

Exercice "Text Adventure"

© Protégé par le droit d'auteur

Dylan Guichard

Situation

Un manga très en vogue désire être adapté en jeu vidéo par son comité de production. Afin de valider les idées de gameplay, le ton des dialogues, et pour récapituler rapidement tout ce que l'on pourra faire dans le jeu, le studio mandaté a décidé de réaliser un prototype sous forme de jeu textuel qui se joue dans le terminal.

Dans le scénario du manga, le lycée A.U. vise à former les meilleurs héros des pays où il se trouve. Son nom vient de la prononciation des lettres "AU" en anglais qui est semblable à la prononciation de "héros" en japonais, qui se dit "eiyuu". Toute référence à un autre lycée n'est que fortuite...

L'action du jeu se passe donc dans ce lycée. Après avoir rempli un questionnaire au début du jeu pour créer le personnage principal, le joueur peut se déplacer librement dans différents endroits du lycée, chacune proposant des actions que l'on peut réaliser.

Objectifs

Les fondations du jeu sont posées, mais il reste encore beaucoup à faire : les lieux à créer pour y accéder et pour y réaliser des actions, et surtout la salle d'entraînement où l'on peut se combattre de façon assez rudimentaire.

Cette fois-ci, il faudra autant lire des variables que les manipuler, selon les actions réalisées par l'utilisateur. Il n'y a que deux choses que l'utilisateur peut entrer : le nom d'un autre **lieu** vers lequel se déplacer, ou le nom d'une **action**.

Bien sûr, au-delà d'avoir un code fonctionnel, l'objectif est de réussir à s'amuser un peu. Pour cela, n'hésitez pas à rajouter votre touche d'originalité à travers les dialogues, les actions réalisables, voire en rajoutant des lieux ou des fonctionnalités au jeu. Si une idée supplémentaire que vous tenez absolument à réaliser vous semble trop difficile, vous pouvez bien sûr me demander conseil.

🔍 Gestion des objets

Le jeu commence avec la présence d'un smartphone comme **objet clé** (qui ne sert à rien de spécial) puis avec un **inventaire** vide.

Les objets clés n'ayant toujours qu'un seul exemplaire, on utilise une liste pour les stocker.

L'inventaire permet cependant de garder plusieurs quantités d'un même objet, on utilise alors un dictionnaire pour stocker la quantité au nom d'un objet.

Les deux types d'objets pourront servir tout au long du jeu. On peut récupérer des objets selon les endroits visités ou les actions réalisées.

🔍 Les lieux

Pour être complet, le jeu doit proposer au minimum les lieux suivants :

- * Hall d'entrée (RDC)
- * Couloir du RDC
- * Classe 1-A (RDC)
- * Couloir du 1er étage
- * Cafétéria (1er étage)
- * Salle d'entraînement (1er étage)

On ne peut aller d'un lieu à un autre que si l'on y est proche (du couloir du 1er étage à la classe 1-A par exemple). On ne peut pas rentrer dans la salle d'entraînement sans avoir un passe.

🔍 Les actions

Une action est un verbe à l'infinitif. Qu'importe l'endroit, il faut prévoir l'action `observer` qui sert juste à afficher dans le terminal une description de l'endroit. Par exemple, dans le hall, cela affichera "Vous êtes dans un hall au Vu que le jeu est textuel, il faudra être très descriptif en échange : une action commune à tous les lieux est donc `observer` .

Le hall :

- Action `quitter` => arrête simplement le script.

La cafétéria :

- Action `manger` => Après avoir demandé le plat voulu, le joueur gagne 10 PV (points de vie). Il n'y a que 2 exemplaires de chaque plat en stock, il faudra donc **soustraire** un plat en stock à chaque utilisation, et **annuler** l'action quand un plat demandé n'est plus en stock.

La classe 1-A :

- Action `demander` => Sert à recevoir le passe (un objet clé) de la part du professeur, et qui permettra d'entrer dans la salle d'entraînement du 1er étage.

- Action `montrer` => Sert à montrer au professeur le nombre de badges que l'on a sur soi : si on possède trois badges gagnés à la suite de trois combats contre le mannequin d'entraînement, le jeu est réussi et le script s'arrête.

La salle d'entraînement :

- Action `combattre` => Lance une **boucle** qui continuera tant que le mannequin d'entraînement n'aura pas ses PV (points de vie) à zéro. Au début d'un combat, le

mannequin commencera toujours à 50 PV, qu'importe ce qu'il s'est passé auparavant. Dans cette boucle, il faudra proposer à l'utilisateur soit l'action `fuir` qui **annule** la boucle, soit l'action `attaquer`. L'attaque retire 15 PV au mannequin, et ce dernier répliquera en retirant 10 PV au joueur. Si le joueur perd tous ses PV, le script s'arrête. Au contraire, si le mannequin est combattu avec succès, on gagne un **badge** dans notre inventaire qui atteste de notre réussite.