

## Avoid Viruses: Development of 3D Arcade Educational Game

Chotibul Umam Wiranda<sup>1</sup>, Giri Wahyu Wiriasto<sup>1</sup>

<sup>1</sup>Jurusan Teknik Elektro – Universitas Mataram, 83127 – Lombok, Indonesia

### ARTICLE INFO

#### Article history:

Received October 19, 2023

Revised October 30, 2023

Accepted October 30, 2023

#### Keywords:

Keyword-1; Pengembangan Game

Edukasi

Keyword-2; Unity 3D

Keyword-3; Virus

Keyword-4; Kesadaran kesehatan

### ABSTRACT

This research aims to develop an educational game using Unity 3D that focuses on avoiding viruses falling from above. The game is designed to provide an enjoyable playing experience while imparting an understanding of the importance of maintaining health and preventing virus spread. We utilize Unity 3D as the main development platform and implement gameplay mechanics that involve the player character avoiding viruses descending from the top of the screen. Development methods encompass level design, character control implementation, and providing visual and audio feedback to the player. Evaluation is conducted through internal testing and gathering feedback from users. The results indicate that the game successfully delivers an engaging experience while also enhancing users' understanding of the importance of virus prevention. This research contributes to the development of interactive educational games aimed at promoting health awareness.

### Corresponding Author:

Giri Wahyu Wiriasto, Jurusan Teknik Elektro Universitas Mataram, Jalan Majapahit 63 Kota Mataram, 83127 - Lombok, Indonesia

Email: [giriwahyuwiriasto@unram.ac.id](mailto:giriwahyuwiriasto@unram.ac.id)

## 1. PENDAHULUAN

Salah satu bentuk edukasi publik yang cukup efektif pada masa kini adalah dengan memanfaatkan media permainan digital atau yang lazim disebut dengan game atau gim. Video game atau gim adalah bentuk permainan yang menggunakan media elektronik berupa layar sebagai wadah visualisasi untuk melakukan berbagai interaksi yang berkaitan dengan permainan tersebut. Pada zaman modern ini, gim telah menjadi bagian dari sarana hiburan maupun pembelajaran bagi berbagai kalangan[10]. Gim yang pada mulanya bertujuan untuk menghibur dan mengisi waktu luang akan tetapi gim dapat menjadi lebih bermanfaat jika tema dari game tersebut mengandung pesan yang mendidik[6]. Game berasal dari kata bahasa Inggris yang memiliki arti dasar “Permainan”[7]. Games merupakan permainan elektronik yang melibatkan interaksi antarmuka dengan pengguna untuk menghasilkan umpan balik secara visual pada perangkat komputer[2]. Dalam beberapa tahun terakhir, penggunaan game edukasi telah mendapatkan perhatian yang signifikan sebagai pendekatan yang efektif untuk melibatkan dan mendidik para pembelajar di berbagai bidang. Pengalaman interaktif dan mendalam ini memberikan platform yang menarik untuk menyampaikan konten pendidikan sambil tetap menjaga minat dan motivasi pengguna. Dengan meningkatnya kekhawatiran tentang kesehatan masyarakat dan perlunya meningkatkan kesadaran tentang pencegahan virus, pengembangan game edukasi yang mengatasi topik ini menjadi semakin relevan.

Studi ini bertujuan untuk mengeksplorasi pengembangan sebuah game edukasi menggunakan Unity 3D, sebuah platform pengembangan game populer, dengan fokus pada tema menghindari virus yang jatuh dari atas. Game ini berusaha menggabungkan unsur hiburan dan pembelajaran, memberikan pemain pengalaman yang menyenangkan sambil menyampaikan pengetahuan tentang pentingnya menjaga kesehatan dan mencegah penyebaran virus. Agar pembuatan game dapat tercapai, pembuat game memerlukan tools yang dapat digunakan sebagai media pembantu dalam pembuatan game. Salah satu jenis tools yang digunakan dalam pembuatan game adalah game engine, unity[1]. Unity salah satu game engine yang mudah digunakan, hanya membuat objek dan diberikan fungsi untuk menjalankan objek tersebut[9]. Dalam setiap objek mempunyai variabel, variabel inilah yang harus dimengerti supaya dapat membuat

game yang berkualitas. Unity merupakan game engine yang mendukung grafis secara 2D maupun 3D, yang pada umumnya menggunakan bahasa pemrograman C#[8]. Dengan menggunakan Unity 3D, kami dapat memanfaatkan kemampuannya untuk menciptakan lingkungan yang menarik secara visual dan interaktif yang melibatkan para pemain dan meningkatkan pemahaman mereka tentang strategi pencegahan virus.

