

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane MIRA – Bejaia
Faculté des Sciences Exactes
Département d'Informatique



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

Mémoire de Fin de Cycle Master

En vue de l'obtention du Diplôme de Master en Informatique
Option : Génie Logiciel & Intelligence Artificielle

Thème

**Conception & Réalisation d'une Plateforme
Dédiés au Jeu d'Échecs**

Présenté par :

Katia KLAAI

Zeyneb ZERMANI

Soutenu le 14/09/2023 devant le jury composé de :

Présidente : Mme YAICI Malika

Examinatrice : Mme HAMZA Lamia

Encadrant : M. OUZEGGANE Redouane

Année Universitaire 2022-2023

REMERCIEMENTS

Au terme de ce travail, je tiens à exprimer ma profonde gratitude et mes sincères remerciements.

Avant tout propos, louange à Dieu tout puissant qui nous a donné la force, le courage, l'espoir nécessaire et la patience pour accomplir ce travail et surmonter l'ensemble des difficultés rencontrées.

En guise de reconnaissance, nous adressons nos sincères remerciements à notre encadreur M. OUZEGGANE Redouane pour son soutien, sa disponibilité, ses précieux conseils et son aide tout au long de l'élaboration de ce travail.

Nous remercions les membres de jury de leur attention et intérêt portés à notre travail et la correction de ce modeste mémoire.

Enfin, nous nous acquittons, volontiers d'un devoir de gratitude et de remerciements à nos chères familles et à tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce travail.

DÉDICACES

Je dédie ce modeste travail . . .

À mes chers parents

Sans les quel je ne serais pas là aujourd'hui, eux qui étaient derrière moi à chaque seconde de ma vie dans les bons et les mauvais moments.

À mes chers sœurs et frères.

À tous mes amis et collègues

Katia KLAAI

Je dédie cet événement marquant de ma vie :

À mes très chers parents, mon support dans ma vie qui ont toujours répondu présents dans les moments les plus difficiles, leur soutien et leurs encouragements m'ont toujours donné la force. Autant de phrases et d'expressions aussi éloquentes soit-elles ne sauraient exprimer ma reconnaissance.

À mes chers frères et sœurs et à toute ma famille

À mes chers enseignantes et enseignants qui ont contribué à ma formation.

À tous les collègues d'études

Zeyneb ZERMANI

TABLE DES MATIÈRES

Table des Matières.....	i
Liste des Figures.....	iv
Liste des Tableaux.....	v
Liste des Abréviations.....	vi
<i>Introduction Générale.....</i>	<i>1</i>
<i>Chapitre I : Introduction au jeu d'échecs.....</i>	<i>3</i>
I.1 – Introduction.....	3
– Définitions et notions de base.....	3
– C'est quoi le jeu d'échecs.....	3
– Échiquier.....	3
– Pièces et position initiale.....	4
– Valeurs des pièces.....	5
– Historique des échecs.....	7
– Règles de jeu d'échecs.....	8
– Règles de déplacement des pièces.....	8
a- Règles générales.....	8
b- Déplacement du roi.....	8
b- Déplacement de la dame.....	9
c- Déplacement de la tour.....	10
d- Déplacement du fou.....	10
e- Déplacement du cavalier.....	11
f- Déplacement du pion.....	12
h- Le roque.....	14
– La fin d'une partie.....	14
– Écriture des coups : Notation algébrique.....	15
– Notation algébrique complète.....	16

– Logiciels et échecs en ligne	18
– Conclusion	18
<i>Chapitre II : Moteurs des échecs</i>	19
– Introduction	19
– Concept de moteur d'échecs	19
– Protocole UCI.....	19
– Caractéristiques du protocole UCI	20
– Commandes principales de UCI	20
– Compétition entre les moteurs d'échecs.....	21
– Conclusion	21
<i>Chapitre III : Analyse & conception</i>	23
– Introduction	23
– Architecture de la solution.....	23
– Cas d'utilisation	25
– Identification des acteurs	25
– Identification des cas d'utilisation	25
– Diagramme de cas d'utilisation	26
– Diagrammes de séquence.....	26
– Diagramme de séquence : « Réaliser un coup »	28
– Diagramme de séquence : « Annuler un coup »	29
– Diagramme de séquence : « Lancer le moteur »	29
– Diagramme de classe	31
– Conclusion	31
<i>Chapitre IV : Réalisation & Tests</i>	33
– Introduction	33
– Technologies et langages utilisés.....	33
– Sublime Text	33
– IDE Eclipse.....	33
XAMPP	33

Table des matières

JavaScript.....	34
NodeJS	34
– Taux d’avancement	34
– Captures d’écran.....	34
– Conclusion.....	39
<i>Conclusion Générale</i>	<i>41</i>
Bibliographie	43
Annexe A : Histoire de jeu d’échecs.....	47
Annexe B : Programmes & applications sur les échecs.....	49
Annexe C : Protocole UCI.....	52
Annexe D : Diagrammes UML.....	55

LISTE DES FIGURES

Figure I-1: Échiquier, ses colonnes, traverses et diagonales	4
Figure I.2: Position initiale des pièces	6
Figure I.3: Zones de l'échiquier	7
Figure I.4: Déplacement du roi.....	9
Figure I.5: Mouvement d'une dame	9
Figure I.6: Marche de la tour.....	10
Figure I.7: Déplacement du fou	10
Figure I.8: Obstruction.....	11
Figure I.9: Déplacement du cavalier	11
Figure I.10: Marche des pions.....	12
Figure I.11: Pion capture de biais	13
Figure I.12: Promotion du pion	13
Figure I.13: Prise en passant.....	14
Figure I.14: Le premier coup des blancs : 1.Ng1-f3	17
Figure I.15: Commencer par le coup des noirs.....	17
Figure III.1: Architecture générale de la solution.....	24
Figure III.2: Diagramme des cas d'utilisation.....	27
Figure III.3: Diagramme de séquence « Réaliser un coup »	28
Figure III.4: Diagramme de séquence «Annuler un coup »	29
Figure III.5: Diagramme de séquence «Lancer le moteur ».....	30
Figure III.6: Diagramme de classe	31
Figure IV.1: Interface d'accueil de l'application	35
Figure IV.2: Bases de partie ouverte.....	36
Figure IV.3: Configuration et Personnalisation de l'échiquier	37
Figure IV.4: Apprendre les codes des cases de l'échiquier.....	38
Figure IV.5: Jouer contre la machine	38

LISTE DES TABLEAUX

<i>Tableau I.1:</i> Codes et figures des pièces du jeu d'échecs.....	5
<i>Tableau I.2:</i> Valeurs théorique des pièces.....	6
<i>Tableau III.1 3:</i> Identification des cas d'utilisation.....	25

LISTE DES ABRÉVIATIONS

API	: Application Programming Interface
GUI	: Graphical User Interface.
IBM	: International Business Machines.
ICC	: Internet Chess Club.
IDE	: Integrated Development Environment.
IGM	: International Grand Master.
IM	: International Master.
JS	: JavaScript.
PHP	: Hypertext Preprocessor.
REST	: Representational State Transfer.
TCEC	: Top Chess Engine Championship.
UCI	: Universal Chess Interface.
UML	: Unified Markup Language.
UP	: Unified Process.
URL	: Unified Resource Locator.
SCID	: Shane's Chess Information Database
WCCC	: World Computer Chess Championship.
WCSCC	: World Computer Speed Chess Championship.
XAMPP	: Cross Apache Mysql/MariaDB PHP Perl.
XP	: Extreme Programming.

INTRODUCTION GÉNÉRALE

INTRODUCTION GÉNÉRALE

L'informatique, comme science rationnelle de traitement automatique de l'information, et aussi comme technique et outil de travail et de loisir, est disponible dans tous les aspect de la vie humaine : éducation, travail, économie, communication, tourisme voir même les jeux et loisir.

Le domaine de jeu d'échecs ne fait pas exception à cette réalité, qui est devenue comme une règle, et beaucoup d'avancé et de progrès en terme de logiciels et site web pour jouer et apprendre le noble jeu (échecs), et aussi l'implantation d'une intelligence artificielle qui a surpasser la puissance de calcul humain, voir même des grands maîtres.

Dans ce contexte, notre modeste travail se focalise sur la conception et la réalisation d'une plateforme, extensible et ouverte en terme de technologies, dédiée aux échecs : analyser des partie, jouer avec l'ordinateur, joueur à distance avec une autre personnes, soit en réseau local ou via l'internet. Cette plateforme contiendra aussi des leçons et des exercices pour apprentissage.

Cette plateforme sera constitué de plusieurs modules, principalement un serveur et une application bureau, avec une ouverture vers d'autre type d'application (Web et mobile).

Pour bien présenter notre modeste travail, ce mémoire est organisé autour de quatre à chapitres, allant de la présentation du jeu ancestral des échecs, passant par les moteurs d'échecs, qui représentent l'intelligence artificiel pour analyser et déterminer les meilleurs coups, par la suite la phase d'identification et analyse des besoins, et la conception de notre application. Enfin, notre mémoire se termine par la phase de réalisation de l'application.

CHAPITRE I
INTRODUCTION AUX
ÉCHECS

CHAPITRE I

Introduction au jeu d'échecs

I.1 – Introduction

Dans ce premier chapitre, nous allons voir une introduction aux échecs (Chess en anglais), en passant par sa définition, son histoire, le champs de bataille, à savoir l'échiquier. Nous verrons aussi les règles de jeu qui régissent une partie d'échecs, les différents types de compétition et titres internationaux, et comment les joueurs sont classés. Nous terminons ce chapitre par la notation algébrique (avec ses différentes variantes) qui permet d'écrire et de transcrire les coups d'une partie de jeu d'échecs.

– Définitions et notions de base

– C'est quoi le jeu d'échecs

Au cours de notre travail, nous avons trouvé plusieurs définitions du jeu d'échecs, dans divers articles en lignes et livres. Dans la référence [01], le jeu d'échecs est défini comme un « Jeu dans lequel deux joueurs font manœuvrer l'une contre l'autre, sur un plateau divisé en 64 cases, deux séries de 16 pièces de valeurs diverses ». Dans la même référence, une définition complémentaire est donnée : « Jeu de stratégie et de combinaison, où le hasard n'a aucune part, les échecs sont tout à la fois un divertissement, un art et un sport cérébral » [01]. Chaque déplacement d'une pièce est dit : coup (move en anglais), et les deux joueurs déplacent leurs pièces alternativement et obligatoirement (un joueur ne peut pas dire : *je passe*).

– Échiquier

Le plateau du jeu d'échecs, dit échiquier (voir la *figure 1.1*), représente une grille carrée de 8 cases de chaque côté, avec des couleurs alternativement blanches et noires (ou, d'une façon générale, claires et sombres). Une suite

verticale de 8 cases s'appelle colonne ; une suite horizontale de 8 cases s'appelle rangée (ou traverse). Les colonnes sont indiquées par une lettre alphabétique allant de A à H, et les rangées sont numérotées de 1 à 8. On y distingue aussi les diagonales, composées de plusieurs cases de même couleur variant de 2 à 8 colonnes. Colonnes, rangées et diagonales sont les lignes d'action de l'échiquier.

Chaque case de l'échiquier est référencée par la lettre de sa colonne et le numéro de sa traverse. Ceci est dit « *système de coordonnées* ».

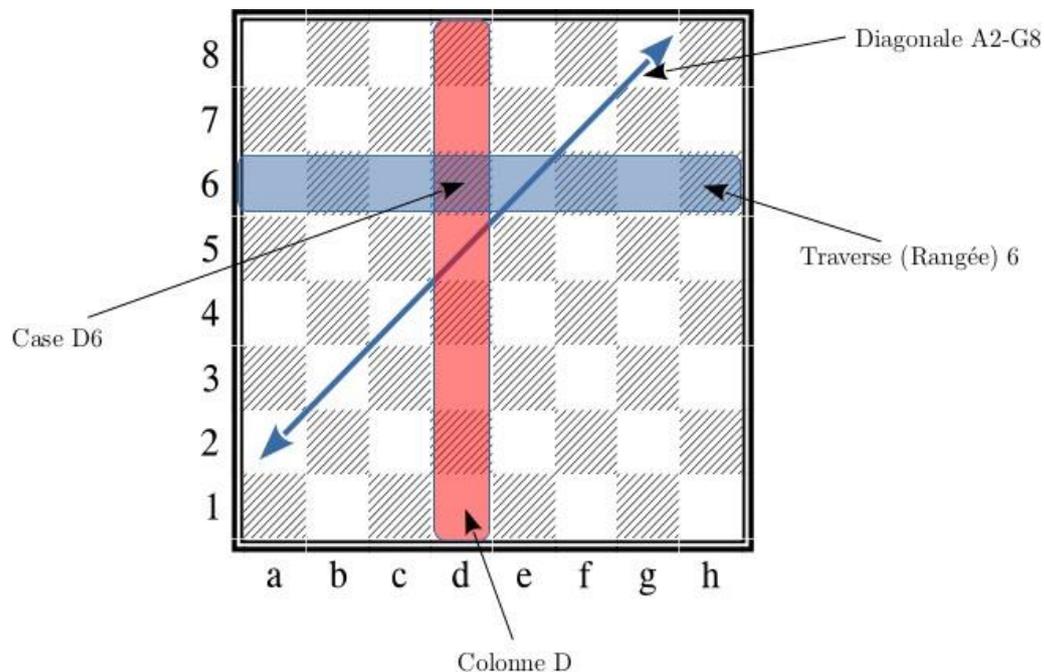


Figure I-1: Échiquier, ses colonnes, traverses et diagonales

L'échiquier peut être divisé en zones, selon la position initiale des pièces, voir la figure I.3

– Pièces et position initiale

Comme nous l'avons mentionné précédemment, chaque joueur dispose, au début de la partie, de seize pièces : soit huit pions et 8 figures qui sont : un roi, une dame (ou reine), deux tours, deux cavaliers et deux fous. Ces pièces sont disposées, initialement (au début de la partie), dans un ordre donné, et chacune d'elles possède une forme et une valeur différentes. La figure I.2 illustre la position initiale des pièces avant de commencer toute partie d'échec.

À noter que l'échiquier est placé de telle sorte d'avoir une case blanche à droite de chaque joueur (on dit en anglais : *White in right*), ces cases sont **h1** pour les blancs et **a8** pour les noirs.

