



Proyecto sistema de control y seguridad

MANUAL DE USUARIO



Facultad de Instrumentación
Electrónica y Ciencias Atmosféricas
UNIVERSIDAD VERACRUZANA



Experiencia educativa:

Internet de las cosas

Catedrático:

M. I. Sergio Francisco. Hernández
Machuca

Alumnos:

Hernández Reyes Roberto Saúl
S17012854

Sánchez López Luis Uriel
S17012865

Velásquez Reyes Román Gabriel
S17012869

APLICACIÓN DEL PROTOTIPO

Esta propuesta está pensada durante la estancia en casa en tiempos de cuarentena, un sistema en relación con la domótica (Conjunto de técnicas orientadas a automatizar una vivienda, que integran la tecnología en los sistemas de seguridad, gestión energética, bienestar o comunicaciones.) en nuestras vidas como apoyo de seguridad, ya que algunas veces es necesario dejar alguna luz encendida o una alarma que nos notifique algún movimiento dentro o fuera del hogar todo esto desde nuestro dispositivo móvil, con nuestra voz y permitiendo al mismo tiempo el ahorro de energía.

APLICACIÓN DEL PROTOTIPO

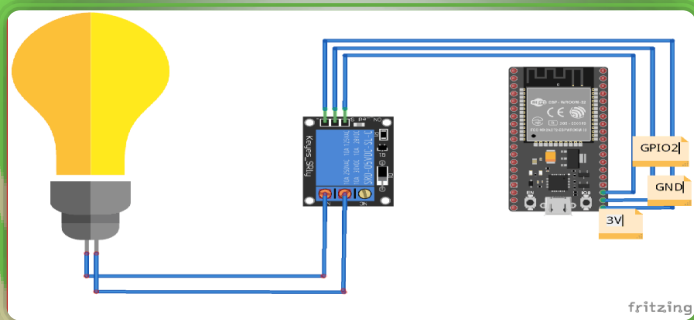
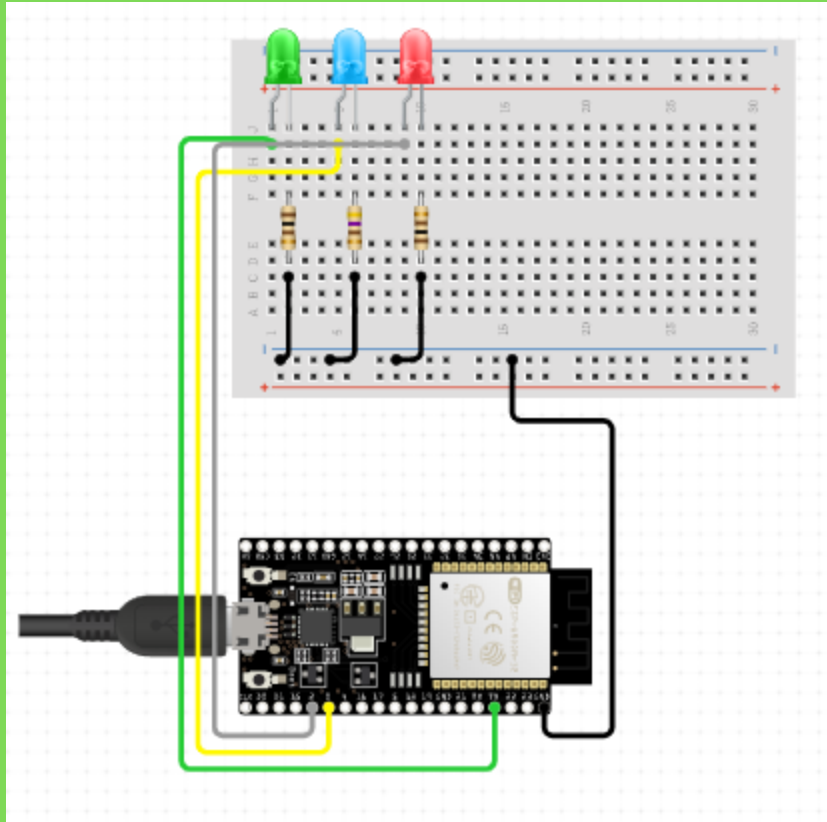
El prototipo que implementamos, se conforma de la siguiente manera

