

Build Your Own Love Messengers

**Today,
3:00PM - 5:00PM**

Connect to Wifi:

Username: NYCR24

Password: clubmate



To do

Set up space:

- Put soldering stations
 - Change soldering tips
 - Wet sponges
 - Extension wires
- Put pliers
- Prep materials in the front

Projector:

- Connect Laptop
- Open all tabs (Firebase, Github, Slides)
- Prep Notes on Ipad

Electronics:

- Test demo (with new wifi already uploaded)

Extra:

- Download Arduino on the laptops
- Find adapters at the space (USBC-USB)
- Take documentation pics

To bring:

Julia:

- Camera, SD card, battery
- Ipad
- Soldering iron + tips
- Luggage of material
- Laptop + Charger + Adapto

Pepi:

- 1x third hand
- Soldering iron + tips
- Laptops, charger +
usbc-usba adaptors

Links

[Notes Julia](#)

[Slides DB](#)

[Slides Code Upload](#)

❤️ Introductions ❤️

- Name
- Pronouns
- What was your last biggest fuck-up?







Before we begin, we need to Download:

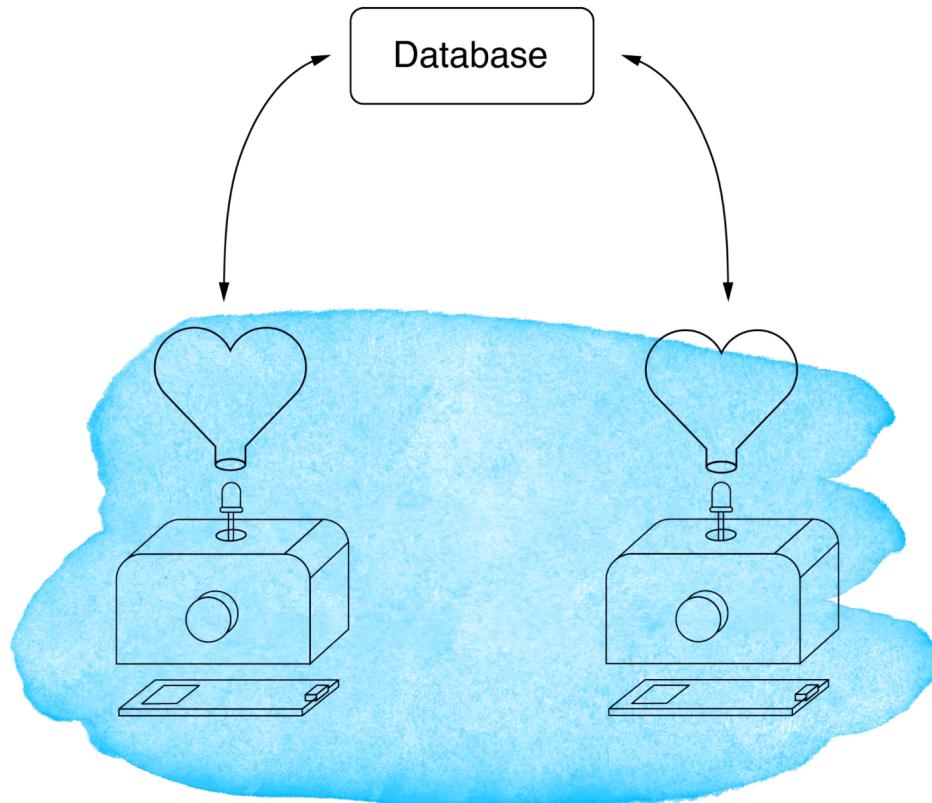
Arduino IDE
Board Packages



Download Arduino IDE:

www.arduino.cc/en/software

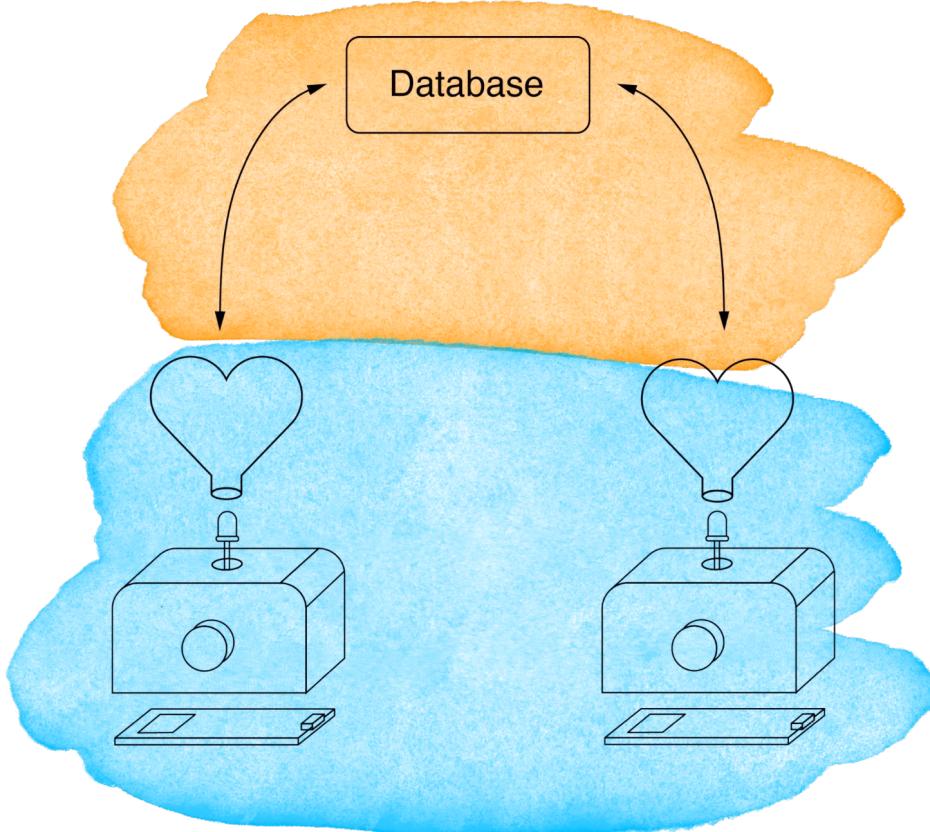
Schedule:



Physical Segment (1hr)

- Soldering
- Assembling

Schedule:

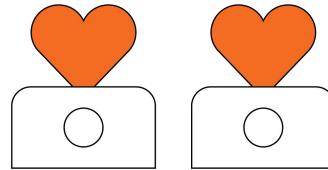


Digital Segment (1hr)

- Database
- Code

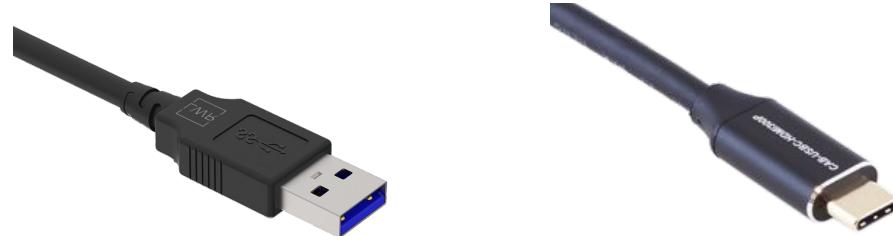
Physical Segment (1hr)

- Soldering
- Assembling



We will now give out the base material kit!

