



ISSN : 2339 - 1871

JURNAL ILMIAH BETRIK

Besemah Teknologi Informasi dan Komputer

Editor Office : LPPM Sekolah Tinggi Teknologi Pagar Alam, Jln. Masik Siagim No. 75
Simpang Mbacang, Pagar Alam, SUM-SEL, Indonesia
Phone : +62 852-7901-1390.
Email : betrik@sttpagaralam.ac.id | admin.jurnal@sttpagaralam.ac.id
Website : <https://ejournal.sttpagaralam.ac.id/index.php/betrik/index>

PERANCANGAN GAME ARCADE 2D ”SPACESHIP BATTLE” MENGGUNAKAN FLAME GAME ENGINE BERBASIS MOBILE

Bambang Widodo

Program Studi Perhotelan Politeknik Sahid

Jalan Kemiri Raya, Kota Tangerang Selatan, Banten 15418

Sur-el :bam.widodo@polteksahid.ac.id¹

Abstrak: Pengembangan Game 2D saat ini banyak dikembangkan dengan berbagai platform Bahasa pemrograman salah satu Bahasa pemrograman yang saat ini sedang naik daun adalah dart dengan flutter Framework. Tujuan penelitian ini adalah untuk merancang sebuah game 2D ”SpaceShip Battle” berbasis mobile dengan menggunakan flame sebagai game engine pada flutter framework sedangkan metode pengembangan game menggunakan metode Game Development Life Cycle (GDLC) dan Unifield Modeling Language (UML) sebagai model perancangan game. Game mengadopsi game klasik pertempuran pesawat luar angkasa untuk usia anak 7-10 tahun, judul game yang akan dikembangkan adalah ”SpaceShip Battle” dengan genre game arcade, dengan elemen game terdiri dari pesawat luar angkasa, pesawat musuh, kekuatan bantuan, peluru, joystick, paralaks, dasbor, efek suara dan musik serta menu pemain. Mekanika dari game terdiri dari poin dan nyawa, pesawat pemain dapat bergerak pada layar dengan gerakan atas, bawah, kiri dan kanan dengan posisi pesawat menghadap ke atas. Hasil penelitian berupa prototype game dua dimensi bergenre arcade yang berjalan pada platform android mobile dan hasil ujicoba penggunaan flame game engine yang dirancang dengan UML membuat pengembangan game lebih terstruktur dan membuat proses coding lebih efisien.

Kata kunci : Game 2D, Flame, Flutter, GDLC, UML, Mobile

Abstract: Development of 2D games is currently being developed with various programming language platforms. One of the programming languages that is currently on the rise is Dart with the Flutter Framework. The purpose of this study was to design a mobile-based 2D game ”SpaceShip Battle” using flame as the game engine on the Flutter framework while the game development method used the Game Development Life Cycle (GDLC) and Unifield Modeling Language (UML) methods as game design models. The game adopts the classic spaceship battle game for children aged 7-10 years, the title of the game to be developed is ”SpaceShip Battle” with the arcade game genre, with game elements consisting of spaceships, enemy planes, auxiliary forces, bullets, joysticks, parallax, dash, sound effects and music and player menu. The mechanics of the game consist of points and lives, the player's plane can move on the screen in up, down, left and right motions with the plane facing up. The results of the research are a two-dimensional arcade game prototype that runs on the Android mobile platform and the results of trials using the flame game engine designed with UML make game development more structured and make the coding process more efficient.

Keywords: 2D Games, Flame, Flutter, GDLC, UML, Mobile.

1. PENDAHULUAN

Perkembangan teknologi informasi khususnya pada perangkat mobile telah mengubah cara kita berinteraksi dan mengkonsumsi hiburan. Game mobile telah menjadi salah satu bentuk hiburan paling populer dan diminati oleh sebagian besar pengguna perangkat seluler di seluruh dunia. Keberhasilan pengembangan game mobile tidak hanya ditentukan oleh gameplay yang menarik, akan tetapi juga bagaimana efisiensi dalam pengembangan dan kualitas tampilan visual yang menarik dan juga model pemasaran yang digunakan [1].

Maka dalam upaya untuk membuat game mobile 2D yang menarik, banyak pengembang yang menggunakan Flutter, Flutter yaitu sebuah framework open-source yang dikembangkan oleh google untuk membangun aplikasi mobile lintas platform dengan cepat dan efisien. Flutter memberikan berbagai keuntungan seperti performa tinggi, dukungan animasi yang halus, dan mudahnya integrasi dengan platform mobile utama seperti Android dan iOS serta dapat berjalan baik pada lingkungan Desktop dan Website [2].

Pada umumnya pengembangan game dibuat dengan menggunakan game engine Unity dan Bahasa pemrograman C#, seperti pada penelitian Anthony, Pragantha, and Haris [3] meneliti tentang pengujian kecepatan dari 3 buah bola yang memiliki karakter yang berbeda dengan pengujian *blackbox* dan pengujian *alpha* menggunakan menggunakan *game engine* Unity dan Bahasa pemrograman C# begitu juga pada penelitian Kurniawan [4] mengembangkan Game 2D "Omar's Adventure" menggunakan game engine Unity dengan metode *Finite State Machine* (FSM) dengan memasukkan unsur islami sehingga pemainnya dapat mengenal dan mempelajari elemen-elemen islami melalui game yang menarik. Pada penelitian Khairani, Fadila, and Nugroho [5] pembuatan Game 2D petualangan anak sebagai sarana bagi anak-anak untuk bermain dan belajar dikembangkan dengan

Game Engine Unity dan bahasa pemrograman C# dengan waterfall sebagai metode perancangannya. sedangkan pengembangan game dengan flutter masih sangat sedikit padahal pada flutter terdapat flame engine yang sangat powerfull dalam membuat game 2D yang menarik.

