KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ
ANABİLİM DALI

YÜKSEK LİSANS TEZİ

SAYISAL TAKOGRAFLAR İÇİN MOBİL VERİ İNDİRME

KADİR AKTAŞ

KOCAELİ 2018

KOCAELİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ

ANABİLİM DALI

YÜKSEK LİSANS TEZİ

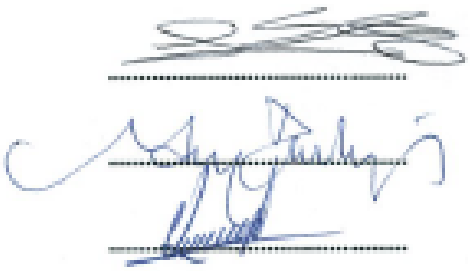
SAYISAL TAKOGRAFLAR İÇİN MOBİL VERİ İNDİRME

KADİR AKTAŞ

Prof. Dr. Sarp ERTÜRK
Danışman, Kocaeli Üniversitesi

Prof. Dr. Adnan DERDİYOK
Jüri Üyesi, Sakarya Üniversitesi

Yrd. Doç. Dr. Oğuzhan KARAHAN
Jüri Üyesi, Kocaeli Üniversitesi



Tezin Savunulduğu Tarih: 09.07.2018

ÖNSÖZ VE TEŞEKKÜR

Karayoluyla yolcu veya eşya taşımacılığı yapan araçların hareketlerini ve sürücülerinin faaliyetlerini otomatik veya yarı otomatik olarak göstermek ve kaydetmek için araçlara monte edilen sayısal takograflar gün geçtikçe uluslararası piyasada zorunlu hale gelmektedir. Bu tezde bahsi geçen sayısal takograf veri indirme uygulaması ile sayısaltakografa kaydedilen verilerin Android işletim sistemine sahip bir cihazla toplanması ile karayolu güvenliğinin artırılması amaçlanmaktadır.

Yüksek lisans eğitim ve tez çalışma sürecim içerisinde yardımlarını esirgemeyen danışmanım Prof. Dr. Sarp ERTÜRK başta olmak üzere tüm hocalarıma, çalışma arkadaşım Tolga TEKİN'e ve diğer tüm çalışma arkadaşlarıma teşekkür ederim.

Ayrıca hayatım boyunca benden emeğini, güvenini, sevgisini hiç esirgemeyen, eğitim sürecim boyunca hep yanımda olan aileme en içten teşekkürlerimi sunarım.

Temmuz-2018 Kadir AKTAŞ

İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR i
İÇİNDEKİLER ii
ŞEKİLLER DİZİNİ iv
TABLOLAR DİZİNİ vi
SİMGELER VE KISALTMALAR DİZİNİ vii
ÖZET ix
ABSTRACT x
GİRİŞ 1
1. MOBİL UYGULAMA 3
1.1. Mobil Uygulama Nedir? 3
1.2. Mobil Uygulamaların Geçmişi 4
1.3. Mobil Uygulama Geliştirme 5
1.4. Mobil Uygulama İşletim Sistemlerinin Karşılaştırılması 6
2. ANDROID PLATFORM 8
2.1. Android İşletim Sistemi 8
2.2. Android Yazılım Geliştirme 11
2.2.1. Android yazılım geliştirme kiti 11
2.2.2. Android programlama dili 12
2.2.3. Android debug bridge (ADB) 14
2.2.4. Android programlamada aktivite ve düzen yapıları 14
2.2.5. Android programlama proje dosyaları 17
3. SAYISAL TAKOGRAF 19
3.1. Sayısal Takografa Ait Kart Çeşitleri 20
3.1.1. Sürücü kartı 20
3.1.2. Şirket kartı 21
3.1.3. Denetimkartı 21
3.1.4. Servis kartı 22
4. SAYISAL TAKOGRAF VERİ İNDİRME VE ANALİZ UYGULAMASI 23
4.1. Sayısal Takograf Veri İndirme 23
4.1.1. İndirme yöntemi 23
4.1.2. Veri indirme protokolü 23
4.1.2.1. Mesaj yapısı 24
4.1.2.2. Mesaj tipleri 24
4.1.2.3. Mesaj akışı 25
4.1.2.4. Zamanlama 26
4.1.2.5. Kart indirme 27
4.1.3. İndirilen verilerin çözümlenmesi 28
4.2. Veri İndirme Cihazları Piyasa Taraması ve Karşılaştırma 32
4.3. Uygulamanın Çalışma Yapısı 34
4.3.1. Sistemin genel mimarisi 35
4.3.2. Seri haberleşme (donanımsal arayüz) 35
4.3.3. Veri indirme 40
4.3.4. Veri görüntüleme 42

4.3.4.1. Araç ve kart dosya isimlendirme formatı 42
4.3.4.2. Araç verisi görüntüleme 42
4.3.4.3. Kart verisi görüntüleme 48
5. SONUÇLAR VE ÖNERİLER 51
KAYNAKLAR 52
EKLER 54
KİŞİSEL YAYINLAR VE ESERLER 62
ÖZGEÇMİŞ 63

ŞEKİLLER DİZİNİ

Şekil 2.1. Android işletim sistemi mimarisi 10
Şekil 2.2. Donanım cihazıyla birleşmiş Android ve Java akış şeması 11
Şekil 2.3. Androidsdk manager 12
Şekil 2.4. Android sisteminin aktivite yaşam döngüsü akış şeması 16
Şekil 3.1. Sürücü kartı ön ve arka yüz 21
Şekil 3.2. Şirket kartı ön ve arka yüz 21
Şekil 3.3. Denetim kartı ön ve arka yüz 22
Şekil 3.4. Servis kartı kartı ön ve arka yüz 22
Şekil 4.1. Mesaj tipleri 25
Şekil 4.2. Mesaj akışı 26
Şekil 4.3. Mesaj akışı zamanlama 26
Şekil 4.4. Zamanlama için alt ve üst sınır değerler 27
Şekil 4.5. Kart dosya kimlikleri 28
Şekil 4.6. Genel bilgi verisi yorumlama 29
Şekil 4.7. Faaliyet (aktivite) verisi yorumlama 30
Şekil 4.8. Olay ve hata verilerini yorumlama 31
Şekil 4.9. Detaylı hız verisini yorumlama 32
Şekil 4.10. Teknik bilgi verisini yorumlama 32
Şekil 4.11. Piyasada bulunan veri indirme cihazlarından örnekler 33
Şekil 4.12. Uygulamanın açılış ekranı 34
Şekil 4.13. Uygulamanın ana menüsü 35
Şekil 4.14. Sayısal takograf ön bağlantı yapısı 36
Şekil 4.15. Altı bacaklı fiş çizimi 37
Şekil 4.16. Bir haberleşme çerçevesinin yapısı 38
Şekil 4.17. Normal kablo ile otg kablo arasındaki fark 38
Şekil 4.18. Donanımın bitmiş hali 39
Şekil 4.19. Donanıma ait kapaklar 39
Şekil 4.20. Veri indirme türü seçenek menüsü 40
Şekil 4.21. Araç verisi indirme seçenek menüsü 41
Şekil 4.22. Kart verisi indirme seçenek menüsü 41
Şekil 4.23. İndirme kalan gün sayısı bilgi ekranı 41
Şekil 4.24. Veri görüntüleme seçimi 42
Şekil 4.25. Görüntülenecek aracın seçimi 43
Şekil 4.26. Genel bilgi görüntüleme ekranı 44
Şekil 4.27. Şirket kilit kayıtları görüntüleme ekranı 44
Şekil 4.28. Kontrol aktivite bilgileri görüntüleme ekranı 45
Şekil 4.29. Aktivite örnek analiz ekranı 46
Şekil 4.30. Arıza kaydı örnek ekran görüntüsü 47
Şekil 4.31. Olay kayıtları örnek ekran görüntüsü 47
Şekil 4.32. Detaylı hız verisi örnek analiz ekranı 48
Şekil 4.33. Teknik bilgi verisi örnek analiz ekranı 48
Şekil 4.34. Kart verisi analiz örnek ekran görüntüsü, kart bilgisi 49
Şekil 4.35. Kart verisi analiz örnek ekran görüntüsü, olay kayıtları 50

Şekil 4.36. Kart verisi analiz örnek ekran görüntüsü, aktivite kayıtları 50
Şekil A.1. Uygulamanın USB bağlantı aşaması ekran görüntüsü 55
Şekil A.2. Uygulamanın veri indirme seçenek ekran görüntüsü 55
Şekil A.3. Aktivite tarih seçimi ekran görüntüsü 55
Şekil A.4. Şirket kilit kayıtları ekran görüntüsü 56
Şekil A.5. Aşırı hız kayıtları ekran görüntüsü 56
Şekil A.6. Zaman ayarlamaları ekran görüntüsü 56
Şekil A.7. Kalibrasyon kayıtları ekran görüntüsü 57
Şekil A.8. Kalibrasyon parametreleri ekran görüntüsü 57
Şekil A.9. Olay kayıtları ekran görüntüsü 57
Şekil A.10. Kart yetkinlik belgesi ekran görüntüsü 58
Şekil A.11. Kart bilgileri ekran görüntüsü 58
Şekil A.12. Kart sahibi bilgileri ekran görüntüsü 58
Şekil A.13. Karttaki olay bilgileri ekran görüntüsü 59
Şekil A.14. Karttaki hataların ekran görüntüsü 59
Şekil A.15. Kartın takıldığı araç bilgilerinin ekran görüntüsü 59
Şekil A.16. Kart konum kayıtları ekran görüntüsü 60
Şekil A.17. Avrupa Birliği Yönetmeliği'ne göre sürüş ve dinlenme ihlalleri ekran görüntüleri 60
Şekil A.18. Karayolları Trafik Yönetmeliği'ne göre sürüş ve dinlenme süre ihlalleri ekran görüntüsü 60
Şekil A.19. Karta kayıtlı olayların örnek ekran görüntüsü 61

TABLOLAR DİZİNİ

Tablo 1.1. Mobil işletim sistemi, üretici firmaları ve uygulama mağazaları 3
Tablo 2.1. Mobil Android yazılım yığını katmanları 9
Tablo 4.1. Veri indirme işlemi mesaj yapısı 24
Tablo 4.2. Ön bağlantı bacakları işlevleri 36

SİMGELER VE KISALTMALAR DİZİNİ

C : Kondansatör
D : Data
F : Farad
J : Jumper
R : Direnç
U : Entegre
V : Volt

Kısaltmalar

ADB : Android Debug Bridge (Android Hata Ayıklama Köprüsü)
AETR : Accord Européen sur les Transports Routiers (Uluslararası Karayolu Taşımacılığı Yapan Taşıtlarda Çalışan Personelin Çalışmalarına İlişkin Avrupa Anlaşması
ALW : Always (Daima)
API : Aplication Programming Interface (Uygulama Programlama Arayüzü
APK : Android Application Package (Andorid Uygulama Paketi)
AUT : Authenticated (Kimlik Doğrulamalı)
BD : Baud (Sembol Oran Birimi)
BM-AEK : Birleşmiş Milletler Avrupa Ekonomik Konseyi
CA : Certification Authority (Sertifika Otoriteleri)
CS : Checksum Byte (Denetim Toplam Baytı)
DATA : Veri
DS : Diagnostic Session Byte (Teşhis Oturum Baytı)
DVD : Digital Versatile Disc (Çok Amaçlı Sayısal Disk)
EC : European Commission (Avrupa Komisyonu)
EEC : European Economic Community (Avrupa Ekonomik Topluluğu)
EF : Elementary Files (Temel Dosyalar)
FMT : Format Byte (Biçim Baytı)
G/Ç : Giriş / Çıkış
GND : Ground (Toprak Hattı)
GPS : Global Positioning System (Küresel Konumlama Sistemi)
HGV : Heavy Goods Vehicle (Ağır Yük Taşıtı)
ICC : International Certificate of Competence (Uluslararası Yetkinlik Belgesi)
IDE : Integrated Development Environment (Tümleşik Geliştirme Ortamı)
ISO : International Organisation of Standardization (Uluslararası Standartlar Teşkilatı)
iOS : Iphone Operating System (Iphone İşletim Sistemi)
KWP : Keyword Protocol (Şifre Kelime Protokolü)
LEN : Length Byte (Uzunluk Baytı)
MF : Main File (Ana Dosya)

MLK :Memory Leak (Bellek Sızıntısı)
NEV : Never (Hiç)
OTG : On the Go (Tak Çalıştır, Faal)
PCB : Printed Circuit Board (Baskılı Devre Kartı)
PRO SM : Protected with Secure Messaging (Güvenli Mesaj ile Korunan)
RxD : Receive Data (Veri Alma)
SDK : Software Development Kit (Yazılım Geliştirme Kiti)
SID : Service Identifier (Hizmet Tanımlayıcısı)
SRC : Source Byte (Kaynak Baytı)
TGT : Target Byte (Hedef Baytı)
TREP : Transfer Response Parameter (Aktarım Cevap Parametresi)
TRTP : Transfer Request Parameter (Aktarım İstek Parametresi)
TTL : Transistor to Transistor Logic (Sadece Transistör ve Dirençten Oluşan Mantık Devresi
TxD : Transmit Data (Veri İletimi)
UART : Universal Asynchronous Receiver Transmiter (Evrensel AsenkronAlıcı Verici)
USB : Universal Serial Bus (Evrensel Seri Hat)
V_{cc} : Collector Supply Voltage (Kolektör Besleme Gerilimi)
VCD : Video Compact Disc (Video Disk Kayıt Standardı)
VU : Vehicle Unit (Takograf Cihazı)

SAYISAL TAKOGRAFLAR İÇİN MOBİL VERİ İNDİRME

ÖZET

Bu çalışmada sayısal takograflar için hızlı ve yerinde veri indirme, görüntüleme ve analiz imkanı sağlayan veri indirme yazılımı geliştirilmiştir. Geliştirilen yazılım Android işletim sistemli cihazlar üzerinde çalıştırılarak mobil hale getirilmiştir. Bu uygulama sayesinde, sayısal takograf verilerini ve/veya sürücü kartı verilerini indirip bu verileri anında ve herhangi bir harici teçhizata ihtiyaç duymadan çözümleyip analizine imkan sağlamak mümkündür. Böylelikle, normal şartlarda yapılan veri indirmelere oranla zaman, hız ve kullanım kolaylığı ile büyük bir fark meydana gelecektir.

Anahtar Kelimeler: Sayısal Takograf, Sayısal Takograf Verisi, Veri Analizi.

MOBILE DATA DOWNLOADING FOR DIGITAL TACHOGRAPHS

ABSTRACT

In this study, data downloading software for digital tachographs that provides quick download, data analysis and data displaying has been developed. By adaptingon Android operating system, the software is made mobile. Downloading of digital tachograph data and/or driver card data and data analysis without any external equipment is possible with this application. Thus, the big differences like time saving, speed and easy of use will be seen when compared mobile downloading to other classic data downloading systems.

Keywords: Digital Tachograph, Digital Tachographs Data, Data Analysis.

GİRİŞ

Günümüzde hızla ilerleyen teknoloji mobil alanda da etkisini iyiden iyiye göstermeye başlamıştır. Gelişen teknoloji ile mobil cihazlar yaşamımızın her anına etki etmektedir. Dolayısıyla da mobil uygulamaların sayısı gün geçtikçe artmaktadır. Kullanıcı sayısı da göz önünde bulundurulursa Android platformunun altın çağını yaşadığını söyleyebiliriz. Android işletim sisteminin açık kaynak kodlu olması cihaz ve kullanıcı sayısına etki etmiş ve böylece çok sayıda kullanıcıya ulaşılmasına imkan sağlamıştır.

Mobil cihazlara yönelik yaygın olarak kullanılan işletim sistemleri Google tarafından geliştirilen Android, Apple tarafından geliştirilen iOS ve Microsoft tarafından geliştirilen Windows Mobile işletim sistemleridir [1]. Android, Linux tabanlı mobil bir işletim sistemidir. Android'in açık kaynak kodlu olması, uygulama geliştiricilerinin çokluğu ve buna bağlı olarak Android ortamında çalışabilen yüz binlerce uygulamanın olması, Android işletim sisteminin diğer mobil işletim sistemlerinin bir adım önüne geçmesini sağlamaktadır. Android şu anda dünya üzerinde yüz milyonlarca mobil cihaz üzerinde kurulu olan mobil bir işletim sistemi olmakla beraber her gün binlerce yeni Android uygulaması Google Play'e yüklenmektedir. Google Play, Google tarafından işletilen kurumsal Android uygulama mağazasıdır. Bunun yanı sıra bu uygulamalara çeşitli sitelerden de ulaşılabilmektedir [2].

Karayolu taşımacılığı yapan belirli araçlara yasa ve yönetmelikler kapsamında yürürlüğe konulan kanun veya kanun hükmündeki kararnameler ile sayısal takograf kullanımı Avrupa ülkelerinde 1 Mayıs 2006 tarihinden itibaren tüm yeni kayıt olmuş araçlarda zorunlu olmuştur. Ülkemiz gibi AETR anlaşmasına taraf ülkelerde ise söz konusu uygulama 16 Haziran 2010 tarihinden itibaren zorunlu hale gelmiştir. Yapılan düzenleme 16 Haziran 2010 tarihinden sonra tescil edilen 3,5 ton ve daha fazla yük taşımacılığı yapan araçlar ve/veya sürücü dahil 9 kişi veya daha fazla yolcu taşıyan araçlar için uygulanmaktadır. Sayısal takografların dışarıdangelebilecek

sabotaj veya hileli yönlendirmeye daha kapalı olması sebebiyle ve ayrıca yol ve sürüş güvenliğinin artmasına katkı sağlayabilmek adına kullanılması düşünülmüştür[3].

Sayısal takograf kullanımının 2010 tarihinden itibarenkademeli olarak zorunlu hale gelmesiyle beraber sayısal takograf cihazlarının hafızasına kaydedilen verilere ulaşıp inceleyebilmek ve yasal gereksinimleri karşılayacak şekilde indirilen verileri elektronik olarak saklayabilmek için veri indirme cihazları geliştirilmiştir. Geliştirilen veri indirme cihazları sayısal takograf hafızasına kaydedilen verileri kendi hafızasına kopyalama yolu ile kaydederek daha sonra bu bilgiler üzerinde yapılabilecek analizlere imkan sağlar.

