



# Rapport TOB : Seven Quest

COUX Juliette  
FELLAHI Medhi  
GIRARD Antoine  
PAUL Julie  
PICCO Enguerrand  
WEISBECKER Lisa

Département Sciences du Numérique - Première année  
2021-2022

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Fonctionnalités</b>	<b>3</b>
<b>3</b>	<b>Découpage application</b>	<b>3</b>
<b>4</b>	<b>Diagramme de classe</b>	<b>4</b>
<b>5</b>	<b>Choix de conception, problèmes rencontrées et solutions</b>	<b>6</b>
<b>6</b>	<b>Organisation</b>	<b>6</b>
<b>7</b>	<b>Améliorations possibles</b>	<b>7</b>

## Table des figures

1	Scenario . . . . .	3
2	Diagramme de classe général . . . . .	4
3	Diagramme de classe Personnage . . . . .	5
4	Diagramme d'un scénario . . . . .	6

# 1 Introduction

Nous avons choisi pour ce projet de créer un jeu vidéo de type jeu de rôle : Seven Quest. Ce jeu consiste à faire jouer un personnage avec des caractéristiques propres (selon sa race : Guerrier ou Magicien). Ce personnage va suivre un scénario, une aventure au cours de laquelle il va devoir faire des choix et effectuer des combats. Ces choix influencent la suite de l'aventure mais aussi ses statistiques. Ces choix le mèneront à une fin personnalisée.

## 2 Fonctionnalités

Dans un premier temps, nous avons voulu que notre jeu puisse être joué sur un terminal à la fin de l'itération 1. Nous avons également souhaité que l'utilisateur puisse créer son personnage, avoir accès aux statistiques et aux informations du personnage, de pouvoir lancer un combat et enfin de pouvoir faire des choix. Nous avons fait également du dérisquage sur l'interface pour les lancers de dés car nous savions que cela allait prendre du temps.

Ensuite, pour la deuxième itération, nous avons voulu voir le jeu sur un écran, et non plus sur le terminal. Nous avons souhaité voir les statistiques et informations, pouvoir voir des choix, pouvoir accéder à son inventaire et créer son personnage. En résumé, nous avons à nouveau dérisqué les affichages graphiques, sans pour autant qu'elles soient fonctionnelles. Nous avons également voulu obtenir le scénario final de notre jeu. Voici le scénario développé :

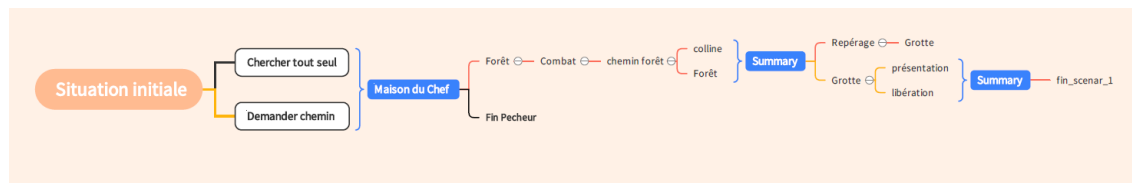


FIGURE 1 – Scénario

Nous avons consacré la dernière itération à implanter un jeu qui fonctionne. Les objectifs étaient de pouvoir lancer une partie, créer son personnage, changer son inventaire, faire évoluer les statistiques du héros, pouvoir faire des choix et pouvoir faire un combat.

## 3 Découpage application

Nous avons choisi de généraliser les classes : Pour coder notre jeu, nous avons réfléchi à plusieurs classes qui représentent les éléments essentiels de notre jeu.

1. classe abstraite Personnage : cette classe permet de caractériser tous les personnages de notre jeu (Guerrier, Magicien, ennemi, PNJ (personnage non-joueur)).
2. classe PersonnageGuerrier : classe concrète pour la race Guerrier.
3. classe PersonnageMagicien : classe concrète pour la race Magicien.
4. classe PersonnageBandit : classe concrète pour un bandit, un ennemi du héros.
5. classe Jouer : permet de lancer un jeu : création d'une fenêtre, accès au menu puis création du personnage et enfin début du scénario.
6. classe Combat : permet de faire combattre deux personnages.
7. classe Equipement : élément que peut récupérer le héros pour gagner des compétences.
8. classe Scenario : elle permet de proposer un choix au personnage et traite son choix. Nous aurons un grand nombre de classes de ce type pour notre jeu, afin de proposer une aventure avec des obstacles variés à l'utilisateur.

