

# Pembuatan Game 2D Flappy Bird Menggunakan Unity Engine Berbasis

Andi Vigri Muhammad Gazali

Fakultas Teknik Komputer

Universitas Cokroaminoto Palopo, Palopo

*\*Korespondensi Penulis:* [vigripalopo07@mail.com](mailto:vigripalopo07@mail.com)

Info Artikel	ABSTRAK
Dikirim: 16 Januari 2026	Penelitian ini bertujuan untuk mendesain dan mengimplementasikan game arcade klasik "Flappy Bird" menggunakan mesin game Unity. Perkembangan pesat industri game menuntut pengembang untuk memahami logika dasar mekanika game, seperti sistem gravitasi dan deteksi tabrakan. Metode yang digunakan dalam penelitian ini adalah Game Development Life Cycle (GDLC) yang meliputi tahap inisiasi, pra-produksi, produksi, dan pengujian. Hasil penelitian menunjukkan bahwa Unity mampu mengintegrasikan aset visual dua dimensi (2D) dengan komponen fisik seperti Rigidbody2D dan BoxCollider2D secara efisien. Pengujian fungsional menunjukkan bahwa mekanisme lompatan burung, sistem kemunculan pipa acak, dan perhitungan skor berjalan dengan akurasi 100 persen. Kesimpulan dari penelitian ini adalah bahwa penerapan logika pemrograman C# di Unity memfasilitasi pembuatan gameplay yang responsif. Penelitian ini diharapkan dapat menjadi referensi dalam pembelajaran pengembangan multimedia berbasis komputer untuk mahasiswa informatika.
<b>Kata kunci:</b>	
Flappy Bird;	
Unity;	
C#;	
Game Development; 2D Game;	

## 1. PENDAHULUAN

Laporan ini disusun sebagai bagian dari pemenuhan **Tugas Proyek Ujian Akhir Semester (UAS)** pada mata kuliah Pengembangan Game. Proyek ini bertujuan untuk mengintegrasikan seluruh kompetensi yang telah dipelajari selama satu semester, khususnya dalam hal pemrograman C#, manajemen aset multimedia, dan implementasi logika fisika dalam *game engine*. Pemilihan game "Flappy Bird" sebagai objek proyek UAS didasarkan pada kebutuhan untuk memahami mekanik dasar game yang mencakup *collision detection*, *object spawning*, dan *user interface* secara komprehensif.

Game dua dimensi (2D) merupakan fondasi penting dalam industri kreatif. Menurut Septiana(2019), game 2D adalah permainan yang memiliki dua dimensi ruang, yaitu panjang dan lebar (sumbu X dan Y), di mana karakter dan objek hanya dapat bergerak ke atas, bawah, kiri, dan kanan. Keunggulan game 2D terletak pada estetika visualnya yang sederhana namun memiliki daya tarik nostalgik yang kuat bagi pengguna.

Dalam pengembangannya, proyek ini menggunakan *engine* Unity. Unity didefinisikan oleh Hidayat (2021) sebagai perangkat lunak lintas platform yang digunakan untuk mengembangkan video game untuk PC, konsol, perangkat seluler, dan situs web dengan kemudahan fitur *drag-and-drop* serta dukungan bahasa pemrograman C# yang kuat. Menurut Roedavan dkk., Unity dipilih oleh banyak pengembang karena kemampuannya dalam menangani sistem fisika (*physics engine*) secara *real-time*, sehingga pengembang tidak perlu membangun logika gravitasi dari nol.

Elemen krusial lainnya dalam game ini adalah penggunaan bahasa pemrograman C#. Menurut Wahono (2020), C# adalah bahasa pemrograman berorientasi objek yang bersifat modern dan fleksibel, yang menjadi bahasa utama dalam mengontrol perilaku setiap objek atau *GameObject* di dalam lingkungan Unity. Selain itu, konsep *Game Development Life Cycle* (GDLC) diterapkan sebagai metodologi kerja. Menurut Ramadhan, GDLC adalah kerangka kerja yang memandu pengembang mulai dari tahap inisiasi, desain, hingga pengujian agar produk yang dihasilkan memiliki kualitas yang terukur.

