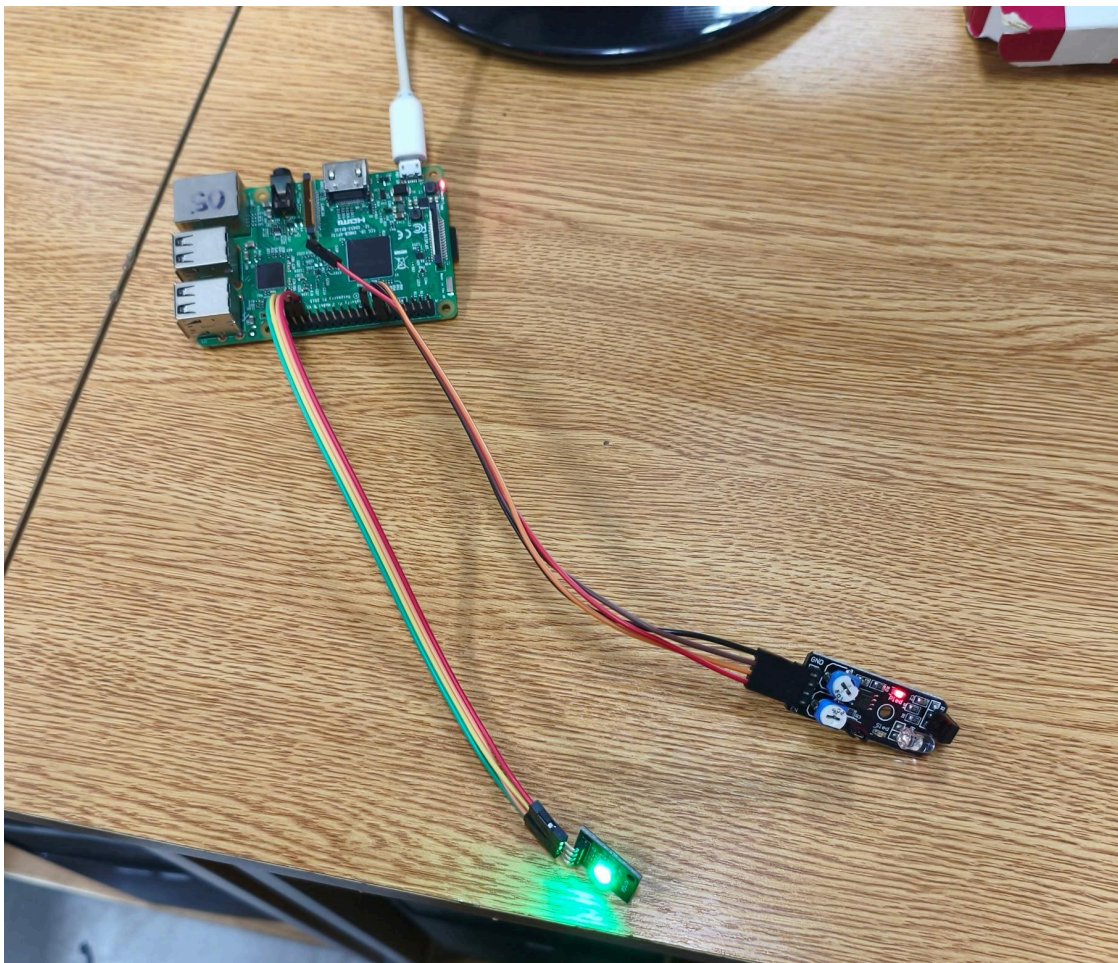


Cloud Computing y Cloud Robotics

Actividad 4 - Integración de Raspberry Pi con AWS IoT Core

Grupo: A17

Integrantes: Garcia, Nabila,
Luna, Leonardo,
Marcos, Santiago



Descripción del sistema: La funcionalidad que implementamos es, mediante un sensor de proximidad, poder dar alerta de cuando hay algún objeto que provocará una colisión cerca.

El LED conectado a la Raspberry Pi estará de color verde por defecto, al detectarse una colisión, este cambiará a rojo durante unos segundos para dar aviso que hay un objeto delante.

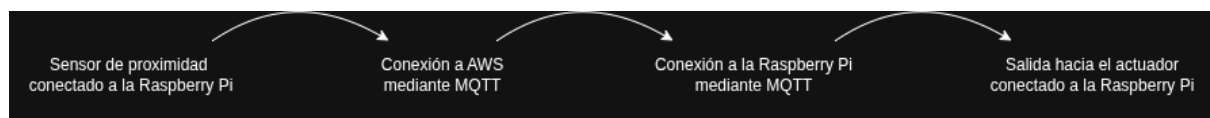
Las conexiones con AWS las realizamos mediante los nodos de MQTT que vimos en clases anteriores.

