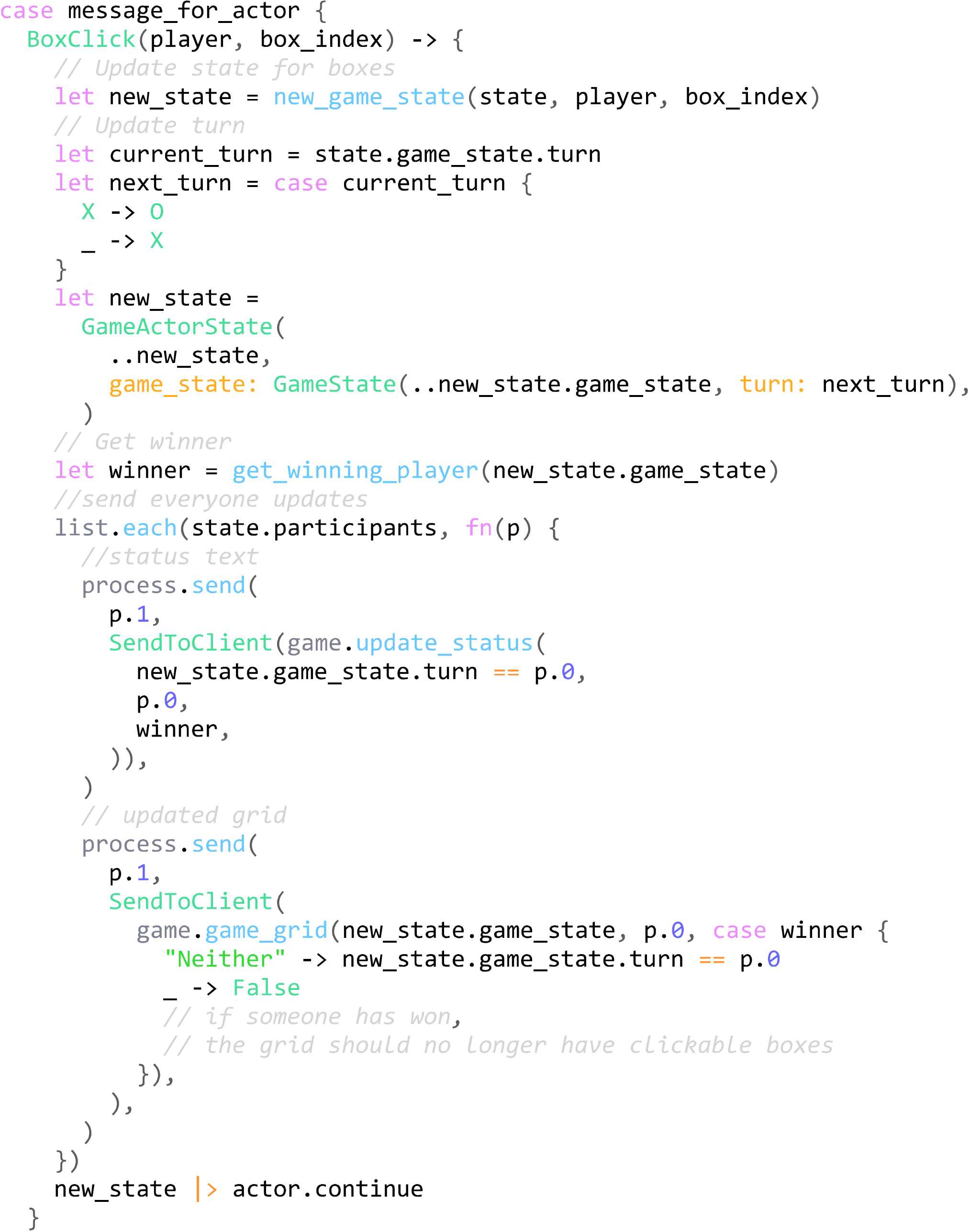

```
case message_for_actor {
  BoxClick(player, box_index) -> {
    // Update state for boxes
    let new_state = new_game_state(state, player, box_index)
    // Update turn
    let current_turn = state.game_state.turn
    let next_turn = case current_turn {
      X -> O
      _ -> X
    }
    let new_state =
      GameActorState(
        ..new_state,
        game_state: GameState(..new_state.game_state, turn: next_turn),
      )
    // Get winner
    let winner = get_winning_player(new_state.game_state)
    //send everyone updates
    list.each(state.participants, fn(p) {
      //status text
      process.send(
        p.1,
        SendToClient(game.update_status(
          new_state.game_state.turn == p.0,
          p.0,
          winner,
        )),
      )
      // updated grid
      process.send(
        p.1,
        SendToClient(
          game.game_grid(new_state.game_state, p.0, case winner {
            "Neither" -> new_state.game_state.turn == p.0
            _ -> False
            // if someone has won,
            // the grid should no longer have clickable boxes
          }),
        ),
      )
    })
    new_state |> actor.continue
  }
```