



FIRAT ÜNİVERSİTESİ

TEKNOLOJİ FAKÜLTESİ

Yazılım Mühendisliği Bölümü

YMH459 – Yazılım Mühendisliği Güncel Konular Proje Dökümantasyonu

KidsGames

Geliştiren

195541027 - Erhan Baran

Proje Yürütücüleri

Doç. Dr. FATİH ÖZKAYNAK

Arş. Gör. BEYZANUR DURMUŞ

Mayıs -2023

İÇİNDEKİLER

1- Giriş

1.1 Projenin Tanıtımı

1.2 Projenin Amacı

1.3 Projenin Kapsamı

1.4 Kullanılacak Teknolojiler

2- Proje Planı

2.1 Sistemin Kullanıcıları

2.2 İşlevsel İhtiyaçlar

2.3 İşlevsel Olmayan İhtiyaçlar

3- Proje Gerçekleştirimi

3.1 Modüllerin ve Formların Tasarımı

4- Projede Öngörülen Eksiklikler

4.1 Proje Planında Yapılması Planlanmış Ancak Eksik Kalan Modüller

4.2 Projeye Eklenmesi İçeriğini Zenginleştirecek Modüller

4.3 Projeye yapılan değerlendirmeler ve Test süreci

5- Sonuç

1. Giriş

1.1 Projenin Tanıtılması

KidsGames, gelecekte çeşitli oyunları bünyesinde barındırmayı hedefleyen, ancak özellikle 7 yaş ve üzeri çocuklar için tasarlanmış bir mobil oyun platformudur. Şu an itibarıyla platformda bulunan "Tombala" oyunu, çocukların matematiksel düşünme yeteneklerini geliştirmeye yönelik tasarlanmıştır. Bu oyun, çocuklara sadece eğlence sunmakla kalmayıp, aynı zamanda problem çözme ve stratejik düşünme yeteneklerini de geliştirecek şekilde tasarlanmıştır. En son yapılan eklemelerle birlikte, platforma iki yeni oyun daha eklenmiştir. Bu oyunlar çocukların pattern design mantığını anlamalarına yardımcı olmayı ve ayrıca matematiksel modellemeyi öğrenmeyi amaçlar. KidsGames'in genel amacı, çocukların genel gelişimine katkıda bulunmak ve öğrenmeyi eğlenceli hale getirmektir.

1.2 Projenin Amacı

Projenin çıkış amacı ülkemizde yaşanan deprem sonrasında özellikle en çok etkilenen yaş gruplarından olan çocukların psikolojik iyileştirme sağlayacak ve bir yandan da geliştirici faydalı oyunlar platformu olmayı amaçlıyor. Projenin geliştiricisi olan bendeniz de bu depremzedelerin içerisinde bulunmaktan kaynaklı bu proje amacını, fikrini bize aşıl原因 ve proje yürütücülerinden olan Doç. Dr. FATİH ÖZKAYNAK' a teşekkürü burada yeniden bir borç bilirim.

1.3 Projenin Kapsamı

KidsGames, başlangıçta depremzede çocukları hedef alarak geliştirilmiş olsa da, şu anda 7-15 yaş arası tüm çocuklara hitap eden bir oyun yelpazesine sahiptir. Bu platform, geniş bir yaş grubuna hitap ederek çocukların farklı aşamalarında gelişimlerine destek olmayı amaçlamaktadır.

1.4 Kullanılacak Teknolojiler

KidsGames'in geliştirilmesinde kullanılan ana teknoloji Unity'dir. Bu çok yönlü ve güçlü oyun motoru, çeşitli türde ve karmaşıklıkta oyunları geliştirilmesine olanak sağlar. Ancak, projenin ilerleyen aşamalarında veritabanı gerekliliği ortaya çıkarsa, MongoDB veya PostgreSQL gibi modern ve güçlü veritabanı yönetim sistemleri de kullanılabilir.

2. PROJE PLANI

2.1 Sistemin Kullanıcıları

KidsGames'in hedef kullanıcı kitlesi, genel tema ve oyunları göz önünde bulundurarak, 7-15 yaş arası çocuklardır. Bu platform, çocukların eğlenceli bir şekilde öğrenmesine yardımcı olmayı hedeflerken, aynı zamanda onların gelişim süreçlerini desteklemeyi amaçlamaktadır.

2.2 İşlevsel İhtiyaçlar (Olmazsa Olmazlar)

KidsGames'in işlevsel olarak en önemli ihtiyacı, hedef yaş grubunun mevcut olanakları dahilinde indirebileceği ve oynayabileceği bir platform olmasıdır. Bu platformun, birçok farklı cihaz ve sistem özellikleriyle uyumlu olması ve ortalama bir telefonun bile yüksek performansla çalıştırabileceği bir oyun platformu olması, işlevsel ihtiyaçların başında gelmektedir.

