

Projet Techniques de Développement logiciel - Tower Defense

CORVISIER Jean-Christophe, DELORO Yonatan, KHELDOUNI Mohammed-Amine, MARCHAIS Victor

ENPC

Encadré par : Olivier CROS, Docteur en Temps Réel et Systèmes Embarqués chez ECE Paris.

Introduction

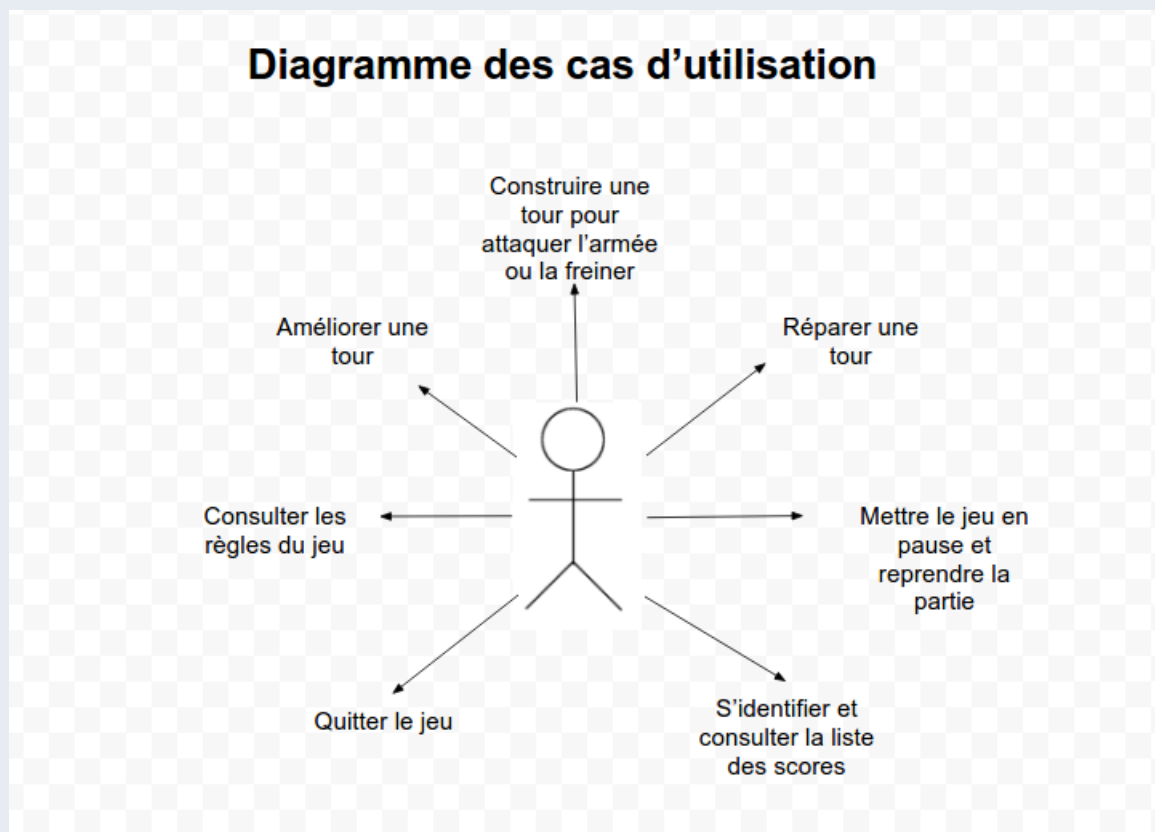
Nous avons choisi dans le cadre de ce projet de développer en agile une version facile à prendre en main et facilement améliorable d'un jeu de Tower Defense.

Il s'agit d'un jeu de stratégie à un joueur dont le principe est d'acheter ou d'améliorer un certain nombre de tours. Celles-ci sont placées le long d'un chemin où avance une armée, afin de l'empêcher d'atteindre les bases du joueur. Les soldats arrivent par vagues de plus en plus nombreuses et résistantes. Le jeu prend fin lorsque les bases du joueur sont détruites.

