

Spine Animasyon Kütüphanesinin Oyun Yazılımı Geliştirmede Kullanımı Deneyimi

On the use of Spine Animation Library for the Game Software Development: An Experience Report

Muzaffer Tepekule
Masomo Oyun Teknolojileri A. Ş.
Tasarım Merkezi
İzmir, Türkiye
muzaffer.tepekule@masomo.com

Emre Şimşirli
Masomo Oyun Teknolojileri A. Ş.
Tasarım Merkezi
İzmir, Türkiye
emre.simsirli@masomo.com

Mehmet Ali Sakarya
Masomo Oyun Teknolojileri A. Ş.
Tasarım Merkezi
İzmir, Türkiye
mehmetali.sakarya@masomo.com

Rudi Diriman
Masomo Oyun Teknolojileri A. Ş.
Tasarım Merkezi
İzmir, Türkiye
rudi.diriman@masomo.com

Geylani Kardeş
Ege Üniversitesi Uluslararası
Bilgisayar Enstitüsü
İzmir, Türkiye
geylani.kardas@ege.edu.tr

Özet—Bu deneyim bildirisinde Spine animasyon kütüphanesinin endüstriyel mobil oyun yazılımlarının geliştirilmesi sırasında kullanılması değerlendirilmiştir. Spine ve desteklediği yazılım araçlarının kullanımının özellikle aynı animasyonun birden fazla karakterde görülmesini ve hızlıca uygulanmasını sağladığı, geliştirme ekibindeki animatörlerin, oyun geliştiricilerine yapılmak istenen animasyon kurgusunu çok kısa sürelerde aktarabilmesine imkan verdiği ve gerekli yazılımların üretimini yaklaşık 7 katına çıkardığı belirlenmiştir. Ancak bu kütüphanenin oyundaki ışık olayları için bazı durumlarda yetersiz kaldığı ve takip eden iki animasyonun beraber görüntülenmesinde sorunlara neden olduğu görülmüştür.

Anahtar Kelimeler—oyun yazılımı, oyun yazılımı geliştirme süreci, Spine animasyon kütüphanesi

Abstract—This paper evaluates the use of Spine animation library during industrial mobile game software development. The evaluation results showed that the use of both Spine and its supporting tools provides the rapid application of the same animation to various characters, shortens the transfer of animation constructs between animators and game developers and increases the production of required software approximately seven times. However, we also observed that the library was incapable of properly reflecting some of the light events inside games and caused problems in viewing the two animations together.

Keywords—game software, game software development process, Spine animation library

I. GİRİŞ

Bilgisayar oyunu yazılımlarının geliştirilmesi süreci bilgisayar bilimleri, sanat, medya tasarımı vb. bir çok disiplinden gelen kişilerin oluşturduğu geniş ekiplerin birlikte çalışmasına dayanan ve simülasyon, grafik, yapay zeka, fizik, ses programlama ve çevrimiçi çok oyunculu oyunlar için ağ ve veri tabanı programlama bilgisi gerektiren karmaşık faaliyetleri içermektedir [1]. Bir oyunun oluşturulmasında

genellikle konsept, plan, tasarım, geliştirme ve test adımlarından oluşan bir süreç izlenmektedir [2].

Mobil oyun sektörünün çok rekabetçi olmasından dolayı özellikle oyun firmaları oyun içerisinde bulunan karakterlerin özelliklerini ön plana çıkarmak adına oyuna daha fazla animasyona sahip karakterleri eklemeye çalışmaktadır. Karakterlere hareket kazandırarak animasyon algısını oluşturmak için C++, vb. programlama dillerinin kullanılmasını içeren geleneksel yaklaşım giderek yerini bu ihtiyaçlara cevap veren çeşitli yazılım kütüphanelerinin ve araçlarının sistematik kullanılmasına bırakmaktadır. Söz konusu bu kütüphanelerden biri mobil oyunlarda animasyonların oluşturulmasında giderek popülerleşen Spine'dır [3]. Her ne kadar Spine'in kullanım örnekleri bulunsada bu kütüphanenin ticari oyunların geliştirilmesi ihtiyacını ne ölçüde karşılayabildiğini göz önüne alan ve kullanımını değerlendiren bir çalışmaya rastlanmamıştır. Bu eksiklikten hareketle bu bildiriye, Spine animasyon kütüphanesinin Masomo firmasının günümüzde tüm mobil platformlarda en çok indirilen ve oynanan oyunlarından biri olan Kafa Topu oyununun animasyonlarının üretilmesi sırasında kullanılmasından elde edilen deneyimler aktarılmış; kullanımının avantaj ve dezavantajları gösterilmiştir.

