

Unity Game Engine Sebagai Media Pengembangan Game Musik Cross Platform

Unity Game Engine as a Media for Cross Platform Music Game Development

Muhammad Yakub Ramadhan^a, Rika Rosnelly^b

^aJurusan Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Potensi Utama

^bDosen Jurusan Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Potensi Utama

Universitas Potensi Utama, Jl.K.L. Yos Sudarso Km. 6,5 No 3A Tanjung Mulia Medan

myr1299.vfx@gmail.com

ABSTRAK

Sebagian besar pengembang game hanya merilis game untuk platform yang terkenal seperti Windows karena basis pengguna yang besar, yang membuat platform lain seperti Linux tidak bisa menikmati game-game terbaru yang hanya bisa di mainkan di Windows. Namun dengan perkembangan dalam pengembangan game yang sangat pesat sekarang para pengembang game bisa merilis game di lebih dari satu platform menggunakan Unity Game Engine yang mendukung banyak platform termasuk Linux. Penelitian ini dilakukan untuk mengembangkan game cross platform menggunakan Unity dengan game musik atau rhythm game sebagai jenis game yang akan dikembangkan karena kebutuhan perangkat keras yang tidak terlalu tinggi dan ikut berperan dalam mengembangkan fungsi kognitif pemain.

Kata Kunci : Game, Pengembangan Game, Cross Platform, Rhythm Game, Unity.

ABSTRACT

Majority of game developers only release their games for mainstream and well known platforms like Windows because of their huge user base which makes any other platforms like Linux can't enjoy some new games only Windows users have. But with vast improvements in game development scene now a game developer can release a cross platform game with Unity Game Engine that supports many platforms including Linux. This research is conducted to develop an Unity cross platform game with rhythm game as the game genre which can be played with less demanding hardware and also taking part in improving players cognitive function.

Keywords : Game, Game Development, Cross Platform, Rhythm Game, Unity.

Info Artikel :

Disubmit: 10 Februari 2023

Direview: 23 Maret 2023

Diterima : 15 April 2023

Copyright © 2023 - Journal UPU. All rights reserved.

1. PENDAHULUAN

Perkembangan game yang sangat pesat seiring dengan perkembangan teknologi yang juga tak kalah pesat membuat banyak orang antusias dengan inovasi yang dihadirkan oleh para pelaku industri *game* [1], namun bukan berarti semua jajaran platform untuk bermain *game* diperlakukan sama rata, dengan platform seperti Windows yang menguasai pasar *game PC* [2] membuat para pengembang *game* besar kurang tertarik mengembangkan *game* untuk platform lainnya sekalipun *game* itu di rilis secara *cross platform*, salah satu contohnya adalah Linux yang dikarenakan kurangnya basis pengguna jika dibandingkan dengan Windows [3]. Walaupun dengan berbagai kekurangannya bukan berarti tidak ada upaya yang dilakukan beberapa pengembang *game* seperti Valve untuk membuat sistem operasi berbasis Linux bisa dan layak dijadikan platform penting untuk pengembangan *game* [4], salah satunya terdapat pada *Game Engine* yang terkenal saat ini yaitu Unity yang menjadi salah satu *Game Engine* yang mendukung banyak platform terkenal dan termasuk Linux, memungkinkan bagi para pengembang untuk membuat *game* dan merilisnya ke lebih dari satu platform [5].

Penelitian yang ditulis oleh Ruben Rodrigues Rebelo (2016) dengan judul “Building a Music Rhythm Video Game” mengembangkan *prototype game* musik yang terinspirasi dari karakteristik *Rhythm Game* lain [6] dan dikembangkan menggunakan *Android SDK* sebagai *Game Engine* untuk mendukung berbagai fungsi dan mekanik yang terdapat dalam game tersebut [7]. Penelitian yang Ditulis oleh Dzulfickar Rhezaa Algadhari (2020) dengan judul “Pengembangan Rhythm Game Berbasis Android dengan Unity 2D Sebagai Media Pembelajaran Seni Musik Gamelan” memanfaatkan *Unity* sebagai *Game Engine* untuk membuat *rhythmgame* yang ditujukan sebagai media audiovisual pembelajaran seni musik Gamelan [8]. Penelitian yang Ditulis oleh Alphianno Steady Kambodji (2016) dengan judul “Rancang Bangun Rhythm Game Sebagai Sarana Belajar Bermain Gitar” mengimplementasikan sebuah *rhythm game* yang menggunakan deteksi frekuensi nada gitar sebagai input dengan *DPE Framework* sebagai media perancangan *game* [9] dan *Unity* sebagai media pengimplementasian fungsi *game* [10].

