

---

**sim1**

**iin**

# **Yazılım Gereksinim Belgesi**

**Sürüm 1.0**

**Karadeniz Teknik Üniversitesinden**

**NGAsim grubu üyeleri Nurullah Öztürk 365233, Göktuğ Ersöz  
383247, Ahmet Kutsi Yılmaz 365297 tarafından hazırlanmıştır**

**17 Kasım 2020**

# İçindekiler

<b>İÇİNDEKİLER .....</b>	<b>ii</b>
<b>REVİZYON GEÇMİŞİ.....</b>	<b>iii</b>
<b>1. GİRİŞ.....</b>	<b>1</b>
1.1 AMAÇ.....	1
1.2 KAPSAM.....	1
1.3 TANIMLAR VE KISALTMALAR.....	1
1.4 REFERANSLAR.....	2
<b>2. GENEL TANIM .....</b>	<b>2</b>
2.1 ÜRÜNE BAKIŞ.....	2
2.1.1 Kullanıcı Arayüzleri .....	3
2.1.2 Donanım Arayüzleri .....	4
2.1.3 Yazılım Arayüzleri .....	4
2.1.4 İletişim Arayüzleri .....	5
2.2 ÜRÜN İŞLEVLERİ .....	5
2.3 KULLANICI ÖZELLİKLERİ .....	6
2.4 KISITLAR.....	6
2.5 VARSAYIMLAR VE BAĞIMLILIKLAR .....	6
<b>3. ÖZEL GEREKSİNİMLER.....</b>	<b>7</b>
3.1 İŞLEVSEL GEREKSİNİMLER.....	7
3.2 SİSTEM ÖZELLİKLERİ .....	9
3.2.1 Hareket Penceresi .....	9
3.2.2 Girdi/Çıktı Arayüzü .....	9
3.2.3 Bilgi Penceresi.....	10
3.2.4 Simülasyon Penceresi .....	10
3.3 PERFORMANS GEREKSİNİMLERİ.....	11
3.4 TASARIM KISITLARI.....	11
3.5 KALİTE ÖZELLİKLERİ .....	11
3.5.1 Güvenilirlik.....	11
3.5.2 Erişilebilirlik.....	11
3.5.3 Güvenlik .....	11
3.5.4 Sürdürülebilirlik.....	12
3.5.5 Taşınabilirlik.....	12
3.5.6 Kullanılabilirlik .....	12
4. GELECEKTE YAPILMASI PLANLANANLAR .....	12
<b>EKA .....</b>	<b>14</b>

## Revizyon Geçmiři

İsim	Tarih	Deęiřiklik	Sürüm

# 1. Giriş

## 1.1 Amaç

Bu belgenin amacı “sim1” yazılımı gereksinimlerini ortaya koymaktır. Yazılımın amaç ve özellikleri arayüzleri ve ne yapması gerektiği ve hangi şartlar altında işlem göreceği belirtilecektir. Bu belge, yazılımın kullanıcıları ve potansiyel geliştiriciler için hazırlanmıştır.

## 1.2 Kapsam

“sim1” yazılımı bir Atış hareketi simülasyonudur. Tek boyutta ve İki boyuttaki Atış hareketlerini(Serbest düşme, Yukarıdan aşağı düşey atış hareketi, Aşağıdan yukarı düşey atış hareketi, Yatay atış hareketi, Eğik atış, Pike atışı) dinamik bir şekilde simüle edebilir. Atış simülasyonu yazılımı çeşitli eğitim kurumları tarafından eğitim amaçlı kullanılabilir. Öğrencilerin yaygın bir Fiziksel olay olan Atış hareketlerini daha iyi anlayabilmelerine yardımcı olur.

## 1.3 Tanımlar ve Kısaltmalar

Atış hareketi, Dünya yüzeyine yakın yerlerde; düşen, fırlatılan cisimlerin yaptığı harekettir. Bu harekette cismin ivmesi sabittir ve yerçekimi ivmesine eşittir.

Serbest düşme, Yukarıdan serbest bırakılan cismin(yatay ve düşeydeki ilk hız sıfırdır) yaptığı harekettir.

Yukarıdan aşağı düşey atış hareketi, yukarıdan aşağıya doğru bir ilk hızla atılan cismin yaptığı harekettir. Yatay yöndeki hız sıfırdır.

Aşağıdan yukarı düşey atış hareketi, aşağıdan yukarıya doğru bir ilk hızla atılan cismin yaptığı harekettir. Yatay yöndeki hız sıfırdır.

