



FITKING

- Proje Adı : FİTKİNG
- Ekip Üyeleri
 - Furkan Akkamış
 - Muhammed Tarık Mengüç
 - Mehmet Emin Yılmaz
 - Zeliha Sena Güllü

Proje Özeti

İçerik :

- **FitKing**, kullanıcıların beslenme alışkanlıklarını yönetmelerine, fitness hedeflerine ulaşmalarına ve genel sağlık durumlarını takip etmelerine olanak tanıyan, yapay zeka destekli yenilikçi bir mobil sağlık platformudur.
- **Amaç:** Kullanıcıların manuel veri girişiyle uğraşmadan, fotoğraf tanıma ve akıllı analizlerle **sağlıklı yaşam hedeflerine ulaşmalarını** sağlamak.
- Uygulama; beslenme, spor, su tüketimi, ilaç hatırlatma, rozet kazanma gibi modüllerle kullanıcıyı motive eder.
- **Problem:** Beslenme takibindeki manuel zorluklar (kalori/makro sayma) ve dağınık sağlık verilerinin yönetimi.
- **Çözüm:** Yapay Zeka kullanarak bu zorlukları ortadan kaldıran, hepsi bir arada mobil sağlık platformu.

Temel Çözüm ve Fonksiyonlar :

- **Yapay Zeka Destekli Görüntü Tanıma:** Kullanıcılar yemek fotoğrafı çekerek ilgili öğünün otomatik olarak tanınmasını, kalori ve makro değerlerinin hesaplanmasını sağlar.
- **Kişiselleştirilmiş Planlama:** Kullanıcı hedeflerine (kilo alma/koruma/verme) yönelik günlük kalori ve makro besin önerileri oluşturulur.
- **Sağlık Takibi:** Girilen kan tahlili değerleri analiz edilerek, kural tabanlı potansiyel risklere dair bilgilendirme kartları sunulur (her zaman doktor uyarısı içerir).
- **Ek Modüller:** Spor programları, sağlıklı tarifler, alışkanlık takibi, motivasyonel rozet sistemi ve hatırlatıcılar (ilaç/su/spor) platformda yer alır.
- **Yapay Zekâ ile Yemek Tanıma:**
Fotoğraf çek → AI tanır → Kalori & makro hesaplanır.
- **Hedef Yönetimi:**
Kilo alma, koruma veya verme hedefi belirle → Günlük öneriler otomatik oluşturulur.
- **Kan Tahlili Analizi:**
Kullanıcı tahlil sonuçlarını girer → Sistem kurallara göre bilgilendirici kartlar üretir.
- **Spor ve Sağlıklı Yaşam Modülleri:**
Spor planları, tarif önerileri, su/ilaç hatırlatıcılar, rozet sistemi.

Teknoloji ve Mimari

- **Mobil:** React Native + Expo

Tek Kod Tabanı: iOS ve Android için tek bir kodla geliştirme; zaman ve kaynak verimliliği.

Expo: Kamera, bildirimler gibi yerel özelliklere erişimi standartlaştırarak geliştirme hızını artırır.

- **Backend:** Node.js + NestJS

Mimari: Modüler ve ölçeklenebilir arka uç sistemi.

TypeScript: Tip güvenliği sağlar; kod okunabilirliğini ve sürdürülebilirliğini artırır.

- **Firebase Servisleri:**

Avantaj: Güvenli, ölçeklenebilir ve yönetimi kolay hazır çözümler sunarak altyapı maliyetlerini ve geliştirme eforunu minimize eder.

Kullanım Alanları: Kullanıcı kimlik doğrulama (Auth), dosya depolama (Storage), anlık bildirim (FCM).

Teknoloji ve Mimari

- **Veri Tabanı: Hibrit mimari**
- Strateji: Farklı veri tiplerinin gereksinimlerini karşılamak için Hibrit Model (İlişkisel + NoSQL).
- **PostgreSQL (İlişkisel):Kullanım:** Kullanıcı profilleri, hedefler, spor programları, tarifler gibi yapılandırılmış ve ilişkisel bütünlüğün kritik olduğu veriler.
Neden: ACID uyumluluğu ile veri tutarlılığını garanti eder; karmaşık sorgulama (JOIN) ve raporlama için idealdir.
- **MongoDB (NoSQL):Kullanım:** Yemek fotoğraflarına ait ham log kayıtları (OCR metinleri, güven skorları) ve kan tahlili ham kayıtları gibi şeması değişken veriler.
Neden: Esnek şema yapısı, yapay zeka çıktısı gibi sürekli evrilebilen veri formatlarını kolayca saklar; yoğun veri akışı ve loglama için yüksek yazma hızı ve yatay ölçeklenebilirlik sağlar.
- **Özet:** PG'nin kararlılığı ve tutarlılığı ile Mongo'nun esnekliği ve hızı birleştirildi.

