

# **LAPORAN UTS TEORI**

Mata Kuliah : Teori Pemrograman Berbasis Objek

Dosen : Irsyad Arif Mashudi, S.Kom., M.Kom



**Ilham Dharma Atmaja**

**24410702020**

**Kelas :TI 2D**

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG TAHUN 2025**

## I. Bagian Teori: Desain Konseptual

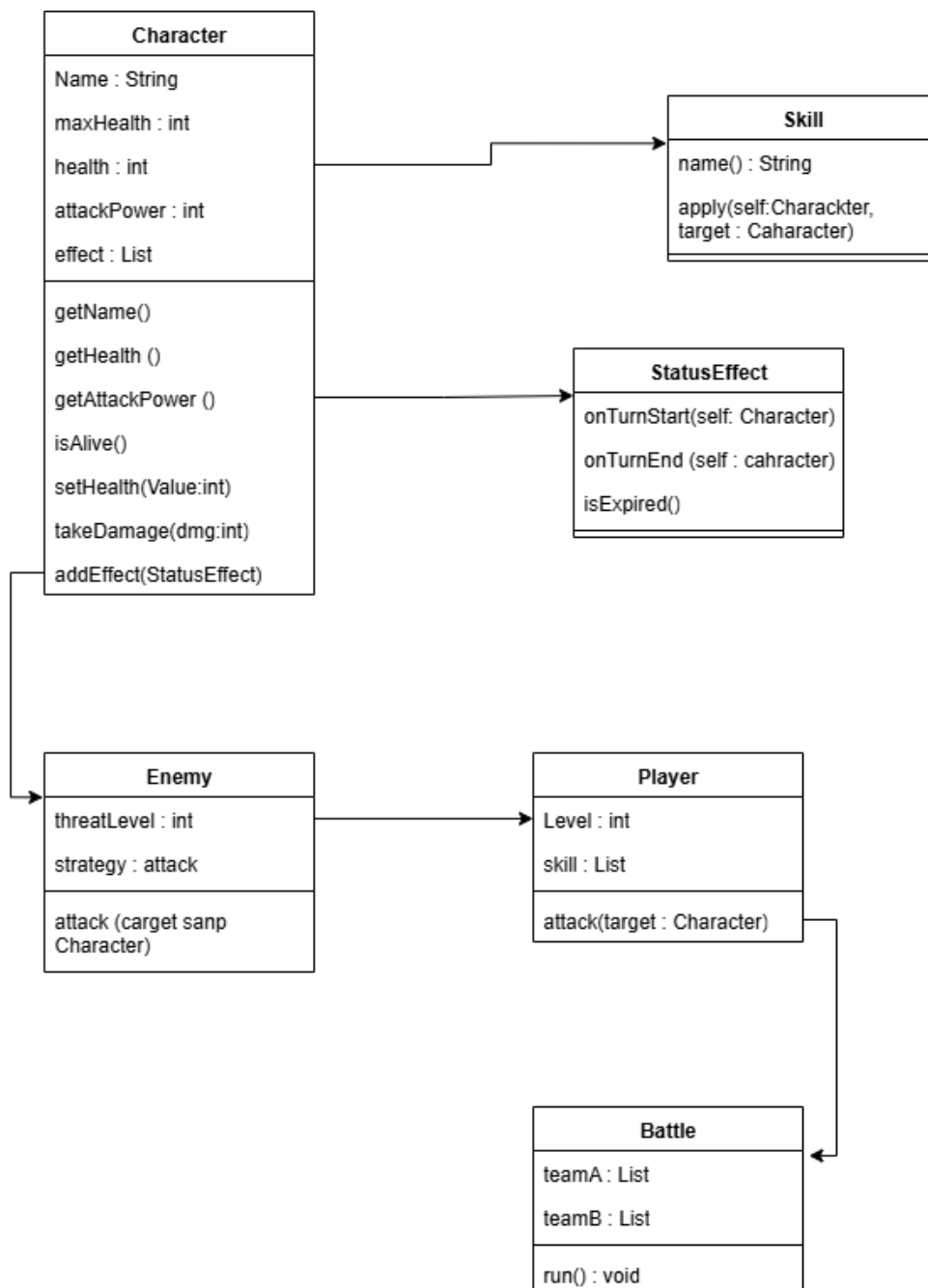
Bagian ini berfokus pada **UML Class Diagram** dan penjelasan konsep OOP yang diterapkan dalam proyek *Monster Battle*.

### A. UML Class Diagram

Sajikan diagram visual yang menggambarkan struktur dan hubungan entitas utama.

**Entitas yang Wajib Ada:**

- **Kelas Abstrak:** Character, Enemy
- **Kelas Konkret:** Player, Monster, BossMonster, Battle
- **Interface:** Skill, StatusEffect, AttackStrategy



## B. Penjelasan Konsep OOP

Konsep OOP	Penerapan dalam Proyek	Bukti dalam Kode (Contoh Class)	
<b>Abstraction</b>	Penggunaan <i>abstract class</i> (Character, Enemy) dan <i>interface</i> (Skill, StatusEffect, AttackStrategy) untuk mendefinisikan kontrak dan perilaku dasar <sup>7777</sup> .		public abstract void attack(Character target); <sup>8</sup> di Character atau <i>interface</i> Skill <sup>9</sup> .
<b>Inheritance</b>	Hierarki pewarisan di mana kelas spesifik mewarisi sifat dan metode dari kelas umum <sup>10101010</sup> .		Monster extends Enemy, Enemy extends Character <sup>11</sup> .
<b>Polymorphism</b>	Berbagai bentuk implementasi dari metode atau <i>interface</i> yang sama <sup>12121212</sup> .	Implementasi attack() yang berbeda di Monster dan BossMonster <sup>13</sup> , serta berbagai implementasi <i>interface</i> (HealSkill, ShieldEffect, FixedStrategy) <sup>1414141414141414</sup> .	
<b>Encapsulation</b>	Pembatasan akses langsung ke data (private fields) dan pengaksesan melalui <i>getter/setter</i> yang tervalidasi <sup>151515</sup> .		<i>Private fields</i> (name, health, attackPower) <sup>16</sup> dan validasi di <i>constructor</i> dan <i>setter</i> (setHealth menolak nilai negatif) <sup>17171717</sup>