

**PENGEMBANGAN GAME SIMULASI PERAWATAN
TANAMAN BERBASIS ANDROID DENGAN METODE
FINITE STATE MACHINE DAN ALGORITMA TIME-BASED
GROWTH**

SKRIPSI

DISUSUN OLEH

M RAYYAN AL FARIZ PRASETYA

NPM. 2209020174



UMSU

Unggul | Cerdas | Terpercaya

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA**

MEDAN

2026

**PENGEMBANGAN GAME SIMULASI PERAWATAN
TANAMAN BERBASIS ANDROID DENGAN METODE
FINITE STATE MACHINE DAN ALGORITMA TIME-BASED
GROWTH**

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer
(S.Kom) dalam Program Studi Teknologi Informasi, pada Fakultas Ilmu Komputer
dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara**

M RAYYAN AL FARIZ PRASETYA

NPM. 2209020174

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
MEDAN
2026**

LEMBAR PENGESAHAN

Judul Skripsi : PENGEMBANGAN GAME SIMULASI PERAWATAN
TANAMAN BERBASIS ANDROID DENGAN
METODE FINITE STATE MACHINE DAN
ALGORITMA TIME-BASED GROWTH

Nama Mahasiswa : M RAYYAN AL FARIZ PRASETYA

NPM : 2209020174

Program Studi : TEKNOLOGI INFORMASI

Menyetujui
Komisi Pembimbing



(Dr. Al-Khowarizmi, S.Kom., M.Kom.)

NIDN. 0116079201

Ketua Program Studi



(Fatma Sari Hutagalung, S.Kom., M.Kom.)

NIDN. 0117019301

Dekan



(Dr. Al-Khowarizmi, S.Kom., M.Kom.)

NIDN. 0127099201

PERNYATAAN ORISINALITAS

PENGEMBANGAN GAME SIMULASI PERAWATAN TANAMAN BERBASIS ANDROID DENGAN METODE FINITE STATE MACHINE DAN ALGORITMA TIME-BASED GROWTH

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, 05 Maret 2026

Yang membuat pernyataan



M Rayyan Al Fariz Prasetya

NPM. 2209020174

**PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama : M Rayyan Al Fariz Prasetya
NPM : 2209020174
Program Studi : Teknologi Informasi
Karya Ilmiah : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bedas Royalti Non-Eksekutif (*Non-Exclusive Royalty free Right*) atas penelitian skripsi saya yang berjudul:

**PENGEMBANGAN GAME SIMULASI PERAWATAN TANAMAN
BERBASIS ANDROID DENGAN METODE FINITE STATE MACHINE
DAN ALGORITMA TIME-BASED GROWTH**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, 05 Maret 2026

Yang membuat pernyataan



M Rayyan Al Fariz Prasetya

NPM. 2209020174

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap : M Rayyan Al Fariz Prasetya
Tempat dan Tanggal Lahir : Medan, 03 April 2004
Alamat Rumah : JL KLY Sudarso Tanjung Mulia GGMadio
Telepon/Faks/HP : -
E-mail : mrayyanalfarizprasetya@gmail.com
Instansi Tempat Kerja : -
Alamat Kantor : -

DATA PENDIDIKAN

SD : SDS LAKS. MARTADINATA MEDAN TAMAT: 2016
SMP : SMP NEGERI 11 MEDAN TAMAT: 2019
SMA : SMA NEGERI 3 MEDAN TAMAT: 2022

KATA PENGANTAR



Puji syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, taufik, dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “PENGEMBANGAN GAME SIMULASI PERAWATAN TANAMAN BERBASIS ANDROID DENGAN METODE FINITE STATE MACHINE DAN ALGORITMA TIME-BASED GROWTH”. Skripsi ini disusun sebagai salah satu syarat untuk menyelesaikan program Sarjana Strata satu (S1) Program Studi Teknologi Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi.

Penulis tentunya berterima kasih kepada berbagai pihak dalam dukungan serta doa dalam penyelesaian skripsi. Penulis juga mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. Agussani, M.AP., Rektor Universitas Muhammadiyah Sumatera Utara (UMSU)
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. Dekan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
3. Ibu Fatma Sari Hutagalung, S.Kom., M.Kom. Ketua Program Studi Teknologi Informasi .
4. Bapak Okvi Nugroho, S.Kom., M.Kom. Sekretaris Program Studi Teknologi Informasi.
5. Ibu Dr. Firahmi Rizky, S,Kom., M.Kom Selaku Wakil Dekan 1 Fakultas Ilmu Komputer Dan Teknologi Informasi.

6. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. selaku pembimbing skripsi yang telah memberikan arahan, saran, dan bimbingan dengan penuh kesabaran dan ketulusan.
7. Ayahanda dan Ibunda yang telah memberikan dukungan kepada penulis dari segi moral maupun materil serta menjadi motivasi penulis untuk dapat menyelesaikan penulisan skripsi ini.
8. Seluruh Bapak/Ibu Dosen Fakultas Ilmu Komputer & Teknologi Informasi Universitas Muhammdiyah Sumatera Utara atas ilmu dan didikannya selama perkuliahan.
9. Sahabat dan teman-teman seperjuangan, serta semua pihak yang tidak dapat penulis sebutkan satu per satu yang telah membantu dalam penyelesaian skripsi ini.

Penulis menyadari bahwa dalam penulisan skripsi ini masih terdapat banyak kekurangan. Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan demi perbaikan di masa yang akan datang. Akhir kata, penulis berharap skripsi ini dapat bermanfaat bagi semua pihak yang berkepentingan.

Medan, 05 Maret 2026



M Rayyan Al Fariz Prasetya
NPM. 2209020174

PENGEMBANGAN GAME SIMULASI PERAWATAN TANAMAN BERBASIS ANDROID DENGAN METODE FINITE STATE MACHINE DAN ALGORITMA TIME-BASED GROWTH

ABSTRAK

Perkembangan teknologi digital membuka peluang baru dalam penyampaian materi edukasi secara interaktif, salah satunya pada topik siklus pertumbuhan tanaman yang sering kali sulit dipahami secara teoritis melalui metode pembelajaran tradisional. Penelitian ini bertujuan untuk mengembangkan game simulasi perawatan tanaman berbasis Android menggunakan metode Finite State Machine (FSM) dan algoritma Time-Based Growth yang dapat digunakan sebagai alternatif media hiburan sekaligus edukasi. FSM digunakan untuk mengelola transisi antar fase pertumbuhan tanaman secara terstruktur, sedangkan algoritma Time-Based Growth digunakan untuk mengatur durasi pertumbuhan setiap fase berdasarkan waktu serta interaksi dalam permainan, seperti penyiraman dan pemberian pupuk. Metode penelitian yang juga digunakan dalam pengembangan game simulasi ini adalah Multimedia Development Life Cycle (MDLC) yang terdiri dari tahap concept, design, material collecting, assembly, testing, dan distribution. Hasil implementasi menunjukkan bahwa sistem mampu merepresentasikan siklus pertumbuhan tanaman secara dinamis, melalui mekanisme pengaruh Kesehatan (HP) tanaman, kadar air, fase delay kematian, serta percepatan waktu melalui pupuk. Berdasarkan hasil pengujian, aplikasi berjalan sesuai dengan perancangan dan mampu memberikan pengalaman pembelajaran yang interaktif dan menghibur.

Kata Kunci: Game Edukasi, Simulasi, Finite State Machine, Time-Based Growth, Android.

DEVELOPMENT OF AN ANDROID-BASED PLANT CARE SIMULATION GAME USING THE FINITE STATE MACHINE METHOD AND TIME-BASED GROWTH ALGORITHM

ABSTRACT

The advancement of digital technology has created new opportunities for delivering educational content in an interactive manner, particularly on the topic of plant growth cycles, which are often difficult to understand theoretically through traditional learning methods. This study aims to develop an Android-based plant care simulation game utilizing the Finite State Machine (FSM) method and the Time-Based Growth algorithm, which can serve as an alternative medium for both entertainment and education. The FSM method is implemented to manage structured transitions between plant growth phases, while the Time-Based Growth algorithm regulates the duration of each growth stage based on time and in-game interactions, such as watering and fertilizing. The development process follows the Multimedia Development Life Cycle (MDLC), which consists of the stages of concept, design, material collecting, assembly, testing, and distribution. The implementation results indicate that the system is capable of dynamically representing the plant growth cycle through mechanisms involving plant health (HP), water levels, a delay-death phase, and growth acceleration through fertilizer usage. Based on testing results, the application functions according to the design specifications and provides an interactive and engaging learning experience.

Keywords: Educational Game, Simulation, Finite State Machine, Time-Based Growth, Android.

DAFTAR ISI

LEMBAR PENGESAHAN	ii
PERNYATAAN ORISINALITAS.....	iii
KATA PENGANTAR.....	vi
ABSTRAK	viii
ABSTRACT	ix
DAFTAR ISI.....	x
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah	3
1.4. Tujuan Penelitian	4
1.5. Manfaat Penelitian	4
BAB II LANDASAN TEORI	5
2.1. Game.....	5
2.2. Siklus Pertumbuhan Tanaman	7
2.3. Karakteristik Tanaman dan Perawatannya.....	9
2.3.1 Tomat (<i>Solanum Lycopersicum</i>).....	9
2.3.2 Terong (<i>Solanum Melongena L.</i>).....	10
2.3.3 Timun (<i>Cucumis Sativus L.</i>).....	10
2.4. Software Unity 3D	11
2.5. Metode Pengembangan.....	12
2.6. Finite State Machine (FSM).....	13
2.7. Algoritma Time-Based Growth	14
2.8. Ringkasan Penelitian Terdahulu	15
2.9. Analisis GAP	22
BAB III METODOLOGI PENELITIAN	24
3.1. Metode Penelitian	24
3.2. Penelitian dan Pengumpulan Informasi (Research and Information Collection)	25
3.3. Perencanaan (Planning).....	26
3.4. Pengembangan Bentuk Awal Produk (Developing Preliminary Form Of Product).....	27
3.4.1 Konsep (Concept)	28
3.4.2 Desain (Design).....	28
3.4.3 Pengumpulan Material (Material Collecting)	36
3.4.4 Penggabungan (Assembly)	37
3.4.5 Pengujian (Testing).....	39

3.4.6	Distribusi (Distribution).....	40
3.5.	Pengujian Awal (Preliminary Testing)	40
3.6.	Revisi Produk Awal (Preliminary Product Revision)	41
BAB IV	HASIL DAN PEMBAHASAN	42
4.1.	Implementasi Desain Sistem.....	42
4.1.1	Tampilan Menu Utama (Main Menu).....	42
4.1.2	Tampilan Tutorial	45
4.1.3	Tampilan Gameplay.....	46
4.1.3.1	Tampilan Awal <i>Gameplay</i>	46
4.1.3.2	Tampilan Panel Pengaturan	47
4.1.3.3	Tampilan Panel Penyimpanan (Warehouse/Inventory)	48
4.1.3.4	Tampilan Menu Toko (Shop)	49
4.1.3.5	Tampilan Timer, Status Bar HP & Air Tanaman	50
4.1.3.6	Tampilan Panel Info Fase Tanaman	52
4.1.3.7	Tampilan Hama Pada State Idle.....	53
4.1.3.8	Tampilan Minigame Battle Dengan Hama	54
4.1.3.9	Tampilan Ending Permainan	56
4.1.4	Uji Coba Interaksi Tombol	57
4.1.5	Uji Coba FSM Pada Model Tanaman	59
4.1.6	Uji Coba FSM Pada Model Hama	62
4.1.7	Uji Coba Algoritma Time-Based Growth Pada Timer.....	64
4.1.7.1	Pengaruh Status Bar HP dan Air Pada Timer	65
4.1.7.2	Pengaruh Fitur Pupuk Terhadap Percepatan Durasi Timer	66
BAB V	PENUTUP.....	67
5.1.	Kesimpulan	67
5.2.	Saran	68
DAFTAR PUSTAKA		69

DAFTAR TABEL

Tabel 2. 1 Ringkasan Penelitian Terdahulu	16
Tabel 4. 1 Informassi Tombol Pada Scene Menu Utama.....	42
Tabel 4. 2 Tabel Uji Fungsi Tombol.....	57
Tabel 4. 3 Tabel Uji Perubahan State Tanaman.....	61
Tabel 4. 4 Tabel Uji Perubahan State Hama	64
Tabel 4. 5 Pengaruh Status HP dan Air Pada Timer	65

DAFTAR GAMBAR

Gambar 3. 1 Metode Penelitian Kombinasi R&D dan MDLC	25
Gambar 3. 2 Wireframe Tampilan Menu Utama	29
Gambar 3. 3 Wireframe Tampilan Tutorial	30
Gambar 3. 4 Wireframe Tampilan Gameplay	31
Gambar 3. 5 Wireframe Tampilan Mini-Game Battle Hama	32
Gambar 3. 6 Wireframe Tampilan Warehouse Menu	33
Gambar 3. 7 Wireframe Tampilan Shop Menu.....	33
Gambar 3. 8 Diagram Finite State Machine Tanaman.....	34
Gambar 3. 9 Diagram Finite State Machine Hama	35
Gambar 3. 10 Flowchart Alur Pertumbuhan Tanaman Dengan Algoritma Time-Based Growth.....	38
Gambar 4. 1 Tampilan <i>Scene</i> Menu Utama (New Game)	43
Gambar 4. 2 Tampilan <i>Scene</i> Menu Utama (Save Data)	44
Gambar 4. 3 Tampilan <i>Scene</i> Tutorial	45
Gambar 4. 4 Tampilan Awal Gameplay	46
Gambar 4. 5 Tampilan Panel Pengaturan.....	47
Gambar 4. 6 Tampilan Panel Penyimpanan (Warehouse/Inventory).....	48
Gambar 4. 7 Tampilan Menu Toko.....	49
Gambar 4. 8 Tampilan <i>Timer</i> , Status HP & Air Saat Permainan Dimulai	50
Gambar 4. 9 Tampilan <i>Timer</i> Saat Memasuki Fase “Delay Death”	51
Gambar 4. 10 Tampilan Panel Info Fase Tanaman.....	52
Gambar 4. 11 Tampilan State Idle Hama.....	53
Gambar 4. 12 Tampilan Minigame Battle Hama	54
Gambar 4. 13 Tampilan <i>Scene</i> Victory dan Defeat.....	55
Gambar 4. 14 Tampilan <i>Scene</i> Ending Permainan.....	56
Gambar 4. 15 Unity Console: Perubahan State Model Tanaman	59
Gambar 4. 16 Unity Console: Perubahan State Ke Model Dead	60
Gambar 4. 17 Model State Tanaman Tomat	60
Gambar 4. 18 Unity Console: Spawn Hama Lalat (Idle).....	62
Gambar 4. 19 Unity Console: Hama Memasuki State Menyerang.....	62
Gambar 4. 20 Unity Console: Hama Kalah dan Kabur.....	63
Gambar 4. 21 Unity Console: Hama Menang dan Kembali Berkeliling (Idle)	63

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Perkembangan teknologi digital dan juga game membuka peluang baru untuk menyajikan materi edukasi dengan cara yang lebih interaktif dan menarik. Salah satunya adalah topik dalam ilmu biologi dasar yang sering kali sulit dipahami secara konseptual oleh para pelajar, topik itu ialah siklus pertumbuhan tanaman. Menurut Purba et al., (2020), permainan simulasi dapat menjadi media edukasi yang menggabungkan visualisasi proses biologis dengan mekanisme interaksi *real-time*, sehingga memungkinkan pemain untuk belajar melalui pengalaman dan dan pemahaman visual tanpa perlu fasilitas laboratorium.

