

# **Livrable D3.2 :**

## **Rapport de développement et de tests**

**Projet : Walkyries**

**Groupe 1.4**



PRADES Mickaël

GODART Quentin

DELARBRE Morgan

BOGACZYK Pierre

## Introduction:

L'application que nous voulions développer était un jeu de rôle. Nous voulions développer un jeu avec un système de déplacement en deux dimensions mais nous avons dû revoir nos ambitions à la baisse car, après consultation d'un membre du comité de pilotage, celui-ci nous a éclairé, en cours de projet, sur la difficulté que nous allions rencontrer et sur la faible maîtrise du langage JAVA que nous avions. Nous avons donc changé le mode de jeu pour un jeu à boutons, c'est-à-dire que l'utilisateur doit utiliser les boutons pour jouer, comme un jeu avec un style *visual novel*.

## État du développement :

Aujourd'hui l'application est partiellement fonctionnelle, nous avons eu différents problèmes avec l'interface homme machine, dont certains restent encore non résolus, mais aussi des problèmes en terme de connaissances pour l'interaction entre les boutons de l'IHM et le code derrière l'application. Dans l'état actuel, l'utilisateur peut lancer l'application, commencer une nouvelle partie et créer un personnage mais ne peut exercer un combat ou encore interagir avec des personnages non joueurs comme nous avons défini le jeu au départ.

Le lancement de l'application s'effectue depuis la classe Main qui lance un objet MenuHCI.

## Fonctions réalisées :

Les fonctions complètement réalisées sont les fonctions liées à la création de personnage. Maintes autres fonctions sont partiellement fonctionnelles à cause de la partie de l'IHM, comme le système de combat, l'utilisation des caractéristiques du personnage, le système d'expérience du personnage ou encore le changement de zone. Cependant nous avons dû abandonner des fonctions comme le déplacement du personnage en deux dimensions ou le système de scénario avec les différents choix qui interagissent avec le destin du personnage à cause de notre manque de connaissance et de maîtrise du langage, ces fonctions étant trop complexes pour notre groupe.

## Difficultés et solutions :

Nous avons rencontré un problème pour permettre à l'utilisateur d'interagir avec le programme depuis l'IHM, nous avons résolu cela en ajoutant un attribut de type Character dans GameHCI permettant de lancer les méthodes d'autres classes et d'accéder aux informations de jeu.

## Rapport de tests :

Les classes testées sont les classes Characteristic et ExperienceBar. Pour la classe Characteristic nous avons écrit des tests pour essayer la méthode setCharacteristic, pour savoir si la méthode fonctionnait et donnait la valeur que l'on voulait à l'attribut en question. Nous avons ensuite écrit des tests pour savoir si la méthode AddExperience d'ExperienceBar ajoute bien l'entier en paramètre à l'attribut currentExperienceAmount.