



# PROJET PROLOG – ABALONE

.....  
Antoine CHABERT - Lisa COURANT - Paul DAUTRY - Pierre JARSAILLON - Estelle LEPEIGNEUX - Hugues VERLIN

Hexanome 4401

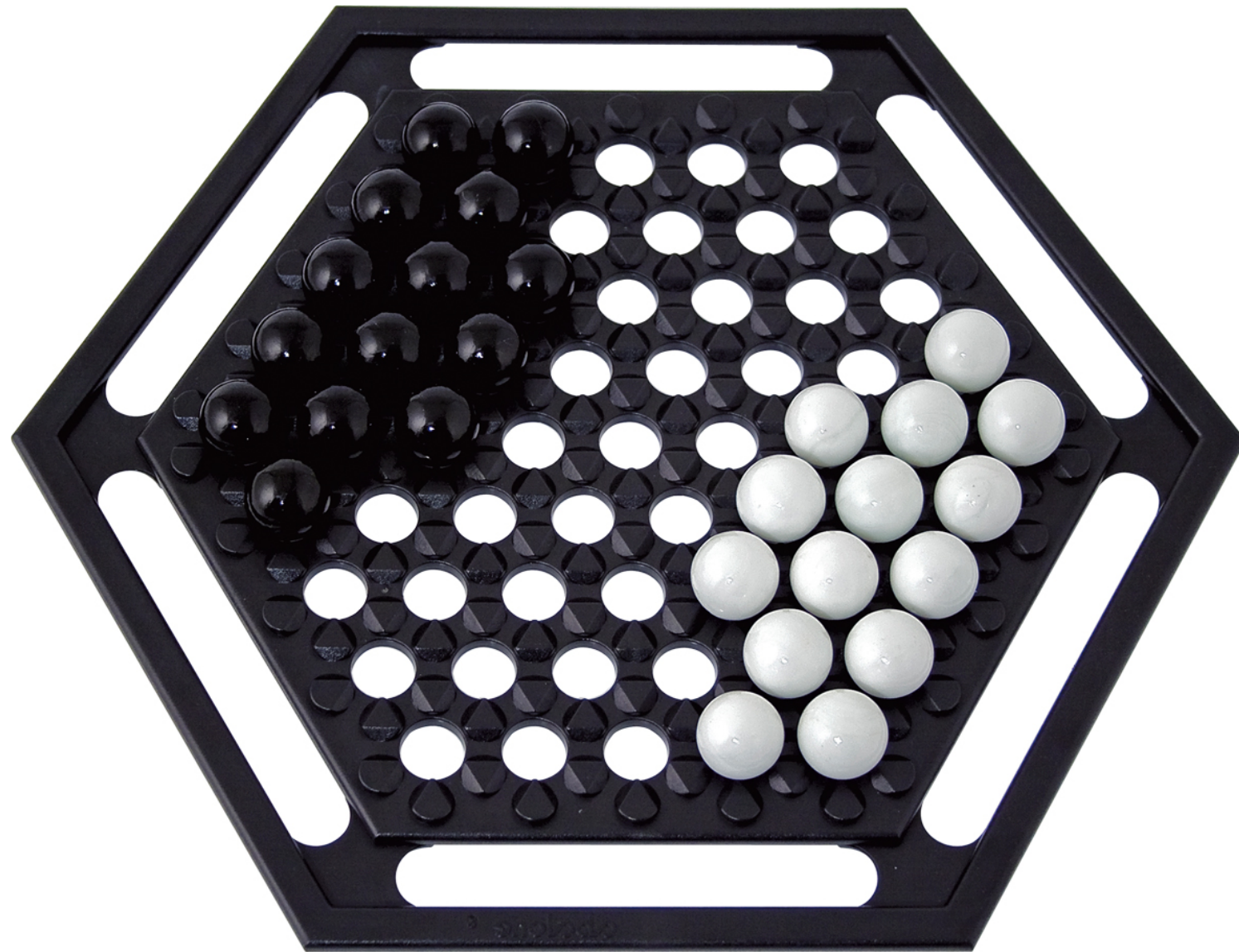
# PLAN

---

1. Présentation du jeu et motivations
2. Déroulement du projet
3. Intelligences artificielles et stratégies
4. Interfaces et démonstration
5. Résultats
6. Conclusion

