



Teknoloji Fakültesi

## BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

**MEDORA**

# YAPAY ZEKA DESTEKLİ UZAKTAN HASTA TAKİP VE DEĞERLENDİRME SİSTEMİ

"

**BİTİRME PROJESİ**

**2. ARA RAPORU**

Bilgisayar Mühendisliği Bölümü

**DANIŞMAN**

Dr. Öğr. Üyesi TİMUR İNAN

İSTANBUL, 2025

**MARMARA ÜNİVERSİTESİ**  
**TEKNOLOJİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

Marmara Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği Öğrencileri Mustafa ŞENOVA, Muhammet Emin Anlar, Emir MURATOĞLU ve Muhammed Cüneyt ERİŞ tarafından “YAPAY ZEKA DESTEKLİ UZAKTAN HASTA TAKİP ve DEĞERLENDİRME” başlıklı proje çalışması, 2025 tarihinde savunulmuş ve jüri üyeleri tarafından başarılı bulunmuştur.

**Jüri Üyeleri**

Dr. Öğr. Üyesi xxx xxx  
Marmara Üniversitesi  
Prof. Dr. Xxx xxx  
Marmara Üniversitesi  
Prof. Dr. Xxx xxx  
Marmara Üniversitesi

**(Danışman)**

(Üye)

(Üye)

(İMZA).....

(İMZA).....

(İMZA).....

## ÖNSÖZ

Proje çalışmamız süresince karşılaştığım bütün problemlerde, sabırla yardım ve bilgilerini esirgemeyen, tüm desteğini sonuna kadar yanımda hissettiğim değerli hocalarım, sayın Dr. Öğr. Üyesi Xxx xxx ve sayın Prof. Dr. Xxx xxx' a en içten teşekkürlerimi sunarım.

Bu proje çalışması fikrinin oluşması ve ortaya çıkmasındaki önerisi ve desteğinden dolayı değerli hocam Dr. Öğr. Üyesi Xxx xxx' a teşekkür ederim.

Proje çalışmam sırasında maddi ve manevi desteklerini esirgemeyen okul içerisinde ve okul dışında her zaman yanımda olan değerli çalışma arkadaşlarım ve hocalarım Doç. Dr. Xxx xxx ve Dr. Öğr. Üyesi ' xxx xxx a sonsuz teşekkürlerimi sunarım.

# **İÇİNDEKİLER**

## **1. GİRİŞ**

### **1.1 Proje Çalışmasının Amacı ve Önemi**

## **2. LİTERATÜR TARAMASI**

### **2.1.1 Mobil Sağlık (mHealth) Kavramı ve Gelişimi**

### **2.1.2 Kardiyovasküler Sağlıkta Mobil Uygulamaların Rolü ve Önemi**

### **2.1.3 Mevcut Kardiyovasküler Sağlık Mobil Uygulamalarının Sınırlılıkları ve Araştırma Boşlukları**

### **2.2 Yapay Zeka ve Makine Öğrenmesi Sağlık Alanında (EKG Analizi ve Kardiyovasküler Risk Tahmini Odaklı)**

### **2.2.1 Sağlıkta Yapay Zeka ve Makine Öğrenmesi Uygulamalarına Genel Bakış**

### **2.3 React Native ve Expo ile Mobil Sağlık Uygulaması Geliştirme**

### **2.3.1 React Native ve Expo Frameworklerine Genel Bakış**

### **2.3.2 Mobil Sağlık Uygulamaları Geliştirmede React Native ve Expo Kullanımı**

### **2.4 Firebase Veritabanı ve Bulut Bilişim ile Sağlık Verisi Yönetimi**

### **2.4.1 Firebase Platformuna Genel Bakış ve Sağlık Uygulamaları İçin Uygunluğu**

### **2.4.2 Bulut Bilişim ve Sağlık Verisi Güvenliği ve Gizliliği**

## **3. MATERYALLER VE METODLAR**

### **3.1.1 PTB Diagnostic ECG Veri Seti**

### **3.1.2 Veri Seti Görselleştirme Detayları**

### **3.1.3 Veri Ön İşleme Detayları**

### **3.1.4 Model Mimarisi Detayları**

**3.1.5 Model Eğitimi ve Değerlendirme Detayları**

**3.2.1 Kardiyovasküler Hastalık (KVH) Veri Seti**

**3.2.2 Veri Seti Görselleştirme Detayları**

**3.2.3 Veri Ön İşleme Detayları**

**3.2.4 Model Mimarisi Detayları**

**3.2.5 Model Eğitimi ve Değerlendirme Detayları**

#### **4. MOBİL UYGULAMA GELİŞTİRME**

**4.1. Uygulama Mimarisi ve Tasarımı**

**4.2. React Native ve Expo ile Mobil Uygulama Geliştirme Süreci**

**4.3. Firebase Entegrasyonu ve Veritabanı Tasarımı**

#### **5. YAPILAN VE YAPILACAK ÇALIŞMALAR**

**5.1. Şu Ana Kadar Yapılan Çalışmalar**

**5.1.1 Mobil Uygulama Altyapısı ve Temel Arayüzler**

**5.1.2 Sağlık Verisi Analizi ve İşleme Sistemi**

**5.1.3 Veritabanı Yapısı ve Firebase Entegrasyonu**

**5.1.4 Yapay Zekâ Modelleme ve Backend Entegrasyonu**

**5.2. Bundan Sonra Yapılacak Çalışmalar**

**5.2.1. Yapay Zekâ Modellerinin Tam Entegrasyonu**

**5.2.2. Gerçek Veri Entegrasyonunun Tamamlanması**

**5.2.3. Doktor-Hasta İletişim Sisteminin Geliştirilmesi**

**5.2.4. Test ve Optimizasyon**

## ÖZET

**Bu proje çalışmasında, kalp atış hızı, kan basıncı, vücut kitle indeksi ve diğer hayati vücut parametrelerini işleyen, yapay zeka destekli, uzaktan hasta takip ve değerlendirme sistemi geliştirilmiştir.**

İkinci ara rapor döneminde, React Native ve Expo teknolojileri kullanılarak kapsamlı bir mobil uygulama geliştirilmiş, Firebase Firestore ile güvenli veri yönetimi sağlanmış ve Google Fit entegrasyonu ile otomatik sağlık verisi toplama sistemi kurulmuştur. Uygulama, kullanıcıların sağlık verilerini görselleştirmesine, trend analizleri yapmasına ve kardiyovasküler risk tahminleri almasına olanak tanımaktadır.

Sistem, Z-Score tabanlı anomali tespiti, AHA standartlarına göre kan basıncı analizi, taşikardi/bradikardi tespiti ve BMI sınıflandırması gibi gelişmiş analiz algoritmaları içermektedir. Ayrıca, doktor-hasta iletişimi için mesajlaşma sistemi, randevu yönetimi ve hasta listesi özellikleri de entegre edilmiştir.

Proje, PTB Diagnostic ECG veri seti ile eğitilen Convolutional Autoencoder modeli ve 70.000 hasta verisi ile geliştirilen kardiyovasküler risk tahmin modelleri ile desteklenmektedir. Flask backend API ve OpenAI GPT entegrasyonu ile sağlık verisi analizi ve yorumlama yetenekleri sağlanmıştır.

**Mart, 2025**

**Mustafa ŞENOVA**

**Emir MURATOĞLU**

**Muhammed Cüneyd ERİŞ**

**Muhammet Emin ANLAR**

## 1. GİRİŞ

Günümüz sağlık sistemleri; kronik hastalıkların artışı, yaşlanan nüfus ve sağlık hizmetlerine olan talebin yükselmesi gibi önemli zorluklarla karşı karşıyadır. Bu zorlukların üstesinden gelmek için, sağlık hizmetlerinin daha etkin, verimli ve hasta odaklı hale getirilmesi gerekmektedir. Teknolojinin sağlık alanındaki hızlı gelişimi, bu hedeflere ulaşmak için önemli fırsatlar sunmaktadır. Özellikle mobil teknolojiler, hasta ve doktor arasındaki iletişimi güçlendirme, sağlık verilerini toplama ve analiz etme, ve kişiselleştirilmiş sağlık hizmetleri sunma potansiyeline sahiptir.

Bu proje çalışması, mobil teknolojilerin bu potansiyelinden yararlanarak hasta vücut verilerini toplayan, analiz eden ve ilgili doktorlara sunan yenilikçi bir mobil uygulama geliştirmeyi amaçlamaktadır. Uygulama, sadece veri görselleştirmesi sunmakla kalmayıp, aynı zamanda bu verileri anlamlandırarak, erken teşhis ve proaktif sağlık yönetimi için önemli bilgiler sağlamayı hedeflemektedir.

### 1.1. Proje Çalışmasının Amacı ve Önemi

Bu projenin temel amacı, hasta vücut verilerini etkin bir şekilde yönetebilen, analiz edebilen ve sağlık profesyonelleriyle paylaşabilen kullanıcı dostu bir mobil uygulama geliştirmektir. Uygulama, çeşitli sensörler ve manuel girişler aracılığıyla kalp atış hızı, kan basıncı, uyku düzeni, aktivite seviyesi gibi hayati verileri toplayacak ve bu verileri anlamlı bilgilere dönüştürecektir.

Projenin önemi çeşitli açılardan değerlendirilebilir:

**Hasta Odaklılık ve Yetkilendirme:** Uygulama, hastaların kendi sağlık verilerini daha iyi anlamalarını ve sağlık durumları hakkında bilinçli kararlar almalarını sağlayarak hasta odaklı bir yaklaşımı destekler. Hastalar, verilerini takip ederek sağlıkları üzerinde daha fazla kontrol sahibi olacak ve proaktif sağlık yönetimine katkıda bulunacaklardır.

Erken Teşhis ve Önleyici Sağlık Hizmetleri: Uygulama, verileri analiz ederek taşikardi riski gibi potansiyel sağlık sorunlarını erken aşamada tespit edebilir. Bu sayede, doktorlar zamanında müdahalede bulunarak hastalıkların ilerlemesini önleyebilir ve tedavi süreçlerini iyileştirebilir. Önleyici sağlık hizmetlerinin önem kazandığı günümüzde, bu tür uygulamalar büyük bir potansiyele sahiptir.

Doktor-Hasta İletişiminin Güçlendirilmesi: Uygulama, hastaların verilerini doktorlarıyla kolayca paylaşabilmelerini sağlayarak doktor-hasta iletişimini güçlendirir. Doktorlar, daha kapsamlı ve düzenli verilere erişerek hastalarının sağlık durumlarını daha iyi değerlendirebilir, daha doğru teşhisler koyabilir ve tedavi planlarını kişiselleştirebilirler.