Choose between: **USB-A** or **USB-C**

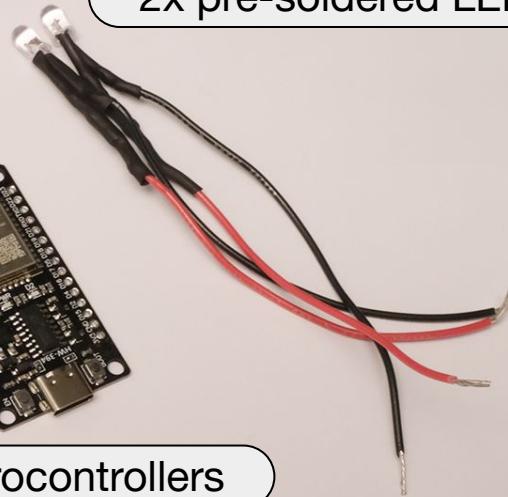


Materials provided in your base materials kit

2x Cables to upload code



2x pre-soldered LEDs



2x ESP32 microcontrollers



4x screws

1x Wormi Collective Sticker <3

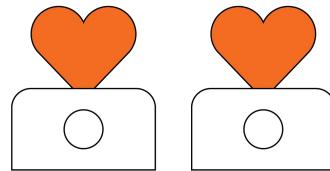


Items to customize!



1x box floor
labelled “2”





Come to the front to customize your own Love Messengers!

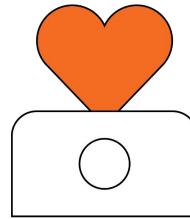
Settle at the front soldering station



Who here has soldered before?



Live demo of soldering 

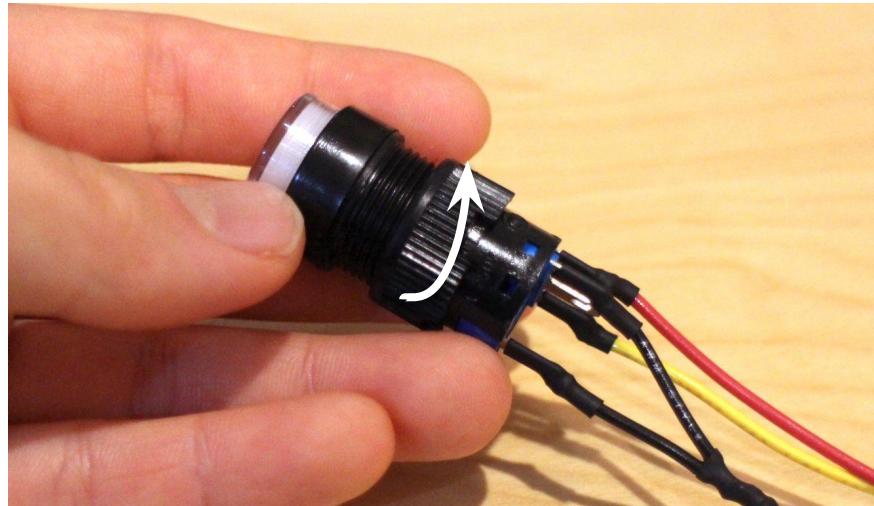


Let's start with just one love messenger!

Part 1: Pre-assembly

STEP 1

- Unscrew the place piece from the back of the button

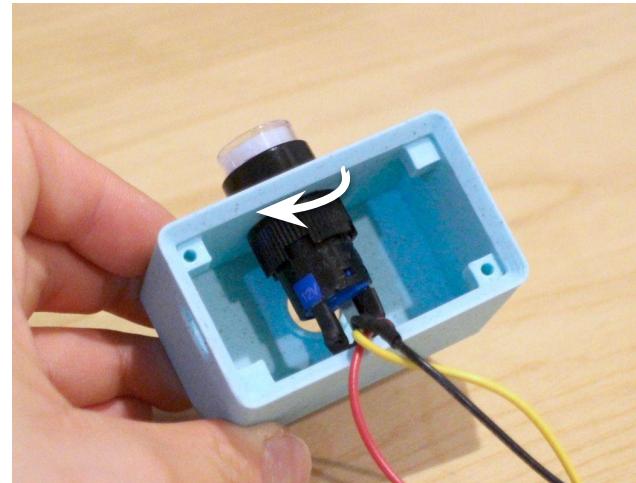
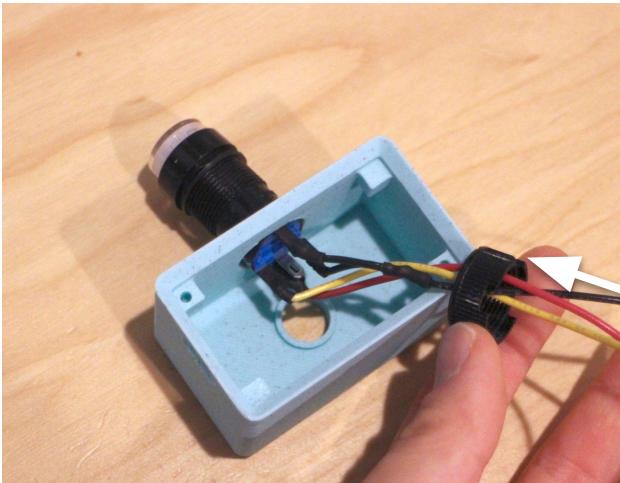
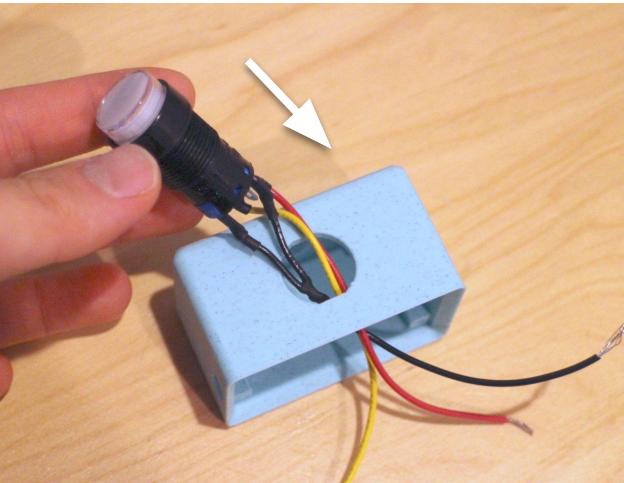


