



The City College of New York  
*The Grove School of Engineering*

*Presented to Professor Ihab DARWISH*

**SMAI**

*by*

*Anna Taylor and Andrii Aluchko*

**Smart Mirror**

**AI**



## Abstract

To design an advanced Smart Mirror technology which is also known as **Magic Mirror** that provides the best advanced user interface and the best user experience.

## What is Magic Mirror?

It is our Smart Mirror. So, if you place the glass' on a dark surface, you will see just a mirror. When you put a light source behind it, you can see through it. If you put a black page on the monitor, add some widgets on them, they seem to be part of the mirror's reflection. A Smart Mirror is a great example of an iOT device.

The mirror offers natural mode of interaction through which users can use and experience the AI chat assistant. The AI has speech recognition and can respond to a variety of commands. Ability to interact with the personal assistant through Google chat assistant that provides the most natural and convenient mode of interaction. **Our Smart Mirror** is assembled using a one-way mirror, computer monitor using HDMI socket along with HDMI cable, raspberry pi 4, USB microphone and a speaker. As Voice assistants become more powerful in their functionality, we realize that there is a big security risk here. We are going to attempt a surfing attack using solid materials as transmission media. The attacker's goal is to remotely converse with victim's smart device to inject unauthorized voice commands. Discussions and defenses to minimize and prevent surfing attacks in rapidly growing functionality of voice assistants in iOT devices.

New threat has been discovered from a recent research from Washington University where scientists have demonstrated that these voice assistant systems are susceptible to signal injection at the inaudible frequencies.



# The City College of New York

## The Grove School of Engineering

## Raspberry Pi set

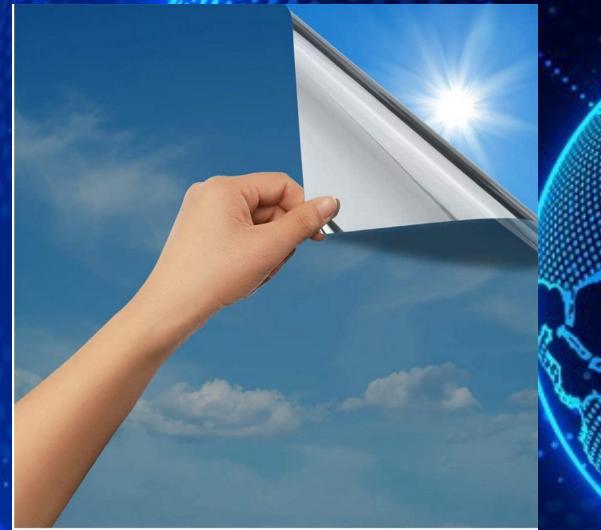


HDMI Cable



## Parts List

### Two way mirror



### Old Dell Monitor



### Raspberry Pi Camera



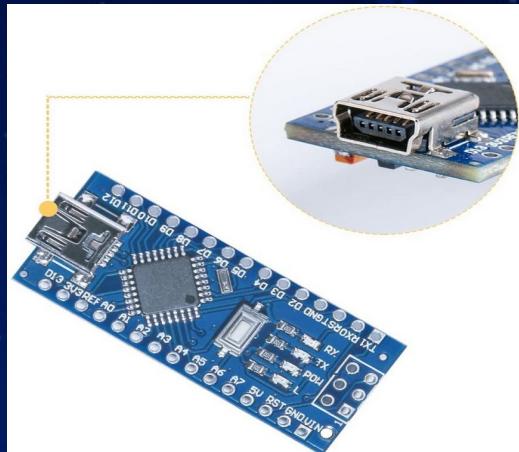


# The City College of New York

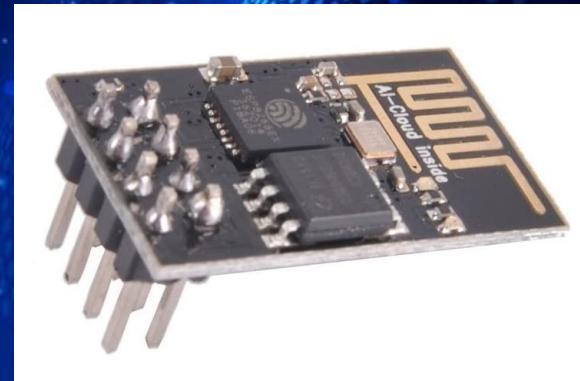
## The Grove School of Engineering

### Parts List

Arduino Nano



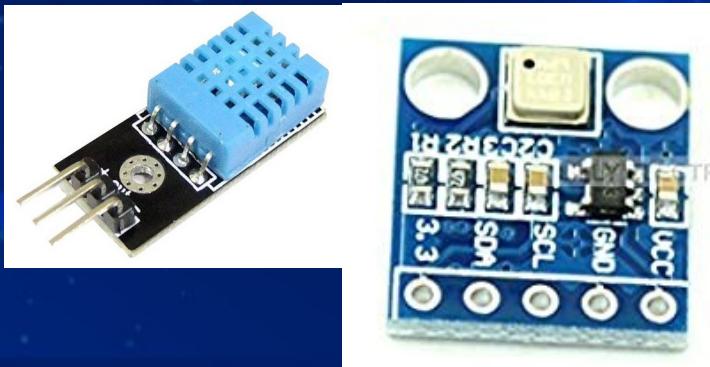
ESP8266



Voltage regulator



Environmental Sensors



Motion sensor



Relay



LED strip



Gas Sensors





## SOFTWARE and PLATFORMS:

- 01 Raspberry Pi  
Raspbian (Magic Mirror OS)  
Magic Mirror Module

Google Assistant

MQTT module

- 03 3d Printing software

- 04 Arduino

# Smart Mirror AI



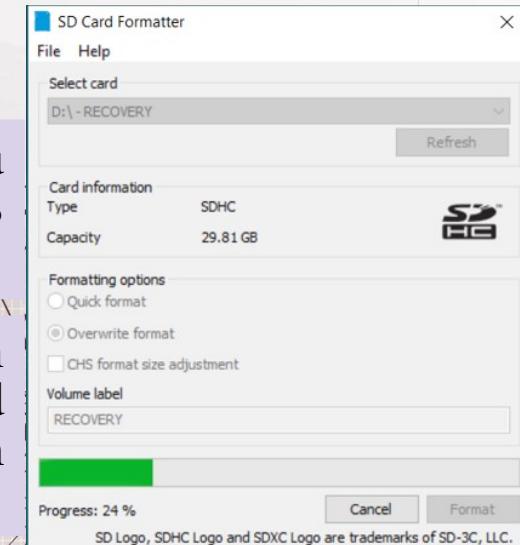
# The City College of New York

## The Grove School of Engineering

## INSTALLING RASPBIAN OS

The Raspbian OS was flashed onto the 32 GB Micro SD Card. You could find more information on how to install the latest Raspbian OS on the [Raspberry Pi site](#).

All the data was transferred onto SD card, that was extracted from NOOBS download. Next mouse, keyboard and HDMI were plugged into a raspberry pi. USB power as well as SD card was removed from PC and inserted into raspberry pi.



# Smart Mirror AI



# The City College of New York

## *The Grove School of Engineering*

# INSTALLING MAGIC MIRROR

Sunday, November 22, 2020

12:16<sup>34</sup> pm

US HOLIDAYS

- Thanksgiving Day  
In 4 days
- Christmas  
In a month



NEW YORK

→ 6 NE - 6:51 am

Cloudy 48.3°

Feels like 40°

Magic Mirror is an open-source modular smart mirror platform developed by MichMich. It's has a great community and support which makes this a solid project

# Smart Mirror AI



# The City College of New York

## The Grove School of Engineering

# INSTALLING MAGIC MIRROR

# MagicMirror<sup>2</sup>

The open source modular smart mirror platform.

Dependency Status DevDependency Status CI best practices build vulnerabilities 3

MagicMirror<sup>2</sup> is an open source modular smart mirror platform. With a growing list of installable modules, the MagicMirror<sup>2</sup> allows you to convert your hallway or bathroom mirror into your personal assistant. MagicMirror<sup>2</sup> is built by the creator of the original MagicMirror with the incredible help of a [growing community of contributors](#).

MagicMirror<sup>2</sup> focuses on a modular plugin system and uses [Electron](#) as an application wrapper. So no more web server or browser installs necessary!

### Documentation

For the full documentation including [installation instructions](#), please visit our dedicated documentation website:  
<https://docs.magicmirror.builders>.

