

Documentation des Frameworks et Bibliothèques Utilisés

Ce document explique l'utilisation des différents frameworks et bibliothèques dans le projet R-Type. Chaque outil est décrit en termes de son rôle, de sa configuration et de la raison de son choix.

1. vcpkg

Rôle :

vcpkg est un gestionnaire de paquets C++ qui facilite l'installation et la gestion des dépendances tierces.

Utilisation :

- vcpkg est utilisé pour gérer les dépendances comme SFML et libconfig.
- Le fichier vcpkg.json dans le projet définit les dépendances requises.
- La commande vcpkg install est utilisée pour installer automatiquement les bibliothèques nécessaires.

Raisons du choix

- Simplifie la gestion des dépendances multi-plateformes.
- Intégration facile avec CMake pour inclure les bibliothèques.

2. SFML (Simple and Fast Multimedia Library)

Rôle :

SFML est une bibliothèque multimédia pour la gestion des graphismes, des entrées utilisateur et des sons.

Utilisation

- Fournit les outils pour créer et afficher des fenêtres.
- Utilisé pour le rendu des éléments graphiques du jeu et la gestion des événements.

Raisons du choix

- **API intuitive et orientée objet** : SFML utilise le paradigme C++, ce qui correspond bien à l'architecture globale du projet.
- **Simplification des tâches courantes** : Les fonctionnalités comme la gestion des fenêtres, des textures, des sons et des événements sont fournies "prêtes à l'emploi", réduisant ainsi le temps de développement.
- **Communauté et documentation** : SFML dispose d'une communauté active et d'une documentation détaillée, ce qui facilite la résolution des problèmes.

Comparatif avec SDL2 et Raylib

Bien que d'autres bibliothèques puissent être envisagées, SFML a été choisie pour ses avantages spécifiques :

Caractéristique	SFML	SDL2	Raylib
Facilité d'apprentissage	Simple pour les débutants	Plus complexe, orienté C	Très simple, orienté jeux
Langage	C++	C	C
Modularité	Orientée objet, claire	Bas niveau, nécessite plus de code	Très haut niveau, limité
Graphismes 2D	Intégrés, performants	Nécessite des extensions	Performants mais limités
Portabilité	Bonne	Excellente	Bonne
Communauté	Active, nombreux exemples	Très active, large support	Croissante

Pourquoi SFML a été choisie :

- Comparée à SDL2, SFML est plus intuitive pour un projet orienté objet, et elle offre des abstractions utiles qui réduisent la complexité du code.
- Par rapport à Raylib, SFML est plus mature et mieux documentée, avec des fonctionnalités plus robustes pour les graphismes 2D avancés.

3. libconfig

Rôle

libconfig est une bibliothèque pour la gestion des fichiers de configuration dans un format clair et lisible.

Utilisation

- Chargement et analyse des fichiers de configuration pour définir des paramètres tels que les options de jeu ou les paramètres réseau.

Raisons du choix

- Format compact et lisible.
- Prise en charge native des types de données courants (entiers, flottants, chaînes, etc.).

4. CMake

Rôle :

CMake est un outil de génération de scripts de construction multiplateformes.

Utilisation

- Gère la configuration et la construction du projet.
- Intègre vcpkg pour inclure automatiquement les dépendances dans le projet.
- Les fichiers CMakeLists.txt contiennent la logique de construction.

Raisons du choix

- Standard dans l'industrie pour les projets C++.
- Compatible avec divers environnements de développement.

5. Doxygen

Rôle :

Doxygen est un outil de génération de documentation à partir de commentaires dans le code source.

Utilisation

- Génère une documentation complète et navigable pour le code.
- Utilise le fichier Doxyfile pour configurer la génération.

Raisons du choix

- Crée une documentation professionnelle à partir du code existant.
- Permet de maintenir une bonne compréhension du projet.