



BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
2021-2022 YAZ DÖNEMİ
BİL372 VERİTABANI SİSTEMLERİ PROJE ARA RAPORU

01.07.2022

İçindekiler

1) PROJE GRUBU	2
2) GERÇEK DÜNYA PROBLEMİ.....	2
3) GEREKSİNİM ANALİZİ	3
A) MINIWORLD	3
B) MEVCUT SİSTEMLERİN ÖZELLİKLERİ	3
C) İHTİYACIN BELİRLENMESİ	3
D) YAZILIM VE DONANIM MALİYET ANALİZLERİ.....	4
4) KAVRAMSAL TASARIM	5
A) VARLIKLAR VE ÖZNİTELİKLER	5
B) İLİŞKİLER	6
C) YAPISAL KISITLAR.....	6
D) EER VERİ MODELİ	7
5) MANTIKSAL TASARIM	9
A) MANTIKSAL MODEL	9
B) İLİŞKİLERİN NORMALİZASYONU	9
6) TASARIMIN UYARLANMASI	9

1) Proje Grubu

Projenin 4 kişilik bir grup olarak yürütülmesi planlanmaktadır. Ekip üyeleri arasında görev paylaşımının eşit şekilde yapılması ve tüm ekip üyelerinin projenin her aşamasında detaylıca rol alması, ilgili konular hakkında bilgi sahibi olması istenmektedir.

Ekip üyeleri şu şekildedir:

- Fatih Arslan Tugay - 181101008
- Mert Can Gönen - 181101039
- Elif Nebioğlu - 191101024
- Gökçe Başak Demirok - 191401005

2) Gerçek Dünya Problemi

Projede bir üniversitenin (örnek olarak TOBB ETÜ) ders seçim sistemini ele alınacaktır. Üniversitelerde öğrenci ve ders bilgi sistemleri altyapısal olarak önemli bir konumdadır. Bütün öğrenci ve ders işlemlerinin gerçekleştirildiği bu sistemlerde hem veritabanı hem de kullanıcı arayüzü önemli bir konuma sahiptir. Bu bileşenlerin işlevini hatasız yerine getirmesi gerekmektedir. Bu projede de örnek ve basit bir ders seçim sistemi oluşturularak bileşenlerin işlevleri tanımlanacak ve çalışan bir sistem ortaya konulmaya çalışılacaktır.

3) Gereksinim Analizi

a) Miniworld

Gerçek dünya problemi kısmında da bahsedildiği üzere Miniworld olarak bir üniversitenin ders seçim sistemi seçilmiştir.

b) Mevcut Sistemlerin Özellikleri

Mevcut ders seçim sistemlerinde temel olarak yer alan özellikler:

- Farklı kullanıcı grupları, grupların nitelikleri ve işlem kısıtları (giriş yapılan kullanıcıya göre farklı özellikler),
- Yeni kullanıcı oluşturma,
- Kişisel kullanıcı işlemleri (şifre değiştirme, bilgileri düzenleme vs.),
- Üniversite yapısının bölümlere ayrılarak bölümler altında işlemlerin gerçekleştirilmesi,
- Sistem yöneticileri tarafından ders oluşturup akademisyene atanması,
- Öğrencilerin ders seçimi yapması ve onay/ret durumları,
- Akademisyenlerin öğrenci dersini onay/ret durumları, derse asistan ataması (yüksek lisans veya doktora öğrencileri arasından) ve dersi alan öğrenci listeleme,
- Akademisyenlerin ders bazında duyuru atamaları (ders bilgi sistemi kapsamında olmasa da not/ödev girişleri de yapılabilmektedir),
- Öğrencilerin ders listesini ve ders programlarını görüntülemesi olarak sıralanabilir.

