
GAZİ ÜNİVERSİTESİ
Gazi
1926

OTİSTİK ÇOCUKLAR İÇİN MOBİL OYUN GERÇEKLEŞTİRME

131180013 Melis BIYIKLI

131180029 Esra EROĞLU

BİLGİSAYAR PROJESİ II

RAPORU

BİLGİSAYAR MÜHENDİSLİĞİ

GAZİ ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

HAZİRAN 2017

OTİSTİK ÇOCUKLAR İÇİN MOBİL OYUN GERÇEKLEŞTİRME
(BİLGİSAYAR PROJESİ)

Melis BIYIKLI

Esra EROĞLU

GAZİ ÜNİVERSİTESİ

BİLGİSAYAR MÜHENDİSLİĞİ

Haziran 2017

ÖZET

Bu çalışmada otistik çocuklar için mobil oyun geliştirme işlemleri anlatılmaktadır. İlk olarak otistik çocukların özellikleriyle ilgili yapılan araştırmalara yer verilmiştir. Oyun için gerekli görsel ve işitsel materyaller Gazi Eğitim Fakültesi Özel Eğitim Bölümünden alınmıştır. Bu alanda yapılmış benzer çalışmalar mevcuttur. Tüm bu çalışmalar incelenmiş olup avantaj ve dezavantajlarıyla değerlendirilmiştir. Oyunun gelişimi bu sonuçlara göre tasarlanmıştır. Çünkü diğer çalışmalarda karmaşık açıklama ve tasarımların yapılması ve animasyon resimlerinin kullanılması dezavantaj oluşturabilmektedir.

Oyunun geliştirme aşamaları ayrıntılı bir şekilde açıklanmaktadır. Oyun içerisinde kullanılan materyaller ve tüm aşamaları görsel ögelerle birlikte anlatılmaktadır. Oyun, otistik bir çocukla test edilmiş ve bu değerlendirmeye sonuçlar raporda yer almaktadır.

Bilim Kodu : 708.3.026

Anahtar Kelimeler : Otizm, görsel ve işitsel materyal

Sayfa Adedi : 46

Danışman : Doç. Dr. Suat Özdemir

TEŞEKKÜR

Proje geliştirim süresince desteklerini esirgemeyen sevgili ailemize, Gazi Üniversitesi'ndeki değerli hocalarımıza, projenin uygulanması aşamasında yardımcı olan bilgilerini paylaşan özel ve kamu kurum/kuruluşlarında çalışan arkadaşlarımıza, projenin test aşamasında bizimle ortak çalışan ve yardım eden Bireysel Birikim ve Yusuf Karaman Özel Eğitim Uygulama Merkezlerine, proje geliştirim sürecinde büyük emeği bulunan, bize yol gösteren ve proje süreci boyunca bizden desteğini ile anlayışını esirgemeyen saygı değer hocamız Sayın Doç. Dr. Suat Özdemir'e şükranlarımızı sunarız.

İÇİNDEKİLER

Sayfa

ÖZET 1
TEŞEKKÜR 2
İÇİNDEKİLER 3
1. GİRİŞ 5
2. SİSTEM TASARIMI 10
Şekil 2.2. Veri akış diyagramı (VAD) 11
2.1. Bağımlılıklar ve kısıtlamalar 12
2.2. Riskler ve beklenmedik olaylar 12
2.3. Maliyet 13
3. VERİ TASARIMI 14
4. ARAYÜZ TASARIMI 15
4.1. Kullanıcı arayüzü 15
4.2. Donanım arayüzü 15
4.3. Yazılım arayüzü 15
4.4. Kullanıcı Özellikleri 15
4.5. Varsayımlar Ve Bağımlılıklar 16
4.6. Oyun Ekranındaki Arayüzler 16
5. ARAÇLAR VE KÜTÜPHANELER 20
5.1. YKE Dış Arayüz Gereksinimleri 20
5.2.YKE Dahili Arayüz Gereksinimleri 20
5.3. YKE Dahili Veri Gereksinimleri 20
5.4. Güvenlik ve Gizlilik Gereksinimleri 20
5.5. YKE Ortam Gereksinimleri 21
5.6. Bilgisayar Kaynak Gereksinimleri 21
5.7. Tasarım ve Uygulama Kısıtlamaları 21
5.8. Kullanıcıyla İlgili Gereksinimler 21
5.9. Lojistikle İlgili Gereksinimler 21

5.10. Gereksinimlerin Önceliği ve Kritikliği 21
6. PROJE TAKVİMİ 22
6.1. Yazılım Süreç Modeli 22
6.2. Roller ve Sorumluluklar 22
7. OYUN ÇALIŞMASI 23
8. KARŞILAŞILAN ZORLUKLAR 26
9. OYUN TEST EDİLMESİ 28
10. SONUÇ VE TARTIŞMA 32
REFERANSLAR 33
EKLER 34
EK 1. Giriş ekranı 35
EK 2. Puzzle 39
EK 3.Çıkış 40
EK 4. Öğrenme 44

1. GİRİŞ

Bu çalışmada otistik çocuklar için mobil oyun geliştirilmiştir. Oyunun amacı, otistik çocukları günlük hayata hazırlamaktır. Otizmli çocukların temel yetersizliği sosyal alandadır. Bu tür kişiler genellikle göz kontağı kurmamakta, insanlardan uzak durmakta ve alışık olmayan sesler çıkarmaktadır. İnsan duygularını anlamada zorluk çekmektedirler. Aynı zamanda kendi duygularını göstermekte yetersiz olmaktadırlar. Bu nitelikteki çocuklar diğer insanların duygu, düşünce ve yüz ifadelerini anlamakta sorun yaşadıkları için sosyal becerileri yetersiz olmaktadır. Bu durum sosyal körlük olarak da adlandırılabilmektedir. Otistik çocukların eğitiminde duygu anlamlandırma oldukça önemli rol oynamaktadır. Çünkü çocukların sosyal gelişimleri öncelikle duygu ifadelerini tanımak ve kullanabilmekle doğru orantılıdır [1].

Geliştirilen bu oyun, çocuklara eğlenceli bir öğrenme ortamı sunmaktadır. Oyunda çocuğun ilgisini çekebilecek görsel ögelere yer verilmektedir. Çünkü bu tür çocuklar görsel, işitsel uyarıcılara karşı duyarlı olmaktadırlar. Bu yüzden geliştirdiğimiz oyunda renkli arka plan ve videolu sesli geçişler kullanılmaktadır. Öğrenirken eğleneceği bir ortam sunulmaktadır. Oyunumuz otistik çocuklara duygu ifadelerini öğretmektedir.

Oyun Unity 2D'de geliştirilmektedir. Çünkü Unity platformu C# veya Java dillerini kullanmaktadır. Ayrıca sahip olduğu araçlarla geliştiriciye fayda ve kolaylık sağlamaktadır. Oyun içinde kullanılan görsel ve işitsel ögeler Gazi Eğitim Fakültesi Özel Eğitim Bölümünden alınmıştır. Unity'de kodlar Visual Studio'da C# dilinde yazılmaktadır. Android cihaz üzerinde çalıştırılmaktadır. Bunun için yardımcı uygulama Unity Remote 5 uygulamasıdır.

Oyun içinde kullanılan videolar için QuickTime programından yararlanılmıştır. Video sorunu çözünmüş ve videolar başarılı bir şekilde Android ortamına taşınabilmiştir.

Geliştirilen bu oyun, çocuklara eğlenceli bir öğrenme ortamı sunmaktadır. Oyunda çocuğun ilgisini çekebilecek görsel ögelere yer verilmektedir. Çünkü bu tür çocuklar görsel, işitsel uyarıcılara karşı duyarlı olmaktadırlar. Bu yüzden geliştirdiğimiz oyunda renkli arka plan ve öğrenme videosu, sesli geçiş ve talimatlar kullanılmaktadır. Öğrenirken eğleneceği bir ortam sunulmaktadır. Oyunumuz otistik çocuklara duygu ifadelerini öğretmektedir. Buna benzer oyunlar bulunmaktadır. Ancak çoğu farklı kavramlar öğretmeye yöneliktir. Duyguları öğreten oyunlar ise animasyon kullanılarak yapılmıştır. Otizmli bir çocuk animasyonlarla yapılan bir öğretmede gerçek insanların duygularını net anlayamamaktadır. Bizim oyunumuzda ise bu duygu öğretimi gerçek insan yüzleri kullanılarak yapılmaktadır. Bu sayede otizmli bir çocuk duyguları gerçek hayattaki olaylar ve insanlar üzerinde daha iyi öğrenebilmektedir. Ayrıca duygu ifadeleri ayrı ayrı kategorilere bölünmüştür. Ayrıca oyunumuzda eğitim dışında öğrenime yardımcı kategori de mevcuttur. Bu kategorilerde çocuğun duyguları öğrenmesi sağlanmaktadır. Oynanacak her duygu durumu aşaması öncesinde o aşamaya dair duygu ifadesi animasyon videosu ile tanıtılmaktadır.

Öğrenme aşamasını geçmiş olan çocukların öğrendiklerini pekiştirebilmesi için puzzle oyunları mevcuttur. Bu puzzle oyunları gerçek hayattaki insan yüzlerinden oluşmaktadır. Oyunumuzun adı yapılan işlemlere uygun olarak EMPOTİ adını almıştır.

Otistik çocuklar üzerinde genellikle eğitimciler tarafından hazırlanan, yüz yüze eğitime dayalı çalışmalar yapılmaktadır. Çocuk bu eğitimi sadece o ortamlarda sürdürmektedir. Örneğin 2014 yılında yapılan bir çalışmaya göre otizmli çocuklara duyguları kazandırma, replik silikleştirme yöntemi kullanılarak gerçekleştirilmiştir. Bu araştırma otizmli 3 çocuk ile 4 sene boyunca yürütülmüştür. 4 sene sonunda hedef çocukların eğitim bittikten sonra da kazandıkları becerileri sürdürdükleri gözlemlenmiştir. Çalışmada öncelikle 12 adet resimli kartla duygu durumları öğretilmiştir. Sesli efektlerle eğitim sürdürülmüştür. Araştırmada "baskonuş" cihazları kullanılarak bir takım repliklerin öğretilmesi sağlanmıştır. Çocuğun bu cihazdan çıkan sesleri taklit ederek günlük hayatta karşılaşacağı durumlara karşı göstermesi gereken davranış öğretilmektedir. Eğitim süresince, çocuğa eşleştirme, ayırt etme ve kendini sözel olarak ifade edebilme becerileri kazandırılmıştır. Replik bittikten sonra çocukların konuşma becerilerinde artış olduğu gözlemlenmiştir. Eğitimi biten öğrencilerin aldıkları kazanımlarla hayatlarını çok daha rahat sürdürdükleri gözlemlenmiş ve araştırma sonuçları %100'e yakın olduğu tespit edilmiştir [1].

Otizm alanında yapılan bir başka çalışma ise Otistik Çocuklar İçin Davranışsal Eğitim Programı (OÇİDEP) çalışmasıdır. Bu çalışmada alanında uzman kişiler yer almaktadır. Otizmli çocuklara uygulanan bu çalışma sonrasında birçok çocukta istenilen hedeflere ulaşıldığı ve pek çok değer öğretildiği saptanmıştır. OÇİDEP içeriğinde çocuğa kazandırılması istenen çeşitli seviyelerdeki bilgi ve beceriler bulunmaktadır. OÇİDEP projesi içerisinde eşleme, taklit, dil ve oyun becerileri yer almaktadır. Temel ve orta düzeyi bulunan bu eğitimde çocuğa pek çok beceri kazandırılmaktadır [2].

