



# VIRTUAL DRAWING SYSTEM

COURSE: INTERNET OF THINGS  
HOST: GUY TEL ZUR

BY: LIRON ADI, OMAR JABER

# PROBLEM DEFINITION & MOTIVATION

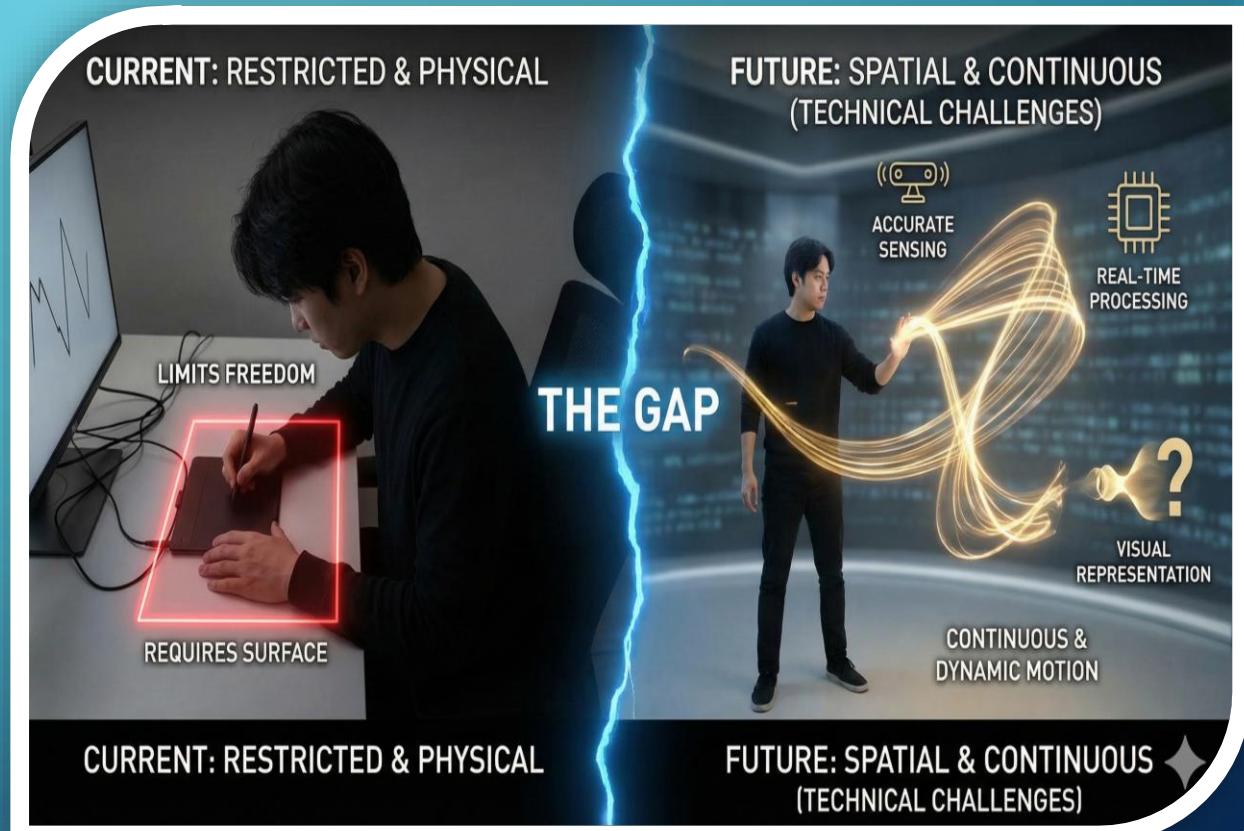
## Problem Definition:

Digital drawing is usually performed using physical contact with a screen or a dedicated drawing device

These methods:

- Limit freedom of movement
- Require a physical surface
- Do not utilize natural spatial motion

There is a gap between human motion and digital creation



# PROBLEM DEFINITION & MOTIVATION

## Technical Perspective of the Problem:

Hand motion is continuous and dynamic  
but is rarely used as a direct input

### Main challenges:

- Accurate motion sensing
- Real-time processing
- Meaningful visual representation

Most systems focus on analysis,  
not on visualizing motion itself

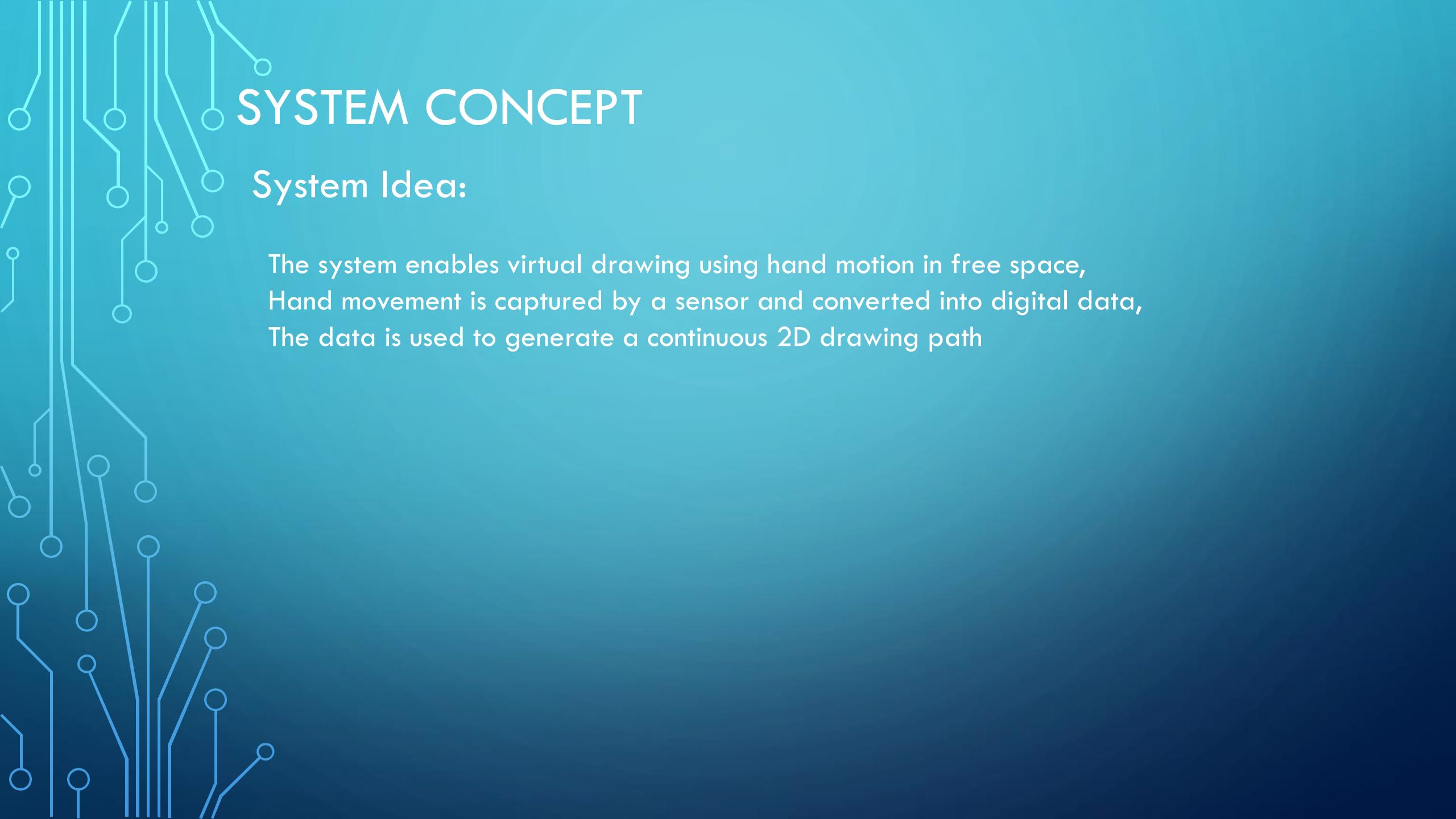
# PROBLEM DEFINITION & MOTIVATION

## Motivation:

The motivation is to enable:

- Drawing without physical contact
- Use of natural hand movements
- Free creation in space

IoT technologies enable real-time connection between  
the physical and digital worlds



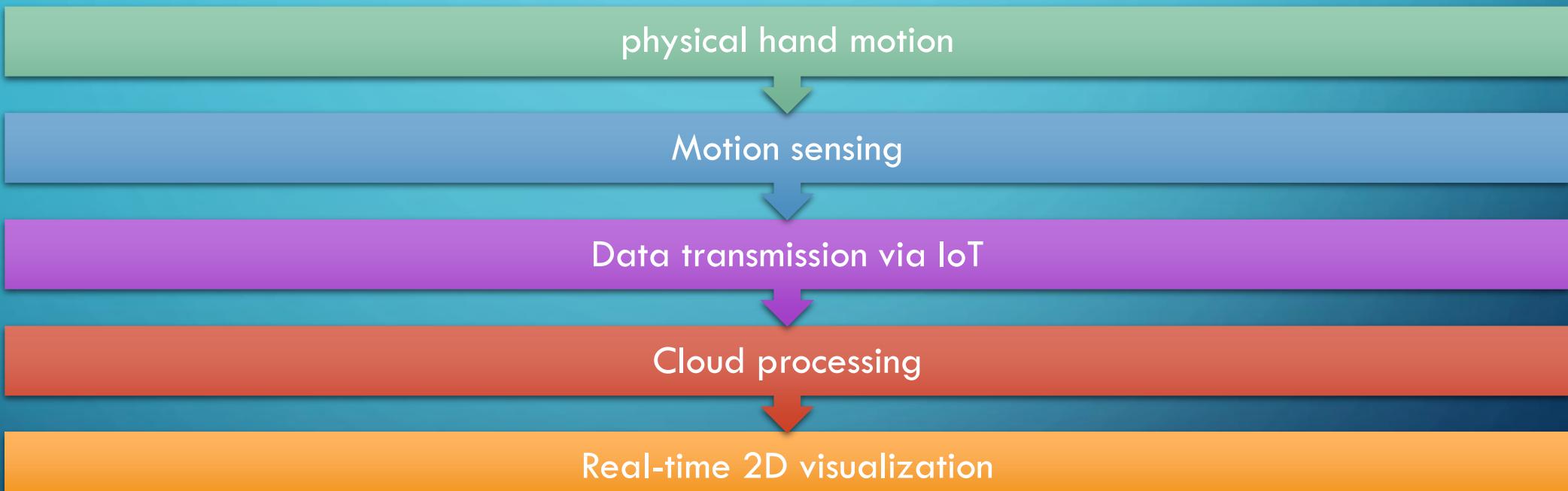
# SYSTEM CONCEPT

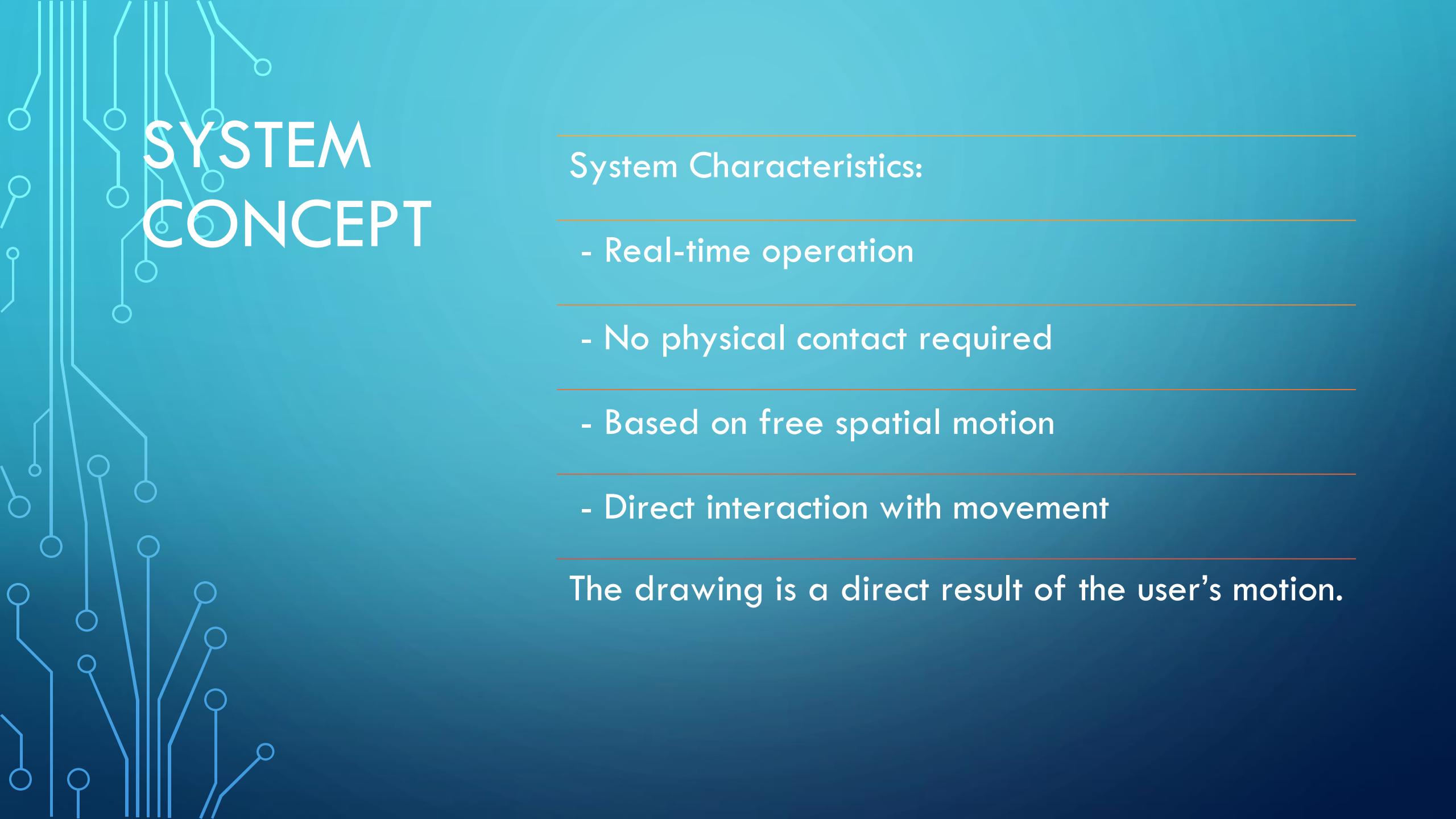
## System Idea:

The system enables virtual drawing using hand motion in free space,  
Hand movement is captured by a sensor and converted into digital data,  
The data is used to generate a continuous 2D drawing path

# SYSTEM CONCEPT

## Overall Operation Flow:





# SYSTEM CONCEPT

---

## System Characteristics:

---

- Real-time operation
  - No physical contact required
  - Based on free spatial motion
  - Direct interaction with movement
- 

The drawing is a direct result of the user's motion.