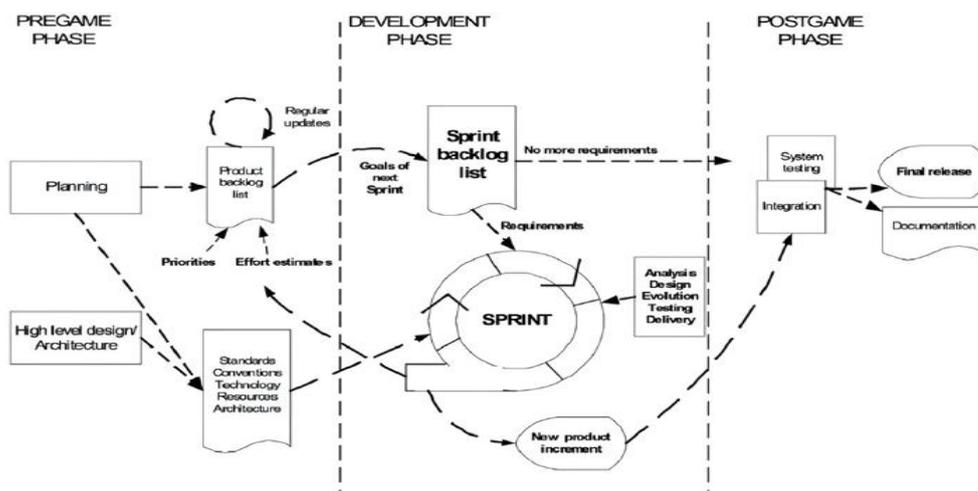
Proses pengembangan akan melibatkan desain level permainan yang menyajikan tantangan terkait menghindari virus yang jatuh, implementasi kontrol yang intuitif untuk karakter pemain, dan penggunaan umpan balik visual dan audio untuk meningkatkan pengalaman bermain. Selain itu, game ini akan dievaluasi melalui pengujian internal dan pengumpulan umpan balik dari pengguna untuk memastikan keefektifannya dalam mencapai tujuan pendidikan.

Hasil penelitian ini memiliki potensi yang signifikan untuk mempromosikan kesadaran kesehatan dan menyediakan pendekatan inovatif dalam mendidik individu tentang strategi pencegahan virus. Dengan memanfaatkan kekuatan game edukasi dan kemampuan Unity 3D, penelitian ini bertujuan untuk memberikan kontribusi dalam bidang teknologi pendidikan interaktif sambil mengatasi kebutuhan mendesak dalam menyebarkan pengetahuan tentang strategi pencegahan virus secara menarik dan menyenangkan.

## 2. METODE

### 2.1. Desain dan Pengembangan Game

Scrum telah menunjukkan hasil yang baik dalam pengembangan game, ada beberapa pengembang sekarang lebih memilih scrum daripada metodologi lainnya karena keefektifan hasilnya. Scrum klasik proses akan terlibat dengan tiga proses dimulai dari pregame (fase perencanaan), game (fase pengembangan), dan postgame seperti yang ditunjukkan pada gambar 1



Gambar 1 Metode agile scrum

Scrum justru merupakan evolusi dari Agile Management. Metodologi scrum didasarkan pada serangkaian praktik dan peran yang sangat jelas yang harus dilibatkan selama proses pengembangan perangkat lunak. Ini adalah metodologi fleksibel yang menghargai penerapan 12 prinsip tangkas dalam konteks yang disetujui oleh semua anggota tim produk. Scrum dijalankan dalam blok-blok sementara yang pendek dan periodik, disebut Sprint, yang biasanya berkisar antara 2 hingga 4 minggu, yang merupakan istilah untuk umpan balik dan refleksi. Setiap Sprint adalah entitas tersendiri, yaitu memberikan hasil yang lengkap, variasi dari produk akhir yang harus dapat dikirimkan ke klien dengan upaya seminimal mungkin saat diminta. Proses tersebut memiliki titik awal, daftar tujuan/persyaratan yang membentuk rencana proyek. pada gambar metode scrum diatas terdapat tiga phase yaitu:

1. Pregame Phase: Fase pra-pengembangan di mana visi dan kebutuhan game ditentukan, tim pengembangan terbentuk, dan rencana awal proyek dibuat.
2. Development Phase: Fase utama di mana tim bekerja dalam siklus sprint untuk mengembangkan game secara iteratif dan inkremental, termasuk perencanaan sprint, pengembangan, pengujian, dan ulasan sprint.
3. Post-game Phase: Fase pasca-pengembangan di mana game dirilis, didistribusikan, dan mendapat umpan balik. Melibatkan peluncuran, evaluasi, dukungan, dan pemeliharaan game[4].

