# SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB

SKRIPSI



OLEH:

SLAMET SUPRIYADI

211011400351

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PAMULANG**

**TANGERANG SELATAN**

**2025**

# SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB

**SKRIPSI**

Diajukan Untuk Melengkapi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer



OLEH:

SLAMET SUPRIYADI

211011400351

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PAMULANG**

**TANGERANG SELATAN**

**2025**

# LEMBAR PERNYATAAN

Yang bertanda tangan di bawah ini:

Nama : SLAMET SUPRIYADI

NIM : 211011400351

Program Studi : Teknik Informatika

Fakultas : Ilmu Komputer

Jenjang Pendidikan : Strata 1

Menyatakan bahwa skripsi yang saya buat dengan judul:

SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB

1. Merupakan hasil karya tulis ilmiah sendiri, bukan merupakan karya yang pernah diajukan untuk memperoleh gelar akademik oleh pihak lain, dan bukan merupakan hasil plagiat.
2. Saya ijinkan untuk dikelola oleh Universitas Pamulang sesuai dengan norma hukum dan etika yang berlaku.

Pernyataan ini saya buat dengan penuh tanggung jawab dan saya bersedia menerima konsekuensi apapun sesuai aturan yang berlaku apabila di kemudian hari pernyataan ini tidak benar.

|  |
| --- |
| Tangerang Selatan, .........................2025 |
|  |
|  |
| (Slamet Supriyadi) |

# LEMBAR PERSETUJUAN

|  |  |  |
| --- | --- | --- |
| NIM | : | 211011400351 |
| Nama | : | SLAMET SUPRIYADI |
| Program Studi | : | TEKNIK INFORMATIKA |
| Fakultas | : | ILMU KOMPUTER |
| Jenjang Pendidikan | : | STRATA 1 |
| Judul Skripsi | : | SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB |

Skripsi ini telah diperiksa dan disetujui oleh pembimbing untuk persyaratan sidang skripsi

Tangerang Selatan, ..............................

Pembimbing

|  |
| --- |
| Pembimbing, S.Kom. |
| NIDN: 0 |

Mengetahui,  
Ketua Program Studi Teknik Informatika

|  |
| --- |
| Kaprodi, S.Kom., M.Kom. |
| NIDN: 0 |

# LEMBAR PENGESAHAN

|  |  |  |
| --- | --- | --- |
| NIM | : | 211011400351 |
| Nama | : | SLAMET SUPRIYADI |
| Program Studi | : | TEKNIK INFORMATIKA |
| Fakultas | : | ILMU KOMPUTER |
| Jenjang Pendidikan | : | STRATA 1 |
| Judul Skripsi | : | SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB |

Skripsi ini telah dipertahankan di hadapan dewan penguji ujian skripsi fakultas Ilmu Komputer, program studi Teknik Informatika dan dinyatakan LULUS.

Tangerang Selatan, ..............................

|  |  |
| --- | --- |
| Penguji I | Penguji II |
|  |  |
|  |  |
| Nama Penguji 1 | Nama Penguji 2 |
| NIDN: - | NIDN: - |

Pembimbing

|  |
| --- |
| Nama Pembimbing, S.Kom. |
| NIDN: 0 |

Mengetahui,  
Ketua Program Studi Teknik Informatika

|  |
| --- |
| Nama Kaprodi TI, S.Kom., M.Kom. |
| NIDN: 0 |

# *ABSTRACT*

*Japanese language has become one of the foreign languages increasingly favored by students and university students in Indonesia..*

*Keywords: Educational Games, Japanese Language, 3D Simulation, Android, Unity Engine, Multimedia Development Life Cycle (MDLC)*

xi+111 pages; 72 figures; 19 tables; 4 attachments

Bibliography: 40 (2012-2023)

# ABSTRAK

Bahasa Jepang kini menjadi salah satu bahasa asing yang sedang digemari pelajar hingga mahasiswa di Indonesia.

Kata Kunci: Game Edukasi, Bahasa Jepang, Simulasi 3D, *Android*, *Unity Engine*, *Multimedia Development Life Cycle (MDLC)*

xi+111 halaman; 72 gambar; 19 tabel; 4 lampiran  
Daftar acuan: 40 (2012-2023)

# KATA PENGANTAR

Puji syukur Alhamdulillah ke hadirat Allah SWT yang telah melimpahkan segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi ini yang berjudul **“SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB”**.

Skripsi ini disusun untuk memenuhi salah satu syarat kelulusan program studi strata satu (S1) pada program studi Teknik Informatika di Universitas Pamulang.

Penulis mengucapkan terima kasih kepada:

1. **Allah SWT** atas nikmat iman, Islam, kesehatan, dan umur panjang.
2. Bapak **Dr. Pranoto, S.E., M.M.,** selaku Ketua Yayasan Sasmita Jaya.
3. Bapak **Dr. E. Nurzaman A.M., MM., M.Si.,** selaku Rektor Universitas Pamulang.
4. Bapak **Yan Mitha Djaksana, S.Kom., M.Kom.,** selaku Dekan Fakultas Teknik Universitas Pamulang.
5. Bapak **Dr. Eng. Ahmad Musyafa, S.Kom., M.Kom.,** selaku Ketua Program Studi Teknik Informatika Universitas Pamulang.
6. Ibu **Elfi Fauziah, S.Si., M.Pd., M.Si.,** selaku Dosen Pembimbing.
7. Kedua **orang tua** yang selalu mendoakan dan mendukung.
8. Seluruh **dosen**, **kerabat** dan **sahabat** serta **teman-teman** **UNPAM** seperjuangan.

Semoga Allah SWT membalas segala kebaikan. Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Semoga skripsi ini bermanfaat bagi pembaca.

|  |
| --- |
| Tangerang Selatan, 2 September 2025 |
| Slamet Supriyadi |

# DAFTAR ISI

[SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB i](#_Toc207747935)

[SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB ii](#_Toc207747936)

[LEMBAR PERNYATAAN iii](#_Toc207747937)

[LEMBAR PERSETUJUAN iv](#_Toc207747938)

[LEMBAR PENGESAHAN v](#_Toc207747939)

[*ABSTRACT* vi](#_Toc207747940)

[ABSTRAK vii](#_Toc207747941)

[KATA PENGANTAR viii](#_Toc207747942)

[DAFTAR ISI ix](#_Toc207747943)

[DAFTAR GAMBAR xii](#_Toc207747944)

[DAFTAR TABEL xiii](#_Toc207747945)

[DAFTAR LAMPIRAN xiv](#_Toc207747946)

[BAB I PENDAHULUAN 1](#_Toc207747947)

[1.1. Latar Belakang 1](#_Toc207747948)

[1.2. Identifikasi Masalah 2](#_Toc207747949)

[1.3. Rumusan Masalah 2](#_Toc207747950)

[1.4. Batasan Penelitian 3](#_Toc207747951)

[1.5. Tujuan Penelitian 3](#_Toc207747952)

[1.6. Manfaat Penelitian 3](#_Toc207747953)

[1.7. Metodologi Penelitian 4](#_Toc207747954)

[1.8. Sistematika Penulisan 7](#_Toc207747963)

[BAB II LANDASAN TEORI 8](#_Toc207747964)

[2.1. Penelitian yang Relevan 8](#_Toc207747966)

[2.2. Tinjauan Pustaka 10](#_Toc207747967)

[2.2.1. Sistem Pendukung Keputusan (SPK) 10](#_Toc207747968)

[2.2.2. Logika *Fuzzy* 10](#_Toc207747969)

[2.2.3. Metode *Fuzzy* Tsukamoto 10](#_Toc207747975)

[2.2.4. *Website* 11](#_Toc207747976)

[2.2.5. *Database* 11](#_Toc207747977)

[2.2.6. *Unified Modeling Language* (UML) 12](#_Toc207747978)

[2.2.7. Teori Perancangan Basis Data 17](#_Toc207747979)

[2.2.8. Aplikasi Pendukung 18](#_Toc207747980)

[2.3. Aplikasi Pendukung 19](#_Toc207747981)

[2.3.1. *Unity Engine* 19](#_Toc207747982)

[2.3.2. *Mixamo* 19](#_Toc207747983)

[2.3.3. *Adobe Photoshop* 19](#_Toc207747984)

[2.3.4. *Figma* 20](#_Toc207747985)

[2.3.5. *Microsoft Visual Studio* 20](#_Toc207747986)

[2.4. Teori Pengujian Sistem 21](#_Toc207747987)

[2.4.1. Sistem *Black Box Testing* 21](#_Toc207747988)

[2.4.2. *User Response* (Kuesioner) 21](#_Toc207747989)

[BAB III ANALISA DAN PERANCANGAN 22](#_Toc207747990)

[3.1. Analisa Sistem 22](#_Toc207747994)

[3.1.1. Analisa Sistem Berjalan 23](#_Toc207747995)

[3.1.2. Analisa Sistem Usulan 24](#_Toc207747996)

[3.2. Arsitektur Perangkat Lunak 26](#_Toc207747997)

[3.3. Metode Pengembangan 26](#_Toc207747998)

[1.2.1. Konsep (*Concept*) 27](#_Toc207748002)

[1.2.2. Perancangan (*Design*) 27](#_Toc207748003)

[1.2.3. Pengumpulan Bahan (*Material Collecting*) 29](#_Toc207748004)

[1.2.4. Perakitan (*Assembly*) 29](#_Toc207748005)

[1.2.5. Pengujian Aplikasi (*Testing*) 31](#_Toc207748006)

[1.2.6. Distribusi (*Distribution*) 32](#_Toc207748007)

[3.4. Perancangan *Unified Modeling Language (UML)* 32](#_Toc207748008)

[3.4.1. *Use Case Diagram* 33](#_Toc207748009)

[3.4.2. *Activity Diagram* 35](#_Toc207748010)

[3.4.3. *Sequence Diagram* 48](#_Toc207748011)

[1.4.4. *Class Diagram* 59](#_Toc207748017)

[3.5. *User Interface* 60](#_Toc207748018)

[BAB IV IMPLEMENTASI DAN PENGUJIAN 67](#_Toc207748019)

[4.1. Spesifikasi 67](#_Toc207748021)

[2.1.1. Spesifikasi Perangkat Lunak 67](#_Toc207748024)

[2.1.2. Spesifikasi Perangkat Keras 67](#_Toc207748025)

[4.2. Implementasi Program 68](#_Toc207748026)

[4.2.1. Tampilan *Splash Screen* 68](#_Toc207748033)

[4.2.2. Tampilan *Main Menu* 69](#_Toc207748034)

[4.2.3. Tampilan Pengaturan 69](#_Toc207748035)

[4.2.4. Tampilan Kredit 70](#_Toc207748036)

[4.2.5. Tampilan Keluar Permainan 70](#_Toc207748037)

[4.2.6. Tampilan *Character Selection* 71](#_Toc207748038)

[4.2.7. Tampilan *Gameplay* 71](#_Toc207748039)

[4.2.8. Tampilan *Gameplay* – Pembelajaran *Online* 72](#_Toc207748040)

[4.2.9. Tampilan *Classroom* 72](#_Toc207748041)

[4.2.10. Tampilan *Classroom –* Pembelajaran *Offline* 73](#_Toc207748042)

[4.2.11. Tampilan *Classroom –* Ujian 73](#_Toc207748043)

[4.3. Pengujian Sistem 74](#_Toc207748044)

[4.3.1. *Functional Testing* 74](#_Toc207748045)

[4.3.2. Kuesioner *User Acceptance Testing* 84](#_Toc207748046)

[4.4. Uji Normalitas 91](#_Toc207748047)

[BAB V PENUTUP 96](#_Toc207748048)

[5.1. Kesimpulan 96](#_Toc207748050)

[5.2. Saran 96](#_Toc207748051)

[DAFTAR PUSTAKA 97](#_Toc207748052)

[LAMPIRAN 100](#_Toc207748053)

# DAFTAR GAMBAR

# DAFTAR TABEL

# DAFTAR LAMPIRAN

# BAB I PENDAHULUAN

## Latar Belakang

Dalam era industri otomotif yang semakin kompetitif, kualitas layanan menjadi salah satu faktor kunci yang menentukan keberhasilan sebuah bengkel. Salah satu aspek penting dalam menjaga kualitas layanan adalah pemilihan teknisi yang tepat. Teknisi yang berkualitas tidak hanya memiliki keterampilan teknis yang memadai, tetapi juga pengalaman dan kemampuan untuk menyelesaikan masalah dengan efisien. Namun, proses pemilihan teknisi sering kali dihadapkan pada tantangan, seperti ketidakpastian dalam penilaian kualifikasi dan keterampilan, serta subjektivitas dalam pengambilan keputusan.

Bengkel RD Speed Joglo, sebagai salah satu penyedia layanan otomotif, menyadari pentingnya memiliki sistem yang dapat membantu dalam proses pemilihan teknisi. Dengan banyaknya teknisi yang tersedia, manajemen perlu mempertimbangkan berbagai kriteria, seperti pengalaman kerja, keahlian spesifik, dan performa sebelumnya. Oleh karena itu, diperlukan suatu sistem yang dapat mengolah data dan memberikan rekomendasi yang objektif dan akurat.

Metode *Fuzzy* Tsukamoto dipilih karena ada beberapa kelebihan yang menonjol yaitu dapat mendefinisikan nilai yang kabur dari inputan penilaian, membangun, serta mengaplikasikan pengalaman-pengalaman dari pakar secara langsung sehingga tidak melalui proses *training* (Hutahaean & Hutagalung, 2022). Metode ini memungkinkan pengolahan data kuantitatif, sehingga dapat menghasilkan keputusan yang lebih baik dalam pemilihan teknisi. Selain itu, dengan mengembangkan sistem berbasis *web*, aksesibilitas dan kemudahan penggunaan bagi manajemen bengkel dapat ditingkatkan, sehingga proses pengambilan keputusan menjadi lebih efisien.

Dengan latar belakang tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem penunjang keputusan yang dapat membantu Bengkel RD Speed Joglo dalam memilih teknisi yang paling sesuai, sehingga dapat meningkatkan kualitas layanan dan kepuasan pelanggan.

Berdasarkan latar belakang di atas, maka penelitian yang diambil pada penelitian ini adalah **“SISTEM PENUNJANG KEPUTUSAN DALAM MENENTUKAN PERINGKAT TEKNISI BERDASARKAN TINGKAT KERUSAKAN DI BENGKEL RD SPEED JOGLO MENGGUNAKAN METODE FUZZY TSUKAMOTO BERBASIS WEB”**. Diharapkan dari pembuatan aplikasi ini, memberikan solusi yang efektif dalam membantu manajemen bengkel untuk menentukan teknisi yang paling tepat berdasarkan kriteria yang telah ditentukan.

## Identifikasi Masalah

Berdasarkan latar belakang yang telah dipaparkan, dapat diidentifikasi beberapa permasalahan sebagai berikut:

1. Proses pemilihan teknisi di Bengkel RD Speed Joglo masih dilakukan secara manual sehingga rentan terhadap subjektivitas.
2. Tidak adanya sistem pendukung keputusan yang mampu mengolah data teknisi berdasarkan kriteria tertentu secara objektif.
3. Kesulitan manajemen dalam menentukan teknisi yang paling sesuai dengan jenis kerusakan kendaraan secara cepat dan akurat.

## Rumusan Masalah

Berdasarkan latar belakang dan identifikasi masalah, maka rumusan masalah penelitian ini adalah:

1. Bagaimana merancang sistem penunjang keputusan untuk menentukan peringkat teknisi berdasarkan tingkat kerusakan kendaraan di Bengkel RD Speed Joglo?
2. Bagaimana menerapkan metode *Fuzzy* Tsukamoto dalam proses perhitungan dan penentuan peringkat teknisi?
3. Bagaimana mengimplementasikan sistem tersebut dalam bentuk aplikasi berbasis *web* agar mudah diakses oleh manajemen bengkel?

## Batasan Penelitian

Dalam penelitian ini, terdapat sejumlah batasan yang telah dirinci dan difokuskan sebagai berikut:

1. Sistem hanya mencakup proses penilaian teknisi berdasarkan kriteria yang telah ditentukan, yaitu: Tingkat Kerusakan yang Ditangani, Jumlah Kerusakan yang Diselesaikan dan Waktu Penyelesaian.
2. Hasil yang dihasilkan dari sistem adalah tingkat teknisi yang akan menjadi rekomendasi bagi manajemen bengkel.
3. Data teknisi yang digunakan merupakan data dari Bengkel RD Speed Joglo.
4. Sistem dibangun berbasis *web* dengan menggunakan metode *Fuzzy* Tsukamoto sebagai metode pengambilan keputusan.
5. Penelitian ini tidak membahas aspek lain di luar kriteria yang disebutkan, seperti penggajian, rekrutmen, atau penjadwalan teknisi.

## Tujuan Penelitian

Penelitian ini bertujuan untuk mengatasi permasalahan yang telah diidentifikasi. Tujuan dari penelitian ini adalah sebagai berikut:

1. Merancang sistem penunjang keputusan yang mampu memberikan rekomendasi teknisi secara objektif berdasarkan tiga kriteria utama.
2. Menerapkan metode *Fuzzy* Tsukamoto dalam menentukan peringkat teknisi.
3. Menghasilkan aplikasi berbasis *web* yang dapat membantu manajemen bengkel dalam menentukan teknisi yang tepat sesuai tingkat kerusakan.

## Manfaat Penelitian

Penelitian ini diharapkan dapat diambil beberapa manfaat yang mencakup dua hal pokok berikut:

1. Manfaat Untuk Penulis
2. Penelitian ini diharapkan menjadi sarana penerapan ilmu yang telah dipelajari, khususnya dalam penerapan metode *Fuzzy* Tsukamoto dan pengembangan sistem berbasis *web*.
3. Penelitian ini diharapkan dapat memberikan kontribusi dalam bidang pendidikan dan teknologi dengan menggabungkan konsep logika *fuzzy* dan penerapan sistem pendukung keputusan berbasis *web*, sehingga dapat meningkatkan pemahaman dan keterampilan penulis dalam implementasi metode *Fuzzy* Tsukamoto.
4. Penelitian ini merupakan persyaratan yang harus dipenuhi untuk menyelesaikan program Sarjana (S1).
5. Manfaat Untuk Manajemen Bengkel
6. Mempermudah manajemen dalam menentukan teknisi yang paling sesuai berdasarkan tingkat kerusakan, jumlah kerusakan yang diselesaikan, dan waktu penyelesaian.
7. Membantu manajemen mengurangi subjektivitas dalam pengambilan keputusan dengan sistem yang memberikan rekomendasi berbasis data.
8. Meningkatkan efisiensi proses pemilihan teknisi sehingga kualitas layanan dan kepuasan pelanggan dapat ditingkatkan.

## Metodologi Penelitian

Dalam perancangan dan implementasi Sistem Penunjang Keputusan untuk Menentukan Peringkat Teknisi di Bengkel RD Speed Joglo menggunakan metode *Fuzzy* Tsukamoto, penulis menerapkan metodologi penelitian sebagai berikut:

2. Metode Pengumpulan Data

Untuk memperoleh informasi yang akurat, penulis menggunakan metode penelitian kuantitatif dengan data numerik yang dapat diolah menjadi masukan (input) bagi sistem. Metode pengumpulan data yang digunakan meliputi:

1. Metode Observasi

Melakukan pengamatan langsung pada proses penanganan kendaraan di Bengkel RD Speed Joglo untuk mendapatkan data terkait tingkat kerusakan, jumlah kerusakan yang diselesaikan, dan waktu penyelesaian.

1. Metode Kuesioner

Kuesioner diberikan kepada pihak manajemen dan teknisi di Bengkel RD Speed Joglo untuk mengumpulkan data terkait kriteria penilaian teknisi, seperti tingkat kerusakan yang ditangani, jumlah kerusakan yang diselesaikan, dan waktu penyelesaian rata-rata.

1. Metode Studi Pustaka

Mengumpulkan data sekunder dari jurnal, dan penelitian terdahulu terkait Sistem Penunjang Keputusan, logika *fuzzy*, serta metode *Fuzzy* Tsukamoto.

1. Metode Perancangan Sistem

Perancangan sistem dilakukan menggunakan model Waterfall. Waterfall dipilih karena proses pengembangan sistem dilakukan secara terstruktur dan berurutan, dimulai dari tahap analisis hingga tahap implementasi. Tahapan yang dilakukan adalah:

1. Analisis Kebutuhan *(Requirement Analysis)*

Menentukan kebutuhan sistem, mengidentifikasi kriteria penilaian (tingkat kerusakan, jumlah kerusakan, waktu penyelesaian), serta mendefinisikan kebutuhan *fungsional* dan *non-fungsional* dari sistem.

1. Desain Sistem *(System Design)*

Perancangan sistem dilakukan secara menyeluruh mulai dari perancangan basis data hingga antarmuka pengguna. Tahap ini dimulai dengan pembuatan *Entity Relationship Diagram* (ERD) untuk menggambarkan hubungan antar entitas, kemudian ditransformasikan ke *Logical Record Structure* (LRS) dan dinormalisasi agar terhindar dari redundansi data. Setelah itu disusun spesifikasi basis data yang berisi detail *atribut*, *tipe* data, dan *key*. Selanjutnya dilakukan perancangan menggunakan *Unified Modelling Language* (UML) yang mencakup *Use Case Diagram* untuk memodelkan interaksi pengguna dengan sistem, *Activity Diagram* untuk menggambarkan alur proses, *Sequence Diagram* untuk menjelaskan interaksi antar objek secara berurutan, dan *Class Diagram* untuk memodelkan struktur kelas. Tahap perancangan diakhiri dengan pembuatan antarmuka pengguna yang dirancang intuitif dan mudah digunakan sehingga mempermudah manajemen bengkel dalam menginput data serta memahami hasil perhitungan peringkat teknisi.

1. Implementasi *(Implementation)*

Tahap implementasi dilakukan dengan membangun sistem berbasis *web* menggunakan bahasa pemrograman seperti PHP dengan basis data MySQL. Pada tahap ini metode *Fuzzy* Tsukamoto diintegrasikan ke dalam sistem untuk melakukan perhitungan peringkat teknisi berdasarkan data yang telah diinputkan.

1. Pengujian *(Testing)*

Setelah implementasi selesai, sistem diuji menggunakan metode *black box testing* untuk memastikan setiap fitur berjalan sesuai kebutuhan dan hasil perhitungan yang dihasilkan sistem sesuai dengan perhitungan manual menggunakan metode *Fuzzy* Tsukamoto.

1. Penerapan *(Deployment)*

Sistem yang telah lulus tahap pengujian kemudian diterapkan di lingkungan Bengkel RD Speed Joglo sehingga dapat langsung digunakan oleh manajemen untuk membantu proses pengambilan keputusan dalam pemilihan teknisi. Sistem ini di *hosting* pada alamat <https://hakolabdev.com/spk_tingkat_teknisi> sehingga dapat diakses secara *online* dengan mudah oleh pihak manajemen.

1. Pemeliharaan *(Maintenance)*

Setelah sistem digunakan, dilakukan proses pemeliharaan secara berkala untuk memperbaiki *bug* yang ditemukan serta melakukan penyesuaian terhadap kriteria atau bobot penilaian teknisi apabila terjadi perubahan kebutuhan dari pihak manajemen.

## Sistematika Penulisan

Dalam penyusunan skripsi ini, sistematika penulisan diatur dan disusun ke dalam lima bab yang masing-masing terdiri dari subbab. Adapun urutan sistematika penulisan adalah sebagai berikut:

**BAB I PENDAHULUAN**

Bab ini berisi uraian mengenai latar belakang, identifikasi masalah, rumusan masalah, batasan penelitian, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

**BAB II LANDASAN TEORI**

Bab ini memuat teori-teori yang mendukung penelitian, antara lain konsep Sistem Pendukung Keputusan (SPK), logika *fuzzy*, metode *Fuzzy* Tsukamoto, serta tinjauan pustaka dari penelitian-penelitian sebelumnya yang relevan.

**BAB III ANALISA DAN PERANCANGAN**

Bab ini menjelaskan analisis kebutuhan sistem, perancangan basis data, perancangan *Unified Modelling Language* (UML) seperti *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*, serta perancangan antarmuka pengguna (*User Interface*).

**BAB IV IMPLEMENTASI DAN PENGUJIAN**

Bab ini berisi implementasi sistem berbasis *web* yang telah dirancang, integrasi metode *Fuzzy* Tsukamoto, serta pengujian sistem menggunakan metode *black box testing* untuk memastikan hasil perhitungan sesuai dengan perhitungan manual.

**BAB V PENUTUP**

Bab ini merupakan bagian akhir skripsi yang berisi kesimpulan dari hasil penelitian dan saran yang dapat digunakan untuk pengembangan sistem di masa mendatang.

# BAB II LANDASAN TEORI



## Penelitian yang Relevan

Dalam bab ini, dibahas penelitian-penelitian sebelumnya yang memiliki keterkaitan dengan topik penelitian dan mendukung pengembangan sistem penunjang keputusan menggunakan metode *Fuzzy* Tsukamoto.

Penelitian pertama dilakukan oleh (Hutahaean & Hutagalung, 2022) yang berjudul “Sistem Pendukung Keputusan Pemilihan Teknisi Terbaik Menggunakan Metode *Fuzzy* Tsukamoto”. Penelitian ini bertujuan membantu manajemen bengkel dalam menentukan teknisi terbaik berdasarkan kriteria pengalaman kerja, keterampilan, dan kecepatan menyelesaikan pekerjaan. Hasil penelitian menunjukkan bahwa metode *Fuzzy* Tsukamoto mampu memberikan hasil yang lebih objektif dibandingkan penilaian manual.

