



---

# IOT POREJESİ

---

IoT Tabanlı Akıllı Minder

Ad Soyad : Mustafa Can AVCI

Konu: Duruş Bozukluğu Tespiti ve Çalışma Takip Asistanlığı



16 ŞUBAT 2025

## BÖLÜM 1:PROJENİN AMACI

Günümüzde masa başı çalışanların ve öğrencilerin yaşadığı en büyük sorunlardan biri olan duruş bozuklukları ve hareketsizliğe bağlı sağlık problemleri yaşamaktadır. Bu proje, kullanıcının oturma alışkanlıklarını analiz ederek; yanlış duruşlarda (sağa/sola yatık, kambur) anlık geri bildirim verir ve uzun süreli hareketsizlikte kullanıcıyı uyararak daha sağlıklı ve uzun saatler çalışma imkanı yaratır.

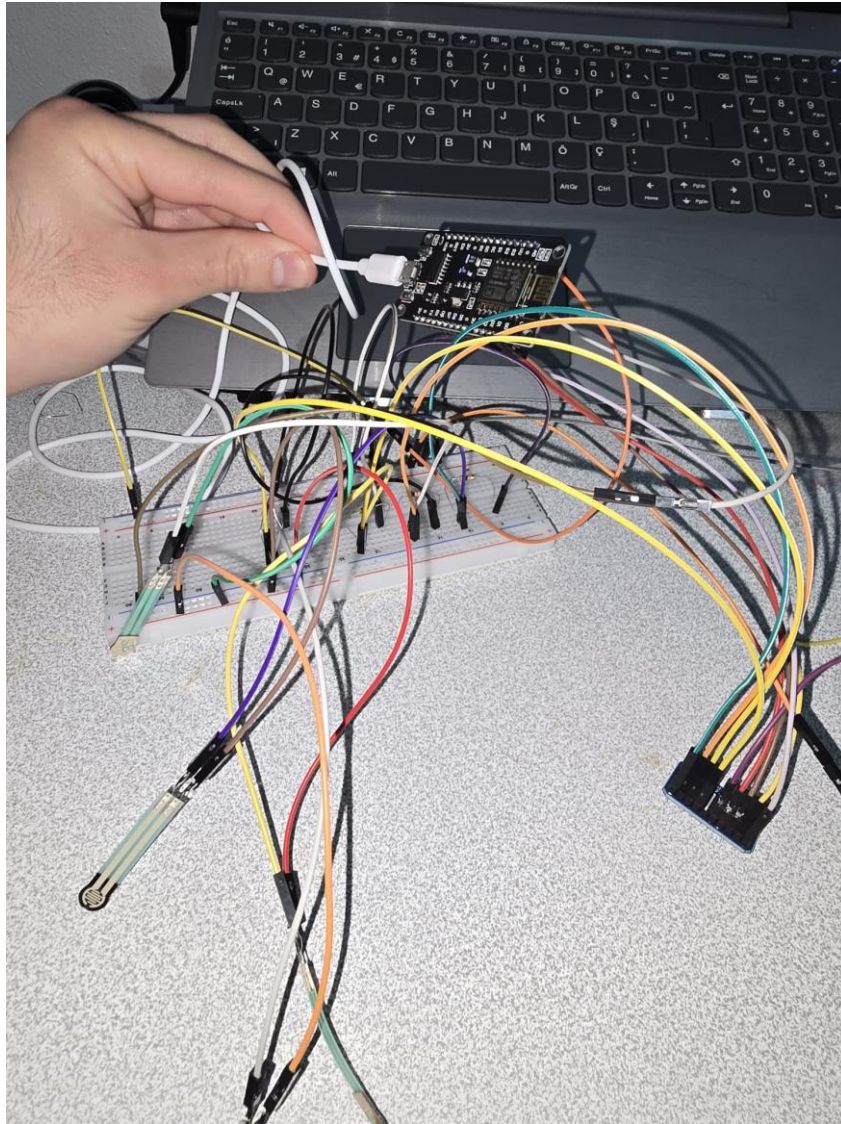
## BÖLÜM 2: DONANIM VE DEVRE TASARIMI

Sistemin donanım katmanı, yüksek hassasiyetli veri toplama ve kablosuz veri iletimi üzerine kurgulanmıştır. Kullanılan temel bileşenler şunlardır:

