Collective data management

In this chapter we will learn how to:

1) Set a player’s turn

2) End a game, declare a winner, give each player a score and a certain percentage of the pot.

Tutorial introduction

Untill now we learned how to handle subjective data, such as what move did a player choose to do, this information was sent using the doStoreState function.  
In this tutorial we are going to learn about functions that are associated with collective data, these functions will start with a doAll marking them as actions everyone must agree on, such as whose turn it is, or who won. Therefore can be no disagreement in this matter, because those are collective facts.

doAll functions can only be called while processing gotStateChanged, gotMatchStarted, or gotMatchEnded (gotMatchEnded will not be discussed in this tutorial). Note that doAll functions must be triggered by something all the users got, and not by a timer or a mouse click.

Setting a player’s turn

In the third tutorial we learned how to send our move to the entire set of users, in this tutorial we will learn how to set the next player's turn.

1. To set a player’s turn, all players must call doAllSetTurn(userId:int, milliSecondsInTurn:int)  
   userId – the userId of the player whose turn it is.  
   milliSecondsInTurn – time to give the player to make his move in milliseconds, before declaring a profit, set -1 for default value.
2. Setting a turn, like the rest of the doAll functions will only be called as a result of the gotStateChanged, gotMatchStarted or gotMatchOver.
3. All players must call doAllSetTurn with the same values to make sure the decision is unanimous and that no one is cheating.

For more information about doAllSetTurn visit: <http://code.google.com/p/multiplayer-api/wiki/doAllSetTurn>

Ending the game

In our game every time the user calls the makeTurn function, the games logic checks if the game ended and if so, it dispatches a GameOverEvent.

To declare the game’s result we should end the match:

1. To end the game for all or some players, all players must call doAllEndMatch(finishedPlayers:Array)  
   finishedPlayers – is an Array of PlayerMatchOver elements, each representing a player in the game, which we want the game to end for.
2. Ending a match, like the rest of the doAll functions will only be called as a result of the gotStateChanged, gotMatchStarted or gotMatchOver.
3. All players must call doAllEndMatch with the same values to make sure the decision is unanimous and that no one is cheating.

To create a PlayerMatchOver entry we call the PlayerMatchOver.create(playerId:int, score:int, potPercentage:int)  
playerId – the user id of the player to whom this entry refers.  
score – the score the user got in the game.  
potPercentage – the percentage of the pot the player will get.

In our example the game ends for everyone when one of the players wins, but in your game you can make the game continue, ending the game only for the wining/losing player.

\*for more information about doAllEndMatch visit:

<http://code.google.com/p/multiplayer-api/wiki/doAllEndMatch>

http://code.google.com/p/multiplayer-api/wiki/PlayerMatchOver