

İSTANBUL OKAN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI



BİR UZAKTAN PROGRAM
YÜKLEME HİZMETİ TASARIMI
VE UYGULANMASI

YÜKSEK LİSANS TEZİ

NİLGÜN İNCEREİS

tarafından

YÜKSEK LİSANS

derecesi şartını sağlamak için hazırlanmıştır.

Mayıs 2019

Program: Bilgisayar Mühendisliği

BİR UZAKTAN PROGRAM
YÜKLEME HİZMETİ TASARIMI
VE UYGULANMASI

YÜKSEK LİSANS TEZİ

NİLGÜN İNCEREİS

tarafından

İSTANBUL OKAN ÜNİVERSİTESİ
Bilgisayar Mühendisliği Anabilim Dalına
Yüksek Lisans

derecesi şartını sağlamak için sunulmuştur.

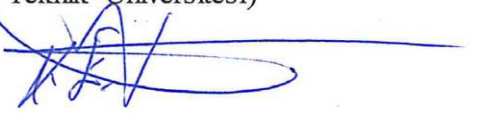


Danışman
Prof. Dr. Bekir Tevfik AKGÜN

Üye
Doç. Dr. Pınar Yıldırım



Üye
Doç. Dr. Erchan APTOULA
(Gebze Teknik Üniversitesi)



Mayıs 2019

Program: Bilgisayar Mühendisliği

ÖZET

BİR UZAKTAN PROGRAM YÜKLEME HİZMETİ TASARIMI VE UYGULANMASI

Yeni teknolojilerden biri olan Nesnelerin İnterneti (IoT) teknolojisi, internete bağlı bilgi alış verişi yapabilen nesnelere bir arada tutan bir ağ ve bu nesnelerin yönetimine katkıda bulunan yazılım destekli bir sistemin oluşumudur.

Bu çalışma, OTA hizmeti olarak adlandırılan bir uzaktan program yükleme hizmeti tasarlamayı ve uygulamayı amaçlamaktadır. Bu çalışmanın sistem tasarımı için, genel IoT sistemleri, yerel sistem, yerel yönetim/uzaktan erişim sistemi, uzaktan yönetim sistemi ve bulut üzerinden dağıtılmış yönetim sistemi olarak incelenmiştir. Uzaktan program yükleme hizmeti sistemi için bir öneri yapılmış ve uygulanmıştır. Sistemi geliştirmek amacıyla senaryo durumları oluşturulmuştur. Bunlar, kullanıcıların sisteme kayıt olma durumlarına göre sisteme veya aygıtta program yükleme, aygıtların sisteme bağlı olma durumlarına göre karar verme yapısı, belli aralıklarla aygıtların sistemde aktif veya pasif olması durumuna göre gözetlenebilmesi, güncelleme için güvenlik önlemleri şeklindedir. Önerilen sistemde, NodeMcu aygıtları, IoT birimleri olarak kullanılmıştır.

Anahtar Kelimeler: Nesnelerin İnterneti, OTA Güncelleme, Bulut, Kablosuz Sensör Ağı, İzleme Sistemi, Hizmetler, Gömülü Sistem, Uzaktan Yazılım Güncelleme.

ABSTRACT

A REMOTE PROGRAM LOADING SERVICE

DESIGN AND IMPLEMENTATION

Internet of Things (IoT) technology, which is one of the new technologies, is a network that holds the objects that can exchange information via the internet and the formation of a software supported system which contributes to the management of these objects.

This study aims to design and implement a remote program installation service called OTA service. For the system design of this study, general IoT systems, local system, local administration/remote access system, remote management system and distributed management system are examined. A recommendation has been made and implemented for the remote program installation service system. Scenario conditions have been created to improve the system. These are in the form the installation of users to the system or device according to their registration status, decision-making structure of devices according to their dependence on the system, monitoring of the devices at certain intervals according to the condition of being active or passive in the system, security measures for updating. In the proposed system, NodeMcu devices are used as IoT units.

Keywords: Internet of Things (IoT), Over The Air (OTA) Update, Cloud, Wireless Sensor Network, Monitoring System, Services, Embedded System, Remote Firmware Update.

TEŐEKKÜR

Tez alıőmam boyunca emeęi geen, bilgi ve tecrübeleriyle bana yol gsteren ve destek veren saygıdeęer danıőman hocam Prof. Dr. Bekir Tevfik AKGÜN' ne tm katkılarından dolayı teőekkr ederim.

Yksek Lisans ęrenimim boyunca her zaman destek olan ve anlayıőlı davranan alıőma arkadaőlarım ve tm ęrenim hayatım boyunca ve her zaman bana destek olan aileme teőekkr ederim.



İÇİNDEKİLER

ÖZET	i
ABSTRACT	ii
TABLO LİSTESİ	vii
ŞEKİL LİSTESİ	viii
KISALTMALAR	xi
I. GİRİŞ	1
1.1. Önceki çalışmalar	1
1.1.1. Program yükleme	5
1.1.2. Otomatik program yükleme	5
1.2. Uzaktan program yükleme hizmeti tasarımı önerisi	6
II. KULLANILAN TEKNOLOJİLER	8
2.1. Nesnelerin interneti teknolojisi	8
2.1.1. IoT elemanları	8
2.1.2. IoT'nin temel özellikleri	10
2.1.3. Nesnelerin interneti mimarisi	10
2.1.4. Nesnelerin internetinde haberleşme	11
2.1.4.1. Makine-makine haberleşmesi	12
2.1.4.2. Makine-bulut haberleşmesi	13
2.1.5. Nesnelerin İnternetinde Güvenlik	14
2.1.5.1 Şifreleme ve sertifikalar	14
2.1.5.2 Yetkilendirme ve doğrulama	15
2.2. Bulut teknolojisi	15

2.2.1 Bulut ortamı	16
2.3. Uzaktan program yükleme örneği.....	18
2.4. ESP8266	19
2.5. OTA kütüphaneleri, fonksiyonları ve işlevleri.....	19
III. BİR UZAKTAN PROGRAM YÜKLEME HİZMETİNİN SİSTEM	
TASARIMI	23
3.1. Genel IoT sistemleri.....	23
3.1.1. Yerel sistem.....	23
3.1.2. Yerel yönetim/uzaktan erişim sistemi	25
3.1.3. Uzaktan yönetim sistemi.....	27
3.1.4. Bulut üzerinden dağıtılmış yönetim sistemi	29
3.2. Önerilen uzaktan program yükleme hizmeti sistemi.....	30
3.2.1. Uzaktan program yükleme sistem bileşenleri	33
3.2.2. IoT yöneticilerinin işleri	34
3.3. Gerçekleştirme aşamaları	34
3.3.1. Tümüleşik yönetim aşaması	34
3.3.2. Uzaktan yönetim sistemi aşaması.....	36
IV. YEREL YÖNETİM/UZAKTAN ERİŞİM SİSTEMİ ÜZERİNE BİR	
UYGULAMA.....	37
4.1. Tümüleşik yönetim ortamı	37
4.2. Tümüleşik yönetimde yapılan işlemler.....	37
4.3. Tümüleşik yönetim veritabanı tasarımı	39
4.4. Tümüleşik yönetimde senaryolar.....	42
4.5. Kullanıcının sisteme kayıt olma durumu.....	43

4.6. Kullanıcı işlemleri	45
4.7. Yönetici işlemleri	47
4.8. Tümleşik yönetimde kullanılan cihazlar ve teknolojiler	50
4.9. Tümleşik yönetimde kullanılan fonksiyonlar ve kütüphaneler	50
4.9.1. Hizmet fiyatını hesaplayan fonksiyon	50
4.9.2. Kullanılan Kütüphaneler	52
4.10. Tümleşik yönetimin çalışma prensibi	53
4.11. Uç birimde OTA sağlanması	54
V. UZAKTAN YÖNETİM SİSTEMİ ÜZERİNE BİR UYGULAMA	56
5.1. Önerilen uzaktan yönetim sisteminin çalışma prensibi	56
5.2. Önerilen uzaktan yönetim sistemindeki kullanıcı ekranı	59
5.3. IoT cihaz yönetim paneli	61
5.4. IoT cihazların durum takibi	61
5.5. Kullanıcıların sisteme dosya yükleyebilme ekranı	62
5.6. Yönetici ekranı	63
5.7. Uzaktan yönetim sisteminin bir web sunucusuna taşınması	66
VI. SONUÇLAR	69
VII. KAYNAKLAR	71

TABLO LİSTESİ

Tablo 2.1 AWS IoT ile Bağlanabilen Hizmetler.....	17
Tablo 2.2 Bulut Bilişiminde Hizmetler	17
Tablo 2.3 Basit ve Gelişmiş Güncelleme İşlemlerini Yapan Fonksiyonlar	20
Tablo 4.1 Tümüleşik Yönetimde Yapılan İşlemler	38
Tablo 4.2 Kullanılan Fonksiyonlar.....	52
Tablo 4.3 Kullanılan Kütüphaneler	52
Tablo 5.1. Tetikleyiciler	57

ŞEKİL LİSTESİ

Şekil 2.1 IoT Elemanları.....	8
Şekil 2.2 IoT'nin Temel Mimarisi	11
Şekil 2.3 Bluetooth protokolü ile direkt bağlantı kuran cihazlar.....	12
Şekil 2.4.a Makine bulut haberleşmesi.....	13
Şekil 2.4.b Ağ geçidi kullanarak makine bulut haberleşmesi	13
Şekil 2.5 AWS IoT Hizmeti.....	16
Şekil 2.6 Arduino Ortamında Kullanılan Dosya Yerleşimi.....	18
Şekil 3.1 Yerel Sistem	24
Şekil 3.2 Yerel Yönetim/Uzaktan Erişim Sistemi	26
Şekil 3.3 Uzaktan Yönetim Sistemi.....	28
Şekil 3.4 Bulut Üzerinden Dağıtılmış Yönetim Sistemi.....	29
Şekil 3.5 Önerilen Uzaktan Program Yükleme Hizmeti	32
Şekil 3.6 Tümüleşik Yönetim.....	35
Şekil 3.7 İki Yönetimli Sistem.....	36
Şekil 4.1. Tümüleşik Yönetimdeki hizmet_db Adlı Veritabanı	39
Şekil 4.2 kullanıcılar Tablosu	39
Şekil 4.3 firmalar Tablosu	40
Şekil 4.4 hizmetler Tablosu	40
Şekil 4.5. Tümüleşik Yönetimdeki iot_db Adlı Veritabanı.....	41
Şekil 4.6. iot_kayıt.....	41
Şekil 4.7. iot_tablosu	42
Şekil 4.8. Sisteme Giriş	43

Şekil 4.9. Kullanıcı İşlemleri	44
Şekil 4.10 Kullanıcının Firma Listesi	44
Şekil 4.11 Hizmet İşlemleri Paneli	44
Şekil 4.12 Firmanın Hizmet Listesi	45
Şekil 4.13 IoT Yönetim Paneli	46
Şekil 4.14. Dosya Gönderme Hizmeti	47
Şekil 4.15 IoT Cihaz Listesi	47
Şekil 4.16 Yönetici İşlemleri	47
Şekil 4.17 Tüm Kullanıcıların Listesi.....	48
Şekil 4.18. Kullanıcı Güncelleme.....	48
Şekil 4.19 Tüm Hizmetleri Listeleme.....	49
Şekil 4.20 Hizmetleri Güncelleme.....	49
Şekil 4.21 Tüm Firmaları Listeleme	49
Şekil 4.22 Uç Birimde OTA Gerçekleştirilmesi.....	55
Şekil 5.1 Önerilen Uzaktan Yönetim Sisteminin Çalışma Prensibi	58
Şekil 5.2 Kullanıcı Kayıt ve Giriş Ekranı	59
Şekil 5.3 Hizmet Alma Sistemi Ödeme Ekranı	59
Şekil 5.4 Hizmet Yönetimine Kayıt Formu	60
Şekil 5.5 Demo Girişi	60
Şekil 5.6 Ödeme Bilgi Giriş Formu.....	61
Şekil 5.7 IoT Cihaz Yönetim Paneli	61
Şekil 5.8.a Cihaz Takip Ekranı 1	62
Şekil 5.8.b Cihaz Takip Ekranı 2.....	62
Şekil 5.9. Dosya Yükleme Ekranı.....	63

Şekil 5.10. Dosya Seçme Durumu.....	63
Şekil 5.11. Yönetici Ekranı.....	64
Şekil 5.12. Sistemdeki Cihazlar.....	64
Şekil 5.13. Versiyon Değişirme Ekranı	64
Şekil 5.14 Sistemi Test Etmek için Farklı Deneme Durumları	65
Şekil 5.15. Panel Girişi	66
Şekil 5.16. Web Sunucu Ortamı	67
Şekil 5.17. Dosyalar.....	67
Şekil 5.18. Veritabanı Ortamı.....	68

KISALTMALAR

IoT: Internet of Things

5G: Fifth Generation

WSN: Wireless Sensor Network

AP: Access Point

IP: Internet Protocol

Wi-Fi: Wireless Fidelity

MD5: Message Digest 5th version

SHA: Secure Hash Algorithm

OTA: Over The Air

SSL: Secure Sockets Layer

TLS: Transport Layer Security

I. GİRİŞ

Günümüzde kullanılan yeni teknolojilerden biri olan Nesnelerin İnterneti (IoT) teknolojisi, internete bağılı bilgi alış verişı yapabilen nesnelere bir arada tutan bir ağı ve bu nesnelere yönetimine katkıda bulunan yazılım destekli bir sistemin oluşumudur. Bu gibi teknolojileri kullanarak bir sistem geliştirmek ve uygulamak bazı kolaylıklar ve avantajlar sağlar.

Bu çalışmanın amacı, bir uzaktan program yükleme hizmeti tasarımı oluşturmak ve uygulamaktır. İlk olarak, önceki yapılan çalışmalardan bahsedilmiştir, program yükleme ve otomatik program yükleme konuları incelenmiştir. Son olarak, uzaktan program yükleme hizmeti tasarımı önerisi açıklanmaktadır.

1.1. Önceki çalışmalar

Nesnelerin İnterneti kavramı ilk olarak, 1991 yılında Cambridge Üniversitesi'ndeki on beş akademisyenin çalıştıkları binadaki bir kahve makinasının boş olup olmadığını uzaktan öğrenmek amacıyla yaptıkları kamera ile bağlantılı bir sistemde bahsedilmektedir [1].

Nesnelerin İnterneti (IoT), kendi aralarında ve çevredekileri (yani akıllı şeyleri veya akıllı nesnelere) algılayabilen, harekete geçiren ve iletişim kurabilen birbirine bağılı şeylerin bir dünyası olarak tanımlanabilir [2]. Buna ek olarak, IoT, süreçleri tetikleyerek ve doğrudan insan müdahalesiyle veya müdahalesi olmadan hizmetler yaratarak gerçek/fiziksel dünya olaylarına özerk bir şekilde bilgi paylaşma ve cevap verme yeteneği sağlar [2].

Nesnelerin interneti teknolojisinde, kablosuz cihazların yaygınlaşmasıyla sistemler oluşturma ve IoT uygulamaları yaratma çalışmaları her geçen gün artmaktadır. Yapılan bir çalışmada, endüstrilerde IoT'nin gelişimini anlamak için, IoT teknolojileri, IoT uygulamaları, IoT çalışmalarında karşılaşılan zorlukların nasıl olduğundan bahsedilir [3].

Bulut hizmetlerine dayanan ticari IoT platformlarının çoğu (AWS, IBM Watson vb.), kullanıcılara çeşitli hizmetler tanıyarak, bazı IoT ürünlerine hızlı ve kolay bir şekilde bütünleşebilen sistem oluşturma ve uygulamalar hazırlamalar konusunda IoT'nin gücünden faydalanmalarını sağlamaktadır. Uygulama pazarı ile son kullanıcılar arasında, etkileşimli bir güven modeline dayalı çalışmalar bulunmaktadır. Örneğin bir çalışmada, davranış temelli tespit aracından, kullanıcıların aygıtlarının gizliliği ve güvenliğinden bahsedilir [4].

Hizmet veya uygulama kalitesinde, güvenilirlik kullanıcılarla ilgili önemli parçalardan biridir. Uygulamanın güven değerlendirilmesi, aygıtlarının güvenliği ile ilgilenen kullanıcılar için gereklidir.

IoT uygulamaları geliştirmek için, endüstri tarafından kullanılan çeşitli modellerin karşılaştırmalı bir analizi yapılarak, modellerin ya da tasarım patentlerinin kullanıldığı tekniklerin karşılaştırılmasından, özgünlüğünden ve eksikliklerinden söz edilen çalışmalar bulunmaktadır [5].

IoT uygulamaları geliştirmek için tasarlanan modellerde, nesnelere, aygıtlara, kablosuz iletişim ortamları, ağ geçidi, bulut, uygulama sunucusu, yazılımlar, uygulamalar, kullanıcılar, yöneticiler kullanılır. Nesnelere, çevredeki verileri toplar ve bunları kablosuz iletişim ortamı ve ağ geçitleri aracılığıyla bir sunucuya gönderir.

Buluttaki veriler, uygulamalar ve hizmetler aracılığıyla depolanır, işlenir veya analiz edilir ve son kullanıcılara sunulur. Bir sistemdeki uç birimlerin çok fazla olmasıyla, kolay bağlantıyı, kontrol sistemini, iletişimi ve aynı zamanda sağlam ve güvenli bağlantıyı zorlaştırmaktadır. Bu nedenle uygun sistem tasarım mimarisinin tasarlanması gerekir. Çoğu IoT mimarisindeki zorluk ve sorunlar, standardizasyonun olmamasından kaynaklanır. Standardizasyon eksikliği ve IoT uygulama geliştirmedeki zorluklar [6] ve [7]'de açıklanmaktadır.

Nesnelerin İnterneti (IoT) platformu (genellikle IoT ara katman yazılımı olarak adlandırılır), makine ve aygıtların bağlanmasını ve ardından makine ve sensör verilerinin toplanmasını, işlenmesini, dönüştürülmesini, düzenlenmesini ve saklanmasını sağlayan bir yazılımdır. Zdravkovic ve diğerleri [8], 16 farklı bulut tabanlı IoT platformunu sunmuşlardır. Zamanında akıllı aygıtlar her yerde bulunur. Onların kontrolü kolay olması gerekirken, birbirleriyle ilişkilendirmek daha zor bir iştir. Buluttaki tüm IoT aygıtlarını kontrol eden esnek bir platform sağladığı söylenen bir çalışmada, senaryoya çözüm önerisi olarak bir olaya cevap verme sürecini değiştirme esnekliği sağlanmaktadır. Yapılan bu çalışmada, bu sistem yerel makine üzerinde çalıştırılmış ancak IBM Bluemix platformu yardımıyla da buluta getirilebileceğinden bahsedilmiştir [9]. Uzaktan kontrol edilebilen Wi-Fi, 3G veya 4G işlevselliğine sahip akıllı aygıtların sayısı artıyor. IoT, bu tür aygıtların içindeki birimi, onların uzaktan bağlanmasını ve kontrol edilmesini sağlar [10]. Buluttaki aygıtların sayısı arttıkça, bunların yazılımının daha sık güncellenmesi gerekir. Bu işlemler sonraki paragrafta ele alınmaktadır.

Uzaktan güncellenmenin uygulanmasıyla ilgili bir çalışmada [11], sistem programlarının seçici olarak yüklenmesi yapılmıştır. Yeni teknolojilerin oluşması ve bu teknolojilerin hızlı bir şekilde ilerlemesiyle, IoT' nin ortaya çıkışı günlük yaşamın her alanında hayati bir rol oynamaktadır. Ev otomasyonu ve güvenlik alanındaki IoT çalışmaları, IoT' nin bazı özellikleri sayesinde yaşam tarzını da kolaylaştırır. Wi-Fi teknolojisini kullanarak, kablosuz ev otomasyonu kontrolünün uygulanması ve tasarımı için IoT, tümleşik, korumalı ve güvenli sisteme sahip aygıtları yönetir. IoT ve mikro denetleyici tabanlı bir NodeMcu (Esp8266) ve bir Android mobil uygulaması kullanılarak güvenlik ve otomasyonda nasıl uygulanabileceği hakkında ayrıntılı bilgi veren çalışmalar bulunmaktadır [12].

Amazon Web servisleri (AWS), bir çevrimiçi bulut bilişim hizmetleri sunar. AWS, M2M bağlantılarının temeli olan birkaç MQTT (Message Queue Telemetry Transport) aracısı oluşturmaya yardımcı olur [13]. AWS, bulut IoT için ölçeklenebilir ve genişletilebilir bir alt yapı sağlar. Hem sanal hem de fiziksel donanım ile etkileşime giriyorlar [12]. Böylece AWS, en uygun maliyetli bulut hizmetlerinden biri olarak bulundu [14]. Dell [15] tarafından kullanılan mimaride, ağın kenarındaki verileri güvence altına almak için uygulamalara özel ağ geçidi kullanılır. Toplanan veriler üzerinde analitik işlemler yaparlar ve buluta bilgi gönderirler. Teknolojideki yeni gelişmelerle çok sayıda algılayıcı aygıt, WSN (Wireless Sensor Networks- Kablosuz Algılayıcı Ağlar) sistemleri aracılığıyla internet ortamına entegre edilebilmektedir [16].