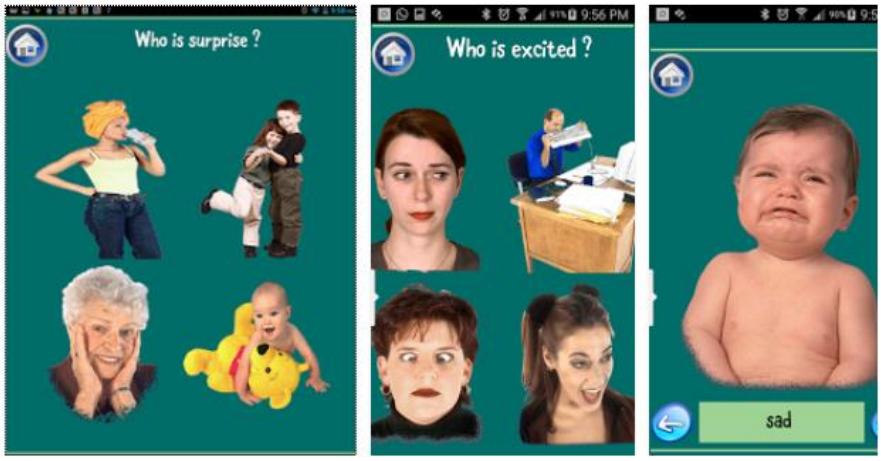
Otizmli çocuklar üzerinde yapılan bir diğer çalışma ise oyun ve oyuncakların otizmli çocuklar üzerindeki etkisini ölçmeye dayalıdır. Liccordello adlı bir uzman tarafından yapılan çalışmada, özellikle okul öncesi eğitimde telkin, övmek ve ödüllendirmenin otistik çocukların sosyal cevaplarını ve davetlerini arttırdığı görülmüştür. Uygun oyunların doğru yönlendirmeler ve uygulamalarla otistik bireyler otizmli çocukların sosyal aktivitelere davet edildiğinde olumlu sonuç vermesinin mümkün olacağını kanıtlamıştır. Manipülatif oyun, sosyo-dramatik oyun, sembolik oyun ve fonksiyonel ve tanıtıcı oyun olarak 4 farklı gruba bölümlenen oyun türlerinin her birinin çocuğa farklı şeyler kattığı gözlemlenmiştir. Bireysel ya da okullarda toplu olarak gerçekleştirilen oyunlarda çocuklarda sosyalleşme ve iyileşme sürecinin arttığı, oyunların çok faydalı olduğu gözlemlenmiştir [3, 4].

Bu eğitimler ülkemizde ve diğer yerlerde özel eğitim merkezlerinde verilmektedir. Maddi imkanı olan çocuklar bu eğitimden çoğunlukla ve kısıtlı sürede yararlanabilmektedir. Ancak sanal ortama taşınmış bir eğitim çocuğun bulunduğu her ortamda ve maddi imkanı olmayan çocuklar için de kullanılabilmektedir [1].

Projemiz için pek çok benzer oyun ve uygulama mevcuttur. Bizim projemizdeki gibi gerçek insan yüzlerinin kullanıldığı oyunlar da mevcuttur ancak çok karmaşık ve birbirine yakın ifadelerin yakın olduğu resimler kullanılmıştır. Bu resimler arasındaki ayrım sadece otizmli çocuklar için değil yetişkin bireyler için bile ayrımı zor kılmaktadır. Oyunda kullanılan açıklamalar ve oyun içeriği zor ve anlaşılmaz olmaktadır. Bu durum oyunu kullanışsız hale getirmektedir.

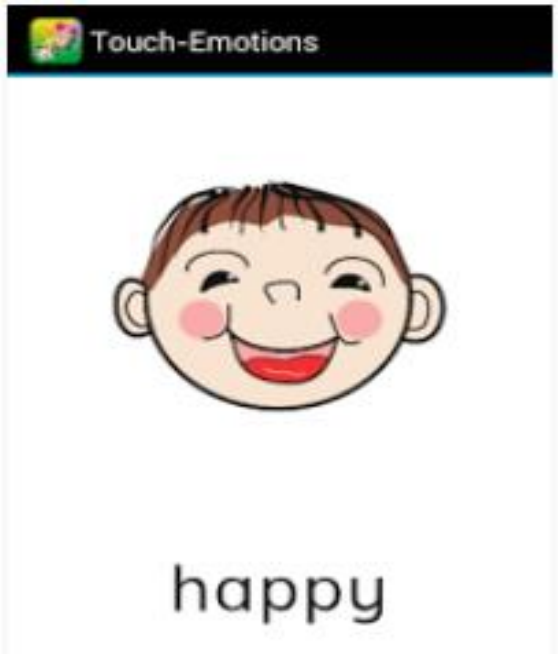
Ayrıca bu üst seviye oyunlar farklı dillerde yapılmaktadır. Türkçe olarak bu türde bir çalışma bulunmamaktadır. Oyunumuz otizmli Türk çocuklar için öğrenim sürecini kolaylaştırmaktadır.

Google Play ve AppStore gibi kullanıcılara hizmet veren sanal marketlerde duyguları öğretmeye yönelik pek çok oyun mevcuttur. Google Play'de yer alan duyguları öğretmeye yönelik örnek bir uygulama Şekil 2.1'deki gibidir. Oyunda var olan resimlerde hem gerçek insan resimleri hem de animasyonlar kullanılarak duygu durumları öğretilmektedir [5].



Şekil 2.1. Örnek oyun çıktısı [5]

Bir başka oyun örneği de Şekil 2.2'deki gibidir. Bu oyunda da duyguların öğretimi animasyon tipleri ile sağlanmaktadır.



Şekil 2.2. Başka bir örnek oyun çıktısı [5]

Bir diğer uygulama puzzle oyununa yöneliktir. Otistik çocukların oyunu oynamasına yardımcı olabilmek için puzzle parçalarının yerleşimi kolay yapılmıştır. Renkli ve görsel ögelerden yararlanılmıştır.



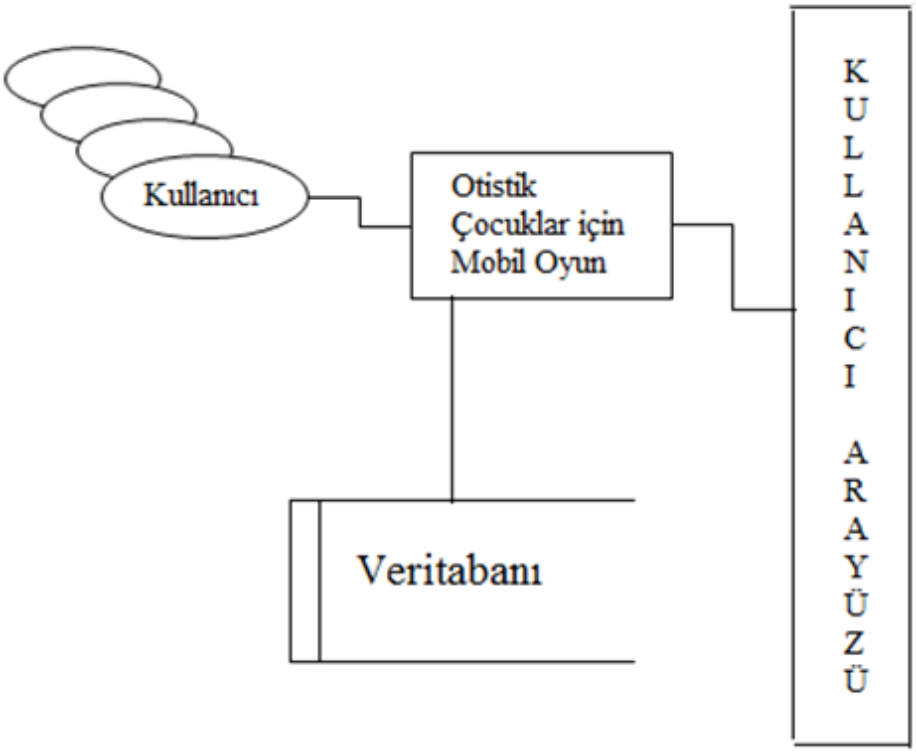
Şekil 2.3. Örnek puzzle oyunu [5]

Oyunumuzda Şekil 2.1 ve Şekil 2.3'deki gibi gerçek insan yüzlerinin kullanıldığı eğitim bölümü ve puzzle aşaması yer almaktadır. Böylece çocuğun duygu durumlarının gerçek insan yüzleri üzerinden gözlemleme ve öğrenme şansı yakalamakta ve puzzle oyunlarıyla pratiğe dökmektedir.

Otistik çocuklar için mobil oyun uygulaması oluşturulurken kullanıcı tipleri ve işlevleri üzerinden tasarım gerçekleştirilmiştir. Kullanıcı tipleri ve işlevleri şunlardan oluşur:

- **Kavramlar ve Tanımlar:**
  - **Kullanıcı**:
    - ✓ Oyun içindeki talimatlara göre oyunu oynar.
    - ✓ Öğrenme aşamasını ve sonrasında puzzle'yi tamamlar.
    - ✓ Eğitmenler tarafından oyun oynatılır.
  - **Eğitmen:**
    - ✓ Oyunda çocuğa yardımcı olur.
    - ✓ Öğrenme veya puzzle aşamalarından çocuğa uygun olanı seçer.
  - **Otizm**:
    - ✓ Bu hastalık beyin gelişimsel bir bozukluktur.
    - ✓ Duyguları anlayamama, iletişim kuramama temel belirtileridir.
  - **EMPOTİ (Oyun adı):**
    - ✓ Oyunumuzun adı EMPOTİ olarak belirlenmiştir. EMOJİ + PUZZLE + OTİZM kelimelerinin kombinasyonundan oluşmaktadır.

2. SİSTEM TASARIMI



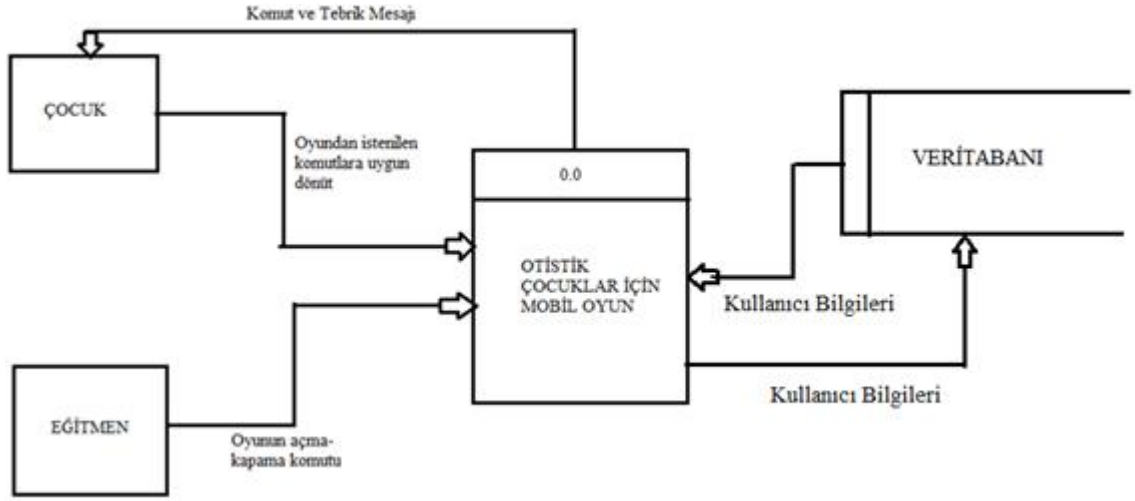
Şekil 2.1. Uygulamanın Mimari Tasarımı

Şekil 2.1'de gösterildiği gibi uygulamanın mimari tasarımında kullanıcı giriş ekranında gerekli bilgileri eğitmen tarafından verilerek iki seçenekli oyun ekranına giriş yapar. Giriş yapmak isteyen kullanıcı ekrandaki "giriş" butonuna basması gerekmektedir. Bu işlemden sonra kullanıcının veri tabanına bağlanmasını ve bilgilerinin alınmasını sağlayacak bir pop-up ekranı çıkmaktadır. Pop-up ekranında kullanıcı bilgisinin alınabilmesi için bir adet InputField ve bilgilerin kaydedilmesini sağlayan bir adet buton yer almaktadır. Eğer daha önceden kayıtlı bir kullanıcıysa kullanıcı bilgisi korunarak oyun süresince elde edilen yeni skor ve level bilgileri veri tabanında güncellenmektedir. Kullanıcı ve skor bilgisi ekranda gösterilmektedir.

