

3.914

**TÜRKİYE CUMHURİYETİ**  
**YILDIZ TEKNİK ÜNİVERSİTESİ**  
**ELEKTRİK - ELEKTRONİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**VERİ MERKEZİ OTOMASYON SİSTEMİ**

Sinan Can Keskin

**BİLGİSAYAR PROJESİ**

Danışman  
Doç. Dr. Ali Gökhan YAVUZ

Ocak, 2017



## TEŞEKKÜR

---

Bu çalışmanın yürütülmesi sırasında desteğini esirgemeyen danışmanım Sn. Doç. Dr. Ali Gökhan Yavuz'a, projeyi beraber geliştirdiğim arkadaşım Alikemalcem Özdemir'e, tecrübelerini benimle paylaşmaktan çekinmeyen kıymetli dostum Edin Ramani'ye, fikirleri benim için her zaman değerli olacak olan ağabeyim Emin Keskin'e, maddi ve manevi desteğine her zaman ihtiyaç duyacağım aileme teşekkür ederim.

Sinan Can Keskin

# İÇİNDEKİLER

---

<b>SİMGE LİSTESİ</b>	<b>v</b>
<b>KISALTMA LİSTESİ</b>	<b>vi</b>
<b>ŞEKİL LİSTESİ</b>	<b>vii</b>
<b>TABLO LİSTESİ</b>	<b>viii</b>
<b>1 Giriş</b>	<b>1</b>
1.1 Ön İnceleme . . . . .	1
1.2 Fizibilite . . . . .	2
1.2.1 Teknik Fizibilite . . . . .	2
1.2.2 Zaman Fizibilitesi . . . . .	6
1.2.3 Yasal Fizibilite . . . . .	7
1.2.4 Ekonomik Fizibilite . . . . .	7
<b>2 Sistem Analizi ve Mimari Tasarım</b>	<b>9</b>
2.1 Elektronik Devre Tasarımı . . . . .	9
2.1.1 Arduino Uno ve NRF24L01 Devre Şeması . . . . .	9
2.1.2 Arduino Uno ve DHT11 Nem ve Sıcaklık Sensörü . . . . .	10
2.1.3 Arduino ve BMP180 . . . . .	10
<b>3 Sistem Tasarımı</b>	<b>12</b>
3.1 Yazılım Tasarımı . . . . .	13
3.1.1 Arduino Yazılım Tasarımı . . . . .	13
3.1.2 RPi Yazılım Tasarımı . . . . .	13
3.1.3 Web Arayüzü ve Girdi-Çıktı Tasarımı . . . . .	14
3.2 Veritabanı Tasarımı . . . . .	14
<b>4 Uygulama</b>	<b>16</b>
<b>5 DeneySEL Sonuçlar</b>	<b>17</b>
<b>6 Sonuç</b>	<b>18</b>

<b>EKLER</b>	<b>19</b>
<b>Referanslar</b>	<b>19</b>
<b>Özgeçmiş</b>	<b>19</b>

## SİMGE LİSTESİ

---

C	Celsius Santigrat Derece
Omega	Ohm
K	Kilo
mA	Mili amper
s	Saniye
V	Voltaj
A	Amper

## KISALTMA LİSTESİ

---

DC	Direct Current
AC	Alternate Current
GPIO	General Purpose Input Output
LED	Light Emitting Diode
MISO	Master In Slave Out
DFRAC	Demura's Fall Risk Assessment Chart
EMG	Electromyography
RPi	Raspberry Pi (3 Model B)
FPRI	Serial Peripheral Interface Bus
SPI	Fall Probability
Wi-Fi	Wireless Fidelity
DCAS	Data Center Automation System
SQL	SQLite Database
RF	Radio Frequency



## ŞEKİL LİSTESİ

---

Şekil 1.1	Raspberry Pi 3 . . . . .	3
Şekil 1.2	DHT11 Sıcaklık ve nem sensörü . . . . .	4
Şekil 1.3	BMP180 Basınç Sensörü . . . . .	5
Şekil 1.4	Arduino Uno . . . . .	6
Şekil 1.5	NRF24L01 . . . . .	6
Şekil 2.6	Arduino ne NRF bağlantı şeması . . . . .	10
Şekil 2.7	Arduino ne DHT11 bağlantı şeması . . . . .	11
Şekil 2.8	Arduino ne DHT11 bağlantı şeması . . . . .	11
Şekil 3.9	Genel Veri Akış Diyagramı . . . . .	14
Şekil 3.10	Veritabanı Şeması . . . . .	15

## TABLO LİSTESİ

---

Tablo 1.1	Sistemlerin Karşılaştırılması . . . . .	2
Tablo 1.2	My caption . . . . .	4
Tablo 1.3	Yazılım araçları . . . . .	5
Tablo 1.4	Görev-Zaman Diyagramı . . . . .	7
Tablo 1.5	Maliyet . . . . .	7
Tablo 1.6	Yazılım Maliyet Analizi . . . . .	8
Tablo 2.7	NRF24L01 Pinleri . . . . .	9

# VERİ MERKEZİ OTOMASYON SİSTEMİ

Sinan Can Keskin

Bilgisayar Mühendisliği Bölümü  
Bilgisayar Projesi

Danışman: Doç. Dr. Ali Gökhan YAVUZ

Veri Otomasyon Sistemi Projesi’nde amaçlanan, günümüzde çok önemli hale gelen veri merkezlerinin güvenliğini arttırmak ve çalışabilirliğini üst seviyeye çıkarmaktır. Günümüz teknolojisinde otomasyon sistemleri giderek önemlerini arttırmaktadırlar. Gerek elektrik kontrolünde gerek akıllı sistemler geliştirmelerinde bize büyük faydalar ve kolaylıklar sağlamaktadırlar. Buradan yola çıkarak veri merkezlerine de otomasyon sistemleri entegre edilerek her türlü sorunun önüne geçilmeye, oluşan sorunlara en hızlı şekilde müdahale imkanı sağlanmaya çalışılmıştır.

Sistemin genel çalışmasını tarif etmek gerekirse, veri merkezindeki her türlü çevresel faktörlere duyarlı bir sistem olması düşünülmektedir. Kabinetlerden sensorler ile alınacak sıcaklık, nem, hava akımı bilgileri, Arduino modüllerine, oradan da kablo-suz haberleşme ile master device olarak düşünülen RPi’a aktarılacaktır. RPi’a alınan veriler yerel veri tabanında depolanacak, buradan işlenerek bulut veri tabanına aktarılacaktır. Bulut veri tabanı aynı zamanda bir web ara yüzü ile bağlantılı olup, bu web arayüzünde belirli aralıklarla yenilenen veriler görüntülenebilecek, oluşan olaylar bu web ara yüzünden gözlenebilecektir.