Chaque pièce possède un code (lettre) (Pour notre cas, nous allons adopter la notation anglaise) et représentation figurine. Pour la réalisation des diagrammes d'échiquier et ses pièces, nous avons utilisé une police dédiée : Chess Mérida (voir [05]) . Le *tableau I.1*, illustre les pièces, leur code ainsi que leur figurine :

Pièce	Code		Figure
	En français	En anglais	
Roi / King	R	K	 
Dame (Reine) / Queen	D	Q	 
Tour / Rook	T	R	 
Fou / Bishop	F	B	 
Cavalier / Knight	C	N	 
Pion / Pawn	P	P	 

Tableau I.1: Codes et figures des pièces du jeu d'échecs

– Valeurs des pièces

Comme indiqué dans les références [01] et [02], chaque pièce possède une *valeur absolue (théorique)* et une *valeur réelle (positionnelle)* qui dépend de la position actuelle sur l'échiquier. La valeur théorique est déterminée par le nombre de cases contrôlées par une pièce (le nombre de cases vers lesquelles elle peut se déplacer). Ceci dépend des règles de déplacements et de capture de chaque pièce. Le *tableau I.2* montre les valeurs (théoriques) des pièces avec le nombre de cases contrôlées par celles-ci.

Chapitre I : Introduction aux Échecs

Pièce	Nombre de cases contrôlés		Valeur théorique
	Pièce installée sur une case d'angle	Pièce installée au milieu de l'échiquier	
Dame	21	27	9
Tour	14	14	5
Fou	7	13	3
Cavalier	2	8	3
Pion	1	2	1

Tableau I.2: Valeurs théorique des pièces

Il faut noter que le « Roi » est une pièce spéciale, qui a une valeur inestimable, puisque perdre son roi est synonyme à la perte de la partie.

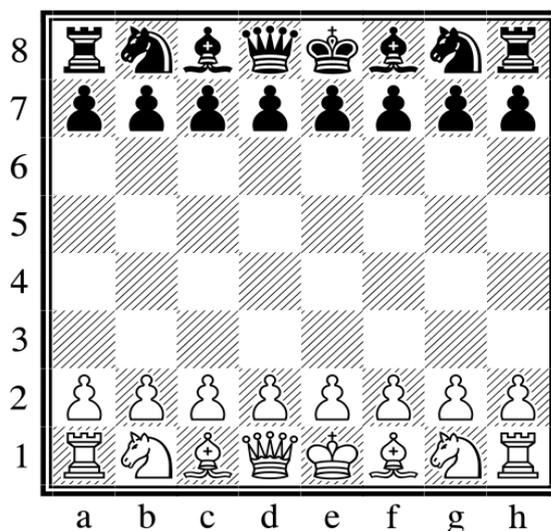


Figure I.2: Position initiale des pièces.

À base de cette disposition initiale des pièces, nous pouvons diviser l'échiquier en zones ou secteurs, comme suit :

- Territoire des blancs et territoire des noirs.
- Aile dame et Aile roi.
- Bandes (colonnes A et B, et rangées 1 et 2).
- Centre restreint, centre large et les quatre coins.

Voir la *figure I.3* pour les différentes zones de l'échiquier.

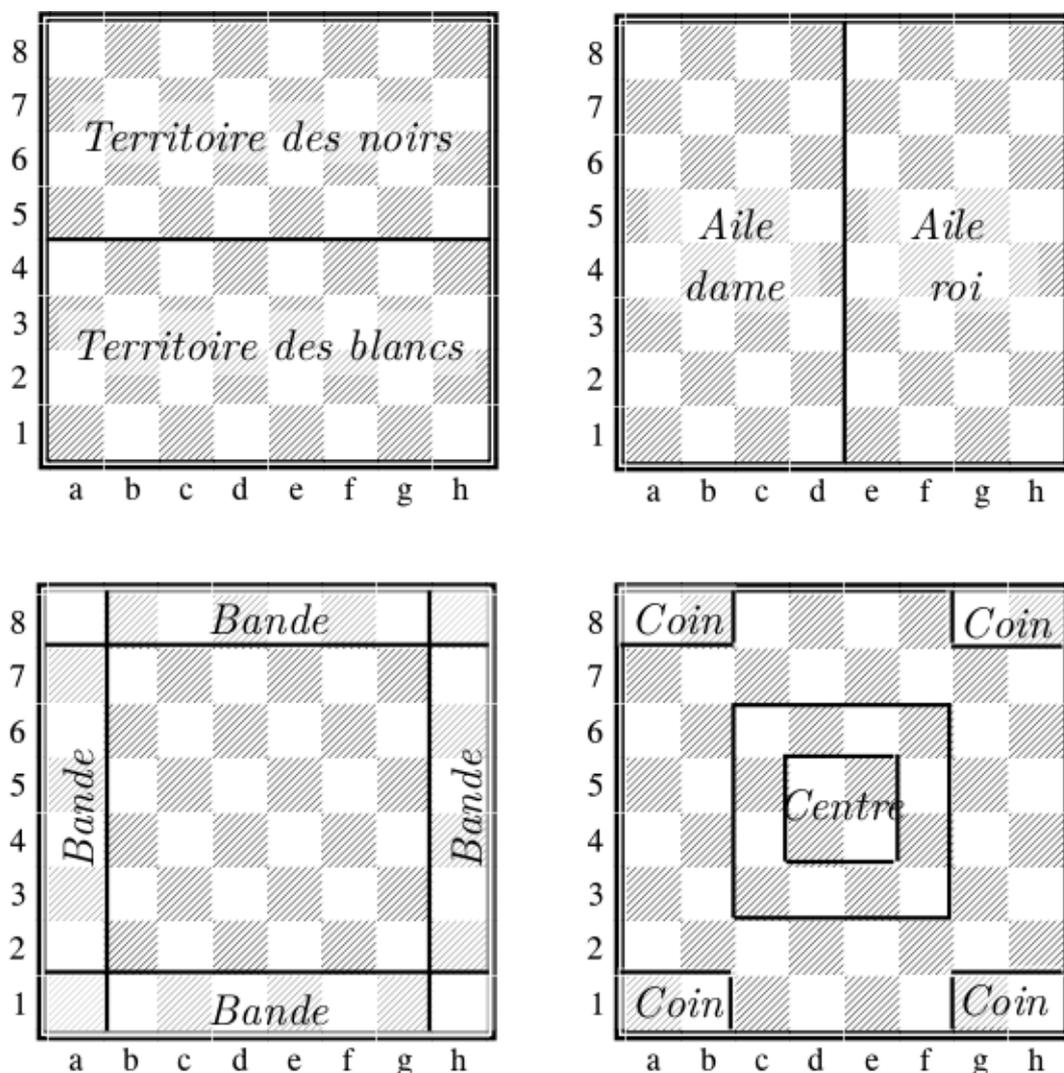


Figure I.3: Zones de l'échiquier

– Historique des échecs

Il est difficile de dater précisément l'apparition du jeu d'échecs, toute fois toute les thèses s'accorde pour situer son origine géographique en Inde et plus précisément au Cachemire [03]. Plusieurs légendes et hypothèses relatent l'origine de ce jeu ancestral. Le jeux, dans sa forme primitives est né entre III^e et VI^e siècle.

Le jeu d'échecs moderne diffère des versions anciennes, comme les échecs persans (chantrang), indiens(chaturangua), arabes (shatranj), mongole (shatar), européens, birmans, ...

Pour bien se documenter sur l'histoire des échecs, nous orientons le lecteur vers le livre [03].

– Règles de jeu d'échecs

Comme nous l'avons précisé précédemment, une partie de jeu d'échecs se joue entre deux joueurs qui déplacent deux séries de pièces, seize pièces pour chacune d'elles. Dans ce qui suit, nous présenterons les règles de déplacement des différentes pièces et pions, l'objectif du jeu d'échecs et les différents résultats d'une partie d'échecs.

– Règles de déplacement des pièces

a- Règles générales

Le mouvement d'une pièce se définit par son déplacement de sa case de départ et son positionnement sur une nouvelle case d'arrivée. Sachant que deux pièces ne peuvent jamais occuper une même case en même temps. Les blancs commencent toujours une partie de jeu d'échecs, en partant de la position initiale (voir la *figure 1.2*).

Une pièce ne peut jamais se déplacer vers une case occupée par une autre pièce de la même couleur (de son camp), cependant, si la case d'arrivée est occupée par une pièce adverse, cette dernière est *capturée* et doit être retirée de l'échiquier pour le reste de la partie. Bien évidemment, nous ne pouvons jamais capturer ses propres pièces, et il n'y a pas d'obligation de capturer les pièces de l'adversaire.

Toutes les pièces ont le droit de reculer, sauf les pions qui se déplacent uniquement vers l'avant. Un pion qui peut atteindre sa dernière rangée (traverse) est *promu* à l'une des pièces suivantes : dame, tour, fou ou cavalier.

b- Déplacement du roi

Le roi se déplace dans n'importe quelle direction d'une seule case à la fois. Cependant, le roi ne peut pas se déplacer à une case adjacente contrôlée (attaquée) par une pièce adverse.

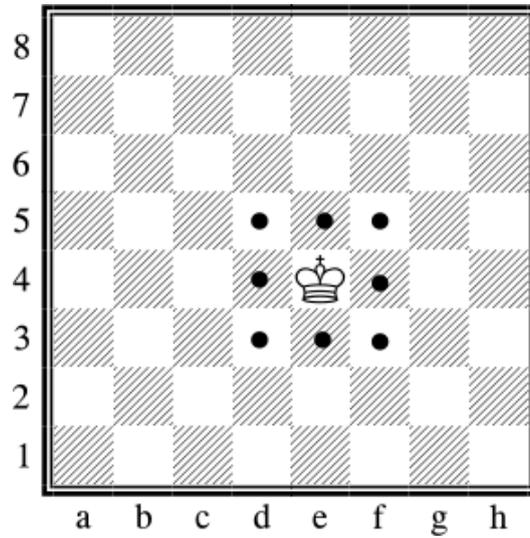


Figure I.4: Déplacement du roi

Dans la figure ci-dessus, le roi placé dans la case **e4**, peut se déplacer vers les cases : **d3, e3, f3, d4, f4, d5, e5** et **f5** (8 cases possibles)

b- Déplacement de la dame

La figure suivante, illustre la marche de la dame, qui représente une pièce de ligne à longue portée :

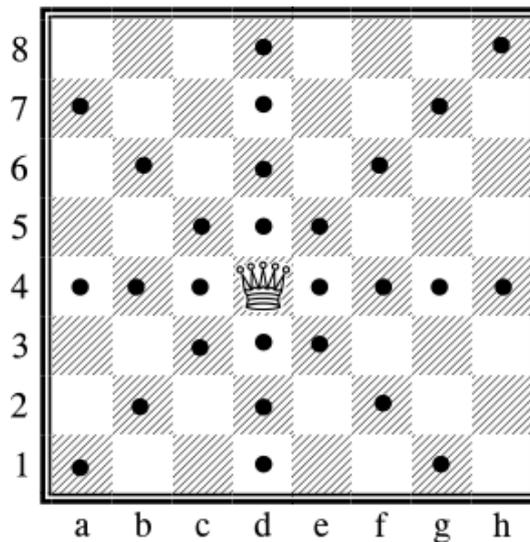


Figure I.5: Mouvement d'une dame

La dame se déplace dans tous les sens, horizontalement et verticalement.

c- Déplacement de la tour

De la même façon que la dame, une tour est une pièce de ligne qui se déplace uniquement horizontalement ou verticalement. Voir la figure suivante :

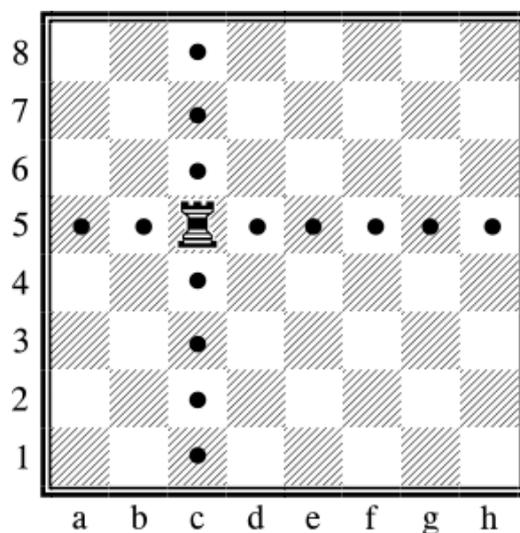


Figure I.6: Marche de la tour

d- Déplacement du fou

Le fou se déplace sur les diagonales autant de cases possibles.

