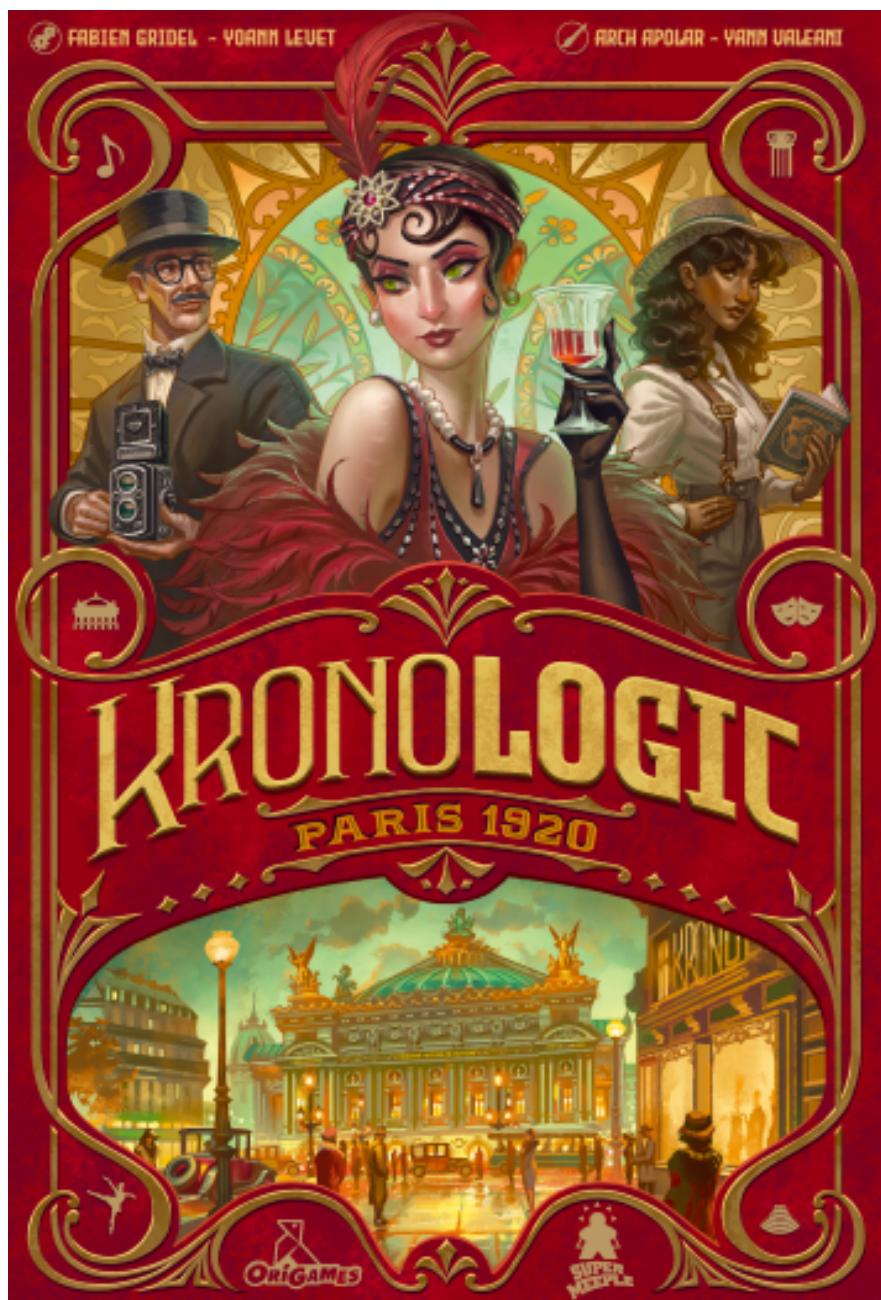


Etude préalable



Sommaire

1. Présentation du projet	3
2. Explication du jeu	4
3. Cahier des charges	7
A. Objectifs Fonctionnels	7
B. Contraintes et Limitations	9
C. Outils et Technologies	10
4. Liste des fonctionnalités	13
A. Fonctionnalités liées à l'interface utilisateur (Vue Carte)	13
B. Fonctionnalités liées à l'interface utilisateur (Vue Tableaux)	15
C. Fonctionnalités liées à l'IA de déduction	16
D. Fonctionnalités liées à l'IA d'assistance	16
E. Fonctionnalités liées à l'IA joueuse	17
F. Fonctionnalités liées au générateur de scénarios	17
5. Cas d'utilisation - Diagrammes de CU - Scénarios - Conditions de validation	18
Diagramme de cas d'utilisation	25
6. Diagrammes de classe, de séquences, d'activité	26
7. Maquette de l'application	27
8. Recensement et évaluation des risques	35
A. Risques Organisationnels	35
B. Risques Liés à l'Expérience Utilisateur (UX)	36
C. Évaluation Globale des Risques	36
9. Planning des itérations et rôle de chacun	37

1. Présentation du projet

Le projet tutoré "Kronologic" a comme but principal de concevoir une intelligence artificielle (IA) qui puisse accompagner ou remplacer un joueur dans le jeu de société Kronologic, publié en 2024 par SuperMeeple et créé par Fabien Gridel et Yoann Levet. Ce jeu est un défi de déduction où les joueurs enquêtent sur un meurtre en reconstituant les événements à partir d'indices, tout en respectant des règles strictes de déduction.

Objectifs du Projet :

- **Assimilation et intégration des règles** : Concevoir une interface qui intègre les règles du jeu, sur laquelle on peut jouer.
- **IA déduction** : Développer une IA qui se base sur les choix du joueur afin de faire des déductions.
- **IA assistante** : Développer une IA capable de guider le joueur vers des choix de questions adaptée pour accélérer la fin du jeu.
- **IA joueuse** : Développer une IA capable de résoudre des enquêtes de manière indépendante.
- **Générateur de scénarios** : Créer un système de génération automatique de nouvelles enquêtes avec des solutions équilibrées.

Description du jeu de société “Kronologic” :

Kronologic est un jeu de déduction où les joueurs enquêtent sur un meurtre en identifiant **le coupable, le lieu et le moment du crime**. Chaque enquête se déroule sur une carte avec plusieurs lieux, personnages, et moments dans le temps.

Les joueurs posent des questions pour obtenir des indices publics et privés, puis utilisent leur logique pour rassembler les informations et résoudre l'énigme. Le jeu propose un mode solo et multijoueur, avec des règles strictes sur les déplacements des personnages et des mécanismes favorisant la stratégie et l'observation.

2. Explication du jeu

1. Objectif du Jeu :

Résoudre une enquête de meurtre en découvrant **qui est le tueur, où le meurtre a eu lieu**, et à quel moment précis il s'est produit. Chaque enquête se déroule sur une carte avec plusieurs lieux, personnages et moments dans le temps. Les joueurs posent des questions pour obtenir des informations publiques et privées, et utilisent leur logique pour assembler les indices.

2. Éléments de Jeu :

- ❖ **Mode solo** : Le joueur enquête seul et doit résoudre le mystère en utilisant ses questions.
- ❖ **Mode multijoueur** : Plusieurs joueurs enquêtent en tour par tour. Le premier à résoudre l'enquête gagne, mais tous doivent utiliser astucieusement leurs questions et écouter les indices révélés par les questions des autres joueurs.

3. Composition d'une Enquête :

- ❖ Une enquête correspond à une carte avec une **difficulté** déterminée.

<i>- Poison Mondain -</i>					
	Enquête 1	Enquête 2	Enquête 3	Enquête 4	Enquête 5
Difficulté	★	★★	★★	★★	★★★

Exemple : enquête Poison Mondain et ses 5 difficultés.

- ❖ Chaque enquête a une seule carte comportant plusieurs lieux (ex : 6).



Exemple : Carte de l'Opéra National de Paris.

- ❖ Un lieu correspondant au **lieu du meurtre**.
- ❖ Plusieurs **personnages** (ex : 6) sont présents sur la carte.

- ❖ **Un tueur** est à identifier.
- ❖ **Plusieurs temps** (ex : 6) définissent les différents moments du **meurtre**.
- ❖ **Un temps spécifique** est celui du **meurtre**.

4. Règles de Déplacement :

- ❖ **Chaque personnage** se trouve dans **un lieu à un moment donné**.
- ❖ **Les personnages se déplacent obligatoirement à chaque temps vers un lieu relié par un passage.**
- ❖ **Un personnage NE PEUT PAS rester sur le même lieu.**
- ❖ **Un personnage** peut retourner **plusieurs fois dans le même lieu à des temps différents**.
- ❖ **Chaque lieu** peut être **occupé par un nombre variable de personnages** (de 0 à 6).
- ❖ **Chaque lieu** est relié à d'autres lieux par **des passages**. **Chaque lieu** doit être **rélié à un minimum de deux autres lieux**.



