

## Case Study: Flight Search API *(Backend Developer)*

### Açıklama

Bir uçuş arama uygulaması için backend API geliştirilecek.

### Beklentiler:

- **Data Modeling:** Veritabanında saklanacak bilgilerin tasarlanması ve modellenmesi gerekiyor.
  - Veritabanı olarak herhangi bir relational (SQL) ya da NoSQL database kullanılabilir.
  - Saklanacak bilgiler:
    - Uçuşlar
      - ID
      - Kalkış havaalanı
      - Varış havaalanı
      - Kalkış tarih/saat
      - Dönüş tarih/saat
      - Fiyat
    - Havaalanları
      - ID
      - Şehir
- **CRUD yapısı kurgulanmalı:** CRUD (Create, Read, Update, Delete) yapısı, bir veritabanı üzerinde en yaygın olarak gerçekleştirilen temel işlemleri temsil eder.
  - Bu yapının kurgulanması, verilerin tutarlı ve organize bir şekilde yönetilmesini sağlar. Bu sayede kullanıcılar gerektiği gibi veri oluşturabilir, okuyabilir, güncelleyebilir ve silebilir.
  - CRUD yapısının uygulanacağı kaynaklar.
    - Uçuşlar
    - Havaalanları
- **Search API**
  - Verilen kalkış yeri, varış yeri, kalkış tarihi ve dönüş tarihine uygun uçuşları listeleyen bir API endpoint yapılmalı.
  - Dönüş tarihi verilmediyse tek yönlü uçuş, verildiyse çift yönlü uçuştur.
  - Tek yönlü uçuş için tek uçuş bilgisi, çift yönlü uçuş için iki uçuş bilgisi verilmeli.

- **REST ile dışarıya servis sunulmalı:** REST mimarisi ile dışarıya servis sunmak, uygulamanın diğer sistemlerle etkileşim kurabilmesini sağlar. Bu, veri alışverişini kolaylaştırır ve genel olarak sistemler arası entegrasyonu mümkün kılar.
- **Java kullanılmalı** (Spring/Spring Boot vb.)
- **Authentication yapısı olmalı:**
  - Authentication yapısı, kullanıcının kimliğini doğrulamak ve yetki vermek için kullanılır. Bu yapı sistemlerin daha güvenli olmasını sağlar. İstenilen authentication mimarisi kullanılabilir.
- **Scheduled background jobları:** Her gün third-party bir API'a istek atarak uçuş bilgilerini alan ve database'e kaydeden bir scheduled job yapılmalı.
  - Gerçek bir third-party API kullanılmayacak. Mock bir API isteği yapıp yapay veri üretilerek çalışılabilir.
- **Git versiyon sistemi kullanılmalı.** Proje GitHub'a yüklenilecek.
- **Dokümantasyon:** Swagger ile API dokümantasyonu yapılmalı.

Case ile ilgili soruların olursa [helloworld@coderspace.io](mailto:helloworld@coderspace.io) 'dan bize ulaşabilirsin.

**Başarılar dileriz!** 💪🚀★