

# Rapport sur l'Interface Homme Machine du Jeu Donjons & Dragons

## Présentation du projet Donjons et Dragons :

Dans le cadre de notre projet de programmation en Java, nous avons entrepris de créer une version informatique du célèbre jeu de rôle Donjons et Dragons (D&D). Le jeu a pour objectif d'immerger les joueurs dans un univers fantastique où ils incarnent un courageux aventurier cherchant à vaincre un redoutable dragon. L'expérience de jeu est construite autour de l'exploration d'un labyrinthe mystérieux, la rencontre de bonus pour renforcer le personnage, et enfin, l'ultime confrontation contre le redoutable dragon géré par une IA.

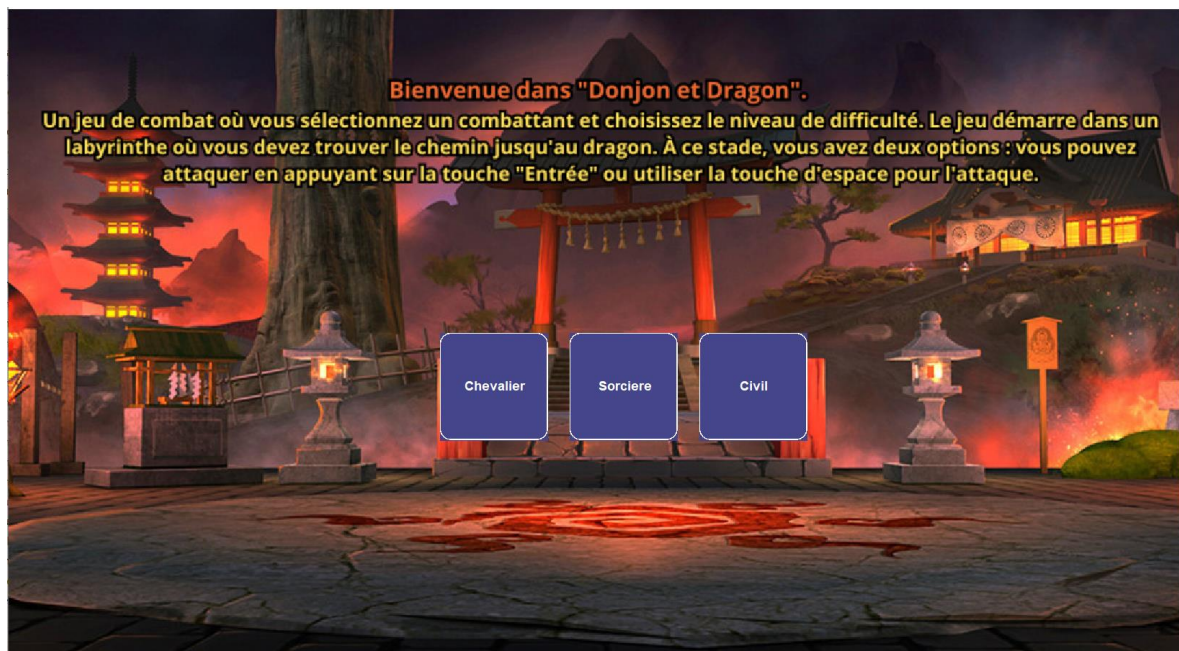
## Objectif de Conception de l'interface graphique :

L'objectif principal de la conception de l'interface graphique pour le jeu était de la rendre facile à utiliser, mettant l'accent sur l'accessibilité au détriment d'une beauté graphique élaborée. L'objectif était de créer une interface intuitive qui permettrait aux joueurs de se plonger rapidement dans l'action du jeu sans être surchargés par des éléments graphiques complexes.

## Description des interfaces graphiques :

L'étape initiale de notre application se déroule sur la page d'accueil, où les utilisateurs sont accueillis par une brève présentation du jeu ainsi que ses règles. L'utilisateur aura également la possibilité de choisir votre personnage parmi trois options : le chevalier, la sorcière et combattant humain.

La palette de couleurs choisie pour l'interface graphique est basée sur des tons rouges, oranges et marrons, évoquant l'ambiance de volcans et le contexte médiéval d'un donjon. Les icônes ont été conçues de manière sobre, évitant d'alourdir l'interface avec des éléments graphiques superflus, tout en conservant une esthétique cohérente avec le thème du jeu.



Ensuite le joueur est invité à sélectionner le niveau de difficulté qui leur convient parmi les options suivantes : facile, difficile ou moyen.



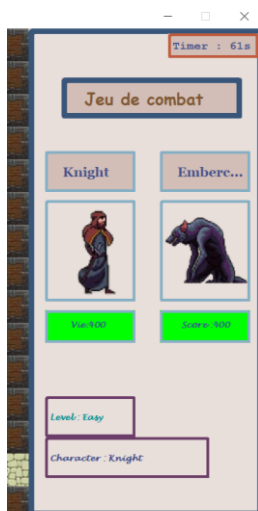
Le joueur se retrouve plongé dans un labyrinthe où il devra explorer pour collecter des bonus avant de se confronter au redoutable dragon. Cette confrontation se déroulera sous la forme d'un combat, où le joueur aura la possibilité d'attaquer et de se défendre, tout comme le dragon.



Pendant le combat, la santé du joueur peut diminuer s'il subit des attaques de la part du dragon, mais il peut également la restaurer en collectant les bonus dispersés dans le labyrinthe.