Informasi mengenai siklus hidup tumbuhan biasanya disampaikan dalam bentuk tulisan atau melalui demonstrasi yang terbatas, baik secara waktu maupun sumber daya. Hal ini menyebabkan kurangnya pengalaman untuk memahami dinamika pertumbuhan dan efek dari tindakan kita terhadap tanaman. Penelitian mengenai pengembangan metode pengajaran berbasis permainan untuk konsep pertumbuhan, menunjukkan perlunya media interaktif yang dapat menyediakan latihan berulang serta pengamatan terhadap perubahan yang terjadi secara bertahap (Purba et al., 2020).

Metode pengajaran tradisional mengenai pertumbuhan tanaman sering kali kurang terlihat menarik. Sejumlah penelitian mengindikasikan bahwa digitalisasi dalam pembelajaran dapat meningkatkan semangat dan pemahaman konsep dari

materi yang disampaikan, asalkan permainan tersebut menawarkan mekanisme yang mendukung eksplorasi serta memberikan *feedback* secara langsung. Agar permainan dapat digunakan sebagai alat pendidikan, permainan harus menunjukkan dinamika perubahan fase tanaman dan dampak dari tindakan pemain. Penerapan model state FSM (*Finite State Machine*) memungkinkan penggambaran tahapan pertumbuhan dan transisi keadaan dengan cara yang terorganisir. Selain itu, penerapan algoritma *Time-Based Growth* juga menjadi hal yang krusial untuk menggambarkan durasi fase dan dampak dari intervensi (Hitti et al., 2021; Jagdale, 2021)

Beberapa studi yang berkaitan dengan pengembangan *game* edukatif bertema serupa menunjukkan peningkatan motivasi, pemahaman konsep, serta penggunaan berlanjut dalam pembelajaran para pelajar (Sari et al., 2015). Penelitian ini diharapkan dapat menghasilkan prototipe yang tidak hanya memenuhi standar teknis menggunakan Unity dan FSM, tetapi juga efektif dalam hal pembelajaran untuk meningkatkan pemahaman dan motivasi belajar. Dengan demikian, pengembangan dan pengujian permainan ini relevan sebagai langkah untuk mengatasi masalah dalam pembelajaran tentang pertumbuhan tanaman yang telah diidentifikasi.

1.2. Rumusan Masalah

Berdasarkan penjelasan dari latar belakang, maka dapat dirumuskan permasalahan pada penelitian ini adalah sebagai berikut:

1. Bagaimana rancangan game simulasi perawatan tanaman berbasis Android dengan menggunakan Unity?
2. Bagaimana metode *Finite State Machine* (FSM) di-implementasikan ke dalam tahapan pertumbuhan tanaman?
3. Bagaimana menerapkan algoritma *time-based growth* untuk menentukan durasi pertumbuhan tanaman dan memungkinkan perubahan waktu berdasarkan interaksi pemain?

1.3. Batasan Masalah

Penelitian ini memiliki batasan permasalahan yang dibahas agar dapat menyelesaikan permasalahan utama, yaitu:

1. Platform game dibatasi hanya untuk Android, sehingga implementasi dan pengujian game hanya dilakukan pada perangkat mobile berbasis Android.
2. Tanaman yang ada hanya dibatasi pada beberapa jenis tanaman saja.
3. Game tidak menggunakan sistem *stage* untuk mengatur kesulitan, tetapi menggunakan *State*
4. Software yang digunakan dalam pengembangan game ini adalah game engine Unity.

1.4. Tujuan Penelitian

Berdasarkan rumusan masalah di atas, tujuan dari penelitian ini, ialah:

1. Mengembangkan game simulasi perawatan tanaman berbasis Android yang mampu memberikan pengalaman belajar interaktif mengenai proses pertumbuhan tanaman.
2. Menerapkan metode Finite State Machine (FSM) untuk mengatur status dan transisi tahap pertumbuhan tanaman secara sistematis dan mudah dipahami.
3. Menerapkan algoritma *time-based growth* untuk mensimulasikan waktu pertumbuhan tanaman.

1.5. Manfaat Penelitian

Manfaat yang diharapkan dapat diperoleh dari penelitian ini adalah sebagai berikut:

1. Penelitian ini dapat menjadi referensi akademik bagi penelitian di masa mendatang yang membahas pemodelan FSM, Time-Based Growth dan simulasi proses biologis dalam game.
2. Penelitian ini dapat menghasilkan game pembelajaran edukatif mengenai perawatan tanaman, serta dapat menjadi media pembelajaran interaktif.

BAB II

LANDASAN TEORI

2.1. Game

Game secara umum dapat diartikan sebagai suatu kegiatan interaktif yang memberikan pengalaman bagi pemain dengan memadukan berbagai elemen seperti tantangan, aturan, interaksi, serta *feedback* yang membuatnya menarik dan menghibur bagi setiap orang dari berbagai kalangan usia. Menurut Arjoranta, (2019), *game* menekankan bahwa penting untuk memahami karakteristik permainan digital melalui aspek-aspek interaktif yang dibangun dalam desain sistemnya, sehingga *game* tidak hanya merujuk pada hiburan semata.

Dalam arti lain, *game* memiliki tujuan yang bervariasi, mulai dari hiburan, mengasah keterampilan, sarana pembelajaran dan lain lain. Menurut Azizah & Salehudin, (2023), *game* yang dirancang secara interaktif dapat meningkatkan motivasi dan pemahaman belajar dengan memanfaatkan struktur permainan yang menarik dan menyenangkan.

Game juga dikategorikan dalam berbagai genre yang memiliki karakteristik masing-masing dan bagaimana *game* tersebut berinteraksi dengan pengguna.

1. Action

Genre *Action* merupakan jenis genre yang menekankan keterampilan refleks dan koordinasi antara mata dengan tangan pemain. Pemain biasanya dituntut untuk memiliki respon cepat terhadap situasi yang berubah secara dalam permainan. Salah satu genre *game* yang mengasah skill.

2. Adventure

Genre petualangan/*Adventure* berfokus pada eksplorasi lingkungan dan dunia yang ada di dalam game dengan menjelajah dan menyelesaikan tantangan. Game jenis ini sering kali menonjolkan unsur narasi dan interaksi dengan objek atau *NPC* di dalam dunia permainan.

3. Role-Play

Role-Playing games (RPG) adalah genre yang memberi pemain kebebasan untuk berperan sebagai karakter dalam suatu skenario. Pemain biasanya diminta untuk membuat keputusan sekaligus meningkatkan kemampuan karakter, serta mengikuti alur cerita yang berkembang.

4. Simulation

Genre *Simulation* dibuat dengan meniru kehidupan nyata atau situasi tertentu. Genre ini menciptakan dunia virtual yang memberikan pemain akses untuk berinteraksi dengan model sistem secara *real-time*. *Game* Simulasi mampu membangun motivasi serta meningkatkan pemahaman pengguna melalui interaksi yang menyenangkan dari pengalaman bermain di dalam *game* (Jasmin et al., 2025).

5. Strategy

Game dengan genre ini menekankan aspek perencanaan, taktik, dan pengambilan keputusan. Di dalam *game* strategi pemain harus mempertimbangkan sumber daya, gerakan, dan konsekuensi untuk mencapai tujuan permainan.

6. Puzzle

Genre *puzzle* dirancang untuk menantang pemain dalam memecahkan teka-teki atau tantangan. Jenis game ini menguji kreativitas dan kemampuan berpikir kritis dalam menyelesaikan masalah.

Pada pengembangan *game* dalam penelitian ini, penulis akan berfokus pada genre simulasi atau *Simulation Genres*.

2.2. Siklus Pertumbuhan Tanaman

Siklus pertumbuhan tanaman merupakan serangkaian tahapan biologis yang dilalui oleh tanaman mulai dari biji hingga mencapai fase berbuah. Berdasarkan penjelasan Taiz & Zeiger (1946), pertumbuhan tanaman adalah suatu proses dinamis dan dipengaruhi oleh faktor-faktor seperti genetika, lingkungan, ketersediaan air, nutrisi, pencahayaan, dan juga interaksi fisiologis di dalamnya.

Secara umum, tahapan siklus pertumbuhan tanaman terdiri dari beberapa fase utama, yaitu:

1. Fase Benih (*Seed Stage*)

Di fase ini, tanaman masih berada dalam keadaan benih/biji yang mengandung embrio dan cadangan makanan. Proses aktivasi fisiologis benih dimulai saat benih tersebut menyerap air.

2. Fase Perkecambahan (*Germination Stage*)

Perkecambahan terjadi ketika embrio mulai melakukan aktivitas metabolisme dan akar pertama mulai tumbuh. Faktor-faktor seperti kelembapan, cahaya, dan nutrisi mempengaruhi keberhasilan fase ini. Sejumlah penelitian

mengindikasikan bahwa tahap ini merupakan momen penting dalam pertumbuhan tanaman yang sering dianalisis dalam konteks pendidikan karena perubahan yang terlihat dengan jelas (Nonogaki, 2014).

3. Fase Tanaman Muda (*Vegetative Growth*).

Fase perubahan Vegetatif ditandai dengan daun, batang, dan akar yang mengalami pertumbuhan signifikan. Pada fase ini tanaman memerlukan cahaya matahari dan nutrisi yang cukup untuk mendukung proses fotosintesis serta pembentukan massa (*biomassa*) (Hidayah et al., 2022). Kurangnya air dapat berakibat serius bagi tanaman pada tahap ini, menjadikan fase ini sangat penting bagi tanaman agar dapat memasuki tahap selanjutnya.

4. Fase Berbunga (*Flowering Stage*)

Pada tahap ini, tanaman mengalami perubahan fisiologis untuk memulai fase reproduksi. Produk fotosintesis yang dihasilkan selama fase vegetatif diarahkan untuk pembentukan bunga. Faktor pencahayaan dan hormon seperti florigen memiliki peran penting dalam proses ini.

5. Fase Berbuah (*Fruiting/Reproductive Stage*)

Setelah proses penyerbukan dan fertilisasi selesai, tanaman mulai memasuki fase pembuatan buah untuk menghasilkan biji baru. Kebutuhan nutrisi menjadi sangat krusial karena tanaman memerlukan lebih banyak energi untuk mendukung perkembangan buah.

6. Fase Buah Matang/Siap Panen (*Harvest Stage*)

Buah muda akan mengalami perubahan dalam warna, tekstur, rasa, dan komposisi kimianya saat mencapai fase pematangan. Di akhir fase ini, buah telah siap untuk di Buah Matang/Siap Panen.

2.3. Karakteristik Tanaman dan Perawatannya

Tanaman hortikultura memiliki banyak jenis, diantaranya adalah tanaman dengan jenis sayuran (olerikultura), buah-buahan (frutikultura), tanaman hias (florikultura), dan tanaman obat (biofarmaka), masing-masing jenis tanaman juga memiliki teknik perawatan yang berbeda-beda. Pada penelitian ini, jenis tanaman yang akan digunakan sebagai objek penelitian adalah tanaman bertipe buah-buahan, sering juga dikategorikan sebagai sayuran namun masih termasuk jenis buah-buahan juga, tanaman-tanaman itu ialah: tomat, terong dan timun.

2.3.1 Tomat (*Solanum Lycopersicum*)

Tanaman tomat merupakan sayuran buah yang memiliki struktur batang yang tegak, daunnya bertangkai, serta buah berwarna merah yang akan tumbuh ketika memasuki fase panen. Hal yang perlu diperhatikan dalam merawat tomat yaitu kebutuhan air yang konsisten dan nutrisi seimbang untuk mendukung pembentukan bunga dan buah secara maksimal. Penelitian oleh Puspitasari et al., (2023) menyatakan bahwa pemberian pupuk dan zat pengatur tumbuh sangat berpengaruh terhadap pertumbuhan dan hasil tanaman tomat, hal ini terlihat dari peningkatan tinggi tanaman dan bobot buah saat dosis pupuk diatur secara benar.

