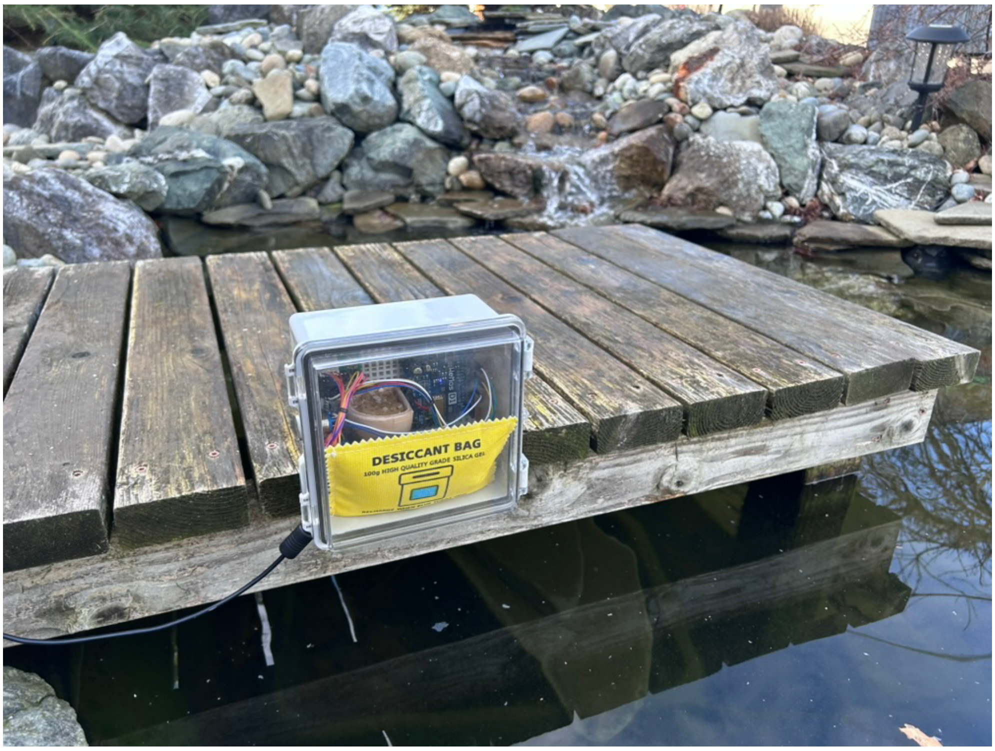
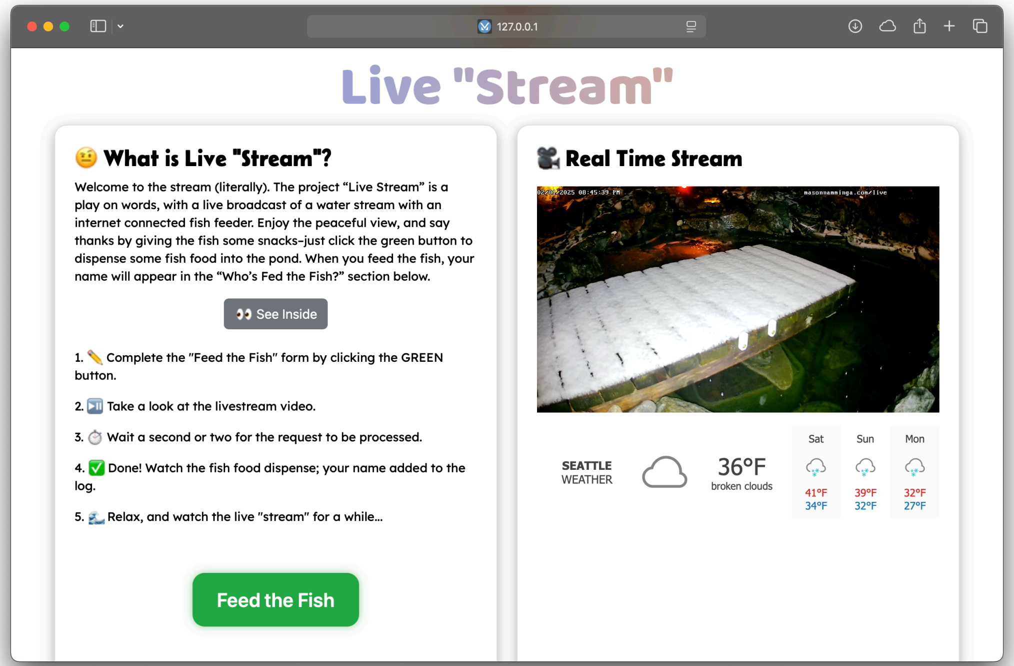
🐟 Smart Fish Feeder | Build Guide





Live "Stream"

🤨 What is Live "Stream"?

Welcome to the stream (literally). The project “Live Stream” is a play on words, with a live broadcast of a water stream with an internet connected fish feeder. Enjoy the peaceful view, and say thanks by giving the fish some snacks–just click the green button to dispense some fish food into the pond. When you feed the fish, your name will appear in the “Who’s Fed the Fish?” section below.

1. ✏️ Complete the "Feed the Fish" form by clicking the GREEN button.

2. ⏯️ Take a look at the livestream video.

3. ⏱️ Wait a second or two for the request to be processed.

4. ✅ Done! Watch the fish food dispense; your name added to the log.

5. 🌊 Relax, and watch the live "stream" for a while...

🎥 Real Time Stream

SEATTLE WEATHER

36°F
broken clouds

Sat	Sun	Mon
41°F 34°F	39°F 32°F	32°F 27°F

Section One: Creating the Livestream Camera

Summary:

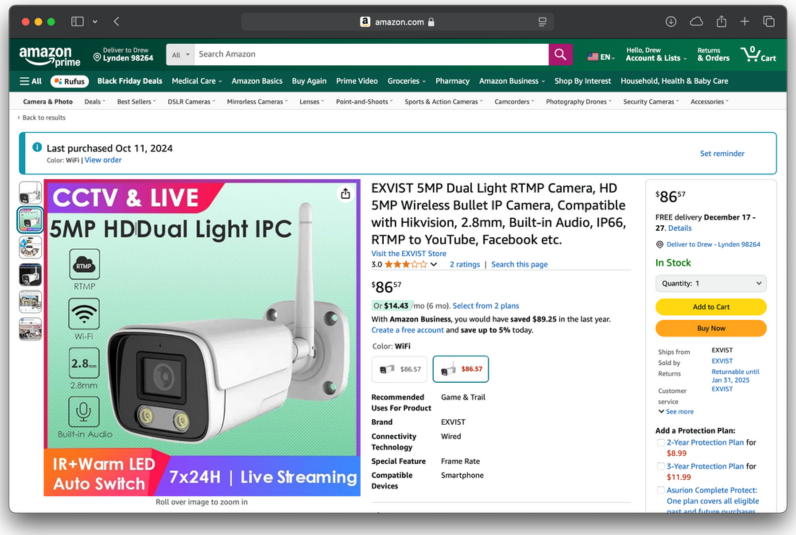
In this section, you will learn how to live stream a security camera to the internet via Twitch.

