



# Glowmance: AI Skin Advisor

# CİLDİNİ TANI - DOĞRU ÜRÜNÜ SEÇ


Günümüzde birçok kişi hâlâ kendi cilt tipini tanımakta ve cilt özelliklerine uygun ürünler bulmakta zorlanıyor. Bu durum, bazen yanlış ürün kullanımıyla cildin daha da kötü hale gelmesine neden olabiliyor.

Nasıl ki parmak izlerimiz birbirinden farklıysa, her insanın cilt tipi de kendine özgüdür. Çünkü beslenme şeklimiz, yaşam tarzımız ve çevresel faktörler cildimizi doğrudan etkiler.

Biz de bu sorunlardan yola çıkarak,

**“Senin cildin, senin hikâyen — Cildini tanı ve doğru ürünü seç.”**

sloganı ile bu projeye başladık

Cildinizi tanımaya hazır mısınız? .

# TEMEL HEDEFİMİZ

Bu projenin temel hedefi, kullanıcının **cilt tipini uygulamaya yüklediği görsel** aracılığıyla analiz etmek ve **kişiye özel ürün önerileri** sunmaktır.

Bu sayede kullanıcı, **cildini tanıyarak** kendi cilt yapısına uygun ürünleri kolayca bulabilecek ve **doğru ürünlerle cildine parlaklık kazandıracaktır.**

Projemizin adı “**Glowmance**”, İngilizce’de “glow” (parlaklık) kelimesinden gelmektedir.

Yani, hedefimiz her kullanıcının kendi cilt hikâyesini keşfetmesi ve doğal parlaklığına kavuşmasıdır. 🌸

✨ Kişisel Cilt Analizi • Ürün Önerisi • Doğal Parlaklık

İsim	Ekip İçi Rol	Görev ve Sorumluluklar
Beyza Nur OZANALP	Scrum Master & Veritabanı Kısmı	-Proje planlama, görev takibi -PostgreSQL & MongoDB veritabanı tasarımı -Veri güvenliği ve entegrasyonu
Tuğba BARS	Mobil Arayüz(Frontend) Kısmı	-React Native ile mobil arayüz geliştirme -Fotoğraf yükleme ve analiz ekranları -UI/UX tasarımı -API entegrasyonu
Kumru ÇELİK	Backend Kısmı	- <a href="#">Node.js</a> + <a href="#">Express.js</a> ile API geliştirme -Yapay zeka API entegrasyonu -Kullanıcı kayıt/giriş, analiz ve öneri uç noktaları
Demhat GEZER	Yapay Zeka Kısmı	-Görsel tabanlı cilt analizi modeli -Görüntü ön işleme ve analiz ekranı -Ürün öneri algoritması geliştirme -Modelin mobil uygulama ile entegrasyonu

# USER STORIES

Bir **Kullanıcı** olarak, mevcut cilt durumumu anında görmek için **mobil kameramla fotoğraf çekip analiz başlatmak** istiyorum ki, zaman kaybetmeden cildimdeki sorunları tespit edebileyim.

Bir **Kullanıcı** olarak, AI'ın yaptığı **cilt analizi sonuçlarını ve skorlarını** net bir şekilde görmek istiyorum ki, cildimdeki sorunlu alanları (akne, kırıxıklık vb.) bilimsel verilere dayanarak anlayabileyim.

Bir **Kullanıcı** olarak, **analiz sonucuma ve geçmişime göre kişiselleştirilmiş ürünlerin listesini** görmek istiyorum ki, hangi tür ürünü (marka bağımsız) kullanmam gerektiği konusunda bilinçlenebileyim.

Bir **Kullanıcı** olarak, **geçmişteki cilt rahatsızlıklarımı veya kullandığım ilaçları** sisteme kaydetmek istiyorum ki, AI bana öneri yaparken bu bilgileri dikkate alarak **güvenli ve doğru** yönlendirme yapsın.

Bir **Kullanıcı** olarak, **zaman içinde yaptığım tüm cilt analizlerini** listeleyip karşılaştırmak istiyorum ki, cildimin günden güne iyileşip iyileşmediğini ve bakımı işe yarıyor mu takip edebileyim.

