

# Prolab-2

## Proje 3

Gökhan Çobanoğlu  
200201015  
Bilgisayar Mühendisliği

Bilal Cihangir Erdem  
200201076  
Bilgisayar Mühendisliği

### I. ÖZET

Bu projede bizden bir bankanın yönetim sistemi için bir veritabanı tasarlamamız ve bu veritabanının üzerinde gerekli işlemleri gerçekleyen bir uygulama geliştirmemiz amaçlanmaktadır. Bu proje geliştirilirken Java programlama dili ve MySQL veritabanı kullanılmıştır.

Tasarlanan veritabanı, bankanın müşterilere, çalışanlara, hesaplara ve işlemlere ilişkin bilgileri organize bir şekilde işlemesine yardımcı olacaktır. Bu şekilde bankanın ve müşterilerin ihtiyacı olan bilgilere daha kolay ulaşabilmesi sağlanmış olacaktır. Ayrıca, veritabanı kullanarak müşteri istekleri ve bankanın ihtiyaçları doğrultusunda raporlar hazırlanabilecektir.

Bu veritabanının tasarımında ihtiyaçları görmek için ilk olarak E-R diyagramı oluşturduk. E-R diyagramı sistem içerisinde var olabilecek varlıkların ve aralarındaki ilişkilerin görsel olarak ifade edilmesinde bize yardımcı oldu. Veritabanında oluşturulan tablolar normalizasyon kurallarına uyarak oluşturulmuş olup 3NF formundadır.

### II. GİRİŞ

Bu projede banka içerisinde müşteri, temsilci ve banka müdürü olmak üzere 3 adet rol bulunmaktadır. Her rolün kendine ayrı arayüzü ve rolleri bulunmaktadır.

Müşteriler hesaplarından para çekebilirler ve yatırabilirler. Yeni bir hesap açabilirler ya da hesaplarını silebilirler. Birbirleri arasında para transferi yapabilirler. Bilgilerini güncelleyebilirler. Bankadan para transferi yapabilir, kredi talep edebilirler.

Temsilciler her müşterinin bir temsilcisi vardır. Müşteri ekleme, silme ve düzenleme yapabilirler. Müşteri bilgilerini güncelleyebilirler. Müşterilerden gelen hesap açma, silme ve kredi taleplerini görüntüleme ve onaylama sorumluluğu temsilcilere aittir. İlgilendikleri müşterilerin işlemlerini görüntüleyebilir.

Müdür bankanın genel durumunu (gelir, gider, kar ve toplam bakiye) görüntüleyebilmektedir. Ni para birimi (Dolar, Euro, Sterling vs.) ekleyebilir ve kur değerlerini güncelleyebilir. Çalışanların maaş ücretlerini belirleyebilecektir. Kredi ve gecikme faiz oranını belirler. Müşteri ekleyebilir. Sistemi bir ay ilerletebilir. Sistemi bir ay ilerletebilir. Bu ilerletme işlemi sonucunda maaşların ödenmesi, gelir-gider durumlarının güncellenmesi ve müşterilerin bir sonraki aya ait borçlarının kendilerine yansıtılması gerekmektedir.



Fig. 1: GİRİŞ EKRANI

### III. YÖNTEM

Bu projede arayüzler için JFrame kullandık. Bir kullanıcı arayüzü oluşturmak için ya bir tane JFrame nesnesi veya Applet oluşturmak gerekmektedir.

Arayüzde kullanıcı 3 şekilde giriş yapabilmektedir: Müşteri, Müdür veya Temsilci olacak şekilde. Müşteri olarak giriş yaptığını varsayalım o zaman karşısına bir menü çıkmaktadır. Bu menüde para çekme, yatırma, hesap açma, hesap silme, para transferi yapma, bilgilerini güncelleme, kredi talep etme ve aylık özeti görüntüleme gözükmemektedir.

Müdür menüsünde bankanın genel durumu görüntüleme, yeni para birimi ekleme, kur değerini güncelleme, temsilcilerin maaşını belirleme, kredi ve gecikme faiz oranını belirleme ve müşteri ekleme kısımları vardır.

Temsilci menüsünde müşteri ekleme, silme, düzenleme, müşteri bilgi güncelleme, müşterilerin genel durumunu görüntüleme, müşterilerden gelen hesap açma, silme ve kredi taleplerini görüntüleyebilir ve onaylama kısımları vardır.

