

Nesnelerin İnterneti (IoT) Sensor Monitoring

1. Giriş

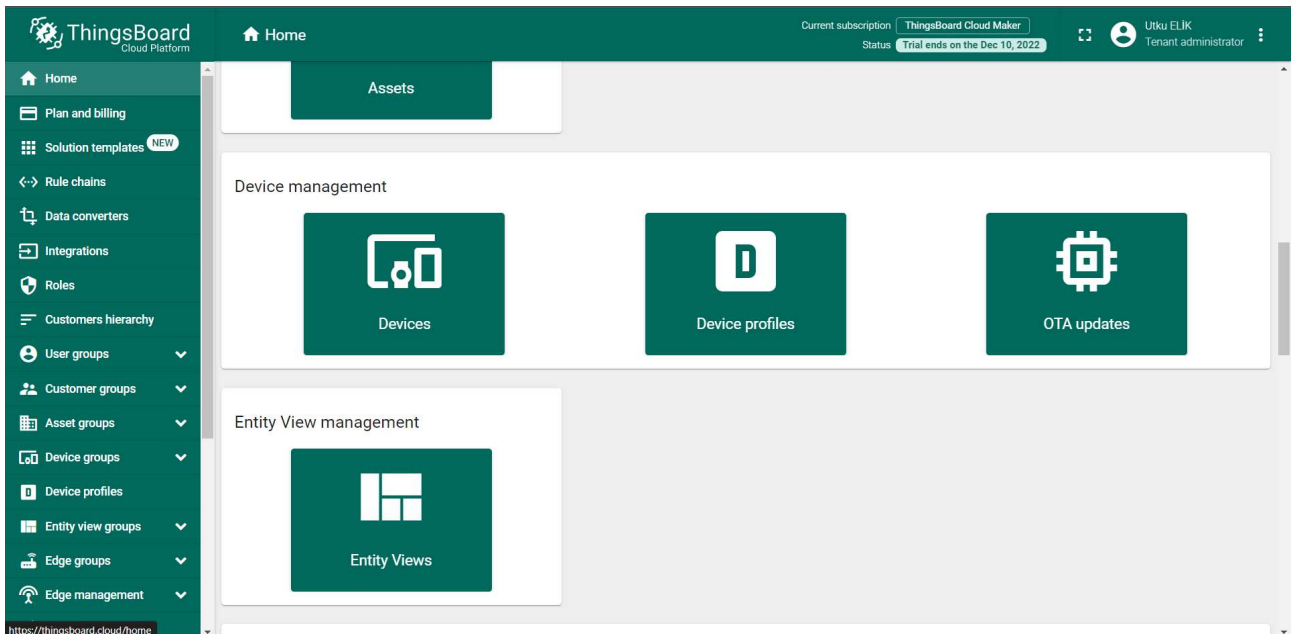
Çevrede bulunan fiziksel özelliklerin sensörler aracılığı ile algılanıp, bu sensörlerin ürettiği sinyallerin mikrodnetleyici ile anlamlandırılıp, herhangi bir kullanıcının bu verileri uzaktan görüntüleyebilmesi veya müdahale edebilmesi için monitorize işlemlerinin gerçekleştirilmesi gerekmektedir. Bu işlem sens edilen ortamda mikrodnetleyici yapısına LCD Display dahil edilerek gerçekleştirilebilir. Fakat kullanıcının uzaktan sisteme erişip bilgileri görüntüleyebilmesi ve uzaktan müdahale edebilmesi için bu yapının internet ortamına taşınması gerekmektedir. IoT uygulamalarında sıklıkla kullanılan ESP32-DevModule ve ESP8266-12E NodeMCU mikrodnetleyici geliştirme kartları ile geliştirilen uygulamaların internet ortamına taşınması sağlanmaktadır.

Monitorize işlemi çok farklı şekillerde yapılabilmektedir. İnternet üzerinde bu tarz ihtiyaçları karşılamak için farklı uygulamalar bulunmaktadır. Bunlara ThingsBoard, IoT Cloud, Blynk ve AppInventor uygulamalarını örnek verebiliriz. Bunun haricinde IoT geliştirme kartları üzerinde kendi HTML, CSS ve JS dosyalarınızı dahil ederek mikrodnetleyicinin bir WEB Server olarak çalışmasını sağlayabilirsiniz.

Bu dökümanda ESP32 geliştirme kartı kullanılarak HDC1080 Isı/Nem sensöründen okunan veriler ThingsBoard uygulaması üzerinden monitorize işlemlerinin nasıl yapılacağı gösterilecektir.

