

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Perkembangan teknologi digunakan dalam berbagai bidang termasuk bidang *entertainment*, salah satunya adalah dalam bentuk *game*. *Game* merupakan kegiatan yang kompleks yang di dalamnya terdapat peraturan, *play* dan *action*, oleh sebab itu *game* memiliki daya tarik tersendiri<sup>[1]</sup>.

*Game* adalah sarana hiburan yang sangat menyenangkan dan memberi efek perasaan gembira saat memainkannya, kepuasan jika pemain menang atau mendapat penghargaan dalam bentuk item atau naik *level*, bahkan sampai memberi efek candu pada pemain, *game* dalam segi grafis ada yang 2D (dua dimensi) dan 3D (tiga dimensi). Berkat kemajuan teknologi banyak sekali *game* yang menggunakan teknologi 3D, selain itu membuat *game* 3D sekarang tidaklah sulit. Saat ini, banyak *game engine* yang bisa digunakan untuk membantu pembuatan *game* salah satu contoh *game engine* yang terkenal adalah *Unity*<sup>[2]</sup>.

Kecerdasan buatan adalah salah satu ilmu komputer yang mempelajari bagaimana membuat suatu sistem dapat bekerja seperti kecerdasan manusia atau bahkan lebih baik. Kecerdasan buatan dapat diterapkan di berbagai ilmu komputer seperti *microcontroller*, *game*, *network* dan aplikasi lainnya. Salah satu unsur terpenting dalam *game* adalah kecerdasan buatan yang dapat memberikan respon sebaik mungkin atau sealam mungkin. Penerapan kecerdasan buatan dalam *game* dapat dimasukkan dalam berbagai aspek seperti pada karakter utama, karakter musuh, NPC (*Non Playable Character*) dan sistem penentuan reward dalam *game* serta proses kenaikan level dan sebagainya. Banyak metode yang dapat diterapkan dalam sebuah *game* untuk membangun sebuah kecerdasan buatan seperti *Collision Detection*<sup>[3]</sup>.

*Unity 3D* adalah aplikasi pembuat *game engine* yang bermultiplatform, *Unity* dipublikasi menjadi *standalone* (.exe), berbasis *web*, *Android*, *IOS*, *XBOX*, dan *PS3*. Walau bisa digunakan ke berbagai platform, *Unity* perlu lisensi untuk dapat digunakan ke platform tertentu. Namun, *Unity* menyediakan untuk *free user*

dan bisa dipublikasi dalam bentuk *standalone* (.exe) dan *web*. Unity 3D adalah sebuah aplikasi yang digunakan untuk membuat *game* atau sering juga disebut dengan *game engine*. Pada dasarnya *Unity* merupakan *game engine* yang berbasis 3D, namun *Unity* juga bisa digunakan untuk membuat *game* 2D <sup>[4]</sup>.

*Game save the red bird* itu sendiri adalah *game* di mana pemain mengetuk jari ke layar smartpone untuk menerbangkan burung, dengan menjaga burung agar tidak menabrak objek yang ada didepan nya, hampir mirip *game flappy bird* dimana *game* ini sedikit perbedaan objek yang harus di hindari adalah burung yang terbang lurus secara acak (AI) mengarah ke *red bird* (player) dengan teknik kecerdasan buatan atau *Artificial Intelligence* dan di implementasi kan dengan Algoritma *Collision Detection*.

*Android* adalah sebuah sistem operasi yang dirancang oleh perusahaan *Google* dengan basis kernel *Linux* dan juga berbagai perangkat lunak seperti *Open Source* dan lainnya <sup>[5]</sup>. Ponsel yang menggunakan *Android* dapat digunakan untuk perangkat dengan layar sentuh seperti pada *smartphone* dan juga komputer tablet. *Android* memiliki perbedaan dengan sistem operasi *Windows 10* milik *Microsoft* yang mengharuskan perusahaan untuk membayar *royalty* apabila ingin menggunakan sistem operasi tersebut. Maka dari itu, tidak heran jika banyak *vendor* yang menggunakan sistem operasi yang bekerja sama dengan *Google* ini. Sedangkan untuk sistem operasi seperti *iOS* milik perusahaan *Apple* hanya dapat digunakan pada perangkat *iPhone* dan juga *iPad* <sup>[6]</sup>.

Algoritma *Collision Detection* merupakan proses pengecekan apakah buah objek spasial saling bertumpuk atau tidak, jika ternyata ada paling sedikit dua buah objek yang bertumpuk, maka kedua objek tersebut dikatakan saling bertumpukkan, *Collision Detection* merupakan teknik deteksi tabrakan untuk mengetahui objek-objek apa saja yang bersentuhan dalam bidang koordinat tertentu, kelebihan algoritma ini menggunakan sensor pemicu (*triger*) untuk mendeteksi keberadaan objek ketika berada di lingkungan *virtual* dengan mekanisme pemicu pendukung fitur ini adalah sensor <sup>[7]</sup>.