Parts, Materials, & Software:

An RTMP-compatible camera is required to stream from the camera directly to the internet without a dedicated computer.

💡 Note: *RTMP is different than RTSP*

- Exvist RTMP Camera (https://a.co/d/iKmvdlX)
- Camera Software (https://support.exvist.com/portal/en/kb/network-camera/device-manager/articles/ipc-device-manager#Download_Device_Manager)



Last purchased Oct 11, 2024
Color: WiFi | View order
Set reminder

EXVIST 5MP Dual Light RTMP Camera, HD 5MP Wireless Bullet IP Camera, Compatible with Hikvision, 2.8mm, Built-in Audio, IP66, RTMP to YouTube, Facebook etc.
Visit the EXVIST Store
3.0 ★★★☆☆ 2 ratings | Search this page

$86.57

Or $14.43 /mo (6 mo). Select from 2 plans
With Amazon Business, you would have saved $89.25 in the last year.
Create a free account and save up to 5% today.

Color: WiFi
$86.57 $86.57

Recommended Uses For Product	Game & Trail
Brand	EXVIST
Connectivity Technology	Wired
Special Feature	Frame Rate
Compatible Devices	Smartphone

$86.57
FREE delivery December 17 - 27. Details
Deliver to Drew - Lynden 98264
In Stock
Quantity: 1

Ships from	EXVIST
Sold by	EXVIST
Returns	Returnable until Jan 31, 2025
Customer service	EXVIST

Add a Protection Plan:
□ 2-Year Protection Plan for $8.99
□ 3-Year Protection Plan for $11.99
□ Asurion Complete Protect: One plan covers all eligible past and future purchases

Part One: Connecting the Camera to WiFi:

🚨 Warning: *Disregard the instructions in the box, they don't work. The companion software for this camera is AWFUL; be patient and keep trying. There is NO user documentation for this camera, but it is the most cost-effective streaming camera that could stream to Twitch directly without a computer. (That I could find…)*

Download the IPC Camera manager software (link above)

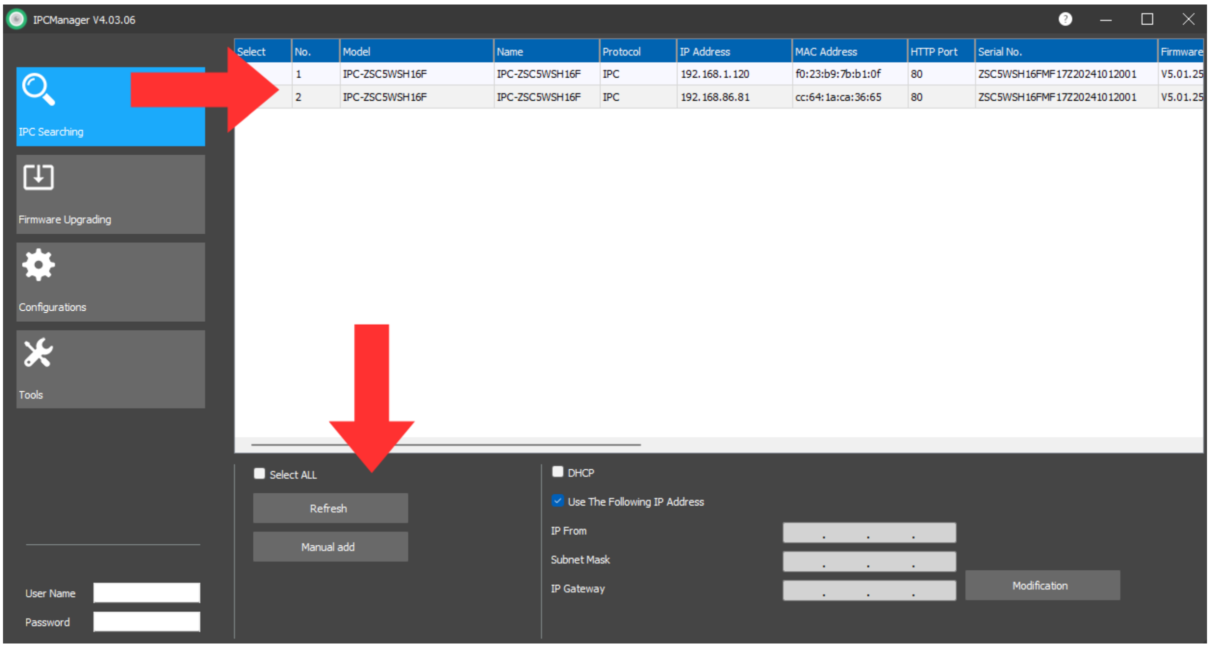
Connect the camera to power

Connect your computer to the camera's WiFi hotspot

- WiFi Name: "HSIPC-XXXXXX"
- Password "12345678"

Once you are connected, open the camera manager software and click refresh, two camera options should appear; try each. (One always says "incorrect login")

- User: admin
- Password: 12345



Select	No.	Model	Name	Protocol	IP Address	MAC Address	HTTP Port	Serial No.	Firmware
	1	IPC-ZSC5WSH16F	IPC-ZSC5WSH16F	IPC	192.168.1.120	f0:23:b9:7b:b1:0f	80	ZSC5WSH16FMF17Z20241012001	V5.01.25
	2	IPC-ZSC5WSH16F	IPC-ZSC5WSH16F	IPC	192.168.86.81	cc:64:1a:ca:36:65	80	ZSC5WSH16FMF17Z20241012001	V5.01.25

