

T.C.

SELÇUK ÜNİVERSİTESİ

TEKNOLOJİ FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ

## YAPAY ZEKA DESTEKLİ ÜNİVERSİTE BİLGİ ASİSTANI: SELÇUK AI ASİSTAN

Doğukan BALAMAN (203311066)

Ali YILDIRIM (203311008)

MÜHENDİSLİK TASARIMI / BİLGİSAYAR MÜHENDİSLİĞİ UYGULAMALARI

OCAK-2025

KONYA

Her Hakkı Saklıdır

## PROJE KABUL VE ONAYI

Doğukan Balaman ve Ali Yıldırım tarafından hazırlanan "Yapay Zeka Destekli Üniversite Bilgi Asistanı: Selçuk AI Asistan" adlı proje çalışması .../.../.../2025 tarihinde aşağıdaki jüri üyeleri tarafından oy birliği/oy çokluğu ile Selçuk Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği bölümünde Bitirme Projesi olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Danışman

Prof. Dr. Nurettin DOĞAN

Üye

Dr. Öğr. Üyesi Onur İNAN

Üye

Unvanı Adı SOYADI

Yukarıdaki sonucu onaylarım.

**\*\*Dr.Öğr.Üyesi Onur İNAN** bu proje çalışmasının ikinci danışmanıdır.

## PROJE BİLDİRİMİ

Bu projedeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve proje yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

## DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by project rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

İmza

Doğukan BALAMAN

Ali YILDIRIM

Tarih: ....../....../2025

## ÖZET

### BİTİRME PROJESİ

**YAPAY ZEKA DESTEKLİ ÜNİVERSİTE BİLGİ ASİSTANI: SELÇUK AI ASİSTAN**

**Doğukan BALAMAN - Ali YILDIRIM**

**SELÇUK ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Danışman: Prof. Dr. Nurettin DOĞAN**

**2025, 20 Sayfa**

**Jüri**

**Prof. Dr. Nurettin DOĞAN**

**Dr. Öğr. Üyesi Onur İNAN**

## Diğer Üyenin Unvanı Adı SOYADI

Bu çalışmada, Selçuk Üniversitesi öğrencileri ve personelinin akademik ve idari süreçlere ilişkin bilgilere hızlı, doğru ve erişilebilir biçimde ulaşabilmesi amacıyla geliştirilen yapay zekâ destekli üniversite bilgi asistanı sunulmaktadır. Sistem, Flutter tabanlı çoklu platform istemci, FastAPI tabanlı servis katmanı ve Retrieval-Augmented Generation yaklaşımıyla zenginleştirilmiş dil modeli altyapısından oluşmaktadır. RAG katmanı, üniversiteye ait doğrulanmış kaynaklardan oluşturulan bilgi tabanını vektörleştirerek sorgu anında ilgili bağlamı çıkarmakta, böylece halüsinasyon riskini azaltmakta ve yanıtların kaynak gösterimiyle denetlenebilirliğini artırmaktadır. Model katmanında, verinin kurum dışına çıkmaması hedefiyle Ollama üzerinde yerel LLM varsayılan seçenek olarak tasarlanmış, ayrıca donanım ve performans gereksinimlerine göre alternatif sağlayıcılarla genişletilebilecek bir provider mimarisi uygulanmıştır. Geliştirilen prototip, sık sorulan sorular, bölüm ve fakülte bilgileri, akademik takvim, yurt ve ulaşım ve kampüs hizmetleri gibi başlıklarda doğal dilde etkileşim sağlamaktadır. Test sonuçları, kritik bilgi doğruluğunun yüzde yüz seviyesine ulaştığını ve yanıt sürelerinin ortalama 1,5-5 saniye aralığında seyrettiğini göstermiştir.