Upon boot-up of the Raspberry Pi, complete the remaining setup, and ensure you are connected to the internet via Wi-Fi

```
pi@raspberrypi:~/MagicMirror
File Edit Tabs Help
pi@raspberrypi:~$ cd ~
pi@raspberrypi:~$ cd MagicMirror/
pi@raspberrypi:~/MagicMirror$ npm start

> magicmirror@2.13.0 start /home/pi/MagicMirror
> DISPLAY="${DISPLAY:=:0}" ./node_modules/.bin/electron js/electron.js

[2020-11-22 12:23:33.245] [LOG] Starting MagicMirror: v2.13.0
[2020-11-22 12:23:33.252] [LOG] Loading config ...
[2020-11-22 12:23:33.256] [LOG] Loading module helpers ...
[2020-11-22 12:23:33.258] [LOG] No helper found for module: clock.
[2020-11-22 12:23:33.678] [LOG] Initializing new module helper ...
[2020-11-22 12:23:33.679] [LOG] Module helper loaded: calendar
[2020-11-22 12:23:33.680] [LOG] No helper found for module: currentweather.
[2020-11-22 12:23:33.680] [LOG] No helper found for module: weatherforecast.
[2020-11-22 12:23:34.021] [LOG] Initializing new module helper ...
[2020-11-22 12:23:34.021] [LOG] Module helper loaded: MMM-NowPlayingOnSpotify
[2020-11-22 12:23:34.022] [LOG] All module helpers loaded.
[2020-11-22 12:23:34.150] [LOG] Starting server on port 8080 ...
[2020-11-22 12:23:34.164] [LOG] Server started ...
[2020-11-22 12:23:34.165] [LOG] Connecting socket for: calendar
[2020-11-22 12:23:34.165] [LOG] Starting node helper for: calendar
[2020-11-22 12:23:34.166] [LOG] Connecting socket for: MMM-NowPlayingOnSpotify
```

Here's the main snippet

# Smart Mirror AI



# The City College of New York

## The Grove School of Engineering

# INSTALLING MAGIC MIRROR

Download and install the latest Node.js version:

```
curl -sL https://deb.nodesource.com/setup_10.x | sudo -E bash -  
sudo apt install -y nodejs
```

Clone the repository and check out the master branch:

```
git clone https://github.com/MichMich/MagicMirror
```

Enter the repository:

```
cd MagicMirror/
```

Install the application:

```
npm install
```

Make a copy of the config sample file:

```
cp config/config.js.sample config/config.js
```

Start the application (Recommended for this setup):

```
npm run start
```

For Server Only use:

```
npm run server
```

**Note:** You could decrease the MagicMirror size of the font with CTRL + -

# Smart Mirror AI



# The City College of New York

## *The Grove School of Engineering*

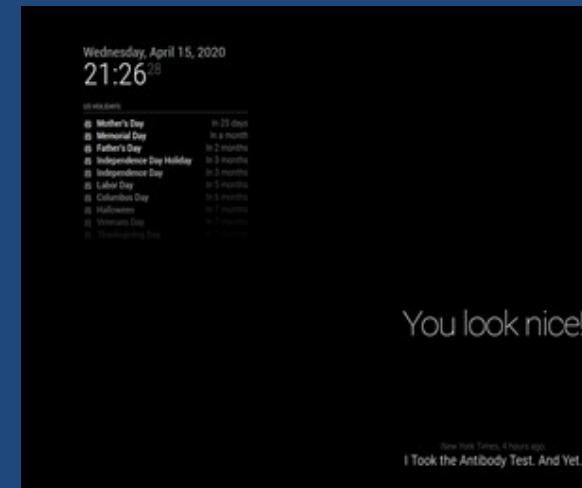
# MagicMirrorOS

<https://github.com/guysoft/MagicMirrorOS>

OS image  
flashing



Running MagicMirror OS





## Installing Face Recognition Module – Personalizing your

The Face detection module is a nice feature to complement the touch interaction. The use cases for this are endless spanning to home automation. In the below example the Face Detection module was used.



MMM-Face-recognition-SMAI module requires a few dependencies before operating

Steps are outlined below

**Smart Mirror**  
AI



# The City College of New York

## The Grove School of Engineering

# Installing Face Recognition Module – Personalizing your Mirror

## Step 1 – Install the module

In your MagicMirror directory:

```
git clone https://github.com/EbenKouao/MMM-Face-Recognition-SMAI.git  
cd MMM-Face-rec  
npm install
```

## Step 2 – Add files to the Config.js

Here is an example for an entry in `config.js`

```
{  
  module: "MMM-Face-Recognition-SMAI",  
  position: "top_right",  
  config: {  
    //prompt: "Put in your own text"  
  }  
}
```

### Changing your profile image:

By default, the Profile Image is Tony Stark. However this could be changed to yours using the steps below:

#### 2) Changing your profile:

- The beta version of this module is restricted to only 1 user.

Go into the directory (preferably via GUI)

```
cd /home/pi/MagicMirror/modules/MMM-Face-Recognition-SMAI/public
```

Replace the two .png images with your desired face while keeping the same naming format. Using the below naming convention.

- face.png
- id.png e.g Tony-id.png – This is what would be displayed on the display, being Tony.

Watch your faceID pop up on display every-time you're detected. Using this you could build your own applications.

The face detection interacts with the Python script to recognise you and display your name. Since Python runs independently of the MagicMirror with a bit of programming you could program the face detection to trigger an event externally e.g. Turn on Smart lights.

### How it works the face detection module works:

Here are the 3 main files you would need to understand:

- 1) SMAI-face-detection.py – OpenCV face detection Python script that constantly tries to identify persons based on the images already provided. (You'll only need to program this if you wish to develop use cases beyond this project scope).
- 2) MMM-Face-Recognition-SMAI.js – Displays the profile image detected as well as the name
- 3) Sample.txt – Works as a temporary buffer. Stores the name of the identified person from SMAI-face-detection.py as a temporary variable and read to the Face-recognition-module.



# Smart Mirror AI



# Google Assistant Module-AI Integration

Upon using the google cloud platform and Console we have registered our app to integrate Google Assistant with Magic Mirror

The screenshot shows the Google Cloud Platform API Credentials page for the project "MagicMirror akeysny". The left sidebar lists "APIs & Services" with options like Dashboard, Library, Credentials (selected), OAuth consent screen, Domain verification, and Page usage agreements. The main content area shows two sections: "API Keys" and "OAuth 2.0 Client IDs".

**API Keys:**

| Name                                   | Creation date | Restrictions | Key                     |
|--|---------------|--------------|-------------------------|
| Browser key (auto created by Firebase) | Nov 21, 2020  | None         | AizaSyCCJN...ZPuMVLoCkU |

**OAuth 2.0 Client IDs:**

| Name                                   | Creation date | Type                 | Client ID            |
|--|---------------|----------------------|----------------------|
| TV client 1                            | Nov 22, 2020  | TV and Limited input | 300817278403-tue9... |
| magicmirror-akeysny-magicmirror-muSkel | Nov 21, 2020  | Desktop              | 300817278403-121j... |
| New Actions on Google App              | Nov 21, 2020  | Web application      | 300817278403-qe7v... |

Once we have our app registered- we can exec our commands as demonstrated below

# Smart Mirror AI



# Google Assistant Module- AI Integration

File Edit Tabs Help

pi@raspberrypi:~ \$





# The City College of New York

## The Grove School of Engineering

# The application is endless. Devices and industries are getting smarter!

magic mirror mich mich

All Images Videos News Maps Shopping Settings

All Regions Safe Search: Moderate Any Time

[GitHub - MichMich/MagicMirror: MagicMirror<sup>2</sup> is an open ...](#)  
https://github.com/MichMich/MagicMirror  
MagicMirror<sup>2</sup> is an open source modular smart **mirror** platform. With a growing list of installable modules, the MagicMirror<sup>2</sup> allows you to convert your hallway or bathroom **mirror** into your personal assistant. MagicMirror<sup>2</sup> is built by the creator of the original MagicMirror with the incredible help of a growing community of contributors..  
MagicMirror<sup>2</sup> focuses on a modular plugin system and ...

[Magic Mirror Michigan - Mirror Photo Booth Rental](#)  
https://www.magicmirrormichigan.com  
The **Magic Mirror** is the original **Mirror Me**, not a knockoff **mirror**. This has all the latest and greatest technology and the one that has won all the awards! We only do quality at **Magic Mirror Michigan**. We use DSLR cameras, DNP sub dye printers and the most beautiful sought after premium backdrops in the industry. We do not cut corners!

Images for magic mirror mich mich



After installation we would have to integrate our modules into our Config.js file and this is how we did this.