Perawatan tomat yang baik dan benar dapat dilakukan dengan penyiraman yang teratur, terutama saat fase vegetatif dan akan memasuki proses pembentukan bunga dan buah, serta nutrisi yang cukup melalui pemupukan rutin. Penyiraman yang tepat akan menjaga kelembapan tanah pada batas optimal, tidak terlalu basah maupun terlalu kering, sehingga akar dapat menyerap air dan unsur hara dengan efisien.

2.3.2 Terong (*Solanum Melongena L.*)

Tanaman terong memiliki ciri batang yang tegak dan daun berukuran sedang, terong menghasilkan buah yang bervariasi baik dari bentuk maupun ukuran. Dalam merawat terong dibutuhkan asupan sinar matahari yang cukup, penyiraman yang teratur, penyiangan untuk mengurangi gulma, serta pemupukan berkala sesuai kebutuhan tanaman. Selain itu, pengaturan jarak tanam dan pemilihan media tanam yang cocok juga berpengaruh terhadap pertumbuhan seperti tinggi tanaman, jumlah daun, dan hasil buah.

2.3.3 Timun (*Cucumis Sativus L.*)

Tanaman timun memiliki batang merambat yang menyerupai sulur, daun timun yang relatif lebar, dan buah yang mengandung banyak air. Timun memiliki kebutuhan air yang lebih tinggi dibanding beberapa tanaman lain, terutama pada fase awal pertumbuhan hingga pembentukan buah. Untuk merawat timun dengan benar, timun perlu disiram secara teratur agar tanah tidak mengalami kekeringan yang dapat memicu pada terhambatnya pertumbuhan sulur. Selain itu, pemberian

pupuk secara berkala membantu persediaan unsur hara yang dibutuhkan tanaman untuk memproduksi buah. Beberapa penelitian menemukan bahwa variasi pupuk tertentu dapat memberikan efek pertumbuhan yang lebih baik pada timun, seperti peningkatan panjang dan bobot buah dibandingkan perlakuan tanpa pupuk yang sesuai (Savitri et al., 2025).

2.4. Software Unity 3D

Unity 3D adalah salah satu game engine multi-platform yang sering dipakai dalam pembuatan game 2D dan 3D karena kemudahan penggunaan, fleksibilitas, serta API yang lengkap. Menurut Goldstone, (2011). Unity menawarkan sistem berbasis komponen yang memungkinkan pengembang untuk menambahkan fungsionalitas pada objek melalui *script*, sehingga membuat proses pengembangan game lebih mudah dikelola. Unity juga mendukung berbagai platform seperti Android, iOS, Windows, dan WebGL.

Selain itu, Unity dilengkapi dengan sistem untuk rendering, animasi, *physique machine*, dan dukungan integrasi pustaka dari pihak ketiga yang luas. Kuznetsov et al., (2022) mengungkapkan bahwa banyak institusi pendidikan memanfaatkan Unity karena dapat mempercepat proses pembuatan prototipe, menyediakan antarmuka visual yang mudah dipahami, serta mendukung pembelajaran berbasis pengalaman. Kemampuan Unity dalam mengintegrasikan C# sebagai bahasa pemrograman utama juga membantu pengembang untuk menerapkan logika permainan seperti pengendalian karakter, sistem interaksi, dan pengelolaan status objek (Bustare et al., 2024).

Dalam konteks pengembangan game simulasi merawat tanaman, Unity memungkinkan implementasi *Finite State Machine* guna mengelola kondisi pertumbuhan tanaman. Unity juga ideal digunakan dalam proyek simulasi karena mendukung algoritma *Time-Based Growth* yang menggunakan waktu untuk melakukan pembaruan status tanaman secara langsung.

2.5. Metode Pengembangan

Penelitian ini menggunakan 2 kombinasi metode, yaitu metode Multimedia Development Life Cycle (MDLC) dan juga metode Research & Development (R&D) untuk mendapatkan hasil yang lebih baik. Berikut dijabarkan penjelasannya.

2.5.1 Research & Development (R&D)

Metode Research and Development merupakan metode penelitian yang bertujuan untuk merancang dan menghasilkan suatu produk serta menguji tingkat efektivitas-nya. Metode ini banyak digunakan dalam penelitian pengembangan sistem karena tidak hanya berfokus pada analisis permasalahan, tetapi juga menghasilkan solusi dalam bentuk produk yang dapat digunakan oleh pengguna.

Dalam penelitian ini, metode R&D digunakan sebagai kerangka penelitian dalam proses pengembangan game simulasi perawatan tanaman berbasis Android. Melalui metode ini, penelitian dilakukan secara mulai dari tahap pengumpulan informasi awal, perencanaan, pengembangan produk, hingga tahap pengujian dan perbaikan sistem yang dikembangkan.

Secara umum, metode R&D terdiri dari sepuluh tahapan yang dimulai dari proses pengumpulan informasi awal (*research and preliminary information collecting*) sampai tahap implementasi produk (*dissemination and implementation*).

2.5.2 Multimedia Development Life Cycle (MDLC)

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah Multimedia Development Life Cycle (MDLC). MDLC merupakan metode pengembangan perangkat lunak yang dirancang khusus untuk pembangunan aplikasi multimedia seperti media pembelajaran interaktif, animasi, maupun permainan digital.

Metode MDLC digunakan karena pengembangan game simulasi perawatan tanaman melibatkan berbagai elemen multimedia seperti gambar, animasi, antarmuka pengguna, serta interaksi antara pemain dengan sistem permainan. Dengan menggunakan metode MDLC, proses pengembangan aplikasi dapat dilakukan secara sistematis dan terstruktur. Metode MDLC terdiri dari enam tahapan utama yaitu concept, design, material collecting, assembly, testing, dan distribution.

2.6. Finite State Machine (FSM)

Finite State Machine (FSM) adalah sebuah model komputasi yang menggambarkan sistem melalui beberapa *state* yang berbeda dan peralihan antar *state* yang dipicu oleh input atau kondisi tertentu. FSM adalah metode yang populer dalam pembuatan game karena membantu pengembang dalam mengelola perilaku

objek sesuai dengan tahap atau kondisi tertentu, seperti: keadaan musuh, animasi karakter, atau kondisi lingkungan. FSM beroperasi berdasarkan prinsip bahwa sebuah objek hanya bisa berada dalam satu *state* pada suatu waktu dan akan berpindah ke *state* yang lain ketika terkena suatu event atau logika program (Rabin, 2009).

FSM juga menawarkan struktur logis yang teratur, sehingga jalur transisi mudah untuk diawasi, dan mengurangi kemungkinan munculnya perilaku yang tidak diinginkan. Dengan setiap *state* memiliki fungsi dan logika yang terpisah, pengembang dapat memperbarui atau mengubah perilaku sistem tanpa merusak keseluruhan struktur permainan. Dalam penelitian ini, FSM diterapkan untuk menggambarkan tahapan-tahapan perkembangan tanaman, seperti biji, tunas, tanaman muda, berbunga, berbuah, Buah Matang/Siap Panen, serta *state* “Mati” jika pemain lalai dalam merawat tanaman. Model ini memungkinkan permainan untuk mensimulasikan siklus hidup tanaman secara teratur dan sesuai dengan teori pertumbuhan tanaman.

2.7. Algoritma Time-Based Growth

Algoritma *Time-Based Growth* merupakan salah satu logika dalam pengembangan *game* yang mengatur perubahan status objek berdasarkan waktu. Dalam konteks permainan simulasi merawat tanaman, logika ini memungkinkan terjadinya tahap-tahap perkembangan tanaman setelah periode waktu tertentu yang telah ditentukan sebelumnya, serta memberikan kesempatan bagi pemain untuk

melakukan intervensi seperti menyiram atau memberikan pupuk guna mempercepat proses pertumbuhan.

Amacker et al., (2022) menciptakan sebuah simulator yang meniru pertumbuhan tanaman di dalam ruangan menggunakan kurva pertumbuhan dan machine learning. Simulator ini memanfaatkan prediksi untuk pertumbuhan tanaman berdasarkan faktor lingkungan seperti kualitas udara, pencahayaan, dan ketersediaan air, serta mengatur kecepatan perkembangan tanaman sesuai dengan waktu dan kondisi yang ada. Algoritma *Time-Based Growth* menjadi bagian krusial dari penelitian dan pengembangan game ini, karena algoritma ini mampu untuk membuat pemain merasakan jalannya pertumbuhan tanaman di setiap fase.

2.8. Ringkasan Penelitian Terdahulu

Penelitian yang dilakukan diperlukan untuk memberikan perspektif mengenai karya-karya sebelumnya yang berkaitan dengan tema ini. Terdapat beberapa kajian sebelumnya yang membahas tentang pengembangan, studi kasus, desain permainan, serta penerapan beberapa metode dalam permainan yang memiliki tema sama. Analisis ini berfungsi sebagai landasan untuk mengevaluasi posisi dan kontribusi dari penelitian yang sedang dilakukan.

Tabel 2. 1 Ringkasan Penelitian Terdahulu

NO	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
1	<p>Creation of a meaningful gardening game: Increasing players' nature awareness through soil management and plant recognition mechanics (Summerley et al., 2020)</p>	<p>Pengembangan prototipe game berkebun yang mereplikasi siklus kehidupan tanaman dalam grid; dengan memanfaatkan model simulasi automata sel, pengintegrasian API untuk identifikasi tanaman (Pl@ntNet), serta pelaksanaan uji coba permainan guna menilai pengaruh terhadap kesadaran. Penting untuk menggambarkan</p>	<p>Research-creation; prototyping di Unity; playtest (user study) & analisis kualitatif.</p>	<p>Kelebihan: Desain simulasi yang realistis, menunjukkan dampak mekanisme pemeliharaan dan dampak dari air/nutrisi terhadap ekosistem. Kekurangan: Fokus pada kesadaran akan pentingnya alam bukan pada edukasi</p>

NO	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
		siklus tanaman dan cara perawatannya.		tentang biologi tanaman.
2	<p>GAME EDUKASI PENGENALAN PERTUMBUHAN HEWAN DAN TUMBUHAN PADA ANAK TUNAGRAHITA RINGAN DI SLB YPASP GONDANGREJO (Assidiq, 2020)</p>	<p>Pengembangan <i>game</i> edukatif untuk memperkenalkan pertumbuhan fauna dan flora di tingkat sekolah dasar; mencakup pembahasan tentang perancangan pembelajaran, alur permainan, serta pengujian pada anak didik dengan kebutuhan khusus.</p>	<p>Pengembangan aplikasi (Unity), uji coba pengguna (observasi/kuis ioner).</p>	<p>Kelebihan: Teruji pada target pengguna yaitu Sekolah Dasar, fokus pedagogis kuat (validasi pembelajaran), UI/UX yang nyaman untuk anak-anak dengan kebutuhan khusus</p> <p>Kekurangan: Kecenderungan konten pada tingkat</p>

NO	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
				SD, tidak mencakup rincian tentang siklus biologis yang lebih kompleks.
3	Gamifikasi untuk Meningkatkan Kesadaran Lingkungan pada Anak Usia Dini: Studi Kasus pada Game Simulasi Merawat Tanaman (Dewi et al., 2023)	Artikel ini berisi studi kasus yang mengkaji pengembangan simulasi perawatan tanaman sebagai bentuk gamifikasi untuk meningkatkan kesadaran lingkungan di kalangan anak-anak dan membahas desain mekanik perawatan serta hasil pengujiannya.	Studi kasus: perancangan game, implementasi prototipe, dan evaluasi (kuesioner, observasi, dan analisis hasil).	Kelebihan: Menyajikan hasil penilaian dan saran untuk rancangan mekanis perawatan tanaman yang bermanfaat untuk perancangan <i>UX/flow</i> . Kekurangan: Artikel jurnal (bukan skripsi)

NO	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
				tetapi keterkaitan tema cukup kuat; untuk skripsi komparatif.
4	<p>PENGEMBANGAN GAME PERTANIAN MENGGUNAKAN FINITE STATE MACHINE DAN FUZZY LOGIC (WA'DAN, 2023)</p>	<p>Penelitian ini merancang sebuah permainan yang berfokus pada pertanian dengan menggunakan Finite State Machine (FSM) untuk mengelola keadaan perilaku tanaman dan NPC, Fuzzy Logic juga digunakan sebagai alat untuk menentukan kondisi tanaman seperti</p>	<p>FSM dan Fuzzy Logic</p>	<p>Kelebihan: Menggunakan dua pendekatan AI (FSM dan Logika Fuzzy) secara bersamaan, model pertumbuhan tanaman menjadi lebih akurat karena mempertimbangkan faktor-</p>

NO	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
		<p>kesuburan atau kebutuhan air. Permainan ini dibuat untuk mengilustrasikan bagaimana faktor-faktor lingkungan secara dinamis memengaruhi perkembangan tanaman.</p>		<p>faktor lingkungan. Kekurangan: Tidak menerapkan pertumbuhan yang berbasis waktu secara langsung, tingkat kompleksitas Fuzzy Logic cukup tinggi untuk skripsi awal, dan fokusnya bukan pada edukasi melainkan pada pemodelan</p>

NO	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
				dinamika dalam pertanian.
5	Rancang Bangun Game Edukasi Pengenalan Bagian-Bagian Tumbuhan untuk Siswa Sekolah Dasar Berbasis Android (Satria et al., 2022)	Penelitian ini mengembangkan sebuah aplikasi pendidikan berbasis Android untuk memperkenalkan elemen-elemen tumbuhan kepada anak-anak di sekolah dasar. Aplikasi ini menyediakan konten visual, permainan ular tangga, teka-teki, dan kuis untuk meningkatkan interaksi dalam proses belajar.	MDLC (Multimedia Development Life Cycle)	Kelebihan: Memiliki struktur pengembangan multimedia yang lengkap, berfokus pada pendidikan tanaman yang terkait dengan tema, memanfaatkan elemen interaktif seperti kuis, teka-teki. Kekurangan:

NO	Judul dan peneliti	Pembahasan	Metode	Kelebihan atau Kekurangan
				Tidak menampilkan sistematis biologis dari tumbuhan dan lebih berfokus pada <i>Mini-games</i> .