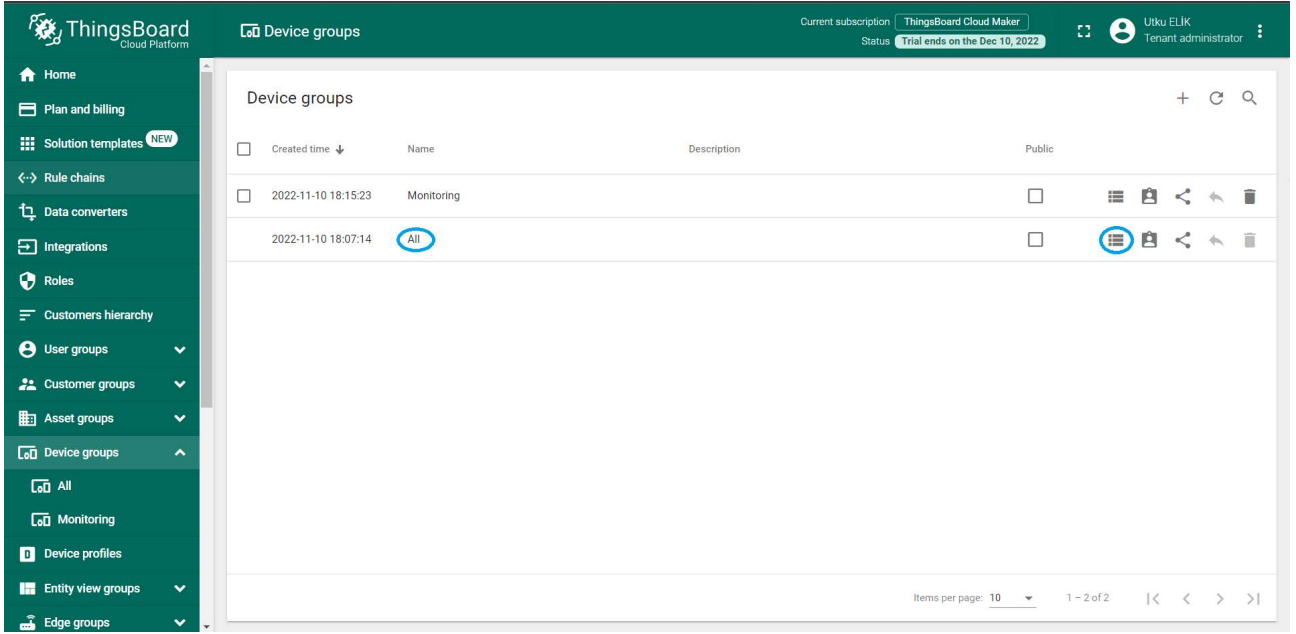
2. ThingsBoard Ortamının Hazırlanması

- İlk olarak ThingsBoard sitesine gidip üye oluyoruz. (<https://thingsboard.io>)
- Device Management kısmından Devices modülünü seçiyoruz.



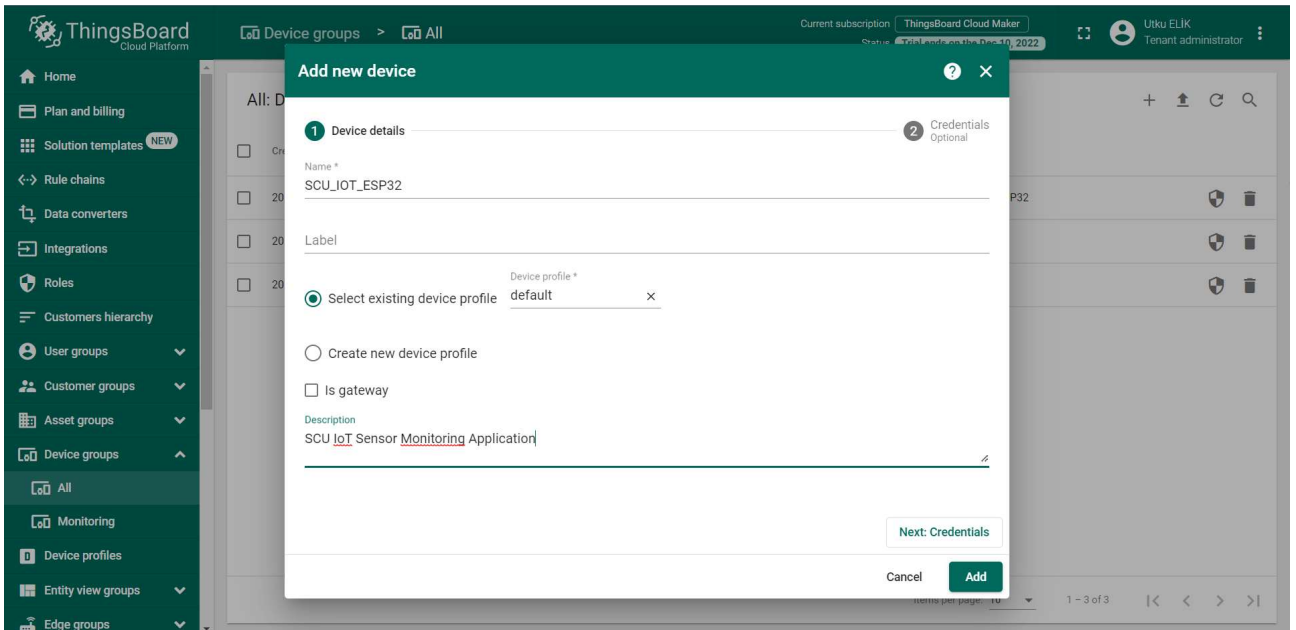
Şekil-1 Devices

- Device Groups kısmında Name: All olan sekmeyi açıyoruz.