Karşısına gelen 2. ekranda puzzle ya da eğitim seçeneklerinden birini seçerek oyunu başlatır. Oyun başlamadan önce giriş ekranından alınan ismin veri tabanında önceden kayıtlı olup olmadığına bakar. Önceden kayıtlı bir isim ise kullanıcıya seçtiği seçenekle ilgili olan önceki kayıtlı level bilgisini veri tabanından okur. Daha sonra kullanıcıya eski levelından devam edip etmek istemeyeceğini soran bir pop-up ekranı daha çıkar. Bu yeni pop-up ekranında iki adet buton yer alır. Bu butonlardan "evet" isimli buton kayıtlı leveldan devam edeceğini ifade ederken " hayır" butonu oyunu baştan oynamak istediğini belirtmektedir.

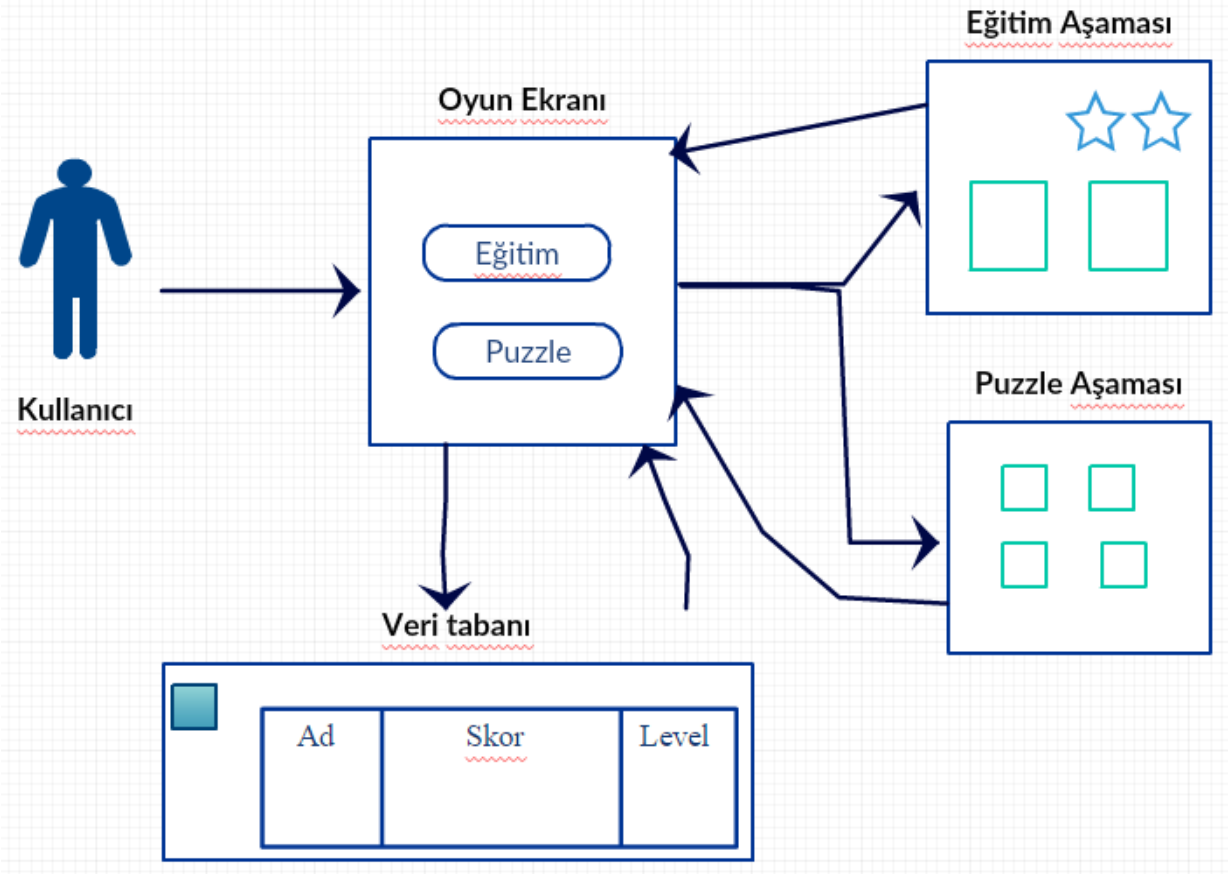
Yapılan bu seçime göre istenilen oyun seviyesi oyuna yüklenmektedir. Daha sonra oyundaki işlemleri, talimatları başarıyla tamamladıktan sonra başarı durumu oyun tarafından hesaplanmaktadır. Oyun içerisinde verdiği her doğru cevaba karşılık 50 puan eklenmekte yanlış cevaba karşılık 10 puan çıkarılmaktadır. Kullanıcı her sayfada yer alan geri dönüş butonu ile seçenekler arası değişim yapabilmektedir. Puzzle ya da eğitim seçeneğinin her birinde kendine özgü skor puanı hesaplanmaktadır.

Kullanıcının çıkış yapmak istediğinde iki seçenekli giriş ekranında bulunan çıkış butonuna basması yeterlidir. Çıkış butonuna bastığı andan itibaren ikinci pop-up ekranı kullanıcıya sunulmaktadır. Bu ekranda kullanıcı bilgisi ve oyunda elde ettiği skorun gösterilmesini sağlayan InputField ve pop-up ekranının kapatılmasını sağlayan buton yer almaktadır. Bu sayede kullanıcıya gösterilmektedir (EK 3). Daha sonra bu bilgiler her iki seçeneğe ait level bilgisi ile beraber veri tabanına kayıt olmaktadır (Şekil 2.2).



Şekil 2.2. Veri akış diyagramı (VAD)

Şekil 2.3'de projenin genel kullanımı ve veri akış diyagramı gösterilmektedir. Kullanıcı oyun ekranına girdiğinde ilgili seçeneği seçerek eğitim ya da puzzle alanına ve veri tabanına bağlanmaktadır.



Şekil 2.3. Kullanım durum diyagramı (Use-Case)

2.1. Bağımlılıklar ve kısıtlamalar

Oyunun mantıksal bağımlılığı eğitim aşaması tamamlandıktan sonra puzzle oyununun yapılabilmesidir. Yazılımsal olarak bu şekilde bir kısıt yoktur. Ancak uzman görüşlerine göre duygu durumunu bilen çocukların daha ileri seviyeleri oynaması gerekmektedir.

Oyunun diğer kısıtları ise duygu durumlarının (mutlu, mutsuz, kızgın ve şaşkın) olacak şekilde belli bir sıraya sahip olması ve bu sırayla oynanmasıdır. Ayrıca oyun içerisinde kullanılan video, ses ve resim dosyalarının Unityle uyumlu olması gerekmektedir.

2.2. Riskler ve beklenmedik olaylar

Oyunun arayüz tasarımında puzzle aşamasında kullanılan resimlerin parçalarını yerleştirilmesi aşamasında yanlış yere yanlış parçanın yerleştirimi söz konusu olduğunda uygulamada beklenmedik durumlar oluşabilmektedir. Oyunda seviyeler arası geçişte kullanılan video ve ses dosyalarının Unity ve android cihaz için uyumsuzluk oluşturabilmesi mümkündür. Kodlama aşamasında kullanılan değişkenler birbirini kısıtlamayacak şekilde ve bazı durumlarda birbirine erişebilecek şekilde yapılmadığı takdirde oyun akışında problemler oluşmaktadır.

2.3. Maliyet

Projenin toplam maliyeti gereksinimlerin ayrı ayrı toplamına eşittir. Buna göre Visual Studio 2012 $499 olarak belirlenmiştir. Oyunun testi için kullanılan android cihazların toplam maliyeti 2000TL'dir.

Unity platformunun ve QuickTime'ın herkesin kullanımına açık olması sebebiyle özel fiyatlandırması bulunmamaktadır. Kullanılan görsel ve işitsel ögeler Gazi Eğitim Fakültesi Özel Eğitim Bölümünden alındığı için bu materyallerin maliyeti bölüme aittir.

3. VERİ TASARIMI

Şekil 3.1'de internet veri tabanının veri şeması gösterilmektedir. Veri tabanına kayıtlı olan bilgilerde kullanıcı bilgisi ilk girişte eğitmen tarafından sisteme verilmektedir. Skor bilgisi ise kullanıcının oyunu oynadıktan sonra elde ettiği bilgidir. Level bilgisi ise kullanıcının çıkış yapmadan önce oyunu bitirdiği seviyedir. Puzzle ve öğrenme için ayrı seviyeler şeklinde veri tabanında tutulmaktadır. Kullanıcı oyun esnasında sadece iki seçenekten birini bile oynasa oynamadığı diğer seçeneğe ait skor bilgisi veri tabanında aynen korunmaktadır. Tüm bu bilgiler toplanarak en sonunda veri tabanına kayıt edilmektedir. Veri tabanındaki bilgilerin tipleri geldikleri kaynaklara göre otomatik ayarlanmaktadır. Kullanıcı bilgisi text alanından alındığı için string tipindedir. Skor bilgisi oyun içerisinde hesaplandığından integer tipindedir. Level bilgisi ise oynanan level bilgisinden alındığı için integer tipindedir. Oyunda kullanılan veri tabanında yer alan bilgiler çok olmadığı için gelişmiş bir veri tabanına ihtiyaç yoktur. Tek tablo ile gerekli bilgilerin depolanması sağlanmaktadır. Bu tablonun şematik gösteriminde ER ya da EER diyagramlarının kullanılmasına ihtiyaç duyulmamaktadır.

Şekil 3.1'deki gibi veri tabanı görünümüne sahip olan Firebase veri tabanına internet bağlantısı ile kullanımı mümkündür. Bu platformda projelere ait bir veri tabanı oluşturmak için Google hesabı ile kayıt olunması ve daha sonra giriş yapılması gerekmektedir. Veri tabanındaki kayıtlar sistem tarafından atanan otomatik bir id ile tutulmaktadır. Id ile birlikte kullanıcı adı, skor, eğitim ve puzzle seviyesi tutulmaktadır.



3.1. Firebase veri tabanına örnek bir kayıt eklenmesi

4. ARAYÜZ TASARIMI

4.1. Kullanıcı arayüzü

ARAYÜZLER	TANIM
Öğrenme aşaması	Eğitmen tarafından duygu ifadelerinin öğrenmesine geçiş yapılır.
Puzzle aşaması	Eğitmen tarafından duygu ifadelerine ait puzzle aşamasına geçiş yapılır.