Sıralanan özellikler haricinde gerçek sistemlerde birçok farklı özellik bulunabilmektedir fakat projede örnek bir sistem hazırlanması açısından yukarıdaki özelliklerin yeterli olduğu kararlaştırılmıştır. İlerleyen süreçlerde projeye özellik eklenmesi veya gereksiz görülen özelliklerin çıkartılması gibi güncellemeler yapılabilir.

c) İhtiyaçın Belirlenmesi

Sistemde ihtiyaç duyulacağı öngörülen ve projeye entegre edilecek nitelikler şu şekilde sıralanabilir:

- Tüm kullanıcıların bilgileri ve nitelikleri tutulmalı, sistemde kullanıcı işlem kısıtlarında bu özellikler kullanılmalıdır.
- Sistem güvenli olmalıdır. Örneğin parola bilgileri açık olarak tutulmamalıdır.

- Sistemi kullanacak olan kullanıcılar; öğrenci, akademisyen ve sistem yöneticileri olarak ayrıştırılmalı ve bu ayrıştırmaya göre kullanıcılara belirlenen arayüzler sunulmalıdır.
- Sistem yöneticileri yeni kullanıcıları ekleyebilmeli, kullanıcı eklerken gerekli bilgileri atayarak (örneğin kullanıcı adı, e-posta adresi, kullanıcı grubu vb.) kullanıcının seçememesi gereken özellikleri kendisinin belirlemesi gerekmektedir.
- Akademisyenler kendi derslerini ve bu dersin bilgilerini görüntüleyebilmeli, dersi alan öğrencileri listeleyebilmeli ve dersi talep eden öğrencileri onaylama/reddetme mekanizmasına sahip olmalıdır.
- Öğrenciler rahatlıkla alabileceği dersleri seçebilmeli, bölümü ile alakası olmayan (veya seçemeyeceği) dersleri görememeli ve bu dersler arasından seçim yapamamalıdır.
- Veritabanında yukarıda belirtilen özellikleri sağlayacak varlık ve öznitelikler dışında gereksiz veri tutulmamalıdır. Ayrıca bu özellikleri sağlayacak minimum verilerin tutulması zorunlu olmalıdır. Örneğin kullanıcıların sisteme giriş işlemi ‘kullanıcı adı’ ve ‘parola’ bilgileri ile gerçekleştirileceğinden sistem yöneticisi bir kullanıcı oluşturmak istediğinde bu kullanıcının ‘kullanıcı adı’ bilgisini boş bırakarak bu işlemi yapamamalıdır.
- Ek bir özellik olarak duyuru, ders ataması, ders talebi/onayı/reddi veya ödev/not girişi vb. durumlarda bildirim sistemi oluşturulması amaçlanmaktadır. Bu işlev için kullanıcı girişi olduğunda bildirim kontrolü yapılması planlanmaktadır.

d) Yazılım ve Donanım Maliyet Analizleri

Projenin çalışması için bir veritabanı sistemi, bir kullanıcı arayüzü (ön-uç, front-end) ve bir de kullanıcı arayüzü fonksiyonlarının gerçekleştirileceği arka-uç (back-end) mimarisine ihtiyaç vardır. Sistem, güçlü bir donanım gerektirmeyeceği için tüm sistemin (veritabanı, front-end ve back-end) tek bir bilgisayar üzerinde çalıştırılacağı öngörülmektedir.

Hem örnek bir proje olması açısından hem de sistemin çalıştığını gözlemek için veritabanına aşırı bir veri girişi gerekmeyeceğinden ve gerçek ortamda kullanılmayacağından veriler çok ağır bir boyutta olmayacaktır. Fazla ileri düzeyde bir donanım gerektirmeyen bu sistemin şu anki ekonomiye göre yaklaşık 800-1000\$’lık bir sistemde rahatlıkla işlevini gerçekleştireceği düşünülmektedir.

4) Kavramsal Tasarım

İlk aşamada oluşturulan tasarım aşağıdaki gibidir. Projenin ilerleyen aşamalarında tasarımda değişiklikler (varlık veya öznitelik ekleme-çıkarma vb.) yapılabileceği öngörülmektedir.

a) Varlıklar ve Öznitelikler

Veritabanında bulunması gereken varlıklar ve bu varlıkların öznitelikleri aşağıdaki gibidir. Format VARLIK → (öznitelik1, öznitelik2, öznitelik3, ...) şeklinde okunmalıdır. Altı çizili öznitelikler anahtar öznitelik durumundadır. Alt kategoriler maddelenerek belirtilmiştir.