Şekil-2 Device Groups

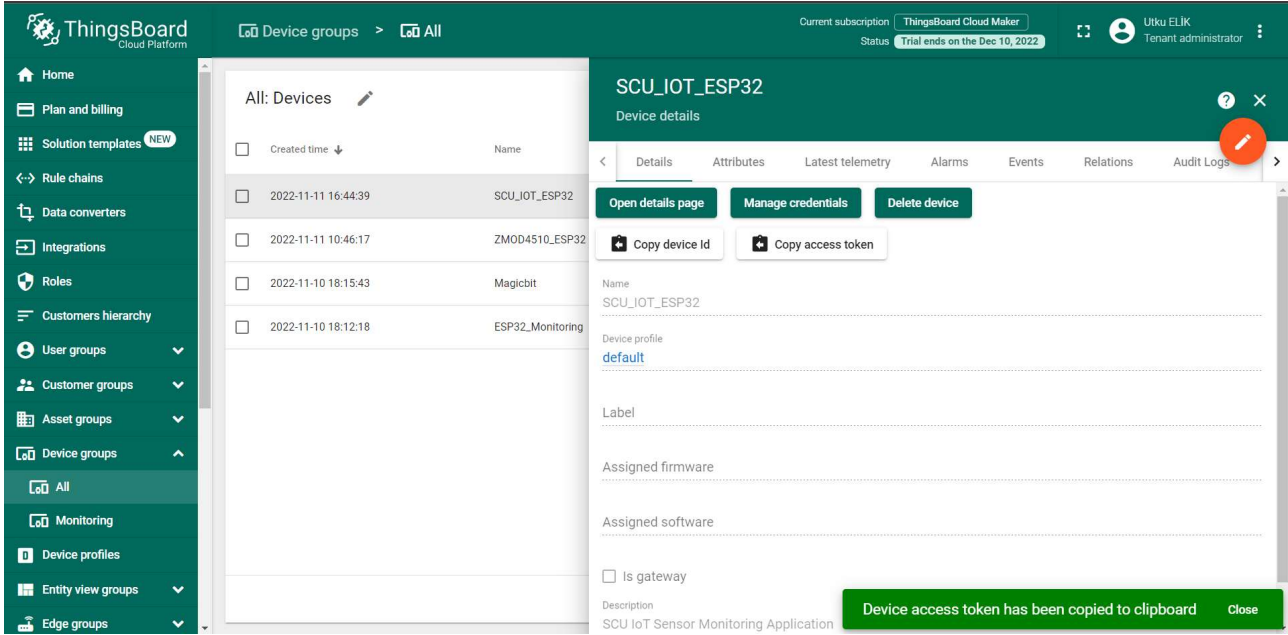
- Açılan ekranda sağ üstteki (+ Add Device) işareti ile sisteme cihaz ekliyoruz.



Şekil-3 Adding Device

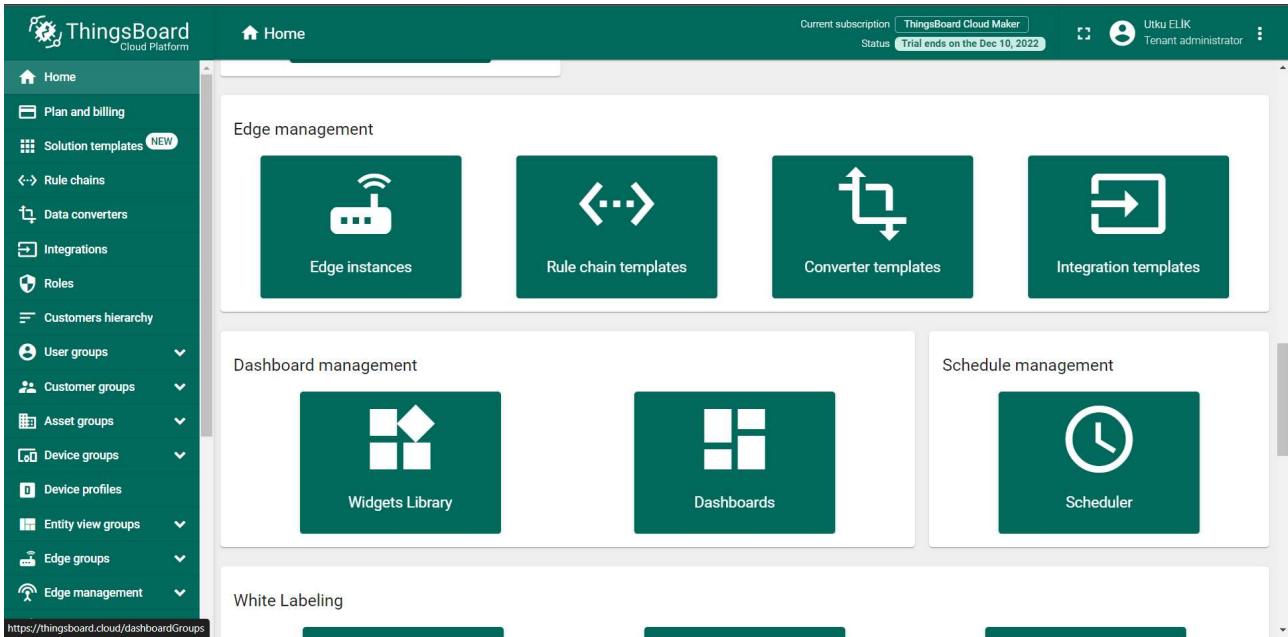
- Şekil-3'te açılan ekran üzerinde cihazımızın ismini belirleyip diğer ayarları yapıyoruz ve Credentials kısmını es geçip Add butonu ile cihazı ekliyoruz.