# 1. PRÉSENTATION DU JEU ET MOTIVATIONS

---



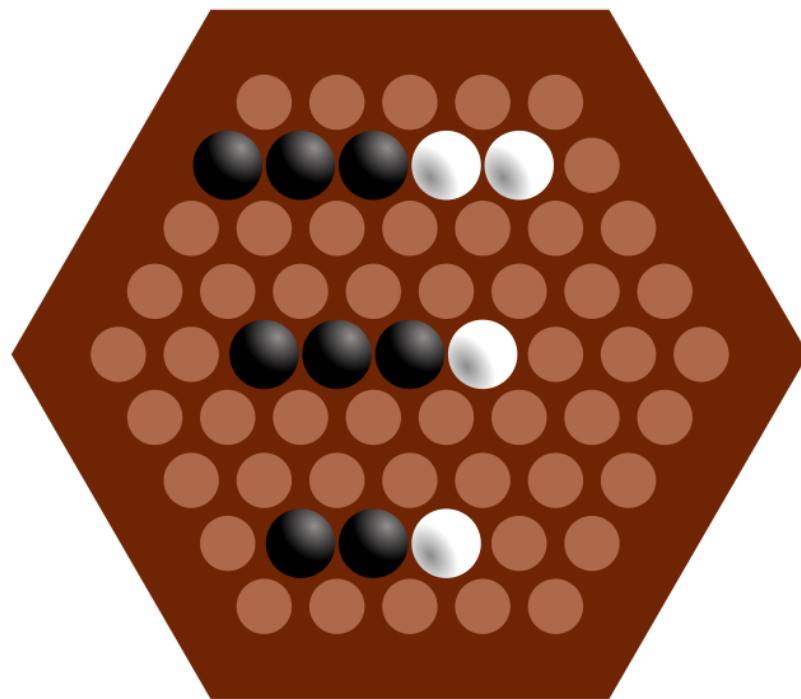
---

Abalone - Plateau de jeu

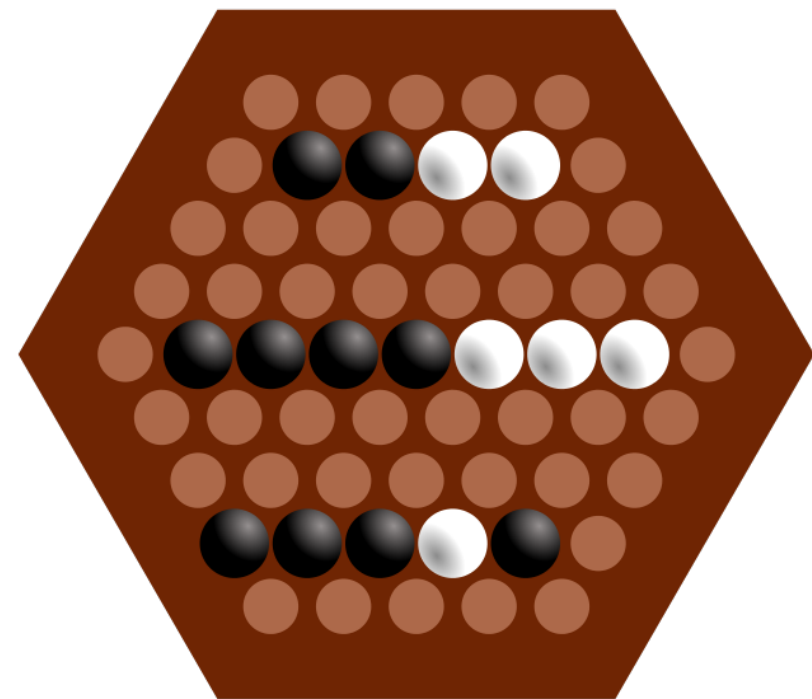
# 1. PRÉSENTATION DU JEU ET MOTIVATIONS – RÈGLES DU JEU

---

- Objectif : Éjecter 6 billes adverses
- Déplacements autorisés :
  - Une, deux ou trois billes adjacentes sur une même rangée
  - Supériorité numérique (sumito) requise