Aynı zamanda kabinetler haricinde odanın içerisindeki sıcaklık bilgisi alınacaktır. Duman ve su seviyesi sensorleri ile oda içerisindeki yangın ve su baskınları durumlarına karşı uyarı sistemleri geliştirilmesi amaçlanmıştır.

**Anahtar Kelimeler:** Veri merkezi, otomasyon, Raspberry Pi, Arduino, Nesnelerin haberleşmesi

---

**YILDIZ TEKNİK ÜNİVERSİTESİ**  
**ELEKTRİK - ELEKTRONİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

## ABSTRACT

---

# DATA CENTER AUTOMATION SYSTEM

Sinan Can Keskin

Department of Computer Engineering  
Computer Project

Advisor: Ass. Prof. Dr. Ali Gökhan YAVUZ

The main purpose of developing a Data Center Automation System is increasing the security and operating capability of the data centers. With today's technology, its becoming more and more important to save our data and with this necessity, automation projects are also becoming more important in every field.

Shortly, the system is going to have a hierarchical structure among a master device RPi and multiple slave (Arduino) devices that connected to the racks, which are responsible for reading and transferring data over a wireless communication protocol. We want to read temperature, humidity, and air flow information from the racks, which are very important for us. The data will be read through sensors and transmitted to the master device. What RPi is going to do is collecting and storing the data in a local database (SQL), processing the data and sending to the cloud database which will be connected with a web interface to monitor the events and changes that are happening in the racks and inside of the data center.

Besides, we want to measure the temperature of the room and trying to implement an alarm system for fire and flooding inside of the data center with using smoke and water level sensors.

**Keywords:** Data center, Automation, Raspberry Pi, Arduino, Internet of things

---

**YILDIZ TECHNICAL UNIVERSITY**  
**FACULTY OF ELECTRICAL AND ELECTRONICS**  
**DEPARTMENT OF COMPUTER ENGINEERING**

Bu çalışma veri merkezleri için RPi ve Arduino tabanlı, sıcaklık, nem, hava akımı, yangın, su baskını durumlarını ölçecek, ölçülen değerleri izlememize imkan sağlayacak, çok fonksiyonlu bir kontrol sistemi gerçeklemek amacıyla. Kabinetlerin içersine, üzerinde sensorler, devre elemanları, arduino, haberleşme modülü ve arduinonun besleneceği 9V pili içeren boardlar yerleştirilecektir. Sistem kablosuz haberleşecektir, aynı zamanda güç kaynağı da arduinoların ve sensörlerin düşük güç tüketimi düşünülerek ve kabinetlerin içersindeki kablo kalabalığını önlemek amacıyla piller olacaktır. Arduinolar ile bu pillerin güç seviyesi ölçülmeye çalışılacak ve piller zayıfladığı anda uyarı veren bir sistem de oluşturulmaya çalışılacaktır. Kabinetlerden alınan veriler, kablosuz haberleşme ile RPi'ya iletilecek ve burada yerel veritabanında saklanacaktır. Sistemi kullanacak olan sorumlular, tanımlanan kullanıcı adları ile bir web ara yüzüne diledikleri yerden giriş yapıp sistemin o anki durumunu izleyebileceklerdir.

### 1.1 Ön İnceleme

Bu kısımda daha önce yapılmış ve kullanıma hazır projeleri inceleyip kendi sistemiz ile karşılaştırdık. Ayrıca bu araştırma sırasında incelediğimiz sistemlerin artılarını ve eksilerini bulup, iyi yanlarından kendi projemizde yararlanmak istedik, eksik taraflarından ise kaçınmaya çalıştık. Öncelikle WebCTRL firmasının dünyanın en büyük veri merkezi olan Verizon binasına kurduğu otomasyon sistemini inceledik. Genel olarak bina otomasyonları için özelleştirilmiş toollar kullanan şirket, rüzgar, sıcaklık ve diğer ölçüm gerektiren değişkenlerin hemen hepsini kullanmaktadır. Elbette profesyonel bir ekipleri oldukları için bizden çok daha fazla ölçüm yapabilmekteler. Ve ayrıca kendi yazılımlarını yaparak sistemin kontrolünü kolaylaştırmaktalar. Diğer bir ilgimizi çeken firma ise çok daha bilindik bir firma olan Schneider Elektrik. Otomasyon sektöründe en kapsamlı şirketlerden birisi olan Schneider Elektrik, yine oldukça kapsamlı bina ve veri merkezi otomasyon projelerine imza atmış bulunmakta. Aşağıdaki Tabloda kendi sistemimiz ve yukarda bahsettiğim iki sistem üzerinde bir karşılaştırma Tablosu verilmektedir.

**Tablo 1.1** Sistemlerin Karşılaştırılması

Sensors/Devices	WebCTRL	Schneider Electric	DCAS
Işık	+	+	-
CO2	+	+	+
Sıcaklık	+	+	+
Hava Akışı	-	+	+
Nem	+	+	+
Kendini Yenileme	-	+	-
Wi-Fi	-	-	+

## 1.2 Fizibilite

Fizibilite çalışmaları, bu çalışmadaki en önemli adımlardan biridir, özellikle sistem analizi kısmı bizim için büyük önem arz etmektedir. İyi bir fizibilite çalışması başarı şansımızı kesinlikle arttıracaktır. Fizibilite çalışmalarını yaparken sistemimizi gerçeklerken bize gereken teknik detayları araştıracağız ve raporlayacağız. Ayrıca fizibilite çalışmalarımız bizim için yine büyük öneme sahip olan ekonomik maliyetleri de aşağı yukarı hesaplamamızda yardımcı olacaktır.

### 1.2.1 Teknik Fizibilite

Proje için gerekli yazılım ve donanım komponentleri aşağıda gösterilmiştir.