Tablo 4.1. Kullanıcı Arayüzü

4.2. Donanım arayüzü

Yapılan bu proje için donanım arayüzü olarak android bir cihaz, kullanıcı ile program arasında bağlantı sağlamaktadır.

4.3. Yazılım arayüzü

YAZILIM ARAYÜZÜ	ADI	VERSİYON	KAYNAK
İşletim sistemi	Windows	Windows 10	Microsoft
Uygulama ekranı	Note 3 Neo/ e-tab4	Android 4.4.2/4.2.2	Android
Donanım Platformu	Unity	Unity 5.4.1	Microsoft
Yazılım Platformu	Visual Studio	Visual Studio 2015	Microsoft
Programlama dili	C#	.NET Framework 4.5	Microsoft
Android SDK Platformu	Android SDK Manager	Android SDK Manager 24.1	Android
Video Yükleme Platformu	Quick Time	Quick Time 7.7.9	Apple
Gerçek zamanlı veri tabanı	Firebase	Firebase SDK 3.0.0	Google

Tablo 4.2. Yazılım Arayüzü

4.4. Kullanıcı Özellikleri

Kullanıcıların;

- Android işletim sistemine sahip bir cihazlarının olması beklenmektedir.

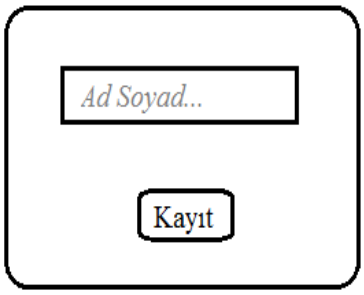
- Çocuklar için onları yönlendirebilecek bir gözetmene ihtiyaç duyulması beklenmektedir.
- Kullanıcıların uygulamayı kullanabilecek seviyede olması beklemektedir.

4.5. Varsayımlar Ve Bağımlılıklar

- Uygulamanın kullanılacağı cihazlarda Android işletim sistemine sahip olduğu varsayılmaktadır.
- Çocukların oyunu oynayabilmesi ve yönlendirilebilmesi için gözetmen olduğu varsayılmaktadır.

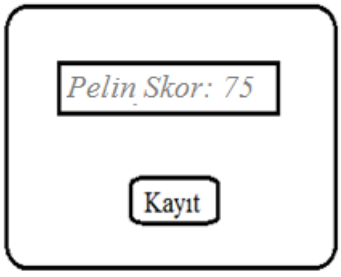
4.6. Oyun Ekranındaki Arayüzler

Şekil 4.1'de gösterildiği gibi kayıt ekranında kullanıcı bilgileri alınmakta ve veri tabanına kayıt edilmektedir. Bu işlem için öncelikle text alanından isim-soyisim alınmakta ve bu bilgi oyun içinde muhafaza edilmektedir. Oyun bitimi sonrasında elde edilen skor ve level bilgisi kullanıcıya sunulmakta ve kayıt işlemi gerçekleştirilmektedir (EK 1).

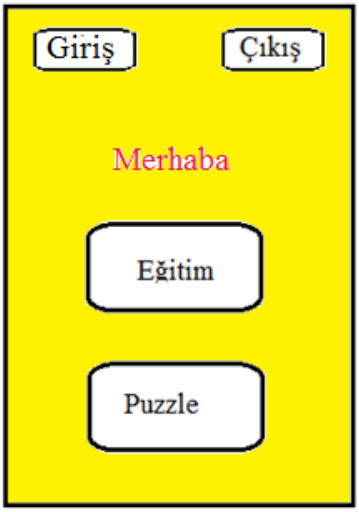


Şekil 4.1. Kayıt ekranı

Şekil 4.3'de olduğu gibi giriş ekranından sonra sunulan iki seçenekli ekran eğitim ve puzzle olmak üzere iki adet oyun butonundan oluşmaktadır. Kullanıcı seçimine göre diğer ekranlara yönlendirme yapılmaktadır. Çıkış butonu ile Şekil 4.2'deki ekrana bağlantı sağlanmakta ve kullanıcı bilgileri ile skor bilgisi pop-up ekranında gösterilmektedir. Aynı zamanda bu bilgilerle birlikte her iki oyun türüne ait level bilgileri de veri tabanına kayıt edilmektedir (EK 1).

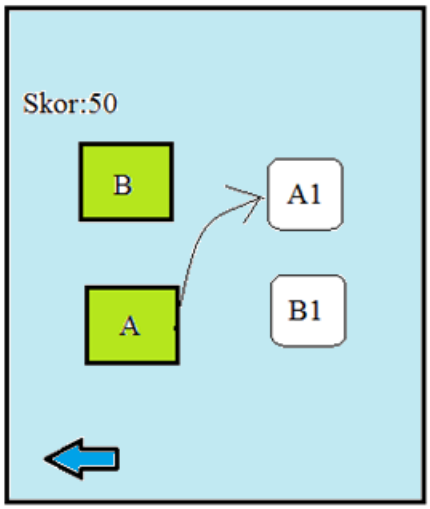


Şekil 4.2. Skor gösterim ekranı



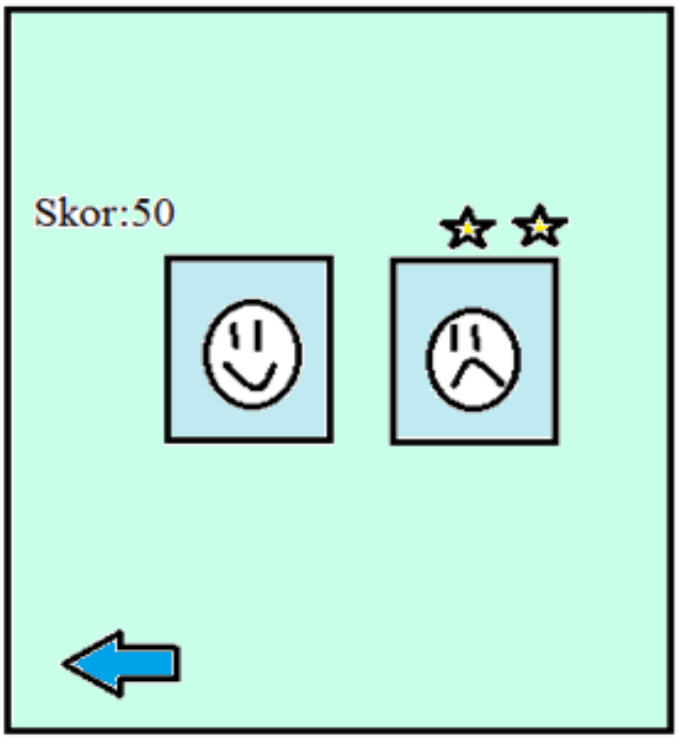
Şekil 4.3. Bölüm seçim ekranı

Oyunun puzzle seçeneği seçilirse Şekil 4.4'deki gibi bir ekran kullanıcıya sunulmaktadır. Bu ekranda kullanıcının puzzle parçalarını doğru yerlere yerleştirmesi beklenmektedir. Örneğin Şekil 4.4'deki A parçasını A1 parçası üzerine, B parçasını ise B1 parçası üzerine yerleştirmesi gerekmektedir. Yanlış yerleştirim engellenmekte ve kullanıcının doğru yapması için yönlendirme yapılmaktadır (EK 2).



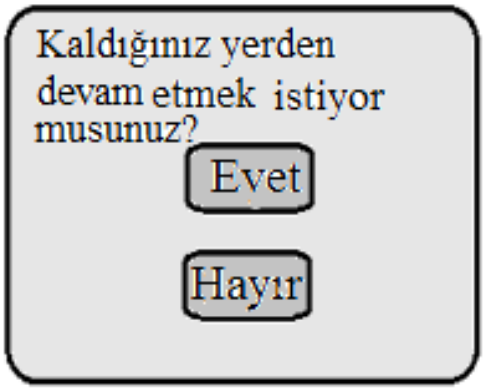
Şekil 4.4. Puzzle ekranı

Şekil 4.5'de kullanıcının eğitim ekranını seçmesi durumunda iki ya da üç seçenekli seçim ekranı gösterilmektedir. Bu ekranda mutlu, mutsuz, kızgın ve korkmuş duygularına ait resimler sunulmaktadır. Doğru seçimin yapıldığında ekrana yıldız figürü konulmakta ve skor bilgisi ekranda gösterilmektedir. Üç yıldız tamamlandığında bir sonraki duygu durumuna geçiş yapılmaktadır (EK 4).



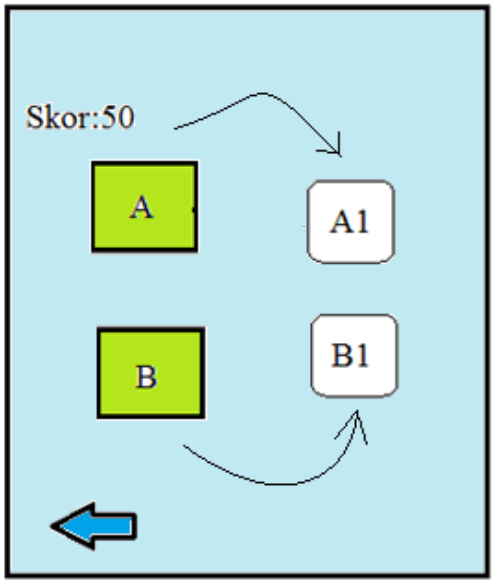
Şekil 4.5. Eğitim ekranı

Oyunda hangi seçenek seçilirse seçilsin kullanıcıya Şekil 4.6'daki gibi bir pop-up ekranı sunulmaktadır. Eğer kullanıcı yeni bir kullanıcı ise bu ekran açılmamaktadır. Bu durum için veri tabanında kullanıcı adı ile kontrol yapılmaktadır. Kayıtlı bir kullanıcı ise ilgili oyunda en son oynadığı seviye bilgisinin yüklenip yüklenmeyeceğine dair bu pop-up ekranı açılmaktadır. Hayır butonu oyunun baştan başlayacağını belirtmektedir. Kullanıcı evet butonuna bastığı zaman ise oyunda kaldığı yerden devam etmesi sağlanmaktadır.



Şekil 4.6. Level seçim ekranı

Oyunun puzzle seçeneği seçildikten sonra Şekil 4.4'deki gibi bir ekran yüklenmeden önce puzzlenin nasıl oynanacağına dair Şekil 4.7'deki gibi bir simülasyon ekranı gösterilmektedir. Bu ekranda her parçanın uygun olduğu yer alanına nasıl yerleşmesi gerektiği otomatik olarak gösterilmektedir. Bu ekrandan sonra puzzle oyunu açılmaktadır.



Şekil 4.7. Eğitim ekranı

5. ARAÇLAR VE KÜTÜPHANELER

5.1. YKE Dış Arayüz Gereksinimleri

Bu uygulamada dış arayüz olarak kullanıcı ile sistem arasındaki bağlantıyı sağlayan Android cihaz mevcuttur.

Harici arayüz olarak Android cihaz ve kullanıcılar örnek verilebilir. Geliştirilen yazılım Android cihaza Android SDK platformu ve Unity platformu aracılığıyla yüklenmektedir. Oyunun kullanımı esnasında eğitmen ve çocuk aktif rol oynamaktadır. Test için kullanılan cihazlar Galaxy Note 3 Neo ve e-tab4 cihazlarıdır. Bu cihazlarda test işleminin başarılı bir şekilde yapılabilmesi için Unity Remote 5 uygulamasının cihazlarda yüklü olması gerekmektedir.

