

KARE PUZZLE OYUNU

Ata GÜLALAN - Oğuzhan TÜRKER

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

xavaneo@gmail.com oguzturker8@gmail.com

Özet

Uygulama, kullanıcının seçtiği resmi 16 eşit parçaya bölüp, bunları karşılaştıran ve doğruluğuna göre puanlayan bir puzzle oyunudur.

1. Giriş

Öncelikle başlangıçta kullanıcının bir resim seçmesi gerekmektedir. **Select an Image** butonu ile kullanıcı istediği resmi uygulamaya yükler.

Yüklenen resim 16 eş parçaya bölünür. Her bir kare resmin bir parçasını oluşturmaktadır.

Seçilen resim rastgele karıştırılır. Eğer karıştırılan parçalar arasından doğru yere yerleşmiş parça veya parçalar varsa oyun kullanıcının kullanımına sunulur. Parçalardan herhangi biri ilk karıştırmadan sonra doğru yerde değilse kullanıcının **Shuffle** butonuna tekrar basılması istenir. Kullanıcı oyun kuralları gereği karışım sonrası en az bir doğru parça yerleşene kadar karıştırma işlemine devam eder.

Karıştırma işleminde yerine oturan doğru parçalar sayısınca kullanıcı skor kazanır. Daha sonra karıştır butona pasif hale geçer. Eğer hiçbir parça doğru yere oturmamışsa karıştır butonu aktif kalır ve kullanıcıya puzan yazılmaz.

Kullanıcı istediği parçayı üzerine tıklayarak seçer. Doğru olduğunu düşündüğü yere tıklar ve parçalar birbiriyle yer değiştirir. Eğer kullanıcı parçanın yerini doğru tahmin etmiş ise skor kazanır. Ama seçtiği yer yanlış ise puan kaybeder. Ayrıca kullanıcının doğru düşündüğü parçayla değiştirdiği parça da doğru yere gelirse kullanıcı ekstra skor kazanır.

En yüksek skor sadece ilk karıştırma işleminde tüm parçaların doğru yere gelmesiyle sağlanabilir.

Kullanıcı tüm parçaları doğru yere yerleştirdikten sonra oyun biter ve kullanıcıdan bir kullanıcı adı ister.

Kullanıcı istediği kullanıcı adını yazar ve eğer sağlamışsa en yüksek skorlar listesine girer. Eğer kullanıcı kayıt sırasında kullanıcı adını yazmaz ise uygulama kullanıcıyı **'Anonymous'** olarak kaydeder.

Skor tablosunda en yüksek beş oyuncunun kullanıcı adları ve skorları gözükmektedir.

2. Temel Bilgiler

Proje gelişiminde;
Programlama dili olarak "JavaScript" kullanılmıştır.
Tümleşik geliştirme ortamı olarak "Visual Studio Code" kullanılmıştır.
Çalışma ortamı olarak "NodeJs" kullanılmıştır.
Framework olarak ise "Express" kullanılmıştır.

3. Tasarım

Proje aşağıdaki başlıklar altında geliştirilmiştir.

3.1.1 Arayüz



Şekil 1 - Arayüz ve Butonlar

Puzzle : Uygulamada puzzle bölgesi **640x640**'lık bir alandır. Yüklenen resim bu alanda **640x640** olarak yeniden boyutlandırılır ve istenilen parça sayısına göre parçalara ayrılıp bu bölgede parçalara ilgili resim bölümü atanır. Puzzle parçaları istenilen sayıda dinamik olarak oluşturulmaktadır.

Score (1) : Skor bölümü oyun başlangıcında 100/100'dür. Kullanıcının resim yüklemesiyle birlikte karıştırıldıktan sonra gelen doğru parçasına göre skor değişir. Ve kullanıcının bulmacayı doğru çözümüne göre belirlenir.

Doğru hamle yapmak 6 puan kazandırır.

Yanlış hamle yapmak veya doğru yerdeki parçayı tekrar olduğu yere koymaya çalışmak puan kaybettirir.