Tujuan dari penelitian ini adalah untuk mengembangkan suatu *game* yang bisa dimainkan di lebih dari satu *platform* dengan *Windows* dan *Linux* sebagai *platform* yang di tuju menggunakan *Unity game engine* sebagai media pengembangan *game*, dengan *rhythm game* yang dijadikan jenis *game* yang akan dibuat karena mudah untuk dimainkan tanpa memerlukan perangkat dengan performa tinggi untuk dimainkan [11], dan bantuan plugin *DryWetMIDI* dari *Melanchall* untuk membaca file *chart MIDI* suatu lagu yang di buat [12] dengan menggunakan *LMMS* yang juga digunakan untuk mensinkronisasi antara lagu dengan *chart*.

2. METODE

A. Metode Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan dengan menggunakan metode sebagai berikut:

a) Metode Studi Pustaka

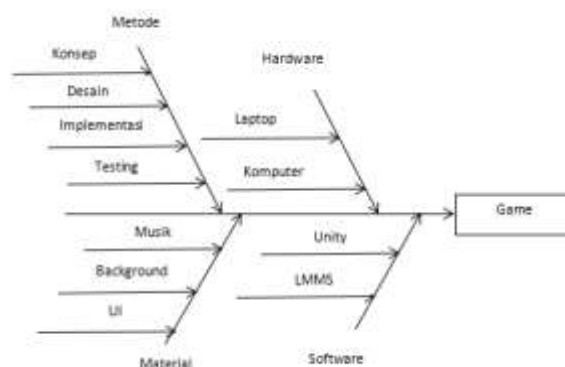
Mencari dan mengumpulkan data dari berbagai sumber, data yang diambil berupa konsep dasar yang terdapat informasi mengenai fungsi-fungsi dari komponen yang digunakan selama penelitian.

b) Metode Observasi

Mengamati objek yang akan dibuat dengan melakukan berbagai percobaan baik secara langsung maupun tidak langsung.

B. Diagram *Fishbone* Metodologi Penelitian

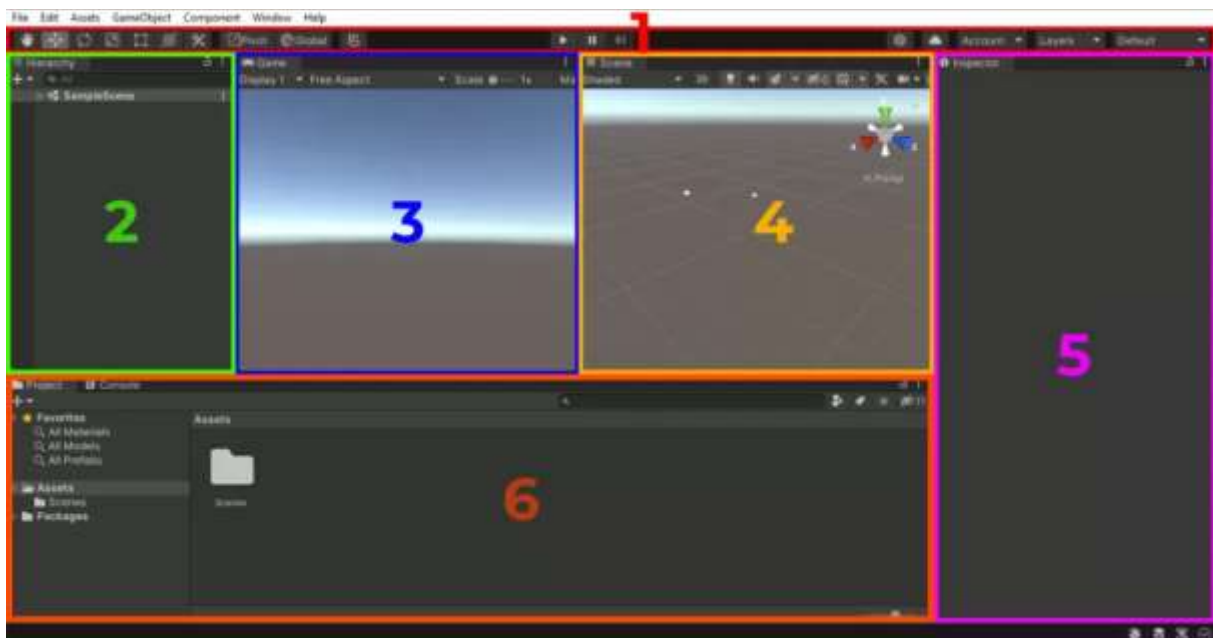
Pengumpulan data dalam penelitian ini dapat digambarkan pada diagram *Fishbone* berikut ini:



Gambar 1. Diagram *Fishbone*

Keterangan:

