

**AKILLI EVLER İÇİN  
MOBİL UYGULAMA GELİŞTİRİLMESİ**

YÜKSEK LİSANS TEZİ  
Ali ERBEY

DANIŞMAN

Doç. Dr. Uçman ERGÜN

İNTERNET VE BİLİŞİM TEKNOLOJİLERİ  
YÖNETİMİ ANABİLİM DALI

Ocak, 2016

Bu tez çalışması 14.FEN.BİL.16 numaralı proje ile Afyon Kocatepe Üniversitesi  
Bilimsel Araştırma Projeleri Koordinasyon Birimi tarafından desteklenmiştir.

**AFYON KOCATEPE ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**YÜKSEK LİSANS TEZİ**

**AKILLI EVLER İÇİN MOBİL UYGULAMA GELİŞTİRİLMESİ**

**Ali ERBEY**

**DANIŞMAN**

**Doç.Dr. Uçman ERGÜN**

**İNTERNET VE BİLİŞİM TEKNOLOJİLERİ YÖNETİMİ**  
**ANABİLİM DALI**

**Ocak, 2016**

## TEZ ONAY SAYFASI

Ali ERBEY tarafından hazırlanan “Akıllı Evler İçin Mobil Uygulama Geliştirilmesi” adlı tez çalışması lisansüstü eğitim ve öğretim yönetmeliğinin ilgili maddeleri uyarınca 28/01/2016 tarihinde aşağıdaki jüri tarafından oy birliği ile Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü **İnternet ve Bilişim Teknolojileri Yönetimi Anabilim Dalı’nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

**Danışman** : Doç. Dr. Uçman ERGÜN

**Başkan** : Doç. Dr. Uçman ERGÜN  
Afyon Kocatepe Üniversitesi, Mühendislik Fak.

**Üye** : Doç. Dr. Sabri Koçer  
Necmettin Erbakan Üniversitesi, Mühendislik-Mimarlık Fak.

**Üye** : Yrd. Doç. Dr. Uğur FİDAN  
Afyon Kocatepe Üniversitesi, Mühendislik Fak.

Afyon Kocatepe Üniversitesi  
Fen Bilimleri Enstitüsü Yönetim Kurulu’nun  
...../...../..... tarih ve  
..... sayılı kararıyla onaylanmıştır.

.....  
Prof. Dr. Hüseyin ENGİNAR  
Enstitü Müdürü

**BİLİMSEL ETİK BİLDİRİM SAYFASI**  
**Afyon Kocatepe 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 bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

**beyan ederim.**

**28/01/2016**

**İmza**  
**Ali ERBEY**

**ÖZET**  
Yüksek Lisans Tezi

**AKILLI EVLER İÇİN MOBİL UYGULAMA GELİŞTİRİLMESİ**

Ali ERBEY

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü

İnternet ve Bilişim Teknolojileri Yönetimi Anabilim Dalı

**Danışman:** Doç. Dr. Uçman ERGÜN

Günümüzde çeşitli algılayıcılar aracılığıyla mobil teknolojilerin gelişimi, teknolojik cihazlar arasındaki etkileşimi artırmış ve bu etkileşimler günlük yaşamımızın her köşesinde yer almaya başlamıştır. Bu şekilde her biri kendi içinde ayrı ayrı özellik ve görev sergileyen bazı teknolojiler, içinde yaşadığımız ortamlara entegre edilmekte ve günlük alışkanlıklarımızın yönetiminde rol oynamaktadır. Akıllı evler, evinizde oturduğunuz yerden aydınlatma sistemlerini kontrol etmek, sabah kalktığınızda perdeleri otomatik olarak açmak, dışarı çıktığımızda ütüyü açık unuttuğunuz düşüncesi ile geri dönmeden tehlikeyi engellemek, bahçenizde çimlerin nem oranına göre otomatik sulanmasını sağlamak gibi pek çok rutin ev işlerini tek bir tuşla yönetebilen sistemidir.

Literatürdeki çalışmalar incelendiğinde; kullanıcıların akıllı ev sistemlerini bir mobil uygulama aracılığıyla kontrol edebildikleri çalışmalara rastlanılmasına rağmen; kullanıcılardan ziyade akıllı ev sistem geliştiricilerini hedef kitle olarak belirleyen ve onlara hizmet olarak yazılmış mobil uygulamalara pek rastlanmamıştır. Dahası var olan standart protokollerinin yüksek maliyette olması geliştiriciler için bir dezavantaj olabilmektedir. Bu nedenlerle, kendi protokollerini yazacak geliştiricilere, akıllı ev test aşamasını hızlandırma adına mobil uygulama standartlarını kullanarak ortak paydada buluşturabilecek bir mobil uygulama hazırlamak önemli olarak görülebilir. Dolayısıyla bu tezde, akıllı ev geliştiricilerinin akıllı ev teknolojisindeki kontrol sistemlerini takip etmede kullanabilecekleri bir mobil uygulama geliştirmek amaçlanmıştır.

Çalışma kapsamında bir akıllı ev teknolojisinin kontrol sistemlerini geliştirmeye yönelik Android tabanlı bir mobil uygulama hazırlanmış, kullanım kolaylığı ve kodların okunabilirliği üzerine prototip test edilmiş ve karşılaşılan sorunlar giderilmiştir. Nihai haline erişen mobil uygulama akıllı ev yönetim sistemlerinde yer alan aydınlatma, perde-panjur, sulama, güvenlik, klima gibi menülerin yanı sıra senaryolar, sesle kontrol gibi gelişmiş özellikleri de barındırmaktadır.

**2016, x + 64 sayfa**

**Anahtar Kelimeler:** Akıllı ev, Mobil uygulama, Android.

**ABSTRACT**  
M.Sc. Thesis

DEVELOPING MOBILE APPLICATION FOR SMART HOME

Ali ERBEY

Afyon Kocatepe University

Graduate School of Natural and Applied Sciences

Department of Internet and Information Technologies Management

**Supervisor:** Assoc. Prof. Dr. Uçman ERGÜN

The development of mobile technologies via the various of sensors have rendered interaction invisible between technological devices and those interactions started to playing a part in everywhere of daily life. In this way, invisibly kinds of technology having different features in itself, being integrated into environments we live in and taking a role in management of our habits and routines under name of “Smart Home”. Smart home is a system which can manage remotely various routine housework like controlling lighting while keeping your seat, opening curtains automatically when you woke up, preventing the danger without return home when you forget iron opened, providing the watering garden automatically based on the rate of humidity of grasses.

However there have been different studies about remote controlled smart homes by users via mobile applications, it has been noted that there are limited studies that addresses on developers rather than users of smart homes and mobile application for the developers. Moreover, since the current standard protocols are at high cost it can be a disadvantage for the developers. Therefore, developing a ready-made application can be important which can find the lowest common denominator for developers by using mobile application standards for stepping up test process of smart home, similarly to developers’ who will write their own protocols. Consequently, in this study it was aimed to develop a mobile application for the smart home developers to use following the smart home technologies.

At the end of the study, an Android mobile application was developed aimed at improving control systems of a smart home technology, the prototype was tested for

ease of use and readability of codes and the encountered problems were solved. The finalized mobile application included menus like lightning, curtain-shutter, safety, air conditioning as well as improved properties like scenarios, control with voice.

**2016, x + 64 pages**

**Key Words:** Smarthome, Mobile application, Android.



## TEŐEKKÜR

Bu arařtırmanın konusu, deneysel alıřmaların ynlendirilmesi, sonuların deęerlendirilmesi ve yazımı ařamasında yapmıř olduęu byk katkılarından dolayı tez danıřmanım Sayın Do. Dr. Uman ERGN'e, arařtırma ve yazım sresince yardımlarını esirgemeyen Arř. Gr. Sevil ORHAN ZEN'e ve ZEN ailesine, her konuda neri ve eleřtirileriyle yardımlarını grdęm Yrd. Do. Dr. Gzin MAZMAN ve arkadařlarıma teőekkr ederim. Ayrıca bu tez alıřmasını 14.FEN.BİL.16 numaralı proje ile destekleyen Afyon Kocatepe niversitesi Bilimsel Arařtırmalar Koordinasyon Birimine de teőekkr ederim.

Ali ERBEY

AFYONKARAHİSAR, 2016

## İÇİNDEKİLER DİZİNİ

	<b>Sayfa</b>
ÖZET .....	i
ABSTRACT .....	iii
TEŞEKKÜR .....	v
İÇİNDEKİLER DİZİNİ.....	vi
KISALTMALAR DİZİNİ .....	viii
ŞEKİLLER DİZİNİ .....	ix
RESİMLER DİZİNİ .....	x
1. GİRİŞ.....	1
2. LİTERATÜR BİLGİLERİ .....	3
2.1 Akıllı Evler .....	3
2.2 Akıllı Evlerin Gelişimi .....	3
2.3 Evin Akıllanması ve Kullanım Alanları .....	4
2.4 Akıllı Evlerin Avantaj ve Dezavantajları .....	6
2.5 Akıllı Evlerin Mobil Uygulamalar ile Yönetimi .....	8
2.6 Günümüzde Var Olan Akıllı Evler.....	9
3. MATERYAL ve METOT .....	11
3.1 Uygulamanın Tasarımı İçin İhtiyaçların Belirlenmesi .....	11
3.1.1 Yazılım Gereksinimlerinin Belirlenmesi .....	12
3.1.2 Ağ Programlama Protokolleri .....	13
3.2 Tasarım ve Geliştirme .....	14
3.2.1 Kurulum ve Sisteme Giriş.....	15
3.2.2 Aydınlatma Kontrolleri .....	18
3.2.3 Perde/Panjur Kontrolleri .....	19
3.2.4 Priz Kontroller .....	22
3.2.5 Klima.....	22
3.2.6 Sulama Sistemleri Kontrolleri.....	22
3.2.7 Senaryolar .....	23
3.2.8 Kameralar.....	23
3.2.9 Medya .....	25
3.2.10 Sensörler .....	25
3.2.11 Odalar ve Daireler.....	25
3.3 Elektronik Uygulayıcı .....	25
3.3.1 Elektronik Uygulayıcı Kart Donanım Özellikleri.....	27
3.3.2 Elektronik Uygulayıcı Kartı Yazılım Özellikleri.....	27

3. 4 Akıllı Ev Uygulaması Geliştiricilerinin Ortamı Özelleştirilmesi .....	28
4. BULGULAR .....	29
4.1 Mobil Uygulamanın Test Edildiği Akıllı Ev Sisteminin Mevcut Diğer Akıllı Evlerden Farkı .....	29
5. SONUÇ VE ÖNERİLER .....	32
5.1 Sonuç .....	32
5.2 Öneriler.....	33
6. KAYNAKLAR.....	34
6.1 İnternet Kaynakları .....	35
ÖZGEÇMİŞ.....	36
EKLER .....	37

## KISALTMALAR DİZİNİ

### Kısaltmalar

---

TCP	Transmission Control Protocol (İletim Kontrol Protokolü)
UDP	User Datagram Protocol (Kullanıcı Veribloğu İletişim Kuralları)
BKB	Bireyselleştirilmiş Kod Bloğu
API	Application Programing Interface - Uygulama Programlama arayüzü
GPS	Global Positioning System; Küresel Konumlama Sistemi
SQL	Structured Query Language, Yapılandırılmış Sorgu Dili
MCU	Micro Controller Unit (Mikro Kontrol Birimi)
TCP/IP	Transmission Control Protocol/İnternet Protocol (İletim Kontrol Protokolü/İnternet Protokolü)
PCB	Baskı Devre Kartı
EEPROM	Electronically Erasable Programmable Read-Only Memory (Elektronik olarak programlanabilir silinebilir hafıza)

---

## ŞEKİLLER DİZİNİ

	<b>Sayfa</b>
<b>Şekil 3.1</b> Veri tabanı şeması.....	18
<b>Şekil 3.2</b> Sistemin akış diyagramı .....	19
<b>Şekil 3.3</b> TCPSetsend komut satırları.....	20

## RESİMLER DİZİNİ

	<b>Sayfa</b>
<b>Resim 3.1</b> Kullanıcı Girişi ve Yönetim Sayfası .....	15
<b>Resim 3.2</b> Kurulum Sayfalarında Örnek Ekran Görüntüleri.....	16
<b>Resim 3.3</b> Sisteme Giriş ve Ana Ekran .....	20
<b>Resim 3.4</b> Aydınlatma Kontrolü Ekran Görüntüsü.....	21
<b>Resim 3.5</b> Perde-Panjur Kontrolleri.....	21
<b>Resim 3.6</b> Priz kontrolü .....	22
<b>Resim 3.7</b> Sulama kontrolü .....	23
<b>Resim 3.8</b> Önceden tanımlanmış senaryolara ait ekran görüntüsü .....	24
<b>Resim 3.9</b> Kamera menüsüne ait ekran görüntüsü.....	24
<b>Resim 3.10</b> Aktif olan algılayıcılara ait ekran görüntüsü .....	26
<b>Resim 3.11</b> Odalara ait ekran görüntüsü .....	26
<b>Resim 3.12</b> Elektronik Uygulayıcı Kartın Görüntüsü.....	27
<b>Resim 4.1</b> Sistemin test süreci .....	30
<b>Resim 4.2</b> Sistemin kablolama süreci .....	31

## 1. GİRİŞ

Günümüzde teknolojinin giderek gelişmesi, yalnızca insanlar arasındaki iletişim ve etkileşimi artırmamış, ayrıca sahip olduğumuz teknolojik aletlerin birbirleri ile iletişimini de gündeme taşımıştır. Özellikle çeşitli algılayıcılar aracılığıyla mobil teknolojilerin gelişimi, teknolojik cihazlar arasındaki etkileşimi artırmış ve bu etkileşimler günlük yaşamımızın her köşesinde yer almaya başlamıştır. Bu şekilde birçok görünmez teknoloji çalıştığımız ya da yaşadığımız ev ortamı gibi pek çok fiziki alanda hayatımızı kolaylaştırmaktadır. Her biri kendi içinde ayrı ayrı özellikler ve görevler sergileyen pek çok teknoloji görünmeyecek bir şekilde içinde yaşadığımız ortamlara entegre edilerek günlük alışkanlıklarımızın ya da rutinlerimizin yönetiminde rol oynamaktadır. Bu bağlamda evinizde oturduğunuz yerden aydınlatma sistemlerini kontrol etmek, sabah kalktığınızda bütün perdeleri otomatik olarak açmak, dışarı çıktığımızda ütüyü açık unuttuğunuz düşüncesi ile geri dönmeden tehlikeyi engellemek, bahçenizde çimlerin nem oranına göre otomatik sulanmasını sağlamak gibi rutin ev işlerini tek bir tuşla yönetebilen bir sisteme olan ihtiyaç artmaktadır. Günümüz teknolojilerinden yararlanılarak oluşturulan ve bu tür ihtiyaçlara yönelik sistemler, akıllı ev adı altında geliştirilmektedir. Akıllı ev teknolojisini geliştirmek için kurulan çeşitli şirketler ve kuruluşların yanı sıra, bilimsel anlamda da yeni bir çalışma alanı doğmuş ve bu tarz sistemlerin geliştirilmesi, geliştirme basamaklarının boyutları ve karşılaşılan sorunların giderilmesi üzerine çalışmalar yapılmaktadır. Akıllı ev sistemleri ilk olarak ev ve çevre güvenliğini sağlama ve bunlara ek olarak soğutma, ısıtma, aydınlatma, çocuk güvenliği, ofis kontrolü gibi pek çok işlemi gerçekleştirebilmek için kullanılmaktadır (Gül 2010). Özer (2005) çalışmasında, evde bulunan bitki ve hayvanların yem ve su ihtiyacını karşılayabildiği, evin sıcaklığını ayarlayabildiği, pişirme fırınının, klima ve kalorifer sisteminin çalıştırılabildiği, aydınlatma ve bahçe sulama gibi ihtiyaç duyulan pek çok işlemin kontrol edilebildiği, ayrıca su baskını, hırsızlık, yangın vb. durumlarda uyarıcı mesajlar gönderebilen ve sistem yönetiminin bir bilgisayar tarafından yönetildiği akıllı ev otomasyon sistemi geliştirmiştir.

Akıllı ev sistemlerinin gelişmesiyle birlikte evimizde bulunan her elektronik cihazın, mobil uygulamalar ile kontrol edilmesi şeklinde farklı ihtiyaçlar ortaya çıkmıştır. Öyle ki genel olarak günümüzde akıllı ev sistemlerinde yer alan ancak kontrol edemediğimiz

her bir teknolojinin mobil uygulamalar ile yönetilebilmesi gerekir. Bu anlamda hafta içi ayarladığınız saatte alarmın çalması, güvenlik sistemini devreden çıkarma, perdeleri açma, harekete duyarlı aydınlatmaları kapatma, kahve makinenizi çalıştırma, uzun süre evden ayrı kaldığınızda evin vanasını açık unuttuğunuzu düşünüp müdahale edebilmek vb. birçok senaryo için mobil bir uygulama aracılığıyla tanımlamalar yapılabilmesi ile bu ihtiyaçlar ile giderilmelidir. Ancak literatürde kullanıcıların akıllı ev sistemlerini bir mobil uygulama aracılığıyla kontrol edebildikleri çalışmalara rastlanılmasına rağmen, kullanıcılardan ziyade akıllı ev sistem geliştiricilerini hedef kitle olarak belirleyen ve onlara hizmet olarak yazılmış mobil uygulamalara pek rastlanmamıştır. Dahası var olan standart protokollerinin yüksek maliyette olması geliştiriciler için bir dezavantaj olabilmektedir. Bu nedenlerle geliştiriciler için benzer şekilde mobil uygulama standartları kullanarak kendi protokollerini yazmaları ve akıllı ev test aşamasını hızlandırmaları adına geliştiricileri ortak payda da buluşturabilen bir mobil uygulama hazırlanması önemlidir. Dolayısıyla bu tez çalışmasının hedef kitlesini akıllı ev geliştiricileri oluşturmuştur. Akıllı ev kullanıcıları için, bir mobil uygulama hazırlarken belirli standartlara bağlı kalınması ve bu bağlamda akıllı ev teknoloji geliştiricileri için mobil uygulama geliştirilmesi amaçlanmıştır.