Sağlık Sistemi Verimliliğinin Artırılması: Veri toplama ve analiz süreçlerini otomatikleştirerek, sağlık profesyonellerinin zamanını daha verimli kullanmalarına olanak tanır. Manuel veri girişi ve dosya paylaşımı gibi zaman alıcı işlemlerin azalması, sağlık hizmetlerinin genel verimliliğini artırabilir.

Maliyet Etkinliği: Erken teşhis ve önleyici sağlık hizmetleri sayesinde, ilerleyen dönemlerde ortaya çıkabilecek daha ciddi sağlık sorunlarının ve dolayısıyla tedavi maliyetlerinin azaltılmasına katkıda bulunabilir.

Sonuç olarak, bu proje çalışması, mobil teknolojileri kullanarak sağlık hizmetlerini daha erişilebilir, etkili ve hasta odaklı hale getirme potansiyeline sahip yenilikçi bir çözüm sunmayı amaçlamaktadır. Geliştirilecek uygulama, hem hastaların sağlık yönetim süreçlerine aktif katılımını destekleyecek, hem de sağlık profesyonellerine daha iyi veri ve analiz imkanları sunarak sağlık hizmetlerinin kalitesini artıracaktır.



## **2. LİTERATÜR TARAMASI**

### **2.1. Mobil Sağlık (mHealth) Uygulamaları ve Kardiyovasküler Sağlık**

Mobil sağlık (mHealth), mobil cihazlar ve uygulamalar üzerinden sağlık hizmetlerinin sağlanmasını ifade eden bir kavramdır. Son yıllarda, mHealth uygulamaları sağlık hizmetlerinde önemli bir rol oynamaya başlamış ve kardiyovasküler sağlık alanında da birçok yenilik sunmuştur. Bu alt başlıkta, mHealth'in tanımı, kapsamı, gelişimi ve özellikle kardiyovasküler sağlıktaki yeri incelenecektir.

#### **2.1.1. Mobil Sağlık (mHealth) Kavramı ve Gelişimi**

Mobil sağlık (mHealth), mobil cihazlar ve uygulamalar aracılığıyla sağlanan sağlık hizmetlerini tanımlayan bir kavramdır. Teknolojinin hızlı gelişimi ile birlikte mHealth, dünya genelinde sağlık hizmetlerine erişimin kolaylaştırılması ve sağlık hizmetlerinin daha etkili bir şekilde sunulmasında önemli bir rol oynamaktadır. mHealth uygulamaları, kronik hastalık yönetiminden kişisel sağlık takibine, uzaktan hasta izlemeye kadar çeşitli alanlarda kullanılmaktadır. Özellikle, hastaların sağlık verilerini takip etme ve yorumlama imkanı sunarak sağlık profesyonelleri ile iletişimlerini güçlendirmektedir.

mHealth'in sağlık sistemlerine entegrasyonu; sağlık hizmetlerinin kalitesini artırırken, hasta memnuniyetini de yükseltmektedir. Ancak, bu süreç bazı zorluklar da doğurmakta; veri güvenliği, kullanıcı uyumu ve klinik validasyon eksiklikleri gibi sorunlar ortaya çıkmaktadır. Literatürde, bu zorlukların üstesinden gelinmesi için çeşitli stratejiler önerilmiştir. Özellikle, sağlık profesyonellerinin mHealth uygulamalarını entegre etmesi ve kullanıcıların ihtiyaçlarına uygun uygulamalar geliştirilmesi kritik önem taşımaktadır.

Kardiyovasküler sağlıkla ilgili mHealth uygulamaları, kronik hastalık yönetimi, kişisel sağlık takibi ve uzaktan hasta izleme gibi alanlarda öne çıkmaktadır. Örneğin, hipertansiyon veya diyabet gibi kronik hastalıkları olan bireyler için mHealth uygulamaları, hastaların sağlık durumlarını izlemeye ve tedavi süreçlerini yönetmeye yardımcı olmaktadır. Giyilebilir cihazlar vasıtasıyla elde edilen verilerin analiz edilmesi, sağlık profesyonellerinin hastalarını daha yakından takip etmelerine olanak tanımakta ve

acil durumlarda hızlı müdahale şansı sunmaktadır.

Literatür incelendiğinde, mHealth uygulamalarının hasta ve sağlık profesyonelleri üzerindeki etkileri de oldukça olumlu yöndedir. Hastalar, sağlık durumlarını daha iyi anlama ve yönetme fırsatı bulurken, sağlık profesyonelleri, hasta verilerine daha hızlı ulaşarak daha etkili kararlar alabilmektedir. Bununla birlikte, mHealth uygulamalarının kullanıcılar arasında yaygın kabul görmesi, teknoloji kabul teorileri aracılığıyla daha iyi anlaşılabilir ve geliştirilmelidir [1].

Sonuç olarak, mHealth uygulamaları, kardiyovasküler sağlık başta olmak üzere birçok sağlık alanında önemli fırsatlar sunmakta, aynı zamanda bazı zorlukları da beraberinde getirmektedir. Bu uygulamaların etkin kullanımı, sağlık hizmetlerinin kalitesini artırmak ve sağlık sistemlerini güçlendirmek adına kritik bir rol oynamaktadır.

### **2.1.2. Kardiyovasküler Sağlıkta Mobil Uygulamaların Rolü ve Önemi**

Kardiyovasküler hastalıklar (KVH), dünya genelinde önemli bir sağlık sorunu teşkil etmekte ve hastalıkların önlenmesi, yönetimi ve rehabilitasyonu için yeni çözümlere ihtiyaç duyulmaktadır. Mobil uygulamalar, KVH ile mücadelede önemli bir araç olarak ortaya çıkmakta ve hastaların sağlık yönetiminde aktif rol oynamalarına olanak tanımaktadır.

#### **Mobil Uygulamaların KVH Yönetimindeki Rolü**

Mevcut mobil uygulamalar, kardiyovasküler Sağlık için çeşitli alanlarda etkili çözümler sunmaktadır. Örneğin:

- **Kardiyovasküler Hastalıkların Yönetimi:** Uygulamalar, hastaların kalp sağlığı ile ilgili çeşitli verileri takip etmelerine ve uzmanlar ile hızlı bir şekilde etkileşimde

bulunmalarına imkan tanımaktadır. Tele-kardiyoloji uygulamaları, hastaların durumlarını uzaktan izlemeye, acil durumlarda hızlı yanıt almak için verileri anında paylaşmaya olanak sağlar.

- **Önleme ve Rehabilitasyon:** Mobil uygulamalar, hastaların sağlık bilgilerini kaydetmelerine, tedavi süreçlerine uymalarına ve sağlık hedeflerine yönelik adımlar atmalarına yardımcı olur. Örneğin, kalp hastaları için özel olarak geliştirilmiş uygulamalar, düzenli egzersiz yapmayı teşvik eden aktiviteleri takip etme ve beslenme düzenini gözlemleme özellikleri sunmaktadır.
- **KVH Risk Faktörlerinin Yönetimi:** Uygulamalar, aktivite takibi, beslenme izleme ve stres yönetimi gibi özelliklerle, bireylerin KVH risk faktörlerini yönetmelerine yardımcı olmaktadır. Örneğin, bu tür uygulamalar, kullanıcıların günlük kalori alımını ve fiziksel aktivitelerini izlemelerine imkan tanıyabilir. Bunun yanı sıra, stres yönetimi için meditasyon ve nefes egzersizleri gibi içerikler sunarak kalp sağlığını destekler.

Literatürde yapılan çalışmalar, KVH odaklı mHealth uygulamalarının klinik etkinlik ve maliyet etkinlik açısından değerlendirilmiş olduğunu göstermektedir. Bu tür uygulamalar, sağlık hizmetlerinin maliyetlerini düşürerek, hastalara daha iyi bir destek sunmaktadır.

### **2.1.3. Mevcut Kardiyovasküler Sağlık Mobil Uygulamalarının Sınırlılıkları ve Araştırma Boşlukları**

Mevcut KVH mobil uygulamalarının etkinliğine rağmen, çeşitli sınırlılıkları ve zorlukları bulunmaktadır:

- **Kullanıcı Uyumu:** Kullanıcıların mobil uygulamaları düzenli ve etkili bir şekilde kullanmalarını sağlamak zor olabilmektedir. Düşük kullanıcı uyumu,

uygulamaların etkinliğini azaltmaktadır.

- **Veri Güvenliği:** KVH ile ilgili sağlık verilerinin güvenliği, kullanıcıların gizliliği açısından kritik öneme sahiptir. Ancak, mevcut uygulamalarda veri güvenliği ile ilgili endişeler bulunmaktadır.
- **Doğruluk ve Klinik Validasyon Eksikliği:** Birçok mHealth uygulaması, klinik çalışmalarda yeterince test edilmemiştir. Dolayısıyla, bu uygulamaların sağladığı bilgilerin doğruluğu ve güvenilirliği konusunda belirsizlikler mevcuttur.

### **Araştırma Boşlukları**

Literatürde, kardiyovasküler sağlıkta yapay zeka destekli mobil uygulamaların potansiyeli ile ilgili yeterli araştırma yapılmamıştır. Yapay zeka ve makine öğrenimi tekniklerinin entegrasyonu, veri analizi ve risk tahmini süreçlerinde önemli fırsatlar sunabilir [2].  
Örneğin:

- **Yapay Zeka Entegrasyonu:** Uygulamalarda yapay zeka kullanarak, kullanıcıların kalp sağlığı durumu hakkında daha doğru tahminler yapmak mümkündür. EKG analizi işlemleri gibi gelişmiş özellikler, hastalıkların erken teşhis edilmesini destekleyebilir.

Projemiz, bu araştırma boşluklarını doldurarak, yapay zeka entegrasyonu, veri analiz yöntemleri ve kullanıcı deneyimini iyileştirecek özelliklerle mevcut sınırlılıkların üstesinden gelmeyi hedefleyebilir. Böylece, kardiyovasküler sağlıkta mobil uygulamaların etkinliğini artırmak ve kullanıcılarının iyi kalp sağlığına yönelik hedeflerine ulaşmalarına yardımcı olmak mümkün olacaktır.