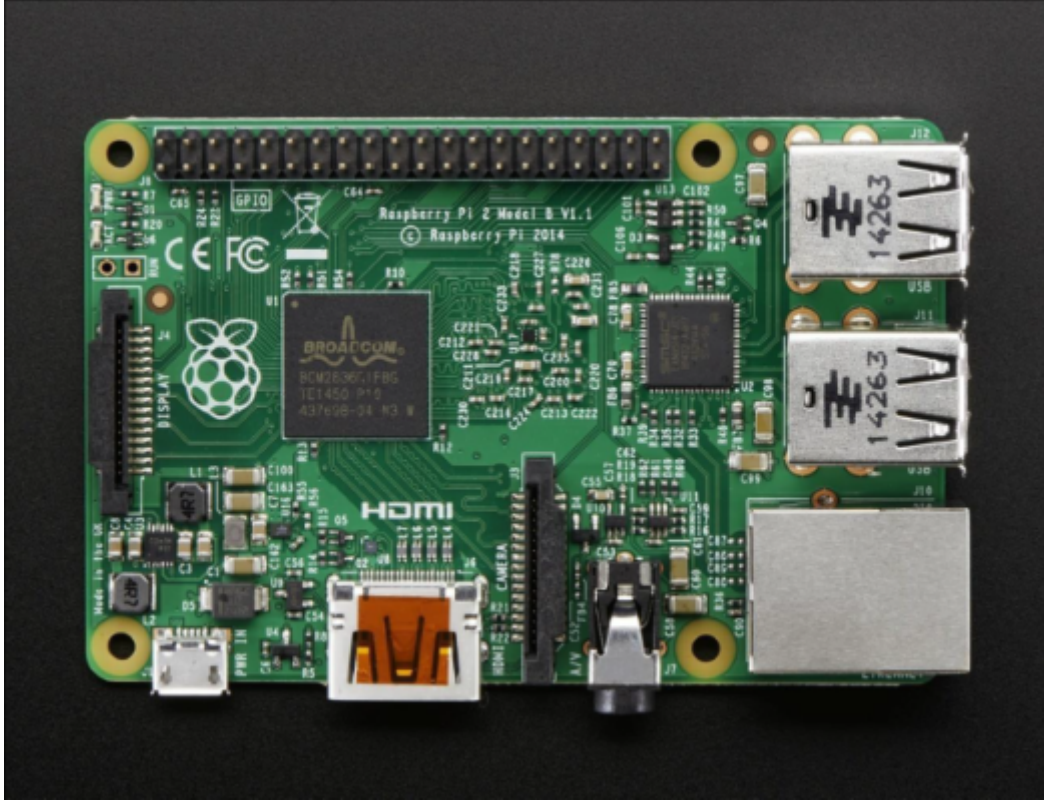
#### 1.2.1.1 Donanımsal Teknik Analiz

Proje online bir yapıya sahip olacaktır. Bu sayede bir web sunucusu üzerinden uzaktan kontrol şansına sahip olacağız. Bunu gerçekleştirebilmek için kabinetlere koyacağımız Arduino modüllerinin NRF24L01 kablosuz haberleşme modülleri ile RPi ile haberleşmelerini sağlayacağız. NRF24L01 hem fonksiyonları, hem de ekonomik olması hasebiyle projemiz için çok uygun bir modül. Araştırmamızı yaparken aslında Zigbee modüllerini kullanmak istemiştik ancak bu modüllerin oldukça pahalı olması bizi alternatif arayışlara sürükledi ve son olarak NRF24L01 da karar kıldık. Sıcaklık sensörü olarak DHT11 sıcaklık ve nem sensorunu kullanacağız. Hava akımının ölçümünü basınç farklılıkları üzerinden yapacağımız için BMP180 dijital basınç sensorunu kullanacağız. Arduinoları beslemek için 9V GP ultra piller kullanacağız. Bu piller şarj edilebilir olabilir. MQ-7 CO sensörü ile kabinetlerde ve oda içerisinde herhangi bir duman-ateş oluşma durumunu ölçmeye çalışacağız ve su detektörü ile de veri merkezindeki su baskınlarından haberdar olmaya çalışacağız.



### 1.2.1.2 Raspberry Pi

Raspberry Pi kredi kartı ölçülerinde tek boardlı İngiliz menşeli bir bilgisayardır. Fiyatı modeline göre değişiklik gösterir. Bizim elimizde en son çıkan RPi 3 Model B bulunmaktadır. Yaklaşık 35usd fiyatı olan RPi 3 Mode B oldukça yetenekli bir bilgisayardır. Üzerinde 1.2 GHz 64-bit quad-core Arm-v8 işlemci ve 1024 Mb ram bulunmaktadır. 802.11n kablosuz haberleşme protokollerini desteklemektedir. Ayrıca Bluetooth 4.1 de yeni versiyon ile gelen özelliklerdendir. 4 adet usb portu bulunan RPi 40 adet GPIO pine sahiptir. Full hdmi desteğinin yanı sıra Ethernet portu, 3.5 mm ses jak çıkışı, kamera ve ekran arayüzü bulundurmaktadır. VideoCore IV 3d grafik kartı da RPi 3 üzerindeki donanımlardan biridir. Ek bellek olarak da microSD hafıza kartı desteği kullanan RPi 3, adeta bir Tazmanya canavarıdır. Üretici firma, RPi üzerinde çalışabilmesi için Linux Arm dağıtımları sağlamıştır. Son zamanlarda Windows IOT de oldukça yaygın olarak kullanılmaya başlanmıştır. Programlama dili olarak Python, en yaygın kullanılan dildir.



Şekil 1.1 Raspberry Pi 3

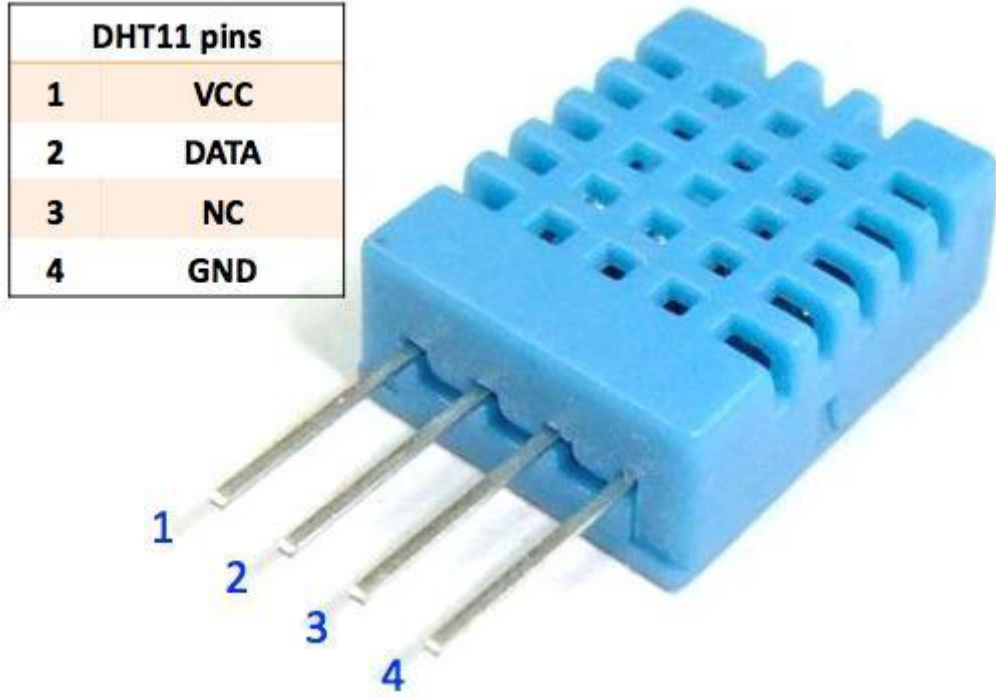
### 1.2.1.3 DHT11 Sıcaklık ve Nem Sensörü