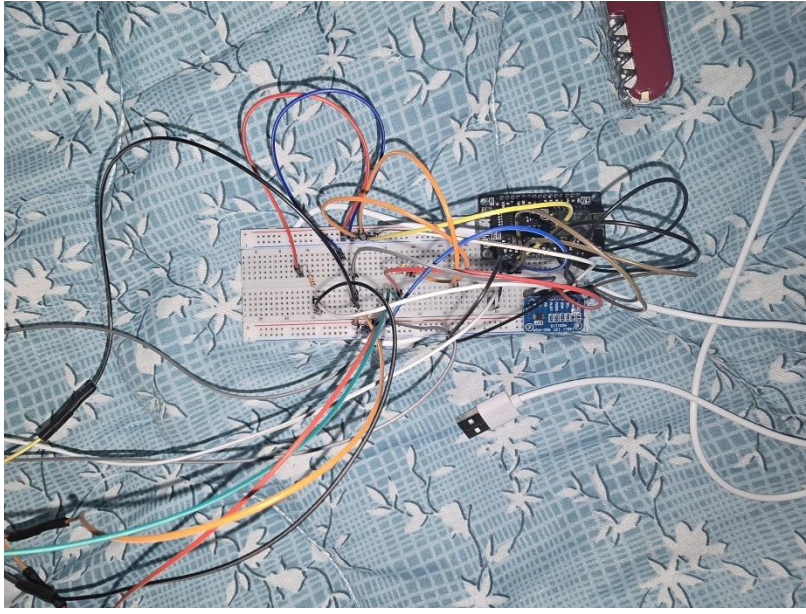
- Ana Kontrolcü (NodeMCU ESP8266):** Üzerinde bulunan Wi-Fi modülü sayesinde sensör verilerini internete aktaran sistemin beynidir.
- Hassas Veri Dönüştürücü (ADS1115 ADC):** 16-bit çözünürlüğü ile basınç sensörlerinden gelen analog verileri, standart mikrodenetleyicilere göre çok daha yüksek hassasiyetle dijital veriye dönüştürür.
- Basınç Sensörleri (FSR):** Mindere yerleştirilen 4 adet kuvvet algılayıcı sensör, kullanıcının ağırlık dağılımını ölçmek için kullanılır.
- Uyarı Birimi (Aktif Buzzer):** Belirlenen hareketsizlik süreleri aşıldığında kullanıcıya fiziksel sesli uyarı verir.
- Güç Kaynağı:** Sistem, taşınabilirlik ve kullanım kolaylığı açısından standart bir USB güç kaynağı (Powerbank veya Adaptör) ile beslenmektedir.

Bileşen	Pin	Bağlanacağı Yer (NodeMCU / Güç)
ADS1115	VDD	3.3V (veya 5V)
ADS1115	GND	GND
ADS1115	SCL	D1 (GPIO 5)
ADS1115	SDA	D2 (GPIO 4)

Bileşen	Pin	Bağlanacağı Yer (NodeMCU / Güç)
FSR 1 (Sol Ön)	Bir Uç	ADS1115 A0 Pini
FSR 2 (Sağ Ön)	Bir Uç	ADS1115 A1 Pini
FSR 3 (Sol Arka)	Bir Uç	ADS1115 A2 Pini
FSR 4 (Sağ Arka)	Bir Uç	ADS1115 A3 Pini
Tüm FSR'ler	Diğer Uç	3.3V ve 10kΩ Direnç ile GND (Gerilim Bölücü)



Şekil 1 burada projenin test yapılırken ki devre hali



## BÖLÜM 3: YAZILIM VE ALGORİTMA MANTIĞI

Sistemin yazılım altyapısı, nesnelerin interneti (IoT) prensiplerine uygun olarak C++ dili ile Arduino IDE ortamında geliştirilmiştir. Yazılım, ham veriyi işleme, karar verme ve buluta iletme süreçlerini kapsayan hibrit bir algoritma yapısına sahiptir.

### 3.1. Hassas Veri Okuma ve Normalizasyon

Sistem, standart 10-bitlik analog girişler yerine, 16-bitlik ADS1115 ADC modülünü kullanarak 4 farklı noktadan gelen verileri milimetrik hassasiyetle okur.

- **Turbo Mod (Gain Control):** Kod içerisinde `ads.setGain(GAIN_TWO)` yapılandırması kullanılarak sinyal kazancı artırılmış, böylece en hafif basınç değişiklikleri dahi tespit edilebilir hale getirilmiştir.
- **Normalize Etme:** Sensörlerden gelen 0-32767 arası ham veriler, `map` fonksiyonu ile %0-100 arasına çekilerek standart bir analiz birimine dönüştürülmüştür.

### 3.2. Pozisyon Algılama Algoritması

Kullanıcının duruşu, 4 sensörün (Sol Arka, Sağ Arka, Sol Ön, Sağ Ön) birbirine göre matematiksel farkı (Delta) üzerinden analiz edilir.

- **Dinamik Eşik (TOLERANS):** Yazılımda tanımlanan `TOLERANS = 30` değeri sayesinde, sağ ve sol taraf arasındaki fark bu eşiği geçtiği anda sistem "Sağa Yatık" veya "Sola Yatık" kararını verir.
- **Postür Sınıflandırması:** Eğer ön sensörlerin toplamı arka sensörlerden fazlaysa "Kambur", tam tersi durumda ise "Rahat/Yaslanmış" duruşu tespit edilir. Farklar minimal ise "Dik Oturuş" olarak kaydedilir.

### 3.3. Gelişmiş "Kalkma" ve "Debounce" Mantığı

Basınç sensörlerinde yaygın olarak görülen veri sıçramalarını (noise) engellemek için iki aşamalı bir koruma mekanizması kurgulanmıştır:

- **Yazılımsal Debounce:** Sensörlerden anlık gelen yanlış "0" verileri, sistemin hemen "Kalktı" uyarısı vermesine sebep olmaz.
- **Zaman Bazlı Doğrulama (Tolerans):** Kodda yer alan `KALKMA_TOLERANS_SURESI` (3000ms) değişkeni sayesinde, sinyal kesilse dahi sistem 3 saniye boyunca verinin gelmesini bekler. Eğer bu süre sonunda hala veri yoksa "Kullanıcı Gerçekten Kalktı" onayı verilir. Bu sayede kullanıcının sandalye üzerindeki zıplama veya kıpırdama hareketleri "yanlış alarm" üretmez.

