

Dumas Antoine
p1802788
ESCRIVA William
p1702268
MESSELMANI Ali
p1811574

Cahier des Charges

Project Orcus

1-Présentation des acteurs

L'université Claude Bernard à Lyon 1, est une université spécialisée dans le domaine des Sciences.

En 2ème année de licence informatique, dans le cadre de la matière LIFAP4 nous devons réaliser un projet par groupe de 3. Notre groupe étant composé de Dumas Antoine, ESCRIVA William, MESSELMANI Ali.

2-Description du produit

Nous avons décidé de créer un jeu qui a pour nom de code « *Projet Orcus* ».

Notre jeu est un *shoot'em up* qui traduit de l'anglais nous donne « tirer leur dessus » de la violence sera donc présente, le jeu est à destination d'un public averti. la caméra du joueur sera située sur le dessus.

Règles du jeu

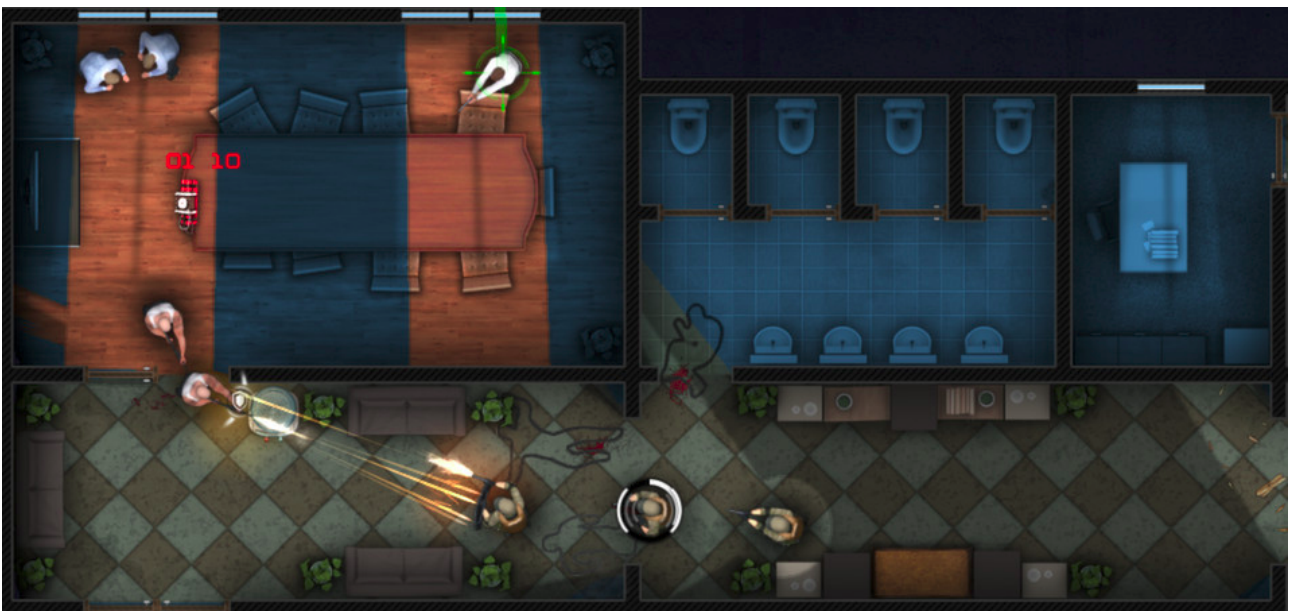
Un ou plusieurs joueurs évolueront dans différents niveaux où ils devront survivre à de multiples vagues ennemies à l'aide d'un arsenal d'armes variées qu'ils débloquent au fur et à mesure de leur progression dans les niveaux.

Le joueur devra traverser une carte d'un point A à un point B sans se faire tuer par les vagues ennemies. S'il y parvient, il pourra accéder au niveau suivant et ainsi de suite.

En multijoueur, un joueur pourra se faire réanimer par un autre si il se fait tuer par les ennemies.