Yatay atış hareketi, Yukarıdan yatay yönde bir ilk hızla atılan cismin yaptığı harekettir. Düşey yöndeki ilk hız sıfırdır.

Eğik atış yatay düzlemle açı yapacak şekilde atılan bir cismin hareketidir. Düşey yönde ve Yatay yönde bir ilk hız vardır.

Pike atışı yukarıdan yatay ve düşey aşağı yönde bir ilk hızla atılan cismin yaptığı harekettir.

SDL (Simple Directmedia Layer): Cross-platform bir yazılım geliştirme kütüphanesidir. Ses, klavye, joystick, grafik kartı gibi donanımlara düşük seviyeli erişim sağlar.

## 1.4 Referanslar

Belge standardı IEEE 830-1998:

<http://www.math.uaa.alaska.edu/~afkjm/cs401/IEEE830.pdf>

Desteklenen işletim sistemleri:(2)

<https://wiki.libsdl.org/Installation>

sdl2 kütüphanesi ana sayfa(1):

<https://wiki.libsdl.org/>

## 2. Genel Tanım

### 2.1 Ürüne Bakış

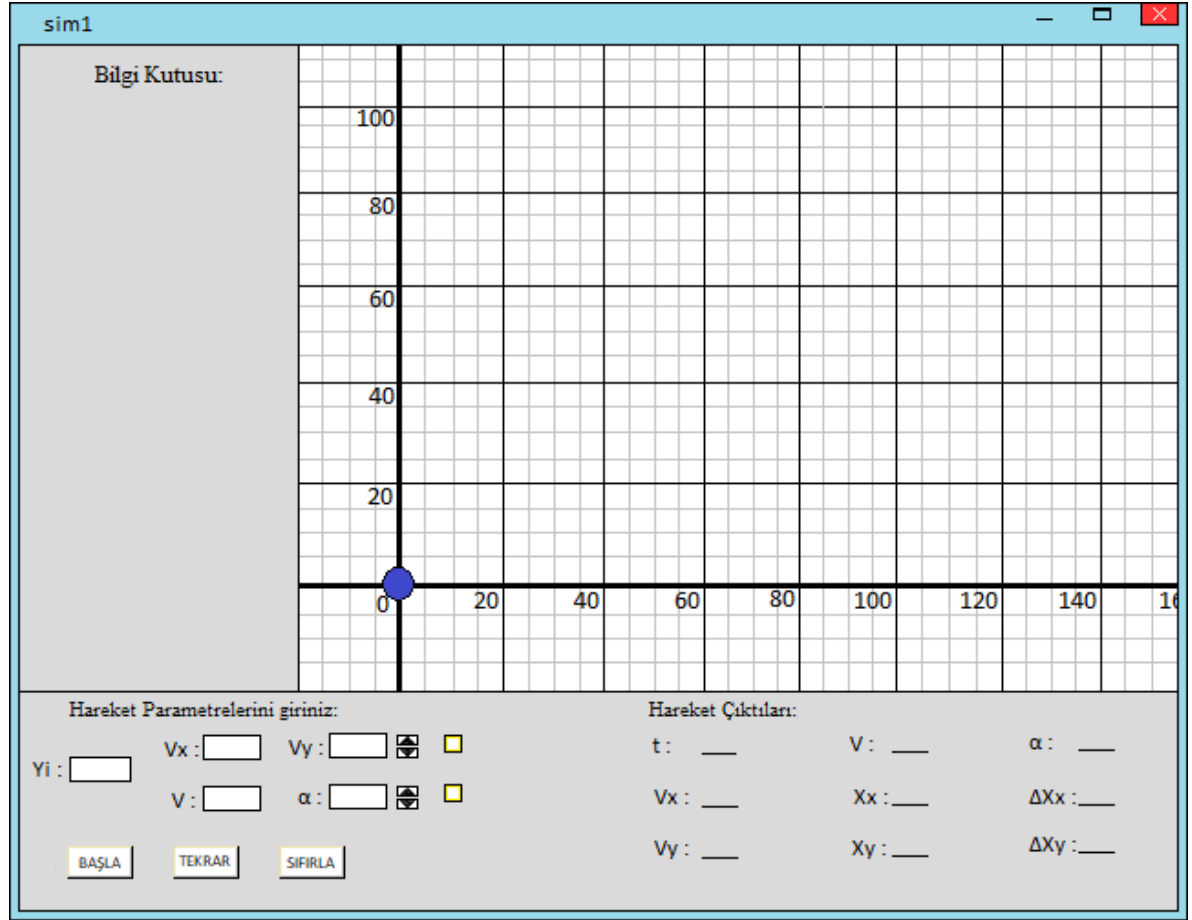
“sim1” yazılımı atış hareketlerini daha iyi anlayabilmek için deneyler yapmak isteyen veya bunu öğretim amaçlı kullanmak isteyen veya sadece tecrübe edinmek isteyen herkes için geliştirildi. Ürün çeşitli hareket olaylarının bütün çıktılarını hareket esnasında göstererek canlı bir deneyim sunar.