1. **Metode**
Proses yang diterapkan dalam penelitian ini di mulai dari perancangan konsep *game*, desain UI dan *gameplay*, implementasi, dan diakhiri dengan proses uji coba atau *testing*.
2. **Material**
Material yang diperlukan dalam pembuatan *game* ini adalah desain UI, gambar latar belakang atau *background*, dan musik.
3. **Hardware**
Hardware yang dibutuhkan dalam pengembangan *game* ini adalah PC/Laptop sebagai media pengembangan dan testing game dengan spesifikasi sebagai berikut:
 - a) Laptop Dell Latitude E6440
 - b) CPU Intel Core i5-4310M
 - c) GPU Intel HD Graphics 4600
 - d) RAM 2x2 GB DDR3L-1600
 - e) mSATA SSD 256 GB
 - f) SATA HDD 500 GB 7200rpm
4. **Software**
Software yang digunakan dalam pengembangan *game* ini terdiri dari *Unity* sebagai *game engine* untuk mengembangkan *game* yang dibantu dengan plugin *DryWetMIDI* dari *Melanchall* untuk membaca file *chart* berjenis MIDI dan *LMMS* sebagai program *Digital Audio Workstation (DAW)* untuk media pembuatan musik [13] dan *chart* lagu.

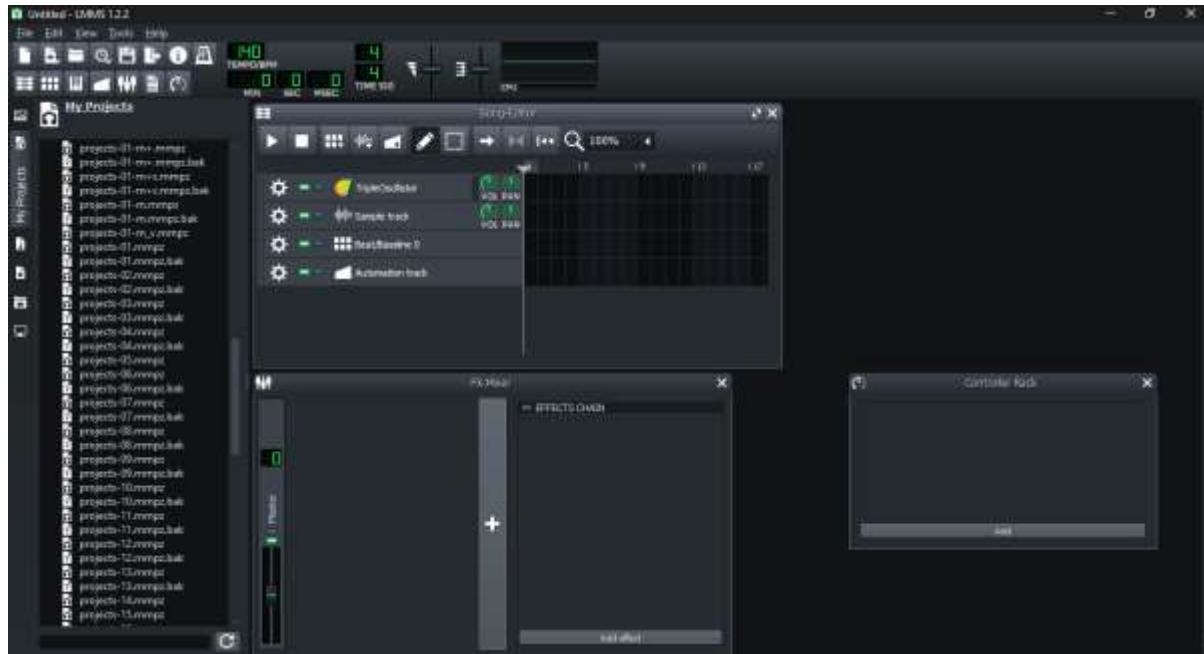


Gambar 2. Tampilan dasar dari *Unity*

Pada saat pertama kali membuka *project Unity*, terdapat enam *window* utama yang muncul seperti yang diperlihatkan di gambar 2 [14] yang terdiri dari:

- a. *Toolbar* yang menyediakan tombol tool untuk mengatur berbagai *GameObject*/komponen dalam *game scene*.

- b. *Hierarchy* yang berisi kumpulan semua *GameObject*, seperti *3D object*, *light*, *UI*, *sound*, *effect*, *video* dan *camera*.
- c. *Game View* yang menampilkan permainan pada saat dijalankan setelah menekan tombol *Play*.
- d. *Scene View* untuk membuat dan mengatur tampilan dunia *game* yang sedang di buat.
- e. *Inspector* untuk melihat dan mengedit nilai dari properti *GameObject* yang dipilih saat ini.
- f. *Project Window* yang menyimpan semua asset (*script*, *texture*, *3D model*, *audio clip*, dll.) yang digunakan dalam proses pengembangan *game*.