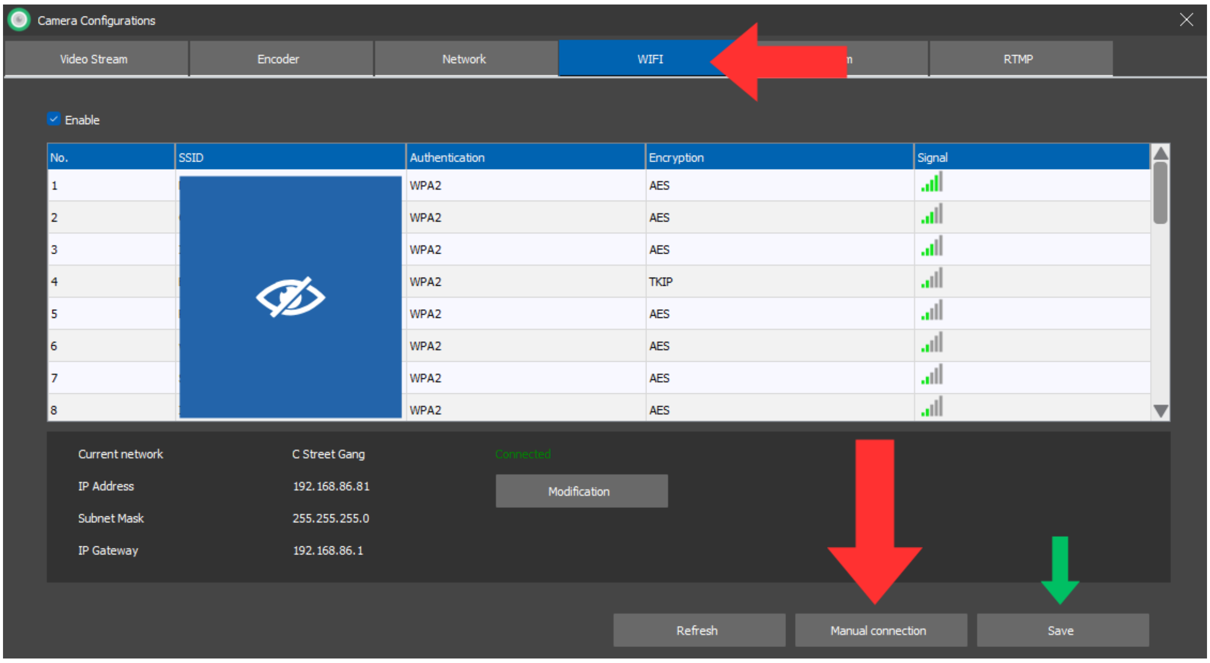
☐ Select ALL

☐ DHCP

☑ Use The Following IP Address

Now, once you have logged in, connect the camera to your home internet so you can connect without the camera's built-in hotspot.

💡 Note: *Use the manual connection method, enter your network name and password, and submit. There is no "connection successful message."* ***CLICK SAVE AFTER EVERY CHANGE***



☑ Enable

No.	SSID	Authentication	Encryption	Signal
1	[redacted]	WPA2	AES	📶
2	[redacted]	WPA2	AES	📶
3	[redacted]	WPA2	AES	📶
4	[redacted]	WPA2	TKIP	📶
5	[redacted]	WPA2	AES	📶
6	[redacted]	WPA2	AES	📶
7	[redacted]	WPA2	AES	📶
8	[redacted]	WPA2	AES	📶

Current network C Street Gang Connected

IP Address 192.168.86.81

Subnet Mask 255.255.255.0

IP Gateway 192.168.86.1

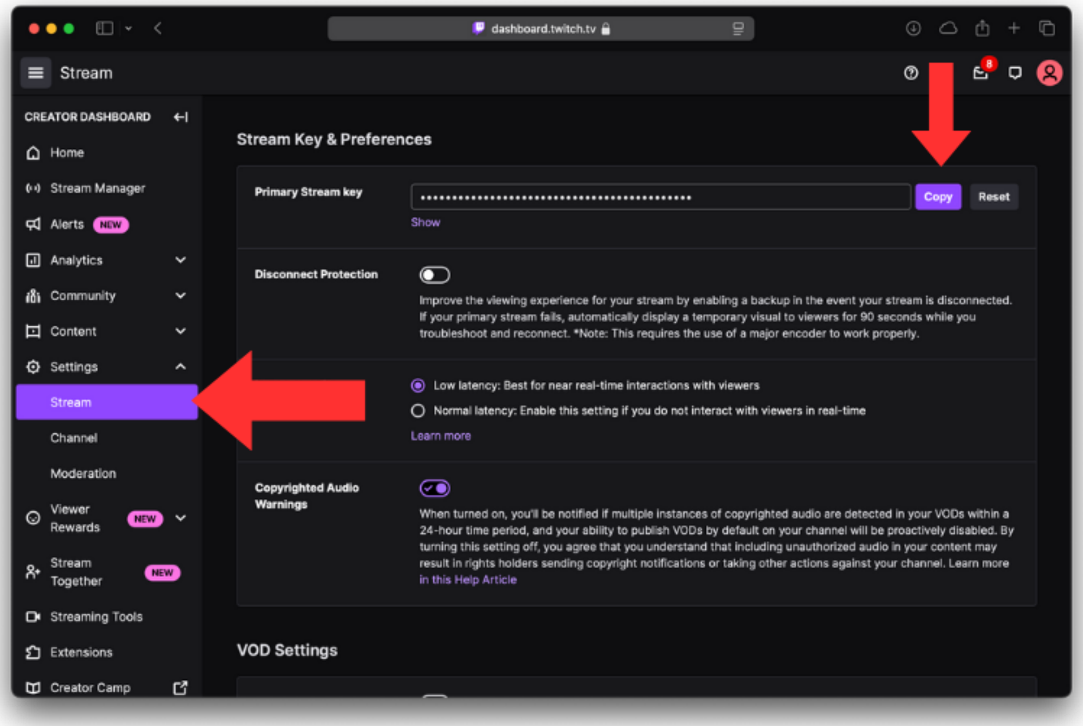
⏯ Pause: Verify your camera is connected to the internet by disconnecting from the hotspot, connecting to your home wifi, and connecting to the camera.

Part Two: Creating your Twitch Channel

Next, you need a Twitch streaming URL that you can have the camera send the video feed to. Create a Twitch account if you don't have one already.

Create a Twitch account and navigate to: https://dashboard.twitch.tv.

Click on settings, then stream, and copy your stream key.



Stream Key & Preferences

Primary Stream key ••

Disconnect Protection

Improve the viewing experience for your stream by enabling a backup in the event your stream is disconnected. If your primary stream fails, automatically display a temporary visual to viewers for 90 seconds while you troubleshoot and reconnect. *Note: This requires the use of a major encoder to work properly.

