

**HACETTEPE UNIVERSITY  
ENGINEERING FACULTY  
DEPARTMENT OF COMPUTER ENGINEERING**

**BBM 325  
INTERNSHIP REPORT**

**Deniz Ece AKTAŞ  
21626901**

**Performed at  
ATOM  
Animasyon Teknolojileri ve  
Oyun Geliştirme Merkezi**

**16/07/19 – 29/08/19  
30 Work Days**

## TABLE OF CONTENTS

<b>1</b>	<b>Introduction</b>	2
<b>2</b>	<b>Company Information</b>	2
2.1	<i>About the company</i>	2
2.2	<i>About your department</i>	4
2.3	<i>About the hardware and software systems</i>	4
<b>3</b>	<b>Work Done</b>	4
<b>4</b>	<b>Performance and Outcomes</b>	9
4.1	<i>Applying Knowledge and Skills Learned at Hacettepe</i>	9
4.2	<i>Solving Engineering Problems</i>	9
4.3	<i>Teamwork</i>	9
4.4	<i>Multi-Disciplinary Work</i>	10
4.5	<i>Professional and Ethical Issues</i>	10
4.6	<i>Impact of Engineering Solutions</i>	10
4.7	<i>Locating Sources and Self-Learning</i>	11
4.8	<i>Using New Tools and Technologies</i>	11
<b>5</b>	<b>Conclusions</b>	11
	<b>References</b>	12
	<b>Appendices</b>	12

# 1 Introduction

Animasyon Teknolojileri ve Oyun Geliştirme Merkezi, by short name ATOM, is pre-incubation center established under METU TEKNOKENT Yönetim INC.

The aim of the Center is to strengthen the sector and provide a healthy ecosystem to the sector by providing various support to the teams that are active or interested in game development and animation technologies for a year.

I love playing video games and I find the creative process of a game very interesting so I wanted to learn about this subject in more detail because of this I decided to do my internship at ATOM.

During my internship I worked on five mobile game prototypes and three existing mobile games of one of the publishers in ATOM. For existing games, I learnt about level design mostly and with the prototypes I improved my Unity and C# skills.

## 2 Company Information

### 2.1 About the company

Animation Technologies and Game Development Center, shortly known as ATOM, was established in 2008 by METU TEKNOKENT Yönetim A.Ş. is a pre-incubation center.

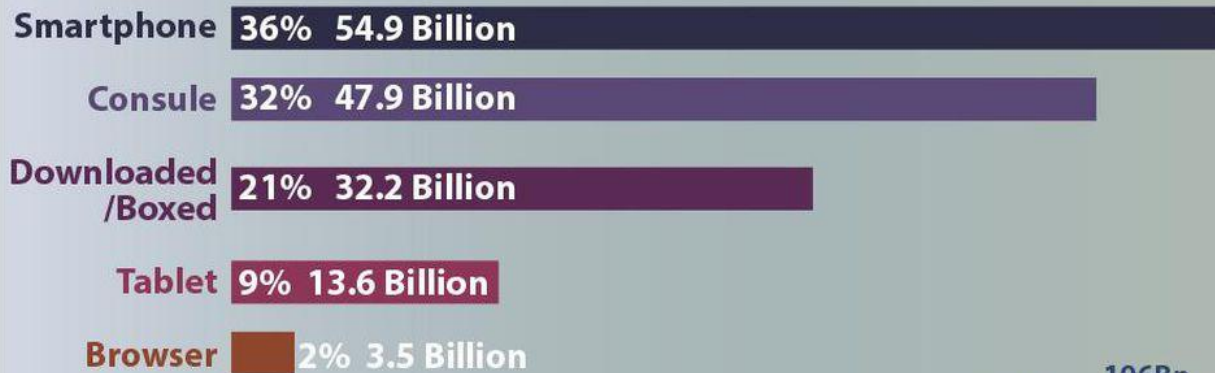
The aim of the foundation was to support the creative industries and to make an enjoyable contribution to the formation of the game and animation industries.

The gaming industry has surpassed \$ 100 billion worldwide. This value is greater than the sum of the film and music industries. The animation industry, on the other hand, is intertwined with other creative areas, but is reported to be over \$ 250 billion in size.

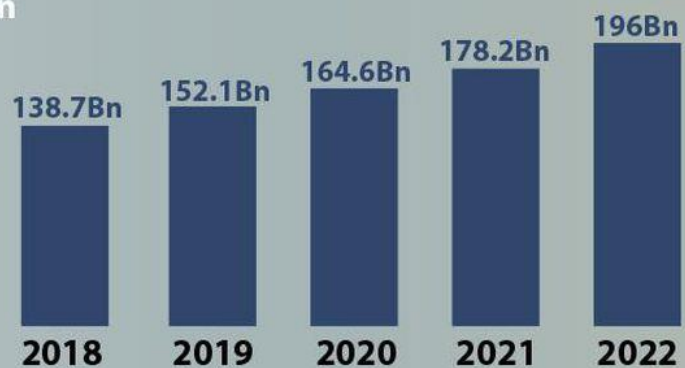
Last year the gaming market generated somewhere around \$138.7 billion in revenue. The prediction from Newzoo is that in 2019 it will increase by about 9.6% to generate 152.1 billion. The video game industry grew about 10% in 2018 so to say that it will grow a similar amount seems to make sense.[1]

Today, ATOM continues to progress and develop as the cuisine and representative of the gaming industry. For example, “Darklings” and “Palyaço Evi” are two games made by the teams in ATOM.

