

Rapport de conception : Projet Munchkin UTBM

Introduction / Présentation :

L'objectif de ce projet est de réaliser une version simplifiée du jeu Munchkin via une implémentation en Java. L'utilisation d'une interface graphique est également demandée.

Pour la réalisation de ce projet, nous avons simplifié les règles du jeu initial dû à l'importante quantité de règles et de spécifications présentent dans le jeu. Nos règles seront donc expliquées un peu plus loin.

Table des matières

Introduction / Présentation :	1
Règles :	2
Cartes du jeu :	2
Différentes phases de jeu	3
Diagrammes :	6
Use Case :	6
Diagramme de classes :	7

Règles :

Cartes du jeu :

On retrouve deux grandes classes de carte, la première représente les cartes de type **donjon**, ces cartes sont :

- Les monstres :

- Prof de Psycho niveau 1
- Prof de DataBase niveau 3
- Prof de Dev niveau 5
- Prof de Web niveau 7
- Prof de Maths niveau 9

Plus le niveau du monstre est élevé, moins il a de chance d'apparaître dans les cartes.

- Les malédictions :

- Potion :
 - Semestre supérieur = perte d'un niveau
 - Mauvaise note = fait perdre un objet au joueur
 - Partiel = Change la race en étudiant

- Les races :

- Etudiant Aucun bonus / Aucun malus
- Puant Peut avoir 6 cartes
- Geek 1 de bonus damage
- Alternant Vend ses objets 2 fois plus cher

- Les classes :

- Informaticien A 50% de chance de tirer une carte trésor en plus après avoir battu un monstre
- Génie Mécanique A 50% de chance de voler un objet lors d'un échange
- Génie Electrique Augmente le player damage par 2
- Génie Industriel N'est pas affecté par les cartes malédictions

La deuxième grande classe représente les cartes de type **trésor**, ces cartes sont :

- Les équipements :

- Couvre-chef :
 - Helmet glasses +3
 - Helmet for Bald +2
 - Helmet of Domination +1
- Armure :
 - Body Starcraft tshirt +3
 - Body Sweaty Shirt +2
 - Body Teemo sweat +1
- Arme :
 - Cyberweapon +1
 - Nvidia GPU book +2
 - GunPen +3

Les joueurs ne peuvent porter qu'un seul équipement par type au maximum

- **Bonus :**
 - Gagner de l'or (+6 crédits)

Pour gagner une partie, il existe 2 façons :

- Atteindre le niveau 10
- Avoir plus de 300 crédits

Chaque objet possède un nom et une valeur en crédits

Il est possible de vendre ses objets à un autre joueur au cours d'une partie mais seulement pendant son tour.

Lorsqu'un joueur meurt, il perd l'ensemble de son équipement et son niveau est réinitialisé à 1. Cette mort mène à un combat entre les joueurs qui souhaiteraient récupérer les équipements du défunt joueur. Le joueur mort ne peut quant à lui rejoindre la partie qu'au tour suivant et tire alors 2 cartes de chaque paquet.

Le jeu se déroule en de multiples phases comme suit :

[Différentes phases de jeu](#)

Phase 0 :

Tous les joueurs se voient distribuer 2 cartes donjon et 2 cartes trésor.

Les joueurs peuvent alors un à un poser leur carte équipement, crédits, classe ou race s'ils possèdent une de ces cartes. Dans le cas où un joueur possède plusieurs de ces cartes, il peut également en poser plusieurs.

Phase 1 :

Le premier joueur tire une carte donjon :

S'il tire une carte :

- Malédiction : Les effets de la carte s'applique
 - **Phase 2 :**
 - Le joueur peut poser une carte monstre de sa main et le combattre.
 - **Phase 3 :**
- Monstre : Attaque immédiatement
 - Combattre (monstre niveau inférieur)
 - Le joueur a alors une chance sur x de gagner dépendant de son équipement et du niveau du monstre.
 - Victoire : le joueur tire une ou deux carte(s) trésor et gagne un ou deux niveau(x) en fonction du niveau du monstre
 - Défaite : Perd de l'or et un équipement, s'il ne possède rien, alors le joueur meurt

Fuir (monstre niveau égal ou supérieur) :

