

Card
Consumable
Effect
Image
Type ENUM
AttackCard
CreatureCard

AttackCard
Extends Card
Attack Power
CreatureCard
Card

CreatureCard
Extends AttackCard
Max Health
Creature
AttackCard
Card

PlayerHand
List of Cards
Draw Cards
Card
Card Pile

Creature
Name
Damage
Health
Health Status
CreatureCard
Card

Player
Health
Current Hand
Player Deck
Health Status
DiscardPile
PlayerHand
PlayerDeck
Health Status

Battle Manager
Player
Enemy Creature
Victory Status
Summoned Creatures
Current Effects
Player
Creature

Effects
Type ENUM
Animation
Card
Card Pile

Battle Manager
- Player: Player - Enemy Creature: Creature - Victory Status: Boolean
+ startBattle(): void + checkVictory(): boolean + updateBattleState(): void

Player
- currentHealth: int - isAlive: boolean - currentHand: ArrayList<Card> - playerDeck: ArrayQueue<Card> - summonedCreatures: List<Creature>
+ Player(maxHealth: int) + givePlayerCards(cardDeck: ArrayList<Card>): void + discardCard(usedCard: Card): void + drawCard(): Card + addCardToHand(newCard: Card): void + checkIsAlive(): boolean + getPlayerDeck(): ArrayQueue<Card> + useCard(card: Card): void

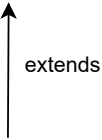
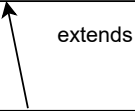
Card
-reusable: boolean -type: enum -String: name -image: String -effect: enum
+ Card(name: String, type: enum, image: String, effect: enum)

AttackCard
- damage: int
+useAttackCard(): int

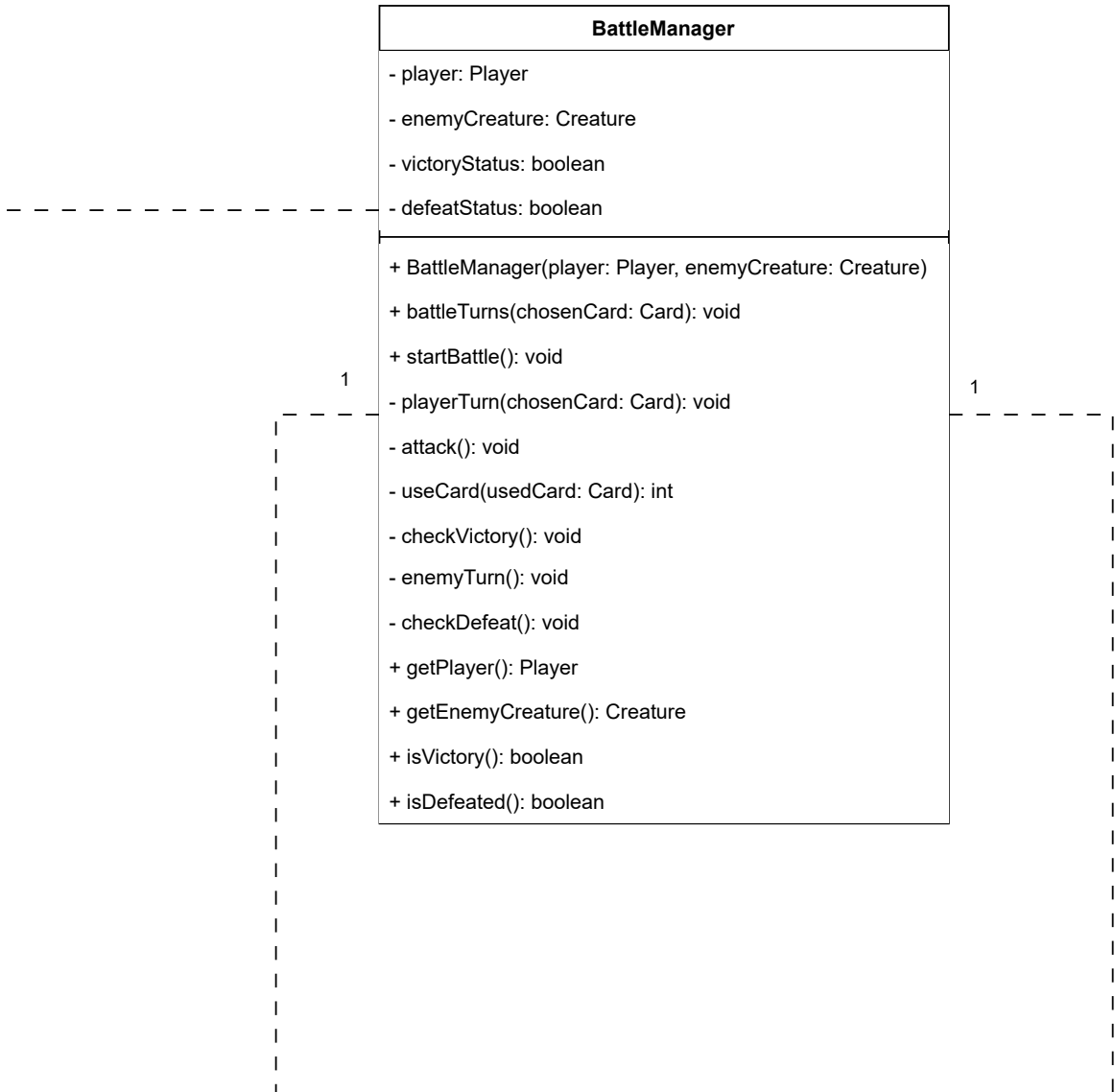
CreatureCard
- health: int
+useCreatureCard(): Creature