Aşağıdaki varlıklar ve öznitelikler EER model ve diyagramlar henüz oluşturulmadan belirlenmiştir. Varlıkların veritabanında barındıracağı öznitelikleri belirtildiği gibi olsa da model ve diyagramlarda sırasıyla ve tam olarak belirtilmeyebilir çünkü özniteliklerin bir kısmı ilişkilerin tanımlanmasından sonra aktarılacaktır. İlişki vasıtasıyla alınan öznitelikler *italik* olarak belirtilmiştir.

- KULLANICI → (kullanıcı adı, adı-soyadı, doğum tarihi, adres, telefon numarası, e-posta adresi, kullanıcı grubu)
 - SİSTEM YÖNETİCİSİ → (yetki seviyesi)
 - AKADEMİSYEN → (*bölüm*, unvan¹)
 - ÖĞRENCİ → (*bölüm*, *akademik danışman*, öğrenci numarası, sınıf, eğitim programı²)
- BÖLÜM → (bölüm kodu, *bölüm başkanı*, maksimum kredi sınırı³)
- DERS → (ders kodu, *bölüm kodu*, *akademisyen*⁴, *asistan*⁵, kredi, kontenjan, ders saatleri)
- BİLDİRİM → (bildirim id, kullanıcı adı, tarih - saat, başlık, mesaj, okundu bilgisi⁶)

¹ Öğretim görevlisi, doçent veya profesör

² Lisans, yüksek lisans veya doktora seviyesi

³ Bölüm öğrencilerinin alabileceği maksimum kredi

⁴ Dersi veren akademisyen

⁵ Öğrenci varlığında eğitim durumu yüksek lisans veya doktora olan öğrenciler arasından

⁶ Okunan bildirimin tekrar tekrar kullanıcıya gösterilmemesi için

b) İlişkiler

- KULLANICI, BİLDİRİM alır.
- BÖLÜM, ÖĞRENCİ'ler içerir.
- BÖLÜM, AKADEMİSYEN'ler içerir.
- BÖLÜM, DERS'ler içerir.
- ÖĞRENCİ, DERS alır.
- ÖĞRENCİ, DERS'e asistanlık yapar.
- AKADEMİSYEN, DERS verir.
- AKADEMİSYEN, BÖLÜM'e dekanlık yapar.
- AKADEMİSYEN, ÖĞRENCİ'ye danışmanlık yapar.

