Done	To-Do
- Make it executable everywhere DONE! #Board - input of team names DONE! -initial positions can be added NO NEED! - Get enough information a about pygame and start building it ABORTED! - flip baord method and fix working DONE! -To string methods for all the classes DONE! - Validation of move 30% DONE! #Main - Initiate DONE! -- Connect GUI also DONE! -Connect with Board and use methods DONE! #AI - toStrng no need! - Plan out minimax of chess board SOME! -returns names and teams done!	-Just use board direct updation -Start Controller method of input and output #Board - Complete validation of moves #Main -Then act this as a tunnel between Board, AI and GUI -start finishing player to player #AI -Just use board direct updation -Complete the non working minimax

Time-line:

Task	October	November	December
Main-(10)	-	10%	-
Game-(20)	~~20%~~	-	-
GUI-(10)	~~2%~~	10%	-
AI-(60)	~~5%~~	50%	60%

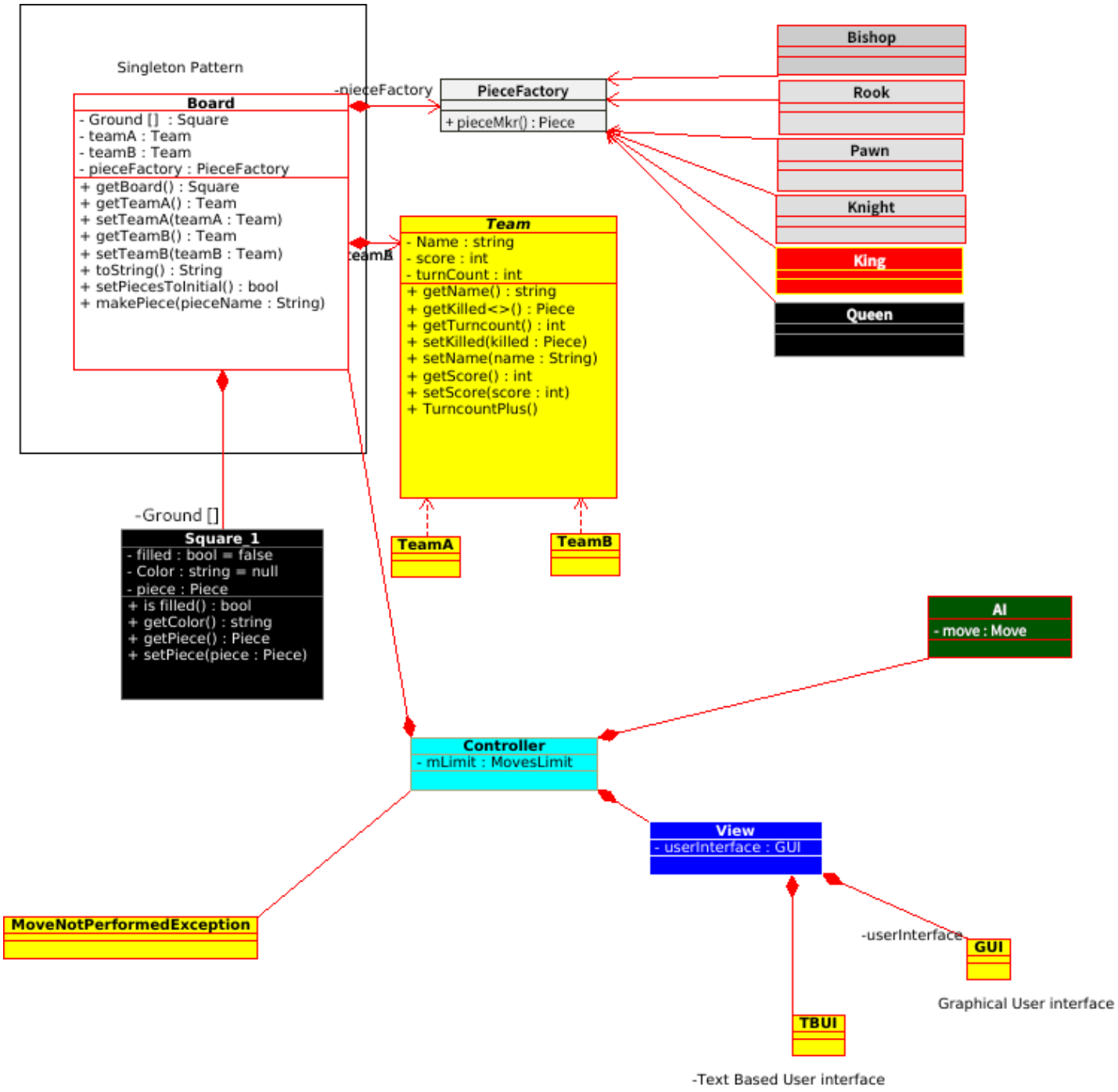
Notes:

- Don’t just think about losing pieces, Think about losing specific pieces. Give priorities or preferences.

- Check and go to recurse for Alpha and beta.

- Whole Board would be passed as parameter in Ai class

- we might need to construct the gui inside board which would be moving methods because it is just printing nothing


Singleton Pattern
Board
- Ground [] : Square
- teamA : Team
- teamB : Team
- pieceFactory : PieceFactory
+ getBoard() : Square
+ getTeamA() : Team
+ setTeamA(teamA : Team)
+ getTeamB() : Team
+ setTeamB(teamB : Team)
+ toString() : String
+ setPiecesToInitial() : bool
+ makePiece(pieceName : String)
-pieceFactory
PieceFactory
+ pieceMkr() : Piece
Bishop
Rook
Pawn
Knight
King
Queen
Team
- Name : string
- score : int
- turnCount : int
+ getName() : string
+ getKilled<>() : Piece
+ getTurncount() : int
+ setKilled(killed : Piece)
+ setName(name : String)
+ getScore() : int
+ setScore(score : int)
+ TurncountPlus()
-Ground []
Square_1
- filled : bool = false
- Color : string = null
- piece : Piece
+ is filled() : bool
+ getColor() : string
+ getPiece() : Piece
+ setPiece(piece : Piece)
TeamA
TeamB
AI
- move : Move
Controller
- mLimit : MovesLimit
View
- userInterface : GUI
MoveNotPerformedException
-userInterface
GUI
Graphical User interface
TBUI
-Text Based User interface