● Low latency: Best for near real-time interactions with viewers

○ Normal latency: Enable this setting if you do not interact with viewers in real-time

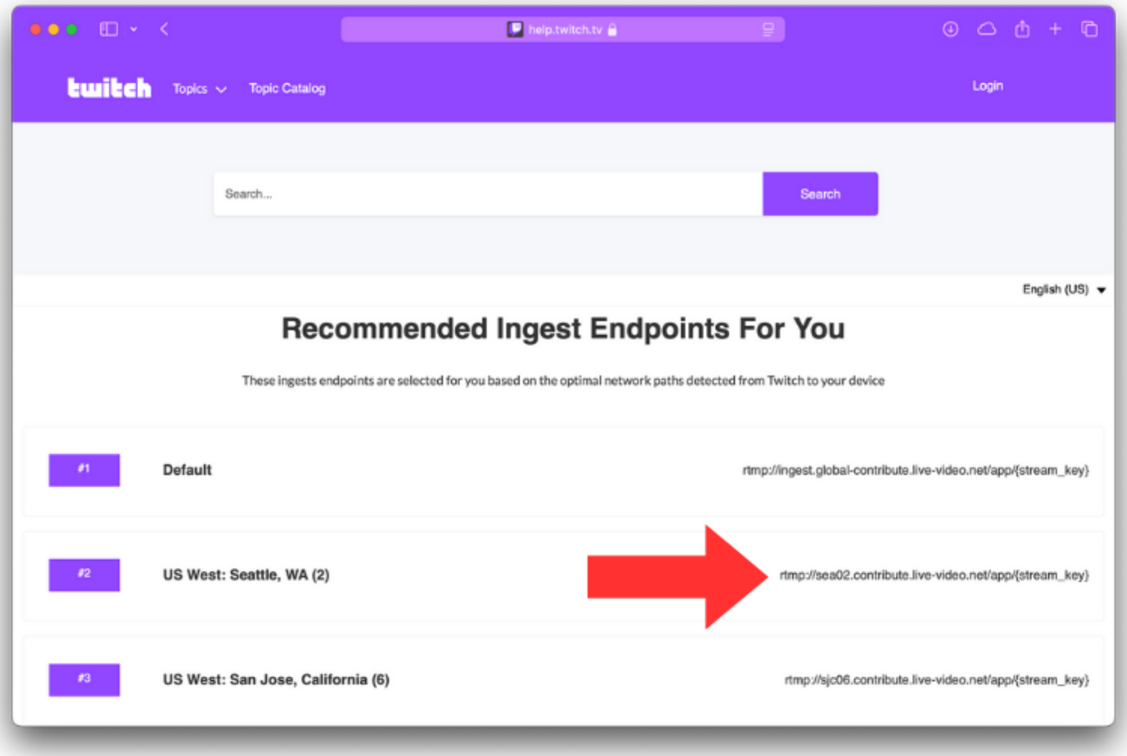
Copyrighted Audio Warnings

When turned on, you'll be notified if multiple instances of copyrighted audio are detected in your VODs within a 24-hour time period, and your ability to publish VODs by default on your channel will be proactively disabled. By turning this setting off, you agree that you understand that including unauthorized audio in your content may result in rights holders sending copyright notifications or taking other actions against your channel. Learn more in this Help Article

VOD Settings

Now, you need to find your local ingest server, essentially, this takes the video from your camera and sends it to Twitch.

Copy the URL of the recommended server: https://help.twitch.tv/s/twitch-ingest-recommendation



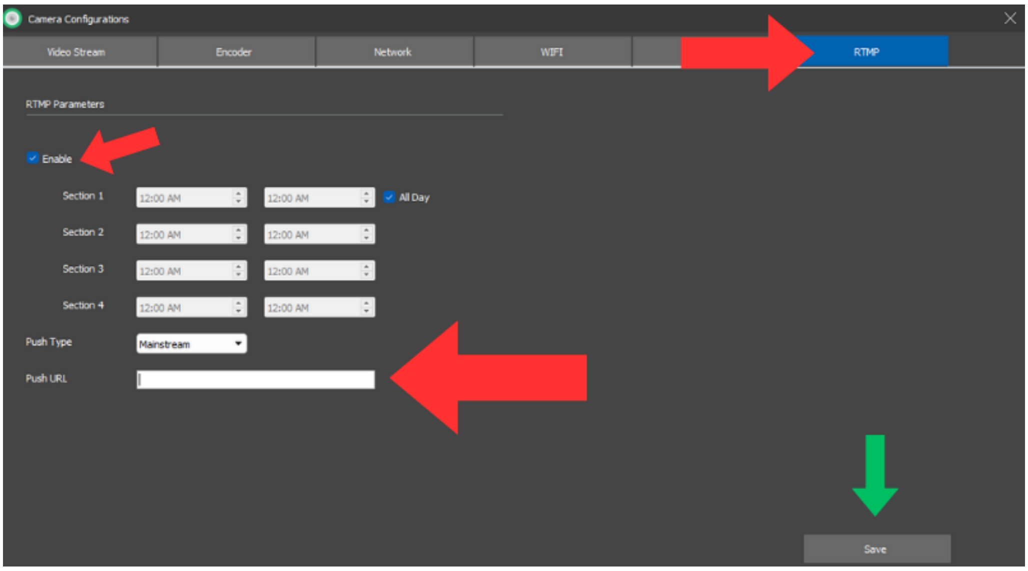
Recommended Ingest Endpoints For You

These ingests endpoints are selected for you based on the optimal network paths detected from Twitch to your device

#1	Default	rtmp://ingest.global-contribute.live-video.net/app/{stream_key}
#2	US West: Seattle, WA (2)	rtmp://sea02.contribute.live-video.net/app/{stream_key}
#3	US West: San Jose, California (6)	rtmp://sjc06.contribute.live-video.net/app/{stream_key}

Create your Stream URL by adding your stream key copied from the Twitch dashboard to the end of the URL.

Part Three: Connecting the Camera & Twitch



RTMP Parameters

☑ Enable

Section 1 12:00 AM 12:00 AM ☑ All Day

Section 2 12:00 AM 12:00 AM

Section 3 12:00 AM 12:00 AM

Section 4 12:00 AM 12:00 AM

Push Type Mainstream

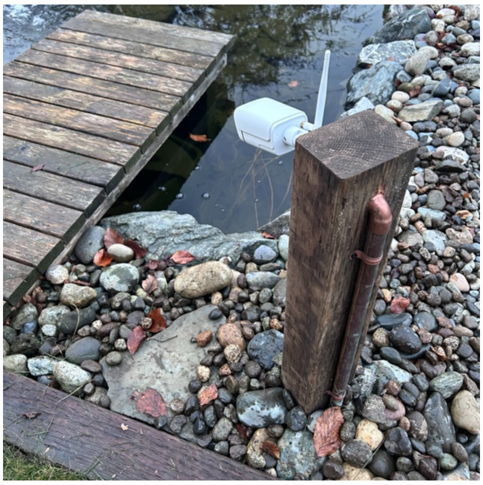
Push URL

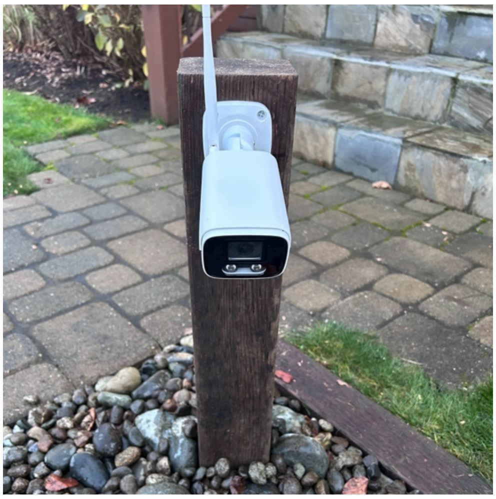
Copy your stream URL result into the “Push URL” field on the camera software, save it, and close the program. After a few minutes, check your Twitch stream and verify that it was successful.