# USER STORIES

**Admin** olarak, **yeni bir cilt bakım ürününü, içeriği ve uygun cilt tipleriyle birlikte** sisteme eklemek istiyorum ki, AI Bir modelinin öneri yapabileceği ürün kataloğunu güncel tutabileyim.

Bir **Admin** olarak, kritik veritabanı tablolarını (örneğin ürün kategorilerini veya kullanıcı loglarını) **doğrudan yönetmek** istiyorum ki, sistemin veri bütünlüğünü ve güvenliğini sağlayabileyim.

# FONKSİYONEL GEREKSİNİMLER

- ➡ Kullanıcı, **fotoğraf çekebilir veya galeriden yükleyebilir**.
- ➡ Uygulama, **yapay zeka ile cilt analizini otomatik** olarak gerçekleştirmeli (yağ, nem, akne, kızarıklık, kırıışıklık vb.).
- ➡ Kullanıcı, **analiz sonuçlarını ve geçmiş analizlerini** görebilmeli.
- ➡ Sistem, **kişiselleştirilmiş ürün önerileri** sunmalı.
- ➡ Kullanıcı, **önceki cilt rahatsızlıklarını ve kullandığı ürün/ilaçları** girebilmeli.
- ➡ Kullanıcı, **önerilen ürünleri favorilere ekleyebilmeli** veya kaydedebilmeli.
- ➡ **Giriş / Kayıt sistemi** olmalı (JWT tabanlı kimlik doğrulama).
- ➡ Admin, **ürün veritabanını güncelleyebilmeli** (manuel, API veya scraping ile).


# FONKSİYONEL OLMAYAN GEREKSİNİMLER

- ➡ **Kullanıcı dostu ve sade arayüz:** Mobil cihazlarda kolay kullanılabilir, sezgisel tasarım.
- ➡ **Performans:** Görsel analiz işlemleri kısa sürede tamamlanmalı.
- ➡ **Veri güvenliği:** Kullanıcı fotoğrafları ve kişisel bilgiler şifreli biçimde saklanmalı (JWT, bcrypt, güvenli depolama).
- ➡ **Doğruluk:** Yapay zeka modeli, analiz sonuçlarında yüksek doğruluk oranına sahip olmalı.
- ➡ **Gizlilik:** Kullanıcı fotoğrafları yalnızca analiz amacıyla kullanılmalı, üçüncü kişilerle paylaşılmamalı.
- ➡ **Uygulama stabilitesi:** Mobil cihazlarda donma veya çökme yaşanmamalı.
- ➡ **Çok platform desteği:** iOS ve Android'de sorunsuz çalışmalı.




