

Rhythm Hero: Game Musik Guitar Hero–Like Berbasis Object-Programming Menggunakan Pygame



Oleh

Muhammad Nadhiful Asyror

24091397044

2024B

**PROGRAM STUDI D4 MANAJEMEN INFORMATIKA
FAKULTAS VOKASI
UNIVERSITAS NEGERI SURABAYA
2025**

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemrograman Berorientasi Objek (Object-Oriented Programming/OOP) merupakan paradigma pemrograman yang menekankan pada penggunaan objek sebagai representasi data dan perilaku. Konsep OOP seperti encapsulation, inheritance, dan polymorphism banyak digunakan dalam pengembangan aplikasi modern karena mampu menghasilkan kode yang modular, mudah dipahami, dan mudah dikembangkan.

Dalam proyek UAS mata kuliah Pemrograman Berorientasi Objek ini, dikembangkan sebuah aplikasi game sederhana berbasis Python dan Pygame dengan konsep permainan musik mirip Guitar Hero. Game ini dipilih karena mampu menerapkan konsep OOP secara jelas serta memiliki interaksi pengguna yang menarik.

1.2 Tujuan

Tujuan dari pembuatan aplikasi ini adalah:

1. Menerapkan konsep Object-Oriented Programming dalam sebuah aplikasi nyata.
2. Mengimplementasikan prinsip encapsulation, inheritance, dan polymorphism.
3. Mengembangkan game sederhana berbasis Pygame dengan sistem lane, scoring, dan game over.
4. Menyediakan dokumentasi dan desain sistem yang sesuai standar OOP.

1.3 Deskripsi Singkat Aplikasi

Rhythm Hero adalah game musik di mana pemain harus menekan tombol A, S, D, dan F sesuai dengan note yang jatuh pada lane tertentu. Pemain mendapatkan skor berdasarkan ketepatan waktu menekan tombol, dengan kategori Perfect, Good, dan Miss. Game akan berakhir ketika jumlah miss mencapai batas tertentu dan menampilkan statistik hasil permainan.

BAB II

ANALISIS DAN PERANCANGAN SISTEM

2.1 Analisis Sistem

Aplikasi ini merupakan game single-player berbasis desktop yang dikembangkan menggunakan Python dan library Pygame. Sistem dirancang menggunakan pendekatan OOP dengan pemisahan tanggung jawab setiap class agar kode lebih terstruktur.

2.2 Class Diagram

Aplikasi ini terdiri dari beberapa class utama:

- GameObject (superclass)
- Note
- SpecialNote
- Player
- ScoreManager
- GameManager

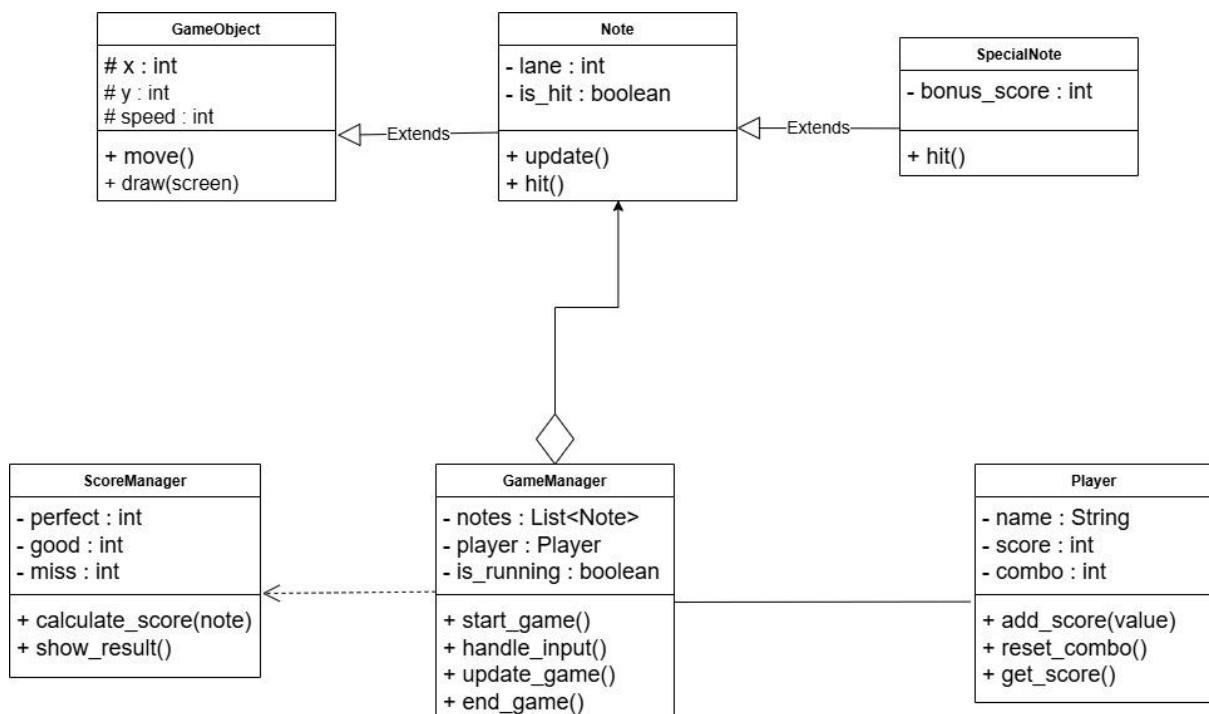
Class diagram menunjukkan hubungan inheritance antara GameObject, Note, dan SpecialNote, serta hubungan association dan aggregation dengan GameManager.

