

# PRODUCT REQUIREMENTS DOCUMENT (PRD)

**Project Name:** PrePath

**Team Members:** 4

**Version:** v1.0

## 1. Product Overview

### Vizyon

*Prepath, mülakat sürecindeki belirsizliği ve stresi ortadan kaldırarak her adayın potansiyelini güvenle ortaya koyabileceği, fırsat eşitliğine dayalı bir kariyer dünyası oluşturmayı hedefler.*

### Misyon

Yapay zeka destekli kişiselleştirilmiş, veri odaklı ve her yerden erişilebilir bir mülakat hazırlık platformu sunmak.

PrePath, her kullanıcıya kendi “kişisel AI kariyer koçu” deneyimini sağlayarak profesyonel gelişimini hızlandırmayı amaçlar.

## 2. Hedefler ve Başarı Kriterleri

### Product Goals

- Kullanıcıların 2 temel işlem yapabildiği basit fakat işleyen bir web arayüzü sağlamak:
  - CV optimizasyonu
  - Mülakat simülasyonu

- **Kullanıcının CV'sini analiz edip optimize eden tam çalışan bir “CV Optimization” modülü geliştirmek.**  
(CV parsing, eksik skill/keyword analizi, ATS uyumluluk, optimize metin üretimi)
- **Kullanıcının CV'sine göre kişisel mülakat soruları üreten ve sonunda rapor veren temel bir “Interview Simulator” modülü sunmak.** (RAG pipeline, ChromeDB/VectorDB, follow-up soruları,basit performans raporu)
- **AI modüllerinin backend ile entegre edildiği, uçtan uca çalışan bir monolitik mimari oluşturmak.**  
(API endpoint'leri, dockerize backend, Colab → ai/ klasörü bağlantısı)

## Success Metrics

**1 ay sonunda tüm MVP modülleri (CV Optimization + Interview Simulator + RAG + backend integration) minimum seviyede eksiksiz çalışıyor olacak.**

- Yüklenen CV'nin en az **%90 doğrulukla** parse edilmesi (eğitim, deneyim, projeler, skill'ler çıkarılacak).
- Kullanıcının CV'sine göre **en az 5 kişisel soru** ve **en az 1 follow-up sorusu** üretebilmesi.
- ChromeDB RAG pipeline'ın **soru bulma doğruluğunu %70+** olması (pozisyon'a göre doğru seti çekmesi).
- Tüm API endpointlerinin (**6 endpoint**) çalışır durumda olması ve swagger üzerinden test edilebilir olması.
- ChromeDB'nin backend'den erişilebilir olması ve embedding yükleme işlemlerinin stabil çalışması.
- Kullanıcıların hiçbir teknik bilgi olmadan:
  - **CV yükledikten sonra optimize edilen CV yi pdf formatında almak**
  - **Mülakat simülasyonunu başlatıp tamamlayabilmesi**
- Monorepo içinde **ai/, backend/, frontend/** yapılandırması tamamlanmış olacak.

### 3. User Personas

#### Persona 1: Yeni Mezun Yazılım Mühendisi – Ayşe(22)

Hedef:

- İlk işine girmek için teknik ve davranışsal mülakatlara hazırlanmak.
- CV'sini ATS uyumlu hale getirip daha profesyonel göstermek.
- Pozisyonu özel gerçekçi sorularla pratik yapmak.
- Eksik yönlerini anlamak ve kendini geliştirmek.

Zorluklar:

- Internetteki mülakat kaynakları çok dağınık → nereden başlayacağını bilmiyor.
- Gerçek mülakata benzer bir ortamda pratik yapma fırsatı yok.
- CV'sinin yeterince güçlü olup olmadığını bilmiyor.
- Staj projelerini nasıl anlatacağını bilmiyor.
- Sorulara cevap verirken panikliyor, follow-up sorularla zorlanıyor.

#### Persona 2: Kariyer Değiştiren Junior Backend Developer-Mehmet (26)

Hedef:

- Kariyerini frontend'den backend'e geçiş yaparak değiştirmek.
- Backend mülakatlarında sorulan teknik sorulara hazırlanmak.
- CV'sini hedef pozisyonuna göre yeniden şekillendirmek.

