



**T.C. İSTANBUL T CARET
 NİVERSİTESİ**

FEN B LİMLERİ ENSTİT S 

**IOT TEKNOLOJİSİ KULLANAN PRATİK VE G VENİLİR
AKILLI KAPI KİLİDİ TASARIMI**

Ay eg l KALYONCU

**Dan şman
Dr.  ğr.  yesi Metin TURAN**

**Y KSEK LİSANS TEZİ
B LGİSAYAR M HENDİSLİĐİ ANAB LİM DALI
İSTANBUL – 2020**

KABUL VE ONAY SAYFASI

Ayşegül KALYONCU tarafından hazırlanan "**IoT Teknolojisi Kullanan Pratik ve Güvenilir Akıllı Kapı Kilidi Tasarımı**" adlı tez çalışması 08/09/2020 tarihinde aşağıdaki jüri üyeleri önünde başarı ile savunularak, İstanbul Ticaret Üniversitesi Fen Bilimleri Enstitüsü **Bilgisayar Mühendisliği Anabilim Dalı'nda Yüksek Lisans Tezi** olarak kabul edilmiştir.

Danışman **Dr. Öğrt. Üyesi Metin TURAN**
İstanbul Ticaret Üniversitesi

Jüri Üyesi **Dr. Öğr. Üyesi Arzu KAKIŞIM**
İstanbul Ticaret Üniversitesi

Jüri Üyesi **Dr. Öğr. Üyesi Ertuğrul ÇETİNSOY**
İstanbul Şehir Üniversitesi

Onay Tarihi: 18.09.2020

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsünün 18.09.2020 tarih ve 2020/290 numaralı Yönetim Kurulu Kararının 2. maddesi gereğince, ders yüklerini ve tez yükümlülüğünü yerine getirdiği belirlenen "Ayşegül KALYONCU" (TC:38980989992) adlı öğrencinin mezun olmasına oy birliği ile karar verilmiştir.

Prof. Dr. Necip ŞİMŞEK
Enstitü Müdürü

AKADEMİK VE ETİK KURALLARA UYGUNLUK BEYANI

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

18/09/2020



Ayşegül KALYONCU

İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER	i
ÖZET.....	ii
ABSTRACT.....	iii
TEŞEKKÜR.....	iv
ŞEKİLLER.....	v
SİMGELER VE KISALTMALAR.....	vi
1.GİRİŞ	1
2. LİTERATÜR ÖZETİ.....	3
3. NESNELERİN İNTERNETİ	9
3.1. Güvenlik	10
3.2. IoT Uygulama Alanları.....	10
3.2.1. Akıllı sağlık	10
3.2.2. Akıllı şehir	11
3.2.3. Endüstriyel alan	11
3.2.4. Akıllı tarım.....	12
3.2.5. Akıllı ev	12
4. MATERYAL VE YÖNTEM	14
4.1. Raspbberri Pi 3 B+	14
4.1.1. Raspbberri Pi 3 B+ Donanım Özellikleri.....	14
4.1.2. İşletim Sistemi	15
4.2. Usb Kamera.....	16
4.3. Röle Devresi	17
4.4. Andorid Studio	17
4.5. OpenCV.....	18
4.6. OpenVpn	19
5. ARAŞTIRMA BULGULARI VE TARTIŞMA	20
6.SONUÇ VE ÖNERİLER	27
KAYNAKLAR	31
EKLER.....	35
EK A. Kodlar.....	36
ÖZGEÇMİŞ	48

ÖZET

Yüksek Lisans Tezi

IOT TEKNOLOJİSİNİ KULLANAN PRATİK VE GÜVENİLİR AKILLI KAPI KİLİDİ TASARIMI

Ayşegül KALYONCU

**İstanbul Ticaret Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı**

Danışman: Dr. Öğr. Üyesi Metin TURAN

2020, 48 sayfa

Teknolojinin hızla gelişmesi ve internetin yaygınlaşması, ayrıca modern yaşamın ortaya çıkardığı ihtiyaçlar dolayısıyla Nesnelerin İnterneti (IoT)'ne olan ilgi gün geçtikçe artmaktadır. IoT artık yaygın olarak kullandığımız, sürekli duyduğumuz bir kavram olmuştur. "Akıllı sistemler" olarak adlandırdığımız IoT uygulamaları, ev, tarım ve sağlık alanları başta olmak üzere neredeyse birçok alanda karşımıza çıkmaktadır. Bu çalışmada "Akıllı Ev" sisteminin bir parçası olarak algılanabilecek, kapı kontrol IoT uygulaması geliştirilmiştir. Projenin motivasyonunu, sık sık karşılaşılan bir durum olan evin dış kapısında kalan birisinin (çocuk, bakıcı), uzak konumdaki ev sahibine durumunu haberdar ederek, güvenli bir biçimde IoT teknolojisi ile kapının açılabilmesi oluşturmaktadır. Süreç, kapıdaki kişinin telefonunda bulunan Android uygulama ile ev sahibine (genellikle ebeveyn) kapının açılması için istekte bulunması ile başlar. Bu istek, Raspberry Pi üzerindeki PHP Web servisinin uyarılması ve Python ile kodlanmış yazılım kullanılarak kamera görüntüsünün ev sahibine mail olarak gönderilmesi ile devam eder. Nihai olarak, ev sahibinin telefonunda olan diğer Android uygulama ile onay verilirse Raspberry Pi üzerindeki diğer PHP Web servisinin uyarılması sonucu elektrikli kapı kilidine yeterli gerilim verilerek kapının açılması ile sonlanır. Özellikle, düşük maliyetli ve güvenlik ihtiyaçları gözetilerek geliştirilmiş, pratik olarak uygulanabilir bir çalışmadır.

Anahtar Kelimeler: Akıllı ev uygulamaları, nesnelerin interneti, Raspberry Pi.

ABSTRACT

M.Sc. Thesis

PRACTICAL AND RELIABLE SMART DOOR LOCK DESIGN USING IOT TECHNOLOGY

Ayşegül KALYONCU

**İstanbul Commerce University
Graduate School of Applied and Natural Sciences
Department of Computer Engineering**

Supervisor: Assist. Prof. Dr. Metin TURAN

2020, 48 pages

Due to the rapid development of technology and the widespread of the internet, and the needs of modern life, the interest in the Internet of Things (IoT) is increasing day by day. IoT is now a concept that we use extensively and that we constantly hear. IoT applications, which we call "smart systems", appear in many areas, especially in the fields of home, agriculture and health. In this study, door control IoT application, which can be perceived as a part of "Smart Home" system, has been developed. The motivation of the project is the fact that someone (child, childminer) who is at the outer door of the house, which is a frequent situation, can be opened safely with IoT technology by informing the home owner in the remote location. The process starts with the request by the cell phone of the person on the door using installed Android application to the homeowner (generally parent) to open the door. This request continues by activating the PHP Web service on Raspberry Pi and sending the camera image to the homeowner via an e-mail, using Python-encoded software. Ultimately, if it is approved by the other Android application on the homeowner's phone, it will end with the opening of the door by giving sufficient voltage to the electric door lock as a result of the activation of the other PHP Web service on the Raspberry Pi. In particular, it is a practically feasible study, developed with low cost and security needs in mind.

Keywords: Internet of things, Raspberry Pi, smart home application.

TEŞEKKÜR

Bu çalışmanın gerçekleştirilmesinde, değerli bilgilerini benimle paylaşan, kendisine ne zaman danışsam bana kıymetli zamanını ayırıp sabırla ve büyük bir ilgiyle bana faydalı olabilmek için elinden gelenden fazlasını sunan her sorun yaşadığımda yanına çekinmeden gidebildiğim, güler yüzünü ve samimiyetini benden esirgemeyen kıymetli ve danışman hoca statüsünü hakkıyla yerine getiren Dr. Öğr. üyesi Metin TURAN'a teşekkürü bir borç biliyor ve şükranlarımı sunuyorum.

Hayatımın her döneminde yanımda olan aileme teşekkür ederim.

Ayşegül KALYONCU
İSTANBUL, 2020

ŞEKİLLER

	Sayfa
Şekil 1.1. Nesnelerin İnterneti girdiği alanlar	1
Şekil 3.1. Kahve makinesinin görüntüsü	9
Şekil 3.2 Akıllı ev	12
Şekil 4.1. Raspberry Pi 3B+ model	14
Şekil 4.2. Raspberry Pi 3B+ model pin diyagramı	15
Şekil 4.3. Raspbian işletim sistemi masaüstü görüntüsü	16
Şekil 4.4. Usb kamera	16
Şekil 4.5. 1kanal 5v röle kartı	17
Şekil 4.6. Android studio ekran görüntüsü	18
Şekil 5.1. Sistemin genel yapısı	20
Şekil 5.2. Kapıyı Aç Android Uygulaması giriş paneli.....	22
Şekil 5.3. Kapı açılması için istek gönderme paneli	23
Şekil 5.4. kapıdaKaldim.php servisi	23
Şekil 5.5. Kapı önü kamera görüntüsü alma kodu ve e-posta gönderme kodu ..	24
Şekil 5.6. Raspberry Pi'ye gelen taleplerin değerlendirilmesi	24
Şekil 5.7. Kapı açılması için istek gönderme paneli	25
Şekil 5.8. kapiyiAc.php web servisi	25
Şekil 5.9. Kapı açma	26
Şekil 6.1. Önerilen sistem prototipi	27
Şekil 6.2. Tasarlanan sistemin şeması	28

SİMGELER VE KISALTMALAR

IoT Nesnelerin İnterneti

1.GİRİŞ

Endüstri 1.0'dan itibaren gelişen teknoloji günümüzde insanların yaşam biçimini şekillendirmeye başlamıştır. İnternet, artık vazgeçilmezlerimizin arasında yerini almış bulunmaktadır. İnternet sayesinde bankacılık, veri depolama, e-posta, alışveriş, görüntülü konuşma gibi birçok işlem gerçekleşmektedir.

İnternetin sınırlarının genişlemesi ve günlük hayatımızda kullandığımız birçok cihazın ve nesnenin kablosuz iletişim gibi yeni yetileri kazanmasıyla bu aygıtların İnternete bağlanması, Nesnelerin İnterneti (Internet of Things - IoT) kavramını ortaya çıkarmıştır.

Nesnelerin İnterneti, çevremizdeki nesnelerin fiziksel olaylarını kontrol etmemizi ve aynı zamanda bu olayları gözlemleyerek takip etmemizi, bunun neticesinde topladığımız verileri analiz etmemizi sağlayacak olan aygıt, yazılım ve erişim hizmetlerini kapsayan bir iletişim ağıdır. Bu fiziksel olaylar, üretim süreçleri, enerji şebekeleri, hasta takip sistemleri, geri dönüşüm süreçleri, taşımacılık, akıllı binalar, akıllı ev, akıllı kapı, akıllı tarım, akıllı sağlık vs. gibi alanlarda kullanılan nesnelerin birbirleriyle iletişime geçmesidir. Ayrıca bu ağda bulunan cihazlar ve algılayıcılar insan makine, makine-makine iletişimi kurabilen organizasyon yapısına sahiptir. (Şekil 1.1).



Şekil1.1. Nesnelerin interneti girdiği alanlar (IoT Türkiye 2020).

Nesnelerin İnterneti, günümüzde cihaza ait verilerin ağ üzerinden paylaşılması olarak tanımlanır. Böylesi büyük bir ağ ortamı farklı teknolojilerin ve platformların bir arada kullanılmasını gerektirmektedir. Bununla birlikte kurulacak olan sistem birçok problemi de beraberinde getirmektedir. Bu problemlerin başında da “güvenlik” unsuru gelmektedir.

Aman ve Anitha (2017)’ye göre Akıllı kapı, dijital kod, şifreler, tarama ve parmak izi taraması, göz taraması veya retina taraması gibi tanıma faktörleri temelinde çalışan bir yapıdır. Çalışmada ele alınan akıllı kapı modelinde fiziksel temas söz konusu değildir. Bunun yerine akıllı kapı uzaktan yetki kullanarak kumanda edilebilen (açılıp/kapatılabilen) bir sistemdir. Çalışmanın sınırları ise sadece kapının açılma özelliğini kullanmaya yönelik olarak belirlenmiştir.

