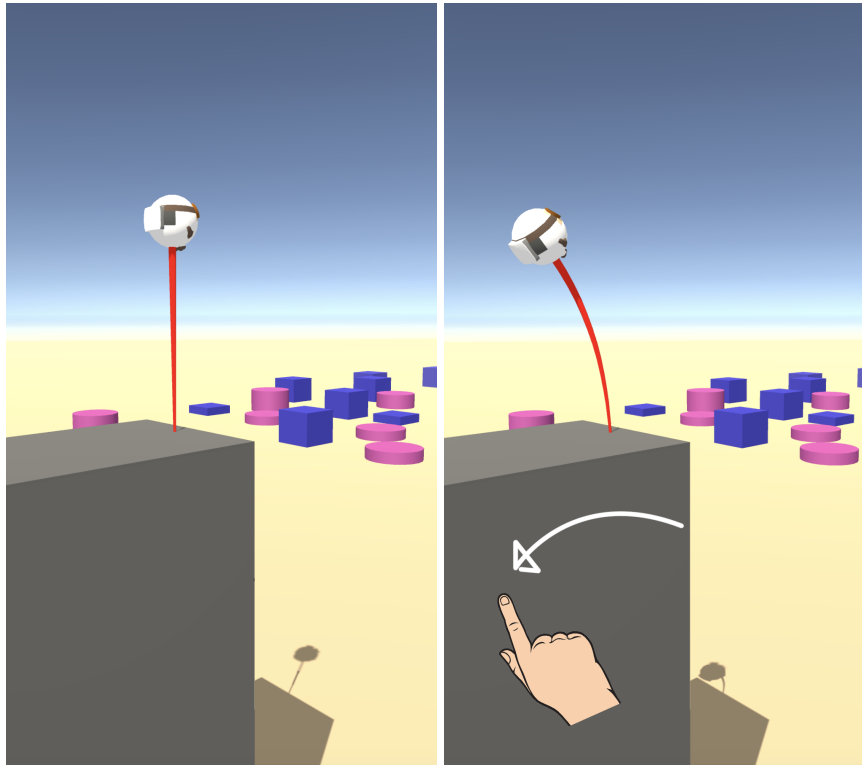


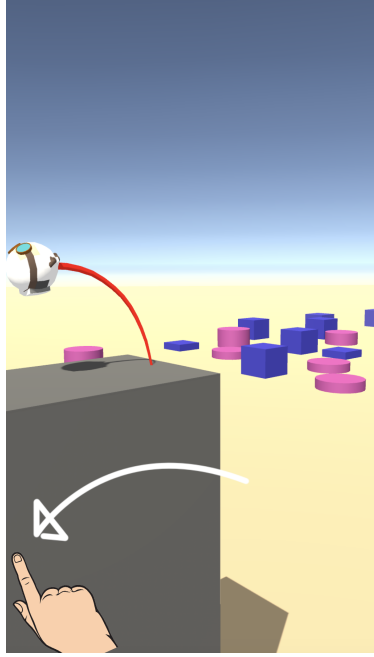
- Beklenen projenin kapsamı şu şekilde özetlenebilir:
 - Oyunumuzda verdiğimiz inputlara göre hareket eden topa benzer bir karakter bulunmaktadır. Karakter modeli ve animasyonları proje assetleri içerisinde sağlanacaktır.



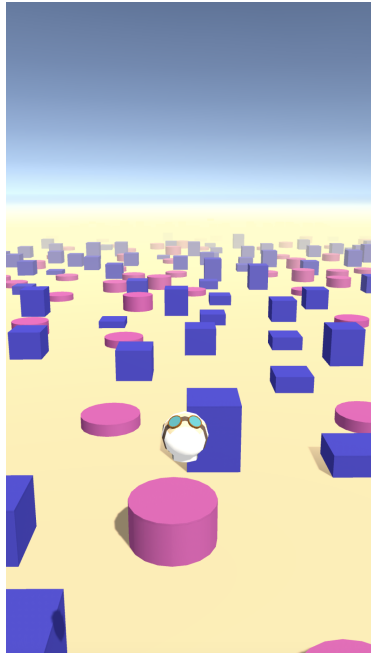
- Top, ilk olarak bir çubuğun üzerinde takılı haldedir. Kullanıcı, parmağıyla ekranın herhangi bir noktasında tutup sola doğru çekme hareketi yapar ve çubuğu top ile birlikte geriye doğru gerilmesini sağlar. Kamera bu sırada altta görüne benzer bir açıdadır. Çubuk modeli ve hareketleri de proje assetleri içerisinde bulunmaktadır.