# PROJECT OBJECTIVES

## General Objectives:

Develop a complete IoT-based system for motion tracking and visualization.

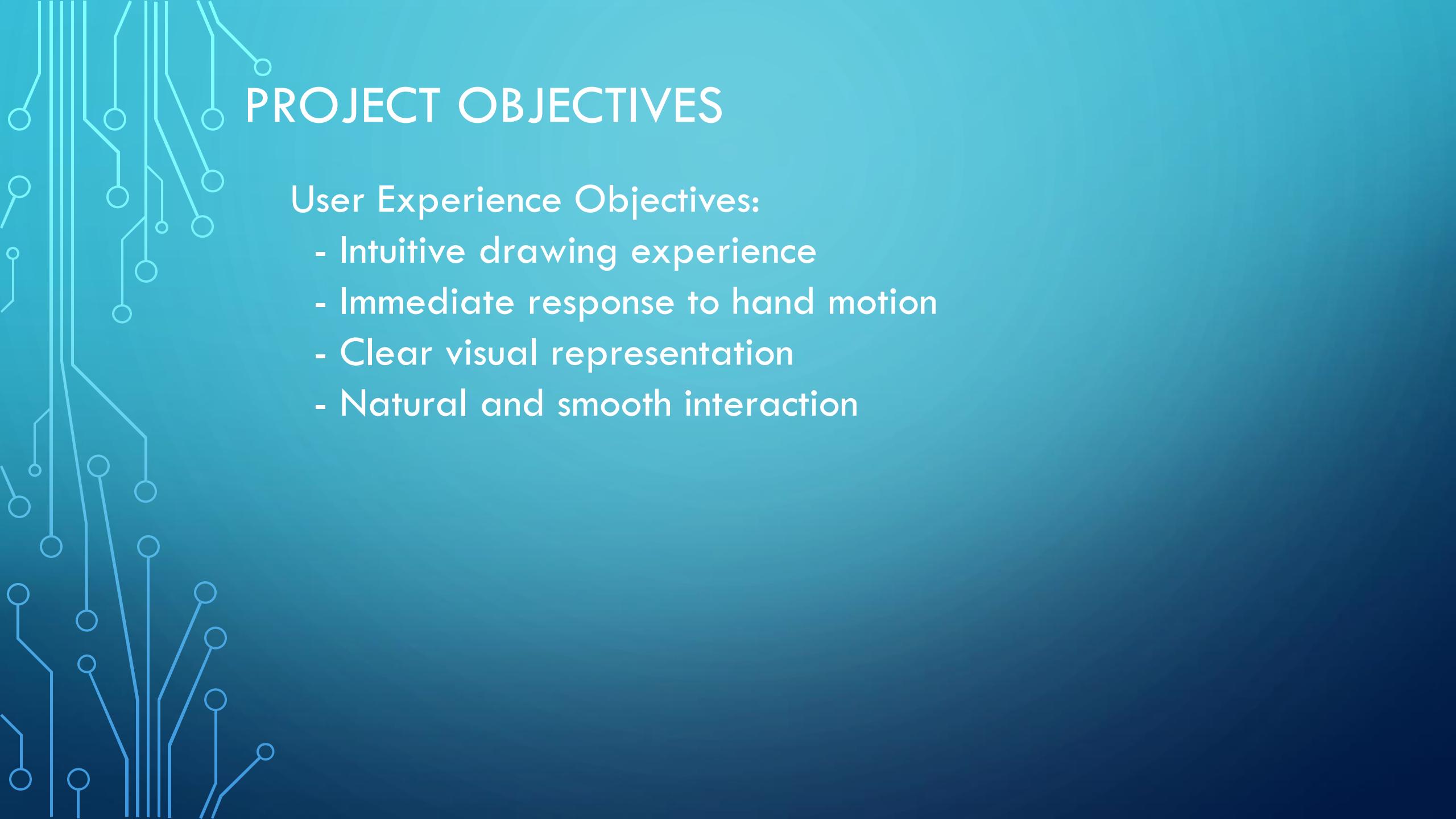
## Integrating:

- Sensors
- Wireless communication
- Cloud processing
- Visual rendering

# PROJECT OBJECTIVES

## Technical Objectives:

- Continuous real-time motion sensing
- Use of an IMU sensor
- Wireless data transmission via ESP32
- Cloud-based data processing
- Generation of a smooth 2D trajectory



# PROJECT OBJECTIVES

## User Experience Objectives:

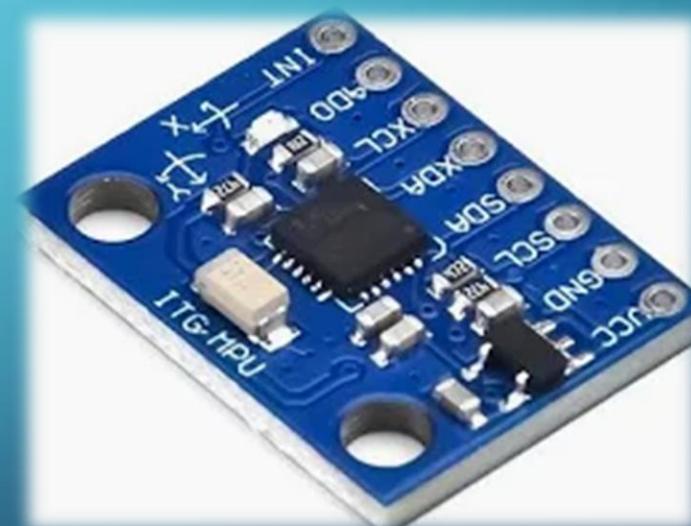
- Intuitive drawing experience
- Immediate response to hand motion
- Clear visual representation
- Natural and smooth interaction

# MOTION SENSOR EXPLANATION

## Motion Sensor in the System:

The system uses the MPU6050, an IMU motion sensor,

The sensor is attached to a stick, and tracks hand movement in space



# MOTION SENSOR EXPLANATION

## What Does the Sensor Measure?

The sensor includes:

3-axis accelerometer

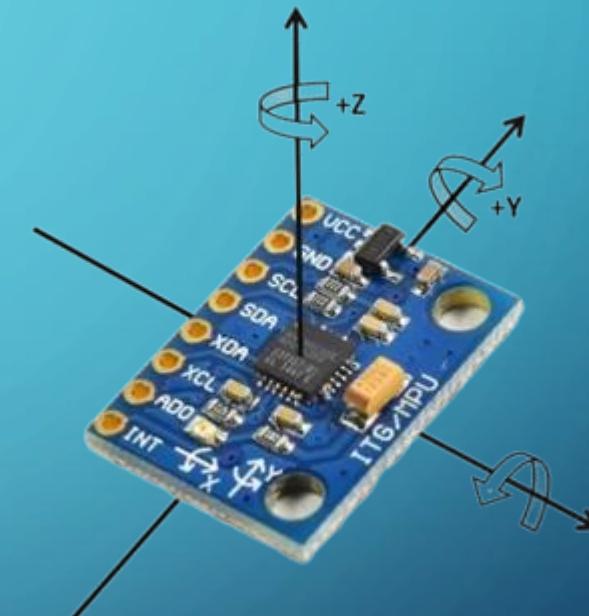
3-axis gyroscope

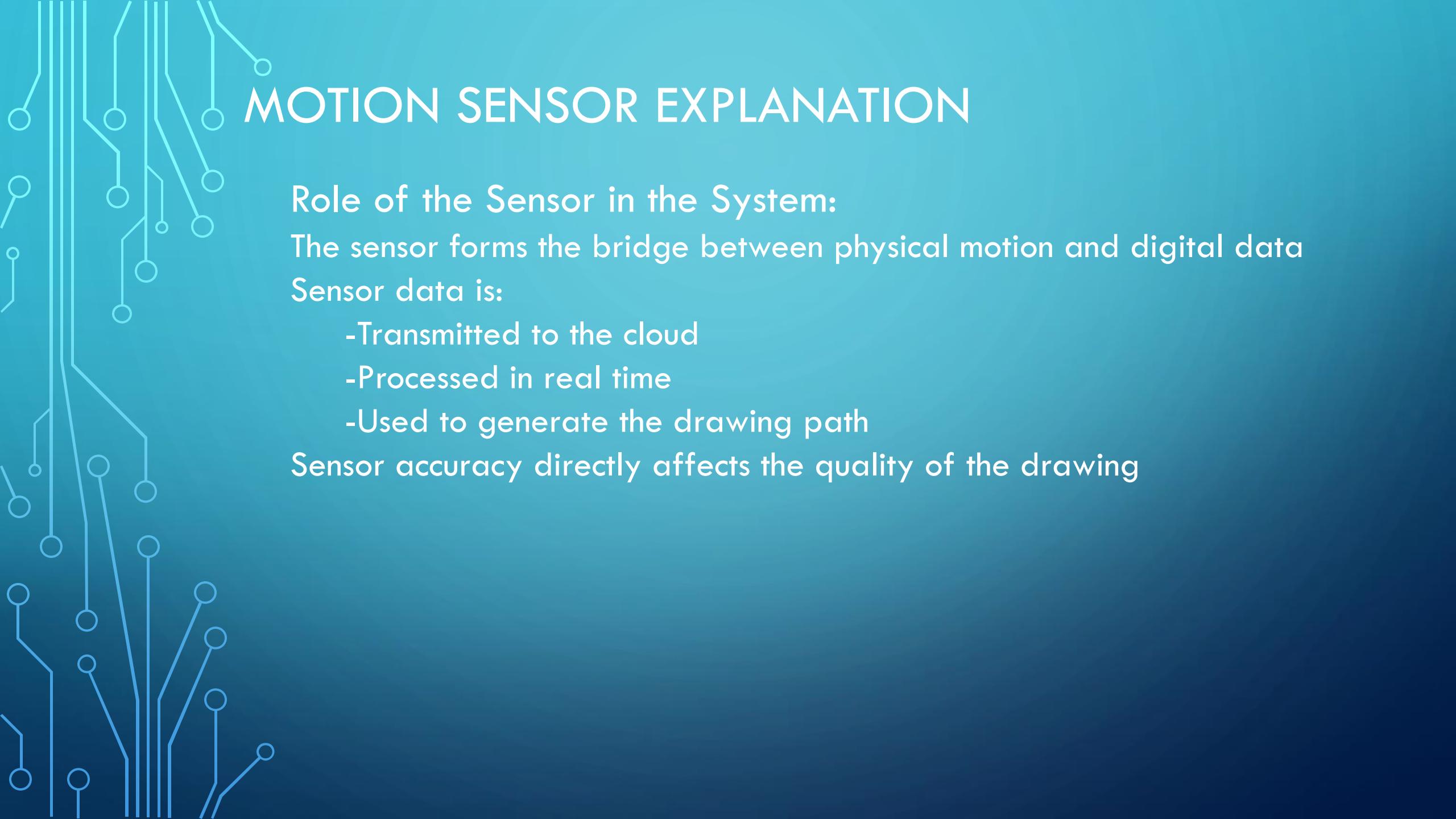
These measurements enable:

Motion detection

Rotation tracking

Continuous movement tracking





# MOTION SENSOR EXPLANATION

## Role of the Sensor in the System:

The sensor forms the bridge between physical motion and digital data

Sensor data is:

- Transmitted to the cloud
- Processed in real time
- Used to generate the drawing path

Sensor accuracy directly affects the quality of the drawing

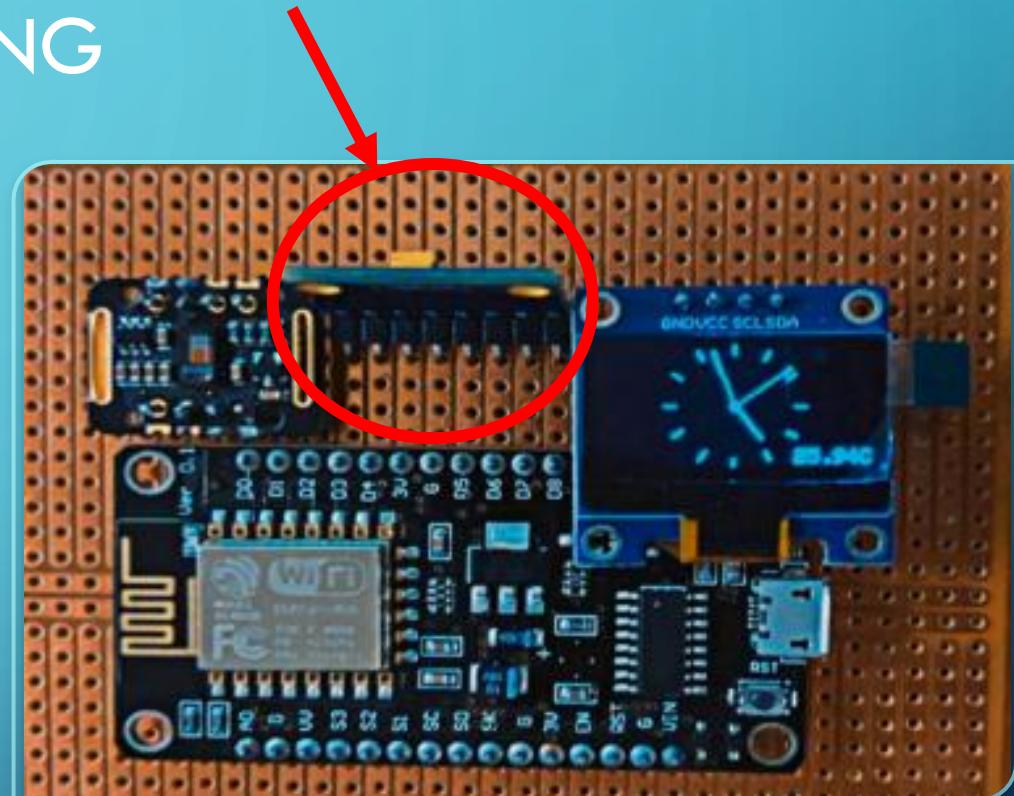
## RELATED WORK – MOTION TRACKING USING THE MPU-6050 SENSOR

### Content:

- Extensive use of IMU sensors for hand motion tracking
- Accelerometer and gyroscope measurements in wearable systems
- Integration of IoT technologies for real-time data transmission

