**PROJE SÜRECİNE DAİR GÜNCELLEME VE İLERLEME RAPOR**

**Proje Adı:** **Masaüstü Öğrenci Bilgi Sistemi** PostgreSQL ile .NET Core 8 Kullanarak (MVC)

**Ekip Üyeleri:  
Fatma Ulaş  
Sena Altıparmak**

**1. KARŞILAŞILAN SORUNLAR VE YENİDEN YAPILANDIRMA**

Projemizde, veritabanı yapısı ve bağımlılık ilişkileri alanında beklenmedik zorluklarla karşılaştık. Önceki veritabanı kurgumuz başta işlevsel görünse de, uygulamada veri girişi yapılmasını engelleyen bazı **döngüsel bağımlılıklar ve kısıtlayıcı foreign key** ilişkileri içeriyordu. Aşağıda bu sorunların detaylı olarak açıklaması ve çözüm yolları ile yapılan düzenlemeler sunulmuştur.

**Bağımlılık Döngüsü ve FK Sırası Problemi**

Veritabanındaki tablolar arasında karşılıklı bağımlılıkların neden olduğu döngüsel bir yapı mevcuttu. Foreign Key (FK) bağımlılıkları döngüsel hâle gelmişti.

Örneğin:  
universite tablosunda yer alan rektor alanı personel tablosuna bağlıydı.  
personel tablosu ise FK ile fakulteler tablosuna bağlıydı.  
fakulteler tablosu da FK üzerinden tekrar universite tablosuna bağlıydı.

Bu yapı nedeniyle **hiçbir tabloya ilk veri doğrudan eklenememekteydi**, çünkü her biri başka bir tablonun daha önce var olmasını gerektiriyordu. Bu döngüsel ilişki veri girişini imkânsız hâle getirmişti.

Migration sonrası oluşturulan veritabanında çakışan veya geçersiz FK kısıtları nedeniyle Insert işlemleri yapılamıyordu.

**NOT NULL ve UNIQUE Kısıtların Aşırı Kullanımı**

Bazı alanlarda kullanılan NOT NULL ve UNIQUE kısıtları, sistemin esnekliğini azaltmış ve ilişkisel hatalara yol açmıştı. Özellikle:

ogrenci tablosundaki danisman alanı hem NOT NULL hem de UNIQUE olarak tanımlanmıştı. Bu durum, her akademisyenin yalnızca bir öğrenciye danışmanlık yapabileceği anlamına gelmekteydi. Ve veri girişi sırasında esneklik sağlamadığı için hata alınmasına sebep olmuştu.

**2.YAPILAN TEKNİK DÜZELTMELER VE İYİLEŞTİRME ADIMLARI**

-Veritabanı İlişki Şeması Gözden Geçirildi

-Tablolar arası ilişkiler sadeleştirildi ve mantıksal veri akışı yeniden kurgulandı.

- Döngüsel bağımlılıklar kaldırılarak, veri girişi yapılabilir bir hiyerarşi oluşturuldu:

Fakülte, Bölüm, Akademisyen, Ders, Öğrenci, Not

-Veri Girişi Test Senaryoları Hazırlandı

-Her tablo için örnek veriler oluşturularak FK bağlantıları veritabanı üzerinde test edildi.

**3. DEĞİŞİKLİK SONRASI ETKİLENEN KISIMLAR VE HATA MEYDANA GETİREN DURUMLAR**

Gerçekleştirilen bu veritabanı değişiklikleri, proje kod tarafında kullanılan Entity Framework varlıkları ile uyumsuzluklara neden olmuştur. Yapılan değişimlerden sonra EFramework üzerinden oluşturulan varlıklar ile yeni veritabanı şeması arasında hatalar meydana gelmiş ve çalışma sırasında çeşitli sorunlar ortaya çıkmıştır. Bu problemleri çözmek amacıyla farklı çözüm yolları ve iyileştirme yöntemleri denenmiş ancak istenilen sonuca ulaşılamamıştır.

Bu nedenle, proje geliştirme sürecinde ilerleme sağlamak adına, proje yeniden yapılandırılmış ve yeni bir çözüm (solution / .sln) üzerinden geliştirme sürecine sıfırdan başlanmıştır.

-Entity Framework ve Dependency Injection Yapıları’nı güncellemek isterken eski yapılarla çakışması ve üzerine yazma yapılamaması nedeniyle aynı proje dosyası üzerinde devam etmekte zorlandık.