# UYGULAMA İŞLEYİŞİ

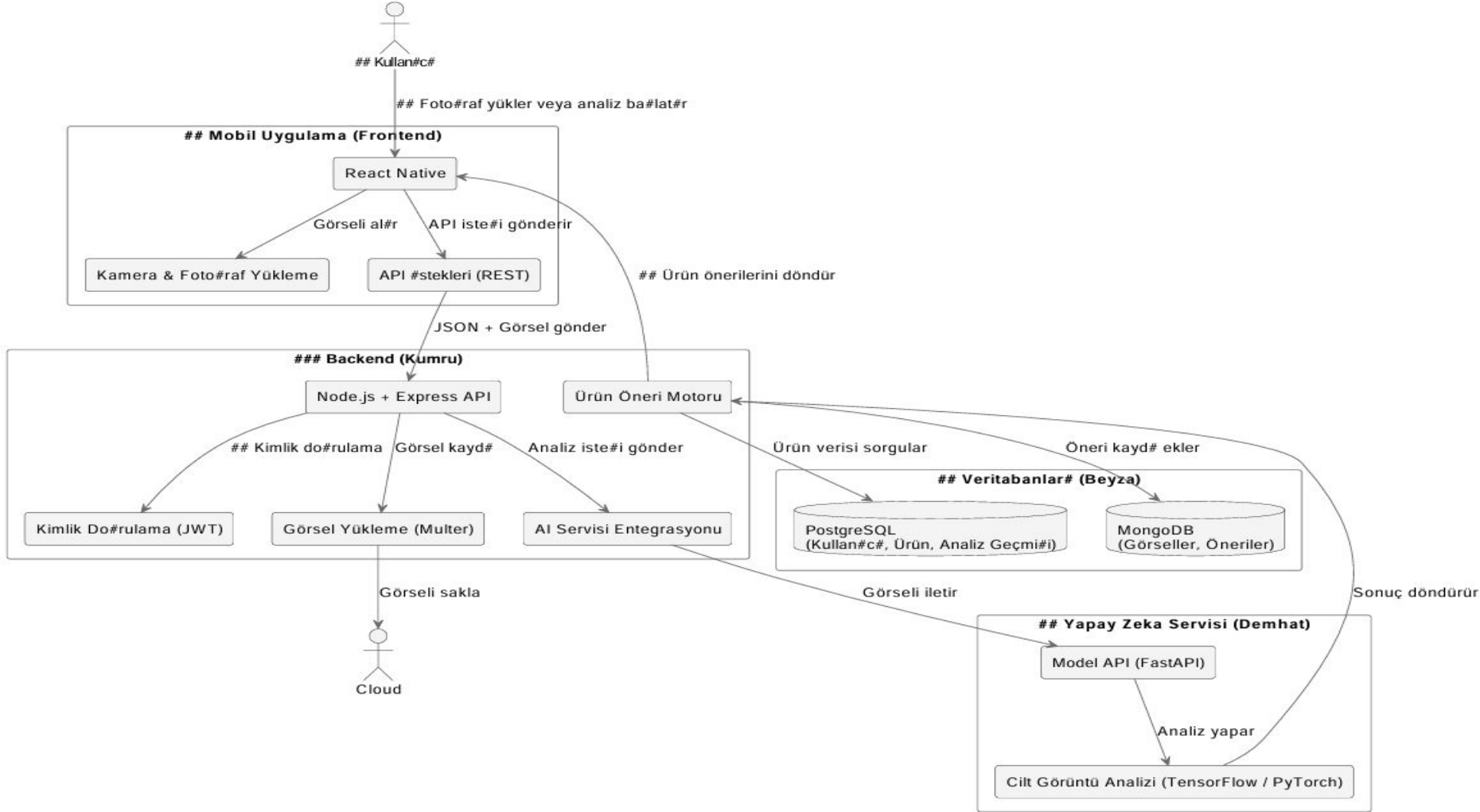
1. Kullanıcı uygulamayı açar.
2. Cilt fotoğrafını çeker veya galeriden yükler.
3. AI modeli fotoğrafı işler ve cilt durumunu analiz eder.
4. Analiz sonucu (örneğin: karma cilt, kuru bölgeler, akne eğilimi) döner.
5. Sistem veritabanından en uygun ürünleri filtreler.
6. Kullanıcıya öneri listesi gösterilir.
7. Kullanıcı ürünü kaydedebilir, favorileyebilir veya geçmişini görebilir.