Exemple de « sumito »



« Sumito » impossibles

# 1. PRÉSENTATION DU JEU ET MOTIVATIONS – POURQUOI ABALONE ?

.....

- Jeu présentant plusieurs difficultés :
  - Peu d'informations sur les meilleurs coups à jouer
  - Environ 60 mouvements possibles par coup (jusqu'à 148)
  - Utilisation impossible d'un algorithme brute force
  - Les meilleurs programmes actuels sont incapables de battre les champions du monde d'Abalone, avec un **temps limité par coup**.

Game	Branching factor	log(game-tree size)
Checkers	8-10	31
Othello	~5	58
Chess	30-40	123
Xiangqi	75	150
Abalone	60-80	154
Go	360	360



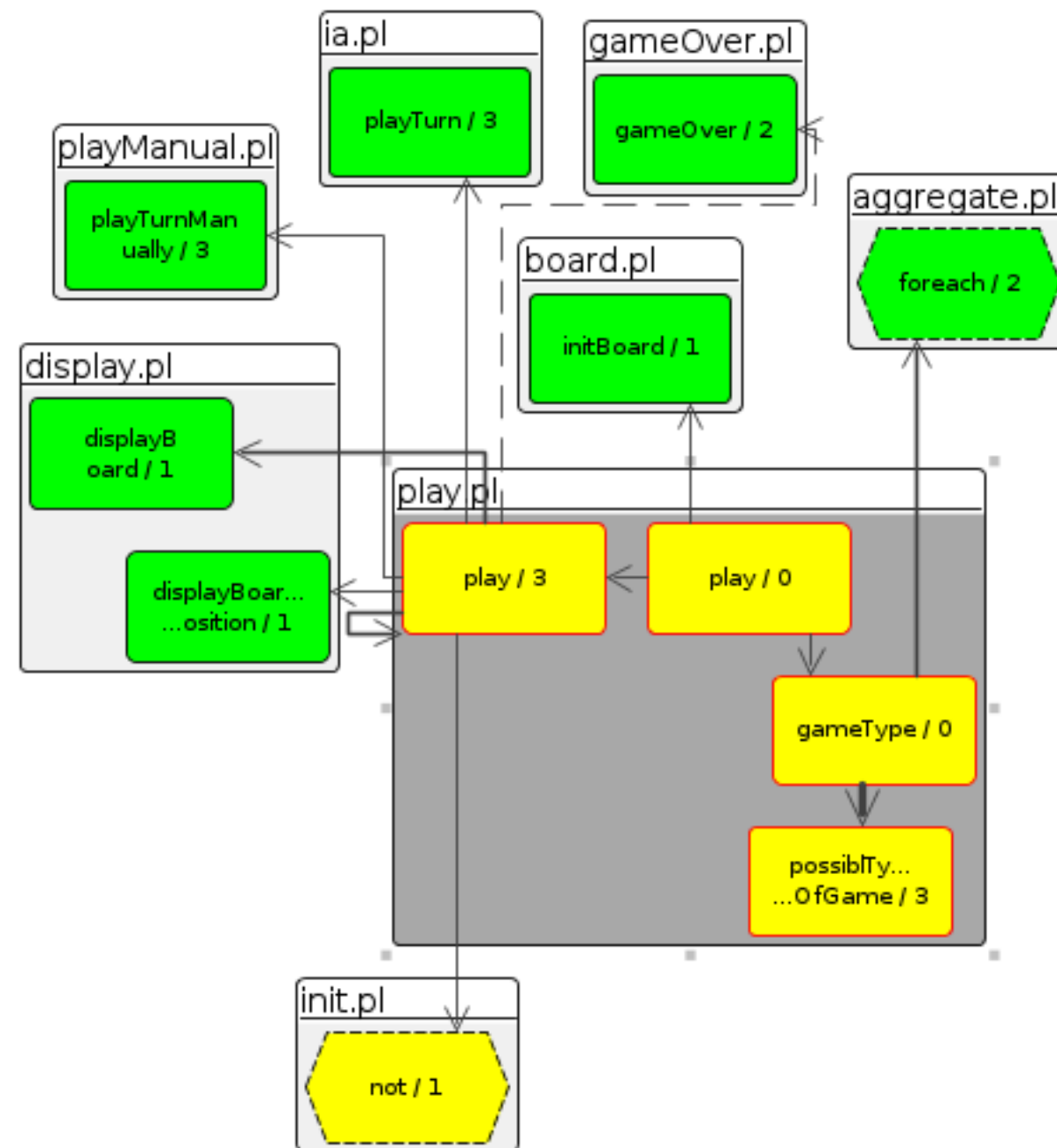
## 2. DÉROULEMENT DU PROJET – ORGANISATION