Tabel 1 ABC Sprint

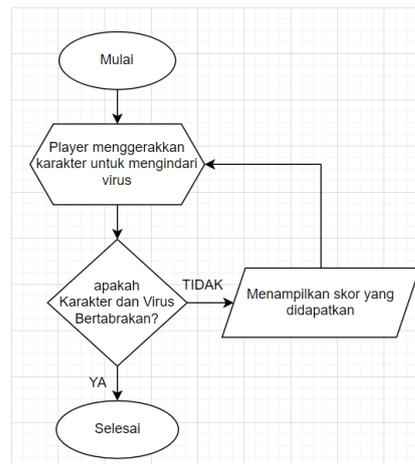
ABC Sprint	Alpha Sprint (Bulan Februari-Maret)	Membuat konsep dan ide dalam game, pencarian asset awal
	Beta Sprint (Bulan April-Mei)	Asset final, pembuatan game dan melengkapi fitur-fitur
	Completion Sprint (Bulan Juni)	Tahap dimana permainan telah selesai dengan bug-free dan siap untuk dimainkan
	Completion Sprint (Bulan Juni)	Tahap dimana permainan telah selesai dengan bug-free dan siap untuk dimainkan

Dalam pengembangan game avoid viruses: game edukasi 3D Unity berbasis arcade ini saya menggunakan metodologi scrum dengan tiga sprint, dengan waktu pengerjaan 1 semester atau 5-6 bulan. Tiga sprint yaitu Alpha Sprint, Beta Sprint, dan Penyelesaian Sprint dengan masing-masing detail seperti pada tabel 1.

1. Alpha Sprint, yaitu tahap membuat konsep dan ide dan mencari asset awal serta memahami dasar dasarnya fungsionalitas yang dibutuhkan dalam pengembangan game juga membuat backlog untuk melakukannya. Tahap ini saya lakukan dari bulan february hingga maret
2. Beta Sprint, di mana tahap ini saya sudah mulai membuat game 3D Unity lengkap fitur dengan aset yang telah di dapatkan dan siap untuk dilakukan tahap pengujian game tersebut apakah masih ada yang kurang atau tidak. Tahap ini saya lakukan dari bulan April hingga bulan mei
3. Completion Sprint, tahap dimana permainan telah selesai dengan bug-free dan siap untuk dimainkan oleh pengguna. Tahap ini adalah tahap penyempurnaan yaitu dilakukan pada bulan juni.

### 3. Game Skenario

1. Narasi: Pemain memasuki dunia virtual yang terinfeksi oleh virus misterius yang dapat menyebabkan wabah mematikan. Sebagai seorang ilmuwan terkemuka, tugas pemain adalah menghentikan penyebaran virus dengan menghindari virus-virus yang jatuh dari atas dengan pola yang acak.
2. Mekanika Permainan:  
Pemain mengendalikan karakter ilmuwan berani menggunakan kontroler permainan untuk bergerak bebas di lingkungan game 3D. Virus-virus akan jatuh dari langit dengan pola yang acak dan pemain harus menghindari mereka dengan menggerakkan karakter ke tempat yang aman.
3. Peningkatan Kesulitan:  
Seiring pemain melaju melalui level-level permainan, tingkat kesulitan akan meningkat secara bertahap. Virus-virus akan jatuh dengan kecepatan yang lebih tinggi, pola yang lebih kompleks, dan variasi karakteristik yang lebih menantang. Terdapat variasi virus dengan karakteristik berbeda, seperti virus yang bergerak dengan cepat dan virus yang bergerak lambat
4. Skor dan Tujuan:  
Setiap kali pemain berhasil menghindari atau melawan virus, mereka akan mendapatkan poin skor dan kemajuan dalam misi penyelamatan. Tujuan utama pemain adalah menghindari virus yang jatuh dengan kecepatan random dan mencapai skor tertinggi.



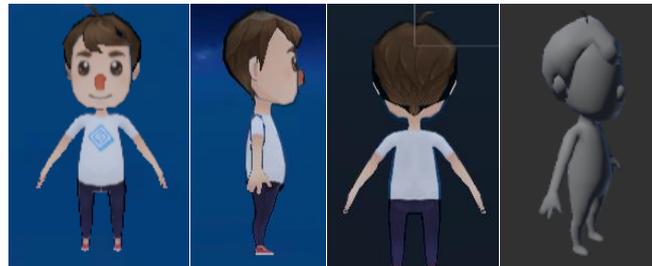
Gambar 2. Flowchart Alur Permainan

Pada gambar 2 merupakan alur permainan dari game avoid viruses. Saat permainan dimulai player akan menggerakkan karakter untuk menghindari virus yang jatuh, jika karakter dan virus tidak bertabrakan maka akan menampilkan skor yang didapatkan dan jika karakter bertabrakan dengan virus maka permainan akan selesai.

