T.C.
DİCLE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GSM TABANLI ÇOKLU TAKİP SİSTEMİ

Remziye CENGİZ

YÜKSEK LİSANS TEZİ

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ
ANABİLİM DALI

DİYARBAKIR
Mart - 2018

T.C

DİCLE UNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜ

DİYARBAKIR

Remziye Cengiz tarafından yapılan "GSM Tabanlı Çoklu Takip Sistemi" konulu bu çalışma, jürimiz tarafından Elektrik-Elektronik Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir

Jüri Üyesinin

Ünvanı Adı Soyadı

Başkan: Doç. Dr. Mehmet Siraç ÖZERDEM

Üye: Yrd. Doç. Dr. Muhammet Ali ARSERİM

Üye: Yrd. Doç. Dr. Serhat Berat EFE

Tez Savunma Sınavı Tarihi: 26/01/2018

Yukarıdaki bilgilerin doğruluğunu onaylarım.

.../...../20

Doç. Dr. Sevtap SÜMER EKER

ENSTİTÜ MÜDÜR V.

(MÜHÜR)

TEŞEKKÜR

Bu çalışma, Dicle Üniversitesi DÜBAP MÜHENDİSLİK.17.001 numaralı proje kapsamında desteklenmiştir. Desteklerinden ötürü DÜ Bilimsel Araştırma Projeleri Koordinatörlüğüne teşekkür ederim.

Tez çalışmam sırasında kıymetli bilgi, birikim ve tecrübeleri ile bana yol gösterici ve destek olan değerli danışman hocam sayın Doç. Dr. Mehmet Siraç ÖZERDEM'e, teşekkür ve saygılarımı sunarım.

Çalışmalarım boyunca manevi yardımını hiç esirgemeyen büyük bir sabır ve özveri gösteren değerli arkadaşım Zeynep AKKAYA' ya teşekkürü bir borç bilirim.

Ayrıca kıymetli zamanını benim hazırladığım bitirme projesine ayırıp çalışmalarım boyunca maddi manevi destekleriyle beni hiçbir zaman yalnız bırakmayan başta kardeşim Orhan CENGİZ'e ve ailem'e sonsuz teşekkür ederim.

İÇİNDEKİLER

		Sayfa
TEŞEKKÜR……..….		I
İÇİNDEKİLER…….....		II
ÖZET…….....		IV
ABSTRACT……..…….....		V
ÇİZELGE LİSTESİ…….....….....		VI
ŞEKİL LİSTESİ.....…….....……		VII
KISALTMA VE SİMGELER…….....……		X
1.	**GİRİŞ**…….....……....	1
1.1.	Ebeveyn Çocuk Takip Sistemi.....	1
1.2.	Denetimli Serbestlik Takip Sistemleri.....	2
1.3.	Lojistik Araçlar için Takip Sistemi.....	3
1.4.	Hasta Takip Uygulama Sistemleri.....	3
1.5.	Akıllı Ev Uygulama Sistemleri.....	4
2.	**KAYNAK ÖZETLERİ**…….....	7
3.	**MATERYAL ve METOT**…….....	13
3.1.	Global Konum Belirleme Sistemi (GPS-Global Positioning System)….....	13
3.2.	Önerilen Sistemin Donanımsal ve Yazılımsal Bileşenleri…….....	18
3.2.1.	Sistemin Donanımsal Bileşenleri…….....	18
3.2.2.	Sistemin Yazılımsal Bileşenleri…….....	23
3.3.	Önerilen Sistem Tasarımı…….....	24
3.3.1.	Gerçeklenen Tasarımın Donanım Yapısı…….....	35
3.3.2.	Gerçeklenen Tasarımın Yazılımsal Uygulamaları……...…....	37

3.3.3. Gerçeklenen Tasarımın Genel Özellikleri……..…….……..... 50

4. BULGULAR..... 55

5. SONUÇLAR…….....…..... 57

6. KAYNAKLAR…….……..... 59

EKLER..... 61

ÖZGEÇMİŞ..... 73

ÖZET

GSM TABANLI ÇOKLU TAKİP SİSTEMİ
YÜKSEK LİSANS TEZİ

Remziye CENGİZ

DİCLE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK-ELEKTRONİK ANABİLİM DALI

2018

GPS tabanlı çoklu takip sisteminin hem donanımsal hem de yazılımsal olarak gerçeklenmesi, bu tezin amacını teşkil etmektedir. Donanımsal olarak üç farklı donanım kullanılmıştır. Bunlar; GPS modülü, Arduino Uno geliştirme kartı ve GSM modülüdür. Yazılımsal olarak kullanılan iki farklı platform kullanılmıştır. Mobil uygulama için Arduino Studio platform ve geliştirme kartı için Arduino IDE kod platform kullanılmıştır. Başarıyla gerçeklenmiş olan çalışmada, sistemin oldukça yüksek performansta çalıştığı ve takip edilmek istenen kişi sayısının da artırılabileceği görülmüştür.

Literatürde bu konuda çok geniş çaplı çalışmaların olduğu görülmüştür. Bu çalışmalarda farklı amaçlar güdülmüş olsa da, düşünülen devre ve yazılım tasarımlarının aynı yapıda olduğu görülmüştür. Bu çalışma kapsamında literatürde konuyla ilişkili var olan çalışmalar detaylı olarak incelenmiştir ve piyasada var olan düzeneğin akademik açıdan incelenmesi sağlanmıştır.

Hızla gelişen teknolojinin elektronik ve otomasyon alanına yansıması ile geliştirme kartları çıkmıştır. Bu geliştirme kartları ile mikrodenetleyicili sistemler kolay ve düşük maliyetle tasarlanabilmektedir. Bu çalışma ile düşük maliyetli bir takip sisteminin oluşturulabileceği görülmüştür.

Anahtar kelimeler: GPS, GSM, Arduino Uno, mobil uygulama, konum tespiti

ABSTRACT

GSM BASEDMULTI TRACKING SYSTEM

MASTER THESİS

Remziye CENGİZ

DEPARTMENT OF ELECTRICAL ELECTRONICS ENGINEERING
INSTITUTE OF NATURAL AND APPLIED SCIENCES
UNIVERSITY OF DICLE

2018

The purpose of this thesis is that the GPS based multi tracking system is implemented both hardware and software. Three different hardware are used. These; GPS module, Arduino Uno development card and GSM module. Two different software platforms are used. Arduino Studio platform for mobile application and Arduino IDE code platform for development card. In the successful implementation, it was seen that the system worked very well and the number of people could be increased.

It has been seen in the literature that there is a wide range of studies on this subject. Although different goals have been pursued in these studies, it has been seen that the circuit and software designs considered have the same structure.

In the scope of this study, the studies related to the subject in the literature have been examined in detail and the academic basis of the existing system in the market has been examined.

Development cards have emerged as a result of the rapid development of technology in the field of electronics and automation.

With these development cards, microcontroller systems can be designed easily and at low cost. This study shows that a low cost tracking system can be created.

Keywords: GPS, GSM, Arduino Uno, mobile app, location determination

ÇİZELGE LİSTESİ

Çizelge No		Sayfa
Çizelge 3.1.	Çeşitli baz değerlerine göre elde edilen efemeris hataları	16

ŞEKİL LİSTESİ

Şekil No		Sayfa
Şekil 1.1.	Ebeveyn Çocuk Takip Sistemi için Örnek Bir Uygulama	2
Şekil 1.2.	Denetimli Serbestlik Takip Sistemi için Örnek Bir Uygulama	2
Şekil 1.3.	Lojistik araçlar için takip sistemine ilişkin örnek bir uygulama	3
Şekil 1.4	Mobil Hasta Takip Sistemi Yapısı	4
Şekil 1.5	(a) Sisteme Giriş ve (b) Ana Ekran	5
Şekil 2.1	NAVSTAR Sisteminin Fiziksel Görünümü	7
Şekil 3.1.	Uyduların 6 ayrı yörüngede konumlandırılması	13
Şekil 3.2.	Uydular tarafından üretilen L1 taşıyıcı işareti	14
Şekil 3.3.	Uydular tarafından üretilen L2 taşıyıcı işareti	14
Şekil 3.4.	Uyduların konumlarını izleyen istasyonlar	15
Şekil 3.5.	Uydu konum hatası bağıntısı	15
Şekil 3.6.	GPS tekmolojisnde konum belirleme yaklaşımı	16
Şekil 3.7.	GPS ile mutlak konum belirleme	17
Şekil 3.8.	GPS ile göreli konum belirleme	18
Şekil 3.9.	GY-NEO6MV2 donanımın dış kılıfı	19
Şekil 3.10.	GSM modülüne ilişkin dış kılıfı	20
Şekil 3.11.	Arduino UNO geliştirme kartı	21
Şekil 3.12.	Anket Sonuçları	24
Şekil 3.13	Android Studio platformunda başlangıç penceresi	25
Şekil 3.14.	Android Studio platformunda proje isminin belirlenmesi	25
Şekil 3.15.	Uygulamanın çalışacağı platformun (API) belirlenmesi	26
Şekil 3.16.	Uygulamanın teması belirlenir	26

Şekil 3.17. Aktivite ve Layout'un adlarının belirlenmesi 27

Şekil 3.18. Oluşturulan uygulamanın ana aktivite penceresi 27

Şekil 3.19. Oluşturulan ana aktivitenin ekran görüntüsü 28

Şekil 3.20. AVD Manager ile sanal makina kurulum penceresi 28

Şekil 3.21. Oluşturulan sanal cihazın API seçimi 29

Şekil 3.22. Sanal cihaza ilişkin işletim sistemi belirlenir 29

Şekil 3.23, Sanal cihaza ilişkin konfigürasyon ayarları belirlenir 30

Şekil 3.24. Oluşturulan sanal cihaz penceresi 30

Şekil 3.25. Oluşturulan sanal cihazın görüntüsü 31

Şekil 3.26. Ana aktivitenin ekranına eklenen öğeler 31

Şekil 3.27. Uygulamanın çalıştırılması ile elde edilen ekran görüntüsü 32

Şekil 3.28. Arduino'da kodun dönüşüm düzeneği 33

Şekil 3.29. Arduino IDE kod editörü 33

Şekil 3.30. IDE kod editöründe Arduino geliştirme kartının belirlenmesi 34

Şekil 3.31. Tasarlanan GSM tabanlı takip sistemi ve sistem bileşenleri 35

Şekil 3.32. Tasarlanan Sistemin Devre Şeması 36

Şekil 3.33. Sistemdeki birimlerin kendi aralarındaki işaret akışı 36

Şekil 3.34. Arduino geliştirme kartı için gerçeklenen işlem adımları 37

Şekil 3.35. Samsung Galaxy Cihazında "Cihaz hakkında"opsiyonu 43

Şekil 3.36. Samsung Galaxy Cihazında "Yapım numarası"bölümü 43

Şekil 3.37. Samsung Galaxy Cihazında "USB hata ayıkla"opsiyonu 44

Şekil 3.38. Mobil uygulama için tasarlanan bilgisayar arayüzü 45

Şekil 3.39. Mobil uygulama için tasarlanan arayüz 46

Şekil 3.40. Konum bilgilerinin global olarak tanımlanması 47

Şekil 3.41. Butonlar için ana aktivitede yazılan kod 48

Şekil 3.42. Konumun haritada gösterilmesi için yazılan kod 49

Şekil 3.43. SMS lerin dinlenmesi için Receive SMS Activity bölümünde yazılan kod 49

Şekil 3.44. Mobil uygulamaya gelen SMS'lerin kayda alınması 50

Şekil 3.45. Arduino Studio platformunda hazırlanan akış diyagramı 52

Şekil 3.46. SMSDemo Uygulaması 53

Şekil 3.47. Mobil uygulamaya gelen SMS örneği 53

Şekil 3.48. Haritadan konumu bulunan örnek cihaz 54

KISALTMA VE SİMGELER

ÇKS	: Çoklu Takip Sistemi
FYS	: Filo Yönetim Sistemleri
GPS	: Global Positioning System
GSM	: Mobil İletişim için Küresel Sistem
NAVSTAR	: Zamanlama Kullanımı ve Menzilleme NavigasyonSistemi
SMS	: Kısa Mesaj Hizmeti

1. GİRİŞ

Teknolojik gelişmeler artıkça elektronik,bilişim ve yazılım alanlarındaki gelişmeler de hızla büyümektedir.Özellikle donanım alanındaki malzemelerin hızlı bir şekilde temin edilmesi oldukça kolaylaşmıştır.Kullanılan mikroişlemcilerin küçülmesi ve ucuzlaması ile sistemlerin kendi işlemcisini içerek forma geldiği görülmektedir. Bu kolaylıklar sayesinde, sistemlere ilişkin devrelerin kurulumu,hızlı bir şekilde gerçeklenebilmektedir. Tez konumuz olan GSM tabanlı çoklu takip sistemi, hem donanımsal ve hem de yazılımsal olarak söz konusu kolaylıklar çerçevesinde başarıyla gerçeklenmiştir. Bu gerçeklenen sistem ile istenilen bir objenin yada canlı bir nesnenin enlem ve boylam bilgilerini kolaylıkla tespit edebilmekte ve tespit edilen konum verileri ile nesnesinin haritadaki konumu belirlenebilir.

Günümüz teknolojik gelişmeler dikkate alındığında, Küresel Konumlama Sistemi (GPS) ve GSM entegrasyonlu sistemlerin kullanışlı olduğu gözlenmektedir. GPS ve GSM entegrasyonlu sistemler, gerçek zamanlı takip sistemlerinin temelini oluşturmaktadır. Bu sistemlerin genel olarak çalışma düzeneği şu şekilde tanımlanabilir: i) Canlı veya cansız nesneye GPS donanımı entegre edilir. Mobil olarak gezinen nesnenin tüm koordinat verileri, GPS tarafından belirlenir. ii) Koordinat (Enlem ve boylam) verileri, GSM üzerinden izleme sunucusuna iletilir. iii) Sunucu (bilgisayar, tablet veya akıllı telefon), aldığı verileri harita üzerinde konumlandırır ve ekrana yansıtır. Bu sistemlerin, çoklu nesne takibi yapabilme özelliklerinin de olduğu görülebilmektedir.

Yukarıda anlatılan sistemler için örnekler verilebilir. Günümüzde aktif olarak kullanılan önemli uygulamaların bazıları aşağıda belirtilmiştir.

1.1. Ebeveyn Çocuk Takip Sistemi

Türkiye İstatistik Kurumu'nun 81 ilde yaptığı araştırmaya göre 2008-2011 yılları arasında kaybolan çocuk sayısı 27000’i geçmiş olmasından (TÜİK,2011) ve gündelik hayatımızda karşılaşılan kötü olaylardan dolayı, aileler çocuklarının eş zamanlı olarak konumlarını öğrenme ihtiyacı duymuşlardır. Günümüzde,çocuk güvenliği gittikçe artan bir sorun olmakla birlikte, bu sorunun teknolojik gelişmeler ile aşılabileceği görülmüştür. GPS ve GSM entegrasyonlu sistemler ile ebeveynler, çocuklarının bulundukları yeri/konumuher an öğrenebilmektedirler. Bunu sağlayan sistemlerden bir

tanesi de, GPS entegreli GSM cep telefonlarıdır (Şekil 1.1). Bu telefonlar ile hem iletişim kurulabilmekte hem de konum tespiti yapılabilmektedir. Bu alan, birçok akademik çalışmaların konusu olmuştur (Bayıroğlu,H.2013).



Şekil 1.1. Ebeveyn çocuk takip sistemi için örnek bir uygulama

1.2. Denetimli Serbestlik Takip Sistemleri (e-kelepçe)

Bu sistemin amacı; denetimli sebestlik hizmetlerinin kurumsal kapasitelerini artırmak, suç mağdurlarının yeniden suç işlemesini engellemek ve topluma faydalı bireyler haline getirmektir. Herhangi bir adli sebepten denetimli olarak serbest bırakılmış kişilerin, el veya ağak bileklerine bu sistem entegre edilir. Bu sistem günümüzde e-kelepçe olarak adlandırılır ve kişinin takibi sağlanır. Konuyla ilişkili bir örnek uygulama Şekil 1.2'de gösterilmiştir.Bu sistem günümüzde,Elektronik İzleme Sistemi olarak da adlandırılır ve Adalet Bakanlığınca uygulanan bir infaz yönetimidir. Bu sistem ile toplum içindeki suç mağdurunun sürekli takibi ve denetimi yapılabilmektedir (Havelsan ,2017).



Şekil 1.2. Denetimli serbestlik takip sistemi için örnek bir uygulama

1.3. Lojistik Araçlar için Takip Sistemi

Günümüzde firmalar, ürünlerinin fabrikadan çıkıp, son tüketiciye ulaşıncaya kadar ki süreçte, araçlarının takip edilmesini isteyebilmektedirler. Bu takip sisteminin çeşitli nedenleri vardır. Bu takip sistemi ile araçların konumu, doğru amaçlar için kullanılıp kullanılmadığı, doğru noktalarda mola verilip verilmediği anlaşılabilmektedir. Bu sistem, araç içerisine entegre edilen vericiler ile sayısal bir ağ oluşturur ve araçların izlenmesini sağlar. Bu sistemler, lojistik araç takip sistemleri olarak bilinir ve firma araçlarının takibinin sağlanması için kullanılır. Konuyla ilişkili bir örnek uygulama şeması, Şekil 1.3'de gösterilmiştir.



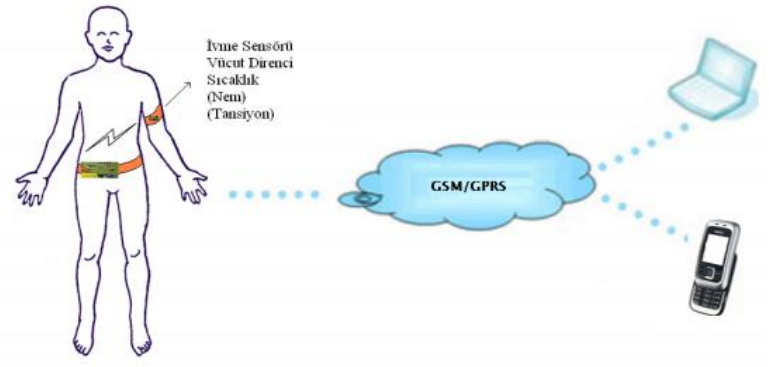
Şekil 1.3. Lojistik araçlarda takip sistemine ilişkin örnek bir uygulama

Bu sistem ile iş üretimi artırılabilir, standart koşullar (hız sınırı, zamanlama, kritik yükler için soğuk-sıcak zincirinin kontrolü) iyileştirilebilir ve filo düzeni sağlanabilir. Dolayısıyla bu sistem, mobil araçlar ile firma arasında bir köprü niteliğindedir (Erdal, M., Çancı, M., 2003).

1.4. Hasta Takip Uygulama Sistemleri

Tasarlanan sistemde;mobil hasta üzerinden alınan veriler, kablosuz olarak genel hasta takip birimine aktarılabilmektedir. Hasta takip biriminde alınan anlık yaşamsal bilgiler, hem kayıt edilmekte hem de olası acil durumlarda hastanın konumu saptanabilmekte ve hastaya GPRS veya SMS servisleri üzerinden mesaj gönderilebilmektedir. Böylelikle hem kişiye acil durumlarda hızlı müdahale imkanı sağlanır hem de kişideki hastalığının seyri takibe alınır. Tasarlanan sistem 2 ana bölümden oluşur. Birinci kısım; hastanın sıcaklık, nem, nabız, konum gibi bilgilerinin

ölçüldüğü ve kablosuz olarak hasta takip birimine aktarıldığı birimdir. İkinci kısım ise ölçüm biriminden alınan verilerinin yorumlandığı birimdir(Şanlı,M., 2009). Konuyla ilişkili bir örnek uygulama şeması Şekil 1.4'de gösterilmiştir.



Şekil 1.4. Mobil Hasta Takip Sistemi Yapısı

1.5. Akıllı Ev Uygulama Sistemleri

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 gömülü teknoloji, çalıştığımız ya da yaşadığımız ev ortamı gibi pek çok fiziki alanda hayatımızıkolaylaştırmaktadır.Bu bağlamda evinizde oturduğunuz yerden aydınlatma sistemlerinin kontrol edilmesi, sabah kalktığınızda perdelerin otomatik olarak açılması, dışarı çıktığınızda ütüyü açık unuttuğunuz düşüncesi ile geri dönmeden tehlikenin engellenmesi, çimlerin nem oranına göre otomatik sulanmasının sağlanması gibi rutin ev işlerini yönetebilen sistemlere gereksinim artmaktadır. Günümüz teknolojilerinden yararlanılarak oluşturulan bu tür 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 üzere tasarlanmıştır (Gül, F, 2010). Konuyla ilişkili bir örnek

uygulama Şekil 1.5'de gösterilmiştir. Şekilde, akıllı telefon üzerinden evin kontrolünü sağlamayı amaçlayan bir örnek uygulamanın önyüzü gösterilmiştir.



