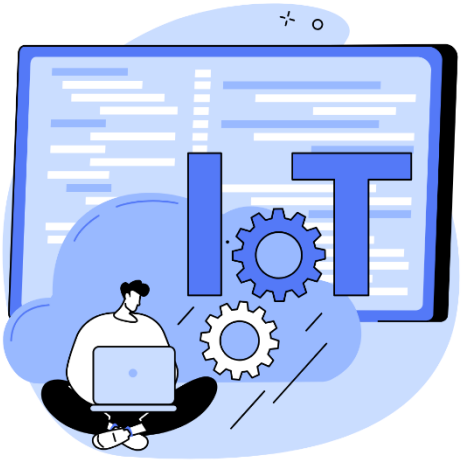
RASPBERRY Pİ TABANLI AKILLI HAVA KALİTESİ İZLEME VE ERKEN UYARI SİSTEMİ



Üniversite adı: İskenderun Teknik Üniversitesi

Fakülte / Bölüm: Bilgisayar Mühendisliği

Bitirme Projesi Başlığı: Raspberry Pi Tabanlı Akıllı Hava Kalitesi İzleme ve Erken Uyarı Sistemi

Öğrenci adı / numarası: Hüseyin Üçer 232503218

Danışman adı: Halil İbrahim Okur

Teslim tarihi:23.12.2025

PROJE AMACI

1.1 Projenin Amacı

Bu bitirme projesinin temel amacı, Raspberry Pi tabanlı akıllı bir hava kalitesi izleme, analiz ve erken uyarı sistemi geliştirmektir. Günümüzde hava kirliliği, özellikle kapalı alanlarda yaşayan veya çalışan bireyler için önemli bir sağlık ve güvenlik problemi oluşturmaktadır. Bu nedenle ortam havasının sürekli izlenmesi, elde edilen verilerin analiz edilmesi ve kritik durumlarda kullanıcıların zamanında bilgilendirilmesi büyük önem taşımaktadır.

Geliştirilen sistem; ortamdan elde edilen sıcaklık, nem ve gaz sensör verilerini gerçek zamanlı olarak toplayarak bu verileri kayıt altına almakta, geçmiş ölçümler üzerinde analiz yapılmasına olanak sağlamaktadır. Bununla birlikte sistem, yalnızca ham verileri kullanıcıya sunmakla kalmayıp, makine öğrenmesi tabanlı bir analiz mekanizması sayesinde hava kalitesini “İyi (Good)”, “Orta (Moderate)” ve “Kötü (Bad)” olmak üzere sınıflandırmaktadır.

Projenin bir diğer önemli amacı ise, elde edilen ölçümlerin harici çevresel verilerle karşılaştırılmasıdır. Bu kapsamda OpenWeather API kullanılarak dış ortam hava durumu ve hava kirliliği verileri sisteme entegre edilmiştir. Böylece kapalı ortamdan elde edilen sensör verileri ile dış ortam koşulları karşılaştırılabilmekte ve daha anlamlı bir değerlendirme yapılabilmektedir.

Ayrıca sistem, kritik eşik değerlerin aşılması veya makine öğrenmesi modelinin “kötü hava kalitesi” tahmini üretmesi durumunda Telegram mesajlaşma platformu üzerinden otomatik uyarı gönderecek şekilde tasarlanmıştır. Bu sayede sistem, pasif bir izleme aracı olmaktan çıkmaktadır.

PROJENİN İŞ PAKETLERİ VE AÇIKLAMALARI

2.1 İş Paketi 1 – Donanım Altyapısının Kurulması ve Sensör Seçimi

Bu iş paketinde, hava kalitesini ölçmek için kullanılacak sensörler araştırılmış ve proje gereksinimlerine uygun sensörler seçilmiştir. Yanıcı gaz, duman ve LPG benzeri gazların algılanması için MQ-2 gaz sensörü, genel hava kalitesinin değerlendirilmesi ve zararlı gazların ölçümü için ise MQ-135 gaz sensörü kullanılmıştır. Ortam koşullarının daha doğru yorumlanabilmesi için sisteme ayrıca sıcaklık ve nem sensörü eklenmiştir.