Part 1: Pre-assembly

STEP 2

Insert button

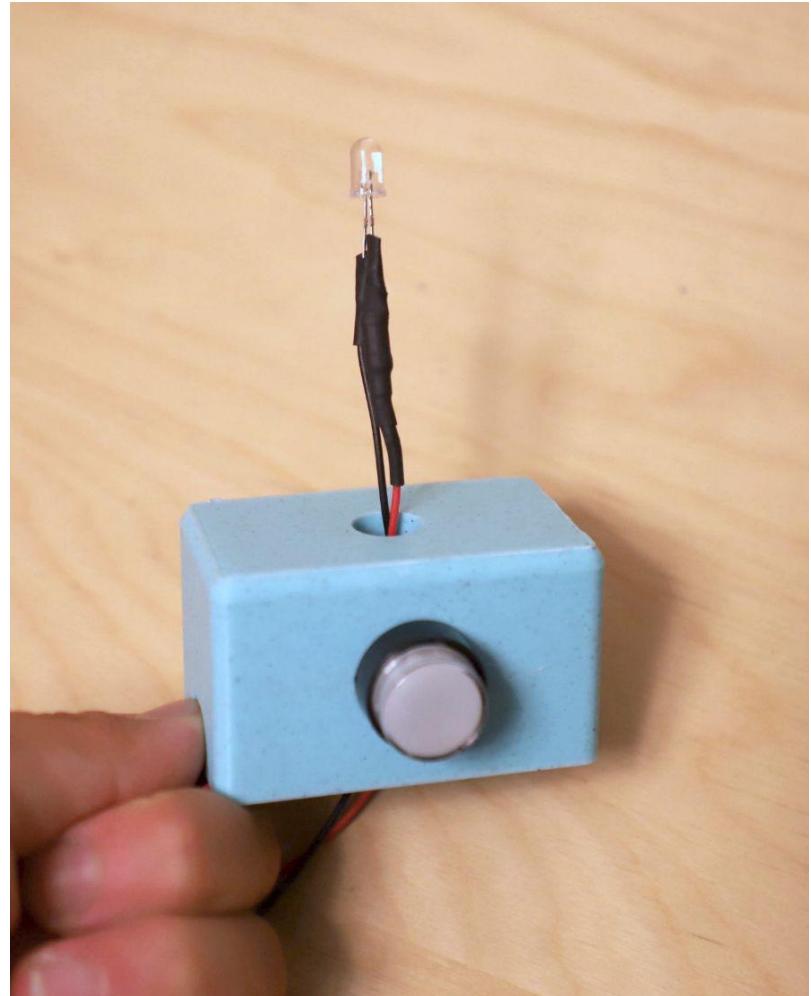
Screw button into the encasing



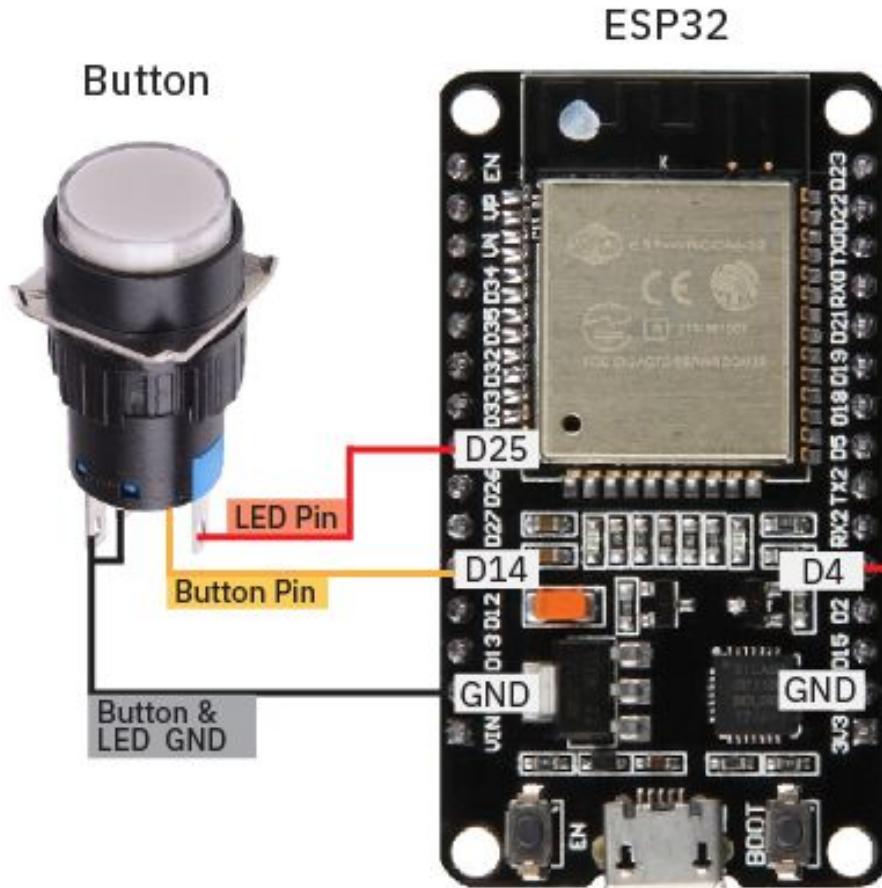
Part 1: Pre-assembly

STEP 3

Stick the pre-soldered LED through the hole at the top of the box encasing.



