



T.C
KOCAELİ SAęLIK VE TEKNOLOJİ ÜNİVERSİTESİ
MÜHENDİSLİK VE DOęA BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR/YAZILIM MÜHENDİSLİęİ

PROJE KONUSU:
LİMAN OTOMASYONU

YAVUZHAN KURŞUN
230502047
İBRAHİM KEÇİCİ
230502049

DERS SORUMLUSU:
PROF. DR. NEVCİHAN DURU

02.05.2024

1 GİRİŞ

1.1 Projenin amacı

Projenin amacı, Gezin Gemi Şirketi'nin seferlerini ve gemi filosunu yönetmek için bir yazılım geliştirmektir. Bu yazılım, gemiler, seferler, limanlar, kaptanlar ve mürettebat gibi çeşitli varlıkların verilerini yönetmeyi amaçlamaktadır.

Projenin amacı kısaca şunları içerir:

Gemi Bilgilerinin Yönetimi:

Yolcu gemileri, petrol tankerleri ve konteyner gemileri gibi farklı türlerdeki gemilerin seri numarası, adı, ağırlığı, yapım yılı, kapasiteleri gibi bilgilerin yönetilmesi.

Sefer Yönetimi:

Her bir seferin kaydedilmesi, seferin hangi gemi tarafından gerçekleştirildiğinin belirlenmesi, seferin başlangıç ve bitiş tarihleri ile uğranan limanların kaydedilmesi.

Liman Bilgilerinin Yönetimi:

Liman adı, ülkesi, nüfusu, pasaport gerekliliği, demirleme ücreti gibi bilgilerin yönetilmesi.

Kaptan ve Mürettebat Yönetimi:

Kaptanların ve mürettebatın kişisel bilgilerinin (ID, ad, soyad, adres, vb.) yanı sıra işe giriş tarihi, lisans bilgileri, görevleri gibi bilgilerin yönetilmesi.

Veritabanı ve Form Ekranları Arasında İletişim:

Verilerin form ekranları üzerinden eklenmesi, silinmesi, düzenlenmesi gibi işlemlerin veritabanına doğru şekilde iletilmesi ve yönetilmesi.

Nesne Tabanlı Programlama İle Veri Yönetimi:

Gerekli sınıfların oluşturulması ve bu sınıflar üzerinden verilerin yönetilmesi.

Veritabanı Tasarımı:

Veritabanı tablolarının oluşturulması ve verilerin doğru bir şekilde saklanmasının sağlanması.

Bu amaçlar doğrultusunda, yazılımın gemi şirketinin operasyonlarını verimli bir şekilde yönetmesi ve seferlerin, gemilerin ve personelin verimli bir şekilde planlanması ve takip edilmesi sağlanacaktır.

2 GEREKSİNİM ANALİZİ

2.1 Arayüz gereksinimleri

Gemi Ekleme, Düzenleme ve Silme Ekranları:

Kullanıcıların yeni gemi bilgilerini ekleyebileceği, mevcut gemi bilgilerini düzenleyebileceği ve varsa gemileri silebileceği ekranlar.

Sefer Planlama Ekranı:

Kullanıcıların yeni seferleri planlayabileceği, mevcut seferleri görüntüleyebileceği ve düzenleyebileceği ekran.

Liman Yönetimi Ekranı:

Liman bilgilerini eklemek, düzenlemek ve silmek için kullanılabilir bir arayüz.

Kaptan ve Mürettebat Yönetimi Ekranı:

Kullanıcıların kaptan ve mürettebat bilgilerini yönetebileceği bir arayüz.

Sefer Detayları Ekranı:

Belirli bir seferin detaylarını görüntülemek ve düzenlemek için kullanılacak bir ekran.

Raporlama Ekranları:

Kullanıcıların gemi filosu, planlanmış ve gerçekleşmiş seferler, liman bilgileri, personel durumu gibi verilere erişebileceği raporlama ekranları.