⏯ Pause: Verify your camera is working by checking your Twitch stream before mounting.

Part Four: Mounting the Camera

Mount the camera in a space with a GREAT WiFi connection; this will reduce buffering on Twitch. I mounted mine outside, far away from my router, and the connection is not great





Section Two: Creating The Brain

Summary:

In this section, you will learn how to use the microcontroller (WEMOS D1) to control a stepper motor over the internet via the Blynk service.

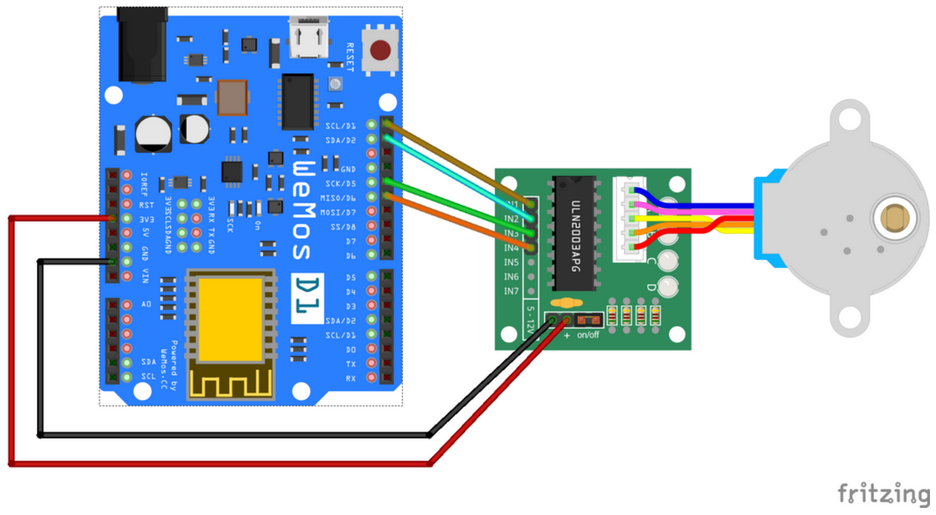
Video:

Blynk With ESP32: https://youtu.be/T45r2ZldTZo?si=352IuRWdBJe7BgUN

Parts, Materials, & Software:

- Arduino IDE (https://www.arduino.cc/en/software)
  - Code: (https://github.com/Tech-Trends-Shameer/Blynk-2.0-Projects/blob/main/Control-Stepper-Motor-Using-ESP-8266-and-Blynk-IoT/control-stepper-motor-using-esp8266-and-blynk-iot.ino)
- Blynk Account (https://blynk.cloud)
- Wemos D1 or D1 Mini (https://a.co/d/fKpZDha)
- 28BYJ-48 Stepper Motor & Driver (https://a.co/d/f08uF1a)
- Micro USB Cord and Wall Adaptor

Part One: Connecting the Wires



WeMos	Driver
3V3	+
GND	–
D1	IN1
D2	IN2
D5	IN3
D6	IN4

Connect the wires from the stepper driver to the WeMos board using the schematic

Part Two: Set Up Blynk

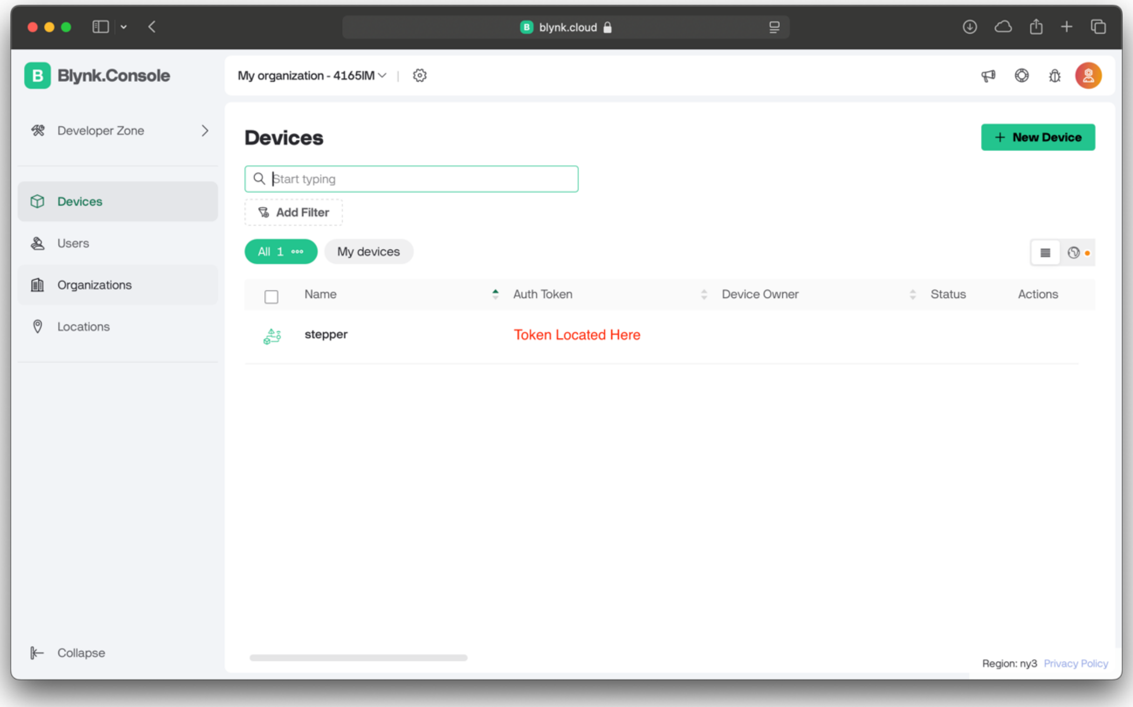
Following the steps in the attached video, create a Blynk account, configure the virtual pins, and create a dashboard to test the fish feeder.

Once your dashboard has been created, create a webhook to control the motor connected to the fish feeder.

