



Clairvoyant Cloud

WIT Project I // Jonathan Gillespie // May 5th 2017

Outline

- The Concept
- The Journey
- How does it work?
- What was learned?
- The Final Product
- Demonstration
- Questions



The Concept

- Richard Clarkson
- Please, no more notifications.
- A Hardware Puzzle
- Instructable
- The Deep End...



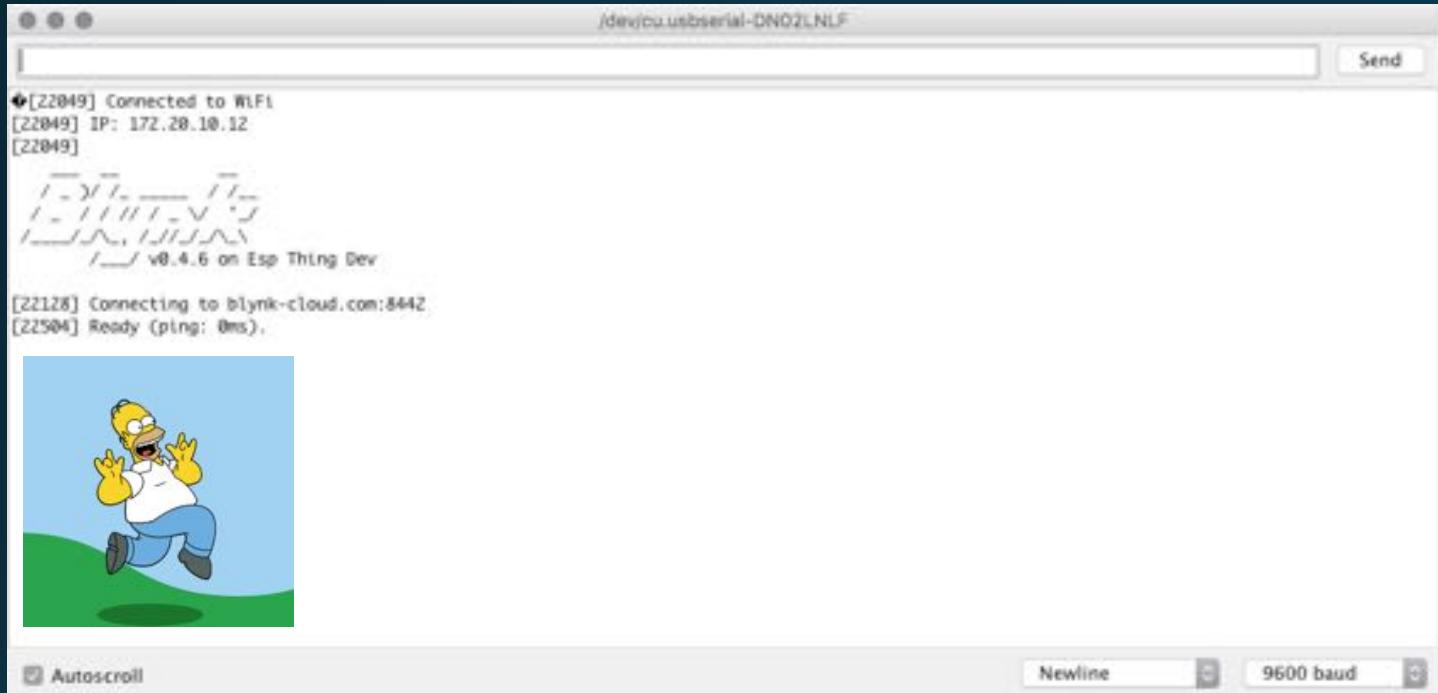
The Journey

The Bits of Kit

- Arduino Pro Mini
- Thing Dev Board
- FTDI Breakout
- Solderable Breadboard
- $1000\mu\text{F}$ Capacitor
- 330Ω Resistor
- 5v6A Power Supply
- 120 Addressable RGB LEDs (2x 1m Strips of 60LEDs)
- Arts and Crafts

