

Projet Prog 2 : Tower Defense

Compte Rendu 3

Raphaël LE BIHAN, Rida LALI

31 mai 2020

1 Présentation du projet

1.1 Présentation générale

L'objectif du projet est de réaliser un jeu de Tower Defense. Le jeu présente une carte 2D où est située une tour appelée tour principale, reliée au bord de l'écran par un chemin. Lors de différentes manches, des ennemis arrivent du bord de l'écran et tentent de prendre d'assaut la tour principale. L'objectif du joueur est de défendre la tour principale de l'assaut ennemi. Pour cela il peut disposer différentes tours de défense sur la carte, qui attaqueront les ennemis à proximité.

1.2 Présentation pour la partie 3

La partie du développement présentée dans ce document concerne l'implémentation d'un système d'enregistrement et d'édition de niveaux. Le joueur a la possibilité d'enregistrer une partie en cours de jeu, pour la reprendre plus tard. Il a aussi la possibilité d'éditer lui-même un fichier au format '.tdp' pour pouvoir jouer cette partie ensuite.

1.3 Format du rendu

Ce document présente le développement général effectué lors de la troisième partie. La syntaxe utilisée dans les fichier '.tdp' et les possibilités d'édition de niveau sont présentées dans le fichier 'GRAMMAIRE.tdp'.

2 Fonctionnalités développées

On développe une nouvelle fonctionnalité permettant d'enregistrer ou de charger une partie de TowerDefProj depuis un fichier.

Un fichier peut soit correspondre à une nouvelle partie, soit à une partie déjà commencée et enregistrée pour être reprise plus tard. Une partie ne pourra être enregistrée qu'entre deux manches (lorsqu'une manche est chargée et est

en train de se dérouler, on ne peut pas enregistrer une partie). On développera aussi dans l'interface graphique un bouton permettant d'enregistrer la partie, un autre pour charger une partie.

Les fichiers correspondants à une partie seront enregistrés au format '.tdp'. Ils seront enregistrés par défaut dans le répertoire '/src/main/resources/saves/'.

3 Informations enregistrées

Les informations suivantes sont enregistrées dans un fichier '.tdp' :

- Endommageables présents sur la carte avec leurs caractéristiques (il n'y aura que des tours) ;
- Etat du gestionnaire de manche : liste des manches suivantes et de leurs caractéristiques (apparition d'ennemis, condition de fin de manche) ;
- Zoom, positionnement de la carte

Le joueur aura la possibilité de :

- Choisir les dimensions de la carte, les tuiles au sol et l'argent possédé par le joueur
- Choisir le type, l'emplacement et les caractéristiques des tours sur la carte
- Choisir le contenu des différentes manches : quels ennemis apparaissent durant la manche (type d'ennemi, emplacement), suivant plusieurs méthodes d'apparition (une seule fois, à intervalle de temps fixe, ou suivant une certaine probabilité), ainsi que la durée minimale avant la fin de la manche.

4 Organisation du code

On développe le code source destiné à l'enregistrement des parties dans le package 'sauvegardes'.

Le code est organisé dans les fichiers suivant, pour le chargement d'une partie :

parser.scala : Analyse lexicale et syntaxique des fichiers .tdp

verif.scala : Vérification de la validité de l'AST

charger.scala : Création des instances pour une partie, affectation des valeurs lues dans l'AST, et chargement des instances créées.

pour la sauvegarde d'une partie :

enregistrer.scala : Ecriture des données de la partie en cours dans un fichier

pour la partie graphique :

boutons_jeu.scala, c_bcharger.scala, c_benregistrer.scala : Ajout des boutons de chargement et sauvegarde à l'interface graphique

file_chooser.scala : Ajout du sélectionneur de fichier

5 Evolution du projet

L'organisation du code pour l'enregistrement et la sauvegarde des fichiers .tdp permet d'étendre la grammaire utilisée actuellement pour ajouter des nouvelles fonctionnalités. On pourra par exemple rendre possible de faire apparaître des ennemis ou des tours avec des effets, ou d'ajouter des conditions de fin de manche plus variées, comme éliminer un ennemi particulier ou éliminer suffisamment d'ennemis.

Une autre évolution possible serait l'implémentation d'un éditeur de niveau, pour permettre au joueur de créer des parties via une interface graphique sans rédiger directement de fichier .tdp.