2.2 Fonksiyonel gereksinimler

Gemi Ekleme Fonksiyonu:

Kullanıcıların yeni gemi bilgilerini sisteme ekleyebilmesi.

Gemi Düzenleme Fonksiyonu:

Varolan gemi bilgilerini güncelleyebilme yeteneği.

Gemi Silme Fonksiyonu:

Sistemden gemi kayıtlarını silme yeteneği.

Sefer Planlama Fonksiyonu:

Kullanıcıların yeni seferleri planlayabilmesi ve bu seferlere gemi atayabilmesi.

Sefer Düzenleme ve İptal Etme Fonksiyonu:

Var olan seferleri düzenleyebilme ve iptal edebilme yeteneği.

Liman Ekleme, Düzenleme ve Silme Fonksiyonları:

Liman bilgilerini sisteme ekleyebilme, güncelleyebilme ve silebilme yeteneği.

Kaptan ve Mürettebat Yönetimi Fonksiyonları:

Kaptan ve mürettebat bilgilerini sisteme ekleyebilme, güncelleyebilme ve silebilme yeteneği.

Sefer Detaylarını Görüntüleme ve Düzenleme Fonksiyonu:

Belirli bir seferin detaylarını görüntüleme ve düzenleme yeteneği.

Raporlama Fonksiyonları:

Kullanıcıların çeşitli raporları görüntüleyebilmesi, örneğin gemi filosu raporu, planlanmış ve gerçekleşmiş seferler raporu, liman bilgileri raporu, personel durumu raporu vb.

Kullanıcı Yetkilendirme ve Kimlik Doğrulama Fonksiyonu:

Kullanıcıların sisteme giriş yapabilmesi ve yetkilerine göre belirli işlemleri gerçekleştirebilmesi için kimlik doğrulama ve yetkilendirme mekanizması.

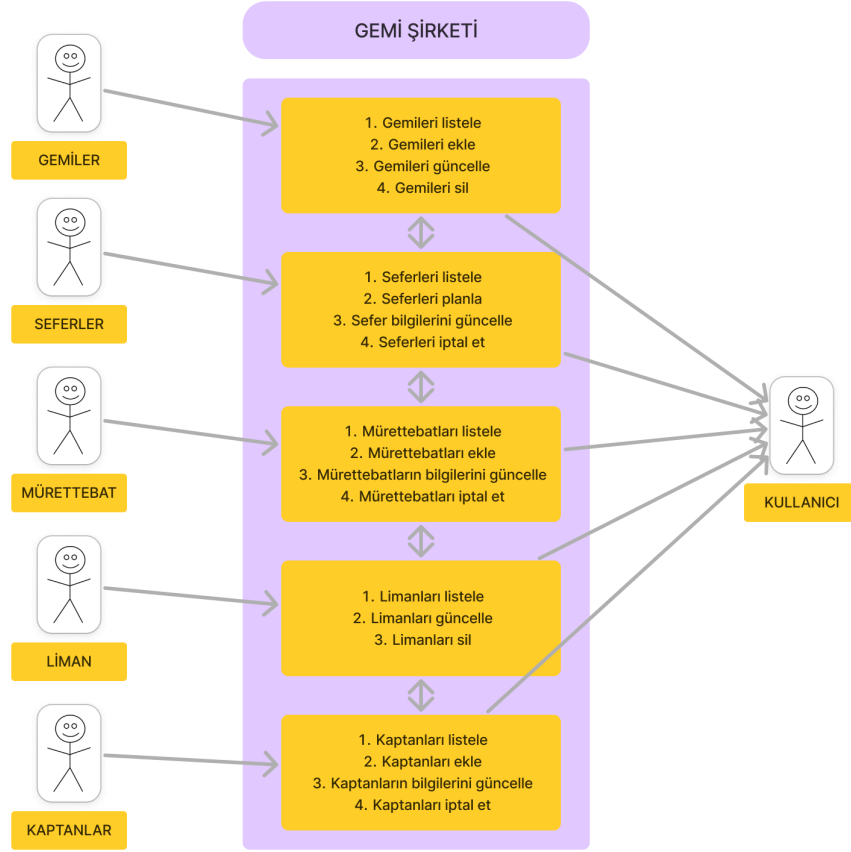
Veri Tabanı Yönetimi Fonksiyonları:

Veri tabanı bağlantısını yönetme, veri tabanı işlemlerini gerçekleştirme (ekleme, güncelleme, silme, sorgulama vb.) gibi işlevlerin gerçekleştirilmesi.

Hata Yönetimi Fonksiyonları:

Kullanıcıların karşılaştığı hataların yönetilmesi ve kullanıcıya uygun geri bildirimlerin sağlanması.

2.3 Use-Case diyagramı



3 TASARIM

3.1 Mimari tasarım

Katmanlı Mimari:

Projede katmanlı bir mimari kullanılabilir. Bu, kullanıcı arayüzü, iş mantığı ve veritabanı erişim katmanlarını içerir. Bu katmanlar arasında sıkı bir ayırım sağlanır ve her bir katmanın belirli bir sorumluluğu vardır.

Kullanıcı Arayüzü Katmanı:

Kullanıcı arayüzü, kullanıcıların sistemle etkileşimde bulunduğu katmandır. Bu katmanda, kullanıcıların veri girişi yapabileceği formlar, raporlama ekranları ve diğer görsel öğeler bulunur.

İş Mantığı Katmanı:

İş mantığı katmanı, uygulamanın iş kurallarını, veri işleme mantığını ve iş süreçlerini içerir. Bu katman, kullanıcı arayüzünden gelen istekleri işler, verileri işler ve gerekli işlemleri gerçekleştirir.

Veritabanı Erişim Katmanı:

Veritabanı erişim katmanı, veritabanı ile iletişim kurmak için gereken işlevleri sağlar. Bu katman, veritabanıyla doğrudan etkileşim kurar, veri tabanı işlemlerini gerçekleştirir ve iş mantığı katmanına veri sağlar.

Nesne Tabanlı Programlama:

Projede nesne tabanlı programlama prensipleri kullanılabilir. Her bir varlık için bir sınıf oluşturulabilir ve bu sınıfların birbirleriyle etkileşimi sağlanabilir.

Veritabanı Tasarımı:

Veritabanı, varlık ve ilişkilerin doğru bir şekilde modellediği ve verilerin tutulduğu bir yerdir. Projede, veritabanı tablolarının oluşturulması ve ilişkilerin tanımlanması gerekmektedir.

3.2 Kullanılacak teknolojiler

Yazılım python programlama dili ile kodlanmıştır. Pythonda arayüz tasarımı için tkinter kütüphanesi kullanılmıştır. Ayrıca veritabanı bağlantısı için psycopg2 kütüphanesi kullanılmıştır. Veritabanı olarak PostgreSQL kullanılmıştır.

3.3 Veri tabanı tasarımı

Gemi Tablosu:

Gemiye ait bilgilerin tutulduğu tablo. Seri numarası, adı, ağırlığı, yapım yılı gibi bilgiler içerir. Ayrıca gemi türüne göre özel bilgiler de bulunabilir (yolcu kapasitesi, petrol kapasitesi, konteyner kapasitesi vb.).

Sefer Tablosu:

Gemilerin yapacağı seferlerin kaydedildiği tablo. Her seferin bir ID'si, yola çıkış tarihi, dönüş tarihi, yola çıkış limanı gibi bilgiler içerir.

Liman Tablosu:

Seferlerin uğrayabileceği limanların bilgilerinin tutulduğu tablo. Liman adı, ülkesi, nüfusu, pasaport gerekliliği, demirleme ücreti gibi bilgiler içerir.

