



**T.C**  
**KOCAELİ SAĞLIK VE TEKNOLOJİ ÜNİVERSİTESİ**  
**MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR/YAZILIM MÜHENDİSLİĞİ**

**PROJE KONUSU: Gemi şirketinin operasyonlarını yönetmek için  
bir yazılım geliştirmektir**

**ÖĞRENCİ ADI: Zeynep KEDİKLİ-Nurdan BULUT**  
**ÖĞRENCİ NUMARASI: 220501012-220502021**

**DERS SORUMLUSU:**  
**PROF. DR./DR. ÖĞR. ÜYESİ Ercan ÖLÇER**

**TARİH:04.04.2024**

# 1 GİRİŞ

## 1.1 Projenin amacı

- Bu proje, Gezgin Gemi Şirketi'nin gemi seferlerini, gemi bilgilerini, personel yönetimini ve liman ilişkilerini yönetmek için bir yazılım çözümü geliştirmeyi amaçlar. Yazılım, gemi detaylarını, sefer kayıtlarını, kaptan ve mürettebat bilgilerini, liman detaylarını saklar ve bu verilere erişimi sağlar. Böylece şirketin operasyonlarını daha verimli, düzenli ve izlenebilir hale getirir.

## 1.2 Projede beklenenler

|  |
|--|
| <ul style="list-style-type: none"><li><b>Veri Tabanı Yönetimi:</b><ul style="list-style-type: none"><li>Gerekli tabloların (Gemiler, Seferler, Kaptanlar, Mürettebat, Limanlar) oluşturulması ve ilişkilerin doğru bir şekilde kurulması.</li><li>Veri tabanı işlemlerinin (ekleme, silme, düzenleme) yazılım tarafından yönetilmesi.</li></ul></li></ul>  |
| <ul style="list-style-type: none"><li><b>Sınıf Yapısının Oluşturulması:</b><ul style="list-style-type: none"><li>Her bir varlık için uygun sınıfların (Gemiler, Seferler, Kaptanlar, Mürettebat, Limanlar) oluşturulması.</li><li>Sınıfların gerekli özelliklere ve işlevlere sahip olması.</li></ul></li></ul>  |
| <ul style="list-style-type: none"><li><b>Form Ekranları ve Arayüz Tasarımı:</b><ul style="list-style-type: none"><li>Kullanıcıların verilere erişebileceği ve yönetebileceği form ekranlarının oluşturulması.</li><li>Form ekranlarının kullanıcı dostu ve etkili bir şekilde tasarlanması.</li></ul></li></ul>  |
| <ul style="list-style-type: none"><li><b>Nesne-Veri Tabanı İletişimi:</b><ul style="list-style-type: none"><li>Nesnelerin ve veri tabanının etkili bir şekilde iletişim kurması, veri tabanı işlemlerinin nesneler aracılığıyla gerçekleştirilmesi.</li></ul></li></ul>  |
| <ul style="list-style-type: none"><li><b>Projenin Fonksiyonel Olarak Çalışması:</b><ul style="list-style-type: none"><li>Yazılımın beklenen işlevleri doğru bir şekilde yerine getirmesi: gemi ve sefer bilgilerini saklamak, personel yönetimini sağlamak, liman ilişkilerini düzenlemek.</li><li>Verilerin doğru bir şekilde eklenmesi, düzenlenmesi, silinmesi ve görüntülenmesi.</li></ul></li></ul>                   |
| <ul style="list-style-type: none"><li><b>Güvenlik ve Performans:</b><ul style="list-style-type: none"><li>Veri güvenliğinin sağlanması ve yetkilendirme mekanizmalarının uygulanması.</li><li>Yazılımın performanslı bir şekilde çalışması ve büyük veri miktarlarıyla başa çıkabilmesi.</li></ul></li></ul>   |
| <ul style="list-style-type: none"><li><b>Dökümantasyon ve Raporlama:</b><ul style="list-style-type: none"><li>Projenin kodunu, tasarım kararlarını ve kullanım kılavuzunu içeren kapsamlı bir dökümantasyonun oluşturulması.</li><li>Projenin raporlanması: gereksinimlerin nasıl karşılandığını, hangi teknolojilerin kullanıldığını ve hangi zorlukların aşıldığını açıklayan bir raporun sunulması.</li></ul></li></ul> |