En una placa ESP32, se añadieron componentes de salida

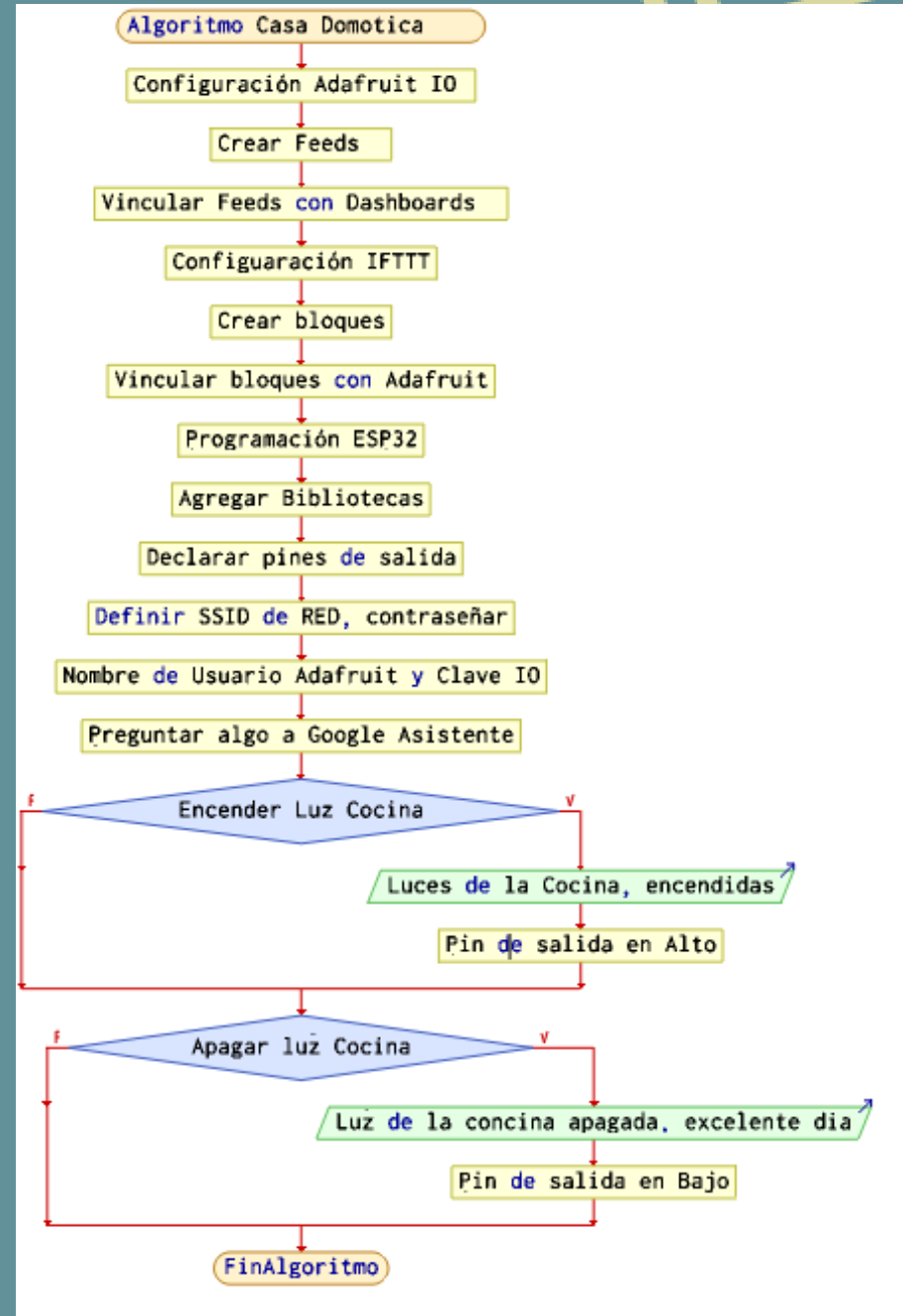
- Una placa ESP32
- Modulo relevador(Opcional)
- Leds
- Buzzer(Bocina)

A continuación, se describe la configuración o pasos para lograr una comunicación con la placa y adentrarnos a desarrollar la función del proyecto

