

# KARE PUZZLE OYUNU

160201079 İlknur GÖK,

160201012 Elanur OCAK

## Bilgisayar Mühendisliği Bölümü Kocaeli Üniversitesi

[ilknurgok.1@hotmail.com](mailto:ilknurgok.1@hotmail.com) , [elaocak45@gmail.com](mailto:elaocak45@gmail.com)

### Özet

Bizden herhangi bir geliştirme ortamında kare puzzle oyunu yapmamız istendi. Bunun için bir kullanıcı GUI si oluşturmamız istendi. Dosyadan okuduğumuz herhangi bir resim dosyasını 16 eşit parçaya bölüp rastgele dağıtmamız istendi. Bu parçalardan en az bir tanesi doğru yerinde bulunana kadar kullanıcının karıştır butonuna basması istendi. En az bir tanesi doğru yerde bulunduktan sonra kullanıcıdan iki ayrı resme tıklayıp yerlerini değiştirmesi ve programın bu hamlenin doğru olup olmadığını kontrol etmesi istendi. Doğru parçaları bulma işlemi bitince elde edilen skoru txt dosyasına kaydetmemiz ve oyun her başladığında bu kayıtlı skorlardan en yüksekini programda göstermemiz istendi.

### 1.Giriş

Projenin konusu kare puzzle oyunu yapmaktır.

Projenin amacı, herhangi bir geliştirme ortamında kullanıcıların oynayabileceği bir kare puzzle oyunu hazırlamaktır.

### 2.Temel Bilgiler

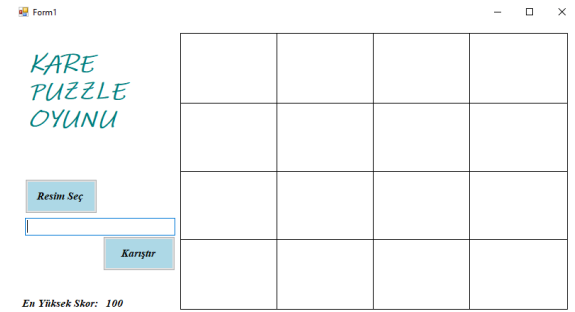
Proje C# dilinde yazıldı ve Visual Studio'da hazırlandı.

- **Microsoft Visual Studio**, Microsoft'un entegre bir geliştirme ortamıdır (IDE). Bilgisayar programlarının yanı sıra web siteleri, web uygulamaları, web

servisleri ve mobil uygulamalar geliştirmek için kullanılır.

### 3.Diğer Bölümler

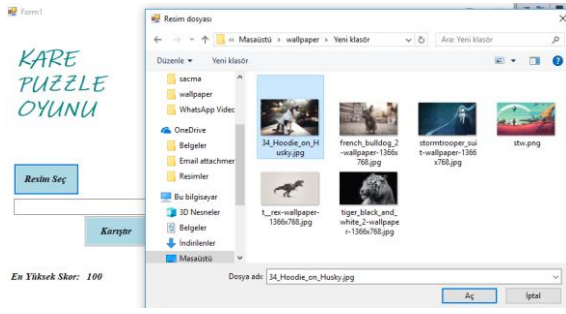
Uygulama ilk açıldığında ekrana gelen formda kullanıcılara iki farklı buton ile iki ayrı işlem gerçekleştirmeyi sağladık. Bu ekran açılır açılmaz ayrıca oyun skorlarından en yüksek olan puanı da burada gösterdik.



Şekil 1-Kare Puzzle Oyunu

#### 3.1. Resim Seç

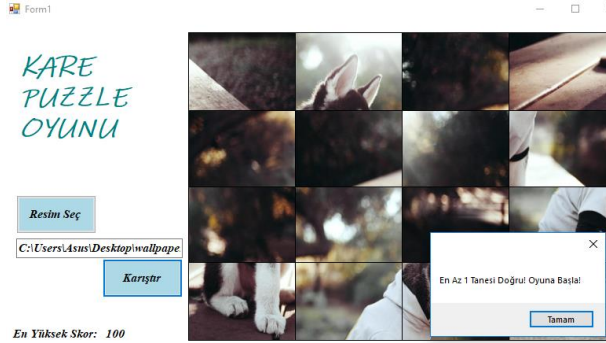
İlk olarak basılan “Resim Seç” butonu ile kullanıcıdan yapboz oyununda yapmak üzere kullanacağı resmi seçmesini sağladık. Resim seçilmesi sağlandıktan sonra resmin yolunu formda bulunan textbox içinde göstermeyi sağladık.