2.9. Analisis GAP

Berdasarkan analisis dari sejumlah penelitian sebelumnya, terlihat bahwa mayoritas pengembangan game dengan tema tanaman masih terfokus pada pengenalan dasar tentang jenis dan bagian tanaman. Beberapa penelitian telah menggunakan metode tertentu seperti Finite State Machine (FSM) atau logika pendukung seperti Fuzzy Logic, namun penggunaannya masih terbatas pada perilaku dasar tanaman atau pengambilan keputusan yang sederhana. Tidak ada penelitian yang khusus menggabungkan FSM sebagai pengatur urutan tahap pertumbuhan tanaman secara menyeluruh dan mengaitkannya dengan algoritma *Time-Based Growth* untuk memodelkan durasi perkembangan tanaman sesuai dengan waktu dan interaksi pengguna.

Penelitian sebelumnya juga lebih banyak ber-fokus pada aspek edukatif yang berbasis quiz atau teka-teki, alih-alih mensimulasikan siklus biologis yang dinamis. Oleh karena itu, penelitian ini hadir sebagai upaya untuk menutup beberapa kekurangan tersebut dengan mengembangkan *game* simulasi merawat tanaman menggunakan Unity yang memadukan FSM (*Finite State Machine*) untuk memodelkan tahapan pertumbuhan dan algoritma *Time-Based Growth* untuk mengatur mekanisme durasi perkembangan, sehingga dapat memberikan pengalaman belajar yang lebih realistis, interaktif, dan mendalam dibandingkan penelitian sebelumnya.

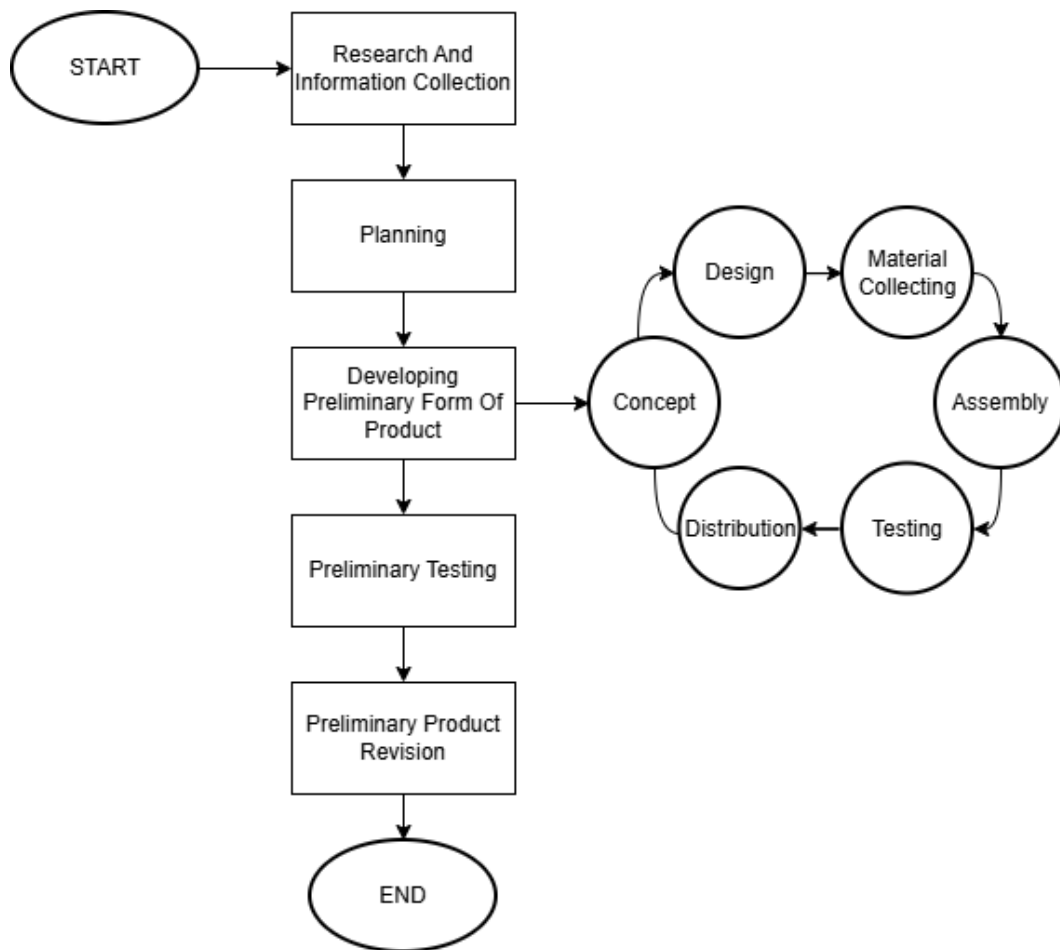
BAB III

METODOLOGI PENELITIAN

3.1. Metode Penelitian

Penelitian ini menggunakan pendekatan yang menggabungkan model Research and Development (R&D) dengan Multimedia Development Life Cycle (MDLC). Metode R&D merupakan metode penelitian yang bertujuan untuk menghasilkan suatu produk sekaligus menguji tingkat efektivitas dari produk yang dikembangkan tersebut. Mengacu pada model Borg & Gall, metode R&D memiliki sepuluh tahapan yang dimulai dari tahap research and preliminary information collecting hingga tahap Dissemination and Implementation, namun dalam penelitian ini, penerapan metode R&D hanya dilakukan dalam 5 tahap saja (Wahyuni et al., 2022).

Pada penelitian ini, model R&D dimanfaatkan sebagai kerangka dalam proses perancangan sistem, sedangkan metode MDLC digunakan untuk mendukung proses pengembangan produk awal berupa rancangan permainan. Menurut Luther, metode MDLC terdiri dari enam tahapan utama yaitu Concept, Design, Material Collecting, Assembly, Testing, Dan Distribution. Integrasi antara model R&D dan MDLC dilakukan pada tahap ketiga dari proses R&D, yaitu Developing Preliminary Form Of Product. Skema kombinasi antara model R&D dan MDLC dalam penelitian ini dapat dilihat pada Gambar



Gambar 3. 1 Metode Penelitian Kombinasi R&D dan MDLC

3.2. Penelitian dan Pengumpulan Informasi (Research and Information Collection)

Tahap Research And Preliminary Information Collecting merupakan tahap awal dalam metode Research and Development (R&D) yang memiliki tujuan untuk memperoleh informasi awal terkait permasalahan yang akan diteliti. Pada tahap ini dilakukan proses pengumpulan data dan informasi yang relevan sebagai dasar dalam pengembangan sistem yang akan dibangun.

Pengumpulan informasi dilakukan melalui beberapa teknik, antara lain studi literatur dan analisis penelitian sejenis yang berkaitan dengan pengembangan game

simulasi maupun media interaktif. Studi literatur dilakukan dengan mempelajari berbagai sumber seperti jurnal ilmiah, buku, dan penelitian terdahulu yang membahas mengenai metode pengembangan game, tahapan pertumbuhan siklus tanaman, penerapan finite state machine, serta algoritma time-based growth pada sistem simulasi permainan.

3.3. Perencanaan (Planning)

Tahapan perencanaan bertujuan untuk merumuskan tujuan penelitian serta menentukan strategi pengembangan sistem yang akan dilakukan. Pada tahap ini dilakukan perancangan konsep sistem yang terdiri dari pengembangan aplikasi, target pengguna, serta fitur-fitur utama yang akan diimplementasikan dalam game simulasi perawatan tanaman.

Selain itu, pada tahap ini juga dilakukan perencanaan terhadap mekanisme permainan yang meliputi interaksi pemain dengan objek dalam permainan seperti tanaman, sistem penyiraman, pemberian pupuk, serta penanganan hama. Tahap perencanaan ini akan menghasilkan gambaran awal mengenai struktur sistem yang akan dikembangkan serta spesifikasi kebutuhan yang diperlukan dalam proses pengembangan produk.

Pada penelitian ini, mekanisme permainan dirancang dengan mensimulasikan proses pertumbuhan tanaman secara bertahap, mulai dari fase biji hingga siap panen. Setiap tahapan pertumbuhan direpresentasikan sebagai *state* yang saling terhubung, di mana perpindahan antar *state* dipengaruhi oleh durasi waktu dan tindakan pemain.

Durasi pertumbuhan tanaman dalam game ini tidak merepresentasikan waktu biologis nyata pada tumbuhan asli, melainkan waktu simulasi yang telah disesuaikan untuk kebutuhan permainan agar tetap menarik dan tidak membosankan. Mekanisme ini memungkinkan pemain memahami bahwa setiap tanaman memiliki karakteristik pertumbuhan yang berbeda, serta tindakan kita selama perawatan yang dapat mempengaruhi pertumbuhan dari tanaman.

3.4. Pengembangan Bentuk Awal Produk (Developing Preliminary Form Of Product)

Tahap Developing Preliminary Product merupakan tahap pengembangan produk awal dalam metode R&D. Pada penelitian ini, proses pengembangan produk dilakukan dengan menggunakan metode Multimedia Development Life Cycle (MDLC).

Metode MDLC digunakan karena sesuai dengan proses pengembangan aplikasi multimedia seperti game yang melibatkan berbagai elemen visual, interaktif, dan audio. Menurut Luther, metode MDLC terdiri dari enam tahapan utama yaitu concept, design, obtaining content material, assembly, testing, dan distribution.

3.4.1 Konsep (Concept)

Tahap Konsep merupakan tahap awal dalam pengembangan aplikasi pada metode MDLC untuk menentukan dasar dari sistem yang akan dibangun. Pada tahap ini dilakukan identifikasi terhadap tujuan pembuatan aplikasi, sasaran pengguna, serta jenis aplikasi yang akan dikembangkan.

Dalam penelitian ini, aplikasi yang dikembangkan berupa game simulasi perawatan tanaman berbasis Android yang bertujuan untuk memberikan pengalaman interaktif kepada pengguna dalam merawat tanaman secara virtual. Konsep permainan dirancang agar pemain dapat melakukan berbagai aktivitas seperti menyiram tanaman, memberikan pupuk, melihat informasi mengenai tanaman, serta mengatasi gangguan hama yang muncul selama proses pertumbuhan tanaman.

Target pasar untuk pengguna aplikasi permainan ini adalah kalangan anak-anak hingga remaja yang memiliki ketertarikan dalam merawat tanaman tanpa harus melakukan aktivitas menanam langsung dengan media tanah. Permainan ini juga dapat menjadi media pembelajaran interaktif untuk membantu pemain dalam menguji kesabaran dan ketekunan dalam merawat tanaman, sekaligus belajar tentang tumbuhan.

3.4.2 Desain (Design)

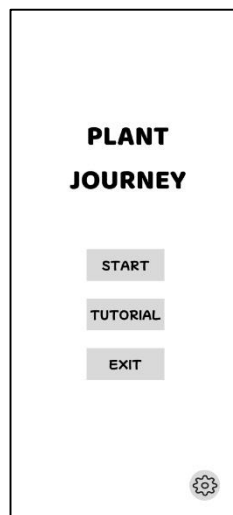
Pada tahap ini dilakukan proses perancangan struktur sistem, desain antarmuka pengguna, perancangan mekanisme logika dari sistem permainan, serta perancangan mekanisme perilaku objek permainan menggunakan pendekatan finite

state machine untuk mengatur perubahan kondisi pada objek utama yaitu tanaman dan hama.

3.4.2.1 Wireframe

Wireframe merupakan rancangan atau desain awal dari tampilan antarmuka permainan yang terdiri dari gambar dan tata letak yang menampilkan setiap tombol dan gambaran awal dari *game* yang akan dikembangkan sebelum masuk ke dalam proses pembuatan (Ramadhan & Adhisa, 2024). Desain wireframe yang telah dirancang untuk membantu pengembangan game ini mencakup lima tampilan utama, yaitu: Main Menu, Tutorial, Gameplay, Mini-game Hama, Warehouse Menu, dan Shop Menu.

1. Menu Utama

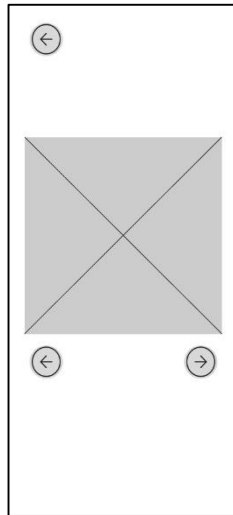


Gambar 3. 2 Wireframe Tampilan Menu Utama

Bagian ini menampilkan *interface* awal permainan saat dibuka, yang terdiri dari beberapa struktur, yaitu: judul permainan yang terletak di bagian atas dan tiga

tombol utama di bawahnya, yaitu: *Start*, *Tutorial*, Tombol *option* di pojok kanan bawah, dan *Exit* untuk keluar aplikasi.

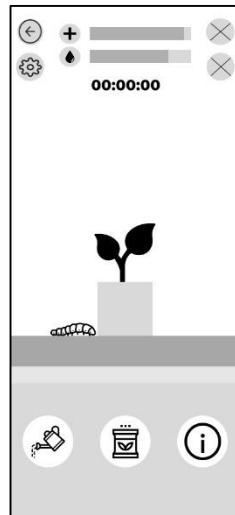
2. Tutorial



Gambar 3. 3 Wireframe Tampilan Tutorial

Halaman *Tutorial* berguna untuk membantu pemain baru dalam mengenali dasar permainan. Terdapat tombol kembali ke menu utama dibagian pojok kiri atas, dibagian tengah ada panel yang akan menampilkan informasi bagaimana cara bermain *game* ini, mulai dari fungsi-fungsi tombol hingga alur dari permainan, dan di bawah panel terdapat 2 tombol yang dapat digunakan untuk mengganti *slide tutorial* di panel.