Diagramas de Aplicación

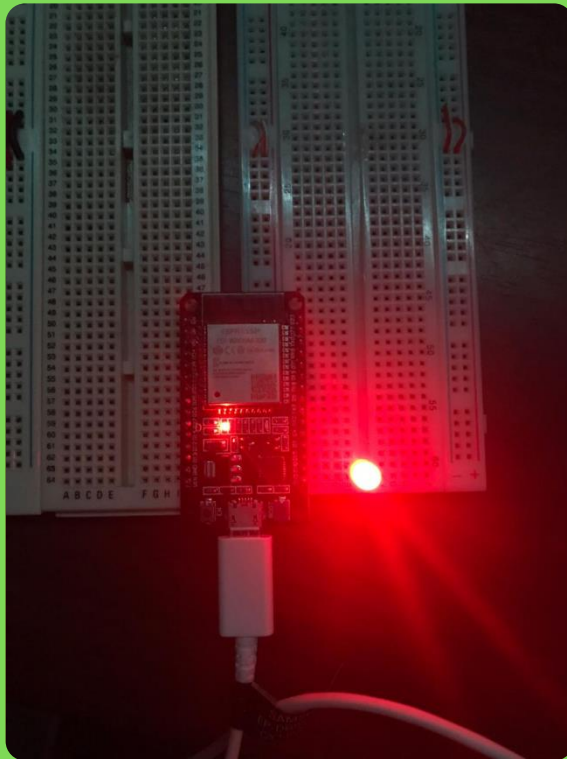


Diagramas de Flujo



INTRODUCCIÓN

Para desarrollar este proyecto implementamos: una placa ESP32, usando IFTTT (servicio web que permite crear y programar acciones para automatizar diferentes tareas y acciones en Internet) y Adafruit IO (plataforma de código libre)

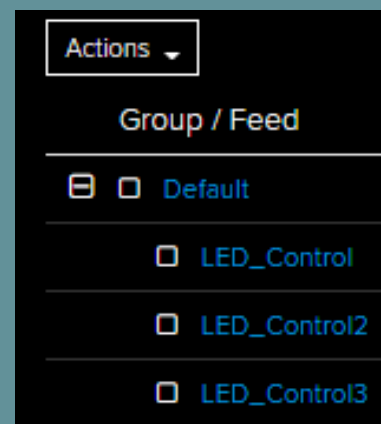


Inicialización en ADAFRUIT IO



Esta plataforma de IOT construida alrededor del Protocolo MQTT. MQTT es un protocolo de mensajería liviano que brinda a los clientes de red con recursos limitados una forma simple. Nos permitirá crear “Feeds”(y “Dashboards”

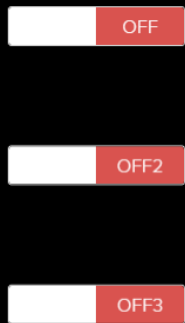
1. Debemos ingresar a la página de “<https://io.adafruit.com/>” y crearnos una cuenta
2. Crearemos “Feeds/Acciones”



Hemos creado 3 Feeds, dentro de estos, se almacenaran sus datos, si queremos que tengan acción de ON-OFF, lo desarrollaremos en los Dashboards, encargados de darles una acción, mientras en las carpetas podemos ver su historial, en el momento que se activaron y desactivaron

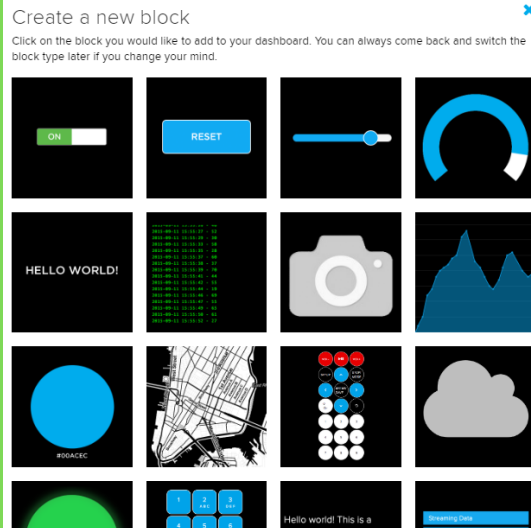
Configuración Dashboard

LEDSwitch



Creamos un bloque “LEDSwitch”, en el cual agregaremos nuestros feeds antes creados, en esta parte podremos activar manualmente nuestras salida.