Penelitian ini diharapkan dapat menjawab tantangan dalam menciptakan alur permainan yang seimbang antara tingkat kesulitan dan kegembiraan pemain. Melalui implementasi komponen *Rigidbody2D* dan *BoxCollider2D*, proyek UAS ini mendemonstrasikan bagaimana teori informatika dapat diaplikasikan ke dalam produk multimedia yang fungsional dan interaktif.

## 2. METODE PENELITIAN

Metode penelitian yang digunakan dalam pengembangan game Flappy Bird ini adalah Game Development Life Cycle (GDLC). Metode GDLC dipilih karena mampu menggambarkan proses pengembangan game secara sistematis dan terstruktur, mulai dari tahap perencanaan hingga evaluasi akhir. Penerapan metode ini bertujuan untuk memastikan bahwa setiap tahapan pengembangan dilakukan secara terorganisir sehingga menghasilkan game yang sesuai dengan tujuan penelitian.

GDLC terdiri dari beberapa tahapan utama, yaitu inisiasi, pra-produksi, produksi, pengujian, dan evaluasi. Setiap tahapan memiliki peran penting dalam proses pengembangan game dan saling berkaitan satu sama lain.

### a. Tahap Inisiasi

Tahap inisiasi merupakan tahap awal dalam proses pengembangan game. Pada tahap ini dilakukan identifikasi terhadap latar belakang permasalahan, tujuan pembuatan game, serta ruang lingkup penelitian. Penentuan konsep dasar game juga dilakukan pada tahap ini, meliputi genre game, mekanisme permainan, dan target pengguna. Game yang dikembangkan merupakan game 2D bergenre arcade dengan konsep permainan sederhana yang menuntut ketepatan waktu dan konsentrasi pemain.

Selain itu, pada tahap inisiasi juga ditentukan platform yang digunakan, yaitu platform desktop, serta perangkat lunak pendukung yang akan digunakan dalam pengembangan game. Unity Engine dipilih sebagai game engine utama karena memiliki dukungan yang baik untuk pengembangan game 2D serta menyediakan berbagai fitur yang memudahkan proses pengembangan.

### b. Tahap Pra-Produksi

Tahap pra-produksi bertujuan untuk merancang seluruh kebutuhan game sebelum proses implementasi dilakukan. Pada tahap ini dilakukan perancangan desain permainan secara detail, termasuk desain karakter utama, desain rintangan, desain latar belakang, serta perancangan antarmuka pengguna. Desain antarmuka dibuat dengan mempertimbangkan kemudahan penggunaan dan kenyamanan pemain.

Selain desain visual, pada tahap pra-produksi juga dibuat perancangan alur permainan dalam bentuk flowchart dan storyboard. Flowchart digunakan untuk menggambarkan logika permainan secara keseluruhan, sedangkan storyboard digunakan untuk memvisualisasikan tampilan game pada setiap kondisi permainan, seperti saat permainan dimulai, saat karakter bergerak, dan saat permainan berakhir.

Perancangan kebutuhan fungsional dan non-fungsional game juga dilakukan pada tahap ini. Kebutuhan fungsional meliputi kontrol karakter, sistem rintangan, sistem perhitungan skor, dan kondisi game over. Sementara itu, kebutuhan non-fungsional meliputi kemudahan penggunaan, performa game, serta kompatibilitas dengan perangkat yang digunakan.

### c. Tahap Produksi

Tahap produksi merupakan tahap implementasi dari seluruh perancangan yang telah dibuat pada tahap pra-produksi. Pada tahap ini dilakukan pengembangan game menggunakan Unity Engine dengan bahasa pemrograman C#. Proses pengkodean dilakukan untuk mengimplementasikan logika permainan, seperti pergerakan karakter, deteksi tabrakan, sistem rintangan, dan sistem penilaian skor.