The Journey

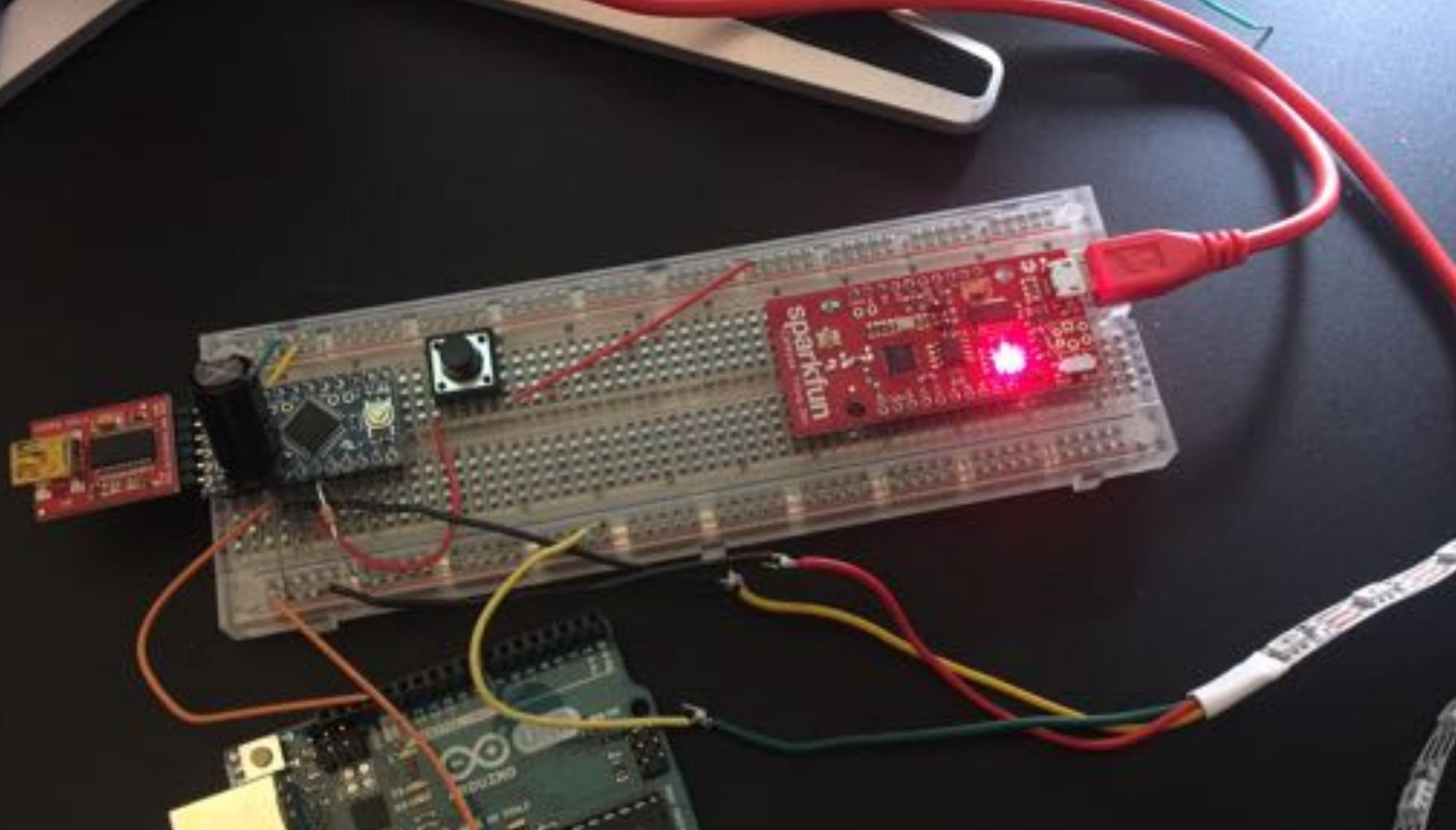
- Hardware Configuration
 - Mini & Thing
 - Additional Boards
 - Libraries
 - Baud
 - Signal, what signal?
 - Wifi & Blynk
 - Bite Size Tests
 - The API
 - Lat/Long & Authorization
 - Rules are Rules.



The Journey

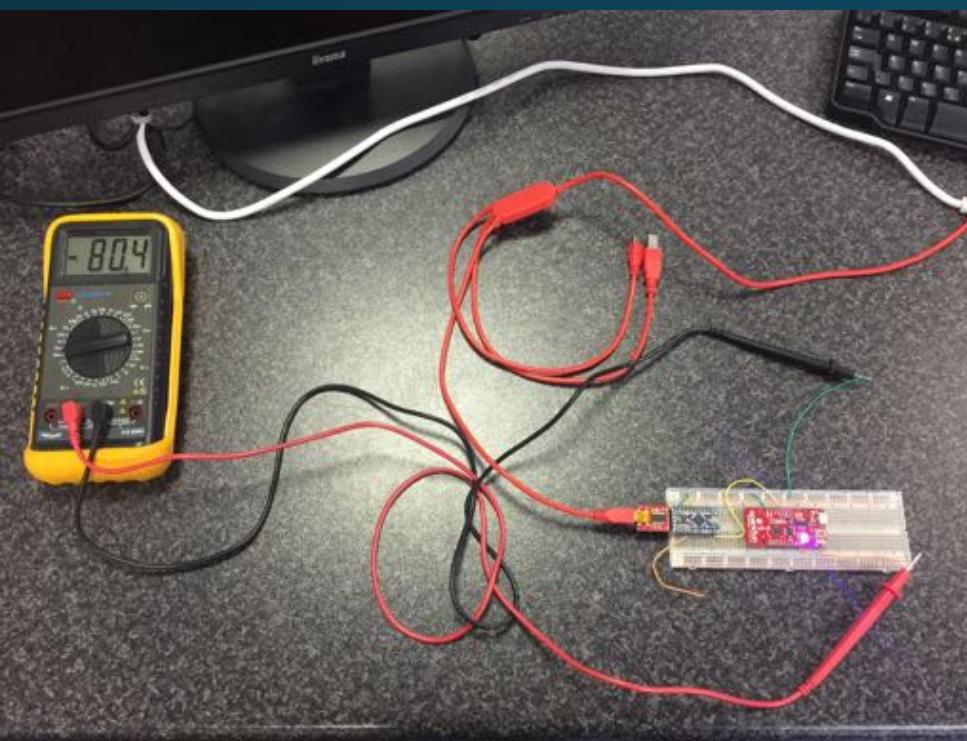
- Getting crafty.





