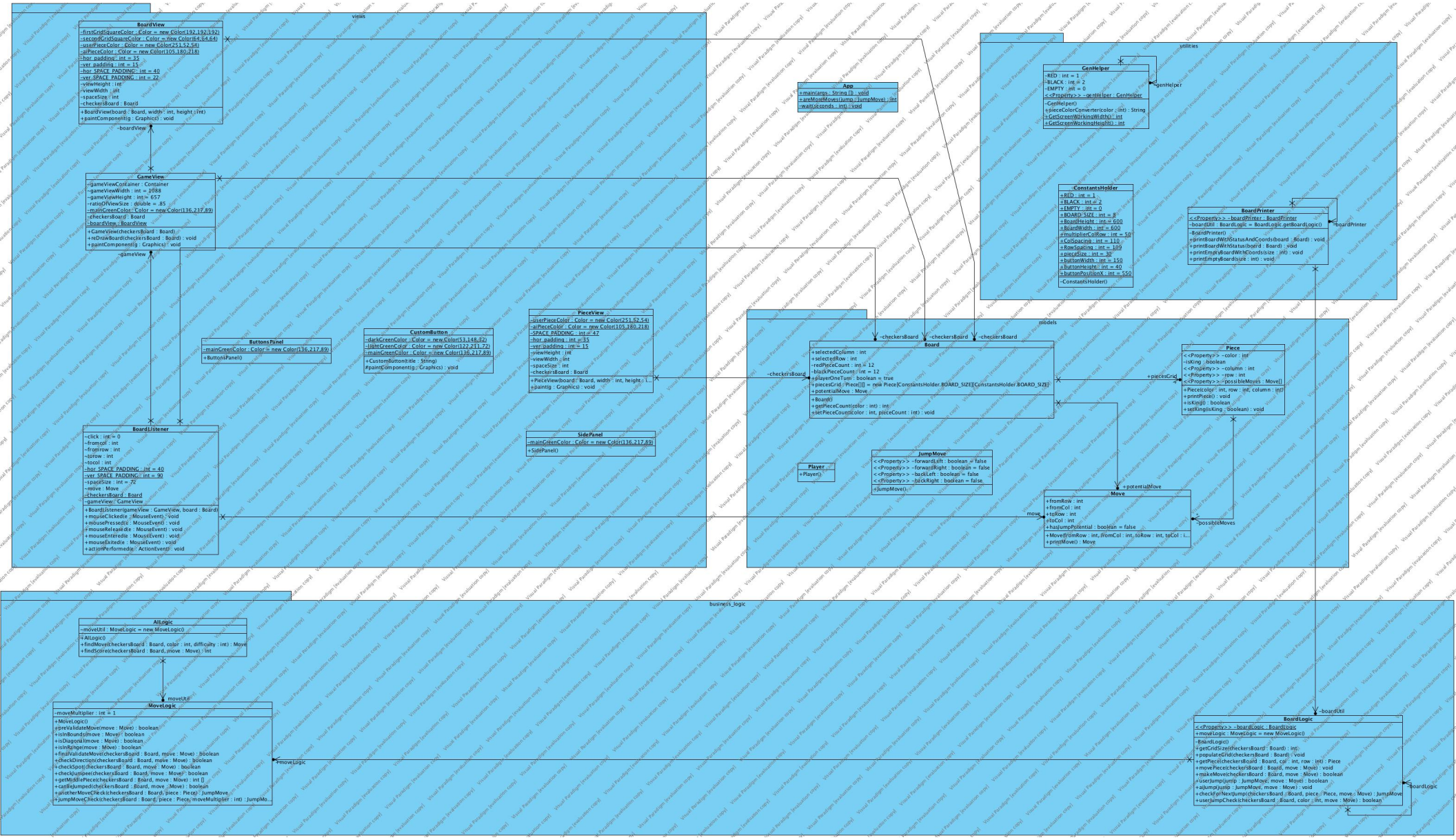
views
BoardView
-firstGridSquareColor : Color = new Color(192,192,192)
-secondGridSquareColor : Color = new Color(64,64,64)
-userPieceColor : Color = new Color(251,52,54)
-aiPieceColor : Color = new Color(105,180,218)
-hor_padding : int = 35
-ver_padding : int = 15
-hor_SPACE_PADDING : int = 40
-ver_SPACE_PADDING : int = 22
-viewHeight : int
-viewWidth : int
-tspaceSize : int
-checkersBoard : Board
+BoardView(board : Board, width : int, height : int)
+paintComponent(g : Graphics) : void
-boardView
GameView
-gameViewContainer : Container
-gameViewWidth : int = 1088
-gameViewHeight : int = 657
-ratioOfViewSize : double = .85
-mainGreenColor : Color = new Color(136,217,89)
-checkersBoard : Board
-boardView : BoardView
+GameView(checkersBoard : Board)
+reDrawBoard(checkersBoard : Board) : void
+paintComponent(g : Graphics) : void
-gameView
ButtonsPanel
-mainGreenColor : Color = new Color(136,217,89)
+ButtonsPanel()
CustomButton
-darkGreenColor : Color = new Color(53,148,82)
-lightGreenColor : Color = new Color(122,211,72)
-mainGreenColor : Color = new Color(136,217,89)
+CustomButton(title : String)
#paintComponent(g : Graphics) : void
PieceView
-userPieceColor : Color = new Color(251,52,54)
-aiPieceColor : Color = new Color(105,180,218)
-SPACE_PADDING : int = 47
-hor_padding : int = 35
-ver_padding : int = 15
-viewHeight : int
-viewWidth : int
-spaceSize : int
-checkersBoard : Board
+PieceView(board : Board, width : int, height : i...
