

## Introduction :

Afin de mettre en application vos connaissances en intelligence artificielle pour les jeux vidéos, vous allez travailler sur les différents aspects de l'IA pour un mini-RTS.

### Objectifs:

Réaliser une IA comprenant les couches déplacement, décisionnel et stratégie pour jeu de type RTS (Real Time Strategy).

Votre projet devra :

- Gérer le **déplacement individuel** d'une unité
  - cela englobe :
    - sa **navigation** (choix du meilleur chemin dans un environnement)
    - son **mouvement**, qui doit être fluide et paramétrable
  - **Il est conseillé d'utiliser les outils de navigation de Unity** pour gérer la navigation et le suivi de chemin (**Navmesh, NavmeshAgent**, etc.). Cette partie est globalement **déjà implémentée**.
- Gérer le **déplacement en formation** des unités
  - Les unités pourront se regrouper en formations ordonnées de votre choix (cercle, rectangle, flèche, adaptative...)
  - Le déplacement devra alors être coordonné (déplacement à vitesse homogène, respect des positions...)
- Gérer un **système d'état logique et décisionnel** pour chacune des unités
  - état de déplacement, état d'attaque en cas d'ennemis ou de cible à portée, retraite, capture de point stratégique...
  - un système d'attaque simple est implémenté (envoi de dégâts à une unité adverse qui est à distance de tir)  
⇒ il faudra le reprendre pour qu'il soit piloté par une IA et pas activé automatiquement comme c'est le cas
- Implémenter un **système de décision stratégique** pour l'armée adverse
  - Cette partie de l'IA devra être capable de créer des unités, les regrouper en formations, attaquer et défendre des points situés sur la carte.  
L'armée pourra capturer des cibles situées sur la carte lui permettant de construire davantage d'unités.  
Le but final d'une armée est de détruire la base de l'adversaire.

Vous êtes libre dans les algorithmes et structures utilisés (formation steering, FSM, Decision Tree, Behaviour Tree, Utility System, Action Planning...)

Vous pouvez modifier le projet template à votre guise, tant que cela va dans le sens de l'amélioration de l'intelligence artificielle. Des ajouts et modifications du gameplay sont possibles mais ne seront pas évaluées s'il n'y a pas de lien avec l'IA.

## Informations

Le **projet template RTS** propose :

- un système d'unités avec déplacement simple basé sur le component NavmeshAgent
- un système de bâtiments "factory" permettant de créer des unités de différents types
  - 2 types de **factories** sont disponibles:
    - FactoryHQ qui permet de construire des unités légères (troupes, véhicules)
    - FactoryHeavy qui permet de construire des unités lourdes (tanks)
  - 3 types d'**unités légères** sont disponibles
    - Trooper : soldat de base, peut tirer et réparer des unités
    - Trike : moto rapide, peut tirer
    - Quad : buggy assez rapide, peut tirer
  - 3 types d'**unités lourdes** sont disponibles
    - Tank : vitesse moyenne, tir assez puissant
    - Tank Heavy A : lent, tir puissant, grande portée
    - Tank Heavy B : lent, tir très puissant, portée moyenne
  - chaque type d'unité aura ses propres réglages des caractéristiques suivantes
    - MaxHP
    - DPS (damage per second), Attack Frequency, Attack Distance
    - Capture Distance
    - RPS (repair per second), Repair Frequency, Repair Distance
    - Speed, Angular Speed, Acceleration
- un contrôleur permettant de sélectionner des unités (+raccourcis de sélection par type)
- un niveau de jeu de test vide "BasicLevel", avec juste une factory.
- des niveaux "BattleField" et "LargeBattleField", plus vastes et accidentés, permettant d'affronter une armée adverse. **LargeBattleField** est la map de référence pour le rendu.
- un système de capture de points stratégiques sur la carte

- les **features gameplay** suivantes sont aussi disponibles :
  - chaque joueur démarre avec un certain nombre de points de construction faisant office de ressources
  - créer une unité coûte des points de constructions, variable en fonction du type d'unité : Trooper, Quad, Tank, Heavy Tank, etc.
  - la capture d'une cible fera gagner de nouveaux points de construction.
  - plus il y a d'unités de votre camp autour d'une cible, plus vite elle sera capturée. Plus les unités sont lourdes, plus la capture sera rapide.
  - perdre une unité vous rend ses points de construction.
  - gestion du Fog of War et d'une Minimap
- les unités ont à disposition les **actions** suivantes
  - **déplacement** vers un point de la carte
  - **capture** d'une cible
  - **attaque** d'une unité ou de la base ennemie
  - **réparation** d'une unité ou base alliée

**Ces actions** sont pour le moment branchées sur un controller pour simple test, mais **devront être intégrées et pilotées par une IA**

Il s'agit d'une base de travail que vous êtes libres de modifier et d'améliorer.

#### Idées de features Bonus :

- **Éditeur visuel décisionnel** : créer un outil de type graphe pour éditer la structure décisionnelle des entités du jeu (FSM, Decision Tree, Behaviour Tree...)
  - **Éditeur de formations**
  - **Formations dynamiques** : la structure de la formation dépendra de la topologie, du type d'unités, de leur nombre...
  - Utilisation d'une **Map d'Influence** pour gérer les décisions stratégiques et tactiques
  - **Prise en compte du "fog of war"** au niveau de la perception de l'IA
  - **Ajout d'unités aériennes** avec leur propre comportement (bombardement, tir en piqué) et un déplacement caractéristique (vol continu, vol stationnaire, ...). L'aspect particulier de leur navigation devra être pris en compte.
- Des prefabs et data d'unités aériennes sont présents dans le projet mais non intégrées.
- **Ajout d'unités de type "tourelles"** fixes (création à partir d'un nouveau type de factory, comportement spécifique Sol-Sol / Sol-Air...) et positionnement stratégique par l'IA adverse

# Organisation

- Il s'agit d'un **projet de groupe de 2 personnes**.
- Vous devez rendre ce projet pour le **16 avril à 20h00** maximum.
- Le projet devra se trouver sur l'espace **GIT** mis à disposition.
- Vous travaillerez avec le moteur **Unity** et coderez en **C#**
- Vous vous focaliserez sur les modifications qui concernent l'IA, ou les moyens de l'améliorer
- Un **TDD** vous sera demandé en début de projet pour bien préparer votre réflexion sur le contenu du projet

## Ressources :

- voir cours IA Déplacement, Décision et Stratégie
- il est autorisé d'ajouter des assets graphiques externes
- les plugins non natifs ne sont pas autorisés