**LAPORAN PRAKTIKUM**

**PEMROGRAMAN BERORIENTASI OBJEK**

**MODUL VIII**

**INHERITANCE II**

Logo

Description automatically generated with medium confidence

**Disusun Oleh:**

**Adri Hidayat 105223046**

**PROGRAM STUDI ILMU KOMPUTER**

**FAKULTAS SAINS DAN KOMPUTER**

**UNIVERSITAS PERTAMINA**

**2025**

1. **Pendahuluan**

Perkembangan teknologi dan kebutuhan hiburan digital mendorong pengembangan berbagai jenis permainan komputer, termasuk game berbasis teks yang sederhana namun menarik untuk dipelajari. Game berbasis teks memungkinkan interaksi tanpa grafis kompleks, sehingga cocok sebagai media pembelajaran dasar pemrograman dan logika game. Proyek ini bertujuan membuat game petualangan berbasis teks dengan fitur registrasi, login, serta mekanisme eksplorasi, pertarungan, dan pengelolaan status pemain menggunakan bahasa pemrograman Java. Dengan struktur modular yang memisahkan tanggung jawab tiap kelas, pengembangan dan pemeliharaan kode menjadi lebih terorganisir. Melalui proyek ini diharapkan dapat memahami konsep pemrograman berorientasi objek, interaksi input-output teks, serta penerapan logika permainan sederhana namun fungsional.

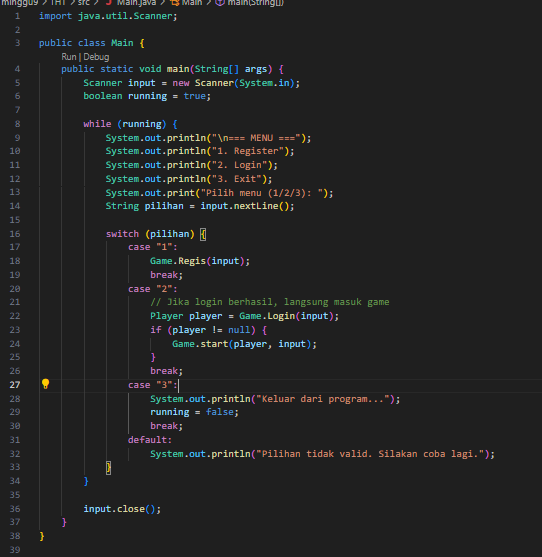
1. **Variabel**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Variabel** | **Tipe data** | **Fungsi** |
| 1 | input | Scanner | Menerima input dari pengguna |
| 2 | running | boolean | Mengontrol perulangan menu utama di Main |
| 3 | pilihan | String | Menyimpan pilihan menu dari pengguna |
| 4 | usr | ArrayList<String> | Menyimpan daftar username yang terdaftar |
| 5 | pss | ArrayList<String> | Menyimpan daftar password yang terdaftar |
| 6 | players | ArrayList<Player> | Menyimpan objek Player yang terdaftar sesuai username |
| 7 | Username | String | Menyimpan input username dari pengguna saat login/register |
| 8 | Password | String | Menyimpan input password dari pengguna saat login/register |
| 9 | player | Player | Menyimpan objek Player yang berhasil login |
| 10 | playing | boolean | Mengontrol perulangan menu dalam game loop (start()) |
| 11 | choice | String | Menyimpan input pengguna saat memilih aksi dalam game |