- Eklemiş olduğumuz cihaza tıklayarak **access token** bilgisini alıyoruz. Örneğin UZgsXAYnZLPJgYgq5Pcl



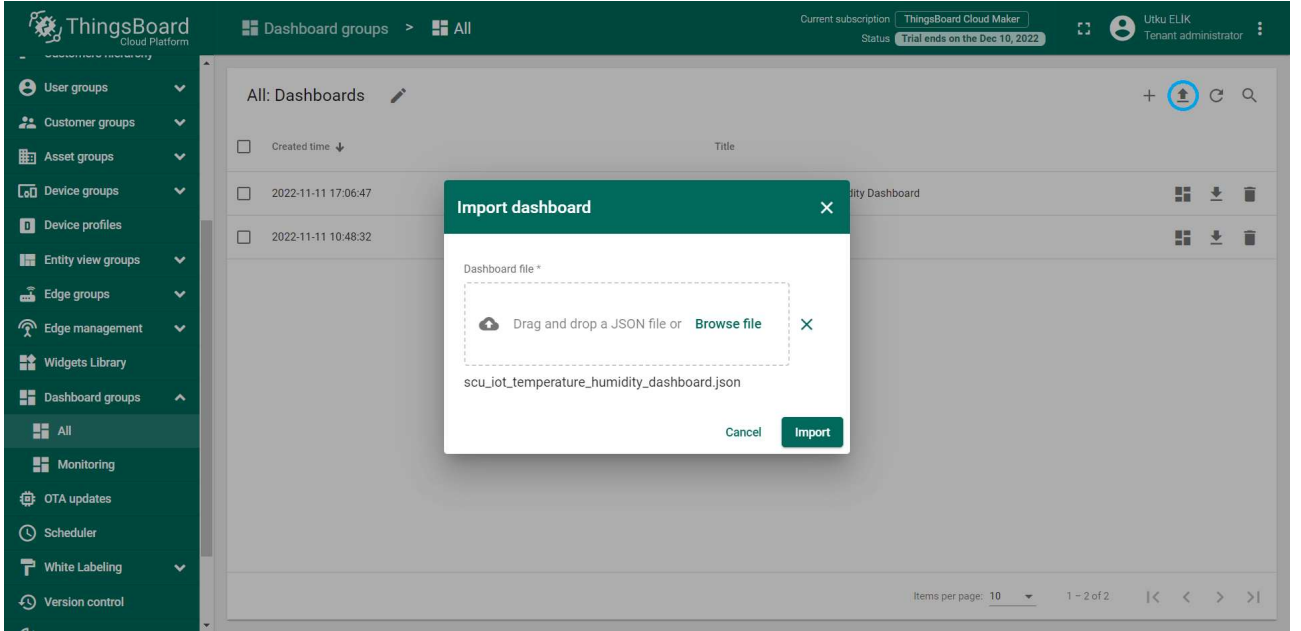
Şekil-4 Copy Access Token

- Gönderecek olduğumuz verilerin monitorize edilebilmesi için uygulama üzerinde bu kez Dashboard Management kısmından Dashboards modülünü seçiyoruz.



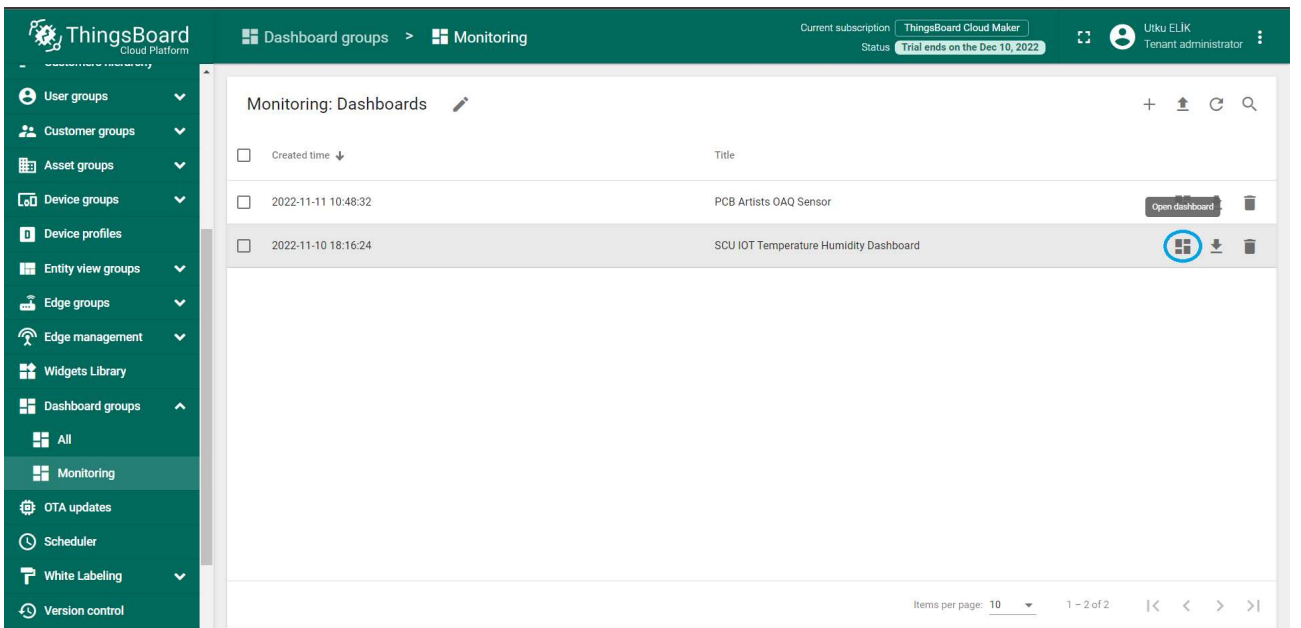
Şekil-5 Dashboards

- Dashboard Groups kısmından All olan sekmeyi açıp sağ üst tarafta bulunan (Yükleme Add Dashboard) işareti ile uygulamaya önceden hazırladığımız JSON dosya formatında olan görüntü dosyamızı dahil ediyoruz.