Flame Engine adalah sebuah package open-source yang secara khusus dikembangkan untuk membantu pembuatan game 2D di dalam framework Flutter. Flame engine hadir dengan kemampuan Cross-platform, Grafika 2D yang Kuat, Animasi, simulasi fisika, Input dan Interaksi, Suara dan Musik, UI (Antarmuka Pengguna), Kontrol Kamera, Tilemaps, Dokumentasi dan dukungan komunitas yang besar [2]. Dengan flame engine, pengembang dapat memanfaatkan berbagai fitur dan komponen yang telah disediakan, sehingga mempercepat dan mempermudah proses pengembangan game dengan kualitas yang tinggi.

Pada penelitian Andriyat Krisdiawan and Darsanto [6] membangun mobile Action Game dengan teknik pengembangan game GDLC (*Game Development Life Cycle*), menggunakan model UML (*Unifield Modeling Language*) berhasil diterapkan dan sangat memudahkan dalam proses coding, maka dari itu penggunaan flame engine akan lebih mudah jika dalam proses perancangan game menggunakan model UML sebab pada dasarnya flame adalah kumpulan class objek yang dibuat dalam satu package.

Pada penelitian ini bertujuan untuk merancang dan mengimplementasikan mobile game 2D "Spaceship Battle" menggunakan Flame Engine di dalam framework Flutter. Pendekatan dalam perancangan menggunakan UML (*Unifield Modeling Language*) sebagai basis class objek yang akan digunakan dalam implementasi coding di flutter. Penelitian ini diharapkan akan memberikan kontribusi dalam pengembangan game development pada platform Flutter dan Flame Engine, serta

menginspirasi pengembang lainnya untuk menciptakan game 2D yang menarik, inovatif, dan berkualitas tinggi bagi para pengguna perangkat mobile.

2. METODE PENELITIAN

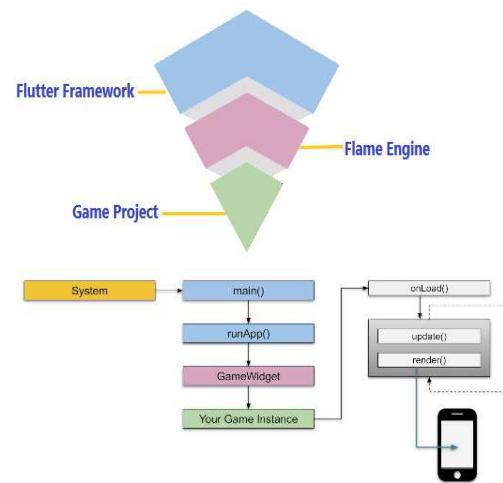
2.1 Flame Engine

Flame adalah package game flutter yang menyediakan paket lengkap solusi untuk pembuatan game. Flame memanfaatkan infrastruktur andal yang telah disediakan oleh Flutter tetapi dengan menyederhanakan kode yang diperlukan untuk membangun sebuah game. Flame menjadikan proses implementasi loop game yang sederhana dan efektif, serta memiliki fungsionalitas baik. Contohnya; input, gambar, sprite, sprite sheet, animasi, deteksi tabrakan, dan sistem komponen yang disebut Flame Component System (FCS) [2]. Game Loop adalah inti dari Flame Game Engine, Flame secara konstan menggambar ulang layar berdasarkan status game yang terjadi saat ini atau dalam kata lain Game Loop adalah proses menggambar ulang tanpa henti dilakukan terus-menerus melalui 3 tahapan yang berulang yaitu : Input data, Update status game dari data input dan Render status game (layar, suara dsb) yang dapat digambarkan pada gambar berikut :



Gambar 1. Game Loop

Flame Engine merupakan class turunan dari Flutter sehingga class Game yang berisi game loop merupakan class turunan dari Flame Engine yang dapat digambarkan pada gambar berikut ini :



Gambar 2. Struktur Class Game

Pada penelitian terdahulu mengenai Mobile Game telah cukup banyak dilakukan seperti pada penelitian Bao [1] menilai situasi industri game seluler di negara Tiongkok, menentukan bagaimana industri game berkembang dalam beberapa dekade terakhir, dan melihat model pemasaran yang digunakan pengembang game untuk agar berhasil di Tiongkok sedangkan Derakhshandi dkk [7] menyajikan studi tentang pendekatan dan kerangka kerja, yang disebut MAndroid, dengan menghasilkan permainan papan 2D multi pemain klasik berbasis Android dalam mode otomatis, dengan menerapkan konsep Model-Driven Engineering (MDE). Model ini kemudian secara otomatis diubah menjadi kode yang dapat dijalankan di semua ponsel dan tablet yang menjalankan Android 4.4 atau lebih tinggi.