Creature
- name: String
- damage: int
- currentHealth: int
- isAlive: boolean
+Creature(name: String, damage: int, maxHealth: int)

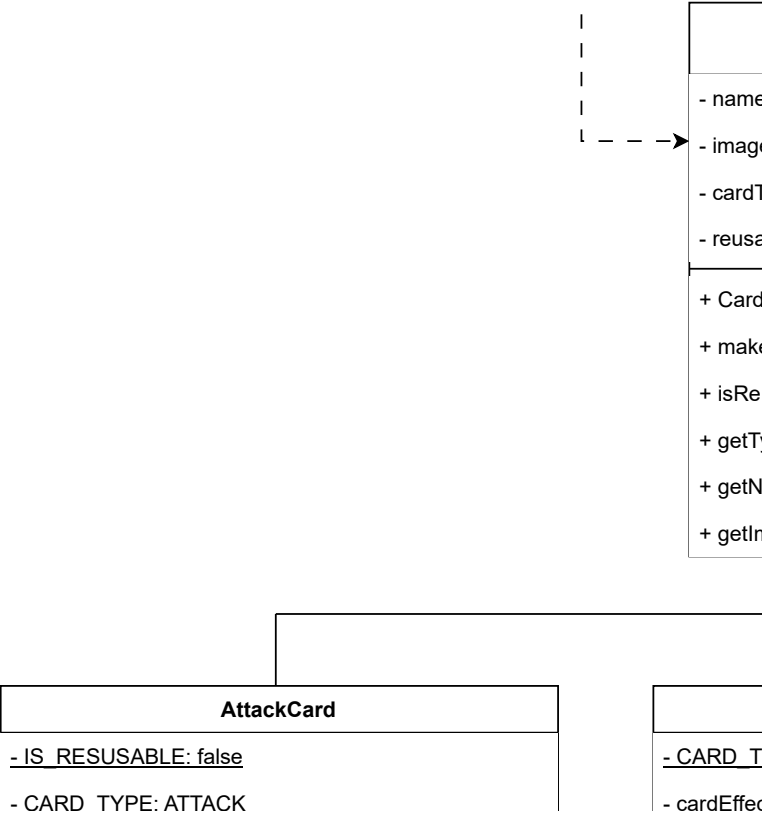
StatusCard
+useStatusCard(): void

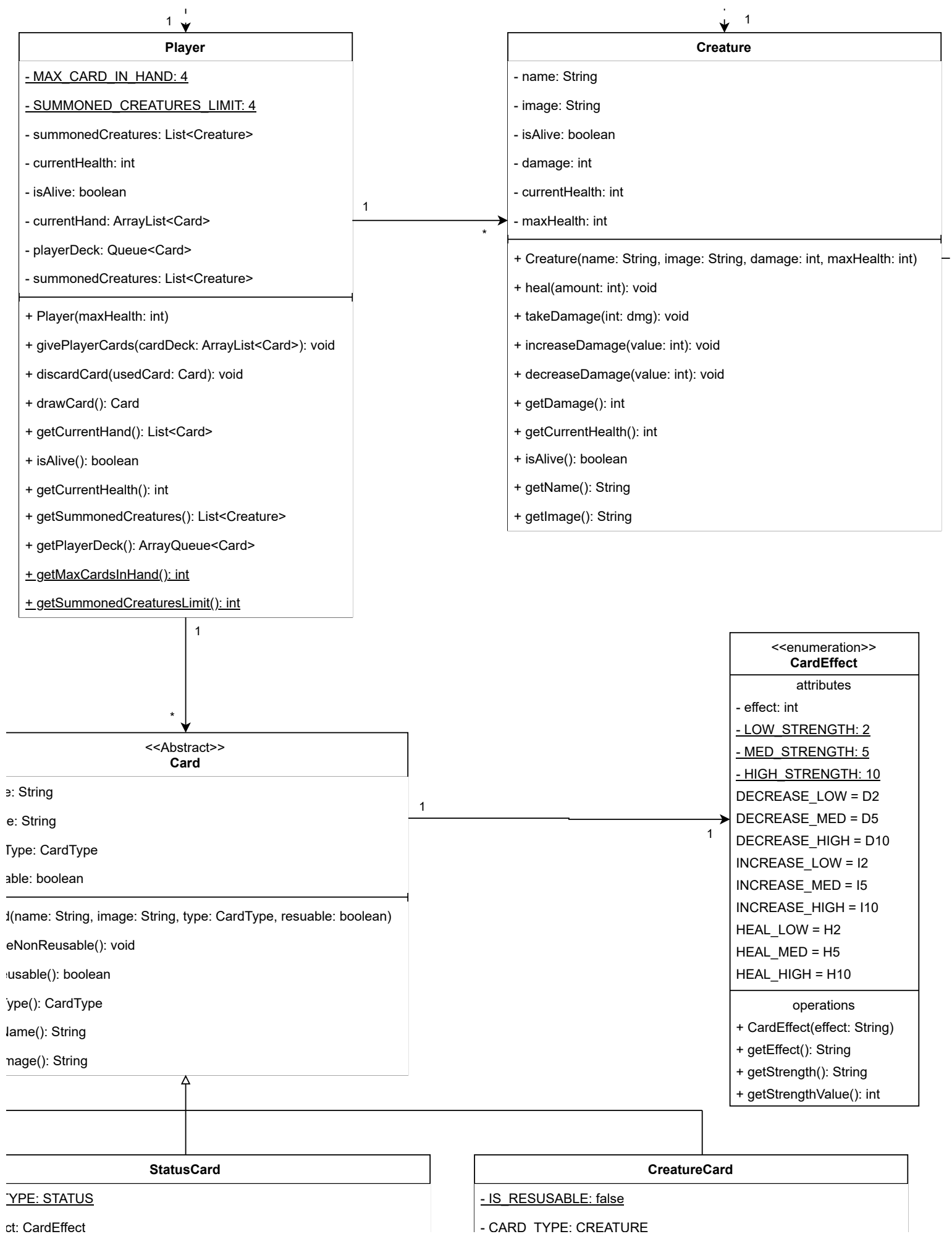


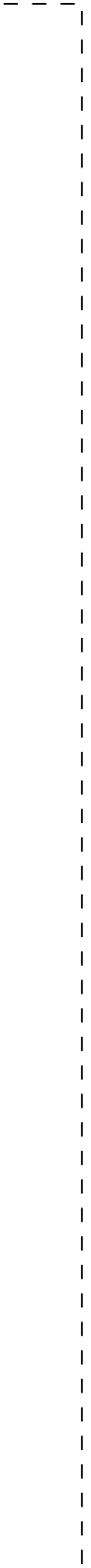














Player

- damage: int

---

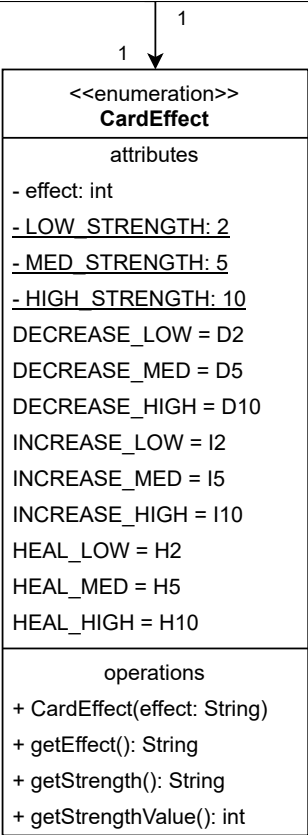
+ AttackCard(name: String, image: String, damage: int)  
+ givePlayerCards(cardDeck: ArrayList<Card>): void  
+ useAttackCard(): int  
+ getDamage(): int

Game

+ StatusC:  
+ useStat:  
+ applyStr:  
+ applyMo:  
+ applyEff:  
+ getCard!

CardEffect

```
Card(name: String, image: String, effect: CardEffect, reusable: boolean)
useCard(creature: Creature): void
longEffect(creature: Creature): void
moderateEffect(creature: Creature): void
effect(creature: Creature, upgrade: String, strength: int): void
Effect(): CardEffect
```



CreatureCard

```
- damage: int
- health: int

+ CreatureCard(name: String, image: String, damage: int, health: int)
+ useCreatureCard(): Creature
+ getDamage(): int
```



