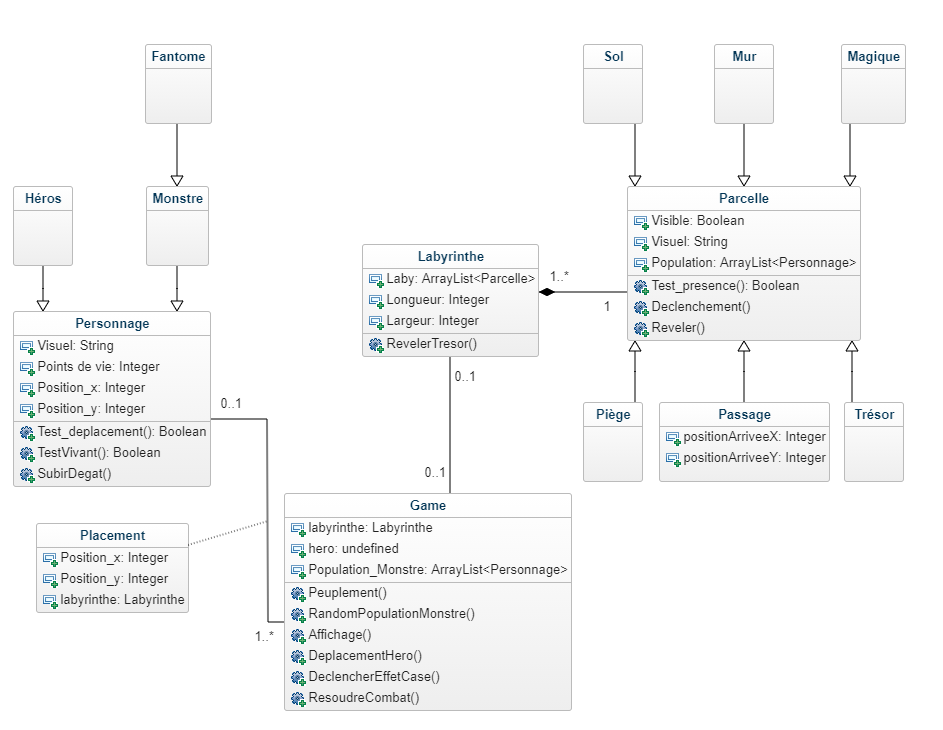
Usecase Diagram :

Class Diagram :

Titre : Déplacement

Branche permettant à l’utilisateur de déplacer le héros dans le labyrinthe.

Acteur : Joueur

Précondition : le héros est sur une case valide (position initiale)

Scénario nominal :

1. Le joueur choisit la direction (z,q,s,d)
2. La case est vérifiée :

2.1) La case choisie est une case où se trouve déjà un monstre ou un piège. Le héros perd un point de vie.

2.1.1) Il ne reste plus de point de vie au héros et il meurt. Fin de la partie.

2.1.2) Malgré la perte de point de vie, il en reste encore au héros. Le monstre ou le piège disparait, le héros se retrouve sur cette nouvelle case et il ne se passe rien jusqu’à la prochaine action du joueur. Retour au scénario nominal 1).

2.2) La case choisie est vide. Le héros est maintenant sur la nouvelle case. Il ne se passe rien jusqu’à la prochaine action du joueur. Retour au scénario nominal 1).

2.3) La case choisie est un mur. Le héros est bloqué sur sa case actuelle. Il ne se passe rien jusqu’à la prochaine action du joueur. Retour au scénario nominal 1).

2.4) La case choisie est la case trésor. Le héros gagne la partie. Fin de partie.

2.5) La case choisie est une case spéciale/magique. Le héros est placé sur cette nouvelle case. Le trésor est révélé sur le labyrinthe. Il ne se passe rien jusqu’à la prochaine action du joueur. Retour au scénario nominal 1).

2.6) La case choisie est une case de passage(téléportation). Le héros est placé sur la case de téléportation. Il ne se passe rien jusqu’à la prochaine action du joueur. Retour au scénario nominal 1).

