

PROJE II Baggage Security Simulator

Projenin Amacı Nedir?

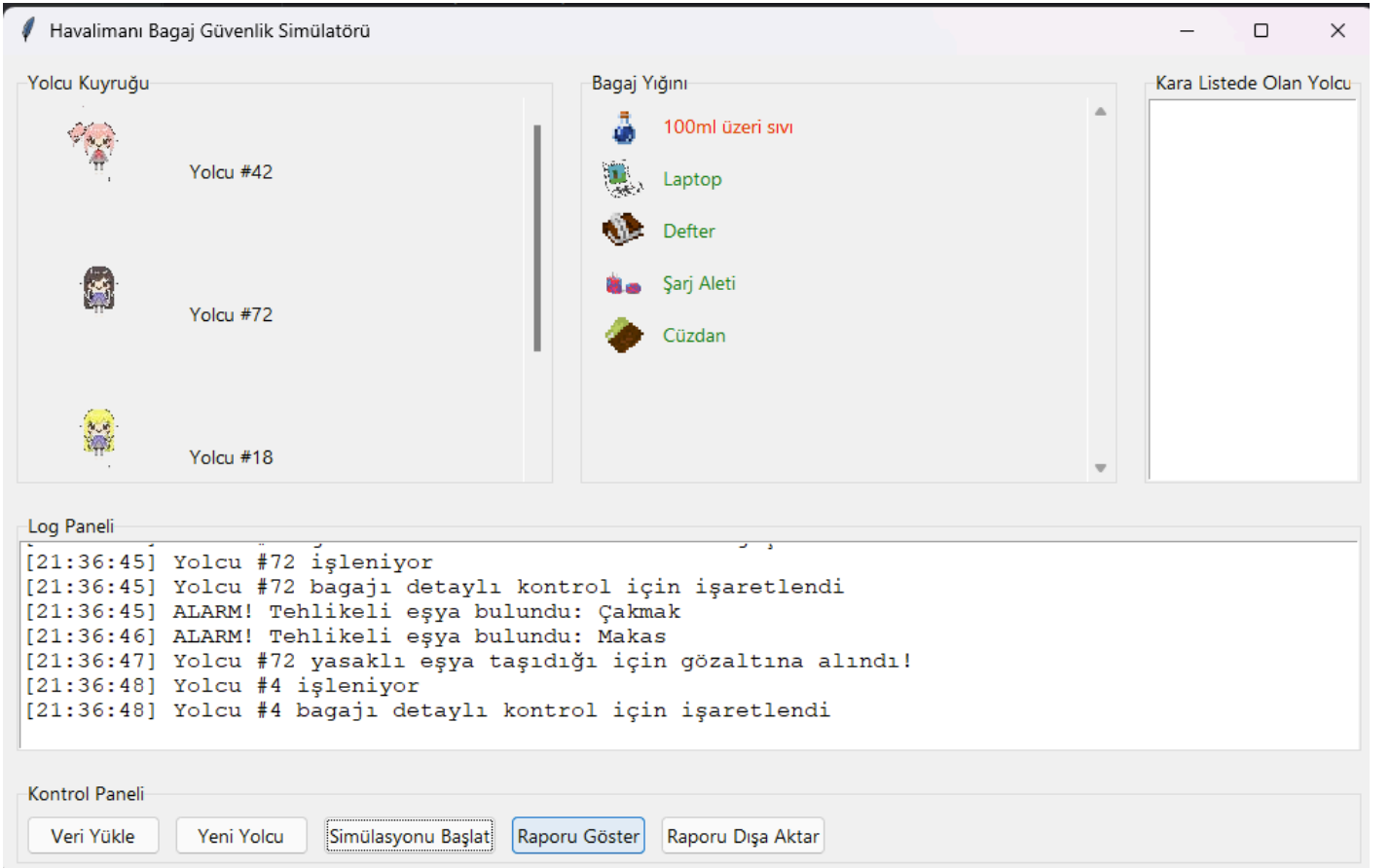
Havalimanında bagaj güvenliği, hem yolcu hem de uçuş güvenliği açısından büyük bir öneme sahiptir. Birçok yolcu, uçak seyahati hakkında yeterli bilgiye sahip olmayabilir ve bilmeden tehlikeli eşyalar taşıyabilir. Örneğin, parfüm şişesi gibi günlük yaşamda zararsız görünen nesneler, uçuş güvenliği için tehlike oluşturabilir. Ayrıca, bagaj güvenliği, sadece uçuşları güvence altına almakla kalmaz, diğer yolcuların güvenliğini sağlamak adına da kritik bir rol oynar.

Bu projede, bagaj güvenliğini minimalist bir şekilde ele aldık ve süreci görsel olarak kullanıcıya sunmayı amaçladık. Pixel art desteğiyle yolcuları ve eşyaları görselleştirerek, bagaj güvenliğini simüle ettik. Bu sayede, güvenlik sürecini hem görsel hem de fonksiyonel olarak etkili bir biçimde modelledik. Proje, hem kullanıcı dostu bir arayüz sunarak güvenlik sürecini izlemeyi kolaylaştırıyor, hem de bagaj güvenliğine dair önemli adımları basit ama etkili bir şekilde simüle ediyor.

Kullanıcı Nasıl Kullanır?

Veri Yükle Butonu:

- Kullanıcı, “Veri Yükle” butonuna tıkladığında, sistem rastgele yolcu profilleri ve bagaj içerikleri oluşturur. Bu yolcuların bagajlarında, her türlü tehlikeli veya güvenli eşya rastgele seçilir ve sisteme eklenir. Bu, bagaj güvenliği sürecinin simüle edilmesi için başlangıç adımıdır.



Simülasyonu Başlat Butonu:

- Kullanıcı, “Simülasyonu Başlat” butonuna tıklayarak simülasyonu başlatır. Bu işlemle birlikte, yolcuların görsel temsilleri (pixel art) ve bagajlarının içeriği ekranda gösterilir.
- Güvenli eşyalar yeşil renkte, tehlikeli eşyalar ise kırmızı renkte gösterilir. Bu sayede, kullanıcı tehlikeli eşyaların hangileri olduğunu kolayca görür.
- Ayrıca, yolcuların bagajlarındaki tehlikeli eşyalar, sistemin güvenlik kontrolüne takılmasına neden olur. Güvenlik kontrolü sonucunda, tehlikeli eşya tespit edilen yolcu gözaltına alınır ve kara listeye eklenir.