### 3.4. Kritik Süre ve Sağlık Uyarısı

Algoritma sadece pozisyonu değil, aynı zamanda sağlık parametrelerini de denetler. `HAREKETSIZLIK_LIMITI` (30 saniye) aşıldığında, sistem otomatik olarak bir "Mola Ver" döngüsü başlatır. Bu döngü hem Telegram üzerinden dijital bir uyarı, hem de Buzzer üzerinden fiziksel bir sesli ikaz tetikleyerek kullanıcıyı uyarır.

```
94 // --- FIREBASE ---
Output Serial Monitor X
Not connected. Select a board and a port to connect automatically.

.....
WiFi Baglandı.
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
Kullanıcı oturdu.
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
SA: 0 | OA: 100 | SO: 0 | OO: 0 -> ORT: 25
```

Şekil 2 burada serial monitördeki bastırılan sensörlerden gelen yüzdelerik basınç değeri

## BÖLÜM 4: FIREBASE REALTIME DATABASE ENTEGRASYONU

Sistemin bulut tabanlı veri yönetimi için Google'ın sunduğu **Firestore Realtime Database** çözümü tercih edilmiştir. Bu entegrasyon, fiziksel cihaz (minder) ile web/mobil arayüz arasında köprü görevi görerek verilerin **0.5 - 1 saniye** gibi düşük gecikme süreleriyle senkronize edilmesini sağlar.

### 4.1. Veri Yapısı ve JSON Hiyerarşisi

Firestore üzerinde "NoSQL" mimarisine uygun, ilişkisel olmayan bir JSON ağacı oluşturulmuştur. Cihazdan gönderilen veriler /KoltukDurumu ana düğümü altında şu alt parametrelerle dinamik olarak güncellenmektedir:

- **Analiz:** Kullanıcının mevcut oturuş pozisyonu (Dik, Sağa/Sola Yatık, Kambur vb.) metinsel olarak tutulur.
- **Durum:** Koltuğun doluluk bilgisini (Dolu/Boş) ifade eden ana değişkendir.
- **OrtalamaYük:** 4 adet basınç sensöründen gelen ham verilerin normalize edilmiş ortalama değeridir.
- **Tarih/Zaman Damgası:** (Opsiyonel) Verinin gönderildiği anlık zaman bilgisini içerir.

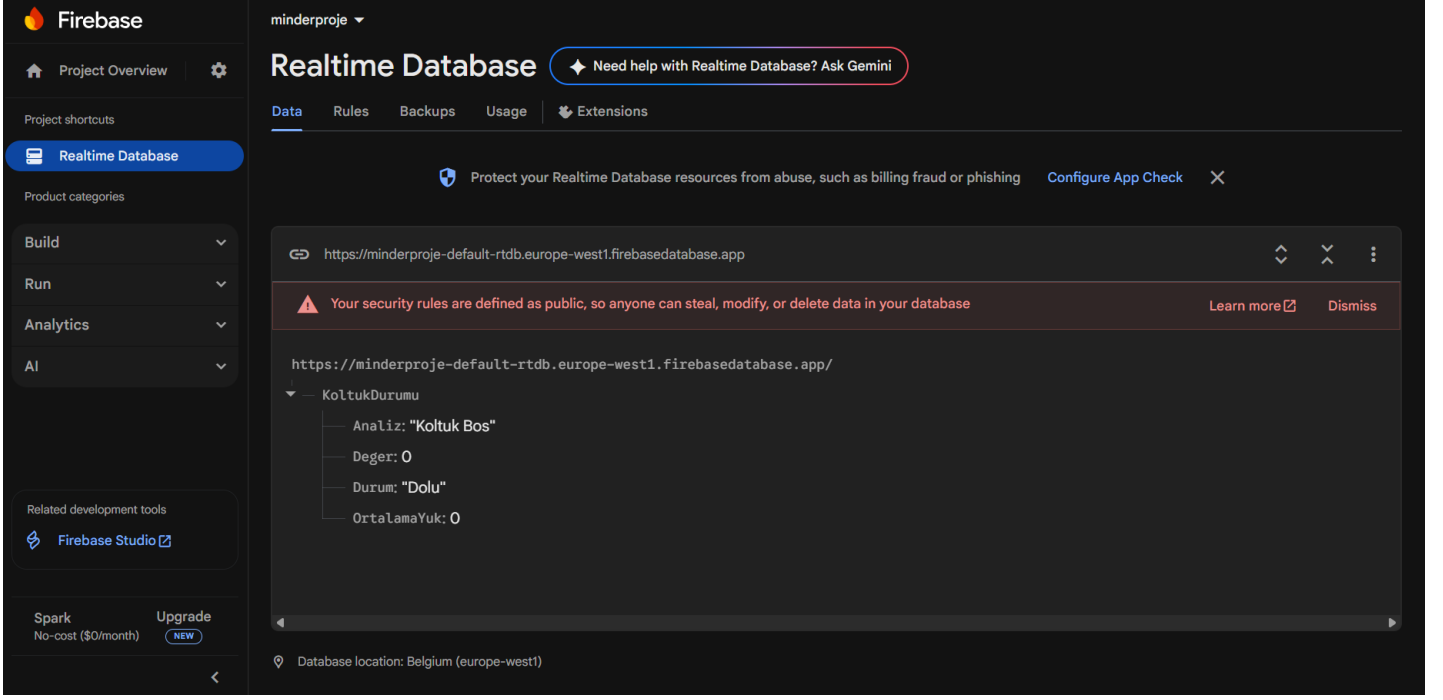
### 4.2. Gerçek Zamanlı Veri Akışı ve Bant Genişliği Yönetimi