Exemple : Lieu des Escaliers avec les passages en blanc.

5. Le Rôle du Joueur :

Le joueur peut poser un nombre illimité de questions pour résoudre l'enquête. Chaque question porte sur un lieu spécifique et concerne soit le nombre de personnages présents à un moment donné, soit les déplacements d'un personnage spécifique.

6. Les Questions :

Les questions donnent **deux types d'informations** : **une information publique** que **tous les joueurs reçoivent** et une **information privée** que **seul le joueur qui a posé la question reçoit**.

- ❖ Question sur un lieu et un temps :
 - **Information publique** : Donne le nombre de personnages présents à ce moment précis.

- **Information privée :** Donne également l'identité d'un personnage présent sur ce lieu et ce temps choisi.
- ❖ **Question sur un lieu et un personnage :**
 - Donne le nombre de fois que le personnage est passé dans le lieu.
 - Donne également le temps d'un des passages du personnage dans ce lieu.

Exemple :

1. Vous demandez : "Combien de personnages étaient dans la salle S au temps T ?"
 - **Réponse publique :** 2 personnages.
 - **Réponse privée :** Un des personnages présents était le DéTECTIVE.
2. Vous demandez : "Combien de fois est-ce que le personnage Z est passé dans la salle S ?"
 - **Réponse publique :** 2 fois.
 - **Réponse privée :** Le personnage est passé au temps 3.

7. Cas du "Rejouer" :

Le "rejouer" est parfois présent à la place de l'**information privée** de la réponse.

- ❖ **En multijoueur :** cela permet de poser une nouvelle question au lieu de mettre fin à son tour, et la question révélant le "rejouer" ne compte pas dans le total des questions posées.
- ❖ **En solo :** ne possède pas d'utilité, il s'agit d'une information privée manquée ! En plus de cela, celle-ci compte dans le total des questions posées.

8. Déduction :

Chaque joueur peut à tout moment effectuer une déduction.

- ❖ Si cette dernière est juste, il gagne la partie et cette dernière s'arrête.
- ❖ S'il s'est trompé, il ne peut plus faire de déduction mais il peut quand même continuer de jouer.

9. Fin de l'Enquête :

Le nombre de questions posées détermine votre performance dans l'enquête :

	Loupe d'Or	1-7	1-9	1-8	1-9	1-12
	Loupe d'Argent	8-13	10-15	9-14	10-15	13-18
	Loupe de Bronze	14+	16+	15+	16+	19+

Exemple : les difficultés d'une enquête et le nombre de questions correspondant à la réussite de l'enquête.

3. Cahier des charges

A. Objectifs Fonctionnels

Reproduction des Règles du Jeu

L'objectif principal est de reproduire fidèlement les règles du jeu Kronologic pour pouvoir réaliser une énigme, notamment en ce qui concerne :

- ❖ **Mode de jeu** : Implémentation d'un mode de jeu solo mais pas multijoueur.
- ❖ **Gestion des indices** : Le jeu doit permettre l'affichage et la collecte des indices disponibles à chaque étape.
- ❖ **Déplacements des personnages** : Chaque personnage du jeu suit un itinéraire et est lié à une salle précise pour chacun des 6 temps. Les déplacements entre les salles doivent respecter des passages adjacents, comme spécifiés dans les règles du jeu. Il est également interdit qu'un personnage reste dans la même salle durant 2 pas de temps successifs.
- ❖ **Conditions de victoire** : La victoire est déterminée par la découverte du coupable, du lieu et du moment du crime. Le joueur doit interpréter les indices pour innocenter des personnages ou pour désigner le véritable coupable.

IA de Déduction

Une IA doit être réalisée pour analyser les indices collectés et effectuer des déductions logiques. Contrairement à l'IA d'assistance, qui guide le joueur, l'IA de déduction se concentre sur la formulation d'hypothèses solides et la réduction des suspects en fonction des informations disponibles.

- ❖ **Formulation d'hypothèses** : L'IA analysera les indices collectés pour établir des hypothèses sur les personnages innocents ou suspects. Par exemple, si un personnage est présent dans une pièce à un moment où le meurtre a eu lieu ailleurs, il pourra être écarté.

❖ **Réalisation de l'IA**

On va tester deux possibilités d'IA et voir à la fin ce qui est le plus rapide :

- **IA de Déduction Heuristique** : Cette version de l'IA repose sur des règles prédéfinies et un raisonnement heuristique pour effectuer des déductions simples et rapides. L'objectif est de simuler le raisonnement humain en appliquant des algorithmes basés sur des logiques intuitives (par exemple, éliminer un suspect s'il est dans un lieu différent de celui du crime).
- **IA de Déduction Choco-Solver** : En utilisant Choco-Solver, cette IA modélisera les indices sous forme de contraintes logiques (comme les relations entre personnages, lieux et temps). Grâce à la propagation de contraintes, elle pourra éliminer

rapidement les hypothèses non valides et proposer des déductions optimisées. Par exemple, elle pourra confirmer le coupable en tenant compte de plusieurs indices simultanément, tout en respectant les règles du jeu.

IA d'Assistance

Une IA doit être réalisée afin de guider le joueur dans son processus de réflexion, en fournissant des suggestions basées sur les indices collectés. Cette assistance ré utilisera l'IA de Déduction et se traduira par :

- ❖ **Interprétation des indices** : L'IA sera capable d'analyser les indices disponibles pour suggérer des actions au joueur afin de le rapprocher de la victoire.
- ❖ **Innocenter et déduire** : En fonction des indices disponibles, l'IA devra être en mesure de proposer des hypothèses pour écarter certains personnages ou, au contraire, les suspecter davantage. Par exemple, si on découvre qu'à un moment la personne tuée a été vue toute seule avec quelqu'un dans une même pièce alors il faudra concentrer les déductions de l'IA aux alentours de cette pièce, pour des temps adjacents à celui du crime.

❖ **Réalisation de l'IA**

On va tester deux possibilités d'IA et voir à la fin ce qui est le plus rapide :

- **IA d'Assistance Heuristique** : Cette version de l'IA repose sur une approche heuristique, afin de simuler le raisonnement humain. Cette IA est axée sur la découverte et l'expérimentation, afin d'assurer la fiabilité des algorithmes.
- **IA d'Assistance Choco-Solver** : Elle utilise une bibliothèque de programmation par contraintes, Choco-Solver, qui permettra de modéliser les contraintes et déterminer les actions les plus pertinentes. L'IA pourra ainsi analyser les indices en fonction de plusieurs variables (lieu, moment, personnages).

IA Joueuse (Optionnel)

Un objectif optionnel qui sera réalisé si le temps nous le permet et qui consiste à développer une version autonome de l'IA qui pourrait jouer seule et résoudre une enquête sans intervention humaine. Elle sera différente de l'IA assistante mais on verra qu'il est possible de s'en servir. Cette IA joueuse:

- ❖ **Collecte d'indices pertinents** : L'IA joueuse se concentrera sur les indices qui feront le plus avancer l'enquête. Pour ce faire, elle analysera en priorité les indices qui permettent de réduire efficacement l'espace des suspects ou de confirmer des hypothèses. Elle exclura les questions redondantes ou peu utiles en fonction de l'état actuel de l'enquête. Par exemple, si un indice a déjà permis d'innocenter un personnage, l'IA évitera de poser des questions supplémentaires sur ce même personnage et se concentrera sur les suspects restants.

- ❖ **Résolution optimisée** : L'IA pourra déterminer le coupable de manière optimisée en limitant au maximum le nombre d'étapes nécessaires pour aboutir à une conclusion. Cette optimisation repose sur l'utilisation de notre IA assistante afin de réduire les hypothèses non valides et prioriser les actions les plus prometteuses en fonction des indices disponibles, minimisant ainsi le temps de résolution par rapport à un raisonnement exhaustif.

Génération Automatique de Nouvelles Enquêtes (Optionnel)

Pour enrichir le gameplay, le système pourra inclure une fonction de génération d'enquêtes qui crée des scénarios variés en termes de difficulté et de complexité.

- ❖ **Création de scénarios solvables** : La génération de scénarios doit s'assurer qu'il est possible de résoudre chaque enquête à partir des indices fournis. L'IA joueuse devra donc tester cette énigme pour vérifier si elle est réalisable.