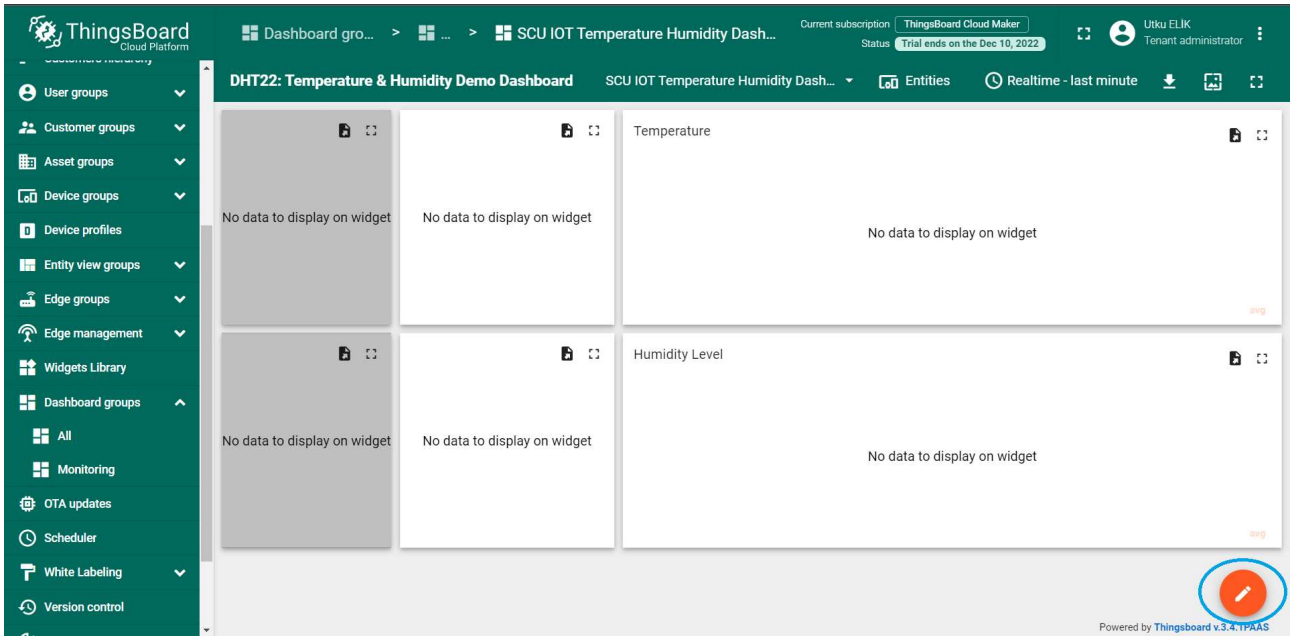
Şekil-6 Import Dashboard

- SCU IOT Temperature Humidity Dashboard dosyasını sağ tarafında bulunan Open Dashboard simgesi ile açıyoruz.



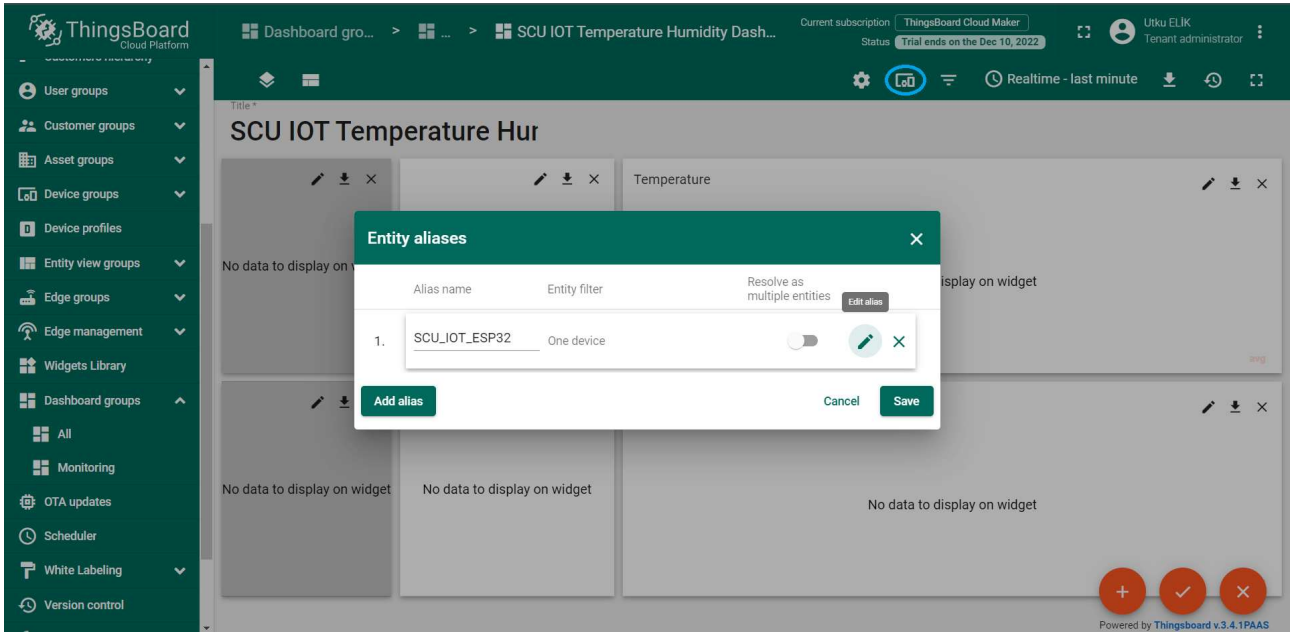
Şekil-7 Open Dashboard

- Şekil-8’de gelen ekran üzerinde sağ alt tarafında bulunan Edit sembolü ile Device-Dashboard bağlantısını gerçekleştirmemiz gerekiyor.



