



# Datalogger Tasarımı

Yazılım Mühendisliği Ana Bilim Dalı

Dönem Projesi

Görkem Gül

ORCID 0009-0007-7860-7127

Proje Danışmanı: Prof. Dr. Femin Yalçın Küçükbayrak

Ocak 2024

# Datalogger Tasarımı

## Öz

Günümüzde endüstriyel alanlarda, evlerde, ofislerde, yemekhanelerde, soğuk hava depolarında kısacası sıcaklığın önemli olduğu sektör fark etmeksizin birçok alanda sıcaklık ölçümleri ve takipleri yapılmaktadır. Bu sensörlerden veri almak, bu verilerin kaydını tutmak ve yükselen teknoloji ile bu verileri sadece fiziksel olarak değil de dijital platformda gözlemleyebilmek, müdahale edebilmek ve bu değerlere göre proseslerde ve gerekli alanlarda iyileştirmek yapabilmek için çeşitli fikirler ve Ar-Ge çalışmaları hızla devam etmektedir.

Arduino tabanlı Datalogger, sıcaklık değerini ölçüp web tabanlı statik IP ile aynı VLAN da olan bilgisayar tarayıcıları tüm saat, zaman, IP konfigürasyonu, Sıcaklık Set değerinin ayarlaması ve bu sıcaklık değerlerinin txt formatında bir hafıza kartında depolaması gibi birçok fonksiyonu ile son kullanıcıya bilgi vermeyi hedeflemiştir.

**Anahtar Sözcükler:** Mühendislik, Teknoloji, Ar-Ge, Arduino, Web Tabanlı, POE

# Datalogger Design

## Abstract

Today, temperature measurements and follow-ups are carried out in industrial areas, homes, offices, dining halls, cold storage facilities, in short, in many areas, regardless of the sector where temperature is important. Various ideas and R&D studies are continuing rapidly in order to receive data from these sensors, keep a record of these data, and with the rising technology, be able to observe these data not only physically, but also on a digital platform, intervene decently, and improve processes and necessary areas according to these values.

Arduino-based Datalogger, computer scanners that measure the temperature value and have the same Vlan as web-based static IP, aimed to provide information to the end user with many functions such as setting the entire time, time, IP configuration, Temperature setpoint and storing these temperature values on a memory card in txt format.

**Keywords:** Engineering, Technology, R&D, Arduino, Web Based, POE

# Teşekkür

İlk olarak, projenin yönetimini üstlenen danışman hocam Sayın Prof. Dr. Femin Yalçın Küçükbayrak'a teşekkür etmek istiyorum. Sizin bilgeliğiniz, uzmanlığınız ve önerileriniz projenin kalitesini arttırmama katkı sağladı. Sunduğunuz değerli geri bildirimler ve rehberlik bu projenin anlaşılmasına ve başarıya ulaşmasına olanak tanıdı.

Bu projede elde ettiğim deneyimler, sizlerle çalışmanın getirdiği değeri göstermektedir. İlham verici destekleriniz için teşekkürlerimi sunarım.

# İçindekiler

Öz .....	i
Abstract .....	ii
Teşekkür .....	iii
Şekiller Listesi.....	vi
Kısaltmalar Listesi .....	vii
Semboller Listesi.....	viii
<b>1 Giriş .....</b>	<b>1</b>
<b>2 Projenin Amacı ve Önemi.....</b>	<b>2</b>
2.1 Araştırma Soruları ve Hipotezler .....	2
2.2 Araştırmanın Genel Alanı .....	3
2.3 Metodoloji Hakkında .....	5
2.4 Kapsam ve Sınırlamalar .....	7
<b>3 Literatür Taraması.....</b>	<b>8</b>
3.1 Genel Bakış .....	8
3.2 Temel Kavramlar ve Tanımlar .....	9
3.3 Önceki Çalışmalar.....	29
3.4 Gelecekteki Araştırmalar için Öneriler .....	30
<b>4 Metodoloji .....</b>	<b>32</b>
4.1 Araştırma Deseni .....	32
4.2 Araştırma Popülasyonu ve Örneklem .....	33
4.3 Veri Toplama Yöntemleri .....	34
4.4 Veri Analizi Yöntemleri .....	35
<b>5 Beklenen Katkılar.....</b>	<b>37</b>
3.1 Uygulama Görselleri .....	39
<b>6 Sonuç.....</b>	<b>43</b>

<b>Kaynaklar .....</b>	<b>44</b>
------------------------	-----------

# Şekiller Listesi






Şekil 3.1	Arduino Mega 2560 .....	9
Şekil 3.2	Arduino Ethernet Shield.....	10
Şekil 3.3	DHT11 Sıcaklık ve Nem Sensörü .....	12
Şekil 3.4	Liquid Crystal 16x2 .....	13
Şekil 3.5	Timer DS1307 .....	15
Şekil 3.6	XL6009 Regülatör.....	17
Şekil 5.1	Web Arayüz.....	39
Şekil 5.2	Web Set Arayüz .....	39
Şekil 5.3	Cihazın Tamamlanmış Hali.....	40
Şekil 5.4	Cihazın Tamamlanmış Hali.....	40
Şekil 5.5	SD Kart LOG .....	41

# Kısaltmalar Listesi

Bakiođlu Holding	Bakiođlu Holding
FBE	Fen Bilimleri Enstitüsü
İKÇÜ	İzmir Kâtip Çelebi Üniversitesi
Atmel	Arduino Atmel
ORCID	Open Researcher and Contributor ID



# Semboller Listesi

	HTTPS
	Histerezis
	Sıcaklık Deęeri
	Tarih
	Müşteri Web Arayüzü

# Bölüm 1

## Giriş

Günümüzde, çeşitli endüstriyel ve ev kullanımı uygulamalarında sıcaklık ve nem ölçümleri büyük önem taşımaktadır. Bu değerler, tarım, enerji yönetimi, iklim kontrolü, sağlık ve birçok diğer alanlarda kritik rol oynamaktadır. Bu bağlamda, sıcaklık ve nem sensörleri, çeşitli veri toplama sistemlerinde yaygın olarak kullanılmaktadır.

Bu tez çalışması, sıcaklık ve nem ölçümlerinin hassas bir şekilde gerçekleştirebilmesi amacıyla geliştirilen “Sıcaklık Nem Datalogger” adlı Arduino tabanlı bir projeyi incelemektedir. Proje, çevresel koşulları izlemek ve ölçmek amacıyla geliştirilmiş bir sensör sistemini içermektedir. Bu sistem mikrodenetleyicilerle entegre edilen sensörler kullanılarak oluşturulur. Arduino platformunu, düşük maliyeti, açık kaynaklı yapısı ve geniş kullanıcı kitlesi ile popüler bir seçenek haline getirmiştir. Bu çalışmada kullanılan Arduino tabanlı sistem, sıcaklık ve nem Ölçümlerini hassas bir şekilde gerçekleştirebilen sensörleri içermekte ve bu verileri Web üzerinden statik IP ile arayüz olarak kullanıcıya set değer aralığı sağlama, endüstriyel alanda aynı domainde kalmasını sağlayarak IP konfigürasyon ayarlaması yapılabilmektedir. Bununla birlikte üzerinde SD kart okuyucu sayesinde okuduğu değerlerin geçmişini tutmaktadır.

Bu tez, sıcaklık ve nem ölçümlerinin gerçekleştirilmesi, veri toplama süreçleri, Arduino tabanlı sistemlerin avantajlarını ve proje uygulama alanları gibi konuları detaylı ele almaktadır. Bu çalışma sıcaklık ve nem ölçümlerine yönelik bir pratik bir uygulama sunarak, “Sıcaklık Nem Datalogger” adlı projenin çeşitli endüstriyel ve ev kullanımı senaryolarında kullanılacak güvenilir bir sensör sistemini ortaya koymaktadır. Tezin devamındaki bölümlerde, kullanılan sensör teknolojileri, sistem tasarımı, uygulama alanları ve elde edilen sonuçlar detaylı bir şekilde incelenecektir.

# Bölüm 2

## Projenin Amacı

Bu projenin amacı, çevresel koşulları ve hassas bir şekilde izleyen ve ölçen bir sıcaklık ve nem datalogger sistemi geliştirmektir. Bu sistem, sıcaklık ve nem ölçümlerini mikrodenetleyiciler üzerinden toplamakta ve elde edilen verileri kullanıcı dostu bir arayüzle görselleştirmekte ve kontrol edebilmektedir.

- **Hassas Ölçümler:** Projede kullanılan sensörler, sıcaklık ve nem ölçümlerini yüksek hassasiyetle gerçekleştirebilmektedir. Bu, çeşitli endüstriyel ve ev kullanımını senaryolarında güvenilir veri elde etmeyi sağlamaktadır.
- **Veri Saklama ve Analiz:** Datalogger sistemi, ölçülen sıcaklık ve nem verilerini belirli aralıklarla kaydederek uzun süreli izleme ve analiz imkânı sunmaktadır. Bu özellik, çeşitli uygulama alanlarında geniş veri setlerinin elde edilmesine olanak tanımaktadır.
- **Kullanıcı Dostu Arayüz:** Proje, kullanıcıların ölçülen verilere kolayca erişebilmesi için kullanıcı dostu bir arayüz sunmaktadır. Bu arayüz, verilerin grafiklerle ve tablolarla görselleştirilmesini sağlayarak kullanıcılara anlamlı bir gözleme imkânı sunmaktadır. [6]

### 2.1 Araştırma Soruları ve Hipotez

Araştırma soruları proje tasarımında neyi ne amaç uğruna yaptığımızın en somut fikirlerini veren kısımdır. Bu aşamada ilham alınan sorular ve bunlara yönelik geliştirilen kısımlardan tezin ileri aşamalarında bahsedilmiştir.

- Sıcaklık ve nem ölçümlerinin hassasiyetini artırmak için kullanılan sensör teknolojileri nelerdir?
- Arduino tabanlı sıcaklık ve nem datalogger sistemlerinin endüstriyel uygulamalardaki potansiyeli nedir?

- Datalogger verilerinin uzun vadeli saklanması ve analizi için en etkili yöntemler nelerdir?
- Projenin kullanıcı dostu arayüzü, sıcaklık ve nem verilerini hangi şekillerde görselleştirmektedir?
- Sıcaklık ve nem ölçümleri, tarım sektöründe nasıl kullanılabilir ve bu alanda nasıl bir katkı sağlayabilir?
- Projede kullanılan sensörlerin kalibrasyonu ve doğrulama süreçleri nelerdir?
- Datalogger sistemi, enerji yönetimi alanında nasıl kullanılabilir ve enerji verimliliğine nasıl katkı sağlayabilir?
- Projede elde edilen verilerin uzaktan izlenmesi ve kontrolü için hangi iletişim protokolleri tercih edilebilir?