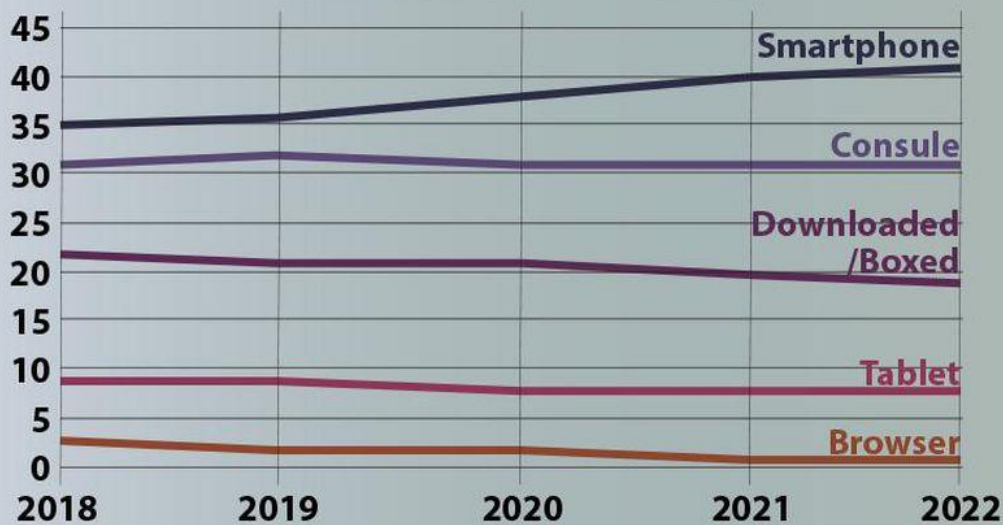
## 2019 market share by gaming platform



## Predicted market growth from 2018 to 2022



## Gaming platform market share growth from 2018 to 2022



Forbes

## 2.2 About your department

Because ATOM is a pre-incubation center there is no department, there are teams. Therefore, I will describe my team as my department. Our team is called “Xpanse” and I was doing my internship as game developer/programmer. My team’s focus was on casual and hyper casual games. Some products that our team published are “Bal Ayısı : Tatlı Kelime Avı[8], Swingomania[9] , Flip’a’Coin 3D[10]”.

## 2.3 About the hardware and software systems

I was using my own computer at internship. As Program I used Unity 2018.4.4f1 and for coding Visual Studio 2017 Community Edition and used C# as the coding language.

## 2.4 About your supervisor

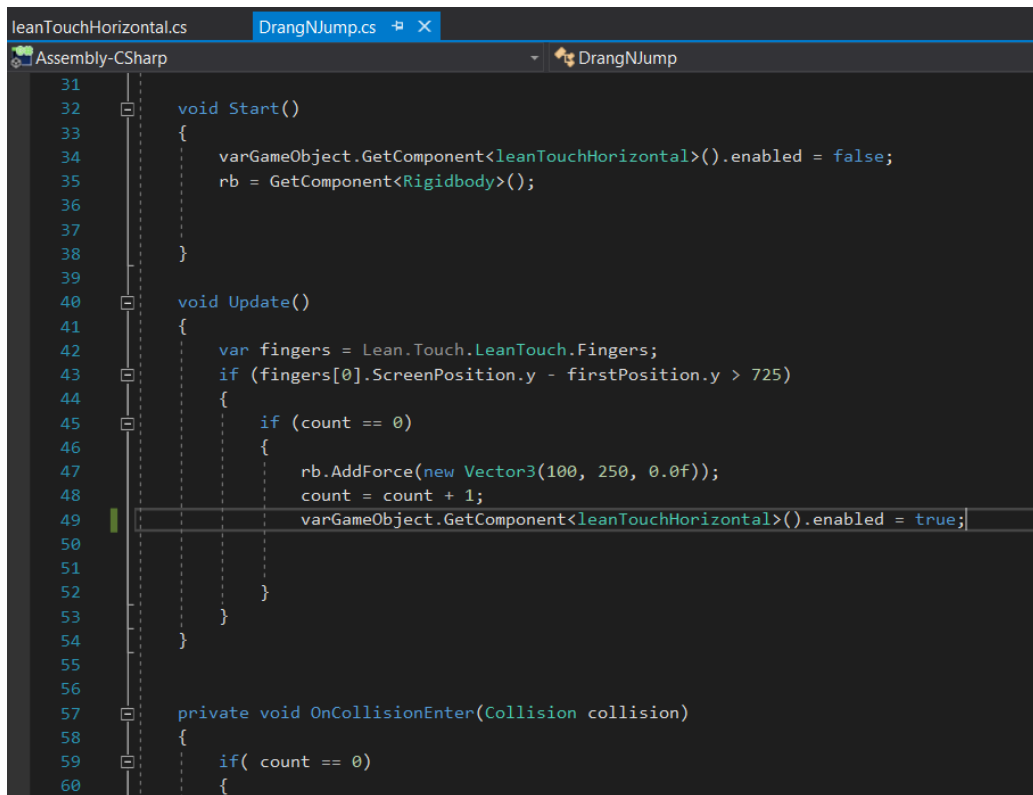
- **Name:**Egemen Can GÜLER
- **Mail:**egemencanguler@gmail.com
- **Tel:** 0 536 675 6174
- **Education:** Hacettepe University – Department of Computer Engineering (2018)
- **Work Address:** Animasyon Teknolojileri ve Oyun Geliştirme Merkezi(ATOM) ODTÜ Teknokent Galyum Blok, Bodrum Kat No:15 06800/ODTÜ - Ankara

## 3 Work Done

During the first week of my internship I focused on learning the systems that I would be working with, Unity and C#, at that time I made three untitled works on Unity. In these works I just got to know the program and played around with physics of the games and at the same time our instructor wanted us to learn assets to use on our projects and I started to learn about lean touch asset and used it on later projects. Lean touch helps to use the game on a touch screen device.