## 2. LİTERATÜR BİLGİLERİ

### 2.1 Akıllı Evler

Günümüzde ev ve iş hayatımızda günlük faaliyetleri daha kolay yapabilmek için evlerde kullanılan teknolojinin kontrol edilebilmesi için otomasyon alanında tasarlanan sistemler akıllı ev otomasyon sistemlerini ortaya çıkarmıştır (İnal ve Akçayol 2009). Özellikle son yıllarda ev otomasyon ve iletişim teknolojilerindeki ilerlemeler, akıllı evler için yenilikçi çözümler üretme fırsatı sunmaktadır (Capitanelli *et al.* 2014). Endüstrinin pek çok alanında kullanılan kontrol sistemlerinin günlük hayattaki ihtiyaçlara uyarlanması ev teknolojileri ile sağlanırken, ev otomasyonu da bu teknolojilerin bireye özgü ihtiyaç ve isteklere uygulanması için kullanılır. Akıllı evlerde, ev sakinlerine özgü ihtiyaçlara yanıt veren, onların hayatlarını kolaylaştıran, daha güvenli, daha konforlu ve daha tasarruflu bir yaşam sunan evler oluşturmak için gelişmiş teknolojiler kullanılır (Kurbetçi *vd.* 2003).

Gül (2010)'e göre, akıllı evler, bir ortak kumanda aracılığıyla birçok alt sistemin bütünleşik bir biçimde çalışmasının kontrolünü sağlayan karmaşık bir sistem olarak tanımlanmıştır. Balta-Ozkan *vd.* (2013) ise akıllı evleri, ortam ısısından tutun da aydınlatma ve suyun sıcaklığını ayarlamaya kadar çeşitli günlük ev aktivitelerinin konfor ve verimliliğini artırmak için kullanılan enerjiyi, akıllı teknolojik cihaz ve ağlar aracılığıyla kontrol edebilen bir sistem olarak ifade etmiştir. Gerçekte, akıllı evler yüksek performans sağlamak, kullanıcı güvenliği geliştirmeye izin vermek, konfor ve yaşam kalitesini artırırken enerji tasarrufu ve maliyetleri azaltmak için her bir cihaz ile iş birliği yapabilir (Capitanelli *et al.* 2014). Bu anlamda akıllı ev, kablosuz uzaktan kontrolleri ile yüksek hızda veri geçişini sağlayan ve bilgi taşıma ihtiyacına yanıt veren multimedya kontrollere ve ev içinde birbiriyle uyumlu bir şekilde çalışan anahtar teknolojilere sahiptir (Liu 2014).

### 2.2 Akıllı Evlerin Gelişimi

1950'li yılların başında bina otomasyon kavramı gündeme gelmiştir (Özçekiç 2005). Bina otomasyon sistemlerinin gelişiminden sonra, bu sistemler fiziksel engeli olmayan

insanlar için ev konforunun sağlanması için düşünülmüş ve 1980'lerin başında Akıllı Ev fikrini ortaya çıkarmıştır (Stefenov *et al.* 2004). 1988 yılında ise ilk olarak bilgisayar programlarının evlere uygulanması düşüncesi ile dikkat çekmiştir (Aslan 2014). Bu anlamda akıllı ev tarihçesi çok da eski olmayan bir geçmişe sahiptir.

Akıllı evin örneklerine baktığımızda bu sektördeki ilk gelişmeleri büyük teknoloji firmaları tarafından gerçekleştirildiğini görüyoruz. Bu konuda Honeywell, Intel, Compaq, Microsoft, Mitsubishi ve Philips gibi teknoloji firmaları ortak bir standart olan HOME API (Application Programming Interface - Uygulama Programlama arayüzü)'sini oluşturmuşlardır. Bu ortak standart ara birim, evlerde var olan aydınlatma, ısıtma, güvenlik sistemleri ve diğer otomatik cihazları yönetebilecek şekilde tasarlanmıştır. Bu tasarımda programların işe koşulmasıyla, bilgisayarlar ve çevre birimlerinin kolay kullanımı için amaçlanan tak ve kullan (plug and play) mantığı ev aletleri içinde düşünülmüştür. Evde var olan her elektronik alet bu sistemin bir parçası olacak şekilde ara birimin bütünlük içinde çalışması sağlanmıştır (Aslan 2014). Bu gelişmeler doğrultusunda akıllı ev ilk gösterimi Tokyo'da elektronik fuarında gerçekleşmiştir (Bayram 2006).

Akıllı ev kavramı ve teknolojinin gelişimine paralel olarak ortaya çıkan ev aletleri; televizyon, müzik setleri, garaj kapıları için uzaktan kumandalar, kahve makineleri için zamanlayıcılar, pek çok seçenek sunan çamaşır makinesi, buzdolabı gibi hayatımızı kolaylaştırmak için daha kullanışlı hale gelmeye başladı. Artık günümüzde tüm bu cihazlar birbirleri ile haberleşerek tek bir ana merkezden yönetilmesini sağlayacak şekilde kullanıcılara sunulmaktadır (Güçül 2008). Son zamanlarda ortaya çıkan akıllı telefonlar ise bu yönetimi sağlayabilecek donanımlara sahiptir. Kendi işletim sistemlerine sahip olması, işlevselliğini ve programlanabilir özelliklerini ön plana çıkararak tek bir kaynaktan bütün evi kontrol edebilmek için günümüzde bu cihazlardan yararlanılabilir. Gelecekte ise akıllı evleri, akıllı şebekeler ve akıllı şehirler takip ederek yeni yeni servisler hayatımıza girecektir (Balta-Ozkan *et al.* 2014).

### **2.3 Evin Akıllanması ve Kullanım Alanları**

Evlerin akıllanması temel anlamda ev içerisinde bulunan elektronik cihazların birbiriyle

iletişime geçmesinin sağlanması olarak düşünülebilir. Her bir cihazın çok fonksiyonlu bir yapıya sahip olması ve aynı zamanda bu fonksiyonların programlanabiliyor olması, onların akıllı ev olarak düşünülmesi için tek başına yeterli değildir. Bir evin akıllı olabilmesi için kullanıcıların her ihtiyacını karşılayabilecek şekilde donatılması ve uzaktan kontrol edilebilen güvenlik, aydınlatma, sulama gibi çeşitli senaryolarda kendi kendini kontrol edebilmesi önemlidir. Öyle ki akıllı bir evde çeşitli algılayıcılar, denetleyiciler, ağ yoluyla iletişime geçen cihazlar ve bu cihazlar arasındaki bir bilgi aktarımının sağlanması, bu bilginin ise tek bir ana kaynaktan yönetilmesi söz konusudur.

Akıllı evler, bina yapım aşamasında döşenen elektrik sistemi ya da sonrasında hiçbir tadilat gerektirmeyen kablosuz varyasyonları aracılığıyla bir kontrol paneli şeklinde karşımıza çıkabilir. Bu bağlamda var olan sistemlerin genel olarak bir ana kontrol kutusu, ortalama bir kitap büyüklüğünde evin girişine yerleştirilen bir kontrol paneli, kontrol kutusu ile haberleşen çeşitli algılayıcı ve cihaz denetleyicileri, uzaktan kumandalar ve diğer modülleri kapsadığı söylenebilir. Hiçbir tadilata neden olmayan kablosuz ya da elektrik şebekesine sahip algılayıcı ve cihaz denetleyicilerin yanı sıra pile ihtiyaç duymayan, elektrik kesintilerinden etkilenmeyen ya da elektrik hatlarının olmadığı yerlerde de işlevselliğini devam ettirebilen kablolu algılayıcı ve cihaz denetleyicileri de bulunmaktadır (Bayram 2006). Bir akıllı ev sistemi bu çevre birimleri ile eş zamanlı olarak bilgi transferinde bulunarak onların koruma devrelerini, çalışma sürelerini, içeriklerini kontrol edebilir (Şahinoğlu 2006).

Genel olarak akıllı ev sistemleri özel gereksinime sahip insanlar için uygulanmaya başlanmış ve bu nedenle çok sayıda birçok akıllı ev çeşidi türetilmiştir. Bu anlamda farklı amaçlar için kurulan cihazların düzenlenmesine göre akıllı evler çeşitlilik göstermektedir. Bu konuda Aslan (2014) kontrol edilebilir, programlanabilir, yapay zekaya sahip evler, engelli insanlar için akıllı evler, vücut hareketlerini kullanan akıllı evler şeklinde bir sınıflama yapmıştır. Nitekim bahsedilen bu akıllı ev çeşitlerinden biri olan kontrol edilebilir evler; çeşitli uzaktan kumanda sisteminin yanında el çırpma ya da ses komutu aracılığıyla perde, ışık gibi var olan cihaz ve sistemlerin kontrol edildiği evlerdir. Programlanabilir evler; senaryoları işleyebilen, değişen koşullara tepki

verebilen evler olarak karşımıza çıkmaktadır. Bu evler ikiye ayrılıp, zamanlama ayarlı sensörlere tepki verebilen veya koşul ve duruma göre eyleme geçebilen evlerdir. Bu evler sabah olduğunda perdeleri açabilen, ev sahibinin isteklerini yapabilen evlerdir. Yapay zekaya sahip evler ise onu kullananların yaşam şekillerini taklit ederek öğrenebilen evlerdir. Kişinin sürekli olarak yaptığı işleri ya da alışkanlıklarını belirleyerek, o davranış karşısında uygun tepkiyi verebilirler. İnsanın karmaşık dünyasından dolayı günümüzde hala uygulamaya geçememiştir. Engelli insanlar için akıllı evler; fiziksel engeli olan insanların ya da günlük hayatlarında dışarıdan yardıma ihtiyacı olanların karşılaştıkları yaşam problemleriyle baş edebilmelerini sağlamak amacıyla eve yerleştirilmiş birçok akıllı cihaz sayesinde ev sakininin hem hareket etmesinde yardımcı olabilen hem de sağlık kontrolü için güvenliğini sağlayabilen evlerdir. Son olarak vücut hareketlerini kullanan akıllı evler ise, bir akıllı evdeki aydınlatma, perdeleri açma kapama gibi senaryoların kontrolü için yüz ve eller gibi insan vücudunun belli bölümlerinin hareketlerini kullanan evlerdir.

#### **2.4 Akıllı Evlerin Avantaj ve Dezavantajları**

Akıllı evler enerji tasarrufu, konfor ve güvenlik olmak üzere üç ana başlık altında insanlara yararlı olduğu söylenebilir (Aslan 2014, Bayram 2006). Akıllı ev teknolojisi'nin en büyük getirilerinden biri olan enerji tasarrufu hakkında Güğül (2008), ısıtma sistemlerinin otomasyonla denetimi sayesinde bir evin ısı enerjisi tüketiminin %10 oranında, açık olan aydınlatmaların %90 parlaklık verirken, ihtiyaç olmayan aydınlatmaların kapalı tutulması, cihazların ekonomik tarifelerine göre zamanlanarak programlanması gibi çeşitli yöntemler kullanılarak elektrik enerjisinde %30'a kadar bir tasarruf sağlanabileceğini vurgulamıştır. Aslan (2014) ise aydınlatma armatürlerinin doğru seçilmesiyle elektrik tüketimindeki bu oranın % 75'e çıkabileceğinin üzerinde durmuştur. Bunun yanı sıra evin tüm bölümlerinin aynı derecede ısınması için tek bir noktadan ısı değerini ölçen kontrollü yanma sistemlerinden yararlanılarak, kullanılmayan alanların da kapatılmasının önemli olacağını belirtmiştir. Dahası gün ışığından gerektiği kadar faydalanma ve yağmurlu günlerde bahçe sulamasının kapatılması gibi açık bırakılan cihaz ve durumların kontrolünün sağlanması ile enerji tasarrufunun önemli ölçüde artırılacağı söylenebilir. Bu konuda Fidan ve Karasekreter (2011) çalışmasında SMS kontrollü sulama otomasyonu kontrol birimi

geliştirmiş ve toprak sıcaklığının düşük olduğu sabah erken saatlerde bitkinin ihtiyacı kadar sulama yapılmasını sağlayarak %40 oranında su tasarrufu elde etmiştir.

Akıllı ev teknolojisinin getirdiği diğer bir önemli boyut ise konfordur. Bir akıllı evin konfor sağlanmasının temelinde, bireye gereksiz yere zaman kaybettiren günlük aktivitelerin otomasyon sistemi tarafından yerine getirilmesi ve normal şartlarda bireyin gerçekleştiremeyeceği pek çok işlemin yerine getirilmesi olarak düşünülebilir. Bu konudaki en büyük kolaylık, ev otomasyon sistemlerinin komutları sırasıyla gerçekleştirileceği senaryolandırma özelliğinin kullanılmasıdır (Kurbetçi vd. 2003). Bu özellik aracılığıyla tek bir merkezden evinizdeki teknolojileri oturduğunuz yerden kontrol etmek, uzakta olmanıza rağmen tamamlamanız gereken günlük rutin işlemleri senaryolar halinde yorulmadan gerçekleştirmek akıllı evlerin sağladığı önemli kolaylıklar olarak görülebilir. Öyle ki bu konuda akıllı evlerdeki sistemlerin sınırı yalnızca tasarımcısı ve kullanıcısının hayal gücü kadardır. Akşam olduğunda perdelerin kapanması, siz gelmeden evin sıcaklığının ayarlanması, gerektiğinde alarmin devreye girmesi gibi pek çok işlem akıllı ev teknolojisinin konfor boyutundaki faydalarındandır (Aslan 2014).

Son olarak güvenlik akıllı ev teknolojisinde yadsınamayacak düzeydeki diğer bir boyuttur. Akıllı evlerin güvenlik boyutunda sağladığı faydalar da yine konfor boyutundaki gibi büyük ölçüde hayal gücüne ve ev teknolojisinin kapasitesine bağlı olarak bireyin ihtiyaçlarına göre oldukça fazla türde çeşitlenebilir (Kurbetçi vd. 2003). Kullanıcı evinden uzakta iken evine girmek isteyen bir hırsızın uzaklaştırmak için senaryolar yardımı ile aydınlatmaları, müzik seti ya da televizyon gibi cihazların çalışmasını kontrol ederek ev sahiplerinin evde olduğu izlenimi verilebilir. Bu şekilde hırsızın evden uzak durması sağlanarak, evin güvenliğini sağlamak mümkün olabilmektedir. Hareket algılayıcılar, kapı ve pencerelere yerleştirilen manyetik ya da optik algılayıcılar kullanılarak tüm ev gözetim altında tutulabilir. Bu şekilde evde aktif caydırıcılar etkili olmaktadır. Dolayısıyla akıllı bir evin sağlayacağı güvenlik önlemlerinin var olan klasik alarm sistemlerine göre en büyük avantajı hırsızlık, gaz kaçağı, yangın ya da su baskını gibi tehlikelere karşı uyarılmanın yanı sıra tehlikelerin oluşmadan önlenmesidir (Bayram 2006).

Genel olarak akıllı ev teknolojilerinin avantaj ve dezavantajlarından yola çıkarak, akıllı ev sistemlerinden güvenliği ve konforunu artırma, enerji tasarrufunu sağlama, basit ve kullanımının kolay olması ya da fiziksel anlamda kaybı olan insanların hayatını kolaylaştırması gibi konularda faydalanması beklenmektedir. Ancak akıllı evlerin yararları yani avantajlarının dışında çeşitli dezavantajlarının da bulunduğu yadsınamaz bir gerçektir. Güğül (2008) bu konuda aşağıdaki maddeleri sıralamıştır:

- (i). Uzaktan erişimli sistem nedeniyle evinizin kontrolünün başkalarına geçmesi,
- (ii). Sistemin kontrolünde oluşabilecek beklenmedik aksaklıkların neden olabileceği çeşitli sonuçlar,
- (iii). Nem sensörünün hasar görmesi ile bahçenin gereğinden fazla sulanması gibi sistem arızalarının tasarruf yerine israfa yol açması,
- (iv). Sorumlulukları azaltıp, insanı düşünmemeye dolayısıyla tembelliğe iterek, monoton bir hayata sürüklemesi,
- (v). Sesle verilen komutlarda aksaklıklar,
- (vi). İnsanı mekanikleştirmesi.

## **2.5 Akıllı Evlerin Mobil Uygulamalar ile Yönetimi**

Android, açık kaynak kodlu işletim sistemi olarak mobil cihazlar için geliştirilmiş bir işletim sistemidir (Narman 2010). Bu işletim sistemi aynı zamanda mobil cihazlar için açık standartlara dayanan ve Google'ın öncülüğündeki "Open Handset Alliance" birliğini oluşturan çok sayıda donanım ve yazılımcı ile birlikte Linux çekirdeği üzerine kurulan bir sistemdir. Bunun yanı sıra, pek çok algılayıcı ve yeni donanımları kapsayacak şekilde tasarlanmış olup, kendisine özgü kullanışlı bir SQLite veritabanına sahiptir (Çavuş 2012, Taç 2011). Bu bağlamda akıllı evlerin yönetim sistemi için mobil işletim sistemi tercihi açık kaynak kodlu kütüphane kullanma özgürlüğü, geliştirilen yazılımları dağıtabilme özgürlüğü ve farklı işletim sistemlerinde (windows, linux vb.) uygulama geliştirebilme esnekliğine sahip olması nedeniyle Android daha önceliklidir (Taç 2011). Android, modern bilgisayar bilimlerinde özel bir donanım gerektirmediği için büyük bir esneklik sağlar (Goadrich and Rogers 2011). Bu nedenlerin yanı sıra akıllı cep telefonları için birden fazla işletim sistemi arasında öne çıkan Android işletim

sisteminin, uygulama geliştiriciler için rahat bir platform olması ve günümüzde yaygın olarak kullanılması nedeniyle tercih edildiği görülmektedir (Kardaş 2015). Öyle ki bu konuda Menteşoğlu vd. (2015) çalışmasında mobil cihazlarla akıllı ev sisteminin gözlenmesi ve kontrol edilmesi için android işletim sistemini kullanmış ve mobil cihaz - kontrol birimi - algılayıcılar arasındaki uygulamaya özgün bir haberleşme protokol tasarımı ile uygulanması üzerinde durmuştur.

Akıllı evlerin yönetiminde kullanılan android tabanlı mobil işletim sisteminin yanı sıra, bir diğer önemli işletim sistemi de IOS'tur. IOS işletim sistemine sahip akıllı telefonlar için bir akıllı ev uygulaması tasarlayabilmek adına Mac OS X 10.6 (Snow Leopard) işletim sistemine sahip bir Macintosh bilgisayar gerekmektedir (Goadrich and Rogers 2011). Ancak bu bilgisayar ya da işletim sisteminin yüksek maliyetli olması, geliştiriciler için bir dezavantaj olarak görülebilir. Goadrich ve Rogers (2011) geliştiricilerin Objective-C ve Xcode yerine, Java ve Eclipse'i daha çok tercih ettiklerini vurgulamıştır. Bu bağlamda iOS işletim sisteminin Android kadar fazla tercih edilmemesinin nedenleri olarak bahsedilen bu dezavantajların yanı sıra, android işletim sisteminin açık kaynak kodlu olması ve android mobil cihazların yaygın olarak kullanılması gösterilebilir.

