

CHESSE ENGINE GAMBYTE

Player Name

PLAY



White



Random



Black

HISTORY

History

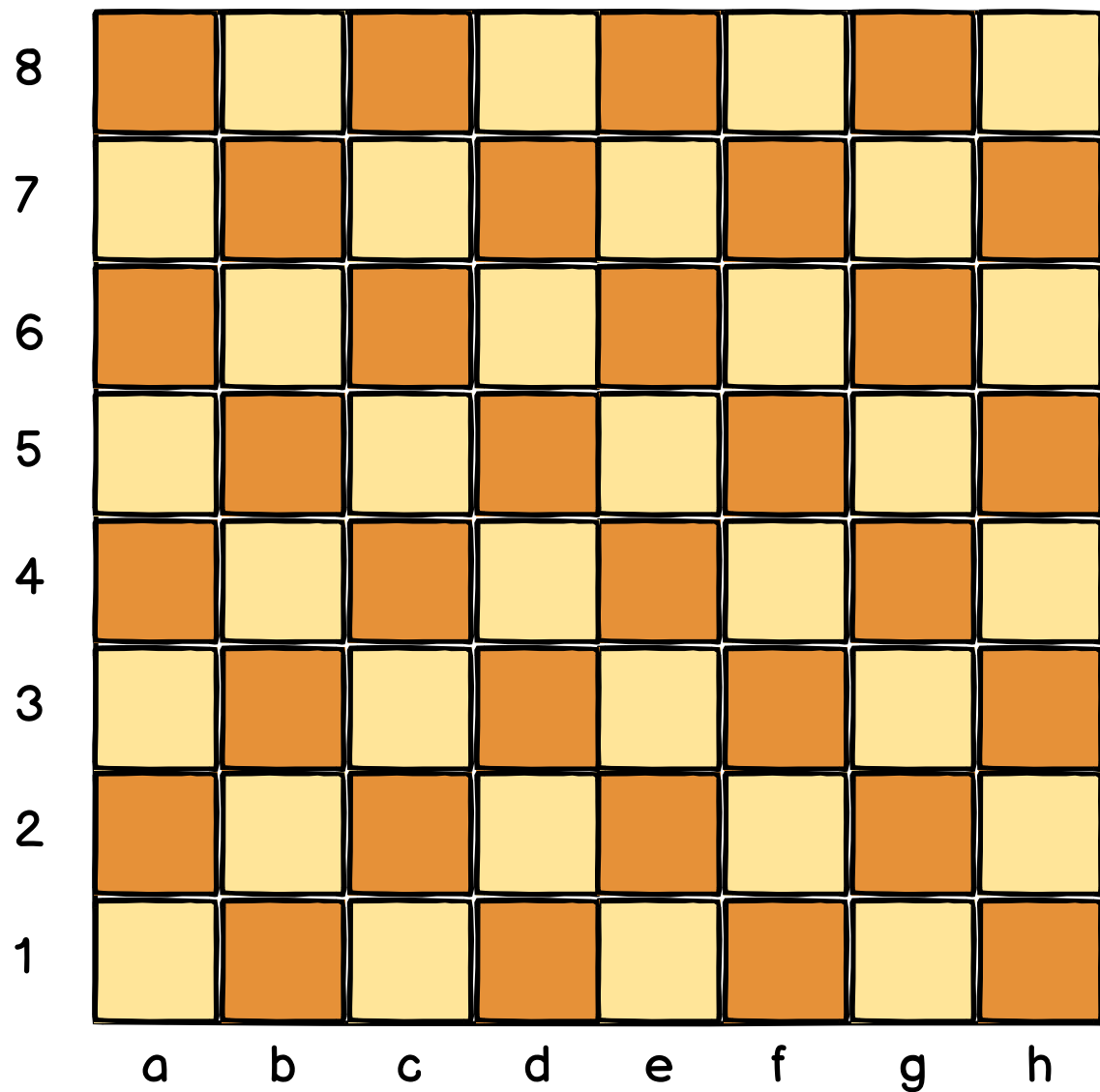
×

▲

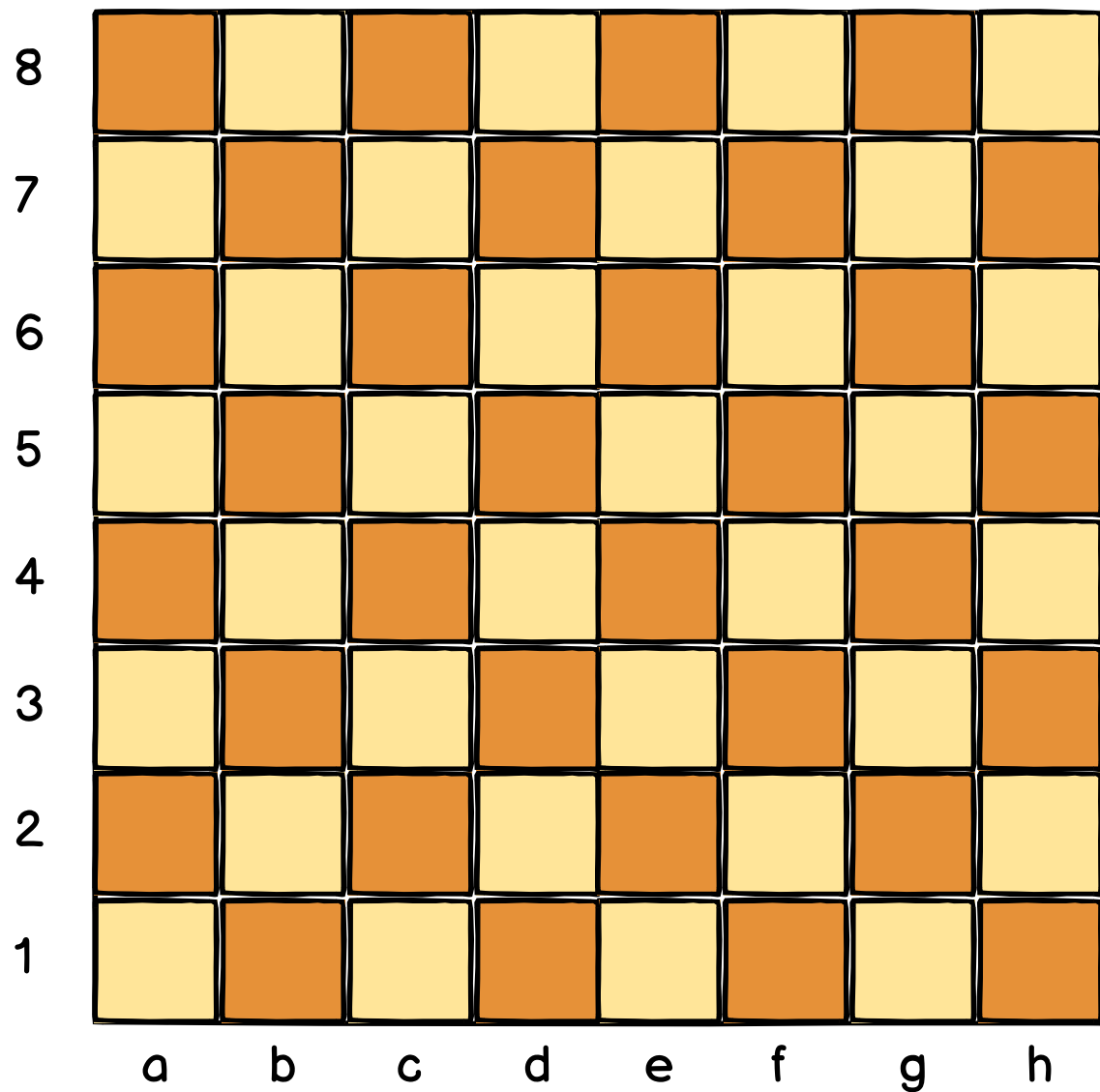
▼

Player	Winner	Moves	Result	Date	
Player Name	AI	35	Checkmate	23-02-2021	<div>Visualize game</div>
Player Name	AI	21	Resigned	24-02-2021	<div>Visualize game</div>

Menu



«	<	>	»
Move 1	d4	e4	
Move 2	Nf3	Nc6	
Move 3	Bc4	Nf6	
⏮		✖	



Evaluation

+ 5.2

«

<

>

»

