



Bütünleşik Ev Sağlık İzleme Sistemi: Raspberry Pi Tabanlı Bir Yaklaşım

Yazılım Mühendisliği Ana Bilim Dalı

Dönem Projesi

Mehmet Hakan Selek

ORCID 0009-0006-5623-318X

Proje Danışmanı: Prof. Dr. Doğan AYDIN

Ocak 2024

Bütünleşik Ev Sağlık İzleme Sistemi: Raspberry Pi Tabanlı Bir Yaklaşım

ÖZ

Bu proje raporu, Raspberry Pi tabanlı bir evde sağlık takip sistemi geliştirmeyi detaylı bir şekilde ele almaktadır. Sistem, hastaların temel sağlık verilerini sürekli olarak izleyerek, anormal durumları hızla tespit edebilir. Hareketlilik seviyeleri ve yatak içi pozisyonları takip edilerek yatak yaralarının önlenmesine yardımcı olurken, ev otomasyonu entegrasyonu ile çevresel koşullar kontrol altına alınmaktadır. Bu teknolojik yenilikler, hasta bakımını kolaylaştırmakta ve hastaların yaşam kalitesini artırmaktadır. Sağlık hizmetlerinin verimliliği ve erişilebilirliği artırılarak genel sağlık hizmetleri alanında bir kolaylık sunulmaktadır.

Anahtar Sözcükler: Evde Sağlık Takibi, Uzaktan Sağlık İzleme Teknolojisi, Hasta Durum İzleme, Sağlık Veri Analitiği, Akıllı Sağlık Cihazları

Integrated Home Health Monitoring System: A Raspberry Pi-Based Approach

Abstract

This project report thoroughly discusses the development of a Raspberry Pi-based home health monitoring system. The system continuously tracks patients' basic health metrics to quickly detect abnormal conditions. It aids in the prevention of bedsores by monitoring mobility levels and in-bed positions, while environmental conditions are regulated through home automation integration. These technological innovations simplify patient care and enhance the quality of life for patients. They also offer convenience in the general health services sector by improving efficiency and accessibility.

Keywords: Home Health Monitoring, Remote Health Monitoring Technology, Patient Status Monitoring, Health Data Analytics, Smart Health Devices

*Her adımda yanımda olan, rüyalarımın ve hedeflerimin en büyük destekçisi, nişanlım;
bu yolculuğun her kısmında bana ilham veren ve her zorluğu aşmam için güç katan
Ayça ÖZARICILAR'a bu projeyi sevgiyle ithaf ediyorum.*

Teşekkür

Proje danışmanım Dođan AYDIN'a, bu alıřmayı řekillendirmemde sađladıđı kıymetli ynlendirmeler ve bilimsel katkıları iin derin minnettarlıđımı sunarım.

İçindekiler

Öz	i
Abstract	ii
Teşekkür	iv
Şekiller Listesi	viii
Kısaltmalar Listesi	ix
Semboller Listesi	x
1 Giriş	1
2 Tasarım Ve Cihaz.....	2
2.1 Kontrol Ünitesi	3
2.2 16 Bit Analog-Dijital Çevirici	3
2.2.1 Sürekli Mod	4
2.2.2 Tek Atış Modu	4
2.3 DS18B20 Sıcaklık Sensörü	5
2.4 SHT 11 sıcaklık ve nem sensörü.....	6
2.5 Raspberry Pi'nin hazırlanması.....	7
2.5.1 Apache sunucusu, PHP, MariaDB, phpMyAdmin ve web sayfası kurulumu.....	9
2.6 Python Scriptleri	13
2.6.1 ADS1115 ADC	13
2.6.2 DS18B20 ve DS18S20 sıcaklık sensörleri	15
2.6.3 SHT11 sıcaklık ve nem sensörü.....	17
2.7 Veritabanı	17
2.8 Web sayfası ve içeriği	19
2.8.1 Sisteme tetikleyici pinler atama	20
2.8.2 Cihaz-Pin Tanım Listesi	21

2.8.3	Raporlar ve Grafikler Sayfası	22
2.8.4	Veri Birimi Ekleme Sayfası.....	23
2.8.5	Veri Birimi Listeleme Sayfası	23
2.8.6	Fonksiyon Ekleme Sayfası	24
2.8.7	Fonksiyon Listeleme Sayfası.....	25
2.8.8	Sensör Veri Türü Ekleme Sayfası.....	25
2.8.9	Sensör Veri Türü Listeleme Sayfası	26
2.8.10	Tanımsız sensör listeleme sayfası	26
2.8.11	Sensör ekleme sayfası	27
2.8.12	Sensör listeleme sayfası.....	28
3	Sonuçlar	29
3.1	Sistem Kullanımının Kontrolü.....	29
3.2	"Device Pin Options" İşlevselliğinin Kontrolü.....	30
3.3	Sensör Alanının İşlevselliğinin Kontrolü.....	31
4	Tartışma	32
	Kaynaklar	33

Şekiller Listesi

Şekil 2.1	ADS1115 ADC şematik.....	4
Şekil 2.2	Sıcaklık kayıt formatı	5
Şekil 2.3	Sıcaklık/Data bağlantısı	6
Şekil 2.4	DS18B20 Pin Çıkışları	6
Şekil 2.5	apt-get update ekran alıntısı	8
Şekil 2.6	apt-get upgrade ekran alıntısı	9
Şekil 2.7	Apache2 yüklenişi	9
Şekil 2.8	MariaDB yüklenişi	10
Şekil 2.9	PhpMyAdmin Yüklenişi	10
Şekil 2.10	Dosyalar için kullanıcı erişim izinleri	11
Şekil 2.11	Veritabanı yığını aktarımı	11
Şekil 2.12	phpMyAdmin veritabanı görüntüsü	12
Şekil 2.13	Giriş Sayfası	12
Şekil 2.14	I2C ayarları	13
Şekil 2.15	I2C aktif etme.....	14
Şekil 2.16	I2C aktif edildi ekranı.....	14
Şekil 2.17	Veri yolundaki I2C adresleri.....	15
Şekil 2.18	1-Wire ayarları	16
Şekil 2.19	1-Wire Aktif Etme.....	16
Şekil 2.20	Örnek sensör okuması.....	16
Şekil 2.21	Ham sensör aktivite tablosu	18
Şekil 2.22	Fonksyon hesaplama tablosu	18
Şekil 2.23	Sensör aktivite tablosu.....	19
Şekil 2.24	Çıkış kontrol tetikleyicileri	19
Şekil 2.25	Canlı sensör alanı	20
Şekil 2.26	Raspberry pi pin çıkışları.....	21
Şekil 2.27	Pin tanımlama sayfası.....	21
Şekil 2.28	Tanımlı pin çıkış listesi.....	21
Şekil 2.29	Rapor ve grafik sayfası	22
Şekil 2.30	Tanımlı pin çıkış listesi.....	23
Şekil 2.31	Değer birim ekleme sayfası.....	23

Şekil 2.32	Değer birimleri listeleme sayfası.....	24
Şekil 2.33	Örnek fonksiyon ekleme.....	24
Şekil 2.34	Fonksiyon listeleme sayfası	25
Şekil 2.35	Sensör veri türü ekleme sayfası.....	25
Şekil 2.36	Sensör veri türü listeleme sayfası.....	26
Şekil 2.37	Tanımsız sensör listeleme sayfası	26
Şekil 2.38	Tanımsız sensör ekleme sayfası	27
Şekil 2.39	Sensör ekleme sayfası.....	28
Şekil 2.40	Sensör listeleme sayfası	28
Şekil 3.1	Veritabanı sorgusu sırasında kaynak kullanımı	29
Şekil 3.2	Veritabanı tablo yapısı.....	29
Şekil 3.3	Örnek pin tetikleme	30
Şekil 3.4	Giriş/Çıkış pin durumları	30
Şekil 3.5	Giriş/Çıkış pin durumları	31
Şekil 3.6	Sensör alanı	31

Kısaltmalar Listesi

ADC	Analog-to-Digital Converter
SBC	Single-Board Computer
CSI	Camera Serial Interface
SD	Secure Digital
SQL	Structured Query Language
PHP	Hypertext Preprocessor (originally Personal Home Page)
I2C	Inter-Integrated Circuit
PGA	Programmable Gain Amplifier
VCC	Voltage at the Common Collector
GND	Ground
OS	Operating System
EXT4	Fourth Extended Filesystem
AJAX	Asynchronous JavaScript and XML
SSH	Secure Shell
PDF	Portable Document Format
CSV	Comma-Separated Values
mmHg	Millimeters of Mercury
kPa	Kilopascal
Atm	Atmosphere
IC	Integrated Circuit

Semboller Listesi

°C Santigrat derece

Bölüm 1

Giriş

Teknolojinin gelişimi ile birlikte, sağlık alanında, özellikle evde hasta bakımında önemli ilerlemeler kaydedilmektedir. Bu yenilikler, hareket kısıtlılığı olan hastalar için evde daha rahat ve özgür bir yaşamı mümkün kılmaktadır. Uzaktan sağlık takibi sistemleri, sağlık personelinin sürekli fiziksel varlığı olmadan hasta bakımını iyileştirerek sağlık hizmetlerinin verimliliğini artırmaktadır.(2020)

Yatalak hastaların karşılaştığı sorunlar yalnızca tıbbi değil, sosyal ve psikolojik boyutları da kapsamaktadır. Uzun süre yatakta kalan hastalar, yatak yaraları ve eklem rahatsızlıkları gibi fiziksel komplikasyonlar yanında, izolasyon ve depresyon gibi psikolojik zorluklarla da karşılaşabilirler. Bu yeni takip sistemi, hastaların vücut sıcaklığı, kalp atış hızı, kan basıncı gibi temel sağlık verilerini sürekli izleyerek, anormal durumları hızla tespit edebilir. Ayrıca, hastaların hareketlilik düzeyini ve yatak içi pozisyonlarını takip ederek yatak yaralarının önlenmesine yardımcı olabilir.(2022)

Raspberry Pi tabanlı bu sistem, ev otomasyonu ile entegre edilerek, soba ve lamba gibi aletlerin kontrolünü sağlayabilir, yatak ayarlarının otomatik değişimini gerçekleştirebilir. Hava kalitesi, sıcaklık, ışık miktarı ve nem gibi çevresel faktörlerin izlenmesi için kullanılan çeşitli sensörler, hastanın konfor ve sağlık standartlarını korumada kritik öneme sahiptir. Örneğin, hava kalitesi sensörleri, ortamın temiz ve sağlıklı olmasını sağlarken, sıcaklık sensörleri hastanın rahatını artırmak için ideal oda sıcaklığının korunmasına yardımcı olur. Işık ve nem sensörleri ise, cilt sağlığı ve uyku kalitesi gibi önemli faktörleri optimize eder. Raspberry Pi'ye bağlı bir kamera ile hastaların durumu uzaktan izlenebilir, böylece sağlık görevlileri ve aile üyeleri hastanın durumunu sürekli olarak takip edebilirler.

Bu sistem, hasta bakım süreçlerini kolaylaştırır ve evde tedavi gören hastaların yaşam kalitesini önemli ölçüde artırır. Hastalar ve bakım verenler arasında daha etkili bir iletişim kurulmasına imkan tanır, hastaların kendi sağlık durumları hakkında daha fazla bilgi sahibi olmalarını ve kendilerini daha güvende hissetmelerini sağlar. Sağlık hizmetlerinin erişilebilirliğini ve verimliliğini artırarak, hem hastalar hem de sağlık çalışanları için yararlı bir çözüm sunar.

Bu proje yazım kılavuzunun hazırlanmasında yurt içindeki ve yurt dışındaki çeşitli üniversitelerin ilgili yazım kılavuzlarından ve şablonlarından yararlanılmış, güncel ve estetik bir proje şablonu oluşturulmasına gayret gösterilmiştir. Bu kılavuzda, giriş kısmının ardından bazı bölümler bulunmaktadır. İkinci bölümde, yazım kuralları, format, görünüm ve ilgili diğer bilgiler bulunmaktadır. Üçüncü bölümde, projenin içeriğini oluşturan belli başlı bölümlerden bahsedilmiştir. Dördüncü bölüm projenin son kontrolleri ile ilgilidir. Son bölümde ise genel bir değerlendirme yapılmıştır.