Penelitian lain dilakukan oleh Anggraini, Fadila, dan Nugroho [8] merancang game 2D untuk pembelajaran ilmu tajwid menggunakan metode Finite State Machine dengan game engine Unity. Begitu juga Saraubon [9] mengembangkan game edukasi berbasis mobile pembelajaran bahasa Inggris untuk pembelajar ESL berdampak positif dengan peningkatan kemampuan siswa secara signifikan dibandingkan dengan metode konvensional. Sedangkan Faten dkk [10] menggunakan permainan Game 2D dalam kegiatan hafalan Qur'an yang dikembangkan dengan metode ADDIE menggunakan pemrograman game engine Unity untuk terapi stress.

Muhammad Rumakey, Dedi Irawan, dan Wahid [11] mengembangkan game 3D menggunakan platform Unity3D sebagai

game engine dan menerapkan metode FSM (Finite State machine) pada karakter musuh sehingga dapat berinteraksi dengan player, dan mendukung interaksi yang terdapat pada game. Antonio dan Pragantha [12] mendesain Game “Weapowize” adalah game 2D untuk platform Android yang berada di ponsel sehingga dapat dimainkan di mana saja dan kapan saja. Game Engine Unity2D dan C# sebagai bahasa pemrograman digunakan dalam perancangannya.

Sari, Nurtamam, dan Hanik [13] mengembangkan media pembelajaran berbentuk Multimedia Interaktif Berbasis Game 2D Flash pada pembelajaran matematika materi pecahan sederhana untuk siswa kelas III SD. Cubukcu, Canbazoglu, dan Ozerdem [14] melakukan studi untuk mengoptimalkan proses pembelajaran anak down syndrome dengan menyediakan permainan elektronik bagi pendidik yang dapat digunakan untuk mengajar dalam aplikasi seluler yang akan memperkaya proses pengajaran selain memperbarui metode pengajaran mereka ke standar Industri 4.0.

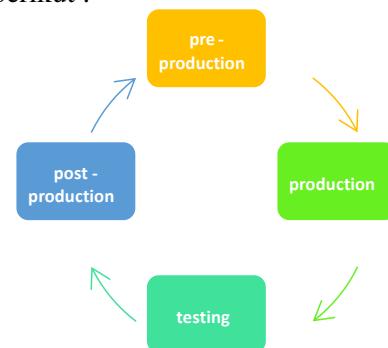
Panduwinata, Arhandi, dan Nugraha [15] membuat gamifikasi sistem penukaran botol plastik menjadi tiket digital dengan metode evaluasi UAT (User Acceptance Test) untuk mengetahui respon masyarakat terhadap aplikasi yang telah dibuat. Md Shukri dan Ariffin [16] mengembangkan game mobile digital 2D “Awang Sains” menggunakan design and develop research (DDR) yang disematkan dengan model ADDIE (Analysis, Design, Development, Implementation, dan Evaluation), berbasis pedagogi dan instruksional untuk membantu siswa menguasai mata pelajaran IPA.

Mau [17] membuat Game “Shooter Platformer” di mana permainan akan menjalankan karakter shooter platformer yang harus menyelamatkan bumi dari kehancuran yang ingin di kuasai oleh musuh dan pengikutnya. Sofware Unity 3D digunakan dalam pengembangan aplikasi ini dan menerapkan kecerdasan buatan FSM (Finite State Machine). Ardyanto dan Pamungkas [18] membuat Game 2D “The Adventure of Hanuman” menggunakan Unity dan C#, game dibuat dari storyboard dengan desain karakter, background dan rintangan menggunakan Adobe Illustrator CS 6. Sagala, Jonemaro, dan Wardhono [19]

mengembangkan game Augmented reality game platformer 2D “Slime Collector” menggunakan teknik projection mapping. Metode formal elements digunakan dalam game ini kemudian digital prototype sebagai prototype game.

pada penelitian ini akan menitikberatkan pada penggunaan flame engine pada flutter dalam pembuatan game 2D “Spaceship Battle” berdasarkan metode perancangan menggunakan *Unifield Modeling Language* (UML).

Game Development Life Cycle (GDLC) digunakan sebagai metode pengembangan game ini. Metode GDLC sangat baik diterapkan dalam proses pengembangan game sebab tahapannya sederhana dan sesuai dengan kondisi dan tujuan penelitian, tahapan tersebut dibagi dalam 4 tahapan yaitu praproduksi, produksi, pengujian dan pascaproduksi seperti digambarkan dalam gambar 1 [20] berikut :



Gambar 3. Tahapan Pembuatan Game

a. Pre-Production (Pra-Produksi)

Tahap pra produksi dilakukan untuk menentukan desain game seperti konsep game yang meliputi: judul game, genre game, gameplay, mekanik, alur cerita, karakter, tantangan, faktor kesenangan, aspek teknis, dan dokumentasi elemen dalam dokumen desain permainan. Setelah dokumen desain game dibuat, maka selanjutnya akan dibuat prototype untuk menilai keseluruhan ide dari game yang akan dikembangkan [21]. Pada tahap ini dalam pembuatan desain prototype game menggunakan model UML (*Unified Modelling Language*) yaitu Use Case Diagram, Activity Diagram, Sequence Diagram, Class Diagram dan Perancangan Antarmuka. Class Diagram yang akan dihasilkan mengacu dari class objek pada

flame engine sebagai engine game berbasis flutter framework.

b. Production (Produksi)

Tahap produksi merupakan inti dari proses pembuatan game, pada tahap ini akan menterjemahkan prototype menjadi bentuk nyata dari mulai asset yang diperlukan, proses pembuatan program, dan integrasi asset dengan program dan sourcecode sehingga karakter player, enemy ataupun atribut menjadi satu kesatuan alur cerita pada game [6].

c. Testing (Pengujian)

Pengujian dilakukan baik internal maupun eksternal, jenis pengujian yang dilakukan yaitu pengujian Black-Box yang berfokus pada persyaratan fungsional perangkat lunak. Pengujian black-box ini di desain untuk dapat mengungkapkan kesalahan pada persyaratan fungsional tanpa mengabaikan kerja internal dari aplikasi game [22].