#### 4. HASIL DAN PEMBAHASAN

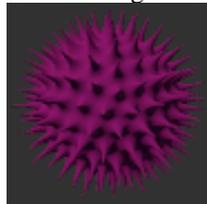
##### 4.1. Hasil:

Dalam pengembangan game ini, kami berhasil menciptakan sebuah game edukasi yang menggabungkan elemen-elemen hiburan dengan pembelajaran tentang virus. Game ini didesain menggunakan teknologi Unity 3D yang memberikan grafis yang memukau dan lingkungan game yang realistis. Pemain akan mengendalikan karakter ilmuwan berani yang bertugas menghindari virus-virus yang jatuh dari atas untuk mencegah penyebaran wabah. Selain itu, pemain juga akan memperoleh poin setelah berhasil menghindari virus-virus yang jatuh. Game ini menawarkan pengalaman bermain yang seru sekaligus mendidik, dengan tingkat kesulitan yang meningkat secara bertahap dan tantangan yang menantang.



Gambar 3. Tampilan Player

Pada gambar 3 dapat dilihat bahwa gambar tersebut merupakan tampilan player[5]. Player[5] ini yang akan menghindari virus-virus[8] yang jatuh dari atas dengan kecepatan jatuh yang random.



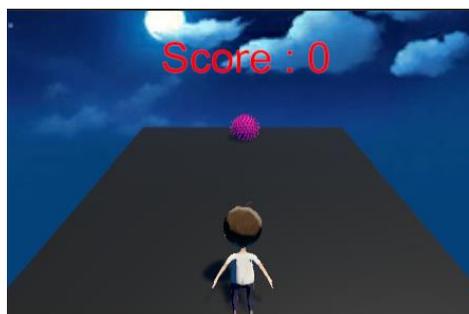
Gambar 4. Tampilan Virus

Pada Gambar 4 dapat dilihat bahwa gambar tersebut merupakan tampilan virus[5]. Virus[5] ini yang akan jatuh dengan kecepatan random dari atas untuk dihindari oleh player.

A red digital score display showing "Score : 0" against a dark blue background.

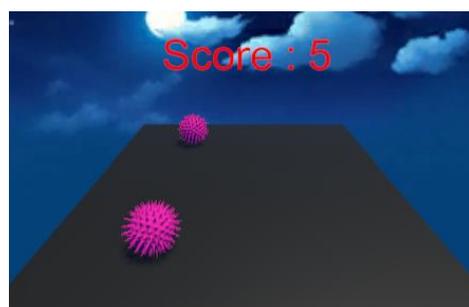
Gambar 5. Tampilan Score

Pada gambar 5 dapat dilihat bahwa gambar tersebut merupakan tampilan score. Score ini merupakan nilai yang didapat player dalam menghindari virus yang jatuh, dimana 1 virus yang berhasil dilewati dihitung 1 score.



Gambar 6. Tampilan Gameplay

Pada gambar 6 dapat dilihat bahwa gambar tersebut merupakan tampilan gameplay. Tahap gameplay ini merupakan sebuah proses dimana karakter ilmuwan[5] akan menghindari virus-virus yang jatuh dari atas dengan kecepatan jatuh yang random. Pada antarmuka ini terdapat tampilan score yang merupakan jumlah dari virus yang berhasil dihindari.



Gambar 7. Tampilan Game Over

Pada gambar 7 dapat dilihat bahwa gambar tersebut merupakan tampilan “game over” yang artinya game telah berakhir, pada tampilan antarmuka ini terdapat total skor yang telah didapatkan dan untuk memulai ulang game (merestart game) player cukup menekan tombol R yang ada di keyboard.

#### 4.2. Pembahasan:

##### a. Edukasi tentang Virus:

Dengan memperkenalkan virus-virus yang jatuh dengan kecepatan random, pemain akan dihadapkan pada situasi yang dinamis dan perlu mengambil keputusan dengan cepat. Ini dapat melatih keterampilan reaksi dan koordinasi motorik pemain dalam menghindari virus-virus tersebut. Pemain perlu mengantisipasi arah dan kecepatan jatuh virus agar karakter ilmuwan dapat menghindarinya dengan sukses.