Penelitian kedua adalah penelitian oleh (Adoe dkk., 2022) berjudul “Penerapan Metode *Fuzzy* Tsukamoto Dalam Penentuan Jumlah Produksi Roti (Studi Kasus: Dwi Jaya Bakery Kupang)”. Dalam penelitian ini, menerapkan metode *Fuzzy* Tsukamoto melalui tiga tahapan utama, yaitu fuzzifikasi, inferensi, dan defuzzifikasi, untuk menentukan jumlah produksi roti berdasarkan data permintaan dan persediaan. Validasi dilakukan dengan metode MAPE pada 28 data uji dan menghasilkan tingkat kesalahan kurang dari 10% pada semua jenis roti, sehingga metode ini dinilai akurat dan dapat membantu pengambilan keputusan jumlah produksi. Relevansi penelitian ini terhadap penelitian yang dilakukan penulis adalah menunjukkan bahwa metode *Fuzzy* Tsukamoto efektif untuk pengambilan keputusan berbasis data dan dapat diadaptasi pada konteks penentuan peringkat teknisi.

Penelitian ketiga adalah penelitian yang ditulis oleh (Sitepu dkk., 2020) yang memiliki judul “Implementasi Sistem Pendukung Keputusan Kepuasan Pelayanan Pelanggan Showroom Honda Simpang Pemda Menggunakan Metode *Fuzzy* Tsukamoto”. Dalam jurnal ini penerapan metode *Fuzzy* Tsukamoto untuk menentukan tingkat kepuasan pelanggan berdasarkan lima kriteria: Pelayanan, Keahlian, Fasilitas, Kelengkapan, dan Komunikasi. Sistem berbasis Visual Basic dan Microsoft Access ini melalui tahapan fuzzifikasi, pembentukan *rule base*, inferensi (*min–max*), dan defuzzifikasi (*average*) sehingga menghasilkan nilai *crisp* yang menunjukkan kategori kepuasan (Tidak Puas, Puas, atau Sangat Puas). Hasil penelitian menunjukkan sistem dapat membantu manajemen *showroom* mengetahui bagian pelayanan yang perlu ditingkatkan. Relevansi penelitian ini terletak pada penerapan *Fuzzy* Tsukamoto dalam pengambilan keputusan multi-kriteria yang dapat diadaptasi untuk penentuan peringkat teknisi, meskipun keterbatasannya adalah sistem belum berbasis *web* dan evaluasi kuantitatif terbatas.

Penelitian keempat yang dilakukan oleh (Miftah dkk., 2024) yang berjudul “Sistem Pendukung Keputusan Untuk Mencari Calon Karyawan Dari Mahasiswa Yang Telah Lulus Menggunakan Metode *Fuzzy* Tsukamoto”. Penelitian ini membahas pembangunan sistem berbasis *web* untuk membantu perusahaan memilih calon karyawan yang sesuai dengan kriteria yang ditentukan, menggunakan metode *Fuzzy* Tsukamoto dengan tiga tahap utama: fuzzifikasi, inferensi, dan defuzzifikasi. Hasil penelitian menunjukkan bahwa sistem bekerja dengan baik dan output yang dihasilkan 100% sesuai dengan perhitungan manual, sehingga mendukung akurasi dan reliabilitas metode. Relevansi penelitian ini adalah penerapan *Fuzzy* Tsukamoto untuk pengambilan keputusan berbasis data dalam konteks pemilihan individu, serupa dengan penentuan peringkat teknisi, dengan keunggulan implementasi berbasis *web* yang memudahkan pengguna. Namun, penelitian ini belum menyajikan evaluasi kuantitatif seperti MAPE dan tidak menjelaskan secara rinci jumlah sampel atau data uji yang digunakan.

Terakhir penelitian kelima adalah penelitian yang ditulis oleh (Mustika, 2021) yang memiliki judul “Metode *Fuzzy* Sugeno Untuk Penilaian Kinerja Guru Smk Tamansiswa 3 Jakarta”. Penelitian ini membahas pembangunan sistem penilaian kinerja guru menggunakan logika *fuzzy* Sugeno untuk membantu manajemen sekolah menentukan guru berprestasi berdasarkan 11 kriteria, seperti kesiapan materi, penggunaan media pembelajaran, pengendalian kelas, hingga motivasi siswa. Sistem ini melalui tahapan fuzzifikasi, aplikasi aturan (*rule*), komposisi aturan, dan defuzzifikasi untuk menghasilkan skor akhir yang merepresentasikan kinerja guru (baik, cukup, buruk). Hasil pengujian menunjukkan tingkat akurasi 98,92%, sehingga sistem ini efektif dalam memberikan penilaian yang objektif dan konsisten. Namun, penelitian ini terbatas pada satu sekolah dan belum berbasis *web*, sehingga penggunaannya belum optimal untuk implementasi yang lebih luas.

## Tinjauan Pustaka

Tinjauan pustaka ini mencakup berbagai aspek yang mendukung pemahaman pada penelitian ini. Tinjauan pustaka ini akan menguraikan beberapa topik yang relevan untuk memahami konsep, teori, serta konteks penelitian dengan lebih mendalam.

### Sistem Pendukung Keputusan (SPK)

Sistem pendukung keputusan adalah sistem informasi yang dibangun guna membantu aktivitas manajerial di dalam menangani permasalahan yang dihadapi (Suarnatha, 2023).

### Logika *Fuzzy*

Logika *Fuzzy* merupakan komponen pembentuk *softcomputing* yang banyak digunakan berbagai perangkat lunak. Pada logika *fuzzy*, suatu nilai dapat bernilai benar (1) atau salah (0) (Verdian dkk., 2023).



### Metode *Fuzzy* Tsukamoto

Metode *Fuzzy* Tsukamoto merupakan metode sistem pengambil keputusan yang menggunakan aturan atau *rules* berbentuk “sebab-akibat” atau “*if-then*”. Terdapat 3 tahap dalam metode ini yaitu: fuzzifikasi, inferensi *fuzzy*, dan defuzzifikasi. Fuzzifikasi didefinisikan sebagai *input*, di mana nilai *true* yang sebenarnya (*input crisp*) diubah menjadi *input* *fuzzy*, dalam bentuk nilai lingustik berdasarkan fungsi keanggotaan yang disimpan pada basis pengetahuan (kumpulan *rules*). Inferensi, yaitu proses mengubah *input* *fuzzy* menjadi *output* *fuzzy* dengan cara mengikuti *rules* (“*if-then*”) yang telah ditetapkan pada basis pengetahuan *fuzzy*. Defuzzifikasi merupakan proses mengubah hasil dari tahap inferensi menjadi keluaran yang bernilai tegas (*crisp*) menggunakan fungsi keanggotaan yang telah ditetapkan (Gloria & Sediyono, 2022).

### *Website*

*Website* atau *web* merupakan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk digital, baik berupa teks, gambar, video, audio dan animasi lainya yang disediakan melalui jalur koneksi internet. Lebih jelasnya, *website* merupakan halaman–halaman yang berisi informasi yang ditampilkan oleh browser seperti Mozilla, Firefox, Google Chrome atau yang lainnya (Arafat dkk., 2022).

Dari pengelolaan dan pembaruan konten, *website* terbagi menjadi dua kelompok yaitu:

1. *Website* Statis

*Website* statis merupakan jenis *website* yang isinya tidak berubah kecuali pengembang melakukan perubahan secara manual. *Website* statis seringkali cocok untuk *website* dengan konten yang tidak sering berubah, seperti *website* profil bisnis, *website* pribadi, atau *website* portofolio.

1. *Website* Dinamis

*Website* dinamis merupakan jenis *website* yang isinya dapat berubah secara otomatis sesuai permintaan pengguna atau berdasarkan informasi yang disimpan dalam *database*. *website* dinamis sering kali digunakan untuk membuat situs *web* *e-commerce*, forum, jejaring sosial, atau *web* aplikasi bisnis.

### *Database*

*Database* adalah kumpulan informasi yang disimpan secara sistematis di dalam komputer sehingga dapat dikendalikan oleh program komputer untuk mengambil informasi dari *database* (Aswiputri, 2022).

## Teori Perancangan Basis Data

Dalam perancangan basis data yang dijelaskan pada penelitian ini, penulis membahas konsep dan model-model yang akan digunakan untuk merancang struktur basis data.

### Entity Relationship Diagram (ERD)

*Entity Relationship* Diagram (ERD) adalah diagram berbentuk notasi grafis yang berada dalam pembuatan *database* yang menghubungkan antara data satu dengan yang lain. Fungsi ERD adalah sebagai alat bantu dalam pembuatan *database* dan memberikan gambaran bagaimana kerja *database* yang akan dibuat (Afiifah dkk., 2022).

**Tabel 2. 5 Simbol *Entity Relationship* Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
|  |  | *Associate* | Menunjukan dua hubungan antara dua objek. |
|  |  | *Interface* | Menggambarkan sebuah kontak yang menetapkan daftar metode tanpa adanya implementasi yang spesifik. |
|  |  | *Table* | Menggambarkan sebagai entitas yang terkait dengan struktur penyimpanan data. |

### Logical Recrod Structure (LRS)

LRS (*Logical Record Structure*) adalah representasi dari struktur *record-record* pada tabel-tabel yang terbentuk dari hasil antar himpunan entitas. Menentukan kardinalitas, jumlah tabel dan *Foreign Key* (FK) (Hasan & Nurlelah, 2020).

## Unified Modeling Language (UML)

UML (*Unified Modeling Language*) adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. Diagram UML yang sering digunakan adalah *use case* diagram, *activity* diagram, *sequence* diagram, class diagram, *state machine* diagram dan *component* diagram (Marthiawati dkk., 2024).

### *Use Case* Diagram

*Use case* diagram adalah sebuah diagram yang menunjukkan hubungan antara *actors* dan *use cases*. Digunakan untuk analisis dan 8 desain sebuah sistem (Arianti, Fa’izi, Adam, & Wulandari, 2022).

**Tabel 2. 1 Simbol Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
|  |  | *Use case* | Mempresentasikan fungsi atau skenario yang dapat dilakukan sistem. |
|  |  | *Actor* | Menggambarkan Entitas luar yang berinteraksi dengan sistem. |
|  |  | *Association* | Hubungan antara dua atau lebih *use case* dalam suatu sistem yang memiliki ketergantungan satu sama lain. |
|  |  | *Include* | Menggambarkan situasi satu *use case* memasukan fungsionalitas dari *use case* lainnya. |
|  |  | *Extend* | Menggambarkan situasi dimana suatu fungsionalitas tambahan dapat ditambahkan ke dalam *use case* utama berdasarkan skenario tertentu. |
|  |  | *System* | Merupakan garis kotak yang mengelilingi kumpulan *use case* untuk menunjukan batas sistem, memisahkan antara sistem yang dianalisis dengan entitas luar. |

### *Activity* Diagram

*Activity* diagram merupakan rancangan aliran aktivitas atau aliran kerja dalam sebuah sistem yang akan dijalankan. *Activity* diagram juga digunakan untuk mendefinisikan atau mengelompokan aluran tampilan dari sistem tersebut. *Activity* diagram memiliki komponen dengan bentuk tertentu yang dihubungkan dengan tanda panah. Panah tersebut mengarah ke-urutan aktivitas yang terjadi dari awal hingga akhir (Simare Mare & Yana, 2022).

**Tabel 2. 2 Simbol Activity Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
|  |  | *Initial* | Merupakan titik awal dari alur kerja. Menandakan tempat dimana aliran kerja dimulai. |
|  |  | *Activity* | Menggambarkan operasi atau tindakan yang dilakukan dalam alur kerja. |
|  |  | *Decision* | Menggambarkan percabangan dalam alur kerja dan merupakan titik dimana keputusan harus diambil dalam aliran kerja. |
|  |  | *Control Flow* | Merupakan Garis yang menghubungkan aktivitas-aktivitas dalam diagram. |
|  |  | *Fork* | Merupakan percabangan atau pembagian jalur untuk membagi aliran kontrol menjadi jalur-jalur paralel. |
|  |  | *Join* | Merupakan tempat dimana jalur yang terbagi dari *Fork* bergabung kembali. |
|  |  | *End Node* | Merupakan titik akhir dari aliran kerja. |

### *Sequence* Diagram

*Sequence* diagram adalah diagram UML paling umum kedua yang merepresentasikan bagaimana objek berinteraksi dan bertukar pesan dari waktu ke waktu (*over time*). *Sequence* diagram menggambarkan bagaimana peristiwa (aktivitas) dalam *use case* yang dipetakan ke dalam operasi kelas objek dalam *class* diagram. Penerimaan umum *sequence* diagram dapat dikaitkan dengan sifatnya yang relatif intuitif dan kemampuan untuk menggambarkan perilaku parsial (Wayahdi & Ruziq, 2023).

**Tabel 2. 3 Simbol Sequence Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
|  |  | *Actor* | Menggambarkan pengguna atau entitas yang berinteraksi dengan sistem. |
|  |  | *Boundary* | Merepresentasikan antarmuka atau batasan antara sistem dan aktor. |
|  |  | *Control* | Merupakan komponen yang menggambarkan logika pengendalian atau kontrol dalam sistem |
|  |  | *Entity* | Merupakan objek yang berisi data atau informasi yang terkait dalam basis data. |
|  |  | *Object Lifeline* | Menggambarkan durasi keberadaan suatu objek atau entitas selama berlangsungnya interaksi dalam sistem. |
|  |  | *Activation* | Menggambarkan kapan objek melakukan suatu tindakan tertentu selama interaksi. |
|  |  | *Message* | Merupakan cara objek untuk berkomunikasi dengan cara mengeirim pesan yang berisi informasi |
|  |  | *Return* | Menggambarkan kapan suatu objek memberikan respons atau hasil setelah menerima pesan. |
|  |  | *Callback* | Menggambarkan suatu sistem menjalankan tindakan khusus saat kondisi tertentu terpenuhi. |
|  |  | *Self call* | Merupakan gambaran saat objek melakukan suatu tindakan pada dirinya dengan memanggil metode nya sendiri. |

### *Class* Diagram

*Class* diagram merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam *model* desain dari suatu sistem, juga memperlihatkan aturan- aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Jadi dapat dikatakan bahwa *class* diagram adalah visual dari struktur sistem program pada jenis-jenis yang di bentuk. *Class* diagram merupakan alur jalannya sebuah *database* pada *system* yang akan dibangun atau dibuat. *Class* diagram juga disebut kumpulan dari beberapa *class* dan relasinya. *Class* identik dengan *entity* yang direpresentasikan dalam bentuk persegi di mana pada bagian atas ditulis nama *class*, kemudian ke bawah ditulis *attribute* yang terdapat pada *class*, kemudian ke bawah lagi ditulis metode yang ada pada *class*. Sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek (Ramdany dkk., 2024).

**Tabel 2. 4 Simbol Class Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Gambar | Nama | Keterangan |
|  |  | *Class* | Menggambarkan sebagai kotak dengan tiga bagian: bagian atas untuk nama *class*, bagian tengah untuk atribut/*property*, bagian bawah untuk metode. |
|  |  | *Association* | Menggambarkan untuk menunjukan hubungan antar *class*. |
|  |  | *Generalization* | Menggambarkan hubungan hierarki antara kelas, di mana *child class* mewarisi atribut dan metode dari *parent class*. |
|  |  | *Aggregation* | Menggambarkan hubungan "bagian-dari," di mana sebuah kelas terdiri atas objek-objek dari kelas lain. |
|  |  | *Compositon* | Sama seperti *aggregation* tetapi hubungannya jauh lebih kuat; objek bagian tidak dapat berdiri sendiri tanpa objek utama. |
|  |  | *Depedency* | Menggambarkan untuk menunjkan bahwa satu *class* bergantung pada *class* lain. |

## Aplikasi Pendukung

Aplikasi pendukung merupakan perangkat lunak yang digunakan untuk menunjang proses pembangunan sistem dalam penelitian ini. Aplikasi tersebut meliputi *platform* yang dimanfaatkan untuk analisis sistem, pengolahan data, hingga pengembangan *web*. Pemilihan aplikasi pendukung bertujuan untuk memastikan proses analisis, perancangan, implementasi, dan pengujian sistem dapat dilakukan secara lebih terstruktur, efisien, dan sesuai dengan kebutuhan penelitian.



### Visual Studio Code

Visual Studio Code adalah *software* yang sangat ringan, namun kuat *editor* kode sumbernya yang berjalan dari *desktop*. Visual Studio Code digunakan untuk pembuatan kode-kode program dibutuhkan sebuah aplikasi yang mumpuni (M. N. Syarif dkk., 2023).

### XAMPP

XAMPP adalah sebuah *software web server* Apache yang di dalamnya sudah tersedia *database server* MySQL dan dapat mendukung pemrograman PHP. XAMPP merupakan *software* yang mudah digunakan, gratis dan mendukung instalasi di Linux dan Windows. Keuntungan lainnya adalah cuma menginstal satu kali sudah tersedia Apache *web server*, MySQL *database server*, PHP *support* (PHP 4 dan PHP 5) dan beberapa modul lainnya (Surya Ningsih dkk., 2022).

### Draw.io

Draw.io adalah sebuah *website* yang didesain khusus untuk menggambarkan diagram UML secara *online*. Di mana punya tampilan yang sangat *responsive* dan terintergrasi dengan layanan penyimpanan *file* milik Google yaitu Google Drive sehingga draw.io menjadi alternatif dalam pembuatan diagram UML dengan waktu yang lebih singkat (Marthiawati dkk., 2024).

## Teori Pengujian Sistem

Pengujian sistem merupakan proses evaluasi untuk memastikan bahwa perangkat lunak yang telah dikembangkan dapat beroperasi sesuai dengan standar yang ditetapkan. Pengujian sistem dapat menggunakan metode *black box testing*. *Black box testing* efektif dalam mengidentifikasi kesalahan secara menyeluruh. Fokus *black box testing* terletak pada pengujian persyaratan fungsional perangkat lunak, dengan penekanan pada kondisi *input* yang memenuhi persyaratan fungsional. Oleh karena itu, pengujian sistem melibatkan pengecekan *input*, *output*, dan proses. Selain itu, pengujian juga dapat melibatkan penerapan skala *Likert*, yang digunakan untuk mengukur sikap, pendapat, dan persepsi individu atau kelompok terhadap fenomena sosial tertentu (Damayanti dkk., 2022).

### Sistem *Black Box Testing*

*Black box testing* adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja. Pengetahuan khusus dari kode aplikasi / struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. Menggunakan deskripsi eksternal perangkat lunak, termasuk spesifikasi, persyaratan, dan desain untuk menurunkan uji kasus (Pradana Putra dkk., 2020).

### *User Response* (Kuesioner)

Kuesioner adalah instrumen yang digunakan untuk mengukur suatu peristiwa atau kejadian yang berisi kumpulan pertanyaan untuk memperoleh informasi terkait penelitian yang dilakukan (Nur Amalia dkk., 2022).

# BAB III ANALISA DAN PERANCANGAN



## Analisa Sistem

Analisa sistem merupakan sebuah proses yang sangat penting dalam pengembangan dan implementasi suatu sistem, baik itu dalam konteks pengembangan perangkat lunak, perancangan organisasi, perbaikan proses bisnis, maupun dalam berbagai aplikasi lainnya. Dalam konteks implementasi *game* *Android* 3D simulasi dalam pembelajaran bahasa Jepang pada mahasiswa Universitas Pamulang dengan *Unity* *Engine* menggunakan metode *Multimedia* *Development* *Life* *Cycle* *(MDLC)*, analisis sistem akan menjadi kunci utama untuk memahami, mendokumentasikan, dan memeriksa seluruh proses yang terlibat dalam pembangunan dan penggunaan *game* tersebut.

Analisa ini mencakup pemahaman mendalam terhadap kebutuhan belajar bahasa Jepang dari perspektif mahasiswa. Ini mencakup penilaian terhadap *level* pemahaman bahasa, tujuan pembelajaran, serta area-area spesifik yang perlu ditingkatkan. Analisa sistem juga mencakup identifikasi fitur-fitur dan konten yang harus ada dalam *game* untuk mendukung pembelajaran bahasa Jepang dengan efektif. Ini meliputi kosakata, perhitungan, tata bahasa, keterampilan berbicara, serta fitur tambahan seperti ujian. Penilaian terhadap pengalaman mahasiswa juga merupakan bagian penting dari analisa ini. Aspek-aspek seperti antarmuka mahasiswa, navigasi, dan interaksi dalam konteks pembelajaran bahasa Jepang dianalisis secara menyeluruh.

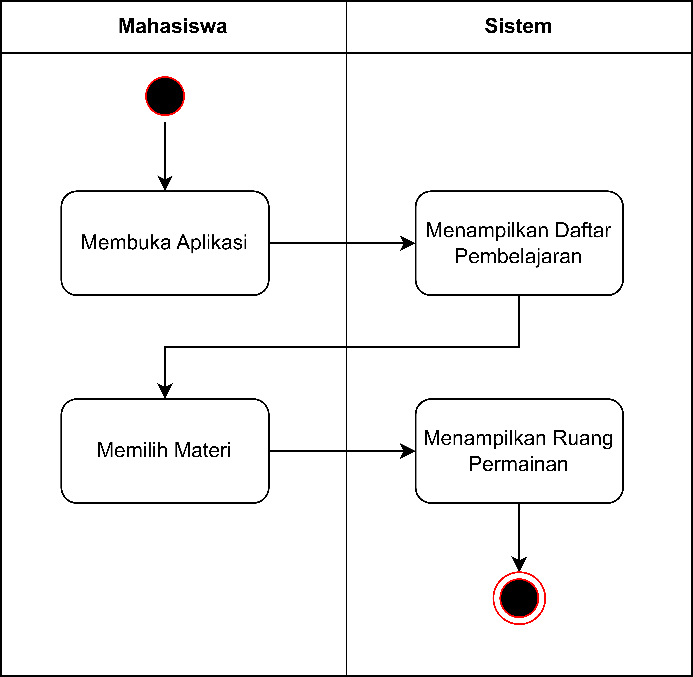
Selain itu, analisa sistem juga mencakup kesiapan dan persyaratan yang diperlukan untuk mempublikasikan *game* pada *platform* distribusi seperti *Play Store*. Ini termasuk pemahaman tentang persyaratan teknis dan kebijakan yang harus diperhatikan dalam proses distribusi. Dengan melakukan analisa sistem yang relevan, penulis dapat memahami secara menyeluruh kebutuhan dan *input* yang diperlukan untuk mengimplementasikan *game* *Android* 3D simulasi yang efektif dalam pembelajaran bahasa Jepang. Ini akan membantu dalam merancang, mengembangkan, dan mengimplementasikan *game* dengan memenuhi harapan mahasiswa dan tujuan pembelajaran yang diinginkan.

Berikut adalah referensi bahasa Jepang yang digunakan:

1. “*Minna no Nihongo 2nd Edition Shokyu 1*” – Buku ini menyediakan materi dasar bahasa Jepang yang mencakup kosakata, tata bahasa, dan latihan percakapan, ideal untuk pemula (3A Corporation, 2012).
2. “*Minna no Nihongo Shokyuu 1 Second Edition – Indonesian Edition*” –Versi bahasa Indonesia dari buku "*Minna no Nihongo*", yang memudahkan pemahaman bagi pelajar yang menggunakan bahasa Indonesia sebagai bahasa pengantar (Etsuko & Sachiko, 2013).
3. “Kiat Sukses Mudah & Praktis Mencapai N5 Edisi Baru Metode *Gakushudo*” – Buku ini dirancang khusus untuk membantu pelajar mempersiapkan diri menghadapi ujian JLPT N5 dengan metode yang mudah dan praktis (Shiang, 2018).

### Analisa Sistem Berjalan

Analisis sistem yang sedang berjalan merupakan proses penguraian dari suatu sistem dengan tujuan mengidentifikasi dan mengevaluasi masalah yang terjadi pada suatu sistem. *Activity* *diagram* yang menggambarkan sistem tersebut dapat dijelaskan pada gambar di bawah ini:



Gambar 3. 1 activity diagram sistem berjalan

Dalam konteks aplikasi *game* pembelajaran bahasa yang tersedia di *platform* *Play Store*, seperti (Cari Kata Jepang dan Tebak Kata Jepang) menerapkan sistem yang terlihat pada gambar di atas dapat dijelaskan sebagai berikut: Setelah mahasiswa membuka aplikasi, sistem akan menampilkan daftar materi pembelajaran yang tersedia. Mahasiswa selanjutnya memilih salah satu dari daftar pembelajaran, dan sistem akan mengarahkannya ke ruang permainan yang dirancang untuk belajar sambil bermain. Umumnya, permainan terdiri dari beberapa tingkat, dimulai dari tingkat kesulitan terendah hingga tertinggi, dengan tingkat kesulitan yang meningkat di setiap tingkatan permainan.

### Analisa Sistem Usulan

Dalam merancang sistem *game* simulasi 3D pembelajaran bahasa Jepang yang diusulkan sebagai pengembangan dari sistem berjalan sebelumnya, penelitian ini bertujuan untuk memberikan solusi terhadap berbagai kebutuhan yang perlu dipenuhi dalam sistem yang akan dikembangkan, didasarkan pada hasil analisis yang telah dilakukan.

Analisa sistem yang cermat telah disusun untuk menarik minat mahasiswa, dengan menggunakan *Unity Engine* dan bahasa pemrograman *C#* sebagai fondasi pengembangan. Sistem metode pengembangan yang digunakan adalah *Multimedia Development Life Cycle (MDLC)* untuk mengembangkan *game*. Melalui analisa yang mendalam, penelitian ini mengusulkan implementasi sistem *game* simulasi 3D yang mempermudah mahasiswa dalam mempelajari pengetahuan bahasa Jepang.