Tehlikeli Eşyaların ve Güvenlik Kontrolünün Görselleştirilmesi:

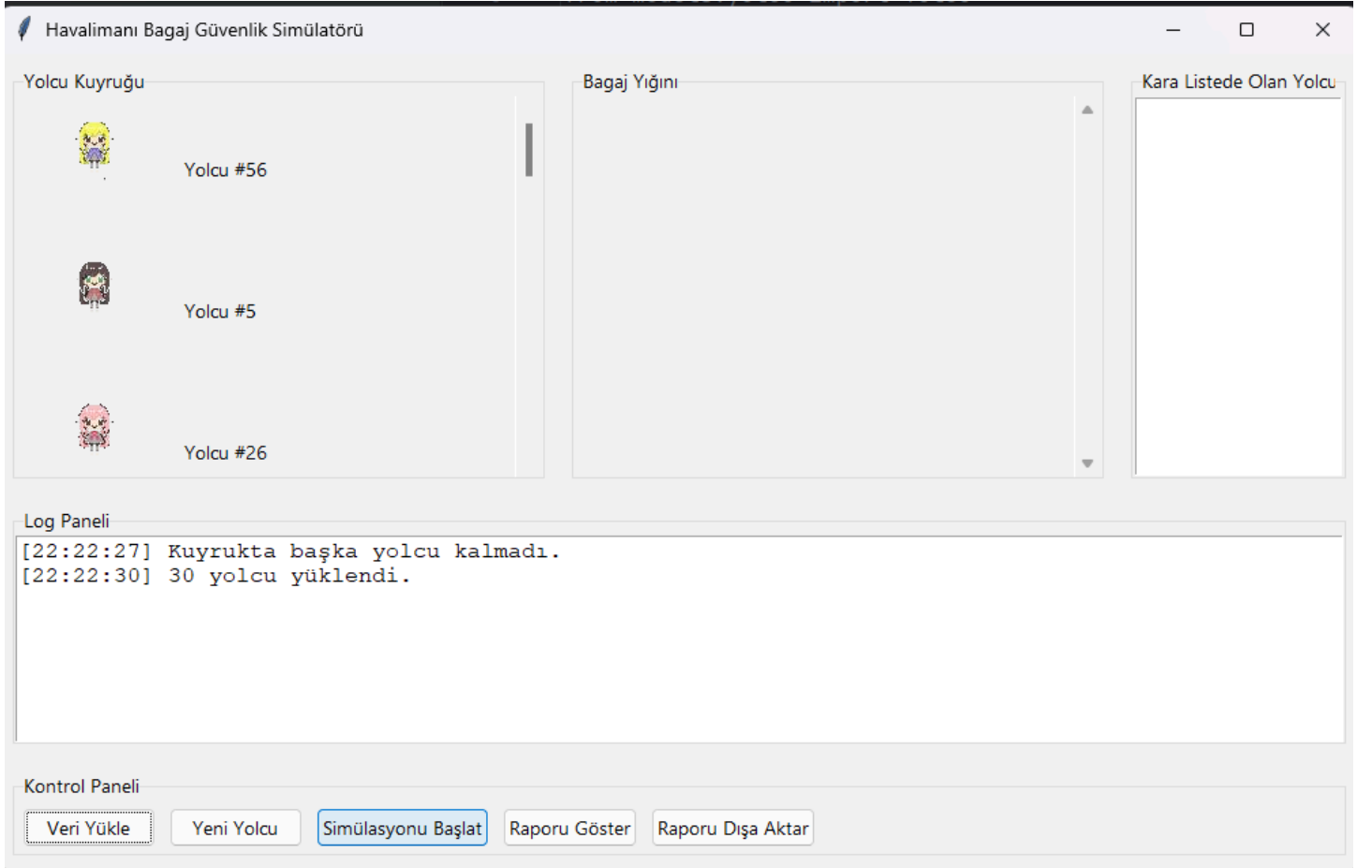
- Yolcuların bagajlarındaki eşyalar görsel olarak ve yazılı olarak ekranda görüntülenir. Güvenli eşyalar yeşil renkte, tehlikeli eşyalar ise kırmızı renkte vurgulanır.
- Eğer bir yolcunun bagajında tehlikeli bir eşya varsa, bu eşya güvenlik kontrolüne takılır ve yolcu hakkında bir işlem yapılır. Tehlikeli eşya bulunan yolcular gözaltına alınır ve kara listeye eklenir.

Kara Liste ve Gözaltına Alma:

- Kullanıcı, gözaltına alınan yolcuları kara listeye ekler veya çıkarabilir. Kara liste, şüpheli yolcuların takip edilmesi amacıyla kullanılır.
- Bu işlemde kullanıcı, kara listeye eklenen yolcuları ve onların bagaj içeriklerini görüntüleyebilir, ayrıca gerektiğinde kara listeden çıkarabilir.

Yeni Yolcu ve Bagaj Ekleme:

- Kullanıcı, “Yeni Yolcu Ekle” butonunu kullanarak sisteme yeni yolcular ve bagajlarını ekleyebilir. Yeni yolcu eklerken, sistem rastgele bir yolcu profili oluşturur ve bu yolcunun bagajında rastgele eşyalar yer alır.
- Bu yeni yolcu ve bagajlar, sistemde mevcut yolcuların arasına eklenir ve simülasyona dahil olur.



Simülasyonun Durumu ve Olay Günlüğü:

- Kullanıcı, simülasyonun ilerleyişini ve anlık durumunu takip etmek için bir olay günlüğü paneli kullanabilir. Bu panelde, her yolcunun bagajındaki eşyalar ve güvenlik durumu gibi bilgiler anlık olarak güncellenir.
- Bu, kullanıcıya simülasyonun her aşamasında ne olduğunu görsel ve yazılı olarak bildirir, böylece olayları daha kolay takip edebilir.

PROJE II Baggage Security Simulator

Projenin Amacı Nedir?

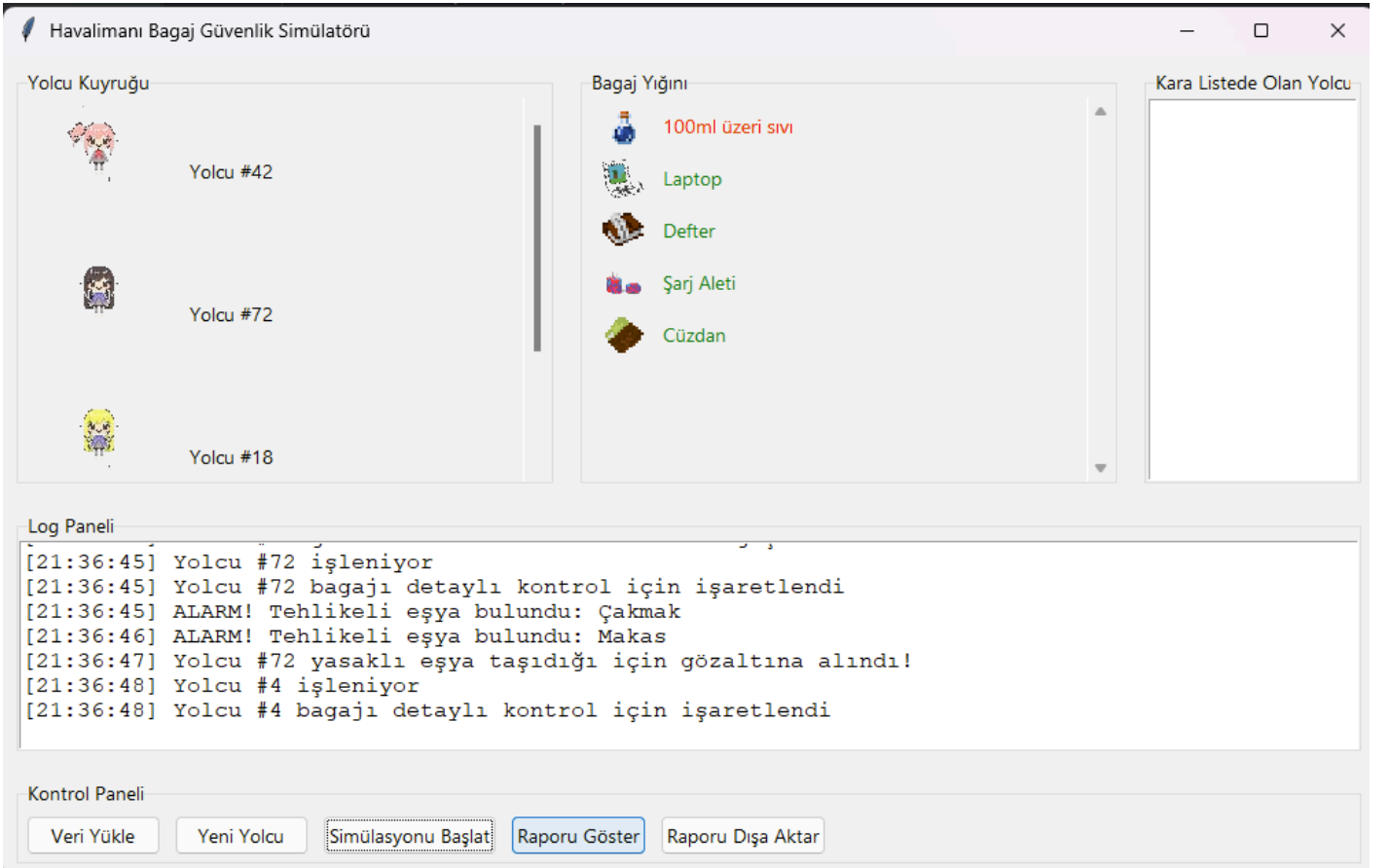
Havalimanında bagaj güvenliği, hem yolcu hem de uçuş güvenliği açısından büyük bir öneme sahiptir. Birçok yolcu, uçak seyahati hakkında yeterli bilgiye sahip olmayabilir ve bilmeden tehlikeli eşyalar taşıyabilir. Örneğin, parfüm şişesi gibi günlük yaşamda zararsız görünen nesneler, uçuş güvenliği için tehlike oluşturabilir. Ayrıca, bagaj güvenliği, sadece uçuşları güvence altına almakla kalmaz, diğer yolcuların güvenliğini sağlamak adına da kritik bir rol oynar.