Selain itu, penggunaan kecepatan yang random pada virus juga mencerminkan sifat yang realistis dari penyebaran virus dalam kehidupan nyata. Dalam situasi nyata, virus dapat menyebar dengan kecepatan yang bervariasi, tergantung pada faktor-faktor seperti lingkungan dan tingkat interaksi sosial. Dengan memperlihatkan kecepatan jatuh yang random pada virus di dalam game, pemain dapat memahami bahwa penyebaran virus tidak selalu terprediksi dan perlu diwaspadai.

##### b. Penggunaan Teknologi Unity 3D:

Dalam pengembangan game ini, kami memilih Unity 3D sebagai platform utama karena kemampuannya dalam menciptakan lingkungan game yang realistis dan menarik. Teknologi ini memungkinkan kami untuk membuat grafis yang memukau, efek visual yang kaya, dan suara yang realistis. Hal ini memberikan pengalaman bermain yang lebih mendalam dan imersif bagi pemain. Selain itu, Unity

3D juga memberikan fleksibilitas dalam mengatur gameplay, tingkat kesulitan, dan mekanika permainan yang dapat disesuaikan dengan kebutuhan.

c. Mekanika Permainan yang Menarik:

Game ini menawarkan mekanika permainan yang menarik dan seru untuk pemain. Pemain akan mengendalikan karakter ilmuwan berani yang harus menghindari virus-virus yang jatuh dari atas. Mekanika permainan ini melibatkan keterampilan motorik, ketepatan gerakan, dan pengambilan keputusan yang cepat. Selain menghindari virus, pemain juga dapat mengumpulkan power-up dan item yang memberikan keuntungan, seperti perlindungan tambahan atau peningkatan kesehatan. Dengan adanya variasi dan kompleksitas mekanika permainan ini, pemain akan terus tertantang dan terlibat dalam permainan.

d. Tingkat Kesulitan yang Meningkat:

Game ini dirancang dengan tingkat kesulitan yang meningkat secara bertahap. Pada awal permainan, pemain akan menghadapi virus-virus yang lebih mudah dihindari, namun seiring berjalannya permainan, virus-virus yang muncul akan semakin cepat dan lebih sulit untuk dihindari. Hal ini memberikan tantangan yang berkelanjutan bagi pemain dan mendorong mereka untuk meningkatkan keterampilan bermain mereka. Dengan adanya tingkat kesulitan yang meningkat, pemain akan merasa terus-menerus tertantang dan terlibat dalam permainan.

e. Dampak Positif dan Relevansi:

Game ini memiliki dampak positif yang signifikan dalam mendidik pemain tentang virus dan kebersihan. Melalui pengalaman bermain yang interaktif, pemain dapat memahami konsep-konsep penting tentang virus, menjaga kebersihan, dan tindakan pencegahan yang perlu dilakukan. Selain itu, game ini juga relevan dengan situasi dunia nyata, terutama dalam konteks pandemi global yang sedang berlangsung. Dengan memperhatikan aspek edukasi dan relevansi ini, diharapkan game ini dapat memberikan kontribusi positif dalam meningkatkan kesadaran masyarakat akan pentingnya menjaga kesehatan dan kebersihan.

f. Algoritma Game

1. Program Gerakan Player

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class GerakanPlayer : MonoBehaviour
{
    public float kecepatan;
    Rigidbody rb;
    Animator anim;

    public Transform playerPutaran;

    private void Awake()
    {
        rb = GetComponent<Rigidbody>();
        anim = GetComponent<Animator>();
    }
    // Start is called before the first frame update
    void Start()
    {
    }

    // Update is called once per frame
    void FixedUpdate()
    {
        Bergerak();
    }

    void Bergerak()
    {
        float gerak = Input.GetAxis("Horizontal");
        rb.velocity = Vector3.right * gerak * kecepatan;
        anim.SetFloat("Kecepatan", Mathf.Abs(gerak), 0.1f, Time.deltaTime);
        playerPutaran.localEulerAngles = new Vector3(0, gerak * 90, 0);
    }
}
```