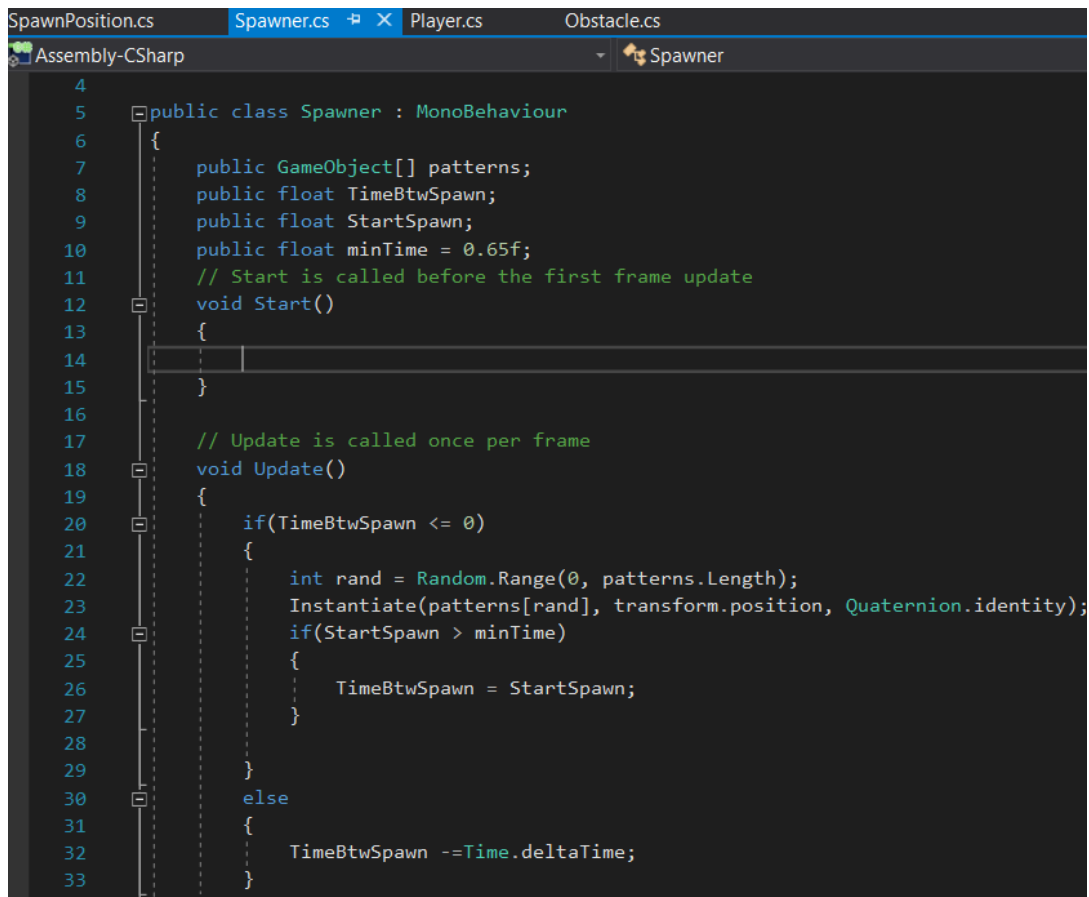
My first untitled work was just a ball that every time player touched that would jump upwards and to the right of the screen. I just used one script that adds force to the object every time the ball was tapped.

My second untitled project was again just a ball and a box but this time player had to drag the box under the ball to keep the ball’s jumping motion going. This time I used two scripts one was a lean touch script that was ready to use I just implemented it to the second script that did the whole motion.



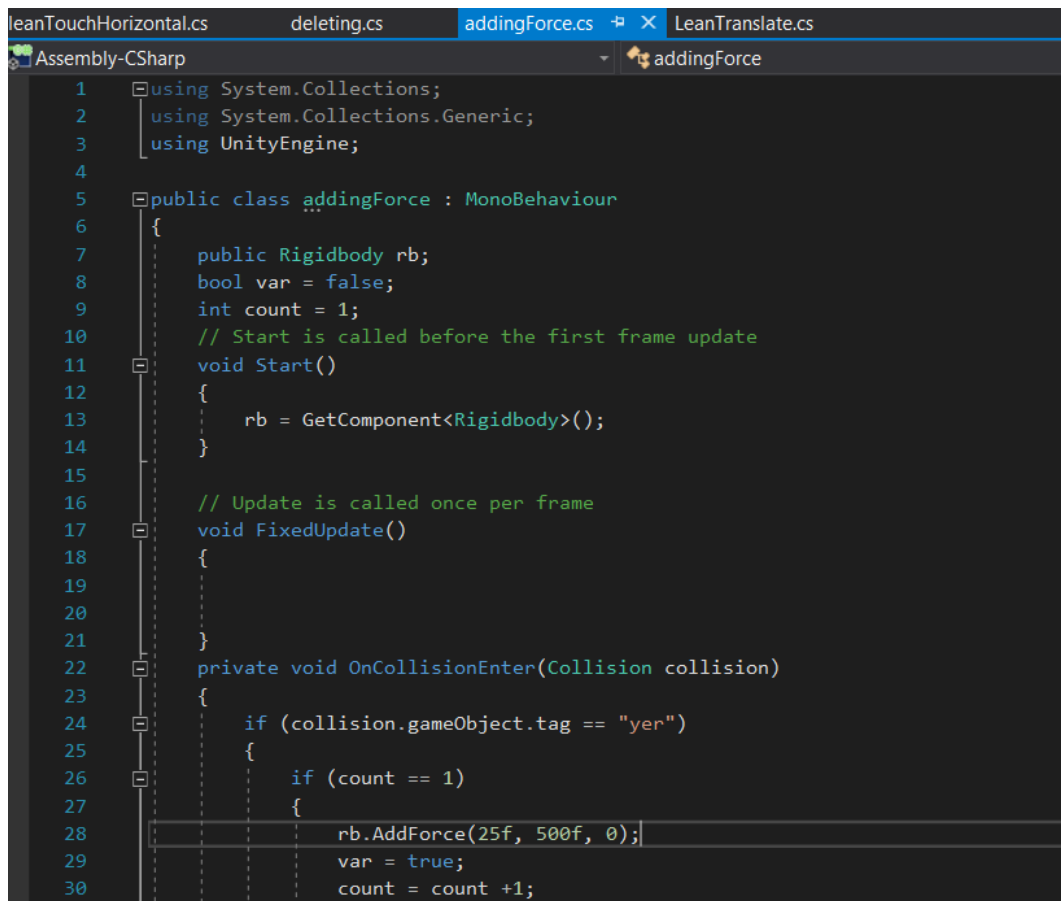
```
31
32 void Start()
33 {
34     varGameObject.GetComponent<LeanTouchHorizontal>().enabled = false;
35     rb = GetComponent<Rigidbody>();
36
37 }
38
39
40 void Update()
41 {
42     var fingers = Lean.Touch.LeanTouch.Fingers;
43     if (fingers[0].ScreenPosition.y - firstPosition.y > 725)
44     {
45         if (count == 0)
46         {
47             rb.AddForce(new Vector3(100, 250, 0.0f));
48             count = count + 1;
49             varGameObject.GetComponent<LeanTouchHorizontal>().enabled = true;
50         }
51     }
52 }
53
54
55
56
57 private void OnCollisionEnter(Collision collision)
58 {
59     if( count == 0)
60     {
```

