

Kullanılan Teknolojiler

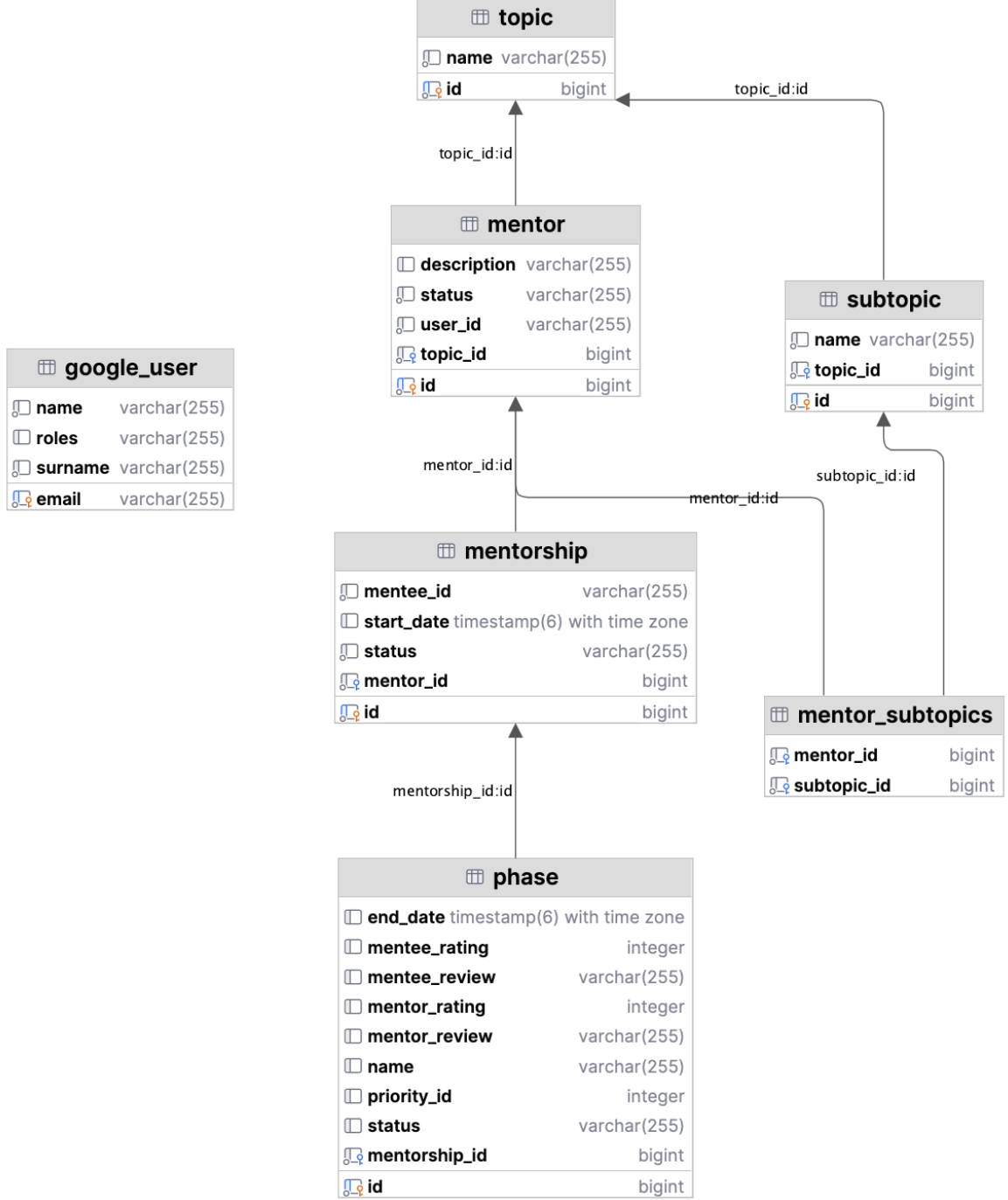
Backend :