Part 2: Soldering



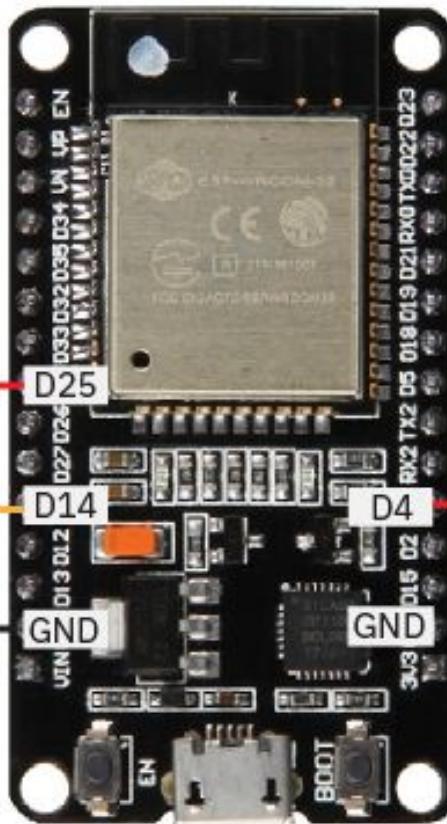
Pre-soldered LED

ESP32

Button



Button &
LED GND

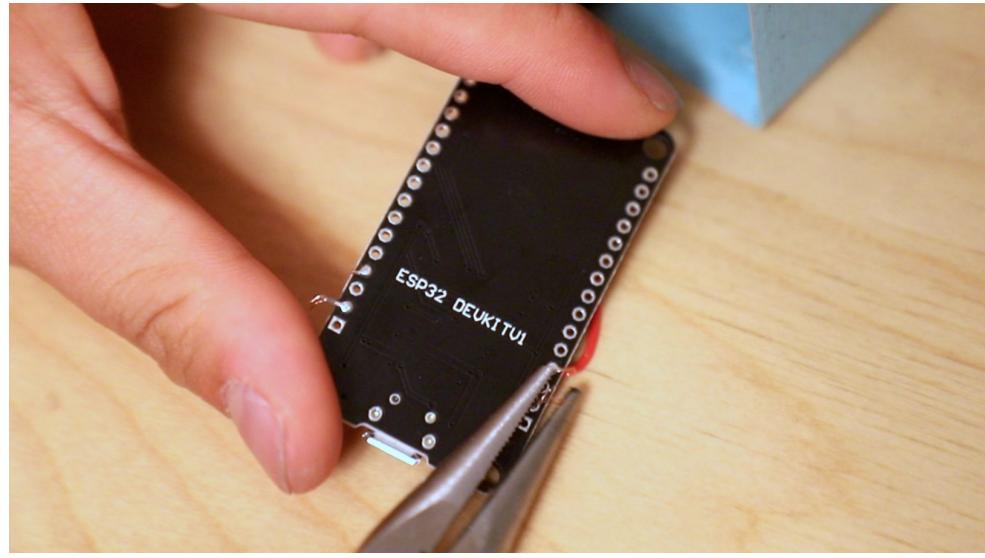
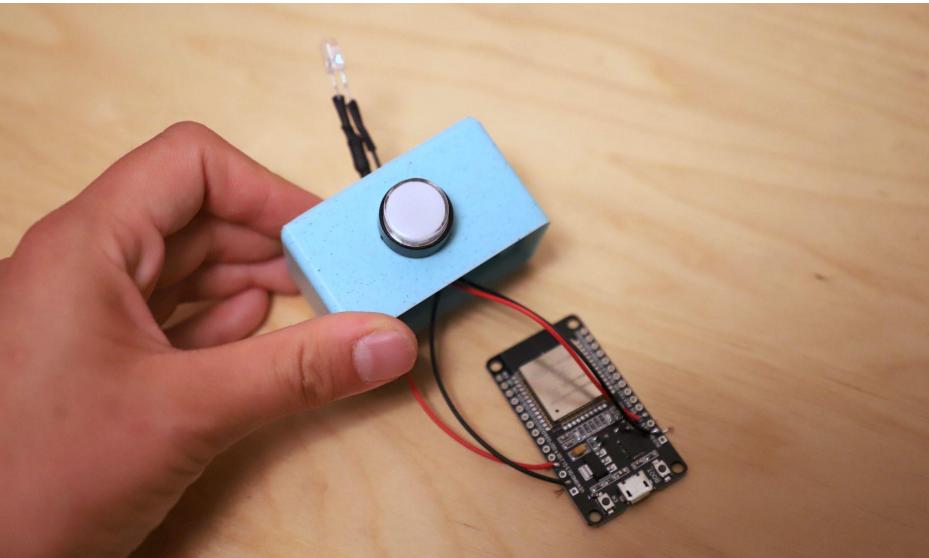


220 Ω
Resistor

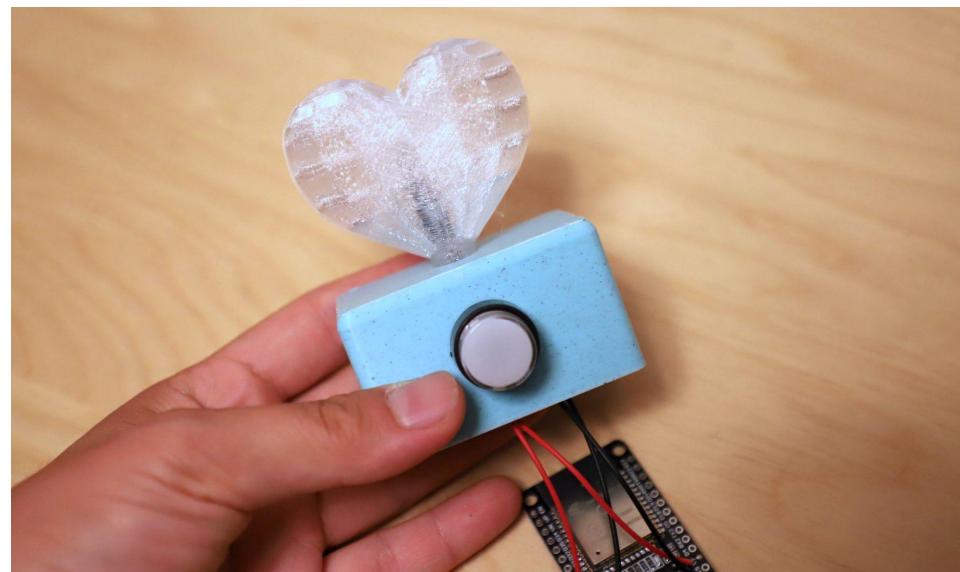
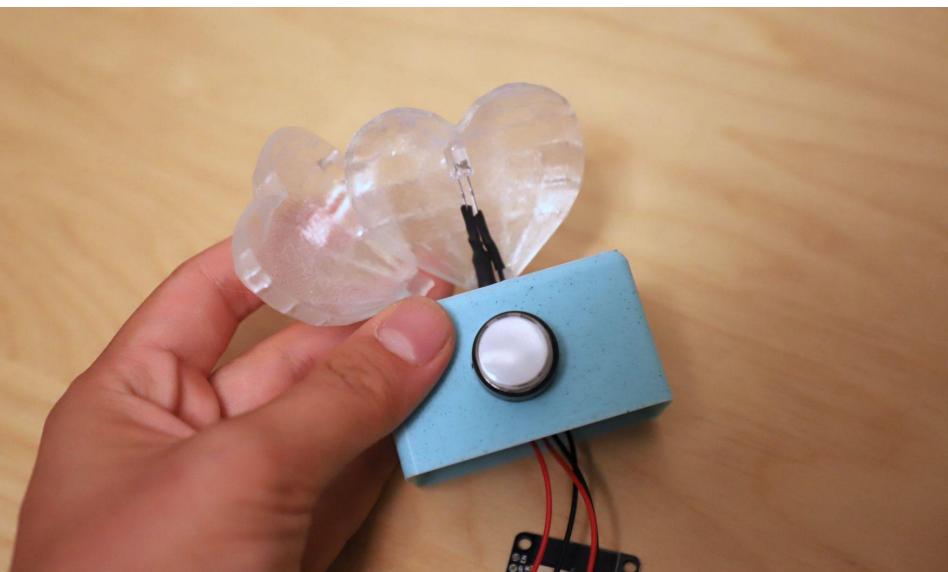


