Öğrenci,

Ad Soyad : Nurullah Samed SAVAŞ

Numara : 130201069

Bölümü : Bilgisayar Mühendisliği

Öğrenci,

Ad Soyad : Büşra KARNUCU

Numara : 130201076

Bölümü : Bilgisayar Mühendisliği

Öğrenci,

Ad Soyad : Hande AYDEMİR

Numara : 130201003

Bölümü : Bilgisayar Mühendisliği

Ders Adı : Yazılım Laboratuvarı II

Proje numarası : 2

Proje konusu : Balon Patlatma Oyunu

Proje danışmanı : Arş.Gör Hikmetcan ÖZCAN

Proje Veriliş Tarihi : 20.04.2016

Proje Ortamı : Java / LibGdx

**1 . GİRİŞ**

Bu rapor oyun projesinin yazılım konusunu,yapımı aşamasında karşılaşılan problemlerini ve çözümlerini bulundurduğu yazılım mimarisini, nasıl açılacağı ve kullanılacağı, ekran çıktısını ve yapılırken yararlanılan kaynaklarını içermektedir.

**2. PROBLEM TANIMI**

Projede bizden istenen mobil platformda bir balon patlatma oyunu geliştirmektir. Bu oyunu geliştirirken bizden bazı kurallara uymamız istenmektedir. Bunlar aşağıda maddeler halinde yazılmıştır.

1) Öncelikle proje geliştirilirken herhangi bir oyun geliştirme platformu kullanılmayacaktır. (Örn : Unity)

2) Proje Android,Object C ya da Windows Phone platformlarında geliştirilebilir. Bunlar arasındaki seçim yazılımcıya bırakılmıştır.

3) Oyun başlangıç arayüzünde (Ana Menüde) Yeni Oyun, Hakkında(Geliştiricilerin okul numaraları Ad ve Soyadları bulunması yeterlidir.) , ses aç/kapa özelliği ve Çıkış sekmesinin bulunması gerekmektedir.

4) Oyun içinde ses efekti eklememiz gerekmektedir.

5) Oyuncu ana menüden yeni oyun sekmesiyle giriş yapacaktır.

6) Oyunda 4 farklı balon tipi bulunacaktır. Bunlar kırmızı, sarı, yeşil, siyah balonlardır.

7) Kırmızı balonun özelliği harita içerisinde hızlı bir şekilde rasgele hareket etmektedir. Kullanıcı balonu patlatırsa 10 puan kazanır.

8) Sarı balonun özelliği harita içerisinde rasgele bir yerde sadece 1 sn. görünecektir. Kullanıcı balonu patlatırsa 20 puan kazanır.

9) Yeşil balonun özelliği devamlı olarak haritanın en aşağısından yukarıya doğru hızlı bir şekilde ilerleyecektir. Ayrıca yeşil balon rasgele bir anda siyah balona dönüşebilir. Kullanıcı yeşil balonu patlatırsa 5 puan kazanacaktır.

10) Siyah balonun özelliği devamlı olarak haritanın en aşağısından yukarıya doğru hızlı bir şekilde ilerleyecektir. Ayrıca siyah balon rasgele bir anda yeşil balona dönüşebilir. Kullanıcı siyah balonu patlatırsa 10 puan kaybedecektir.

11) Oyunda seviye atlayabilmek için 30 sn. içerisinde her balondan en az bir kez patlatmış olması ve toplamda en az 100 puan kazanmış olması gerekmektedir. Şartlar sağlanmazsa oyun sonlanarak 13. maddedeki işlemler gerçekleştirilip, ana menüye yönlendirilecektir.

12 ) Oyun 3 tip bölümden oluşacaktır. Bölüm tipleri yazılımcının tercihine bırakılmıştır.

13) Oyunda her bölüm sonunda kaç puan aldığı hesaplanıp oyuncuya gösterilmelidir.

14) Toplam puan oyun sonunda tüm bölümlerden almış olduğu puanların toplamı olarak gösterilmelidir.