Piyasada kullanılan sayısal takograf veri indirme cihazları genel olarak bir bellek vazifesi görmektedir. Sayısal takograftan indirilen veriler veri indirme cihazının hafızasına kaydedilir. Sonrasında harici bir teçhizat üzerinden harici bir yazılım vasıtasıyla analizi yapılabilir. Bu durum daha fazla maliyet, çaba ve zaman gerektirdiğinden dolayı bu çalışmada akıllı bir uygulama geliştirilmiştir. Bu uygulama ile sayısal takograf hafızasından kopyalanan veriler harici bir cihaz veya yazılıma ihtiyaç duymadan, Android işletim sistemine ve OTG bağlantı desteğine sahip bir tablet veya telefon üzerinden veriler anında indirilebilir. İndirilen veriler uluslararası belirtilen yönetmeliklerdeki teknik veriler esas alınarak anlamlandırılır.

1. MOBİL UYGULAMA

1.1. Mobil Uygulama Nedir?

Teknolojinin gelişmesi ve bilgisayarların hayatımızın her alanına girmesiyle cep telefonları ve tablet bilgisayarlar gibi farklı elektronik cihazların gelişimi çok hızlı bir şekilde ilerlemektedir. Mobil cihaz olarak adlandırılan bu cihazlardaçalışması için tasarlanmış ve geliştirilmiş yazılımlara mobil uygulama denilmektedir. Mobil uygulamalar, mobil cihazların kullandığı mobil işletim sistemine uygun olarak yazılmaktadır [4].

Elektronik cihazların kullandığı birçok mobil işletim sistemi bulunmaktadır. En çok kullanılan Apple firmasının geliştirdiği ve kendi ürettiği mobil cihazlarda kullandığı iOS işletim sistemi ve Google firmasının geliştirdiği ve birçok mobil cihaz üreticisi ve kullanıcıları tarafından kullanılan Android işletim sistemidir. Mobil uygulamalar üreticilerine ait uygulama marketleri üzerinden indirilebilir veya izin verilmişse önce bilgisayarınıza indirip daha sonra indirdiğiniz mobil uygulamaları mobil cihazlarınıza aktarıp kurulum gerçekleştirilebilir. Tablo 1.1'de üretici firmalar, başlıca işletim sistemleri, cihazlar ve uygulama mağazaları gösterilmiştir.

Tablo 1.1. Mobil işletim sistemi, üretici firmaları ve uygulama mağazaları

Firma	Mobil İşletim	Cihazlar	Uygulama
Apple	iOS	Iphone ve Apple Tabletler	Apple App
Google	Android	Başlıcaları; Samsung, HTC, Siemens,	Google Play
Microsoft	Windows Mobile	Nokia Lumia, Windows Phone,	Windows Phone
RIM	Blackberry OS	Blackberry	BlackBerry App

Mobil uygulamalar günlük hayatımızda bilgiye erişmeyi kolaylaştırır. Aynı zamanda bir web sitesi üzerinden yapılabilecek bir işlemi daha az veri ve kolaylıkla tamamlanmasına olanak sağlar. Masaüstü uygulamalarının erişilebilirliği cihazla

sınırlı olduğu için mobil cihazlar ve uygulamalar kullanım kolaylığı ve bütçe açısından daha avantajlıdır.

Uygulamalar, mobil işletim sistem üreticileri tarafından uygulama marketlerinde sınıflandırılmıştır. Mobil uygulamalar genel olarak yapılacak işleri kolaylaştırmaya yönelik olduğundan süre ve verim açısından kullanıcılara fayda sağlar. Bilgisayarlarda kullanılan ve ihtiyaç olan uygulamaların mobil sürümlerinin olması her an bu uygulamalara ulaşılabilmesi anlamına gelmektedir. Mobil uygulamalar bu sebeplerle teknoloji ve insan hayatında önemli bir yer tutmaya devam edecektir [4].

1.2. Mobil Uygulamaların Geçmişi

Cep telefonlarının kullanıma başlanmasıyla mobil uygulamalara zamanla ihtiyaçduyulmuştur. İlk mobil uygulamalar sadece arama yapmak, mesaj göndermek ve alarm kurmaktan ibarettir. İlk mobil uygulamalar cihazla birlikte gelirken, üzerlerinde herhangi bir değişiklik yapılmasına imkan vermemekteydi.Mobil uygulama yazılımlarının geliştirilmesiyle beraber uygulamalarda yapılabilecek değişiklikler ile günlük hayatı daha kolay hale getirmeye başlamıştır.

Cep telefonları ve mobil cihazların yaygınlaşmasıyla yüksek seviyeli yazılım dilleri ile daha esnek olan mobil işletim sistemleri ortaya çıkmıştır. Mobil işletim sistemi üzerine ilk adımı atan Nokia şirketidir. Yayınlamış olduğu işletim sistemi "Symbian" işletim sistemidir. Bu işletim sistemi eskisinden daha da gelişmiş bir yapıya sahiptir ve gelişmiş mobil işletim sistemlerine ve uygulamalarına öncülük etmiştir [4].

Yeni çıkan ve geliştirilen mobil cihazların depolama alanlarında artış ve harici bir bellek ile genişleme olanağı sunulmuştur. Mobil cihazlardan internete bağlantı olanağı daha da kolaylaşmıştır ve internet üzerinden yapılacak işlemlerin sayısı daha da artmıştır. Bilgiye erişimin hızlanması ve internet hızının zamanla artması bireylerin mobil cihazlarda hem eğlence hem sosyal hem de kişisel faaliyetlerini yapma amaçlı işlerinde önemli bir rol oynamıştır.

Mobil uygulama mağazalarının ortaya çıkması ile birlikte şirketler kendi mobil işletim sistemlerine uygun şirket dışı geliştiricilerin ve kullanıcıların mobil uygulama geliştirmesine imkan sağlamıştır. Geliştirilen mobil uygulamaların ve geliştiricilerin

sayısının hızla artmasının sebebini yeni nesil mobil işletim sistemleri olduğunu söyleyebiliriz. Mobil uygulamaların gelişmesinde en fazla pay sahibi olan işletim sistemleri, Apple şirketinin 2007 yılında iPhone ile birlikte tanıttığı iOS mobil işletim sistemi ve Google şirketinin 2005 yılından beri geliştirilme aşamasında olan, 2008 yılında ilk kez piyasaya sunulan Android mobil işletim sistemidir.

Günümüzde mobil cihazların ve uygulamalarının artması ve her mobil platforma özel uygulamaların yazılması, her platformda çalışabilen uygulamaların geliştirilmesine ihtiyaç doğurmuştur.

Sonuç olarak mobil uygulamalar son yıllarda hızla gelişerek milyonlarca insanın hayatını değiştirmiş, günlük hayatı ve iş hayatını kolaylaştırmış ve aynı zamanda sosyal paylaşım sitelerine her an bağlı olabilme imkanı sunarak her şeyden haberdar olmamızı sağlamıştır[4].

1.3. Mobil Uygulama Geliştirme

Yakın zamanda kullanıcıların sahip oldukları cep telefonları ve tablet bilgisayarlardan oluşan mobil cihazların tutulmasının en önemli iki faktörü vardır. Bunlardan birinicisi işletim sisitemidir. İkincisi ise uygulama marketinin ne kadar kaliteli olduğudur. Dünyada mobil uygulama pazarında en büyük paya sahip üç işletim sistemi vardır. Bunlar; Android, iOS ve Windows Phone 8 işletim sistemleridir.

Mobil cihaza işletim sisteminin desteklediği yazılım dilleri ile uygulama geliştirmeye mobil uygulama geliştirme denir. Her platform kendine özgü yazılım diline sahiptir. Android için geliştirilen uygulama aynı zamanda iOS platformunda çalışmayabilir. Aynı durum iOS geliştirme ortamı için de geçerlidir. Mobil uygulamalar geliştirilirken yüksek seviye diller kullanılır. Android geliştirme ortamı için Java programlama dilini bilmek gerekir, iOS işletim sistemi için ise Objective-C dilini bilmek gerekir. Her mobil işletim sisteminin yazılım dili ve geliştirme ortamları farklıdır.

Mobil uygulama geliştirmede Native, Web, Hibrit tabanlı olmak üzere üç çeşit yazılım geliştirme yolu vardır. Tez uygulamasında native dil kullanılmıştır. Native

uygulama, performans ve güvenilirlik açısından daha üstündür. Native uygulama ile işletim sisteminin donanımsal kaynaklarına rahatlıkla erişebiliriz. Varolan kamera, mikrofon ve diğer giriş, çıkış kaynaklarını kontrol etmek daha kolaydır. Native dil ile geliştirilen uygulamaların animasyon ve grafik kalitesi daha yüksek olup, diğer dillere oranla da daha hızlı tepki veririler. Native yazılım geliştirme kullanıcılara daha geniş yelpazede deneyim sunmaktadır. Android işletim sistemi için Native dil Java, iOS için ise Objective-C'dir.

Mobil uygulamaları geliştirebilmek için mobil cihaza uygun platformu seçmeniz gerekmektedir. Android uygulama geliştirmek için Windows, Linux veya MacOS işletim sistemli bir bilgisayar ve Eclipse SDK ya da Android Studio yeterli olacaktır. iOS için ise MacOS tabanlı bir bilgisayarınız olması gereklidir. Geliştirme ortamı olarak Xcode kullanılır. Alternatif programlar da bulunmaktadır.

Mobil uygulamalar, mobil marketler üzerinden dağıtılır. Her işletim sisteminin ayrı bir uygulama marketi bulunmaktadır. Android uygulamaları için Google Play Store, iOS uygulamaları için Apple AppStore, Windows için Windows Phone Store marketleri bulunmaktadır. Uygulamaları ücretli veya ücretsiz olarak kullanabilir ya da yayınlayabilirsiniz[4].

1.4. Mobil Uygulama İşletim Sistemlerinin Karşılaştırılması

Son yıllarda teknoloji dünyasında en sıkı rekabet mobil dünyada ve özellikle cep telefonu pazarında yaşanıyor. Bu rekabetin önemli bir bölümü ise Appletarafından geliştirilen iOSve Googletarafından geliştirilen Androidarasında yaşanıyor. Hem iOS hem de Android işletim sistemleri sadece cep telefonlarında değil aynı zamanda tablet bilgisayarlarda da çalışabiliyor. Birbirlerine göre avantaj ve dezavantajlara sahip olan işletim sistemlerinden iOS kapalı bir mimariye sahipken, Android ise açık kaynak mimarisine sahiptir. Apple tarafından geliştirilen iOS işletim sistemi kapalı mimari olduğundan dolayı dış saldırılara karşı daha güvenlidir. Ancak bu cihazlarda dışarıdan erişim biraz daha zordur. Cihaza veri eklemek için özel bir yazılım programı gerekmektedir. Android açık kaynak kodlu olduğundan dolayı cihaz çeşitliliği çok fazladır. Bu özellik cihazlara esneklik kazandırmakta ama diğer bir yandan da saldırılara karşı açık hedef haline gelmesine sebep olmaktadır. Cihaz çeşitliliği uygulama geliştiricilerini zorlayan bir etken haline gelmektedir. Her iki

işletim sistemi de hız ve kararlı çalışma konusunda gayet iyi noktalara gelmiştir ve böylece yüksek donanımlı cihazlarda sorunsuz çalışmaktadırlar.

Genel olarak kullanım kolaylığı ve arayüz, performans, uygulamalar, erişilebilirlik ve güncellemeler temel alındığında iOS rakibi Android işletim sistemine oranla daha üstün durumda olmakla beraber sınırlamalar ve esneklik alanında rakibine oranla biraz daha geri planda kalmıştır. Kolay, kullanışlı ve kafa karıştırmayan bir işletim sistemi kullanmak istiyorsanız iOS, uğraşmayı, kurcalamayı seviyor ve teknolojinin içine dahil olmak istiyorsanız Android size daha uygun bir işletim sistemi olacaktır[5].

2. ANDROID PLATFORM

Bu bölümde gerçekleştirilen mobil veri indirme sisteminde kullanılan Android işletim sisteminin yapısı hakkında bilgiler verilmiştir. Mobil veri indirme sisteminde, Android tabanlı işletim sisteminin tercih edilme nedenleri açıklanmıştır.

2.1. Android İşletim Sistemi

Android işletim sistemi, mobil cihazlar için, uygulamalar ve uygulama sunucuları üzerine geliştirilen yazılım (middleware) içeren Linux tabanlı açık kaynak kodlu (Open Source) bir yazılımdır[6]. Açık kaynak kodlu olan Linux çekirdeğe sahip Android ücretsiz olarak dağıtılabilir, değiştirilebilir ve kullanılabilir.

2003 yılında Andy Rubin, Nick Sears, Rich Miner ve Chris White tarafından Android Inc. adlı bir şirket kurulmuştur. Bu şirket, çalışmaları sonucunda Linux çekirdeğe sahip bir işletim sistemini cep telefonlarında koşturmayı başarmıştır. Bu çalışmaların peşine ilk mobil işletim sistemi üzerinde çalışmalar yoğunlaştırılmıştır. Google 2005 yılında Android'i kendi bünyesine katmıştır. Android platformunu aktif bir şekilde geliştirmektedir. Ücretsiz olarak Android işletim sistemini cihazlarında kullanmak isteyen telefon kullanıcılarının, donanım üreticilerinin ve yazılım geliştiricilerinin hizmetine sunmaktadır.

5 Kasım 2007 tarihinde Google'ın başta olduğu otuz dört donanım, yazılım ve mobil operatör firması tarafından Open Handset Alliance adında bir birlik oluşturuldu. Bu birliğin amacı mobil cihaz sektöründe ortak, açık ve standart bir topluluk oluşturmaktır[7].

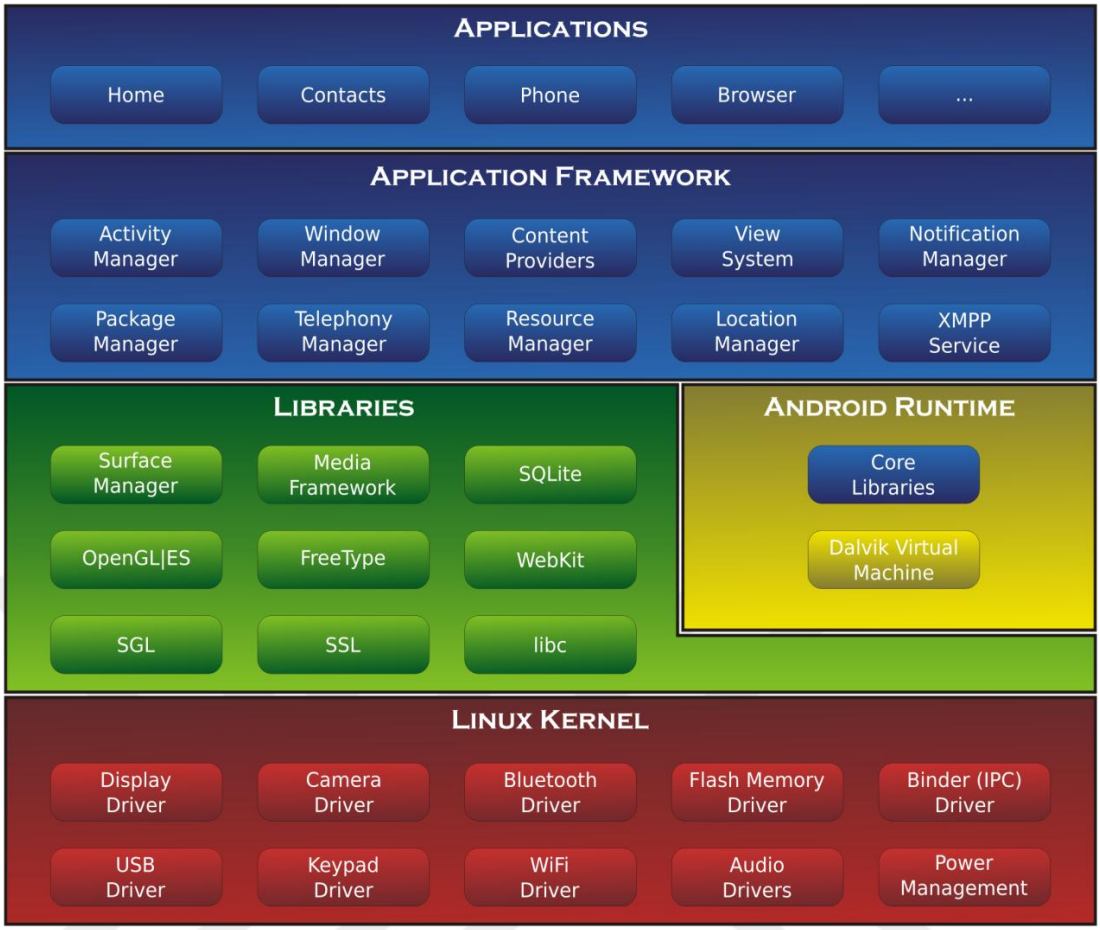
2008 yılında Google Android'in kaynak kodlarını açması ile telefon üreticileri bu işletim sistemini kendi ürettikleri cep telefonlarına dahil etmeye başladılar. Böylece HTC firması Android işletim sistemine sahip ilk cep telefonunu (HTC Dream) piyasaya sürmüştür[8]. Android 1.0 sürümünü kullanan HTC Dream; internet tarayıcısı, medya yürütücüsü ve Gmail hesap yöneticisini desteklemekle beraber uygulama mağazası olan Android Market'e de uyum sağlamıştır[4].

Android işletim sistemi birçok ayrı cihaz tarafından kullanılmakta ve geliştirilmektedir. Farklı ürünlerde kullanılan bu işletim sistemi bu sayede yeteneklerini de geliştirmektedir. Genel olarak kullanım alanları; akıllı telefonlar, tablet bilgisayarlar, televizyonlar, GPS cihazları, dizüstü bilgisayarlar, ortam oynatıcıları (DVD, VCD vb.), oyun konsolları, beyaz eşyalar ve elektronik kitap okuyucularıdır [9].

Android işletim sisteminin yazılım geliştirme kiti (SDK) Java programlama dilini kullanarak Android işletim sistemindeki uygulamaları geliştirmeye başlamak için gereken araç ve uygulama programlama arayüzlerini (API) sağlamaktadır. Java dışında C++,HTML5,JavaScript gibi yazılım dilleri ile de Android platformunda yazılım geliştirilmektedir. Google Android kurucuları en verimli programlamanın Java ile yapılmasını duyurmuşlardır. Tablo2.1'deve Şekil 2.1'de Android platformun mobil cihazlar için bir işletim sistemi, ara katman (middleware) ve bazı önemli uygulamalar içeren açık bir yazılım çatısı olduğu gösterilmiştir.