Scénarios exceptionnels :

1. Le joueur décide de quitter le jeu
2. Problèmes d’affichage : Il y a eu un bug ou une erreur d’affichage du programme

Titre : Générer Labyrinthe

Branche permettant de générer graphiquement le labyrinthe souhaité

Acteur : Système

Précondition : Le labyrinthe créé par le joueur ou depuis un fichier doit être valide

Scénario nominal :

1. Sauvegarde des différentes cases et de leur emplacement
2. Lecture de la visibilité des cases

2.1) Si la case est dite « visible », son visuel sera affiché par la suite (comme les murs, sols…)

2.2) Si la case est dite « cachée », son visuel ne sera pas affiché par la suite (case trésor, piège…)

1. Affichage du labyrinthe avec les visuels de chacune des cases

Scénarios exceptionnels :

* Problème de génération : 2 cases se superposent alors que ce n’est pas prévu (par exemple un mur et le trésor)
* Problème d’affichage : une case avec un visuel caché (piège, trésor…) est tout de même affichée

Titre : Générer un labyrinthe à partir d’un fichier

Branche permettant de générer le labyrinthe souhaité à partir d’un fichier

Acteur : Système

Précondition : Le labyrinthe transcris dans le fichier doit être valide.

Scénario nominal :

1. Lecture de la visibilité des cases

1.1) Si la case est dite « visible », son visuel sera affiché par la suite (comme les murs, sols…)

1.2) Si la case est dite « cachée », son visuel ne sera pas affiché par la suite (case trésor, piège…)

1. Affichage du labyrinthe avec les visuels de chacune des cases

Scénarios exceptionnels :

Le fichier est illisible à cause d’une des erreurs suivantes :

* La première ligne du fichier ne concerne pas les dimensions du labyrinthe.
* 2 personnages sont sur la même case.
* Plusieurs instructions de création de case pour une même position.

Le système signale à l’utilisateur que le fichier est illisible, puis lui demande de saisir le nom d’un autre fichier.

Titre : Le joueur crée son labyrinthe

Branche permettant au joueur de créer son propre labyrinthe

Acteur : Joueur

Prérequis : L’utilisateur a choisi de créer son propre labyrinthe.

Scénario nominal :

1. L’utilisateur indique les dimensions du labyrinthe
2. L’utilisateur place les murs
3. L’utilisateur tape fin pour passer à la suite
4. L’utilisateur choisit son point de départ
5. L’utilisateur choisit le nombre de monstres présent dans le labyrinthe
6. L’utilisateur choisit l’emplacement de l’arrivée
7. L’utilisateur choisit l’emplacement de la case magique
8. L’utilisateur choisit l’emplacement de la case piège
9. L’utilisateur choisit l’emplacement de l’entrée et de la sortie du passage.

Scénario alternatif :

A -Ce scénario commence à l’étape 1 si l’utilisateur indique des coordonnées non valides

1-l’interface affiche que les données entrées sont invalides

2-retour étape 1

B -Ce scénario commence à l’étape 2 si l’utilisateur indique des coordonnées non valides

1-l’interface affiche que les données entrées sont invalides

2-retour étape 2

C -Ce scénario commence à l’étape 4 si l’utilisateur indique des coordonnées non valides

1-l’interface affiche que les données entrées sont invalides

2-retour étape 4

D -Ce scénario commence à l’étape 5 si l’utilisateur indique des coordonnées non valides

1-l’interface affiche que les données entrées sont invalides

2-retour étape 5

E-Ce scénario commence à l’étape 6 si l’utilisateur indique des coordonnées non valides

1-l’interface affiche que les données entrées sont invalides

2-retour étape 6

F -Ce scénario commence à l’étape 7 si l’utilisateur indique des coordonnées non valides

1-l’interface affiche que les données entrées sont invalides

2-retour étape 7

G -Ce scénario commence à l’étape 8 si l’utilisateur indique des coordonnées non valides

1-l’interface affiche que les données entrées sont invalides

2-retour étape 8

H -Ce scénario commence à l’étape 4 si l’utilisateur indique des coordonnées non valides

1-l’interface affiche que les données entrées sont invalides

2-retour étape 9

Scénario exceptionnel :

Ce scénario commence après l’étape 7 dans le cas où le labyrinthe est plein

1-Les étapes 7 et 8 sont sautées

Postconditions : le jeu commence