1.1.1. Program yükleme

Program yükleme, bir bilgisayara kablo ile bağlı bir gömülü sistem bileşenine program dosyalarının aktarılması ile onun programlanmasıdır.

1.1.2. Otomatik program yükleme

Otomatik program yükleme veya başka bir deyişle kablosuz güncelleme (Over the Air-OTA), bir kablo bağlantısı gerektirmeden bir birime program yüklenmesi işlemidir. Bu işlevsellik, birimlere sınırlı bir erişimin olması veya fiziksel erişim olmaması durumunda son derece yararlı olur.

Temelde OTA 2 farklı şekilde yapılabilir. Geliştirme ortamı yazılımıyla (IDE) veya dosya aktarım komutu ile http protokolü kullanan diğer uygulamalarla yapılır. Günümüzde yaygın olarak kullanılan Arduino sistemine yönelik ESP aygıtlarını içeren uygulamalar mevcuttur [17, 18].

Arduino IDE seçeneği öncelikle yazılım geliştirme aşamasına yöneliktir. Diğer seçenekler, aygıtta bir web tarayıcısıyla uygulama güncellemeleri veya bir http sunucusu kullanarak otomatik olarak uygulama güncellemeleri için kullanılır. Bir aygıtta, ilk defa programın yüklenmesinin kablolu bağlantı ile yapılması gerekir. OTA doğru bir şekilde yüklenirse, sonraki tüm yüklemeler kablosuz ortam üzerinden OTA ile de yapılabilir. Bu uygulamalarda, OTA işleminde kısıtlı bir güvenlik vardır. Güncellemelere yalnızca güvenilir kaynaklardan izin verildiğinden emin olmak, geliştiriciye bağlı bir husustur. Güncelleme tamamlandığında, aygıt yeniden başlatılır ve yeni kod yürütülür. Bu çalışma şekli daha çok yeni sürümler deneme gibi aygıt işlemlerini habersiz yarıda kesmenin önemli bir zararın olmadığı durumlara uygun

olacaktır. Ancak sahada çalışan ve kendisine atanmış görevleri kesintisiz ya da veri kayıpsız veya kontrol edilen sistemlerin beklenmedik davranışlar göstermemesi gereken durumlarda; aygıtın işlemlerini güvenli bir şekilde ara vermesini sağlamak gerekir. Aygıt üzerinde çalışan bir uygulamanın korumalı bir şekilde ara verilmesi ve yeniden başlatılması ayrıca tasarlanmalıdır.

OTA güncellemelerini destekleyen iki yazılım yapılandırması vardır [18]. Bunlar aşağıda sıralanmıştır.

- **Classic OTA:** Arduino IDE 1.6.5 ve 1.6.5-947-g39819f0 (23 Temmuz 2015'tir) veya 1.6.5-1160-gef26c5f (30 Eylül 2015'in) ilk OTA uygulanmasını sağlayan platform paketinin sürümüdür ve ArduinoOTA kütüphanesi olmadan destek için yazılmıştır. Bu özel yapılandırmayı Arduino IDE'de yapılandırmak daha kolaydır ve bu nedenle az tecrübeli kullanıcılar için önerilir [18].
- **ArduinoOTA:** Arduino-PR-4107-BUILD-421 ve ArduinoOTA kütüphanesini içeren platform paketinin son sürümüdür. Bu yapılandırmada Arduino IDE ön hazırlığı bulunur ve daha tecrübeli kullanıcılar için tasarlanmıştır [18].

1.2. Uzaktan program yükleme hizmeti tasarımı önerisi

Önerilen uzaktan program yükleme hizmeti sistemi, sahadaki IoT uç birimlere, yeni program yükleme işlemlerini firma adına güvenli ve korumalı bir şekilde yerine getiren bir sistemdir.

Amacımız, OTA işlemleri yapmak isteyenler için bir hizmet sağlamaktır. Bu çalışmada, uzaktan program yükleme hizmeti için önerilen sistem, 2 aşamada gerçekleştirilmiştir. İlk olarak, sistem yazılımlarını denemek için tümleşik yönetim

sistemi gerekleřtirilmiřtir. Sonra tmleřik sistem, uzaktan ynetim sistemine dađıtılmıřtır. Kullanıcılar ile hizmet ynetimi arasındaki iletiřim, gvenlikli bir iletiřim ortamı zerinden sađlanmaktadır. Benzer řekilde hizmette ynetim yazılımları ile u birimler arasında da gvenlikli bir haberleřme ortamı sađlanmıřtır. Kullanıcılar kendi cihazlarındaki yeni bir uygulama programı srmn hizmet ynetimine aktararak uzaktan u birimler zerine yklenmektedir.



II. KULLANILAN TEKNOLOJİLER

Bu bölümde, bu çalışma için gerekli olan nesnelerin internet teknolojisinden, bulut teknolojisinden, uzaktan program yüklemeyen, ESP8266 ve kütüphanelerinden bahsedilmiştir.

2.1. Nesnelerin interneti teknolojisi

Nesnelerin İnterneti teknolojisi, adreslenebilir nesnelerin önceden belirlenmiş ölçütler ve kurallar çerçevesinde veri toplama, işleme, analiz, görüntüleme ve depolama gibi işlemlerin yapıldıktan sonra farklı nesneler ile paylaşılmasına imkân veren bir ağ sisteminin oluşumudur [19].

Nesnelerin İnterneti Teknolojisi, IoT'den gelen verilerin kısıtlamalarla gerçek zamanlı olarak programlanması gerekir. Yapılan bir çalışmada, IoT'nin talep ve verileri, düşük ve yüksek önceliğe göre bölünmüştür [20].

2.1.1. IoT elemanları

Daha iyi bir sistem tasarlayabilmek için, IoT yapı taşlarını anlamak ve incelemek gerekir. Şekil 2.1'de IoT'nin kimlik, algılama, iletişim, hesaplama, hizmetler ve semantik elemanları olduğu görülmektedir [21].



Şekil 2.1 IoT Elemanları

IoT için Kimlik, IoT aygıtlarının istekleriyle hizmetlerini eşleştirmede ve IoT aygıtlarını isimlendirmede önemlidir. Ayrıca, IoT aygıtlarını adresleme, nesne kimliği ile nesne arasında ayırım yapmak için çok önemlidir. Adresleme yöntemleri, genel olarak benzersiz olmadığından adresleme zorunludur, bu nedenle adresleme, nesnelerin benzersiz olarak tanımlanmasına yardımcı eder [21].

IoT için algılama, ağ içindeki veritabanına veya buluttaki veri deposuna ilgili nesnelere veri toplanması anlamına gelir. Toplanan veriler, gerekli hizmetlere dayalı olarak belirli eylemler yapmak için analiz edilir. IoT sensörler, akıllı sensörler, aktüatörler ve giyilebilir algılama cihazları olabilir. IoT ürünlerini gerçekleştirmek için, Arduino Yun, Raspberry Pi, BeagleBone Black vb. aygıtlar kullanılabilir. Bu tür aygıtlar, müşteriler tarafından istenen verileri sağlamak için bir merkezi yönetim ortamına bağlanır [21].

IoT iletişim teknolojileri, belirli akıllı hizmetleri vermek için heterojen nesnelere birbirine bağlar. Tipik olarak, IoT düğümleri kayıplı ve gürültülü iletişim bağlantıları varlığında düşük güç kullanarak çalışmalıdır. IoT için kullanılan haberleşme protokollerine örnek olarak Wi-Fi, Bluetooth, IEEE 802.15.4 vb. gibi verilebilir. Ayrıca, bazı özel iletişim teknolojilerinden RFID, yakın alan iletişimi (NFC) veya ultra geniş bant genişliği (UWB) kullanılabilir [21].

İşleme birimleri (örneğin mikrodenetleyiciler, mikroişlemciler gibi) ve yazılım uygulamaları, IoT'nin hesaplama yeteneğini ve beynini temsil eder. Arduino gibi IoT uygulamalarını çalıştırmak için çeşitli donanım platformları geliştirilmiştir. [21].

Genel olarak, IoT hizmetleri dört sınıfta kategorize edilebilir [22]: Kimlikle ilgili hizmetler, bilgi toplama hizmetleri, işbirliğine dayalı hizmetler ve her yerde birden bulunan hizmetler [21].

IoT'deki anlambilim ise, bilgiyi elde etme kabiliyetini ifade eder [21].

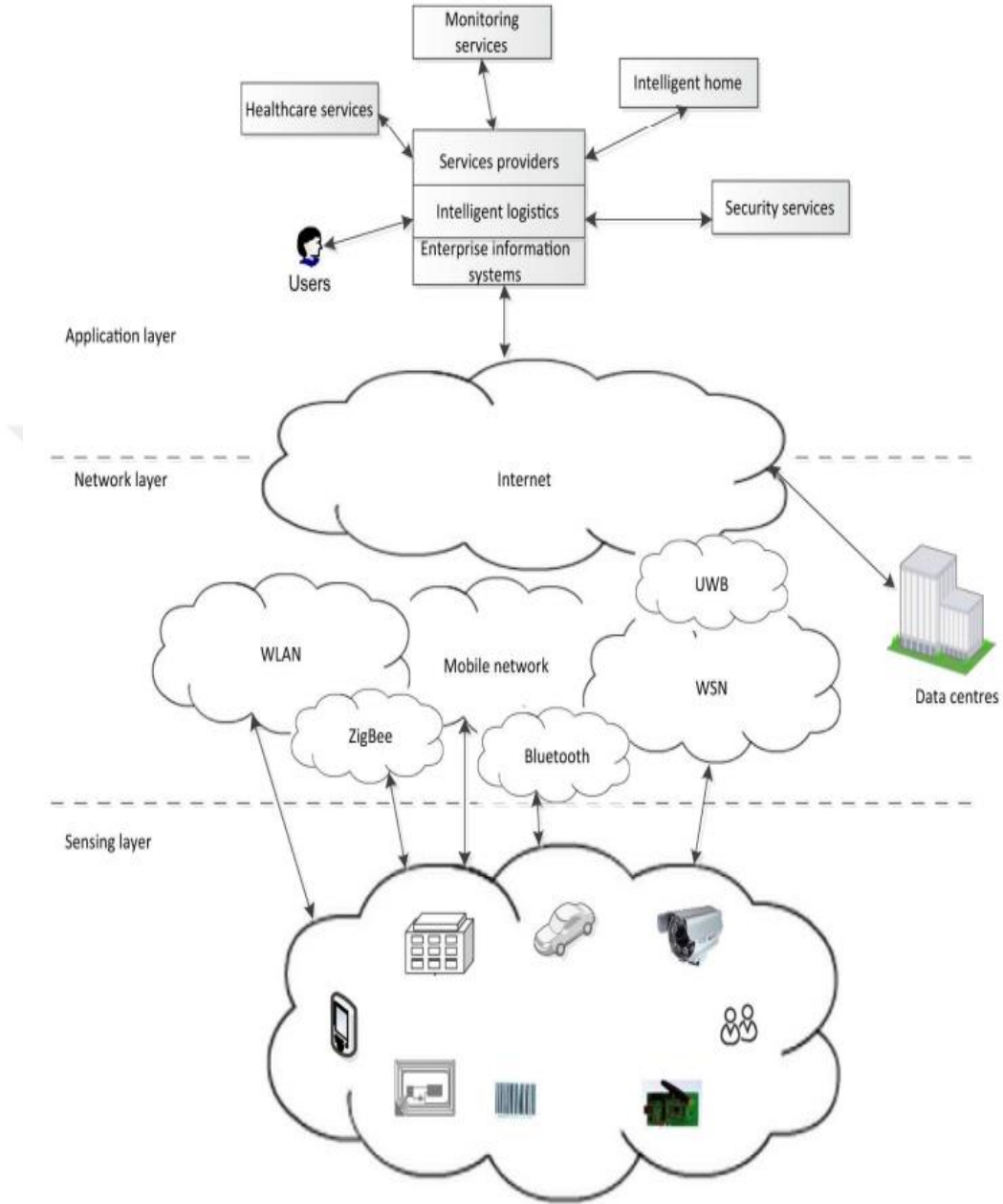
2.1.2. IoT'nin temel özellikleri

IoT'nin temel özellikleri aşağıdaki şekilde verilmektedir [1].

- Nesneler, gerçek fiziksel dünya nesnesi ya da sanal nesne olabilir.
- Nesnelerin, güvenli ortamda ticari adları vardır ve bu ticari ad tarafından otomatik olarak tanımlanır.
- Nesneler, IoT alt yapısıyla ve diğer nesnelerle iletişim kurmak için bir anlaşma kullanır.
- Nesneler, gerçek fiziksel dünya ve dijital sanal dünya arasında bilgi alış verişinde bulunur.

2.1.3. Nesnelerin interneti mimarisi

Şekil 2.2, hizmet yönelimli IoT'nin yeni özelliklerini dikkate alan temel bir IoT mimarisini göstermektedir. Bu üç katlı mimari, uygulama, ağ ve algılama katmanlarından oluşmaktadır. Kullanıcılar çeşitli hizmetler vasıtasıyla uygulama katmanından isteklerini internet iletişim ortamı üzerinden ağ katmanında bildirirler. Ağ katmanından da algılama katmanına geçerken çeşitli kablosuz iletişim ortamları kullanılabileceği [23] ve [24]'te görülmektedir.



Şekil 2.2 IoT'nin Temel Mimarisi

2.1.4. Nesnelerin internetinde haberleşme

Nesnelerin İnterneti, İngilizce ifadesiyle “Things“, Türkçe ifadesiyle “Şeylerin“ özne, internet ve ağ sisteminin de nesne ve yüklemine oluşturduğu bir modeli anlatmaktadır. Bu modelde nesneler, birbirleriyle veya bilgisayar

programlarıyla iletişime geçmektedir. Bu bağlamda makine-makine haberleşmesi ve makine-bulut haberleşmesi Nesnelerin İnternetinde kullanılan iki temel haberleşme modelidir [25].

2.1.4.1. Makine-makine haberleşmesi

Makine-Makine Haberleşmesi (M2M), arada herhangi bir yardımcı donanım olmadan nesnelerin birbirleriyle bağlantı kurdukları ve veri transferi yaptıkları bir modeldir [26]. Şekil 2.3'te Bluetooth protokolü ile doğrudan bağlantı kuran bazı aygıtlar gösterilmiştir. Bu modelde birebir iletişim sadece internet ortamında IP tabanlı olarak gerçekleşmeyebilir [25].

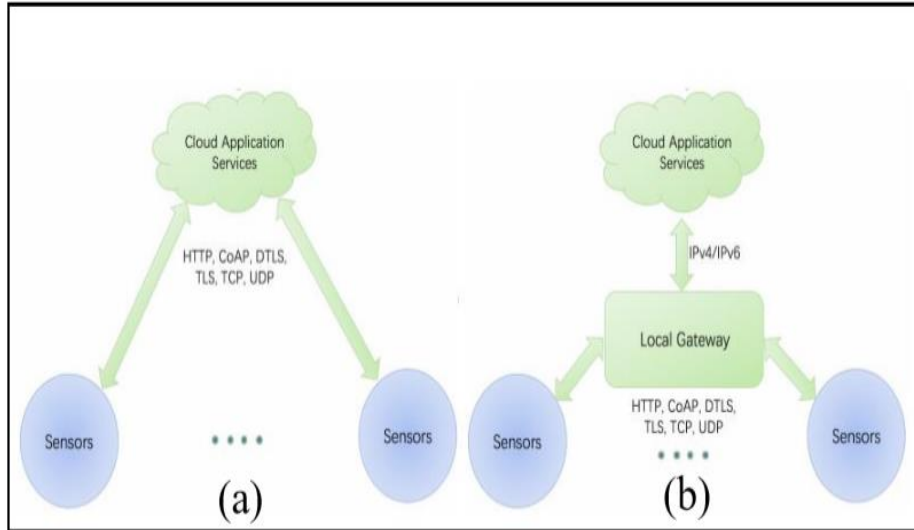


Şekil 2.3 Bluetooth protokolü ile direkt bağlantı kuran cihazlar

Kullanıcı tarafından bakıldığında makine-makine haberleşmesinin uyumluluk sorunu vardır. Aygıtların haberleşmesini sağlayan pek çok protokol birbirini desteklememektedir. Örneğin Z-Wave protokolünü kullanan bir akıllı ev sistemi Zigbee protokolünü kullanan cihaz ile iletişim kuramaz [27]. Bu durumda ara sistemler kullanılarak uyum sorunu giderilmektedir [25].

2.1.4.2. Makine-bulut haberleşmesi

Makine-bulut haberleşmesinde nesnelere, bulut ortamında servis sağlayıcı olarak yer alan uygulama ve programlar ile iletişime geçerek işlem veya veri depolama, veri indirme vb. hizmetlerden faydalanırlar [26]. Bu modelde İnternete bağlantı geleneksel kablolu veya Wi-Fi protokolü ile sağlanırken, veri işleme ve transfer işlemlerinde ağ sisteminin diğer eleman, yöntem ve protokolleri kullanılmaktadır. Doğrudan makine bulut haberleşmesi ve ağ geçidi kullanarak makine bulut haberleşmesi Şekil 2.4.a ve Şekil 2.4.b' de gösterilmiştir [28].



Şekil 2.4.a Makine bulut haberleşmesi

Şekil 2.4.b Ağ geçidi kullanarak makine bulut haberleşmesi

2.1.5. Nesnelerin İnternetinde Güvenlik

Nesnelerin internetinde güvenliği sağlamak için, şifreleme, sertifikalar, yetkilendirme veya kimlik doğrulama gibi çözümler bulunmaktadır.

2.1.5.1 Şifreleme ve sertifikalar

Nesnelerin internetinde güvenliği sağlamak için güvenli yuva katmanı (SSL) veya ulaşım katmanı güvenliği (TLS) gibi şifreleme katmanları bulunmaktadır [29].

SSL, kullanılan sunucu ve internet tarayıcısı arasındaki işlemleri şifreleyen ve üçüncü şahıslar tarafından yapılacak olan kötü niyetli kullanımın önüne geçen, standart bir algoritma üzerine kurulmaktadır. SSL sertifikası ile birlikte, sahip olunan internet sitesinin başında, https protokolü bulunmaktadır. Bu durum, güvenilir bir hizmet sunulduğu anlamına gelmektedir. SSL'in önemli diğer özellikleri ise şunlardır: İletilen dokümanların, özellikle de site üzerinden yapılan alışverişlerin, tarih ve saatlerini kaydederek bu işlemi doğrular ve böylelikle bir doküman arşivinin oluşturulmasını sağlamış olur. Bunun dışında internet üzerinden yapılan işlemlerde veriyi gönderen kişinin ve alan tarafın güvenilir olduğundan emin olduktan sonra işlem yapılmasına izin verir.

Ulaşım katmanı güvenliği ise, uçtan uca güvenliği sağlamak amacıyla AES şifreleme ile eşleşen geniş alan ağı (WAN) üzerinden şifreli veri gönderen bir standart iletişim katmanıdır. TLS/SSL, şifreli veriyi cihazdan cihaza aktarırken veri akışına yüksek düzeyde koruma sağlar [29].

2.1.5.2 Yetkilendirme ve doğrulama

Sisteme, programa, hizmete veya ağı erişim hakkının kullanıcı ya da kullanıcılara verilmesine yetkilendirme (Authentication), sunucu, anahtar veya yönlendirici kullanımlarında aygıt ya da kullanıcının kimliğinin onaylanmasına ise kimlik doğrulama (Authorization) denir.

Nesnelerin interneti sisteminde, çoğu cihazın güvenliği için kimlik doğrulama ve yetkilendirme kritik bir aşamadır. Cihazlar uygulamalara, hizmetlere ve veri yollarına erişmeden önce kendi kimliklerini kurmak zorundadır. Fakat çoğu cihaz basit parola ile ya da ilk parolayı değiştirmeden kullanarak kimlik doğrulama yaptığından başarısızlığa uğramaktadır [29].

İki faktörlü kimlik doğrulama (2FA) yöntemi, iki farklı adımda sisteme girmeye izin veren bir güvenlik yöntemidir. Bir IoT platformunda güvenliğinin sağlanması için güçlü bir parola ile giriş, sertifikalar ya da iki faktörlü kimlik doğrulama yöntemleri gibi daha kalıcı çözümlerin olması gerekir [29].

2.2. Bulut teknolojisi

Bulut bilişim teknolojisinde büyük veriler, internete depolanabilir ve istenirse bu verilere erişilebilir. Bulut teknolojisinin dört farklı çeşidi, aşağıdaki şekilde verilmektedir [30].

