

**Kocaeli Üniversitesi**  
**Bilgisayar Mühendisliği Bölümü**  
**Programlama Labaratuvarı 2**

Ahmet Suna

210202071

### 1.Özet

Bu proje kapsamında sanal ortamda arsa ve market ile ilgili (alım-satım-kiralama-işletme vb.) ticari aksiyonları yerine getiren bir platformun veritabanı sistemini tasarlamamız ve bu veritabanı üzerinde gerekli işlemleri gerçekleyen bir oyun geliştirmemiz amaçlanmaktadır.

Proje C++/Java/Python/C# dillerinden biri kullanılarak form veya internet tabanlı bir proje olarak gerçekleştirilebilir. Veri tabanı yönetim sistemi için MySQL, PostgreSQL veya MsSQL kullanılabilir

### 2.Giriş

Tasarlanacak veritabanı, metaland platformu üzerindeki kullanıcılar için varlıklara yönelik hesaplara ve tüm işlemlere ilişkin bilgileri organize bir şekilde yönetilmesine yardımcı olacaktır. Bu sayede kullanıcıların ihtiyacı olan bilgilere daha kolay ulaşabilmesi sağlanmış olacaktır. Bir veritabanı tasarımının ilk aşamasında sistemin ihtiyaçlarının belirlenmesi ve depolanacak bilgi türlerinin tanımlanması için Varlık-İlişki (ER) diyagramı oluşturulmalıdır. ER diyagramı sistem içerisinde var olabilecek varlıkların ve aralarındaki ilişkilerin görsel olarak ifade edilmesi için kullanılır. Geliştirme sırasında, ER diyagramı gereksinimlerin daha açık ve özlü bir şekilde haritalanmasına yardımcı olmaktadır.

Sizlerden problemlerin çözümü için bir veritabanı tasarımı yapmanız

istenmektedir. Tablo sayısı tüm tablolar en az 3NF normalizasyon formuna uyacak şekilde size bırakılmıştır. Projede oluşturulacak veritabanının ER diyagramının oluşturulması ve arayüzde gösterilmesi gerekmektedir. Diyagram üzerinden gerçekleştirilen normalizasyon işlemlerinin adım adım gösterilmelidir

### 3. Yöntem

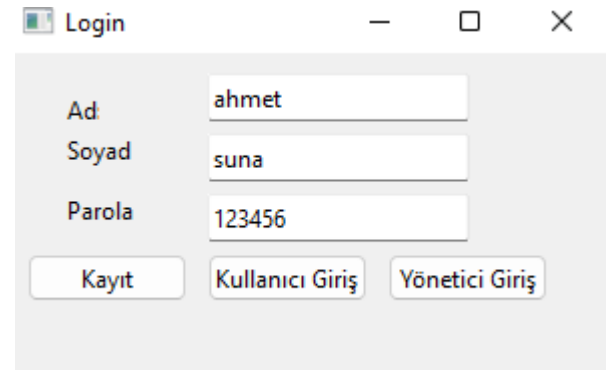
Proje python dili kullanılarak geliştirilmiştir.

Projede arayüz için pyqt6 kullanılmış hazır ui için qtCreator kullanılmıştır.

Projede istenilen isterler büyük ölçüde yapılmıştır.

Projede istenilen zaman kavramı yapılmamıştır.

Giriş ekranı aşağıdaki gibidir.



Admin yönetici paneli aşağıdaki gibi tasarlanmıştır.

Yönetici

Starter changes:

Start food:  Market ücreti:  yiyecek ücreti:

Start money:  Mağaza ücreti:  eşya ücreti:

Start item:  emlak ücreti:  Upgrade

Start area:

Daily changes:

food:  Build business cost:

money:  Market:

item:  Emlak:  Upgrade

Mağaza:  Upgrade

Mülkler ve sahipleri:

	sahip id	ad	soyad	mülk id	mülk t
1	1	admin	admin	1	emlak
2	1	admin	admin	2	market
3	1	admin	admin	3	magaza
4	1	admin	admin	4	tarla
5	1	admin	admin	5	tarla

Level\_oran

1:

2:

3:  Upg

Database sıfırlama

Database içindeki örnek tablo aşağıdaki gibidir.

	id	emlak_cost	sold_price	rent_price	owner_id
▶	1		9999	9999	1
	2		9999	9999	1
	3		9999	9999	1
	4		200	9999	1
	5		200	9999	1
	6		200	9999	1
	7		200	9999	1
	8		200	9999	1
	9		200	9999	1
	10		200	9999	1
	11		200	9999	1
	12		200	9999	1
	13		200	9999	1
	14		200	9999	1
	15		200	9999	1
	16		200	9999	1
	17		200	9999	1
	18		200	9999	1
	19		200	9999	1
	20		200	9999	1
*	NULL		NULL	NULL	NULL

User bilgilerinin tutulduğu tablo aşağıdaki gibidir.

	id_user	name	surname	password	food	item	money
▶	1	admin	admin	admin	999999	999999	999999
	2	ahmet	sunu	123456	500	500	500
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Genel olarak oluşturulmuş olan tablolar aşağıdaki gibidir.

▶	business_start
▶	business
▶	business_level_oran
▶	business_salary
▶	daily_coast
▶	emlak_cost
▶	game_area
▶	komisyon
▶	rent
▶	show_coast
▶	sold
▶	start
▶	type_business
▶	users

Normalizasyon için:

Aynı tablo içinde tekrarlayan kolonlar kaldırıldı

Her kolonda yalnızca bir değer bulundu

Her satır bir primary key tanımlanmalıdı.