Bu projede, bagaj güvenliğini minimalist bir şekilde ele aldık ve süreci görsel olarak kullanıcıya sunmayı amaçladık. Pixel art desteğiyle yolcuları ve eşyaları görselleştirerek, bagaj güvenliğini simüle ettik. Bu sayede, güvenlik sürecini hem görsel hem de fonksiyonel olarak etkili bir biçimde modelledik. Proje, hem kullanıcı dostu bir arayüz sunarak güvenlik sürecini izlemeyi kolaylaştırıyor, hem de bagaj güvenliğine dair önemli adımları basit ama etkili bir şekilde simüle ediyor.

Kullanıcı Nasıl Kullanır?

Veri Yükle Butonu:

- Kullanıcı, “Veri Yükle” butonuna tıkladığında, sistem rastgele yolcu profilleri ve bagaj içerikleri oluşturur. Bu yolcuların bagajlarında, her türlü tehlikeli veya güvenli eşya rastgele seçilir ve sisteme eklenir. Bu, bagaj güvenliği sürecinin simüle edilmesi için başlangıç adımıdır.



Simülasyonu Başlat Butonu:

- Kullanıcı, “Simülasyonu Başlat” butonuna tıklayarak simülasyonu başlatır. Bu işlemle birlikte, yolcuların görsel temsilleri (pixel art) ve bagajlarının içeriği ekranda gösterilir.
- Güvenli eşyalar yeşil renkte, tehlikeli eşyalar ise kırmızı renkte gösterilir. Bu sayede, kullanıcı tehlikeli eşyaların hangileri olduğunu kolayca görür.
- Ayrıca, yolcuların bagajlarındaki tehlikeli eşyalar, sistemin güvenlik kontrolüne takılmasına neden olur. Güvenlik kontrolü sonucunda, tehlikeli eşya tespit edilen yolcu gözaltına alınır ve kara listeye eklenir.

Tehlikeli Eşyaların ve Güvenlik Kontrolünün Görselleştirilmesi:

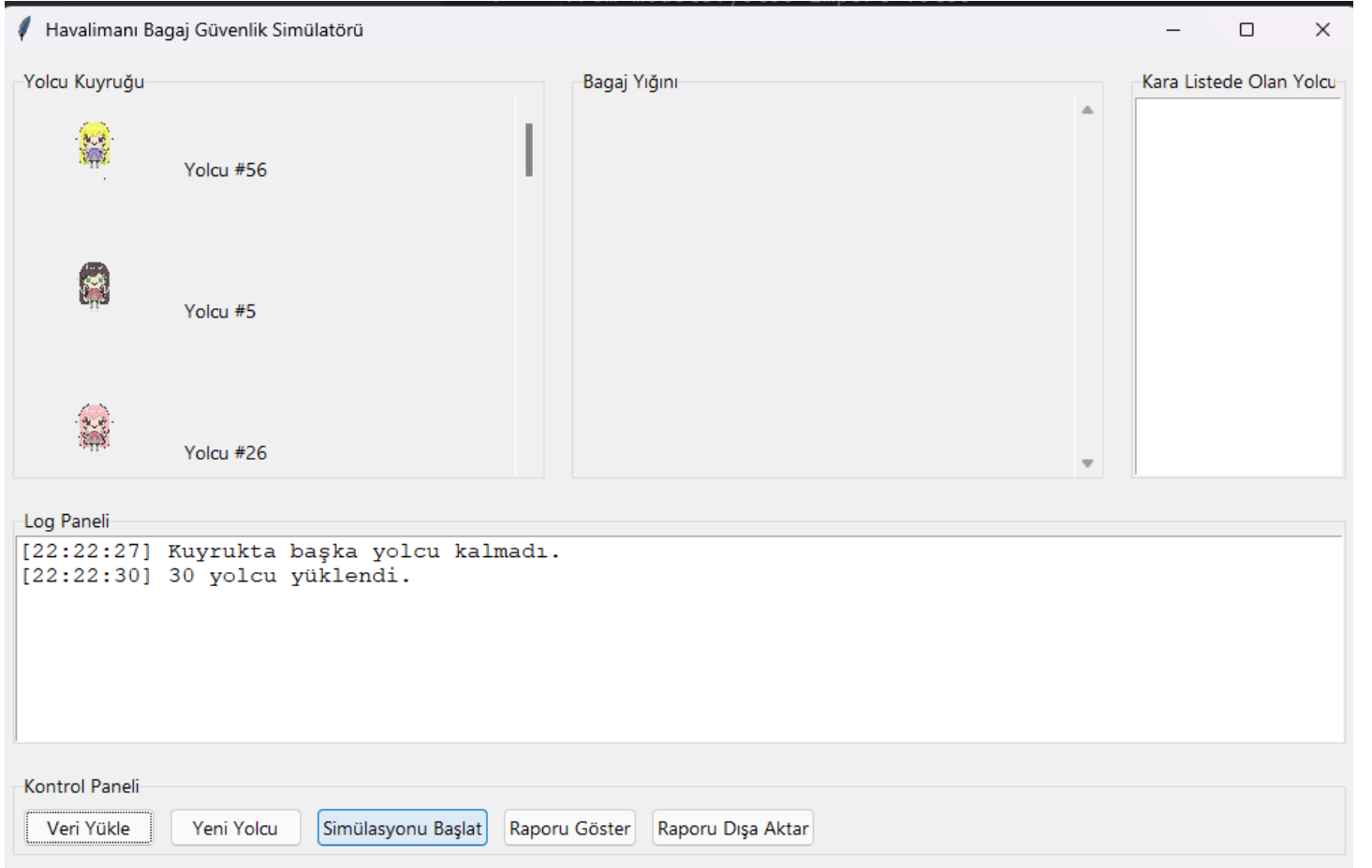
- Yolcuların bagajlarındaki eşyalar görsel olarak ve yazılı olarak ekranda görüntülenir. Güvenli eşyalar yeşil renkte, tehlikeli eşyalar ise kırmızı renkte vurgulanır.
- Eğer bir yolcunun bagajında tehlikeli bir eşya varsa, bu eşya güvenlik kontrolüne takılır ve yolcu hakkında bir işlem yapılır. Tehlikeli eşya bulunan yolcular gözaltına alınır ve kara listeye eklenir.

Kara Liste ve Gözaltına Alma:

- Kullanıcı, gözaltına alınan yolcuları kara listeye ekler veya çıkarabilir. Kara liste, şüpheli yolcuların takip edilmesi amacıyla kullanılır.
- Bu işlemde kullanıcı, kara listeye eklenen yolcuları ve onların bagaj içeriklerini görüntüleyebilir, ayrıca gerektiğinde kara listeden çıkarabilir.

Yeni Yolcu ve Bagaj Ekleme:

- Kullanıcı, “Yeni Yolcu Ekle” butonunu kullanarak sisteme yeni yolcular ve bagajlarını ekleyebilir. Yeni yolcu eklerken, sistem rastgele bir yolcu profili oluşturur ve bu yolcunun bagajında rastgele eşyalar yer alır.
- Bu yeni yolcu ve bagajlar, sistemde mevcut yolcuların arasına eklenir ve simülasyona dahil olur.



•

Simülasyonun Durumu ve Olay Günlüğü:

- Kullanıcı, simülasyonun ilerleyişini ve anlık durumunu takip etmek için bir olay günlüğü paneli kullanabilir. Bu panelde, her yolcunun bagajındaki eşyalar ve güvenlik durumu gibi bilgiler anlık olarak güncellenir.
- Bu, kullanıcıya simülasyonun her aşamasında ne olduğunu görsel ve yazılı olarak bildirir, böylece olayları daha kolay takip edebilir.

Mimari Tasarım

Projede temel yazılım mühendisliği prensiplerine bağlı kalınarak modüler bir mimari kurgulanmıştır:

1. Bagaj.py

- **Amacı:** Bagaj sınıfını tanımlar. Her yolcunun sahip olduğu bagajlar bu sınıfla temsil edilir.
- **İçeriği:** Bagajların içeriği (eşya listesi), tehlikeli olup olmadıkları gibi bilgiler burada saklanır ve işlenir.

2. linkedlist.py

- **Amacı:** Bağlı liste veri yapısını tanımlar.
- **İçeriği:** İşlenen bagajların geçmişi burada saklanır. Örneğin, güvenlikten geçen veya sorun çıkaran bagajların sıralı bir kaydı tutulur.

3. Queue.py

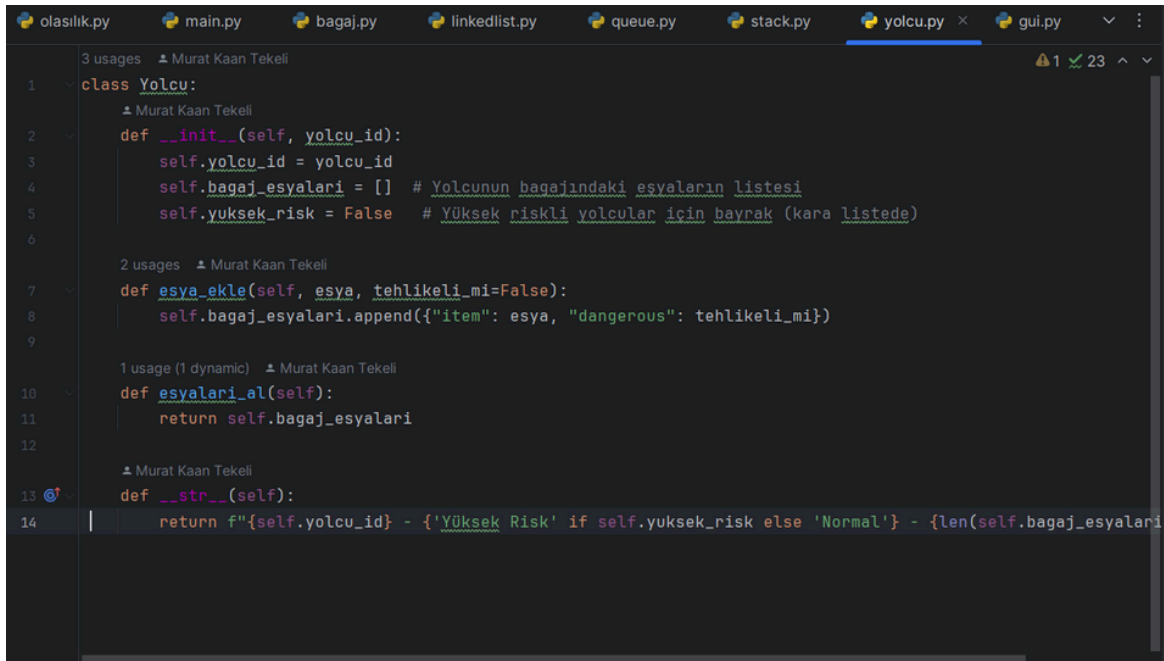
- **Amacı:** Kuyruk (FIFO) yapısını uygular.
- **İçeriği:** Yolcuların veya bagajların sırayla işlem görmesini sağlar. Örneğin, bagaj kontrolü için sıraya giren yolcular bu yapı ile işlenir.

4. Stack.py

- **Amacı:** Yığın (LIFO) yapısını tanımlar.
- **İçeriği:** Öncelikli ve acil işlem gereken bagajlar bu yapıda tutulur. Son giren ilk çıkar mantığıyla, öncelikli kontrolleri sağlar.

5. Yolcu.py

- **Amacı:** Yolcu sınıfını tanımlar.
- **İçeriği:** Her yolcunun adı, ID'si ve sahip olduğu bagajları içerir. Rastgele yolcu oluşturma işlemleri de burada yer alabilir.



```
1 class Yolcu:
2     # Murat Kaan Tekeli
3     def __init__(self, yolcu_id):
4         self.yolcu_id = yolcu_id
5         self.bagaj_esyalari = [] # Yolcunun bagajındaki eşyaların listesi
6         self.yuksek_risk = False # Yüksek riskli yolcular için bayrak (kara listede)
7
8     2 usages  Murat Kaan Tekeli
9     def esya_ekle(self, esya, tehlikeli_mi=False):
10         self.bagaj_esyalari.append({'item': esya, 'dangerous': tehlikeli_mi})
11
12     1 usage (1 dynamic)  Murat Kaan Tekeli
13     def esyalari_al(self):
14         return self.bagaj_esyalari
15
16     # Murat Kaan Tekeli
17     def __str__(self):
18         return f"{self.yolcu_id} - {'Yüksek Risk' if self.yuksek_risk else 'Normal'} - {len(self.bagaj_esyalari)}
```

6. Olasilik.py

- **Amacı:** Olasılık hesaplamalarını ve risk analizini yapar.
- **İçeriği:** Tehlikeli eşya olasılığı, risk puanı hesaplama, tespit ihtimalleri gibi güvenlik simülasyonunun mantıksal kısmı burada gerçekleşir.

7. gui.py

- **Amacı:** PyQt5 ile kullanıcı arayüzünü oluşturur.
- **İçeriği:** Butonlar (Başlat, Durdur, Veri Yükle, Yeni Yolcu Ekle), görseller (pixel art), olay günlüğü, yolcu listesi gibi tüm görsel öğeler bu dosyada tanımlanır ve yönetilir.

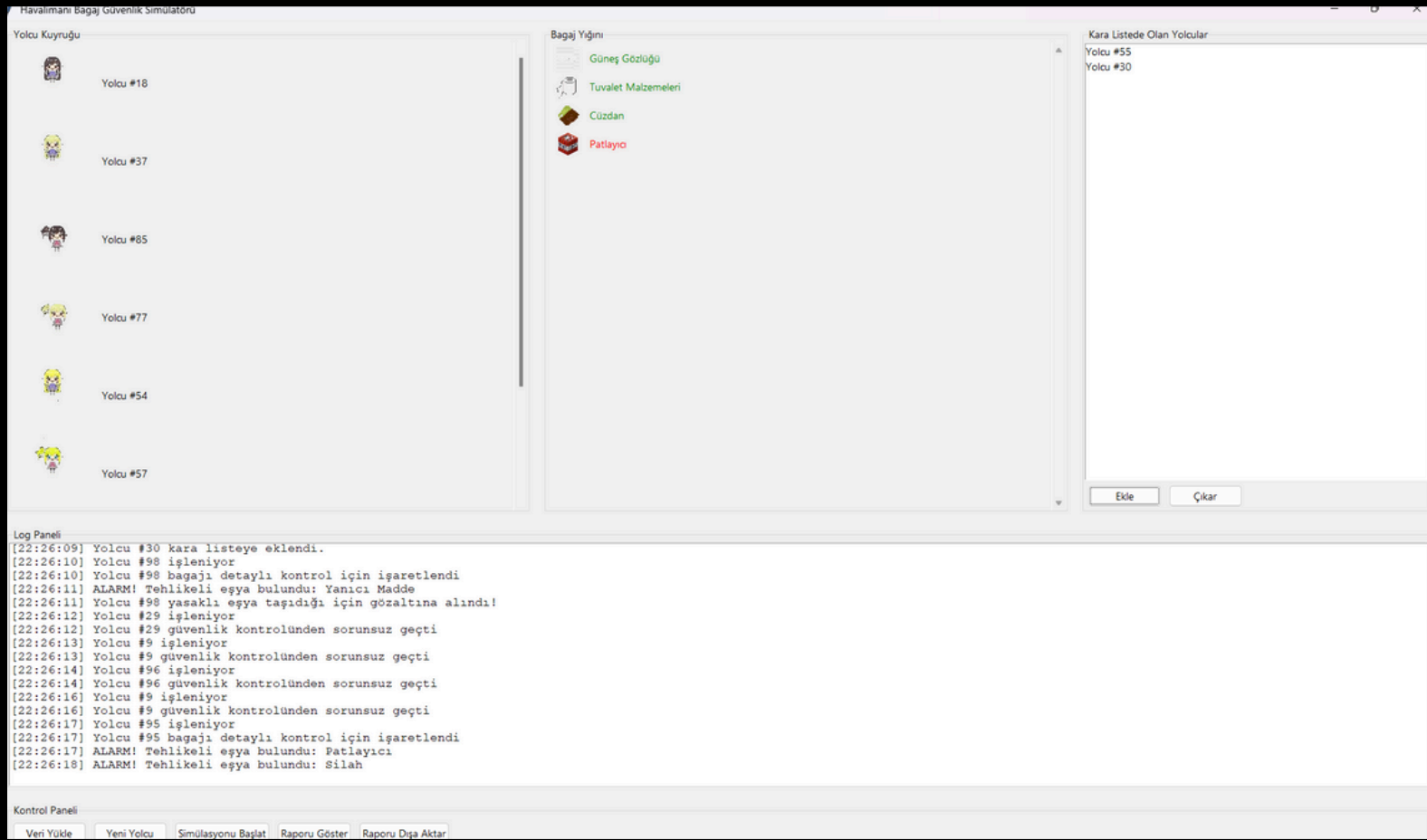
8. main.py

- **Amacı:** Projenin başlangıç dosyasıdır.
- **İçeriği:** Tüm modülleri bir araya getirerek simülasyonu başlatır. gui.py'yi çağırır ve kullanıcı arayüzünü çalıştırır.

Arayüz Gereksinimleri Ve Diyagramlar

- **Kullanıcı Arayüzü:**

- "Veri Yükle", "Simülasyonu Başlat", "Durdur" ve "Yeni Yolcu Ekle" butonları ile kontrol.
- Yolcu ve bagaj listesi paneli.
- Olay günlüğü (log) ile sistemde neler olduğunu takip etme.
- Tehlikeli ve güvenli eşyaların görsel (renkli) olarak ayırt edilmesi.
- PyQt5 ile hazırlanmış sade ve işlevsel bir arayüz.



- **Donanım Gereksinimi:**

- Python 3.10+ kurulu bir bilgisayar.
- PyQt5 kütüphanesi yüklü olmalı.
- Ortalama işlemci ve RAM yeterlidir (özel donanım gerekmez).

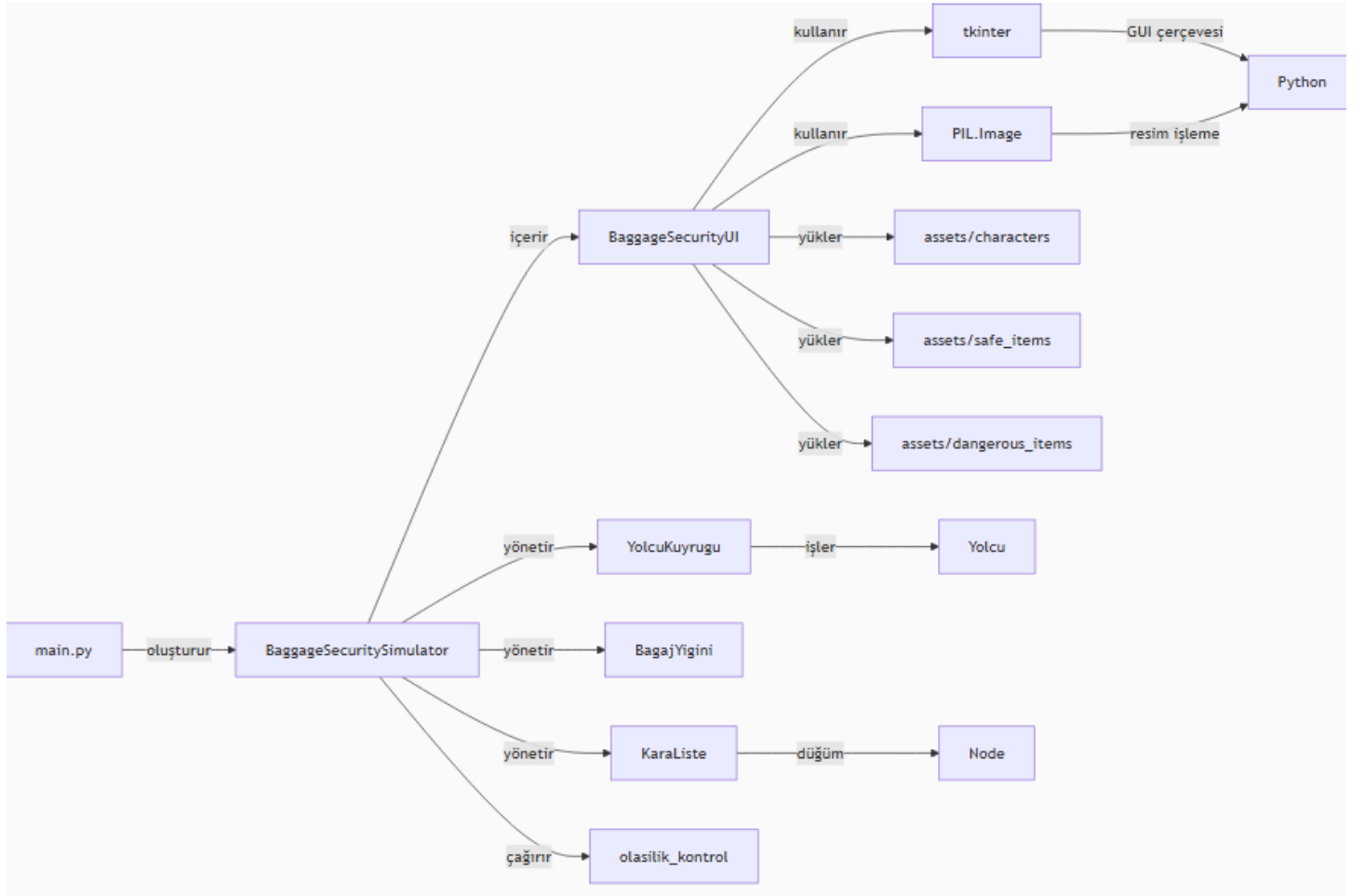
2. Fonksiyonel Gereksinimler

- **Yolcu ve Bagaj Yönetimi:**

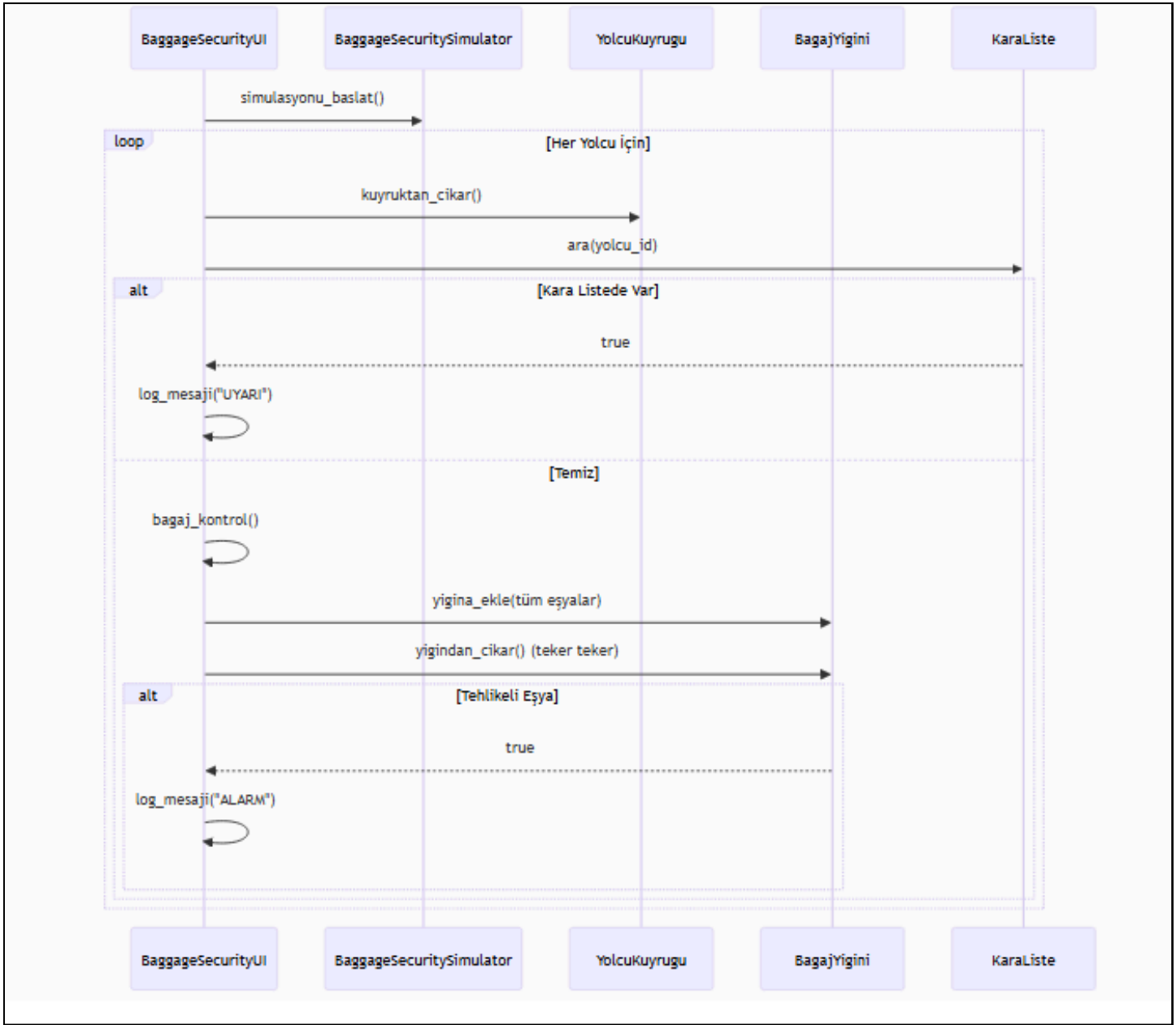
- Rastgele yolcu ve bagaj oluşturma.

- Bagajların FIFO (queue) yapısıyla işlenmesi.
- Tehlikeli bagajların LIFO (stack) yapısıyla kontrolü.
- Tüm bagajların geçmişinin bağlı liste (linked list) ile saklanması.
- Olasılık hesaplamaları ile güvenlik kontrollerinin gerçekçi hale getirilmesi.
- **Simülasyon Kontrolü:**
 - Kullanıcının simülasyonu başlatması ve durdurması.
 - Tehlikeli eşya varsa bagajın incelenmesi, yolcunun kara listeye alınması.
 - Kullanıcının yeni yolcu/bagaj ekleyebilmesi.
 - Simülasyon boyunca görsel ve yazılı geribildirimlerin gösterilmesi.

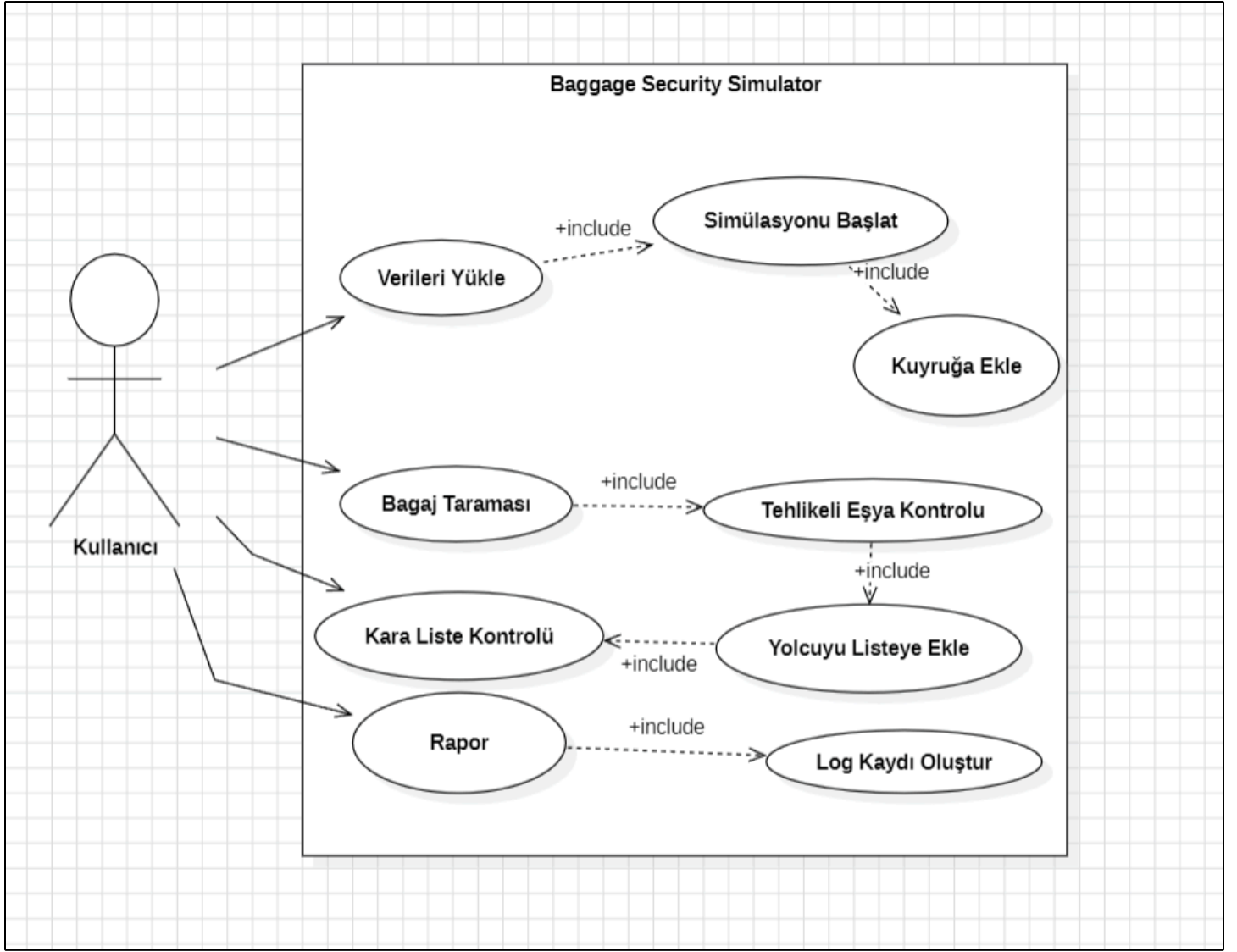
Modul Diyagramı:



Sequence Diyagramı:

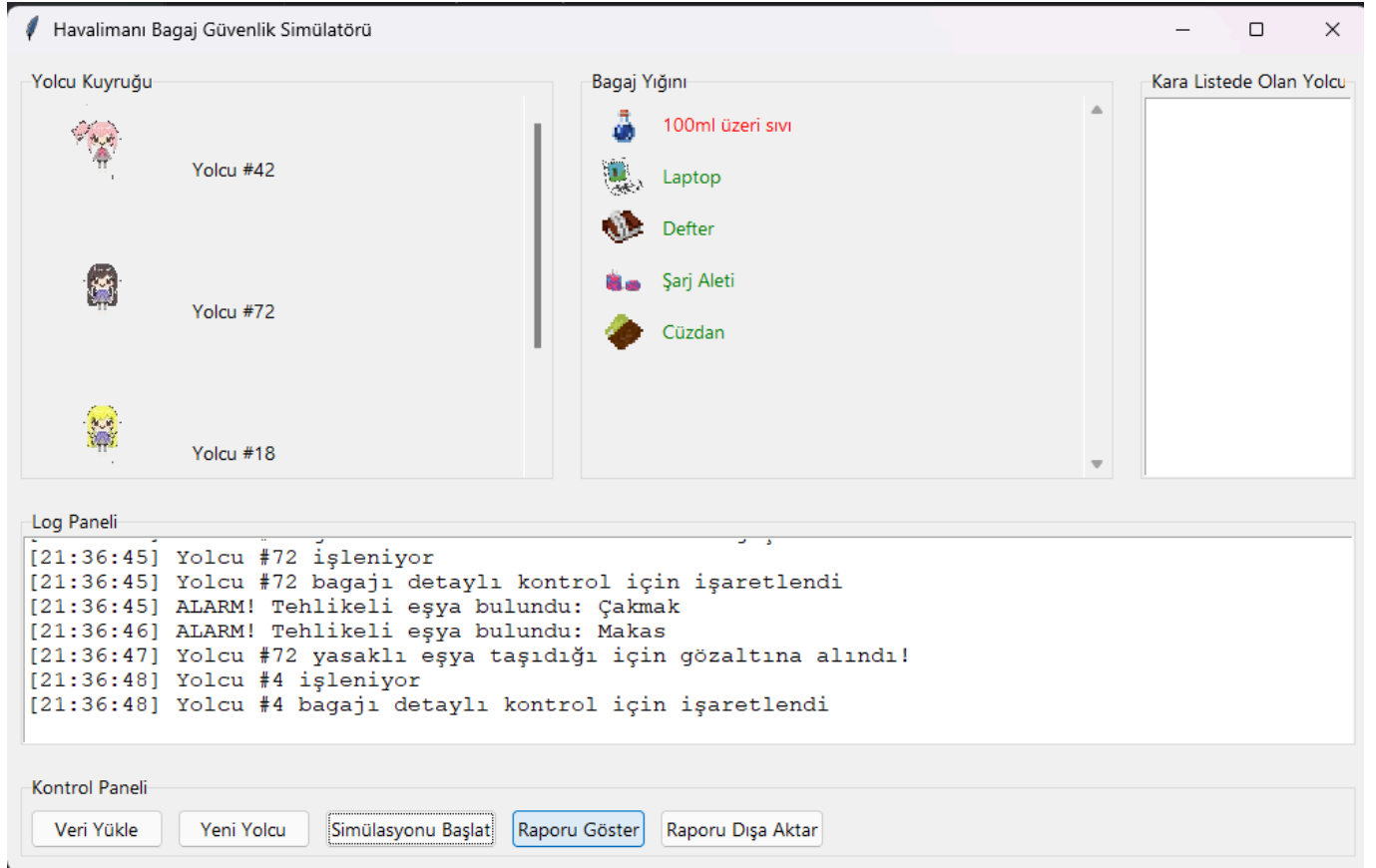


Use-Case Diyagramı:



Kullanıcı Arayüzü Tasarımı

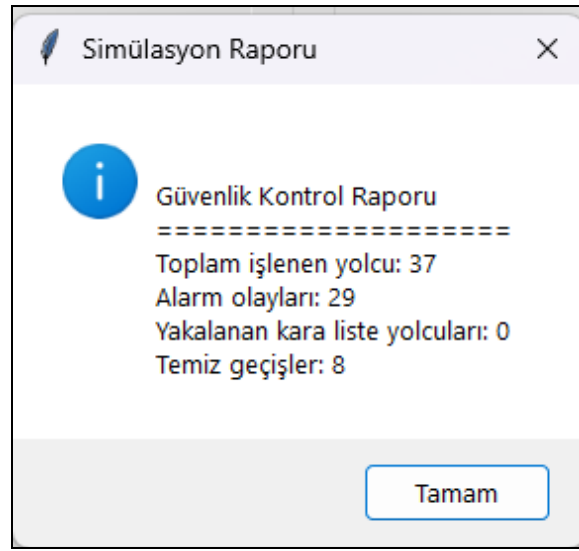
- Başlat/Durdur butonları



- Yolcu ve bagaj bilgilerini gösteren bilgi ekranı

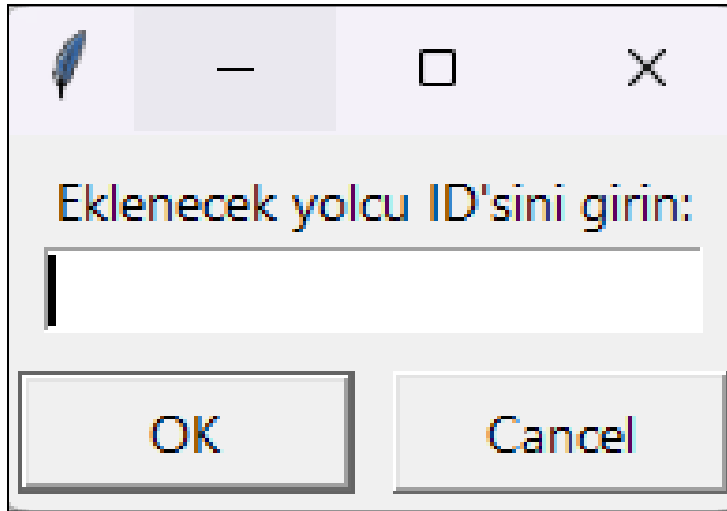


- Durum penceresi ile anlık olay günlüğü



- Arayüz gui.py içinde tanımlanmıştır, simülasyon main.py çalıştırılarak başlatılır.

- Kara liste :



- Kara listeye eklenecek yolcu ID'sini girme:

Kara Listede Olan Yolcular

Yolcu #55
Yolcu #30

EkleÇıkar

• Yolcu kuyruğu:

Log Paneli	
[22:31:18]	Yolcu #88 işleniyor
[22:31:18]	Yolcu #88 bagajı detaylı kontrol için işaretlendi
[22:31:20]	ALARM! Tehlikeli eşya bulundu: Makas
[22:31:20]	Yolcu #88 yasaklı eşya taşıdığı için gözaltına alındı!
[22:31:21]	Yolcu #90 işleniyor
[22:31:21]	Yolcu #90 güvenlik kontrolünden sorunsuz geçti
[22:31:22]	Yolcu #76 işleniyor
[22:31:22]	Yolcu #76 bagajı detaylı kontrol için işaretlendi
[22:31:24]	ALARM! Tehlikeli eşya bulundu: Patlayıcı
[22:31:24]	Yolcu #76 yasaklı eşya taşıdığı için gözaltına alındı!
[22:31:25]	Yolcu #7 işleniyor
[22:31:26]	Yolcu #7 güvenlik kontrolünden sorunsuz geçti
[22:31:27]	Yolcu #66 işleniyor
[22:31:27]	Yolcu #66 güvenlik kontrolünden sorunsuz geçti
[22:31:28]	Yolcu #44 işleniyor
[22:31:28]	Yolcu #44 güvenlik kontrolünden sorunsuz geçti
[22:31:29]	Kuyrukta başka yolcu kalmadı.

• Bagaj kontrol:

Bagaj Yığını



Kulaklık



Kalem



Defter



Cüzdan



Laptop



Kalem



Kitap

Proje Özeti

Havalimanlarında bagaj güvenliği, yolcu güvenliği ve operasyonel verimlilik açısından kritik öneme sahiptir. Bu projede, yolcuların havalimanına girişinden itibaren bagaj güvenliği süreci yazılımsal olarak modellenmiş ve simüle edilmiştir.

Simülasyon, yolcu ve eşyaların rastgele oluşturulması, görsel olarak izlenmesi (pixel-art destekli), tehlikeli eşyaların tespiti ve kara listeye alınma sürecini kapsamaktadır. Projede kullanılan PyQt5 arayüzü, kullanıcıya minimal ve işlevsel bir görsel deneyim sunar.

Sorunsuz Çalışanlar:

- Veri yapıları (Queue, Stack, LinkedList)
- Olasılık hesaplama
- Arayüz ve kullanıcı etkileşimi
- Simülasyon akışı
-

Geliştirilmeye Açık Noktalar:

- Kara listeye otomatik ekleme
- Arayüzde dinamik kullanıcı ayarları (ör. yolcu sayısı, eşya sayısı)

Eksik Yönler

- Kara listeye alma işlemi otomatik değildir. Kullanıcıya bırakılmıştır.
- Yeni yolcu eklemede kullanıcının dış görünüşünü özelleştirme seçeneği yoktur.
- GUI'de kullanıcı, yolcu sayısı veya eşya türünü değiştirememektedir.

Use Case (Kullanım Senaryosu) - Özet

- Kullanıcı, "Veri Yükle" butonuna basar → yolcu & bagajlar oluşur
- "Simülasyonu Başlat" ile simülasyon başlar → riskli eşyalar tespit edilir
- Kullanıcı, riskli yolcuları gözlemleyip kara listeye ekler veya çıkarır
- "Yeni Yolcu Ekle" ile sistem yeni yolcu üretir
- "Simülasyonu Durdur" ile işlem sonlandırılır

Proje Görev Dağılımı

Bu proje, iki kişilik bir ekip tarafından ortak şekilde geliştirilmiştir:

- **Murat Kaan Tekeli (230502007)**

Projede ağırlıklı olarak yazılım geliştirme sürecinde görev almıştır. Kodlama sürecinde özellikle arayüz (GUI), veri yapıları (Queue, Stack, LinkedList) ve simülasyon mantığının kurulmasında aktif rol oynamıştır.

- **Sıla Serdar (230502014)**

Proje geliştirme sürecine yazılım geliştirme sürecinde proje arkadaşı olan Murat Kaan Tekeli'ye katkıda bulunmuş, aynı zamanda proje raporunun hazırlanmasında sorumluluk üstlenmiştir. Kodların belgelenmesi, açıklamaların yazılması ve proje çıktılarının düzenli sunulması konularında görev almıştır.

Her iki ekip üyesi, yazılımın planlanması, test edilmesi ve son kontrolleri aşamalarında birlikte çalışarak süreci tamamlamıştır.

Sonuç

Bu simülasyon projesi, havalimanlarındaki bagaj güvenliği süreçlerini temel düzeyde modelleyerek yazılım mühendisliği prensiplerine uygun bir yapı sunmuştur. Veri yapılarının doğru ve amaca uygun seçimi, modüler yapı ve etkileşimli kullanıcı arayüzü sayesinde hem eğitsel hem de işlevsel bir yazılım geliştirilmiştir.

Eksik yönlerine rağmen proje, geliştirilmeye açık ve genişletilebilir bir temel sunmaktadır. Simülasyon ortamında görselleştirme, kullanıcı etkileşimi ve güvenlik mantığının sade bir şekilde birleştirilmesi bu projeyi anlamlı kılmaktadır.