**3. YAPILAN ARAŞTIRMALAR**

Oyun geliştirilirken öncelikle oyunun hangi platformda gerçekleştirileceği karar verildi. Bu konuda bizden 3 farklı platform arasında bir seçim yapmamız istenmekteydi ( Android,Object C ya da Windows Phone) . Biz yaygın kullanımı gerekçesiyle oyunumuzu Android platformunda gerçekleştirme kararı verdik ve öncelikle Android Uygulama geliştirme hakkında bilgi edinmeye başladık. [1]

Genel bir bilgi edindikten sonra daha özele inerek Androidde oyun geliştirme kütüphanelerini araştırmaya başladık. Bizim işimizi görecek ve yeterli basitliğe sahip olan LibGdx kütüphanesi kullanma kararı aldık. Oyunumuzu geliştirirken Libgdx oyun programlama kütüphanesi kullanıldı. Libgdx, Badlogic Games ve bir çok bağımsız katılımcı tarafından sürekli olarak güncellenmekte olan , open-source, openGl ES kullanan 2d, 3d oyunların geliştirilme işlemini çok basitleştiren bir kütüphanedir.Libgdx'in en iyi yönlerinden biri cross platform olmasıdır. Kodda çok ufak değişikler yaparak bilgisayar için yazdığımız bir programı android/html platformlarında kolaylıkla çalıştırabiliriz. Diğer bir özelliği ise içersinde birden fazla UI kütüphanesi bulundurmasıdır. Örneğin oyunumuzu “Game” kütüphanesi kullanarak yazarken ,server ya da geliştirici araçlarınızı “Scene2d” kütüphanesi ile oluşturabiliriz [2].

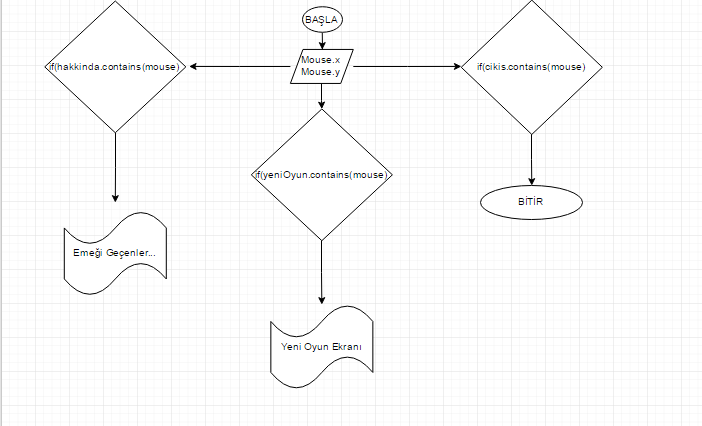
Kütüphanenin nasıl kurulacağı kullanılacağı hakkında bilgi edinmeye başladık.Referanslarda belirtilen link doğrultusunda bu sorunuda çözmüş olduk. Kurulum için kaynakçada da belirtilen Libgdx'in resmi sitesinden indirdiğimiz jar dosyasını çift tıklayarak açıyoruz. Proje ismi, oyun ismi, klasör ismi gibi seçenekleri istediğimiz doğrultuda doldurduktan sonra projemizin çalışmasını istediğimiz platformları seçerek generate butonuna tıklıyoruz. Belli bir süre bekledikten sonra açılan pencerenin altındaki konsol penceresinde Build Succesfuly yazısını gördükten sonra pencereyi güvenli bir şekilde kapatabiliriz. Daha sonra Android Studio derleyicisini açarak sol üstten file sekmesine gelerek import project diyoruz ve Libgdx'in jarıyla oluşturduğumuz dosyayı seçerek import diyoruz. Böylelikle Libgdx'in çekirdeğini oluşturmuş oluyoruz[3].

Kütüphanenin kurulumundan sonra kütüphanenin barındırdığı fonksiyonlara ve sınıflara alışmak amacıyla basit örneklere bakmaya başladık. [4] . Öncelikle yine kaynakçada belirtilen basit örneği android studio da gerçekleştirdikten sonra libgdx hakkında temel bir bilgiye sahip olmuş olduk ve bu proje, geliştirmiş olduğumuz oyunda bizim temel aldığımız yapı oldu. Ancak bu yapıda bize sadece oyun içi ekranın nasıl koda döküleceği anlatılmaktaydı menü oluşturulması için herhangi bir içerik barındırmıyordu. Bunun içinde başka kaynaklara bakıldı ve yine basit bir proje aracılığıyla menü oluşturmak için gerekli bilgiyi öğrendik [5]. Böylelikle projenin geliştirilmesi için temel bilgiye sahip olmuş olduk.