Kapıda kalan bireyin (çocuk, yaşlı, bakıcı) akıllı cep telefonunda bulunan Android uygulama sayesinde kapı açılma isteğinde bulunabilmesi, ev sahibi konumunda bulunan yetişkin bireyin (anne, baba) akıllı cep telefonunda bulunan diğer bir Android uygulama sayesinde kapıyı açabilmesi üzerinedir. Çok katmanlı yapı ile kapı gibi önemli bir donanım üzerinde maksimum güvenlik sağlanmaya çalışılmıştır.

Bu tezde önerilen sistem, akıllı kapı sistemidir. Bir “akıllı kapı kilidi” modeli tasarlanmıştır. Bu problem, diğer birçok uygulama gibi hem veri hem de uygulama güvenliği ile yakından ilişkilidir. Veriler internet ortamında iletildiğinden ve karmaşık uygulama platformundan dolayı iki güvenlik ayağı da dikkatlice ele alınmalıdır. Motivasyon sağlayan problem, ev sahiplerinin, çocukların, yaşlıların anahtarlarını unutması durumuyla ilişkilidir. Ebeveynler uzak bir yerde, hatta şehir dışında bulunuyorsa problem daha da kronikleşmektedir. Gerçek hayat problemi olan bu duruma çözüm üretmek ihtiyacı, akıllı kapı sistemlerine olan ilgiyi daha da arttırmıştır.

2. LİTERATÜR ÖZETİ

Akıllı kapı sistemleri hakkında çok çeşitli araştırmalar yapılmıştır. Bunlardan Bluetooth teknolojisini kullanarak yapılan çalışmalara ilk örnek Rajadurai vd. (2015), tarafından yapılan çalışma verilebilir. Bu çalışmada, ARM denetleyicisine dayalı bir ev otomasyon tekniği geliştirmişlerdir. Bu teknik kişiyi tespit etmek için bir IR algılayıcısı kullanmaktadır. IR algılayıcısı bir kişi algılsa, geçiş kodunu girebilmek üzere tuş takımı etkinleştirir. Ayrıca kimlik doğrulaması için yetkili kişiye bir SMS gönderilir. Yetkili kişi onayı ile kapı açılır. Kapıda bekleyen kişi yanlış bir şifre girerse, sahibine bir bildirim mesajı gönderilir ve ayrıca sesli uyarı etkinleştirilir. Han vd. (2017) ise yaptıkları çalışmada, akıllı kapı kilidi sistemine gönderilen ve alınan verilerin sahteciliğe ve çökertilmeye karşı savunmasız olmasını ele almışlardır. Bu güvenlik sorunlarını iyileştirmek üzere, blok zinciri (block-chain) tabanlı bir akıllı kapı kilidi sistemi önermişlerdir. Diğer bir önemli çalışmada, Hadis vd. (2018), bir kapı açmak veya kilitlemek için doğrudan temas gerektirmeyen kilit sistemini geliştirmişlerdir. Ayrıca, kilit sistemi farklı fiziksel özelliklere sahip insanlar tarafından da kullanılabilir niteliktedir. Sistem düşük güçle Bluetooth teknolojisini kullanmaktadır ve neredeyse tüm cihazlarda kullanılabilmesi amaçlanmıştır. Arifin vd. (2018) tarafından önerilen modern sistemde, akıllı telefon uygulamasına konuşma komutu veya şifre girilerek kimlik doğrulaması ile kapı kilidinin açılması üzerine bir çalışma örneklenmiştir. Önerilen yöntemde, Android akıllı telefon uygulaması seri iletişim için Bluetooth kullanmaktadır. Düşük maliyetinin yanı sıra, kablosuz ev otomasyon sistemi olarak kolayca uygulanabilir. Gudavalli vd. (2020) yapılan çalışmada, iki aşamalı akıllı güvenlik doğrulaması kullanan yeni bir kapı kilidi sistemi tasarlanmıştır. Sistem, RFID tabanlı anahtar, parmak izi sensör modülü kullanılarak iki aşamalı güvenlik doğrulamasının başarıyla tamamlanmasıyla çalıştırılabilen kapı sisteminden oluşmaktadır. RFID tabanlı veya Bluetooth tabanlı kimlik doğrulamasından herhangi biri başarıyla doğrulandığında kapı kilidi açılır. Bu nedenle sistem, kapı kilidi erişiminde yüksek güvenlik sağlar ve saldırı şansını önler.

Bluetooth teknolojisine ek olarak görüntü işleme teknolojisi de kullanan çalışmalar bulunmaktadır. Örneğin, Adalan ve Erkmen (2016), işletim sistemi Android olan

akıllı telefon için geliştirilen mobil uygulamayı kullanılarak, yüz tanıma, NFC (Near Field Communication) ve ses komutu ile kapı kilidi sistemine erişimi kablosuz haberleşme kullanarak gerçekleştirmişlerdir. Akıllı telefona ses komutu iletilerek ve telefonun NFC etiketine yaklaştırılmasıyla kapı kilidi sistemi kontrol edilmiştir. Yüz tanıma işleminden yararlanarak kapı kilidi sisteminin kontrolü yapılmıştır. Kavde vd. (2017) “akıllı dijital kilit” sistemini kullanarak, kullanıcının canlı yayın yoluyla kimlik doğrulamasını yapabilmekte, Bluetooth aracılığı ile kapıyı açabilmekte ve bekleyen kişinin eve girmesine izin verebilmektedir. Bu çözüm, çalışan anne ve meşgul aileler için, çocukları veya beklenen kişiler için bir kişinin evde bulunması gerekliliğini ortadan kaldırmış olmaktadır. Hussein ve Mansoori (2017) tarafından yapılan çalışmada ziyaretçi kodu tuş takımı modülüne girilir. Raspberry pi kontrol modülüne sinyal gönderir ve daha sonra veri tabanından tarafından ziyaretçi kimliğini kontrol eder. Ziyaretçi kodu sunucuya gönderilir. Sunucu misafir-id ögesini arar. Kimlik veri tabanında mevcutsa kapı açılır ve kamerayı etkinleştirir. Kapıda kullanıcının görüntüsünü alır ve ev sahibine e-posta ile gönderir.

Daha gelişmiş Wi-fi teknolojisini kullanan araştırmalara da rastlanmaktadır. Bunlardan Mrinal vd. (2017) önerdiği sistemde, bazı işlevlere sahip ev otomasyonunun, Wi-Fi internet bağlantı protokolü ile kontrol edilmesi üzerinde çalışmışlardır. İşlevler arasında alarm tabanlı akıllı kilit, ev aletlerini uzaktan kumanda etmek, sivrisineek algılama ve akıllı su deposu bulunmaktadır. Tilala vd. (2017) tarafından önerilen diğer bir sistemde, kapı kilidine gömülü WeMos D1 Wi-Fi birimi ve Firebase bulut mesajlaşma servisiyle iletişim kuran bir Android uygulaması kullanılarak, Wi-Fi protokolü ile kapı kilidi kontrol edilmektedir. Cep telefonu uygulaması, kapı kilit sistemine güvenli erişim için güvenli bir kullanıcı şifresi ile kişiselleştirilmiştir. Sistem kullanıcının kimlik bilgilerini doğrular ve ardından bulut tabanlı güvenli mesajlaşma servisini kullanarak mobil uygulama ara yüzüne bir Tek Kullanımlık Şifre (One Time Password - OTP) gönderir ve kapının açılıp açılmadığını kontrol eder.

Wi-fi teknolojisi ile görüntü işleme teknolojisi de kullanan diğer bir çalışmada Vaidya vd. (2017) Python, OpenCV ve Android uygulamaları ve Raspberry Pi kartını kullanarak, bir akıllı ev otomasyon sistemi önermişlerdir. Sistem, Android uygulama aracılığı ile telefondan erişilebilen, ana kapının dışına monte edilmiş bir kameradan

yüz tanıma dayalı olarak tasarlanmıştır. Bu çalışmanın en yenilikçi özelliğini, tüm cihazların kullanıcı sesiyle kontrol edilebilmesi oluşturmaktadır. Girme ve Patil (2019) tarafından geliştirilen sistem, kapıdaki sensörler sayesinde ziyaretçi veya olası bir davetsiz misafiri akıllıca tespit eder. Android uygulaması sayesinde çekilen pi kamera ile çekilen görüntü mobil cihaz sahibine gönderilir. Daha sonra, ev sahibi kişiyi kabul ederse kapı kontrol komutu sisteme gönderilerek, kapı açılır. Ahmed vd. (2020) önerdiği tasarım, kullanıcının özel olarak tasarlanmış android uygulamasını kullanarak internet üzerinden canlı video geri bildirimi almasını sağlayan ucuz bir IoT tabanlı ev güvenlik sistemi sunmaktadır. Android uygulaması sensör verilerini yorumlamak için gerçek zamanlı bir veritabanı (Firebase) kullanır. Android uygulaması ayrıca şifre yönetimi, sistemin etkinleştirilmesi veya devre dışı bırakılması ve uyarı seçeneği gibi diğer özellikleri de sağlamaktadır. Güvenlik sistemi, bir hareket sensörünü ve bir kamera modülünü davetsiz misafir tespiti için birleştirir ve kullanıcıyı çevrimiçi bir uyarı bildirimi yoluyla anında uyarır. Kanıt amacıyla, davetsiz misafirin anlık görüntüleri android uygulamasından canlı video izleme sırasında alınabilir. Ayrıca hücresel şebeke üzerinden davetsiz misafir hakkında bir SMS uyarısı göndermek için bir GSM modülü kullanılmaktadır.

Sadece görüntü işleme kullanan, Kim vd. (2013) tarafından yapılan çalışma, bir kapı telefonu aracılığıyla yabancı yüzleri daha verimli ve doğru bir şekilde tanımlamak üzere, hem yüz tanıma hem de ses yerleştirme tekniklerini içermektedir. Özellikle, bir ziyaretçinin sesi konuşmacının doğru konumunu belirlemek için kullanılmaktadır. Daha sonra bu konum bilgisi kullanılarak, kapı telefonu kamerasının konumu güncellenmekte, görüş alanındaki yüz çerçevelenerek görüntü alınmaktadır. Ziyaretçinin yüz konumunu doğru bir şekilde bulmak için 4 ayrı mikrofona çapraz olarak konumlandırılmışlardır. Ayrıca, ziyaretçiyi tanımak ve bilgileri ev sahibinin telefonuna göndermek için yüz algılama ve yüz tanıma algoritmaları ile kablosuz bir arabirim de kullanılmaktadır. Tüm kontrol sistemi bir FPGA yongası kullanılarak üretilmiştir ve kapı telefonu ortamlarında gerçek kullanım için test edilmiştir. Shashank vd. (2015) önerdikleri çalışmada, kapıya yerleştirilen web kamerası ile zile basan kişinin görüntüsünü mikrodenetleyici kullanarak bilgisayara göndermektedir. Web kamerası aynı zamanda bilgisayara bağlıdır. Bilgisayarda kurulan MATLAB kodu, görüntü tanıma işlemi yapar. Tanıma tamamlandıktan sonra, MATLAB mikro denetleyiciye sinyal gönderir. Görüntü bilinen kişi ise,

mikrodenetleyici kilidi açar. Jahnavi ve Nandini (2019)'nin geliştirdiği otomatik kapı erişim sistemi, yüz tanıma ve algılama işlemi dahil edilerek sunulmuştur. Otomatik yüz tanıma, bilgisayardaki MATLAB programı tarafından gerçekleştirilir. PC'den gelen yanıtı göre kapı açma ve kapama sistemini kontrol etmek için bir step motor kullanılır. Kişi doğrulanırsa kapı açılır. Yüz tanıma için Viola-Jones yüz algılama yöntemi bir görüntüdeki yüz noktalarının konumunun algılanması için kullanılmaktadır.