---

- Chef de projet : Lisa
  
- Répartition globale des tâches
  - Réflexion commune au problème
  - **Coeur algorithmique** : Lisa, Paul, Pierre
  - **Fonctionnement de base des IA** : Antoine, Pierre
  - **Implémentation des heuristiques** : Estelle, Hugues
  - **Interfaces** : Antoine, Estelle
  - **Profilage des solutions** : Lisa, Paul

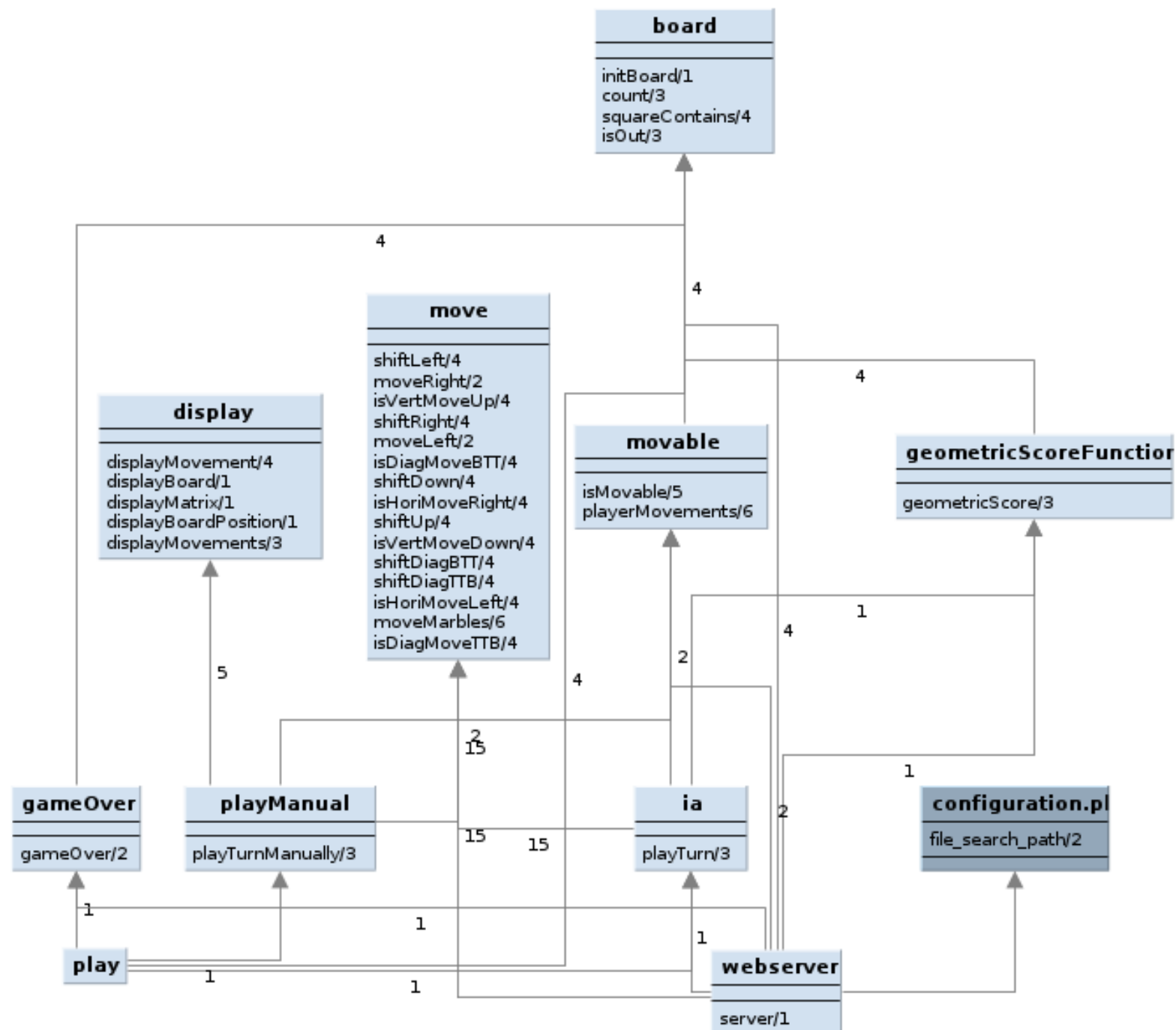
## 2. DÉROULEMENT DU PROJET – ARCHITECTURE

