Playing a card (Socket)

View	SocketClient	SocketConnectionHandler	Controller	PlayerField	ScoreBoard

View ← showMessage("It's your turn!") ← SocketClient ← new YourTurn() ← SocketConnectionHandler ← yourTurn() ← Controller

View ← setMyTurn(true)

View ← setPlayPhase(true)

View → playCardResponse(card, where, isFront) → SocketClient → getController() → SocketConnectionHandler

Controller → playCard(connectionHandler, card, where, isFront)

opt [the client tries to play not in its turn]

View ← showMessage(this.message) ← SocketClient ← new TextMessage(message) ← SocketConnectionHandler ← sendTextMessage("You aren't the current player") ← Controller

return

opt [the client tries to play when it's supposed to draw]

View ← showMessage(this.message) ← SocketClient ← new TextMessage(message) ← SocketConnectionHandler ← sendTextMessage("It's not the time to play a card") ← Controller

return

Controller → isPlayable(where, card) → PlayerField

alt [cannot play a card in that coordinate]

View ← showMessage(this.message) ← SocketClient ← new TextMessage(message) ← SocketConnectionHandler ← sendTextMessage("Unable to play the card, try again") ← Controller

SocketClient ← new YourTurn() ← yourTurn()

[else]

Controller → play(where, card) → PlayerField

opt [cannot play a card in that coordinate]

View ← showMessage(this.message) ← SocketClient ← new TextMessage(message) ← SocketConnectionHandler ← sendTextMessage("Unable to play the card, try again") ← Controller

return

Controller → addPoints(currentPlayer, score → ScoreBoard

View ← setGame(game) ← SocketClient ← new UpdateGameMessage(game) ← SocketConnectionHandler ← updateGame(game) ← Controller
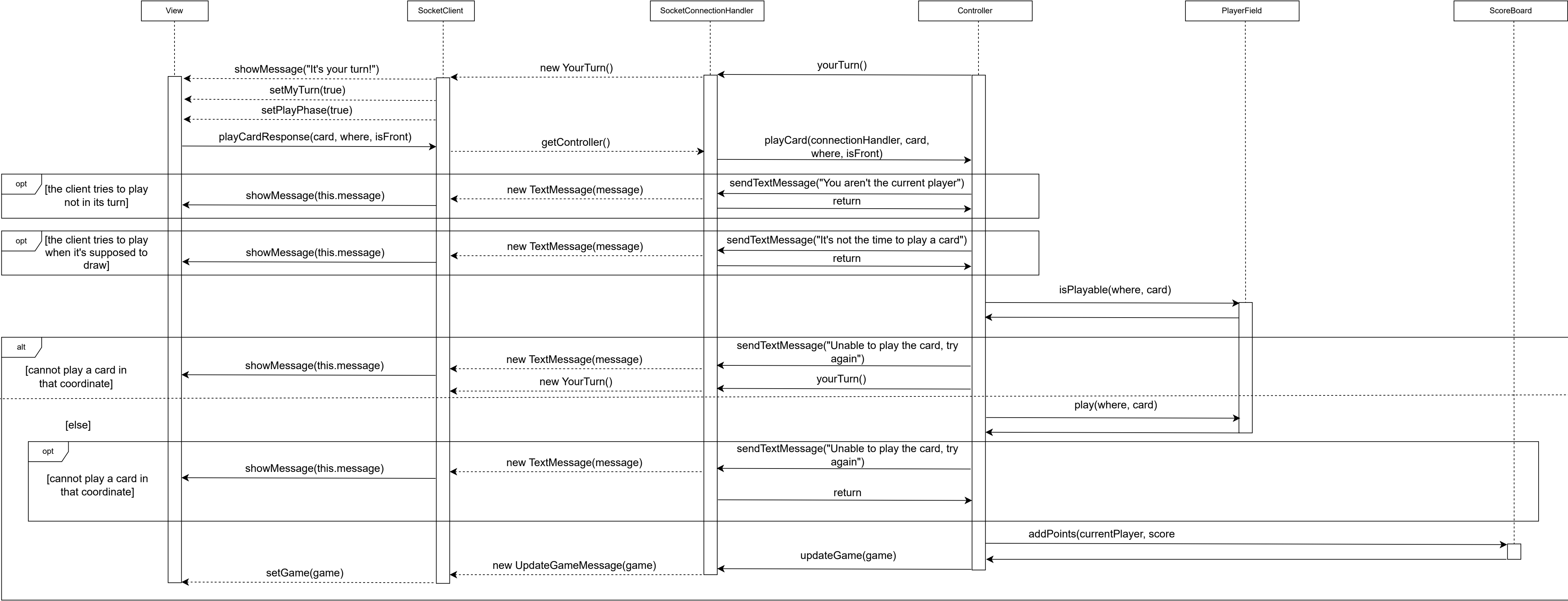
The controller notifies the client that its turn has started by setting
some booleans useful for filtering the client's possibile actions.
Then the client plays a card in a specific position.
The controller proceeds on doing some controls on the play and, if
successful, updates the game for every client.