## **2.6 Günümüzde Var Olan Akıllı Evler**

Günümüzde akıllı ev sistemi geliştirip ticari anlamda pazarlayan pek çok firma bulunmaktadır. Control4, Schneider, Berker, Fibaro gibi firmalar sektörde öne çıkan firmalardan sadece bir kaçıdır. Var olan akıllı ev firmaları, bir akıllı evde olması gereken aydınlatma, perde - panjur, iklimlendirme gibi ortak özellikleri barındırmasının yanı sıra ekstra özellikler ekleyerek sistemlerini özelleştirmektedir.

Berker firması (İnt.Kyn.1) genel hatlarıyla;

- (i). KNX protokolünü tercih etmiştir.

Control4 firması (İnt.Kyn.2) ise;

- (i). Zamanlayıcılar, hareket sensörleri ve ışık azaltıcı cihazlar (dimmer) kullanarak

- güvenliđi destekleyen aynı zamanda enerji masraflarını azaltan,
- (ii). Her noktadan sisteme erişim sunan,
  - (iii). IP kameralar ve güvenlik sistemleri ile evi izleyebilmeyi sağlayan,
  - (iv). Odadan odaya farklı müzik sistemi sunan,
  - (v). Ev sinema sistemi ile birden çok kumanda kullanımını ortadan kaldıran,
  - (vi). ZigBee desteđine sahip olan bir sisteme sahiptir.

Fibarö firması (İnt.Kyn.3) ise:

- (i). Kablosuz modüllere,
- (ii). Z-wave ile kablosuz ve kolayca haberleşebilen elektronik cihazlara,
- (iii). GPS konumuna bađlı olarak eve dođru yaklaştığınızda evinizi ısıtan bir sisteme sahiptir.

Schneider Electric firması (İnt.Kyn.4) ise;

- (i). Uzaktan kontrol edilebilen,
- (ii). Hırsızları yanıltmak için evde hareket dedektörleri ile günlük normal aydınlatma ve perde kontrollerini kaydederek, kiři evde olmadığında ve tatil zamanları kendini tekrarlayan,
- (iii). KNX çözümleri ile esneklik sağlayan bir sisteme sahiptir.



### 3. MATERYAL ve METOT

#### 3.1 Uygulamanın Tasarımı İçin İhtiyaçların Belirlenmesi

Bu tezde bir akıllı ev sistemi geliştirmek isteyen kişiler için, bir mobil uygulama sistemi geliştirilmiştir. Bu mobil uygulama temel olarak aydınlatma kontrollerini yönetebilecek, yine bu kontrollerin mantığı ile çalışan perde-panjur kontrolleri, priz yönetimi, sulama kontrolleri yapabilen bir sistem olarak tasarlanmıştır. Bu sistemin ana sayfasında her kontrol için ayrı bir menü oluşturulmuş ve bu menülerin her biri, kendine ait kontrollerin yer aldığı sayfalarda detaylandırılmaktadır. Bu anlamda mobil uygulamaya ait yönetim sayfası dışında toplamda 12 menü bulunmaktadır. Ayrıca menülerde var olan kontrollere uzun süreli ya da kısa süreli basma olmak üzere iki farklı komut gönderme özelliği mevcuttur. Bu menülerden çeşitli örnekler aşağıda verilmiştir:

- (i). Oda menüsüne baktığımızda, yönetim bölümünden girilen salon, oturma odası, mutfak gibi oda çeşitleri menüler halinde sunulmuş ve bunlardan herhangi birine tıklandığında ise tıklamanın yapıldığı oda için akıllı ev sisteminde çalışacak kontrollerin görünmesi sağlanmıştır.
- (ii). Lambalar menüsü aydınlatma sistemlerinin yer aldığı menü olup, her bir aydınlatma kontrolü bu menünün altında yer almaktadır. Bu menüde var olan kontrollere kısa süreli basıldığında kontrol açıksa kapalı, kapalıysa açık komutu gönderilmektedir. Dahası akıllı evdeki mevcut aydınlatmaların parlaklık şiddeti ayarlanabiliyorsa, kontrolün olduğu menüye uzun süre basılarak kontrollerin yer aldığı detay sayfasında aydınlatmanın parlaklık şiddetinin sıfırdan yüzde yüze yükseltilebileceği bir bar bulunmaktadır.
- (iii). Ana sayfadan perde-panjur menüsüne tıklandığında var olan bütün perde-panjur kontrolleri detay sayfasında görüntülenmekte ve bu sayfada perdenin aç, kapat ya da durdur butonları ile kontrol edilmesi sağlanmaktadır.
- (iv). Güvenlik kameraları menüsünde bütün kameraların kontrolleri verilmekte, buradan kontrolün kendisine tıklandığında ise kamera görüntüsü gösterilmektedir.
- (v). Sulama menüsünde, bahçedeki sulama sistemlerine ulaşılabilir.

Mobil uygulamanın diđer bir özelliđi senaryolar menüsünde tanımlanmış olan çeşitli durumlardır. Bu durumlar, evdeki belli teknolojilerin belirlenen zaman diliminde çalışma ya da kapanma özelliklerinin senaryo olarak tanımlanması şeklinde düşünülebilir. Senaryolar menüsü bu anlamda günlük rutin alışkanlıkların bir otomasyon sürecine bağlanmasının sağlanmasıdır. Dolayısıyla uygulamada yer alan senaryolar menüsü ile yeni senaryolar tanımlanabilmektedir.

Mobil uygulamada bir de medya menüsü bulunmaktadır. Bu medya menüsünün mantığı televizyon, klima, uydu gibi evde var olan tüm kumandaların tek bir çatı altında toplanmasıdır. Bu farklı kumandanların her birinde yer alan aç, kapat, ses artırıp azaltma, kanal deđiştirme ve numara blođu gibi standart butonlar tespit edilmiş ve medya bölümünde bunların kontrolüne imkan tanınmıştır.

Mobil uygulamada senaryo ve medya menüsünün dışında sesli komut özelliđi de bulunmaktadır. Sesli komut özelliđi ile örneđin “salon lamba aç” sözlü komutu verildiđinde, uygulama programlama arayüzlerinden biri olan Google API aracılıđıyla bu komutu metin haline dönüştürür. Devamında metin haline gelen sözlü komut “salon lamba” ve “aç” olmak üzere iki ayrı koda bölünerek, kod ID’si tanımlanır. Burada “salon lamba” mobil uygulamada yer alan kontrol adı, “aç” ise komutu oluşturmaktadır. Sonuçta sesli komutta yer alan kontrol adının ID’si ve aç ya da kapat komut ID’si birleştirilerek, sistem sesle kontrol edilebilir hale getirilmiştir.

Bahsedilen bu menü ve özelliklerin bir akıllı ev sistemini yöneten mobil uygulamada gerçekleştirilmesi için bir takım yazılım gereksinimlerinin belirlenmesi ve network programlama araçlarından hangilerinin kullanılacağına karar verilmesi gerekmektedir. Bu tezde geliştirilen mobil uygulamada kullanılacak yazılım ve network programlama araçları detayları ile devam eden başlıklarda verilmektedir.

### **3.1.1 Yazılım Gereksinimlerinin Belirlenmesi**

Bir mobil uygulama tasarlanmasının amacı mekan sınırı olmaksızın bulunulan herhangi bir konumdan kullanım sağlanmasıdır. Bu bağlamda var olan mobil cihazların, bir akıllı ev sistem tasarımı için geliştirilecek mobil uygulamayı karşılayabilmesi için gerekli

altyapıya sahip olup olmadığına yönelik avantaj ve dezavantajları incelenmiştir. Günümüzde öne çıkan Android ve iOS olmak üzere iki mobil işletim sistemi mevcuttur. Bu tezde geliştirilen mobil uygulamanın ise Android tabanlı tasarlanmasına karar verilmiştir. Bunun nedenleri olarak Android'in geliştiricilere yönelik daha fazla imkan tanınması ve Android destekli cihazların mali açıdan uygun olması gösterilebilir.

Android işletim sistemi ile çalışan bir mobil uygulama geliştirmek için Android Studio programının kurulması gerekmektedir. Android Studio programında yapmış olduğumuz uygulamanın ön izlemesini, Android Studio programının içinde yer alan emülatörden izleyebileceğimiz gibi, üçüncü parti bir yazılım ile ya da doğrudan Android işletim sistemine sahip bir telefon, tablet ile test edebilmektedir. Bu tezde mobil uygulamaların testini gerçekleştirmek üzere Android çözünürlük oranlarından xxhdpi özelliğindeki bir mobil cihaz kullanılmıştır.

Sistemde lamba, perde-panjur ve sulama gibi bütün kontrollerin tutulacağı bir veri tabanı gerekmektedir. Bunun içinse Android içinde mevcut olan SQLite veri tabanı hız ve sadelik açısından rahat bir kullanıma sahip olması nedeniyle tercih edilmiştir. Bu yazılımların ve ortamların kurulması aşamalarından sonra sistemin hazırlanmasında geliştiricilerin ya da kullanıcıların ileri düzeyde bir programlama becerisinin yanı sıra, Android işletim sistemi altyapısında kullanılan Java programlama dilinin söz dizimine (syntax) hakim olması gerekmektedir. Dahası geliştiriciler için tasarlanan bu akıllı ev mobil uygulamasında diğer elektronik cihazlar ile haberleşme gerektiği için ağ (network) programlama bilgisine ve bu ağ protokollerinden birinin tercihinin ihtiyaç duyulacaktır.

### **3.1.2 Ağ Programlama Protokolleri**

Elektronik cihazların birbiri ile iletişime geçebilmesi için bir ağ yapısı kullanılması gerekmektedir. Bu cihazların haberleşebilmesi için ortak protokollerin kullanılmasına ihtiyaç duyulmaktadır. Bu protokollerden biri TCP (Transmission Control Protocol), diğeri ise UDP'dir (User Datagram Protocol).

Akıllı ev sisteminde karar vermemiz gereken protokollerden biri olan UDP paketin teslim garantisini vermemesine rağmen, hız yönüyle TCP'den daha başarılıdır (Menteşoğlu *et al.* 2015). Burada geliştirilen mobil uygulama için de önemli olan kontrol butonuna tıkladığında komutun anında işlenmesini beklenir. Dolayısıyla butona tıkladıktan sonra komutun derlenip, var olan ağ yollarıyla elektronik cihaza gitmesi ve beraberinde kontrol butonunun komutu göndermesi arasında geçen sürenin uzaması bir dezavantajdır. Menteshoglu ve digerleri (2015) UDP protokolünün hız özelliğine rağmen, paket teslim garantisini vermemesi ve komutun ulaşip ulaşamadığı bilgisine erişilememesi gibi nedenlerle iletişim için bir haberleşme protokolü olan TCP protokolünü kullanmıştır. Aynı nedenlerden dolayı bu tezde geliştirilen mobil uygulama için de TCP protokolünün kullanılmasına karar verilmiştir.

### 3.2 Tasarım ve Geliştirme

Mevcut bir akıllı ev tasarımı için geliştirilmiş mobil uygulamada ilk olarak iş akışı çıkarılmış ve sistemli bir şekilde eyleme geçirilmesi amaçlanmıştır. Bu tasarım ve geliştirme sürecinde;

- (i). Uygulamada gerekli olacak kontrollerin listesi ve neleri kontrol edeceği: Uygulamada olması gereken kontroller aydınlatma, perde-panjur, priz, kameralar, algılayıcılar, hareket dedektörleri, sulama, medya ve alarm gibi.
- (ii). SQL veri tabanı şemasının tasarlanması
- (iii). Menü ve sayfaların belirlenmesi: Toplamda 12 menü ve 58 sayfa oluşturulmuştur.
- (iv). Uygulamada veri iletimi için kodlarının yazılması ve test edilmesi: Uygulamanın arayüzünden daha çok, sistem kodlarının çalışması, haberleşme protokollerinin sağlıklı bir şekilde veri iletimi yapıp yapmadığı ve temel kontrollerin uygulamada çalışıp çalışmadığı gözlenmiştir.
- (v). Uygulama kodlarının düzenlenmesi: Uygulamanın kodlarının yazılması aşamasında programlama mantığına uygun daha okunaklı bir kod yazılması ve kodların optimizasyonu üzerinde çalışılmıştır.
- (vi). Uygulama görsellerinin ve arayüz tasarımının gerçekleştirilmesi: Geliştiricilerin bu mobil uygulamayı rahat bir şekilde kullanabilmesi için

uygulamanın arayüz tasarımı ve tasarımdaki görseller düzenlenmiştir. Bu şekilde sistem programlaması tamamlandıktan sonra uygulamanın işlevsel olarak kullanımını artırmak için arayüz tasarımı üzerinde ayrıca çalışılmış, sade bir arayüz ortaya çıkarılmıştır.

### 3.2.1 Kurulum ve Sisteme Giriş

Tasarım ve geliştirme sürecinde mobil uygulamanın ilk aşamasında sistem kurulumu gerçekleştirilmelidir. Bunun için kullanıcı girişi menüsü ya da sayfanın alt bölümünde yer alan yönetim kısmına ulaşabileceğimiz Ayarlar butonu kullanılabilir. Ancak sistem kurulumu yapmadan önce menüleri incelemek isteyen kullanıcılar için demo kullanıcı girişi tanımlanmıştır. Ayarlar menüsünde yönetim sayfasına eriştiğinizde, Kurulumu Tamamla ve sisteme ait tüm ayarların yapılabileceği menülerle karşılaşılacaktır. Resim 3.1’de Kullanıcı Girişleri ve Yönetim sayfası ekran görüntüsü gösterilmektedir.



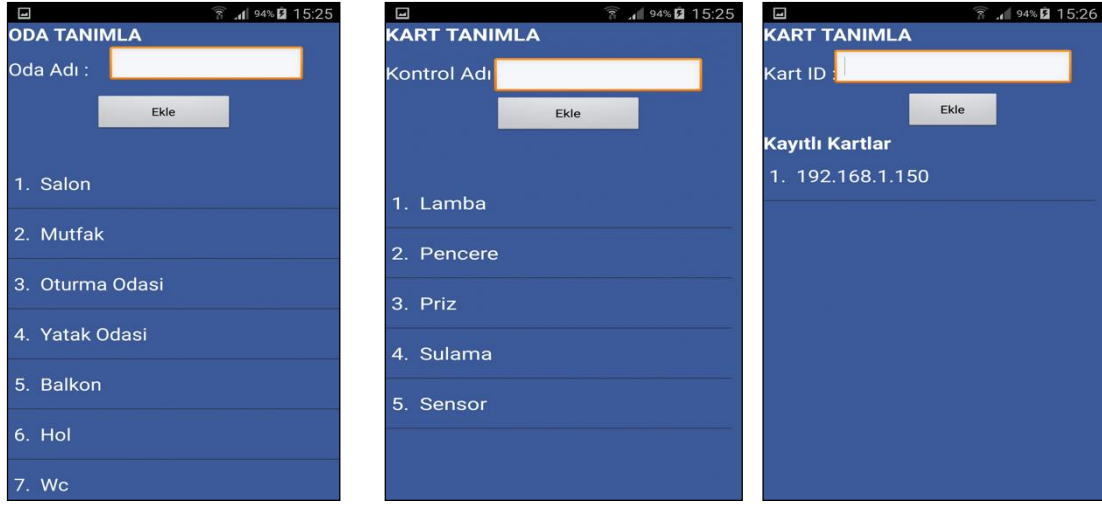
(a)



(b)

**Resim 3.1** (a) Kullanıcı Girişi ve (b) Yönetim Sayfası

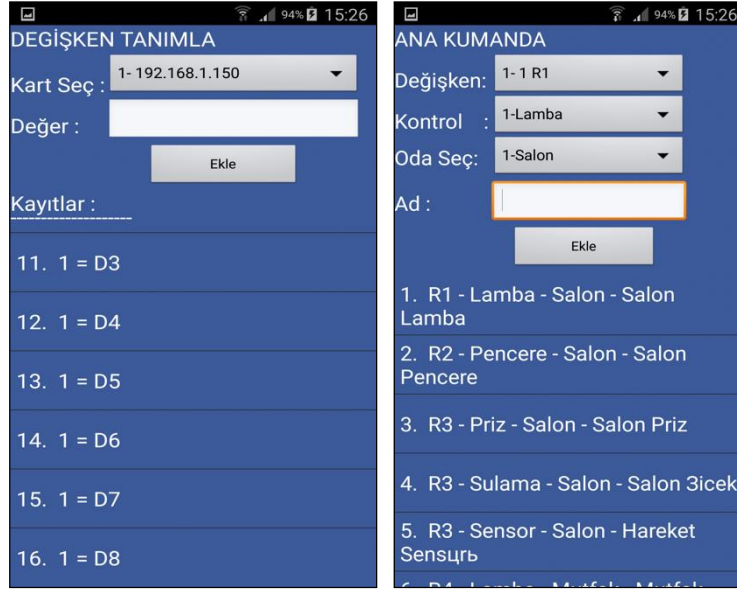
Resim 3.1.b’deki Yönetim Sayfası’nda sırasıyla şu tanımlamaların yapılması gerekmektedir. Akıllı evinize ait odalarınızı tanımlayacağınız Oda Tanımla menüsü aracılığıyla mobil uygulama sistemine mevcut odalar girilmelidir (Resim 3.2.a).