Using the webhook template below, update the RED with your own token. *Save this URL for later.*

https://ny3.blynk.cloud/external/api/update?token=yld7fSxKoKs8WQrOjDyk4iAJbUitL250&v1=1

- v1=1 is when the stepper motor is ON
- v1=0 is when the stepper motor is OFF

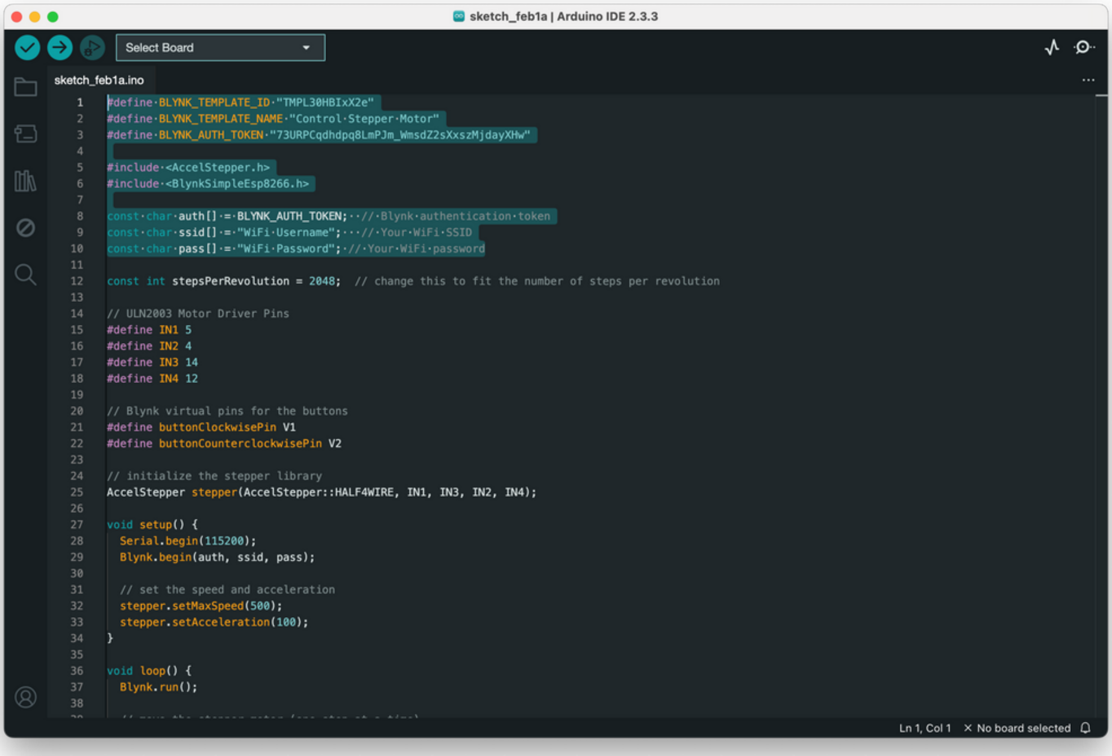


Name	Auth Token	Device Owner	Status	Actions
stepper	Token Located Here			

💡 A webhook is a URL that your code “looks up” on your behalf that can trigger an action on your hardware. In this case, we have a link that turns the motor off and one that turns the motor on.

Part Three: Upload the Code

Open the Arduino IDE and copy the code from the provided GitHub repository.



```
#define BLYNK_TEMPLATE_ID "TMPL30HBIxX2e"
#define BLYNK_TEMPLATE_NAME "Control Stepper Motor"
#define BLYNK_AUTH_TOKEN "73URPCqdhdpq8LmPJm_WmsdZ2sXxszMjdayXHw"

#include <AccelStepper.h>
#include <BlynkSimpleEsp8266.h>

const char auth[] = BLYNK_AUTH_TOKEN;  // Blynk authentication token
const char ssid[] = "WiFi Username";   // Your WiFi SSID
const char pass[] = "WiFi Password"; // Your WiFi password

const int stepsPerRevolution = 2048;  // change this to fit the number of steps per revolution

// ULN2003 Motor Driver Pins
#define IN1 5
#define IN2 4
#define IN3 14
#define IN4 12

// Blynk virtual pins for the buttons
#define buttonClockwisePin V1
#define buttonCounterclockwisePin V2

// initialize the stepper library
AccelStepper stepper(AccelStepper::HALF4WIRE, IN1, IN3, IN2, IN4);

void setup() {
  Serial.begin(115200);
  Blynk.begin(auth, ssid, pass);

  // set the speed and acceleration
  stepper.setMaxSpeed(500);
  stepper.setAcceleration(100);
}

void loop() {
  Blynk.run();

```

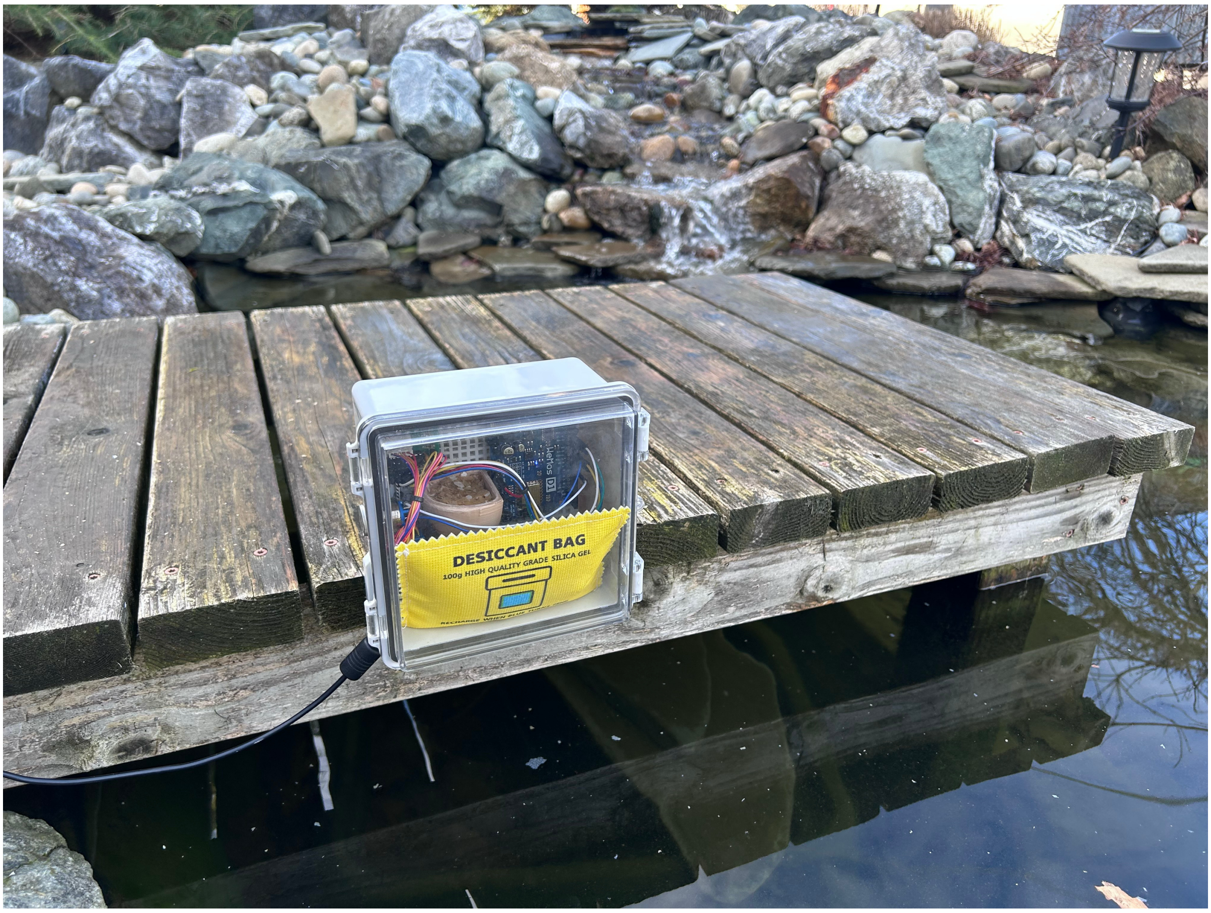
Update the highlighted text with the information provided from the Blynk online dashboard.