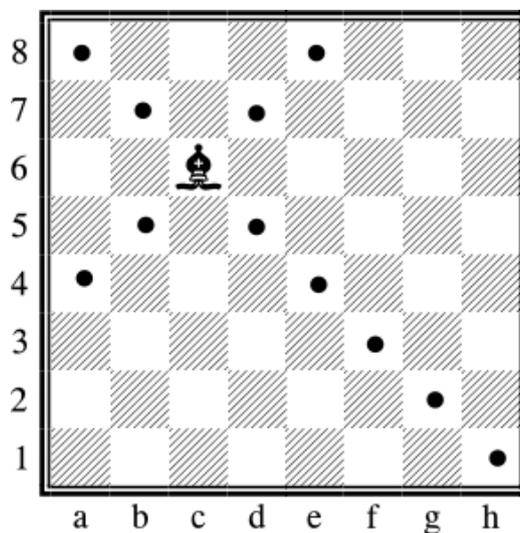


Figure I.7: Déplacement du fou

Remarque

Les pièces de ligne sont les pièces à longue portée qui se déplacent soit horizontalement et verticalement (colonnes et traverses) soit en diagonale, à

condition, que les cases soient contiguës, comme indiqué par la figure ci-dessous :

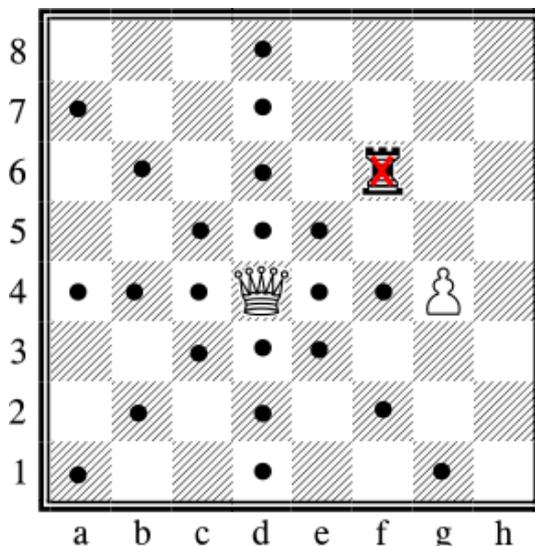


Figure I.8: Obstruction

Dans la figure ci-dessus, la dame ne peut se déplacer vers les cases $g4$, $h4$, $g7$ et $h8$. Par contre, la dame blanche peut occuper la case $f6$ en retirant la tour noire de l'échiquier : ce que nous appelons une *capture* ou *prise*.

e- Déplacement du cavalier

La cavalier se déplace d'une seule case horizontale ou verticale, (comme une tour mais d'une seule case), puis d'une seule case en diagonal (comme un fous mais d'une seule case). On peut dire aussi, que le cavalier se déplace selon la forme de la lettre « L », dans tous les sens). La marche du cavalier est indiqué par la figure suivante :

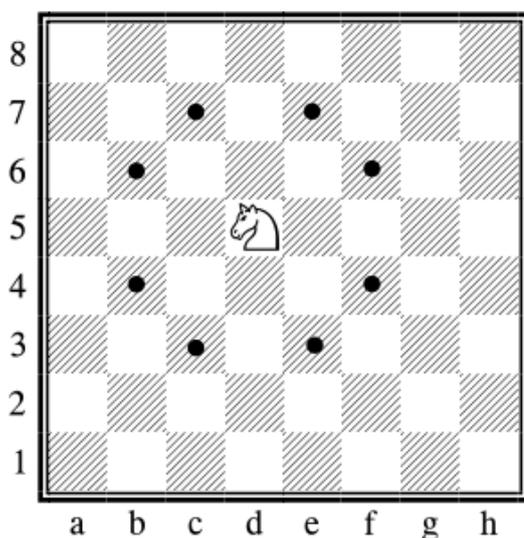


Figure I.9: Déplacement du cavalier

Remarques

- Les cases vers lesquelles une pièce peu se déplacer sont dites les cases contrôlée par cette pièce.

- Le roi, la dame, tour, fou ou bien cavalier peut capturer une pièce adverse qui se trouver dans l'une des cases contrôlée par la première pièce.

f- Déplacement du pion

Malgré que le pion est la plus faible pièce sur l'échiquier, son déplacement est régie par un plus grand nombre de règles que les autres pièces. Ces règles sont comme suit :

- **Règle N° 01** : Un pion se déplace toujours devant lui, sa sa colonne, sans jamais pouvoir reculer ni franchir une case occupée. Dans ce cas, on peut avoir un pion bloqué.

- **Règle N° 02** : Un pion avance d'une seul case à la fois. Cependant, lorsqu'il est sur sa case initiale (la rangée deux pour les blancs ou la septième rangée pour les noirs), il peut avancer d'une ou de deux cases (voir la *figure I.9*).

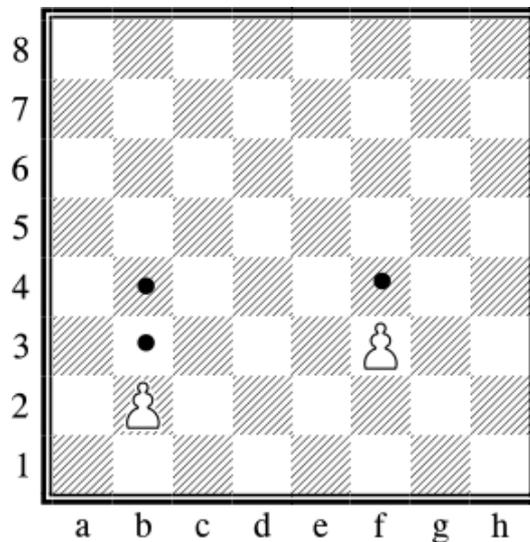


Figure I.10: Marche des pions

- **Règle N° 03** : Le pion est la seule pièce qui ne capture pas comme elle se déplace.

Un pion peut capturer une pièce adverse qui se trouve sur une (seule) case située devant lui et de même couleur que celle de ce pion, comme suit :

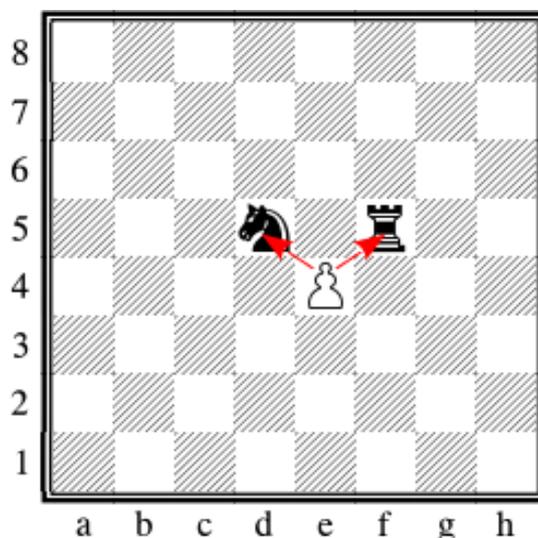


Figure I.11: Pion capture de biais

- **Règle N° 04** : Un pion qui peut atteindre sa dernière rangée (il arrive au bout de sa course), doit se transformer. Il peut devenir une dame, une tour, un fou ou un cavalier (mais jamais un roi). Cette transformation, qui concerne uniquement les pions, est dite « Promotion », et elle est obligatoire, immédiate et irréversible.

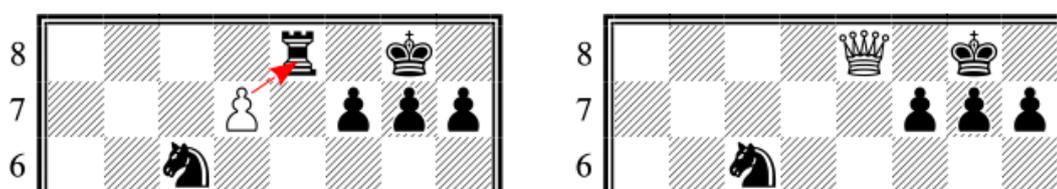


Figure I.12: Promotion du pion

- **Règle N° 05** : Lorsqu'un pion adverse avance de deux cases (ceci est possible uniquement s'il est à sa case initiale (voir la règle N°02)) et vient se loger à côté d'un pion adverse, ce dernier peut, au coup suivant, capturer l'autre pion comme s'il n'avait avancé que d'une seule case.

Cette forme de capture ne concerne que les pions et se nomme prise « en passant », et elle doit se faire tout de suite après le mouvement du pion adverse, sans pouvoir être reportée au trait suivant. La *figure I.12* illustre la capture *en Passant* :

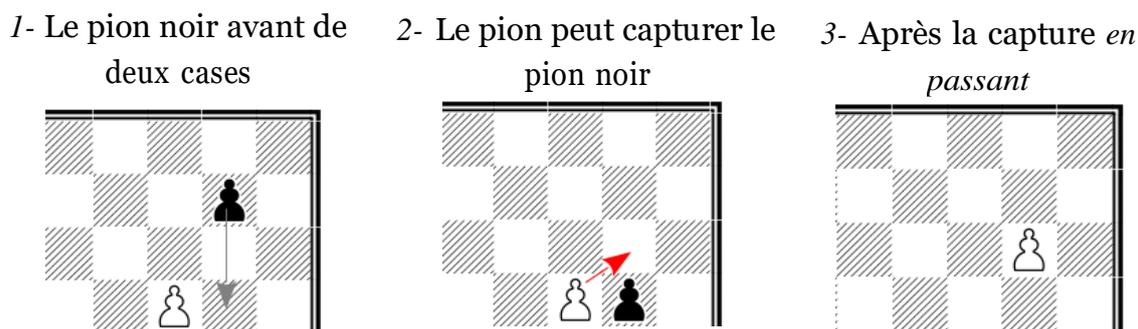


Figure I.13: Prise *en passant*

h- Le roque

Le roque implique le mouvement du roi et d'une tour. C'est le seul coup réglementaire qui suppose le déplacement de deux pièces. Il vise à mettre le roi en sécurité et à centraliser une tour. Il y a deux type de roque : « petit roque » (roque coté aile roi), et « grand roque » (roque coté aile dame).

Le « petit roque » et le « grand roque » sont nommés ainsi en raison du nombre de cases vides entre le roi et la tour avant l'exécution du coup. La *figure I.13*

– La fin d'une partie

Une partie d'échecs se termine soit par la victoire d'un joueur sur l'autre, soit par la nulle.

Aucun règlement n'y oblige, mais l'usage veut qu'un joueur qui met le roi adverse en prise l'annonce en disant « échec ».

Le joueur dont le roi est en échec est obligé de l'y soustraire avant de faire quoi que ce soit d'autre.

Il n'existe que trois manières de soustraire un roi à un échec :

- capturer la pièce qui attaque le roi ;
- déplacer le roi sur une case où il n'est plus en prise ;
- interposer une pièce entre l'attaquant et le roi.

Lorsqu'un roi est en échec et qu'aucun coup réglementaire ne permet de l'y soustraire, il est « mat » et la partie est terminée.

Un échec se note avec le signe plus « + », et un mat, avec le signe dièse « # ».

*Pour les cas de la nullité, il y a six cas :

1 - Pat :

Le joueur dont le tour est arrivé n'a pas de coup autorisé, mais son Roi n'est pas en échec.

2-Matériel insuffisant:

Aucun des deux joueurs ne dispose de suffisamment de pièces pour faire échec et mat

Exemples : Roi contre Roi.

3-Trois fois la même position :

Une position identique (toutes les pièces dans les mêmes cases) se retrouve trois fois au cours de la partie.

4- Échec perpétuel :

Un joueur continue de déjouer, le Roi et l'adversaire ne parvient pas à trouver une solution satisfaisante pour l'arrêter.

5- La règle des 50 coups :

Au cours de 50 coups (50 coups blancs + 50 coups noirs), il n'y a pas de prise ni de mouvement de pion. Il est rare et ne se produit que dans les régions très longues.

6-le consentement mutuel des joueurs :

Une fois son coup joué, un joueur a la possibilité de proposer nulle. Son adversaire a le pouvoir d'accepter ou de refuser. Si elle accepte, la partie prend fin.

– Écriture des coups : Notation algébrique

Pour écrire les coups joués dans une partie d'échecs, la notation algébrique est utilisée, ceci permet la reproduire de cette partie. Plusieurs

Les systèmes de notation ont été proposés, mais c'est le système de notation « algébrique » qui est le plus utilisé.

Ce système existe en deux versions (ou variantes) : version « complète » et en une version dite « abrégée ».

– Notation algébrique complète

Nous présentons tout d'abord la notation complète, qui suit les étapes d'écriture suivantes :

- 1- le numéro du coup ;
- 2- un point, s'il s'agit d'un coup des blancs, trois points dans le cas d'un coup des noirs ;
- 3- l'initiale de la pièce jouée (voir le *tableau I.1*), sauf pour le pion, on laisse vide (pour la suite, l'initiale de la pièce sera en anglais);
- 4- un trait d'union « - » s'il s'agit d'un déplacement ou un « x » s'il s'agit d'une prise (capture) ;
- 5- les coordonnées de la case d'arrivée ;
- 6- les autres signes de notation, comme échec « + », mat « # », et les annotations et commentaires (?, ??, !, !!, ...)

La *figure I.13* représente, au début de la partie, le coup **1. Ng1-f3**, qui signifie le premier coup des blancs, et qui s'agit du déplacement du cavalier (*kNight* en anglais) de la case **g1** vers la case **f3** :

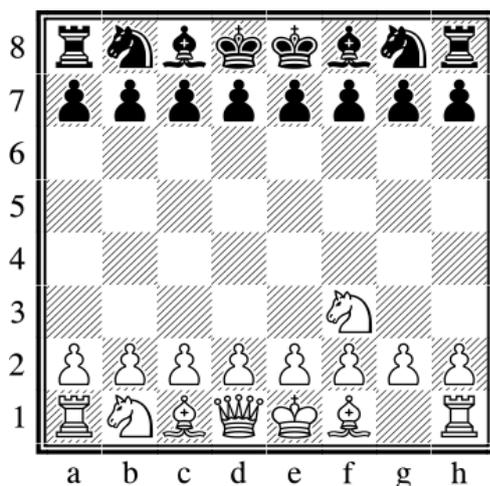


Figure I.14: Le premier coup des blancs : 1.Ng1-f3

La figure I.14, illustre comment écrire le coup des noirs, avec la notation algébrique complète.

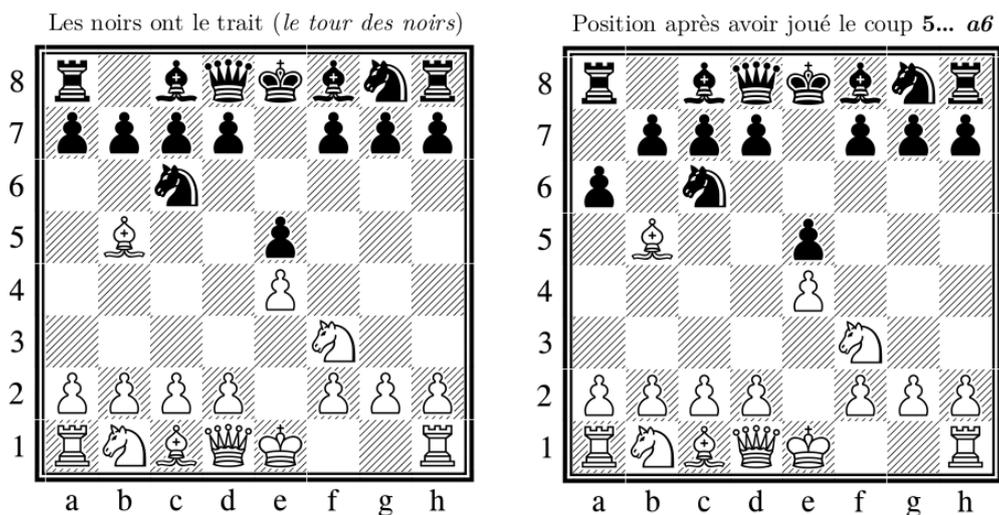


Figure I.15: Commencer par le coups des noirs

– Logiciels et échecs en ligne

Dans le domaine des logiciels des jeux d'échecs, il y a une séparation entre l'interface qui représente l'échiquier et les différentes options du logiciel, d'un côté, et le programme responsable de raisonner sur des positions et de jouer automatiquement, d'un autre côté. Ces derniers programmes sont dits moteurs d'échecs (en anglais : chess engine). Dans ce sujet, nous présentons uniquement les logiciels d'interface, et la partie intelligente (moteur d'échecs) fera l'objet du deuxième chapitre.

