



Design and Development of the Moving Score Virtual Reality Game as a Game Programmer

Muhammad Ridho Maghribi¹, Geovanne Farell², Sheanny Ocmi Sakti³, Wiki Lofandri⁴

^{1,2,3,4}Program Studi Animasi, Sekolah Vokasi, Universitas Negeri Padang

e-mail: ridhomaghribi123@gmail.com, geovannefarell@ft.unp.ac.id, sheannyos@ft.unp.ac.id, wiloleaks@unp.ac.id

ABSTRACT

The development of Virtual Reality (VR) technology opens up new opportunities to create exciting gameplay experiences while encouraging players to be more physically active. This study aims to design a VR-based game called *Moving Score*, focusing on the development of motion-based interaction systems and the implementation of responsive game mechanics. The game was developed using Unity and the XR Interaction Toolkit, featuring various elements such as a dynamic scoring system, visual and audio feedback, and real-time motion detection. The method used is prototyping, starting from needs identification, prototype development, to testing and iteration. The final results show that the game successfully delivers intuitive interaction, provides effective feedback, and enhances user engagement in a virtual environment. *Moving Score* not only offers entertainment but also holds potential as an educational medium that supports users' physical activity and movement coordination.

Keywords: Virtual Reality, Unity, XR Interaction Toolkit, Game Programming, Scoring System, Educational Game.

PENDAHULUAN

Game merupakan salah satu bentuk hiburan interaktif yang terus berkembang di dunia modern. Sebagai medium yang populer, *game* tidak hanya memberikan kesenangan tetapi juga memiliki potensi besar dalam mendukung pendidikan dan pengembangan keterampilan. Menurut Juul dalam *Half-Real: Video Games between Real Rules and Fictional Worlds* (2005) menjelaskan bahwa *game* menggabungkan aturan-aturan yang terstruktur dengan elemen naratif untuk menciptakan pengalaman yang mendalam dan bermakna bagi pemainnya. Dalam konteks ini, *game* dapat menjadi alat eksplorasi sekaligus pembelajaran.

Virtual Reality membawa pengalaman bermain *game* ke tingkat yang lebih tinggi melalui teknologi yang memungkinkan interaksi langsung dengan dunia virtual yang realistis. Teknologi ini menawarkan simulasi imersif yang memungkinkan pengguna untuk belajar melalui pengalaman. Sherman dan Craig dalam *Understanding Virtual Reality* (2003) menyebutkan bahwa *Virtual Reality* dapat menciptakan lingkungan belajar yang efektif sekaligus menyenangkan. Potensi ini menjadikan *Virtual Reality* sebagai alat yang menjanjikan untuk mendukung berbagai bidang, termasuk pendidikan.

Game Moving Score merupakan inovasi yang berfokus pada interaksi pemain dengan objek dalam dunia virtual untuk memperoleh skor. Sistem ini memungkinkan pemain untuk melakukan berbagai aksi, seperti menangkap, menghindari, atau mengumpulkan objek dalam waktu tertentu guna mendapatkan nilai terbaik. Menurut Gee dalam *What Video Games Have to Teach Us About Learning and Literacy* (2003), game yang dirancang dengan baik dapat meningkatkan keterlibatan pemain dan memfasilitasi pembelajaran melalui pengalaman interaktif. Dengan pendekatan ini, game *Moving Score* tidak hanya menghadirkan tantangan yang menghibur tetapi juga melatih keterampilan pemain dalam mengoordinasikan gerakan dan strategi permainan.

Perancangan game berbasis *Virtual Reality* memerlukan pemilihan perangkat lunak dan teknologi yang sesuai untuk memastikan pengalaman yang optimal bagi pemain. *Unity* dipilih sebagai *game engine* utama karena fitur-fitur canggihnya, seperti *XR Interaction Toolkit*, yang memungkinkan pengembangan interaksi berbasis gerakan dengan lebih mudah dan efisien. Zhang dalam *Ergonomics in Virtual Reality Games for Children* (2021) mencatat bahwa desain perangkat keras dan lunak dalam *Virtual Reality* harus mempertimbangkan aspek ergonomi dan kenyamanan pengguna agar pengalaman bermain tetap optimal dan tidak menyebabkan kelelahan berlebih.

