

**FIRAT ÜNİVERSİTESİ**

TEKNOLOJİ FAKÜLTESİ

Yazılım Mühendisliği Bölümü

**YMH314 - FONKSİYONEL PROGRAMLAMA**

**YMH310 - WEB TASARIMI VE PROGRAMLAMA**

Dersi Proje Ön Raporu

**Yapay Zeka Destekli Hava Durumu Verilerine Göre Kişisel Kombin Önerileri**

**Geliştiren**

215541059 - Mehmet Özerli

**Proje Yürütücüleri**

Öğr.Gör. Eyüp ERÖZ

Dr. Öğr. Üyesi Vahdettin Cem BAYDOĞAN

**ŞUBAT – 2025**

İÇİNDEKİLER

[1. Proje Özeti 3](#_Toc191892627)

[2. Problem Tanımı 3](#_Toc191892628)

[3. Çözüm Önerisi 3](#_Toc191892629)

[**3.1 Temel Özellikler** 3](#_Toc191892630)

[**3.2 Teknik Altyapı** 4](#_Toc191892631)

[4. Fonksiyonel Programlama Uygulamaları 4](#_Toc191892632)

[5. Kullanıcı Senaryoları 4](#_Toc191892633)

[**Senaryo 1: Becayiş Eşleşmesi** 4](#_Toc191892634)

[**Senaryo 2: Mevzuat Bilgisi Alma** 5](#_Toc191892635)

[6. İş Planı Ve Zaman Çizelgesi 5](#_Toc191892636)

[7. Teknik Gereksinimler 5](#_Toc191892637)

[8. Beklenen Faydalar 5](#_Toc191892638)

[9. Riskler Ve Zorluklar 6](#_Toc191892639)

[10. Kullanılması Planlanan Kaynaklar 6](#_Toc191892640)

# 1. Proje Özeti

Bu proje, yapay zeka destekli bir web sitesi ve mobil uygulama olarak tasarlanmıştır. Kullanıcıların hava durumuna uygun kombinler oluşturmasına yardımcı olacak bir sistem geliştirilmesi hedeflenmektedir. Sistem, hava durumu verilerini bir API aracılığıyla çekecek ve Pinterest'ten alınan kombin fikirleriyle kullanıcılara öneriler sunacaktır. Ayrıca kullanıcılar, kendi gardıroplarındaki kıyafetleri fotoğraflayarak favori kombinlerini oluşturabileceklerdir.

# 2. Problem Tanımı

Günlük giyim seçimleri, özellikle hava şartları göz önünde bulundurulduğunda zaman alıcı ve zahmetli olabilir. Mevcut stil öneri platformları genellikle statik olup kişisel dolap entegrasyonuna izin vermez. Bu proje, hava durumuna dayalı kişiselleştirilmiş kombin önerileri sunarak kullanıcıların giyim kararlarını kolaylaştırmayı amaçlamaktadır.

# 3. Çözüm Önerisi

**3.1 Temel Özellikler**

* Hava durumu API entegrasyonu ile anlık hava bilgisi alma.
* Pinterest API kullanarak kombin önerileri sunma.
* Kullanıcının kendi kıyafetlerini yükleyip favori kombinlerini oluşturmasını sağlama.
* Web platformu için JavaScript, mobil uygulama için React Native teknolojilerinin kullanımı.

**3.2 Teknik Altyapı**

* **Frontend Teknolojileri:** Web uygulaması için **React.js**, mobil uygulama için **React Native** kullanılacaktır. Responsive tasarım için **Tailwind CSS** ve **Styled Components** gibi modern UI araçları tercih edilecektir.
* **Backend Teknolojileri:** Firebase **Firestore** veritabanı kullanılacak olup, kullanıcı kimlik doğrulama işlemleri **Firebase Authentication** ile sağlanacaktır. Sunucu tarafı işlemleri için **Node.js ve Express.js** kullanılabilir.
* **Veri Kaynakları ve API Entegrasyonu:**
  + **Hava Durumu API:** Güncel hava durumu verilerini almak için **OpenWeatherMap API** veya **WeatherAPI** kullanılacaktır.
  + **Pinterest API:** Kombin önerileri için **Pinterest API** kullanılacak ve öneriler filtrelenerek sunulacaktır.
  + **Görsel İşleme:** Kullanıcıların kendi dolabındaki kıyafetleri sisteme yükleyebilmesi için **Google Cloud Vision API** veya benzeri bir teknoloji entegrasyonu düşünülebilir.
* **Depolama ve Dosya Yönetimi:** Kullanıcıların yüklediği kıyafet fotoğraflarını saklamak için **Firebase Storage** kullanılacaktır.
* **Yapay Zeka ve Öneri Motoru:** Kullanıcının hava durumu ve önceki tercihlerine göre kişiselleştirilmiş öneriler sunabilmesi için **TensorFlow.js** veya **Scikit-learn** tabanlı bir model eğitilebilir.
* **Gerçek Zamanlı Bildirimler:** Kullanıcıya hava durumu değişiklikleri ve yeni öneriler hakkında bilgilendirme göndermek için **Firebase Cloud Messaging (FCM)** kullanılacaktır.