Şekil 2-Resim seçme

### 3.2. Karıştır

Resim seçildikten sonra oyuna başlamak için kullanıcı "Karıştır" butonuna bastıktan sonra önceden seçmiş olduğu resmi 16 parçaya bölüp bu parçaları da karıştırarak ekranda gösterilmesini sağladık. Kullanıcının eşleştirme işlemine başlayabilmesi için bu 16 parçadan en az birinin doğru olması koşulunu da butona her basışta kontrol ettirdik. En az bir parça doğru yerde olana kadar "Karıştır" butonuna basılmasını istedik. Eğer en az bir parça doğru yerde olursa kullanıcıyı diğer parçalar için eşleştirme işlemine geçmesini sağladık yani diğer forma geçiş yaptık.

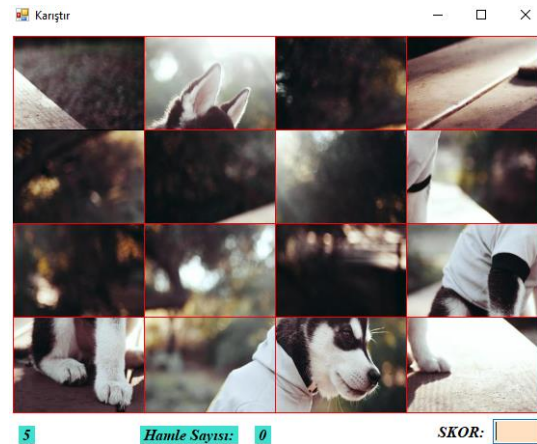


Şekil 3-Karıştır butonuna basma

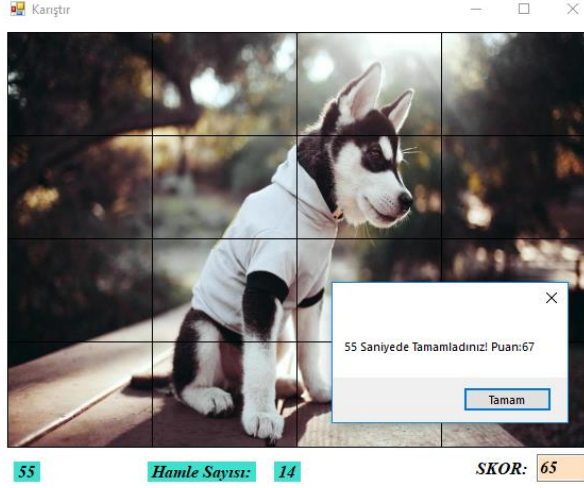
### 3.3. Eşleştirme

Karıştırılan parçalardan en az biri doğru olduğu koşulda eşleştirme işlemine geçildiğinde ekrana ilk olarak önceki formda ekranda karıştırılan parçaların olduğu ve en az bir parçanın doğru yerde olduğu durumun olduğu hali gösterdik ve kullanıcının eşleştirme işlemine doğru yerde en az bir parça olduğu bu durumdan başlamasını sağlamış olduk. Bu ekranda aynı zamanda oyunun süresini de saydırıp süreyi göstermeyi de sağladık. Burada kullanıcı için ayırt edici olması sebebiyle doğru yerde olan ve olmayan parçaları farklı renklerle gösterdik. Doğru yerde