## 2 GEREKSİNİM ANALİZİ

### 2.1 Arayüz gereksinimleri

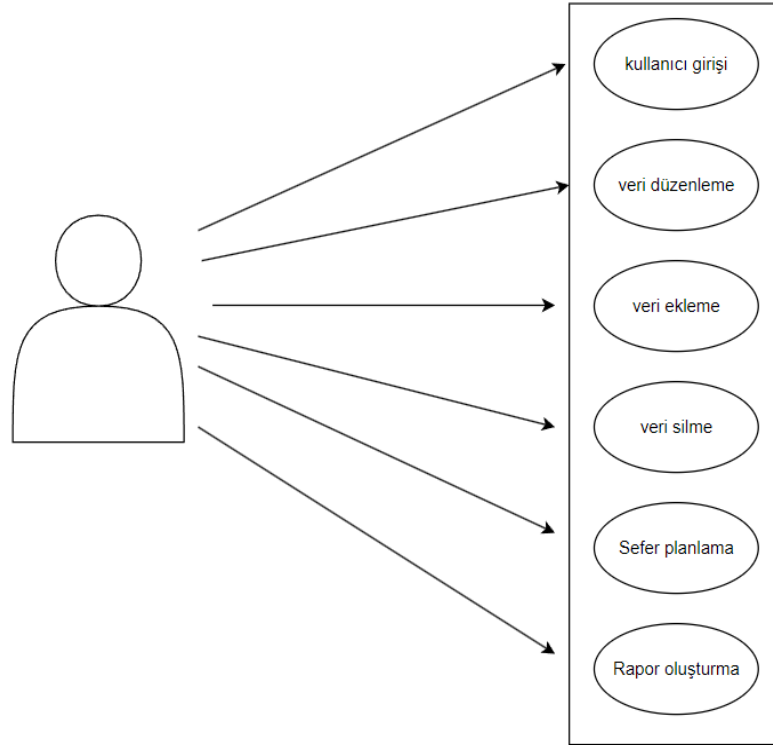
|   |   |
|---|---|
| • | <b>Giriş Ekranı:</b>  |
| ○ | Kullanıcıların sisteme giriş yapabilmeleri için kullanıcı adı ve şifre alanları.                                |
| ○ | Yeni kullanıcıların kayıt olabileceği bir seçenek.  |
| • | <b>Ana Menü:</b>  |
| ○ | Gemiler, seferler, kaptanlar, mürettebat ve limanlar gibi temel veri varlıklarına erişim sağlayan bir ana menü. |
| ○ | Her bir veri varlığı için ekleme, düzenleme, silme ve arama gibi işlevleri içeren alt menüler.                  |
| • | <b>Gemiler Ekranı:</b>  |
| ○ | Gemilere ait detaylı bilgilerin görüntülenebildiği bir ekran.   |
| ○ | Yeni gemi ekleyebilme, mevcut gemileri düzenleyebilme ve silebilme işlevleri.                                   |
| • | <b>Seferler Ekranı:</b>   |
| ○ | Seferlere ait bilgilerin listelendiği ve yönetilebildiği bir ekran.   |
| ○ | Yeni sefer oluşturma, mevcut seferleri düzenleme ve silme işlevleri.  |
| • | <b>Kaptanlar ve Mürettebat Ekranı:</b>  |
| ○ | Kaptanlar ve mürettebatın detaylı bilgilerinin görüntülendiği bir ekran.  |
| ○ | Yeni kaptan ve mürettebat ekleyebilme, mevcut bilgileri düzenleyebilme ve silebilme işlevleri.                  |
| • | <b>Limanlar Ekranı:</b>   |
| ○ | Limanların detaylı bilgilerinin listelendiği bir ekran.   |
| ○ | Yeni liman ekleyebilme, mevcut limanları düzenleme ve silme işlevleri.  |
| • | <b>Sefer Planlama Ekranı:</b>   |
| ○ | Yeni seferler planlamak için kullanılan bir ekran.  |
| ○ | Seferlerin gemi, kaptan, mürettebat ve liman bilgilerini içeren formlar.  |
| • | <b>Raporlar Ekranı:</b>   |
| ○ | Şirketin geçmiş ve gelecek seferleri hakkında raporlar oluşturulabilen bir ekran.                               |
| ○ | Seferlerin tarih aralığına, gemi türüne veya limana göre filtrelenebilir.                                       |