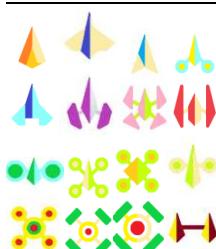
d. Post-Production (Pasca-Produksi)

Pasca produksi dilakukan untuk menyajikan dokumentasi terkini, berbagi pengetahuan dan kegiatan post-mortem serta perencanaan untuk pemeliharaan dan perluasan game [20].

3. HASIL DAN PEMBAHASAN

Tabel 1. Spesifikasi Elemen

Elemen	Detail
Pesawat luar angkasa	<p>Pemain mengontrol pesawat luar angkasa melalui joystick dan dapat bergerak ke atas, bawah, kanan dan kiri</p> 
	<p>Joystick akan memungkinkan kontrol kecepatan pesawat luar angkasa sedemikian rupa sehingga semakin jauh tongkat ditarik dari tengah, semakin cepat pesawat luar angkasa akan bergerak.</p>
	<p>Pemain akan mengetuk layar. Setiap ketukan akan menembakkan peluru. Setiap peluru harus berupa ketukan yang berbeda.</p>
	<p>Saat pesawat ditabrak pesawat musuh, ia kehilangan daya hidup dan dianggap hancur.</p>
	<p>Setiap pemain akan memiliki N% daya hidup. Setiap kali pesawat luar angkasa dihancurkan, pesawat luar angkasa akan muncul kembali di tengah dan pengguna akan memiliki satu daya hidup yang berkurang. Setelah daya hidup hilang, game dapat dimulai ulang atau berhenti.</p>
	<p>Pemain bisa mendapatkan daya hidup ekstra melalui kekuatan bantuan yang ketika diambil mendapat daya hidup ekstra.</p>
	<p>Pemain dapat memindahkan pesawat ruang angkasa ke mana saja di layar.</p>
Pesawat Musuh	<p>Pesawat musuh yang buat secara acak harus dihindari dan ditembakkan oleh Pesawat luar angkasa untuk mendapatkan poin.</p>
	<p>Jika pesawat musuh menabrak pesawat luar angkasa, maka pesawat luar angkasa itu akan hancur.</p>
	<p>Jika sebuah peluru mengenai pesawat musuh itu akan menyebabkan pesawat musuh hancur.</p>



Semua pesawat musuh yang dibuat akan diletakkan pada posisi acak, dengan kecepatan sesuai dengan jenis pesawat musuh.



Kekuatan Tambahan

Jenis pesawat musuh:

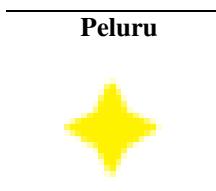
- Ada 16 Jenis Pesawat musuh
- Ada 4 tipe level Pesawat.
- Kecepatan dan Poin masing-masing pesawat berbeda.

Kekuatan tambahan akan menjadi objek yang dibuat secara acak yang akan muncul secara berkala di layar dan menawarkan bonus kepada pemain jika diambil.

Terdapat 4 macam kekuatan tambahan :

- Peluru 3 arah
- Penambah nyawa
- Penghentian sementara pergerakan pesawat musuh
- Bom yang menghancurkan semua pesawat musuh

Kekuatan tambahan akan hilang otomatis setelah 5 detik



Peluru

Peluru dibuat saat pengguna menembak. Untuk setiap ketukan, satu peluru akan dihasilkan.

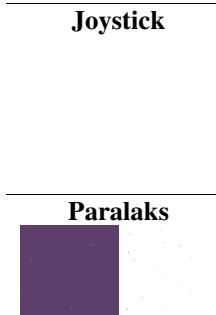
Setiap ketukan dilayar akan dianggap sebagai tindakan

Pemain tidak bisa kehabisan peluru

Peluru akan memiliki kecepatan yang selaras sempurna dengan sudut pesawat ruang angkasa saat ditembakkan. Dengan kata lain karena pesawat luar angkasa selalu mengarah ke atas maka peluru akan selalu bergerak ke atas.

Peluru akan memiliki kecepatan konstan dan kecepatan tidak akan berubah.

Saat ditembakkan, peluru harus mengeluarkan suara yang dapat didengar dan juga harus ada suara yang menahan peluru yang terbang.

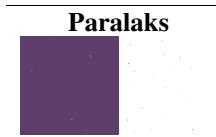


Joystick

Mengontrol pergerakan pemain.

Dapat menggerakkan pesawat luar angkasa ke semua 4 arah (atas, bawah, kiri dan kanan), posisi pesawat ruang angkasa tetap mengarah keatas.

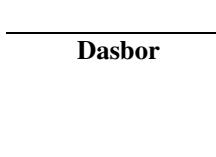
Pesawat luar angkasa harus dipercepat saat joystick menjauh dari tengah dan melambat saat bergerak menuju tengah. Di tengah pemain harus bisa menghentikan pesawat luar angkasa.