```
private void OnCollisionEnter(Collision collision)
{
    if (collision.collider.CompareTag("virus"))
    {
        Time.timeScale = 0;
        Destroy(gameObject);
    }
}
```

Program di atas adalah sebuah skrip dalam bahasa pemrograman C# yang digunakan untuk mengendalikan gerakan seorang pemain dalam permainan menggunakan Unity. Skrip ini terdiri dari beberapa bagian utama. Pertama, terdapat deklarasi dan inisialisasi variabel seperti `kecepatan` yang digunakan untuk mengatur kecepatan pemain, `rb` yang merupakan komponen Rigidbody objek pemain, dan `anim` yang merupakan komponen Animator objek pemain.

Selanjutnya, terdapat metode `Awake()` yang dipanggil saat objek pemain pertama kali dibuat di dalam permainan. Metode ini menginisialisasi variabel `rb` dengan komponen Rigidbody yang terpasang pada objek pemain, serta variabel `anim` dengan komponen Animator yang terpasang pada objek pemain.

Metode `FixedUpdate()` merupakan metode yang dipanggil secara berkala oleh mesin fisika Unity dengan interval waktu yang konsisten. Dalam metode ini, terdapat pemanggilan metode `Bergerak()`, yang bertanggung jawab untuk mengendalikan gerakan pemain berdasarkan input horizontal yang diterima dari pemain.

Metode `Bergerak()` mengatur gerakan pemain dengan mengambil input horizontal dari pemain menggunakan fungsi `Input.GetAxis()`. Kecepatan pemain diatur berdasarkan input horizontal tersebut, dan parameter "Kecepatan" pada animator pemain diatur untuk memicu animasi berdasarkan kecepatan pemain. Selain itu, metode ini juga mengubah rotasi lokal objek pemain sesuai dengan input horizontal, sehingga mengubah orientasi visual pemain.

Terakhir, terdapat metode `OnCollisionEnter(Collision collision)` yang dipanggil ketika terjadi tabrakan dengan objek lain yang memiliki kolider. Dalam metode ini, terdapat pemeriksaan apakah objek yang ditabrak memiliki tag "virus". Jika objek tersebut memiliki tag tersebut, maka permainan akan dihentikan dengan mengatur `Time.timeScale = 0` untuk menghentikan waktu permainan, dan objek pemain akan dihancurkan.

Dengan menggunakan skrip ini, pemain dapat mengendalikan gerakan karakter dalam permainan dan akan mengalami kekalahan jika bertabrakan dengan objek yang memiliki tag "virus".

## 2. Program Munculin Virus

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class MunculinVirus : MonoBehaviour
{
    public GameObject virus;
    public float waktuMinimal, waktuMaximal;
    public float posisiMinimal, posisiMaximal;
    // Start is called before the first frame update
    void Start()
    {
        StartCoroutine(MunculVirus());
    }

    // Update is called once per frame
    void Update()
    {
    }

    IEnumerator MunculVirus()
    {
        Instantiate(virus, transform.position + Vector3.right *
Random.Range(posisiMinimal, posisiMaximal), Quaternion.identity);
yield return new WaitForSeconds(Random.Range(waktuMinimal, waktuMaximal));
    }
}
```

```

        StartCoroutine (MunculVirus ());
    }
}

```

Program di atas merupakan sebuah skrip dalam bahasa pemrograman C# yang digunakan untuk mengatur munculnya objek "virus" secara acak dalam permainan Unity. Skrip ini terdiri dari beberapa bagian utama. Pertama, terdapat deklarasi variabel seperti `virus` yang digunakan untuk menyimpan objek virus yang akan dibuat, `waktuMinimal` dan `waktuMaximal` yang digunakan untuk mengatur rentang waktu antara munculnya virus, serta `posisiMinimal` dan `posisiMaximal` yang mengatur rentang posisi horizontal di mana virus akan muncul.

Metode `Start()` dipanggil saat objek yang mengandung skrip ini pertama kali aktif di dalam permainan. Di dalam metode ini, Coroutine `MunculVirus()` dijalankan untuk memicu munculnya virus secara berulang dengan waktu yang diatur.

Selanjutnya, terdapat metode `Update()`, yang dipanggil setiap frame dalam permainan. Dalam program ini, metode ini tidak digunakan.

Coroutine `MunculVirus()` bertanggung jawab untuk mengatur munculnya objek virus secara acak. Pada awal coroutine, objek virus di-instantiate di posisi transform saat ini ditambahkan dengan nilai acak di antara `posisiMinimal` dan `posisiMaximal`. Rotasi objek diatur menjadi identitas menggunakan `Quaternion.identity`. Setelah itu, coroutine menunggu selama waktu acak yang ditentukan oleh rentang `waktuMinimal` dan `waktuMaximal` menggunakan `yield return new WaitForSeconds()`. Setelah waktu tertunda selesai, coroutine memanggil dirinya sendiri lagi dengan `StartCoroutine(MunculVirus())`. Hal ini mengulangi proses munculnya virus secara terus-menerus.

Dengan menggunakan skrip ini, objek "virus" akan muncul secara acak dalam permainan dengan posisi horizontal yang bervariasi dalam rentang yang ditentukan. Waktu antara munculnya setiap virus juga akan bervariasi dalam rentang waktu yang ditentukan, menciptakan elemen kejutan dan tantangan dalam permainan.