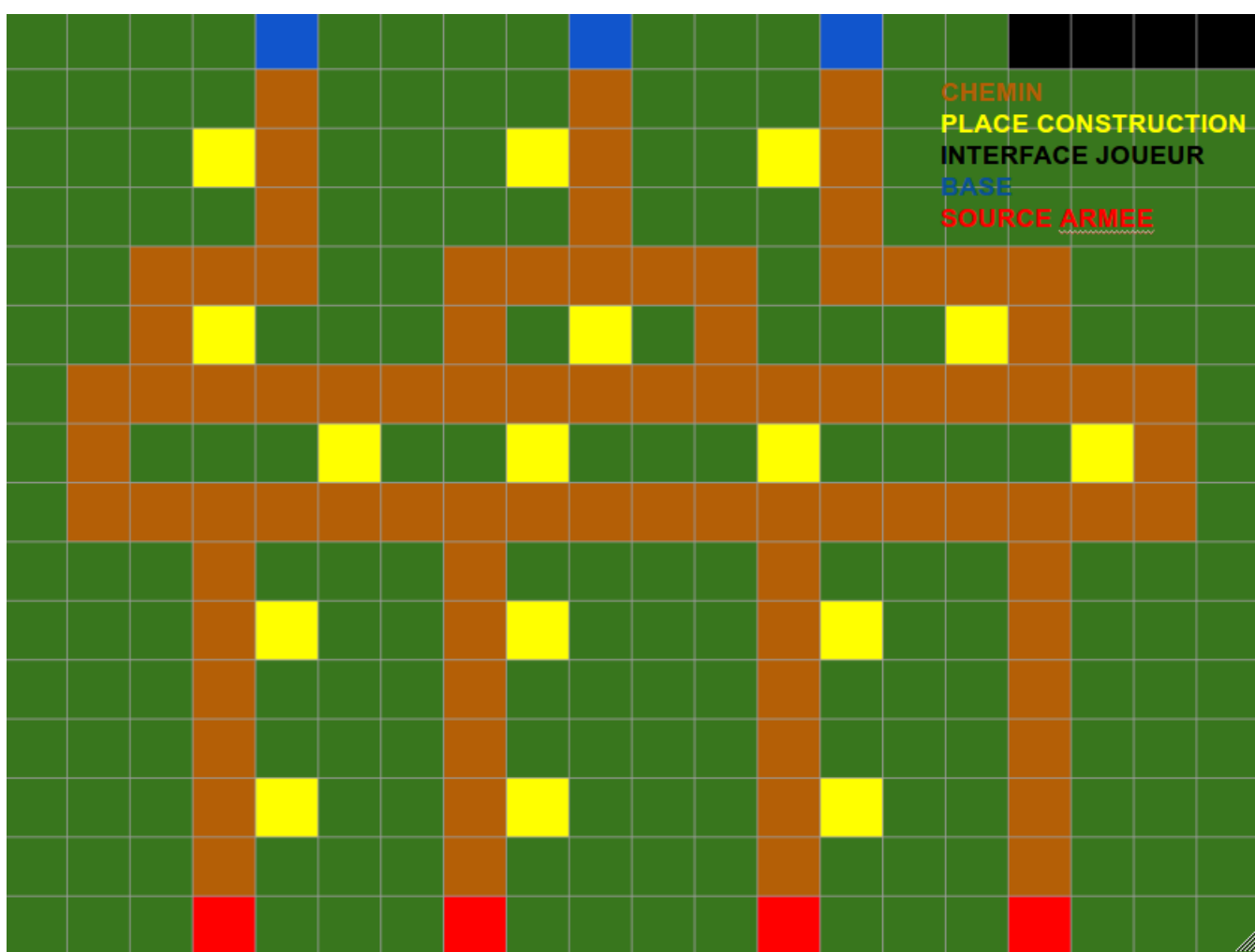
Objectifs

Nous avons pour ambition de créer une expérience de jeu simple, agréable et addictive :

- En proposant un ensemble vaste de stratégies et intéressantes à explorer choisir non seulement l'emplacement de la tour mais aussi son pouvoir, choisir entre réalimenter une tour, améliorer sa résistance et sa portée ou en construire une nouvelle.
- En dotant l'armée d'une *intelligence artificielle*, qui adapte son mouvement à la géographie de la carte (emplacement des bases) mais aussi au comportement du joueur (emplacement des tours).
- En offrant un graphisme suffisamment riche qui n'endommage pas les performances du jeu.