(a)

(b)

(c)



(d)

(e)

**Resim 3.2** Kurulum Sayfalarında Örnek Ekran Görüntüleri

Eklediğimiz verinin üzerinde daha uzun basarak veri güncellenebilir. Kurulumun ikinci sayfasında (Resim 3.2.b) Kontrol Tanımla menüsü sistemimizde var olan lamba, pencere, priz gibi evde kontrol edilmesini istediğimiz kontroller eklenmelidir. Üçüncü olarak Kart Tanımla (Resim 3.2.c) sistemde tanımlanmış olan kontrolün yolunu belirtmek için bu kontrolün iletişime geçeceği elektronik kartımızın IP adresi tanımlanmalıdır. Değişken Tanımla menüsü (Resim 3.2.d) ile değişken değerleri girilmelidir. Bu sayfa birden fazla elektronik kart olması durumu düşünüldüğünde elektronik karta göre değişkeni girebilmek için modüler olarak tasarlanmıştır. Böylece

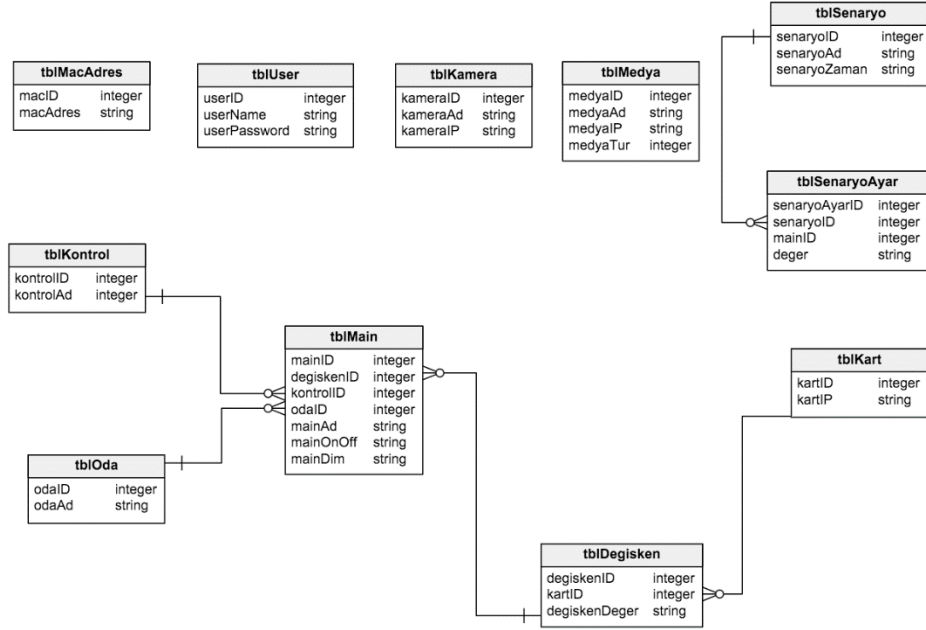
sisteme daha önce tanımlanan elektronik kart IP'leri ve onlara ait değişkenlerin değerleri girilmektedir. Değişken değerleri de sistemde tanımlandıktan sonra, Ana Kontrol menüsü (Resim 3.2.e) ile şimdiye kadar tanımlanan oda, kontrol, kart ve değişkenlerin sistemde bir dizi olarak sıralandığı ana tabloya girilir.

Kontrollerin tanımlanmasından sonra Yönetim sayfasına geri dönülerek, Kullanıcı Tanımla menüsü ile kullanıcı adı ve şifre, Kamera Tanımla menüsü ile evde yer alan kameralar ve Modem/Mac Tanımla menüsü ile sisteme ait olan ağ adresi tanımlanmalıdır. Bunların yanı sıra Medya menüsü ile evimizdeki kumandaların isimleri ve kodlaması tanımlanır. TCP Client ve Sistem Kontrol menüleri ise test amaçlı oluşturulmuş sayfalardır. TCP Client sayfasında bağlanmak istediğimiz Server'ın IP'sini ve portunu girdikten sonra manuel olarak TCP üzerinden komut göndermemizi sağlayan bir sayfa oluşturulmuştur. Sistem Kontrol sayfası ise bağlı olduğumuz elektronik uygulayıcı karttan okunan bütün verilerin anlık olarak görülebildiği bir sayfa olarak tasarlanmıştır.

Yapılan tüm bu kurulum aşamalarının dışında Yönetim sayfasından ayrıca db Dışarı Aktar menüsü ile kurulumla ait veri tabanı dosyası mobil cihaza yedeklenebilir. Bu şekilde herhangi bir silinmede kurulum aşamalarını tekrar tanımlamak yerine kurulum ayarlarını hızlandırmak için yedeklenmiş bu veri tabanı dosyasının sisteme entegre edilmesi için db İçeri Aktar menüsü kullanılabilir. Kurulumun tamamlanmasından sonra sistemin oluşturduğu veri tabanı dosyasının şematik görünümü Şekil 3.1'deki gibidir.

Tüm bu aşamalardan sonra artık Kurulumu Tamamla menüsü aracılığıyla kurulumu tamamlayabilirsiniz. Bu aşamaların akış diyagramı Şekil 3.2'deki gibidir.

Kurulumu tamamla dedikten sonra, kullanıcı adı ve şifrenizi sisteme girmenizi isteyen bir giriş ekranı (Resim 3.3.a) ile karşılaşır ve kurulumu tamamlamış olursunuz. Sisteme giriş yapıldığı takdirde tanımlanan tüm kontrolleri ana ekranda menüler halinde görebilirsiniz.



Şekil 3.1 Veri tabanı şeması

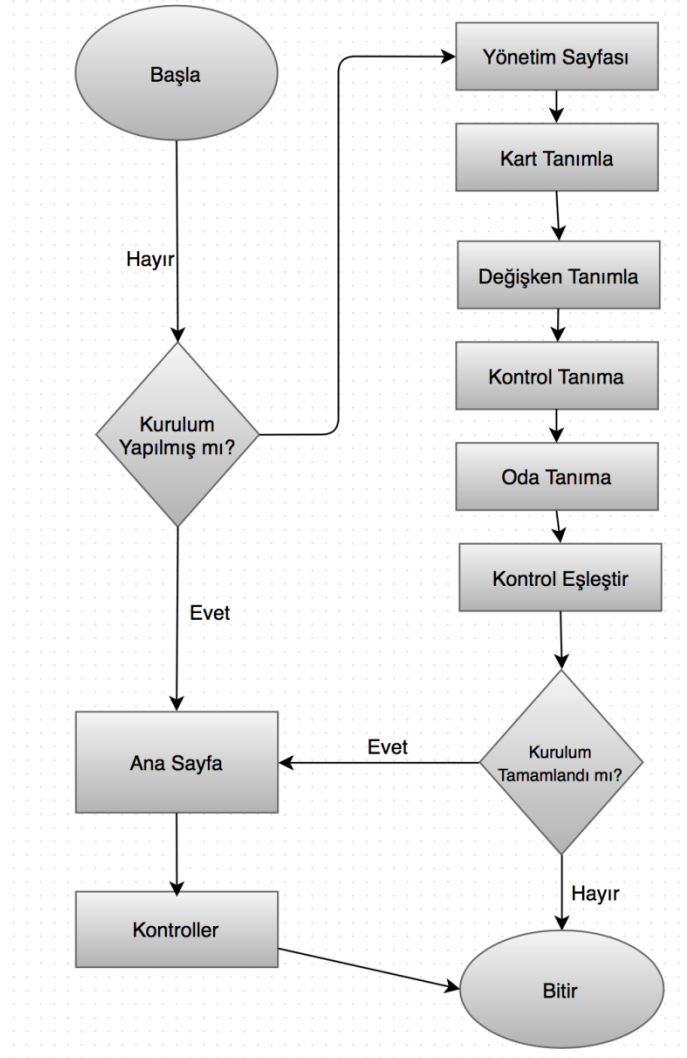
### 3.2.2 Aydınlatma Kontrolleri

Sisteme giriş yaptıktan sonra gelen ana ekranda Aydınlatma menüsü akıllı evde yer alan aydınlatmaların kontrolünü sağlamaktadır. Özellikle parlaklık şiddeti değiştirilebilen aydınlatmalarda bu hız, gözle görülebilecek şekilde fark edilebilmektedir. Bu nedenle aydınlatma kontrolleri mobil uygulamanın hız bakımından testinin gözlenmesi için kullanılan en pratik menüdür.

Aydınlatma menüsü içindeki tüm lamba kontrolleri, ana tablodaki adıyla ve image butonu ile birlikte sisteme tanımlanan öğeleri iki boyutlu şekilde düzenleyen bir görüntüleme tablosu kullanılarak Grid View içerisinde gösterilmiştir. Sonra Grid View içindeki image butonuna kısa süreli tıklama komutu eklenmiştir. Bu tıklama komutunda açık ya da kapalı olup olmadığına yönelik aydınlatma kontrolünün son değeri TCP protokolü aracılığıyla kontrol edilip, değişkene atanır. Bu değişkenin değeri 0 ise 100, 100 ise 0 değeri, Relaybas adını verdiğimiz fonksiyon, Kontrol ID'si, değişken değeri ve Kart ID'sini alarak, service.java dosyasındaki tcpSetSend üzerinden elektronik uygulayıcıya gönderilir (Şekil 3.3). Bu şekilde eğer aydınlatma açık ise, aydınlatma



kontrolü ekranında açık lamba simgesi, kapalı ise kapalı lamba simgesi Resim 3.3.a'da gösterilmektedir.

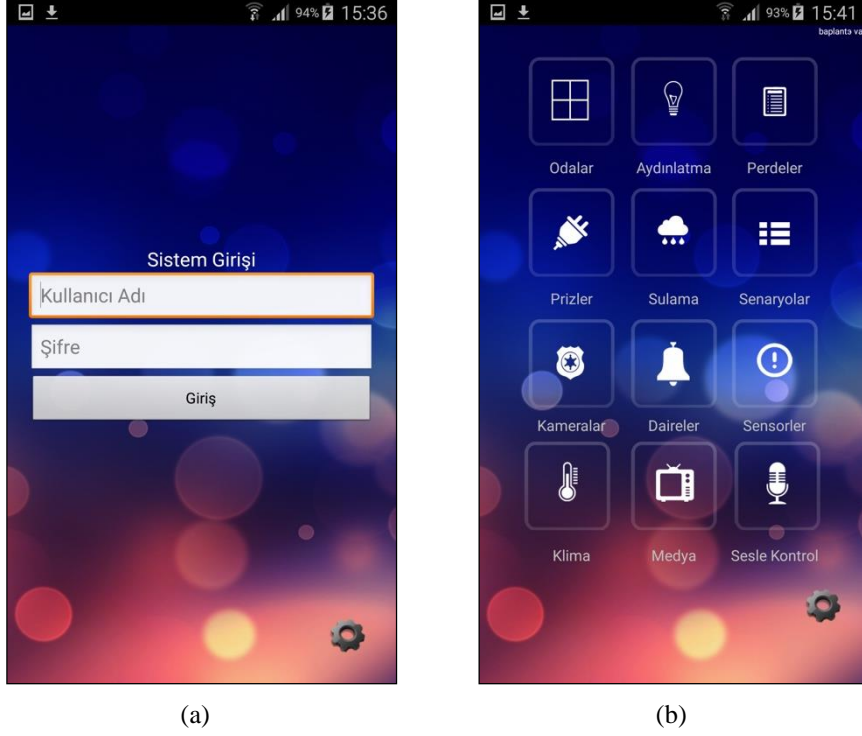


Şekil 3.2 Sistemin akış diyagramı

Yukarıdaki işlemler simgeye kısa tıklandığında çalışacak komutlar ve olayları gösterirken, Resim 3.3.b'de butona uzun süre tıklandığında gerçekleştirilecek kontrol\_lamba aktivitesi gösterilmektedir. Kontrol lamba aktivitesinde ise aydınlatma kontrolünün açma-kapama butonu Resim 3.4.a'da ve kontrol parlaklık şiddeti ayarlanabilen bir slider ile parlaklık şiddeti üzerinden düzenleme Resim 3.4.b'de görüldüğü şekilde yapılabilir.

### 3.2.3 Perde/Panjur Kontrolleri

Perde/panjur menüsü içinde tüm perde/panjur kontrollerinin gösterilmesi ve arkasından image butonlarına tıklandığında perdenin detay sayfasına erişilmektedir (Resim 3.5.a).



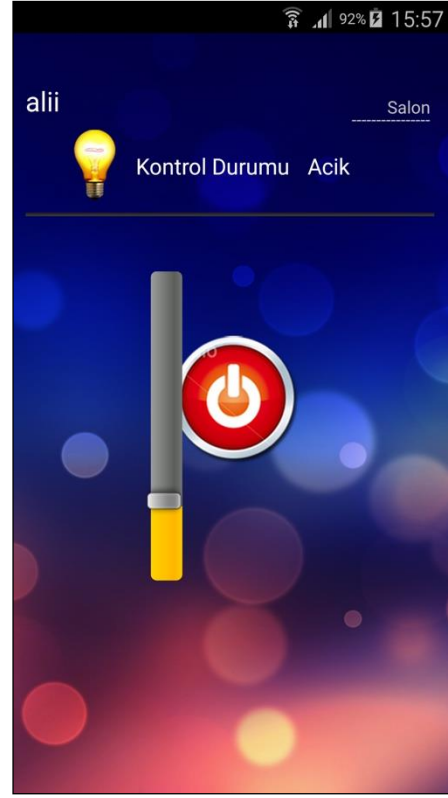
**Resim 3.3** (a) Sisteme Giriş ve (b) Ana Ekran

```
@SuppressWarnings("unused")
public static void tcpSetSend(final String mesaj, Integer wCard) {
    try {
        OutputStream ou;
        PrintWriter out;
        DataOutputStream os;
        if (wCard == 1) {
            ou = classvariables.soc.getOutputStream();
            out = new PrintWriter(new BufferedWriter(
                new OutputStreamWriter(ou)), true);
            os = new DataOutputStream(classvariables.soc.getOutputStream());
            os.writeBytes(mesaj);
        }
    } catch (SocketException e) {
    } catch (UnknownHostException e) {
    } catch (IOException e) {
    } catch (Exception e) {
    } finally {
    }
}
```

**Şekil 3.3** TCPSetSend komut satırları



(a)

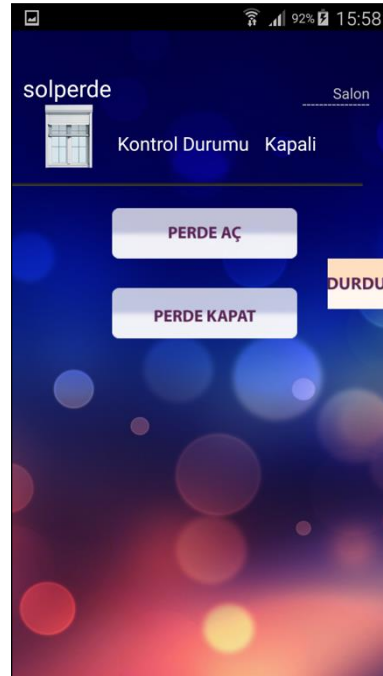


(b)

**Resim 3.4** (a) Aydınlatma Kontrolü Ekran Görüntüsü ve (b) Dimmerli lambalar için kontrol



(a)



(b)

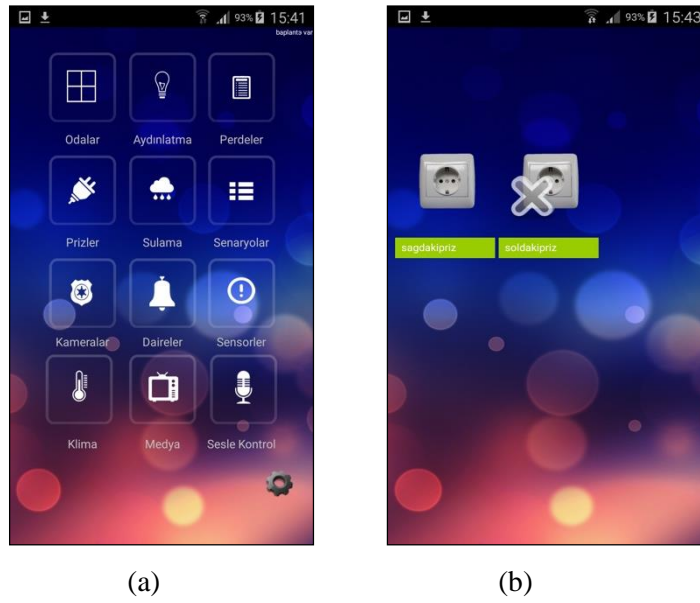
**Resim 3.5** (a) Perde Kontrolleri ve (b) Panjur Kontrolleri

Kontrolün detay sayfası kontrol\_perde aktivitesinden oluşmaktadır. Kontrol perde

aktivitesinde ise perde ya da panjur için aç-kapat ve durdur butonları bulunmaktadır (Resim 3.5.b). Perde kontrolünün detay sayfasının kodları Ek 6'da verilmiştir.

### 3.2.4 Priz Kontroller

Priz kontrolleri menüsüne gelindiğinde, çalışma mantığı aydınlatma kontrollerinin mantığı ile aynı temele dayandığı için kodlamada çok fazla değişiklik yapılmamıştır (Resim 3.6.a). Burada yalnızca Prizler menüsünde tıkladığında prizde enerji varsa kesme komutu, prizde enerji yoksa enerji verme komutu çalışmaktadır (Resim 3.6.b).



**Resim 3.6** (a) Ana menü ve (b) Priz kontrolü

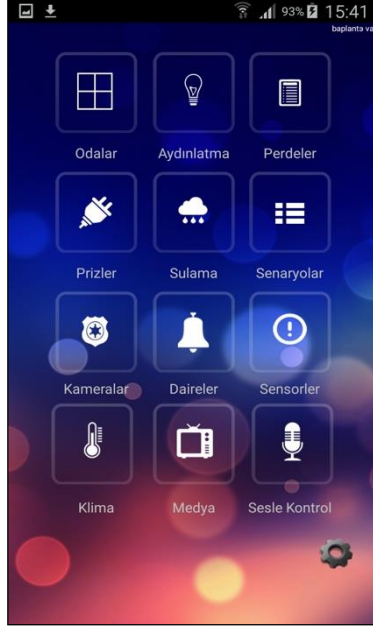
### 3.2.5 Klima

Klima kontrolü çeşitli firmaların farklı elektronik devreleri olmasından dolayıyla, kumanda kontrolü mantığıyla kodlanmıştır. Klimalarda, ortak bulunan sıcaklık arttır-azalt, aç-kapat, fan hızı gibi butonlar tasarıma yerleştirilerek kodlanması sağlanmıştır. Ayrıca ortamın ısısı algılayıcılar yardımı ile ölçülerek klima istenilen değere ayarlanabilir ya da klimanın derecesi manuel olarak sistem üzerinden artırılıp azaltılabilir.