Move 1

d4

e4

Move 2

Nf3

Nc6

Move 3

Bc4

Nf6

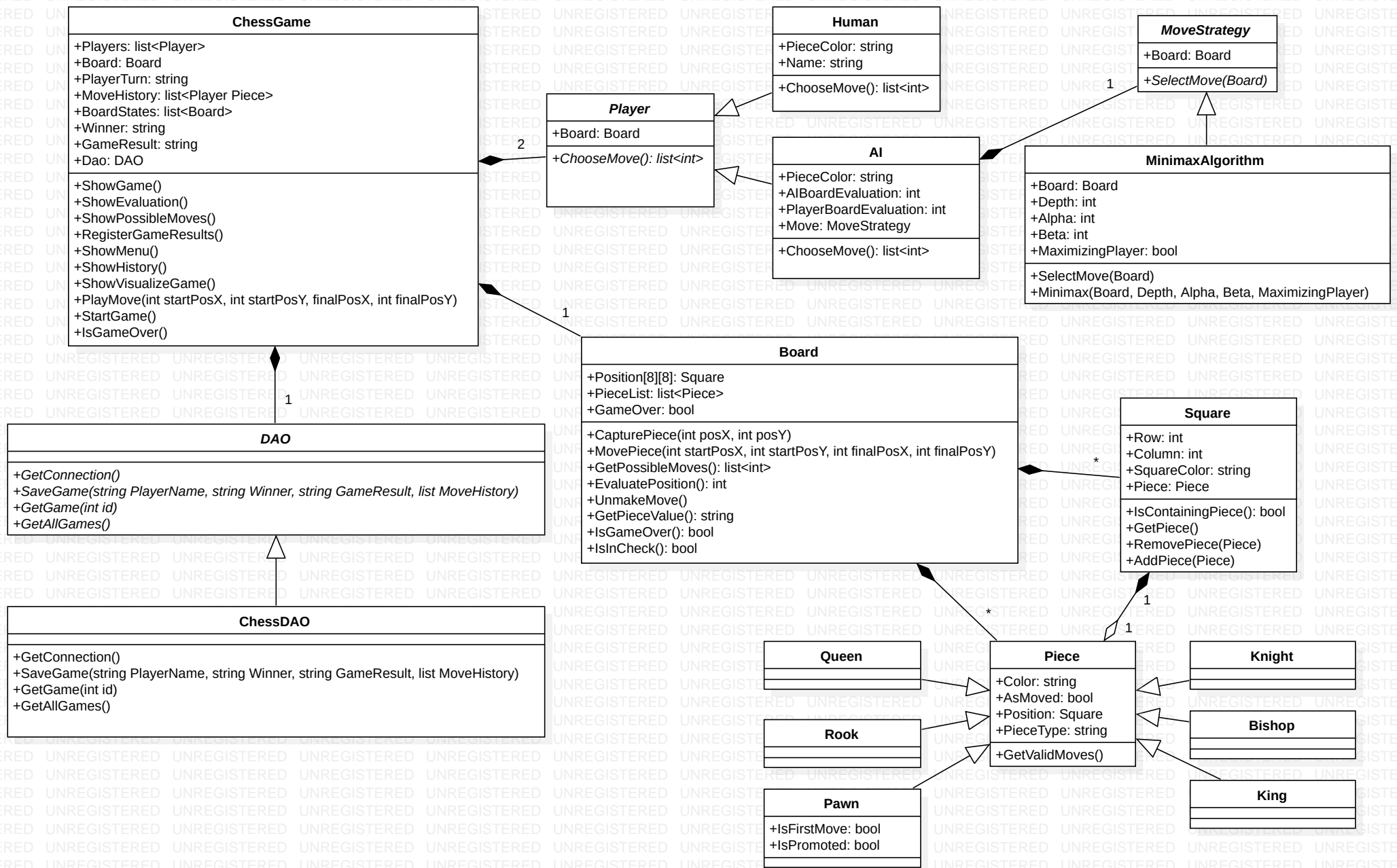


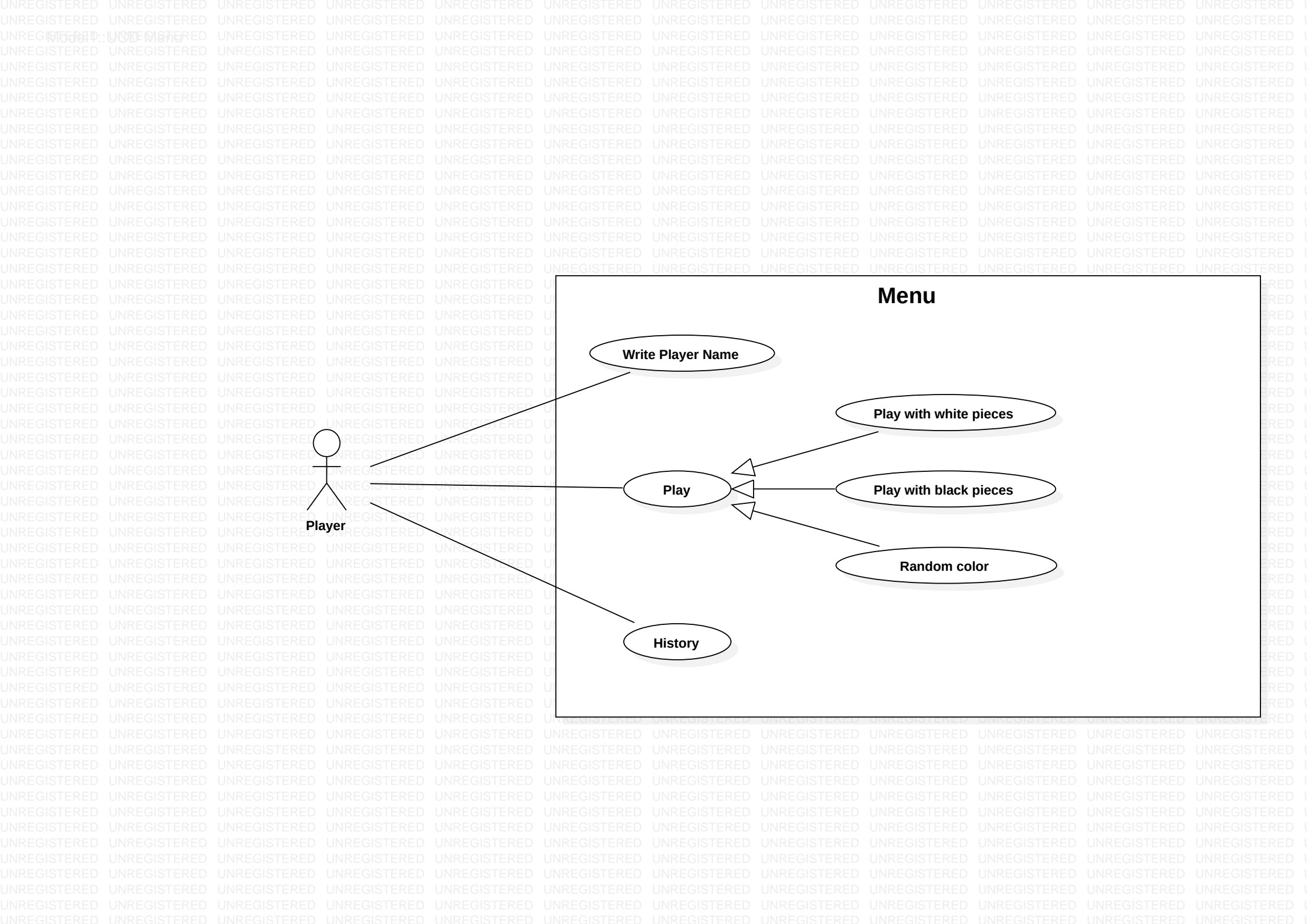
Présentation des patrons de conceptions

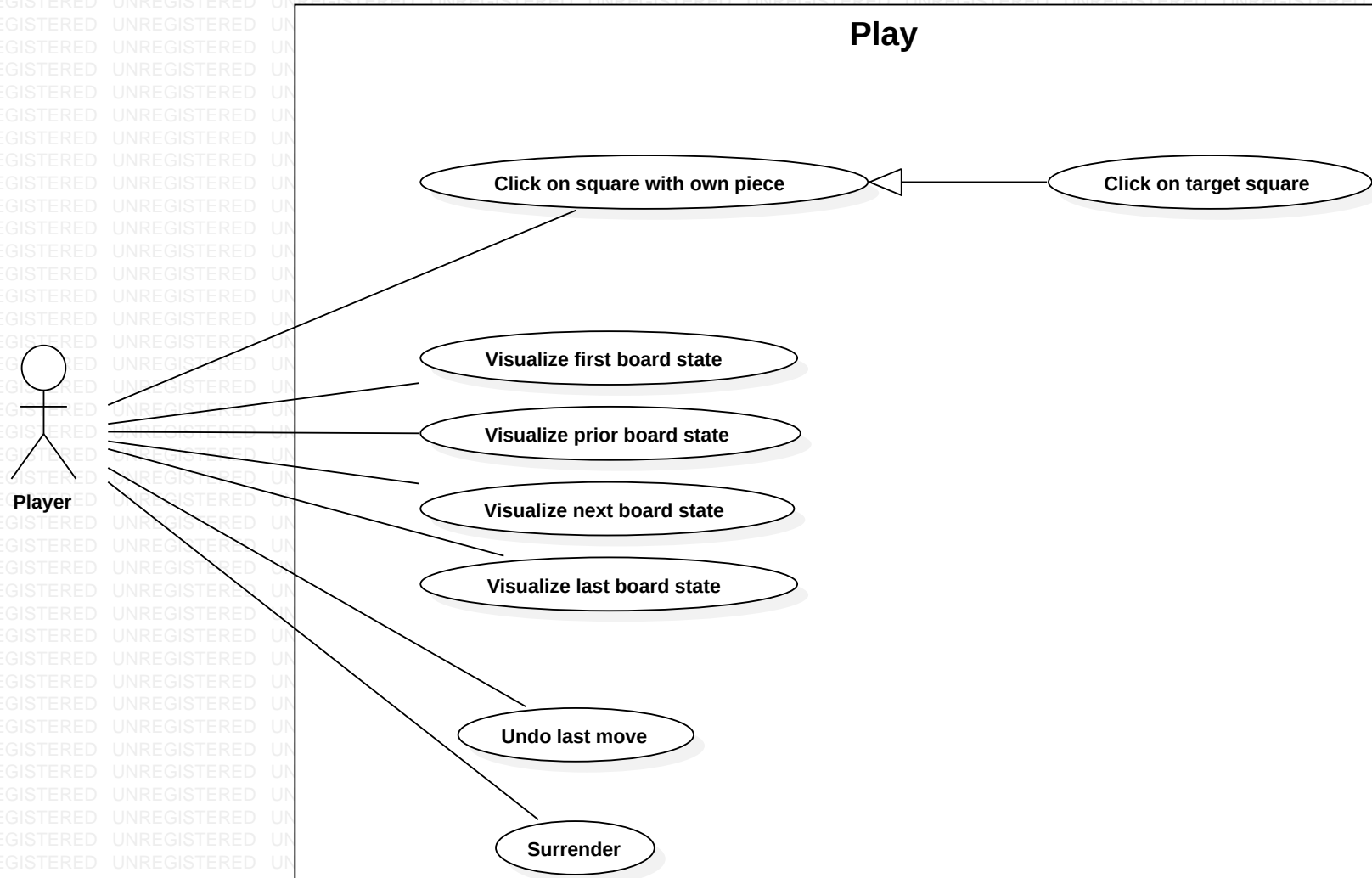
Le premier patron de conception pour le jeu d'échec sera de type stratégie. Le but de ce patron est d'encapsuler une partie de l'algorithme de manière à obtenir une plus grande flexibilité. Cela permet à cette partie du projet d'être séparé du reste du projet et d'être interchangeable avec des éléments provenant de la même classe abstraite, définissant leur structure principale des méthodes à implémenter mais en laissant la logique à l'intérieur de ceux-ci libre au développement. Dans le contexte du jeu d'échec contenant une intelligence artificielle en temps que joueur, le fait d'utiliser un patron de type stratégie permet de rendre flexible et interchangeable l'algorithme responsable de l'évaluation et de la prise de décision du coup à faire selon les pièces en jeu. Il y a plusieurs types d'algorithme de recherche et le projet ne présente qu'une seule méthode, de type minimax, permettant vérifier les positions offrant les meilleurs résultats pour soi-même tout en choisissant la position donnant le moins de chances à l'adversaire. Cette façon spécifique de chercher la position optimale n'est pas la seule bonne façon. Le patron de stratégie permettrait donc d'implémenter une autre version de l'algorithme de recherche et d'offrir l'option de choisir lequel est désiré, tout est ne touchant aucunement au reste du projet. Il est aussi possible de changer la méthode de recherche directement à l'intérieur de jeu même, rendant cette méthode interactive et multipliant les possibilités de jeu.