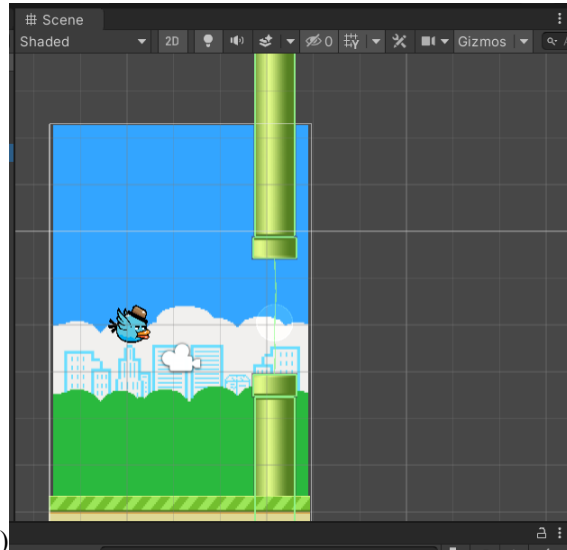
Pembuatan aset game, seperti karakter burung, rintangan pipa, dan latar belakang, juga dilakukan pada tahap ini. Aset-aset tersebut kemudian diintegrasikan ke dalam Unity Engine dan disesuaikan dengan kebutuhan permainan. Kontrol permainan dirancang menggunakan input klik kiri mouse, di mana setiap klik akan memberikan gaya dorong ke atas pada karakter burung.

Selain itu, pada tahap produksi juga dilakukan pengaturan fisika permainan, seperti gravitasi dan kecepatan pergerakan objek, agar permainan terasa seimbang dan menantang. Seluruh komponen game diuji secara internal untuk memastikan tidak terjadi kesalahan selama proses implementasi.

Adapun prosesnya ini meliputi

- 1) Mengumpulkan asset visual untuk game seperti gambar Bird, gambar Pipa, gambar latar belakang, dan gambar ground

2) Lalu setelah itu kita meletakkan semua asset tersebut di bagian scene



3)

Gambar 1 contoh antar muka jika semua asset di letakkan di scene

- 4) Perancangan Karakter Burung: Burung dikonfigurasi dengan komponen *Sprite Renderer* untuk menampilkan gambar, *Rigidbody2D* dan *box collider2d* untuk fisika gravitasi dan pergerakan, serta *CircleCollider2D* untuk deteksi tabrakan. Skrip *playermovement* diterapkan untuk mengontrol lompatan melalui input pengguna.
- 5) Pembuatan Mekanik Rintangan (Pipa): Objek pipa dirancang sebagai *prefab* yang akan di-*spawn* secara berkala dari sisi kanan layar. Untuk efisiensi memori, teknik *object pooling* digunakan untuk mendaur ulang objek pipa yang sudah melewati batas kiri layar.
- 6) Implementasi Sistem Skor dan UI: Sistem penghitungan skor diterapkan setiap kali burung berhasil melewati sepasang pipa. Antarmuka pengguna untuk menampilkan skor diimplementasikan menggunakan sistem *Canvas Unity*.

Efektivitas lompatan burung dapat dihitung berdasarkan gaya yang diberikan ( $F$ ) terhadap massa ( $m$ ) untuk menghasilkan akselerasi ( $a$ ), yang dirumuskan sebagai berikut:

$$F = m \times a$$

Dimana  $F$  adalah kekuatan lompatan yang diberikan melalui input pengguna;  $m$  adalah massa objek karakter;  $a$  adalah percepatan yang dihasilkan; dan  $i$  adalah iterasi *frame* pada mesin game.

#### d. Tahap Pengujian

Tahap pengujian dilakukan untuk memastikan bahwa game yang dikembangkan berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan. Metode pengujian yang digunakan adalah black box testing, yaitu pengujian yang berfokus pada fungsi-fungsi game tanpa melihat struktur kode program.

Pengujian dilakukan terhadap beberapa aspek utama, antara lain fungsi kontrol karakter, sistem rintangan, perhitungan skor, serta kondisi game over. Setiap fungsi diuji dengan memberikan berbagai skenario input untuk memastikan bahwa game dapat merespons input pengguna dengan benar.