Seçilen sensörler Arduino mikrodenetleyici kartına bağlanmış ve gerekli besleme, sinyal ve toprak bağlantıları gerçekleştirilmiştir. Bu aşamada sensörlerin çalışma gerilimleri, analog çıkış karakteristikleri ve ölçüm hassasiyetleri dikkate alınmıştır. Sensörlerden elde edilen analog sinyaller Arduino tarafından okunarak dijital verilere dönüştürülmüştür.

2.2 İş Paketi 2 – Arduino Üzerinde Veri Toplama Yazılımının Geliştirilmesi

Arduino üzerinde geliştirilen yazılım sayesinde sensörlerden alınan veriler belirli zaman aralıklarında okunmuştur. Okunan sensör değerleri, seri haberleşme (UART) protokolü kullanılarak Raspberry Pi'ye aktarılmıştır. Veri iletimi sırasında sistemin esnek ve dayanıklı olması amacıyla hem CSV formatında hem de JSON formatında veri gönderimi desteklenmiştir.

Bu yapı sayesinde seri porttan gelen veriler Raspberry Pi tarafında kolayca ayrıştırılabilmiş, hatalı veya eksik veriler filtrelenerek sistemin kararlılığı artırılmıştır. Arduino tarafındaki yazılım, sistemin uzun süre kesintisiz çalışabilmesi göz önünde bulundurularak tasarlanmıştır.

2.3 İş Paketi 3 – Raspberry Pi Üzerinde Veri Alma ve Kayıt Sistemi

Raspberry Pi üzerinde Python programlama dili kullanılarak seri porttan gelen veriler sürekli olarak dinlenmiştir. PySerial kütüphanesi yardımıyla Arduino'dan gelen veriler okunmuş, doğrulama işlemleri yapılmış ve geçerli ölçümler sensor_log.csv dosyasına zaman bilgisi ile birlikte kaydedilmiştir.

Bu yapı sayesinde sistem yeniden başlatılsa dahi daha önce toplanan veriler korunmakta ve geçmiş ölçümler üzerinde analiz yapılabilmektedir. CSV tabanlı veri kaydı, hem makine öğrenmesi modelinin eğitilmesinde hem de web arayüzünde grafiksel gösterim yapılmasında kullanılmıştır.

2.4 İş Paketi 4 – OpenWeather API Entegrasyonu

Sistemin yalnızca yerel sensör verilerine bağlı kalmaması amacıyla OpenWeather API projeye entegre edilmiştir. Bu API aracılığıyla dış ortam sıcaklık, nem ve hava kirliliği verileri çekilmiştir. Elde edilen bu veriler, Raspberry Pi üzerinde çalışan Python yazılımı tarafından işlenerek sistemde kullanılabilir hale getirilmiştir.

OpenWeather API entegrasyonu sayesinde, iç ortam hava kalitesi ile dış ortam hava koşulları karşılaştırılabilmiş ve kullanıcıya daha kapsamlı bir değerlendirme sunulmuştur. Bu karşılaştırma, özellikle kapalı alanlardaki hava kalitesinin dış ortam koşullarından nasıl etkilendiğini gözlemlemek açısından önemli bir katkı sağlamıştır.

2.5 İş Paketi 5 – Makine Öğrenmesi ile Hava Kalitesi Sınıflandırması

Toplanan sensör verileri kullanılarak makine öğrenmesi tabanlı bir sınıflandırma modeli geliştirilmiştir. Bu amaçla Python programlama dili ve Scikit-learn kütüphanesi kullanılmıştır. Model; sıcaklık, nem, MQ-2 ve MQ-135 sensör değerlerini girdi olarak alarak hava kalitesini sınıflandırmaktadır.