5.2.YKE Dahili Arayüz Gereksinimleri

Bu uygulamada dahili arayüz olarak yazılım ekranı ile kullanıcı arasındaki bağlantıyı sağlayan ekranlar mevcuttur. Dahili arayüz olarak Unity içerisinde yer alan game object'ler ve oyun ekranı kullanılmaktadır. Oyun ekranları ve kullanılan objeler oyun işleyişine ve akışına uygun olarak değişmektedir. Oyun ekranına giriş yapıldığında seçilecek aktiviteye uygun ekranlar görsel ve işitsel ögeler yüklenmektedir. Duygu ifadelerinin öğretimi aşamasında oyun ara yüzü, çocuktan aldığı yanıtlara göre sesli geri bildirim yapmaktadır.

Uygulamada oluşabilecek herhangi aksaklıktan ara yüz de etkilenmektedir. Ara yüzle bağlantı uygulama içinde yer alan kodlar tarafından gerçekleştirilecektir. Alınan yanıtlar belirli bir periyodu takip etmemektedir. Oyun ekranının çocuğa sunulmasında, oluşabilecek ekstra durumlarda (oyundan çıkış) ve oyunun kapatılması aşamalarından eğitmenler sorumludur. Uygulama android işletim sistemine sahip cihazlarda çalışmaktadır.

5.3. YKE Dahili Veri Gereksinimleri

Uygulamada internet üzerinden erişilebilen veri tabanı kullanılmaktadır. Bu veri tabanının kullanılma sebebi oyunun kullanıcılara özgü olabilmesini sağlamak ve kullanıcıların etkileşimli oyun oynaması sağlanmaktadır. Masaüstü veri tabanı kullanılmamasının sebebi android cihaz için uyumlu olmasını sağlamaktır. Bu veri tabanı için gerekli bilgiler kullanıcıdan ve oyundan alınmaktadır. Bu sebeple oyun sırasında internet bağlantının sağlanması gerekmektedir.

5.4. Güvenlik ve Gizlilik Gereksinimleri

Kullanıcı girişi yapılıp isim-soyisim alınarak bilgilerin kullanıcılara özgü olması sağlanmakta ve küçük çaplı gizlilik sağlanmaktadır. Bunun haricinde eğitmenin ve çocukların kullanabilmesi zor ve gereksiz olabileceği için karmaşık bir şifre girişi yapılmamaktadır.

5.5. YKE Ortam Gereksinimleri

Uygulamanın çalışabilmesi için bir android cihazın sahip olması gereken tüm yapılar, Android işletim sistemi, yeterli hafıza alanı ve gerekli bağlantılar sağlanmalıdır. Uygulamada internet üzerinden erişilebilen veri tabanı kullanıldığı için internet bağlantısının olması gerekmektedir.

5.6. Bilgisayar Kaynak Gereksinimleri

Uygulamanın tam çalışabilmesi için bir android cihazın ihtiyaç duyabileceği tüm kaynaklar (içyapısındaki donanım v.b.) mevcut olmalıdır. Android cihazın büyüklüğü ve tipi çok önemli değilken bellek, işlemci gibi içyapısındaki birimlerin özellikleri daha önemli olmaktadır. Cihaz içerisinde en az 70 MB lık bir alanın olması yeterlidir.

5.7. Tasarım ve Uygulama Kısıtlamaları

Uygulamanın çalışması esnasında kullandığı birimler pek çok kısıta sahiptir. Bir duygu ifadesinin 3 etabı da tamamlanmadan diğer duygu ifadesine geçilmemektedir. Aynı zamanda bölümler arasında yıldız animasyon geçişindeki talimatlara uyulmalıdır. C# dışındaki bir programlama dili ve android cihaz dışındaki bir işletim sisteminin kullanımı, uygulama işleyişinde aksaklıklara ve ya eksikliklerine yol açabilmektedir. Ayrıca kullanıcı bilgilerinin güvenli bir şekilde korunmasını sağlamak amacıyla oyuna giriş ve çıkış yapılması zorunludur. Kullanıcılar mutlaka giriş butonunu kullanarak oyuna başlamalı ve oyunu bitirdikten sonra çıkış butonu ile veri tabanına kayıt olması gerekmektedir.

5.8. Kullanıcıyla İlgili Gereksinimler

Uygulamada çocuk aktif rol oynamaktadır. Eğitmen ise oyun sırasında çocuğa yardımcı olmaktadır. Uygulamada eğitmen oyun hakkında yeterli bilgi ve beceriye sahip olmaktadır.

5.9. Lojistikle İlgili Gereksinimler

Uygulamanın taşınmasıyla ilgili herhangi bir gereksinim mevcut değilken bakımı için gereksinimler mevcut olabilmektedir. Uygulamanın genel akışının dışına çıkılması veya telefonun yazılımının bozulması gibi durumlar oluşacak olursa bu durumlar yeni ihtiyaçları doğuracaktır. Ayrıca kayıt işlemi için internet bağlantısına ihtiyaç duyulmaktadır. İnternet bağlantısı olmadan kayıt yapılması sağlanmamaktadır.

5.10. Gereksinimlerin Önceliği ve Kritikliği

Yazılımın yapılacağı sistem gereksinimleri, kodlama için gerekli gereksinimler, uygulamanın çalışması için gerekli ortam bağlantıları en önemli öncelikler ve gereksinimler arasında yer almaktadır. Sonraki gereksinimler ise uygulamanın çalışmasına yöneliktir. Oyunun başarılı bir şekilde çalışması ve programa iletmesi, uygulamanın devamındaki gereksinimlerden daha önceliklidir.

5.11.Araçlar ve Teknikler

Oyun Unity 2D'de geliştirilmektedir. Çünkü Unity platformu C# veya Java dillerini kullanmaktadır. Ayrıca sahip olduğu araçlarla geliştiriciye fayda ve kolaylık sağlamaktadır. Oyun içinde kullanılan görsel ve işitsel ögeler Gazi Eğitim Fakültesi Özel Eğitim Bölümünden alınmıştır. Unity'de kodlar Visual Studio'da C# dilinde yazılmaktadır. Android cihaz üzerinde çalıştırılmaktadır. Bunun için yardımcı uygulama Unity Remote 5 uygulamasıdır. Oyun içinde kullanılan videolar için QuickTime programından yararlanılmıştır. Kullanıcı bilgilerinin kayıt altına alınmasını sağlamak için gerçek zamanlı (real time) veri tabanı olan Firebase kullanılmıştır.

6. PROJE TAKVİMİ

- Şubat 2017 -> Oyuna ait seçenek ekranlarının hazırlanması.
- Mart 2017 -> Puanlama, yıldız, animasyon, video eklenmesi.
- Nisan 2017 -> Veri tabanı bağlantısının kurulması.
- Mayıs 2017-> Uygulamanın bağlantılarının düzenlenmesi, test ve raporlama.

6.1. Yazılım Süreç Modeli

Yazılım süreç modeli olarak Artımlı Süreç Modeli ve UP Modeli de kullanılmaktadır. Bu modelde takım içerisinde keskin bir görev dağılımı yoktur. Herkes raporlama, tasarım, kodlama ve test aşamalarına dâhil olmaktadır. Artımlı süreç modeli UP modelle beraber uyumlu ve yakın olduğu için projede UP modelinin bir takım özelliklerine yer verilecektir. UP modelindeki parçalara ayrılmış iş sonucunda sunulan prototip artımlı modelin böl-yönet tekniği ile uyum sağlamaktadır. Böylelikle her iş parçacığı sonunda oluşacak olan prototip giderek esas ürüne yaklaşmakta ve doğruluğu test edilmektedir. Bu tutum proje ekibi arasındaki bağı güçlendirip ekibe motivasyon sağlamaktadır.

6.2. Roller ve Sorumluluklar

Süreç modeli olarak artımlı süreç programlama seçildiği için projeyi belirleme, projenin veritabanı sisteminin tasarımı, proje esnasında kullanılacak olan teknolojilerin seçimi, proje dokümantasyonunun hazırlanışı, projenin sınıf ve fonksiyonlarının kodlanması ve projenin test aşaması gibi her türlü süreçte proje grubunda yer alan öğrencilerin tümü görev almaktadır. Projenin analiz aşamasından test aşamasına kadar tüm ekip üyeleri aktif rol oynamaktadır. Bu dönüşümlü çalışma modeli her grup üyesinin projenin her aşamasına hakim olmasına katkı sağlamaktadır. Olası bir aksaklığa müdahale edebilecek üyelerin mevcut olması projenin daha doğru ve hızlı ilerlemesine yol açmaktadır.

7. OYUN ÇALIŞMASI

Otistik çocukların eğitiminde 4 temel başlık ön plandadır. Bunlar çocuğun problemli davranışlarıyla başa çıkma, iletişim becerisi kazandırma, diğer becerileri geliştirme ve bağımsızlık kazandırmadan oluşmaktadır. Bizim projemiz duyguları öğrenme yönündeki becerisini geliştirme ve buna bağlı olarak iletişim yeteneği kazandırmaktır.

Uygulamanın tasarım aşamasında kullanıcının kolay kullanabilmesi ve anlaşılır olması için kolay bir ara yüz tasarımı yapılmıştır. Tasarlanan oyunun giriş ekranında öğrenme ve puzzledan oluşan iki seçenek bulunmaktadır.

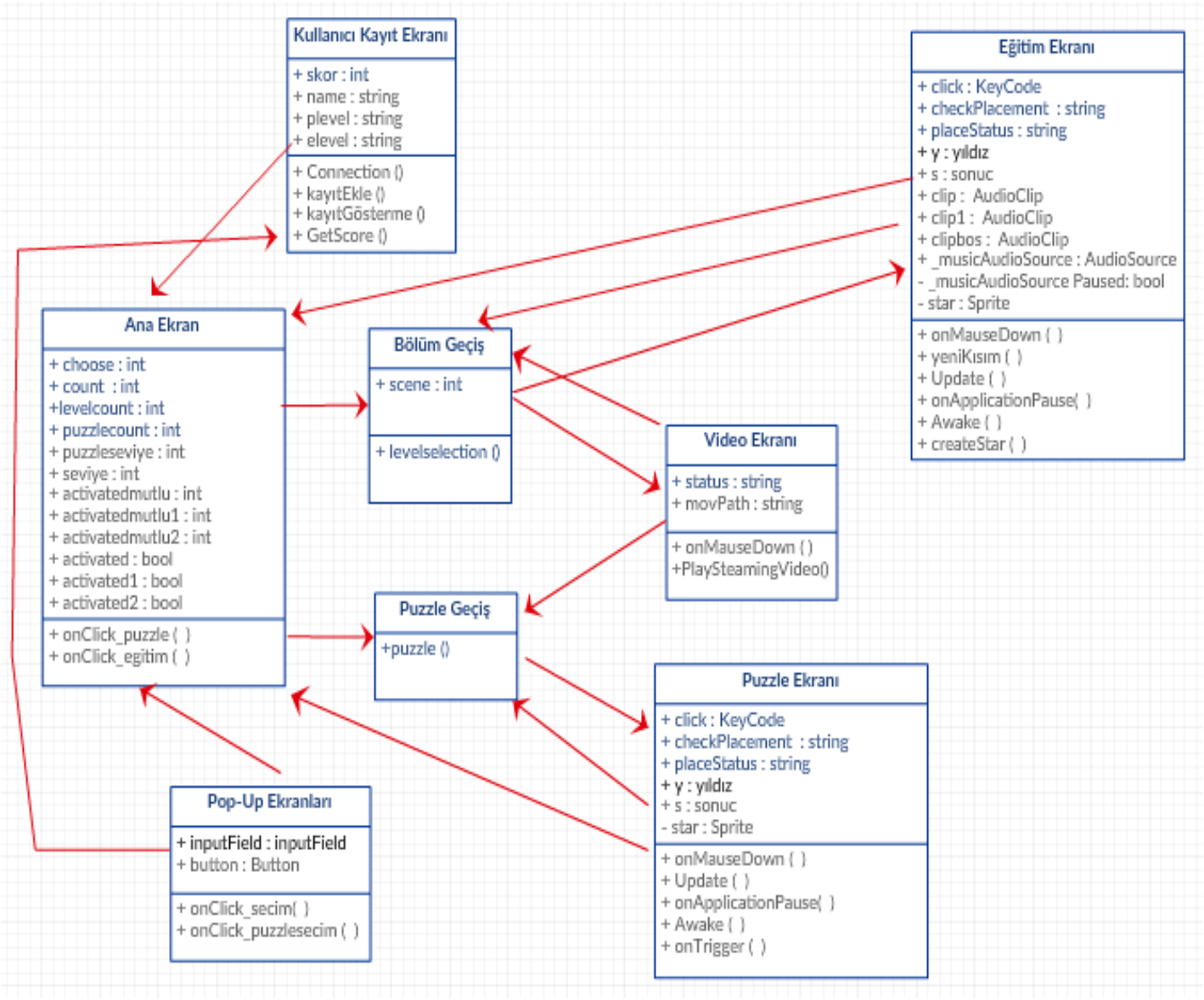
Öncelikle çocuğun öğrenme aşamasını başarıyla tamamlaması istenmektedir. Daha sonra puzzle kısmına geçiş sağlanmaktadır. Öğrenme aşamasında 4 farklı duygu ifadesi için karşılaştırmalı olarak öğrenme ekranları sunulmaktadır. Öncelikle "mutlu" kavramı öğretimi için 3 aşamadan oluşan gerçek insan fotoğrafları kullanılmıştır.

