



UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES
MAESTRÍA EN INGENIERÍA TELEMÁTICA
DESARROLLO DE APLICACIONES PARA PLATAFORMAS UBICUAS

Docente: Juan Pablo Ruiz Rosero jpabloruiz@unicauca.edu.co

9. Bombilla WiFi por MQTT

1. OBJETIVO

Esta guía tiene como objetivo el desarrollo de una aplicación para el control de una bombilla WiFi por MQTT.

2. DESCRIPCIÓN

Las bombillas WiFi basadas en el microcontrolador ESP8266, como las de las marcas Ai Light o UL Listed, ofrecen grandes ventajas para el desarrollo de aplicaciones Smart Home mediante el protocolo MQTT. Estas bombillas permiten la programación de un custom firmware como ESPurna, el cual expande sus capacidades.

3. ESPURNA FIRMWARE

ESPurna, es un custom firmware para dispositivos basados en el microcontrolador WiFi ESP8266. Este usa el core de Arduino framework y varias librerías de terceros para soportar un gran número de dispositivos. Entre sus características se encuentran:

- WiFi en modo access point para su configuración.
- WiFi en modo estación con la capacidad de conectarse a un access point con DHCP o static IP.
- Memoria para definir 5 conexiones a access points WiFi.
- MQTT para el manejo de relés, LEDs y estados.
- Integración con Amazon Alexa, Google Assistant, Domoticz e InfluxDB.
- Mayor información en: <https://bitbucket.org/xoseperez/espurna>

Las bombillas WiFi destinadas para esta práctica ya cuentan con el firmware ESPurna versión 1.9.5, sin embargo en el siguiente enlace puede encontrar instrucciones para programar este firmware en las bombillas:
<https://nathan.chantrell.net/20170527/ai-light-rgbw-led-wi-fi-light-bulbs-with-mqtt-esp8266/>

4. CONFIGURACIÓN DE LA BOMBILLA WiFi

1. Si la bombilla WiFi no es capaz de conectarse a uno de los access point previamente configurados en su memoria, esta pasará a modo Access Point, con el SSID: **AI_LIGHT_XXXXXX**, donde XXXXXX son los 3 últimos bytes de su MAC address.
2. Conéctese al access point **AI_LIGHT_XXXXXX** de su bombilla WiFi. La clave de red es: **fibonacci**. Si la anterior clave no funciona, esto quiere decir que su bombilla ya ha sido configurada previamente, en este caso utilice como clave: **123456Ab**
3. Una vez conectado al access point **AI_LIGHT_XXXXXX**, abra su navegador web e introduzca la dirección: **http://192.168.4.1**. Utilice como username: **admin** y como clave la misma que utilizó para conectarse al access point WiFi de la bombilla.

4. Si la contraseña fue `fibonacci`, se nos pedirá una nueva contraseña. Introducir como nueva contraseña `123456Ab`. Nota (A en mayúscula y b en minúscula). Dar click en Update y luego ingresar con username `admin` y contraseña `123456Ab`.
5. Prueba si la bombilla WiFi funciona mediante la interfaz web, para esto cambien `Switch Status` de OFF a ON y mueva las perillas de los colores RGB.
6. Puede modificar el brillo general de los LEDs RGB con la perilla `Brightness`. Igualmente puede modificar el brillo de los LEDs de luz blanca mediante la perilla `Channel 4`
7. Configure el broker MQTT. Entre a la sección `MQTT` del menu izquierdo, habilite el MQTT modificando `Enable MQTT` de OFF a ON.
8. En la opción de `MQTT Broker` introduzca: `iot.eclipse.org`
9. Deje el `MQTT Port` en 1883
10. Deje `MQTT User` y `MQTT Password` en blanco
11. Cambie `MQTT Root topic` a `/unicauca/light/{identifier}`
12. Para conectar su bombilla a Internet, en la parte izquierda acceda al menu `WIFI`, pulse el botón `Add network` e introduzca el `Network SSID` y `Password` de su red inalámbrica, la misma a la que tiene conectado su computador. Pulse el botón `Update` del lado izquierdo para guardar los cambios y luego el botón `Reset` para que la bombilla se reinicie y se conecte a la red WiFi Configurada.
13. En su computador, si no lo tiene aun instalado, instale en python el paquete `paho-mqtt`

```
sudo pip install paho-mqtt
```

14. Cree una carpeta llamada `guia10`, y en ella un archivo llamado `testMqtt.py` con el siguiente contenido:

```
import paho.mqtt.publish as publish
```

```
# Turn on the relay
publish.single("/unicauca/light/AI_LIGHT_2EF984/relay/0/set", "1", hostname="iot.eclipse.org")
# Set brightness to 255
publish.single("/unicauca/light/AI_LIGHT_2EF984/brightness/set", "255", hostname="iot.eclipse.org")
# Set red ligh to the max
publish.single("/unicauca/light/AI_LIGHT_2EF984/color/set", "255,0,0", hostname="iot.eclipse.org")
```

Este script enviará una serie de comandos MQTT para activar el relé de la bombilla, colocar el brillo al máximo y activar los LEDs rojos.

15. Cree una aplicación que cumpla con una de las siguientes características:
 - Mediante una interfaz web programar alarmas de encendido y apagado de la bombilla.
 - Activar diferentes colores de la bombilla mediante la detección de diferentes beacons Bluetooth.
 - Informar si tiene un nuevo correo electrónico mediante un color en la bombilla.