Yazılım içerisinde Firestore veri trafiği, işlemciyi yormamak ve veritabanı kotasını verimli kullanmak adına belirli bir mantığa oturtulmuştur:

- **Koşullu Güncelleme:** Firestore.setString ve Firestore.setInt komutları, ana döngüden bağımsız olarak FIREBASE\_SURESI (1000ms) değişkeni ile sınırlandırılmıştır. Bu sayede saniyede sadece 1 kez paket gönderilerek sistem kararlılığı artırılmıştır.
- **Kesintisiz Bağlantı (Keep-Alive):** Firestore.reconnectWiFi(true) komutu ile cihazın internet bağlantısı kopsa dahi, bağlantı sağlandığı anda veritabanı ile otomatik olarak el sıkışması (handshake) sağlanır.

### 4.3. Güvenlik ve Kimlik Doğrulama

Projenin test ve geliştirme aşamasında hızlı veri iletimi için **Legacy Token (Firestore Auth)** yöntemi kullanılmıştır. Bu yöntemle, FIREBASE\_HOST ve FIREBASE\_AUTH anahtarları aracılığıyla cihazın veritabanına doğrudan ve güvenli bir şekilde veri yazması sağlanmıştır. Ayrıca Firestore tarafında "Public" olarak tanımlanan kurallar, cihazdan gelen anlık veri akışını doğrulamak amacıyla yapılandırılmıştır.



Şekil 3 : Kullanıcının "Sola Yatık" pozisyonunda oturduğu andaki Firebase Realtime Database

## 5 bölüm: Telegram Bot Entegrasyonu

Sistemin kullanıcı ile olan interaktif iletişimini sağlamak ve kritik durumlarda anlık uyarı iletmek amacıyla **Telegram Bot API** entegrasyonu gerçekleştirilmiştir. Bu sayede sistem, bilgisayar başında olunmasa dahi mobil cihazlar üzerinden takip edilebilir hale getirilmiştir.

### 5.1. Bot Yapılandırması ve Güvenli İletişim

Sistem, Telegram sunucuları ile haberleşmek için UniversalTelegramBot kütüphanesini ve güvenli veri iletimi için WiFiClientSecure protokolünü kullanmaktadır.

- **Kimlik Doğrulama:** Her kullanıcıya özel tanımlanan BOT\_TOKEN ve mesajın iletileceği hedefi belirleyen CHAT\_ID değişkenleri kod içerisine gömülerek sadece yetkili kullanıcıya bildirim gitmesi sağlanmıştır.

- **Güvenlik Sertifikası:** ESP8266 kartının Telegram sunucularına SSL üzerinden bağlanabilmesi için `clientSecure.setInsecure()` komutu ile sertifika doğrulama süreçleri optimize edilmiştir.

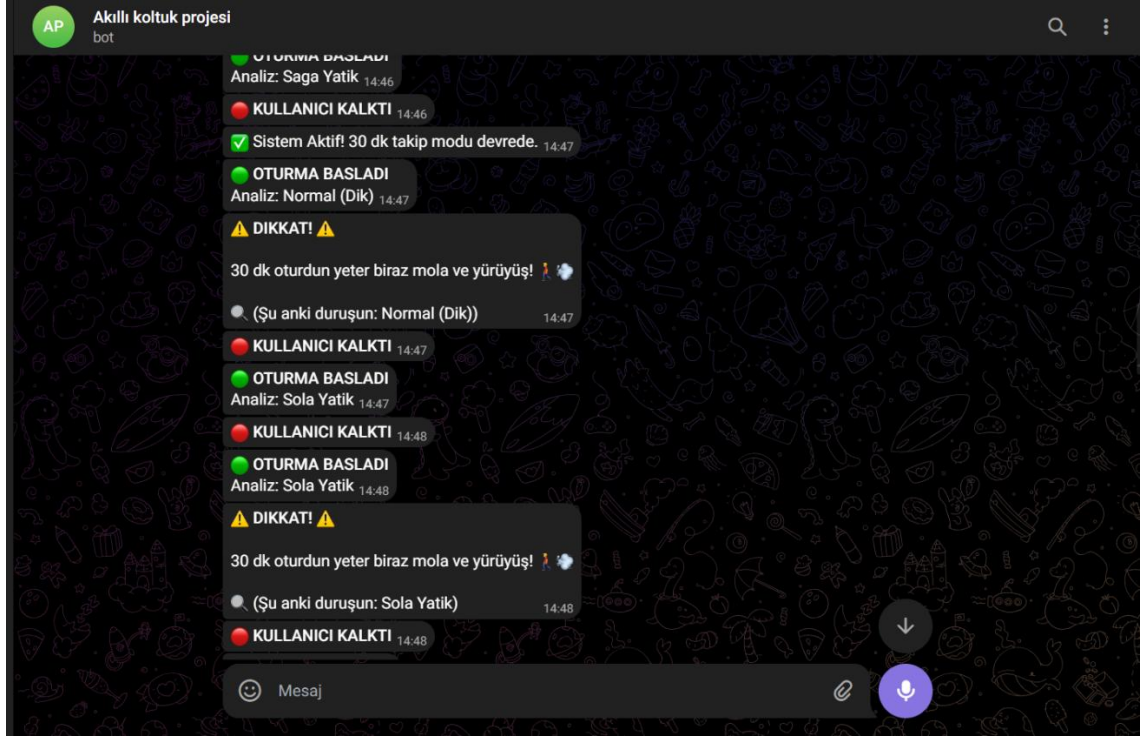
## 5.2. Bildirim Senaryoları ve Akıllı Mantık