**Anahtar Kelimeler:** API, FAISS, FastAPI, Flutter, Gizlilik, LLM, RAG, Üniversite Bilgi Sistemi, Yapay Zekâ, Yerel Çalışma

---

## ABSTRACT

### ENGINEERING DESIGN PROJECT

### AI-POWERED UNIVERSITY INFORMATION ASSISTANT: SELCUK AI ASSISTANT

**Doğukan BALAMAN - Ali YILDIRIM**

**SELCUK UNIVERSITY**

**FACULTY OF TECHNOLOGY**

**DEPARTMENT OF COMPUTER ENGINEERING**

**Advisor: Prof. Dr. Nurettin DOĞAN**

**2025, 20 Pages**

**Jury**

**Prof. Dr. Nurettin DOĞAN**

**Dr. Öğr. Üyesi Onur İNAN**

**Other Member's Title Name SURNAME**

This study presents an AI-powered university information assistant designed to provide fast, accurate, and accessible answers for students and staff at Selçuk University. The system comprises a multi-platform Flutter client, a FastAPI backend service, and a Retrieval-Augmented Generation pipeline that grounds responses in a curated institutional knowledge base. The RAG layer embeds and indexes the knowledge base using FAISS, retrieves relevant context at query time, reduces hallucinations, and improves auditability by returning source-backed answers. To address privacy and data sovereignty, a local LLM via Ollama is used by default, and a provider-based architecture enables optional alternative model backends. The prototype supports natural-

language interaction for typical university queries such as departments, academic calendar, campus services, accommodation, and transportation. Evaluation results indicate strong critical-fact accuracy at one hundred percent and average response times ranging from approximately 1.5 to 5 seconds.

**Keywords:** API, Artificial Intelligence, FAISS, FastAPI, Flutter, LLM, Local Inference, Privacy, RAG, University Information Assistant

## ÖNSÖZ

Bu çalışma, Selçuk Üniversitesi öğrencilerinin ve personelinin üniversiteye ilişkin bilgilere erişimini hızlandırmak amacıyla geliştirilen Selçuk AI Asistan projesinin bitirme projesi raporudur. Çalışma boyunca akademik ve teknik katkılarıyla destek olan danışmanlarımız Prof. Dr. Nurettin Doğan ve Dr. Öğr. Üyesi Onur İnan'a teşekkür ederiz.

Doğukan BALAMAN

Ali YILDIRIM

Konya / 2025

## İÇİNDEKİLER

ÖZET .....	iv
ABSTRACT .....	v
ÖNSÖZ .....	vi
İÇİNDEKİLER .....	vii
SİMGELER VE KISALTMALAR .....	viii
1. GİRİŞ .....	1
1.1. Problem Tanımı .....	1
1.2. Amaç ve Kapsam .....	1
1.3. Katkılar .....	2
2. KAYNAK ARAŞTIRMASI .....	3
2.1. Büyük Dil Modelleri ve Transformer .....	3
2.2. RAG Yaklaşımı .....	3
2.3. Yerel Çalışma ve Gizlilik .....	3
3. MATERYAL VE YÖNTEM .....	4
3.1. Kullanılan Teknolojiler .....	4
3.2. Veri Kaynakları ve Bilgi Tabanı .....	5
3.3. RAG Boru Hattı .....	5
3.4. Değerlendirme Yöntemi .....	5

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA .....	6
4.1. Sistem Tasarımı .....	6
4.2. Test Senaryoları .....	7
4.3. Nicel Sonuçlar .....	8
4.4. Nitel Gözlemler .....	9
5. SONUÇLAR VE ÖNERİLER .....	10
5.1. Sonuçlar .....	10
5.2. Öneriler .....	10
KAYNAKLAR .....	11
EKLER .....	12
ÖZGEÇMİŞ .....	14

---

## SİMGELER VE KISALTMALAR

### Simgeler

Bu projede kullanılan özel simge bulunmamaktadır.

### Kısaltmalar

**AI** - Artificial Intelligence / Yapay Zekâ

**API** - Application Programming Interface

**FAISS** - Facebook AI Similarity Search

**JWT** - JSON Web Token

**LLM** - Large Language Model / Büyük Dil Modeli

**RAG** - Retrieval-Augmented Generation / Geri Getirim Destekli Üretim

**SSE** - Server-Sent Events

---

## 1. GİRİŞ

Üniversitelerde öğrencilerin ve personelin ihtiyaç duyduğu bilgi kaynakları, web sayfaları, duyuru sistemleri, yönetmelikler, akademik takvim ve birimlerin farklı kanallarındaki içeriklere dağılmış durumdadır. Bu durum, özellikle yeni kayıt dönemleri, sınav haftaları veya idari süreçlerin yoğunlaştığı zamanlarda bilgiye erişimi zorlaştırmakta ve telefon, e-posta ve yüz yüze danışma kanallarında iş yükü oluşturmaktadır.