Bulut bilişim teknolojisi kullanan Aman ve Anitha (2017) tarafından geliştirilen modelde, akıllı telefona kurulan bir Android uygulaması tarafından kontrol edilebilen bir akıllı kapı sistemi önerilmiştir. Kapının hareketini kontrol etmek üzere Raspberry Pi ve ilgili gömülü yazılım kullanılmaktadır. Hareket duyargaları, kapının önündeki herhangi bir hareketi tespit etmek için kullanılmaktadır. Herhangi bir kişi kapının önüne gelirse, algılayan duyarga tetiklenmekte, resim çekilerek ilgili kişilere bildirilmektedir. Muthumari vd. (2018) kamera özelliğine sahip akıllı telefonla kapıda bulunan kişi / kişilerin fotoğrafını çekerek, Wi-Fi yardımıyla kapının açılabilmesini sağlayacak bir sistem önermişlerdir. Yetkili kişi kapı önünün görüntüsünü görebilir, akıllı kapı kilidini açabilir veya bir cihazla da kilitleyebilir. Evin içinden Bluetooth ile kapının kilidini açmak mümkündür ve hatta çocuk veya yaşlı insanların Wi-Fi menziline oldukları durumda telefonla kapı kilidini açabilmeleri bazı özel durumlarda (felçli, engelli hastalar) oldukça faydalıdır. Bunun için ev üyelerinin uygulamayı kendi akıllı telefonlarına yüklemeleri yeterli olacaktır. Böylece, engellilerin kimseden yardım almadan kapıyı açabilmesine ve normal bir yaşam sürebilmesine yardımcı olunacaktır. Pinjala ve Gupta (2019), önerdiği sistemde akıllı telefonda bir android uygulaması kullanılarak uzak bir konumdan tamamen izlenebilen ve kontrol edilebilen bir akıllı kapı kilidi sunmaktadır. Bir ziyaretçi kapıda bulunan zile bastığında, HD kamera ile ziyaretçinin görüntüsü alınarak bir Raspberry pi ev sahibine bir mesajla haber verir. Ev sahibi, istekliliğine göre, önceden ayarlanmış bir parola girerek kapıyı kolayca açabilir / kapatabilir. Buna ek olarak, akıllı telefon uygulamasında ile ziyaretçiye sesli bir mesaj iletebilir. Yashwant vd (2020) tarafından önerilen sistemde akıllı bir mobil cihaz, Arduino Uno, Wi-Fi modülü, Bluetooth modülü, Google Home, veritabanı sunucusu ve bir Firebase bulut mesajlaşma servisinden ve internet bağlantısından oluşur. Bir konuk yöneticiyle iletişim kurmaya çalıştığında ve yönetici yanıt vermediğinde, akıllı kilit,

konukların eve erişmesi için benzersiz bir fırsat sağlar. Bu erişim modu için akıllı kilidin Google Home veya Alexa gibi akıllı bir hoparlöre bağlanması gerekir. Akıllı konuşmacı yöneticinin kayıtlarını içerdiğinden, kişi, kişi listesindeyse konuk akıllı kilide erişebilir. Konuk listede varsa, konuğa bir doğrulama bağlantısı gönderilir. Misafirin doğrulama bağlantısını tıklaması ve cep telefonu numarasına veya e-postasına gönderilen benzersiz bir şifre girmesi gerekir (kişi listesindeki kişi türüne bağlı olarak). Parolayı doğruladıktan sonra kapı otomatik olarak açılır. Ancak, yönetici bu tür erişim için kişileri seçebilir. Yönetici, uygulamada istediği zaman konuk modu kullanıcı listesini her zaman düzenleyebilir.

Diğer teknolojileri kullanan Park vd. (2009) tarafından yapılan çalışmada, ev otomasyonu için akıllı bir dijital kapı kilit sistemi önerilmiştir. Burada önerilen kapı kilit sistemi, kullanıcı kimlik doğrulaması için RFID okuyucu, dokunmatik LCD, kapının açılması ve kapatılması için motor birimi, evin içindeki durumu tespit etmek için algılayıcı birim, iletişim birimi ve diğer birimleri kontrol etmek üzere kontrol birimlerinden oluşmaktadır. Mathew ve Divya (2017), çalışmalarında güvenli erişim için kullanıcıyı etkinleştiren, doğrulayan ve gerçek zamanlı olarak kapının kilidini açabilen RFID kullanan iki faktörlü kimlik doğrulama ve çoklu şifreleme özelliklerine sahip güvenli bir kapı kitleme sistemi önermiştir. Temel amaç, yalnızca yetkili kişinin girebileceği kritik bölgede devreye girebilecek bir dijital güvenlik sistemi tasarlamak ve uygulamaktır. Bir diğer uygulama ise, Lita vd. (2017) tarafından yapılmış, burada akıllı evlerde erişim kontrolü amacı ile kullanılacak bir pnömatik (gaz basıncını mekanik harekete dönüştüren) kapı otomasyon sisteminin tasarımı ve örnek uygulaması sunulmuştur. Geliştirilen uygulamanın yapısı, selenoid valfli bir hava dağıtıcısı aracılığıyla kontrol edilebilen çift etkili bir silindiri süren bir pnömatik motor ve onu kontrol eden PIC 16F877A mikro-denetleyicisinden oluşmaktadır. Vongchumyen vd. (2017) önerdikleri sistem donanım ve yazılım olmak üzere iki ana bölümden oluşmaktadır. Donanım, komutu kablolu yönlendirici aracılığıyla yazılımdan alır. Web uygulaması biçimindeki yazılım cep telefonunda çalışır veya tablet kullanıcı için bir arayüz görevi görür. Donanımın iki ana bileşeni vardır: kontrol kutusu ve kapı. Kontrol kutusu, TCP / IP protokolündeki komut paketi aracılığıyla kontrol kutusuna gönderen kullanıcıdan komutu yorumlamak için bir Raspberry Pi kullanır. Daha sonra Raspberry Pi sunucusu, değiştirilmiş kapı kilidini step motor üzerinden kontrol etmek için eylemi

ATtiny2313 kontrol cihazına gönderir. Raspberry Pi, kapının konumunu kontrol etmek için kullanılan limit anahtarına bağlanır. Kapı kapalıysa, kilitleme komutu çalıştırılabilir, eğer değilse kilitleme komutunu gerçekleştiremez. Ahtsham vd. (2019) tarafından geliştirilen sistemde; bilgi güvenliğinin artırılması için AES-128 ve SHA-512 şifreleme algoritmaları kullanılmış, sensörler ve aktüatörlerle programlanabilir donanım kullanan bir Android akıllı telefon uygulamasını içermektedir. Uygulama, sistemle güvenli bir kanal üzerinden iletişim kurar ve güncellemeleri alır. CryptoLock, verileri ağ üzerinden korumak için farklı güvenli şifreleme algoritmaları kullanır. Uzaktan erişim ile ağ üzerinden yetkisiz erişimi ve güvenli bilgileri yönetebilmektedir.

3. NESNELERİN İNTERNETİ

1991 yılında, Cambridge Üniversitesindeki akademisyenler kahve makinesinin dakikada üç adet görüntüsünü yakalayıp bilgisayarlarına aktaran bir sistem tasarlamışlardır. Bir video kamerayla, görüntü yakalama yazılımı ve protokollerin yazılmasından sonra her akademisyen ekranında kahve demliğindeki kahve miktarını çevrimiçi görebilmiştir (Kutup,2011).

1993 yılında web ortamına taşınan bu uygulama günde milyonlarca kez izlenmeye başlanmış. Bu kahve makinesi (Şekil 3.1.) “Nesnelerin İnterneti” ve bağlı nesnelerin varlığının ilk ispatı ile örneğini oluşturmuştur (Kutup,2011). Geliştirilen bu sistem 22 Ağustos 2001 yılına dek kullanılmıştır (Cambridge, 2020).



Şekil 3.1. Kahve makinesinin görüntüsü (Cambridge,2020).

Ashton’a (2009) göre Nesnelerin İnterneti kavramı, 1999 yılında Radyo Frekansı ile Tanımlama (RFID) teknolojisinin Procter&Gamble firmasının tedarik zinciri ilgili bir sunumunda Kevin Ashton tarafından ilk kez kullanılmıştır.

“Nesnelerin İnterneti” kavramının farklı tanımları bulunabilir. En yaygın kullanılan tanımı “Nesnelerin İnterneti, herhangi bir cihazı (açma /kapama düğmesi olduğu sürece) İnternet’e ve diğer bağlı cihazlara bağlama kavramıdır” biçimindedir. Ayrıca IoT, “hepsi nasıl kullanıldıkları ve çevreleri hakkında veri toplayan ve paylaşan devasa bağlantılı şeyler ve insanlar ağıdır” olarak da tanımlanmaktadır (IBM, 2020).

Nesnelerin İnterneti (IoT), internet üzerindeki diğer cihazlar ve sistemler üzerinde veri alışverişi yapmak amacıyla algılayıcıları, yazılımları ve diğer teknolojilerle donatılmış fiziksel nesnelerin ağını tanımlar. Bu cihazlar sıradan ev eşyalarından, karmaşık endüstriyel aletlere kadar uzanmaktadır. Uzmanlar günümüzde 7 milyardan fazla bağlı IoT cihazının, 2020'ye kadar 10 milyara ve 2025'te 22 milyara çıkması beklemektedirler (Oracle, 2020).

Nesnelerin İnterneti; Akıllı Ev, Endüstriyel Alan, Akıllı Sağlık, Giyilebilir Cihazlar , Akıllı Çevre, Akıllı Tarım, Akıllı Hayvancılık, Akıllı Enerji, Akıllı Şehirler, Güvenlik ve Acil Durumlar, Alışveriş, Lojistik gibi uygulamalarda kullanılmaktadır.

3.1. Güvenlik

Nesnelerin interneti günümüzde birçok alanda kullanıma girmiştir. Herhangi bir alanda kullanılan akıllı sistemlerden beklenen güvenliğinin nasıl sağlanması gerektiğidir. Bu sistemlerin güvenliğinin sağlanması için tüm bilişim sistemlerinde olduğu gibi güvenilirlik, bütünlük, kimlik doğrulama, mahremiyet, erişilebilirlik vb. unsurlarının uygulanması gerekmektedir.

3.2. IoT Uygulama Alanları

3.2.1. Akıllı sağlık

Sağlık alanında IoT'nin önem kazanması cihazların doğrudan verilerle birbirlerine bağlanmasıdır. Sağlık alanında IoT kullanılması iki nedenden dolayı önemlidir. Bu nedenler aşağıda verilmiştir (ioturkiye, 2020):

1. Hasta verilerini toplamak.
2. Hekimlerin ihtiyaç duyduğu verilerin otomatik olarak elde edilmesi. Böylece hata riskini azaltır. Daha az hata, verimlilik artışı, düşük maliyetler sağlanmasıdır.

IoT, hastalıkların yönetiminden ve hastalık önleme gibi geniş kapsamlı sağlık uygulamaları alanında önemli bir role sahiptir. Bu uygulamalara ait bazı örnekler şöyledir:

- Klinik Bakım
- Uzaktan izleme
- Erken Müdahale

Bu özellikler bakımından hastaların takibine başlanmıştır. Özellikle kişinin vücuduna yerleştirilen giyilebilir cihazlar sayesinde belirli değerlerin (kan, tansiyon, nabız, şeker, kolesterol) ölçümleri düzenli aralıklarla yapılarak en yakın sağlık kuruluşuna veya kişinin aile hekimine bilgi verilir.

Dünyada olduğu gibi ülkemizde de kullanılmaya başlanmıştır ve üzerinde araştırmalar devam etmektedir.

3.2.2. Akıllı şehir

Akıllı şehir uygulamaları; öncelikli olarak şehir trafiğini rahatlatma, uygun park yeri gösterme, şehrin ışıklandırmasını daha etkili ve daha verimli yapılması, atıkların geri dönüşümü gibi kaynakların daha verimli kullanmalarını sağlar.

Ülkemizde iTaksi uygulaması önemli bir örnektir. iTaksi (2020), akıllı telefondaki uygulama sayesinde şoför ve yolcuyu tam zamanlı buluşturmayı amaçlayan yolcunun bulunduğu bölgeye en yakın konumdaki taksi ile haberleşmeyi sağlayana bir uygulamadır.

3.2.3. Endüstriyel alan

Endüstriyel sistemlere IoT teknolojisinin entegre edilmesiyle üretimde kullanılan birbiriyle konuşabilen akıllı cihazlar insan hatasını en aza indirerek, gerçek zamanlı bilginin teknik personel tarafından kontrol edilmesini sağlamak amaçlanmıştır.