**Hipotez:** Bu projedeki hipotez, çeşitli endüstriyel ve ev kullanımı senaryolarında kullanılabilen güvenilir bir sensör sistemini temsil eder. Bu sistem, sıcaklık ve nem ölçümlerini yüksek hassasiyetle gerçekleştirerek, çevresel koşulların izlenmesi ve kontrolüne katkı sağlar. Ayrıca, uzaktan veri izleme ve enerji yönetimi gibi uygulama alanlarında önemli bir potansiyele sahiptir. Bu hipotez, projenin sıcaklık ve nem ölçümlerine yönelik pratik bir çözüm sunduğunu ve çeşitli sektörlerde kullanılabilirliğini desteklemektedir."

## 2.2 Araştırmanın Genel Alanı

Proje kapsamı ve sınırlamaları, projenin neyi amaçladığını belirlerken aynı zamanda projenin hangi sınırlamalarla karşılaşabileceğini ve hangi alanlarda geçerli olmadığını belirtir.

**Sensör Teknolojileri:** Projede kullanılan sensör teknolojileri, sıcaklık ve nem ölçümlerinin hassasiyetini artırmak amacıyla incelenebilir. Sensör seçiminde kullanılan teknikler ve avantajları proje kapsamında yer alabilir.

**Arduino Tabanlı Sistem:** Projede geliştirilen sıcaklık ve nem datalogger sistemi, Arduino tabanlı olduğu için bu mikrodenetleyici platformunun sağladığı avantajlar ve sınırlamalar ele alınabilir. Arduino'nun genel kullanımı, programlama dili ve desteklediği sensör tipleri proje kapsamında değerlendirilebilir.

**Uygulama Alanları:** Projede odaklanılan sıcaklık ve nem ölçümleri, tarım, enerji yönetimi, ev otomasyonu ve benzeri uygulama alanlarında kullanılabilirlik açısından incelenebilir. Bu, projenin hangi sektörlerde ve senaryolarda etkili olduğunu belirler.

**Veri Görselleştirme:** Projede kullanıcı dostu bir arayüzle gerçekleştirilen veri görselleştirme, projenin kullanıcılara nasıl anlamlı bilgiler sağladığını kapsar. Grafikselleştirme, tablolar ve kullanıcı etkileşimi projenin kapsamını oluşturabilir.

### **Sınırlamalar:**

**Sensör Hassasiyeti:** Projede kullanılan sensörlerin hassasiyeti ve doğruluğu belirli bir limit içinde olabilir. Bu, ölçümlerin tam olarak ne kadar güvenilir olduğunu belirler.

**Enerji Tüketimi:** Arduino tabanlı sistem, enerji tüketimi açısından belirli bir sınırlamaya sahip olabilir. Bu durum, projenin uzun süreli kullanımda enerji verimliliğini sınırlayabilir.

**Uzaktan İzleme:** Projede ele alınan uzaktan izleme yetenekleri, kullanılan iletişim protokollerine ve donanım özelliklerine bağlı olarak sınırlı olabilir.

**Maliyet:** Projenin maliyeti, kullanılan sensörler, mikrodenetleyici ve diğer bileşenler tarafından belirlenebilir. Bu durum, projenin genel erişilebilirliğini etkileyebilir.

**Çalışma Koşulları:** Projede kullanılan sensörlerin ve donanımın çalışma koşulları belirli bir sınırlama içinde olabilir. Örneğin, belirli bir sıcaklık aralığında veya belirli bir nem seviyesinde daha etkili olabilirler.

Bu kapsam ve sınırlamaların açıkça belirlenmesi, projenin amacını ve hedeflerini daha net bir şekilde tanımlamanıza yardımcı olacaktır.

## 2.3 Metodoloji Hakkında

Bu metodoloji bölümü, sıcaklık ve nem verilerini kaydeden bir data logger tasarımını adım adım açıklamaktadır. Proje, donanım ve yazılım bileşenlerini içerir ve aşağıda belirtilen adımları takip eder.

### 1. Literatür Taraması:

Literatür taraması aşamasında, mevcut sıcaklık ve nem ölçüm sistemleri, mikrodenetleyici tabanlı datalogger projeleri ve web arayüzü tasarımları incelenmiştir. Bu inceleme, benzer projelerde kullanılan teknolojik çözümleri ve tasarım yaklaşımlarını anlamamıza yardımcı olmuştur.

### 2. Sistem Mimarisi:

Sistem mimarisi belirlenmiş ve blok diyagramlar kullanılarak detaylandırılmıştır. Bu adımda, sıcaklık ve nem sensörleri, mikrodenetleyici, SD kart kayıt sistemi ve web tabanlı arayüzün entegrasyonu planlanmıştır.

### 3. Donanım Tasarımı:

Donanım tasarımı adımında, sensörlerin seçimi, mikrodenetleyici entegrasyonu, güç yönetimi ve bağlantı şemaları detaylandırılmıştır. Donanımın güvenilirliği, hassasiyeti ve enerji verimliliği ön planda tutulmuştur.

### 4. Yazılım Tasarımı:

Yazılım tasarımı aşamasında, mikrodenetleyici üzerinde çalışacak yazılım geliştirilmiştir. Bu yazılım, sensör verilerini okuma, SD kart üzerine kayıt, web arayüzü ile iletişim kurma ve kullanıcı tarafından ayarlanabilir sıcaklık set değerlerini kontrol etme yeteneklerini içerir.

### 5. Veri Kaydı ve Depolama:

SD kart üzerindeki veri kaydı işlemleri belirlenmiş ve veri formatı standardı oluşturulmuştur. Verilerin düzenli olarak kaydedilmesi ve depolanması için bir strateji oluşturulmuştur.

#### 6. Web Arayüzü Tasarımı:

Web arayüzü tasarımında, kullanıcı dostu bir arayüz geliştirilmiş ve bu arayüz üzerinden sıcaklık set değerlerini ayarlamak, mevcut verileri gözlemlemek ve analiz yapmak mümkün kılınmıştır.

#### 7. Test ve Performans Değerlendirmesi:

Donanım ve yazılım tasarımları test edilmiş ve performans değerlendirilmesi yapılmıştır. Sistem stabilitesi, sıcaklık hassasiyeti ve kullanıcı arayüzü kullanılabilirliği değerlendirilmiştir.

#### 8. Sonuçlar ve Tartışma:

Elde edilen sonuçlar değerlendirilmiş, proje başarıları ve sınırlamaları tartışılmıştır. Ayrıca, gelecekteki geliştirmeler ve çalışmalar için öneriler sunulmuştur.

#### 9. Sonuç:

Bu metodoloji, sıcaklık ve nem datalogger tasarımının detaylı bir şekilde gerçekleştirilmesini sağlamıştır. Her bir adım, proje hedeflerine ulaşmak ve tasarımın etkin bir şekilde uygulanmasını sağlamak amacıyla titizlikle planlanmıştır.

## 2.4 Kapsam ve Sınırlamalar

### Kapsam:

Bu sıcaklık ve nem data logger tasarım projesi, belirli bir sıcaklık ve nem aralığında çalışacaktır. Projede kullanılacak mikrodenetleyici platformu belirlenecek ve donanım özellikleri bu platforma göre tasarlanacaktır. Veri kaydı sıklığı, enerji tüketimi ve SD kartın depolama kapasitesi gibi faktörlere bağlı olarak ayarlanacak ve belirli bir güç kaynağı kullanılacaktır. Web arayüzü, temel işlevselliklere odaklanarak kullanıcıların sıcaklık set değerlerini ayarlamasını, mevcut verileri gözlemlemesini ve analiz yapmasını sağlayacaktır.

### Sınırlamalar:

Projenin hassasiyet ve doğruluk sınırlamaları, kullanılacak sensörlerin özellikleri ile belirlenecektir. SD kartın depolama kapasitesi, veri kaydı sıklığı ve depolanacak veri miktarına bağlı olarak sınırlamalar getirecektir. İletişim arayüzü, belirli bir teknoloji (örneğin, Wi-Fi, Ethernet) üzerinden web arayüzüne bağlanacaktır ve bu, iletişim hızı ve güvenilirlik açısından sınırlamalar getirebilir. Tasarımın kullanılacağı uygulama alanı belirli bir ortamı kapsayabilir ve özellikle endüstriyel kullanımlar için dayanıklılık ve dış ortam koşulları gibi sınırlamalar ortaya çıkabilir. Güvenlik ve veri gizliliği, web arayüzü üzerinden iletişimde önemlidir ve belirli güvenlik önlemleri sınırlamaları şekillendirecektir.



# Bölüm 3

## Literatür Taraması

### 3.1 Genel Bakış

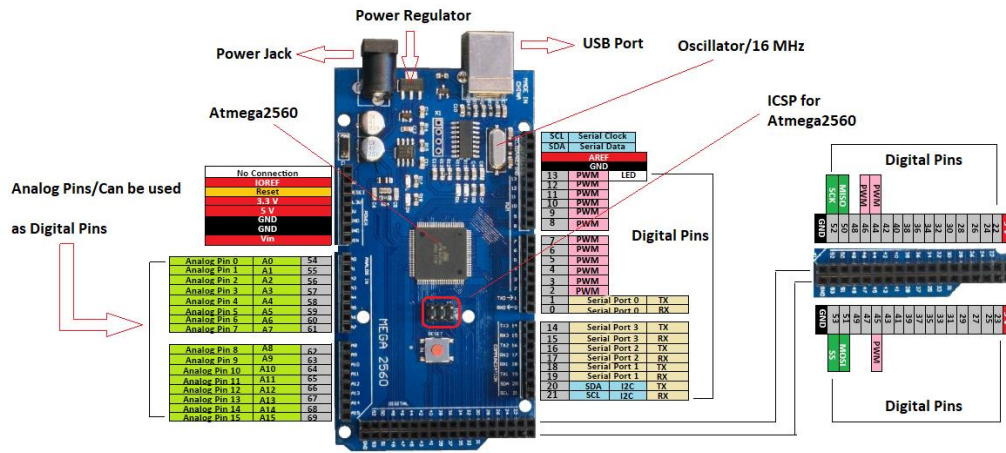
Bu sıcaklık ve nem datalogger tasarımı projesi, literatür taraması aşamasında mevcut bilimsel çalışmaları ve endüstri uygulamalarını detaylı bir şekilde inceleyerek temel alacaktır. Sıcaklık ve nem ölçüm sistemleri, mikrodenetleyici tabanlı datalogger projeleri, web arayüzü tasarımları ve ilgili teknolojik çözümler, literatür taraması kapsamında ayrıntılı bir analizden geçirilecektir. Benzer projelerde kullanılan sensör teknolojileri, mikrodenetleyici platformları ve iletişim arayüzleri üzerine yapılan çalışmalar, bu projenin tasarım aşamalarında ve uygulama süreçlerinde rehberlik edecek önemli referanslar sunacaktır. Literatür taraması, proje hedeflerini belirleme, uygun teknolojik seçimleri yapma ve tasarımın en etkin şekilde uygulanmasını sağlama konusunda temel bir bilgi tabanı oluşturacaktır.

