

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

NESNELERİN İNTERNETİ VE UYGULAMALARI

AKILLI EV OTOMASYONU

HAZIRLAYANLAR:

Uğur ÇELİK B171210042

Hüsneİrem KAYA B171210309

Zeynep ALDİNÇ B181210390

AMAÇ:

Yapılan Akıllı Ev Otomasyonu Projesi; ev içerisindeki günlük yaşamda kolaylık, güvenlik ve sadelik planlamaktadır. Ev içerisindeki priz sistemine uzaktan mobil cihaz aracılığıyla kontrol yapılarak, ev içerisinde oluşabilecek elektrik kaçağı ve buna bağlı oluşabilecek yangın riski ortadan kaldırılmış olur. Ev içerisindeki lamba sistemine mobil cihazla kontrol yapılarak ev içerisindeki anahtar sistemine olan ihtiyaç kaldırılmış, böylelikle sadelik ve kablolama maliyetinden tasarruf edilmiş oldu.

KULLANILAN TEKNOLOJİ, MALZEME VE PROTOKOLLER – MALZEME MALİYETLERİ:

KULLANILAN MALZEMELER:

Çift Kanallı Röle 🡪 6 TL

NodeMCU Wifi Modülü 🡪 25 TL

Mini Breadboard 🡪 2 TL

Adaptör 🡪 45 TL

Duy 🡪 5 TL

Ampul 🡪 9 TL

Sıva Üstü Priz 🡪 12.5 TL

Topraklı Kablo (yarım metre) 🡪 2.5 TL

Jumper (7 tane ( 4 erkek-dişi, 3 dişi-dişi )) 🡪 0.50 TL

KULLANILAN TEKNOLOJİLER VE YAZILIM