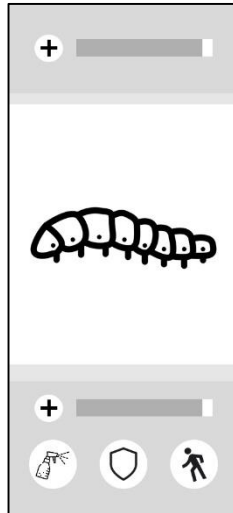
3. Gameplay



Gambar 3. 4 Wireframe Tampilan Gameplay

Tampilan gameplay memiliki fokus utama pada Area utama permainan dimana Tanaman yang berada di tengah layar sebagai objek utama. Di atas Tanaman terdapat Bar Status yang menunjukkan indikator untuk Air dan (Health) Kesehatan tanaman. Selain itu, ada *Timer* Pertumbuhan yang menginformasikan sisa waktu menuju fase selanjutnya. Di bagian bawah layar, terdapat tombol-tombol aksi, seperti: Siram, Pupuk, dan Info Tanaman, pada tombol informasi (i) berisi panel mengenai informasi tentang tanaman yang sedang ditanaman. Lalu ada juga tombol kembali ke menu utama, tombol pengaturan di pojok kiri atas, serta beberapa menu tambahan yang juga akan disertakan dalam permainan. Bagian ini adalah bagian inti dari *game*.

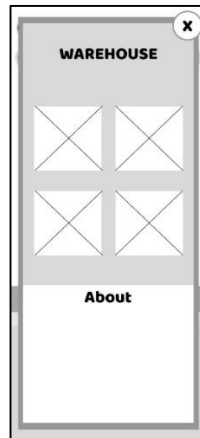
4. Mini-Game Hama



Gambar 3. 5 Wireframe Tampilan Mini-Game Battle Hama

Halaman *Mini-Game* menampilkan area pertarungan dengan objek "Hama", *Health Bar* "Hama" berada di atas layar, dan *Health bar* Pemain di bawah layar. Terdapat tombol aksi seperti *Attack*, *Defend*, dan Kabur yang dapat digunakan pemain untuk melawan "Hama". Desain ini mengikuti format permainan *Turn-Based Battle* sederhana, agar mudah dipahami oleh para pemain, dimana pemain dan hama akan memiliki giliran masing masing untuk beraksi.

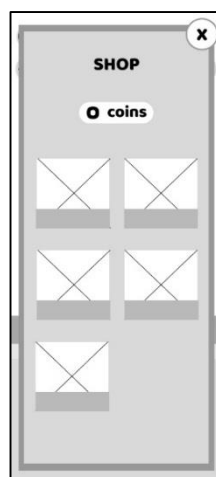
5. Warehouse Menu



Gambar 3. 6 Wireframe Tampilan Warehouse Menu

Halaman *Warehouse Menu* akan menampilkan ikon benih dari buah/sayuran apa saja yang telah pemain beli, pemain juga bisa mengetuk gambar ikon dari sayuran dan buah tersebut untuk melihat informasi lengkapnya dan menggunakannya.

6. Shop Menu

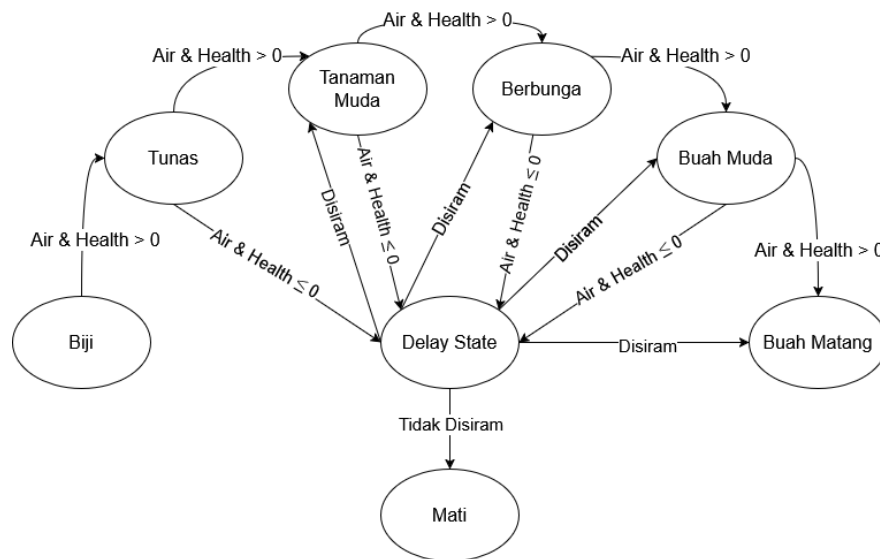


Gambar 3. 7 Wireframe Tampilan Shop Menu

Shop *Menu* akan menampilkan barang-barang apa saja yang bisa pemain beli untuk mendukung jalannya permainan. Barang-barang yang akan dimasukkan ke dalam *shop* adalah pupuk dan benih-benih tanaman.

Pada perancangan desain game ini, penulis juga menerapkan metode Finite State Machine (FSM) kedalam 2 objek, yaitu Tanaman dan Hama yang bertujuan untuk mengatur perilaku dari kedua objek tersebut selama permainan.

3.4.2.2 Diagram FSM Tanaman

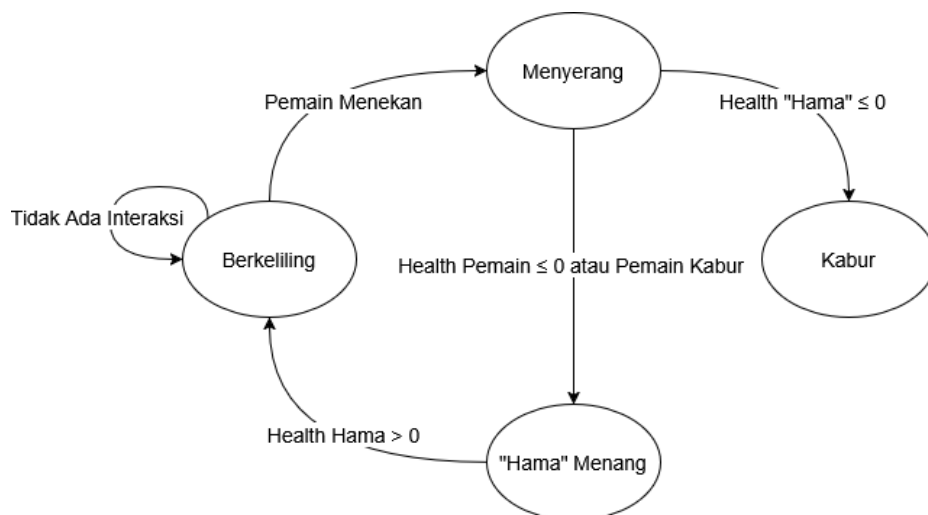


Gambar 3. 8 Diagram *Finite State Machine* Tanaman

Diagram *Finite State Machine* (FSM) Tanaman dibuat untuk mengelola transisi *state* pertumbuhan tanaman agar terorganisir ketika diaplikasikan ke dalam sistem. Perpindahan antar *state* sesuai dengan waktu yang telah ditentukan, artinya setiap keadaan/*state* memiliki periode durasi tertentu sebelum berpindah ke *state* berikutnya. Syarat untuk melakukan perpindahan *state* juga sudah tertera didalam Gambar 3.8, jika kondisi *Air & Health* tanaman > 0 serta durasi dari pertumbuhan

telah habis, maka tumbuhan memenuhi syarat untuk berpindah ke *state* selanjutnya, namun apabila *Air & Health* tanaman ≤ 0 saat durasi habis, maka tumbuhan akan memasuki *Delay Death*, fase ini terjadi ketika tumbuhan masuk pada keadaan sekarat, sistem akan memberikan kesempatan kepada pemain untuk menyelamatkan tanaman dengan cara menyiramnya. Tetapi jika pemain tidak menyiram tanaman selama *Delay Death*, maka tanaman akan memasuki *state* “Mati” dan permainan selesai. *Delay Death* dan *state* “Mati” hanya akan tersedia ketika tumbuhan telah memasuki fase “Tunas”.

3.4.2.3 Diagram FSM Hama



Gambar 3. 9 Diagram *Finite State Machine* Hama

Diagram *Finite State Machine* (FSM) Hama menggambarkan perilaku musuh yang ada di dalam *game* yang bernama Hama. Hama memiliki beberapa jenis dan bentuk yang akan bekerja sebagai pengganggu jalannya pertumbuhan dari tanaman yang sedang dirawat pemain. Hama akan muncul setelah tumbuhan memasuki *state* “Tunas” dan seterusnya. Hama menyerang dengan cara mengurangi “Health”

Tumbuhan secara perlahan, tetapi pemain juga bisa mengusir hama dengan cara mengetuk “Hama” itu sendiri. Sistem akan memulai FSM untuk “Hama” dari *state* "Berkeliling", yaitu situasi di mana Hama akan muncul secara random pada periode waktu tertentu setelah Tanaman melewati fase “Biji”.

Ketika pemain mengetuk objek “Hama” pada layar, maka sistem akan *trigger mini-game* dan Hama memasuki *state* "Menyerang", di mana proses *Turn-Based Battle* berlangsung antara pemain dan Hama. Jika Hama menang, maka ia akan kembali ke *state* “Berkeliling”, dan apabila Hama kalah, maka dia akan memasuki *state* “Kabur” dan pemain bisa melanjutkan merawat tanaman dengan tenang.

3.4.3 Pengumpulan Material (Material Collecting)

Pada tahap ini, penulis mengumpulkan material yang diperlukan dalam proses pengembangan aplikasi. Material tersebut meliputi berbagai elemen multimedia yang akan digunakan dalam game seperti gambar tanaman, ikon-ikon untuk tombol, background, animasi, serta elemen visual lainnya yang mendukung tampilan aplikasi.

Selain itu, pada tahap ini juga dilakukan persiapan berbagai komponen pendukung yang dibutuhkan dalam pengembangan sistem, termasuk asset grafis serta elemen audio yang akan digunakan dalam permainan, seperti music latar belakang, music ambience, efek suara tombol dan efek suara pendukung.

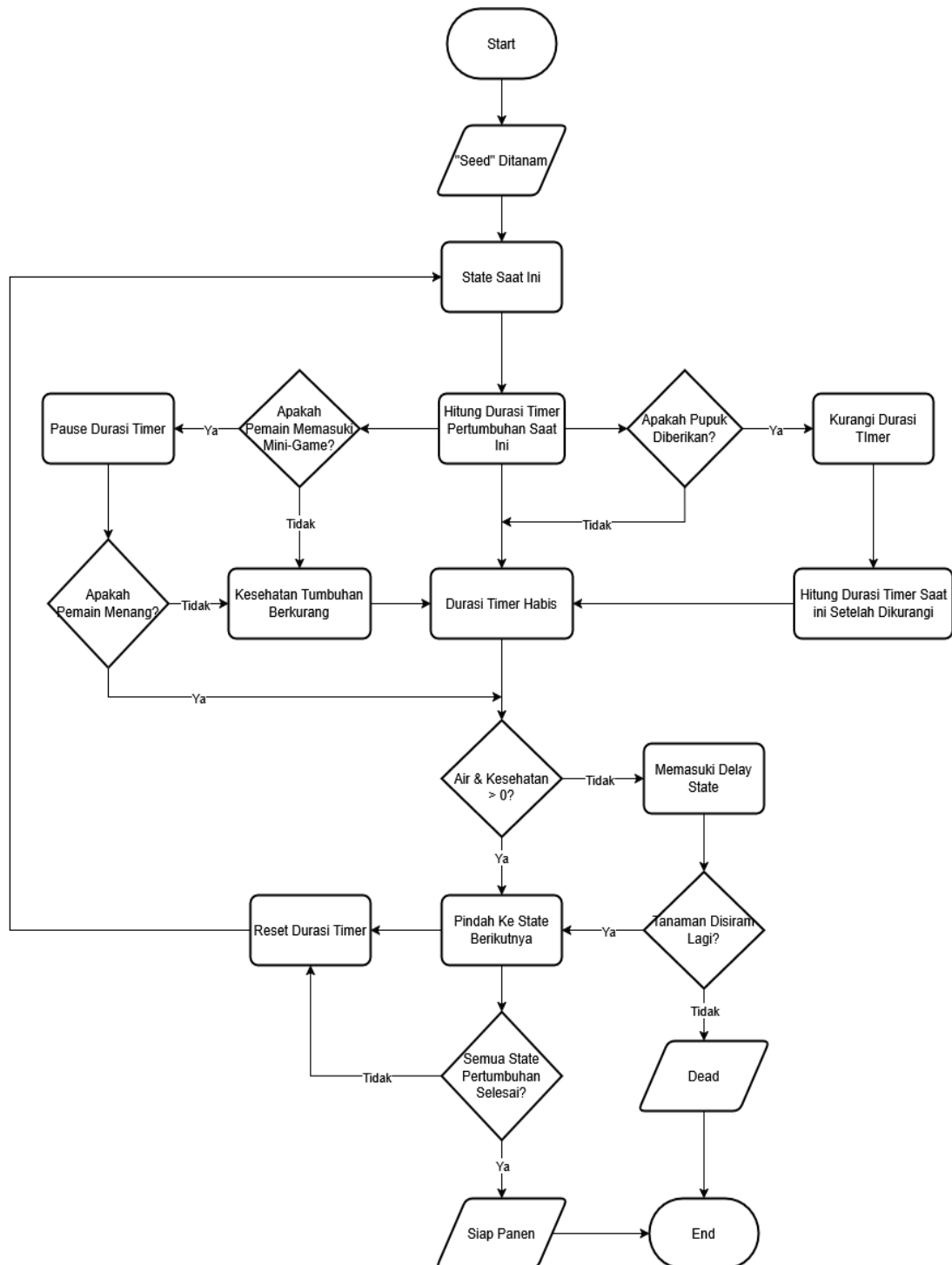
3.4.4 Penggabungan (Assembly)

Tahap assembly merupakan tahapan implementasi dimana seluruh rancangan sistem yang telah dibuat pada tahap sebelumnya disatukan menjadi aplikasi yang dapat dijalankan. Pada tahap ini dilakukan proses pengembangan sistem menggunakan perangkat lunak pengembangan game yaitu Unity.