- **Spring Boot** : Spring Boot, Java dilinde hem geniş bir ekosisteme sahip olması hem de desteklediği zengin kütüphane çeşitliliğiyle hızlı ve etkili bir geliştirme imkanı sunduğu için seçilmiştir.
- **Spring Security** : Uygulamanın güvenlik katmanı, Spring Security framework'ü ile sağlanmıştır. Kullanıcı doğrulama ve yetkilendirme işlemleri, Embedded LDAP ve Google Auth entegrasyonlarıyla birlikte yönetilmektedir.
- **Spring Boot Starter Mail**: Uygulamanın faz hatırlatıcı e-posta gönderme işlevi için Spring Boot Starter Mail kullanılmıştır. Kullanışlılığı ve hızlı kurulumu sayesinde tercih edilmiştir.
- **PostgreSQL**: İlişkisel bir veritabanı olması ve projede bir çok yerde Mentor-Mentorship arası, Faz-Mentorship arası gibi ilişkilere ihtiyaç duyulmasından ötürü SQL veritabanı tercih edilmiştir.
- **Spring Data JPA**: SQL tarafında sağladığı query metodlarıyla veri tabanı sorgularını hem daha güvenli hale getirdii hem de kolaylaştırdığı için tercih edilmiştir.
- **Docker** : Uygulamanın farklı ortamlarda sorunsuz çalışmasını sağlamak ve geliştirme sürecini kolaylaştırmak amacıyla Docker kullanılmıştır. Docker compose da kullanılarak PostgreSQL, Elasticsearch gibi servisler, uygulama konteyneri ile birlikte yönetilebilir ve hızlı bir şekilde başlatılabildiği için projede kullanılmıştır.
- **Elasticsearch** : Projede gerekli olan free text arama işlevini yerine getirebilmesi için tercih edilmiştir.

Frontend :

- **React** : Uygulamanın ön yüzünün dinamik ve kullanışlı bir yapıda geliştirilmesi için React kullanılmıştır.
- **Vite** : Uygulamanın geliştirme ve derleme sürecini hızlandırmak için Vite kullanılmıştır.
- **react-oauth/google** : Google OAuth entegrasyonu için kullanılmıştır.
- **tanstack/react-query** : Veri yönetimi ve sunucu taraflı verilerin işlenmesi için tanstack/react-query kullanılmıştır.
- **react-router-dom** : Uygulamanın içinde yönlendirme işlemlerinin yönetimi için react-router-dom kütüphanesi kullanılmıştır.
- **react-hot-toast** : Kullanıcılara başarılı veya başarısız işlemlerde bildirimleri sunmak için react-hot-toast kütüphanesi kullanılmıştır.

Veri Tabanı Yapısı ve İşlevler



Kullanıcı Girişi : Kullanıcı Ldif dosyasında tanımlı olan bilgileriyle veya gmail hesabı ile giriş yapabilir, eğer gmail ile ilk defa giriş yapıyorsa kullanıcı bilgileri google_user tablosunda kaydedilir. Başarılı giriş sonrası authorization işlemleri için JWT token dönülür.

Mentorlük Başvurusu : Kullanıcı mentorlük başvurusu için istekte bulunur, her mentorlük başvurusu için Mentor tablosuna yeni bir kayıt girilir. Varsayılan olarak mentor üzerindeki status alanı HOLD olarak ayarlanır. Bir kişinin bir mentor kaydı olmasındansa, bir kişinin onlarca mentor kaydı olabilir, bu yaklaşım ile veri tabanı dizaynı sadeleştirilerek işlevsellik artırılmıştır.

Mentorlük Değerlendirme : Admin mentor tablosuna girilen kayıtları listeler ve Mentor kayıtlarının status alanlarını APPROVED veya REJECTED şeklinde ayarlayabilir.

Mentor Arama : Mentor bulmak için Mentor tablosundan status alanı APPROVED olan kişiler filtrelenerek çekilir. Mentor aranırken ana konuya göre, ana konu ve alt konularına göre veya free text arama ile mentorler çekilebilir.

Mentor Seçme: Mentor tablosundan çekilen mentorlerden bir kişinin seçilmesi halinde Mentorship tablosunda seçen kişinin userld'si menteeld olarak, seçilen mentor'un Id'si mentorld olarak ayarlanarak bir Mentorship kaydı oluşturulur. Kaydın status alanı fazlar oluşturulana dek NOT_STARTED olarak ayarlanır.

Faz Oluşturma : Her bir fazın üzerinde ait olduğu Mentorship'in Id'si tutulmaktadır. Faz oluşturulurken liste şeklinde alınan faz verisi faz tablosuna Mentorship Id ile yazılır ve Mentorship status IN_PROGRESS çekilir.

Faz Tamamlama ve Değerlendirme : Faz tamamlama ve değerlendirme için aynı endpoint kullanılır, bir fazın tamamlanması için mentor veya mentee'den birinin tamamlaması yeterlidir, faz tamamlandıktan sonra faz status'u COMPLETE, bir sonraki faz status'u ise ACTIVE olarak ayarlanır. Faz değerlendirmek için de complete endpointine gidilir, eğer faz ACTIVE durumundaysa faz sadece değerlendirilir ve dönülür. Bütün fazlar değerlendirildi ise Mentorship kaydının status'u COMPLETE durumuna çekilir.

Mail ile Hatırlatma : Arka planda dakika başı çalışan bir job faz tablosunda bitiş tarihi'ne 1 saat kalan kayıtları bulur, eğer böyle bir kayıt bulundu ise kullanıcıya mail servisi ile mail gönderilir.