Eğitilen model Raspberry Pi üzerinde gerçek zamanlı olarak çalışmakta ve her yeni ölçümde hava kalitesi tahmini üretmektedir. Modelin çıktıları, hem web arayüzünde gösterilmekte hem de erken uyarı sisteminde kullanılmaktadır.

2.6 İş Paketi 6 – Web Tabanlı Dashboard (Streamlit)

Kullanıcıların sistemi kolayca takip edebilmesi amacıyla Streamlit kütüphanesi kullanılarak web tabanlı bir dashboard geliştirilmiştir. Bu arayüz üzerinden kullanıcılar:

- Anlık sensör değerlerini
- Geçmiş ölçümlerin grafiksel gösterimlerini
- Makine öğrenmesi tahmin sonuçlarını
- OpenWeather API'den alınan dış ortam verilerini

gerçek zamanlı olarak görüntüleyebilmektedir.

2.7 İş Paketi 7 – Telegram Erken Uyarı Sistemi

Sistem, kritik eşik değerlerin aşılması veya makine öğrenmesi modelinin "Bad" tahmini üretmesi durumunda **Telegram botu aracılığıyla otomatik uyarı mesajları** gönderecek şekilde tasarlanmıştır. Bu sayede kullanıcılar, sisteme sürekli bakmasalar dahi olası riskler hakkında anında bilgilendirilmektedir.

Telegram uyarı sistemi, belirli bir zaman aralığında tekrar eden mesajları engellemek amacıyla **bekleme (cooldown) mekanizması** ile desteklenmiştir. Bu yapı, sistemin profesyonel ve kullanıcı dostu bir şekilde çalışmasını sağlamaktadır.

KARŞILAŞILAN ZORLUKLAR VE ÇÖZÜMLERİ

Bu bitirme projesi sürecinde donanım, yazılım ve sistem entegrasyonu aşamalarında çok sayıda teknik ve operasyonel zorlukla karşılaşılmıştır. Karşılaşılan bu problemler, projenin ilerleyişini zaman zaman yavaşlatmış olsa da mühendislik bakış açısıyla ele alınarak sistematik çözümler geliştirilmiş ve proje başarıyla tamamlanmıştır. Yaşanan başlıca zorluklar ve bu zorluklara getirilen çözümler aşağıda detaylı olarak açıklanmıştır.

1. Analog Sensör Okuma ve ADS1115 Problemi

Projede kullanılan MQ-2 ve MQ-135 gaz sensörleri analog çıkış veren sensörlerdir. Raspberry Pi üzerinde doğrudan analog giriş bulunmadığından, sensör verilerinin daha hassas okunabilmesi amacıyla ADS1115 analog–dijital çevirici (ADC) modülü kullanılması planlanmıştır. Ancak yapılan ilk testlerde ADS1115 modülünden beklenen değerler alınamamış, okunan voltaj değerlerinin sabit veya hatalı olduğu gözlemlenmiştir.

Çözüm:

Bu durumun donanımsal mı yoksa yazılımsal mı olduğunun anlaşılması için multimetre ile doğrudan voltaj ölçümleri yapılmıştır. Yapılan ölçümler sonucunda ADS1115 modülünün çıkışlarının tutarsız olduğu ve modülün arızalı olduğu kanıtlanmıştır. Bu nedenle sistem mimarisi yeniden düzenlenmiş ve analog sensör okumalarının Arduino üzerinden yapılmasına karar verilmiştir. Arduino'nun analog girişleri kullanılarak sensör verileri güvenilir şekilde okunmuş, ardından seri haberleşme ile Raspberry Pi'ye aktarılmıştır. Bu yaklaşım sistemin kararlılığını önemli ölçüde artırmıştır.

2. Arduino – Raspberry Pi Seri Haberleşme ve Veri Formatı Sorunları

Arduino'dan Raspberry Pi'ye veri aktarımı sırasında seri haberleşme problemleri yaşanmıştır. Veri paketlerinin eksik gelmesi, yanlış formatta okunması veya bağlantının kopması gibi sorunlarla karşılaşılmıştır. Ayrıca Raspberry Pi üzerinde doğru seri portun belirlenmesi süreci zaman almıştır.