+paint(g : Graphics) : void
BoardListener
-click : int = 0
-fromCol : int
-fromRow : int
-toRow : int
-toCol : int
-hor_SPACE_PADDING : int = 40
-ver_SPACE_PADDING : int = 90
-spaceSize : int = 72
-move : Move
-checkersBoard : Board
-gameView : GameView
+BoardListener(gameView : GameView, board : Board)
+mouseClicked(e : MouseEvent) : void
+mousePressed(e : MouseEvent) : void
+mouseReleased(e : MouseEvent) : void
+mouseEntered(e : MouseEvent) : void
+mouseExited(e : MouseEvent) : void
+actionPerformed(e : ActionEvent) : void
SidePanel
-mainGreenColor : Color = new Color(136,217,89)
+SidePanel()
App
+main(args : String[]) : void
+areMoveMoves(jump : JumpMove) : int
-waitCoords : int[] : void
utilities
GenHelper
-RED : int = 1
-BLACK : int = 2
-EMPTY : int = 0
<<Property>> -getHelper : GenHelper
-GenHelper()
+pieceColorConverter(color : int) : String
+GetScreenWorkingWidth() : int
+GetScreenWorkingHeight() : int
-genHelper
ConstantsHolder
+RED : int = 1
+BLACK : int = 2
+EMPTY : int = 0
+BOARD_SIZE : int = 8
+BoardHeight : int = 600
+BoardWidth : int = 600
+multiplierColRow : int = 58
+ColSpacing : int = 110
+RowSpacing : int = 109
+pieceSize : int = 30
+buttonWidth : int = 150
+buttonHeight : int = 40
+buttonPositionX : int = 550
-ConstantsHolder()
BoardPrinter
<<Property>> -boardPrinter : BoardPrinter
-boardUtil : BoardLogic = BoardLogic.getBoardLogic()
-BoardPrinter()
+printBoardWithStatusAndCoords(board : Board) : void
+printBoardWithStatus(board : Board) : void
+printEmptyBoardWithCoords(size : int) : void
+printEmptyBoard(size : int) : void
-boardPrinter
models
Board
+selectedColumn : int
+selectedRow : int
-redPieceCount : int = 12
-blackPieceCount : int = 12
+playerOneTurn : boolean = true
+piecesGrid : Piece[][] = new Piece[ConstantsHolder.BOARD_SIZE][ConstantsHolder.BOARD_SIZE]
+potentialMove : Move
+Board()
+getPieceCount(color : int) : int
+setPieceCount(color : int, pieceCount : int) : void
-checkersBoard
-checkersBoard
-checkersBoard
-checkersBoard
+piecesGrid
Piece
<<Property>> -color : int
-isKing : boolean
<<Property>> -column : int
<<Property>> -row : int
<<Property>> -possibleMoves : Move[]
+Piece(color : int, row : int, column : int)
+printPiece() : void
+isKing() : boolean
+setKing(isKing : boolean) : void
Player
+Player()
JumpMove
<<Property>> -forwardLeft : boolean = false
<<Property>> -forwardRight : boolean = false
<<Property>> -backLeft : boolean = false
<<Property>> -backRight : boolean = false
+jumpMove()
+potentialMove
Move
+fromRow : int
+fromCol : int
+toRow : int
+toCol : int
+hasJumpPotential : boolean = false
+Move(fromRow : int, fromCol : int, toRow : int, toCol : i...
+printMove() : Move
move
+possibleMoves
business_logic
AiLogic
-moveUtil : MoveLogic = new MoveLogic()
+AiLogic()
+findMove(checkersBoard : Board, color : int, difficulty : int) : Move
+findScore(checkersBoard : Board, move : Move) : int
-moveUtil
MoveLogic
-moveMultiplier : int = 1
+MoveLogic()
+preValidateMove(move : Move) : boolean
+isInBounds(move : Move) : boolean
+isDiagonal(move : Move) : boolean
+isInRange(move : Move) : boolean
+finalValidateMove(checkersBoard : Board, move : Move) : boolean
+checkDirection(checkersBoard : Board, move : Move) : boolean
+checkSpot(checkersBoard : Board, move : Move) : boolean
+checkUnique(checkersBoard : Board, move : Move) : boolean
+getMiddlePiece(checkersBoard : Board, move : Move) : int []
+canBeJumped(checkersBoard : Board, move : Move) : boolean
+anotherMoveCheck(checkersBoard : Board, piece : Piece) : JumpMove
+jumpMoveCheck(checkersBoard : Board, piece : Piece, moveMultiplier : int) : JumpMo...
-moveLogic
-boardUtil
BoardLogic
<<Property>> -boardLogic : BoardLogic
+moveLogic : MoveLogic = new MoveLogic()
-BoardLogic()
+getGridSize(checkersBoard : Board) : int
+populateGrid(checkersBoard : Board) : void
+getPiece(checkersBoard : Board, col : int, row : int) : Piece
+movePiece(checkersBoard : Board, move : Move) : void
+makeMove(checkersBoard : Board, move : Move) : boolean
+userJump(jump : JumpMove, move : Move) : boolean
+aiJump(jump : JumpMove, move : Move) : void
+checkForNextJump(checkersBoard : Board, piece : Piece, move : Move) : JumpMove
+userJumpCheck(checkersBoard : Board, color : int, move : Move) : boolean
-boardLogic