Bu sayede üretimde kalitenin artırılması, maliyetlerin azaltılması, enerjinin verimli kullanılması, rekabetçi ürünlerin üretilmesi gibi birçok olumlu etki oluşturacaktır.

3.2.4. Akıllı tarım

Zhao vd. (2010), seralar gibi kapalı ortamda nesnelerin birbirleriyle iletişimi sayesinde oluşan verilerinin kontrol edilebileceği ve farklı tarım ürünlerinin yetiştirilmesinde en uygun şartların sağlanmasına yönelik çalışmalardır. Bir serada kontrol edilmesi gereken sıcaklık, nem ve toprak değerleri gibi IoT sisteminin bir parçası olan makineler arası iletimi bir bilgisayarda veya internet ortamında yapılandırılarak saklanır, değerlendirilir veya mobil kullanıcıların kendi telefon uygulamaları veya web erişimleri üzerinden kontrol edilebilir.

Akıllı tarım; kaynak israfını önlemek, çevre kirliliğini en aza indirmek, ürün kalitesini arttırmak gibi birçok önemli avantaja sahiptir.

3.2.5. Akıllı ev



Şekil 3.2 Akıllı Ev (Aws, 2020)

King (2003)'e göre IoT özellikli “akıllı ev” ortamında, aydınlatma, ev aletleri, bilgisayarlar, güvenlik kameraları vb. çeşitli şeyler birbirine bağlıdır. Diğer bir deyişle, bir ağa bağlı evdeki cihazların kontrol edilmesidir.

Stefanov vd (2003)' göre akıllı ev, otomatik işlevleri olan ve kullanıcı tarafından uzaktan kontrol edilebilen sistemlere sahip cihazları içerir. Böylece insanların evdeki konforu artmaya devam edecektir.

Akıllı bir evin temel amacı, ev sakinleri için konfor, enerji tasarrufu ve güvenliği arttırmaktır. Bayram (2006) yaptığı çalışmaya elverişli bir ev otomasyon sistemi, aşağıdaki kontrollerden hepsini ya da daha fazlasını gerçekleştirebilir.

- Evdeki tüm lambaların açılıp kapanması kontrolünü yapar.
- Su ısısının belirtilen derecede ayarlanması,
- Perdelerin açılıp kapatılması,
- Kişinin evden çıktığı zaman güvenlik sisteminin devreye girmesi,
- Yangın olması durumunda kişiyi ve itfaiyeyi araması,
- Televizyonun istenildiği saatte kapatılması,
- Uzaktan kumanda ve telefonla evdeki tüm elektronik cihazların kontrolü yapılır.

Çalışmada, ev kapısının uzaktan kontrol ile açılması üzerine “akıllı ev” uygulaması örneklenmiştir. Uzaktan kontrol için internet alt yapısı kullanılmaktadır. İnternete bağlı olan bir cep telefonu ve kapıyı kontrol eden bir elektronik kart (Raspberry Pi) ile çözüm sunulmaktadır.

4. MATERYAL VE YÖNTEM

4.1. Raspbberri Pi 3 B+

Raspberry Pi, bilgisayar monitörüne veya TV'ye takılan ve standart bir klavye ve fare kullanabilen düşük maliyetli, kredi kartı boyutunda bir bilgisayardır. Her yaştan insanın bilgisayarları keşfetmesini ve Scratch ve Python gibi dillerde programlamayı öğrenmesini sağlayan küçük ve yetenekli bir cihazdır (Raspberrypi,2020). Farklı özelliklere sahip çeşitli sürümleri üretilmiş olan bu cihazın bir örneği Şekil 3'te verilmiştir.



Şekil 4.1. Raspberry Pi 3B+ model

4.1.1. Raspberry Pi 3 B+ Donanım Özellikleri

BCM2837B0, Cortex-A53 64-bit 1.4GHz işlemci, 1 GB LPDDR2 SDRAM, 2.4GHz ve 5GHz IEEE 802.11.b/g/ n/ac kablosuz LAN, Bluetooth 4.2, BLE, USB 2.0 üzerinden Gigabit Ethernet (maksimum çıkış 300 Mbps), Genişletilmiş 40 pinli GPIO başlığı, Tam boyutlu HDMI, 4 USB 2.0 bağlantı noktası, CSI kamera portu, DSI dokunmatik ekran portu, Video ve ses çıkış portu (4'lü jack çıkışı), İşletim sisteminizi yüklemek ve veri depolamak için Micro SD port, 5V / 2.5A DC güç

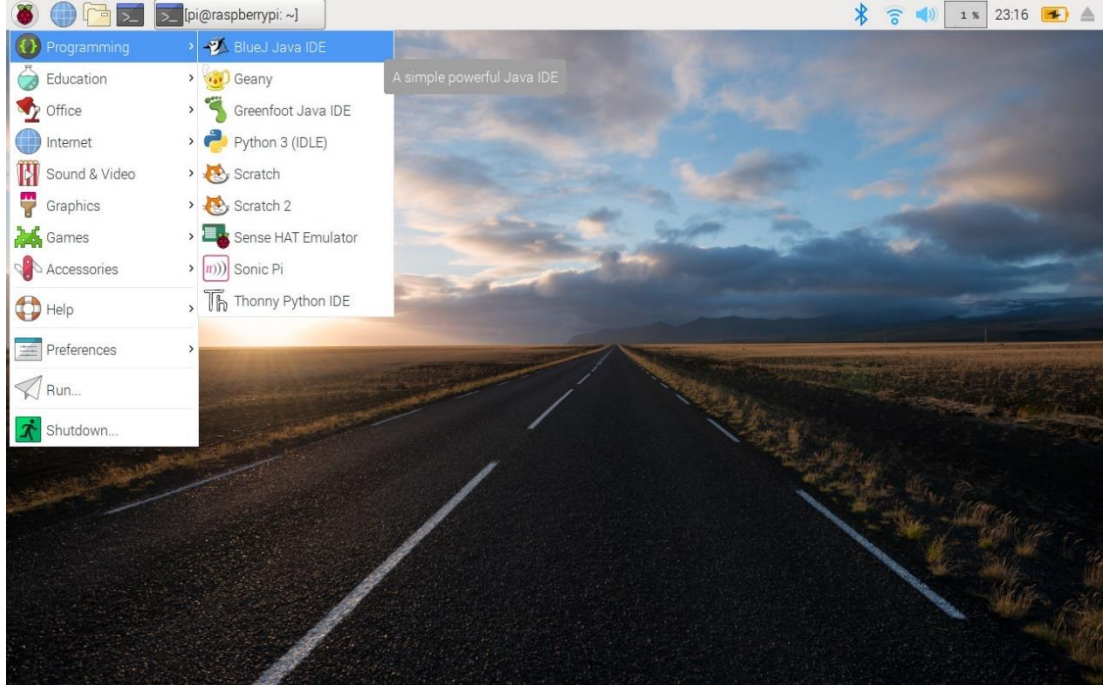
girişi, Ethernet üzerinden Güç (PoE) desteği (ayrı PoE HAT genişletme kartı gerektirir) (Raspberrypi,2020) bulunmaktadır.

Raspberry Pi B+ B+ J8 GPIO Header			
	Pin No.		
3.3V	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
DNC	27	28	DNC
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

Şekil 4.2. Raspberry pi 3B+ model pin diyagramı (Vujoic, 2015)

4.1.2. İşletim Sistemi

Raspberry Pi 3 B+, Linux tabanlı Raspbian işletim sistemi ile çalışır. Windows 10 IoT core işletim sistemide kurulumu yapılabilir. Raspbian, Raspberry Pi için optimize edilmiş Debian'a dayalı ücretsiz bir işletim sistemidir. Bu işletim sistemi, Raspberry Pi'yi çalıştıran temel programlar ve yardımcı programlar kümesidir. Şekil 4.3.'de Raspbian masaüstü görsel verilmiştir.



Şekil 4.3. Raspbian işletim sistemi masaüstü görünümü

4.2. Usb Kamera

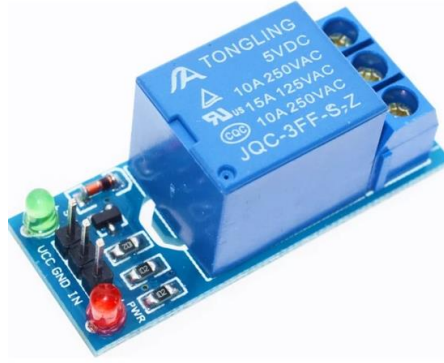
Raspberry Pi’ de bulunan usb bağlantısı sayesinde bağlanan Şekil 4.4.'de görünen USB Kamera ile kapının önündeki kişinin fotoğrafını çekmek mümkün olmaktadır.



Şekil 4.4. Usb kamera

4.3. R le Devresi

R le, d    k g    devrelerinin nispeten y  ksek bir voltajı veya akımı a  p kapatmasını sa  layan elektrikle  alı an anahtarlardır. Bir r lenin  alı ması i in bobininden uygun bir i eri  ekme ve tutma akımı ge irilmelidir. R le bobinleri genellikle 5V veya 12V olan belirli bir voltajdan  alı acak  ekilde tasarlanmı tır. R le s r c  devresinin i levi, y  ksek akıma kar ı devreyi korumaktır.



 ekil 4.5. 1 kanal 5V r le kartı

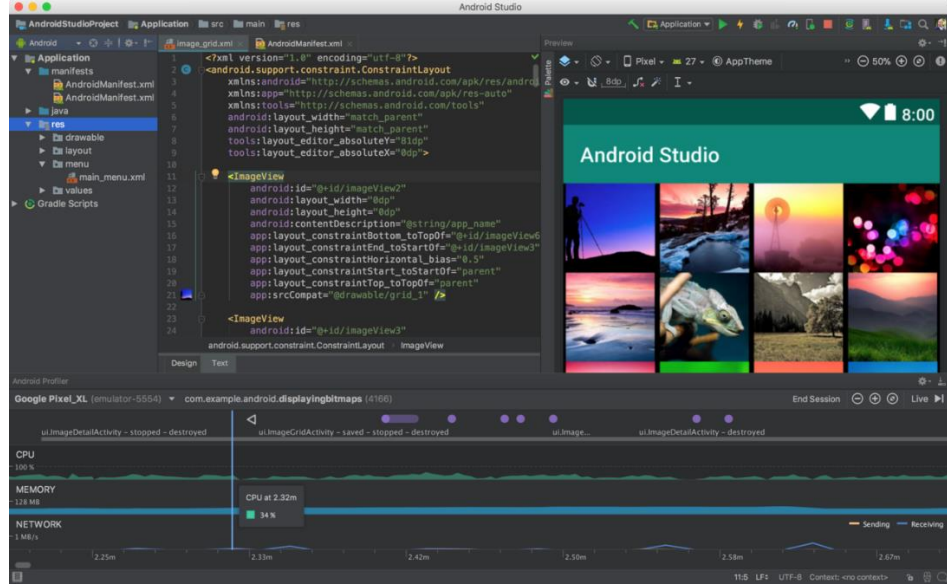
4.4. Andorid Studio

Android Studio, telefonlar ve tabletler, Android Auto, Android Wear ve Android TV t m Android cihazlarda uygulama geli tirmek i in tasarlanmı tır. Android Studio, akıllı bir kod d zenleyici ve hata ayıklayıcı, performans analiz ara ları, em lat rler ara larından olu an uygulama geli tiricidir. Temel  zellikleri (Andorid, 2020):

- Akıllı kod d zenleyici.
- Android em lat r.
- T m Android platformları desteklenir: Android telefonlar ve tabletler, Android Wear, Android Auto ve Android TV.
- Gradle tabanlı esnek yapılı derleme sistemi derleme otomasyonu.
- Yaygın olarak kullanılan uygulama  zellilerini olu turmaya yardımcı olacak kod  ablonları.
- Hata ayıklama deste i.

- Firebase SDK, Firebse Test Lab, Firebase Uygulama Dizinine Ekleme ve Google Cloud Platform için yerleş destek.
- APK Analizcisi

Dünyada en fazla sayıda akıllı telefonda bulunan işletim sistemi Android işletim sistemidir. Bundan dolayı geliştirilen uygulama Android Studio kullanılmıştır.