9. classe Interface... : permet de créer le panel correspondant selon les actions de l'utilisateur (choix, accès à l'inventaire, combat, ...). Ce panel viendra se mettre sur l'interface de base créer lorsqu'on lance Jouer.
10. classe Choix : permet de traiter un choix.
11. classe Inventaire : créer l'inventaire du personnage.

## 4 Diagramme de classe

Voici différents diagrammes de classe. Le premier est un diagramme général qui permet d'avoir une vue globale des classes quand on lance une partie.

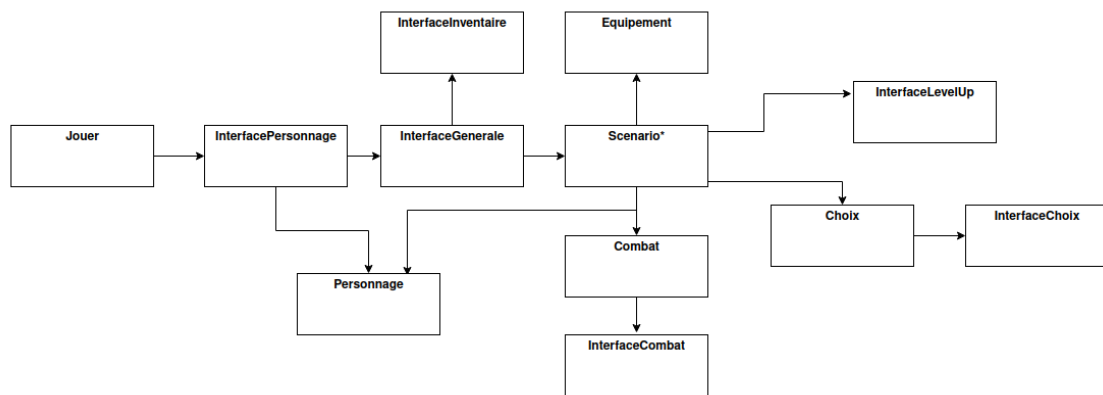


FIGURE 2 – Diagramme de classe général

Ce diagramme présente la classe abstraite Personnage et ses extensions.

