

The Silent Hunt - Présentation du Projet

Noah Laghlali, Hugo Rochedreux, Emma Klingler

The Silent Hunt est un jeu de survie développé avec Roblox Studio en langage Lua. Le joueur incarne un lapin évoluant dans une forêt. L'objectif est de survivre en se nourrissant, en évitant les dangers, et surtout en échappant à un chasseur contrôlé par une intelligence artificielle évolutive. L'IA du chasseur apprend des comportements du joueur au fil du temps, jusqu'à devenir capable d'anticiper ses déplacements.

I. Vue d'ensemble du projet

1. Concept du jeu

Jeu de survie développé sur Roblox Studio en Lua.

Le joueur incarne un lapin traqué par un chasseur doté d'une IA évolutive.

Objectif : Le joueur doit survivre le plus longtemps possible tout en accomplissant des missions secondaires. Il doit se nourrir régulièrement, éviter d'être repéré et esquiver un chasseur dont le comportement devient de plus en plus intelligent. En mode coopératif, plusieurs joueurs peuvent incarner des lapins et collaborer pour survivre plus longtemps.

Chaque session de jeu est limitée dans le temps (une journée). Ce temps est affiché au joueur via une interface discrète, ou peut être caché pour augmenter la tension (coucher du soleil). Quand le temps est écoulé, la partie se termine automatiquement.

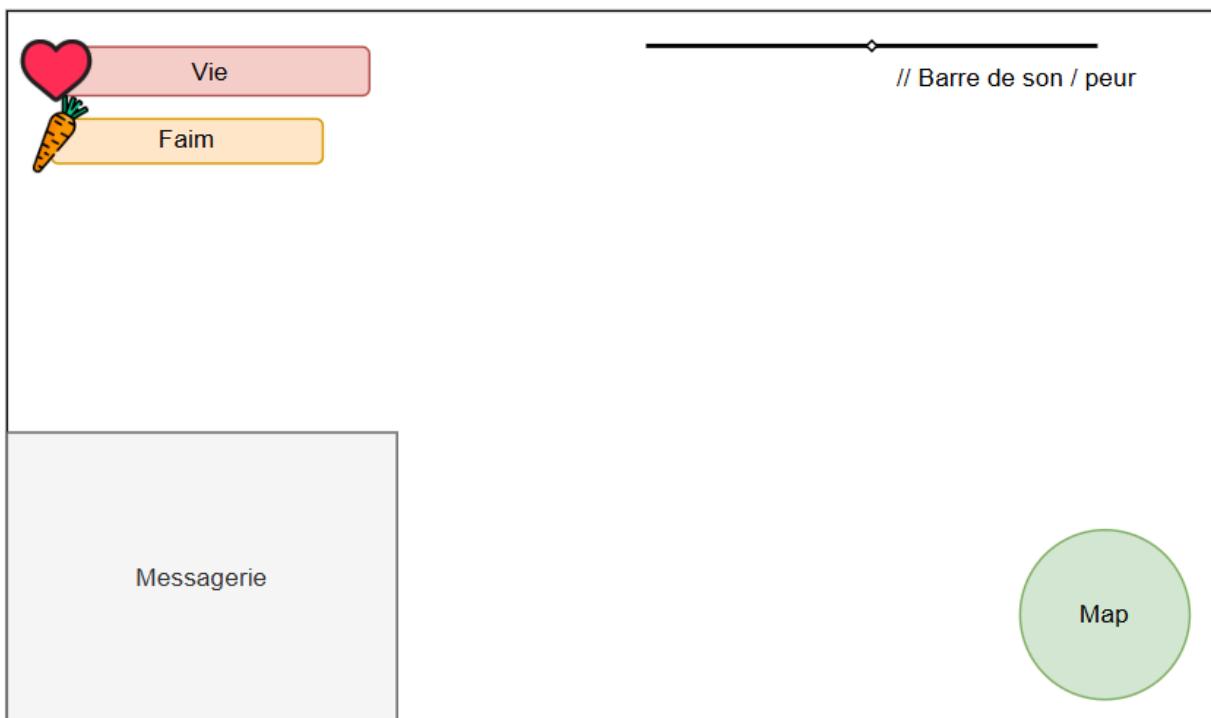
Cette contrainte temporelle structure la partie en phases : exploration prudente, montée en tension, panique finale. Dans les dernières minutes, l'IA devient plus agressive : elle accélère, pose davantage de pièges, et anticipe les comportements du joueur. Des missions spécifiques peuvent être déclenchées pendant cette phase critique.