Birinci bölümün ilk aşamasında "mutlu-üzgün", ikinci aşamasında "mutlu-kızgın" ve son aşamasında "mutlu-korkmuş" olacak şekilde iki seçenek sunularak mutlu duygusunun öğretilmesi sağlanmaktadır.

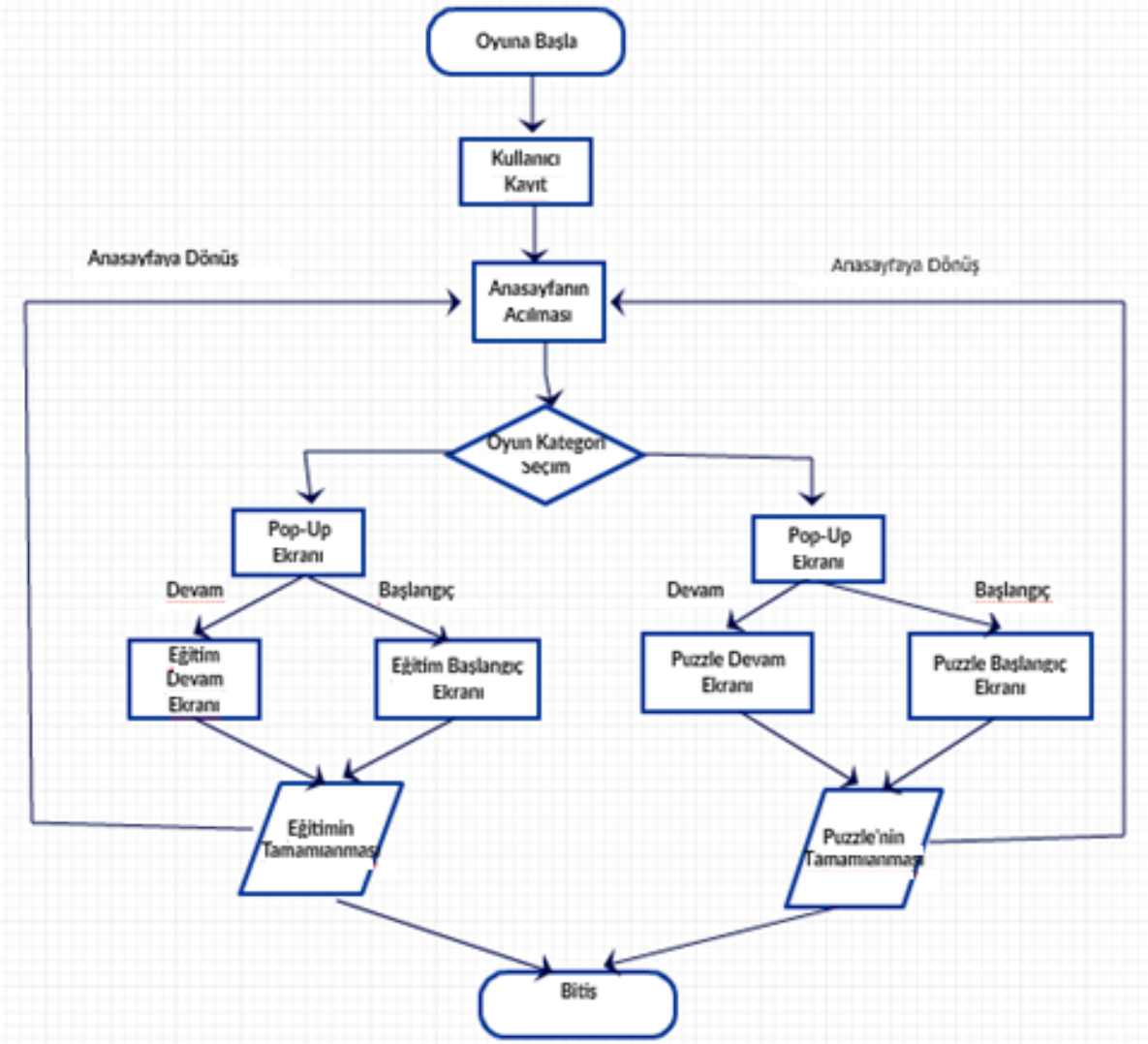
Bu aşamalar diğer üç kavram (mutsuz, kızgın ve korkmuş) için de geçerlidir. Oyunun ilerleyen aşamalarında bu kavramların 3 seçenekle birlikte öğretilmesi amaçlanmaktadır. Her bölümün her aşamasında verilen her doğru cevaba karşılık ekrana "yıldız" figürü konulmaktadır. Her bölümdeki 3'lü aşama tamamlandıktan sonra yeni bölüme geçiş sağlanmaktadır. Oyunun puzzle aşamasında yine öğretilen 4 farklı duygu durumu için puzzle yapılmaktadır. Bu puzzle aşamalı olarak kullanıcıya sunulmaktadır.

Puzzle bölümü 4 farklı duygu durumu için sırasıyla geçiş yapmakta ve kolaylık sağlaması için 2 parçadan oluşmaktadır. Çocuğu bu iki parçalı puzzle parçalarını uygun bir şekilde yerleştirmesi beklenmektedir. Parçalardan birini yanlış yere yerleştirmek istemesi durumu kontrol edilmekte ve oyunda engellenmektedir.

Oyunun kullanımı sırasında gerçekleştirilecek işlemler için birçok obje ve script dosyası oluşturulmuştur (Şekil 7.1). Bu değerlerin birbirleriyle olan bağlantısı sağlanmıştır. Oyunun giriş ekranında bir takım fonksiyonların gerçekleşmesi için buton aracı kullanılmıştır. Oyunun ilk tasarımında butonun kullanılmamasından dolayı android cihazda belirli bir noktaya tıklanması ile oyun aşamalarına geçiş yapılmaktaydı. Ancak bu şekilde tıklama yapmak oldukça zor bir hale geldiği için buton kullanımı ayrıntılı bir şekilde araştırılarak oyuna entegre edilmiştir. Oyuna ait genel çalışma diyagramı Şekil 7.2' de yer almaktadır.



Şekil 7.1. UML Diyagramı



Şekil 7.2. Programın Genel Çalışma Diyagramı

8. KARŞILAŞILAN ZORLUKLAR

Oyunun geliştirilmesi esnasında pek çok zorlukla karşılaşılmıştır. İlk olarak ilk dönemki sorunları düzeltmek adına puzzle ve eğitim aşamasındaki ögeler buton içine konarak hassasiyet ayarları düzenlenmiştir. Böylelikle resimlere tıklanabilirlik kolaylaşmıştır. Projede veri tabanı özelliği katabilmek için bir çok veri tabanı incelenmiştir. Bu süreçte SQLite isimli bir veri tabanı incelenmiştir. Çünkü bu veri tabanının bilgisayar üzerinde sabit değildir, normal şartlarda android cihazla uyum göstern bir yapısı vardır. Ancak unity platformunda bilgisayardaki IP adresini kullandığı için android cihazlarla bağlantı kurulamamıştır. Bu yüzden internet bağlantısını kullanan başka bir very tabanı araştırılması yapılmış ve Firebase adındaki very tabanı keşfedilmiştir. Bu very tabanı projeye entegre edilmiştir. Veri tabanına ilk kayıt denemesi yapıldığında kayıt işleminin gerçekleşmediği gözlemlenmiştir. Araştırmalar sonucunda veri tabanındaki kurallar seçeneği düzenlenmiştir (Şekil 8.1).

```
"rules": {
  ".read": true,
  ".write":true
}
```

Şekil 8.1. Veri Tabanı Edit Ayarı

Veri tabanında ilk kayıt işlemi gerçekleştirildikten sonra yapılan diğer kayıtlar eklenememekteydi. Veri tabanında hep en son eklenen kayıt bulunacak şekilde gösterilmekteydi. Bunun çözümünü eski kod yapısı update görevinde çalıştığı için kodda bir takım düzenlemeler yapılarak sorun giderilmiştir. Kayıtlar başarılı bir şekilde alındıktan sonra kayıtlı kullanıcılar için kaydedilmiş bilgilerin oyun sırasında güncellenmesi işlemine geçilmiştir. Bu işlemde ilk önce araştırma yapılmış ancak update işlemine dair bir unity ve veri tabanında üzerinde çalışılabilecek uygun bir metot bulunamamıştır. Bunun üzerine kendi içerisinde bulunan ekleme (SetRawJsonValueAsync(json)) ve silme (RemoveValueAsync()) fonksiyonları kullanılarak manuel bir şekilde güncelleme fonksiyonu yazılmıştır. Bu fonksiyonun oluşturulması esnasında Hash gibi java objeleri ve pek çok sınıf kullanılmıştır. Bu sayede update işlemi başarı ile gerçekleştirilmiştir.

Tüm bu işlemler gerçekleştirildikten sonra projenin geliştirilmesi esnasında çok büyük bir talihsizlik yaşanmıştır. Unity editorüne ve Firebase veri tabanına 11 Mayıs 2017 tarihinde yapılan bir güncelleme sonucunda Unity eski sürümü açılamaz hale gelmiştir. Bu güncelleme yeni Unity'nin bazı programlarla uyumluluğunu bozmuştur. Güncellemeden önce çalışan Unity programı, güncellemeden sonra bilgisayarda bulunan "Adware" isimli koruma programının varlığında çalışamaz hale gelmiştir. Bunun üzerine Unity eski verisiyonu dahil olmak üzere bilgisayardaki tüm Unity ögeleri ve Unity ile uyumsuz çalışan programlar kaldırılmıştır. Bu sorun çözüldükten sonra aynı tarihte güncellemesi yapılan veri tabanıyla yeni Unity'nin uyum sorunu yaşadığı tespit edilmiştir. Bunun üzerine veri tabanı bağlantısının ve işlemlerini sağlayan tüm kod yapısı yenisi ile uyumlu hale getirilmiştir.

