Holy Warhammer of Doom

# Préalable

Dans le cadre du cours de programmation II, les élèves doivent programmer un jeu de plateau pour développer leur capacité à développer et gérer une interface graphique ainsi que peaufiner leur capacité à programmer en objet. Nous avons été choisis pour offrir un modèle détaillé d’un jeu qui leur offre un défi intéressant à la hauteur de leur capacité.

# Objectif

Nous devons développer un jeu correspondant aux critères établis par le client. Le programme devra couvrir les notions d’objets et d’interface graphique couvertes dans le cadre du cours. Tout en étant simple le programme devra néanmoins offrir un défi intéressant.

# Système proposé

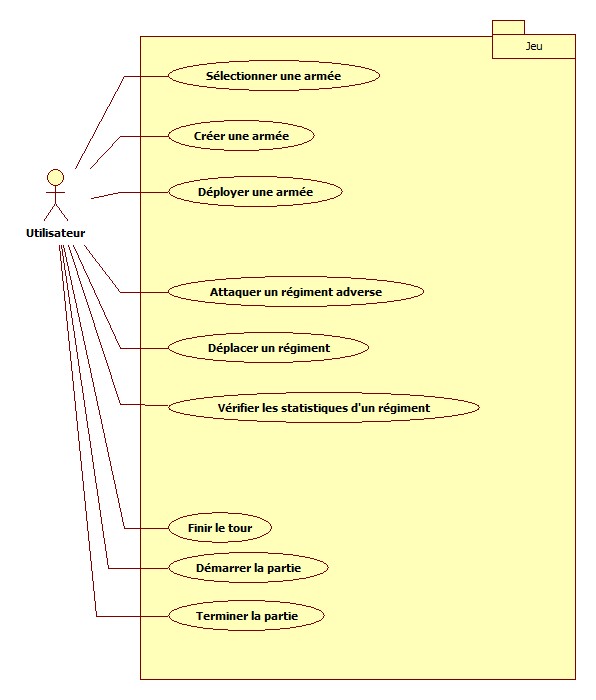
Le jeu proposé est le jeu de stratégie « Warhammer Battle » car il comprend une gestion d’interface 2D, une base de donnée simple à concevoir et oblige une gestion d’objets qui est en relation avec la théorie à appliquer dans le cours.

## Fonctionnement du jeu

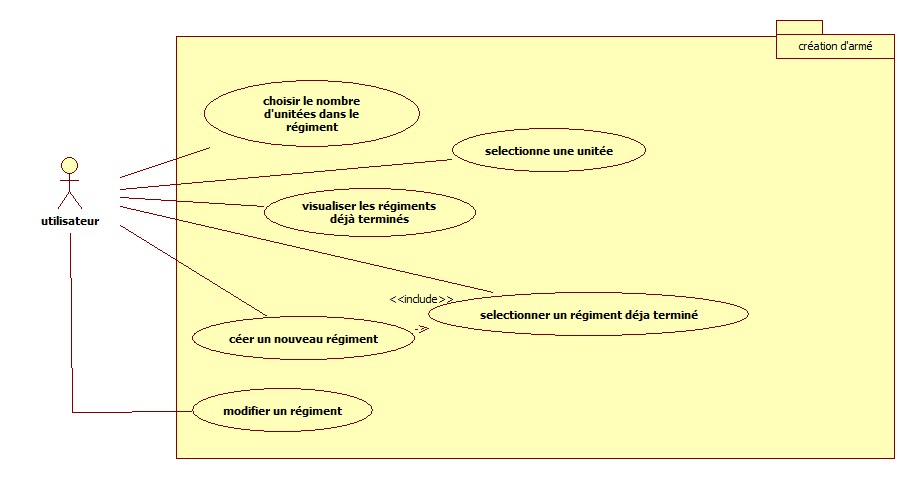
* Chaque joueur à un tour de jeu distinct.
* Le jeu oppose deux armées sélectionnées par les joueurs.
* Les armées sont formées en régiments que le joueur déplace comme les pièces d’un échiquier.
* Les régiments sont formés d’un seul type d’unités.
* Certains régiments possèdes des règles spéciales (IE : Haine des hauts elfes, aura de sang).
* Chaque régiment possède des unités avec des statistiques différentes.
* Les joueurs peuvent créer leurs armées à partir d’un système de points.
* Le jeu incorpore un système de morale, les unités mal positionnées ou en infériorité numérique peuvent subir un test de commandement et si elles échouent, partent en déroute.

