

# Puzzle Game

Caner Ay Celep  
201307036  
[201307036@kocaeli.edu.tr](mailto:201307036@kocaeli.edu.tr)

Eyüp Ayaz  
201307024  
[201307024@kocaeli.edu.tr](mailto:201307024@kocaeli.edu.tr)

## Abstract

This project case we have to a puzzle game. We splitted sixteen parts of images. When users choices 2 parts of picture ,score will be increasing or descreasing. And you will be seeing your score. We developed webui layer (integrated with api ) for made a mobile version.

*Keywords— splitted, images, api, styling, insert (key words)*

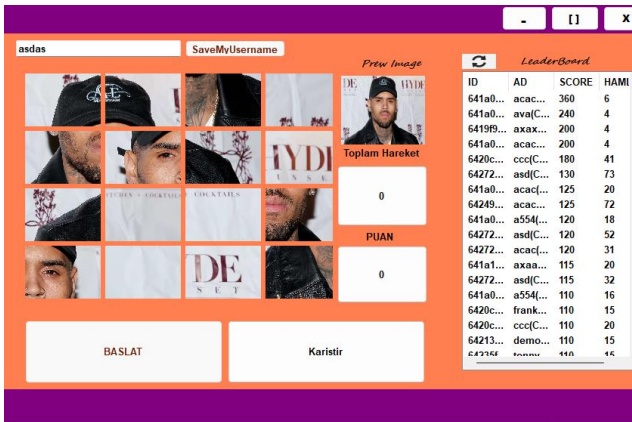
## I. KONUSU

Proje kapsamımızda puzzle oyunu yapmamız istenmekteydi. Bu kapsamda resimlerin 16 parçaya ayrılmış olduğu ve her iki resim üzerine tıklanıldığı zaman ilgili parçaların değiştirildiği bir system yapılması istenmekteydi. Bu bağlamda linkedlist yapısıyla parça takip kontrolü yaptığımız bir oyun geliştirdik. Oyunumuzu yaratırken tek bir platforma bağımlı kalmamak için WebUı katmanı sayesinde api oluşturduk ve verilerimizi dışarıya açmış olduk. Bu sayede mini bir mobil oyun da geliştirmiş olduk.

## II. KULLANIMI

### A. Anasayfa

Oyunumuz .net core teknolojisi ile yapıldığı için herhangi bir platform bağımlılığı yoktur. Oyunumuz masaüstü uygulaması olduğu için setup projesi oluşturup exe'li halini hazırlamış olduk. Bu sayede isteyen kişiler oyunun masaüstü versiyonunu indirip oynayabilir. Oyunumuzu tek bir platforma bağlı tutmamak için WebUı katmanımız ile apiler aracılığıyla verilerimizi açığa çıkardığımızdan bahsetmiştik bu bağlamda verilerimiz sanal bir ortamada bulunması gerekiyordu bunun için verilerimiz sanalda tutulan bir veritabanında olduğu için her yerden erişebilir haldedir. Bu sayede skor bilgileri resim bilgileri gibi bilgiler gelmektedir. Oyunumuza ait anasayfa ve listelerin bulunduğu bilgiler Şekil-1 ' de gösterilmektedir.



Şekil-1

### B. İşleyiş

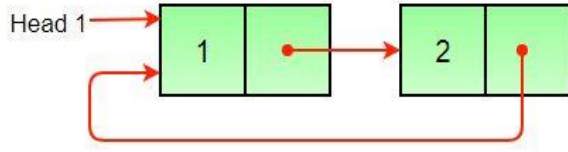
Ressimleri parçalayabilmemiz için Bitmap sınıfının özelliklerinden yararlanmamız gerekiyordu. Bu sayede resimleri istediğimiz parçalara göre bölebilir ve bir dizinin içerisine atabiliriz. Bu bağlamda linkedlist yapısını kullanarak bir dizimizi oluşturduk. Ve içerisine elemanların form ekranı yüklenirkenki değerlerini atmış olduk. Kullanıcı iki resim arasında seçim yaptığı zaman yani resimlerin yerlerini değiştirdiği zaman bu parçaların doğru olup olmadığı linkedlist yapımız tarafından değerlendirilmesi için bir method tanımladık. Bu sayede kullanıcı doğru hamle yapmadığı belli olmuş oluyor. Kullanıcıların puanları ise çoktan aza doğru yani yüksek olan puandan düşük olan puana doğru bir sıralamada kullanıcılara bir liste olarak gösterilecektir. Bu sayede kullanıcılar yapmış oldukları skorları ve bilgileri buradan bakarak bir bilgi sahibi olabilirler. Kullanıcılar her oyun sonu projenin ilgili klasöründe .txt adında oyunu bitirdiğine dair bir belge de sahip olmuş olurlar. Şekil-2 'de skor tablosu gösterilmektedir.

