NOTE: While the game screen interacts normally with the game, the same way as the regular chess game does, in Fire Chess the FireOverlay class is simultaneously acting on the BurningChess instance in order to update it according to how the fire has spread. Most of the logic is contained within the FireOverlay class, simply because which tiles should be incinerated is so intrinsically tied to how the graphics are displayed.

Otherwise, the game is played the same way as before. There are other sequence diagrams for that. The only difference is that BurningChess also filters available moves out if they are in the list of unreachable positions, but the UI doesn't notice the difference.

FireOverlay   **BurningChess**

fireTimer(float delta)

increment a counter for the player whose turn it currently is

getWhiteTimer()

get the white player counter

NOTE: While removing a piece briefly makes all tiles reachable again, this is quickly countered by the next time the FireOverlay interacts with the game (basically the next frame), at which point it will kill all the tiles consumed by fire again, so in practice this only revives tiles in the immediate vicinity of the incinerated piece.

calculate how far the white fire has spread

getBlackTimer()

get the black player counter

calculate how far the black fire has spread

checkCircle()

if enough control points on tile consumed by fire, killTile(Position pos)

make tile unreachable

if piece was on tile killPiece(IChessPiece piece)

piece removed and all unreachable positions reset

return piece that was removed, if any

if piece was removed, create a "splash" in the fire shader that makes surrounding tiles safe for a little while longer

unreachable positions are now updated