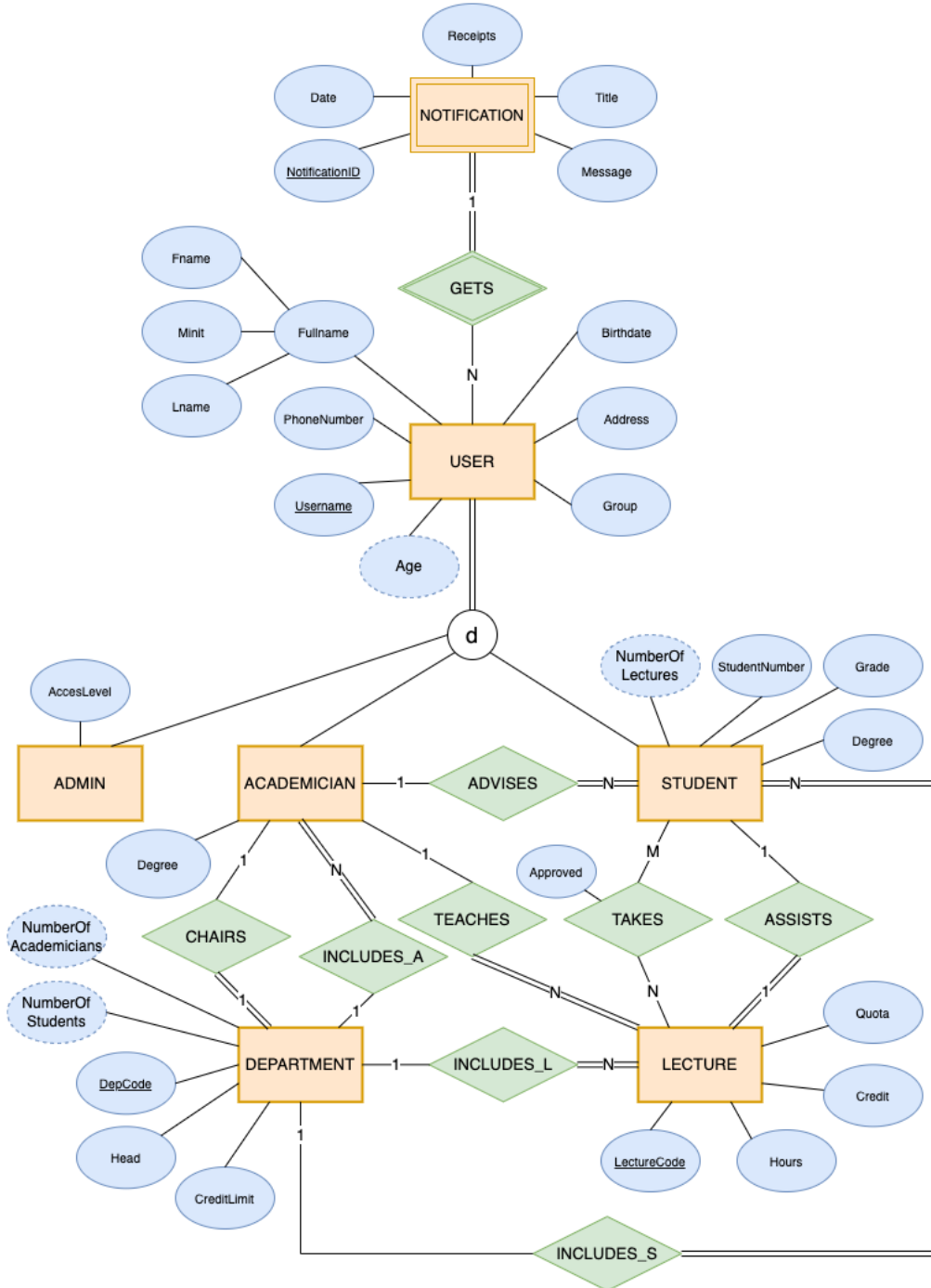
c) Yapısal Kısıtlar

- Bir BİLDİRİM yalnızca bir KULLANICI'ya gidebilir fakat bir KULLANICI birden fazla BİLDİRİM alabilir. (1:N)
- Bir ÖĞRENCİ yalnızca bir BÖLÜM'e bağlıdır fakat bir BÖLÜM birden fazla ÖĞRENCİ içerebilir. (1:N)
- Bir AKADEMİSYEN yalnızca bir BÖLÜM'e bağlıdır fakat bir BÖLÜM birden fazla AKADEMİSYEN içerebilir. (1:N)
- Bir DERS yalnızca bir BÖLÜM'e bağlıdır fakat bir BÖLÜM birden fazla DERS içerebilir. (1:N)
- Bir ÖĞRENCİ birden fazla DERS seçebileceği gibi bir DERS birden fazla öğrenci tarafından seçilebilir. (M:N)
- Bir ÖĞRENCİ yalnızca bir DERS'e asistanlık yapabilir ve bir DERS'in yalnızca bir asistan ÖĞRENCİ'si olabilir. (1:1)
- Bir DERS yalnızca bir AKADEMİSYEN tarafından verilebilir fakat bir AKADEMİSYEN birden fazla DERS verebilir. (1:N)
- Bir AKADEMİSYEN yalnızca bir BÖLÜM'e dekanlık yapabilir ve bir BÖLÜM'ün yalnızca bir dekan AKADEMİSYEN'i olabilir. (1:1)
- Bir ÖĞRENCİ'nin yalnızca bir AKADEMİSYEN danışmanı olabilir fakat bir AKADEMİSYEN birden fazla ÖĞRENCİ'ye danışmanlık yapabilir. (1:N)