### 3.2.6 Sulama Sistemleri Kontrolleri

Ana ekrandaki Sulama menüsüne tıkladığında (Resim 3.7.a), sistemde tanımlı bulunan

elektronik bir vana varsa sulama sistemi açma-kapama aktivitesi gerçekleştirilebilir (Resim 3.7.b).



(a)



(b)

**Resim 3.7** (a) Ana Menü (b) Sulama kontrolü

### 3.2.7 Senaryolar

Ana ekranda yer alan senaryolar menüsü, akıllı evlerin işlevselliğini ön plana çıkarmak için önemlidir. Akıllı evden istenecek senaryolar düşünüldüğünde, belli bir zamanda otomatik çalışan senaryoların yanı sıra, kullanıcının istediği anda değiştirebileceği senaryolar seçilebilmektedir (Resim 3.8.a). Sonuç olarak senaryolar menüsüne tıklandığında, önceden tanımlanmış senaryoları ya da yeni senaryo tanımlamaya izin veren ekran gösterilmektedir (Resim 3.8.b).

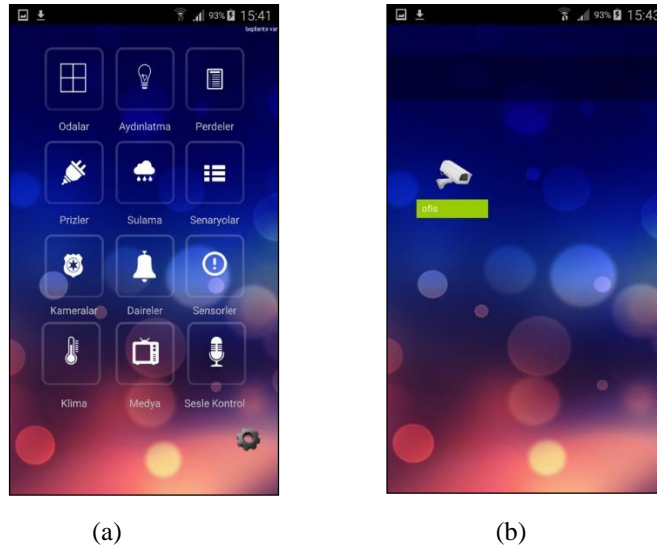
### 3.2.8 Kameralar

Günümüzde ev ya da apartman yaşantılarında güvenlik önemli bir unsur haline gelmiştir (Resim 3.9.a). Evler akıllanmasa bile güvenlik ihtiyacı için kamera sistemleri tercih edilmektedir. Bu nedenle bir akıllı evde en önemli öğelerden biri de kamera görüntülerinin kullanıcılara ulaştırılabilmesidir. Bu bağlamda geliştiriciler için hazırlanan bu mobil uygulamada kameralar menüsü ve kontrolün görüntüsünün

aktarıldığı Resim 3.9.b’de görüldüğü gibi tasarlanmıştır.



**Resim 3.8** (a) Ana Menü (b) Önceden tanımlanmış senaryolara ait ekran görüntüsü



**Resim 3.9** (a) Ana Menü (b) Kamera menüsüne ait ekran görüntüsü

Ana ekranda yer alan kameralar menüsü yalnızca sistemin kurulum aşamasındaki Yönetim sayfasında tanımlanan güvenlik kameralarının kaydettiği görüntülerin izlenmesi için kullanılan bir menüdür. Bu menü aracılığıyla kamera görüntülerine ait IP adresleri girilerek kayıtlar izlenebilmektedir. Bu anlamda tanımlanan kameraların IP

tabanlı olması önemlidir.

### **3.2.9 Medya**

Bir akıllı evde olması gereken özelliklerden biri olarak, Medya menüsü evde var olan tüm kumandaların tek bir kontrol merkezinden yönetilebilmesini sağlamaktır. Televizyon, uydu, klima, müzik seti ya da evde bulunan diğer elektronik aletlerin kumandasını tek bir akıllı telefonda birleştirmek için kullanılan Medya menüsüne tıklanıldığında, yönetim kısmından tanımlanan kumandalar görüntülenir. Bu kumandalardan kullanılmak istenen hangisi ise o kumanda seçildiğinde ise standart bir ekran görüntüsü ile karşılaşılır. Bu ekranda ise aç-kapa, ses arttır-azalt gibi her kumandada yer alan standart menüler bulunmaktadır. Bu şekilde örneğin klima kumandası seçildiğinde, arttır-azalt menüsüne tıklanırsa klima sıcaklık ayarı değiştirilmiş olacaktır.

### **3.2.10 Sensörler**

Akıllı evlerde, ev ile ilgili uyarıcılar elektronik uygulamaya duman dedektörü, su baskını ya da hareket gibi sensörler aracılığı ile iletilmektedir(Resim 3.10.a). Bu sensörler aracılığı ile sistem, herhangi başka bir kontrolü aktifleştirebilmektedir. Örneğin, duman dedektörü sensörü evde yangın olma ihtimali senaryosunu aktif ederek evin doğal gazını kapatabilir, su baskını sensörü ise evin su vanasını kapatabilmektedir (Resim 3.10.b).

### **3.2.11 Odalar ve Daireler**

Geliştirilmiş olan mobil uygulamada sistemde sadece bulunduğunuz oda içindeki kontrollere ulaşmak isterseniz Resim 3.11.a'deki ana menüden Odalar menüsüne tıklayarak bütün odaların listesine oradan da o odadaki bütün kontrollere ulaşabilirsiniz (Resim 3.11.b).

## **3.3 Elektronik Uygulayıcı**

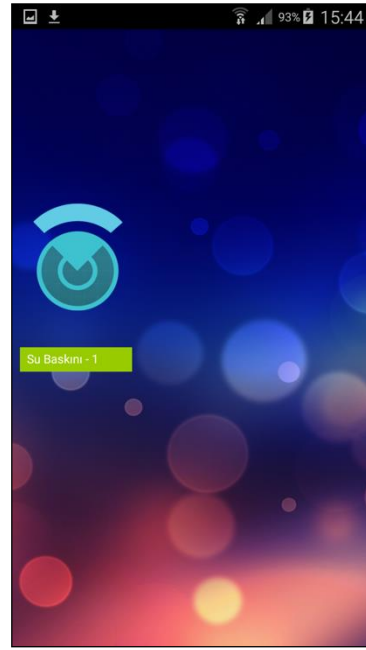
Akıllı ev geliştiricileri için hazırlanmış mobil uygulamanın test aşamasında bir elektronik uygulayıcı kart kullanılmıştır. Bu kartın donanım ve yazılım özellikleri ayrı



başlıklar altında aşağıda detaylarıyla aktarılmıştır.



(a)

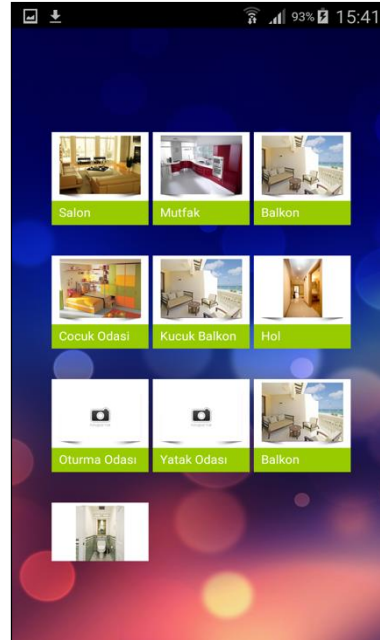


(b)

**Resim 3.10** (a) Ana Menü (b) Aktif olan algılayıcılara ait ekran görüntüsü



(a)



(b)

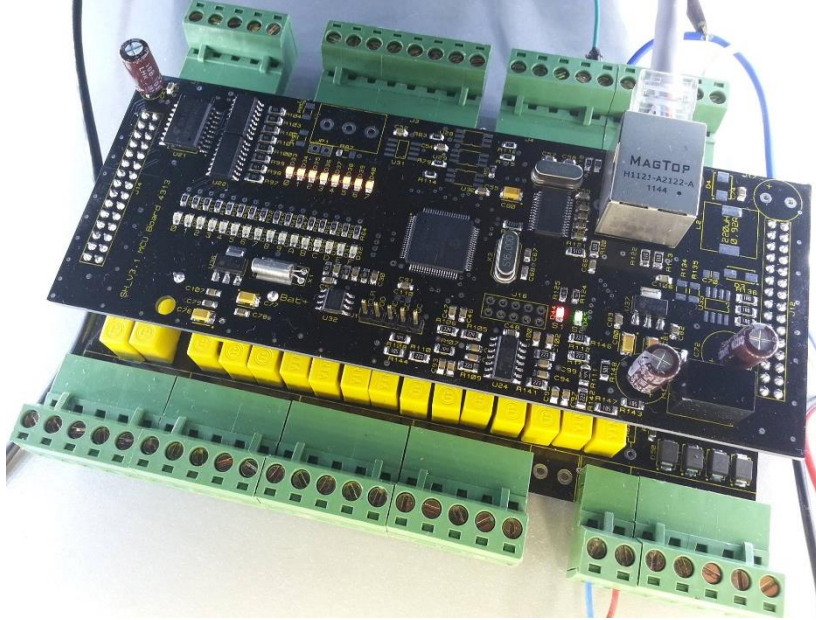
**Resim 3.11** (a) Ana Menü (b) Odalara ait ekran görüntüsü



### 3.3.1 Elektronik Uygulayıcı Kart Donanım Özellikleri

Mobil uygulamanın test aşamasında kullanılan elektronik uygulayıcı kart ethernet kontroller ENC28-J60 kullanılarak geliştirilmiştir. Resim 3.12’de görülen donanımın son halinde 8 bitlik 128 Kb ROM, 3.8 Kb RAM’e sahip 80 pinli bir MCU (Micro Controller Unit) kullanılmıştır. Mikro denetleyici yanında diğer çevresel elemanlar, algılayıcılar, medya cihazı gibi kontrollerle kendi üzerindeki RS485 portundan haberleşmektedir.

Elektronik uygulayıcı birim üzerindeki PCB (Baskı Devre Kartı) aracılığıyla 220 voltluk alternatif akım içeren 16 tane ve 4 tane 0-10 volt 12 bit analog girişe sahipken; 8 tane kuru kontak çıkış ve 8 tane 0-10 volt analog çıkışa sahiptir. MCU dışında 1 tane RS232-RS485 çeviri entegre, 1 tane 4 kanallı (0-10 volt girişi, 0-3 volta dönüştürmek için opamp kullanılmıştır. Yazılım güncellemek ve infrared sinyal bilgilerini kaydetmek için 1 tane 512 kb EEPROM kullanılmıştır. Mevcut donanım özelliklerindeki elektronik uygulayıcı kart 16 mips hızında işlem yapmaktadır.



**Resim 3.12** Elektronik Uygulayıcı Kartın Görüntüsü

### 3.3.2 Elektronik Uygulayıcı Kartı Yazılım Özellikleri

Elektronik uygulayıcı kartının yazılımı, sunucudan gelen komutlara göre çıkışları

düzenler ve girişlerden gelen bilgiyi aktarma işlemini sağlar. Sunucu, elektronik uygulayıcı kart ve algılayıcılar arasında veri giriş çıkışı kartın sunucu ile bağlantısının kesilmesi durumunda, kart ve algılayıcılar arasındaki veri iletişimi tek başına (stand alone) evdeki bütün bağlantılar için aktif olarak çalışmaya devam etmektedir. Aynı zamanda alarm durumunun oluşup oluşmadığına elektronik uygulayıcı kartının yazılımı karar vermektedir. Sunucu ile bağlantı kesilmesine rağmen, alarm durumunda elektronik kart giriş çıkış birimlerini bu yazılım aracılığıyla düzenlemektedir. Elektronik uygulayıcı yazılımı bir evdeki tüm sistemin girişlerini 50 ms’de taramaktadır. Bu taramalar sırasında 100 ms’de bir alarm durumlarını oluşturulacak algılayıcıların tanımlandığı girişlerin tetiklenip tetiklenmediği kontrol edilmektedir. Elektronik uygulayıcı kart ve sunucu arasındaki veri giriş çıkışı, yazılımda tanımlanan 8 soket tarafından sağlanmaktadır. Eğer bu soket bağlantılarından birinde 20 s içinde hiçbir veri giriş-çıkışı gerçekleşmez ise, soketle olan bağlantı yazılım tarafından kapatılır. Yazılımın bu veri giriş çıkışında rol alan soketleri kontrolü, karta bağlanan cihazların haberleşme sisteminin çalışma yoğunluğunu azaltmak açısından önemlidir.

### **3. 4 Akıllı Ev Uygulaması Geliştiricilerinin Ortamı Özelleştirmesi**

Geliştiriciler için tasarlanan bu sistemde, geliştiriciler kendi elektronik uygulayıcı sistemlerinde bu sistemin arayüz ve kodlama sistemini alarak kendi oluşturmuş oldukları kodlama sistemi ile elektronik uygulayıcı sistem ile haberleşebilir ve bu sistemi test amaçlı kullanabilir. Ortamı özelleştirebilmesi için ekte verilen kodlarda geliştiricinin bireyselleştirilmiş kod bloğu (BKB) bölümleri tekrardan tasarlanmalıdır. Bu sistemde geliştirilmiş olan arayüzde 10 haneli sistem kullanmış olup sistem başarı ile çalışmıştır.

## 4. BULGULAR

Sistemin test sürecinde gözlemlenen sorunlardan biri UDP ve TCP haberleşme protokollerinden hangisinin seçilmesi gerektiğiydi. İlk olarak sistemi UDP kullanarak tasarlanmasına rağmen, güvenlik ve doğrulama eksiklikleri nedeniyle UDP'den vazgeçilmiştir. Akıllı ev gibi insanların doğrudan yaşamı ile alakalı bir sistemde güvenlik ve doğrulama ihtiyacını gidermek için TCP protokolü çözüm olarak düşünülmüş ve sistem buna göre baştan düzenlenmiştir.

Haberleşme protokolünün sorunsuz bir şekilde ihtiyaçları karşılamaının ardından sistemin hızının kontrolü için kodların ilk hallerine göre optimize edilmesi ya da iyileştirilmesi ihtiyacı gözlemlenmiştir. Bu nedenle kodlar eskisine oranla daha okunaklı olacak şekilde optimize edilmiş, iş parçacıklarının (thread) planlanması gözden geçirilmiştir. Bu şekilde sistem gereksiz yük olabilecek kodlardan arındırılmıştır. Resim 4.1'de sistemin hızı test edilmiş ve uygulamadan gönderilen komut ve elektronik uygulayıcıdan dönen komutun tepki süresi gözlenmiştir.

Test sürecinde bir diğer gözlenen durum ise elektronik uygulayıcı ile mobil cihaz arasındaki soket bağlantılarının sınırlı olması, mobil uygulama kodlarının tekrar gözden geçirilmesini ve düzenlenmesini gerektirmiştir. Düzeltme işlemlerinden sonra soket bağlantıları minimuma indirilerek, sistemin haberleşme protokollerinin kilitletmesinin önüne geçilmiştir.

### 4.1 Mobil Uygulamanın Test Edildiği Akıllı Ev Sisteminin Mevcut Diğer Akıllı Evlerden Farkı

Mobil uygulamanın test edildiği akıllı ev sisteminde kullanılan donanım düşünüldüğünde, kablo altyapısı konvansiyel tip bağlantıya sahiptir (Resim 4.2). Bu anlamda bus altyapısını kullanan diğer akıllı evlerinkine göre daha fazla kablo ve daha fazla mali destek gerektirmektedir. Kovansiyel tipte kablolama sisteminin bu açıdan, bus alt yapısına sahip sisteme göre dezavantajlı olduğu düşünülebilir. Ancak bu tip kablolamaya sahip akıllı evler birden fazla daireye bağımsız olarak akıllı eve geçme ya da ayrılma konusunda seçenek sunması yönüyle avantajla dönüştürülebilir. Öyle ki akıllı

eve geçmeyi istemeyen daire sahipleri için sistemin elektrik panosundaki bağlantı şekli değiştirilerek günümüz klasik tesisatlarına geçilebilmektedir. Ancak bus tipindekiler birden fazla daireye sahip bir akıllı ev sisteminde sahiplere, akıllı ev sisteminden tek başına ayrılma hakkını tanımamaktadır.



**Resim 4.1** Sistemin test süreci

Mobil uygulamanın test edildiği akıllı ev sisteminin diğer bir özelliği ise sektörde standart olarak kullanılan KNX organizasyonundan bağımsız olarak çalışmasıdır. Bunun nedeni geliştiriciler için KNX'e kayıt olmanın bir maliyet gerektirmesidir. Bunun yanı sıra test aşamasında kullanılan akıllı ev sistemindeki haberleşme sistemi TCP/IP protokolleri üzerinden gerçekleştirilmektedir. Sonuç olarak geliştirilen mobil uygulama TCP/IP protokolü üzerinden haberleşen her akıllı ev sistemi ile çalışabilecek şekilde özelleştirilebilir.





**Resim 4.2** Sistemin kablolama süreci

## 5. SONUÇ VE ÖNERİLER

### 5.1 Sonuç

Bu tezde akıllı ev teknolojisi geliştiricileri için kontrol sistemini hazırlayacak bir mobil uygulama geliştirilmiştir. Geliştirilen bu mobil uygulama TCP üzerinden veri iletimini sağlayan, Android tabanlı bir uygulamadır. Mobil uygulama ile akıllı ev geliştiricileri bir evde yer alan aydınlatma, priz, sulama, medya özelliği ile evdeki elektronik cihazların ortak kumanda ile yönetimi ve kameraların kontrolünü gerçekleştirebilmektedir. Senaryolar özelliği ile hafta içi ayarladığınız saatte alarmın çalması, güvenlik sistemini devreden çıkarma, perdeleri açma, harekete duyarlı aydınlatmaları kapatma, kahve makinenizi çalıştırma, uzun süre evden ayrı kaldığınızda evin vanasını açık unuttuğunuzu düşünüp müdahale edebilmek gibi pek çok çeşitli senaryo tanımlanabilmekte ve kontrol edilebilmektedir. Bunun dışında sistemin sesle kontrolü de tek bir mobil uygulama üzerinden tanımlanabilmektedir.

Mobil uygulama tasarımının testinden sonra yapılan bu çalışmada sistemin başarı ile çalıştığı, geliştiriciler için kullanışlı bir sistem oluşturulduğu söylenebilir. Özellikle uygulamada yer alan kodların daha okunaklı bir şekilde optimize edilerek, hız bakımından iyi bir performans alınması sağlanmıştır. Kodlamadaki düzenlemelerin yanı sıra sistem arayüzünün kullanımının sadeliği ve basitliği üzerinde de ayrıca çalışılmış, geliştiricilerin ayrıca arayüz ve uygulama tasarlama ihtiyaçlarına yanıt verilmiştir.