(a) (b)

Şekil 1.5. (a) Sisteme Giriş önyüzü ve (b) Ana Ekran

Bu tez kapsamında önerilen sistem, gerçek zamanlı olarak birden çok nesnenin izlenmesini amaçlayan Çoklu Takip Sistemdir (ÇKS). Bu çalışmada, GSM(3G) üzerinden haberleşen bir uygulama gerçekleştirilmiştir. Sistemin kapsamında var olan uygulamalar aşağıda listelenmiştir.

-Konum Tespiti (GPS ile konum belirleme)

-Akıllı Telefon Uygulaması (GPS ile iletişime geçilip, mobil nesnesinin konum bilgilerinin alınmasına ilişkin yazılım)

-Akıllı Telefonda, nesnelere ilişkin konumlarının haritada gösterilmesi.

2. KAYNAK ÖZETLERİ

Bu tez çalışmasının amacı, gerçek zamanlı olarak GSM tabanlı Çoklu Takip Sisteminin hem donanımsal hem de yazılımsal olarak gerçeklemektir. Bu bölümde ise söz konusu alanda yapılan temel çalışmalar ve yapılan uygulama örnekleri anlatılmıştır.

Uzay tabanlı konumlama çalışmaları, Amerika Birleşik Devletleri askeri kuvvetleri ve NASA tarafından 1960'lara doğru çalışılmaya başlanmıştır. İki boyutlu konum bilgisi sağlayan bu sistem TRANSIT (U.S. Navy Navigation Satellite System) olarak adlandırılır. Eş zamanlı olarak aynı işlevi gerçekleyen Tsikada adında bir sistem, Rusya tarafından geliştirilmiştir. TRANSIT sistemi, 1967 yılında ticari amaç için dünya çapında kullanılmaya başlanmıştır. Bu uydu sisteminin parsel ve nirengi ölçümleri için gerekli doğruluğu sağlayamadığı tespit edilmiştir. Bunun sebebi ise bu kapsamda kullanılan uydulara ilişkin yüksekliğin 1100km olduğu ve bundan dolayı uyduların yerel yerçekiminden etkilendiği olarak belirtilmiştir. Bu nedenle, 1974 yılında A.B.D. Savunma Bakanlığı, navigasyon hizmetlerini karşılamak amacıyla çalışmalara başlamış ve böylece NAVSTAR-GPS (NAVigation Satellite Timing And Ranging – Global Positioning System) sisteminin temelleri atılmıştır (Şekil 2.1). 28 Haziran 1983 tarihiitibariye, A.B.D. Savunma Bakanlığı tarafından GPS'in kullanımı sivillere açılmıştır (GPS Notları,2017).



Şekil 2.1. NAVSTAR sisteminin fiziksel görünümü

GPS'in sivillere açılması ile geniş bir yelpazede uygulamaların geliştirildiği gözlenmiştir. Farklı alanlarda gerçeklenen temel uygulamaların yer aldığı bazı literatür özetleri aşağıda verilmiştir. Cemiyetlerin nerede yaşadıkları bilgisine olan ihtiyaçları tarihin neredeyse bütün döneminde önemini koruyan bir problem olmuştur. I ve II. Dünya savaşı sırasında, güçlü finansmanasahip ülkelerin, konum bilgisi oluşturma amacına uygun teknolojilere yaptıkları yatırımlar görülebilmektedir. Örneğin, araç filosu barındıran şirketlerin (veya kurumların), araçlarına ait konum bilgilerinin kayda alınması ve kontrolü, ticari (veya kurumsal)stratejilerinin bir parçasıdır. Bu gruba girebilecek uygulamalar; süpermarket filoları, banka araçları, polis araçlarıve/veya önemli ürün taşıyan firma araçlarıdır. Bu araçların takibi için gerçeklenen sistem, Filo Yönetim Sistemi (FYS) olarak tanımlanır (Caner GÜNEY ve ark., 2002).

GPS'in kullanımına ilişkin diğer bir uygulama ise Araç Takip Sistemidir (ATS). Bu sistem, GPS modülünden aldığı koordinat verilerini, bir mikrokontrolörüzerinden SMS sistemi ile kolayca iletilebilen özel bir iletişim protokolüne (AGFES)dönüştürür. Bu protokol ile hücresel ağda bulunan bir mobil telefondan, merkezde bulunan birmobil telefona aktarabilen (enlem-boylam)bilgiler, bilgisayardaki haritagörüntüsüne aktarılabilir (Erhan Küçük ve ark., 2007).Sun Q. ve ark. (2017) tarafından yapılan çalışmada ise; sürücülere ilişkin davranışlarının değerlendirilmesinde, GPS tabanlı sistemlerin kullanılmasının etkisi incelenmiştir.

Tez konusunu ilgilendiren diğer bir önemli çalışma; GPS işaretleri ile konum belirleme - raporlama işlemleri,İbrahim Özgön ve Mesut Konak tarafından mobil - web platformunda tasarlanmıştır (İbrahim Özgön ve Mesut Konak, 2013). Jung-Han Kim ve Jun-Ho Oh (2000) tarafından yapılan çalışmada ise, GPS verisi ile araç takip algoritması geliştirilmiştir.

Android tabanlı akıllı telefonlarla uyumlu GPS ve bilgi iletiminde GPRS kullanılan bir takip sistemi,Gupta ve arkadaşları tarafından geliştirilmiştir. İşlem akışında yeralan adımlar; kullanıcılara ait konum bilgileri GPS sisteminden alınır, IMEI numarası ilave edilir ve MYSQL sunucusundaki veritabanına GPRS ile gönderilir. Java Script Page'de tasarlanan web sayfasında, kullanıcıya ait konum bilgisi Google haritasına.aktarılır(Gupta,R.ve.ark.,2011).Bu çalışmaya benzer bir uygulama da Kristian

Y. ve ark. (2012) tarafından yapılmıştır. Çalışmalarında Android tabanlı telefonlarda GPS ve SMS kullanılarak bir takip sistemi başarıyla geliştirilmiştir.

Çoklu takip konusuna ilişkili diğer bir önemli çalışma ise lojistik uygulamalardır. Bu alanda önemli bir çalışma, Erdal M. ve Çancı M, tarafından gerçeklenmiştir. Çalışma kapsamında, internet erişiminin olduğu her yerde yapılacak olan güncellemeler ile ülkenin tüm adres bilgilerine (şehir, ilçe, sokak, mahalle, köy) erişilebilmektedir. Bu sistemi kullanan kullanıcı gerektiğinde sürücüler ile mesajlaşabilmektedir. Ayrıca yine aynı kanal üzerinden iş emri verebilirmektedir. Bu şekilde araçların, gözetim altında güvenli bir şekilde seyrinin takip edilebildiği gözlenmiştir (Erdal M., Çancı M., 2003). Prasanna K.R. ve Hemalatha M. (2012) tarafından yapılan çalışmada, RFID GPS ve GSM tabanlı lojistik araçların takip mekanizması çalışılmıştır.

Konum belirleme ile ilişkili diğer bir çalışma, ebeveyv çocuk takip sistemidir. Bu alanda çalışmış olan Chandra ve arkadaşları, Java tabanlı telefonlarda koşturulabilecek bir uygulama geliştirmişlerdir. Çalışmada mobil uygulama kısmı, J2ME platformunda yazılmış ve konum verisini alabilmek için GPS kullanılmıştır. Bu çalışmada ayrıca mesaj ile alınan verilerin Google haritasına aktarılması sağlanmıştır (Chandra, A ve ark., 2011). Bu çalışmanın diğer bir benzeri, Liao ve arkadaşları tarafından yapılmıştır. Uygulama, kişi gideceği noktayı önceden ilgili kişilere mesaj yoluyla bildirmesinden oluşur (Liao, Y. ve ark., 2011).

GPS'in kullanımına ilişkin diğer bir alan ise sağlık alanıdır. Bu alanda yapılan çalışmalardan bazıları şöyle özetlenebilir; kronik rahatsızlığı olan çocukların 24 saat takip edilebilecekleri bir mobil sistem tasarlanmıştır. Bu acil durum sisteminde; hasta çocuk, yardıma ihtiyaç duyduğunda bir butona basarak web sunucusuna mesaj atabilmektedir. Mesajın içeriğine, çocuğun ID numarası ve konum bilgisi otomatik olarak eklenir. C# programlamayla oluşturulmuş olan web sayfasında, çocuğun konumu harita üzerinde gösterilir. Ayrıca önceden veri tabanında kayıtlı olan çocuğun ailesine, hastaneye ve doktoruna mesaj iletilir. Bu çalışmada mobil telefon olarak Windows Phone tercih edilmiştir (Ismael, A.G.,2012). Diğer bir çalışmada, Pakistanın kırsal bölgelerinde çocuk felci aşılarının takibi GSM tabanlı yapılmıştır (Subhash C. ve ark. 2017). Sablier J. ve ark. tarafından demans (bunama) hastalığı olan kişilerin GPS tabanlı

takip sistemi ile değerlendirilmesi çalışılmıştır (Sablier J. ve ark., 2011). Buna benzer bir çalışma, yaşlı ve Alzheimer hastaları için yapılmıştır. Çalışmada GPS tabanlı bir takip sistemi kullanılmıştır (Sara P. ve Carlos A., 2012).

Konuyla ilişkili diğer bir önemli alan e-kelepçe uygulamasıdır. Bu sistem, dünyada gelişmiş ülkelerde kullanılmaktadır. Bu sistem, ülkenin sosyo-ekonomik ve kültürel yapısına uygun şekilde tasarlanmış ve uygulamaya konumuştur (Denetimli Serbestlik Takip Sistemleri, 2017). Hendrikx J. ve ark. (2015) tarafından yapılan bir çalışmada, konumların tercihlerine göre GPS tabanlı sesli kelepçe sistemleri çalışılmıştır.

Konuyla ilişkili diğer bir önemli uygulama ise akıllı ev uygulamasıdır. Özer tarafından yapılan çalışmada, evin birçok noktasının kontrolü ve güvenliğini sağlanmıştır. Çalışmada gerçeklenen akıllı ev otomasyon sistemi kapsamında bulunan bazı özellikler;bitki ve hayvanların yem ve su ihtiyacını karşılayabilmesi, evin sıcaklığını ayarlayabilmesi, pişirme fırınının ayarlanması, klima ve kalorifer sisteminin çalıştırılabilmesi, aydınlatma ve bahçe sulama işlemlerinin kontrol edilebilmesi,su baskını/hırsızlık/yangın vb. durumlarda uyarıcı mesajların gönderilebilmesidir (Özer, H.İ. 2005).

GPS tabanlı gerçeklenen diğer farklı uygulamalar ise şöyle listelenebilir:

- Yaşlı kişilere ilişkin araç sürüş davranışarının GPS takip sistemi ile analizi (Zhao Y., ve ark. 2018).
- Özel bir kuş türüne ilişkin davranışların yüksek çözünürlüklü GPS tabanlı sistemi ile belirlenmesi (Silva J.P. ve ark., 2017).
- GPS tabanlı veriler ile akıncı güvercin türüne ilişkin uçuş dinamiğinin modellenmesi (Zaitouny A., ve ark., 2017).
- GPS-LEO takip verisi ile atmosferik yoğunluğun ölçülmesi (Kuang D., 2014).

Literatürden görüldüğü üzere, donanımsal (GSM Modülü, mikrodenetleyici, vs.)olarak gerçeklenen sistemlerin birbirine benzer olduğu ancak uygulandığı alanların farklı olduğu görülmektedir. Bu tezçalışmasında; GPS teknolojisi ile akıllı cep telefonları, navigasyon cihazları ve bazı gömülü sistemlerden konum bilgileri alınarak

kullanıcıya sunulması amaçlanmıştır. Proje kapsamında son yıllarda ücretsiz ve açık kaynak kodlu olması özellikleri ile revaçta olan Google firmasının desteklediği Android OS kullanılmıştır. Bu işletim sisteminde, kökü çok eskiye dayanan bir teknoloji olmasıyla da ve gelişmesini hızla sürdüren Microsoft firmasının NET teknolojisi kullanılmıştır.

3. MATERYAL VE METOT

Bu tez çalışmada GSM tabanlı çoklu takip sistemininngerçeklenmesi amaçlanmıştır. Bu amaç için kullanılan donanımlar ve kullanılan yöntemlerin detayları bu bölümde açıklanmıştır. Öncelikle çalışılan tez konusunun temelini teşkil eden GPS ile konum belirlemenin detayları açıklanmış ve ardından tasarlanan sistem donanımları ile birlikte yazılım aşamaları anlatılmıştır.

3.1. Global Konum Belirleme Sistemi (GPS - Global Positioning System)

GPS, hava şartları ne olursa olsun 7 gün 24 saat, dünyanın herhangi bir yerinde konum bilgisi sağlayan teknolojik bir sistemdir. GPS; uzay, kontrol ve kullanıcı bölümü olmak üzere üç ana bölümden oluşur.

a) Uzay Bölümü: Herhangi bir noktada her an uygun geometride en az dört uydunun görülebileceği şekilde planlanmış 24 uydu (ve 3 yedek uydu), sistemin uzay bölümünü oluşturur. Bu uydular yerden yaklaşık 20200 km uzaklıkta olup, ekvatorla 55° lik açı yapan 6 ayrı yörünge düzlemine yerleştirilmiştir. 12 saatlik peryotlara sahip olan uyduların yerleşim düzeni Şekil 3.1'de gösterilmiştir. Her uydu belirli bir matematiksel modele göre sözde rastgele gürültü (pseudorandom noise - PRN) üretir ve bu üretilen sayı ile uydu tanınır.



Şekil 3.1. Uyduların 6 ayrı yörüngede konumlandırılması

Her uydu L1 ve L2 olmak üzere iki farklı sinyal yayınlar. Bu sinyaller taşıyıcı sinyallerdir. L1 taşıyıcı sinyali, 1575.42 Mhz frekansına ve yaklaşık 19 cm dalga boyuna sahiptir. L2 taşıyıcı sinyali ise 1227.60 Mhz frekansına ve 24 cm dalga boyunda sahiptir (King E.V.ve ark.,1987).

L1 üzerinden yapılan yayınlar; 1.023 MHz frekansındaki (dalga boyu 300m) C/A-Kod (Course Acquisition) ve 10.23 MHz frekansındaki (dalga boyu 30 m) P-Kod(Precise) ilenavigasyon mesajıdır. L2 taşıyıcısı üzerinde ise sadece P-Kod ve navigasyon mesajı bulunur. L1 sinyal bandı, tüm kullanıcılara açık olmakla birlikte L2 ise sadece askeri amaçlı kullanılmaktadır (Sevindi C., 2005).

Uydular tarafından üretilen L1 taşıyıcı işareti, (3.2) bağıntısında belirtildiği şekilde tanımlanır (Hui,1982).

$$L1(t)=A_t(t)P(t)D(t)\cos(\xi_1 t+\varepsilon)+B_tG(t)D(t)\sin(\xi_1 t+\varepsilon) \quad (3.2)$$

Bağıntıda bulunan L1(t) değişkeni, zamana bağlı L1 sinyalini ifade eder. P(t) değişkeni P kodu ve G(t) değişkeni ise C/A kodu ifade eder. A_t ve $B_t$değişkenleri, yayınlanan iki kez kareleri alınmış taşıyıcı sinyalin genlikleri ifade eder. D(t), uydu ile ilgili verileri ifade eder. $\xi_1$tterimi, L1 sinyalinin açısal frekansını ifade eder. ε terimi ise parazit ve osilatörde birikerek oluşan hataları ifade eder.

L2 taşıyıcısı ise (3.3)bağıntısında belirtildiği şekilde tanımlanır

$$L2(t)= C_t\, P(t)\, D(t)\, \cos(\xi_2\, t+\varepsilon) \quad (3.3)$$

Bağıntıda yer alan L2(t) değişkeni, zamana bağlı L2 taşıyıcısını ifade eder. $C_t$değişkeni, sinyalin genliğini ve $\xi_2$tterimi ise L2 sinyalinin açısal frekansını ifade eder.L1 ve L2 taşıyıcıları uydu tarafından yayınlanan, uydu mesajı veya navigasyon mesajını (uydu saati, uydu yörüngesi, uydunun performansı ve verilere getirilecek çeşitli düzeltmeler) taşımaktadır. Bu mesaj 50 Hz gibi alçak bir frekansla kullanıcıya ulaştırılır(Karaali ve Yıldırım, 1996).

Uydular yapı itibariyle üç farklı türü vardır. Bunlar BlockI, BlokII ve BlokII-R olarak adlandırılır.

b) Kontrol Bölümü: Bu uydular, konumları çok iyi bilinen 5 adet izleme (Ana kontrol ve izleme istasyonu (insanlı): Colorado Spring, yer kontrol ve izleme istasyonu (insansız):Hawai, Kwajalein, Ascension adaları, İzleme istasyonu (insansız) Diego Garcia)istasyonundan sürekli olarak kontrol edilir (Şekil 3.4). Bu izleme istasyonları ile uydular sürekli kontrol edilir ve olası sapmaların önüne geçilir.



Şekil 3.4. Uyduların konumlarını izleyen istasyonlar

c) Kullanıcı Bölümü: Alıcı setleri bu bölümü oluşturmaktadır. Alıcı anteni, alıcı ve güç kaynağı, bu setin kapsamında bulunan donanımlardır.

GPS Navigasyon Mesajı içerisinde yayınlanan uydu konum bilgilerine ilişkin doğruluğun düşük olması yada kasıtlı olarak yanlış yayınlanması durumunda, karşılaşılan hataya efemeris hatası denir (Tuşat E. ve Turgut B., 2004). Uydu yörüngelerinin daha duyarlı olabilmesi için efemeris hatası kullanılır. Uydu yörünge hatasının (uydu konum hatasının), ölçülen baz bileşenleri ile ilişkili olduğu (3.5) bağıntısı ile tanımlanabilmektedir.

$$\frac{\Delta b(m)}{b(km)} = \frac{\Delta r(m)}{\rho_R^{SV}(km)} \tag{3.5}$$

Bağıntıda yer alan Δb terimi ölçülen baz bileşenindeki hatayı, b ölçülen baz değerini, Δr yörünge hatasını (uydu konum hatasını) ve ρ_R^{SV} terimi ise uydu-alıcı uzaklığını ifade eder(Tuşat E. ve Turgut B., 2004).Baz değerlerine göre elde edilen çeşitli efemeris hataları (uydu koordinatlarındaki hataları), Çizelge 3.1'de gösterilmiştir.

Çizelge 3.1. Çeşitli baz değerlerine göre elde edilen efemeris hataları. Hesaplarda uydu-alıcı uzaklığı 20200km olarak alınmıştır(Kahveci ve Yıldız, 2001).

Efemeris Hatası (m)	Baz Uzunluğu (km)	Baz Hatası (ppm)	Baz Hatası (mm)
2.5	1	--	--
2.5	10	0.124	1
2.5	100	0.124	12
2.5	1000	0.124	124
0.05	1	--	--
0.05	10	--	--
0.05	100	0.0025	0.2
0.05	1000	0.0025	2.5

Genel olarak GPS doğruluğuna etki eden faktörler;

Kullanılan GPS ölçü tekniği

Çevre faktörleri

Görünen uydu sayısı

Uydu geometrisi

Sabit alıcıdan olan uzaklık

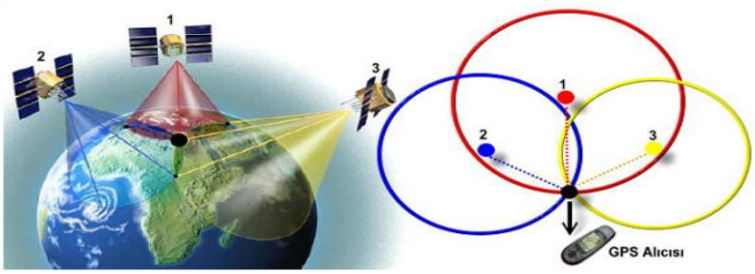
İyonosferik şartlar

GPS alıcısının kalitesi

Ölçü süresi

olarak sıralanabilir (Bean ve Ferguson, 2003).

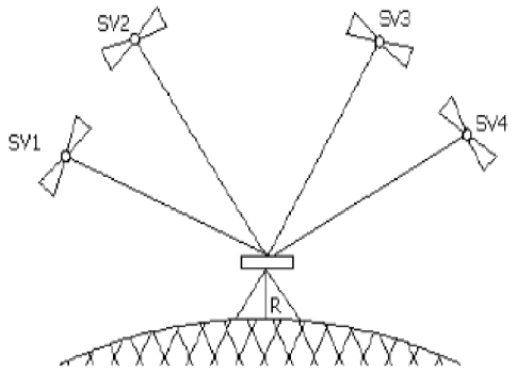
Konum Belirlemede Genel Yaklaşım:GPS teknolojisinde konum belirlemede kullanılan genel yaklaşım şekil 3.6'de gösterilmiştir. Şekilden görüdüğü üzere, her uydunun taradığı belirli bir dairesel alan söz konusudur. Alıcının gördüğü üç uydunun taradığı alanın kesişim noktası,alıcının konumunu belirler.