B. Contraintes et Limitations

Optimisation du Temps de Calcul

L'IA doit être optimisée pour offrir une réponse rapide aux joueurs, même dans les scénarios les plus complexes. Ceci implique :

- ❖ **Réduction du temps de déduction** : L'IA doit idéalement résoudre chaque étape de déduction rapidement, afin de ne pas ralentir l'expérience de jeu. Pour y parvenir, nous allons utiliser nos IA heuristiques et Choco-Solver pour tenter de prioriser les actions les plus pertinentes. Cela pourrait permettre d'éliminer rapidement les possibilités moins probables et de concentrer les efforts sur les hypothèses les plus plausibles. Cependant, nous ne sommes pas sûrs de pouvoir garantir une optimisation complète du temps de calcul, car cela dépend des contraintes en jeu.
- ❖ **Exclusion d'options impossibles** : Pour améliorer la rapidité, on exclura dès le début les scénarios non viables, réduisant ainsi le nombre de calculs nécessaires. Cela sera réalisé par l'utilisation de filtres basés sur les règles du jeu et des contraintes logiques. Par exemple, si un personnage n'a jamais été vu dans un lieu donné selon les indices, il sera immédiatement écarté des hypothèses. De plus, l'IA pourra utiliser des algorithmes de propagation de contraintes pour éliminer rapidement les combinaisons impossibles de personnages, lieux et temps. Cela permettra d'améliorer l'efficacité de la recherche, en réduisant l'espace de solution et en accélérant le processus de déduction.

Traitement d'Indices Partiels et Multiples Sources d'Informations

L'IA sera confrontée à des indices provenant de diverses sources (lieux, personnages, objets) et devra être capable de croiser ces informations pour obtenir des déductions fiables.

- ❖ **Gestion des données incomplètes** : L'IA doit pouvoir interpréter des informations incomplètes et anticiper les conclusions possibles en fonction des éléments obtenus.
- ❖ **Croisement d'informations** : En utilisant la logique, l'IA croisera les indices pour isoler des relations et des hypothèses valides, tout en éliminant les pistes qui ne concordent pas.

Respect des Règles du Jeu Kronologic

L'IA doit se conformer rigoureusement aux règles du jeu Kronologic, sans recourir à des actions ou des stratégies non prévues par les règles.

- ❖ **Suivi strict des règles** : Chaque décision et chaque action de l'IA doivent respecter les limitations du jeu en termes de collecte d'indices et de vérification des hypothèses. Par exemple, l'IA ne doit pas poser des questions ou suggérer des actions qui ne sont pas autorisées par les règles du jeu. Elle doit également s'assurer que les déductions et les hypothèses qu'elle fait sont basées sur des éléments disponibles dans l'état actuel du jeu.

Densité de Personnages en Fonction des Salles

Pour assurer une expérience de jeu réaliste et cohérente, l'algorithme de création d'éénigmes doit gérer la densité des personnages présents dans chaque salle. On suppose que limiter le nombre de personnages dans un même espace pourrait influer sur la difficulté de l'éénigme, mais cela reste à tester.

- ❖ **Priorité aux règles de gameplay** : Lorsqu'un personnage tente de rejoindre une salle possédant le nombre de personnage maximal, l'algorithme devra rediriger ce personnage vers un autre emplacement. Ces choix devront être faits sans perturber l'avancement logique de l'enquête.
- ❖ **Test avec l'IA joueuse** : Afin de vérifier la bonne jouabilité, on utilisera l'IA joueuse pour vérifier que l'enquête est bien solvable.

C. Outils et Technologies

- ❖ **Langage de programmation -> Java**
- ❖ **Bibliothèque de programmation par contraintes -> Choco-Solver**
- **Choco-Solver** : Bibliothèque de programmation par contraintes utilisée pour résoudre des problèmes complexes en modélisant les variables, leurs domaines de valeurs possibles, et les contraintes qui restreignent leurs relations. Son moteur de propagation de contraintes et de recherche systématique permet d'explorer efficacement les solutions valides tout en excluant les options impossibles.

Choco-Solver est particulièrement adapté aux problèmes comme la planification, les puzzles logiques et les jeux, où les règles et les restrictions doivent être rigoureusement respectées.

- **Programmation par Contraintes :** La programmation par contraintes est une méthode algorithmique utilisée pour résoudre des problèmes en définissant :
 1. **Les variables :** Ce sont les éléments inconnus que nous cherchons à déterminer. Par exemple, dans le jeu Kronologic, les variables peuvent inclure la position des personnages à chaque moment ou encore la salle où le meurtre a eu lieu.
 2. **Les domaines :** Chaque variable possède un domaine de valeurs possibles qu'elle peut prendre. Ces valeurs définissent les choix potentiels pour la solution. Par exemple, pour une variable représentant la position d'un personnage, le domaine pourrait être $\{1,2,3,4,5,6\}$, correspondant aux 6 salles du jeu.
 3. **Les contraintes :** Les contraintes sont des restrictions logiques ou mathématiques que les solutions doivent respecter. Elles permettent d'éliminer les combinaisons de valeurs impossibles. Par exemple, une contrainte peut imposer qu'un personnage ne peut pas rester dans la même salle pendant deux pas de temps consécutifs.

- **Contrainte :** Une contrainte est une règle ou une relation entre plusieurs variables qui doivent être satisfaites simultanément. Par exemple :
 - Une variable représentant un personnage doit être associée à une salle à chaque moment du jeu.
 - Un personnage ne peut pas être dans deux salles différentes en même temps.
 - Les salles adjacentes doivent être respectées lors des déplacements.
- **Modélisation mathématique des contraintes avec Choco-Solver :**

1. Variables :

- $P_{i,t}$: Position du personnage i au temps t où $P_{i,t} \in \{1,2,\dots,6\}$ (les 6 salles).
- S_t : Salle dans laquelle se trouve le détective au temps t , où $S_t \in \{1,2,\dots,6\}$.
- C_t : Salle où le meurtre a eu lieu au temps t , où $C_t \in \{1,2,\dots,6\}$.
- T_{crime} : Temps exact où le meurtre a eu lieu.

2. Contraintes :

Contraintes de position et de déplacement des personnages

- **Position unique à chaque instant :**

Un personnage doit être dans une seule salle à chaque temps :

$$\forall i,t : P_{i,t} \in \{1,2,\dots,6\}$$

- **Déplacement obligatoire :**

Chaque personnage doit se déplacer entre t et $t+1$:

$$\forall i,t : P_{i,t} \neq P_{i,t+1}$$

- **Déplacement dans des salles adjacentes :**

Les déplacements doivent respecter la connectivité des salles. Par exemple, si la salle 1 est reliée à 2 et 3, alors :

$$\forall i,t : P_{i,t+1} \in \text{adj}(P_{i,t})$$

Contraintes sur le détective et le coupable

- **Présence du détective et du coupable :**

Si un personnage i est le seul avec le détective dans une salle S_t à un temps donné, alors ce personnage est désigné comme coupable :

$$\forall t : \text{Nbpersonnage dans } S_t = 2 \Rightarrow \text{Coupable} = P_{i,t}$$

- **Isolément au moment du crime :**

Si le meurtre a lieu au temps T_{crime} , le coupable et le détective doivent être seuls dans la salle où le meurtre se déroule :

$$\sum_i \delta(P_{i,T_{\text{crime}}}, C_{T_{\text{crime}}}) = 2$$

où $\delta(x,y)=1$ si $x=y$, sinon 0.

4. Liste des fonctionnalités

A. Fonctionnalités liées à l'interface utilisateur (Vue Carte)

❖ Créer une page d'accueil

➤ Condition de validation :

- La page permet de démarrer une nouvelle partie (bouton jouer : ouverture de l'interface de jeu), de tester l'IA joueuse ou de quitter l'application.

❖ Création de l'interface de jeu

➤ Condition de validation :

- La page affiche :
 - Tous les éléments visuels (les 6 pas de temps avec les cartes),
 - Les pions de déductions (personnages et nombre de personnage),
 - Les checkbox de modification de fonction des pions (hypothèse et absence),
 - L'historique des indices dévoilés,
 - Les différents boutons de jeu fonctionnels (poser une question, déduction de l'IA, faire une déduction, demander un indice),
 - Les boutons d'affichage des présences et des absences (affichent des pions représentant les présences et/ou les absences selon les informations récoltées avec les questions) ainsi que le bouton d'affichage des hypothèses (coché : affiche les hypothèses du joueur, décocher : désaffiche les hypothèses du joueur),
 - Les boutons d'affichage des films des déplacements des personnages de la partie et du joueur,
 - La flèche de retour en arrière,
 - Les règles du jeu,
 - Changer d'affichage.

❖ Permettre au joueur de poser une question

➤ Condition de validation :

- Le joueur est redirigé sur une page lui permettant de sélectionner un lieu, et un temps ou un personnage.
- Dans le cas où il a choisi un temps, on affiche l'indice public (nombre de personnages dans le lieu au temps choisi) et l'indice privé (rejouer ou identité d'un personnage se trouvant dans le lieu au temps choisi).
- Dans le cas où il a choisi un personnage, on affiche l'indice public (nombre de fois où le personnage choisi s'est trouvé dans le lieu) et l'indice privé (rejouer ou un temps où il s'y trouvait).
- Les indices récoltées seront, selon la volonté du joueur, placées automatiquement sur les cartes à l'aide des pions de déduction ou alors le joueur devra les placer lui-même.