Peran *Game Programmer* menjadi salah satu aspek kunci yang menentukan keberhasilan permainan. *Game Programmer* bertanggung jawab untuk mengembangkan mekanisme permainan, sistem interaksi, dan optimasi performa agar game dapat berjalan dengan lancar. Menurut Rabin dalam *Introduction to Game Development* (2010), *Game Programmer* memiliki tugas utama dalam merancang sistem logika permainan, mengimplementasikan kontrol berbasis gerakan, serta memastikan bahwa setiap elemen dalam game dapat berfungsi secara optimal sesuai dengan desain yang telah dirancang. Oleh karena itu, pemrograman yang baik akan meningkatkan pengalaman pengguna dengan memastikan interaksi yang responsif dan realistis.

Perancangan game *Moving Score* berbasis *Virtual Reality* ini bertujuan untuk mengeksplorasi potensi teknologi *Virtual Reality* dalam menciptakan pengalaman game yang menarik dan bermanfaat. Dengan memanfaatkan tools dan platform pengembangan game *Virtual Reality* yang tersedia, diharapkan game ini dapat memberikan kontribusi positif bagi industri game dan masyarakat secara umum.

METODE PENELITIAN

Pengembangan produk game *Moving Score* berbasis *Virtual Reality* ini menggunakan metode prototyping. Metode ini sebagai alat utama untuk memahami kebutuhan pengguna, menguji fitur, dan mengevaluasi kinerja game secara bertahap. Tahapan utama dalam metode prototyping meliputi identifikasi kebutuhan awal, pembuatan prototipe awal, pengujian, iterasi dan perbaikan, serta validasi akhir. Proses pengumpulan data dalam pengembangan game *Moving Score* berbasis *Virtual Reality* ini dilakukan melalui satu teknik utama, yaitu observasi langsung. Sebagai *Game Programmer*, saya melakukan observasi langsung selama proses pengembangan untuk mengumpulkan data mengenai implementasi kode, mekanisme interaksi, sistem skor, dan optimasi performa dalam game. Analisis produk dilakukan setelah tahap pengembangan dan pengujian game *Moving Score* berbasis *Virtual Reality*. Evaluasi ini bertujuan untuk menganalisis performa sistem, keefektifan mekanik permainan, serta stabilitas teknis game. Metode yang digunakan dalam analisis ini adalah Analisis SWOT (Strengths, Weaknesses, Opportunities, Threats), yang mengidentifikasi keunggulan, tantangan, peluang, dan risiko dalam pengembangan game ini.

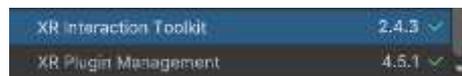
HASIL DAN PEMBAHASAN

A. Penyajian Data Produk

Pada tahap ini penulis akan menyajikan hasil dari tahap produksi yaitu *programming*, dan *integrasi asset* beserta kode program dalam perancangan *game*. *Game* ini akan dibuat dengan *game engine Unity* dengan *XR Interaction Toolkit* beserta *Visual Studio Code*. Pada tahap ini ada beberapa yang perlu disiapkan untuk kebutuhan awal.

1. Persiapan Awal pada Game

Persiapan awal dalam pengembangan *game* dimulai dengan menambahkan beberapa *package* penting yang mendukung pengembangan berbasis *Virtual Reality*. *Package* pertama yang ditambahkan adalah *XR Interaction Toolkit*, yang berfungsi untuk menyediakan komponen dan sistem interaksi dalam lingkungan, seperti *grab*, *teleport*, interaksi UI, dan lain-lain. Selanjutnya, ditambahkan *XR Plugin Management* yang berguna untuk mengelola *plugin platform* seperti *Oculus*. Dengan *package* ini, pengembang dapat dengan mudah mengatur dan mengaktifkan *plugin* yang sesuai dengan perangkat yang digunakan.