Nuestros feeds, fueron asignados como bloques, los cuales pueden variar, tanto a la necesidad que necesitamos, para este caso que es un control ON-OFF utilizaremos un Toggle



Key Adafruit IO



Se despliega una serie de herramientas, como agregar un bloque para los feeds, y para eliminarlos, la imagen que contiene una llave, contiene nuestro usuario y contraseña el cual debemos agregar en nuestro código, así comunicarse y activar nuestros DashBoards

Username

romanvelrey

Active Key

aio_yBTZ07MKEN8IsiXWTQCSiLYb3x7

Usuario y clave de Adafruit IO

Conexión al Asistente de Google a través de IFTTT

En este paso conectaremos nuestro Asistente de Google al Adafruit IO MQTT Broker para permitirnos controlar las luces con comandos de voz.

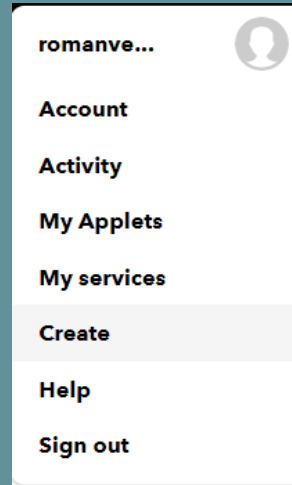
Utilizaremos la plataforma de IFTTT, debemos ingresar a la página y crearnos una cuenta “www.IFTTT.com”

En IFTTT, nos permitirá utilizar el asistente personal de google “Google Assistant” sin la necesidad de tener un dispositivo google home de una forma fácil de manejar y entender



IFTTT

Configuración Google asistente



Después de crear una cuenta, haga clic en su nombre de usuario en la esquina derecha de la ventana y haga clic en el botón "Create".

Create your own

If + This Then That

Build your own service on the **IFTTT** Platform

Su esquema de trabajo se simplifica en un condicional:

“Si esto entonces esto”, significa que si algo sucede en "Esto", entonces tenemos que hacer algo en "eso".

“Elige una frase”

[< atrás](#)

 **Elige disparador**

Paso 1 de 6

Di una frase simple

Este disparador se dispara cuando dices "Ok Google" al Asistente de Google seguido de una frase que eliges. Por ejemplo, diga "Ok Google, estoy llegando tarde" para enviarle un mensaje de texto a un miembro de la familia que está camino a casa.

Di una frase con un número

Este disparador se activa cuando dices "Ok Google" al Asistente de Google seguido de una frase como "Establecer el termostato Nest en 68". ** Use el símbolo # para especificar dónde dirá el ingrediente número


Decir una frase con un ingrediente de texto

Este disparador se dispara cuando dices "Ok Google" al Asistente de Google seguido de una frase como "Publica un tweet que dice 'Nuevo puntaje alto'". ** Usa el símbolo \$ para especificar dónde dirás el ingrediente de texto

Diga una frase con un número y un ingrediente de texto.

Este disparador se dispara cuando dices "Ok Google" al Asistente de Google seguido de una frase como "ejercicio" a las 6 PM". ** Use el símbolo # para especificar dónde dirá el ingrediente número y \$ dónde dirá el ingrediente de texto

Le damos click al botón “+”, seleccionamos la herramienta de google assist y después nos aparecerán las acciones que podemos emplear, para este caso utilizaremos la de “Decir una frase simple”

 **Di una frase simple**

Este disparador se dispara cuando dices "Ok Google" al Asistente de Google seguido de una frase que eliges. Por ejemplo, diga "Ok Google, estoy llegando tarde" para enviarle un mensaje de texto a un miembro de la familia que está camino a casa.

¿Qué quieres decir?

Encender cocina

¿Cuál es otra forma de decirlo? (Opcional)

Cocina ON

¿Y de otra manera? (Opcional)

En cocina

¿Qué quieres que diga el Asistente en respuesta?

Luces de la cocina, Encendidas

Idioma

Español

Aquí nosotros configuramos el comando de voz, con el cual realizara nuestra acción, así como también nos responderá el asistente de google.