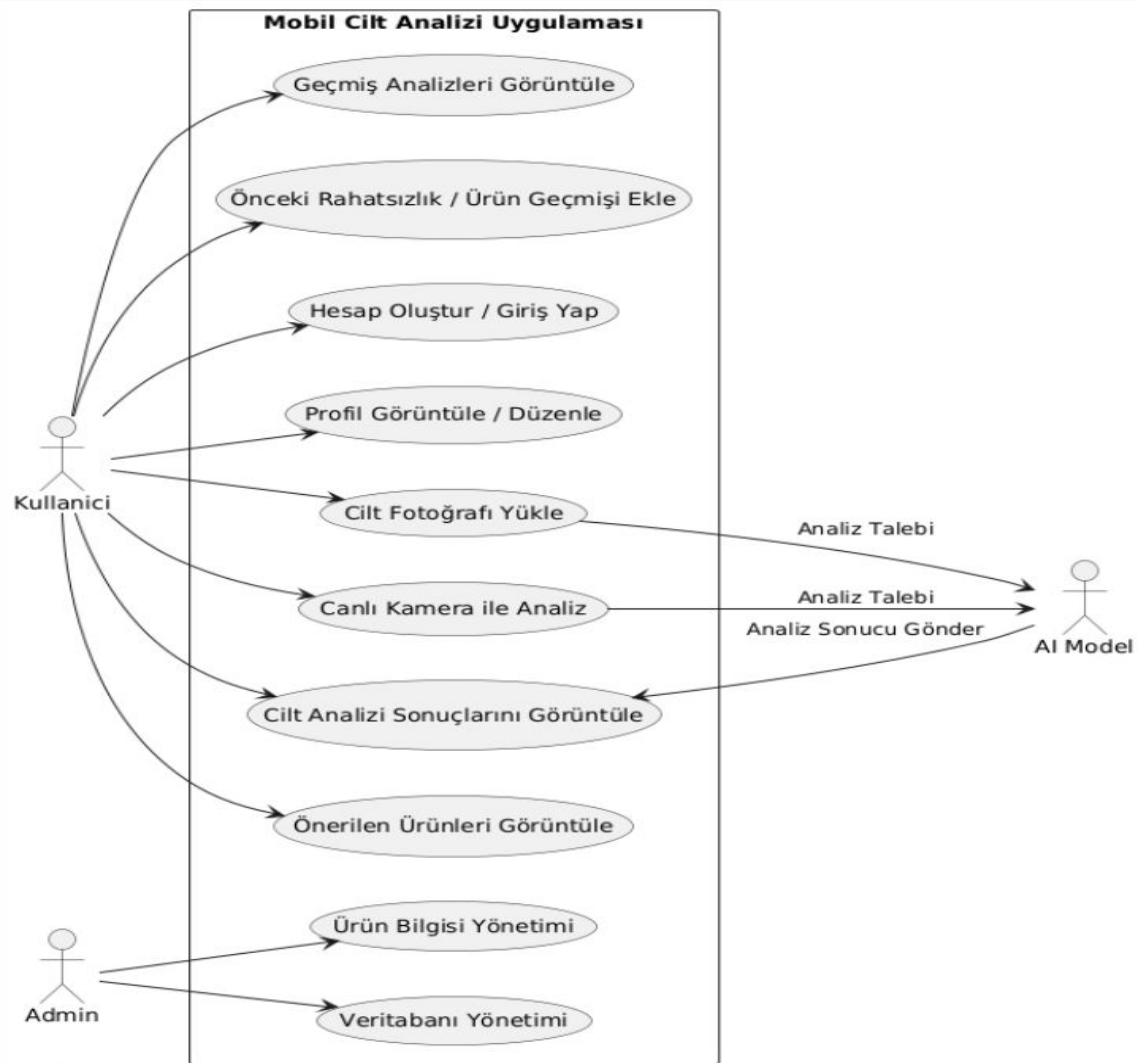
→  **Amaç:** Sürecin baştan sona kullanıcı dostu olması.