Gambar 1. Package XR

Sumber : Dokumentasi Pribadi

2. Pembuatan Scene Main Menu

Pembuatan dari *Scene Main Menu* diawali dengan membuat *scene* baru pada *folder asset Unity*. Pada *Main Menu* terdapat *canvas* yang berisikan *User Interface* (UI) dari tampilan *Main Menu*, *Select Game* dan *Credits* yang dapat ditampilkan dengan menekan tombol yang disesuaikan untuk menampilkan halaman-halaman tersebut seperti tombol *PLAY* untuk menampilkan *panel Select Button*, tombol *CREDITS* untuk menampilkan *panel Credits*, tombol *QUIT* untuk menampilkan *panel Yes or No*, dan tombol *YES* untuk keluar dari *game* yang dapat ditekan pemain diatur menggunakan *script MainMenu* seperti pada gambar 34.

```
public void Quit()
{
    Debug.Log("Keluar dari Game");
    Application.Quit();
}
```

Gambar 2. Script Main Menu Untuk Quit Game

Sumber : Dokumentasi Pribadi

3. Pembuatan Panel Select Game

Jika pemain menekan tombol *PLAY* maka akan ditampilkan panel dari *Select Game*, disini pemain bisa memilih beberapa *mode* atau tempat yaitu *Moon*, *City*, dan *Beach*. Jika pemain menekan tombol *BACK* pada *panel Select Game* maka pemain akan kembali ke *panel Main Menu*.



Gambar 3. Atribut Tombol PLAY Untuk Menampilkan Panel Select Game
 Sumber : Dokumentasi Pribadi



Gambar 4. Atribut Tombol BACK Untuk Menampilkan Panel Main Menu
 Sumber : Dokumentasi Pribadi

Pemain dapat menekan salah satu tombol *Moon*, *City*, dan *Beach* kemudian pemain akan berpindah *scene* sesuai yang dipilih pemain.

```
using UnityEngine;
using UnityEngine.SceneManagement;

[RequireComponent(typeof(Button))]
public class MainMenu : MonoBehaviour
{
    [SerializeField]
    public void Moon()
    {
        SceneManager.LoadScene("Moon");
    }

    [SerializeField]
    public void City()
    {
        SceneManager.LoadScene("City");
    }

    [SerializeField]
    public void Beach()
    {
        SceneManager.LoadScene("Beach");
    }
}
```

Gambar 5. Script Main Menu Untuk Perpindahan Scene Select Game
 Sumber : Dokumentasi Pribadi



Gambar 6. Atribut Tombol Moon Untuk Perpindahan Scene
 Sumber : Dokumentasi Pribadi



Gambar 7. Atribut Tombol City Untuk Perpindahan Scene
 Sumber : Dokumentasi Pribadi



Gambar 8. Atribut Tombol Beach Untuk Perpindahan Scene
Sumber : Dokumentasi Pribadi



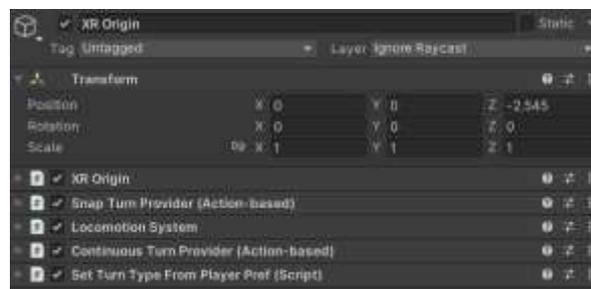
Gambar 9. Implementasi UI Main Menu
Sumber : Dokumentasi Pribadi

4. Pembuatan Player pada scene Main Menu

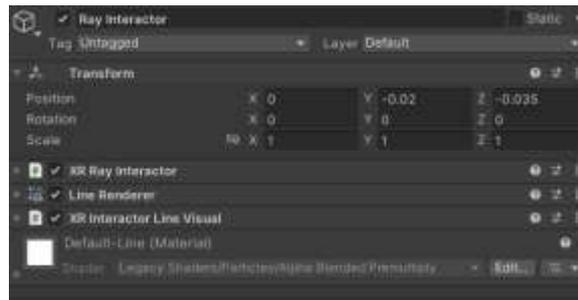
Pada *scene Main Menu* *player* tidak bisa bergerak tetapi bisa menggerakkan tangan dan *camera* atau kepala *player* untuk melihat *environment* sekitarnya. Pemain menggunakan *raycast* untuk mengklik tombol pada *panel Main Menu*.