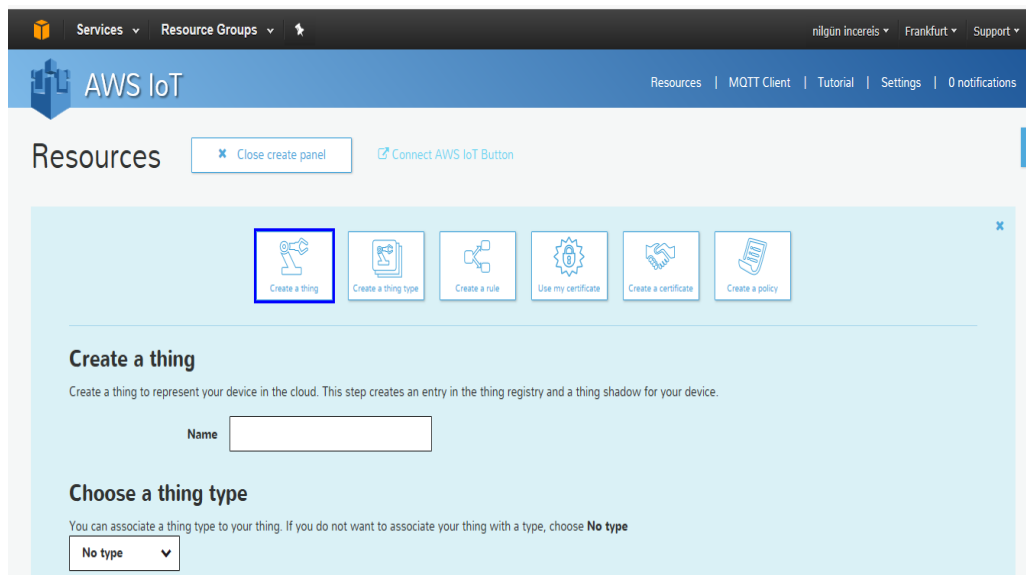
- **Public Cloud (Genel Bulut):** İnternet üzerindeki sunucular ile kurulan bir bulut teknolojisidir.
- **Private Cloud (Özel Bulut):** Bilgileri önemli olan büyük şirketlerin tercih ettiği bir bulut teknolojisidir. Tüm bilgiler kurucunun elinin altındadır ve erişim güvenliği ve gizliliği yüksektir.

- **Hybrid Cloud (Melez Bulut):** Public ve Private Cloud'un birleşiminden ortaya çıkan bulut teknolojisidir.
- **Community Cloud (Topluluk Bulut):** Birkaç şirket ile ortak kullanılan hizmetleri barındıran bulut teknolojisidir. Topluluk üyeleri uygulama ve verilere erişebilmektedir.

2.2.1 Bulut ortamı

Nesnelerin interneti konusunda kullanılacak birçok bulut ortamları bulunmaktadır. Bu ortamlar, Amazon Web Hizmetleri (AWS) IoT, Microsoft Azure IoT, IBM Watson, ThingSpeak, Cisco IoT Cloud Connect örnek olarak verilebilir.

AWS'de, kullanılacak birçok farklı hizmet bulunmaktadır. **Şekil 2.5**'te gösterilen AWS IoT hizmeti, sensörler, gömülü cihazlar veya akıllı aygıtlar gibi internet bağlantısı olan nesnelere arasında çift yönlü iletişimi, güvenliği ve bulut ortamında çalışmayı sağlayan bir hizmettir.



Şekil 2.5 AWS IoT Hizmeti

Aşağıdaki **Tablo 2.1**'de AWS IoT ile ilişkili AWS hizmetleri ve açıklamaları verilmektedir [31].

Hizmet Adı	Açıklaması
Amazon Simple Storage Service (Amazon S3)	AWS bulut içinde ölçeklenebilir depolama sağlar.
Amazon DynamoDB	NoSQL veritabanlarını yönetmeyi sağlar.
Amazon Kinesis	Büyük ölçekte gerçek zamanlı verilerin akışını işlemeyi sağlar.
AWS Lambda	Olaylara cevap vermede Amazon EC2'den kodu sanal sunucular üzerinde çalıştırır.
Amazon Simple Notification Service (Amazon SNS)	Bildirimleri gönderir ve alır.
Amazon Simple Queue Service (Amazon SQS)	Uygulamalar tarafından alınmasını sağlayan kuyrukta veri depolar.

Tablo 2.1 AWS IoT ile Bağlanabilen Hizmetler

Bulut bilişiminde bulunan hizmetler, PaaS, IaaS, SaaS, MaaS **Tablo 2.2**'de verilmektedir [32].

Hizmet Tipi	İngilizce Açılımı	Türkçe Açılımı
PaaS	Platform as a Service	Platform Hizmeti
IaaS	Infrastructure as a Service	Altyapı Hizmeti
SaaS	Software as a Service	Yazılım Hizmeti
MaaS	Manufacture as a Service	Üretim Hizmeti

Tablo 2.2 Bulut Bilişiminde Hizmetler

Bulut üzerinde nesnelerin interneti uygulaması yapmak için kullanılan ortamın sunmuş olduğu hizmetlere mutlaka bakılması gerekmektedir.

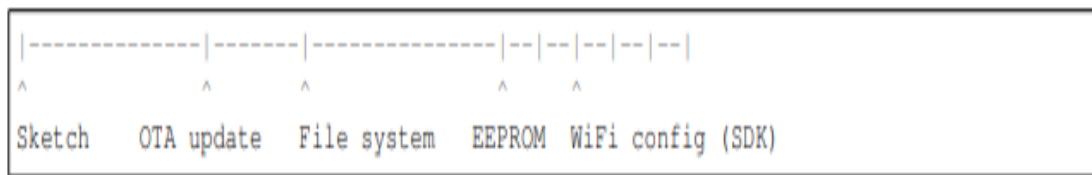
2.3. Uzaktan program yükleme örneği

ESP8266 serisi tüm devre kartları, Wi-Fi ile internete çıkabilen hem boyut olarak küçük hem de fiyat olarak ucuz olduğundan yaygın kullanılmaktadır. ESP8266'nın üretiliş amacı, öncelikle Arduino'yu Wi-Fi üzerinden internete bağlamaktır. Xbee, Ethernet kartı gibi pahalı parçalar kullanmak yerine daha ucuz olan bu birimi kullanmak sistem maliyetini azaltmaktadır.

ESP8266 kütüphanelerini, Arduino IDE üzerinden yükleyerek, NodeMCU kartlar programlanır. Bir kez programlandıktan sonra, internet üzerinden OTA ile tekrar programlanabilir.

Dosya sistemi; yapılandırma dosyalarını, Web sunucusu içeriğini ve diğer dosya verilerini depolayarak yönetir.

Aşağıdaki diyagram Arduino ortamında kullanılan dosya yerleşimini göstermektedir [17].



Şekil 2.6 Arduino Ortamında Kullanılan Dosya Yerleşimi

2.4. ESP8266

ESP8266, seri haberleşme ile kablosuz internet ağına bağlanabilen bir birimdir. Ucuz ve kolay kullanıma sahip olmasından dolayı nesnelerin interneti projelerinde yaygın olarak kullanılmaktadır.

Arduino'yu internete bağlamanın en ucuz yolu ESP8266 Wi-Fi birimi kullanmaktır. Bu birim ortamda bulunan kablosuz ağlara bağlanabileceği gibi, kendi internet ağını yayarak diğer aygıtların bu ağa bağlanabilmesine de olanak sağlamaktadır.

2.5. OTA kütüphaneleri, fonksiyonları ve işlevleri

ESP8266httpUpdate kütüphanesinde, ESPhttpUpdate sınıfı bulunmaktadır. Bu sınıf güncellemeleri kontrol edebilir ve HTTP web sunucusundan bir ikili dosya indirebilir. Güncellemeler, ağdaki ya da internetteki herhangi bir IP ya da alan adından indirilebilmektedir. Bu kütüphanede basit ve gelişmiş güncelleme işlemlerini yapan fonksiyonlar bulunmaktadır. Basit güncelleyici, işlev her çağrıldığında dosyayı indirir. Gelişmiş güncelleyici ise, güncelleme işlevi için sunucudaki bir komut dosyasını göstermesi mümkündür. Sunucu tarafı komut dosyası, güncelleme yapılıp yapılmayacağını kontrol etmektedir.

Fonksiyon	Parametreler	Güncelleme İşlemi
ESP8266HTTPUpdate::update(const char * url, const char * current_version, const char * httpsFingerprint)	<ul style="list-style-type: none"> • url • current_version • httpsFingerprint 	Basit
ESP8266HTTPUpdate::update(const char * host, uint16_t port, const char * url, const char * current_version, bool https, const char * httpsFingerprint)	<ul style="list-style-type: none"> • host • port • url • current_version • httpsFingerprint 	Gelişmiş
ESP8266HTTPUpdate::update(String host, uint16_t port, String url, String current_version, bool https, String httpsFingerprint)	<ul style="list-style-type: none"> • host • port • url • current_version • httpsFingerprint 	Gelişmiş

Tablo 2.3 Basit ve Gelişmiş Güncelleme İşlemlerini Yapan Fonksiyonlar

Sunucu isteği işlemede, basit güncelleyici için sunucunun sadece güncelleme için ikili dosyayı vermesi gerekir. Gelişmiş güncelleme yönetimi için bir betiğin (örneğin bir PHP betiği) sunucu tarafında çalışması gerekir. Her güncelleme talebinde ESP, HTTP başlıklarındaki bazı bilgileri sunucuya gönderir.

Örnek başlık verileri aşağıda verilmiştir.

[HTTP_USER_AGENT] => ESP8266-http-Update

[HTTP_X_ESP8266_STA_MAC] => 18:FE:AA:AA:AA:AA

[HTTP_X_ESP8266_AP_MAC] => 1A:FE:AA:AA:AA:AA

[HTTP_X_ESP8266_FREE_SPACE] => 671744

[HTTP_X_ESP8266_SKETCH_SIZE] => 373940

[HTTP_X_ESP8266_CHIP_SIZE] => 524288

[HTTP_X_ESP8266_SDK_VERSION] => 1.3.0

[HTTP_X_ESP8266_VERSION] => DOOR-7-g14f53a19

Bu bilgi ile komut dosyası bir güncelleme gerekip gerekmediğini kontrol edebilir.

```
t_httpUpdate_return ret = ESPhttpUpdate.update("192.168.137.22", 80,
"/esp/update/arduino.php", "optional current version string here");
switch(ret) {
case HTTP_UPDATE_FAILED:
    Serial.println("[update] Update failed.");
    break;
case HTTP_UPDATE_NO_UPDATES:
    Serial.println("[update] Update no Update.");
    break;
case HTTP_UPDATE_OK:
    Serial.println("[update] Update ok.");
    break;}
}
```

Sunucu tarafı komut dosyası bir önceki sayfadaki gibi bazı yanıtlar verebilir:
Eğer yanıt kodu 200 ise ürün yazılımı görüntüsünü gönderir. ESP'ye hiç bir
güncelleme gerekmediğini bildirmek için ise yanıt kodu 304'ü gönderir.



III. BİR UZAKTAN PROGRAM YÜKLEME HİZMETİNİN SİSTEM TASARIMI

Bu bölümde, genel IoT sistemleri ve önerilen uzaktan program yükleme (OTA) sistemi tanıtılmaktadır. Uzaktan program yükleme; hizmet bileşenleri genelde sistemde 3 türlü yer alabilir: Bunlar, yerelde yönetim, uzaktan yönetim ve bulut üzerinden dağıtılmış yönetimler olarak verilebilir.

3.1. Genel IoT sistemleri

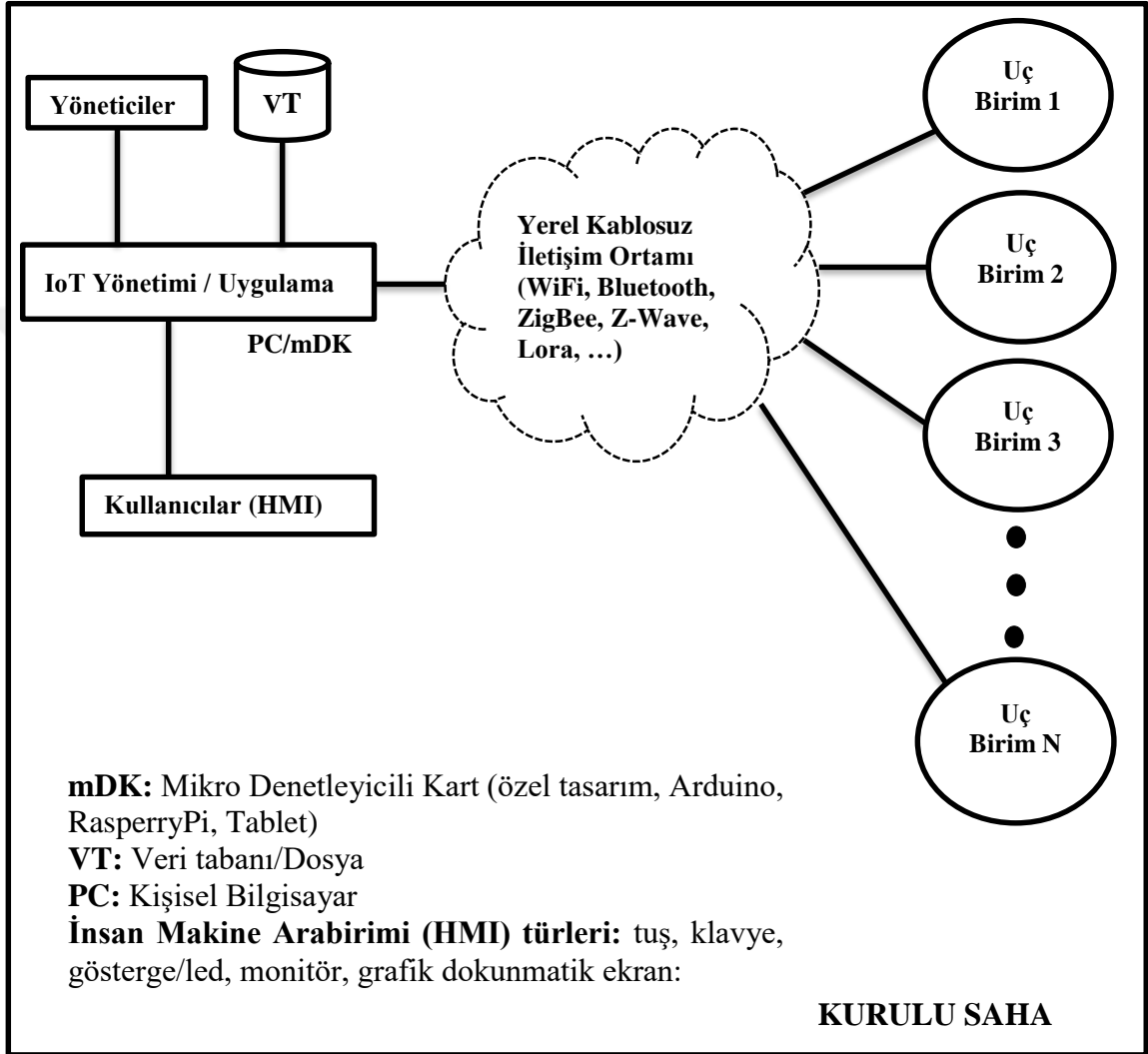
Genel IoT sistemlerindeki yönetimlerin türleri; yerel sistem, yerel yönetim/uzaktan erişim sistemi, uzaktan yönetim sistemi ve bulut üzerinden dağıtılmış yönetim sistemi olarak sınıflandırılabilir.

3.1.1. Yerel sistem

Tek başına (Standalone) tasarımı kapsayan yerel sistem, **Şekil 3.1**'de görülmektedir. Genel bir yapı olarak; kullanıcılar, yöneticiler, PC ya da gömülü sistem bileşeni olarak mikro denetleyicili kart (mDK), veri tabanı (VT) ya da dosya, yerel kablosuz iletişim ortamı (WiFi, Lora, Bluetooth, Z-Wave, ZigBee, vb.) ve uç birimler bu sistemde kurulu sahada yer alır. Mikro denetleyici kart içinde, yalnız mikro denetleyici ya da işletim sistemi çalıştıracak düzeyde gelişmiş bir mikro denetleyici barındırabilir. Mikro denetleyicili kart olarak; özel tasarım kartlar, Arduino ve RasperryPi ya da tabletler sayılabilir.

PC ya da mDK içerisinde, IoT yönetimi ve uygulamalar bulunmaktadır. Uygulama, tüm sistem denetimini/erişimini düzenleyen yazılım parçasıdır. IoT

yönetimi ise, uç birimlerle uygulama arasında veri alış verişini sağlayan bir alt sistemdir.



Şekil 3.1 Yerel Sistem

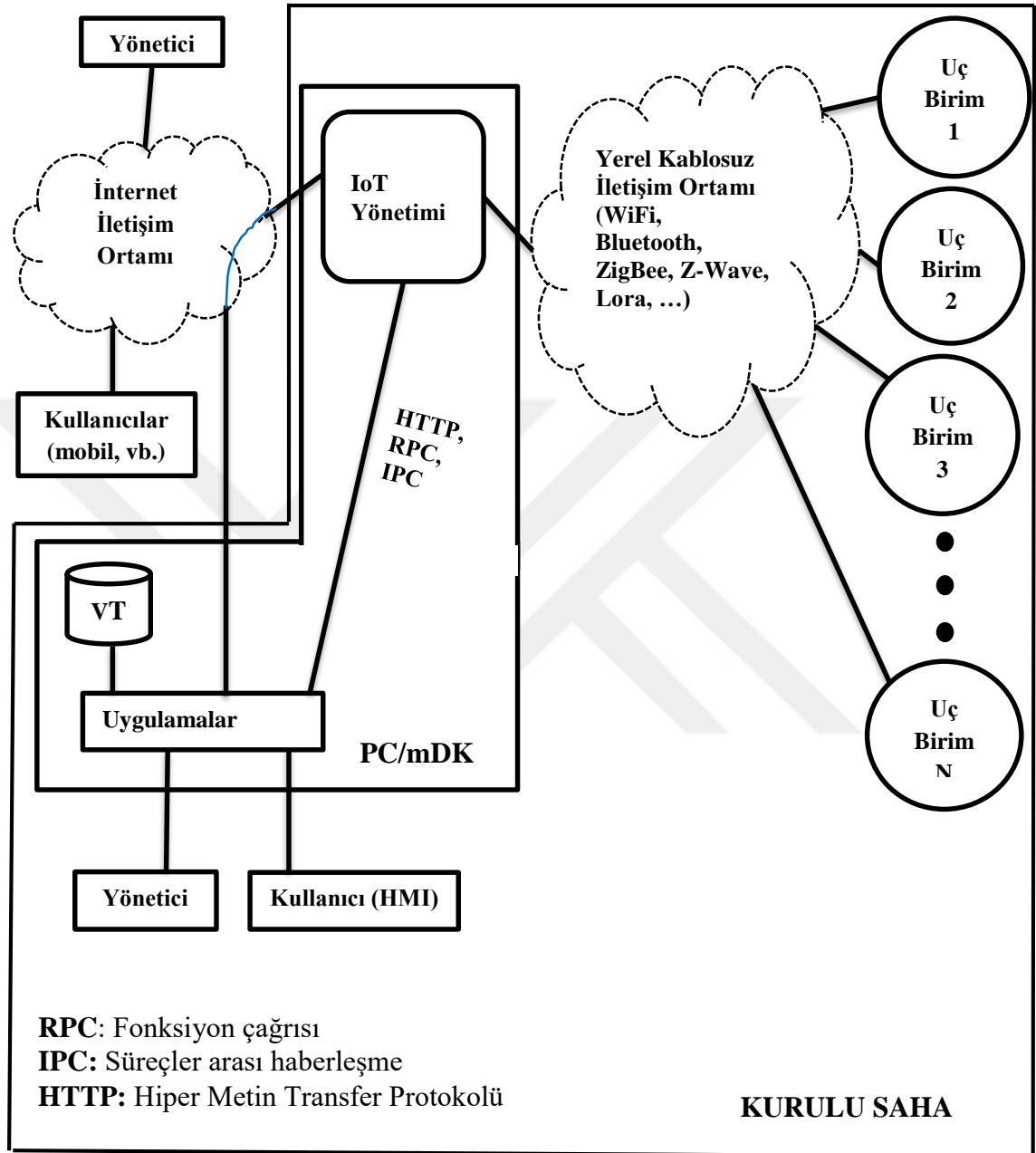
Bu sistemde, örneğin kullanıcılar isteklerini, IoT yönetimine iletebilir. Bu istekler de yerel kablosuz iletişim ortamı üzerinden uç birimlere aktarılır. Aynı zamanda uygulamalar vasıtasıyla, uç birimlerden yerel kablosuz iletişim ortamı üzerinden alınan veriler de veri tabanına ya da dosyaya kaydedilir.

Bu tasarımının dış dünya ile bağlantılı olması gerekmez. Bu nedenle, uzaktan program yükleme de yapılamaz. Bu sisteme yerinde program yüklenmesi söz konusudur.