DHT11 dijital sıcaklık ve nem sensörü üzerinde kalibre edilmiş şekilde, dijital olarak sıcaklık ve nem bilgisini çıkış verebilen bileşik bir sensordur. Modül oldukça güvenilir ve stabil değerler sağlamaktadır. Fiyatı yaklaşık olarak 6usd civarındadır ve oldukça geniş

**Tablo 1.2** My caption

İtem	Aralık	Nem	Sıcaklık	Çözünürlük	Pin
DHT11	20-90 Nem 0-50C	+/- 5	+/- 2C	1	4 ping tek sıra

bir kütüphaneye sahiptir. Sensor, 8 bitlik bir mikro denetleyici ile kontrol edilmektedir.



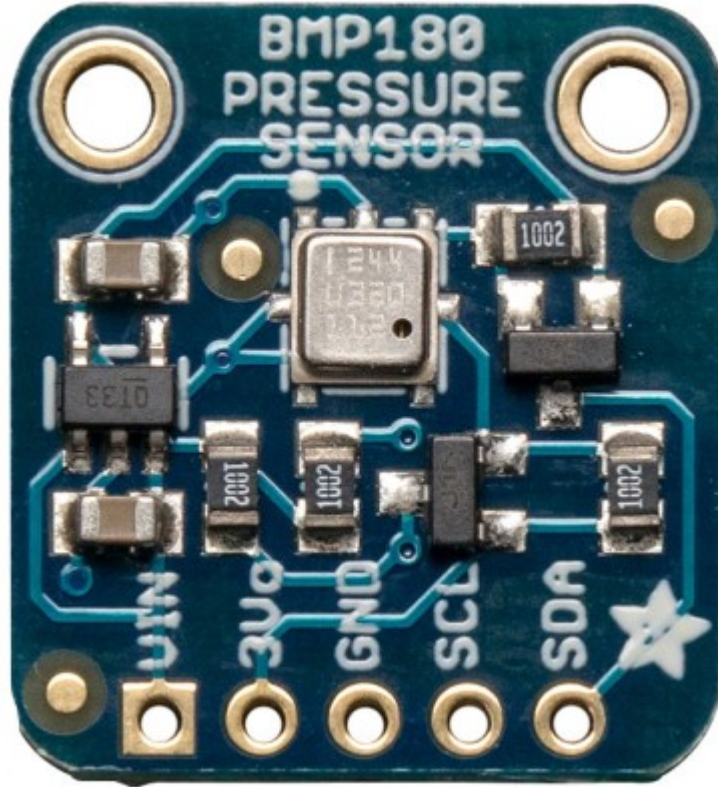
**Şekil 1.2** DHT11 Sıcaklık ve nem sensörü

#### 1.2.1.4 BMP180 Basınç Sensörü

Kabinetlerin içerisindeki hava akımını ölçmek için basınç farkından yararlanmak istiyoruz. Hava akımı olan yerde basınç düşer, biz de basıncın düşme miktarını ölçebilirsek hava akımını da ölçmüş oluruz. Bunun için kullanacağımız cihaz Adafruit BMP180 dijital basınç sensörü. Oldukça hassas ölçüm kabiliyetine sahip olan BMP180 ekonomik fiyatıyla projemiz için çok uygun.

#### 1.2.1.5 Yazılım Fizibilitesi

Sensor ve haberleşme komutları Arduino IDE üzerinden yazılacak ve Arduino üzerinde derlenecek. Diğer taraftan RPi üzerinde kodlanacak olan web ara yüzü kodları HTML-PHP-CSS dillerinde, Zend 1 Framework üzerinde yazılacak. Sistem bütün tarayıcılar



**Şekil 1.3** BMP180 Basınç Sensörü

tarafından desteklenebilir olacak; mobil cihazlar, tabletler veya PC'ler.

**Tablo 1.3** Yazılım araçları

Yazılım	İsim
OS	Raspbian Os
Platform	Phyton, PHP, SQLite, Arduino
Web Server	Apache
Web Framework	Zend 1

#### 1.2.1.6 Arduino

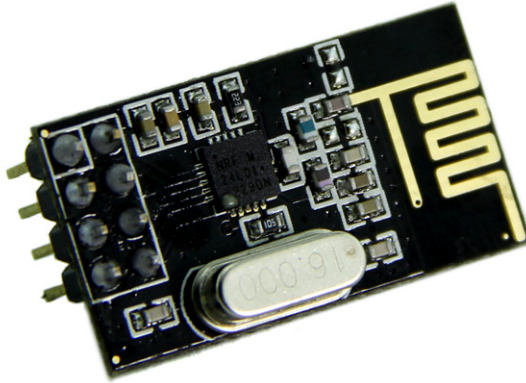
Arduino bir I/O kartı ve Processing/Wiring dilinin bir uygulamasını içeren geliştirme ortamından oluşan bir fiziksel programlama kartıdır. Arduino, tek başına çalışan interaktif nesneler geliştirmek için kullanılabileceği gibi bilgisayar üzerinde çalışan yazılımlara da bağlanabilir. Arduino kartları bir Atmel AVR mikrodenetleyici ve programlama ve diğer devrelere bağlanabilmek için gerekli yan elemanlardan oluşur. Her kartta en az 5V'lık bir regüle entegresi ve bir 16MHz kristal osilator bulunur. Arduino IDE kod editörü, derleyici olarak görev yapan, aynı zamanda derlenen programı karta yükleyebilen, her platformda çalışabilen Java programlama dilinde yazılmış bir uygulamadır. Arduino tamamen açık kaynak kodludur.