olmayanları kırmızı renkte, olanları ise siyah renkte ve bu parçalara da ayrıca seçilememe özelliği verdik yani resmin bulunduğu butonun click özelliğini kapattık. Kullanıcı değiştirmek istediği parçaları seçtikten sonra o parçaları kendi aralarında değiştirmeyi sağladık. Bu değiştirmeyi yaparken aynı zamanda bu yerleri değişen parçaların doğru yerde olup olmadığının da kontrolünü yaptık. Eğer parçalardan herhangi biri doğru yerde ise doğru olduğunu belirtmek için bu parçayı da siyah renkte gösterip seçilememe özelliği verdik. Eğer parçalardan ikisi de doğru yerde ise ikisine de aynı işlemleri uyguladık. Kullanıcının her eşleştirme işlemi için yaptığı hamlelerin sayısını da ekranda gösterdik, her yer değiştirme işlemi sonrası hamle sayısını artırdık. Kullanıcı tüm parçalar doğru yerde olana kadar değiştirme işlemine devam ettikçe parçaları yer değiştirip doğru yerde olup olmadıklarının kontrolünü yaptık. Tüm parçalar doğru yerde olduğunda ise oyunu bitirip kullanıcının oyunu kaç puanla bitirdiğini skor kısmında gösterdik. Ayrıca kullanıcının kaç saniyede bitirdiğini ve kaç puanla oyunu tamamladığını ekrana yazdırdık. Kullanıcının oyunu tamamlama puanını belirlerken karıştırması sonrası gelen doğru parça sayısına göre yapılacak olan en az hamle sayısını ve kullanıcının oyunu bitirme süresini dikkate aldık. Oyun oynama süresine göre kullanıcının skoruna belirli süreye göre belirlediğimiz puan eklemesinde bulunduk. En sonunda ise kullanıcının bu süreye göre ek puan aldığı oyun skorunu enyüksekskor.txt dosyasına kaydettik.



Şekil 4- En az bir doğru parça sonrası eşleştirme işlemine geçme



Şekil 5- Oyunu tamamlama

#### 4.Genel Sonuçlar

Projede istenen arayüzü oluşturduk. enyüksekskor.txt adlı dosyayı okutup oyunun başında en yüksek skoru yazdırdık ve her oyun bitiminde yeni skoru dosyaya kaydettik. Kullanıcının seçtiği resim dosyasını 16 eş parçaya ayırmayı sağladık. Bu parçaları karıştırıp rastgele butonlara yerleştirdik. En az bir doğru gelene kadar kullanıcının karıştır butonuna basmasını sağladık. Karıştırma işlemi sonrası en az bir parça doğru yere geldiğinde ikinci forma yani eşleştirme işlemine geçtik. Bu formda kullanıcının yer değiştirmesini istediği için üzerine tıkladığı 2 resmin yer değiştirmesini sağladık. Yer değiştirmesi sonrası eğer varsa resimlerden doğru yerde olanı belirledik ve doğru resme sahip olan butonun click özelliğini kapattık. Bu formda kullanıcının oyun oynadığı süreyi saydırdık. Ayrıca karıştırma sonrası gelen doğru parça sayısına göre yapılacak olan en az hamle sayısını baz alarak, kullanıcının oyunda yaptığı hamle sayısına göre oyun skorunu belirledik.

#### 5.Kaynakça

[1]. Web Site

<https://sites.google.com/site/mtaskopru/c-resmi-karelere-ayirmak>

[2]. Web Site

<https://social.msdn.microsoft.com/Forums/tr-TR/59821fdf-cba5-442a-947d-38da4f45f0d7/1-20-aras-rastgele-say-atamak-ve-bu-sayy-bir-daha-atamamak?forum=csharptr>

[3]. Web Site

<https://www.c-sharpcorner.com/uploadfile/prathore/image-comparison-using-C-Sharp/>

[4]. Web Site

<https://social.msdn.microsoft.com/Forums/tr-TR/f93531a5-49e2-4106-8f1d-5d18d53a037d/c-mouse-click-olaylarn-tamamen-kapamak?forum=csharptr>

[5]. Web Site

[https://www.chip.com.tr/forum/c-butonlari-aktif-pasif-yapma-gorsel-anlatim\\_t84587.html](https://www.chip.com.tr/forum/c-butonlari-aktif-pasif-yapma-gorsel-anlatim_t84587.html)

[6]. Web Site

<https://www.yazilimkodlama.com/programlama/c-timer-kullanimi-ve-ornekleri/>

[7]. Web Site

<https://bymutluu.wordpress.com/tag/c-dosya-islemleri/>

## 6.Akış Şeması