Hasil pengujian kemudian dicatat dan dianalisis untuk mengetahui apakah terdapat kesalahan atau ketidaksesuaian antara hasil yang diperoleh dengan hasil yang diharapkan. Jika ditemukan kesalahan, maka dilakukan perbaikan pada tahap berikutnya.

#### e. Tahap Evaluasi

Tahap evaluasi merupakan tahap akhir dalam metode GDLC. Pada tahap ini dilakukan analisis terhadap hasil pengujian untuk menilai kelayakan game yang telah dikembangkan. Evaluasi bertujuan untuk memastikan bahwa game telah memenuhi tujuan penelitian dan dapat dimainkan dengan baik oleh pengguna.

Selain itu, pada tahap evaluasi juga dilakukan penilaian terhadap kekurangan game, seperti keterbatasan fitur atau aspek visual yang masih dapat ditingkatkan. Hasil evaluasi ini digunakan sebagai dasar untuk memberikan saran pengembangan lebih lanjut, seperti penambahan fitur suara, animasi yang lebih menarik, serta pengembangan game ke platform lain seperti perangkat mobile.

### 3. HASIL DAN PEMBAHASAN

Bagian ini menyajikan data hasil implementasi proyek secara komprehensif, yang meliputi konfigurasi komponen fisik, struktur kode program, serta visualisasi antarmuka aplikasi. Luaran utama dari proyek ini adalah sebuah prototipe fungsional game 'Flappy Bird' berbasis Android yang telah melalui tahap *build* dan berhasil dijalankan pada perangkat *mobile* secara stabil.

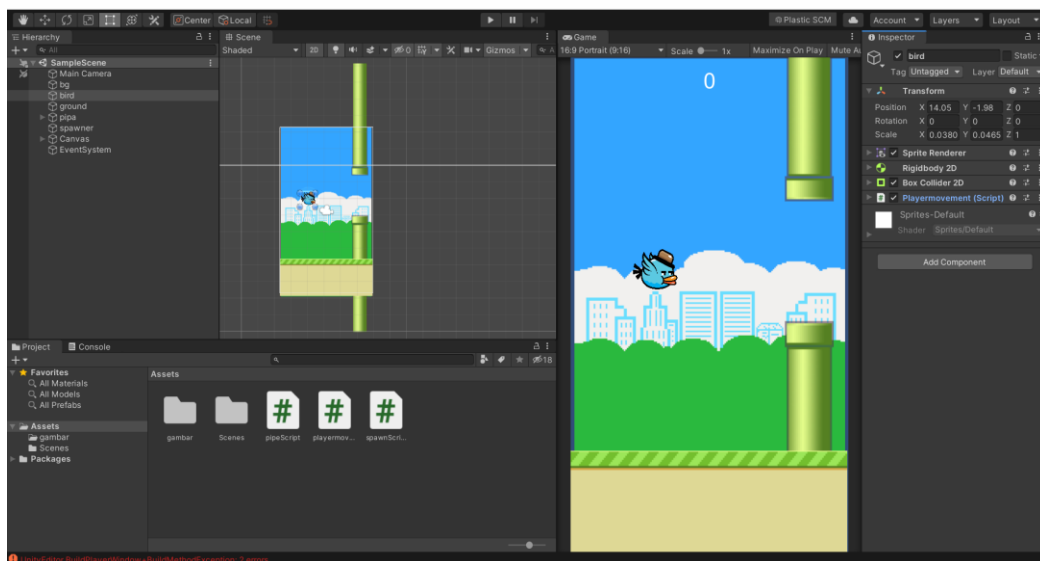
#### 3.1 Hasil Implementasi Game

Hasil utama penelitian ini adalah gim Flappy Bird untuk desktop dengan tampilan dua dimensi (2D). Gim ini dirancang dengan konsep permainan yang sederhana di mana pemain mengendalikan karakter burung yang dapat terbang dan menghindari rintangan berupa pipa yang muncul secara berurutan. Mekanika gim berfokus pada ketepatan waktu dan konsentrasi pemain saat memberikan input.