Implementasi dilakukan dengan mengintegrasikan berbagai elemen multimedia serta logika sistem yang telah dirancang sebelumnya. Selain itu, pada tahap ini juga diterapkan algoritma time-based growth untuk mengatur proses pertumbuhan tanaman berdasarkan waktu tertentu serta metode finite state machine untuk mengelola perubahan kondisi objek dalam permainan.

Algoritma *Time-Based Growth* digunakan untuk mengelola proses perkembangan dari tanaman agar sesuai dengan lama waktu yang telah ditentukan. Setiap tanaman memiliki beberapa fase pertumbuhan yang direpresentasikan sebagai *state*, diantaranya ialah: Biji, Tunas, Tanaman Muda, Berbunga, Buah Muda, dan Buah Matang/Siap Panen. Setiap *state* memiliki durasi tertentu yang harus dilalui sebelum berpindah ke *state* selanjutnya. Cara kerja sistem ini adalah dengan menghitung waktu yang telah berlalu sejak tanaman masuk ke sebuah *state*, lalu memulai waktu baru dengan durasi yang telah ditetapkan. Sistem akan mulai bekerja saat Tanaman memasuki *state* Biji, yang mana ini adalah tahap awal dari fase tumbuhan, kemudian sistem akan mulai menghitung durasi pertumbuhan untuk dapat berubah ke tahap selanjutnya.

3.4.4.1 Perancangan Algoritma Time-Based Growth



Gambar 3. 10 Flowchart Alur Pertumbuhan Tanaman Dengan Algoritma Time-Based Growth

Pada gambar 3.10 dapat dilihat adanya variabel seperti “Hama” dan pemberian Pupuk yang dapat mengintervensi jalannya pertumbuhan tanaman. “Hama” akan menjadi salah satu penghalang utama permainan karena tanaman akan menerima damage secara bertahap dan akan menghabisi kesehatan tanaman, satu satunya cara untuk dapat menghentikan “Hama” ialah dengan melawannya pada mode *Turn-Based Battle*. Ketika hama memasuki *state* bertarung, durasi pertumbuhan tanaman akan berhenti hingga pertandingan selesai, baik *player* menang ataupun kalah.

Sementara Pupuk akan menjadi pembantu perkembangan pertumbuhan untuk mencapai tahap berikutnya, yaitu dengan mengurangi durasi pertumbuhan tanaman tergantung jenis tanaman yang ditanam dan pupuk yang digunakan. Setelah *timer* habis dan berpindah ke tahap selanjutnya, *timer* akan direset untuk menghitung ulang durasi *state* yang telah dimasuki hingga mencapai tahap Buah Matang/Siap Panen dan permainan-pun selesai. Namun apabila sistem mendeteksi jumlah kesehatan dan air pada *Health* bar tanaman ≤ 0 , tanaman akan memasuki fase delay dimana pemain dapat menyelamatkan tanaman dengan cara menyiramnya. Tetapi jika tidak, tumbuhan akan memasuki fase/*state* Mati dan permainan akan berakhir.

3.4.5 Pengujian (Testing)

Tahap testing merupakan tahap pengujian sistem yang bertujuan untuk memastikan bahwa seluruh fungsi dalam aplikasi dapat berjalan sesuai dengan rancangan yang telah dibuat. Pengujian dilakukan untuk mengidentifikasi apakah ada kesalahan atau error yang mungkin terjadi selama proses pengembangan sistem. Selain itu,

tahap ini juga bertujuan untuk mengevaluasi pengalaman pengguna dalam menggunakan aplikasi game yang dikembangkan.

3.4.6 Distribusi (Distribution)

Tahap akhir dalam metode MDLC, yaitu dimana aplikasi yang telah dikembangkan dapat didistribusikan kepada pengguna. Pada penelitian ini, aplikasi game yang telah dikembangkan dan melalui proses pengujian dapat dijalankan pada perangkat Android setelah diubah menjadi format APK sehingga dapat digunakan oleh pengguna sebagai media simulasi perawatan tanaman.

3.5. Pengujian Awal (Preliminary Testing)

Setelah tahap assembly selesai, langkah selanjutnya yang akan dilakukan ialah melakukan pengujian terhadap produk yang telah dikembangkan. Tahap ini merupakan bagian dari proses evaluasi dalam metode R&D yang bertujuan untuk mengetahui tingkat keberhasilan sistem yang telah dibuat. Pengujian dilakukan dengan menggunakan metode *BlackBox Testing* terhadap dua aspek utama dalam permainan yaitu validitas sistem terhadap metode dan algoritma yang digunakan serta keberhasilan tombol sesuai fungsinya.

3.6. Revisi Produk Awal (Preliminary Product Revision)

Tahap Preliminary Product Revision merupakan tahap perbaikan terhadap produk yang telah dikembangkan berdasarkan hasil pengujian yang dilakukan sebelumnya. Pada tahap ini dilakukan analisis terhadap berbagai masukan dan hasil evaluasi yang diperoleh selama proses pengujian sistem.

Perbaikan dilakukan terhadap aspek fungsional sistem maupun elemen multimedia yang digunakan dalam aplikasi agar produk yang dihasilkan dapat berjalan dengan lebih optimal dan memberikan pengalaman pengguna yang lebih baik.

BAB IV

HASIL DAN PEMBAHASAN




4.1. Implementasi Desain Sistem




Pada bab ini akan dipaparkan jabaran lanjutan dari tahap Assembly pada metode MDLC beserta hasil implementasi dari sistem yang telah diterapkan ke dalam prototipe game. Implementasi ini mencakup penerapan metode *Finite State Machine* (FSM) dan algoritma *Time-Based Growth* menggunakan *Unity*. Pengujian dilakukan untuk menilai apakah program yang telah dikembangkan berjalan sesuai dengan tujuan penelitian.

4.1.1 Tampilan Menu Utama (Main Menu)

Tampilan menu utama merupakan tampilan awal saat dimulainya permainan. Sistem akan menampilkan *scene* “Menu Utama” Ketika permainan dibuka. Akan ada tombol seperti: Mulai, Lanjut, Mulai Ulang Tutorial, Keluar dan tombol Pengaturan.

Tabel 4. 1 Informassi Tombol Pada *Scene* Menu Utama

Tombol	Keterangan
	Tombol “Mulai” hanya akan muncul Ketika permainan pertama kali dimainkan atau setelah memulai ulang permainan.
	Tombol “Lanjut” bisa dipakai untuk melanjutkan progress permainan yang tersimpan.
	Tombol “Mulai Ulang” berfungsi untuk me-reset save data permainan dan memulai ulang progress dari awal.

	<p>Tombol “Tutorial” dapat digunakan untuk membuka <i>scene</i> tutorial yang berisi informasi mengenai permainan.</p>
	<p>Tombol “Keluar” berfungsi untuk keluar dari permainan.</p>
	<p>Tombol “Pengaturan” dapat digunakan untuk membuka panel pengaturan yang dapat mengatur volume music latar belakang dan sound effects dalam permainan.</p>



Gambar 4. 1 Tampilan *Scene* Menu Utama (New Game)



Gambar 4. 2 Tampilan Scene Menu Utama (Save Data)

Pada gambar 4.1 menampilkan tampilan dari menu utama permainan yang berjudul *PLANT JOURNEY*. Terdapat tombol tombol yang sudah dijelaskan di dalam table 4.1, seperti tombol “Mulai” untuk memulai permainan, tombol “Mulai Ulang” untuk me-reset progress permainan, tombol “Tutorial” untuk melihat tutorial permainan, serta tombol “Keluar” dan “Pengaturan”.

Lalu pada gambar 4.2, ditampilkan tampilan dari menu utama setelah pemain memainkan permainan. Tombol “Mulai” akan berubah menjadi tombol “Lanjut” dan tombol “Mulai Ulang” akan aktif karena sudah adanya progress permainan yang tersimpan.

4.1.2 Tampilan Tutorial



Gambar 4. 3 Tampilan Scene Tutorial

Scene tutorial akan terbuka ketika pemain menekan tombol “Tutorial” pada menu utama. Setelah berpindah *scene*, pemain dapat melihat informasi-informasi yang berkaitan dengan permainan. Mulai dari fungsi tombol, *icon*, *state* tanaman, dan lainnya. *Scene* tutorial ada untuk membantu pemain dalam memainkan permainan *Plant Journey* ini, karena banyak pemain yang kesulitan dalam memainkan sebuah permainan baru tanpa adanya arahan yang jelas, sehingga terjadi kesalahpahaman dan menyebabkan kegagalan dalam bermain.

4.1.3 Tampilan Gameplay

Tampilan *gameplay* merupakan tampilan utama di dalam permainan, dimana pemain akan memulai semua perjalanan permainannya dari sini.

4.1.3.1 Tampilan Awal *Gameplay*



Gambar 4. 4 Tampilan Awal Gameplay

Tampilan awal dari gameplay dapat dilihat pada gambar 4.4. *Interface* ini adalah awal mula permainan ketika pemain memulai untuk pertama kali. Terdapat beberapa tombol yang nantinya dapat digunakan oleh pemain untuk menjalankan progress permainan. Semua penjelasan mengenai masing-masing tombol dapat dilihat pada *scene* tutorial.

4.1.3.2 Tampilan Panel Pengaturan



Gambar 4. 5 Tampilan Panel Pengaturan

Panel pengaturan dapat dibuka dengan menekan tombol pengaturan yang ber-*icon gear*. Panel ini berfungsi sebagai pengatur *volume* dari *music background* dan juga efek-efek suara di dalam permainan.

4.1.3.3 Tampilan Panel Penyimpanan (Warehouse/Inventory)



Gambar 4. 6 Tampilan Panel Penyimpanan (Warehouse/Inventory)

Panel penyimpanan dapat diakses melalui tombol “Penyimpanan” yang terletak di pojok kanan atas layar. Di panel ini pemain bisa melihat item apa yang telah didapat, contohnya seperti benih-benih tanaman pada gambar 4.6. Pemain dapat mengetuk item untuk membuka panel info mengenai item tersebut, lalu pemain bisa menekan tombol “Gunakan” untuk menggunakannya. Item hanya akan terbuka ketika sudah dibeli dari toko.

4.1.3.4 Tampilan Menu Toko (Shop)



Gambar 4. 7 Tampilan Menu Toko

Menu toko bisa dibuka dengan menekan tombol “*Shop*” yang berada dibawah tombol “Penyimpanan”. Pada menu *shop*, akan ditampilkan jumlah koin yang dimiliki pemain untuk saat ini, dengan dana awal sebesar 100 koin. Koin dapat digunakan untuk membeli item di toko, item benih tanaman yang dibeli dapat digunakan melalui panel penyimpanan, dan item pupuk bisa dibeli untuk me-*reset cooldown* dari tombol “Pupuk”.

4.1.3.5 Tampilan Timer, Status Bar HP & Air Tanaman



Gambar 4. 8 Tampilan *Timer*, Status HP & Air Saat Permainan Dimulai

Setelah menekan tombol “Gunakan”, pemain bisa menanam benih dengan cara menekan “Pot” untuk memulai fase pertumbuhan tanaman yang dipilih. *Timer* akan memulai menghitung mundur durasi *state* saat ini ke *state* selanjutnya. Durasi *timer* tergantung dari benih apa yang ditanam. Status bar HP dan Air juga akan mulai bekerja pada fase ini.

Status Bar HP dan Air berjumlah maksimal masing masing 100. Dan status air akan berkurang secara perlahan sebanyak 1 *tick* setiap detik, setelah Bar Air habis, Bar HP akan mulai berkurang setiap detiknya hingga habis. Setelah habis,

timer akan memasuki fase “*Delay Death*” Dimana *timer* akan mulai berkedip yang menandakan bahwasanya apabila tidak diberi air atau pupuk, tanaman akan mati.



Gambar 4. 9 Tampilan *Timer* Saat Memasuki Fase “*Delay Death*”

Pemain dapat menyelamatkan Tanaman dari kematian dengan cara memberi air dan juga pupuk. Bagaimana caranya? Yaitu dengan menggunakan tombol “Siram” dan “Pupuk” pada panel dibawah *layer*. Pupuk juga berguna untuk mempersingkat durasi *timer* dan menambah HP bar tanaman sebesar 50% dari HP saat ini.

4.1.3.6 Tampilan Panel Info Fase Tanaman



Gambar 4. 10 Tampilan Panel Info Fase Tanaman

Panel Info Fase Tanaman pada gambar 4.10, dapat diakses melalui tombol dengan *icon* “i” yang ada di jajaran tombol “Siram” dan “Pupuk”. Disini pemain dapat membaca informasi mengenai peristiwa apa yang terjadi pada tanaman di fase saat ini.

4.1.3.7 Tampilan Hama Pada State Idle



Gambar 4. 11 Tampilan State Idle Hama

Hama akan muncul secara *random* di *State-state* tanaman tertentu. Selama fase *idle*, Hama akan menyerap HP bar tanaman secara langsung walaupun Bar Air masih terisi. Setiap 1 hama akan menyerang sebesar 0,5 jumlah HP bar tanaman.

4.1.3.8 Tampilan Minigame Battle Dengan Hama



Gambar 4. 12 Tampilan *Minigame Battle Hama*

Pemain dapat memasuki *minigame* dengan cara menekan hama-hama yang berkeliaran di sekitaran tanaman selama permainan. Pemain bisa menyerang hama dengan cara menekan tombol “Serang”, pemain juga dapat bertahan dari serangan hama untuk mengurangi damage yang diterima dengan cara menekan tombol “Bertahan”. Selain itu, pemain dapat menggunakan tombol “Kabur” jika merasa pertandingan tidak berjalan baik.