Part 2: Soldering

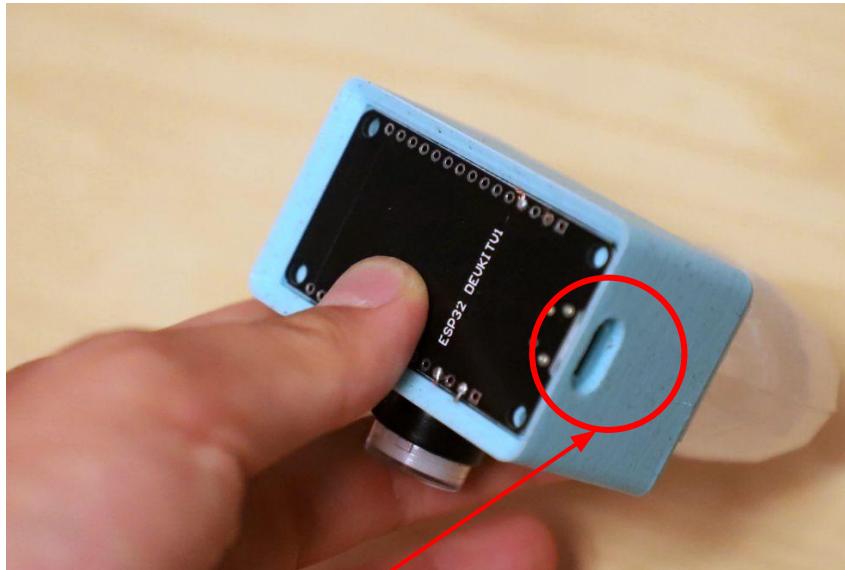
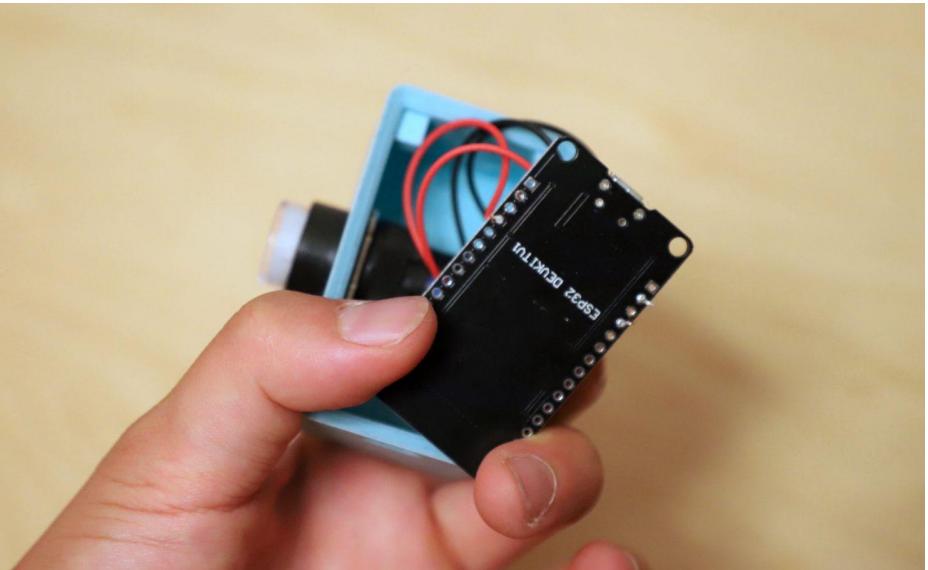
Use pliers to cut off any excess wires



Finally: Putting everything together! 😊

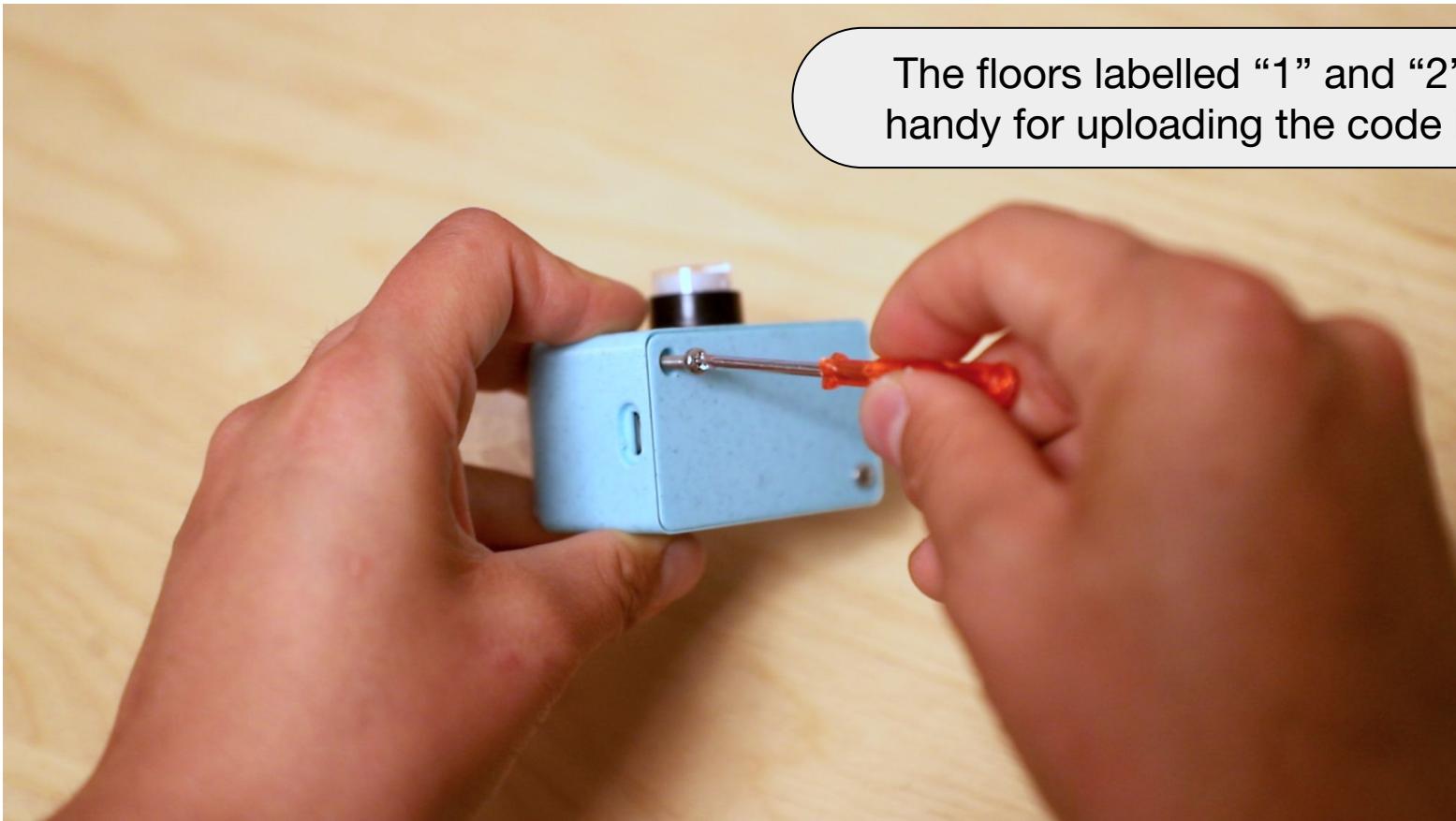


Finally: Putting everything together! 😊



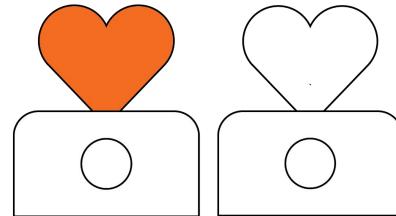
Output of the microcontroller
faces the hole on the encasing.