- Hemos creado 2 disparadores, una para encender una luz y otra para apagarla

“Elige una frase”

Ahora tiene que dar Acción, así que haga clic en el botón + de "Eso", busque Adafruit y haga clic en "Enviar datos a Adafruit IO"

Ahora le pedirá que seleccione el nombre del feed, así que seleccione el feed que realizó anteriormente para este proyecto y en Datos para guardar enviaremos ON2 para este applet y haga clic en Crear acción. Lo mismo debemos hacer para un OFF2

Feed name

LED_Control2

The name of the feed to save data to.

Data to save

ON2

The data to be saved to your feed.

Add ingredient

Después de crear ambas aplicaciones, vaya a "Mis applets" y puede ver ambas aplicaciones aquí.

Mis applets

Si dice "Apagar cocina", envíe los datos al LED_Control2 feed

por romanvelrey

☐ Conectado

Si dice "Encender cocina", envíe los datos a LED_Control2 feed

por romanvelrey

☐ Conectado

Si dice "Apagar Recamara", envíe los datos a LED_Control feed

por romanvelrey

☐ Conectado

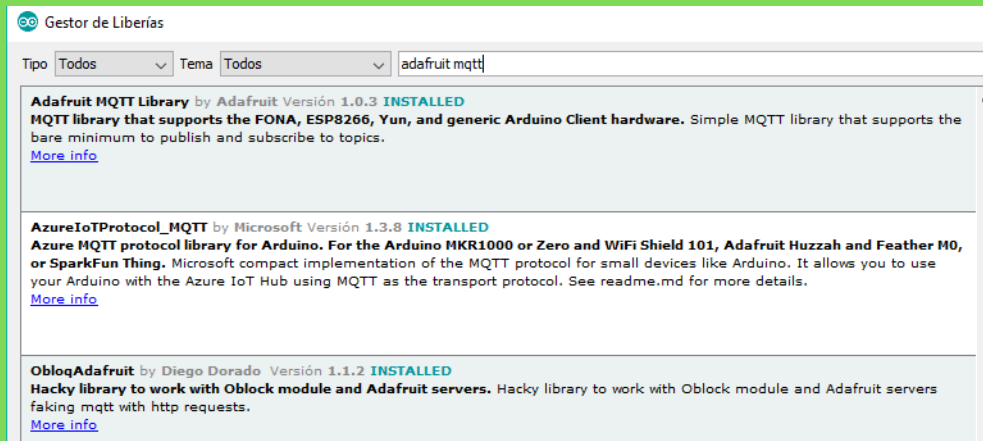
Si dice "Encender Recamara", envíe los datos a LED_Control feed

por romanvelrey

☐ Conectado

Programación de ESP32 para el LED controlado por el Asistente de Google

Pasaremos a la parte de programación para concluir el proyecto. Utilizamos Arduino IDE para programar ESP32. Para conectar ESP32 con Adafruit IO, debe incluir la Biblioteca de Cliente Adafruit MQTT en su IDE, para esto abra su Arduino IDE y vaya a *Sketch -> include library -> Manage library* y busque “**adafruit mqtt** ” Entonces se le mostrará una biblioteca asociada con esto; solo tienes que instalarlo.



En la siguiente página explicaremos como integrar una comunicación entre el código y la plataforma de Adafruit en la cual ya creamos nuestros disparadores

Configuración

Al comienzo del programa, se debe agregar las siguientes bibliotecas:

```
#include <WiFi.h>

#include "Adafruit_MQTT.h"

#include "Adafruit_MQTT_Client.h"
```

Ahora tiene que definir su SSID de red, contraseña, su nombre de usuario Adafruit y su clave AIO que obtuvo al configurar Adafruit IO.

```
#define WLAN_SSID      "INFINITUM5D9E_2.4"

#define WLAN_PASS      "Bacon10deMayo"

#define AIO_SERVER      "io.adafruit.com"

#define AIO_SERVERPORT  1883

#define AIO_USERNAME    "romanvelrey"

#define AIO_KEY          "aio_yBTZ07MKEN8lsilXWTQCsiLYb3x7"
```


Prueba de control LED con Google Assistant