Pada gambar 3.2 menunjukkan bahwa mahasiswa memulai dengan membuka aplikasi. Setelah itu, sistem akan menampilkan *scene* *Main Menu*. Mahasiswa kemudian menekan tombol *Start*, selanjutnya sistem akan memberikan kondisi. Jika mahasiswa sudah memilih karakter, maka akan diarahkan ke *scene Gameplay*. Jika belum, maka akan diarahkan ke *scene* *Character Selection*. Setelah itu, mahasiswa akan diberikan kondisi untuk memilih jenis kelamin karakter, yaitu laki-laki atau perempuan, dan kemudian memasukkan nama karakter. Setelah langkah-langkah tersebut, aplikasi akan membuka *scene Gameplay*, di mana mahasiswa dapat mengerjakan misi untuk menyelesaikan permainan.

Berikut adalah *activity diagram* dari analisa sistem yang diajukan:

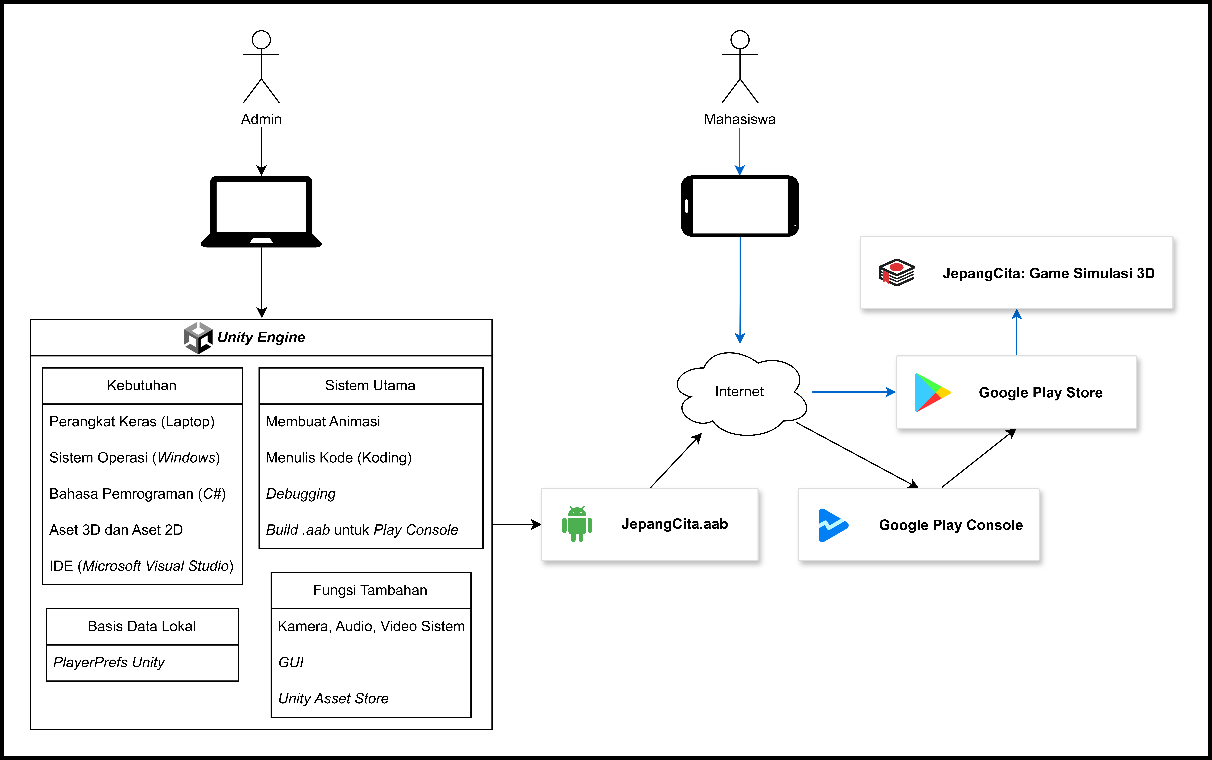


Gambar 3. 2 activity diagram sistem usulan

## Arsitektur Perangkat Lunak

Arsitektur perangkat lunak merupakan struktur-struktur yang menjadikan landasan untuk menentukan keberadaan komponen-komponen perangkat lunak, cara komponen-komponen untuk saling berinteraksi dan organisasi komponen-komponen dalam membentuk perangkat lunak.

Berikut adalah gambaran dari arsitektur perangkat lunak yang diusulkan:



Gambar 3. 3 arsitektur perangkat lunak

Pada gambar 3.3 *admin* berinteraksi dengan *laptop* untuk membangun *game* menggunakan *Unity Engine*. Setelah proses pembangunan *game* selesai, *admin* mengunggah *file* “JepangCita.aab” ke *Google Play Console* melalui internet. Setelah itu, *game* tersebut tersedia di *Google Play Store*. Mahasiswa kemudian mengakses *Google Play Store* melalui perangkat *Android* mereka yang terhubung internet untuk mengunduh *game* JepangCita.

## Metode Pengembangan

Metode pengembangan yang digunakan pada penelitian ini untuk implementasi *game* simulasi 3D pembelajaran bahasa Jepang adalah metode *Multimedia Development Life Cycle (MDLC)*. Metode pengembangan ini memiliki enam tahapan, yaitu: konsep, perancangan, pengumpulan bahan, perakitan, pengujian aplikasi, dan distribusi.



### Konsep (*Concept*)

Pada tahap konsep ini dijelaskan tujuan *game* dan identifikasi mahasiswa. Tujuan *game* yang dibuat adalah untuk memperkenalkan bahasa Jepang, sehingga mahasiswa dapat memiliki pengetahuan lebih mengenai bahasa Jepang dan *game* ini dapat dimainkan di *Android*. Selanjutnya, identifikasi mahasiswa *game* pada penelitian ini adalah mahasiswa dengan rentang umur 13 tahun ke atas.

Berikut ini adalah tabel yang berisi penjelasan mengenai permainan yang akan dibuat:

Tabel 3. 1 tabel deskripsi *game*

|  |  |
| --- | --- |
| **Keterangan** | **Deskripsi** |
| Judul *Game* | JepangCita: Game Simulasi 3D |
| *Target Audiens* | Mahasiswa umur 13 tahun ke atas |
| *Genre* | Simulasi 3D |
| *Audio* | *Background Music (BGM)* dan *Sound Effect (SFX)* |
| Interaktif | Menggunakan *laptop* untuk membaca materi, menggunakan kasur untuk tidur, menggunakan pintu keluar untuk pergi ke tempat kursus, mengikuti jadwal kelas, dan mengerjakan misi |

### Perancangan (*Design*)

Pada tahap *design* ini terdapat penjelasan untuk *storyboard* dan struktur navigasi dalam *game*. *Storyboard* berisi tampilan *Graphical User Interface (GUI)*, sementara struktur navigasi menjelaskan tombol-tombol yang digunakan dalam *game*.

Berikut ini adalah uraian lengkap dari *storyboard* yang tersedia untuk permainan, termasuk penjelasan tentang setiap tampilan GUI dan fungsi masing-masing tombol dalam struktur navigasi:

1. *Storyboard Main Menu*



Gambar 3. 4 storyboard main menu

Pada *storyboard main menu*, mahasiswa dapat melakukan beberapa struktur navigasi, seperti tombol mulai untuk memulai permainan, tombol pengaturan untuk melakukan pengaturan permainan, tombol kredit untuk menampilkan data diri *developer* dan aset-aset yang digunakan pada *game*, dan tombol keluar untuk keluar dari permainan.

1. *Storyboard Character Selection*



Gambar 3. 5 storyboard character selection

Pada *storyboard character selection*, mahasiswa dapat melakukan navigasi melalui beberapa struktur, seperti tombol pilih karakter untuk memilih karakter yang ingin digunakan, tombol panah kanan/kiri untuk mengganti karakter yang ditampilkan di layar, dan tombol *continue* untuk melanjutkan permainan ke *storyboard gameplay* berikutnya. Struktur navigasi ini dirancang untuk memberikan alur yang intuitif dan memudahkan pengguna dalam memilih dan melanjutkan permainan.

1. *Storyboard Gameplay*



Gambar 3. 6 storyboard gameplay

Pada *storyboard gameplay*, mahasiswa dapat melakukan beberapa aktivitas struktur navigasi, seperti tombol *joystick* untuk menggerakkan karakter, tombol aksi untuk melakukan aksi sesuai dengan objeknya (tombol aksi akan muncul jika karakter dekat dengan objek aksi), tombol misi untuk menampilkan misi yang harus diselesaikan, tombol *pause* untuk menghentikan permainan dan menampilkan tombol-tombol, seperti tombol melanjutkan permainan untuk melanjutkan permainan, tombol pengaturan untuk melakukan pengaturan dan tombol *main menu* untuk kembali ke *main menu*.

### Pengumpulan Bahan (*Material Collecting*)

Proses pengumpulan data dengan studi literatur yang digunakan sebagai referensi dalam proses penelitian untuk *game* pembelajaran bahasa Jepang. Studi literatur berupa buku-buku, jurnal yang terkait dan *website* yang relevan. Selanjutnya, untuk aset-aset pada *game* seperti objek 3 dimensi, karakter, *audio* dan lain-lain didapat melalui *website* yang bernama *Unity Asset Store*. *Unity Asset Store* adalah kumpulan *material* untuk membuat *game* sedangkan untuk pembuatan animasi karakter 3 dimensi melalui *website* bernama *Adobe Mixamo*.

### Perakitan (*Assembly*)

Berdasarkan dari tahap *design* yang telah dibuat, setelah tahap *material collecting* dilanjutkan ke tahap perakitan. Pada tahap perakitan *game*, dimulai dengan membuat *main menu* dengan aset 2D yang telah dibuat sebelumnya menggunakan *Figma*. Setelah itu, membuat mekanisme *game* seperti mekanisme misi, menampilkan materi, menggerakkan karakter, menghitung skor ujian, mengatur jadwal kelas, dan lain-lain. *Game* ini dibuat menggunakan *Unity*, *design UI/UX* menggunakan *Figma* dan *Adobe Photoshop* serta animasi karakter 3D menggunakan *Mixamo*.



Gambar 3. 7 proses pembuatan animasi 3D

Pada gambar 3.7 adalah proses pembuatan animasi 3D yang dimulai dengan mengunggah karakter 3D yang didapat dari *Aset Store Unity*. Pada proses ini dilakukan untuk membuat animasi yang dibutuhkan seperti animasi *idle*, *walk*, *sit*, dan lain-lain.



Gambar 3. 8 proses pembuatan main menu

Pada gambar 3.8 merupakan proses pembuatan *game* yang dimulai dari membuat *scene main menu*. Pada proses ini, langkah-langkah yang dilakukan seperti memasang *background*, membuat tombol, dan membuat mekanisme menggunakan bahasa pemrograman *C# script*.

### Pengujian Aplikasi (*Testing*)

Pada tahap ini uji coba dilakukan untuk mengetahui apakah aplikasi sudah berjalan dengan baik atau tidak. Jika sudah tidak ada galat pada aplikasi, maka dilakukan pengujian oleh mahasiswa.



Gambar 3. 9 scene main menu



Gambar 3. 10 scene character selection

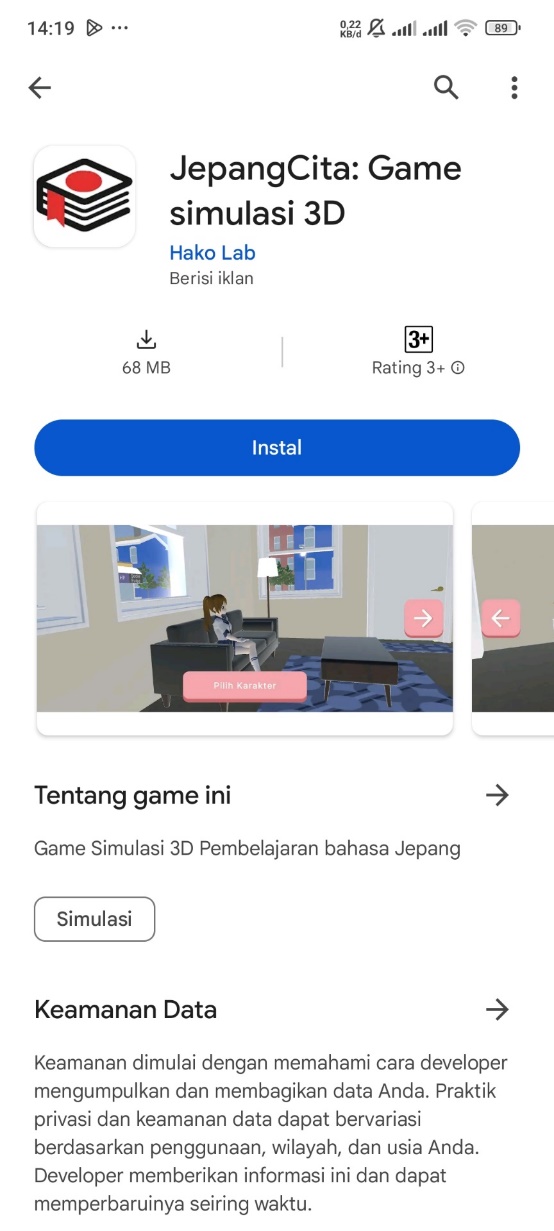


Gambar 3. 11 scene gameplay

### Distribusi (*Distribution*)

Berdasarkan dari aplikasi *game* yang telah dibuat, dilanjutkan pada tahap distribusi. Distribusi dapat dilakukan setelah aplikasi *game* dinyatakan telah layak pakai. Aplikasi ini didistribusikan atau dipublikasikan melalui *Google* *Play Store*. Berikut adalah *link* untuk mengunduh *game*:

<https://play.google.com/store/apps/details?id=com.HakoLab.JepangCita>



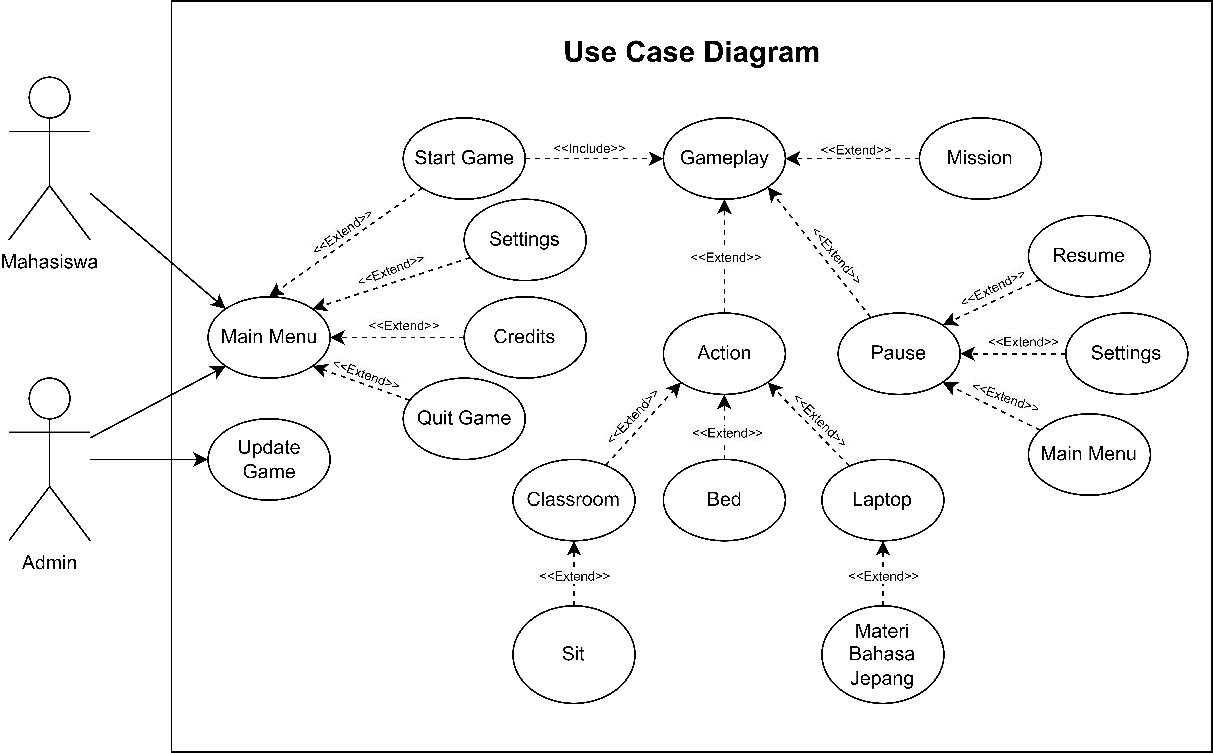
Gambar 3. 12 aplikasi tersedia di Google Play Store

## Perancangan *Unified Modeling Language (UML)*

*Unified Modeling Language (UML)* merupakan suatu standar bahasa yang umum digunakan dalam industri untuk mengidentifikasi kebutuhan, melakukan analisis dan perancangan, serta mengilustrasikan arsitektur dalam konteks pemrograman berbasis objek (Wira dkk., 2019).

### *Use Case Diagram*

*Use case diagram* adalah penjelasan tentang fungsi-fungsi suatu sistem dari sudut pandang para mahasiswa sistem. *Use case* memuat definisi tentang bagaimana sistem dan komponennya akan diproses. *Use case* beroperasi dengan menggunakan skenario, yang merupakan urutan atau langkah-langkah yang menjelaskan aktivitas yang dilakukan oleh mahasiswa terhadap sistem atau sebaliknya (Setiyani, 2021).



Gambar 3. 13 use case diagram JepangCita

Pada gambar 3.13, mahasiswa merupakan pemain dan *admin* merupakan *developer* yang memiliki kepentingan sebagai berikut:

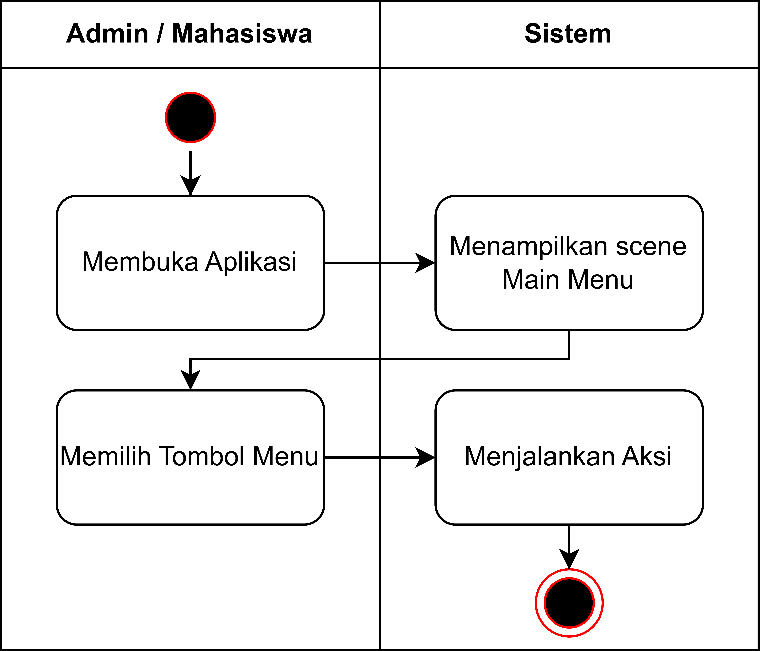
1. *Main menu*, pada saat *admin* atau mahasiswa membuka *game* JepangCita, maka akan langsung di arahkan ke *scene main menu* dan dapat memilih beberapa *menu*.
2. *Start game*, pada *menu* ini *admin* atau mahasiswa akan mulai permainan dan diarahkan ke *scene gameplay*.
3. *Settings*, pada *menu* ini *admin* atau mahasiswa akan ditampilkan *panel settings*, pada *panel* ini dapat melakukan konfigurasi untuk mengatur *volume* musik, *volume* efek suara, sensitivitas kamera, dan jarak kamera.
4. *Credits*, pada *menu* ini *admin* atau mahasiswa akan ditampilkan *panel credits*, pada *panel* ini dapat melihat biodata dari *developer* dan aset-aset yang digunakan pada *game*.
5. *Quit game*, pada *menu* ini *admin* atau mahasiswa akan ditampilkan *panel quit game*, pada *panel* ini akan ditampilkan dua buah kondisi, di mana jika menekan tombol *quit game* maka akan menutup aplikasi *game* JepangCita dan jika menekan tombol *cancel* maka akan menutup *panel quit game*.
6. *Gameplay*, *scene gameplay* ini dapat diakses setelah *admin* atau mahasiswa menekan tombol *start game* pada *main menu*. Pada *scene* ini dapat membuka *menu* *mission*, *pause* atau dapat melakukan *action*.
7. *Mission*, pada *menu* ini *admin* atau mahasiswa akan ditampilkan *panel mission*, pada *panel* ini akan ditampilkan detail dari misi yang harus dijalankan.
8. *Pause*, pada *menu* ini *admin* atau mahasiswa akan ditampilkan *panel pause* dan menghentikan permainan, pada *panel* ini akan ditampilkan beberapa *menu*, yaitu: *menu* *resume*, *settings* dan *main menu*.
9. *Pause – resume*,pada *menu* ini *admin* atau mahasiswa akan menutup *panel pause* dan melanjutkan permainan.
10. *Pause – settings*, pada menu ini *admin* atau mahasiswa akan ditampilkan *panel settings*, pada *panel* ini dapat melakukan konfigurasi untuk mengatur *volume* musik, *volume* efek suara, sensitivitas kamera, dan jarak kamera.
11. *Pause – main menu*, pada *menu* ini *admin* atau mahasiswa akan ditampilkan *panel main menu*, pada *panel* ini akan ditampilkan dua buah kondisi, di mana jika menekan tombol *main menu* maka akan diarahkan ke *scene main menu* dan jika menekan tombol *cancel* maka akan menutup *panel main menu*.
12. *Gameplay – action*, *admin* atau mahasiswa dapat berinteraksi dengan berbagai objek dalam *game* seperti pergi ke *classroom*, *bed*, atau *laptop*.
13. *Action – classroom*, pada aksi ini *admin* atau mahasiswa akan diarahkan ke *scene classroom.*
14. *Action – bed*, pada aksi ini *admin* atau mahasiswa dapat melewatkan satu hari dalam *game*.
15. *Action – laptop*, pada aksi ini *admin* atau mahasiswa akan ditampilkan *desktop* seperti pada komputer sungguhan. Pada *desktop* terdapat beberapa *menu* seperti *mission*, *calendar*, *mail*, *browser*, *music* dan *shutdown*.
16. *Classroom – sit*, pada aksi ini *admin* atau mahasiswa harus berada pada *scene classroom* dan jika sesuai dengan jadwal kelas, aksi *sit* digunakan untuk memulai pelajaran.
17. *Action laptop* – materi bahasa Jepang, untuk mengakses ini *admin* atau mahasiswa harus berada di *desktop laptop* kemudian, harus membuka *menu* *browser* dan mengetikan “JepangCita” pada *search bar*. Setelah itu, akan ditampilkan *menu* *register* dan *login*. Jika belum pernah mendaftar akun JepangCita, maka harus memilih menu *register* dan mengisi *form*. Jika sudah pernah mendaftar, maka harus memilih menu *login* dan melakukan *login*. Jika sudah *login*, buka *menu* materi untuk mengakses materi bahasa Jepang.
18. *Update game*, *admin* dapat melakukan pembaruan pada *game* yang memungkinkan untuk menambahkan fitur-fitur baru, memperbaiki *bug*, meningkatkan performa, serta memperluas konten.

### *Activity Diagram*

*Activity diagram* adalah representasi grafis dari konsep aliran data/kontrol dan aksi terstruktur yang dirancang secara baik dalam suatu sistem. *Diagram* ini membantu dalam memvisualisasikan proses-proses yang terjadi dalam sistem dengan jelas dan detail (Arianti, Fa’izi, Adam, Wulandari, dkk., 2022).

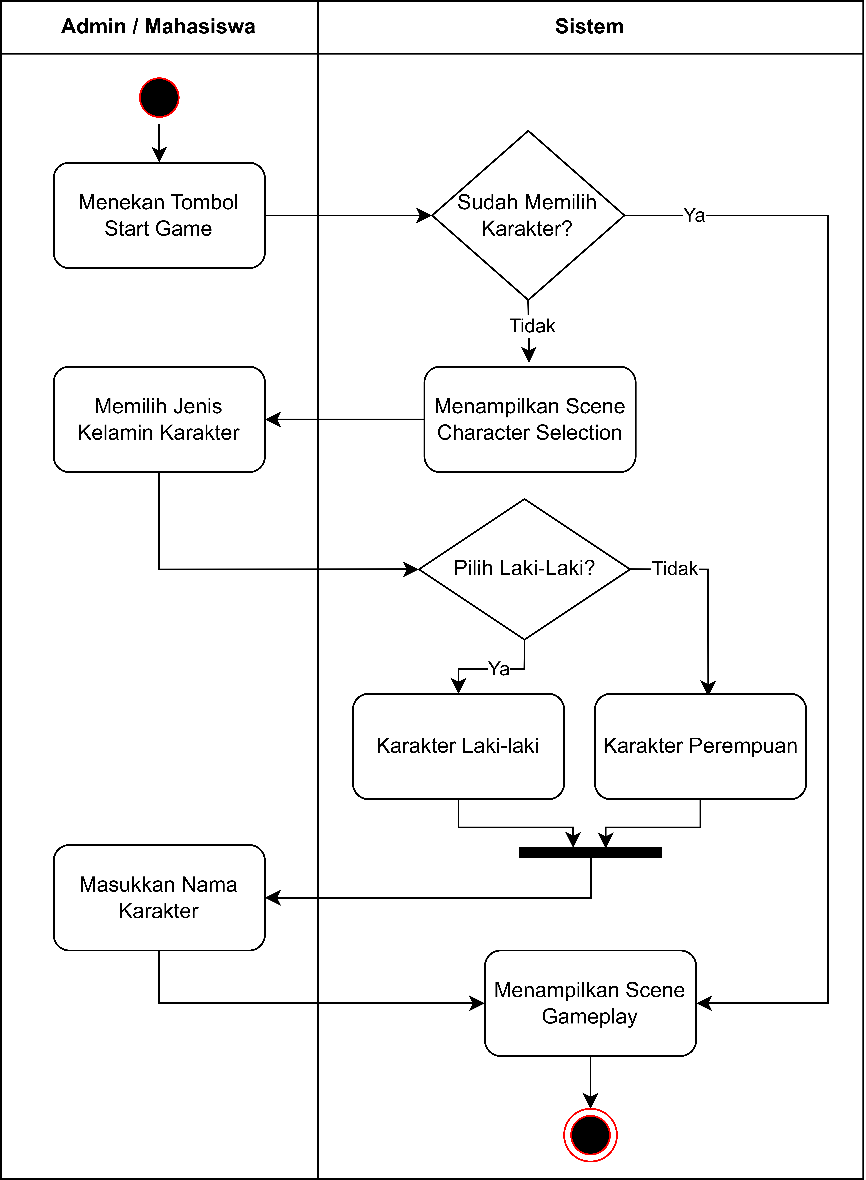
1. *Activity diagram main menu*

Pada gambar 3.14, setelah *admin* atau mahasiswa membuka aplikasi maka, akan ditampilkan *scene* *main menu*. *Scene* ini berfungsi sebagai gerbang utama untuk mengakses berbagai fitur dan fungsi dalam aplikasi. Pada *scene* ini dapat memilih beberapa *menu* seperti *start game*, *settings*, *credits*, dan *quit game*.