Kaybedilen puan sahip olunan puanla doğru orantılıdır.

En yüksek skor olan 100 sadece tüm parçalar ilk karıştırmada denk geldiği takdirde kazanılabilir.

Select an Image (2) : Uygulamaya yüklenilmesi istenilen resim bu buton ile yüklenir. Butona tıkladıktan sonra açılan pencerede istenilen dosya seçilir.

Shuffle (3) : Görüntünün karıştırılmasını sağlayan butondur. Kullanıcı ilk resmi yüklediğinde otomatik olarak bir defa resim rastgele karıştırılır.

Doğru eşleşme sağlanamaz ise kullanıcıdan tekrar karıştırılması istenir.

Karıştırma sonrası doğru parçalar yerlerine geldikten sonra karıştırma butonu pasif hale gelir.

Best Scores : enyüksekskor.txt dosyasındaki skorlar sıralanır en yüksek beş skor veya daha az varsa tümü bu bölümde sıralanır.

Bir oyun bittikten sonra skor kaydedilir ve ardından Best Scores bölümü güncellenir. Eğer yeni kaydedilen skor sıralamadaki en iyi beş skordan daha iyi ise buraya yerleşir ve eski beşinci listeden çıkar.

Ayrıca kullanıcı kayıt sırasında ismini yazmaz ise uygulama kullanıcıyı 'Anonymous' olarak kaydeder.

3.1.3 Hesaplama

Puzzle parçalarının doğruluğu, verilen parçanın puzzle'daki parçalar yardımıyla görüntü işlenmesiyle bulunur.

Resim uygulamaya yüklendikten sonra bir tane kopyası oluşturulur. Karıştırma işleminde veya kullanıcının parçaları hareket ettirmesinde parçaların birbirine benzerliğine bakılır. Seçilen parçalar seçilmiş dizisinde tutulur. İlk parça ve ikinci parça farklı etiketlerle işaretlenir. Parçalar birbirleriyle yer değiştirdikten sonra iki parça için de doğruluk kontrolü yapılır. Doğruluk kontrolünde yer değiştirilen iki parçanın, yer değiştirildikten sonra tüm parçalara oranla eski puzzle parçalarına en benzeyen parçalara göre belirlenir. Parçaların güncel yerleri kopyalanan görüntü üzerinde aranır ve benzerlikleri araştırılır. Benzerlik istenilen seviyenin üstünde ise parçaların doğruluğu belirtilir ve puan kazanılır.

3.1.4 Dosya Yapıları ve Fonksiyonlar

Uygulamada kullanılan fonksiyonlar aşağıda dosyalarıyla birlikte listelenmiştir.

public/js/main.js

Uygulamanın dinamik işlem kısmıdır. Birçok olay burada gerçekleşir.

- **toImageData(image) :** Görüntü canvas'a yazdırılıp dönüştürülür.
- **equal(a,b,tolerance) :** Görüntülerin benzerliği hesaplanır. İstenilen eşiği aşıyorsa true aşmıyorsa false döner.
- **resetGame() :** Oyunu başlatan fonksiyondur. Kontrolcü global verileri sıfırlar. İstenilen miktarda parça bölgesi oluşturur ve bu bölgelere tıklanabilirlik fonksiyonallitesi tanımlar.
- **getImagePortion(imgObj, newWidth, newHeight, startX, startY, scaleX, scaleY) :** Parametre olarak gelen görüntüyü canvaslara çizip boyutlandırır.
- **onImageLoaded(resized) :** Görüntünün yüklenmesini test eder. Yüklenene kadar bekletir. Yüklendiğinde resmi ekrana çizer ve karıştırır.
- **readFile(e) :** İşlenecek resmi uygulamaya yükler. Select an Image butonu ile tetiklenir.
- **fillImages() :** Görüntüyü istenilen miktarda dinamik olarak parçalar. Puzzle parça alanlarına bu görüntü parçalarını atar.
- **changePoints(correction) :** Puan doğruluk fonksiyonudur. Kullanıcının hamlelerine göre puanlama yapar.
- **boxControl() :** Resim parçalarının hepsinin işlenip işlenmediğini kontrol eder. Görüntüler parçalanıp çizdirilene kadar bekler.
- **swapBoxes(boxes) :** Seçilen iki puzzle parçasını yer değiştirir.
- **shuffle() :** Puzzle parçalarının yerlerini rastgele olarak karıştırır.