## **2.2. Yapay Zeka ve Makine Öğrenmesi Sağlık Alanında (EKG Analizi ve Kardiyovasküler Risk Tahmini Odaklı)**

### **2.2.1. Sağlıkta Yapay Zeka ve Makine Öğrenmesi Uygulamalarına Genel Bakış**

Tanım, Kapsam ve Potansiyel Faydaları: Yapay zeka (YZ), insan zekasına özgü olan düşünme, karar verme, konuşma, görsel algı ve yorumlama gibi yeteneklerin bilgisayar sistemlerine aktarılmasıdır . Sağlıkta YZ ve makine öğrenmesi (ML) uygulamalarının potansiyel faydaları arasında, sağlık personelinin iş yükünün azalması, süreçlerin hızlanması ve zaman tasarrufu sağlanması yer almaktadır. Ek olarak, tıbbi hataların ve malpraktis davalarının azalması gibi avantajlar da bulunmaktadır.

Farklı Alanlardaki Örnekler: Sağlıkta YZ uygulamaları; görüntü işleme, doğal dil işleme, genomik analiz, ilaç keşfi ve kişiselleştirilmiş tıp gibi çeşitli alanlarda kullanılmaktadır. YZ, idari süreçlerde sağlık yönetimi, dokümantasyon yönetimi, hastane kapasitesinin verimli kullanılması, hataların azaltılması gibi amaçlarla kullanılabilirken, klinik uygulamalarda erken tanı, acil müdahale, görüntüleme analiz araçları, YZ destekli robotik ameliyatlar, kişiye özel tedavi ve ilaç geliştirme gibi alanlarda kullanılmaktadır.

Etik, Yasal ve Sosyal Boyutlar: Sağlıkta YZ kullanımının veri gizliliği, algoritmik önyargı, hasta güvenliği ve doktor sorumluluğu gibi önemli etik, yasal ve sosyal boyutları bulunmaktadır. Tıbbi hata durumlarında sorumluluğun kime yükleneceği konusundaki belirsizlikler ve kişisel sağlık verilerinin gizliliği ile ilgili riskler etik sorunlara yol açabilir.

Klinik Ortamlara Entegrasyon Zorlukları ve Çözüm Önerileri: Sağlıkta YZ uygulamalarının klinik ortamlara entegrasyonunda veri gizliliği ve işlerin otomasyonu gibi konularda etik kaygılar ortaya çıkabilir. Bu kaygıların giderilmesi, etik standartların belirlenmesi, insan unsurunun önceliğinin vurgulanması ve sağlık profesyonellerinin YZ hakkında bilgilendirilmesi entegrasyon sürecini kolaylaştırabilir [3].

## 2.3. React Native ve Expo ile Mobil Sağlık Uygulaması Geliştirme

Mobil sağlık uygulamaları, sağlık hizmetlerine erişimi kolaylaştıran, kullanıcıların sağlık durumlarını izlemesine ve sağlık hizmet sağlayıcılarıyla iletişim kurmasına olanak tanıyan yenilikçi çözümler sunar. Bu bağlamda, React Native ve Expo gibi modern mobil uygulama geliştirme araçları, hem platformlar arası geliştirme kolaylığı hem de hızlı prototipleme olanaklarıyla bu tür uygulamaların geliştirilmesinde önemli bir rol oynamaktadır. Bu bölümde, React Native ve Expo'nun mobil sağlık uygulamaları geliştirme bağlamındaki önemi, literatürdeki yeri ve bu teknolojilerin sağlık sektörü uygulamaları için sunduğu avantajlar ele alınmaktadır.

### 2.3.1. React Native ve Expo Frameworklerine Genel Bakış

#### React Native ve Expo'nun Temel Özellikleri ve Avantajları

React Native, Facebook tarafından geliştirilen, JavaScript tabanlı bir çerçevedir ve geliştiricilere tek bir kod tabanı ile hem iOS hem de Android platformları için uygulama geliştirme imkanı sunar. React Native, "native-like" bir deneyim sağlayarak yüksek performanslı ve kullanıcı dostu uygulamalar oluşturulmasında kullanılır. Expo ise React Native çerçevesinin yeteneklerini genişleten bir platformdur ve geliştiricilere hızlı prototipleme, kolay test süreçleri ve cihaz sensörlerine erişimi sağlayan kullanıma hazır API'ler sunar [4].

React Native ve Expo'nun avantajları şunlardır:

**Platformlar Arası Geliştirme:** Tek bir kod tabanı ile hem iOS hem de Android için uygulama geliştirme imkanı sunar, bu da geliştirme sürecini hızlandırır ve maliyetleri düşürür.

**Hızlı Geliştirme Süreçleri:** Expo'nun sunduğu hazır bileşenler ve araçlar, hızlı prototipleme ve uygulama yayınlama süreçlerini kolaylaştırır.

**Geniş Topluluk Desteği:** React Native ve Expo, geniş bir geliştirici topluluğu ve zengin bir eklenti ekosistemi sayesinde hızlı çözüm geliştirme imkanı sağlar.

**Kullanıcı Deneyimi (UX):** React Native, native bileşenlere yakın bir performans ve kullanıcı deneyimi sunar.

Ancak, React Native ve Expo'nun bazı dezavantajları da bulunmaktadır:

**Performans Sınırlamaları:** Özellikle yüksek işlem gücü gerektiren uygulamalarda, tamamen native bir uygulamaya kıyasla performans kayıpları yaşanabilir.

**Karmaşık Native Modüller:** Bazı özel donanım erişim ihtiyaçlarında, native kod yazma gerekliliği ortaya çıkabilir.

**Güvenlik Limitasyonları:** React Native uygulamalarında, özellikle sağlık sektörü gibi yüksek güvenlik gerektiren alanlarda, ekstra önlemler alınması gerekebilir.

### **Mobil Uygulama Geliştirme Ekosistemindeki Yeri ve Popülerliği**

React Native, günümüzde mobil uygulama geliştirme ekosisteminde en çok tercih edilen çerçevelerden biridir. 2023 yılı itibarıyla, dünya çapında birçok büyük şirket (örneğin Facebook, Instagram, Airbnb, Uber Eats) bu çerçeveyi kullanmaktadır [5]. Expo ise React Native uygulamalarını daha hızlı ve kolay bir şekilde geliştirmek için kullanılan popüler bir araç seti haline gelmiştir.

### **Başarılı Mobil Uygulama Örnekleri**

React Native ve Expo kullanılarak geliştirilen başarılı uygulamalardan bazıları şunlardır:

**Facebook ve Instagram:** React Native'in kullanımına öncülük eden Facebook, bu çerçeveyi Instagram gibi birçok uygulamasında kullanmaktadır.

**Gyroscope:** Sağlık ve fitness alanında bir uygulama olan Gyroscope, kullanıcıların sağlık verilerini izlemesine olanak sağlar ve React Native ile geliştirilmiştir.

**AccuWeather:** Hava durumu uygulaması olan AccuWeather, Expo kullanılarak geliştirilerek hızlı prototipleme avantajlarından faydalanmıştır.

### 2.3.2. Mobil Sağlık Uygulamaları Geliştirmede React Native ve Expo Kullanımı

#### Performans, UI/UX, Sensör Erişimi ve Güvenlik Açısından Değerlendirme

React Native ve Expo, mobil sağlık uygulamalarında aşağıdaki avantajları sunar:

- **Performans:** React Native, native modüller kullanarak yüksek performanslı uygulamalar geliştirilmesine olanak tanır. Ayrıca, Expo'nun sunduğu hazır API'ler, cihaz sensörlerine (örneğin, kalp atış hızı monitörleri, adım sayıcılar) kolayca erişim sağlar [4].
- **UI/UX:** React Native, kullanıcı dostu ve modern tasarımlar oluşturmak için güçlü bir araçtır. Sağlık uygulamaları için kritik olan kullanıcı deneyimi, React Native'in esnek bileşenleri sayesinde optimize edilebilir.
- **Güvenlik:** Sağlık uygulamalarında kişisel verilerin korunması önceliklidir. Ancak, React Native uygulamalarında güvenlik önlemleri, genellikle ek kütüphaneler ve şifreleme yöntemleriyle sağlanır. Özellikle HIPAA gibi düzenlemelere uyum sağlamak için özel güvenlik protokollerinin uygulanması gerekebilir.

#### Sağlık Verisi Yönetimi ve Düzenlemelere Uyumluluk

Sağlık uygulamalarında güvenlik ve veri gizliliği kritik öneme sahiptir. React Native ve Expo, doğrudan HIPAA (Health Insurance Portability and Accountability Act) gibi düzenlemelere uyumlu bir yapı sunmasa da, bu uyumluluğu sağlamak için ek güvenlik önlemleri alınabilir. Örneğin:

**Veri Şifreleme:** Sağlık verilerinin şifrelenmesi için React Native uygulamalarında AES (Advanced Encryption Standard) gibi şifreleme yöntemleri kullanılabilir.

**Güvenli Giriş:** Kullanıcıların kimlik doğrulama süreçlerinde biyometrik doğrulama veya OAuth gibi güvenli giriş yöntemleri uygulanabilir.



## **Hızlı Prototipleme ve MVP Geliştirme Süreçlerindeki Avantajları**

Expo, sağlık uygulamalarında hızlı prototipleme ve MVP geliştirme sürecini kolaylaştırır.

Örneğin:

**Hızlı Test Süreçleri:** Expo Go uygulaması sayesinde, geliştirilmekte olan bir uygulama gerçek cihazlarda hızlıca test edilebilir.

**Minimum Kodlama Gereksinimi:** Expo'nun sunduğu hazır bileşenler, sağlık uygulamalarını hızlı bir şekilde geliştirmek için idealdir.

## **2.4. Firebase Veritabanı ve Bulut Bilişim ile Sağlık Verisi Yönetimi**

Sağlık sektörü, dijital dönüşümle birlikte verilerin güvenli depolanması, hızlı erişimi ve paylaşılabilirliği açısından bulut bilişim teknolojilerinden büyük ölçüde faydalanmaktadır. Firebase, Google tarafından sağlanan bir Backend-as-a-Service (BaaS) platformu olarak, sağlık uygulamalarında veri yönetimi için uygun bir çözüm sunmaktadır. Bu bölümde, Firebase platformunun sağlık uygulamaları için sunduğu avantajlar ve bulut bilişimin sağlık verisi yönetimi bağlamındaki rolü ele alınmaktadır.

### **2.4.1. Firebase Platformuna Genel Bakış ve Sağlık Uygulamaları İçin Uygunluğu**

**Firestore'in Temel Özellikleri ve Sunduğu Servisler**

Firestore, mobil ve web uygulamaları için eksiksiz bir backend altyapısı sunan güçlü bir platformdur. Sağlık uygulamaları dahil olmak üzere birçok sektörde kullanılabilecek bir dizi hizmet sunar:

**Cloud Firestore:** Gerçek zamanlı ve ölçeklenebilir bir veritabanı çözümü. Sağlık verilerinin hızlı bir şekilde yazılıp okunmasını sağlar.