Zorluklar:

- Backend mülakat sorularını tek bir yerde toplamak zor.
  - CV'sinin hedef pozisyonuna uygun olmadığını biliyor ama nasıl optimize edeceğini bilmiyor.
  - Follow-up sorulara cevap vermekte zorlanıyor.
- 

## 4. MVP Scope – Core Features

### Feature 1 — CV Upload & Parsing

#### Description:

Kullanıcının yüklediği PDF formatındaki CV'nin yapay zeka ile okunması, eğitim, deneyim, projeler, skill set ve keyword'lerin çıkarılması.

#### User Story:

“Bir kullanıcı olarak, CV'mi yüklediğimde sistemin önemli bilgileri otomatik çıkarmasını istiyorum ki optimizasyon süreci hızlı olsun.”

#### Acceptance Criteria:

- Kullanıcı PDF dosyası yükleyebilir.
- Sistem CV'den eğitim, deneyim, projeler, skill'ler ve keywordleri çıkarır.
- Parsing işlemi 10 saniyeden kısa sürer.
- API: POST /cv/parse çalışır durumda.

#### Priority:

Zorunlu

---

### Feature 2 — CV Optimization Engine

#### Description:

Parse edilen CV üzerinden eksik keyword tespiti yaparak CV'nin ATS uyumlu şekilde yeniden düzenlenmesi ve optimize edilmiş metnin kullanıcıya sunulması.

#### **User Story:**

“Bir kullanıcı olarak, CV'min daha güçlü ve ATS uyumlu hale getirilmiş versiyonunu görmek istiyorum.”

#### **Acceptance Criteria:**

- Sistem CV'de eksik keywordleri listeler.
- Cümleleri güçlendirilmiş, daha profesyonel bir CV metni üretir.
- ATS uyumluluk sağlanmış olur.
- API: POST /cv/optimize aktif.

**Priority:** Zorunlu

---

### **Feature 3 — ChromeDB Setup & Question Retrieval (RAG)**

#### **Description:**

Mülakat sorularının embedding'lenip ChromeDB'ye kaydedilmesi ve CV'deki skilllere göre ilgili soruların getirilebilmesi.

#### **User Story:**

“Bir kullanıcı olarak, deneyimlerime uygun soruların sistem tarafından seçilmesini istiyorum.”

#### **Acceptance Criteria:**

- ChromeDB vector DB kurulmuş olmalı.
- Sorular embedding'lenmiş ve DB'ye kaydedilmiş olmalı.
- CV'de Python varsa Python soruları gelebilmeli.
- RAG doğruluğu %70+ olmalı.

**Priority:** Zorunlu

---

## Feature 4 — Interview Simulator (Question + Follow-up)

### Description:

Kullanıcının CV'sine göre 5–8 temel mülakat sorusu ve kullanıcı yanıtına göre follow-up sorusu üretilmesi. Davranışsal, teknik ve genel yetenek tarzında mülakat sorularının üretilmesi.

### User Story:

“Bir kullanıcı olarak, gerçek bir mülakata benzer şekilde kişiye özel sorularla pratik yapmak istiyorum.”

### Acceptance Criteria:

- /interview/start → 5–8 soru üretir.
- /interview/next-question → follow-up soruları üretir.
- Sistem kısa, net ve profesyonel soru üretebilir.
- Sorular pozisyonuna göre değişir (backend/frontend/data).

**Priority:** Zorunlu

---

## Feature 5 — Interview Evaluation Report

### Description:

Kullanıcının verdiği cevaplar doğrultusunda basit bir performans raporu oluşturulması.

### User Story:

“Bir kullanıcı olarak, verdiğim cevapların güçlü ve zayıf yönlerini görmek istiyorum.”

### Acceptance Criteria:

- Min. 3 paragraf değerlendirme yapılır.
- Güçlü yanlar listelenir.
- Geliştirilmesi gereken alanlar listelenir.
- API: /interview/evaluate & /interview/report çalışır.

**Priority:** Gerekli

---

## **Feature 6 — Minimal UI (CV + Interview)**

### **Description:**

Kullanıcının CV yükleyip optimize edilmiş sonucu görebildiği ve mülakat simülasyonunu başlatabildiği temel web arayüzü.