Literatür taraması aşamasında, benzer projelerde elde edilen deneyimler ve bu projelerin karşılaştığı zorluklar da göz önünde bulundurulacaktır. Sensör teknolojileri ile ilgili güncel gelişmeler, mikrodenetleyici tabanlı sistemlerin performans özellikleri, enerji tüketimi optimizasyonu, veri güvenliği ve web arayüzü tasarımındaki en iyi uygulamalar gibi konular üzerinde durularak literatürdeki açıklıklar belirlenecektir. Ayrıca, endüstri standartları ve regülasyonlar, benzer projelerin kullanıldığı sektörlerdeki ihtiyaçlar ve bu tür sistemlerin günlük uygulamalardaki etkinliği konularında literatür taraması genişletilecektir. Bu sayede, proje, literatürdeki eksiklikleri doldurarak önceki çalışmalardan elde edilen bilgileri en iyi şekilde kullanma amacını taşıyacaktır. [5]

## 3.2 Temel Kavramlar ve Tanımlar

Bu projede çeşitli modüller kullandık ve bunların projedeki görevlerinden bahsetmek gerekirse öyle özetleyebiliriz.

### A. Arduino Mega 2560



**Arduino Mega 2560 Pinout**

Şekil 3.1: Arduino Mega 2560

**Mikrodenetleyici:** Arduino Mega 2560, Atmel ATmega2560 mikrodenetleyicisini temel alır. Bu mikrodenetleyici, 8-bit AVR mimarisine sahip ve 16 MHz saat hızında çalışır.

**Bellek Kapasitesi:** Mega 2560, 256 KB Flash belleğe ve 8 KB SRAM'e sahiptir. Bu, karmaşık projeleri kolayca barındırabilme yeteneği sağlar.

**Giriş/Çıkış Pinleri:** Toplamda 54 adet dijital giriş/çıkış pini vardır. Bu pinlerin 14'ü PWM çıkışı verebilir ve 16'sı analog giriş olarak kullanılabilir.

**Analog Girişler:** Arduino Mega 2560, 16 adet analog giriş pini sunar. Bu pinler, analog sensörler ve diğer analog sinyalleri okumak için kullanılır.

**Dijital Giriş/Çıkış Pinleri:** 54 adet dijital I/O pinine sahiptir. Bu pinler, dijital sensörlerle iletişim kurmak veya dijital cihazları kontrol etmek için kullanılabilir.

**İletişim Portları:** Mega 2560, USB bağlantısı, seri iletişim için UART portları, SPI ve I2C gibi popüler iletişim protokollerini destekler.

**Güç Besleme:** Kart, USB bağlantısı üzerinden veya harici bir güç kaynağı kullanılarak beslenebilir. Harici güç kaynağı 7V ile 12V arasında olmalıdır.

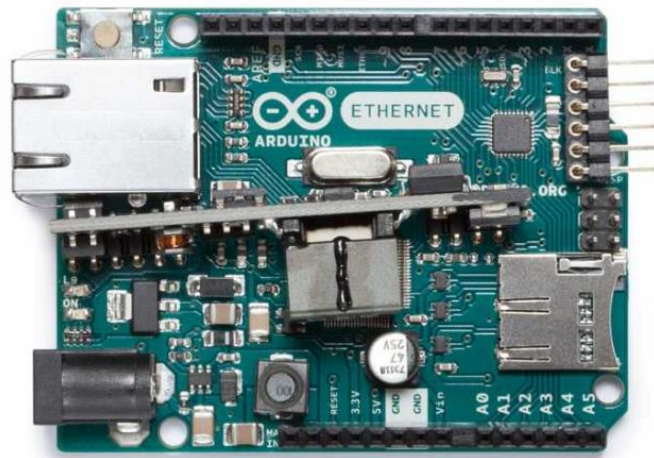
**Programlama:** Arduino Mega 2560, Arduino IDE (Integrated Development Environment) üzerinden programlanabilir. Programlama için C++ tabanlı bir dil kullanılır.

**Geniştirilebilirlik:** Kart, şeffaf bir tasarıma sahiptir ve üzerindeki header pinleri sayesinde çeşitli genişleme shield'ları (kartları) bağlamak mümkündür. Bu, projelerinizi çeşitlendirmenize olanak tanır.

**Çeşitli Uygulamalar:** Arduino Mega 2560, büyük ve karmaşık projeleri yönetmek için uygun bir seçenektir. Robotik projeler, otomasyon sistemleri, veri toplama uygulamaları gibi geniş bir uygulama yelpazesine hitap eder.

Arduino Mega 2560, geniş pin sayısı, büyük bellek kapasitesi ve çeşitli iletişim portları gibi özellikleri ile karmaşık projelerde kullanılmak üzere tasarlanmış bir mikrodenetleyici kartıdır. [3]

## B. Arduino Ethernet Shield POE



Şekil 3.2: Ethernet Shield

**Ethernet Bağlantısı:** Modül, Arduino kartınızı bir ağa bağlamak için bir Ethernet portu sağlar. Bu sayede Arduino'nuz internet üzerinden veri alabilir ve gönderebilir.

**Power over Ethernet (PoE):** Modül, Ethernet kablosu üzerinden güç alabilme özelliğine sahiptir. Bu, ayrı bir güç kaynağı kullanmadan, sadece Ethernet kablosu aracılığıyla Arduino'nuzu besleyebilmenizi sağlar.

**IEEE 802.3af Standardı:** Arduino Ethernet Shield PoE modülü, IEEE 802.3af standardına uyan PoE özelliğini destekler. Bu standart, Ethernet kablosu üzerinden güç iletimini sağlar.

**RJ45 Bağlantı Noktası:** Ethernet Shield, üzerinde bir RJ45 Ethernet bağlantı noktasına sahiptir. Bu sayede cihaz, bir ağa doğrudan bağlanabilir.

**SPI Arabirimi:** Ethernet Shield, SPI (Serial Peripheral Interface) üzerinden Arduino ile iletişim kurar. SPI, yüksek hızlı seri iletişim sağlayan bir protokoldür.

**Bellek Kapasitesi:** Ethernet Shield üzerinde, özellikle W5100 çipine sahip olan modellerde, bellek kapasitesi bulunur. Bu bellek, ağ üzerinden gelen ve giden verileri geçici olarak depolamak için kullanılır.

**IP Tahsisi ve DHCP Desteği:** Ethernet Shield, kullanıcının belirleyebileceği bir IP adresi atanabilir veya DHCP (Dynamic Host Configuration Protocol) ile otomatik olarak bir IP adresi alabilir.

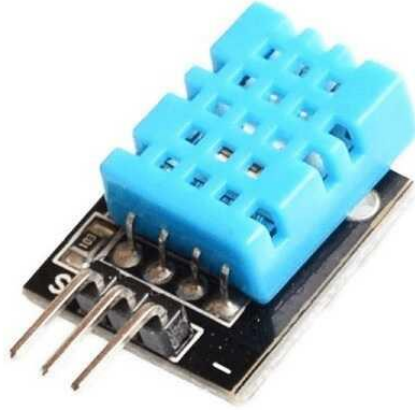
**Web Sunucu ve İstemci Yetenekleri:** Ethernet Shield, Arduino'yu bir web sunucusu veya web istemcisi olarak kullanma yeteneği sağlar. Bu sayede Arduino, ağ üzerinden web sayfaları gönderebilir veya alabilir.

**Arduino IDE Desteği:** Ethernet Shield, Arduino IDE (Integrated Development Environment) üzerinden programlanabilir. Arduino'nun resmi kütüphaneleri, Ethernet Shield ile ilgili hazır fonksiyonları içerir.

Ethernet Shield, özellikle IoT (Internet of Things) projeleri ve uzaktan kontrol uygulamaları gibi senaryolarda Arduino tabanlı cihazların ağ üzerinden

haberleşme ihtiyaçlarını karşılamak için kullanılır. Bu shield, Arduino'nun ağ yeteneklerini genişleterek, cihazların internet üzerinden kontrol edilebilmesini ve veri alışverişinde bulunabilmesini sağlar. [7]

### C. DHT11 Sıcaklık ve Nem Sensörü



Şekil 3.3: Sıcaklık ve Nem Sensörü

**Çalışma Prensipleri:** DHT11, kapasitif ölçüm prensibiyle çalışan bir sensördür. Nem ölçümü, sensörün üzerindeki özel bir kapasitif polimeri kullanarak gerçekleştirilir. Sıcaklık ölçümü için ise bir NTC (Negatif Sıcaklık Katsayılı) direnci bulunur.

**Ölçüm Aralıkları:** DHT11, 0 ila 50°C sıcaklık aralığında  $\pm 2^{\circ}\text{C}$  hassasiyetle ölçüm yapabilir. Nem ölçüm aralığı ise %20 ila %80 arasındadır.

**Dijital İletişim:** DHT11, dijital bir sensördür ve verilerini dijital formatta iletebilir. Genellikle bir mikrodenetleyiciye bağlandığında, sensörden alınan sıcaklık ve nem değerleri dijital sinyaller olarak okunabilir.

**Bağlantı ve Besleme:** DHT11 sensörü, genellikle üç pin içerir: VCC (besleme), GND (toprak), ve DATA (veri). 5V besleme gerilimine ihtiyaç duyar.

**Ölçüm Süresi:** DHT11, bir ölçüm yapma süresi olan 1-2 saniye arasında değişen bir ölçüm periyoduna sahiptir.

**Düşük Maliyet ve Basit Kullanım:** DHT11, düşük maliyetli bir sensördür ve genellikle başlangıç seviyesindeki projelerde tercih edilir. Arduino ve diğer mikrodenetleyicilere kolayca entegre edilebilir.

**Kalibrasyon:** DHT11 genellikle fabrikada kalibre edilir ve kullanıcı tarafından kalibrasyon gerektirmez.

**Uygulama Alanları:** DHT11 sıcaklık ve nem sensörleri, ev otomasyonu, hava durumu istasyonları, iklimlendirme sistemleri ve bitki yetiştirme uygulamaları gibi çeşitli alanlarda kullanılır.