### 2.2 Fonksiyonel gereksinimler

|   |   |
|---|---|
| • | <b>Kullanıcı Girişi ve Yetkilendirme:</b> |
|---|---|

|   |  |
|---|--|
|   | <ul style="list-style-type: none"> <li>○ Kullanıcıların sisteme giriş yapabilmesi ve yetkilendirme işlemlerinin gerçekleştirilmesi.</li> <li>○ Farklı kullanıcı rollerine göre erişim kontrollerinin sağlanması.</li> </ul>                                  |
| • | <b>Veri Ekleme, Düzenleme ve Silme:</b>  |
|   | <ul style="list-style-type: none"> <li>○ Kullanıcıların gemiler, seferler, kaptanlar, mürettebat ve limanlar gibi veri varlıklarını ekleyebilmesi, düzenleyebilmesi ve silebilmesi.</li> <li>○ Veri girişinde doğrulama kontrollerinin yapılması.</li> </ul> |
| • | <b>Sefer Planlama ve Yönetimi:</b>   |
|   | <ul style="list-style-type: none"> <li>○ Kullanıcıların yeni seferler planlayabilmesi ve mevcut seferleri yönetebilmesi.</li> <li>○ Seferlerin gemi, kaptan, mürettebat ve liman bilgileriyle ilişkilendirilmesi.</li> </ul>                                 |
| • | <b>Rapor Oluşturma ve İşleme:</b>  |
|   | <ul style="list-style-type: none"> <li>○ Kullanıcıların geçmiş ve gelecek seferlere ilişkin raporlar oluşturabilmesi.</li> <li>○ Raporların belirli kriterlere göre filtrelenmesi ve sıralanması.</li> </ul>   |
| • | <b>Veri Tabanı Yönetimi:</b>   |
|   | <ul style="list-style-type: none"> <li>○ Veri tabanında bulunan verilerin doğru bir şekilde işlenmesi, saklanması ve güncellenmesi.</li> <li>○ Veri tabanı ile ilgili performans iyileştirmelerinin yapılması.</li> </ul>                                    |