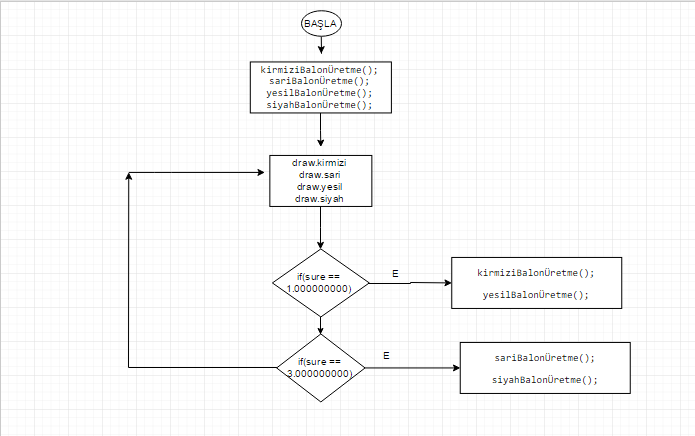
**4. TASARIM**

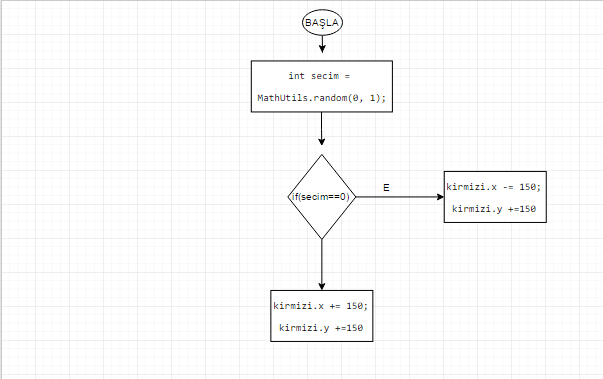
Bu başlıkta projenin akış şeması çizilecek ve yazılım mimarisinden bahsedilecektir.

**4.1. Akış Şeması**

****

*Resim 1 : Ana Menü Akış Diyagramı*

*Resim 2 : Balonların Üretilmesi*

*Resim 3 : Kırmızı Balonların Random Hareketi*

**4.2. Yazılım Mimarisi**

Oyun Java dilinin libgdx kütüphanesi kullanılarak geliştirilmiştir. Oyunun menü ekranı Oyna, Hakkında, Çıkış ve Ses Aç/Kapa butonlarını barındırmaktadır. Bu butonların çalışma mantığı şu şekildedir:

Ekranda görülen butonlar aslında birer resimdir. Libgdx kütüphanesinin içinde bulunan Rectangle sınıfı aracılığıyla bu resimler etrafına görünmez kareler çizdirilir. Bu karelerin amacı mouse tıklamasını algıladıktan sonra tıklanan yerin bu karelerin içi mi yoksa dışı mı olduğunu belirlemektir. Bu şekilde hangi resmin üstüne tıkladığımızı belirledikten sonra tıkladığımız resme göre gerekli sınıf veya metodları çağırarak ekrana çizdirdiğimiz bu resimlere bir buton gibi işlev vermiş oluyoruz.

**4.2.1 Hakkında Butonu**

Yukarıda belirtilen yol ile hakkında butonuna tıklanıldığını belirledikten sonra aşağıdaki işlemler yapılır:

Hakkında butonuna tıklanıldığında mevcut MainMenu sınıfındaki tüm resimler dispose() methodu ile yokedilerek yeni sınıfa ve ekrana geçmek için önhazırlık yapılır daha sonra setScreen metodu ile yeni ekranımıza geçmemizi sağlayacak Hakkında sınıfının yapıcı fonksiyonu çağrılır. Böylelikle artık Hakkında sınıfına geçmiş oluruz ve bu sınıf içindeki komutlar doğrultusunda yeni ekranımızı çizme işlemi yapabiliriz. Bu sınıfta ekrana projeyi gerçekleştiren grup üyelerinin ismi yazan bir resim ekrana çizdirilir. Bu resmin y ekseni sabit bir sayı değildir. Render() fonksiyonun içinde y ekseni sürekli artırılarak isimlerin yukarı doğru çıkması sağlanır. Ayrıca bir if komutu sayesinde resimlerin ekran dışına çıktığında tekrar y ekseninin baştakı değere dönmesi sağlanarak bu işlemi döngüsel şekilde yapmış oluruz. Ayrıca bu ekrandayken **if** (Gdx.*input*.isTouched()) komutu sayesinde ekrana tıklanıldığını anladığımızda herhangi bir koordinat algılama yapmaksızın yine mevcut sınıfın tüm nesneleri dispose edilir ve yine MainMenu sınıfıda geçilir.