### ➤ Architecture en modules



## 2. DÉROULEMENT DU PROJET – ARCHITECTURE

### ➤ Architecture en modules





# 3. INTELLIGENCES ARTIFICIELLES ET STRATÉGIES

---

## ➤ Stratégies générales

- Avoir un jeu compact
- Être proche du centre, les bords étant désavantageux
- Avoir un grand nombre de « sumito » possibles

## ➤ Pistes d'heuristiques

- Nombre de billes sur le plateau
- Calcul d'un score en fonction de la position des billes
- Nombre de « sumito » possibles

# HEURISTIQUE PRINCIPALE

---

## ➤ Fonctionnement :

1. Calcul du **centre de masse** des deux joueurs
2. Calcul du **barycentre  $M$**  de ces deux centres de masse et du centre du plateau
3. Calcul des **distances de Manhattan** entre chaque bille et  $M$ . Ajout d'une très grande distance pour les **billes éjectées**.
4. **Comparaison** des deux scores

# INTELLIGENCES ARTIFICIELLES DÉVELOPPÉES

---

## ➤ IA Aléatoire

- Sélection aléatoire du déplacement
- Permet de tester les déplacements et le coeur algorithmique

## ➤ IA AlphaBeta

- Utilisation de l'algorithme Alpha/Beta
- Utilisation d'heuristiques pour évaluer les solutions
- Deux niveaux d'heuristiques :
  - Calcul basé uniquement sur le nombre de billes
  - Amélioration avec la position des billes

# INTELLIGENCES ARTIFICIELLES DÉVELOPPÉES

---

## ➤ IA AlphaBeta

- Personnalisation du niveau de l'IA
  - Fonction de la profondeur de l'arbre de recherche
- Personnalisation de l'agressivité de l'IA
  - Fonction du score attribué aux déplacements favorisant l'éjection de billes