Database kısmına gelecek olursak ise tabloları 3NF formunda yaptık. 3NF formunda olan tablolarda multivalued attributes (çok değerli nitelik), partial functional dependencies (kısmi fonksiyonel bağlantılar) ve transitive dependencies (geçici bağlantılar) olmaz. Oluşturduğumuz tablolarda primary key ve foreign keylerle birbiri arasında relation (ilişki) kurabildik. Primary key ile primary keyin

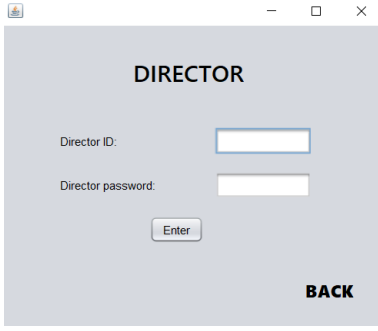


Fig. 2: MÜDÜR GİRİŞ EKRANI

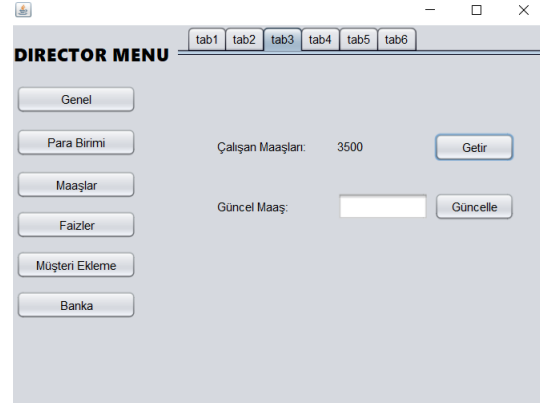


Fig. 4: MÜDÜR MENÜ 2

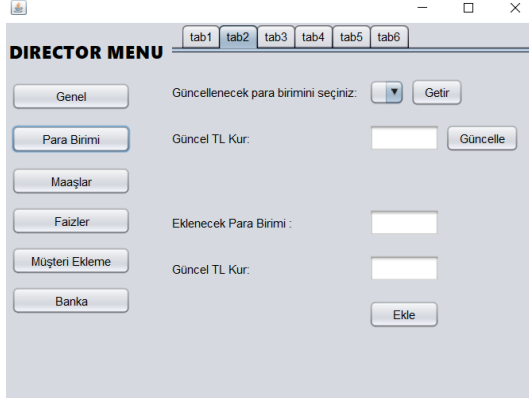


Fig. 3: MÜDÜR MENÜ

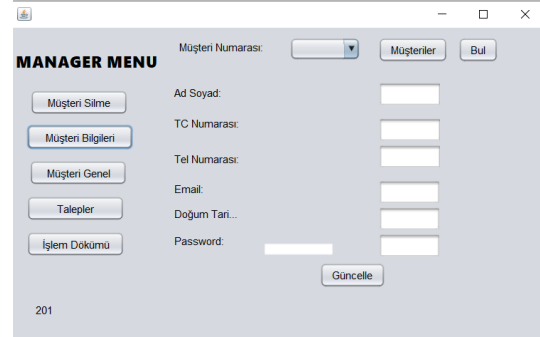


Fig. 5: TEMSİLCİ MENÜ

bulunduğu tablodaki tüm bilgilere erişilebilir. Primary keyler unique(eşsiz) olurlar ve asla null(boş) olamazlar. Örneğin müşteri tablosundaki müşteriID müşteri tablosu için bir primary keydir ve müşteriID ile o müşterinin her bilgisine erişebiliriz. Foreign Key (yabancı anahtar) ikincil anahtar olarak da ifade edilmektedir. Bir veri tablosuna girilebilecek değerleri başka bir veri tablosundaki alanlarla ilişkilendirmeye yarar. Özetle, başka bir tablonun birincil anahtarının bir diğer tablo içerisinde yer almasıdır. Çoğunlukla bir ana tablo (parent) ile alt tablonun (child) ilişkilendirilmesinde kullanılır. Bu sayede olası veri tekrarlarının önüne geçilebilmekte ve ilerleyen zamanda söz konusu olabilecek güncellemelerde ilgili verilerin her yerde güncellenmesi sağlanabilmektedir.

Java ile database arasındaki connection(bağlanma) ise MySQL içinde bulunan JDBC(JAVA DATABASE CONNECTOR) ile yapıldı. Netbeans'de bulunan add library kısmı ile MySQL JDBC ekledik. Bu şekilde yaptığımız kayıtlar database'e eklenmiş ve database'de bulunan veriler ile giriş yapıp bu veriler sonucunda diğer verileri görüntüleyebildik.