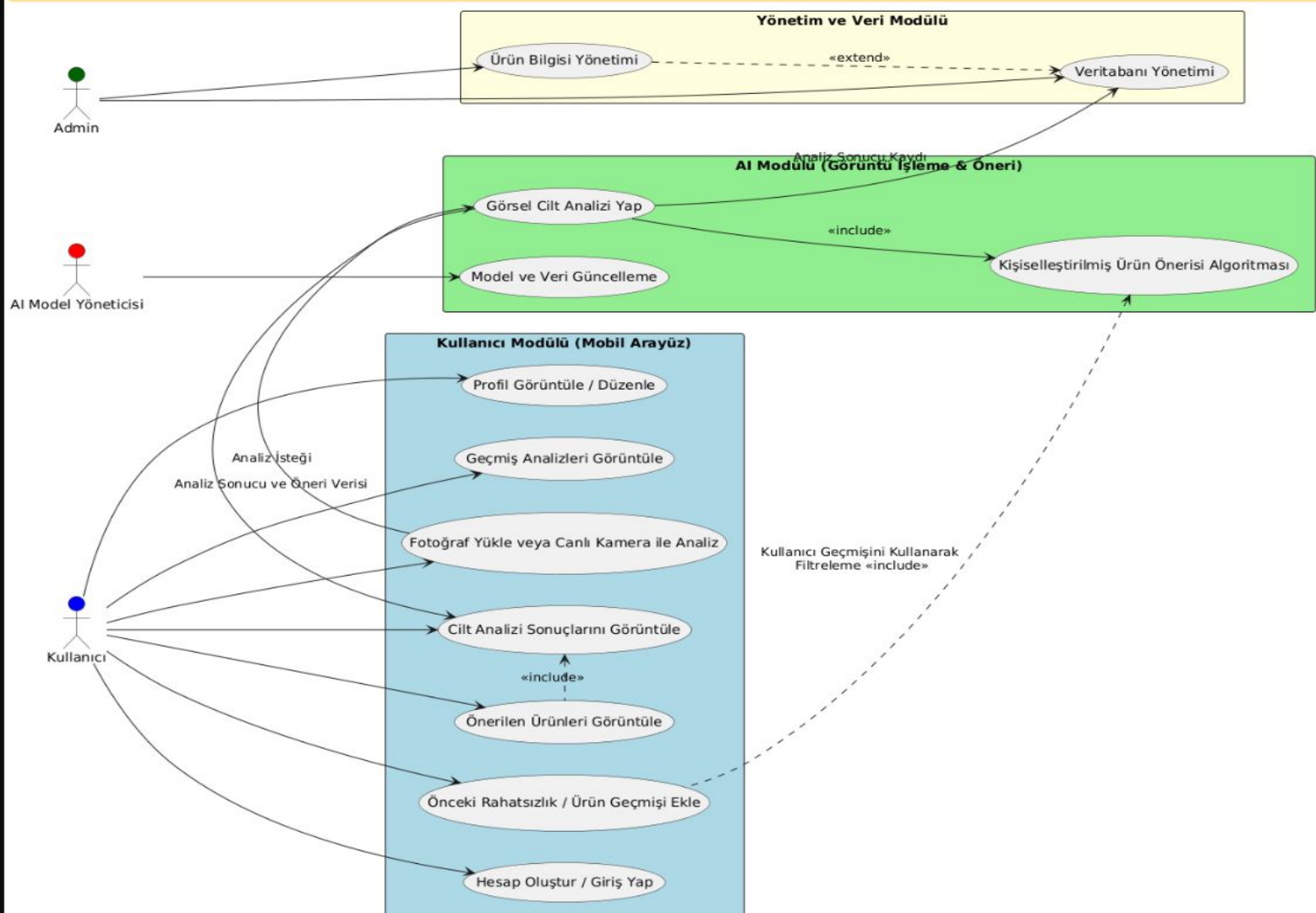
- Yapay zekâ modeli, kullanıcının yüklediği fotoğrafı analiz eder.
- Görüntü işleme teknikleriyle cilt tipi, akne, leke vb. problemleri tespit eder.
- Analiz sonuçlarına göre sistem, veritabanındaki uygun ürünleri filtreler.
- Kullanıcıya en uygun ürün önerileri gösterilir.

 **Amaç:** Kullanıcıyı dermatolog gerektirmeden, doğru ürünle buluşturmak.

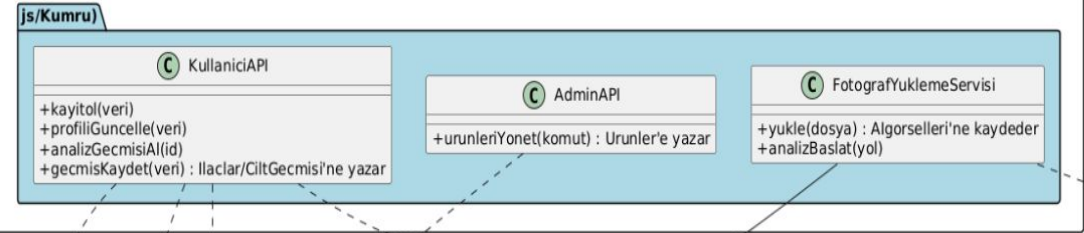
# ## Mobil Yapay Zeka Destekli Cilt Analizi ve Ürün Öneri Sistemi (Basit Dikey Sistem Mimarisi)



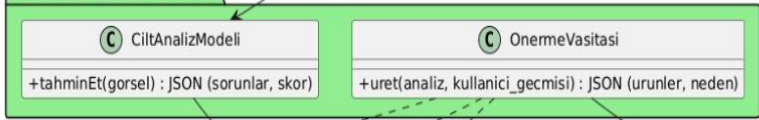




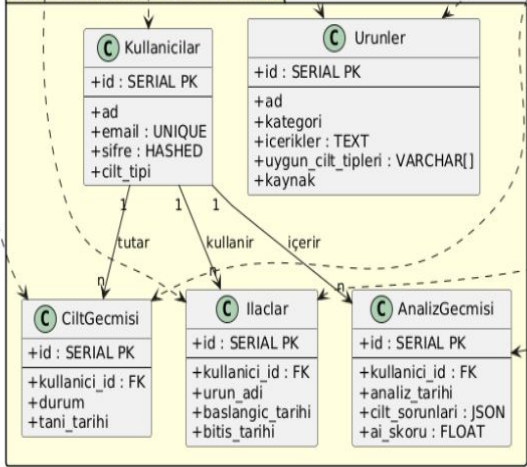
Arka Uç Servis (Node)