II. SPINE ANİMASYON KÜTÜPHANESİ

Spine, mobil oyun dünyasında özellikle iki boyutlu (2D) oyunların ve animasyonların oluşturulmasında kullanılan bir yazılım aracı ve kütüphanesidir [3]. Spine, hem bir editör kullanarak animasyonlar oluşturmayı hem de Spine Runtime kütüphanelerini kullanan oyunlarda bu animasyonlardan faydalanmak için verimli bir iş akışını sunmaktadır. Spine ile bir animasyon geliştirmek için oyun karakteri kemiklerine görüntüleri bağlamak ve kemikleri canlandırmak (animate) yeterlidir —ki bu yaklaşım bilindik *frame-by-frame* animasyonlara göre özellikle karakterlerin farklı bileşenlerle giydirilmesi ve farklı efektler verilebilmesi, animasyonların harmanlanabilmesi (aynı karakter için yürümeden koşmaya ya

da yüzmeye pürüzsüz hareket geçişinin sağlanması) ve prosedürel animasyon (kemiklerin kodla manüpile edilebilmesi) açısından daha etkindir. Spine şu an 19 oyun araştırmacı ve 7 farklı programlama dilini desteklemektedir [4].

Oyun karakterlerinin oluşturulması sürecinde Spine ile canlandırma [5] tüm kemik hareketlerinin sürekliliğini sağlamaktadır ve bir nesne zıplarken, omurga her zaman harekete uymaktadır. Böylece karakterler gerçekmiş gibi canlandırılabilir. Karakter saha/zemin eğiminde yukarı çıktığında, vücut dönüşü otomatik olarak ona uymaktadır. Dört farklı animasyon yazılımı kütüphanesi için gerçekleştirilen bir karşılaştırmada [6], Spine iskelet animasyonları desteği, farklı oyun motorlarına aktarılabilme, görüntü deformasyonları, ters kinematik ve çizim araçları sunabilme kriterleri göz önünde bulundurularak en uygun yazılım kütüphanesi seçilmiştir. Ancak Spine'in büyük çaplı endüstriyel oyun yazılımlarının geliştirilmesi sırasında kullanılması sonucunda elde edilen deneyimlerin aktarıldığı bir çalışmaya bu bildiri yazıldığı tarihte ulaşılammıştır.

III. OYUN GELİŞTİRME SÜRECİ

Aynı zamanda T.C. Sanayi ve Teknoloji Bakanlığı tarafından onaylı bir tasarım merkezi olan Masomo'da bir oyun geliştirilmeden önce, oyunun genel yapısı hakkında ekip genel bir bilgi paylaşımı gerçekleştirir. Oyun tasarımcısı tarafından oyunun tasarım dokümanı hazırlanır. Oyun içerisinde bulunan bütün özellikler/bölmeler *Sketch* yapısında hem geliştirici ekiple hem de tasarımcı ekibi ile paylaşılır. İlgili dokümanda oyunun ana mekanı, sistemler arası etkileşim, oyun oynama ekranı ve oyun içerisinde olacak her bölümü anlatan detaylı açıklamalar bulunur. Ayrıca oyunda bulunan her bölümün hangi amaçla eklendiğini ve oyunun genel yapısı için önemi anlatılır. *Sketch*'te oyun içerisinde ön plana çıkacak alanlar belirlenir. Bu dokümanın paylaşılmasının ardından karakterler ve animasyonlar karşılıklı olarak belirlenip bunlar için gerekli olan ihtiyaçlar ortaya çıkarılır. *Art* ekibi belirlenen ihtiyaçlar doğrultusunda gerekli animasyonların planlamasını yapar. Animasyonlar zıplama, koşma, sandık açma vb. adlandırılarak tasarlanan oyunun animasyon kütüphanesine yerleştirilir. Bundan sonra önyüze aktarılması için geliştirici ekip kütüphane içerisinde bulunan dosyaları kullanır. Gerekli senaryolarda harekete geçecek/tetiklenecek animasyonlar kütüphaneden alınarak ön yüze yerleştirilir ve hareketin/animasyonun ilgili senaryo gerçekleştiği zaman hareketlenmesi sağlanır. Daha önce firmadaki oyun geliştirme projelerinde bu animasyonların gerçekleştirimi için sadece C++ kodlama yapılırken bu çalışmada Spine animasyon aracı kullanılmıştır. Şekil 1'de bu oyun geliştirme süreci akışı gösterilmektedir. Burada oyun yazılımı geliştirme sürecinde, tasarım ve kodlama her süreç adımında yer aldığından burada ayrı bir şekilde belirtilmemiştir. Bu geliştirme adımlarının beraber hareket etmesi gerekmektedir. Anahtar Performans Göstergeleri (KPI'lar) her oyun için değişiklik gösterebilir. Yaygın kullanılan KPI'lara ilk gün oyunu indiren kişilerin kaçının tekrar oyuna giriş yapma isteği duyması ("Day 1 Retention") veya 7 gün sonra kaçının tekrar girmek istemesi ("Day7 Retention") örnek verilebilir fakat oyun çeşidi ve kurgusuna göre farklı KPI'lar oyun üreticileri tarafından belirlenebilir.