### **User Story:**

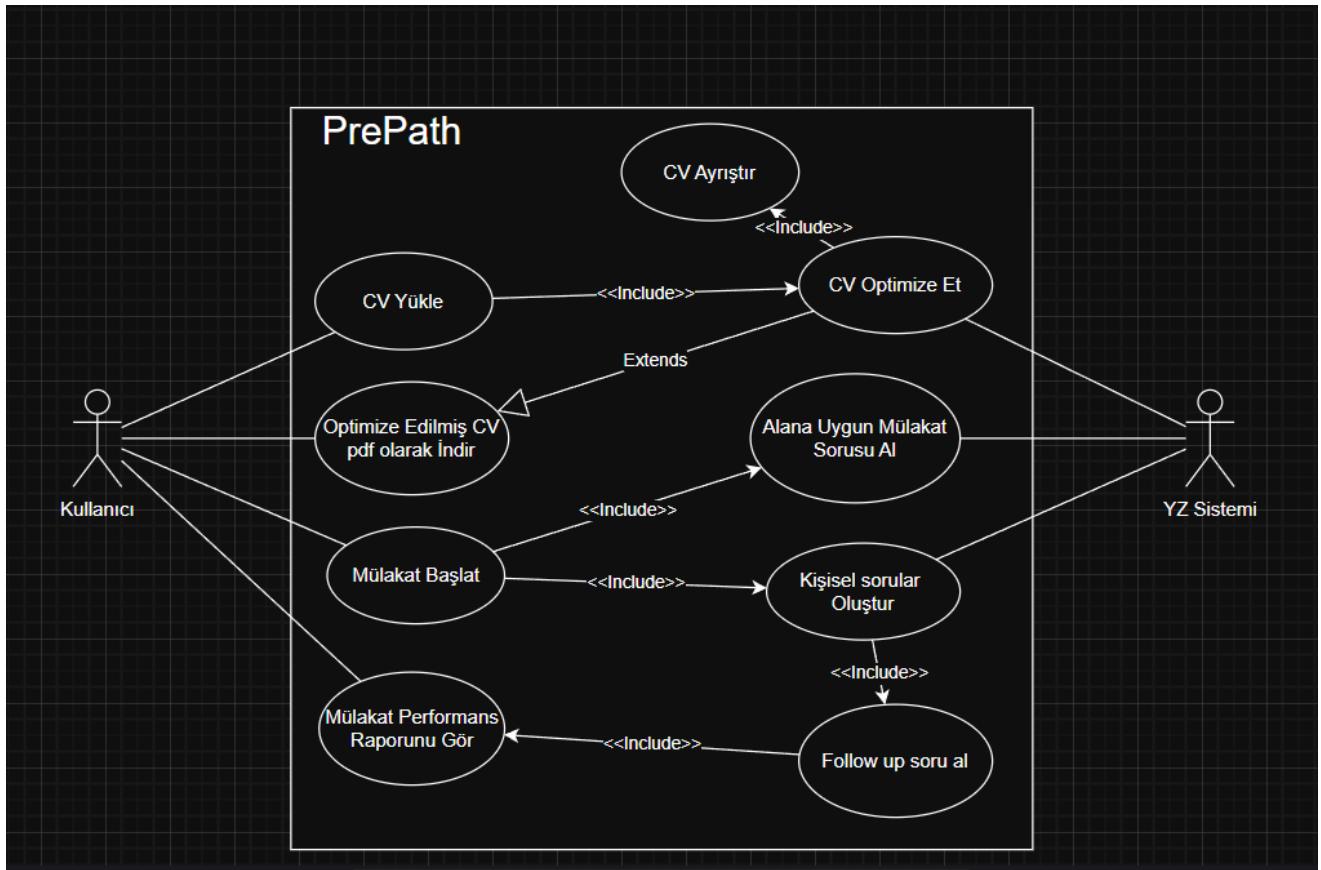
“Bir kullanıcı olarak, karmaşık adımlar olmadan kolayca CV yüklemek ve mülakata başlamak istiyorum.”

