Ripoff The Card Game

Ripoff Card Game Azkaban Dabmentors Tyler Wallschleger, Brandon Kyle,
Dakota Simpkins | December 9th 2019

gui

RipoffGUI

- primaryStage:Stage
- \# pvc:PlayerViewController
- \# sgc:SimpleGameController

- \+ RipoffGUI(Stage _primaryStage)
- \+ loadMainPanel(): void
- \+ loadGamePanel(): void
- \+ loadPlayerPanel(): void
- \+ loadRegisterPanel(): void
- \+ loadMarketPanel(): void
- \+ loadPlayPanel(): void
- \+ loadAiPanel(): void
- \+ loadPlayerWonPanel(): void
- \- buildMainPanel(): Scene
- \- buildGamePanel(): Scene
- \- buildPlayerPanel(): Scene
- \- buildRegisterPanel(): Scene
- \- buildMarketPanel(): Scene
- \- buildPlayPanel(): Scene
- \- buildAiPanel(): Scene
- \- buildPlayerWonPanel(): Scene
- \- addButton(_title String, _throwMessage RipoffMessage): Button
- \- addVBox(_header String): VBox

model

AITester

+ main(args[] String): void

DeckCardTester

+ main(args[] String): void

Card

- cardName: String
- cardId: String
- cardPower: int

- \+ Card(_name String, _id String, _power int)
- \+ toString(): String
- \+ comparePower(_compareCard Card): int

Class

+ attribute1:type = defaultValue
+ attribute2:type
- attribute3:type

+ operation1(params):returnType
- operation2(params)
- operation3()

Contender

\# myDeck: Deck = new Deck()
\# wonRounds: int = 0
\# lostRounds int = 0

+ Contender()

Game

+ howToPlay(): void
+ playAI(): void
+ playHoomans(): void
+ messageReceived(_event RipoffEvent): void

Deck

+ deckCards: ArrayList<Card> = new _selectedCards ArrayList<Card>()

+ Deck(_cards ArrayList<Card>)
+ Deck()
+ toString(): String
+ sortByID(): void
+ shuffleCards(): void
+ calculateTotalPower(): int
+ calculateAveragePower(): int
+ combineDecks(_deckToAdd Deck): void
+ drawCards(_numberDrawn int): ArrayList<Card>
+ randomPick(): Card
+ addCard(_cardToAdd Card): void
+ filldeck(): void
- findMinIDIndex: int

ComputerOpponent

- advantagePoints: int

+ ComputerOpponent(_usedDeck Deck)
+ ComputerOpponent()
+ queryForCard(_pickedCards int): int
+ compareInitialPower(_enemyUser Contender): void
+ decideCard(_selectedCards ArrayList<Card>): Card
- decideAIState(): String
- indexAggressiveCard(_selectedCards ArrayList<Card>): int
- indexProtectiveCard(_selectedCards ArrayList<Card>): int

Market

+ Market
+ messageReceived(_event RipoffEvent)

Play

+ Play
+ messageReceived(_event RipoffEvent)

Player

+ Player(): void
+ deckShow(): void
+ sendMessage(): void
+ registerNewUser(_userInfo String[]): RipoffMessage
+ messageReceived(_event RipoffEvent): void

GameBoardAI

- playerOneWins: int = 0
- playerTwoWins: int = 0
- playerOne: Contender = new Contender()
- ComputerOpponent playerTwo = new ComputerOpponent()
- playerOneDrawn: ArrayList<Card>
- playerTwoDrawn: ArrayList<Card>

+ GameBoardAI(_playerOneDeck Deck, _playerTwoDeck Deck)
+ GameBoardAI()
+ userPickCard(_playerInput int): Card
+ aiPickCard(): Card
+ determineRoundWinner(_playerCard Card, _aiCard Card): RipoffMessage
+ determineGameWinner(_roundResult RipoffMessage): RipoffMessage
+ getPlayerOneChoices(): ArrayList<Card>