Bu çalışmada Masomo'nun tüm dünyada en çok oynanan mobil oyunu olan Kafa Topu'nun yeni versiyonunun (Kafa Topu 2) (<https://linktr.ee/KafaTopu2>) içerdiği animasyonların

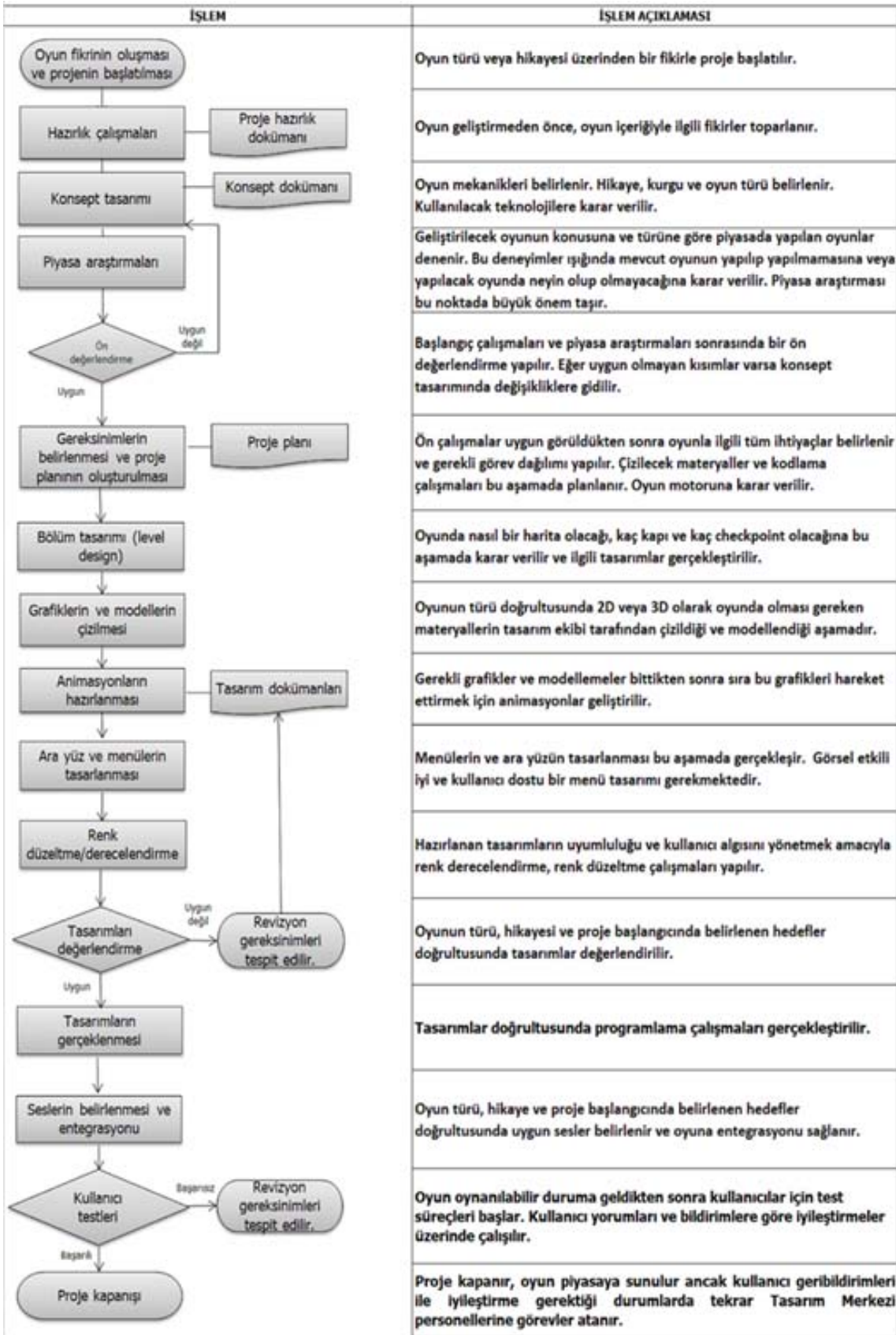
geliştirilmesi göz önüne alınmıştır. Dünyada halen 100 Milyon'un üzerinde indirilen ve günlük 1.5 Milyon aktif oyuncunun kullandığı bu mobil oyun Türkiye'nin dijital oyun ihracatının çok önemli bir yüzdesini oluşturmaktadır.