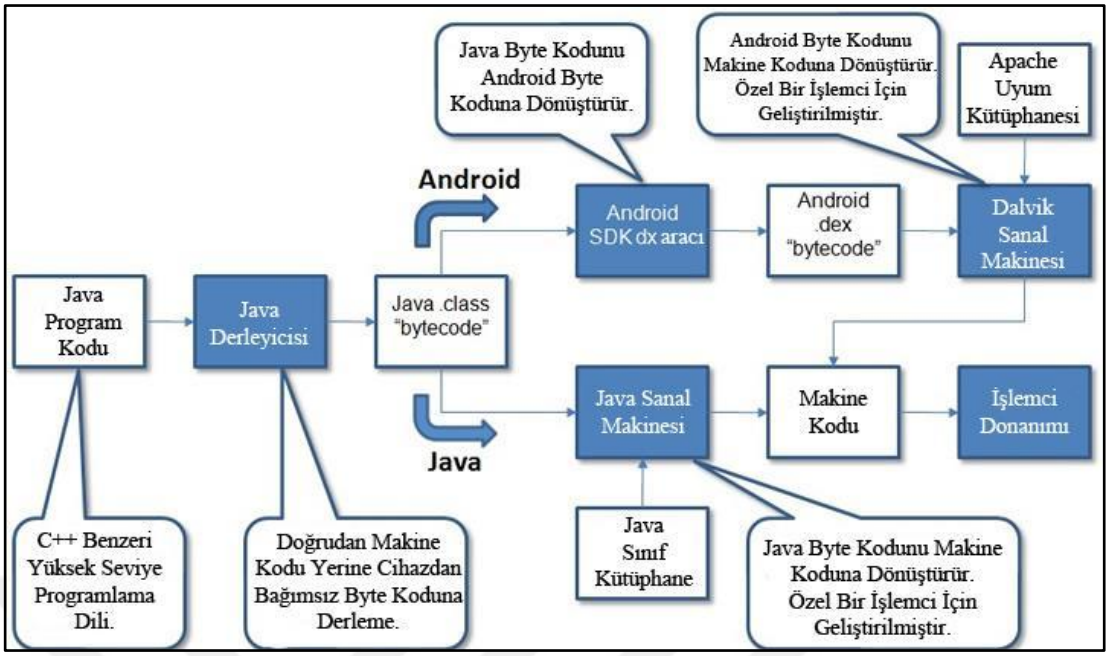
Tablo 2.1.Mobil Android yazılım yığını katmanları [10]

Uygulamalar (Mesajlaşma, E-mail, Tarayıcılar)
Uygulama Çerçevesi
Kütüphaneler ve Çalıştırılma Ortamı
Linux 2.6 Kernel (Çekirdeği) Versiyonu



Şekil 2.1. Android işletim sistemi mimarisi [11]

Android işletim sisteminde geliştirilen uygulamalar Java dili kullanılarak yazılır. Java ile yazılan kod derlendiği zaman bytecode'a dönüşür. Uygulama cihaza kurulmadan evvel .class uzantılı bytecode dosyaları .dex dosyalarına Java sanal makinesi tarafından çevirilir. Java sanal makinesi ile Dalvik sanal makinesi arasındaki en önemli fark Java sanal makinesi stack makine mimarisine sahipken Dalvik sanal makinesi ise kayıt tabanlı mimariye sahiptir. Dalvik sanal makinesi işlemcide daha az hafıza kullanarak çalışabilmesi için en uygun hale getirilmiştir[12].



Şekil 2.2. Donanım cihazıyla birleşmiş Android ve Java akış şeması [10]

Şekil 2.2'de söz konusu bir yazılım cihazında kullanılabilir bir program üretmek için Java ve Android işletim sisteminde, çeşitli parçaların nasıl bir arada çalışabileceği gösterilmiştir [13]. Biri Android işletim sisteminden giden ve diğeri daha klasik ve sade olan Java'dan geçen iki yol gösterilmektedir. Her ikisinde de bir işlemcide çalışabilen makine kodları üretilmekte fakat işlemciye ulaşmak için farklı yollar izlenmektedir. Java kodu makine diline Java Sanal Makinesi kullanarak oluşturmakta ancak Android işletim sistemi için kullanılan sanal makine Dalvik Sanal Makinesi'dir. Dalvik Sanal Makinesi sayesinde çalışabilir makine koduna çevrilir[14].

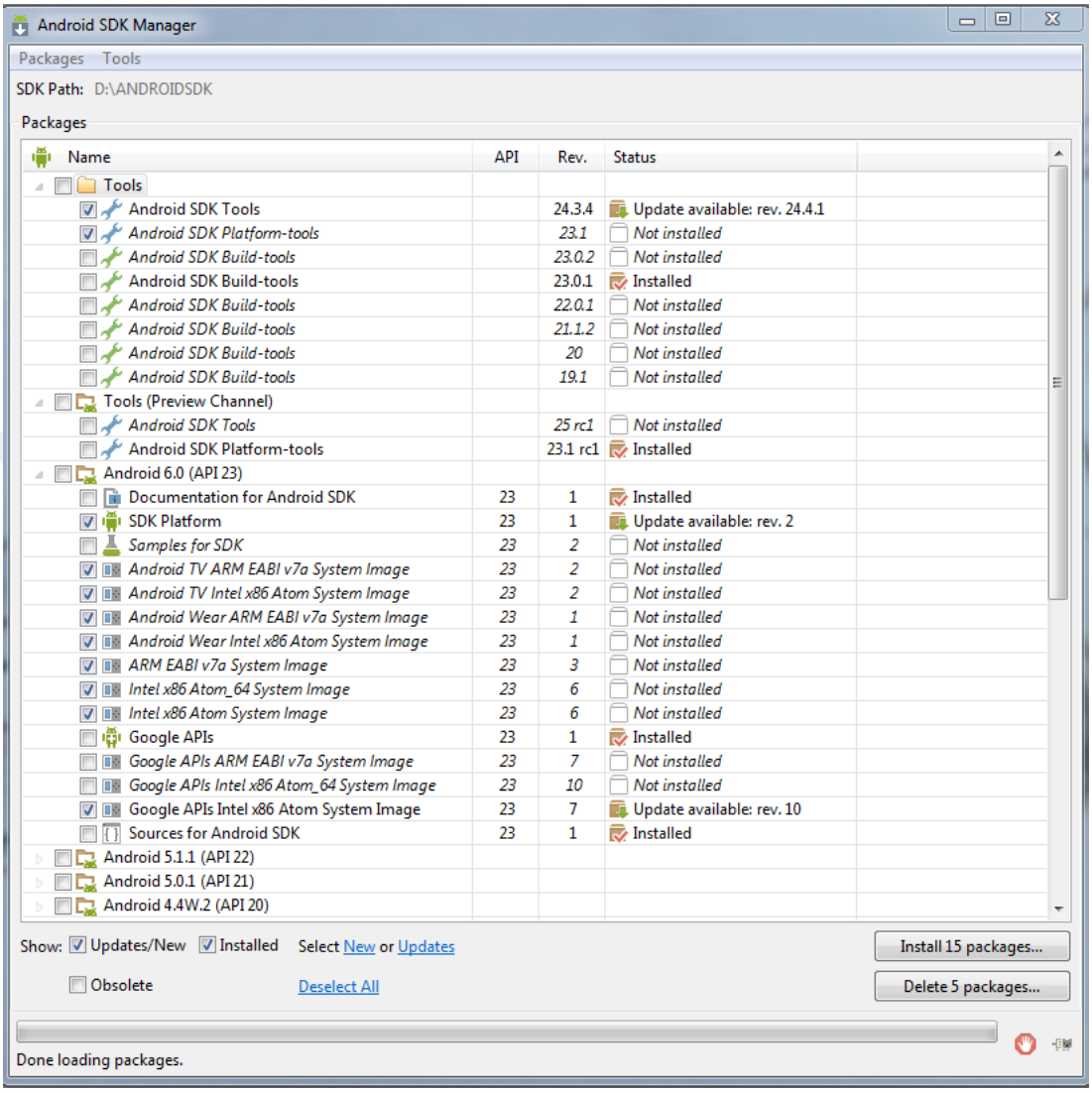
2.2. Android Yazılım Geliştirme

Android işletim sistemi ortamında yazılım geliştirebilmek için bilgisayarımızda bazı bileşenlerin olması gerekmektedir. Android SDK, ADB, Java ve kullanılacak olan yazılım geliştirme ortamının (IDE) sistemde bulunması zorunludur.

2.2.1. Android yazılım geliştirme kiti

Android SDK, bir uygulamanın geliştirilmesi, çalıştırılması ve test edilmesi için gerekli araçları sunmaktadır. AndroidSDK'nın kullanılabilmesi için Android SDK Manager aracı kullanılmaktadır. Herhangi bir Android işletim sistemi sürümüne

yazılım yapılabilmesi için ilgili sistemin yazılım bileşenlerinin SDK Manager aracılığıyla indirilmesi gerekmektedir.



Şekil 2.3. Androidsdk manager

Şekil 2.3'te görüldüğü gibi sisteme yüklenmiş ve yüklenmemiş paketlerin listeleri görüntülenmektedir. Bu listeleme Android işletim sistemi sürümlerine göre yapılmaktadır.

2.2.2. Android programlama dili

Java ilk olarak James Gosling tarafından geliştirilmeye başlandı ve Sun Microsystems tarafından 1995 yılında piyasaya sürüldü. Sun Microsystem firması (Oracle) Java'nın geliştirici firmasıdır. Java, bir programlama dili ve bilgi işlem platformudur. Yüksek verimli, çok işlevli, yüksek seviye, adım adım işletilen ve açık

kodlu yazılım dili olan Java platformdan bağımsız çalışan bir yazılım dilidir. Java ilk çıktığında daha çok küçük cihazlarda kullanılmak üzere tasarlanmış ortak bir platform dili olarak düşünülmüştü. Ancak platform bağımsızlığı özelliği ve standart kütüphane desteği C ve C++'tan çok daha üstün ve güvenli bir yazılım geliştirme ve işletme ortamı sunduğundan, hemen her yerde kullanılmaya başlanmıştır.

Java ile yazılan bir uygulama hem Windows hem MacOS hem de Linux işletim sistemi üzerinde çalışabilir. Bu uygulamaların çalışabilmesi için Java Sanal Makinesi'nin sisteme yüklü olması gerekmektedir. Sanal makine donanımdan bağımsız yazılım geliştirme ihtiyacına cevap verme amacıyla geliştirilen bir teknolojdir. Java`nın temel felsefesi olan "bir kere yaz, her yerde çalıştır" sanal makine sayesinde ortaya çıkmıştır. Sanal makineyi bir yönden bir tür hayali bir mikroişlemci gibi düşünebiliriz. Gerçek tüm mikroişlemciler (Intel Pentium, AMD Athlon, Sun Sparc vs) belirli bir grup komutu işlemek üzere tasarlanmıştır. Bu komutlara işlemcinin komut kümesi adı verilir. Örneğin x86 komut kümesi gibi. Tüm yazılımlar çalışabilmek için önce bu komut kümesine dönüştürülür, daha sonra işlemci bu komutları sıra ile gerçek işlemci komutlarına dönüştürüp işletir. Java Sanal makinesi de bytecode komut kümesini tıpkı işlemci gibi adım adım işletir. Java`nın yorumlayacı(interpreted) bir dil olarak adlandırılmasının nedeni budur.

Java'nın sık kullanılan önemli özelliklerinden biri de çöp toplayıcısı özelliğinin olmasıdır. Çöp toplama teknolojisi Java` dan önce de var olan ama Java ile adını duyurmuş ve yaygın olarak kullanılmaya başlanmış bir kavramdır. C++, C gibi dillerin en büyük engellerinden birisi dinamik bellek yönetimidir. Yazılımda işaretçi (pointer) kullanarak dinamik olarak bellek ayırdıktan sonra o bellek ile işiniz bittiğinde mutlaka ayrılan belleği bellek yöneticiye özel metodlar yardımıyla (delete, free vs.) iade etmeniz gerekir. Yoksa bellek sızıntısı (Memory Leak, MLK) oluşur ve bu bir süre sonra yazılımın ve işletim sisteminin beklenenden farklı davranmasına yol açabilir. Sızıntıların tespiti oldukça güçtür ve bulunması zor hatalara yol açar. Bu nedenle bugünün tüm büyük C ve C++ yazılımları işletim sistemleri de dahil olmak üzere az da olsa bellek sızıntısı içerir.

Çöp toplayıcı sayesinde Java`da bir nesne oluşturulduktan sonra o nesne ile işiniz bittiğinde hiçbir şey yapmanız gerekmez. Sanal makine akıllı bir biçimde

kullanılmayan bellek bölümlerini belirli aralıklarla ya da uyarlamalı metodlarla otomatik olarak temizler ve sisteme iade eder. Bu işleme çöp toplama, ya da "garbage collection" adı verilir. Çöp toplama sistemlerinin yapısı oldukça karmaşıktır ve geçen yıllar içinde büyük gelişmeler kaydedilmiştir. Çöp toplayıcının varlığı Java` da bellek sızıntısı olmayacağı anlamına gelmez ama bellek sızıntıları daha ender olarak ve farklı şekillerde karşınıza çıkar ve genellikle tedavi edilmesi daha kolaydır[15].

Java programlama dili günümüzde en çok kullanılan diller arasındadır. Nesneye yönelik bir programlama dilidir. Veritabanı yazılımları, web tabanlı yazılımlar, sim kart ve mobil cihaz yazılımlarında kullanılmaktadır. Android yazılımlarının geliştirilebilmesi için Java dilinin kullanılması gerekmektedir. Android üzerinde derlenmiş bir uygulamanın çalışabilmesi için sanal makine olarak Dalvik sanal makinesi kullanılır. Dalvik sanal makinesi kullanım ve erişim kolaylığı açısından tercih edilme sebebidir[16].

2.2.3. Android debug bridge (ADB)

AndroidDebug Bridge, Android SDK ile birlikte gelen Google'ın komut kümesidir. ADB içinde yer alan komutlar ile Android'e uygulama yükleme, kaldırma vs. gibi işlemleri gerçekleştirebiliriz. Android cihazı herhangi bir şekilde kontrol etmek istiyorsak bu işlemler ADB ile mümkündür. USB'den bağlı olan cihaza komut işletim dizini ile işlemler gerçekleştirmek mümkündür[17]. Üç ayrı bileşeni vardır. Bunlar; istemci(client), sunucu ve daemon'dır. İstemci; geliştirmenin yapıldığı ortamda çalışır ve ADB deamon emülatör üzerinde çalışan taraftır. Sunucu; arka planda çalışmaktadır ve istemci (client) ve ADB daemon arasındaki iletişimi yönetmektedir. Deamon ise emülatör veya cihazlarda arka planda çalışan işlemdir.

2.2.4. Android programlamada aktivite ve düzen yapıları

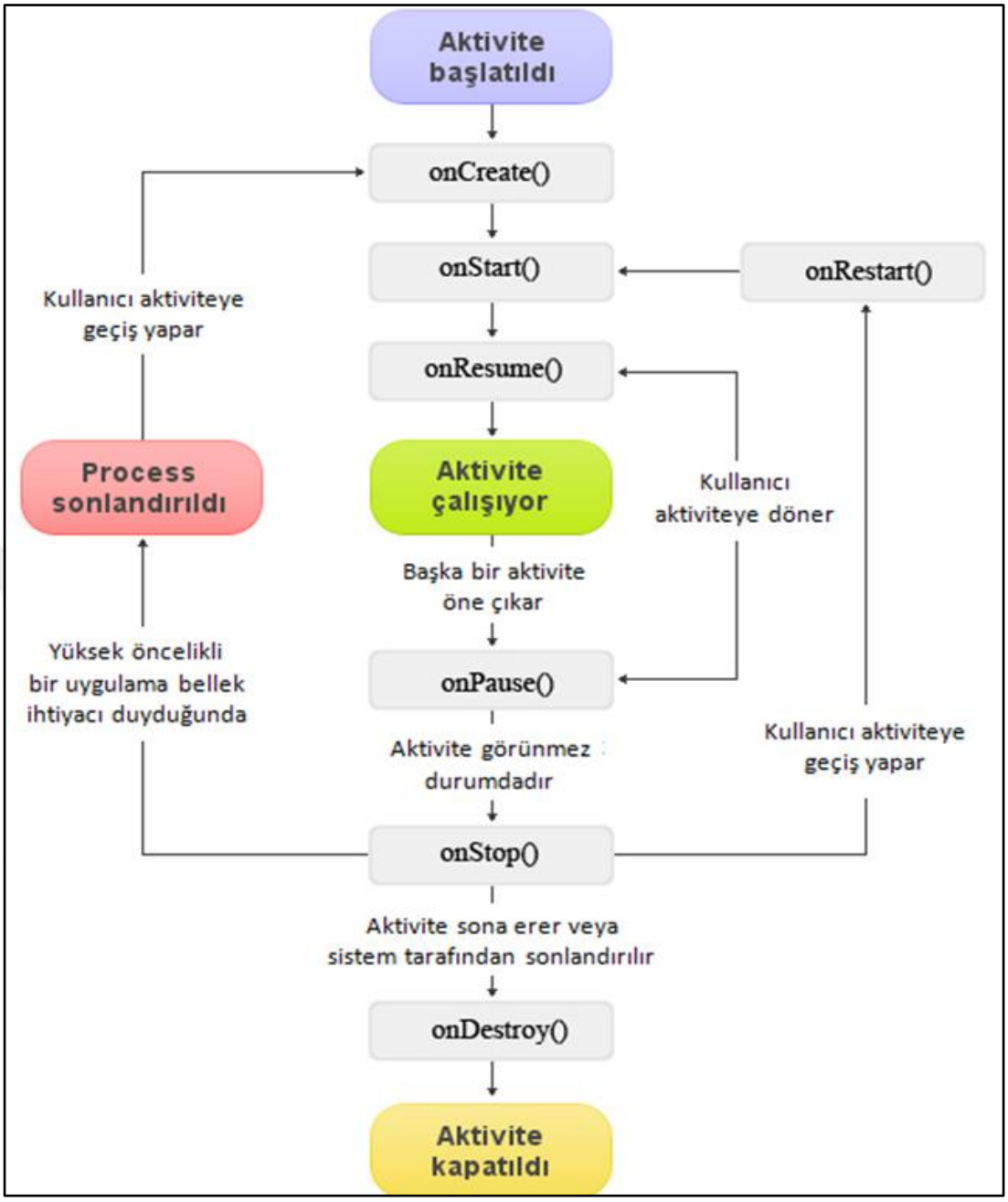
Aktiviteler, uygulamanın görsel olarak sunulma kısmıdır. Aktivite sınıfı tarafından oluşturulan nesneler uygulama içerisinde kullanılan ekranları temsil eder. Kullanıcılar uygulamada ekranlar arası geçiş yaparken aktivite değişikliği yaparlar. Aktivite çalışmasının bir yaşam döngüsü vardır. Bu fonksiyonları tetiklememize ihtiyaç yoktur çünkü Android işletim sistemi tarafından otomatik olarak

gerçekleştirilir[18]. Aktivite yaşam döngüsü; onStart, onCreate, onResume, onPause, onStop ve onDestroy aktiviteleri şeklindedir. Aktivite yaşam döngüsü içerisinde başka metodları da çağırmaktadır.

Bu metodları sıralarsak;

- onCreate(): Aktivite ilk oluşturulduğunda çağrılan metoddur. Aktivite herhangi bir nedenden ötürü yok edilmezse (bellek ihtiyacı ya da finish metodu) bir daha çalıştırılamaz.
- onStart(): onCreate() metodu çalıştırılıp görsel öğeler oluşturulduktan sonra çağrılır.
- onResume(): Aktivite herhangi bir sebepten dolayı durdurulduysa (başka bir ekrana geçiş) tekrar ekrana geldiğinde bu metot harekete geçer. Aktivite çalıştırılmadan önce harekete geçen son metottur. Bu metottan sonra belirlenen koda göre aktivite yaşamına başlar.
- onPause(): Aktivite bir sebepten dolayı arka plana atılırsa harekete geçer. Bugerituşuna basılmasından ya da sistemin bellek ihtiyacından kaynaklanabilir. Bu metot içerisinde anlık bilgilerin kaydedilmesi önerilir. ‘onPause()’işlemi hızlı çalışmak zorundadır, çünkü buradaki işlem bitmeden bir sonrakiaktiviteharekete geçmez.
- onStop(): Aktivite arka plana atıldığı anda işleme girer. Bu noktada iki seçenek vardır; ya kullanıcı yeni ekrandaki işini bitirip geri gelir ya da aktivite tamamen kapatılır. Kullanıcı geri geldiği takdirde onResume() metoduna geçilir ve döngüonStart()metodundan tekrar başlar. Eğer kullanıcı geri gelmeyecekseonDestroy()metodu çalıştırılır.
- onDestroy(): Aktiviteye ait bütün kaynaklar yok edilir. Aktivite bu noktada yaşam döngüsünü tamamlamıştır[19].

Şekil 2.4’te dikdörtgen biçimli kutuların her biri durum akışı süresince uygulamanızda çağrılan metotları temsil etmektedir [20]. Aynı şekilde, oval kutuların her biri de aktivitenin içinde olabileceği ana durumları temsil etmektedir.



Şekil 2.4. Android sisteminin aktivite yaşam döngüsü akış şeması [10]

Android işletim sistemininde, Android 3.0'dan sonra gelen özellikler içerisinde aktiviteler alt aktivitelere bölünebilir hale gelmiştir. Bu aktivitelere fragment denir. Bir aktivite içersinde birden fazla fragment kullanılabilir veya bir fragment birden fazla aktivite yapıları içersinde kullanılabilir. Bu fragmentler; 'blank activity', 'empty activity', 'full screen activity', 'google admob ads activity', 'google maps activity', 'login activity', 'master/detail flow', 'navigation drawer activity', 'scrolling

activity', 'setting activity' ve 'tabbed activity''dir. Bu fragmentler programcıya kolaylık sağlamaktadır.

'Swipe Views (yürüyen görüntüler)', Android üzerinde yan gezinti yapılarından biridir. Sayfalar arasında geçiş yapmak için kullanılır. Sayfaları sağa veya sola kaydırarak rahatlıkla geçiş yapılmasını sağlar.

Bir Android projesi derlenerek '.apk' uzantılı paketler haline dönüştürülür ve cihazlara yüklenebilir hale getirilir.

2.2.5. Android programlama proje dosyaları

Android Studio üzerinde bir proje oluşturulduğunda projenin gerekli dosyaları bulunmaktadır. Bu dosyalar Android cihaza uygulama yüklendiğinde çalışabilmesi için gerekli dosyalardır.

AndroidManifest.xml; bir Android projesinin içerisinde mutlaka bulunması gereken dosyadır. Bu dosya .xml formatta olduğu için hem kullanıcı tarafından rahatlıkla okunur hem de cihaz tarafından okunur. AndroidManifest.xml dosyası uygulama ile ilgili önemli bilgileri bulundurur. Cihazda uygulamaya verilen özel izinler ve uygulama sırasında hangi dosyalara erişeceğine dair bilgileri AndroidManifest.xml içerisinde tanımlarız. AndroidManifest.xml içerisinde manifest etiketi altında uygulamamızın paket adı, versiyon numaraları özelliklerini barındırır. Bu bölgeye girilecek etiketler sistemin nasıl ve neye göre çalışacağını belirler. Versiyon kodu, sistem paketlerinin tanımlanmasına yardımcı olur. Android cihaza bir veri geldiğinde ya da bir cihaz bağlandığında sistemin uyarılmasını istiyorsak "intent" yapılarını "manifest" dosyasında tanımlarız. Uygulamanın dikey mi yoksa yatay mı çalışabileceğini "screenOrientation" komutu ile belirleriz ve nasıl tanımlandıysa uygulama hep o ekran durumunda çalışır. Buna dosya üzerinden izin vermenizde mümkündür. İnternet üzerinden veri alış verişi yapan bir uygulamanız olduğunu düşünelim, ilk önce AndroidManifest.xml dosyasında "<uses-permission Android:name="Android.permission.INTERNET"/>" komutunu ekleyerek internet iznini vermiş oluyoruz. Aksi takdirde internet ile ilgili işlemleri gerçekleştirememekteyiz. "Application" etiketi altında uygulamamızın temel

değişkenlerini tanımlayabiliyoruz. Uygulamaya ait logo, tema, simgeve açıklamalar burada yapılmaktadır.

Android içerisinde bulunan diğer dizinlerden bir tanesi de “res” dizinidir. Bu dizin içerisinde uygulama kaynakları bulunmaktadır. Sistem için gerekli olan resim, değerler, xml dosyaları ve renkler gibi kaynakları bulundurur.“Res” dizini altındaki dizinler color, drawable, layout, menu, mipmap, raw, values ve xml dizinleridir. Color dizini renk tanımlamaları içeren .xml dosyalarını bulundurur. Sistemde kullanılan renk dosyaları buradan alınır. Drawable dizini JPEG, PNG ve diğer formatlardaki imge ve imge olarak gösterilebilecek dosyaları içeren .xml dosyalarını bulundurur. Layout dizini her bir ekranda gösterilecek olan tasarım dosyalarını bulundurur. Ekranda tasarlanan arayüzler bu dizinden çekilir. Menu dizininden menüler ile ilgili .xml dosyaları bulunur. Menü tuşuna basıldığında bir menü çıkmasını istiyorsak menü elemanları bu klasör içerisinde saklanır. Mipmap dizininde ise tema ve ikon için imge verilerini tutar. Values diziniuygulamalarda kullanılan sabit değişkenlerin saklandığı yerdir. Raw dizininde isteğe bağlı dosyaları tutan klasörler bulunmaktadır (mp3, ogg). Uygulamamıza dışarıdan dosyalar okutmak ve yüklemek istiyorsak bu dosyaya ekleriz ve bu dosyadan okunmasını sağlarız. Gradle içerisinde Java, resim, font, ve xml içeren “.aar” dosyalarını kullanılabilmektedir. Build.gradle dosyası içerisinde projeye ait kütüphane, en düşük sdk versiyonu, hedef sdk versiyonu ve derleme sdk versiyonu bulunur.

3. SAYISAL TAKOGRAF

Takograf cihazı, karayolu araçlarının hareketinin ve sürücülerinin belli çalışma sürelerinin detaylarını otomatik veya yarı otomatik olarak göstermek ve kaydetmek için karayoluyla taşımacılık yapan araçlara monte edilmesi tasarlanan kayıt cihazıdır [21].

Takograf cihazı;

- Araç kullanım sürelerini denetler.
- Araç hızını denetler.
- Kanunun belirttiği sınırlar içinde aracın kullanımını denetler.
- Denetlediği kayıtları hafızasında tutar.
- Hafızasındaki kayıtları grafik veya yazılı rapor olarak verir [21].

Sayısal takograf uygulaması ile profesyonel sürücülerin davranışlarına/faaliyetlerine ilişkin daha fazla ve güvenilir veri toplayarak, karayolu güvenliğinin artırılması amaçlanmaktadır.Birleşmiş Milletler Avrupa Ekonomik Komisyonu (BM-AEK) "Uluslararası Karayolu Taşımacılığı Yapan Araç Mürettebatının Çalışmasıyla ilgili Avrupa Anlaşması (AETR)" nın 13. maddesi uyarınca, Avrupa Birliği'ne üye olmayan fakat AETR'ye taraf olan, içerisinde Türkiye'nin de bulunduğu ülkelere uluslararası karayolu taşımacılığında, sayısal takograf uygulamasına geçilmesi konusunda bir geçiş süreci tanınmıştır[21].

Sayısal takograf uygulamasının zorunlu olduğu, AETR anlaşmasına taraf ülkeler şunlardır: Almanya, Andora, Arnavutluk, Avusturya, Azerbaycan, Belarus, Belçika, Bosna Hersek, Bulgaristan, Çek Cumhuriyeti, Danimarka, Ermenistan, Estonya, Finlandiya, Fransa, Hırvatistan, Hollanda, İngiltere, İrlanda, İsveç, İsviçre, İspanya, İtalya, Karadağ, Kazakistan, Kıbrıs Rum Kesimi, Letonya, Lihteştayn, Litvanya, Lüksemburg, Macaristan, Makedonya, Malta, Moldova, Monako, Norveç, Özbekistan, Polonya, Portekiz, Romanya, Rusya, San Marino, Sırbistan, Slovakya, Slovenya, Türkiye, Türkmenistan, Ukrayna, Yunanistan [22].

AB ülkelerinde 561 / 2006 numaralı düzenleme gereğince, Mayıs 2006'dan itibaren yeni tescil edilen ve 3,5 tondan büyük tüm HGV'ler ile sürücü dahil 9'dan fazla yolcu taşıyan araçlarda sayısal takograf bulunması gerekmektedir. Sayısaltakograflardan elde edilen tüm araç verilerinin ve sürücü kartlarından elde edilen tüm sürücü verilerinin mevzuata uygun olarak arşivlenmesi zorunludur. Benzer şekilde ülkemizde de T.C. Ulaştırma, Denizcilik ve Haberleşme Bakanlığı, Karayolu Düzenleme Genel Müdürlüğü'nün yayınlamış olduğu 06.06.2014 tarihli genelgede de hükümler belirtilmiştir.

(EC) No. 561/2006, (EEC) No. 3821/85 ve (EC) No. 1360/2002 yönetmeliklerine uygun olarak aşağıdaki hükümler geçerlidir:

- Sürücü kartı verileri en geç 28 günde bir indirilmeli ve elektronik olarak arşivlenmelidir.
- Sayısal takograf araç verileri en geç 3 ayda bir indirilmeli ve elektronik olarak arşivlenmelidir.
- Tüm araç ve sürücü verileri en az bir yıl süreyle kayıt altında tutulmalı ve gerekli görüldüğünde yetkililerin inceleyebilmesi için ulaşılabilir olmalıdır[23].

Yasal mevzuatların gerekliliklerinin sağlanmasıyla firmaların yetkilileri indirilen verileri inceleyebilir, yönetmeliklerce belirtilen dinlenme ve sürüş sürelerinin uygunluğunu teyit edebilir veya bir ihlal söz konusuysa ilgili sürücüyü bilgilendirip uyarabilir.

3.1. Sayısal Takografa Ait Kart Çeşitleri

3.1.1. Sürücü kartı

Belirli bir sürücüye verilen takograf kartıdır. Sürücü kartı sürücüyü tanıtır ve sürücü faaliyeti verisinin hafızaya alınmasına yarar. Şekil 3.1'de örnek sürücü kartı gösterilmiştir.



Şekil 3.1. Sürücü kartı ön ve arka yüz [24]

3.1.2. Şirket kartı

Takograf cihazı takılı araç sahiplerine veya hamillerine verilen bir takograf kartıdır. Şirket kartı, şirketi tanıtır ve şirket tarafından kilitlenen veya herhangi bir şirket tarafından kilitlenmeyen takograf cihazında hafızaya alınan verinin görüntülenmesini, yüklenmesini ve yazdırılmasını sağlar. Şekil 3.2'de örnek şirket kartı gösterilmiştir.



Şekil 3.2. Şirket kartı ön ve arka yüz [24]

3.1.3. Denetimkartı

Ulusal yetkili denetim otoritesine verilen takograf kartıdır. Denetim kartı denetim otoritesini tanıtır ve veri hafızasında veya sürücü kartlarında hafızaya alınmış veriyi okumak, yazdırmak ve/veya indirmek için bu veriye erişmek amacıyla kullanılır. Şekil 3.3'de örnek denetim kartı gösterilmiştir.



Şekil 3.3. Denetim kartı ön ve arka yüz [24]

3.1.4. Servis kartı

Bir takograf cihazı imalatçısına, bir araç imalatçısına veya servis ve montajcıya ülkenin yetkili kuruluşları tarafından verilen bir takograf kartıdır. Servis kartı kart hamilini tanıtır ve takograf cihazının testini, ölçümlemesini ve/veya indirilmesini sağlar. Şekil 3.4'te örnek servis kartı gösterilmiştir.



Şekil 3.4. Servis kartı kartı ön ve arka yüz [24]

4. SAYISAL TAKOGRAF VERİ İNDİRME VE ANALİZ UYGULAMASI

4.1. Sayısal Takograf Veri İndirme

4.1.1. İndirme yöntemi

Takograf cihazı (VU) istendiğinde veri belleğinden veya bir sürücü kartından harici hafıza ortamına indirme bağlantısı aracılığıyla veri indirebilmektedir. Takograf cihazı, kendi hafızasındaki verileri harici ortama indirme işlemini yapabilmesi için yönetmelik tarafından belirtilen servis kartı, denetim kartı veya şirket kartından birinin takograf cihazında takılı olması gerekmektedir. Takograf cihazına takılı durumda olan sürücü kart verilerini indirebilmek için ise diğer kartlara ihtiyaç duymadan veri indirme bağlantısı aracılığıyla bu işlem gerçekleştirilebilmektedir. Veri indirme işlemi yapılırken şu adımlar izlenmelidir:

- Takograf cihazı kart yuvasına gerekli kartın takılması.
- Veri indirme cihazının veya veri indirme yazılımının takograf bağlantı elemanına bağlanması.
- Veri indirme cihazı veya veri indirme yazılımı ile takograf arasındaki bağlantının kurulması.
- Veri indirme cihazından veya veri indirme yazılımından indirilecek verinin seçilmesi ve talebin takograf cihazına gönderilmesi.
- Veri indirme oturumunun kapatılması.

4.1.2. Veri indirme protokolü

Protokol, veri indirme cihazının alıcı (master) rolünü ve takografın da verici (slave) rolünü aldığı bir tek taraflı veri akışı (master-slave) temelinde yapılandırılmıştır. Mesaj yapısı, tipleri ve akışı temelde Şifre Kelime Protokolü 2000 (KWP) üzerinde kurulmuştur. Uygulama katmanı temelde ISO 14229-1'in en son taslağı üzerinde kurulmuştur [25].

4.1.2.1. Mesaj yapısı

Veri indirme cihazı ile takograf arasında alış veriş yapılan bütün mesajlar üç kısımdan oluşan bir yapıyla biçimlendirilir. Tablo4.1'debu üç kısım gösterilmiştir.

Tablo 4.1. Veri indirme işlemi mesaj yapısı [25]

Ön Bilgi				Veri Alanı					Denetim Toplamı
FMT	TGT	SRC	LEN	SID	DATA	…	…	…	CS
4 Bayt				Azami 255 Bayt					1 Bayt

- Bir biçim baytı (FMT), bir hedef baytı (TGT), bir kaynak baytı (SRC) ve uzunluk baytından oluşan bir ön bilgi
- Bir hizmet tanıtıcısı (SID) ve isteğe bağlı bir teşhis oturum baytını (DS) veya isteğe bağlı bir aktarım parametresi baytını (TRTP veya TREP) içeren değişken sayıda veri baytlarından oluşan veri alanı
- Bir denetim toplamı baytından (CS) oluşan denetim toplamı

4.1.2.2. Mesaj tipleri

Takograf cihazı ile veri indirme cihazı arasındaki veri indirme iletişim protokolü sekiz farklı mesaj tipinin alış verişi Şekil 4.1'de belirtilmiştir. Bu şekilde araç ile veri indirme cihazı arasındaki haberleşmenin bilgi, veri ve denetim toplamı baytlarını görmek mümkündür.

Mesaj Yapısı	Azami 4 bayt Ön bilgi				Azami 255 bayt Veri			1 Bayt Denetim Toplamı
IDE -> <-VU	FMT	TGT	SRC	LEN	SID	DS_TRTP	VERİ	CS
İletişimi başlat talebi	81	EE	F0	[redacted]	81	[redacted]	[redacted]	E0
İletişimi başlat olumlu cevabı	80	F0	EE	03	C1	[redacted]	EA, 8F	9B
Teşhis oturumunu başlat talebi	80	EE	F0	02	10	81	[redacted]	F1
Teşhis başlat olumlu cevabı	80	F0	EE	02	50	81	[redacted]	31
Bağlantı kontrol hizmeti Baud hızını doğrula (aşama 1)								
9600 Bd	80	EE	F0	04	87	[redacted]	01,01,01	EC
19200 Bd	80	EE	F0	04	87	[redacted]	01,01,02	ED
38400 Bd	80	EE	F0	04	87	[redacted]	01,01,03	EE
57600 Bd	80	EE	F0	04	87	[redacted]	01,01,04	EF
115200 Bd	80	EE	F0	04	87	[redacted]	01,01,05	F0
Baud hızını doğrula olumlu cevabı	80	F0	EE	02	C7	[redacted]	01	28
Geçiş Baud hızı (aşama 2)	80	EE	F0	03	87	[redacted]	02,03	ED
Karşıya yükleme talebi	80	EE	F0	0A	35	[redacted]	00,00,00,00,00,FF,FF, FF,FF	99
Karşıya yükleme talebi olumlu cevabı	80	F0	EE	03	75	[redacted]	00,FF	D5
Veri aktarım talebi								
Genel Bilgi	80	EE	F0	02	36	01	[redacted]	97
Faaliyetler	80	EE	F0	06	36	02	Tarih	CS
Olaylar ve hatalar	80	EE	F0	02	36	03	[redacted]	99
Detaylı hız	80	EE	F0	02	36	04	[redacted]	9A
Teknik veri	80	EE	F0	02	36	05	[redacted]	9B
Kart indirme	80	EE	F0	02	36	06	[redacted]	9C
Veri aktarım olumlu cevabı	80	F0	EE	LEN	76	TREP	Veri	CS
Aktarım çıkışı talebi	80	EE	F0	01	37	[redacted]	[redacted]	96
Aktarım çıkışı talebi olumlu cevabı	80	F0	EE	01	77	[redacted]	[redacted]	D6
İletişimi sonlandır talebi	80	EE	F0	01	82	[redacted]	[redacted]	E1
İletişimi sonlandır olumlu cevabı	80	F0	EE	01	C2	[redacted]	[redacted]	21
Alt-mesajı teyit et	80	EE	F0	LEN	83	[redacted]	Veri	CS
Olumsuz cevaplar								
Genel Ret	80	F0	EE	03	7F	Sid Req	10	CS

Şekil 4.1. Mesaj tipleri [25]

4.1.2.3. Mesaj akışı

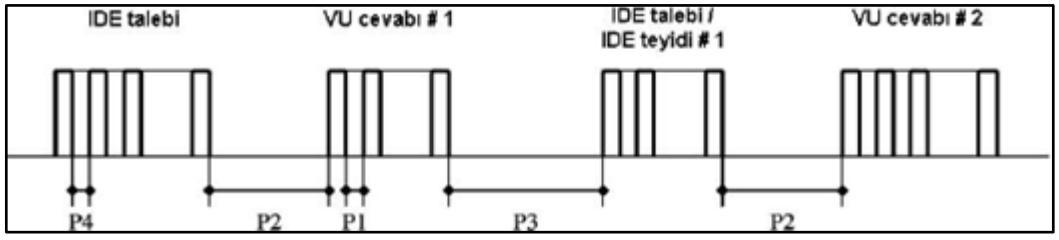
Normal bir veri indirme işlemi sırasında örnek bir mesaj akışı Şekil 4.2'de gösterildiği gibidir. Tüm iletişim adımları istek ve cevap üzerine kurulmuştur. Yapılan isteğe olumlu cevap geldiği takdirde iletişim oturumu bir sonraki adımları yapmak üzere akışına devam etmektedir.

IDE		VU
İletişimi başlat talebi	⇨	
	⇦	Olumlu cevap
Teşhis oturumu başlat talebi	⇨	
	⇦	Olumlu cevap
Karşıya yükleme talebi	⇨	
	⇦	Olumlu cevap
Veri aktarım talebi genel bilgi	⇨	
	⇦	Olumlu cevap
Veri talebi # 2	⇨	
	⇦	Olumlu cevap # 1
Alt-mesaj # 1 teyidi	⇨	
	⇦	Olumlu cevap # 2
Alt-mesaj # 2 teyidi	⇨	
	⇦	Olumlu cevap # m
Alt-mesaj # m teyidi	⇨	
	⇦	Olumlu cevap (veri alanı < 255 bayt)
Alt-mesaj teyidi (ihtiyari)	⇨	
	...	
Veri aktarım talebi # n	⇨	
	⇦	Olumlu cevap
Aktarım çıkışı talebi	⇨	
	⇦	Olumlu cevap
İletişimi sonlandır talebi	⇨	
	⇦	Olumlu cevap

Şekil 4.2. Mesaj akışı [25]

4.1.2.4. Zamanlama

Normal bir veri indirme işlemi sırasında Şekil 4.3'te gösterilen zamanlama parametreleri kullanılmaktadır.



Şekil 4.3. Mesaj akışı zamanlama [25]

Zamanlama parametrelerinden P1, takograf cevabı için baytlar arası zamanı ifade eder. P2, veri indirme cihazı talebinin sonu ile takografın cevabının başı veya veri indirme cihazının teyidinin sonu ile bir sonraki takograf cevabının başı arasındaki zamanı ifade eder. P3, takograf cihazı cevabının sonu ile veri indirme cihazı talebinin

veya teyidinin başı arasındaki zamanı ifade eder. P4, veri indirme cihazının talebi için baytlar arası zamanı ifade ederken P5 ise kart indirme için P3'ün uzatılmış zaman değerini ifade etmektedir.

Zamanlama parametreleri için izin verilen değerler Şekil 4.4'de gösterilmiştir.

Zamanlama parametresi	Alt sınır değeri (ms)	Üst sınır değeri (ms)
P1	0	20
P2	20	1000 (*)
P3	10	5000
P4	5	20
P5	10	20 dakika
(*) VU, "talep doğru alındı, cevap bekletiliyor" anlamına gelen bir kod içeren bir olumsuz cevap ile cevap verirse, bu değer P3'ün aynı üst sınır değerine kadar uzatılır.		

Şekil 4.4. Zamanlama için alt ve üst sınır değerler [25]

4.1.2.5. Kart indirme

Kart verisi indirme basamakları kart yapısından dolayı bazı gereklilikler doğurmaktadır. Kart verisi indirilirken, kartın ortak bilgilerini, ICC ve IC EF'lerini indirmek gerekmektedir. Bu bilgi isteğe bağlıdır ve sayısal bir imza ile güvenli hale getirilmez. Kart belgesi (Card_Certificate) ve CA belgesi EF'leri indirilmesi zorunlu olan diğer temel dosyalardır. Genel olarak kısaca bir sürücü kartı indirilirken olaylar, hatalar, sürücü faaliyet, kullanılan araçlar, yerler, kontrol faaliyet verisi ve özel koşul verilerinin indirilmesi zorunludur. Bir sürücü kartı Şekil 4.5'de belirtilen dosya yapısı ve erişim koşullarına sahiptir.

Dosya	Dosya ID	Erişim koşulları		
		Oku	Güncelle	Şifrelenmiş
MF	3F00			
├─ EF ICC	0002	ALW	NEV	Hayır
├─ EF IC	0005	ALW	NEV	Hayır
└─ DF Takograf	0500			
├─ EF Uygulama_Tanıtımı	0501	ALW	NEV	Hayır
├─ EF Kart_Belgesi	C100	ALW	NEV	Hayır
├─ EF CA_Belgesi	C108	ALW	NEV	Hayır
├─ EF Tanıtım	0520	ALW	NEV	Hayır
├─ EF Kart_İndirme	050E	ALW	ALW	Hayır
├─ EF Sürücü_Belgesi_Bilgisi	0521	ALW	NEV	Hayır
├─ EF Olaylar_Verisi	0502	ALW	PRO SM / AUT	Hayır
├─ EF Hatalar_Verisi	0503	ALW	PRO SM / AUT	Hayır
├─ EF Sürücü_Faaliyet_Verisi	0504	ALW	PRO SM / AUT	Hayır
├─ EF Kullanılan_Araçlar	0505	ALW	PRO SM / AUT	Hayır
├─ EF Yerler	0506	ALW	PRO SM / AUT	Hayır
├─ EF Geçerli_Kullanım	0507	ALW	PRO SM / AUT	Hayır
├─ EF Kontrol_Faaliyet_Verisi	0508	ALW	PRO SM / AUT	Hayır
└─ EF Özel_Koşullar	0522	ALW	PRO SM / AUT	Hayır

Şekil 4.5. Kart dosya kimlikleri [25]

4.1.3. İndirilen verilerin çözümlenmesi

Bir sayısal takograftan indirilen veriler genel bilgi, faaliyetler (aktiviteler), olaylar ve hatalar, detaylı hız ve teknik veriler olmak üzere beş farklı sınıftadır. Bu veriler takograf yönetmeliklerince verilen sınıflandırmaya uygun olarak çözümlenebilmektedir. Şekil 4.6'da genel bilgi, Şekil 4.7'de faaliyet (aktivite), Şekil 4.8'de olay ve hata, Şekil 4.9'da detaylı hız ve son olarak Şekil 4.10'da teknik verilerin yönetmeliklerce belirtilen veri elemanı, uzunluk ve yorumlama bilgileri verilmiştir.

Veri elemanı			Uzunluk (Bayt)	Yorum
TarafÜlkeBelgesi			194	VU güvenlik belgeleri
VuBelgesi			194	
AraçTanıtımNumarası			17	Araç tanıtımı
AraçTescilTanıtımı				
	AraçTescil Ülkesi		1	
	AraçTescilNumarası		14	
OankiTarihSaat			4	VU o anki tarih ve saati
VuİndirilebilirlikSüresi				İndirilebilirlik Süresi
	AsgariİndirilebilirlikZamanı		4	
	AzamiİndirilebilirlikZamanı		4	
KartYuvalarıDurumu			1	VU'ya takılan kartların tipi
VuİndirmeFaaliyetiVerisi				Önceki VU indirmesi
	İndirmeZamanı		4	
	TamKartNumarası		18	
	ŞirketVeyaServisAdı		36	
VuŞirketKilitleriVerisi				Bütün şirket kilitleri hafızaya alınır. Kısım boşsa sadece KilitlerinSayısı = 0 gönderilir.
	KilitlerinSayısı		1	
	...		(98)	
	VU Şirket Kilitleri Kaydı	KilidiYerleştirmeZamanı	4	
		KilidiKaldırmaZamanı	4	
		ŞirketAdı	36	
		ŞirketAdresi	36	
		ŞirketKartNumarası	18	
	...			
VuKontrolFaaliyetiVerisi				VU'da hafızaya alınan bütün kontrol kayıtları. Kısım boşsa, sadece KontrollerinSayısı = 0 gönderilir.
	KontrollerinSayısı		1	
	...		(31)	
	VU Kontrol Faaliyeti Kaydı	KontrolTipi	1	
		KontrolZamanı	4	
		DenetimKartNumarası	18	
		İndirmeSüresiBaşlamaZamanı	4	
		İndirmeSüresiBitmeZamanı	4	
	...			
İmza			128	AraçTanıtımNumarası'ndan başlayarak son VuKontrolFaaliyetiKaydı'nın en son baytına kadar bütün verinin (belgeler hariç) RSA imzası.

Şekil 4.6. Genel bilgi verisi yorumlama [25]

Veri elemanı	Uzunluk (Bayt)	Yorum
GerçekZaman	4	İndirilen günün tarihi
KilometreSayacıDeğeriGeceyarısı	3	İndirilen günün sonunda kilometre sayacı değeri
VUKartTakmaÇıkarmaVerisi		
VUKartTakmaÇıkarmaKayıtlarıSayısı	2	
...	(129)	
VuKartTakmaÇıkarmaKaydı		Kartların takma çıkarma çevrimleri verisi. - Bu kısımda ulaşılabilir veri yoksa, sadece VuKartTakmaÇıkarmaKayıtlarıSayısı = 0 gönderilir. - Bir VUKartTakmaÇıkarmaKaydı 00:00 üzerinden (önceki gün içerisinde kartın takılması) veya 24:00 üzerinden (sonraki gün içerisinde kartın çıkarılması) geçiyorsa, ilgili iki gün boyunca bütün olarak görülmelidir.
KartHamiliAdı		
HamilSoyadı	36	
HamilİlkAdları	36	
TamKartNumarası	18	
KartınGeçerliliğininBittiğiTarih	4	
KartTakmaZamanı	4	
TakmaAnıAraçKilometreSayacıDeğeri	3	
KartYuvasıNumarası	1	
KartÇıkarmaZamanı	4	
ÇıkarmaAnıAraçKilometreSayacıDeğeri	3	
ÖncekiAraçBilgisi		
AraçTescilTanıtımı		
AraçTescil Ülkesi	1	
AraçTescilNumarası	14	
KartTakmaZamanı	4	
ManuelGirişİşareti	1	
...		
VuFaaliyetGünlükVerisi		
FaaliyetDeğişiklikleriSayısı	2	00:00'da yuvaların durumu ve indirilen gün için kaydedilen faaliyet değişiklikleri
...		
FaaliyetDeğişikliğiBilgisi	2	
...		
VuGünlükÇalışmaSüresiYerVerisi		
YerKayıtlarıSayısı	1	
...	(28)	
VUGünlükÇalışmaSüresiYerKaydı		İndirilen gün için kaydedilen yerler ile ilgili veri. Bu kısım boşsa, sadece YerKayıtlarıSayısı = 0 gönderilir.
TamKartNumarası	18	
YerKaydı		
GirişZamanı	4	
GirişTipiGünlükÇalışmaSüresi	1	
GünlükÇalışmaSüresiÜlkesi	1	
GünlükÇalışmaSüresiBölgesi	1	
AraçKilometreSayacıDeğeri	3	
...		
VuÖzelKoşulVerisi		
ÖzelKoşulKayıtlarıSayısı	2	İndirilen gün için kaydedilen özel koşullar verisi. Bu kısım boşsa, sadece ÖzelKoşulKayıtlarıSayısı = 0 gönderilir.
...	(5)	
ÖzelKoşulKaydı		
GirişZamanı	4	
ÖzelKoşulTipi	1	
...		
İmza	128	GerçekZaman'dan başlayarak son ÖzelKoşulKaydı'nın en son baytına kadar bütün verinin RSA imzası.

Şekil 4.7. Faaliyet (aktivite) verisi yorumlama [25]

Veri elemanı		Uzunluk (Bayt)	Yorum
VUHataVerisi			
VUHatalarıSayısı		1	
...		(82)	
VU Hata Kaydı	HataTipi	1	VU'da hafizaya alınan veya devam eden bütün hatalar. Bu kısım boşsa, sadece VUHatalarıSayısı = 0 gönderilir.
	HataKayıtAmacı	1	
	HataBaşlamaZamanı	4	
	HataBitmeZamanı	4	
	KartNumarasıSürücüYuvasıBaşlangıç	18	
	KartNumarasıYardımcıSürücüYuvasıBaşlangıç	18	
	KartNumarasıSürücüYuvasıBitim	18	
	KartNumarasıYardımcıSürücüYuvasıBitim	18	
...			
VUOlayVerisi			
VUOlaylarıSayısı		1	
...		(83)	
VuOlayKaydı	OlayTipi	1	VU'da hafizaya alınan veya devam eden bütün olaylar. Bu kısım boşsa, sadece VUOlaylarıSayısı = 0 gönderilir.
	OlayKayıtAmacı	1	
	OlayBaşlamaZamanı	4	
	OlayBitmeZamanı	4	
	KartNumarasıSürücüYuvasıBaşlangıç	18	
	KartNumarasıYardımcıSürücüYuvasıBaşlangıç	18	
	KartNumarasıSürücüYuvasıBitim	18	
	KartNumarasıYardımcıSürücüYuvasıBitimi	18	
	BenzerOlaylarSayısı	1	
...			
VUAşırıHızKontrolVerisi			
SonAşırıHızKontrolZamanı		4	Son aşırı hız kontrolü ile ilgili veri (veri yoksa varsayılan değer).
SonrakiIlkAşırıHız		4	
SonrakiAşırıHızSayısı		1	
VUAşırıHızOlayVerisi			
VUAşırıHızOlaylarıSayısı		1	
...		(31)	
VuAşırıHızOlayKaydı	OlayTipi	1	VU'da hafizaya alınan bütün aşırı hız olayları verisi. Bu kısım boşsa, sadece VUAşırıHızOlaylarıSayısı = 0 gönderilir.
	OlayKayıtAmacı	1	
	OlayBaşlamaZamanı	4	
	OlayBitmeZamanı	4	
	AzamiHızDeğeri	1	
	OrtalamaHızDeğeri	1	
	KartNumarasıSürücüYuvasıBaşlangıç	18	
	BenzerOlaylarSayısı	1	
...			
VUZamanAyarlamaVerisi			
VUZamanAyarlamaKayıtlarıSayısı		1	
...		(98)	VU'da hafizaya alınan bütün zaman ayarlama olayları (bir tam kalibrasyon çerçevesi dışında). Bu kısım boşsa, sadece VUZamanAyarlamaKayıtları-Sayısı = 0 gönderilir.
VuZaman Ayarlama Kaydı	EskiZamanDeğeri	4	
	YeniZamanDeğeri	4	
	ServisAdı	36	
	ServisAdresi	36	
	ServisKartNumarası	18	
...			
İmza		128	VuHatalarıSayısı'ndan başlayarak son ZamanAyarlamaKaydı'nın en son baytına kadar bütün verinin RSA imzası.

Şekil 4.8. Olay ve hata verilerini yorumlama [25]

Veri elemanı	Uzunluk (Bayt)	Yorum
VuDetaylıHızVerisi		
HızBloklarıSayısı	2	
...		
VuDetaylıHızBloğu: HızBloğuBaşlamaTarihi	4	VU'da hafızaya alınan bütün detaylı hız (Araç hareketi boyunca dakikada bir hız bloğu). Dakikada 60 hız değeri (Saniyede bir)
VuDetaylıHızBloğu: SaniyeBaşınaHız	60	
...		
İmza	128	HızBloklarınınSayısı'ndan başlayarak son HızBloğu'nun en son baytına kadar bütün verinin RSA imzası.

Şekil 4.9. Detaylı hız verisini yorumlama [25]

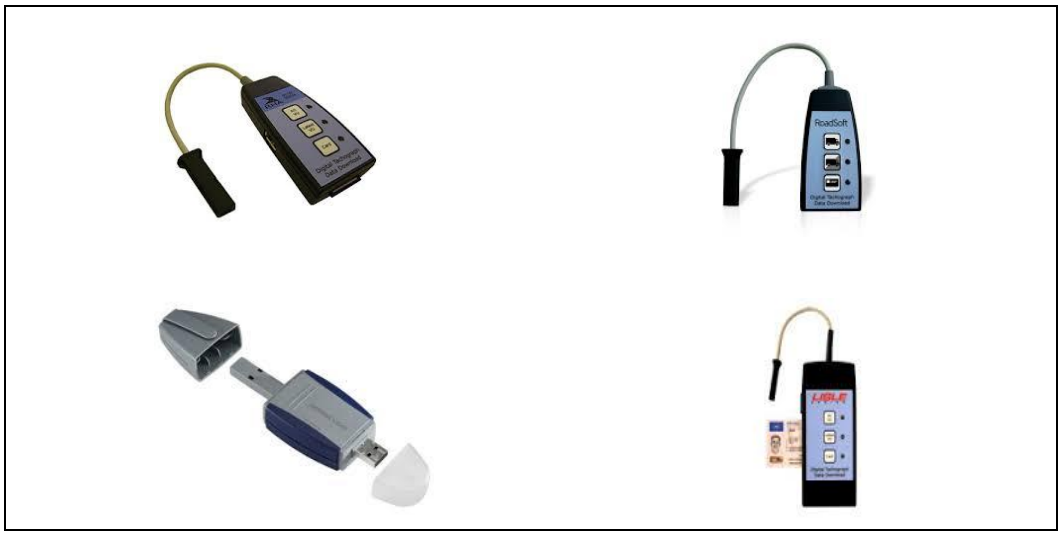
Veri elemanı	Uzunluk (Bayt)	Yorum
VuTanıtımı		
VuİmalatçıAdı	36	
VuİmalatçıAdresi	36	
VuParçaNumarası	16	
VuSeriNumarası	8	
VuYazılımTanıtımı		
VuYazılımSürümü	4	
VuYazılımKurmaTarihi	4	
VuİmalatTarihi	4	
VuOnayNumarası	8	
EşlenenSensör		
SensörSeriNumarası	8	
SensörOnayNumarası	8	
SensörİlkEşlemeTarihi	4	
VuKalibrasyonVerisi		
VuKalibrasyonKayıtlarıSayısı	1	
...	(167)	
VuKalibrasyonKaydı: KalibrasyonAmacı	1	VU'da hafızaya alınan bütün kalibrasyon kayıtları.
VuKalibrasyonKaydı: ServisAdı	36	
VuKalibrasyonKaydı: ServisAdresi	36	
VuKalibrasyonKaydı: ServisKartıNumarası	18	
VuKalibrasyonKaydı: ServisKartınınGeçerliliğininBittiğiTarih	4	
VuKalibrasyonKaydı: AraçTanıtımNumarası	17	
VuKalibrasyonKaydı: AraçTescilTanıtımı		
VuKalibrasyonKaydı: AraçTescilÜlkesi	1	
VuKalibrasyonKaydı: AraçTescilNumarası	14	
VuKalibrasyonKaydı: w-AraçKarakteristikSabiti	2	
VuKalibrasyonKaydı: k-KayıtCihazıSabiti	2	
VuKalibrasyonKaydı: l-LastikÇevresi	2	
VuKalibrasyonKaydı: LastikEbadı	15	
VuKalibrasyonKaydı: İzinVerilenHız	1	
VuKalibrasyonKaydı: EskiKilometreSayacıDeğeri	3	
VuKalibrasyonKaydı: YeniKilometreSayacıDeğeri	3	
VuKalibrasyonKaydı: EskiZamanDeğeri	4	
VuKalibrasyonKaydı: YeniZamanDeğeri	4	
VuKalibrasyonKaydı: SonrakiKalibrasyonTarihi	4	
...		
İmza	128	VuİmalatçıAdı'ndan başlayarak son VuKalibrasyonKaydı'nın en son baytına kadar bütün verinin (belgeler hariç) RSA imzası.

Şekil 4.10. Teknik bilgi verisini yorumlama [25]

4.2. Veri İndirme Cihazları Piyasa Taraması ve Karşılaştırma

Piyasada birçok marka ve model veri indirme cihazı bulunmaktadır. Bu veri indirme cihazları genel olarak tasarımları farklı olsada hepsi aynı kurallara göre ve belirli standartlar altında veri indirebilme işlemlerini gerçekleştirdiklerinden dolayı bu işlemler üzerinde herhangi bir avantajları söz konusu değildir. Tabi burada veri indirme hızları saniyede 9600 bit ile 115200 bit arasında bir seçimin söz konusu olduğunu unutmamak gerekir. Söz konusu en yüksek veri iletimhızı belirlenmişken daha düşük hızlarda veri indirmek belirli cihazları bir adım geriye düşürmektedir.

Şekil 4.11'de örnek olarak gösterilen veri indirme cihazları verileri indirip cihaz hafızalarında depolamaktadırlar. Bu veriler daha sonra analiz edilebilmek amacıyla USB üzerinden bilgisayara kopyalanabilmektedir. Piyasada fark oluşturmak, kullanıcıyı tatmin etmek bu noktada çok büyük önem arz etmektedir. Veri analizlerinin görselliği, hızı ve kolay anlaşılabilir olması bir veri indirme cihazını diğerlerinden ayıran en önemli etken olduğunu söyleyebiliriz. Önemli bir diğer etken ise veri indirme cihazının takografa ihtiyacı olmadan sürücü kart verilerini indirebilme yeteneği olduğunu söyleyebiliriz.



Şekil 4.11. Piyasada bulunan veri indirme cihazlarından örnekler

Görüldüğü üzere piyasada bulunan veri indirme cihazları analiz için kullanıcılara harici bir teçhizat ihtiyacı doğurmaktadır. Bu çalışmada yapılan mobil uygulama sayesinde verileri indirip, anında ve kolay anlaşılabilir bir şekilde analiz etme imkanı sunulmaktadır. Ayrıca diğer veri indirme cihazlarından bir farkı ise istenilen verileri

ayrı ayrı indirebilme imkanının kullanıcıya sunulmasıdır. Kullanıcı bilgi alma amacıyla bir araçtaki sadece detaylı hız verilerini indirip inceleyebilmektedir. Ya da isterse sadece teknik verileri indirip analizini yapabilmektedir.

4.3. Uygulamanın Çalışma Yapısı

Sayısal takograf veri indirme mobil uygulama yazılımının genel çalışma yapısını şöyle özetleyebiliriz:

- Program ilk başladığında çalışan Şekil 4.12'de gösterildiği gibi bir aktivitemiz bulunmaktadır. Burada bir iş parçacığı (thread) yardımıyla üç saniye beklenerek ana menü aktivitesine geçiş yapılır. Bu bekleme sırasında ekranda görüntü olması amacıyla logo vs. gösterimi yapılabilir.
- Program, Şekil 4.13'de gösterildiği gibi ana menü kısmına geldiğinde kullanıma hazır hale gelir. Burada veri görüntüleme, veri indirme ve indirme tarihlerini gösteren seçenekler bulunmaktadır.



Şekil 4.12. Uygulamanın açılış ekranı



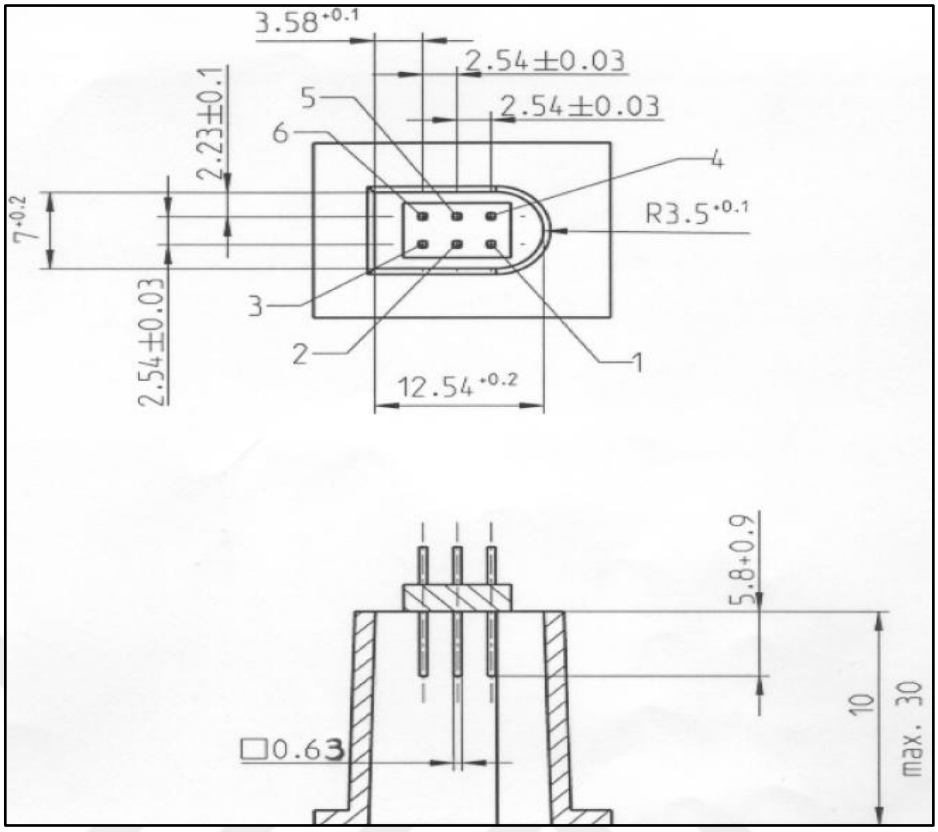
Şekil 4.13. Uygulamanın ana menüsü

4.3.1. Sistemin genel mimarisi

Tez çalışmasında gerçekleştirilen mobil uygulamanın mimarisini ele alırsak bir mobil cihaz, bir sayısal takograf, veri indirme izni olan takograf kartı ve bir adet takograf-tablet arası bağlantı kablosu yeterli olmaktadır. Yani sistem bir mobil uygulamadan oluşmaktadır. Kullanıcı veri indirme şartlarını sağlayarak istediği veriyi indirebilmektedir. İndirilen veriler anında mobil cihaz içerisinde analiz edilerek kullanıyıca sunulmaktadır.

4.3.2. Seri haberleşme (donanımsal arayüz)

Sayısal takograflar, veri indirme ve ölçümleme için altı bacaklı bir ön bağlantı noktası barındırmaktadırlar. Bu bağlantının yapısı Şekil 4.14'te verilmiştir. Verilen ölçüler milimetre cinsindendir. Tablo 4.2'de de ön bağlantı bacak işlevleri gösterilmiştir.

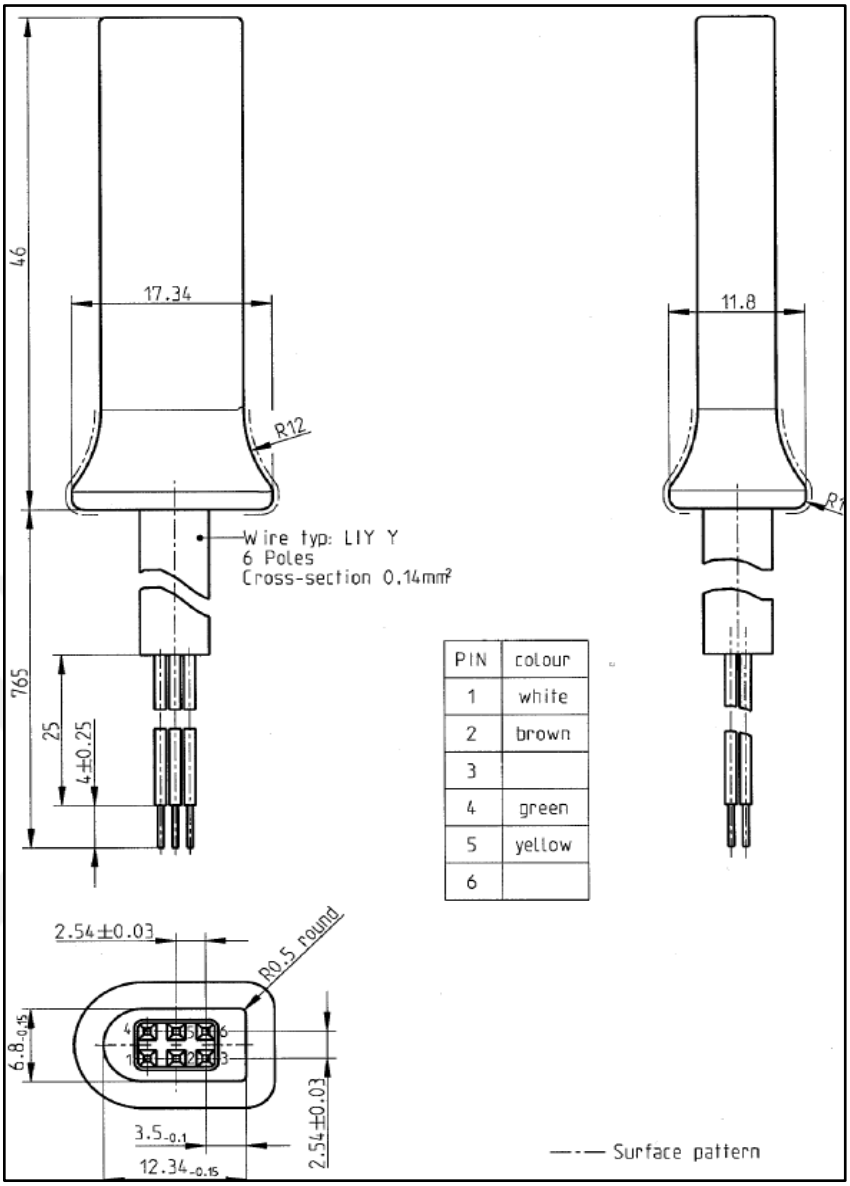


Şekil 4.14. Sayısaltakograf ön bağlantı yapısı [25]

Tablo 4.2. Ön bağlantı bacakları işlevleri

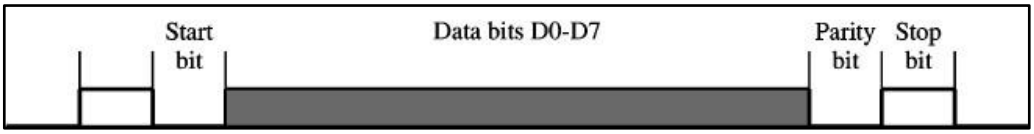
Pin	**Açıklama**
1	Toprak
2	Kalibrasyon veri hattı
3	Veri indirme RxD bacağı
4	Kalibrasyon sinyal hattı
5	Besleme
6	Veri indirme TxD bacağı

Sayısal takograf yönetmeliğinde bu bağlantı noktasına girecek fişe ait bilgiler verilmiştir. Buna göre fişin ölçüleri Şekil 4.15'da verilmiştir. Ölçüler milimetre cinsindendir. Takografın tablete bağlanıp haberleşme sağlanabilmesi için bir dönüştürücü devre yapılmıştır. Fiş için verilen ölçülerde plastik bir kutu ürettirilmiş ve yapılan dönüştürücü bu kutunun içerisine konulmuştur.



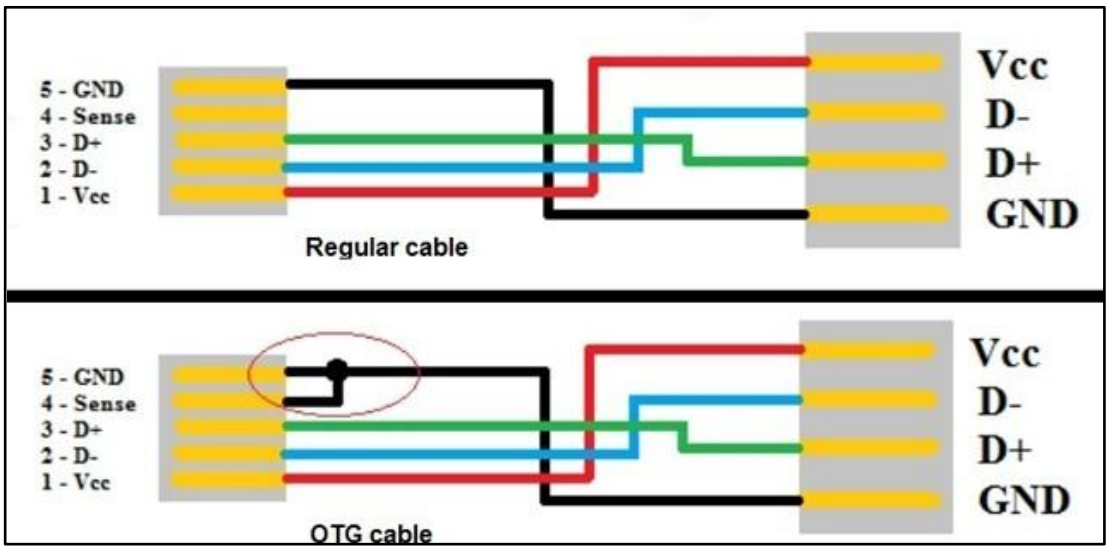
Şekil 4.15. Altı bacaklı fiş çizimi[26]

Veri indirme işlemi için bağlantı noktasının RxD(3), TxD(6) ve toprak(1) bacakları kullanılacaktır. Veri indirme TxD ve RxD bacakları RS232 standardında veri haberleşmesine izin vermektedir. Haberleşme hızı 9600–115200 sembol/saniye aralığındaki bir değere ayarlanabilmektedir. Bir haberleşme çerçevesinde bir adet başlangıç biti, en değersiz bit başta olmak üzere 8 adet veri biti, 1 adet eşleme(parity) biti, son olarak da 1 adet bitiş biti olmak üzere toplam 11 bit bulunmaktadır. Eşleme, çift eşleme(even) olarak verilmiştir. Şekil 4.16'da bir haberleşme çerçevesinin yapısı verilmiştir.



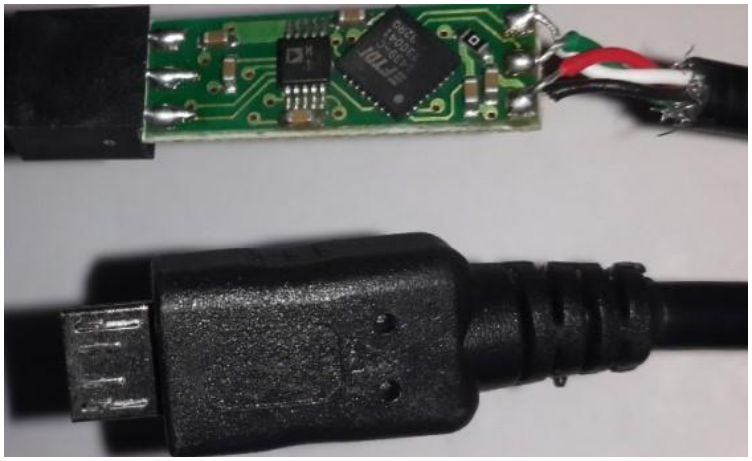
Şekil 4.16. Bir haberleşme çerçevesinin yapısı[25]

Bu çalışmada kullanılan tablet, dışarıdan USB ile çalışan bir cihaz bağlanmasına izin veren USB-OTG bağlantısına sahiptir. USB-OTG sayesinde tablet, üzerindeki mikro USB girişine bir mikro USB-OTG konnektörü bağlandığında Host olarak çalışmakta, dışarıdan kendisine bağlanan ve Device modunda çalışan cihazlarla olan haberleşmeyi kontrol edebilmektedir. Tablette bulunan mikro USB girişine normal bir mikro USB konnektörü bağlandığında ise tablet Device modunda çalışmaktadır. Bu senaryoda haberleşmenin başka bir cihaz tarafından (bilgisayar vb.) kontrol edildiği düşünülmelidir. Piyasada bulunan OTG kabloları bir ucu mikro USB-OTG, diğer ucu dişi USB A şeklinde ve 10–15 cm. uzunluğundadır. Yapılacak çalışma için piyasadaki kablolar elverişli olmadığından bu çalışma için 1 metre uzunluğunda USB-OTG kablosu ürettirilmiştir. Bu kablonun bir ucu Şekil 4.17'de alt kısımda görüldüğü gibi, Micro USB-OTG şeklinde, diğer ucu ise dönüştürücü devreye lehimlenmek üzere tel olarak bırakılmış şekildedir.



Şekil 4.17. Normal kablo ile otg kablo arasındaki fark[27]

Sayısal takografın sunduğu RS232 ara yüzü ile tabletin sunduğu USB-OTG ara yüzü arasında bağlantı kurabilmek için bir dönüştürücü devre gerekmektedir. Dönüştürücü devre gücünü USB'den almaktadır. Şekil 4.18'de ve Şekil 4.19'da donanımın bitmiş hali ve donanıma ait kapaklar gösterilmiştir.



Şekil 4.18. Donanımın bitmiş hali



Şekil 4.19. Donanıma ait kapaklar

4.3.3. Veri indirme

Ana ekran aktivitesinde görünen ‘araç verisi indir’ ve ‘kart verisi indir’ seçenekleri programın indirme özelliğini gösteren aktivitelerdir. Şekil 4.20’de araç veya kart verisi indirme seçeneği gösterilmektedir.

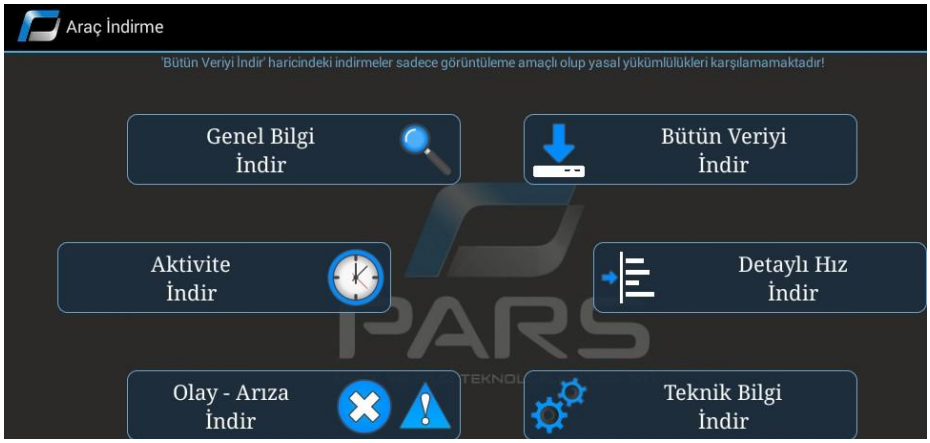
‘Araç verisi indir’ aktivitesi ile takograf verilerinin indirilmesi ve belirli bir formata göre yazılması sağlanmaktadır. Bu aktivitede FTDI yongası ile haberleşmek için USART yapılandırma ayarları yapılır. İndirme seçeneklerinden biri seçildiğinde daha önce Şekil 4.1’de gösterilen mesaj tipleri vasıtasıyla yapılan haberleşme ile veriler alınır. Alınan veriler mobil cihaz hafızasına araç dosya formatıyla kaydedilir. Bu formattan veri görüntüleme kısmında bahsedilmektedir.

‘Kart verisi indir’ aktivitesi ile karta ait veriler kaydedilmekte ve kart dosya formatına uygun bir şekilde hafızada tutulmaktadır. Kart indirme işlemi takograf cihazının içerisindeki bir karttan yapılabileceği gibi harici bir kart okuyucusunun tablete bağlanmasıyla da yapılabilmektedir. Daha önce Şekil 4.1’de gösterilen mesaj tipleri vasıtasıyla yapılan haberleşme ile veriler alınır. Alınan veriler mobil cihaz hafızasına kart dosya formatıyla kaydedilir. Bu formattan veri görüntüleme kısmında bahsedilmektedir.



Şekil 4.20. Veri indirme türü seçenek menüsü

Araç verisi indirme seçenek menüsü ve kart verisi indirme seçenek menüsü sırasıyla Şekil 4.21’de ve Şekil 4.22’de gösterilmiştir.

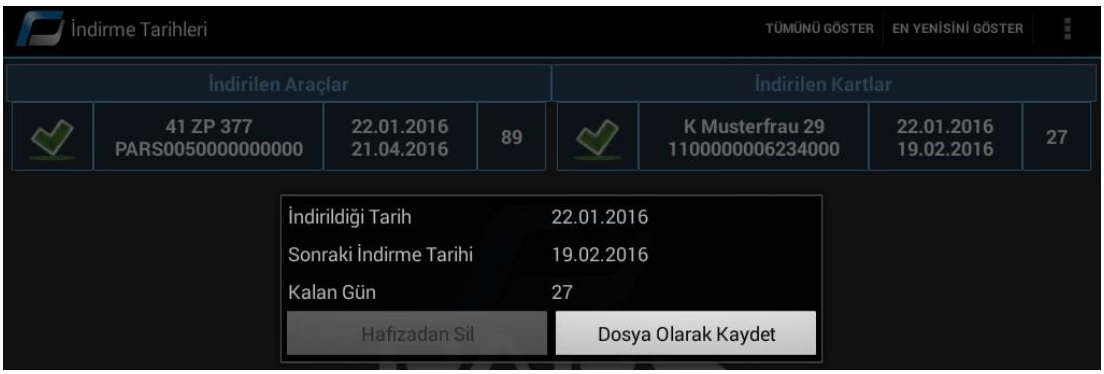


Şekil 4.21. Araç verisi indirme seçenek menüsü



Şekil 4.22. Kart verisi indirme seçenek menüsü

Araç verilerinin doksan günde bir kart verilerinin ise yirmisekiz günde bir yasal olarak indirilme zorunluluğu olduğundan bahsetmiştik. Ana ekranda bulunan indirme tarihleri seçeneği ile daha önce indirilen araç veya kart bilgilerinin bir sonraki indirme süresine ne kadar kaldığı bilgisi sunulmaktadır. Şekil 4.23'te bu bilgi ekranı gösterilmiştir.



Şekil 4.23. İndirme kalan gün sayısı bilgi ekranı

4.3.4. Veri görüntüleme

Ana ekranda veri görüntüleme seçildiğinde ne tür verinin görüntülenmek istediği sorulur. Araç veya kart verisi görüntüleme seçeneklerinden biri seçilip ilgili görüntüleme aktivitesine geçiş yapılır. Şekil 4.24'te veri görüntüleme seçeneği gösterilmiştir.



Şekil 4.24. Veri görüntüleme seçimi

4.3.4.1. Araç ve kart dosya isimlendirme formatı

Araç dosya formatı M_YYYYAAGG_SSDD_AA12BB_x.DDD şeklinde gösterilmektedir. Burada 'M' araç dosyasını, YYYYAAAGG indirme tarihini, SSDD ise indirme saat ve dakika bilgileridir. Sonrasında araç plakası (AA12BB) ve araç kimlik numarası (x) gelmektedir.

Kart dosya formatı C_YYYYAAAGG_SSDD_A_TURK_x şeklinde olması yönetmelik tarafından belirlenmiştir. Burada C kart dosyasını, YYYYAAAGG indirme tarihini, SSDD ise indirme saat ve dakika bilgileridir. Sonrasında A_TURK sürücü adını, x ise kart numarasını temsil etmektedir.

4.3.4.2. Araç verisi görüntüleme

Bu aktivitedehafızada bulunan kayıtların listelenmesi ve kayıtların aranması işlemleri yapılmaktadır. Kayıtların listelenmesi araç plakası, tarih ve tam ad şeklinde yapılmaktadır. Aynı zamanda kayıtların sınıflandırılmasında filtreleme yapılabilmektedir. Bunun için arama kısmına istenen arama kıstasları yazılmalıdır. Bulunan sonuçlar bu kıstaslara göre listelenmektedir.

Cihazda kayıtlı olan araç verilerinin yani daha önce indirilmiş araçların plaka, tarih ve tam ad sıralaması gösterilir. Bu şekilde kullanıcıya sunulan indirilmiş verilerden birinin seçilmesi durumunda o araca ait bilgiler görüntülenecektir. Şekil 4.25'de görüntülenecek aracın seçim aşamaları gösterilmiştir.

Gösterilen kayıtlardan herhangi biri seçildiğinde dosyanın yolu intent ile çözümleme aktivitesine aktarılır ve burada bu dosya okunarak tabhost üzerinden çeşitli aktiviteler ile verinin analizi yapılabilir.

Aktivite fonksiyonları olarak dosyanın okunması, dosya formatının çözümlenmesi ve indirme tipinin seçilmesi mevcuttur. Bu aktivitede yapılan ilk iş intent ile gelen dosya yolundaki dosyanın okunmasıdır. Dosya okunduktan sonra indirme dosyasında hangi veri tiplerinin olduğu (genel bilgi, aktiviteler, olay ve hata, detaylı hız, teknik veri) belirlenir. Bu veri tipleri Şekil 4.1'de 'veri aktarım talebi' bölümü altında gösterilmiştir. Tipler belirlendikten sonra indirilmiş veri parçalanarak uygun nesnelerde tutulur. Aktivite menüsünde indirme tipleri görüntülenmektedir. Bu seçenekler indirilen veride bu TREP'lerin olup olmamasına göre aktifleştirilir. Aktifleştirilenler arasından yapılan seçimden sonra bu seçimle ilgili layout(düzen) yüklenir.



Şekil 4.25. Görüntülenecek aracın seçimi

Genel bilgi görüntüleme:

Seçenekler menüsünden genel bilgi seçildikten sonra aktif hale gelen aktivitedir. Bu aktivite ile genel bilgi verisi anlamlandırılır ve kullanıcıya gösterilir. Genel bilgi analizine uygun fonksiyonlar kullanılarak veriler dizi (string) olarak tutulur ve ekranda gösterilir.Şekil 4.6'da gösterilen genel bilgi verisi yorumlamasına göre

analizi yapılır. Örneğin; İlk 388 bayt takograf güvenlik belgeleridir ve bu veri elemanları ekranda kullanıcıya sunulmamaktadır. Bunu takip eden 32 baytın içerisinde 17 bayt araç şasi numarasını, 1 bayt araç tescil ülkesini, 14 bayt araç plakasını, 4 bayt o anki tarih ve saati, 8 bayt indirilebilirlik süresini gösterir. Diğer veri elemanları da şekildeki gibi bayt uzunluklarına göre sınıflandırılır. Sınıflandırılan veriler yönetmelikte belirtilen veri tipi tariflerine göre yorumlandırılır. Örnek teşkil etmesi amacıyla bir bayt olan araç tescil ülkesini ele alırsak eğer sayısal takograf yönetmeliğinde bu baytın 'nation numeric' kodlamasında olduğu belirtilmektedir. Bu da tam sayı (integer) cinsinden hangi ülkeye denk geldiğine göre yorumlanır. İndirilebilirlik süresi 8 bayt demiştik. Yine yönetmeliğe göre 4 bayt ilk indirme tarihi, 4 bayt son indirme tarihi olarak belirtilmiştir. Peşpeşe gelen 4'er bayt veriden 'time real' cinsine dönüşüm veya direk tarih/zaman verisine dönüşüm yapılabilmektedir.

Genel bilgi görüntüleme menüsünde araca ait bilgi ekranı, şirket kilit kayıtları ve kontrol aktivite bilgileri mevcuttur. Bu ekranlar sırasıyla Şekil 4.26'da, Şekil 4.27'de ve Şekil 4.28'de gösterilmiştir.



Şekil 4.26. Genel bilgi görüntüleme ekranı



Şekil 4.27. Şirket kilit kayıtları görüntüleme ekranı



Şekil 4.28. Kontrol aktivite bilgileri görüntüleme ekranı

Aktivite görüntüleme:

Seçenekler menüsünden ‘aktiviteler’ seçildikten sonra aktif hale gelen aktivitedir. Bu aktivite ile veriler anlamlandırılır ve kullanıcıya sunulur. Veriler Şekil 4.7’de gösterildiği gibi analiz edilir. Örneğin; herhangi bir günlük veri ele alındığında ilk 4 bayt indirilen gün zamanını, sonraki 3 bayt gece yarısı kilometre değerini ve sırasıyla 2 bayt kart takma ve çıkartma sayısını, bu sayılar kadar her biri için 129 bayt olacak şekilde kart takma ve çıkartma kayıt bilgilerini (kart sahibinin adı, kart numarası, geçerlilik tarihi vb.) gösterir. Bu başlığa ait diğer veri elemanları da bayt uzunluklarına göre sınıflandırılır. Sınıflandırılan veriler yönetmelikte belirtilen veri tipi tariflerine göre yorumlandırılır. Bu veri tipi tarifleri sayısal takograf yönetmeliklerinde mevcuttur.

Aktivite fonksiyonları, günlerin bulunup gösterilmesi ve o güne ait detayların gösterilmesi olacak şekilde işlem yapmaktadır. İndirilen araç verisinde her aktivite günü ayrı ayrı olarak gönderildiğinden bu günlerin kullanıcı tarafından seçilerek istenilen günün görüntülenmesi sağlanmaktadır. Bunun için bütün kayıtlar için günler bulunup seçilebilir liste halinde kaydedilir. Seçilen günde dört farklı bilgi türü olabilir. Bunlar kart tak/çıkart kayıtları, günlük aktivite verisi, günlük çalışma dönemi yeri ve özel durumlardır. Bu verilerin görüntülenmesi aktivite içerisindeki buton grupları ile olmaktadır. Burada seçilen bir bilgi türünün verisigörsel olarak bu aktiviteye eklenir. Bu işlemler aktivitenin ilk açılışında otomatik olarak olan bilgi türünü gösterecek şekilde yapılır. Daha sonra kullanıcı istediğini buton grubundan seçebilmektedir. Şekil 4.29’da araçta yapılan aktiviteleri temsilen günlük aktivite görüntüsü sunulmuştur.



Şekil 4.29. Aktivite örnek analiz ekranı

Hata ve olay görüntüleme:

Seçenekler menüsünden hatalar seçildikten sonra aktif hale gelen aktivitedir. Bu aktivite ile hata verileri anlamlandırılır ve kullanıcıya sunulur. Hata sayıları araçlar için değişebilir olduğundan hataların görüntülenmesi için listeleme yapısı kullanılmıştır. Veriler Şekil 4.8'da gösterildiği gibi analiz edilir. Örneğin; ilk 1 bayt takograf cihazındaki hata sayısını, sonrasında bu hata sayısı kadar her biri için 82 bayt olacak şekilde hata kayıtlarını (hata tipi, kayıt amacı, başlama ve bitme zamanı vb.) gösterir. Bu başlığa ait diğer veri elemanları da bayt uzunluklarına göre sınıflandırılır. Sınıflandırılan veriler yönetmelikte belirtilen veri tipi tariflerine göre yorumlandırılır. Bu veri tipi tarifleri sayısal takograf yönetmeliklerinde mevcuttur.

Seçenekler menüsünden olaylar seçildikten sonra aktif hale gelen aktivitedir. Bu aktivite ile veriler anlamlandırılır ve kullanıcıya sunulur. Araçlar için olayların sayısı değişebilir olduğundan dolayı olayları görüntülemek için listeleme görünümü kulanılmıştır. Şekil 4.30'da örnek arıza (hata) kayıt görüntüsü ve Şekil 4.31'de örnek olay kayıt görüntüsü gösterilmiştir.



Şekil 4.30. Arıza kaydı örnek ekran görüntüsü



Şekil 4.31. Olay kayıtları örnek ekran görüntüsü

Detaylı hız görüntüleme:

Seçenekler menüsünden seçim yapıldıktan sonra aktif halen gelen bu aktivite detaylı hız verilerini anlamdırır ve kullanıcıya sunar. Veriler Şekil 4.9'da gösterildiği gibi analiz edilir. İlk 2 bayt hız blok sayısını (araç hareketi boyunca dakikada bir hız bloğu), sonraki 4 bayt hız bloğu başlama tarihini ve 60 bayt saniye başına düşen hız verisini gösterir. Son 128 bayt ise hız blok sayısından başlayarak son hız bloğunun son baytına kadar tüm verinin imzasını belirtir. Bu aktivitede detaylı hız verisi gün ve saat seçilerek grafiksel hale getirilir. Fonksiyon çağrıldığında ilk olarak bir dakikalık paketler halinde gelen verilerin aynı gün içinde olanlarının gruplanması yapılmaktadır. Sonrasında ise aynı günlerin saatlerinin ayrıştırılması sağlanmaktadır. Şekil 4.32'de detaylı hız verisi örnek analiz ekranı gösterilmektedir. Limit hız verisi de şekilde görülebileceği üzere istenen değerlere ayarlanabilmektedir. Ayrıca detaylı hız verisi incelenmesi istenen tarihi ve saati kullanıcı tarafından belirlenmesi mümkündür.



