



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

ANİMASYONLU ÇİZİM EKRANI

Hilal AYDIN - 220501030
Şevval ÖZEREN - 220501028

PROF. DR. Tarık DURU

24.11.2024

1 GİRİŞ

1.1 Projenin amacı

- Bu projede, kullanıcı etkileşimiyle hareket eden toplar içeren bir animasyon uygulaması geliştirilmesi amaçlanmıştır. Kullanıcılara renk ve boyut seçimi gibi özelleştirme seçenekleri sunulmuştur. Kullanıcılar, topların hızını ayarlayarak, animasyonu başlatıp durdurabilme gibi kontrol fonksiyonlarını kullanarak uygulamayı yönetebilirler. Aynı zamanda, topların birbirlerinin üzerine çakışmadan yerleşmesini sağlamak amacıyla çakışma kontrolü eklenmiştir. Bu proje, Python programlama dili ve Tkinter kütüphanesi kullanılarak yapılmıştır ve temel nesne yönelimli programlama (OOP) teknikleriyle yapılandırılmıştır.
- **Projede Gerçekleştirilmesi Beklenenler**
 1. **Kullanıcı Etkileşimi:**

Kullanıcıların renkli topların rengini seçme; top boyutunu seçme; animasyonu başlatma, durdurma, sıfırlama ve hızını arttırma gibi kontrolleri yapabilmesi.
 2. **Grafik Arayüz:**

Top animasyonunun kolay öğrenilir ve kullanılabilir bir arayüz oluşturularak gösterilmesi. Top hareketlerinin gözlemlenebilmesi.
 3. **Topların Animasyonu:**

Topların ekranda hareket etmesi ve ekranın kenarlarına çarptığında geri sekmesi sağlanmalı. Topların hızları, kullanıcı tarafından kontrol edilebilmeli.
 4. **Topların Hız Kontrolü:**

Kullanıcı, topların hızını artırabilmeli ve hız değişiklikleri animasyonu etkili bir şekilde yansıtmalı.
 5. **Top Animasyonunu Sıfırlayabilme ve Yeniden Başlatabilme:**

Kullanıcı animasyonu durdurduktan sonra tüm topları ekrandan silebilmeli ve yeni bir animasyon başlatabilmeli.

2 GEREKSİNİM ANALİZİ

2.1 Arayüz gereksinimleri

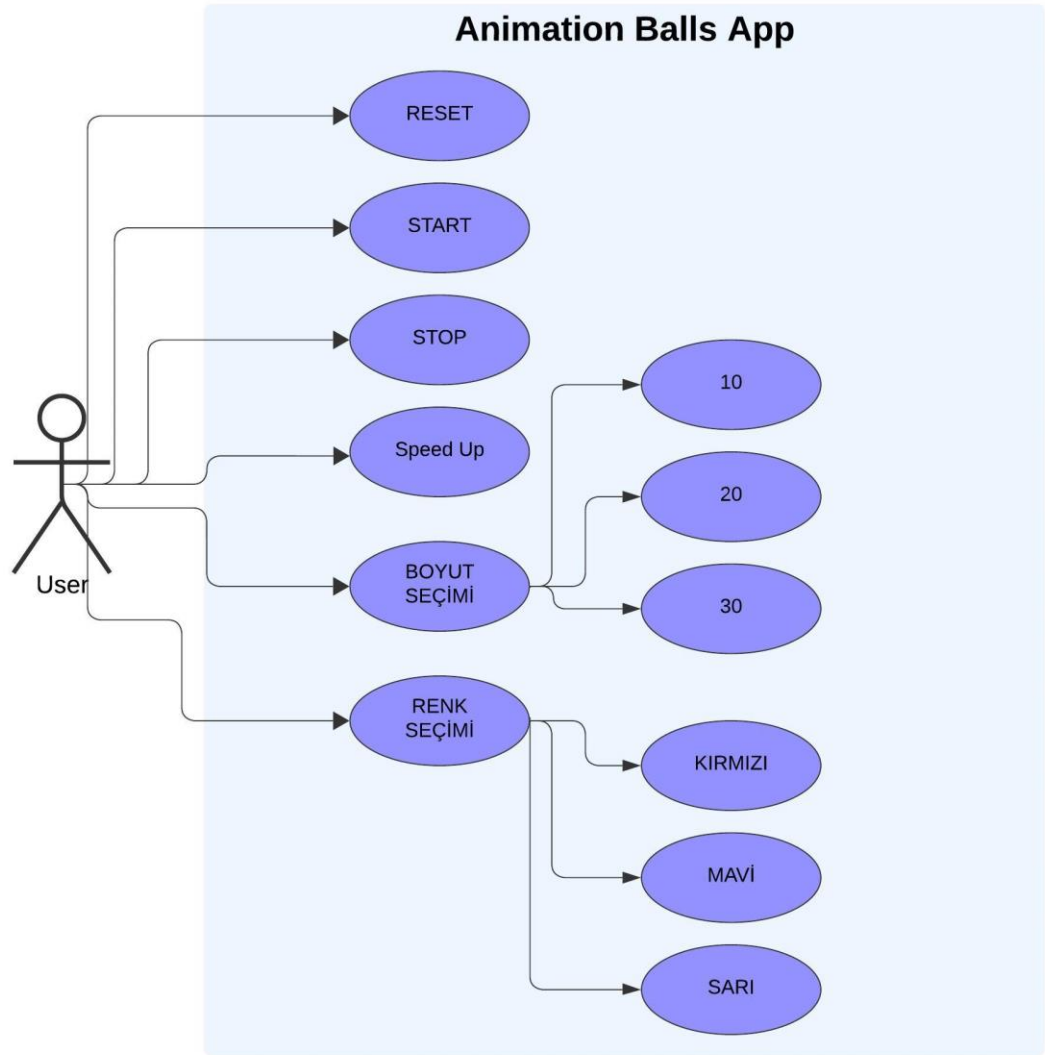
- Geliştirmiş olduğumuz arayüzde top boyutu seçebilmek için 3 farklı boyutta top butonu, topların rengini seçebilmek için kırmızı, sarı ve maviden oluşan 3 renk butonu bulunmaktadır. Bunlarla birlikte animasyonu başlatmak için start butonu, durdurmak için stop butonu,

sıfırlamak için reset butonu ve hızını arttırmak için speed up butonu bulunmaktadır.

2.2 Fonksiyonel gereksinimler

- Oluşturulan fonksiyonlarla seçilen renk ve boyutlara göre topların eklenmesi, animasyonun başlatılması, hızlandırılması, durdurulması ve sıfırlanması işlemleri gerçekleştirilir.

2.3 Use-Case diyagramı



3 TASARIM

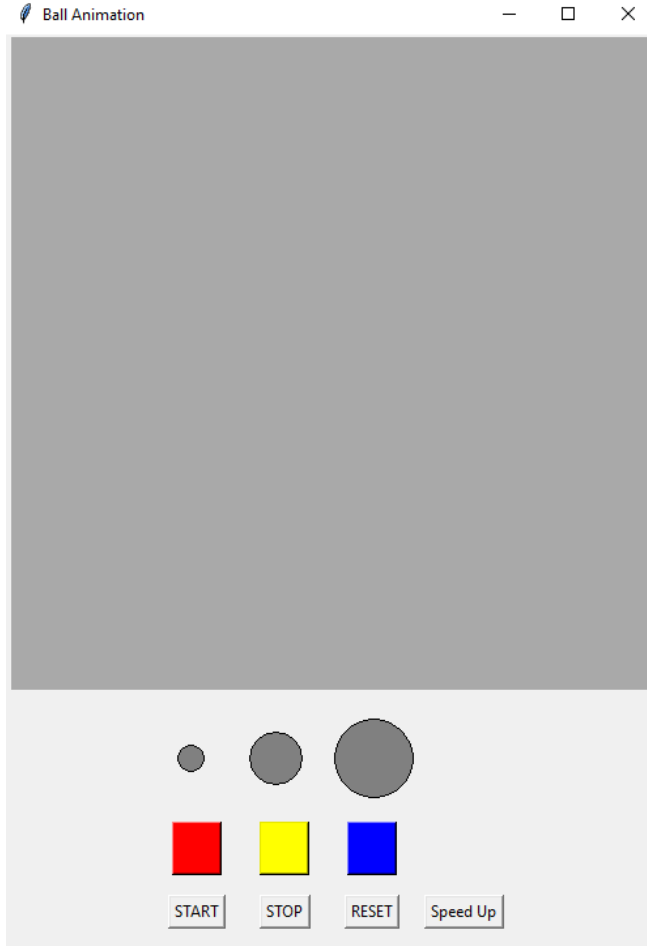
3.1 Kullanılacak teknolojiler

- Bu program Python dili kullanılarak oluşturulmuştur.
- **Tkinter** kütüphanesi kullanılmıştır. Bu kütüphane grafiksel arayüzlü uygulamalar geliştirmek için kullanılır. Düğmeler (Button), metin

kutuları (Entry), etiketler (Label), menüler (Menu) gibi çeşitli bileşenler içerir.