# 4. INTERFACES ET DÉMONSTRATION

- Interface console
  - Réalisée en Prolog
  - Choix du type de jeu
  - Proposition des déplacements possibles

```
Sélection du type de jeu :
1: Humain VS Ordinateur
2: Humain VS Humain
3: Ordinateur VS Ordinateur
Choix 1
Lancement du mode : Humain VS Ordinateur
-----
Iteration : 1
Tour du joueur 1 ( x )

      A B C D E
      / / / / / F
      ----- / G
1-/ x x . . . \ / H
2-/ x x . . . \ / I
3-/ x x x . . . \ /
4-/ x x x . . . o \
5-| x x x . . . o o o |
6-\ x . . . . o o o /
7-\ . . . . o o o /
8-\ . . . . o o /
9-\ . . . . o o /
      -----

Sélection une bille à déplacer, donner la ligne (ex. 3) |: 1
Donner la colonne (ex. D) |: A
Liste des déplacements possibles :
1: 1A -> 1B
2: 1A -> 2B
Sélection le déplacement |: 2

      A B C D E
      / / / / / F
      ----- / G
1-/ . x . . . \ / H
2-/ x x . . . \ / I
3-/ x x x . . . \ /
4-/ x x x x . . . o \
5-| x x x . . . o o o |
6-\ x . . . . o o o /
7-\ . . . . o o o /
8-\ . . . . o o /
9-\ . . . . o o /
      -----
```

# 4. INTERFACES ET DÉMONSTRATION

---

## ➤ Interface web

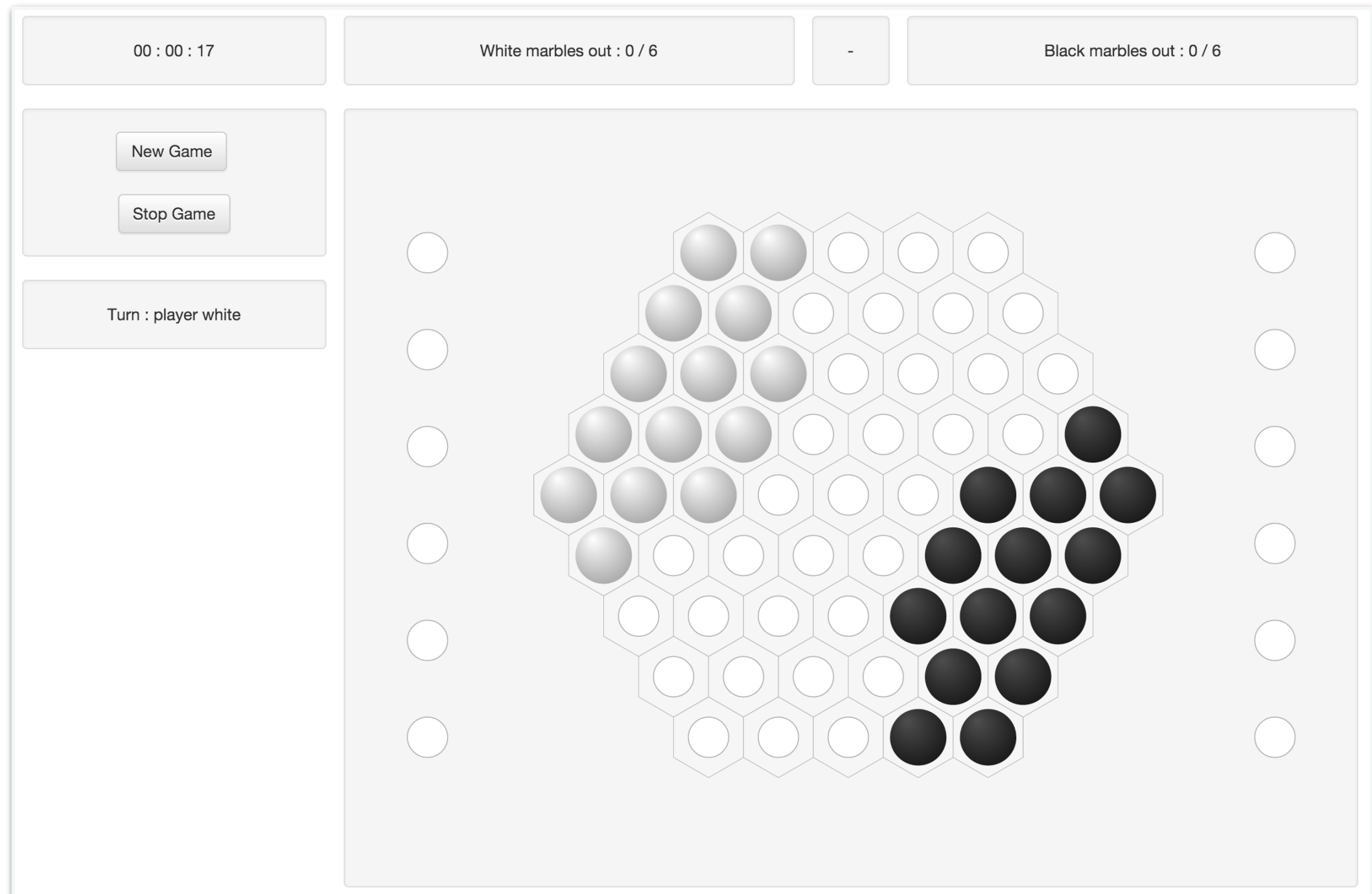
- Utilise un serveur Prolog et du javascript
- Plateau dessiné en SVG
- Modes de jeu de l'interface console
  - Humain vs. Humain
  - Humain vs. IA
  - IA vs. IA
- Configuration de l'IA
  - Agressivité
  - Niveau de recherche



