

Entrega Fase 3 – Feed Tech Solution.

Ramírez Vargas, Lina María^{#1} González Rozo, David Esteban^{#2} Torres Mesa, Francy Yuliana^{#3}

*Ingeniería mecatrónica, Pontificia Universidad Javeriana
Bogotá, Colombia*

¹lina.ramirezv@javeriana.edu.co ²desteban.gonzalezr@javeriana.edu.co ³torresmfrancy@javeriana.edu.co

Se realizaron ajustes iniciales en el código para permitir la transmisión de datos a ThingsBoard. El primer cambio se centró en la conexión, que se dividió en dos partes: el host y el token del dispositivo. Inicialmente, el dominio era 10.195.28.12; sin embargo, se observó que este dominio solo era funcional con el internet de la universidad, lo que impedía la conexión con la Raspberry. Por lo tanto, se creó una nueva cuenta en ThingsBoard, a partir de la cual se desarrolló toda la interfaz de usuario.

```
# Configuración de ThingsBoard
THINGSBOARD_HOST = 'demo.thingsboard.io'
ACCESS_TOKEN = 'yIJfWlkEsf9AKKdWqt72'

# Configuración de pines GPIO para LEDs RGB
LED_RED = 17
LED_GREEN = 27
LED_BLUE = 22
```

Además, se incorporó un LED RGB para indicar cuando el sensor RFID RC522 identificaba un tag. Cuando la ternera estaba registrada en la base de datos, el LED mostraba luz verde; en caso contrario, se encendía en rojo. Por lo que no se está utilizando el canal azul del Led.

```
# Función para enviar datos a ThingsBoard
def enviar_datos_thingsboard(data):
    client.publish('v1/devices/me/telemetry', json.dumps(data), 1)
    print(f'Datos enviados a ThingsBoard: {data}')

# Función para encender el LED verde
def encender_led_verde():
    GPIO.output(LED_GREEN, GPIO.HIGH)
    GPIO.output(LED_RED, GPIO.LOW)
    GPIO.output(LED_BLUE, GPIO.LOW)

# Función para encender el LED rojo
def encender_led_rojo():
    GPIO.output(LED_GREEN, GPIO.LOW)
    GPIO.output(LED_RED, GPIO.HIGH)
    GPIO.output(LED_BLUE, GPIO.LOW)
```

El actuador utilizado fue un servomotor, encargado de abrir o detener el flujo de alimento para las terneras. Su funcionamiento se divide en dos fases: primero, cuando un tag se acerca al sensor de identificación, el servomotor se activa y gira su palanca 180°, permitiendo el flujo de leche hacia la taza de las terneras a través de una manguera. Dado que la cantidad de alimento está estipulada

según la edad de las terneras, una vez que se alcanza el volumen esperado, el servomotor se activa nuevamente, regresando a su posición inicial para detener el flujo.

```
# Verificar si el volumen ha alcanzado o superado el parámetro
if volumen >= parametro_volumen:
    print("Volumen objetivo alcanzado:", parametro_volumen, "L")
    contador_activo = False # Pausar el contador

# Enviar datos a ThingsBoard
enviar_datos_thingsboard({'id_ternera': id_ternera, 'volumen': round(volumen, 3)})
encender_led_verde()

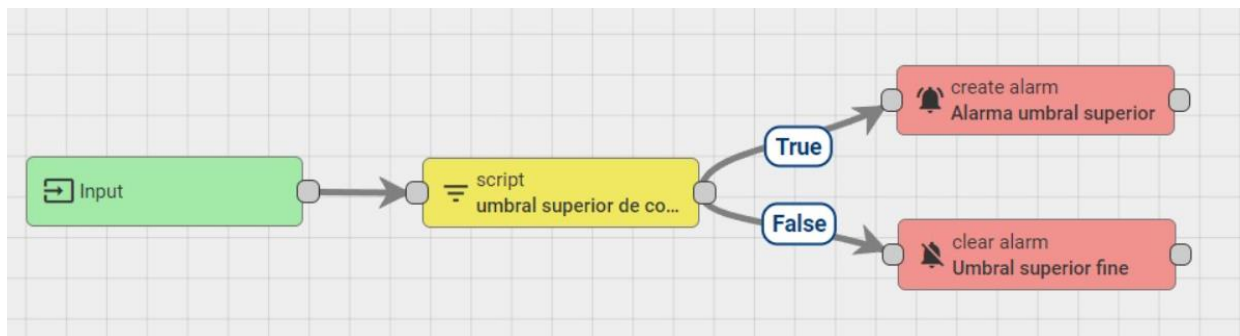
# Mover el servo 90 grados
servo.ChangeDutyCycle(7.5) # Cambia este valor según la posición neutral de tu servo
time.sleep(0.5) # Da tiempo para que el servo se mueva
servo.ChangeDutyCycle(12.5) # Gira a 90 grados (cambiar según tu servo)
time.sleep(5) # Mantener la posición durante 5 segundos

# Detener el servo
servo.stop()
```