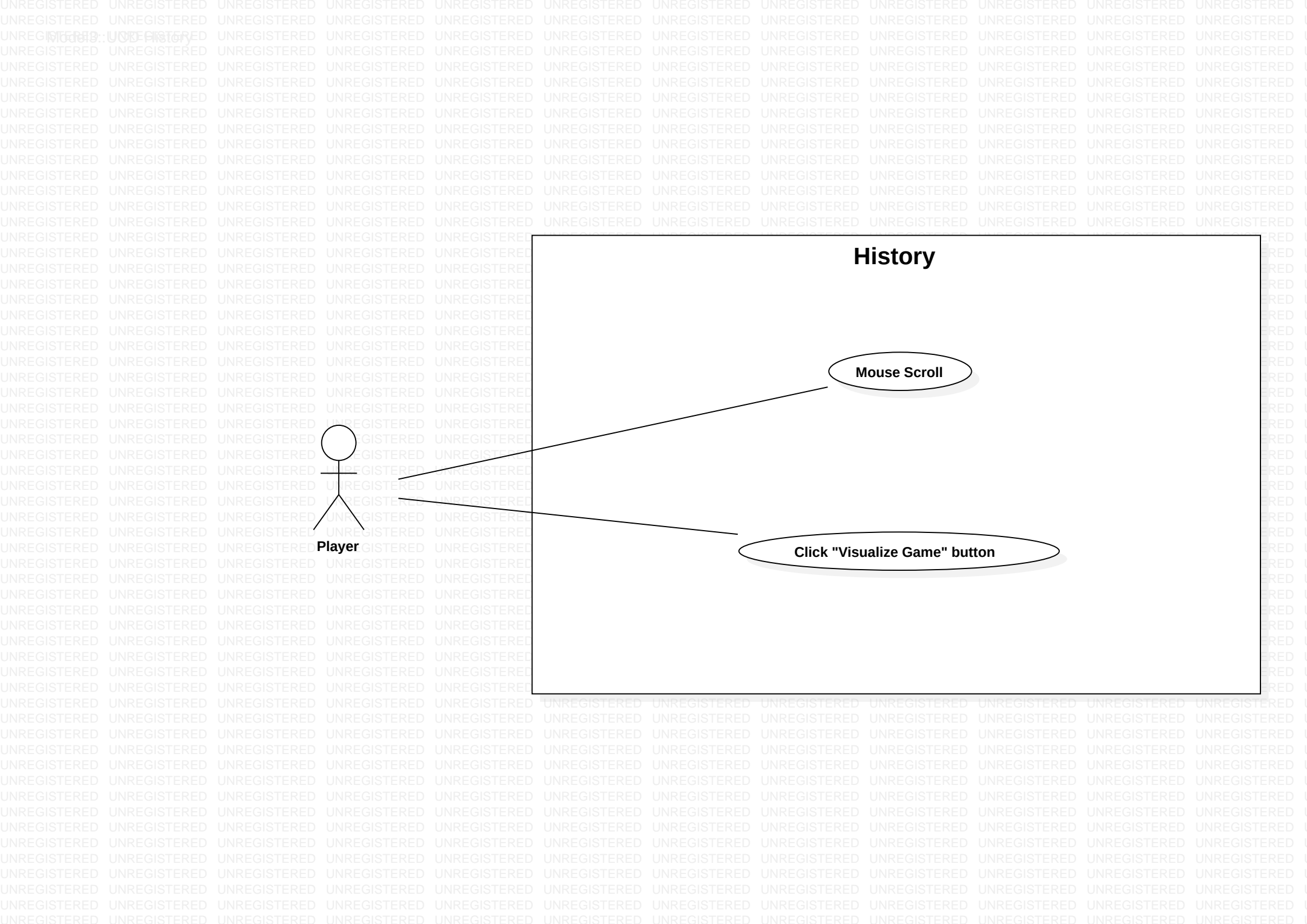
Le deuxième patron de conception est le concept de DAO (*Data Access Object*). Le but est d'effectuer une séparation entre les données du jeu et l'accès aux données dans la base de données. La base de données étant une entité externe au programme, le fait d'utiliser un DAO permet de simplifier et de restreindre l'accès à la base de données en limitant les fonctions à ce que le programme nécessite seulement. De plus, similaire au patron stratégie, le fait d'isoler une partie de l'algorithme à travers une classe abstraite offrant un modèle à suivre permet d'offrir une flexibilité au niveau du type de base de données, tant que les méthodes nécessaires à la communication sont respectées. Il serait donc possible de modifier la technologie de base données utilisée sans affecter le reste du programme. La fonction faisant appel au DAO ne sait pas comment les données seront enregistrées. Peu importe la manière dont les données sont gérées par la suite, la façon d'interagir avec le DAO demeure inchangée. Pour enregistrer mes parties d'échecs, MongoDB comme base de données. Si un jour, la décision est de changer pour MySQL, l'appel à la fonction du DAO et les informations envoyés resteront les mêmes.