Büyük dil modellerindeki gelişmeler, doğal dilde soru-cevap sistemlerinin kullanımını yaygınlaştırmıştır. Ancak bu modeller, doğrulanmış kurumsal kaynaklarla beslenmediğinde halüsinasyon olarak adlandırılan hatalı fakat ikna edici yanıtlar üretebilmektedir. Üniversite gibi kritik bilginin önemli olduğu ortamlarda bu riskin azaltılması gerekir.

Bu bitirme projesinde geliştirilen Selçuk AI Asistan, Selçuk Üniversitesi kapsamında sık sorulan sorulara, yönetmelik ve birim bilgilerine ve kampüs hizmetlerine ilişkin sorulara doğal dilde yanıt üretmeyi amaçlayan,

kaynak temelli ve yerel çalışabilen bir bilgi asistanıdır.

## 1.1. Problem Tanımı

Problem, kullanıcıların farklı sistemlerde dağınık halde bulunan üniversite bilgilerine hızlı ve güvenilir biçimde erişememesi ve mevcut kanalların yoğunluk ve erişilebilirlik açısından yetersiz kalmasıdır. Bu kapsamda tek bir arayüzden erişim, doğrulanabilir kaynak temelli yanıt, gizlilik ve veri egemenliği gereksinimleri öne çıkmaktadır.

## 1.2. Amaç ve Kapsam

Çalışmanın amacı, Selçuk Üniversitesi için mobil öncelikli bir bilgi asistanı tasarlamak ve prototipini geliştirmektir. Sistem, Flutter istemci, FastAPI servis katmanı ve RAG destekli LLM altyapısından oluşur. Kapsam, üniversite tanıtım bilgileri, fakülte ve bölüm bilgileri, akademik takvim, kampüs hizmetleri, yurt ve ulaşım ile sık sorulan soru alanlarını içerir.

## 1.3. Katkılar

Bu çalışmanın başlıca katkıları şunlardır: Yerel LLM ile varsayılan olarak kurum içi çalışabilen mimari, RAG tabanlı kaynak gösterimli yanıt üretimi, provider tabanlı tasarım ile model sağlayıcılarının değiştirilebilir olması ve Flutter ile çoklu platform istemci sunulmasıdır.

# 2. KAYNAK ARAŞTIRMASI

## 2.1. Büyük Dil Modelleri ve Transformer

Transformer mimarisi, kendine dikkat mekanizması ile diziler üzerindeki bağımlılıkları paralel şekilde modelleyerek doğal dil işleme alanında önemli bir sıçrama sağlamıştır. Güncel büyük dil modelleri, geniş ölçekli veri üzerinde ön eğitim ve talimatla ince ayar gibi süreçlerle çok çeşitli görevlerde yüksek performans gösterebilmektedir. Buna karşın, eğitim verisi dışındaki spesifik kurumsal bilgilere erişimde yetersiz kalabilir veya hatalı yanıtlar üretebilir.

## 2.2. RAG Yaklaşımı

Retrieval-Augmented Generation, üretici modeli dış bilgi kaynaklarıyla birleştirerek sorgu sırasında ilgili belgeleri geri getirir ve yanıt üretiminde bu bağlamı kullanır. Böylece hem doğruluk artar hem de kaynak temelli yanıtlar üretmek mümkün olur. Bu projede kurumsal bilgi tabanı metinleri parçalara ayrılarak vektör uzayına gömülmekte ve FAISS üzerinde indekslenmektedir.