Gambar 10. Game Object Yang Ada Di Player
Sumber : Dokumentasi Pribadi



Gambar 11. Atribut Player pada XR Origin
Sumber : Dokumentasi Pribadi



Gambar 12. Atribut Player Pada Ray Ineractor

Sumber : Dokumentasi Pribadi

5. Pembuatan FadeScreen pada scene Main Menu

Pembuatan efek *FadeScreen* diawali dengan menambahkan *gameobject* baru yang berisi komponen *Renderer* dan *script FadeScreen* ke dalam *gameobject camera* pada *player*. Efek *FadeScreen* digunakan untuk memberikan transisi visual berupa efek gelap perlahan (*fade out*) atau terang perlahan (*fade in*) saat berpindah antar halaman atau *scene*. Efek transisi ini dibuat agar perpindahan antar tampilan tidak terasa mendadak dan lebih nyaman dilihat oleh pemain.



Gambar 13. Script Fade Screen Bagian 1

Sumber : Dokumentasi Pribadi

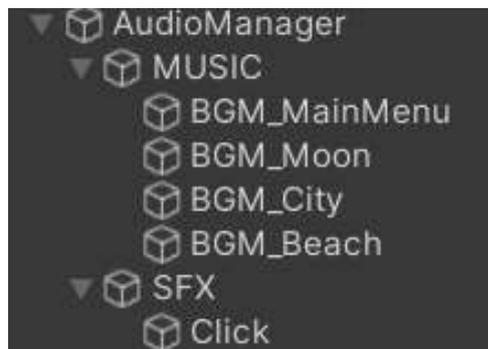


Gambar 14. Script Fade Screen Bagian 2

Sumber : Dokumentasi Pribadi

6. Pembuatan AudioManager pada scene Main Menu

Pembuatan *AudioManager* pada *scene Main Menu* diawali dengan menambahkan *gameobject* baru yang diberi nama "*AudioManager*" dan dilengkapi *script AudioManager*. *Script* ini bertanggung jawab untuk mengatur dan memutar efek suara (SFX) serta musik latar belakang (BGM) pada *game*. *AudioManager* bersifat *global* dan tidak akan dihancurkan ketika berpindah *scene* menggunakan *DontDestroyOnLoad*, sehingga suara dapat terus diputar secara konsisten antar *scene*. Secara otomatis memainkan musik latar sesuai *indeks scene* menggunakan fungsi *PlayBGMByScene()* yang dipanggil saat *scene* dimuat. Selain itu, *AudioManager* juga dapat digunakan untuk memutar dan menghentikan efek suara secara dinamis selama interaksi pemain dengan *elemen UI* maupun objek *game*.



Gambar 15. Gameobject Audio Manager Pada Hierarchy

Sumber : Dokumentasi Pribadi



Gambar 16. Script Audio Manager Bagian 1

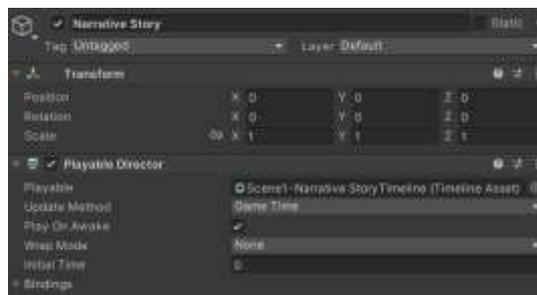
Sumber : Dokumentasi Pribadi



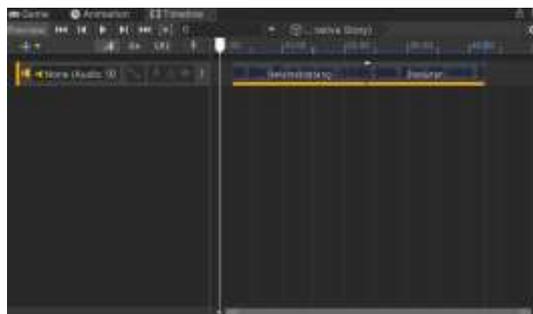
Gambar 17. Script Audio Manager Bagian 2
Sumber : Dokumentasi Pribadi

7. Pembuatan NarrativeStory pada scene Main Menu

Menambahkan *gameobject* baru yang diberi nama “*Narrative Story*” kemudian menambahkan *plugin Playable Director* dan membuat *Timeline* dan masukkan *audio*.