Register

+ Register
+ createProfile(): void
+ messageReceived(_event RipoffEvent)

RipoffBase

- LISTENERS: int = new ArrayList<>

+ addListener(listener ListenerInterface):void
+ removeListener(listener ListenerInterface):void
\# fireEvent(_event RipoffEvent): void

games

DBTranslator

+ createUser(_username String, _firstName String, _password String): RipoffMessage

Main extends Application

+ start(primaryStage Stage): void
+ main(args String[]): void

SimpleGameController implements ListenerInterface

\# gui: RipoffGUI
\# activeModule: RipoffBase = null
\# gbAI: GameBoardAI

SimpleGameController(_primaryStage Stage)
- mainPanel: void
- gamePanel: void
- playerPanel: void
- marketPanel: void
- registerPanel: void
- aiWinPanel: void
- ripoffPanelListener(_newModule RipoffBase): void
+ messageRecieved(_event RipoffEvent): void

test

CardTesterClass

- testCardOne Card = new Card("Test Card One", "01", 1)
- testCardTwo Card = new Card("Test Card Two", "02", 2)
- testCardThree Card = new Card("Test Card Zero", "0", 0)
- testCardTen Card = new Card("Test Card Ten", "10", 10)
- testCardTwenty Card = new Card("Test Card Twenty", "20", 20)

+ runCardTests(): void
- testCardComparePower(): void

DeckTesterClass

- genericCards ArrayList<Card> = new ArrayList()
- testCardOne Card = new Card("Card One", "01", 1)
- testCardTwo Card = new Card("Card Two", "02", 2)
- testCardThree Card = new Card("Card Three", "03", 3)
- testCardFour Card = new Card("Card Four", "04", 4)
- testCardFive Card = new Card("Card Five", "05", 5)
- testCardSix Card = new Card("Card Six", "06", 6)
- testCardSeven Card = new Card("Card Seven", "07", 7)
- testCardEight Card = new Card("Card Eight", "08", 8)
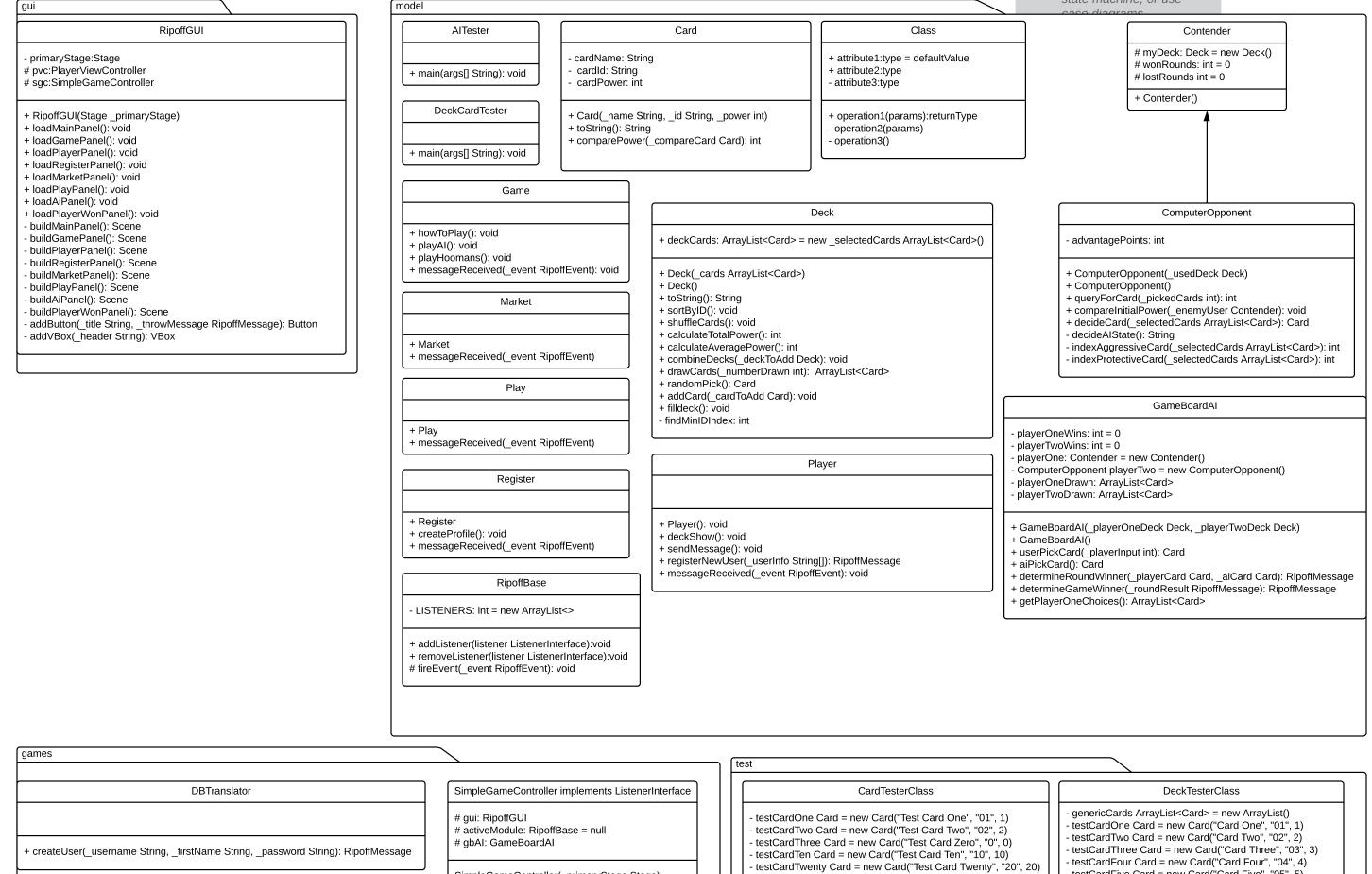- testCardNine Card = new Card("Card Nine, "09", 9)
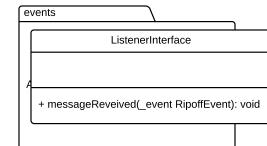- testCardTen Card = new Card("Card Ten", "10", 10)
- genericDeck Deck = new Deck()