**Firestore Authentication:** Uygulama kullanıcılarının güvenli bir şekilde kimlik doğrulamasını gerçekleştiren bir hizmettir. E-posta/şifre, telefon numarası veya üçüncü taraf sağlayıcılar (Google, Facebook) ile entegrasyon sağlar.

**Cloud Functions:** Sunucusuz işlevler sayesinde, sağlık verilerinin işlenmesi gibi özel backend işlemleri kolayca gerçekleştirilebilir.

**Firestore Hosting:** SSL destekli güvenli bir ortamda uygulama barındırma hizmeti sunar.

**Firestore Analytics:** Kullanıcı davranışlarını izleyerek, uygulamanın kullanımını analiz etme ve optimize etme imkanı sunar.

### **Firestore'in Mobil Uygulama Geliştirme ve Backend Altyapısı İçin Avantajları**

Firestore, sağlık uygulamaları geliştirme sürecinde aşağıdaki avantajları sunar:

**Ölçeklenebilirlik:** Firestore, artan kullanıcı sayısına göre otomatik olarak ölçeklenebilir bir altyapı sunar. Sağlık uygulamaları gibi büyük miktarda veri işleyen uygulamalarda bu özellik önemlidir [6].

**Maliyet Etkinliği:** Firestore, küçük uygulamalardan büyük projelere kadar farklı bütçelere uygun bir çözüm sunar. Ücretsiz plan, hızlı prototipleme için idealdir.

**Kolay Entegrasyon:** React Native ve Expo gibi frontend çerçeveleriyle kolayca entegre edilebilir, bu da sağlık uygulamalarının hızlı geliştirilmesini sağlar.

**Gerçek Zamanlı Veri Erişimi:** Cloud Firestore, sağlık uygulamalarında hasta verilerinin gerçek zamanlı olarak güncellenmesini sağlar. Örneğin, bir doktor hastanın verilerine anında erişebilir.

### **Firestore'in Sağlık Verisi Yönetimi ve Depolama İçin Uygunluğu**

Sağlık verilerinin yönetimi, veri gizliliği ve güvenliği açısından bir dizi gereksinim içerir. Firestore, bu ihtiyaçlara şu şekillerde yanıt verebilir:

**Veri Güvenliği:** Firestore, verilerin güvenliğini sağlamak için şifreleme (hem veri aktarımı sırasında hem de veri depolama sırasında) ve güvenlik kuralları sunar.

**HIPAA Uyumluluğu:** Firestore'in kendisi doğrudan HIPAA uyumluluğu sunmasa da, Google Cloud Platform (GCP) üzerinde HIPAA uyumlu bir çözüm oluşturulabilir.

Google, HIPAA uyumlu hizmetler için Business Associate Agreement (BAA) sağlar [6].

**Ölçeklenebilirlik:** Sağlık verilerinin artan hacmi, Firebase'in ölçeklenebilir yapısıyla kolayca yönetilebilir.

### **Firestore Kullanılarak Geliştirilen Sağlık Uygulamalarından Örnekler**

**CareClinic:** Firestore kullanılarak geliştirilen bu uygulama, kullanıcıların tedavi planlarını yönetmelerine, sağlık verilerini takip etmelerine ve doktorlarıyla paylaşımlarına olanak tanır.

**MyTherapy:** İlaç takibi ve sağlık izleme uygulaması olan MyTherapy, Firestore'in gerçek zamanlı veri senkronizasyonu özelliklerinden faydalanmaktadır.

**Fitly:** Firestore altyapısını kullanan bu uygulama, kullanıcıların diyet ve fitness hedeflerini izlemelerine yardımcı olur.

### **2.4.2. Bulut Bilişim ve Sağlık Verisi Güvenliği ve Gizliliği**

#### **Bulut Bilişimin Sağlık Sektöründeki Rolü ve Avantajları**

Bulut bilişim, sağlık sektöründe veri yönetimi ve güvenliği açısından birçok avantaj sunar:

**Veri Depolama ve Paylaşım:** Hasta verileri merkezi bir bulut ortamında saklanabilir ve sağlık hizmeti sağlayıcıları arasında kolayca paylaşılabilir.

**Erişilebilirlik:** Bulut tabanlı sistemler, sağlık profesyonellerinin verilere herhangi bir yerden erişmesine olanak tanır.

**Veri Analizi:** Büyük veri analitiği ve yapay zeka için uygun bir altyapı sağlayarak, sağlık hizmetlerinin daha verimli ve doğru bir şekilde sunulmasını sağlar.

#### **Sağlık Verisi Güvenliği ve Gizliliği İçin Dikkat Edilmesi Gerekenler**

**Şifreleme:** Veri aktarımı sırasında (TLS/SSL) ve depolama sırasında (AES-256) şifreleme yöntemleri kullanılmalıdır.

**Eriřim Kontrolü:** Saęlık verilerine yalnızca yetkili kiřilerin eriřmesini saęlamak için güçlü kimlik doęrulama ve yetkilendirme mekanizmaları uygulanmalıdır.

**Uyumluluk Standartları:** Saęlık verisi yönetimi, HIPAA, GDPR, ve ISO 27001 gibi standartlara uygun řekilde yapılmalıdır. Firebase, bu standartlarla uyumlu bir altyapı sunmak için gerekli araçları saęlar [6].

### **Literatürde Bulut Tabanlı Saęlık Veri Yönetimi Sistemlerinin Güvenlik Riskleri ve Çözüm Önerileri**

Literatürde bulut biliřim tabanlı sistemler için bazı güvenlik riskleri ve çözüm önerileri tartışılmaktadır:

**Risk:** Saęlık verilerinin bulutta saklanması, siber saldırılara ve veri ihlallerine açık olabilir.

**Çözüm:** Veri řifreleme, düzenli güvenlik taramaları ve güvenlik duvarları uygulanmalıdır.

**Risk:** Saęlık verilerinin yetkisiz eriřime açık hale gelmesi.

**Çözüm:** Güçlü eriřim kontrol mekanizmaları ve iki faktörlü kimlik doęrulama (2FA) kullanılmalıdır.

### **Firebase'in Saęlık Verisi Güvenlięi ve Uyumluluk Özellikleri**

Firebase, saęlık uygulamalarında veri güvenlięi ve gizlilięi saęlamak için ařaęıdaki özellikleri sunar:

**Şifreleme:** Firebase, tüm verileri hem aktarım sırasında (TLS) hem de depolama sırasında řifreler.

**Eriřim Kontrolü:** Firebase Security Rules, uygulama verilerine eriřimi kontrol etmek için özelleřtirilebilir bir yapı sunar.

**HIPAA Uyumluluęu:** Firebase'in HIPAA uyumluluęu, Google Cloud Platform'un ilgili

hizmetleriyle sağlanabilir. Ancak, GCP üzerinde HIPAA uyumlu bir çözüm oluşturmak için BAA (Business Associate Agreement) imzalanması gereklidir [7].

### 3. MATERYALLER VE METODLAR

#### 3.1.1 PTB Diagnostic ECG Veri Seti

PTB Diagnostic ECG Database, kardiyak anormalliklerin teşhisi için özel olarak tasarlanmış kapsamlı bir veri setidir. Bu veri seti, 14,552 tek kurşunlu elektrokardiyogram (ECG) kaydını içerir. Her kayıt, 187 sütundan oluşur ve 125Hz örnekleme frekansında kaydedilmiştir. Bu yüksek örnekleme frekansı, ECG sinyallerinin ayrıntılı analizine olanak tanınması açısından önemlidir.

Veri seti, normal kalp atışları ve anormal kalp atışları olmak üzere iki ana sınıfa ayrılmıştır. Normal kalp atışları sağlıklı bireylerden elde edilen ECG kayıtlarını içermektedir. Anormal kalp atışları ise çeşitli kardiyak anormalliklerden etkilenmiş bireylerden elde edilen ECG kayıtlarını içerir. Anormallikler arasında miyokard enfarktüsü, kalp yetmezliği ve kardiyak aritmiler bulunur.

Veri seti, iki ayrı CSV dosyasında saklanır: "ptbdb\_normal.csv" ve "ptbdb\_abnormal.csv". "ptbdb\_normal.csv" dosyası 4,046 normal ECG kaydı içerirken, "ptbdb\_abnormal.csv" dosyası 10,506 anormal ECG kaydı içerir [8].

#### 3.1.2 Veri Seti Görselleştirme Detayları

Veri setinin görselleştirilmesi, normal ve anormal ECG sinyalleri arasındaki farklılıkları anlamak için kritik öneme sahiptir. Bu çalışmada, veri setinin görselleştirilmesi için çeşitli teknikler kullanılmıştır:

**Rastgele Örnek Çizimleri:** Normal ve anormal veri setlerinden rastgele seçilen örnekler, yan yana çizilerek karşılaştırılmıştır. Bu, normal ve anormal sinyaller arasındaki görsel farklılıkların doğrudan gözlemlenmesini sağlar. Özellikle, QRS kompleksindeki değişiklikler, ST segmentindeki yükselmeler veya çökmeler ve T dalgasındaki anormallikler gibi farklılıklar incelenmiştir.

**Düzeltilmiş Ortalama Çizimleri:** Her sınıfın ortalama sinyali, kayan ortalama ve standart sapma kullanılarak düzeltilmiştir. Kayan ortalama, sinyallerdeki gürültüyü azaltırken, standart sapma sinyallerdeki varyasyonu gösterir. Bu, sınıflar arasındaki genel eğilimleri ve varyasyonların karşılaştırılmasına olanak tanır. Özellikle, normal ve anormal sinyaller arasındaki temel çizgi kaymaları, tepe genlikleri ve dalga şekli farklılıkları incelenmiştir.

### 3.1.3 Veri Ön İşleme Detayları

Veri ön işleme adımları, modelin performansını artırmak için kritik öneme sahiptir. Bu çalışmada, aşağıdaki ön işleme adımları uygulanmıştır:

- **Veri Yükleme:** CSV dosyaları Pandas kütüphanesi kullanılarak yüklenmiştir. Pandas, büyük veri setlerinin verimli bir şekilde işlenmesine olanak tanır.
- **Sınıf Etiketleri Ekleme:** Normal ve anormal veri setlerine sınıf etiketleri eklenmiştir ("Normal" ve "Anomaly"). Bu etiketler, modelin denetimli öğrenme için eğitilmesini sağlar.
- **Veri Birleştirme:** Normal ve anormal veri setleri, analiz ve modelleme için tek bir DataFrame'de birleştirilmiştir.
- **Eğitim ve Test Veri Setlerine Bölme:** Normal veri seti, %15 test boyutu ile eğitim ve test veri setlerine ayrılmıştır. Anormal veri seti, tamamen test verisi olarak kullanılmıştır. Bu, modelin normal verileri öğrenmesini ve anormal verileri tespit etmesini sağlar.