Todo lo que sea detectado por el sensor se procesa, en parte utilizando la función filter dada por la cátedra, y posteriormente es enviado a AWS mediante el tópic **RASPI-AWS**.

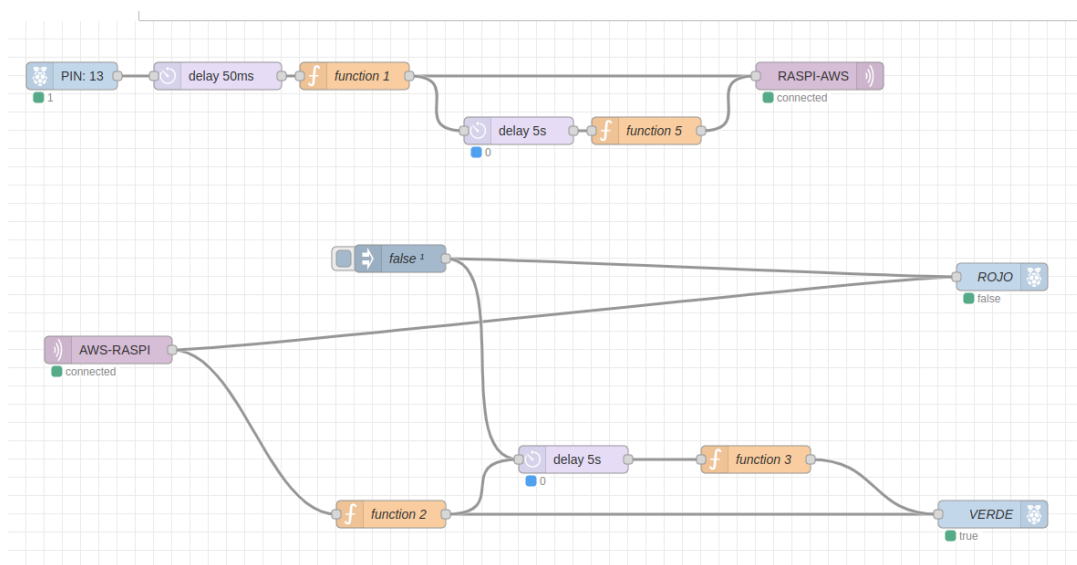
Una vez en Amazon, se determina qué se debe hacer y esa salida final es enviada de regreso a la Raspberry Pi mediante el tópico **AWS-RASPI** para poder encender el LED de color rojo.

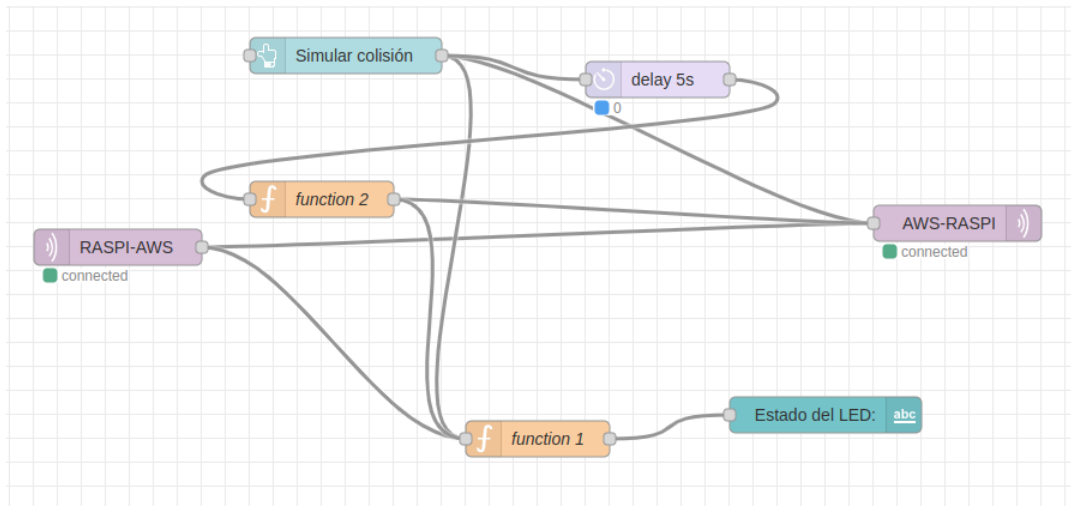
En caso de que no se necesite encender los LEDs no es necesario hacer nada ya que el sensor no envía la señal opuesta ante la falta de una detección de un objeto en la proximidad.

