

Documento UML

Riva – Pieruz – Sartore – Tonoli

All'interno di questo documento viene spiegato il diagramma UML del model, contenente le relative classi con attributi e metodi. I setter e getter non verranno descritti.

Game:

Attributi:	MAX_PLAYERS	indica il massimo numero di giocatori per ogni partita (4)
	MIN_PLAYERS	indica il minimo numero di giocatori per ogni partita (2)
	gameBoard	riferimento al campo comune della partita
	gameState	indica lo stato della partita
	players	lista contenente il nome di ogni giocatore
	firstPlayer	riferimento al primo giocatore della partita
	playerInTurn	riferimento al giocatore in turno

Metodi:	startGame()	inizializza i parametri per l'inizio della partita
	endGame()	si occupa della fase finale della partita
	addPlayer()	aggiunge un giocatore alla partita

GameState (enumerazione):

WAIT_PLAYERS	stato di attesa giocatori
START	stato di inizio della partita
IN_GAME	stato di partita in corso
END	stato di fine partita

GameBoard:

Attributi:	game	riferimento alla partita del giocatore
	achievementDeck	mazzo delle carte obiettivo
	goldDeck	mazzo delle carte oro
	resourceDeck	mazzo delle carte risorsa
	starterDeck	mazzo delle carte iniziali

Metodi:	drawCard()	pesca una carta da uno specifico mazzo
---------	------------	--

Player:

Attributi:	username	stringa contenente il nickname del giocatore
	game	riferimento alla partita del giocatore
	playerBoard	riferimento al campo del giocatore in cui può posizionare le sue carte
	firstToEnd	attributo booleano settato a true quando il primo giocatore della partita raggiunge 20 punti
	cardInHand	lista delle carte che il giocatore può posizionare in campo
	personalObj	contiene le due carte obiettivo che vengono fatte scegliere al giocatore
	chosenObj	carta obiettivo scelta dal giocatore
	points	punti totali del giocatore
	playerState	stato corrente del giocatore
	chat	lista dei messaggi del giocatore

Metodi:	sendMessage()	metodo invocato dal giocatore per inviare un messaggio
	displayMessage()	aggiunge un messaggio alla chat del giocatore
	addInHand()	aggiunge una carta alla mano del giocatore

PlayerState (enumerazione):

PLAY_CARD	il giocatore deve posizionare una carta in campo
DRAW_CARD	il giocatore deve pescare una carta
NOT_IN_TURN	il giocatore è in attesa del suo turno

PlayerBoard:

Attributi:	player	riferimento al giocatore
	starterCard	carta iniziale data al giocatore
	cardPosition	hashmap la cui chiave è la posizione nel campo e il valore è la carta in quella posizione
	symbolCount	hashmap la cui chiave è un simbolo e il valore è il conteggio totale di quel simbolo nel campo del giocatore

Metodi:	addSymbolCount()	pesca una carta da uno specifico mazzo
---------	------------------	--

Card:

Attributi:	corners	array che contiene gli angoli della carta
	type	tipo della carta
	color	colore della carta
	card_number	numero identificativo della carta
	current_side	indica il lato della carta visibile dal giocatore
	back	indica se il giocatore sta vedendo il retro della carta
	retro	retro della carta collegato al fronte

CardBack:

Attributi:	symbols	lista dei simboli al centro del retro della carta
------------	---------	---

ResourceCard:

Attributi:	point	indica l'eventuale punteggio dato dal posizionamento della carta
------------	-------	--

GoldCard:

Attributi:	basePoint	indica il punteggio base dato dal posizionamento della carta
	condition	moltiplicatore del punteggio base
	cost	array che indica per ogni posizione il numero di simboli che devono essere presenti nel campo

StarterCard: rappresenta le carte iniziali

Achievement (interfaccia):

Metodi:	calcPoints()	calcola i punti della carta obiettivo
---------	--------------	---------------------------------------

AchievementDiagonal: carte obiettivo legate al posizionamento di tre carte dello stesso colore in diagonale

Attributi:	basePoint	indica il punteggio base dato della carta obiettivo
	color	indica il colore della diagonale

AchievementItem: carte obiettivo legate al conteggio del numero di oggetti

Attributi:	basePoint	indica il punteggio base dato della carta obiettivo
	symbol	lista di simboli contenente gli oggetti da avere nel campo

AchievementL: carte obiettivo legate al posizionamento di tre carte a “L”

Attributi:	basePoint	indica il punteggio base dato della carta obiettivo
	color	indica il colore della “L”

AchievementResources: carte obiettivo legate al conteggio del numero di risorse

Attributi:	basePoint	indica il punteggio base dato della carta obiettivo
	symbol	lista di simboli contenente le risorse da avere nel campo

Symbols (enumerazione):

FUNGI
PLANT
ANIMAL
INSECT
QUILL
INKWELL
MANUSCRIPT
EMPTY
NOCORNER

Color (enumerazione):

RED
GREEN
BLUE
PURPLE
WHITE

Attributi:	associatedColor	indica per ogni colore il colore associato all’obiettivo di posizionamento a “L”
------------	-----------------	--

Corner:

Attributi:	symbol	indica il simbolo contenuto nell'angolo
	state	indica lo stato dell'angolo

CornerEnum (enumerazione):

TL	angolo in alto a sinistra
TR	angolo in alto a destra
BL	angolo in basso a sinistra
BR	angolo in basso a destra

CornerState (enumerazione):

VISIBLE	angolo della carta sul quale si può posizionare una carta sopra
NOT_VISIBLE	angolo della carta sul quale non si può posizionare una carta sopra
OCCUPIED	angolo della carta inizialmente visibile sul quale è stata posizionata una carta

Chat:

Attributi:	CHATDIM	costante che indica il numero di messaggi visibili nella chat del giocatore
	game	riferimento alla partita
	player	riferimento al giocatore

Metodi:	forwardMessage()	metodo che richiama il metodo displayMessage() per ogni giocatore che deve vedere il messaggio
---------	------------------	--