Paralaks

Kita harus menambahkan efek paralaks dengan dua set bintang kecil yang bergerak mengikuti gerakan joystick. Dengan kata lain, jika pemain menggerakkan joystick ke atas, paralaks harus merespons secara efektif dengan menggerakkan latar belakang mulai dari bawah. Secara umum: berlawanan arah dengan pergerakan pemain.

Pemain harus bisa mematikan efek paralaks untuk permainan.



Dasbor

Merupakan umpan balik ke pemain yang terdiri dari :

- Poin – nilai yang didapat setelah menghancurkan pesawat musuh
- Daya Hidup – menunjukkan berapa banyak nyawa yang tersisa

Dasbor juga ditampilkan kepada pemain saat permainan selesai.

Pemain juga memiliki kemampuan untuk mendapatkan akses ke pengaturan dari dasbor.

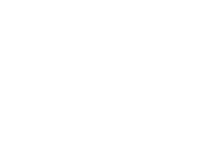


Efek Suara

Pada game ini akan menggunakan sejumlah efek suara dan musik:

- Latarbelakang musik - Dimainkan sepanjang permainan secara berulang terus-menerus.
- Tembakan peluru - Menjadi suara saat pemain menembakkan peluru dari pesawat ruang angkasa.
- Penerbangan peluru - Merupakan suara trailing saat peluru terbang
- Kemampuan tambahan - Suara yang dihasilkan ketika pesawat luar angkasa mengambil Kekuatan Tambahan saat terbang.

Tingkat suara dapat disesuaikan oleh pemain melalui pengaturan.



Saat pengguna menjeda permainan, musik juga harus dijeda.

Menu Pemain

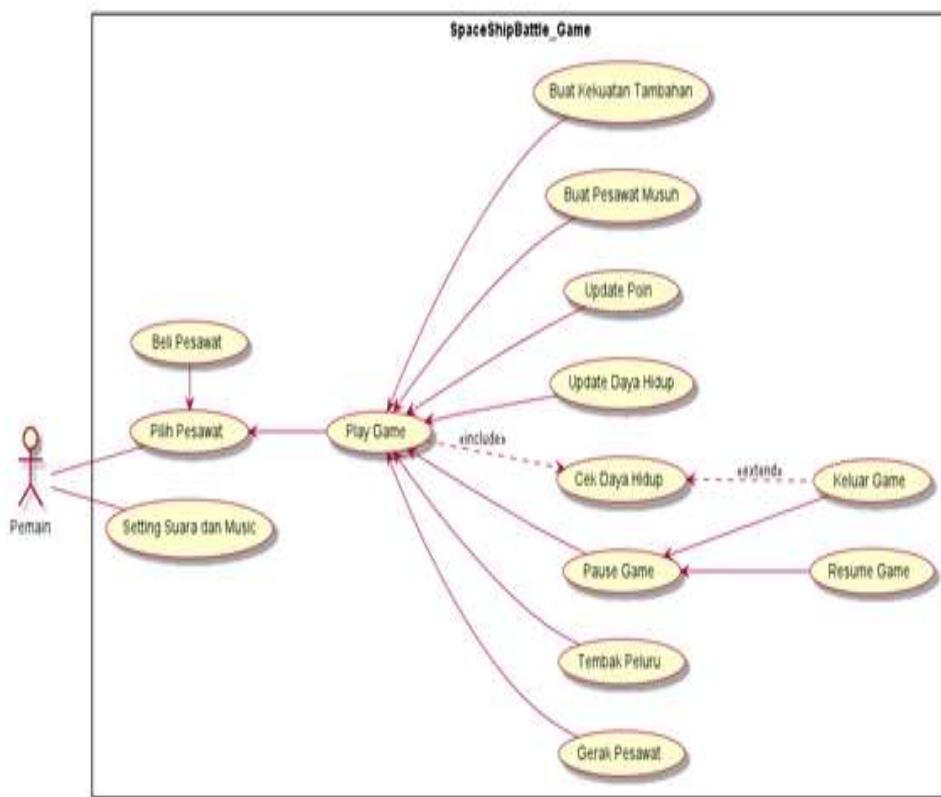
Pemain memiliki kemampuan untuk melakukan beberapa hal dengan game selain bermain game:

- Memiliki tombol "jeda" untuk memungkinkan pemain menjeda permainan.
- Memiliki tombol menu di mana pengguna dapat mengubah beberapa aspek permainan.

Item berikut akan ada di menu:

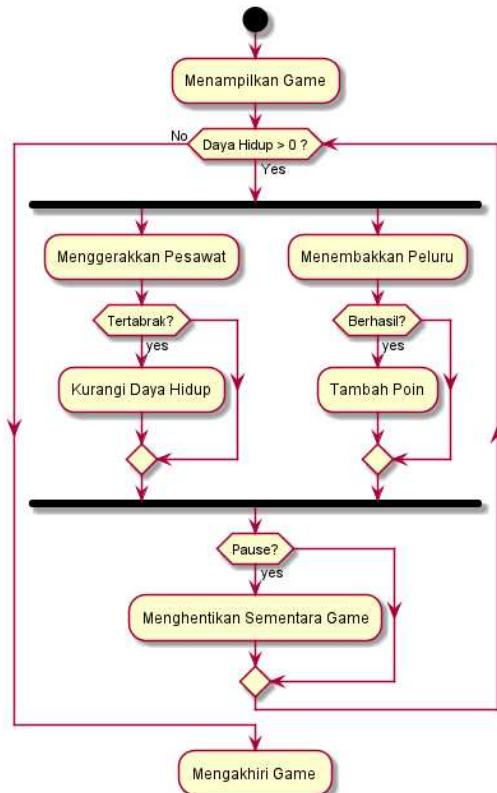
- Keluar dari game
- Mengaktifkan/menonaktifkan efek suara
- Mengaktifkan/menonaktifkan musik
- poin terakhir