Cette approche renforce la pression psychologique sur le joueur, l'oblige à gérer ses priorités, et rend chaque décision encore plus cruciale dans les dernières secondes.

Dans The Silent Hunt, le joueur ne subit pas les événements : il est constamment confronté à des choix décisifs. Ces choix déterminent son style de jeu, sa survie, et l'issue de la partie. Le jeu propose des mécaniques qui encouragent l'observation, la prise de risque, la prudence ou la coopération. Voici les principaux types de choix stratégiques que le joueur peut rencontrer :

2. Vision globale

- Créer une expérience immersive et dynamique où chaque partie est différente.
- L'IA apprend du joueur et adapte son comportement au fil du temps.



Maquette de l'HUD



Objectif Visuel

II. Fonctionnalités principales et secondaires

A-Documentation

- Trello
- Diagrammes UML : use case, classes, états, séquence
- Rapport technique détaillant chaque itération
- Architecture modulaire : séparation des responsabilités (client, serveur, IA...)

B- Fonctionnalités primaires

- Contrôle du lapin (vue 1re personne, mouvements, collisions).
- Systèmes de survie : faim, vie, stress.
- « IA » du chasseur (FSM : patrouille, poursuite, investigation, attente).
 - IA sous forme d'arbre de décision au début
- Système de bruit et traces.
- Terriers (cachettes + points de respawn).

- Carte principale (forêt, clairière, rivière, zones rocheuses).
- Design

C- Fonctionnalités secondaires

- IA adaptative (mémoire, anticipation, apprentissage).
- Mode multijoueur (coop à 2–100 joueurs).
- Missions secondaires et quêtes (totems, artefacts).
- Objets interactifs (leurre, caméras, filets).
- Système de stress avancé (panique, ralentissement).
- Météo dynamique et cycle jour/nuit.
- Easter Eggs et zones de lore.
 - Un lapin fantôme apparaît la nuit pour guider les joueurs en danger.
 - Une cache secrète contenant des références à d'autres jeux Roblox ou à des classiques de la survie.
 - Un terrier abandonné avec un journal crypté racontant la création du chasseur.
 - Un déguisement de buisson permettant une invisibilité temporaire.
 - Une ancienne statue de lapin qui débloque un filtre visuel spécial (nuit pixel art, style rétro).
- Simulation de parties IA vs IA
- IA lapin simulée (aléatoire ou semi-stratégique)
- Missions secondaires, objets interactifs, dispositifs de la carte activés par l'IA
 - Atteindre une clairière sans être vu : gain de nourriture
 - Trouver une plante médicinale rare : gain de vie
 - Attirer l'IA ailleurs avec un leurre : gain d'XP
 - Rejoindre un joueur et survivre ensemble : réduction de stress
 - Creuser un terrier en zone sûre : point de respawn
 - Activer un totem de forêt : vision temporaire du chasseur

- Options pour l'IA (objet que le chasseur peut utiliser au fur et à mesure des niveaux)

III. Organisation du développement (5 itérations)

Itération	Objectifs techniques	Niveau de l'IA Chasseur	IA Lapin d'entraînement	Livrables attendus
1 – Prototype	Base du gameplay : déplacement, faim, bruit	FSM simple (patrouille / poursuite)	Aucun	Prototype jouable solo
2 – Interaction	Terriers, stress, HUD	Arbre de décision réactif (bruit + vision)	Lapin aléatoire	Démo jouable avec IA crédible
3 – Adaptation	Ajout mémoire + apprentissage zones	IA adaptative (mémoire courte)	Lapin semi-stratégique	Build IA dynamique
4 – Coopération	Coop multijoueur + missions	IA multi-cible (priorisation)	Lapins simulés coordonnés	Version beta multijoueur
Finition & ML (Machine Learning)	Polish + équilibrage + tests IA	ML léger (entraînement 100 runs)	IA lapin entraînée	Version améliorée avec IA plus entraînée

IV. Organisation de l'équipe

Domaine de référence	Contributions principales
----------------------	---------------------------

IA & architecture technique	Conception des comportements du chasseur, intégration du Pathfinding, structure client/serveur, optimisation des scripts.
Gameplay & design système	Création et équilibrage des mécaniques de survie, stress, bruit, terriers, et interactions joueur-IA.
Interface & environnement	Développement du HUD, des retours visuels/sonores, création des zones de la carte, mise en scène et polish visuel.