Kod içerisinde Telegram bildirimleri, kullanıcıyı gereksiz mesaj yağmuruna tutmayacak bir "durum bazlı" (event-driven) mantıkla kurgulanmıştır:

- **Sistem Başlatma:** Cihaz ilk açıldığında ve internete bağlandığında kullanıcıya *"Sistem Aktif! Tolerans Modu Devrede"* mesajı gönderilerek cihazın hazır olduğu teyit edilir.
- **Oturma Bildirimi:** Kullanıcı mindere oturduğu anda sistem bunu algılar ve Markdown formatında zengin içerikli bir mesaj ile pozisyon analizini (Dik, Yatık vb.) bildirir.
- **Mola ve Sağlık Uyarısı:** HAREKETSİZLİK\_LIMITI (30 saniye) aşıldığında, sistem kullanıcının sağlığını korumak amacıyla *"DİKKAT! 30 saniye doldu! Mola ver."* uyarısını bir alarm ikonuyla birlikte iletir.
- **Kalkma Bildirimi:** Kullanıcı kalktığında, sistem bunu anlık olarak değil, kodda tanımlanan **3 saniyelik tolerans süresi** sonunda doğrular ve ardından "Kullanıcı Kalktı" bilgisini gönderir.

## 5.3. Kullanıcı Deneyimi ve Görsellik

Bildirimlerde kullanılan Markdown desteği sayesinde mesajlar kalın puntolar (bold) ve emojilerle desteklenerek okunabilirlik artırılmıştır. Bu özellik, acil durum uyarılarının (kırmızı buton veya uyarı ikonu gibi) kullanıcı tarafından fark edilmesini kolaylaştırır.



Şekil 4 Sistem tarafından kullanıcıya gönderilen gerçek zamanlı Telegram analiz ve mola uyarı mesajları.

## BÖLÜM 6: VERİ ANALİZİ VE GRAFİKLER (THINGSPEAK)

Kullanıcının oturma alışkanlıklarını uzun vadede analiz edebilmek için veriler ThingSpeak platformuna gönderilmektedir.

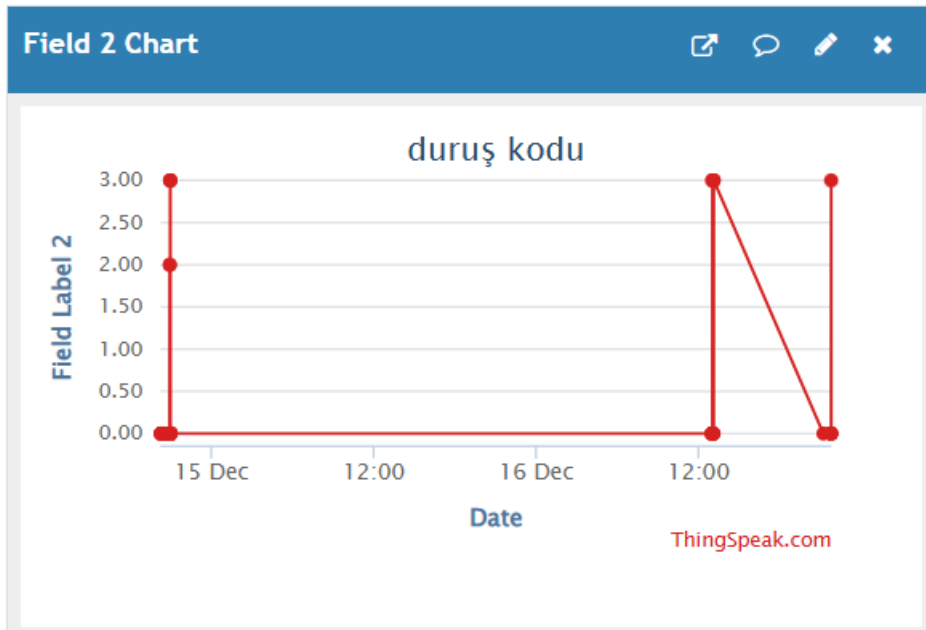
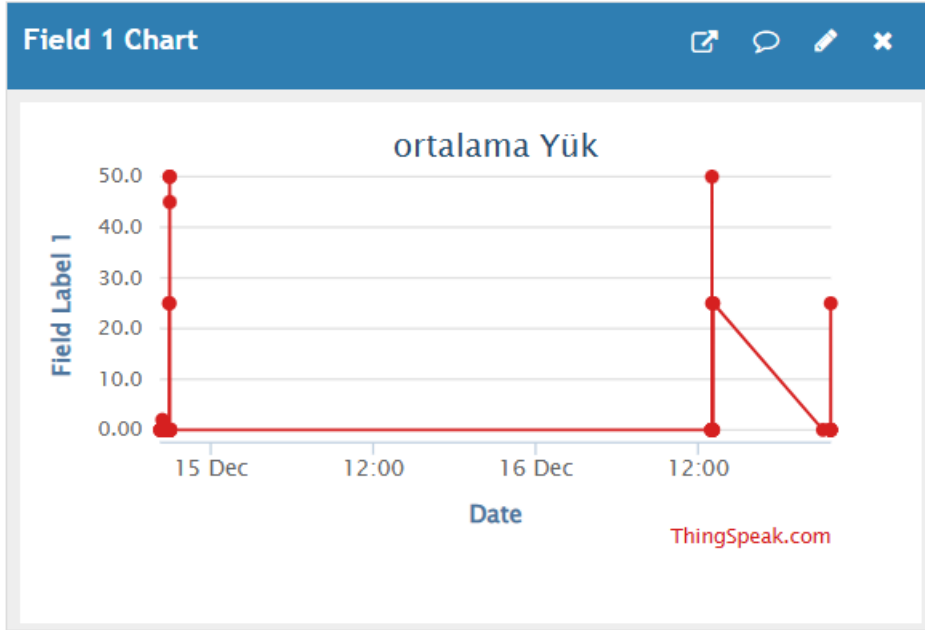
- **Grafik 1 (Field 1):** Ortalama yük miktarını gösterir. Grafikteki ani yükselmeler oturma eylemini, sıfıra düşüşler ise kalkma eylemini temsil eder.
- **Grafik 2 (Field 2):** Duruş kodlarını (1: Dik, 2: Sağa Yatık vb.) zaman çizelgesi üzerinde gösterir.