Gambar 3. 14 activity diagram main menu

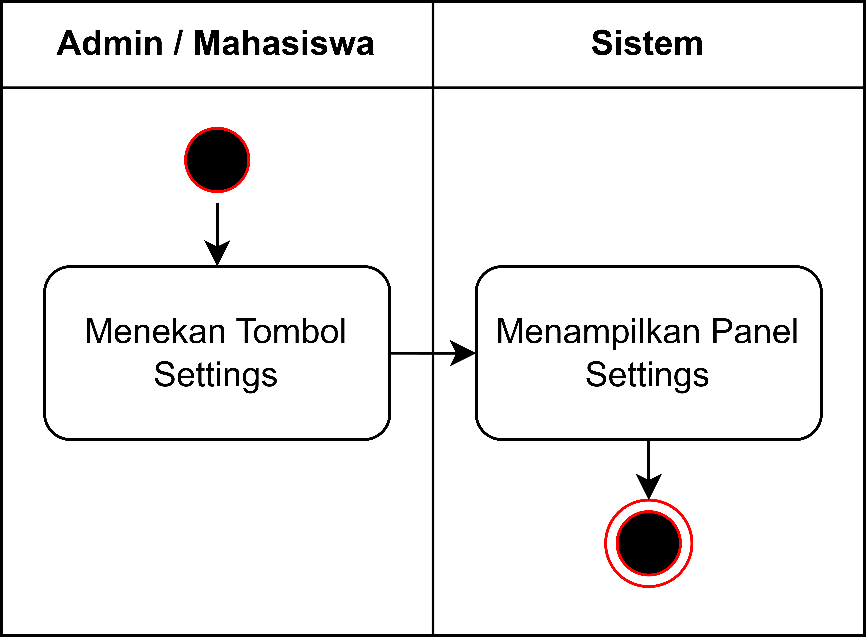
1. *Activity diagram main menu – start game*



Gambar 3. 15 activity diagram main menu – start game

Pada gambar 3.15, setelah *admin* atau mahasiswa menekan *tombol start game* pada *scene* *main menu*. Sistem akan mendeteksi apakah sudah memilih karakter atau belum. Jika sudah maka, akan langsung diarahkan ke *scene gameplay*. Jika belum maka, akan diarahkan ke *scene character selection* lalu, akan diminta untuk memilih salah satu karakter. Setelah memilih karakter, akan diminta juga untuk mengisi nama karakternya. Setelah itu, akan diarahkan ke *scene gameplay*.

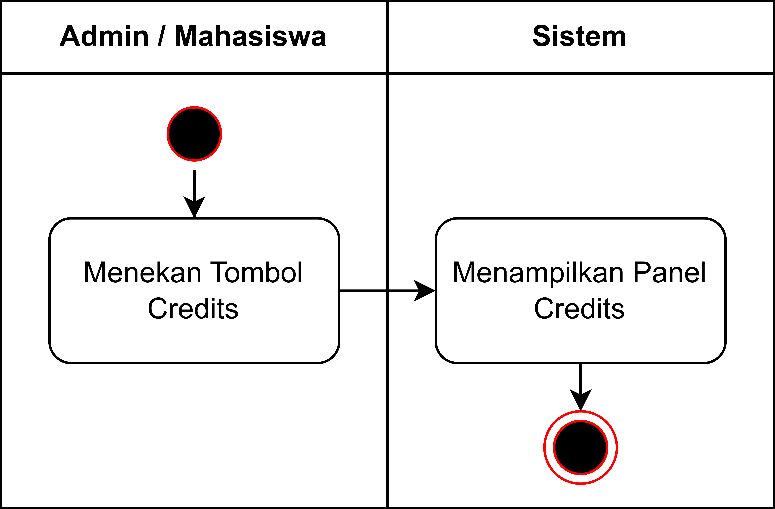
1. *Activity diagram main menu – settings*



Gambar 3. 16 activity diagram main menu – settings

Pada gambar 3.16, setelah *admin* atau mahasiswa menekan *tombol settings* pada *scene* *main menu* maka, akan ditampilkan *panel settings*. Pada *panel* ini dapat mengatur *volume* musik, *volume* efek suara, sensitivitas kamera, dan jarak kamera.

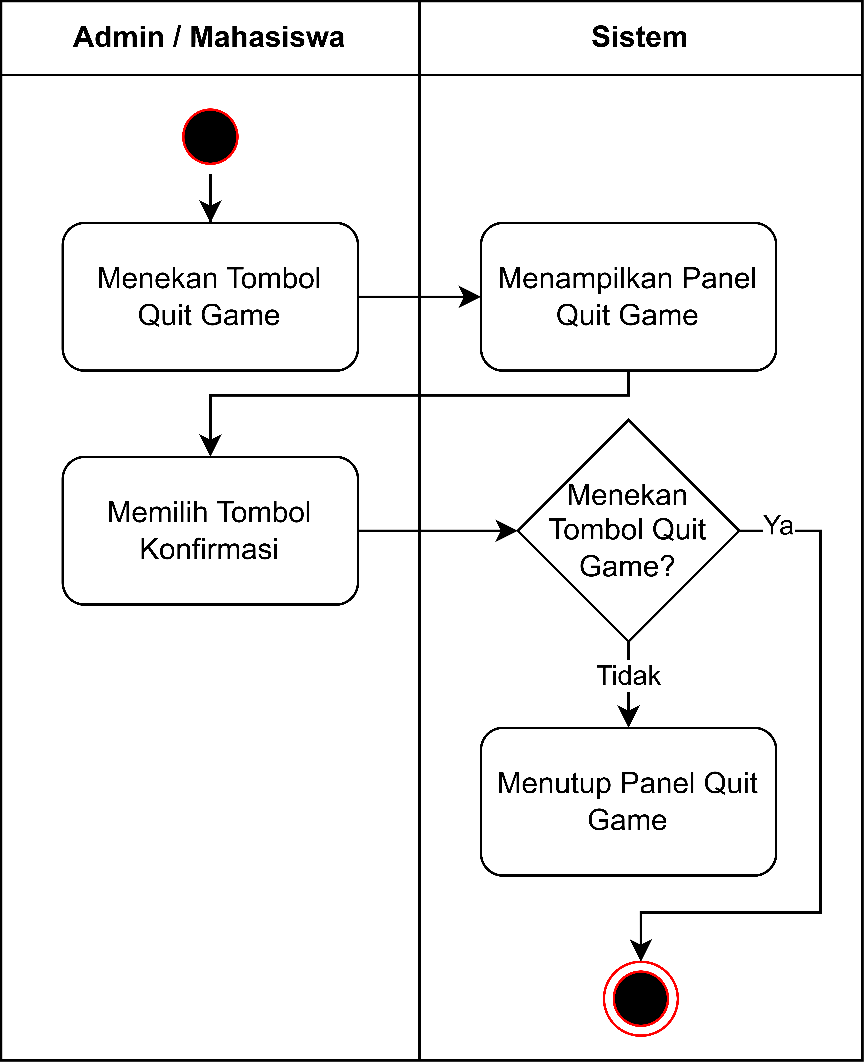
1. *Activity diagram main menu – credits*



Gambar 3. 17 activity diagram main menu – credits

Pada gambar 3.17, setelah *admin* atau mahasiswa menekan *tombol credits* pada *scene* *main menu* maka, akan ditampilkan *panel credits*. Pada *panel* ini akan ditampilkan biodata dari *developer* dan aset-aset yang digunakan pada *game*.

1. *Activity diagram main menu – quit game*

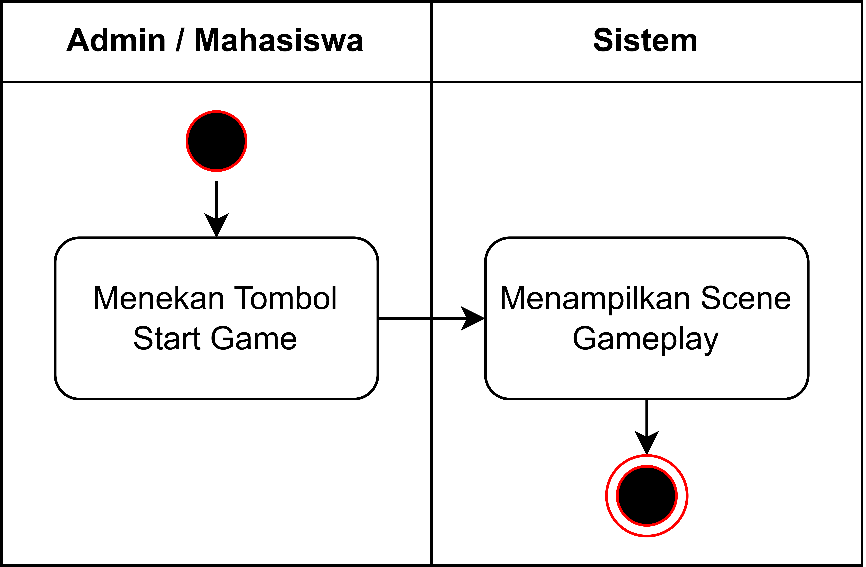


Gambar 3. 18 activity diagram main menu – quit game

Pada gambar 3.18, setelah *admin* atau mahasiswa menekan *tombol quit game* pada *scene* *main menu* maka, akan ditampilkan *panel quit game*, pada *panel* ini akan diminta untuk memilih tombol *quit game* atau tombol *cancel*. Jika menekan tombol *quit game* maka, akan menutup aplikasi *game*. Jika menekan tombol *cancel* maka, akan menutup *panel quit game*.

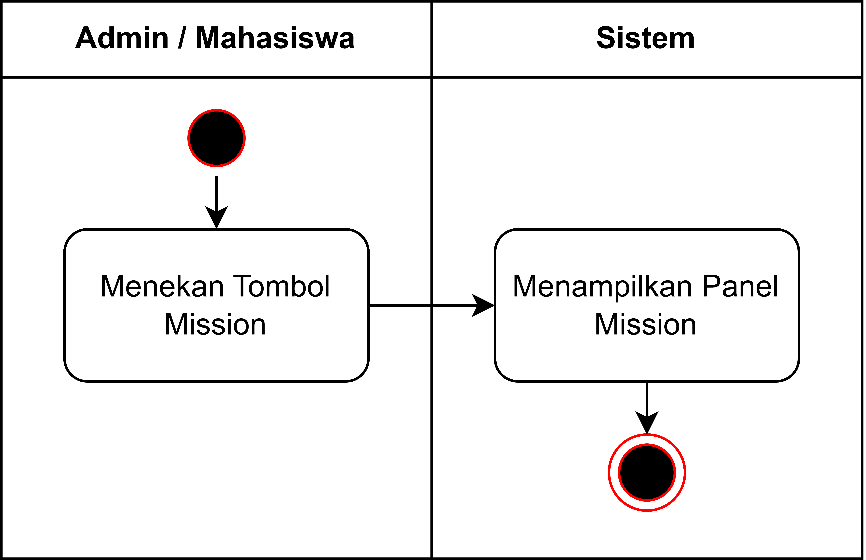
1. *Activity diagram gameplay*

Pada gambar 3.19, setelah *admin* atau mahasiswa menekan *tombol start game* pada *scene* *main menu* maka, akan diarahkan ke *scene gameplay*.



Gambar 3. 19 activity diagram gameplay

1. *Activity diagram gameplay – mission*

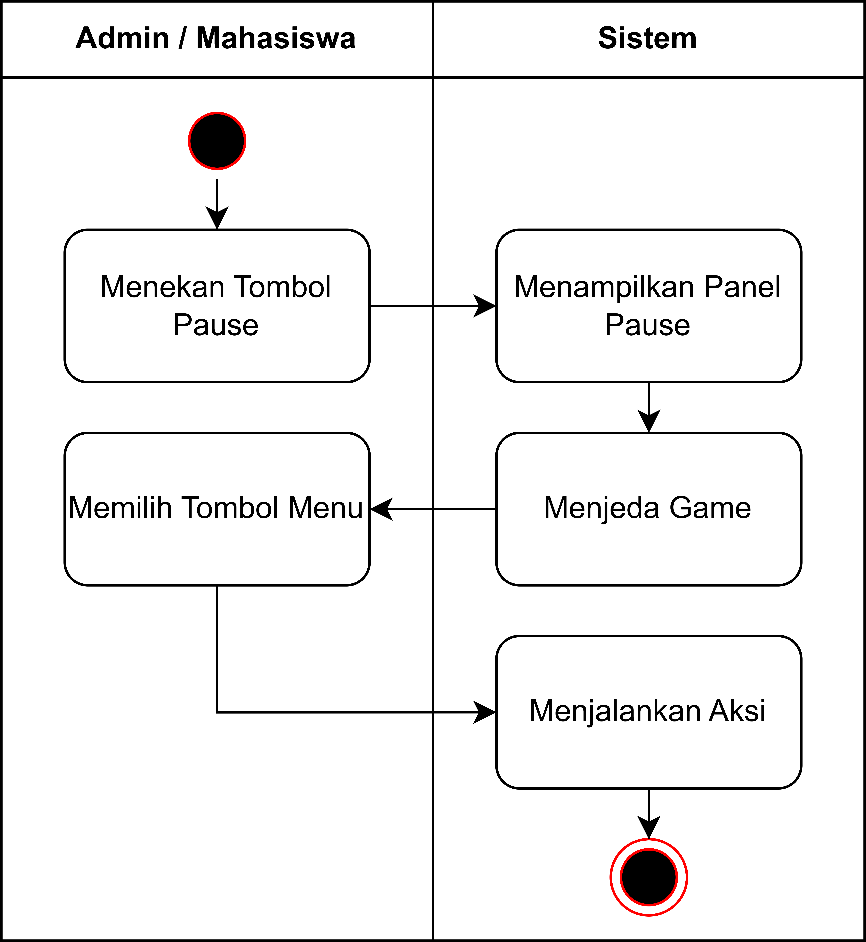


Gambar 3. 20 activity diagram gameplay – mission

Pada gambar 3.20, setelah *admin* atau mahasiswa menekan *tombol mission* pada *scene* *gameplay* maka, akan ditampilkan *panel* *mission*. Pada *panel* ini akan diberikan detail misi yang harus dilakukan untuk menyelesaikan permainan.

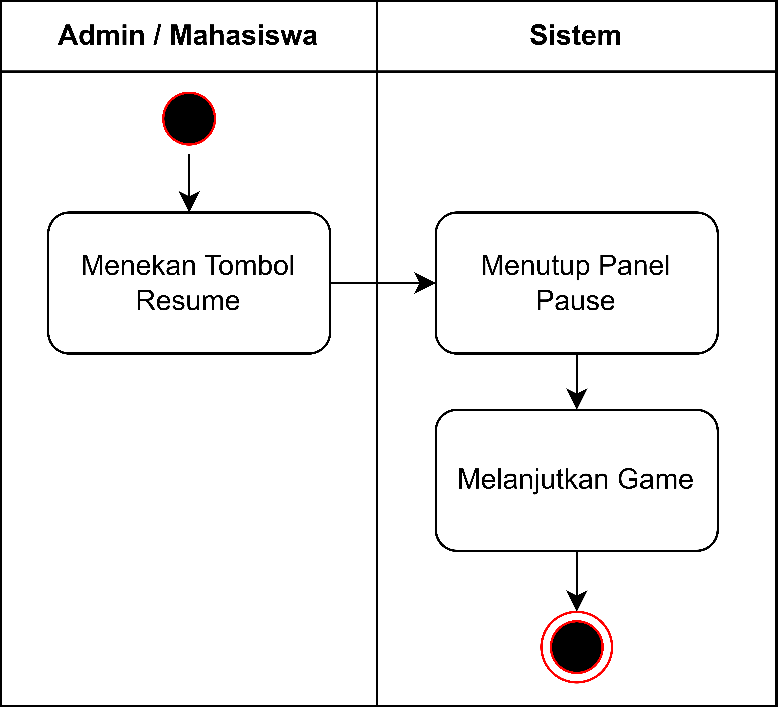
1. *Activity diagram gameplay – pause*

Pada gambar 3.21, setelah *admin* atau mahasiswa menekan *tombol pause* pada *scene* *gameplay* maka, akan ditampilkan *panel* *pause* dan menghentikan permainan. Pada *panel* ini akan ditampilkan beberapa pilihan tombol menu, yaitu: tombol *resume*, tombol *settings*, dan tombol *main menu*.



Gambar 3. 21 activity diagram gameplay – pause

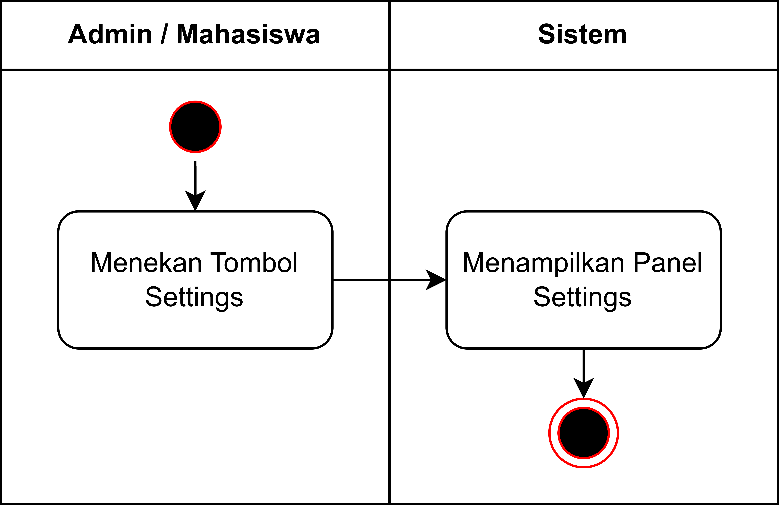
1. *Activity diagram pause – resume*



Gambar 3. 22 activity diagram pause – resume

Pada gambar 3.22, setelah *admin* atau mahasiswa menekan *tombol resume* pada *panel pause* maka, akan menutup *panel* *pause* dan melanjutkan permainan.

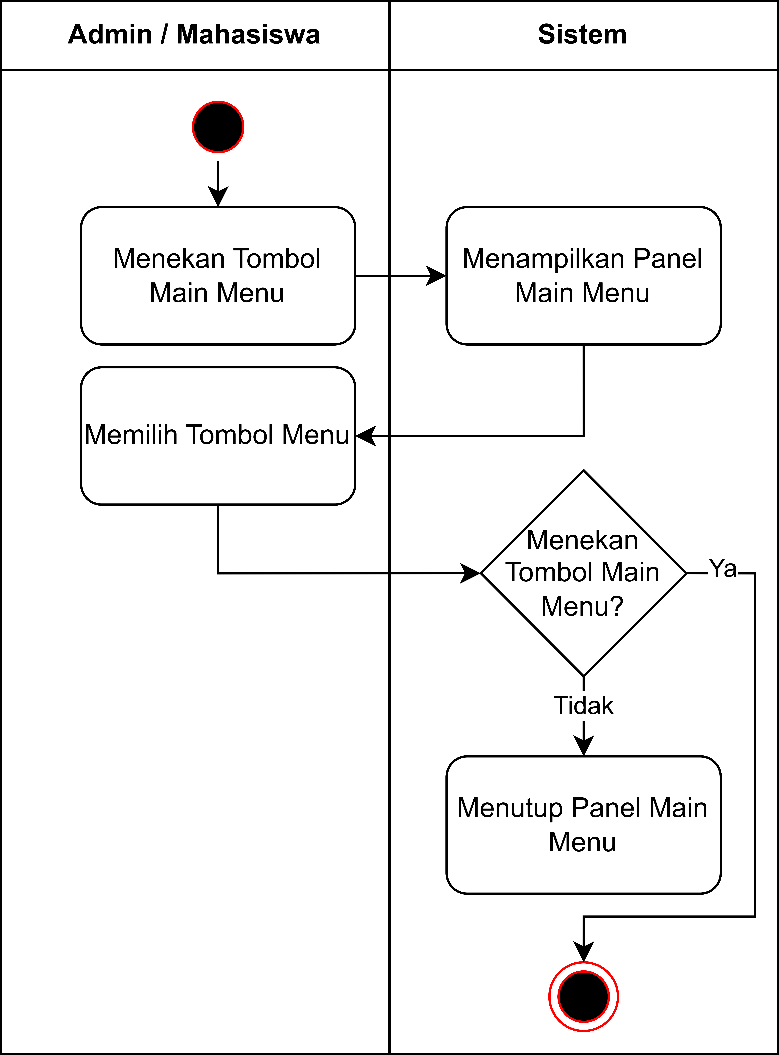
1. *Activity diagram pause – settings*



Gambar 3. 23 activity diagram pause – settings

Pada gambar 3.23, setelah *admin* atau mahasiswa menekan *tombol settings* pada *panel pause* maka, akan ditampilkan *panel* *settings*. Pada *panel* ini dapat melakukan konfigurasi untuk mengatur *volume* musik, *volume* efek suara, sensitivitas kamera, dan jarak kamera.

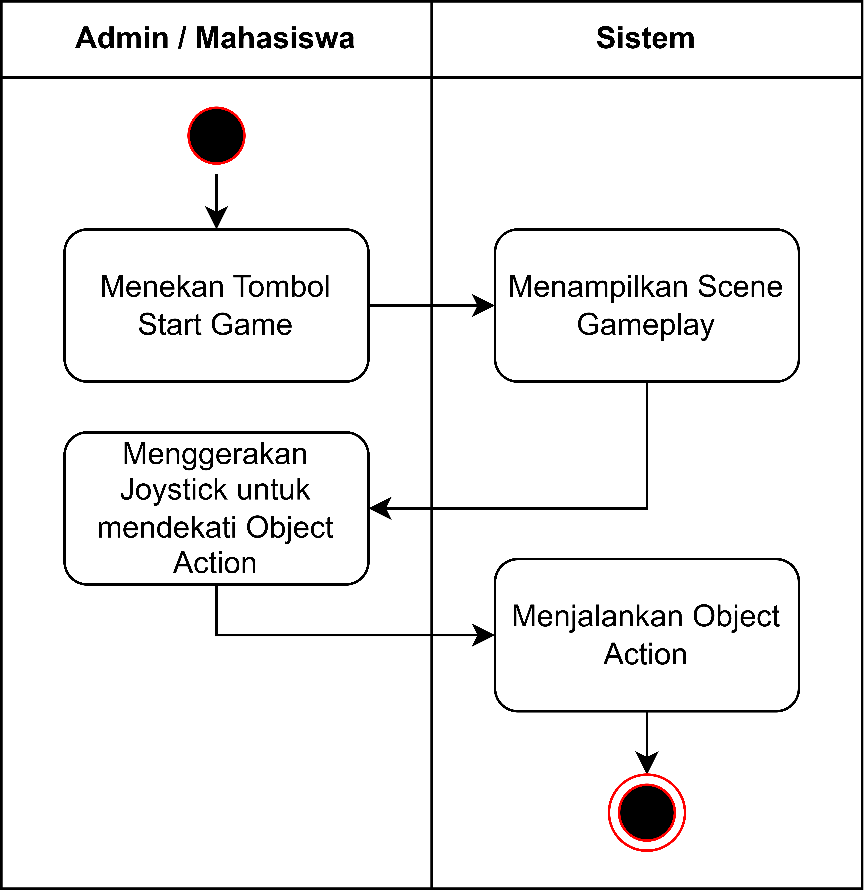
1. *Activity diagram pause – main menu*



Gambar 3. 24 activity diagram pause – main menu

Pada gambar 3.24, setelah *admin* atau mahasiswa menekan *tombol main menu* pada *panel pause* maka, akan ditampilkan *panel* *main menu*. Pada *panel* ini akan diminta untuk memilih tombol *main menu* atau tombol *cancel*. Jika menekan tombol *main menu* maka, akan diarahkan ke *scene* *main menu*. Jika menekan tombol *cancel* maka, akan menutup *panel main menu*.