Şekil 3.6. GPS tekmolojisnde konum belirleme yaklaşımı (Sevindi, S., 2005)

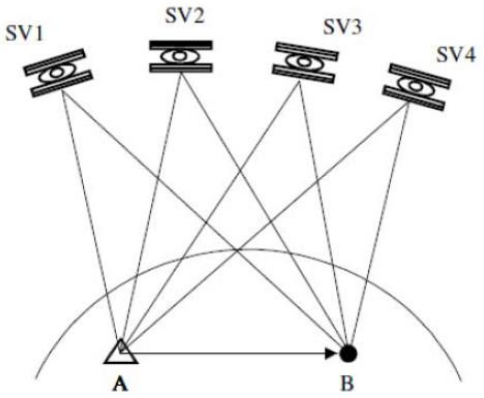
GPS ile Konum Belirleme Metodları: GPS alıcıları ile dünya üzerindeki 3 boyutlu koordinatlar (X,Y,Z), zaman ve kullanıcı aynı zamanda hareketli ise hızı da hesaplanabilmektedir. GPS ile konum belirlemede iki yöntem kullanılır. Bunlar; Mutlak ve göreli konum belirlemedir.

i) Mutlak Konum Belirleme: Mutlak konum belirlemede tek bir alıcı ile normal olarak dört yada daha fazla uydudan kod gözlemleri yapılarak, alıcının koordinatları (ϕ,λ,h veya X,Y,Z) belirlenir (Şekil 3.7). Yöntem, sinyalin uydu çıkışından alıcıya varışına kadar geçen zamanın ışık hızı ile çarpılarak hesaplanan uydu-alıcı uzaklıkları ve uyduların bilinen koordinatları ile uzaydageriden kestirme esasına dayanmaktadır.Alıcı koordinatları, kullanılan kod bilgisine (P kod, C/A kod) ve uydu geometrisine bağlı olarak anında ve mutlak anlamda belirlenebilmektedir. Bu yöntem alıcının(Nirengi, Poligon, Detay vb.) sabit olması durumunda statik, hareketli(Uçak, Gemi, Tank vb.) olması durumunda ise kinematik konum belirleme olarak tanımlanır (Küresel Konumlama Sistemi, 2010).



Şekil 3.7. GPS ile mutlak konum belirleme

ii) Göreli Konum Belirleme: Göreli konum belirlemede koordinatları bilinen bir noktaya göre diğer nokta yada noktaların koordinatlarının belirlenmesidir (Şekil 3.8). Diğer bir deyişle, göreli konum belirleme ile iki nokta arasındaki baz vektörübelirlenmektedir.Göreli konum belirleme ile elde edilen doğruluk mutlak konum belirlemeden çok daha iyi olup, alıcı tipi (P kodlu, P kodsuz), ölçü süresi gözlenen uydu geometrisi, uydu sayısı ve kullanılan efemeris bilgisine (yayın yada duyarlı) bağlı olarak elde edilen doğruluk 0.001 ile 100 ppm arasında değişmektedir. (Küresel.Konumlama-Sistemi,2010).



Şekil 3.8. GPS ile göreli konum belirleme

3.2. Önerilen Sistemin Donanımsal ve Yazılımsal Bileşenleri

Tez kapsamında tasarlanan GSM tabanlı çoklu takip sisteminindonanımsal ve yazılımsal bileşenleri bu bölümde anlatılmıştır.

3.2.1. Sistemin Donanımsal Bileşenleri

GSM tabanlı çoklu takip sistemi için tasarlanan sistemde bulunan temel donanımlar; GY-NEO6MV2, Arduino UNO geliştirme kartı ve SIM900R GSM modülüdür. Bu donanımların detayları aşağıda anlatılmıştır.

a) GPS (GY-NEO6MV2) Modülüile Konum Belirleme

Tez kapsamında GPS işlevini sağlamak için GY-NEO6MV2 donanımı kullanılmıştır. Teknik açıdan donanım özelliklerine bakıldığında;boyutunun küçük ve hafifliği nedeniyle her türlü hava aracında kullanılabildiği, UART arayüzü ile kolay kullanıma sahip olduğu, seramik anten içerdiği, yerleşik bellekbataryasının olduğu, güç kesilmelerinde veri kaybını önlemek için entegre EEPROM içerdiği, bağlantı kurulum durumu için led göstergesi ve değiştirilebilir arayüz hızının (varsayılan 9600kbps) olduğu görülmektedir. GY-NEO6MV2 donanımın dış kılıfı şekil-3.9'de gösterilmiştir.



Şekil 3.9. GY-NEO6MV2 donanımın 4 adet bağlantı pini vardır. 25x25mm ve 25x35mm boyutlarındadır. Modül anten içermektedir. (GY-NEO6MV2 datasheet)

b) SIMCom SIM900R Modülü ile Mobil Haberleşme (GSM)

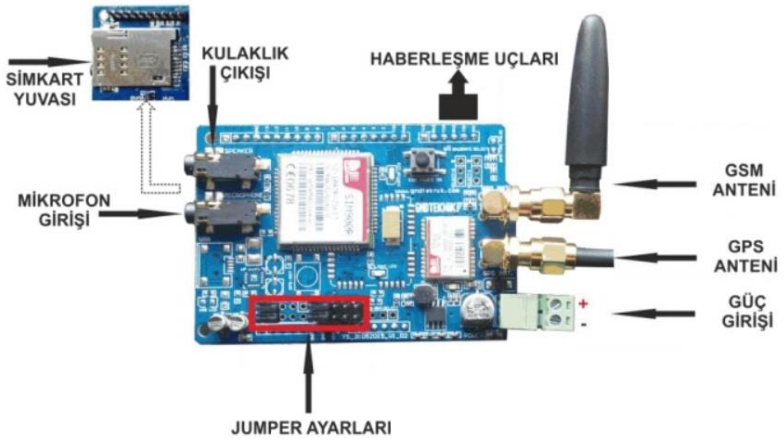
Tasarlanan sistemde GPS ünitesinden alınan konum bilgileri, GSM modülü üzerinden mobil telefona iletilir. Bu çalışmada kullanılan GSM modülünün temel özellikleri ve nasıl çalıştığı aşağıda açıklanmıştır.

Tez kapsamında GPS işlevini sağlamak için SIM900R modülü kullanılmıştır. Küresel pazarda kullanılmak üzere tasarlanan SIM900R, GSM genelindeki frekanslarda çalışan çift bantlı GSM/GPRS modülüdür. Donanımın teknik özellikleri: boyutları 24x24x3 mm, ağırlığı 3.4 gr, besleme gerilimi aralığı 3.2/4.8V, çalışma sıcaklığı -40°C ile +85 °C aralığındadır (SIM900R Hardware Design, 2017). Modülün bilinen diğer teknik özellikleri aşağıda listelenmiştir.

- Quad-Band 850 / 900/ 1800 / 1900 MHz
- GPRS çoklu-slot class 10/8
- GPRS mobilistasyon class B
- GSM phase 2/2+ için esnek
- Class 4 (2 W @ 850 / 900 MHz)
- Class 1 (1 W @ 1800 / 1900MHz)
- AT komutları üzerinden kontrol – Standard Komutlar: GSM 07.07 & 07.05 | İyileştirilmiş Komutlar: SIMCOM AT Komutları
- Kısa Mesaj Servisi – şebeke üzerinden küçük boyutlu verilerin iletimi için (ASCII veya ham hexadecimal formatında).

- Gömülü TCP/UDP stack – bir web sunucusuna very yüklenmesine olanak sağlar.
- RTC destekli.
- Seçilebilir seri port
- Mikrofon ve kulaklık girişi
- Düşük güç tüketimi – 1.5mA(uyku modunda)
- Ensdüstriyel sıcaklık aralığı -40°C to +85 °C

GSM modülüne ilişkin dış kılıfı, şekil 3.10'da gösterilmiştir.



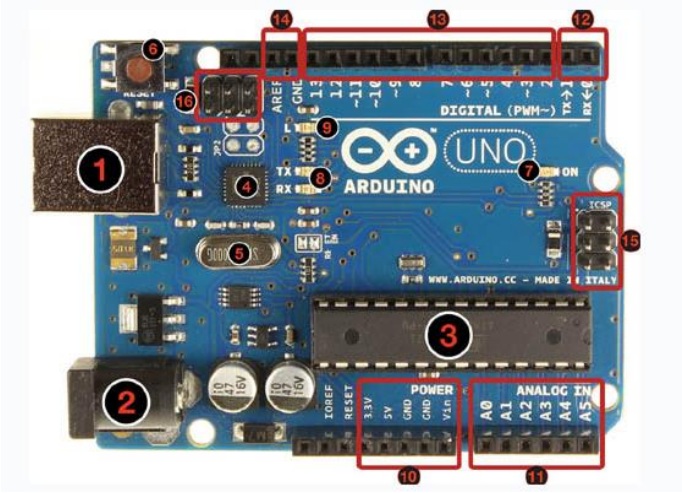
Şekil 3.10. SIM900R GSM modülün görüntüsü ve bağlantı noktaları.

- Modülün uygulama açısından kullanılabileceği alanlar:
- M2M (Machine 2 Machine) Uygulamaları
- Uzaktan veri ve cihaz kontrol uygulamaları
- Uzak mesafe meteoroloji ve diğer sensör uygulamaları
- SMS Gönderme
- SMS Alma
- Telefon Uygulamaları (Ses alış – veriş)
- Scada – Telemetri uygulamalar
- GGS01 – GSM/GPS modeli ile araç takip uygulamaları
- GGS01 – GSM/GPS modeli ile insan-nesne-cihaz takip uygulamalarıdır. Donanımın bu özelliğinden dolayı, bu çalışmada tercih sebebi olmuştur.

c) Arduino UNO Geliştirme Kartı

Arduino, Atmel'in AVR serisi mikrodenetleyicileri üzerine kurulmuş, prototip oluşturma aşamasını kolaylaştıran, ek genişlemeler için konnektörlere ve bu konnektörlere takılabilen shield adı verilen eklentilere sahip bir platformdur (Kayaalp ve Özkorucuklu,2015).Arduino; donanım özelliklerine göre Due, Uno, Mega, LillyPad, Esplora, Pro Mini, Mini, Nano, BT ve Fio çeşitleri bulunmaktadır.

Tezin kapsamında önerilen sistemin tüm kontrolü için Arduino UNO geliştirme kartı kullanılmıştır. Bu geliştirme kartının seçilmesindeki en önemli neden, Arduino ailesinde bulunan geliştirme kartlarına ait gerek donanım gerek ise yazılımların açık kaynaklı ve uygun fiyatlı olmasıdır. Arduino UNO, ATmega328 tabanlı birgeliştirmekartı olup,14 adet Dijital giriş / çıkışı (6 tanesi PWM çıkışını destekler) ve 6 adet Analog girişi bulunur (Şekil 3.11). Ayrıca 32KB Flash belleğe ve 16 MHZ hızında açık kaynak donanıma sahiptir.



Şekil 3.11. Arduino UNO geliştirme kartı

Şekilde görülen mikrodenetleyicisinin içerdiği üniteler numaralandırılmıştır. Numara sırasına göre kılıfta bulunan alt birimler aşağıda listelenmiştir (Arduino UNO REV3).

1. USB Soketi: Veri alış-verişini sağlar

2. Güç (Power) Soketi: Ünitenin beslemesini sağlar

3. Mikrodenetleyici: ATmega328 İşlemci

4. Haberleşme Entegresi: Diğer elektronik kartlar ile seri olarak haberleşmeyi sağlar

5. 16 MHz kristal: İşlemci hızı

6. Reset butonu: Sistemin baştan çalışmasını sağlar.

7. Güç (Power) Ledi: Ünite açık veya kapalıyken led yanar/söner

8. TX / RX Ledleri: Transfer (veri alış-veriş) işlemi sürecinde yanan/sönen ledler

9. Led

10. Güç (Power) Pinleri: Ek beslemenin sağlandığı pinler

11. Analog Girişler: Analog girişlerin sağlandığı pinlerdir. Bu girişler etiketlendirildiğinde, A harfi ile başlanır (Örneğin A0, A1 gibi). Bu pinler, sensörler ile iletişim kurulmasını sağlar.

12. TX / RX Pinleri: Transfer işlemi için veri alış-veriş pinleri

13. Dijital Giriş / Çıkış Pinleri: Dijital girişlerin ve çıkışların sağlandığı pinlerdir. ~ işareti görülen sayısal pinlerden analog sinyal alınabilir.

14. Analog referans pinidir. Arduino regülatörü 1023 adıma sahiptir. Örneğin 3 voltluk bir gerilim için 3V/1023 = 0.00293V hassasiyetine sahip gerilim adımları elde edilebilir.

15. ATmega328 için ICSP: Harici programlama pinleri

16.USB arayüzü için ICSP: Harici programlama pinleri

Kontrolöre, iki farklı şekilde güç sağlanabilir; i) USB üzerinden bilgisayar'a bağlanan kablo ile sağlanır, ii) 2 nolu pin olan Güç pini üzerinden sağlanır (Aduino/Genuino Uno, 2017). Arduino Uno'nun sahip olduğu 14 adet sayısal giriş ve çıkış pinlerinin tümü pinMode(), digitalWrite() ve digitalRead() işlevleri ile giriş/çıkışlar kontrol edilebilmektedir. Bu pinler 5 volt ile beslenir. Her pin en fazla 40mA akımın geçişine olanak sağlamakla birlikte, 20-50kohm aralığında bir direnç gösterir.

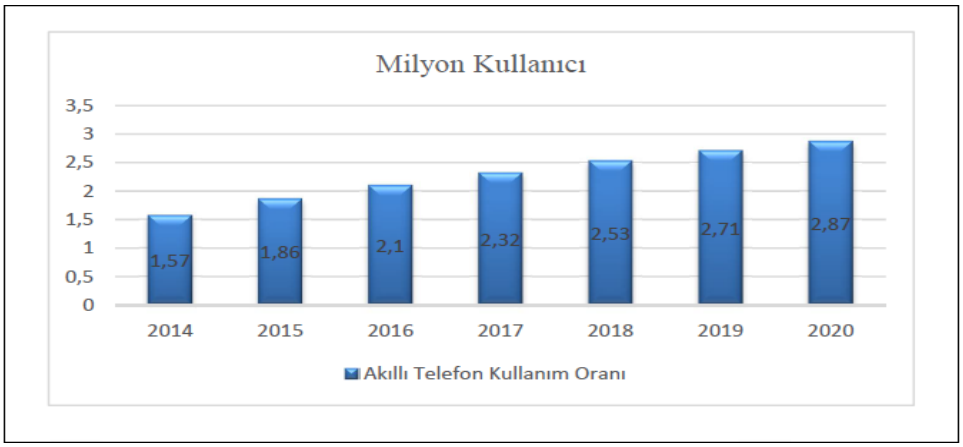
3.2.2. Sistemin Yazılımsal Bileşenleri

Bu tez kapsamında gerçeklenen GSM tabanlı çoklu takip sistemininyazılımsal aşamasında, iki faklı yazılım üzerinde çalışılmıştır. Çalışılan yazılım platformları; a)Android Studio Programı ile akıllı telefon uygulaması, b)Arduino yazılımı (IDE) ile sisteme ilişkin fonksiyonlarının belirlenmesi.

a) Android Studio Programı ile Akıllı Telefon Uygulamasının Geliştirilmesi

Android işletim sistemi, günümüz akıllı telefon sektörünün temelinde yer alan işletim sistemlerinden bir tanesidir. Akıllı telefon sektörü ilk olarak 90 yılların ortasında Apple şirketi üretimi olan PDA (*Personal Data Assistant* - Kişisel Veri Yardımcısı) ile başlasa da çok fazla ilgi görmeyerek yaygınlaşamamıştır.Fakat 1996 yılında Palm şirketinin üretmiş olduğu Palm Pilot isimli cihaz ile şimdi ki adı Smartphone olarak bilinen akıllı telefon sektörüne ilkadım atılmıştır (PDA, 2017). 90'lı yılların sonunda bir Microsoft üretimi olan Pocket PC (cep bilgisayarı) ile devam eden akıllı telefon macerası, 2007 yılının başında Apple firmasının ürettiği ilk dokunmatiktelefon olan IPhone'u, 2008 yılında ilk Android telefon HTC Dream (T-Mobile G1) takipetmitir.

Açık kaynak kodlu olmasıyla bilinen ve bu sebeple daha fazla kullanıcı ve geliştirici tarafından tercih edilen Google firmasına ait Android ve Microsoft firmasının bir parçası olan Windows Mobile akıllı telefon işletim sistemleri, Apple firmasına özgü işletim sistemi olan IOS karşılık daha fazla tercih edilmektedir. (Wilson , M. 2017).Şekil 3.3.1'de görüldüğü üzere International Data Corporation (IDC) şirketinin "*Worldwide Quarterly Mobile Phone*" adı altında dünya çapında yapmış olduğu anket sonuçlarıgöstermiştir ki; akıllı cep telefonu kullanıcı sayısı 2020 yılında 3Milyona ulaşması öngörülmektedir (Scarsella, A, Stofega W. 2017).Tümüyle gerçeklenen iyi uygulamalar ile akıllı cep telefonlarının gelişiminin sağlandığı görülebilmektedir.



Şekil 3.12. Anket Sonuçları (Scarsella, A, Stofega W. 2017)

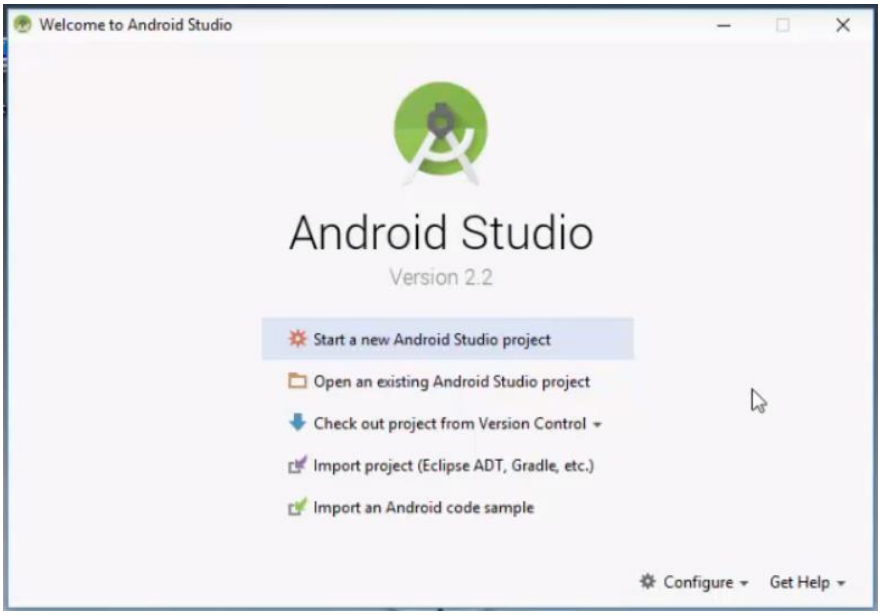
Android Studio, Android işletim sistemi için Google tarafından piyasaya sürülmüş bütünleşik gelişim ortamıdır. Uygulama geliştirmede yeni araçlar sağlamak ve şu anda yaygın olarak kullanılan IDE olan Eclipse'e bir alternatif oluşturmak için tasarlanmıştır.

Android Studio'da yeni bir proje başlatıldığında, projenin yapısı SDK dizini içinde tutulan tüm dosyalarla görülür.Platform, Gradle tabanlı yönetim sistemine dönüş süreci oluşturmak için daha fazla esneklik sunar. Android Studio, gerçek zamanda uygulamalara yapılan görsel değişikliklerin görülmesine olanak sağlar ve aynı anda farklı düzenlemelerle ve çözünürlüklerle farklı Android cihazlarda nasıl görüleceği gözlemlenir.

Android Studio'da kullanıcı arayüzüyle parçaların hareketi, sürükle & bırak adımları ile sağlanır. Platform, Google Cloud Messaging ile birlikte kurulur ve bu bulut yoluyla Android cihazlara sunucudan veri gönderilmesine olanak sağlanır. Android Studio'nun sağladığı diğer olanaklar; dayanıklı ve doğrudan geliştirme ortamı, diğer cihaz çeşitlerindeki performansını test edebilme, tüm Android programlarında bulunan yaygın öğeler için sihirbazların / taslakların yer alması ve uygulamalarıngeliştirilmesi için zengin özellikli editor (Arduino Studio, 2018).