## 2.3 Use-Case diyagramı

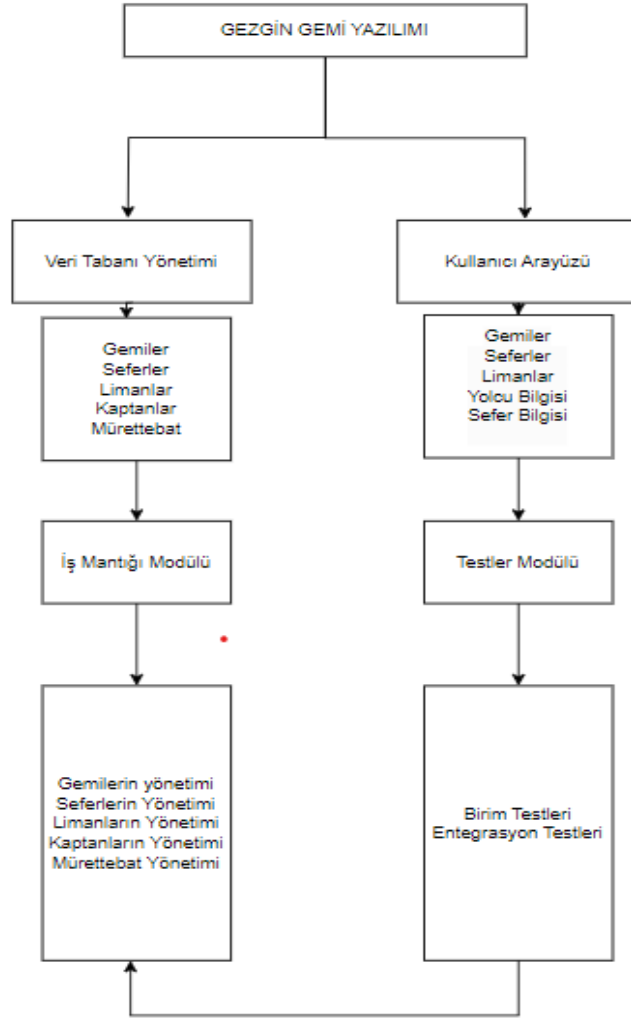


## 3 TASARIM

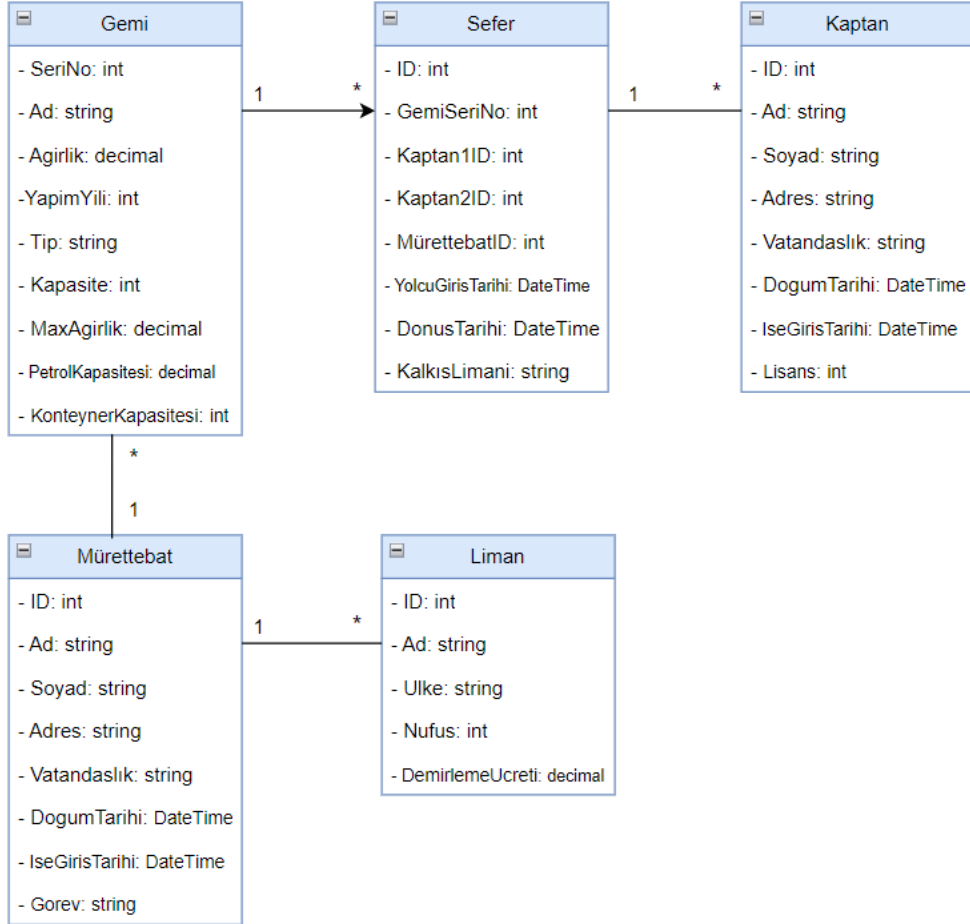
### 3.1 Mimari tasarım

- Gemiler Sınıfı:
  - Bu sınıf, gemilerin özelliklerini temsil eder. Özellikleri, gemilerin seri numarası, adı, ağırlığı, yapım yılı, tipi, kapasitesi, maksimum ağırlığı, petrol kapasitesi ve konteyner kapasitesi gibi bilgileri içerir.
- Kaptanlar sınıfı :
  - Sınıfın özellikleri, kaptanların kimlik numarası, adı, soyadı, adresi, vatandaşlık bilgisi, doğum tarihi, işe giriş tarihi ve sahip oldukları lisans bilgisidir. Bu sınıf, gemi kaptanlarının verilerini saklamak ve işlemek için kullanılır.
- Kullanıcılar Sınıfı
  - Sınıfın özellikleri, kullanıcıların kimlik numarası, kullanıcı adı ve şifresidir. Bu sınıf, kullanıcı hesaplarının verilerini saklamak ve işlemek için kullanılır.
- Limanlar Sınıfı:
  - Sınıfın özellikleri, limanların kimlik numarası, adı, bulunduğu ülke, nüfusu ve demirleme ücretidir. Bu sınıf, limanların verilerini saklamak ve işlemek için kullanılır
- Mürettebatlar Sınıfı
  - Sınıfın özellikleri, mürettebatın kimlik numarası, adı, soyadı, adresi, vatandaşlık bilgisi, doğum tarihi, işe giriş tarihi ve görevidir. Bu sınıf, gemi mürettebatının verilerini saklamak ve işlemek için kullanılır.

- **Modül Diyagramı:**



- **Sınıf Diyagramı:**



### 3.2 Kullanılacak teknolojiler

- Proje C# ve SQL programlama dilleri kullanılarak yazılmıştır.
- **Entity Framework:** Bu, ilişkisel veritabanı tablolarını nesne mantığıyla programlamayı sağlar.
- **DbContext Sınıfı:** Bu sınıf, veritabanı işlemlerini yönetir ve DbSet<T> özellikleri aracılığıyla veritabanı tablolarını temsil eder.
- **LINQ (Language Integrated Query):** LINQ, C# diline entegre edilmiş bir sorgu dili ve standart bir dil genelinde koleksiyonlar üzerinde sorgular yapmayı sağlayan bir teknolojidir.
- **SQL Server:** Veritabanı bağlantı dizesinde belirtilen

"server=LAPTOP-352T58CS\SQLEXPRESS" ifadesi, SQL Server veritabanı sunucusuna bağlanmayı sağlar. Bu kod, bir SQL Server veritabanı ile etkileşim kurar ve veritabanı işlemlerini gerçekleştirir.