# Smart Mirror AI



# Installing MQTT module

## Installation commands

```
git clone https://github.com/ottopaulsen/MMM-MQTT
cd MMM-MQTT
npm install
```

## config.js

```
{
  module: 'MMM-MQTT',
  position: 'bottom_left',
  header: 'MQTT',
  config: {
    logging: false,
    useWildcards: false,
    mqttServers: [
      {
        address: 'localhost', // Server address or IP address
        port: '1883', // Port number if other than default
        user: 'user', // Leave out for no user
        password: 'password', // Leave out for no password
        subscriptions: [
          {
            topic: 'smoky/1/inside/temperature', // Topic to look for
            label: 'Temperature', // Displayed in front of value
            suffix: '°C', // Displayed after the value
            decimals: 1, // Round numbers to this number of decimals
            sortOrder: 10, // Can be used to sort entries in the same table
            maxAgeSeconds: 60, // Reduce intensity if value is older
          }
        ]
      }
    ]
  }
}
```



# Installing MQTT module

```
mqttServers: [
  {
    address: 'mqtt.eclipseprojects.io', // Server address or IP address
    //port: '',                      // Port number if other than default
    //user: '',                      // Leave out for no user
    //password: '',                  // Leave out for no password
    subscriptions: [
      {
        topic: 'luch/sens1', // Topic to look for
        label: 'Temperature', // Displayed in front of value
        suffix: 'C',          // Displayed after the value
        decimals: 1,           // Round numbers to this number of decimals
        sortOrder: 10,          // Can be used to sort entries in the same table
        maxAgeSeconds: 60,       // Reduce intensity if value is older
        jsonpointer: '/temperature_1',
        colors: [               // Value dependent colors
          { upTo: -10, value: "blue", label: "blue", suffix: "blue" },
          { upTo: 0, value: "#00ccff", label: "#00ccff", suffix: "#00ccff" },
          { upTo: 10, value: "yellow" },
          { upTo: 30, label: "green", suffix: "green" },
          { upTo: 100, label: "red" }, // The last one is used for higher values too
        ],
      },
    ],
  },
]
```



# The City College of New York

## The Grove School of Engineering

## Building Mirror Frame and Assembly Parts:





The City College of New York  
*The Grove School of Engineering*

## Building Mirror Frame and Assembly Parts:

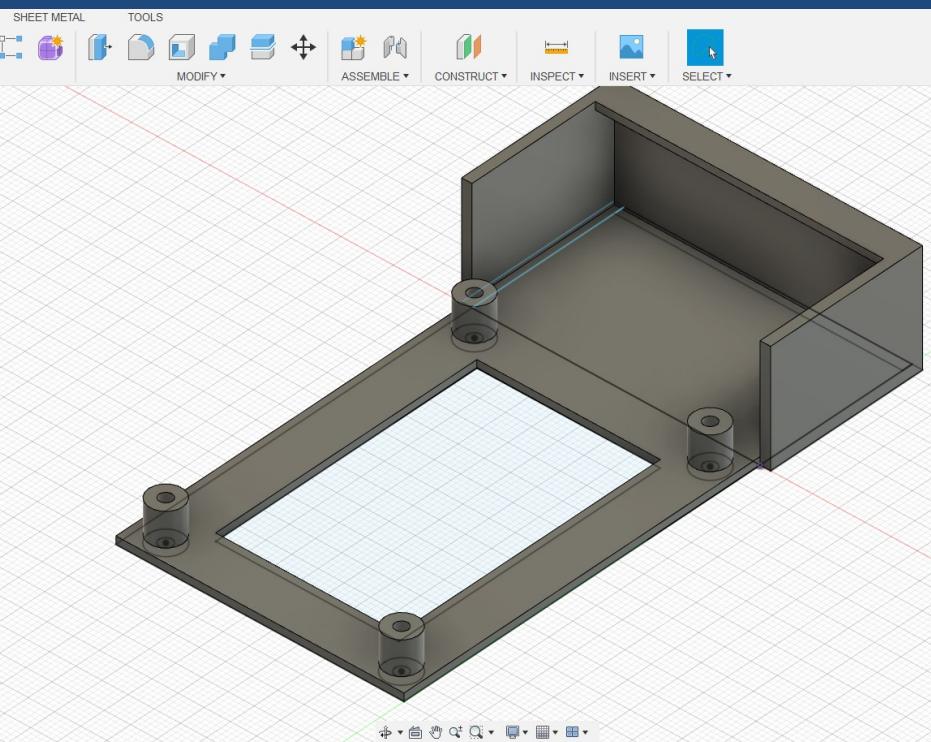




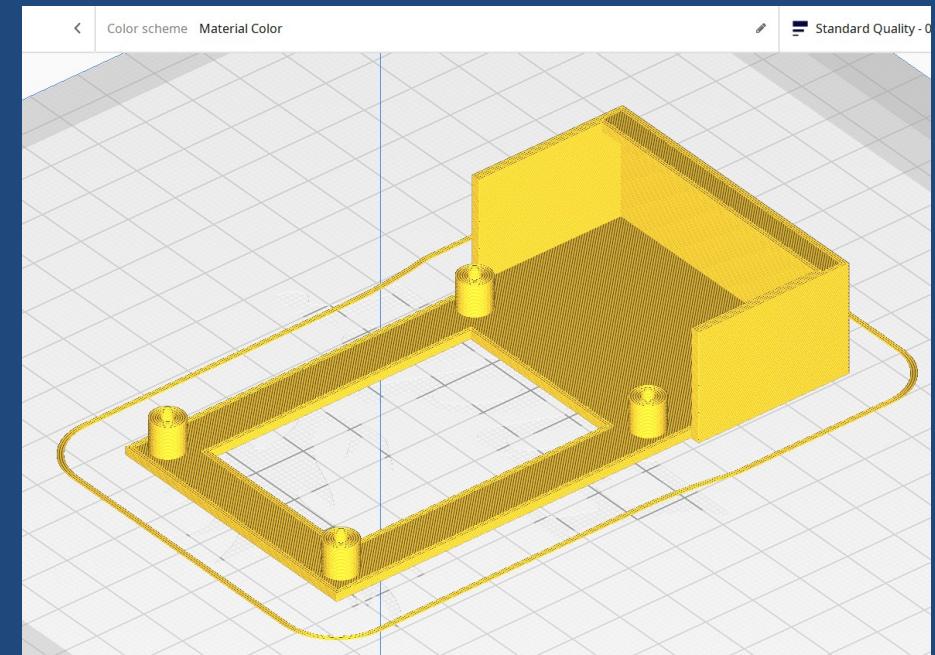
The City College of New York  
*The Grove School of Engineering*

# Electric Isolation and 3D printing

AutoDesk Fusion 360



Ultimaker Cura slicing





The City College of New York  
*The Grove School of Engineering*

## Building Mirror Frame and Assembly Parts:





# The City College of New York

## The Grove School of Engineering

### Finished Result



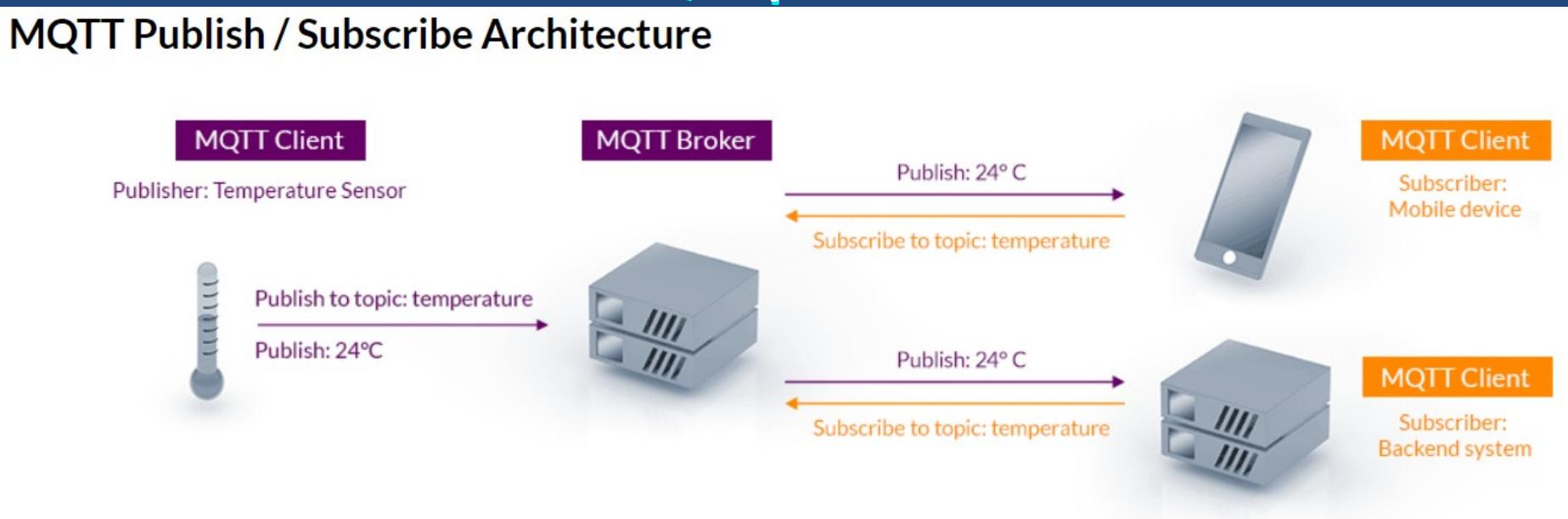


# The City College of New York

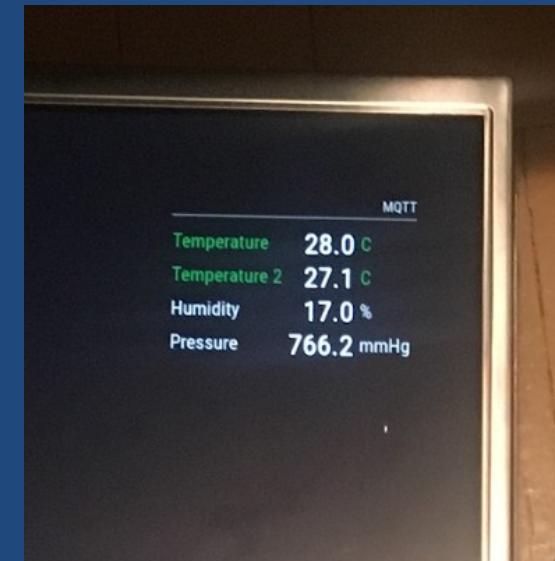
## The Grove School of Engineering

# MQTT protocol

## MQTT Publish / Subscribe Architecture



[mqtt.eclipseprojects.io](http://mqtt.eclipseprojects.io)