Çözüm:

Bu sorun, Arduino ve Raspberry Pi tarafında baud rate ayarlarının birebir eşleştirilmesi ve veri formatının (ayraçlar, satır sonları) standart hâle getirilmesiyle çözülmüştür. Python

tarafında seri okuma işlemleri için hata yakalama mekanizmaları eklenmiş ve veri doğrulama kontrolleri uygulanmıştır.

3. Raspberry Pi Şifre ve Sistem Erişim Problemleri

Proje sürecinde Raspberry Pi cihazının kullanıcı şifresinin unutulması nedeniyle sisteme erişim sağlanamamış ve geliştirme süreci ciddi şekilde aksamıştır. Bu durum, proje ilerleyişinde zaman kaybına neden olmuştur.(ek olarak kullanılan yardımcı laptopun proje zaman takvimi içerisinde bozulması haliyle 1 haftalık bir yavaşlama kaydedilmiştir.)

Çözüm:

Sorunun çözümü için Raspberry Pi işletim sistemi yeniden yapılandırılmış, gerekli güvenlik ayarları yapılmış ve sistem baştan kurulmuştur. Kurulum sonrasında SSH ve VNC ayarları yeniden aktif edilerek uzaktan erişim sağlanmıştır. Bu süreç, sistem yönetimi ve Linux tabanlı cihazlarda kullanıcı yönetimi konusunda önemli bir deneyim kazandırmıştır.

4. Raspberry Pi Terminal ve Komut Satırı Zorlukları

Python scriptlerinin terminal üzerinden çalıştırılması, arka planda servis olarak başlatılması ve hata mesajlarının yorumlanması aşamalarında önemli zorluklar yaşanmıştır. Özellikle terminal komutlarının yanlış kullanılması, dosya yollarının hatalı verilmesi ve izin problemleri proje sürecini zorlaştırmıştır.

Çözüm:

Bu zorluklar, Linux terminal yapısı hakkında detaylı araştırmalar yapılarak ve deneme–yanılma yöntemiyle aşılmıştır. Süreç boyunca teknik forumlardan, dokümantasyonlardan ve yapay zekâ destekli kaynaklardan faydalanılmıştır. Elde edilen bilgiler sayesinde Python kodlarının doğru şekilde çalıştırılması, log alınması ve otomatik başlatılması sağlanmıştır.

5. Makine Öğrenmesi Model Entegrasyonu Problemleri

Hava kalitesi sınıflandırması için kullanılan makine öğrenmesi modelinin Raspberry Pi ortamına entegre edilmesi sırasında model dosyasının yüklenmesi, giriş verilerinin

uygun formatta modele verilmesi ve tahmin sonuçlarının doğru yorumlanması aşamalarında zorluklar yaşanmıştır.

Çözüm:

Model, eğitim ortamında test edildikten sonra joblib kütüphanesi kullanılarak Raspberry Pi'ye aktarılmıştır. Modelin beklediği giriş formatı dikkatle incelenmiş ve sensör verileri bu formata uygun şekilde düzenlenmiştir. Böylece sınıflandırma sonuçları doğru ve tutarlı hâle getirilmiştir.

6. Gerçek Zamanlı Telegram Uyarı Sistemi Problemleri

Telegram üzerinden gönderilen uyarı mesajlarının çok sık tetiklenmesi veya beklenmeyen zamanlarda gönderilmesi gibi problemler yaşanmıştır. Bu durum, kullanıcı deneyimini olumsuz etkilemiştir.

Çözüm:

Uyarı mekanizması belirli eşik değerler ve zaman aralıkları ile sınırlandırılmıştır. Aynı durum için tekrar eden mesajların gönderilmesini engelleyen kontrol yapıları eklenmiştir. Böylece sistem yalnızca kritik durumlarda kullanıcıyı bilgilendiren daha verimli bir yapıya kavuşturulmuştur.