- Bu teknolojiler, C# ve .NET platformu üzerinde veritabanı işlemleri yapmak için yaygın olarak kullanılan araçlardır.

### 3.3 Veri tabanı tasarımı

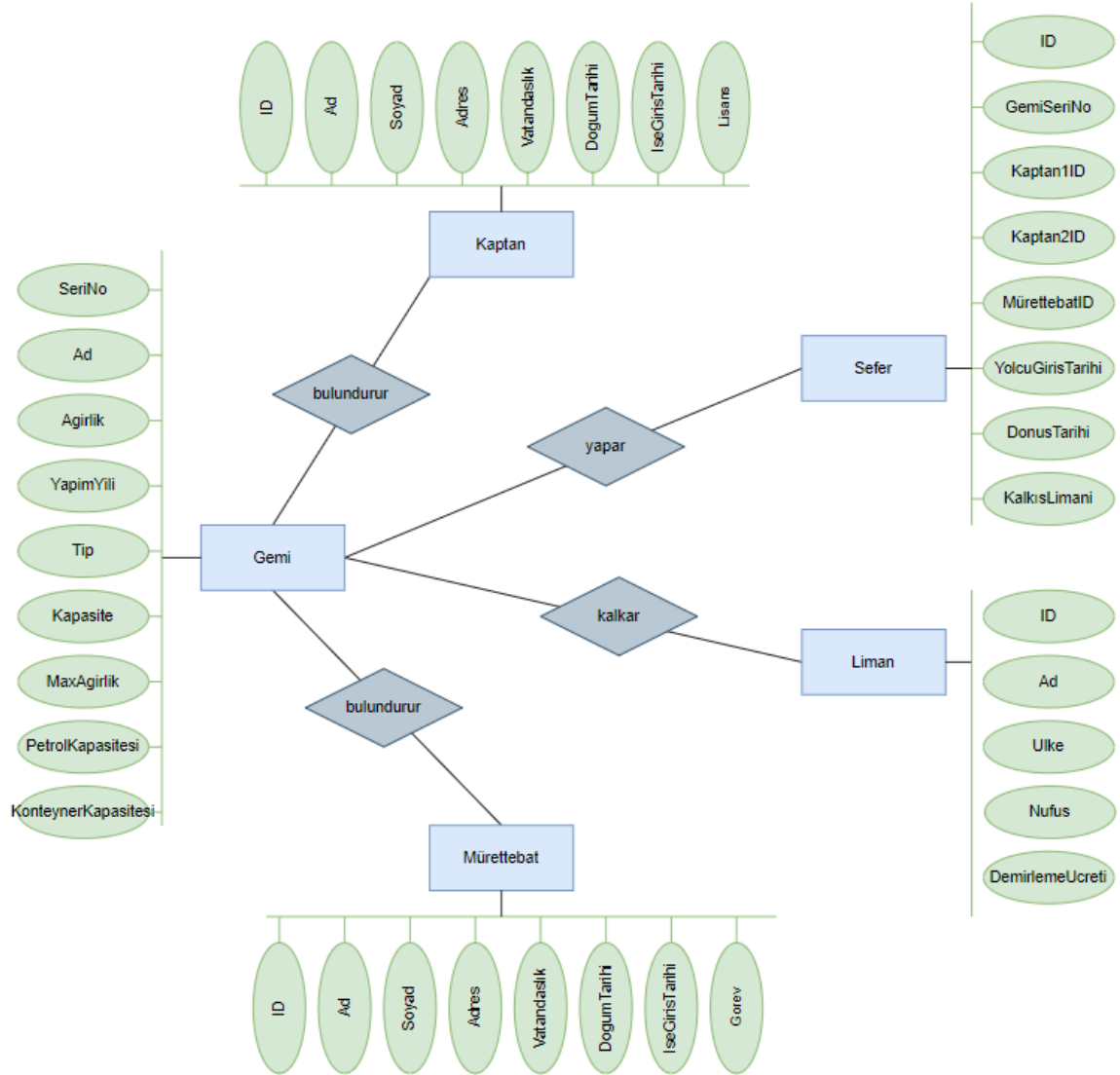
MS SQL server kullanılmıştır

1. Gemiler Tablosu (Gemiler):
  - Gemilerin kaydedildiği tablodur.
  - Her gemi için benzersiz bir ID alanı bulunur.
  - Geminin adı (Ad), ağırlığı (Ağırlık), yapım yılı (YapımYılı), tipi (Tip), kapasitesi (Kapasite), maksimum ağırlığı (MaxAğırlık), petrol kapasitesi (PetrolKapasitesi) ve konteyner kapasitesi (KonteynerKapasitesi) gibi özelliklerini içerir.
2. Kaptanlar Tablosu (Kaptanlar):
  - Kaptanların kaydedildiği tablodur.
  - Her kaptan için benzersiz bir ID alanı bulunur.
  - Kaptanın adı (Ad), soyadı (Soyad), adresi (Adres), vatandaşlık bilgisi (Vatandaşlık), doğum tarihi (DogumTarihi), işe giriş tarihi (IseGirisTarihi) ve lisansı (Lisans) gibi özelliklerini içerir.
3. Kullanıcılar Tablosu (Kullanıcılar):
  - Uygulamanın kullanıcılarını ve bunların giriş bilgilerini saklar.
  - Her kullanıcı için benzersiz bir ID alanı bulunur.
  - Kullanıcı adı (Username) ve şifre (Password) gibi giriş bilgilerini içerir.
4. Limanlar Tablosu (Limanlar):
  - Limanların kaydedildiği tablodur.