### 3.1.4 Model Mimarisi Detayları

Bu çalışmada, anormal ECG sinyallerini tespit etmek için bir Evrişimsel Sinir Ağı - Convolutional Neural Network (CNN) modeli kullanılmıştır. Model, encoder (kodlayıcı) ve decoder (kod çözücü) olmak üzere iki ana bölümden oluşmaktadır. Encoder, 187 veri noktasından oluşan ECG sinyalini alır ve 1D evrişim, batch normalizasyon ve maksimum havuzlama katmanları kullanarak 32 boyutlu bir latent uzay oluşturur. Decoder, bu latent uzaydan orijinal sinyali yeniden oluşturmak için 1D transpoze evrişim, batch normalizasyon ve düzleştirme katmanlarını kullanır.

Model, Adam optimize edici ve Ortalama Mutlak Hata (MAE) kaybı ile derlenmiştir. Adam, öğrenme oranını uyarlayan ve hızlı yakınsamaya olanak tanıyan etkili bir optimize edicidir. MAE kaybı, yeniden oluşturulmuş sinyal ile orijinal sinyal arasındaki farkı ölçer.

### 3.1.5 Model Eğitimi ve Değerlendirme Detayları

Model, eğitim veri seti üzerinde 100 epoch boyunca eğitilmiştir. Erken durdurma (Early Stopping) tekniği, doğrulama kaybının iyileşmemesi durumunda eğitimi durdurmak için kullanılmıştır.

Modelin performansı, test ve anormal veri setleri üzerinde MAE kaybı kullanılarak değerlendirilmiştir. Ayrıca, normal ve anormal sinyaller arasındaki ayrım için bir eşik değeri belirlenmiştir. Eşik değeri, eğitim veri setinin MAE kaybının ortalaması ve standart sapması kullanılarak hesaplanmıştır.

### 3.2.1 Kardiyovasküler Hastalık (KVH) Veri Seti

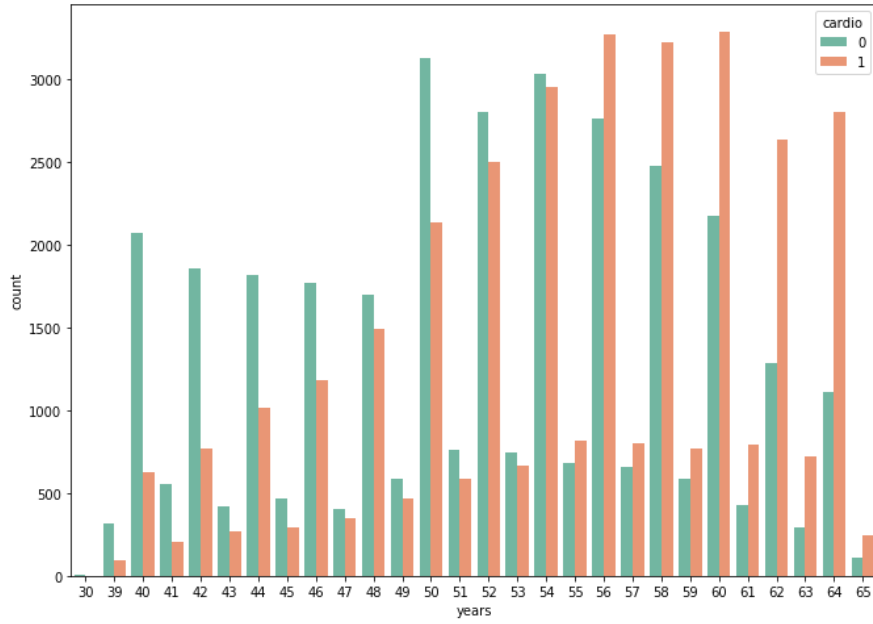
Kardiyovasküler hastalıkların (KVH) teşhis ve tahmini için kullanılan bu tıbbi veri seti, 70.000 hasta kaydı içermektedir. Her bir kayıt, 12 farklı özellik ve bir hedef değişken içermektedir. Veri seti; hastaların yaş, boy, kilo, kan basıncı, kolesterol düzeyi, glikoz seviyesi, sigara içme durumu, alkol tüketimi ve fiziksel aktivite gibi bilgilerini içermektedir. Bu veriler, hastaların KVH durumunu tahmin etmek amacıyla kullanılmaktadır.

Değişken Adı	Açıklama	Birim/Değerler
Yaş (age):	Hastaların yaşı	Gün
Boy (height):	Hastaların boyu	Cm
Kilo (weight):	Hastaların kilosu	Kg
Cinsiyet	Cinsiyet	1: kadın, 2: erkek
Sistolik kan basıncı (ap_hi):	Sistolik kan basıncı	mmHg
Diastolik kan basıncı (ap_lo):	Diastolik kan basıncı	mmHg
Kolesterol düzeyi (cholesterol)	Kolesterol seviyeleri	1: normal, 2: yüksek normal, 3: çok yüksek
Glikoz düzeyi (gluc)	Glikoz seviyeleri	1: normal, 2: yüksek normal, 3: çok yüksek
Sigara içme durumu (smoke)	Sigara içme durumu	0: hayır, 1: evet
Alkol tüketimi (alco)	Alkol tüketimi durumu	0: hayır, 1: evet
Fiziksel aktivite (active)	Fiziksel aktivite durumu	0: hayır, 1: evet
Kardiyovasküler hastalık durumu (cardio)	KVH durumu	0: yok, 1: var

### 3.2.2 Veri Seti Görselleştirme Detayları

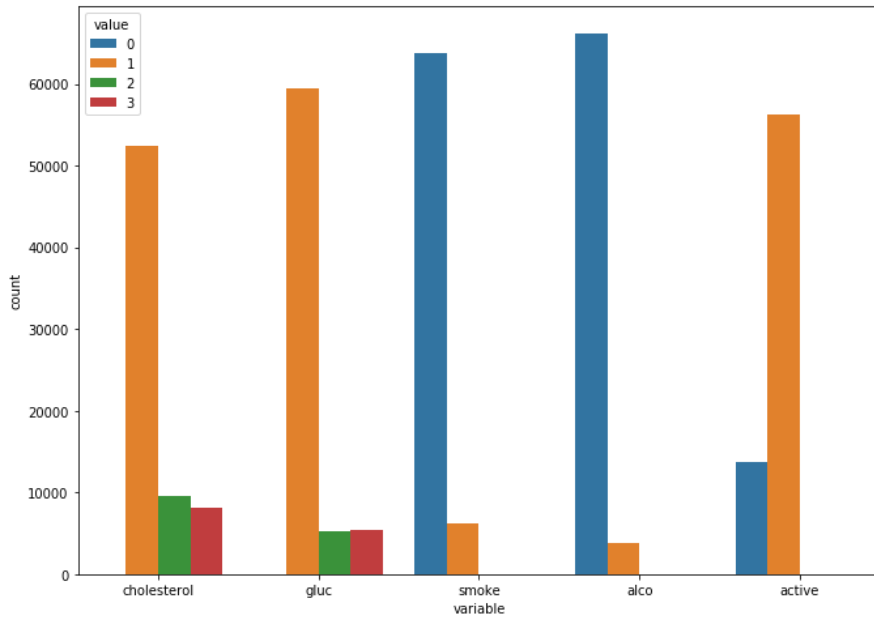
Veri setinin görselleştirilmesi, kardiyovasküler hastalık (KVH) verilerinin dağılımını anlamak ve bu hastalıkla ilgili belirli özellikler arasındaki ilişkileri keşfetmek için kritik bir öneme sahiptir. Bu süreç, veri集中的indeki değişkenlerin etkili bir şekilde analiz edilmesini sağlayarak, KVH'nin altında yatan potansiyel risk faktörlerini ortaya çıkarmaya yardımcı olur.

Öncelikle, yaş ve KVH arasındaki ilişkiyi incelemek için histogramlar ve sayım grafikleri (countplot) kullanılmıştır. Histogramlar, yaş dağılımını gösterirken, sayım grafikleri farklı yaş gruplarındaki KVH görülme sıklığını karşılaştırmalı olarak sunmuştur. Bu analizler sonucunda, özellikle 55 yaş ve üzerindeki bireylerin KVH riskinin belirgin şekilde arttığı gözlemlenmiştir. Bu bulgu, yaşın KVH için önemli bir risk faktörü olduğunu doğrulamaktadır.



Şekil 1. Yaş ve KVH arasındaki ilişkiyi açıklayan histogram

Kategorik değişkenlerin (kolesterol, glikoz, sigara içme, alkol tüketimi ve fiziksel aktivite) KVH durumu ile ilişkisi, sütun grafikleri ve yığılmış çubuk grafikleri gibi görselleştirme yöntemleri ile analiz edilmiştir. Bu grafikler, her bir kategorik değişkenin farklı seviyelerindeki bireylerin KVH görülme oranlarını karşılaştırmalı olarak göstermiştir. Örneğin, yüksek kolesterol veya glikoz seviyelerine sahip bireylerin KVH riskinin daha yüksek olduğu tespit edilmiştir. Aynı şekilde, sigara içen ve düzenli alkol tüketen bireylerde de KVH riskinin arttığı gözlemlenmiştir. Fiziksel olarak aktif olmayan bireylerde ise KVH riskinin daha yüksek olduğu görülmüştür.



Şekil 2. Kategorik Değişkenler ile KVH arasındaki ilişkiyi açıklayan histogram



Cinsiyetin KVH üzerindeki etkisini ve diğer risk faktörleriyle olan ilişkisini incelemek için gruplandırılmış sütun grafikleri ve kutu grafikleri kullanılmıştır. Bu grafikler, cinsiyetler arasında KVH görülme oranlarını karşılaştırmanın yanı sıra, alkol tüketimi gibi diğer değişkenlerin cinsiyetler arasındaki dağılımını da göstermiştir. Bu analizler, erkeklerin KVH riskinin kadınlara göre daha yüksek olduğunu ve alkol tüketimi gibi bazı risk faktörlerinin cinsiyete göre farklılık gösterdiğini ortaya koymuştur.

### 3.2.3 Veri Ön İşleme Detayları

Model performansını optimize etmek için veri ön işleme aşaması büyük bir titizlikle gerçekleştirilmiştir. Veri setinde herhangi bir eksik veri bulunmamasına rağmen, fiziksel olarak tutarsızlık gösteren aykırı değerler (özellikle boy ve kilo değişkenlerinde) tespit edilmiş ve uygun yöntemlerle temizlenmiştir. Ayrıca, fizyolojik olarak mümkün olmayan diyastolik ve sistolik kan basıncı değerleri de düzeltilmiştir. Bu işlemler, veri setinin güvenilirliğini ve tutarlılığını artırarak modelin daha sağlıklı verilerle çalışmasını sağlamayı amaçlamaktadır.