# 4. Fonksiyonel Programlama Uygulamaları

Fonksiyonel programlama, yazılım geliştirme sürecinde kodun daha okunaklı, hata ayıklanabilir ve test edilebilir olmasını sağlar. Bu projede fonksiyonel programlama prensipleri aşağıdaki şekillerde uygulanacaktır:

* **Saf Fonksiyonlar (Pure Functions):** Fonksiyonlar, dış etkenlerden bağımsız olarak sadece girdiye bağlı çıktılar üretecektir. Örneğin, hava durumu verisine göre kombin önerisi sunan fonksiyon, herhangi bir yan etki oluşturmadan sadece verilen girdiye göre sonuç döndürecektir.
* **Immutable Veri Yapıları:** State yönetimi sırasında değişmez (immutable) veri yapıları tercih edilecektir. Bu sayede veri mutasyonu önlenecek ve hata yönetimi daha kolay olacaktır. **Redux** veya **Recoil** gibi state yönetim kütüphaneleri kullanılabilir.
* **Fonksiyonların Birleştirilebilirliği (Function Composition):** Küçük ve bağımsız fonksiyonlar birleştirilerek kompleks işlemler gerçekleştirilecektir. Örneğin, hava durumu verisini alıp analiz eden ve kıyafet önerisi sunan iki ayrı fonksiyon birbirine bağlanarak çalışacaktır.
* **Yüksek Mertebeden Fonksiyonlar (Higher-Order Functions):** Map, filter, reduce gibi fonksiyonlar kullanılarak veri manipülasyonu sağlanacaktır. Örneğin, kullanıcı dolabındaki kıyafetleri filtreleyerek hava durumuna uygun olanları listeleyen bir fonksiyon geliştirilecektir.
* **Yan Etkilerin Minimize Edilmesi:** Asenkron işlemler, özellikle API çağrıları, merkezi bir yönetim mekanizmasıyla (örneğin Redux middleware veya React Query) kontrol edilecek ve yan etkiler minimize edilecektir

# 5. Kullanıcı Senaryoları

***Senaryo 1:*** *Becayiş Eşleşmesi* Kullanıcı, hava durumu verilerine dayalı olarak kendisine en uygun kombin önerilerini alır. Eğer kendi dolabındaki kıyafetleri de sisteme yüklediyse, önerilen kombinler arasına kendi kıyafetlerinden oluşan alternatifler de eklenir.

***Senaryo 2:*** *Mevzuat Bilgisi Alma* Hava durumu güncellemelerine dayalı olarak belirli etkinlikler için uygun kombinler önerilir. Örneğin, yağmurlu bir gün için su geçirmez ayakkabı ve şemsiye önerilir.

# 6. İş Planı ve Zaman Çizelgesi

|  |  |
| --- | --- |
| Proje Planlama ve Analiz | 2 hafta |
| Veritabanı Tasarımı | 1 hafta |
| Backend Geliştirme | 3 hafta |
| Frontend Geliştirme (Web ve Mobil) | 4 hafta |
| Güvenlik Yapılandırmaları | 1 hafta |
| Test ve Hata Ayıklama | 2 hafta |
| Proje Tamamlama ve Sunum | 1 hafta |

# 7. Teknik Gereksinimler

* Firebase veritabanı ve kimlik doğrulama
* Pinterest API entegrasyonu
* Hava durumu API entegrasyonu
* React Native ve JavaScript ile geliştirme

# 8. Beklenen Faydalar

* Kullanıcılara günlük kombin seçiminde kolaylık sağlar.
* Hava durumu faktörünü göz önünde bulundurarak kişiselleştirilmiş öneriler sunar.
* Kullanıcının kendi dolabını dijital ortamda yönetmesine olanak tanır.

# 9. Riskler ve Zorluklar

* Pinterest API erişimi ve kota sınırlamaları
* Kullanıcıların kıyafet fotoğraflarını düzenlemesi ve sisteme ekleme süreci
* Hava durumu API’sinin doğru ve güncel veri sağlaması

# 10. Kullanılması Planlanan Kaynaklar

* Firebase (veritabanı ve kimlik doğrulama)
* Pinterest API
* Hava Durumu API
* JavaScript ve React Native
* TensorFlow.js veya Scikit-learn (öneri motoru için)
* Google Cloud Vision API (görsel analiz için)