+ runDeckTests()
- testDeckShuffleCards(): void

RunTests

+ main(args[] String): void

events

ListenerInterface

+ messageReveived(_event RipoffEvent): void

RipoffMessage

+ GAME_PANEL: int = 1
+ PLAYER_PANEL: int = 2
+ EXIT_PANEL: int = 3
+ REGISTER_PANEL: int =4
+ MARKET_PANEL: int = 5
+ PLAY_PANEL: int = 6
+ SEE_DECK int = 10
+ SEND_MEMES: int = 20
+ CREATE_PROFILE: int = 30
+ CHECK_LOGIN: int = 31
+ MISSING_USERNAME: int = 32
+ MISSING_FIRSTNAME: int = 33
+ MISSING_PASSWORD: int = 34
+ HOW_TO_PLAY: int = 100
+ PLAY_AGAINST_HOOMANS:int = 200
+ PLAY_AGAINST_AI: int = 300
+ PLAYER_WON_ROUND: int= 400
+ AI_WON_ROUND: int = 401
+ DRAW_ROUND: int = 402
+ PLAYER_WON_GAME: int = 403
+ AI_WON_GAME: int = 404
+ NO_WINNER: int = 405
+ ERROR: int = 999
- int _thismessage

+ RipoffMessage(_message int)
+ getCode(): int

RipoffEvent

- message RipoffMessage

+ RipoffMessage(source Object, _message RipoffMessage)
+ getMessage(): RipoffMessage