Modelin öğrenme hızını ve verimliliğini artırmak amacıyla, sayısal değişkenler üzerinde normalizasyon işlemi uygulanmıştır. Bu işlem, değişkenlerin ölçeklerini eşitleyerek modelin eğitim sürecini optimize etmektedir. Bu ön işleme adımları, veri setinin kalitesini yükselterek modelin daha doğru ve güvenilir tahminler yapmasına olanak tanımaktadır.

### 3.2.4 Model Mimarisi Detayları

Kardiyovasküler hastalık (KVH) tahmininde, çeşitli sınıflandırma algoritmaları kullanılmıştır. Bu algoritmalar arasında, lojistik regresyon, karar ağaçları, rastgele ormanlar (random forests) ve destek vektör makineleri (support vector machines - SVM) yöntemleri bulunmaktadır. Bu modellerin temel amacı, veri setindeki örüntüleri ve ilişkileri öğrenerek, yeni hasta verileri üzerinden KVH durumunu doğru bir şekilde tahmin etmektir.

Model geliştirme sürecinde, veri seti %80 eğitim ve %20 test olmak üzere ikiye ayrılmıştır. Eğitim verileri, modellerin öğrenmesi için kullanılırken, test verileri modelin performansını değerlendirmek için ayrılmıştır.

### 3.2.5 Model Eğitimi ve Değerlendirme Detayları

Modelin eğitimi sürecinde, 50 ila 100 epoch arasında bir eğitim döngüsü uygulanmıştır. Bu süreçte, modelin aşırı öğrenmesini (overfitting) engellemek amacıyla erken durdurma (early stopping) tekniği kullanılmıştır. Erken durdurma, modelin eğitim verisi üzerindeki performansının iyileşmeyi durdurduğu noktada eğitimi sonlandırarak, modelin genelleme yeteneğini artırmayı hedefler.

## 4. MOBİL UYGULAMA GELİŞTİRME

Bu bölümde, Medora mobil uygulamasının geliştirilme sürecine genel bir bakış sunulmaktadır. Uygulama, çapraz platform uyumluluğu ve hızlı geliştirme avantajları sunan React Native çatısı ve Expo geliştirme platformu kullanılarak inşa edilmiştir. Kullanıcı kimlik doğrulama ve veri yönetimi için Firebase servisleri entegre edilmiştir. Bu bölüm, uygulamanın mimari tasarımını, geliştirme sürecini ve Firebase entegrasyon detaylarını kapsamlı bir şekilde ele almaktadır.

### 4.1. Uygulama Mimarisi ve Tasarımı

Bu bölüm, Medora uygulamasının yazılım mimarisine ve tasarım kararlarına odaklanmaktadır. Uygulamanın nasıl yapılandırıldığı, temel bileşenleri ve tasarım prensipleri açıklanmaktadır.

#### Uygulama Mimarisi ve Katmanlı Tasarım

**Bileşen Tabanlı Mimari:** Medora uygulaması, React Native'in bileşen tabanlı mimarisi üzerine inşa edilmiştir. Arayüz, tekrar kullanılabilir ve modüler bileşenler halinde yapılandırılmıştır. Bu yaklaşım, kodun okunabilirliğini, bakımını ve yeniden kullanılabilirliğini artırmaktadır. Uygulama arayüzü, components klasöründe yer alan özel bileşenler (örneğin Button, Input, Typo, ScreenWrapper) aracılığıyla tutarlı bir tasarım dilini yansıtmaktadır.

**Katmanlı Yapı ve Klasör Organizasyonu:** Proje, işlevselliğe göre katmanlara ayrılmıştır.

- **app dizini:** Expo Router ile dosya tabanlı yönlendirme kullanılarak uygulama ekranlarını ve navigasyonu içerir. Özellikle (auth) ve (tabs) gruplamaları, kimlik doğrulama akışı ve ana uygulama sekmelerini net bir şekilde ayırır.

- **components dizini:** Tekrar kullanılabilir arayüz bileşenlerini barındırır. Bu, UI tutarlılığını ve geliştirme hızını artırır.
- **config dizini:** Firebase yapılandırma dosyalarını içerir (firebase.ts).
- **contexts dizini:** Uygulama genelinde paylaşılan durum yönetimini (örneğin, authContext.tsx ile kullanıcı kimlik doğrulama durumu) sağlar. Context API kullanılarak global durum yönetimi basit ve etkili bir şekilde uygulanmıştır.
- **utils dizini:** Yardımcı fonksiyonları ve stil araçlarını (örneğin, styling.ts ile ekran boyutlarına göre ölçekleme) içerir.

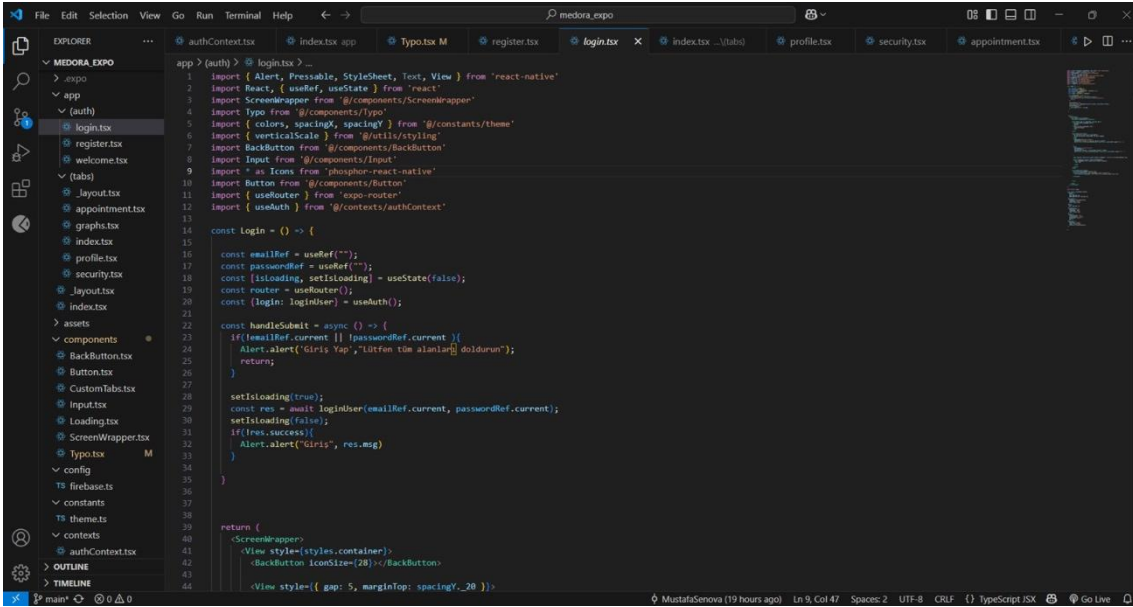
**Navigasyon Mimarisi (Expo Router):** Expo Router kullanılarak dosya sistemi tabanlı bir navigasyon yapısı benimsenmiştir. app dizini içindeki dosyalar ve klasörler, uygulama rotalarını otomatik olarak tanımlar. Bu yaklaşım, navigasyon yönetimini basitleştirir ve geliştirme sürecini hızlandırır. Sekmeler arası geçişler için (tabs)\\_layout.tsx ve özel CustomTabs.tsx bileşeni kullanılmıştır.

**Durum Yönetimi (Context API):** Uygulama genelindeki kimlik doğrulama durumu (user bilgisi) Context API aracılığıyla yönetilmektedir. authContext.tsx dosyası, AuthProvider ile tüm uygulamayı sararak, bileşenlerin kimlik doğrulama durumuna kolayca erişmesini ve güncelleyebilmesini sağlar. Bu, prop drilling sorununu önler ve durum yönetimini merkezileştirir.

**UI/UX Tasarım İlkeleri:** Uygulama arayüzü, kullanıcı deneyimini ön planda tutan prensiplerle tasarlanmıştır.

- **Tutarlılık:** components klasöründeki tekrar kullanılabilir bileşenler ve constants/theme.ts dosyasındaki renk ve stil tanımlamaları sayesinde uygulama genelinde görsel tutarlılık sağlanmıştır.
- **Responsive Tasarım:** utils/styling.ts dosyasındaki scale ve verticalScale fonksiyonları, farklı ekran boyutlarına uyum sağlamak için kullanılmıştır. Bu sayede uygulama, çeşitli cihazlarda tutarlı bir görünüm sunar.
- **Kullanılabilirlik:** Giriş ve kayıt ekranlarında kullanıcı girdilerini kolaylaştıran Input bileşenleri ve geri bildirim sağlayan Alert kullanımı, kullanıcı dostu bir arayüz oluşturur.

**Modülerlik ve Ölçeklenebilirlik:** Bileşen tabanlı mimari ve katmanlı yapı, uygulamanın modülerliğini ve ölçeklenebilirliğini artırır. Yeni özelliklerin eklenmesi veya mevcut özelliklerin güncellenmesi, projenin genel yapısını bozmadan kolayca yapılabilir.



Şekil 3. Proje yapısı

## 4.2. React Native ve Expo ile Mobil Uygulama Geliştirme Süreci

Bu bölüm, Medora uygulamasının React Native ve Expo kullanılarak geliştirilme sürecini detaylandırmaktadır. Geliştirme ortamı, kullanılan araçlar, karşılaşılan zorluklar ve çözümler ele alınmaktadır.

## **React Native ve Expo ile Çapraz Platform Geliştirme Süreci**

**Teknoloji Seçimi ve Gerekçeler:** Medora uygulamasının geliştirilmesinde React Native ve Expo teknolojilerinin tercih edilmesinin temel nedenleri şunlardır:

- **Çapraz Platform Uyumluluğu:** React Native, tek bir kod tabanıyla hem iOS hem de Android platformları için uygulama geliştirme imkanı sunar. Bu, geliştirme maliyetlerini ve süresini önemli ölçüde azaltır.
- **Hızlı Geliştirme:** Expo, React Native geliştirme sürecini kolaylaştıran araçlar ve hizmetler sunar. Hazır bileşenler, hızlı önizleme ve kolay dağıtım imkanları sayesinde geliştirme süreci hızlanır.
- **Geniş Topluluk ve Ekosistem:** React Native ve Expo, geniş ve aktif bir topluluğa sahiptir. Bu, karşılaşılan sorunlara çözüm bulmayı ve yeni kütüphanelerden yararlanmayı kolaylaştırır.