2.3 İşlevsel Olmayan İhtiyaçlar (İlave Özellikler)

Projemiz, belirli bir hedef ve amaca sahip olduğu için, ilave özellikler konusunda herhangi bir ihtiyaç duyulmamıştır. Ancak, projenin ilerleyen aşamalarında, kullanıcı deneyimini daha da zenginleştirecek ilave özellikler eklemeyi düşünüyoruz. Bu özelliklerin neler olacağı kullanıcı geri bildirimleri ve projenin ilerleyen aşamalarındaki ihtiyaçlara bağlı olacaktır.

3. PROJE GERÇEKLEŞTİRİLMESİ

3.1 Modüller ve Formların Tasarımı



Resim 1.1

Uygulama Açıldığında Karşılaşılan Ekran

KidsGames uygulaması açıldığında, kullanıcıyı neşeli ve çocuk dostu bir arayüz karşılar. Butonlar üzerine gelindiğinde ortaya çıkan ufak büyüme animasyonları, uygulamanın interaktif ve eğlenceli doğasını yansıtır. Bu tasarım, hedef kitemiz olan çocuk grubuna hitap etmek üzere özenle hazırlanmıştır.

Bu ana ekranın tasarımındaki temel amaç, projede birden fazla oyunu entegre etmeyi planladığım için, kullanıcıları hızlı ve kolay bir şekilde seçtikleri oyunlara yönlendirmektir. Bu nedenle, "Oyna" butonuna tıkladığında, kullanıcıları bir oyun seçim ekranına yönlendiriyoruz.

Bu tasarım sayesinde, çocuklar hangi oyunu oynamak istediklerini kolayca seçebilir ve hızlı bir şekilde oyun deneyimine dalebilirler.

Tasarım konusunda daha iyi bir iş çıkartabilirdik mi? Evet, kesinlikle daha gelişmiş ve karmaşık tasarımlar olabilirdi. Ancak, bu proje Unity'yi kullanmaya yeni başladığım bir dönemde gerçekleşti ve hedefim, basit ama işlevsel bir arayüz oluşturmaktı. Bu konuyu daha sonra commitlerimde de daha ayrıntılı bir şekilde ele alacağım.

Ana sayfamız, temel bir kullanıcı arayüzüne (UI) dayanıyor ve kullanıcıyı çeşitli oyunlara yönlendirmek için tasarlandı. Ek olarak, sol alt ve sağ alttaki butonlar, kullanıcıları farklı ekranlara yönlendiriyor. Bu butonlara tıkladığında, kullanıcıları karşılayan ekranlar şu şekildedir: (Ekranların tanımlamasını buraya ekleyebilirsiniz.)