Şekil 1.4 Arduino Uno

#### 1.2.1.7 NRF24L01

NRF24L01, uygun fiyatı, düşük güç tüketimi ve 2.4 GHz çalışma frekansına sahip bir RF haberleşme modülüdür. Anten kullanılmadan sağladığı 250 metre menzil, anten kullanılmasıyla birlikte 1000 metreye kadar çıkabilmektedir. Modül, half-duplex veya full-duplex haberleşme için kullanılabilir. Modül SPI protokolü kullanarak haberleşmektedir, arduino ve RPi üzerinde bulunan SPI pinlerini haberleşme için kullanacağız.



Şekil 1.5 NRF24L01

#### 1.2.2 Zaman Fizibilitesi

Aşağıdaki tabloda görev isimleri ve görevlerin yapılacağı zaman aralıkları gösterilmektedir.

Görev Listesi: 1-Veri merkezi ön fizibilite çalışması 2-Sistem tasarımı 3-Siparişlerin verilmesi 4-Modüllerin oluşturulması 5-Modüllerin birleştirilmesi 6-Test

**Tablo 1.4** Görev-Zaman Diyagramı

Görevler	Ekim	Ekim	Kasım	Kasım	Aralık	Ocak
1						
2		+	+			
3			+			
4				+	+	
5					+	
6					+	+

**1.2.3 Yasal Fizibilite**

Bu proje Yıldız Teknik Üniversitesi, Elektrik-Elektronik Fakültesi, Bilgisayar Mühendisliği Bölümü tarafından gerçekleştirilmiştir. Tüm haklar Fakülte adına saklıdır, izinsiz kopyalanamaz ve kullanılamaz.

**1.2.4 Ekonomik Fizibilite**

Aşağıda elimizde olan parçalar ve ilk etapta sipariş verdiğimiz parçaların ayrıntılı bir listesi yapılmıştır.

**1.2.4.1 Donanımsal Ekonomik Fizibilite**

Projenin ilk aşamasındaki maliyetler tabloda gösterilmiştir.

**Tablo 1.5** Maliyet

Cihaz	Özellik	Fiyat
Raspberry Pi	3 Model b	190 tl
Arduino Uno	Clone(x2)	40 tl
Breadboard	7 inc(x2)	20 tl
Dirençler	330, 470, 1k	5 tl
DHT11 sensor 65 tl	Nem	Sıcaklık Ölçer(x5)
Jumper Kablolar	M-M, M-F	5 tl
BMP180	Basınç Sensörü(x4)	52 tl
NRF24L01	RF Modülü(x4)	21 tl
Toplam	1	380 tl

**1.2.4.2 Yazılımsal Maliyet Analizi**

Kullanacağımız yazılımlar ve maliyetleri aşağıda gösterilmiştir.

**Tablo 1.6** Yazılım Maliyet Analizi

Araç	Program	Fiyat
OS	Raspbian	0
Geliştirme Ortamı	Sublime	0
Web Platform	WAMP	0
IDE	Phyton	0
Toplam		0

Bu proje, donanım ve yazılım bilgisini birleştirip, insanların çevresini daha akıllı hale getirmeyi amaçlamaktadır. Diğer taraftan donanıma karşı olan ilgim ve çalıştığım iş beni bu projeyi yapmak için oldukça motive etmiştir. Umuyorum ki elektronik ve merakımı birleştirmek iyi sonuçlar doğuracaktır. DCAS donanımı, yazılımı ve haberleşmeyi bir arada içeren oldukça karmaşık bir proje. Böyle bir projeyi gerçekleştirebilmek için parçaları iyi tanımak ve iyi birleştirmek gerekli. Gerekli devreleri, donanımsal yapıları tasarlamak ve gerçekleştirmek en önemli parça. Diğer taraftan sistemin kullanıcı ara yüzü de iyi tasarlanmış ve kullanıcı dostu olmalı. Aksi takdirde sistemden alınabilecek verim düşebilir. Verinin donanımdan alınıp aktarılması ve işlenmesi ise projedeki en zor kısım olarak düşünülebilir. Aşağıdaki bölümde donanımsal parçaların fiziksel bağlantıları açıklanmaya çalışılmıştır.

## 2.1 Elektronik Devre Tasarımı

Devrelerin tasarımını oldukça basit tutmaya çalıştık. Tasarım yapılırken board, Arduino ve sensorler arasındaki bağlantıları jumperler aracılığıyla yaptık. Öncelikle haberleşme modülünün Arduino'ya eklenişini aşağıdaki gibi yaptık.

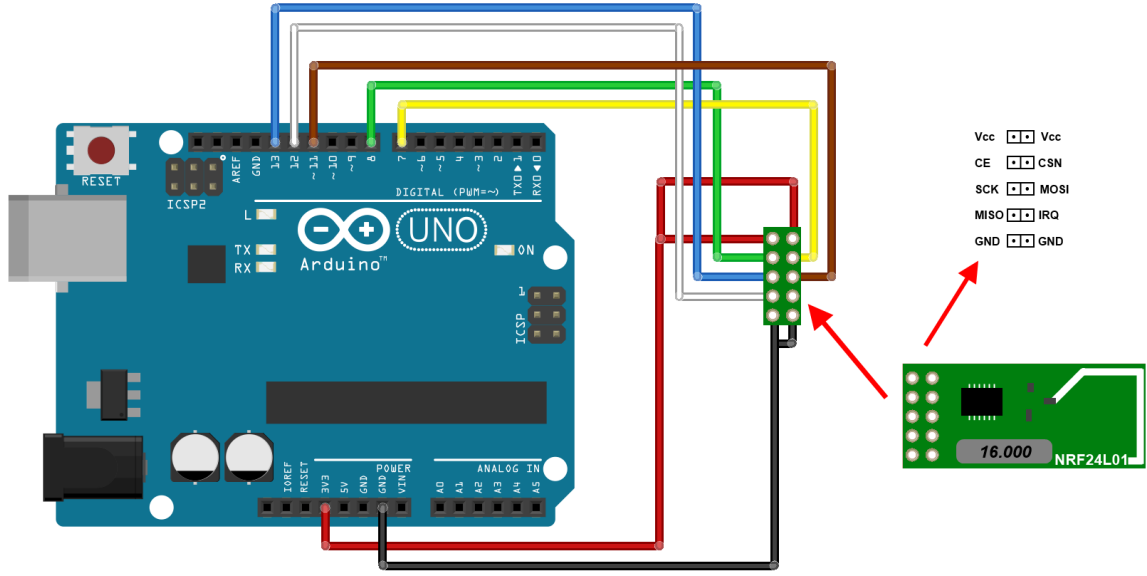
### 2.1.1 Arduino Uno ve NRF24L01 Devre Şeması

Devre şeması aşağıda gösterilmiştir.