Mobil cihazlar üzerinden kullanıcıların birbirlerine gol atmaya çalıştıkları bu oyun için farklı animasyonlar bir hareket olarak birleştirilmekte ve kütüphane çeşitlendirilmektedir. Şekil 2'de bu animasyonlardan birer ekran görüntüsü verilmiştir. Animasyonların tam haline [7]'den erişilebilir. Özellikle Spine'in "Skin" özelliği kullanılarak her karakterin aynı animasyonlarla hareket ettirilmesi ve böylece hazırlanan animasyonları farklı karakterlere eşleştirilmesinin daha kolay ve hızlı gerçekleştirilmesi hedeflenmiştir.

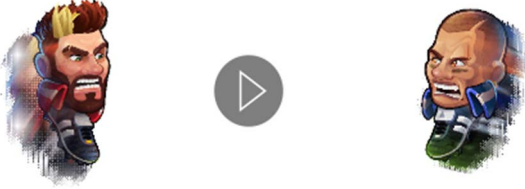
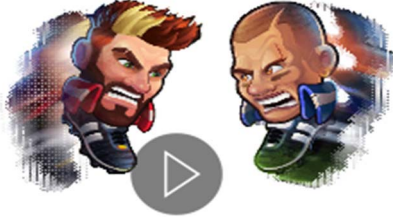
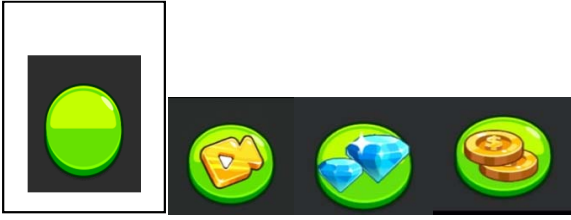
Spine kullanarak hazırlanan animasyonların Kafa Topu oyununun geliştirme süreçlerine eklenmesine örnek olarak zıplama animasyonunun önceliklendirilmesi ve farklı varyasyonlarının üretilmesi ve testi verilebilir. Bunun için oyunun farklı zıplama varyasyonlu ve tek zıplama animasyonlu örnekleri üretilerek hedef marketlerde oyuncular tarafından oynanmasını sağlanmaktadır. Test edilen örneklerde iki versiyon arasında çok büyük bir fark yoksa burada zıplama animasyonunu daha fazla geliştirmek için kaynak harcamaya gerek kalmamaktadır. Eğer zıplama animasyonlarının çeşitliliği oyunda daha fazla süre geçirmesi ve daha fazla kullanıcının bir sonraki gün oyunu açmasını tetikliyorsa zıplama animasyon çeşitliliği artırılmaktadır. Bu süreç animasyon özelinde farklı şekillerde çeşitlendirilerek özellikle hangi animasyonlara daha fazla kaynağın harcanması gerektiği belirlenmekte ve animasyon kütüphanesi yine Spine kullanılarak geliştirilmektedir. Sonrasında bir önceki bölümde anlatılan şekilde animasyonun oyuna entegrasyonu sağlanmaktadır.