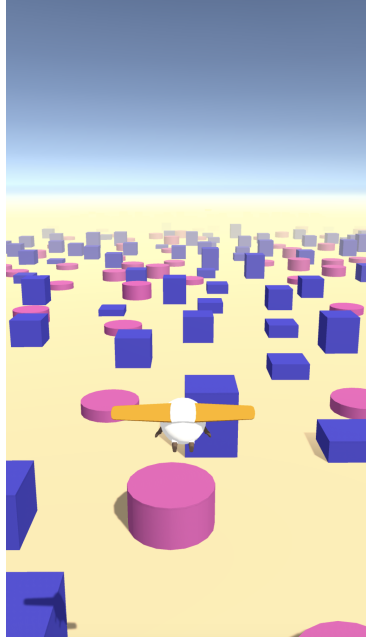
- Parmak bırakıldığında çubuk tekrar eski haline dönmek ister ve kazandığı enerji ile birlikte topu fırlatır. Benzer bir execution'ı Pokey Ball ([iOS](#) / [Android](#)) oyununda da görmek mümkün.
- Topun ileriye doğru fırlatılma kuvveti de birebir olarak bu çekme miktarıyla doğru orantılıdır. Yani eğer çubuğu maksimum seviyede geriye çekersek haliyle topa aktaracağı enerji de yüksek olacağı için çok daha uzun menzilli bir uçuş sağlanacaktır. Çekme miktarımız düşük olursa topun uçuş menzili kısılacaktır.



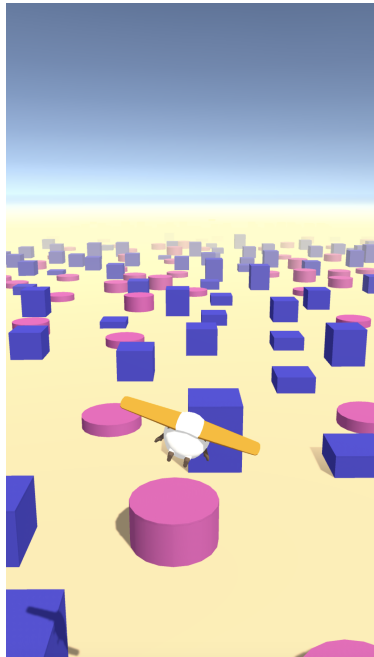
- Top, fırladıktan sonra ileriye doğru eğik atış hareketi yapar. Kamera, smooth bir şekilde topun arkasına geçer ve onu takip eder. **Top ileriye doğru atılırken aynı zamanda kendi etrafında da yüksek bir devirde rotate eder.**



- Kullanıcı, top uçarken herhangi bir t anında parmağını ekrana dokundurabilir. Bunun karşılığında karakterimiz animasyonik bir biçimde kanatlarını açar, rotate smooth bir şekilde kesilir ve süzölmeye başlar. Kullanıcının parmağı ekranda hold olduğı süre boyunca bu süzölme işlemi devam eder.



- Kullanıcı parmağı ekrandayken aynı zamanda süzölmenin yönünü de sağa sola slide ederek ayarlayabilir.