Diagrama de la arquitectura:



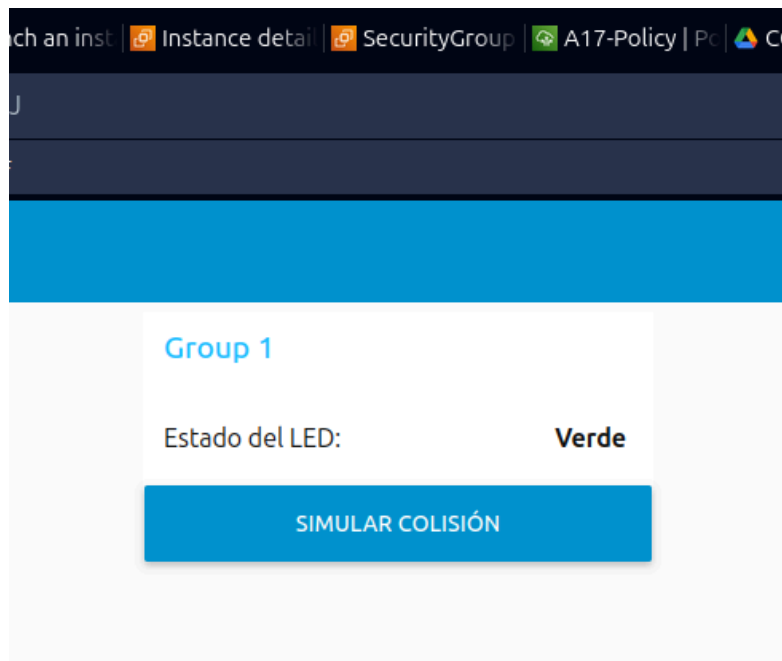
Flujos de Node-RED:

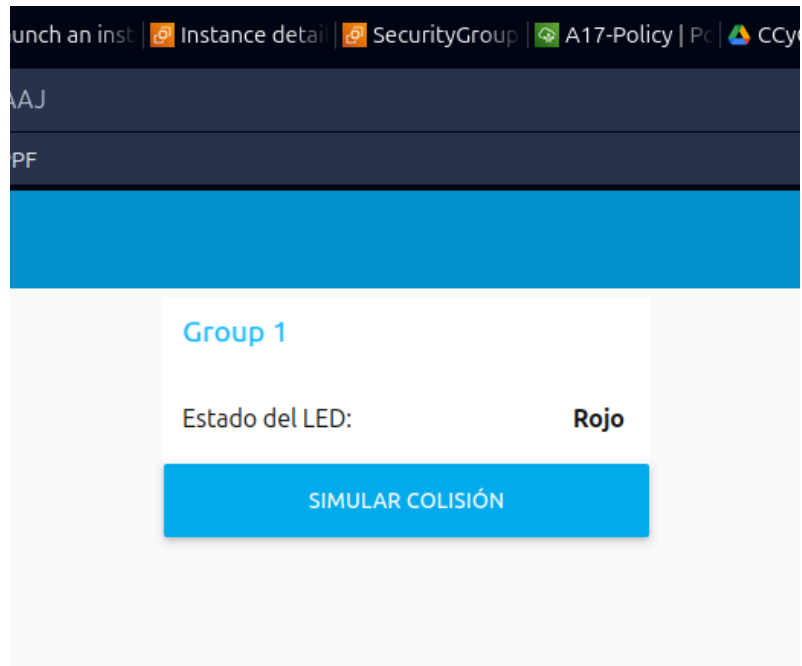




Dashboard: El dashboard que definimos permite saber el color actual del LED y simular una colisión con un botón.

Simular una colisión tiene el mismo efecto que poner un objeto delante del sensor, este envía una señal para que el LED se vuelva rojo durante unos breves segundos antes de volver a cambiar a verde.