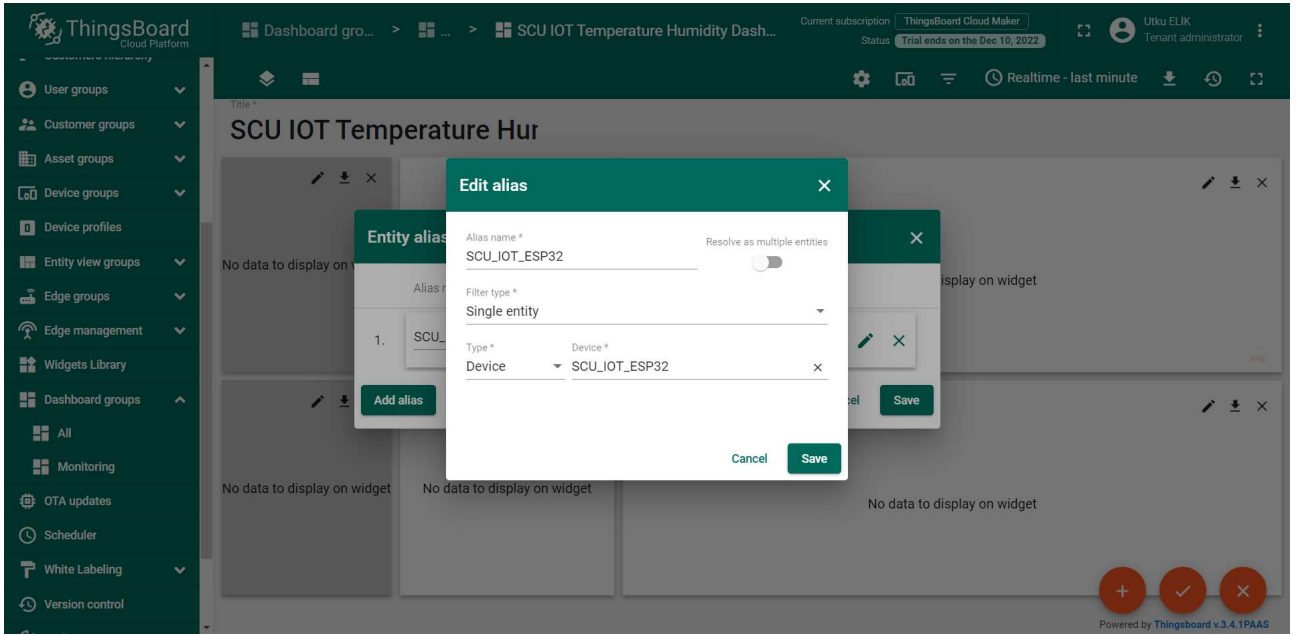
Şekil-8 Edit Dashboard

- Entitiy aliases sekmesi üzerinden Dashboard’a Device ekleme işlemini gerçekleştiriyoruz.



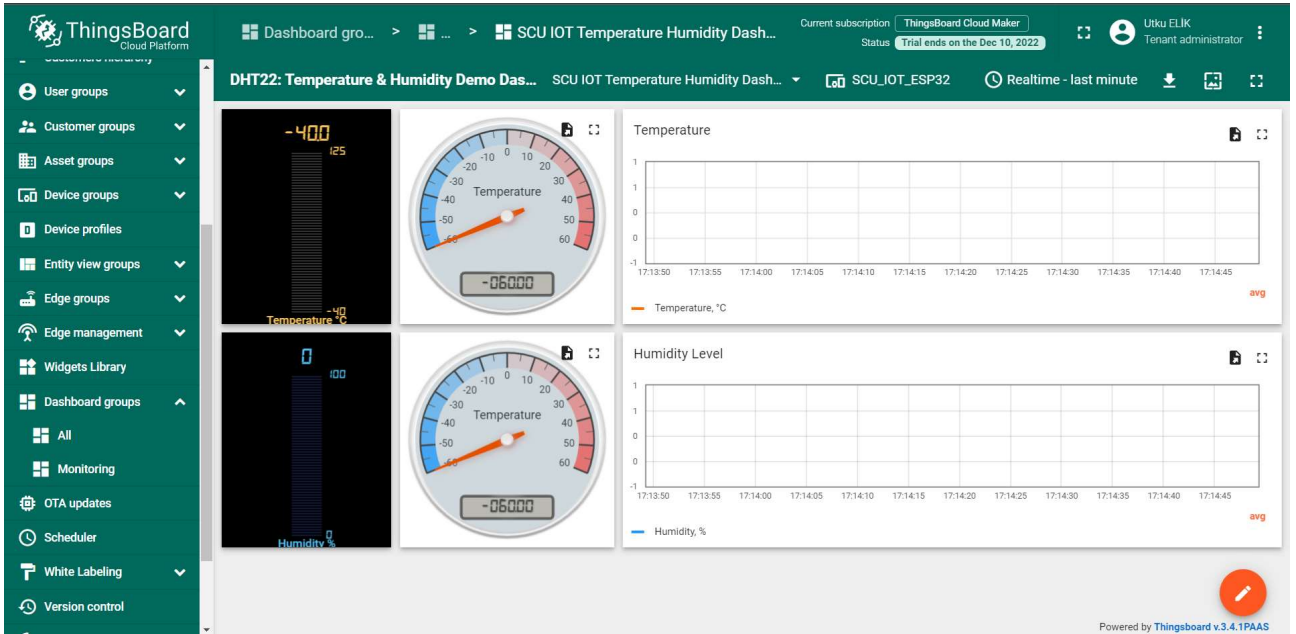
Şekil-9 Entity Aliases

- Bu sekmede Filter Type ve Device'ı seçtikten sonra ayarları kaydedip çıkıyoruz.