Kaptan Tablosu:

Kaptanların kişisel ve lisans bilgilerinin tutulduğu tablo. ID, ad, soyad, adres, vatandaşlık, doğum tarihi, işe giriş tarihi gibi bilgiler içerir.

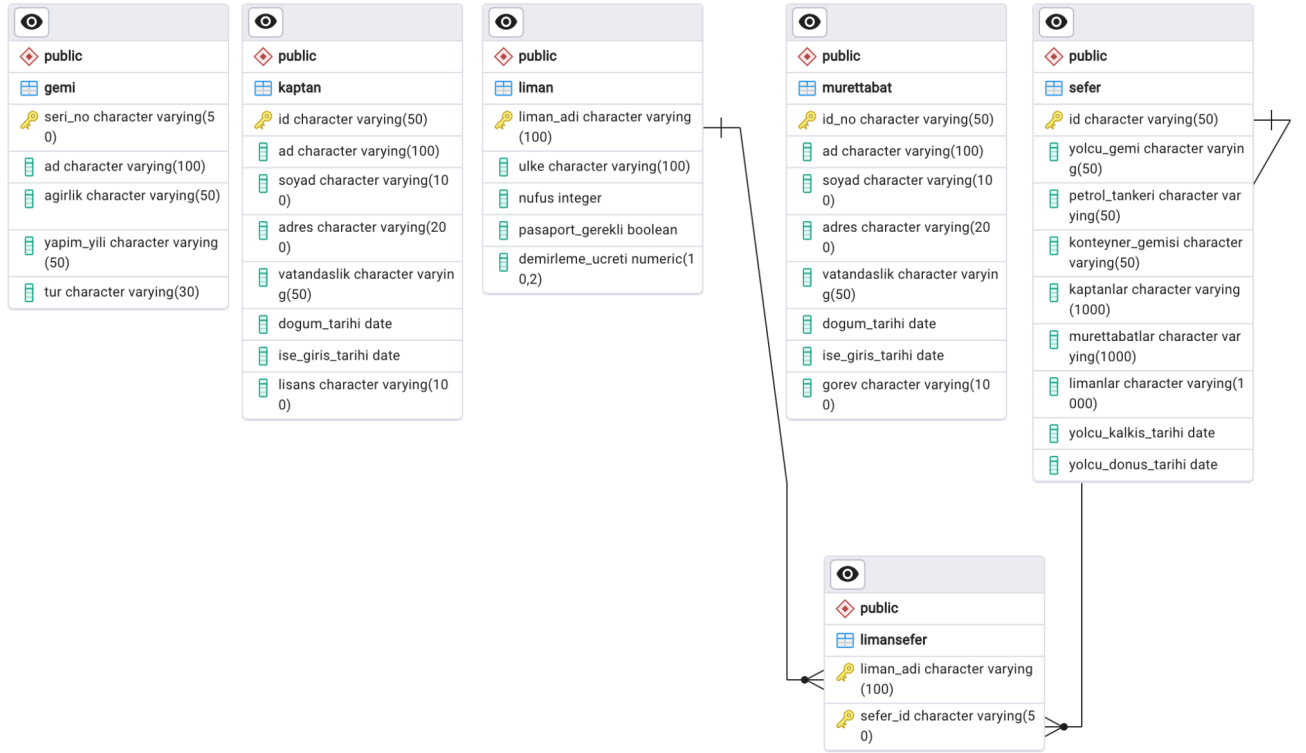
Mürettebat Tablosu:

Mürettebatın kişisel bilgilerinin ve görevlerinin tutulduğu tablo. ID, ad, soyad, adres, vatandaşlık, doğum tarihi, işe giriş tarihi, görev gibi bilgiler içerir.

Sefer-Liman İlişkisi:

Seferlerin hangi limanlara uğradığını gösteren ilişki tablosu.