-Önceki veritabanı bağlantısı ve Entity Framework ile oluşturulmuş olan Models, DbContext gibi dosyaları kaldırarak sıfırdan yapılandırmak istedik fakat dosyaları kaldırmamıza rağmen güncel veritabanı bağlantısını sağlayamadık ve yeni düzenlenmiş varlıklarımızın Entity Framework ile sıfırdan oluşturulması mümkün olmadı.

**Bu nedenle yeni bir çözüm dosyası üzerinde aynı adımları tekrarlayarak projeye olabildiğince hızlı biçimde kaldığımız yerden devam etmeyi amaçladık:**

-Solution oluşturuldu ve içine katmanlı mimariye uygun şekilde katmanlar eklendi. (Presentation Layer, Data Access Layer, Business Layer)

-Yeniden oluşturulan katmanlar arasında bağımlılıklar düzenlendi. Business-Data ve Presentation-Business arasında proje bağımlılıkları bulunmakta fakat Presentation Layer doğrudan Data katmanına erişemeyip, Business katmanı ile erişilebilmektedir.

-Veritabanı yapısına uygun olarak Entity Framework varlıkları güncellenmiş olarak yeniden oluşturuldu. (Ayrıca bir Entity Katmanı kullanmak yerine DAL katmanında EF ile oluşturduğumuz Models ve DbContext’i kullanmayı tercih ettik.)

-Veritabanı bağlantı ayarları (connection string) yeni yapıya uygun şekilde yeniden tanımlandı.

-Kod tarafında uyum sorunlarını ortadan kaldırmak amacıyla ilgili alanlarda gerekli güncellemeler yapıldı, Form yapılarımız yeni proje dosyamıza doğru şekilde aktarıldı.

Bu kapsamlı düzenleme ve yeniden yapılandırma süreciyle birlikte proje, daha kararlı ve nihayet geliştirilmeye uygun bir yapıya kavuşturulmuştur. Bu süreç içerisinde sorunları düzeltip projeye kaldığımız yerden devam edebilecek düzeyde ilerleme sağlamayı başardık.

**Çalışma Süreci**

| **Tarih** | **Çalışılan Konu** | **Açıklama** |
| --- | --- | --- |
| **11 Mart** | Veritabanı Bağımlılıklarının Analizi | Tablolar arası FK ilişkileri incelendi. universite, personel, fakulteler arasında döngüsel bağımlılık tespit edildi. |
| **14 Mart** | ER Diyagramı Güncellemesi | FK sırasını bozmayacak şekilde yeni ilişki akışı oluşturuldu. rektor, dekan, baskan gibi alanlar geçici olarak nullable tanımlandı. |
| **18 Mart** | Veri Denemeleri ve FK Hataları | Yeni ilişki yapısına rağmen veri girişi sırasında hatalar devam etti. Composite FK’lerde birebir eşleşme zorunluluğu olduğu görüldü. |
| **21 Mart** | Veri Girişi Sırasının Planlanması | Önce bağımsız tabloların doldurulmasına karar verildi (örneğin: bolum, sinif), ardından ilişkili tablolar için veriler oluşturuldu. |
| **26 Mart** | Entity Güncellemeleri | Güncellenen veritabanı yapısına uygun olarak entity sınıfları yeniden oluşturulmaya çalışıldı fakat başarısız oldu. |
| **29 Mart** | EF ve Bağlantı Sorunları | DbContext ve Models silinerek sıfırdan EF kurulmaya çalışıldı fakat hata alındı. |
| **4 Nisan** | Kodun Veritabanıyla Uyuşmaması | Mevcut sorunlar için ilgili kaynaklarda bulunan çözüm yöntemleri denendi. |
| **8 Nisan** | Yapının Çalışmadığının Netleşmesi | İşlemler başarısız olduğu için kod üzerinden ilerleme sağlanamadı. Proje yapısının sıfırlanmasına karar verildi. |
| **10 Nisan** | Yeni Proje Oluşturma Kararı | Mevcut proje dosyasındaki karışıklık ve yapı bozulmaları nedeniyle yeni bir çözüm dosyası oluşturulmasına karar verildi. |
|  | Yeni .NET Çözüm Dosyasının Oluşturulması | Yeni ve temiz bir PL, BLL, DAL katmanlarıyla yapılandırılmış .NET projesi başlatıldı. |
| **13 Nisan**  **18 Nisan** | Yeni Entity Sınıflarının Oluşturulması | Tüm tablolar için yeniden entity tanımlamaları yapıldı. Bire bir ve bire n ilişkiler açık şekilde yazıldı. |
| **24 Nisan** | Proje Günlükleri ve Raporlama | Sürece dair ilerlemeler ve kararlar yazılı rapor haline getirilmeye başlandı. |