3.1.2. Yerel yönetim/uzaktan erişim sistemi

Yerel yönetim/uzaktan erişim sistemi, **Şekil 3.2**'de görülmektedir. Bu tasarımın kurulu sahasında, kişisel bilgisayar(PC) ya da mikro denetleyicili kart (mDK), yönetici, kullanıcı (HMI-İnsan Makine Arayüzü), yerel kablosuz iletişim ortamı (WiFi, Lora, Bluetooth, Z-Wave, ZigBee, vb.) ve uç birim bileşenleri bulunmaktadır. PC içerisinde, IoT Yönetimi, uygulamalar, veri tabanı (VT) vardır. Bu sistemde internet iletişim ortamına bağlantılı olan yönetici veya kullanıcılar (mobil, vb.) internet iletişim ortamı üzerinden, kurulu sahadaki PC içinde bulunan uygulamalara bağlanır ve daha sonra uygulamalar da IoT yönetimine bağlanır. PC içerisinde bulunan uygulamalara doğrudan bağlı olan yönetici veya kullanıcı (HMI) ise, ya PC içerisindeki uygulamalar vasıtasıyla doğrudan ya da internet iletişim ortamı üzerinden dolaylı olarak IoT yönetimine bağlanır. IoT yönetimi de yerel kablosuz iletişim ortamı üzerinden uç birimlerle iletişindedir.

Bu sistemde, IoT yönetimi ile uygulamaların birbirinden ayrılması söz konusudur. IoT yönetimi, kurulu saha içindedir. Dış dünyada bulunan kullanıcılar ya da yönetici internet iletişim ortamı üzerinden bu kurulu sahaya uzaktan erişim sağlayabilmektedir.

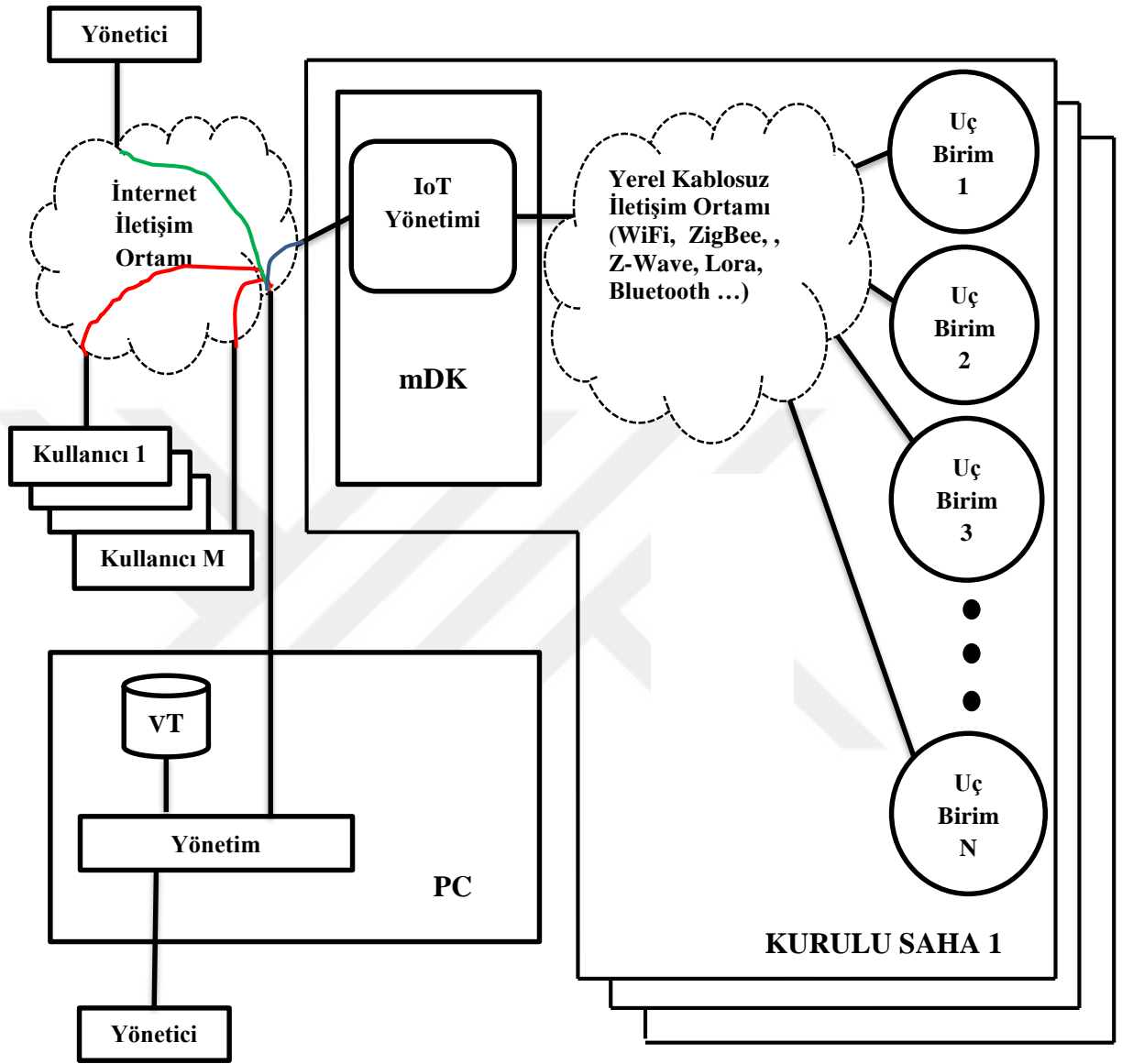


Şekil 3.2 Yerel Yönetim/Uzaktan Erişim Sistemi

Yerel yönetim/uzaktan erişim sisteminde IoT yönetimi ile uygulamalar kendi içinde de farklı yöntemlerle (RPC, IPC, HTTP) bağlanabilir.

3.1.3. Uzaktan yönetim sistemi

Uzaktan yönetim sistemi, **Şekil 3.3**'te görülmektedir. Bu tasarımın kurulu sahasında, gömülü sistem (mDK), yerel kablosuz iletişim ortamı (WiFi, Lora, Bluetooth, Z-Wave, ZigBee, vb.) ve uç birim bileşenleri bulunmaktadır. IoT yönetimi mDK içerisinde bulunmaktadır. Burada birden fazla kurulu saha da olabilir. PC içerisinde ise, yönetim, veri tabanı (VT) vardır. Yönetici, PC üzerinden yönetim ile bağlantılıdır. Her kurulu sahada ayrı bir IoT yönetimi olacaktır. Yönetim her bir IoT yönetimi üzerinden tüm sistemi yönetir. Bu sistemde, internet iletişim ortamına bağlı olan yönetici ve birden fazla kurulu sahaya atanmış kullanıcılar, yönetime doğrudan bağlanırken, IoT yönetimine doğrudan bağlanmazlar. Kullanıcılar, kendileri ile ilişkili kurulu saha işlemlerini yürütürler.



Kullanıcılar, birden fazla kurulu sahaya

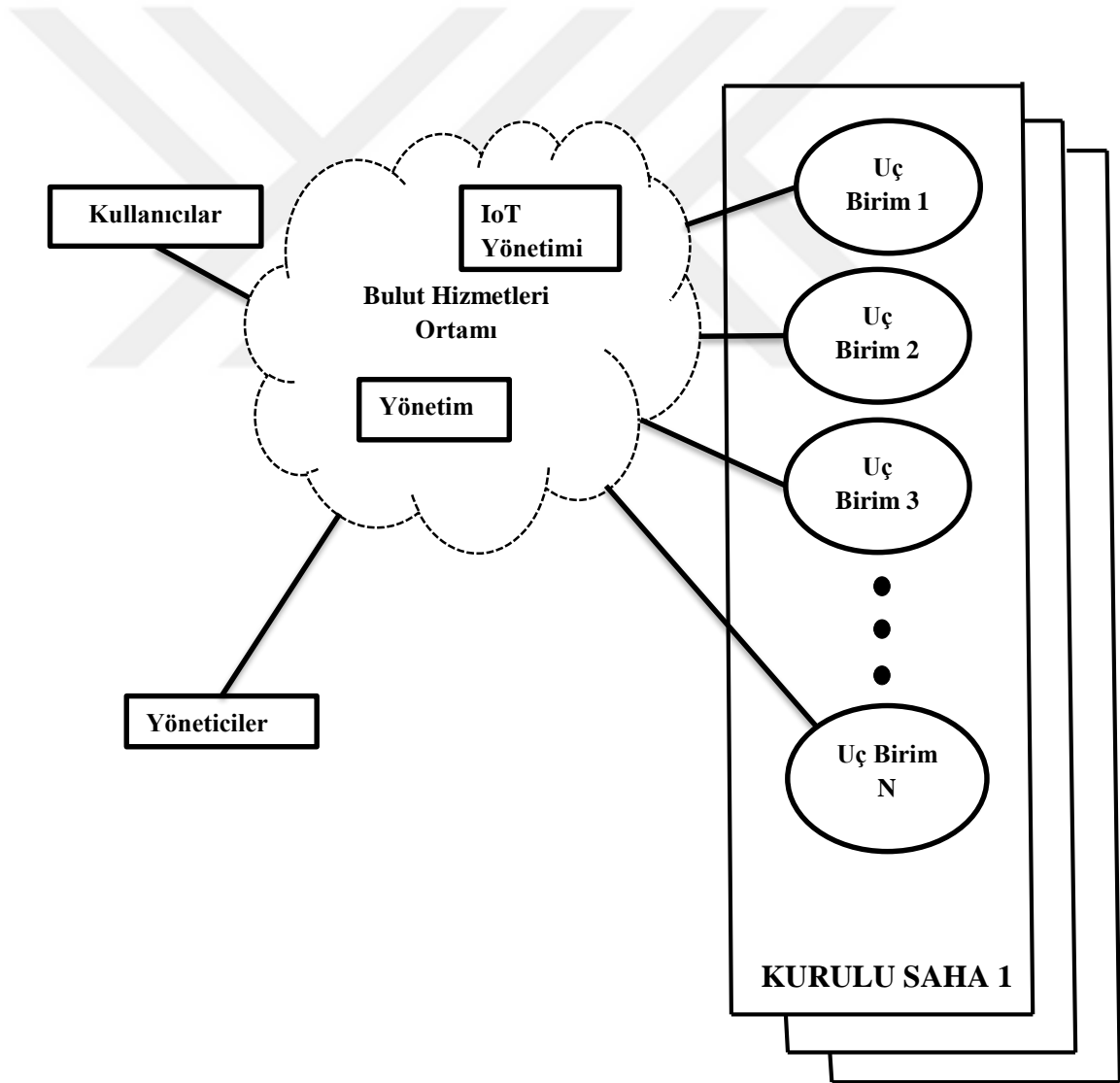
Birden Fazla Kurulu Saha

Şekil 3.3 Uzaktan Yönetim Sistemi

Bu tasarımda, bir kurulu saha için bir adet mDK gösterilmiştir. Kurulu sahanın karmaşık ya da dağılık olması durumunda ise mDK sayısı çoğaltılabilir.

3.1.4. Bulut üzerinden dağıtılmış yönetim sistemi

Bulut üzerinden dağıtılmış yönetim sistemi kullanıcılar, yöneticiler, bulut hizmetleri ortamı ve uç birim bileşenlerinden oluşmaktadır. Bu tasarımın özelliği, her nesnenin örneğin 4G veya yakın gelecekte 5G ile doğrudan buluta bağlanması ile tüm sistemin dağıtılmasının sağlanabilmesidir. Şekil 3.4'te, kurulu sahada uç birimler bulunmaktadır. Uç birimler de doğrudan buluta bağlıdır.



Şekil 3.4 Bulut Üzerinden Dağıtılmış Yönetim Sistemi

Sistem tasarımı, yönetim ve IoT yönetimi bulut hizmetleri olarak gerçekleştirilir. IoT yönetimi, kurulu sahadaki uç birimleri yönetirken, yönetim ise birden fazla kurulu sahanın IoT yönetimlerini ve kullanıcılarla olan veri alışverişlerini denetler. Kullanıcılar ve yöneticiler bulut hizmetleri ortamı üzerinden işlemlerini yürütür.

Dağıtılmış sistemde Wi-Fi türü bağlantılar da yer alabilir. Ancak burada uç birimler bir geçit (gateway) üzerinden buluta bağlanmak durumunda kalırlar.

3.2. Önerilen uzaktan program yükleme hizmeti sistemi

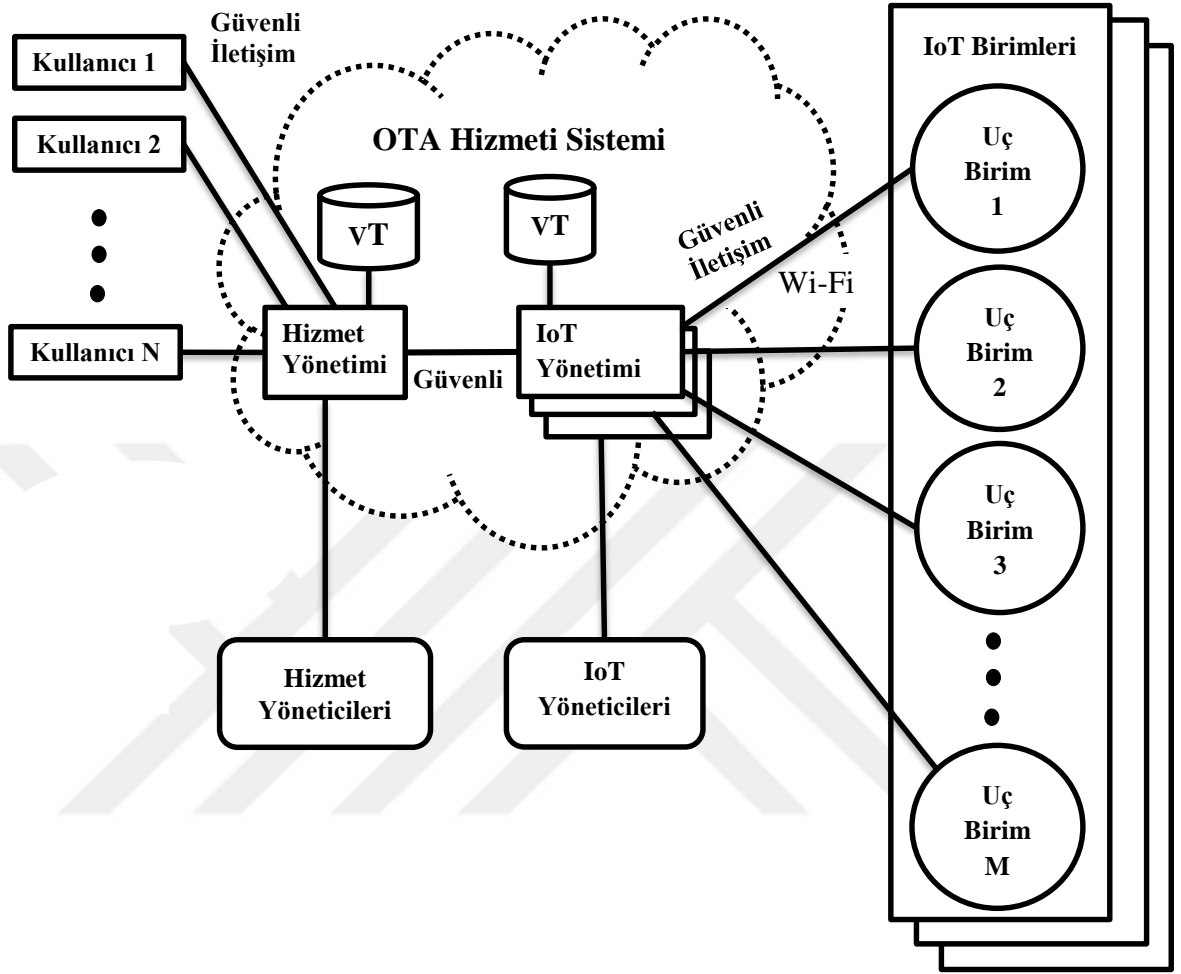
Uzaktan program yükleme (OTA) hizmeti sistemi, sahadaki IoT uç birimlere yeni program yükleme işlemini firma adına güvenilir bir şekilde yerine getiren bir sistemdir. Önerilen uzaktan program yükleme hizmetinin sistem tasarımı **Şekil 3.5**'te verilmektedir. OTA hizmeti sistemi; hizmet yönetimi, IoT yönetimi, kullanıcılar, yöneticiler ve uç birimlerden oluşmaktadır. Hizmet ve IoT yönetimlerinin, ayrı ayrı hizmet ve IoT yöneticileri vardır.

OTA hizmeti sistemi olarak, uç birimlerle kullanıcı arasında, doğrudan veri alışverişini güvenlik nedeniyle kısıtlanmış ve işlemlerin hizmet yazılımı aracılığı ile kullanıcılar, kendi uygulamalarına ait işlemlerinin, haberleşmesinin yapılması sağlamıştır. OTA hizmeti haberleşmesini ayrı bir ortam üzerinden yürütür.

OTA uygulamasında, kullanıcı A'nın istediği uç birime program yüklemesi için, kullanıcı A, "Sisteme kayıt yaptırdı mı?", "Kayıtlı ise sözleşmeleri imzaladı mı?" veya "Almak istediği hizmet için ücretini ödedi mi?" gibi mali aşamaları da geçmesi, sonra da isteğini hizmet yönetimine bildirmesi istenir. Eğer gelen uygun bir istek ise hizmet yönetimi isteği ve program dosyasını IoT yönetimine aktarır. IoT yönetimi de

gelen programı uç birimlere gönderir. Sonuç olarak, uç birimde gerçek program yüklemesi yapılır.





Kullanıcı İşlemleri

- Kullanıcı kaydı
- Firma kaydı
- IoT uç birim listesi yükleme
- Yeni sürüm program gönderme
- Uç birim yeni sürüm program sorgulama/durum
- Mali işlemler

Hizmet Yönetim İşlemleri

- Kullanıcı listesi
- Firma listesi
- Mali işlemler listesi
- IoT uç birim listesi
- IoT yönetim listesi
- Kurulu saha listesi
- Uç birim kayıtları/durumları listesi

Uç Birim İşlemleri

- Kendini kayıt etme
- Yeni sürüm program sorgulama
- Yeni sürüm program alma (OTA)

IoT Yönetim İşlemleri

- Yeni sürüm program sorgulama/alma
- Yeni sürüm program gönderimi (uç birime)
- Uç birim kaydı alma
- Uç birim kayıtlarını/durumlarını hizmet yönetimine bildirme

Şekil 3.5 Önerilen Uzaktan Program Yükleme Hizmeti

Önerilen Sistemde Yapılan İşlemler:

- Uç Birim İşlemleri
- Hizmet Yönetim İşlemleri
- IoT Yönetim İşlemleri
- Kullanıcı İşlemleri

şeklindedir.

Yeni üye kayıt, yeni cihaz kayıt, yeni program yükleme, raporlama, ücret ödeme ekranları üzerinden işlemler yapılır.

3.2.1. Uzaktan program yükleme sistem bileşenleri

Uzaktan program yükleme sistem bileşenleri:

- Uç Birimler
- Yerel yönetim yazılımları
- Dağıtılmış yönetim
- Kullanıcılar
- IoT yöneticisi
- Hizmet yöneticisi

şeklindedir.

Hizmet yönetimi, sertifikalar, yeni düğüm ekleme, düğüm çıkarma, sistemin bakımı, veri tabanı gibi işlemlerden sorumludur. IoT yönetimi ise uç birimleri programlama ve uç birimleri izleme görevlerine sahiptir. Kullanıcılar sistemi kullanabilmek için önce sisteme kaydını yapmalıdır. Daha sonra kullanıcıya ait firma kaydı ve hizmet alma işlemleri yapılır.

3.2.2. IoT yöneticilerinin işleri

IoT yöneticilerinin işleri, birden çok IoT yönetimlerin olması, birden çok uç birim işlemlerinin olması, arıza ve durum bilgilerini izleme, kendisine atanmış firmaların onayını gerçekleştirme durumunda ortaya çıkar. Bu yöneticiler, yeni bir IoT yönetimi yaratma ve bunları firmalara bağlama, arıza ve istek bildirimlerini izleme, özet rapor çıkarma, kalp atışları izleme, veritabanı bakımı gibi işleri yapar.

Uç birimlerde haberleşme sayısı fazla olacağından, haberleşmede darboğaz oluşmaması için birden fazla IoT yönetim modülleri yaratılır. Bunların bazıları firmalara özel olarak, bazıları ise firmalarca ortak kullanılabilir.

Diğer bir durum ise, haberleşme bağlantılarının kesilmesi ya da kesintiye uğraması durumlarıdır. Belli bir süre kesintinin olması ve sonra haberleşme bağlantılarının düzelmesi durumunda sistemin ne yapması gerektiği konusunda IoT yönetimi karar verir.