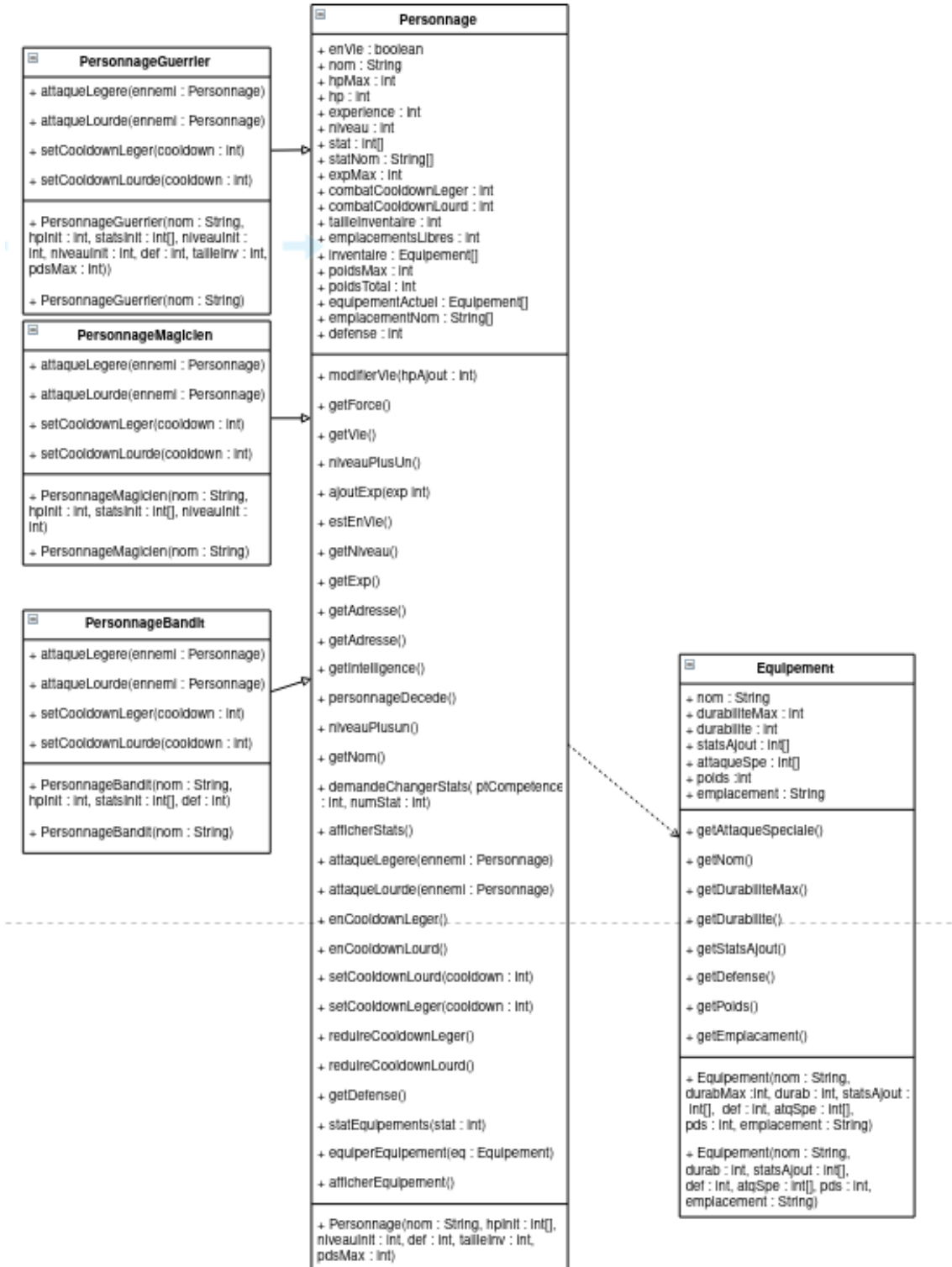


FIGURE 3 – Diagramme de classe Personnage

Enfin, voici le diagramme qui rassemble les différentes relations qu'à un scénario en particulier. Nous pouvons voir qu'il appelle beaucoup de classes différentes. Nous avons essayé de tout centraliser dans la classe Scénario.