#### IV. DENEYSEL SONUÇLAR

##### REFERENCES

- [1] <https://www.youtube.com/watch?v=t8VH8QNTToBwlist=PLKe9hJMFer2F2JXcF8txUCxJGtbIZ6Eindex=1>
- [2] <https://www.youtube.com/watch?v=0cU3bo1FHAlis=PLfXp1hx6BsEfJHgD2gGdW9OK6suIMBlyWindex=1>
- [3] <https://www.youtube.com/watch?v=PXZ0I7DLK0list=PLtW6-UgwXur6W9aSMHVfLugu9Jcq0nwfindex=1>
- [4] <https://www.youtube.com/watch?v=F4vpJJH2PZUlist=PLtW6-UgwXur4AFnlm1pTkL9li-V663hsindex=8>

- [5] <https://www.youtube.com/watch?v=1Z85O-W9Cwglist=PLtW6-UgwXur76NrZJiXaeIDssbT8fMxfQindex=3>
- [6] <https://www.youtube.com/watch?v=kh1cRw7AdAclist=PLtW6-UgwXur76NrZJiXaeIDssbT8fMxfQindex=4>
- [7] <https://serhatpehlivan.com/er-model-nedir/>
- [8] <https://medium.com/turuncu-internet-solutions/normalizasyon-nedir-f212541f15fb>
- [9] <https://www.javatpoint.com/steps-to-connect-to-the-database-in-java>
- [10] <https://www.youtube.com/watch?v=z1O0pnd07GAlis=PLtW6-UgwXur4AFnlm1pTkL9li-V663hsindex=9>

Fig. 6: TEMSİLCİ MENÜ 2

Fig. 9: MÜŞTERİ MENÜ 3

Fig. 7: MÜŞTERİ MENÜ

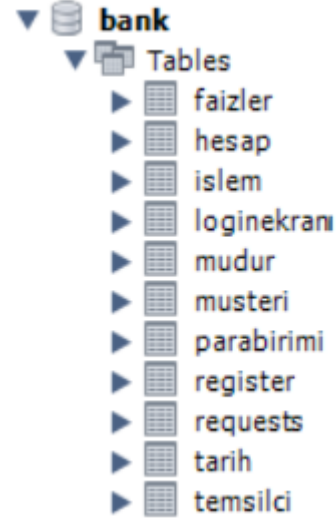


Fig. 10: tablolar

Fig. 8: MÜŞTERİ MENÜ 2

```
//işlemler tablosuna transfer işlemini ekleme
String queryfor="Select Count(islemid) from islem";
Statement statefor=Conn.createStatement();
ResultSet resultfor=statefor.executeQuery(queryfor);
resultfor.next();
int islemsayisi=Integer.parseInt(resultfor.getString("Count(islemid)"))+1;
String islemno=String.valueOf(islemsayisi);
String islem="Para Transferi";
String queryforislem="insert into islem values (?, ?, ?, ?, ?, ?)";
String gönderilenstring=String.valueOf(gönderilen);
PreparedStatement stateforislem=Conn.prepareStatement(queryforislem);
stateforislem.setString(1, islemno);
stateforislem.setString(2, musteriid);
stateforislem.setString(3, account);
stateforislem.setString(4, hedefhesap);
stateforislem.setString(5, islem);
stateforislem.setString(6, gönderilenstring);
stateforislem.executeUpdate();
```