Tous participent à :

- La conception et l'évolution du gameplay.
- Le développement de scripts Lua dans Roblox Studio.
- Les tests et calibrages des comportements IA / joueurs.
- La documentation (itérations, diagrammes, rapports).

Méthode de travail :

- Organisation Agile sur 5 sprints (une itération = un sprint) (la 6^e itération est prévue pour des améliorations et peaufinage) .
- Outils : VSCode + darw.io + RobloxStudio + GitHub + Trello.
- Communication : Discord.

V. Description des systèmes clés

1) IA du chasseur

Le chasseur constitue le cœur de The Silent Hunt. Il s'agit d'une intelligence artificielle évolutive conçue pour traquer, analyser et anticiper les actions du joueur. Son comportement est fondé sur une logique d'états et un système décisionnel complexe, capable de s'adapter progressivement à la stratégie des joueurs.

1.1. Structure générale de l'IA

Le chasseur repose d'abord sur une FSM (Finite State Machine), évoluant vers un arbre de décision puis une IA adaptative.

Les principaux états sont : patrouille, investigation, alerte, poursuite, recherche et embuscade.

Chaque transition dépend des stimuli perçus (bruit, traces, vision) et d'une mémoire à court terme qui conserve les informations récentes (zones suspectes, empreintes, sons entendus).

1.2. Prise de décision

L'arbre de décision détermine les priorités selon la situation :

- Si le joueur est visible : poursuite.
- Si un bruit est perçu : investigation ou alerte selon l'intensité.
- Si des traces sont détectées : suivi des pistes.
- Si aucune activité récente : patrouille ou embuscade.

L'IA hiérarchise ces comportements en fonction du niveau d'alerte et du contexte.

1.3. Système d'alerte et mémoire

L'IA dispose d'une variable d'alerte évoluant de calme à traque totale. Chaque action du joueur (bruit, empreinte, fuite) augmente cette valeur, qui redescend lentement avec le temps.

Une mémoire temporaire lui permet de revisiter des zones suspectes, de poser des pièges ou d'attendre en embuscade.

1.4. Objets et capacités secondaires

Le chasseur dispose d'outils qui enrichissent sa stratégie :

- Scanner de zone (détection temporaire des lapins visibles).
- Mines et filets automatiques (pièges déclenchés par passage).
- Caméras animales (surveillance à distance).
- Brouilleurs (désactivation temporaire des interactions).
- Balises et leurres (pièges sonores et appâts).
- Sabotage de nourriture et sondes thermiques.

Ces outils apparaissent progressivement selon le jour ou le niveau de difficulté, rendant l'IA plus imprévisible.

1.5. Évolution et niveaux de difficulté

L'IA progresse sur plusieurs niveaux :

- Niveau 1 : logique FSM simple (patrouille, poursuite, alerte).
- Niveau 2 : arbre de décision avec priorités.
- Niveau 3 : IA adaptative avec mémoire comportementale.
- Niveau 4 : apprentissage léger basé sur des simulations (100 tests).
- Niveau 5 : apprentissage avancé et comportement prédictif (10 000 tests).

L'évolution se traduit par une meilleure anticipation des mouvements, une utilisation plus fine des objets et une coordination de pièges.

1.6. Réaction aux actions du joueur

L'IA perçoit les bruits, traces et mouvements. Elle ne connaît jamais la position exacte du joueur, mais déduit une zone probable d'activité.

Elle réagit aux habitudes répétées : une cachette utilisée plusieurs fois ou un itinéraire fréquent devient dangereux, car l'IA y posera des pièges ou y tendra une embuscade.

1.7. Intention de conception

L'objectif est de créer une IA crédible et stressante, sans triche ni omniscience.

Elle ne réagit pas instantanément, mais apprend, patiente et punit les erreurs répétées.

Sa progression dans la partie reflète celle du joueur : plus le joueur maîtrise les mécaniques, plus le chasseur devient intelligent et menaçant.

2) Lapin (joueur ou IA d'entraînement)

- Systèmes vitaux : faim, vie, stress.
- Actions : manger, creuser, se cacher, fuir, détourner l'attention.
- Lapin IA (entraînement) : comportements aléatoires ou semi-logiques servant à calibrer l'IA du chasseur.