Gambar 3. Tampilan dasar dari LMMS

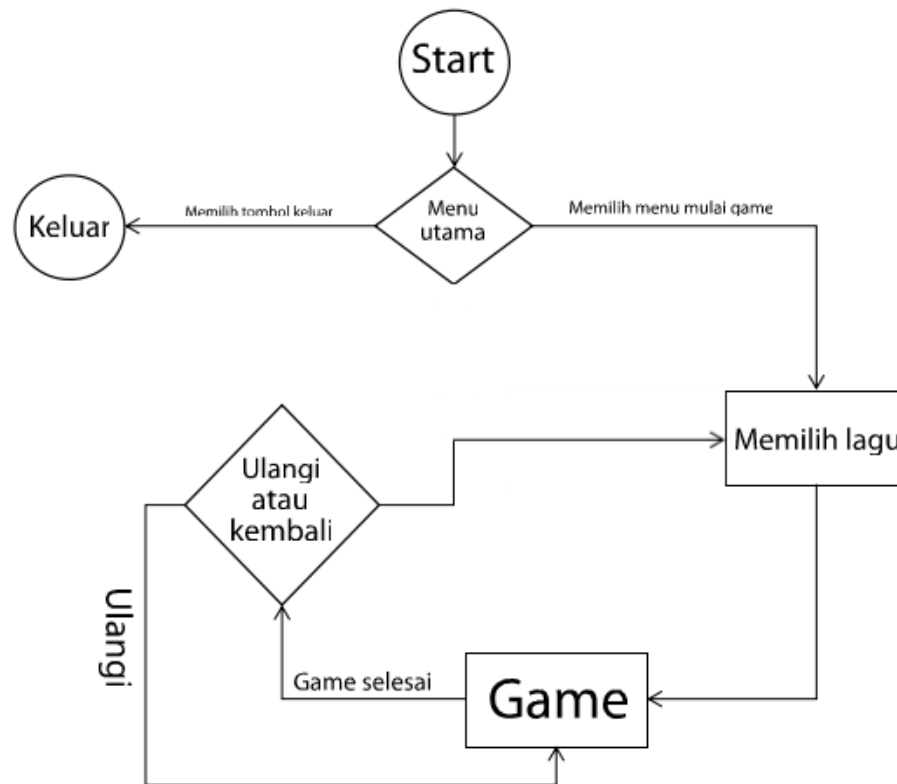
Pada saat membuka *project* baru dari LMMS, terdapat tampilan yang muncul seperti pada gambar 3 [15] yang terdiri dari:

1. *Sidebar* yang menampilkan akses ke beberapa elemen yang digunakan seperti “*My Projects*”, “*Instrument Plugins*”, dll.
2. *Song Editor* yang menampilkan struktur susunan lagu yang sedang di buat.
3. *FX Mixer* yang mengatur volume dari suatu instrumen tersendiri atau lagu secara keseluruhan.
4. *Controller Rack* yang mengatur posisi kontroler instrumen yang digunakan.

3. HASIL DAN PEMBAHASAN

A. Diagram Flowchart

Flowchart diagram pada *game* yang akan dibuat untuk memudahkan pembacaan cara kerja *game* secara keseluruhan yang dapat di lihat di gambar 4 di bawah ini:



Gambar 4. Diagram *Flowchart*

Ketika pertama kali membuka *game*, *User* akan ditampilkan layar menu utama dimana terdapat 5 buah tombol yang terdiri dari mulai *game* untuk memulai permainan, petunjuk untuk mengetahui cara bermain, pengaturan untuk mengatur beberapa komponen di dalam *game*, tentang untuk melihat pembuat *game*, dan keluar untuk keluar dari permainan.

Setelah *User* menekan tombol mulai *game* akan ditampilkan menu pilihan lagu berdasarkan daerah yang di pilih. Jika sudah di pilih lagu yang akan dimainkan maka permainan pun di mulai.

Kemudian jika *User* telah menyelesaikan suatu lagu akan ditampilkan hasil skor, nada beruntun, dan kelipatan skor yang *User* dapatkan beserta benda yang *User* berhasil didapatkan dan pilihan untuk ulangi atau kembali. Jika *User* memilih ulangi maka *game* pun akan di ulangi kembali, sedangkan jika *User* memilih kembali maka Pemain akan diarahkan kembali ke menu pilihan lagu.

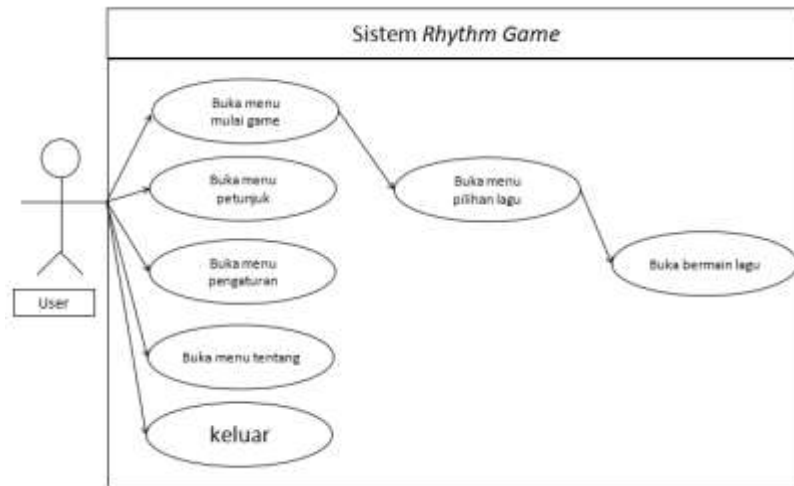
B. Desain Sistem

Desain diagram UML yang digunakan antara lain *use case diagram*, *sequence diagram*, dan *activity diagram* [16].