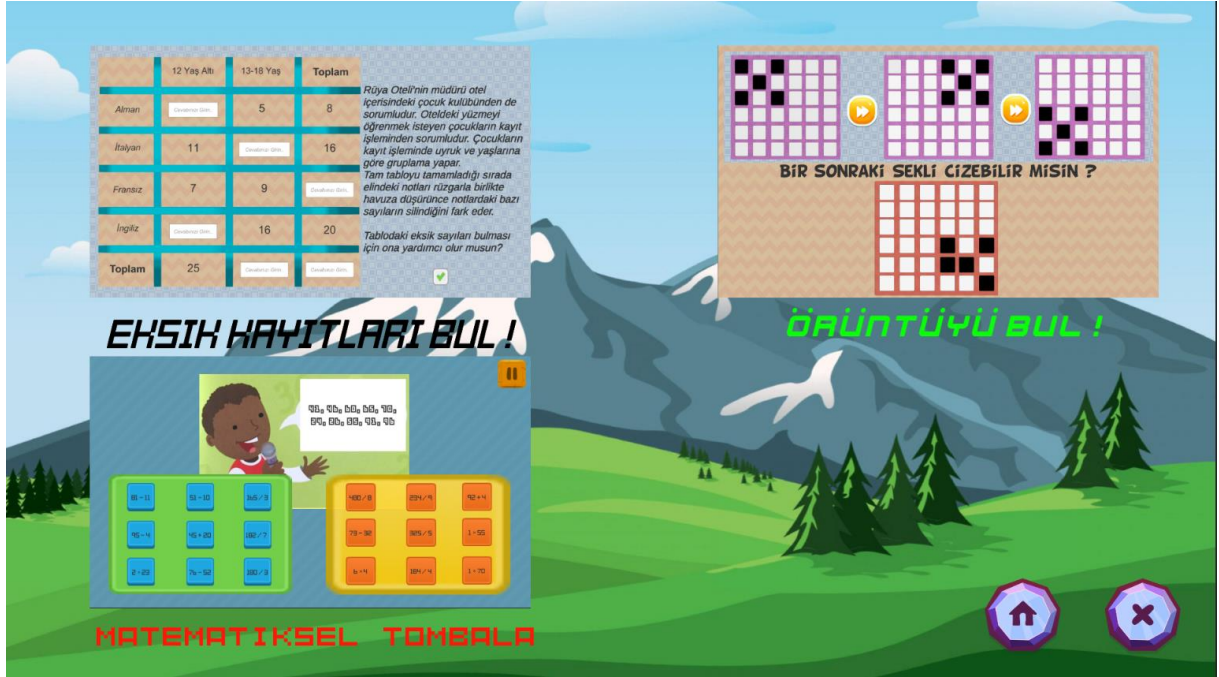


Resim 1.2

[Uygulama Açıldığında Karşılaşılan Ekran v2](#)

Burada bahsettiğim butonlara basıldığında yukarı ve aşağı yönlü olmak üzere açılıp kapanan bir animasyon ile butonlar açılıyor. Soldaki butonlarda bir yönlendirme yapmıyorum fakat bu oyun platformu altından bir markalaşma yapılmak istenirse bu platformun sayfalarına yönlendirme yapılabilir.

Sağ tarafta bulunan ayarlar butonuyla gelen “Ses Aç” ve “Ses Kapat” butonları ise oyun açıldığında çalan müziği açıp kapamaya yaramaktadır.



Resim 1.2.1
Oyun Seçme Ekranı



Resim 1.2.2
İlk Oyun: Eksik Kayıtları Bul



Resim 1.2.3

İkinci Oyun: Pattern Game

Son iki resimdeki oyunla ilgili şunları söylemek istiyorum. Bu dönem bizim 2 kart yani 2 oyun yapmamız gerekiyordu. Ama ben 3. Bir oyun olan resim 1.2.3'deki pattern game'i de yaptım. Çünkü hem bu alanda gelişmek istiyorum, hem de belki oyunlar fikir olarak beğenilirse görsel olarak değiştirildiğinde iyi yerlere gelebileceğine inanıyorum. Öncelikle ilk oyunda özellikle günümüzdeki liseye geçiş sınavlarında öğrencilere sorulan uzun hikayeli ama matematiksel modelleme yapılarak çok basit formülüze edilebilen sorulardan sordum. Ve arka tarafta kodunu geliştirirken çok algoritmasını oluşturmak çok keyifliydi.

Bir diğer oyun Pattern Game. Bu oyunda çocukların örüntüyü devam ettirmeleri noktasında gelişmelerini sağlamak istedim. Buradaki kodu inceleyecek olursak aslında arka tarafta işin matematiğinin çok basit olduğu görülebilir. Öncelikle çizim için bir 2 boyutlu dizi matrisi oluşturdum. Ardından yukarıda gözüken 3 adımı belirli isimlendirme ile örneğin H1,H2,H3 sırasıyla png olarak oraya yerleştirdim. Sonucunda da basit bir şekilde bu oyunu tamamlamış oldum.



Resim 1.3

Üçüncü Oyun: Tombala Oyunu

Burada göze en çok çarpan şey oyunun çok uyumsuz renkler içermesi. Gerçekten hem bulduğum assetleri seçme konusunda hem de genel olarak renk – buton her anlamda uyumsuz şeyler seçmemde tasarım –frontend- alanında ne kadar zayıf olduğumu bir kez daha gördüm.

Şimdi buradaki kurguyu anlatmadan önce benim projemin fikri olan kartı anlatayım ardından bu kısmın nasıl oynandığına geçebilirim.

Tombala

Tolga ve Levent bir çeşit tombala oynuyorlar. Bir sayı söylendiğinde, bu sayıyı veren işlemin bulunduğu karenin üstünü çizmeleri gerekiyor. Aşağıdaki tombala kartlarını ve söylenen sayıları inceleyin. Sonra da kartını ilk bitiren kişinin adını daire içine alın.