DHT11 sensörü, uygun maliyeti, basit kullanımı ve düşük güç tüketimi nedeniyle birçok projede tercih edilen bir sensördür. Ancak, daha yüksek hassasiyet gerektiren uygulamalarda daha gelişmiş sensörler de düşünülebilir. [2]

#### D. Liquid Crystal 16x2 LCD



Şekil 3.4 : LCD Ekran

**Ekran Boyutu ve Karakter Sayısı:**16x2 LCD ekranı, 16 karakter genişliğinde ve 2 satır yüksekliğindedir. Toplamda 32 karakterin görüntülenebildiği bir ekrandır.

**Çalışma Prensi:**16x2 LCD ekranlar, genellikle HD44780 veya uyumlu bir kontrolcüye sahiptir. Bu kontrolcü, karakterlerin ve komutların ekran üzerinde düzenlenmesini sağlar.

**Çözünürlük:** Her bir karakter alanı 5x8 piksel çözünürlüğünde olup, her bir karakter, 5 piksel genişliğinde ve 8 piksel yüksekliğindedir.

**Aydınlatma:**16x2 LCD ekranları genellikle bir aydınlatma kaynağına sahiptir. Aydınlatma, genellikle LED'lerle sağlanır ve kullanıcının ekranı daha iyi görmesini sağlar.

**Kontrol ve Bağlantı:** LCD ekranları, genellikle bir mikrodenetleyici veya başka bir kontrol ünitesi üzerinden kullanılır. Veri ve kontrol sinyalleri genellikle paralel olarak sağlanır.

**Çalışma Gerilimi:**16x2 LCD ekranlar genellikle 5V gerilimle çalışır. Güç tüketimi düşük olduğundan, taşınabilir veya düşük güçlü projelerde sıklıkla tercih edilir.

**Karakter Seti ve Dil Desteği:** LCD ekranlar genellikle standart ASCII karakter setini destekler ve farklı dillerde yazı yazma yeteneğine sahiptir. Programlanabilir Karakterler Bazı LCD ekranlar, kullanıcıların belirli karakterleri programlamasına izin veren özel karakter (custom character) özelliklerine sahiptir.

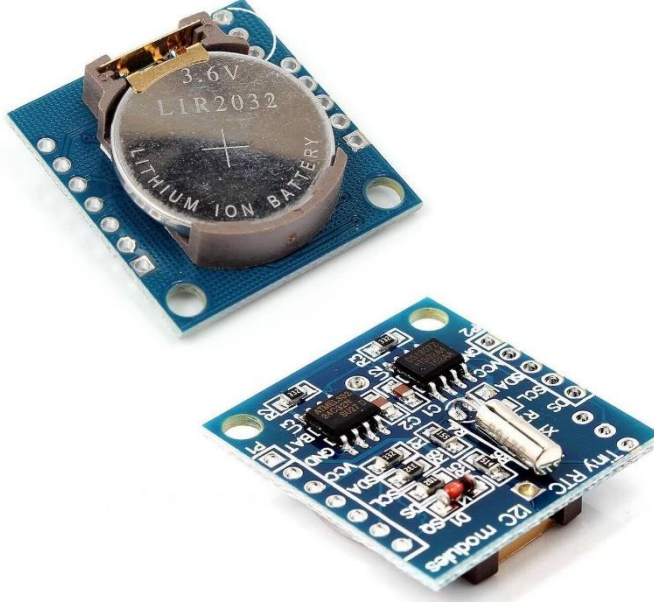
**Uygulama Alanları:**16x2 LCD ekranları, mikrodenetleyici projelerinde, otomasyon sistemlerinde, ölçüm cihazlarında, saatlerde ve diğer birçok elektronik uygulamada bilgi göstermek için kullanılır.

**Arduino ve Raspberry Pi Uyumluluğu:**16x2 LCD ekranları, Arduino ve Raspberry Pi gibi popüler geliştirme platformları ile uyumludur ve bu platformlarda kolayca kullanılabilir.

Bu LCD ekranlar, bilgi gösterme, kullanıcı arayüzü oluşturma ve projelerde geri bildirim sağlama gibi birçok elektronik projede yaygın olarak kullanılır. LCD

ekranların kullanımını genellikle Arduino veya diğer mikrodenetleyici platformları ile kodlanır ve kontrol edilir. [4]

#### E. Timer DS1307



Şekil 3.5: Zaman Modülü

**RTC ve Saat Fonksiyonu:**DS1307, Real-Time Clock (RTC) fonksiyonları sağlar. Mikrodenetleyici üzerinden sağlanan veri ve kontrol sinyalleriyle haberleşerek zaman ve tarih bilgisini tutar.

**Entegre Saat Devresi:**DS1307, içerisinde entegre bir saat devresi barındırır. Bu saat devresi, saniye, dakika, saat, gün, ay ve yıl bilgilerini tutabilir.

**Hassas Saat Tutma:**DS1307, kristal osilatör kullanarak zamanı oldukça hassas bir şekilde tutar. Genellikle saniye başına küçük bir sapma ile zamanı ölçer.

**Paralel Arayüz:**DS1307, paralel bir veri ve kontrol arayüzü kullanır. Bu arayüz, mikrodenetleyici ile bağlantı yapılabilmesi için gereken pimleri içerir.



**I2C (Inter-Integrated Circuit) İletişim Protokolü:**DS1307, I2C iletişim protokolünü destekler. Bu, mikrodenetleyici ile veri alışverişi için kullanılır ve birkaç telli bir bağlantı gerektirir.

**Batarya Destekli:**DS1307, RTC'nin zaman ve tarih bilgilerini saklamak için bir batarya ile desteklenmiştir. Bu, ana güç kaynağı kesildiğinde bile zamanın kaybolmamasını sağlar.

**Programlanabilir Zaman Ayarları:**DS1307, mikrodenetleyici üzerinden programlanabilir ve okunabilir zaman ayarlarına sahiptir. Saat, dakika, saniye, gün, ay ve yıl gibi parametreler ayarlanabilir.

**Alarmlar ve Kesmeler:**DS1307, belirli zamanlarda veya belirli bir zaman geldiğinde mikrodenetleyiciye alarm sinyali gönderebilir. Bu, belirli olaylara tepki vermek için kullanılabilir.

**Arduino ve Raspberry Pi Uyumluluğu:**DS1307, popüler mikrodenetleyici platformları olan Arduino ve Raspberry Pi ile uyumlu olup, bu platformlarda kullanım için hazır kütüphanelere sahiptir.

DS1307 RTC yongası, zamanı takip etme ihtiyacı olan birçok projede kullanılır. Özellikle saat uygulamaları, günlük kayıt sistemleri, zamanla kontrol edilen cihazlar ve benzeri projelerde DS1307 RTC yongası sıklıkla tercih edilen bir zaman çözümüdür. [4]

## F. XL6009 Regülatör (1.25V- 35V)



Şekil 3.6: Regülatör Modül

Bu devredeki görevi Ethernet Shield PoE beslemeli enerji alacağı için Çıkış voltajını Arduino'ya aktarabilmesi için Arduino'nun maksimum giriş voltajı 12 voltur.

**İşlev ve Kullanım:** XL6009, giriş voltajını daha yüksek bir çıkış voltajına dönüştüren bir adım yükseltici regülatördür. Özellikle taşınabilir cihazlarda veya pil tabanlı sistemlerde, düşük voltajlı bir pilin voltajını artırmak için kullanılır.

**Çalışma Prensipleri:** XL6009, anahtarlama modu güç kaynağı (SMPS) olarak çalışır. Anahtarlama modu regülatörler, anahtarlama frekansında enerjiyi depolayıp transfer ederek verimli bir şekilde voltaj yükseltme veya düşürme işlemini gerçekleştirir.

**Giriş ve Çıkış Voltajı:** XL6009, geniş bir giriş voltaj aralığına sahiptir ve çıkış voltajını daha yüksek bir seviyeye çıkarmak için kullanılır. Genellikle ayarlanabilir bir çıkış voltajı sunar.

**Ayarlama Fonksiyonu:** XL6009, giriş ve çıkış kondansatörlerinin değerlerini değiştirerek veya bir potansiyometre kullanarak çıkış voltajını ayarlama imkânı sunar.

**Verimlilik:** Anahtarlama modu regülatörler, genellikle lineer regülatörlere göre daha yüksek verimliliğe sahiptirler, çünkü enerjiyi depolar ve transfer ederler.

**Koruma Özellikleri:** XL6009, aşırı akım, aşırı voltaj ve termal koruma gibi güvenlik özelliklerine sahip olabilir.

**Uygulama Alanları:** XL6009 regülatörü, güç kaynağı ihtiyacı olan birçok projede kullanılır. Özellikle taşınabilir elektronik cihazlar, güneş enerjisi uygulamaları ve düşük voltajlı pil sistemlerinde yaygın olarak tercih edilir. Arduino ve Elektronik Projeler: XL6009, Arduino veya diğer mikrodenetleyici projelerinde kullanılarak düşük voltajlı bir kaynağı daha yüksek bir voltaja çıkarmak için kullanılabilir.

Bu regülatör entegresi, geniş uygulama alanları ve ayarlanabilir çıkış voltajı özellikleri nedeniyle birçok elektronik projede kullanılmaktadır. Kullanıldığı yerlerden biri de, bir mikrodenetleyici tabanlı projede sabit bir voltaj kaynağı sağlamak veya pil seviyesini artırmak için kullanılmasıdır.