Şekil 4.6. Android Studio ekran görüntüsü

4.5. OpenCV

Açık kaynaklı bir bilgisayarlı görü makine öğrenimi yazılım kütüphanesidir. Birçok algoritması vardır. Bu algoritmalar yüzleri algılamak ve tanımak, nesneleri tanımlamak, videolardaki insan eylemlerini sınıflandırmak, kamera hareketlerini izlemek, hareketli nesneleri izlemek, nesnelerin 3B modellerini çıkartmak, stereo kameralardan 3B nokta bulutları üretmek, yüksek çözünürlük üretmek, bütün bir sahnenin yüksek çözünürlüklü görüntüsünü elde etmek, görüntüleri birleştirmek, flaş kullanarak çekilen görüntülerden kırmızı gözleri kaldırmak, göz hareketlerini takip etmek, sahne tanımak ve artırılmış gerçeklik için kullanılır. (OpenCV, 2020). Bu çalışmada OpenCV kullanılmasıdaki amaç kamera önündeki kişinin görüntüsünü alarak Raspberry Pi üzerindeki dosyaya kaydetmesini sağlamaya yarar.

4.6. OpenVpn

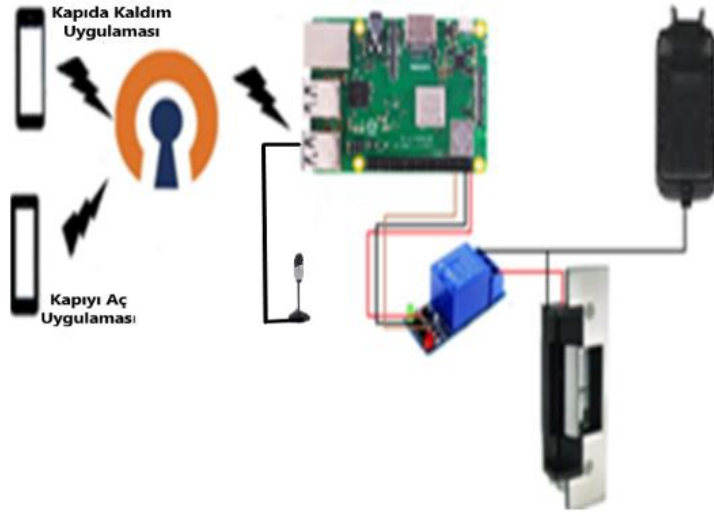
OpenVpn, güvenilir, ölçeklenebilir iletişim servisidir. OpenVpn temel özellikleri (OpenVpn, 2020):

- Açık kaynak kodlu
- Donanım bağımsız
- Çok sayıda işletim sistemi desteğine sahiptir
- Çok kolay kurulum kullanılabilen
- Performansı yüksek bir VPN uygulamasıdır.

Bundan dolayı birçok IoT uygulamalarında OpenVpn kullanılmaktadır. Bu çalışmada, Raspberry Pi ve ev sahibinin kullandığı android işletim sistemi olan akıllı cep telefonuna OpenVpn yüklenerek güvenliği en üst düzeyde tutulması sağlanmıştır.

5. ARAŞTIRMA BULGULARI VE TARTIŞMA

Bu çalışmada önerilen sistem, kapıda kalan bireyin (çocuk, yaşlı, bakıcı) akıllı cep telefonunda bulunan Android uygulama sayesinde kapı açılma isteğinde bulunabilmesi, ev sahibi konumunda bulunan yetişkin bireyin (anne, baba) akıllı cep telefonunda bulunan diğer bir Android uygulama sayesinde kapıyı açabilmesi üzerinedir. Çok katmanlı yapı ile kapı gibi önemli bir donanım üzerinde maksimum güvenlik sağlanmaya çalışılmıştır.



Şekil 5.1. Sistemin genel yapısı

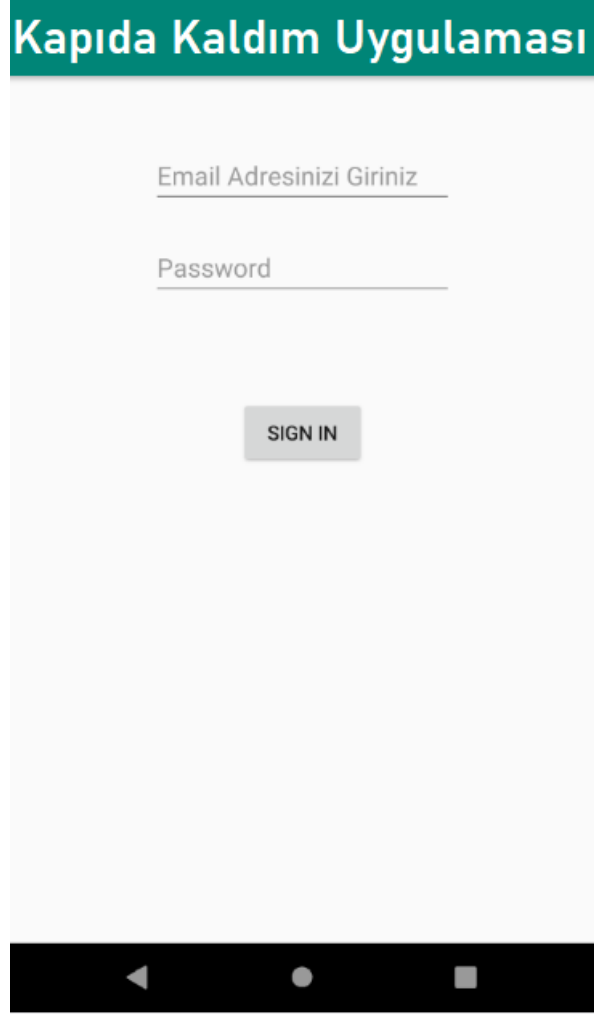
Sistemin detaylı uygulama süreci, kapıdaki kişinin telefonunda bulunan Android uygulama ile ev sahibine (genellikle ebeveyn) kapının açılması için istekte bulunması ile başlar. Bu istek, Raspberry Pi üzerindeki PHP Web servisinin uyarılması ve Python ile kodlanmış yazılım kullanılarak kamera görüntüsünün ev sahibine mail olarak gönderilmesi ile devam eder. Nihai olarak, ev sahibinin telefonunda olan diğer Android uygulama ile onay verilirse Raspberry Pi üzerindeki diğer PHP Web servisinin uyarılması sonucu elektrikli kapı kilidine yeterli gerilim verilerek kapının açılması ile sonlanır. Akıllı cep telefonları ve Raspberry Pi güvenli bir şekilde veri alışverişi yapabilmesi için OpenVPN kullanılmıştır. Sistemin genel yapısı Şekil 5.1.' de verilmiştir.

Sistem üzerinde çok farklı donanım ve yazılım ürünleri kullanılmıştır. Akıllı kart olarak Raspberry Pi B+ modeli belirlenmiştir. Bu mini bilgisayarı projede kullanabilmek için, sırasıyla aşağıdaki adımlar uygulanmıştır:

1. Raspbian işletim sistemi kurulmuştur.
2. Web kameradan fotoğraf almak için OpenCV kurulumu yapılmıştır.
3. Çekilen fotoğrafı Raspberry Pi'den göndermek için projeraspberry@gmail.com isimli e-posta adresi tanımlanmıştır.
4. Gönderilen e-postayı ev sahibinin alması için proje.human@gmail.com isimli e-posta adresi alınmıştır.
5. Raspberry Pi ile, akıllı cep telefonunda bulunan Andorid Studio üzerinde geliştirilen uygulama yazılımının birbirleriyle iletişim kurabilmesi için Apache (web sunucu (Apache, 2020)), Myadmin (MySQL'in Web üzerinden yönetimini ele almayı amaçlayan, PHP ile yazılmış ücretsiz bir yazılım aracıdır. (phpmyadmin, 2020)) ve Flask(Pyhon web uygulamalarında kullanılan frameworktür.(pypi, 2020)) kurulmuştur.
6. IoT ağındaki cihazların güvenilir bir şekilde iletişim sağlamaları için OpenVpn kurulmuştur. OpenVpn hem akıllı cep telefonlarına hem de Raspberry Pi' ye kurulmuştur.

Ayrıca Android uygulamalarında yer alan kullanıcı ve şifre girişlerine ait panel bilgilerinin kontrolü için FireBase (gerçek zamanlı veri tabanı (firebase, 2020)) kurulmuştur. Kapının açılması esnasında Raspberry Pi'yi yüksek akımlara karşı korumak için bir röle kullanılmıştır.

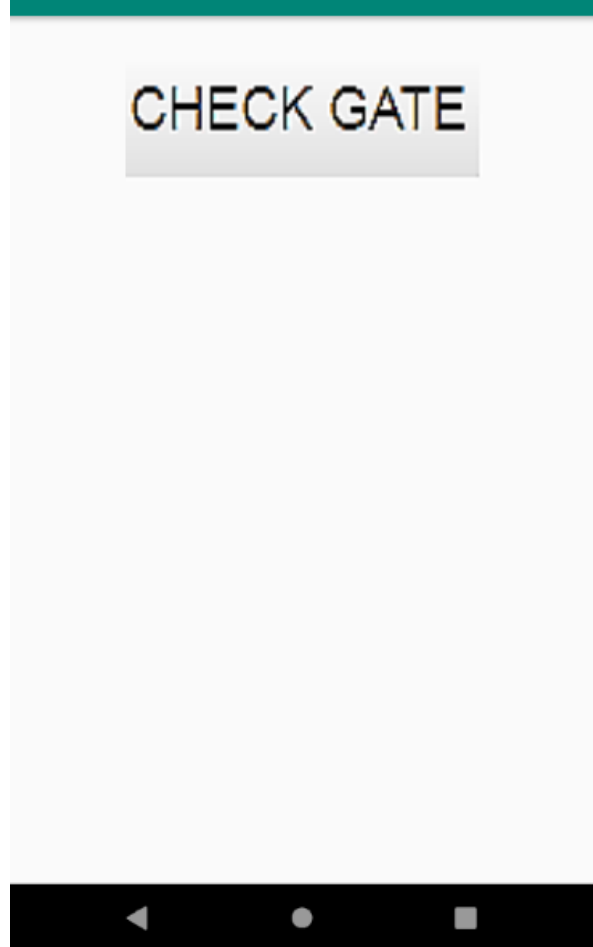
Android uygulamalarında güvenliği artırmak üzere e-posta ve şifre giriş paneli tasarlanmıştır. Bu basit ekranın görüntüsü Şekil 5.2' de görülmektedir. Benzer yetkili giriş ekranı “Kapıda Kaldım” Android uygulaması içinde kullanılmaktadır.



Şekil 5.2. Kapıyı Aç Android Uygulaması giriş paneli

"Kapıda Kaldım" Android uygulamasında e-posta ve şifre doğrulaması olduktan sonra Şekil 5.3.' de görünen basit ekran görüntülenir. Burada yer alan "CHECK GATE" düğmesine basılınca Şekil 5.4' de verilen kapidaKaldim.php web servisi çalışır. Bu servis Raspberry Pi üzerindeki Python kodunu çağırır. Kapı önünün kameradan görüntüsü alınarak kapidaGoruntu.jpg olarak kaydeden Python kodu Şekil 5.5'de verilmiştir.