3) Système de bruit

- Chaque action produit un son (marche, course, creusage, objet).
- Intensité et durée influent sur la réaction du chasseur.
- Le joueur peut exploiter le bruit (leurre, distraction, chaos sonore).

4) Terriers

Dans *The Silent Hunt*, les terriers sont des éléments de gameplay essentiels servant de cachettes, points de repos et lieux stratégiques. Toutefois, afin d'éviter qu'ils ne deviennent une stratégie dominante, différentes mécaniques équilibrivent leur utilisation, tout en apportant profondeur et variété.

Fonctions de base :

- Cachette temporaire : permet de réduire le stress et d'échapper au chasseur.
- Point de respawn : certains terriers deviennent des zones de réapparition si le joueur y reste en sécurité.
- Construction de terriers : le joueur peut creuser son propre abri à un emplacement stratégique, mais cela prend du temps et le rend vulnérable.

Mécaniques d'équilibrage classiques :

- Durée d'utilisation limitée : impossible de rester indéfiniment dans un terrier.
- Temps de recharge : un terrier utilisé récemment devient indisponible pendant un certain temps.
- Bruit à l'entrée et à la sortie : détectable par l'IA si elle est proche.
- Stress inversé : rester trop longtemps dans un terrier peut faire remonter la jauge de stress.
- Faim persistante : la faim continue de baisser même dans un terrier.
- IA adaptative : si un terrier est trop utilisé, l'IA peut le piéger, le surveiller ou y attendre en embuscade.
- Instabilité : certains terriers s'effondrent après plusieurs utilisations.

Mécaniques avancées et originales :

- Terriers à usage unique : certains abris disparaissent après une seule utilisation.
- Terriers maudits ou spéciaux : réduisent le stress mais déclenchent des effets secondaires (hallucinations, bruit parasite...).
- Terrier-trap : l'IA peut améliorer ou piéger un terrier pour tendre une embuscade.
- Terrier réseau : permet de se déplacer rapidement entre plusieurs cachettes reliées, une fois débloqué.

- Odeur résiduelle : l'IA peut détecter l'odeur laissée par un joueur trop longtemps caché.
- Amélioration de terriers : certains joueurs peuvent débloquer des options (stockage, sortie secondaire, dissimulation renforcée).
- Terriers sociaux (coop) : en multijoueur, un joueur peut héberger un coéquipier, mais cela génère plus de bruit.
- Terriers occupés : certains terriers sont gardés par des animaux (hostiles ou alliés potentiels).
- Terriers narratifs / secrets : certains abris sont liés au lore, débloquent des bonus, ou mènent à des zones cachées.
- Système de réputation : l'IA garde en mémoire les terriers souvent utilisés et ajuste son comportement en conséquence.

5) Système de survie

- **Faim** : diminue avec le temps. À zéro, elle fait baisser la vie.
- **Vie** : diminue en cas de blessure ou de faim extrême. Si elle atteint zéro, la partie est terminée.
- **Stress** : augmente en cas de proximité du chasseur ou de bruit excessif. Affecte la vitesse et génère du bruit.
 - Variable influençant le bruit, la vitesse, et la perception.
 - Montée du stress = panique / perte de contrôle.
 - Réduction : cachette, entraide, missions réussies.

6) Carte et environnement

- Forêt principale (zones de camouflage et de danger).
- Clairières et rivières (visibilité et bruit fort).
- Zones rocheuses (vue stratégique).
- Tours de guet (lieux d'observation IA).
- Tanières et zones spéciales (missions, lore).
- Rivière

7) Missions et défis (multi ou solo)

- Collecter 3 plantes rares dans la forêt sans se faire repérer.
- Atteindre un terrier éloigné avant la tombée de la nuit.
- Créer un leurre pour détourner le chasseur d'une zone stratégique.
- Sauver un lapin blessé et l'escorter en sécurité.
- Trouver un ancien artefact pour gagner un bonus de vision.
- Accomplir une mission bonus dans un temps imparti (ex : explorer une tour, saboter un dispositif).

8) Contraintes

- Joueur immobile trop longtemps : IA fouille davantage
- Cachette utilisée plusieurs fois : IA la surveille
- Trop de bruit : IA pose des pièges
- Joueurs proches : IA scanne la zone plus vite
- Météo ?
- Pièges :
 - Piège sonore : déclenché par le bruit, attire le chasseur.
 - Faux terrier : le joueur s'y enferme et se retrouve temporairement bloqué.
 - Plante toxique : diminue la vie ou le champ de vision.
 - Zone marécageuse : ralenti le joueur et laisse plus de traces.
 - Brouillard artificiel : réduit la vision mais masque les traces.
 - Trappe activée par le chasseur dans un point stratégique.