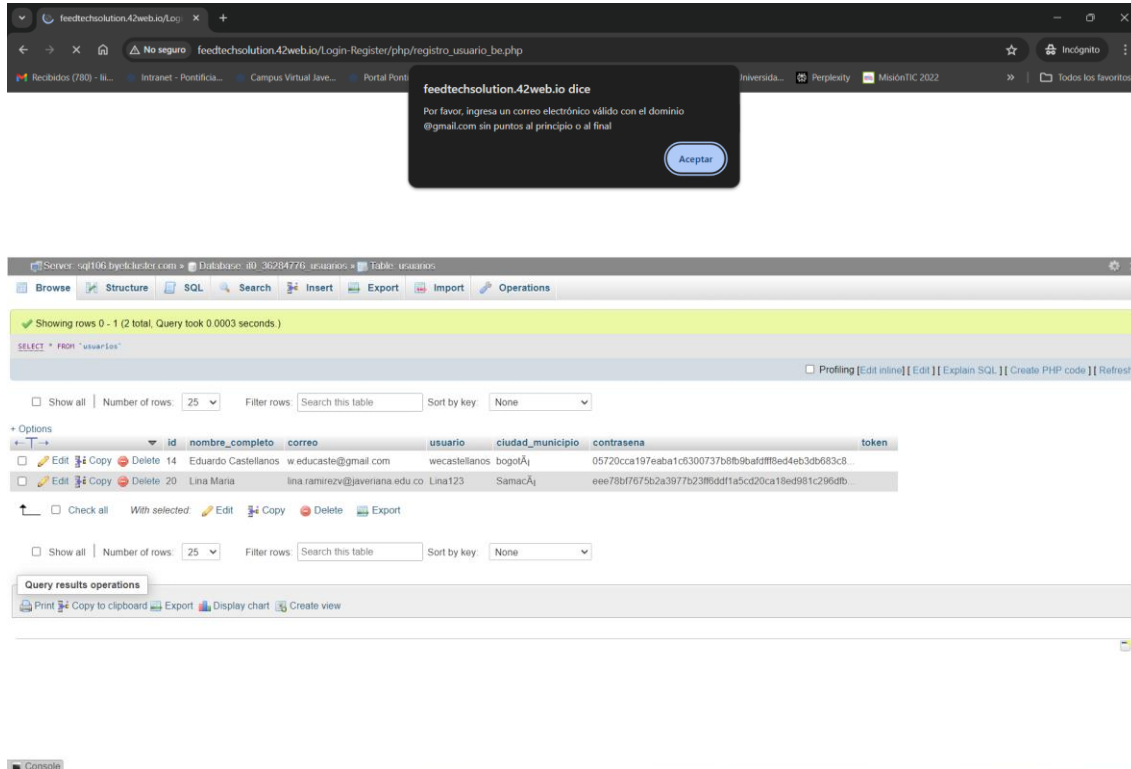
Para implementar las alertas en el dashboard, se crearon diagramas de flujo en ThingsBoard. En este caso, se configuró una alerta para el sensor de identificación. Esta alerta se activa cuando una ternera se registra más de dos veces al día.