Kapıda Kaldım Uygulaması



Şekil 5.3. Kapı açılması için istek gönderme paneli

```
<?php
try{
    exec("sudo python3 pinKontrol.py");
    echo 'Request is succesfully';

}
catch (Exception $e) {
    echo $e->getMessage();
}
?>
```

Şekil 5.4. kapidaKaldim.php servisi

```
camera = cv2.VideoCapture(0)
return_value,image = camera.read()
cv2.imwrite(kapidaGoruntu.jpg',image)
camera.release()
cv2.destroyAllWindows()
fileLocation='/home/pi/Desktop/ornek/ kapidaGoruntu.jpg'
email='proje.raspberry@gmail.com'
```

```

password='proje.deney'
sendToMail='proje.human@gmail.com'
subject='Kapda ki kim'
message='Resmi kontrol et! - http://10.8.0.9/ kapidaKaldim.php'
fileLocation='/home/pi/Desktop/ornek/ kapidaGoruntu.jpg'
msg=MIMEMultipart()
msg['From']=email
msg['To']=sendToMail
msg['Subject']=subject
msg.attach(MIMEText(message,'plain'))
print(msg)
fileName=" kapidaGoruntu.jpg" attachment = open(os.getcwd()+'/
kapidaGoruntu.jpg', "rb")
part = MIMEBase('application', 'octet-stream')
part.set_payload(attachment.read())
encoders.encode_base64(part)
part.add_header('Content-Disposition',"attachment;fileName=%s"
%fileName)
msg.attach(part)
server=smtplib.SMTP("smtp.gmail.com",587)
print(server)
server.starttls()
server.login(email, password)
text=msg.as_string()
server.sendmail(email, sendToMail,text)
server.quit()

```

Şekil 5.5. Kapı önü kamera görüntüsü alma kodu ve e-posta gönderme kodu

Raspberry Pi'de kapidaGoruntu.jpg olarak kaydedilen fotoğraf proje.raspberry@gmail.com e-posta adresinden yetişkin bireyin (anne, baba) kullandığı proje.human@gmail.com e-posta adresine gönderilir. Web servisi ile veri iletişimi sağlamak istendiğinden Raspberry Pi 'de 5000 portu açılır ve bu porttan veri alınıp, gönderilir. Raspberry Pi üzerinde Flask ile entegre çalışan Şekil 5.5' de ki Python kodları 5000 portuna gelecek taleplerin değerlendirilmesi için kullanılmıştır.

```

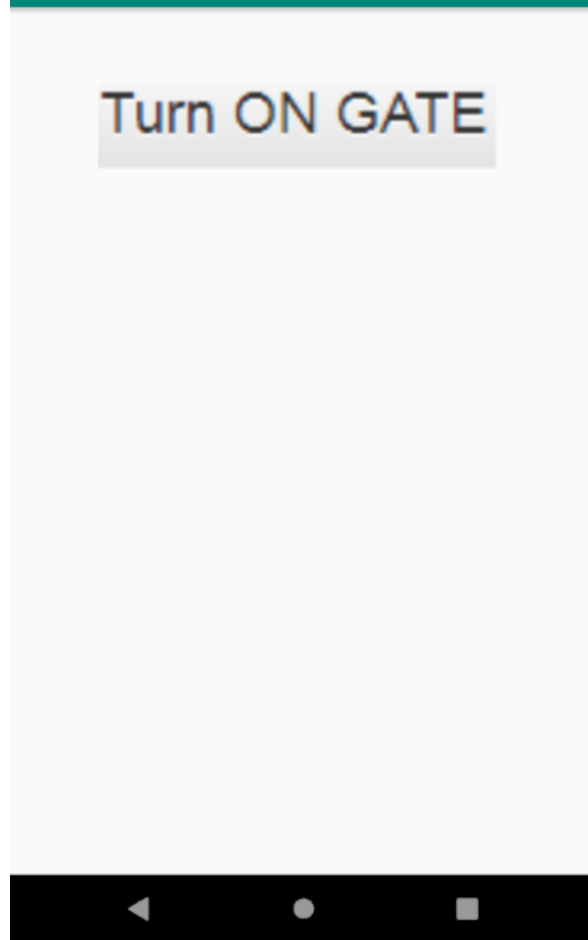
if __name__ == "__main__":
app.debug=True
app.run(host='0.0.0.0', port=5000)

```

Şekil 5.6. Raspberry Pi'ye gelen taleplerin değerlendirilmesi

E-posta ile gelen kapidaGoruntu.jpg resmindeki bireyin tanıdık bir kişi olduğu yetkili kişi tarafından anlaşıldığında, kendi cep telefonunda bulunan “Kapıyı Aç” Android uygulamasını kullanır. Şifre giriş panelinden doğru bilgileri girdiğinde, Şekil 5.6'da gösterildiği gibi "TURN ON GATE" düğmesini içeren ekran karşısına gelir. Bu düğmeye basıldığında Şekil 5.7'deki kapıyıAc.php web servisi çalışır. Bu web servisi Raspberry Pi üzerindeki Şekil 5.8' de verilen Python kodunu çağırır. Kod incelendiğinde görüleceği üzere, 7. pin'e verilen gerilimle elektrikli kapı kilidi açılmış olacaktır. 5V'luk röle Raspberry Pi' nin kapı açılma esnasında maruz kalabileceği yüksek akım zararlarından korunması için kullanılmıştır.

Kapıyı Aç Uygulaması



Şekil 5.7. Kapı açılması için istek gönderme paneli

```
<?php
try{
    exec("sudo python kapiac.py");
    echo 'SUCCESFULLY OPEN THE GATE';
}
catch (Exception $e) {
    echo $e->getMessage();
}
?>
```

Şekil 5.8. kapiyiAc.php web servisi

```

pin = 7
gp.setmode(gp.BOARD)
gp.setup(pin, gp.OUT)
def kapi_acik(pin):
    gp.output(pin, gp.HIGH)
def kapi_kapali(pin):
    gp.output(pin, gp.LOW)
gp.cleanup()
print('kapali')
if __name__ == '__main__':
    try:
        time.sleep(1)
    except KeyboardInterrupt:
        gp.cleanup()

```

Şekil 5.9. Kapı açma

6.SONUÇ VE ÖNERİLER

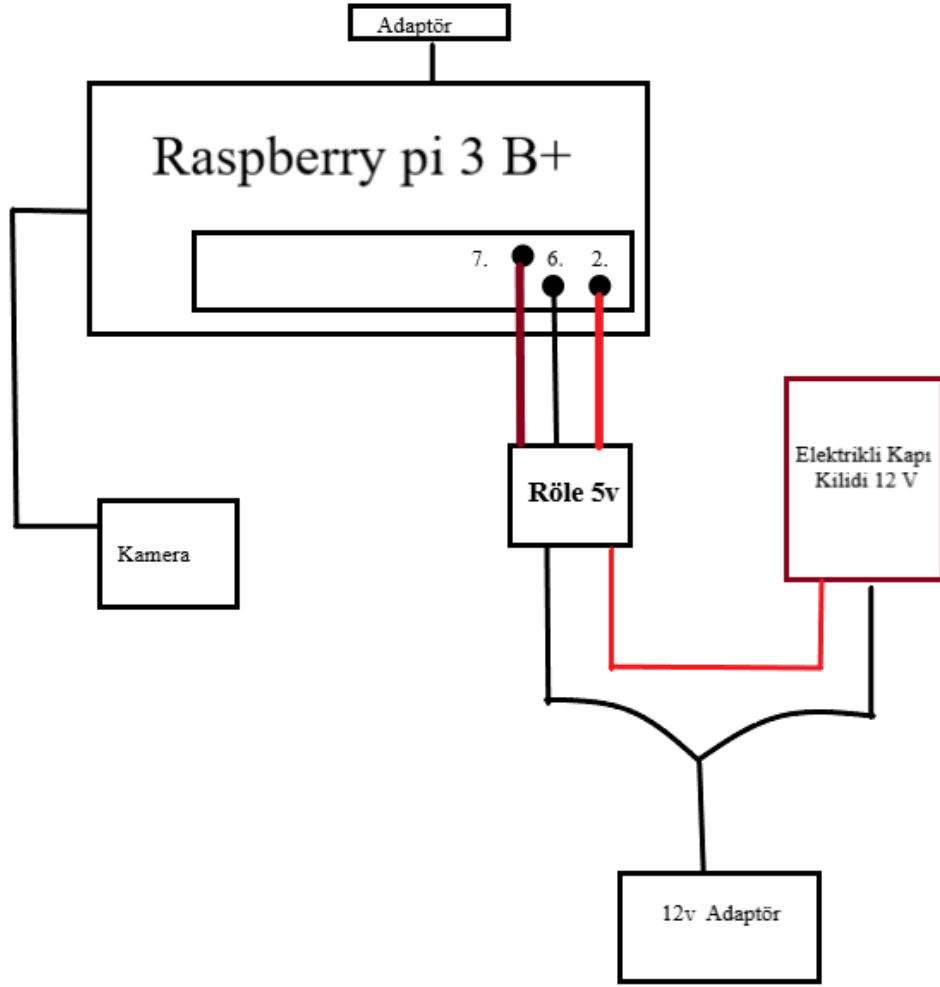
Güvenlik, tehditlere karşı korunmak için geliştirilen bir savunmadır. Güvenlik; evimizi, ofislerimizi vb. yerlerde, diğer yetkisiz kişilerin bunlara erişemeyeceği şekilde gizli tutmak için en büyük endişelerden biri olmuştur. Daha önceleri yeterince emniyetli olmayan mekanik kilitler mevcuttu. Teknoloji ilerledikçe, daha fazla hırsızlık ve yetkisiz erişimi önlemek için modern elektronik kilitler piyasaya sürülmüştür.

Nesnelerin İnterneti (IoT) teknolojisinin kullanımı geliştikçe geleneksel kilit sistemleri de geliştirilmiştir. Geleneksel kilit üzerindeki gelişmeler yenilik, zayıflıkları azaltmayı amaçlamaktadır.

Önerilen sistem modelinin bir prototipi Şekil 6.1’de görülmektedir. Sistemin maksimum güvenli olması amacı ile çeşitli teknik çözümler entegre edilmiştir. Örneğin, özel e-posta kullanılarak kapıdaki kişinin görüntüsü güvenli bir biçimde iletilmektedir. Daha da önemlisi, her iki kullanıcı tipi için bağımsız Android uygulamaları geliştirilmiş, yetki girişleri bağımsız kılınmıştır.



Şekil 6.1. Önerilen sistem prototipi.



Şekil 6.2. Tasarlanan sistemin şeması.

Ayrıca günümüzde siber saldırıların artmasından dolayı, IoT cihazları için güvenlik önlemlerinin alınması zaruri olmuştur. Bu amaçla, internet ağı üzerinden gelebilecek olası saldırılardan korunmak üzere özellikle VPN kullanılmıştır. OpenVPN noktadan noktaya bağlantı kurduğu ve gönderilen veriler şifrelendiği için kullanılmıştır.

Önerilen model, sunduğu üst düzey ve katmanlı güvenlik mimarisinin yanı sıra, maliyet olarak pratik biçimde uygulanabilir. Kurulum ve kullanım kolaylığı, minimum karmaşıklık en büyük avantajlarındandır.

Mevcut piyasa koşullarında, Raspberry pi B+ modelin 320 TL, elektrikli kapı kilidi 60 TL, 12V adaptör 20 TL, röle kartı 5 TL olarak ele alındığında, toplam fiyat yaklaşık olarak 405 TL olmaktadır. Sonuç olarak, sadece ev için değil iş yerleri içinde kullanılabilir.

