

**Grup 78 VERİTABANI PROJESİ RAPORU**

Fatih Bilgin  
Bilişim Sistemleri Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli , Türkiye  
231307019@kocaeli.edu.tr

Furkan Demirci  
Bilişim Sistemleri Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli , Türkiye  
231307061@kocaeli.edu.tr

Yekta Cengiz  
Bilişim Sistemleri Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli , Türkiye  
231307080@kocaeli.edu.tr

*Bu çalışma, Grup 78 tarafından geliştirilen bir uçuş yönetim sisteminin tasarımı ve uygulamasını sunmaktadır. Hava trafiğinin artmasıyla birlikte doğru ve gerçek zamanlı uçuş bilgilerine erişim büyük önem kazanmıştır. Geliştirilen sistem; kullanıcıların uçuş sorgulayabileceği, yöneticilerin ise uçuş, uçak, pist ve havalimanı bilgilerini yönetebileceği kullanıcı dostu bir arayüz sunmaktadır. Sistem; frontend tarafında React.js, backend tarafında Node.js ve veritabanı olarak MySQL kullanılarak geliştirilmiştir. Rol seçimi, kullanıcı ile yönetici görünümünü arasında kolay geçiş sağlayan menü aracılığıyla sağlanmıştır. Bu platform, havacılık sistemlerinde uçuş bilgilerine erişimi ve yönetimini kolaylaştırmayı hedeflemektedir.*

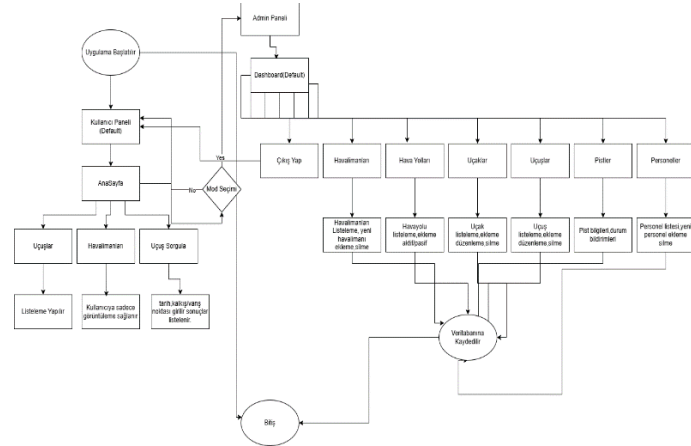
## I. INTRODUCTION (*GIRIS*)

Günümüzde hava yolu taşımacılığı önemli ölçüde artış göstermektedir. Bu artış, uçuşlara ait bilgilere hızlı ve düzenli erişim ihtiyacını da beraberinde getirmiştir. Mevcut sistemlerin birçoğu kullanıcı dostu olmayan arayüzlere ve yetersiz veri güncellemelerine sahiptir. Bu doğrultuda geliştirilen bu proje, kullanıcıların uçuş bilgilerini kolayca sorgulayabileceği; yöneticilerin ise uçuş, pist, havalimanı ve uçak bilgilerini güncelleyebileceği bir platform sunmayı amaçlamaktadır. Sistem; React.js tabanlı modern bir arayüz, Node.js ile kurulmuş bir sunucu altyapısı ve MySQL veritabanı ile desteklenmektedir. Kullanıcı ve yönetici rolleri, sayfanın üst kısmında bulunan bir dropdown menü aracılığıyla ayrılmıştır. Bu sayede hem kullanıcı hem de yönetici tarafı için sade, hızlı ve güvenilir bir kullanım deneyimi sağlanmaktadır. Bu raporda, geliştirilen sistemin araştırma süreci, yazılım mimarisi, veritabanı yapısı ve uygulama detayları açıklanacaktır.

## II. RESEARCH CONDUCTED(YAPILAN ARAŞTIRMALAR)

1) Proje geliştirme sürecinde aşağıdaki konular üzerine detaylı araştırmalar yapılmıştır. **Veritabanı Tasarımı** : Tablolar arası ilişkiler,normalizasyon kuralları ve ER diyagramları incelenmiştir. **Rol Seçimi** : Kullanıcı ve yönetici rolleri için görünüm bazlı yetki kontrolü uygulanmıştır. **Uçuş Filtreleme** : MySQL üzerinde tarih ve havalimanına göre filtreleme sorguları optimize edilmiştir. **Backend-Frontend Bağlantısı** : Node.js ile kurulan backend, React.js frontend ile JSON formatında veri alışverişi yapacak şekilde yapılandırılmıştır. Frontend ile Backend arasında veri alışverişi sağlanmış,istekler başarıyla işlenmiş ve sistemin bütünlüğü korunmuştur.