7. Donanım Bağlantıları ve Ölçüm Kararsızlıkları

Breadboard üzerinde yapılan bağlantılarda temas problemleri, gevşek kablolar ve besleme dalgalanmaları ölçüm sonuçlarının kararsız olmasına neden olmuştur.

Çözüm:

Tüm bağlantılar yeniden düzenlenmiş, besleme hatları kontrol edilmiş ve sensörlerin doğru gerilim seviyelerinde çalıştığı multimetre ile doğrulanmıştır. Donanım düzeni sadeleştirilerek sistem daha güvenilir hâle getirilmiştir.

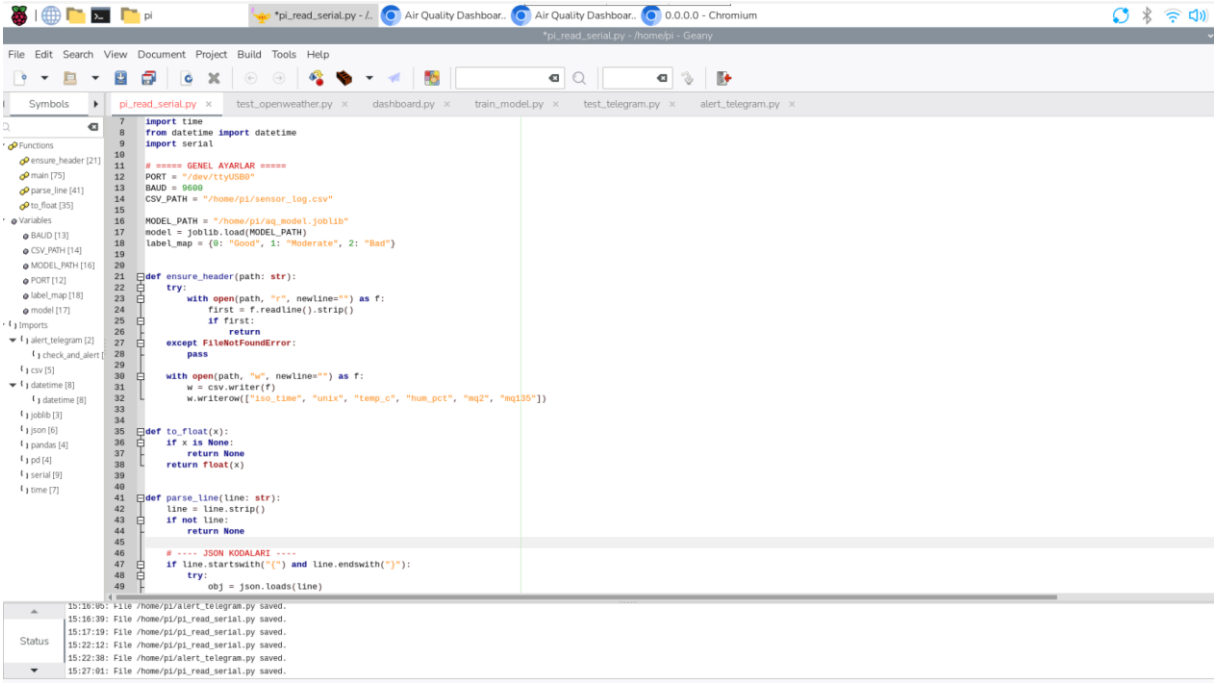
Genel Değerlendirme

Bu proje sürecinde karşılaşılan zorluklar, gömülü sistemlerde donanım ve yazılım entegrasyonunun ne kadar kritik olduğunu açıkça göstermiştir. Forumlar, teknik

dokümanlar ve yapay zekâ destekli kaynaklardan faydalanılarak problemler analiz edilmiş ve kalıcı çözümler geliştirilmiştir. Tüm bu süreçler sonucunda sistem daha kararlı, güvenilir ve geliştirilebilir bir yapıya ulaşmıştır.