Bölüm 2

Tasarım Ve Cihaz

Tasarlanan cihaz çeşitli parçalardan oluşmaktadır. Bu bölümler; Kontrol Ünitesi(Raspberry Pi), 16bit ADC çevirici(ADS1115), Dijital Sensörler(SHT11, DS18B20), Python Scriptleri(ADS1115 ADC entegrasyonu, DS18B20 ve DS18S20 sıcaklık sensörü entegrasyonu, SHT11 sıcaklık sensörü ve nem sensörü entegrasyonu).

DS18B20 sensörleri tarafından algılanan çevresel sıcaklıklar, çalışan bir Python betiği tarafından I2C veri yolu üzerinden okunur ve her sensör için belirlenen yerel bir PHP sayfasına bir "HTTP GET" isteği oluşturur. Bu PHP sayfası "web servisi" olarak adlandırılabilir. Bu PHP sayfasında, her sensör için kaydedilen veriler için SQL sorgularını çağıran fonksiyonlar bulunmaktadır.

Kaydedilen sıcaklık verileri, alarmlar üretmek ve hasta bakım odası ve hasta durumunu göstermek için bir indeks sayfası tarafından işlenecektir.

Kullanıcı, web arayüzünü kullanarak hasta odası uzaktan kontrol edebilir ve hasta odasında ya da hasta verilerinde bir sorun oluşursa web arayüzü bir alarm tetikleyebilir.

2.1 Kontrol Ünitesi

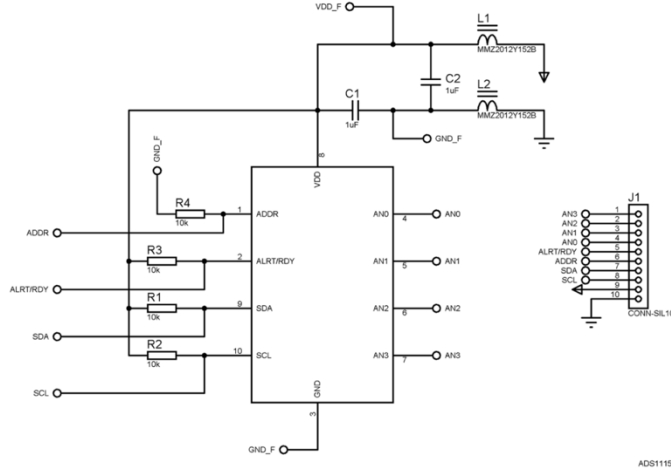
Raspberry Pi, Raspberry Pi Vakfı tarafından geliştirilen küçük bir tek kartlı bilgisayardır (SBC).

Geliştirilen sistem Raspberry Pi 2 versiyonunu kullanmaktadır ve Raspberry Pi 2 bu sürümün özellikleri şu şekildedir: 900MHz hızında dört çekirdekli ARM Cortex-A7 CPU'ya sahip, 1GB RAM belleği bulunuyor. Cihazda dört adet USB portu, 40 adet GPIO pini, tam bir HDMI portu, bir Ethernet portu ve birleşik 3.5mm ses jakı ve kompozit video çıkışı mevcut. Ayrıca bir kamera arayüzü (CSI), bir ekran arayüzü (DSI), bir mikro SD kart yuvası ve VideoCore IV 3D grafik çekirdeği bulunuyor. Raspberry Pi 2 tek kartlı bilgisayarın bize sunduğu özellikler projenin yapılabilmesi için yeterlidir.

Gömülü Linux sistemi, sensörlerden veri toplamak için yazılmış Python kodunu çalıştırabilir. Aynı zamanda, gömülü Linux sistemi SQL ve PHP'yi desteklediğinden bir web arayüzü geliştirilebilir. 16Bit Analog-Dijital dönüştürücü ve DS18B20 Sensörü, I2C haberleşmesiyle kontrol edilirken, SHT11 Sıcaklık ve Nem sensörü, I2C protokolüne benzer ancak tam olarak aynı olmayan bir protokol kullanarak 2 telli iletişim hattı ile kontrol edilir.

2.2 16 Bit Analog-Dijital Çevirici

Analog veriler, 16-bit yüksek aralık çözünürlüğü ölçebilen ADS1115 analog-dijital dönüştürücü ile dönüştürülür. Bu sensör, I2C ile uyumludur. ADS1115 ayrıca programlanabilir kazanç amplifikatörü (PGA) ve bir dijital karşılaştırıcı içerir. ADS111x, saniyede 860 örnekleme hızına kadar dönüşümler gerçekleştirebilir.



Şekil 2.1: ADS1115 ADC şematik

Şema Proteus ile yapılmıştır. Adres herhangi bir çıkışa bağlı değilse, varsayılan olarak değeri 0 olarak ayarlanır. VCC ve GND üzerinde bir ferit boncuğu bulunmasının ve bunların bir kapasitör ile desteklenmesinin nedeni, ADC devresinin girişindeki statik gürültüyü filtrelemek ve daha sağlıklı veri okuması elde etmektir. (Şekil 2.1).

ADS1115 için 2 ayrı çalışma modu vardır. Çalışma modu, yapılandırma kaydındaki bit ile değiştirilebilir.

2.2.1 Sürekli Mod

ADS1115'in sürekli modda çalışması için, mod bitinin 0 olarak ayarlanması gerekir. Devre bu moda ayarlandığında, sürekli bir konuşma gerçekleştirir ve sonucu konuşma kaydına bildirir. Yeni bir yapılandırma ayarı tanımlandığında, devam eden işlem eski yapılandırma ayarına göre gerçekleştirilir ve daha sonra yeni ayara göre başlar.

2.2.2 Tek Atış Modu

ADS1115'in tek atış modunda çalışması için, yapılandırma bitini 1 olarak ayarlamak veya cihazı sıfırlamak yeterlidir. Ayrıca, cihaz güce bağlandığında, varsayılan olarak tek atış modunda çalışmaya başlar.

2.3 DS18B20 Sıcaklık Sensörü

DS18B20 doğrudan dijital sıcaklık sensörüdür. Sıcaklık sensörünün çözünürlüğü, kullanıcı tarafından 9, 10, 11 veya 12 bit olarak yapılandırılabilir ve bunlar sırasıyla 0.5°C, 0.25°C, 0.125°C ve 0.0625°C artışlara karşılık gelir. Güç verildiğinde varsayılan çözünürlük 12-bittir. DS18B20, düşük güçte bekleme durumunda açılır. Sıcaklık ölçümü ve A'dan-D'ye dönüşümü başlatmak için, ana cihazın Convert T (44h) komutunu vermesi gerekir. Dönüşümün ardından, elde edilen termal veriler belleğindeki 2 baytlık sıcaklık kayıt alanında saklanır ve DS18B20 bekleme durumuna döner. Eğer DS18B20 dış bir kaynaktan güç alıyorsa, ana cihaz Convert T komutundan sonra "okuma zaman dilimleri" çıkarabilir ve DS18B20 sıcaklık dönüşümü devam ederken 0 ile ve dönüşüm tamamlandığında 1 ile yanıt verecektir.

DS18B20'nin çıkış sıcaklık verisi Santigrat derece olarak kalibre edilir. Sıcaklık verisi, sıcaklık kaydında 16 bit işaret genişletilmiş iki tamamlayıcı sayı olarak saklanır (Şekil 2.2). İşaret bitleri (S), sıcaklığın pozitif veya negatif olup olmadığını gösterir: pozitif sayılar için S = 0 ve negatif sayılar için S = 1'dir. Eğer DS18B20 12-bit çözünürlüğe yapılandırılmışsa, sıcaklık kaydındaki tüm bitler geçerli veriler içerecektir. 11-bit çözünürlük için, 0. bit tanımsızdır. 10-bit çözünürlük için, 1 ve 0. bitler tanımsızdır ve 9-bit çözünürlük için 2, 1 ve 0. bitler tanımsızdır. Şekil 2.3, 12-bit çözünürlük dönüşümleri için dijital çıkış verileri ve karşılık gelen sıcaklık okumalarının örneklerini vermektedir.

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹	2 ⁻²	2 ⁻³	2 ⁻⁴
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2 ⁶	2 ⁵	2 ⁴

S = SIGN

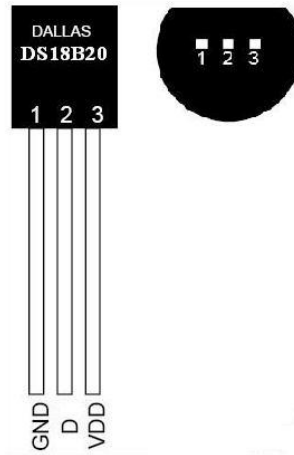
Şekil 2.2: Sıcaklık kayıt formatı

Sıcaklık sensörünün 3 pini vardır (Şekil 2.4) ve aşağıdaki gibi bağlanmalıdır:

- Vcc – 3,0 ila 5V
- Toprak - Toprak
- Data -dijital pin 2
- Data ile besleme gerilimi arasında 4,7 kohm değerinde pull-up direnç.

TEMPERATURE (°C)	DIGITAL OUTPUT (BINARY)	DIGITAL OUTPUT (HEX)
+125	0000 0111 1101 0000	07D0h
+85	0000 0101 0101 0000	0550h
+25.0625	0000 0001 1001 0001	0191h
+10.125	0000 0000 1010 0010	00A2h
+0.5	0000 0000 0000 1000	0008h
0	0000 0000 0000 0000	0000h
-0.5	1111 1111 1111 1000	FFF8h
-10.125	1111 1111 0101 1110	FF5Eh
-25.0625	1111 1110 0110 1111	FE6Fh
-55	1111 1100 1001 0000	FC90h

Şekil 2.3: Sıcaklık/Data bağlantısı



Şekil 2.4: DS18B20 Pin Çıkışları

2.4 SHT 11 sıcaklık ve nem sensörü

SHT11, İsviçre'nin Sensirion Şirketi tarafından patentli CMOSens teknolojisi ile üretilmiştir. SHT11'de dahili 14 bitlik AD dönüştürücü ve seri iletişim birimleri bulunmaktadır. Mikrodenetleyiciye 14-bit çözünürlükte sıcaklık verilerini (varsayılan değer 14 bit, 12 bit olarak ayarlanması da mümkündür) ve 12-bit çözünürlükte nem verilerini (varsayılan değer 12 bit, 8 bit olarak ayarlanması da mümkündür) iletmektedir. Bu SMD sensör, ± 0.5 °C hata payı ile -40 °C'den +125 °C'ye kadar sıcaklık ölçülebilir. Ayrıca, nem ölçümünü $\pm \%3.5$ hata payı ile gerçekleştirir.

Nem sensörleri, 14-bit analog-dijital dönüştürücü ve seri arayüz devresi ile sorunsuz bir şekilde birleştirilmiştir. Bu, üstün sinyal kalitesi, hızlı tepki süresi ve dış etkenlere karşı duyarsızlık sağlar.

SHT11, sıcaklık algılama, nem algılama, sinyal dönüştürme, ısıtıcı ve A/D dönüşümü gibi işlevleri tek bir çip içinde entegre eder. Çipin ölçüm aşamaları şu şekildedir:

- Nem ve sıcaklık, kapasitif polimer nem sensörü ve enerji boşluk malzemesinden yapılmış sıcaklık hassas bir eleman tarafından analog sinyallere dönüştürülür.
- Elektrik sinyali, zayıf sinyal amplifikatöründen geçerek amplifikasyon için hazırlanır.
- Amplifiye edilen analog sinyal, dijital ölçüm için 14 bit A/D dönüştürücünden geçer.
- Dijital sinyal, iki telli seri dijital arayüz aracılığıyla çıktı alır.

Çip, küçük boyutlu paketi, basit veri arayüzü, basit veri protokolü ve yüksek doğruluğu ile dikkat çeker. SHT11'in seri arayüzü, güç tüketimi ve sensör okumaları için optimize edilmiştir. Sensör, I2C protokolü üzerinden kullanılamaz. SHT11'in iki telli seri protokolü, standart I2C protokolünden farklıdır. SHT11 iletişim zamanlaması ile çalışmalıdır.