El proceso comenzó con el diseño de un diagrama de flujo que define los criterios y las condiciones necesarias para generar la alerta. Si el sistema detecta que una ternera ha sido registrada más de dos veces en un día, se envía una notificación al panel de control.

Esta alerta es indispensable para asegurar que las terneras no reciban más alimento del que está establecido, manteniendo un control sobre su alimentación.



Otro cambio que se realizó, siguiendo la recomendación del profesor, fue la modificación del sistema de registro por correo electrónico. Ahora es posible registrarse utilizando cualquier dominio, como 'gmail.com', 'javeriana.edu.co', entre otros, lo cual era necesario, ya que anteriormente, el sistema de registro estaba restringido a 'gmail.com', lo cual limitaba la accesibilidad para algunos usuarios.

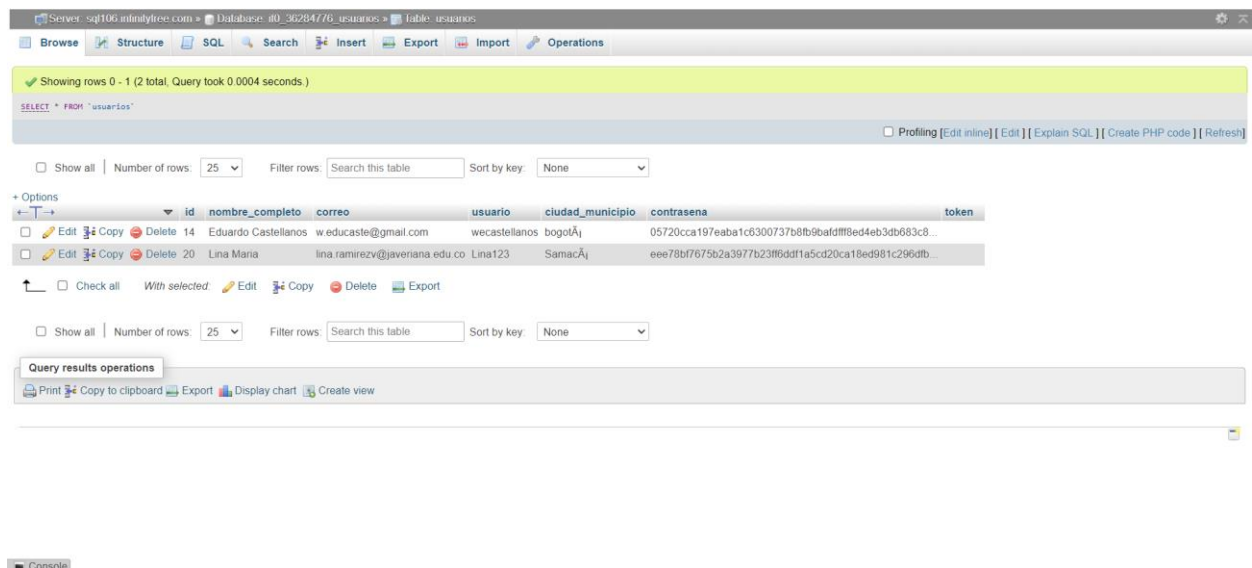


The screenshot shows a web browser window with a security warning from 'feedtechsolution.42web.io' asking to accept cookies. Below the warning, a database management interface (phpMyAdmin) is visible, displaying a table named 'usuarios' with the following data:

id	nombre_completo	correo	usuario	ciudad_municipio	contrasena	token
14	Eduardo Castellanos	w.educaste@gmail.com	wecastellanos	bogotá	05720cca197eaba1c6300737b8fb9bafdf8ed4eb3db683c8...	
20	Lina Maria	lina.ramirezv@javeriana.edu.co	Lina123	Samacá	eee78bf7675b2a3977b23f86ddf1a5cd20ca18ed981c296db...	