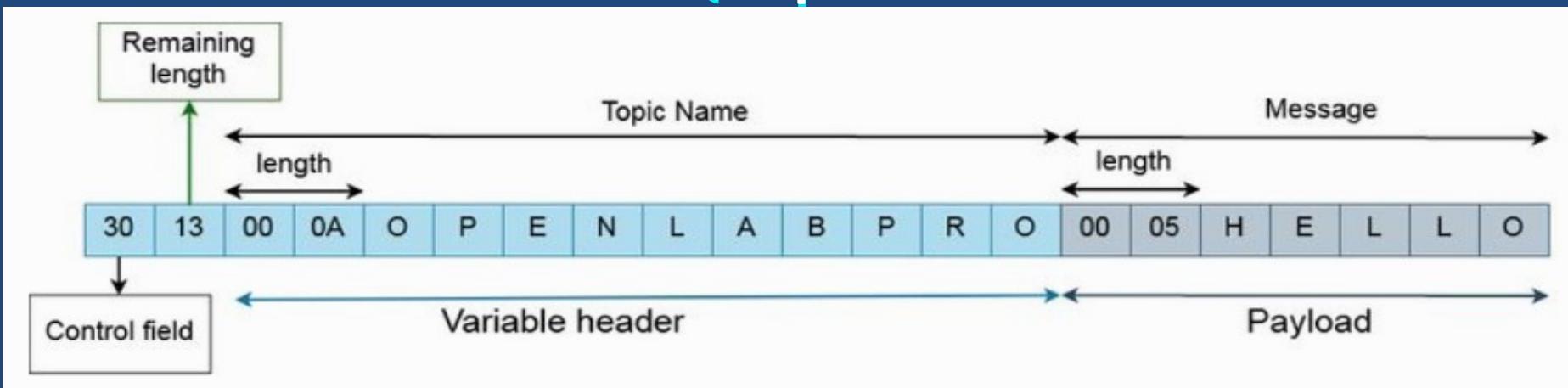




# The City College of New York

## The Grove School of Engineering

### MQTT packet



MQ Telemetry Transport Protocol, Publish Message

- Header Flags: 0x30, Message Type: Publish Message, QoS Level: At most once delivery (Fire and Forget)  
0011 .... = Message Type: Publish Message (3)  
.... 0.... = DUP Flag: Not set  
.... .00. = QoS Level: At most once delivery (Fire and Forget) (0)  
.... ...0 = Retain: Not set
- Msg Len: 105
- Topic Length: 10
- Topic: luch/sens1
- Message: `7b2274797065223a22726573706f6e7365222c2274656d70657261747572655f31223a32...`

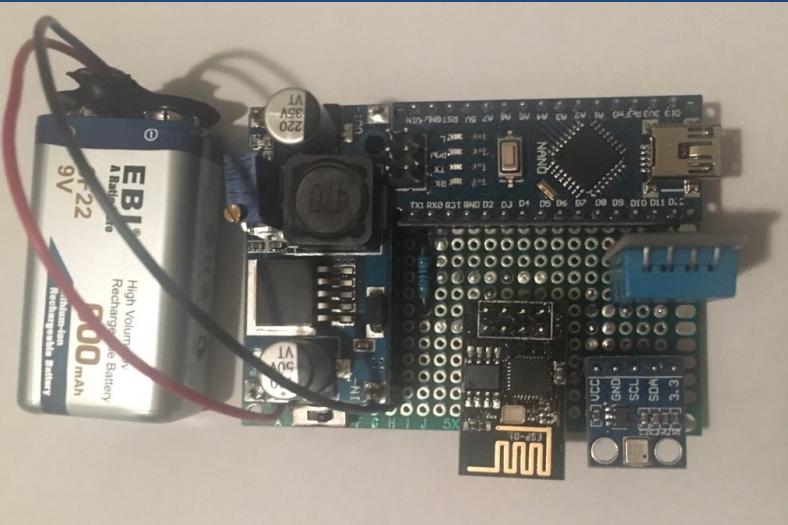
|      |   |                     |
|------|---|---------------------|
| 0000 | c8 f7 33 5a 9a ce 50 c7 bf 1d d0 c4 08 00 45 00 | ..3Z..P. ....E.     |
| 0010 | 00 93 12 71 40 00 36 06 93 87 89 87 53 d9 c0 a8 | ...q@-6- ....S...   |
| 0020 | 00 64 07 5b c7 a0 5b 1e cc 64 13 84 d1 8a 50 18 | .d.[...[. .d.....P. |
| 0030 | 01 f6 89 19 00 00 30 69 00 0a 6c 75 63 68 2f 73 | .....0i ..luch/s    |
| 0040 | 65 6e 73 31 7b 22 74 79 70 65 22 3a 22 72 65 73 | ens1{"ty pe":"res   |
| 0050 | 70 6f 6e 73 65 22 2c 22 74 65 6d 70 65 72 61 74 | ponse"," temperat   |
| 0060 | 75 72 65 5f 31 22 3a 32 37 2c 22 68 75 6d 69 64 | ure_1":2 7,"humid   |
| 0070 | 69 74 79 5f 31 22 3a 31 31 2c 22 74 65 6d 70 65 | ity_1":1 1,"tempe   |
| 0080 | 72 61 74 75 72 65 5f 32 22 3a 32 37 2c 22 70 72 | nature_2 ":"27,"pr  |
| 0090 | 65 73 73 75 72 65 5f 32 22 3a 31 30 32 36 32 31 | essure_2 ":"102621  |
| 00a0 | 7d  | }                   |