Mobil uygulama bu tez kapsamında şimdiye kadar yalnızca laboratuvar ortamında test edilmiş ve karşılaşılan haberleşme protokolünün tercihi, kodların optimize edilmesi ve soket bağlantılarının minimum düzeye çekilmesi gibi sorunlara çözüm üretilerek, aslında bir prototip olan uygulamanın nihai haline erişilmiştir. Ancak sistemin laboratuvar ortamında gerçekleşen uygulamalarında çıkan sorunların, geniş çaplı kullanımlarında tekrar etmeyeceği ya da farklı sorunlarla karşılaşmayacağını söylemek yanlış olacaktır. Bu nedenle akıllı ev geliştiricilerine birtakım önerilerde bulunulmuştur.

## 5.2 Öneriler

Bu sistemi daha ileriye taşıyacak olan geliştiriciler için önerilenlerin en başında sistemin bütün iş ve işleyişi, android tabanlı bir işletim sistemli bir cihaz üzerinde gerçekleştirilmesinden işletim sistemine sahip bir sunucu kullanılarak android sistemi üzerindeki yük azaltılabilir. Günümüzde Android tabanlı mobil işletim sistemine sahip cihazların donanımsal olarak güçlü olmamaları uygulamada kilitlenmeler yaşatabilmektedir. Ayrıca bu sistemlerde elektronik uygulayıcıya yapılacak olan çok fazla durum bilgisi sorgusu mobil uygulamanın yüklenmiş olduğu cihazın gücünü aşırı tüketeceğinden bu cihazlarda enerji tüketimleri çok daha hızlı olabilmektedir. Bu ve benzeri sorunlarda geliştiriciler elektronik uygulayıcı ve mobil uygulama arasına sunucu görevi görebilecek bir bilgisayar ve o bilgisayarda çalışan hem istemci hem sunucu olabilecek bir program yazmaları sistemin daha kararlı çalışmasını sağlayabilir. Dahası elektronik uygulayıcı kısmında güçlü bir ana kontrol ve diğerlerinde daha az iş yükü olan gömülü sistem kontrolleri olabilir. Ana kontrol üzerinden ethernet kontrolü ile haberleşme protokolleri kullanılabilir. Diğer modüller ile ise RS485 üzerinden haberleşebilir.

Akıllı ev sistemi için geliştirilen mobil uygulama, kişilerin yaşantılarını doğrudan etkilediği için güvenlik unsuru da çok önemli yer tutmaktadır. Gerçekleştirilmiş olan bu sistemde güvenlik unsuru üzerinde çalışılmamıştır ve bu nedenle sistemin güvenli hale getirilmesi için kapsamlı bir çalışma yapılması gerekmektedir. Sistemin daha akıllı hale getirilmesi için doğal kaynakları daha az tüketen akıllı enerji yönetimi üzerinde daha fazla çalışma yapılabilir. Ayrıca öğrenen sistemler akıllı ev sistemlerinde kullanılabilir.

Gelecek çalışmalar içinse geliştirilen mobil uygulamanın test edildiği akıllı ev sistemi dışında, diğer akıllı ev sistemlerinde de test edilerek sonuçların izlenmesi sağlanabilir. Bu konuda bu tezde geliştirilmiş uygulama aynı zamanda diğer akıllı ev sistemlerinin özelliklerine uygun olarak özelleştirilebilir.

## 6. KAYNAKLAR

- Aslan, A. (2014). Akıllı Ev Kavramı ve Otomasyon Sistemleri. Yüksek Lisans Tezi, Haliç Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Balta-Ozkan, N., Boteler, B. and Amerighi, O. (2014). European smart home market development: Public views on technical and economic aspects across the United Kingdom, Germany and Italy. *Energy Research & Social Science*, **3**: 65-77.
- Balta-Ozkan, N., Davidson, R., Bicket, M. and Whitmarsh, L. (2013). The development of smart homes market in the UK. *Energy*, **60**: 361-372.
- Bayram, U. (2006). *Akıllı Ev Otomasyonu*. Yüksek Lisans Tezi, Çanakkale On Sekiz Mart Üniversitesi, Fen Bilimleri Enstitüsü, Çanakkale.
- Capitanelli, A., Papetti, A., Peruzzini, M. and Germani, M. (2014). A Smart Home Information Management Model for Device Interoperability Simulation. *Procedia CIRP*, **21**: 64-69.
- Çavuş, M. (2012). Android. Seçkin Yayıncılık, 1. Baskı. İstanbul.
- Fidan, U. ve Karasekreter, N. (2011). Gsm/Sms tabanlı sulama otomasyonu kontrol biriminin geliştirilmesi ve uygulanması. *E-journal of New World Sciences Academy*, **6**: 71-77.
- Goadrich, M. H. and Rogers, M. P. (2011). Smart smartphone development: iOS versus Android. Proceedings of the 42nd ACM technical symposium on Computer science education, 9-12 Mart, 607-612.
- Gügül, N, G. (2008). Akıllı Ev Sistemleri ve Uygulaması. Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- Gül, F. (2010). Akıllı Ev Sistemleri. Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- İnal, K. and Akcayol, M. A. (2009). GSM Tabanlı Akıllı Ev Uygulaması. *International Journal of Informatics Technologies*, **2**: 39-45.
- Kardaş, Y. Y. (2014). Gsm Kontrollü Akıllı Ev Uygulaması. Yüksek Lisans Tezi, Dicle Üniversitesi, Fen Bilimleri Enstitüsü, Diyarbakır.



- Kurbetçi, Z.E., Şen, N. ve Başkan, B. (2003). Akıllı Ev Teknolojisi, Elektrik-Elektronik-Bilgisayar Mühendisliği 10. Ulusal Kongresi, İstanbul, 18-21 Ekim, 287-293.
- Liu, Z. Y. (2014). Hardware Design of Smart Home System based on zigBee Wireless Sensor Network. *AASRI Procedia*, **8**: 75-81.
- Menteşoğlu, M., Kavak, A., Yakut, M., Tangel, A., Şahin, S. and Özcan, H. (2015). Design and implementation of a communication protocol for mobile device controlled smart home management system. *Signal Processing and Communications Applications Conference (SIU), 2015 23th*. IEEE, 16-19 Mayıs, 1370-1373.
- Narman, A. E. (2010). Android Studio ile Programlama (10). KodLab Yayın Dağıtım Yazılım LTD, İstanbul.
- Özçekiç, E. (2005). Akıllı Ev Sistemleri. Yüksek Lisans Tezi, Beykent Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Özer, H. İ. (2005). Akıllı Ev Otomasyon Sistemi Tasarımı. Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya.
- Stefanov, D. H., Bien, Z. and Chul Bang, W. (2004). The smart house for older persons and persons with physical disabilities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **12**: 228-250.
- Şahinoğlu, G. (2006). Akıllı Evlerde Otomasyon. Yüksek Lisans Tezi, Marmara Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Taç, M. (2011). Android Programlama. Seçkin Yayıncılık, Ankara.

## 6.1 İnternet Kaynakları

- 1- <http://www.berker.com>, 10 Ocak 2016.
- 2- <http://www.control4.com.tr/brosur.html>, 10 Ocak 2016.
- 3- <http://www.fibaro.com.tr/tr/sistem/neler-yapilir.html>, 10 Ocak 2016
- 4- <http://www.schneider-electric.com.tr/tr/home/house-electrical-products/knx-control-system/>, 10 Ocak 2016.

## ÖZGEÇMİŞ

Adı Soyadı : Ali ERBEY  
Doğum Yeri ve Tarihi : Uşak – 16.08.1984  
Yabancı Dili : İngilizce  
İletişim (Telefon/e-posta) : 0506 323 93 16 - ali.erbey@usak.edu.tr

### Eğitim Durumu (Kurum ve Yıl)

Lise : Uşak Orhan Dengiz Anadolu Lisesi 2004  
Lisans : Anadolu Üniversitesi, Bilgisayar ve Öğretim  
Teknolojileri Eğitimi, 2007

Çalıştığı Kurum/Kurumlar ve Yıl: Milli Eğitim Bakanlığı 2008-2013

Uşak Üniversitesi 2013- Devam ediyor.

## EKLER

### Ek-1. veritabani.java içeriği

```
package com.alierbey.akilliev;
import java.util.ArrayList;
import android.content.ContentValues;
import android.content.Context;
import java.util.List;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class veritabani extends SQLiteOpenHelper {

    private static final String VERITABANI = "akilliev.db";
    private static final int SURUM = 4;
    public veritabani(Context con) {
        super(con, VERITABANI, null, SURUM);
    }
    private static final String tblKontrol = "CREATE TABLE " + "tblKontrol"
        + " (" + "kontrolID" + " INTEGER PRIMARY KEY AUTOINCREMENT, "
        + "kontrolAd" + " TEXT);";
    private static final String tblMain = "CREATE TABLE " + "tblMain" + " ("
        + "mainID" + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
"degiskenID"
        + " INTEGER , " + "kontrolID" + " INTEGER , " + "odaID"
        + " INTEGER , " + "mainAd" + " TEXT, " + "mainOnOff" + " BOOL, "
        + "mainDim" + " INTEGER );";
    private static final String tblOda = "CREATE TABLE " + "tblOda" + " ("
        + "odaID" + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
"odaAd"
        + " TEXT);";
    private static final String tblKart = "CREATE TABLE " + "tblKart" + " ("
        + "kartID" + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
"kartIP"
        + " TEXT);";
    private static final String tblDegisken = "CREATE TABLE " + "tblDegisken"
        + " (" + "degiskenID" + " INTEGER PRIMARY KEY AUTOINCREMENT,
"
        + "kartID" + " INTEGER, " + "degiskenDeger" + " TEXT, " + "dim"
        + " BOOL );";
    private static final String tblGallery = "CREATE TABLE " + "tblGallery"
        + "(GalleryID INTEGER PRIMARY KEY AUTOINCREMENT, "
        + " Name TEXT(100), " + " Path TEXT(100));";
    private static final String tblKamera = "CREATE TABLE " + "tblKamera"
        + "(kameraID INTEGER PRIMARY KEY AUTOINCREMENT, "
        + " kameraAd TEXT(100), " + " kameraIP TEXT(100));";
    private static final String tblMedya = "CREATE TABLE " + "tblMedya"
        + "(medyaID INTEGER PRIMARY KEY AUTOINCREMENT, "
        + " medyaAd TEXT(100), " + " medyaIP TEXT(100), "
        + " medyaTur INTEGER );";
    private static final String tblMacAdres = "CREATE TABLE " + "tblMacAdres"
        + "(macID INTEGER PRIMARY KEY AUTOINCREMENT, " + "
macAdres TEXT);";
```

## Ek-1. (devam) veritabani.java içeriği

```
private static final String tblSenaryo = "CREATE TABLE " + "tblSenaryo"
    + "(senaryoID INTEGER PRIMARY KEY AUTOINCREMENT,"
    + " senaryoAd TEXT , senaryoZaman TEXT);";
private static final String tblSenaryoAyar = "CREATE TABLE "
    + "tblSenaryoAyar"
    + "(senaryoAyarID INTEGER PRIMARY KEY AUTOINCREMENT,"
    + " senaryoID INTEGER , mainID INTEGER, deger TEXT);";
private static final String tblUser = "CREATE TABLE " + "tblUser"
    + "(userID INTEGER PRIMARY KEY AUTOINCREMENT,"
    + " userName TEXT , userPassword INTEGER);";
// private static final String tblSenaryo = "CREATE TABLE " +
// "tblSenaryolar"
// + "(senaryoID INTEGER PRIMARY KEY AUTOINCREMENT,"
// + " senaryoAd TEXT(100)," + " senaryoOzellik TEXT(100));";
// private static final String tblKamera = "CREATE TABLE " + "tblKamera"
// + "(" + "kameraID" + " INTEGER PRIMARY KEY AUTOINCREMENT, "
// + "kameraAd" + " TEXT, "
// + "kameraIP" + " TEXT);";
@Override
public void onCreate(SQLiteDatabase db) {
    db.execSQL(tblKamera);
    db.execSQL(tblMedya);
    db.execSQL(tblKart);
    db.execSQL(tblDegisken);
    db.execSQL(tblMain);
    db.execSQL(tblKontrol);
    db.execSQL(tblOda);
    db.execSQL(tblGallery);
    db.execSQL(tblMacAdres);
    db.execSQL(tblSenaryo);
    db.execSQL(tblSenaryoAyar);
    db.execSQL(tblUser);
}
@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

    db.execSQL("DROP TABLE IF EXIST tblKamera");
    db.execSQL("DROP TABLE IF EXIST tblKart");
    db.execSQL("DROP TABLE IF EXIST tblDegisken");
    db.execSQL("DROP TABLE IF EXIST tblMain");
    db.execSQL("DROP TABLE IF EXIST tblKontrol");
    db.execSQL("DROP TABLE IF EXIST tblOda");
    db.execSQL("DROP TABLE IF EXIST tblGallery");
    db.execSQL("DROP TABLE IF EXIST tblMacAdres");
    db.execSQL("DROP TABLE IF EXIST tblSenaryo");
    db.execSQL("DROP TABLE IF EXIST tblSenaryoAyar");
    db.execSQL("DROP TABLE IF EXIST tblMedya");
    db.execSQL("DROP TABLE IF EXIST tblUser");
    onCreate(db);
}
```

## Ek-2. service.java

```
package com.alierbey.akilliev;
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.io.PrintWriter;
import java.net.Socket;
import java.net.SocketException;
import java.net.UnknownHostException;
import android.annotation.SuppressLint;
import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.app.Service;
import android.content.Intent;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.os.Handler;
import android.os.IBinder;
import android.os.StrictMode;
import android.util.Log;
import android.widget.Toast;
public class servisim extends Service {
    String readData = "";
    String readSplitData;
    String baglantiSonucu;
    String ip1 = "192.168.1.2150"; // Kart ip'leri manuel atanabilir
    Integer portNumber = 34400; // Port numarası burada giriliyor
    boolean run = false;
    protected Object mHandler;
    protected Object cHandler;
    @SuppressWarnings("unused")
    private Handler handler;
    @SuppressWarnings("NewApi")
    @Override
    public void onCreate() {
        super.onCreate();
        StrictMode.ThreadPolicy policy = new
StrictMode.ThreadPolicy.Builder().permitAll().build();
        StrictMode.setThreadPolicy(policy);
        classvariables.uyari = 0;
        classvariables.sensorActive=false;
        // for connection status
        this.kHandler = new Handler();
        ((Handler) this.kHandler).postDelayed(k_Runnable, 0);
        // for alarm control
        this.aHandler = new Handler();
        ((Handler) this.aHandler).postDelayed(a_Runnable, 1000);
    }
}
```

## Ek-2.(devam) service.java

```
public void connect() {
    try {
        classvariables.soc = new Socket(ip1, portNumber);
        classvariables.soc.isConnected();
        classvariables.baglanti = "baglanti var";
        run = true;
    } catch (Exception e) {
        Toast.makeText(getApplicationContext(), "Baglanti yok", 100).show();
        Log.d("service", "baglanti yok");
        run = false;
        classvariables.baglanti = "baglanti yok";
    }
}

private final Runnable m_Connect = new Runnable() {
    public void run()
    {
        if (run == false) {
            classvariables.soc.isClosed();
            // Toast.makeText(getApplicationContext(), "Tekrar baglanilacak", //
100).show();
            connect();
        }
        ((Handler) servisim.this.cHandler).postDelayed(m_Connect, 15000);
    }
};

private final Runnable k_Runnable = new Runnable() {
    public void run()
    {
        if (classvariables.soc1 != null) {
            classvariables.soc1.isClosed();
        }
        connect();
        kontrolSensor();
        ((Handler) servisim.this.kHandler).postDelayed(k_Runnable, 19000);
    }
};

@Override
public void onDestroy() {
    // Log.i("onDestroy", "onDestroy");
    tcpListener("kapan");
}

@SuppressWarnings({ "deprecation", "unused" })
private void notificationGoster() {
    NotificationManager notificationManager = (NotificationManager)
getSystemService(NOTIFICATION_SERVICE);
    Notification notification = new Notification(R.drawable.ic_launcher, "",
System.currentTimeMillis());
```

## Ek-2.(devam) service.java

```
Intent intent;
intent = new Intent(this, splash.class);
intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);// her tiklandığında yeni bir
aktivite baslamaması için
PendingIntent contentIntent = PendingIntent.getActivity(this, 0, intent, 0);
notification.setLatestEventInfo(this, "akilliev", "" + run + "", contentIntent);
notificationManager.notify(0, notification);
}
@SuppressWarnings("unused")
@SuppressLint("NewApi")
public void tcpListener(final String mesaj) {
    if (run == true) {
        try {
            OutputStream ou = classvariables.soc1.getOutputStream();
            PrintWriter out = new PrintWriter(new BufferedWriter(new
OutputStreamWriter(ou)), true);
            DataOutputStream os = new
DataOutputStream(classvariables.soc1.getOutputStream());
            os.writeBytes(mesaj);
            BufferedInputStream bfstr = new
BufferedInputStream(classvariables.soc1.getInputStream());
            DataInputStream din = new DataInputStream(bfstr);
            byte[] buffer = null;
            String str = null;
            for (int i = 0; i < 10; i++) {
                if (din.available() > 0) {
                    buffer = new byte[din.available()];
                    din.read(buffer);
                    str = new String(buffer);
                    break;
                } else
                    Thread.sleep(50, 0);
            }
            readData = str;
        } catch (SocketException e) {
            readData = e.toString();
            readData = "aga ulasilamiyor";
        } catch (UnknownHostException e) {
            readData = e.toString();
        } catch (IOException e) {
            < String message) {
                Integer don = null;
                try {
                    String mesaj = message;
                    OutputStream ou;
                    PrintWriter out;
                    BufferedInputStream bfstr = null;
                    DataOutputStream os;
                    if (cardID == 1) {
                        ou = classvariables.soc.getOutputStream();
                        out = new PrintWriter(new BufferedWriter(
                            new OutputStreamWriter(ou)), true);
                        os = new DataOutputStream(classvariables.soc.getOutputStream());
                        os.writeBytes(mesaj);
```