12, 76, 21, 33,
64, 90, 84, 8,
59, 22

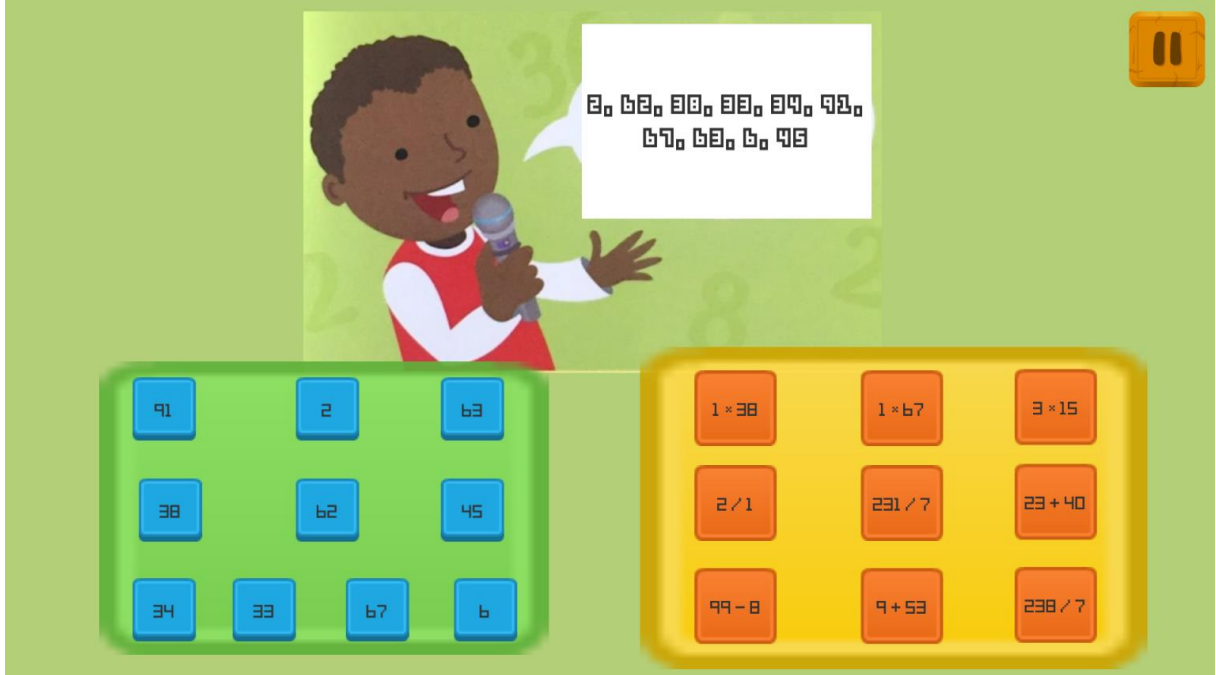
Tolga

45×2	$22 - 14$	$99 \div 3$
$52 + 24$	$84 \div 7$	$70 - 11$
4×16	$63 + 21$	$57 - 36$

Levent

6×14	$108 \div 9$	$16 + 43$
$98 - 22$	$17 + 47$	5×18
$96 \div 12$	$73 - 51$	$18 + 15$

Özetleyecek olursam: 3 kişi bu oyuna başlıyor. Birisi kartların içerisindeki işlemlerin sonucu olan sayılardan; rastgele seçerek diğer iki kişi de bu sırada kart listesinden bu sayıları kartlarındaki işlem sonuçlarıyla eşleştiriyor. Fakat ben bu kişinin söylediği sayıların hepsini tek tek eşleştirdiğimde 10 adet sayının 8 tanesi iki kişide ortak diğer iki kişide ise 1'er sayının farklı olduğunu gördüm. Bu yüzden bu oyunun algoritmasını kendimce biraz daha farklı yorumladım. Şimdi ben nasıl yorumladım oradan devam edeyim.



Resim 1.4

Tombala Oyunu v2

Öncelikle benim algoritmam yukarıdaki kişi yine rastgele olarak tüm sayıları veriyor. Fakat bu sayılardan 1'er tanesi yine farklı olduğu için orada o kişiler şaşırabiliyor. Yani oyunu zor kılan nokta burası.

Resim üzerinden daha da açık olacak olursam: Sağ taraftaki oyuncunun şuan hiçbir işleme tıklamadığını görüyoruz. Eğer tıklarsa sağdaki kişi gibi bir panel açılıyor. Aslında bu panel yukarıdaki kişinin söylediği sayıları içeren bir panel. Fakat oyunun zorluğu oradaki sıra ile oyunculara gelen sıra her seferinde karıştırılıyor. Oyundaki algoritmamı temel olarak 2 fonksiyon 1 class oluşturuyor.

Birinci fonksiyon: Rastgele 10 adet sayı üreten ve bu sayı listesinden dokuzuncu sayıyı “Birinci oyuncu”, onuncu sayıyı ise “İkinci Oyuncu” listesine atıyor. Bu sayede iki oyuncunun da toplam 9 işlem butonu, 10 adet tahmin sayısı ve 8 tanesi ortak 1’er tanesi farklı sayı listeleri oluyor.

İkinci fonksiyon: Oluşan bu sayı listelerini kartlara butonlara dağıtan ve genel oyun kurgusunu sağlayan fonksiyonum.