### **Acceptance Criteria:**

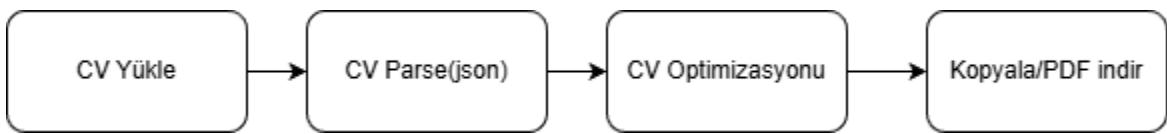
- CV upload ekranı çalışır.
- Optimize edilmiş sonuç ekranında gösterilir.
- Mülakat başlatma butonu vardır.
- Sorular tek tek gelir, kullanıcı cevap yazar.
- Performans raporu UI'da görüntülenir.

**Priority:** Zorunlu

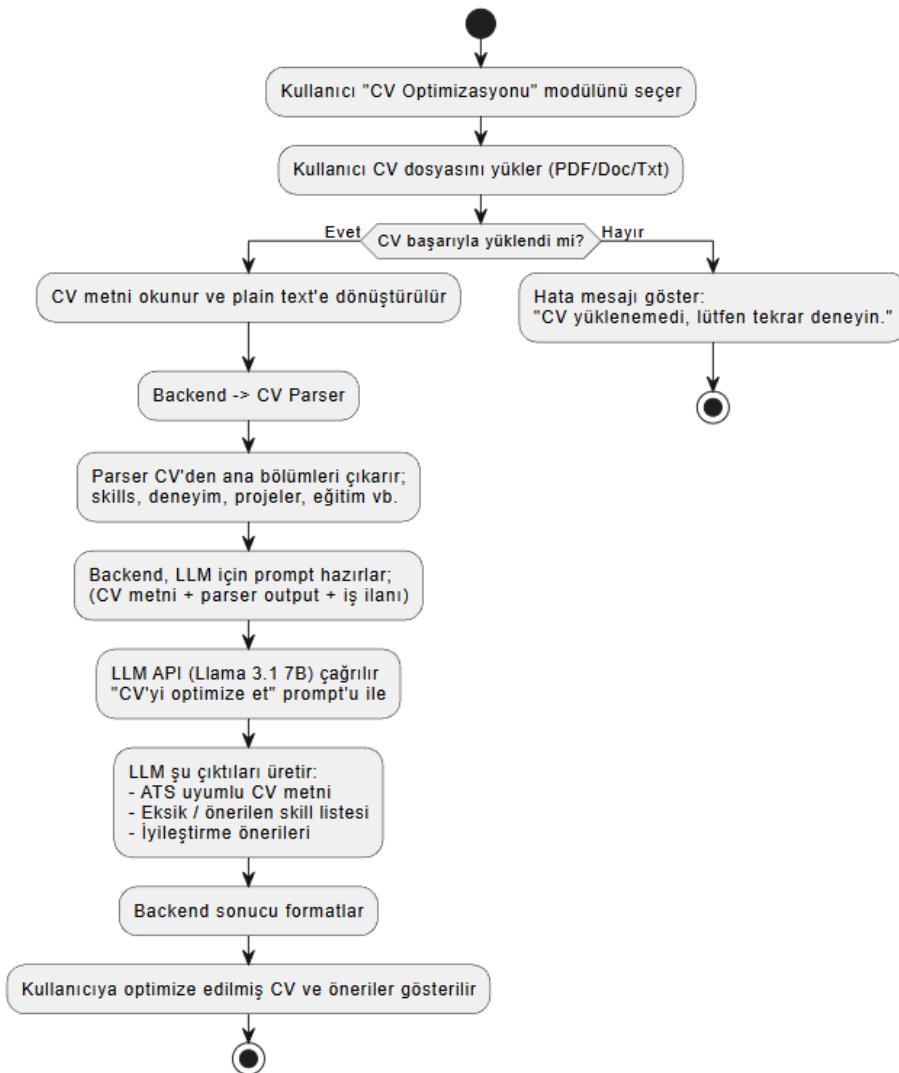
---

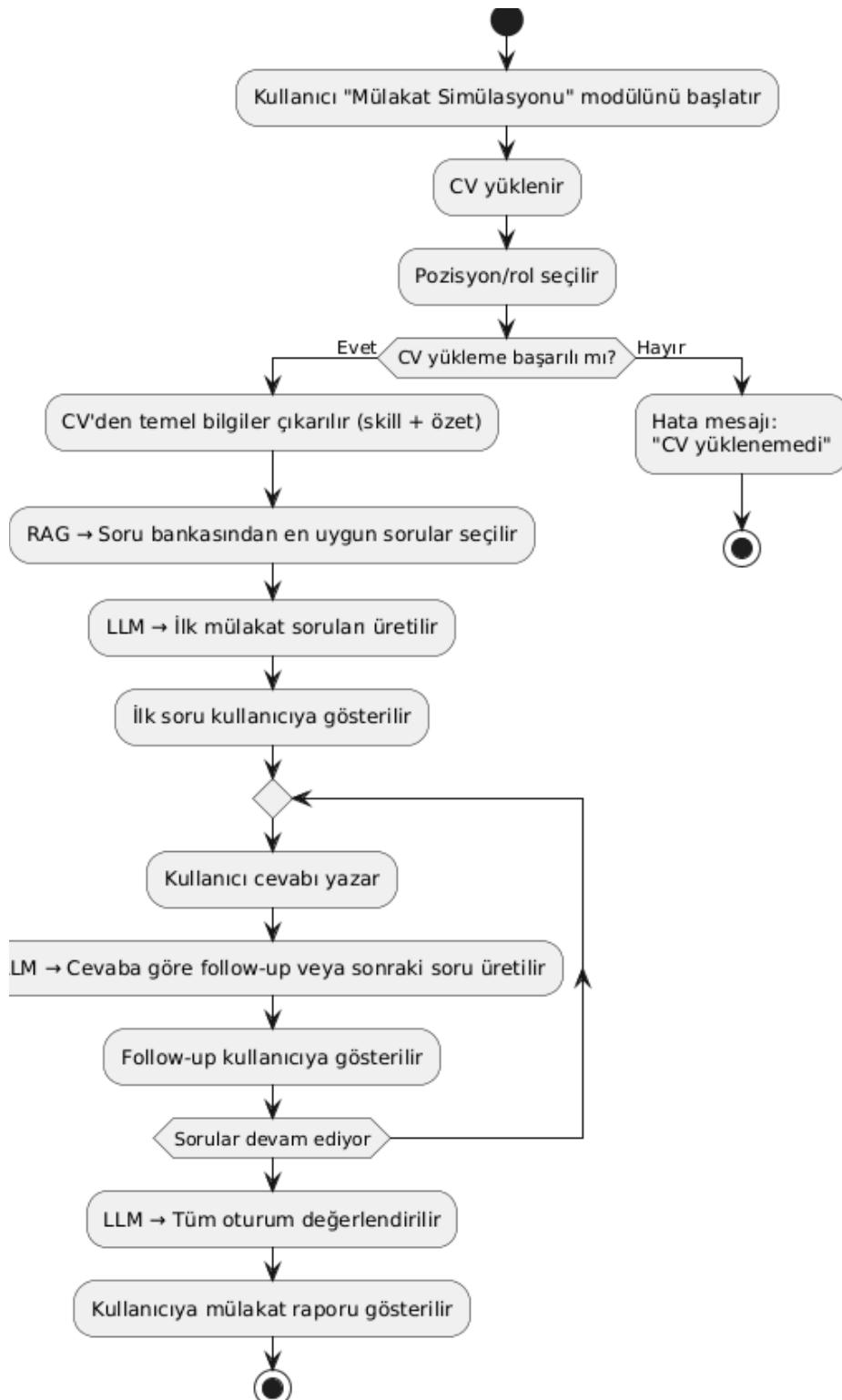


Use Case Diyagramı

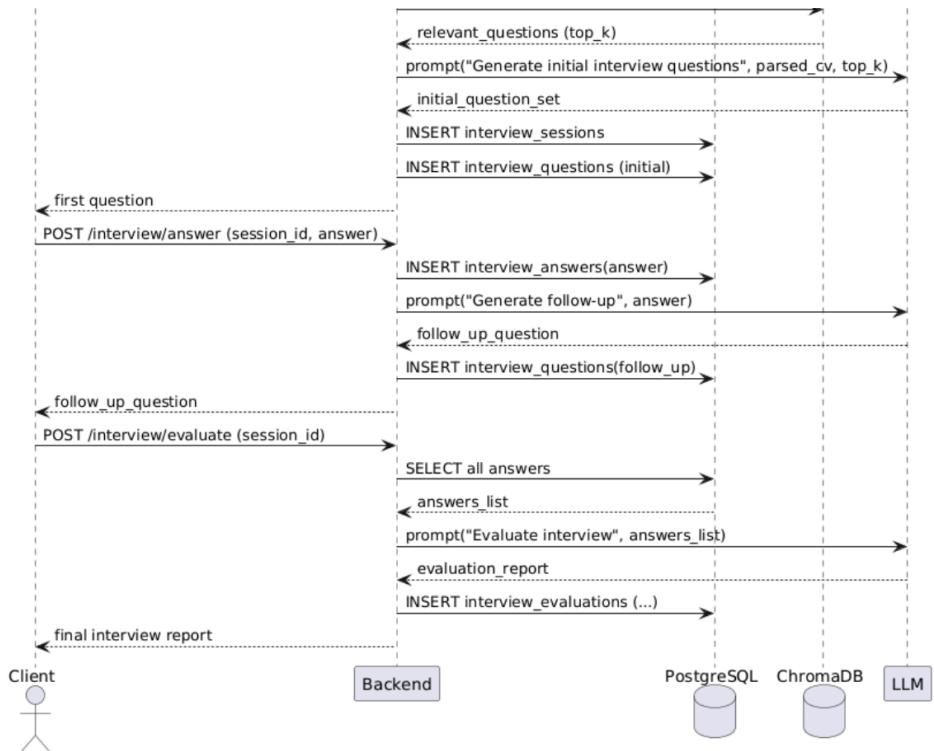
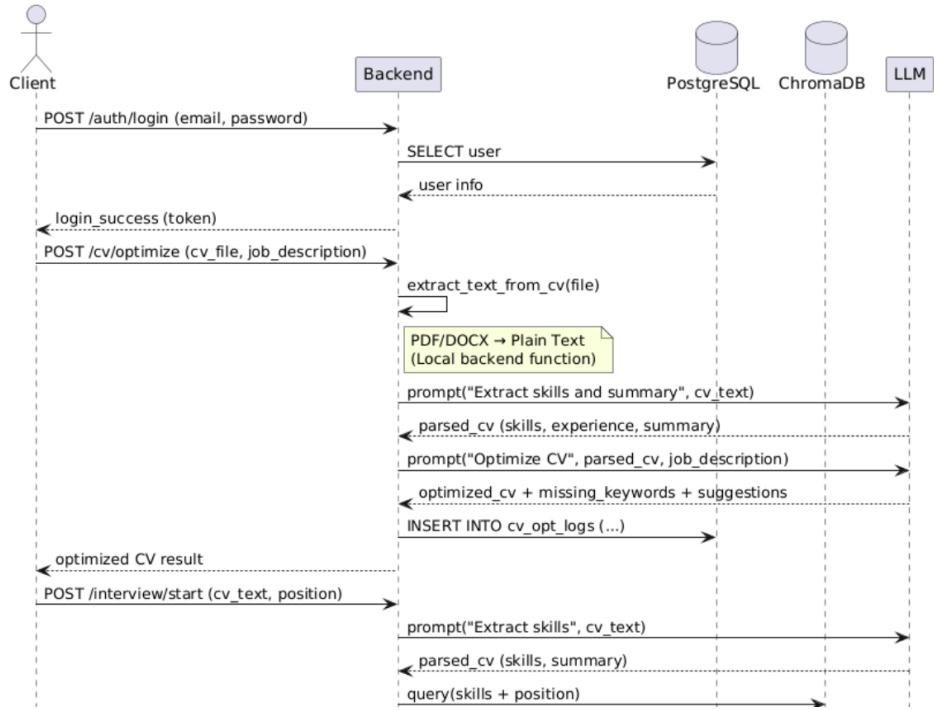


### CV Optimizasyon Modülü Akış Diyagramı





### FULL SYSTEM SEQUENCE DIAGRAM



## **6. System Architecture**

*Add diagram or text description.*

---

## **7. Risks & Assumptions**

- -
- 

## **8. Timeline & Responsibility Breakdown**

- Frontend:
- Backend:
- AI:
- DevOps & Scrum: