

HOŞ GELDİNİZ!!

MÜZİK UYGULAMASI PROJESİ

EVLA KARAGÖZ

231041040

Bilgisayar Mühendisliği
İstanbul Gedik Üniversitesi

Nesneye Yönelik Programlama

THAT'S WHAT SHE PLAYED





PROJE ÖZETİ

Projemi, bir müzik oynatma uygulaması simülasyonu olarak tasarladım. Kullanıcılar sisteme "Standart Kullanıcı" veya "Sanatçı" olarak giriş yapabilir ve her rolün kendine özgü özelliklerini bulunmaktadır. Projemi, Object-Oriented Programming (OOP) prensipleri (kapsülleme, kalıtım, çok biçimlilik ve soyutlama) kullanılarak geliştirdim. Aslında özetle; Kullanıcıların şarkы çalma, şarkы ekleme (sadece sanatçilar için) ve mevcut şarkы listesini görüntüleme işlemelerini gerçekleştirebileceği bir komut satırı tabanlı uygulamadır.

PROJE GEREKSİNİMLERİ



1.Şarkıların Yönetimi:

- Şarkılarla ilgili ad, sanatçı, albüm,süre ve tür bilgileri tutulur.
- Şarkıların eklenmesi ve listelenmesi işlemleri yapılır.

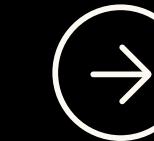
2.Kullanıcı Türleri:

- İki tür kullanıcı vardır: Standart Kullanıcı ve Sanatçı.
- Standart Kullanıcılar yalnızca şarkı çalabilir.
- Sanatçılar, şarkı calma işlemlerine ek olarak şarkı ekleyebilir.

3.Kullanıcı Etkileşimi:

- Kullanıcılar isimlerini girerek ve sanatçı olup olmadıklarını belirterek sisteme giriş yapar.
- Kullanıcıların menüden bir işlem seçmesi sağlanır:
 - Şarkı calma
 - Şarkı ekleme (sadece sanatçılar)
 - Şarkı listesini görüntüleme
 - Uygulamadan çıkış

PROJEDE KULLANILANLAR



1. Encapsulation (Kapsülleme):

- Şarkı sınıfındaki özellikler (ParcaAdı, Sanatci, Album, Sure, Tur) kapsülleme ile korunur.

2. Inheritance (Kalıtım): StandartKullanici ve Sanatci sınıfları, Kullanici sınıfından türetilmiştir.

3. Polymorphism (Çok Biçimlilik):

- ParcaCal методу Kullanici sınıfında soyut (abstract) olarak tanımlanmış, türetilen sınıflar kendine özgü şekilde bunu kullanır.

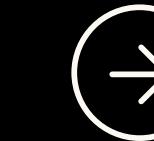
4. Abstract (Soyutlama):

- Kullanici sınıfı soyut sınıf olarak oluşturulmuş, temel davranışlar burada tanımlanmıştır.

5. Enum (Sabit Listesi): SarkiTuru: Şarkı türlerini yönetmek için bir sabit listesi kullanılmıştır.

6. Yapıcılar (Constructors): Kullanici Sınıfı Yapıcısı, StandartKullanici Sınıfı Yapıcısı, Sanatci Sınıfı Yapıcısı, Muzik Sınıfı Yapıcısı

PROJEDE KULLANILANLAR



6.List<Muzik>: Şarkıları saklamak için generic bir liste kullanılmıştır.

7.LINQ (Language Integrated Query):

- Şarkı listesinden FirstOrDefault ile arama yapılmıştır.

8.Kullanıcı Girişleri ve Kontrolleri:

- Kullanıcı türü ve seçimlerine bağlı işlemler dinamik olarak yönetilmiştir.

ALGORİTMA

1.BAŞLANGIÇ

- Kullanıcıyı karşıla
- Kullanıcıdan kullanıcı adı al
- Kullanıcı türünü sor (Sanatçı mı? Evet/Hayır)

2.KULLANICI

TÜRÜNÜ BELİRLE:

- Eğer kullanıcı sanatçısı, Sanatçı sınıfını oluştur
- Eğer kullanıcı standart kullanıcısa, Standart Kullanıcı sınıfını oluştur

3.ANA MENÜYE

GEÇ:

Sonsuz bir döngü başlat.

- Kullanıcıya aşağıdaki işlemleri sun:

- 1 - Şarkı Çal
- 2 - Şarkı Ekle (Sadece Sanatçılar için)
- 3 - Şarkı Listesini Görüntüle
- 4 - Çıkış

4.KULLANICI İŞLEM SEÇİMİ YAPACAK

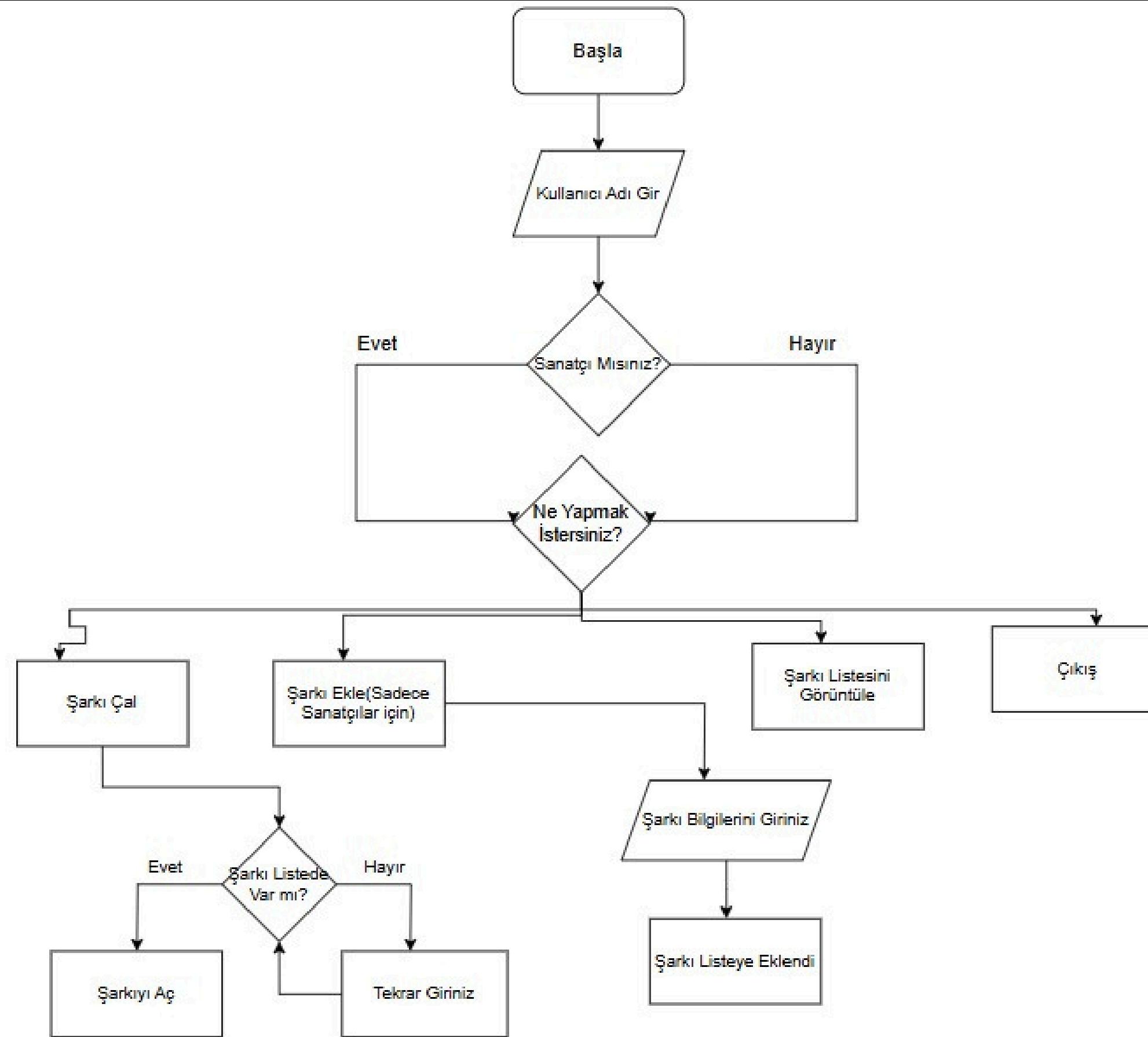
- Şarkı Çalma:
 - Şarkı adı al ve listeyi kontrol et. Şarkı varsa çal, yoksa hata mesajı göster.
- Şarkı Ekleme:
 - Eğer sanatçısı, şarkıyı ekle. Standart kullanıcısa hata mesajı göster.
- Şarkı Listesi:
 - Tüm şarkıları listeleye.
- Çıkış:
 - Uygulamayı kapat.