KOD AÇIKLAMALARI

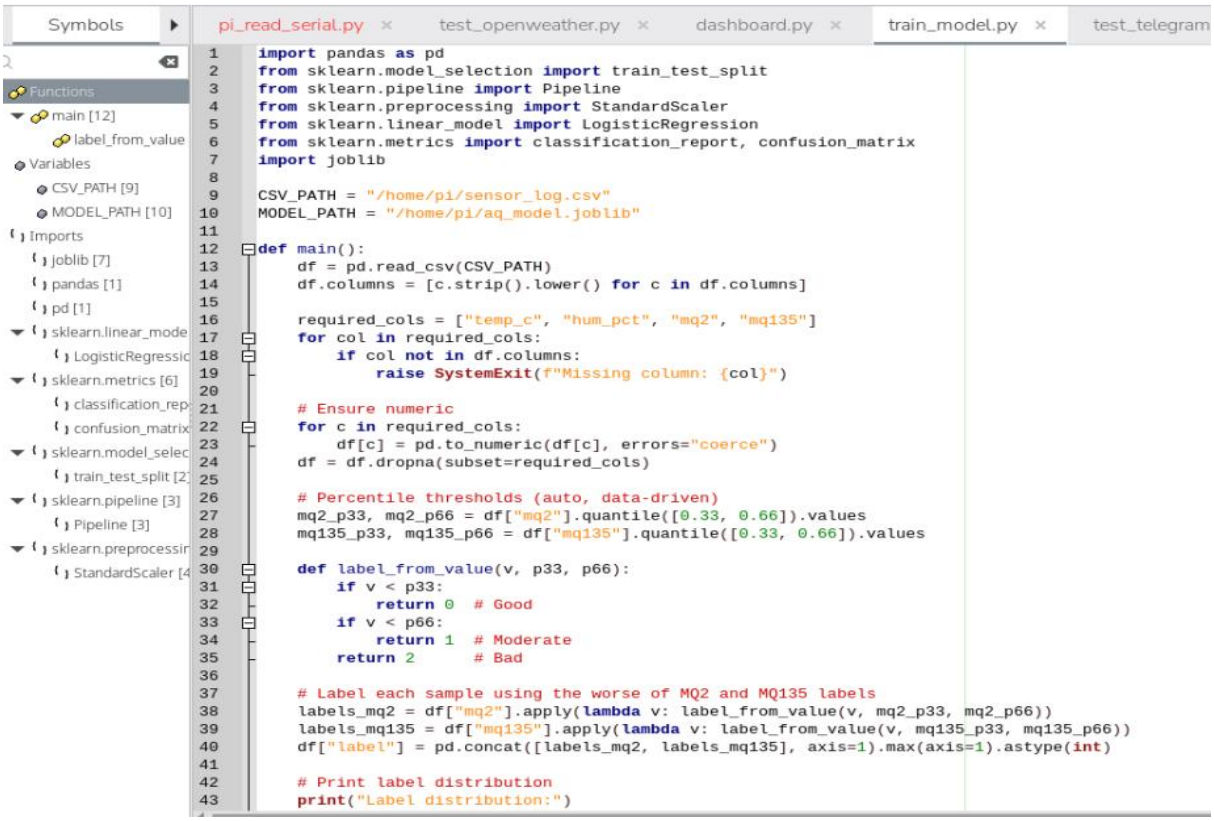
5.1 Arduino Sensör Okuma Kodu



Arduino tarafında geliştirilen yazılımın temel görevi, ortama yerleştirilen MQ-2 ve MQ-135 gaz sensörleri ile sıcaklık ve nem sensöründen elde edilen verileri okumaktır. Analog çıkış veren gaz sensörlerinden alınan değerler Arduino'nun analog giriş pinleri kullanılarak ölçülmektedir. Okunan sensör verileri belirli aralıklarla güncellenmekte ve seri haberleşme üzerinden Raspberry Pi'ye gönderilmektedir.

