

Introduction

Le projet consiste en la création d'un jeu de combat en mode console où deux entités génériques, le "Guerrier du Feu" et le "Sombre Lutin", s'affrontent à l'aide d'attaques prédéfinies. L'objectif est de permettre au joueur de choisir une attaque à chaque tour, d'infliger des dégâts à l'adversaire et de remporter le combat en réduisant les points de vie (PV) de l'adversaire à zéro. Le jeu sera exécutable en ligne de commande grâce à Node.js.

Déroulement du Combat :

Le joueur et l'ordinateur (Sombre Lutin) s'affrontent tour à tour.

À chaque tour, le joueur choisit une attaque parmi les quatre attaques disponibles pour le Guerrier du Feu.

L'ordinateur choisit une attaque pour le Sombre Lutin de manière aléatoire.

Caractéristiques des Attaques :

Chaque attaque a une puissance de frappe spécifique (en PV) et une probabilité de toucher.

Les attaques peuvent infliger des dégâts ou soigner le combattant selon le cas.

Liste des Attaques Disponibles

Attaque 1 :

Nom: Frappe Rapide

Puissance: 10 PV

Précision: 50% (1 chance sur 2 de toucher)

Attaque 2 :

Nom: Soin Léger

Puissance: Soigne 15 PV

Précision: 33.33% (1 chance sur 3 de toucher)

Attaque 3 :

Nom: Coup Puissant

Puissance: 20 PV

Précision: 33.33% (1 chance sur 3 de toucher)

Attaque 4 :

Nom: Frappe Dévastatrice

Puissance: 30 PV

Précision: 25% (1 chance sur 4 de toucher)

Affichage en Mode Console :

Afficher le nom et les PV restants de chaque entité à chaque tour.

Afficher les options d'attaque disponibles pour le joueur.

Afficher les attaques effectuées par chaque combattant et les dommages infligés.

Gestion des Dommages et de la Santé :

Calculer les dégâts infligés à l'adversaire en fonction de la puissance de l'attaque et de la probabilité de toucher.

Gérer la santé des entités en fonction des dégâts reçus.

Fin du Combat :

Le combat se termine lorsque les PV de l'une des entités (le joueur ou le Sombre Lutin) atteignent zéro.

Afficher un message indiquant le vainqueur du combat.

Interface Utilisateur :

Guider l'utilisateur tout au long du combat avec des messages informatifs et des invitations à choisir une attaque.

Technologies Utilisées :

JavaScript (ES6+)

Node.js pour l'exécution en ligne de commande

Contraintes Techniques :

L'application doit fonctionner entièrement en mode console, sans interface graphique.

Les caractéristiques des attaques et des entités doivent être prédéfinies dans le code source.

Livraisons Attendues :

Code source du jeu correctement commenté et structuré.

Le jeu doit être capable de gérer des erreurs d'entrée utilisateur et de guider le joueur pour qu'il fasse un choix valide.