**Tablo 2.7** NRF24L01 Pinleri

Araç	Program	Fiyat
OS	Raspbian	0
Geliştirme Ortamı	Sublime	0
Web Platform	WAMP	0
IDE	Phyton	0
Toplam		0





**Şekil 2.6** Arduino ne NRF bağlantı şeması

Modül SPI kullanarak haberleşmektedir. VCC(+, 3.3V) ve GND(-) uçlarından elektrik beslemesi yapılmaktadır. Üzerinde CS, CE, MOSI, MISO, SCK pinleri bulunmaktadır. Bu pinlerin Arduino üzerinde nereye bağlanacaklarına ilişkin tablo yukarıdadır.

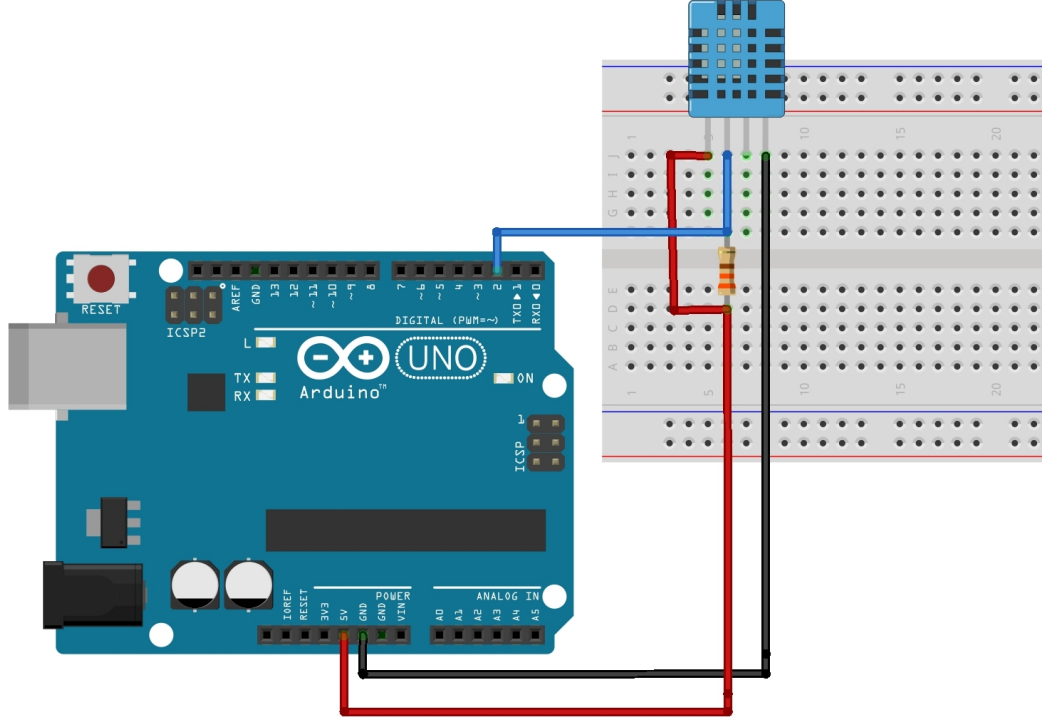
### 2.1.2 Arduino Uno ve DHT11 Nem ve Sıcaklık Sensörü

Bağlantı şeması görseldeki gibidir. Üzerinde 3 adet pin bulunan DHT11, çok basit bir devre yapısına sahiptir. VCC(3.3V+), GND(-) ve bir adet output pini içeren cihaz, sıcaklık ve nem bilgisini dijital olarak okumamıza olanak sağlar. DHT11, 0-50 derece arası sıcaklıkları yüzde 2 sapmayla ve yüzde 20-90 arası nemi yüzde5 sapmayla ölçebilir.

### 2.1.3 Arduino ve BMP180

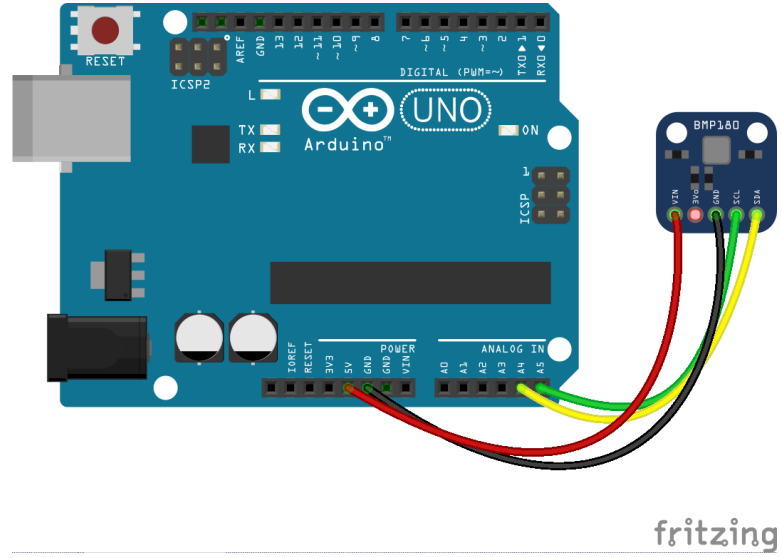
BMP180 basınç sensorunu bu projede kullanmamızdaki amaç, kabinler içerisindeki hava akımının istediğimiz şekilde direk olarak ölçecek hiçbir uygun sensor bulamamamızdır. Hava akımının olduğu yerde hava basıncının düşeceği prensibinden yola çıkarak, kabinlere yerleştireceğimizi BMP180ler ile dışarıdaki açık hava basıncını karşılaştırarak, çeşitli matematik hesaplamalar ile hava akımını ölçmek istiyoruz. Dijital değerler üreten bu sensor, 500-900 metre arası yükseklikleri de ölçebilmekte. BMP180, bağlantı olarak DHT11 gibi basit bir yapıya sahip. 3.3V VCC, GND, sensorun üzerinde bulunan 5 pinden ikisi. DA(SDA) pini arduino üzerindeki herhangi bir LDA pinine, CL(SCL) pini ise yine herhangi bir SCL pinine bağlanılarak bağlantılar tamamlanır. BMP180 üzerindeki 5. pin ise I/O cihazımız üzerindeki voltajın 1.8V





Şekil 2.7 Arduino ne DHT11 bağlantı şeması

altına düşmesi durumunda sensörü otomatik olarak devre dışı bırakmaktadır.



Şekil 2.8 Arduino ne DHT11 bağlantı şeması

# 3

## Sistem Tasarımı

---

Sitem genel olarak iki ana parçadan oluşmaktadır, bunlardan bir tanesi yukarıda ayrıntılı şekilde anlatılan elektronik tasarım ve devrelerin gerçekleştirilmesidir. Diğer ana parça ise sensorlerden veri alınması, alınan verilerin arduinodan RPi'a aktarılması, burada verilerin depolanması ve web ara yüzünden izlenilmeye uygun olacak şekilde işlenmesidir. Sistemden normal değerlerin dışında değerler okunması durumunda-örneğin kabinet içerisinde sıcaklığın anormal derecede artması- olay kayıtları oluşturulacaktır. Sisteme gelen veriler sürekli olarak izlenebilecektir ancak sürekli bir veri kaydı mümkün değildir. RPi'nın kısıtlı depolama alanı ve çok sürekli işlemlerde ısınma problemlerinin olması, sürekli veri kaydının imkansız olmasına neden olmaktadır.

Tasarladığımız sistemlerde farklı arduino modüllerinden iki saniye aralıklarla veri alınmaktadır. Projenin şu anki aşamasında iki adet modül gerçekleştirilmiştir. Bu iki modülden veriler senkron şekilde alınabilmektedir.

Verinin karışmadan alınabilmesi için arduino modüllerine etiketler konulmuştur. RPi, gelen verileri etiketlerine göre ayırarak veri tabanına yazmaktadır.

Kullandığımız veri tabanı gayet basit bir yapıya sahiptir. Veri tabanı tasarımıyla ayrıntılı olarak bahsedilecektir.

Veri tabanından verilerin alınabilmesi için yazdığımız web sayfası, RPi üzerinde verileri depoladığımız MySql'e bağlanmaktadır ve web arayüzüne verileri farklı modülleri temsil edecek şekilde gösterebilmektedir.

Web arayüzümüz üzerinden kabinetlerdeki durum anlık olarak izlenebilmektedir.

## 3.1 Yazılım Tasarımı

### 3.1.1 Arduino Yazılım Tasarımı

Arduino üzerinde programlama yapmamıza iman veren Arduino Playground, prosedural bir programlama dili olup, C diline çok benzemektedir. Herhangi bir nesneye dayalı tasarıma veya MVC gibi yapıların kullanılmasına gerek yoktur. Sensorler, modüller, haberleşme için kullanılacak olan kodlar, parçalar halinde yazılacak ve daha testler parçalar halinde yapılacaktır. Son aşamada ise ayrı ayrı yazılan kod parçaları birleştirilerek sistemin bütünlüğü sağlanacaktır.

Verileri RPi'ye gönderen kod parçası aşağıdaki gibidir.

```
void loop() delay(2000); float tmp; values[0]=1; values[1] = dht.readHumidity();  
values[2] = dht.readTemperature(); values[3] = bmp.readPressure();  
  
// Check if any reads failed and exit early (to try again). if (isnan(values[0]) ||  
isnan(values[1]) || isnan(values[2]) || isnan(values[3])) Serial.println("Failed to  
read from DHT sensor!"); return;
```

Veriler integer olarak sensörlerden okunup, dizi halinde iletilmektedir. Dizinin ilk değeri verinin hangi modülden geldiğini anlamamızı sağlar.

Diğer değerler sırasıyla nem, sıcaklık, basınç değerleridir. Sistem istenildiği takdirde her türlü verinin eklenmesine kolayca imkan sağlamaktadır.

Kullandığımız kütüphanelerden bahsetmek gerekirse, her sensör için farklı kütüphane kullanılmaktadır.

### 3.1.2 RPi Yazılım Tasarımı

RPi üzerinde çok fazla bir kod yükü bulunmamasına rağmen, haberleşme modülünün programlanması ve olay kayıtlarının alınıp veri tabanına kaydedilmesi işlemleri, Python ile yapılmıştır.

Arduino Playground'da olduğu gibi, Python üzerinde de herhangi bir nesneye dayalı programlamaya ihtiyaç yoktur. Prosedural olarak programlama yapılacaktır. Olay kayıtları SQLite veri tabanında tutulacaktır. İhtiyaç olduğunda web ara yüzü üzerinden olay kayıtlarına bakılabilecektir.

### 3.1.3 Web Arayüzü ve Girdi-Çıktı Tasarımı

Web arayüzü ASP.net üzerinden yazılacaktır. Tasarlanan site modüllerimizi tek tek gösterecektir. RPi üzerindeki veri tabanına bağlanıp verilere erişecektir.

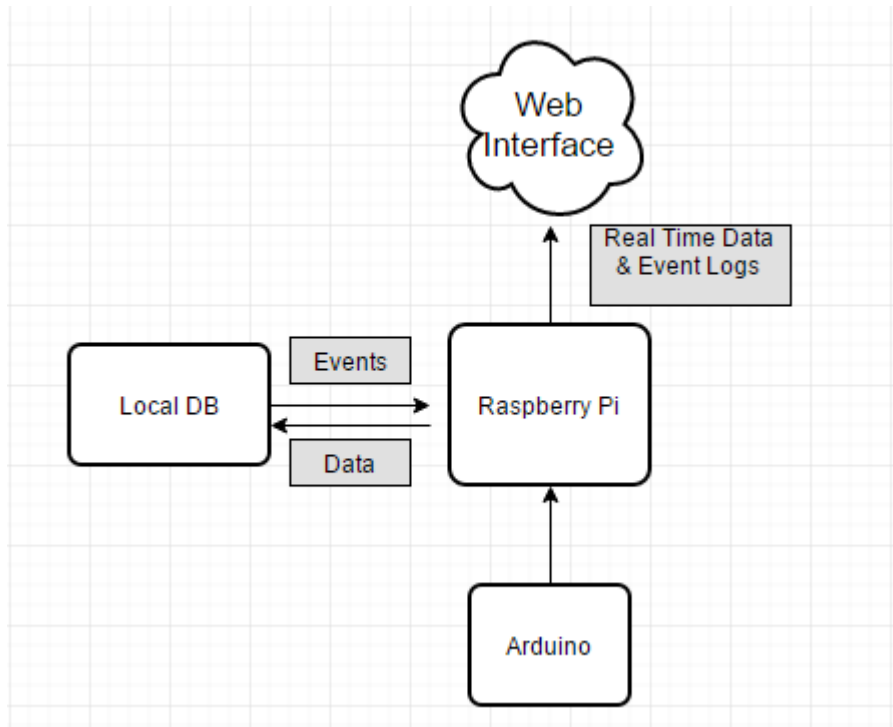
Hem olay kayıtlarını gösterebilecek hem de canlı durumu izleyebilecektir.