Finally: Putting everything together! 😊



The floors labelled “1” and “2” are handy for uploading the code later!





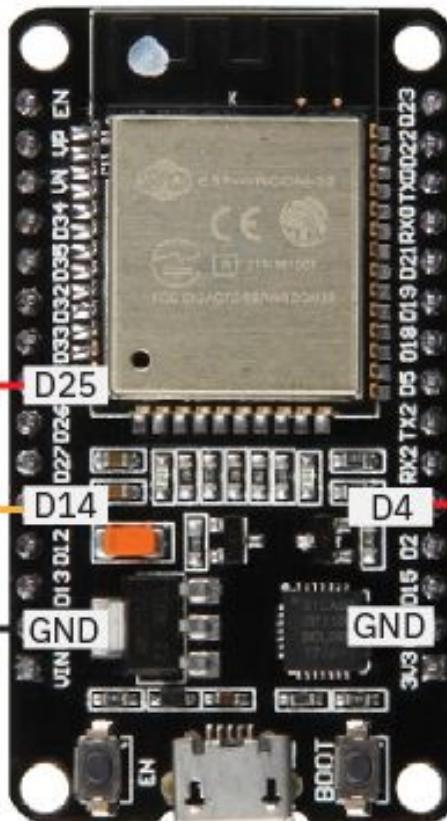
You're all done with one love messenger!

Let's now make the second one!

Pre-soldered LED

ESP32

Button

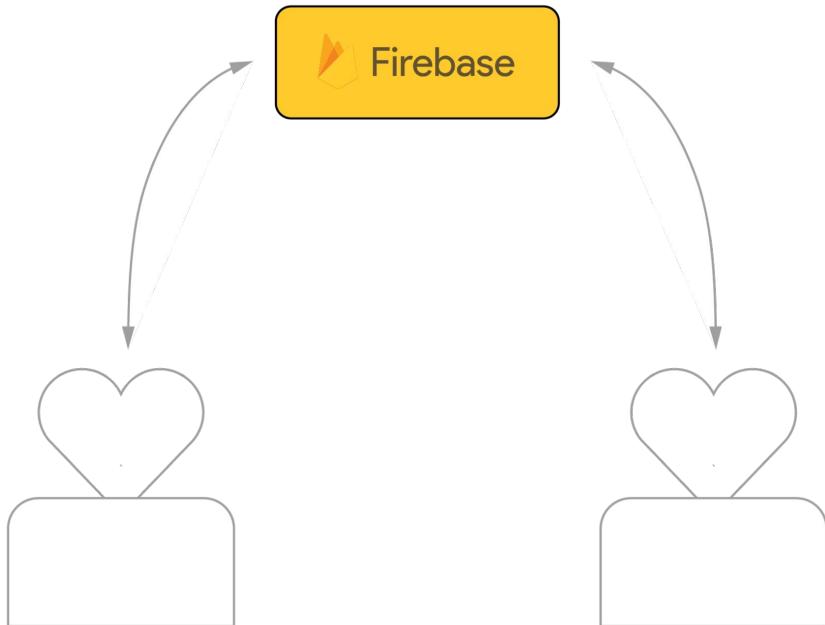


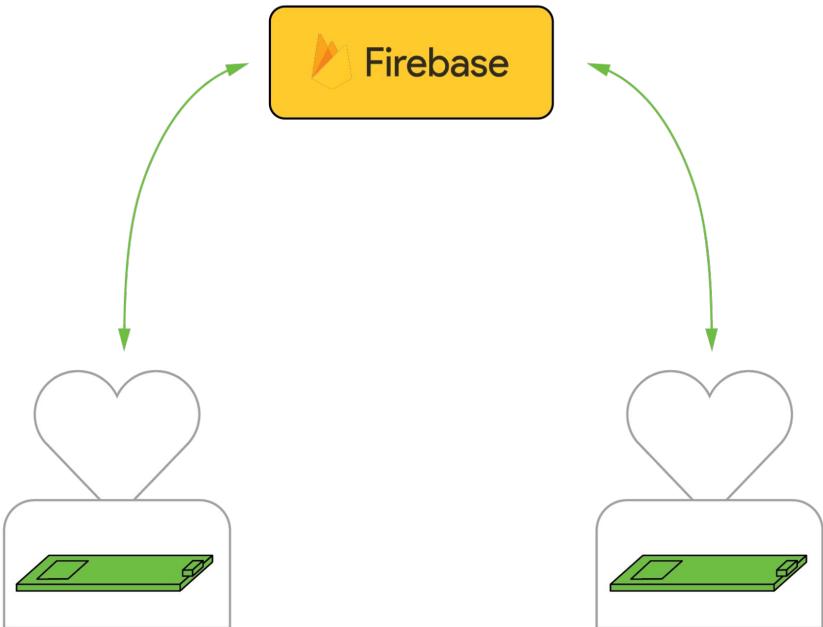
220 Ω
Resistor



Digital Segment

Step 1: Set up Database





Step 1: Set up Database

Step 2: Upload Code

Instructions: shorturl.at/HUtTH

We will regroup when everyone is done!



Who has written Arduino code before?



Pseudo Code

Setup-code:

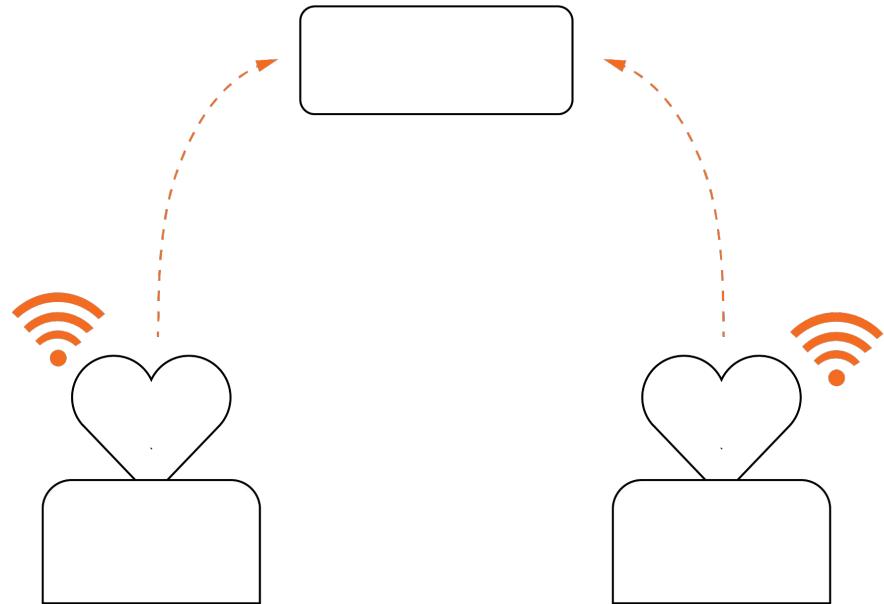
Main Code (runs in a loop):

Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):