SHT11'in sensör iletişimi gerçekleşmesi bir sıraya göre olur. Ana program için başlatma işlemi yapıldıktan sonra, iletişim sıfırlama işlemi uygulanır. Bu işlemde, Raspberry Pi ve SHT11 arasındaki zamanlama hatalarını önlemek için iletişim bağlantısı yenilenir. İletişim sıfırlama işlemi yapıldıktan sonra, ölçüm işlemine başlamak için komut Raspberry Pi tarafından gönderilir. Sensör Raspberry Pi'dan komut aldığı anda, ölçüm işlemine başlar. Ölçüm işlemi tamamlandığında ve veri SHT11'den gönderilmeye hazır olduğunda, Sensör veri pini durumunu 0 olarak ayarlar. Bu noktada Raspberry Pi, SHT11'in veri pinini yoklar. Veri transfer için hazırsa, Raspberry Pi SHT11'den veriyi alır. Belirli bir süre için veri hazır değilse, bu süreç sırasında bir hata ile karşılaşıldığı anlamına gelir ve SHT11'den ölçüm verisi alınmaz. Bu sürecin çıkmaz durumuna düşmesini önlemek için, işlem zaman aşımına uğrar ve baştan başlar.

2.5 Raspberry Pi'nin hazırlanması

Raspberry Pi'nin kullanılabilir olması için, ARM için özel olarak derlenmiş işletim sistemlerinden birini içeren bir imajın boş bir SD kart ile yüklenmesi gerekmektedir.

Bu projede, Raspberry Pi için özel olarak hazırlanmış bir Raspbian işletim sistemi imajı 16GB boş bir SD kart ile yüklenmiş ve Raspberry Pi bu imaj üzerinden başlatılmıştır.

Projede kullanılan Raspberry Pi OS imajıdır. Raspberry Pi OS imajı SD karta yüklendikten sonra, bilgisayarda "Boot" adında bir disk görülebilir. Bu disk, Raspberry'nin SD karttan başlamasını sağlayan bölümdür. İkinci bölüm EXT4 formatında olduğu için bilgisayar tarafından algılanamaz ve görünmez.

SD kartı Raspberry Pi'ye yerleştirdikten sonra, Raspberry Pi'ye IP üzerinden erişebilmek için bilgisayarda görünen "Boot" dizinine bir dosya oluşturulmalıdır. Bu dosya, Raspberry Pi'ye imajı göndermek için kullanılan bilgisayarda oluşturulur ve herhangi bir uzantıya sahip olmamalı ve "SSH" olarak adlandırılmalıdır. Projede kullanılan bilgisayar OSX işletim sistemine sahip bir bilgisayar olduğu için, bu dosya OSX'in terminal penceresi açılarak "touch /Volumes/Boot/SSH" komutu girilerek oluşturulur. Ardından SD kart, Raspberry Pi'nin SD kart yuvasına yerleştirilir ve Raspberry Pi güç kaynağına bağlanır. Bu noktadan sonra Raspberry Pi'nin başlangıç süreci başlar.

Başlangıç tamamlandığında, Linux'taki tüm servisler artık çalışıyor durumdadır ve Raspberry Pi SSH üzerinden erişilebilir hale gelir. SSH erişiminden sonra, terminal üzerinden önce "sudo apt-get update" ve ardından "sudo apt-get upgrade" komutları girilerek tüm paketlerin güncel olduğundan emin olunmalıdır.

Terminalde "apt-get update" komutu girildiğinde, yerel sistemde yüklü paketler ile paket deposundaki versiyon arasındaki farklar aranır ve liste güncellenir (Şekil 2.5). "apt-get update" komutu, yerel sistemde herhangi bir yükleme gerçekleştirmez.

```
pi@raspberrypi:~ $ sudo apt-get update
Get:1 http://archive.raspberrypi.org/debian stretch InRelease [25.3 kB]
Get:2 http://raspbian.raspberrypi.org/raspbian stretch InRelease [15.0 kB]
Get:3 http://archive.raspberrypi.org/debian stretch/main armhf Packages [168 kB]
Get:4 http://archive.raspberrypi.org/debian stretch/ui armhf Packages [34.0 kB]
Get:5 http://raspbian.raspberrypi.org/raspbian stretch/main armhf Packages [11.7 MB]
Get:6 http://raspbian.raspberrypi.org/raspbian stretch/contrib armhf Packages [56.9 kB]
Fetched 12.0 MB in 21s (546 kB/s)
Reading package lists... Done
pi@raspberrypi:~ $
```

Şekil 2.5: apt-get update ekran alıntısı

"apt-get update" komutu ile paket listesini güncelledikten sonra, "apt-get upgrade" komutu yerel sistemdeki paket listesinin versiyonunu güncellenmiş paket listesi ile karşılaştırır (Şekil 2.6). Bu işlemin sonunda, kullanımdaki yerel sistemde eksik bir paket varsa, o paket mevcut paket versiyonuna yükseltilir.

```
pi@raspberrypi:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  ca-certificates dpkg dpkg-dev file libdpkg-perl libmagic-mgc libmagic1 libpam-systemd libraspberrypi-bin libraspberrypi-dev
  libudev1 patch raspberrypi-bootloader raspberrypi-kernel shared-mime-info systemd systemd-sysv tzdata udev
22 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 80.2 MB of archives.
After this operation, 239 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Şekil 2.6: apt-get upgrade ekran alıntısı

"apt-get install <paket adı veya paket adları>" komutu, paketin halihazırda kurulu olmayan ancak "apt-get update" komutuyla güncellenen güncel sürümünü yerel sisteme yükler.

2.5.1 Apache sunucusu, PHP, MariaDB, phpMyAdmin ve web sayfası kurulumu

Apache, Apache Software Foundation tarafından geliştirilen açık kaynak kodlu ve ücretsiz bir Web sunucusu programıdır. Dünya Çapında Web'in genişlemesinde önemli bir rol oynamıştır ve bugüne kadar İnternet üzerindeki en yaygın Web sunucusudur.

Apache, geliştirilen projenin web sayfasına ev sahipliği yapacak web sunucusudur. "apt-get install apache2 -y" komutu ile paket deposundan kolayca kurulum yapılabilir. Kurulum adımı sırasında karşılaşılabilecek ekran görüntüsü Şekil 2.7'da gösterilmiştir.

```
pi@raspberrypi:~$ sudo apt-get -y install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0 ssl-cert
Suggested packages:
  www-browser apache2-doc apache2-suexec-pristine | apache2-suexec-custom openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 0 not upgraded.
Need to get 4,953 kB of archives.
After this operation, 6,275 kB of additional disk space will be used.
Get:1 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf libapr1 armhf 1.5.2-5 [79.8 kB]
Get:2 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf libaprutil1 armhf 1.5.4-3 [75.9 kB]
Get:3 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf libaprutil1-dbd-sqlite3 armhf 1.5.4-3 [17.9 kB]
Get:4 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf libaprutil1-ldap armhf 1.5.4-3 [16.0 kB]
Get:5 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf liblua5.2-0 armhf 5.2.4-1.1 [82.8 kB]
Get:6 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf apache2-bin armhf 2.4.25-3+deb9u5 [1,042 kB]
Get:7 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf apache2-utils armhf 2.4.25-3+deb9u5 [219 kB]
Get:8 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf apache2-data all 2.4.25-3+deb9u5 [162 kB]
Get:9 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf apache2 armhf 2.4.25-3+deb9u5 [235 kB]
Get:10 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf ssl-cert all 1.0.39 [20.8 kB]
Fetched 4,953 kB in 12s (155 kB/s)
```

Şekil 2.7: Apache2 yüklenişi

Apache2 kurulumu tamamlandıktan sonra, Apache için PHP desteği sağlayacak paketler "apt-get -y install php7.0 libapache2-mod-php7.0 php7.0-mysql php7.0-curl php7.0-gd php7.0-intl php-pear php-imagine php7.0-imap php7.0-mcrypt php-memcache php7.0-pspell php7.0-recode php7.0-sqlite3 php7.0-tidy php7.0-xmlrpc php7.0-xsl" komutu ile çalışan sisteme yüklenebilir.

MariaDB, MySQL'in yeni sürümü için yeni isimdir. MariaDB'nin kurulumu, MySQL sunucusunun kurulumu ile aynıdır. Sensör verileri, kullanıcılar vb. gibi bilgiler MariaDB'de saklanacaktır. MariaDB'nin kurulumu, terminalde "sudo apt-get -y install mariadb-server mariadb-client" yazılarak yapılır. Bu komut terminal üzerinden gönderildiğinde, ekran çıktısı Şekil 2.8'de gösterildiği gibi olacaktır.

```
pi@raspberrypi:~$ sudo apt-get -y install mariadb-server mariadb-client
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  galera-3 gawk libaio1 libarchive13 libcgilint-perl libcgilint-perl libdbd-mysql-perl libdbi-perl libencode-locale-perl libfcgi
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl libjemalloc1 liblwp-mediatypes-perl liblz2-2 lib
  libterm-readkey-perl libtimedate-perl liburi-perl lsof mariadb-client-10.1 mariadb-client-core-10.1 mariadb-common mariadb-ser
Suggested packages:
  gawk-doc lrzip libclone-perl libmldbm-perl libnet-daemon-perl libsql-statement-perl libdata-dump-perl libipc-sharedcache-perl
The following NEW packages will be installed:
  galera-3 gawk libaio1 libarchive13 libcgilint-perl libcgilint-perl libdbd-mysql-perl libdbi-perl libencode-locale-perl libfcgi
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl libjemalloc1 liblwp-mediatypes-perl liblz2-2 lib
  libterm-readkey-perl libtimedate-perl liburi-perl lsof mariadb-client-10.1 mariadb-client-core-10.1 mariadb-client-core-10.1 mariadb-com
mariadb-server-core-10.1 mysql-common socat
0 upgraded, 35 newly installed, 0 to remove and 0 not upgraded.
Need to get 23.7 MB of archives.
After this operation, 174 MB of additional disk space will be used.
```

Şekil 2.8: MariaDB yüklenişi

Phpmyadmin, MariaDB'de bulunan veritabanlarını yönetmek için görsel bir web arayüzü sağlayan bir yazılımdır. Bu yazılım ile MariaDB üzerine SQL veritabanlarını içe aktarmak, herhangi bir veritabanındaki tabloları değiştirmek mümkündür. Bu, veritabanı tarafında gerçekleştirilmesi gereken işlemlerin kolay ve görsel bir şekilde yapılmasını sağlar. PhpMyAdmin'i kurmak için, terminal üzerinden "apt-get install phpmyadmin -y" komutu çalıştırılmalıdır (Şekil 2.9). Kurulum sırasında, program kullanıcıyı yönlendirir ve gerekli işlemleri tamamlar.