9) Gameplay solo

- Contrôle du lapin à la première personne
- Gestion de la faim, de la vie et du stress

- Mécaniques de base : se cacher, creuser un terrier, manger pour survivre
- Système de traces et de bruit détectables par l'IA

10) Mode multijoueur (coopératif)

En mode coop, jusqu'à 100 joueurs peuvent collaborer. Le chasseur devient plus agressif, l'IA s'adapte aux stratégies de groupe. Les joueurs peuvent partager des ressources, coordonner des leurre, ou se sauver mutuellement.

10.1) Intérêt du multijoueur

- Permet une coordination stratégique (attirer, distraire, explorer)
- Apporte une communication indirecte (signaux, bruits)
- Entraîne une IA plus complexe qui doit prioriser ses cibles
- Accroît la tension (partage de responsabilité, erreurs collectives)
- Ajoute des défis coopératifs uniques
- Augmente la re jouabilité avec des stratégies variées

11) Système de bruit

Dans The Silent Hunt, le bruit est une mécanique cruciale qui influence la manière dont l'IA perçoit, localise et poursuit le joueur. Chaque action du lapin produit un niveau sonore spécifique, ce qui peut déclencher des réactions variées de l'IA chasseur.

1. Production du bruit

Le joueur produit du bruit selon :

- **Sa vitesse de déplacement :**
 - Courir = bruit élevé
 - Marcher = bruit modéré
 - Ramper / rester immobile = silence quasi total
- **Le type de sol :**
 - Feuilles sèches, branches, cailloux = plus de bruit
 - Terre humide, herbe, mousse = atténuation
- **Les actions contextuelles :**
 - Entrer/sortir d'un terrier
 - Creuser un abri
 - Manger à la hâte
 - Toucher certains objets interactifs (pièges, leviers)

2. Réaction de l'IA au bruit

L'IA chasseur possède une "zone d'écoute" autour de sa position. Plus le bruit est fort, plus la zone de détection est large.

Lorsqu'un bruit est perçu, l'IA peut :

- Changer d'état (ex : passer de patrouille à investigation)
- Se déplacer vers la source approximative du bruit
- Entrer en mode "alerte" : vitesse augmentée, vision renforcée
- Déployer un piège ou scanner la zone
- Marquer l'endroit dans sa mémoire comme zone à surveiller dans le futur

L'IA ne détecte pas la position exacte, mais une zone approximative. Elle se rend sur place, inspecte, puis adapte son comportement selon les indices trouvés (empreintes, mouvements, objets déplacés...).

3. Intensité et durée du bruit

Le bruit est caractérisé par deux paramètres :

- Intensité : à quel point le son porte
- Durée : combien de temps l'IA "entend" quelque chose

4. Stratégies possibles du joueur

Le système de bruit permet aux joueurs :

- De faire diversion volontairement (ex : courir dans une direction opposée puis se cacher)
- D'éviter les zones sonores (bois mort, cailloux, branches)
- De créer des routines que l'IA peut apprendre et anticiper
- D'utiliser le silence comme arme : immobilité, ramper, préparation tactique

5. IA évolutive

L'IA peut évoluer :

- En réduisant son seuil de tolérance au bruit (elle devient plus attentive)
- En modifiant sa patrouille selon les zones où elle entend souvent du bruit
- En apprenant à ignorer les leurres trop répétés
- En posant des pièges sonores dans les chemins fréquentés

12) Système de stress

Dans *The Silent Hunt*, la jauge de stress représente l'état de tension psychologique du lapin. Elle influe sur ses performances (bruit, vitesse, perception) et ajoute une pression indirecte au gameplay.