Son olarak class’ım ise sağdaki şekilde rastgele sayılara göre işlem üretecek sınıfımdır. Bu sınıfı kullanarak rastgele üretilen sayı dizisinden işlemleri üretiyorum. Ve sonrasında o işlemleri butonlara dağıtıyorum.

```
class Operations
{
    private static System.Random random = new System.Random();

    public static Tuple<int, int> GenerateAddition(int result)
    {
        int firstNumber = random.Next(1, result);
        int secondNumber = result - firstNumber;
        return Tuple.Create(firstNumber, secondNumber);
    }

    public static Tuple<int, int> Generatesubtraction(int result)
    {
        int firstNumber = random.Next(result, 101);
        int secondNumber = firstNumber - result;
        return Tuple.Create(firstNumber, secondNumber);
    }

    public static Tuple<int, int> GenerateMultiplication(int result)
    {
        int firstNumber = random.Next(1, Math.Min(result + 1, 10));
        while (result % firstNumber != 0)
        {
            firstNumber = random.Next(1, Math.Min(result + 1, 10));
        }
        int secondNumber = result / firstNumber;
        return Tuple.Create(firstNumber, secondNumber);
    }

    public static Tuple<int, int> GenerateDivision(int result)
    {
        int secondNumber = random.Next(1, Math.Min(result + 1, 10));
        int firstNumber = result * secondNumber;
        return Tuple.Create(firstNumber, secondNumber);
    }
}
```

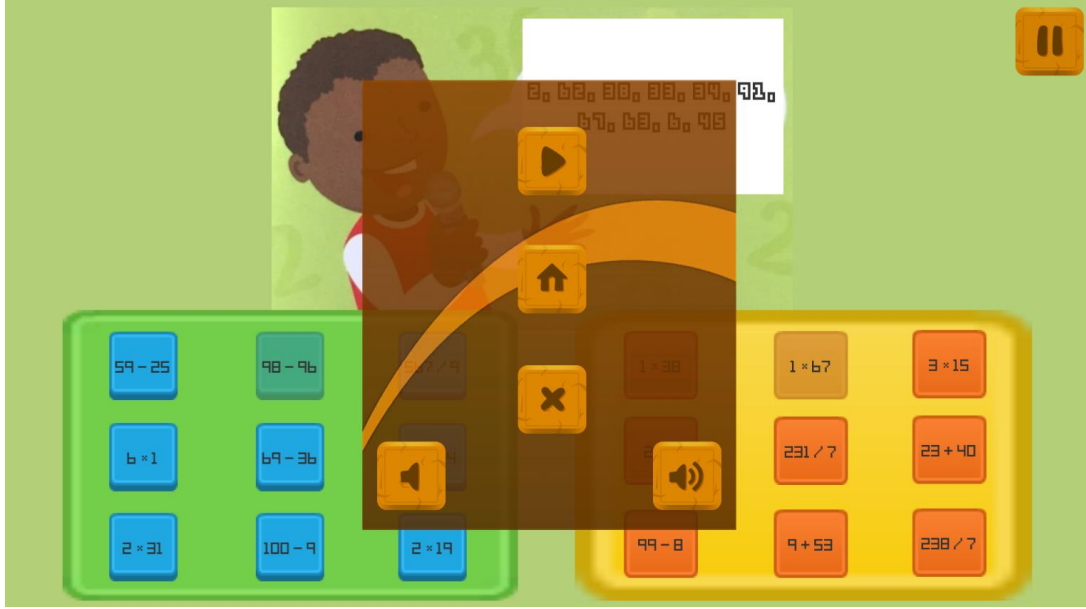
Tabii oyunun tamamının kurgusunu bunlar oluşturmuyor daha birçok ekran ve butonlar var ama ana omurgayı oluşturan algoritma bu fonksiyon ve sınıflar üzerinden ayağa kaldırılıyor.



Resim 1.5

[Tombala Oyunu v3](#)

Burada doğru seçimlerin ardından oyunun ilerleyişini görebiliriz. Örneğin birinci oyuncunun “98-96” işlemine yaptığı “2” seçiminden sonra hem bu tarafta bu işlem seçilemez hale geliyor hem de sonuç seçme tarafındaki “2” seçimi bloke edilmiş oluyor. Bu da oyuncuların oyunun sonlarına gelince yapmış olduğu doğru seçimlerinden dolayı daha da hızlanmasını sağlıyor. Bu şekilde ilk tüm butonlarını yani işlemlerini bitirirse o oyuncu kazanmış oluyor. Kazanma ekranından önce; ekranın sağ üstünde bulunan durdurma butonuna ve ardından gelen panel ekranını inceleyecek olursak.



