws have been noted and corrections will be implemented when developing the third and final prototype. By having the second prototype focus on the UI alone, the team was able to focus on specific attributes related to both the functionality and user experience. Moving forwards from this prototype, all of the feedback and results will be taken into account to further improve Orwell and create its final prototype.