### Applications:

Fitness monitoring, Rehabilitation systems, Motion analysis and tracking



# GENERAL OVERVIEW OF THE RESEARCH STUDY

The study presents a complete IoT-based system designed for real-time motion tracking

- A wearable system attached to the user's hand
- A microcontroller integrated with WiFi connectivity
- Sensor data transmission to the cloud via ThingSpeak
- Real-time data processing and analysis

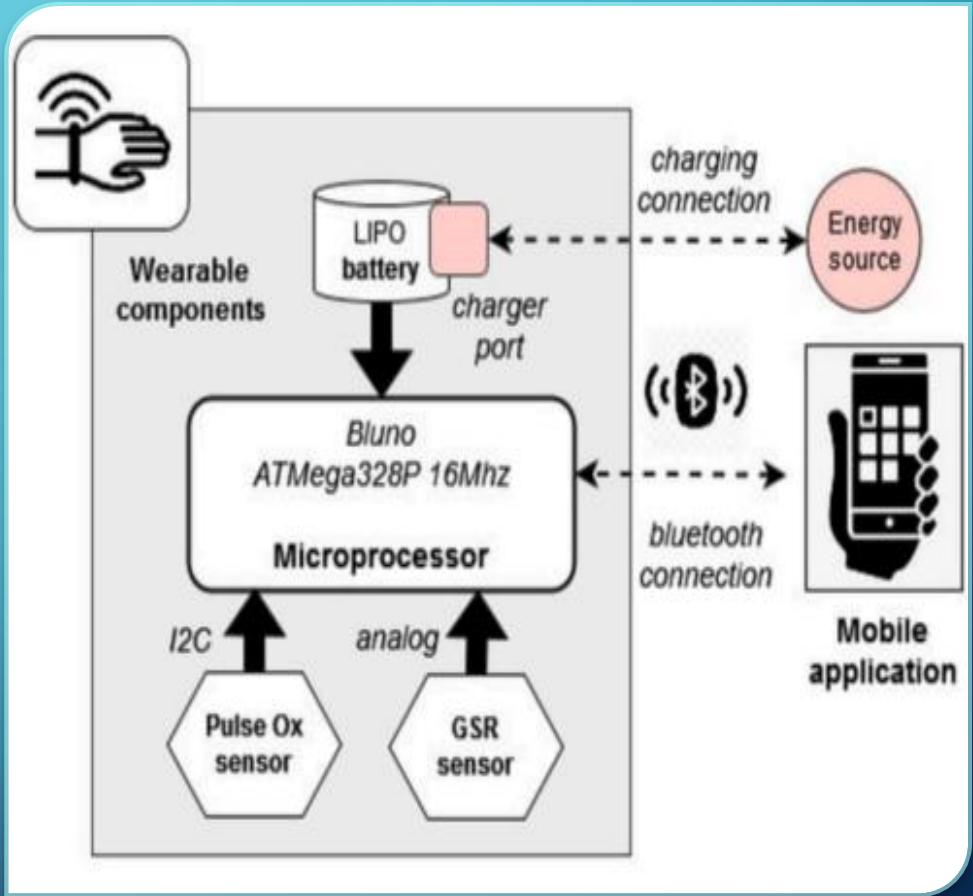
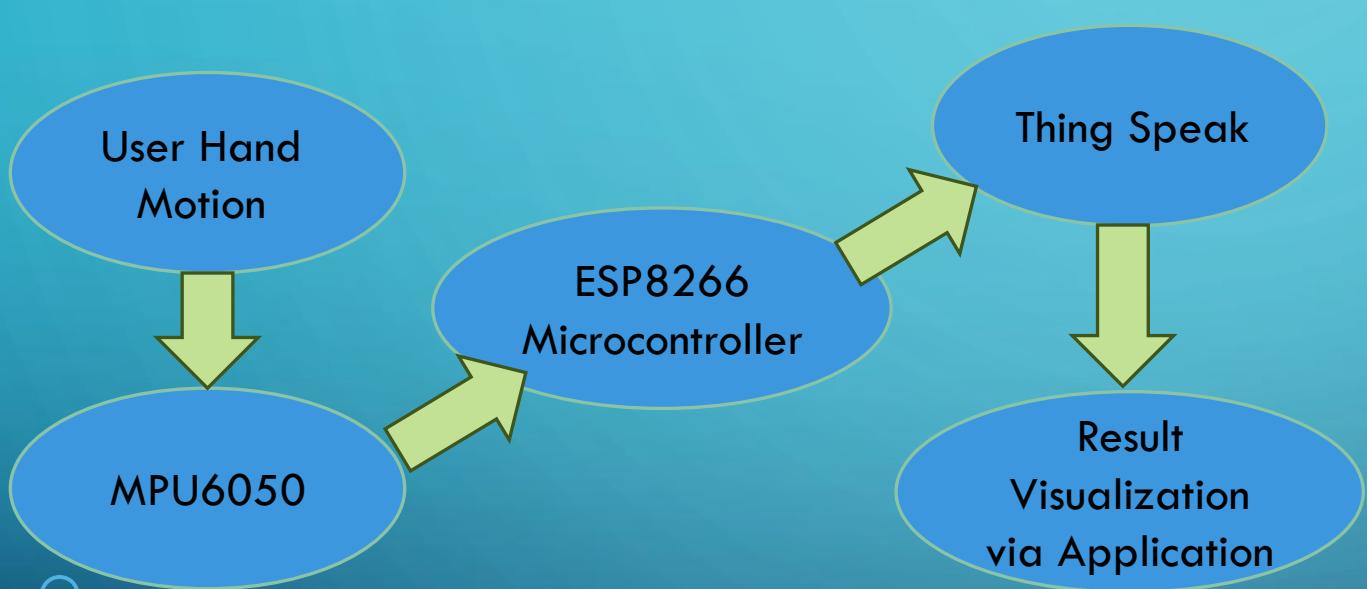
*FitnessFirst*



Sign Up

Already have an account?

# SYSTEM ARCHITECTURE IN THE RESEARCH STUDY



# CLOUD-BASED DATA PROCESSING AND MOTION CLASSIFICATION

- Acceleration data collected from the X, Y, and Z axes
- Sensor data transmitted from the microcontroller to the cloud
- Noise filtering and data preprocessing
- Application of machine learning algorithms

## Algorithms used:

- Fine KNN
- Support Vector Machine (SVM)
- Boosted Trees

## Best achieved accuracy:

91.3%

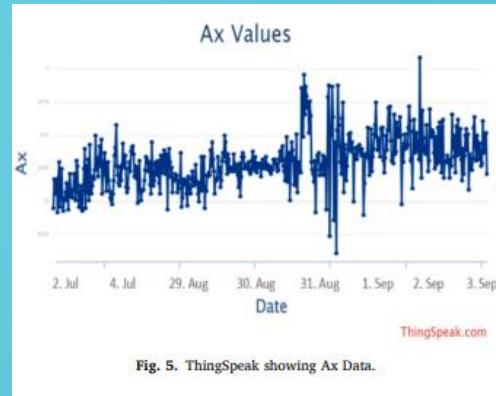
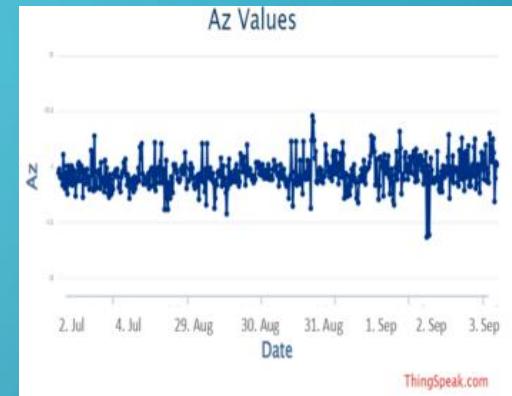
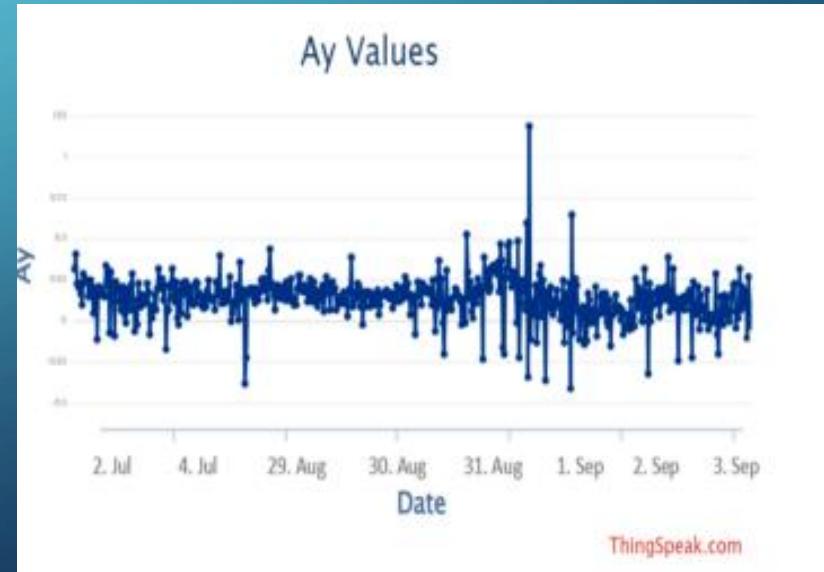


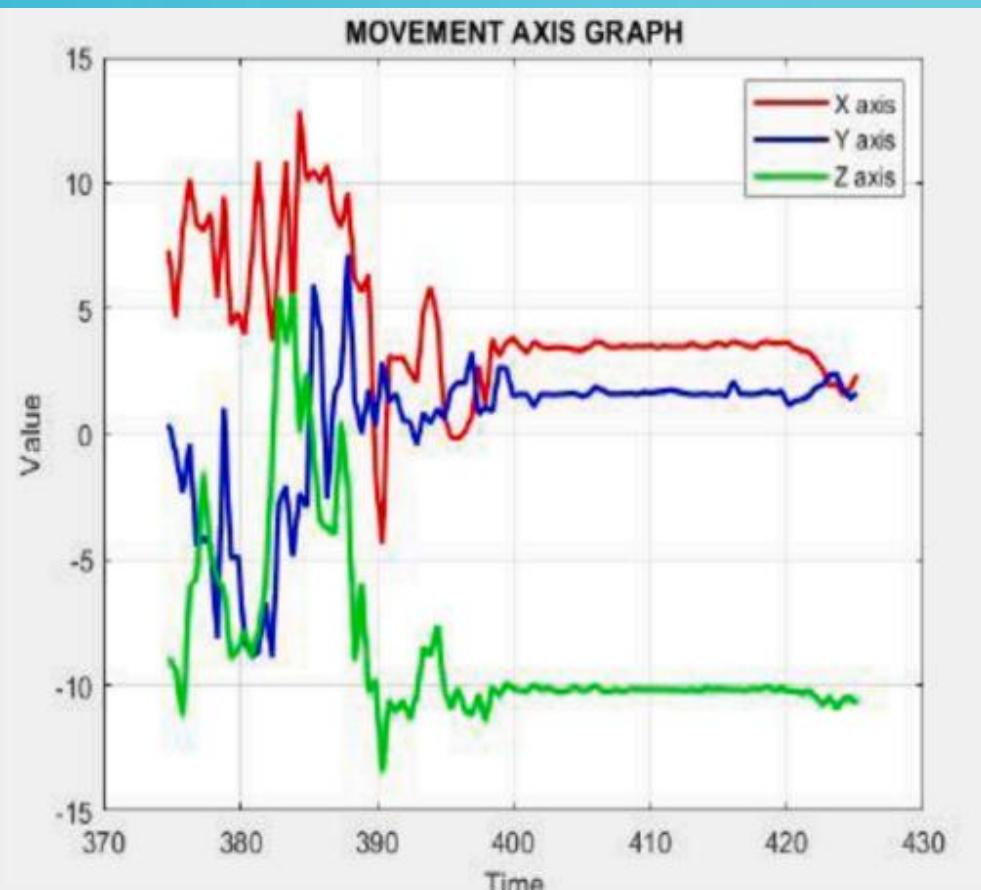
Fig. 5. ThingSpeak showing Ax Data.