Voici tout ce qui peut déclencher ou augmenter le stress, avec des exemples concrets, et en face, les effets négatifs associés :

Situation	Détail / Exemple concret
Proximité du chasseur	Le chasseur est dans un rayon de 20 mètres, même sans être vu.
Bruit fort produit par le joueur	Le lapin court dans des feuilles mortes, déclenchant une alerte sonore.
Détection visuelle par le chasseur	L'IA regarde dans la direction du joueur pendant plus de 2 secondes.
Traces laissées trop longtemps	Le joueur marche en boucle au même endroit, laissant une empreinte exploitable.
Vision du chasseur en pleine course	Le chasseur fonce vers la position du joueur : le stress grimpe en flèche.
Un autre joueur est capturé (mode multi)	En coop, voir un coéquipier se faire attraper déclenche un pic de stress collectif.
Mission échouée ou abandonnée	Le joueur échoue à activer un totem, ou fuit une mission à moitié faite.
Rester immobile trop longtemps	Par exemple, dans un terrier instable ou sans savoir si l'IA est proche.

Seuil de stress	Conséquences
Faible (0–30%)	Aucun effet visible.
Modéré (30–60%)	Le lapin commence à faire plus de bruit (mouvements moins discrets).
Élevé (60–90%)	- Baisse de vitesse de déplacement

	<ul style="list-style-type: none"> - Respiration audible - Moins précis dans ses actions
Critique (90–100%)	<ul style="list-style-type: none"> - Vision altérée (écran trouble) - Déclenche parfois une panique automatique (fuite non contrôlée) - Incapacité temporaire à se cacher

Pour réduire le stress:

- Se cacher dans un terrier sécurisé
- Ne pas bouger dans une cachette valable
- Être proche d'un autre joueur (coop)
- Réussir une mission secondaire
- Manger certains aliments relaxants (menthe)
- Sauver un allié capturé

13 LE CHASSEUR, L'IA

VI Choix stratégiques du joueur

The Silent Hunt est conçu pour encourager la réflexion et la prise de décisions tactiques. Le joueur incarne un lapin confronté à des choix permanents : se cacher ou fuir, utiliser une ressource ou la conserver, aider un autre joueur ou se sauver soi-même.

Chaque décision a des conséquences : bruit généré, traces laissées, opportunités perdues ou créées. La stratégie du joueur repose donc sur l'observation, l'anticipation des mouvements de l'IA, et l'adaptation aux situations imprévues.

Par exemple, courir pour fuir augmente le bruit, mais peut sauver la vie. Se cacher plusieurs

fois au même endroit devient risqué, car l'IA apprend. Certaines missions offrent des récompenses, mais obligent à sortir de sa zone de sécurité. En multijoueur, la coopération stratégique devient essentielle : diversion, sacrifice, synchronisation d'actions, partage d'objets.

1 Discréption vs Rapidité

Courir permet de fuir rapidement un danger mais génère du bruit et laisse plus de traces. Marcher lentement ou ramper réduit la détection, mais rallonge les déplacements.

Exemple : un joueur poursuivi peut choisir de courir vers une zone de ronces pour semer l'IA rapidement, ou marcher doucement jusqu'à une cachette connue au risque d'être rattrapé.

2 Sécurité vs Récompense

Certaines ressources sont faciles d'accès mais peu efficaces, d'autres sont plus bénéfiques mais situées dans des zones à haut risque.

Exemple : choisir entre manger une carotte en bord de forêt (peu nourrissante) ou tenter de récupérer une baie énergétique au centre d'une clairière très surveillée.

3 Réutilisation vs Innovation

Le joueur peut utiliser plusieurs fois les mêmes tactiques ou cachettes, mais l'IA apprend ses habitudes. Il doit varier ses stratégies pour rester imprévisible.

Exemple : un joueur qui revient systématiquement au même terrier pourra y être attendu par l'IA lors des prochaines minutes.

4 Sacrifice vs Coopération

En multijoueur, aider un allié en danger peut sauver l'équipe mais met le joueur en risque. Fuir seul assure la survie personnelle.

Exemple : un joueur peut déclencher volontairement un piège sonore pour détourner l'IA pendant qu'un autre libère un coéquipier capturé.

5 Gestion des objets limités

Le joueur dispose d'un nombre limité d'objets (leurre, camouflage, soins) et doit choisir le bon moment pour les utiliser.

Exemple : utiliser un leurre sonore pour échapper à une poursuite immédiate ou le conserver pour la phase finale où l'IA est plus agressive.

6 Détour vs Trajectoire directe

Le joueur peut prendre un raccourci risqué ou un long détour sécurisé pour atteindre un objectif.

Exemple : atteindre une plante médicinale rapidement à travers une rivière bruyante, ou faire le tour par un sentier caché plus long.

7 Déclenchement de distraction

Poser volontairement des traces ou générer du bruit peut servir à tromper l'IA.