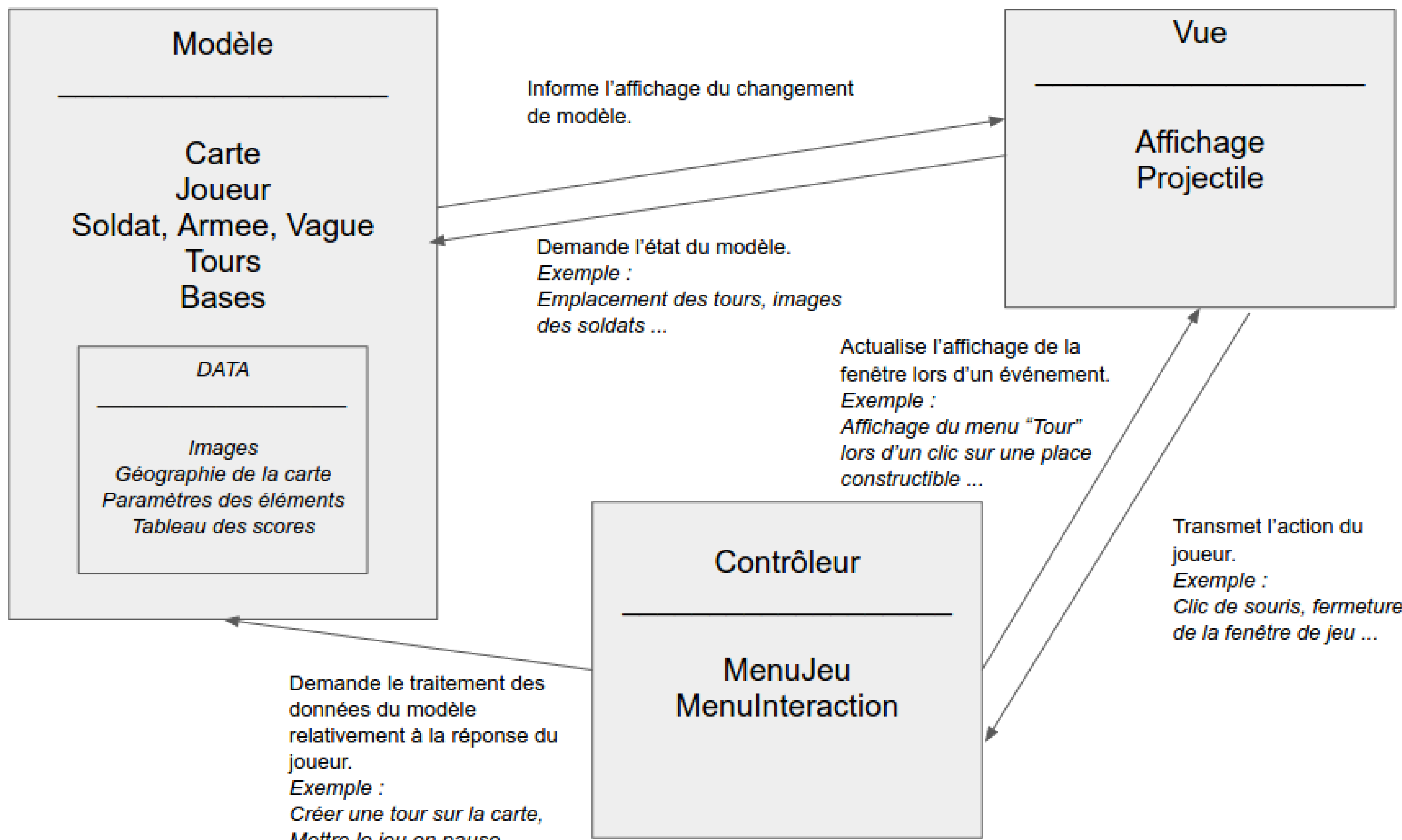


Les éléments du jeu (features)



- Le tapis de jeu, composé de différentes zones : chemin, emplacement constructible de tour, décors et bases.
- Les tours sont positionnées sur des emplacements constructibles. Une tour attaque le soldat le plus proche tant qu'il est à sa portée. Elle se caractérise par ses points de vie, ses munitions, ses dégâts, sa portée, son cout d'amélioration et son cout d'entretien.