“sim1” yazılımı bağımsızdır. Herhangi daha büyük bir sistemin parçası değildir. Tamamen müstakildir. x86 temelli bilgisayarlar ile windows 10 işletim sistemi üzerinde C++ dili kullanılarak Windows başta olmak üzere linux ve macOS işletim sistemlerinde çalıştırılmak üzere geliştirilmiştir.

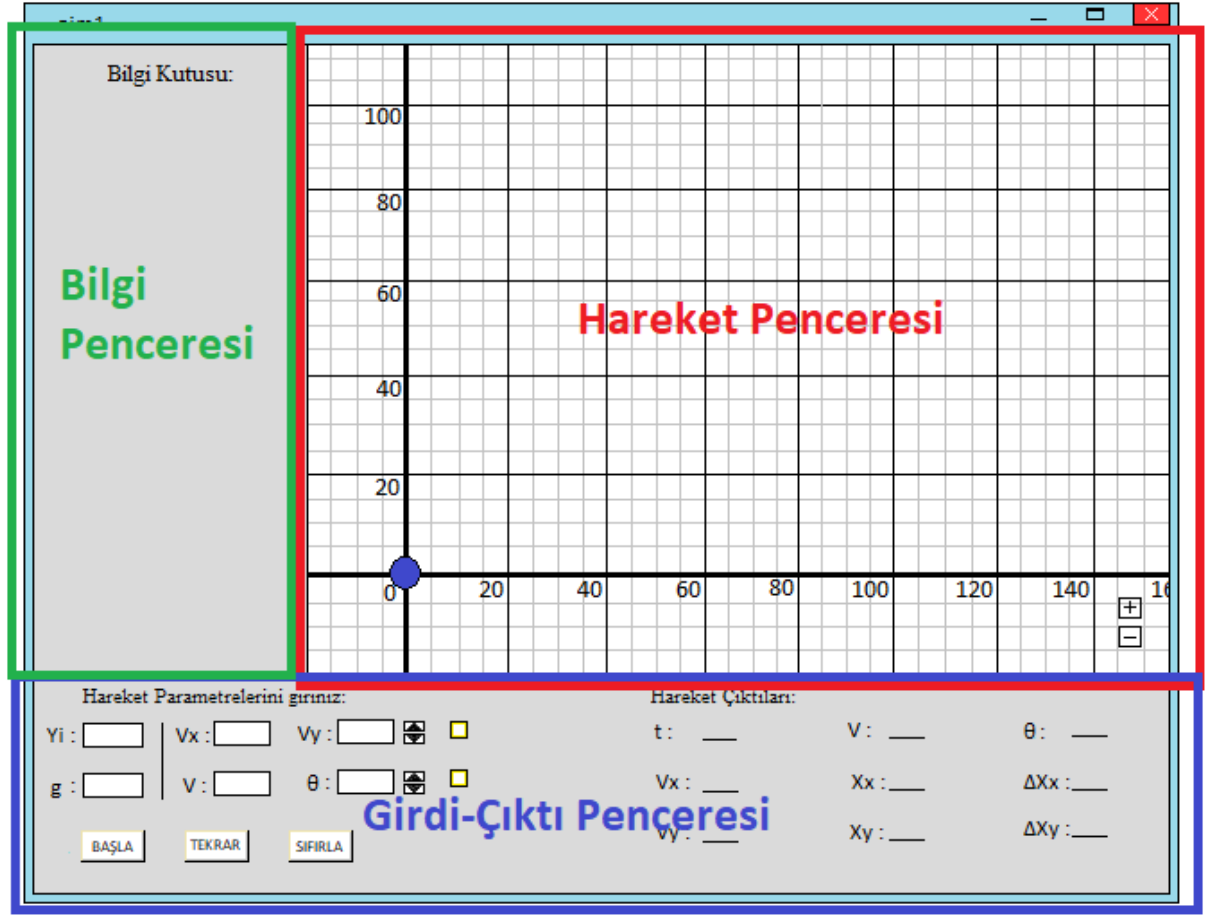
Ürün 2D grafikleri oluşturabilmek ve kullanıcı donanımı ile etkileşime geçebilmek için sdl2 kütüphanesi(1) kullanır.