Akıllı kapı kilidi konusunda yapılan birçok çalışmada,

- Kapı kilidi için servo motor ve kapı kilidi kullanılmıştır. Bu çalışmada ise elektrikli kapı kilidi kullanılmıştır. Bunun nedeni servo motorun düzenli olarak bakımı yapılması gerekmektedir ve maliyeti düşürmek ve amaçlıdır.
- Kapının üzerine monte edilen sensörler sayesinde görüntü alınmaktadır. Sensörlerin dezavantajı kapının önünden geçen her şeyin görüntüsünü alarak ev sahibinde bulunan uygulamaya gönderecektir. Kişinin meşguliyetine sebep olacaktır. Sadece ihtiyaç durumunda kullanılması için sensör yerine “Kapıda Kaldım” android uygulaması geliştirilmiştir.
- Aynı ağda bulunan akıllı cep telefonlarının haberleşmesi dolayısıyla kapının açılma/kapanma durumu kontrol edilmektedir. Bu çalışmanın en büyük avantajlarından biri farklı ağlarda bulunan akıllı cep telefonları tarafından kapı durumu kontrol edilmektedir.
- Güvenlik unsuru arka plana atıldığı görülmektedir. OpenVpn hem Raspberry pi hemde 2 tane android işletim sistemine sahip akıllı cep telefonuna kurularak uçtan uca şifreleme yöntemiyle haberleşmeleri sağlanmıştır

Han vd. (2017) tasarladığı çalışmada sensör kullanılmıştır. Sensörün önünden geçen her canlı için sistem defalarca gereksiz çalışmasına sebep olacaktır. Uyguladığımız çalışmada bu durum göz önüne alınarak sensör kullanılmamıştır. Bunun yerine kapıda kalan kişi için ayrı bir android uygulaması geliştirilmiştir. Hussein vd. (2017) çalışmasında tuş takımına şifrenin girilmesiyle kapıdaki kişinin görüntüsü alınarak veri tabanında kayıtlı resimle karşılaştırma yapar. Çekilen resimle veri tabanındaki resim kişi aynı olduğu halde resimler uyuşmazsa kapı açılmayacaktır. Resim güncellemesi yapılması için raspberry pi’ye direk müdahale edilmesi gerekmekte uzaktan kontrolü bulunmamaktadır. Yaptığımız çalışmada ise kapı kontrolü android program ile uzaktan kontrol edilmekte. Hadis vd. (2018) yaptığı çalışmada güvenlik unsurunun göz ardı edildiği görülmektedir. Akıllı telefon ev sahibi dışında bir başkasının eline geçmesi dahilinde olası kötü sonuçlar doğurması kaçınılmaz. Önerdiğimiz çalışmada güvenlik en üst seviyede tutulmuştur. OpenVpn ve Firebase veri tabını kullanarak sağlanmıştır. Gudavalli (2020)’nin yaptığı çalışmada uzaktan erişimin olmayışı ve yapılan kapı kilidinin maliyetinin fazla olması ve malzemelerin

kullandığı enerjinin yüksek olması yaptığımız çalışmanın bu unsurlarla kıyaslandığında daha avantajlı olduğu ortaya çıkmıştır.

Uygulamanın dezavantajı olarak görülebilecek en önemli unsurlar internet bağlantısının kopması veya elektriğin gitmesi olarak düşünülebilir. Kapıda kalan kişinin görüntüsü alınırken hareket etmesi ya da ışığın az olması fotoğrafın düzgün çekilmemesine sebep olacaktır. Aktif enerji tüketimi az olduğundan, elektrik kesintilerine karşı önlem olarak maliyeti düşük bataryalar kullanılabilir.

KAYNAKLAR

- Adalan, K., Erkmen, B., 2016. Face Recognition, NFC and Voice Controlled Door Lock System, 2016 National Conference on Electrical, Electronics and Biomedical Engineering (ELECO), 1-3 December, Bursa.
- Ahmed, T., Tammam, A., Nuruddin, B., Latif A.B., ; Arnob, S.S.; Rahman, R., 2020. A Real-Time Controlled Closed Loop IoT Based Home Surveillance System for Android using Firebase, 2020 6th International Conference on Control, Automation and Robotics (ICCAR), 18-20 October, Bangaluru.
- Ahtsham, M., Yan, H.Y., Ali, U., 2019. IoT Based Door Lock Surveillance System Using Cryptographic Algorithms, 2019 IEEE 16th International Conference on Networking, Sensing and Control (ICNSC), 9-11 May, Banff.
- Akıllı Şehir Uygulamaları, 2020. Erişim Tarihi: 07.04.2020. <https://ioturkiye.com/2017/06/akilli-sehir-uygulamalari/>
- Aman, F., Anitha, C., 2017. Motion Sensing and Image Capturing based Smart Door System on Android Platform, 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), 1-2 August, Chennai
- Android Studio ile her Android cihaz için yüksek kalitede uygulama geliştirme, 2020. Erişim Tarihi: 10.04.2020. <https://developer.android.com/distribute/best-practices/develop/build-with-android-studio>
- Arifin, R., Sarno, R., 2018. Door Automation System Based on Speech Command and PIN using Android Smartphone, 2018 International Conference on Information and Communications Technology (ICOIACT), 6-7 March, Yogyakarta.
- Ashton, K., 2009. That 'Internet of Things' Thing. Erişim Tarihi: 03.02.2020. <http://www.rfidjournal.com/articles/pdf?4986>
- Aws IoT, 2020. Erişim Tarihi: 12.03.2020. <https://aws.amazon.com/tr/iot/>
- Bayram, U., 2006. Akıllı Ev Otomasyonu, Çanakkale Onsekiz Mart Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 84, Çanakkale.
- Bringing MySQL to the web, 2020. Erişim Tarihi: 03.02.2020. <https://www.phpmyadmin.net>
- Flask , 2020. Erişim Tarihi: 05.04.2020. <https://pypi.org/project/Flask/>
- Firebase, 2020. Erişim Tarihi: 05.02.2020. <https://firebase.google.com>

- Girme, G.S., Patil, S.R., 2019. Internet of Things Based Intelligent Security using Android Application, 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT), 27-29 November, Tirunelveli.
- Gudavalli, D.K.P., Monica, I.S., Sagar, M.E.C.V, 2019. A Novel Door Lock Operation Using Two Staged Smart Security Verification, 2019 Global Conference for Advancement in Technology (GCAT), 18-20 October, Bangaluru.
- Hadis, M. S., Palantei, E., Ilham ,A. A., Hendra, A., 2018. Design of Smart Lock System for Doors with Special Features using Bluetooth Technology, 2018 International Conference on Information and Communications Technology (ICOIACT), 6-7 March, Yogyakarta.
- Han, D., Kim, H., Jang, J., 2017. Blockchain based Smart Door Lock system, 2017 International Conference on Information and Communication Technology Convergence (ICTC), 18-20 October, Jeju.
- Hussein, N.A., Mansoori, I.A.,2017. Smart Door System for Home Security Using Raspberry pi3, 2017 International Conference on Computer and Applications (ICCA), 6-7 September, Doha.
- iTaksi nedir?, 2020. Erişim Tarihi:08.05.2020. <https://www.itaksi.com/yolcu-sss.html#!iTaksi-nedir>
- Jahnavi, S., Nandini, C., 2019. Smart Anti-Theft Door locking System, 2019 1st International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE), 19-20 March, Bangalore.
- Kavde, S., Kavde, R., Bodare, S., Bhagat G.,2017. Smart Digital Door Lock System Using Bluetooth Technology, 2017 International Conference on Information Communication and Embedded Systems (ICICES), 23-24 February, Chennai.
- Kim, T., Park, H., Hong, S.H., Chung, Y., 2013.Integrated System of Face Recognition and Sound Localization for A Smart Door Phone. IEEE Transactions on Consumer Electronics ,59(3)
- King, N., 2003. Smart home a definition, 2003 Intertek Research and Testing Center.
- Kutup, N., 2011, Nesnelerin Interneti; 4H, Her yerden, Herkesle, Her Zaman, Her nesne ile bağlantı, 16. Türkiye’de İnternet Konferansı inet-tr’11.
- Lita, A.I., Visan, D.A., Mazare, A.G.,2017. Door Automation System for Smart Home Implementation, 2017 23rd International Symposium for Design and Technology in Electronic Packaging (SIITME), 26-29 October, Constanta.
- Mathew, M., Divya, R.S., 2017. Super secure door lock system for critical zones. 2017 International Conference on Networks & Advances in Computational Technologies (NetACT), 20-22 July, Thiruvanthapuram.

- Mrinal, M., Priyanka L., Saniya, M., Poonam, K., Gavali, A.B., 2017. Smart Home – Automation and Security System Based on Sensing Mechanism, 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), 22-24 February, Coimbatore.
- Muthumari, M., Sah, N.K., Raj, R., Saharia J., 2018. Arduino Based Auto Door Unlock Control System by Android Mobile Through Bluetooth and Wi-Fi, 2018 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 13-15 December, Madurai.
- OpenCV About, 2020. Erişim Tarihi: 07.04.2020. <https://opencv.org/about/>
- Park, Y.T., Sthapit, P., Pyun, J.Y., 2009. Smart Digital Door Lock for the Home Automation. TENCON 2009 - 2009 IEEE Region 10 Conference, 23-26 January, Singapore.
- Rajadurai, S., Nehru, P.P., Selvarasu, R., 2015. Android Mobile Based Home Security and Device Control Using GSM, 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), 19-20 March, Coimbatore.
- Pinjala, S.R., Gupta, S., 2019. Remotely Accessible Smart Lock Security System with Essential Features, 2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET), 21-23 March, Chennai.
- Raspberry pi3 B+ model, 2020, Erişim Tarihi: 03.02.2020. <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>
- Raspbian 2020. Erişim Tarihi: 03.02.2020. <https://www.raspberrypi.org/documentation/raspbian/>
- Sağlıkta IoT Çözümleri, 2020. Erişim Tarihi: 07.04.2020. <https://ioturkiye.com/2017/06/saglikta-iot-cozumleri/>
- Shashank, K., Gopalakrishna, M.T., Hanumantharaju, M.C., 2015. i-Door: Intelligent Door Based on Smart Phone, 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications, Bhubaneswar, India, 14-15 November, 2014
- Stefanov, D.H., Bien, Z., Bang, W.C., 2004. The smart house for older persons and persons with physical disabilities: structure, technology arrangements, and perspectives. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 12(2), 228-250.
- The Story of the Trojan Room Coffee Pot, 2020. Erişim Tarihi: 03.02.2020. <http://www.cl.cam.ac.uk/coffee/qsf/timeline.html>
- This is OpenVpn, 2020. Erişim Tarihi: 03.02.2020. <https://openvpn.net/about/>

- Tilala, P., Roy, A.K., Das, M.L., 2017. Home Access Control through a Smart Digital Locking-Unlocking System, TENCON 2017-2017 IEEE Region 10 Conference, 5-8 November, Penang.
- Vaidya, B., Patel, A., Panchal, A., Mehta, R., Mehta, K., Vaghasiya, P., 2017. Smart Home Automation with a Unique Door Monitoring System for old age People using Python, OpenCV, Android and Raspberry pi, 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), 15-16 June, Madurai.
- Vongchumyen, C., Watanachaturaporn, P., Jinjakam, C., Watcharapupong, A., Kasemsiri, W., Tongprasert, K., Walairacht, A., Penpokai, T., Jenweerawat, T., Hami, A., 2017. Door lock system via web application, 2017 International Electrical Engineering Congress (iEECON), 8-10 March, Pattaya.
- Vujoic, V., Makshimovic, M., 2015. Raspberry Pi as a sensor Web node for home automation, Elsevier Transaction on Computer and Engineering, 42, 153-171.
- What is the ASF?, 2020. Erişim Tarihi: 24.04.2020.
<https://www.apache.org/foundation/>
- What is the internet of things?, 2020 Erişim Tarihi: 24.01.2020.
<https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/>
- What is IoT?, 2020. Erişim Tarihi: 23.01.2020. <https://www.oracle.com/internet-of-things/what-is-iot.html>
- What is Raspberry pi, 2020, Erişim Tarihi: 03.02.2020.
<https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>
- Yashwant, S.K., Krishna, P.V., Kumar, B.N.V. S.B., Chandan, G., Prasad, J.V.D., 2020. iLock: State-of-the-art Sophisticated Door Lock for Wireless Devices. 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), 5-7 March, Bangalore.
- Zhao, J.C., Zhang, J.F., Feng, Y., Guo, J.X., 2010. The Study and Application of the IOT Technology in Agriculture. 3rd International Conference on Computer Science and Information Technology (ICCSIT), 9-11 July, Chengdu.