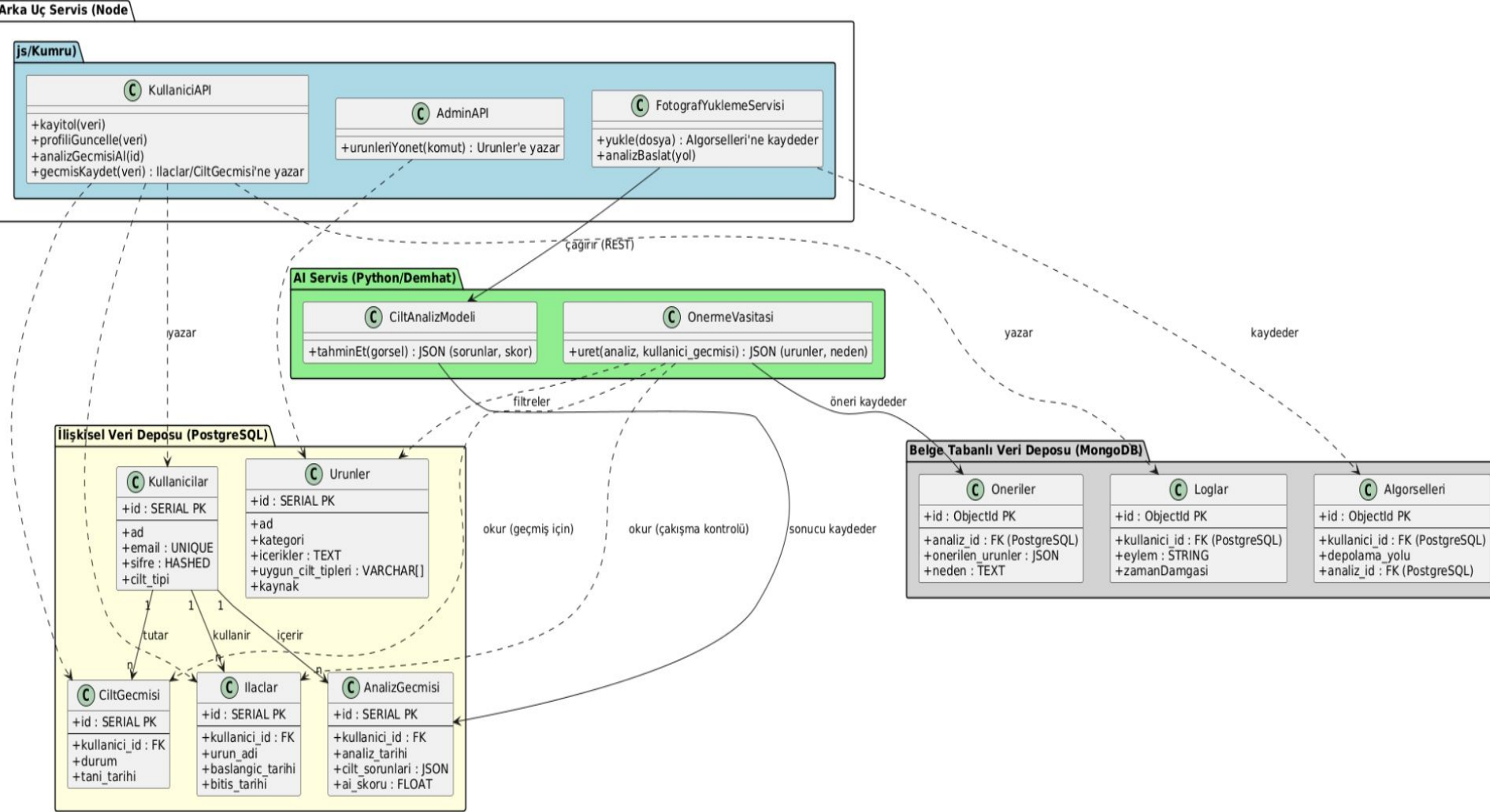
AI Servis (Python/Demhat)