## 2.1.1 Kullanıcı Arayüzleri

- Simülasyon Penceresi:



- Hareket Penceresi(sağ üstteki dörtgen), Bilgi Penceresi(sol üstteki dörtgen) Girdi-Çıktı Penceresi(alttaki dörtgen), Cisim(Hareket penceresindeki koordinat düzleminin merkezindeki mavi daire)



### 2.1.2 Donanım Arayüzleri

Ürün çalıştırıldığı sistemin varsayılan ekranı, faresi, ve klavyesi ile etkileşime geçer. (bakınız: 4.bölüm)

### 2.1.3 Yazılım Arayüzleri

Programın ihtiyaç duyduğu SDL2 kütüphanesine ait bütün dll dosyalarının ve image'larının yanında .exe dosyasının da bulunması gerekmektedir. Ayrıca İşletim Sistemi sdl2 kütüphanesi kullanımını desteklemelidir. Ürünün herhangi bir veritabanı sistemi ile ilişkisi yoktur. Herhangi bir Web sunucusuyla ilişkisi yoktur. Kendi başına her yerde çevrimdışı kullanılabilir. Bir işletim sistemi üzerinde çalışır. Desteklenen tüm işletim sistemleri linki referanslarda verilmiştir.(2)

## 2.1.4 İletişim Arayüzleri

Ürüne internet üzerinden güncelleme yapılmamaktadır dolayısıyla internet bağlantısına ihtiyacı yoktur. Ürünün herhangi bir sistemle iletişim kurmasına gerek yoktur.

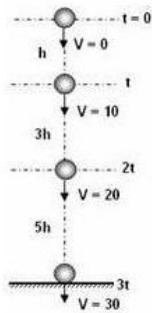
## 2.2 Ürün işlevleri

Ürün 6 çeşit hareketi simüle edebilmektedir. Bunlar; Serbest Atış Hareketi , Yukarıdan Aşağıya Düşey Atış Hareketi, Aşağıdan Yukarıya Düşey Atış Hareketi, Yatay Atış Hareketi, Eğik Atış Hareketi ve Pike Atışı'dır.

Bu hareketlerin her biri sisteme belirli girdi değerlerinin verilmesiyle elde edilir. Sisteme hangi girdiler verilirse verilsin mutlaka bu hareket sınıflarının birine dahildir.

Aşağıda Atış hareketleri ve formülleri verilmiştir. Yatay Atış, Eğik Atış ve Pike Atışı diğer hareketlerde kullanılan formüllerin kullanımı ile hesaplanabilir.

### SERBEST DÜŞME



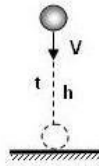
$$V_0 = 0$$

$$h = \frac{1}{2} \cdot g \cdot t^2$$

$$V_y = g \cdot t$$

$$V_y^2 = 2 \cdot g \cdot h$$

### YUKARIDAN AŞAĞI DÜŞEY ATIŞ HAREKETİ

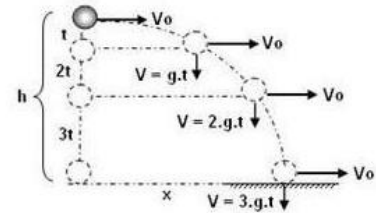


$$h = V_0 \cdot t + \frac{1}{2} \cdot g \cdot t^2$$

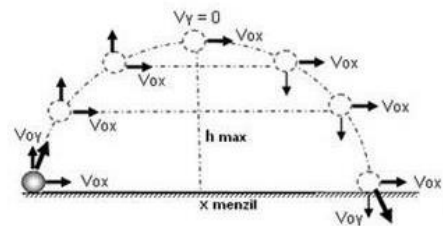
$$V_y = V_0 + g \cdot t$$

$$V_y^2 = V_0^2 + 2 \cdot g \cdot h$$

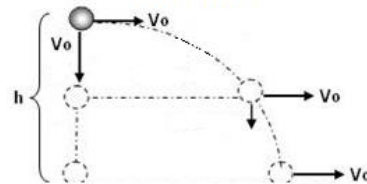
### YATAY ATIŞ HAREKETİ



### EĞİK ATIŞ



### PİKE ATIŞI



Figür 2.3: Atış hareketleri ve hesaplamada kullanılan formüller



## 2.3 Kullanıcı Özellikleri

Bu uygulama üç sınıf kullanıcıyı hedeflemektedir:

Genel kullanıcılar atış hareketlerini canlı olarak gözlemlemek isteyen öğrenciler, öğretimi kolaylaştırmak için kullanmak isteyen öğretmenler ve uygulamaya ilgi duyup deneyimlemek isteyen herkes.