Plusieurs logiciels de jeux d'échecs existent sur le marché, nous pouvons citer, dans une liste non exhaustive, les suivants :

- ChessX – Version 1.5.8 de février 2023, disponible pour Windows, Linux et Mac OS X
- Lucas Chess – Version R 2.02a de juillet 2022, disponible sous Windows
- SCID – Version de juin 2022, disponible sous Windows, Linux et Mac OS X

En ce qui concerne les sites web et plateformes de jeux en ligne, nous pouvons citer :

- Lichess (<https://lichess.org/>) : plateforme open source
- Chess.com (<https://www.chess.com/>)
- Chess24 (<https://chess24.com/en>)
- Chessable (<https://www.chessable.com/>) pour l'apprentissage des ouvertures.

Pour notre travail, nous sommes très reconnaissants du site Lichess, pour élaborer la section d'apprentissage et d'exercices.

– Conclusion

Dans ce premier chapitre, nous avons vu une introduction sur le jeu d'échecs, son historique, les règles de jeu et quelques logiciels et sites web pour jouer à ce noble jeu. Le chapitre suivant sera dédié à la partie intelligente à savoir les moteurs de jeu d'échecs.

CHAPITRE II

MOTEURS DES ÉCHECS

CHAPITRE II

Moteurs des Échecs

– Introduction

Dans ce deuxième chapitre, nous allons présenter la partie d'intelligence artificielle des échecs, qui connue sous terme anglais : *Chess Engine*, traduit en français en moteur d'échecs. Ceci a permet de séparer entre l'aspect visuel du jeu qui s'agit principalement d'un échiquier de la partie de raisonnement automatique exécuté par un ordinateur. Pour assurer le lien entre les deux composantes, le protocole UCI est le plus utilisé pour assurer cette communication.

– Concept de moteur d'échecs

Un moteur d'échecs est un programme, implémentant un algorithme, qui permet d'analyser des positions d'échecs et de donnée le coup et la suite des coups qu'il juge les plus puissants.

Ces moteurs différent entre eux, en puissance de jeu, dans l'algorithme d'évaluation et aussi dans le nom. Dans la littérature, il y a plusieurs moteurs d'échecs (commerciaux, open source ou libre), nous pouvons citer :

- Moteurs open source : Stockfish, Berserk, Leela Chess Zero, Komodo, ...
- Moteurs commerciaux : Shredder, Fritz, Rybka, Houdini, HIARCS

Pour plus de détail, vous pouvez consulter la référence [06].

– Protocole UCI

Comme nous l'avons vu précédemment, il faut distinguer entre l'interface graphique et le moteur d'échecs. Cette interface graphique permet gérer l'environnement visible par l'utilisateur : échiquier, paramétrage, résultat d'évaluation d'une position, ... Cette séparation, permet aux développeurs de

se concentrer sur les performance des moteurs d'échecs sans devoir gérer la représentation visuelle de l'échiquier.

Pour assurer le lien entre les interfaces graphique (GUI : Graphical User Interface) et le moteurs d'échecs, des protocoles dédiés sont utilisés. D'après notre recherche, il y a trois type de protocole :

- Protocole UCI (Universal Chess Interface) : c'est le protocole le plus répandu. [07] (Voir l'annexe C pour plus de détail).
- Protocole Winboard [08]
- Protocoles commerciaux : Fritz, Junior, Shredder, Hiarcs, ...

En ce qui concerne notre travail, nous avons utilisé le protocole UCI afin de communiquer avec le moteur d'échecs *stockfish*.

– Caractéristiques du protocole UCI

Voici quelques caractéristiques du protocole UCI :

1. **Standardisation:** UCI fournit une série de commandes standard que les moteurs d'échecs et les interfaces peuvent utiliser pour communiquer. Cela permet aux moteurs d'être interchangeables et de fonctionner avec différentes interfaces sans modification majeure.
2. **Simplicité:** Les commandes UCI sont simples et faciles à comprendre, ce qui facilite leur implémentation par les développeurs de moteurs et d'interfaces.
3. **Évolutivité:** UCI permet d'ajouter de nouvelles fonctionnalités au moteur ou à l'interface sans casser la compatibilité avec les versions précédentes.

– Commandes principales de UCI

Dans cette section, nous présentons quelques commandes du protocole UCI, pour plus d'information, voir l'annexe C.

- **uci** : Le moteur répond à cette commande en envoyant des informations sur lui-même, y compris son nom et sa version.

- **isready** : Commande envoyée par l'interface pour vérifier si le moteur est prêt à recevoir des commandes.
- **position** : Utilisée pour définir la position actuelle sur l'échiquier, soit en utilisant la notation FEN (Forsyth-Edwards Notation), soit en indiquant les coups joués depuis la position de départ.
- **go** : Lance le moteur pour calculer le meilleur coup à partir de la position actuelle. Cette commande peut inclure des paramètres tels que la profondeur de recherche ou le temps alloué pour le calcul.
- **stop** : Arrête le calcul du moteur et retourne le meilleur coup trouvé jusqu'à présent.
- **quit** : Termine l'exécution du moteur.

– Compétition entre les moteurs d'échecs

À fin de comparer et de classer la puissance des moteurs d'échecs, des compétitions entre ces derniers sont organisées. Historiquement parlant, les moteurs commerciaux ont été considérés comme les plus puissants (si un moteur libre et amateur est vainqueur dans une compétition (par exemple ZAPPA in 2005), il est rapidement commercialisé.

Cependant, et depuis 2010, des moteurs libres et open source arrivent à sur-dépasser la puissance des moteurs commerciaux.

Voici quelques compétitions et tournois entre les programmes d'échecs :

- Top Chess Engine Championship (TCEC)
- World Computer Chess Championship (WCCC)
- World Computer Speed Chess Championship (WCSCC)

– Conclusion

Le moteur d'échecs incarne l'intelligence artificielle qui permet d'analyser une position de pièces sur un échiquier et de sélectionner une liste de meilleurs coups à réaliser. Le prochain chapitre sera dédié à la conception de notre plateforme et application pour étudier, analyser des parties et jouer aux échecs.

CHAPITRE III
ANALYSE & CONCEPTION

CHAPITRE III

Analyse & Conception

– Introduction

Dans ce chapitre, nous allons encadrer les périmètres de notre travail, en proposant une architecture à implémenter, et schématisant les besoins fonctionnels de l'application à réaliser. Par la suite, nous passons à la conception, en utilisant le diagramme de classe et le schéma relationnel.

Les étapes et leur enchaînement sont pris d'une adaptation de la méthode UP (Unified Process), afin d'alléger les étapes et d'utiliser un sous-ensemble minimal des diagrammes UML (Unified Modeling Language). Cette méthode décrite dans [09], situé à mi-chemin entre UP et XP (eXtrem Programming), possède les caractéristiques suivantes :

- Conduite par les cas d'utilisation, comme UP
- Relativement légère et restreinte, comme XP
- Fondé sur l'utilisation de diagrammes UML nécessaires et suffisants (Modéliser 80% des problème en utilisant 20% d'UML).

– Architecture de la solution

Pour bien illustrer l'idée principale de notre travail, et l'objectif que nous voulons atteindre, la *figure III.1* montre l'architecture que nous voulons implémenter.

Il faut noter l'aspect extensible de cette architecture, dans le sens où d'autres types d'application (Mobile et Web) peuvent être greffé en consommant l'API offerte par le serveur RESTFUL API.

Cette architecture a été établi et élaboré en prenant en compte les exigence de notre encadreur.

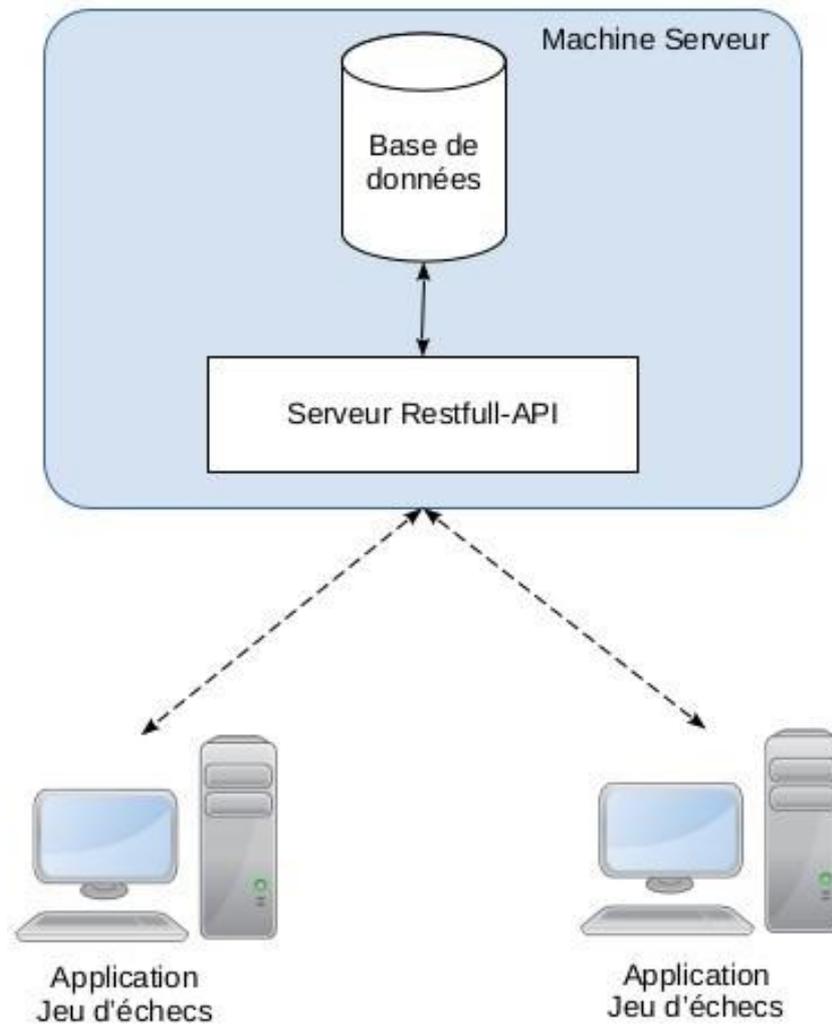


Figure III.1 : Architecture générale de la solution

Dans cette architecture, nous avons deux modules à savoir :

Module Serveur : responsable de la sauvegarde des informations nécessaires aux différents joueurs, parties jouées, Nous pouvons aussi mettre dans le serveur les différents moteur d'échecs qui seront accessible via l'internet.

Module Interface : Il s'agit d'une application lourde (bureau) qui accède aux services du premier module, et pouvant fonctionner localement et indépendamment du serveur.

– Cas d'utilisation

Dans ce qui suit, nous allons identifier les besoins fonctionnels de notre application, en utilisant le formalisme des cas d'utilisation.

– Identification des acteurs

Nous avons un seul acteur, savoir le joueur d'échecs. Ainsi, nous nommons cet action : « Joueur ».

– Identification des cas d'utilisation

Pour l'acteur Joueur, nous avons les cas d'utilisation suivants :

Cas d'utilisation		Acteur principal / Acteurs secondaires	Description
S'authentifier		Joueur	S'identifier pour pouvoir jouer en ligne.
Gérer les bases de parties	Ouvrir une base	Joueur	Manipuler les bases de parties, et créer une nouvelle base,
	Ajouter une base		
	Dupliquer une partie		
	Supprimer une partie		
	Sauvegarder la partie		
Manipuler l'échiquier	Réaliser un coup	Joueur	Manipuler l'échiquier, montrer des coups par des flèches, pouvoir saisir la notation algébrique, ...
	Annuler un coup		
	Aller à un coup		
	Lire une partie		
	Lancer le moteur		
	Personnaliser		
	Annoter la partie		
Apprendre à jouer	Coordonnées	Joueur	Apprendre les différents aspects du jeu d'échecs.
	Pièces		
	Captures		
	Protection		
	Position initiale		
	Fin d'une partie		

Jouer	Contre la machine	Joueur	
	Contre une personne en réseau local		
	Contre une personne en internet		

Tableau III.1 : Identification des cas d'utilisation

– Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation permet de regrouper dans un seul schéma, les différents cas d'utilisation avec, éventuellement, des relations entre ces cas, comme par exemple, *extends*, *includes* et *généralisation* (héritage)

La figure II.1 illustre les différents cas d'utilisation de notre application.

– Diagrammes de séquence

Un diagramme de séquence est un type de diagramme UML qui illustre la succession de messages échangés entre des objets lors d'une interaction. Il se compose de plusieurs objets, représentés par des lignes de vie, ainsi que des messages que ces objets s'envoient durant cette interaction. Son rôle est de décrire un cas d'utilisation avec ses différents scénarios en utilisant les stéréotypes LOOP, ALT ou OPT.

Dans cette section, nous présentons quelques diagrammes de séquences, les autres seront présentés dans l'Annexe D.