Resim 1.6

Tombala Oyunu v4 – Pause Ekranı

Resim 1.6’da gördüğümüz şekilde basit bir pause menüsü bizi karşılıyor. Oyuna devam etme, ana menüye dönme ve oyundan çıkma butonları mevcut. Bunlara ek olarak oyun esnasında ana menüdeki müzikten farklı bir müzik çaldığı için ayrıyeten ses açma ve kapama tuşları mevcut. Son ekranlarımızdan olan kazanma ekranlarını da inceleyip genel proje kritiklerine geçebiliriz.



Resim 1.7

Tombala Oyunu v5 – Birinci Oyuncu Kazandığında Karşılayan Ekran



Resim 1.8

Tombala Oyunu v5 – İkinci Oyuncu Kazandığında Karşılayan Ekran

4. PROJEDE ÖNGÖRÜLEN EKSİKLİKLER

4.1 Proje Planında yapılması planlanmış ancak eksik kalan modüllerin listesi

Şuan için eksiğim yok hatta fazladan bir oyun daha yapma fırsatım oldu. Dediğim gibi tasarım olarak daha güzel şeyler ortaya çıkarılabilirdi fakat ben elimden geleni yaptığımı düşünüyorum.

4.2 Projeye eklenmesi içeriği zenginleştirecek modüllerin listesi

Geliştirdiğim oyun seçme ekranında aslında birden fazla oyunumuz olsa ve orayı kategorize edebilssek çok daha güzel olurdu. Örneğin bu sahneden sonra eğer bu alanda devam edilirse oyunlar bu sahnenin altından kategorize edilebilir “Bilim, Matematik, Bulmaca” gibi alt kategorilerle bir oyun platformu oluşturulabilir.

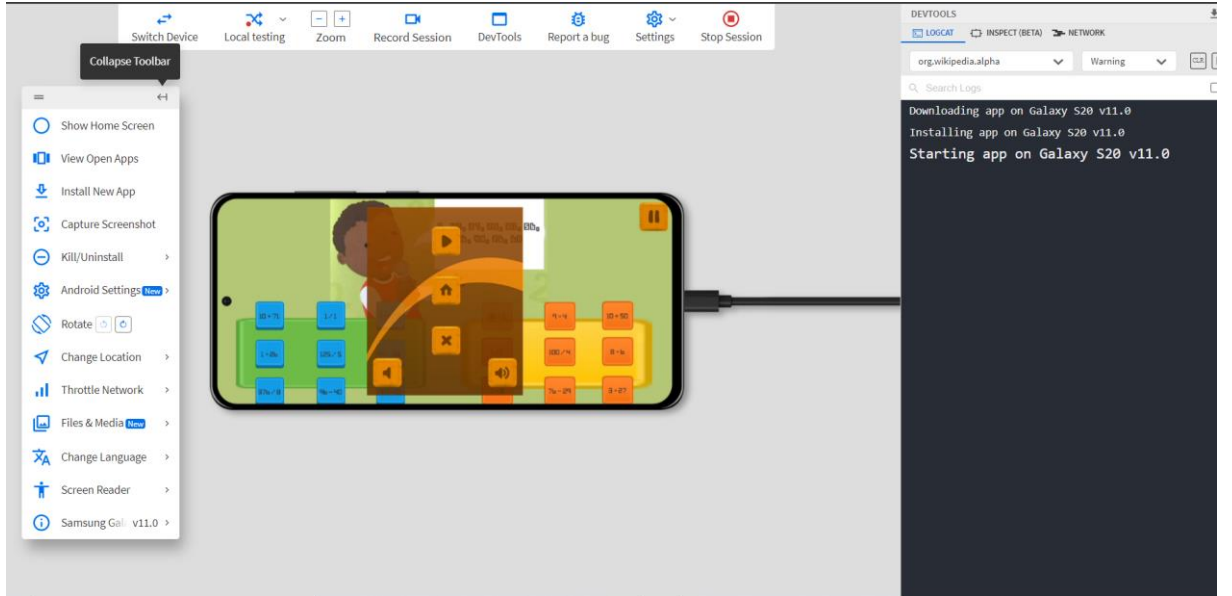
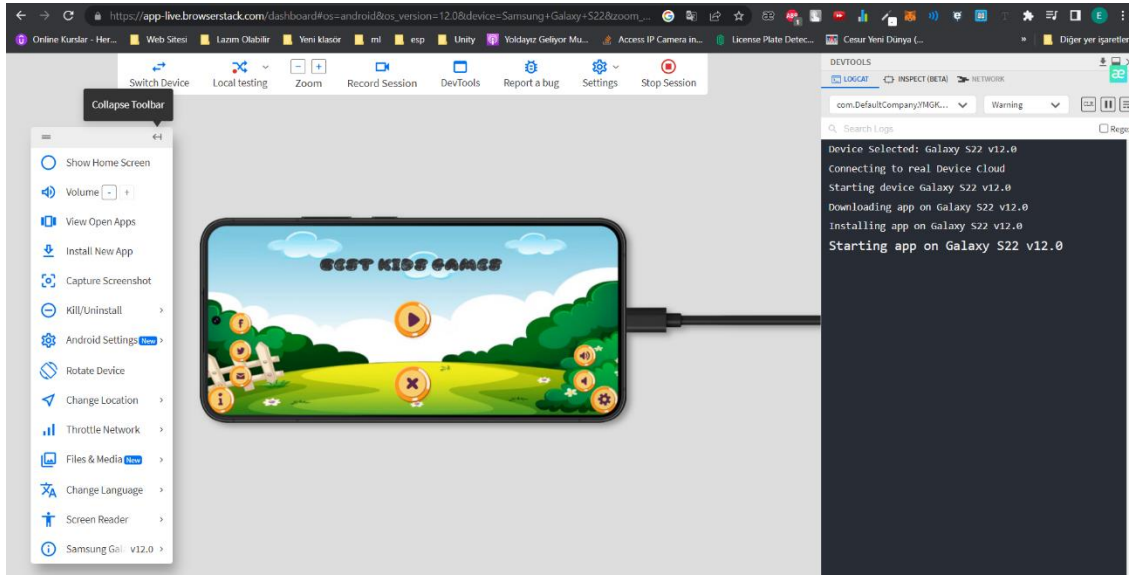
4.3 Projeye yapılan değerlendirmeler ve Test süreci