⏯️ Pause: Verify that the code will compile correctly by clicking the checkmark in the top right of the screen. You may need to download additional Arduino libraries to your computer.

Plug the board into your computer, select the board name, and click the arrow to send the code.

Unplug the board from your computer and monitor the Blynk dashboard for a successful connection.

Section Three: Creating the Weatherproof Housing



Summary:

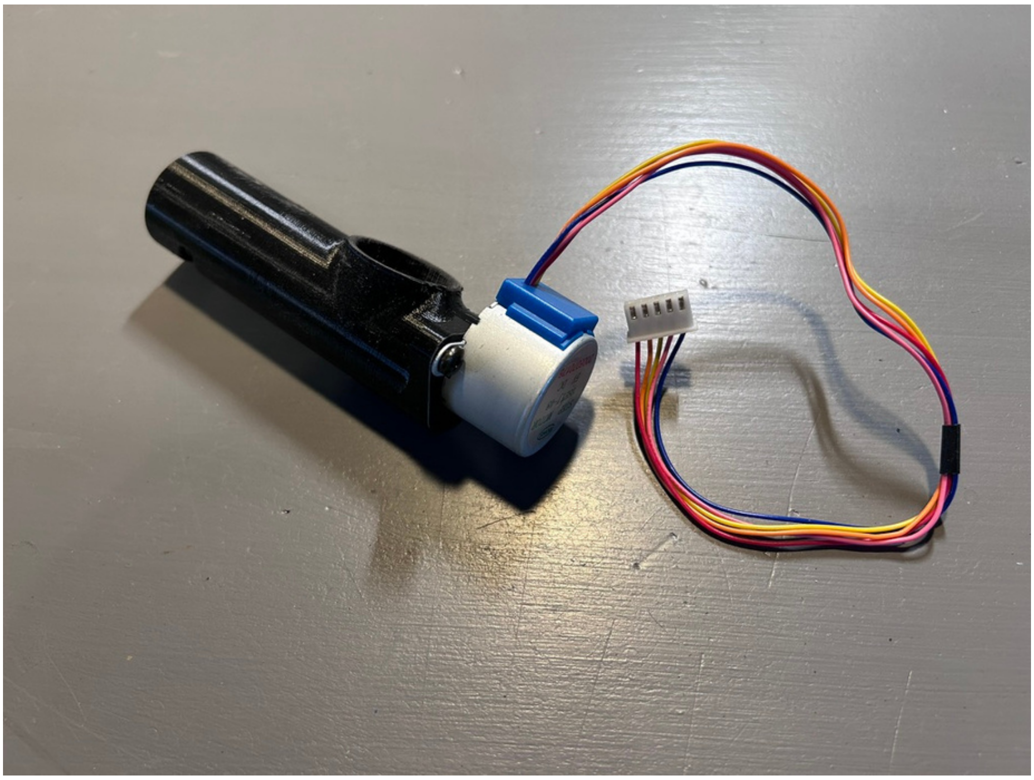
In this section, you will learn how to 3D print the fish food augar, and waterproof the computer.

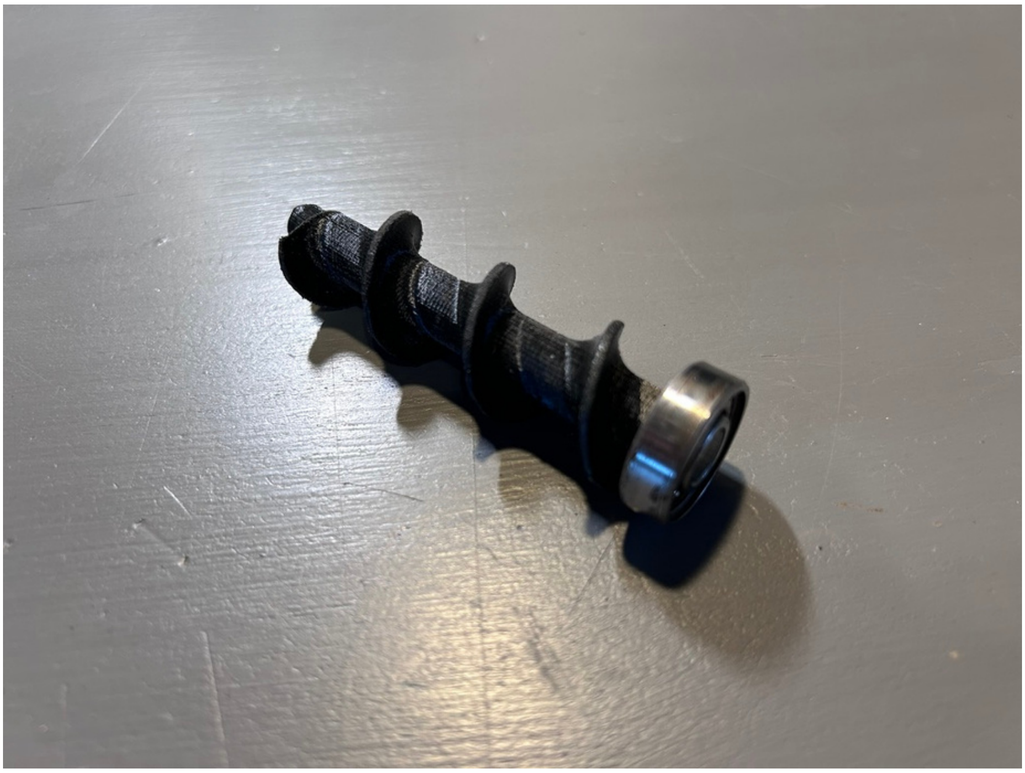
Parts, Materials, & Tools:

- Augar Feeder STL: https://www.thingiverse.com/thing:2959685
- 3D Printer
- Waterproof Box (https://a.co/d/3iuaJoK)
- Moisture Absorption Bag (https://a.co/d/eVfFtRL)