1. **Constructor dan Method**

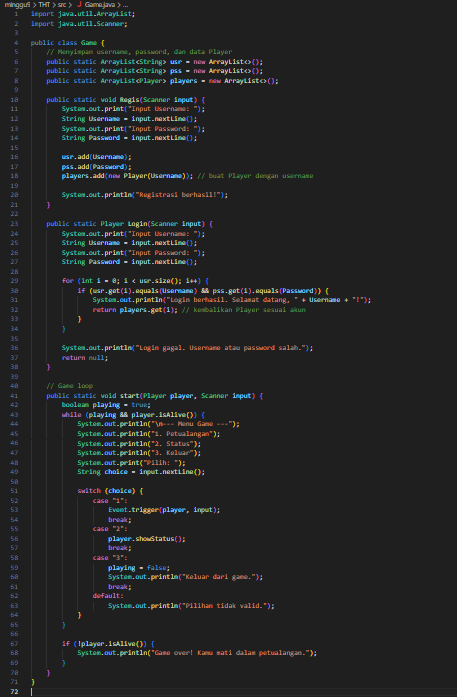
|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Metode** | **Jenis Metode** | **Fungsi** |
| 1 | main() | static | Mengatur alur utama program dan menampilkan menu |
| 2 | Regis(Scanner input) | static | Mengatur proses registrasi user dan menyimpan data username, password, dan Player baru |
| 3 | Login(Scanner input) | static | Memverifikasi username dan password, lalu mengembalikan objek Player jika berhasil login |
| 4 | start(Player player, Scanner input) | static | Memulai loop permainan setelah login dan menangani aksi petualangan, status, dan keluar |

1. **Dokumentasi dan Pembahasan Code**

****

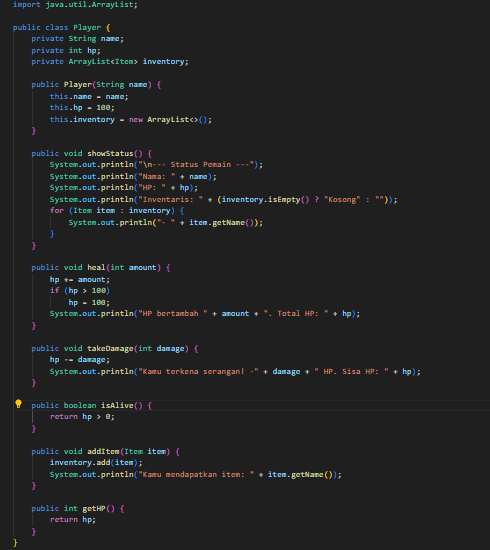
Gambar 1 Class Main

Kelas Main adalah entry point aplikasi yang mengelola menu utama untuk interaksi pengguna. Menu ini menyediakan opsi Register, Login, dan Exit. Saat user memilih Register atau Login, kelas ini memanggil method di kelas Game dan jika login berhasil, langsung mengalihkan ke permainan utama dengan method start(). Main berfungsi sebagai pengatur alur awal sebelum masuk ke logika game.

****

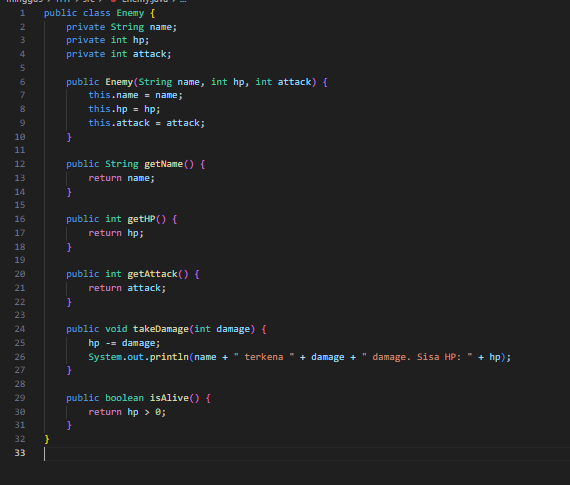
Gambar 2 Class Game

Kelas Game bertanggung jawab menyimpan data akun (username, password, dan objek Player), serta menyediakan method untuk registrasi dan login. Setelah login berhasil, kelas ini menjalankan loop permainan dengan opsi petualangan, melihat status pemain, atau keluar dari game. Kelas ini juga mengatur interaksi dasar game dan koordinasi dengan kelas lain selama permainan.

****

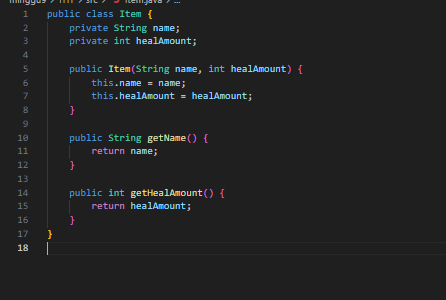
Gambar 3 Class Player

Kelas Player merepresentasikan karakter pemain dalam game, menyimpan atribut penting seperti nama, kesehatan, dan status. Kelas ini menyediakan method untuk menampilkan status pemain, memeriksa kondisi hidup, dan aksi dasar lain yang dapat dilakukan pemain selama petualangan.

