# Josias Hoffman Personal Portfolio

## 2022/11/07:

We started the day by doing a couple of exercises regarding IOT. The first regarded domains, protocols, and devices around IOT, the second was directed towards the esp8266. Afterwards we began to flash our very first IOT device, a raspberry pi which we would use as a local network host. Funnily enough the device was built into a keyboard which I mistakenly thought was just a regular keyboard. The task went about as smoothly as could be expected of students who have never worked with IOT devices before (mainly myself). I forgot to download the flashing software which was required to flash the device (Etcher). After downloading the software and flashing the device we were at last able to connect to our own local network.

### Personal Reflections:

I quite liked the first day of class since I’ve personally never built or programmed IOT devices before. I would have liked to start with breadboards and logic gates but since we had a time constraint, I understand why the first task already seemed so daunting.

## 2022/11/08:

The second day of class, it was time to start the real nitty gritty of firmware coding. We downloaded the Arduino IDE which I did not like very much to be quite frank. It had no intellisense which, all things considered, isn’t really the end of the world. We started on the basics, blinking the on-board LED of the esp8266. This task was very easy since we had access to the example code snippets which helped immensely.

### Personal Reflections:

Personally, I think the lesson was very productive. Whilst I have coded in C before, I don’t have a lot of experience with the language so it would have been nice to have a bit of a crash course before attempting this exercise but in all fairness the idea was more to use the example code given and just adapt it for our individual case.

## 2022/11/09:

The third day we finally got to start working with NODE Red, a very fun and visual coding aid which can have multiple logical functionalities as well as a robust UI. We connected to NODE Red using our Raspberry pi that we created on the first day. The first tasks revolved around creating MQTT connections between different nodes and playing around with the general logic of the environment. Afterwards we moved on to connecting actual devices to these MQTT connections and displaying the data on our NODE Red UI. We also created in-depth user stories which revolved around a system that we wanted to create, users of the system, and how these users (or misusers) interacted with the system.

### Personal Reflections:

To be honest the NODE Red environment was extremely fun, whilst it might be simple and boring for some, I really enjoyed the visual elements and “Building nature” of NODE Red. The most enjoyable part was definitely when we managed to connect a temperature sensor to our NODE Red dashboard and see real-time data being displayed in our dashboard.

## 2022/11/11:

Next, we finally got to work with the IOTEmpire framework, which I was utterly amazed at. It has streamlined the first two days of work to simple one line function calls and console commands. We ran through tasks by simply using the built-in documentation for any function calls we needed to make and once the device was flashed, we could deploy remote firmware updates to it from our laptops as long as it was connected to our Raspberry pi. We had some trouble with flashing the devices for the first time since an error occurred which caused us to have to use a work around in the form of a downloadable flashing software.

### Personal Reflections:

In the end I learned a lot about not only the framework, but how these types of software can drastically streamline firmware coding to the point only needing one line of code per functionality for each esp8266. This type of software is amazing in my opinion and I would love to develop my skills enough to be able to contribute some day.