Exemple : courir dans une direction opposée puis se cacher dans un buisson pour faire croire à une fuite.

8 Choix de mission ou de survie

Certaines missions bonus apparaissent durant la partie, mais y participer peut compromettre la survie.

Exemple : tenter d'activer un totem de forêt (récompense : vision du chasseur) dans une zone à risque, ou rester caché pour préserver ses chances de survie

9 Sauver un autre lapin

Cela rapporterait au joueur: une réduction de stress, un bonus de faim, un buff temporaire (pendant un certain temps le joueur devient plus discret ou plus rapide) etc

Matière et concepts académiques qui sont inclus dans le projet :

1. Algorithmique et Intelligence Artificielle

Conception d'arbres de décision, de systèmes à états finis et d'une IA adaptative. Gestion de la logique, de la perception et de l'apprentissage progressif du chasseur.

2. Programmation et Architecture logicielle

Utilisation du langage Lua et mise en place d'une architecture modulaire (client / serveur / IA). Respect des principes de programmation orientée objet et de la séparation des responsabilités.

3. Réseaux et communication client-serveur

Gestion des échanges entre le serveur Roblox et les clients, synchronisation des actions multijoueur, et optimisation des performances réseau.

4. Conception logicielle et modélisation UML

Modélisation du jeu à travers des diagrammes de classes, d'états et de séquences. Définition claire des entités, interactions et comportements.

5. Gestion de projet et méthode agile

Organisation du travail en cinq itérations successives, définition des objectifs par sprint, planification et suivi des tâches via Trello et GitHub.

6. Qualité logicielle et tests

Mise en place de scénarios de test, validation du comportement de l'IA, ajustement des paramètres de jeu et documentation du code.

7. Communication et collaboration

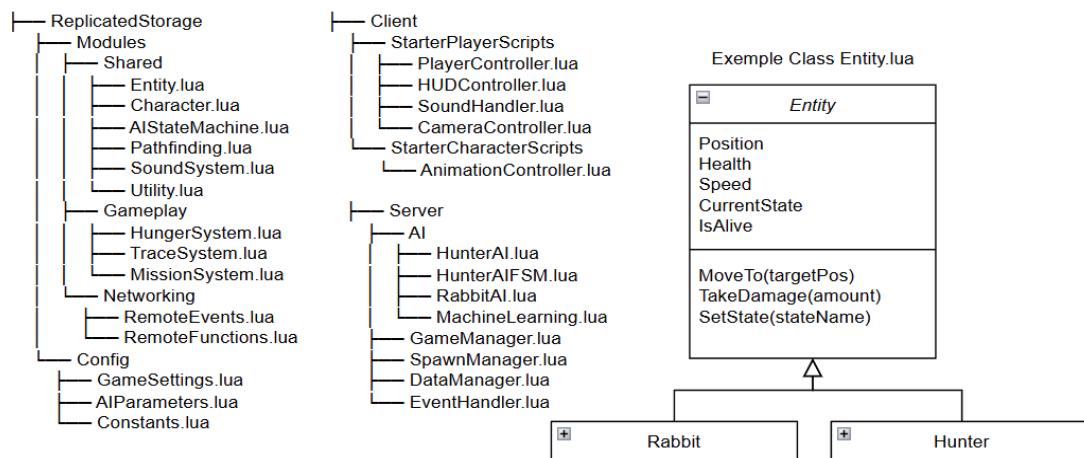
Rédaction du dossier de conception, documentation technique, réunions de coordination.

étude de l'existant :

- Hello neighbor : IA adaptative et apprentissage du joueur
- Dead by daylight : Multijoueur asymétrique et tension constante
- Alien Isolation : IA organique et imprévisible
- Shadow of mordor : Personnalisation et persistance des ennemis

Jeu Source	Élément retenu	Adaptation prévue
Hello neighbor	IA qui apprend les habitudes	Mémoire des zones et des actions
Dead By Daylight	Asymétrie et tension	un chasseur, plusieurs proies
Alien Isolation	IA organique et imprévisible	FSM + bruit + patrouille libre
Shadow of Mordor	Ennemis évolutifs	Chasseur qui apprend des joueurs

exemple de structure du projet :



Exemple Class Entity.lua

```

Entity
Position
Health
Speed
CurrentState
IsAlive

MoveTo(targetPos)
TakeDamage(amount)
SetState(stateName)

```



Rabbit Hunter