a. Use Case Diagram

Use case diagram pada *game* yang akan dibuat menggambarkan interaksi antara pemain dengan *game* yang digambarkan pada gambar 3 Diagram tersebut memiliki 1 aktor yaitu *User* dan 7 *use case*. *Game* ini memiliki 5 menu utama yaitu menu mulai *game*, petunjuk, pengaturan, tentang, dan keluar.

Jika *User* membuka menu mulai *game*, terdapat pilihan lagu yang tersedia. Kemudian setelah *User* memilih lagu yang ingin dimainkan, maka *User* akan membuka menu bermain.

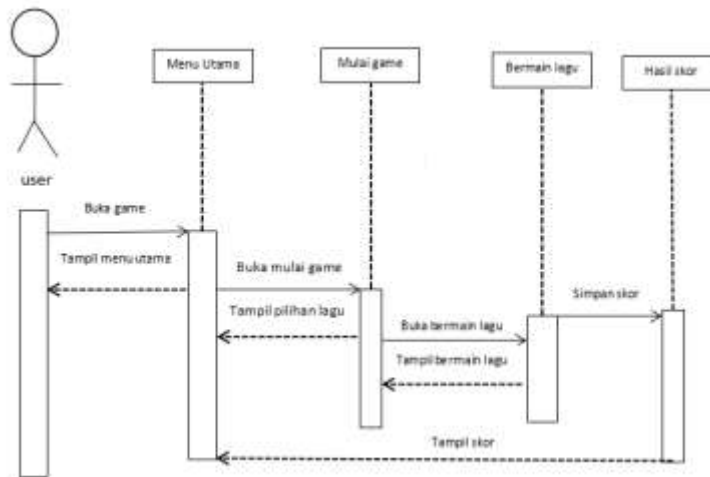


Gambar 5. Use Case Diagram

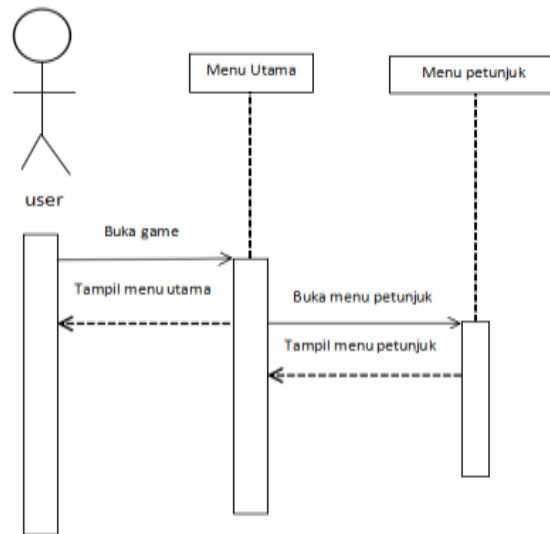
b. Sequence Diagram

Sequence diagram pada *game* yang akan dibuat menggambarkan urutan *User* mengakses menu-menu yang terdapat di dalam *game*. *Sequence diagram* dari *rhythm game* ini terbagi kedalam 4 diagram: *Sequence diagram* menu mulai *game*, *Sequence diagram* menu petunjuk, *Sequence diagram* menu pengaturan, dan *Sequence diagram* menu tentang.

Gambar 6 menggambarkan urutan *User* mengakses menu mulai *game*. Pertama user membuka *game*, lalu *game* tersebut menampilkan halaman menu utama. Pada halaman menu utama, *User* memilih menu mulai *game*. Pada halaman mulai *game*, *User* memilih lagu dan tingkat kesulitan yang akan dimainkan, kemudian akan tampil halaman bermain lagu. Pada halaman ini *User* harus menekan blok yang muncul dari atas layar sesuai dari ketukan lagu untuk mendapatkan skor. Setelah lagu selesai maka akan ditampilkan hasil skor yang diperoleh *User*.