Örnek çıktı:

	idgame_area	type_area	area_users_id
▶	1	1	1
	2	2	1
	3	3	1
	4	4	1
	5	4	1
	6	4	1
	7	4	1
	8	4	1
	9	4	1
	10	4	1
	11	4	1
	12	4	1
	13	4	1
	14	4	1
	15	4	1
	16	4	1
	17	4	1
	18	4	1
	19	4	1
	20	4	1
*	NULL	NULL	NULL

	idtype_business	type	build_cost
▶	1	emlak	250
	2	market	250
	3	magaza	250
	4	tarla	0
*	NULL	NULL	NULL

birbiriyle ilişkisi olan ve sadece 1 kişiye ait olan verilerin silinmesi durumunda o verinin kaybolmaması için:

idbusiness_salary	salary	owner_id	business_level_or
1	50	1	NULL
2	50	1	NULL
3	50	1	NULL
NULL	NULL	NULL	NULL

idbusiness	level	capacity	total_worker	id_owner
1	3	99	0	1
2	3	99	0	1
3	3	99	0	1
NULL	NULL	NULL	NULL	NULL

#### 4. Deneysel Sonuçlar

Kullanıcıdan gerekli bilgiler alındı.

Oyun haritaları başarı ile oluşturuldu.

Kullanıcı giriş işlemleri kayıt işlemleri yapıldı.

Kullanıcının satın alma işe girme işlemleri yapıldı.

Kullanıcının kiralama işlemi tamamlandı ancak fonksiyonelleştirilemedi.

Tarih kavramı eklenemedi.

#### 5. Katkılar

Python kullanımı geliştirildi.

Python kütüphanesi olan pyqt6 öğrenildi.

QtCreator hakkında bilgi edinildi ve kullanıldı.

MySQL öğrenildi ve normalizasyon işlemi gerçekleştirildi.

MySQL ile Python kordinasyonu sağlandı.

```
def set_emlak_rent_sold_price(bid, sold_price, rent_price):
    try:
        cnx = connector()
        cursor = cnx.cursor()
        cursor.execute("""UPDATE emlak_cost SET sold_price=%s,rent_price=%s where business_id=%s """,
                        (sold_price, rent_price, bid))
        cnx.commit()
        cursor.close()
        cnx.close()

    except mysql.connector.Error as error:
        print("Failed to insert into MySQL table {}".format(error))

cnx = connector()
cursor = cnx.cursor()
cursor.execute("""
    INSERT INTO rent
    (rent_time, rent_date, rent_finish_date, id_emlak, id_user, id_
    VALUES (%s, %s, %s, %s, %s, %s)
    """, (days, start_day, finish_day, emlak_id, uid, bid, owner))
cnx.commit()
cursor.close()
cnx.close()
```

#### 6.Yalancı Kod

IF kullanıcı adı soyadı şifre database ile eşleştiyse giriş yap

IF kullanıcı arsaya tıklarsa arsa sayfasını göster eğer arsa kenidisine ait ise yönetici modunda aç

IF kullanıcı mağazaya tıklarsa mağaza sayfasını göster eğer mağaza kenidisine ait ise yönetici modunda aç

IF kullanıcı markete tıklarsa market sayfasını göster eğer market kenidisine ait ise yönetici modunda aç

IF kullanıcı emlakçıya tıklarsa emlak sayfasını göster eğer emlak kenidisine ait ise yönetici modunda aç

IF Satın alma butonlarına tıklanırsa kullanıcıdan para eksilt sahibine parayı aktar emlakçı komisyonunu ver

IF güncelle butonuna tıklanırsa kullanıcıya ait olan değiştirilmesi istenen bilgileri databaseden değiştir

IF admin girişi yapılmışsa yönetici sayfasını aç

IF database sıfırlama butonuna basılmışsa databasede bulunan bütün verileri sıfırla

IF güncelle butonuna basılırsa ilgili kutucuklardaki yazıları databasede değiştir.

IF showmap butonuna basılırsa yönetici modunda haritayı aç

IF yönetici mağazaya tıklarsa mağaza sayfasını yönetici modunda aç

IF yönetici markete tıklarsa market sayfasını yönetici modunda aç

IF yönetici emlakçıya tıklarsa emlak sayfasını ise yönetici modunda aç

IF güncelle butonuna basılırsa gerekli kutucuklardaki bilgileri databasede ilgili yerlerdeki verilerle değiştir.

## 7.Kaynakça

[PyQt5 tooltip on Mouse over - Python \(pythonprogramminglanguage.com\)](https://pythonprogramminglanguage.com/pyqt5-tooltip-on-mouse-over/)

[QSpinBox Class | Qt Widgets 6.5.0](https://www.qt.io/faq/faq-qt-6-5-0-qt-widgets-6-5-0)

[python - PyQt: is there an better way to set objectName in code? - Stack Overflow](https://stackoverflow.com/questions/48112422/python-pyqt-is-there-an-better-way-to-set-objectname-in-code/)

[Python MySQL Insert Into Table \[Complete Guide\] \(pynative.com\)](https://pynative.com/python-mysql-insert-into-table-complete-guide/)

[How to Use MySQL Database in Python? \[A Step by Step Guide\] \(techgeekbuzz.com\)](https://techgeekbuzz.com/how-to-use-mysql-database-in-python-a-step-by-step-guide/)

[How can you perform inner join on two tables using MySQL in Python \(tutorialspoint.com\)](https://tutorialspoint.com/how-can-you-perform-inner-join-on-two-tables-using-mysql-in-python/)