EKLER

EK A. Kodlar

EK A. Kodlar

pinKontrol.py

```
import time
import cv2
import smtplib
import os.path
import RPi.GPIO as GPIO
from email.mime.base import MIMEBase
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from email import encoders
from flask import Flask
from flask_responses import json_response, xml_response,
auto_response
import datetime

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

zil=19

#GPIO.setup(uc,GPIO.OUT)
GPIO.setup(zil,GPIO.IN)
#GPIO.output(uc, GPIO.LOW)
zilCal=GPIO.input(zil)
print(zilCal)
if zilCal==0:
    camera = cv2.VideoCapture(0)
    return_value,image = camera.read()
    cv2.imwrite('test1.jpg',image)
    camera.release()
    cv2.destroyAllWindows()

    email='proje.raspberry@gmail.com'
    password='proje.deney'

    sendToMail='proje.human@gmail.com'
    subject='Kapda ki kim'
    message='Resmi kontrol et! - http://10.8.0.9/test.php'

    fileLocation='/home/pi/Desktop/ornek/test1.jpg'

    msg=MIMEMultipart()
    msg['From']=email
    msg['To']=sendToMail
    msg['Subject']=subject

    msg.attach(MIMEText(message,'plain'))
    print(msg)

    fileName="test1.jpg"
    attachment = open(os.getcwd()+'/test1.jpg', "rb")
    part = MIMEBase('application', 'octet-stream')
    part.set_payload(attachment.read())
```

```

        encoders.encode_base64(part)
        part.add_header('Content-
Disposition',"attachment;fileName=%s" %fileName)
        msg.attach(part)

        server=smtplib.SMTP("smtp.gmail.com",587)
        print(server)
        server.starttls()
        server.login(email, password)
        text=msg.as_string()
        server.sendmail(email, sendToMail,text)

        server.quit()
        open("app.py","r")

```

app.py

```

import RPi.GPIO as GPIO
from flask import Flask, render_template, request
import datetime
print('abc')
app = Flask(__name__)

GPIO.setmode(GPIO.BCM)

pins = {
    7 : {'name' : 'GPIO 7, 'state' : GPIO.LOW},

}

for pin in pins:
    GPIO.setup(pin, GPIO.OUT)
    GPIO.output(pin, GPIO.LOW)

@app.route("/")
def main():

    for pin in pins:
        pins[pin]['state'] = GPIO.input(pin)

    templateData = {
        'pins' : pins
    }
    return render_template('index.html', **templateData)

@app.route("/<changePin>/<action>")
def action(changePin, action):

    changePin = int(changePin)

    deviceName = pins[changePin]['name']

    if action == "on":

        GPIO.output(changePin, GPIO.HIGH)

```

```

        message = "Turned " + deviceName + " on."
    if action == "off":
        GPIO.output(changePin, GPIO.LOW)
        message = "Turned " + deviceName + " off."

    for pin in pins:
        pins[pin]['state'] = GPIO.input(pin)

    templateData = {
        'pins' : pins
    }

    return render_template('index.html', **templateData)

if __name__ == "__main__":
    app.debug=True
    app.run(host='0.0.0.0', port=5000)

```

kapiac.py

```

import RPi.GPIO as gp
import time
import json
gp.setwarnings(False)
pin = 7

gp.setmode(gp.BOARD)
gp.setup(pin, gp.OUT)

def kapi_acik(pin):
    gp.output(pin, gp.HIGH)

    print('acik')

def kapi_kapali(pin):
    gp.output(pin, gp.LOW)
    gp.cleanup()
    print('kapali')

if __name__ == '__main__':
    try:
        kapi_acik(pin)
        time.sleep(1)
        kapi_kapali(pin)

        print('hatasiz')

    except KeyboardInterrupt:
        gp.cleanup()
        print('hatali')

```

test.php

```
<?php
try{
    exec("sudo python kapiac.py");
    echo 'SUCCESFULLY OPEN THE GATE';
}
catch (Exception $e) {
    echo $e->getMessage();
}
?>
```

test2.php

```
<!DOCTYPE html>
<head>
</head>
<body>
<?php
try{
    exec("sudo python3 pinKontrol.py");
    header("Location: http://10.8.0.9/bekle.html");
}
catch (Exception $e) {
    echo $e->getMessage();
}
?>
</body>
</html>
```

bekle.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"><html><head><META http-equiv="Content-Type"
content="text/html; charset=utf-8"></head><body>

<div>
<center>
<div style="width:90%">
<div>
<a href="http://test.php" target="_blank"> Turn ON GATE</a>
</div>
</div></center>

</div>
</body></html>
```

index.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"><html><head><META http-equiv="Content-Type"
content="text/html; charset=utf-8"></head><body>

<div>
<center>
<div style="width:90%">
<div>
<a href="http://test2.php" target="_blank"> CHECK GATE</a>
</div>
</div></center>
<h1></h1>
</div>
</body></html>
```

MainActivity.java

```
public class MainActivity extends AppCompatActivity {

    EditText emailText,passwordText;

    private FirebaseAuth mAuth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        emailText = findViewById(R.id.user_email_edit_text);
        passwordText =
findViewById(R.id.user_password_edit_text);
        mAuth = FirebaseAuth.getInstance();

        FirebaseUser user = mAuth.getCurrentUser();

        if (user != null){
            Intent intent = new Intent(getApplicationContext(),
KapiAc.class);
            startActivity(intent);
        }
    }
}
```

```

        public void signIn(View view) {

 mAuth.signInWithEmailAndPassword(emailText.getText().toString(),p
 asswordText.getText().toString()).addOnCompleteListener(this, new
 OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult>
task) {

                if (task.isSuccessful()){

                    Intent intent = new
Intent(getApplicationContext(), KapiAc.class);
                    startActivity(intent);

                }

                else {

Toast.makeText(MainActivity.this, "Failed", Toast.LENGTH_LONG).show
();

                }

            }
        });
    }
}

```

KapiAc.java

```

import android.app.Activity;
import android.os.Bundle;
import android.webkit.WebView;
public class KapiAc extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_chat);

        WebView webview = (WebView) findViewById(R.id.webview);
        webview.getSettings().setJavaScriptEnabled(true);
        webview.loadUrl("http://10.8.0.9/index.html");

    }

}

```

ProfileActiviyt.java

```
import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.net.Uri;
import android.provider.MediaStore;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.v4.app.ActivityCompat;
import android.support.v4.content.ContextCompat;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

import com.google.android.gms.auth.api.signin.internal.Storage;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;
import com.squareup.picasso.Picasso;

import java.io.IOException;
import java.util.HashMap;
import java.util.UUID;

public class ProfileActivity extends AppCompatActivity {

    EditText ageText;
    ImageView userImageView;
    Uri selected;
    private FirebaseDatabase database;
    private DatabaseReference databaseReference;
    private StorageReference storageReference;
    private FirebaseAuth mAuth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```

        setContentView(R.layout.activity_profile);

        ageText = findViewById(R.id.ageText);
        userImageView = findViewById(R.id.userImageView);

        database = FirebaseDatabase.getInstance();
        databaseReference = database.getReference();
        storageReference =
        FirebaseStorage.getInstance().getReference();
        mAuth = FirebaseAuth.getInstance();

        getData();

    }

    public void getData() {
        DatabaseReference newReference =
        database.getReference("Profiles");
        newReference.addValueEventListener(new
        ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot
        dataSnapshot) {

                for(DataSnapshot ds: dataSnapshot.getChildren())
                {
                    HashMap<String,String> hashMap =
                    (HashMap<String,String>) ds.getValue();

                    String username = hashMap.get("useremail");

                    if(username.matches(mAuth.getCurrentUser().getEmail().toString())
                    ){

                        String userAge = hashMap.get("userage");
                        String userImageUrl =
                        hashMap.get("userimageurl");

                        if(userAge !=null && userImageUrl
                        !=null) {

                            ageText.setText(userAge);

                            Picasso.get().load(userImageUrl).into(userImageView);

                        }

                    }

                }

            }

        }

    }

```

```

        }

        @Override
        public void onCancelled(@NonNull DatabaseError
databaseError) {

        }

    });

}

public void upload (View view){

    final UUID uuidImage = UUID.randomUUID();

    String imageName = "image/"+uuidImage+".jpg";
    StorageReference newReference =
storageReference.child(imageName);

    newReference.putFile(selected).addOnSuccessListener(new
OnSuccessListener<UploadTask.TaskSnapshot>() {
        @Override
        public void onSuccess(UploadTask.TaskSnapshot
taskSnapshot) {

            StorageReference profileImageRef =
FirebaseStorage.getInstance().getReference("image/"+uuidImage+".j
pg");

profileImageRef.getDownloadUrl().addOnSuccessListener(new
OnSuccessListener<Uri>() {
            @Override
            public void onSuccess(Uri uri) {
                String downloadURL =
uri.toString();

                UUID uuid =
UUID.randomUUID();

                String uuidString =
uuid.toString();

                String userAge =
ageText.getText().toString();

                FirebaseUser user =
mAuth.getCurrentUser();

                String useremail =
user.getEmail().toString();
            }
        }
    });
}

```

```

databaseReference.child("Profiles").child(uuidString).child("user
imageurl").setValue(downloadURL);

databaseReference.child("Profiles").child(uuidString).child("user
age").setValue(userAge);

databaseReference.child("Profiles").child(uuidString).child("user
email").setValue(useremail);

Toast.makeText(getApplicationContext(),"Uploaded!",Toast.LENGTH_L
ONG).show();

        Intent intent = new
Intent(getApplicationContext(),ChatActivity.class);
        startActivity(intent);
    }
});

    }
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception e) {

Toast.makeText(getApplicationContext(),e.getMessage().to
String(),Toast.LENGTH_LONG).show();
    }
});

}

public void selectPicture (View view){

    if (ContextCompat.checkSelfPermission(this,
Manifest.permission.READ_EXTERNAL_STORAGE) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this,new
String[]{Manifest.permission.READ_EXTERNAL_STORAGE},1);
    } else {

        Intent intent = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
        startActivityForResult(intent,2);
    }
}

@Override
public void onRequestPermissionsResult(int requestCode,
@NonNull String[] permissions, @NonNull int[] grantResults) {

```

```

        if(requestCode == 1) {
            if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                Intent intent = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                startActivityForResult(intent,2);
            }
        }

        super.onRequestPermissionsResult(requestCode,
permissions, grantResults);
    }

    @Override
    protected void onActivityResult(int requestCode, int
resultCode, @Nullable Intent data) {

        if (requestCode == 2 && requestCode == RESULT_OK && data
!= null) {
            selected = data.getData();

            try {
                Bitmap bitmap =
MediaStore.Images.Media.getBitmap(this.getContentResolver(),selec
ted);

                userImageView.setImageBitmap(bitmap);
            } catch (IOException e) {
                e.printStackTrace();
            }

        }

        super.onActivityResult(requestCode, resultCode, data);
    }
}

```

RecyclerViewAdapte.java

```

import android.support.annotation.NonNull;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import java.util.List;

```

```

public class RecyclerViewAdapter extends
RecyclerView.Adapter<RecyclerViewAdapter.MyViewHolder> {

    private List<String> chatMessages;

    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup
parent, int i) {

        View itemView =
LayoutInflater.from(parent.getContext()).inflate(R.layout.recycle
r_list_row,parent,false);

        return new MyViewHolder(itemView);
    }

    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder,
int position) {

        String chatMessage = chatMessages.get(position);
        holder.chatMessage.setText(chatMessage);

    }

    @Override
    public int getItemCount() {
        return chatMessages.size();
    }

    public RecyclerViewAdapter(List<String> chatMessages) {
        this.chatMessages = chatMessages;
    }

    public class MyViewHolder extends RecyclerView.ViewHolder{

        public TextView chatMessage;
        public MyViewHolder(View itemView){
            super(itemView);

            chatMessage =
itemView.findViewById(R.id.recycler_view_text_view);

        }
    }
}

```

ÖZGEÇMİŞ

Adı Soyadı : Ayşegül KALYONCU
Doğum Yeri ve Yılı : RİZE, 13/08/1987
Medeni Hali : Bekar
Yabancı Dili : İngilizce
E-posta : kalyoncu.aysegul.ogr@gmail.com



Eğitim Durumu

Lise : Çayeli Vakıfbank Lisesi, 2004
Lisans : Gazi Üniversitesi, Endüstriyel Sanatlar Fakültesi,
Bilgisayar Öğretmenliği Bölümü, 2013
Yüksek Lisans : İstanbul Ticaret Üniversitesi,
Fen Bilimleri Enstitüsü,
Bilgisayar Mühendisliği Anabilim Dalı

Mesleki Deneyim

Milli Eğitim Bakanlığı,
Bilişim Teknolojileri Öğretmeni 2018-...(devam ediyor)

Yayınları

Kalyoncu, A., Turan, M., 2020. IoT Teknolojisi Kullanan Pratik ve Güvenilir Akıllı Kapı Kilidi Tasarımı. 2nd International Congress on Human-Computer Interaction, Optimization and Robotic Applications, 26-27 Haziran, Ankara