```
pi@raspberrypi:~$ sudo apt-get -y install phpmyadmin
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  dbconfig-common dbconfig-mysql javascript-common libjs-jquery libjs-jquery libjs-sphinxdoc libjs-underscore libzip4 php-bz2 php-mbstring php-mysql php-php-gettext php-phoseclib php-tcpdf
  php-zip php7.0-bz2 php7.0-mbstring php7.0-zip
Suggested packages:
  php-libsodium php-gmp php5-imagick www-browser
Recommended packages:
  php5-gd php5-mcrypt
The following NEW packages will be installed:
  dbconfig-common dbconfig-mysql javascript-common libjs-jquery libjs-jquery libjs-sphinxdoc libjs-underscore libzip4 php-bz2 php-mbstring php-mysql php-php-gettext php-phoseclib php-tcpdf
  php-zip php7.0-bz2 php7.0-mbstring php7.0-zip phpmyadmin
0 upgraded, 18 newly installed, 0 to remove and 0 not upgraded.
Need to get 13.4 MB of archives.
After this operation, 50.3 MB of additional disk space will be used.
Get:2 http://rasbian.raspberrypi.org/raspbian stretch/main armhf dbconfig-mysql all 2.0.0 [996 B]
Get:1 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf dbconfig-common all 2.0.0 [647 kB]
Get:3 http://mirrors.linux-bg.org/raspbian/raspbian stretch/main armhf javascript-common all 11 [6,120 B]
```

Şekil 2.9: PhpMyAdmin Yüklenişi

Kurulum tamamlandıktan sonra, phpMyAdmin'i çalıştırmak için yapılması gereken son işlem, Apache için hazırlanan Phpmyadmin'in yapılandırma dosyası olan "Include

"/etc/phpmyadmin/apache.conf" satırını, "nano /etc/apache2/apache2.conf" komutu girildikten sonra Apache2'nin yapılandırmasına eklemektir.

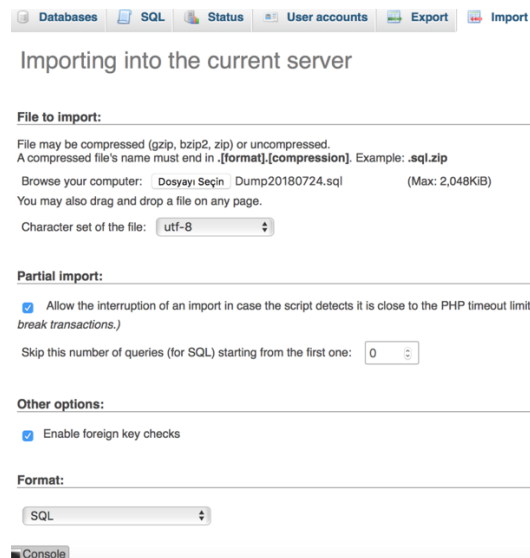
Kurulum tamamlandıktan sonra, /var/www dizinleri için dosya izin ayarları ayarlanmalıdır. Bu ayar sonucunda, "pi" ve "www-data" kullanıcıları bu klasörlere ve klasördeki tüm dosyalara erişim izni verilecek ve bu grup yetkisiyle "www-data" ve "pi" kullanıcıları adı geçen klasörlere erişebileceklerdir. Bu, Şekil 2.10'da gösterilen komut dizisi işlenerek yapılır.

```
pi@raspberrypi: ~$  
pi@raspberrypi: ~$ chgrp -R www-data /var/www  
pi@raspberrypi: ~$  
pi@raspberrypi: ~$ chmod -R g+rw /var/www  
pi@raspberrypi: ~$
```

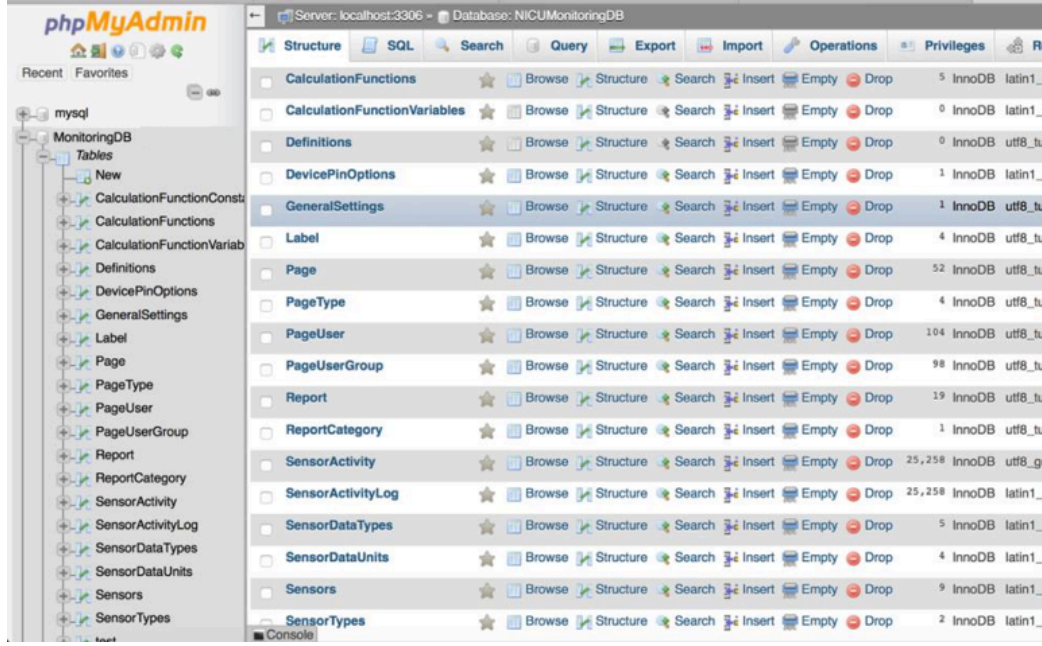
Şekil 2.10: Dosyalar için kullanıcı erişim izinleri

Bunu yaptıktan sonra, PHP ve AJAX ile geliştirilen web sayfası kodu, "Filezilla" yazılımı aracılığıyla "/var/www/html" dizinine yüklenir. Ardından, kodun geliştirildiği sunucudan veritabanı dökümü alınır. PhpMyAdmin aracılığıyla Şekil 2.11'da gösterildiği gibi içe aktarılır. İçe aktarma işleminden sonra alınan bir sonraki ekran görüntüsü Şekil 2.12'de görülebilir.

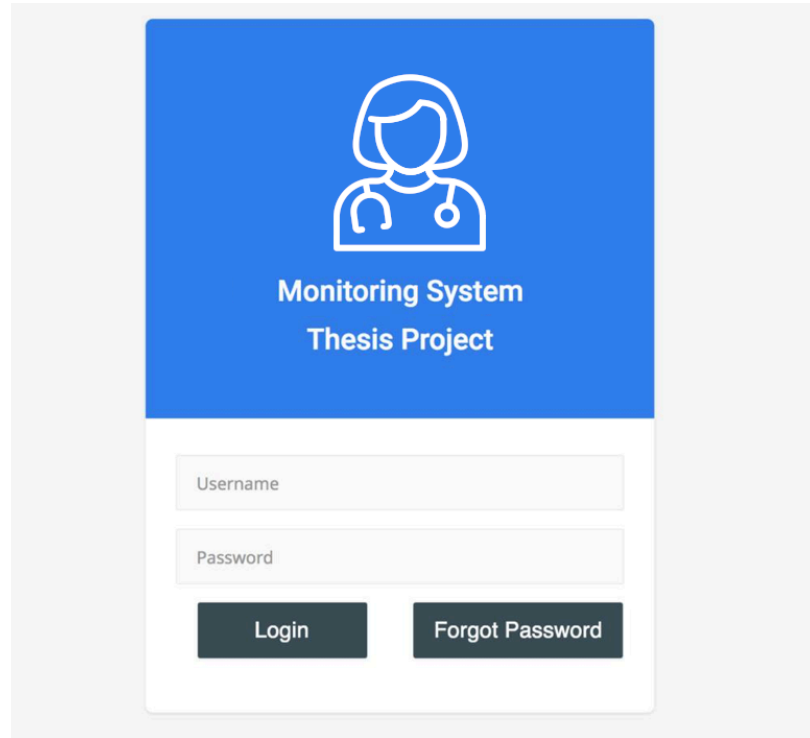
Veritabanı içe aktarıldıktan sonra, web sayfası ile veritabanı arasındaki bağlantıyı sağlayan veritabanı kullanıcı adı ve şifresi, Config.php dosyasına kaydedilir. Ardından, aynı ağa bağlı herhangi bir bilgisayar, Raspberry Pi'nin IP adresini tarayıcının web sayfasına girerek siteye erişebilir (Şekil 2.13).



Şekil 2.11: Veritabanı yığını aktarımı



Şekil 2.12: phpMyAdmin veritabanı görüntüsü



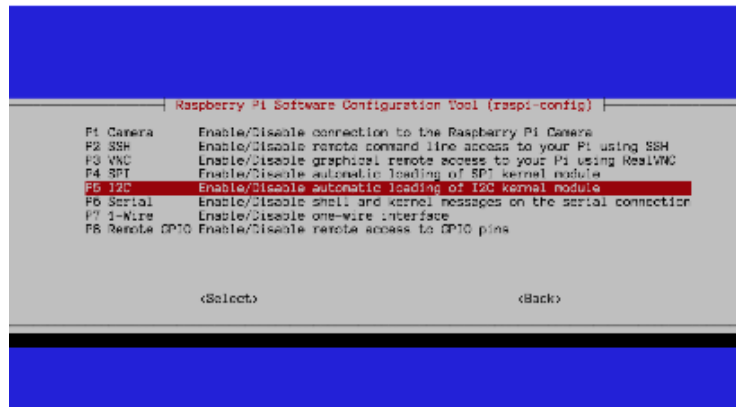
Şekil 2.13: Giriş Sayfası

2.6 Python Scriptleri

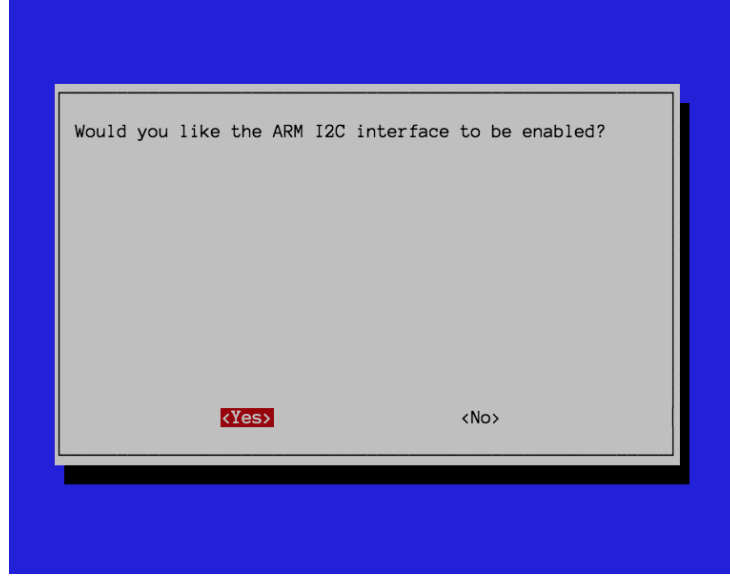
İzleme Sistemi projesi, MariaDB'nin içerdiği "MonitoringDB" veritabanındaki "SensorActivity" tablosuna sensör değerlerini kaydetmek için "GET" yöntemiyle çalışan bir web servis sayfasını sürdürmektedir. Bu sayfanın yolu "/var/www/html/MonitoringProject/Services/SensorActivityWS/URL/index.php" şeklindedir. Bu sayfadaki fonksiyon iki değişken alır. Bu değişkenler sensorId (sensör adı) ve sensorDataValue (okunan sensör değeri)dir. Bu değerleri sensörlerden okumak ve İzleme Sistemi sitesine web servisi aracılığıyla göndermek için her sensör türü için ayrı bir script hazırlanmıştır.

2.6.1 ADS1115 ADC

ADS1115, 16-bitlik, 4-kanallı analog verileri okumak ve göndermek için I2C protokolünü kullanır. Bu IC'den veri okumak için Raspberry Pi'nin I2C arayüzünün etkinleştirilmesi gerekmektedir. I2C arayüzünü etkinleştirmek için, Raspberry ayarlarının değiştirilmesi ve bunu yapmak için öncelikle "raspi-config" komutunun girilmesi gerekmektedir. Sonraki ekrandan, 5 numara ile belirtilen "Interfacing Options" menüsü seçilmelidir. Menü Şekil 2.14'te görülebilir. Daha sonra, "P5 I2C" olarak belirtilen öğe seçilmeli ve son olarak, Şekil 2.15'te gösterildiği gibi gelen sorgu için "EVET" seçeneği seçilmelidir. Raspberry Pi'nin I2C arayüzü artık etkinleştirilmiştir (Şekil 2.16).

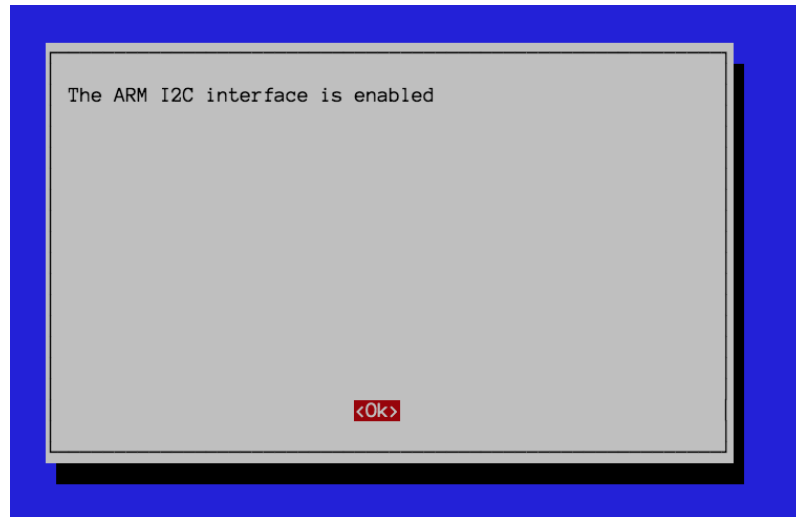


Şekil 2.14: I2C ayarları



Şekil 2.15: I2C aktif etme

Etkinleştirildikten sonra, I2C protokolü üzerinden iletişim kuran IC'ler I2C adreslerinde görünür ve iletişime geçilebilir hale gelir. Kullanıcı, satırdaki IC'leri ve adreslerini görmek için "sudo i2cdetect -y 1" komutunu kullanabilir. Bu komutun çıktısı Şekil 2.17'da görüldüğü gibidir.



Şekil 2.16: I2C aktif edildi ekranı

ADS1115 IC'nin adresi "48"dir. Bu IC ile iletişim kurmak gerektiğinde, 48 adresi kullanılmalıdır. ADS1115 IC için Python için geliştirilmiş paketler bulunmaktadır.

"git clone https://github.com/adafruit/Adafruit_Python_ADS1x15.git" komutu ile paket, sisteme kolayca indirilip kurulabilir.