9. OYUN TEST EDİLMESİ

Oyunumuzun geliştirilmesi tamamlandıktan sonra otistik çocuklar üstünde denemiştir. Oyunun amacına yönelik çalışma durumunu, çocuklar üzerindeki başarı sonuçlarını elde edebilmek için otistik çocuklar ve eğitmenler eşliğinde bir çalışma yapılmıştır. Bu sebeple Gazi Eğitim Fakültesi Özel Eğitim Bölümü Uygulama Merkezi ziyaret edilmiştir. Eğitmenler ve çocuklarla görüşülmüştür (Şekil 9.1 ve Şekil 9.2). Orada bulunan çocuklara eğitmenle eşliğinde oyunun oynatılması sağlanmış ve çocukların verdiği tepkiler, oyunu tamamlayabilme becerileri gözlemlenmiştir.

Bu merkezde 2-6 yaş aralığındaki çok sayıda otistik çocukla test gerçekleştirilmiştir. Ancak geliştirilen oyun çok küçük yaştaki çocukların bazılarında duygu durumlarını bilmedikleri için tam anlamıyla sonuca ulaşılamamıştır. Duyguları bilen çocukların ise oyunu rahatlıkla oynayabildikleri gözlemlenmiştir. Resimlere tıklamadaki hassasiyetin geliştirilmesi çocukların oyunu daha rahat oynayabilmesini sağlamıştır. Çocukların eğitmenleri eşliğinde oyunu daha kolay oynayabildikleri, oyundan sıkıldıklarında ise “geri dönüş” butonunu kendilerinin kullandığı gözlemlenmiştir. Ayrıca oyundan sıkıldıklarında eğitmenlerinin yardımı ile oyunu sürdürmeleri sağlanmaktadır.



Şekil 9.1. Gazi Eğitim Fakültesi Özel Eğitim Merkezi



Şekil 9.2. Gazi Eğitim Fakültesi Özel Eğitim Merkezi Eğitmenleri

Bu merkezde farklı özellik ve yaş aralığındaki pek çok otizmli çocuklarla oyunun test yapılmıştır. Yapılan bu çalışmada çocukların oyuna verdikleri tepkiler gözlemlenmiştir. Gözlem sonuçlarına göre oyunda kullanılan ses, video efektleri ve renkli arka plan çocukların dikkatini çekmiş ve oyuna odaklanmalarını sağlamıştır. Oyunun ilerleyen aşamalarında çocukların dikkati dağılmakta ve eğitmenlerin yönlendirmesine ihtiyaç duymaktadırlar.

Daha önceki merkezde yapılan test sonuçlarına göre çocukların, oyunun hem eğitim hem de puzzle kategorilerinde yer alan resimlere basmakta zorluk çektikleri gözlemlendiği için resimlere basılmadaki sorunlar giderilmiştir. Yapılan geliştirme sayesinde çocukların doğru seçimleri daha kolay yaptıkları gözlemlenmiştir. Oyunun öğrenme aşamasında kullanılan animasyon videoları çocukların dikkatini çekmiş ve onları mutlu etmiştir. Oyun içerisinde yer alan sesli talimatlar çocukların görevleri yerine getirmesinde büyük fayda sağlamıştır. Puzzle aşamasının 2 parçadan oluşması çocuğun anlamasını ve görevi tamamlayabilmesini kolaylaştırmıştır. Ayrıca puzzle oynamadan önce ekranda gösterilen puzzle animasyonu ile çocukların puzzle oynamaları kolaylaştırılmıştır. Bu tür oyunlar duygu durumlarını bilen çocukların bilgilerini pekiştirmelerini ve hoş vakit geçirmelerini sağlamaktadır. Daha karmaşık, görselliği az olan, ses materyalleri bulunmayan ve yazılı oyunlar çocukların oynamasını zorlaştırmaktadır. Bu yüzden yapmış olduğumuz proje özellikle duygu durumlarını bilen çocuklar için daha yararlı ve kullanışlı olmaktadır. Ayrıca kullanılan pop-up ekranlarındaki veri tabanına kayıt işlemi sayesinde eğitmenler çocukların durumlarını, gelişimlerini daha iyi şekilde gözlemlemişlerdir.

10. SONUÇ VE TARTIŞMA

Projemizde otistik çocuklar için yararlı olmak ve iyi bir öğrenme sağlamak amacıyla pek çok araştırma yapılmıştır. Oyun geliştirilmeden önce bu çocuklar hakkında geniş kapsamlı araştırmalar yapılmıştır. Pek çok makale okunmuş, bu konuda yardımcı olabilmesi için hocamız Suat Özdemir'in tavsiyesi üzerine otistik çocukların eğitiminde görev alan uzman kişilerle görüşülmüştür. Onların tavsiye ve istekleri üzerine oyunun gidişatı belirlenmiştir. Ayrıca oyunda kullanılan eğitim seti, Selda Özdemir ve ekibi tarafından tedarik edilmiştir. Oyuna ait çocuğu uyarıcı ses ögeleri kullanılmıştır. Duygu ifadelerinin öğretimi için kullanılan animasyon videoları sayesinde çocuğun dikkati dağılmadan oyunu oynaması kolaylaşmıştır. Çocuğun başarısı yıldız animasyonu ve ses ögeleri ile teşvik edilmiştir. Gerçek insan fotoğraflarını kullanılması çocuğun duygu durumlarını gerçek hayatla eşleştirebilmesi kolaylaşmıştır. Oyunun en son testi ilgili kurumlar aracılığıyla yapılmıştır.

Oyunun test işleminde sonuçlar gözlemlenmiştir. Çocuklar oyunu ilgi ve başarı ile eğitmenleri eşliğinde takip edebilmektedirler. Çocukların yaş seviyeleri ve hastalık dereceleri oyunu oynamalarında etkili olmaktadır. Oyunun puzzle aşamasında yer alan 2 parçalı zorluk düzeyi otizmli bir çocuk için yeterli görülmüştür. 4 veya daha fazla düzeydeki seviyelerde çocuğun zorlanabileceği uzman eğitmenlerle kararlaştırılmıştır. Bu sebeple oyun 2 seviyeli puzzledan oluşması gerektiği sonucu çıkarılmıştır. Oyunda duygu durumu için kullanılan insan resimlerinin otistik çocuklar için daha hassas olması gerektiği gözlemlenmiştir. Bu yüzden resimlerde buton kullanılmıştır. Her çocuğun başarısını ölçmek için de veri tabanı eklenmesi gerçekleştirilmiştir. Oyunu çocukların eğitmenleri eşliğinde oynamasına ihtiyaç olduğu anlaşılmıştır. Çünkü oyuna yapılan güncellemeler daha yetişkin bir bireyin müdahalesine ihtiyaç duymaktadır. Oyunu önce yetişkin anlamalı ve çocuğu ona göre yönlendirmelidir. Oyundaki kısıtlamalarla çocuklar oyunu oynamada zorlanmaktadır.

Otizmli çocuklar için geliştirilen mobil oyunlar daha kolay oynanabilmesi ve başarılı sonuçların elde edilebilmesi için anlaşılabilir ve basit ara yüzlerle oluşturulması, ses ve görsel ögelerle desteklenmelidir. Çünkü bu tür ögeler otistik çocukların ilgisini çekerek öğrenmelerini kolaylaştırmaktadır. Bu tür uygulamaların her kademeden insanın erişip fayda verebilmesi için kâr amacı güdülmeden geliştirilmelidir.

REFERANSLAR

[1] Eliçin Ö., Avcıoğlu H. (2014). "Otizmi Olan Çocuklara Duyguları Ayırt Etme Becerisi Kazandırmada Replik Silikleştirme İle Yapılan Öğretimin Etkililiği", ***Uluslararası Eğitim Teknolojileri Konferansı***, 171(39), 317 – 328.

[2] Aslan, Y. G., İftar, G. K., Uzuner, Y. (2009). "Otistik Çocuklar İçin Davranışsal Eğitim Programı (OÇİDEP) Ev Uygulamasının Bir Çocukla İncelenmesi", ***Ankara Üniversitesi Eğitim Bilimleri Fakültesi Özel Eğitim Dergisi,*** 10 (1) 1-25.

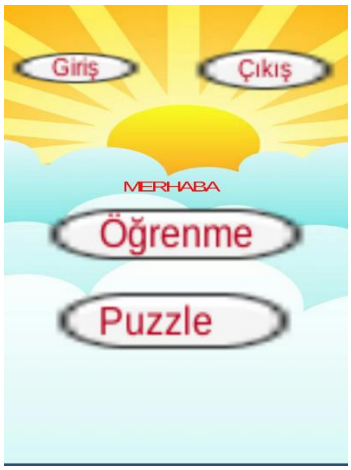
[3] Veziroğlu, M., Gönen, M. (2012). "Resimli Çocuk Kitaplarının M. E. B. Okul Öncesi Eğitim Programı'ndaki Kazanımlara Uygunluğunun İncelenmesi", ***Eğitim ve Bilim,*** 163 (37) 227-238.

[4] Korkmaz, M., Yiğiter, Ş. Y., Güney, S. Ceylan, N. (2012). "2010 Öncesindeki Sağlık Sisteminin Literatür Açısından Değerlendirilmesi", ***Uluslararası Hakemli Akademik Spor Sağlık Ve Tıp Bilimleri Dergisi,*** 4(2) 1-97.

[5] İnternet: Autism games, play.google.com. https://play.google.com/store?hl=tr (26.12.2016).

EKLER

EK 1. Giriş ekranı



Şekil 1.1. Giriş ekranı



Şekil 1.2. Giriş butonuna tıklanması



Şekil 1.3. Seviye seçilmesi

```
public GameObject nameDialog2, nameDialog3;
public void onClick_egitim()
{
    choose = 1;
    FirebaseDatabase.DefaultInstance
  .GetReference("friends")
  .GetValueAsync().ContinueWith(task1 =>
  {
      if (task1.IsFaulted)
      {
          // Handle the error...
          Debug.Log("hataaa");
      }
      else if (task1.IsCompleted)
      {
          DataSnapshot snapshot1 = task1.Result;

          Debug.Log("girdim");
          User2 u = new User2();
          foreach (var childSnapshot1 in snapshot1.Children)
          {
              if (childSnapshot1.Child("username").Value.ToString().Equals(optionSelect.isim))
              {
                  totalScore = Convert.ToInt32(childSnapshot1.Child("score").Value);
                  elevel = Convert.ToInt32(childSnapshot1.Child("level").Value);
                  levelcount = elevel;
                  eslesme = true;
                  Debug.Log(eslesme);
              }
          }

      }
      if (eslesme)
          nameDialog2.SetActive(!nameDialog2.activeSelf);
      else
          SceneManager.LoadScene("leveltransition");

  });

}
public static Boolean eslesme = false,peslesme=false;
public void onClick_secim(Button button)
{
```