# The City College of New York

## The Grove School of Engineering

### Environmental Sensor module

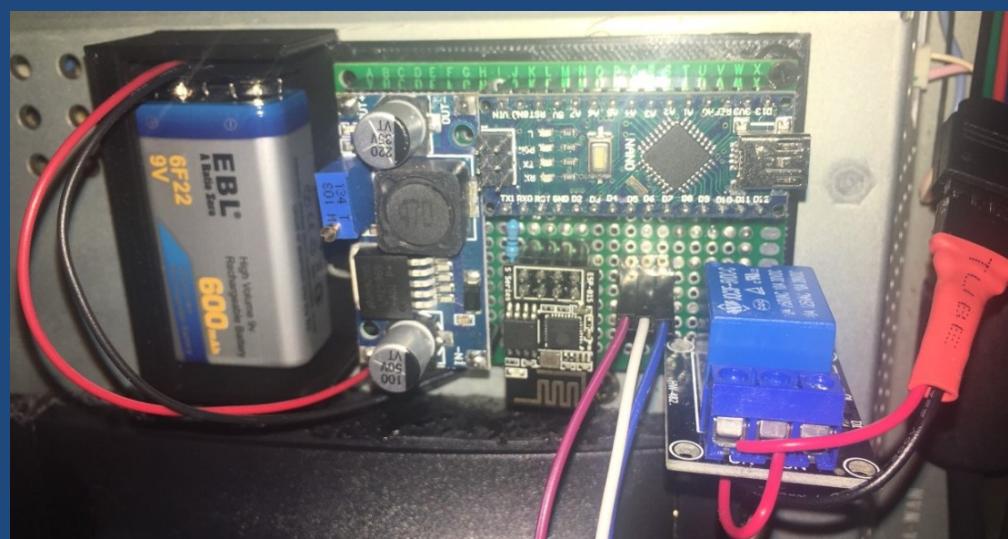


### IoT Devices

### Gas Sensor module



### Light Switch Module





# The City College of New York

## The Grove School of Engineering

# IOT Devices - Code, Prototyping, and Soldering

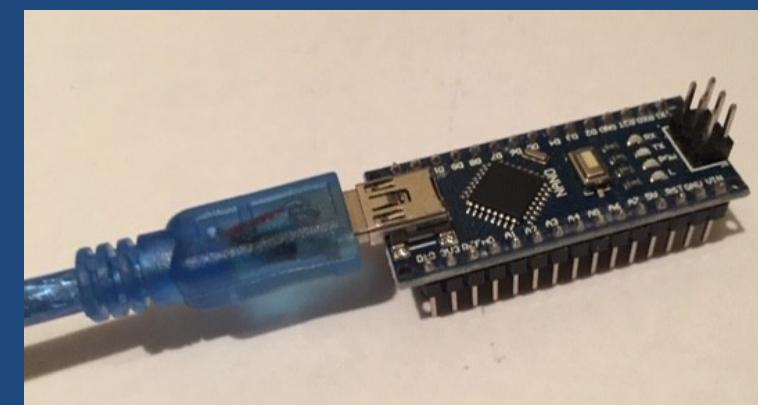
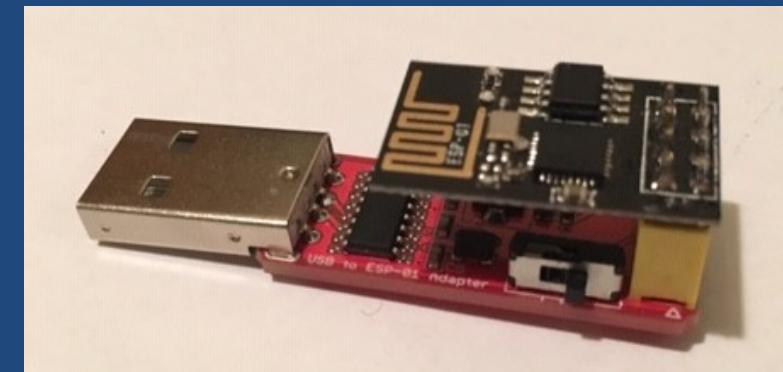
```
#include <ArduinoJson.h>

#include <Wire.h>
#include <Adafruit_BMP085.h>
#include <dht_nonblocking.h>
#define DHT_SENSOR_TYPE DHT_TYPE_11
Adafruit_BMP085 bmp;
static const int DHT_SENSOR_PIN = 11;
DHT_nonblocking dht_sensor(DHT_SENSOR_PIN, DHT_SENSOR_TYPE);

String message = "";
bool messageReady = false;

void setup() {
  Serial.begin(9600);
  bmp.begin();
}

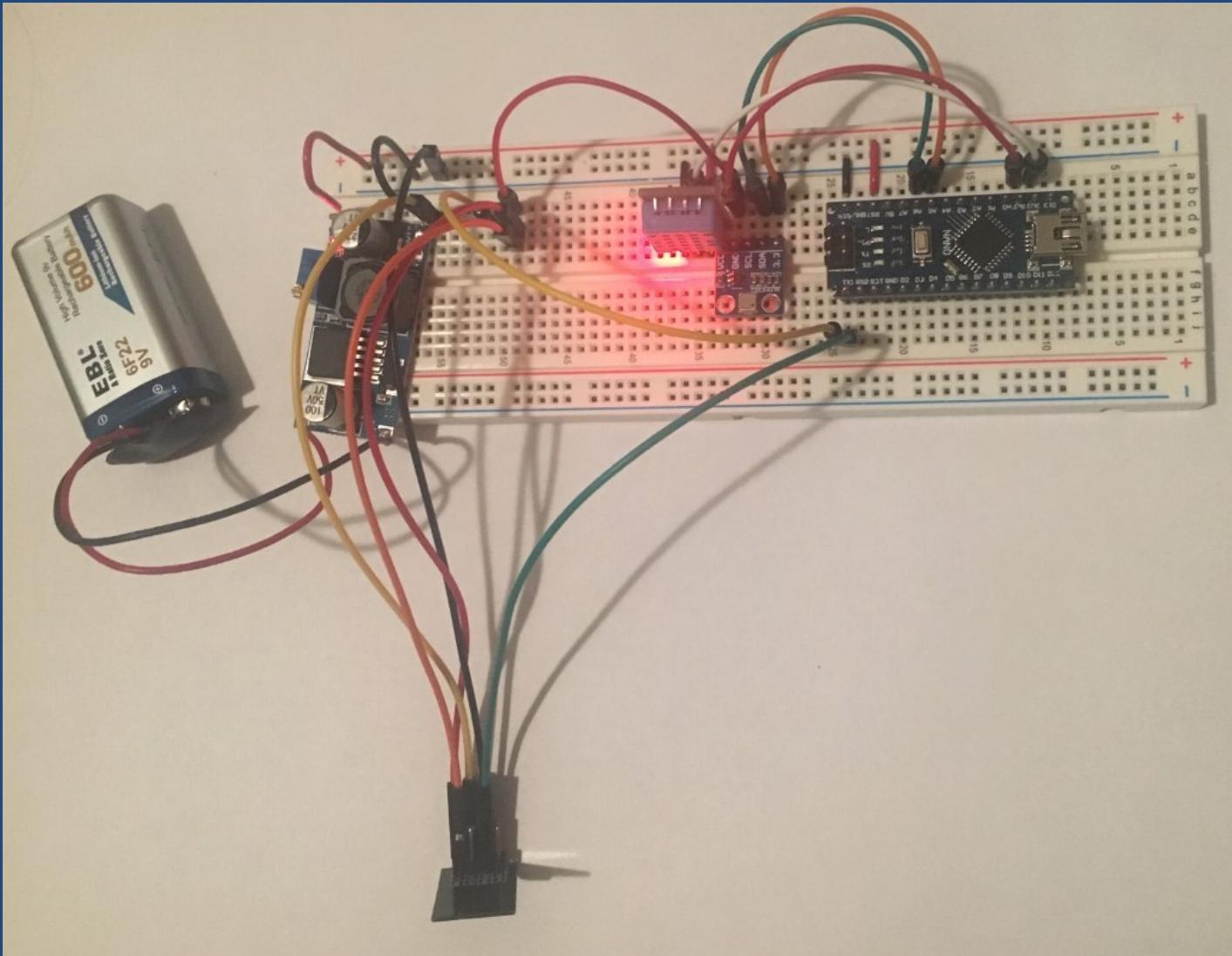
void loop() {
  // Monitor serial communication
  while(Serial.available()) {
    message = Serial.readString();
    messageReady = true;
  }
  // Only process message if there's one
  if(messageReady) {
    // The only messages we'll parse will be formatted in JSON
    DynamicJsonDocument doc(1024); // ArduinoJson version 6+
    // Attempt to deserialize the message
    DeserializationError error = deserializeJson(doc,message);
  }
}
```





The City College of New York  
*The Grove School of Engineering*

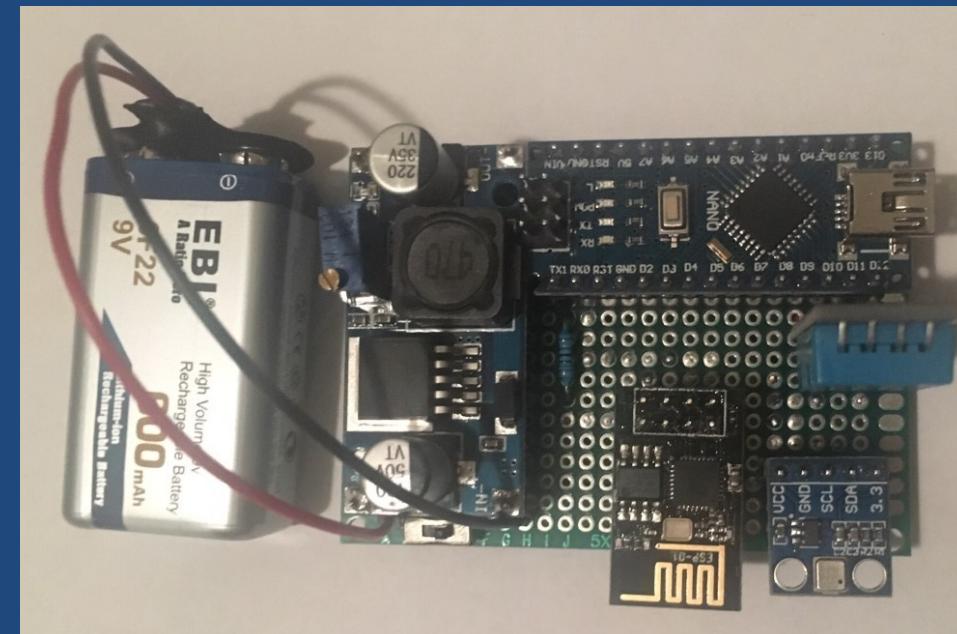
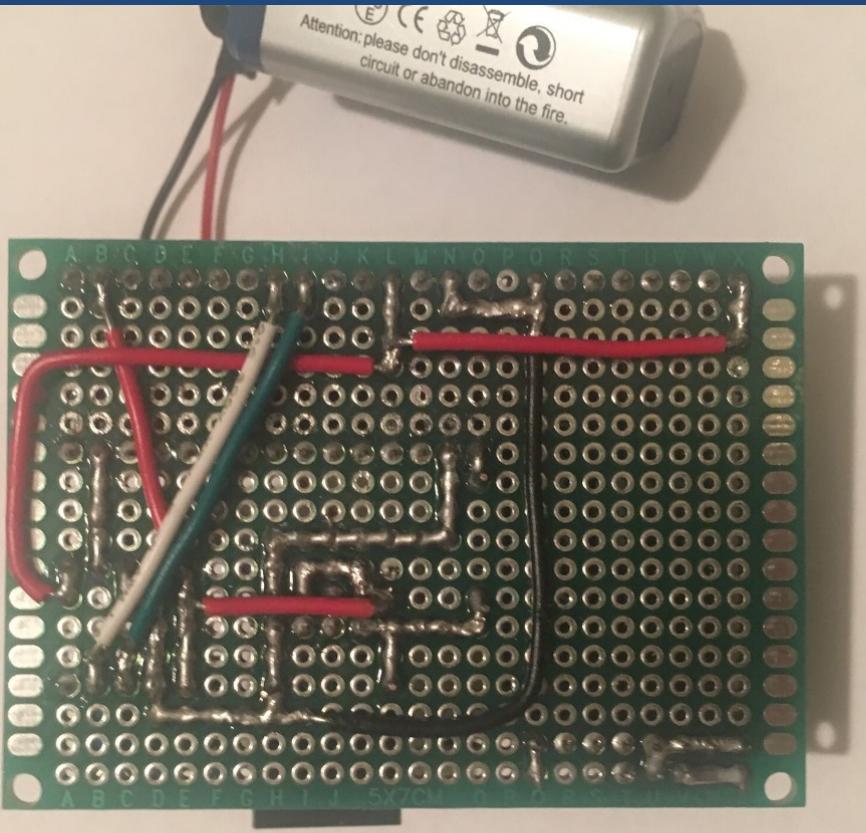
# IOT Devices - Code, Prototyping, and Soldering