1. *Activity diagram gameplay – action*

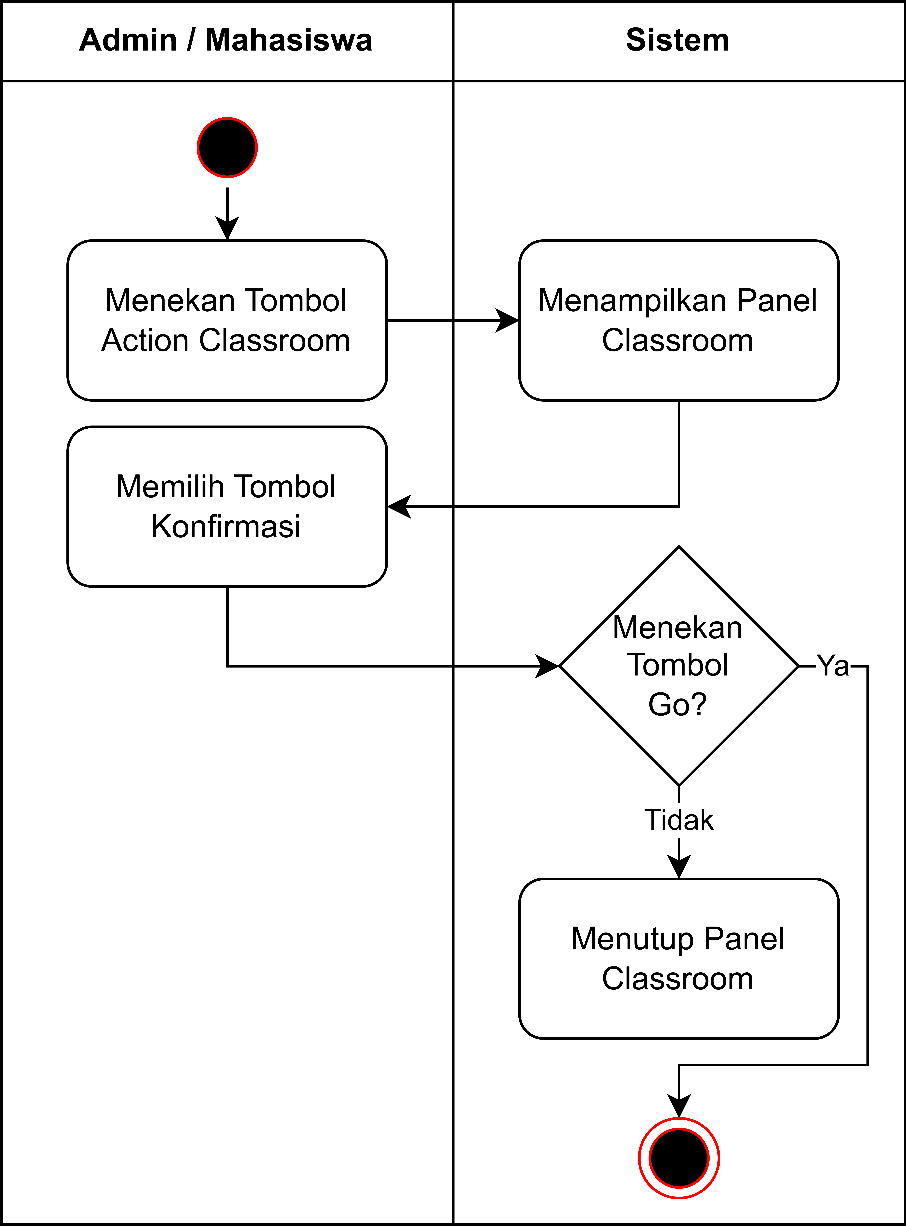


Gambar 3. 25 activity diagram gameplay – action

Pada gambar 3.25, setelah *admin* atau mahasiswa berada pada *scene gameplay* maka, pada *scene* ini dapat melakukan aksi dengan mendekat ke objek aksi kemudian, menekan tombol *action*.

1. *Activity diagram action – classroom*

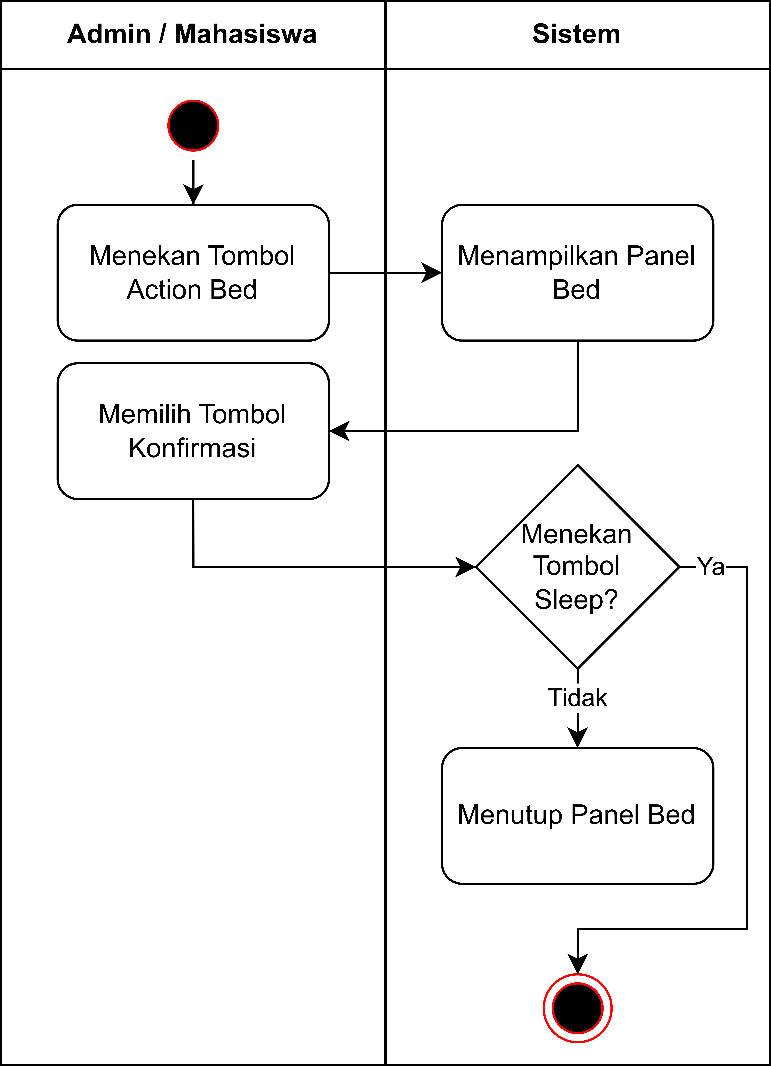
Pada gambar 3.26, setelah *admin* atau mahasiswa membuka objek pintu keluar dan keluar rumah maka, akan ditampilkan *panel* *confirm* *classroom*. Pada *panel* ini akan diminta untuk memilih tombol *go* atau tombol *cancel*. Jika menekan tombol *go* maka, akan diarahkan ke *scene classroom*. Jika menekan tombol *cancel* maka, akan menutup *panel confirm classroom* dan karakter akan kembali ke dalam rumah*.*



Gambar 3. 26 activity diagram action – classroom

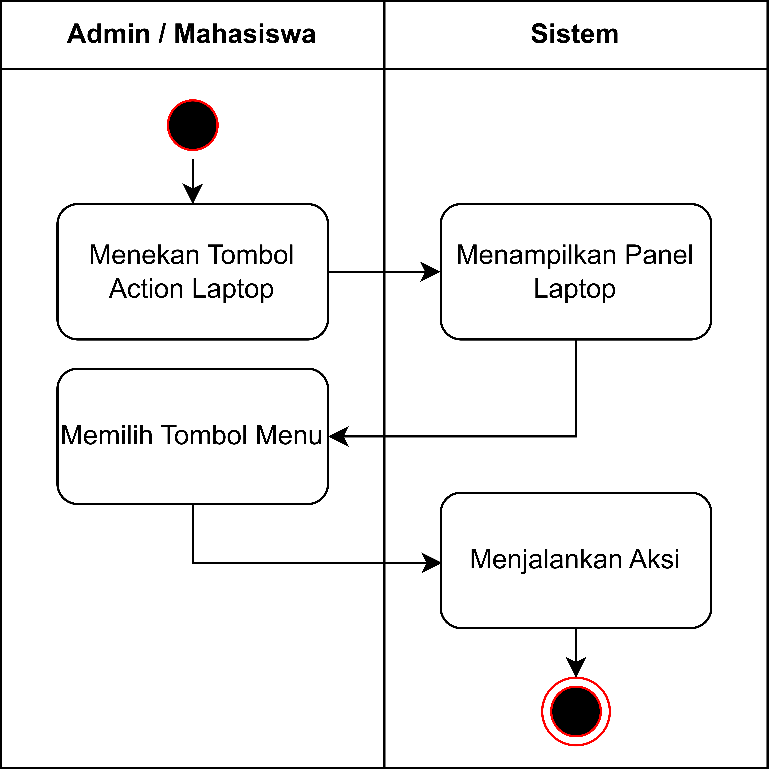
1. *Activity diagram action – bed*

Pada gambar 3.27, setelah *admin* atau mahasiswa menggerakkan karakter ke objek *bed* maka, akan ditampilkan tombol *action* *bed*. Setelah menekan tombol tersebut maka, akan ditampilkan *panel* *bed*. Pada *panel* ini akan diminta konfirmasi untuk memilih tombol *sleep* atau tombol *cancel*. Jika menekan tombol *sleep* maka, akan mengganti satu hari kemudian pada *game*. Jika menekan tombol *cancel* maka, akan menutup *panel bed.*



Gambar 3. 27 activity diagram action – bed

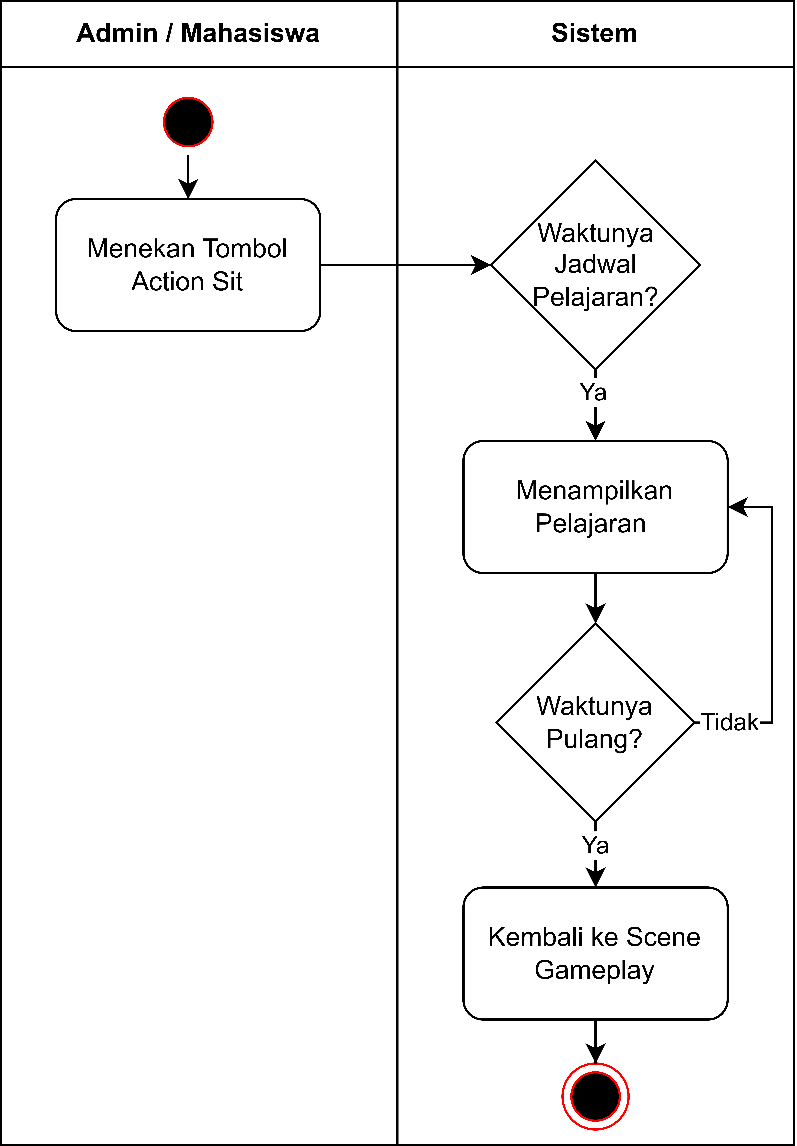
1. *Activity diagram action – laptop*



Gambar 3. 28 activity diagram action – laptop

Pada gambar 3.28, setelah *admin* atau mahasiswa mendekat ke objek *laptop* dan menekan *tombol action* pada *scene gameplay* maka, akan ditampilkan *panel* *laptop*. Pada *panel* ini terdapat beberapa *menu* seperti *mission*, *calendar*, *mail*, *browser*, *music*, dan *shutdown*. Jika *menu* *mission* dipilih maka, akan menampilkan *panel mission.* Jika *menu* *calendar* dipilih maka, akan menampilkan *panel calendar.* Jika *menu* *mail* dipilih maka, akan menampilkan *panel mail.* Jika *menu* *browser* dipilih maka, akan menampilkan *panel browser.* Jika *menu* *music* dipilih maka, akan menampilkan *panel music.* Jika *menu* *shutdown* dipilih maka, akan menutup *laptop.*

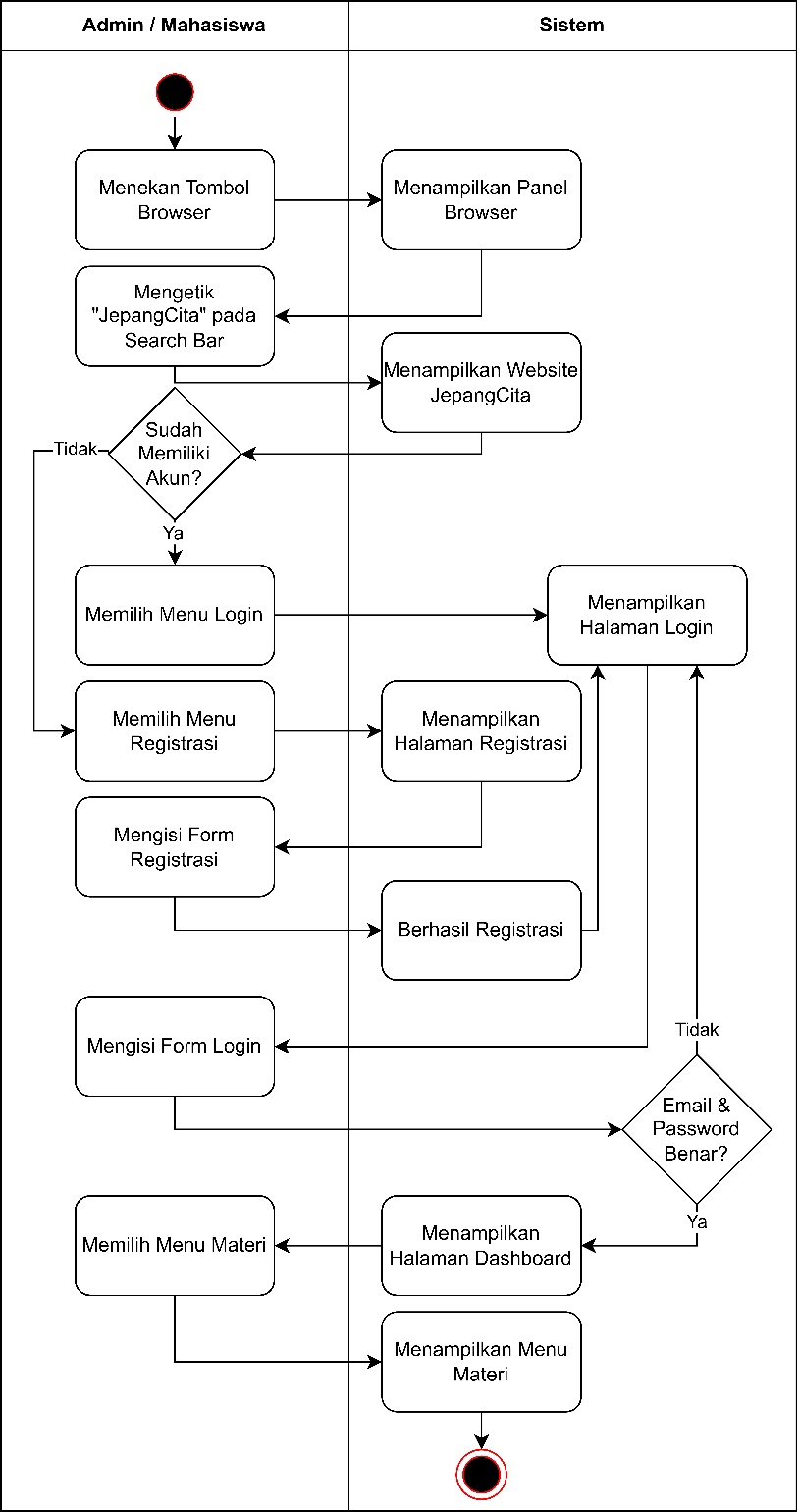
1. *Activity diagram action classroom – sit*



Gambar 3. 29 activity diagram action classroom – sit

Pada gambar 3.29, setelah *admin* atau mahasiswa berada di *scene* *classroom*, untuk memulai pelajaran harus duduk di salah satu kursi di kelas sesuai dengan jadwal pelajaran. Setelah itu, akan ditampilkan materi pembelajaran sesuai dengan pertemuan yang sedang berlangsung*.* Kemudian, jika sudah waktunya pulang maka, akan di arahkan ke *scene* *gameplay*.

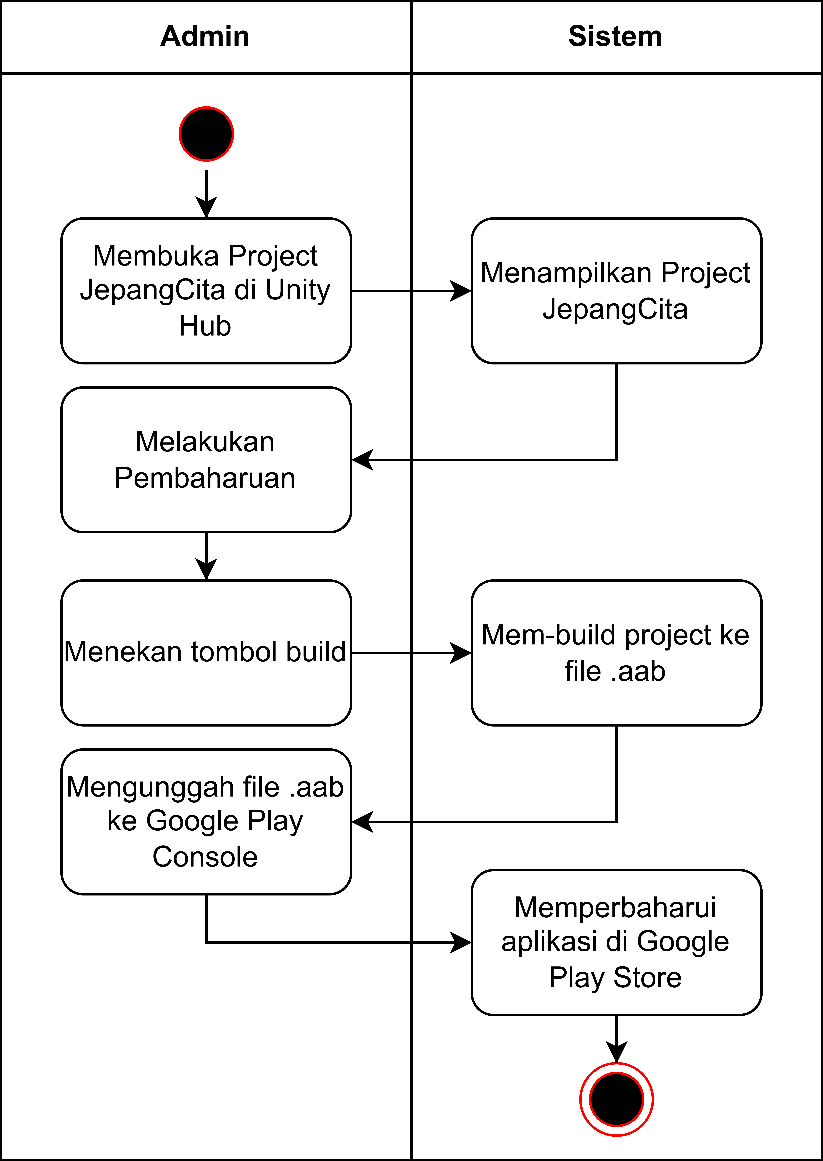
1. *Activity diagram action laptop – materi bahasa Jepang*



Gambar 3. 30 activity diagram action laptop – materi bahasa Jepang

Pada gambar 3.30, setelah *admin* atau mahasiswa membuka *laptop*, untuk mengakses materi harus membuka *menu* *browser* dan mengetikan “JepangCita” pada *search bar*. Setelah itu, akan ditampilkan *landing page*, *menu* *register* dan *menu* *login*. Jika belum pernah mendaftar akun JepangCita, maka harus memilih *menu* *register* dan mengisi *form*. Jika sudah pernah mendaftar, maka harus memilih *menu* *login* dan melakukan proses *login*. Jika sudah *login*, buka *menu* materi untuk mengakses materi bahasa Jepang*.*

1. *Activity diagram update game*



Gambar 3. 31 activity diagram update game

Pada gambar 3.31, setelah *admin* membuka *Unity Hub* dan membuka *project* JepangCita, *admin* melakukan pembaruan pada *game* tersebut. Selanjutnya, *admin* menekan tombol *build* dan membangun *project* ke bentuk *file* “JepangCita.aab”. Setelah *build* selesai, *admin* mengunggah *file* tersebut ke *Google Play Console* dan akan otomatis memperbarui aplikasi di *Google Play Store*.

### *Sequence Diagram*

*Sequence diagram* adalah representasi *UML* yang menggambarkan interaksi di antara objek-objek di dalam dan sekitar sistem, termasuk mahasiswa, tampilan, dan lainnya, melalui pesan-pesan yang digambarkan secara sekuensial sepanjang waktu (M. Syarif & Nugraha, 2020).

1. *Sequence diagram main menu*



Gambar 3. 32 sequence diagram main menu

Pada gambar 3.32, saat *admin* atau mahasiswa membuka aplikasi akan langsung menampilkan *scene main menu*. Pada *scene* ini terdapat berbagai *menu*, seperti: *start game*, *settings*, *credits* dan *quit game*.

1. *Sequence diagram – start game*



Gambar 3. 33 sequence diagram start game

Pada gambar 3.33, setelah membuka aplikasi, *admin* atau mahasiswa dapat memilih *menu* *start game* pada *scene main menu*, kemudian jika sudah pernah memilih karakter, maka akan langsung diarahkan ke scene *gameplay*. Jika belum, maka akan diarahkan ke *scene* *character selection*.

1. *Sequence diagram main menu – settings*



Gambar 3. 34 sequence diagram main menu – settings

Pada gambar 3.34, setelah membuka aplikasi, *admin* atau mahasiswa dapat memilih *menu* *settings* pada *scene main menu* dan akan ditampilkan *panel settings*. Kemudian, dapat mengatur *volume* musik, *volume* efek suara, sensitivitas kamera, dan jarak kamera.

1. *Sequence diagram main menu – credits*



Gambar 3. 35 sequence diagram main menu – credits

Pada gambar 3.35, setelah membuka aplikasi, *admin* atau mahasiswa dapat memilih *menu* *credits* pada *scene main menu*, setelah itu akan menampilkan *panel credits*. Kemudian, akan ditampilkan biodata dari *developer* dan aset-aset yang digunakan pada *game*.

1. *Sequence diagram main menu – quit game*



Gambar 3. 36 sequence diagram main menu – quit game

Pada gambar 3.36, setelah membuka aplikasi, *admin* atau mahasiswa dapat memilih *menu* *quit game* pada *scene main menu*, setelah itu akan ditampilkan *panel quit game*. Kemudian, dapat memilih antara menekan tombol *quit game* untuk keluar dari permainan atau tombol *cancel* untuk menutup *panel quit game*.

1. *Sequence diagram start game – gameplay*



Gambar 3. 37 sequence diagram start game – gameplay

Pada gambar 3.37, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Kemudian, dapat menggerakkan karakter menggunakan *joystick* atau berinteraksi dengan objek dengan cara mendekatkan karakter ke objek yang dipilih, kemudian menekan tombol *action* untuk menjalankan aksi.

1. *Sequence diagram gameplay – mission*



Gambar 3. 38 sequence diagram gameplay – mission

Pada gambar 3.38, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika menekan tombol *mission* maka, akan ditampilkan *panel mission*. Pada *panel* tersebut akan ditampilkan detail dari misi yang harus diselesaikan untuk menyelesaikan permainan.

1. *Sequence diagram gameplay – pause*

Pada gambar 3.39, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika menekan tombol *pause* maka, akan ditampilkan *panel pause* dan menghentikan permainan. Ketika permainan dihentikan maka, waktu dalam permainan akan ikut berhenti. Pada *panel* tersebut akan ditampilkan beberapa *menu*, seperti: *menu* *resume*, *menu* *settings*, dan *main menu*.



Gambar 3. 39 sequence diagram gameplay – pause

1. *Sequence diagram pause – resume*



Gambar 3. 40 sequence diagram pause – resume

Pada gambar 3.40, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika menekan tombol *pause* maka, akan ditampilkan *panel pause* dan menghentikan permainan. Pada *panel* tersebut jika menekan tombol *resume* maka, akan menutup *panel pause* dan melanjutkan permainan.