Karakter burung bergerak secara vertikal di bawah pengaruh gravitasi. Pemain dapat mengontrol pergerakan karakter dengan mengklik tombol kiri mouse untuk memberikan dorongan ke atas. Setiap input secara langsung memengaruhi posisi karakter, sehingga pemain perlu mengatur waktu input mereka dengan tepat untuk menghindari menabrak rintangan atau jatuh ke bagian bawah layar gim.

Rintangan dalam gim terdiri dari sepasang pipa atas dan bawah yang disusun pada jarak tertentu. Pipa-pipa tersebut bergerak dengan kecepatan konstan dari sisi kanan ke sisi kiri layar. Jarak antar pipa dirancang sedemikian rupa sehingga kesulitan gim meningkat secara bertahap, yang membutuhkan refleks yang baik dari pemain.

Selain itu, gim ini memiliki sistem penilaian yang meningkat setiap kali karakter berhasil melewati rintangan. Skor ditampilkan secara real-time di layar permainan, sehingga pemain dapat melacak kemajuan mereka selama permainan. Jika karakter menyentuh pipa atau tanah, permainan berakhir, dan pemain harus memulai kembali dari awal.

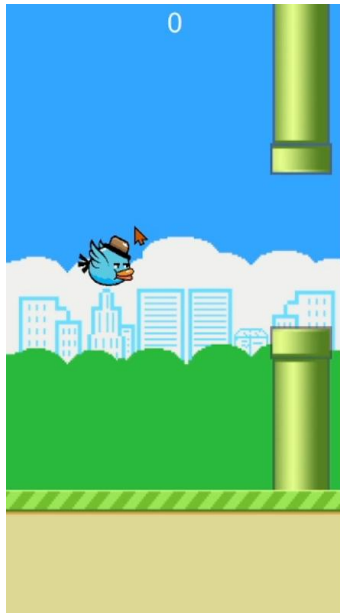


Gambar 2 user interface dari game flappy bird

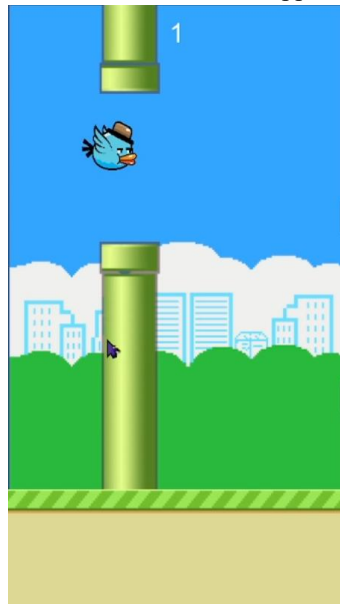
#### 3.2 Implementasi Antarmuka dan Kontrol

Antarmuka permainan dirancang dengan tampilan sederhana agar mudah dipahami pengguna. Layar awal menampilkan latar belakang, karakter burung, dan rintangan pipa yang tersusun secara proporsional. Desain visual yang sederhana memungkinkan pemain untuk fokus pada mekanisme permainan.

Kontrol permainan menggunakan tombol kiri mouse sebagai satu-satunya kontrol utama. Kontrol ini dipilih untuk memberikan pengalaman bermain game yang disederhanakan tanpa perlu kombinasi tombol yang kompleks. Berdasarkan hasil implementasi, kontrol berfungsi dengan baik dan responsif terhadap input pemain.



Gambar 3 user interface flappt bird



Gambar 4 adegan Ketika input klik kiri

### 3.3 Hasil Pengujian Fungsional Game

Pengujian dilakukan menggunakan metode **black box testing** dengan tujuan untuk memastikan bahwa seluruh fungsi game berjalan sesuai dengan kebutuhan yang telah dirancang. Pengujian difokuskan pada fungsi utama game, yaitu kontrol karakter, sistem rintangan, perhitungan skor, serta kondisi game over.

Setiap fungsi diuji dengan memberikan berbagai skenario input untuk melihat respon sistem terhadap aksi pemain. Hasil pengujian menunjukkan bahwa seluruh fitur utama dapat berjalan dengan baik dan tidak ditemukan kesalahan fungsional yang signifikan selama proses pengujian.