❖ Permettre au joueur de formuler des hypothèses

➤ Condition de validation :

- Le joueur peut placer les pions de déduction dans chaque lieux aux différents pas de temps sur la feuille de jeu en fonction des indices qu'il connaît et des déductions qu'il réalise.
- Le joueur peut modifier le rôle de ces pions en cochant les checkbox de modification.
 - La checkbox **hypothèse** transforme tous les pions en pion "hypothèse" (permet au joueur de déduire la position éventuelle d'un personnage sans pouvoir l'affirmer),
 - La checkbox **absence** transforme tous les pions en pion d'absence (permet au joueur d'affirmer l'absence d'un personnage dans un lieu pour un temps donné).
- Les deux checkbox interagissent entre elles : Si les deux sont activés, les pions sont des pions d'hypothèse d'absence et si les deux sont désactivés, alors les pions sont des pions de déduction.

❖ **Permettre au joueur de formuler des déductions**

➤ Condition de validation :

- Le joueur peut, à tout moment, appuyer sur le bouton "Faire une déduction". Ainsi, il choisit un coupable, un lieu, et un moment, puis valide, ou non, sa déduction. Le jeu s'arrête et informe le joueur de sa victoire ou de sa défaite.

❖ **Permettre au joueur d'accéder aux déductions de l'IA déductrice**

➤ Condition de validation :

- Le joueur peut, à tout moment, appuyer sur le bouton "Déduction de l'IA". Une fenêtre va s'ouvrir et va afficher l'historique des déductions de l'IA, du plus récent au plus ancien.

❖ **Permettre au joueur de demander de l'aide à l'IA assistante**

➤ Condition de validation :

- Le joueur peut, à tout moment, appuyer sur le bouton "Demander un indice". Après validation, l'IA pose la meilleure question et le joueur récupère les indices.

❖ **Permettre au joueur de changer d'affichage**

➤ Condition de validation :

- Un bouton permet de passer à un deuxième type d'affichage sous forme de tableaux. Les fonctionnalités de ce deuxième affichage seront développées ci-après.

❖ **Permettre au joueur de visualiser les règles**

➤ Condition de validation :

- Un bouton permet au joueur d'afficher les règles du jeu ainsi que le rôle de chaque bouton.

❖ **Permettre au joueur de visualiser le film des déplacements des personnages**

➤ Condition de validation :

- Un bouton permet d'afficher le trajet que les personnages ont suivis, conformément au scénario de l'enquête.
 - Un bouton permet d'afficher le trajet de chacun des personnages selon les hypothèses du joueur.
-

B. Fonctionnalités liées à l'interface utilisateur (Vue Tableaux)

❖ Création de l'interface de jeu

➤ Condition de validation :

- La page affiche :
 - Tous les éléments visuels (les 2 groupes de 6 pas de temps avec les groupes de pion de personnage et de pion de lieu),
 - L'historique des indices dévoilés,
 - Les différents boutons de jeu fonctionnels (poser une question, déduction de l'IA, faire une déduction, demander un indice),
 - Les deux tableaux de déduction :
 - ◆ Le premier tableau permet de renseigner combien de personnages sont présents dans tel lieu à tel moment.
 - ◆ Le deuxième tableau permet au joueur d'indiquer dans quelle salle se trouve, ou ne se trouve pas, chaque personnage aux différents pas de temps.
 - Les boutons d'affichage des films des déplacements des personnages de la partie et du joueur,
 - La flèche de retour en arrière,
 - Les règles du jeu,
 - Changer d'affichage.

❖ Permettre au joueur de poser une question

➤ Condition de validation :

- Comme pour la Vue Carte.

❖ Permettre au joueur de formuler des hypothèses

➤ Condition de validation :

- Le joueur peut reporter dans les tableaux les indices qu'il a découverts.
- Il peut aussi effectuer des hypothèses dans les tableaux.

❖ Permettre au joueur de formuler des déductions

➤ Condition de validation :

- Comme pour la Vue Carte.

❖ Permettre au joueur d'accéder aux déductions de l'IA déductrice

➤ Condition de validation :

- Comme pour la Vue Carte.

❖ Permettre au joueur de demander de l'aide à l'IA assistante

➤ Condition de validation :

- *Comme pour la Vue Carte.*
 - ❖ **Permettre au joueur de changer d'affichage**
 - Condition de validation :
 - Un bouton permet de repasser au type d'affichage en cartes. Les hypothèses et les déductions du joueur seront reportées sur la nouvelle vue.
 - ❖ **Permettre au joueur de visualiser les règles**
 - Condition de validation :
 - *Comme pour la Vue Carte.*
 - ❖ **Permettre au joueur de visualiser le film des déplacements des personnages**
 - Condition de validation :
 - *Comme pour la Vue Carte.*
-

C. Fonctionnalités liées à l'IA de déduction

On disposera de deux IA de déduction : une heuristique et une utilisant Choco-Solver. Bien qu'ayant des fonctionnements différents, elles devront posséder les mêmes fonctionnalités.

- ❖ **Analyser les indices collectés et en tirer des conclusions logiques**
 - Condition de validation :
 - L'IA croise les indices disponibles et déduit des informations supplémentaires non explicites avec les indices. Ces conclusions sont affichées dans un historique de déduction, lorsque le joueur le demande (pression du bouton "Déduction de l'IA").
 - Les suspects impossibles sont exclus en temps réel après chaque nouvel indice.
 - ❖ **Proposer une hypothèse logique basée sur les indices**
 - Condition de validation :
 - L'IA utilise les indices pour formuler une hypothèse probable (coupable, lieu, moment).
 - L'hypothèse est mise en évidence dans l'historique des déductions.
-

D. Fonctionnalités liées à l'IA d'assistance

On disposera également de deux IA d'assistance, une heuristique et une utilisant Choco-Solver. Ayant le même objectif, elles possèderont les mêmes fonctionnalités.

- ❖ **Recommander une question optimale à poser**
 - Condition de validation :
 - L'IA analyse les indices et suggère la meilleure question à poser pour maximiser les informations obtenues. La recommandation est visible.

❖ **Simuler les conséquences de différentes questions**

➤ Condition de validation :

- L'IA évalue plusieurs scénarios potentiels basés sur des questions possibles et propose l'option faisant avancer le plus l'enquête.

❖ **Corriger les déductions du joueur**

➤ Condition de validation :

- L'IA compare ses déductions avec celles du joueur et le corrige si elle remarque des erreurs dans ses hypothèses.
-

Partie optionnelle : Toutes les fonctionnalités qui suivent seront développées uniquement si nous avons le temps de terminer l'ensemble des parties A à C avant la fin du projet.

E. Fonctionnalités liées à l'IA joueuse

Cette IA se servira des IA de déduction et d'assistance afin de fonctionner.

- ❖ Raisonner sur les indices collectés et en tirer des déductions logiques.
 - ❖ Poser les questions permettant d'obtenir les informations les plus pertinentes pour la complétion de l'enquête.
 - ❖ Générer des hypothèses intermédiaires qui seront ajustées en fonction des nouveaux indices.
 - ❖ Éliminer les options impossibles au fur et à mesure afin de pouvoir raisonner sur un ensemble d'informations plus petit.
 - ❖ Formuler une déduction une fois qu'elle a trouvé le coupable, le temps et le lieu du crime.
-

F. Fonctionnalités liées au générateur de scénarios

- ❖ Permettre au joueur de choisir entre un scénario prédéfini ou un nouveau scénario généré aléatoirement.
- ❖ Permettre au joueur de choisir la difficulté du scénario à générer.
- ❖ Tester la faisabilité du scénario grâce à l'IA joueuse avant de le proposer au joueur (une seule solution possible et trouvable).

5. Cas d'utilisation - Diagrammes de CU - Scénarios - Conditions de validation

Cas d'utilisation : Créer une page d'accueil

Acteur principal : Joueur.

Objectifs : Permettre au joueur de démarrer une nouvelle partie ou de quitter l'application.

Préconditions : L'application est lancée et affiche la page d'accueil.

Postconditions : Le joueur peut accéder à l'interface de jeu ou quitter l'application.

Scénario principal :

1. Le joueur ouvre l'application.
 2. La page d'accueil s'affiche, avec deux options : *Jouer* et *Quitter*.
 3. Le joueur sélectionne *Jouer*.
 4. L'interface de jeu s'ouvre, et la partie est initialisée.
 5. Alternativement, le joueur sélectionne *Quitter*, et l'application se ferme.
-

Cas d'utilisation : Créer l'interface de jeu

Acteur principal : Joueur.

Objectifs : Générer une interface de jeu fonctionnelle, qu'elle soit sous forme de carte ou de tableaux, pour permettre au joueur d'interagir avec le jeu.

Préconditions : Le joueur a démarré une nouvelle partie ou a basculé entre les affichages.

Postconditions : Tous les éléments nécessaires au jeu sont affichés selon le type de vue sélectionné.