### III. FLOW CHART (AKIŞ ŞEMASI)



1) Uygulama açıldığında kullanıcı paneli varsayılan olarak gelir. Ana sayfada uçuşlar, havalimanları ve uçuş sorgulama bölümleri bulunur. Kullanıcı sadece verileri görüntüler ve uçuş sorgulaması yapabilir.

2) Admin paneline sağ üstten geçiş yapılır. Admin tarafında dashboard ile başlanır, sistemin genel durumu gösterilir. Admin, havalimanı, havayolu, uçak, uçuş, pist ve personel işlemlerini (ekleme, silme, güncelleme) yapabilir.

3) Tüm bu işlemler veritabanına kaydedilir ve sistem genelinde güncellenir.

#### IV. SOFTWARE ARCHITECTURE(YAZILIM MIMARISI)

1) Bu projede **çok katmanlı (multi-layered)** mimari yaklaşımı benimsenmiştir. Sistem, genel olarak **kullanıcı paneli** ve **admin paneli** olmak üzere iki ayrı arayüzle çalışmaktadır. Uygulama açıldığında varsayılan olarak kullanıcı paneli aktif olur, admin paneline ise sağ üst köşedeki geçiş ile erişilir.

2) **Frontend (İstemci Tarafı):** *React.js* kullanılarak oluşturulmuş kullanıcı dostu arayüzler sayesinde hem kullanıcılar hem de adminler ihtiyaç duydukları bilgilere hızlıca ulaşabilir. Sayfalar bileşen tabanlı yapıya sahiptir ve routing yapısı ile desteklenmiştir.

3) **Backend (Sunucu Tarafı):** Node.js kullanılarak RESTful API yapısıyla geliştirilmiştir. Tüm veri işlemleri (CRUD işlemleri) API uç noktaları üzerinden yönetilir.

4) **Veritabanı:**  
MySQL kullanılarak veriler ilişkisel olarak yönetilir. Tablolar arasında güçlü ilişkiler kurulmuş, veri bütünlüğü sağlanmıştır.

5) **Katmanlar:**

**Kullanıcı Arayüzü Katmanı:** React ile yazılmış olup, kullanıcı ve admin işlemleri için ayrı ekranlar barındırır.

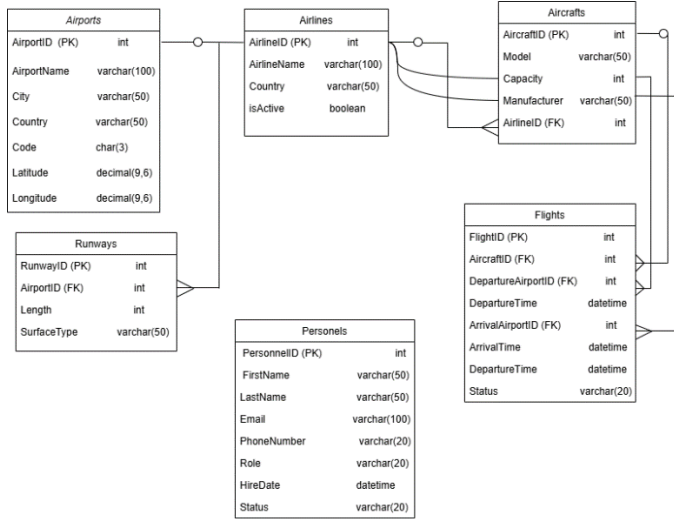
**İş Mantığı (Business Logic) Katmanı:** Kullanıcı ve admin işlemlerini yöneten, uçuş yönetimi, havalimanı, pist gibi modüllerin kontrol edildiği kısımdır.

**Veri Erişim Katmanı:** Veritabanı bağlantılarını, SQL sorgularını ve CRUD işlemlerini içerir.

#### 6) Yetkilendirme & Erişim:

Kullanıcı sadece verileri görüntüleyebilirken; admin tarafı, sistemdeki tüm verileri yönetme yetkisine sahiptir.

### V. DATABASE DIAGRAM(ER DIAGRAM)



1) Bu projede geliştirilen **uçuş yönetim sistemi**, ilişkisel bir veritabanı yapısı üzerine kurulmuştur. Sistemdeki temel varlıklar (entity'ler) ve bu varlıklar arasındaki ilişkiler **ER (Entity Relationship) Diyagramı** ile modellenmiştir.