### 6.1. Grafiksel İzleme ve Kayıt

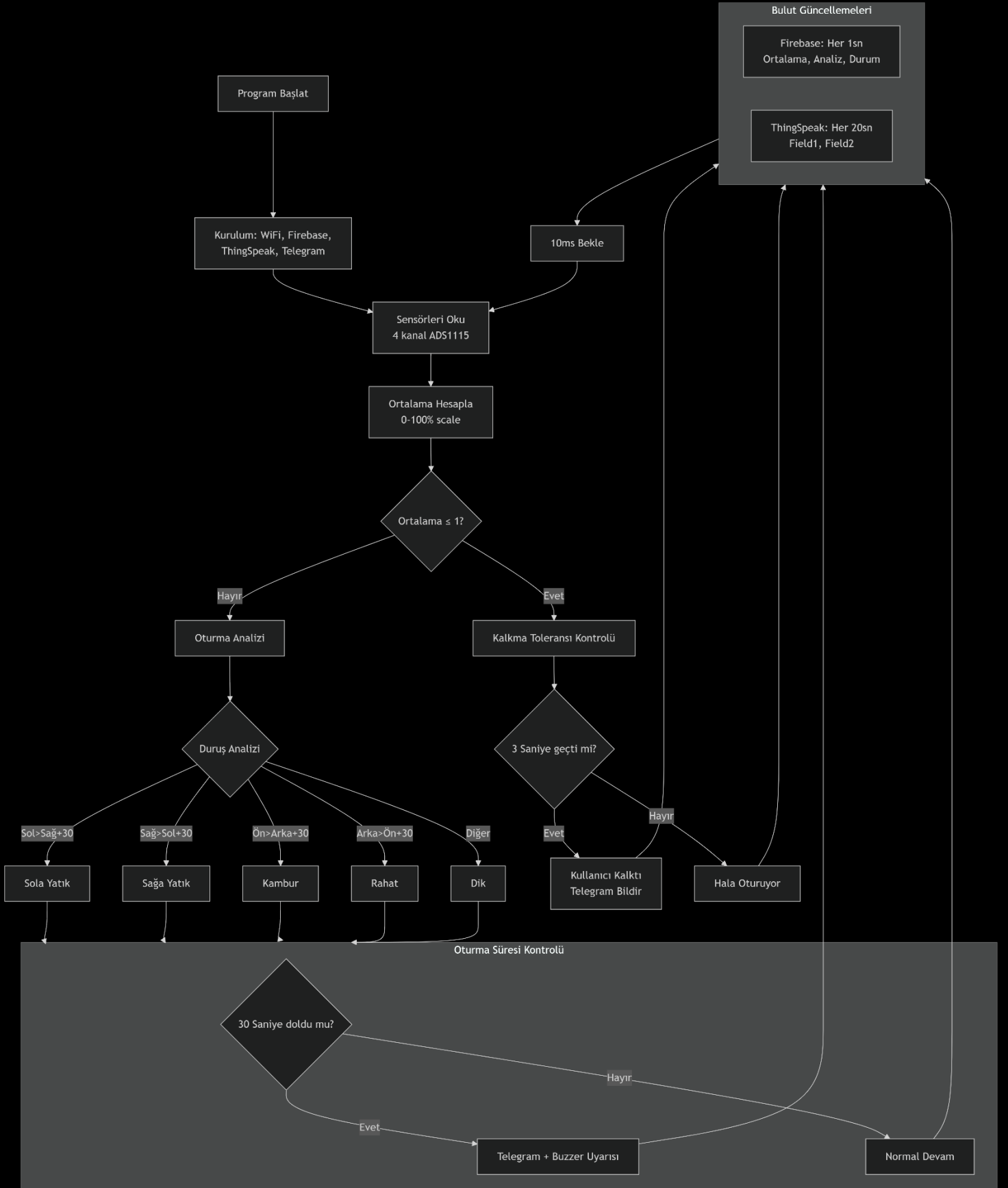
- **Görsel Analiz:** Sensörlerden gelen veriler ThingSpeak paneli üzerinde otomatik olarak 2D çizgi grafiklerine dönüştürülür. Bu sayede kullanıcının gün içindeki oturma alışkanlıkları geriye dönük olarak incelenebilir.
- **Zamanlama:** Veri iletimi, platformun ücretsiz katman limitlerine uygun olarak THINGSPEAK\_SURESI (20 saniye) aralıklarla gerçekleştirilir.