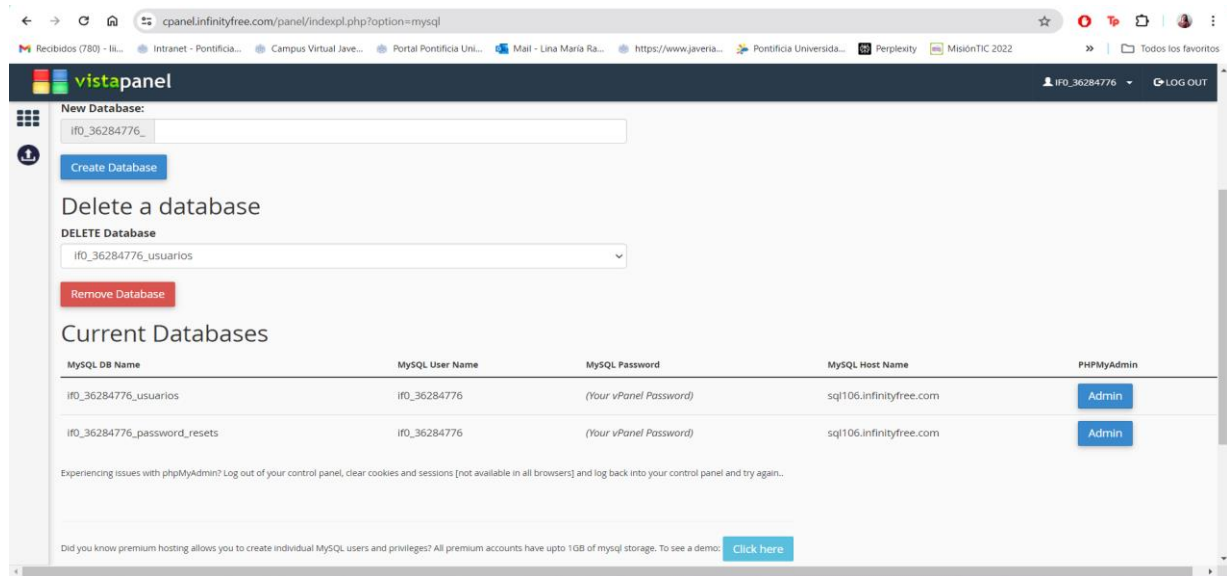
Además, se intentó implementar la función de 'Olvidó su contraseña'. Sin embargo, a pesar de numerosos intentos, no fue posible debido a las limitaciones del hosting gratuito, que restringe ciertas opciones, incluida la recuperación de contraseñas.

Se intentó resolver este problema utilizando el protocolo de comunicación SMTP para enviar correos electrónicos de recuperación de contraseña. Sin embargo, las restricciones del hosting gratuito impidieron completar esta configuración. Por lo tanto, esta funcionalidad no pudo ser implementada en la plataforma actual.



This screenshot shows the same database management interface as the previous one, displaying the 'usuarios' table. The data is identical to the previous screenshot:

id	nombre_completo	correo	usuario	ciudad_municipio	contrasena	token
14	Eduardo Castellanos	w.educaste@gmail.com	wecastellanos	bogotá	05720cca197eaba1c6300737b8fb9bafdf8ed4eb3db683c8...	
20	Lina Maria	lina.ramirezv@javeriana.edu.co	Lina123	Samacá	eee78bf7675b2a3977b23f86ddf1a5cd20ca18ed981c296db...	



Para la implementación del dashboard, se conectó el código unificado de los sensores utilizando el dominio y el token del dispositivo, como se mencionó anteriormente. Las primeras pruebas se realizaron con el sensor de identificación, utilizando la información almacenada en la base de datos local de cada tag, que incluía la edad y el ID único de cada ternera.

Posteriormente, se realizó una prueba enviando los datos del sensor de flujo. En este caso, se configuró una alerta para detectar si se recibe más leche de la esperada según la tabla de alimentación disponible en la página web del proyecto. Esta alerta indica un posible fallo en el actuador encargado de detener el flujo de la leche.

Además, se creó un gráfico que muestra el peso inicial de cada ternera y su evolución a lo largo del tiempo. Este gráfico permite monitorear el crecimiento de las terneras y ajustar su alimentación en consecuencia, asegurando un desarrollo saludable y adecuado para cada una.

