

## Odeon Airlines Case Study

Bu örnek çalışma, adayların hem backend hem de frontend yeteneklerini test etmek amacıyla hazırlanmıştır. İstenilen özellikler aşağıdaki gibidir:

### Genel İsterler:

- Sistemde iki farklı rol olacaktır: **ADMIN** ve **USER**.
- **ADMIN** rolündeki kullanıcı, uçuş planı oluşturma yetkisine sahip olacak.
- **USER** rolündeki kullanıcılar, sisteme kayıt olduktan veya giriş yaptıktan sonra planlanmış uçuşları görüntüleyebilecektir.

### Frontend İsterleri:

#### 1. Komponent Oluşturulması:

- Aşağıda belirtilen temel component'ler oluşturulmalıdır:
  - **INPUT**
  - **SELECT**
  - **BUTTON**
  - **CHECKBOX**
  - **DATEPICKER**
- Bu component'ler, "base component" klasöründe yer almalı ve **state** tutmamalıdır. Gerekli veriler **props** aracılığıyla iletilecektir.

#### 2. Kullanıcı İşlemleri:

- **USER** rolündeki kullanıcılar için:
  - **Kayıt olma (Signup) süreci:** Kullanıcılar, isim, soyisim, bulunduğu şehir, kullanıcı adı (username) ve şifre (password) bilgileri ile kayıt olabilmelidir.
  - **Giriş yapma (Login) süreci:** Kullanıcılar, kullanıcı adı ve şifre ile giriş yapabilmelidir.
  - Giriş yaptıktan sonra kullanıcılar **anasayfa**ya yönlendirilir.

#### 3. Anasayfa:

- Kullanıcı anasayfası iki senaryoya göre çalışacaktır:
  1. Kullanıcının bulunduğu şehir bilgisi sisteme kaydedildiyse, bu şehre göre **kalkış** yapılan uçuşlar listelenecektir.
  2. Eğer kullanıcı uçuş araması yaparsa, şu kriterlere göre uçuşlar listelenmelidir:
    - **Kalkış yeri**
    - **Varış yeri**
    - **Uçuş günü**

#### 4. ADMIN Sayfası:

- **ADMIN** rolü sistemde varsayılan olarak tanımlı olacaktır.
- ADMIN kullanıcıları, sisteme giriş yaptıktan sonra mevcut uçuş planlarını görüntüleyebilir ve yeni uçuş planları oluşturabilir.
- Uçuş planlaması yapılırken dikkate alınması gereken kurallar:
  - Bir uçuşun iniş veya kalkış yaptığı hava sahasında, uçak 30 dakika boyunca başka bir iniş veya kalkış yapamaz.

- o Aynı şehirden yapılan uçuşların 30 dakikalık bu süre içinde birbirleriyle çakışmaması gerekir.
- **Örnek:**
  - o **ANKARA - X uçuşu** saat 14:00'te kalkıp 18:00'de inecektir. Bu durumda ANKARA'da 14:00-14:30 arasında başka bir uçuş yapılamaz (ne kalkış ne de iniş).

### **Teknoloji Seçimleri:**

- **Frontend** için **ReactJS** kullanılacaktır.
- **Backend** için **Java Spring Boot** tercih edilmelidir.
- Veritabanı tercihi adaya bırakılmıştır (örneğin: H2, MySQL, MongoDB).

### **Ekstra İsterler:**

- **Frontend** tarafında **API çağrıları** için **Axios** kullanılmalıdır.
- **Backend** tarafında **REST** mimarisine uygun şekilde API'ler tasarlanmalıdır.