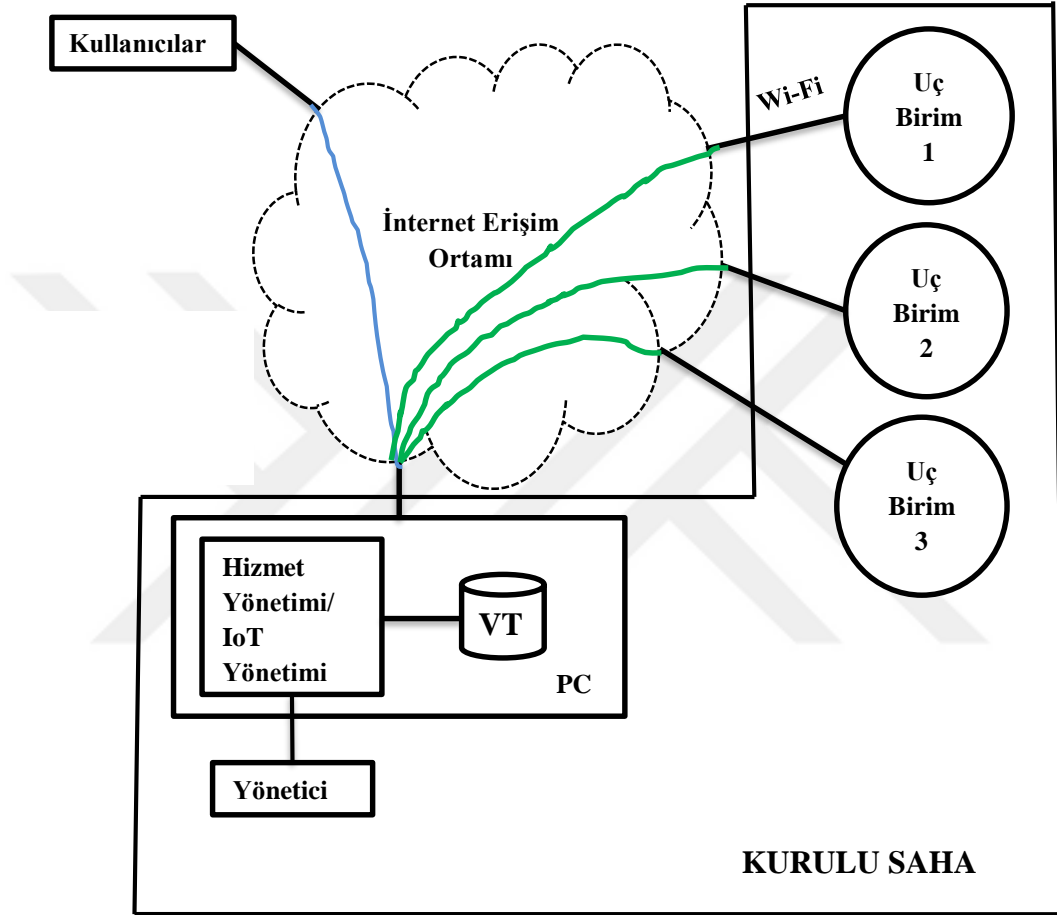
3.3. Gerçekleştirme aşamaları

Gerçekleştirme aşamaları, tümleşik yönetim ve uzaktan yönetimli sistem olarak uygulanmıştır.

3.3.1. Tümleşik yönetim aşaması

Tümleşik yönetim sistem yapısı, **Şekil 3.6**'da verilmektedir. Uç birimler olarak NodeMCU cihazı kullanılmıştır. Bu cihazların kullanılmasının sebebi hem ucuz olması hem de Wi-Fi ile ağa kolayca bağlanabilmesidir.

Bu sistemde, kullanıcı ile hizmet yönetimi, IoT yönetimi tümleşiktir ve uç birimler arasında haberleşme ortamı vardır.

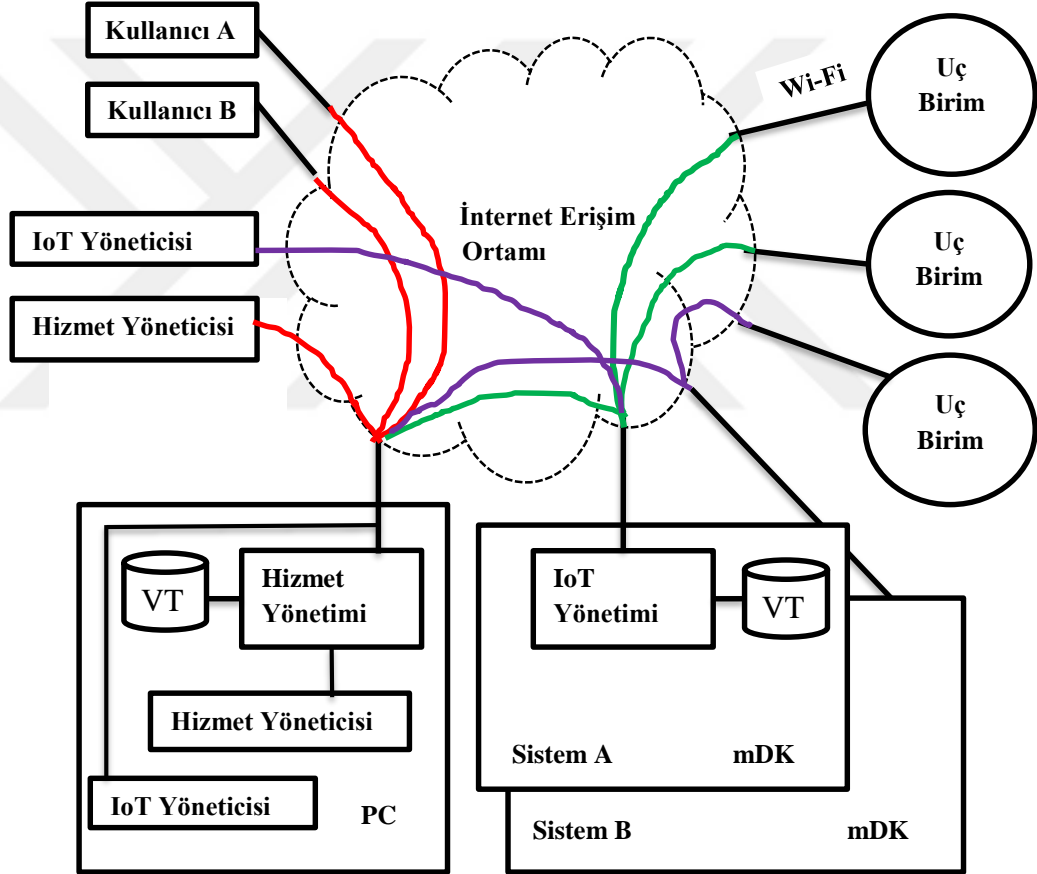


Şekil 3.6 Tümüleşik Yönetim

Tümüleşik yönetimde, sisteme kayıt olan kullanıcılar, internet iletişim ortamı üzerinden PC üzerinden hizmet yönetimiyle iletişim kurar. Hizmet yönetimi gelen isteği uygun bulursa isteği IoT yönetimine iletir. IoT yönetimi de uç birimlerde isteği uygular.

3.3.2. Uzaktan yönetim sistemi aşaması

İki yönetimli sistem Şekil 3.7’de verilmektedir. Bu sistem tasarımında hizmet yönetimi PC üzerinde iken, kullanıcı A’ya ait olan IoT yönetimi ile kullanıcı B’ye ait olan IoT yönetimi farklı kurulu sahalarda bulunur. Burada PC kullanma yerine Bulut üzerinden web servisi hizmeti alınarak öngörülen işlevler gerçekleştirilebilir.



Şekil 3.7 İki Yönetimli Sistem

IV. YEREL YÖNETİM/UZAKTAN ERİŞİM SİSTEMİ ÜZERİNE BİR UYGULAMA

Genel IoT sistemlerinde tanıtılmış olan yerel yönetim/uzaktan erişim sisteminin önerilen IoT hizmeti şeklinde uygulanması, tümleşik yönetim başlığı altında gerçekleştirilmiştir.

4.1. Tümleşik yönetim ortamı

Tümleşik yönetim ortamında, kullanıcılar, internet erişim ortamı ve kurulu saha bulunmaktadır. Kurulu sahada, uç birimler, bir adet PC bulunur. PC içerisinde veritabanı, hizmet yönetimi ve IoT yönetimi bulunmaktadır. PC'ye bağlı bir yönetici vardır.

Tümleşik yönetim sisteminde Xampp kullanılmıştır. Xampp, yerel web sunucusu oluşturmayı sağlayan, bir Apache ve diğer uygulamaları kapsayan bir dağıtımdır. Apache de bir web sunucusu uygulamasıdır. Bu bölümde gerçekleştirilen uygulama, sunucu tarafı betik bir dil olan PHP dili ile yazılmıştır. Verileri tutmak için hem açık kaynak kodlu olan hem de php ile çok iyi bir şekilde çalışan MySQL veritabanı kullanılmıştır. Uç birimler için, NodeMCU cihazları kullanılmıştır. Uç birimlerin yazılım ve donanım olarak program yüklemeye hazır olması gerekir.

4.2. Tümleşik yönetimde yapılan işlemler

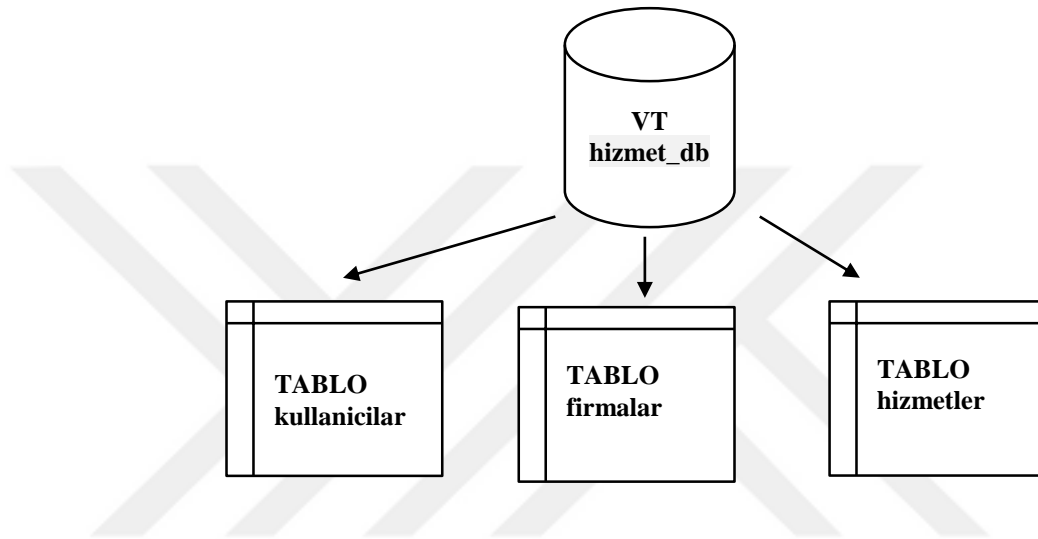
Tümleşik yönetimde yapılan işlemler aşağıdaki **Tablo 4.1**'de verilmektedir.

Tümleşik Yönetimde Yapılan İşlemler	Yapılan İşlemler
Kullanıcı İşlemleri	<ul style="list-style-type: none"> • Kullanıcı kaydı • Firma kaydı/ kendine ait olanları listeleme • Hizmet alma / kendine ait olanları listeleme • IoT uç birim listesi • Yeni sürüm program gönderme • Uç birim yeni sürüm program sorgulama/durum
Hizmet Yönetim İşlemleri	<ul style="list-style-type: none"> • Tüm kullanıcıların listesi • Tüm firmaların listesi • Tüm hizmetlerin listesi • Kullanıcıdan gelen hizmet talebini onaylama • IoT uç birim listesi • Uç birim kayıtları/durumları listesi • Kullanıcıdan gelen istekleri IoT yönetime bildirme
IoT Yönetim İşlemleri	<ul style="list-style-type: none"> • Yeni sürüm program sorgulama/alma • Yeni sürüm program gönderimi (uç birime) • Uç birim kaydı alma • Uç birim kayıtlarını/durumlarını hizmet yönetimine bildirme
Uç Birim İşlemleri	<ul style="list-style-type: none"> • Kendini kayıt etme • Yeni sürüm program sorgulama • Yeni sürüm program alma (OTA)

Tablo 4.1 Tümleşik Yönetimde Yapılan İşlemler

4.3. Tümleşik yönetim veritabanı tasarımı

Tümleşik yönetimde, hizmet yönetimi için hizmet_db adlı veritabanı oluşturulmuştur ve Şekil 4.1’de verilmektedir.




Şekil 4.1. Tümleşik Yönetimdeki hizmet_db Adlı Veritabanı

Şekil 4.2, Şekil 4.3 ve Şekil 4.4’te tabloların içinde oluşturulan alanlar görülmektedir.

#	Adı	Türü
1	id	int(11)
2	kullanici_adi	varchar(15)
3	sifre	varchar(15)
4	yetki	varchar(15)


Şekil 4.2 kullanıcılar Tablosu

Bu sistemde, yönetici (admin) ve kullanıcı oluşturabilmek için Şekil 4.2 ‘de görüldüğü üzere tabloda yetki alanı bulunmaktadır.

#	Adı	Türü
1	id 	int(11)
2	firma_adi	varchar(15)
3	email	varchar(15)
4	kullanici_id	int(15)

Şekil 4.3 firmalar Tablosu

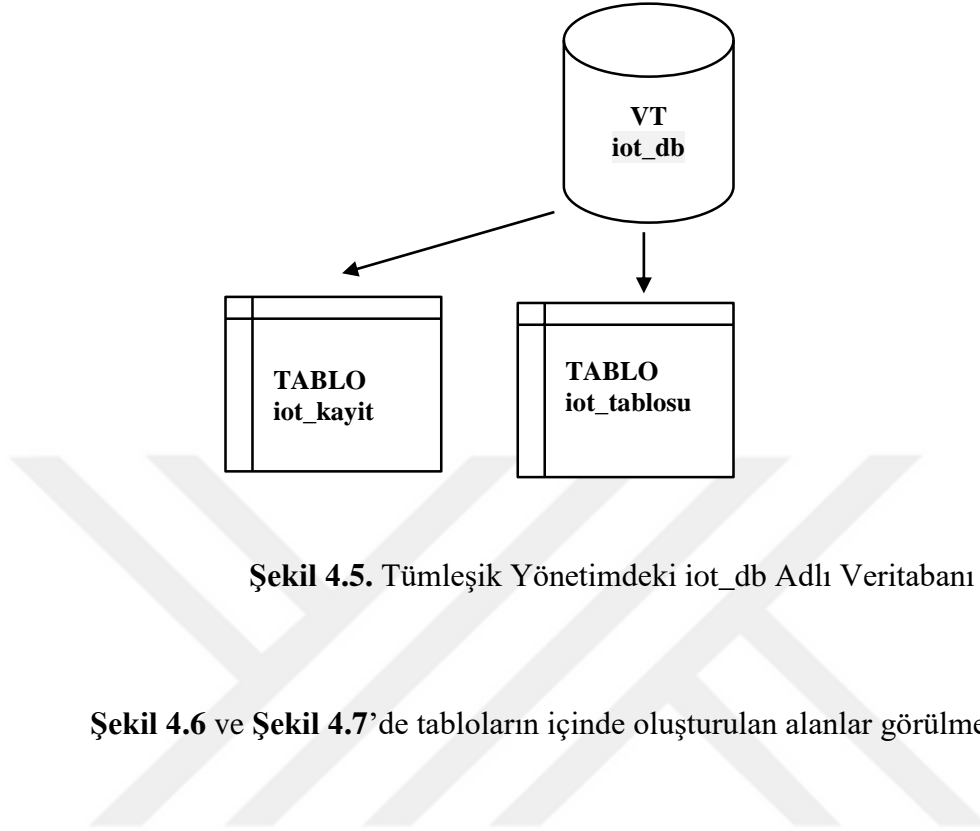
Bu sistemde kullanıcılara ait firmaları oluşturabilmek için Şekil 4.3’te verilen tabloda kullanıcı_id alanı oluşturulmaktadır.

#	Adı	Türü
1	h_kaydi_zamani	timestamp
2	id 	int(11)
3	kullanici_id	int(15)
4	firma_id	int(15)
5	h_baslama_zamani	varchar(15)
6	h_bitis_zamani	varchar(15)
7	dosya_gonderme	varchar(5)
8	yeni_surum_yukleme	varchar(5)
9	cihaz_durumu_gorme	varchar(5)
10	cihaz_listeleme	varchar(5)
11	odeme_tutari	float
12	odeme_durumu	varchar(15)

Şekil 4.4 hizmetler Tablosu

Şekil 4.4’teki tabloda hizmetler kullanıcı ve firmaya bağlı olduğundan kullanıcı_id ve firma_id alanları oluşturulmaktadır.

Tümleşik yönetim sisteminin IoT yönetimi için iot_db adlı veritabanı oluşturulmuştur ve Şekil 4.5’te verilmektedir.



Şekil 4.5. Tümlleşik Yönetimdeki iot_db Adlı Veritabanı

Şekil 4.6 ve Şekil 4.7’de tabloların içinde oluşturulan alanlar görölmektedir.

#	Adı	Türü
1	id	int(6)
2	ip	varchar(30)
3	iot_adi	varchar(30)
4	ver	varchar(30)
5	yeniver	varchar(30)
6	Date	date
7	Time	time
8	Time Stamp	timestamp

Şekil 4.6. iot_kayit

#	Adı	Türü
1	id 📄	int(6)
2	ip	varchar(30)
3	iot_adi	varchar(30)
4	ver	varchar(30)
5	yeniver	varchar(30)
6	Date	date
7	Time	time
8	durum	text
9	Time Stamp	timestamp

Şekil 4.7. iot_tablosu

4.4. Tümleşik yönetimde senaryolar

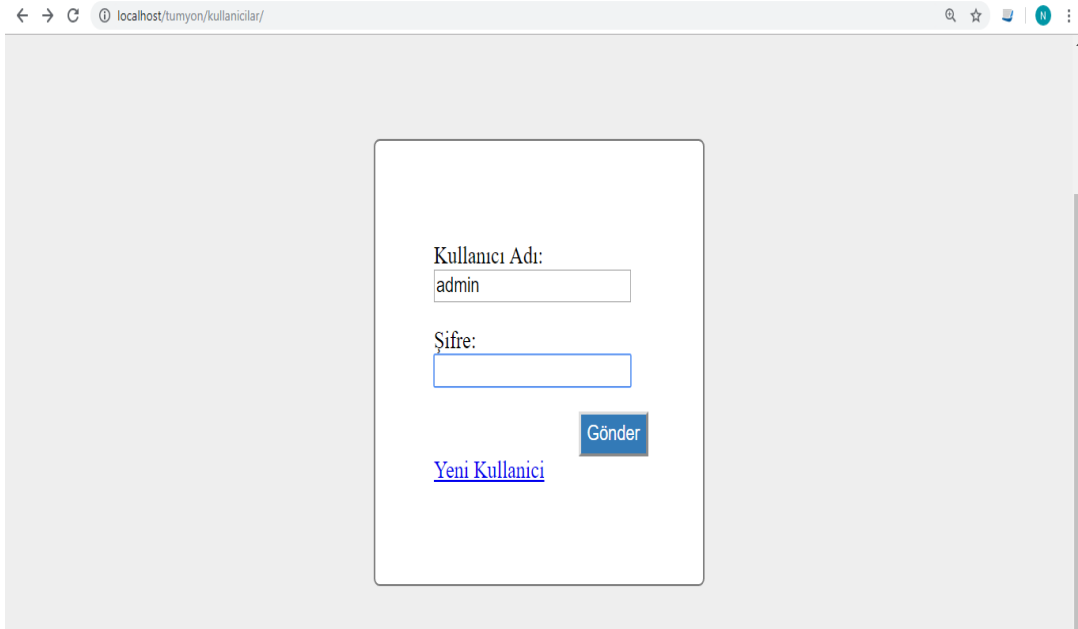
Tümleşik yönetimde senaryolar aşağıdaki şekilde oluşturulmuştur:

- Kullanıcı kaydı, firma kaydı ve hizmet alımı olmayan kullanıcının sistemde işlem yapamaması.
- Firma hizmet alım talebini gönderdikten sonra firmanın hizmeti kullanabilmek için yöneticinin onay vermesini beklemesi.
- Uç birimlerin bağlantısının kesilmesi durumunda bağlantı tekrar geldiğinde işlemine kaldığı yerden devam edebilmesi.
- Uç birimlerin sisteme kendini kaydettirmesi.
- Kullanıcıların uç birimlere doğrudan müdahale edememesi ve kullanıcılar isteklerini hizmet yönetimine bildirerek IoT yönetimi üzerinden uç birimlere iletmesi.

4.5. Kullanıcının sisteme kayıt olma durumu

Kullanıcı sisteme kayıt olmazsa sistemi kullanamaz. Sisteme kayıt olan kullanıcı firma kaydı yapar. Firma kaydı olan kullanıcı da firma için hizmet satın alır. Bütün hizmetleri alan kullanıcı, kendi sistemine dosya gönderebilir, uç birimlere yeni sürüm yükleyebilir, uç birimleri listeleyebilir ve uç birimlerin durumlarını görebilir.