– Diagramme de séquence : « Réaliser un coup »

La figure ci-dessous, illustre les messages échangés entre l'acteur « Joueur » et le système pour réaliser un coup :

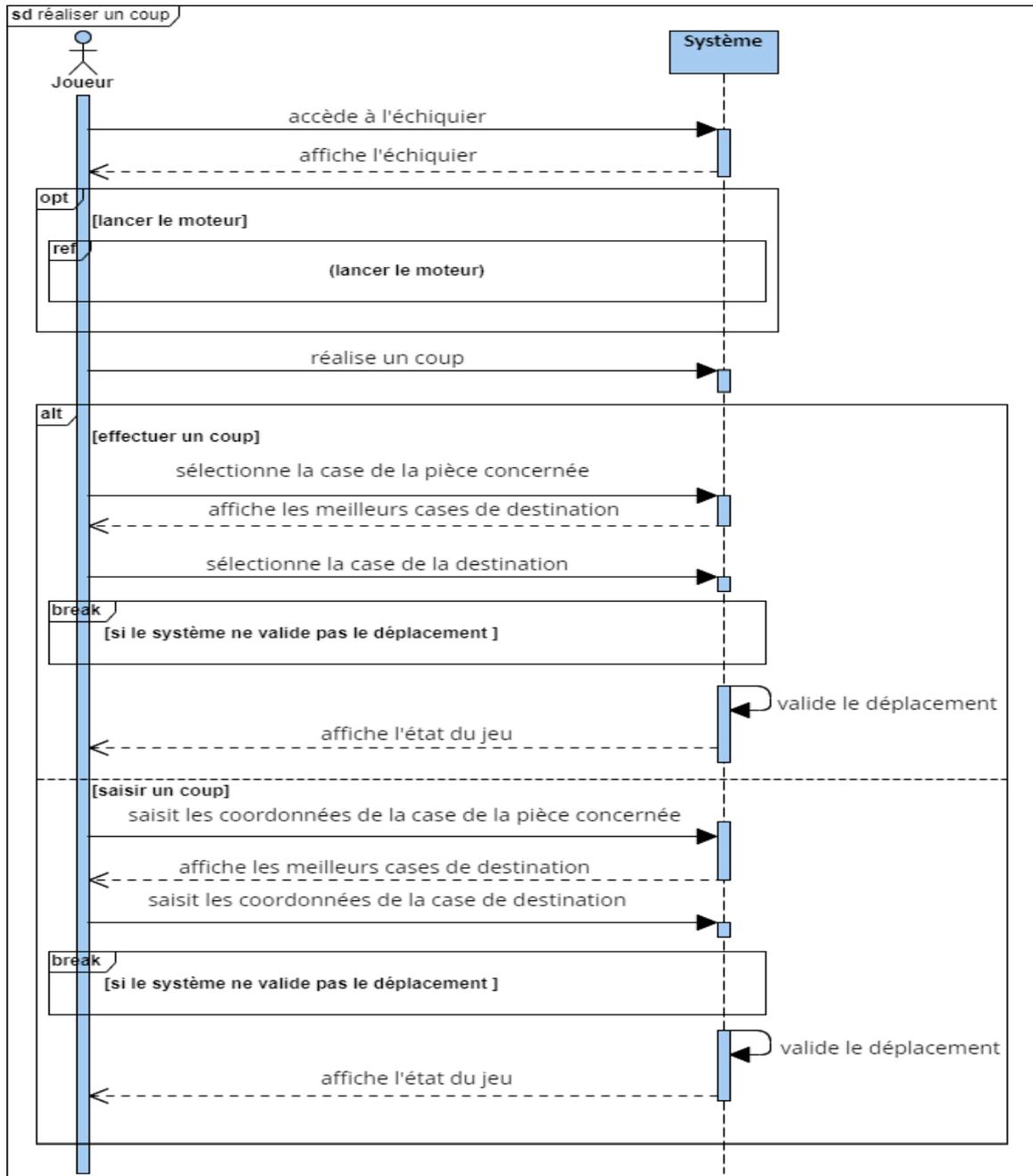


Figure III.3 : Diagramme de séquences « Réaliser un coup »

– Diagramme de séquence : « Annuler un coup »

Voici le diagramme pour le cas : « Annuler un coup » :

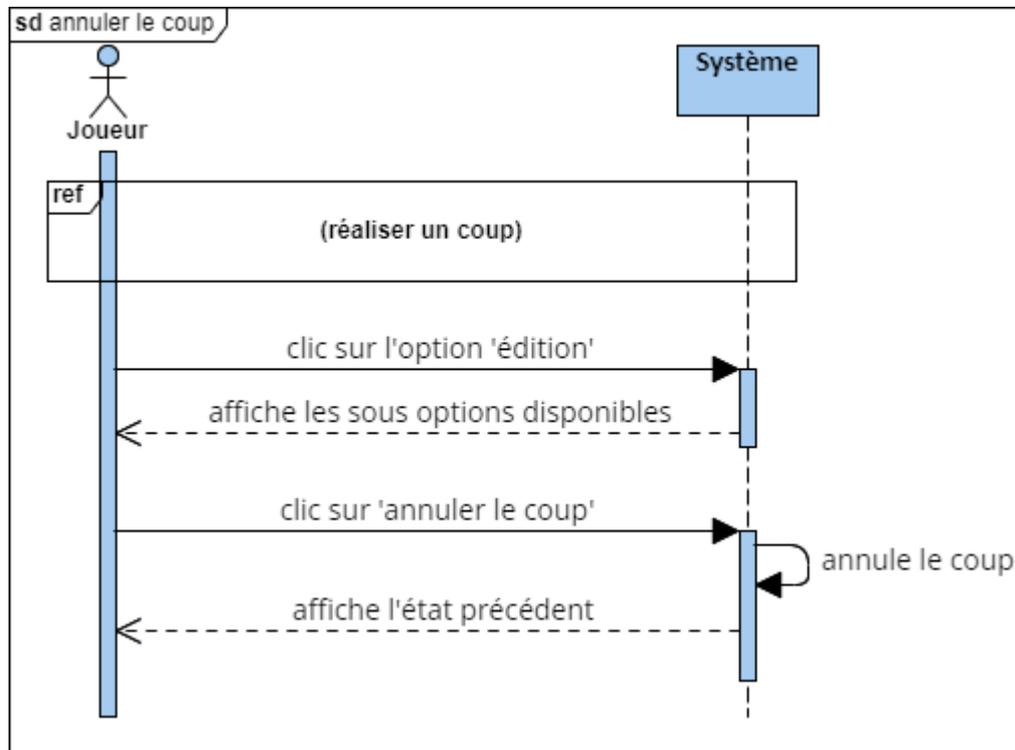


Figure III.4 : Diagramme de séquences «Annuler un coup »

– Diagramme de séquences : « Lancer le moteur »

Le moteur d'échecs permet d'évaluer la position actuelle de l'échiquier et de proposer le(s) meilleur(s) coup(s) à jouer.

Le diagramme de séquence du cas d'utilisation « Lancer le moteur » est illustré par la figure dans la page suivante.

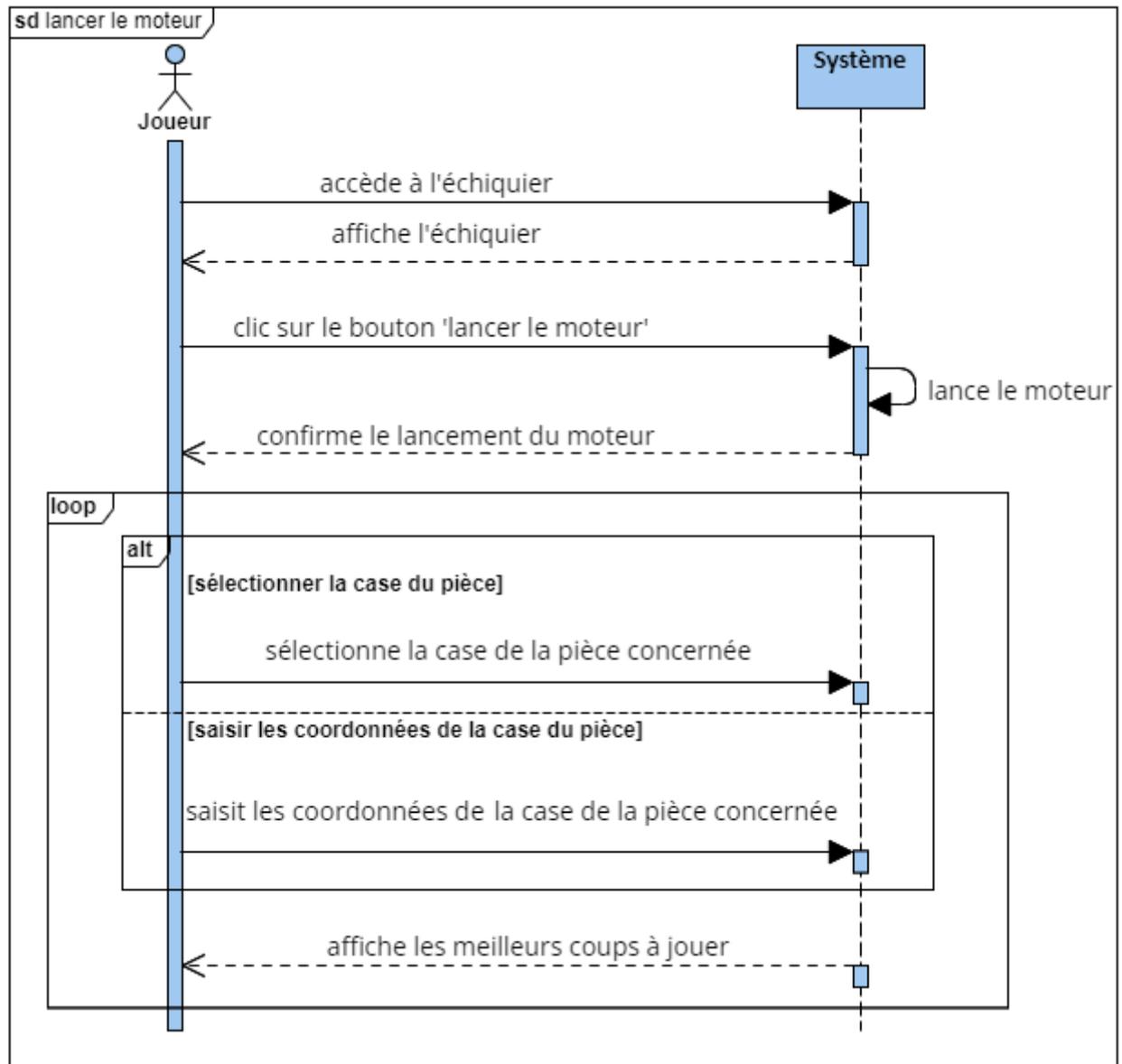


Figure III.5 : Diagramme de séquence “Lancer le moteur”

Pour les autres diagrammes, veuillez les consulter au niveau de l’annexe D.

– Diagramme de classe

Les diagrammes de classes servent de plans pour un système ou un sous-système. Ils permettent de modéliser les objets composant le système, de montrer les relations entre ces objets, ainsi que de décrire leurs fonctions et les services qu'ils offrent.

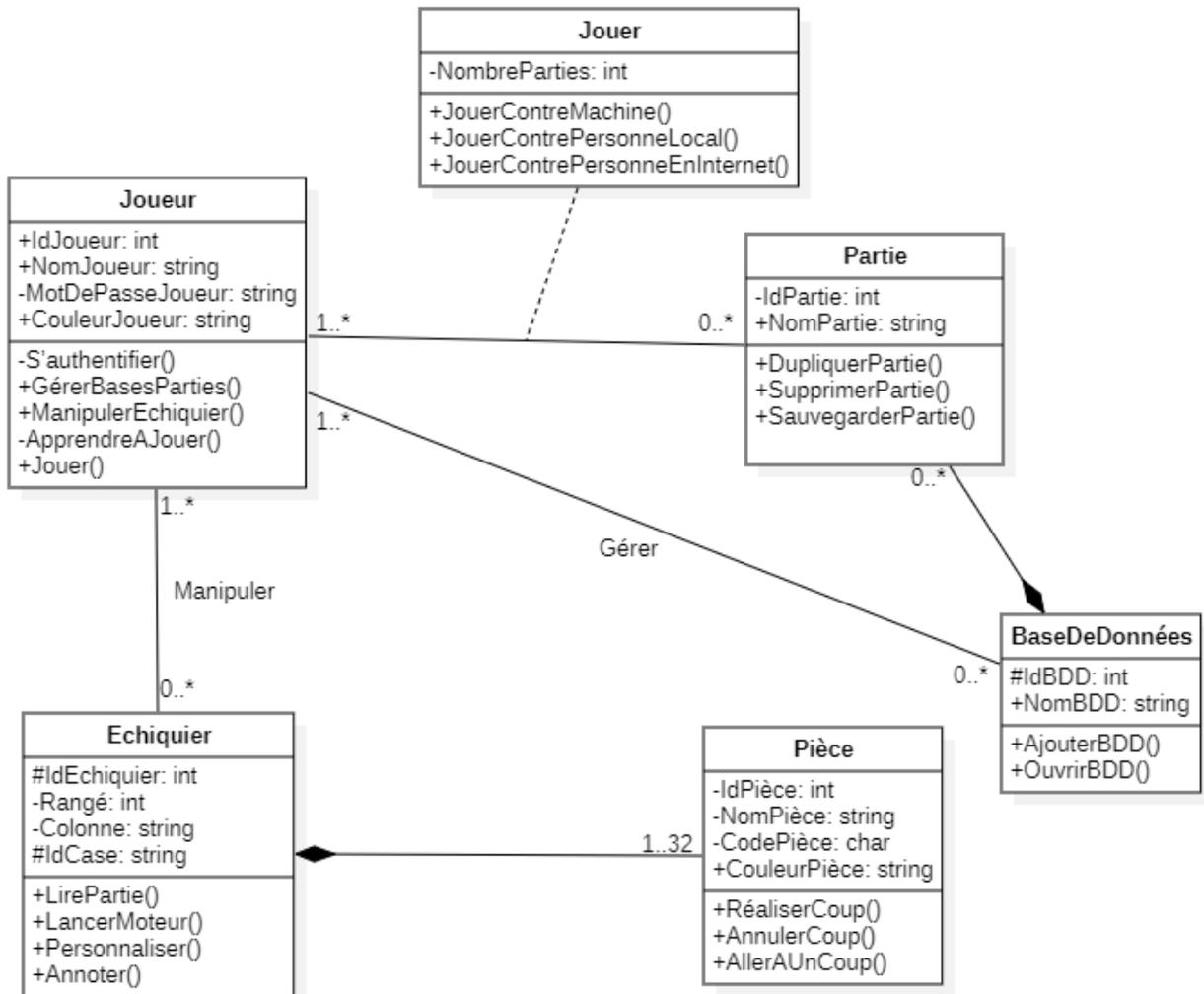


Figure III.6 : Diagramme de classe.

– Conclusion

Dans ce chapitre, dédié à l'identification, analyse et conception des besoins, nous a permis de répondre aux aspects liés aux fonctionnalités à développer, les exigences que nous devons respecter, une conception basée sur le diagramme de classes et sa transformation en schéma relationnel. Ceci nous permettra de préparer la phase suivante, à savoir, l'étape de réalisation et d'implémentation.

CHAPITRE IV
RÉALISATION & TESTS

CHAPITRE IV

Réalisation & Tests

– Introduction

Ce présent chapitre sera dédié à la partie de réalisation notre application et la mise en œuvre de l'ensemble des techniques pour atteindre cette objectif. Par la suite, nous allons montrer quelques captures d'écran de notre application.