Geliştiriciler uygulamayı daha iyi ve kullanışlı hale getirmek, hataları bulup düzeltmek ve yeni işlevler eklemek isteyen geliştiriciler bu durumda NGAsim üyeleri.

Yazılımı ve/veya dokümantasyonu incelemek isteyen gözlemciler, geliştiriciler veya aday geliştiriciler.

## 2.4 Kısıtlar

Ürünün çalıştırılacağı işletim sistemi SDL'yi desteklemesi gerekir. Bunun yanında SDL sistemde OpenGL, OpenGL ES, Direct3D varsa en uygununu kullanır. Video kartı bir gereklilik değildir fakat olması programın çizim hızını arttıracaktır.

Simülasyon -x yönündeki hızları desteklememektedir.

Ürünün 18 Ocak 2020 tarihine kadar geliştirilmesi beklenmektedir.

Koordinat düzlemi en fazla 1000m ye kadar en az 1m ye kadar ölçeklenebilir.

Girdi kısıtları 3.1 de verilmiştir.

## 2.5 Varsayımlar ve Bağımlılıklar

Kullanıcının sistem ile etkileşime geçebilecek donanımları(mouse, touchpad, keyboard) bulunduğu varsayılır. Kullanıcının SDL2 çalıştırabilen bir İşletim Sistemine sahip olduğu varsayılır.

### 3. Özel Gereksinimler

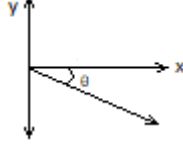
#### 3.1 İşlevsel Gereksinimler

Sistemin girdileri :

- **Yi** : Cismin yüksekliğini ifade eder. Cismin y ekseninde bulunduğu noktanın değeridir. Örneğin (0,6) noktasının Yi değeri 6 dır.
- **Vx** : Cismin x eksenindeki hızının büyüklüğünü ifade eder.
- **Vy+** : Cismin y eksenindeki yukarı yönlü hızının büyüklüğünü ifade eder.
- **Vy-** : Cismin y eksenindeki aşağı yönlü hızının büyüklüğünü ifade eder.
- **V** : Cismin bileşke hızını ifade eder.
- **θ+** : Cismin (+|+) bölgesinde yatay x eksenini ile yaptığı açığı ifade eder.



- **θ-** : Cismin (+|-) bölgesinde yatay x eksenini ile yaptığı açığı ifade eder.



- **g** : Yerçekimsel ivmeyi ifade eder.

Sistemin çıktıları :

- **t** : Cismin havada kaldığı süreyi ifade eder. Uçuş süresidir.
- **Xx** : Cismin x ekseninde aldığı toplam yolu temsil eder.
- **Xy** : Cismin y ekseninde aldığı toplam yolu temsil eder.
- **ΔXx** : x eksenindeki yer değiştirmeyi ifade eder.
- **ΔXy** : y eksenindeki yer değiştirmeyi ifade eder.
- Bunlara ek olarak **Vx, Vy, V, θ** nın anlık değerleri de çıktılarda gösterilir.

Girdi ve Çıktı Formatları :

- Girdiler ve çıktılar metre ve saniye birimleri ile gösterilecektir. girdiler tamsayı olarak çıktılar float(.00) formatında gösterilecektir.

Girdi Değer Aralıkları :

- **Vx, Vy** için [0,25]
- **V** için [0,50]
- **θ** için [0,90]
- **Yi** için [0,250]
- **g** için [0,10] aralığında olmalıdır.

#### Çıktı Değer Aralıkları :

- Atış açısı( $\theta$ ) için  $[0,90]$
- Hız değerleri( $V$ ,  $V_x$ ,  $V_y$ ) için  $[0,300]$
- Konum değerleri( $X_x$ ,  $X_y$ ,  $\Delta X_x$ ,  $\Delta X_y$ ) için  $[0,500]$
- zaman( $t$ ) için  $[0,50]$  aralığında olmalıdır.

#### Simülasyon sonlanma şartları :

- yukarıya doğru düşey atış, eğik atış, yatay atış, serbest düşme hareketleri ve  $g=0$  için: süre sınırı ile: 10 sn
- Diğer hareketler için: Cisim yere çarptığı anda.

#### Yasak girdiler :

- Sistem illegal bir girdi tespit ettiğinde (örn:  $V$  : -10 m/s) uyarı vermelidir. İzin verilen girdi aralıkları yukarı belirtilmiştir.