## 2.3. Yerel Çalışma ve Gizlilik

Kurum içi kullanım senaryolarında, kullanıcı verilerinin üçüncü taraf servis sağlayıcılarına gönderilmesi gizlilik ve mevzuat açısından risk oluşturabilir. Bu nedenle yerel çıkarım yaklaşımı önem kazanmıştır. Ollama gibi araçlar, belirli büyük dil modellerinin yerel makinede çalıştırılmasını kolaylaştırır. Bu çalışmada performans ve gizlilik dengesi, provider tabanlı mimariyle yönetilmiştir.

## 3. MATERİYAL VE YÖNTEM

### 3.1. Kullanılan Teknolojiler

Sistemin istemci ve sunucu bileşenlerinde kullanılan başlıca teknolojiler aşağıdaki çizelgede verilmiştir.

**Çizelge 3.1.** Kullanılan teknoloji yığını

Katman	Teknoloji	Amaç
İstemci	Flutter + Dart	Çoklu platform mobil ve web arayüzü
Sunucu	FastAPI + Python	REST ve SSE servisleri ve iş mantığı
RAG	LangChain + FAISS	Geri getirim ve bağlam oluşturma
LLM	Ollama Llama 3.x	Yerel ve alternatif model çalıştırma
Test	pytest, GitHub Actions	Kalite ve otomasyon

### 3.2. Veri Kaynakları ve Bilgi Tabanı

Bilgi tabanı, üniversiteye ait doğrulanmış sayfalar, yönetmelikler, sık sorulan sorular ve birim tanıtım metinlerinden oluşacak şekilde kurgulanmıştır. İçerik, sürüm kontrolü ile yönetilecek biçimde düz metin ve yapılandırılmış kaynaklara dönüştürülmüştür. İçerik dosyaları ayrı bir dizinde tutulmakta ve indeksleme adımı otomasyonla yeniden çalıştırılabilmektedir.

### 3.3. RAG Boru Hattı

RAG boru hattı üç temel aşamadan oluşur: ön işleme ve parçalara ayırma, gömme üretimi ve FAISS indeksleme, sorgu sırasında geri getirim ve yanıt üretimi. Bu çalışmada parça boyutu sekiz yüz karakter, örtüşme yüz karakter, geri getirimde en çok beş parça ve düşük alakalı parçaları elemek için benzerlik eşiği kullanılmıştır. Yanıt üretimi sırasında strict mode etkinleştirilerek bilgi tabanı dışı spekülasyonlar azaltılmıştır.

### 3.4. Değerlendirme Yöntemi

Değerlendirme, doğruluk, gecikme ve gizlilik hedefleri üzerinden yapılmıştır. Test seti, üniversite hakkında kritik ve kritik olmayan genel sorular olacak şekilde iki gruba ayrılmıştır. Ayrıca RAG katmanının kaynak eşleşmesi, halüsinasyon oranı, eşzamanlı kullanıcı davranışı ve istemci tarafı kullanıcı deneyimi gözlenmiştir.

## 4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

### 4.1. Sistem Tasarımı

Sistem üç katmanlı bir mimari ile tasarlanmıştır: Flutter istemci uygulaması, FastAPI servis katmanı, RAG ve LLM katmanı. İstemci, kullanıcıdan aldığı soruyu HTTP üzerinden sunucuya iletir, sunucu RAG geri getirmesi yaparak bağlamı oluşturur ve seçili sağlayıcı ile yanıt üretir. Yanıtlar, kullanıcı deneyimini iyileştirmek amacıyla Server-Sent Events üzerinden akış şeklinde iletilebilir.

Sunucu katmanı FastAPI ile geliştirilmiştir. Uç noktalar, sağlık kontrolü, sohbet, RAG indeks yönetimi ve sistem durum bilgilerini kapsar. Model katmanı, farklı çalışma yöntemlerini tek bir arayüz altında birleştiren provider yaklaşımıyla tasarlanmıştır. Varsayılan provider, verinin kurum dışına çıkmaması için Ollama üzerindeki yerel modeldir.

Sistemde gizlilik hedefi gereği, kullanıcı mesajlarının üçüncü taraf servislere zorunlu olarak gönderilmemesi amaçlanmıştır. RAG katmanında strict mode ve kaynak temelli yanıt yaklaşımı, yanlış bilgi üretimi riskini azaltmaya yönelik önemli bir kontrol mekanizmasıdır.