```
pi@raspberrypi:~ $ sudo i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  48  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

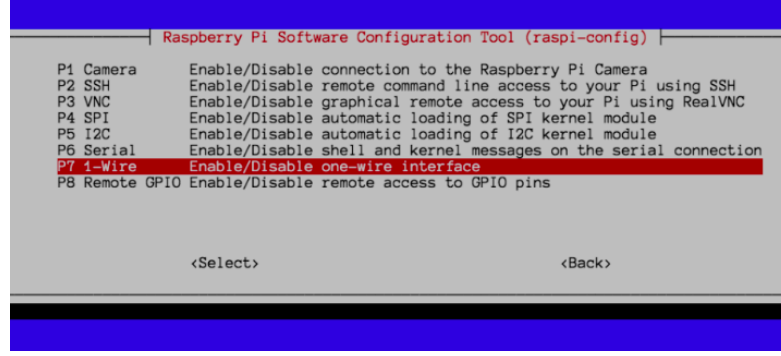
Şekil 2.17: Veri yolundaki I2C adresleri.

Sisteme kurulum için, kullanıcının sistemine "apt-get install python3-setuptools python3-pip" paketlerini kurması gerekebilir. Paketler zaten indirilmişse, indirilen Adafruit_Python_ADS1x15 klasörüne girilerek "python3 setup.py install" komutu çalıştırılarak sisteme yüklenir. Ardından ADS1115 için Python betiğini çalıştırmak için "python3 ADS1115_READ.py" komutu kullanılır. Kod içindeki kazanç değeri, okunacak voltaj aralığını belirler.

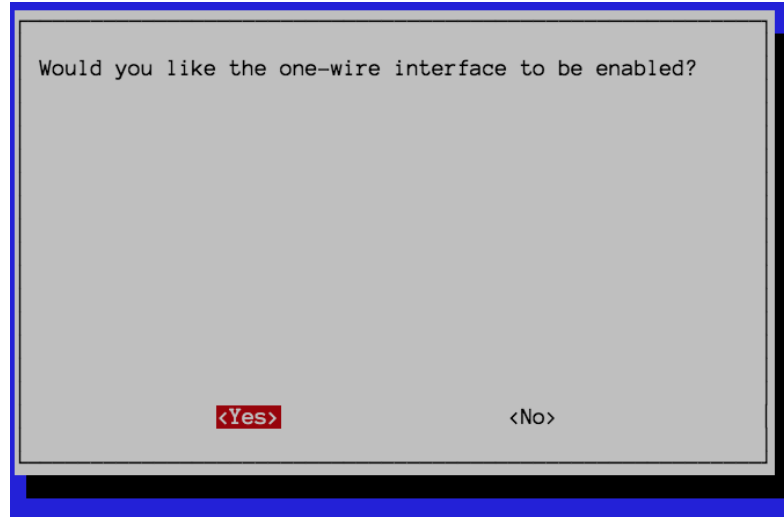
Bu betik, her beş saniyede bir analog kanalları okur ve her birini ayrı ayrı web servisine gönderir.

2.6.2 DS18B20 ve DS18S20 sıcaklık sensörleri

DS18B20 ve DS18S20, onewire olarak adlandırılan tek bir hatta iletişim kurar. Veri alışverişi bu hat üzerinden gerçekleşir. Raspberry Pi, 1-wire protokolünü desteklemektedir. Bu iletişim arayüzünü etkinleştirmek için, "raspi-config" komutu girildikten sonra, kullanıcının sonraki ekranda 5 numara ile belirtilen "Interfacing Options" menüsüne girmesi, Şekil 2.18'de gösterildiği gibi seçeneği seçmesi ve ardından Şekil 2.19'da gösterildiği gibi "Evet" seçeneğini seçmesi gerekmektedir. Bu işlemin sonunda, one-wire etkinleştirilir.



Şekil 2.18: 1-Wire ayarları.



Şekil 2.19: 1-Wire Aktif Etme.

Etkinleştirildikten sonra, one-wire veri hattına bağlı sensörler iletişime başlar. “ls /sys/bus/w1/devices” komutunun çıktısında “10-” ve “28-” ile başlayan sensörler, DS18B20 ve DS18S20 sensörleridir. Bu sensörler, “tail /sys/bus/w1/devices/10000xxxxxxxx/w1_slave” komutu ile kolaylıkla okunabilir. Örnek çıktı Şekil 2.20’de gösterilmiştir.

```
/sys/bus/w1/devices/10-000002288677/w1_slave  
2d : crc=2d YES  
2d t=30187
```

Şekil 2.20: Örnek sensör okuması

/devices klasöründe listelenen her klasör, bir sensöre aittir. “10-” ve “28-” ile başlayan klasör isimleri, sensörlerin benzersiz tanımlayıcılarıdır. Yazılan Python betiği, bu klasördeki klasörlerin listesini alır, her klasördeki "w1_slave" dosyasının çıktısını okur

ve metin kırpma işleminden sonra, elde edilen sayı 1000'e bölünerek sıcaklık değeri elde edilir.

DS1820.py betiği, sensör okumalarını gerçekleştirir ve "GET" yöntemi kullanarak değerleri web servisine gönderir.

2.6.3 SHT11 sıcaklık ve nem sensörü

SHT11, nem ve sıcaklık ölçümünde yüksek doğruluk sağlayan Sensirion tarafından geliştirilmiş bir sensördür. Bu sensörün kendine özgü bir veri protokolü vardır. Python için yazılmış paketler Github üzerinde bulunabilir.

Projedeki sht11 sensöründen veri okumak için, "pip3 install git + <https://github.com/drohm/pi-sht1x>" komutu ile pi-sht1x paketi sisteme kuruldu. Bu adımdan sonra SHT11 için bir Python betiği yazılabilir., SHT11 için yazılan Python betiğini göstermektedir. Bu betik, on saniyede bir sensörden sıcaklık ve nem değerlerini okur ve bu değerleri web servisi aracılığıyla siteye gönderir.

2.7 Veritabanı

Web sayfası, olabildiğince kullanıcı dostu olacak şekilde tasarlanmıştır. Üst bölümde, sensörlerden anlık verilerin gösterildiği bir alan bulunmaktadır. Bu bölüme ulaşmadan önce sensör verileri, Python betikleri yardımıyla ham veri olarak okunur. Daha sonra, proje "GET" yöntemiyle web servis bölümüne aktarılır. Veritabanına iletilen ham veriler, "SensorActivityLog" adlı tabloda ham veri olarak saklanmaktadır. Örnek bir çıktı Şekil 2.21'te gösterilmiştir.

Python betiği ve web servisi aracılığıyla alınan veriler, Şekil 2.22'da gösterilen "CalculationFunctions" adlı tabloda tanımlanan fonksiyonların formüllerine uygun olarak hesaplandıktan sonra Şekil 2.23'de gösterilen "SensorActivity" adlı tabloda kaydedilir. Kullanılacak sensör ve fonksiyon, sensör veri türüne bağlı olarak ayarlanır.

	id	receivedSensorId	receivedSensorDataValue
<input type="checkbox"/> Edit Copy Delete	58476	5	30.88
<input type="checkbox"/> Edit Copy Delete	58477	13	13246
<input type="checkbox"/> Edit Copy Delete	58478	7	30.88
<input type="checkbox"/> Edit Copy Delete	58479	9	30.71
<input type="checkbox"/> Edit Copy Delete	58480	8	30.88
<input type="checkbox"/> Edit Copy Delete	58481	10	26276
<input type="checkbox"/> Edit Copy Delete	58482	11	23410
<input type="checkbox"/> Edit Copy Delete	58483	12	13144
<input type="checkbox"/> Edit Copy Delete	58484	13	13250
<input type="checkbox"/> Edit Copy Delete	58485	6	47.03
<input type="checkbox"/> Edit Copy Delete	58486	5	30.88
<input type="checkbox"/> Edit Copy Delete	58487	9	30.71
<input type="checkbox"/> Edit Copy Delete	58488	10	26256
<input type="checkbox"/> Edit Copy Delete	58489	7	30.88
<input type="checkbox"/> Edit Copy Delete	58490	11	23406
<input type="checkbox"/> Edit Copy Delete	58491	8	30.94
<input type="checkbox"/> Edit Copy Delete	58492	12	13142
<input type="checkbox"/> Edit Copy Delete	58493	13	13244
<input type="checkbox"/> Edit Copy Delete	58494	6	47.1
<input type="checkbox"/> Edit Copy Delete	58495	9	30.72
<input type="checkbox"/> Edit Copy Delete	58496	5	30.88
<input type="checkbox"/> Edit Copy Delete	58497	7	30.88
<input type="checkbox"/> Edit Copy Delete	58498	10	26264

Şekil 2.21: Ham sensör aktivite tablosu

	id	functionName	functionDescription	formulation
<input type="checkbox"/> Edit Copy Delete	4	celsius	general celsius function for digital temperature s...	x
<input type="checkbox"/> Edit Copy Delete	5	humidity	general function for digital humidity sensors	x
<input type="checkbox"/> Edit Copy Delete	6	ADS1115 VOLT	ADS1115 VOLT CONVERT	(4.096/65535)*x

Şekil 2.22: Fonksyon hesaplama tablosu

Bu bilgilerin bulunduğu tablo, "Sensors" tablosuna ayarlanmıştır. "Sensors" tablosu, sensörün adını, analog mu yoksa dijital mi olduğunu, hangi tür veriye sahip olduğunu, "GET" komutuyla gönderilen sensör adını, sensör için ayarlanmış alarm durum değerini ve alarm durumunu içerir. Veritabanı ayrıca, bağlı cihazların işlem yapmaları için tetik sinyalleri üretebilmelerini sağlayan pin çıkışlarını ve pin adlarını "DevicePinOptions" tablosunda içermektedir.

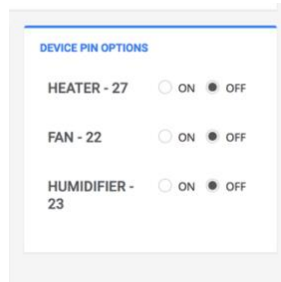
id	sensorid	activityValue
58476	6	47.07
58477	13	0.8278750
58478	7	30.88
58479	9	30.71
58480	8	30.88
58481	10	1.6422500
58482	11	1.4631250
58483	12	0.8215000
58484	13	0.8281250
58485	6	47.03
58486	5	30.88
58487	9	30.71
58488	10	1.6410000
58489	7	30.88
58490	11	1.4628750
58491	8	30.94
58492	12	0.8213750
58493	13	0.8277500
58494	6	47.1
58495	9	30.72
58496	5	30.88
58497	7	30.88

Şekil 2.23: Sensör aktivite tablosu

2.8 Web sayfası ve içeriği

Raspberry Pi'ye atanmış olan DHCP sunucusu ile ilişkilendirilen IP adresi girildiğinde karşılaşılabilecek ekran Şekil 2.13'de gösterildiği gibidir. Bu sayfada kullanıcı bilgileri girilerek sisteme giriş yapıldığında karşılaşılabilecek ekran temelde 3 bölümden oluşmaktadır.