**4.2.2 Çıkış Butonu**

Çıkış butonunun çalışma mekanizması hakkında butonuyla büyük benzerlikler göstermektedir. Bu butona tıklanıldığını anlayan program Gdx.*app*.exit(); komutunu çağırır ve böylelikle oyundan çıkılır.

**4.2.3 Oyna Butonu**

Oyna butonu diğer butonlar gibi kısa işlemler yapan bir buton değildir. Kullanıcı Oyna butonuna tıklandığında birden fazla sınıf çağırılıp yokedilecek uzun bir döngü başlatmış olur. Bu işlem döngüsü aşağıda anlatılacaktır.

Öncelikle diğer iki butonda olduğu gibi kullanıcın Oyna butonuna tıkladığı tesipt edildiğinde mevcut sınıfın içinde bulundurdğu resim müzik gibi nesneler yokedilir ve GameScreen sınıfına geçilir. GameScreen sınıfının yapıcı fonksiyonu çağırıldığında öncelikle her bir balon çeşidinden bir adet üretilir. Bu üretme işlemi Rectangle tipinde değişkenler tutan bir dizi sayesinde gerçekleştirilir. Üretilecek her balonun koordinatları, genişliği ve yüksekliği bu dizide tutulur. Bu bilgiler kullanılarak daha önce yüklenen balon resimleri ekrana çizdirilir.

Render() fonksiyonu içerisinde TimeUtils.*nanoTime*() komutu kullanılarak geçen süre hesaplanır ve bu şekilde her saniyede bir kırmızı balon ve bir yeşil balon, her 3 saniyede bir siyah balon, her 4 saniyede bir ise sarı balon üretme fonksiyonları çağırılır. Oluşturduğumuz Rectangle diziilerini sürekli dolaşan bir while döngüsü yazılarak oluşturulan tüm balonların hareketleri sağlanır. Bu hareketler her balon için farklı olduğundan her balon için ayrı bir Rectangle dizisi oluşturulmuştur. Bunların ismi kirmizibalonlar, saribalonlar, yesilbalonlar, siyah balonlar şeklinde belirlenmiştir.

Kırmızı balonun hareketi random oalcak şekilde ayarlanmıştır. Bu rastgelelik ise 0 ile 1 arasında bir tam sayı üreten matematik kütüphanesi aracılığıyla sağlanmıştır. Eğer random fonksiyonumuz 0 üretirse kırmızı balonumuz sola 1 üretirse de sağa gidecek şekilde ayarlanmıştır. Sari balonun hareketi ise tamamen random bir yerde 1 saniye gözüküp kendiliğinde yok olacak şekilde ayarlanmıştır. Bu hareket ise yine random fonsiyonu kullanılarak sağlanmıştır. Random fonksiyonlarımız bu sefer 2 tanedir . Biri 0 ile 800 arasında random bir değer üreterek x eksinin randomlığını sağlarken diğeri 0 ile 480 arasında random bir değer üreterek y ekseninin randomlığını sağlamış olur.Böylelikle sarı balonumuz ekranın random bir koordinatında belirmiş olur. Bu belirme işleminin hemen artıdan 1 saniye tutulur ve 1 saniyenin geçmesi durumunda oluşturulan balon iter.remove() metodu aracılığıyla yokedilir. Yeşil balonun hareketi ise diğerlerine göre daha basittir. Yine x ekseninde 0 ile 800 arasında bir random değer üretilerek rastgele bir noktadan çıkarken y ekseninin başlangıç değeri sabit ve sıfırdır. Bir döngü içersinde y değeri artılarak balonun hızlı bir şelikde yukarı çıkması sağlanır. Yeşil balonun birde istisnai bir durumu vardır. Projede bizden yeşil balonun random bir anda siyah balona dönmesi geretiği söylenmiştir. Bunun için ise 0 ile 100 arasında bir random değer üretilir. Bu değerimizin 0 olması durumunda yeşil balonumuz iter.remove() metodu ile silinir ve yerine aynı koordinatlarda bir siyah balon oluşturulur. Random üretme sınırının bu kadar büyük olmasının sebebi render fonksiyonun saniyede 60 kere dönerek küçük sınırlar için 0 gelme değerinin çok artmasıdır. Çok fazla 0 gelme durumunda ise yeşilden siyaha geçişler çok artıcaktır ve bu da oynanışı baltalayacaktır. Siyah balonsa tamamen yeşil balonun aynısıdır. Sadece dönüşüm siyahtan yeşile olacak şekilde değişir.