Bu oyunun son halini 13 kişi ile paylaşabildim. Aldığım geri dönütler hemen hemen vizedeki geliştirmelerimle aynıydı. Öncelikle tasarım olarak giriş ekranı ve kazanma ekranları bir nebze daha güzel olsa da oyunun oynandığı asıl ekranda daha güzel tasarım tercihleri kesinlikle yapılabilirdi.

Sadece bu proje kapsamında değil benim genel olarak kendimi tanımlarken de frontend kısmında iyi olmadığımı her zaman belirtiyorum. Çünkü hem bu alana dair görsel becerimin olmadığını düşünüyorum hem de açıkçası pek benim ilgimi çekmiyor. Yani ‘Butonların renk geçişi daha yumuşak olmalı, Paneller yazılar daha küçük olmalı’ dan ziyade o rastgele sayılara göre rastgele işlemlerin oluşmasındaki matematik daha çok ilgimi çekiyor. Bu yüzden tasarımla ilgili yapılan tüm yorumlara katılıyorum.

Bir diğer aldığım değerlendirme ise uygulamam hala her telefon için optimize değil. Yani her buton her yazı her ekranda aynı şekilde gözüküyor. Taşma problemleri de var. Bu da aslında bir nevi tasarım problemine giriyor. Çünkü web tarafında da başlı başına bir tasarım problemi olan responsive kavramı burada da geçerli.

Test sürecimde ise ios cihazım olduğu için apk çıktılarımı deneyemedim. Bu yüzden:



app-live.browserstack.com Sitesini kullanarak apk testlerimi yaptım. Kendi cihazım ios olduğu için ve apk olarak çıktı aldığım için testi online platformda yapabildim. Burada test yaparken kendi responsive hatalarım direkt gözüme çarptı. Fakat dediğim gibi bunları nasıl çözebileceğime dair de çok bir bilgim olmadığı için tamamlayamadım.