Şekil 4.8’de kullanıcı ya da yönetici girişi yapılabilmektedir. Eğer kayıtlı bir kullanıcı yoksa yeni kullanıcı kaydı yapılabilmektedir.

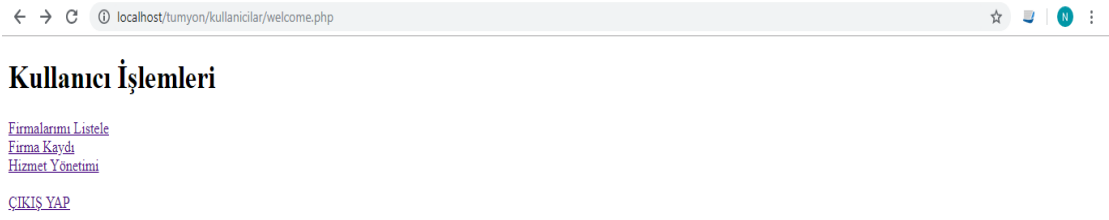


The screenshot shows a web browser window with the address bar displaying 'localhost/tumyon/kullanici/'. The main content area features a login form with the following elements:

- A label 'Kullanıcı Adı:' followed by a text input field containing the text 'admin'.
- A label 'Şifre:' followed by a password input field.
- A blue button labeled 'Gönder' positioned to the right of the password field.
- A blue link labeled 'Yeni Kullanıcı' located below the password field.

Şekil 4.8. Sisteme Giriş

Şekil 4.9’da kullanıcı işlemleri görülmektedir. Kullanıcı firma kaydı yapmışsa kayıtlı olan firmalarını listeleme yapabilmektedir.



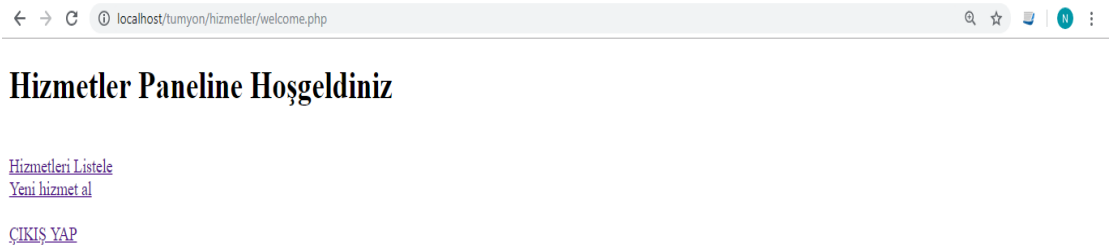
Şekil 4.9. Kullanıcı İşlemleri

Şekil 4.10'te kullanıcının firma listesi gözükmemektedir. Kullanıcı firma adına tıklarsa hizmet alımı yapabilmektedir.

id	Firma Adı	Email Adresi	Kullanici_id
13	aaa	nilgunincereis@	3
12	fni1211	ni2@fni.com	3
11	fniqw	qweqwe@efrefe	3
10	fni3232	n2i@fni.com	3

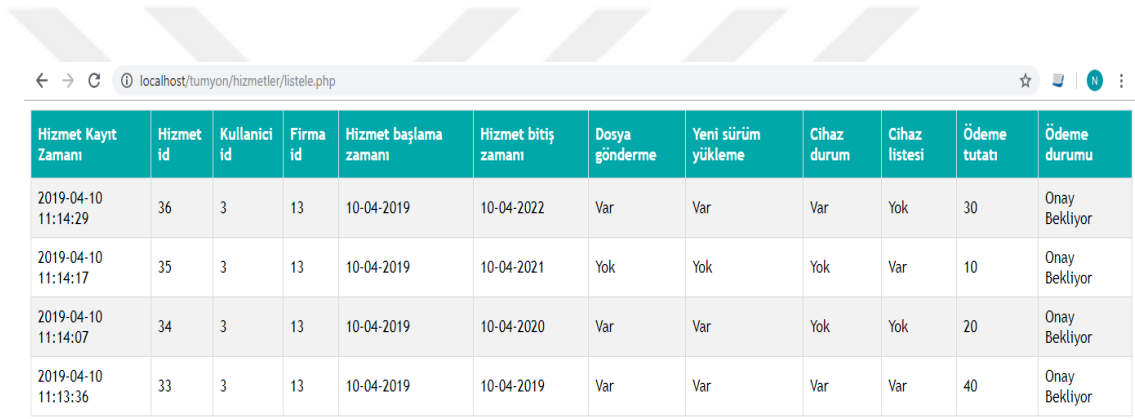
Şekil 4.10 Kullanıcının Firma Listesi

Şekil 4.11'teki hizmet işlemleri panelinde, kullanıcı eğer hizmet alımı yaptıysa hizmetleri listeledebilmektedir.



Şekil 4.11 Hizmet İşlemleri Paneli

Şekil 4.12’de firmaların hizmet listesinde hizmet talebi yapılmış fakat hizmeti kullanabilmek için ödeme durumunda yöneticinin onay vermesi beklenmektedir. Yönetici ödeme durumu ödendi olarak değiştirirse kullanıcı ve dosya gönderme, yeni sürüm yükleme, cihaz durumu ve cihaz listesi hizmetlerinden sadece var olanları kullanıcı kullanabilmektedir.



Hizmet Kayıt Zamanı	Hizmet id	Kullanıcı id	Firma id	Hizmet başlama zamanı	Hizmet bitiş zamanı	Dosya gönderme	Yeni sürüm yükleme	Cihaz durumu	Cihaz listesi	Ödeme tutarı	Ödeme durumu
2019-04-10 11:14:29	36	3	13	10-04-2019	10-04-2022	Var	Var	Var	Yok	30	Onay Bekliyor
2019-04-10 11:14:17	35	3	13	10-04-2019	10-04-2021	Yok	Yok	Yok	Var	10	Onay Bekliyor
2019-04-10 11:14:07	34	3	13	10-04-2019	10-04-2020	Var	Var	Yok	Yok	20	Onay Bekliyor
2019-04-10 11:13:36	33	3	13	10-04-2019	10-04-2019	Var	Var	Var	Var	40	Onay Bekliyor

Şekil 4.12 Firmanın Hizmet Listesi

4.6. Kullanıcı işlemleri

Kullanıcılar, dosya gönderme, yeni sürüm yükleme, cihaz durumu ve cihaz listesi hizmetlerini aldığında ve yönetici ile ilgili gerekli anlaşmalar tamamlandığında firma adına bu hizmetleri kullanabilmektedir.

Dosya gönderme hizmeti, kullanıcıların sisteme dosya yükleme yapabildikleri hizmettir. Kullanıcının sisteme gönderdiği dosyanın boyutunda kısıtlama yapılmıştır. Kullanıcı, 4 MB’ dan büyük dosyaları sisteme yükleyemez. Bu boyut yönetici tarafından değiştirilebilir.

Yeni sürüm yükleme hizmeti, kullanıcının sistemdeki dosyalardan istediğini istediği uç birimine yükleme isteğini hizmet yönetimine bildirilmesidir. Hizmet yönetimi de IoT yönetimine bu isteği ileterek IoT yönetimi onay verirse uç birime yükleme yapılmaktadır.

Cihaz durum hizmeti, kullanıcı tarafından her hangi uç birimin verilerinin gözlenebilmesidir. Burada programa göre kurgulanmış sensörden gelen veriler eş zamanlı gözükmetedir.

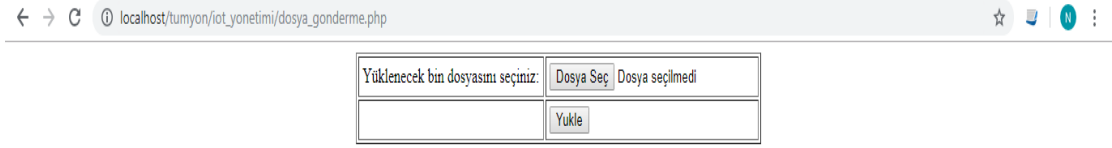
Cihaz listesi hizmetinde, kullanıcının kendine ait uç birimlerinin listesini görebilmesidir. Bu listede cihazın aktif ya da pasif olması durumları eş zamanlı gözükmetedir.

Kullanıcı hizmet satın aldığıında **Şekil 4.13**'teki IoT yönetim paneline erişebilmektedir.



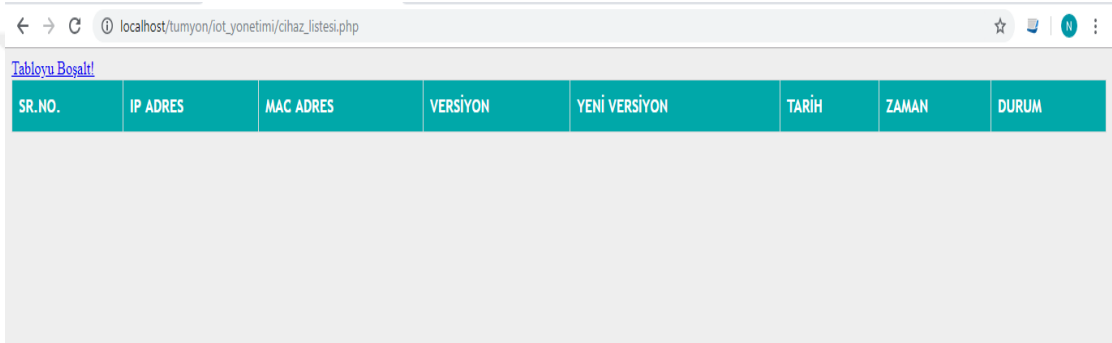
Şekil 4.13 IoT Yönetim Paneli

Kullanıcı, sisteme dosya göndermek istediğinde hizmet satın alınmış ve yönetici tarafından onay verilmişse, **Şekil 4.14**'deki dosya gönderme işlemi kullanıcı tarafından yapılabilmektedir.



Şekil 4.14. Dosya Gönderme Hizmeti

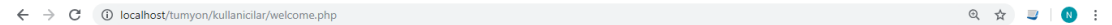
Şekil 4.15’te IoT cihaz listesi ve cihazla ilgili bilgiler görülmektedir.



Şekil 4.15 IoT Cihaz Listesi

4.7. Yönetici işlemleri

Yöneticinin yapabildiği işlemler Şekil 4.16’da görülmektedir. Bu işlemler tüm kullanıcıları listeleme, tüm firmaları listeleme, hizmetleri listeleme, hizmet yönetimi, firma kaydı şeklindedir.



Yönetici İşlemleri

[Kullanıcıları Listele](#)
[Firma Kaydı](#)
[Firma Listele](#)
[Hizmetleri Listele](#)
[Hizmet Yönetimi](#)

[ÇIKIŞ YAP](#)

Şekil 4.16 Yönetici İşlemleri

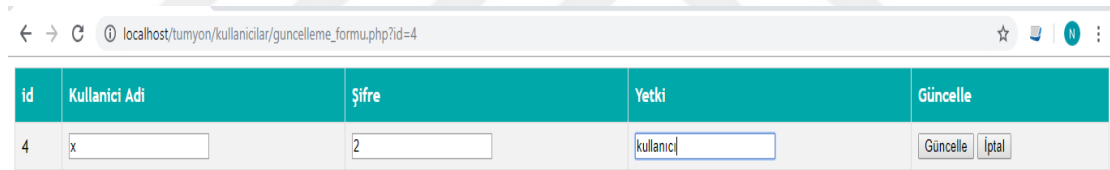
Tüm kullanıcıların listesi **Şekil 4.17**'de görülmektedir. Burada yönetici ve yetki şeklinde yetki alanı oluşturulmuştur. Yönetici isterse kullanıcıların bilgilerini güncelleyebilmektedir. Yönetici isterse istediği kullanıcıyı sistemden silebilir.



id	Kullanici Adi	Şifre	Yetki	Güncelle	Sil
5	nilguncereis	12345	kullanici	Güncelle	Sil
4	x	2		Güncelle	Sil
3	ni	1	kullanici	Güncelle	Sil
1	admin	123456	yönetici	Güncelle	Sil

Şekil 4.17 Tüm Kullanıcıların Listesi

Yönetici kullanıcı güncellemeyi **Şekil 4.18**'de görülen ekrandan yapmaktadır.



id	Kullanici Adi	Şifre	Yetki	Güncelle
4	<input type="text" value="x"/>	<input type="text" value="2"/>	<input type="text" value="kullanici"/>	<input type="button" value="Güncelle"/> <input type="button" value="İptal"/>

Şekil 4.18. Kullanıcı Güncelleme

Tüm hizmetlerin listesi **Şekil 4.19**'te görülmektedir. Yönetici isterse hizmetlerin bilgilerini güncelleyebilmektedir. Yönetici isterse istediği hizmeti sistemden silebilir. Kullanıcıların istediği hizmeti kullanabilmesi için hizmet talebi “Var” şeklinde ve ödeme durumu yönetici tarafından “Ödendi” olarak onaylanmış olması gerekmektedir. Örneğin kullanıcı id değeri 3 hizmet id değeri 30 olan kullanıcı ödeme durumu “Ödendi” yazsa da cihaz durum hizmetinde “Yok” yazdığı için cihaz durum hizmeti kullanılamaz.

localhost/tumyon/hizmetler/listele.php

Hizmet Kayıt Zamanı	Hizmet id	Kullanıcı id	Firma id	Hizmet başlama zamanı	Hizmet bitiş zamanı	Dosya gönderme	Yeni sürüm yükleme	Cihaz durum	Cihaz listesi	Ödeme tutarı	Ödeme durumu	Güncelle	Sil
2019-04-10 16:29:59	30	3	12	09-04-2019	09-04-2019	Var	Var	Yok	Var	20	Ödendi	Guncelle	Sil
2019-04-09 19:52:58	29	3	10	09-04-2019	09-04-2019	Var	Var	Var	Var	40	Onay Bekliyor	Guncelle	Sil
2019-04-09 19:41:05	28	3	10	09-04-2019	09-04-2019	Yok	Var	Var	Yok	0	Onay Bekliyor	Guncelle	Sil
2019-04-10 17:07:51	27	3	10	09-04-2019	09-04-2019	Var	Var	Yok	Yok	20	Ödendi	Guncelle	Sil

Şekil 4.19 Tüm Hizmetleri Listeleme

Hizmet bilgilerini güncelleme ekranı Şekil 4.20’de verilmektedir.

localhost/tumyon/hizmetler/guncelleme_formu.php?id=29

Dosya gönderme	Yeni sürüm yükleme	Cihaz durum	Cihaz listesi	Ödeme tutarı	Ödeme durumu	Güncelle
Var	Var	Var	Var	40	Onay Bekliyor	Güncelle
					Onay Bekliyor	İptal
					Ödendi	

Şekil 4.20 Hizmetleri Güncelleme

Tüm firmaların listesi Şekil 4.21’de görülmektedir.

localhost/tumyon/firmalar/listele.php

id	Firma Adı	Email Adresi	Kullanıcı_id	Güncelle	Sil
12	fni1211	ni2@fni.com	3	Guncelle	Sil
11	fniqw	qweqwe@efrefe	3	Guncelle	Sil
10	fni3232	n2i@fni.com	3	Guncelle	Sil
7	fni3	ni3@fni.com	0	Guncelle	Sil

Şekil 4.21 Tüm Firmaları Listeleme

4.8. Tümleşik yönetimde kullanılan cihazlar ve teknolojiler

Tümleşik yönetimde, kurulu saha içinde, veritabanı, hizmet ve IoT yönetimlerini barındıran PC ve uç birim olarak NodeMCU cihazları kullanılmıştır. NodeMCU üzerine yüklenen uygulamalar Arduino'da yazılmıştır.

Bu sistemde, web sunucu yazılımı olan Xampp ile yerel bir şekilde bilgisayarda çalışılmıştır. Xampp'da, phpmyadmin kurulu bir şekilde gelmektedir. Phpmyadmin, Php ile yazılmış açık kaynak kodlu bir araçtır. MySQL veritabanının yönetimi bu araç üzerinden sağlanmaktadır.

4.9. Tümleşik yönetimde kullanılan fonksiyonlar ve kütüphaneler

4.9.1. Hizmet fiyatını hesaplayan fonksiyon

Kullanıcı hizmet seçimini yaptıktan sonra fiyatını hesaplayan basit bir fonksiyon yazılmıştır. Bu fonksiyonda 4 adet hizmet durumu bulunur.

```
function hesapla(){  
var deger1=0;  
var deger2=0;  
var deger3=0;  
var deger4=0;  
if(document.getElementById("dosya_gonderme").checked) {  
    deger1=10;  
} else {  
    deger1=0;  
}  
}
```

```
if(document.getElementById("yeni_surum_yukleme").checked) {  
    deger2=10;  
} else {  
    deger2=0;  
}  
  
if(document.getElementById("cihaz_listeleme").checked) {  
    deger3=10;  
} else {  
    deger3=0;  
}  
  
if(document.getElementById("cihaz_durumu_gorme").checked) {  
    deger4=10;  
} else {  
    deger4=0;  
}  
  
document.getElementById("odeme_tutari").value = deger1+deger2+deger3+deger4;  
}
```

Tümleşik yönetim sistemi uygulaması için kullanılan fonksiyonlar **Tablo 4.2**'de açıklamalarıyla verilir.

Kullanılan Fonksiyonlar	Açıklama
void wifi_baglan()	WiFi bağlantısı yapar
void iot_guncelle(String ver)	IoT güncelleme işleri yapar
void iot_islemler()	IoT okuma ve IoT yazma işlemi yapar
void updateUI()	Arayüz güncelleme
void handlePage()	El sıkışma

Tablo 4.2 Kullanılan Fonksiyonlar

4.9.2. Kullanılan Kütüphaneler

Bu sistemde, kullanılan kütüphaneler **Tablo 4.3**'te verilmiştir.

Kullanılan Kütüphaneler:	Açıklama
#include <ESP8266WiFi.h>	WiFi ile ilgili işlevleri yapmak için gereklidir.
#include <WiFiClient.h>	Web tarayıcısına istek göndermek için gereklidir.
#include <ESP8266WebServer.h>	ESP8266WebServer, web sunucusu
#include <ESP8266HTTPClient.h>	ESP8266HTTP, http istemcisi
#include <ESP8266HTTPUpdateServer.h>	ESP8266HTTPUpdateServer, Güncelleme Sunucusu
#include <ESP8266httpUpdate.h>	ESP8266httpUpdate, http güncelleme
#include <EmblAJAX.h>	Ajax kütüphanesi

Tablo 4.3 Kullanılan Kütüphaneler

Bu çalışmada, IoT aygıtlarının yeni program güncellemesi kablosuz olarak yapılması gerekir. Böylece, aygıtların kötü amaçla saldırıya uğrayabileceği ve başka bir kod yüklenme ihtimallerini ortaya çıkarmaktadır. Saldırıya uğramanın olasılığını azaltmak için, yüklemelerinizi bir şifreyle, belirli OTA bağlantı noktası vb. seçerek korumalıyız. Güvenliği artırmak için, ArduinoOTA kütüphanesi ile sağlanan işlevselliği kontrol edebiliriz [17].

```
void setPort(uint16_t port);
void setHostname(const char* hostname);
void setPassword(const char* password);
```

Bunun gibi, belli koruma işlevleri önceden oluşturulmuştur ve geliştirici tarafından herhangi bir ek kodlama gerektirmez. ArduinoOTA ve espota.py, karşıdan yüklemeyi doğrulamak için Digest-MD5 kullanıyor. Aktarılan verilerin bütünlüğü, ESP tarafında MD5 kullanılarak doğrulanır [17].

4.10. Tümleşik yönetimin çalışma prensibi

Bu çalışmanın 1. aşamasında, hizmet yönetimi ve IoT yönetimi aynı PC üzerinde bulunmaktadır. Her iki yönetiminde ayrı ayrı veritabanı oluşturulmuştur. Aslında tek bir veritabanı da yeterli olur. Çünkü bütün sistem yerdedir. Sonraki aşama uygulamalarına yönelik veritabanları ve yönetim sistemleri şimdiden ayrı ayrı oluşturuldu.