5.SON

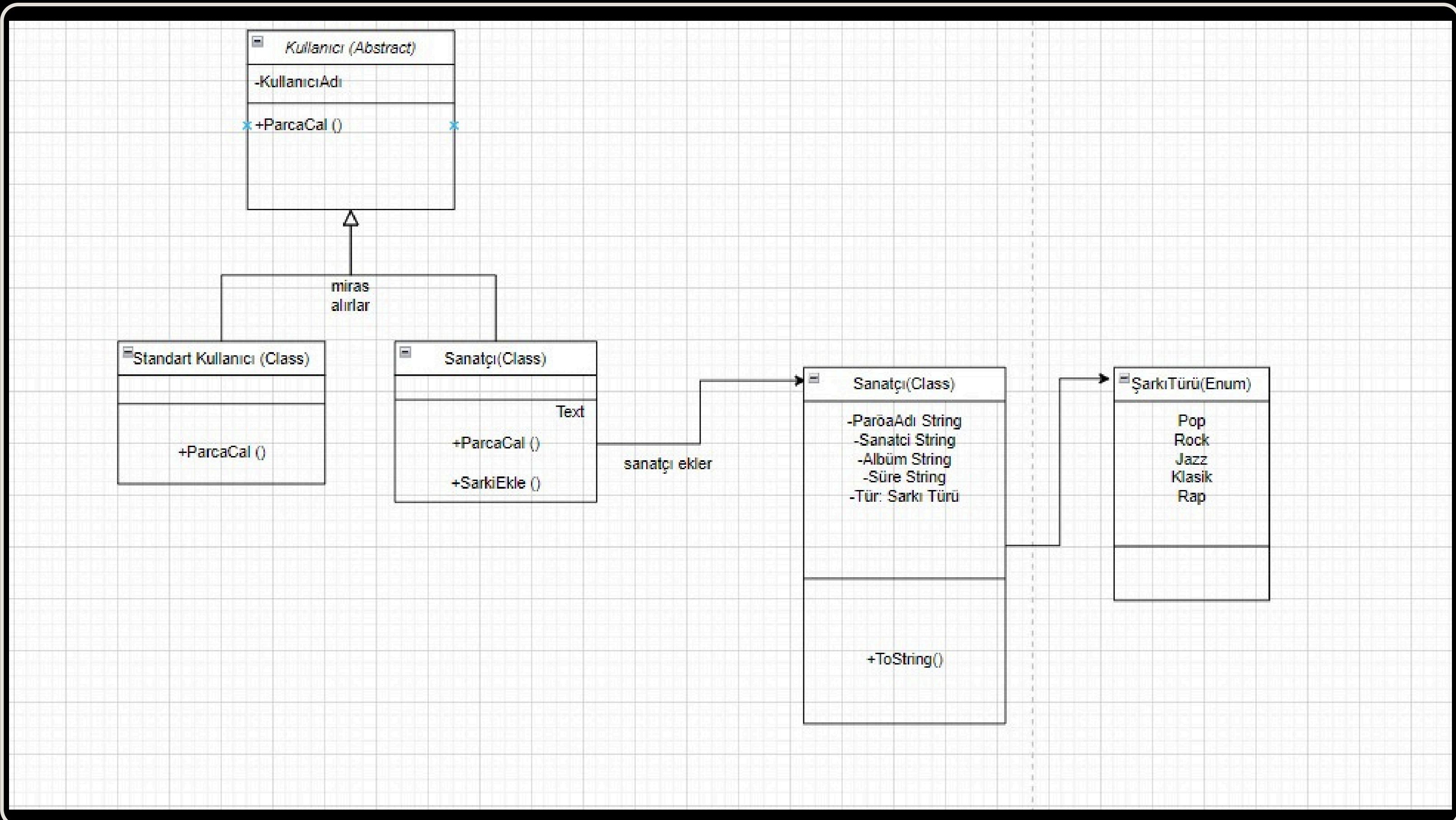
- Kullanıcı çıkışsa, uygulama sonlanır.



FLOW CHART



UML DİYAGRAMI





**BENİ DİNLEDİĞİNİZ
İÇİN ÇOK TEŞEKKÜR
EDERİM!**

**EVLA KARAGÖZ
231041040**