Class Overview

Class over view
Class Game:
- The main class that manages the flow of the game (rounds, stages, executing commands)
- Attributes:
- Player p1, p2
- int round, maxRounds
- bool printBoard
- Methods:
- void start()
- void nextRound()
- void executeCommand()
Class Players

Class Player:

- It represents one player.
- Attributes:
- int id, health, mana, manaPoolSize
- Deck deck
- Hand hand
- Graveyard graveyard
- Zone fieldZone, battleZone
- bool canRedraw
- Methods:
- void drawCard()
- void playCard()

Class Deck, Hand, Graveyard:

- Specialized card collections (can be same class with different context).

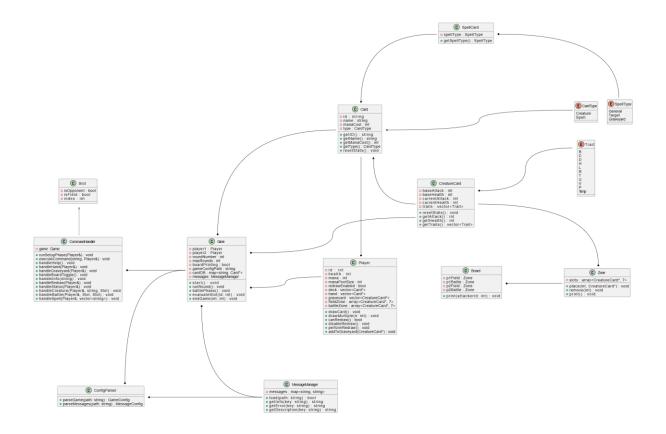
- Attributes:
- std::vector <card*> cards</card*>
- Methods:
- Card* drawTop()
- void add(Card*)
- void remove(Card*)
- void print()
- Methods:
- Card* drawTop()
- void add(Card*)
- void remove(Card*)
- void print()
Class Card:
- Base class for all cards.
- Attributes:
- std::string id
- std::string name
- int manaCost
- Methods:
- virtual void print() const = 0,
Class CreatureCard : public Card:
- Creature card with attack, health and traits.
- Attributes:
- int baseAtk, baseHp
- int curAtk, curHp
- std::vector <trait> traits</trait>
- Methods:
- bool hasTrait(Trait)
- void applyTraitEffect()

Class SpellCard: public Card - A spell card with an instant effect. - Attributes: - SpellType type (General, Target, Graveyard) - Methods: - void cast(Player& caster, ...) **Class Zone:** - Represents the Field or Battle Zone (7 slots). - Attributes: - std::array<CreatureCard*, 7> slots - Methods: - void place(int slot, CreatureCard*) void remove(int slot) - void print() **Class Board:** - It represents the entire board - all zones for both players. - Attributes: - Zone p1Field, p1Battle - Zone p2Field, p2Battle - Methods: void print(int attackerId) Class MessageConfig: - Loads messages from a file and maps them to the appropriate key. - Attributes:

- std::map<std::string> messages
- Methods:
- void loadFromFile()
- std::string get(const std::string&)

Class ConfigParser:

- Parses the configuration files required for the game.
- Methods:
- GameConfig parseGame(std::string path)
- MessageConfig parseMessages(std::string path)



Gruppe 051 Florian Kerman Miloš Đukarić Stefan Jović