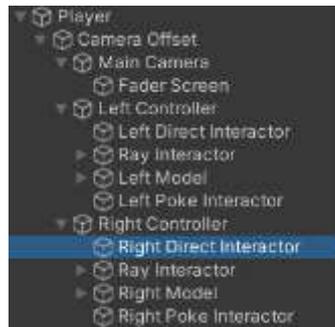
Gambar 18. Atribut Pada Game Object Narrative Story
Sumber : Dokumentasi Pribadi



Gambar 19. Timeline Narrative Story Di Scene Main Menu
Sumber : Dokumentasi Pribadi

8. Pembuatan Player pada scene Moon

Pada *scene Moon* player bisa bergerak, melompat dan mengambil bintang dengan menggunakan tangan kanan untuk mendapatkan *score*. Menambahkan *gameobject* baru ke dalam *player* seperti *direct interactor*, *ray interactor* agar *player* bisa mengambil bintang dan untuk interaksi UI menggunakan *raycast* pada *scene Moon*.



Gambar 20. Game Object Player Pada Hierarchy

Sumber : Dokumentasi Pribadi

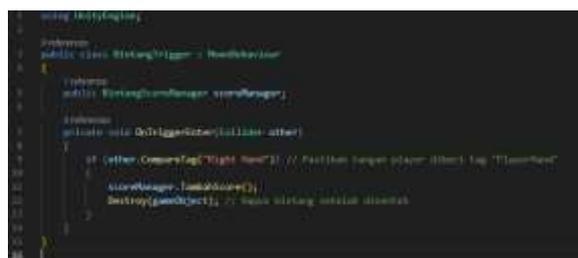


Gambar 21. Atribut Player Pada Scene Moon

Sumber : Dokumentasi Pribadi

9. Pembuatan Mekanik Bintang pada scene Moon

Diawali dengan memasukkan *gameobject* Bintang ke *scene Moon*, kemudian menambahkan *script BintangTrigger*, *BintangMoving*, dan *BintangScoreManager*. *Script BintangTrigger* untuk mendeteksi jika *player* sudah mengambil Bintang atau menyentuh Bintang yang sedang bergerak. *Script BintangMoving* untuk pergerakan *gameobject* Bintang dari posisi A ke posisi B, dan *BintangScoreManager* berguna untuk menyimpan *score* yang sudah didapat dan sebagai *update* dari UI *score* juga.



Gambar 22. Script Bintang Trigger

Sumber : Dokumentasi Pribadi



Gambar 23. Script Bintang Moving
Sumber : Dokumentasi Pribadi



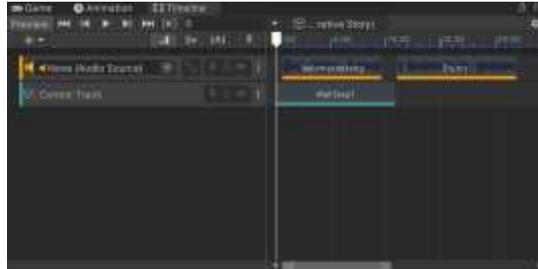
Gambar 24. Script Bintang Score Manager Bagian 1
Sumber : Dokumentasi Pribadi



Gambar 25. Script Bintang Score Manager Bagian 2
Sumber : Dokumentasi Pribadi

10. Pembuatan NarrativeStory pada scene Moon

Menambahkan *gameobject* baru yang diberi nama “*Narrative Story*” ke dalam *scene Moon*, kemudian menambahkan *plugin Playable Director* dan membuat *Timeline* dan masukkan *audio*.