ThingSpeak.com



ThingSpeak.com



**Table 1**  
Accuracy of Algorithms.

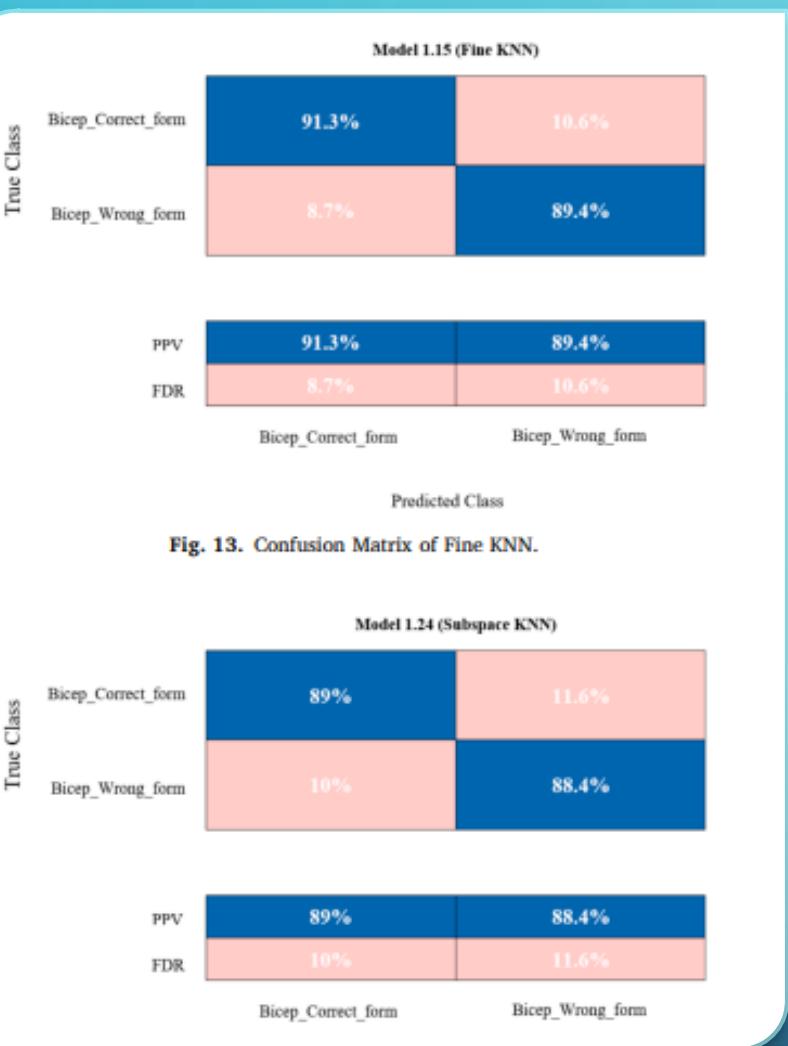
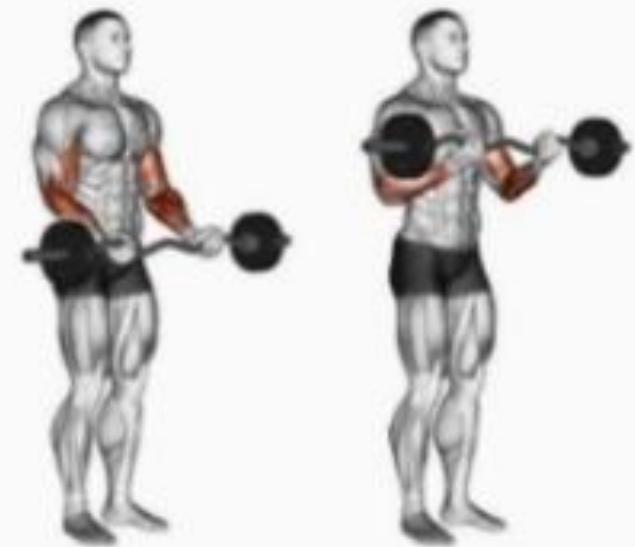
Algorithm Name	Precision	Recall	F1Score	Accuracy %
Fine KNN	0.90	0.92	0.90	91.3
Quadratic SVM	0.94	0.81	0.87	85.2
Boosted Trees	0.94	0.81	0.87	85.5
Subspace KNN	0.91	0.88	0.89	89.0

# WHAT DOES THE STUDY SUCCESSFULLY ACHIEVE ?

- Accurate classification of hand movements
  - Real-time system operation
  - Integration of sensors, IoT, and cloud computing
  - A stable and practical system implementation