The City College of New York  
*The Grove School of Engineering*

# IOT Devices - Code, Prototyping, and Soldering

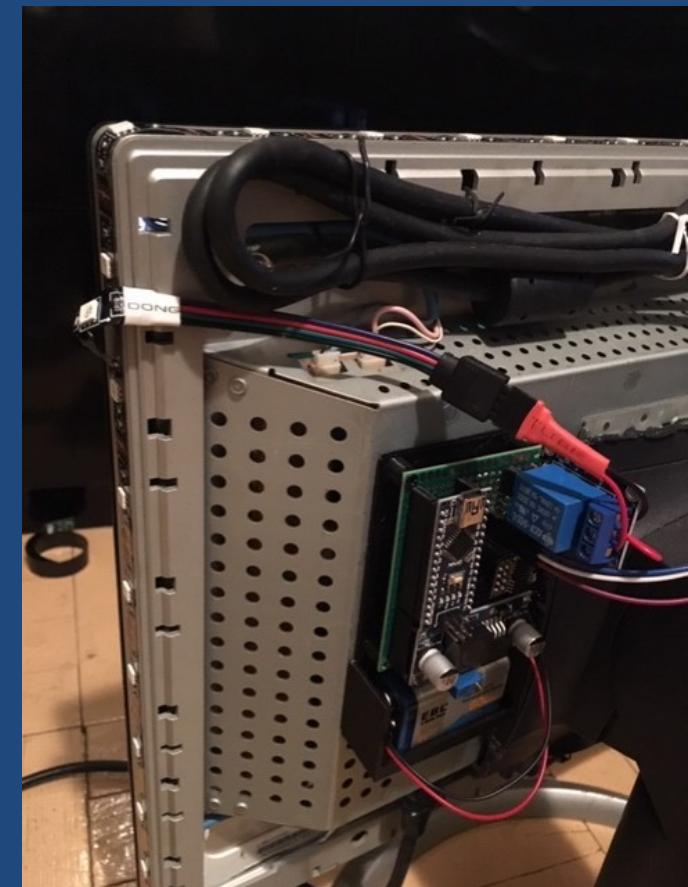
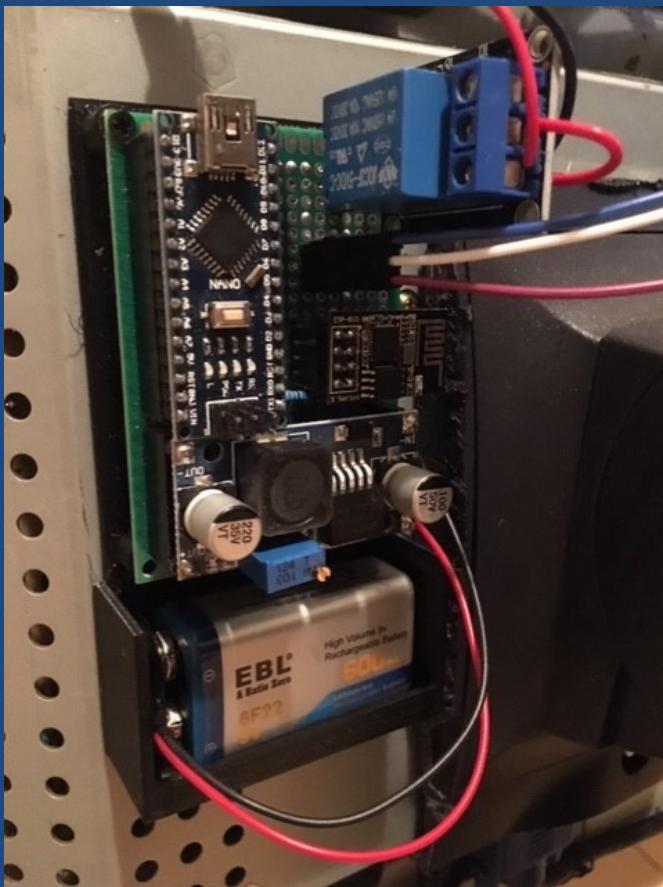




The City College of New York  
*The Grove School of Engineering*

## IOT Devices - Light Switch Module

```
{"type": "request", "mode": "manual", "status": "off"}  
  
{ "type": "response", "mode": "manual", "status": "off"}
```





# The City College of New York

## The Grove School of Engineering

| ENVIRONMENTAL SENSORS       |            |
|-----------------------------|------------|
| Temperature                 | 24.0 °C    |
| Temperature 2               | 24.7 °C    |
| Humidity                    | 16.0 %     |
| Pressure                    | 770.2 mmHg |
| GAS SENSORS                 |            |
| Methane, Butane, LPG, smoke | 33         |
| Natural gas, LPG            | 37         |
| Carbon Monoxide             | 22         |
| LIGHTS                      |            |
| mode                        | manual     |
| status                      | ①          |

## IOT Devices Data



# The City College of New York

## The Grove School of Engineering

## Capturing packets with Wireshark

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

mqtt

| No.     | Time         | Source         | Destination    | Protocol | Length | Info                         |
|---------|--------------|----------------|----------------|----------|--------|------------------------------|
| 1209... | 12772.775740 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1209... | 12777.790639 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1209... | 12782.804092 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1209... | 12787.804833 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1209... | 12792.823755 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1210... | 12797.826449 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1210... | 12802.822598 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1210... | 12807.827985 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1210... | 12812.890564 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1210... | 12817.859147 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |
| 1210... | 12817.860004 | 192.168.0.100  | 137.135.83.217 | MQTT     | 56     | Ping Request                 |
| 1210... | 12817.871556 | 137.135.83.217 | 192.168.0.100  | MQTT     | 56     | Ping Response                |
| 1211... | 12822.852857 | 137.135.83.217 | 192.168.0.100  | MQTT     | 163    | Publish Message [luch/sens1] |

Header Flags: 0x30, Message Type: Publish Message, QoS Level: At most once delivery (Fire and Forget)

0011 .... = Message Type: Publish Message (3)  
.... 0... = DUP Flag: Not set  
.... .0.. = QoS Level: At most once delivery (Fire and Forget) (0)  
.... ...0 = Retain: Not set

Msg Len: 107  
Topic Length: 10

| Hex  | Dec                     | Text                    |                      |
|------|-------------------------|-------------------------|----------------------|
| 0000 | c8 f7 33 5a 9a ce 50 c7 | bf 1d d0 c4 08 00 45 00 | ..3Z..P.....E.       |
| 0010 | 00 95 fd 94 40 00 36 06 | a8 61 89 87 53 d9 c0 a8 | ....@.6..a..S...     |
| 0020 | 00 64 07 5b ce 24 0f f7 | 4f 10 e7 3d ca 1c 50 18 | .d.[ \$. .. 0..=..P. |
| 0030 | 01 f6 43 8d 00 00 30 6b | 00 0a 6c 75 63 68 2f 73 | ..C...Ok ..luch/s    |
| 0040 | 65 6e 73 31 7b 22 74 79 | 70 65 22 3a 22 72 65 73 | ens1{"ty pe":"res    |
| 0050 | 70 6f 6e 73 65 22 2c 22 | 74 65 6d 70 65 72 61 74 | ponse"," temperat    |
| 0060 | 75 72 65 5f 31 22 3a 32 | 37 2c 22 68 75 6d 69 64 | ure_1":2 7,"humid    |
| 0070 | 69 74 79 5f 31 22 3a 31 | 37 2c 22 74 65 6d 70 65 | ity_1":1 7,"tempe    |
| 0080 | 72 61 74 75 72 65 5f 32 | 22 3a 32 37 2e 36 2c 22 | rature_2 ":"27.6,"   |
| 0090 | 70 72 65 73 73 75 72 65 | 5f 32 22 3a 31 30 32 33 | pressure _2":1023    |
| 00a0 | 39 32 7d                |                         | 92}                  |