Fig. 11: işlem tablosuna işlemleri ekleme kodu

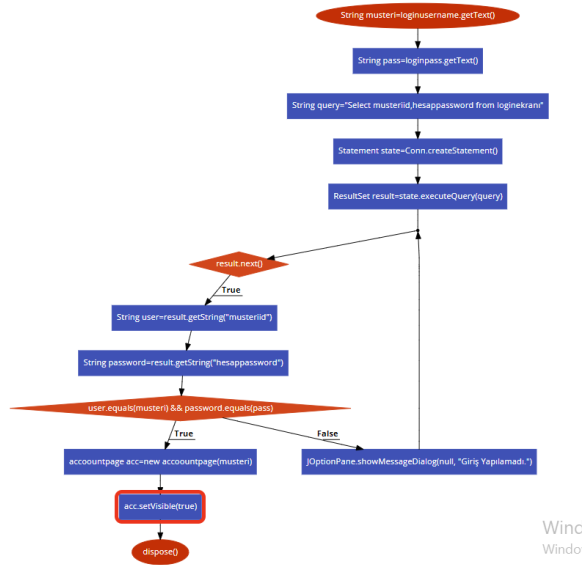


Fig. 12: akış diyagramı

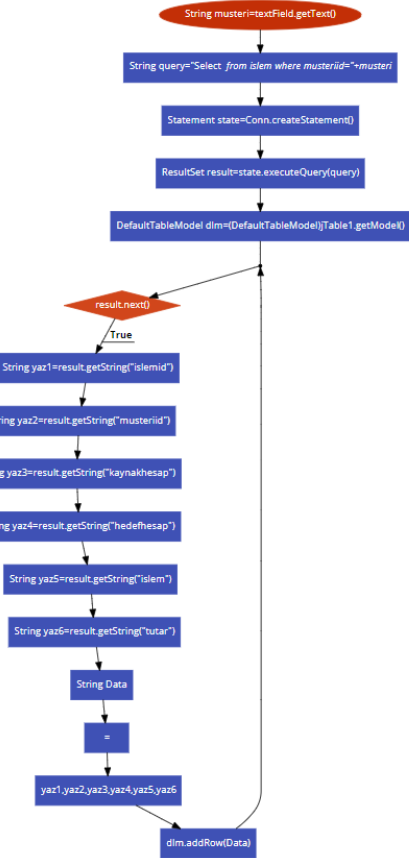


Fig. 13: akış diyagramı 2

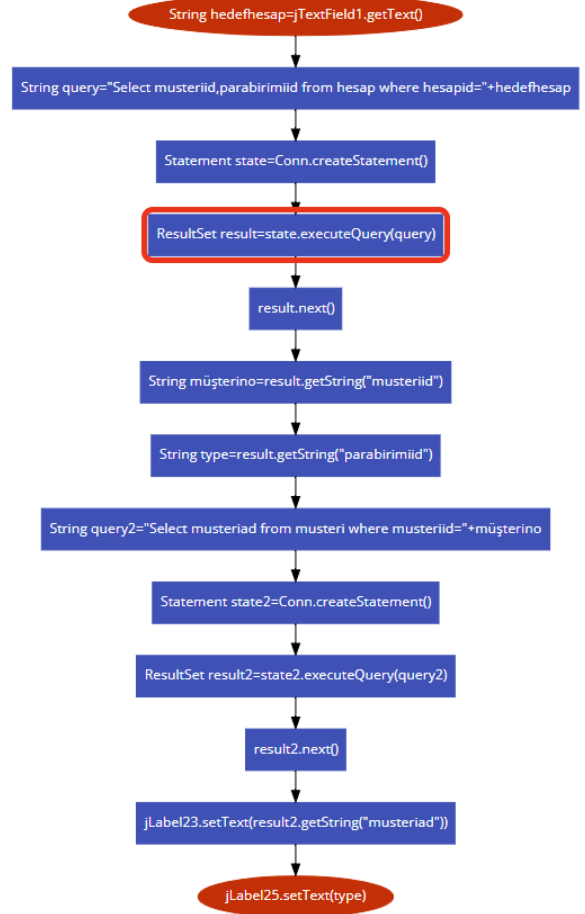


Fig. 14: akış diyagramı 3

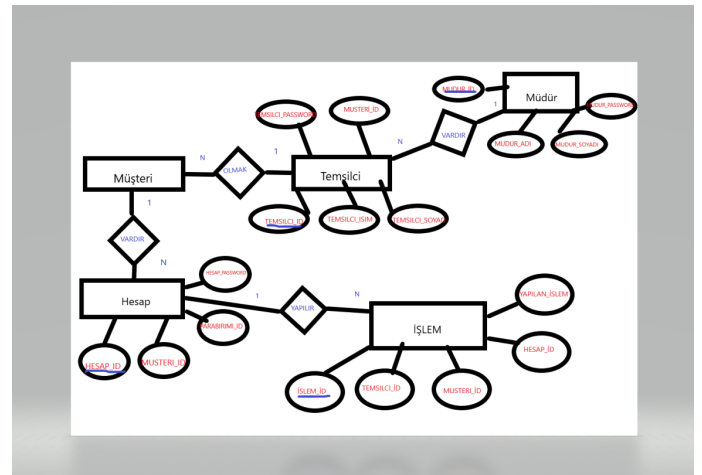


Fig. 15: ER