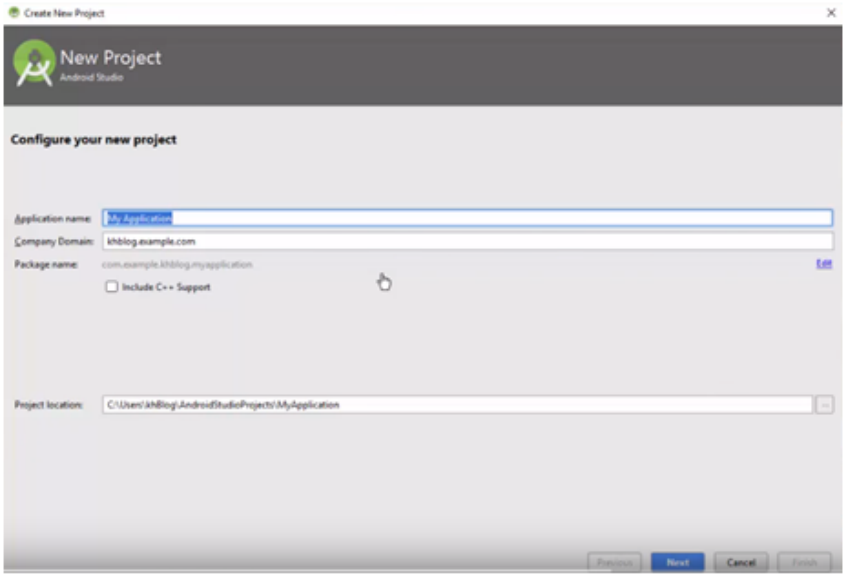
Android Studio'da basit bir uygulama gerçeklemek için izlenebilecek yol aşağıda açıklanmıştır.

1. Program ilk çalıştırıldığında karşımıza Şekil 3.13'de gösterilen pencere çıkar. Burada programa nasıl başlanacağına karar verilir. Örneğin önceden başlanmış bir proje açılabilir veya yeni bir proje ile başlanabilir.



Şekil 3.13. Android Studio platformunda başlangıç penceresi

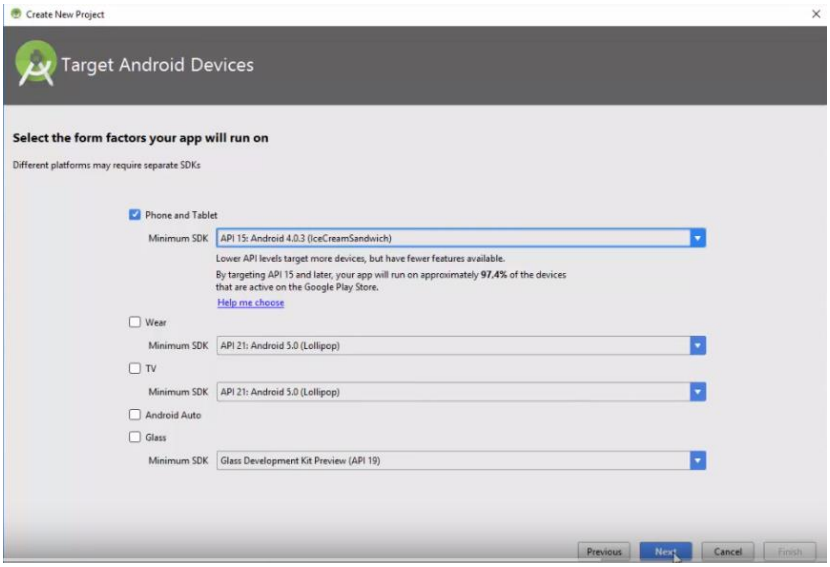
2. Bir sonraki aşamada projenin adı belirlenir (Şekil 3.14).



Şekil 3.14. Android Studio platformunda proje isminin belirlenmesi

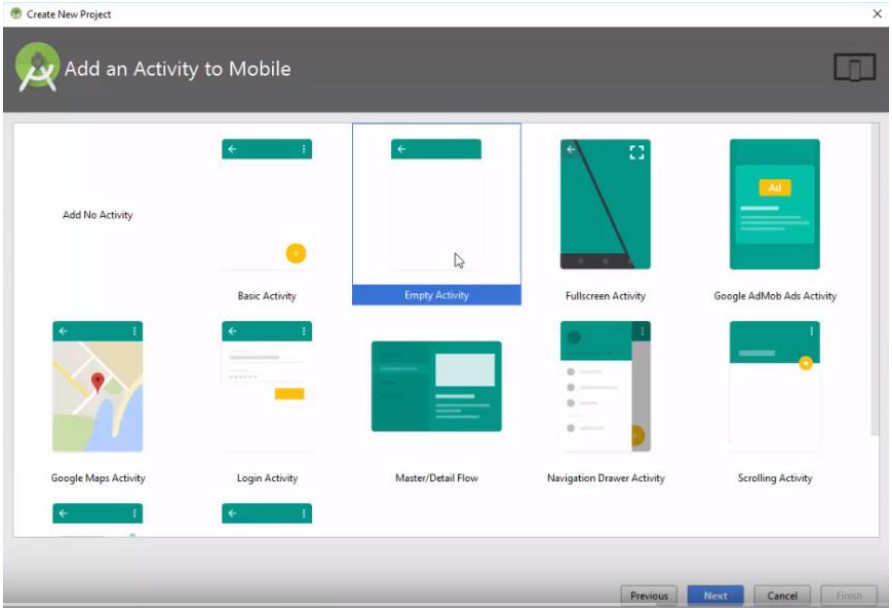
3. Ardından uygulamanın nasıl bir platformda çalışacağı belirlenir (Şekil 3.15). Seçilecek platform (API) yüksek bir versiyonda seçilir ise, uygulamanın daha düşük versiyonlarda çalışması mümkün olmayacaktır. Bundan dolayı burada seçimin özellikle

en temel özellikleri içeren düşük versiyonun seçilmesi, uygulamaların farklı platformlarda da çalışmasına olanak sağlayacaktır.



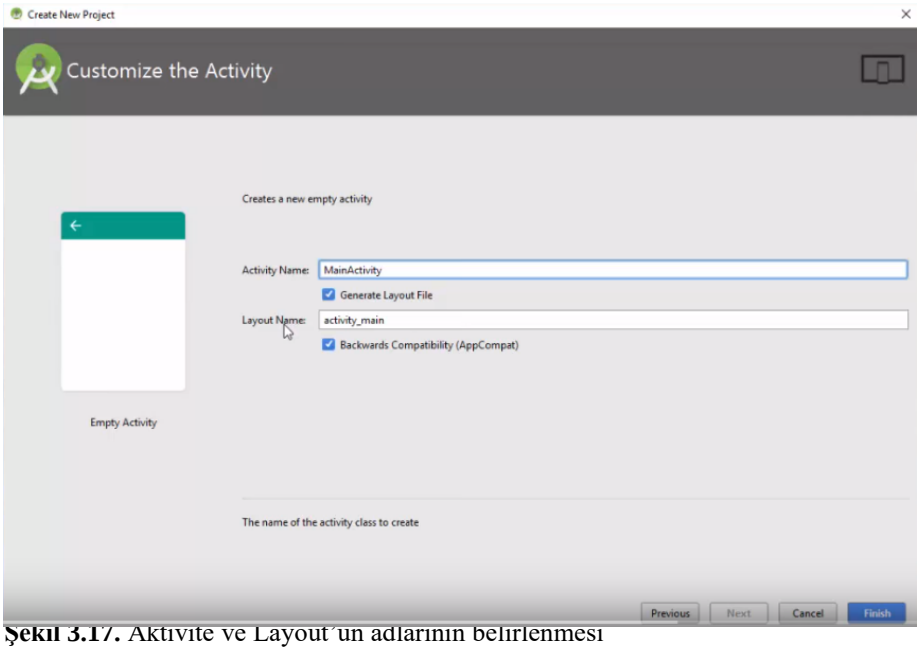
Şekil 3.15. Uygulamanın çalışacağı platformun (API) belirlenmesi

4. Ardından uygulamanın nasıl bir temaya sahip olacağı belirlenir (Şekil 3.16).



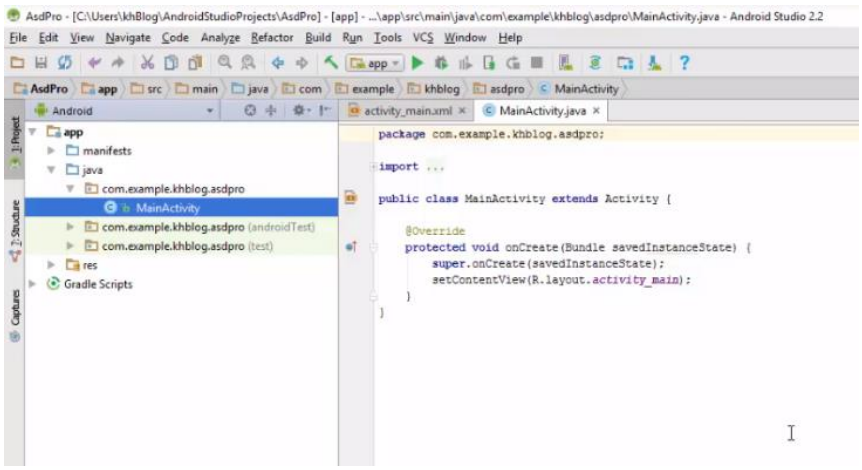
Şekil 3.16. Uygulamanın teması belirlenir

5. Tema seçildikten sonar oluşturulacak aktivite ve Layout'un adları belirlenir (Şekil 3.17).



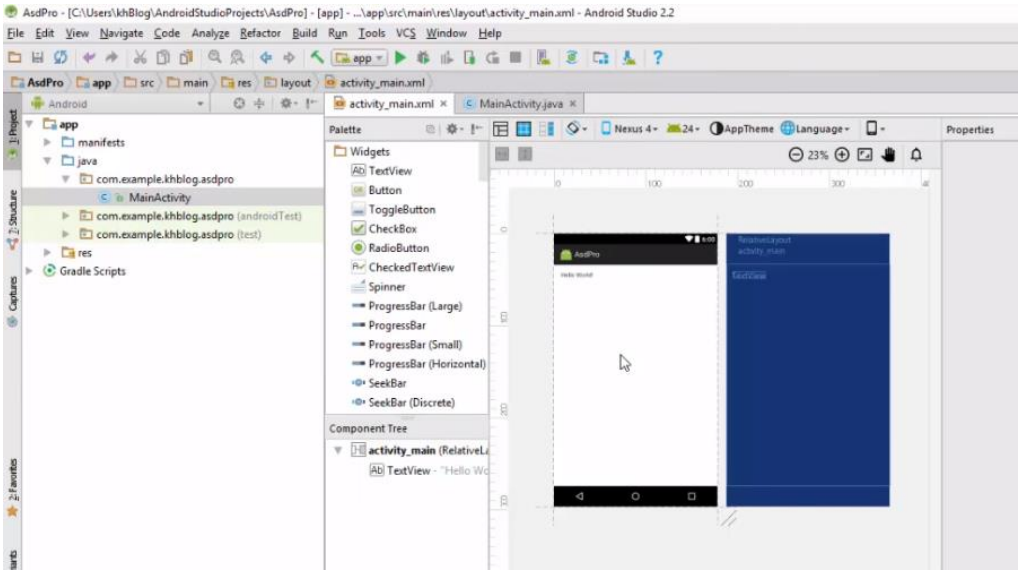
Şekil 3.17. Aktivite ve Layout'un adlarının belirlenmesi

6. Uygulama hazırlandıktan sonar ekranda oluşan pencere Şekil 3.18'de gösterilmiştir.



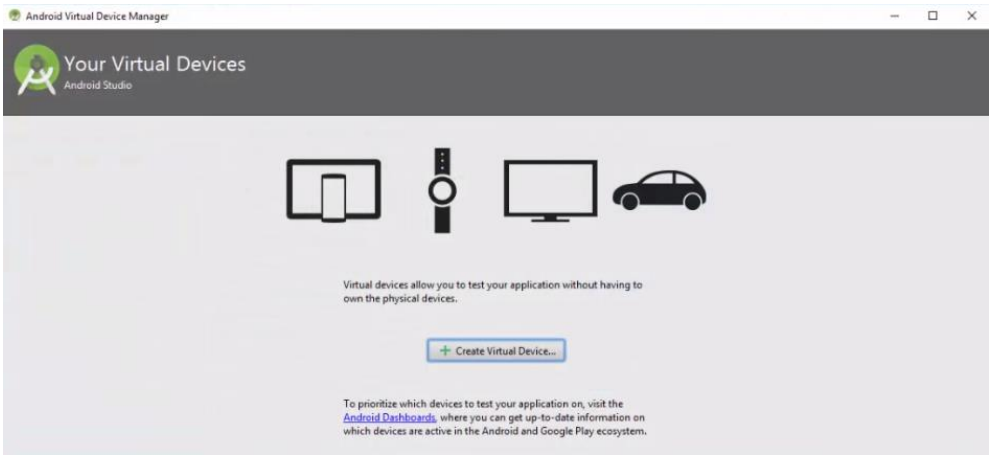
Şekil 3.18. Oluşturulan uygulamanın ana aktivite penceresi

7. Bir önceki aşamada oluşturulan ana aktivitenin ekran görüntüsü Şekil 3.19'de gösterilmiştir.



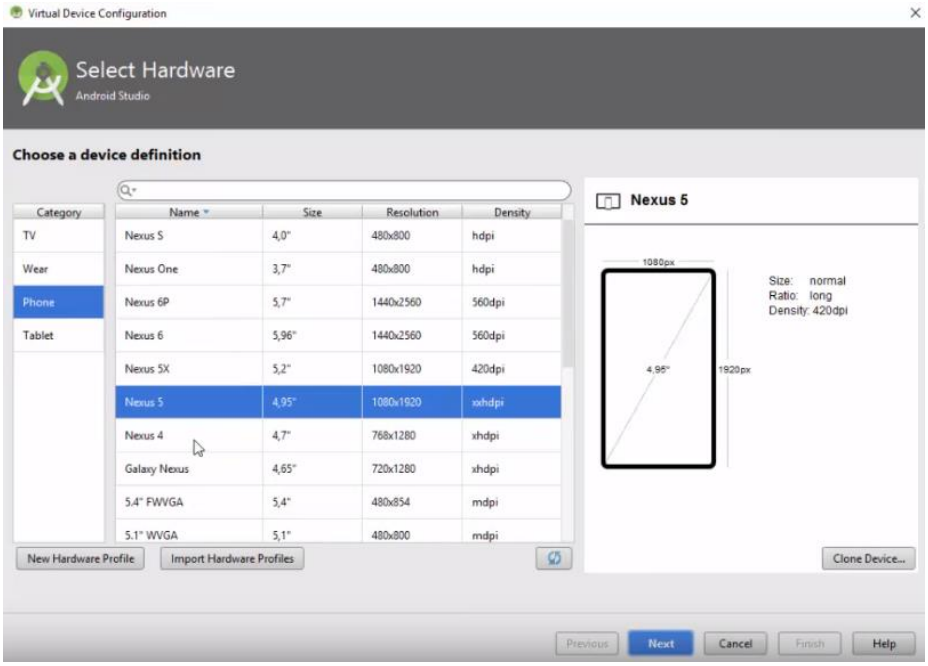
Şekil 3.19. Oluşturulan ana aktivitenin ekran görüntüsü

8. Bu aşamadan sonar bir simülatöre gereksinim vardır. Bunu için SDK içinde yer alan sanal makinalar kullanılır. Bundan dolayı, öncelikle bir sanal makinanın oluşturulması gerekmektedir. Bunun için menüye yer alan **AVD Manager** butonu tıklanır.(Şekil 3.20) ve oluştur butonuna tıklanır.



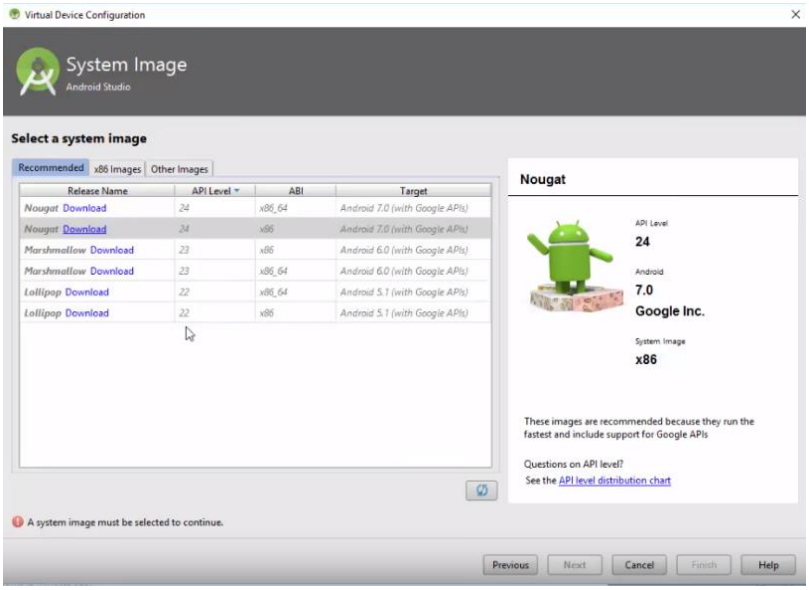
Şekil 3.20. AVD Manager ile sanal makina kurulum penceresi

9. Bu aşamada sanal cihazın nasıl bir API'ye sahip olacağı belirlenir (Şekil 3.21).



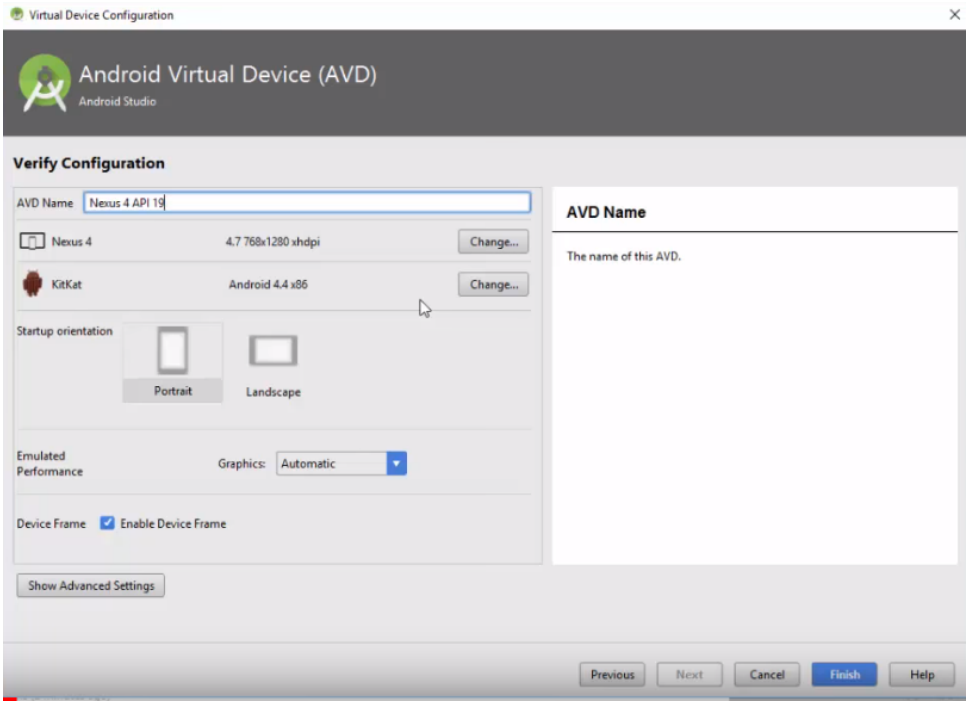
Şekil 3.21. Oluşturulan sanal cihazın API seçimi

10. Bu aşamadan sonar cihazın işletim sistemi seçilir (Şekil 3.22).



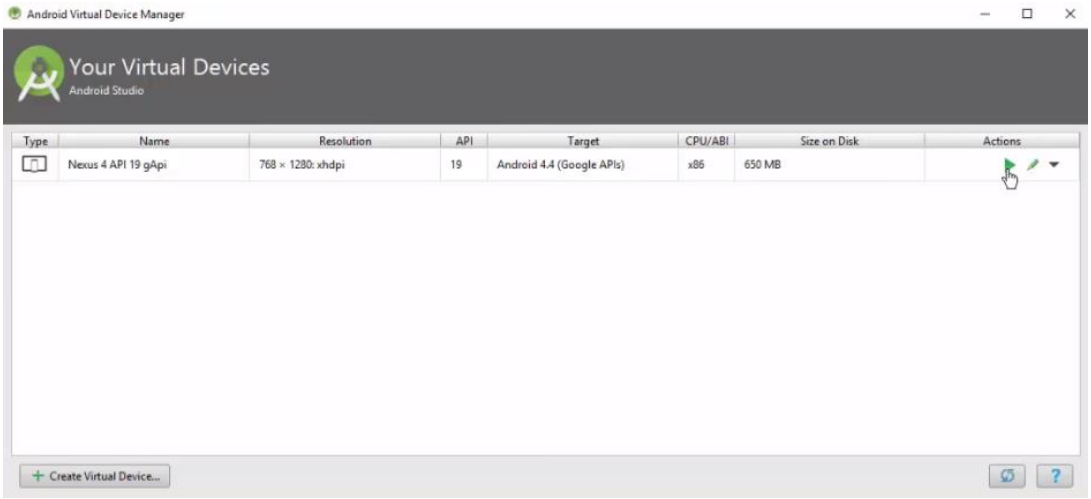
Şekil 3.22. Sanal cihaza ilişkin işletim sistemi belirlenir

11. Ardından cihazın konfigürasyon ayarları yapılır (Şekil 3.23).



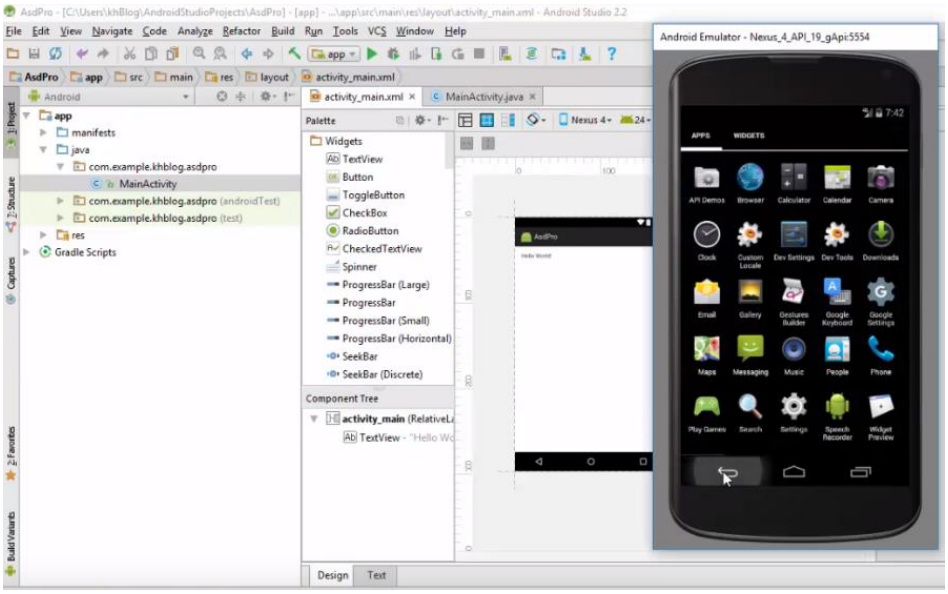
Şekil 3.23. Sanal cihaza ilişkin konfigürasyon ayarları belirlenir

12. Bir sonraki aşamada oluşturulan sanal cihaz Şekil 3.24'de gösterildiği şekilde elde edilmiş olunur. Play butonuna tıklanarak, sanal makina çalıştırılır.