Command Window

# Bicep Screen



## LIMITATIONS OF THE EXISTING STUDY

Motion classification only

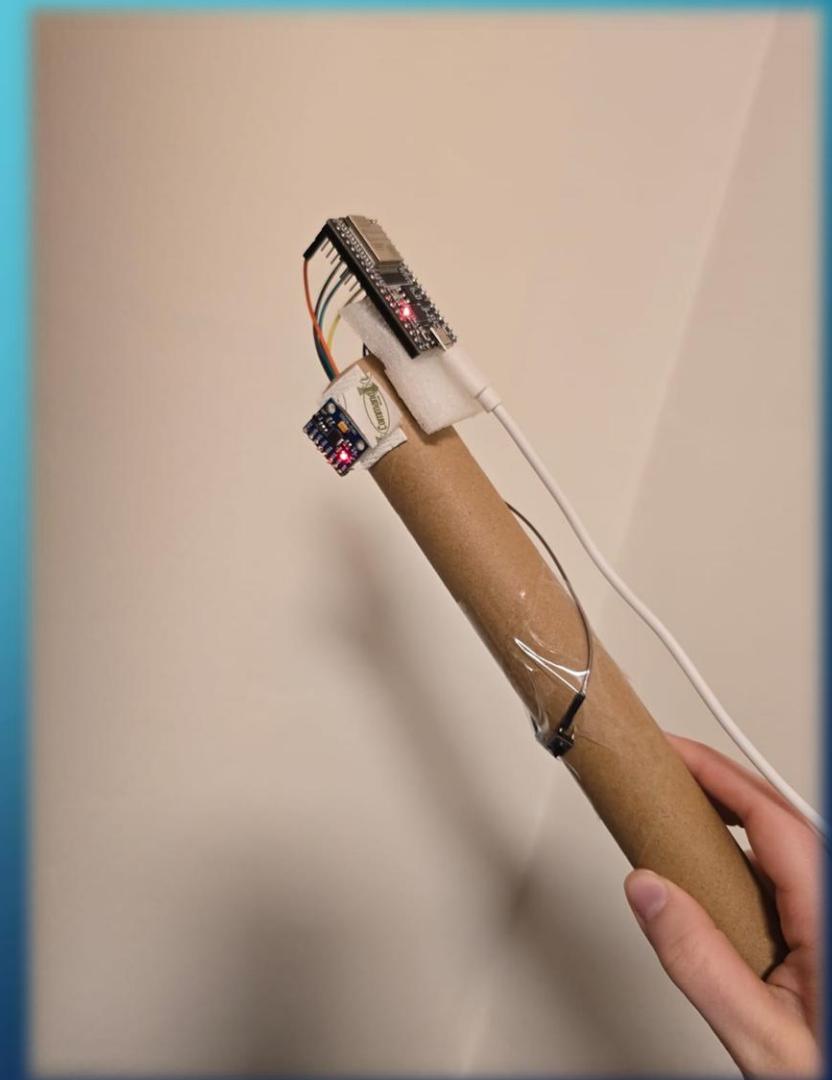
No continuous trajectory generation

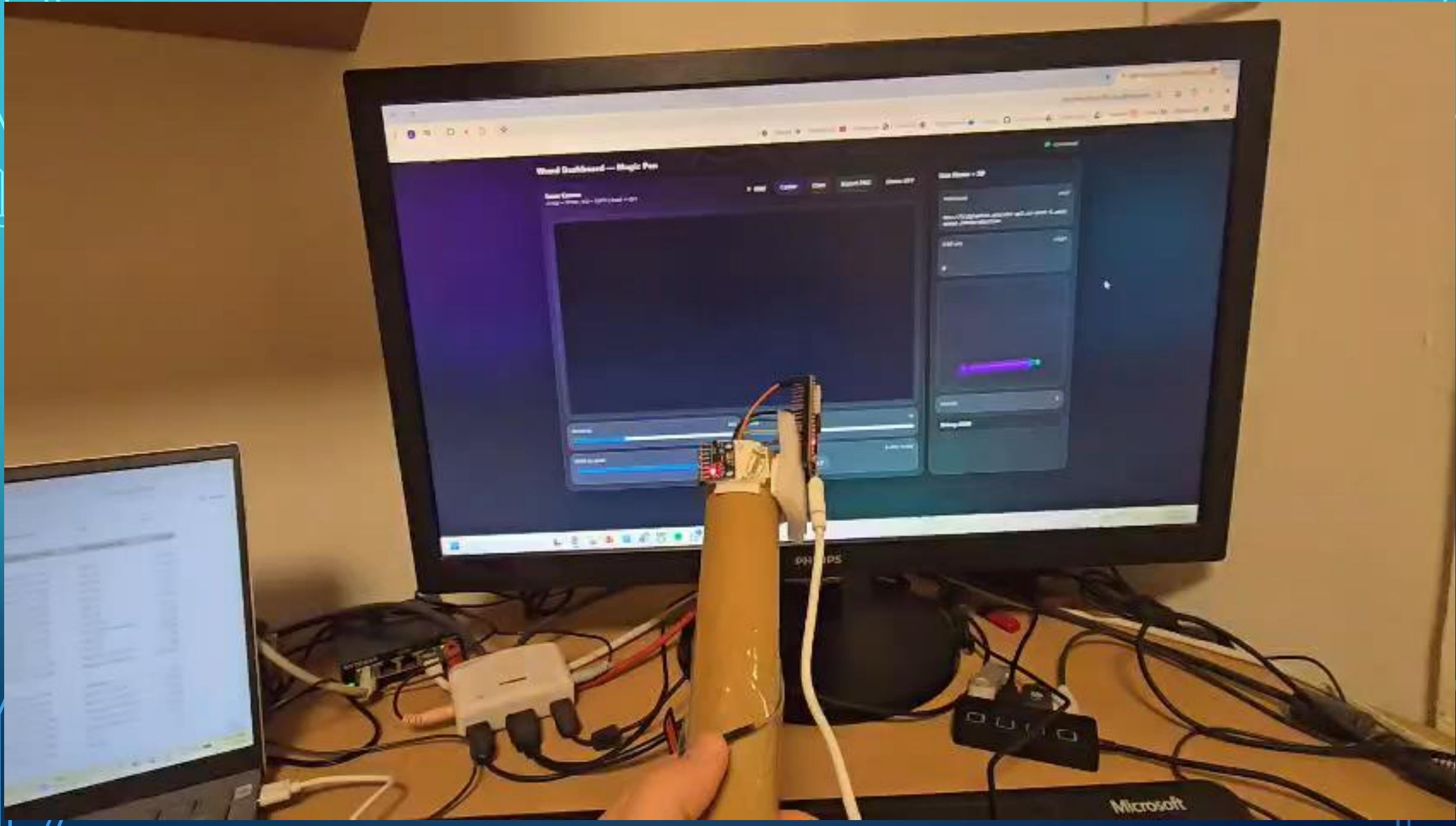
No geometric representation of motion

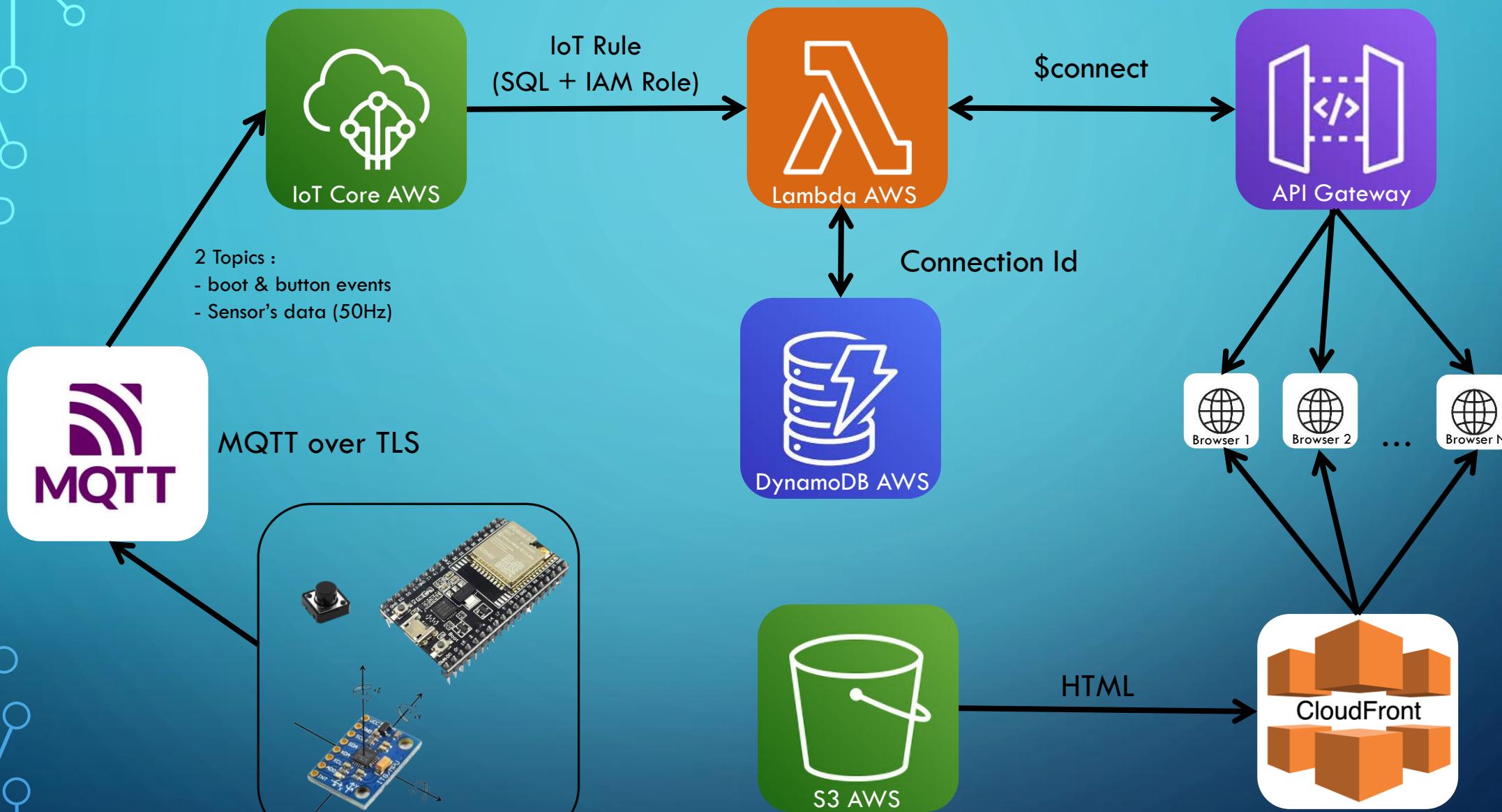
No free or interactive visualization

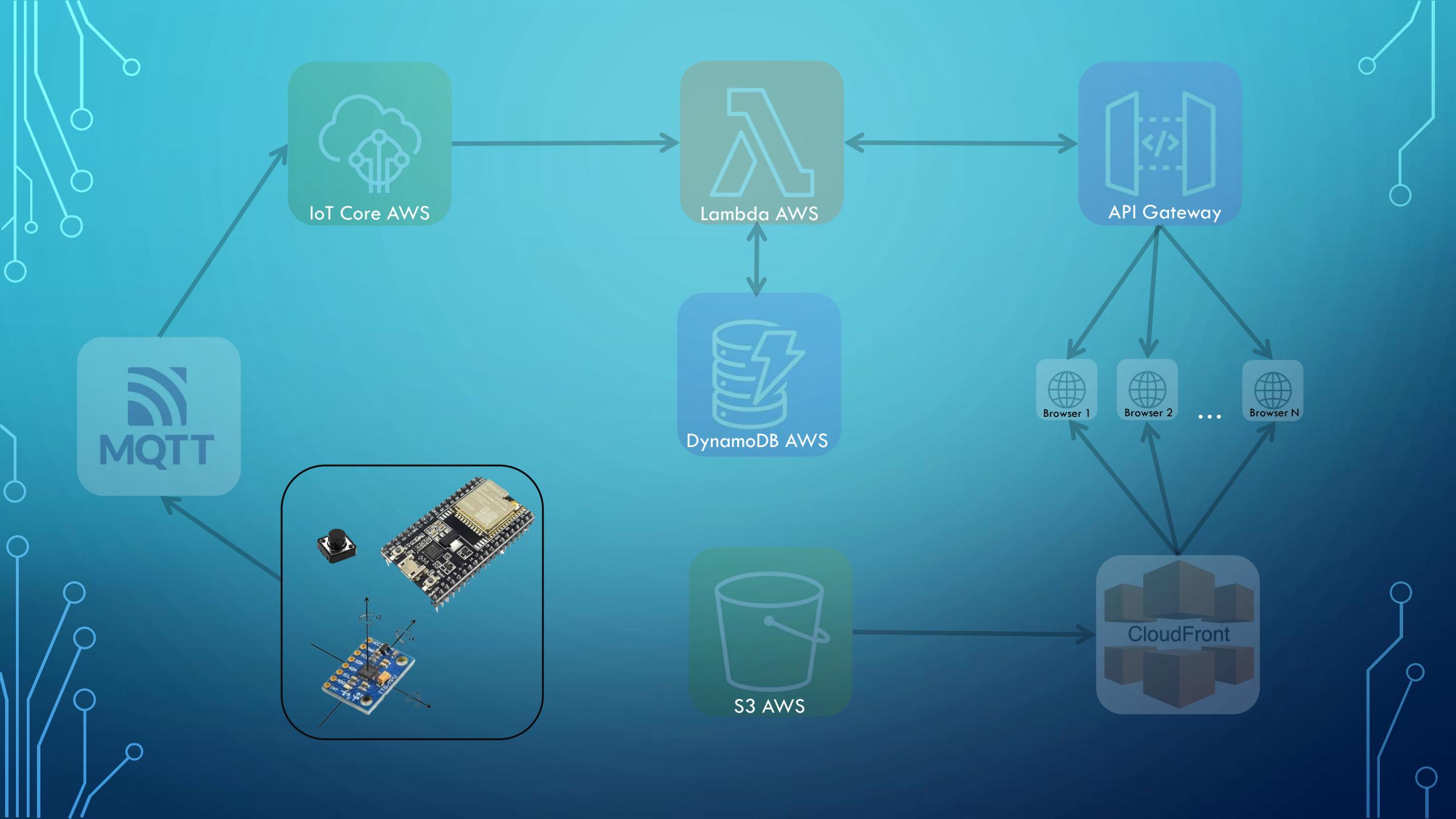
# RELATION TO OUR PROJECT

- Uses the same MPU6050 motion sensor
- Real-time system operation
- Provides an interactive user experience









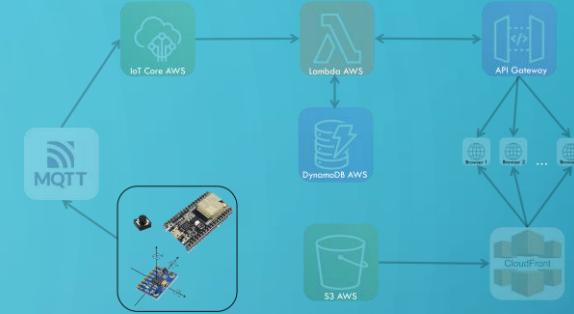
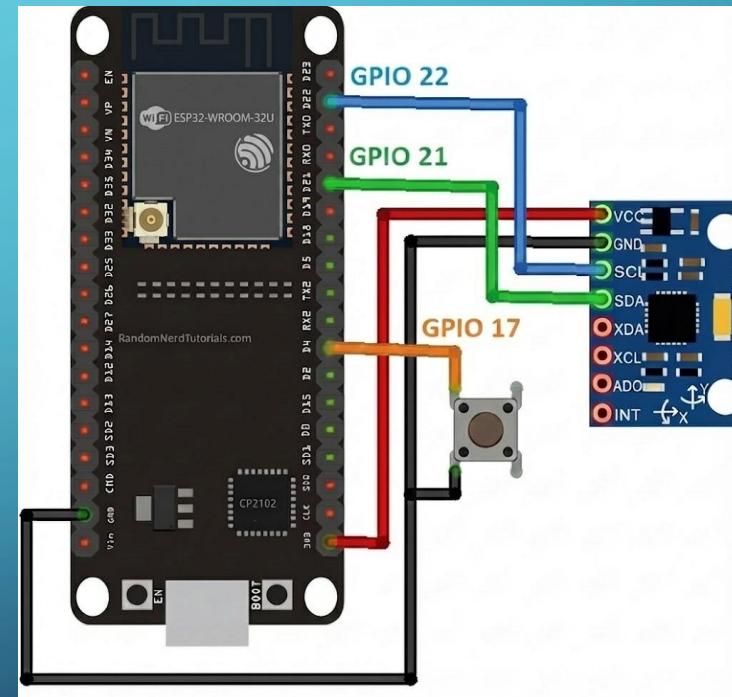
# Hardware Connections

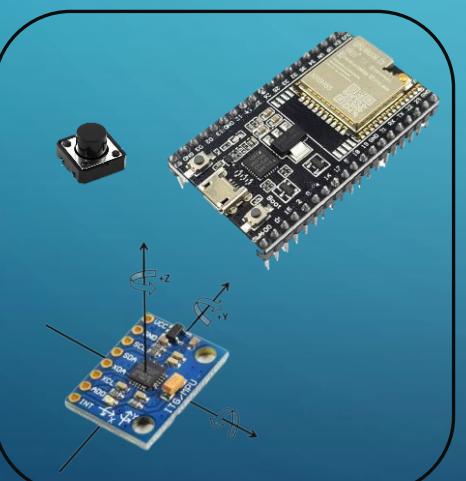
ESP32 acts as the central controller

MPU6050 provides motion and orientation data

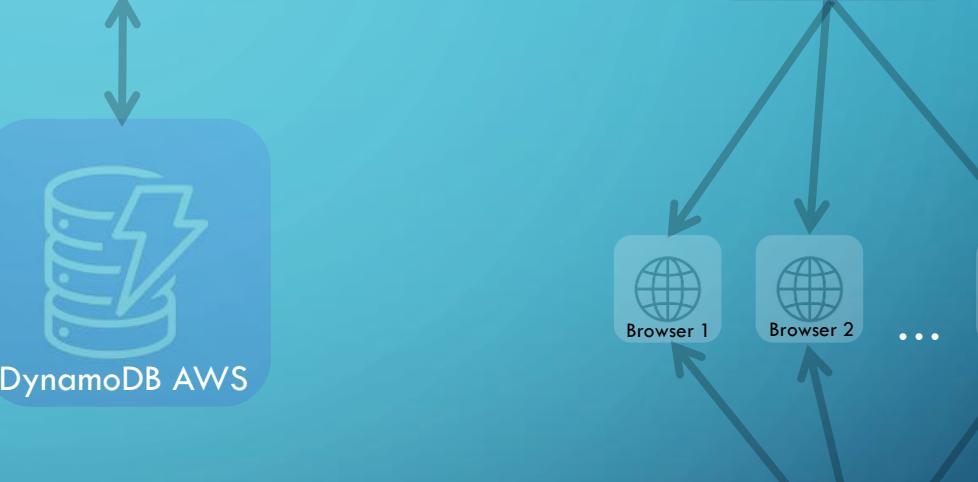
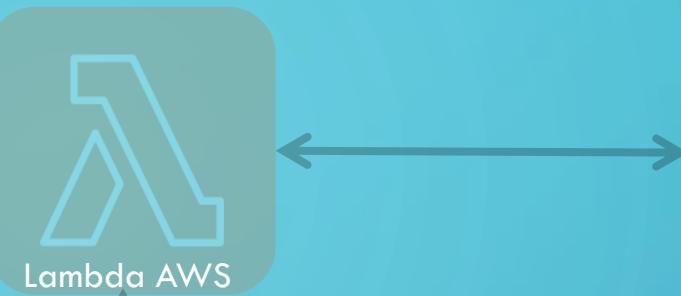
Button enables user interaction (start / stop action)

Simple and minimal wiring ensures reliable operation





...

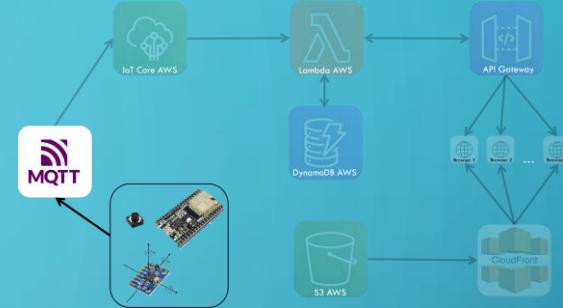


# MQTT Connection

The ESP32 publishes messages to MQTT topics

Different topics are used for sensor data (50Hz) and control events

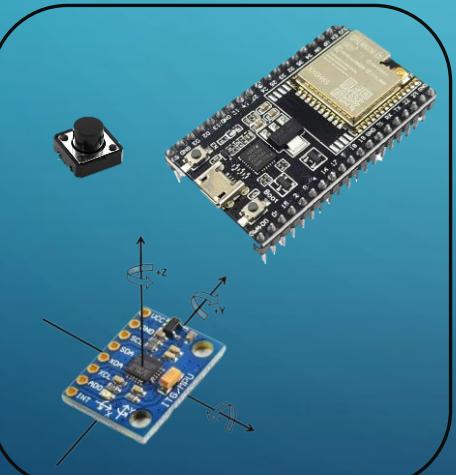
Messages are sent asynchronously to the cloud



```
WiFi connected
IP: 172.23.48.217
RSSI: -58
Syncing time..
Time synced
UTC time: 2026-01-11 16:23:59
TLS certificates loaded
Testing TLS TCP connect...
TLS TCP connect OK
Connecting to MQTT...
MQTT connected
publish_event -> OK
pressed=0
pressed=0
pressed=0
```



IoT Core AWS



Lambda AWS



DynamoDB AWS



API Gateway



Browser 1



Browser 2



Browser N



S3 AWS



CloudFront

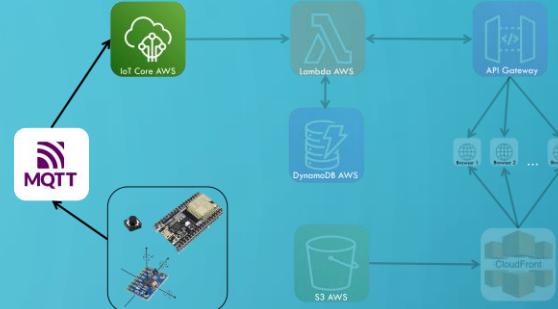
# IoT Core

Acts as the **central entry point** for device data

Receives MQTT messages from the ESP32

Authenticates devices and manages secure communication

Routes messages to cloud services using rules



**Subscriptions**

wand/event	X
wand/raw	X

**wand/raw**

**Message payload**

```
{ "message": "Hello from AWS IoT console" }
```

**► Additional configuration**

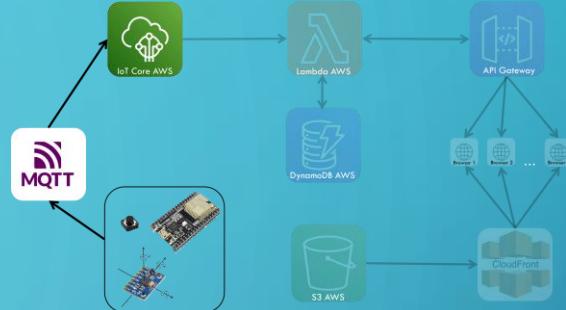
**Publish**

**wand/raw**

```
{ "ts_ms": 24014, "seq": 16, "imu": { "ax": 0.939, "ay": -0.059, "az": 0.411, "gx": 0.420, "gy": 5.382, "gz": 1.550 } }
```