- **getTop5()** : En iyi beş oyuncuyu “enyüksekskor.txt” ‘den okur ve sıralayıp “Best Scores” bölümüne yükler.
- **saveScore()** : Puzzle bittiğinde kullanıcının skorunu ve kullanıcı adını “enyüksekskor.txt” ‘ye kaydeder.

public/index.html

Uygulamada bulunan elementleri içeren html dosyasıdır.

public/css/styles.css

Uygulama elementlerinin şekillendirildiği stil dosyasıdır.

public/data

Uygulamada kullanılan örnek görsellerin topladığı klasördür.

index.js

Uygulamanın arka planda çalışan servis kısmıdır. Dosyaların birbirleri ve bilgisayar ile iletişimini sağlar.

- **app.post** : Uygulamadan gelen kullanıcı kayıt bilgilerini “enyüksekskor.txt” ‘ye ekler.
- **app.get** : “enyüksekskor.txt” ‘deki bilgileri çeker ve önyüze gönderir.
- **app.use** : Uygulamanın statik sayfalarını root’a bağlar. Böylece istenen porta istek atıldığında eğer bir back-end endpoint eşleşmesi yok ise önyüzdeki sayfa çağırılır.
- **app.listen** : Uygulama kullanımı için ‘4540’ portunu kullanır.

readme.MD

Uygulama kullanım ve kurulum talimatlarını içeren el kılavuzudur.

enyüksekskor.txt

Kullanıcıların skor ve kullanıcı adlarının saklandığı dosyadır.

- Scores dizisi içinde kullanıcı bilgileri name : kullanıcı-adı ve score : puan olarak ikili şekilde saklanır.

3.1.5 Karşılaşılan Sorunlar ve Çözümleri

[1] Puzzle parçası doğruluk testi

Id ve konum kullanmadan puzzle parçalarına doğruluk testi yapmak. Her bir parçanın nerede olduğunu id ve konum bilgisi bulundurmada yapmak için tüm resmi puzzle parçası kadar resme böldükten sonra o parçanın diğer tüm parçalarla olan benzerliğine baktık ve birden fazla eşleşme dahi olsa kabul edilmesini sağladık. Böylece tamamen kırmızı bir resim yüklendiğinde, ne kadar karıştırılırsa karıştırılsın her zaman tam puan alınıyor, çünkü her parça birbiriyle eş.

[2] Görüntü dosya formatları

Tüm görüntüleri ortak bir şekilde işleyebilmek için şekillendirmek adına uygulamaya yüklenen görüntüleri canvas olarak tutup üzerinde işlem yaptık. Canvas ile işlem yapmamız bize performans konusunda çok büyük kazanç sağladı.

[2] Görüntü dosya boyutu

Dosya boyutu çok küçük veya çok büyük olduğunda boyutlandırma yapılması istendiğinde bu resmin istenen parçalara tam olarak bölünmemesi veya Puzzle’ın dışında kalması sebebiyle resmin tekrar boyutlandırılmasını sağladık.