ID	AD	SCORE	HAMI
641a0...	acac...	360	6
641a0...	ava(C...	240	4
6419f9...	axax...	200	4
641a0...	acac...	200	4
6420c...	ccc(C...	180	41
642ae...	caner...	150	3
64272...	asd(C...	130	73
641a0...	acac(...	125	20
64249...	acac...	125	72
641a0...	a554(...	120	18
64272...	asd(C...	120	52
64272...	acac(...	120	31
641a1...	axaa...	115	20
64272...	asd(C...	115	32
641a0...	a554(...	110	16
6420c...	frank...	110	15
6420c...	ccc(C...	110	20
64213...	demo...	110	15

Şekil-2

## III. KULLANILAN TEKNOLOJILER

Projemizde çapraz platform desteği için asp.net kullandık. Projemizde parça kontrolleri içinde linkedlist yapısını kullanmış olduk. Mobil halini herhangi bir platforma bağlı kalmadığımızı göstermek için react-native ile geliştirmiş olduk bu sayede kullanıcılarımız herhangi bir platforma bağlı olmadan oyunumuzu istedikleri zaman oynayabilirler. Linkedlist yapısını daha iyi anlamak ve parça takibini nasıl yaptığımıza daha yakından bakmak için ilgili adımı Şekil-3 ' de gösterilmiştir.



Şekil-3

#### A. Oynanış Hakları

Kullanıcı oyunu oynarken kullanıcının sahip olduğu özelliklere de değinmek isteriz. Kullanıcı oyunumuza başladığı zaman ve oynarken bu özelliklerden yararlanabilir ve oyunu bu şekilde oynayabilir :






#### B. Haklar

- İki ressim arasında seçim yapabilir.
- Ressimleri karıştırabilir.
- Kendine özel takma isim kullanarak oyuna başlayabilir.

#### C. Admin

Projemizde bir WebUI katmanı olduğundan bahsetmiştik bu katmanımızı sadece api geliştirmek için değil aynı şekilde projemize yeni resimler eklemek , silmek ve güncellemek için de kullanmış olduk. Bu sayede istenildiği zaman yeni resim eklemeleri yapılabilir veya varolan bir resim üzerinde silme , ekleme , güncelleme işlemlerini yapmış olabiliriz. İlgili katmanımıza ait bilgiler Şekil-4 de verilmektedir.






PuzzleGame.WebUI Home Add Images

Picture Name	Picture Ico	Update
Chris Brown		<button>Update</button>
Dusunceli Caillou		<button>Update</button>
EdEddy		<button>Update</button>
Lamborghini		<button>Update</button>
Pepe		<button>Update</button>

Şekil-4

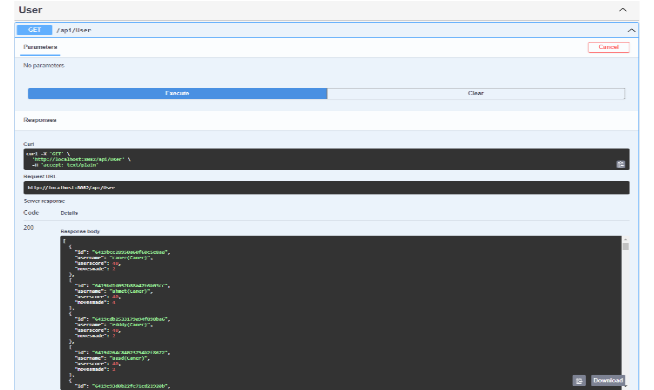
Admin panelimizde tüm işlemleri tek bir yerde tutmak yerine güncelleme ve silme işlemi için ayrı bir sayfa yaratmış olduk bu sayede kullanıcılar bir işlem birimine tıkladığı zaman ilgili sayfaya yönlendirilmiş olacak. Şekil-5’de ilgili bilgiler gözükmektedir.