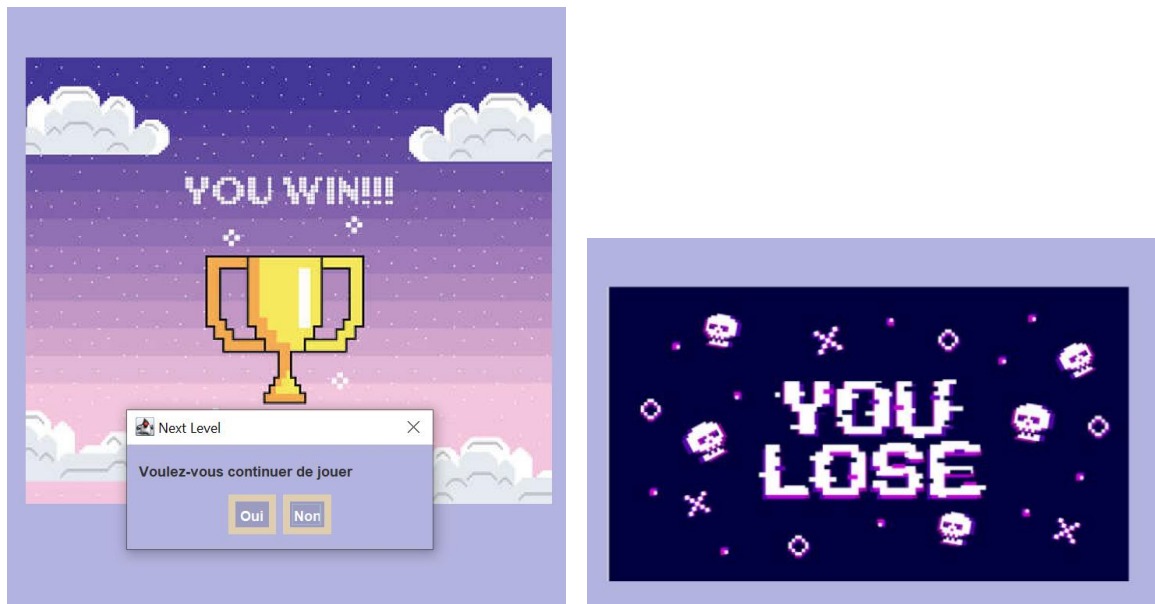


Dans la capture d'écran ci-dessous, vous trouverez des informations essentielles, notamment le niveau de la partie (facile, moyen ou difficile), le personnage choisi par l'utilisateur (chevalier, sorcière ou l'humain), ainsi que les niveaux de santé et de score du combattant. Il est important de noter que si la vie du combattant du joueur atteint zéro, le joueur perd la partie, tandis que si la vie du dragon atteint zéro, le joueur remporte la victoire.





À la fin du combat contre le dragon, le joueur peut soit perdre contre le dragon, soit remporter la victoire. En cas de victoire, il accède au niveau suivant avec des défis accrus, affrontant des dragons plus puissants tout en conservant son niveau de vie du tour précédent.



### Choix de la librairie Java pour l'IHM :

Nous avons opté pour Java Swing pour développer l'interface utilisateur de notre jeu. Swing, avec ses riches bibliothèques de composants graphiques, nous a permis de créer une interface utilisateur attractive et réactive. Cette bibliothèque est également bien intégrée avec le reste de l'écosystème Java, ce qui facilite le développement et la maintenance du jeu.

### Concepts d'ergonomie appliqués :

L'ergonomie a été au cœur de notre réflexion lors de la conception de l'interface utilisateur. Nous avons consciemment travaillé à rendre l'interface aussi intuitive que possible, en organisant de manière claire les menus et en proposant des commandes simples. Notre objectif était de minimiser la courbe d'apprentissage pour les nouveaux joueurs, tout en offrant une expérience de jeu fluide et agréable. Pour atteindre cet objectif, nous avons appliqué des concepts d'ergonomie tels que la loi de gestalt, assurant une disposition visuelle cohérente et une compréhension naturelle des éléments interactifs.

Nous avons regroupé visuellement des éléments similaires selon la loi de la Similarité, positionné des éléments connexes proches les uns des autres avec la loi de la Proximité, et assuré une disposition fluide en suivant la loi de la Continuité. La loi de la Clôture a délimité visuellement des zones spécifiques, et la recherche de simplicité a éliminé les distractions.

inutiles. Ces principes ont abouti à une interface intuitive, réduisant les efforts cognitifs des joueurs pour une expérience de jeu plus agréable.

### **Intégration de l'IHM avec la logique du jeu :**

L'intégration de l'interface utilisateur avec la logique du jeu a été soigneusement planifiée. Chaque action dans l'IHM se traduit par des réactions correspondantes dans le jeu, assurant une expérience immersive. Cette synchronisation entre l'IHM et la logique du jeu contribue à une expérience utilisateur cohérente et engageante.

### **Conclusion :**

En résumé, notre conception d'Interface Homme-Machine (IHM) pour le jeu a privilégié l'ergonomie et l'accessibilité. À travers Java Swing, nous avons créé une interface réactive et attrayante, tandis que l'intégration harmonieuse avec la logique du jeu assure une expérience utilisateur fluide. En appliquant des principes d'ergonomie, notamment la loi de gestalt, nous avons optimisé la disposition visuelle des éléments interactifs. Le résultat est une IHM intuitive, favorisant une expérience de jeu agréable et engageante pour tous les utilisateurs.