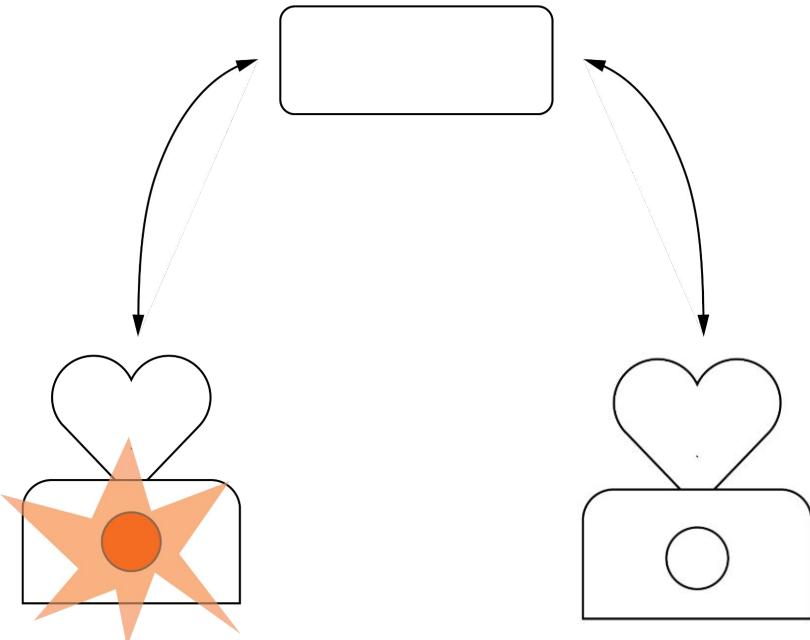
Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**



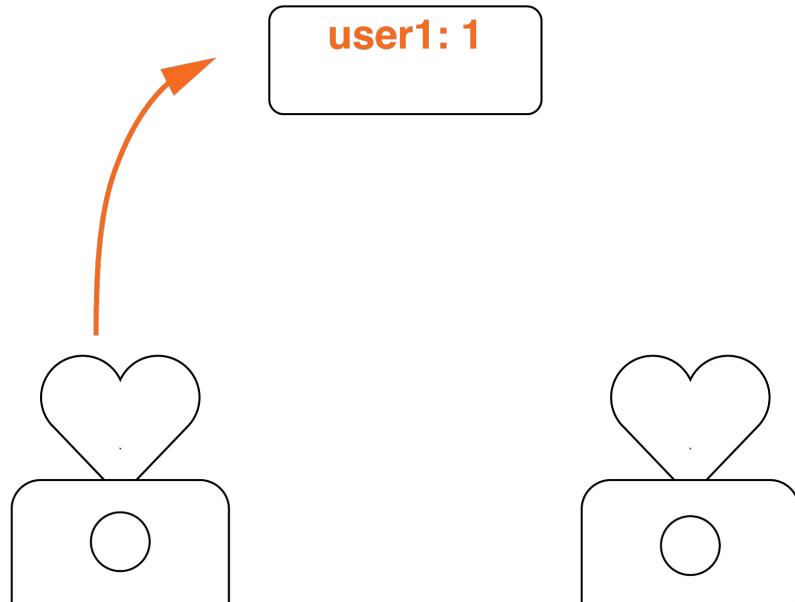
Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase



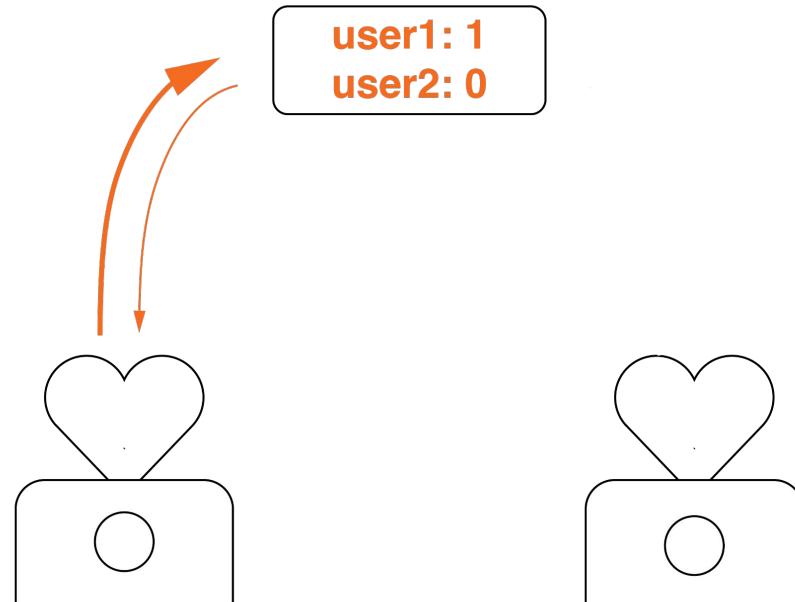
Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase



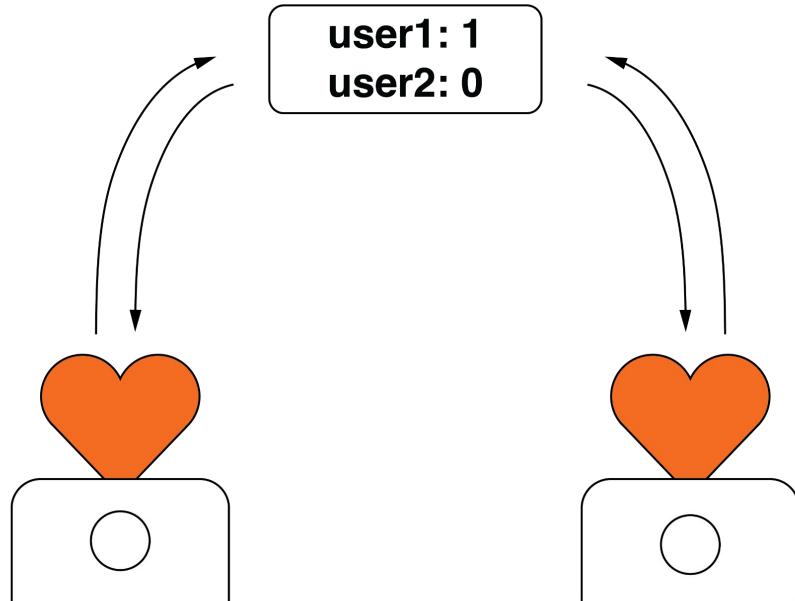
Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)



Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**



Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)

Arduino Code (C++)

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK && (millis() -  
lastFirebaseUpdate > 2000)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState = digitalRead(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        manageLED(buttonState, firebaseData);  
  
    }  
    delay(5);  
}
```

Pseudo Code

Arduino Code (C++)

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}
```

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)

```
void loop() {  
    if (Firebase.ready() && signupOK && (millis() -  
lastFirebaseUpdate > 2000)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState = digitalRead(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        manageLED(buttonState, firebaseData);  
  
    }  
    delay(5);  
}
```

Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)

Arduino Code (C++)

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK && (millis() -  
lastFirebaseUpdate > 2000)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState = digitalRead(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        manageLED(buttonState, firebaseData);  
  