### 3. Program Menampilkan Score

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class ScoreScript : MonoBehaviour
{
    public float score;
    public Text scoreUI;
    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {

    }

    private void OnCollisionEnter(Collision collision)
    {
        if (collision.collider.CompareTag("virus"))
        {
            score += 1;
            scoreUI.text = "Score : " + score.ToString();
            Destroy(collision.collider.gameObject);
        }
    }
}

```

Program di atas merupakan skrip dalam bahasa pemrograman C# yang digunakan untuk mengatur sistem skor dalam permainan Unity. Skrip ini memiliki dua komponen utama. Pertama, terdapat deklarasi variabel seperti `score` yang digunakan untuk menyimpan nilai skor pemain, dan `scoreUI` yang merupakan referensi ke komponen UI Text yang akan menampilkan skor di antarmuka pengguna.

Metode `Start()` akan dipanggil saat objek yang mengandung skrip ini pertama kali aktif di dalam permainan. Namun, dalam program ini, metode `Start()` tidak digunakan dan tidak berisi kode.

Metode `Update()` juga tidak digunakan dalam program ini. Metode ini akan dipanggil setiap frame dalam permainan, tetapi karena tidak ada kode di dalamnya, metode ini tidak melakukan apa pun.

Metode `OnCollisionEnter(Collision collision)` akan dipanggil ketika terjadi tabrakan dengan objek lain yang memiliki kolider. Dalam metode ini, terdapat pemeriksaan apakah objek yang ditabrak memiliki tag "virus". Jika objek tersebut memiliki tag tersebut, maka skor pemain akan bertambah 1 dengan `score += 1`. Selanjutnya, nilai skor yang baru akan ditampilkan di UI Text dengan mengubah teks pada `scoreUI` menggunakan `scoreUI.text = "Score : " + score.ToString()`. Terakhir, objek yang ditabrak akan dihancurkan menggunakan `Destroy(collision.collider.gameObject)`.

Dengan menggunakan skrip ini, saat pemain melakukan tabrakan dengan objek yang memiliki tag "virus", skor pemain akan bertambah dan akan ditampilkan di antarmuka pengguna melalui komponen UI Text. Skrip ini memungkinkan pemain untuk melacak dan menampilkan skor saat bermain permainan yang melibatkan tabrakan dengan objek "virus".

#### 4. Program Restart Game

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Restart : MonoBehaviour
{
    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        if (Input.GetKeyDown(KeyCode.R))
        {
            Time.timeScale = 1;
            SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex);
        }
    }
}
```

Program di atas adalah sebuah skrip dalam bahasa pemrograman C# yang digunakan untuk mengatur mekanisme restart dalam permainan menggunakan Unity. Skrip ini terdiri dari beberapa komponen utama. Metode `Start()` dipanggil saat objek yang mengandung skrip ini pertama kali aktif di dalam permainan. Namun, dalam program ini, metode `Start()` tidak digunakan dan tidak berisi kode. Metode `Update()` dipanggil setiap frame dalam permainan. Di dalam metode ini, terdapat pemeriksaan kondisi dengan menggunakan `Input.GetKeyDown(KeyCode.R)`. Jika tombol "R" pada keyboard ditekan, maka kondisi tersebut akan terpenuhi.

Jika kondisi terpenuhi, maka dilakukan beberapa tindakan. Pertama, `Time.timeScale` diatur kembali menjadi 1 untuk mengembalikan aliran waktu permainan menjadi normal, jika sebelumnya diatur ke 0 untuk memberhentikan waktu. Selanjutnya, `SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex)` digunakan untuk memuat kembali atau restart scene yang sedang aktif. Dengan menggunakan `SceneManager.GetActiveScene().buildIndex`, kita mendapatkan indeks scene yang sedang aktif, dan menggunakan metode `LoadScene()` untuk memuat kembali scene tersebut.

Dengan menggunakan skrip ini, ketika pemain menekan tombol "R", permainan akan di-restart dengan mengembalikan aliran waktu permainan menjadi normal dan memuat kembali scene yang sedang aktif. Skrip ini memberikan kemudahan bagi pemain untuk melakukan restart permainan dengan cepat dan mudah tanpa harus keluar dan masuk kembali ke dalam permainan.

#### 5. KESIMPULAN

Pertama, game ini menyajikan edukasi tentang virus dan kebersihan melalui gameplay yang melibatkan penghindaran virus yang jatuh dengan kecepatan yang acak. Konsep ini memberikan pemahaman kepada

pemain tentang pentingnya menjaga kebersihan dan melindungi diri dari virus. Dengan demikian, game ini tidak hanya menjadi hiburan, tetapi juga berfungsi sebagai alat edukasi yang bermanfaat.

