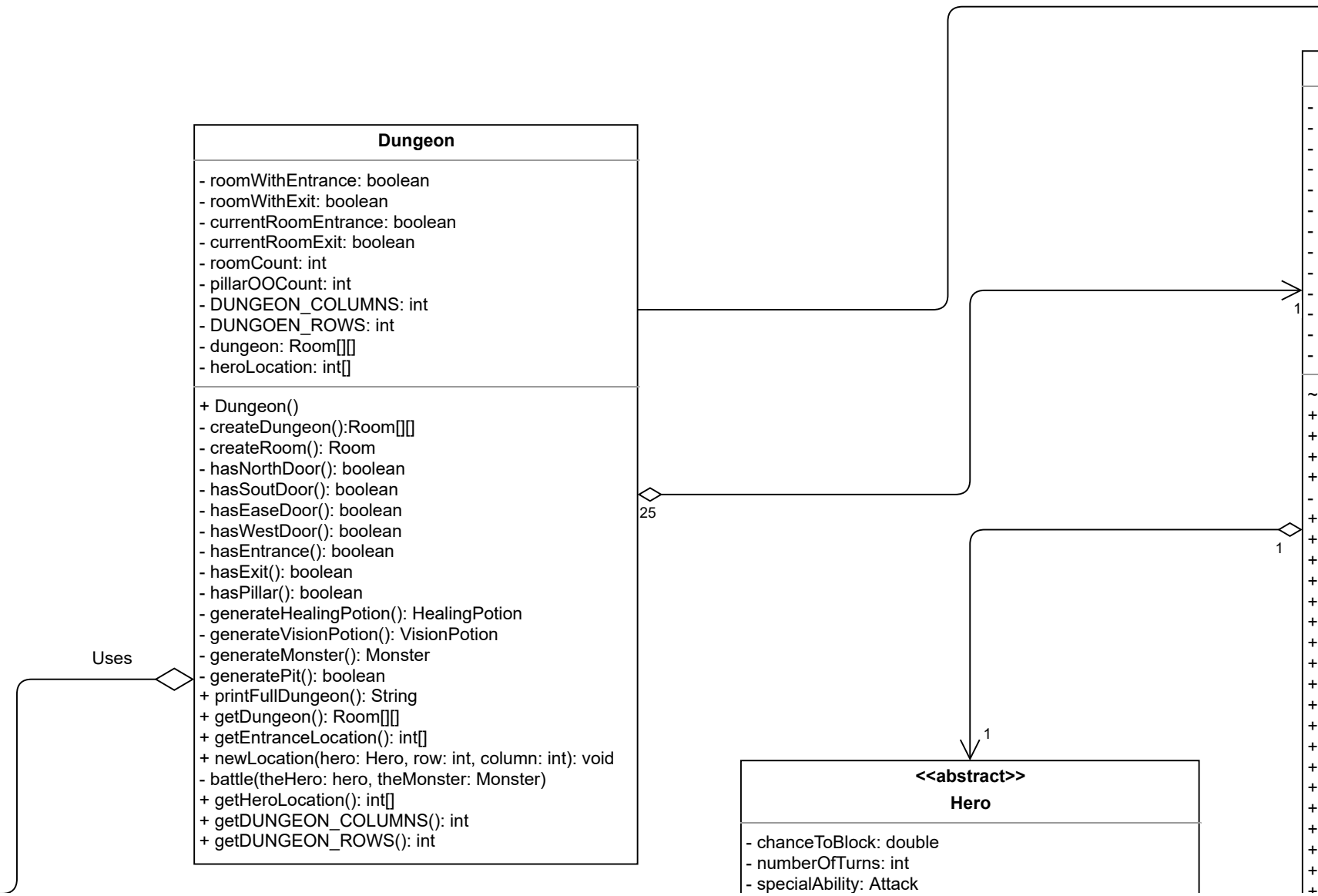