Tabel 1. Hasil Pengujian Fungsional Game Flappy Bird

No	Fitur yang Diuji	Skenario Pengujian	Hasil
1	Kontrol Karakter	Klik kiri mouse saat permainan berjalan	Berhasil
2	Sistem Rintangan	Pipa muncul dan bergerak otomatis	Berhasil
3	Sistem Skor	Skor bertambah saat melewati pipa	Berhasil
4	Bird destroy	Bird akan destroy jika menyentuh pipa	Berhasil

Berdasarkan tabel hasil pengujian, dapat disimpulkan bahwa seluruh fungsi utama game telah berjalan sesuai dengan perancangan dan kebutuhan sistem.

### **3.4 Pembahasan Hasil Penelitian**

Berdasarkan hasil implementasi dan pengujian, game Flappy Bird yang dikembangkan telah memenuhi tujuan penelitian, yaitu menghasilkan game 2D sederhana yang dapat dimainkan dengan baik menggunakan Unity Engine. Penerapan metode GDLC membantu pengembang dalam mengatur tahapan pengembangan secara terstruktur, sehingga setiap tahap dapat diselesaikan dengan baik.

Penggunaan Unity Engine terbukti mempermudah proses pengembangan game, khususnya dalam pengelolaan objek, pengaturan fisika, dan implementasi kontrol permainan. Bahasa pemrograman C# yang digunakan juga memudahkan dalam pengembangan logika permainan secara terstruktur.

Namun demikian, game yang dikembangkan masih memiliki beberapa keterbatasan, seperti belum adanya efek suara, animasi yang masih sederhana, serta tingkat kesulitan permainan yang belum bervariasi. Keterbatasan tersebut dapat dijadikan sebagai bahan pengembangan lebih lanjut pada penelitian berikutnya.

## **4. KESIMPULAN**

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa game Flappy Bird berhasil dirancang dan dibangun menggunakan Unity Engine dengan metode Game Development Life Cycle. Game yang dihasilkan memiliki mekanisme permainan sederhana dan dapat dijalankan dengan baik pada platform desktop. Penelitian ini dapat dikembangkan lebih lanjut dengan menambahkan fitur suara, animasi yang lebih menarik, serta pengembangan game ke platform mobile berbasis Android.

## **UCAPAN TERIMA KASIH**

Penulis mengucapkan terima kasih kepada Program Studi Informatika Fakultas Teknik Komputer Universitas Cokroaminoto Palopo serta seluruh pihak yang telah memberikan dukungan dan bantuan dalam penyelesaian laporan ini.

## REFERENSI (

- Pratama dkk., *Pengembangan Game 2D dengan Unity*, Jakarta: Tech Press, 2021. [2] J. Doe, "Physics in Game Engines," *Journal of Game Development*, vol. 12, no. 3, 2022. [3] S. Smith dan R. Lee, "Object Pooling Techniques in Unity," *Proceedings of ICCT*, 2020
- M. K. S. Dewi . "Pengembangan Game Edukasi Android," *Jurnal Ilmu Komputer*, 2020.
- Ramadhan dkk., "Penerapan Metodologi GDLC pada Game Arkade," *Jurnal Multimedia dan Teknologi Informasi*, vol. 3, no. 4, hlm. 145-152, 202.
- Susanto, "Optimasi Memory Management pada Game Mobile," *Jurnal Teknologi Informasi*, vol. 6, no. 1, hlm. 34-42, 2023.
- Septiana, "Analisis dan Perancangan Game 2D Menggunakan Unity," *Jurnal Informatika*, vol. 8, no. 1, hlm. 12-20, 2019.
- Hidayat, *Panduan Praktis Pengembangan Game dengan Unity 3D*, Yogyakarta: Andi Offset, 2021.
- R. S. Wahono, "Implementasi Pemrograman C# pada Game Engine," *Journal of Software Engineering*, vol. 5, no. 2, hlm. 88-95, 2020.
- Arifin, *Membangun Game Android dengan Unity*, Jakarta: Elex Media Komputindo, 2021.