Gambar 26. Timeline Narrative Story Di Scene Moon

Sumber : Dokumentasi Pribadi

11. Pembuatan Rintangannya pada scene City

Pada *scene City* terdapat rintangan yang harus dihindari *player* untuk mendapatkan *score* dengan cara bergerak kiri atau kanan. Diawali dengan membuat *gameobject* baru kemudian tambahkan *script RintangannyaController* dan *script Spawner* untuk mengatur *spawn* rintangannya seberapa cepat.



Gambar 27. Script Rintangannya Controller

Sumber: Dokumentasi Pribadi

Bintang Laut pada *scene Beach* bisa di *grab* dengan tangan dan memasukkannya ke dalam keranjang agar mendapatkan *score*, Bintang Laut tersebar di tepi Pantai dan *player* harus mencarinya dan mengambil untuk dimasukkan ke keranjang. Terdapat *plugin XR Grab Interactable* di setiap gameobject Bintang Laut.



Gambar 34. Atribut Gameobject Pada Bintang Laut

Sumber: Dokumentasi Pribadi

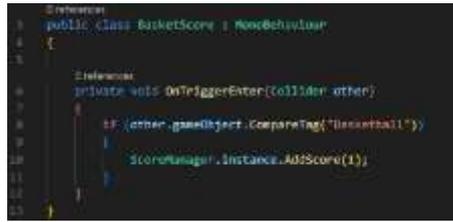
15. Pembuatan Keranjang pada scene Beach

Pada awalnya menambahkan *gameobject* keranjang, kemudian menambahkan *script BasketScore* ke dalam *gameobject* keranjang. Pada saat Bintang Laut masuk ke keranjang maka *score* akan bertambah. Tambahkan juga *script ScoreManager* untuk mengatur *text score* UI, *audio* jika Bintang laut masuk ke keranjang, dan menampilkan *finish* UI jika *score* sudah tercapai.



Gambar 35. Script Score Manager

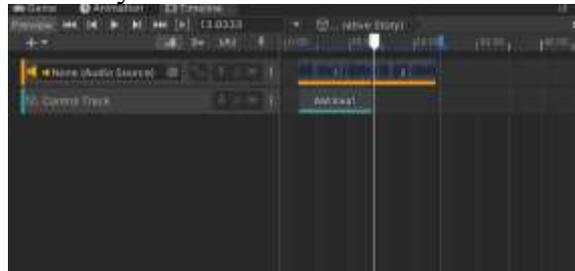
Sumber: Dokumentasi Pribadi



Gambar 36. Script Basket Score
Sumber: Dokumentasi Pribadi

16. Pembuatan NarrativeStory pada scene Beach

Menambahkan *gameobject* baru yang diberi nama “*Narrative Story*” kemudian menambahkan *plugin Playable Director* dan membuat *Timeline* dan masukkan *audio* sesuai dengan suaranya.



Gambar 37. Timeline Narrative Story Di Scene Beach
Sumber: Dokumentasi Pribadi

Analisis dan Interpretasi

Tabel 1. Tabel Analisis dan Interpretasi Fitur Game

No.	Fitur Game	Teknik/Aspek yang Dianalisis	Metode/Teknik Analisis	Hasil	Interpretasi
1.	Sistem Skor	Respons pemain terhadap sistem skor	Observasi langsung saat bermain	Pemain memahami bahwa skor bertambah saat aksi berhasil dilakukan	Skor berperan sebagai motivasi dan bentuk feedback positif terhadap keberhasilan tugas
2.	Deteksi Gerakan Melompat	Akurasi deteksi gerakan	Pengujian fungsional dan observasi	Gerakan melompat terdeteksi dengan baik menggunakan XR Toolkit	Fitur ini mendukung pengembangan motorik kasar dengan memberikan ruang untuk aktivitas fisik
3.	Efek Suara Interaktif	Respon audio saat interaksi	Observasi dan feedback subyektif	Efek suara muncul setiap interaksi dengan objek	Efek audio memperkuat keterlibatan dan memberikan petunjuk berhasil tidaknya suatu aksi
4.	Pemilihan Level Game	Navigasi level dan pemahaman	Pengamatan UI/UX dan	Pemain mampu memilih level	Memberikan fleksibilitas dan