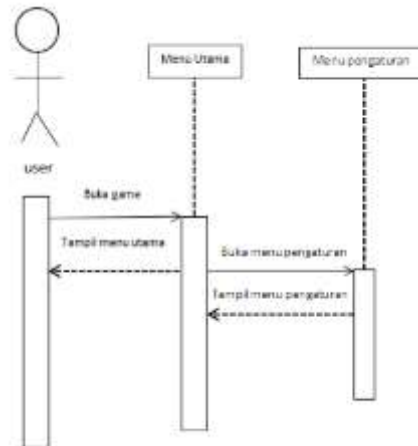
Gambar 6. Sequence Diagram menu mulai *game*

Gambar 7 menggambarkan urutan *User* mengakses menu petunjuk. Pertama *User* membuka *game*, lalu *game* tersebut menampilkan halaman menu utama. Pada halaman menu utama, *User* memilih menu petunjuk kemudian akan tampil halaman petunjuk.



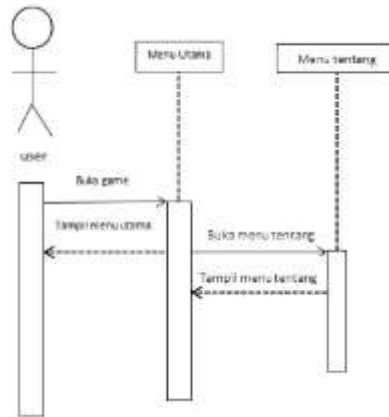
Gambar 7. *Sequence Diagram* menu petunjuk

Gambar 8 menggambarkan urutan *User* mengakses menu pengaturan. Pertama *User* membuka *game*, lalu *game* tersebut menampilkan halaman menu utama. Pada halaman menu utama, *User* memilih menu pengaturan kemudian akan tampil halaman pengaturan.



Gambar 8. *Sequence Diagram* menu pengaturan

Gambar 9 menggambarkan urutan *User* mengakses menu tentang. Pertama *User* membuka *game*, lalu *game* tersebut menampilkan halaman menu utama. Pada halaman menu utama, *User* memilih menu tentang kemudian akan tampil halaman tentang.



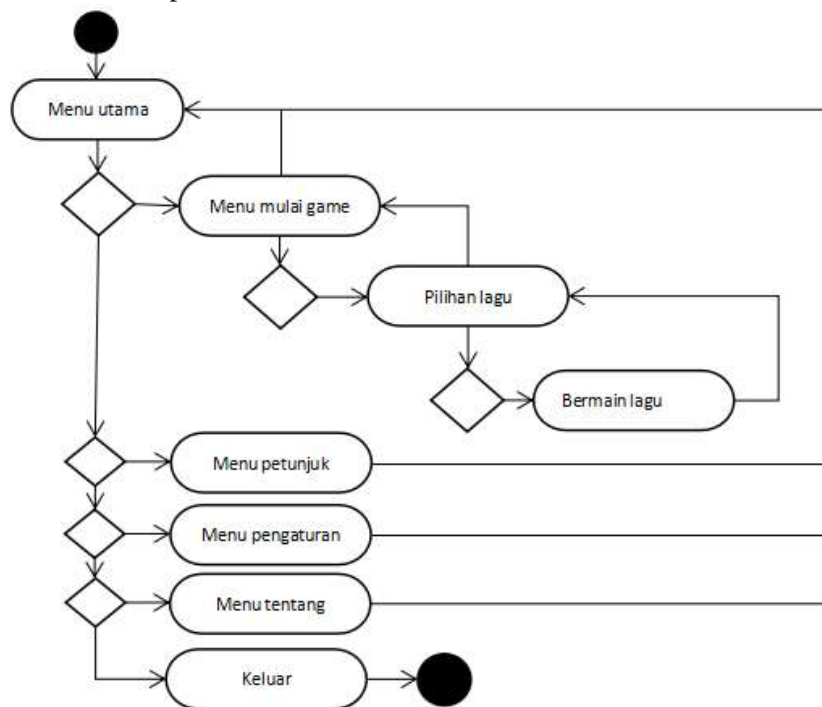
Gambar 9. Sequence Diagram menu tentang

c. Activity Diagram

Activity diagram pada *game* yang akan dibuat menggambarkan aliran kerja dari *rhythm game* yang akan dibuat. Gambar 10 menggambarkan menu-menu yang dapat dipilih *User* pada *game* tersebut, terdiri dari menu mulai *game*, petunjuk, pengaturan, tentang, dan keluar.

Pada menu mulai *game*, *User* dapat memilih lagu yang ada beserta tingkat kesulitannya. Setelah *User* menyelesaikan suatu lagu, maka akan ditampilkan hasil skor yang diperoleh *User*.

Pada halaman menu mulai *game*, petunjuk, pengaturan, dan tentang terdapat tombol untuk kembali ke menu utama. Apabila *User* memilih tombol keluar, maka *User* akan keluar dari *game*.