**Geliştirme Ortamının Kurulumu (Expo):** Expo CLI kullanılarak geliştirme ortamı kolayca kurulmuştur. Node.js ve npm (veya yarn) kurulumunun ardından, npm install -g expo-cli komutu ile Expo CLI global olarak yüklenmiştir. Proje oluşturma ve çalıştırma işlemleri Expo CLI komutları ile basitçe gerçekleştirilmiştir.

**Expo Router ile Navigasyonun Uygulanması:** Expo Router, uygulamanın navigasyon yapısını tanımlamak için kullanılmıştır. app dizinindeki dosya ve klasör yapısı, rotaları otomatik olarak oluşturur. Bu, geleneksel navigasyon kütüphanelerine kıyasla daha basit

ve sezgisel bir yaklaşım sunar. (auth) ve (tabs) gibi layout rotaları, farklı navigasyon yapılarının (kimlik doğrulama akışı ve ana uygulama sekmeleri) düzenli bir şekilde yönetilmesini sağlar.

**React Native Bileşenleri ve Özel Bileşenler:** Uygulama arayüzü, React Native'in temel bileşenleri (View, Text, TextInput, Switch vb.) ve components klasöründe oluşturulan özel bileşenler (Button, Input, Typo vb.) kullanılarak geliştirilmiştir. Özel bileşenler, uygulamanın tasarım dilini ve tekrar kullanılabilirliği artırmıştır. Örneğin, Typo bileşeni, metin stillerini merkezi olarak yönetmeyi ve tutarlılığı sağlamayı kolaylaştırır.

**Expo Özelliklerinin Kullanımı:** Expo, geliştirme sürecini kolaylaştıran birçok özellik sunmaktadır.

- **Expo Go:** Geliştirme sırasında uygulamayı gerçek cihazlarda veya emülatörlerde anında önizlemek için Expo Go uygulaması kullanılmıştır. Bu, hızlı iterasyon ve anında geri bildirim imkanı sağlar.
- **Expo Build:** Uygulamanın mağazalara dağıtımını için Expo Build hizmeti kullanılabilir. Bu, build süreçlerini otomatikleştirir ve dağıtımını kolaylaştırır.

**Stil Yönetimi ve Responsive Tasarım:** Uygulama stilleri, React Native'in StyleSheet API'si kullanılarak tanımlanmıştır. Renkler, boşluklar ve diğer tasarım değerleri constants/theme.ts dosyasında merkezi olarak tanımlanarak stil tutarlılığı sağlanmıştır. utils/styling.ts dosyasındaki fonksiyonlar ile farklı ekran boyutlarına uyumlu responsive tasarım uygulanmıştır.

**Test ve Hata Ayıklama:** Geliştirme sürecinde, Expo Go üzerinde anlık önizleme ve React Native Debugger gibi araçlar kullanılarak test ve hata ayıklama işlemleri yapılmıştır. Expo'nun sağladığı uzaktan hata ayıklama imkanı, geliştirme sürecini kolaylaştırmıştır.

**Karşılaşılan Zorluklar ve Çözümler (Genel):** Çapraz platform geliştirme sürecinde platform farklılıkları, kütüphane uyumsuzlukları gibi zorluklarla karşılaşılabilir. Ancak Expo ve React Native'in geniş topluluğu ve kapsamlı dokümantasyonu sayesinde bu tür sorunlara genellikle hızlı çözümler bulunabilir. Proje özelinde karşılaşılan zorluklar ve çözümleri proje kodları üzerinden daha detaylı incelenebilir.

#### 4.3. Firebase Entegrasyonu ve Veritabanı Tasarımı

Bu bölüm, Medora uygulamasında Firebase servislerinin nasıl entegre edildiğini ve veritabanı tasarımını açıklamaktadır. Kullanılan Firebase servisleri, veritabanı şeması, güvenlik kuralları ve veri yönetimi detayları ele alınmaktadır.

#### Firestore ile Backend Entegrasyonu ve Firestore Veritabanı Kullanımı

**Firestore Servislerinin Seçimi ve Gerekçeleri:** Medora uygulamasında backend ihtiyaçlarını karşılamak için Firestore servisleri tercih edilmiştir. Özellikle Firestore Authentication ve Firestore servisleri kullanılmıştır.

- **Firestore Authentication:** Kullanıcı kimlik doğrulama işlemlerini (kayıt, giriş, çıkış) güvenli ve kolay bir şekilde yönetmek için Firestore Authentication kullanılmıştır. E-posta/şifre tabanlı kimlik doğrulama yöntemi uygulanmıştır.
- **Firestore:** Uygulama verilerini (kullanıcı bilgileri, gelecekte eklenecek sağlık verileri vb.) depolamak ve yönetmek için Firestore NoSQL veritabanı kullanılmıştır. Firestore, gerçek zamanlı veri senkronizasyonu, ölçeklenebilirlik ve esneklik gibi avantajlar sunar.

**Firestore Projesi Kurulumu ve Entegrasyonu:** Firestore projesi Firestore konsolu üzerinden oluşturulmuş ve uygulama için gerekli yapılandırma ayarları (API anahtarı, proje ID vb.) alınmıştır. Bu yapılandırma bilgileri config/firestore.ts dosyasına eklenerek Firestore SDK'sı projeye entegre edilmiştir. Firestore SDK'sı, firestore/app, firestore/auth ve firestore/firestore paketleri aracılığıyla projeye dahil edilmiştir.

**Firestore Authentication Entegrasyonu:** Kullanıcı kimlik doğrulama işlemleri contexts/authContext.tsx dosyasında useAuth hook'u aracılığıyla yönetilmektedir.

- **Kullanıcı Kayıt (Register):** register fonksiyonu, createUserWithEmailAndPassword metodu ile yeni kullanıcı hesabı oluşturur ve Firestore'da patients koleksiyonuna kullanıcı bilgilerini kaydeder.

Şekil 4. Medora kayıt sayfası



- **Kullanıcı Giriş (Login):** login fonksiyonu, signInWithEmailAndPassword metodu ile mevcut kullanıcıların kimlik doğrulamasını yapar.



Şekil 5. Medora giriş sayfası

- **Kullanıcı Çıkış (Logout):** profile.tsx içinde signOut fonksiyonu ile kullanıcı oturumu sonlandırılır.
- **Oturum Yönetimi:** onAuthStateChanged fonksiyonu ile kullanıcı oturum durumu sürekli olarak izlenir. Kullanıcı giriş yaptığında uygulama ana ekranına (tabs), çıkış yaptığında ise karşılama ekranına yönlendirilir. Kullanıcı bilgileri (user state) AuthContext içinde saklanır ve uygulama genelinde erişilebilir hale getirilir.

**Firestore Veritabanı Tasarımı:** Uygulamada Firestore NoSQL veritabanı kullanılmıştır. Şu anda sadece patients koleksiyonu bulunmaktadır.

**Veri Erişim ve Yönetimi:** React Native uygulamasından Firestore veritabanına veri erişimi ve yönetimi Firebase SDK'sı aracılığıyla yapılmaktadır.

**Güvenlik Kuralları (Firebase Security Rules):** Firebase Security Rules, Firestore veritabanına erişimi kontrol etmek ve veri güvenliğini sağlamak için kullanılır. Proje kodlarında güvenlik kuralları doğrudan belirtilmemiş olsa da, Firebase konsolu üzerinden Firestore Security Rules yapılandırılarak veri erişim yetkilendirmesi ve veri doğrulama kuralları tanımlanmalıdır. Örneğin, patients koleksiyonuna sadece yetkili kullanıcıların erişebilmesi ve veri okuma/yazma işlemlerinin belirli koşullara bağlı olması sağlanabilir. Güvenlik kuralları, uygulamanın hassas sağlık verilerini işlemesi nedeniyle kritik öneme sahiptir.

## 5. YAPILAN VE YAPILACAK ÇALIŞMALAR

### 5.1. Şu Ana Kadar Yapılan Çalışmalar

Bu proje kapsamında, mobil cihazlar üzerinden hasta vücut verilerinin toplanmasını, bu verilerin yapay zekâ algoritmalarıyla analiz edilerek anlamlı bilgiler hâline getirilmesini ve ilgili sağlık personeli ile güvenli biçimde paylaşılmasını mümkün kılacak bütünleşik bir sistemin önemli bileşenleri geliştirilmiştir. İkinci ara rapor döneminde, projenin üç ana bileşeni olan mobil uygulama, veritabanı altyapısı ve yapay zekâ modelleme süreci önemli ölçüde ilerlemiştir.

#### 5.1.1 Mobil Uygulama Altyapısı ve Temel Arayüzler

Projenin mobil uygulama bileşeni, React Native ve Expo teknolojileri kullanılarak kapsamlı bir şekilde geliştirilmiştir. İlk ara rapordan bu yana, uygulamanın işlevselliği önemli ölçüde genişletilmiştir

#### Tamamlanan Ekranlar ve Özellikler:

- **Kimlik Doğrulama Sistemi:** Kayıt, giriş, hoş geldiniz ekranları ve onboarding modal'ı tamamlanmıştır.
- **Ana Dashboard (Kardiyovasküler Risk Tahmini):** Kullanıcıların yaş, boy, kilo, kan basıncı, kolesterol, glikoz seviyesi, sigara/alkol kullanımı ve fiziksel aktivite bilgilerini girerek KVVH risk tahminini alabilecekleri interaktif bir form geliştirilmiştir.

### Kardiyovasküler Risk Tahmini

Sağlık verilerinizi girerek risk seviyenizi öğrenebilirsiniz.

Yaş

Boy (cm)

Kilo (kg)

Üst Tansiyon (Sistolik) Alt Tansiyon (Diastolik)

Örnek: 120 Örnek: 80

Sigara Kullanıyorum: ☐

Alkol Kullanıyorum: ☐

Fiziksel Olarak Aktifim: ☐

Home Heart Line User Share Calendar

- Sağlık Verisi Yönetimi Ekranı:** Google Fit entegrasyonu ile kalp atış hızı, kan basıncı, kilo, adım sayısı, uyku, su tüketimi ve kalori verilerinin otomatik olarak çekilmesi ve analiz edilmesi sağlanmıştır.
- Sağlık Metrikleri Detay Ekranı:** Kalp atış hızı, kan basıncı ve kilo verilerinin detaylı grafik görünümü, trend analizleri ve anomali tespiti özellikleri eklenmiştir.
- Randevu Yönetimi:** Doktor randevularının listelenmesi, yeniden planlanması ve iptal edilmesi için kullanıcı arayüzü geliştirilmiştir.