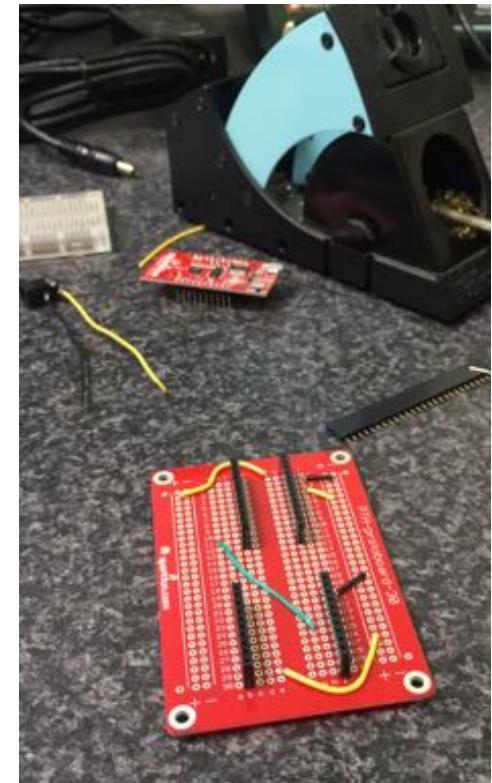
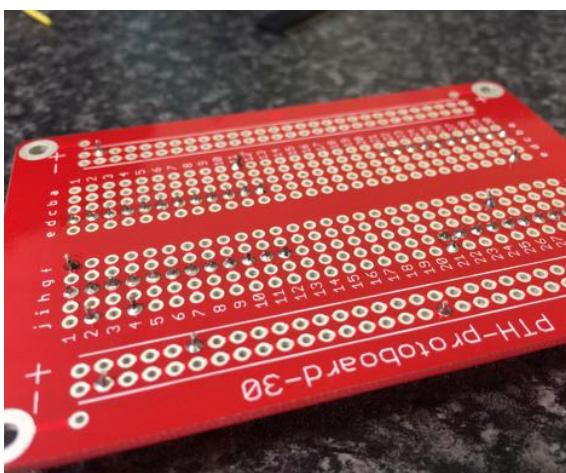
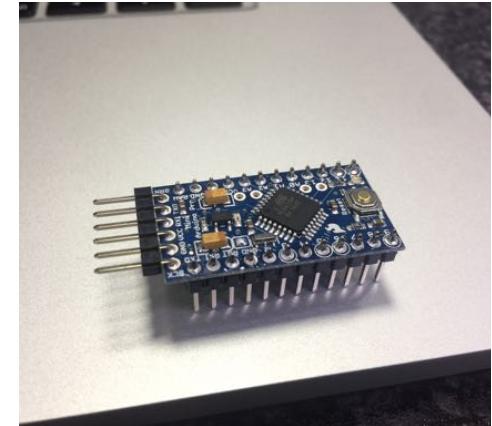
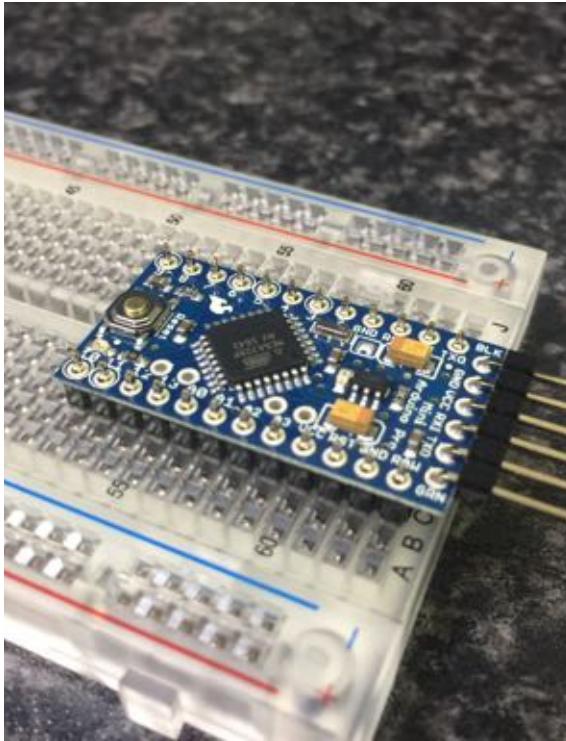
The Journey