Bunlar, Şekil 2.24'da gösterildiği gibi çıkış kontrol tetikleyicilerini sağlayan pin ayarları, herhangi bir sensör tanımlanan alarm değerine ulaştığında alarm veren ve Şekil 2.25'te gösterildiği gibi alarm durumunda olmadığında sensörlerin canlı değerlerini gösteren bir alan, ayar yapma, izleme, raporlama ve tüm sistemle ilgili yapılandırmaları kontrol etme için bir menü alanı bulunmaktadır.



Şekil 2.24: Çıkış kontrol tetikleyicileri



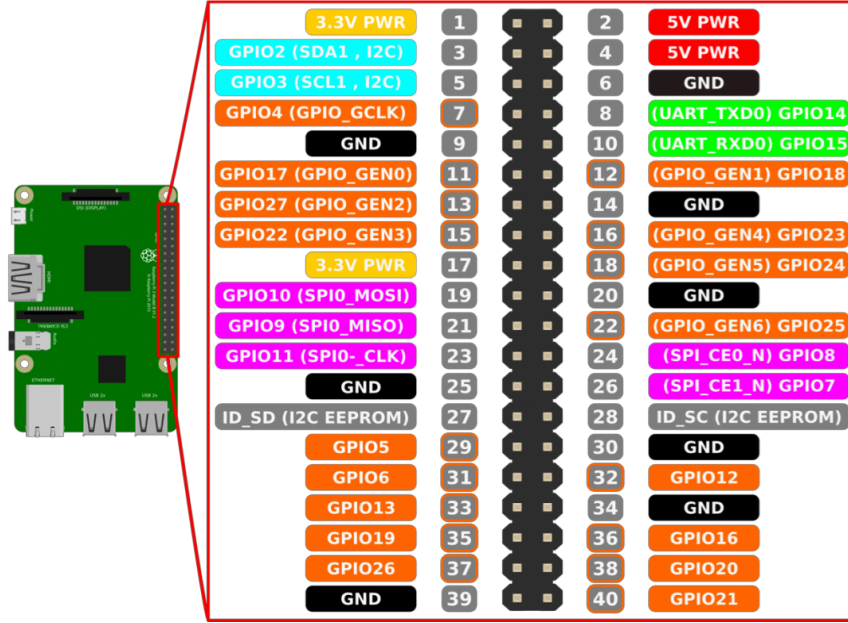
Şekil 2.25: Canlı sensör alanı

Python betiği tarafından okunan sensör verilerinin projenin web servisine gönderilebilmesi için, verilerin gerekli sayfalar ve çeşitli işlemlerden geçtikten sonra sistemdeki veritabanı tablolarına eklenmesi gerekmektedir.

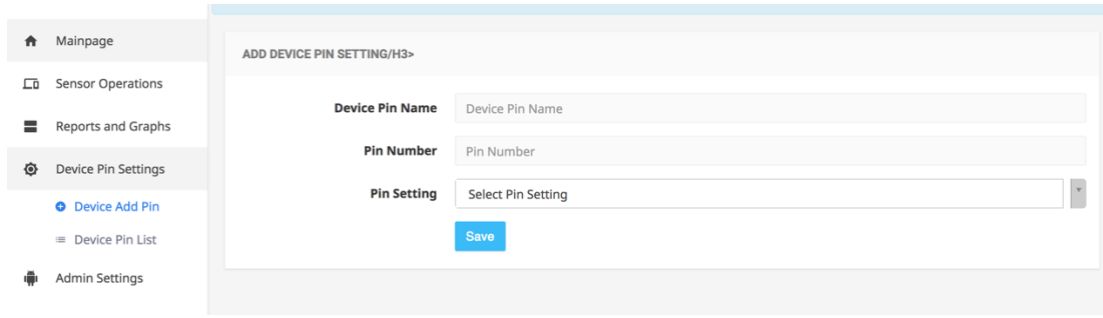
2.8.1 Sisteme tetikleyici pinler atama

Bakım odası içindeki cihazları, örneğin fan, nemlendirici ve ısıtıcıyı uzaktan kontrol etmek hasta için yararlı olacaktır ve hastanın rahatsız olmasını önleyecektir. Bu amaçla, ekranda sağ tarafta "DEVICE PIN OPTIONS" olarak adlandırılan belirli bir alan, Raspberry Pi'deki tüm kullanılabilir GPIO pinlerinin kontrolü için tasarlanmıştır ve bu Şekil 2.24'da görülebilir.

Burada bahsedilen cihazlar "DEVICE NAME – GPIO ID in Use" formatında gösterilir. Örneğin, ISITICI için, GPIO pin çizelgesinde 17 numara ve sıralı olarak 11 numara olan pin kullanılır (Şekil 2.26). Başka bir cihaz tanımlanması gerekiyorsa, sol taraftaki menüden "Device Add Pin" seçeneği seçilmelidir. Bu bölümde, cihazın adı, tetiklenecek olan GPIO pin numarasının kimliği ve mevcut bir bağlantı durumunda açık veya kapalı varsayılan değeri seçilir (Şekil 2.27). Yeni cihaz eklendiğinde, Şekil 2.24'da görülebilen "DEVICE PIN OPTIONS" altında sağ taraftaki cihazlar listesine eklenir. Eklenen cihaz, atanan pin üzerinden kontrol edilebilir hale gelir.



Şekil 2.26: Raspberry pi pin çıkışları



Şekil 2.27: Pin tanımlama sayfası

2.8.2 Cihaz-Pin Tanım Listesi

Bu sayfada, "Device Add Pin" sayfası aracılığıyla sisteme tanımlanan cihaz-pin tanımları, Şekil 2.28'de gösterildiği gibi, listelenir ve PDF, EXCEL ve CSV dosya formatlarına dışa aktarılabilir.

The screenshot shows the 'DEVICE PIN SETTING LIST' web interface. It has a sidebar menu on the left with the following items: Mainpage, Sensor Operations, Reports and Graphs, Device Pin Settings (selected), Device Add Pin, Device Pin List, and Admin Settings. The main content area contains the following table:

DEVICE PIN SETTING LIST

Total Row Count: 3

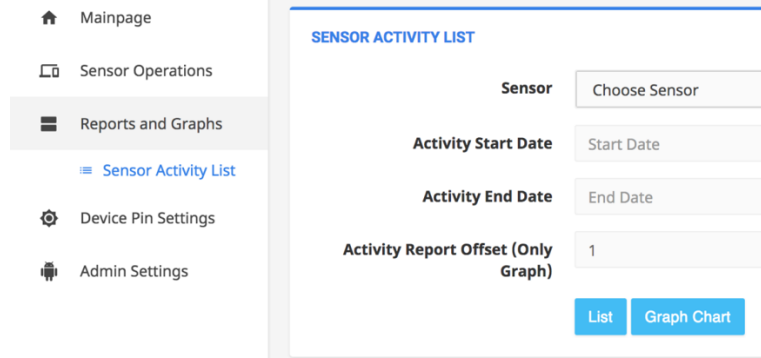
Copy Excel CSV PDF

Order No	Option Name	Pin Number	Pin Value
1	HEATER	27	OFF
2	FAN	22	OFF
3	HUMIDIFIER	23	OFF

Şekil 2.28: Tanımlı pin çıkış listesi

2.8.3 Raporlar ve Grafikler Sayfası

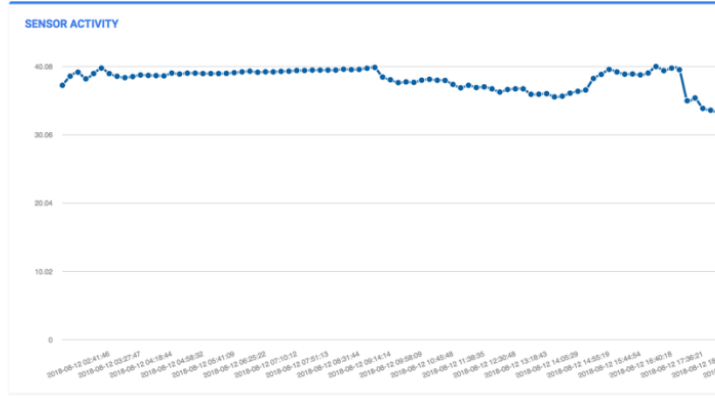
Bu sayfadan (Şekil 2.29), sistemde kayıtlı sensörlerin değerlerinin listesi alınabilir veya grafik olarak görüntülenebilir.



Şekil 2.29: Rapor ve grafik sayfası

Liste veya grafik istenen sensör seçildikten sonra, "Activity Start Date" ve "Activity End Date" bölümleri, bu tarihler için alınan verileri filtrelemek için kullanılır.

Daha sonra, eğer sadece verilerin listelenmesi gerekiyorsa, "List" Butonuna tıklanır. Sistem işlemi tamamlandığında, seçilen sensör belirtilen tarih aralığında verileri listeler. Grafik çıktısı isteniyorsa, "Activity Report Offset" bölümü de belirtilmelidir. Bu alanın varsayılan değeri 1'dir. "Activity Report Offset" alanı, örnekleme miktarını belirler. Örneğin, kullanıcının 100 verisi varsa ve bu alana 10 değerini girerse, her 10 veriden 1'i alınır ve kullanıcının toplamda 10 verisi olur. Grafik sayfası örneğinde "Activity Report Offset" 50 olarak ayarlanmıştır, bu Şekil 2.30'te görülebilir. Bu durumda, her 50 örnekten 1'i alınarak grafikler oluşturulmuştur. Oluşturulan grafik aynı zamanda tarih bilgisini de içerir.



Şekil 2.30: Tanımlı pin çıkış listesi

2.8.4 Veri Birimi Ekleme Sayfası

Bu sayfada (Şekil 2.31), fonksiyon sonucunda çıktı olarak verilecek değerin birimleri tanımlanır. Bu sayfada tanımlanan birim, ana sayfanın sensörler bölümünde gösterilecektir. Örneğin, basınç sensörü kullanılacaksa, kullanılacak fonksiyon için sonuç değeri "mmHg", "kPa" veya "atm" gibi bir değer olmalıdır.

Şekil 2.31: Değer birim ekleme sayfası

2.8.5 Veri Birimi Listeleme Sayfası

Bu sayfada, "Veri Birimi Ekle" bölümünde (Şekil 2.31) sistemde tanımlanmış birim türleri (Şekil 2.32) listelenir. Ayrıca, bu sayfadan bu veriler PDF, EXCEL, CSV formatlarına dışa aktarılabilir.

Order No	Unit Name
4	celsius
5	volt
6	%

Şekil 2.32: Değer birimleri listeleme sayfası

2.8.6 Fonksiyon Ekleme Sayfası

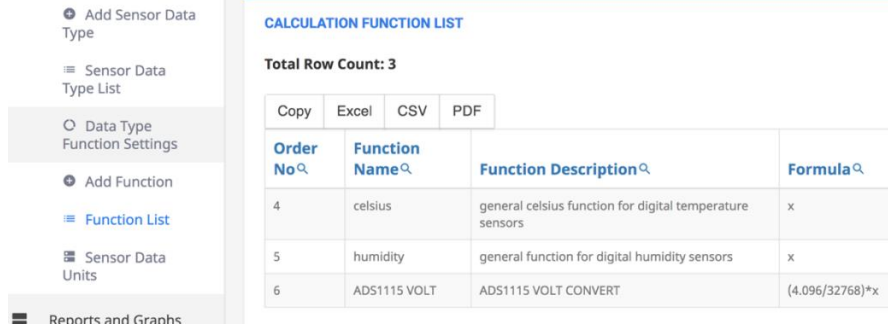
Bu sayfada (Şekil 2.33), ham verileri işleyecek fonksiyonlar tanımlanır. Örneğin, daha önce bahsedildiği gibi, ADS1115 IC 16 bit çözünürlüğe sahip bir ADC IC'dir. Bu IC'den alınacak analog değerler 2 olacaktır, ancak ADS1115 IC'nin negatif alma kapasitesi olduğu için bu değer -32768 ile 32767 arasında olacaktır. Python betiğinde analog alım sırasında kazanç değeri 1 olarak ayarlandığı için alınan voltaj değeri -4096 ile +4096 arasında değişecektir. Sensörden alınan verileri voltaja dönüştürmek için yapmamız gereken formüller şunlardır: $Voltage = \frac{4096}{32768} * x$

"x" Python betiğinden gelen ham veri olarak ele alınır. Bu denklem Şekil 2.33'de gösterildiği gibi sisteme girilmelidir.