Verilerin RPi'dan çıkarılabilmesi için APACHE server kullanılacaktır. APACHE'nin seçilmesindenki neden MySQL ile birlikte kolay kullanılabilmesi ve daha önceden bildiğimiz bir sever olmasındandır.

## 3.2 Veritabanı Tasarımı

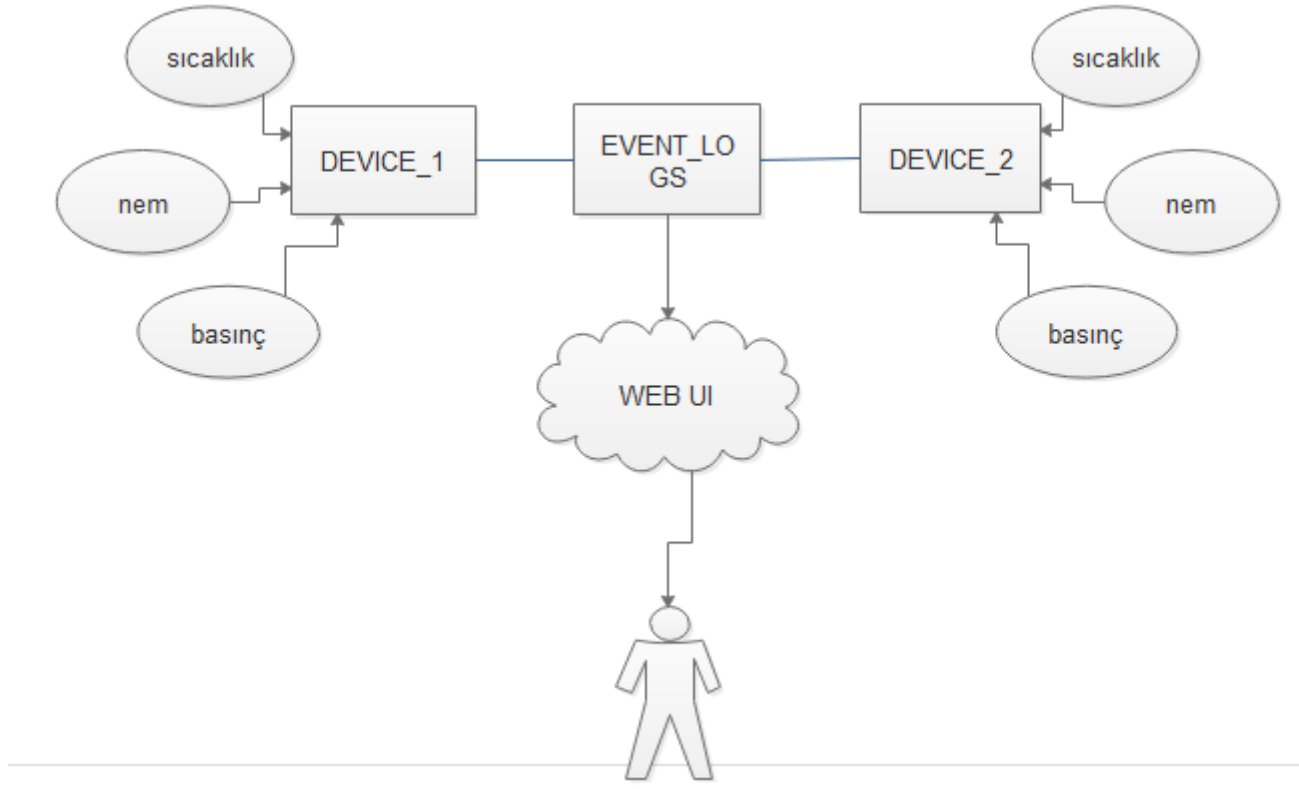
Veritabanımız, ihtiyaçlarımızı doğrultusunda oldukça basit tutulmuştur. Her modül için verilerin ayrı ayrı tutulduğu bir tablo, ve verilerden elde edilen olay kayıtlarının (sıradışı durumların) tutulduğu bir tablodan ibarettir.

Sistemin genel veri akış tasarımı aşağıdaki gibi olacaktır.



Şekil 3.9 Genel Veri Akış Diyagramı

Veri tabanımızın şeması aşağıdaki gibidir.



Şekil 3.10 Veritabanı Şeması

## 4 Uygulama

---

Projeyi ana başlıklar altında özetlemek gerekirse, arduinolardan verilerin alınması, arduino ve RPi arasında haberleşmenin kurulması, verilerin RPi tarafından işlenmesi ve web arayüzünden görüntülenmesi şeklinde yapılabilir. Geldiğimiz noktada bütün özellikler eksiksiz çalışmaktadır.

## 5 Deneysel Sonuçlar

---

Deneysel sonuçlar daha sonra eklenecektir.

## 6 Sonuç

---

Gelinen nokta itibariyle yüzde doksan ölçüde tamamlanmış bulunmaktadır.

Yapılan testlerde hava basıncı sensörleri kullanılarak kabinetlerdeki hava akımının ölçülmesinin mümkün olduğu gözlenmiştir. Sensörlerin bize sağladığı değerler yeteri kadar hassastır. Ancak doğru sonuçların elde edilebilmesi için veri merkezinde çeşitli ölçümler yapılmalı ve kod içersinde optimizasyonlara gidilmelidir.

İhtiyaca göre her türlü sensör projeye eklenebilmektedir.



### 1. ÜYENİN KİŞİSEL BİLGİLERİ

**İsim-Soyisim:** Sinan Can Keskin  
**Doğum Tarihi ve Yeri:** 01.09.1993, İskilip  
**E-mail:** keskinsinancan@gmailcom  
**Telefon:** 0553 627 30 14  
**Staj Tecrübeleri:** Kesir Mühendislik

### Proje Sistem Bilgileri

**Sistem ve Yazılım:** Windows İşletim Sistemi, Python, Arduino, PHP, SQLite, Raspbian İşletim Sistemi  
**Gerekli RAM:** 1GB  
**Gerekli Disk:** 256MB