– Technologies et langages utilisés

Dans cette section, nous allons énumérer les différentes technologies qui sont utilisées pour développer notre système.

– Sublime Text

Est un éditeur de texte multi-langage qui reconnaît la coloration syntaxique des la plupart des langages de programmation [10]. Pour notre cas, nous l'avons utilisé pour coder la partie serveur d'API en utilisant NodeJS (Javascript).

– IDE Eclipse

IDE (Integrated Development Environment) Eclipse est un environnement de programmation très complet, dédié principalement pour le langage Java [11]. Dans notre projet, nous l'avons utilisé pour implémenter la partie interface de l'application.

XAMPP

XAMPP est une plateforme pour le développement Web qui fonctionne sous Windows, Linux et MacOS. Cette plateforme est utilisée pour créer des applications Web avec PHP et gérer les bases de données avec le SGBD MariaDB. Cette dernière tâche se fait en utilisant l'application PHPMyAdmin [12].

JavaScript

JavaScript est un langage de script incorporé dans les documents HTML et exécuté par le navigateur (Google chrome, Firefox, ...) [13].

NodeJS

Node.js® est un environnement d'exécution JavaScript construit sur le moteur JavaScript V8 de Chrome [14].

– Taux d'avancement

Pour la réalisation de notre application, il y a des fonctionnalités et des aspects non encore achevés. Dans ce qui suit, nous indiquons l'état d'avancement des modules de notre application :

Module Serveur : dans le module serveur, nous avons une base de données implémenté en Maria-DB [15] (variante de MySQL) : 60 %. Pour l'avancement de code javascript, nous sommes à 25 %.

Module Interface : réalisé en langage Java, avec la bibliothèque JavaFx [16], pour implémenter les fonctionnalités liées à l'interface graphique (échiquier, annotation, analyse de parties, ...), nous sommes à 55 % de réalisation.

Pour la communication entre les deux modules, cet aspect n'est pas encore implémenté, et nous estimons sa réalisation à sept jours de travail.

– Captures d'écran

Dans ce qui suit, nous montrons quelques interfaces de notre application, réalisées en JavaFX.

La première interface, *figure IV.*, concerne la fenêtre d'accueil.

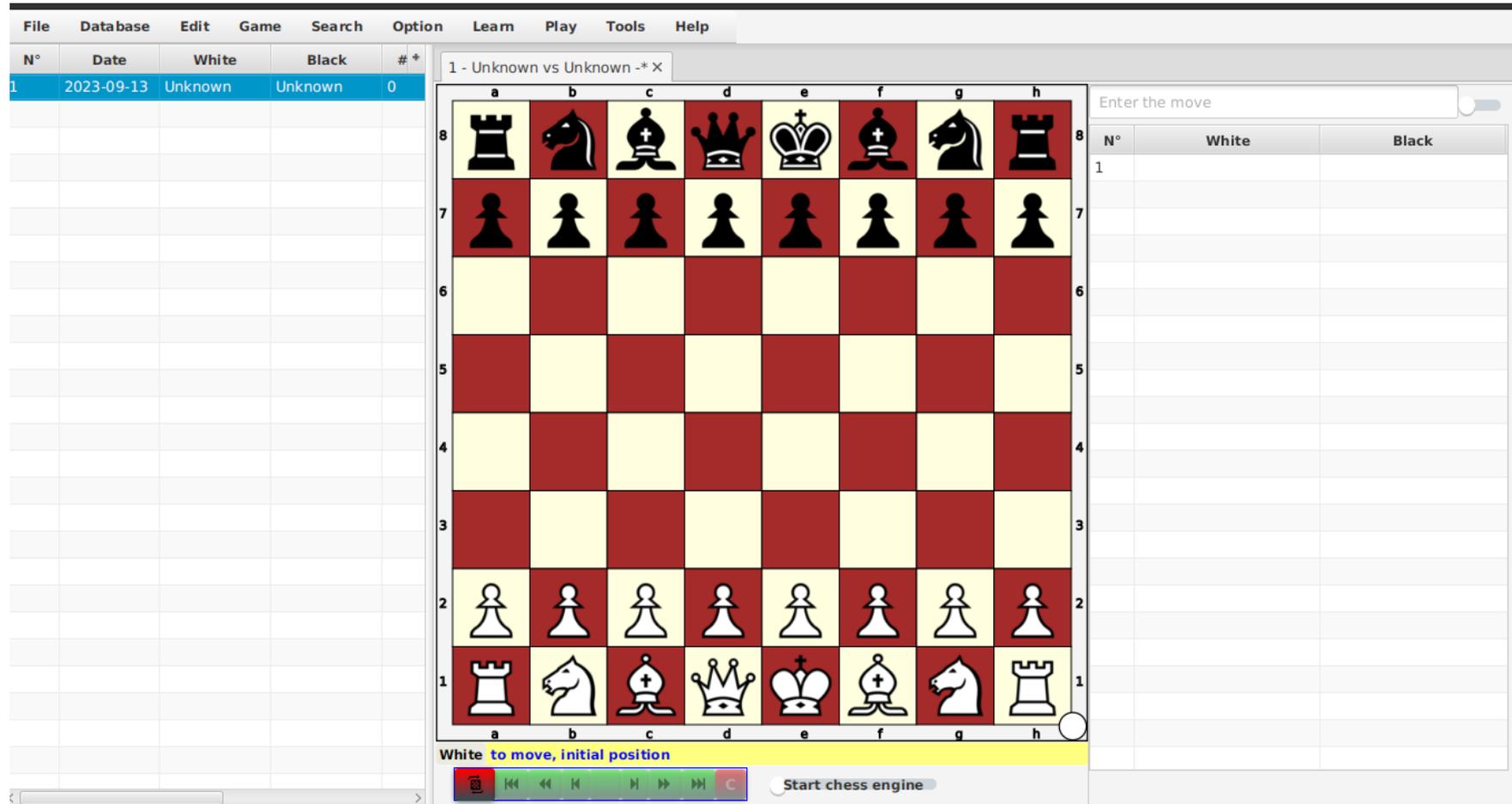


Figure IV.1 : Interface d'accueil de l'application

1 - Carlsen, Magnus vs Cordts, Ingo -0-1 (30 moves) X

N°	White	Black
1	d4	Nf6
2	c4	c5
3	Nf3	cxd4
4	Nxd4	e5
5	Nb5	d5
6	cxd5	Bc5
7	N5c3	O-O
8	e3	e4
9	h3	Re8
10	g4	Re5
11	Bc4	Nbd7
12	Qb3	Ne8
13	Nd2	Nd6
14	Be2	Qh4
15	Nc4	Nxc4
16	Qxc4	b5
17	Qxb5	Rb8
18	Qa4	Nf6
19	Qc6	Nd7
20	d6	Re6
21	Nxe4	Bb7
22	Qxd7	Bxe4

Black to move, white's last move : 11. Bc4, black's previous move : 10... Re5

Figure IV.2 : Bases de partie ouverte

La seconde interface, concerne la fenêtre de configuration et de personnalisation de l'échiquier, des différentes couleurs utilisées pour l'annotation et flèches.

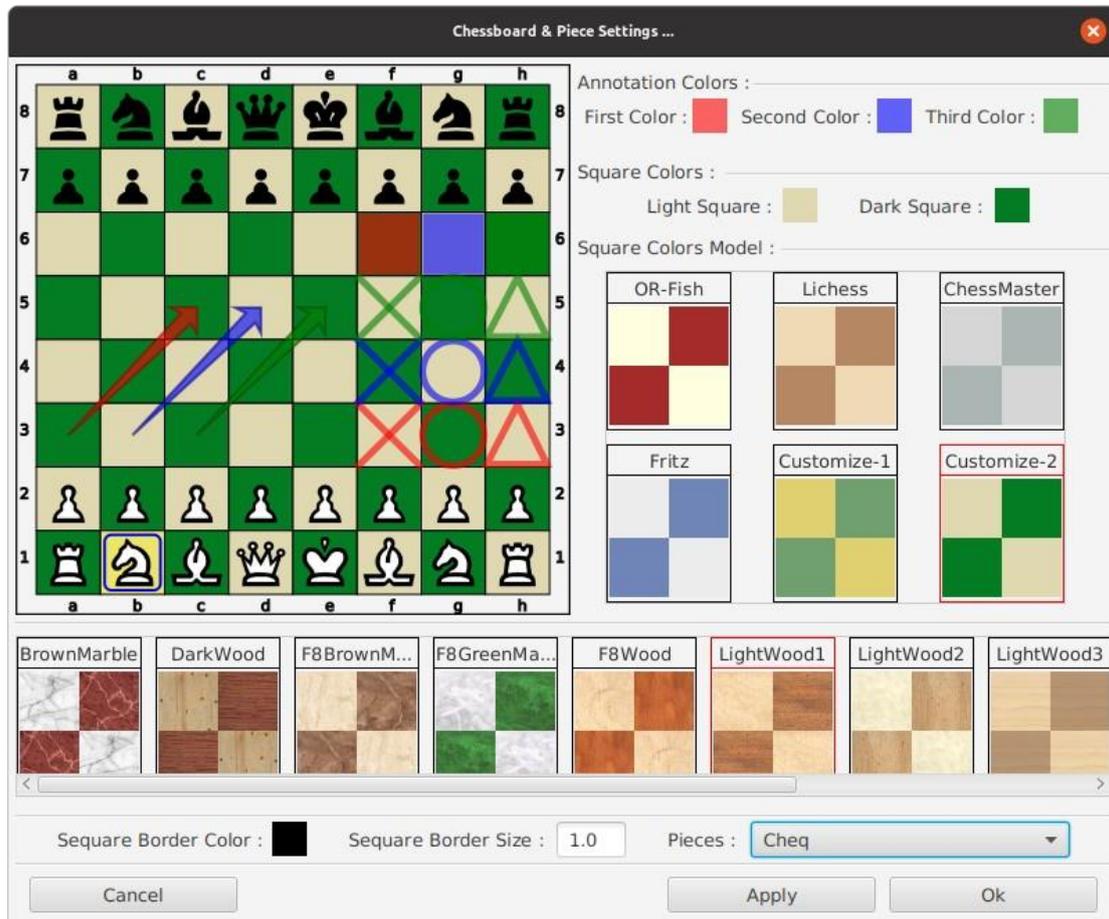


Figure IV.3 : Configuration et Personnalisation de l'échiquier

La troisième interface, concerne l'exercice d'entraînement sur les codes des cases. C'est un simple jeu qui consiste à trouver la case (cliquer sur la bonne case) en sachant son code. Cette exercice s'achèvent au bout de quelques seconds et affiche le nombre de cases correctes.

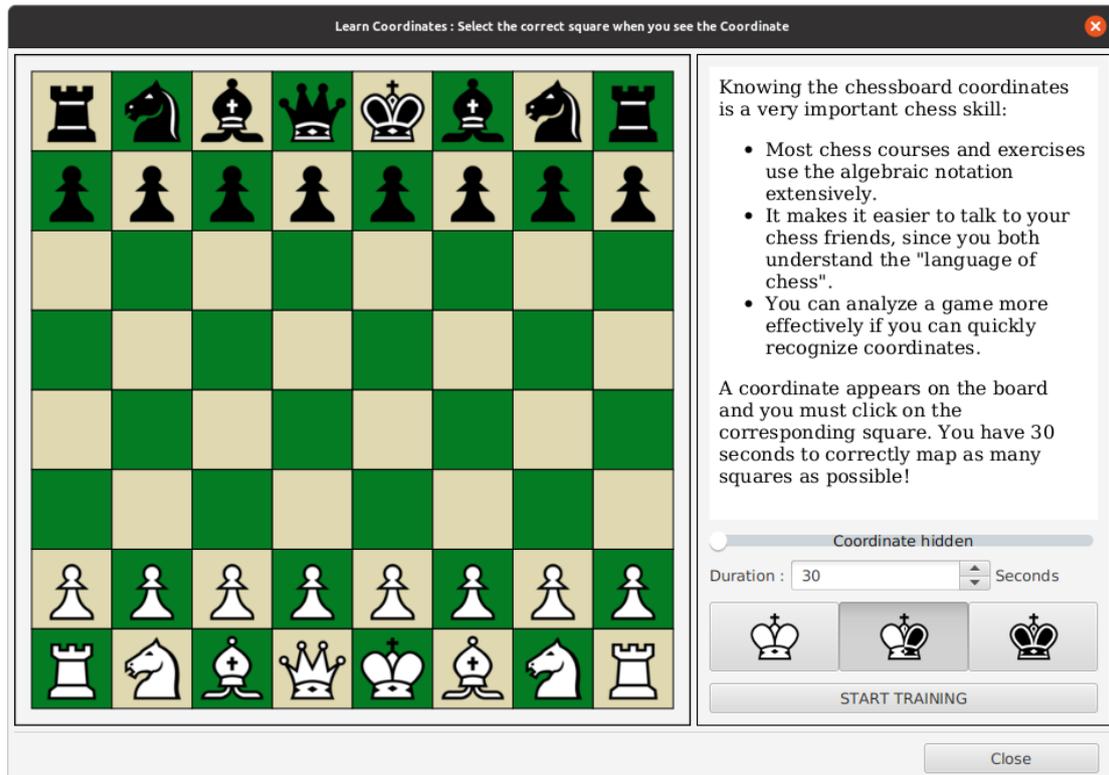


Figure IV.4 : Apprendre les code des case de l'échiquier

La dernière interface, l'interface de jeu avec la machine (contre un moteur d'échecs précédemment configuré).

