Julliana do Couto Almeida 13/0060119

Foram criadas as classes: Bispo, Cavalo, Peao, Piece, Rainha, Rei, Square, Torre. Dentro da classe de cada peça tem as restrições dos movimentos, incluindo as direções e as exceções associadas a cada peça.Por exemplo, ao pensar na classe cavalo, o movimento dessa peça é um L, dois quadrados e uma direção e avança mais um quadrado na direção perpendicular, foram feitas exceções para as bordas pensando em cada possível "L" que pode ser movimento.

No caso do peão, ele pode se movimentar um ou dois quadrados na primeira jogada para frente, enquanto nas outras jogadas apenas uma vez, foi pensando nas exceções de borda. O bispo move em diagonal, por isso foi feito 4 for´s varrendo cada diagonal implementada dos seus limites das bordas. A Torre pode se movimentar verticalmente ou horizontalmente, para projetar esse movimento foi feito dois for's com seus limites, um ate a quantidade de linhas e o outro, a quantidade de colunas. O rei foi criado avançando um quadrado em todas as direções usando oito if's para cada posição correspondente. Por fim, o movimento da rainha, é o conjunto da lógica usada para a torre e para o bispo.

Na controladora, foi definido os possíveis movimentos devem ser marcados de vermelho, para fazer esse comando foi necessário dentro da função o quadrado selecionado (selectedSquare) mudar a cor de todos os possíveis movimentos, no caso para vermelho, para isso em cada posição possível mudou-se a cor para "colorMove".E por meio do "unselectedSquare" quadrado não selecionado, ao passar o mouse sobre as posições vermelhas, não muda de cor, ou seja permanece de vermelho. Além disso, quando passar o mouse sobre os possíveis movimentos, eles continuarão marcados de vermelho.

O método "onselectEvent" define que após um time realizar uma jogada, o seu adversário deve realizar a próxima e assim vai, impedindo que um mesmo jogador jogue mais de uma vez.

Se a ultima jogada foi do time de cima, então a próxima jogada vai ser do time de baixo.