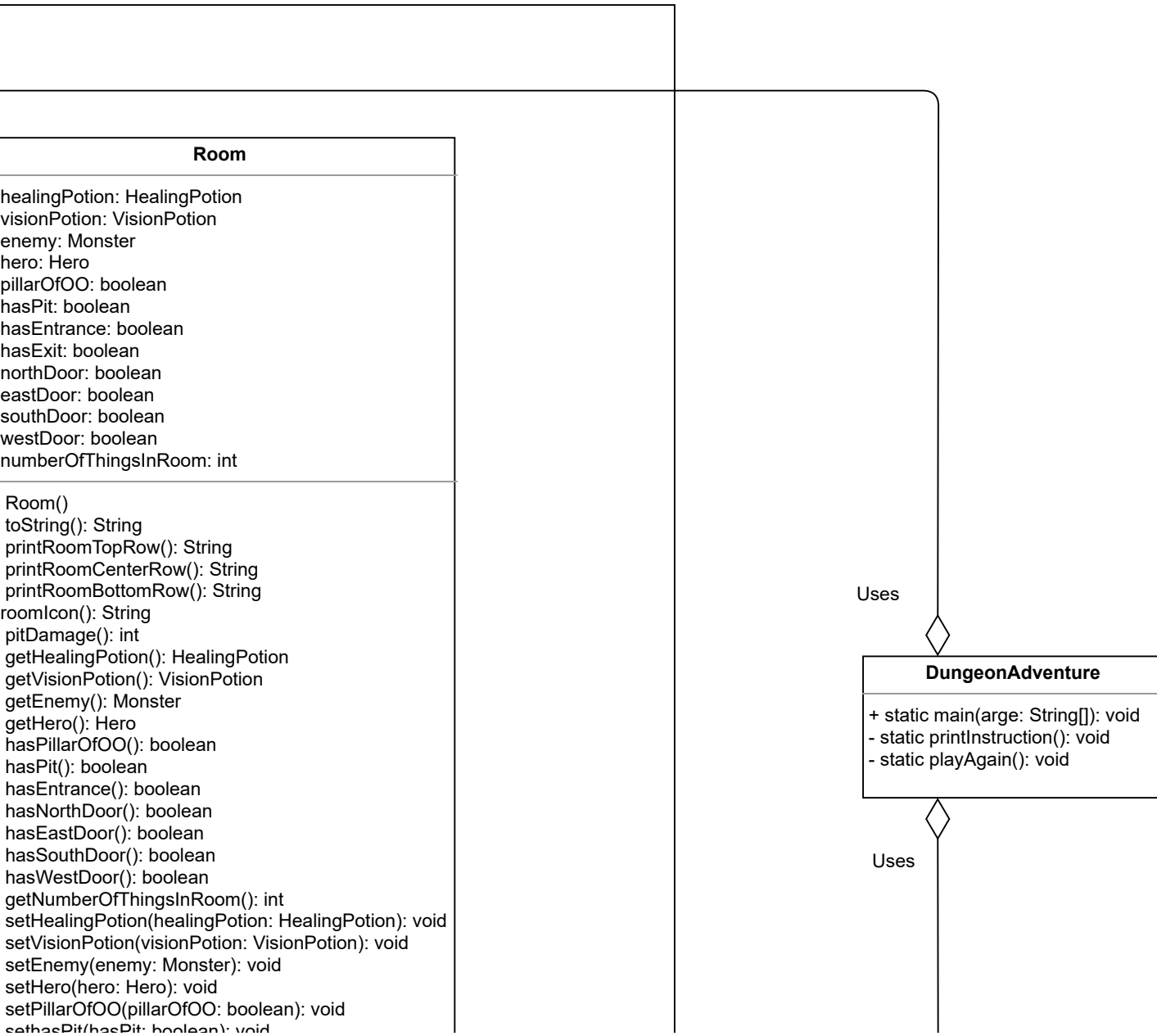
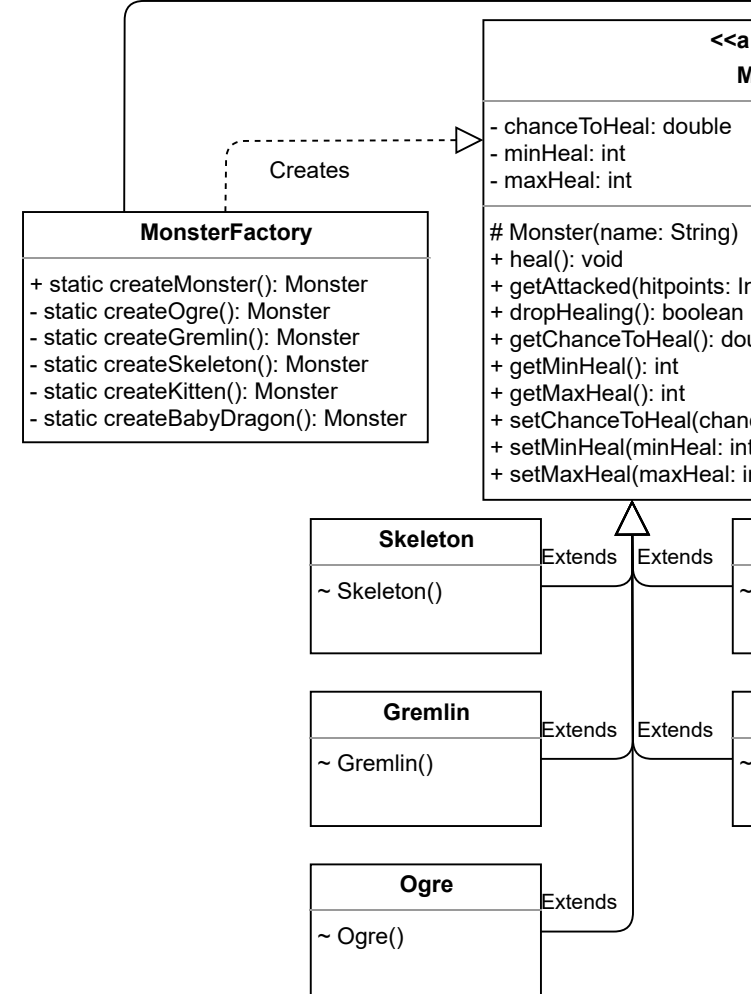