In my third work I made a simple prototype that has a box and capsule objects. Capsule objects are generated right outside of the game screen and they are generated in different positions and quantities. Capsule objects come towards the screen in stable speed and the player has to tap the screen to make the box jump. When the player touches capsules three times it is game over. For this prototype I wrote four scripts. One was an “Obstacle” script that decreased the health every time player touched the capsule. And the second script was “Player”, got the data of how many fingers were touching the screen via one lean touch’s features -touchcount- and adds force according to finger numbers. In “Spawner” script it spawns obstacles at a random number and makes the spawn time faster each turn. In the “SpawnPosition” script program actually spawns the obstacles to the screen.



```
4
5 public class Spawner : MonoBehaviour
6 {
7     public GameObject[] patterns;
8     public float TimeBtwSpawn;
9     public float StartSpawn;
10    public float minTime = 0.65f;
11    // Start is called before the first frame update
12    void Start()
13    {
14    }
15
16
17    // Update is called once per frame
18    void Update()
19    {
20        if(TimeBtwSpawn <= 0)
21        {
22            int rand = Random.Range(0, patterns.Length);
23            Instantiate(patterns[rand], transform.position, Quaternion.identity);
24            if(StartSpawn > minTime)
25            {
26                TimeBtwSpawn = StartSpawn;
27            }
28        }
29        else
30        {
31            TimeBtwSpawn -= Time.deltaTime;
32        }
33    }
```

My forth prototype was an elevated version of my second work. I did the same things like dragging the box under the ball to keep ball jumping but this time I added obstacles that move on the upper part of the screen that the ball shouldn't touch. After the player touches the objects three times game ends. In this work I made three scripts "deleting" delete the obstacles that player touched, "leanTouchHorizontal" is again is already made script lean touch assed provides. And the final script is "addingForce", it adds force to the ball every time the ball collides with the box underneath. But to make obstacles move in the screen and to spawn them for the first time I used Unity's animation.



```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class addingForce : MonoBehaviour
6  {
7      public Rigidbody rb;
8      bool var = false;
9      int count = 1;
10     // Start is called before the first frame update
11     void Start()
12     {
13         rb = GetComponent<Rigidbody>();
14     }
15
16     // Update is called once per frame
17     void FixedUpdate()
18     {
19
20     }
21
22     private void OnCollisionEnter(Collision collision)
23     {
24         if (collision.gameObject.tag == "yer")
25         {
26             if (count == 1)
27             {
28                 rb.AddForce(25f, 500f, 0);
29                 var = true;
30                 count = count + 1;
```

After I finished all of these me and another intern were told to make a “Hang point and collider generator” for levels in their published game called “Swingomania”. They needed this generator to create more levels, faster and easier. After assigning this project us discussed how to approach this project because of Unity we couldn’t use two different computers for coding and we decided to write algorithm on paper then write in computer. The generator would be worked when editor gives points where players will pass and with this given points generator will create hookable points with given distances from each other and create zones that will be used to decide who is in lead. Script basically works like this, with list of points script finds calculates vector for every two point and with given distance creates hookable points, for zones scripts calculates what angles needed to create zones and with given positions and given distance creates zones in level. After algorithm finished my friend wrote the code in his computer, finished generator and send for control and use.

One day all the teammates and our supervisor sat in a room and brainstormed about new game projects or ideas. After this meeting I was assigned to make level designs for “Bal Ayısı : Tatlı Kelime Avı”. This is a game that was made before our internship and our supervisor had the level generator ready in a tablet. For three days I made around 40 levels for this game. I had to

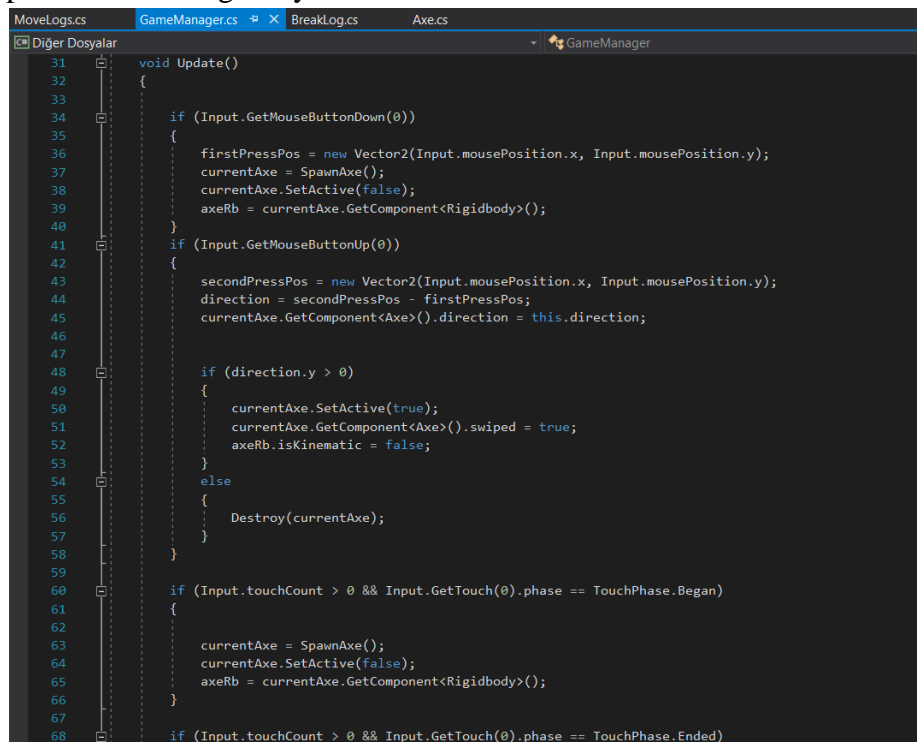


make games get progressively harder as the player plays the game and because it is a word game I had to consider how the player would see the layout.

After this one of our intern friends finished their prototype and because I was the only one with experience on level design at this point, I was assigned to generate levels for “Flip’a’Coin 3D”. I made twelve levels for this game and send them to our supervisor who collected and cleaned up the game and published it.

Then, again, I made eight levels for the game “Swingomania”. I used the hang point and collider generator” that me and my friend had made. I also used models made by the designers as well.