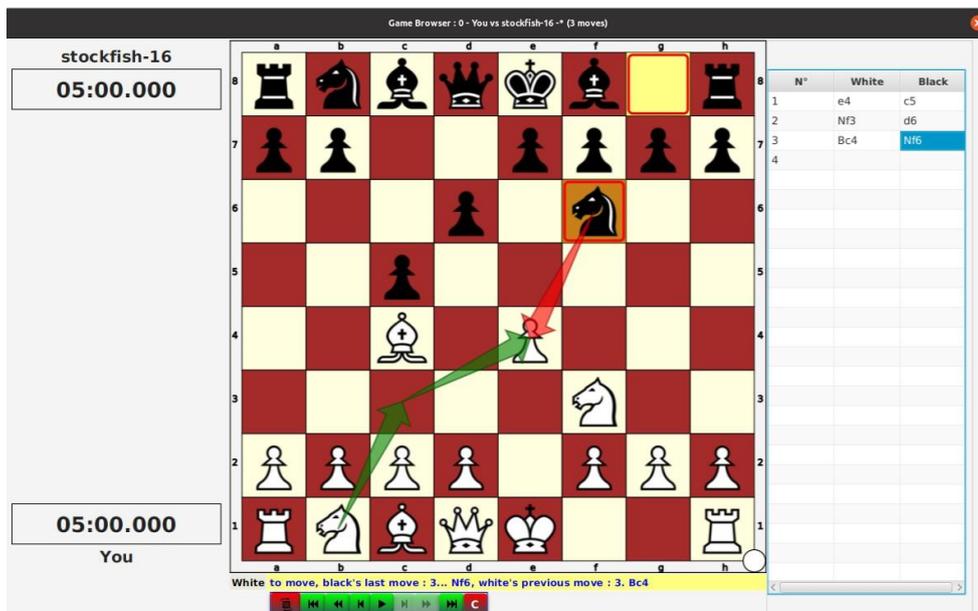


Figure IV.5 : Jouer contre la machine

– Conclusion

Dans ce dernier chapitre, qui représente la phase finale du processus de développement logiciel, nous avons présenté les différents outils et langages informatiques utilisés pour développer notre application. Par la suite, nous avons montré la structure physique de notre base de données, en montrant les tables et leurs interrelations.

Nous avons clôturé ce chapitre par la présentation de quelques interfaces des utilisateurs. En ce qui concerne la phase des tests, le travail est toujours en cours de développements.

**CONCLUSION
GÉNÉRALE**

CONCLUSION GÉNÉRALE

Tout au long de mémoire, nous avons conçu et réalisé une application qui permet d'apprendre et de jouer aux échecs, soit avec un être humain, ou contre une machine.

La problématique principal et de créer une plateforme ouverte et extensible, qui permet à différentes type d'application (web, bureau et mobile) de pouvoir accéder aux différentes bases de parte, de jouer en ligne ou en local. Ceci dit, notre réalisation n'a pas atteint tous ses objectif, néanmoins, une base a été faite pour de future projet afin de perfectionner et compléter les fonctionnalité manquante.

Nous avons pu intégrer deux applications (modules) hétérogènes, une réalisée avec le langage Java (JavaFx), et qui représente l'interface graphique de notre application, et la seconde avec Node-JS afin d'offrir une RESTFUL API afin d'accéder à différent type de service (principalement, l'accès à une base de données).

Comme perspective à notre travail, nous pouvons indiquer les éléments suivant :

- Ajouter une option pour faire jouer une machine contre machine pour comparer leurs performance
- Bien étudier les algorithmes d'intelligence artificielle et proposer des implémentations pou les moteur d'échecs.

BIBLIOGRAPHIE

BIBLIOGRAPHIE

- [01] Larousse, '*échecs – LAROUSSE*', article en ligne, accessible via le lien : <https://www.larousse.fr/encyclopedie/divers/%C3%A9checs/45405>, consulté en mai 2023.
- [02] Larousse, '*Larousse du jeu d'échecs – Découvrir – Approfondir – Maîtriser*', Préface de Joël Lautier. Publié le 1 septembre 1997.
- [03] Brigitte Roederer, Jérôme Mauffras et Pascal Mellent, '*Mon cahier « Jeu d'Échecs »*', document en ligne, accessible via le lien : https://www.dsden93.ac-creteil.fr/spip/IMG/pdf/ECHECS_final-TP.pdf, Académie Créteil (Jeunesse, éducation, recherche).
- [04] Amandine Mussou et Sarah Troche, '*Le jeu d'échecs comme représentation*', article en ligne, accessible via le lien : <https://books.openedition.org/editionsulm/8975?lang=fr>. Éditions Rue d'Ulm (Presse de l'école normale). OpenEdition Books. Consulté en mai 2023.
- [05] Eric Bentzen, '*Chess Fonts for Diagrams and Figurine Notation*', Lien : <https://www.enpassant.dk/chess/fonteng.htm>. @Nørresundby Skakklub. Consulté le : juin 2023.
- [06] Chess.com, '*Chess Engine*' ,Lien : <https://www.chess.com/terms/chess-engine>. Consulté en août 2023.
- [07] Stefan-Meyer Kahlen, article en ligne, lien : <https://www.wbec-ridderkerk.nl/html/UCIProtocol.html>. Consulté en août 2023.

- [08] Free Software Foundation, Inc., 'Chess Engine Communication Protocol', article en ligne, lien : <https://www.gnu.org/software/xboard/engine-intf.html>. Consulté en août 2023.
- [09] Laurant AUDIBERT. 'UML 2 - De l'apprentissage à la pratique'. Documentation en ligne, lien : <https://laurent-audibert.developpez.com/Cours-UML/?page=mise-en-oeuvre-uml>. Publié le 31 octobre 2006 - Mis à jour le 12 janvier 2009 . Consulté en août 2023.
- [10] Sublime HQ Pty Ltd, 'Sublime Text - A sophisticated text editor for code, markup and prose'. Site web. URL : <https://www.sublimetext.com>. Consulté en août 2023.
- [11] Eclipse Foundation, 'The community for open Innovation and Collaboration', ressource en ligne, lien : <https://www.eclipse.org/>. Consulté en mai 2023.
- [12] Kai 'Oswald' Seidler, Kay Vogelgesang, ..., 'XAMPP Apache + MariaDB + PHP + Perl'. Page web. Lien : <https://www.apachefriends.org/index.html>. Consulté en Août 2023.
- [13] Pluralsight team, 'JS JavaScript.com. Ready to try JavaScript ?'. Page Web. URL : <https://www.javascript.com>. Consulté en août 2023.
- [14] Node.js Foundation, 'Node JS. A JavaScript runtime built on Chrome's V8 Javascript engine'. Page Web. Le lien : <https://nodejs.org>. Consulté en août 2023.

- [15] MariaDB Foundation, 'The MariaDB Foundation – Supporting continuity and open collaboration in the MariaDB ecosystem'. Page web. URL : <https://mariadb.org>. Consulté en août 2023.

- [16] Gluon OpenJFX, '*JavaFX*', page web, lien : '<https://openjfx.io/>'. Consulté en avril 2023.

- [17] Chess Programming Wiki – UCI. Page web. URL : <https://www.chessprogramming.org/UCI>. Consultée en août 2023.

ANNEXES

ANNEXE A

Histoire de jeu d'échecs

C'est très délicat de tracer plus précisément l'histoire de jeu d'échecs, néanmoins , dans cet annexe nous résentons les grandes étapes par lesquelles a passé ce noble jeu.

Origines et Développement

Le jeu d'échecs, tel que nous le connaissons aujourd'hui, a une histoire riche et complexe qui s'étend sur plus de mille ans. Ses origines remontent à l'Inde du VI^e siècle, où il était connu sous le nom de **chaturanga**. Ce jeu ancien était joué sur un plateau de 8x8 cases et impliquait quatre divisions militaires : l'infanterie, la cavalerie, les éléphants et les chars, qui correspondent respectivement aux pions, cavaliers, fous et tours dans le jeu moderne.

Diffusion en Perse et en Arabie

Chaturanga s'est propagé de l'Inde à la Perse, où il est devenu **shatranj**. Le shatranj a ensuite été introduit dans le monde islamique après la conquête de la Perse au VII^e siècle. Les Arabes ont adopté et diffusé le jeu à travers leur empire, en apportant des modifications mineures aux règles et aux pièces. Le jeu a atteint l'Europe par le biais de l'Espagne musulmane et de la Sicile au IX^e siècle.

Évolution en Europe

En Europe, les échecs ont subi plusieurs transformations majeures à la fin du Moyen Âge. Au XV^e siècle, en Espagne et en Italie, les mouvements de certaines pièces ont été modifiés pour rendre le jeu plus dynamique et intéressant. Par exemple, la reine (à l'origine un conseiller) est devenue la

pièce la plus puissante sur l'échiquier, capable de se déplacer dans toutes les directions sur n'importe quel nombre de cases.

Normalisation et Tournois

Les règles modernes des échecs ont été standardisées au XIX^e siècle. La première compétition officielle de tournois d'échecs a eu lieu à Londres en 1851 et a été remportée par l'Allemand Adolf Anderssen. Depuis lors, les échecs ont évolué pour inclure des championnats du monde, des Olympiades d'échecs, et une structure formelle de classements et de titres, tels que Grand Maître International (IGM) et Maître International (IM).

L'Ère Numérique

L'avènement des ordinateurs et d'Internet a révolutionné le jeu d'échecs. Les moteurs d'échecs, comme Deep Blue d'IBM, qui a battu le champion du monde Garry Kasparov en 1997, ont poussé les limites du jeu. Aujourd'hui, des plateformes en ligne comme Chess.com et Lichess permettent aux joueurs du monde entier de s'affronter instantanément, de suivre des cours et d'analyser leurs parties.

Le jeu d'échecs a une histoire fascinante qui traverse plusieurs cultures et époques. Depuis ses origines anciennes jusqu'à son évolution moderne avec les technologies numériques, les échecs continuent de captiver et de défier des millions de personnes dans le monde entier.

ANNEXE B

Programmes & applications sur les échecs

Dans ce deuxième annexe, nous présentons quelques sites web et programmes utilisés dans le jeu d'échecs, soit pour jouer en ligne ou étudier des parties et des bases de données.

1. Chess.com

Chess.com est l'un des sites d'échecs les plus populaires au monde, offrant des parties en ligne, des leçons, des puzzles, et des analyses de parties. Il propose des fonctionnalités pour les joueurs de tous niveaux, du débutant au grand maître.

- URL: <https://www.chess.com/>

2. Lichess

Lichess est une plateforme gratuite et open-source pour jouer aux échecs en ligne. Elle offre des parties en direct, des puzzles, des analyses de parties, et des tournois. Lichess se distingue par son absence de publicité.

- URL: <https://lichess.org/>

3. Fritz

Fritz est un logiciel d'échecs commercial développé par ChessBase. Il offre des fonctionnalités avancées pour l'analyse des parties, des leçons interactives, et des parties contre l'ordinateur.

- URL: <https://fritz.chessbase.com/>

4. Arena

Arena est un logiciel gratuit qui sert d'interface graphique pour divers moteurs d'échecs. Il supporte les protocoles UCI et Winboard et permet de jouer contre des moteurs ou d'analyser des parties.

- URL: <http://www.playwitharena.de/>

5. Stockfish

Stockfish est un moteur d'échecs open-source considéré comme l'un des plus puissants au monde. Il est souvent utilisé en combinaison avec d'autres interfaces graphiques comme Arena ou ChessBase.

- URL: <https://stockfishchess.org/>

6. Playchess

Playchess est une plateforme en ligne gérée par ChessBase. Elle propose des parties en ligne, des tournois, et des analyses de parties. Elle est utilisée par de nombreux joueurs de haut niveau.

- URL: <https://play.chessbase.com/en/>

7. ChessBase

ChessBase est un logiciel de base de données d'échecs utilisé pour l'analyse des parties, la recherche dans les bases de données de parties, et la préparation contre des adversaires spécifiques. C'est un outil essentiel pour les joueurs professionnels.

- URL: <https://en.chessbase.com/>

8. Shredder

Shredder est un logiciel d'échecs qui offre une forte capacité d'analyse et la possibilité de jouer contre l'ordinateur à différents niveaux de difficulté. Il propose également des puzzles et des leçons.

- URL: <https://www.shredderchess.com/>

9. Internet Chess Club (ICC)

ICC est une plateforme d'échecs en ligne qui propose des parties, des leçons, des tournois, et des analyses. Elle est fréquentée par de nombreux joueurs de haut niveau.

- URL: <https://www.chessclub.com/>

10. FIDE Online Arena

La FIDE Online Arena est la plateforme officielle de la Fédération Internationale des Échecs (FIDE) pour jouer aux échecs en ligne et obtenir des classements officiels.

- URL: <https://chessarena.com/>

11. SCID (Shane's Chess Information Database)

SCID est un système populaire de gestion de base de données d'échecs qui permet aux utilisateurs de stocker, d'organiser et d'analyser les parties d'échecs, les tournois et les joueurs. C'est un outil puissant pour les amateurs d'échecs, les joueurs et les entraîneurs, qui leur permet d'améliorer leurs compétences et de se préparer aux compétitions.

- URL: <https://scid.sourceforge.net/>

12. Lucas Chess

Lucas Chess est un programme d'échecs gratuit et open-source qui offre une gamme de fonctionnalités pour vous aider à améliorer vos compétences aux échecs. Il s'agit d'une alternative populaire aux logiciels d'échecs commerciaux, et il est disponible pour Windows, macOS et Linux.

- URL: <https://lucaschess.pythonanywhere.com/>

En guise de conclusion, ces programmes et sites web couvrent une large gamme de besoins, de l'apprentissage et l'amélioration des compétences à l'analyse avancée et la compétition en ligne. Que vous soyez débutant ou joueur expérimenté, vous trouverez des ressources adaptées à votre niveau et à vos objectifs.

ANNEXE C

Protocole UCI

UCI, pour Universal Chess Interface, est un protocole de communication pour les moteurs du jeu d'échecs, pour analyser des positions ou jouer des parties automatiquement. Ce protocole est conçu et développé par Rudolf Huber et Stefan Meyer-Kahlen [17].

La spécification du protocole UCI est disponible, pour téléchargement, sur le lien : <http://download.shredderchess.com/div/uci.zip> .

Voici quelques spécifications, en anglais, de ce protocole :

- * all communication is done via standard input and output with text commands,

- * The engine should boot and wait for input from the GUI, the engine should wait for the "isready" or "setoption" command to set up its internal parameters as the boot process should be as quick as possible.

Move format:

The move format is in long algebraic notation.

A nullmove from the Engine to the GUI should be sent as oooo.

Examples: e2e4, e7e5, e1g1 (white short castling), e7e8q (for promotion)

GUI to engine:

These are all the command the engine gets from the interface.

- * uci

 - tell engine to use the uci (universal chess interface),

 - this will be sent once as a first command after program boot

 - to tell the engine to switch to uci mode.

After receiving the uci command the engine must identify itself with the "id" command and send the "option" commands to tell the GUI which engine settings the engine supports if any. After that the engine should send "uciok" to acknowledge the uci mode. If no uciok is sent within a certain time period, the engine task will be killed by the GUI.