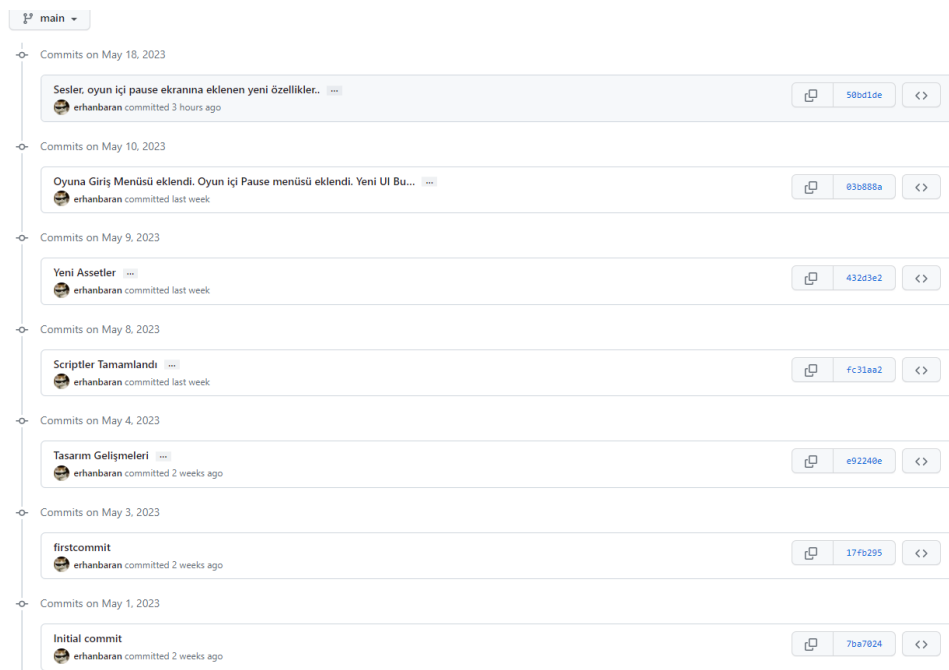
5. SONUÇ

İlk aşamadaki hedefim; Bir oyun platformu oluşturmaktı. Bu platform; Bir ana giriş ekranı, ardından birden fazla oyuna sahip olduğumuzu göstereceğim bir ara oyun seçme sahnesi ve bu ara oyun sahnesiyle birlikte geçtiğim oyunlarım hedefimdi.

Fakat planlama ve geliştirme sürecinde yaptığım hatalar;

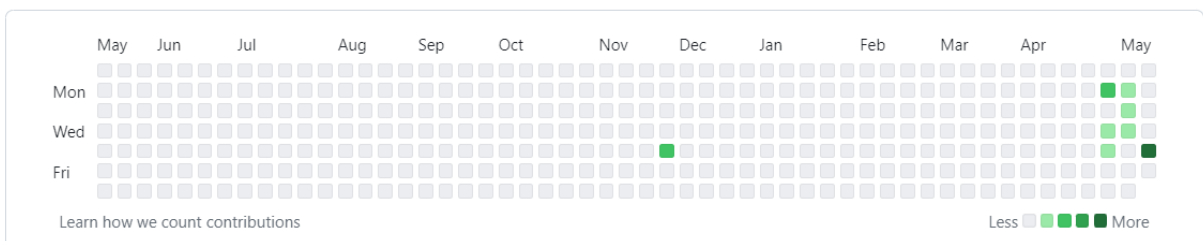
- Unity eğitim sürecine vermiş olduğum kısa sürede yüksek hedefler,
- Oyun tasarımına önem vermeye çalışırken kaybettiğim zaman,
- GitHub üzerinde projeyi sürekli dinamik olarak güncelleme ve commitleme çabam,
- Son olarak bu alandaki deneyimsizliğimden kaynaklı debug yeteneğimin yoksunluğundan dolayı sil baştan çok projeye girişmem

Bu süreçteki ilerlememi yavaşlattı ve hedeflerimin biraz gerisinde kalmama sebep oldu. Github üzerinde yaşadığım aksaklıkları bahane olarak sunmamak adına açtığım günlük niyetiyle commitlerimi göstermek için tuttuğum [repya](#) bakacak olursam:



16 contributions in the last year

Contribution settings ▼



[Burada](#) yedi commit gözüküyor fakat 1 Mayıs'tan itibaren hemen hemen her gün uğraştım. Özellikle aldığım hatalar sonucu çok vakit kaybetmiş olmam bazı commitler arası zaman farkını oluşturdu.

Gelinen noktada final tesliminde bu bahsettiğim genel eksikliklerimi ve isteklerimi de tamamladığımı ve projeyi güzel bir yere getirdiğimi düşünüyorum.

Son olarak amaç kısmında da değindim fakat tekrardan bu projenin özellikle böyle bir amaç doğrultusunda olması beni çok etkiledi. Bu süreci bizzat Gaziantep'te 6 Şubattan itibaren paylaşılan biri olarak bu uygulamanın her aşamasında yoğun duygularla geliştirmemi tamamladım. Bu uygulama benim adıma en azından geliştirme sürecinde kafamı dağıtmamı ve hala benden daha kötü durumda kişilerin olabileceğini aklıma getirmemi sağladı. Bu yüzden;

Proje geliştirme sürecinde yardımcı olan hocamız Arş. Gör. BEYZANUR DURMUŞ ve hem geliştirme sürecinde hem de bu derse getirdiği vizyondan dolayı Doç. Dr. FATİH ÖZKAYNAK'a bir teşekkürü borç bilirim.

Sonuç kısmında bahsettiğim;

Github Hesabım: <https://github.com/erhanbaran>

Projenin APK halini yüklediğim repom: <https://github.com/erhanbaran/YMGK-Final-195541027-Apk>