UML event-based program flow

Date: 10th November 2021

START

START-VIEWMODEL

USER | VIEW | MAIN-VIEWMODEL | MODEL

setup request

binding

setup response

startViewmodel.createAndStartMatch()

change viewmodel

entities creation

mainViewmodel.createMatchFromSetupAndStartGame(createSetup())

match.startNewGame()

entities creation

gameStatusProperty=STARTED

gameStatusProperty=STARTED

OUT << "New game!"

currentPlayerProperty=blackPlayer

currentPlayerProperty=blackPlayer

OUT << "Turn of ${currentPlayer}"

currenPlayer.makeMove(currentGame)

waitForAValidMoveOfPlayer()
OUT << "Insert coordinates"

userMustPlaceStoneProperty=true

coordinatesRequiredToContinueProperty=true

only IF human player (not for CPU player)

if player is CPU, coordinates are not required to continue

IN >> coordinates

viewmodel.placeStoneFromUser(coordinates)

currentPlayer.placeStone(coordinates)

view does nothing if userMustPlaceStoneProperty==false

userMustPlaceStoneProperty=false

coordinatesRequiredToContinueProperty=false

occupy position on the board and change player turn.

lastMoveCoordinatesProperty=coordinates

lastMoveCoordinatesProperty=coordinates

OUT << ${boardWithLastCoordinates}

only IF game is ended

gameStatusProperty=ENDED

gameStatusProperty=ENDED

viewmodel.getSummaryOfGame()

match.getSummaryOfGame()

viewmodel.getSummaryOfGame(): RESPONSE

match.getSummaryOfGame(): RESPONSE

OUT << "Game ended, ${summary}"

viewmodel.isMatchEnded()

match.isEnded()

viewmodel.isMatchEndend(): RESPONSE

match.isEnded(): RESPONSE

viewmodel.isMatchEndedWithADraw()

match.isADraw()

only IF match is ended

viewmodel.isMatchEndedWithADraw(): RESPONSE

match.isADraw(): RESPONSE

OUT << "Extra game to play off?"

only IF match is ended with a draw

IN >> userWantsExtraGame

viewmodel.startExtraGame()

only IF user chooses to start an extra game

END

ELSE IF user chooses NOT to start an extra game

OUT << "Start new match?"

ELSE IF match is ended without a draw

IN >> userWantsNewMatch

viewmodel.startNewMatch()

IF user chooses to start a new match

END

ELSE IF user chooses NOT to start a new match

viewmodel.startNewGame()

ELSE IF match is NOT ended

ELSE IF game is NOT ended

change current player (change turn: black player swapped with withe player)

[SAME AS ABOVE]

currentPlayerProperty=currentPlayer

currentPlayerProperty=currentPlayer

OUT << "Turn of ${currentPlayer}"

currenPlayer.makeMove(currentGame)

[ITERATIONS CONTINUE UNTIL END OF PROGRAM]