

## TAKIM KAPTANI GERİ BİLDİRİM FORMU

### Bölüm 1 - Takım Bilgileri

Muhammed Salih Üstün

### Bölüm 2 - Ekip içi Çalışma Süreci

Kriter	√	X
Ekip içi iletişim düzenliydi	√	<input type="checkbox"/>
Görev dağılımı netti	√	<input type="checkbox"/>
Tüm üyeler sürece aktif katıldı	√	<input type="checkbox"/>
Toplantılar planlandığı şekilde yapıldı	√	<input type="checkbox"/>
Zaman yönetimi yeterliydi	√	<input type="checkbox"/>
Ekip içi anlaşmazlıklar uygun şekilde çözüldü	√	<input type="checkbox"/>

### Bölüm 3 - Projenin İlerleme Durumu

Kriter	√	X
Problem tanımı netleşti	√	<input type="checkbox"/>
İş paketleri tanımlandı	√	<input type="checkbox"/>
Sorumlular belirlendi	√	<input type="checkbox"/>
Proje başlangıç çıktıları oluşturuldu	√	<input type="checkbox"/>

T.C.  
FIRAT ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ  
BİLGİSAYAR MÜHENDİSLİĞİ TASARIM DERSİ PROJESİ KAPSAMINDA  
TAKIM ÇALIŞMASI PROJESİ

### 1. GİRİŞ

Bu proje, kullanıcıların doğal dil ile etkileşime geçebileceği, sesli komutları anlayıp sesli yanıtlar verebilen yapay zeka tabanlı bir "Kişisel Asistan" geliştirmeyi amaçlamaktadır. Günümüzde mobil cihazların yaygınlaşması ve yapay zeka teknolojilerinin (LLM) gelişmesiyle birlikte, kullanıcıların teknolojiyi daha sezgisel kullanma ihtiyacı artmıştır.

"Jarvis" kod adlı bu proje, sadece basit komutları yerine getiren bir araç değil, aynı zamanda bağlamı anlayabilen ve doğal bir sohbet sürdürebilen akıllı bir sistem olarak tasarlanmıştır. Proje, modern yazılım mimarilerinden biri olan Mikroservis yaklaşımını benimseyerek; güvenliği ve yönetimi C# (.NET Core) üzerinde, yapay zeka işlemlerini Python (FastAPI) üzerinde ve kullanıcı arayüzünü React Native mobil platformunda kurgulamıştır. Bu yapı, sistemin ölçeklenebilir, güvenli ve modüler olmasını sağlamaktadır.

## 1.1. Amaç ve Hedefler

Projenin temel amacı, sesli iletişim teknolojilerini modern web ve mobil teknolojileriyle birleştirerek uçtan uca çalışan bir asistan ortaya koymaktır. Projenin alt hedefleri şunlardır:

1. Doğal Sesli Etkileşim: Kullanıcının sesini yüksek doğrulukla metne çevirmek (STT), anlamlandırmak ve insan doğallığında sesli yanıt (TTS) üretmek.
2. Hibrit Zeka Mimarisi: Veri gizliliğini ve maliyet optimizasyonunu sağlamak amacıyla Yerel LLM (Ollama/Llama 3.2) ile yüksek kaliteli ses sentezi (ElevenLabs) servislerini entegre etmek.
3. Güvenli ve Yönetilebilir Backend: Kullanıcı kimlik doğrulama, yetkilendirme ve loglama işlemlerini merkezi bir C# Gateway üzerinden yönetmek.
4. Platform Bağımsız Mobil Erişim: React Native ile geliştirilen arayüz sayesinde Android ve iOS cihazlarda, arka planda dahi çalışabilen (Wake-Word simülasyonu) bir deneyim sunmak.
5. Veri Kalıcılığı: Tüm etkileşimleri ve kullanıcı verilerini PostgreSQL veritabanında güvenli ve optimize bir şekilde saklamak.

## 2. YÖNTEM

Proje geliştirme sürecinde N-Katmanlı (N-Tier) ve Mikroservis mimarileri bir arada kullanılmıştır. Sistem, sorumlulukların ayrılığı ilkesine (Separation of Concerns) uygun olarak üç ana bileşene ayrılmıştır:

1. Backend Gateway (C# / ASP.NET Core):
  - Sistemin dış dünyaya açılan kapısıdır.
  - Kimlik Doğrulama: JWT (JSON Web Token) kullanılarak güvenli oturum yönetimi sağlanmıştır. Kullanıcı şifreleri BCrypt ile hashlenerek saklanmıştır.
  - Veri Erişimi: Entity Framework Core (Code-First) kullanılarak PostgreSQL veritabanı ile iletişim kurulmuştur.
  - Orkestrasyon: Mobil uygulamadan gelen ses dosyalarını alır, Python AI servisine yönlendirir ve dönen cevabı işleyerek mobile iletir.
2. AI Mikroservisi (Python / FastAPI):
  - Sistemin "beyni" olarak görev yapar. FastAPI framework'ü ile yüksek performanslı asenkron endpoint'ler oluşturulmuştur.
  - Sessizlik Kontrolü: Gelen ses dosyasının desibel (dB) seviyesi ölçülerek boş veya gürültülü kayıtların işlenmesi engellenmiş, kaynak tasarrufu sağlanmıştır.
  - STT & TTS: ElevenLabs API entegrasyonu ile ses-metin ve metin-ses dönüşümleri yüksek kalitede gerçekleştirilmiştir.
  - LLM (Büyük Dil Modeli): Ollama üzerinden yerel olarak çalıştırılan Llama 3.2 modeli kullanılarak, internet bağımlılığı azaltılmış ve hızlı yanıt süresi hedeflenmiştir.
3. Mobil Uygulama (React Native / Expo):
  - Kullanıcı arayüzü Expo framework'ü ile geliştirilmiştir.
  - Ses Yönetimi: expo-av ve expo-file-system kütüphaneleri ile ses kaydı alma ve sunucudan gelen sesi oynatma işlemleri yapılmıştır.
  - Wake-Word Mantığı: Uygulama içerisinde "Jarvis" veya "Knk" gibi anahtar kelimeleri algılayan algoritmalar geliştirilmiş, arka planda çalışma yeteneği (staysActiveInBackground) kazandırılmıştır.