## 6.2. Veri Güvenliđi

Veri gönderimi sırasında myChannelNumber ve myWriteAPIKey parametreleri kullanılarak, verilerin sadece tanımlı kanala güvenli bir şekilde yazılması sağlanmıştır. İşlem sonucunda seri port ekranına dönen "200" kodu, verinin buluta başarıyla ulaştığını teyit eder.



## BÖLÜM 7: burada kodun akış diyagramı çizili



## BÖLÜM 9: Projenin Business Canvas Modeli

Duruş Bozukluğu Tespiti ve Çalışma Takip Asistanlığı				
TEMEL ORTAKLIKLAR	TEMEL FAALİYETLER	DEĞER ÖNERİLERİ	MÜŞTERİ İLİSKİLERİ	Customer Segments
<ul style="list-style-type: none"> <li>KİMLERLE İSBİRLİĞİ YAPIYORUZ?</li> <li><b>Bulut Sağlayıcılar:</b> Google Firebase (Veritabanı), ThingSpeak (Analiz), Telegram (İletişim).</li> <li><b>Bulut Sağlayıcılar:</b> Google Firebase (Veritabanı), ThingSpeak (Analiz), Telegram (İletişim).</li> <li><b>Donanım Tedarikçileri:</b> Elektronik bileşen sağlayıcıları.</li> <li><b>Sağlık Danışmanları:</b> (Gelecek planı) Duruş algoritmalarını doğrulamak için fizyoterapistler.</li> </ul>	<ul style="list-style-type: none"> <li>BİZİM SUREKLİ YAPMAMIZ GEREKEN İŞLER NELER?</li> <li><b>Ar-Ge ve Prototip Geliştirme:</b> Sensör verilerinin işlenmesi ve kalibrasyonu.</li> <li><b>Veri Analizi:</b> Kullanıcıdan gelen basınç verilerinin anlamlı sağlık bilgisine dönüştürülmesi.</li> </ul> <div>Temel Kaynaklar</div> <ul style="list-style-type: none"> <li>Bu işi yapmak için elimizde ne var?</li> <li>Donanım Bileşenleri: NodeMCU (ESP8266), FSR Sensörleri, ADS1115 ADC Modülü.</li> <li>Yazılım Algoritmaları: Duruş tespiti yapan C++ kodu ve Debounce algoritmaları.</li> <li>Veri Tabanı: Firebase ve ThingSpeak bulut entegrasyonu.</li> </ul>	<ul style="list-style-type: none"> <li>BİZ MÜŞTERİLERE NE SUNUYORUZ HANGİ SORUNU ÇÖZÜYÜZ?</li> <li><b>Gerçek Zamanlı Duruş Düzeltme:</b> Sensörler sayesinde kullanıcının oturuş pozisyonunu (sağa/sola yatık, kambur) anlık olarak algılayıp uyararak duruş bozukluğunu engeller.</li> <li><b>Aktif Sağlık Asistanı:</b> Sadece destek sağlamaz; 30 dakikayı aşan hareketsizlik durumunda Telegram üzerinden bildirim göndererek kullanıcıyı hareket etmeye teşvik eder.</li> <li><b>Düşük Maliyetli Çözüm:</b> Pahalı ergonomik ofis sandalyeleri satın almak yerine, mevcut koltuğu "akıllı" hale getiren ekonomik ve taşınabilir bir teknoloji sunar.</li> <li><b>Veriye Dayalı Takip:</b> ThingSpeak entegrasyonu ile kullanıcının oturma alışkanlıklarını ve gelişimini grafikler üzerinden raporlar.</li> <li><b>Omurga Sağlığını Koruma:</b> Ofis çalışanları ve öğrencilerde yaygın görülen bel/sırt ağrılarını ve duruşa bağlı sağlık sorunlarını önlemeye yardımcı olur.</li> </ul>	<ul style="list-style-type: none"> <li>Müşteriye ne sunuyoruz ve hangi sorununu çözüyoruz?</li> <li><b>Anlık Duruş Düzeltme:</b> Yanlış oturma (sağa/sola yatık, kambur) durumunda anında bildirim göndererek farkındalık yaratır.</li> <li><b>Sağlık Koruma:</b> Uzun vadeli omurga ve bel ağrılarının önüne geçilmesine yardımcı olur.Hareketsizlik Uyarısı: 30 dakikadan fazla sabit oturulursa mola vermesi gerektiğini hatırlatır.Düşük Maliyetli Çözüm: Pahalı ergonomik koltuklar yerine, mevcut koltuğu akıllı hale getiren ekonomik bir donanım sunar.</li> <li><b>Veri Takibi:</b> Kullanıcının oturma alışkanlıklarını grafiklerle raporlar (ThingSpeak entegrasyonu).</li> </ul> <div>KANALLAR</div> <ul style="list-style-type: none"> <li>Müşteriye nasıl ulaşıyoruz?</li> <li>Telegram Botu: Kullanıcıya doğrudan uyarı ve bildirim göndermek için.</li> <li>IoT Platformları (Web/Mobil): Veri analizi ve grafiklerin gösterimi için ThingSpeak arayüzü.</li> <li><b>E-Ticaret Siteleri:</b> (Gelecek planı) Ürünün son kullanıcıya satışı.</li> </ul>	<ul style="list-style-type: none"> <li>Ürün kimin için tasarlandı?</li> <li><b>Masa Başı Çalışanlar:</b> Günün büyük kısmını bilgisayar başında geçiren beyaz yakalılar.Öğrenciler: Uzun süre ders çalışan ve duruş bozukluğu riski taşıyan öğrenci grubu.</li> <li><b>Kurumsal Şirketler:</b> Çalışanlarının ofis ergonomisine ve sağlığına önem veren İK departmanları.</li> <li><b>Omurga Sağlığı Sorunu Yaşayanlar:</b> Fizik tedavi gören veya duruşunu düzeltmek isteyen bireyler.</li> </ul>
MALİYET YAPISI		GELİR AKIŞI		
<ul style="list-style-type: none"> <li>Giderlerimiz neler?</li> <li>Donanım Maliyetleri: Sensörler, mikrodenetleyici, kablolu ve güç kaynağı maliyetleri.</li> <li>Sunucu/Bulut Giderleri: İleride artan veri trafiği için platform üyelik ücretleri.</li> <li>Prototipleme Giderleri: Breadboard, jumper kablolar ve test süreçleri.</li> </ul>		<ul style="list-style-type: none"> <li>Donanım Satış: Akıllı minder cihazının (NodeMCU + Sensör kiti) doğrudan satışı.</li> <li>Premium Abonelik: (Gelecek planı) Detaylı sağlık analizleri ve geçmişe dönük uzun vadeli raporlar için aylık üyelik.</li> <li>Kurumsal Satış: Ofislere toplu satış anlaşmaları.</li> </ul>		