		konteks	usability testing	sesuai kemampuan tanpa kebingungan	tantangan bertahap bagi pemain dalam menyelesaikan game
5.	Interaksi Objek (Grab/Drop)	Kemampuan grab/drop objek virtual	Observasi langsung pemain	Fitur berjalan lancar, anak-anak bisa mengambil dan meletakkan objek dengan benar	Mendukung koordinasi mata-tangan dan memperkuat pembelajaran berbasis aksi nyata
6.	Sistem Navigasi Menu	Kemudahan akses menu VR	Pengujian langsung	Menu dapat digunakan dengan kontrol VR tanpa kendala teknis	UX VR yang baik, cocok untuk anak-anak yang baru mengenal lingkungan virtual
7.	Lingkungan Virtual Sederhana	Kompleksitas dunia virtual	Observasi reaksi pemain	Lingkungan tidak membingungkan dan nyaman dipandang	Memudahkan adaptasi pengguna baru terhadap VR tanpa overstimulasi visual
8.	Sistem Feedback Visual	Efektivitas visual saat aksi berhasil	Observasi dan tanggapan pengguna	Highlight dan animasi membantu pemain tahu bahwa aksi berhasil dilakukan	Memberi kejelasan visual terhadap keberhasilan aksi, memperkuat konsep reward secara eksplisit

KESIMPULAN

Perancangan *game Moving Score* berbasis *Virtual Reality* berhasil dibuat dengan tujuan memberikan pengalaman bermain yang seru sekaligus membuat pemain lebih aktif bergerak. Dalam *game* ini, pemain diminta untuk melakukan gerakan seperti melompat, menghindari, dan menangkap benda, lalu sistem akan memberi poin setiap kali gerakan dilakukan dengan benar. *Game* ini dibangun menggunakan *Unity* dan *XR Interaction Toolkit* yang memungkinkan gerakan pemain terbaca dengan baik dan terasa responsif. Adanya tampilan skor serta suara dan efek visual yang menarik membuat pemain semakin semangat bermain. Meskipun sudah cukup lancar dan nyaman digunakan, *game* ini masih memiliki kekurangan, seperti hanya bisa dimainkan dengan perangkat *Virtual Reality* tertentu dan belum banyak variasi tempat dalam *game*. Namun secara keseluruhan, *Moving Score* tidak hanya menghibur, tapi juga bisa menjadi cara baru yang menyenangkan untuk mengajak orang khususnya remaja lebih aktif bergerak.

DAFTAR PUSTAKA

- Bailenson, J. N. (2018). *Experience on Demand: What Virtual Reality Is, How It Works, and What It Can Do*. W. W. Norton & Company.
- Burdea, G. C., & Coiffet, P. (2003). *Virtual Reality Technology*. John Wiley & Sons.



- Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). *From game design elements to gamefulness: Defining gamification*. Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, 9-15.
- Gallahue, D. L., & Ozmun, J. C. (2013). *Understanding Motor Development: Infants, Children, Adolescents, Adults*. McGraw-Hill Education.
- Gee, J. P. (2003). *What Video Games Have to Teach Us About Learning and Literacy*. Palgrave Macmillan.
- Juul, J. (2005). *Half-Real: Video Games between Real Rules and Fictional Worlds*. MIT Press.
- Milgram, P., & Kishino, F. (1994). *A taxonomy of mixed reality visual displays*. IEICE Transactions on Information and Systems, E77-D(12), 1321-1329.
- Prensky, M. (2001). *Digital Game-Based Learning*. McGraw-Hill.
- Schell, J. (2019). *The Art of Game Design: A Book of Lenses*. CRC Press.
- Sherman, W. R., & Craig, A. B. (2003). *Understanding Virtual Reality: Interface, Application, and Design*. Morgan Kaufmann.
- Trawick-Smith, J. (2014). *Early Childhood Development: A Multicultural Perspective*. Pearson.
- Williams, R. (2001). *The Animator's Survival Kit*. Faber & Faber.
- Zhang, Y. (2021). *Ergonomics in Virtual Reality Games for Children*. Springer.
- Zyda, M. (2005). *From visual simulation to virtual reality to games*. Computer, 38(9), 25-32.