### 3. PROJE YÖNETİMİ

#### 3.1. Takım Üyeleri Tanımı

Üye No	Numara	Adı Soyadı	Görevi
1	220260020	Muhammed salih üstün	Projeyi tek yaptım tüm görevler bana aittir.

#### 3.2. Görev Dağılımı

İP No	İş Paketinin Adı ve Hedefleri	Görevlendirilen Takım Üyesi
1	<b>Backend Mimarisi:</b> C# Web API kurulumu, JWT Auth, Katmanlı Mimari ve PostgreSQL veritabanı tasarımı.	Muhammed Salih Üstün
2	<b>AI Servis Geliştirme:</b> Python FastAPI kurulumu, ElevenLabs ve Ollama entegrasyonları, ses işleme algoritmaları.	Muhammed Salih Üstün
3	<b>Mobil Uygulama Tasarımı:</b> React Native arayüz kodlaması, animasyonlar ve kullanıcı deneyimi (UX) tasarımı.	Muhammed Salih Üstün
4	<b>Entegrasyon:</b> Mobil, C# ve Python servislerinin birbirine bağlanması ve veri akışının sağlanması.	Muhammed Salih Üstün
5	<b>Test ve Optimizasyon:</b> Sistem hatalarının giderilmesi, ses gecikmelerinin düşürülmesi ve Android Build (APK) işlemleri.	Muhammed Salih Üstün
6	<b>Dokümantasyon:</b> Proje raporlarının hazırlanması ve kodların dokümante edilmesi.	Muhammed Salih Üstün

#### 4. TEORİ VE SONUÇLAR

Proje sonucunda, teorik olarak kurgulanan hibrit mikroservis mimarisi başarıyla hayata geçirilmiştir. Yapılan testlerde şu sonuçlar elde edilmiştir:

1. **Entegrasyon Başarısı:** Farklı dillerde (C# ve Python) yazılan servislerin HTTP protokolü üzerinden sorunsuz haberleştiği ve veri bütünlüğünün korunduğu görülmüştür.
2. **Yapay Zeka Performansı:** Yerel olarak çalışan Llama 3.2 modelinin, M-serisi işlemciler üzerinde kabul edilebilir hızlarda (düşük gecikme) yanıt ürettiği ve Türkçe dil desteğinin yeterli seviyede olduğu gözlemlenmiştir.
3. **Ses Kalitesi:** ElevenLabs entegrasyonu sayesinde, robotik olmayan, vurgu ve tonlamaları doğru yapan bir asistan sesi elde edilmiştir.
4. **Mobil Deneyim:** Uygulama, Android cihaz üzerinde derlenmiş (APK) ve test edilmiştir. Arka planda çalışma ve sürekli dinleme özellikleri, cihaz kaynaklarını verimli kullanarak çalışmaktadır. "Bas-Konuş" ve "Otomatik Dinleme" modları kullanıcıya esneklik sağlamıştır.

Sonuç olarak; geliştirilen "Sesle Uyanan Kişisel Asistan", modüler yapısı sayesinde gelecekte yeni yetenekler (takvim erişimi, ev otomasyonu vb.) eklenmeye açık, kararlı ve modern bir ürün haline gelmiştir.

#### 5. PROJE DOSYALARININ BULUNDUĞU LİNK(DRIVE, GITHUP VB.)

<https://github.com/SalihUstun/VoiceAssistant>

#### KAYNAKLAR

- **Microsoft.** (n.d.). *ASP.NET Core Documentation*. Erişim adresi: <https://learn.microsoft.com/en-us/aspnet/core/>
- **Microsoft.** (n.d.). *Entity Framework Core Documentation*. Erişim adresi: <https://learn.microsoft.com/en-us/ef/core/>
- **Meta AI.** (2024). *Llama 3.2 Model Card*. Erişim adresi: [https://github.com/meta-llama/llama-models/blob/main/models/llama3\\_2/MODEL\\_CARD.md](https://github.com/meta-llama/llama-models/blob/main/models/llama3_2/MODEL_CARD.md)
- **Ollama.** (n.d.). *Ollama Documentation*. Erişim adresi: <https://github.com/ollama/ollama/tree/main/docs>
- **React Native.** (n.d.). *Expo Documentation*. Erişim adresi: <https://docs.expo.dev/>
- **React Navigation.** (n.d.). *React Navigation Documentation*. Erişim adresi: <https://reactnavigation.org/docs/getting-started>
- **ElevenLabs.** (n.d.). *API Reference*. Erişim adresi: <https://elevenlabs.io/docs/api-reference>
- **ElevenLabs.** (n.d.). *Python SDK*. Erişim adresi: <https://github.com/elevenlabs/elevenlabs-python>
- **FastAPI.** (n.d.). *FastAPI Documentation*. Erişim adresi: <https://fastapi.tiangolo.com/>
- **PostgreSQL.** (n.d.). *PostgreSQL Documentation*. Erişim adresi: <https://www.postgresql.org/docs/>