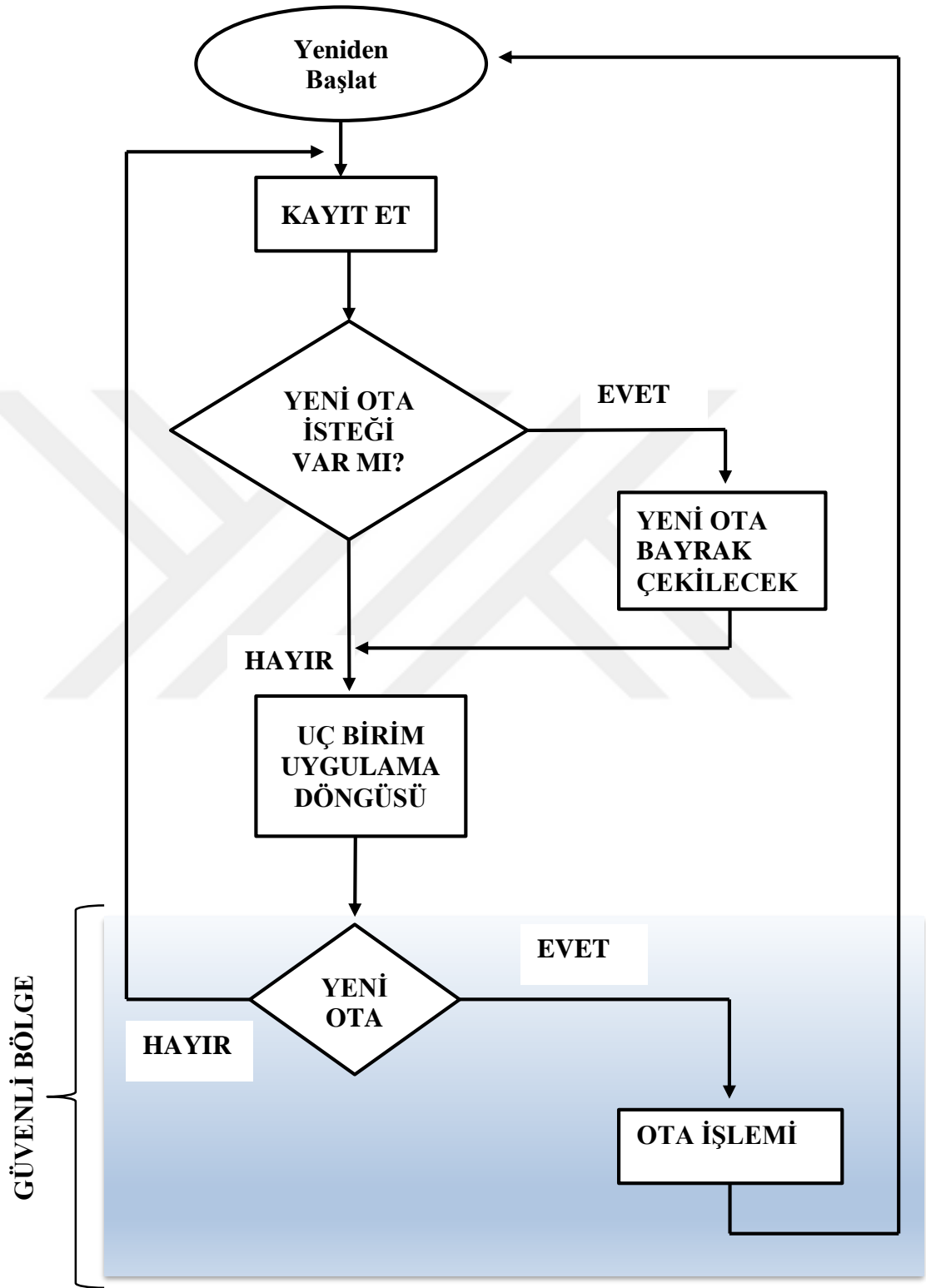
Bu sistemde, kullanıcılar uzaktan sisteme erişebilmektedir. Kullanıcıların doğrudan IoT yönetime müdahale edebilmesi güvenlik nedeniyle istenmez. Sisteme çok fazla cihaz bağlanması durumunda, bu cihazlara tek tek elle program yüklemek iş yükünü artırır ve zaman kaybına neden olur. Belki de bu cihazlara yüklenmesi

gereken programların eş zamanlı yüklenmesi gerekebilir. Parça parça programlar büyük bir sistemi oluşturabilir.

4.11. Uç birimde OTA sağlanması

Şekil 4.22'de verilen, uç birim uygulama programı bir sonsuz döngü içinde çalışır. Uygulamanın kesintiye uğramaması için OTA işlemleri uç birim uygulama programının izin verdiği güvenli bir bölgede yürütülür.

Eğer bir OTA işlemi bir başka deyişle yeni bir program yüklenmesi gerçekleşecek ise uç birim uygulama programının gereken önlemlerini örneğin aktif çıkışlarının pasife alınması ve verilerini almasına, saklamasına izin verilir. Uygulama programının izin verdiği aşamada OTA gerçekleşir ve sistem yeniden başlat noktasından (Reset) çalışmaya başlar.



Şekil 4.22 Uç Birimde OTA Gerçekleştirilmesi

V. UZAKTAN YÖNETİM SİSTEMİ ÜZERİNE BİR UYGULAMA

Burada Tümüleşik sisteme benzer olarak PHP ve MySQL tabanında uygulanması gerçekleştirilmektedir. Bu seçim; Uzaktan Yönetim Sisteminin bir PC üzerinde gerçekleştirilmesi sağlandığı gibi kolaylıkla Bulut üzerine alınarak Web Hizmeti Sunucuları üzerinde de gerçekleştirilmesine olanak sağlamaktadır. Çalışma öncelikle PC üzerinde Uzaktan Yönetim Sisteminin gerçekleştirilmesi üzerinde tamamlanmıştır. Daha sonra bir Web Sunucu üzerine yazılımlar taşınarak yönetim sisteminin çalışması gözlenmiştir.

5.1. Önerilen uzaktan yönetim sisteminin çalışma prensibi

Önerilen uzaktan yönetim sisteminin çalışma prensibi Şekil 5.1'de gösterilmektedir. Bu sistem çalışmaya başladığında önce, IoT tarafında Wi-Fi bağlantısı fonksiyonu çalışır. Wi-Fi bağlantısı yoksa tekrar başa döner. Eğer Wi-Fi bağlantısı var ise `iot_oku` fonksiyonu çalışır. `iot_oku` fonksiyonu bulut üzerindeki `iot_tablosu` ile bağlantılıdır. `iot_oku` fonksiyonu, buluta istek gönderir ve cevabını alır. Daha sonra `iot_oku` fonksiyonun işleri bitince kayıt varsa ve versiyon farklıysa `iot_guncelle` fonksiyonu çalışır. Eğer güncelleme tamamsa cihaz resete gider ve sonra tekrar sistemin çalışması başa döner.

Sistemin çalışmasındaki diğer bir yol ise, sistem çalışmaya başladığında Wi-Fi bağlantısı varsa, `iot_oku` fonksiyonu işini bitirince eğer kayıt yoksa `iot_yaz` fonksiyonu çalışır. `iot_yaz` fonksiyonu işlerini bitirince Wi-Fi bağlantısı var mı

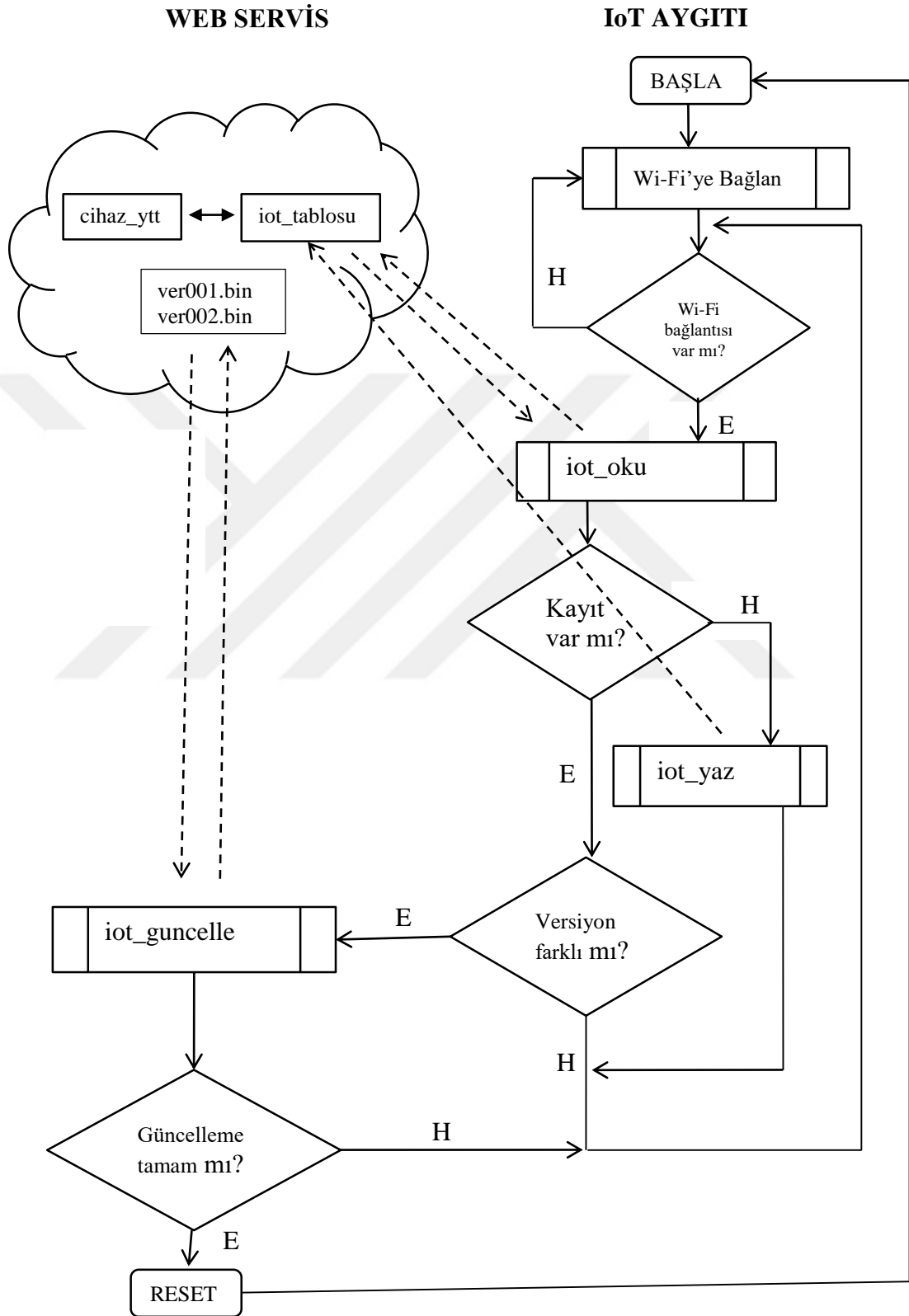
noktasına geri döner. Sistemin çalışmasındaki diğer tüm yollar şekil üzerinde gösterilmektedir.

Bu sistemde, `iot_oku.php`, `iot_yaz.php` ve `iot_guncelle.php` dosyaları bulut üzerinde bulunmaktadır ve PHP dilinde yazılmıştır. Aynı zamanda `ver001.bin` ve `ver002.bin` uzantılı dosyalarımız da bulut üzerinde bulunmaktadır. `iot_guncelleme` fonksiyonu ile bulut üzerinde bulunan versiyon cihaz üzerindeki farklı ise güncelleme başarılı şekilde yapılabilmektedir.

Aşağıdaki **Tablo 5.1**'de sistemde kullanılan tetikleyiciler ve SQL sorguları gösterilmektedir.

TETİKLEYİCİLER	SQL KOD
istekEkle	INSERT INTO cihaz_ytt(ip, iot_adi, ver, yeniver, Date, Time, durum, TimeStamp, islem) VALUES (NEW.ip, NEW.iot_adi, NEW.ver, NEW.yeniver, NEW.Date, NEW.Time, NEW.durum, NEW.TimeStamp, 'EKLE')
istekGuncelle	INSERT INTO cihaz_ytt(ip, iot_adi, ver, yeniver, Date, Time, durum, TimeStamp, islem) VALUES (NEW.ip, NEW.iot_adi, NEW.ver, NEW.yeniver, NEW.Date, NEW.Time, NEW.durum, NEW.TimeStamp, 'GÜNCELLE')
istekSil	INSERT INTO cihaz_ytt(ip, iot_adi, ver, yeniver, Date, Time, durum, TimeStamp, islem) VALUES (OLD.ip, OLD.iot_adi, OLD.ver, OLD.yeniver, OLD.Date, OLD.Time, OLD.durum, OLD.TimeStamp, 'SİL')

Tablo 5.1.Tetikleyiciler



Şekil 5.1 Önerilen Uzaktan Yönetim Sisteminin Çalışma Prensibi

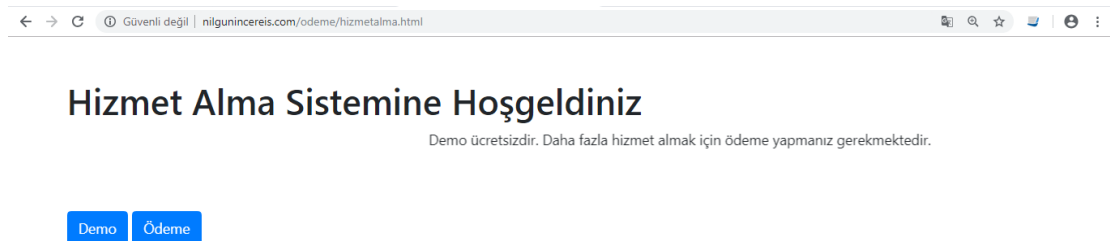
5.2. Önerilen uzaktan yönetim sistemindeki kullanıcı ekranı

Bu sistemde kullanıcılar, uzaktan bulut üzerinde bulunan istediği dosyayı cihazına (Bu sistem NodeMCU'ya göre tasarlanmıştır.) uzaktan yükleyebilir ve kullanıcılar sistemini uzaktan takip edebilir veya bulut ortamına istediği yazılım dosyasını (OTA hizmeti bulunan .bin dosyası) gönderebilir ve isterse sisteme yüklediği bu dosyayı cihazına uzaktan yükleyebilir. Bu sistem bu tür hizmet almak isteyenler için hazırlanmış bir uzaktan yönetim sisteminin hizmet yazılımıdır.

Şekil 5.2 ve Şekil 5.3'te kullanıcı ekranları oluşturulmuştur. Sisteme kaydı olmayan sistemi hiç kullanamaz ya da sistemin sınırlı özelliklerini kullanabilir.



Şekil 5.2 Kullanıcı Kayıt ve Giriş Ekranı



Şekil 5.3 Hizmet Alma Sistemi Ödeme Ekranı

Şekil 5.4 ve Şekil 5.5'te sırasıyla kayıt olma formu ve demo girişi gösterilir.

Hoşgeldiniz
Kayıt olduktan

Üye Giriş Kayıt Ol

Hizmet Yönetimine Kayıt Olma

Ad Soyad
nilgün incereis

Mail Adresi
nilgun.incereis@okan.edu.tr

Telefon
05325437948

Şirket Adı
ni

Adres
İstanbul

Şifreniz

Kayıt Ol Kapat

Şekil 5.4 Hizmet Yönetimine Kayıt Formu

Hizmet Alma Sistemi
deme yapmanız gerekmektedir.

Demo Ödeme

Demo Bilgi Girişi

Hizmet Ad

Hizmet Türü

Demo Giriş Yap Kapat

Şekil 5.5 Demo Girişi

Şekil 5.6'de ödeme bilgisi gösterilir.

The image shows a web browser window with a payment form. The form is titled "Ödeme Bilgi Girişi" and has a close button (X). It contains the following fields: "Hizmet Ad" (text input), "Hizmet Zamanı" (date input), "Hizmet Türü" (text input), and "Hizmetmet Fiyatı" (number input with a dropdown arrow). At the bottom of the form are two buttons: "Ödeme Yap" (green) and "Kapat" (red). The background of the browser shows a page titled "Hizmet Alma Sist" with "Demo" and "Ödeme" buttons. A "Demo ücretsizdir." message is visible on the right side of the page.

Şekil 5.6 Ödeme Bilgi Girişi Formu

5.3. IoT cihaz yönetim paneli

Kullanıcılar veya yöneticiler, Şekil 5.7’da IoT cihaz yönetim panelinden sisteme bağlı cihazları görebilir.

The image shows a web browser window with the title "IOT Cihaz Yönetim Paneline Hoşgeldiniz". Below the title, there is a link "Sisteme Bağlı Cihazları Gör" and a button "ÇIKIŞ YAP". The browser address bar shows "nilgunincereis.com/iot/welcome.php".

Şekil 5.7 IoT Cihaz Yönetim Paneli

5.4. IoT cihazların durum takibi

IoT cihazların takibi, Şekil 5.8.a ve Şekil 5.8.b’de görülür. Cihazlar sisteme hazır duruma getirildikten sonra bu sisteme bağlandığında cihazların bilgileri veri tabanına otomatik olarak kaydedilir. Şekil 5.8.a’da görüldüğü gibi cihazın IP adresi, MAC adresi, üzerindeki versiyon adı, tarih, zaman ve durum bilgileri vardır.

SR.NO.	IP ADRES	MAC ADRES	VERSİYON	YENİ VERSİYON	TARİH	ZAMAN	DURUM
10	192.168.137.100	84:F3:EB:89:67:22	ver002		2019-03-26	02:56:19	AKTİF

Şekil 5.8.a Cihaz Takip Ekranı 1

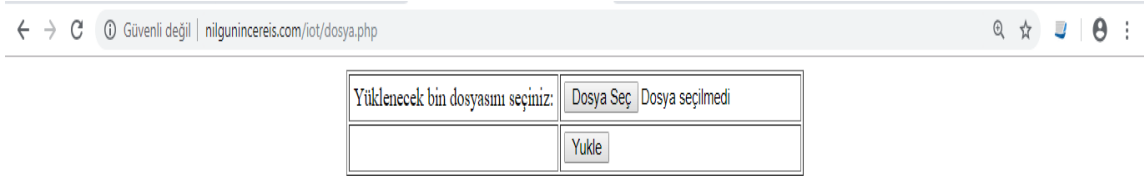
SR.NO.	IP ADRES	MAC ADRES	VERSİYON	YENİ VERSİYON	TARİH	ZAMAN	DURUM
10	192.168.137.100	84:F3:EB:89:67:22	ver002	ver001	2019-03-26	02:57:00	AKTİF

Şekil 5.8.b Cihaz Takip Ekranı 2

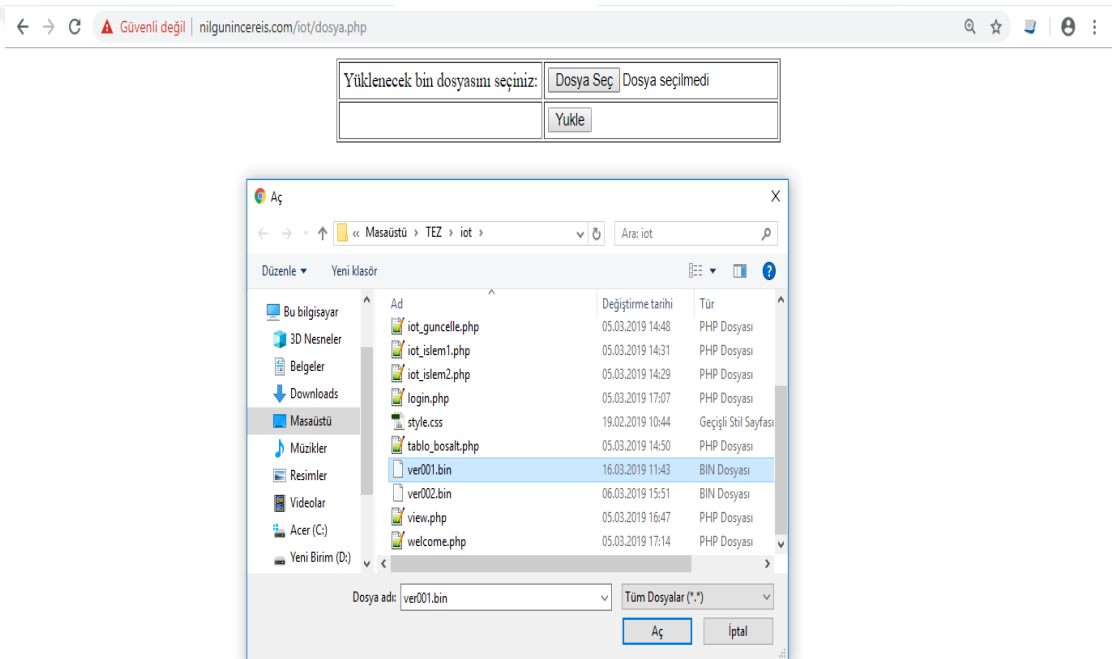
Şekil 5.8.b'de cihaz için yeni versiyon talebi yapılmıştır ve belli bir süre sonra sistem yeni versiyonu cihaza yükleyinceye kadar tekrar tekrar çalışacaktır. Eğer yeni versiyon talebi verildikten sonra internet bağlantısını keserseniz ve sonra cihazı tekrar sisteme bağlarsanız cihaz kaldığı yerden yeni versiyon talebini yükleyinceye kadar tekrar tekrar çalışacaktır.

5.5. Kullanıcıların sisteme dosya yükleyebilme ekranı

Şekil 5.9 ve **Şekil 5.10**'da kullanıcıların sisteme dosya yükleyebilecekleri ekranlar oluşturulmuştur ve nasıl yükleme yapılacağı görülmektedir.