İlişkisel Veri Deposu (PostgreSQL)



Belge Tabanlı Veri Deposu (MongoDB)



# TEKNOLOJİK GEREKSİNİMLER

- ➡ **Platform:** Mobil (iOS & Android)
- ➡ **Frontend:** React Native
- ➡ **Backend:** Node.js + Express.js
- ➡ **Veritabanı:** PostgreSQL (kullanıcı, ürün, analiz geçmişi) + MongoDB (AI verisi, görseller, loglar)
- ➡ **Yapay Zeka:** Python + TensorFlow,
- ➡ **Proje Yönetimi:** Jira, GitHub, Postman

- **React Native** : Mobil uygulama arayüzü (fotoğraf çekme, sonuç gösterme).
- **Node.js + Express**: Sunucu tarafı, analiz ve öneri isteğini yönetir.
- **Python (TensorFlow / PyTorch)**: Görsel analizi yapan AI modeli.
- **PostgreSQL**: Kullanıcı, ürün, analiz geçmiş bilgilerini saklar.
- **MongoDB**: Görsel veriler ve AI loglarını depolar.

🧩 **Amaç:** Uygulama parçalarının birbiriyle uyum içinde çalışmasını sağlamak.



# Neden Bu Teknolojiler Kullanılacak?

Projemizde hem mobil arayüzü hem de sunucu tarafını **JavaScript tabanlı teknolojilerle geliştirmeyi planlıyoruz**. **React Native**, kullanıcı arayüzünü oluşturmak ve uygulamanın **iOS ile Android cihazlarda sorunsuz çalışmasını sağlamak** için kullanılacak.

Bu teknoloji sayesinde **tek bir kod tabanıyla iki platforma da çıktı alınabilecek** ve kullanıcıya **native uygulama deneyimi** sunulabilecek.

**Node.js** ise **backend (sunucu tarafı)** geliştirmesinde kullanılacak.

**Express.js** çatısı ile birlikte kullanıcı girişi, ürün verilerinin yönetimi ve öneri sonuçlarının iletilmesi gibi işlemler burada yürütülecek.

React Native ve Node.js'in **aynı dili (JavaScript)** kullanması, proje geliştirme sürecinde **hız, bütünlük ve kolay entegrasyon** sağlamayı hedefliyor.

### ▲ PostgreSQL

Projemizde **ilişkisel (yapısal) verilerin yönetimi için PostgreSQL kullanmayı planlıyoruz.**

Kullanıcı bilgileri, ürün verileri ve geçmiş kayıtlar gibi **tablo yapısına uygun, düzenli veriler** bu veritabanında tutulacak.

PostgreSQL; **veri bütünlüğü, güvenlik, karmaşık sorgular ve performans** açısından güçlü bir yapıya sahip.

Bu sayede kullanıcıların **işlem geçmişleri, favorileri ve kayıtlı bilgileri** güvenli ve tutarlı bir şekilde saklanabilecek.

### ▲ MongoDB

Projemizde, **yapay zekâ analizlerinden gelen verilerin esnek biçimde saklanabilmesi için MongoDB kullanılacak.**

Cilt görsellerinden elde edilen **analiz sonuçları, cilt tipi özellikleri ve öneri sistemine ait veriler** genellikle JSON formatında olacağı için, **MongoDB'nin döküman tabanlı yapısı** burada büyük avantaj sağlayacak.

Bu yapı, yapay zekâ modelinden gelecek **değişken veri boyutlarına ve esnek veri yapısına** kolayca uyum sağlayabilecek.

➡ **Kısacası, PostgreSQL düzenli ve ilişkisel verilerin, MongoDB ise yapay zekâ çıktılarının ve dinamik verilerin yönetimi için tercih edilecek.**

# VERİ GEREKSİNİMLERİ

- ➡ **Kullanıcı Verileri:** Ad, e-posta, yaş, cilt tipi, geçmiş rahatsızlıklar, kullanılan ürün/ilaç bilgileri.
- ➡ **Analiz Verileri:** Fotoğraf yolu, analiz tarihi, tespit edilen cilt sorunları, yapay zeka skorları.
- ➡ **Ürün Verileri:** Ürün adı, kategori, içerik, uygun cilt tipleri, görsel, kaynak (API / manuel / scraping).
- ➡ **Öneri Verileri:** Kullanıcı analizi + ürün eşleşmesi + öneri nedeni.
- ➡ **Log Verileri:** Kullanıcının yaptığı işlemler (fotoğraf yükleme, analiz isteği, ürün görüntüleme vb.).