## Fonctionnement planifié du système

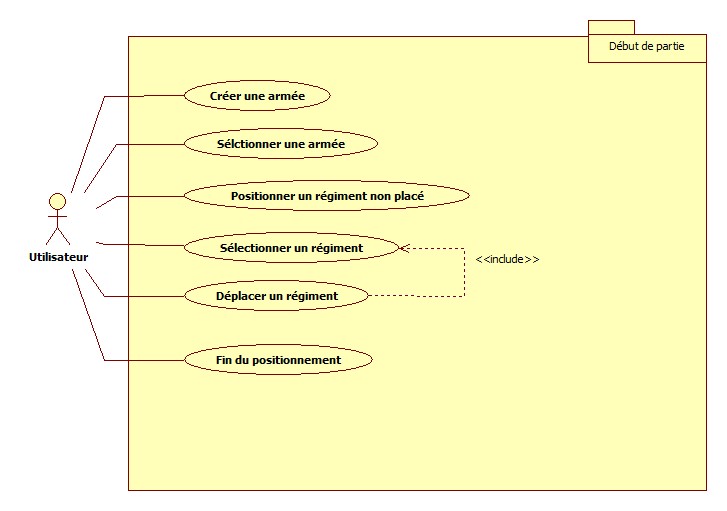
Avant le début de la partie l’utilisateur doit créer une armée ou prendre une armée préconfigurée et ensuite la positionner sur le terrain, régiment par régiment, au premier tour. La sélection d’un régiment affiche les statistiques du régiment et de ses unités et permet à l’utilisateur de passer à l’attaque ou de déplacer le régiment. L’utilisateur peut en tout temps terminer son tour. L’utilisateur peut en tout temps abandonner la partie.



## Module de gestion d’une armée

Le module de gestion d’une armée est complémentaire. Il permet de créer une armée à partir d’un système de points. L’utilisateur créé ses propres régiments à partir d’unités et choisi le nombre d’unités. 

## Phase de déploiement

L’utilisateur doit au départ créer une armée ou sélectionner une armée préconfigurée. Les joueurs entrent ensuite dans la phase de positionnement. Ils peuvent positionner un régiment qui n’est pas déjà sur la carte ainsi que déplacer un régiment qui était déjà positionné. Une fois tous les régiments déployés, les joueurs peuvent terminer la phase de déploiement.

# Faisabilité

## Technique

Les besoins matériels seront minimaux, Warhammer Battle est un jeu plateau très semblable aux échecs. Par conséquent, une dimension 3D serait grandement superflue pour le fonctionnement du jeu, une vue d’oiseau du champ de bataille et des icones en 2D pour représenter les régiments respecte le concept original. Les commandes en tour par tour plutôt qu’en temps réel diminuent grandement la complexité de la programmation, ce qui permettra à des ordinateurs bon marché d’opéré le programme d’une façon optimale. Le jeu devra être développé autour d’un langage permettant l’héritage et la programmation orienté objet. Ces méthodes de programmation seront facilement applicables avec notre expérience sur ces projets.

## Monétaire

Nous estimons être capables de maintenir 10 heures par semaines pendant 15 semaines. Ce qui fait un total d’approximativement 150 heures par personne. Nous sommes six à pouvoir maintenir ce rythme de travail pour un total de 900 heures. Nous avons également un chef de projet qui peut travailler 4 heures semaine, pour un grand total de 960 heures.

## Conclusion

Compte tenu que les contraintes techniques du programme sont minimes et que le programme traite de notions déjà vues, nous estimons le projet réalisable dans les temps établis.