During the last two weeks of my internship I was working with another intern friend to work on a new prototype. Because we were having problems with the unity’s connection we wrote the code on one computer but we used the other computer for research purposes. This work is called “LogGame” and it is a game that whenever the player swipes at the screen the game spawns and sends an axe towards to a log object that continuously spawns right outside the screen and moves toward the right of game space. The player has to hit the log on the specific spots to cut the log and gain point. The most challenging parts of this project was making the axe move faster and hit it to log at a known angle every time and cutting the log as much as we want at hit points. In this prototype we wrote four scripts and they were, “MoveLogs”, “GameManager”, “BreakLogs” and “Axe”. “MoveLogs” moved the log object across the screen. “GameManager” spawned the axes and rotated them. “BreakLogs” broke the logs as much as we want and “Axe” got the collusion point and added gravity to the axe.



```
31 void Update()
32 {
33
34     if (Input.GetMouseButtonDown(0))
35     {
36         firstPressPos = new Vector2(Input.mousePosition.x, Input.mousePosition.y);
37         currentAxe = SpawnAxe();
38         currentAxe.SetActive(false);
39         axeRb = currentAxe.GetComponent<Rigidbody>();
40     }
41
42     if (Input.GetMouseButtonUp(0))
43     {
44         secondPressPos = new Vector2(Input.mousePosition.x, Input.mousePosition.y);
45         direction = secondPressPos - firstPressPos;
46         currentAxe.GetComponent<Axe>().direction = this.direction;
47
48         if (direction.y > 0)
49         {
50             currentAxe.SetActive(true);
51             currentAxe.GetComponent<Axe>().swiped = true;
52             axeRb.isKinematic = false;
53         }
54         else
55         {
56             Destroy(currentAxe);
57         }
58     }
59
60     if (Input.touchCount > 0 && Input.GetTouch(0).phase == TouchPhase.Began)
61     {
62         currentAxe = SpawnAxe();
63         currentAxe.SetActive(false);
64         axeRb = currentAxe.GetComponent<Rigidbody>();
65     }
66
67
68     if (Input.touchCount > 0 && Input.GetTouch(0).phase == TouchPhase.Ended)
```

## **4 Performance and Outcomes**

### **4.1 Applying Knowledge and Skills Learned at Hacettepe**

For Game Scripts in Unity I wrote my scripts using C#. In Hacettepe I learnt Java, C and Python but because we studied many languages in a short time I learnt how to code in C# and how to use Unity very fast.

During game development process physics (FIZ137) and math (MAT123) knowledge was very useful.

### **4.2 Solving Engineering Problems**

I didn't solve lot of engineering problems because there were not many. But when I need to look for something I searched online, read articles and documents about problem. Also I watched videos about the topic I was researching on Youtube. If I couldn't find anything useful for me on the documents firstly I asked my other intern friends and then if we couldn't figure it out I asked my supervisor.

### **4.3 Teamwork**

My teammates were also mostly interns as well. The team consisted of two designers who modelled the visuals and the rest of us were developers.

When I first started working, I didn't have any knowledge about how to use unity nor how to code with C# so the first week I focused on learning these programs but after this first week I worked a lot with my teammates.

On one of the prototypes we worked as a team with my friend and we discussed how to calculate a component we needed for our prototype and divided our work. Also on another project I worked with another teammate and we did the prototype from start to finish together and it really helped solving problems. Because if I made a mistake then my friend warned me and visa versa. These two teammates are both students in our university and were both doing their internship in ATOM just like me. We also had two other developer friends and they are both students who were doing their internships as well. And even if we were working on our own prototypes we were always working with our designer teammates. They made the models we used in our games.

## **4.4 Multi-Disciplinary Work**

In our team there were two 3D model designers who worked with Blender. They helped to make models and assets for the games that we created but there weren't any other disciplines between our teammates. The rest of us were developers.

## **4.5 Professional and Ethical Issues**

During my internship, I did not observe any big or important professional issues or work-related ethical issues.

Even if there were little issues between the teammates everybody was kind to each other and the issues were handled with talking or asking other members or supervisors.

## **4.6 Impact of Engineering Solutions**

While I was in ATOM, we had a lesson about the game development. In that lesson we learned how our prototype developing and our project's effect on users and how can we make our projects better to be used more. For example, we learnt that how to make a hyper casual game and how to make it more enjoyable for the players.

In an economic way, If a prototype finishes faster, It goes to the publishers to test if the prototype can become a real mobile game. Because of this, when there are more prototypes that means there is a higher chance of that game getting published. So, all of our team didn't work on one project collectively. Even if we were working with a teammate there was always one more prototype which our other teammates were working on. Also, as soon as one project finished, we started on another one almost immediately because of these reasons our team constantly made new prototypes to send to publishers.

For global impact, If we want the game to be successful around the world then the game should be accessible to everyone, children adults and even elderly, every single person should get some sort of enjoyment out of the game. If we are working on a hyper casual game like we did in ATOM then the game should be very easy to understand and play something that adds a fair challenge and makes the player want to play more.

## **4.7 Locating Sources and Self-Learning**

During my internship I learned how to use Unity and C# mostly with YouTube tutorials from Brackeys Channel [2] and also from the channel Blackthornprod [3]. I also used the site stackoverflow[4] if I was researching a certain problem. If there was a feature that I needed to use I searched on google. To implement our projects to mobile phones we used an asset which is called Lean Touch from Carlos Wilkes. There were not many videos about Lean Touch so we learned what we need from Carlos Wilkes's own Documentation about Lean Touch [5]. For C# Scripts in Unity, I also looked to Unity's own Documentation site [6]. I also made research on Unity's asset store [7] to find assets to use in my prototypes.

## **4.8 Using New Tools and Technologies**

Although I always wanted to learn Unity during the school terms I couldn't find time to do so. Naturally I was very happy to start learning Unity.

Because I didn't know Unity nor C# at the start of the internship so I had to learn them from basics but with the help of this internship, now I can make my own games with Unity and with Visual Studio and C#. And before this internship I hadn't used Visual Studio too.

I learned how to use new game development systems and a new programming language.

# **5 Conclusions**

In conclusion, I worked on five different prototypes and made level designs for three existing games. I learnt how to use C# and Unity. I used the things I learnt in my physics and math classes. I also worked in teams which I didn't do until now because in Hacettepe we work solo but because we used so many languages in school, I learnt C# really fast. Me and my teammates were challenged to do our works from scratch and I think that really helped me to be more responsible as well.

# References

[1]”Forbes”

<https://www.forbes.com/sites/kevinanderton/2019/06/26/the-business-of-video-games-market-share-for-gaming-platforms-in-2019-infographic/#44bcd9427b25>

[2] “Brackeys, How to make a video game playlist”. “Brackeys, Menus in Unity”

<https://www.youtube.com/user/Brackeys/playlists>

[3] “Blackthornprod, Endless runner tutorial series”

[https://www.youtube.com/playlist?list=PLBIb\\_auVtBwBkYGKni2wKHGVFP5b4pVwj](https://www.youtube.com/playlist?list=PLBIb_auVtBwBkYGKni2wKHGVFP5b4pVwj)

[4] “Stackoverflow”

<https://stackoverflow.com/>

[5] “Lean Touch Documentation Charles Wilkes”.