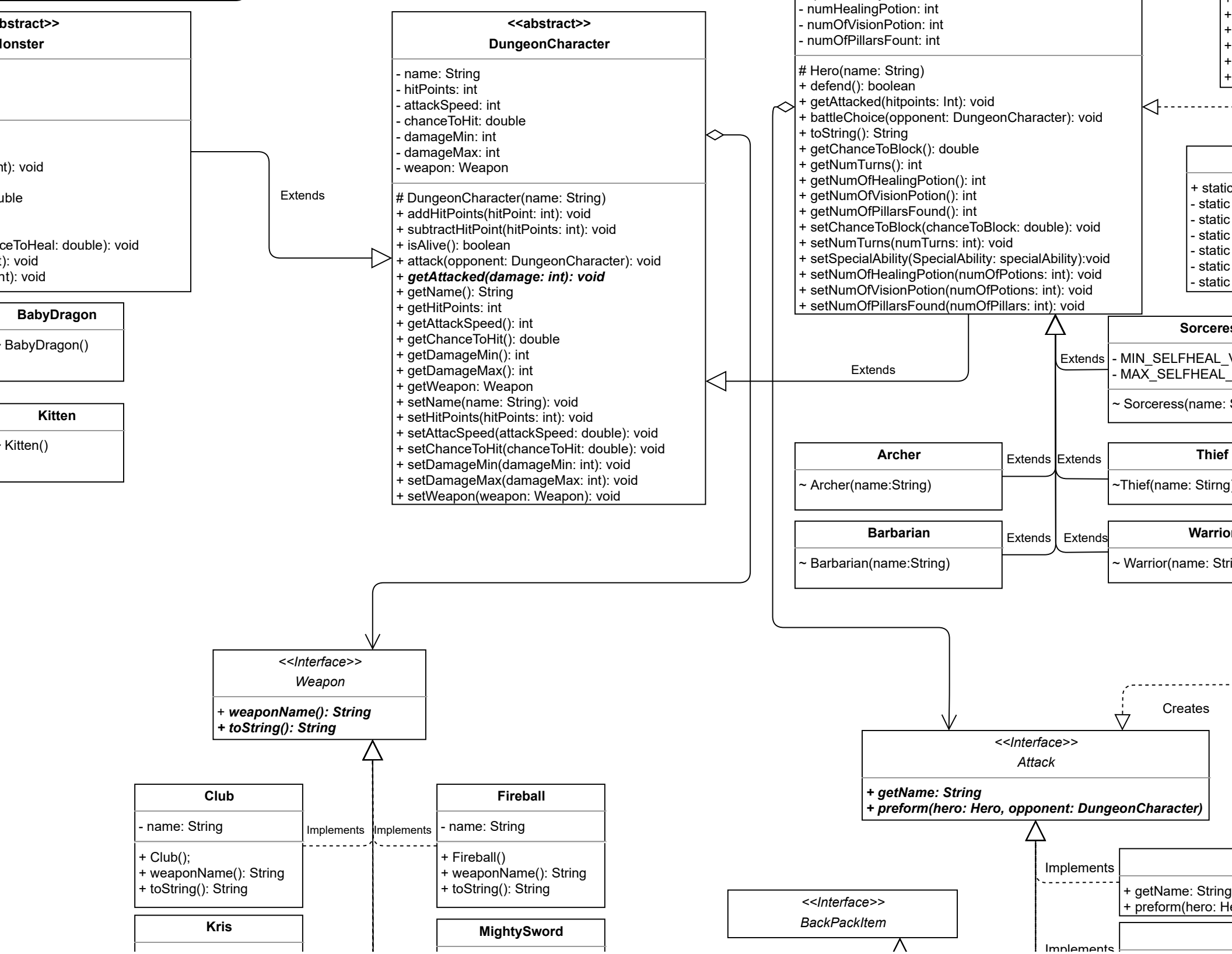