****

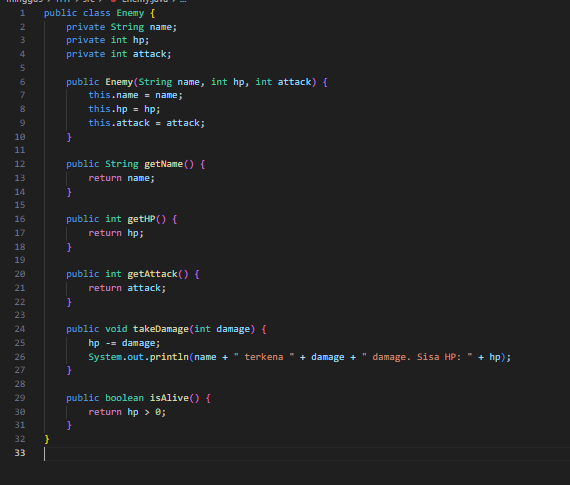
Gambar 4 Class Enemy

Kelas Enemy menggambarkan musuh yang dihadapi pemain, dengan atribut dan metode yang mengatur serangan dan pertahanan selama pertarungan. Kelas ini memungkinkan pembuatan berbagai tipe musuh dengan karakteristik yang berbeda.

****

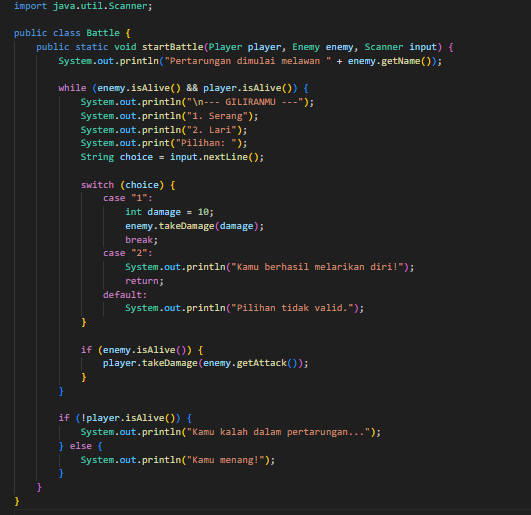
Gambar 5 Class Item

Kelas Item berfungsi sebagai representasi barang yang bisa ditemukan pemain selama permainan, seperti senjata, penyembuh, atau item khusus. Kelas ini mengatur properti item dan cara item digunakan atau diambil oleh pemain.

****

Gambar 6 Class Event

Kelas Event mengelola kejadian acak yang terjadi selama petualangan, seperti kemunculan musuh, penemuan item, atau peristiwa khusus lainnya. Kelas ini bertanggung jawab menghubungkan situasi acak dengan reaksi dan konsekuensi dalam permainan.

****

Gambar 7 Class Battle

Kelas Battle mengatur sistem pertarungan antara pemain dan musuh, termasuk giliran serang, perhitungan damage, dan pengecekan kondisi kalah atau menang. Kelas ini bertindak sebagai inti mekanik bertarung dalam game.

1. **Kesimpulan**

Proyek game ini dibangun dengan pendekatan modular di mana setiap kelas memiliki peran spesifik untuk menjaga keteraturan dan kemudahan pengembangan. Kelas Main berfungsi sebagai titik awal yang mengatur menu utama dan interaksi pengguna sebelum masuk ke game, sedangkan kelas Game mengelola data akun serta logika permainan dasar setelah login. Entitas utama dalam game seperti Player, Enemy, dan Item diwakili oleh kelas masing-masing yang menyimpan atribut dan perilaku terkait. Kejadian acak selama permainan ditangani oleh kelas Event, sedangkan kelas Battle mengatur mekanik pertarungan secara terpisah. Dengan struktur ini, kode menjadi lebih terorganisir, mudah dikembangkan, dan memungkinkan penambahan fitur secara modular tanpa mengganggu komponen lain.

1. **Daftar Pustaka**

* Modul 8 Praktikum PBO