PuzzleGame.WebUI Home Add Images

Picture Name	Picture Ico	Update
Chris Brown		<button>Update</button>
Dusunceli Caillou		<button>Update</button>
EdEddy		<button>Update</button>
Lamborghini		<button>Update</button>
Pepe		<button>Update</button>

Şekil-5

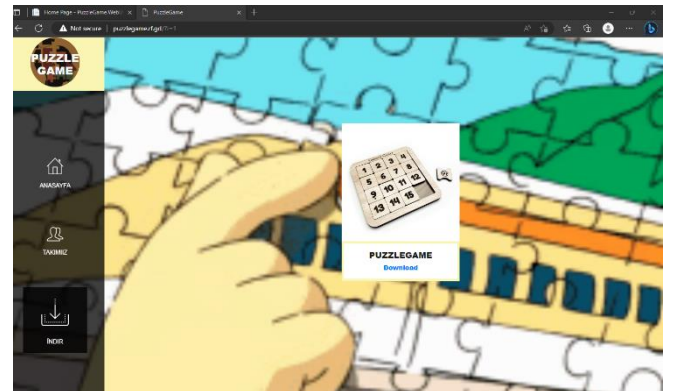
Api katmanımızda bulunan bilgilere ulaşmak için bir fetch işlemi gerekiyor bunun için mobil uygulamamız tarafında axios paketini kullandık bu sayede fetch ve post işlemlerimizi hızlı ve basit bir şekilde gerçekleştirmiş olduk. Api katmanımızdan alınan bir yanıt Şekil-6’da gösterilmektedir.



Şekil-6

#### D. WEB KISMI

Web kısmında hazır template kullanarak projemizi web sayfasına uyarladık PuzzleGame isimli web sayfasımız 3 sayfa dan oluşmaktadır. Bunlar anasayfa, takımımız ve indirme sayfalarıdır.

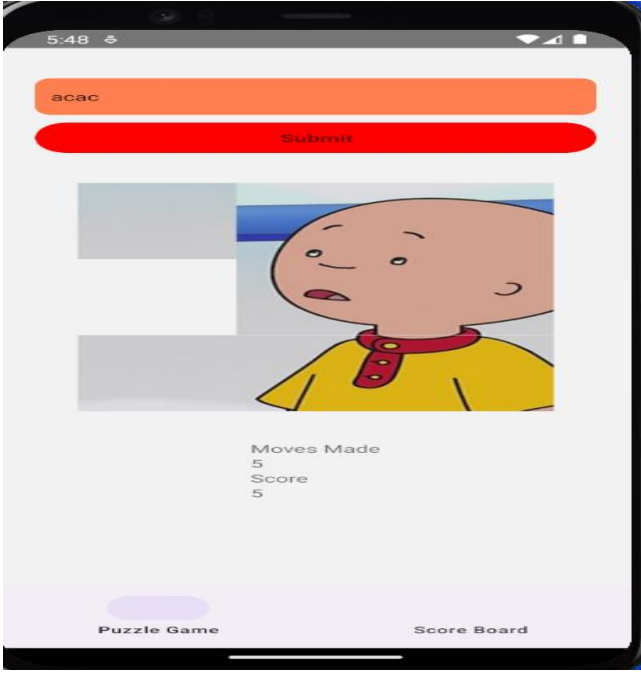


Şekil-7

Şekil 7 de görüldüğü üzere indir sayfasında oyunu indirmemiz için bir Download butonu bulunmaktadır. Projemizi setup haline getirerek kullanıcıların oyunu web sayfasından kolay bir şekilde indirip masaüstüne kurarak oynama imkanı sağlandı.

#### E.Mobil Kısım

Api ile açmış olduğumuz verileri kullanarak mobil uygulama geliştirmek için ilk adımları bir yazılım dili seçmek oldu.React-native hem çapraz platform desteklemesi hem de javascript nimetlerinden yararlandığı için react-native dilini seçmiş olduk.Mobil uygulamamızın anasayfasına ilişkin ilgili görsele Şekil-8’de görebilirsiniz.



Şekil-8

Projemizde kullanıcıların önceki kullanıcıların yapmış olduğu Skorları görebilmeleri için de ikinci bir sekme hazırladık.Bu Sayede toplam skorlarını ve bilgilerini görebilirler.İlgili resim Şekil-9’de gösterilmektedir.

AD	SCORE	MOVES
caner(Caner)	40	2
ahmet(Caner)	40	4
edddy(Caner)	40	2
aasd(Caner)	40	2
caner(Caner)	40	2
ax2y(Caner)	40	4
ax2y(Caner)	50	8
ax2y(Caner)	70	14
ax2y(Caner)	80	16
axax(Caner)	40	4
axax(Caner)	60	8

ŞEKİL-9

#### KAYNAKÇA

- [1] <https://stackoverflow.com/questions/9484935/how-to-cut-a-part-of-image-in-c-sharp>
- [2] <https://stackoverflow.com/questions/69535928/react-native-split-one-image-to-multiple-parts>
- [3] <https://www.npmjs.com/package/react-native-slice-image>
- [4] <http://puzzlegame.rf.gd/>