Neden Bu Teknolojiler?

- React Native → Zaman ve maliyet tasarrufu (tek kod tabanı)
- NestJS → Modüler, ölçeklenebilir, temiz mimari
- Firebase → Güvenli, düşük bakım maliyetli servis altyapısı
- PostgreSQL → Tutarlılık ve güvenilirlik
- MongoDB → Hız ve esneklik
- Docker → Takım üyeleri arasında uyumlu geliştirme ortamı

İşlevsel Gereksinimler

- Kullanıcı sisteme kayıt olabilir, giriş yapabilir ve profilini güncelleyebilir.
- Kullanıcı, öğün fotoğrafı yüklediğinde sistem bu görüntüyü analiz ederek yemeği tanır ve besin değerlerini hesaplar.
- Kullanıcı kilo alma, kilo verme veya mevcut kilosunu koruma hedefi belirleyebilir.
- Sistem, hedeflere uygun günlük bilgilendirmeler ve öneriler sunar.
- Kullanıcı geçmiş öğün kayıtlarını görüntüleyebilir.
- Sistem, kullanıcılara hatırlatmalar ve bildirimler gönderebilir.
- Kullanıcı kötü alışkanlık bırakma takibi yapabilir (örneğin sigara, fast food, şekerli içecekler).
- Sistem sağlıklı tarif önerileri sunabilir.
- Sistem kullanıcıya uygun spor programları gösterebilir.
- Kullanıcı ilaçlarını ve kullanım saatlerini takvim üzerinden sisteme girebilir.
- Sistem belirlenen saatlerde kullanıcılara ilaç hatırlatma bildirimi gönderebilir.
- Yönetici kullanıcı verilerini ve sistem işleyişini yönetebilir.

İşlevsel Olmayan Gereksinimler

- **Performans:** Görüntü tanıma işlemi maksimum 5 saniye içinde tamamlanmalıdır.
- **Güvenlik:** Tüm veriler HTTPS üzerinden iletilmeli ve şifreli saklanmalıdır.
- **Kullanılabilirlik:** Uygulama sade, sezgisel ve kullanıcı dostu bir arayüze sahip olmalıdır.
- **Taşınabilirlik:** Uygulama iOS ve Android cihazlarda sorunsuz çalışmalıdır.
- **Bakım:** Kod yapısı modüler olmalı, hata ayıklama ve güncelleme süreçleri kolay olmalıdır.
- **Erişilebilirlik:** Uygulama ekran okuyucularla uyumlu olacak şekilde tasarlanmalıdır.
- **Genişletilebilirlik:** Yeni özelliklerin sisteme kolayca eklenebilmesi sağlanmalıdır.
- **Bildirim Güvenirliği:** Hatırlatma bildirimleri yüksek doğrulukla çalışmalıdır.

Sistem Mimarisi

- FitKing projesi, modüler ve ölçeklenebilir bir yapı üzerine kurulmuştur. Sistem mimarisi dört ana katmandan oluşur:

1. Kullanıcı Katmanı (Mobil Uygulama)

1. Kullanıcılar React Native tabanlı mobil uygulama aracılığıyla sisteme erişir.
2. Uygulama üzerinden yemek fotoğrafı çekme, hedef belirleme, veri görüntüleme ve bildirim alma işlemleri yapılır.

2. Sunucu Katmanı (Backend API - NestJS)

1. Node.js tabanlı NestJS framework'ü kullanılarak geliştirilmiştir.
2. Mobil uygulamadan gelen istekleri alır, doğrular, veritabanlarına yönlendirir.
3. API uç noktaları REST mimarisine uygun biçimde tasarlanmıştır.
4. Docker üzerinde çalışarak geliştirici ortamlarında tutarlılık sağlar.

Sistem Mimarisi

1. Veri Katmanı (Hibrit Veri Tabanı Yapısı)

1. PostgreSQL:

İlişkisel verilerin tutulduğu kısım (kullanıcılar, planlar, tarifler, hatırlatıcılar).
ACID uyumlu, güvenli ve tutarlı yapı sağlar.

2. MongoDB:

Esnek şemalı verilerin (fotoğraf analiz logları, OCR çıktıları, tahlil verileri) depolandığı kısım.
Yüksek yazma hızı ve esneklik sağlar.

