

Enoca Challenge CRUD Project

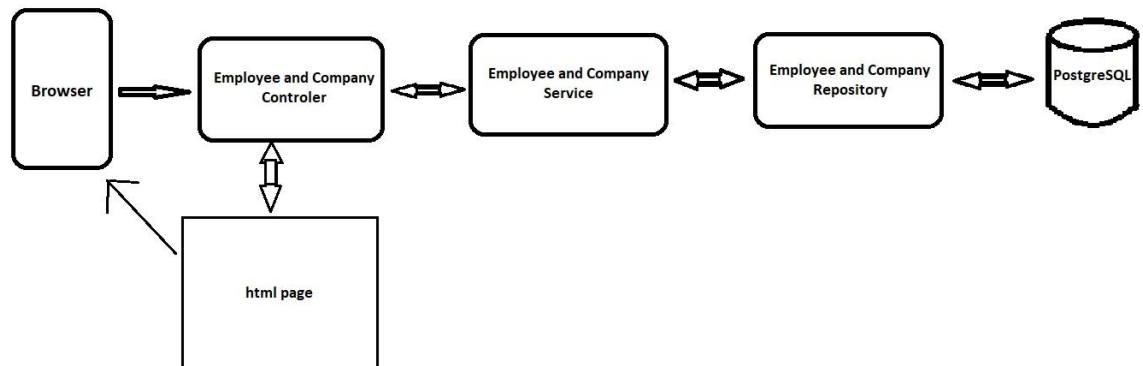
1. Proje Tanımı

Bu proje, şirketleri ve çalışanlarını yönetmek için temel bir CRUD uygulamasıdır. Veri kalıcılığı için Spring Framework'ü, Spring Boot ve Spring Data JPA'yı kullanır. Proje üç ana katman içerir: veri erişim katmanı (repositories), hizmet katmanı (business logic) ve sunum katmanı (controller). Bu projede kullanılan mimari model, Model-View-Controller (MVC) modelidir.

2. Proje Oluşturma Aşamaları

1. Spring initializr web arayüzü ile Spring projemi oluşturmaya başladım. Bu projede bağımlılık olarak Spring Data JPA , Thymeleaf , Spring Web , Spring Boot Devtools , PostgreSQL Driver kullandım. Projem bu bağımlılıkları ekledikten sonra otomatik olarak yapılandırıldı.
2. PostgreSQL'de yeni bir database oluşturdum.
3. IDE olarak IntelliJ IDEA kullandım.
4. Projem açıldığında ilk olarak database bağlantısını oluşturmak için application.properties dosyamda gerekli konfigürasyonu yaptım.
5. Model katmanı- veritabanındaki verileri temsil etmesi için Employee ve Company model sınıflarını oluşturdum. Sınıflarda, database tabloları arasındaki karşılık gelen değişkenleri tanımladım. İki tablo arasındaki bağlantıyı sağlaması için @OneToMany ve @ManyToOne annotationları kullandım.
6. Repository - Database'e erişim işlemleri için EmployeeRepository ve CompanyRepository interface olarak oluşturdum. Bu repositorylerde JpaRepository implement ettim.
7. Servis - CompanyService ve EmployeeService interface'leri kullanılarak uygulanır. Bu interface'lerde varlıklar üzerinde gerçekleştirilebilecek tüm verileri alma, yeni bir veri kaydetme ve mevcut bir veriyi silme gibi yöntemleri tanımladım. Bu interface'lerin uygulanmasını, CompanyServiceImpl ve EmployeeServiceImpl sınıfları tarafından sağladım.
8. Controller – HTTP requestlerine cevap verebilmek için CompanyController ve EmployeeController sınıflarını oluşturdum. Bu kontroller yeni bir veriyi veritabanına kaydetmek, verileri göstermek, verileri silmek için methodlara sahiptir.
9. View – Resource paketinde templates klasöründe index.html, new_company.html, new_employee.html, update_company.html, update_employee.html sayfalarını oluşturdum. Bu sayfalar ile kullanıcıya database işlemlerini gerçekleştirebileceği arayüzü sağladım.

3. Dinamik Model



4. Database Modeli – ER Diyagramı