Use Case Diagram



Gambar 4. Use Case Diagram Play Game

Activity Diagram



Gambar 5. Activity Diagram Play Game

b. Produksi

Pada bagian produksi inilah hasil akhir berbentuk aplikasi yang terdiri dari kumpulan asset baik file image maupun suara serta pengkodean dengan menggunakan flame game engine dan bahasa pemrograman dart pada flutter framework sehingga terbentuk dalam satu kesatuan alur game semua ini tertuang dalam Game Design Document (GDD). Gambar 8 menunjukkan tampilan menu awal “SPACESHIP BATTLE” dengan pilihan untuk main atau setting, jika memilih main maka akan muncul tampilan “PILIH PESAWAT” dimana pemain dapat membeli pesawat sesuai dengan harga dan poin yang ada kemudian memilih tombol MULAI untuk memulai permainan. Jika pemain memilih setting maka akan muncul tampilan menu setting yang berfungsi untuk menjalankan efek suara dan background musik.



Gambar 8. Tampilan Menu Awal

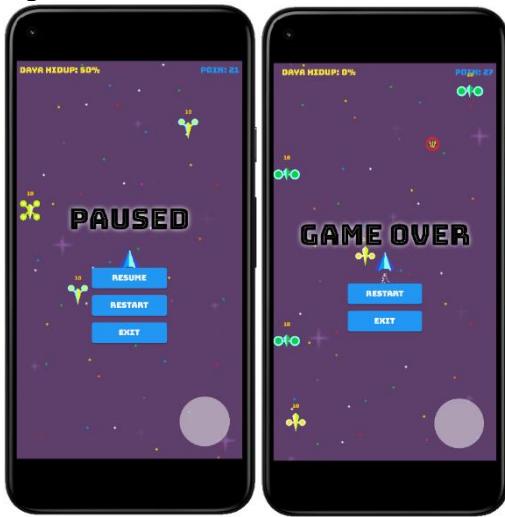
Gambar 9 merupakan tampilan permainan dimana layar game menampilkan latar belakang berupa langit luas dengan bintang-bintang dan planet-planet dilihat dari kejauhan, posisi awal pesawat luar angkasa berada di tengah layar dan dapat bergerak ke segala arah (atas, bawah, kanan dan kiri) pada layar bagian atas terdapat indikator daya hidup dan poin serta tombol pause, pesawat musuh akan muncul secara acak dari atas layar bergerak kearah bawah sedangkan pemain dapat menghindari atau menembakan peluru untuk menghancurkan musuh, poin akan ditambahkan setiap kali pesawat musuh berhasil dihancurkan akan tetapi jika pesawat pemain hancur maka indikator daya hidup akan berkurang.



Gambar 9. Tampilan Play Game

Pada gambar 10 merupakan tampilan layar jika tombol pause ditekan maka akan muncul pilihan resume untuk lanjut, restart untuk mengulang dari awal permainan dan exit untuk keluar permainan. Jika dalam

permainan pada akhirnya pemain kehabisan daya hidup maka otomatis akan muncul layar game over.



Gambar 10. Tampilan Game Over

Deskripsi di atas adalah gambaran umum tentang tampilan aplikasi game "Spaceship Battle." Sebagai pengembang kita dapat menyesuaikan desain tampilan yang sesuai dengan konsep visual dan gaya yang diinginkan untuk game tersebut. Penggunaan Flame Engine pada framework Flutter memberikan fleksibilitas dalam merancang tampilan game yang menarik dan responsif.

c. Pengujian

Rencana Pengujian yang akan dilakukan dengan menguji game yang dibangun secara black box. Pengujian Mobile Game 2D 'Spaceship Battle' menggunakan Flame Engine pada Flutter akan menguji berdasarkan pemain yang telah mencoba permainan. Rencana pengujian selengkapnya terlihat pada tabel dibawah ini:

Tabel 2. Hasil Pengujian Black Box

No.	Test Case	Hasil Harapan	Hasil Keluaran
1.	Pada icon game mobile Pemain menjalankan game	Masuk menu awal "SPACESHIP BATTLE"	Sesuai
2.	Pada menu awal Pemain mengklik tombol "SETTING"	Muncul menu "SETTING"	Sesuai
3.	Pada menu "SETTING" Pemain mengubah on/off Sound Effect dan Background Music	Status dan aksi pada game ikut berubah	Sesuai
4.	Pada menu awal Pemain mengklik tombol "MAIN"	Muncul menu "PILIH PESAWAT"	Sesuai
5.	Pada menu "PILIH PESAWAT" Pemain mengklik tombol "MULAI"	Muncul tampilan layar permainan game dengan nilai indikator daya hidup kembali 100% dan nilai poin 0	Sesuai
6.	Pada layar permainan Pemain mengetuk layar dengan cepat	Muncul gambar peluru beserta suara tembakan	Sesuai
7.	Pada layar permainan Pemain berhasil menembak pesawat musuh	Indikator poin bertambah dan pesawat musuh hilang	Sesuai
8.	Pada layar permainan Pemain menggeser posisi pesawat	Gambar pesawat bergerak sesuai dengan arah pergeseran	Sesuai
9.	Pada layar permainan Pemain mengambil kekuatan tambahan	Indikator otomatis berubah sesuai dengan jenis kekuatan yang diambil	Sesuai
10.	Pada layar permainan Pemain tertabrak pesawat musuh	Indikator daya hidup berkurang dan pesawat muncul kembali ditengah layar	Sesuai
11.	Pada layar permainan Pemain menekan tombol "Pause"	Muncul tampilan menu "PAUSE"	Sesuai
12.	Pada menu "PAUSE" Pemain mengklik tombol "RESUME"	Muncul kembali layar permainan dan melanjutkan permainan sebelumnya.	Sesuai