<http://carloswilkes.com/Documentation/LeanTouch>

[6] “Unity Documentation”

<https://docs.unity3d.com/Manual/index.html>

[7] “Unity Asset Store”

<https://assetstore.unity.com/>

[8] “Bal Ayısı : Tatlı Kelime Avı”

<https://play.google.com/store/apps/details?id=com.Badu.BalAyisi&hl=tr>

[9]“Swingomania”

<https://apps.apple.com/tr/app/swingomania/id1472201996?l=tr>

[10] “Flip’a’Coin 3D”

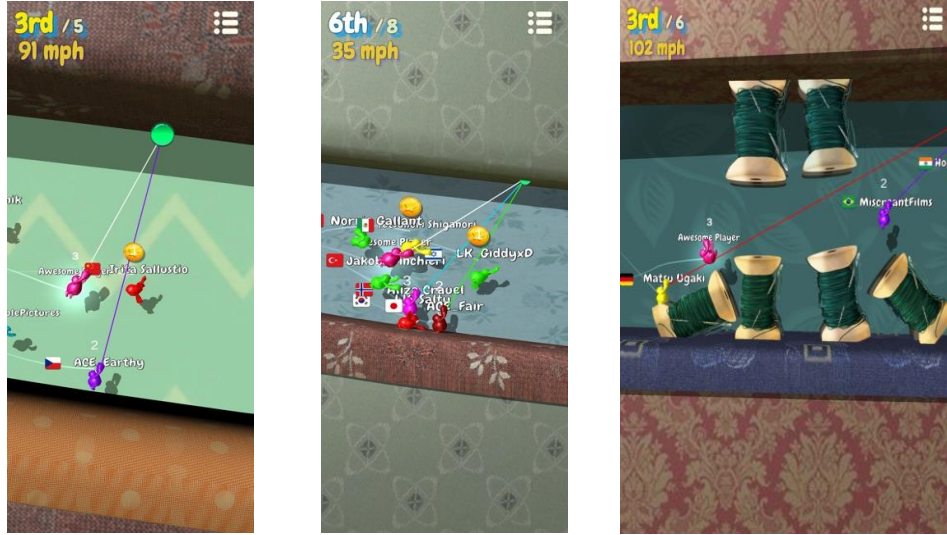
<https://apps.apple.com/tr/app/flipacoin-3d/id1476935174?l=tr>

# Appendices

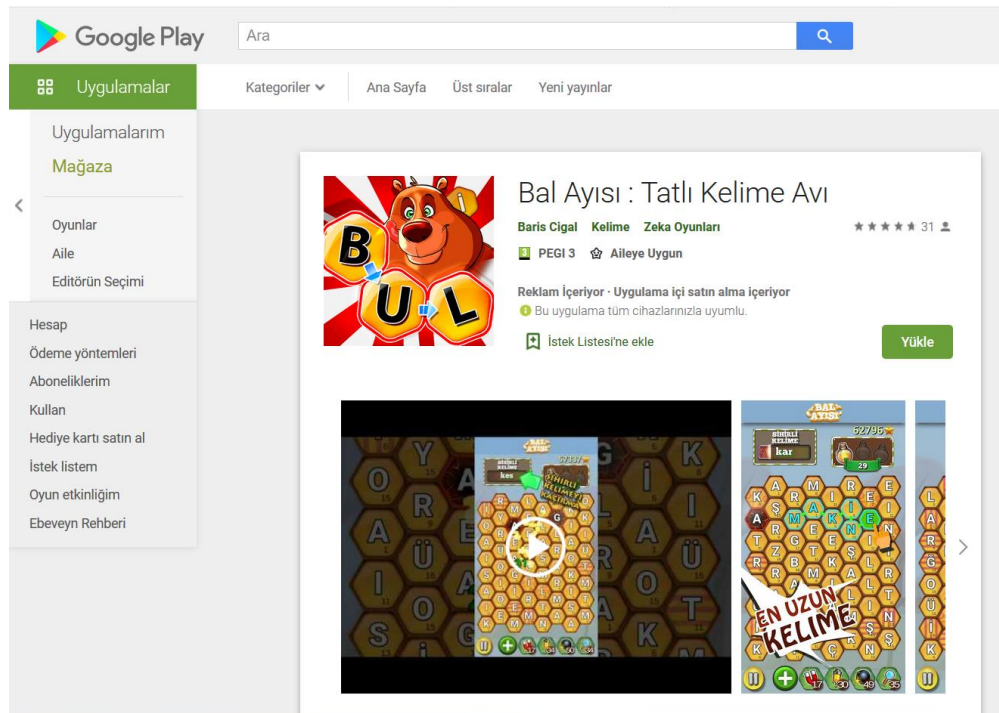
Swingomania hang point generator

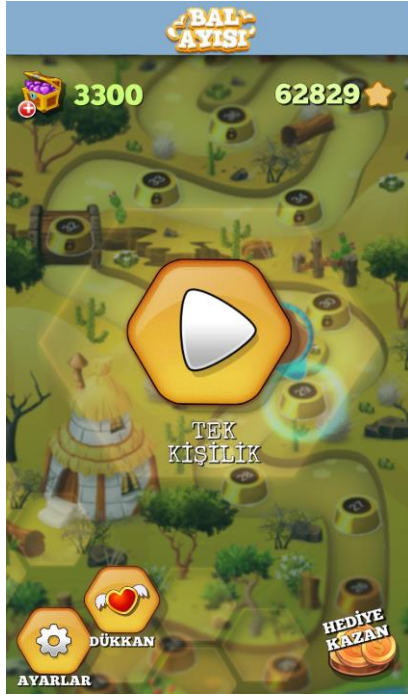


## The gameplay of Swingomania



## Bal Ayısı: Tatlı Kelime Avı

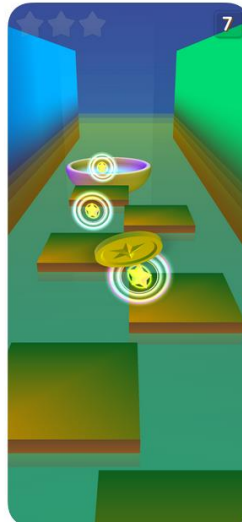
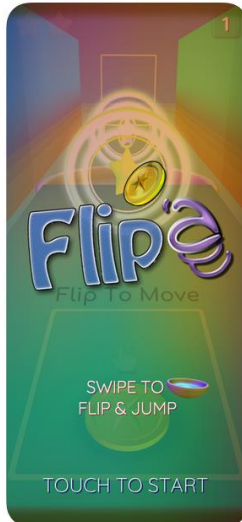