**► Properties**

# Roles

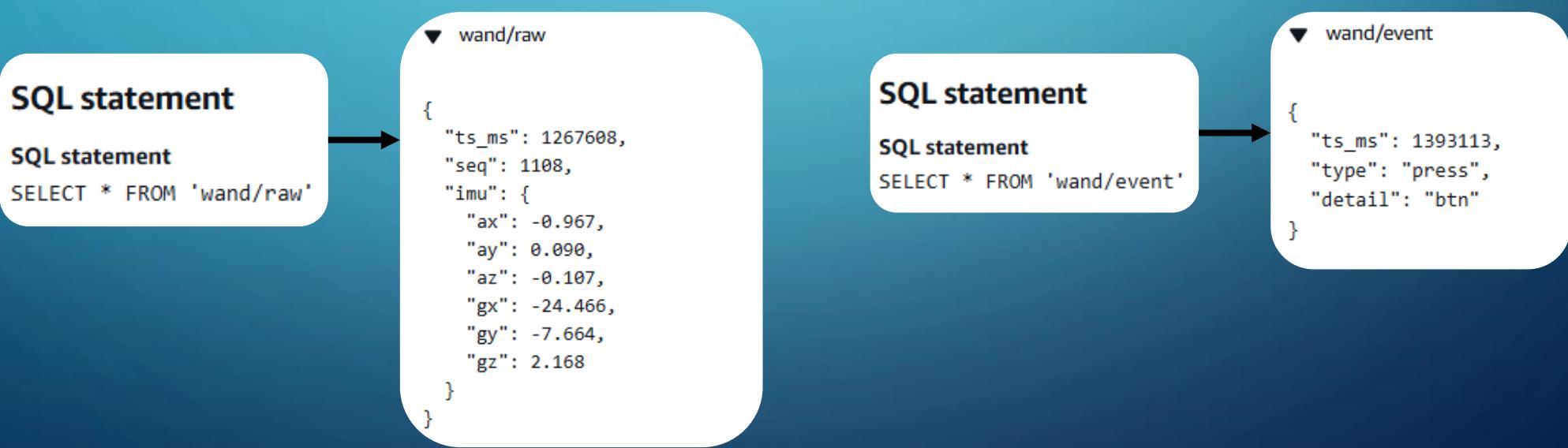


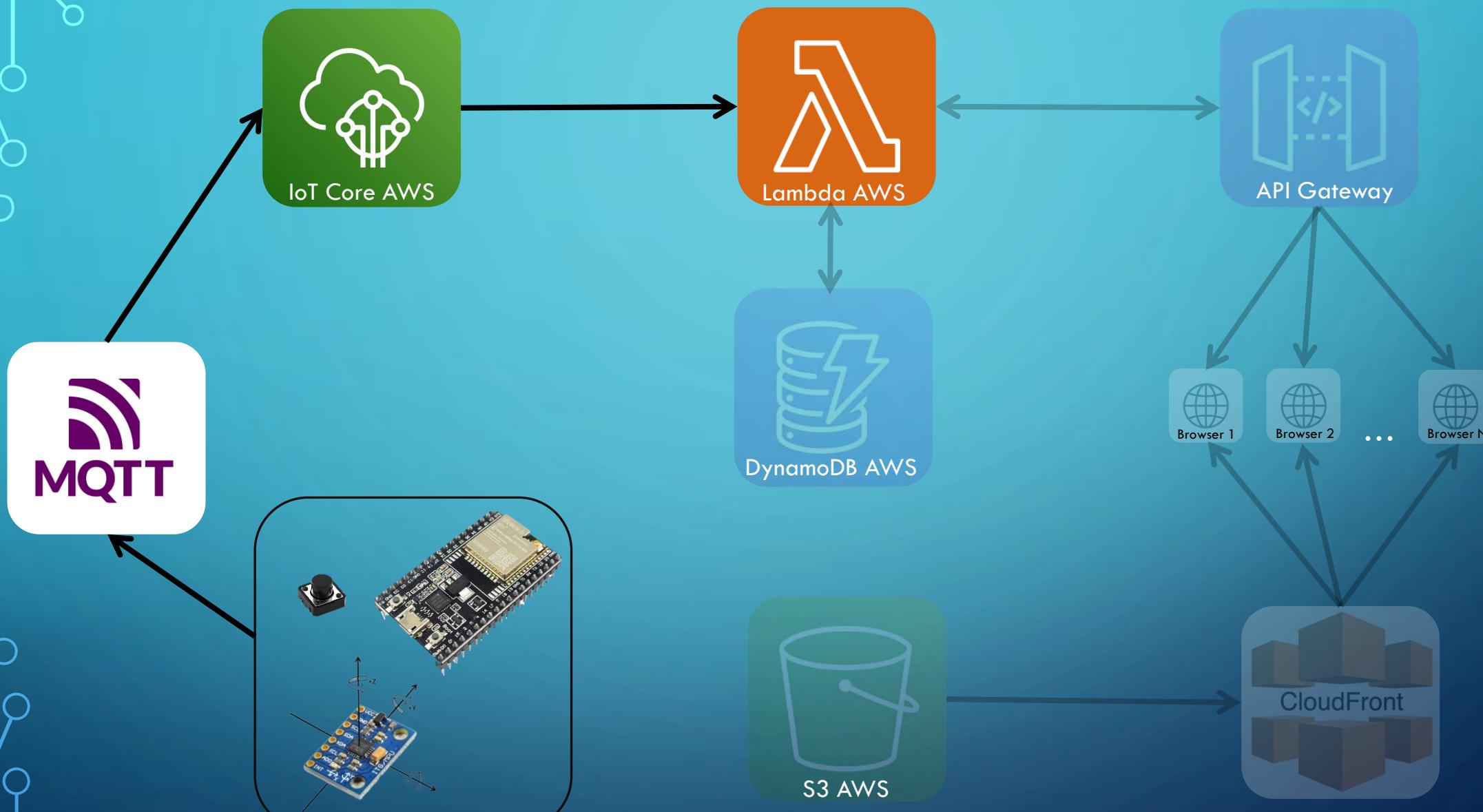
Rules (4) [Info](#)

Rules allow your things to interact with other services. Rules are analyzed and perform specific actions based on messages published by your devices.

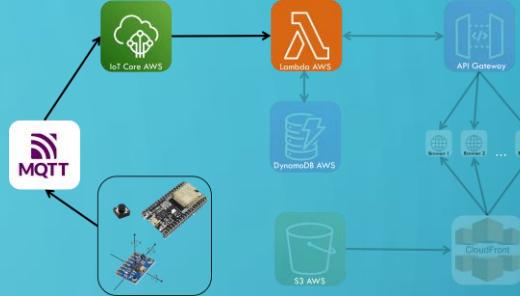
<input type="checkbox"/>	Name	Status	Rule topic	Created date
<input type="checkbox"/>	wand_raw_to_ws	<span>Active</span>	wand/raw	January 09, 2026, 12:41:08 (UTC+02:00)
<input type="checkbox"/>	wand_event_to_ws	<span>Active</span>	wand/event	January 09, 2026, 12:36:49 (UTC+02:00)

[Find rules](#) [Activate](#) [Deactivate](#) [Edit](#) [Delete](#) [Create rule](#)





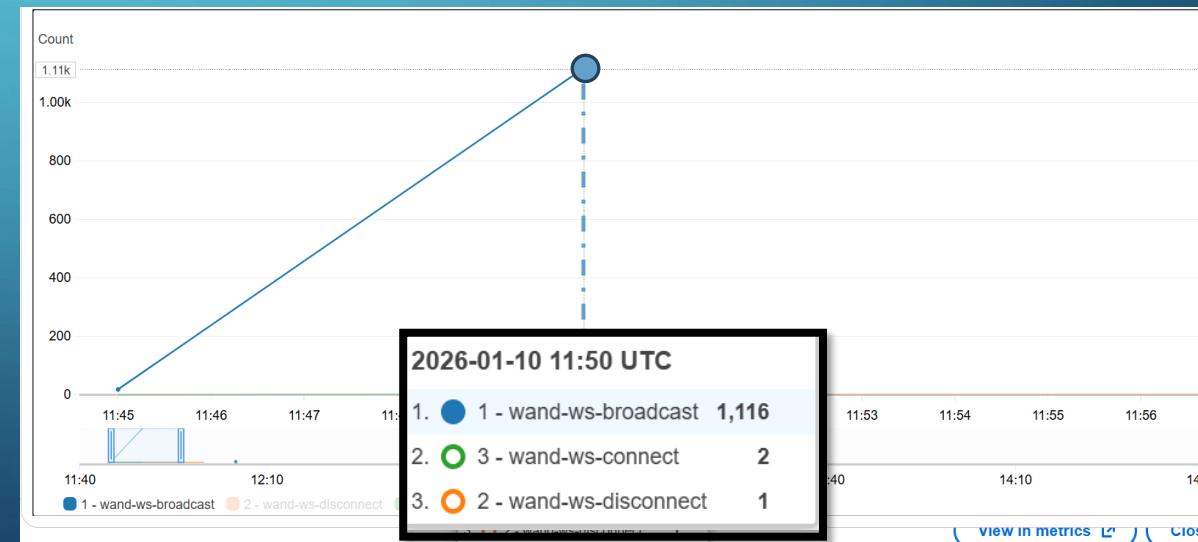
# Roles to Lambda

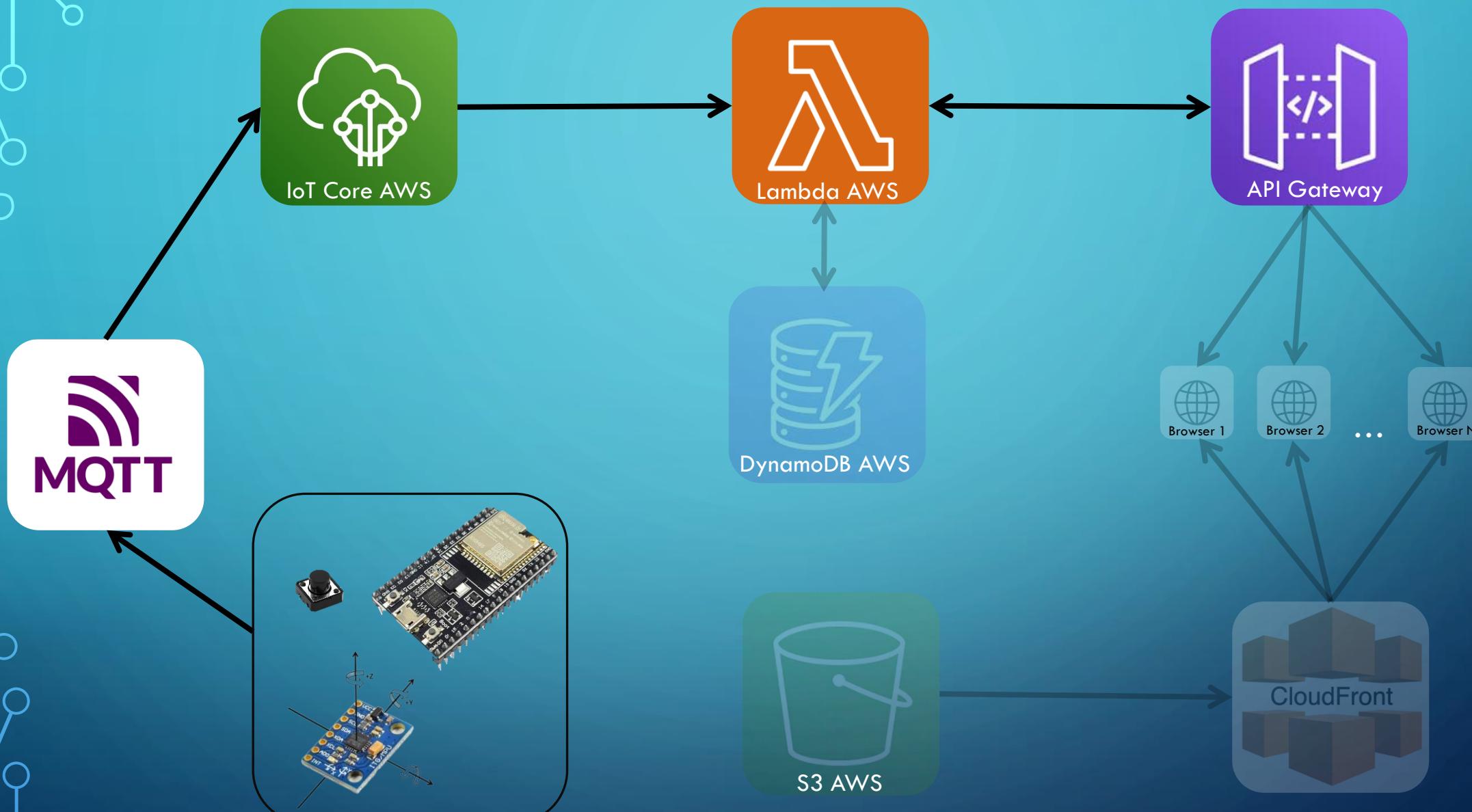


**Functions (4)**

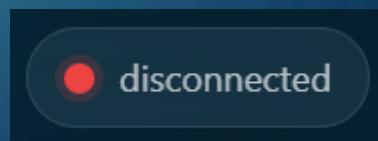
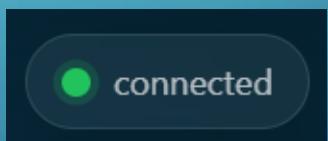
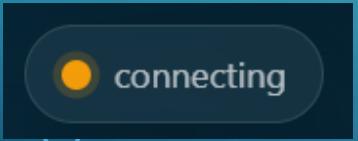
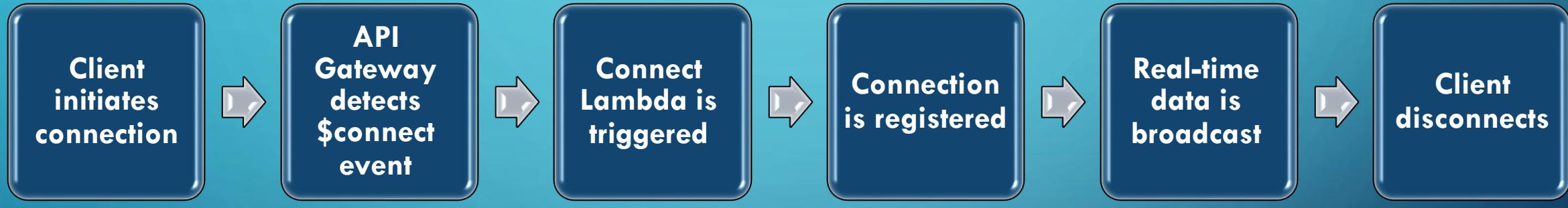
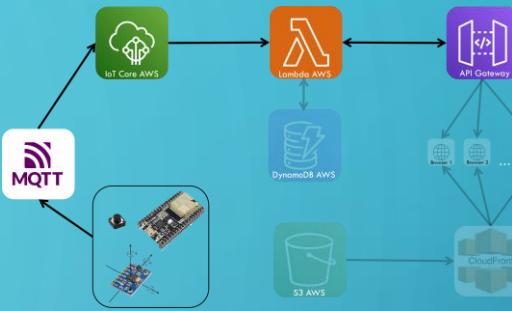
Last fetched 10.1.2026, 16:29:58 [Actions](#) [Create function](#)