Akan ada 2 keadaan akhir dalam pertandingan *minigame battle* ini, yaitu “*Victory* (menang)” dan “*Defeat* (kalah)”.



Gambar 4. 13 Tampilan Scene Victory dan Defeat

Keadaan “*Victory*” dapat dicapai jika pemain berhasil mengalahkan hama hingga HP bar hama mencapai 0. Pemain akan mendapatkan *reward* berupa koin yang nantinya akan dapat digunakan untuk membeli item di toko. Sedangkan jika Hama berhasil membuat HP pemain mencapai 0 terlebih dahulu, atau pemain menekan tombol “Kabur”, maka keadaan “*Defeat*” akan terpenuhi dan tanaman menerima penalti berupa damage besar terhadap HP bar Tanaman.

4.1.3.9 Tampilan Ending Permainan



Gambar 4. 14 Tampilan Scene Ending Permainan

Pada permainan ini, pemain bisa mendapatkan 2 jenis *Ending*. Yang pertama yaitu *good ending* Dimana pemain berhasil menyelesaikan semua fase pertumbuhan tanaman hingga berhasil Panen. Pemain dapat mengetuk pot untuk menyelesaikan permainan jika sudah mencapai fase ini dan *scene* "Panen!" akan ditampilkan. Pemain juga bisa memilih untuk memulai ulang semua progres permainan atau melanjutkan progress yang sudah ada beserta rewardnya.

Sebaliknya, jika pemain gagal dalam merawat tanaman hingga ia berbuah dan panen, *scene* "Gagal Panen..." akan ditampilkan. Pada *scene* ini, pemain hanya bisa memilih untuk memulai ulang dari awal. Pada kedua *ending scene* ini, akan ada Panel informasi yang bisa dibaca oleh pemain untuk motivasi.

4.1.4 Uji Coba Interaksi Tombol

Pengujian interaksi tombol dilakukan untuk memastikan bahwa setiap tombol yang terdapat pada interface permainan dapat berfungsi sesuai dengan perintah yang diberikan oleh pemain. Pada game simulasi perawatan tanaman yang dikembangkan, terdapat berbagai tombol yang dapat digunakan untuk mengakses fitur-fitur permainan seperti memulai permainan, melakukan penyiraman tanaman, memberikan pupuk, membuka menu toko, serta melakukan navigasi antar scene.

Proses pengujian dilakukan dengan cara menekan tombol-tombol yang tersedia dalam permainan dan melihat respon yang diberikan sistem. Jika tombol memberikan respon sesuai dengan fungsi yang telah dirancang, maka tombol tersebut dinyatakan valid atau berfungsi dengan baik. Sebaliknya, jika tombol tidak memberikan respons atau menghasilkan perilaku yang tidak sesuai, maka diperlukan perbaikan pada sistem.

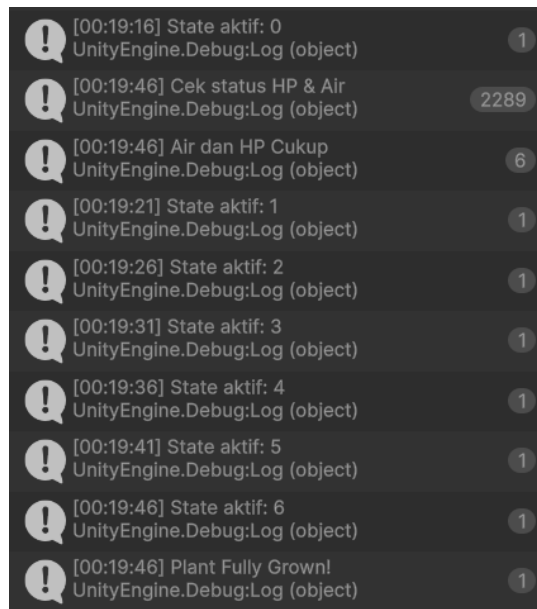
Tabel 4. 2 Tabel Uji Fungsi Tombol

Nama Tombol	Fungsi	Hasil Yang Diharapkan	Hasil Pengujian	Status
Mulai	Memulai permainan	Berpindah dari scene menu utama ke scene gameplay	Berhasil	Valid
Lanjut	Melanjutkan permainan	Melanjutkan permainan dari data tersimpan	Berhasil	Valid
Mulai Ulang	Mereset progres	Memulai permainan dari awal	Berhasil	Valid
Tutorial	Membuka Tutorial	Scene tutorial terbuka	Berhasil	Valid
Panah Kanan-Kiri	Mengganti slide tutorial	Slide tutorial berganti	Berhasil	Valid
Keluar	Keluar permainan	Aplikasi ditutup	Berhasil	Valid
Pengaturan	Membuka panel pengaturan	Panel pengaturan terbuka	Berhasil	Valid

Slider BGM dan SFX	Mengatur volume suara	Volume berubah	Berhasil	Valid
Tombol (X)/Kembali	Menutup panel dan berganti menu	Panel atau menu tertutup dan berganti	Berhasil	Valid
Warehouse/Penyimpanan	Membuka panel penyimpanan	Panel penyimpanan terbuka	Berhasil	Valid
Item	Membuka item info panel	Panel info item dan tombol gunakan keluar	Berhasil	Valid
Gunakan	Menggunakan item	Item bisa digunakan	Berhasil	Valid
Shop/Toko	Membuka panel toko	Panel toko terbuka	Berhasil	Valid
Beli	Membeli item terpilih	Item terbeli dan coin berkurang	Berhasil	Valid
Pot	Menanam benih, panen, reset tanaman	Benih tertanam, timer dimulai, status bar bereaksi, model tanaman di reset	Berhasil	Valid
Siram	Menyiram Tanaman	Menambah status bar air	Berhasil	Valid
Pupuk	Memulihkan Kesehatan tanaman dan mempercepat pertumbuhan	Waktu timer berkurang dan kesehatan tanaman dipulihkan	Berhasil	Valid
Info (i)	Membuka panel info tanaman	Panel info tanaman terbuka dan menampilkan informasi state tanaman saat ini	Berhasil	Valid
Hama	Membuka minigame	Scene minigame terbuka	Berhasil	Valid
Serang/Spray	Menyerang hama	Bar status hama berkurang	Berhasil	Valid
Bertahan/Defense	Bertahan dari serangan hama	Mengurangi serangan hama yang diterimaa pemain	Berhasil	Valid
Kabur/Run	Keluar dari minigame	Scene Kembali ke gameplay utama, pemain menerima penalti	Berhasil	Valid

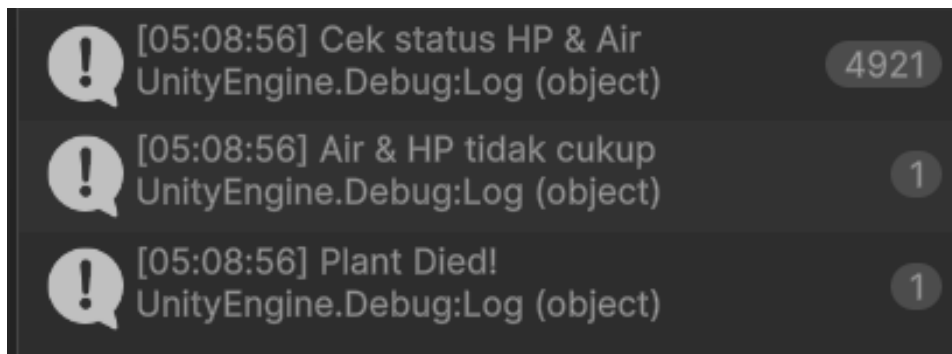
4.1.5 Uji Coba FSM Pada Model Tanaman

Pada pengujian FSM terhadap model tanaman ini dilakukan untuk menguji keadaan *state-state* tanaman mulai dari tahapan benih hingga mencapai fase panen. Faktor-faktor yang mempengaruhi perubahaan *state* tersebut seperti kadar Air dan juga HP tanaman.



Gambar 4. 15 Unity Console: Perubahan State Model Tanaman

Pada gambar 4.15 dapat terlihat bahwa setiap kali *timer* habis dan tanaman akan memasuki *state* selanjutnya, sistem akan melakukan cek status bar Air dan juga HP untuk memastikan bahwa *fill bar* dari status masih berisi. Jika iya, sistem akan mengatakan bahwa Air dan HP cukup sehingga tanaman bisa memasuki *state* selanjutnya dan model akan berubah sesuai fase yang telah dimasuki saat ini.



Gambar 4. 16 Unity Console: Perubahan State Ke Model Dead

Tanaman akan memasuki *state* model *Dead* apabila kriteria untuk berlanjut ke fase selanjut tidak terpenuhi. Pada gambar 4.16 dapat terlihat ketika sistem melakukan cek status HP & Air, sistem menemukan bahwa *fill bar* dari status tersebut kosong, sehingga memberi isyarat bahwasanya kriteria Air dan HP tidak terpenuhi dan perubahan fase terhenti dan dialihkan ke *state Dead*.



Gambar 4. 17 Model State Tanaman Tomat

Gambar 4.17 menampilkan salah satu fase pertumbuhan tanaman yaitu tanaman tomat. *State 0* atau *state* awal dari tanaman ketika pertama kali ditanam akan menampilkan benih diatas pot, kemudian setelah melewati durasi *timer*

pertumbuhan dan memenuhi kriteria untuk berpindah *state*, tanaman akan memasuki *state* 1 atau tunas, lanjut berlanjut hingga mencapai *state* 6 atau terakhir yaitu panen. Adapun *state* tambahan yang hanya akan muncul ketika tanaman gagal memenuhi kriteria pertumbuhan dan berakhir mati, yaitu *state* 7 atau *Dead*.

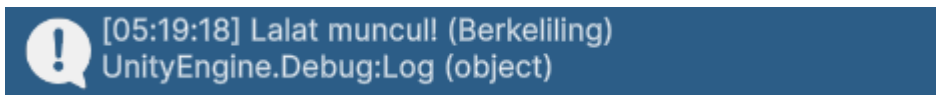
Tabel 4. 3 Tabel Uji Perubahan State Tanaman

No	Pengujian	Input	Output	Status
1	Perubahan State Benih hingga Panen (1-6)	Status HP & Air Memenuhi Kriteria.	State Selanjutnya Aktif.	Valid
2	Perubahan ke State Dead (7)	Status HP & Air Tidak Memenuhi Kriteria.	Plant Died (Tanaman Mati).	Valid

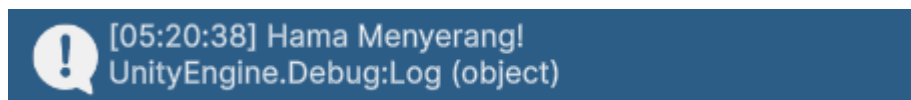
Setelah diuji, perubahan *state state* tanaman bekerja sesuai dengan rancangan diagram *Finite State Machine* (FSM) untuk tanaman, Dimana masing masing *state* akan mendapatkan output yang sesuai tergantung kriteria input yang dimiliki, sehingga tidak ada *state* yang terlewatkan.

4.1.6 Uji Coba FSM Pada Model Hama

Hama berperan sebagai pengganggu di dalam permainan ini, pada perancangannya hama akan muncul dan berkeliling (*idle*) disekitar tanaman saat baru pertama kali *spawn*. Hama akan memasuki *state* serang apabila ia diinterupsi selama *state idle*, caranya dengan cara menekan model hama yang berkeliaran. Lalu diakhiri dengan 2 *state*, jika pemain berhasil menyingkirkan hama, model hama akan menghilang dan berhenti mengganggu pertumbuhan tanaman, dan jika pemain kalah, maka hama akan Kembali ke *state* awalnya yaitu berkeliling dan terus mengganggu tanaman.

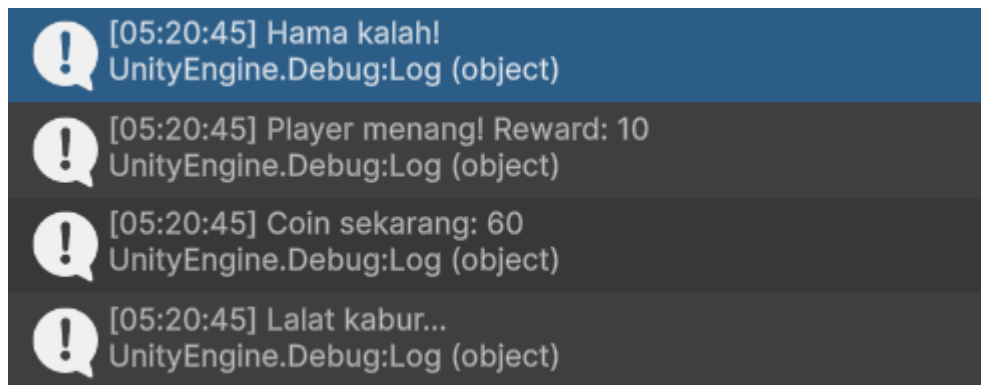


Gambar 4. 18 Unity Console: Spawn Hama Lalat (Idle)



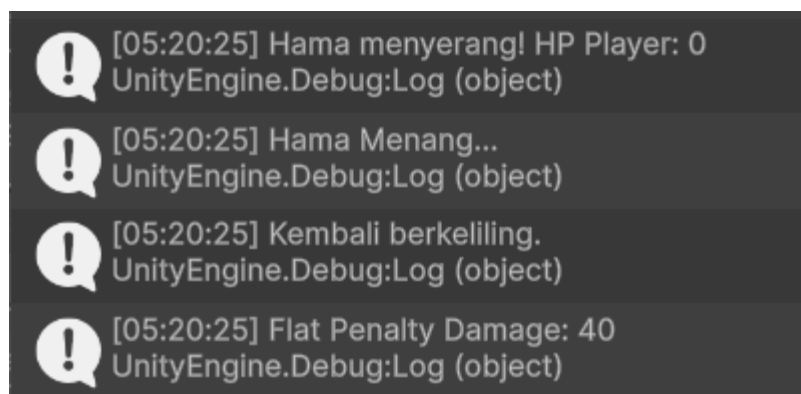
Gambar 4. 19 Unity Console: Hama Memasuki State Menyerang

Gambar 4.18 menunjukkan bahwa ketika Hama *spawn*, dia akan mulai berkeliling (*idle*). Lalu di gambar 4.19 terlihat bahwa hama menyerang, hal ini hanya akan terjadi jika pemain sudah mengetuk model hama yang berkeliling. *Scene* akan berpindah ke minigame dan memulai pertarungan.



