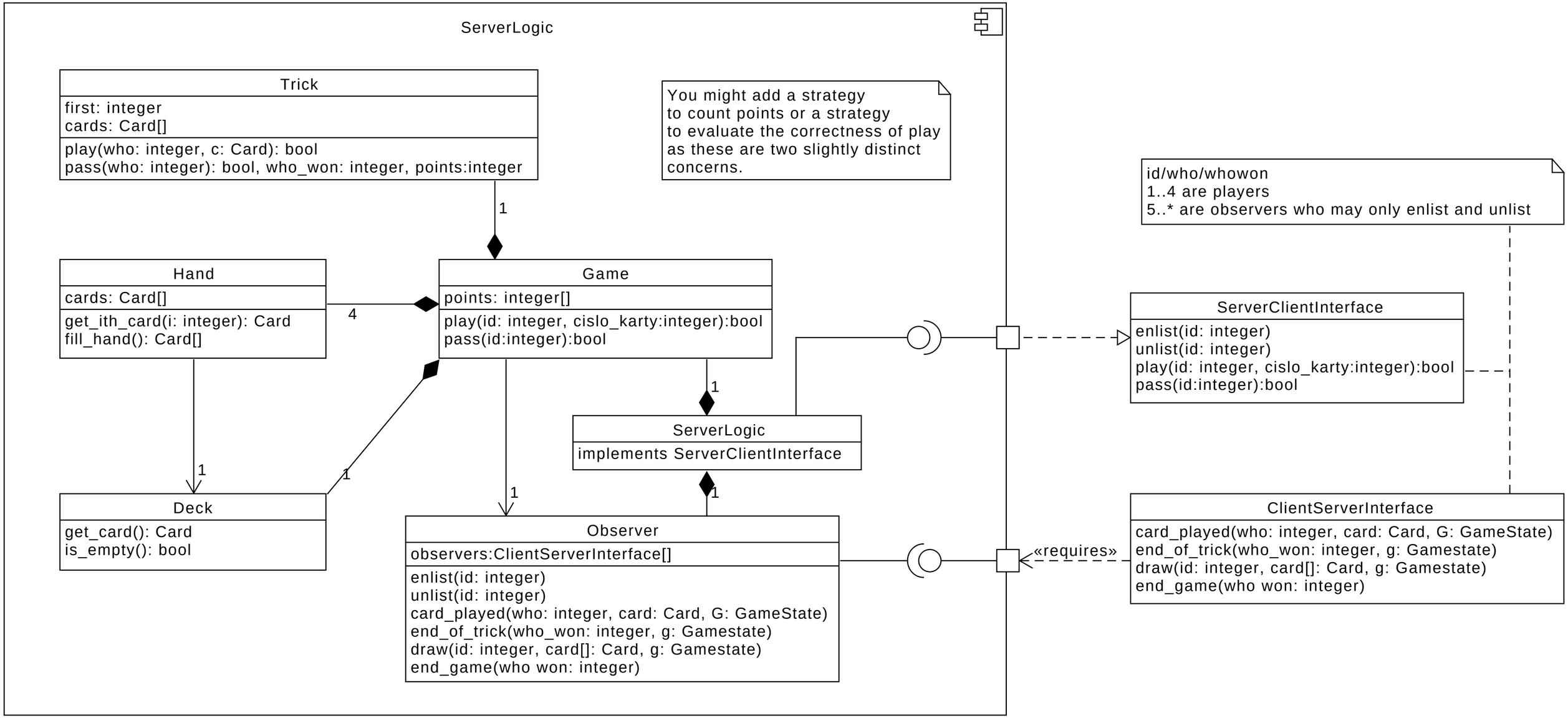
ServerLogic
Trick
first: integer
cards: Card[]
play(who: integer, c: Card): bool
pass(who: integer): bool, who_won: integer, points:integer
You might add a strategy
to count points or a strategy
to evaluate the correctness of play
as these are two slightly distinct
concerns.
id/who/whowon
1..4 are players
5..* are observers who may only enlist and unlist
1
Hand
cards: Card[]
get_ith_card(i: integer): Card
fill_hand(): Card[]
4
Game
points: integer[]
play(id: integer, cislo_karty:integer):bool
pass(id:integer):bool
ServerClientInterface
enlist(id: integer)
unlist(id: integer)
play(id: integer, cislo_karty:integer):bool
pass(id:integer):bool
1
ServerLogic
implements ServerClientInterface
1
1
1
1
Deck
get_card(): Card
is_empty(): bool
Observer
observers:ClientServerInterface[]
enlist(id: integer)
unlist(id: integer)
card_played(who: integer, card: Card, G: GameState)
end_of_trick(who_won: integer, g: Gamestate)
draw(id: integer, card[]: Card, g: Gamestate)
end_game(who won: integer)
«requires»
ClientServerInterface
card_played(who: integer, card: Card, G: GameState)
end_of_trick(who_won: integer, g: Gamestate)
draw(id: integer, card[]: Card, g: Gamestate)
end_game(who won: integer)