### 4.2. Test Senaryoları

Testler, kritik bilgi doğrulama, genel soru-cevap doğruluğu, RAG kaynak eşleşmesi, performans, eşzamanlı kullanıcı ve hata dayanıklılığı başlıklarında yürütülmüştür. Kritik sorular, üniversitenin kuruluş yılı, şehir ve konum, fakülte ve bölüm gibi yanlış yanıtlandığında doğrudan kullanıcıyı yanıltabilecek bilgilerdir.

### 4.3. Nicel Sonuçlar

Elde edilen bulgular, farklı model sağlayıcıları ile elde edilen tipik sonuçları özetlemektedir.

**Çizelge 4.1.** Özet performans ve doğruluk sonuçları

Metrik	Hedef	Gözlem	Not
Kritik bilgi doğruluğu	$\geq \%95$	$\%100$	Örnek kritik soru seti
Genel doğruluk	$\geq \%90$	$\%96$	Genel SSS seti
Halüsinasyon oranı	$\leq \%5$	$\%0$	Kaynak dışı iddia üretimi
Ortalama yanıt süresi	$\leq 3$ sn	1,47 sn	Hızlı sağlayıcı senaryosu
Ortalama yanıt süresi yerel	$\leq 7$ sn	3-5 sn	Ollama Llama 3.x
Eşzamanlı kullanıcı	$\geq 50$	100+	Yük testi ile gözlem

### 4.4. Nitel Gözlemler

RAG temelli yaklaşım, özellikle yönetmelik ve birim bilgisi gibi metne dayalı sorularda yanıt kalitesini belirgin



biçimde artırmıştır. Kaynak gösterimi, kullanıcıların yanıtı doğrulamasını kolaylaştırmış ve güven algısını yükseltmiştir. Yerel model kullanımında gecikme artmakla birlikte, gizlilik kazanımı ve kapalı ağ çalışabilme özelliği önemli bir avantaj sağlamıştır.

Sonuçlar, kaynak temelli yanıt üreten RAG yaklaşımının üniversite bilgi asistanı senaryosunda uygun olduğunu göstermektedir. Bununla birlikte, sistemin başarısı bilgi tabanının güncelliği ve kapsamı ile doğrudan ilişkilidir.

## 5. SONUÇLAR VE ÖNERİLER

### 5.1. Sonuçlar

Bu bitirme projesinde Selçuk Üniversitesi için yapay zekâ destekli bir bilgi asistanı tasarlanmış ve prototipi geliştirilmiştir. Sistem, Flutter istemci, FastAPI backend ve RAG destekli LLM katmanından oluşmakta ve yerel çalışabilen bir çözüm sunmaktadır.

Test ve değerlendirme bulguları, kritik bilgi doğruluğu, kaynak eşleşmesi ve kullanıcı deneyimi açısından yaklaşımın uygulanabilir olduğunu göstermiştir. Yerel model seçeneği, gizlilik hedefleri açısından güçlü bir çözüm sunarken, provider mimarisi performans ihtiyaçlarına göre genişletilebilir bir yapı sağlamıştır.

Çalışmanın kısıtları şunlardır: Bilgi tabanının kapsamı sınırlıdır ve düzenli güncelleme gerektirir, yerel model performansı kullanılan donanımına bağlıdır, öğrenci bilgi sistemi gibi kişisel veri gerektiren entegrasyonlar bu prototipin kapsamı dışındadır.

### 5.2. Öneriler

Bu çalışma, aşağıdaki geliştirmelerle daha güçlü bir kurumsal ürüne dönüştürülebilir: Bilgi tabanının otomatik güncellenmesi, üniversite sistemleriyle güvenli entegrasyon ve kimlik doğrulama, gelişmiş gözlemlenebilirlik ve metrikler, çok dilli destek ve sesli etkileşim, alan uyarlamalı ince ayar veya kurum içi RAG sıralayıcı ile daha yüksek isabet oranı sağlanması önerilmektedir.