Şekil 4.32. Detaylı hız verisi örnek analiz ekranı

Teknik bilgi görüntüleme:

Seçenekler menüsünden seçildiği anda aktif hale gelir. Veriler Şekil 4.10'da gösterildiği gibi analiz edilir. Örneğin; ilk 36 bayt takograf imalatçı adını, sonraki 36 bayt takograf imalatçı adresini, sırasıyla 16 bayt takograf parça numarasını, 8 bayt takograf seri numarasını gösterir. Bu başlığa ait diğer veri elemanları da bayt uzunluklarına göre sınıflandırılır. Sınıflandırılan veriler yönetmelikte belirtilen veri tipi tariflerine göre yorumlandırılır. Bu veri tipi tarifleri sayısal takograf yönetmeliklerinde mevcuttur. Şekil 4.33'te teknik veri örnek analiz ekranı gösterilmiştir.



Şekil 4.33. Teknik bilgi verisi örnek analiz ekranı

4.3.4.3. Kart verisi görüntüleme

Kart görüntüleme seçildiğinde karşımıza bir liste vasıtasıyla cihazda kayıtlı olan kart verileri çıkmaktadır. Burada kayıtların daha hızlı aranması için arama kısmı mevcuttur. Kayıtlar arasında filtreleme işlemi ile bir iletişim kutusu üzerinden

gösterilen ‘ara’dizini ile girilen karakterler aranarak kayıtlarda eşleşip eşleşmeme durumuna göre gösterilen kayıtlar güncellenir. Kayıtlar, kart sahibinin adı, kart indirme tarihi ve kart numarası sıralaması ile gösterilmektedir.

Kartların veri boyutları farklılık gösterebilir. Bu farkı kart içerisindeki DF_Takograf klasöründe EF_Uygulama_Tanıtımı içinde olay, hata, aktivite, konum kaydı ve araç kayıtları sayısı belirler. Verilerin analizi takograf yönetmeliğindeki esaslar dikkate alınarak yapılmaktadır. Örneğin; ilk 25 bayt EF_ICC klasörüdür. Bu 25 bayt içerisinde 1 bayt saat durması (clock stop), 8 bayt kart genişletilmiş seri numarası, 8 bayt kart onay numarası, 1 bayt kart kişiselliştirme bilgisi, 5 bayt gömülü montaj bilgisi, 2 bayt devre tanıtıcı bilgisidir. Bu başlığa ait diğer veri elemanları da bayt uzunluklarına göre sınflandırılır.

Gösterilen kayıtlardan herhangi biri seçildiğinde bu dosyanın yolu intent aracılığıylaana aktiviteye aktarılır. Burada dosya okunarak tabhost (ekranı daha iyi kullanabilmek için farklı parçalara bölme) üzerinde çeşitli aktiviteler ile verinin analizi yapılır. Bu aktivitede dosyanın okunması, formatının çözümlenmesi ve uygun layout oluşturulması için üç farklı fonksiyon kullanılmıştır. Şöyle ki; ilk olarak intent ile gelen dosya okunur ve sonrasında indirme dosyasında hangi veri tiplerinin olduğu belirlenir. Tipler belirlendikten sonra indirme verisi parçalanarak uygun nesnelerde tutulur. Çözümlemeden sonra verinin sürücü veya servis kartlarından hangisi olduğuna bakılır. Bu fark EF_Uygulama_Tanıtımı bilgisinde mevcuttur. Kart türünüe göre uygun layout yüklendikten sonra görüntüleme moduna geçilir. Şekil 4.35’te, Şekil 4.35’te ve Şekil 4.36’da indrilmiş kart verisi örnek analiz ekranı gösterilmektedir. Analiz ekranında bulunan simgelere dokunarak simge başlığına ait verilerin incelenmesi mümkündür.



Şekil 4.34. Kart verisi analiz örnek ekran görüntüsü, kart bilgisi



Şekil 4.35. Kart verisi analiz örnek ekran görüntüsü, olay kayıtları



Şekil 4.36. Kart verisi analiz örnek ekran görüntüsü, aktivite kayıtları

5. SONUÇLAR VE ÖNERİLER

Karayolu taşımacılığı yapan belirli araçlara yasa ve yönetmeliklerle yürürlüğe konulansayısal takograf kullanım zorunluluğu ile birlikte, bu cihazlardan veri indirme işleminin en kısa yoldan ve en etkin şekilde sağlanmasıylaönemli bir ihtiyacın giderileceği belirlenmiştir.

Sayısal takografların hafızasına kaydedilen verilerin en etkin inceleme yöntemi veri indirme sonrasındaki analiz kısmına bağlıdır. Çünkü piyasada kullanılan tüm veri indirme cihazları verileri aynı biçim ve özellikte almaktadır. Dolayısıyla kullanıcı açısından oluşabilecek tek fark verilerin analizi sonrası açık ve anlaşılabilir gösterimidir.

Bu tez kapsamında geliştirilen uygulama ile veriler ister parça parça ister bir bütün halinde indirilebilmektedir. Ancak yasal yükümlülüklerden dolayı sağlanabilmesi açısından araçtanson 90 gün veya tüm verinin indirilmesi gerekmektedir. Bunların dışında indirilen tüm veriler tamamen kullanıcıyı bilgilendirme maksadıyladır. Ayrıca tüm veriyi indirme seçeneği ile yapılan indirmelerde bir sonraki zorunlu indirme tarihi sürekli kontrol edilebilir haldedir. Bu sayede kullanıcının indirme işlemlerini harici bir şekilde takip etmesine gerek yoktur. Aynı durum 28 günde bir indirme zorunluluğu olan kartlar içinde geçerlidir. Dolayısıyla mobil cihaz üzerinde kullanılan bu uygulama kullanıcı açısından hem indirme takibini onun yerine yapmaktadır hem de verileri ayırt edici biçimlerde kullanıcıya sunmaktadır.

KAYNAKLAR

[1] Lee J.K., Jong, Y.L., Android Programming Techniques For Improving Performance, *Awareness Science and Technology (iCAST)* , 2011, 386-389.

[2] Sonuç E., OrtakcıY., ElenY., Karabük Üniversitesi Bilgi Sistemi Android Uygulaması, *2013 Akademik Bilişim Kongresi*, Antalya, Türkiye, 23-25 Ocak 2013.

[3] http://www.kugm.gov.tr/BLSM_WIYS/KUGM/tr/Belgelik/Mevzuat/y_genelgeler/20140606_161615_2769_1_64.doc (Ziyaret tarihi: 13 Kasım 2015).

[4] http://web.itu.edu.tr/kamasak/pubs/pdf/Baris_Sarikaya_tez.pdf (Ziyaret tarihi: 14 Aralık 2015).

[5] http://shiftdelete.net/ios-ile-Androidi-karsilastirdik-34697?p=1 (Ziyaret tarihi: 15 Aralık 2015).

[6] Yaşar E., Android Tabanlı Anne ve Anne Adayları İçin Bilgi Paylaşımı Platformu, Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya, 2017, 468286.

[7] https://faculty.ist.psu.edu/bagby/432Spring12/T9/Background.html (Ziyaret tarihi: 27 Eylül 2015).

[8] https://tr.wikipedia.org/wiki/Android_%28i%C5%9Fletim_sistemi%29 (Ziyaret tarihi: 19 Kasım 2015).

[9] Taç M., *Android Programlama*, 1. Baskı, Dikeyeksen Yayınları, İstanbul, 2011.

[10] LamichhaneR., Mobile Biometric Challenge Recording Interface, Master's Thesis, University of Tartu, Institute of Computer Science, Tartu, 2012.

[11] https://en.wikipedia.org/wiki/Android_%28operating_system%29#/media/File:Android-System-Architecture.svg (Ziyaret tarihi: 29 Kasım 2015).

[12] Yıldız G., Renk Analizi Tabanlı Öneri Sistemleri İçin Mobil Uygulama Geliştirme, Yüksek Lisans Tezi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Kocaeli, 2012, 315743.

[13] http://www.technomicon.com/TechnomiconImages/ElectroPoliticsImages/EP-11-15-10Images/JavaAndroidFlowchart.jpg (Ziyaret tarihi: 29 Kasım 2015).

[14] Çınar S., Mobil Android Ortamında Parmak İzi Tanıma ve Kimlik Doğrulama Sisteminin Geliştirilmesi, Yüksek Lisans Tezi, Haliç Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2014, 357287.

[15] http://www.turkcebilgi.com/Java_%28programlama_dili%29 (Ziyaret tarihi: 15 Aralık 2015).

[16] Jing Y., Ahn G-J., Hu H., Model-based Conformance Testing For Android, *Proceedings of the 7th International Workshop on Security (IWSEC)*, Fukuoka, Japan, 7-9 November, 2012.

[17] http://e-nokta.org/content/Android-debug-bridge-adb-kullanimi (Ziyaret tarihi: 16 Aralık 2015).

[18] http://www.emrekoc.net/activity-nedir/ (Ziyaret tarihi: 28 Aralık 2015).

[19] https://gelecegiyazanlar.turkcell.com.tr/konu/Android/egitim/Android-201/activity-sinifi (Ziyaret tarihi: 28 Aralık 2015).

[20] http://developer.Android.com/reference/Android/app/Activity.html (Ziyaret tarihi: 28 Aralık 2015).

[21] http://www.kto.org.tr/sayisal-takograf-nedir-162s.htm (Ziyaret tarihi: 10 Eylül 2015).

[22] https://www.atso.org.tr/icerik/3/237/sayisal-takograf-nedir?.html(Ziyaret tarihi: 10Eylül2015).

[23] http://www.vdo.com/media/182602/flc_i_cvam_tco_broschuere_dtco_en.pdf (Ziyaret tarihi: 25 Eylül 2015).

[24] https://staum.tobb.org.tr/kartlar.jsp (Ziyaret tarihi: 18 Ekim 2015).

[25] Uluslararası Karayolu Taşımacılığı Yapan Araçlarda Kullanılan Takograf Cihazları Hakkında Yönetmelik, Ek-1 ve Ek-1B, *T.C. Resmi Gazete, 27587*, 21 Mayıs 2010.

[26] EU 165/2014 of the European Parliament and Council Regulation (EEC) 3821/85, On Recording Equipment In Road Transport No, 1985

[27] https://www.quora.com/How-different-is-a-normal-usb-cable-from-OTG-usb-cable (Ziyaret tarihi: 9 Ekim 2015)

EKLER

Ek-A

Ek kısmında sayısal takograflardan veri indirme mobil uygulamasının diğer arayüzleri ve indirilen verilerin analiz ekran görüntüleri aşağıda gösterilmiştir.



Şekil A.1. Uygulamanın USB bağlantı aşaması ekran görüntüsü



Şekil A.2. Uygulamanın veri indirme seçenek ekran görüntüsü



Şekil A.3. Aktivite tarih seçimi ekran görüntüsü

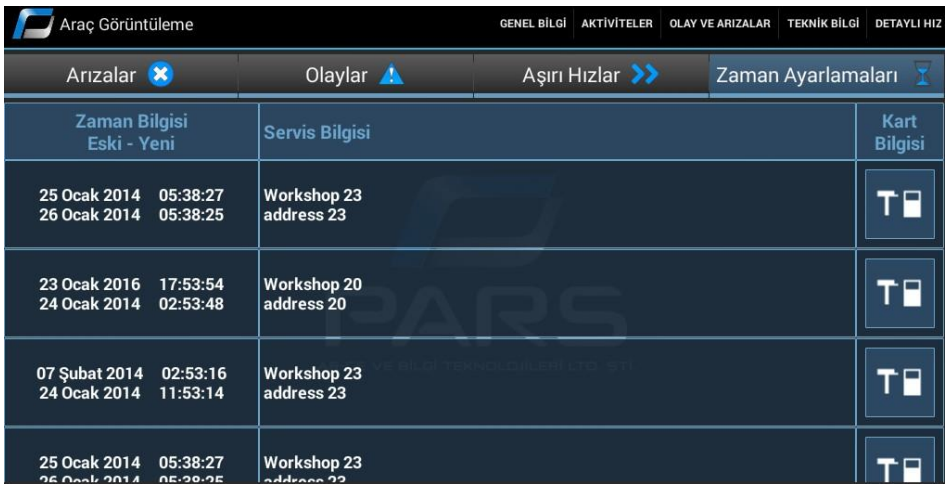


Şirket Adı	Şirket Adresi	Kilit Tarihi Açma - Kapatma	Kart Bilgisi
Company 47	address 47	31 Aralık 2013 10:18:50 07 Temmuz 2014 22:37:19	3100000003433 0 0 0
Company 36	address 36	07 Temmuz 2014 22:37:19 05 Mayıs 2015 21:22:29	3100000003422 0 0 0
Pars Ar-Ge Information Tech. Ltd.	Company 03	05 Mayıs 2015 21:22:29 05 Mayıs 2015 21:25:00	PARSARGECP03 0 0 0
Company 62	address 62	05 Mayıs 2015 21:25:00 27 Haziran 2015 17:22:48	3100000003462 0 0 0
Pars Ar-Ge Information Tech. Ltd.	Company 03	27 Haziran 2015 17:22:48 11 Ağustos 2015 17:03:28	PARSARGECP03 0 0 0
Company 45	address 45	11 Ağustos 2015 17:03:28 --h--	3100000003431 0 0 0

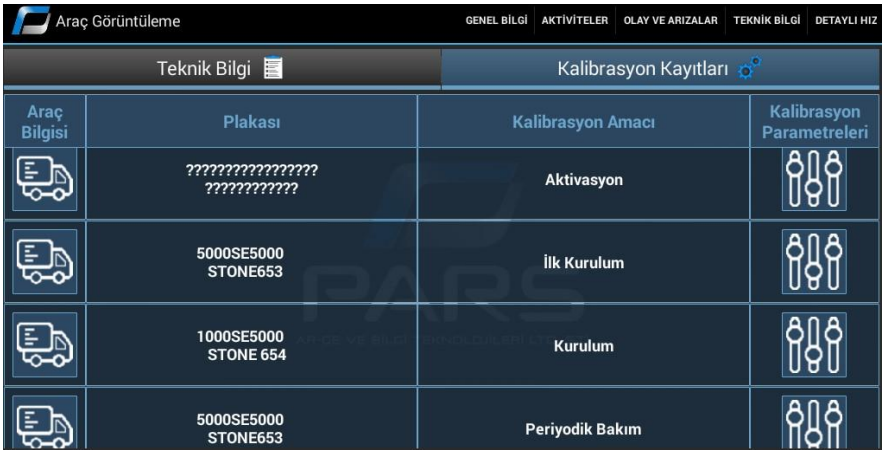
Şekil A.4. Şirket kilit kayıtları ekran görüntüsü



Şekil A.5. Aşırı hız kayıtları ekran görüntüsü



Şekil A.6. Zaman ayarlamaları ekran görüntüsü



Şekil A.7. Kalibrasyon kayıtları ekran görüntüsü



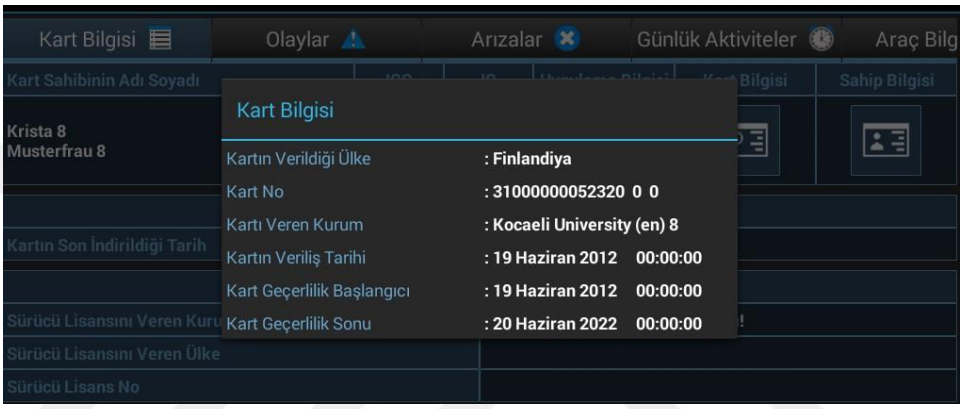
Şekil A.8. Kalibrasyon parametreleri ekran görüntüsü



Şekil A.9. Olay kayıtları ekran görüntüsü



Şekil A.10. Kart yetkinlik belgesi ekran görüntüsü



Şekil A.11. Kart bilgileri ekran görüntüsü



Şekil A.12. Kart sahibi bilgileri ekran görüntüsü



Şekil A.13. Karttaki olay bilgileri ekran görüntüsü



Şekil A.14. Karttaki hataların ekran görüntüsü



Şekil A.15. Kartın takıldığı araç bilgilerinin ekran görüntüsü



Şekil A.16.Kart konum kayıtları ekran görüntüsü



Şekil A.17. Avrupa Birliği Yönetmeliği'ne göre sürüş ve dinlenme ihlalleri ekran görüntüleri



Şekil A.18. Karayolları Trafik Yönetmeliği'ne göre sürüş ve dinlenme süre ihlalleri ekran görüntüsü



Şekil A.19. Karta kayıtlı olayların örnek ekran görüntüsü

KİŞİSEL YAYINLAR VE ESERLER

[1] **Aktaş K.**, Ertürk S., Türkoğlu Ç., Eyigel L., Tekin T.,Sayısal Takograflar İçin Mobil Veri İndirme Cihazı, *6. Karayolu Trafik Güvenliği Sempozyumu Ve Sergisi*, Ankara, Türkiye,12-14 Kasım 2015.

ÖZGEÇMİŞ

1989 Kadıköy/İstanbul doğumlu olan Kadir Aktaş ilk ve orta öğretimini İstanbul'da tamamladı. Lise eğitimine 2003 yılında Maltepe Anadolu Lisesi'nde başlayarak son sene tam burslu olarak Bostancı Doğa Koleji'ne geçiş yaptı ve 2007 yılında mezun oldu. 2007 yılında başladığı Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü'nden 2011 yılında lisans derecesini aldı.

Eylül 2011 yılında girişimci ve ortak olarak Marmara Otomasyon şirketinin kurulmasında yer aldı. 2012 yılında kalite kontrol mühendisi ünvanı ile Marmara Teknik Makine Ltd. Şti. bünyesinde görev aldı. 2013-2017 yılları arasında Pars Ar-Ge firmasında ar-ge mühendisi olarak çalıştı. 2017 yılının Şubat ayından bu yana TUSAŞ-TEI şirketinde kıdemli gömülü yazılım mühendisi olarak çalışmaktadır.