## Ek-2.(devam) service.java

```
        bfstr = new BufferedInputStream(
            classvariables.soc.getInputStream());
    }
    DataInputStream din = new DataInputStream(bfstr);
    byte[] buffer = null;
    String str = null;
    for (int i = 0; i < 10; i++) {
        if (din.available() > 0) {
            buffer = new byte[din.available()];
            din.read(buffer);
            str = new String(buffer);
            break;
        } else
            Thread.sleep(50, 0);
    }
    readData = str;
} catch (SocketException e) {
    readData = e.toString();
    readData = "Aa ulasilamiyor!!!";
} catch (UnknownHostException e) {
    readData = e.toString();
} catch (IOException e) {
    readData = e.toString();
    readData = "IO problemi var!!!";
} catch (Exception e) {
    readData = e.toString();
    readData = "Genel problem";
} finally {
    if (readData == null) {
        // Toast.makeText(getApplicationContext(),
        // "baglantı kesildi...",
        // Toast.LENGTH_SHORT).show();
        run = false;
        classvariables.baglanti = "baglanti yok";
        this.cHandler = new Handler();
        ((Handler) this.cHandler).postDelayed(m_Connect, 0);
    } else {
        don = Integer.parseInt(readData.substring(6, 7).toString());
    }
}
return don;
}
@SuppressWarnings("unused")
public static void tcpSetSend(final String mesaj, Integer wCard) {
    try {
        OutputStream ou;
        PrintWriter out;
        DataOutputStream os;
        if (wCard == 1) {
            ou = classvariables.soc.getOutputStream();
            out = new PrintWriter(new BufferedWriter(
                new OutputStreamWriter(ou)), true);
            os = new DataOutputStream(classvariables.soc.getOutputStream());
            os.writeBytes(mesaj);
        }
    }
}
```



## Ek-2.(devam) service.java

```
        } catch (SocketException e) {
        } catch (UnknownHostException e) {
        } catch (IOException e) {
        } catch (Exception e) {
        } finally {
        }
    }
    public void parcala(final String readed, final String hangiVeri) {
        if (run == true) {
            new Thread(new Runnable() {
                public void run() {
                    int bastaki = 6;
                    int virgul = 1;
                    int uzunluk = readed.length();
                    for (int i = 6; i < uzunluk; i++) {
                        String s = readed.substring(i, i + 1).toString(), t =
";";
                        int sondaki = i;
                        if (s.equals(t)) {
                            if (hangiVeri.equals(classvariables.kok+ classvariables.dimStatus)) {
                                if (bastaki == 6) {
                                    classvariables.D1 = readed.substring(bastaki, sondaki);
                                } else {
                                    if (virgul == 2) {
                                        classvariables.D2 = readed.substring(bastaki + 1, sondaki);
                                    } else if (virgul == 3) {
                                        classvariables.D3 = readed.substring(
                                        bastaki + 1, sondaki);
                                    } else if (virgul == 4) {
                                        classvariables.D4 = readed.substring(bastaki + 1, sondaki);
                                    } else if (virgul == 5) {
                                        classvariables.D5 = readed.substring(
                                        bastaki + 1, sondaki);
                                    } else if (virgul == 6) {
                                        classvariables.D6 = readed.substring(
                                        bastaki + 1, sondaki);
                                    }
                                }
                            }
                            bastaki = sondaki;
                            virgul++;
                        }
                    }
                }
            }).start();
        }
    }
    public void sistemKapali() {
        if (classvariables.R1.equals("0")) {
            Intent intentim = new Intent(this, sistemkapali.class);
            intentim.putExtra(
```

## Ek-2.(devam) service.java

```
        "com.alierbey.akilliev.SISTEMKAPALI", strName);
        intentim.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        startActivity(intentim);
        startActivity(new Intent(
            "com.alierbey.akilliev.SISTEMKAPALI"));
    }
}
String strName;
public void alarmcontrol() {
    if (tcpListenerCheckMessage(2, "alarmstatus") == 0) {
        // txtUyari.setText("Alarm Pasif Durumda");
        Log.d("alarm durumu", "pasif");
    } else {
        // txtUyari.setText("Alarm Aktif Durumda");
        Intent myIntent = new Intent(this, mainalarm.class);
        myIntent.putExtra("com.alierbey.akilliev.MAINALARM",
            strName);
        myIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        startActivity(myIntent);
    }
}
public void kontrolSensor() {

    Log.d("kontrol sensor", "kontrol ediliyor");
    final MediaPlayer ses = MediaPlayer.create(this, R.raw.tehlike);
    // tcpListenerCheck(1,R5);
    Integer kontrol = tcpListenerCheck(2, "I9");
    if (kontrol == null) {
    } else {
        if (kontrol == 0) {
            if (classvariables.sensorActive == true) {
                Log.d("su alarm", "aktif");
                Intent myIntent = new Intent(this, alarmsu.class);
                myIntent.putExtra(
                    "com.alierbey.akilliev.ALARMSU",
                    strName);

                myIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                startActivity(myIntent);
                classvariables.sensorActive = false;
            } else {
                this.uHandler = new Handler();
                ((Handler) this.uHandler).postDelayed(u_Runnable, 10000);
            }
        }
    }
}
private final Runnable a_Runnable = new Runnable() {
    public void run() {
        alarmcontrol();
        ((Handler) servisim.this.aHandler).postDelayed(a_Runnable, 10000);
    }
}
```

## Ek-2.(devam) service.java

```
};
private final Runnable u_Runnable = new Runnable() {
    public void run() {
        classvariables.uyari = 0;
        sistemKapali();
        ((Handler) servisim.this.kHandler).postDelayed(k_Runnable, 30000);
    }
};
@Override
public IBinder onBind(Intent intent) {
    return null;
}
}
```

### E- 3. mainhome.java

```
package com.alierbey.ckilliev;
import java.util.ArrayList;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.content.ActivityNotFoundException;
import android.content.ComponentName;
import android.content.ContentValues;
import android.content.Context;
import android.content.Intent;
import android.content.ServiceConnection;
import android.content.SharedPreferences;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Build;
import android.os.Bundle;
import android.os.Handler;
import android.os.IBinder;
import android.os.Message;
import android.os.Messenger;
import android.os.RemoteException;
import android.os.StrictMode;
import android.speech.RecognizerIntent;
import android.util.Log;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.ImageButton;
import android.widget.TextView;
import android.widget.Toast;
public class mainhome extends Activity {
    private static veritabani veritabanim;
    protected static final int RESULT_SPEECH = 1;
    private ImageButton btnKonus;
    private static TextView txtYazi;
    private TextView txtConnection;
protected Object mHome;
    static String degiskenDeger = "";
    static Integer mainOnOff;
    static Integer kontrollID = 0;
    static String degiskenID = "";
    static String dim = "";
    static Integer kartID;
    Integer demo = 0;
    static String gelenveri;
    final static servisim servisim = new servisim();
    final classvariables allVariables = new classvariables();
    protected static final boolean DEBUG = false;
    protected static final String TAG = "Speech";
    private int mBindFlag;
    private mainhome activityContext;
    private Intent speechService;
```

### Ek-3.(devam) mainhome.java

```
public static TextView methodText;
public static TextView resultsText;
private static Messenger mServiceMessenger;
@SuppressLint("NewApi")
@Override
protected void onCreate(Bundle savedInstanceState) {
    StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder()
        .permitAll().build();
    StrictMode.setThreadPolicy(policy);
    super.onCreate(savedInstanceState);
    setContentView(R.layout.mainhome);
    // speech
    methodText = (TextView) findViewById(R.id.textView1);
    resultsText = (TextView) findViewById(R.id.textView2);
    activityContext = this;
    speechService = new Intent(activityContext, MyService.class);
    activityContext.startService(speechService);
    mBindFlag = Build.VERSION.SDK_INT <
Build.VERSION_CODES.ICE_CREAM_SANDWICH ? 0
        : Context.BIND_ABOVE_CLIENT;
    //
    veritabanim = new veritabani(this);
    txtConnection = (TextView) findViewById(R.id.txtConnection);
    control();
    if (classvariables.boolDemo == true) {
        txtConnection.setText("DEMO");
        demo = 1;
    } else {
        this.mHome = new Handler();
        ((Handler) this.mHome).postDelayed(m_Home, 0);
    }
    btnKonus = (ImageButton) findViewById(R.id.mikrofon);
    txtYazi = (TextView) findViewById(R.id.txtYazi);
    txtConnection = (TextView) findViewById(R.id.txtConnection);
    ImageButton btnOdalar = (ImageButton) findViewById(R.id.menuBtnOdalar);
    ImageButton btnLambalar = (ImageButton) findViewById(R.id.menuBtnLambalar);
    ImageButton btnPencereler = (ImageButton) findViewById(R.id.menuBtnPencereler);
    ImageButton btnSulama = (ImageButton) findViewById(R.id.menuBtnSulama);
    ImageButton btnKameralar = (ImageButton) findViewById(R.id.menuBtnKameralar);
    ImageButton btnSenaryolar = (ImageButton) findViewById(R.id.menuBtnSenaryolar);
    ImageButton btnAlarm = (ImageButton) findViewById(R.id.menuBtnAlarm);
    ImageButton btnSensorler = (ImageButton) findViewById(R.id.menuBtnSensorler);
    ImageButton btnPrizler = (ImageButton) findViewById(R.id.menuBtnPrizler);
    ImageButton btnKlima = (ImageButton) findViewById(R.id.menuBtnKlima);
    ImageButton btnMedya = (ImageButton) findViewById(R.id.menuBtnMedya);
    ImageButton btnSettings = (ImageButton) findViewById(R.id.btnSettings);
    btnOdalar.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View arg0) {
            startActivity(new Intent("com.alierbey.akilliev.MAINODALAR"));
        }
    });
}
```

### Ek-3.(devam) mainhome.java

```
        }
    });
    btnLambalar.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View arg0) {
            startActivity(new
Intent("com.alierbey.akilliev.MAINLAMBALAR"));
        }
    });
    btnPencereler.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent("com.alierbey.akilliev.MAINPENCERELELER"));
        }
    });
    btnSulama.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new
Intent("com.alierbey.akilliev.MAINSULAMA"));
        }
    });
    btnKameralar.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent("com.alierbey.akilliev.MAINKAMERALAR"));
        }
    });
    btnSenaryolar.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent("com.alierbey.akilliev.MANUELSENARYO"));
        }
    });
    btnAlarm.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent("com.alierbey.akilliev.MAINALARM"));
        }
    });
    btnSensorler.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent("com.alierbey.akilliev.MAINSENSORLER"));
        }
    });
    btnPrizler.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent("com.alierbey.akilliev.MAINPRIZLER"));
        }
    });
}
```

### Ek-3.(devam) mainhome.java

```
        }
    }
});
btnKlima.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        startActivity(new Intent("com.alierbey.akilliev.MAINKLIMA"));
    }
});
btnMedya.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        startActivity(new Intent("com.alierbey.akilliev.MAINMEDYA"));
    }
});
btnKonus.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        Intent intent = new Intent(
            RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
            "tr");
        intent.putExtra(
            RecognizerIntent.EXTRA_SPEECH_INPUT_MINIMUM_LENGTH_MILLIS,
            "10000");
        try {
            startActivityForResult(intent, RESULT_SPEECH);
            txtYazi.setText("hazir");
        } catch (ActivityNotFoundException a) {
            Toast.makeText(getApplicationContext(), "Aygitiniz" bu
                uygulamay" desteklemiyor",
                Toast.LENGTH_SHORT).show();
        }
    }
});
}
@Override
protected void onStart() {
    super.onStart();
    bindService(new Intent(this, MyService.class), mServiceConnection,
        mBindFlag);
}
@Override
protected void onStop() {
    super.onStop();
    if (mServiceMessenger != null) {
        unbindService(mServiceConnection);
        mServiceMessenger = null;
    }
}
private final ServiceConnection mServiceConnection = new ServiceConnection() {
```

### Ek-3.(devam) mainhome.java

```
@Override
public void onServiceConnected(ComponentName name, IBinder service) {
    if (DEBUG) {
        Log.d(TAG, "onServiceConnected");} //$NON-NLS-1$
    mServiceMessenger = new Messenger(service);
    sendMessage(MyService.MSG_RECOGNIZER_START_LISTENING);
}

@Override
public void onServiceDisconnected(ComponentName name) {
    if (DEBUG) {
        Log.d(TAG, "onServiceDisconnected");} //$NON-NLS-1$
    mServiceMessenger = null;
}
}; // mServiceConnection
public static void sendMessage(int type) {
    Message msg = new Message();
    msg.what = type;
    try {
        mServiceMessenger.send(msg);
    } catch (RemoteException e) {
        e.printStackTrace();
    }
}

@Override
protected void onDestroy() {
    super.onDestroy();
    activityContext.stopService(speechService);
}

private final Runnable m_Home = new Runnable() {
    public void run() {
        // Bundle bundle = getIntent().getExtras();
        // if (bundle.getInt("demo") == 1) {
        // txtConnection.setText("demo");
        // } else {
        txtConnection.setText(classvariables.baglanti);
        esitle();
        // }
        ((Handler) mainhome.this.mHome).postDelayed(m_Home, 500);
    }
};

public static void eslesme(String kontrolAd) {
    SQLiteDatabase db = veritabanim.getReadableDatabase();
    Cursor cursor = db.query("tblMain", new String[] { "mainID",
        "mainAd", "mainOnOff" },
        "mainAd LIKE " + kontrolAd + "%", null, null, null, null);
    while (cursor.moveToNext()) {
        degerleriOku(cursor.getInt(0));
    }
}
```



### Ek-3.(devam) mainhome.java

```
    }
    public static void degerleriOku(long id) {
        SQLiteDatabase db = veritabanim.getReadableDatabase();
        String query = "SELECT tblMain.*,tblKontrol.*,tblDegisken.*,tblOda.*,tblKart.* "
            + " from tblMain inner join tblKontrol on
tblMain.kontrolID=tblKontrol.kontrolID "
            + " inner join tblOda on tblMain.odaID=tblOda.odaID"
            + " inner join tblDegisken on
tblMain.degiskenID=tblDegisken.degiskenID "
            + " inner join tblKart on tblDegisken.kartID=tblKart.kartID "
            + " where tblMain.mainID=?";
        Cursor cursor = db.rawQuery(query, new String[] { String.valueOf(id) });
        while (cursor.moveToNext()) {
            kartID = cursor.getInt(10);
            kontrolID = cursor.getInt(2);
            degiskenID = cursor.getString(9);
            degiskenDeger = cursor.getString(11);
            Integer onOff;
            Integer onData = servisim.tcpListenerCheck(
                Integer.parseInt(kartID.toString()),
                degiskenDeger.toString());
            if (onData != 0) {
                onOff = 0;
            } else {
                onOff = 50;
            }
            shortClick(onOff, degiskenDeger, kartID);
        }
    }
    public static void shortClick(Integer kontrol, String degisken,
        Integer sCard) {
        Relaybas(kontrol, degisken, sCard);
    }

    @SuppressWarnings("static-access")
    public static void Relaybas(Integer kontrol, String degisken, Integer kartID) {
        Integer sira = Integer.parseInt(degisken.substring(1, 2));
        String mesaj = null;
        if (degisken.substring(0, 1).equals("D")) {
            mesaj = classvariables.kok + classvariables.setDms + "," + (sira - 1) + "," +
kontrol + ",";
        } else {
            mesaj = classvariables.kok + classvariables.setRls + "," + (sira - 1) + "," +
kontrol + ",";
        }
        servisim.tcpSetSend(mesaj, kartID);
        Log.w("kontrol tetiklendi = ", mesaj);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        super.onCreateOptionsMenu(menu);
    }
}
```

### Ek-3.(devam) mainhome.java

```
        MenuInflater menutanim = getMenuInflater();
        menutanim.inflate(R.menu.anamenu, menu);
        return true;
    }

    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case R.id.yonetim:
                startActivity(new Intent(
                    "com.alierbey.akilliev.ADMIN"));
                return true;
            }
        return false;
    }
}
```

#### Ek-4. adminmainkontrol.java

```
package com.alierbey.akilliev;
import java.util.ArrayList;
import java.util.List;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.ContentValues;
import android.content.DialogInterface;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Spinner;
import android.widget.Toast;
public class adminmainkontrol extends Activity implements
    OnItemSelectedListener {
    private veritabani veritabanim;
    String itemId;
    Spinner spinnerDegisken;
    Spinner spinnerKontrol;
    Spinner spinnerOda;
    String secilenDegisken;
    String secilenKontrol;
    String secilenOda;
    String itemDegisken;
    String itemKontrol;
    String itemOda;
    @SuppressWarnings("unused")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.adminmainkontrol);
        veritabanim = new veritabani(this);
        Cursor cursor = KayitGetir();
        KayitGoster(cursor);
        spinnerDegisken = (Spinner) findViewById(R.id.degisken_spinner);
        spinnerKontrol = (Spinner) findViewById(R.id.kontrol_spinner);
        spinnerOda = (Spinner) findViewById(R.id.oda_spinner);
        spinnerDegisken.setOnItemSelectedListener(this);
        spinnerKontrol.setOnItemSelectedListener(this);
        spinnerOda.setOnItemSelectedListener(this);
        loadSpinnerDegiskenData();
        loadSpinnerKontrolData();
        loadSpinnerOdaData();
        final int id = 0;
```

#### Ek-4.(devam) adminmainkontrol.java

```
AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(this);
alertDialogBuilder
    .setTitle("Satır Güncellemek yada Silmek ister misiniz ?");
alertDialogBuilder

    .setCancelable(false)
    .setPositiveButton("Sil",
        new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface
dialog, int id) {

                KayitSil(Long.parseLong(itemId));

            }
        })
    .setNeutralButton("Güncelle",
        new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface
dialog, int id) {

                //
                guncellenecekKayitGetir(Long.parseLong(itemId));

                //
                btnGuncelle.setVisibility(View.VISIBLE);

                //
                edtTxtEkle.setVisibility(View.VISIBLE);

                //
                btnKayit.setVisibility(View.INVISIBLE);

                //
                btnVazgec.setVisibility(View.VISIBLE);

            }
        })
    .setNegativeButton("VazgeÁ",
        new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface
dialog, int id) {

                dialog.cancel();

            }
        });
final AlertDialog alertDialog = alertDialogBuilder.create();
ListView listView = (ListView) findViewById(R.id.listView1);
listView.setOnItemClickListener(new OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> arg0, View arg1,
        int position, long id) {
        String satir = array.get(position);
        int it = satir.indexOf(".");
        String sr = "";
        for (int i = 0; i < it; i++) {
            sr += satir.charAt(i);
        }
    }
});
```

