

## UE Projet Logiciel

Le but de ce projet est d'étudier la conception objet sur un problème de taille plus importante que ce dont vous avez eu l'habitude en TP. Il s'agit également d'appliquer les principes et techniques de conception vus en POO et COO lors des semestres 4 et 5.

Au cours des quelques séances de TD une partie du travail d'analyse sera réalisée, l'accent sera notamment mis sur la conception. Les séances de TP seront dans un premier temps consacrées à la découverte de nouveaux outils ou aspects du langage : XML, les "properties" et ant. Dans un second temps, les séances de TP seront destinées à vous permettre de disposer d'un temps de développement pour votre projet avec un enseignant encadrant. Il est clair que le temps offert par ces créneaux sera insuffisant pour la réalisation de son projet.

Cette année, nous vous proposons de réaliser un jeu d'aventure simple qui consiste en un personnage, le héros, qui se promène à travers un monde fini et interagit dans ce monde.

### Aperçu du jeu

Le jeu est constitué d'une carte qui représente le monde fini dans lequel le héros évolue. La carte est constituée d'unités de terrain qui peuvent être des champs, des forêts, des montagnes, etc. Le franchissement de chaque terrain requiert une certaine dépense d'énergie pour le personnage qui veut le traverser. Cette dépense est dépendante du type de terrain. Sur les unités de terrain sont présents d'autres personnages ou des choses.

Les personnages sont caractérisés par la possession d'une certaine énergie. Le héros est un personnage particulier qu'incarne le joueur du jeu d'aventure. C'est donc le joueur qui, à travers une interface texte, choisit les actions du héros. Lors de son parcours à travers le monde, le héros rencontrera d'autres personnages : des méchants avec qui il devra se battre, des guérisseurs qui lui permettront d'augmenter son énergie contre de l'or. Les méchants agissent de manière aléatoire et les guérisseurs restent sur place. La liste des personnages n'est pas exhaustive, il devra être très simple pour un utilisateur de votre API d'en ajouter un autre.

Si le héros décide d'interagir avec un méchant alors il devra se battre. Les batailles sont organisées simplement : chacun des deux combattants perd autant d'énergie que la valeur de force de son adversaire. Il n'y a pas de riposte possible. Lorsqu'un méchant meurt (son énergie est devenue nulle ou négative), l'or qu'il possédait est déposé sur place sous la forme d'une pépite de la valeur du montant d'or qu'il possédait.

Un personnage peut utiliser des choses disséminées sur le terrain, celles-ci sont par exemple des pépites et de la nourriture. Les personnages peuvent ramasser ces choses. Utiliser des pépites signifie les transformer en or et ajouter cet or au capital du joueur, cet or peut ensuite servir auprès du guérisseur (par exemple). Utiliser la nourriture c'est la consommer pour augmenter son énergie. De la même manière que pour les personnages, il peut y avoir d'autres objets qu'il doit être possible d'intégrer facilement.

A chaque tour chaque personnage effectue une et une seule action. Par exemple le héros pourra choisir soit de se déplacer, soit de rester sur place, soit d'interagir avec un autre personnage présent sur la même unité de terrain, soit de ramasser une chose parmi celles disponibles, soit d'utiliser un objet qu'il possède, etc. Il en est de même pour les autres personnages, on supposera pour commencer que le guérisseur reste sur place à se reposer ou attendre qu'on le sollicite, quant aux monstres, ils peuvent ne rien faire, ou se déplacer ou attaquer un autre personnage rencontré. Cette fois encore, il doit être possible d'ajouter de nouvelles actions non encore prévues. De plus, les actions possibles pour un type de personnage donné doit facilement pouvoir évoluer.

### Précisions

- Les paramètres de l'application seront externalisés afin de pouvoir être facilement modifiés.
- La gestion du héros par le joueur se fait au travers d'un menu offrant à choisir les différentes actions possibles. Ce menu doit pouvoir facilement être étendu si le jeu évolue.
- On veut pouvoir supporter plusieurs langues pour l'interface, notamment pour le menu mentionné ci-dessus.

- La carte du monde du jeu doit pouvoir être générée aléatoirement, éventuellement en paramétrant les pourcentages de chaque type de terrain. On ne vérifiera pas la viabilité de cette carte (que toute zone est accessible par exemple).

Il doit notamment être possible de créer un monde (c'est-à-dire une carte avec ses objets et personnages) à partir de données décrites dans un fichier au format XML.

- Le plateau du jeu est fini et constitué de parcelles ou cases. Celles-ci doivent pouvoir être de n'importe quelle forme. Cependant, toutes les parcelles du plateau ont la même forme. Chaque case dispose de cases voisines mais la topologie de la carte (càd l'organisation des unités de terrain entre elles – leur voisinage) ne doit pas être imposée et connue a priori.

Vous implémenterez le cas où les cases sont de forme rectangulaire et ont chacune 4 voisines en Nord, Sud, Est et Ouest.

Bien sûr le code que vous écrivez doit être indépendant de cela.

- Aucun affichage du plateau n'est demandé.
- Le source et la javadoc devront être rédigés en anglais.
- Les étudiants qui souhaitent développer une interface graphique pour leur projet peuvent le faire, néanmoins il est **obligatoire** que le projet puisse fonctionner en mode console (donc sans l'interface graphique).
- Travaillez régulièrement !

## Attention

Le but du projet est la conception objet, pas le jeu, inutile donc d'en faire trop d'un point de vue ludique si cela se fait aux dépens d'une bonne conception.

## Document en Anglais pour le lundi 4 mai

Vous devez rédiger une page recto-verso double interligne en anglais sur le projet. Ce document présentera le projet, son contexte ainsi que le travail que vous avez réalisé, que ce soit en décrivant le "produit" réalisé et/ou en décrivant votre démarche de travail au cours de la réalisation ou toute autre contenu que vous jugerez pertinent en relation avec le projet.

Ce travail est à rendre sous la forme d'un PDF à votre enseignante d'anglais du S5 par mail au plus tard pour le lundi 4 mai.

Vous pouvez (et devez) solliciter par mail l'aide de votre enseignante d'anglais sur ce travail pendant toute la période qui précède ce rendu.

## Ce qui est à rendre pour le lundi 11 mai

Le projet se fait en **binôme**. Vous devez rendre une archive `tar` dont la structure et le contenu seront détaillés ultérieurement.

Vous rendrez un rapport au format PDF présentant votre travail et contenant au minimum l'organisation de vos paquetages et classes (diagrammes UML) et décrivant la structure des fichiers XML.

Vous serez évalué sur la qualité de la conception (découpage du projet en objets par exemple), la qualité de la documentation, la clarté de votre code, la bonne exécution du jeu, etc.

Note : le respect de ces consignes sera pris en compte, entre autre toute archive qui ne se désarchive pas correctement impliquera la non-correction du projet.

## Ce qui sera à rendre pour le lundi 18 mai

Vous serez informés d'une modification à faire sur le projet dès le lundi 11 mai. Vous devrez alors rendre une nouvelle archive avec le projet modifié et un fichier `modif` décrivant les modifications qui ont été nécessaires.

Une interrogation orale individuelle aura lieu la semaine du 18 mai.