ER Diyagramı:



3.4 Kullanıcı arayüzü tasarımı

- Kullanıcı arayüzü tasarımı ile ilgili açıklama.
- Yazılımdan ekran çıktıları alınarak üzerinden açıklama yapılması
- Uygulamanın nasıl çalıştırılacağı ile ilgili açıklama

4 UYGULAMA

4.1 Kodlanan bileşenlerin açıklamaları

Gemi Sınıfı:

Bu sınıf, gemi varlığını temsil eder. Geminin özellikleri (seri numarası, adı, ağırlığı, yapım yılı vb.) ve gemi türüne özgü bilgiler (yolcu kapasitesi, petrol kapasitesi, konteyner kapasitesi vb.) bu sınıfta tutulur. Ayrıca gemilerin yönetimi için gerekli işlevler (ekleme, düzenleme, silme vb.) bu sınıfta tanımlanabilir.

Sefer Sınıfı:

Bu sınıf, sefer varlığını temsil eder. Seferin özellikleri (ID, yola çıkış tarihi, dönüş tarihi, yola çıkış limanı vb.) bu sınıfta tutulur. Ayrıca seferlerin yönetimi için gerekli işlevler (sefer planlama, düzenleme, iptal vb.) bu sınıfta tanımlanabilir.

Liman Sınıfı:

Bu sınıf, liman varlığını temsil eder. Limanın özellikleri (adı, ülkesi, nüfusu, pasaport gerekliliği, demirleme ücreti vb.) bu sınıfta tutulur. Ayrıca limanların yönetimi için gerekli işlevler (ekleme, düzenleme, silme vb.) bu sınıfta tanımlanabilir.

Kaptan Sınıfı:

Bu sınıf, kaptan varlığını temsil eder. Kaptanın özellikleri (ID, adı, soyadı, lisans bilgileri vb.) bu sınıfta tutulur. Ayrıca kaptanların yönetimi için gerekli işlevler (ekleme, düzenleme, silme vb.) bu sınıfta tanımlanabilir.

Mürettebat Sınıfı:

Bu sınıf, mürettebat varlığını temsil eder. Mürettebatın özellikleri (ID, adı, soyadı, görevi vb.) bu sınıfta tutulur. Ayrıca mürettebatın yönetimi için gerekli işlevler (ekleme, düzenleme, silme vb.) bu sınıfta tanımlanabilir.

Veritabanı Yönetim Modülü:

Bu modül, veritabanı ile iletişim kurmak ve veri tabanı işlemlerini gerçekleştirmek için gereken işlevleri içerir. Veritabanına erişim, veri ekleme, düzenleme, silme, sorgulama gibi işlemler bu modülde gerçekleştirilir.

Kullanıcı Arayüzü Modülü:

Bu modül, kullanıcı arayüzü ile etkileşim kurmak için gereken işlevleri içerir. Kullanıcıların veri girişi yapabileceği formlar, raporlama ekranları ve diğer görsel öğeler bu modülde bulunur. Bu modül, kullanıcıların sistemi kolayca kullanabilmesini sağlar.

4.2 Görev dağılımı

Rapor ve ödevi beraber ortak hazırladık. Kimi zaman buluşup yaptık kimi zaman internet üzerinden konuşarak.

4.3 Karşılaşılan zorluklar ve çözüm yöntemleri

Uygulama arayüzünü tasarlarken biraz zorlandık. Bunu internetten yardım alarak çözdük. Ayrıca veritabanı ve arayüz bağlantısı kısmında da bazı hatalar aldık. Yaptığımız araştırmalar sonucu psycopg2 kütüphanesini bulduk ve PostgreSQL ile bağlantıyı sağladık.

Kaynakça

- <https://www.youtube.com/watch?v=Fv82RX4cWW4>
- <https://www.geeksforgeeks.org/python-gui-tkinter/>

Github

- https://github.com/yavuzhankursun/Liman_isletmesi
- <https://github.com/ibrahimibo7/gemi-sirketi>