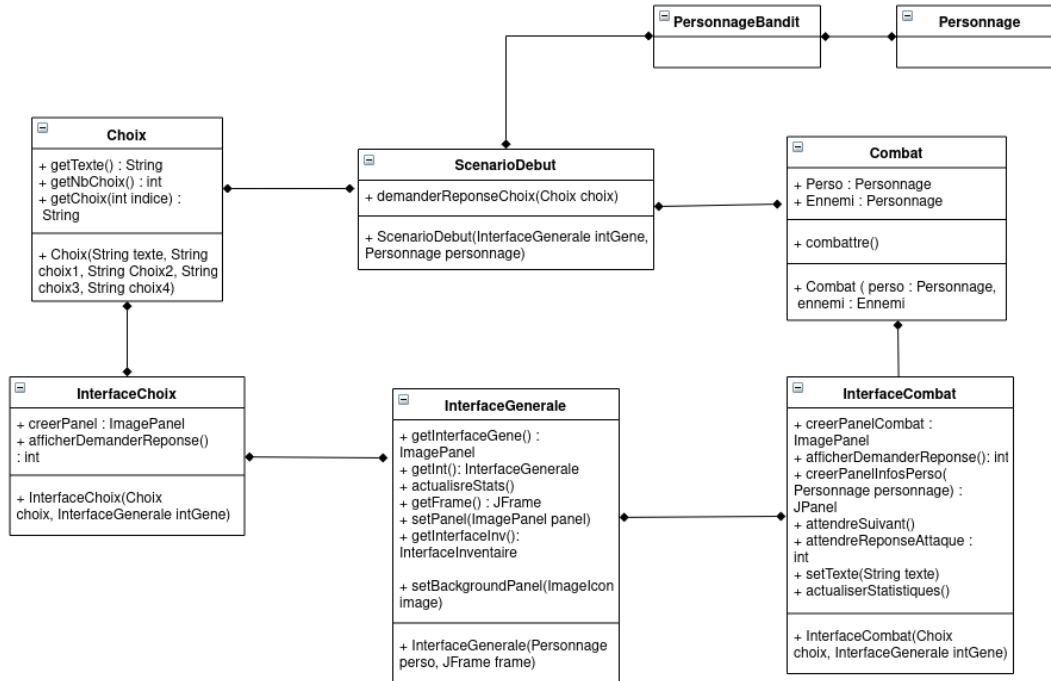


FIGURE 4 – Diagramme d'un scénario

## 5 Choix de conception, problèmes rencontrés et solutions

Nous avons choisi de faire une classe abstraite Personnage qui nous permet de créer plusieurs classes par la suite de nos différents personnages du jeu. Nous avons réalisé que les méthodes étaient similaires pour tous les personnages, qu'il soit ennemi ou le personnage principal. Cette classe abstraite nous évite de créer plein de classes en plus, similaires à celle-ci.

Nous avons rencontré des problèmes pour créer nos interfaces graphiques. En effet, le problème était de n'avoir qu'une seule fenêtre et d'y afficher différentes interfaces en fonction de l'étape dans le jeu (menu, choix du personnage, jeu, combat, choix...). On a donc une classe Jouer qui crée une fenêtre et qui récupère les panels du menu, du choix de personnage et de du jeu. Le panel du jeu est fait à partir d'une interface générale qui a des composants constants (statistiques du personnage, bouton inventaire) et qui a un panel qui peut changer en fonction de l'étape dans le jeu (combat, choix, inventaire, option...).

Un autre problème que l'on a rencontré est l'imbrication de nombreux scénario et la possibilité d'ajouter des étapes de scénario. Pour cela on a construit des scénarios qui peuvent s'imbriquer en se lançant les uns les autres.

## 6 Organisation

Lors des séances de méthodes agiles, nous avons pu échanger nos idées et définir la manière dont nous allons fonctionner pendant ce projet. Nous avons d'abord listé toutes les étapes importantes pour avancer petit à petit sur notre jeu puis relevé celles que nous réaliserons pour la première

itération. Les objectifs étaient :

- lister les classes/interfaces et avoir une idée du contenu
- faire le diagramme UML
- coder les classes de base.

Nous nous sommes mis d'accord sur les classes à coder pour avoir la même vision du jeu et de comment il allait être implanté. Au niveau des fonctionnalités, notre objectif était de pouvoir jouer au jeu à partir d'un terminal, c'est-à-dire de créer son personnage, de pouvoir faire un choix et de pouvoir combattre un autre personnage.

Nous avons ensuite continué à implanter nos classes, celle du combat, de l'inventaire et du personnage, ainsi que du développement du scénario. Nous avons en parallèle commencé les interfaces graphiques notamment : le menu, l'interface d'une étape classique et l'interface de la création d'un personnage. L'objectif est d'avoir un premier jet de nos interfaces graphiques et de commencer à lier toutes les interfaces pour avoir un jeu fonctionnel.

Enfin, lors de la dernière itération, nous nous sommes focalisés sur l'interface graphique en réalisant des classes simples que nous pouvions réutiliser. Nous avons donc implanter le menu, l'interface de création de personnage et celle des scénarios.

Nous avons aussi créer les classes de scénarios, qui sont liées entre elles et qui dépendent des choix de l'utilisateur.

## 7 Améliorations possibles

Nous avons réussi à implanter un jeu fonctionnel, avec une prise en main facile et les fonctionnalités que nous souhaitions. Il serait possible d'améliorer notre jeu.

Par exemple, nous pourrions exploiter beaucoup plus les statistiques du héros, chose que nous ne faisons pas dans notre jeu (par exemple, avoir des attaques plus ou moins fortes selon les statistiques du personnage). De plus, nous pourrions allonger le scénario et avoir bien plus de branches et créer des fins différentes. Nous pourrions également ajouter de la musique, pour être plus immerger dans la partie. Nous n'avons pas eu le temps d'intégrer nos animations des lancers de dés dans notre jeu, il pourrait être intéressant de les mettre lors des combats. Enfin, nous pourrions ajouter des classes à nos armes, pour augmenter plus ou moins leur dégats.