  - Her liman için benzersiz bir ID alanı bulunur.
  - Limanın adı (Ad), bulunduğu ülke (Ulke), nüfusu (Nufus), pasaport gerekliliği (PasaportGerekli) ve demirleme ücreti (DemirlemeUcreti) gibi özelliklerini içerir.
5. Mürettebat Tablosu (Mürettebat):
  - Gemi mürettebatının kaydedildiği tablodur.
  - Her mürettebat üyesi için benzersiz bir ID alanı bulunur.
  - Mürettebat üyesinin adı (Ad), soyadı (Soyad), adresi (Adres), vatandaşlık bilgisi (Vatandaşlık), doğum tarihi (DogumTarihi), işe giriş tarihi (IseGirisTarihi) ve görevi (Gorev) gibi özelliklerini içerir.
6. Seferler Tablosu (Seferler):
  - Gemilerin seferlerinin kaydedildiği tablodur.
  - Her sefer için benzersiz bir ID alanı bulunur.
  - Seferin yapıldığı geminin seri numarası (GemiSeriNo), sefere katılan



kaptanların (Kaptan1ID, Kaptan2ID), mürettebatın (MurettebatID), yolcu giriş tarihi (YolcuGirisTarihi), dönüş tarihi (DonusTarihi) ve kalkış limanı (KalkisLimani) gibi özelliklerini içerir.

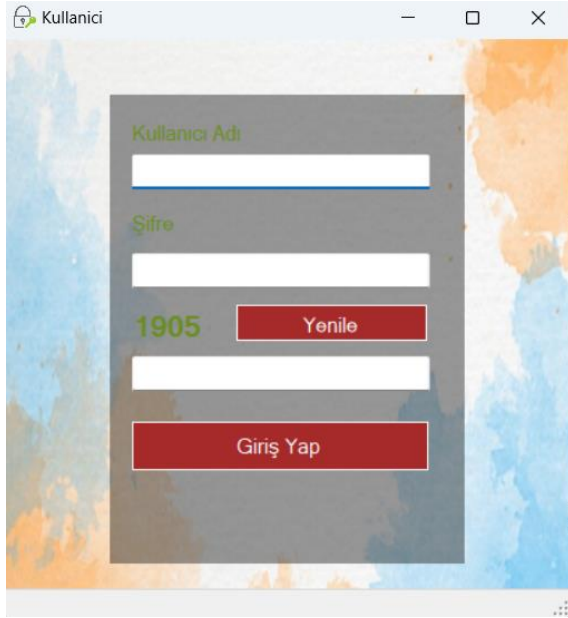
## ER Diyagramı



### 3.4 Kullanıcı arayüzü tasarımı

Kullanıcı arayüzü Windows Form uygulaması ile oluşturulmuştur. Veritabanıyla etkileşime geçerek kullanıcıya veri girişi, düzenleme ve silme gibi işlemleri gerçekleştirmesine olanak sağlar.