## G. Gerekli Kodlar

Gerekli kodların yüklenmesi için Arduino IDE platformunda çalışarak proje hazır hale getirildi.

```
#include <SPI.h>
#include <Wire.h>
#include <I2C_RTC.h>
#include <SD.h>
#include <LiquidCrystal_I2C.h>
#include <DHT.h>
#include <EEPROM.h>
#include <Ethernet2.h>
//-----GLOBAL
DEĞİSKENLER-----
float temp1,hum1;
int SET1;
float his=0.5;
float sonuc;
int ipA=10;
int ipB=192;
int ipC=3;
int ipD=252;
#define ethernet 10
```

```

#define DHTPIN 8
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal_I2C lcd(0x3F,16,2);
File sdkart;
static DS1307 RTC;
const int pinCS = 4;
int buzzerpin=7;
int sddongu = 0;
int ekranyenile = 0;
int sdsira = 1;
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
IPAddress ip(ipA,ipB,ipC,ipD);
EthernetClient client;
EthernetServer server(80);

//#####
#####

void setup() {
  Serial.begin(9600);
  pinMode(pinCS, OUTPUT);
  pinMode(ethernet, OUTPUT);
  pinMode(buzzerpin, OUTPUT);
  digitalWrite(pinCS, HIGH);
  digitalWrite(ethernet, HIGH);
  lcd.init();
  lcd.init();
  lcd.backlight();
  dht.begin();
  RTC.begin();
  SD.begin();
  SET1 = EEPROM.read(20);
  int ipi1 =EEPROM.read(25);
  int ipi2 =EEPROM.read(26);
  int ipi3 =EEPROM.read(27);
  int ipi4 =EEPROM.read(28);
  int hours =EEPROM.read(29);
  int minutes = EEPROM.read(30);
  int gunn = EEPROM.read(31);
  int ayy = EEPROM.read(32);
  int yill = EEPROM.read(33);

  server.begin();
  Ethernet.begin(mac,ip);
  Serial.println(Ethernet.localIP());
}

```

```

lcd.setCursor(2, 0);
lcd.print("firm.ver 1.00");
lcd.setCursor(0, 1);
lcd.print(Ethernet.localIP());
delay(4000);
AnaEkran();

if (SD.begin(pinCS))
{
    Serial.println("SD kart tanındı..");
}
else {
    Serial.println("SD kart bulunamadı.");
    return;
}

sdkart = SD.open("Sicaklik.txt", FILE_WRITE);

sdkart.print("SIRA");
sdkart.print(" ");
sdkart.print("TARİH");
sdkart.print(" ");
sdkart.print ("SAAT");
sdkart.print(" ");
sdkart.print("T");
sdkart.print(" ");
sdkart.println("H");
sdkart.close();
}

void loop() {

    float temp1 = dht.readTemperature();
    float hum1 = dht.readHumidity();
    Web();
    sddongu++;
    ekranyenile++;
    if (ekranyenile == 10000) {
        ekranyenile = 0;
        AnaEkran();
    }
    if (sddongu == 20000) {
        sddongu = 0;
        sdsira++;
        Sdkart();
    }
}

```

```

}
void Sdkart() {
  sdkart = SD.open("Sicaklik.txt", FILE_WRITE);
  if (sdkart)
  {
    float temp1 = dht.readTemperature();
    float hum1 = dht.readHumidity();

    sdkart.print(sdsira);
    sdkart.print(" ");
    sdkart.print(RTC.getDay());
    sdkart.print("/");
    sdkart.print(RTC.getMonth());
    sdkart.print("/");
    sdkart.print(RTC.getYear());
    sdkart.print(" ");
    sdkart.print(RTC.getHours());
    sdkart.print(":");
    sdkart.print(RTC.getMinutes());
    sdkart.print(":");
    sdkart.print(RTC.getSeconds());
    sdkart.print(" ");
    sdkart.print(temp1);
    sdkart.print(" ");
    sdkart.print("%");
    sdkart.println(hum1);
    sdkart.println("-----");
    -----");
    sdkart.close();
  }

  else
  {
    Serial.println("data.txt açılmadı.");
  }
}

//#####
#####

void Web(){

float temp1 = dht.readTemperature();
float hum1 = dht.readHumidity();
String set11;
int set11i;
String HTTP_req;
String ipp;

```

```

int i;
String times;
int hours,minutes,gunn,ayy,yill;

EthernetClient client = server.available();

Ethernet.maintain();
while (client.connected()) {
  if (client.available()) {
    char c = client.read();

    HTTP_req += c;
    if (!client.available()) {
      if (HTTP_req.indexOf("POST / HTTP/1.1") > -1){
        i = HTTP_req.indexOf("SET11=");
        if ( i > -1) { // see if URL ends in our get request
          //len = HTTP_req.length();
          set11 = HTTP_req.substring(i + 6);
          Serial.println(set11);
          set11i = set11.toInt();
          if (set11i > 0)
          {
            SET1 = set11i;
            EEPROM.write(20,SET1);
            sonuc =(SET1-his);
            Serial.println(sonuc);
          }
        }
      }
      //Serial.println(HTTP_req);
      i = HTTP_req.indexOf("ipayari=");
      if(i>-1)
      {
        Serial.println(i);
        ipp=HTTP_req.substring(i+9);
        Serial.println(ipp);
        i = ipp.indexOf(".");
        if(i>-1)
        {
          String ip1 = ipp.substring(0,i);
          Serial.println("ip1:");
          Serial.println(ip1);

          int ipi1=ip1.toInt();
          int i2 = ipp.indexOf(".", i+1);

          if(i2>-1)
          {

```

```

String ip2 = ipp.substring(i+1,i2);
Serial.println("ip2:");
Serial.println(ip2);

int ipi2=ip2.toInt();
int i3 = ipp.indexOf(".", i2+1);

if(i3>-1)
{
String ip3 = ipp.substring(i2+1,i3);
Serial.println("ip3:");
Serial.println(ip3);

int ipi3=ip3.toInt();

String ip4 = ipp.substring(i3+1);

Serial.println("ip4:");
Serial.println(ip4);

int ipi4=ip4.toInt();

if(ipi1>=0 && ipi2>=0 && ipi3>=0 && ipi4>=0 && ipi1<=255
&& ipi2<=255 && ipi3<=255 && ipi4<=255)
{

EEPROM.write(25,ipi1);
EEPROM.write(26,ipi2);
EEPROM.write(27,ipi3);
EEPROM.write(28,ipi4);
ipA=ipi1;
ipB=ipi2;
ipC=ipi3;
ipD=ipi4;

IPAddress ip(ipA,ipB,ipC,ipD);
server.begin();
Ethernet.begin(mac,ip);
Serial.println(Ethernet.localIP());

delay(30);
}
}
}
}
}

i = HTTP_req.indexOf("appt=");

```



```

if(i>-1)
{
    times=HTTP_req.substring(i+5);

    i = times.indexOf("%3A");

    if(i>-1)
    {
        String hour=times.substring(0,i);
        Serial.println(hour);
        Serial.println(i);
        int hours=hour.toInt();

        String min = times.substring(i+3);
        Serial.println(min);
        int minutes=min.toInt();

        if(minutes >-1 && hours >-1)
        {
            EEPROM.write(29,hours);
            EEPROM.write(30,minutes);
            RTC.begin();
            RTC.setTime(hours,minutes,00);

        }

    }

}

i = HTTP_req.indexOf("birthday=");
if(i>-1)
{
    String date = HTTP_req.substring(i+9);
    Serial.println(date);
    i = date.indexOf("-");
    if(i>-1)
    {
        String yil= date.substring(0,i);
        Serial.println(yil);
        int yill= yil.toInt();
        int i2 = date.indexOf("-", i+1);
        if(i2>-1)
        {
            String ay = date.substring(i+1,i2);

```

```

        Serial.println(ay);
        int ayy = ay.toInt();

        String gun = date.substring(i2+1);
        Serial.println(gun);
        int gunn = gun.toInt();
        if (gunn >-1 && ayy > -1 && yill >-1)
        {
            EEPROM.write(31,gunn);
            EEPROM.write(32,ayy);
            EEPROM.write(33,yill);
            RTC.begin();
            RTC.setDate(gunn,ayy,yill);

        }
    }
}

}

client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println("Connection: close");
client.println("Refresh: 25");
client.println();
client.println("<!DOCTYPE HTML>");
client.println("<html lang='tr'>");
client.println("<head>");
client.println("<title>TEMPRATURE DEVICE </title>");
client.print("<style>");
client.print("body {background-color: #c0c0c0; text-align:
center; border: 2px solid black; width:1000px;height:800px;
margin:auto;}");
client.print("th,h2 {background-color: #3333FF;}");
client.print("table, th, td { border: 5px solid black;
margin:auto;}");
client.print(".f { border: 2px solid black;
width:200px;height:100px; font-size:40px;background-color:
#3333FF; margin:auto;}");
client.print(".c { border: 2px solid black;
width:200px;height:100px; font-size:40px;background-color:
#3333FF; margin:auto;}");
client.print("@media screen and (max-width:900px){
h1,body,table,th,td,.f,.c{font-size:40px;}}");
client.print("@media screen and (max-width:600px){
h1,body,table,th,td,.f,.c{font-size:30px;}}");

```

```

        client.print("@media screen and (max-width:400px){
h1,body,table,th,td,.f,.c{font-size:20px;}}");
        client.print(".box {display:none;}#trigger:checked
+.box{display:block;}#trigger {display:none;}");
        client.print(".dugme {back-ground-color:silver; 1px
solid;}");
        client.print("</style>");
        client.println("</head>");

        client.println("<body>");
        client.println("<h1>TEMPRATURE DEVICE </h1>");
        client.println("<hr><br>");
        client.println("<table>");
        client.println("<tr>");
        client.print("<th>SENSORLER</th>");
        client.print("<th>SICAKLIK DEGERI</th>");
        client.print("<th>NEM DEGERI</th>");
        client.print("<th>SET</th>");
        client.print("<th>DURUM</th>");
        client.print("</tr>");

        client.println("<tr>");
        client.print("<th><u>SICAKLIK 1</u></th>");
        client.print("<td>");
        client.print(temp1);
        client.print("</td>");
        client.print("<td>");
        client.print("%");
        client.print(hum1);
        client.print("</td>");
        client.print("<td>");
        client.print(SET1);
        client.print("</td>");
        if(temp1>=sonuc)
        {
        digitalWrite(buzzerpin,HIGH);
        client.print("<td>");
        client.print("ALARM");
        client.print("</td>");
        }

        if(temp1<sonuc)
        {
        digitalWrite(buzzerpin,LOW);
        client.print("<td>");
        client.print("-");
        client.print("</td>");
        }

```

```

client.print("</tr>");
client.println("</table>");
client.println("<br><br><br>");
client.print("<div class='c'>");
client.print(RTC.getHours());
client.print(":");
client.print(RTC.getMinutes());
client.print(":");
client.print(RTC.getSeconds());
client.println("<br>");
client.print(RTC.getDay());
client.print("/");
client.print(RTC.getMonth());
client.print("/");
client.print(RTC.getYear());
client.print("</div>");
client.println("<br>");

client.print("<div class = 'set'>");
client.println("<form method= \"post\">");
client.print("<input type='number' id='SET11' name='SET11'
min='1' max='60' value=''>");
client.print(SET1);
client.println(">");
client.print("<input type='submit' value='SET'>");
client.println("</form>");

client.print("<label for='trigger' class='dugme'>
Ayarlar</label>");
client.print("<input id='trigger' type='checkbox'>");
client.print("<div class='box'><br>");
client.println("<form method= \"post\">");
client.print("IP Adres: <input type='text' id='ipayari'
name='ipayari' value=' '>");
client.print(Ethernet.localIP());
client.print(">");
client.print("<input type='submit' value='Apply'><br>");
client.println("</form>");

client.println("<form method= \"post\">");
client.print("<label for='appt'>Saat:</label>");
client.print("<input type='time' id='appt' name='appt'
value=''>");
client.print(RTC.getHours());
client.print(":");
client.print(RTC.getMinutes());
client.print(">");