- Peut demander de l'aide à un autre joueur pour retourner au combat (un joueur seulement peut aider le joueur en combat)
 - Accepte :
 - Aide en ajoutant sa force au combat
 - Victoire : pioche tous les deux une carte trésor et le joueur principal gagne un niveau
 - Défaite : Perd 6 crédits et un équipement, s'il ne possède rien, alors le joueur meurt
 - Trahison : Le joueur n'ajoute pas sa force au combat
 - Refuse :
 - Peut demander de l'aide à un autre joueur, mais ne peut plus demander de l'aide à quelqu'un à qui il a déjà demandé.
 - Le joueur à encore la possibilité de partir avec une chance sur x dépendant de son niveau

Fuir définitivement (Une chance sur x de fuir dépendant de son niveau)

- Réussi à fuir : passe au joueur suivant
- Echoue : perd de l'or et un équipement, s'il ne possède rien, alors le joueur meurt.
- Cartes race, classe :

- Le joueur peut choisir entre les poser directement ou les garder dans sa main

Phase 4 :

Si le joueur possède plus de 5 cartes, alors celui-ci doit jouer toutes les cartes qu'il a pour redescendre à une main de 5 cartes. Ou alors celui-ci peut donner ses cartes à un joueur de plus bas niveau. Dans le cas où plusieurs joueurs ont un niveau égal, alors le joueur choisi à qui il souhaite donner ses cartes.

Dans le cas où le joueur qui possède plus de 5 cartes est le joueur de plus bas niveau et qu'il ne peut pas les jouer, alors il doit les jeter à la défausse.

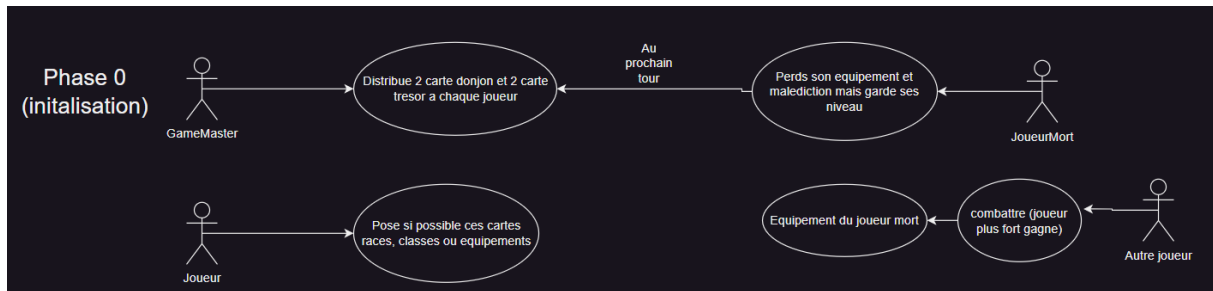
Disclamer / Reminder : les nains ont le droit d'avoir 6 cartes.

Diagrammes :

Use Case :