1. *Sequence diagram pause – settings*



Gambar 3. 41 sequence diagram pause – settings

Pada gambar 3.41, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika menekan tombol *pause* maka, akan ditampilkan *panel pause* dan menghentikan permainan. Pada *panel* tersebut jika menekan tombol *settings* maka, akan ditampilkan *panel settings*. Pada *panel* tersebut dapat mengatur *volume* musik, *volume* efek suara, sensitivitas kamera, dan jarak kamera.

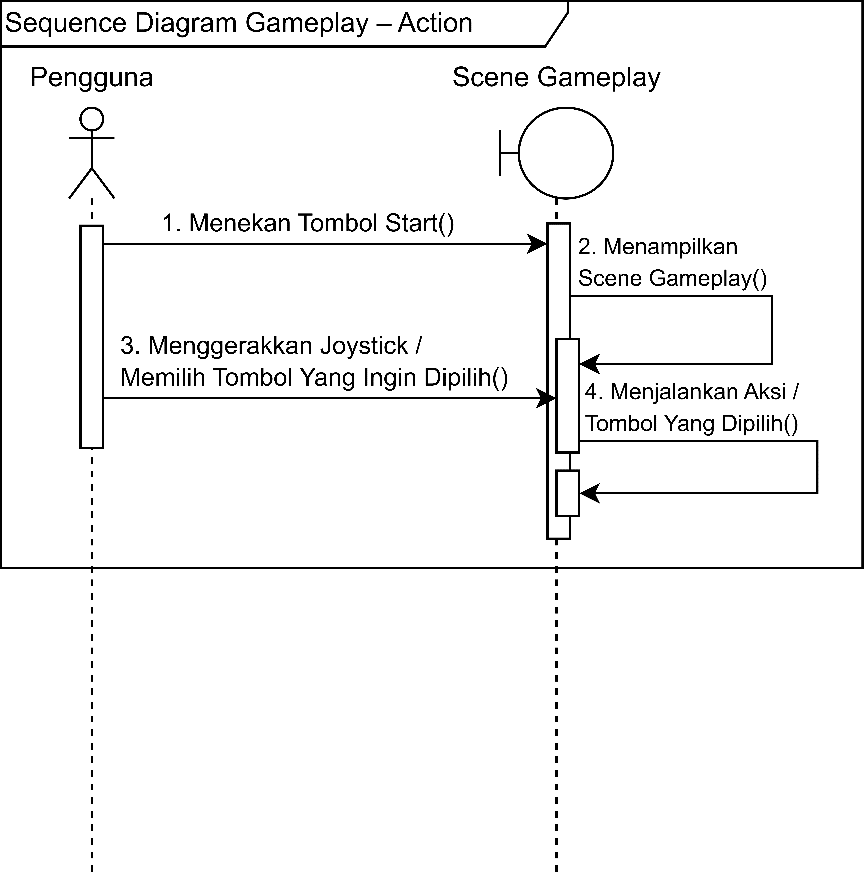
1. *Sequence diagram pause – main menu*

Pada gambar 3.42, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika menekan tombol *pause* maka, akan ditampilkan *panel pause* dan menghentikan permainan. Pada *panel* tersebut jika menekan tombol *main menu* maka, akan ditampilkan *panel main menu*. Pada *panel* tersebut dapat memilih tombol konfirmasi antara menekan tombol *main menu* untuk kembali ke *main menu* atau tombol *cancel* untuk menutup *panel main menu*.



Gambar 3. 42 sequence diagram pause – main menu

1. *Sequence diagram gameplay – action*



Gambar 3. 43 sequence diagram gameplay – action

Pada gambar 3.43, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Pada *scene* ini, karakter dapat melakukan aksi dengan mendekat ke objek aksi. Kemudian, menekan tombol *action*.

1. *Sequence diagram action – classroom*



Gambar 3. 44 sequence diagram action – classroom

Pada gambar 3.44, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika membuka objek pintu keluar dan karakter pergi keluar rumah maka, akan ditampilkan *panel confirm classroom*. Pada *panel* tersebut akan diminta untuk memilih tombol *go* atau tombol *cancel*. Jika menekan tombol *go* maka, akan diarahkan ke *scene classroom*. Jika menekan tombol *cancel* maka, akan menutup *panel confirm classroom* dan kembali ke dalam rumah*.*

1. *Sequence diagram action – bed*



Gambar 3. 45 sequence diagram action – bed

Pada gambar 3.45, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika mendekat ke objek *bed* lalu, menekan tombol *action* maka, akan ditampilkan *panel bed*. Pada *panel* tersebut akan diminta untuk memilih tombol *sleep* atau tombol *cancel*. Jika menekan tombol *sleep* maka, akan mengganti satu hari kemudian pada *game*. Jika menekan tombol *cancel* maka, akan menutup panel *bed*.

1. *Sequence diagram action – laptop*

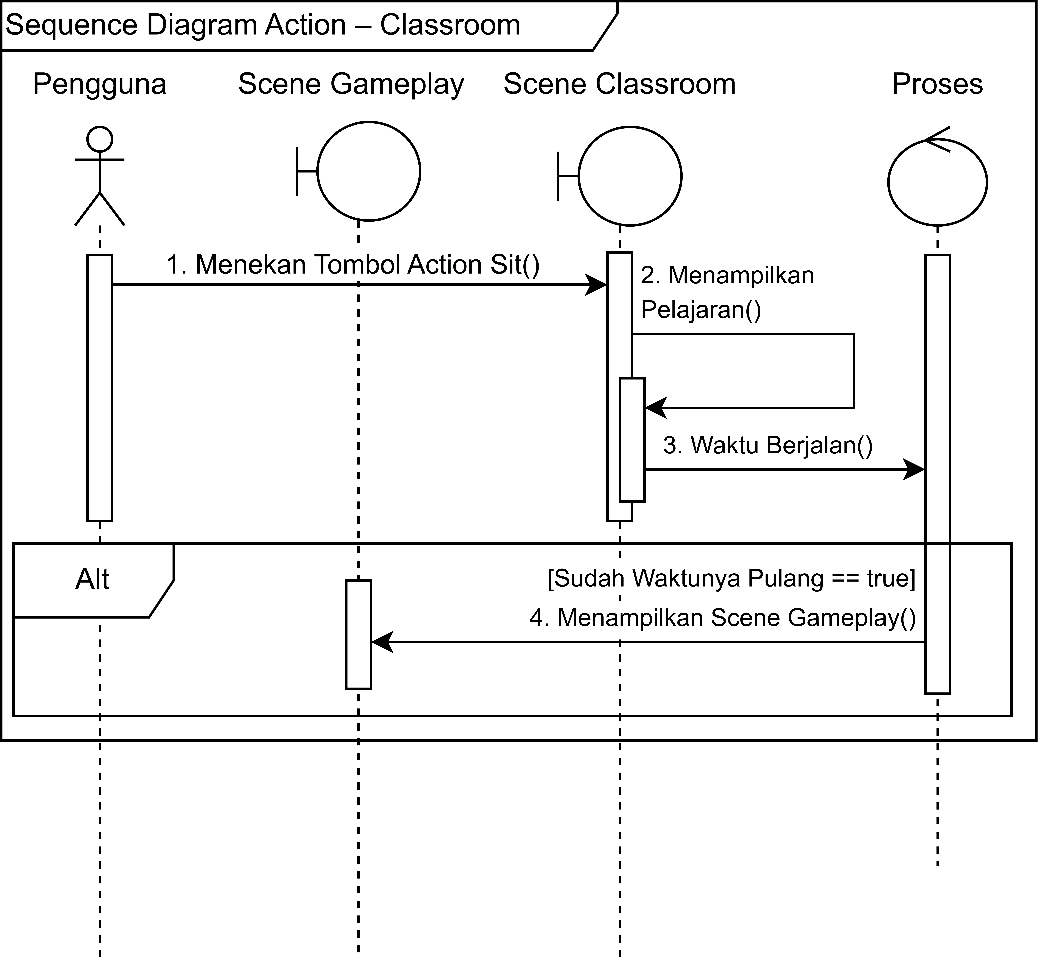


Gambar 3. 46 sequence diagram action – laptop

Pada gambar 3.46, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika mendekat ke objek *laptop* lalu, menekan tombol *action* maka, akan ditampilkan *panel desktop laptop* . Pada *panel* tersebut akan ditampilkan beberapa *menu*, seperti: *menu* *mission*, *menu* *calendar*, *menu mail*, *menu browser*, *menu* *music*, dan *menu* *shutdown*.

1. *Sequence Diagram Action Classroom – Sit*

Pada gambar 3.47, setelah *admin* atau mahasiswa berada pada *scene classroom,* untuk memulai pelajaran, karakter harus duduk di salah satu kursi di kelas dan waktu pada saat itu, sesuai dengan jadwal pelajaran. Setelah itu, akan ditampilkan materi pembelajaran sesuai dengan pertemuan yang sedang berlangsung. Kemudian, jika sudah waktunya pulang maka, akan di arahkan ke *scene gameplay*.



Gambar 3. 47 sequence diagram action classroom – sit

1. *Sequence diagram laptop – materi bahasa Jepang*

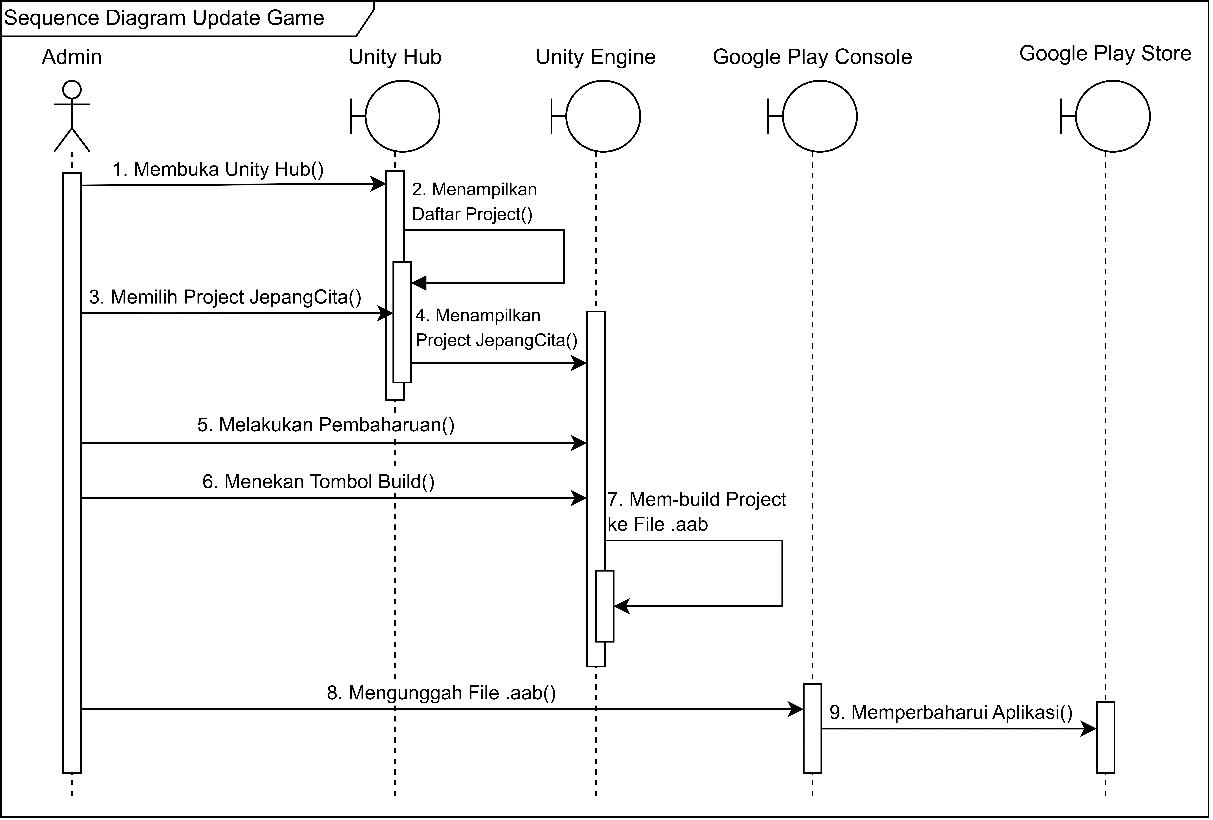
Pada gambar 3.48, setelah *admin* atau mahasiswa menekan tombol *start game* pada *scene main menu* maka, akan diarahkan ke *scene gameplay*. Jika mendekat ke objek *laptop* lalu, menekan tombol *action* maka, akan ditampilkan *panel desktop laptop*. Untuk mengakses materi bahasa Jepang, pada *panel* tersebut pemain harus menekan tombol *browser* kemudian, mengetikan “JepangCita” pada *search bar*. Setelah itu, akan ditampilkan *landing page*, *menu* *register* dan *menu* *login* dari simulasi *website* JepangCita. Jika pemain tersebut belum pernah mendaftar akun JepangCita, maka harus memilih *menu* *register* dan mengisi *form* registrasi. Jika pemain tersebut sudah pernah mendaftar, maka memilih *menu* *login* dan melakukan proses *login*. Jika sudah *login*, buka *menu* materi untuk mengakses materi bahasa Jepang. Materi akan tersedia sesuai dengan pertemuan berlangsung.



Gambar 3. 48 sequence diagram laptop – materi bahasa Jepang

1. *Sequence Diagram Update Game*

Pada gambar 3.49, setelah *admin* membuka *Unity Hub* dan membuka *project* JepangCita, *admin* melakukan pembaruan pada *game* tersebut. Selanjutnya, *admin* menekan tombol *build* dan membangun *project* ke bentuk *file* “JepangCita.aab”. Setelah *build* selesai, *admin* mengunggah *file* tersebut ke *Google Play Console* dan akan otomatis memperbarui aplikasi di *Google Play Store*.



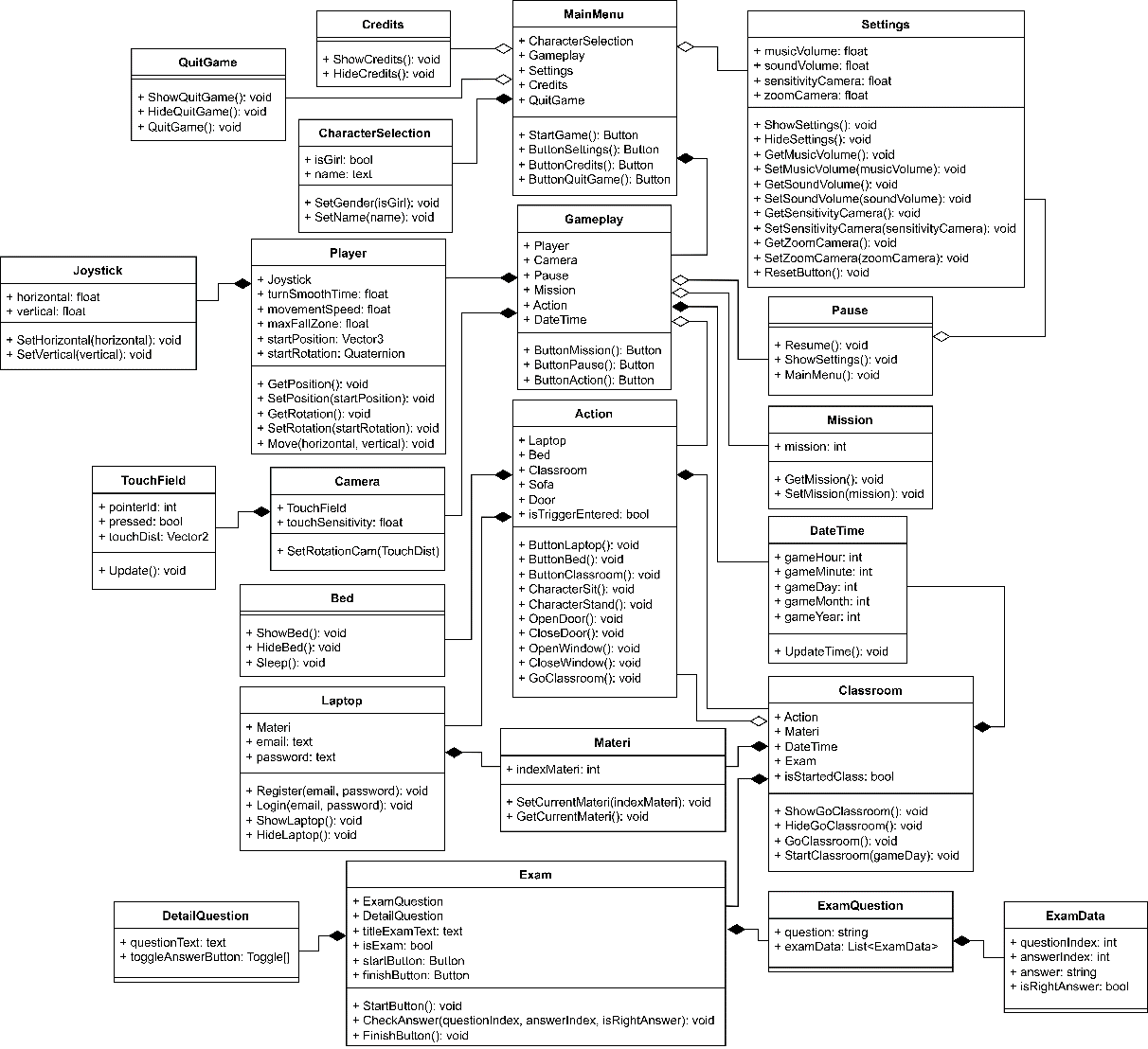
Gambar 3. 49 sequence diagram update game



### *Class Diagram*

*Class diagram* menggambarkan hubungan antar *class* yang di dalamnya terdapat atribut dan fungsi dari suatu objek (Arianti, Fa’izi, Adam, Wulandari, dkk., 2022).

Pada gambar 3.50 adalah *class diagram* yang menggambarkan struktur dan hubungan antar kelas dalam sistem *game* JepangCita. *Class diagram* ini terdiri dari beberapa kelas utama yang saling berhubungan untuk mendukung fungsionalitas *game*, seperti navigasi *menu*, kontrol karakter, pengaturan, manajemen misi, dan evaluasi pembelajaran melalui ujian. Setiap kelas dalam *diagram* ini memiliki atribut dan metode yang dirancang untuk menjalankan fungsi spesifik. *Class* *MainMenu* bertanggung jawab atas navigasi antar *menu* utama, sedangkan *class Player* mengelola kontrol karakter pemain. *Class* *Gameplay* menggabungkan berbagai elemen permainan, termasuk interaksi pemain dengan lingkungan melalui *class Action*, *Mission*, dan *Classroom*. Selain itu, *class* *DateTime* mengatur simulasi waktu dalam permainan, yang relevan untuk aktivitas pembelajaran dan penyelesaian misi. Struktur ini juga mempermudah proses pengembangan *game* karena setiap kelas memiliki tanggung jawab yang jelas dan terpisah.



Gambar 3. 50 class diagram JepangCita

## *User Interface*

*User Interface* adalah tampilan grafis yang berhubungan langsung dengan mahasiswa. Berikut adalah beberapa *user interface* dari *game* simulasi 3D.

1. Rancangan tampilan *main menu*

Pada gambar 3.51, terdapat beberapa tombol yang memiliki fungsinya masing-masing, seperti: tombol mulai/*start* untuk memulai permainan, tombol pengaturan/*settings* untuk mengatur permainan, tombol kredit/*credits* untuk menampilkan detail *developer* dan aset yang digunakan dalam permainan, dan tombol keluar/*quit* untuk keluar dari permainan.



Gambar 3. 51 rancangan tampilan main menu

1. Rancangan tampilan *credits*



Gambar 3. 52 rancangan tampilan credits

Pada gambar 3.52 adalah rancangan layar yang berisi biodata *developer* dan daftar aset-aset yang digunakan dalam permainan.

1. Rancangan tampilan *settings*



Gambar 3. 53 rancangan tampilan settings

Pada gambar 3.53 adalah rancangan yang berisi pengaturan dalam permainan seperti pengaturan *volume* musik, *volume* efek suara, sensitivitas kamera, jarak kamera dan tombol *reset* untuk kembali ke pengaturan semula.

1. Rancangan tampilan *quit game*



Gambar 3. 54 rancangan tampilan quit game

Pada gambar 3.54 adalah rancangan yang berisi konfirmasi untuk pemain apakah ingin keluar permainan atau tidak. Jika ya, pemain memilih tombol keluar permainan dan jika tidak, pemain memilih tombol batal.

1. Rancangan tampilan *character selection*



Gambar 3. 55 rancangan tampilan character selection

Pada gambar 3.55 adalah rancangan yang akan ditampilkan jika pemain pertama kali menekan tombol mulai. Pada rancangan ini berisi dua karakter, yaitu: laki-laki dan perempuan. Pemain diminta memilih salah satu karakter dan mengisi nama karakter tersebut. Setelah pemain menekan tombol lanjutkan, maka akan diarahkan ke *scene gameplay*.

1. Rancangan tampilan *gameplay*



Gambar 3. 56 rancangan tampilan gameplay

Pada gambar 3.56 adalah rancangan yang akan ditampilkan setelah pemain memilih karakter pada *scene character selection* atau ketika pemain menekan tombol mulai pada *scene main menu* dan telah memilih karakter sebelumnya. Pada rancangan ini berisi *panel* tanggal dan jam pada pojok kiri atas lalu, tombol *pause* dan tombol misi pada pojok kanan atas. Kemudian, tombol *joystick* pada kiri bawah dan tombol aksi pada kanan bawah yang akan muncul hanya saat karakter mendekat ke objek aksi dan yang berada di tengah adalah karakter yang bisa digerakkan menggunakan *joystick* dan kamera dengan mengeser layar pada bagian tengah.

1. Rancangan tampilan *mission*



Gambar 3. 57 rancangan tampilan mission

Pada gambar 3.57 adalah rancangan yang akan ditampilkan jika pemain menekan tombol misi pada *scene gameplay*. Pada rancangan ini berisi tombol *close* untuk menutup *panel* misi, judul misi dan detail misi yang harus dilakukan pemain.

1. Rancangan tampilan *pause*



Gambar 3. 58 rancangan tampilan pause

Pada gambar 3.58 adalah rancangan yang akan ditampilkan jika pemain menekan tombol *pause* pada *scene gameplay*. Pada rancangan ini berisi beberapa *menu*, yaitu: tombol *close*, tombol lanjutkan, tombol pengaturan dan tombol *main menu*.

1. Rancangan tampilan *action – laptop*



Gambar 3. 59 rancangan tampilan action – laptop

Pada gambar 3.59 adalah rancangan yang akan ditampilkan jika pemain menekan tombol aksi *laptop* pada *scene gameplay*. Pada rancangan ini berisi beberapa *menu*, yaitu: *menu* misi, *menu* kalender, *menu* pesan, *menu* *browser*, *menu* musik dan tombol *shutdown*.

1. Rancangan tampilan *action – bed*



Gambar 3. 60 rancangan tampilan action – bed

Pada gambar 3.60 adalah rancangan yang akan ditampilkan jika pemain menekan tombol aksi *bed* pada *scene gameplay*. Pada rancangan ini berisi konfirmasi untuk pemain apakah ingin melakukan tidur atau tidak. Jika ya, pemain memilih tombol tidur dan jika tidak, pemain memilih tombol batal.

1. Rancangan tampilan *action – classroom*



Gambar 3. 61 rancangan tampilan action – classroom

Pada gambar 3.61 adalah rancangan yang akan ditampilkan jika pemain keluar rumah pada *scene gameplay*. Pada rancangan ini berisi konfirmasi untuk pemain apakah ingin pergi ke kelas atau tidak. Jika ya, pemain memilih tombol pergi dan jika tidak, pemain memilih tombol batal.

# BAB IV IMPLEMENTASI DAN PENGUJIAN



## Spesifikasi

Spesifikasi adalah serangkaian karakteristik atau persyaratan teknis yang menggambarkan fitur, fungsi, dan kinerja suatu sistem. Spesifikasi ini bertujuan untuk memastikan bahwa sistem tersebut memenuhi kebutuhan dan harapan mahasiswa serta standar yang ditetapkan.



### Spesifikasi Perangkat Lunak

Spesifikasi dalam implementasi dan pengujian *game* “JepangCita: Game Simulasi 3D”menggunakan perangkat lunak sebagai berikut:

Tabel 4. 1 tabel spesifikasi perangkat lunak

|  |  |  |
| --- | --- | --- |
| **No.** | **Perangkat Lunak** | **Keterangan** |
| 1. | Sistem Operasi | *Windows 10 Pro* |
| 2. | *Integrated Development Environment (IDE)* | *Visual Studio 2019* |
| 3. | *Game Engine* | *Unity Engine 2020* |
| 4. | 3D Animasi Karakter | *Adobe Mixamo* |
| 5. | Desain Grafis | *Adobe Photoshop CC 2021* |
| 6. | *Web Browser* | *Google Chrome* |
| 7. | Desain *UI/UX* | *Figma* |

### Spesifikasi Perangkat Keras

Spesifikasi dalam implementasi dan pengujian *game* “JepangCita: Game Simulasi 3D”menggunakan perangkat keras *laptop* sebagai berikut:

Tabel 4. 2 tabel spesifikasi perangkat keras laptop

|  |  |  |
| --- | --- | --- |
| **No.** | **Perangkat Keras** | **Keterangan** |
| 1. | *Model* | *Laptop Lenovo Ideapad 320 14IKB-80XK* |
| 2. | *Processor* | *Intel(R) Core(TM) i5-7200U CPU @ 2.50 GHz, 2712 Mhz, 2 Core(s), 4 Logical Processor(s)* |
| 3. | *Memory RAM* | *4GB DDR4* |
| 4. | Penyimpanan Internal | *SSD Verbatim 256GB Vi550 S3 SATA III 2,5 inch* |
| 5. | Internet | *WiFi IndiHome 20 Mbps* |

Spesifikasi dalam pengujian *game* “JepangCita: Game Simulasi 3D”menggunakan perangkat keras *smartphone* sebagai berikut:

Tabel 4. 3 tabel spesifikasi perangkat keras smartphone

|  |  |  |
| --- | --- | --- |
| **No.** | **Perangkat Keras** | **Keterangan** |
| 1. | *Model* | *Smartphone Poco X5 5G* |
| 2. | *Processor* | *Snapdragon® 695* |
| 3. | *Memory RAM* | *6GB* |
| 4. | Penyimpanan Internal | *128GB* |
| 5. | *Android Version* | *14 Upside-down Cake* |
| 6. | Sistem Operasi | *Xiaomi HyperOS v1.0.2.0.UMPIDXM* |

## Implementasi Program

Implementasi adalah tahap di mana sistem mulai dioperasikan secara nyata, sehingga dapat diketahui apakah sistem tersebut benar-benar dapat mencapai tujuan yang diinginkan.