Şekil 2.33: Örnek fonksiyon ekleme

2.8.7 Fonksiyon Listeleme Sayfası

Bu sayfada, fonksiyonel denklemler Şekil 2.34'da görülebileceği gibi listelenir. Ayrıca, bu sayfadan bu veriler PDF, EXCEL, CSV formatlarına dışa aktarılabilir.

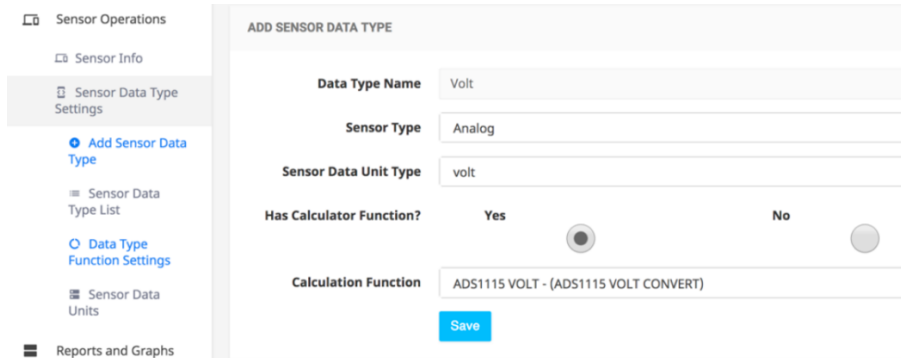


Order No	Function Name	Function Description	Formula
4	celsius	general celsius function for digital temperature sensors	x
5	humidity	general function for digital humidity sensors	x
6	ADS1115 VOLT	ADS1115 VOLT CONVERT	(4.096/32768)*x

Şekil 2.34: Fonksiyon listeleme sayfası

2.8.8 Sensör Veri Türü Ekleme Sayfası

Bu sayfada, daha önce tanımlanmış fonksiyonun ve ayrıca birim türü tanımlarının bir sensör türü için birleştirilmesi gerçekleşir. "Veri Türü Adı" bölümünde, oluşturulacak veri türü adı girilir. "Sensör Türü" bölümünde, kullanılacak sensör türü girilir. Şekil 2.31'da gösterilen sayfada eklenen birim "Sensör Veri Birimi Türü" bölümünde seçilir. "Hesaplama Fonksiyonu Var mı" seçeneğinde, bu veri türünün bir fonksiyon kullanıp kullanmayacağına ilişkin "Evet" veya "Hayır" yanıtı verilir. Şekil 2.35'da gösterilen örnekte, daha önce oluşturulan fonksiyon Şekil 2.33 seçilir ve "Kaydet" düğmesine basılır. Şekil 2.35'da gösterilen örnekte, birim olarak "volt" kullanılan ve yalnızca analog sensörler için çalışabilen "ADS1115 VOLT" fonksiyonunu kullanan bir veri türü belirlenmesi yapılır.



ADD SENSOR DATA TYPE

Data Type Name: Volt

Sensor Type: Analog

Sensor Data Unit Type: volt

Has Calculator Function? Yes No

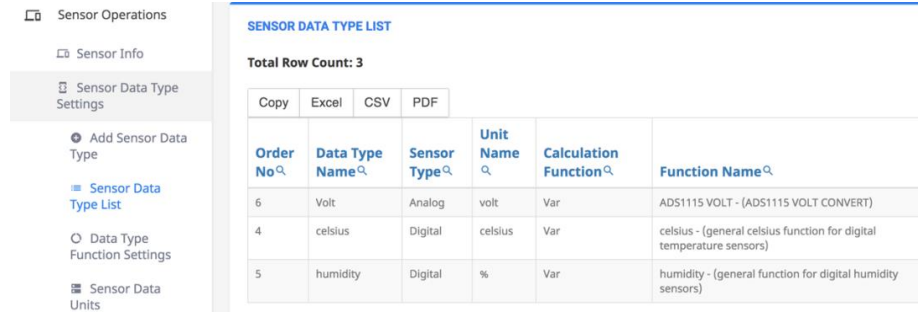
Calculation Function: ADS1115 VOLT - (ADS1115 VOLT CONVERT)

Save

Şekil 2.35: Sensör veri türü ekleme sayfası

2.8.9 Sensör Veri Türü Listeleme Sayfası

Bu sayfada, Şekil 2.35'da görülebilecek örnekte olduğu gibi, "Sensör Veri Türü Ekle" sayfasında (Şekil 2.36) sisteme tanımlanan fonksiyon tanımları listelenir. Bu sayfadan kullanıcı ayrıca bu verileri PDF, EXCEL ve CSV dosya formatlarına dışa aktarabilir.



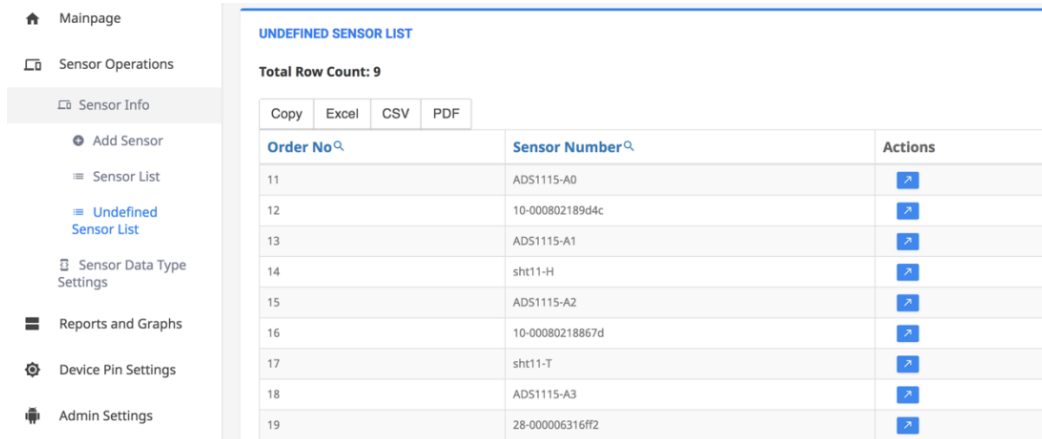
Order No	Data Type Name	Sensor Type	Unit Name	Calculation Function	Function Name
6	Volt	Analog	volt	Var	ADS1115 VOLT - (ADS1115 VOLT CONVERT)
4	celsius	Digital	celsius	Var	celsius - (general celsius function for digital temperature sensors)
5	humidity	Digital	%	Var	humidity - (general function for digital humidity sensors)

Şekil 2.36: Sensör veri türü listeleme sayfası

2.8.10 Tanımsız sensör listeleme sayfası

Bu sayfada, sisteme eklenmemiş olan sensörler listelenir (Şekil 2.37) ve bu liste PDF, CSV veya Excel formatlarında bilgisayara indirilebilir.

Python betiği aracılığıyla sisteme gönderilen sensör, sisteme eklenmemişse, sensörün adı burada kaydedilir. Eklenmesi gereken sensör "Eylemler" sırasındaki düğmeye tıklayarak seçilirse, sistem kullanıcıyı otomatik olarak tanımsız sensör ekleme sayfasına yönlendirir, bu Şekil 2.38'te gösterildiği gibi, seçilen sensörün sisteme eklenmesine izin verir.



Order No	Sensor Number	Actions
11	ADS1115-A0	[Action]
12	10-000802189d4c	[Action]
13	ADS1115-A1	[Action]
14	sht11-H	[Action]
15	ADS1115-A2	[Action]
16	10-00080218867d	[Action]
17	sht11-T	[Action]
18	ADS1115-A3	[Action]
19	28-000006316ff2	[Action]

Şekil 2.37: Tanımsız sensör listeleme sayfası

Tanımsız sensör ekleme sayfasındaki "Sensör Numarası / ID" alanı, Şekil 2.34'de seçilen sensöre göre otomatik olarak doldurulur. Örneğin, "ADS1115-A0" sensörü seçildiyse, sensör türü Analog veri taşıdığı için "Analog" olarak seçilir. Bu seçim yapıldıktan sonra, ADS1115'ten voltaj almak için daha önce Şekil 2.32'da oluşturulan "Sensör Veri Türü" seçilir. Daha sonra "Sensör Adı" alanı doldurulmalıdır. Şekil 2.35'te "AN0" olarak belirtilmiştir. Sonraki iki bölümde, bu sensörden alınan değer "3.2"den düşük olması normal olarak kabul edilir, yani sensör değeri "3.2"yi aşarsa, ana ekranda bir alarm gösterilecektir.

The screenshot shows a web interface for adding a sensor. The main content area is titled "ADD UNDEFINED ADS1115-A0 TO SENSORS". It contains several input fields: "Sensor Type" with the value "Analog", "Sensor Data Type" with "Volt - Analog - volt", "Sensor Number / ID" with "ADS1115-A0", "Sensor Name" with "AN0", and "Alarm Rate" with "3.2". Below these fields, there are two radio buttons for "Alarm Rate Control Type?", with "Higher" selected. A blue "Save" button is located at the bottom of the form. On the left side, there is a sidebar menu with items: Mainpage, Sensor Operations, Sensor Info, Add Sensor (selected), Sensor List, Undefined Sensor List, Sensor Data Type Settings, Reports and Graphs, Device Pin Settings, and Admin Settings.

Şekil 2.38: Tanımsız sensör ekleme sayfası

2.8.11 Sensör ekleme sayfası

"Sensörler" sayfasında listelenemeyen bir sensör, henüz sisteme bağlanmadığı için, Şekil 2.39'da gösterilen ekranda manuel olarak eklenebilir. Bu alan doldurulurken, daha sonra kurulacak olan sensörün sensör ID'si bu alanda doğru bir şekilde yazılmalıdır. Aksi takdirde, sensör kurulduğunda, sensör "Tanımsız Sensör Listesi" sayfasında görüntülenir ve sistemde hiçbir veri kaydedilmez. Bu alandaki işlem, Şekil 2.38'de gerçekleştirilecek işlemle aynıdır. İki sayfa arasındaki tek fark, manuel ekleme işlemi sırasında "Sensör Numarası / ID" bölümünün otomatik olarak doldurulmamasıdır.

Şekil 2.39: Sensör ekleme sayfası

2.8.12 Sensör listeleme sayfası

Şekil 2.40'de gösterilen örnekte, sistemde kayıtlı sensörler, tüm temel tanımlamalarıyla birlikte listelenir. Bu listede görünen sensörlerin değeri Veritabanına kaydedilir. İstenen sensör, Eylemler sekmesi altından silinebilir. Bir sensör silindiğinde, o sensör için sensör verileri de silinir. Bu sayfadan kullanıcı ayrıca bu verileri PDF, EXCEL veya CSV dosya formatlarına dışa aktarabilir.