#### Kullanıcı Girişi



#### Gemi



#### Kaptan

Kaptan

|         |                   |                    |                    |
|---------|-------------------|--------------------|--------------------|
| ID :    | 1                 | Vatandaslik :      | Türk               |
| Ad :    | Ahmet             | Dogum Tarihi :     | 15.05.1980         |
| Soyad : | Yilmaz            | İşe Giriş Tarihi : | 20.01.2005         |
| Adres : | İstanbul, Türkiye | Lisans :           | Denizcilik Lisansı |

Kayıt Listele Kayıt Ekle Kayıt Güncelle Kayıt Sil Temizle

Kaptanlar Listeleniyor!

## Liman

Liman

|        |                |                    |          |
|--------|----------------|--------------------|----------|
| ID :   | 1              | Nüfus :            | 2300000  |
| Ad :   | Houston Limanı | Demirleme Ücreti : | 15000.00 |
| Ülke : | ABD            |                    |          |

Kayıt Listele Kayıt Ekle Kayıt Güncelle Kayıt Sil Temizle

Limanlar Listeleniyor!

## Sefer

Sefer

|                |   |                      |            |
|----------------|---|----------------------|------------|
| ID :           | 1 | Murettebat ID :      | 1          |
| Gemi Seri No : | 1 | Yolcu Giriş Tarihi : | 10.04.2024 |
| Kaptan1 ID :   | 1 | Dönüş Tarihi :       | 20.04.2024 |
| Kaptan2 ID :   | 2 | Kalkış Limanı :      | İstanbul   |

Kayıt Listele Kayıt Ekle Kayıt Güncelle Kayıt Sil Temizle

Seferler Listeleniyor!

## Mürettebat

Mürettebat

ID :

Ad :

Soyad :

Adres :

Vatandaşlık :

Doğum Tarihi :

İşe Giriş Tarihi :

Görev :

Kayıt Listele Kayıt Ekle Kayıt Güncelle Kayıt Sil Temizle

- Uygulama ms sql server ve visual studio indirilip proje kurulumu yapıldıktan sonra kullanılabilir.

## 4 UYGULAMA

### 4.1 Kodlanan bileşenlerin açıklamaları

- **MyNewMigration Sınıfı** : Bu sınıf, veritabanı migrasyonlarını tanımlar. Entity Framework, bu sınıftaki Up() ve Down() metodlarını kullanarak veritabanı şemasını günceller ve geri alır.
  - Up() Metodu: Bu metod, veritabanı şemasını güncellemek için kullanılır. Bu migrasyon sürümünde, CreateTable() metodu aracılığıyla yeni tablolar oluşturulur. Her bir CreateTable() çağrısı, veritabanına yeni bir tablo ekler. ForeignKey() ve Index() çağrıları, tablolar arasında ilişkileri ve dizinleri tanımlar.
  - Down() Metodu: Bu metod, bir önceki migrasyon sürümüne geri dönmek için kullanılır. Bu yöntemde, veritabanındaki değişiklikler geri alınır. DropTable() metodu, veritabanından belirtilen tabloları kaldırır. DropIndex() ve DropForeignKey() çağrıları, indeksleri ve dış anahtar kısıtlamalarını kaldırır.
- **Configuration Sınıfı**: DbMigrationsConfiguration sınıfından kalıtım alarak migrasyon ayarlarını içerir.
  - Seed() Metodu: Veritabanına başlangıç verileri eklemek için kullanılır. Bu örnekte herhangi bir işlem gerçekleştirilmemiştir.
- **GemiContext sınıfı**: Entity Framework kullanarak veritabanı işlemlerini yönetmek için tasarlanmıştır. Bu sınıf, DbContext sınıfından türetilmiştir ve aşağıdaki özelliklere sahiptir:
  - Constructor : Veritabanı bağlantısını yapılandırır. Bağlantı dizesi, SQL Server veritabanına bağlanmak için gerekli bilgileri içerir.
  - DbSet Özellikleri: Her bir özellik, bir veritabanı tablosunu temsil eder. DbSet özellikleri, veritabanındaki ilgili tablolara erişim sağlar.