Ürünün işlevleri bölüm 2.2'de anlatıldığı gibi altı atış hareketinden ibarettir. Bu hareketlerin nasıl oluştuğu, hangi girdi aralıkları ile hangi hareketin gerçekleştiği, hareketlerin alternatif girdilerle nasıl gerçekleştirileceği Ek A'da verilmiştir.

## 3.2 Sistem Özellikleri

### 3.2.1 Hareket Penceresi

- Kullanıcılar cismin ilk konumunu direkt olarak istedikleri noktaya(y ekseninde) taşıyabilmeli ve “Yi” girdi alanında cismin konumu otomatik gösterilmelidir.
- Koordinat eksenleri Hareket ekranında sabit olmalıdır. x ve y eksenleri ile hareket ekranının kenarları arasında biraz boşluk olmalıdır.
- Kullanıcı mouse scroll yaparak veya +,- butonlarıyla koordinat düzlemini istedikleri kadar büyültüp küçültebilmelidir.
- Cisim hareket ekranında varsayılan olarak (0,0) noktasındadır.
- Cisim üzerine tıklandığında konumu (0,3) şeklinde gösterilmeli. Cisme tekrar tıklandığında konum bilgisi kaybolmalı. Hareket ederken konum bilgisi anlık olarak değişmeli.
- Cisim hareket halindeyken hareket ekranından çıkarsa ölçek otomatik olarak büyümeli.
- Cisim hareket halindeyken hareket rotası kesik çizgilerle gösterilmeli.
- Koordinat düzlemi varsayılan olarak 20şer 20şer gösterilmeli. Ölçeğe göre bu değerler değişebilmelidir.

### 3.2.2 Girdi/Çıktı Arayüzü

- Kullanıcı iki farklı girdi yoluyla simülasyonu başlatabilmeli;
- 1.girdi yolu: ilk hız(V), Atış açısı( $\theta$ )
- 2.girdi yolu: x eksenindeki ilk hız( $V_x$ ), y eksenindeki ilk hız( $V_y$ ).
- şeklinde olmalıdır.
- Diğer girdiler: ilk konum( $Y_i$ ) ve yerçekimi ivmesi(g) nin her hareket için girilmesi zorunlu olmalı.
- Kullanıcının girdi yollarından birini seçebilmeli.
- Simülasyon başlatılabilmeli, tekrarlanabilmeli, sıfırlanabilmeli. Simülasyon sırasında ise durdurulabilmeli ve devam ettirilebilmeli.
- Simülasyon sırasında çıktıların yanında birimleri yazmalı. Ayrıca çıktının özel anlamı varsa yanında yazmalı (örneğin V : 42.53 m/s (çarpma hızı) )

- $g$  yerçekimi ivmesi girdisine varsayılan olarak  $10 \text{ m/s}^2$  değeri girilmiş bulunmalıdır.
- $y$  eksenindeki hız( $V_y$ ) değerinin yönü belirlenebilmeli.
- Atış açısı( $\theta$ ) değerinin yönü belirlenebilmeli
- $V_y$  ve  $\theta$  girdilerine 0 değeri girildiğinde yön seçimi, sonucu etkilememeli. Eğer simülasyon yön seçmeden başlatılırsa sistem otomatik olarak girdiye 0 muamelesi yapmalı.
- Çıktıların tepki süresi anlık olmalı. Kullanıcı, cisim hareket halindeyken değişen çıktıları görebilmeli.
- Girdilerin değerine göre hareketin tipi kullanıcıya gösterilmelidir.
- Simülasyon sırasındaki görünüm aşağıdakine benzer olmalıdır:

Hareket Tipi : Yatay Atış Hareketi				Hareket Çıktıları:		
Yi : <input type="text" value="50"/>	Vx : <input type="text" value="5"/>	Vy : <input type="text" value="0"/>	<input checked="" type="checkbox"/>	t : <input type="text" value="2.23 (s)"/>	V : <input type="text" value="22.85 (m/s)"/>	$\theta$ : <input type="text" value="77.04°"/>
g : <input type="text" value="10"/>	V : <input type="text" value=""/>	$\theta$ : <input type="text" value=""/>	<input type="checkbox"/>	Vx : <input type="text" value="5 (m/s)"/>	Xx : <input type="text" value="11.15 (m)"/>	$\Delta Xx$ : <input type="text" value="11.15 (m)"/>
<input type="button" value="DUR"/> <input type="button" value="DEVAM"/>				Vy : <input type="text" value="22.30 (m/s)"/>	Xy : <input type="text" value="24.86 (m)"/>	$\Delta Xy$ : <input type="text" value="24.86 (m)"/>

### 3.2.3 Bilgi Penceresi

Simülasyonun kullanımına ve temel fonksiyonların icrasına dair kısa ve anlaşılır bilgi içermeli. Girdi ve çıktıların anlamlarına dair bilgi vermeli.