```

```

        client.print("<input type='submit' value='Apply'><br>");
        client.println("</form>");

        client.println("<form method=\"post\">");
        client.print("<label for='birthday'>Tarih:</label>");
        client.print("<input type='date' id='birthday'
name='birthday' value =\"");
        client.print(RTC.getYear());
        client.print("-");
        client.print(RTC.getMonth());
        client.print("-");
        client.print(RTC.getDay());
        client.print(">");
        client.print("<input type='submit' value='Apply'><br>");
        client.print("</div>");
        client.println("</form>");

        client.println("<br><br><br>");
        client.println("<a><img
src='https://baknet.com.tr/images/logo.png' alt='Baknet'/></a>");
        client.println("</body></html>");
        client.stop();
        HTTP_req = "";
        break;
    }
}
}
delay(1);

Ethernet.maintain();
}

void AnaEkran()
{
    lcd.clear();
    float temp1 = dht.readTemperature();
    float hum1 = dht.readHumidity();

    lcd.setCursor(0, 1);
    lcd.print("T:");
    lcd.print(temp1);

    lcd.setCursor(8, 1);
    lcd.print("H:");
    lcd.print("%");
    lcd.print(hum1);

    lcd.setCursor(11, 0);

```

```
lcd.print(RTC.getHours());  
lcd.print(":");  
lcd.print(RTC.getMinutes());  
  
lcd.setCursor(0, 0);  
lcd.print(RTC.getDay());  
lcd.print("/");  
lcd.print(RTC.getMonth());  
lcd.print("/");  
lcd.print(RTC.getYear());  
}
```

Bu çalışmaları yaparken kodların düzenlenmesi, syntax gibi komutların hatırlanması ve düzenleme gibi konularda yapay zekadan destek alınmıştır. [8]

### 3.3 Önceki Çalışmalar

Genel olarak sıcaklık ve nem ölçüm projeleriyle ilgili araştırmalar ve çalışmalar, birçok yıl öncesine dayanmaktadır. Ancak, özellikle bu konudaki yoğun AR-GE çalışmalarının ve projelerinin sayısı zaman içinde artmıştır. İşte sıcaklık ve nem ölçümü ile ilgili projelerde dünyada gerçekleştirilen bazı önceki çalışmalara yönelik birkaç örnek:

**Sensör Teknolojileri ve Veri Toplama Sistemleri:** Sıcaklık ve nem ölçümü için kullanılan sensör teknolojileri üzerine birçok araştırma yapılmıştır. Özellikle daha hassas, doğru ve dayanıklı sensör tasarımları geliştirmeye odaklanan çalışmalar bulunmaktadır.

**Endüstriyel Otomasyon ve Kontrol Sistemleri:** Sıcaklık ve nem ölçümü, endüstriyel otomasyon ve kontrol sistemlerinde kritik bir rol oynamaktadır. Bu alanda yapılan çalışmalar, endüstriyel süreçlerin verimliliğini artırmaya yönelik sensör teknolojilerini ve kontrol sistemlerini içermektedir.

**Tarım ve Çevre Uygulamaları:** Tarım sektöründe sıcaklık ve nem ölçümü, bitki büyüme koşulları üzerinde önemli bir etkiye sahiptir. Bu bağlamda, tarım alanında yapılan projeler, bitki sağlığını ve verimliliğini artırmak amacıyla sıcaklık ve nem sensörleri kullanımını ele almaktadır.

**Bina Otomasyonu ve Enerji Yönetimi:** Bina otomasyon sistemleri ve enerji yönetimi, sıcaklık ve nem ölçümü ile entegre bir şekilde çalışan projeleri içermektedir. Bu tür çalışmalar, enerji verimliliğini artırmak ve konfor seviyelerini optimize etmek amacıyla gerçekleştirilmiştir.

**Telekomünikasyon ve Veri Merkezi Yönetimi:** Sıcaklık ve nem ölçümü, telekomünikasyon ekipmanları ve veri merkezlerinde kritik öneme sahiptir. Bu alanlarda yapılan projeler, ekipmanların güvenliği ve performansını artırmaya yönelik sensör teknolojilerini içermektedir.

**IoT (Internet of Things) ve Akıllı Şehir Projeleri:** Sıcaklık ve nem sensörleri, IoT ve akıllı şehir projelerinde kullanılan temel unsurlardan biridir. Bu tür projeler, çevresel koşulları izlemek, enerji verimliliğini artırmak ve genel yaşam kalitesini iyileştirmek amacıyla sıcaklık ve nem ölçüm sistemlerini entegre etmektedir.

Bu örnekler, sıcaklık ve nem ölçüm projelerinin geniş bir uygulama yelpazesine sahip olduğunu ve dünya genelinde birçok sektörde kullanıldığını göstermektedir. Tezinzide, belirli bir uygulama alanına odaklanarak veya sensör teknolojileri üzerinde detaylı bir inceleme yaparak bu genel bilgileri daha spesifik bir şekilde ele alabilirsiniz.

### 3.4 Gelecekteki Araştırmalar için Öneriler

Gelecekte sıcaklık ve nem ölçüm projeleri alanında yapılacak çalışmalara yönelik öneriler şunlar olabilir:

**Hassasiyet ve Doğruluk İyileştirmeleri:** Sensör teknolojilerinin geliştirilerek daha hassas ve doğru ölçümler yapabilen sensörlerin tasarlanması önemlidir. Bu, özellikle endüstriyel ve bilimsel uygulamalarda kullanılan sensörler için kritiktir.

**Enerji Verimliliği Odaklı Çalışmalar:** Sensör sistemlerinin enerji tüketimini optimize etmek ve düşük güçlü cihazlar için daha etkili enerji yönetimi stratejileri geliştirmek, uzun pil ömrü ve enerji verimliliği açısından önemli bir araştırma alanıdır.

**Kablosuz İletişim Teknolojileri:** Veri iletimi için kablosuz iletişim teknolojileri üzerine odaklanan çalışmalar, sensör ağları arasındaki iletişimi geliştirerek daha geniş kapsamlı projelerin uygulanmasına olanak tanıyabilir.

Veri Analizi ve Yapay Zeka Uygulamaları: Toplanan sıcaklık ve nem verilerinin etkili bir şekilde analiz edilebilmesi için yapay zeka (AI) ve veri analizi yöntemlerine odaklanan çalışmalar, daha akıllı ve öğrenen sistemlerin tasarımını mümkün kılabilir.

Çoklu Sensör Entegrasyonu: Farklı fiziksel özellikleri ölçebilen çeşitli sensörlerin entegrasyonunu içeren çalışmalar, çoklu parametreleri aynı anda ölçebilen daha kapsamlı sensör sistemlerinin geliştirilmesine katkı sağlayabilir.

Uzaktan İzleme ve Kontrol: Uzaktan izleme ve kontrol sistemlerine odaklanan çalışmalar, sensör verilerinin uzaktan toplanması, izlenmesi ve kontrol edilmesi için daha etkili yöntemlerin tasarlanmasına yönelik olabilir.

Çevresel Uygulama Alanları: Sıcaklık ve nem ölçüm projelerinin çevresel etkileşimler üzerindeki etkilerini anlamak amacıyla çeşitli uygulama alanlarına odaklanan çalışmalar, özellikle tarım, su kaynakları yönetimi ve çevresel izleme konularında değerli olabilir.

Güvenlik ve Gizlilik: Veri güvenliği ve gizliliği konusundaki endişeleri ele alacak çalışmalar, özellikle bu tür sensör sistemlerinin kullanıldığı endüstriyel ve kişisel uygulamalarda önemlidir.

Bu öneriler, sıcaklık ve nem ölçüm projelerinin daha gelişmiş, verimli ve çeşitli uygulamalarda kullanılabilir hale getirilmesine yönelik genel bir çerçeve sunmaktadır.



# Bölüm 4

## Metodoloji

### 4.1 Araştırma Deseni

Bu tez, endüstriyel ve ticari uygulamalarda kullanılan sıcaklık ve nem ölçümlerini daha etkili bir şekilde yönetmek ve izlemek amacıyla Power over Ethernet (POE) beslemeli akıllı sıcaklık ve nem datalogger'larının geliştirilmesine odaklanmaktadır. Bu cihazlar, kullanıcıların uzaktan erişim sağlayabilecekleri, sıcaklık ayarlarını kontrol edebilecekleri ve veri kayıtlarını web üzerinden izleyebilecekleri bir çözüm sunmaktadır.

Araştırma Soruları:

POE beslemeli sıcaklık ve nem datalogger'larının mevcut pazarlardaki durumu nedir?

Bu cihazlar, enerji verimliliği, uzaktan yönetim ve kullanıcı dostu arayüz açısından nasıl geliştirilebilir?

Sıcaklık ve nem ölçümlerini doğru ve güvenilir bir şekilde sağlamak için sensör teknolojileri nasıl optimize edilebilir?

Araştırma Yöntemi:

Bu tez, hem literatür taraması hem de deneysel çalışmalar içerecektir. İlk aşamada, mevcut POE beslemeli datalogger'larının incelenmesi ve bu cihazların performans özelliklerinin değerlendirilmesi yapılacaktır. Ardından, önerilen geliştirmelerin bir prototip üzerinde test edilmesi planlanmaktadır.

Veri Toplama:

Prototip cihazlar kullanılarak laboratuvar ortamında sıcaklık ve nem ölçümleri yapılacaktır. Ayrıca, kullanıcı geri bildirimini almak amacıyla test kullanıcıları ile bir anket uygulanacaktır.

Analiz ve Sonuçlar:

Elde edilen veriler, istatistiksel analizler ve karşılaştırmalarla değerlendirilecek, geliştirmelerin etkinliği ve kullanıcı memnuniyeti üzerindeki etkisi analiz edilecektir.

Sonuçlar ve Öneriler:

Tezin sonuçları, POE beslemeli akıllı sıcaklık ve nem datalogger'larının performansını artırmak için öneriler içerecek ve bu cihazların gelecekteki uygulamalarda nasıl daha etkili bir şekilde kullanılabileceğine dair önerilere odaklanacaktır.

## 4.2 Araştırma Popülasyonu ve Örneklem

Bu tez, endüstriyel tesislerde sıcaklık ve nem kontrolünü sağlamak üzere kullanılan sıcaklık ve nem datalogger'larının performansını değerlendirmeyi amaçlamaktadır. Genel popülasyon, farklı endüstri sektörlerine ait ve çeşitli coğrafi konumlarda bulunan endüstriyel tesisleri içermektedir. Örneklem, belirli endüstri sektörleri ve coğrafi konumları temsil eden bir grup tesis içermektedir. Bu tesislerde kullanılan datalogger'lar, belirli bir süre boyunca sıcaklık ve nem ölçümleri gerçekleştirerek performansları değerlendirilecektir. Elde edilen veriler, örneklemdaki tesislerin datalogger'larını kapsamlı bir şekilde analiz ederek, genel popülasyona dair bilgiler sağlayacak ve endüstriyel tesislerde bu teknolojinin etkinliği hakkında değerli bilgiler sunacaktır.

