

LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL VIII
JAVA INHERITANCE (PEWARISAN) II



Disusun Oleh:

Mohamad Dandung Sadat 105222033

PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN KOMPUTER
UNIVERSITAS PERTAMINA
2025

I. Pendahuluan

Game Chronos Echoes: The Shifting Sands dikembangkan sebagai contoh aplikasi penerapan konsep PBO dalam bentuk permainan teks interaktif. Game ini memanfaatkan konsep pewarisan antar kelas, di mana Player dan Enemy merupakan turunan dari kelas abstrak Character. Penggunaan enkapsulasi juga diterapkan untuk menjaga keamanan data pengguna dan karakter dengan metode getter dan setter. Selain itu, fitur seperti login pengguna, pendaftaran akun, penggunaan item, dan pertarungan musuh dibangun secara modular untuk mencerminkan praktik rekayasa perangkat lunak yang baik.

Dalam game ini, pemain dapat memilih satu dari tiga peran (role) yaitu "Arbiter Waktu", "Relik Pembawa", atau "Gema Pertempuran", yang masing-masing memiliki atribut awal yang berbeda. Setelah itu, pemain akan menjelajahi dunia yang terinspirasi dari berbagai era waktu dengan kemungkinan menghadapi musuh, menemukan item, atau mengalami anomali temporal. Semua fitur ini dibangun dengan pendekatan berbasis objek yang kuat dan dikombinasikan dengan interaksi pengguna melalui antarmuka berbasis teks. Penggunaan file serialization juga menambah dimensi kompleksitas dengan menyimpan data akun secara permanen dalam file, sehingga pengguna bisa melanjutkan permainan di waktu yang berbeda.

Melalui pengembangan game ini, mahasiswa tidak hanya memahami teori PBO, tetapi juga mendapatkan pengalaman praktis dalam mengimplementasikan prinsip-prinsipnya dalam proyek nyata, sesuai dengan pendekatan berbasis proyek yang sering dianjurkan dalam pembelajaran informatika (Ardiansyah, 2020; Oracle, 2023).

II. Variabel

No	Nama Variabel	Tipe data	Fungsi
1	username	String	Menyimpan nama pengguna saat login
2	password	String	Menyimpan password untuk autentikasi
3	player	Player	Menyimpan objek pemain
4	health	double	Menunjukkan jumlah darah karakter
5	temporalEnergy	double	Menyimpan energi sihir/temporal
6	attackPower	double	Menyimpan kekuatan serangan karakter
7	backpack	ArrayList	Menyimpan daftar item yang dimiliki
1	username	String	Menyimpan nama pengguna saat login
2	password	String	Menyimpan password untuk autentikasi

III. Constructor dan Method

No	Nama Metode	Jenis Metode	Fungsi
1	Player(String name, String type)	Constructor	Membuat karakter pemain sesuai tipe krononaut
2	attack(Character target)	Procedural	Menyerang musuh dengan damage acak
3	addExperience(double exp)	Procedural	Menambah XP dan menaikkan level jika mencapai batas tertentu
4	usePotion()	Procedural	Menggunakan potion untuk memulihkan health dan temporal energy
5	startGameMenu()	Procedural	Menyediakan menu utama untuk navigasi dalam game
6	login(String username, String password)	Functional	Mengembalikan objek User jika kredensial benar
7	register(String username, String password)	Functional	Menyimpan user baru ke file dan map pengguna

IV. Dokumentasi dan Pembahasan Code

4.1.Kelas Player

```
class Player extends Character implements Serializable {  
    private static final long serialVersionUID = 1L;  
    private int level;  
    private double experience;  
    private String krononautType;  
    private ArrayList<String> backpack;
```

Mendeklarasikan class Player sebagai subclass dari Character, dan dapat diserialisasi untuk disimpan ke file.ID serial versi untuk memastikan kompatibilitas saat deserialisasi.Variabel-variabel untuk level, XP, tipe karakter (role), dan tas item pemain.

4.2.Construcuter Player

```
public Player(String name, String krononautType) {  
    super(name, 100.0, 80.0, 10.0);  
    this.level = 1;  
    this.experience = 0;  
    this.krononautType = krononautType;  
    this.backpack = new ArrayList<>();  
}
```

Constructor Player, menerima nama dan tipe karakter.Memanggil constructor dari Character sebagai superclass, memberikan nilai awal.Inisialisasi level, XP, tipe karakter, dan backpack kosong.

```
switch (krononautType) {  
    case "Arbiter Waktu":  
        this.temporalEnergy = 100.0;  
        this.attackPower = 8.0;  
        break;  
}
```

Menyesuaikan statistik karakter berdasarkan tipe peran yang dipilih pemain.

```
public void attack(Character target) {  
    double damage = this.attackPower + (new Random().nextInt(5) - 2);  
    System.out.println("You hit the enemy for " + damage + " damage");  
    target.setHealth(target.getHealth() - damage);  
}
```

Method untuk menyerang karakter lain (musuh).Menentukan jumlah damage dengan variasi acak.Menampilkan hasil serangan.Mengurangi health dari target berdasarkan damage yang dihasilkan.

V. Kesimpulan

Berdasarkan hasil pengembangan dan pengujian terhadap program Chronos Echoes: The Shifting Sands, dapat disimpulkan bahwa game ini berhasil menjadi sarana efektif dalam menerapkan dan memahami konsep dasar Pemrograman Berorientasi Objek. Seluruh elemen penting PBO seperti enkapsulasi, inheritance, dan polymorphism dapat diidentifikasi dalam struktur program. Misalnya, proses pertarungan antara pemain dan musuh mencerminkan implementasi polimorfisme, sedangkan struktur akun dan karakter menunjukkan penerapan enkapsulasi dan pewarisan.

Penggunaan class-class terpisah seperti User, AccountManager, Character, Player, Enemy, dan Game menunjukkan pemisahan tanggung jawab yang baik antar bagian program (modularitas). Pendekatan ini tidak hanya meningkatkan keterbacaan kode, tetapi juga memudahkan dalam proses debugging, pengembangan lanjutan, dan pemeliharaan sistem. Sistem penyimpanan data pengguna menggunakan serialisasi objek menambahkan nilai tambah berupa persistensi data, yang penting dalam sistem game modern.

Secara keseluruhan, proyek ini memperlihatkan bagaimana teori PBO dapat diterapkan dalam dunia nyata melalui game sederhana. Tidak hanya meningkatkan pemahaman teknis mahasiswa terhadap struktur OOP di Java, tetapi juga mendorong kreativitas dalam membangun fitur dan interaksi dalam permainan. Oleh karena itu, pengembangan proyek seperti ini sangat direkomendasikan dalam kegiatan praktikum sebagai bagian dari kurikulum pembelajaran berbasis praktik.

VI. Daftar Pustaka

- Ardiansyah, Y. (2020). *Belajar OOP dalam Java*. Jakarta: Informatika Press.
- Oracle. (2023). [Java Object-Oriented Programming Concepts](#) • GeeksforGeeks. (2023). *Object Oriented Programming in Java*.