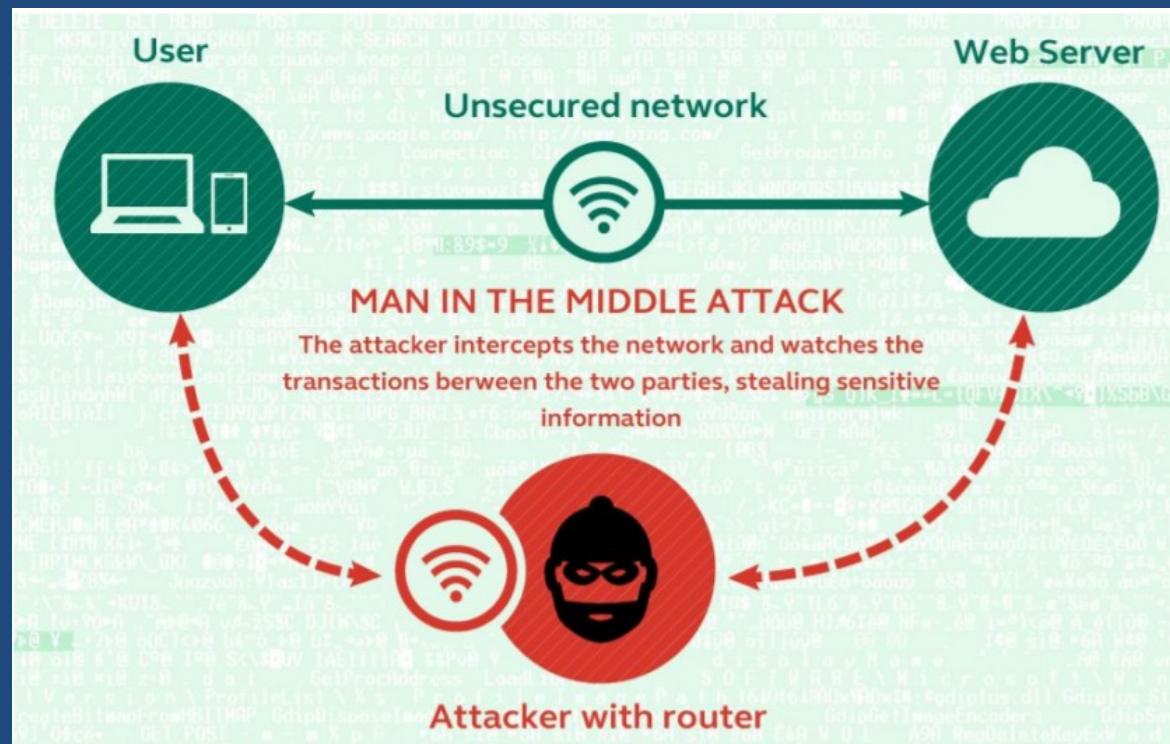
Msg Len (mqtt.len), 1 byte

Packets: 1211251 · Displayed: 2614 (0.2%)

Profile: Default

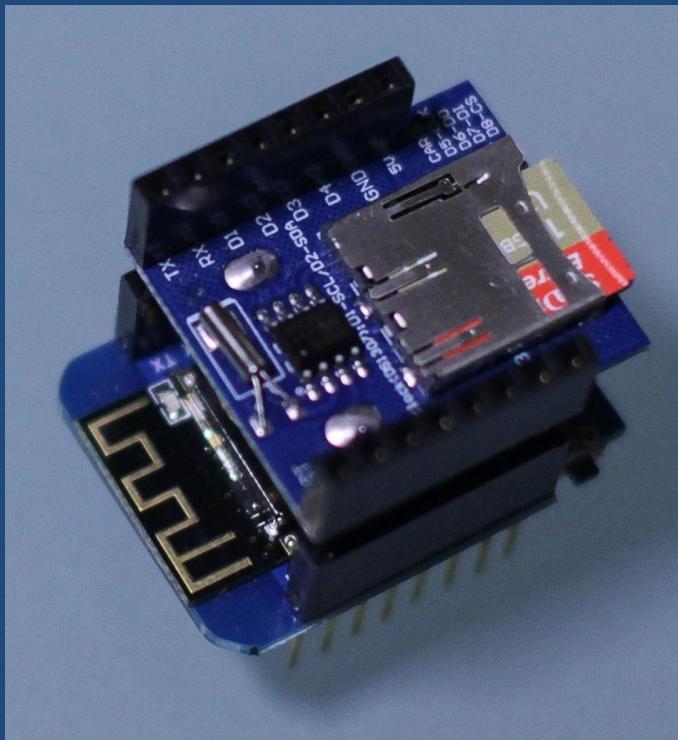


## Man in the Middle Attack





# Wi-Fi Data Packets Sniffing Using an ESP8266



## ArduinoPcap

```
/dev/cu.wchusbserial14530
Send

?1??J?9)ç弊sll??|?1?| ? d? c|?? ? ??b? #??oo?lg'?? c x??lrlslp?g? ? l?? c g?| d? ?c?nn?l??l
? oo l` o{??g c ? l r??n c ? d? d?? l?d` ??g? ?
starting...
initialization done.
opened: example_28.pcap
Sniffer started!
```

Autoscroll      No line ending      115200 baud      Clear output



# Config.js

```
/* Magic Mirror Config Sample
var config = {
var config = {
    address: "localhost", // Address to listen on, can be:
        // - "localhost", "127.0.0.1", "::1" to listen on loopback interface
        // - another specific IPv4/6 to listen on a specific interface
        // - "0.0.0.0", ":" to listen on any interface
        // Default, when address config is left out or empty, is "localhost"
    port: 8080,
    basePath: "/", // The URL path where MagicMirror is hosted. If you are using a
                    Reverse proxy
        // you must set the sub path here. basePath must end with a /
    ipWhitelist: ["127.0.0.1", "::ffff:127.0.0.1", "::1"], // Set [] to allow all IP addresses
        // or add a specific IPv4 of 192.168.1.5 :
        // ["127.0.0.1", "::ffff:127.0.0.1", "::1", "::ffff:192.168.1.5"],
        // or IPv4 range of 192.168.3.0 --> 192.168.3.15 use CIDR format :
        // ["127.0.0.1", "::ffff:127.0.0.1", "::1", "::ffff:192.168.3.0/28"],
```



# Config.js

```
useHttps: false, // Support HTTPS or not, default "false" will use HTTP
httpsPrivateKey: "", // HTTPS private key path, only require when useHttps is true
httpsCertificate: "", // HTTPS Certificate path, only require when useHttps is true

language: "en",
logLevel: ["INFO", "LOG", "WARN", "ERROR"],
timeFormat: 12,
units: "imperial",
// serverOnly: true/false/"local" ,
// local for armv6l processors, default
// starts serveronly and then starts chrome browser
// false, default for all NON-armv6l devices
// true, force serveronly mode, because you want to.. no UI on this device
```



# Config.js

```
modules: [
  {
    module: "clock",
    position: "top_left"
  },
  {
    module: "calendar",
    header: "US Holidays",
    position: "top_left",
    config: {
      calendars: [
        {
          symbol: "calendar-check",
          url:"webcal://www.calendarlabs.com/ical-calendar/ics/76/US_Holidays.ics"
        }
      ],
      }
    },
  ]
```



# Config.js

```
{  
  module: "currentweather",  
  position: "top_right",  
  config: {  
    location: "New York",  
    locationID: "", //ID from http://bulk.openweathermap.org/sample/city.list.json.gz; unzip the gz file and find your city  
    appid: "de784e5c0ffe56fdfe7460f32b0569f3"  
  }  
,  
{  
  module: "weatherforecast",  
  position: "top_right",  
  header: "Weather Forecast",  
  config: {  
    location: "New York",  
    locationID: "5128581", //ID from http://bulk.openweathermap.org/sample/city.list.json.gz ; unzip the gz file and find your city  
    appid: "YOUR_OPENWEATHER_API_KEY"  
  }  
,  
  _____
```



# Config.js

```
{  
  module: "MMM-NowPlayingOnSpotify",  
  position: "top_left",  
  config: {  
    clientID: "f61677a0a17c4390b18d159c73e6a678",  
    clientSecret: "6ac97812dbc14d93b305997ca045ad9c",  
    accessToken: "BQCrjUjeTYmrbfCFVOzwGAq6Bsaa3hVSlpkYc2Vfl5bsZI5  
      tB47CISY0vfV26osuGwsRdhLpyduL3skZxCjIWvFkF5_-KTyaNeyde  
      IW9X6-iOEjt4EoqZhCE6kQApC4U04522y7dpH1hAKe-nHAig",  
    refreshToken: "AQC8WS3a0x4rdEt72ojN6a09i0vES_AAI5PV7OPd45c-  
      Mv1hJ3Q1kU0yjrcy8eaS2EDzbKtaQsyefTHI7DICpwJpq34  
      U95RXhoZ8ncjthT5ZJAPCriFfoaBtBCEY_9G0h5M"  
  }  
},  
]  
};
```



The City College of New York  
*The Grove School of Engineering*

## What SMAI should be able to execute:

| Feature Name      | Watch YouTube Videos  |
|-------------------|---|
| Description       | Allow user to screencast YouTube videos from mobile phone or laptop.  |
| User Problem      | Inability to watch YouTube videos anywhere without a laptop or mobile phone frustrates the user.                  |
| Value Proposition | At the user's convenience, YouTube can be screencasted on Smart Mirror.   |
| Assumptions       | 1. The user wants to have access to YouTube videos without having to take mobile phone or laptop to the restroom. |
| Out of scope      | Starting YouTube directly from Smart Mirror.  |
| MVP               | Screencast YouTube videos from mobile phone or laptop.  |

| Feature Name      | Current Weather & Forecast  |
|-------------------|---|
| Description       | A widget to help user (Nosa) see the current weather and forecast once he jumps off the bed in the morning. |
| User Problem      | Inability to see the current weather in the morning without reaching out to a smart phone is frustrating.   |
| Value Proposition | To provide access to weather information anywhere.  |
| Assumptions       | 1. User (Nosa) wants to see the current weather & forecast in a glance.                                     |
| Out of scope      | Information such as air quality, sunrise and sunset, humidity, and wind.                                    |
| MVP               | Weather widget with current weather and forecast.   |