13.	Pada menu “PAUSE” Pemain mengklik tombol “RESTART”	Muncul kembali layar permainan dari awal dengan nilai indikator daya hidup kembali ke-100% dan nilai poin kembali ke-0	Sesuai
14.	Pada menu “PAUSE” Pemain mengklik tombol “EXIT”	Muncul kembali ke menu awal ““SPACESHIP BATTLE””	Sesuai
15.	Pada layar permainan Nilai indikator daya hidup = 0	Muncul tampilan menu “GAME OVER”	Sesuai
16.	Pada menu “GAME OVER” Pemain mengklik tombol “RESTART”	Muncul kembali layar permainan dari awal dengan nilai indikator daya hidup kembali ke-100% dan nilai poin kembali ke-0	Sesuai
17.	Pada menu “GAME OVER” Pemain mengklik tombol “EXIT”	Muncul kembali ke menu awal ““SPACESHIP BATTLE””	Sesuai

Berdasarkan hasil pengujian didapat kesimpulan bahwa pada proses game secara fungsional aplikasi sudah dapat menghasilkan output yang diharapkan walaupun secara desain game perlu penyempurnaan.

d. Pasca-Produksi

Setelah pembuatan game sudah sampai tahap final maka game siap untuk dirilis dan dipasarkan, proses rilis terdiri dari : peluncuran produk, dokumentasi projek, berbagi pengetahuan dan kegiatan post-mortem serta perencanaan untuk pemeliharaan dan perluasan game.

4. SIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan mobile game 2D "Spaceship Battle" menggunakan Flame Engine dalam framework Flutter. Melalui proses perancangan konsep, implementasi dengan Flame Engine, serta pengujian didapatkan hasil yang signifikan terkait dengan pembuatan game 2D yang menarik dan efisien. Hasil implementasi menunjukkan bahwa Flame Engine merupakan pilihan yang tepat dalam pembuatan game 2D pada platform Flutter. Flame Engine menjadi solusi yang efisien dan efektif dalam menghadirkan game dengan fitur-fitur yang menarik dan kualitas yang baik. Game "Spaceship Battle" diharapkan dapat menjadi pilihan menarik bagi para pecinta game mobile, serta memberikan kontribusi dalam pengembangan game development pada platform Flutter dan Flame Engine.

Daftar Rujukan

- [1] W. Bao, “Current Situation And Development Of Trend Of Chinese Mobile Game Development,” In *Proceedings Of The 2022 2nd International Conference On Enterprise Management And Economic Development (Icemed 2022)*, 2022, Vol. 656. Doi: 10.2991/Aebmr.K.220603.075.
- [2] P. Teale, *Building Games With Flutter*. Birmingham Mumbai: Packt Publishing Ltd., 2022.
- [3] A. Anthony, J. Pragantha, And D. A. Haris, “Pembuatan Game 2d Platform ‘Bola’ Berbasis Website,” *J. Ilmu Komput. Dan Sist. Inf.*, Vol. 10, No. 1, Mar. 2022, Doi: 10.24912/Jiksi.V10i1.17793.
- [4] R. F. Kurniawan, “2d Game ‘Omar’s Adventure’ Design Using The Finite State Machine Method,” *J. Informatics Telecommun. Eng.*, Vol. 6, No. 1, 2022.
- [5] N. Khairani, J. N. Fadila, And F. Nugroho, “Perancangan Game 2 Dimensi Petualangan Anak Menyelamatkan Orangtua Sebagai Media Edukatif Bagi Anak Dengan Metode Waterfall,” *J. Teknol. Inf.*, Vol. 5, No. 1, 2021, Doi: 10.36294/Jurti.V5i1.1779.
- [6] R. Andriyat Krisdiawan And Darsanto, “Penerapan Model Pengembangan Game Dlc (Game Development Life Cycle)Dalam Membangun Game Platform Berbasis Mobile,” *Teknokom*, Vol. 2, No. 1, Pp. 31–40, Mar. 2019, Doi: 10.31943/Teknokom.V2i1.33.
- [7] M. Derakhshandi, S. Kolahdouz-Rahimi, J. Troya, And K. Lano, “A Model-Driven Framework For Developing Android-Based Classic Multiplayer 2d Board