Gambar 10. Activity Diagram

C. Tampilan Hasil

Dalam tahapan ini penulis menampilkan hasil dari permainan yang sudah bisa dijalankan di sistem operasi *Windows* dan *Linux*, dimulai dari tampilan paling awal yaitu tampilan menu utama yang kemudian akan diarahkan ke tampilan permainan yang terdiri dari tampilan pilihan daerah untuk memilih lagu, tampilan permainan, dan tampilan akhir.

1. Tampilan Menu Utama

Tampilan menu utama adalah tampilan pertama yang ditampilkan kepada pemain, pada gambar 11 terdapat 5 tombol yaitu tombol “Mulai Game” untuk memulai permainan, tombol “Petunjuk” untuk melihat cara bermain *game*, tombol “Pengaturan” untuk mengatur beberapa komponen dalam permainan, tombol “Tentang” untuk melihat keterangan dan informasi dari pengembang, dan tombol “Keluar” untuk keluar dari permainan.



Gambar 11. Tampilan Menu Utama

2. Tampilan Pilihan Daerah

Tampilan halaman ini akan digunakan sebagai antarmuka untuk memilih daerah dan mengetahui lagu yang akan dimainkan, terdapat juga informasi mengenai sejarah dari lagu daerah yang akan dimainkan. Lalu terdapat tombol “Item” yang menampilkan item atau benda yang pemain dapatkan setelah menyelesaikan suatu lagu di daerah tertentu.



Gambar 12. Tampilan Pilihan Daerah

Seperti yang dapat dilihat dari gambar 12 ini terdapat peta Indonesia yang juga terdapat beberapa lingkaran di setiap daerah di dalam peta, jika lingkaran tersebut di tekan akan muncul tampilan dari gambar 13 yang menampilkan judul lagu yang akan dimainkan beserta informasi mengenai lagu tersebut.



Gambar 13. Tampilan Informasi Lagu

Lalu jika pemain menekan tombol “Item” akan muncul tampilan seperti pada gambar 14 berikut.

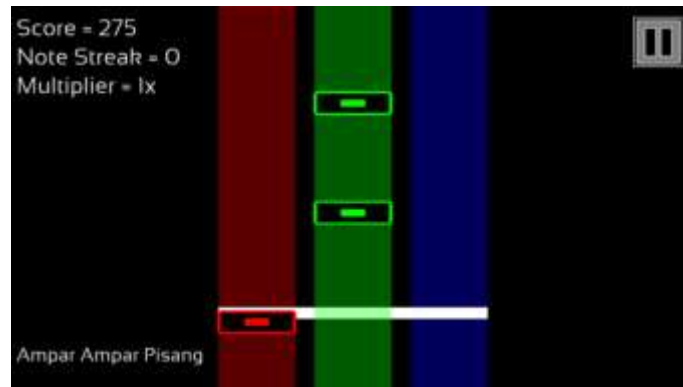


Gambar 14. Tampilan Pilihan Item

Dari gambar 14 di atas ditampilkan item-item yang berhasil didapatkan setelah pemain menyelesaikan suatu lagu dengan informasi dari benda tersebut jika pemain menekan tombol benda yang sudah didapatkan.

3. Tampilan *Gameplay*

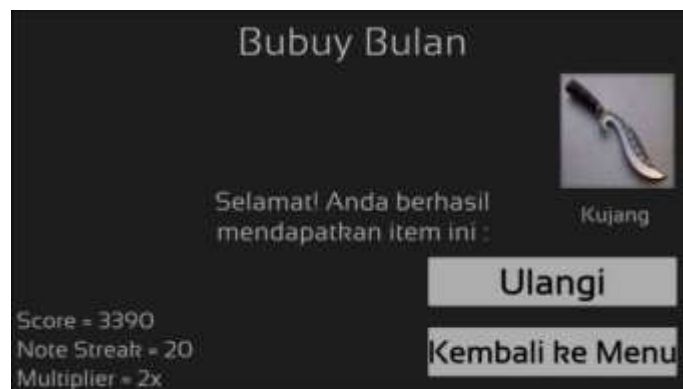
Tampilan ini merupakan bagian *gameplay* utama dari permainan ini dimana pemain harus menekan blok-blok yang muncul dari atas ke bawah secara tepat dengan menekan tepat dalam garis. Terdapat 3 tombol yang harus di tekan setiap blok menyentuh garis dengan menekan tombol *keyboard* J, K, dan L. Dari Gambar 15 ini juga terdapat 3 data yaitu data “*Score*” yang menampilkan skor yang di dapat dengan 10 skor per blok, data “*Note Streak*” di mana data ini menampilkan seberapa banyak blok yang pemain telah di tekan secara beruntun, jika pemain melewati satu atau lebih blok maka data Note Streak akan kembali ke 0, dan data “*Multiplier*” yang menampilkan kelipatan penambahan skor setiap pemain menekan blok dengan jumlah tertentu.