Kod içerisinde sensörlerden alınan ham veriler belirli bir formatta birleştirilerek tek bir veri paketi hâline getirilmektedir. Bu veri paketi, Raspberry Pi tarafında kolayca ayrıştırılabilecek şekilde tasarlanmıştır. Böylece veri kaybı ve yanlış okuma ihtimali azaltılmıştır.

5.2 Veri İşleme ve Sınıflandırma Algoritması



Toplanan sensör verileri, hava kalitesi değerlendirmesi yapılabilmesi için işlenmektedir. Belirlenen eşik değerler ve/veya eğitilmiş makine öğrenmesi modeli kullanılarak ortam hava kalitesi sınıflandırılmaktadır. Model tarafından üretilen sonuçlar kullanıcı tarafından anlaşılabilir etiketler hâline dönüştürülmektedir.

Bu aşamada sistem, hava kalitesini “iyi”, “orta” ve “kötü” gibi kategoriler altında değerlendirmekte ve sonuçları kayıt altına almaktadır.

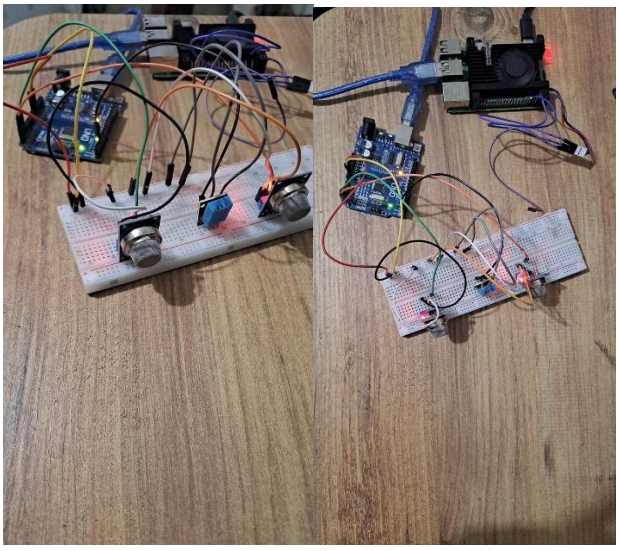
5.4 Telegram Uyarı Sistemi Kodu

```python
import os
import requests

TOKEN = os.getenv("TG_BOT_TOKEN")
CHAT_ID = os.getenv("TG_CHAT_ID")

if not TOKEN or not CHAT_ID:
    raise SystemExit("Missing TG_BOT_TOKEN or TG_CHAT_ID")

url = f"https://api.telegram.org/bot{TOKEN}/sendMessage"
payload = {"chat_id": CHAT_ID, "text": "Test: Telegram notification works ?"}
r = requests.post(url, data=payload, timeout=10)
print("Status:", r.status_code)
print(r.text)
r.raise_for_status()

```

Geliştirilen yazılım, hava kalitesi kritik seviyelere ulaştığında kullanıcıyı bilgilendirmek amacıyla Telegram botu üzerinden otomatik uyarı mesajları göndermektedir. Kod içerisinde tanımlanan eşik değerler aşıldığında uyarı fonksiyonu tetiklenmektedir.

Tekrarlayan ve gereksiz mesajların önüne geçmek için zaman kontrolü ve durum takibi yapılmaktadır. Böylece kullanıcı yalnızca gerekli durumlarda bilgilendirilmektedir.

ÖN YÜZ – ARKA YÜZ GÖSTERİMLERİ VE KULLANIM BİLGİSİ



Geliştirilen sistem, donanım ve yazılım bileşenlerinin birlikte çalıştığı bütünleşik bir yapıya sahiptir. Sistem genel olarak sensör katmanı, veri işleme katmanı ve kullanıcı bilgilendirme katmanından oluşmaktadır.