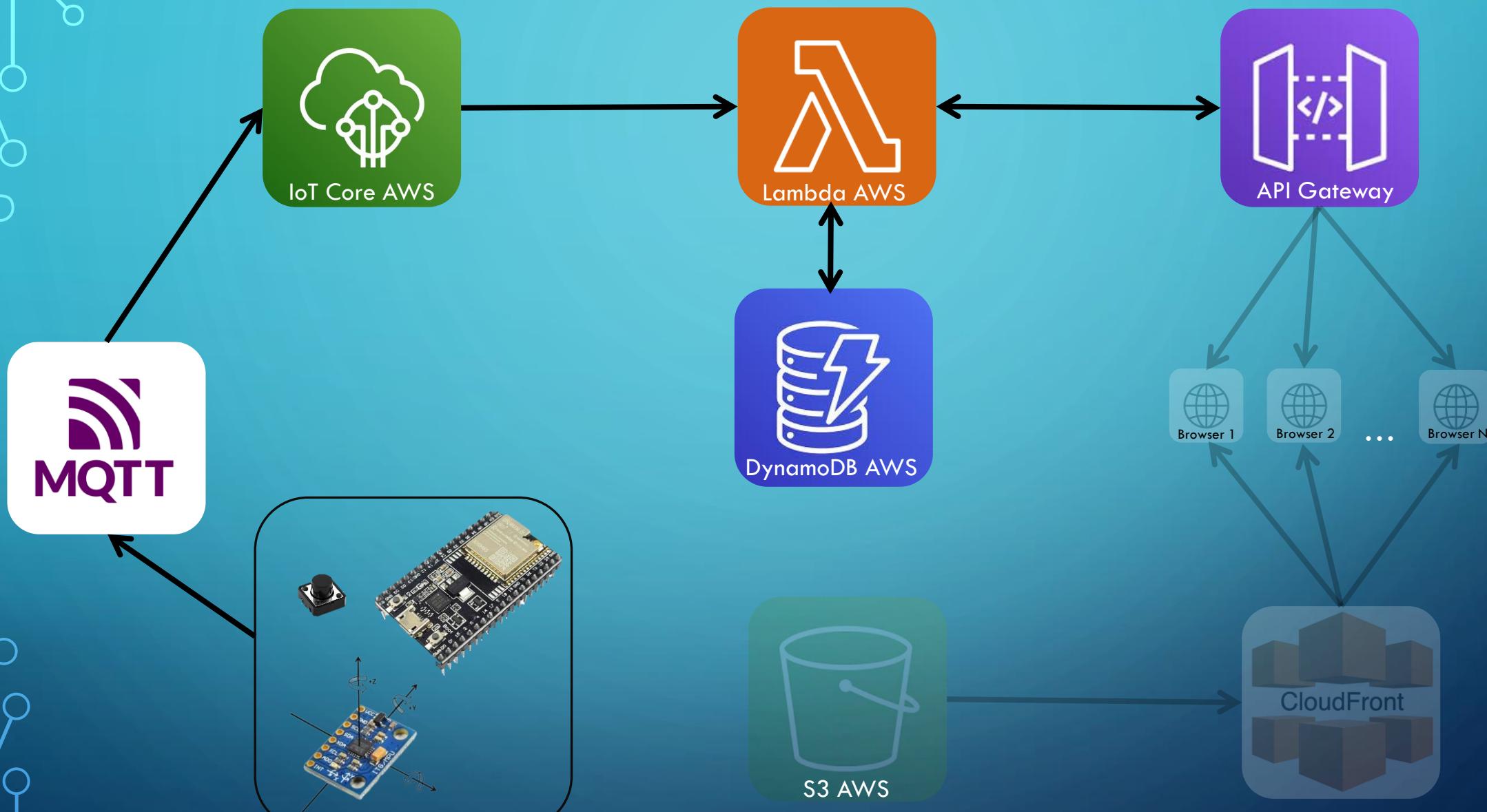
<input type="checkbox"/> Function name	Description	Package type	Runtime	Type	Last modified
<a href="#">wand-ws-broadcast</a>	-	Zip	Python 3.11	Standard	1 day ago
<a href="#">wand-ws-disconnect</a>	-	Zip	Python 3.11	Standard	1 day ago
<a href="#">wand-ws-connect</a>	-	Zip	Python 3.11	Standard	1 day ago





# API Gateway





# DynamoDB

Stores active WebSocket connections

Each item represents a connected client

Used by Lambda to manage real-time broadcasts

Automatically removes inactive connections (TTL)

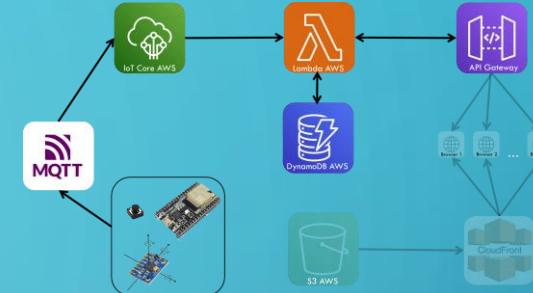
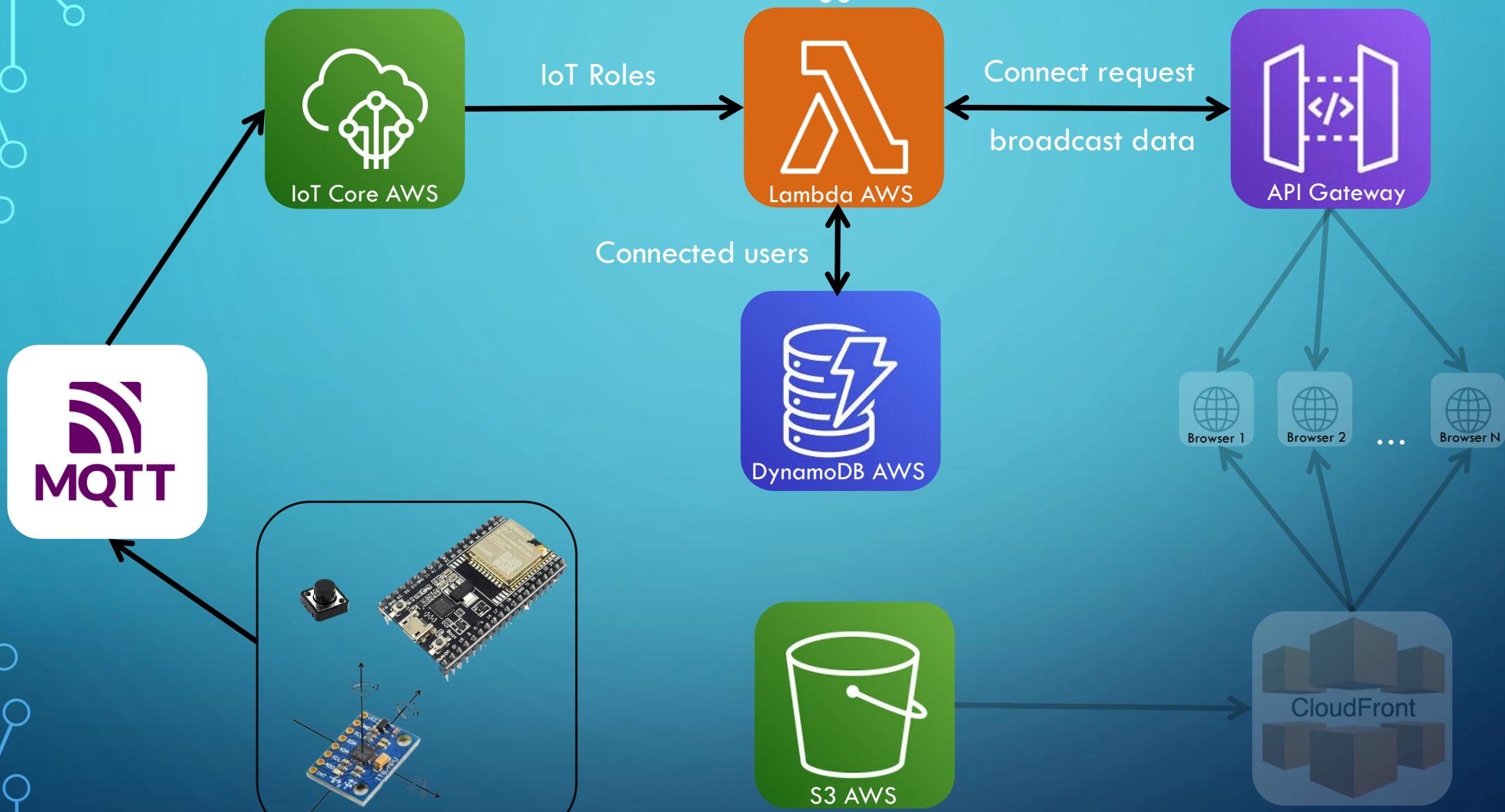


Table: wand-ws-connections - Items returned (2)

Scan started on January 10, 2026, 17:37:26

<input type="checkbox"/> connectionId (String)	connectedAt	ttl (TTL)
<a href="#">W-bGMebmoAMCFLA=</a>	1768059431	1768063031
<a href="#">W-bIEf7RoAMCEkg=</a>	1768059443	1768063043



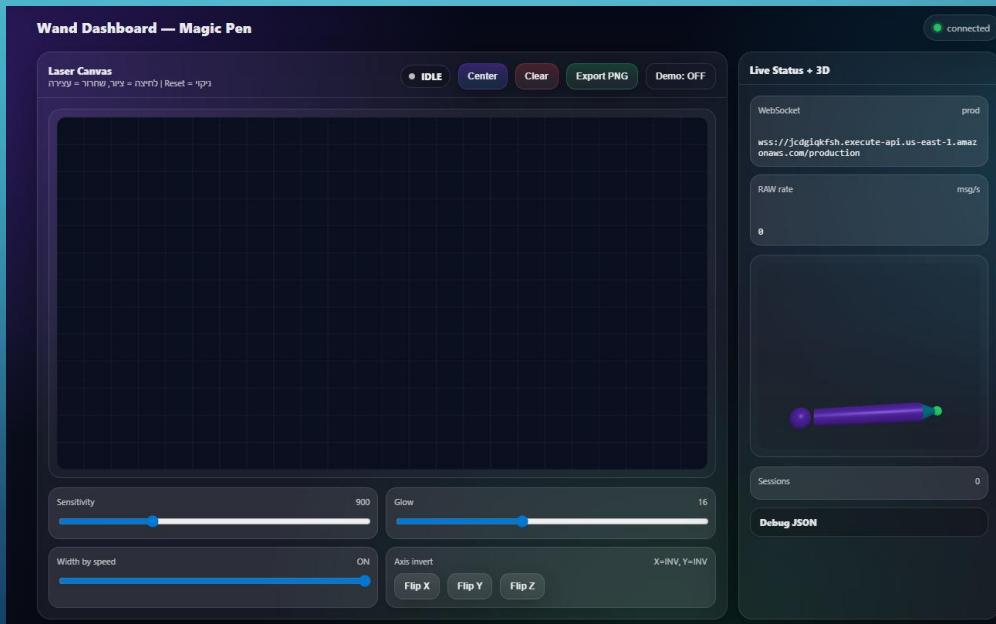
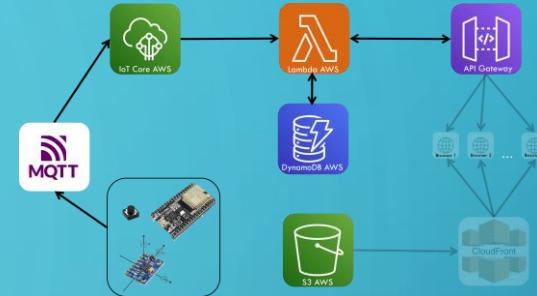
# S3