### 3.2.4 Simülasyon Penceresi

Responsive bir tasarıma sahip olmalı. belirli ekran boyutlarında nasıl görüneceği ayarlanmalıdır.

Aynı anda birden fazla istemcinin açılmasına izin verilmemelidir.

### 3.3 Performans Gereksinimleri

Genel olarak fazla kaynak tüketmez. Hemen hemen her bilgisayarda düzgünce çalışır.

### 3.4 Tasarım Kısıtları

Minimum girdi aralıkları için sorunsuzca 6 çeşit atış hareketini 1 saniye aralıklarla anlık olarak hem cismin hareketi hem değişen çıktı değerlerini gösterebilen ürün başarılı sayılmaktadır. Gereksinim dosyasının şu anki hali minimum beklenti için hazırlanmıştır. Gelecekte eklenecek özellikler minimum beklentinin dışındadır.

### 3.5 Kalite Özellikleri

#### 3.5.1 Güvenilirlik

Yazılımın çökme ihtimali oldukça düşüktür. Yeni güncellemelerle birlikte yazılım, geliştiriciler tarafından test edilerek sistem bugları ve çökme sorunları giderilmektedir.

#### 3.5.2 Erişilebilirlik

Bu program, geliştiriciler tarafından yayınlanan haliyle kullanıma hazırdır. Ancak bu belgede de belirtildiği üzere SDL2.dll dosyasının programın bulunduğu dosyada olması gerekir.

Yazılım çevrimdışı olarak kullanılabilir.

#### 3.5.3 Güvenlik

Yazılım herhangi bir veritabanı kullanmadığı gibi herhangi harici bir sistemle veri transferi de yapmaz. Ayrıca kullanıcı sistemi, kullanıcı modları(admin gibi) olmadığından herhangi kullanıcı bilgileri de tutulmaz. Dolayısıyla veri güvenliği kaygısı taşımaz.

Yazılım ücretsizdir. Kaynak kodları incelenebilir dağıtılabilir veya değiştirilebilir. Dolayısıyla herhangi bir kriptografi kullanılması gerekmemektedir.

### 3.5.4 Sürdürülebilirlik

Yazılım mümkün olduğunca modüler hazırlanacaktır. Yazılımın değişiklikleri kolaylıkla uygulanabilecek şekilde geliştirilmesi hedeflenmektedir. Değiştirilmesi muhtemel özellikler 4.bölümde verilmiştir.

Yazılımın bakımı, sistem beklentileri karşılanıncaya kadar geliştirici grubu tarafından yapılacaktır.

### 3.5.5 Taşınabilirlik

Yazılım SDL destekleyen bütün işletim sistemlerine çalışan cihazlara taşınabilir

### 3.5.6 Kullanılabilirlik

Simülasyon Penceresinde bulunan tüm yazılar kullanıcıların kolay anlayacağı bir yazı tipinde ve boyutunda gösterilmelidir. Program işlevlerinin kullanımı hakkında kısa ve öz bilgi verilmelidir. Kolay bir kullanım sağlanmalıdır.

## 4. Gelecekte Yapılması Planlananlar

Değiştirilebilecek olan özellikler:

- Ürün girdilerinin birimleri gelecekte değiştirilebilir. Bu yüzden değiştirilebilir tasarlanmalıdır.
- Ürün girdilerinin sınırları gelecekte arttırılabilir. Bu yüzden değiştirilebilir tasarlanmalıdır.
- Yazılımın ismi değiştirilebilir.
- Cismin şekli gelecekte değiştirilebilir. Değiştirilmeye elverişli tasarlanmalıdır.

Yeni eklenebilecek özellikler:

- Simülasyon tamamlandıktan sonra cismin belirli bir andaki çıktı değerlerini görmek için mouse ile cismi hareket ettirmek dışında 2.bir yol olarak süre(t) girerek görme de eklenebilir.
- Cisim üzerinde hız vektörleri gösterilebilir. Bu vektörlerin değişimine göre büyüyüp küçülebilir yapılabilir.
- Çıktıları basitleştirmek adına harekete göre bazı birbirine denk olan çıktılar tek çıktı olarak şu şekilde  $V_x = V : 10 \text{ m/s}$  gösterilebilir. Simülasyon sıfırlandığında varsayılan haline döner.
- Programa giriş ekranı eklenebilir.
- Giriş ekranında kullanıcıya programı nasıl kullanabileceğini anlatan bir yönerge için buton eklenebilir.
- Cismin konumu ve hız vektörünü fare inputları ile de ayarlanabilir olma özelliği

eklenebilir.