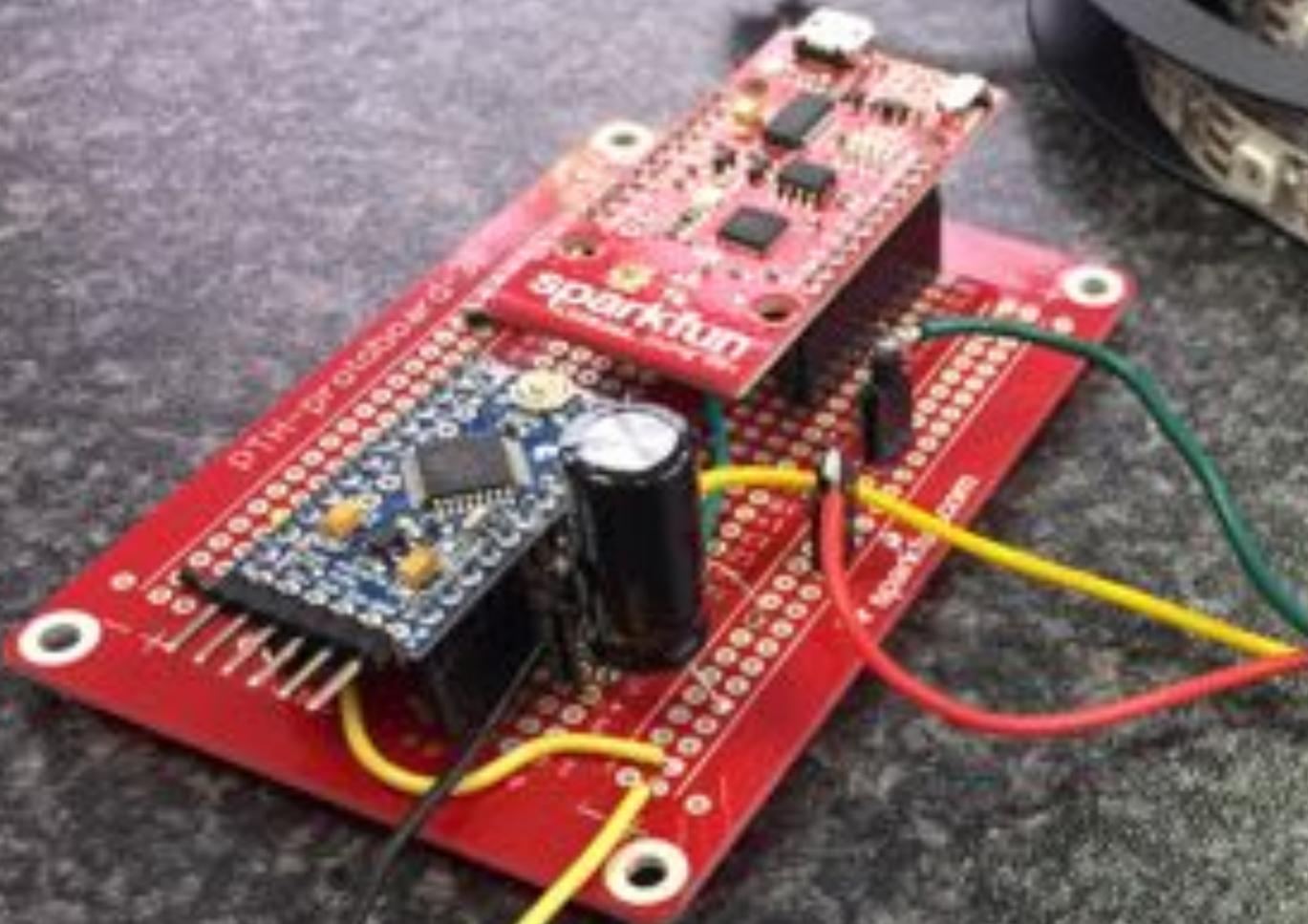
- The Power Struggle.
 - Beyond Physics
 - I/O



The Journey

- Soldering for smiles.





How does it work?