### Tampilan *Splash Screen*



Gambar 4. 1 tampilan dari splash screen

Pada gambar 4.1 merupakan halaman pembuka dari *game* sebelum mahasiswa masuk ke *main menu*.

### Tampilan *Main Menu*



Gambar 4. 2 tampilan dari main menu

Pada gambar 4.2, terdapat beberapa tombol yang bisa dipilih seperti tombol *start game* untuk memulai permainan, tombol *settings* untuk melakukan pengaturan permainan, tombol *credits* untuk menampilkan data diri *developer* dan aset-aset yang digunakan pada *game*, dan tombol *quit game* untuk keluar dari permainan.

### Tampilan Pengaturan



Gambar 4. 3 tampilan dari pengaturan

Pada gambar 4.3, mahasiswa dapat mengatur berbagai opsi pengaturan dalam permainan seperti pengaturan *volume* musik, *volume* efek suara, sensitivitas kamera, jarak kamera dan tombol *reset* untuk kembali ke pengaturan semula.

### Tampilan Kredit



Gambar 4. 4 tampilan dari kredit

Pada gambar 4.4 tampilan dari kredit merupakan *panel* kredit yang berisikan biodata *developer* dan daftar aset-aset yang digunakan.

### Tampilan Keluar Permainan



Gambar 4. 5 tampilan dari keluar permainan

Pada gambar 4.5 merupakan *panel* konfirmasi untuk melakukan konfirmasi apakah mahasiswa ingin keluar permainan atau tidak.

### Tampilan *Character Selection*



Gambar 4. 6 tampilan dari character selection

Pada gambar 4.6, mahasiswa dapat memilih karakter sesuai dengan keinginan, gunakan tombol panah kanan/kiri untuk mengganti karakter, tombol pilih karakter untuk memilih karakter dan melanjutkan permainan ke *scene* *gameplay*.

### Tampilan *Gameplay*



Gambar 4. 7 tampilan dari gameplay

Pada gambar 4.7, mahasiswa dapat melakukan beberapa aktivitas struktur navigasi, seperti: tombol *joystick* untuk menggerakkan karakter, tombol aksi yang akan muncul jika karakter mendekat dengan objek aksi (tombol tersebut digunakan untuk melakukan aksi sesuai dengan objeknya), tombol misi untuk menampilkan misi yang harus diselesaikan, tombol *pause* untuk menghentikan permainan dan menampilkan tombol-tombol *menu*, yaitu: tombol melanjutkan permainan untuk melanjutkan permainan, tombol pengaturan untuk melakukan pengaturan dan tombol *main menu* untuk kembali ke *scene* *main menu*.

### Tampilan *Gameplay* – Pembelajaran *Online*



Gambar 4. 8 tampilan dari gameplay – pembelajaran online

Pada gambar 4.8, mahasiswa dapat melakukan pembelajaran *online* dalam *game* melalui *web browser* pada *game*. Untuk mengaksesnya, mahasiswa harus mendekati *laptop* lalu, menekan tombol aksi. Setelah itu, membuka *web browser* pada *game*. Pada *web browser* mahasiswa mengetikan *keyword* “JepangCita” untuk mengakses halaman *web* simulasi. Jika mahasiswa belum memiliki akun, maka harus melakukan registrasi terlebih dahulu kemudian, melakukan proses *login*. Setelah itu, buka *menu* materi. Seluruh instruksi sudah tersedia dalam *game* melalui misi.

### Tampilan *Classroom*

Pada gambar 4.9, mahasiswa sedang berada di *scene classroom*. Pada *scene* tersebut, mahasiswa dapat mengikuti pembelajaran secara *offline* yang dapat diikuti sesuai dengan jadwal pelajaran yang didapatkan setelah mahasiswa melakukan registrasi pada *website* JepangCita.



Gambar 4. 9 tampilan dari classroom

### Tampilan *Classroom –* Pembelajaran *Offline*



Gambar 4. 10 tampilan classroom – pembelajaran offline

Pada gambar 4.10, mahasiswa sedang mengikuti pembelajaran kelas secara *offline* yang sedang berjalan, pada setiap pertemuan materi yang diberikan akan berbeda-beda. Setelah satu materi yang diikuti selesai, maka akan diadakan ujian materi tersebut untuk pengambilan nilai.

### Tampilan *Classroom –* Ujian

Pada gambar 4.11, mahasiswa sedang mengikuti ujian *hiragana* secara *offline*, mahasiswa akan mengisi soal pilihan ganda dengan empat pilihan jawaban. Waktu ujian sama seperti waktu saat pembelajaran biasa. Jika mahasiswa sudah mengisi semua jawaban dan merasa sudah benar dan ingin segera menyelesaikan ujian, mahasiswa dapat menekan tombol selesai.



Gambar 4. 11 tampilan dari classroom – ujian

## Pengujian Sistem

Pengujian sistem merupakan proses evaluasi untuk memastikan bahwa perangkat lunak yang telah dikembangkan dapat beroperasi sesuai dengan standar yang ditetapkan. Jenis pengujian sistem yang digunakan adalah *Functional Test* dan *User Acceptance Test (UAT)*.

### *Functional Testing*

*Black box testing* adalah pendekatan pengujian perangkat lunak yang mengevaluasi fungsionalitas suatu aplikasi tanpa memperhatikan struktur internal atau cara kerjanya. Metode pengujian *black box testing* memungkinkan pengembang perangkat lunak untuk menyusun serangkaian kondisi *input* yang mencakup semua persyaratan fungsional program. Pengujian dilakukan dengan memilih sejumlah modul yang mencakup berbagai jenis data untuk memastikan bahwa program hanya menerima *input* dengan jenis data yang sesuai. Selain itu, pengujian juga memeriksa antarmuka pengguna aplikasi itu sendiri. Proses pengujian *black box* pada *game* “JepangCita: Game Simulasi 3D” adalah sebagai berikut:

1. Pengujian *black box* pada *scene main menu*

Tabel 4. 4 tabel pengujian scene main menu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kode** | **Deskripsi** | **Hasil yang Diharapkan** | **Hasil Pengujian** | **Status** |
| PSMM1 | tombol mulaipertama kali ditekan | masuk ke *scene character selection* | masuk ke *scene character selection* | Valid |
| PSMM2 | tombol mulaiditekan jika sudah pernah memilih karakter sebelumnya | masuk ke *scene gameplay* | masuk ke *scene gameplay* | Valid |
| PSMM3 | tombol kredit ditekan | menampilkan *panel* kredit | menampilkan *panel* kredit | Valid |
| PSMM4 | tombol tutup *panel* kredit ditekan | menutup *panel* kredit | menutup *panel* kredit | Valid |
| PSMM5 | tombol pengaturan ditekan | menampilkan *panel* pengaturan | menampilkan *panel* pengaturan | Valid |
| PSMM6 | *slider* musik digeser | mengubah *volume* musik | mengubah *volume* musik | Valid |
| PSMM7 | *slider* efek suara digeser | mengubah *volume* efek suara | mengubah *volume* efek suara | Valid |
| PSMM8 | *slider* sensitivitas kamera | mengubah sensitivitas kamera | mengubah sensitivitas kamera | Valid |
| PSMM9 | *slider* jarak kamera | mengubah jarak kamera | mengubah jarak kamera | Valid |
| PSMM10 | tombol *reset* ditekan | mengatur ulang konfigurasi *game* | mengatur ulang konfigurasi *game* | Valid |
| PSMM11 | tombol tutup *panel* pengaturan ditekan | menutup *panel* pengaturan | menutup *panel* pengaturan | Valid |
| PSMM12 | tombol keluar permainan ditekan | menampilkan *panel* keluar permainan | menampilkan *panel* keluar permainan | Valid |
| PSMM13 | tombol keluar permainan ditekan | keluar dari permainan | keluar dari permainan | Valid |
| PSMM14 | tombol batal *panel* keluar permainan ditekan | menutup *panel* keluar permainan | menutup *panel* keluar permainan | Valid |

1. Pengujian *black box* pada *scene character selection*

Tabel 4. 5 tabel pengujian scene character selection

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kode** | **Deskripsi** | **Hasil yang Diharapkan** | **Hasil Pengujian** | **Status** |
| PSCS1 | tombol panah kanan ditekan | mengganti karakter | mengganti karakter | Valid |
| PSCS2 | tombol panah kiri ditekan | mengganti karakter | mengganti karakter | Valid |
| PSCS3 | tombol pilih karakter ditekan | memilih karakter dan menampilkan panel *input* nama | memilih karakter dan menampilkan panel *input* nama | Valid |
| PSCS4 | tombol lanjutkan ditekan | masuk ke *scene gameplay* | masuk ke *scene gameplay* | Valid |

1. Pengujian *black box* pada *scene gameplay*

Tabel 4. 6 tabel pengujian scene gameplay

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kode** | **Deskripsi** | **Hasil yang Diharapkan** | **Hasil Pengujian** | **Status** |
| PSG1 | tombol misi ditekan | menampilkan *panel* misi | menampilkan *panel* misi | Valid |
| PSG2 | tombol tutup *panel* misi ditekan | menutup *panel* misi | menutup *panel* misi | Valid |
| PSG3 | tombol *pause* ditekan pada *panel pause* | menampilkan *panel pause* dan menghentikan *game* | menampilkan *panel pause* dan menghentikan *game* | Valid |
| PSG4 | tombol lanjutkan ditekan pada *panel pause* | menutup *panel pause* dan melanjutkan *game* | menutup *panel pause* dan melanjutkan *game* | Valid |
| PSG5 | tombol pengaturan ditekan pada *panel pause* | menampilkan *panel* pengaturan | menampilkan *panel* pengaturan | Valid |
| PSG6 | tombol tutup *panel* pengaturan ditekan | menutup *panel* pengaturan | menutup *panel* pengaturan | Valid |
| PSG7 | tombol menu utama ditekan pada *panel pause* | menampilkan panel menu utama | menampilkan *panel* menu utama | Valid |
| PSG8 | tombol menu utama ditekan | kembali ke *scene* *main menu* | kembali ke *scene main menu* | Valid |
| PSG9 | tombol batal *panel* menu utama ditekan | menutup *panel* menu utama | menutup *panel* menu utama | Valid |
| PSG10 | tombol *joystick* digerakkan | menggerakkan karakter | menggerakkan karakter | Valid |
| PSG11 | tombol aksi *bed* ditekan | menampilkan *panel bed* | menampilkan *panel bed* | Valid |
| PSG12 | tombol tidur ditekan | karakter tidur dan melewati waktu sehari | karakter tidur dan melewati waktu sehari | Valid |
| PSG13 | tombol batal *panel* *bed* ditekan | menutup *panel bed* | menutup *panel bed* | Valid |
| PSG14 | tombol aksi *laptop* ditekan | menampilkan *panel laptop* | menampilkan *panel laptop* | Valid |
| PSG15 | tombol misi ditekan | menampilkan *panel* misi | menampilkan *panel* misi | Valid |
| PSG16 | tombol *minimize* *panel* misi ditekan | *minimize* panel misi | *minimize* panel misi | Valid |
| PSG17 | tombol tutup *panel* misi ditekan | menutup *panel* misi | menutup *panel* misi | Valid |
| PSG18 | tombol kalender ditekan | menampilkan *panel* kalender | menampilkan *panel* kalender | Valid |
| PSG19 | tombol *minimize* *panel* kalender ditekan | *minimize* *panel* kalender | *minimize* *panel* kalender | Valid |
| PSG20 | tombol tutup *panel* kalender ditekan | menutup *panel* kalender | menutup *panel* kalender | Valid |
| PSG21 | tombol bulan depan kalender ditekan | mengganti bulan berikutnya | mengganti bulan berikutnya | Valid |
| PSG22 | tombol bulan lalu kalender ditekan | mengganti bulan sebelumnya | mengganti bulan sebelumnya | Valid |
| PSG23 | tombol pesan ditekan | menampilkan *panel* pesan | menampilkan *panel* pesan | Valid |
| PSG24 | tombol *minimize* pesan ditekan | *minimize* panel pesan | *minimize* panel pesan | Valid |
| PSG25 | tombol tutup pesan ditekan | menutup *panel* pesan | menutup *panel* pesan | Valid |
| PSG26 | tombol tulis pesan ditekan | menampilkan *panel* tulis pesan | menampilkan *panel* tulis pesan | Valid |
| PSG27 | tombol tutup tulis pesan ditekan | menutup *panel* tulis pesan | menutup *panel* tulis pesan | Valid |
| PSG28 | tombol kirim pesan ditekan | mengirim pesan dengan syarat semua *input* diisi | mengirim pesan dengan syarat semua *input* diisi | Valid |
| PSG29 | tombol *inbox* pesan ditekan | menampilkan *panel inbox* | menampilkan *panel inbox* | Valid |
| PSG30 | tombol terkirim pesan ditekan | menampilkan *panel* pesan terkirim | menampilkan *panel* pesan terkirim | Valid |
| PSG31 | tombol draf pesan ditekan | menampilkan *panel* pesan draf | menampilkan *panel* pesan draf | Valid |
| PSG32 | tombol sampah pesan ditekan | menampilkan *panel* pesan sampah | menampilkan *panel* pesan sampah | Valid |
| PSG33 | tombol *browser* ditekan | menampilkan *panel browser* | menampilkan *panel browser* | Valid |
| PSG34 | tombol *minimize* *browser* ditekan | *minimize panel browser* | *minimize panel browser* | Valid |
| PSG35 | tombol tutup *panel* *browser* ditekan | menutup *panel browser* | menutup *panel browser* | Valid |
| PSG36 | *input search bar* diisi “JepangCita” | menampilkan *website* JepangCita | menampilkan *website* JepangCita | Valid |
| PSG37 | tombol registrasi ditekan | menampilkan *form* registrasi | menampilkan *form* registrasi | Valid |
| PSG38 | tombol *login* ditekan | menampilkan *form login* | menampilkan *form login* | Valid |
| PSG39 | tombol lupa *password* ditekan | menampilkan *form reset password* | menampilkan *form reset password* | Valid |
| PSG40 | tombol jadwal ditekan | menampilkan *panel* jadwal | menampilkan *panel* jadwal | Valid |
| PSG41 | tombol materi ditekan | menampilkan *panel* materi | menampilkan *panel* materi | Valid |
| PSG42 | tombol *logout* ditekan | melakukan *logout* dan menampilkan *panel login* | melakukan *logout* dan menampilkan *panel login* | Valid |
| PSG43 | tombol musik ditekan | menampilkan *panel* musik | menampilkan *panel* musik | Valid |
| PSG44 | tombol *minimize* musik ditekan | *minimize* *panel* musik | *minimize panel* musik | Valid |
| PSG45 | tombol tutup *panel* musik ditekan | menutup *panel* musik | menutup *panel* musik | Valid |
| PSG46 | tombol musik sebelumnya ditekan | mengganti musik sebelumnya | mengganti musik sebelumnya | Valid |
| PSG47 | tombol musik selanjutnya ditekan | mengganti musik selanjutnya | mengganti musik selanjutnya | Valid |
| PSG48 | tombol *pause* musik ditekan | menghentikan musik | menghentikan musik | Valid |
| PSG49 | tombol *shuffle* musik ditekan | mengacak daftar musik | mengacak daftar musik | Valid |
| PSG50 | tombol *repeat* musik ditekan | memutar ulang musik ketika musik selesai | memutar ulang musik ketika musik selesai | Valid |
| PSG51 | tombol *shutdown* ditekan | menutup *panel laptop* | menutup *panel laptop* | Valid |
| PSG52 | tombol aksi pintu kamar mandi ditekan | membuka pintu kamar mandi | membuka pintu kamar mandi | Valid |
| PSG53 | tombol aksi sofa ditekan | karakter duduk di sofa | karakter duduk di sofa | Valid |
| PSG54 | tombol aksi pintu depan rumah ditekan | membuka pintu depan rumah | membuka pintu depan rumah | Valid |
| PSG55 | karakter berjalan keluar rumah | menampilkan *panel* pergi kursus | menampilkan *panel* pergi kursus | Valid |
| PSG56 | tombol pergi ditekan | masuk ke *scene classroom* | masuk ke *scene classroom* | Valid |
| PSG57 | tombol batal pada *panel* pergi kursus ditekan | menutup *panel* pergi kursus | menutup *panel* pergi kursus | Valid |

1. Pengujian *black box* pada *scene classroom*

Tabel 4. 7 tabel pengujian scene classroom

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kode** | **Deskripsi** | **Hasil yang Diharapkan** | **Hasil Pengujian** | **Status** |
| PSC1 | tombol aksi duduk ditekan | karakter duduk di kursi | karakter duduk di kursi | Valid |
| PSC2 | tombol aksi duduk ditekan jika sesuai jadwal pelajaran | karakter duduk di kursi dan mulai pembelajaran | karakter duduk di kursi dan mulai pembelajaran | Valid |
| PSC3 | tombol aksi pintu keluar ditekan | menampilkan *panel* pulang | menampilkan *panel* pulang | Valid |
| PSC4 | tombol pulang ditekan | masuk ke *scene gameplay* | masuk ke *scene gameplay* | Valid |
| PSC5 | tombol batal pada *panel* pulang ditekan | menutup *panel* pulang | menutup *panel* pulang | Valid |

Berdasarkan hasil dari keempat *scene* yang diuji, total skenario uji yang berhasil mencapai 100%. Ini menunjukkan bahwa semua fitur yang diuji dalam *game* “JepangCita: Game Simulasi 3D” berfungsi dengan baik sesuai dengan spesifikasi dan ekspektasi yang ditetapkan. Pengujian *functional test* ini memastikan bahwa aplikasi telah memenuhi persyaratan fungsionalnya dan siap untuk tahap pengujian berikutnya, yaitu pengujian *User Acceptance Test (UAT)*, yang akan dilakukan menggunakan kuesioner yaitu mengumpulkan umpan balik dari mahasiswa.

### Kuesioner *User Acceptance Testing*

Kuesioner adalah suatu teknik pengumpulan data yang melibatkan pemberian serangkaian pertanyaan atau pernyataan tertulis kepada responden. Tujuan dari kuesioner ini adalah mengevaluasi apakah sistem yang sedang digunakan sudah memenuhi kebutuhan mahasiswa, mengukur tingkat kepuasan mahasiswa, dan mengidentifikasi *game* yang perlu ditingkatkan.

Kuesioner dalam *User Acceptance Testing* *(UAT)* berfungsi sebagai alat untuk mengumpulkan data yang penting mengenai pengalaman pengguna, sehingga *developer* dapat membuat perbaikan yang diperlukan sebelum *game* diluncurkan secara resmi. Dengan menggunakan kuesioner, diharapkan pengujian dapat memberikan wawasan yang berharga tentang bagaimana *game* diterima oleh *target audiens* dan apakah ada aspek yang perlu diperbaiki atau ditingkatkan.

#### Daftar Pertanyaan Kuesioner

Berikut adalah daftar pertanyaan yang terdapat dalam kuesioner *game* “JepangCita: Game Simulasi 3D”. Berikut adalah Keterangan dari jawaban kuesioner:

1. SKB: Sangat Kurang Baik
2. KB: Kurang Baik
3. C: Cukup
4. B: Baik
5. SB: Sangat Baik

Tabel 4. 8 tabel daftar pertanyaan kuesioner

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Pertanyaan** | **SKB** | **KB** | **C** | **B** | **SB** |
| **1** | **2** | **3** | **4** | **5** |
| 1. | Bagaimana Anda menilai grafis dan animasi dalam game ini? |  |  |  |  |  |
| 2. | Bagaimana Anda menilai kontrol dan antarmuka mahasiswa (GUI) dalam game ini? |  |  |  |  |  |
| 3. | Bagaimana kualitas audio, termasuk musik dan efek suara, dalam game ini? |  |  |  |  |  |
| 4. | Seberapa efektif game ini dalam membantu Anda belajar kosakata bahasa Jepang? |  |  |  |  |  |
| 5. | Seberapa efektif game ini dalam membantu Anda memahami tata bahasa Jepang? |  |  |  |  |  |
| 6. | Seberapa efektif game ini dalam meningkatkan kemampuan berbicara bahasa Jepang Anda? |  |  |  |  |  |
| 7. | Seberapa efektif game ini dalam meningkatkan kemampuan membaca dan menulis bahasa Jepang Anda? |  |  |  |  |  |
| 8. | Apakah Anda merasa termotivasi untuk belajar bahasa Jepang lebih lanjut setelah bermain game ini? |  |  |  |  |  |
| 9. | Seberapa puas Anda dengan konten edukatif yang disediakan dalam game ini? |  |  |  |  |  |
| 10. | Seberapa baik pengalaman Anda dalam berbahasa Jepang setelah bermain game ini? |  |  |  |  |  |

#### Hasil Kuesioner *User Acceptance Testing*

Hasil kuesioner *User Acceptance Testing* adalah data yang dihasilkan dari pertanyaan yang sudah disebarkan dan sudah diberi jawaban oleh responden melalui *Google Forms*. Data tersebut memberikan gambaran tentang pandangan, opini, atau pengalaman responden terkait topik yang ditanyakan.

Tabel 4. 9 hasil kuesioner user acceptance testing

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Nama Responden** | **Pertanyaan** | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 1. | Milda | B | B | B | SB | B | C | B | SB | B | KB |
| 2. | Muhammad Irgi Al Ghithraf | B | B | SB | B | C | C | B | B | B | SB |
| 3. | Aldo Hermawan Suryana | B | B | B | B | B | B | B | B | B | B |
| 4. | Ikhsan Rasyid | B | B | B | B | B | B | B | B | B | B |
| 5. | Andre Farhan Saputra | SB | SB | SB | SB | SB | SB | B | SB | SB | SB |
| 6. | Sulthan Tangguh Saefullah | B | SB | B | SB | B | B | B | C | B | C |
| 7. | Tharmidzi Hermawan | C | C | C | C | C | KB | C | KB | KB | KB |
| 8. | Muhammad Fahri | SB | SB | SB | SB | SB | SB | SB | SB | SB | SB |
| 9. | Devi Nur Suryanita | SB | SB | B | B | B | B | SB | B | B | B |
| 10. | Aurora Rachel | C | C | B | B | C | C | B | B | B | B |
| 11. | Yaasmiin Nuhaa Asa Putri | B | B | SB | SB | SB | B | B | B | SB | B |
| 12. | Salhan Taris Agusti | SB | SB | SB | SB | SB | SB | SB | SB | SB | SB |
| 13. | Azriel FachrulRezy | B | B | B | C | C | C | C | B | B | C |
| 14. | Ibnu Hajar | B | B | B | SB | SB | B | B | B | B | SB |
| 15. | Rangga Ariansyah | B | B | B | B | C | B | B | C | B | B |
| 16. | Ikmalul ilmi | SB | B | SB | B | SB | SB | SB | SB | B | SB |
| 17. | Yogi Rizky Pangestu | SB | SB | SB | SB | SB | SB | SB | SB | SB | SB |
| 18. | Rafli Febrian Qasthalani | SB | B | B | SB | SB | SB | B | C | B | SB |
| 19. | Rio Antono | SB | SB | SB | B | SB | B | B | B | SB | SB |
| 20. | Bani Maskur Muhammad Al-Walad | SB | SB | SB | SB | SB | SB | SB | SB | SB | SB |
| 21. | Muhammad Irfannurroja | SB | B | SB | SB | B | SB | SB | B | SB | B |
| 22. | Mochamad Aminnur | B | B | B | B | B | B | B | C | B | B |

Pada tabel 4.6 adalah jawaban dari 22 responden dengan jumlah 10 pertanyaan.