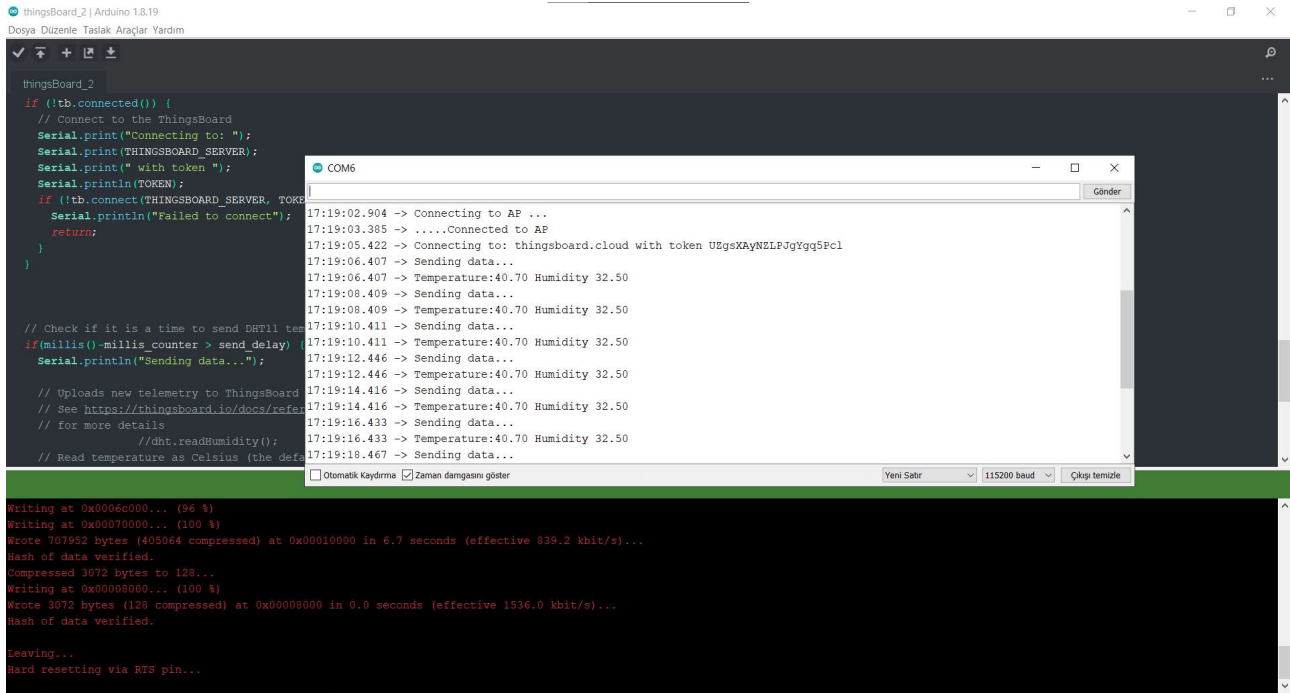
Şekil-10 Edit Alias

- ESP32 kartımız sisteme bağlanıp veri göndermediği için Şekil-11'de görüldüğü üzere boş veri akışı görüntülenmektedir.