Dalam penelitian skripsi ini akan dibahas perancangan sebuah aplikasi *game* penyelamatan burung merah berbasis mobile *Android* yang bertujuan sebagai

sarana melatih ketangkasan dengan judul “Implementasi Algoritma *Collision Detection* Pada Game *Save The Red Bird* Berbasis *Android* Menggunakan *Unity 3D*”. Dalam penelitian akan membuat *game* dengan dasar hiburan, namun juga bertujuan sebagai sarana melatih ketangkasan dengan penerapan algoritma *Collision Detection* untuk deteksi tabrakan atau objek yang saling bertumpuk.

Adapun beberapa penelitian yang dijadikan sebagai referensi oleh penulis adalah penelitian yang dilakukan oleh<sup>[8]</sup> yang berjudul Implementasi Algoritma *Collision Detection* Pada *Game* TPS Berbasis *Android* Menggunakan *Unity 3D*. Penelitian yang dilakukan oleh<sup>[9]</sup> yang berjudul Implementasi Algoritma *Collision Detection* Pada *Game* Simulator Driving Car. Penelitian yang dilakukan oleh<sup>[10]</sup> yang berjudul Rancang Bangun *Game Android* Adventure Finding Diamond Dengan *Unity 3D* Menggunakan Metode Dynamic Weighting A. Penelitian yang dilakukan oleh<sup>[3]</sup> yang berjudul Implementasi Algoritma *Collision Detection* Dan Finite State Machine Untuk Karakter Musuh Pada *Game* Bertipe *Metroidvania*. Penelitian yang dilakukan oleh<sup>[11]</sup> yang berjudul Pembuatan *Game Rpg* (Role Playing *Game*) Quiz Menggunakan Algoritma A\* Dan Perlin Noise Dengan *Game Engine* *Unity 5*.

Berdasarkan latar belakang di atas, maka penulis tertarik untuk membuat aplikasi yang berjudul **Implementasi Algoritma *Collision Detection* Pada *Game* *Save The Red Bird* Berbasis *Android* Menggunakan *Unity 3D***.

## 1.2. Rumusan Masalah

Berdasarkan latar belakang tersebut maka penulis membuat rumusan masalah yaitu sebagai berikut:

1. Bagaimana cara membangun *game* 3D dengan *Unity* berbasis *Android*?
2. Bagaimana cara penerapan algoritma *Collision Detection* pada *Game* *Save The Red Bird* berbasis *Android* menggunakan *unity 3D*?

## 1.3 Batasan Masalah

Adapun batasan dari permasalahan ini adalah sebagai berikut:

1. Aplikasi dibangun menggunakan Unity editor 2017 dan *Android* SDK tool 29.
2. Pada *game* yang di buat pemain hanya bisa klik layar saja untuk menggerakkan burung dan menghindari tabrakan lawan.
3. *Game* yang dibuat bersifat *single player*, *offline*, dan dengan tampilan 2D.

## **1.4 Tujuan dan Manfaat Penelitian**

### **1.4.1 Tujuan Penelitian**

Tujuan penelitian ini untuk membangun, merancang dan mengimplementasikan algoritma *Collision Detection* pada *Game Save The Red Bird* berbasis *Android* menggunakan Unity 3D.

### **1.4.2. Manfaat Penelitian**

Adapun manfaat dari penelitian ini adalah :

1. Menyajikan suatu *game* hiburan simpel yang menarik untuk dimainkan.
2. Pengaplikasian fungsi kinerja algoritma *Collision Detection* pada *game save the red bird*.

## **1.5 Sistematika Penulisan**

Sistematika penulisan dilihat untuk memberikan gambaran umum mengenai penelitian yang dilakukan dan kejelasan mengenai penulisan hasil penelitian. Laporan hasil penelitian ini ditulis dengan sistematika sebagai berikut :

### **BAB I PENDAHULUAN**

Bab ini berisikan latar belakang rumusan masalah, batasan masalah, tujuan dan manfaat penelitian dan sistematika penulisan.

### **BAB II LANDASAN TEORI**

Bab ini membahas mengenai berbagai macam landasan teori yang digunakan dan sesuai dengan kebutuhan, sejarah *Android*, dan *software* atau *development tools* yang

digunakan untuk pengembangan aplikasi ini.

### **BAB III METODOLOGI PENELITIAN**

Bab ini berisikan metodologi penelitian terdiri dari 3 bagian utama yaitu model pengembangan perangkat lunak, metode penelitian, dan *tools* (alat bantu dalam analisis dan merancang aplikasi).

### **BAB IV HASIL DAN PEMBAHASAN**

Pada bab ini berisi antara analisis masalah sistem yang berjalan, analisis hasil solusi, analisis sistem, perancangan sistem, rancangan layar, tampilan layar.

### **BAB V PENUTUP**

Bab ini berisikan kesimpulan mengenai apa saja yang telah dihasilkan dan saran-saran alternatif yang dapat diterapkan untuk meningkatkan pengembangan aplikasi.