    }  
    delay(5);  
}
```

Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)

Arduino Code (C++)

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK && (millis() -  
lastFirebaseUpdate > 1000)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState = digitalRead(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        manageLED(buttonState, firebaseData);  
  
    }  
    delay(5);  
}
```

Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)

Arduino Code (C++)

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK && (millis() -  
lastFirebaseUpdate > 2000)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState = digitalRead(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        manageLED(buttonState, firebaseData);  
  
    }  
    delay(5);  
}
```

Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)

Arduino Code (C++)

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK && (millis() -  
lastFirebaseUpdate > 2000)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState = digitalRead(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        manageLED(buttonState, firebaseData);  
  
    }  
    delay(5);  
}
```

Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)

Arduino Code (C++)

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK && (millis() -  
lastFirebaseUpdate > 2000)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState = digitalRead(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        manageLED(buttonState, firebaseData);  
  
    }  
    delay(5);  
}
```

Pseudo Code

Setup-code:

1. Connecting to **Wifi**
2. Connecting to **Firebase**

Main Code (runs in a loop):

1. Reading the **Button State**
2. **Uploading** Data to Firebase
3. **Downloading** Data from Firebase
4. Manage **LED** (turn on/off)

Arduino Code (C++)

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK && (millis() -  
lastFirebaseUpdate > 2000)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState = digitalRead(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        manageLED(buttonState, firebaseData);  
    }  
    delay(5);  
}
```

Let's Upload the Code!



Access Code from Github:

shorturl.at/fV1KB

Some last bits

```
#include "addons	TokenNameHelper.h"
#include "addons/RTDBHelper.h"

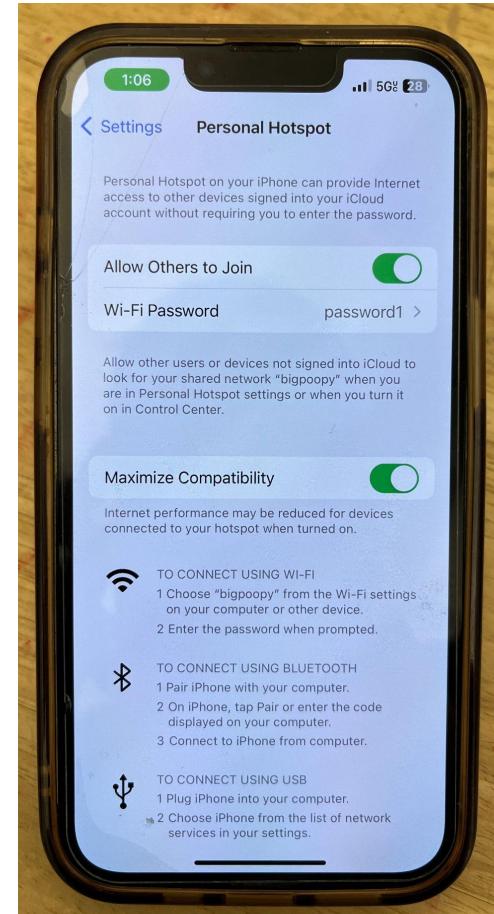
#define WIFI_SSID "Insert SSID"
#define WIFI_PASSWORD "Insert Wifi Password"

#define API_KEY "Insert API Key"
#define DATABASE_URL "Insert Database URL"

const int ledPin = 2;
const int buttonPin = 12;

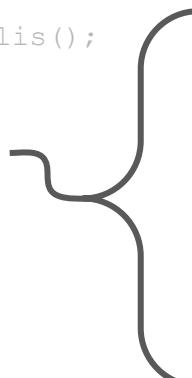
bool firebaseData = false;
int buttonState = 1;

FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig config;
unsigned long lastFirebaseUpdate = 0;
int count = 0;
bool signupOK = false;
```



Arduino Code

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK &&  
(millis() - lastFirebaseUpdate > 100))  
    {  
        lastFirebaseUpdate = millis();  
  
        buttonState =  
digitalRead(buttonPin);  
uploadData(buttonState);  
downloadData();  
ManageLED(buttonState,  
firebaseData);  
  
    }  
}
```



```
void uploadData(int buttonstate) {  
    if (Firebase.RTDB.setInt(&fbdo, "/user_1",  
buttonstate)) {  
        Serial.printf("Data UPLOAD successful, ");  
    } else {  
        Serial.println("Data UPLOAD failed, ");  
    }  
}
```

Arduino Code

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK &&  
(millis() - lastFirebaseUpdate > 100))  
    {  
        lastFirebaseUpdate = millis();  
  
        buttonState =  
digitalRead(buttonPin);  
uploadData(buttonState);  
downloadData();  
ManageLED(buttonState,  
firebaseData);  
  
    }  
}
```

```
void downloadData() {  
    if (Firebase.RTDB.getInt(&fbdo, "/user_2")) {  
        firebaseData = fbdo.intData();  
        Serial.println("Data DOWNLOAD successful");  
    } else {  
        Serial.println("Data DOWNLOAD failed");  
    }  
}
```

Arduino Code

```
void setup() {  
    Serial.begin(115200);  
  
    pinMode(ledPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
  
    connectWiFi();  
    connectFirebase();  
}  
  
void loop() {  
    if (Firebase.ready() && signupOK &&  
        (millis() - lastFirebaseUpdate > 100)) {  
        lastFirebaseUpdate = millis();  
  
        buttonState =  
        digitalWrite(buttonPin);  
        uploadData(buttonState);  
        downloadData();  
        ManageLED(buttonState,  
        firebaseData);  
    }  
}
```

```
void manageLED(int buttonState, int  
firebaseData) {  
    if (buttonState == LOW or firebaseData ==  
LOW) {  
        digitalWrite(ledPin, HIGH);  
        delay(2000);  
        digitalWrite(ledPin, LOW);  
    } else {  
        digitalWrite(ledPin, LOW);  
    }  
}
```

Time for...
Troubleshooting

Everything is in our GitHub page!

 **NYCResistor_LoveMessengers** Public

 main  1 Branch  0 Tags

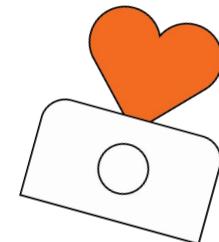
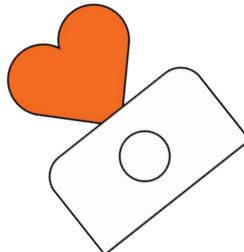
 Go to file 

 Add file  Code

 juliadaser	Update README.md	bcba93d · 1 hour ago	 16 Commits
	3D Models	Add files via upload	4 days ago
	Media	Add files via upload	11 hours ago
	Slides	Add files via upload	1 hour ago
	README.md	Update README.md	1 hour ago
	code.ino	Add files via upload	2 hours ago

PLEASE leave the cables on the table &

Help us fill out this feedback form!



Open to Questions!

 @wormicollective

Access the code:

[https://github.com/pepzicles/
NYCResistor_LoveMessengers.git](https://github.com/pepzicles/NYCResistor_LoveMessengers.git)

Contact us:

yiqing.ng@gmail.com

