Control

int: UP = -1, DOWN =1, LEFT = -1, RIGHT = 1
Piece selectedPiece
int: releasedAtX, oldModelX, enemyX, releasedAtY..
boolean: mandatoryFirstKill, mustCompleteCombo..
Player: playerthisTurn
boolean: killPossible

-boolean selectedPieceCanMoveInDirection(int dx, int dy,
boolean isInKillCombo, boolean doHighlighting)
-boolean canKill(Piece piece)
-boolean canKillThisDirection(Piece piece, int dx, int dy)
boolean tileIsFree(int dx, int dy)
-void highLightTileAt(int dx, int dy)
boolean hasLegalEndPosition(Piece piece)
boolean checkPossibleKills( )
void killCombo( )

TopLevelControl

AI: ai

---

void startThisPiece(Piece piece)
void updateThisPiece(Piece piece, MouseEvent mouse)
void finishThisPiece(Piece piece)
void continueThisPiece(Piece piece)
void updateAllPieces( )
void updateAllTiles( )
void startNextTurn( )
void runAI( )
double toViewCoord(int i)

IO

---

void saveGameData( )
void loadGameData( )

AI

Piece: currentPiece
Tile: targetTile
Player: player
int: difficulty
ArrayList<Tile> possibleTargetTiles

---

void execute( )
void choosePiece( )
void chooseTile( )
void simulateDrag( )
void simulateRelease( )

ResourceManager

+void save(Serializable data, String
filename)
+Object load(String filename)