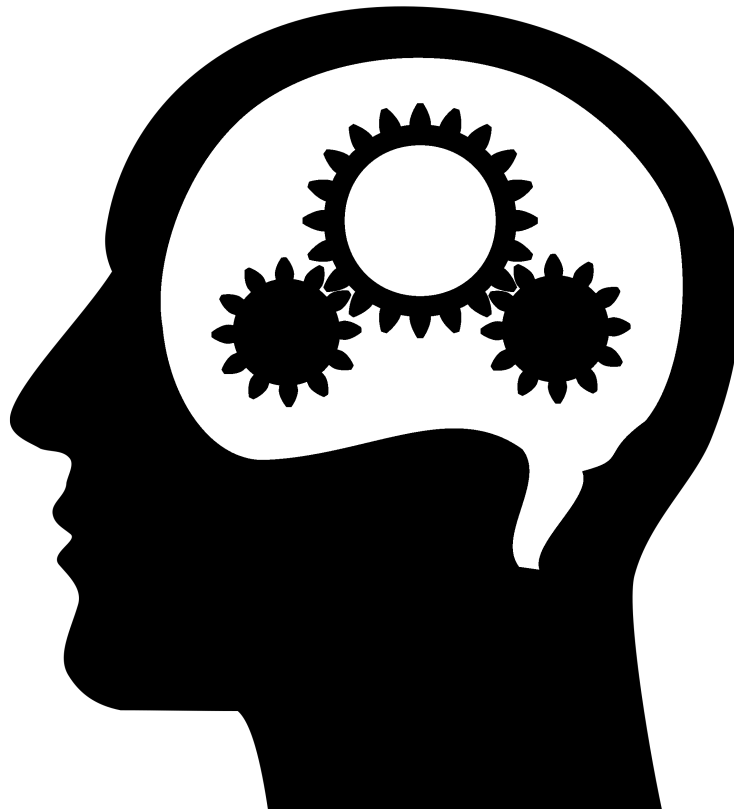
Au début de chaque niveau le joueur disposera d'une arme de base, il pourra en utiliser d'autre car dans chaque carte se cache différents types d'armes. Ces armes cachées sont utilisées sous forme de bonus c'est-à-dire qu'elles seront utilisables pendant un temps donné, après quoi le joueur retrouvera son arme de base.

Illustration



3-Contraintes

- Le jeu sera développé en C/C++ sous Linux
 - Les librairies utilisées seront SDL2
 - Le code respectera le standard suivant : code indenté, variable ayant du sens,...
 - Le code sera géré et archivé surGitLab
 - La documentation du code sera produite par Doxygen
 - Un diagramme des classes permettra d'avoir une vision de haut niveau de l'implantation
 - L'équipe de développement utilisera les outils de debug et de profiling : gdb et valgrind
-



4-Déroulement du projet

Tâche 0 - Rédiger le cahier des charges

Membres impliqués : tous

Durée : 2 semaines

Tâche 1 - Définir le diagramme des classes

Membres impliqués : tous

Durée : 1 semaine

Tâche 2 - Développement premier prototype (SDL2)

Durée : 2 semaines

Tâche 2.1 - Affichage d'une fenêtre SDL2

Membre(s) impliqué(s) : Antoine DUMAS

Une fenêtre est définie par une position et des dimensions.

Définition du module *game* qui nous permet de gérer les événements du jeu.

A ce stade la on doit pouvoir afficher une fenêtre blanche.

Tâche 2.2 - Création de la boucle de jeu

Membre(s) impliqué(s) : Antoine DUMAS

La boucle de jeu se situe dans le fichier *main.cpp*.

C'est une boucle *while* dans laquelle on va gérer, dans un premier temps, les événements claviers, souris et les entrées utilisateur. Ensuite on met à jour le jeu, par exemple on modifie la position du joueur si elle a été changée par l'utilisateur.

Et pour finir, on redessine la fenêtre avec les modifications.

Tâche 2.3 - Déplacement du personnage

Membre(s) impliqué(s): William ESCRIVA

Définition du module *complexe* qui permet de gérer les déplacements.

Définition du module *personnage*.

Gérer les événements clavier pour les déplacements.

Le personnage jouable est définie par une position, une vitesse, point de vie, ...

Tâche 2.4 - Affichage de texture

Membre(s) impliqué(s): Antoine Dumas

Définition du module *map*.

Une carte est définie par un tableau 2D et des textures pour chaque cases. Ce module sert à afficher les niveaux du jeu.

Tâche 3 – Développement du deuxième prototype (SLD2)

Durée : 2 semaines

Tâche 3.1 - Animation des personnages (Sprite)

Membre(s) impliqué(s): Ali MESSELMANI

Modification du module *personnage*.

Ce module anime les personnages du jeu grâce au feuilles de sprite.

Avec un boucle *while* on affiche successivement les sous-images de la feuille de sprite, ce qui nous donne l'animation.

Tâche 3.2 - Création d'une carte

Membre(s) impliqué(s): William ESCRIVA

Création de la carte avec Tiled qui est un éditeur de niveau en 2D.

Importation de la carte dans le programme (TMX map loader).

Tâche 3.3 - Implémentations des classes objets (armes, trousse de soins, ...)

Membre(s) impliqué(s): Antoine DUMAS

Définition du module arme.

Une arme est défini par une texture, les dégâts ainsi qu'un emplacement.

Définition d'un module pour chaque consommable du jeu.

Tâche 4 - Développement du troisième prototype avec les musiques et bruitages (SLD2)

Durée : 2 semaines

Tâche 4.1 – Implémentation du son

Membre(s) impliqué(s): Ali MESSELMANI

Définition du module audio.