dungeon









setEntrance(hasEntrance: boolean): void
setNorthDoor(northDoor: boolean): void
setEastDoor(eastDoor: boolean): void
setSouthDoor(southDoor: boolean): void
setWestDoor(westDoor: boolean): void

Creates

HeroFactory

createNewHero(userName: String): Hero
createWarrior(userName: String): Hero
createSorceress(userName: String): Hero
createThief(userName: String): Hero
createArcher(userName: String): Hero
createBarbarian(userName: String): Hero
createUserNamer(): String

ss

VALUE: int
VALUE: int
String)

)

r

ing)

Uses

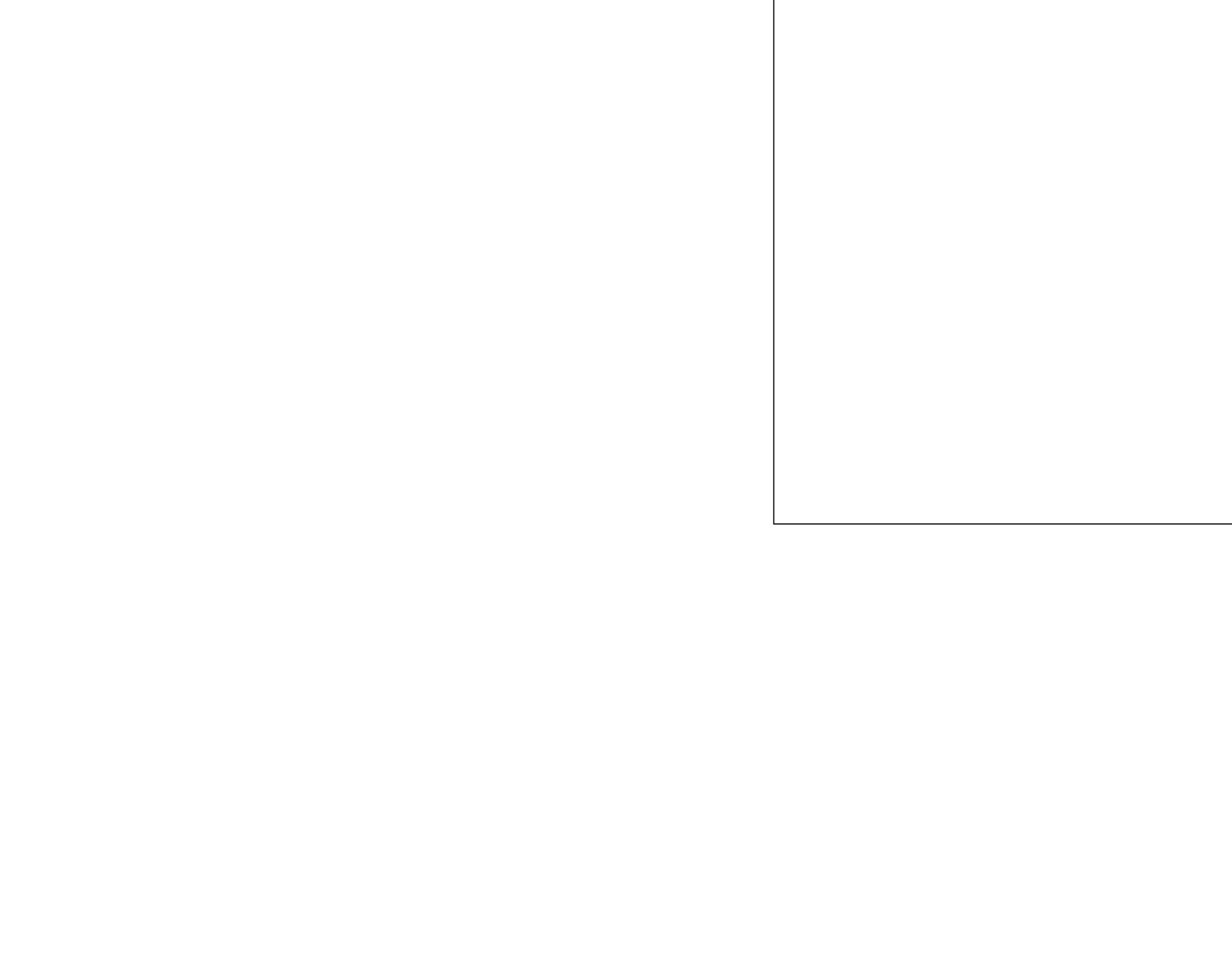
AttackFactory