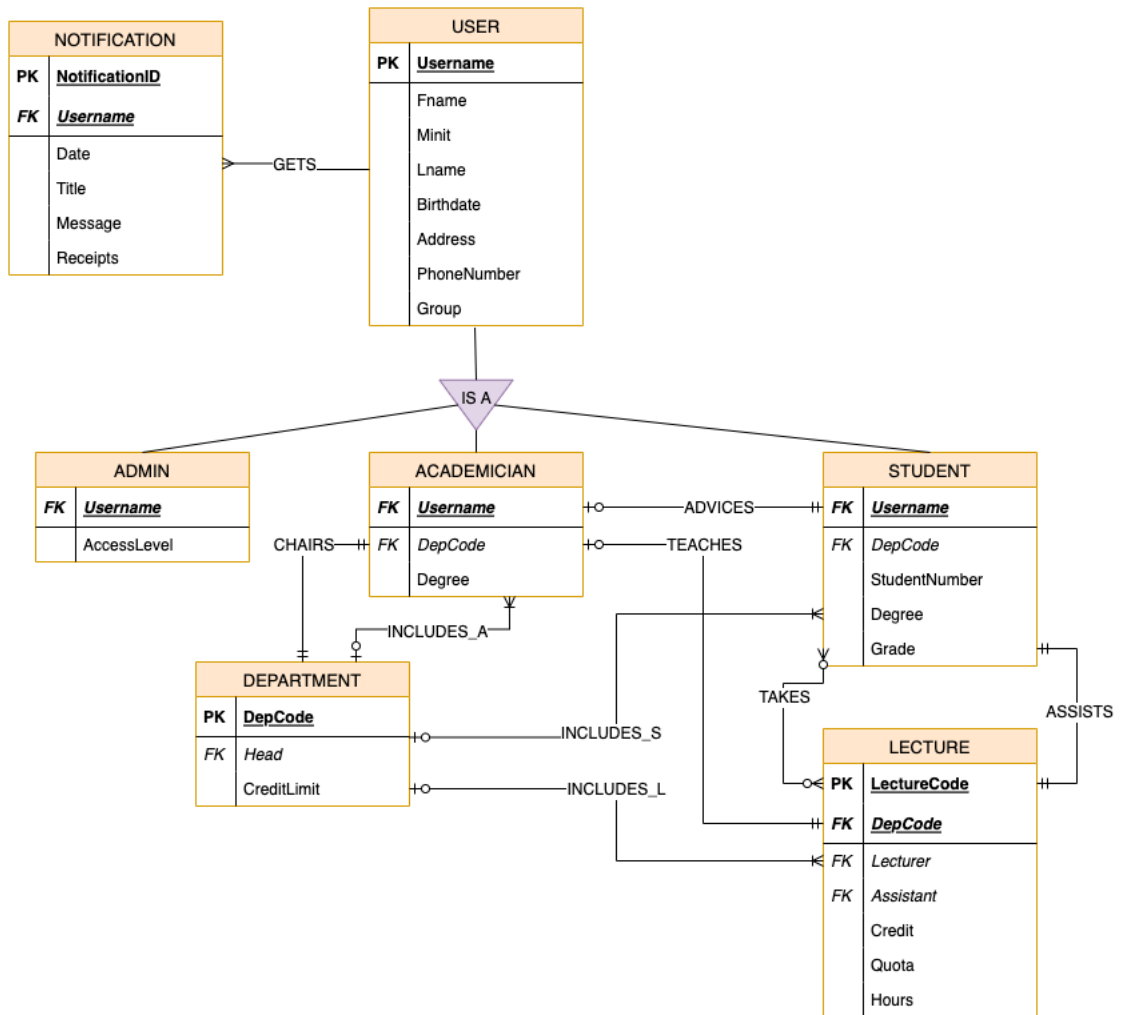
d) EER Veri Modeli

Varlıklar ve özniteliklerinin EER veri modeli öncelikle flowchart tarzı gösterimle hazırlanmıştır. Daha sonra bu tasarım üzerinden UML(ERWin) tarzı diyagram oluşturulmuştur.