Baştada belirtildiği gibi ekrandaki tüm balonların koordinatları,genişilikleri ve yükseklikleri balonlara ait dizilerde tutulmaktadır. Bu sayede ekrana dokunulan koordinat belirlendikten sonra tek yapmamız bu Rectangle değerlerini tek tek kontrol etmek ve tıkladığımız yerin herhangi birinin içinde olup olmadığını bulmaktır. Eğer bu Rectangle değerlerinin içinde bir yere tıkladıysak tıkladığımız Rectangle değerini Rectangle dizisinden silerek balonun ekrandan silinmesi sağlanır ve tıkladığımız balon çeşidine görede puan değerine + veya – bir değer eklenir.

Oyun başladığı anda tutulan zaman eğer 30 saniyeyi geçmiş ise program bir if koşuluna girer. Eğer herbir balondan en az bir tane patlamışsa ve puanımız 100 ün üstündeyse GameScreen sınıfımız dispose edilerek OyunBasarılı sınıfına geçilir. Bu ekranda ise bize bulunduğumuz bölümde kaç puan yaptığımız gösterilerek yeni bölüme yani GameScreen2 sınıfımıza geçiş yapılır. Bu sınıfın çalışma mekanizması GameScreen sınıfıyla çoğunlukla aynıdır. Bu yüzden tekrar anlatılmayacaktır. Sadece farkını söylemek gerekirse farkı sadece arkaplan değiştirilip oyunun bitmesi için gerekli süre 20 saniyeye çekilmiştir. Şayet üstte belirtilen koşul sağlanmaz ise program OyunBitti sınıfına geçer ve mevcut bölümden alınan puan kullanıcıya gösterilerek tekrar en başa MainMenu sınıfına dönülür.

**4.2.4 Ses Aç/Kapa Butonu**

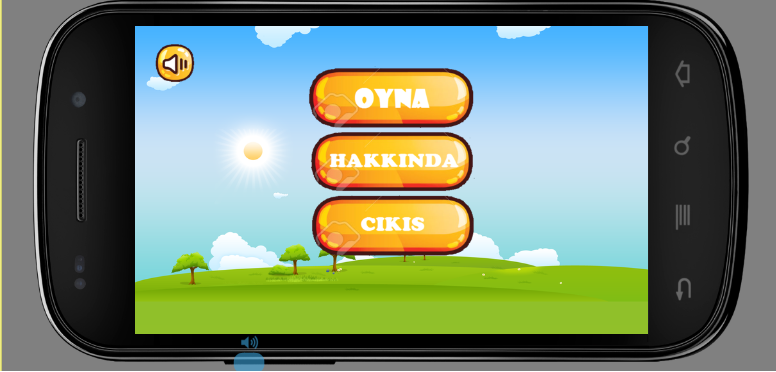
Bu butonun çalışma mantığı oldukça basittir. Programın tüm sınıflarında global bir şekilde aktarılan bir boolean ses adlı değişkenimiz bulunmaktadır. Eğer bu butona tıklanılırsa bu ses değişkeni true ise false, false ise de true yapılır. Programda ses çalınan tüm kod bloklarının başına bu ses değişkeninin değerini kontrol eden bir if komutu konulur(if(ses)). Böylelikle eğer ses değişkenimiz true ise ses efektleri ve müzikler çalarken false olduğunda bu sesler çalınmıyor olacaktır.

**4. GENEL YAPI**

Geliştirilen oyun genel olarak aşağıda belirtilen adımlar doğrultusunda çalışmaktadır.

Oyun ilk çalıştırıldığında kullanıcıyı bir menü ekranı karşılamaktadır. Bu menü ekranında Oyna,Hakkında,Cıkıs ve Ses Ac/Kapa butonları bulunmaktadır. Kullanıcı eğer Hakkında butonuna tıklarsa mevcut ekran silinir ve uygulamayı geliştiren grup üyelerinin isimleri yazan başka bir ekrana geçer. Eğer ekranın herhangi bir yerine tıklanırsa program tekrar MainMenu ekranından devam eder.Kullanıcı eğer Cıkıs butonuna tıklar ise oyunda kullanılan resim, sesler ve müzikler dispose edilir ve oyunumuz sonlanır.Kullanıcı eğer Ses Ac/Kapa butonuna tıklar ise oyunun sahip olduğu tüm sınıflarda çalınan ses efektleri açılır ve kapanır. Kullanıcı eğer Oyna butonuna tıklar ise de oyunun ilk bölümü başlamış olur. İlk bölümde 4 tip balon vardır. Bunlar kırmızı,sarı,yeşil ve siah balonlardır. Kırmızı balonlar en alttan çıkarlar ve bundan sonra hareketlerine sağ ve sol olmak üzere random bir şekilde devam ederler. Sarı balonlar ekranın herhangi bir yerinde random bir şekilde belirir ve 1 saniye sonra yokolur. Yeşil balonlar ekranın en altında çıkarak hızlı bir şekilde yukarıya doğru hareket ederler ve çıkışları esnasında herhangi bir anda siyah balona dönüşebilirler. Siyah balonlar ise hareket olarak yeşil balonlarla aynı mekaniğe sahiptirler. Aynı şekilde yukkuarı çıkarlerken herhangi bir anda yeşil balona dönüşebilirler.