Kedua, penggunaan Teknologi Unity 3D dalam pengembangan game ini memberikan keuntungan yang signifikan. Dengan fitur visual yang luar biasa, animasi yang halus, dan fleksibilitas pengembangan yang ditawarkan oleh Unity 3D, game ini mampu menciptakan pengalaman bermain yang lebih mendalam, imersif, dan memukau bagi para pemain. Keunggulan teknologi ini juga memungkinkan pengembang untuk mengatur gameplay, tingkat kesulitan, dan mekanika permainan yang dapat disesuaikan dengan kebutuhan.

Ketiga, game ini telah diperbarui dengan tema langit malam yang memukau, menambahkan daya tarik visual yang menarik bagi pemain. Karakter player juga telah ditingkatkan kemampuannya sehingga dapat bergerak secara vertikal, memungkinkan pemain untuk menghindari virus yang jatuh dengan lebih lancar dan menantang. Selain itu, fitur skor juga telah ditambahkan, memungkinkan pemain untuk melihat dan mencatat jumlah virus yang berhasil dilewati. Meskipun demikian, game ini masih memiliki potensi untuk pengembangan lebih lanjut. Beberapa fitur yang dapat ditambahkan antara lain penambahan level yang meningkatkan tingkat kesulitan permainan seiring kemajuan pemain, pilihan karakter yang memberikan variasi gameplay, penambahan peta baru yang menawarkan tantangan baru, serta peningkatan fitur lainnya yang dapat meningkatkan pengalaman bermain. Sejauh ini, dalam game ini sudah termasuk karakter player, virus, dan jalanan yang menjadi komponen penting dalam permainan.

#### DAFTAR PUSTAKA

- [1] Shagiarto Ibzani Ilham Shagiarto, Giri Wahyu Wiriasto, Djul Fikry Budiman, Misbahuddin. (2023). "Aplikasi Game berbasis Andorid 2D dengan Logika Fuzzy pada NPC (Non-Player Character)". *Journal of Electrical Engineering and Information technology Department of Electrical Engineering University of Mataram* Vol. 1, No. 1. pp. 41. <https://journal.unram.ac.id/index.php/jeitech/article/view/2673/1231>
- [2] Fikriansyah Muhammad, Giri Wahyu Wiriasto, A. Sjamsjiar Rachman. (2023). "NON-PLAYER CHARACTER IN FIRE FIGHTER GAMES USING GENETIC ALGORITHM". *Jurnal MNEMONIC*. Vol 6 No 1. Pp. 11
- [3] Unity, Game Developer engine, <https://developer.android.com/games/develop/build-in-unity?hl=id&authuser=1>. Accessed 15.06.2023
- [4] D. Prasetya and S. Dian, "Game Development with Scrum methodology", doi: 10.1109/AIT49014.2019.9144963.
- [5] Aset, Avoid Viruses: Pengembangan Game Edukasi 3D Unity Berbasis Arcade., <http://www.mediafire.com/file/poy49yqlqf2wwr/file>
- [6] Fenroy Yedithia, S. R. (2022). "Pengembangan Game Platformer "Ayo Lawan Virus" Berbasis Android dengan Menggunakan Game Engine Unity". *Jurnal Humaniora Teknologi* Volume 8, Nomor 1, page 1.
- [7] Indah Rohmawati, S. I. (2019). "PENGEMBANGAN GAME EDUKASI TENTANG BUDAYA NUSANTARA "TANARA" MENGGUNAKAN UNITY 3D BERBASIS ANDROID". *Jurnal SITECH*, Vol 2, No 2. Page 174
- [8] Joshua Lorenzo Andre, E. H. (2018). Pengembangan Game Virtual Reality Berbasis Android Menggunakan Unity Sebagai Media Penunjang Pengenalan Bahasa Inggris. *J-INTECH* Volume 06 Nomor 02. Page 208.
- [9] Tjahyadi Michello Pratama, Alicia Sinsuw, Virginia Tulenan, Steven Sentinuwo. (2014). "Prototipe Game Musik Bambu Menggunakan Engine Unity 3D". *E-journal Teknik Informatika*, Volume 4, No. 2. Page 1.
- [10] Asyikin Arifin Noor, Ali Watoni. (2019). "JUKUNG BANJAR" GAME ARCADE 2D BERBASIS WEB". *LENTERA Jurnal Ilmiah Kependidikan* Vol. 14 No. 2. Page 8. <https://jurnal.stkipbjm.ac.id/index.php/jpl/article/view/882/411>
- [11] Wiranda, Cotibul Umam. (2023). <https://github.com/otepumam/gameUnityAvoidViruses> Date Accessed: 26 June 2023.