Adafruit IO

IFTTT

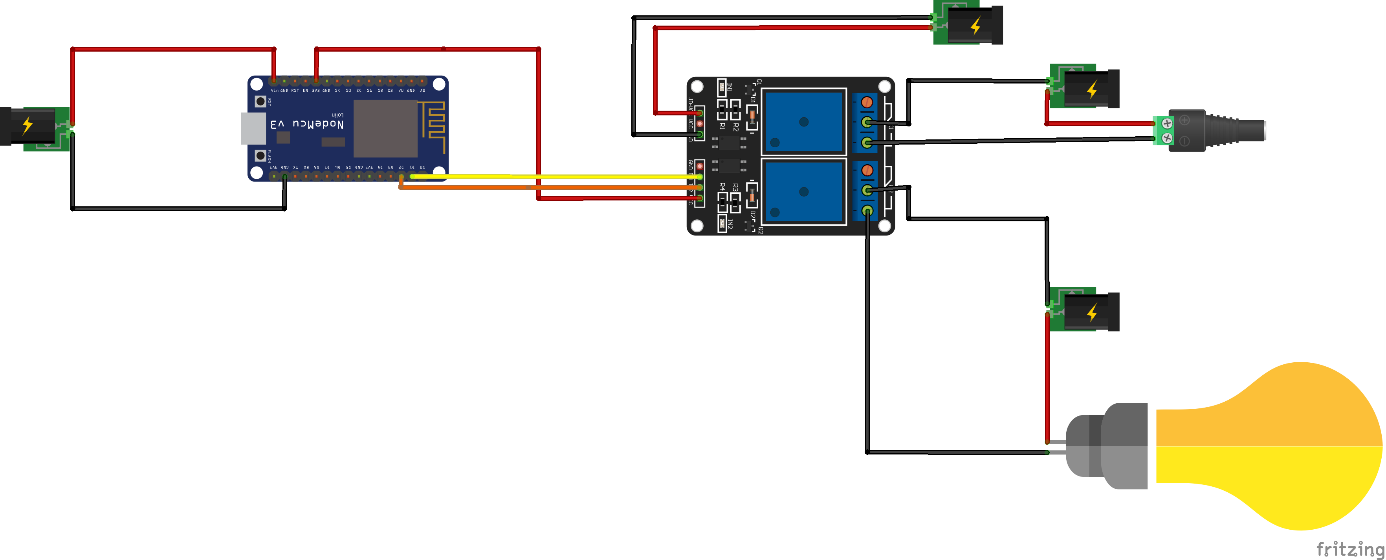
Arduino IDE

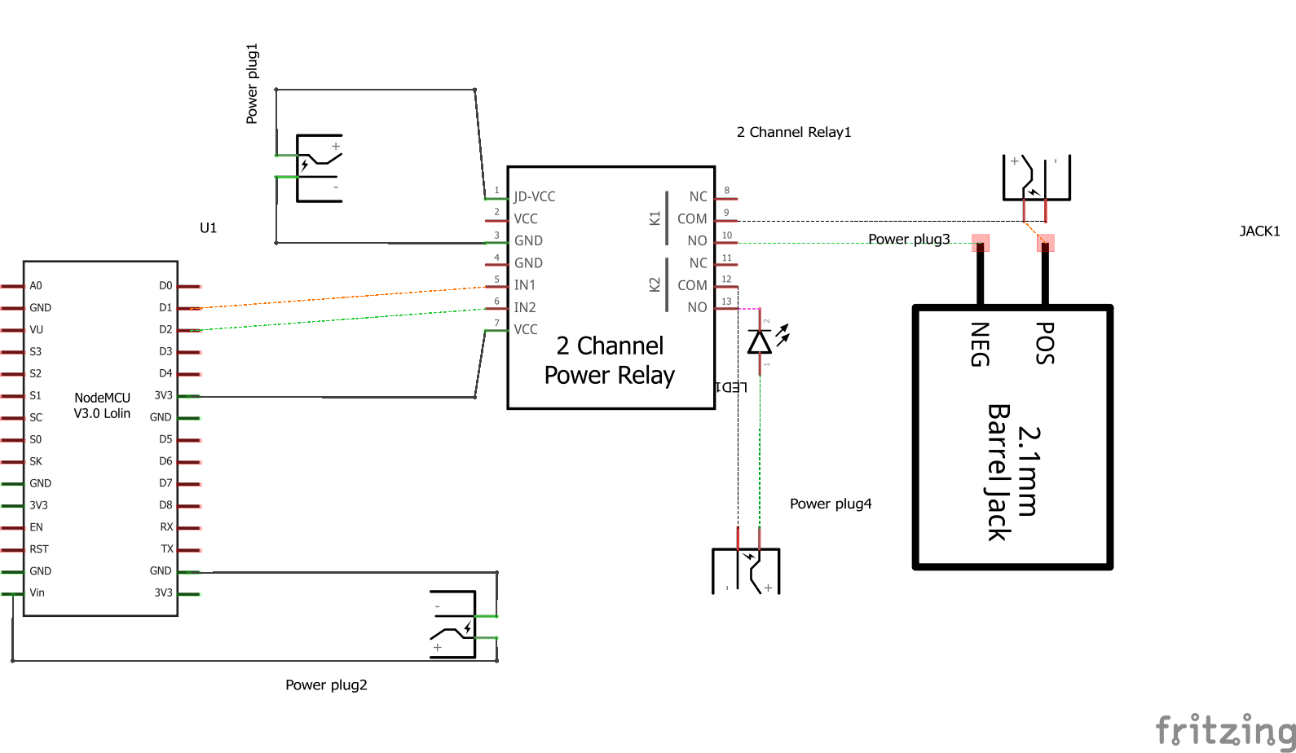
KULLANILAN PROTOKOLLER

MQTT Protokolü

HTTP 1.1 Protokolü

DEVRE ŞEMATİĞİ:





SİSTEMİN YAPILIŞI:

IFTTT, Google ve Adafruit hesapları açılarak başlandı. Adafruit üzerinde kontrol edilecek lamba ve priz için dashboard ve alanlar oluşturuldu.

Arduino IDE içerisinde yeni bir proje oluşturuldu. NodeMCU bağlantısı için gerekli ESP8266, kütüphanesi ve Adafruithttp, Adafruitclient kütüphaneleri kuruldu. Dashboard’ın AIO Key’i Arduino kodundaki ilgili alana yerleştirilerek Adafruit ve NodeMCU arasında bağlantı sağlanmış oldu.

IFTTT üzerinde oluşturduğumuz “Turn On Socket”, “Turn Off Socket”, “Turn On Light”, “Turn Off Light” appletlerine Google ve Adafruit hesapları ile bağlanıldı.

Devre şematiğine bakılarak uygun röle ve kablolama bağlantıları yapıldı. Ayrıca kullanılacak priz ve lambanın kabloları ayrıca rölelerin kontrol uçlarına bağlandı. Arduino ortamında hazırlanan kod derlenip NodeMCU’ya yüklenerek çalıştırıldı. Böylelikle sistem çalışır hale getirilmiş oldu.