2.3 Activity Diagram

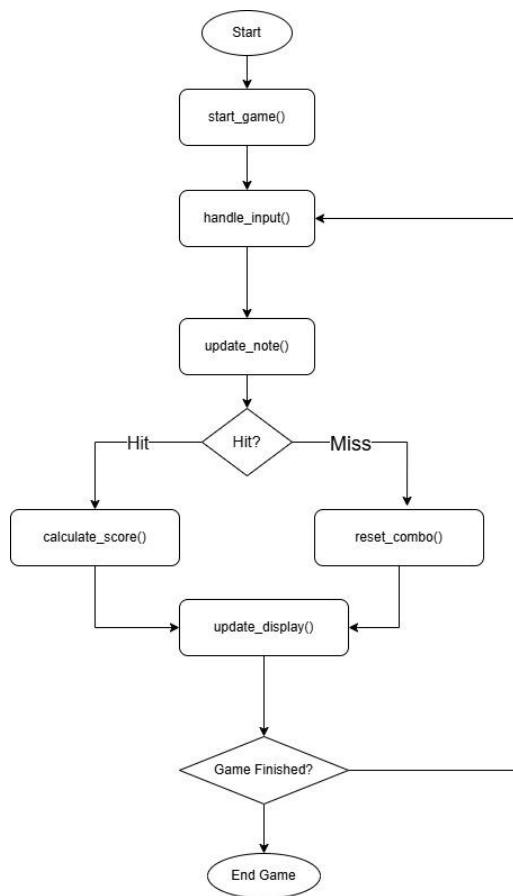
Activity diagram menggambarkan alur permainan dimulai dari start game, gameplay loop (input, update note, hit detection), hingga game over dan penampilan statistik akhir.

2.4 Relasi Antar Class

- Inheritance: Note mewarisi GameObject, SpecialNote mewarisi Note
- Association: GameManager berasosiasi dengan Player
- Aggregation: GameManager mengelola kumpulan Note
- Dependency: GameManager menggunakan ScoreManager



Gambar 1.1 Diagram Class



Gambar 1.2 Activity Diagram

BAB III

IMPLEMENTASI OBJECT-ORIENTED PROGRAMMING

3.1 Encapsulation

Encapsulation diterapkan dengan penggunaan atribut private dan protected pada setiap class.

Contoh:

- Atribut `_score` dan `_combo` pada class `Player` bersifat private.
- Atribut `_x`, `_y`, dan `_speed` pada `GameObject` bersifat protected.

Encapsulation bertujuan untuk melindungi data dan mengontrol akses terhadap atribut objek.

3.2 Inheritance

Inheritance digunakan untuk menghindari duplikasi kode dan memperluas fungsionalitas class.

Contoh:

- Class `Note` mewarisi atribut dan method dari `GameObject`.
- Class `SpecialNote` mewarisi `Note` dan menambahkan perilaku khusus.

3.3 Polymorphism

Polymorphism diterapkan melalui method overriding pada method `hit()`.

Contoh:

- `Note.hit()` memberikan skor normal.
- `SpecialNote.hit()` memberikan skor bonus.

Dengan polymorphism, method yang sama dapat memiliki perilaku berbeda tergantung objek yang memanggilnya.

BAB IV

IMPLEMENTASI FITUR APLIKASI

4.1 Sistem Lane Guitar Hero

Game memiliki 4 lane yang dihubungkan dengan tombol:

- A → Lane 1
- S → Lane 2
- D → Lane 3
- F → Lane 4

Setiap note memiliki lane tertentu dan hanya dapat dihitung sebagai hit jika tombol yang ditekan sesuai.

4.2 Hit Detection dan Timing

Hit detection dilakukan dengan menghitung jarak antara posisi note dengan garis hit. Hasil timing dibagi menjadi:

- **Perfect:** jarak sangat dekat
- **Good:** jarak cukup dekat
- **Miss:** jarak terlalu jauh atau note terlewat

4.3 Sistem Skor dan Combo

- Perfect memberikan skor tertinggi.
- Good memberikan skor lebih rendah.
- Miss mereset combo pemain.
- Combo bertambah setiap hit berhasil.

4.4 Game Over dan Statistik

Game akan berakhir ketika jumlah miss mencapai batas maksimum. Saat game over, sistem menampilkan:

- Total skor
- Jumlah Perfect
- Jumlah Good
- Jumlah Miss

BAB V

PENGUJIAN APLIKASI

5.1 Skenario Pengujian

No	Skenario	Hasil
1	Menjalankan game	Game berjalan normal
2	Menekan tombol sesuai lane	Skor bertambah
3	Timing tepat	Perfect / Good muncul
4	Note terlewat	Miss bertambah
5	Miss mencapai batas	Game Over

5.2 Hasil Pengujian

Berdasarkan hasil pengujian, aplikasi berjalan sesuai dengan rancangan. Semua fitur utama dapat digunakan dengan baik dan tidak ditemukan error fatal selama pengujian.

BAB VI

KESIMPULAN

6.1 Kesimpulan

Aplikasi Rhythm Hero berhasil dikembangkan dengan menerapkan konsep Object-Oriented Programming secara lengkap. Prinsip encapsulation, inheritance, dan polymorphism telah diterapkan dengan baik. Game berjalan sesuai fungsionalitas yang direncanakan dan memiliki antarmuka yang interaktif.

6.2 Saran Pengembangan

Pengembangan selanjutnya dapat dilakukan dengan menambahkan:

- Sinkronisasi musik dan beat map
- Mode permainan (Easy, Normal, Hard)
- Sistem leaderboard

Link repositori:

<https://github.com/Nadhiful-asyror/Rhythm-Hero.git>