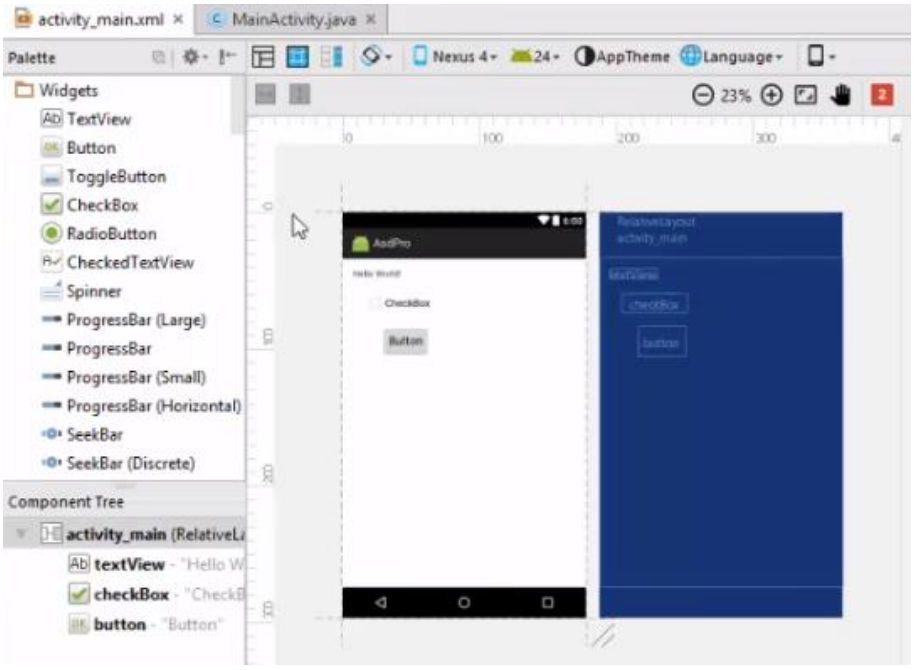
Şekil 3.24. Oluşturulan sanal cihaz penceresi

13. Çalıştırılan sanal cihazın ekran görüntüsü Şekil 3.25'de gösterilmiştir.



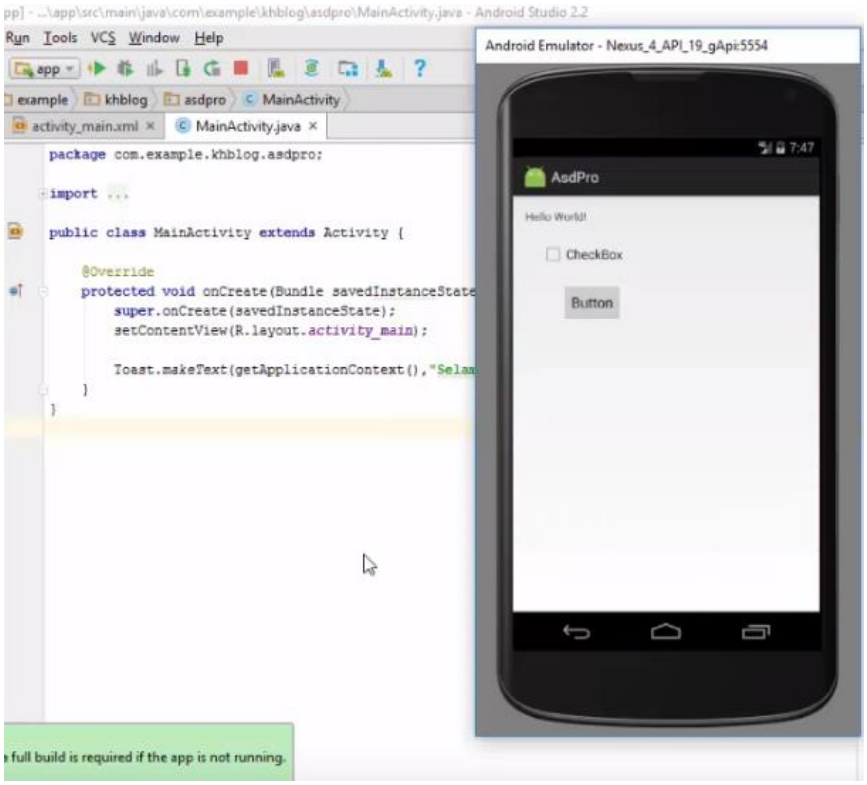
Şekil 3.25. Oluşturulan sanal cihazın görüntüsü.

14. Ardından uygulamaların sanal cihaz üzerinde nasıl çalıştığı incelenebilir. Basit bir örnek olması açısından ana aktivitenin ekranına bir Button ve bir de CheckBox eklenebilir (Şekil 3.26).



Şekil 3.26. Ana aktivitenin ekranına eklenen öğeler

15. Eklenen nesneler ile ilgili yapılması gerekenler ana aktivite sayfasında kodlanır. Eklenen neneler ile uygulamanın çalıştırılması durumunda, sanal cihazda görülecek olan ekran görüntüsü Şekil 3.27'de gösterildiği şekilde olacaktır.

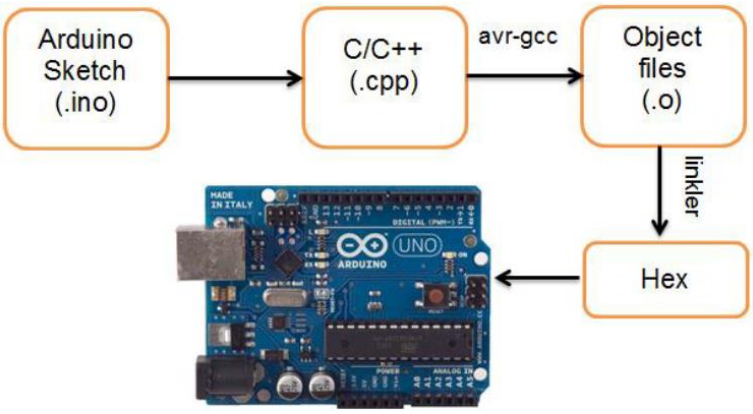


Şekil 3.27. Uygulamanın çalıştırılması ile elde edilen ekran görüntüsü

b) Arduino yazılımı (IDE) ile sisteme ilişkin fonksiyonlarının belirlenmesi

Arduino, elektronik sistemler oluşturmak için kullanılabilen açık kaynak kodlu bir geliştirme platformudur. Platform; temel olarak devre kartı, derleyici ve derlenen programları işlemciye aktaran araçlardan meydana gelir. Arduino, kolay programlanabilir ve hızlı uygulanabilir olmasından, dolayı elektronik dünyasında popüler hale gelmiştir. Arduino IDE platformunda, basitleştirilmiş C++ kullanılır ve yazılan programlar USB ile geliştirme kartına yüklenebilir (Karakuzu ve Gültekin, 2017).Şekil 3.28'dekodun dönüşüm sürecinde görüldüğü gibi Arduino içerisindeki bootloader programı ile programlanması için harici bir programa gerek duymamaktadır.

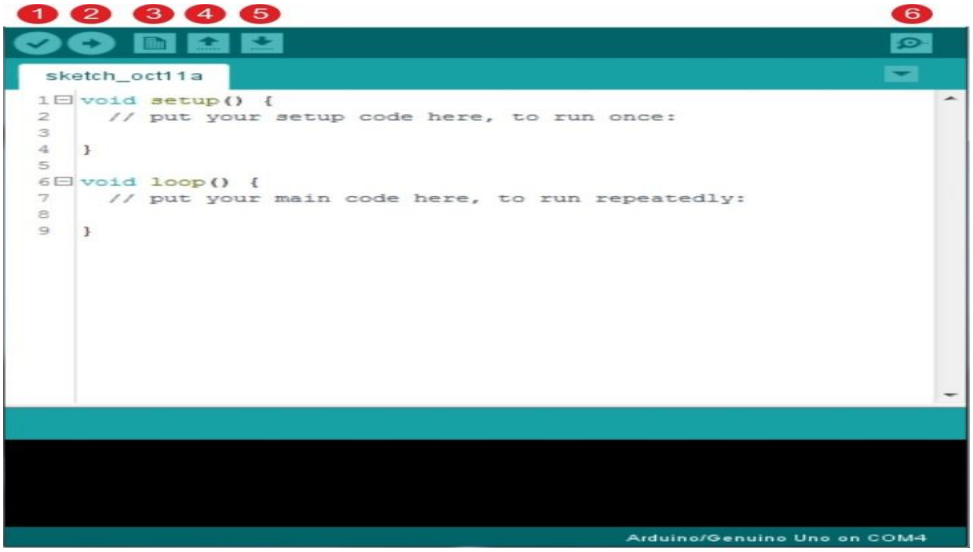
Java platformunda geliştirilen Arduino IDE Kod editörü, Wiring programlama dili ile C ve C++ tabanlı kütüphanelerini kontrol kartına yüklemekte kullanılmaktadır. (Süzen ve ark., 2017).



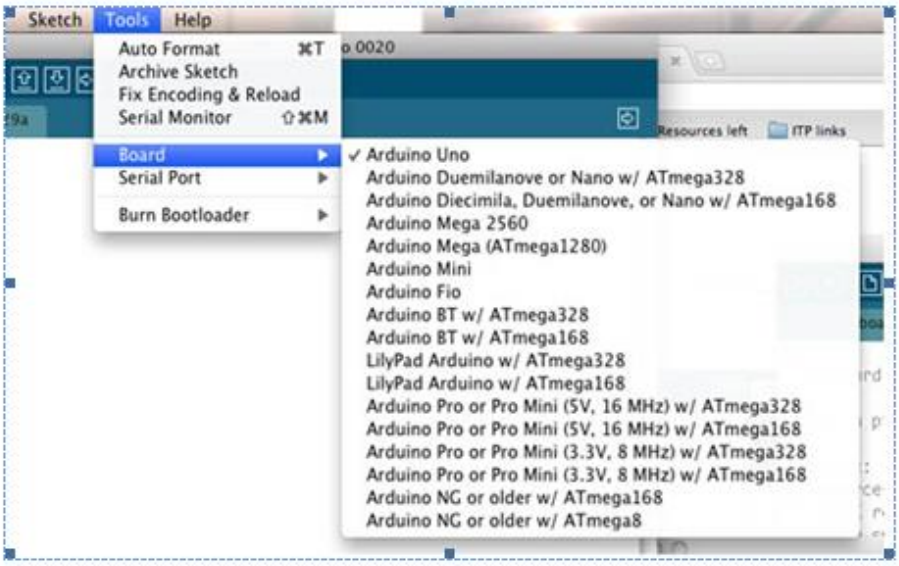
Şekil 3.28. Arduino'da kodun dönüşüm düzeneği (Süzen ve ark., 2017)

Arduino IDE kod editörü platformunun çalıştırılması ile görülen pencere şekil 3.25'de gösterilmiştir. Şekilde yapılan etiketlemelerin fonksiyonları: 1- Verify (Derleme): Yazılan kodların doğruluğunu kontrol etmek için kullanılır.2- Upload (Yükle): Yazılan kodları otomatik olarak derleyip Arduino'ya yüklemek için kullanılır.3- New (Yeni): Yeni Arduino IDE ekranı açmak için kullanılır.4- Open (Aç): Kayıtlı olan bir Arduino dosyasını açmak için kullanılır.5- Save (Kaydet): Yazılan veya üzerinde değişiklik yapılan kodları kaydetmek için kullanılır.6- Serial Port (Seri Port): Seri port ekranını açmak için kullanılır.

Ayrıca IDE kod editöründe kullanılan geliştirme kartı belirlenir.Örnek bir belirleme şekil 3.29'da gösterilmiştir.



Şekil 3.29. Arduino IDE kod editörü



Şekil 3.30. IDE kod editöründe Arduino geliştirme kartının belirlenmesi

3.3. Önerilen Sistem Tasarımı

Tez kapsamında GPS tabanlı çoklu takip sistemi tasarlanmıştır. Tasarlanan sistemin blok şeması şekil 3.31'de gösterilmiştir. Şekilden görüldüğü üzere, GPS uydularının yaydığı konum bilgileri yayınlanır. Bu bilgiler GPS modülü tarafından alınır ve Arduino geliştirme kartına aktarılır.

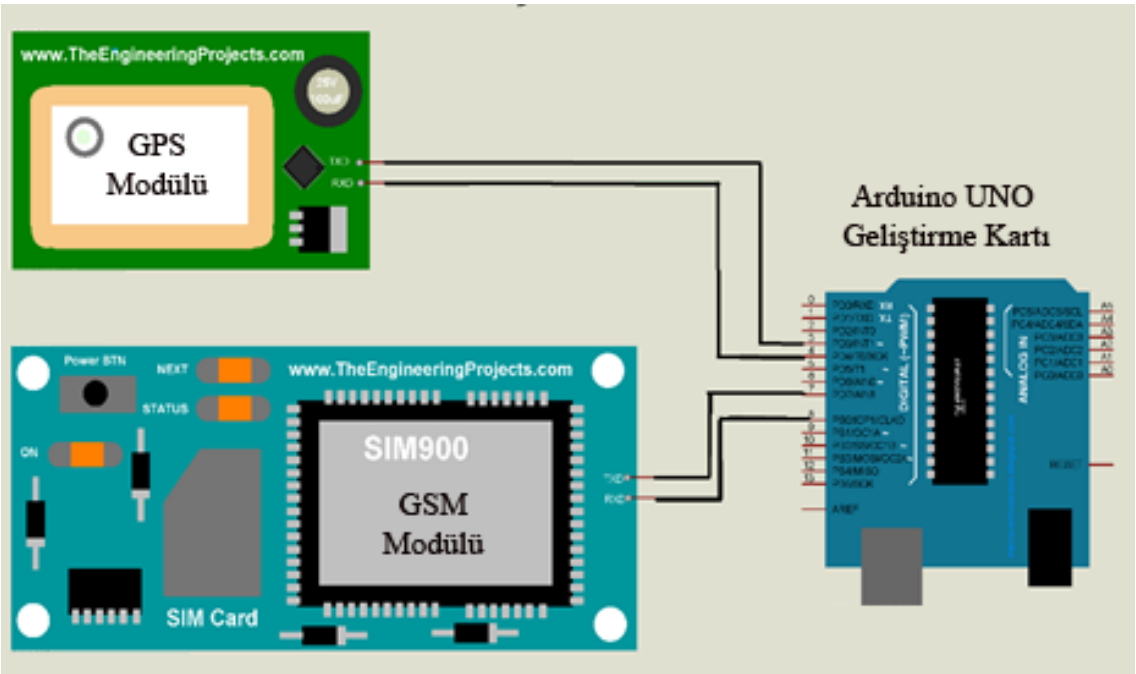
Geliştirilen GSM uygulamasında, talep edilen konum bilgisi, GSM modülü üzerinden mobil telefona Arduino UNO tarafından iletilir.



Şekil 3.31.Tasarlanan GSM tabanlı takip sistemi ve sistem bileşenleri

3.3.1. Gerçeklenen Tasarımın Donanım Yapısı

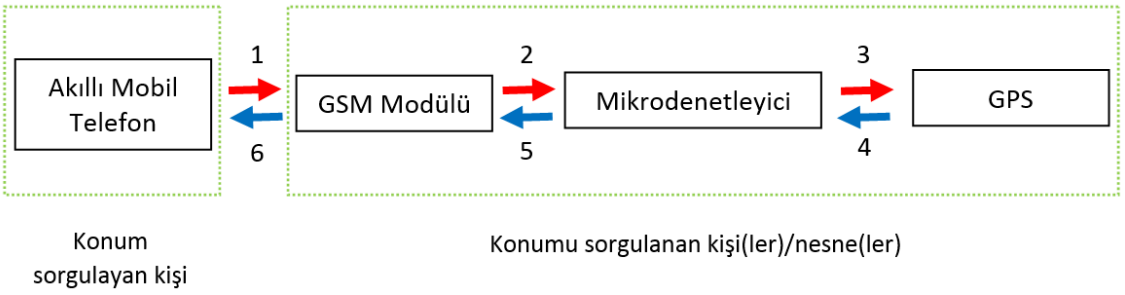
Bu tez kapsamında gerçeklenen devre tasarımı şekil 3.32'de gösterilmiştir. Şekilden görüleceği üzere, Arduino geliştirme kartı, GPS biriminden aldığı konum bilgilerini GSM modülü üzerinden akıllı cep telefona iletebilmektedir.



Şekil 3.32. Tasarlanan Sistemin Devre Şeması

Arduino Uno geliştirme kartı, sürekli olarak GSM ve GPS modülleri ile haberleşme halindedir. Donanımsal birimlerin kendi aralarındaki işaret akışları Şekil-3.29'da gösterilmiştir. Şekilden görüldüğü üzere, mobil telefondan gelen konum talebi (1 nolu ok), GSM modülü üzerinden geliştirme kartına ulaştırılır (2 nolu ok). Geliştirme kartı, GPS birimine konum bilgisini sorar (3 nolu ok). GPS birimi, gelen talep üzerine konum bilgisini belirler ve geliştirme kartına iletir (4 nolu ok). Geliştirme kartı, konum bilgisini GSM modülüne iletir (5 nolu ok). GSM ise mobil telefona kişinin/nesnesinin konumunu iletir (6 nolu ok).

Akıştan görüldüğü üzere, Arduino Uno geliştirme kartı bir köprü görevi üstlendiği görülmektedir.