Part One: Print the Feeding Mechanism

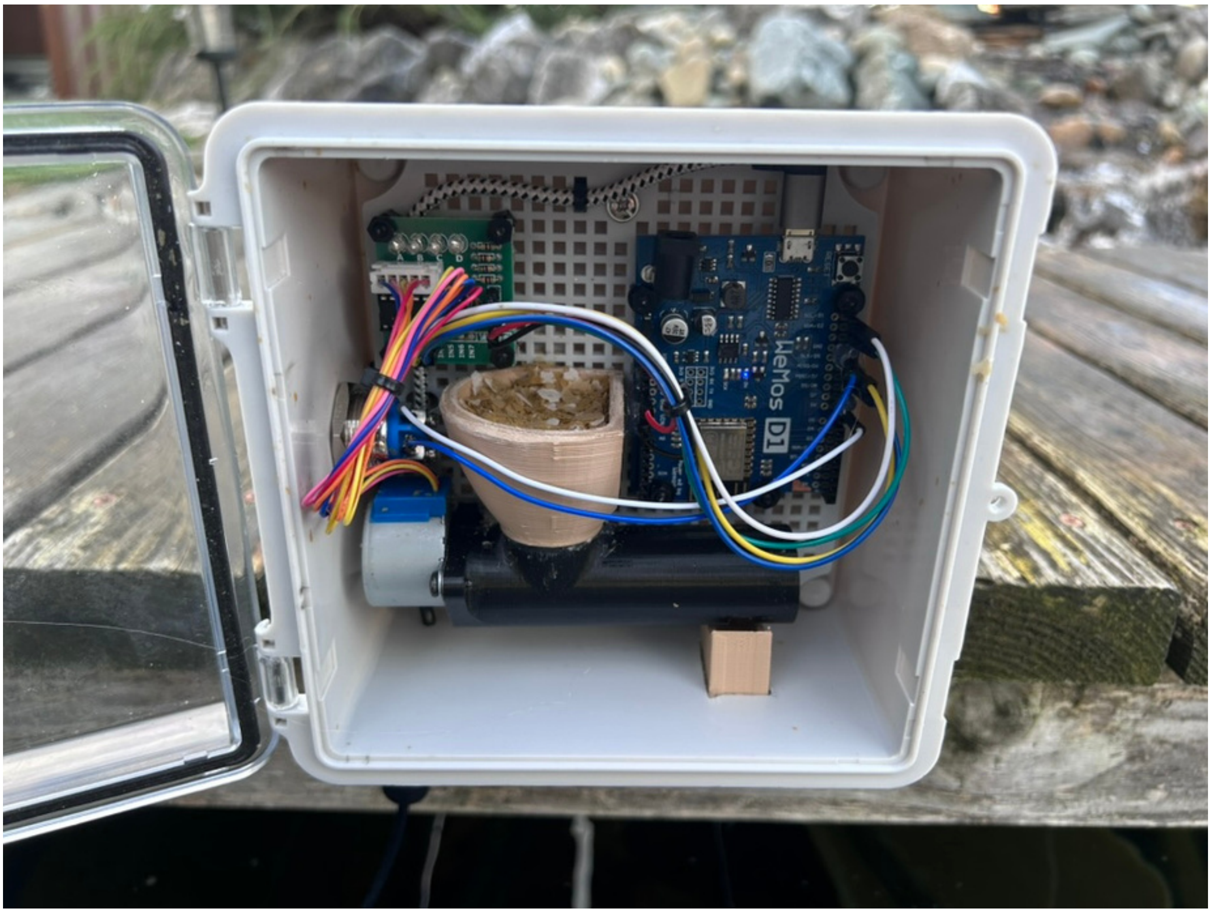
Using a 3D printer, print the fish feeder mechanism and assemble it by attaching the servo and adding a ball bearing. Printing the funnel is optional.





Part Two: Assemble the Box

Take the fish feeder mechanism, controller, and moisture absorber pouch and mount it inside the waterproof box.

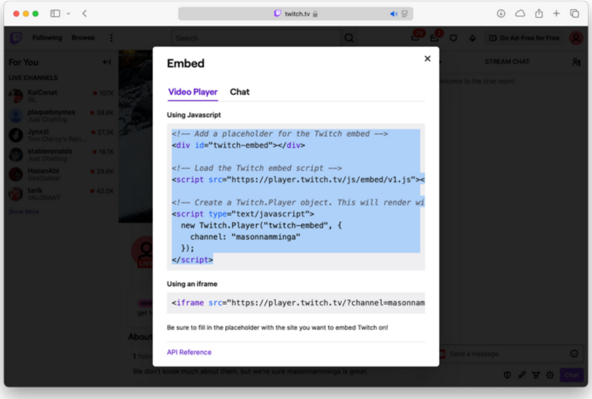


Ensure that there is a hole cut for the food to dispense. Place the box in a location with a good WiFi signal. The project box in the example has a pegboard back that allows for easy mounting of components.

Section Four: Building the Webpage

Summary:

In this section, you will learn how to create an HTML skeleton with an embedded Twitch stream and a button to dispense the fish food.



```
<!-- Add a placeholder for the Twitch embed -->
<div id="twitch-embed"></div>

<!-- Load the Twitch embed script -->
<script src="https://player.twitch.tv/js/embed/v1.js"><

<!-- Create a Twitch.Player object. This will render wi
<script type="text/javascript">
  new Twitch.Player("twitch-embed", {
    channel: "masonnamminga"
  });
</script>
```

```
<iframe src="https://player.twitch.tv/?channel=masonnam
```

Be sure to fill in the placeholder with the site you want to embed Twitch on!

Software:

- Visual Studio Code IDE
- GitHub

Part One: Creating an HTML page

Create a new HTML file in Visual Studio and embed the Twitch stream using the embed code from the Twitch share sheet.

Part Two: Creating a Trigger Button

Now you want a way for users to control the fish feeder, using the on and off webhooks previously created, we can create a button in HTML, as well as a JavaScript function that calls the on webhook, waits five seconds, and then calls the off webhook when the user clicks the feed the fish button.

```
function feedthem() {
  var xhr1 = new XMLHttpRequest();
  xhr1.open('GET', 'ADD ON WEBHOOK HERE', true);
  xhr1.onreadystatechange = function() {
    if (xhr1.readyState === 4 && xhr1.status === 200) {
      console.log('First request successful');
      setTimeout(function() {
        var xhr2 = new XMLHttpRequest();
        xhr2.open('GET', 'ADD OFF WEBHOOK HERE', true);
        xhr2.onreadystatechange = function() {
          if (xhr2.readyState === 4 && xhr2.status === 200) {
            console.log('Second request successful');
          }
        };
        xhr2.send();
      }, 5000);
    }
  };
  xhr1.send();
}
```

*This code was created by generative AI.

Part Three: Publish the Site

Using GitHub pages, publish the site and share it with friends so others can enjoy your creation!