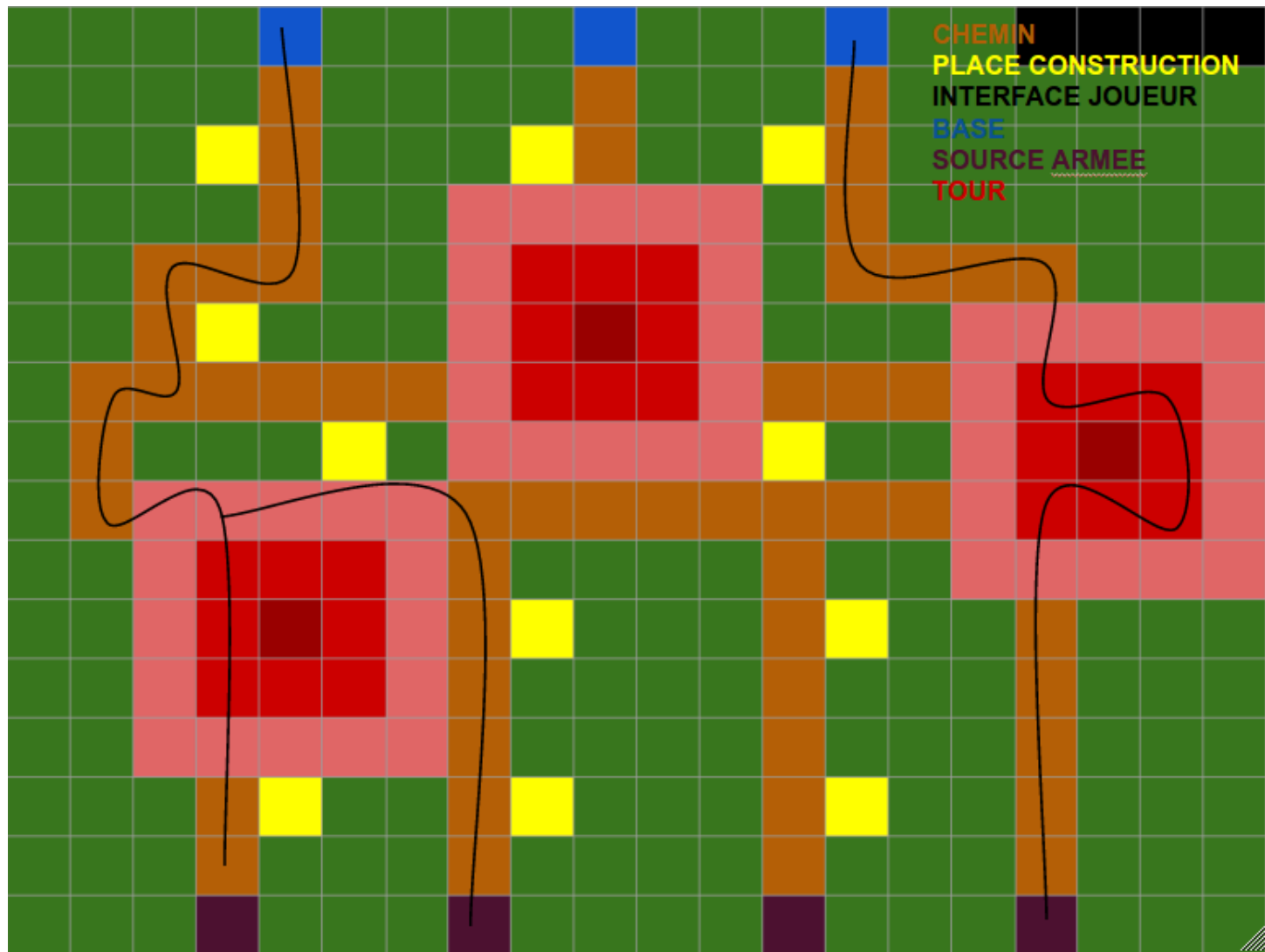
Architecture logicielle



Choix techniques

Mouvement des soldats

Nous avons été confronté au problème du mouvement des soldats. En effet, une fois sorti de la source, un soldat doit s'approprier le chemin et se diriger vers une base pour la détruire. De plus, nous avons imposé une contrainte d'aléa sur ce mouvement pour rendre la partie plus imprévisible. Pour résoudre ce problème, nous avons créé une **carte des coûts** dont l'armée a connaissance. Cette dernière cherche alors à éviter les emplacements à haut niveau de danger (présence d'une puissante tour par exemple).



Interaction avec le joueur

La difficulté rencontrée lors de la conception du contrôleur était celle de l'affichage des interactions et la transmission de l'action du joueur. La solution a été de créer un **menu d'interaction** qui accompagne le joueur tout au long de son expérience de jeu. Ce dernier peut ainsi prendre sa décision et la communiquer au menu qui transmettra l'information au modèle ainsi qu'à la vue.



Fluidité et FPS

Le problème de la fluidité du jeu est l'un des problèmes majeur que rencontre tout concepteur de jeux vidéos. En effet, au début du projet, la fluidité des mouvements n'offrait pas au joueur une bonne expérience de jeu par la lenteur des soldats. De plus, le mouvement de la souris affectait grandement les performances graphiques du jeu. Pour contourner ce problème, il a fallu faire appel à la notion de **FPS** (*Frames per second*) pour limiter le nombres d'images diffusées par le logiciel en une seconde à 60. Il a également fallu limiter les événements pris en compte par le menu pour qu'il ne teste pas son état pour un simple mouvement de la souris.

Tests du logiciel

Implémenter différents types de stratégies du joueur (*ex* : gloutonne, *i.e.* créer toutes les tours possibles puis les améliorer "uniformément") pour calibrer automatiquement et par simulation les paramètres du jeu (ex : durée ou résistance des vagues) de telle manière à satisfaire certains critères du type "tant de pourcents des bases détruites après tant de vagues".

Screenshots

Améliorations souhaitées

- Générer aléatoirement des cartes qui satisfassent un certain nombre de critères (nombre de branchements des chemins, zone de couverture maximale du chemin depuis des emplacements constructibles ...)
- Ajouter des *templates* de carte neige et amélioration des graphismes (transformation des tourelles lors de l'amélioration, animation sur les soldats touchés par un projectile ...)
- Créer le jeu dual où le joueur crée son armée pour éradiquer des bases défendues par une I.A. plus ou moins évoluée.