### Randevular

Yaklaşan Tamamlanan

**Ahmet Yılmaz** Yaklaşan  
Kontrol muayenesi  
24 Haziran 2023 10:00  
Yeniden Planla İptal Et

**Zeynep Taş** Yaklaşan  
İlk muayene  
24 Haziran 2023 11:30  
Yeniden Planla İptal Et

**Mehmet Demir** Yaklaşan  
Sonuç değerlendirme  
24 Haziran 2023 14:00  
Yeniden Planla İptal Et

Home Heart Line User Share Calendar

- **Mesajlaşma Sistemi:** Doktor-hasta iletişimi için mesajlaşma arayüzü ve konuşma listesi oluşturulmuştur.
- **Hasta Listesi (Doktor Görünümü):** Doktorların hastalarını görüntüleyebileceği liste ekranı geliştirilmiştir.
- **Profil ve Güvenlik Ayarları:** Kullanıcı profil yönetimi, güvenlik ayarları ve hesap yönetimi özellikleri eklenmiştir.



### Teknik Altyapı Geliştirmeleri:

- Expo Router ile dosya tabanlı navigasyon sistemi kurulmuştur.

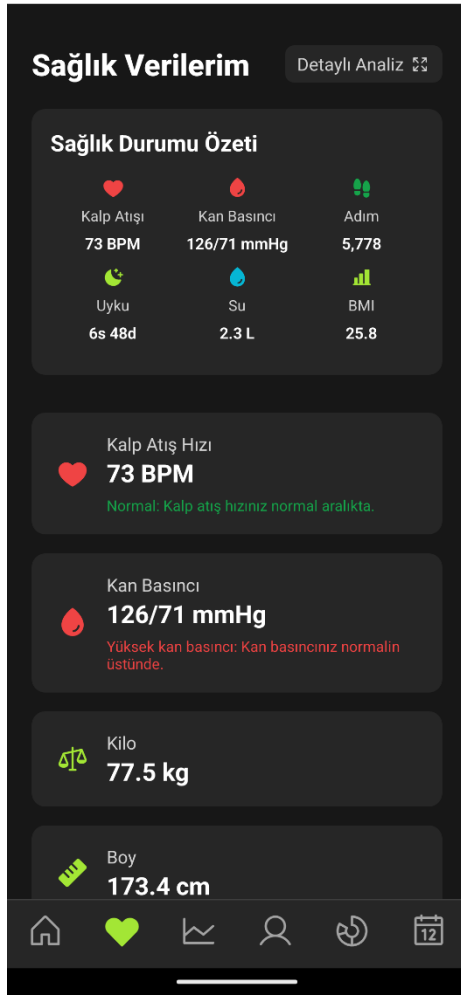
- Context API ile global durum yönetimi uygulanmıştır.
- Responsive tasarım için ölçekleme fonksiyonları geliştirilmiştir.
- Tekrar kullanılabilir bileşen kütüphanesi oluşturulmuştur (Button, Input, Typo, ScreenWrapper, vb.).
- Özel tab navigasyon bileşeni (CustomTabs) geliştirilmiştir.

### **5.1.2 Sağlık Verisi Analizi ve İşleme Sistemi**

Uygulamaya kapsamlı bir sağlık verisi analiz sistemi entegre edilmiştir:

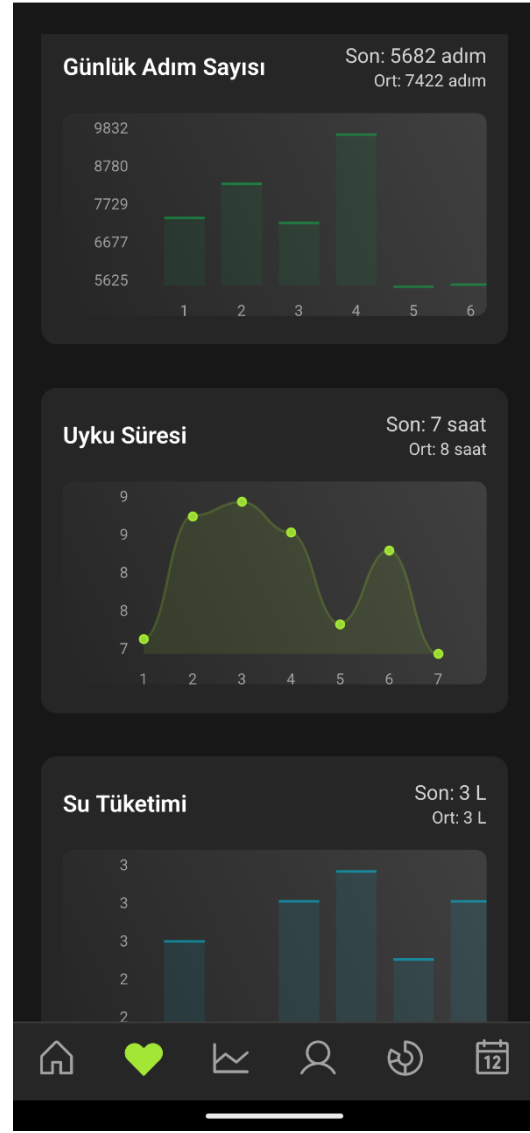
#### **Analiz Fonksiyonları:**

- Kalp Atış Hızı Analizi: Taşikardi ( $>100$  BPM) ve bradikardi ( $<60$  BPM) tespiti
- Kan Basıncı Analizi: AHA sınıflandırmasına göre hipertansiyon aşamalarının belirlenmesi
- BMI Hesaplama ve Sınıflandırma: Vücut kitle indeksi analizi ve kategorilendirme
- Anomali Tespiti: Z-Score tabanlı istatistiksel anomali tespit algoritması
- Trend Analizi: Lineer regresyon ile veri trendlerinin belirlenmesi
- Adım, Uyku, Su Tüketimi ve Kalori Analizleri: Günlük aktivite verilerinin değerlendirilmesi



## Veri Entegrasyonu:

- Google Fit API entegrasyonu (geliştirme aşamasında mock data kullanımı)
- Gerçek zamanlı veri senkronizasyonu altyapısı
- Veri görselleştirme bileşenleri (HealthChart, HealthTrendsCard)



### 5.1.3 Veritabanı Yapısı ve Firebase Entegrasyonu

Firestore veritabanı sistemi genişletilmiştir:

- Kullanıcı Profilleri: Hasta ve doktor rollerini destekleyen kullanıcı sistemi
- Sağlık Profili Yönetimi: Yaş, kilo, boy, cinsiyet, kan grubu, alerji ve kronik hastalık bilgileri
- Güvenlik Kuralları: Firebase Security Rules ile veri erişim kontrolü
- Gerçek Zamanlı Veri Senkronizasyonu: Anlık veri güncellemeleri

#### **5.1.4 Yapay Zekâ Modelleme ve Backend Entegrasyonu**

Yapay zekâ bileşenleri geliştirilmiş ve entegrasyon çalışmaları başlatılmıştır:

- EKG Sinyal Analizi: PTB Diagnostic ECG veri seti ile Convolutional Autoencoder modeli eğitilmiştir.
- Kardiyovasküler Risk Tahmini: 70.000 hasta verisi ile çoklu sınıflandırma modelleri geliştirilmiştir.
- Flask Backend API: OpenAI GPT entegrasyonu ile sağlık verisi analizi için backend servis oluşturulmuştur.
- TensorFlow.js Entegrasyonu: Mobil uygulama içinde AI model çalıştırma altyapısı hazırlanmıştır.

#### **5.2. Bundan Sonra Yapılacak Çalışmalar**

İkinci ara rapor döneminde önemli ilerleme kaydedilen projenin, final aşamasında aşağıdaki çalışmalar tamamlanacaktır:

##### **5.2.1. Yapay Zekâ Modellerinin Tam Entegrasyonu**

- EKG anomali tespit modelinin mobil uygulamaya entegrasyonu
- KVH risk tahmin modelinin gerçek zamanlı çalışır hale getirilmesi
- Model sonuçlarının kullanıcı dostu görselleştirmesi
- Tahmin güvenilirlik skorlarının gösterilmesi



### **5.2.2. Gerçek Veri Entegrasyonu**

- Google Fit API'nin production ortamında aktifleştirilmesi
- Apple HealthKit entegrasyonu (Apple Developer hesabı gerektirdiğinden gerçekleştirilmeyebilir)
- Giyilebilir cihaz bağlantıları (akıllı saat, fitness tracker)
- Manuel veri girişi formlarının tamamlanması

### **5.2.3. Doktor-Hasta İletişim Sisteminin Geliştirilmesi**

- Gerçek zamanlı mesajlaşma sistemi
- Veri paylaşım izinleri ve güvenlik protokolleri
- Randevu planlama ve takip sistemi
- Acil durum bildirimleri

### **5.2.4. Test ve Optimizasyon**

- Kapsamlı kullanıcı testleri
- Performans optimizasyonu
- Hata ayıklama ve stabilite iyileştirmeleri
- Kullanıcı geri bildirimlerinin değerlendirilmesi

## KAYNAKLAR

- [1] M. Değerli, “Mobil Sağlık Uygulamalarına Genel Bir Bakış ve Özgün Bir Mobil Sağlık Uygulaması Geliştirilmesi,” Enformatik Enstitüsü, Orta Doğu Teknik Üniversitesi, 2021.
- [2] C. Türkdoğan Görgün, “Mobil Sağlık Teknolojileri ve Teknoloji Kabul Teorileri Üzerine Bir İnceleme,” Sağlık Hizmetlerinde Kuram ve Uygulama Dergisi, vol. 2, no. 1, pp. 37-49, Eyl. 2021.
- [3] Ş. Güzel, H. Akman Dömbekci, and F. Eren, “Yapay Zekânın Sağlık Alanında Kullanımı: Nitel Bir Araştırma,” CBU-SBED, vol. 9, no. 4, pp. 509–519, Dec. 2022. DOI: 10.34087/cbusbed.1140122.
- [4] Expo Documentation. (2025). Introduction to Expo. Retrieved from <https://docs.expo.dev>
- [5] Statista. (2025). "Cross-platform mobile frameworks used by software developers worldwide from 2019 to 2023" Retrieved from <https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/>
- [6] Google Cloud. (2025). HIPAA Compliance on Google Cloud Platform. Retrieved from <https://cloud.google.com>
- [7] Google Firebase Documentation. (2025). Firebase Security and Privacy. Retrieved from <https://firebase.google.com>
- [8] <https://www.kaggle.com/code/redpen12/anomaly-detection-with-cnn-autoencoders/input>