Örnekleimde yer alan tesisler, belirli bir süre boyunca datalogger'larını kullanarak sıcaklık ve nem ölçümleri gerçekleştirecek ve bu verilerin analizi, datalogger'ların performansının değerlendirilmesi için kullanılacaktır. Bu tez, endüstriyel tesislerde

sıcaklık ve nem ölçüm teknolojilerinin etkinliği konusunda hem akademik hem de pratik bir katkı sağlamayı amaçlamaktadır.

Endüstriyel tesislerde sıcaklık ve nem kontrolü ihtiyacını karşılamak üzere kullanılan sıcaklık ve nem datalogger'larının performansını inceleyerek, genel bir popülasyonu kapsamaktadır. Bu genel popülasyon, farklı endüstri sektörlerinden çeşitli tesisleri içermekte olup, gıda üretiminden enerji üretimine kadar geniş bir yelpazede faaliyet gösteren endüstriyel tesisleri kapsamaktadır. Ayrıca, coğrafi olarak farklı bölgelerde yer alan bu tesisler, popülasyonun çeşitliliğini artırmaktadır. Bu geniş kapsamlı popülasyon, sıcaklık ve nem datalogger'larının endüstriyel uygulamalarda ne kadar yaygın olarak kullanıldığını ve bu teknolojinin genel performansını anlamak açısından önemli bir temsil sağlamaktadır.

### 4.3 Veri Toplama Yöntemleri

Uzaktan izleme ve veri kayıtları, endüstriyel tesislerde sıcaklık ve nem datalogger'ları üzerinde gerçek zamanlı gözlem yapmak ve uzaktan veri toplamak için güçlü bir yöntemdir. Bu yöntem, datalogger'ların anlık ölçümlerini takip etmek, cihazın çalışma durumunu izlemek ve veri akışını uzaktan kontrol etmek amacıyla kullanılır. Özellikle internet bağlantılı datalogger'lar sayesinde, kullanıcılar herhangi bir coğrafi konumdan cihazlara erişebilir ve anlık verilere ulaşabilir. Ayrıca, uzaktan izleme sistemi üzerinden elde edilen veri kayıtları, belirli bir zaman dilimindeki sıcaklık ve nem değişimlerini analiz etmeye olanak tanır. Bu yöntem, datalogger'ların performansını değerlendirmek, olası sorunları tespit etmek ve veri güvenilirliğini sağlamak için etkili bir araç sunar. Veri kayıtları, endüstriyel tesislerde sıcaklık ve nem yönetimi alanında daha iyi anlayış ve verimlilik sağlamak adına kapsamlı analizlere imkân verir.

Uzaktan izleme ve veri kayıtları, endüstriyel tesislerde kullanılan sıcaklık ve nem datalogger'ları üzerine yapılan araştırmalarda önemli bir veri toplama yöntemidir. Bu yöntem, datalogger'ların gerçek zamanlı performansını izlemek ve veri kayıtlarını almak için kullanılır. Uzaktan izleme, araştırmacıların datalogger'ların sensörlerinden gelen verilere web tabanlı bir arayüz veya özel bir yazılım üzerinden erişmesini sağlar. Bu sayede, tesis uzmanları veya araştırmacılar, datalogger'ların anlık sıcaklık ve nem değerlerini, hava koşullarını ve sistem durumlarını takip edebilirler. Ayrıca, uzaktan

izleme sistemi, datalogger'ların yanıt verme sürelerini, veri kayıt aralıklarını ve bağlantı durumlarını izlemek için de kullanılabilir. Veri kayıtları, uzaktan izleme sistemi tarafından toplanan verilerin kaydedilmesini ve arşivlenmesini içerir. Bu veriler daha sonra analiz edilerek, tesisin uzun vadeli performansı, enerji verimliliği ve çeşitli işletme koşullarına yanıt verme yeteneği gibi önemli metrikler değerlendirilebilir. Uzaktan izleme ve veri kayıtları, endüstriyel tesislerde sıcaklık ve nem kontrol sistemlerinin etkinliğini anlamak ve iyileştirmek için değerli bir araştırma aracı sağlar.

Alan çalışması ve deneyler, endüstriyel tesislerde sıcaklık ve nem datalogger'ları kullanımını araştırırken doğrudan saha koşullarında veri toplamak ve cihazların performansını değerlendirmek için etkili bir yöntemdir. Bu tür bir araştırma, belirli endüstriyel tesislerde datalogger'ların yerleştirilme stratejilerini ve kullanım pratiklerini anlamak amacıyla saha ziyaretleri içerir. Araştırmacılar, datalogger'ların fiziksel konumlarını, çevresel etmenlere maruz kaldıkları koşulları ve ölçümleri gerçekleştirdikleri alanları belirlemek için tesislerde saha çalışmaları yaparlar. Aynı zamanda, kontrol altında tutulan deneyler aracılığıyla, farklı sıcaklık ve nem seviyelerinde datalogger'ların tepkileri değerlendirilebilir. Bu deneyler, datalogger'ların hassasiyetini, yanıt sürelerini ve veri kayıt doğruluğunu belirleme açısından kritik bilgiler sağlar. Alan çalışmaları, datalogger'ların saha koşullarında ne kadar etkili olduğunu anlamak ve cihazların gerçek dünya uygulamalarında karşılaştığı potansiyel zorlukları belirlemek için önemlidir. Elde edilen veriler, endüstriyel tesislerde sıcaklık ve nem kontrolü için datalogger'ların daha etkili ve uygun şekilde kullanılmasına yönelik stratejilerin geliştirilmesine katkı sağlayabilir.

## 4.4 Veri Analizi Yöntemleri

Bu projede, endüstriyel tesislerde kullanılan sıcaklık ve nem datalogger'larından alınan verilerin web tabanlı bir sistem üzerinden takip edilmesi ve analiz edilmesi hedeflenmiştir. Datalogger'lar tarafından toplanan veriler, SD kart üzerine kaydedilmekte ve bu veriler web tabanlı bir platforma düzenli aralıklarla yüklenmektedir. Bu metin, bu web tabanlı takip sisteminin veri analizi sürecini ve kullanıcı dostu bir arayüz üzerinden sunulan bilgileri içermektedir.

### Veri Depolama ve SD Kart Kayıtları:

Datalogger'lar tarafından toplanan sıcaklık ve nem verileri, düzenli aralıklarla SD kart üzerine kaydedilmektedir. SD kart üzerindeki kayıtlar, her bir ölçüm noktasının tarih ve saat bilgisiyle birlikte detaylı bir şekilde depolanmaktadır.

### Veri Yükleme ve Web Arayüzü:

SD kart üzerindeki kayıtlar, belirlenen periyodik aralıklarla web tabanlı bir sunucuya yüklenmektedir. Bu süreç, otomatik bir güncelleme mekanizması ile gerçekleşmektedir. Kullanıcılar, web arayüzüne giriş yaparak bu verilere erişebilirler.

### Grafiksel Analiz ve Raporlama:

Web tabanlı arayüz, kullanıcıların sıcaklık ve nem değerlerini grafiklerle görselleştirmelerine ve belirli bir zaman dilimindeki trendleri analiz etmelerine olanak tanır. Kullanıcılar, istedikleri tarih aralıklarını seçerek detaylı analiz yapabilir ve trendleri takip edebilirler. Ayrıca, belirli sıcaklık veya nem limitlerinin aşılması durumunda otomatik uyarılar alabilirler.

### Kullanıcı Yönetimi ve Paylaşım:

Web tabanlı sistem, farklı kullanıcı profilleri oluşturarak tesis içindeki ilgili personelin belirli datalogger'ların verilerini görmesine izin verir. Ayrıca, raporları ve analiz sonuçlarını paylaşmak için kullanıcılar arasında etkileşimli bir platform sağlar. [1]

# Bölüm 5

## Beklenen Katkılar

### Enerji Verimliliği ve Kaynak Tasarrufu:

Proje, endüstriyel tesislerde sıcaklık ve nem kontrolünü optimize etmeye yönelik veri odaklı kararlar almayı mümkün kılabilir. Bu sayede tesisler, enerji tüketimini optimize ederek kaynak tasarrufu sağlayabilir.

### Performans İyileştirmesi:

Analiz edilen veriler, datalogger'lar ve kontrol sistemlerinin performansını değerlendirmek ve geliştirmek için kullanılabilir. Bu, tesislerin sıcaklık ve nem düzeylerini daha hassas bir şekilde kontrol etmelerine ve optimize etmelerine yardımcı olabilir.

### Uyarı ve Hata Tespiti:

Web tabanlı takip sistemi, belirlenen sıcaklık ve nem limitleri aşıldığında anlık uyarılar sağlayarak, olası sorunların hızlı bir şekilde tespit edilmesine katkıda bulunabilir. Bu, tesis yöneticilerine acil müdahale imkânı tanır.

### Veri Temelli Kararlar:

Proje, tesis yöneticilerine veri temelli kararlar almalarına yardımcı olabilir. Analiz edilen veriler, ekipman bakım stratejilerinin belirlenmesi, enerji tüketiminin optimize edilmesi ve operasyonel verimliliğin artırılması gibi konularda yöneticilere rehberlik edebilir.

### Uzaktan İzleme ve Yönetim:

Web tabanlı takip sistemi, tesis yneticilerinin ve personelinin uzaktan eriřim ile verilere ulařmalarını saęlar. Bu da, acil durum mdahaleleri ve genel sistem ynetimini kolaylařtırabilir.