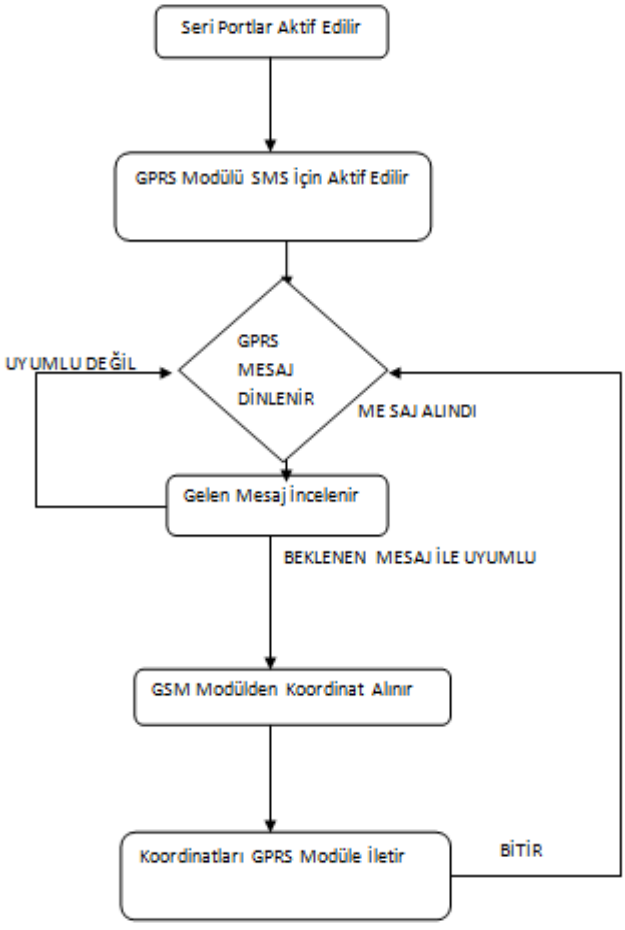
Şekil 3.33. Sistemdeki birimlerin kendi aralarındaki işaret akışı

3.3.2. Gerçeklenen Tasarımın Yazılımsal Uygulamaları

Tez kapsamında gerçeklenen sistem için iki farklı yazılım ve dolayısıyla uygulama gerçeklenmiştir. Konum talepleri ve haritada görüntülemek için Akıllı mobil telefon uygulaması tasarlanmıştır. İstemlerin cevaplanabilmesi için gerçeklenen donanımsal tasarım için arduino kodları yazılmıştır.

a) Tasarlanan Arduino IDE Kod Uygulaması

Arduino IDE kod uygulamasında tasarlanan sistem yazılımının işaret akış diyagramı şekil 3.30'da gösterilmiştir.



Şekil 3.34. Arduino geliştirme kartı için gerçeklenen işlem adımları

Şekil 3.34'da görülen akış diyagramına göre yazılan program kodu aşağıda açıklamalı olarak listelenmiştir.

```
#include <SoftwareSerial.h>
#include<AltSoftSerial.h>
#include <String.h>
#include <TinyGPS.h>
long lat,lon;              // create variable for latitude and longitude object   /enlem boylam
TinyGPS gps;               // create gps object
AltSoftSerial gpsSerial(8, 9);                  // gps sensörünün pinler
SoftwareSerial gprsSerial(4,3);                 // gprs molünün haberleşme pinleri
unsigned char buffer[64];                       // Gelen mesajın kaydedildiği dizi
int count=0;
void setup(){
        Serial.begin(19200);            // bilgisayar ile aruino arasındaki boud rate
        delay(500);                             // bekleme
        gpsSerial.begin(9600);                  // gps boud rate
        gprsSerial.begin(38400);                // gprs boud rate
        delay(5000);
        Serial.println("Sending Text...");      // bilgisayara mesaj yollama
        gprsSerial.print("AT+CMGF=1\r"); // modülün sms özelliğini aktifleştir.
        delay(100);
        gprsSerial.println("AT+CMGS = \"+905437489879\"");
        delay(100);
        gprsSerial.print("Hazırım ");           // telefona mesaj gelir
        delay(100);
        gprsSerial.print((char)26);             //the ASCII code of the ctrl+z is 26
                                                // (required according to the datasheet)
```

```
        delay(100);
        gprsSerial.println();
        Serial.println("Text Sent.");            // bilgisayara mesaj gelir
        delay(100);
        // Modülün gprs mesaj kısmı bazen açılamadığından, bu bölüm tekrarlanır.
        Serial.println("Sending Text...");
        gprsSerial.print("AT+CMGF=1\r");  // Set the shield to SMS mode
        delay(100);
        gprsSerial.println("AT+CMGS = \"+905437489879\"");
        delay(100);
        gprsSerial.print((char)26);              //the ASCII code of the ctrl+z is 26
                                                 // (required according to the datasheet)
        delay(100);
        gprsSerial.println();
        Serial.println("Text Sent.")
}
void loop(){
        while(gpsSerial.available()){                    // gps ten konum bekleme kısmı
                if(gps.encode(gpsSerial.read())){        // sinyal geldi
                        gps.get_position(&lat,&lon);
                        // gelen dataları enlem ve boylama kaydet

                        // display position
                        Serial.print("Position: ");
                        Serial.print(lat);
```

```
                Serial.print(",");
                Serial.print(lat/1000000);
                Serial.print(".");
                Serial.print(lat%1000000);
                Serial.print(" ");                  // print latitude to serialmonitor
                Serial.print(",");
                Serial.print(lon/100000000);
                Serial.print(".");
                Serial.println(lon%100000000);// print longitude to serialmonitor
        }   // if
} // while
if (Serial.available())                 // Seri verinin gelir ise
                switch(Serial.read()) {                 // karakter oku
                        case 't':                   // if the character is 't'
                        SendTextMessage();  // send the text message
                        break;
                        case 'd':                   // if the character is 'd'
                        DialVoiceCall();        // dial a number
                        break;
                }
                if (gprsSerial.available()){    // eğer gprs ten sinyal geliyorsa
                        while(gprsSerial.available()) {
                        // gprs ten sinyal gelene kadar oku
                                buffer[count++]=gprsSerial.read();
                                // buffer arrayine mesajı  yaz
```

```
                    if(count == 64)
                        break;
                }  // while
                count = 0;
                if (Serial.available())// Eğer seri portta veri hazır ise
                    gprsSerial.write(Serial.read());
                    // write it to the GPRS shield
                delay(1000);
                delay(1000);
                Serial.print( (uint8_t) buffer[52] );
                if (buffer[52]=='B'){
                // sms bilgisi B ise mesaj almak için modüle komut iletir
                    SendTextMessage();
                    buffer[52]=NULL;
                    }    // if
                }    // if
{
    //lat=lat/1000;
    // lon=lon/1000;
    // lat=lat*100000+lon;
    // String stringOne = "37,91640,236";
    Serial.println("SendingText...");
    gprsSerial.print("AT+CMGF=1\r");          // Set the shield to SMS mode
    delay(100);
    gprsSerial.println("AT+CMGS = \"+905437489879\"");
```

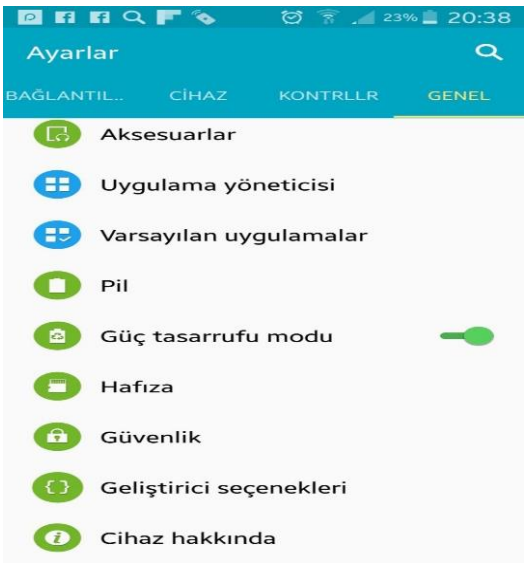
```
        delay(100);
        //gprsSerial.print(lon);                    //the content of the message
        gprsSerial.print("lat");
        gprsSerial.print(lat);                      //gprsSerial.print(".");
        gprsSerial.print((lat%1000000)/100);        // print latitude to serialmonitor
        gprsSerial.print("lon");
        gprsSerial.print(lon);    //gprsSerial.print(".");
        gprsSerial.println((lon%1000000)/100);// print longitude to serialmonitor
        delay(100);
        gprsSerial.print((char)26);
        //the ASCII code of the ctrl+z is 26 (required according to the datasheet)
        delay(100);
        gprsSerial.println();
        Serial.println("Text Sent.");
}
void DialVoiceCall()
{
        gprsSerial.println("ATD+xxxxxxx;");
        //dial the number, must include country code
        delay(100)
gprsSerial.println()
```

b) Android Studio Programı ile Tasarlanan Uygulama

Tasarlanan arayüz ve java kodları olmak üzere bu bölüm iki kısımdan oluşmaktadır. Ancak öncelikle Android tabanlı telefonlarda uyguluma geliştirme için gerekli ayarların yapılması gerekmektedir. Yapılan uygulama ise Samsung Galaxy

platformu üzerinden gerçeklenmiş olup, izlenilen yol aşağıda listelenmiştir. Ardından söz konusu yazılım aşamaları anlatılmıştır.

Öncelikle telefondan "ayarlar" menüsüne girilirve "genel ayarlar" menüsü seçilir. Ardından "cihaz hakkında" opsiyonu seçilir (Şekil 3.35).



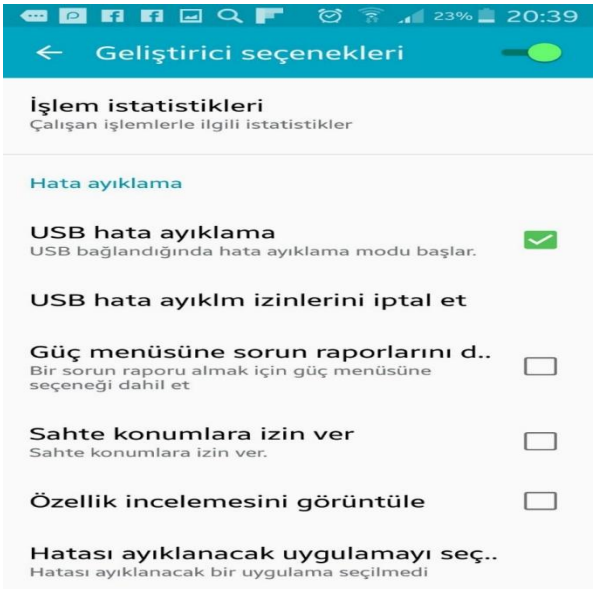
Şekil 3.35. Samsung Galaxy Cihazında "Cihaz hakkında"opsiyonu

Açılan pencerede "Yapım numarası" kısmı 7 kez tıklanarak, telefonda geliştirici mod aktif edilmiş olunur (Şekil 3.36).



Şekil 3.36. Samsung Galaxy Cihazında "Yapım numarası"bölümü

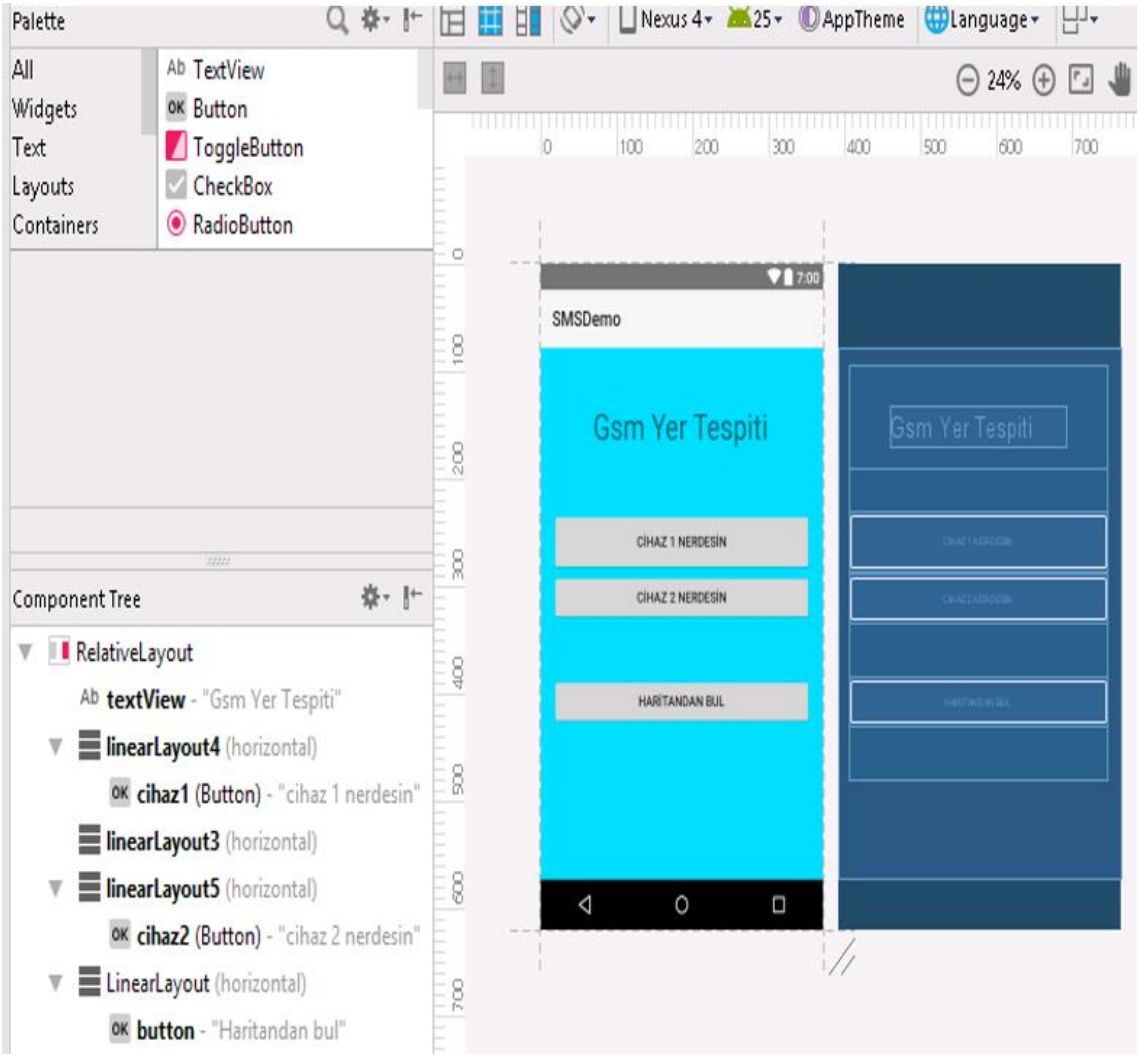
Ardından bir önceki menüye geçilir ve “USB hata ayıklama”opsiyonu işaretlenir. Böylelikle cep telefonu Android studio için hazır hale getirilmiş olunur (Şekil 3.37).



Şekil 3.37. Samsung Galaxy Cihazında “USB hata ayıkla”opsiyonu

Tasarlanan Arayüz

Öncelikle mobil telefonda görsel olarak kullanıcı arayüzünün tasarımı yapılır. Tasarlanan arayüz Şekil 3.38’de gösterilmiştir. Bu kısımda görsel olarak butonların yerleşimi, adları, sayfa tasarımı ve bu butonların adres bilgileri düzenlenir. Her butonun bir adres bilgisi belirlenir ve Properties kısmından söz konusu ayarlar yapılabilmektedir. Bu adres bilgisi ile Java Class bölümünde butonların işlevleri belirlenir.



Şekil 3.38. Mobil uygulama için tasarlanan bilgisayar arayüzü

Şekil 3.39'de görüldüğü üzere mobil uygulamaya ilişkinarayüzde, farklı işlevlere sahip butonlar yerleştirilmiştir. Uygulama örneği için İki nesne/kişinin takibi sağlanmıştır. Bunun için arayüze iki buton konmuştur. Takip edilmek istenen kişi/nesne sayısı artar ise buton sayısı artırılabilir. Şekilden görüldüğü üzere, birinci nesne/kişi için "CİHAZ1 NERDESİN" etiketli bir buton ve ikinci nesne/kişi için "CİHAZ2 NERDESİN" etiketli bir buton konmuştur. Bu butonlardan hangisine tıklanır ise mobil uygulama tarafından ilgili nesne/kişiden mesaj yolu ile konum bilgisi istenir. Bir sonraki adımda, nesne/kişi bu isteme karşılık konum bilgisini SMS mesajı ile mobil uygulamaya iletir. Son olarak kullanıcı, mesaj ile gelen konum bilgisini haritada görmek üzere, arayüzde bulunan "HARİTADAN BUL" butonuna tıklar.

Arduino Studio ile tasarlanan arayüz, öncelikle sanal ortamda oluşturulmuş ve ardından derlenmiştir. Derlendikten sonra elde edilen sanal uygulama görüntüsü, Şekil 3.39de gösterilmiştir. Bu aşamada sistemin çalışması test edildikten sonra, uygulamanın gerçekleştirildiği bilgisayara USB bağlanmış ve söz konusu uygulama gerçek mobil telefona aktarılmıştır.



Şekil 3.39. Mobil uygulama için tasarlanan arayüz.

Bu arayüzde önemli ayarlardan birtanesi, işletim sistemin seçilmesidir.Bu çalışmada sanal uyglama için Android 5 ve Nexus 4 API22 işletim sistemleri kullanılmıştır.

Uygulamada nesne/kişiden gelen mesaj google haritada görüntülenecek şekilde tasarlanmıştır. Android studio kapsamında bulunan google harita eklentisi ile tasarıma google maps kısmı eklenmiştir. Haritanın uygulamada kullanılmabilmesi için Google'ın sağladığı API kodunu maps kısmına eklenmesi gerekmektedir. Mobil telefon uygulamasında google map'in kullılabilmesi için aşağıdaki işlem adımları izlenir.

i. Aşağıdaki bağlantı adresine girilir.

https://console.developers.google.com/flows/enableapi?apiid=maps_android_backend&keyType=CLIENT_SIDE_ANDROID&r=B5:EC:F3:88:00:F3:4A:80:F0:72:35:2A:4A:CE:E7:B1:CD:B1:09:67%3Bcom.android.smsdemo

ii. Ardından girilen web sayfasında aşağıdaki kodlar yazılır.

B5:EC:F3:88:00:F3:4A:80:F0:72:35:2A:4A:CE:E7:B1:CD:B1:09:67

B5:EC:F3:88:00:F3:4A:80:F0:72:35:2A:4A:CE:E7:B1:CD:B1:09:67

iii. Son olarak üretilen kod alınır alınır ve Android studio programında tanımlanır. Tez çalışmasında kullanılan kod aşağıda belirtilmiştir.

AIzaSyCiubIjI8OMfjCsOGgP5VzWSQIipqjUJU0

Sözü edilen veri seti ile ilgili detaylı bilgiye aşağıda belirtilen web sayfası aracılığıyla ulaşılabilir.

https://developers.google.com/maps/documentation/android/start#get-key

Arduino Studio için Java kodları

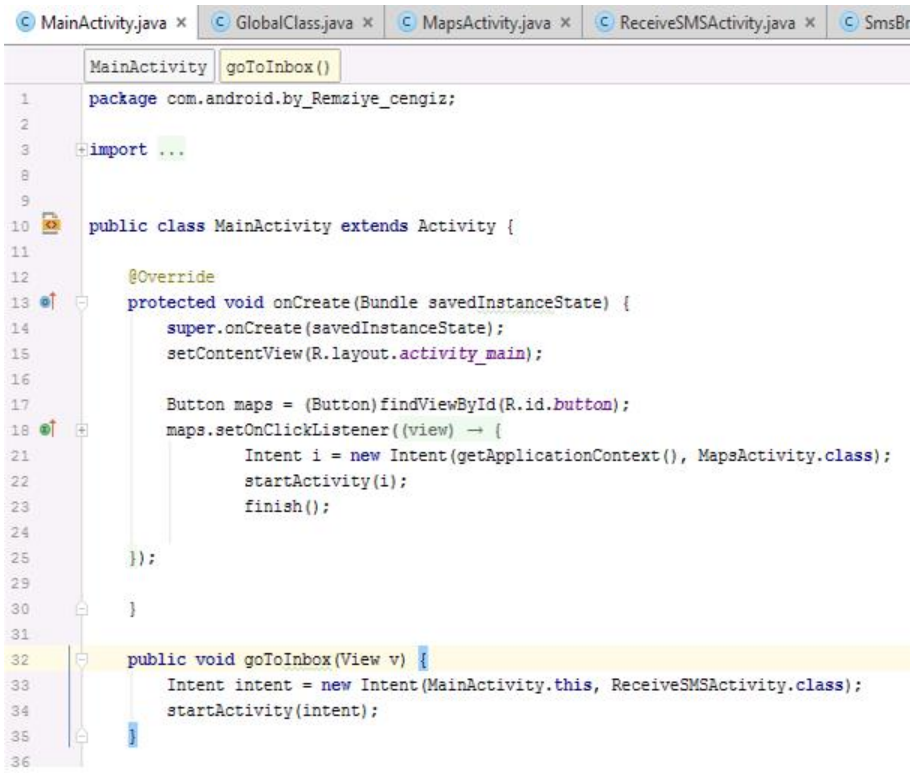
Arayüz aşaması tamamlandıktan sonra, arayüzde bulunan butonların işlevleri için Java kodları yazılır. GPS'ten gelen enlem(lat) ve boylam (lng) verileri, global değişken olarak tanımlanır. Dolayısıyla istenilen class da kullanılabilir yapıya getirilmiştir (Şekil 3.40).