- Kullanıcı parmağını ekrandan çektiği an karakterimiz yine animasyonik bir biçimde kanatlarını kapatır ve kendi etrafında rotate olarak bulunduğu yerden yer çekiminin verdiği etkiyle aşağı düşmeye başlar.
 - Kanat açılma ve geri içeri kapanma animasyonları verilen unity package içerisinde sağlanacaktır. Aday animasyonların hızını kendi uygun bulduğu şekilde ayarlama özgürlüğüne sahiptir.
 - Kullanıcı topun süzülmesini sağlayarak veya onu serbest bırakarak yeryüzünde bulunan belirli şekillere inmeye çalışır.
 - Yeryüzünde 2 tip şekil bulunur. Birisi dikdörtgen prizma diğeri ise silindir. İki şeklin farklı renklerde olması birbirlerinden ayırt edilebilmesi açısından önerilir.
 - Top, eğer dikdörtgen prizmanın üst yüzeyine dokunursa X birim havaya zıplar ve tekrar yükseldiğinden kullanıcıya daha başka bir noktaya gidebilmesi için şans tanır. Top, eğer silindirin üst yüzüne dokunursa bu sefer 2X birim sıçrar. Sıçrama sırasında kanatlar kapalı ve tekrar kendi etrafında rotate olur pozisyona geçilir.
 - Prizma ve silindirler aralarda boşluklar bırakılarak rastgele noktalara konumlandırılabilir. Hatta her oyun başlangıcında şekiller, bir önceki gameplay'den farklı şekilde diziliyor da olabilir.
 - Oyuna biraz çeşitlilik katmak amacıyla her bir yerleştirilen şekil (tipinin ne olduğundan bağımsız bir şekilde) farklı yüksekliklerde bulunabilir.
 - Adaydan bu noktada herhangi bir level design beklenmemektedir. Şekillerin yüzey alanları, birbirlerinden ne kadar uzaklıkta olacakları, hangi sıklıkla silindir veya prizma kullanılacağı, hangisinin ne kadar yükseklikte olduğu, topun değmesi halinde ne kadar yükseğe tekrar havalanabilecekleri gibi tüm parametreler adayın insiyatifine bırakılmıştır.
 - Şu detay akılda bulundurulmalıdır ki kullanıcı sürekli olarak şekillerin üzerine inebilir ve sonsuza dek bu süzülme zıplama hareketini devam ettirebilir.
 - Top çubuktan fırladıktan sonraki uçuş, kendi etrafındaki rotate hareketi, yapılabilecek bütün kontroller ve topu tekrar tekrar yükselten platformlar, Sky Glider ([iOS](#) / [Android](#)) oyunu referans alınarak yapılabilir. Temelde birebir aynı execution beklenmektedir ancak eğer aday daha iyisini yapabileceğini düşünüyorsa bu konuda herhangi bir kısıt bulunmamaktadır.
 - Topun yere düştüğü tüm durumlarda başarısız olunur ve kullanıcı oyunu basit bir buton yardımıyla yeniden başlatabilir.
- Oyun, editor üzerinde ve mobil cihazlarda (iOS / Android) oynanabilir durumda olmalıdır ve 60FPS alınabilmesi gerekmektedir. Düşük donanımlı android cihazlarda da teste tabi tutulur ve FPS durumu kontrol edilir. Ekranın bir köşesinde FPS bilgisinin verilmesi test eden kişiye kolaylık sağlayacaktır. Gerekli olmadığı düşünülse dahi yapılacak olan tüm performans optimizasyonları adaya artı puan olarak yansır.
 - Oyun hissiyatını belli bir düzeye çıkarabilmek için asgari ölçüde juicy kullanılması tavsiye edilir. Oyunu daha eğlenceli hale getirecek olan her bir juicy faktörü, adaya artı puan sağlayacaktır.
 - Grafik tasarım bakımından herhangi bir beklenti bulunmamaktadır. Sahne içerisindeki elementlerin birbirlerinden ayırt edilebilmesine olanak sağlayacak şekilde farklı renklerle giydirilmesi yeterlidir.

- Projenin aktarımı git üzerinden yapılır. Projenin başından sonu kadar düzenli commitler halinde anlamlı mesajlarla gönderilmesi beklenir. Adaya herhangi bir git reposu sağlanmaz bu yüzden kendi açtığı herhangi bir repoda info@apps.com.tr mailini, **proje ilk açıldığı andan itibaren** yetkilendirmiş olması yeterli olacaktır.
- Proje tamamlandığında yetkili kişiye mail ile bildiri yapılır. Mail'e oyunun son halini oynatabilen bir .apk dosyasının da eklenmiş olması önerilir.
- Sonuç olarak teslim edilen oyun,
 - Performans
 - Kontrollerin akıcılığı
 - Animasyonlar arası geçişler
 - Hareketlerin ne kadar gerçekçi olduğu
 - Deneyimlenen oyun hissiyatı / ne kadar juicy olduğu
 - Yapılan optimizasyonlar
 - Yazılan kodun temizliği ve bug-free olması
 - Git kullanımı ve commit mesajları
 - Belli başlı bir kodlama mimarisinin takip edilip edilmediği

gibi kıstaslara göre değerlendirilir.