- Thing
 - Finds Wifi
 - Runs Blynk
 - Port > Open Weather > Buf
 - Buffer Strings = Icons (JSON)
 - Transmit to Mini

Icon list

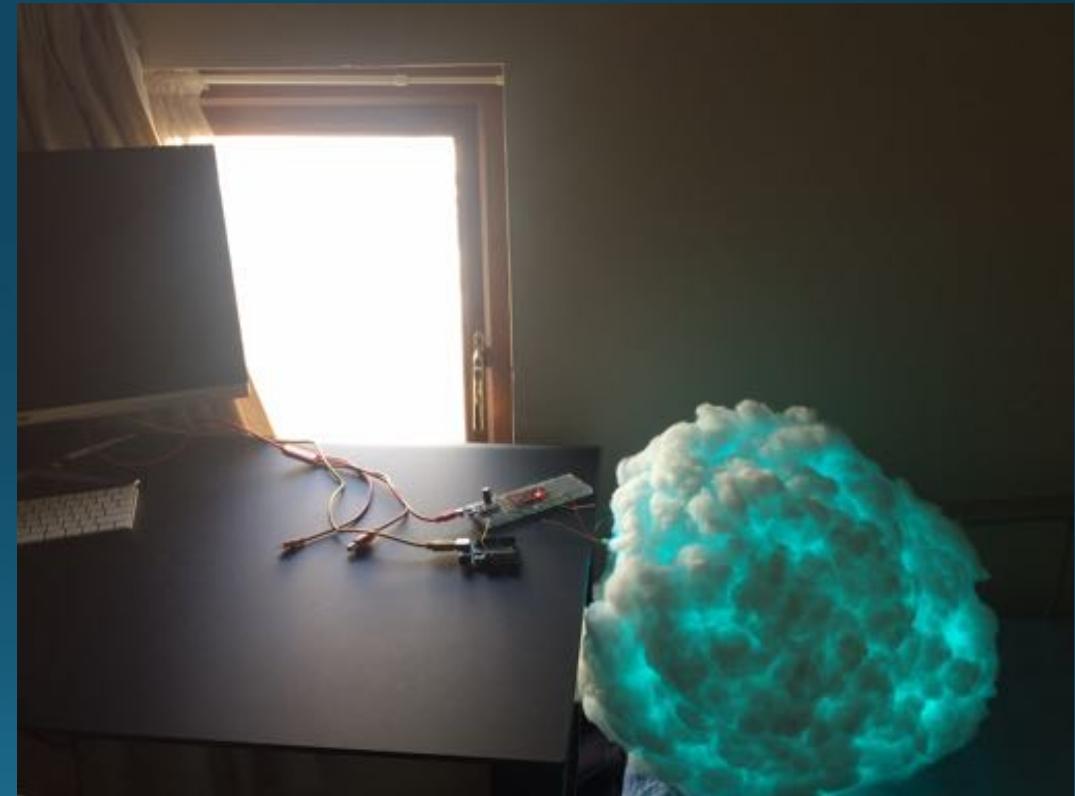
Day icon	Night icon	Description
01d.png 	01n.png 	clear sky
02d.png 	02n.png 	few clouds
03d.png 	03n.png 	scattered clouds
04d.png 	04n.png 	broken clouds
09d.png 	09n.png 	shower rain
10d.png 	10n.png 	rain
11d.png 	11n.png 	thunderstorm
13d.png 	13n.png 	snow
50d.png 	50n.png 	mist

```
605 // Bluesky function
606 void blueSky() {
607     for(int i=0; i<128; i++) {    //for all of the LEDs
608         ledStrip.setPixelColor(i, 0, 170, 175); //set LEDs a sky blue color
609     }
610     ledStrip.show();
611 }
612
613 // White clouds function
614 void whiteClouds() {
615     for(int i=0; i<128; i++) {    //for all of the LEDs
616         ledStrip.setPixelColor(i, 180, 180, 180); //set LEDs white-ish
617     }
618     ledStrip.show();
619 }
620
621
622 // Overcast function
623 void overcast() {
624     for(int i=0; i<128; i++) {    //for all of the LEDs
625         ledStrip.setPixelColor(i, 70, 64, 75); //set LEDs grey-ish
626     }
627     ledStrip.show();
628 }
629
630 // Night function
631 void nighttime() {
632
633     switch ( night_state ) {
634
635         case NIGHT_0:
636             for(int i=0; i<128; i++) {    //for all of the LEDs
637                 ledStrip.setPixelColor(i, 0, 0, 182); //set LEDs dark blue e
638             }
639             ledStrip.show();
640             night_i = 1;
641             night_state = NIGHT_1;
642             led_time = millis();
643             break;
644
645         case NIGHT_1:
646             if ( (millis() - led_time) >= 50 ) {
647                 ledStrip.setPixelColor((night_i-1), 255, 255, 255); //set LEDs white ]
648                 ledStrip.show();
649                 night_state = NIGHT_2;
650                 led_time = millis();
651             }
652         }
653     }
654 }
```

How does it work?

