1) Unit Testing

Testing different LED functionality (Response class)

Execution	Expected result	Status Pass/fail	Comments
Execute function to light up LED	LED is lit up	Pass	
Execute function to turn off LED	LED is turned off	Pass	
Execute function for LED to blink rapidly	LED blinks rapidly	Pass	
Execute function for LED to blink slowly	LED blinks slowly	Pass	LED blinks much slower compared to the when it blinks rapidly

Testing CommandManager class

Execution	Expected result	Status Pass/fail	Comments
Execute addCommand() with random input	commandArray() matches random input	Pass	
Execute identifyCommand() with random input	identifyCommand() correctly identifies command	Pass	Used assertion to test

2) Regression Testing

Implemented function that causes LED to blink fast, to signify that the alarm has been sounded. Function is called when the button “ON ALARM” is pressed on the client page.

a. First Regression Test Case

After that, implemented slow blinking of LED to signify that the registers have been locked. Slow blinking occurs after the alarm has been sounded. So once the alarm has been sounded, LED will blink fast, then slow, then fast henceforth. No error occurs and alarm being sounded works correctly.

b. Second Regression Test Case

Implemented function to disable alarm (turn off LED). The function should only be able to turn off the LED once the slow blinking of LED occurs. No error or bugs occurs, and disable alarm function correctly works.

3) Performance Testing

The time it takes to send the signal from the client to the app takes an average time of 0.15791015624975ms (test case repeated 20 times). The start time is when the button on the client’s web page is clicked, and the end time is when the app receives the signal from the client.

The time it takes to setup the server (server.js) takes an average time of 12146.643900000001ms (test case repeated 20 times).The start time is once the server starts up, and the end time is when the server is fully running and operable. console.time() is the tool that was used to record the times for both cases.

4) Usability Testing

Testing client web page

Execution	Expected result	Status Pass/fail	Comments
Press button “ON ALARM”	LED first blinks rapidly, then slowly, then rapidly henceforth	Pass	
Press button “OFF ALARM”	LED turns off	Pass	Registers have to be locked first (once LED blinks slow)
Press button “ON SAFE”	LED lights up	Pass but fail if alarm is sounded	Does not light up continuously if alarm is sounded
Press button “DISABLE EVENTS”	LED turns continuously	Pass	Registers have to be locked first (once LED blinks slow)

Feedback

a. What do you feel for the current UI?

The current user interface is very simple and dull. However, all of the functions on the web page works correctly. Thus, it is sufficient for now.

b. Is it easy and straightforward to be used?

Yes, it is very straightforward and easy to understand making it very user-friendly.

c. From your point, what can be improved?

The user interface is currently very simple. The fonts could be enlarged to make it easier to read. The buttons could be more spread out and made larger. This is because during an emergency, accidentally pressing the wrong button or taking a longer time to read the buttons could be very costly.