#### 2) Temel Varlıklar:

3) **Uçuşlar (Flights):** Her uçuş, kalkış ve varış noktaları, tarih, saat, uçak ve pist bilgisiyle ilişkilendirilmiştir.

4) **Havalimanları (Airports):** Uçuşların kalkış ve varış yaptığı yerleri temsil eder. Her havalimanı birden fazla uçuşla ilişkilidir.

5) **Hava Yolları (Airlines):** Uçakların bağlı olduğu firmaları temsil eder. Her hava yolu birçok uçağa sahip olabilir.

6) **Uçaklar (Planes):** Uçuşları gerçekleştiren araçlardır. Her uçak bir hava yoluna bağlıdır.

7) **Pistler (Runways):** Havalimanlarına ait pistlerdir. Uçuşlar bir pistle eşleştirilir.

8) **Personeller (Staff):** Sisteme kayıtlı olan ve uçuşları yöneten çalışanlardır. Admin tarafından eklenip yönetilir.

#### 9) İlişkiler:

10) Bir hava yolu, birden çok uçağa sahiptir ( $1 \rightarrow N$ ).

11) Bir uçuş, bir uçak, bir pist, bir kalkış ve varış havalimanı ile ilişkilidir ( $N \rightarrow 1$ ).

12) **Uçuşlar**, zaman, durum gibi bilgileriyle birlikte sistemde listelenebilir ve güncellenebilir.

13) Bu diyagram, veritabanı tasarımının mantıksal temelini oluşturur ve tüm CRUD işlemlerinin doğru ve tutarlı şekilde yapılmasını sağlar.

### VI. GENERAL STRUCTURE(GENEL YAPI)

1) Bu proje, **uçuş yönetim sistemi** olarak tasarlanmış bir web tabanlı uygulamadır. Kullanıcılar ve yöneticiler (admin) için farklı yetkilere sahip paneller sunar. Sistem, modern web teknolojileri kullanılarak geliştirilmiş olup, veritabanı destekli dinamik bir yapıya sahiptir.

2) **Kullanıcı Paneli (Varsayılan Ekran):** Uygulama başlatıldığında kullanıcı paneli varsayılan olarak açılır. Kullanıcılar burada:

3) **Uçuş listesini** görüntüleyebilir,

4) **Havalimanı bilgilerine** ulaşabilir,

5) Belirli kriterlere göre **uçuş sorgulama** işlemi gerçekleştirebilir.

6) Bu panelde yalnızca veri **görüntüleme** işlemleri yapılabilir. Kullanıcılar herhangi bir veriyi düzenleyemez veya silemez.

#### 7) Admin

#### Paneli:

Kullanıcı ekranının sağ üst köşesinde bulunan seçim menüsünden **admin paneline** geçiş yapılabilir. Admin panelinde:

8) **Dashboard** üzerinden güncel verilere genel bakış sağlanır (aktif uçuşlar, iptaller vs.),

9) **Havalimanları, havayolları, uçaklar, uçuşlar, pistler ve personeller** gibi bölümler yer alır,

10) Bu bölümlerde veri **ekleme, silme, düzenleme** işlemleri yapılabilir.

#### 11) Veritabanı

#### Entegrasyonu:

Sistemdeki tüm işlemler MySQL veritabanı ile entegredir. Gerek kullanıcı sorguları, gerekse admin işlemleri veritabanı ile senkronize bir şekilde çalışır. Böylece sistemde yapılan her değişiklik anlık olarak kaydedilir ve güncel verilerle çalışılır.

### VII. REFERENCES(REFERANSLAR)

1) phpMyAdmin, "phpMyAdmin Official Documentation," [Online]. Available: <https://www.phpmyadmin.net/>. Accessed: May 2025.

2) W3Schools, "SQL Tutorial," [Online]. Available: <https://www.w3schools.com/sql/>. Accessed: May 2025.

3) OpenAI, "ChatGPT by OpenAI," [Online]. Available: <https://chat.openai.com>. Accessed: May 2025.

4) DeepSeek, "DeepSeek Coder," [Online]. Available: <https://www.deepseek.com/>. Accessed: May 2025.

5) GitHub, "GitHub - Code Repositories," [Online]. Available: <https://github.com/>. Accessed: May 2025.

6) Mozilla Developer Network (MDN), "Web Docs - JavaScript, HTML, CSS," [Online]. Available: <https://developer.mozilla.org/>. Accessed: May 2025.

7) Stack Overflow, "Stack Overflow - Developer Community," [Online]. Available: <https://stackoverflow.com/>. Accessed: May 2025.