# 4. INTERFACES ET DÉMONSTRATION

---

## ► Démonstration de l'interface web



# 5. RÉSULTATS

---

## ➤ Performance d'exécution des IA

IA	Temps d'exécution pour jouer un coup (secondes)
IA Aléatoire	0,00012
IA AlphaBeta de niveau 1	0,01950
IA AlphaBeta de niveau 2	0,31800
IA AlphaBeta de niveau 3	4,08500
IA AlphaBeta de niveau 4	40,10200

# 5. RÉSULTATS

## ► Performances de l'IA AlphaBeta

		IA Depth 1			IA Depth 2			IA Depth 3		
		Gentle	Default	Aggressive	Gentle	Default	Aggressive	Gentle	Default	Aggressive
IA Depth 1	Gentle	-	$\infty$	1A	1G	1G	1G	$\infty$	$\infty$	$\infty$
	Default	1G	-	1A	1D	1D	1D	$\infty$	$\infty$	$\infty$
	Aggressive	1G	$\infty$	-	1A	1A	1A	$\infty$	$\infty$	$\infty$
IA Depth 2	Gentle	$\infty$	$\infty$	$\infty$	-	2D	0	2G	2G	2G
	Default	1G	1D	1A	$\infty$	-	2A	$\infty$	$\infty$	$\infty$
	Aggressive	1G	1D	1A	$\infty$	$\infty$	-	$\infty$	$\infty$	$\infty$
IA Depth 3	Gentle	1G	$\infty$	1A	3G	3G	3G	-	0	0
	Default	1G	$\infty$	1A	3D	3D	0	$\infty$	-	0
	Aggressive	1G	$\infty$	1A	3A	3A	0	$\infty$	$\infty$	-

# 5. RÉSULTATS

## ➤ Performances de l'IA AlphaBeta - Variations aléatoires

		IA Depth 1			IA Depth 2		
		Gentle	Default	Aggressive	Gentle	Default	Aggressive
IA Depth 1	Gentle	-	3/10	5/10	2/10	1/10	3/10
	Default	3/10	-	6/10	4/10	1/10	2/10
	Aggressive	6/10	3/10	-	1/10	4/10	2/10
IA Depth 2	Gentle	6/10	7/10	6/10	-	3/10	4/10
	Default	7/10	8/10	8/10	4/10	-	2/10
	Aggressive	5/10	8/10	8/10	2/10	5/10	-

# 6. CONCLUSION