Scénario principal :

1. L'interface est générée selon le type d'affichage choisi :
 - **Vue Carte :**
 - Les 6 pas de temps sont affichés sous forme de cartes.
 - Les pions de déduction et leurs options (hypothèses, absences) sont disponibles.
 - Les boutons fonctionnels (poser une question, faire une déduction, demander un indice, etc.) sont visibles et opérationnels.
 - L'historique des indices dévoilés est affiché.
 - **Vue Tableaux :**
 - Deux tableaux principaux permettent de renseigner les informations récupérées sur les lieux et les présences des personnages. Le joueur peut effectuer des hypothèses.
 - Les boutons fonctionnels identiques à la Vue Carte sont affichés.
 - L'historique des indices dévoilés est affiché.
2. Le joueur interagit avec les éléments spécifiques à l'affichage pour avancer dans la partie.

Conditions de validation :

- Tous les éléments visuels sont affichés selon le type de vue.
-

Cas d'utilisation : Permettre au joueur de poser une question

Acteur principal : Joueur.

Objectifs : Permettre au joueur de collecter des indices en posant des questions.

Préconditions : Le joueur est en cours de partie et sur l'interface de jeu.

Postconditions : Les indices récoltés sont ajoutés à l'historique et affichés.

Scénario principal :

1. Le joueur appuie sur le bouton *Poser une question*.
2. Une fenêtre s'ouvre, proposant deux options : *Choisir un lieu et un temps* ou *Choisir un lieu et un personnage*.
3. Si le joueur choisit *Lieu et Temps* :
 - L'indice public (nombre de personnages présents) est affiché.
 - L'indice privé (identité d'un personnage ou rejouer) est révélé.
4. Si le joueur choisit *Lieu et Personnage* :
 - L'indice public (nombre de passages) est affiché.
 - L'indice privé (temps d'un passage ou rejouer) est révélé.
5. Les indices collectés sont placés soit par le joueur soit automatiquement sur les cartes.

Conditions de validation :

- Les indices publics et privés sont correctement affichés selon la question posée.
 - Les indices sont intégrés automatiquement dans l'interface de jeu.
-

Cas d'utilisation : Permettre au joueur de formuler des hypothèses

Acteur principal : Joueur.

Objectifs : Offrir au joueur la possibilité de formuler et d'enregistrer des hypothèses sur les positions des personnages, en fonction de la vue choisie.

Préconditions : Le joueur a récolté des indices et souhaite les exploiter pour formuler des hypothèses.

Postconditions : Les hypothèses sont enregistrées et visibles dans la vue carte uniquement.

Scénario principal :

1. Le joueur utilise l'interface pour formuler ses hypothèses :
 - **Vue Carte :**
 - Le joueur place des pions sur les lieux des cartes.
 - Les pions peuvent être configurés en hypothèses, absences, ou hypothèses d'absence à l'aide des cases à cocher disponibles.
 - **Vue Tableaux :**
 - Le joueur renseigne directement les indices dans les tableaux.
 - Il indique les présences et le nombre de personnages par lieu et par pas de temps.
2. Les hypothèses sont enregistrées dans le système.
3. Les hypothèses peuvent être modifiées ou supprimées par le joueur.

Conditions de validation :

- Les hypothèses enregistrées sont visibles dans les deux types de vue.

Cas d'utilisation : Permettre au joueur de formuler des déductions

Acteur principal : Joueur.

Objectifs : Donner la possibilité au joueur de résoudre l'enquête en déclarant un coupable, un lieu et un moment.

Préconditions : Le joueur dispose d'indices suffisants pour effectuer une déduction.

Postconditions : La partie se termine par une victoire ou une défaite.

Scénario principal :

1. Le joueur appuie sur le bouton *Faire une déduction*.
2. Une fenêtre s'ouvre, permettant au joueur de sélectionner :
 - Un coupable.
 - Un lieu du meurtre.
 - Le moment du crime.
3. Le joueur valide sa déduction.
4. Le système compare la déduction du joueur avec la solution de l'enquête.
5. Le joueur est informé de sa victoire (si la déduction est correcte) ou de sa défaite (si elle est incorrecte).

Conditions de validation :

- La fenêtre de déduction s'affiche correctement.
 - Le système vérifie et compare les choix du joueur aux données réelles de l'enquête.
-

Cas d'utilisation : Permettre au joueur d'accéder aux déductions de l'IA déductrice

Acteur principal : Joueur.

Objectifs : Donner la possibilité au joueur de consulter les déductions effectuées par l'IA déductrice, pour orienter ses propres décisions.

Préconditions : L'IA déductrice est active et a déjà effectué des déductions basées sur les indices disponibles.

Postconditions : L'historique des déductions est affiché, du plus récent au plus ancien.

Scénario principal :

1. Le joueur appuie sur le bouton *Déduction de l'IA*.
2. Une fenêtre s'ouvre, affichant les déductions réalisées par l'IA.
3. Les déductions sont listées chronologiquement, avec des détails. Par exemple :
 - Personnage : Aventurière
 - Lieu : La Salle
 - Temps : 6
 - Statut : NON (confirmant que l'Aventurière ne se trouvait pas dans La Salle au Temps 6).
4. Le joueur peut fermer la fenêtre pour revenir à l'interface principale.

Conditions de validation :

- Les déductions affichées sont cohérentes avec les indices et hypothèses disponibles.
-

Cas d'utilisation : Permettre au joueur de demander de l'aide à l'IA assistante

Acteur principal : Joueur.

Objectifs : Faciliter le raisonnement du joueur en proposant des actions pertinentes.

Préconditions : L'IA assistante est active, et le joueur a récolté des indices.

Postconditions : Le joueur obtient un nouvel indice.

Scénario principal :

1. Le joueur appuie sur le bouton *Demandeur un indice*.
2. L'IA analyse les indices disponibles et détermine la meilleure question à poser.
3. L'indice obtenu est révélé au joueur.
4. L'indice est ajouté à l'historique et manuellement ou automatiquement (selon la volonté du joueur) intégré à la feuille de jeu.

Conditions de validation :

- L'IA pose une question pertinente et cohérente avec l'état actuel de l'enquête.
 - L'indice est correctement affiché et enregistré.
-

Cas d'utilisation : Permettre au joueur de changer d'affichage

Acteur principal : Joueur.

Objectifs : Permettre au joueur de basculer entre la Vue Carte et la Vue Tableaux.

Préconditions : Une partie est en cours et le joueur souhaite changer d'affichage.

Postconditions : L'affichage change et les informations saisies sont conservées dans les deux vues.

Scénario principal :

1. Le joueur appuie sur le bouton *Changer d'affichage*.
2. Si la Vue Carte est affichée, elle est remplacée par la Vue Tableaux.
3. Si la Vue Tableaux est affichée, elle est remplacée par la Vue Carte.
4. Les hypothèses, les indices et les déductions enregistrés sont reportés dans la nouvelle vue.

Conditions de validation :

- Le changement d'affichage n'a pas de perte de données.
-

Cas d'utilisation : Permettre au joueur de visualiser les règles

Acteur principal : Joueur.

Objectifs : Offrir un accès rapide et clair aux règles du jeu pour aider le joueur à comprendre les fonctionnalités qui lui sont proposées.

Préconditions : Une partie est en cours et le joueur souhaite consulter les règles.

Postconditions : Les règles sont affichées et lisibles.

Scénario principal :

1. Le joueur appuie sur le bouton *Règles du jeu*.
 2. Une fenêtre ou une page dédiée s'ouvre, affichant les règles complètes.
 3. Les règles incluent les explications sur :
 - Les déplacements des personnages.
 - Les indices publics et privés.
 - Les conditions de victoire.
 - L'utilisation des pions et hypothèses.
 4. Le joueur peut revenir à l'interface principale.
-

Cas d'utilisation : Permettre au joueur de visualiser les films des déplacements des personnages

Acteur principal : Joueur.

Objectifs : Montrer les trajets suivis par les personnages, selon les scénarios et les hypothèses du joueur.

Préconditions : Une partie est en cours et le joueur a formulé des hypothèses et renseigné des informations.

Postconditions : Les trajets des personnages sont affichés selon le choix du joueur.

Scénario principal :

1. Soit le joueur appuie sur le bouton *Film des déplacements du Joueur* :
 - Afficher les déplacements hypothétiques selon les pions placés par le joueur.
2. Soit le joueur appuie sur le bouton *Film des déplacements du Partie* :
 - Afficher les déplacements réels des personnages selon le scénario.
3. Le film des déplacements s'affiche à l'écran.
4. Le joueur peut repasser ou quitter le film à tout moment.

Conditions de validation :

- Les trajets affichés correspondent aux informations du joueur ou de la partie.
-

Cas d'utilisation : Analyser les indices collectés et en tirer des conclusions logiques

Acteur principal : IA de déduction (heuristique et Choco-Solver).