IV. DEĞERLENDİRME

Hayata geçirilmesi planlanan oyunun animasyon açısından zengin olması hedeflendiğinden bunu sağlayacak alternatif animasyon yazılımı kütüphaneleri bu çalışmada incelenmiştir ve [6]'da da kullanılan iskelet animasyonu, farklı oyun motorlarını destek ve çizim aracı sunabilme kriterleri gözetilerek gerçekleştirilen eleme sonucunda ilk olarak Toon Boom kütüphanesi denenmiştir. Toon Boom'da karakterlerin ve objelerin parçalarının ayrı ayrı hareket ettirilmesi tekniği ile animasyon programları hazırlanabilmektedir. Ancak daha çok çizgi film animasyonları için kullanılan bir yazılım olan Toon Boom'un mobil oyunlar için çok önemli bir unsur olan "hız" açısından Spine'a kıyasla daha kötü performans gösterdiği belirlenmiştir. Ayrıca gerçekleştirilen yazılım testi sürecinde özellikle animasyon için hazırlanan Json dosyasının Toon Boom kullanıldığında doğrudan oyuna aktarılamadığı tespit edilmiştir. Bunu sağlamak için farklı bir eklentinin geliştirilmesi gerekmektedir. Üstelik oyun yazılımı geliştirme ekibi bu eklenti geliştiriminin yazılım istekleri açısından sağlıklı ve pratik bir çözüm olabileceğinden emin olamamıştır. Belirlenen tüm bu eksiklikler nedeniyle Toon Boom yerine Spine'in kullanılması tercih edilmiş ve uygulamada Spine kütüphanesi kullanılmıştır. Spine'a dayalı geliştiriminin bu çalışmada tespit edilen avantajlar ve dezavantajları aşağıda listelenmiştir:



Şekil 1. Masomo oyun yazılımı geliştirme süreci.



Şekil 2. Geliştirilen oyunu animasyonlarına ait örnek ekran görüntüleri.

A. Spine Kullanımının Sağladığı Avantajlar

- Spine animasyon programı ile tek bir JPEG biçimindeki görüntü dosyası üzerinde bir animasyona çeşitli pozlar, *mesh* sayesinde konumlandırılabilmiştir. Bu sayede oyun içerisinde oluşturulan toplam dosya boyutunun küçüldüğü gözlenmiştir. Bu özellik gerçekten mobil oyun geliştiricileri için önem arz etmektedir.
- Hazırlanan her bir animasyonun son haline yakın versiyonları yazılıma aktarılmadan önce Spine içerisinde ön izleme ile görülebilmekte ve test edilebilmektedir. Böylece animasyon oluşturma süreci hızlanmıştır.
- JPEG formatında oluşturulan görüntü dosyalarının renk değişimi Spine programı ile hızlıca gerçekleştirilebilmiştir.

- Bir animasyon sırasında gerçekleşmesi istenen olayların tetiklenmesini (örneğin bir oyun karakterinin yere düşmesinde ses çıkması gibi) mümkün hale getiren “Spine Event” mekanizmasının ekip içerisinde animatörlerin, oyun geliştiricilerine yapılmak istenen animasyon kurgusunu planlı bir şekilde ve çok kısa sürelerde aktarabilmesini sağladığı gözlenmiştir.
- Animatörlerin bir dijital modeli hareket ettirebilmeyi mümkün kılan kontrolleri içeren üç boyutlu *Rigging* hareketlerini iki boyutlu programa aktarmaları Spine kullanımı ile çok kolaylaşmıştır.
- *Spine Skin*’ler aynı animasyonun birden fazla karakterde görülmesini ve hızlıca uygulanmasını sağlamıştır. Bu özellik de daha fazla sayıda ve çeşitte karakterlerin üretilmesine imkan vermiştir.
- Yukarıda listelenen nitel değerlendirme sonuçlarına ek olarak bu çalışmada Spine kullanımının oyun yazılımı projelerindeki nicel kazancı da geliştirilen animasyon adedi üzerinden belirlenmiştir. Aynı oyun için daha önce 6 aylık bir süre içerisinde sadece C++ kodlama ile yaklaşık 10 adet farklı animasyon hazırlanabilirken Spine kullanımı ile aynı çalışma süresinde yaklaşık 70 farklı animasyon üretilebilmiştir. Oyun yazılımı geliştirmede elde edilen hızın yanı sıra üretim miktarı açısından da Spine kullanımı verim sağlamıştır. Ancak bu üretim miktarındaki artış gerçekleştirilen durum çalışmasının kapsamına ve süresine özeldir. Oyun yazılımı ihtiyaçlarının değişmesi durumunda mevcut ölçülen animasyon üretme adedi değişkenlik gösterebilir. Ayrıca yine geliştirilecek oyuna bağlı olarak tasarımcıların animasyon üretme süreleri farklı olabilir. Bu sürenin uzaması durumunda bile yine C++ kodlama ile animasyon üretmeye kıyasla Spine’in kullanımının bu durum çalışmasında ölçülen yaklaşık 7 kat üretim performansından çok daha fazlasını sağlayacağı tahminlenmektedir. Fakat bunun geçerliliğinin sağlanması için çoklu durum çalışmalarının gerçekleştirilmesine ihtiyaç vardır.