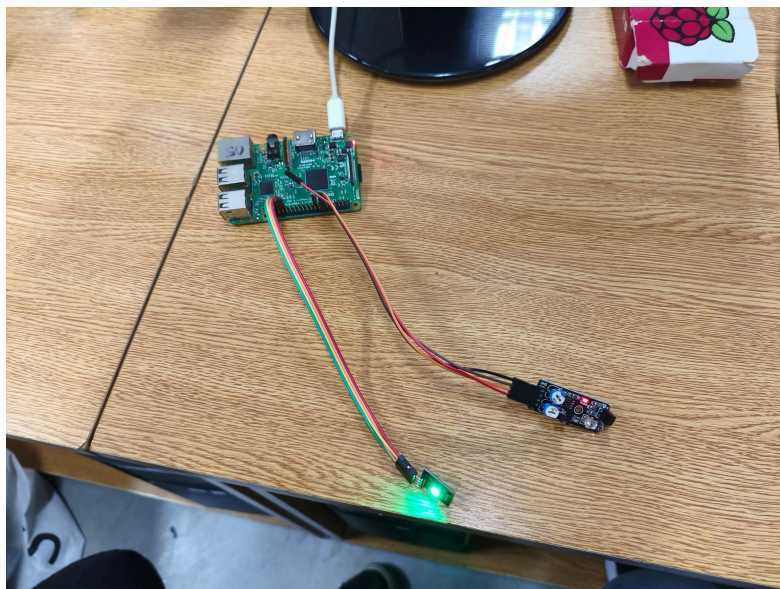


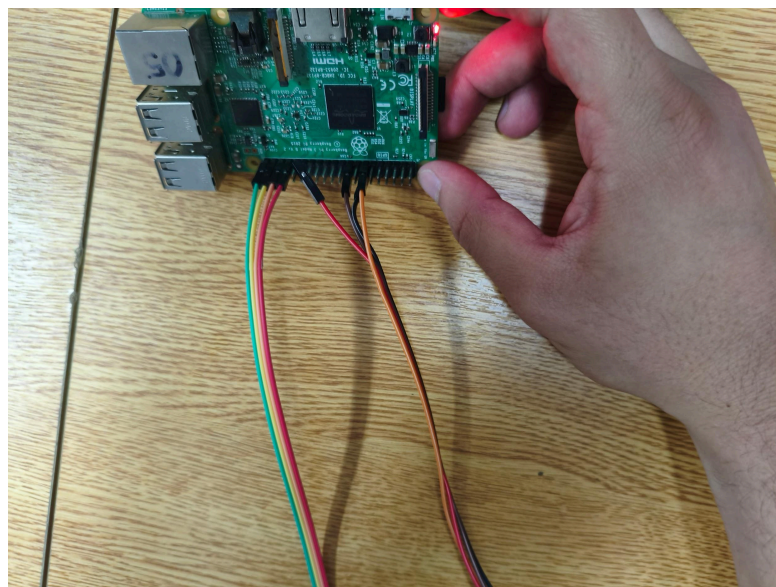
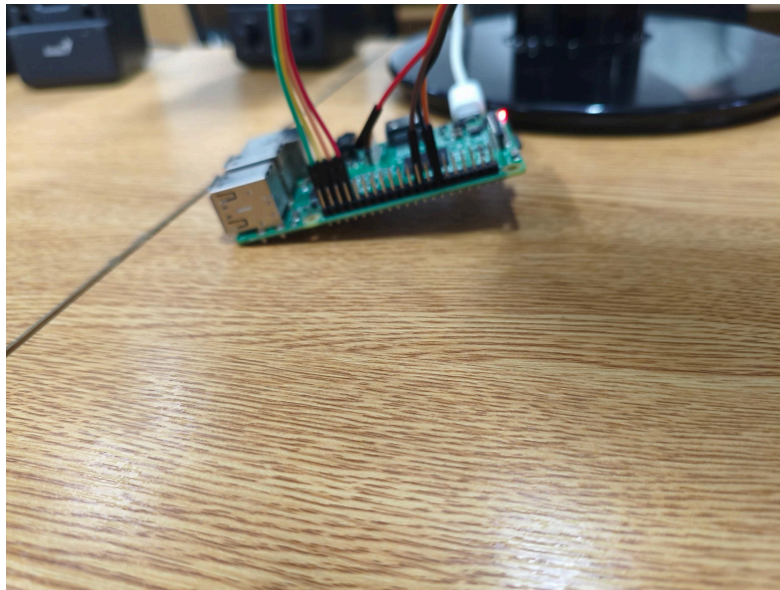
Evidencias:

Enlace al video:

<https://drive.google.com/file/d/16PxGs3i381DqH1I4Zq2brR5Uvs1p8H7S/view?usp=sharing>

Fotos:





Conclusiones: Similar a la entrega anterior, tuvimos problemas con el hardware, esta vez no fue por mal funcionamiento, sino que por desconocimiento nuestro.

Teníamos en mente poder dar un mejor estado ante la detección de un objeto cercano, pero esto no fue posible ya que el sensor que utilizamos envía la señal al remover el objeto de adelante, no al detectar un cambio.

Una mejora posible sería lo mencionado arriba, utilizar un sensor (desconocemos si existe dentro del kit de 37 sensores proporcionado) que envíe una señal al poner un objeto delante, y, otra señal, al quitarlo. De esta manera podríamos dar un indicador mucho más claro de lo que está pasando en el entorno y podría simular, por ejemplo, un detector de colisiones de un auto para asistencia a la hora de estacionar.