Objectifs : Exploiter les indices disponibles pour produire des déductions logiques supplémentaires et tenir à jour un historique des déductions.

Préconditions : Des indices ont été collectés par le joueur ou par l'IA.

Postconditions : Les informations déduites sont visibles dans un historique de déduction et les suspects impossibles sont exclus en temps réel.

Scénario principal :

1. L'IA examine les indices collectés.
2. Elle croise les données pour identifier des relations logiques non explicites (par exemple, un personnage innocenté en fonction de sa localisation).

3. Les suspects impossibles sont automatiquement écartés des hypothèses.
4. Les déductions supplémentaires sont enregistrées et mises à jour dans l'historique.
5. Le joueur peut consulter l'historique en appuyant sur le bouton *Déduction de l'IA*.

Conditions de validation :

- Les déductions de l'IA sont cohérentes avec les indices collectés.
 - Les suspects exclus sont mis à jour en temps réel et ne figurent plus dans les hypothèses.
 - L'historique des déductions est accessible par le joueur.
-

Cas d'utilisation : Proposer une hypothèse logique basée sur les indices

Acteur principal : IA de déduction (heuristique et Choco-Solver).

Objectifs : Formuler une hypothèse probable concernant le coupable, le lieu et le moment du crime.

Préconditions : Des indices suffisants ont été collectés pour permettre à l'IA de formuler une hypothèse.

Postconditions : Une hypothèse logique est générée et affichée dans l'historique des déductions.

Scénario principal :

1. L'IA examine les indices disponibles pour identifier les suspects potentiels, le lieu probable du crime et l'heure associée.
2. Elle génère une hypothèse basée sur une analyse logique des données.
3. Cette hypothèse est mise en avant dans l'historique des déductions.
4. Le joueur peut consulter cette hypothèse en appuyant sur le bouton *Déduction de l'IA*.

Conditions de validation :

- L'hypothèse formulée est logique avec les indices disponibles.
 - L'hypothèse est correctement affichée et identifiable dans l'historique des déductions.
-

Cas d'utilisation : Recommander une question optimale à poser

Acteur principal : IA d'assistance (heuristique et Choco-Solver).

Objectifs : Suggérer au joueur la meilleure question à poser pour maximiser les informations obtenues et faire avancer l'enquête.

Préconditions : Le joueur a collecté des indices, mais l'enquête n'est pas encore résolue.

Postconditions : Une recommandation claire est affichée pour guider le joueur.

Scénario principal :

1. L'IA analyse les indices collectés et évalue les informations manquantes.
2. Elle identifie la question qui a le plus de chances d'apporter des données pertinentes.
3. La recommandation est affichée dans l'interface, visible par le joueur.
4. Le joueur peut choisir de poser la question suggérée ou d'ignorer la recommandation.

Conditions de validation :

- Les suggestions de l'IA sont pertinentes et logiques par rapport à l'état actuel de l'enquête.
- Les recommandations sont facilement compréhensibles pour le joueur.

Cas d'utilisation : Simuler les conséquences de différentes questions

Acteur principal : IA d'assistance (heuristique et Choco-Solver).

Objectifs : Évaluer plusieurs scénarios potentiels en fonction des questions possibles et proposer celle qui avance le plus l'enquête.

Préconditions : Une liste de questions possibles est définie et des indices ont été collectés.

Postconditions : L'IA propose l'option qui a le plus grand impact sur l'avancement de l'enquête.

Scénario principal :

1. L'IA examine les indices collectés et les lacunes restantes.
2. Elle simule les scénarios possibles pour chaque question envisageable.
3. Elle calcule l'impact potentiel de chaque option sur l'avancement de l'enquête.
4. Elle propose la question ayant l'effet le plus significatif.
5. La recommandation est affichée dans l'interface et le joueur peut choisir de suivre ou non l'avis de l'IA.

Conditions de validation :

- Les scénarios simulés par l'IA sont pertinents et basés sur les indices disponibles.
 - La suggestion finale reflète la meilleure option pour faire progresser l'enquête.
-

Cas d'utilisation : Corriger les déductions du joueur

Acteur principal : IA d'assistance (heuristique et Choco-Solver).

Objectifs : Comparer les hypothèses du joueur avec celles de l'IA et fournir des corrections si des erreurs sont détectées.

Préconditions : Le joueur a formulé des hypothèses basées sur les indices disponibles.

Postconditions : Les corrections de l'IA sont affichées et le joueur peut ajuster ses hypothèses.

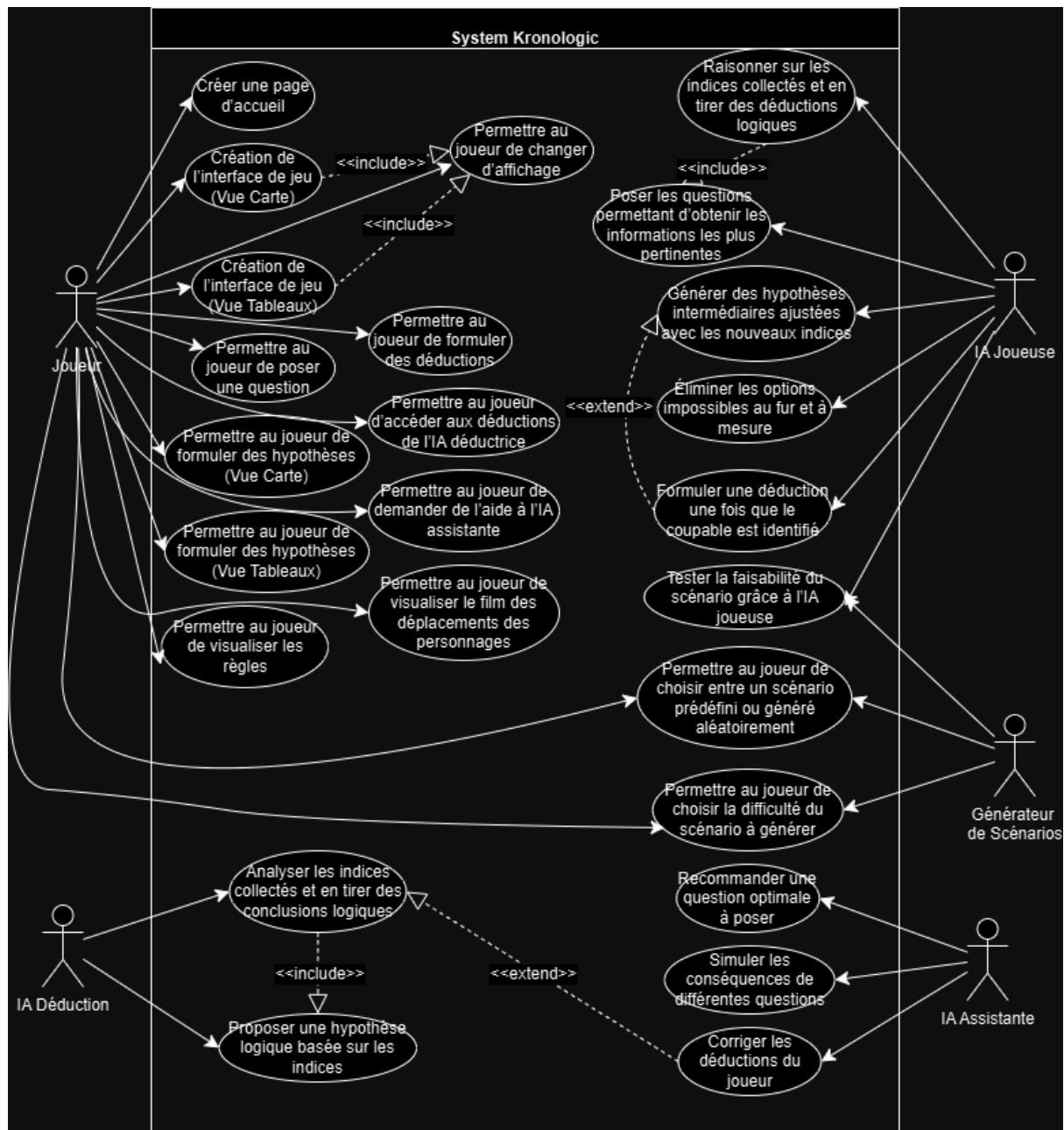
Scénario principal :

1. Le joueur formule ses hypothèses et les enregistre.
2. L'IA compare ces hypothèses avec ses propres déductions.
3. Si des erreurs sont détectées (par exemple, un suspect incorrect ou un personnage mal placé à un pas de temps), l'IA affiche les corrections nécessaires.
4. Les suggestions de l'IA sont présentées clairement, permettant au joueur d'ajuster ses hypothèses.

Conditions de validation :

- Les corrections de l'IA sont pertinentes et logiques par rapport aux indices disponibles.
- Les suggestions sont compréhensibles et permettent au joueur de rectifier ses erreurs.

