**Atlantic Technological University, Sligo**

Obrázok, na ktorom je text, vizitka, snímka obrazovky, písmo

Automaticky generovaný popis

**Individual Report**

Name: Alekss Belavskis

Student ID: S00253368

Email: s00253368@atu.ie

Course: BSc in Software Development L8

Year: 1

# Team Role:

In my group for Project 2, my main objective was to upgrade the code we used in Project 1. Between project 1 and project 2, we decided to change from using the Arduino Yun to the ESP32 because of its smaller and greater performance.

I continued to use the Arduino IDE during the coding process of the ESP32 for Project 2, as it worked very similarly to the Arduino, aside from some minor code changes. I found Visual Studio to be less clear-cut to setup when working with the ESP32.

I also ended up modifying the previously used Python script for generating the RGB LCD’s colour by adding a cubic interpolation function which addressed some issues with colour transitions, but this feature never ended up being implemented in the ESP code due to the language differences between Python and C.

# Problems Faced:

With project 2, as we decided to use the ESP32 instead of the Arduino Yun, I needed to spend some time adjusting to and learning about how the ESP32 works and how to write code for it. Overall, the code way you write code is very similar for both chips, but there are different libraries required. I also experimented with Visual Studio, but settled with Arduino IDE as it was easier to set up. After adjusting to the small differences in how the ESP works compared to the Arduino Yun, I was ready to add more features to the room monitor.

During my time adding new features to the room monitor, I faced several coding problems I’d eventually come to solve. For example, when I added the functionality of the button to make the display change between showing the temperature and the air quality. I hadn’t initially realized that the button’s state only changed from 0 to 1 while the button was pressed, instead of toggling between 0 and 1 after being pressed and released. I fixed this functionality using logic that compares the current button state to the previous button state. This works by toggling the LCD state only when the last button state and current button state are different.

# Overall Reflection:

I found using GitHub to be very helpful in relation to the coding process. There were times where I wanted to revert part of my code to a previous state, and I could easily do so by looking at previous commits. It also helped us to share our documentation and keep it up to date while we worked on it.

We found that the Arduino Yun did not have very reliable Wi-Fi, because it oftentimes wouldn’t recognize the network I was trying to connect to, and switching to the ESP32 solved this problem. It was much easier to connect to Wi-Fi using the ESP32 in general, and I later found out with some experimenting that it was possible to connect the ESP32 to ‘eduroam’, a WPA2-Enterprise network, using my college credentials. (These credentials were kept safe using a ‘.gitignore’ file)

In conclusion, I’ve really enjoyed working with my group towards a common, and I’m overall satisfied with the product that we created. In the future I’d like to revisit this product idea in my own time to polish it and possibly improve it.