- Games,” *Autom. Softw. Eng.*, Vol. 28, No. 2, P. 7, Nov. 2021, Doi: 10.1007/S10515-021-00282-1.
- [8] A. N. Anggraini, J. N. Fadila, And F. Nugroho, “Rancang Bangun Game 2d ‘Finding Tajwid’ Dengan Metode Finite State Machine Menggunakan Software Unity Hub,” *J. Teknol. Inf.*, Vol. 5, No. 1, Pp. 88–93, Jun. 2021, Doi: 10.36294/Jurti.V5i1.1782.
- [9] K. Saraubon, “Development And Evaluation Of A Mobile Game As An English Learning Tool For Esl Learners,” *Int. J. Adv. Appl. Sci.*, Vol. 8, No. 7, Pp. 77–83, Jul. 2021, Doi: 10.21833/Ijaas.2021.07.009.
- [10] A. Faten, S. Rosali, W. Shazlina, And W. Ismail, “Stress Medication With 2d Game And Al-Quran Reading Mobile Application For Relieve The Stress,” *J. Comput. Technol. Creat. Content*, Vol. 6, No. 2, 2021, [Online]. Available: <Https://Jtec.Org.My/Index.Php/Jtec/Article/View/457/136>
- [11] A. Muhamad Rumakey, J. Dedi Irawan, And A. Wahid, “Pembuatan Game 2d ‘Escape Plan’ Dengan Metode Finite State Machine,” *Jati (Jurnal Mhs. Tek. Inform.)*, Vol. 4, No. 2, Pp. 65–72, Oct. 2020, Doi: 10.36040/Jati.V4i2.2712.
- [12] R. Antonio And J. Pragantha, “Pembuatan Game Arcade 2d ‘Weapowize’ Berbasis Android,” *J. Ilmu Komput. Dan Sist. Inf.*, Vol. 8, No. 1, P. 156, Jan. 2020, Doi: 10.24912/Jiksi.V8i1.11488.
- [13] I. P. Sari, M. E. Nurtamam, And U. Hanik, “Pengembangan Multimedia Interaktif Berbasis Game 2d Flash Pada Pembelajaran Matematika Materi Pecahan Sederhana Untuk Siswa Kelas Iii Uptd Sdn Banyuajuh 4 Kamal,” *Widyagogik J. Pendidik. Dan Pembelajaran Sekol. Dasar*, Vol. 7, No. 2, 2020, Doi: 10.21107/Widyagogik.V7i2.7815.
- [14] C. Cubukcu, M. K. Canbazoglu, And Y. Ozerdem, “Mobile Game Development For Children With Down Syndrome,” *Int. J. Interact. Mob. Technol.*, Vol. 14, No. 20, P. 174, Dec. 2020, Doi: 10.3991/Ijim.V14i20.16573.
- [15] A. Panduwinata, P. P. Arhandi, And B. S. D. Nugraha, “Penerapan Metode Gamifikasi Pada Aplikasi Mosam (Money Sampah) Berbasis Mobile,” *J. Minfo Polgan*, Vol. 12, No. 2, Pp. 1032–1044, 2020, [Online]. Available: <Http://Repota.Jti.Polinema.Ac.Id/Id/Eprint/425>
- [16] M. R. Md Shukri And S. A. Ariffin, “The Development Process Of Awang Sains 2d Digital Mobile Game,” *J. Ict Educ.*, Vol. 6, Pp. 28–36, Jun. 2019, Doi: 10.37134/Jictie.Vol6.3.2019.
- [17] G. Mau, “Rancang Bangun Game 2d Shooter Platformer Menggunakan Metode Finite State Machine,” *Jati (Jurnal Mhs. Tek. Inform.)*, Vol. 3, No. 1, Pp. 117–122, 2019, [Online]. Available: <Https://Ejournal.Itn.Ac.Id/Index.Php/Jati/Article/View/614>
- [18] T. Ardyanto And A. R. Pamungkas, “Pembuatan Game 2d Petualangan Hanoman Berbasis Android,” *J. Go Infotech*, Vol. 23, No. 2, Nov. 2018, Doi: 10.36309/Goi.V23i2.79.
- [19] M. L. Sagala, E. M. A. Jonemaro, And W. S. Wardhono, “Pengembangan Game Platformer 2d Menggunakan Teknik Projection Mapping,” *J. Pengemb. Teknol. Inf. Dan Ilmu Komput. Univ. Brawijaya*, Vol. 1, No. 11, 2017, [Online]. Available: <Https://J-Ptiik.Ub.Ac.Id/Index.Php/J-Ptiik/Article/View/418/173>
- [20] L. Husniah, B. F. Pratama, And H. Wibowo, “Gamification And Gdlc (Game Development Life Cycle) Application For Designing The Sumbawa Folklore Game ”The Legend Of Tanjung Menangis (Crying Cape)”,” *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, Vol. 3, No. 4, Pp. 351–358, 2018, Doi: 10.22219/Kinetik.V3i4.721.
- [21] R. Yanwastika Ariyana, E. Susanti, M. Rizqy Ath-Thaariq, And R. Apriadi, “Insologi: Jurnal Sains Dan Teknologi Penerapan Metode Game Devlopment Life Cycle (Gdlc) Pada Pengembangan Game Motif Batik Khas Yogyakarta,” *Media Cetak*, Vol. 1, No. 6, Pp. 796–807, 2022, Doi: 10.55123/Insologi.V1i6.1129.
- [22] A. R. Hakim, K. Harefa, And B. Widodo, “Pengembangan Sistem Informasi Akademik Berbasis Android Menggunakan Flutter Di Politeknik,” *Scan - J. Teknol. Inf. Dan Komun.*, Vol. 14, No. 3, Pp. 27–32, 2019, Doi: 10.33005/Scan.V14i3.1684.