Visual - rules:string + Init(player one, player two) << constructor>> + help() + click save button() + click load button() + surrender() + buttons card(player one, player two, card) + show game() + show winner() + show buttons cards() + show save button() + show labels buttons() + hide board() + first turn() + second turn() + resize cards() + config image played card(card) + config image(deck player one, deck player two, especial card) + get_player() + set player(player) + set rules(rules)

+ set game(game)

Game<abstract class> - player1:Player - player2:Player - rules:string Deck - deck:Deck - suits:vector<string> + init(deck, player one, player two) <base operation> - values:vector<int> + play() <template method> - deck:vector<string> + verify turn() + next round() + init() <<constructor>> + sum score() + create deck() + reset values() + deal cards() + verify number() + get deck() + verify suit(card) + draw() + save game() + load game() + get_especial_card() + get suit(card) + set deck(deck) + set especial card(especial card) Player - hand:vector<Card> - value:int - name:string - playing card:string Durak<controller> - rules:string + append card(card) - deck:Deck + remove card(card) + clean hand() + get_playing_card() + init() override + get_player_name() + play() override + get_size_hand() + verify turn() override + get value() + next round() override + get hand() + sum score() override + set hand() + reset values() override + set_value(value) + verify number() override + set player name(name) + verify suit(card) override + set value hand(card) + draw() override + save game() override + load game() override + get special card() override + get suit(card) override

+ set deck(deck) override

+ set especial card(especial card) override