**UML Diagram**

|  |
| --- |
| **Timer** |
| Server: ChessServer |
| +Timer(ChessServer)  +run ():void |

*Srikanth M*

|  |
| --- |
| **Locker** |
| availableUsers: Hashtable<String, ChessServer> |
| +Locker(Hashtable<String, ChessServer)  +addPlayer (String , ChessServer ):boolean  +deleteUser(String):ChessServer  +getUser(String):ChessServer  +contains(String):Boolean  +size():int  +keys():Enumeration<String> |

*PremSagar Reddy*

|  |
| --- |
| **ChessServer** |
| client\_Socket:Socket  out: PrintWriter  in: BufferedReader  lockAvailableUsers: Locker  playerName: String  rendezvous: Boolean  opponentPlayer: ChessServer  reply: String  timeUp: Boolean  my\_color: String  board\_manager: chessboard  finish: Boolean  my\_turn: Boolean |
| +ChessServer(Socket,Locker)  +getPlayerName():String  +timeUp():void  +getBoard():chessboard  +wait\_4\_other\_players ():Boolean  +stop\_to\_wait (ChessServer):void  +awaitingReply():void  +reply(String): void  +chooseUser():void  +isRendezving ():Boolean  +send\_reply ():Boolean  +send\_my\_color\_code (String): String  +setMyColor(String):void  +sendBoard():void  +waitMyTurn():void  +nowYouCanPlay():void  +run():void |

*Vinoth Naik*

|  |
| --- |
| **OnlineChess** |
| Port:int  server\_IP\_Address: String  out: PrintWriter  stdIn: BufferedReader  total\_time: long |
| + ChessServer():void  + chooseColor():String  + getMove():void  + choosPiece():void  + time(time): String  + Client():void  + main(String):void |

*Srikanth M*

|  |
| --- |
| **Chessboard** |
| chessboard: String[][]  player1:String  player2:String  toss\_coin: Boolean  white\_castling\_left: Boolean  white\_castling\_right: Boolean  black\_castling\_left: Boolean  black\_castling\_right:Boolean  pawn\_promoted: Boolean  promotion\_row:int  promotion\_col:int  white\_Check:Boolean  black\_Check: Boolean  threatening\_row:int  threatening\_col:int  white\_king\_row:int  white\_king\_col:int  black\_king\_row:int  black\_king\_col:int  white\_check\_mate:int  black\_check\_mate:int  whiteTime:String  blackTime:String |
| + chessboard()  + getTime(Boolean): String  + setTime(Boolean,String):void  + is\_pawn\_promoted():Boolean  + win(Boolean): int  + makePromotion(String, Boolean): void  + addPlayer(String): void  + isPlayer1(String): Boolean  + imWhite(String): Boolean  + setPlayer1(String): void  + setPlayer2(String): void  + initBoard():void  + get\_white\_board():String[][]  + get\_black\_board():String[][]  + valid\_rook\_movement(int,int,int,int): Boolean  + check\_rook\_movement(int,int,Boolean): void  + valid\_bishop\_movement(int,int,int,int): Boolean  + check\_bishop\_movement(int,int,Boolean): void  + valid\_queen\_movement(int,int,int,int): Boolean  + check\_queen\_movement(int,int,Boolean):void  + valid\_king\_movement(int,int,int,int,Boolean):Boolean  + valid\_knight\_movement(int,int,int,int): Boolean  + is\_check\_knight(int,int,boolean): Boolean  + check\_kniight\_movement(int,int,Boolean): void  + valid\_pawn\_movement(int,int,int,int,Boolean): Boolean  + check\_pawn\_movement(int,int,Boolean): void  + castling(String,Boolean):Boolean  + check\_onestep\_move(int,int,Boolean):Boolean  + valid\_king\_move(Boolean):Boolean  + check\_capture(Boolean): Boolean  + can\_block\_rook(Boolean): Boolean  + can\_block\_in(int,int,Boolean):boolean  + can\_block\_bishop(Boolean): Boolean  + can\_block\_queen(Boolean): Boolean  + canBlock(Boolean): Boolean  + is\_check\_mate(Boolean):void  + is\_valid\_move(String,string,Boolean):Boolean |

*SaiKrishna Kovi & Srikanth M*