Diagramme de cas d'utilisation

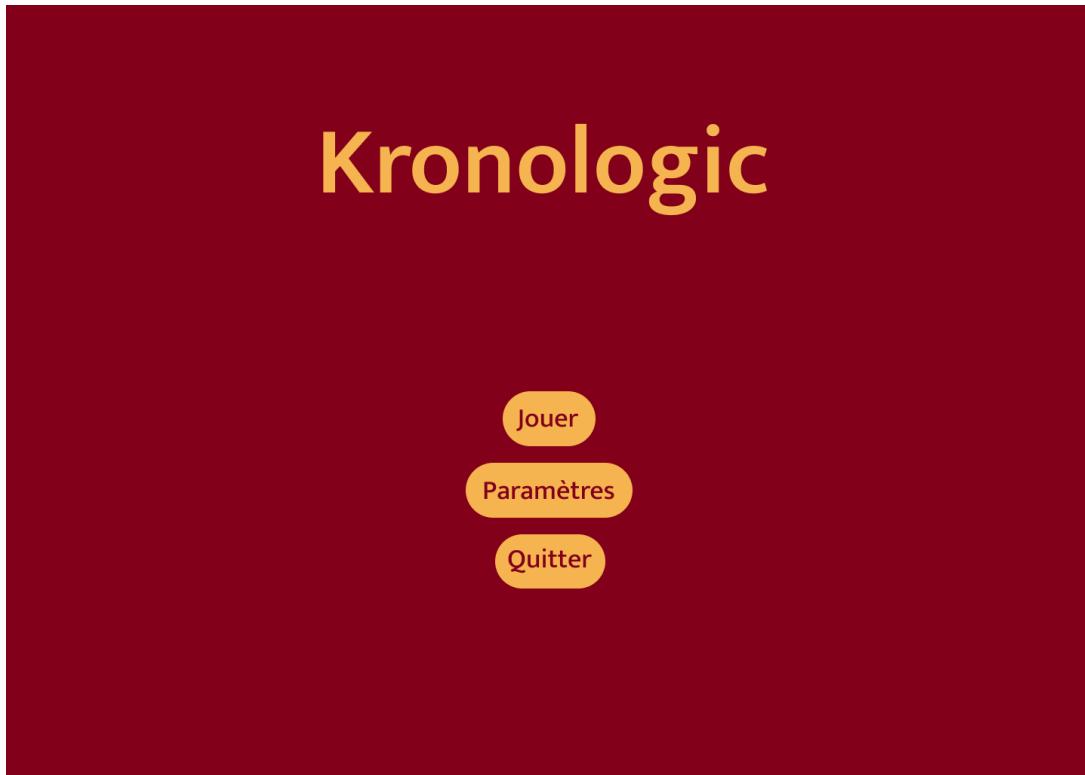


6. Diagrammes de classe, de séquences, d'activité

**Les images des différents diagrammes ont été mises en dehors
du document pour des soucis de lisibilité.**

7. Maquette de l'application

Page d'accueil



Page de jeu (Vue Carte)

The screenshot displays a 3x3 grid of six room icons from a top-down perspective. Each room contains small icons representing characters or objects. To the left of the grid is a sidebar with a back arrow, a "Historique des indices" section containing text about tour 2 and tour 1, and a "Changer affichage" button. To the right of the grid are two buttons: "Déduction de l'IA" and "Masquer les hypothèses". At the bottom of the grid are three buttons: "Faire une déduction", "Poser une question", and "Demander un indice". Above the grid, there are three filter buttons: "Afficher les présences" (checked), "joueur" (checked), "Partie" (unchecked), and "Afficher les absences" (unchecked). A legend below the grid shows colored circles for "Hypothèse" (green), "Absence" (red), and "Masquer les hypothèses" (grey checked).

Page de jeu (Vue Tableaux)

Historique des indices :

Tour 2 : Il y a 0 personne La Salle au Temps 1.

Tour 1 : L'Aventurière est passée 0 fois dans La Salle.

Écran de pose de question (cas initial)

Cas initial

Posez une question :

Choisissez un Lieu :

Choisissez un Temps ou un Personnage :

Quitter **Valider**

Écran de pose de question (sélection d'un lieu) : La Salle

Sélection du Lieu

Posez une question :

Choisissez un Lieu :



Choisissez un Temps ou un Personnage :









Quitter **Valider**

Écran de pose de question (sélection d'un temps) : Temps 1

Sélection du Temps

Posez une question :

Choisissez un Lieu :



Choisissez un Temps ou un Personnage :









Quitter **Valider**

Écran de pose de question (sélection d'un personnage) : L'Aventurière

Sélection du Personnage

Posez une question :

Choisissez un Lieu :



Choisissez un Temps ou un Personnage :



Quitter

Valider

Écran de pose de question (validation)

Voulez-vous demander “Combien de personnage sont présents au Temps 1 dans La Salle ?” ?

Annuler

Valider

Page de jeu (Vue Carte) : Affichage après pose d'une question



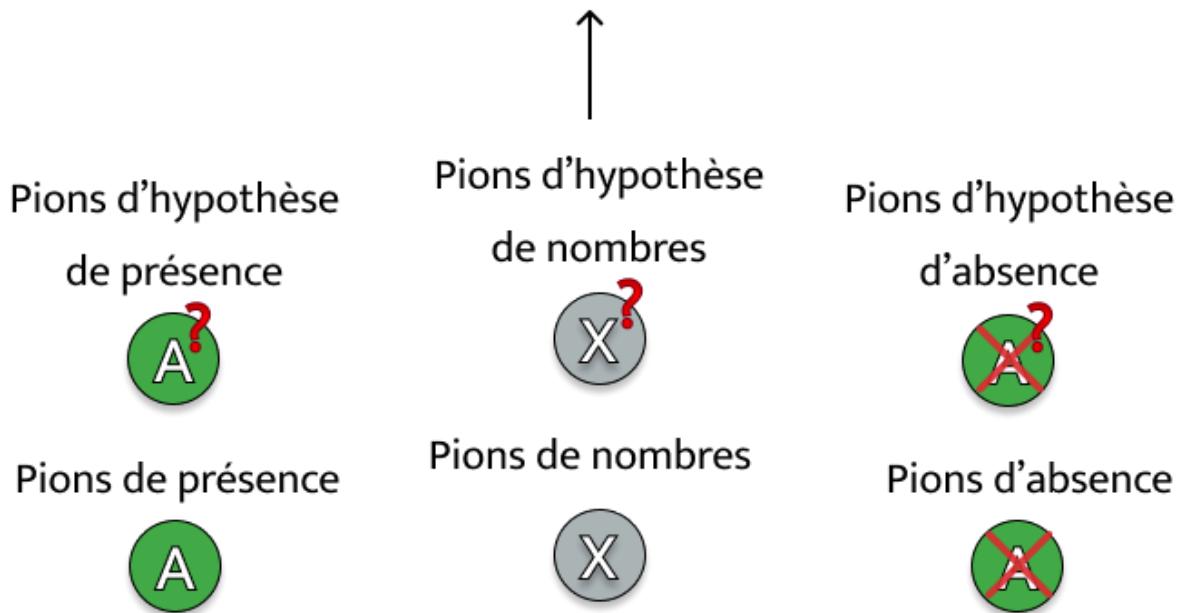
Écran de pose de question (rejouer)



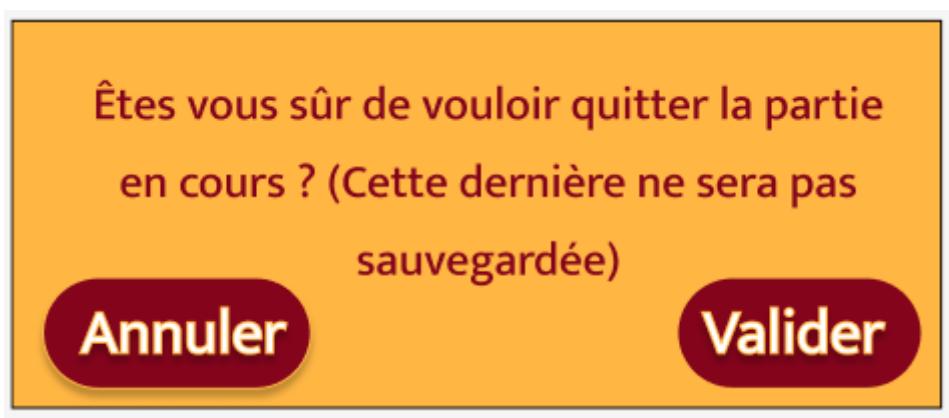
Écran de déduction(cas initial) : Même écran que pour la pose de question

Explication du fonctionnement des pions (Vue Carte)

Pour augmenter le nombre de personnage il faut double cliquer sur le pion. Cela vous permet de choisir un nombre entre 0 et 6.