#### Ek-4.(devam) adminmainkontrol.java

```
        itemId = sr;
        alertDialog.setMessage("secim=" + sr + "___ "
            + arg0.getItemAtPosition(position));
        alertDialog.show();
    }
});
final Button btnEkle = (Button) findViewById(R.id.btnYeniEkle);
btnEkle.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        Toast.makeText(getApplicationContext(),
            "Seçilen Degişken = " +
String.valueOf(spinnerDegisken.getSelectedItemAtPosition(0)) +
            "Seçilen Kontrol = "
+String.valueOf(spinnerKontrol.getSelectedItemAtPosition(0)) +
            "Seçilen Oda = " +String.valueOf(spinnerOda.getSelectedItemAtPosition(0)) ,
            Toast.LENGTH_LONG).show();
        KayitEkle();
    }
});
}
private void loadSpinnerDegiskenData() {
    veritabani db = new veritabani(getApplicationContext());
    List<String> lables = db.getDegiskenLabels();
    ArrayAdapter<String> dataAdapter = new
ArrayAdapter<String>(this,android.R.layout.simple_spinner_item,lables);
    dataAdapter

    .setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spinnerDegisken.setAdapter(dataAdapter);
}
private void loadSpinnerKontrolData() {
    veritabani db = new veritabani(getApplicationContext());
    List<String> lables = db.getKontrolLabels();
    ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this,
        android.R.layout.simple_spinner_item, lables);
    dataAdapter

    .setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spinnerKontrol.setAdapter(dataAdapter);
}
private void loadSpinnerOdaData() {
    veritabani db = new veritabani(getApplicationContext());
    List<String> lables = db.getOdaLabels();
    ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this,
        android.R.layout.simple_spinner_item, lables);
    dataAdapter

    .setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spinnerOda.setAdapter(dataAdapter);
}
}
```

#### Ek-4.(devam) adminmainkontrol.java

```
@Override
public void onNothingSelected(AdapterView<?> arg0) {
    // TODO Auto-generated method stub
}
// private String[] SELECT = { "mainID", "degiskenID", "kontrolID", "odaID",
// "mainAd" };
String query = "SELECT tblMain.*,tblKontrol.*,tblDegisken.*,tblOda.* "
    + " from tblMain inner join tblKontrol on
tblMain.kontrolID=tblKontrol.kontrolID "
    + " inner join tblOda on tblMain.odaID=tblOda.odaID"
    + " inner join tblDegisken on tblMain.degiskenID=tblDegisken.degiskenID ";
@SuppressWarnings("deprecation")
private Cursor KayitGetir() {
    SQLiteDatabase db = veritabanim.getReadableDatabase();
    Cursor cursor = db.rawQuery(query, null);
    startManagingCursor(cursor);
    return cursor;
}
ArrayList<String> array = new ArrayList<String>();
private void KayitGoster(Cursor cursor) {
    array.clear();
    int mainID = 0;
    String degiskenDeger = "";
    String kontrolAd = "";
    String odaAd = "";
    String mainAd = "";
    while (cursor.moveToNext()) {
        mainID = cursor.getInt(cursor.getColumnIndex("mainID"));
        degiskenDeger = cursor.getString(cursor
            .getColumnIndex("degiskenDeger"));
        kontrolAd = cursor.getString(cursor.getColumnIndex("kontrolAd"));
        odaAd = cursor.getString(cursor.getColumnIndex("odaAd"));
        mainAd = cursor.getString(cursor.getColumnIndex("mainAd"));
        array.add(mainID + ". " + degiskenDeger + " - " + kontrolAd
            + " - " + odaAd + " - " + mainAd);
    }
    final ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
        android.R.layout.simple_list_item_1, android.R.id.text1, array);
    final ListView list = (ListView) findViewById(R.id.listView1);
    list.setAdapter(adapter);
}
private void KayitEkle() {
    String satirDegisken = String.valueOf(spinnerDegisken.getSelectedItem());
    int itDegisken = satirDegisken.indexOf("-");
    String srDegisken = "";
    for (int i = 0; i < itDegisken; i++) {
        srDegisken += satirDegisken.charAt(i);
    }
    itemDegisken = srDegisken;
    String satirKontrol = String.valueOf(spinnerKontrol.getSelectedItem());
    int itKontrol = satirKontrol.indexOf("-");
    String srKontrol = "";
```

#### Ek-4.(devam) adminmainkontrol.java

```
        for (int i = 0; i < itKontrol; i++) {
            srKontrol += satirKontrol.charAt(i);
        }
        itemKontrol = srKontrol;
        String satirOda = String.valueOf(spinnerOda.getSelectedItem());
        int itOda = satirOda.indexOf("-");
        String srOda = "";
        for (int i = 0; i < itOda; i++) {
            srOda += satirOda.charAt(i);
        }
        itemOda = srOda;
        final EditText editText = (EditText) findViewById(R.id.editText1);
        SQLiteDatabase db = veritabanim.getWritableDatabase();
        ContentValues veriler = new ContentValues();
        veriler.put("degiskenID", itemDegisken.toString());
        veriler.put("kontrolID", itemKontrol.toString());
        veriler.put("odaID", itemOda.toString());
        veriler.put("mainAd", editText.getText().toString());
        db.insertOrThrow("tblMain", null, veriler);
        Cursor cursor = KayitGetir();
        KayitGoster(cursor);
        editText.setText("kayıt eklendi");
    }
    protected void KayitSil(long id) {
        SQLiteDatabase db = veritabanim.getReadableDatabase();
        db.delete("tblMain", "mainID" + "=" + id, null);
        Cursor cursor = KayitGetir();
        KayitGoster(cursor);
    }
}
```

## Ek-5. kontrollamba.java

```
package com.alierbey.akilliev;
import android.app.Activity;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.os.Handler;
import android.util.Log;
import android.view.View;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.SeekBar;
import android.widget.SeekBar.OnSeekBarChangeListener;
import android.widget.TextView;
import android.widget.Toast;
public class kontrollamba extends Activity {
    private veritabani db;
    String degiskenID = "";
    String dim = "";
    String degiskenDeger = "";
    Integer kartID;
    Integer mainOnOff;
    SeekBar sbar1;
    int ilk = 0;
    protected Object mHandler;
    ImageButton btnLambaAc;
    TextView txtgelen;
    service servisim = new service();
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.kontrollamba);
        db = new veritabani(this);
        this.mHandler = new Handler();
        ((Handler) this.mHandler).postDelayed(m_Runnable, 0);
        Bundle veriler = getIntent().getExtras();
        final Integer mainID = veriler.getInt("mainID");
        dataGet(mainID);
        btnLambaAc = (ImageButton) findViewById(R.id.btnSLambaAyar);
        sbar1 = (SeekBar) findViewById(R.id.seekBar1);
        txtgelen = (TextView) findViewById(R.id.txtGelenVeri);
        txtgelen.setVisibility(View.INVISIBLE);
        btnLambaAc.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (mainOnOff == 0) {
                    Relaybas("50", degiskenDeger, kartID);
                } else {
                    Relaybas("0", degiskenDeger, kartID);
                }
            }
        });
    }
}
```



## Ek-5.(devam) kontrollamba.java

```
sbar1.setOnSeekBarChangeListener(new OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress,
        boolean fromUser) {
        sbar1.setMax(25);
        Relaybas(String.valueOf(sbar1.getProgress()), degiskenDeger,
            kartID);
    }
});
}
public void controlUpdate() {
    if (mainOnOff == 0) {
        final TextView txtKontrolDurumu = (TextView)
findViewById(R.id.txtKontrolDurumu);
        txtKontrolDurumu.setText("Kapal");
        final ImageView img = (ImageView) findViewById(R.id.imageView1);
        img.setImageResource(R.drawable.lambasonuk);
        sbar1.setProgress(0);
    } else {
        final TextView txtKontrolDurumu = (TextView)
findViewById(R.id.txtKontrolDurumu);
        txtKontrolDurumu.setText("AAk");
        final ImageView img = (ImageView) findViewById(R.id.imageView1);
        img.setImageResource(R.drawable.lamba);
        sbar1.setProgress(mainOnOff);
    }
    final TextView txtgelen = (TextView) findViewById(R.id.txtGelenVeri);
    txtgelen.setText(String.valueOf(mainOnOff));
}
public void dataGet(long id) {
    SQLiteDatabase sdb = db.getReadableDatabase();
    String query = "SELECT tblMain.*,tblKontrol.*,tblDegisken.*,tblOda.*,tblKart.* "
        + " from tblMain inner join tblKontrol on
tblMain.kontrolID=tblKontrol.kontrolID "
        + " inner join tblOda on tblMain.odaID=tblOda.odaID "
        + " inner join tblDegisken on
tblMain.degiskenID=tblDegisken.degiskenID "
        + " inner join tblKart on tblDegisken.kartID=tblKart.kartID "
        + " where tblMain.mainID=?";
    Cursor cursor = sdb
        .rawQuery(query, new String[] { String.valueOf(id) });
    while (cursor.moveToNext()) {
        final TextView edtTxtEkle = (TextView) findViewById(R.id.textView2);
        edtTxtEkle.setText(cursor.getString(14));
        final TextView adi = (TextView) findViewById(R.id.txtConnection);
        adi.setText(cursor.getString(4));
        kartID = cursor.getInt(10);
        int kontrolID = cursor.getInt(2);
        degiskenID = cursor.getString(9);
        degiskenDeger = cursor.getString(11);
        if (kontrolID == 7) {
        } else {

```

## Ek-5.(devam) kontrollamba.java

```
        final SeekBar sbar1 = (SeekBar) findViewById(R.id.seekBar1);
        sbar1.setVisibility(View.INVISIBLE);
    }
    mainOnOff = servisim.tcpListenerCheck(kartID, degiskenDeger);
    Log.d("tek_kontrollambaMainOnOff", String.valueOf(mainOnOff));
    if (mainOnOff == null) {
        Log.d("mainonofficerde", String.valueOf(mainOnOff));
        mainOnOff = 0;
    }
}
}
private final Runnable m_Runnable = new Runnable() {
    public void run()
    {
        if (ilk == 0) {
            mainOnOff = servisim.tcpListenerCheck(kartID, degiskenDeger);
            Log.d("kontrollambaMainOnOff_degisti",
                String.valueOf(mainOnOff));
            if (mainOnOff == null) {
                Log.d("mainonofficerde", String.valueOf(mainOnOff));
                mainOnOff = 0;
            }
            controlUpdate();
        } else {
            if (mainOnOff != servisim.tcpListenerCheck(kartID,
                degiskenDeger)) {
                mainOnOff = servisim
                    .tcpListenerCheck(kartID,
degiskenDeger);
                Log.d("kontrollambaMainOnOff_degi ti",
                    String.valueOf(mainOnOff));
                if (mainOnOff == null) {
                    Log.d("mainonofficerde",
String.valueOf(mainOnOff));
                    mainOnOff = 0;
                }
                controlUpdate();
            }
        }
        ilk = 1;
        ((Handler) kontrollamba.this.mHandler).postDelayed(m_Runnable, 250);
    }
};
@SuppressWarnings("static-access")
public void Relaybas(String kontrol, String degisken, Integer kartID) {
    Integer sira = Integer.parseInt(degisken.substring(1, 2));
    String mesaj = null;
    if (degisken.substring(0, 1).equals("D")) {
        mesaj = classvariables.kok + classvariables.setDms + ", "
            + (sira - 1) + ", " + kontrol + ", ";
    } else {
        mesaj = classvariables.kok + classvariables.setRls + ", "
            + (sira - 1) + ", " + kontrol + ", ";
    }
}
```

## Ek-6. kontrolperde.java

```
package com.alierbey.akilliev;
import android.app.Activity;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.os.Handler;
import android.view.MotionEvent;
import android.view.View;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
public class kontrolpanjur extends Activity {
    private veritabani veritabanim;
    String degiskenID = "";
    String degiskenDeger = "";
    String degiskenDeger2 = "";
    Integer mainOnOff = 0;
    Integer kartID;
    Integer kontrolID;
    protected Object mHandler;
    protected Object mPerdeAc;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.kontrolpanjur);
        Toast.makeText(getApplicationContext(), "panjur", Toast.LENGTH_SHORT)
            .show();
        veritabanim = new veritabani(this);
        this.mHandler = new Handler();
        ((Handler) this.mHandler).postDelayed(m_Runnable, 1);
        Bundle veriler = getIntent().getExtras();
        final String mainAd = veriler.getString("mainAd");
        veriGetir(mainAd);
        final ImageButton btnPerdeAc = (ImageButton) findViewById(R.id.btnPerdeAc);
        final ImageButton btnPerdeKapat = (ImageButton)
            findViewById(R.id.btnPerdeKapat);
        final ImageButton btnPerdeUp = (ImageButton) findViewById(R.id.btnPerdeUp);
        final ImageButton btnPerdeDown = (ImageButton)
            findViewById(R.id.btnPerdeDown);
        final ImageButton btnPerdeDurdur = (ImageButton)
            findViewById(R.id.btnPerdeDurdur1);
        btnPerdeUp.setVisibility(View.INVISIBLE);
        btnPerdeDown.setVisibility(View.INVISIBLE);
        btnPerdeDurdur.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(getApplicationContext(), "durdur",
                    Toast.LENGTH_SHORT).show();
                Relaybas("0", degiskenDeger, kartID);
                Relaybas("0", degiskenDeger2, kartID);
            }
        });
        btnPerdeAc.setOnClickListener(new View.OnClickListener() {
```

## Ek-6.(devam) kontrolperde.java

```
@Override
public void onClick(View v) {
    Toast.makeText(getApplicationContext(), "Kapat",
        Toast.LENGTH_SHORT).show();
    Relaybas("0", degiskenDeger, kartID);
    Relaybas("100", degiskenDeger2, kartID);
}
});

btnPerdeKapat.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(getApplicationContext(), "Aç",
            Toast.LENGTH_SHORT).show();
        Relaybas("0", degiskenDeger2, kartID);
        Relaybas("100", degiskenDeger, kartID);
    }
});

btnPerdeDown.setOnTouchListener(new View.OnTouchListener() {
    private Handler btnPerdeDown;
    @Override
    public boolean onTouch(View v, MotionEvent event) {
        switch (event.getAction()) {
            case MotionEvent.ACTION_DOWN:
                if (btnPerdeDown != null)
                    return true;
                btnPerdeDown = new Handler();
                btnPerdeDown.postDelayed(mAction, 500);
                break;
            case MotionEvent.ACTION_UP:
                if (btnPerdeDown == null)
                    return true;
                btnPerdeDown.removeCallbacks(mAction);
                Relaybas("0", degiskenDeger2, kartID);
                btnPerdeDown = null;
                break;
        }
        return false;
    }
}

Runnable mAction = new Runnable() {
    @Override
    public void run() {
        Relaybas("50", degiskenDeger2, kartID);
        btnPerdeDown.postDelayed(this, 500);
    }
};

});

}
public void calistir() {
```

## Ek-6.(devam) kontrolperde.java

```
        this.mPerdeAc = new Handler();
        ((Handler) this.mPerdeAc).postDelayed(m_RunPerdeAc, 1);
    }
    public void kontrolGuncelle() {
        if (mainOnOff == 0) {
            final TextView txtKontrolDurumu = (TextView)
findViewById(R.id.txtKontrolDurumu);
            txtKontrolDurumu.setText("Kapalı");
            final ImageView img = (ImageView) findViewById(R.id.imageView1);
            img.setImageResource(R.drawable.perde);
        } else {
            final TextView txtKontrolDurumu = (TextView)
findViewById(R.id.txtKontrolDurumu);
            txtKontrolDurumu.setText("Açık");
            final ImageView img = (ImageView) findViewById(R.id.imageView1);
            img.setImageResource(R.drawable.perde);
        }
    }
    public void veriGetir(String mainAd) {
        SQLiteDatabase db = veritabanim.getReadableDatabase();

        String query = "SELECT tblMain.*,tblKontrol.*,tblDegisken.*,tblOda.*,tblKart.* "
            + " from tblMain inner join tblKontrol on
tblMain.kontrolID=tblKontrol.kontrolID "
            + " inner join tblOda on tblMain.odaID=tblOda.odaID"
            + " inner join tblDegisken on
tblMain.degiskenID=tblDegisken.degiskenID "
            + " inner join tblKart on tblDegisken.kartID=tblKart.kartID "
            + " where tblMain.mainAd=?";

        Cursor cursor = db.rawQuery(query,
            new String[] { String.valueOf(mainAd) });
        while (cursor.moveToNext()) {
            final TextView edtTxtEkle = (TextView) findViewById(R.id.textView2);
            edtTxtEkle.setText(cursor.getString(14));
            final TextView adi = (TextView) findViewById(R.id.txtConnection);
            adi.setText(cursor.getString(4));
            kontrolID = cursor.getInt(2);
            degiskenID = cursor.getString(9);
            kartID = cursor.getInt(10);
            if (degiskenDeger.equals("")) {
                degiskenDeger = cursor.getString(11);
            } else {
                degiskenDeger2 = cursor.getString(11);
            }
        }
    }
    private final Runnable m_RunPerdeAc = new Runnable() {
        @Override
        public void run() {
            Toast.makeText(getApplicationContext(), "runnable
aktif",Toast.LENGTH_SHORT).show();
            ((Handler) kontrolpanjur.this.mPerdeAc).postDelayed(m_RunPerdeAc,100);
        }
    }
}
```

## Ek-6.(devam) kontrolperde.java

```
    }
};
private final Runnable m_Runnable = new Runnable() {
    public void run()
    {
        kontrolGuncelle();
        ((Handler) kontrolpanjur.this.mHandler).postDelayed(m_Runnable,1000);
    }
};
public void Relaybas(String kontrol, String degisken, Integer kartID) {
    Integer sira = Integer.parseInt(degisken.substring(1, 2));
    String mesaj = null;
    if (degisken.substring(0, 1).equals("D")) {
        mesaj = classvariables.kok + classvariables.setDms + "," + (sira - 1) + "," +
kontrol + ",";
    } else {
        mesaj = classvariables.kok + classvariables.setRls + "," + (sira - 1) + "," +
kontrol + ",";
    }
    servisim.tcpSetSend(mesaj, kartID);
    Toast.makeText(getApplicationContext(),"releybas " + mesaj + " " + kartID,
Toast.LENGTH_SHORT).show();
}
}
```