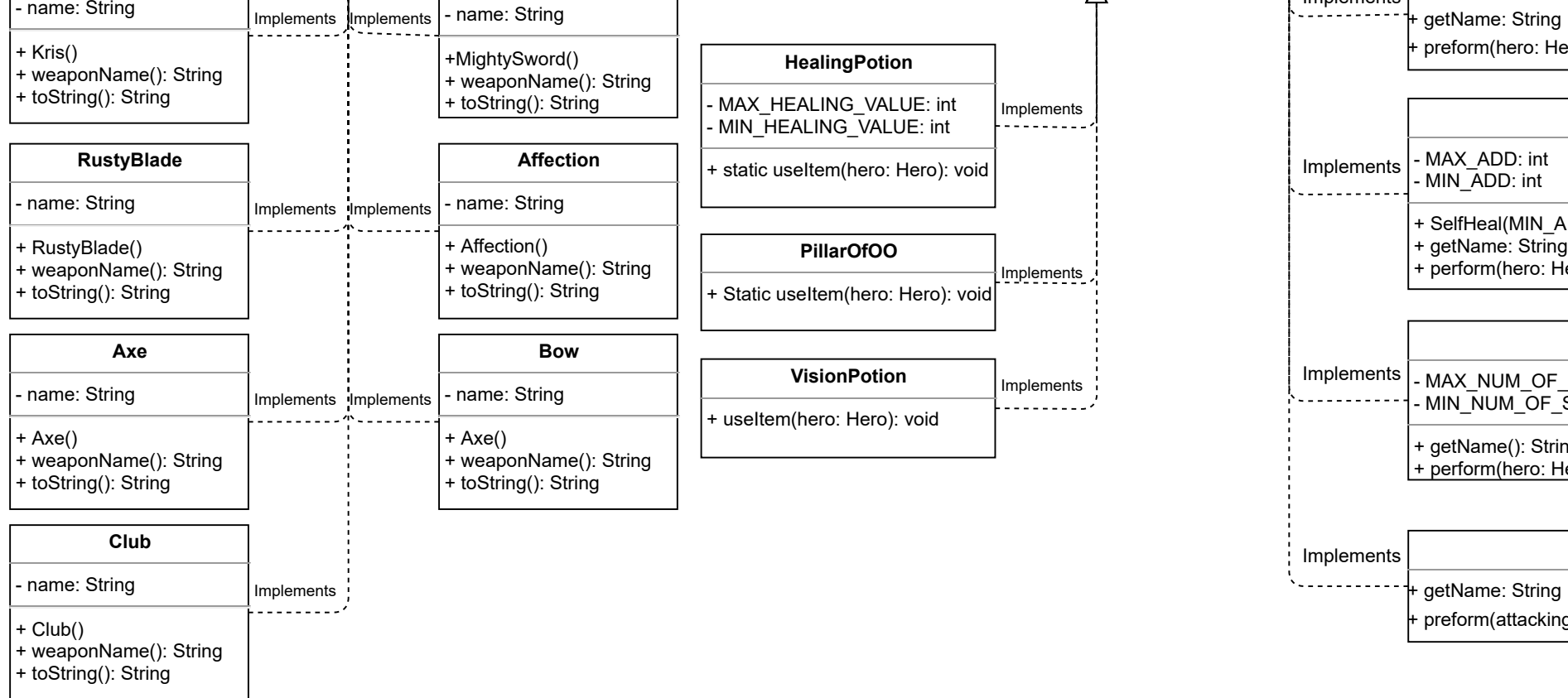
- static attackType: Map<String, Attack>
- static createNewMap(): void
+ static getBasicAttack(): Attack
+ static getCrushingBlowAttack(): Attack
+ static getMultishotAttack(): Attack
+ static getSelHealAttack(): Attack
+ static getSupriseAttack(): Attack

CrushingBlow

hero, opponent: DungeonCharacter): void

SupriseAttack





ero, opponent: DungeonCharacter): void
SelfHeal
DD: int, MAX_ADD: int)
ero, opponent: DungeonCharacter): void
MultiShot
SHOTS: int
SHOTS: int
g
ero, opponent: DungeonCharacter): Void
BasicAttack
g: DungeonCharacter, opponent: DungeonCharacter): void