Groupe BERA DESCAMPS
DUMAS ANDRES

Cahier de recette Buckshot

SOMMAIRE

(Les liens du sommaire sont cliquables en faisant un CTRL + Clic droit)

1) Introduction

Dans ce document, nous allons détailler les tests que nous avons effectués pour vérifier le bon fonctionnement de notre jeu que nous avons développé en C avec l'interface graphique SDL2. Notre jeu implémente des fonctionnalités tel que la distribution aléatoire des cartouches (chargés et à blanc), la gestion des tours du joueur et de l'ordinateur, ainsi que l'affichage et l'interface graphique.

2) Exigences fonctionnelles

2.1) Affichage de la grille principale et des sous-grille

Notre grille principale comporte 2 lignes et 5 colonnes. Nous avons défini notre grille par des positions en pixels, afin de définir nos éléments cliquables. Nous avons également des sous-grilles représentant l'inventaire. Nous affichons notre grille et nos sous-grilles avec notre fonction ''drawGrid''.

2.2) Chargement et affichage des textures

Les textures des objets (cigarette, bière, loupe, lunettes) et du fusil à pompe sont chargées et affichées grâce à notre fonction ''loadTexture''.

2.3) Gestion des clics de la souris

Les clics sur notre interface sont détectés et traités par notre fonction ''handleMouseClick''. Les clics déclencheront des actions en fonction de l'état du jeu.

2.4) Distribution aleatoire des cartouches

Le nombre de balles est tiré aléatoirement entre 2 et 8. Les balles sont de couleur rouge pour les vraies munitions, et les noirs représentent les munitions à blanc.

2.5) Tour de jeu

Les tours de jeu entre le joueur et l'ordinateur sont gérés par les fonctions ''joueurTour'' et ''ordinateurTour''. Des actions seront prises en conséquence par l'ordinateur en fonction des objets qu'il tire, et des munitions dans le fusil.

2.6) Utilisation des objets

Les objets dans l'inventaire du joueur sont utilisables en cliquant dans la bonne zone définie sur notre affichage (le sprite de l'objet en lui-même). Les effets des objets sont alors appliqués par ''giveObject''

3) Plan des tests

Test 1 : Affichage de la fenêtre

Objectif : Vérifier que la fenêtre d'affichage s'affiche correctement, ainsi que l'inventaire du joueur.

Condition(s) : L'application doit être lancé

Procédure : Lancer le programme -> Observer que l'affichage apparait bien

Critère de succès : La fenêtre doit s'afficher correctement et en entier

Test 2 : Chargement des textures

Objectif : Vérifier que tous nos sprites sont chargés et affichées correctement

Condition(s) : Les sprites doivent être présent dans l'arborescence du projet

Procédure : Lancer le programme -> Vérifier que tous les sprites ont bien été chargés et affichés

Critère de succès : Toutes les textures sont affichées correctement et ne sont pas déformées

Test 3 : Gestion des clics de la souris

Objectif : Vérifier que les clics de la souris déclenchent bien des actions

Condition(s) : L'application doit être lancé et la fenêtre affichée

Procédure : Cliquer sur les différentes zones de la fenêtre -> Observer si on obtient des réactions

Critère de succès : Les clics de la souris sont détectés et traités correctement

Test 4 : Distribution aléatoire des munitions

Objectif : Vérifier que la distribution aléatoire de munitions fonctionne

Condition(s) : Le programme doit être lancé et on doit être lors du tour du joueur ou de l'ordinateur

Procédure : Commencer une nouvelle partie -> Observer le nombre et la couleur des munitions générées

Critère de succès : Le nombre de balles est entre 2 et 8, avec au moins une cartouche rouge et une cartouche noire

Test 5 : Tour du joueur et de l'ordinateur

Objectif : Vérifier le bon déroulement des tours

Condition(s) : Le programme doit être lancé et on doit être lors du tour du joueur ou de l'ordinateur

Procédure : Jouer plusieurs tours à la suite (en alternance avec le joueur et l'ordinateur) -> Observer les actions effectuées à chaque tour

Critère de succès : Le tour entre le joueur et l'ordinateur alternent correctement et les actions sont exécutés en fonction des règles

Test 6 : Utilisation des objets

Objectif : Vérifier que les objets soient reconnus, et que leur utilisation est bien celle que l'on attend

Condition(s) : Le programme doit être lancé et des objets doivent être présent dans l'inventaire du joueur

Procédure : Trouver et utiliser des objets dans l'affichage -> Observer les effets des objets

Critère de succès : Les objets sont cliquables et possèdent leurs effets

4) RESULTAT DES TESTS

La plupart des tests ont été passés avec succès dans la version finale de notre projet, mais voici quelques erreurs que nous avons rencontrés lors de l'implémentation des fonctions reliés aux tests.

Test 2 : Rotation du fusil de chasse -> Nous avons eu des problèmes pour faire tourner le sprite du fusil de chasse à chaque tour, nous n'avons pas réussi à le réaliser. Nous avons alors essayé de faire apparaitre une image du fusil vers le croupier pendant le tour du joueur, et vers le joueur pendant le tour du croupier. Le problème est que quand on charge la fenêtre graphique, tous les éléments sont directement mis dans une boucle, et cela est problématique pour rajouter / enlever des éléments

Test 5 : Cas de la balle à blanc (blank) -> Nous avons eu du mal à mettre en place le fait que si une balle à blanc est tirée sur un joueur, alors il rejoue, car la boucle du jeu est de base prévue pour alterner les tours entre le joueur et l'ordinateur.

Test 6 : Menottes -> Des objets ont été difficiles à mettre en place comme les menottes, qui comme le problème précédent du test 5, modifie la boucle de jeu.

5) Conclusion

Nous avons réussi à remplir la majorité de notre cahier des charges, et passés la grande majorité de nos tests (bien que le test 1 reste problématique). Les tests que nous nous sommes imposés nous ont néanmoins permis d'apporter des améliorations à notre code en identifiant des problèmes clés, tout en prenant en compte que ces problèmes étaient détectés un seul à la fois, facilitant leur résolution.