Avec le classe audio on peut lire des fichier audio dans le programme.

Tâche 4.2 – Ajout de cartes ennemis

Membre(s) impliqué(s): Antoine DUMAS & William ESCRIVA

Ajout de plusieurs cartes et de plusieurs sprites d'ennemis.

Tâche 5 - Développement du quatrième prototype avec ajout d'un mode multijoueur local (SLD2)

Durée : 2 semaines

Tâche 5.1 – Implémentation d'un menu.

Membre(s) impliqué(s): tous

Ajout d'un menu permettant de choisir son héros, voir les scores.

Tâche 5.2 – Implémentation de SDL_net

Membre(s) impliqué(s): tous

SDL_net permettra de gérer le mode en ligne.

5-Annexe

Diagramme des classes

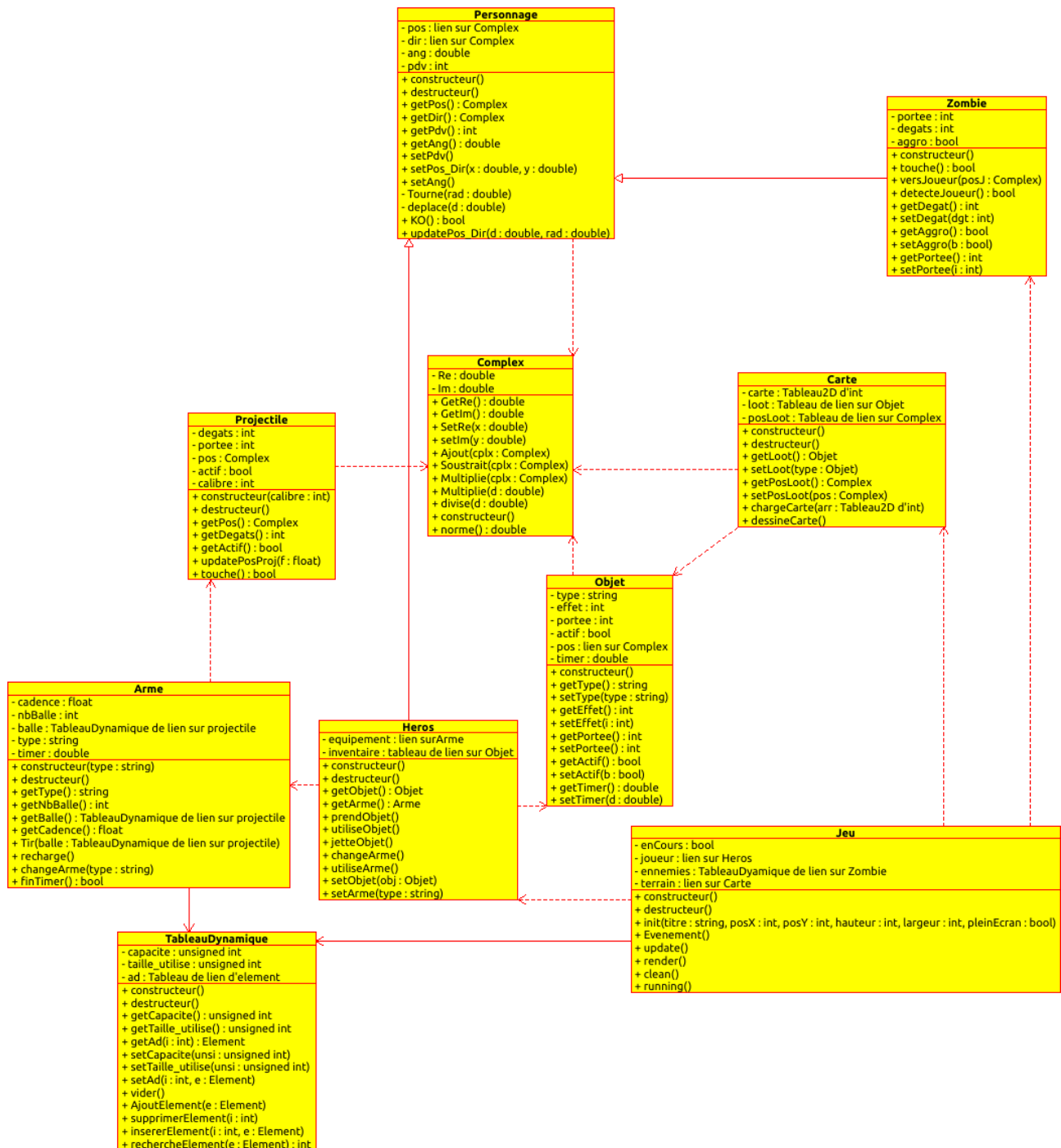


Diagramme de GANTT

[illegible]