## BÖLÜM 10: Maliyet Tablosu

Ürün adı	Adet	Birim Fiyat (TL)	Toplam Fiyat (TL)
Çift taraflı silikon bant	2 adet	49,14	98,28
ADS 1115 16-bit	1 adet	116,10	116,10
FSR400	4 adet	129,00	516,00
Minder	1 adet	297,52	297,52
Direnç	4 adet	1	4-5
Board ve diğer devre malzemeleri	1 adet	250	250
Toplam	-	-	1281,9

## SONUÇ VE DEĞERLENDİRME

Bu proje kapsamında, günümüzde yaygınlaşan masa başı çalışma düzeninin getirdiği duruş bozukluklarını tespit etmek ve önlemek amacıyla Nesnelerin İnterneti (IoT) tabanlı bir "Akıllı Minder Sistemi" geliştirilmiştir.

**Donanım ve Teknik Çözümler:** Projenin donanım aşamasında, maliyet ve performans etkinliği göz önünde bulundurularak NodeMCU ESP8266 mikrodenetleyicisi tercih edilmiştir. ESP8266 modülünün tek bir analog girişe sahip olması kısıtı, sisteme 16-bit çözünürlüklü ADS1115 ADC modülü entegre edilerek profesyonel bir seviyede aşılmıştır. Bu sayede koltuğun dört farklı noktasına yerleştirilen sensörlerden, standart analog girişlere oranla çok daha hassas ve gürültüden arındırılmış veri okuması başarıyla gerçekleştirilmiştir.

**Yazılım ve Entegrasyon:** Yazılım aşamasında, geliştirilen "Kalkma Tolerans Algoritması" ile sensörlerdeki anlık veri dalgalanmalarının (noise) önüne geçilmiş ve sistemin kararlılığı artırılmıştır. Elde edilen analiz sonuçları çok kanallı bir IoT mimarisi ile servis edilmiştir:

- Firebase Realtime Database üzerinden anlık veri takibi sağlanmış,
- ThingSpeak ile verilerin istatistiksel analizi buluta kaydedilmiş,
- Telegram Bot entegrasyonu ile kullanıcıya mobil üzerinden interaktif sağlık uyarıları iletilmiştir.

**Yapılan Testler ve Gözlemler:** Sistem üzerinde yapılan testler sonucunda şu çıktılar elde edilmiştir:

- Sistemin, kullanıcının oturuş ve kalkış anlarını 3 saniyelik hata payı (debounce) ile hatasız algıladığı,
- Duruş bozukluklarının (sağa/sola yatık, kambur vb.) ağırlık merkezi hesaplamalarıyla doğru tespit edildiği,
- 30 saniyelik hareketsizlik durumunda Buzzer ve Telegram üzerinden verilen uyarıların kullanıcı farkındalığını artırdığı gözlemlenmiştir.

**Sonuç:** Sonuç olarak, geliştirilen bu prototip ile omurga sağlığını korumaya yönelik, düşük maliyetli, taşınabilir ve geliştirilebilir bir sağlık teknolojisi çözümü ortaya konulmuştur. Gelecek çalışmalarda, makine öğrenmesi algoritmaları eklenerek kullanıcının uzun vadeli duruş alışkanlıklarının analiz edilmesi hedeflenmektedir.

Projenin bitmiř halinin fotoęrafları