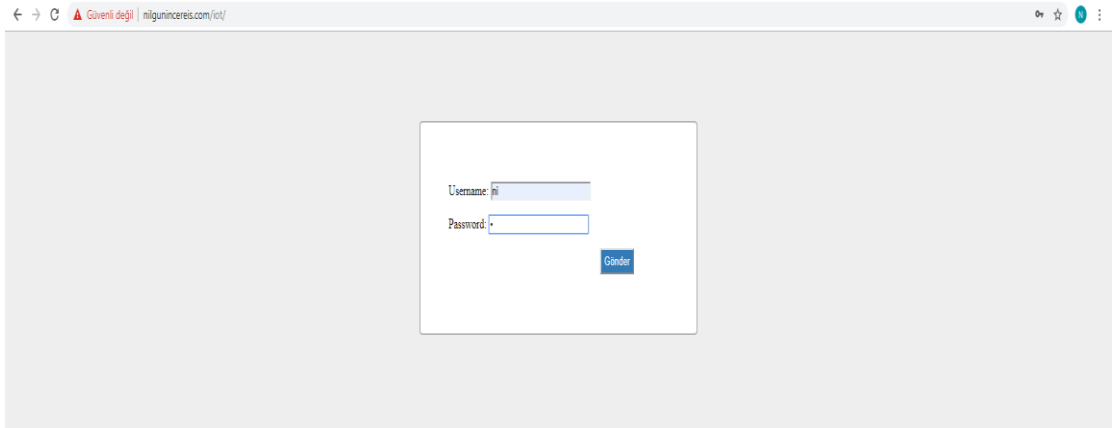
Şekil 5.9. Dosya Yükleme Ekranı



Şekil 5.10. Dosya Seçme Durumu

5.6. Yönetici ekranı

Şekil 5.11' da yönetici ekranı görülmektedir.



Şekil 5.11. Yönetici Ekranı

Yönetici sisteme bağlı olan tüm cihazları görebilir. Aşağıdaki **Şekil 5.12**'de 3 adet NodeMCU cihazı bağlanmış ve sistemde bu cihazların aktif olduğu görülür.

SR.NO.	IP ADRES	MAC ADRES	VERSİYON	YENİ VERSİYON	TARİH	ZAMAN	DURUM
3	192.168.137.104	A0:20:A6:0E:0F:AF	ver002		2019-03-22	11:55:43	AKTİF
2	192.168.137.59	84:F3:EB:89:67:22	ver002		2019-03-22	11:55:42	AKTİF
1	192.168.137.6	68:C6:3A:C3:6D:31	ver001		2019-03-22	11:55:42	AKTİF

Şekil 5.12. Sistemdeki Cihazlar

Şekil 5.13'te ise cihazların versiyonlarını değiştirebileceğimiz ekran görülmektedir.

Sr.No.	IP ADRES	IOT ADI	VERSİYON	YENİ VERSİYON	Date	Time
3	192.168.137.104	A0:20:A6:0E:0F:AF	ver002	<input type="text" value="ver001"/> <input type="button" value="Submit"/>	2019-03-22	11:56:34

Şekil 5.13. Versiyon Değiştirme Ekranı

SR.NO.	IP ADRES	MAC ADRES	VERSİYON	YENİ VERSİYON	TARİH	ZAMAN	DURUM
3	192.168.137.104	A0:20:A6:0E:0F:AF	ver002	ver001	2019-03-22	11:57:25	AKTİF
2	192.168.137.59	84:F3:EB:89:67:22	ver002	ver001	2019-03-22	11:57:25	AKTİF
1	192.168.137.6	68:C6:3A:C3:6D:31	ver001	ver001	2019-03-22	11:57:25	AKTİF

SR.NO.	IP ADRES	MAC ADRES	VERSİYON	YENİ VERSİYON	TARİH	ZAMAN	DURUM
3	192.168.137.104	A0:20:A6:0E:0F:AF	ver001	ver002	2019-03-22	11:59:08	AKTİF
2	192.168.137.59	84:F3:EB:89:67:22	ver002	ver001	2019-03-22	11:59:07	AKTİF
1	192.168.137.6	68:C6:3A:C3:6D:31	ver001	ver001	2019-03-22	11:59:07	AKTİF

SR.NO.	IP ADRES	MAC ADRES	VERSİYON	YENİ VERSİYON	TARİH	ZAMAN	DURUM
4	192.168.137.104	A0:20:A6:0E:0F:AF	ver002	ver001	2019-03-22	12:01:18	AKTİF
2	192.168.137.59	84:F3:EB:89:67:22	ver002	ver001	2019-03-22	12:01:10	AKTİF
1	192.168.137.6	68:C6:3A:C3:6D:31	ver001	ver001	2019-03-22	12:01:10	AKTİF

SR.NO.	IP ADRES	MAC ADRES	VERSİYON	YENİ VERSİYON	TARİH	ZAMAN	DURUM
4	192.168.137.104	A0:20:A6:0E:0F:AF	ver002	ver001	2019-03-22	12:07:46	AKTİF
2	192.168.137.59	84:F3:EB:89:67:22	ver002	ver001	2019-03-22	12:07:38	PASİF
1	192.168.137.6	68:C6:3A:C3:6D:31	ver001	ver001	2019-03-22	12:07:38	PASİF

Şekil 5.14 Sistemi Test Etmek için Farklı Deneme Durumları

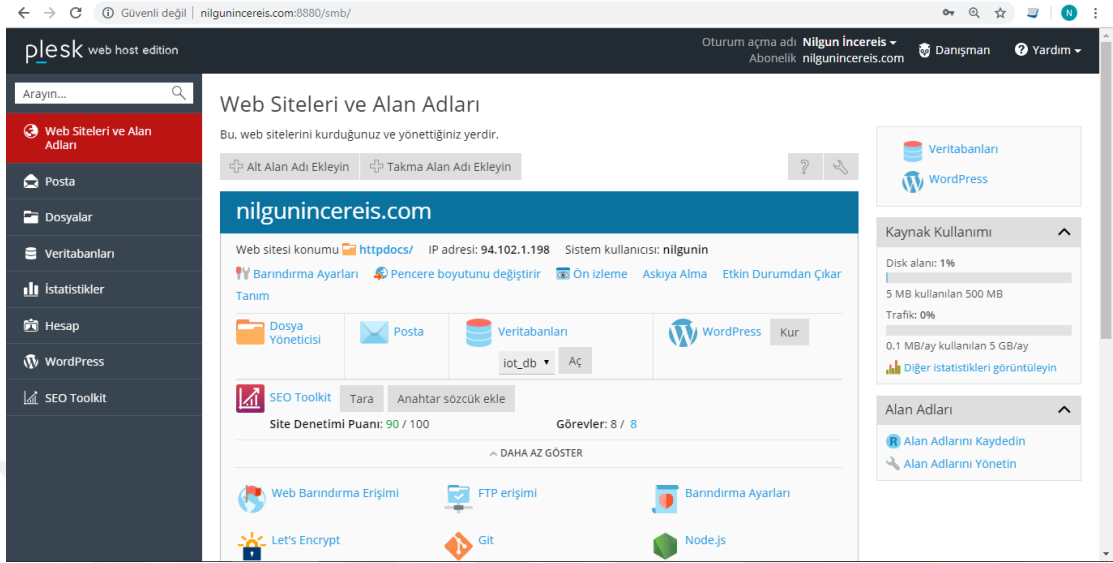
Yerel yönetim/uzaktan erişim sisteminde, IoT yönetimi PC üzerinde bulunmaktadır. Kullanıcılar PC'ye bağlı uygulamalar ya da programlar vasıtasıyla isteklerini IoT yönetimine iletirler. PC yerine RaspberryPi gibi cihazlar kullanılarak IoT yönetimi işlemleri daha küçük ve maliyeti daha az olan cihazlar da kullanılabilir. Aynı zamanda birden çok PC kullanmak yerine RaspberryPi kullanılması hizmet almak isteyen kişilerin ya da şirketlerin maliyetini düşürür.

5.7. Uzaktan yönetim sisteminin bir web sunucusuna taşınması

Bu çalışmada nilguninceeis.com alan adı, web sunucu hizmeti “www.ekonomikhost.net” [33] adresinden sağlanmıştır. Öncelikle alınan hizmet karşılığında verilen http://nilgunincereis.com:8880 adresinden, **Şekil 5.15**'teki Plesk web sunucu paneline [34] girilerek, kullanıcı adı ve şifre ile giriş yapılmaktadır.

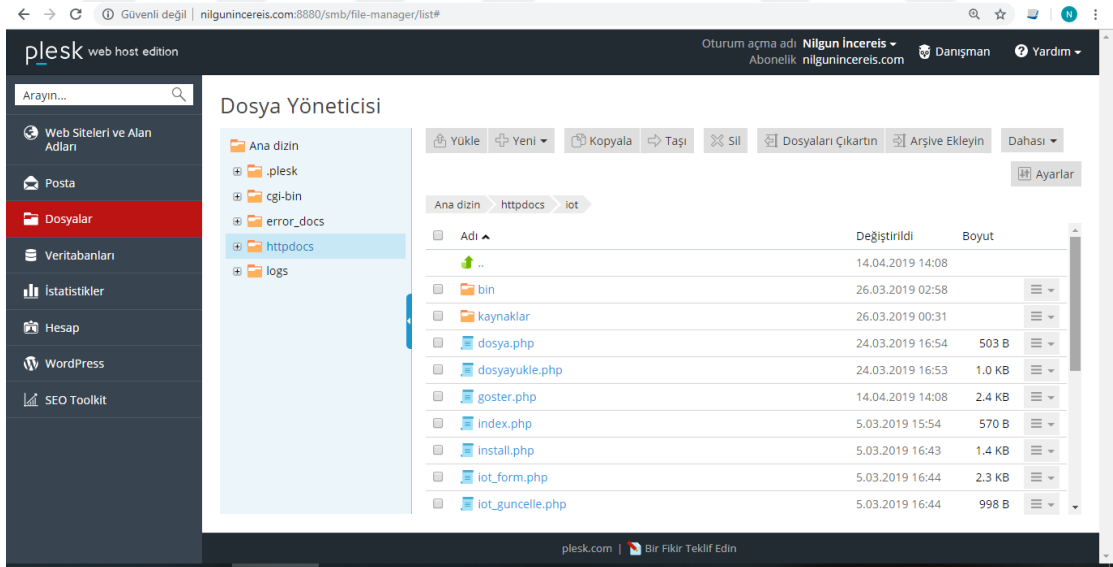
Şekil 5.15. Panel Girişi

Uzaktan yönetim sisteminin bir web sunucusuna taşınması için **Şekil 5.16**'da sol tarafta görülen “Dosyalar” kısmına girilmektedir.



Şekil 5.16. Web Sunucu Ortamı

Şekil 5.17'deki httpdocs klasörüne, dosyalar (örneğin php dosyaları) yüklenmektedir.



Şekil 5.17. Dosyalar

Şekil 5.18'de ise Veritabanları kısmından veritabanı ve tablolar oluşturulmaktadır.

Veritabanları

Burada yeni veritabanları oluşturabilir veya varolanları yönetebilirsiniz.

Veritabanı Ekle

Kullanıcı Yönetimi

iot_db nilgunincereis.com ile ilgili Değiştir phpMyAdmin

Ana bilgisayar: localhost:3306 (MySQL) Kullanıcı Sayısı: nldb Tablolar: 4 Boyut: 288 KB

DAHA AZ GÖSTER

phpMyAdmin Bağlantı Bilgisi Kopyala

Dokümü Dışarı Aktar Dokümü İçeri Aktar Denetle ve Onar

Veritabanını sil

iot_db2 Bu veritabanını bir siteye ata

Ana bilgisayar: localhost:3306 (MySQL) Veritabanı kullanıcıları yok [Oluştur] Tablolar: 4 Boyut: 112 KB

DAHA AZ GÖSTER

Şekil 5.18. Veritabanı Ortamı

VI. SONUÇLAR

Uzaktan program yükleme hizmeti sistemi, sahadaki IoT uç birimlere, yeni program yükleme işlemlerini firma adına güvenli bir şekilde yerine getiren bir sistemdir.

Bu çalışmada, bir uzaktan program yükleme hizmeti tasarımı ve uygulanması gerçekleştirilmiştir. Bu çalışmanın sistem tasarımı için, genel IoT sistemleri, yerel sistem, yerel yönetim/uzaktan erişim sistemi, uzaktan yönetim sistemi ve bulut üzerinden dağıtılmış yönetim sistemi olarak incelenmiştir. Uzaktan program yükleme hizmeti sistemi için bir öneri yapılmış ve uygulanmıştır. Bu çalışmada, 1. aşama olarak, öncelikle bir yerel yönetim/uzaktan erişim sistemi üzerinde bir uygulama yapılmıştır. Bu yönetim tümleşik yönetim olarak adlandırılmıştır. Daha sonra 2. Aşamada önerilen sistem, iki yönetimli sistem olarak dağıtılmaya çalışılmıştır.

Sistemi geliştirmek amacıyla senaryo durumları oluşturulmuştur. Bunlar, kullanıcıların sisteme kayıt olma, firma kaydı yapma ve hizmet alma durumlarına göre, sisteme veya cihaza program yükleme, cihazların sisteme bağlı olma durumlarına göre karar verme yapısı, belli aralıklarla cihazların sistemde aktif veya pasif olması durumuna göre gözetlenebilmesi, güncelleme için güvenlik önlemleri şeklindedir. Kullanıcılar sistemi kullanabilmek için önce sisteme kaydını yapmalıdır. Daha sonra kullanıcıya ait firma kaydı ve hizmet alma işlemleri yapılır.

Oluşturulan sistemde, uç birimler için NodeMcu cihazları kullanılmıştır. ESP8266, WiFi, IoT, OTA, Bulut teknolojilerini içeren mekân ve zamandan bağımsız bir yönetim sistemi oluşturularak ve bu sistem, gerçek zamanlı çalışabilmektedir.

İleri bir çalışma olarak; yönetimlerin birer bulut hizmeti yazılımına dönüştürülerek tamamen bulut üzerinde dağıtık bir sistem olarak gerçekleştirilmesi hedeflenmiştir.



VII. KAYNAKLAR

- [1] Liu Y., Zhou G., “Key Technologies and Applications of Internet of Things”, 2012 Fifth International Conference on Intelligent Computation Technology and Automation, IEEE, 2012.
- [2] Hassan Q. F., “Introduction to the Internet of Things”, Wiley-IEEE Press eBook Chapters, Online ISBN: 9781119456735, 2018.
- [3] Xu L. D., He W., Li S., “Internet of Things in Industries: A Survey”, IEEE Transactions On Industrial Informatics, Vol. 10, No. 4, 2014.
- [4] Kang K., Pang Z., Xu L. D., , Ma L., Wang C., “An Interactive Trust Model for Application Market of the Internet of Things”, IEEE Transactions On Industrial Informatics, VOL. 10, NO. 2, 2014.
- [5] Behura A., Narayan A., Ray A. K., Pani S. K., “A Complete Model for IOT Application”, Proceedings of the International Conference on Intelligent Sustainable Systems (ICISS 2017), ISBN:978-1-5386-1959-9, IEEE Xplore Compliant - Part Number:CFP17M19-ART, 2017.
- [6] Al-Qaseemi S. A., Almulhim H.A., Almulhim M.F., Chaudhry S.R., “IoT Architecture Challenges and Issues: Lack of Standardization”, Future Technologies Conference, San Francisco, United States, 2016.
- [7] Wang C., Daneshmand M., Dohler M., Mao X., Hu R. Q., Wang H., "Guest editorial - special issue on Internet of things (IoT): Architecture protocols and services", IEEE Sensors Journal, vol. 13, no. 10, pp. 35053510, 2013.
- [8] Zdravkovic M., Trajanovic M., Sarraipa J., Jardim-Gonçalves R., Lezoche M., “Survey of Internet-of-Things platforms”, 6th International Conference on

Information Society and Technology, ICIST 2016, Kopaonik, Serbia, pp. 216-220, 2016.

[9] Rajalakshmi A., Shahnasser H., “Internet of Things using Node-Red and Alexa”, 2017 17th International Symposium on Communications and Information Technologies (ISCIT), 2017.

[10] Dey S., Roy A., Das S., “Home automation using Internet of Things”, Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), IEEE Annual, 2016.

[11] Cuijuan G., Changyun M., Zhigang W., Lina X. , “Research and Implementing of Remote Update for Network Telephone System Program Based on Ethernet”, 2010 2nd International Conference on Signal Processing Systems (ICSPS), 2010.

[12] Tarneberg W., Chandrasekaran V., Humphrey M., “Experiences creating a framework for smart traffic control using AWS IOT”, IEEE/ACM 9th International Conference on Utility and Cloud Computing, 2016.

[13] Kang D. H., Park M. S., Kim H. S., Kim D.Y., Kim S.H., “Room temperature control and fire alarm/suppression IoT service using MQTT on AWS”, International Conference on Platform Technology and Service, 2017.

[14] He H., Ma Z. H., Li X., Chen H., Shao W., “An approach to estimating cost of running cloud applications based on AWS”, 19th Asia-Pacific Software Engineering Conference, 2012.

[15] <https://www.businesswire.com/news/home/20160222006280/en/Dell-Enters-Embedded-PC-Market-New-Embedded>, "Dell Enters Embedded PC Market with

- New Embedded Box PCs, Helping Smart Systems Connect to the Internet of Things", 24.03. 2019.
- [16] Akyildiz I.F., Su W., Sankarasubramaniam Y., Cayirci E., "Wireless sensor networks: a survey", Elsevier, Computer Networks 38 (2002) 393–422, 2002.
- [17] <https://media.readthedocs.org/pdf/arduino-esp8266/latest/arduino-esp8266.pdf>, Ivan Grokhotkov, ESP8266 Arduino Core Documentation Release 2.5.0-38-g95cf925, Feb 26, 2019.
- [18] https://arduino-test.esp8266.com/Arduino/versions/2.0.0/doc/ota_updates/ota_updates.html, ESP8266 Arduino Core, 26.03.2019.
- [19] ÖZTÜRK E., "Nesnelerin İnterneti İçin Genel Amaçlı Yeni Bir Mimari Modelin Önerilmesi", Yüksek Lisans Tezi, Bilgisayar Mühendisliği Anabilim Dalı, Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü, 2018.
- [20] Chekired D. A., Khoukhi L., Mouftah H. T., "Industrial IoT Data Scheduling based on Hierarchical Fog Computing: A key for Enabling Smart Factory", Citation information: DOI 10.1109/TII.2018.2843802, IEEE Transactions on Industrial Informatics, 2018.
- [21] Al-Fuqaha A., Guizani M., Mohammadi M., Aledhari M., Ayyash M., "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications", IEEE Communication Surveys & Tutorials, Vol. 17, No. 4, 2015.
- [22] Gigli M., Koo S., "Internet of Things: Services and Applications Categorization", Advances in Internet of Things, 1, 27-31, 2011.

- [23] Li S., Oikonomou G., Tryfonas T., Chen T. M., Xu L. D., “A Distributed Consensus Algorithm for Decision Making in Service-Oriented Internet of Things”, IEEE Trans Ind. Informat., [Online]. Available: doi: 10.1109/TII.2014. 2306331 in press. 2014.
- [24] Li L., Li S., Zhao S. , “QoS-Aware Scheduling of Services-Oriented Internet of Things”, IEEE Transactions On Industrial Informatics, Vol. 10, No. 2, 2014.
- [25] Erken E., “Nesnelerin İnterneti Tabanlı Bebek Uyku Takip Sistemi Tasarımı ve Gerçeklenmesi”, Kocaeli Üniversitesi Fen Bilimleri Enstitüsü, Elektronik ve Haberleşme Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi, 2018.
- [26] Wortmann F., Fluchter K., “Internet of Things - Technology and Value Added”, Business & Information Systems Engineering, 57, 221-224, 2015.
- [27] Guizani M., Mohammadi M., Aledhari M., Ayyash M., “Internet of Things: a survey on enabling technologies, protocols, and applications”, IEEE Communications Surveys & Tutorials, DOI: 10.1109/COMST.2015. 2444095, 2015.
- [28] Yu W., Liang F., He X., Hatcher W. G., Lu C., Lin J., Yang X., “A Survey on the Edge Computing for the Internet of Things”, IEEE Journals & Magazines, Volume:6, Pages:6900-6919, 2018.
- [29] Çetintav I., “Nesnelerin İnterneti Uygulamalarında Aes Blok Şifresinin Yazılımsal ve Donanımsal Performanslarının Karşılaştırılması”, Trakya Üniversitesi, Yüksek Lisans Tezi, 2018.
- [30] <https://www.endustri40.com/bulut-bilisim-cloud-computing-nedir/>, 26.03.2019.
- [31] <https://docs.aws.amazon.com/iot/latest/developerguide/what-is-aws-iot.html>, Amazon Web Services, 2018.

[32] Tao, F., Zuo, Y., Xu L. D., Zhang, L., “IoT-Based Intelligent Perception and Access of Manufacturing Resource Toward Cloud Manufacturing”, IEEE Transactions On Industrial Informatics, Vol. 10, No., 2014.

[33] www.ekonomikhost.net, Ekonomikhost İnternet ve Bilişim Hizmetleri, 2019.

[34] <https://www.plesk.com>, Plesk International GmbH, 2019.

