



**Universidad Nacional Mayor de San Marcos**  
**Universidad del Perú.**  
**Decana de América**

Facultad de Ingeniería de Sistemas e Informática

## **Proyecto de IoT**

### **Internet de las Cosas**

#### **Integrantes**

Albert Williams Perez Santiago  
Miguel Stalin Soria Villanueva  
Erly Toribio Rivera Inche  
Nicolas Alonso Rojas Gala

LIMA, PERÚ  
2024

## **Práctica 1: Análisis de un caso de uso real de IoT**

## Descripción corta:

El proyecto "Comedero para Mascotas basado en IoT controlado por el Asistente de Google mediante NodeMCU" tiene como objetivo desarrollar un sistema automatizado para alimentar mascotas de manera remota y programada. Este sistema utiliza la plataforma NodeMCU para conectar el comedero a Internet y permite su control mediante comandos de voz a través del Asistente de Google.

### Funcionalidades

- Control por Voz: Habilitación del control del comedero mediante comandos de voz a través del Asistente de Google, permitiendo la alimentación de la mascota desde cualquier lugar.
- Programación de Alimentación: Configuración de horarios de alimentación automáticos para dispensar comida a intervalos regulares.
- Reconocimiento de voz a través del asistente de Google mediante el uso de "IFTT" para detectar al momento que se dice "Okay Google, alimentar perro" se mande una señal a ADAFRUIT para que puede ser reconocido por la placa y poder liberar el alimento para la mascota

## My Applets

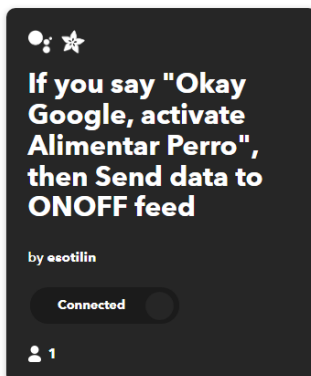
🔍 Filter

All (1 of 20)

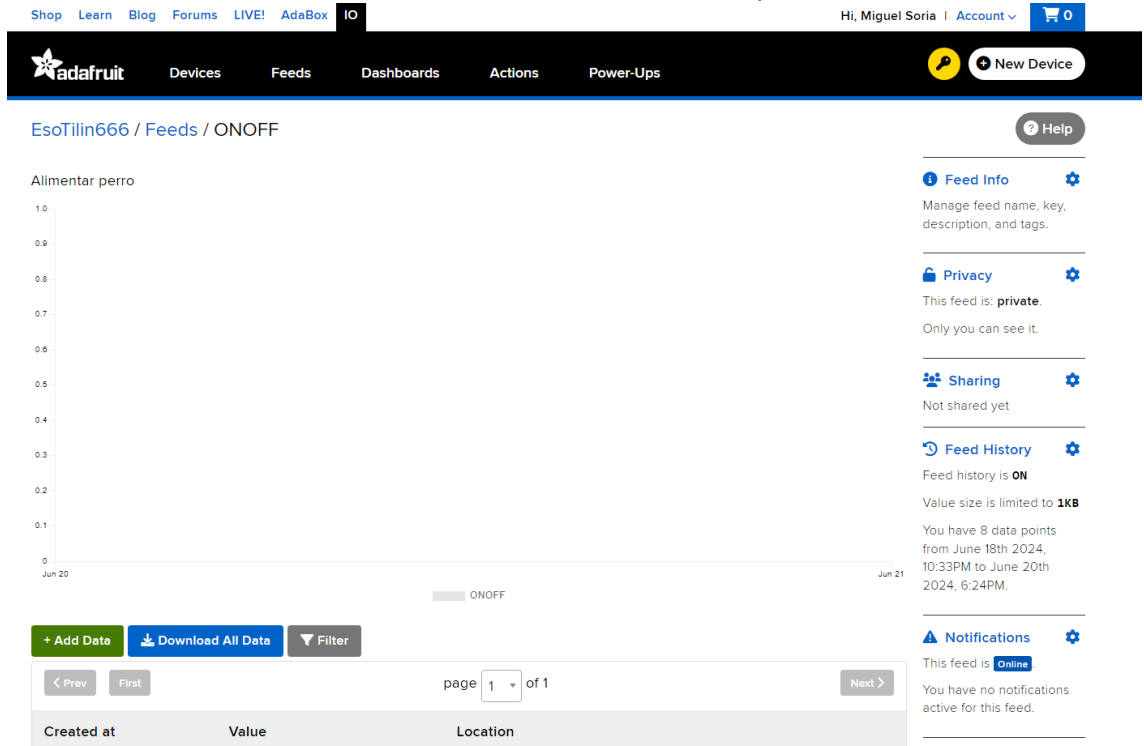
Published

Archive

[Get Pro+ for unlimited Applets](#)



- Usamos **ADAFRUIT** para una vez el asistente de google recibe el orden este le manda la una señal en este caso “ON” que va a ser almacena como un dato que sera almacenado y este asu vez envia el dato de “ON” para que activarlo y se puede alimentar a la mascota asi mismo nos brinda un dasboard donde se muestran las veces que fueron activadas



## ¿Qué dispositivos nos pueden ayudar para nuestro proyecto?

### Componentes del Sistema

- **NodoMCU ESP8266:** Es el microcontrolador principal que proporcionará conectividad WiFi para controlar el comedero a través de internet.
- **Módulo LCD 16x2:** Permite visualizar información en tiempo real
- **Módulo LCD I2C:** Facilita la conexión del módulo LCD 16x2 al NodeMCU, reduciendo la cantidad de pines necesarios para la comunicación.
- **Servo motor:** Controla el mecanismo de dispensado del alimento, abriendo y cerrando el compartimento donde se almacena la comida.

### Código del proyecto(Avance):

```
#include <Servo.h>
```

```
// Crear un objeto Servo para controlar el servomotor
```

```
Servo myservo;
```

```
// Definir el pin del botón y el pin del servomotor
```

```
const int buttonPin = 2;

const int servoPin = 9;


// Variable para almacenar el estado del botón
int buttonState = 0;


// Posición inicial del servomotor
int servoPos = 0;


void setup() {
    // Iniciar el servo en el pin especificado
    myservo.attach(servoPin);

    // Configurar el pin del botón como entrada con resistencia pull-up interna
    pinMode(buttonPin, INPUT_PULLUP);

    // Configurar la posición inicial del servomotor
    myservo.write(servoPos);
}


void loop() {
    // Leer el estado del botón
    buttonState = digitalRead(buttonPin);

    // Si el botón está presionado (estado bajo porque usamos pull-up)
    if (buttonState == LOW) {
        // Mover el servomotor a 180 grados
        myservo.write(180);

        // Esperar un segundo
        delay(1000);

        // Volver el servomotor a 0 grados
```

```
myservo.write(0);
```

```
}
```

```
}
```