## KAYNAKLAR

Vaswani, A., Shazeer, N., Parmar, N., et al., 2017, Attention Is All You Need, *NeurIPS*.

Lewis, P., Perez, E., Piktus, A., et al., 2020, Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, *NeurIPS*.

Johnson, J., Douze, M., & Jégou, H., 2019, Billion-scale similarity search with GPUs, *IEEE T-BD*.

LangChain Documentation, 2025, <https://python.langchain.com/> [Erişim: 2025].

FastAPI Documentation, 2025, <https://fastapi.tiangolo.com/> [Erişim: 2025].

Flutter Documentation, 2025, <https://docs.flutter.dev/> [Erişim: 2025].

Ollama Documentation, 2025, <https://ollama.com/> [Erişim: 2025].

## EKLER

### EK-1

#### Kontrol Edilecek Hususlar Kontrol Listesi

Kontrol Edilecek Hususlar	Evet	Hayır
Sayfa yapısı uygun mu?	✓	
Şekil ve çizelge başlık ve içerikleri uygun mu?	✓	
Denklem yazımları uygun mu?	✓	
İç kapak, onay sayfası, Proje bildirimi, özet, abstract, önsöz uygun yazıldı mı?	✓	
Proje yazımı doğru sıralamada mıdır?	✓	
Kaynaklar soyadı sırasına göre verildi mi?	✓	
Kaynaklarda verilen her bir yayına proje içerisinde atıfta bulunuldu mu?	✓	
Kaynaklar açıklanan yazım kuralına uygun olarak yazıldı mı?	✓	
Proje içerisinde kullanılan şekil ve çizelgelerde kullanılan ifadeler Türkçe'ye çevrilmiş mi?	✓	
Projenin içindekiler kısmı, proje içerisinde verilen başlıklara uygun hazırlanmış mı?	✓	

Yukarıdaki verilen cevapların doğruluğunu kabul ediyorum.

	Unvanı Adı SOYADI	İmza
Öğrenci:	Doğukan BALAMAN	.....
Öğrenci:	Ali YILDIRIM	.....
Danışman:	Prof. Dr. Nurettin DOĞAN	.....

### EK-2: Sistem Mimarisi Detayları

Selçuk AI Asistan sisteminin üç katmanlı mimarisi şu şekilde çalışmaktadır:

**İstemci Katmanı:** Flutter framework kullanılarak geliştirilmiş, iOS, Android ve web platformlarında çalışabilen çoklu platform uygulaması. GetX state management ile durum yönetimi sağlanmıştır.

**Servis Katmanı:** FastAPI ile Python dilinde geliştirilmiş REST API ve Server-Sent Events desteği. Pydantic modelleri ile güçlü tip kontrolü ve doğrulama sağlanmıştır.

**RAG ve LLM Katmanı:** LangChain kütüphanesi ile oluşturulmuş Retrieval-Augmented Generation boru hattı, FAISS vektör veritabanı ile hızlı benzerlik araması ve Ollama üzerinden yerel LLM çalıştırma.

## ÖZGEÇMİŞ

### KİŞİSEL BİLGİLER - Doğukan BALAMAN

**Adı Soyadı:** Doğukan Balaman

**Uyruğu:** T.C.

**Doğum Yeri ve Tarihi:** ...

**Telefon:** ...

**E-mail:** ...

### EĞİTİM

Derece	Adı, İlçe, İl	Bitirme Yılı
Lise:	...	...
Üniversite:	Selçuk Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği	2025

### İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
...	...	...

### UZMANLIK ALANI

Yapay zekâ, veri mühendisliği ve mobil uygulama geliştirme.

### KİŞİSEL BİLGİLER - Ali YILDIRIM

**Adı Soyadı:** Ali Yıldırım

**Uyruğu:** T.C.

Doğum Yeri ve Tarihi: ...

Telefon: ...

E-mail: ...

## EĞİTİM

Derece	Adı, İlçe, İl	Bitirme Yılı
Lise:	...	...
Üniversite:	Selçuk Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği	2025

## İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
...	...	...

## UZMANLIK ALANI

Backend geliştirme, API tasarımı ve yapay zekâ uygulamaları.