- Mini Pro
- Loops – awaiting a signal to instruct LEDs

Sunny, sunny days...



What was learned?

- Depth of Hardware
- Code Hieroglyphics
- ...interconnectivity...



How was it reinforced?

- Physics
 - Capacitors, Resistance – Circuits
 - !! I/O and delta Draw
- Programming / WebDev
 - Switches, Loops, Libraries

Title: Resistor & Capacitor Circuits

Date: 09/03/2017 at 09:15-11:00

Aim:

To investigate a Resistor Capacitor (RC) Circuit using 3 types of capacitors with 2 types of Resistors, observing the behavior of the Current and the changes in Voltage.

Theory:

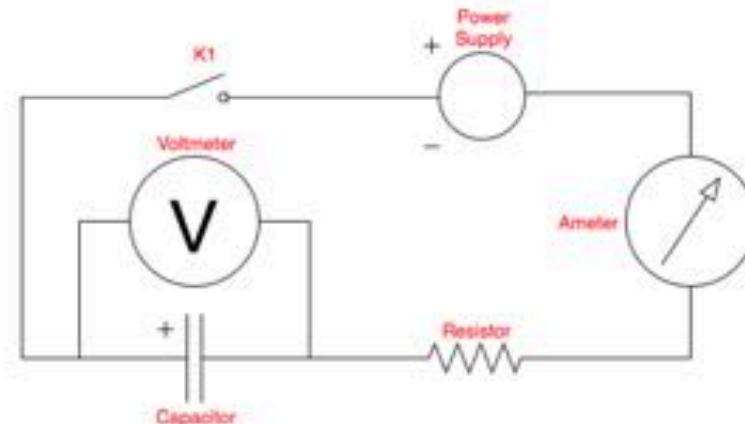
Charge on a capacitor in a RC Circuit increases logarithmically as time increases until the capacitor is fully charged while current in the same circuit drops exponentially.
When a capacitor is discharging in a RC Circuit the capacitance decreases logarithmically until the capacitor is fully depleted while current in the same circuit drops exponentially.

Part A – Charging a Capacitor

Apparatus:

Diagram 1:

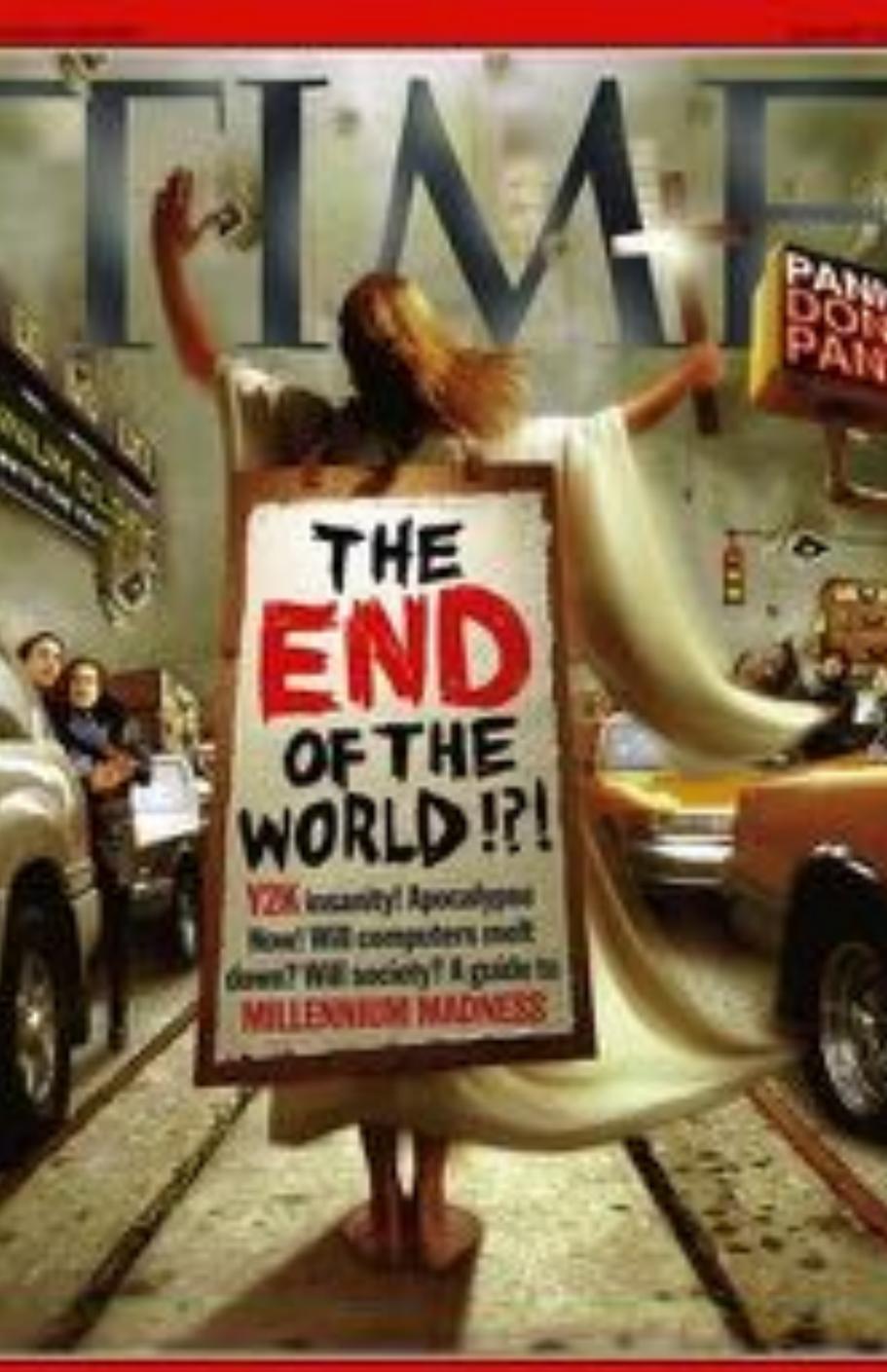
[Drawn by Jon]



Method:

Charging a Capacitor

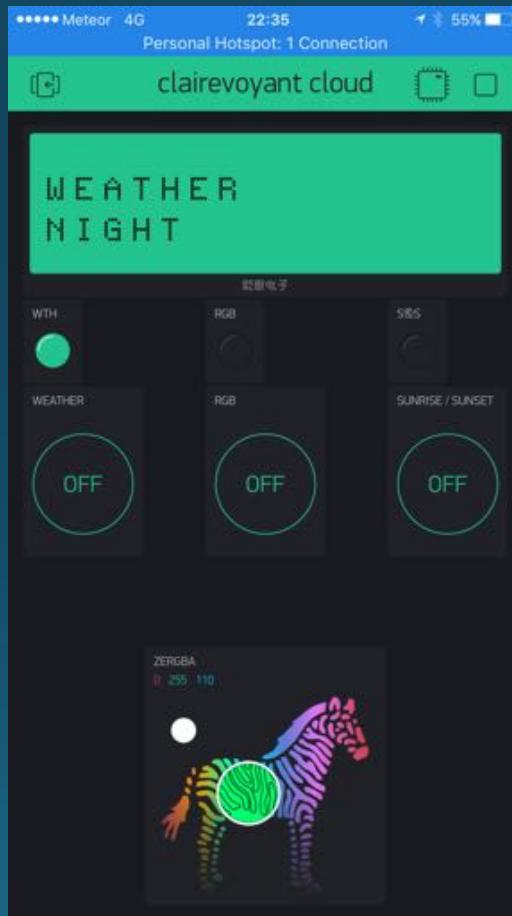
1. For each pairing of:



... something else worth knowing.

- 1,493,943,734 seconds since Jan 01 1970. (UTC)
- Unix Epoch
- Time technically does not change no matter where you are located on the globe
- January 19, 2038 (32-bit)

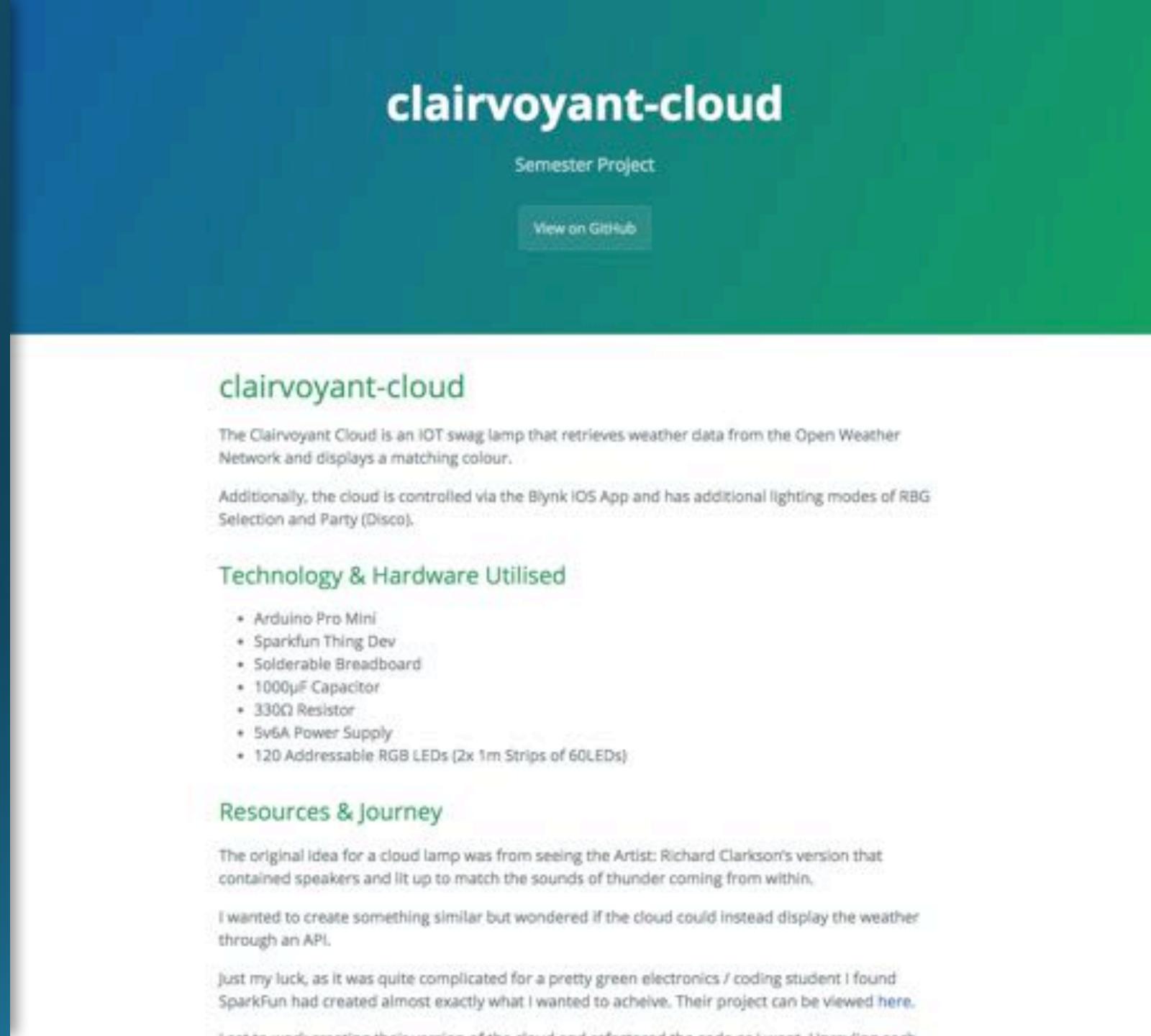
The Final Product – Night Demo





The goods...

- My Repo:
<https://github.com/J-Boz/clairvoyant-cloud>
- My Site: <https://j-boz.github.io/clairvoyant-cloud/>



The screenshot shows the GitHub project page for "clairvoyant-cloud". The title "clairvoyant-cloud" is displayed prominently at the top, followed by the subtitle "Semester Project". A "View on GitHub" button is located below the subtitle. The main content area features a large image of a glowing RGB LED cloud lamp. Below the image, the project name "clairvoyant-cloud" is repeated. A brief description states: "The Clairvoyant Cloud is an IOT swag lamp that retrieves weather data from the Open Weather Network and displays a matching colour. Additionally, the cloud is controlled via the Blynk iOS App and has additional lighting modes of RGB Selection and Party (Disco)." A section titled "Technology & Hardware Utilised" lists the following components: Arduino Pro Mini, Sparkfun Thing Dev, Solderable Breadboard, 1000 μ F Capacitor, 330 Ω Resistor, 5v6A Power Supply, and 120 Addressable RGB LEDs (2x 1m Strips of 60LEDs). Another section, "Resources & Journey", includes a paragraph about the original idea from Richard Clarkson's version and a note about using an API instead. It also mentions that the project was inspired by SparkFun's work.

clairvoyant-cloud

Semester Project

[View on GitHub](#)

clairvoyant-cloud

The Clairvoyant Cloud is an IOT swag lamp that retrieves weather data from the Open Weather Network and displays a matching colour. Additionally, the cloud is controlled via the Blynk iOS App and has additional lighting modes of RGB Selection and Party (Disco).

Technology & Hardware Utilised

- Arduino Pro Mini
- Sparkfun Thing Dev
- Solderable Breadboard
- 1000 μ F Capacitor
- 330 Ω Resistor
- 5v6A Power Supply
- 120 Addressable RGB LEDs (2x 1m Strips of 60LEDs)

Resources & Journey

The original idea for a cloud lamp was from seeing the Artist: Richard Clarkson's version that contained speakers and lit up to match the sounds of thunder coming from within. I wanted to create something similar but wondered if the cloud could instead display the weather through an API. Just my luck, as it was quite complicated for a pretty green electronics / coding student I found SparkFun had created almost exactly what I wanted to achieve. Their project can be viewed [here](#).

[Live Demo](#)

Clairvoyant Cloud

Thank you.
Any questions?