Flip'a'coin

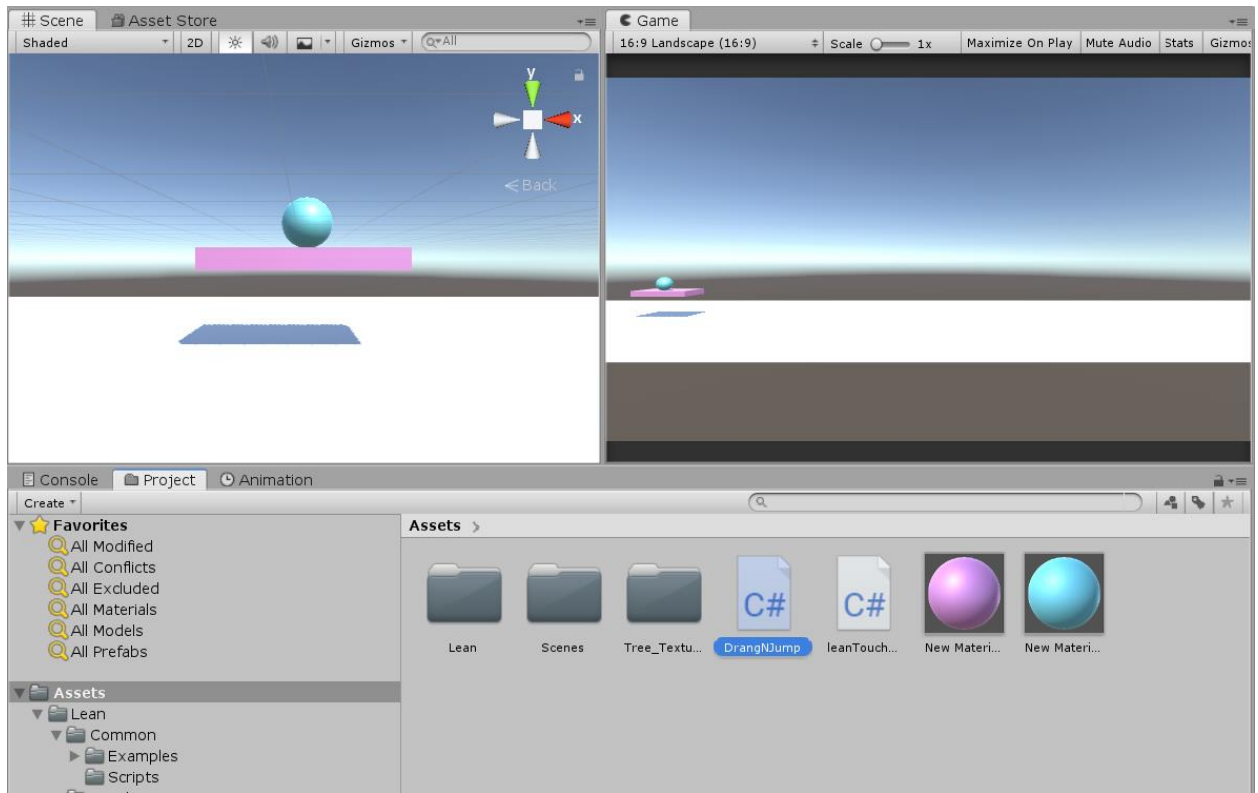


**Flip'a'Coin 3D** 4+  
Baris CIGAL  
★★★★★ 5,0, 1 Oy  
Ücretsiz

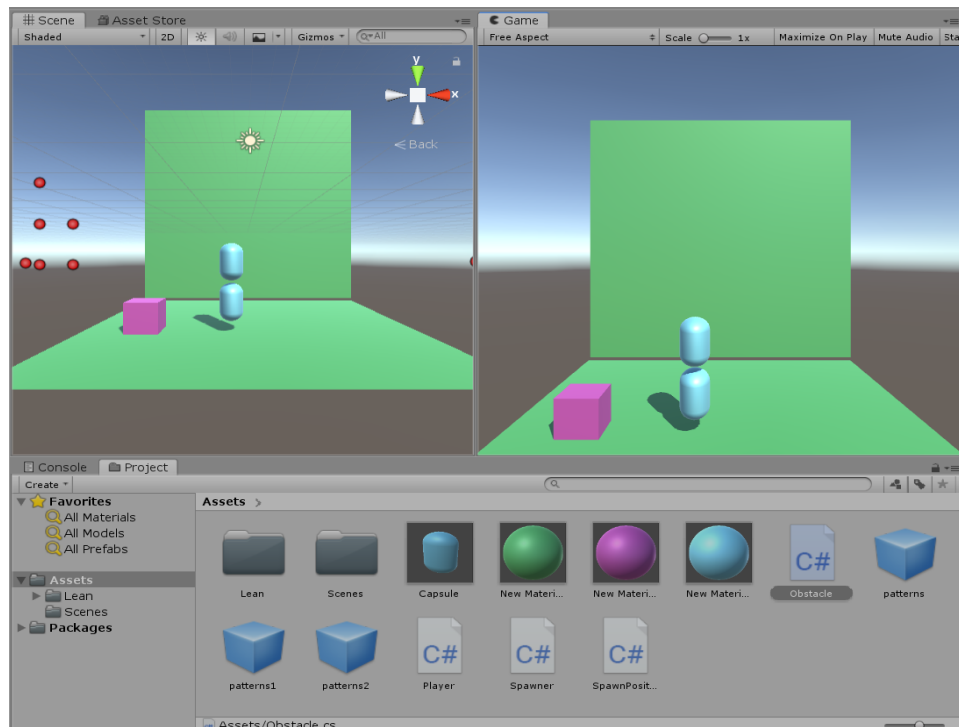