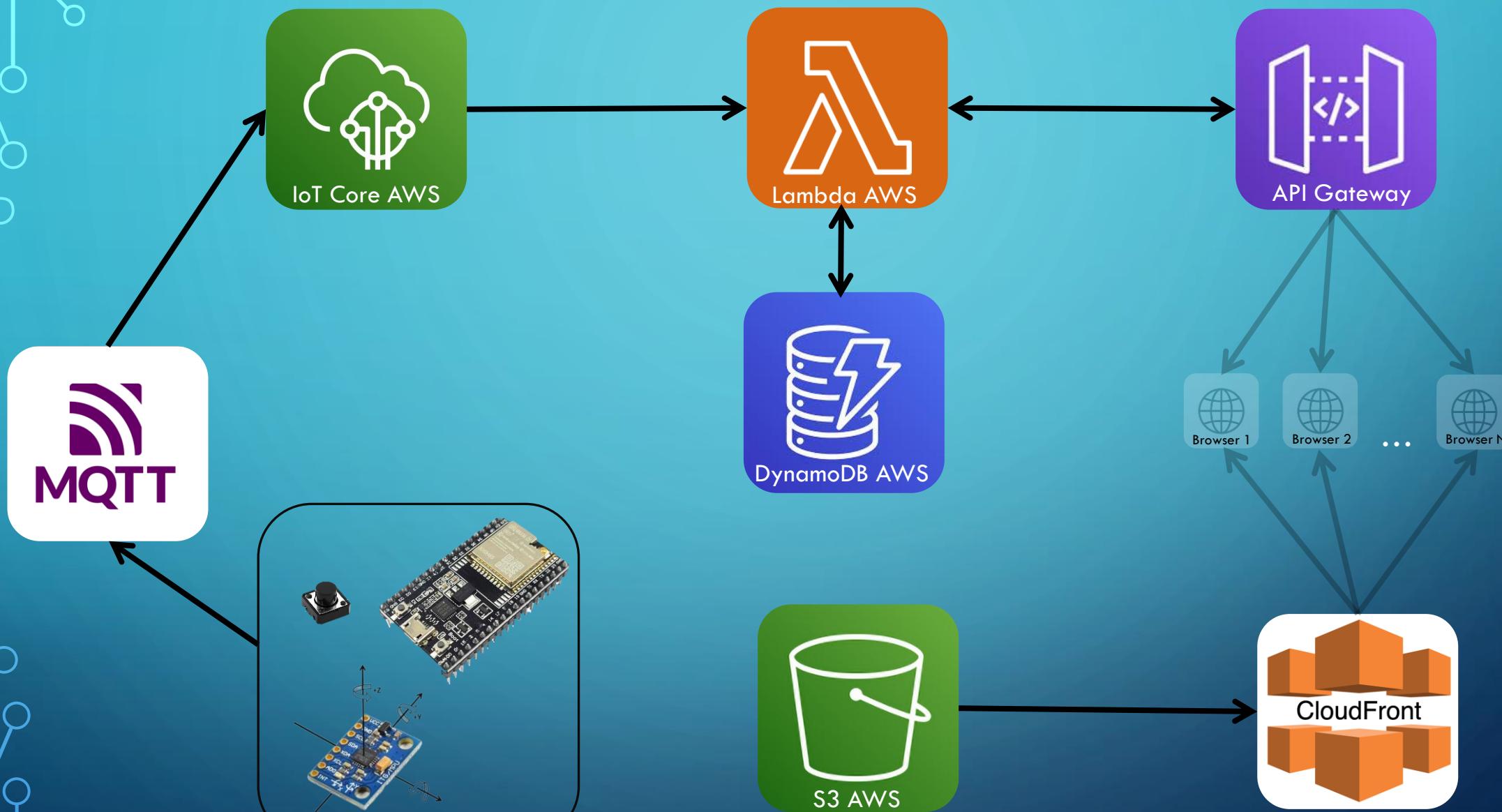
Hosts the static web application

Stores HTML File

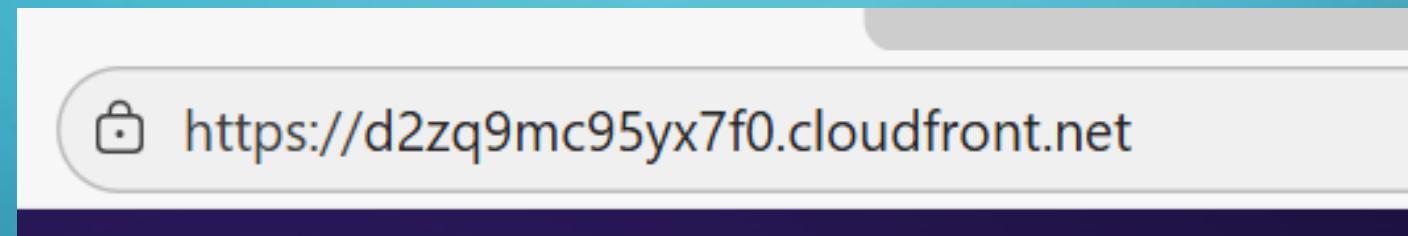
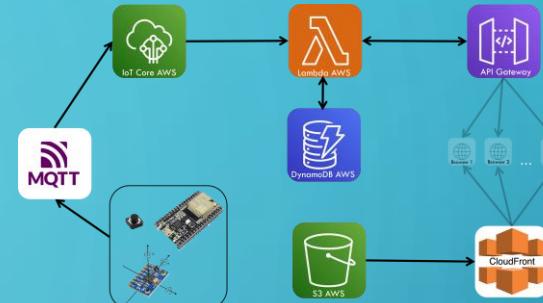
Serves the user interface to browser clients

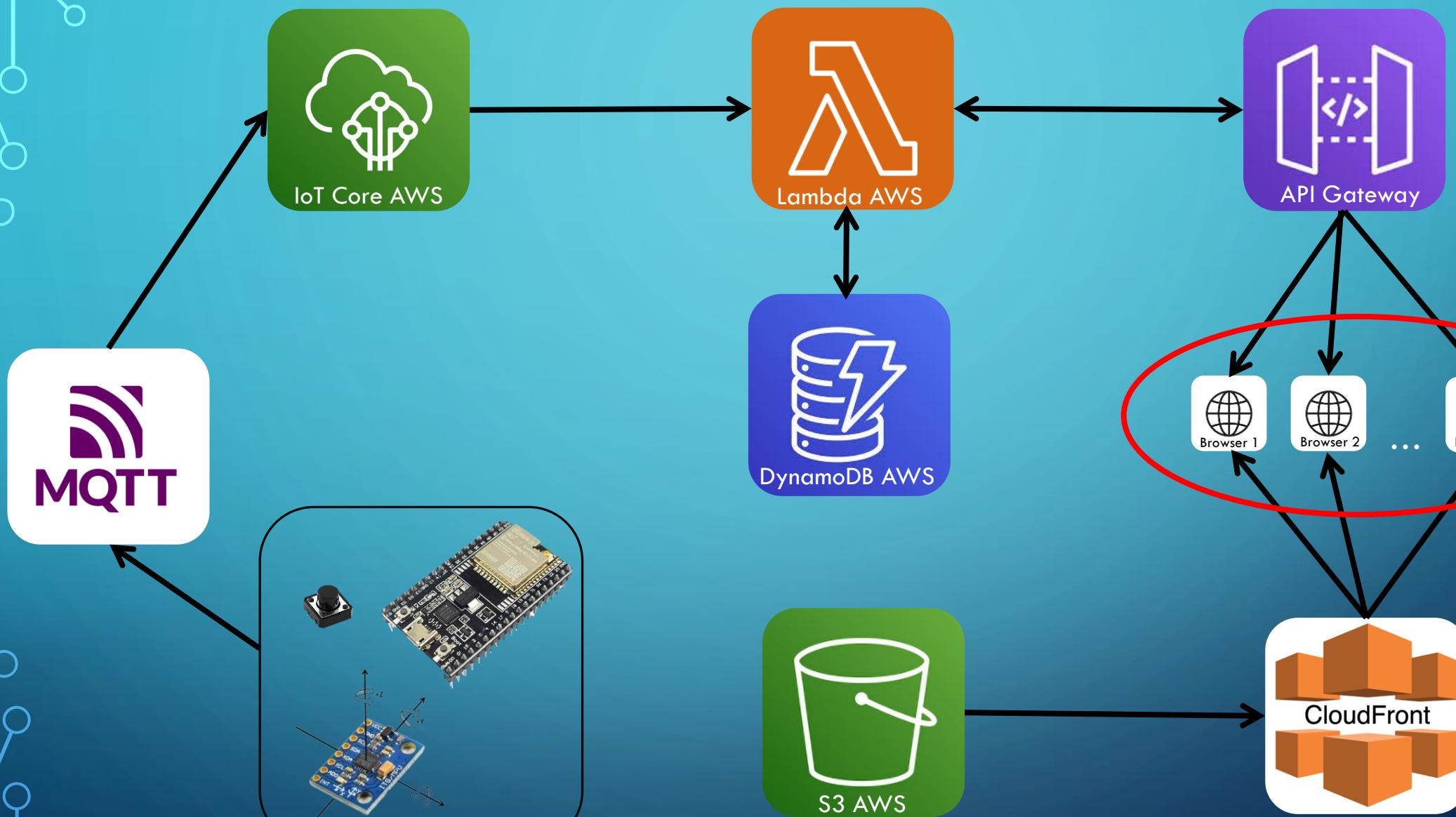
Integrated with CloudFront for content delivery





# CloudFront





# Features

### Wand Dashboard — Magic Pen

connected

Laser Canvas

נקי | לוחיצה = צייר, שחרור = עזיבה | Reset = ניקי

IDLE Center Clear Export PNG Demo: OFF

Sensitivity: 900

Glow: 16

Width by speed: ON

Axis invert: X=INV, Y=INV

Flip X Flip Y Flip Z

### Live Status + 3D

WebSocket prod  
wss://jcdgiqkfsh.execute-api.us-east-1.amazonaws.com/production

RAW rate msg/s  
0

Sessions: 0

Debug JSON

<https://d2zq9mc95yx7f0.cloudfront.net/>

## Limitations and challenges

Limited cloud  
resources  
(free-tier account)

Sending data in  
50 Hz for short  
time

Network stability

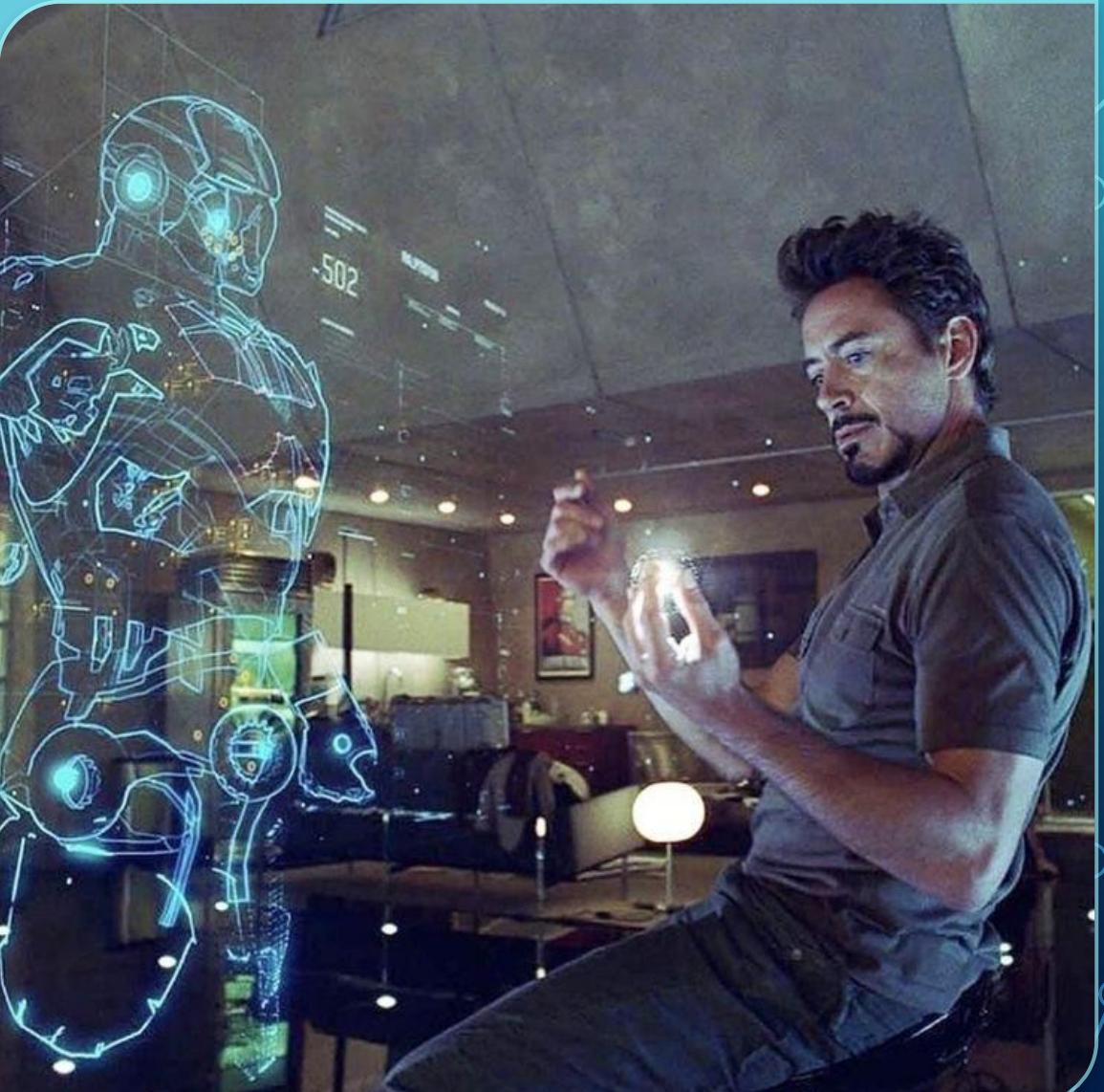
# Future Improvements & Uses

## Future Improvements:

- Store and analyze historical sensor data with ML
- Sending 2Hz data all the time
- Improve data processing and filtering in the cloud
- 3D visualization

## Future Uses:

- VR
- Education
- Training
- Smart environments



### Wand Dashboard — Magic Pen

connected

Laser Canvas  
RTVS9 = 111111, RDX = 12345 | Reset = 456

IDLE Center Clear Export PNG Demo: OFF

# END

Sensitivity: 900 | Glow: 16

Width by speed: ON | Axis invert: X-INV, Y-INV

Flip X | Flip Y | Flip Z

### Line States + 3D

WebSocket prod  
ws://jcdgiokfsh.execute-api.us-east-1.amazonaws.com/production

RAW rate msg/s  
0



Sessions: #6a6f27 787 15752 | 1182  
Replay | Export JSON | Delete

Debug JSON