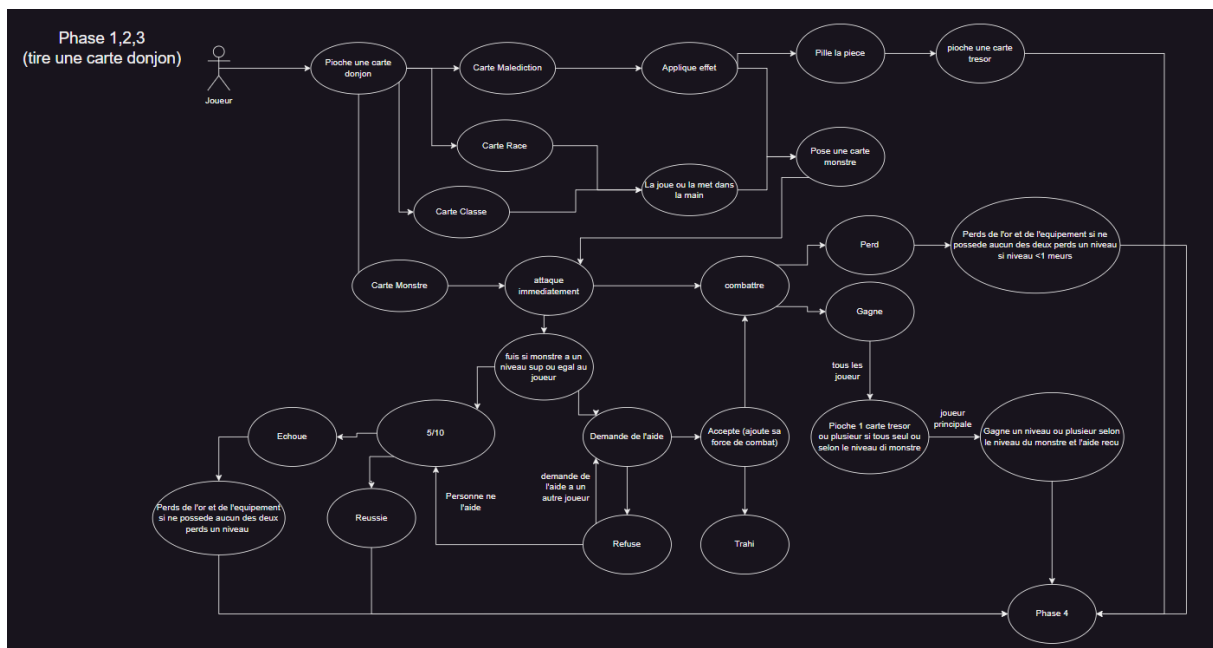
Dans notre cas, le use case est déjà complètement expliqué par les règles vues précédemment.

Le voici :

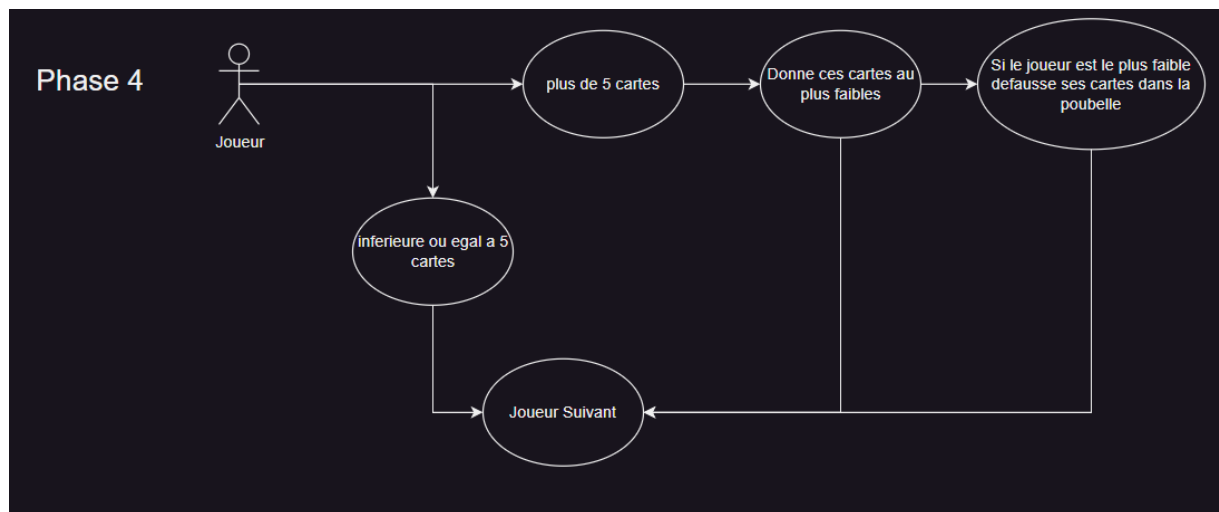


Cette partie du use case décrit donc la phase d'initialisation du jeu, le début de la partie comme décrit dans les règles précédemment.

Les phases 1, 2 et 3 sont décrites par cette partie du use case, toujours basé sur les règles explicitées précédemment.



La dernière et quatrième phase du jeu est donc la suivante :



[Diagramme de classe :](#)



On a choisi de créer une template pour les cartes de jeux (malédiction, monstre, etc.), on utilise les mêmes fonctions et méthodes, il y a quelques méthodes/attributs qui changent, mais globalement c'est le même modèle.

Concernant la pile on a préféré faire uniquement une classe puisqu'il n'y avait pas de différence probante concernant les méthodes/fonctions/attributs entre les piles.