Model Game's Classes









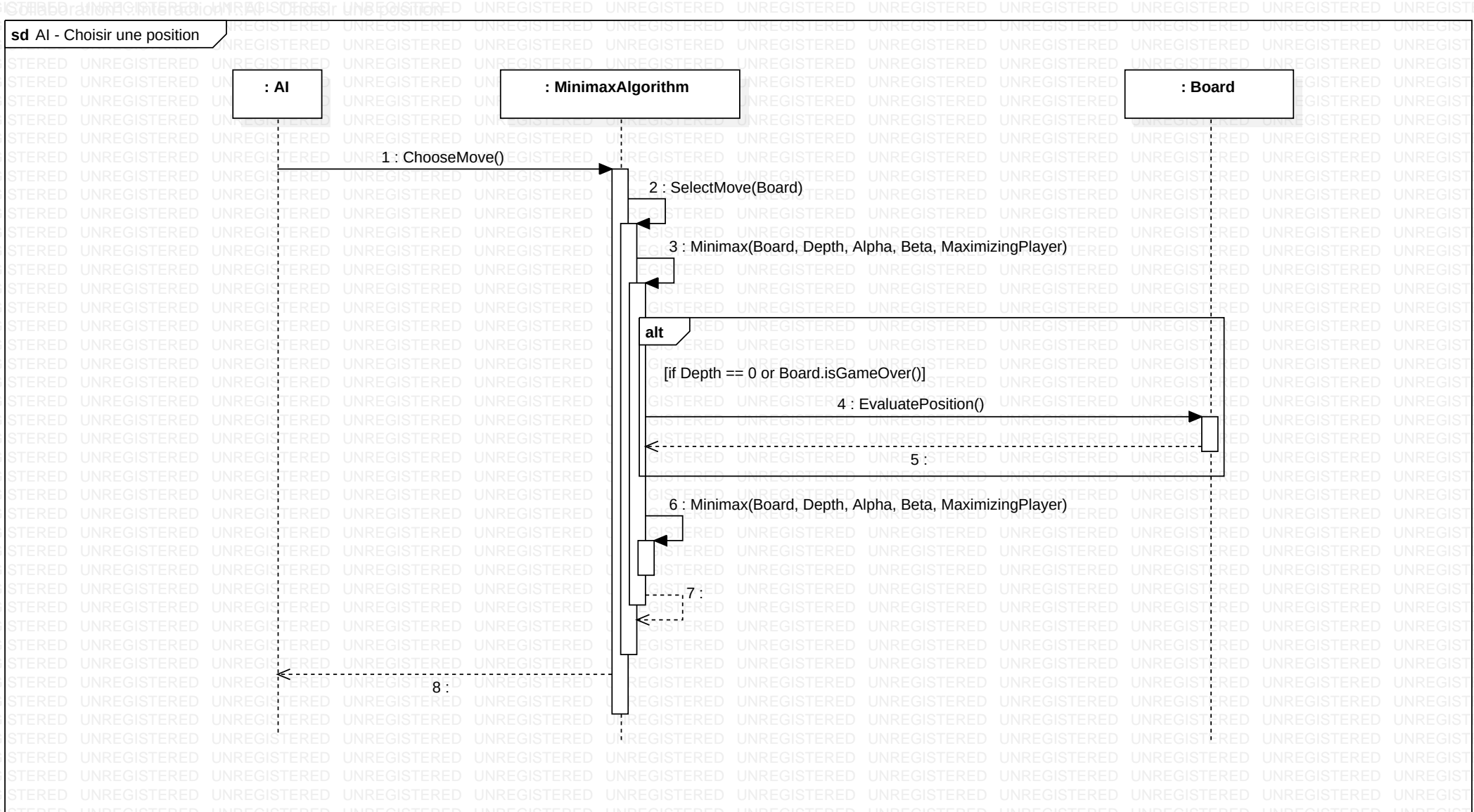
History

Mouse Scroll

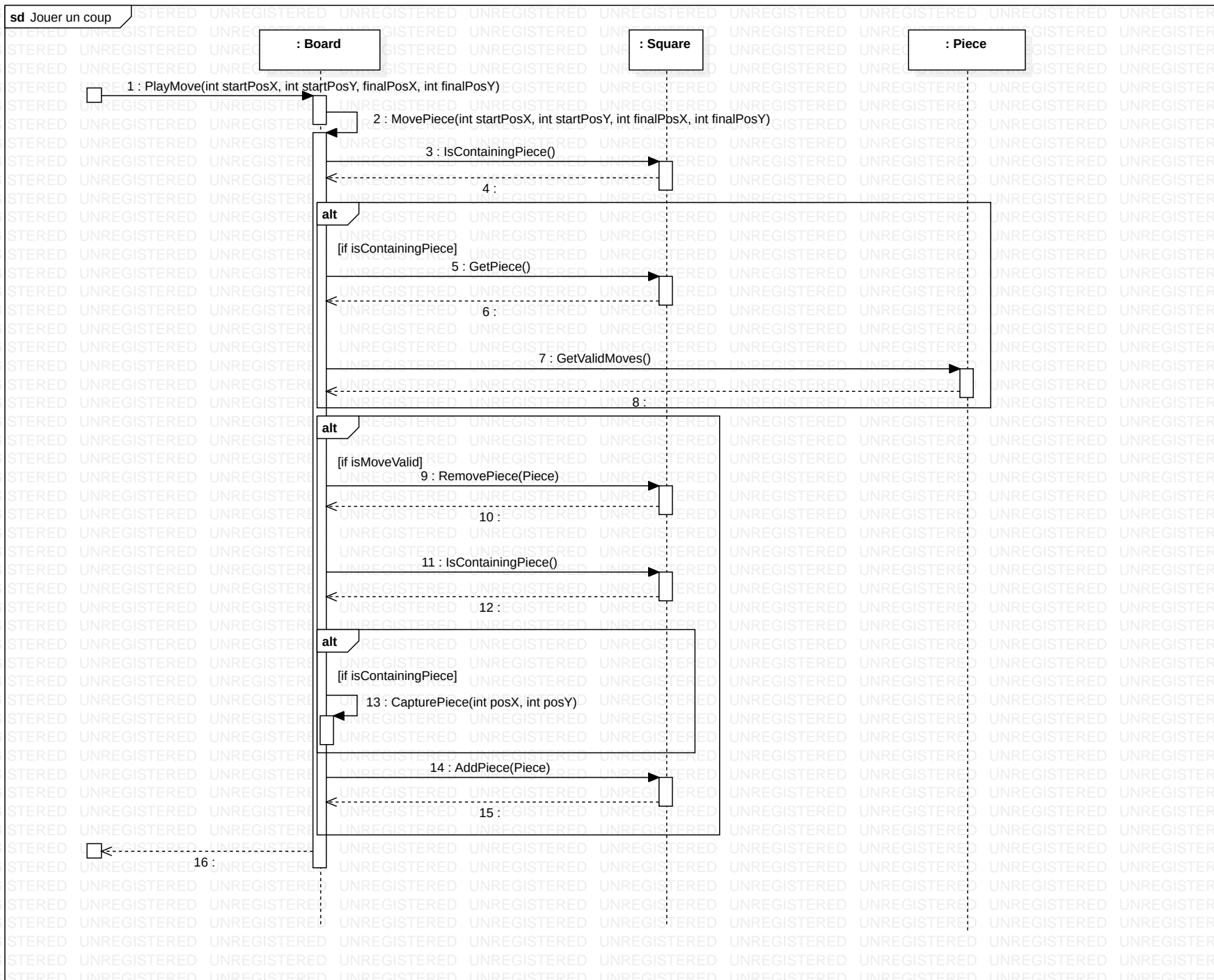


Player

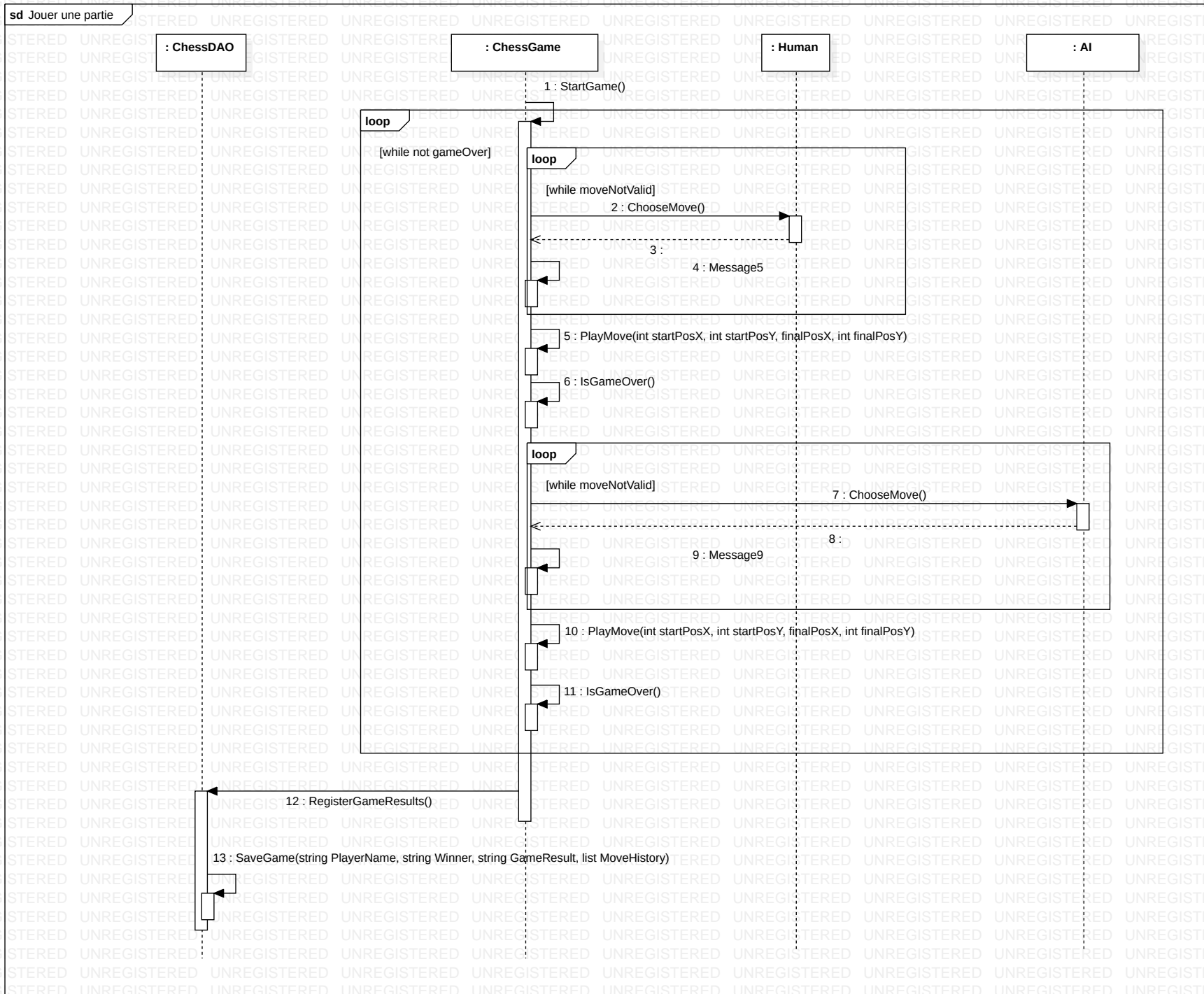
Click "Visualize Game" button



Collaboration: Interaction in a coup



Collaboration: Interaction between the player



Structure de données - JSON

“moves” est une liste de la longueur du nombre de coups joués en tout, contenant un mouvement pour joueur blanc et joueur noir

```
{
  "playerName" : "Nicolas",
  "date" : "23-02-2021",
  "winner" : "AI",
  "result" : "Checkmate",
  "numberOfMoves" : 35,
  "moves" : [
    [
      {
        "playerColor" : "White",
        "typeOfPiece" : "Pawn",
        "originalPosition" : "e4",
        "finalPosition" : "d5",
        "isCapturingPiece" : true,
        "isChecking" : false,
        "isCheckMating" : false,
        "isCastling" : false
      },
      {
        "playerColor" : "Black",
        "typeOfPiece" : "Pawn",
        "originalPosition" : "c7",
        "finalPosition" : "c5",
        "isCapturingPiece" : false,
        "isChecking" : false,
        "isCheckMating" : false,
        "isCastling" : false
      }
    ],
    [
      {
        "playerColor" : "White",
        "typeOfPiece" : "Bishop",
        "originalPosition" : "a4",
        "finalPosition" : "b5",
        "isCapturingPiece" : false,
        "isChecking" : true,
        "isCheckMating" : false,
        "isCastling" : false
      },
      {
        "playerColor" : "Black",
        "typeOfPiece" : "Knight",
        "originalPosition" : "g1",
        "finalPosition" : "f3",
        "isCapturingPiece" : false,
        "isChecking" : false,
        "isCheckMating" : false,
        "isCastling" : false
      }
    ]
  ]
}
```