Ön yüz (kullanıcı tarafı) olarak değerlendirilebilecek kısımda, sistemin çalıştırılması ve kullanıcıya sağladığı çıktılar yer almaktadır. Sistem çalıştırıldığında sensörlerden alınan hava kalitesi, sıcaklık ve nem verileri otomatik olarak okunmakta ve Raspberry Pi

üzerinde işlenmektedir. Elde edilen veriler CSV dosyası içerisinde tarih ve saat bilgisiyle birlikte kayıt altına alınmaktadır. Ayrıca hava kalitesi kritik seviyelere ulaştığında kullanıcıya Telegram uygulaması üzerinden anlık uyarı mesajı gönderilmektedir. Kullanıcı, bu uyarılar sayesinde ortam koşullarını uzaktan takip edebilmektedir.

Arka yüz (sistem tarafı) ise donanım bağlantıları ve arka planda çalışan yazılımlardan oluşmaktadır. Arduino mikrodenetleyicisi sensörlerden gelen analog ve dijital verileri okumakta, bu verileri belirlenen bir formatta seri haberleşme aracılığıyla Raspberry Pi'ye aktarmaktadır. Raspberry Pi üzerinde çalışan Python yazılımları, seri porttan gelen verileri sürekli olarak dinlemekte, gerekli kontrolleri yapmakta ve veri işleme sürecini yürütmektedir.

Sistemin Kullanımı

- Sistem enerji verildiğinde otomatik olarak başlatılır.
- Arduino sensörlerden verileri okumaya başlar.
- Okunan veriler Raspberry Pi'ye seri haberleşme ile iletilir.
- Raspberry Pi verileri işler ve CSV dosyasına kaydeder.
- Hava kalitesi kritik seviyeye ulaştığında Telegram üzerinden uyarı gönderilir.

Bu yapı sayesinde sistem, kullanıcı müdahalesine gerek kalmadan sürekli ve gerçek zamanlı olarak çalışabilmektedir.

SONUÇ VE DEĞERLENDİRME

Bu bitirme projesi kapsamında, Raspberry Pi ve Arduino tabanlı akıllı bir hava kalitesi izleme ve erken uyarı sistemi başarıyla tasarlanmış ve uygulanmıştır. Proje süresince sensör teknolojileri, gömülü sistemler, seri haberleşme, veri kayıt yöntemleri ve yazılım entegrasyonu konularında kapsamlı çalışmalar gerçekleştirilmiştir.

Geliştirilen sistem ile ortam hava kalitesi, sıcaklık ve nem değerleri gerçek zamanlı olarak izlenebilmekte, elde edilen veriler düzenli şekilde kayıt altına alınmakta ve kritik

durumlarda kullanıcı anlık olarak bilgilendirilmektedir. Böylece sistem, yalnızca ölçüm yapan bir yapıdan ziyade erken uyarı özelliğine sahip işlevsel bir çözüm sunmaktadır.

Proje sürecinde karşılaşılan donanım arızaları, yazılım hataları ve sistem erişim problemleri, mühendislik bakış açısıyla analiz edilmiş ve kalıcı çözümler üretilmiştir. Bu süreç, problem çözme yeteneğinin ve teknik bilgi birikiminin gelişmesine önemli katkı sağlamıştır.

Gelecekte yapılabilecek geliştirmeler arasında daha hassas sensörlerin kullanılması, verilerin bulut tabanlı bir platformda saklanması, web tabanlı bir arayüz ile görselleştirilmesi ve makine öğrenmesi modellerinin doğruluğunun artırılması yer almaktadır. Bu yönleriyle proje, geliştirilmeye açık ve farklı uygulama alanlarına uyarlanabilir bir yapıdadır.

Sonuç olarak bu çalışma, düşük maliyetli donanımlar kullanılarak etkili bir çevresel izleme sistemi geliştirilebileceğini göstermiş ve bitirme projesi hedeflerini başarıyla yerine getirmiştir.