#### Perhitungan Skor Kuesioner Menggunakan Skala *Likert*

Dalam perhitungan skor kuesioner yang menggunakan skala *likert*, terdapat beberapa langkah penting yang harus dilakukan untuk mendapatkan interpretasi yang akurat terhadap hasil kuesioner. Langkah-langkah perhitungannya adalah sebagai berikut:

1. Menentukan bobot skala *likert*

Bobot skala *likert* yang digunakan dalam kuesioner adalah sebagai berikut:

1. Sangat Kurang Baik (SKB): 1
2. Kurang Baik (KB): 2
3. Cukup (C): 3
4. Baik (B): 4
5. Sangat Baik (SB): 5
6. Perhitungan total skor berdasarkan pilihan responden

Adapun perhitungan jumlah responden yang mengisi kuesioner adalah sebagai berikut:

Jumlah responden: 22

Jumlah soal: 10

Total jumlah responden yang terkumpul dari kuesioner adalah:

Total jumlah responden = Jumlah responden × Jumlah soal

T = 22 x 10 = 220

Tabel 4. 10 tabel perhitungan kuesioner

|  |  |  |
| --- | --- | --- |
| **Pilihan Angka Skor Likert** | **Total Jumlah Responden yang memilih** | **Hasil** |
| Responden yang menjawab Sangat Kurang Baik (1) | 0 | 0 |
| Responden yang menjawab Kurang Baik (2) | 5 | 10 |
| Responden yang menjawab Cukup (3) | 24 | 72 |
| Responden yang menjawab Baik (4) | 102 | 408 |
| Responden yang menjawab Sangat Baik (5) | 89 | 445 |
| **Total Skor** | | **935** |

1. Menentukan skor tertinggi dan terendah

Untuk mendapatkan hasil interpretasi, diperlukan skor tertinggi (Y) dan skor terendah (X) untuk *item* penilaian dengan rumus sebagai berikut:

1. Skor Tertinggi (Y): Skor tertinggi yang dapat diperoleh dalam kuesioner. Dengan skala *likert*, skor tertinggi adalah 5 (Sangat Baik). Untuk seluruh kuesioner, skor tertinggi dihitung dengan:

𝑌 = Skor Tertinggi × Total Jumlah Responden

1. Skor Terendah (X): Skor terendah yang dapat diperoleh dalam kuesioner. Dengan skala *likert*, skor terendah adalah 1 (Sangat Kurang Baik). Untuk seluruh kuesioner, skor terendah dihitung dengan:

𝑋 = Skor Terendah × Total Jumlah Responden

Jumlah skor tertinggi untuk *item* Sangat Baik adalah 5 x 220 = 1100, sedangkan item Sangat Kurang Baik adalah 1 x 220 = 220. Jadi, jika total skor penilaian responden yang diperoleh adalah 935, maka penilaian interpretasi responden terhadap pemahaman belajar bahasa Jepang melalui *game* simulasi 3D adalah hasil nilai yang dihasilkan dengan menggunakan rumusindeks persentase (%).

1. Menghitung indeks persentase (%)

Indeks persentase digunakan untuk menentukan seberapa baik hasil penilaian responden. Rumus untuk menghitung indeks persentase adalah:

Index % = () x 100

1. Menentukan *interval* kriteria skor

*Interval* digunakan untuk mengategorikan hasil skor ke dalam kriteria tertentu. Rumus *interval* dihitung dengan:

I = ()

I = () = 20

Hasil (I) = 20 (ini adalah intervalnya jarak dari terendah 0% hingga 100%)

Berikut kriteria interpretasi skor berdasarkan *interval*:

Angka 0% - 19,99% = Sangat Buruk

Angka 20% - 39,99% = Buruk

Angka 40% - 59,99% = Cukup

Angka 60% - 79,99% = Baik

Angka 80% - 100% = Sangat Baik

1. Menginterpretasikan hasil

Index % = () x 100

Index % = () x 100 = 85%

**Penyelesaian Akhir (%) = 85% (SANGAT BAIK).**

#### Kesimpulan Kuesioner *User Acceptance Testing*

Berdasarkan hasil kuesioner yang telah dilakukan terhadap 22 mahasiswa yang telah mengisi kuesioner tersebut, maka dapat disimpulkan bahwa IMPLEMENTASI GAME ANDROID 3D SIMULASI DALAM PEMBELAJARAN BAHASA JEPANG PADA MAHASISWA UNIVERSITAS PAMULANG DENGAN UNITY ENGINE MENGGUNAKAN METODE MULTIMEDIA DEVELOPMENT LIFE CYCLE (MDLC) mendapatkan nilai 85% (Sangat Baik) dari skala 100% (Sangat Baik) dari para mahasiswa. Sebagian besar responden menilai *game* ini mendapat penilaian tinggi dalam aspek grafis, animasi, kontrol, dan antarmuka mahasiswa, serta kualitas *audio* yang sangat baik. Dalam sisi edukatif, *game* ini efektif dalam membantu mahasiswa mempelajari kosakata, tata bahasa, membaca, dan menulis bahasa Jepang dengan penilaian rata-rata baik hingga sangat baik. Banyak responden merasa termotivasi untuk belajar lebih lanjut setelah bermain, menunjukkan minat belajar yang tinggi. Secara keseluruhan, *game* ini dianggap layak dan bermanfaat untuk membantu pelajar dalam mempelajari bahasa Jepang melalui pendekatan permainan.

## Uji Normalitas

Uji normalitas adalah metode yang digunakan untuk menentukan apakah distribusi data mengikuti pola distribusi normal atau tidak. Terdapat banyak metode yang dapat digunakan untuk pengujian normalitas data, dan setiap metode dapat menghasilkan keputusan yang berbeda (Sintia dkk., 2022).

Dalam penelitian ini, terdapat 22 responden mahasiswa, sehingga uji normalitas dilakukan menggunakan uji *Shapiro-Wilk*. Jika *p-value* yang dihasilkan dari uji *Shapiro-Wilk* lebih besar dari tingkat signifikansi yang telah ditentukan yaitu (0,05), maka hipotesis nol () diterima, yang berarti data berdistribusi normal. Namun, jika *p-value* lebih kecil dari tingkat signifikansi, hipotesis nol () ditolak, yang menunjukkan bahwa data tidak berdistribusi normal. Kriteria untuk pengambilan keputusan dalam uji ini adalah sebagai berikut:

1. Jika p > 0,05 (), menunjukkan bahwa tidak ada bukti yang cukup untuk menolak hipotesis nol, sehingga data dianggap berdistribusi normal.
2. Jika p ≤ 0,05 (), menunjukkan bahwa ada bukti yang cukup untuk menolak hipotesis nol, sehingga data dianggap tidak berdistribusi normal.

Berikut adalah uji normalitas terhadap hasil kuesioner yang dilakukan menggunakan uji *Shapiro-Wilk*, sebagai berikut:

Tabel 4. 11 data responden dan jawaban skala likert

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Responden** | **Q1** | **Q2** | **Q3** | **Q4** | **Q5** | **Q6** | **Q7** | **Q8** | **Q9** | **Q10** |
| Milda | 4 | 4 | 4 | 5 | 4 | 3 | 4 | 5 | 4 | 2 |
| Muhammad Irgi Al Ghithraf | 4 | 4 | 5 | 4 | 3 | 3 | 4 | 4 | 4 | 5 |
| Aldo Hermawan Suryana | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Ikhsan Rasyid | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Andre Farhan Saputra | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 5 | 5 | 5 |
| Sulthan Tangguh Saefullah | 4 | 5 | 4 | 5 | 4 | 4 | 4 | 3 | 4 | 3 |
| Tharmidzi Hermawan | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 |
| Muhammad Fahri | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| Devi Nur Suryanita | 5 | 5 | 4 | 4 | 4 | 4 | 5 | 4 | 4 | 4 |
| Aurora Rachel | 3 | 3 | 4 | 4 | 3 | 3 | 4 | 4 | 4 | 4 |
| Yaasmiin Nuhaa Asa Putri | 4 | 4 | 5 | 5 | 5 | 4 | 4 | 4 | 5 | 4 |
| Salhan Taris Agusti | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| Azriel FachrulRezy | 4 | 4 | 4 | 3 | 3 | 3 | 3 | 4 | 4 | 3 |
| Ibnu Hajar | 4 | 4 | 4 | 5 | 5 | 4 | 4 | 4 | 4 | 5 |
| Rangga Ariansyah | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 3 | 4 | 4 |
| Ikmalul ilmi | 5 | 4 | 5 | 4 | 5 | 5 | 5 | 5 | 4 | 5 |
| Yogi Rizky Pangestu | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| Rafli Febrian Qasthalani | 5 | 4 | 4 | 5 | 5 | 5 | 4 | 3 | 4 | 5 |
| Rio Antono | 5 | 5 | 5 | 4 | 5 | 4 | 4 | 4 | 5 | 5 |
| Bani Maskur Muhammad Al-Walad | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| Muhammad Irfannurroja | 5 | 4 | 5 | 5 | 4 | 5 | 5 | 4 | 5 | 4 |
| Mochamad Aminnur | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 3 | 4 | 4 |

Tabel 4. 12 hasil uji normalitas pertanyaan

|  |  |  |  |
| --- | --- | --- | --- |
| ***Tests of Normality*** | | | |
|  | *Shapiro-Wilk* | | |
| *Statistic* | *df* | *Sig.* |
| Q1 | 0,768 | 22 | 0,000 |
| Q2 | 0,773 | 22 | 0,000 |
| Q3 | 0,738 | 22 | 0,000 |
| Q4 | 0,756 | 22 | 0,000 |
| Q5 | 0,778 | 22 | 0,000 |
| Q6 | 0,841 | 22 | 0,002 |
| Q7 | 0,767 | 22 | 0,000 |
| Q8 | 0,847 | 22 | 0,003 |
| Q9 | 0,696 | 22 | 0,000 |
| Q10 | 0,784 | 22 | 0,000 |

Pada tabel 4.12 menampilkan data responden berserta jawaban dalam skala *likert* setiap pertanyaan dan hasil uji normalitas sementara menggunakan uji *Shapiro-Wilk* untuk 22 responden dari 10 pertanyaan (Q1 hingga Q10).

Tabel 4. 13 hasil statistik deskriptif

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Descriptive Statistics** | | | | | |
|  | N | Minimum | Maximum | Mean | Std. Deviation |
| Q1 | 22 | 3,00 | 5,00 | 4,3636 | 0,65795 |
| Q2 | 22 | 3,00 | 5,00 | 4,2727 | 0,63109 |
| Q3 | 22 | 3,00 | 5,00 | 4,4091 | 0,59033 |
| Q4 | 22 | 3,00 | 5,00 | 4,4091 | 0,66613 |
| Q5 | 22 | 3,00 | 5,00 | 4,2273 | 0,81251 |
| Q6 | 22 | 2,00 | 5,00 | 4,0909 | 0,86790 |
| Q7 | 22 | 3,00 | 5,00 | 4,2273 | 0,61193 |
| Q8 | 22 | 2,00 | 5,00 | 4,0455 | 0,84387 |
| Q9 | 22 | 2,00 | 5,00 | 4,2727 | 0,70250 |
| Q10 | 22 | 2,00 | 5,00 | 4,1818 | 0,95799 |
| Total\_Skor | 22 | 26,00 | 50,00 | 42,5000 | 6,00595 |

Pada tabel 4.13 menampilkan hasil analisis statistik deskriptif, di mana rata-rata penilaian responden untuk semua pertanyaan berkisar antara 4,0455 hingga 4,4091, yang mengindikasikan evaluasi yang umumnya positif. Nilai standar deviasi yang rendah menunjukkan tingkat konsistensi yang tinggi dalam jawaban responden, dengan variasi yang relatif kecil. Skor total rata-rata sebesar 42,50 dari 50,00 juga memperkuat kesimpulan bahwa penilaian keseluruhan sangat positif. Secara keseluruhan, data ini menunjukkan bahwa responden memiliki pandangan yang seragam dan baik terhadap objek yang dinilai.

Tabel 4. 14 hasil akhir uji normalitas responden

|  |  |  |  |
| --- | --- | --- | --- |
| ***Tests of Normality*** | | | |
|  | *Shapiro-Wilk* | | |
| *Statistic* | *df* | *Sig.* |
| Total\_Skor | 0,922 | 22 | 0,084 |

Dengan menjumlahkan semua variabel pertanyaan yang di mana perhitungan setiap responden dari 10 pertanyaan (Q1 hingga Q10), maka tabel 4.14 menampilkan hasil akhir uji *Shapiro-Wilk* pada variabel “Total\_Skor” menunjukkan nilai statistik sebesar 0,922 dengan *p-value* sebesar 0,084. Karena *p-value* ini lebih besar dari ambang batas 0,05, hipotesis nol () yang menyatakan bahwa data berdistribusi diterima. Hal ini berarti bahwa distribusi data dapat dianggap normal, sehingga memenuhi asumsi normalitas yang diperlukan.

# BAB V PENUTUP



## Kesimpulan

Dari hasil implementasi *game Android* 3D Simulasi dalam pembelajaran bahasa Jepang “JepangCita: Game Simulasi 3D” dengan metode *Multimedia Development Life Cycle (MDLC)* diperoleh kesimpulan sebagai berikut:

1. *Game* JepangCita telah berhasil mempermudah mahasiswa Universitas Pamulang untuk mempelajari bahasa Jepang. Hasil kuesioner menunjukkan bahwa rata-rata penilaian berada pada kategori baik hingga sangat baik dalam hal kemudahan mempelajari kosakata, tata bahasa, serta peningkatan keterampilan membaca dan menulis bahasa Jepang.
2. *Game* ini juga berhasil dalam meningkatkan motivasi mahasiswa Universitas Pamulang untuk tetap konsisten dalam mempelajari bahasa Jepang. Hasil kuesioner menunjukkan bahwa mayoritas responden merasa lebih termotivasi setelah memainkan *game* ini.

## Saran

Dalam upaya meningkatkan kualitas *game* pembelajaran bahasa Jepang ini, penulis menyarankan beberapa penelitian selanjutnya sebagai berikut:

1. Mengembangkan fitur-fitur baru yang lebih interaktif dan edukatif, seperti penambahan level *scene* dan modul pembelajaran yang lebih mendalam.
2. Mempertimbangkan integrasi *game* dengan teknologi lain seperti *VR (Virtual Reality)* untuk memberikan pengalaman belajar yang lebih imersif.
3. Untuk meningkatkan interaksi dan kolaborasi antar mahasiswa, pengembangan fitur *online multiplayer* dapat menjadi langkah yang efektif. Dengan adanya fitur ini, mahasiswadapat belajar bersama teman atau berkompetisi dengan pemain lain secara *online*.

# DAFTAR PUSTAKA

Etsuko, H., & Sachiko, M. (2012). *Minna no Nihongo 2nd Edition Shokyu 1* (2nd ed., Vol. 1). 3A Network.

Etsuko, H., & Sachiko, M. (2013). *Minna no Nihongo Shokyuu 1 Second Edition - Indonesian Edition* (2nd ed., Vol. 1). 3A Network.

Raharjo, B. (2022). *Pemrograman Bahasa C#* (J. Teguh Santoso, Ed.; Edisi Pertama). Yayasan Prima Agus Teknik bekerja sama dengan Universitas Sains & Teknologi Komputer (Universitas STEKOM).

Shiang, T. T. (2018). *Kiat Sukses Mudah & Praktis Mencapai N5 Edisi Baru Metode Gakushudo* (R. Trisno, Ed.). Gakushudo.

Agus Muhyidin, M., Sulhan, M. A., & Sevtiana, A. (2020). *Perancangan UI/UX Aplikasi My CIC Layanan Informasi Akademik Mahasiswa Menggunakan Aplikasi Figma*. *Jurnal Digit*, *10*(2), 208–219. https://my.cic.ac.id/.

Alvendri, D., Huda, Y., Darni, R., Negeri Padang, U., Hamka, J., Tawar Bar, A., Padang Utara, K., Padang, K., & Barat, S. (2023). *Perancangan Media Pembelajaran Interaktif Konsep Dasar Seluler Menggunakan Aplikasi Unity Berbasis Android*. *Journal on Education*, *05*(04).

Arianti, T., Fa’izi, A., Adam, S., Wulandari, M., & Aisyiyah Pontianak, P. ’. (2022). *Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram UML (Unified Modelling Language)*. *Jurnal Ilmiah Komputer Terapan dan Informasi*, *1*(1), 19–25.

Azis, T. N. (2019). *Strategi Pembelajaran Era Digital*. *Annual Conference on Islamic Education and Social Sains*, *1*(2), 308–318.

Damayanti, Ghufroni An’ars, M., & Kurniawan, A. (2022). *Sistem Informasi Manajemen Berbasis Key Performance Indicator (KPI) dalam Mengukur Kinerja Guru*. *JDMSI*, *3*(1), 8–18.

Dewi, N. K., Harira Irawan, B., Fitry, E., Putra, A. S., & Jakarta, S. M. (2021). *Konsep Aplikasi E-Dakwah untuk Generasi Milenial Jakarta*. *Jurnal Ikra-Ith Informatika*, *5*, 26–33.

Faqih, M. (2020). *Efektivitas Penggunaan Media Pembelajaran Mobile Learning Berbasis Android*. *Konfiks: Jurnal Bahasa, Sastra dan Pengajaran*, *7*(2), 27–34. https://doi.org/10.26618/jk.v7i2.4556

Feby Prasetya, A., & Lestari Dewi Putri, U. (2022). *Perancangan Aplikasi Rental Mobil Menggunakan Diagram UML (Unified Modelling Language)*. *Jurnal Ilmiah Komputer Terapan dan Informasi*, *1*(1), 14–18.

Fitriani, Y. (2021). *Pemanfaatan Media Sosial sebagai Media Penyajian Konten Edukasi atau Pembelajaran Digital*. *Journal of Information System, Applied, Management, Accounting and Research*, *5*(4), 1006–1013. https://doi.org/10.52362/jisamar.v5i4.609

Gamma Ramadhan, R., & Surahman, A. (2023). *Media Pembelajaran Aksara Jepang Berbasis Android untuk Siswa SMA Kelas X*. *Jurnal Informatika Dan Rekayasa Perangkat Lunak (JATIKA)*, *4*, 246–252. https://doi.org/10.33365/jatika.v4i2.2602

Gusti Ayu Putu Harry Saptarini, N., Akbar Hidayat, R., Putu Indah Ciptayani, dan, Teknik Elektro, J., Negeri Bali, P., Bukit Jimbaran, K., Selatan, K., & -Bali, B. (2019). *Ajarincode: Aplikasi Pembelajaran Bahasa Pemrograman Berbasis Web*. *JUST TI*, *10*, 21–23.

Herlinawati, N., Yuliani, Y., Faizah, S., Gata, W., Komputer STMIK Nusa Mandiri Jl Damai No, I., Jati Barat, W., & Selatan, J. (2020). *Analisis Sentimen Zoom Cloud Meetings di Play Store Menggunakan Naïve Bayes dan Support Vector Machine*. *CESS (Journal of Computer Engineering System and Science)*, *5*(2), 293–298.

Husin, N. (2021). *Rancang Bangun Game Berbasis Android Bertemakan Cerita Rakyat Betawi Si Pitung*. *Jurnal Esensi Infokom*, *5*, 31–36.

Jafar Adrian, Q., & Apriyanti. (2019). *Game Edukasi Pembelajaran Matematika untuk Anak SD Kelas 1 dan 2 Berbasis Android*. *Jurnal Teknoinfo*, *13*(1), 51–54.

Kumala, A., & Winardi, S. (2020). *Aplikasi Pencatatan Perbaikan Kendaraan Bermotor Berbasis Android*. *Jurnal Intra Tech*, *4*, 112–120.

Kurniawan, Y. I., Paramesvari, D. P., & Purnomo, W. H. (2021). *Game Edukasi Pengenalan Hewan Berdasarkan Habitatnya untuk Siswa Sekolah Dasar*. *Jurnal Penelitian Inovatif*, *1*(1), 57–66. https://doi.org/10.54082/jupin.6

Lempas, F. R., RUA Sompie, S., Sugiarso, B. A., Elektro Universitas Sam Ratulangi Manado, T., & Kampus Bahu-Unsrat manado, J. (2019). *Rancang Bangun Game Action 3D Pengenalan Sejarah Perang Tombulu Melawan Spanyol*. *Jurnal Teknik Informatika*, *14*(4), 435–446.

Mekel, W. J., Sompie, S. R. U. A., & Sugiarso, B. A. (2019). *Rancang Bangun Game 3D Pertahanan Kerajaan Bowontehu*. *Jurnal Teknik Informatika*, *14*, 455–464.

Mustika Ilmiani, A., Fuadi Rahman, N., & Rahmah, Y. (2020). *Multimedia Interaktif untuk Mengatasi Problematika Pembelajaran Bahasa Arab*. *Jurnal Ilmiah Program Studi Pendidikan Bahasa Arab IAIN Palangka Raya*, *8*(1), 17–32.

Nur Amalia, R., Setia Dianingati, R., & Annisaa, E. (2022). *Pengaruh Jumlah Responden Terhadap Hasil Uji Validitas Dan Reliabilitas Kuesioner Pengetahuan Dan Perilaku Swamedikasi*. *Generics : Journal of Research in Pharmacy Accepted : 4 Mei*, *2*(1), 9–15.

Nurcholis, R., Purnamasari, A. I., Dikananda, A. R., Nurdiawan, O., & Anwar, S. (2021). *Game Edukasi Pengenalan Huruf Hiragana Untuk Meningkatkan Kemampuan Berbahasa Jepang*. *Building of Informatics, Technology and Science (BITS)*, *3*(3), 338–345. https://doi.org/10.47065/bits.v3i3.1091

Pradana Putra, A., Andriyanto, F., Dewi Muji Harti, T., & Puspitasari, W. (2020). *Pengujian Aplikasi Point of Sale Berbasis Web Menggunakan Black Box Testing*. *Jurnal Bina Komputer*, *2*, 74–78.

Ramadhona, E. W., Prasetya, T., Purnamasari, A. I., Rinaldi Dikananda, A., & Nurdiawan, O. (2021). *Game Edukasi “Nihongo Kurabu” Belajar Bahasa Menggunakan Unity 2D Berbasis Android*. *Information Management for Educators and Professionals*, *6*(1), 71–80.

S. Kalengkongan, F., A. Sugiarso, B., & D. E. Paturusi, S. (2023). *Japanese Interactive Learning for Student Based on Augmented Reality*. *Jurnal Teknik Elektro dan Komputer*, *12*, 17–24.

Saidi Rahman, M. (2019). *Aplikasi Rekapitulasi Kuesioner Hasil Proses Belajar Mengajar Pada STMIK Indonesia Banjarmasin Menggunakan Java*. *Technologia*, *10*(3), 165–171.

Sari, A. (2021). *Perkembangan Bahasa Anak Usia Dini 5-6 Tahun Ditinjau dari Aspek Sintaksis dan Pragmatik*. *Jurnal Kualita Pendidikan*, *2*(2), 2774–2156.

Sehang, J. D., Tulenan, V., & Sambul, A. M. (2019). Perancangan Game Simulasi Kewirausahaan. *Jurnal Teknik Informatika*, *14*(1), 79–88.

Setiyani, L. (2021). *Desain Sistem : Use Case Diagram*. *LPPM STMIK ROSMA / Prosiding Seminar Nasional : Inovasi & Adopsi Teknologi*, 246–260.

Sintia, I., Danil Pasarella, M., & Andi Nohe, D. (2022). *Perbandingan Tingkat Konsistensi Uji Distribusi Normalitas pada Kasus Tingkat Pengangguran di Jawa*. *Prosiding Seminar Nasional Matematika, Statistika, dan Aplikasinya*, 322–333.

Syarif, M., & Nugraha, W. (2020). *Pemodelan Diagram UML Sistem Pembayaran Tunai pada Transaksi E-Commerce*. *Jurnal Teknik Informatika Kaputama (JTIK)*, *4*(1), 64–70.

Tjahyadi, S., & Antonio, W. (2022). *Perancangan dan Implementasi Media Pembelajaran di SMA Kartini Batam Menggunakan Metode MDLC*. *Prosiding National Conference for Community Service Project (NaCosPro)*, *4*, 1107–1112. http://journal.uib.ac.id/index.php/nacospro

Wahyudi, I., Fadilah, J. N., & Nugroho, F. (2022). *Perancangan Game Pair Matching untuk Pengenalan Huruf Hijaiyah Menggunakan Unity Game Engine*. *Walisongo Journal of Information Technology*, *4*(2), 139–146. https://doi.org/10.21580/wjit.2022.4.2.7102

Wira, D., Putra, T., & Andriani, R. (2019). *Unified Modelling Language (UML) dalam Perancangan Sistem Informasi Permohonan Pembayaran Restitusi SPPD*. *Jurnal TEKNOIF*, *7*(1), 32–39.

Zahir, M., Sabiq Al Mubaraq, M., Misbahuddin, M., & Dipa Makassar, U. (2022). *Perancangan Game Simulasi Rakit Komputer Menggunakan Unity*. *Jurnal Teknik Komputer AMIK BSI*, *8*, 168–173. https://doi.org/10.31294/jtk.v4i2

Ziveria, M., Sefina Samosir, R., & Rusli, M. (2020). *Pelatihan Desain Grafis Menggunakan Perangkat Adobe Photoshop Untuk Manipulasi Foto Bagi Tim Teknologi Informasi YPU*. *ABDIMAS Jurnal Pengabdian Kepada Masyarakat*, *1*, 1–11.

# LAMPIRAN

Lampiran 1 kartu konsultasi mahasiswa



Lampiran 2 hasil kuesioner







Diagram jawaban Formulir. Judul pertanyaan: Apakah Anda telah menggunakan aplikasi atau game pembelajaran bahasa sebelumnya?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Bagaimana Anda menilai grafis dan animasi dalam game ini?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Bagaimana Anda menilai kontrol dan antarmuka pengguna (GUI) dalam game ini?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Bagaimana kualitas audio, termasuk musik dan efek suara, dalam game ini?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Seberapa efektif game ini dalam membantu Anda belajar kosakata bahasa Jepang?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Seberapa efektif game ini dalam membantu Anda memahami tata bahasa Jepang?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Seberapa efektif game ini dalam meningkatkan kemampuan berbicara bahasa Jepang Anda?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Seberapa efektif game ini dalam meningkatkan kemampuan membaca dan menulis bahasa Jepang Anda?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Apakah Anda merasa termotivasi untuk belajar bahasa Jepang lebih lanjut setelah bermain game ini?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Seberapa puas Anda dengan konten edukatif yang disediakan dalam game ini?
. Jumlah jawaban: 22 jawaban.

Diagram jawaban Formulir. Judul pertanyaan: Seberapa baik pengalaman Anda dalam berbahasa Jepang setelah bermain game ini?
. Jumlah jawaban: 22 jawaban.

Lampiran 3 data tabel kuesioner



Lampiran 4 game JepangCita dimainkan oleh mahasiswa