GlobalClass.java × | MapsActivity.java × | ReceiveSMSActivity.java ×

```
1  package com.android.by_Remziye_cengiz;
2
3          import android.app.Application;
4
5  /**
6   * Created by Administrator on 28.9.2017.
7   */
8
9  public class GlobalClass extends Application {
10     public static double lat;
11     public static double lng;
12 }
13
```

Şekil 3.40. Konum bilgilerinin global olarak tanımlanması

Ana aktivite (MainActivity) bölümünde ana sayfada bulunan nesnelerin fonksiyonları tanımlanır. Tasarımda Bu sayfada bulunan butonların işlevleri kodlanır. Listelenir (Şekil 3.41).



Şekil 3.41. Butonlar için ana aktivitede yazılan kod

Enlem ve boylam değerlerini haritada nokta olarak görülmesi için MapsActivity bölümünde yazılan kod Şekil 3.42'de gösterilmiştir.

```
MapsActivity | onCreate()
1  package com.android.by_Remziye_cengiz;
2
3  import ...
14
15 public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {
16
17     Button button1;
18     private GoogleMap mMap;
19
20     @Override
21     protected void onCreate(Bundle savedInstanceState) {
22         super.onCreate(savedInstanceState);
23         setContentView(R.layout.activity_maps);
24         // Obtain the SupportMapFragment and get notified when the map is ready to be used.
25         SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
26                 .findFragmentById(R.id.location_map);
27         mapFragment.getMapAsync(this);
28
29         button1 = (Button) findViewById(R.id.button1);
30         button1.setOnClickListener((view) -> {
33                                     Intent i = new Intent(getApplicationContext(), MainActivity.class);
34                                     startActivity(i);
35                                     finish();
36
37         });
39
40     }
```

Şekil 3.42. Konumun haritada gösterilmesi için yazılan kod

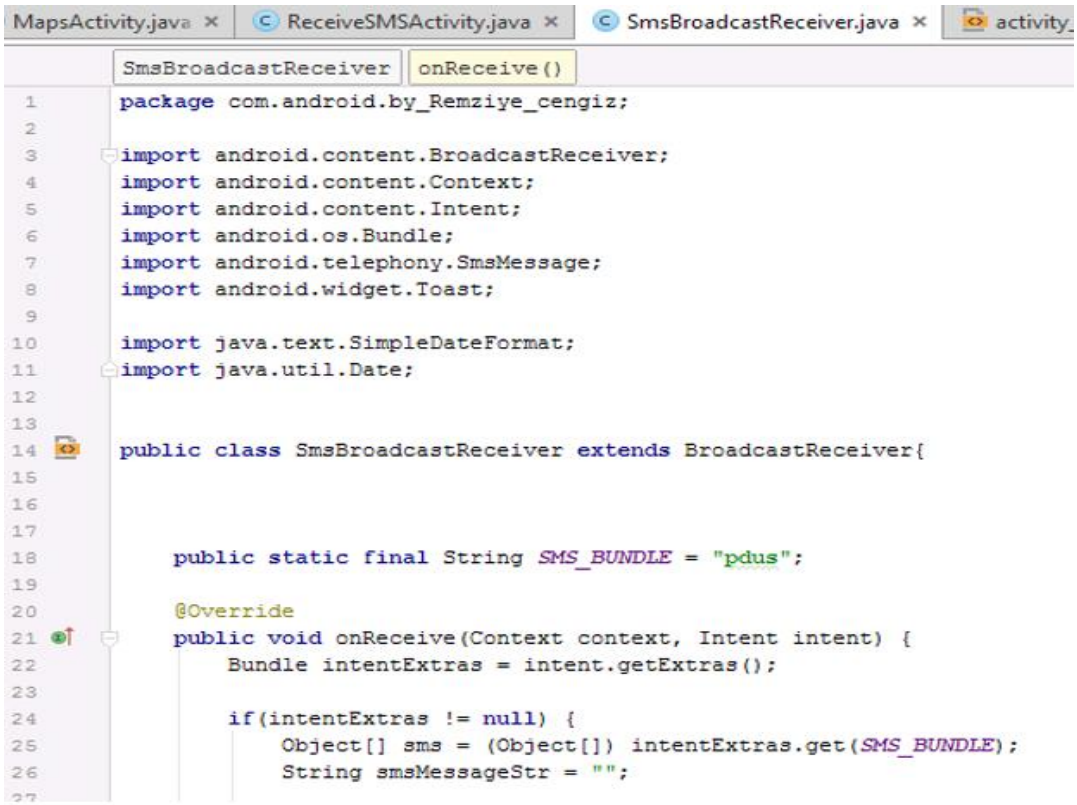
ReceiveSMSActivity bölümünde ise uygulamaya gelen SMS'ler dinlenir. Bu amaç için yazılan kodlar şekil 3.43'de gösterilmiştir

```
MainActivity.java × | GlobalClass.java × | MapsActivity.java × | ReceiveSMSActivity.java × | SmsBroadcastRece
ReceiveSMSActivity | onItemClick()
1  package com.android.by_Remziye_cengiz;
2
3  import android.app.Activity;
4  import android.content.ContentResolver;
5  import android.content.Intent;
6  import android.database.Cursor;
7  import android.net.Uri;
8  import android.os.Bundle;
9  import android.view.View;
10 import android.widget.AdapterView;
11 import android.widget.ArrayAdapter;
12 import android.widget.Button;
13 import android.widget.ListView;
14 import android.widget.Toast;
15
16 import java.text.SimpleDateFormat;
17 import java.util.ArrayList;
18 import java.util.Date;
19
20
21 public class ReceiveSMSActivity extends Activity implements AdapterView.OnItemClickListener{
22     Button button2;
23     private static ReceiveSMSActivity inst;
24     ArrayList<String> smsMessageList = new ArrayList<~>();
25     ListView smsListView;
26     ArrayAdapter arrayAdapter;
27
28     public static ReceiveSMSActivity instance() { return inst; }
```

Şekil 3.43. SMS lerin dinlenmesi için Receive SMS Activity bölümünde yazılan kod.

SmsBroadcastReceiver:Nesne/kişide bulunan sistemin mobil uygulamaya attığı SMS'ler diziye aktarılır. Dizideki veriler enlem ve boylam olarak global değişkene aktarılır.

Bu işlemler, SmsBroadcastReceiver bölümüne yazılır ve yazılan kod Şekil-3.44'de gösterilmiştir.



Şekil 3.44. Mobil uygulamaya gelen SMS'lerin kayda alınması

Tez kapsamında yazılan tüm kodlar Ek 1'de verilmiştir.

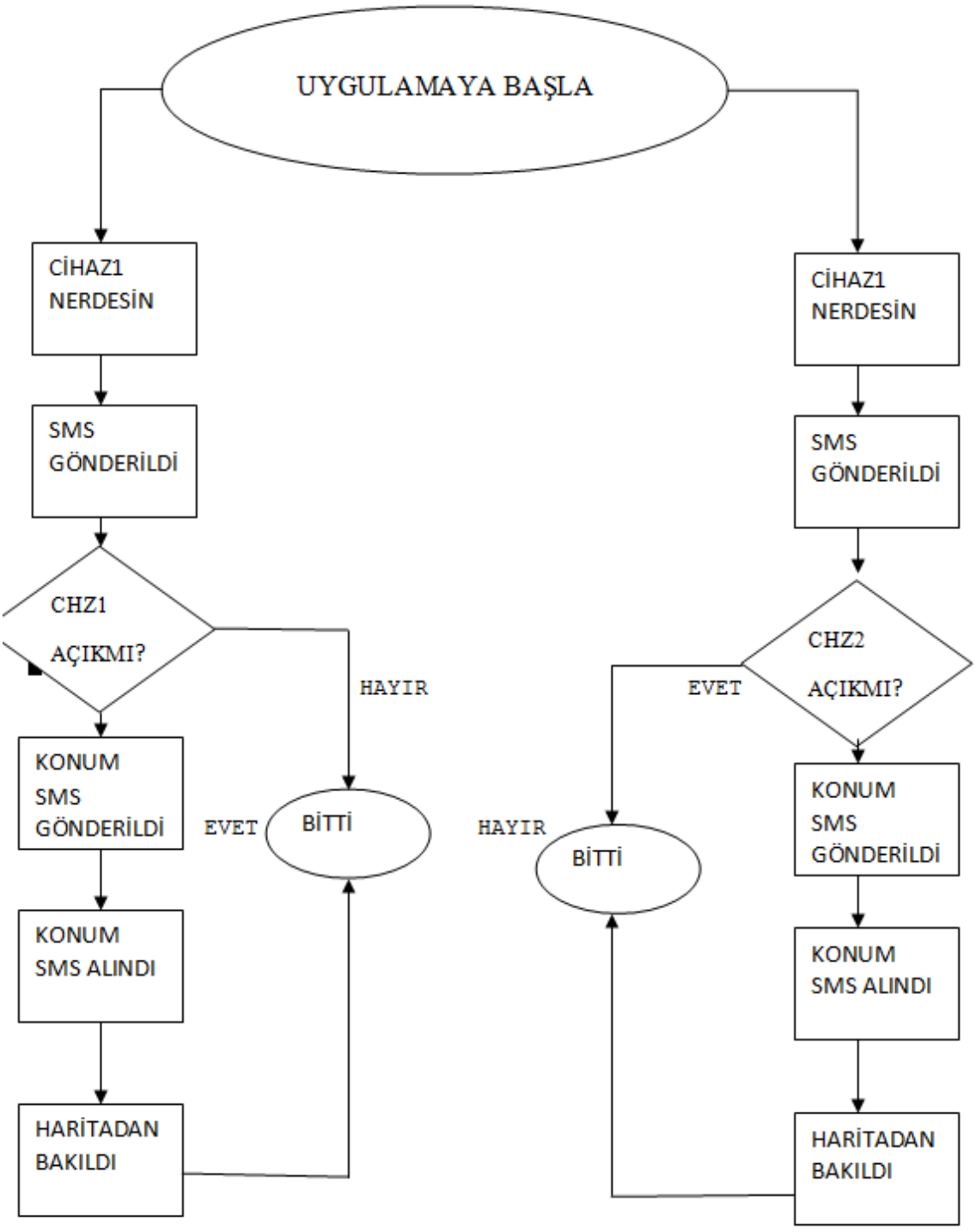
3.3.3. Gerçeklenen Tasarımın Genel Özellikleri

Tez kapsamında çalışılan GSM tabanlı çoklu takip sisteminde, iki kişinin/nesnenin konumunun mobil olarak belirlenmesi hem donanımsal hem de yazılımsal olarak başarıyla gerçeklenmiştir. Gerçeklenen sistemin aynı formda çoğaltılarak takip edilen kişi/nesne sayısı artırılabileceği unutulmamalıdır.

Buna göre Arduino Studio platformunda tasarlanan akış diyagramı, Şekil 3.36'da gösterilmiştir. Şekilde kişi/nesne yerine "Cihaz" terimi kullanılmıştır.Mobil uygulamada tanımlanan"Cihaz1 Nerdesin"ve "Cihaz2 Nerdesin"butonlarından

hangisine tıklanır ise konumu istenen sistem, uydudan aldığı konum bilgisini mobil uygulamaya iletir. Nesne/kişide bulunan sistem, sürekli olarak GPS üzerinden konum bilgilerini günceller. Sistemlerde bulunan GPS modülü, GPS uydulardan aldığı verilere göre konumunu günceller.

Akış diyagramında öncelikle mobil telefon uygulaması koşturulur ve üç butonlu uygulama önyüzü kullanıcıya yansıtılır. “Cihaz1 Nerdesin” butonuna tıklandığını varsayarsak, bu durumda cep telefonundan arduino tabanlı sisteme mesaj iletilir. Arduino tabanlı sistem, GPS’den aldığı konum bilgisini GSM birimi üzerinden cep telefonuna mesaj ile iletir. Bu mesaj, cihazın enlem ve boylam bilgilerini içerir. Ardından, mobil uygulamada haritada bul butonuna tıklanması durumunda, cihazın nerede olduğu tespit edilir ve haritada konumu kullanıcıya gösterilir. Enlem ve/veya boylam mesajı alınamaz ise konumu tespit edilemez ve program sonlanır.



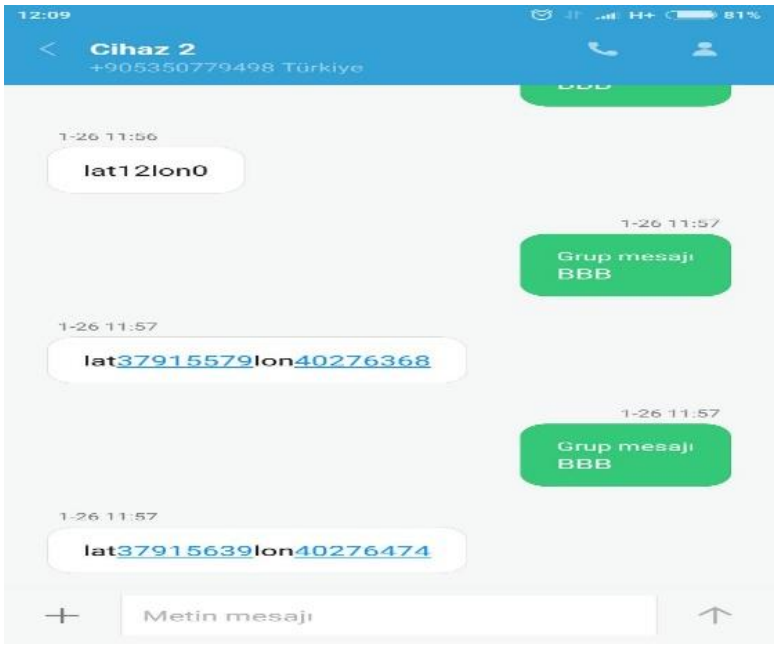
Şekil 3.45. Arduino Studio platformunda hazırlanan akış diyagramı

Bir örnek uygulamanın çalıştırılması ile elde edilebilecek cevaplar ve oluşan pencereler aşağıda listelenmiştir. Burada uygulamanın adı SMSDemo olarak kayda alınmıştır. SMSDemo uygulaması koştuurlur ve açılan pencere Şekil 3.46'de gösterilmiştir.



Şekil 3.46. SMSDemo Uygulaması

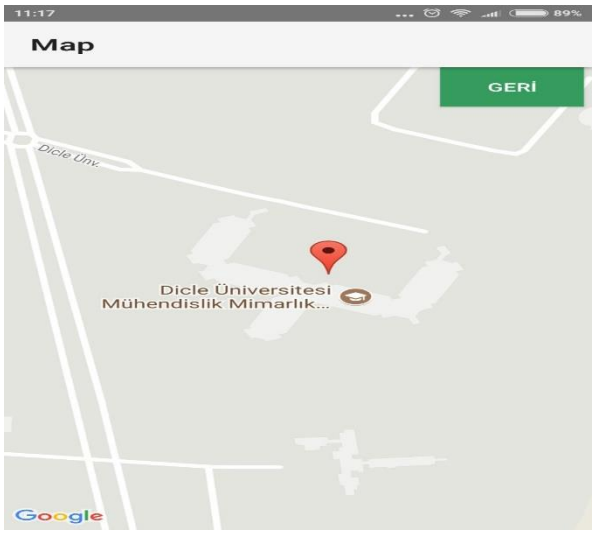
Cihaz1 Nerdesin Butona tıklanarak, önce aktif edilir.Burada takip edilen kişi veya nesneye SMS ile "BBB" mesajı iletilir. Cihaz1 açık ise mesaj başarılı bir şekilde iletilir.Değil ise mesaj olarak herhangi bir bilgi alınamaz ve işlem sonlanır. Aktif olan Cihaz1, enlem ve boylam bilgisinimobil uygulamaya SMS olarak gönderir. Bu bilginin alındığına ilişkin örnek bir uygulama şekil 3.47'de gösterilmiştir.



Şekil 3.47. Mobil uygulamaya gelen SMS örneği

Uygulama gelen SMS'lerden kişi/nesnenin enlem ve boylam verilerini alırve Google haritada nesnenin konumu belirler. Bunun için SMS alındıktan sonra "Haritadan

Bul”butonuna tıklanması gerekmektedir. Örnek uygulama şekil Şekil 3.48’de gösterilmiştir.



Şekil 3.48. Haritadan konumu bulunan örnek cihaz

Kapalı ortamlarda, GPS uydudan işaret alamayabilir. Bu durumda, enlem ve boylam değerleri sıfır (0) olarak kayda alınır. Bunun neticesinde konum yanlış belirlenir.

4. BULGULAR

GSM tabanlı çoklu takip sisteminde yapılan ölçümlere gore standart sapmalar mevcuttur. Bu standart sapmalar üç farklı şekilde farklılık göstermektedir.Bunlar google harita,anten boyu ve uydunun konumu olarak gözlemlenmiştir.200 den fazla konum istenen bu projede her seferinde farklı standart sapmalar gözlemlenmiştir. Gsm tabanlı çoklu takip sisteminde birbirinden bağımsız iki farklı cihazdan aldığımız verilerin standart sapmaları şu şekilde gözlemlenmiştir.Birinci cihazda toplam 133 konum bilgisi istenmiş olup, bu bilgilerin standart sapma ortalaması 21,3 metre olarak gözlemlenmiştir.İkinci cihazdan toplam 85 konum bilgisi istenmiş olup , bu bilgilerin standart sapma ortalaması ise 19,35 metre olarak gözlemlenmiştir.

5. SONUÇLAR

GPS tabanlı çoklu takip sisteminin hem donanımsal hem de yazılımsal olarak gerçeklenmesi, bu tezin amacını teşkil etmektedir. Donanımsal olarak üç farklı donanım kullanılmıştır. Bunlar; GPS modülü, Arduino Uno geliştirme kartı ve GSM modülüdür. Yazılımsal olarak kullanılan iki farklı platform kullanılmıştır. Mobil uygulama için Arduino Studio platform ve geliştirme kartı için Arduino IDE kod platformu kullanılmıştır. Başarıyla gerçeklenmiş olan çalışmada, sistemin oldukça yüksek performansta çalıştığı ve takip edilmek istenen kişi sayısının da artırılabileceği görülmüştür.

Literatürde bu konuda çok geniş çaplı çalışmaların olduğu görülmüştür. Bu çalışmalarda farklı amaçlar güdülmüş olsa da, düşünülen devre ve yazılım tasarımlarının aynı yapıda olduğu görülmüştür. Bu çalışma kapsamında literatürde konuyla ilişkili var olan çalışmalar detaylı olarak incelenmiştir ve piyasada var olan düzeneğin akademik açıdan incelenmesi sağlanmıştır.

Hızla gelişen teknolojinin elektronik ve otomasyon alanına yansıması ile geliştirme kartları çıkmıştır. Bu geliştirme kartları ile mikrodenetleyicili sistemler kolay ve düşük maliyetle tasarlanabilmektedir.

Bu çalışma ile düşük maliyetli bir takip sisteminin oluşturulabileceği görülmüştür.

6. KAYNAKLAR

Arduino/Genuino/Uno.http://www.robotiksistem.com/arduino_uno_ozellikleri.html, Erişim tarihi: 11-10-2017

Arduino Version 1.8.5 Erişim: https://www.arduino.cc/en/Main/Software

Arduino UNO REV3, https://store.arduino.cc/usa/arduino-uno-rev3, Erişim tarihi: 20.02.2018

Arduino Studio, (2018). https://android-studio.tr.uptodown.com/windows, Erişim tarihi: 22.02.2018

Bean, E.J. ,Ferguson, C.R. (2003). Effective Use of the ConnDOT GPS Base Station, JHR 03-289, Project 94-4, Central Connecticut State University.7.

Bayıroğlu, H.2013. Mobil Cihazlar Üzerine Bir Uygulama: Ebeveyn Çocuk Takip Programı.Yüksek İİsans, Fen bilimleri Enstitisü, Sakarya,1.

Chandra, A., Jain S., Abdul Qadeer, M., Implementation of Location Awareness and Sharing System Based on GPS and GPRS using J2ME, PHP and MYSQL, 2011 Computer Research and

Development (ICCRD), 2011 3rd International Conference on (Volume:1), 11-13 March 2011.

Denetimli Serbestlik Takip Sistemleri (Elektronik İzleme, Tutuklu ve Hükümlü Takip,GPSYazılımları,EKelepçe,Sistemleri),http://www.havelsan.com.tr/TR/Main/urun/771/denetimli-serbestlik-takip-sistemleri-elektronik-izleme-tutuklu-ve-hukumlu-takip--gps-yazilimlari-e-kelepce-sistemleri , Erişim tarihi:05-10-2017.