* debug [on | off]

switch the debug mode of the engine on and off. In debug mode the engine should send additional infos to the GUI, e.g. with the "info string" command, to help debugging, e.g. the commands that the engine has received etc. This mode should be switched off by default and this command can be sent any time, also when the engine is thinking.

* isready

this is used to synchronize the engine with the GUI. When the GUI has sent a command or multiple commands that can take some time to complete, this command can be used to wait for the engine to be ready again or to ping the engine to find out if it is still alive.

E.g. this should be sent after setting the path to the tablebases as this can take some time. This command is also required once before the engine is asked to do any search to wait for the engine to finish initializing. This command must always be answered with "readyok" and can be sent also when the engine is calculating in which case the engine should also immediately answer with "readyok" without stopping the search.

....

Engine to GUI:

* id

* name <x>

this must be sent after receiving the "uci" command to identify the engine, e.g. "id name Shredder X.Y\n"

* author <x>

this must be sent after receiving the "uci" command to identify the engine, e.g. "id author Stefan MK\n"

* uciok

Must be sent after the id and optional options to tell the GUI that the engine has sent all infos and is ready in uci mode.

* readyok

This must be sent when the engine has received an "isready" command and has processed all input and is ready to accept new commands now. It is usually sent after a command that can take some time to be able to wait for the engine, but it can be used anytime, even when the engine is searching, and must always be answered with "isready".

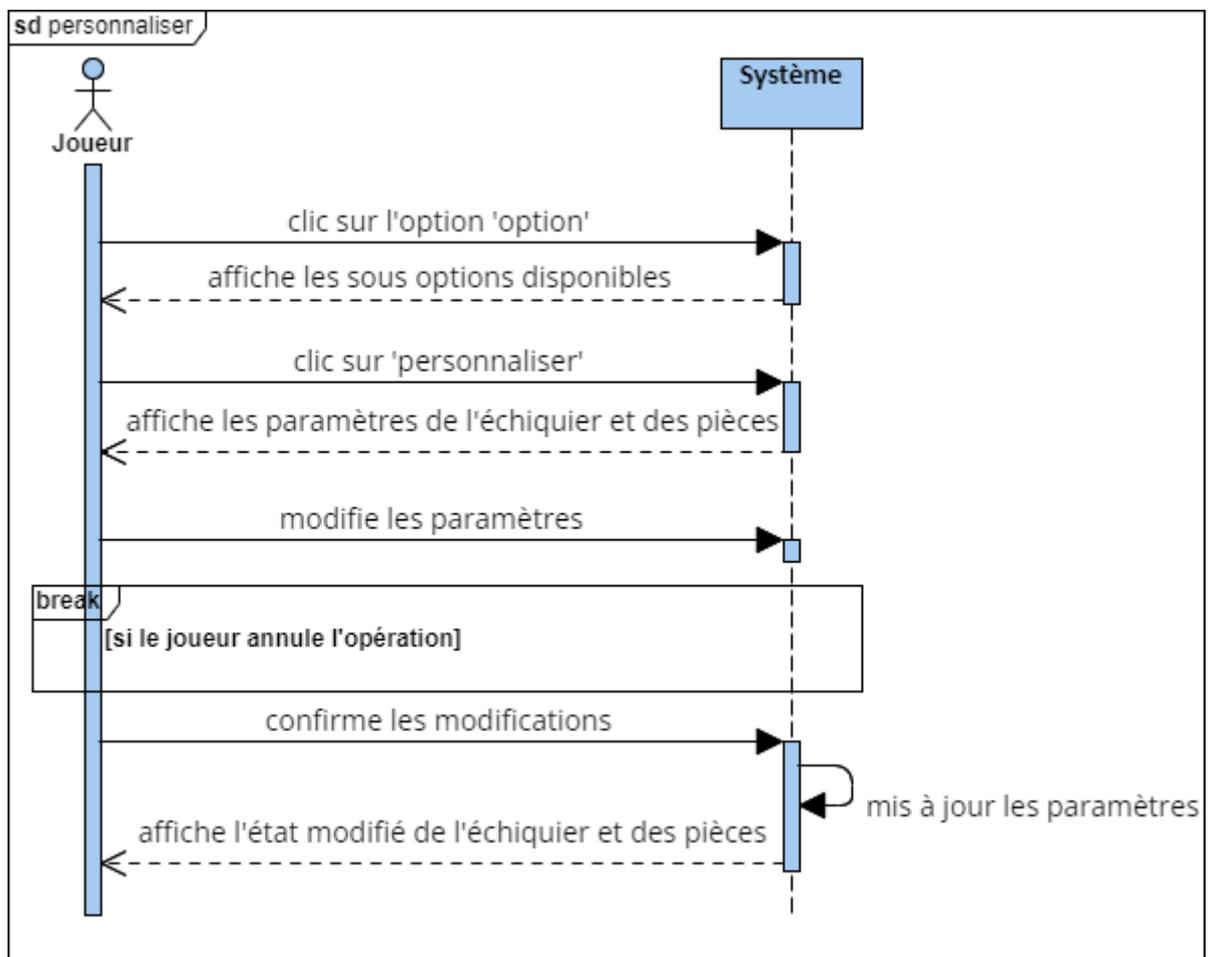
Pour plus, d'information, veuillez consulter :
<https://gist.github.com/DOBRO/2592c6dad754ba67e6dcaec8c90165bf>

ANNEXE D

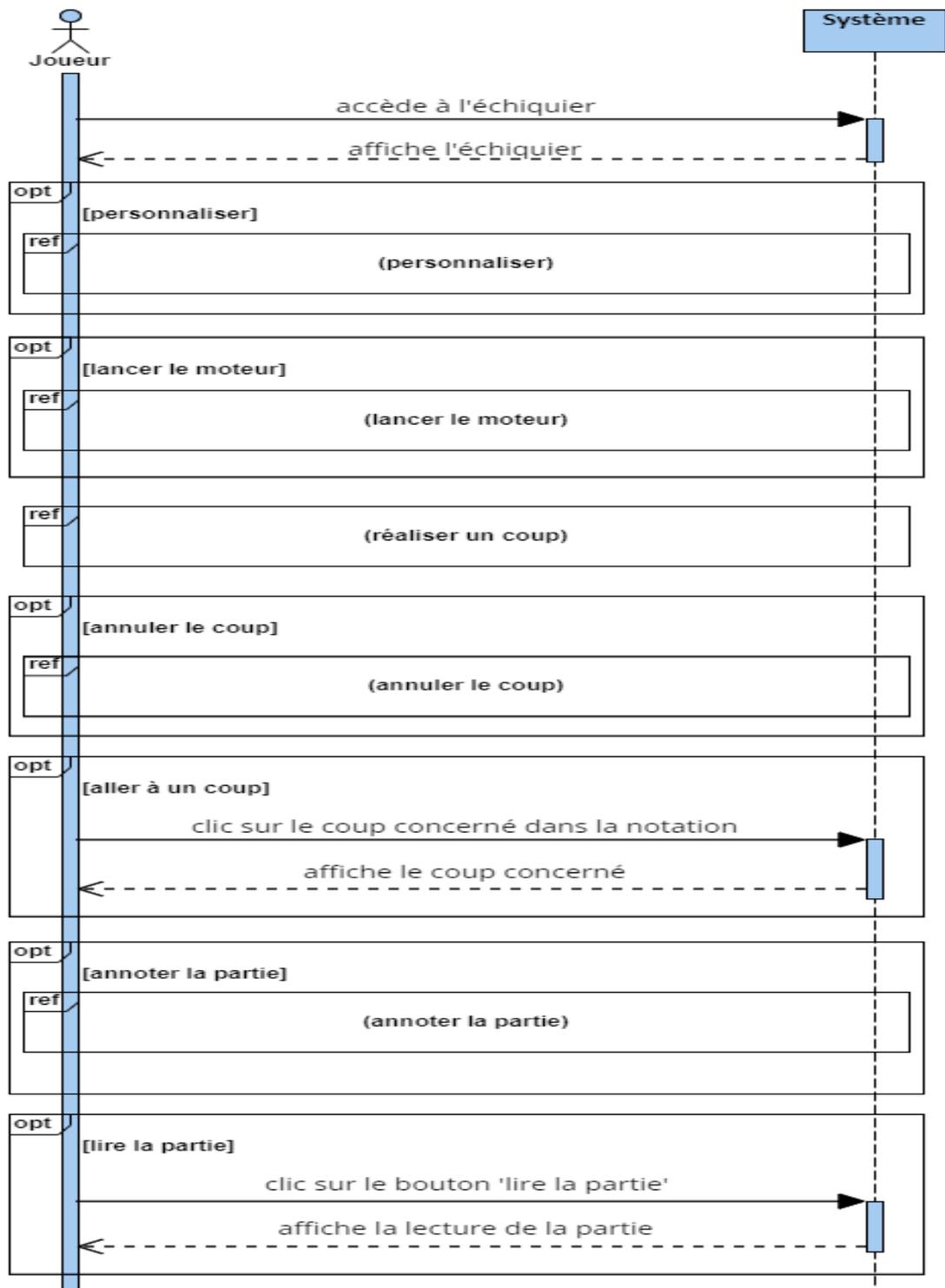
Diagrammes UML

Cette annexe est dédiée au différents diagramme d'UML qui modélise un aspect particulier de notre application. Ces diagrammes sont mis dans cette section pour éviter de saturer notre mémoire (limitation de nombre de pages).

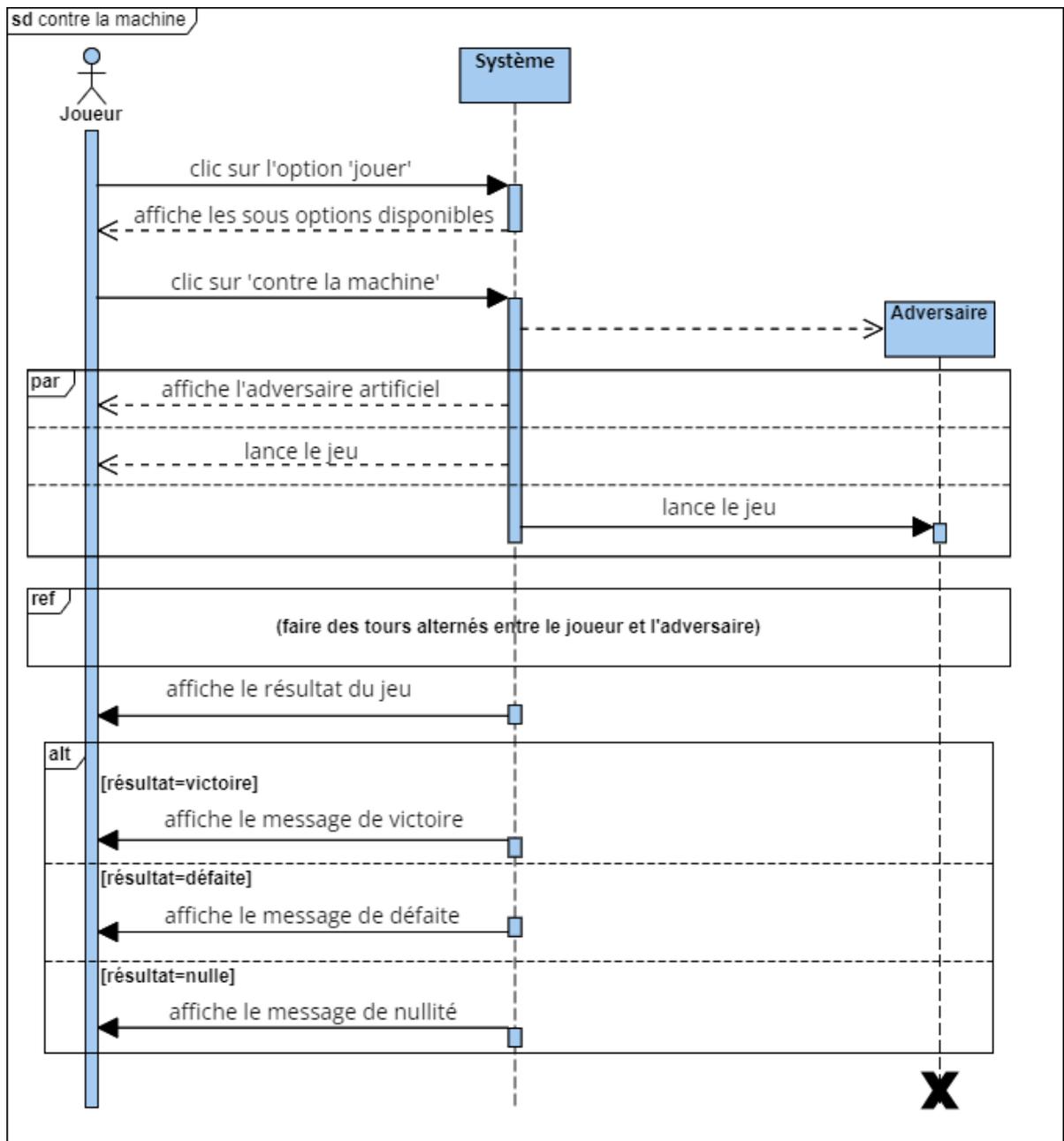
- Diagrammes de Séquences « Personnaliser l'échiquier »



- Diagrammes de Séquences «Manipuler l'échiquier »



- Diagrammes de Séquences «Jouer contre la machine »



Résumé

Tout au long de ce travail, nous avons conçu et réalisé une application qui permet d'apprendre et de jouer aux échecs, La problématique principal et de créer une plateforme ouverte et extensible, qui permet à différentes type d'application (web, bureau et mobile) de pouvoir accéder aux différentes bases de parte, de jouer en ligne ou en local. Nous avons pu communiquer deux application hétérogènes, une réalisée avec le langage Java (JavaFx), et qui représente l'interface graphique de notre application, et la seconde avec Node-JS afin d'offrir une RESTFUL API afin d'accéder à différent type de service (principalement, l'accès à une base de données).

Mots Clés :

Échecs, moteur d'échecs, Stockfish, protocole UCI, Interface graphique de jeu d'échecs

Abstract

Throughout this project, we've been designing and building an application for learning and playing chess. The main challenge has been to create an open, extensible platform that allows different types of application (web, desktop and mobile) to access different game databases, and to play online or locally. We were able to communicate two heterogeneous applications, one built using the Java language (JavaFx), which represents the graphical interface of our application, and the second using Node-JS to offer a RESTFUL API to access different types of service (mainly, access to a database).

Keywords :

Chess, Chess engine, Stockfish, UCI protocol, Chess graphical interface