- Kullanıcının oluşturduğu simülasyonları kayıt edebilip, istediği zaman aynı simülasyonu tekrar gözlemleyebilmesi programa eklenebilir.
- Görsel olarak ekstra bir mod olan gece modu eklenebilir.
- Programa, atış hareketinde hava sürtünmesinin nasıl bir etkisi olduğunu gösterebilmek için yeni bir değişken olarak hava sürtünmesi eklenebilir.

•

Gelecekte tamamlanacak bölümler:

- Ürün için minimum donanım gereksinimleri test aşamasında belirlenecektir. Minimum donanım gereksinimleri 500 MHz CPU + 128 MB RAM olarak hedeflenmektedir. Gelecekte performansı arttırmak için OpenGL kullanılabilir. Bir grafik kartı gerekli olacaktır.
- Programın gerek duyduğu bütün yazılım ve donanımlar tam olarak tanımlanmamıştır. Program test edilme aşamasındayken tanımlanacaktır.
- Program ile ilgili youtube tutorialları, Online documents yapılabilir.



## EK A

Aşağıdaki tabloda tüm Atış hareketlerinin karakteristik özellikleri verilmiştir.

Hareket Adı	Kullanılan Girdiler	Alternatif Girdiler	Çıktılar	Şartlar	Alternatif Şartlar
Serbest Düşme	$Y_i, V_x, V_y$	$Y_i, V, \alpha$	$t, V=V_y, X_y=\Delta X_y, \alpha$	$V_x=0, V_y=0, Y_i \neq 0$	$V=0, Y_i \neq 0, \alpha=90^\circ$
Yukarıdan Aşağı Düşey Atış Hareketi	$Y_i, V_x, V_y$	$Y_i, V, \alpha$	$t, V=V_y, X_y=\Delta X_y, \alpha$	$V_x=0, V_y \neq 0, Y_i \neq 0$	$V \neq 0, Y_i \neq 0, \alpha=90^\circ$
Aşağıdan Yukarıya Atış Hareketi	$Y_i, V_x, V_y$	$Y_i, V, \alpha$	$t, V=V_y, X_y, \Delta X_y, \alpha$	$V_x=0, V_y \neq 0, Y_i \neq 0$	$V \neq 0, Y_i \neq 0, \alpha=90^\circ$
Yatay Atış Hareketi	$Y_i, V_x, V_y$	$Y_i, V, \alpha$	$t, V_x, V_y, V, X_y=\Delta X_y, X_x=\Delta X_x, \alpha$	$V_x \neq 0, V_y=0, Y_i \neq 0$	$V=0, Y_i \neq 0, \alpha=0^\circ$
Eğik Atış Hareketi	$Y_i, V_x, V_y$	$Y_i, V, \alpha$	$t, V_x, V_y, V, X_y, \Delta X_y, X_x=\Delta X_x, \alpha$	$V_x \neq 0, V_y \neq 0, Y_i \neq 0$	$V \neq 0, Y_i \neq 0, \alpha=(0^\circ, 90^\circ)$
Pike Atış Hareketi	$Y_i, V_x, V_y$	$Y_i, V, \alpha$	$t, V_x, V_y, V, X_y=\Delta X_y, X_x=\Delta X_x, \alpha$	$V_x \neq 0, V_y \neq 0, Y_i \neq 0$	$V \neq 0, Y_i \neq 0, \alpha=(0^\circ, 90^\circ)$

**Kullanılan girdiler:** İlgili hareketin başlatılması için girilmesi gereken değerleri ifade eder.

**Alternatif girdiler:** eğer seçilirse hareketin başlaması için gereken alternatif değerleri ifade eder.

**Değiştirilen çıktılar:** hareket sırasında ve sonucunda değişecek çıktıları ifade eder.

**Şartlar:** Hareketin belirtilen hareket olabilmesi için girdilerin hangi aralıklarda verilmesi gerektiğini ifade eder.

**Alternatif şartlar:** Alternatif girdiler kullanıldığında hareketin belirtilen hareket olabilmesi için girdilerin hangi aralıklarda verilmesi gerektiğini ifade eder.

**Önemli Not:** Tabloda  $V_y$  ve  $g$  girdi parametrelerine - veya + eki getirilmemesi bu parametrelerin değerinin sıfır olarak kullanılacağını ifade eder.

**Önemli Not:**  $V_x$  parametresi sadece +x yönündeki hızı ifade eder.