# Smart Mirror AI



## What SMAI should be able to execute:

| Feature Name      | Calendar, Date & Time   |
|-------------------|---|
| Description       | A widget to help user see date, time, and the calendar (can be updated with Alexa).               |
| User Problem      | Adding or removing tasks from calendar should not be restricted to touching the technology screen |
| Value Proposition | Provides the convenience of accessing and updating the calendar anywhere.                         |
| Assumptions       | 1. User wants to update the calendar anywhere.  |
| Out of scope      |   |
| MVP               | A customizable widget for date, time, and calendar, controlled with Amazon Alexa.                 |

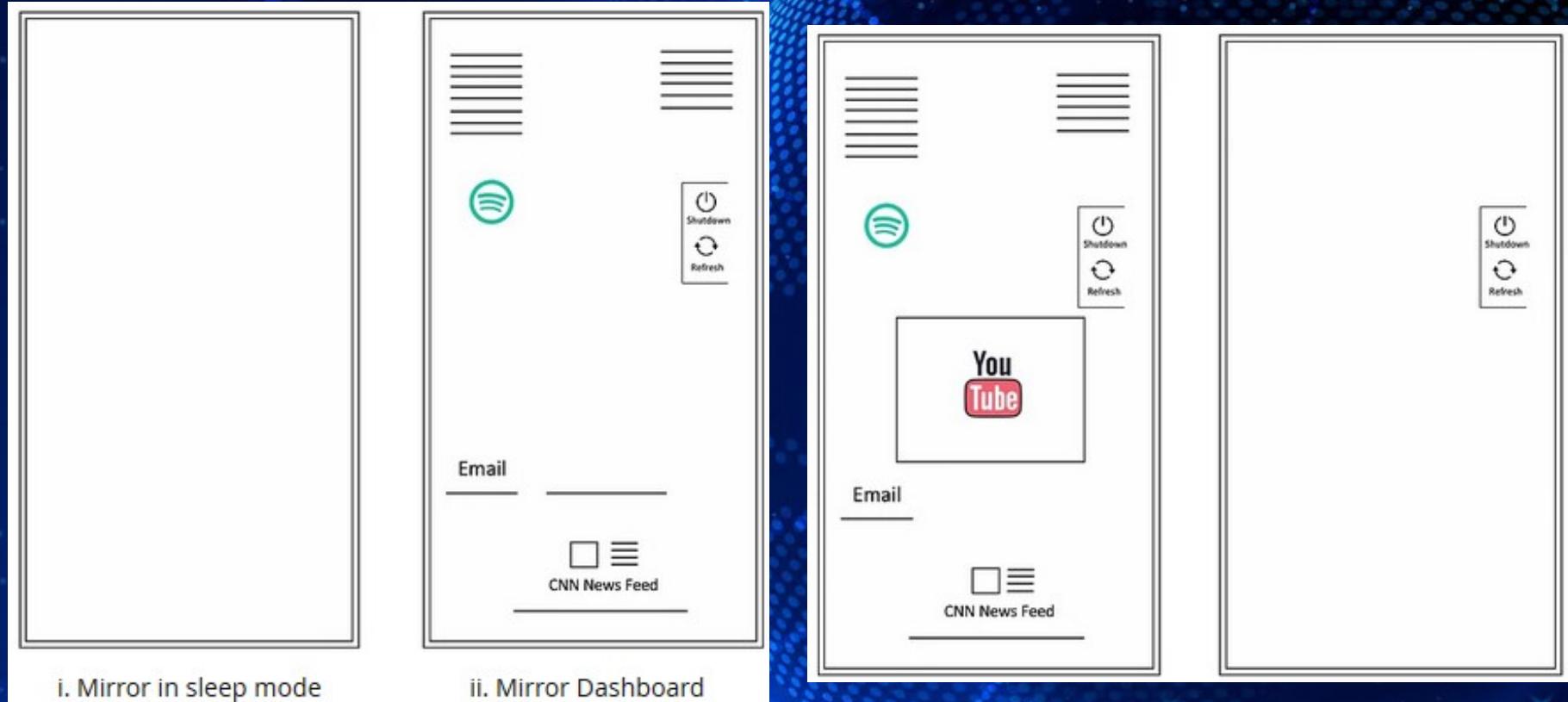


# Smart Mirror AI



The City College of New York  
*The Grove School of Engineering*

## SMAI User Flow



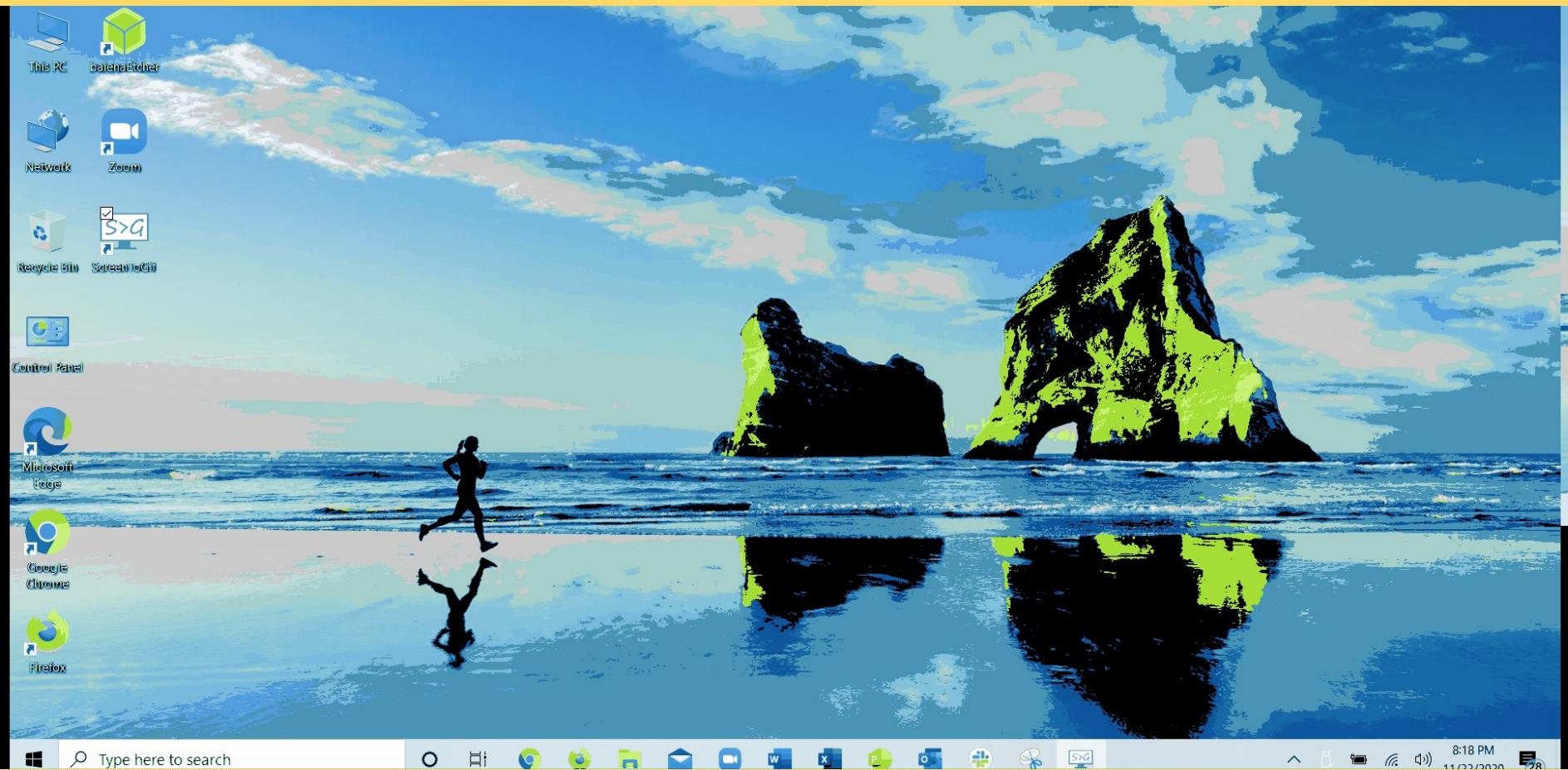
To not have to switch between your PC and raspberry pi 4- we were utilizing a VNC viewer to connect as outlined below

# Smart Mirror AI



The City College of New York  
*The Grove School of Engineering*

# Smart Mirror AI



**SMAI by Anna Taylor and Andrii Aluchko**



The City College of New York  
*The Grove School of Engineering*

# Presentation

## Smart Mirror AI