B. Spine Kullanımının Getirdiği Dezavantajlar

- Her ne kadar görüntü dosyalarının renk değişimi Spine ile sağlanabilse de aynı dosyanın beyaz versiyonunun oluşturulamadığı belirlenmiştir. Bunun için ekstradan çizim yapıp tekrardan Spine’a aktarmak gerekmiştir.
- İki animasyon arasındaki *mix*, bir *track* mantığında Spine ara yüzünde görülebilmiştir. Örnek vermek gerekirse 0. track aşamasında koşma ve 1. track aşamasında onun el animasyonunun *mix*’i görülsün bile daha sonra oynayacak olan *attack* animasyonu Spine arayüzünde beraber görülebilmiştir. Bunun için yazılım tarafında test için bütün animasyonlar tek bir görüntüye dönüştürülmüştür. Ancak bu, tekrar eden işlemlerde zaman kaybettirmiştir.
- Bir animasyon programında standart olarak oluşması gereken ışık olayı (*Ghosting*) Spine’da bulunmakla birlikte bazı durumlarda yetersiz kalabilmektedir. Örneğin sadece belirli görüntü dosyalarının ışık olayı görmek istenirken sahnede bulunan bütün görüntü dosyalarının ışık olaylarının görünmesi sorunu ile karşılaşmıştır. Bu durum uzun vadede zaman kaybına neden olmaktadır.

V. SONUÇ

Gerçekleştirilen çalışma sonucunda Spine animasyon kütüphanesinin dünyada en çok oynanan mobil oyunlardan biri olan Kafa Topu'nun yeni versiyonunun geliştirilmesi sırasında kullanılması değerlendirilmiştir. Spine kullanımının özellikle aynı animasyonun birden fazla karakterde görülmesini ve hızlıca uygulanmasını sağladığı, geliştirme ekibindeki animatörlerin, oyun geliştiricilerine yapılmak istenen animasyon kurgusunu çok kısa sürelerde aktarabilmesine imkan verdiği ve animasyon yazılımlarının üretimini gerçekleştirilen çalışma kapsamında yaklaşık 7 katına çıkardığı belirlenmiştir. Ancak bu kütüphanenin JPEG formatında oluşturulan görüntü dosyalarındaki renk değişimleri ve oyundaki ışık olayları için bazı durumlarda yetersiz kaldığı ve takip eden iki animasyonun beraber görüntülenmesinde sorunlara neden olduğu görülmüştür.

KAYNAKLAR

- [1] A.O. O'Hagan, G. Coleman, R.V. O'Connor, "Software Development Processes for Games: A Systematic Literature Review", *Communications in Computer and Information Science*, vol. 425, pp. 182-193, Springer, 2014, DOI: 10.1007/978-3-662-43896-1_16.
- [2] C.M. Kanode, H.M. Haddad, "Software Engineering Challenges in Game Development", In: 6th Int. Conf. Information Technology: New Generations, pp. 260-265, 2009, IEEE, DOI: 10.1109/ITNG.2009.74.
- [3] Esoteric Software, "Spine: 2D animation for games", <http://esotericsoftware.com/> (son erişim: Haziran 2020).
- [4] E.B. Hunter, "Building Video Game Adaptations of Dramatic and Literary Texts", *Research Methods for the Digital Humanities*. Palgrave Macmillan, 2018, DOI: 10.1007/978-3-319-96713-4_10.
- [5] K. Myller, "Use of Spine software in video game character creation", Karelia University of Applied Sciences, 2018.
- [6] J. Lehtonen, "From 2D-Sprite to Skeletal Animations - Boosting the performance of a mobile application", Turku University of Applied Sciences, Information Technology, 2016.
- [7] Spine ile geliştirilen örnek animasyonlar, https://docs.google.com/presentation/d/1QD2SqQcWVKH3B1qmliRRtF0njfBNgWCTeZ-xU7GOaew4/edit#slide=id.g8a6948fab0_2