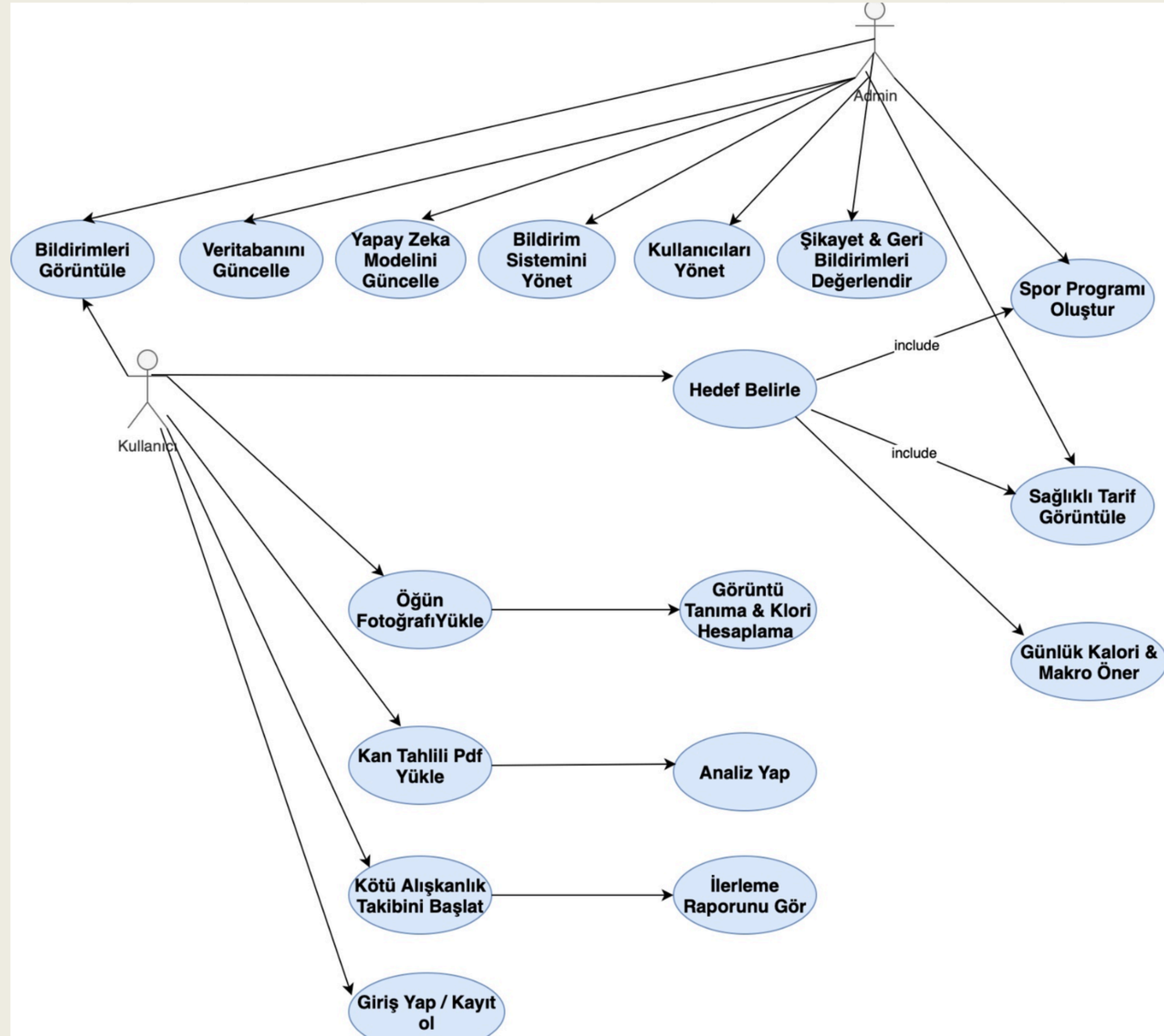
2. Harici Servis Katmanı (Firebase Entegrasyonu)

1. **Firebase Auth:** Kullanıcı giriş/çıkış işlemleri için kimlik doğrulama.

2. **Firebase Storage:** Kullanıcı fotoğrafları ve medya dosyalarının depolanması.

3. **Firebase Cloud Messaging (FCM):** Bildirimlerin gönderilmesi (örneğin “su içmeyi unutma” uyarıları).

Use Case



Ekip Bilgisi & Gelecek Adımlar

- **Ekip:**

Furkan Akkamış: Scrum Master, Veritabanı , Cloud

Muhammed Tarık Mengüç: Backend , AI

Mehmet Emin Yılmaz : AI , Tasarım

Zeliha Sena Güllü: Mobil Developer , Tasarım

- **Gelecek Adımlar:**

Uygulama içi Diyetisyen - Danışan Erişimi

Sağlıklı yemek tarifleri

Kişiye özel spor programı