- **Random** kütüphanesi kullanılmıştır. Bu kütüphane rastgele sayı üretmek ve rastgele işlemler yapmak için kullanılan standart bir kütüphanedir.

3.2 Kullanıcı arayüzü tasarımı



- İlk olarak bir renk seçilir.
- Sonrasında top boyutu seçilir.
- Start butonu ile animasyon başlatılır.
- Speed Up butonu ile animasyon hızlandırılır.
- Stop butonu ile animasyon durdurulur.
- Reset butonu ile animasyon sıfırlanır.

4 UYGULAMA

4.1 Kodlanan bileşenlerin açıklamaları

- BallAnimation sınıfı, bir pencere arayüzü (GUI) oluşturarak topların animasyonlarını gerçekleştiren bir uygulama tasarlar.

- Uygulamanın ana penceresi oluşturulur. Başlık, pencere boyutu ve başlangıç pozisyonu ayarlanır.
- Canvas ile topların çizileceği alan oluşturulur.
- Kontrol düğmeleri için bir çerçeve eklenir.
- Self.balls, top nesnelerinin listesini tutar.
- Self.is_running, animasyonun çalışıp çalışmadığını kontrol eder.
- Self.speed, animasyonun hızını kontrol eder.
- Self.selected_color, kullanıcının seçtiği top rengini saklar.
- Self.selected_size, kullanıcının seçtiği top boyutunu saklar.
- Create_color_buttons fonksiyonuyla kırmızı, sarı ve mavi renklerde 3 buton oluşturulur.
- Create_size_selector fonksiyonunda toplar için seçenek olarak sunulan boyut değerleri sizes listesinde tutulur. Boyutların tuval üzerindeki konumları, koordinat çiftleriyle belirtilir. For döngüsüyle her boyut için bir oval çizilir ve bu ovalin pozisyonu ayarlanır. Tıklanan boyut, self.select_size metoduna aktarılır.
- Create_control_buttons fonksiyonuyla start, stop, reset ve speed up butonları oluşturulur.
- Select_color fonksiyonuyla kullanıcının seçtiği renk tutulur.
- Select_size fonksiyonuyla kullanıcının seçtiği boyut tutulur.
- Add_ball fonksiyonu tuval üzerinde bir top eklemek için kullanılır. Top, belirlenen boyut ve renkte, mevcut toplarla çakışmayacak şekilde yerleştirilir.
 - Top için rastgele bir (x, y) pozisyonu belirlenir.
 - overlap değişkeni, çakışmayı kontrol etmek için kullanılır ve başlangıçta False olarak ayarlanır.
 - Mevcut topların pozisyonları ile yeni pozisyon arasındaki mesafe hesaplanır. Eğer mesafe, yeni topun boyutuyla mevcut topun boyutunun toplamından küçükse, toplar çakışır. Bu durumda overlap = True olur ve döngü devam eder. Eğer çakışma bulunmuyorsa döngü sona erer ve pozisyon doğrulanır.
 - dx ve dy topun hareket yönlerini temsil eder. Her birine rastgele -1 veya 1 atanır (sağa/sola ve yukarı/aşağı hareket).
 - Top, doğrulanan pozisyona çizilir.
 - Her topun özelliklerini içeren bir sözlük oluşturulur ve toplar listesine eklenir.
- Start_animation fonksiyonuyla self.is_running = True yapılır ve animate fonksiyonu çağrılır.
- Stop_animation fonksiyonuyla self.is_running = False yapılır ve animasyon durdurulur.
- Reset_canvas fonksiyonuyla tüm toplar tuvalden silinir.
- Speed_up fonksiyonuyla topların hızı her seferinde 2 arttırılır.
- Topların hareketini ve tuval üzerinde animasyonunu sağlar. Toplar,

belirlenen hızda hareket ederken sınırlarla çarpışma kontrolü yapılır.

- Self.is_running = True ise animasyon çalıştırılır.
- Topların mevcut koordinatları alınır.
- Dx ve dy ile topun hareket yönü ve hızı belirlenir.
- Topun bir sonraki konumu, tuvalin kenar sınırlarını aşarsa, hareket yönü tersine çevrilir.
- Self.canvas.move ile top yeni koordinatlara taşınır.
- Self.root.after(10, self.animate) ifadesi, bu metodu 10 milisaniye sonra tekrar çağırarak animasyonu devam ettirir.

4.2 Görev dağılımı

- Tüm aşamalar ortak yapılmıştır.

4.3 Proje isterlerine göre eksik yönler

- Proje gerçekleşmesi beklenen tüm gereksinimleri karşılamaktadır.

5 GITHUB LINKLERİ

- <https://github.com/sevvalozrn>
- <https://github.com/HilallAydinn>