- **Gemiler Sınıfı:** gemilerin özelliklerini barındırarak gemi verilerinin tutulmasını sağlar. Her bir gemi için bir nesne oluşturulabilir ve bu özellikler üzerinden gemi verileri işlenebilir.
- **Kaptanlar Sınıfı:** gemi kaptanlarının özelliklerini barındırarak kaptan verilerinin tutulmasını sağlar. Her bir kaptan için bir nesne oluşturulabilir ve bu özellikler üzerinden kaptan verileri işlenebilir.
- **Kullanıcılar Sınıfı:** Bu sınıf, sisteme giriş yapacak kullanıcıların bilgilerini içerir.
  - **ID:** Kullanıcının benzersiz kimlik numarasını temsil eden bir tamsayı.
  - **Username:** Kullanıcının kullanıcı adını temsil eden bir dize.
  - **Password:** Kullanıcının şifresini temsil eden bir dize.
- **Limanlar Sınıfı:** limanların özelliklerini barındırarak liman verilerinin tutulmasını sağlar. Her bir liman için bir nesne oluşturulabilir ve bu özellikler üzerinden liman verileri işlenebilir.
- **Murettebat Sınıfı:** gemi mürettebatının özelliklerini barındırarak mürettebat verilerinin tutulmasını sağlar. Her bir mürettebat üyesi için bir nesne oluşturulabilir ve bu özellikler üzerinden mürettebat verileri işlenebilir.
- **Program Sınıfı:** Bu sınıf, uygulamanın giriş noktasını temsil eder.
  - **Main Metodu:** Bu metod, uygulamanın başlangıcında çalıştırılan giriş noktasıdır. Bu metod, uygulamanın ana iş mantığını başlatır.
  - **[STAThread]:** Bu nitelik, uygulamanın bir STA (Single-Threaded Apartment) thread modeli kullanacağını belirtir. Bu, uygulamanın birden çok iş parçacığı arasında uyumsuzluk yaşamadan çalışacağı anlamına gelir.
  - **EnableVisualStyles Metodu:** Bu metod, Windows Görünüm Stillerinin etkinleştirilmesini sağlar. Bu, uygulamanın görsel olarak daha çekici görünmesini sağlar.
  - **SetCompatibleTextRenderingDefault Metodu:** Bu metod, uygulamanın uyumlu metin oluşturma ayarlarını belirler. Bu ayarlar, Windows Forms kontrolleri tarafından kullanılan varsayılan metin oluşturma yöntemlerini ayarlar.
  - **Run Metodu:** Bu metod, belirtilen form nesnesini çalıştırarak uygulamanın ana döngüsünü başlatır. Bu şekilde, belirtilen formun görüntülenmesi ve kullanıcı etkileşiminin alınması sağlanır.
- **Seferler Sınıfı:** gemi seferlerinin temel özelliklerini ve ilgili nesneler arasındaki ilişkileri tanımlar. Bu sayede, gemi seferlerinin veritabanında saklanması ve ilişkilendirilmiş diğer varlıklarla etkileşimi sağlanır.
  - **Gemi, Kaptan1, Kaptan2 ve Murettebat Özellikleri:** Bu özellikler, Seferler sınıfının Gemiler, Kaptanlar ve Murettebat sınıflarıyla ilişkili olduğunu belirtir. ForeignKey niteliği, bu özelliklerin ilgili sınıflar arasında dış anahtar ilişkisi olduğunu belirtir.

## 4.2 Görev dağılımı

- Projede pair programming yapılmıştır.
- Raporun 1, 2, 3.1 ve 3.2 maddeleri Zeynep kedikli tarafından yazılmıştır.
- Raporun 3.3, 3.4, 4 ve 5. Maddeleri Nurdan Bulut tarafından yapılmıştır

### **4.3 Karşılaşılan zorluklar ve çözüm yöntemleri**

- Ms sql server kurulumunda, visual studio 2022 ve sql bağlantısında,Windows form tabloları oluştururken, veri tabanı bağlantısında sıkıntı ve zorluklar yaşandı youtube, udemy ve chatgpt' ye başvurularak çözülmeye çalışılmıştır.
- Öncesinde veri tabanı ile ilgili proje yapmadığımız için proje planını yaparken ve geliştirirken çeşitli zorluklar yaşadık.

### **4.4 Proje isterlerine göre eksik yönler**

- Projede eksik ister bulunmamaktadır.

## **5 TEST VE DOĞRULAMA**

### **5.1 Yazılımın test süreci**

- Yazılım manuel olarak test edilmiştir. Tüm ister ve özelliklerin manuel testleri yapılmış ve sorunsuz çalışmaktadır.

### **5.2 Yazılımın doğrulanması**

- Manuel test yapılırken herhangi bir hata ile karşılaşılmaı. Gemi, Kaptan, Liman, Sefer ve Mürettebat sekmelerine sorunsuz şekilde kayıt eklenebilir, güncellenebilir ve silinebilir durumdadır.

NURDAN BULUT

<https://github.com/nurdanbulut>

ZEYNEP KEDİKLİ

<https://github.com/Zeynepkedikli>