Erdal, M., Çancı , .M .Taşımacılık Yönetimi, UTİKAT, İstanbul, 2003.

Gül, F. (2010).Akıllı Ev Sistemleri. Yüksek Lisans Tezi, Gazi Üniversitesi,Fen Bilimleri Enstitüsü, Ankara.

GPS Notları, https://bilgibirikimimiz.tr.gg/GPS-NOTLARI.htm Erişim tarihi: 12-11-2017.

Güney, C., Avcı, Ö., Doğru, A.Ö. ve diğerleri. 2002. Filo Yönetim Sistemi Tasarımı.Selçuk Üniversitesi Jeodezi ve Fotogrametri Mühendisliği Öğretiminde 30.Yıl Sempozyumu, Konya, 550.

Gupta, R., Reddy, B., GPS and GPRS Based Cost Effective Human Tracking System Using Mobile Phones, Viewpoint, Volume 2, January-June 2011.

GYNEO6MV2datasheet,http://synacorp.my/v2/en/index.php?controller=attachment&id_attachment=630Erişim tarihi: 20.02.2018

GPS'in yapısı ve uygulamaları, http://www.elektrikport.com/teknik-kutuphane/gps-nedir-yaps-ve-uygulamalar/4276#ad-image-0, Erişim tarihi: 18.02.2018

https://developers.google.com/maps/documentation/android/start#get-keyDemiray,A.2017. Olası Dalış Kazalarına Erken Müdehale Etmek Amaçlı Adroid Tabanlı Uygulama Tasarımı. Yüksek lisans, Fen Bilimleri Enstitüsü ,İstanbul,1,2.

http://www.tuik.gov.tr/Start.do Erişim tarihi:17-11-2017.

Ismael, A. G., An Emergency System for Succoring Children Using Mobile GIS, International Journal of Advanced Computer Science and Applications, Volume 3, No. 9, 2012.

Küresel Konumlama Sistemi, (2010). https://www.konya.edu.tr/storage/files/department/insaatmuhendisligi/editor/DersSayfalari/Topografya/Bolum-7.pdf Erişim tarihi: 20.02.2018.

Küçük, E., Güneş, M. ve diğerleri (2007). GPS ile Araç Takip Sistemi. Lisans tezi, Mersin Üniversitesi Tarsus Teknik Eğitim Fakültesi, Tarsus.

Kahveci, M., Yıldız, F., (2001). Global Konum Belirleme Sistemi Teori-Uygulama, Nobel Yayın Dağıtım, s.184, Ankara

King, E.W., Masters, E.G., Rizos, C., Stoltz, A., Collins, C., (1987). Surveying wiht Global Positioning System, Dümmler, Bonn.

Karaali, C., ve Yıldırım, Ö., (1996). Global Konum Belirleme Sistemi (GPS), Pamukkale Üniv. Mühendislik Bilimleri Dergisi, 2(2) 103-108

Kayaalp, K. ve Özkorucuklu, S., (2015). Arduino Teknolojisi Kullanılarak Tarla için Radyasyon Kapı Kontrolü Tasarımı, SDU International Technologic Science, 7(2), 39-47

Karakuzu, C. ve Gültekin, S., (2017). İnternet Üzerinden Görsel Bilgisayar Arayüzü ile Denetlenebilen Nokta Matris LED Tabela. TBV Bilgisayar Bilimleri ve Mühendisliği Dergisi, Cilt 10, Sayı 2.

Liao, Y., Chuang, C., Jeng, J., Chen J., Systematic Design for the Global Positional Systems with Application in Intelligent Google Android phone, 2011. System Science and Engineering (ICSSE), 2011 International Conference, 8-10 June 2011.

Mobil Uygulama Nedir, https://www.hukumdar.com.tr/post/mobil-uygulama-nedir-Erişim tarihi:12-10-2017.Maps Android API, Erişim: 12-10-2017.

Özer, H. İ .(2005). Akıllı Ev Otomasyon Sistemi Tasarımı. Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya

Özgön, İ., Konak, Mesut. 2013. GPS Sinyalleri İle Konum Belirleme ve Raporlama İşlemi Yapmayı Sağlayan Programın Mobil ve Web Platformda Tasarımı. Lisan, Elektronik-Bilgisayar Eğitimi Anabilim Dalı, İstanbul, 2.

PDA, https://tr.wikipedia.org/wiki/Palm_PDA ,Erişim tarihi:12-10-2017.

RFID Teknolojisi , [http://gs1.tobb.org.tr/icerik_goster.php?Id=15] , Erişim Tarihi:11-10-2017.

Şanlı, M. (2009).Mobil hasta takip sistemi. Yüksek Lisans, Fen Bilimleri Enstitüsü,Kocaeli,22

Sevindi, C., (2005). Küresel Konum Belirleme Sistemi (GPS) ve CoğrafyaAraştırmalarında Kullanımı, Coğrafi Bilimler Dergisi, 3(1), 101-112

Süzen, A. A., Ceylan, O., Çetin, A., ve Ulusoy, A., (2017). Arduino Kontrollü Çizim Robotu, Mehmet Akif Ersoy Üniversitesi Fen Bilimleri Enstitüsü Dergisi Özel Sayı 1: 79-87

SIM900R Hardware Design, https://cdn.sos.sk/productdata/dc/4c/3bfd97e4/sim900r-te.pdf, Erişim tarihi:10-11-2017.

Sel,V.2013.Mobil İşletim Sistemleri,http://volkansel.com/mobil-isletim-sistemleri.html/, Erşim tarihi:11-10-2017.

Scarsella ,.A , Stofega.2017 International Data Corporation (IDC) ,Worldwide Quarterly Mobile Phone, https://www.idc.com/getdoc.jsp?containerId=US42776217 , Erişim:13-10-2017 .

The Global Positioning System,http://www.aerospace.org/research/missionassurance/gps/, Erişim tarihi:10/10/2017.

Wilson , M. 2008. T-Mobile G1: Full Details of the HTC Dream Android Phone,https://gizmodo.com/5053264/t-mobile-g1-full-details-of-the-htc-dream-android-phone,Erişim tarihi:12-10-2017.

.

EKLER

EK-1:GSM Tabanlı Çoklu Takip SistemYazılımı

```
package com.android.by_Remziye_cengiz;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.telephony.SmsMessage;
import android.widget.Toast;
import java.text.SimpleDateFormat;
import java.util.Date;
public class SmsBroadcastReceiver extends BroadcastReceiver{
public static final String SMS_BUNDLE = "pdus";
@Override
public void onReceive(Context context, Intent intent) {
Bundle intentExtras = intent.getExtras();
if(intentExtras != null) {
Object[] sms = (Object[]) intentExtras.get(SMS_BUNDLE);
String smsMessageStr = "";
for(int i = 0; i < sms.length; i++) {
SmsMessage smsMessage = SmsMessage.createFromPdu((byte[]) sms[i]);

String smsBody = smsMessage.getMessageBody().toString();
            String address = smsMessage.getOriginatingAddress();
            long timeMillis = smsMessage.getTimestampMillis();
            Date date = new Date(timeMillis);
            SimpleDateFormat format = new

SimpleDateFormat("dd/MM/yy");
            String dateText = format.format(date);
           //smsMessageStr+=address+"at"+"\t"+dateText+"\n";
            smsMessageStr+=smsBody+"\n";
         }

                int[]   intArray   =   new   int[smsMessageStr.length()];
                 for (int i = 0; i < smsMessageStr.length(); i++)
```

```
{rray[i]      =      Character.digit(smsMessageStr.charAt(i),      10);
            }
              //                          intArray[0]=intArray[0]+0;
            Toast.makeText(context,   "   Burdayım   babo   :-)   "   ,
Toast.LENGTH_LONG).show();

            GlobalClass.lat                                          =
intArray[3]*10000+intArray[4]*1000+intArray[5]*100+intArray[6]*10+in
tArray[7];
            GlobalClass.lng                                          =
intArray[14]*10000+intArray[15]*1000+intArray[16]*100+intArray[17]*1
0+intArray[18];
            GlobalClass.lat=GlobalClass.lat/1000;
            GlobalClass.lng=GlobalClass.lng/1000;
            Toast.makeText(context,   "   Enlem   "   +GlobalClass.lat,
Toast.LENGTH_LONG).show();
            Toast.makeText(context,   "   Boylam   "   +GlobalClass.lng,
Toast.LENGTH_LONG).show();
            ReceiveSMSActivity inst = ReceiveSMSActivity.instance();
            if(inst              !=              null)              {
                inst.updateList(smsMessageStr);
```

```
package com.android.by_Remziye_cengiz;
import android.app.Application;
/**
 * Created by Administrator on 28.9.2017.
  */
 public class GlobalClass extends Application {
     public static double lat;
     public static double lng;}
```

```
package com.android.by_Remziye_cengiz;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
public class MainActivity extends Activity {
 @Override
```

```
 protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
 setContentView(R.layout.activity_main);
Button maps = (Button)findViewById(R.id.button);
maps.setOnClickListener(new View.OnClickListener(){
@Override
        public void onClick (View view) {
        Intent i = new Intent(getApplicationContext(), MapsActivity.class);
        startActivity(i);
         finish()
        ;}
        });
    }
    public void goToInbox(View v) {
        Intent intent = new Intent(MainActivity.this,
ReceiveSMSActivity.class);
        startActivity(intent);
    }
    public void goToCompose(View v) {
        Intent intent = new Intent(MainActivity.this,
SendSmsActivity.class);
        startActivity(intent);
    }
}
```

```
package com.android.by_Remziye_cengiz;
import android.content.Intent;
import android.support.v4.app.FragmentActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {
```

```
    Button button1;
    private GoogleMap mMap;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);
        // Obtain the SupportMapFragment and get notified when the map is ready to be used.
        SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager()
                .findFragmentById(R.id.location_map);
        mapFragment.getMapAsync(this);


        button1 = (Button) findViewById(R.id.button1);
        button1.setOnClickListener(new View.OnClickListener(){
                        @Override
                        public void onClick (View view) {
                          Intent i = new
Intent(getApplicationContext(), MainActivity.class);
                                                startActivity(i);
                                                finish();


                                            }
            });
        }
        /**
         * Manipulates the map once available.
         * This callback is triggered when the map is ready to be used.
         * This is where we can add markers or lines, add listeners or
move the camera. In this case,
* we just add a marker near Sydney, Australia.
         * If Google Play services is not installed on the device, the
user will be prompted to install
         * it inside the SupportMapFragment. This method will only be
triggered once the user has
         * installed Google Play services and returned to the app.
```

```
@Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;
        // Add a marker in Sydney and move the camera
        //int lat=intArray[0]+intArray[1]*10;
        LatLng sydney = new LatLng(GlobalClass.lat, GlobalClass.lng);
        mMap.addMarker(new
MarkerOptions().position(sydney).title("buradayım"));
    }
    public void goToInbox(View v) {
        Intent intent = new Intent(MapsActivity.this,
MainActivity.class);
        startActivity(intent);
    }
}.
```

```
package com.android.by_Remziye_cengiz;
import android.app.Activity;
import android.content.ContentResolver;
import android.content.Intent;
import android.database.Cursor;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;
import android.widget.Toast;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
public class ReceiveSMSActivity extends Activity implements
AdapterView.OnItemClickListener{
    Button button2;
    private static ReceiveSMSActivity inst;
    ArrayList<String> smsMessageList = new ArrayList<String>();
    ListView smsListView;
    ArrayAdapter arrayAdapter;
public static ReceiveSMSActivity instance() {
    return inst;
  }
  @Override
```

```
    protected void onStart() {
        super.onStart();
        inst = this;
    }
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_receive_sms);
        smsListView = (ListView) findViewById(R.id.SMSList);
        arrayAdapter = new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1,
smsMessageList);
        smsListView.setAdapter(arrayAdapter);
        smsListView.setOnItemClickListener(this);
        refreshSmsInbox();
        button2 = (Button) findViewById(R.id.button2);
        button2.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick (View view) {
                Intent i = new Intent(getApplicationContext(), MainActivity.class);
        startActivity(i);
                finish();
            }
        });
    }
    public void refreshSmsInbox() {
        ContentResolver contentResolver = getContentResolver();
        Cursor smsInboxCursor = contentResolver.query(Uri.parse("content://sms/inbox"), null,
null, null, null);
        int indexBody = smsInboxCursor.getColumnIndex("body");
        int indexAddress = smsInboxCursor.getColumnIndex("address");
        int timeMillis = smsInboxCursor.getColumnIndex("date");
        Date date = new Date(timeMillis);
        SimpleDateFormat format = new SimpleDateFormat("dd/MM/yy");
```

```
    String dateText = format.format(date);
    if(indexBody < 0 || !smsInboxCursor.moveToFirst()) return;
    arrayAdapter.clear();
    do {
      String str = smsInboxCursor.getString(indexAddress) + " at
     "
                    + "\n" + smsInboxCursor.getString(indexBody) +
dateText + "\n";
            arrayAdapter.add(str);
        } while(smsInboxCursor.moveToNext());
    }

    public void updateList(final String smsMessage) {
        arrayAdapter.insert(smsMessage, 0);
        arrayAdapter.notifyDataSetChanged();
    }

    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {

        try {
            String[] smsMessages =
smsMessageList.get(position).split("\n");
            String address = smsMessages[0];
            String smsMessage = "";

            for(int i = 1; i < smsMessages.length; i++) {
                smsMessage += smsMessages[i];
            }

            String smsMessageStr = address + "\n";
            smsMessageStr += smsMessage;
            Toast.makeText(this, smsMessageStr,
Toast.LENGTH_SHORT).show();
            Toast.makeText(getApplicationContext(), smsMessages[2],
```

```
Toast.LENGTH_SHORT).show();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    public void goToCompose(View v) {
        Intent intent = new Intent(ReceiveSMSActivity.this,
SendSmsActivity.class);
        startActivity(intent);
    }
}
```

```
package com.android.by_Remziye_cengiz;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;public class SendSmsActivity extends
Activity {

    Button sendSmsBtn1;
    Button sendSmsBtn2;
    Button haritabul;

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

        sendSmsBtn1 = (Button) findViewById(R.id.cihaz1);

        sendSmsBtn1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                sendSms();
            }
        });

        sendSmsBtn2 = (Button) findViewById(R.id.cihaz2);

        sendSmsBtn2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                sendSms2();
            }
        });

        haritabul = (Button)findViewById(R.id.button);
        haritabul.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick (View view) {
                Intent i = new Intent(getApplicationContext(),
MapsActivity.class);
                startActivity(i);
```

```
                finish();

            }
        });

    }

private void sendSms() {
    String toPhone = "05312224087";
    String smsMessage = "BBB";
   // toPhone="05308736003";
   // smsMessage="Nerdesin";
    try {
      SmsManager smsManager = SmsManager.getDefault();
      smsManager.sendTextMessage(toPhone, null, smsMessage, null, null);

Toast.makeText(this, "mesaj gönderildi cihaz 1", Toast.LENGTH_LONG).show();
    } catch (Exception e) {
      e.printStackTrace();
    }    private void sendSms2() {
        String toPhone = "05437489879";
        String smsMessage = "BBB";
        // toPhone="05308736003";
        // smsMessage="Nerdesin";
        try {
            SmsManager smsManager = SmsManager.getDefault();
            smsManager.sendTextMessage(toPhone, null, smsMessage,
null, null);

            Toast.makeText(this, "mesaj gönderildi cihaz 2",
Toast.LENGTH_LONG).show();
        } catch (Exception e) {
            e.printStackTrace();
        }

    }
    public void goToInbox(View v) {
        Intent intent = new Intent(SendSmsActivity.this,
MainActivity.class);
        startActivity(intent);
    }
}

package com.android.by_Remziye_cengiz;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.telephony.SmsMessage;
import android.widget.Toast;
import java.text.SimpleDateFormat;
import java.util.Date;
public class SmsBroadcastReceiver extends BroadcastReceiver{
  public static final String SMS_BUNDLE = "pdus";
```

```
    @Override
    public void onReceive(Context context, Intent intent) {
  Bundle intentExtras = intent.getExtras();
        if(intentExtras != null) {
          Object[] sms = (Object[]) intentExtras.get(SMS_BUNDLE);
          String smsMessageStr = "";
  for(int i = 0; i < sms.length; i++) {
                    SmsMessage smsMessage =
  SmsMessage.createFromPdu((byte[]) sms[i]);

                    String smsBody =
  smsMessage.getMessageBody().toString();
                    String address = smsMessage.getOriginatingAddress();
                    long timeMillis = smsMessage.getTimestampMillis();

                    Date date = new Date(timeMillis);
                    SimpleDateFormat format = new
  SimpleDateFormat("dd/MM/yy");
                    String dateText = format.format(date);

                  // smsMessageStr += address + " at " + "\t" + dateText
  + "\n";
                    smsMessageStr += smsBody + "\n";
               }

              int[] intArray = new int[smsMessageStr.length()];

               for (int i = 0; i < smsMessageStr.length(); i++) {
                    intArray[i] =
  Character.digit(smsMessageStr.charAt(i),
10);

  }
         // intArray[0]=intArray[0]+0;
         Toast.makeText(context, " Burdayım babo :-) " , Toast.LENGTH_LONG).show();


              GlobalClass.lat                                          =
intArray[3]*10000000+intArray[4]*1000000+intArray[5]*100000+intArray
[6]*10000+intArray[7]*1000+intArray[8]*100+intArray[9]*10+intArray[1
0];
              GlobalClass.lng                                          =
intArray[14]*10000000+intArray[15]*1000000+intArray[16]*100000+intAr
ray[17]*10000+intArray[18]*1000+intArray[19]*100+intArray[20]*10+int
Array[21];
              GlobalClass.lat=GlobalClass.lat/1000000;
              GlobalClass.lng=GlobalClass.lng/1000000;
              Toast.makeText(context,   "   Enlem   "   +GlobalClass.lat,
Toast.LENGTH_LONG).show();
              Toast.makeText(context,   "   Boylam   "   +GlobalClass.lng,
```

```
Toast.LENGTH_LONG).show();
            ReceiveSMSActivity inst = ReceiveSMSActivity.instance();
            if(inst              !=              null)              {
                inst.updateList(smsMessageStr);
            }
        }
```

ÖZGEÇMİŞ

Adı Soyadı : Remziye CENGİZ

Doğum Yeri : Diyarbakır

Doğum Tarihi : 05/06/1983

Medeni Hali : Bekar

Yabancı Dil : İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise : Diyarbakır Ziya Gökalp Lisesi(1997-2000)

Lisans : Uluslararası Kıbrıs Üniversitesi, Elektrik-Elektronik Mühendisliği (2005-2010)

Çalıştığı Kurum/Kurumlar ve Yıl:

Fırat Telekom (2010-2011), Diyarbakır

Karavil Grup (2011-2012), Diyarbakır

Dicle Üniversitesi, Meslek Yüksek Okulu, Ücretli Öğretim Görevlisi (2012-2013)

Cengiz Ticaret (2013-Halen),Diyarbakır

T.C.
DİCLE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS TEZ ÇALIŞMASI İNTİHAL RAPORU FORMU

ÖĞRENCİ BİLGİLERİ	
ADI VE SOYADI	Remziye CENGİZ
ÖĞRENCİ NO	16805007
EĞİTİM – ÖĞRETİM YILI	2017 - 2018
YARIYIL	☒ Güz ☐ Bahar
ANABİLİM DALI	Elektrik-Elektronik Mühendisliği
PROGRAM	Yüksek Lisans
TEZ KONUSU	GSM Tabanlı Çoklu Takip Sistemi
İNTİHAL RAPORU BİLGİLERİ	
RAPOR TÜRÜ	Tez Savunma Sınavı Sonrası
SAYFA SAYISI	92
BENZERLİK ORANI	%18
RAPORLAMA TARİHİ	26/02/2018

Yukarıda başlığı/konusu gösterilen tez çalışmamın kapak sayfası, giriş, ana bölümler, sonuç ve tartışma kısımlarından oluşan toplam 92 sayfalık kısmına ilişkin, 26/02/2018 tarihinde şahsım/tez danışmanım tarafından *Turnitin* adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan intihal raporuna göre, tezimin benzerlik oranı %18 'tür.

Uygulanan filtrelemeler:

- ☐ Kabul/Onay sayfaları hariç,
- ☐ Kaynakça hariç
- ☐ Alıntılar hariç/dâhil
- ☐ Diğer

Dicle Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Programlarda Tez Çalışması İntihal Raporu Uygulama Esasları'nı inceledim ve bu Uygulama Esasları'nda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edilmesi durumunda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.

(İmza)
Remziye CENGİZ

(İmza) 05/03/2018 Doç.Dr. Mehmet Siraç ÖZERDEM **Tez Danışmanı**	(İmza) 05/03/2018 Prof.Dr. İbrahim KAYA **Anabilim Dalı Başkanı**