Şekil-11 Dafult Values

- ESP32 mikrodnetleyici kartı üzerine yazdığımız yazılım sayesinde HDC1080 Isı/Nem sensöründen okuduğumuz değerleri ThingsBoard üzerinde geliştirdiğimiz arayüze gönderiyoruz.



```
thingsBoard_2 | Arduino 1.8.19
Dosya Düzenle Taslak Araçlar Yardım

thingsBoard_2
// Connect to the ThingsBoard
Serial.print("Connecting to: ");
Serial.print(THINGSBOARD_SERVER);
Serial.print(" With token ");
Serial.println(TOKEN);
if (!tb.connect(THINGSBOARD_SERVER, TOKEN)) {
  Serial.println("Failed to connect");
  return;
}

// Check if it is a time to send DHT11 temperature and humidity
if (millis() - millis_counter > send_delay) {
  Serial.println("Sending data...");

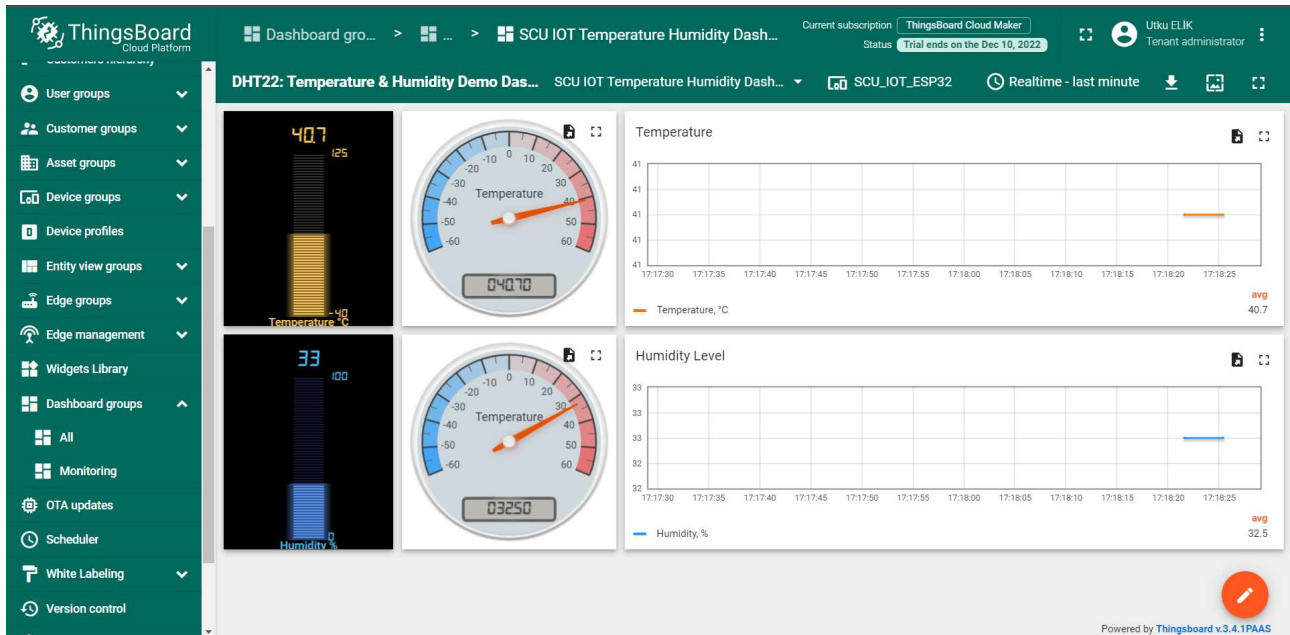
  // Uploads new telemetry to ThingsBoard
  // See https://thingsboard.io/docs/reference/telemetry-api/#send-data
  // for more details
  //dht.readHumidity();
  // Read temperature as Celsius (the default)
  //dht.readTemperature();

  // Send data to ThingsBoard
  tb.sendTelemetryPoint("Temperature", dht.readTemperature());
  tb.sendTelemetryPoint("Humidity", dht.readHumidity());
}

// Writing at 0x0006c000... (96 %)
// Writing at 0x00070000... (100 %)
// Wrote 707952 bytes (405064 compressed) at 0x00010000 in 6.7 seconds (effective 839.2 kbit/s)...
// Hash of data verified.
// Compressed 3072 bytes to 128...
// Writing at 0x00008000... (100 %)
// Wrote 3072 bytes (128 compressed) at 0x00008000 in 0.0 seconds (effective 1536.0 kbit/s)...
// Hash of data verified.
// Leaving...
// Hard resetting via RTS pin...
```

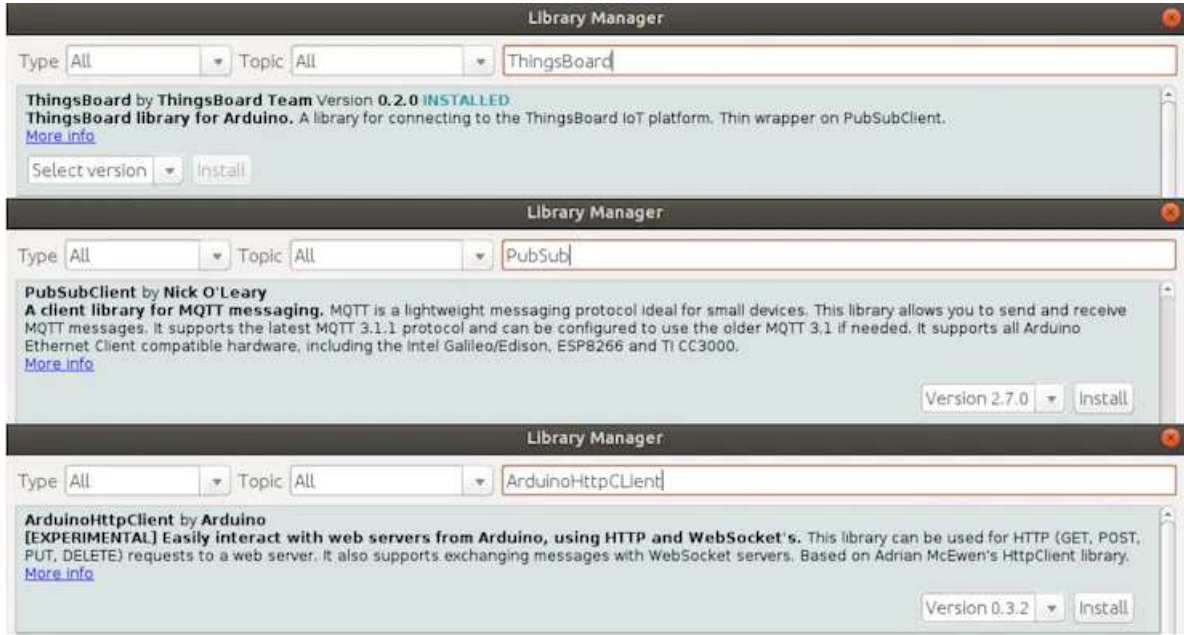
Şekil-12 ESP32 Data Sending to ThingsBoard

- Göndermiş olduğumuz sensör bilgileri oluşturmuş olduğumuz dashboard üzerinde Time-Chart, Gauge ve Level-Bar modülleri üzerinde görebilmekteyiz.



Şekil-13 Sensor Monitoring via ThingsBoard Dashboard

3. ESP32 IoT Geliştirme Kartının Kodlanması



Şekil-14 Gerekli Kütüphaneler-1



Şekil-15 Gerekli Kütüphaneler-2

HDC1080 lib: https://github.com/closedcube/ClosedCube_HDC1080_Arduino

ESP32 Kodları: https://github.com/Rexoes/ESP32_ThingsBoard_SensorMonitoring