Gambar 15. Tampilan *Gameplay*

4. Tampilan Akhir

Tampilan dari gambar 16 ini akan muncul jika pemain telah menyelesaikan lagu terakhir dari permainan ini.



Gambar 16. Tampilan Akhir

Pemain akan ditampilkan teks bahwa pemain telah menyelesaikan permainan dan mendapatkan satu item berupa senjata tradisional dari daerah yang telah dimainkan sebagai koleksi. Dari tampilan ini pemain bisa memilih tombol Kembali ke Menu untuk kembali ke tampilan menu utama, atau memilih tombol Ulangi untuk mengulang kembali lagu yang telah dimainkan sebelumnya.

4. KESIMPULAN

Berdasarkan hasil pembahasan dan uji coba yang telah dilakukan, dapat disimpulkan bahwa:

- Unity Game Engine* mampu membuat *game* dengan mudah dan dapat mengekspor ke lebih dari satu platform.
- Rhythm game* yang telah di rancang dapat dimainkan di *platform Windows* dan *Linux* dengan lancar.

SARAN

Adapun saran yang penulis harapkan untuk menyempurnakan penelitian ini:

1. Adanya pengembangan fitur yang lebih luas seperti pilihan lagu yang lebih luas, kustomisasi *UI*, dan *Multiplayer* dengan perangkat berbeda.
2. *Game* ini juga di rilis ke *platform* lain dengan pasar yang lebih luas seperti *Android*, *iOS*, dan *macOS*.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada keluarga penulis dan Dosen / Staff / Pegawai Universitas Potensi Utama yang telah banyak memberikan masukan dan saran dalam penelitian ini.

REFERENSI

- [1] Ansari, K. D., & Rosnelly, R. (2020). Perancangan Aplikasi Game Edukasi Pengenalan Jenis Buah–Buahan Berbasis Multimedia. *Jurnal Mahasiswa Fakultas Teknik dan Ilmu Komputer*, 1(1), 915-926.
- [2] Steam Hardware & Software Survey. (2022, December). Retrieved January 16, 2023, from <https://store.steampowered.com/hwsurvey/>
- [3] Wikimedia Foundation. (2022, December 1). Usage share of operating systems. Wikipedia. Retrieved January 16, 2023, from https://en.wikipedia.org/wiki/Usage_share_of_operating_systems#Desktop_and_laptop_computers
- [4] Palumbo, A. (2018, April 4). Valve to continue investing into linux and Vulkan ecosystems via SteamOS. *Wccftech*. Retrieved January 16, 2023, from <https://wccftech.com/valve-continue-investing-linux-steamos/>
- [5] Wikimedia Foundation. (2023, January 11). Unity (Game Engine). Wikipedia. Retrieved January 16, 2023, from [https://en.wikipedia.org/wiki/Unity_\(game_engine\)](https://en.wikipedia.org/wiki/Unity_(game_engine))
- [6] Wikimedia Foundation. (2022, July 9). Rhythm game. Wikipedia. Retrieved January 16, 2023, from https://simple.wikipedia.org/wiki/Rhythm_game
- [7] Rebelo, R. R. (2016). Building a Music Rhythm Video Game, Universidade Técnica de Lisboa.
- [8] Alghadhari, D. R. (2020). Pengembangan Rhythm Game Berbasis Android dengan Unity 2D Sebagai Media Pembelajaran Seni Musik Gamelan, Universitas Sebelas Maret.
- [9] Rongas, B. (2016). A model-driven framework for educational game design. *International Journal of Serious Games*, 3(3).
- [10] Kambodji, A. S. (2016). Rancang Bangun Rhythm Game Sebagai Sarana Belajar Bermain Gitar, Universitas Brawijaya
- [11] Song, D. H., Kim, K. B., & Lee, J. H. (2019). Analysis and evaluation of Mobile Rhythm Games : Game structure and playability. *International Journal of Electrical and Computer Engineering (IJECE)*, 9(6), 5263.
- [12] DryWetMIDI. (n.d.). Retrieved January 16, 2023, from <https://melanchall.github.io/drywetmidi/>

- [13] Wikimedia Foundation. (2022, December 14). LMMS. Wikipedia. Retrieved January 16, 2023, from <https://en.wikipedia.org/wiki/LMMS>
- [14] Malik, I. (2022, January 8). Penjelasan User Interface Unity, Fungsi Menu Dan Tool. Oke Teknologi. Retrieved January 16, 2023, from <https://oketeknologi.com/penjelasan-user-interface-unity-fungsi-menu-dan-tool/>
- [15] Interface Overview. 3.1 Interface Overview - User manual. (n.d.). Retrieved January 16, 2023, from <https://docs.lmms.io/user-manual/3-navigating-lmms/general-interface>
- [16] Lano, K. (2005). The unified modelling language. Advanced Systems Design with Java, UML and MDA, 14–39.