Order No	Sensor Name	Sensor Type	Data Type Name	Unit Name	Actions
5	PROBE1	Digital	celsius	celsius	[Action]
7	PROBE2	Digital	celsius	celsius	[Action]
8	PROBE3	Digital	celsius	celsius	[Action]
9	TEMP-SHT	Digital	celsius	celsius	[Action]
6	Humidity	Digital	humidity	%	[Action]
10	AN0	Analog	Volt	volt	[Action]
11	AN1	Analog	Volt	volt	[Action]
12	AN2	Analog	Volt	volt	[Action]
13	AN3	Analog	Volt	volt	[Action]

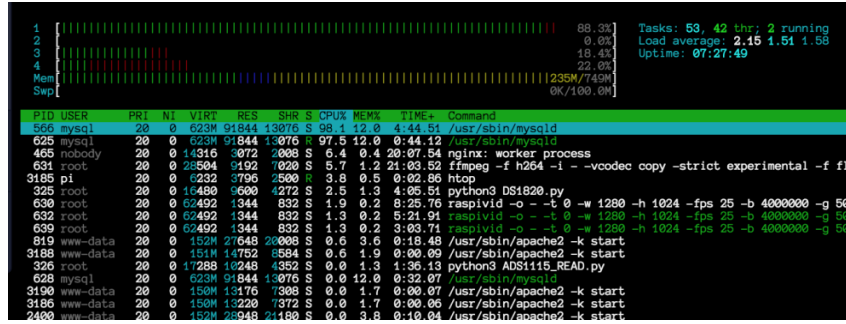
Şekil 2.40: Sensör listeleme sayfası

Bölüm 3

Sonuçlar

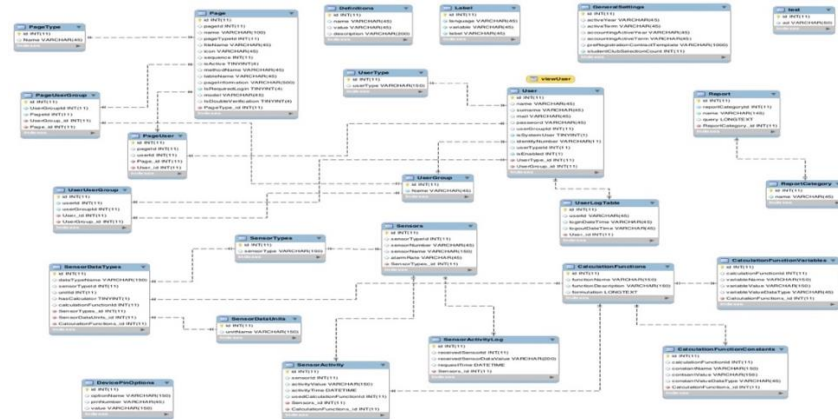
3.1 Sistem Kullanımının Kontrolü

Kullanıcı tarafından fark edilebilecek kadar yavaş olan sistemdeki tek yer, sensör verilerinin listelendiği ve Şekil 2.29'da daha önce görüldüğü gibi grafik olarak gösterildiği sayfadır. Sistem, seçilen tarih aralığında grafiği göstermek veya verilerin listelenmesini istediğinde, veritabanı, içerdiği çok sayıda sensör verisi nedeniyle normalden daha yüksek işlem gücü kullanır. Grafiğin sorgusu devam ederken, sistem kaynaklarının kullanımıyla ilgili "htop" yazılımının çıktısı Şekil 3.1'te gösterilmiştir.



Şekil 3.1: Veritabanı sorgusu sırasında kaynak kullanımı

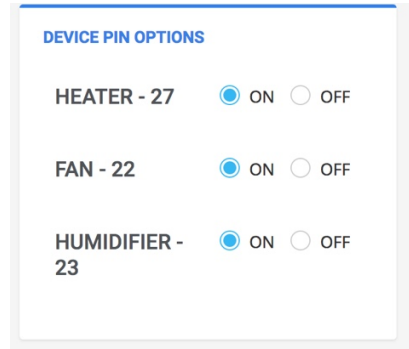
MariaDB veritabanı en verimli şekilde ayarlanmaya çalışılmış ve Şekil 3.2'te gösterildiği gibi düzenlenmiştir.



Şekil 3.2: Veritabanı tablo yapısı

3.2 "Device Pin Options" İşlevselliğinin Kontrolü

Kullanıcı sisteme girdiğinde, ana sayfada sensör verileri görülebilir. Hemen alt tarafta, kameradan canlı görüntüler görülebilir. Sağ taraftaki alanda, kullanıcı Raspberry Pi'deki pinler için tetikleme işlemlerini gerçekleştirebilir. Terminal üzerinden "gpio -g readall" komutu kullanılarak, sistemde daha önce tanımlanmış olan pinlerin BCM sırasına bakarak tetiklemenin başarılı olup olmadığı kontrol edilebilir. Örneğin, Şekil 3.3'te gösterilen durumdan sonra terminalden "gpio -g readall" komutu girildiğinde, sistem tarafından gösterilecek çıktı Şekil 3.4'ya benzer olacaktır. Açılan ekranda, Şekil 3.3'teki pin numaraları BCM sütunu altında bulunur ve "V" sütunu altında karşılık gelen pin durumları kontrol edildiğinde pin durumlarının "1" olarak ayarlandığı görülecektir.



Şekil 3.3: Örnek pin tetikleme

Bu kontrol, "gpio -g read <BCM_Pin_number>" girilerek ve pin durumunun "1" veya "0" olduğu değeri kontrol edilerek de yapılabilir. Şekil 3.3'te gösterilen durum için ekran çıktısı Şekil 3.5'ye benzer olacaktır.

```
pi@raspberrypi:~$ gpio -g readall
```

BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	
		3.3v			1	2		5v			
2	8	SDA.1	ALT0	1	3	4		5v			
3	9	SCL.1	ALT0	1	5	6		0v			
4	7	GPIO.7	IN	1	7	8	1	ALT0	15	14	
		0v			9	10	1	ALT0	16	15	
17	0	GPIO.0	IN	0	11	12	1	IN	1	18	
27	2	GPIO.2	OUT	1	13	14		0v			
22	3	GPIO.3	OUT	1	15	16	1	OUT	4	23	
		3.3v			17	18	0	IN	5	24	
10	12	MOSI	ALT0	0	19	20		0v			
9	13	MISO	ALT0	0	21	22	0	IN	6	25	
11	14	SCLK	ALT0	0	23	24	1	OUT	10	8	
		0v			25	26	1	OUT	11	7	
0	30	SDA.0	IN	1	27	28	1	IN	31	1	
5	21	GPIO.21	IN	1	29	30		0v			
6	22	GPIO.22	IN	1	31	32	0	IN	GPIO.26	26	12
13	23	GPIO.23	IN	0	33	34		0v			
19	24	GPIO.24	IN	0	35	36	0	IN	GPIO.27	27	16
26	25	GPIO.25	IN	0	37	38	0	IN	GPIO.28	28	20
		0v			39	40	0	IN	GPIO.29	29	21

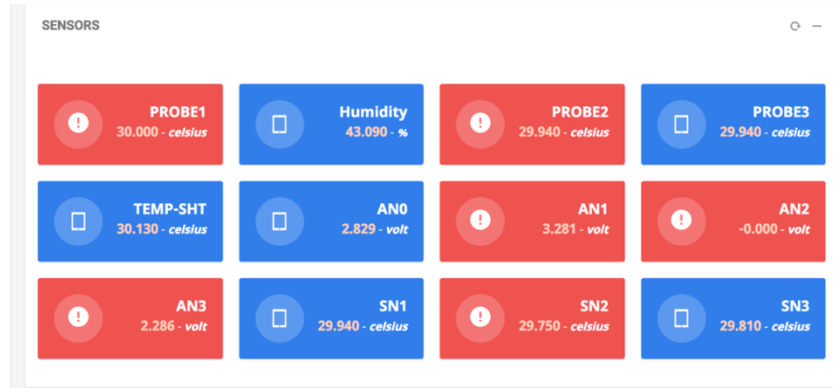
Şekil 3.4: Giriş/Çıkış pin durumları

```
pi@raspberrypi:~ $ gpio -g read 27
1
pi@raspberrypi:~ $ gpio -g read 22
1
pi@raspberrypi:~ $ gpio -g read 23
1
```

Şekil 3.5: Giriş/Çıkış pin durumları

3.3 Sensör Alanının İşlevselliğinin Kontrolü

Analog ve dijital sensörlerin testi gerçekleştirilmiştir. Bu bölümde, sisteme 3 adet daha ds18b20 sensörü eklenmiştir. Ayrıca, ADC IC'yi test etmek için, voltaj bölücü kuralı ile LDR ile birlikte ADS1115'e AN0 pinine bağlanmıştır. AN1, Raspberry Pi'den 3.3 volta, AN2 sistem GND terminaline, AN3 ise analog basınç sensörüne bağlanmıştır. Bu durumda, sistemde AN1 ve AN2 için değerler sırasıyla 3.3 volt ve 0 volt olmalıdır. AN0 ve AN3'ten alınacak değerler, sırasıyla ortam ışığına ve hava basıncına göre değişecektir.



Şekil 3.6: Sensör alanı

Şekil 3.6'de görülebilen sensör alanı örneği parlak bir ortamda test edilmiştir. LDR sensörü GND tarafına bağlandığı için, AN0 pini 3.3 voltan daha düşük bir değere sahiptir. AN1 probu 3.3 volta bağlandığı için, onun değeri 3.281 olup buna çok yakındır. AN2 probunun değeri 0.000'dir çünkü GND noktasına bağlanmıştır. AN3 probu hava basınç sensörüne bağlı olduğu için, hava basıncına göre sensörün voltaj değerini alır ve 2.286 gösterir. Işık almayan AN0 probuna bağlı LDR, Şekil 3.12'de

gösterildiği gibi, AN1'e yakın olan 3.283 değerine sahiptir ve bu sensör için daha önce tanımlanan normal durum koşulunu karşılamadığı için alarm durumunu tetikler.

Şekil 3.6'de gösterilen PROBE1, PROBE2 ve PROBE3 probaları, daha önce eklenen DS18B20 ve DS18S20 sensörleridir. Sistem testi amacıyla eklenen 3 DS18B20 sensörü SN1, SN2 ve SN3 sensörleridir. Yeni eklenen sensörlerle birlikte, sistem tüm tanımlı sensörlerden verileri doğru bir şekilde alabilir ve alınan verileri veritabanında kaydedebilir.

Bölüm 4

Tartışma

Bu çalışmanın amacı, bağlı olduğu sensörlerden veri alacak, gerektiğinde alarm üretecek, hastanın hayati destek cihazlarını kontrol edecek, hastayı uzaktan kontrol etmeye olanak sağlayacak ve tüm bu özellikleri nedeniyle hayati öneme sahip bir hasta bakım odası ve hasta kontrol cihazı geliştirmektir. Geliştirilen prototipin ana kontrolcüsü Raspberry Pi'dir.

Bu projede kullanılan sensörler yüksek hassasiyete sahip, düşük gürültülü ve kolayca bulunabilir olduğundan, Hasta takip sistemine sahip olmayan hastaneler ve hasta yakınları tarafından kolayca bir araya getirilebilir.

Şekil 3.2'te gösterilen veritabanı tablo yapısı incelendiğinde, sistemin ne kadar esnek olduğu ve geliştirmenin ne kadar kolay olduğu daha iyi anlaşılacaktır.

Geliştirilen sistem, ayrı bir çevresel izleme cihazı olarak kullanılacak bir yapıya sahiptir.

Sistem geliştirilirken, 3 kez meydana gelen güç kesintisi nedeniyle SD kartta veri bozulması sorunu yaşandı. Bu arızaların bazılarında, Raspberry Pi, bir veritabanı sorunu oluştuğunda başlatılamadı. Bu nedenle, bir güç kesintisi durumunda, Raspberry Pi'nin sadece verileri için değil aynı zamanda hasta bakım odası ve hasta kontrolü için de sağlıklı olması açısından bir pil ile yedeklenmesi faydalı olacaktır.

Kaynaklar

Karsavuran, N., Özen, M., Eke, R. N., Vural, R., Bulca Acar, A., Aşık, Z., & Koşar, I. (2022). Bir Eğitim ve Araştırma Hastanesi Evde Sağlık Hizmetleri Biriminde Takipli Hastalara Bakım Veren Kişilerde Psikolojik Sıkıntı Görülme Sıklığı. *Journal of Clinical and Medical Education*, 31(4), 254-261
<https://doi.org/10.17942/sted.1059211>

Cafer, Ş., Okyar, Ö., Ateş, M. K., Kıran, M., Taşdemirci, Ç., & Özkan, A. (2020). *Mobil uygulama kontrollü akıllı hasta odası tasarımı [Abstract]*. Kocaeli Üniversitesi.