- Flowchart tarzı gösterim:



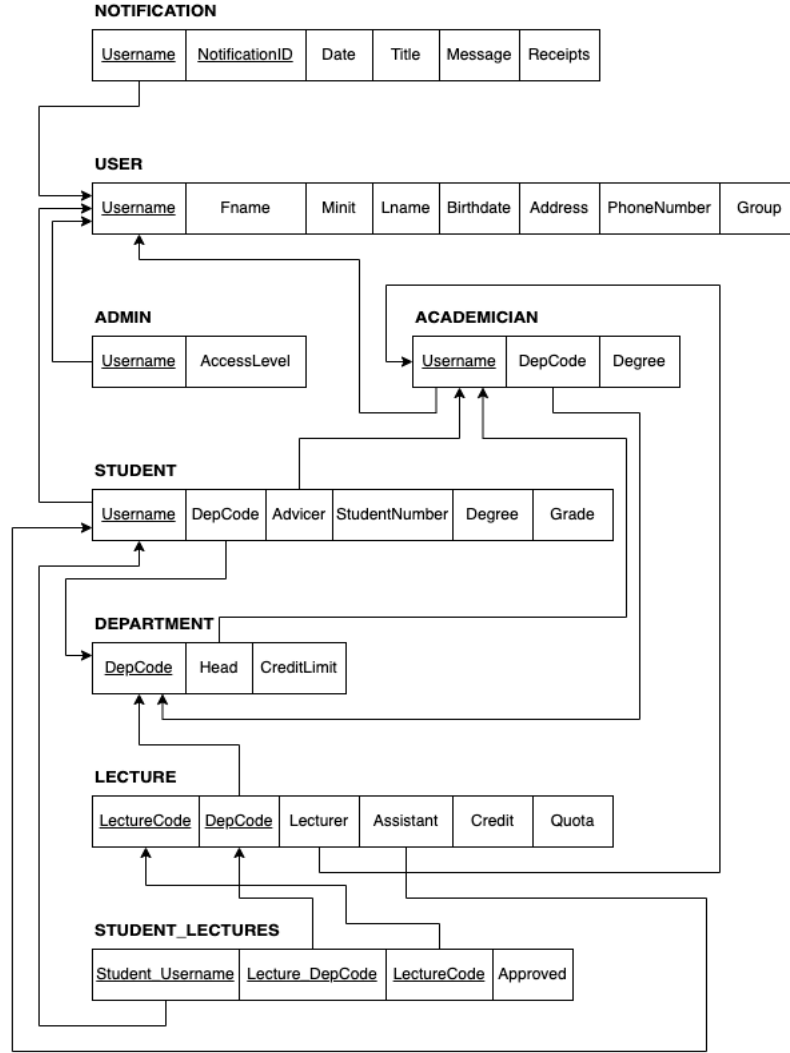
- UML diyagramı (ERWin) tarzı gösterim:



5) Mantıksal Tasarım

a) Mantıksal Model

Mantıksal model dönüşümü yapılırken M:N ilişki olan ÖĞRENCİ-DERS ilişkisi tabloya dönüştürülmüştür.



b) İlişkilerin Normalizasyonu

Final raporunda eklenecektir.

6) Tasarımın Uyarlanması

Tasarımın uyarlanması ve detaylı açıklamaları final raporunda eklenecektir. Şu anlık yalnızca örnek tablo yaratılması ve verilerle doldurulmaları için SQL komutları oluşturulmuştur.