**T.C. SAKARYA ÜNİVERSİTESİ BİLGİSAYAR**

**VE BİLİŞİM BİLİMLERİ FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Ders: Web Programlama**

**Dönem: 2023 – 2024 Güz Dönemi**

**Öğrenci Bilgileri**

Adı:

Soyadı:

No:

Grup:

Proje Adı:

Proje: Web Programlama dersinde teorik ve pratik olarak öğrenilen bilgilerin, gerçek bir Probleme uygulanarak bu probleme çözüm üreten bir web projesi geliştirilmesidir.

Proje Tarih: 30.11.2023 – 31.12.2023

Projede Kullanılan Teknolojiler:

* Asp.NET Core 6
* C#
* MSSQL
* Entity Framework
* Autofac
* FluentValidation
* Microsoft.AspNetCore.Authentication.JwtBearer
* Newtonsoft.Json
* Swashbuckle.AspNetCore
* Angular
* Boostrap
* Ngx-toastr
* Primeicons
* Primeng
* NPM

**Github:**

**Genel Bakış**

Bu rapor, ASP.NET Core MVC kullanılarak oluşturulan, uçuş yönetimi ve kullanıcı rezervasyonlarına odaklanan bir web uygulamasının geliştirme sürecini ve mimarisini özetlemektedir.

A screenshot of a computer program

Description automatically generated

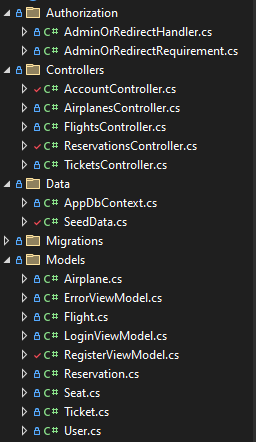
**Proje Yapısı**

**1. Arka Uç Mimarisi:**

MVC Mimarisi: Endişelerin net bir şekilde ayrılması için Model-View-Controller kalıbı kullanıldı.

Entity Framework Çekirdeği: Veritabanı etkileşimlerini sorunsuz bir şekilde yöneten ORM için benimsenmiştir.

Kimlik Yönetimi: Kullanıcı kimlik doğrulama ve yetkilendirme için ASP.NET Core Identity kullanılarak uygulandı.



**Temel Özellikler:**

Uçuş Yönetimi: Uçuş detaylarının oluşturulmasına, değiştirilmesine ve görüntülenmesine izin verir.

Rezervasyon Sistemi: Kullanıcılar rezervasyon yapabilir ve yönetebilir.

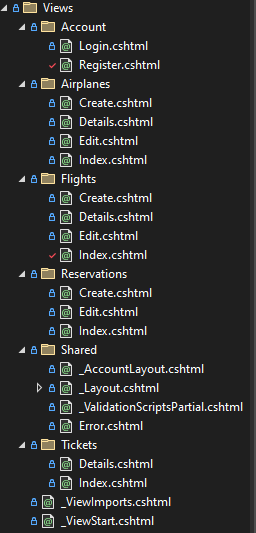
Rol Tabanlı Erişim Kontrolü: Yönetici ve normal kullanıcılar için farklılaştırılmış erişim.

**2. Ön Uç:**

Razor Görünümleri: Dinamik HTML içerik oluşturma için kullanılır.

Bootstrap: Duyarlı kullanıcı arayüzü tasarımı için kullanılır.

İstemci Tarafı Mantığı: Kullanıcı etkileşimlerini geliştirmek için minimal JavaScript ve jQuery.



**3. Veritabanı Tasarımı:**

SQL Veritabanı: Uçuş, kullanıcı ve rezervasyon verilerini depolamak için kullanılır.

Kod Öncelikli Yaklaşım: C# varlık modelleri aracılığıyla tanımlanan veritabanı şeması.

**4. Güvenlik:**

Kimlik Doğrulama ve Yetkilendirme: Güvenli kullanıcı girişi ve rol tabanlı sayfa erişimi.

Veri Doğrulama: Veri bütünlüğü için sunucu tarafı ve istemci tarafı doğrulama.

Geliştirme Süreci

**1. Kurulum ve Yapılandırma:**

ASP.NET Core Kurulumu: Startup sınıfı ve middleware dahil olmak üzere temel yapı yapılandırıldı.

Bağımlılık Enjeksiyonu: Servisleri ve depoları yönetmek için kullanıldı.

**2. Kullanıcı Arayüzü Tasarımı:**

Razor Düzenleri: Tutarlı kullanıcı arayüzü için paylaşılan düzenler oluşturuldu.

Formlar ve Görünümler: Kullanıcı kaydı, giriş, uçuş rezervasyonu gibi çeşitli işlevler için geliştirildi.

**3. İş Mantığı:**

Kontrolörler: Kullanıcı isteklerini işlemek, modellerle etkileşim kurmak ve görünümleri döndürmek için geliştirilmiştir.

Model Doğrulama: Zorunlu iş kuralları ve veri bütünlüğü.

**4. Veritabanı Entegrasyonu:**

Veri Modelleri: Uçuş, Rezervasyon, Kullanıcı gibi tanımlanmış varlıklar.

Depo Modeli: Veri erişim katmanını soyutlamak için benimsenmiştir.

**5. Test ve Hata Ayıklama:**

Manuel Test: Uygulamanın işlevsel testi için yürütülür.

Hata İşleme: İstisnaları incelikle yönetmek için uygulandı.

Zorluklar ve Çözümler

Kullanıcı Rol Yönetimi: Rol tabanlı görünümler ve eylemler uygulandı.

Duyarlı Tasarım: Uygulamanın mobil uyumlu olmasını sağlamak için Bootstrap kullanıldı.

**Sonuç**

Proje, uçuş yönetimi ve kullanıcı rezervasyonları için kapsamlı bir web uygulaması oluşturmada ASP.NET Core MVC'nin yeteneklerini etkili bir şekilde göstermektedir. Uygulama kullanıcı dostu bir arayüz, güvenli oturum açma mekanizmaları ve sağlam arka uç işlemleri sağlamaktadır. Modern geliştirme uygulamalarının ve araçlarının kullanılması, ölçeklenebilir ve bakımı yapılabilir bir uygulamanın oluşturulmasını sağlamıştır.