ARDUINO IDE KODLARI:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* IOT Ev Otomasyon Projesi

\* Röle ile Priz ve Lamba Kontrolü

\*

\* Hazırlayanlar: Zeynep Aldinç B181210390 zeynep.aldinc1@ogr.sakarya.edu.tr

\* Hüsne İrem Kaya B171210309 husneirem.kaya@ogr.sakarya.edu.tr

\* Uğur Çelik B171210042 ugur.celik10@ogr.sakarya.edu.tr

\* Version : 1.00.0

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <ESP8266WiFi.h>

#include "Adafruit\_MQTT.h"

#include "Adafruit\_MQTT\_Client.h"

/\*\*\* Kablosuz Ağ Bilgileri \*\*\*/

#define WLAN\_SSID "EV" // "Kablosuz Ağ Adı"

#define WLAN\_PASSWORD "12344321" // "Kablosuz Ağ Şifresi"

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Adafruit.io Kurulumu \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define AIO\_SERVER "io.adafruit.com"

#define AIO\_SERVERPORT 1883 // use 8883 for SSL

#define AIO\_USERNAME "PlaLoho"

#define AIO\_KEY "0e5859df0c0e4bcb8519985b246c7fe3"

/\*\*\*\*\*\*\*\*\*\*\*\* Global State (you don't need to change this!) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

// MQTT bağlantısı için gerekli WifiClient sınıf nesnesidir.

WiFiClient client;

// WifiClient sınıfını wifi client ve yetkilendirme giriş bilgilerine göre ayarlar.

Adafruit\_MQTT\_Client mqtt(&client, AIO\_SERVER, AIO\_SERVERPORT, AIO\_USERNAME, AIO\_KEY);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Yayınlar \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//Lamba1 yayınının dinlenebilmesi için gerekli kullanıcı ve yayın bilgileri ayarlanır.

Adafruit\_MQTT\_Subscribe lamba1 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/lamba1");

//Priz1 yayınının dinlenebilmesi için gerekli kullanıcı ve yayın bilgileri ayarlanır.

Adafruit\_MQTT\_Subscribe priz1 = Adafruit\_MQTT\_Subscribe(&mqtt, AIO\_USERNAME "/feeds/priz1");

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Taslak Kod \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//MQTT bağlantı durummu kontrol eden fonksiyon.

void MQTT\_connect();

void setup() {

Serial.begin(115000);

delay(10);

WiFi\_Setup();

//Lamba rölesi kontrol pini

pinMode(D1,OUTPUT);

digitalWrite(D1,1);

//Priz rölesi kontrol pini

pinMode(D2,OUTPUT);

digitalWrite(D2,1);

mqtt.subscribe(&lamba1);

mqtt.subscribe(&priz1);

}

void loop() {

MQTT\_connect();

Adafruit\_MQTT\_Subscribe \*subscription;

while ((subscription = mqtt.readSubscription(2000))) {

if (subscription == &lamba1) {

Serial.print(F("LambaDurum: "));

Serial.println((char \*)lamba1.lastread);

char \*oku;

oku=(char \*)lamba1.lastread;

if (strcmp(oku,"ON")==0)digitalWrite(D2,0);

if (strcmp(oku,"OFF")==0)digitalWrite(D2,1);

}

if (subscription == &priz1) {

Serial.print(F("PrizDurum: "));

Serial.println((char \*)priz1.lastread);

char \*oku2;

oku2=(char \*)priz1.lastread;

if (strcmp(oku2,"ON")==0)digitalWrite(D1,0);

if (strcmp(oku2,"OFF")==0)digitalWrite(D1,1);

}

}

}

void WiFi\_Setup(){

delay(10);

Serial.println(); Serial.print(WLAN\_SSID);

Serial.print(" Kablosuz Agina Baglaniyor");

WiFi.begin(WLAN\_SSID, WLAN\_PASSWORD);

// WiFi durum kontrolü

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println();Serial.print(WLAN\_SSID);

Serial.println("Kablosuz Agina Baglandi");

Serial.println("IP adresi: ");

Serial.println(WiFi.localIP());

}

// Bu fonksiyon gerekli bağlantının durumunu kontrol eder. Bağlantı olmaması durumunda bağlanmayı yineler.

void MQTT\_connect() {

uint8\_t ret;

// Bağlı ise çağrıldığı yere döner

if (mqtt.connected()) return;

Serial.print("Connecting to MQTT... ");

uint8\_t retries = 3;

while ((ret = mqtt.connect()) != 0) { // 0 bilgisi ile bağlantı anlaşılır.

Serial.println(mqtt.connectErrorString(ret));

Serial.println("Retrying MQTT connection in 5 seconds...");

mqtt.disconnect();

delay(5000); // wait 5 seconds

retries--;

if (retries == 0) {

// basically die and wait for WDT to reset me

while (1);

}

}

Serial.println("MQTT Connected!");

}