Écran de validation pour quitter le jeu



Écran des règles

←

Demander un indice

Permet de demander un indice. Cela vous posera automatiquement la meilleure question à poser.

Poser une question

En posant une question, vous allez avoir accès à un indice qui va vous permettre d'avancer dans votre raisonnement.

Faire une déduction

Vous permet d'essayer de trouver la solution.
Attention, vous avez le droit à une seule déduction !

 **Règles particulières :**

- Les Personnages se déplacent obligatoirement et d'un lieu maximum.
- Les lieux sont reliés par des passages (en blanc).

Afficher les présences Afficher les absences

Affiche les informations sûres (des indices).

Déduction de l'IA

Vous permet d'afficher et de comparer vos déductions avec celle de l'IA.

Changer affichage

Vous permet de passer de l'affichage sous forme de cartes au tableau.

Écran des déductions de l'IA

←

Déduction de l'IA

Aventurière, La Salle, Temps 6 => NON
Aventurière, La Salle, Temps 5 => NON
Aventurière, La Salle, Temps 4 => NON
Aventurière, La Salle, Temps 3 => NON
Aventurière, La Salle, Temps 2 => NON

Écran de confirmation d'aide de l'IA

Êtes vous sûr de vouloir demander un indice ? (Cela comptera comme une question.)

Annuler

Valider

Écran de victoire

**Bravo ! Vous avez résolu
l'énigme !**

Quitter

Écran de défaite

Votre déduction n'est pas la bonne ! Dommage...
Vous pouvez continuer et comparer vos hypothèses avec la solution ou celles de l'IA.

Quitter

Continuer

8. Recensement et évaluation des risques

Risque	Probabilité	Impact	Plan de migration
Bugs ou incohérences dans la logique des règles	Élevée	Élevé	Implémenter des tests unitaires et de validation rigoureuse pour chaque règle du jeu.
Problèmes liés à la performance de l'IA (temps de calcul élevé dans des scénarios complexes)	Moyenne	Élevé	Optimiser les algorithmes.
Défaillance de la génération d'enquêtes uniques et solvables	Élevée	Élevé	Ajouter des validations automatiques et des contrôles de cohérence pour chaque enquête générée.

A. Risques Organisationnels

Risque	Probabilité	Impact	Plan de mitigation
Difficulté de communication au sein de l'équipe	Moyenne	Moyen	Organiser des réunions hebdomadaires et utiliser des outils collaboratifs.
Mauvaise gestion des délais	Moyenne	Élevé	Établir des jalons intermédiaires réalistes et prévoir des marges dans le planning global.
Dépendance excessive envers des membres clés	Moyenne	Élevé	Encourager le partage de connaissances et la documentation des étapes critiques du projet.
Difficulté à recueillir des retours utilisateurs sur les fonctionnalités développées	Faible	Moyen	Proposer des tests auprès de notre tuteur à chaque itération.

B. Risques Liés à l'Expérience Utilisateur (UX)

Risque	Probabilité	Impact	Plan de mitigation
Interface utilisateur peu intuitive	Moyenne	Élevé	Impliquer les utilisateurs finaux dans le processus de conception et réaliser des tests d'ergonomie.
Difficulté pour le joueur à comprendre les suggestions de l'IA assistante	Moyenne	Moyen	Simplifier les explications fournies par l'IA.
Interactions lentes ou imprécises	Faible	Élevé	Optimiser l'interface et minimiser le temps de réponse entre les interactions.

C. Évaluation Globale des Risques

Pour chaque risque, une **priorité d'action** est déterminée en fonction du couple **probabilité × impact** :

Priorité	Description	Exemples
Haute	Risques critiques nécessitant une attention immédiate	Bugs des règles, performance des algorithmes
Moyenne	Risques importants mais gérables	Délais de développement, UX complexe
Faible	Risques ayant un impact limité ou peu probables	Problèmes mineurs d'interface

9. Planning des itérations et rôle de chacun

Itération 1 : Compréhension de Choco-Solver et mise en place des bases du jeu

- **Objectifs principaux :**

- Un membre du groupe se concentrera sur l'apprentissage de **Choco-Solver** en suivant les tutoriels disponibles et en réalisant des prototypes. Ils auront pour objectif de comprendre le fonctionnement de la programmation par contraintes et comment les modéliser, sans lien direct avec Kronologic.
- Les deux autres membres travailleront sur la mise en place des bases du projet Kronologic, notamment l'encodage des données de jeu et le début du développement de l'interface graphique.

- **Rôle des membres du groupe :**

- **Mathieu :**

- Suivre un tutoriel complet sur Choco-Solver afin de se familiariser avec l'outil.
 - Réaliser des prototypes pour résoudre des problèmes simples. Cela a pour objectif de saisir le fonctionnement de la programmation par contraintes.
 - Comprendre comment intégrer Choco-Solver au projet.

- **Enzo et Corentin :**

- Encoder les données du jeu et créer des classes Java permettant de les modéliser (lieux, personnages, indices, temps).
 - Développer une architecture MVC afin de lier les données de jeu avec l'interface graphique de l'application.
 - Commencer le développement de l'interface graphique (page d'accueil et redirection vers la feuille de jeu).

Itération 2 : Finalisation de l'interface graphique

- **Objectifs principaux :**

- Terminer entièrement l'interface graphique, y compris la feuille de jeu et ses éléments visuels.
- Permettre au joueur de poser une question, de collecter des indices et de formuler des hypothèses et des déductions.

- **Rôle des membres du groupe :**

- **Membre 1 :**

- Terminer le développement de l'interface graphique. Cela passe par les visuels présents sur la feuille de jeu ainsi que les pages annexes permettant de poser les questions par exemple.

- **Membre 2 :**

- Mettre en place tout ce qui est relatif à la pose de questions dans l'interface graphique (réponse publique, réponse privée, choix des éléments sur lesquels portera la question).
 - Permettre au joueur de visualiser les indices qu'il a collecté au cours de la partie.

- **Membre 3 :**

- Permettre au joueur de formuler des hypothèses en fonction des indices dont il dispose. Cela se fera via des pions représentant les personnages qu'il pourra placer sur les cartes aux différents pas de temps.
- Permettre au joueur de faire une déduction s'il pense avoir trouvé le coupable, le lieu et le temps liés au meurtre grâce à un bouton présent sur la feuille de jeu.
- Implémenter les notions de victoire et de défaite.

Itérations 3 et 4 : Développement de l'IA de déduction

- **Objectifs principaux :**
 - Développer deux versions de l'IA de déduction : une reposant sur Choco-Solver et l'autre basée sur un algorithme de recherche que nous allons créer.
 - Comparer les performances des deux IA afin de choisir la plus adaptée à notre projet.
 - Intégrer les suggestions de l'IA dans l'interface graphique afin qu'elles soient visibles pour le joueur.
- **Rôle des membres du groupe :**
 - **Mathieu** : Implémenter l'IA reposant sur Choco-Solver.
 - **Enzo et Corentin** : Trouver un algorithme permettant d'effectuer des déductions et implémenter l'IA qui va l'utiliser.

Itérations 5 et 6 : Développement de l'IA d'assistance

- **Objectifs principaux :**
 - Développer deux versions de l'IA d'assistance (Choco-Solver et heuristique) qui vont aider le joueur en lui proposant les questions optimales à poser et en corrigeant ou complétant ses déductions.
 - Comparer les performances des deux IA afin de choisir la plus adaptée à notre projet.
 - Permettre une interaction fluide entre l'IA d'assistance et le joueur via l'interface graphique.
- **Tâches :**
 - **Mathieu** : Implémenter l'IA reposant sur Choco-Solver
 - **Enzo et Corentin** : Développer l'IA heuristique et s'occuper des interactions de l'IA avec le joueur.

Itération 7 : Finalisation et consolidation

Objectifs principaux :

- Finaliser toutes les fonctionnalités principales développées dans les itérations précédentes.
- Corriger les derniers bugs et incohérences afin de stabiliser l'application.
- Rédiger le document final du projet et préparer la soutenance finale.

Partie optionnelle : IA joueuse et générateur de scénarios

Si le temps le permet, nous explorerons deux fonctionnalités avancées afin d'enrichir le projet. Etant donné que nous ne savons pas si nous allons être capable de les développer, nous allons rester très vague sur le sujet en énonçant uniquement les objectifs principaux.

1. IA joueuse

Objectifs principaux :

- Développer une IA capable de jouer de façon autonome.
- S'aider des IA de déduction et d'assistance pour analyser les indices, poser des questions, et formuler des hypothèses.
- Essayer de développer une IA trouvant la solution le plus rapidement possible..

2. Générateur de scénarios

Objectifs principaux :

- Développer un système capable de générer automatiquement des scénarios de jeu variés.
- S'assurer que les scénarios générés soient uniques, cohérents, et adaptés à différents niveaux de difficulté.