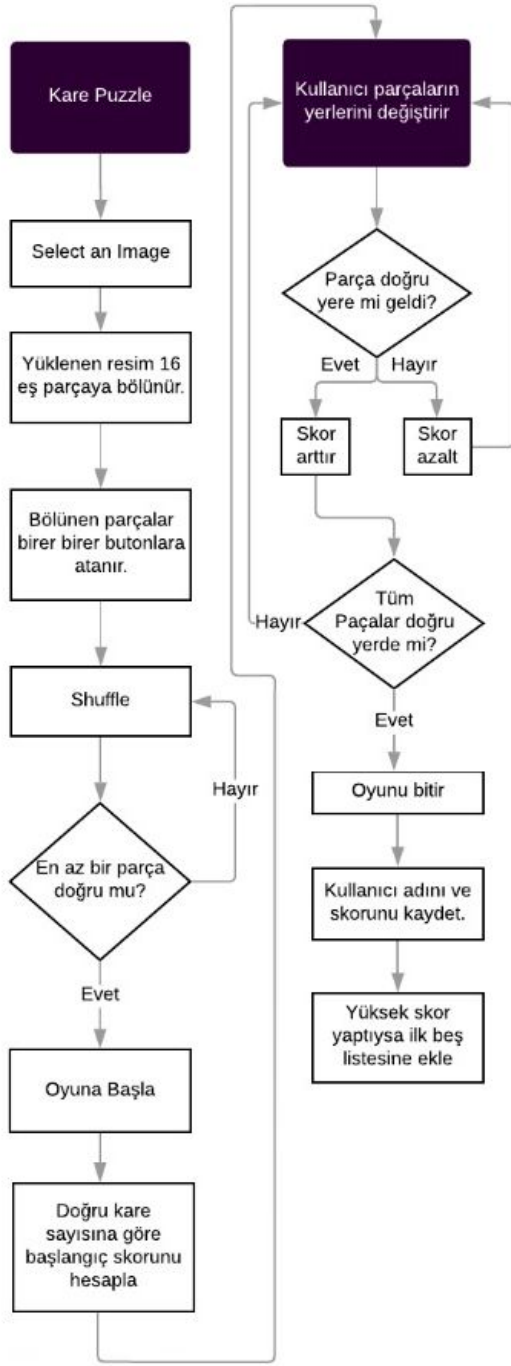
[2] JPEG Artifacts (Bozulmalar)

JPEG formatı sıkıştırılabilen bir format olduğundan kendini 16 kere tekrar eden bir puzzle yüklendiğinde dahi eşleşmenin olmadığını gördük. Bu sorunu çözmek için eşleşme kısmına eşik koyduk ve belirli bir eşliğin üzerine çıktığında bu parçaların eş olduğunun gösterilmesini sağladık.



Şekil 2 - JPEG Artifacts Örneği

4. Akış Şeması



5. Kazanımlar

- Dinamik JavaScript element işlemleri,
- JavaScript ile görüntü işlenmesi,
- Backend fonksiyonlar,
- CSS ile animasyon,
- JSON dosya formatı.

6. Kaynakça

HumbleSoftware / js-imagediff - GitHub

[1]<https://github.com/HumbleSoftware/js-imagediff/>
Erişim Tarihi: 20.03.2019

Resize image with javascript canvas (smoothly) - StackOverflow

[2]<https://stackoverflow.com/questions/19262141/resize-image-with-javascript-canvas-smoothly>
Erişim Tarihi: 20.03.2019

Cropping images with Javascript - Yellow Pencil

[3]<https://yellowpencil.com/blog/cropping-images-with-javascript/>
Erişim Tarihi: 20.03.2019

Array.prototype.sort() - Mozilla Developers

[4]https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/sort
Erişim Tarihi: 20.03.2019

How can I shuffle an array? - StackOverflow

[5]<https://stackoverflow.com/questions/6274339/how-can-i-shuffle-an-array>
Erişim Tarihi: 20.03.2019

Destructuring assignment - Mozilla Developers

[6]https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment
Erişim Tarihi: 20.03.2019

Disable text selection highlighting - StackOverflow

[7]https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment
Erişim Tarihi: 20.03.2019

Exo Font - Google Fonts

[8]<https://fonts.googleapis.com/css?family=Exo+2:300,400,500,600,700,800,900&subset=latin-ext>
Erişim Tarihi: 20.03.2019

Coolers.co - The super fast color schemes generator

[9]<https://coolers.co/app>
Erişim Tarihi: 20.03.2019

Express - Web Server

[10]<https://expressjs.com/>
Erişim Tarihi: 20.03.2019