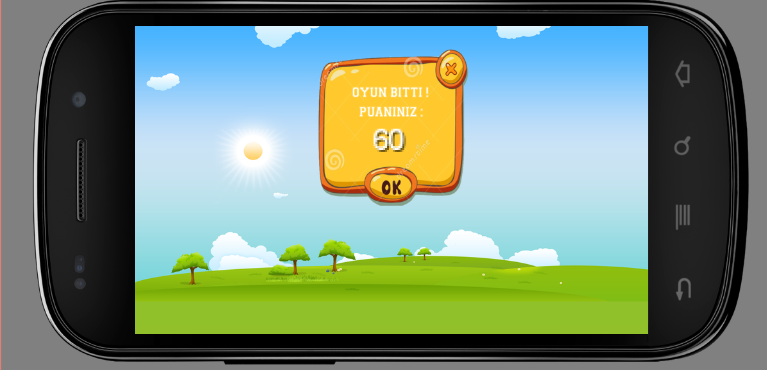
Bölümün amacı genel olarak balonlara tıklayarak patlatmaktır. Ancak bölümü geçmek için belli başlı kurallar vardır. Bölümü bitirebilmek için kullanıcının her balondan en az bir tane patlatmış olması ve 30 saniye içerisinde 100 puanı toplaması gerekir. Eğer kullanıcı 30 saniye sınırı içerisinde bu şartı sağlarsa birincci bölüm başarıyla tamamlanmış olur ve kullanıcıyı 1.bölümde yaptığı puanı gösteren bir pencere karşılar. Kullanıcı Ok butonuna tıklarsa bir sonraki bölüme geçer. Bir sonraki bölümde ise balon mekanikleri aynıdır. Farklı olan şey ise süre sınırının 30 dan 20 ye çekilmesi ve arkaplan resiminin değişmesidir. Bölüm bitirme koşuluda yine birinci bölümde olduğu gibidir. Eğer kullanıcı bu koşulu sağlarsa tekrar puanını gördüğü bir ekranla karşılaşır ve yine Ok butonuna tıkladığında 3.bölüme geçer. 3.bölümde diğer bölümlerle aynı mekaniğe sahiptir. Değişen şeyler ise süre sınırının 20 den 15' e inmesi ve arkaplan resiminin değişmesidir. Oyun bitirme koşuluda diğer iki bölümdekiyle aynıdır. Eğer kullanıcı bu koşulu sağlarsa oyunun 3 bölümüde başarıyla bitmiş olur ve kullanıcıyı 3. bölümde yaptığı puanla beraber toplam puanını da gösteren bir ekran karşılar. Kullanıcı Ok tuşuna basarak tekrar Menü ekranına dönebilir ve tekrardan yeni bir oyun başlatabilir. Eğer kullanıcı bu bölümlerin herhangi birinde bitirme koşulunu yerine getiremez ise Oyun Bitti ekranıyla karşılaşır ve mevcut bölüdeki puanı gösterilerek Menu ekranına gidilir. Kullanıcı aynı şekilde bu ekranda yeni bir oyun başlatabilir. Aşağıda bu bahsedilen ekranların görüntüleri yer almaktadır.

 *Resim 4 : Menu Ekranı*



*Resim 5 : Bölüm 1*



*Resim 6 : Oyun Bitti Ekranı*



*Resim 7 : Hakkında Ekranı*

**5 - KAYNAKLAR**

1. <https://gelecegiyazanlar.turkcell.com.tr/konu/android>
2. <https://libgdx.badlogicgames.com/>
3. <https://libgdx.badlogicgames.com/documentation.html>
4. <https://github.com/libgdx/libgdx/wiki/A-simple-game>
5. [https://github.com/libgdx/libgdx/wiki/Extending%20the%20Simple%20Game](https://github.com/libgdx/libgdx/wiki/Extending the Simple Game)