# KISITLAR

➡ Şu an **yalnızca mobil uygulama** geliştirilecek (web sürümü planlanmıyor).

➡ Ürün önerileri başlangıçta **manuel / sınırlı veri tabanına** dayanacak

Uygulama, ilk aşamada (örneğin proje demosu veya MVP sürümünde) **hazır, önceden eklenmiş ürün verilerini** kullanacak.

Bu ürünler veritabanına **manuel olarak** (siz ekip olarak veya admin panelinden) eklenecek.

Yapay zekâ, başlangıçta sadece “cilt tipi tespiti” yapacak; bu tespit sonucuna göre **manuel eşleştirilmiş ürünleri** listeleyecek.

## ♦ Örnek:

Eğer AI “yağlı cilt” olarak analiz ettiyse, veritabanında “yağlı ciltler için uygun ürünler” kategorisinde sizin önceden girdiğiniz ürünleri gösterecek.

Yani sistem, başlangıçta “otomatik öğrenme” değil “hazır ürün önerisi eşleştirmesi” yapacak.



İlerleyen aşamalarda (model eğitildikçe ve kullanıcı verileri geldikçe), bu manuel kısım yerini:

**otomatik ürün eşleştirmeye,**

hatta **kullanıcıya özel kişiselleştirilmiş önerilere** bırakacak.

➡ AI modeli, **eğitim verisi arttıkça** doğruluk oranını geliştirecek.

➡ Gerçek kullanıcı verileriyle test aşaması ilerleyen sprintlerde yapılacak.

# TİMELINE

## Kurulum & Başlangıç

Proje ortamı kurulacak, ekip görev dağılımı yapılacak, ilk test ekranı hazırlanacak.

1

## GLOWMANCE- AI SKIN ADVISOR

2

## Kullanıcı Sistemi

Kayıt, giriş ve kimlik doğrulama (JWT) işlemleri geliştirilecek.

3

## Ana Sayfa & Veri İşlemleri

Ana sayfa tasarımı yapılacak, veritabanı bağlantıları oluşturulacak.

4

## Yapay Zeka Entegrasyonu

Cilt analizi modeli entegre edilecek, görsel yükleme özelliği eklenecek.

5

## Ürün Öneri Sistemi

AI analiz sonuçlarına göre ürün öneri listesi oluşturulacak.

6

## UI/UX ve Testler

Tasarım iyileştirmeleri yapılacak, performans ve kullanıcı testleri gerçekleştirilecek.

7

## Güvenlik & Entegrasyon

API ve veritabanı güvenliği sağlanacak, hata düzeltmeleri yapılacak.

8

## Son test ve Kontrolller

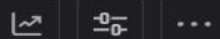
Demo tamamlanacak, PRD ve dokümantasyon hazırlanacak

Panoda arayın



Filtrele

Grupla v



## YAPILACAKLAR 3

4-Basit API endpoint testleri oluşturulacak

☒ GSA-5

7-AI modelini geliştirm için ortam kurulacak

☒ GSA-8

9-Modelin ilk basit analiz scriptini çalıştır ve çıktıları JSON formatında sakla

☒ GSA-10

+ Oluştur

## DEVAM EDİYOR 4

1- Veritabanı oluşturulacak.

☒ GSA-1

2-MongoDB ve PostgreSQL için tablolar oluşturulacak

☒ GSA-2

3-Backend ortamı kurulacak ve projeye başlanılacak

☒ GSA-4

8-Test veri seti hazırlanacak

☒ GSA-9

## TAMAM 3 ✓

5-Frontend tarafında kayıt ve giriş ekran taslakları oluşturulacak

☒ GSA-6

6-Basit bir anasayfa taslağı oluşturulacak

☒ GSA-7

10-İlk sunum hazırlanacak

☒ GSA-11