Şekil 1.4. Oyuna giriş yapılması

```
public static Boolean eslesme = false,peslesme=false;
public void onClick_secim(Button button)
{

    if (button.name == "evet")
    {
        if (levelcount == 0)
        {
            levelkayit = levelcount;
            SceneManager.LoadScene("mutluvideo");
        }
        else if (levelcount == 1)
        {
            levelkayit = levelcount;
            SceneManager.LoadScene("mutsuzvideo");

        }
        else if (levelcount == 2)
        {
            levelkayit = levelcount;
            SceneManager.LoadScene("kizginvideo");
        }
        else if (levelcount == 3)
        {
            levelkayit = levelcount;
            SceneManager.LoadScene("korkmusvideo");
        }
        else if (levelcount == 4)
        {
            levelkayit = levelcount;
            SceneManager.LoadScene("mutlu_stage1");
        }
        else if (levelcount == 5)
        {
            levelkayit = levelcount;
            SceneManager.LoadScene("mutsuzstage1");
        }
        else if (levelcount == 6)
        {
            levelkayit = levelcount;
            SceneManager.LoadScene("kizginstage1");
        }
        else if (levelcount == 7)
        {
```

Şekil 1.4. (devam) Oyuna giriş yapılması

```
            levelkayit = levelcount;
            levelcount=0;
            SceneManager.LoadScene("saskin_stage1");
        }
        else
            SceneManager.LoadScene("maingamemenu");
    }
    else
    {
        levelcount = 0;
        SceneManager.LoadScene("leveltransition");
    }
}
public void onClick_puzzleSecim(Button button)
{

    if (button.name == "evet")
    {
        if (puzzleseviye == 0)
        {
            SceneManager.LoadScene("otizm");
        }
        else if (puzzleseviye == 1)
        {
            optionSelect.levelp=1;
            SceneManager.LoadScene("otizm2");
        }
        else if (puzzleseviye == 2)
        {
            optionSelect.levelp=2;
            SceneManager.LoadScene("otizm3");
        }
        else if (puzzleseviye == 3)
        {
            optionSelect.levelp=3;
            SceneManager.LoadScene("otizm4");
        }
        else
        {
            optionSelect.puzzlecount = 0;
            optionSelect.puzzleseviye = 0;
            SceneManager.LoadScene("maingamemenu");

        }
```

Şekil 1.4. (devam) Oyuna giriş yapılması

```
        SceneManager.LoadScene("animasyon");
    }


}
public void onClick_puzzle()
{
    choose = 2;
    FirebaseDatabase.DefaultInstance
 .GetReference("friends")
 .GetValueAsync().ContinueWith(task1 =>
 {
     if (task1.IsFaulted)
     {
         // Handle the error...
         Debug.Log("hataaa");
     }
     else if (task1.IsCompleted)
     {
         DataSnapshot snapshot1 = task1.Result;
         Debug.Log("girdim");
         User2 u = new User2();
         foreach (var childSnapshot1 in snapshot1.Children)
         {
             if (childSnapshot1.Child("username").Value.ToString().Equals(optionSelect.isim))
             {
                 puzzleScore = Convert.ToInt32(childSnapshot1.Child("score").Value);
                 pzlevel = Convert.ToInt32(childSnapshot1.Child("plevel").Value);
                 puzzleseviye = pzlevel;
                 peslesme = true;
                 Debug.Log(peslesme);
             }
         }

     }
     if (peslesme)
         nameDialog3.SetActive(!nameDialog3.activeSelf);
     else
         SceneManager.LoadScene("animasyon");

 });

}
```

Şekil 1.4. (devam) Oyuna giriş yapılması

EK 2. Puzzle



Şekil 2.1. Puzzle aşaması

EK 3.Çıkış



Şekil 3.1. Eğitim aşaması



Şekil 3.2. Kayıt aşaması

```
void Start()
{
    // Set up the Editor before calling into the realtime database.
    FirebaseApp.DefaultInstance.SetEditorDatabaseUrl("https://fir-453d5.firebaseio.com/");

    // Get the root reference location of the database.
    reference = FirebaseDatabase.DefaultInstance.RootReference;
}
public void onClick()
{

    nameDialog.SetActive(!nameDialog.activeSelf);
}
public void ekle()
{
    if (optionSelect.sayac==0)
    {
        optionSelect.isim = enterName.text;
          Debug.Log("ifteyiz" + optionSelect.isim);
        enterName.text = string.Empty;
        nameDialog.SetActive(!nameDialog.activeSelf);
        optionSelect.sayac++;
    }
    else{
        optionSelect.sayac = 0;
        nameDialog.SetActive(!nameDialog.activeSelf);
    }
}
public void Update()
{
}
public void kayit()
{
     Debug.Log("yazdım2");
    Debug.Log(optionSelect.isim);
    Debug.Log(optionSelect.levelkayit);
    Debug.Log(optionSelect.levelp);
    if (optionSelect.totalScore != 0)
    {
        Debug.Log(optionSelect.isim + optionSelect.totalScore);
        if(optionSelect.puzzleScore!=0)
            UpdateDatabase(optionSelect.isim, optionSelect.totalScore + optionSelect.puzzleScore, optionSelect.levelkayit, optionSelect.levelp);
        else
            UpdateDatabase(optionSelect.isim, optionSelect.totalScore, optionSelect.levelkayit, optionSelect.levelp);
```

```
    }
    if (optionSelect.puzzleScore != 0)
    {
        Debug.Log(optionSelect.isim + optionSelect.puzzleScore);
        if (optionSelect.totalScore != 0)
            UpdateDatabase(optionSelect.isim, optionSelect.totalScore + optionSelect.puzzleScore, optionSelect.levelkayit, optionSelect.levelp);
        else
            UpdateDatabase(optionSelect.isim, optionSelect.puzzleScore, optionSelect.levelkayit, optionSelect.levelp);
    }
    if (optionSelect.totalScore == 0 && optionSelect.puzzleScore == 0)
    {
        Debug.Log("oynamadı");
        UpdateDatabase(optionSelect.isim, 0, optionSelect.levelkayit, optionSelect.levelp);
    }
}
public void goster()
{
    ShowScores();
}

public DataSnapshot dataSnapshot;

private List<HighScore> highScores = new List<HighScore>();

public Text ad;

private void ShowScores()
{

    GetScores();
    FirebaseDatabase.DefaultInstance
  .GetReference("friends")
  .GetValueAsync().ContinueWith(task =>
  {
      if (task.IsFaulted)
      {
          // Handle the error...
          Debug.Log("hataaa");
      }
      else if (task.IsCompleted)
      {
          for (int i = 0; i < topRanks; i++)
          {
```

Şekil 3.3. Kayıt aşamasındaki işlemler

```
            if (i <= highScores.Count - 1)
            {
                HighScore tmpScore = highScores[i];
                ad.text =tmpScore.Name;

            }
        }

    }
});


}

public HashSet<User2> hash = new HashSet<User2>();
public Boolean up = false;
int s = 0;
public void UpdateDatabase(string n,int e,int l, int pl)
{
    FirebaseDatabase.DefaultInstance
   .GetReference("friends")
   .GetValueAsync().ContinueWith(task1 =>
   {
       if (task1.IsFaulted)
       {
           // Handle the error...
           Debug.Log("hataaa");
       }
       else if (task1.IsCompleted)
       {
           DataSnapshot snapshot1 = task1.Result;
           highScores.Clear();
           Debug.Log("girdim");
           User2 u=new User2();
           foreach (var childSnapshot1 in snapshot1.Children)
           {
               if (childSnapshot1.Child("username").Value.ToString().Equals(n))
               {
                   u.key2 = childSnapshot1.Key.ToString();
                   u.username2 = n;
                   u.score2 = e;
                   if (optionSelect.levelkayit != 0 || Convert.ToInt32(childSnapshot1.Child("plevel").Value) == optionSelect.levelp)
                   {
                       u.plevel2 = Convert.ToInt32(childSnapshot1.Child("plevel").Value);
```

```
           u.plevel2 = pl;
           Debug.Log("B");
       }
       if (optionSelect.levelp != 0 || Convert.ToInt32(childSnapshot1.Child("level").Value) == optionSelect.levelkayit)
       {
           u.level2 = Convert.ToInt32(childSnapshot1.Child("level").Value);
           Debug.Log("C");
       }
       else
       {
           u.level2 = l;
           Debug.Log("D");
       }
       hash.Add(u);
       s = u.score2;
       up = true;
   }
   else
   {

       s = u.score2;
       hash.Add(new User2 { key2 = childSnapshot1.Key.ToString(), username2 = childSnapshot1.Child("username").Value.ToString(), score2 = Convert.ToInt32(childSnapshot1.Child("score").Value), level2 = Convert.ToInt32(childSnapshot1.Child("level").Value), plevel2 = Convert
   }
}
reference.RemoveValueAsync();
Debug.Log("sildim");
if (up) {
    foreach (var item in hash)
    {
        writeUser(item.key2, item.username2, item.score2,item.level2,item.plevel2);
        Debug.Log("eski yazdım");
    }
}
else
{
    foreach (var item in hash)
    {
        writeNewUser(item.username2, item.score2,item.level2,item.plevel2);
        Debug.Log("yeni yazdım");
    }
    writeNewUser(n, e,l,pl);
}
nameDialog.SetActive(!nameDialog.activeSelf);
enterName.text = n.ToUpper() + " Skor:" + Convert.ToInt32(e);
```

Şekil 3.3. (devam) Kayıt aşamasındaki işlemler

```
            up = false;
        }

    });
}
public void writeUser(string key, string name, int score, int level, int plevel)
{
    String k = key;
    User friend = new User(name, score,level,plevel);
    string json = JsonUtility.ToJson(friend);

    reference.Child("friends").Child(k).SetRawJsonValueAsync(json);
}

private void GetScores()
{

     FirebaseDatabase.DefaultInstance
  .GetReference("friends")
  .GetValueAsync().ContinueWith(task =>
  {
      if (task.IsFaulted)
      {
          // Handle the error...
          Debug.Log("hataaa");
      }
      else if (task.IsCompleted)
      {
          DataSnapshot snapshot = task.Result;
          highScores.Clear();
          int s = 0;
          foreach (var childSnapshot in snapshot.Children)
          {
              highScores.Add(new HighScore(s, Convert.ToInt32(childSnapshot.Child("score").Value), childSnapshot.Child("username").Value.ToString()));
          }
      }

  });
}
public void writeNewUser(string name, int score,int level,int plevel)
{
    GetScores();

    Debug.Log("write içindeyim");
```

```
        String key = reference.Push().Key;
        User friend = new User(name, score,level,plevel);
        string json = JsonUtility.ToJson(friend);

        reference.Child("friends").Child(key).SetRawJsonValueAsync(json);

    }
}

public class User2
{
    public string key2;
    public string username2;
    public int score2,level2,plevel2;

}

public class User
{
    public string username;
    public int score,level,plevel;

    private User()
    {
    }

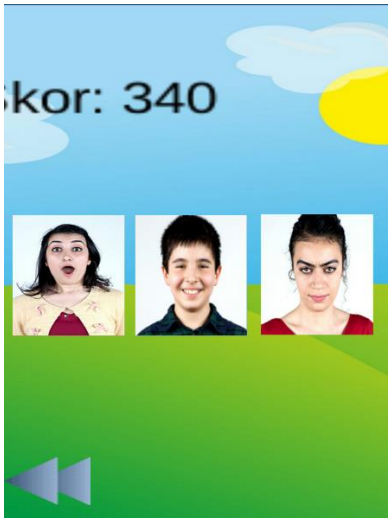
    public User(string username, int score, int level,int plevel)
    {
        this.username = username;
        this.score = score;
        this.level = level;
        this.plevel = plevel;
    }
}
```

Şekil 3.3. (devam) Kayıt aşamasındaki işlemler

EK 4. Öğrenme



Şekil 4.1. İki resimli öğrenme aşaması



Şekil 4.2. Üç resimli öğrenme aşaması