Gambar 4. 20 Unity Console: Hama Kalah dan Kabur

Pada gambar 4.20 menampilkan *state* yang akan dimasuki hama ketika player berhasil mengalahkannya di dalam pertarungan atau HP dari hama mencapai 0. Player akan mendapatkan reward dan hama akan kabur, sekaligus menghapus model hama pada *scene* utama permainan untuk sementara.



Gambar 4. 21 Unity Console: Hama Menang dan Kembali Berkeliling (Idle)

State yang terjadi pada hama di gambar 4.21 akan tercapai jika hama berhasil menang melawan player. Hama akan Kembali ke *state* awalnya yaitu berkeliling dan terus mengganggu pertumbuhan tanaman. Disisi lain, player akan menerima sejumlah damage yang akan mempengaruhi HP bar tanaman.

Tabel 4. 4 Tabel Uji Perubahan State Hama

No	Pengujian	Input	Output	Status
1	<i>State Idle</i>	Hama Spawn Secara Random.	Hama Berkeliling (Idle).	Valid
2	<i>State Menyerang</i>	Pemain Menekan Model Hama.	Masuk Minigame (Battle).	Valid
3	<i>State Menang</i>	HP Pemain ≤ 0 atau Pemain Kabur dan HP Hama ≥ 0 .	Hama Menang, Kembali ke <i>State Idle</i> .	Valid
4	<i>State Kabur</i>	HP Hama ≤ 0 dan Pemain Menang.	Hama Kabur, Hapus Model Hama Dari <i>Scene</i> .	Valid

4.1.7 Uji Coba Algoritma Time-Based Growth Pada Timer

Pada penelitian ini, perubahan *state* tanaman dikelola menggunakan algoritma *time-based growth*, yaitu metode pertumbuhan berbasis waktu. Setiap tanaman memiliki beberapa fase pertumbuhan yang masing-masing memiliki durasi waktu tertentu. Perpindahan fase bekerja secara otomatis ketika waktu pada fase tersebut habis. *Timer* bekerja dengan mengurangi nilai waktu setiap *frame* menggunakan fungsi *Time.deltaTime*. Jika waktu awal suatu fase dinotasikan sebagai T_0 , maka sisa waktu pada saat tertentu dapat dituliskan secara sederhana seperti dengan menggunakan rumus sebagai berikut: $T(t) = T_0 - t$

Di mana t adalah waktu yang telah berlalu sejak *timer* dimulai. Ketika nilai $T(t) \leq 0$, sistem akan mulai memeriksa status bar HP dan Air dari tanaman sebelum berpindah ke fase pertumbuhan berikutnya hingga tanaman mencapai fase akhir panen ataupun dead.

4.1.7.1 Pengaruh Status Bar HP dan Air Pada Timer

Nilai status dari HP dan Water bar juga menjadi faktor penentu dari sistem *Timer* ini. Jika indikator status HP mencapai nol, tanaman tidak akan langsung mati, melainkan memasuki fase *delay death* yang akan berlangsung selama 30 detik sebelum pada akhirnya tanaman akan benar benar mati. Secara logika sistem akan bekerja seperti berikut:

Tabel 4. 5 Pengaruh Status HP dan Air Pada *Timer*

Input	Output	Status
Status Bar Air = 0	<i>Timer</i> berjalan normal.	Valid
Status Bar HP = 0	<i>Timer</i> akan memasuki fase <i>delay death</i> selama 30 detik.	Valid
Status Bar HP diisi	<i>Timer</i> Kembali normal.	Valid
Status Bar Hp dibiarkan kosong hingga durasi <i>delay death</i> habis	Tanaman akan mati.	Valid

Mekanisme ini meniru kondisi biologis tanaman pada dunia nyata, air akan menjadi faktor pertama menjadikan tanaman itu sehat, tetapi Ketika air habis,

tanaman akan mengalami penurunan kesehatan dan stres sebelum akhirnya mati. Dari sisi gameplay, mekanisme ini memberi kesempatan kepada pemain untuk menyelamatkan tanaman tanpa harus takut kehilangan progress secara mendadak.

4.1.7.2 Pengaruh Fitur Pupuk Terhadap Percepatan Durasi Timer

Terdapat fitur pupuk yang berfungsi untuk mempercepat durasi *timer* pertumbuhan dengan cara mengurangi 50% dari sisa waktu fase yang sedang berlangsung, agar pemain dapat melanjutkan permainan tanpa perlu menunggu seluruh durasi *timer* hingga selesai. Secara matematis, penghitungan dapat dilakukan dengan menggunakan rumus sebagai berikut:

$$T \text{ saat ini} = 0,5 \times T \text{ tersisa}$$

Pada rumus ini, T mengacu pada *Timer*, yang mana durasi *timer* saat ini akan dikalikan 0,5 sehingga memotong 50% dari durasi *timer*. Metode ini disebut reduksi waktu proporsional, karena pengurangan waktu bergantung pada sisa waktu yang ada

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil penelitian dan uji coba implementasi metode *Finite State Machine* (FSM) dan algoritma *Time-Based Growth* pada game simulasi perawatan tanaman yang telah dilakukan, dapat disimpulkan bahwa sistem yang dikembangkan mampu bekerja sesuai dengan rencana perancangan dan tujuan penelitian. Fitur-fitur utama yang di desain, mulai dari perubahan *State* objek Tanaman dan Hama, mekanisme *timer* berbasis *time-based growth*, sistem pengurangan status bar HP dan Air, fase *delay death*, pengaruh pupuk terhadap percepatan pertumbuhan, serta faktor pengganggu seperti hama, telah berhasil diimplementasikan dan berfungsi secara logis.

Hasil pengujian menunjukkan bahwa sistem mampu merepresentasikan mekanisme pertumbuhan tanaman secara dinamis dan responsif terhadap interaksi pengguna. Selain itu, berdasarkan hasil evaluasi dan *trial* yang diperoleh melalui pengujian oleh beberapa orang dan penulis sendiri, sistem dinilai cukup efektif dalam memberikan pengalaman simulasi bermain yang interaktif serta meningkatkan pemahaman pemain terhadap konsep yang diangkat dalam penelitian.

5.2. Saran

Dari hasil penelitian dan pengembangan game yang telah dilakukan, masih terdapat beberapa kekurangan sehingga menghasilkan saran yang dapat dipertimbangkan untuk pengembangan lebih lanjut:

1. Pengembangan sistem pertumbuhan dapat diperluas dengan menambahkan variable penentu lain seperti cahaya, suhu, atau cuaca agar permainan simulasi menjadi lebih realistis dan mendekati kondisi biologis di dunia nyata.
2. Perlu dilakukan optimasi performa aplikasi, khususnya untuk perangkat Android dengan spesifikasi rendah.
3. Sistem pendukung dapat dikembangkan untuk membuat permainan lebih menarik, seperti sistem level, achievement, serta menambah variasi jenis tanaman untuk meningkatkan motivasi dan keterlibatan pemain dalam jangka panjang.
4. Pengujian dan evaluasi lebih lanjut sangat disarankan dengan menambah jumlah responden yang lebih banyak untuk penelitian selanjutnya.

DAFTAR PUSTAKA

- Amacker, J., Kleiven, T., Grigore, M., Albrecht, P., & Horn, C. (2022). *A Learned Simulation Environment to Model Plant Growth in Indoor Farming. 1*, 1–8.
<http://arxiv.org/abs/2212.03155>
- Arjoranta, J. (2019). How to Define Games and Why We Need to. *The Computer Games Journal*, 8(3–4), 109–120. <https://doi.org/10.1007/s40869-019-00080-6>
- Assidiq, M. S. (2020). Game Edukasi Pengenalan Pertumbuhan Hewan dan Tumbuhan pada Anak Tunagrahita Ringan di SLB YPASP Gondangrejo. *Komunikasi Dan Informatika*, 1–19.
- Azizah, F., & Salehudin, M. (2023). Journal of Instructional and Development Researches Media Game Edukasi di Gadget: Studi Literatur Manfaat dan Dampaknya Terhadap Perkembangan Anak Usia Dini. *JIDeR*, 3(6), 264–271.
<https://doi.org/10.53621/jider.v3i6.265>
- Bustare, B. C. R., Wahyudin, Aan, H., & Putri, N. S. (2024). *PENGEMBANGAN DAN EVALUASI GAME EDUKASI BERBASIS ANDROID DENGAN UNITY 3D UNTUK MENINGKATKAN. 11*(3), 131–143.
- Dewi, L. P., Putra, S. D., & Subyantoro, E. (2023). Gamifikasi untuk Meningkatkan Kesadaran Lingkungan pada Anak Usia Dini: Studi Kasus pada Game Simulasi Merawat Tanaman. *ROUTERS: Jurnal Sistem Dan Teknologi Informasi*, 1(2), 103–108. <https://doi.org/10.25181/rt.v1i2.3119>
- Goldstone, W. (2011). *Unity 3.x game development essentials : game development with C# and Javascript*.

- Hidayah, R. E., Murtono, Utaminingsih, S., & Pratama, H. (2022). *The Effectiveness of Game-Based Learning in The Media Android-Based Artificial Vegetative Plant Propagation Based on Natural Science Subject*. *1*(15), 61–67.
- Hitti, Y., Buzatu, I., Del Verme, M., Lefsrud, M., Golemo, F., & Durand, A. (2021). *GrowSpace: Learning How to Shape Plants*. <http://arxiv.org/abs/2110.08307>
- Jagdale, D. (2021). Finite State Machine in Game Development. *International Journal of Advanced Research in Science, Communication and Technology*, *10*(1), 384–390. <https://doi.org/10.48175/ijarsct-2062>
- Jasmin, F., Salsabila, L., & Tjandra, W. (2025). Simulation and Game-Based Learning sebagai Strategi Pembentukan Karakter dalam Pembelajaran Pendidikan Pancasila. *Jurnal Ilmiah Mimbar Demokrasi*, *24*(2), 507–513. <https://doi.org/10.21009/jimd.v24i2.54661>
- Kuznetsov, V., Moiseienko, M., Moiseienko, N., Rostalny, B., & Kiv, A. (2022). *Using Unity to Teach Game Development*. *2*(Aet 2020), 506–515. <https://doi.org/10.5220/0010933400003364>
- Nonogaki, H. (2014). Seed dormancy and germination-emerging mechanisms and new hypotheses. *Frontiers in Plant Science*, *5*(MAY), 1–14. <https://doi.org/10.3389/fpls.2014.00233>
- Purba, A. S., Hufad, A., Amal, B. K., Ahyani, A., & Sutarni, N. (2020). Development of games instruction within plant growth concept. *Journal of Engineering Science and Technology*, *15*, 1–10.
- Puspitasari, A. N., Pribadi, D. U., & Suhardjono, H. (2023). *Growth and Yield of*

Tomato (Lycopersicum esculentum Mill .) Due to Application of Hormonic PGR and NPK Fertilizer. 12(4), 948–956.

Rabin, S. (2009). *COLLECTED WISDOM OF GAME AI PROFESSIONALS*.

Ramadhan, D. N., & Adhisa, R. R. (2024). Pengembangan Game Edukasi Materi Sistem Tata Surya Sekolah Dasar Menggunakan Aplikasi Construct 3. *Jurnal Informatika) Universitas Muhammadiyah Tangerang P, 8(4), 2722–2713.*

Sari, D. A., Yulianto, & Fadhlina, N. R. (2015). “Kebunku” Educational Game Using Construct 2. 84–91.

Satria, E., Septiana, Y., & Ramadhan, R. (2022). Rancang Bangun Game Edukasi Pengenalan Bagian-Bagian Tumbuhan untuk Siswa Sekolah Dasar Berbasis Android. *Jurnal Algoritma, 18(2), 633–641.*

<https://doi.org/10.33364/algoritma/v.18-2.984>

Savitri, Puspita, D. E., Usnawiyah, & Akmal, J. (2025). Cucumber (Cucumis sativus L.) Growth and Productivity Optimization through the Application of Various Types of Organic Fertilizers. *Serambi Journal of Agricultural Technology, 2025, 7(2): 264–274 | 264 Serambi Journal of Agricultural Technology (SJAT), 7(2), 264–274.*

Summerley, R., Rubino, G., Lelievre, E., & Phillips, T. (2020). Creation of a meaningful gardening game : Increasing players ’ nature awareness through soil management and plant recognition mechanics. *Journal of Environmental Media, 1(2), 185–207.*

Taiz, L., & Zeiger, E. (1946). Plant physiology. In *Science progress* (Vol. 34, Issue 136). <https://doi.org/10.1017/9781108486392>

WA’DAN, A. M. (2023). Pengembangan Game Pertanian Menggunakan Finite

State. *Journal Sains Dan Teknologi*.

Wahyuni, U. M., Rahmadoni, J., Kartika, A. D., & Hanim, H. (2022). The Combination of R&D and MDLC Models in WEB-Based Interactive Learning Media. *Sistemasi*, *11*(3), 713.

<https://doi.org/10.32520/stmsi.v11i3.2085>