Teknolojik Geliřmelerin Teřvik Edilmesi:

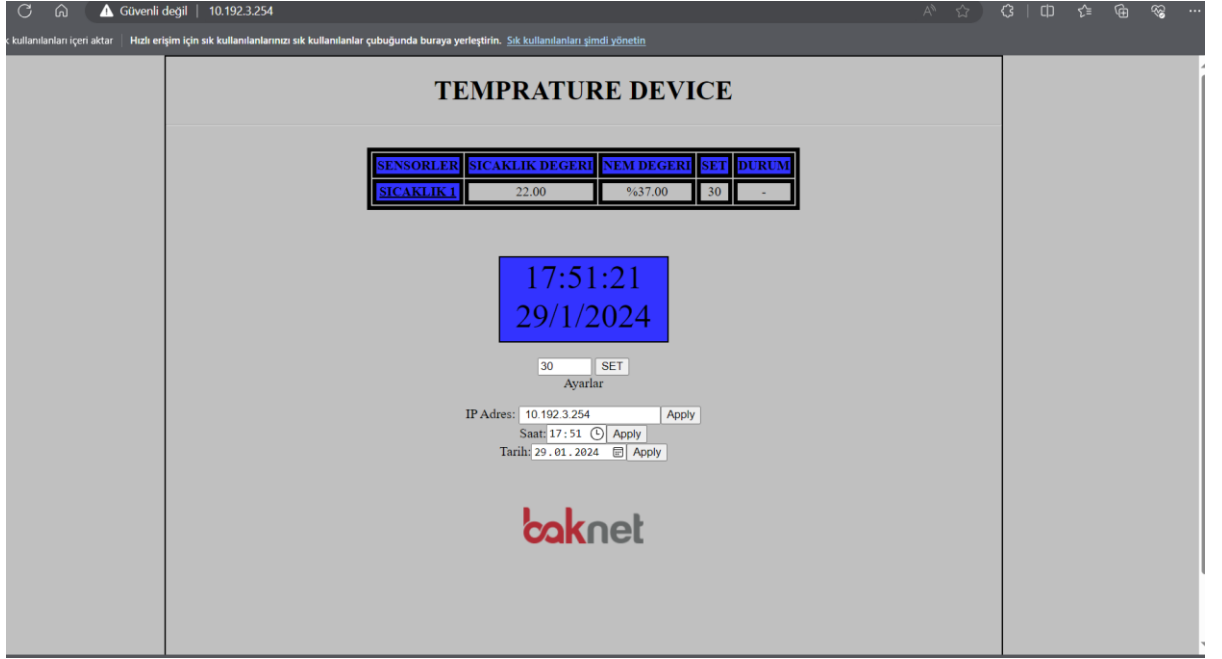
Proje, endstriyel tesislerde sıcaklık ve nem kontrol iin kullanılan teknolojilerin geliřtirilmesini teřvik edebilir. Bu, daha verimli, gvenilir ve evre dostu sistemlerin geliřtirilmesine olanak tanır.

Veri Paylařımı ve Endstri Standartları:

Proje, sektr iinde veri standartlarının oluřturulmasına ve paylařımına katkıda bulunabilir. Bu, farklı tesislerin ve sektrlerin benzer veri setlerini paylařmasını ve karřılařtırmasını kolaylařtırabilir.

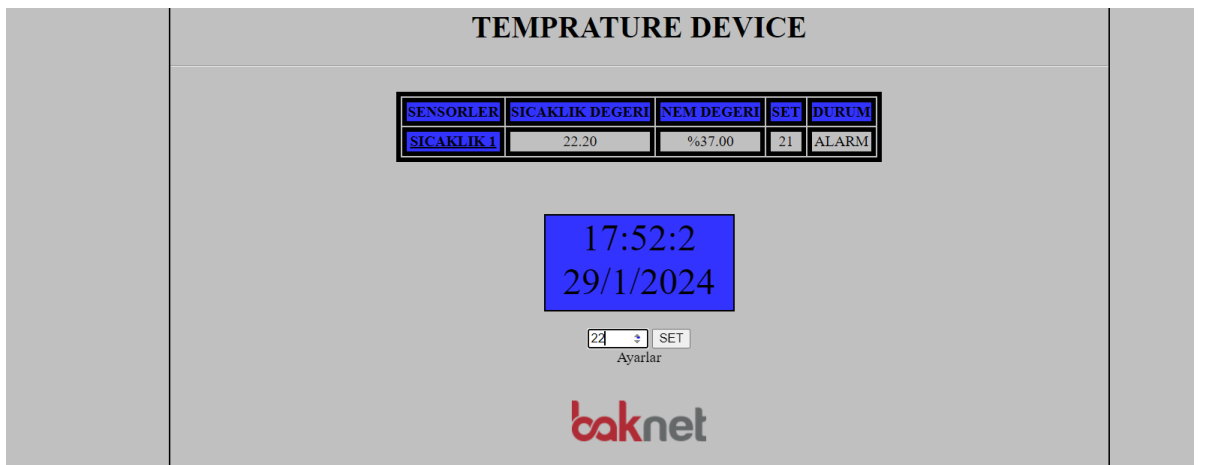
Bu beklenen katkılar, proje kapsamında elde edilen verilerin etkili bir Őekilde kullanılmasının ve endstriyel tesislerin daha srdrlebilir, gvenilir ve verimli hale gelmesinin ngrlen potansiyel yollarını temsil etmektedir.

## 5.1 Uygulama Görselleri



Şekil 5.1: Web Ayar Arayüzü

Şekil 3'te gösterildiği üzere, cihaza statik bir IP atayarak aynı VLAN üzerinden bağlantı sağlanarak, web tabanlı bir arayüz aracılığıyla IP yapılandırması, sıcaklık ayarları, zamanlama gibi çeşitli değerleri kontrol etmek mümkündür. Ayrıca, sıcaklık set değerinin aşıldığı durumlarda alarm durumuyla ilgili bilgileri ekranda okuma olanağı bulunmaktadır. [6]



Şekil 5.2: Web Set Arayüzü





Şekil 5.3: Datalogger

Şekil 5 ve 6 da görüldüğü üzere tasarımını tamamladığımız Dataloggerimiz ektedir. Yapılan testler sonucunda 4 ay kadar süre verileri tutmuş ve stabil olarak sorunsuz çalışmaya devam etmiştir.



Şekil 5.4: Datalogger

SICAKLIK.TXT - Not Defteri

Dosya	Düzen	Biçim	Görünüm	Yardım
23330	11/10/2023	13:34:14	27.20	%35.00
23331	11/10/2023	13:34:42	27.10	%35.00
23332	11/10/2023	13:35:11	27.10	%35.00
23333	11/10/2023	13:35:39	27.20	%35.00
23334	11/10/2023	13:36:7	27.10	%35.00
23335	11/10/2023	13:36:36	27.20	%35.00
23336	11/10/2023	13:37:4	27.20	%35.00
23337	11/10/2023	13:37:32	27.20	%35.00
23338	11/10/2023	13:38:1	27.10	%35.00
23339	11/10/2023	13:38:29	27.20	%35.00
23340	11/10/2023	13:38:57	27.10	%35.00
23341	11/10/2023	13:39:26	27.20	%35.00
23342	11/10/2023	13:39:54	27.20	%35.00
23343	11/10/2023	13:40:22	27.10	%35.00
23344	11/10/2023	13:40:51	27.20	%35.00
23345	11/10/2023	13:41:19	27.20	%35.00
23346	11/10/2023	13:41:47	27.20	%35.00
23347	11/10/2023	13:42:16	27.20	%35.00
23348	11/10/2023	13:42:44	27.20	%35.00
23349	11/10/2023	13:43:12	27.20	%35.00
23350	11/10/2023	13:43:41	27.20	%35.00

Şekil 5.5: SD Kart Log Kayıtları

Şekil 5.5 'de SD karta loglarını kaydettiğimiz veriler görülmektedir.

# Bölüm 6

## Sonuç

Datalogger teknolojisinin kullanımı, çeşitli sektörlerde çevresel parametrelerin sürekli izlenmesi ve kaydedilmesi yoluyla önemli sonuçlar doğurur. Endüstriyel üretimde, sıcaklık, nem ve basınç gibi kritik faktörlerin anlık ölçümleri, üretim süreçlerinin kalitesini artırarak ürün güvenilirliğini sağlar. Tarım alanında datalogger'lar, toprak nem içeriği, hava sıcaklığı ve ışık seviyeleri gibi ölçümleri kullanarak bitki sağlığını optimize etmeye yardımcı olur, bu da verimliliği artırır ve su kullanımını optimize edebilir. Meteoroloji uygulamalarında kullanıldığında, datalogger'lar sürekli hava durumu verilerini kaydederek doğru hava tahminleri ve iklim değişiklikleri analizi sağlar.

Enerji yönetimi bağlamında datalogger'lar, binalarda ve enerji tesislerinde enerji tüketimini izleyerek enerji verimliliği programlarının etkili bir şekilde yönetilmesine katkıda bulunabilir. Çevresel izleme ve araştırma projelerinde kullanılan datalogger'lar, su kalitesi, hava kirliliği düzeyleri gibi önemli çevresel faktörleri sürekli olarak kaydederek çevresel değişiklikleri anlama ve çevre koruma çabalarını destekleme konusunda kritik bir rol oynar. Ayrıca, uzun vadeli trend analizleri için toplanan veriler, belirli bir süre zarfında ortaya çıkan desenleri anlama ve gelecekteki değişimlere daha iyi uyum sağlama açısından önemlidir. Tüm bu uygulama alanları, datalogger teknolojisinin çeşitli sektörlerde verimliliği artırmak, güvenilirliği optimize etmek ve çevresel sürdürülebilirliği desteklemek adına güçlü bir araç olduğunu göstermektedir.

# Kaynaklar

- [1]Osman D.(2019). Eğitimde Arduino Kullanımı İle İlgili Yapılan Çalışmalar. XII. Uluslararası Eğitim Araştırmaları Kongresi,(490-491)
- [2]Mehmet T.(2019). Nesnelerin İnterneti Tabanlı Akıllı Sulama ve Uzaktan İzleme Sistemi.Avrupa Bilim ve Teknoloji Dergisi,(230-232)
- [3]Delebe, E. (2020). Arduino Projeleri: 17. Cilt (s. 113-120). Kodlam.com.
- [4]Kahyaoğlu, M., & Dede, F. (2018). Kitap Başlığı 3. Baskı ( s. 30-35). Pusula Yayınları.
- [5]Practical Control(2015) HT20-HT20IoT (s. 1-8)  
[https://www.practicalcontrol.com.au/pdf\\_files/humidity/HT20-HT20IoT-data-sheet.pdf](https://www.practicalcontrol.com.au/pdf_files/humidity/HT20-HT20IoT-data-sheet.pdf)
- [6]Lumel.(2015) Servis Kılavuzu  
[https://www.lumel.com.pl/resources/Pliki%20do%20pobrania/HT20/HT20\\_service\\_manual.pdf](https://www.lumel.com.pl/resources/Pliki%20do%20pobrania/HT20/HT20_service_manual.pdf)
- [7]Arduino. (2012). Arduino Ethernet Shield (POE Modülü Olmadan). Arduino.  
<https://docs.arduino.cc/retired/shields/arduino-ethernet-shield-without-poe-module/>
- [8]OpenAI. (2023). OpenAI ChatGPT.  
<https://chat.openai.com/>