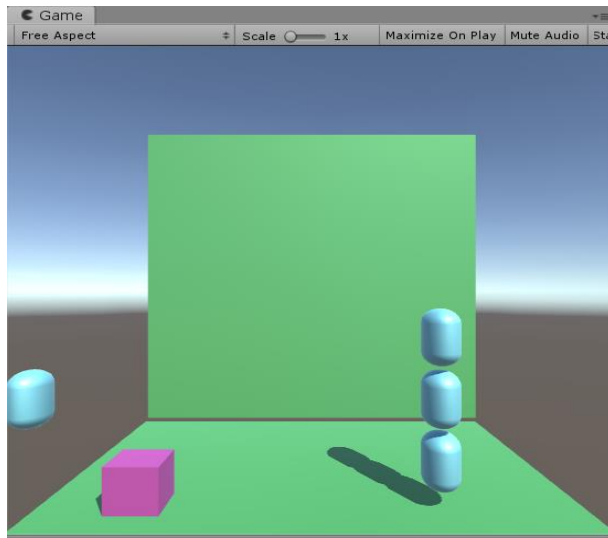
## Second untitled work



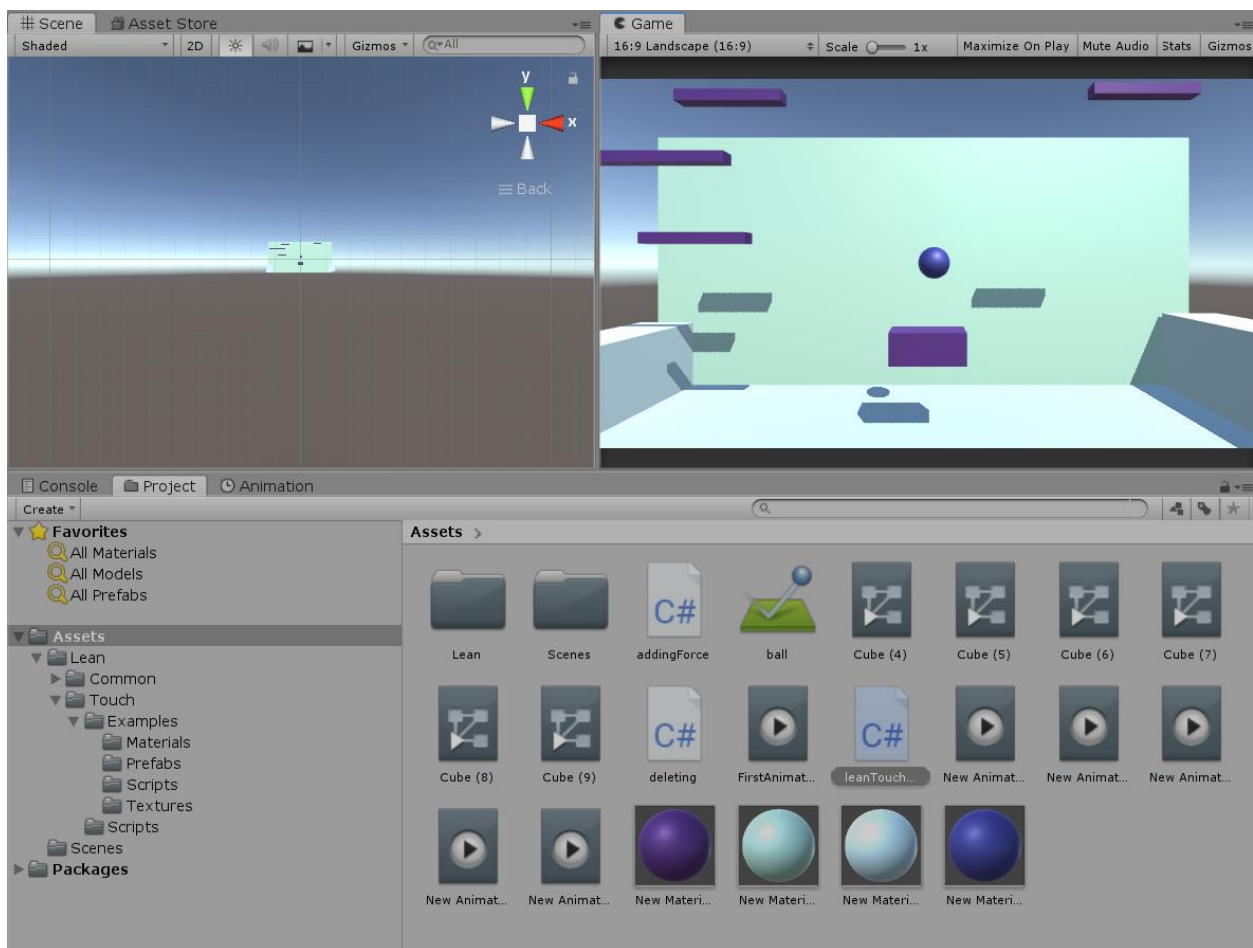
## Third project

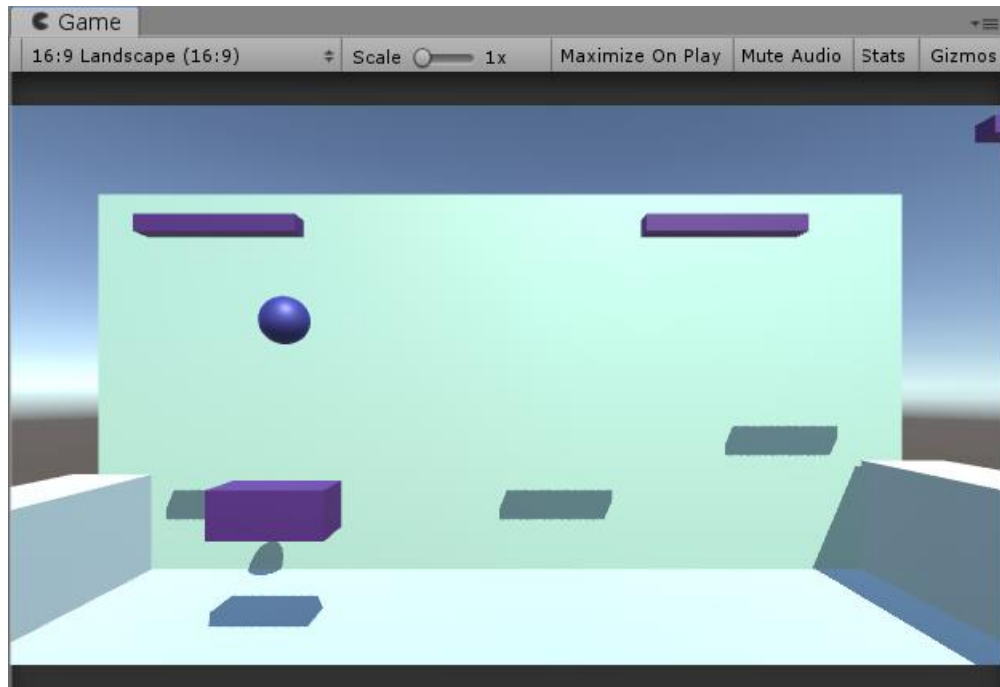






Forth prototype





LogGame

