2.1 Plan of action

* Setup github & install MC Visual Studio Code
* Build html files
* Create img-files
* Code javascript – client side
  + Player1Name - needs to be set to the nickname of player 1
  + Player2Name - needs to be set to the nickname of player 2
  + Player1type – needs to be set to the type (Opponent / You (/ AI?))
  + Player1type – needs to be set to the type (Opponent / You (/ AI?))
  + Player1score – needs to display the score(amount of stones) of player 1
  + Player2score – needs to display the score(amount of stones) of player 2
  + Clock – needs to display the time
  + Board – generated using javasript
  + Board cells – clickable, on click:
    - Send move to server
    - Check for valid move
    - If valid : make move
    - Send move to both clients
    - If not valid: Ask for other move
* Code javascript – server side
* Test & enhance / fix errors
* Make style.css file
* Test & enhance

2.2

* Gamestate:
  + Var scorePlayer1
  + Var scorePlayer2
  + Var state (starting / whiteTurn / blackTurn / finished)
  + Var validMovesRemainingBlack
  + Var validMovesRemainingWhite
  + Var nickNamePlayer1
  + Var nickNamePlayer2
  + Var expectingMoveBoolean
* Game:
  + Var id
  + Var clock
  + Var socketPlayer1
  + Var socketPlayer2
  + Var gameState
  + Var Board
  + Public arrDirections checkNextToOpposingColor(Color, Location)
  + Public arrLocations findFlippedPieces(board, arrayDirections, location)
* Board:
  + Var pieces (2-dimensional array(-1, 0, 1))
  + Public void setPieces(arrLocations, color)
* Location:
  + X
  + Y
  + Public int left
  + Public int right
  + Public int top
  + Public int bottom
  + Public int topRight
  + Public int topLeft
  + Public int bottomRight
  + Public int bottomLeft
* Statistics
  + Var gamesPlayed
  + Var blackWins
  + Var whiteWins
  + Var playersOnline
  + Var ongoingGames
* Leaderbord
  + Var leaderboard (2-dimensional array, nicknames and scores)