**1. Présentation des acteurs**

La société LEB22 est une start-up a été fondé dans le cadre du projet de lifap4 pour répondre

À la demande de l'université Claude Bernard Lyon 1. Constitué de 3 magnifique développeurs

talentueux spécialisé dans la programmation en C/C++ pour diverses applications.

Dans ce contexte, la société Université Claude Bernard commande le produit « Discipuli » à la société

LEB22. Le présent cahier des charges est composé :

- d’une description détaillée du produit

- d’une prévision détaillée du déroulement du développement

- d’un diagramme de Gantt

- d’un diagramme des classes

L'ensemble de ces documents sera maintenu pendant toute la durée de développement du produit.

**2. Description du produit**

**Principe et règles du jeu**

Le jeu se lance sur un menu avec comme option "Commencer" qui lance le jeu, une option choisissant son terrain, d'une option aide et des 3 meilleurs scores réalisés (pouvant être remis à 0).

Lors du début de la partie un plateau généré aléatoirement avec un chemin apparaît.

Des ennemis apparaissent sur le chemin et des tours peuvent être placés sur les autres cases.

Les tours attaquent les ennemies et les ennemies doivent avancer jusqu'à la fin du chemin.

Après un certain nombre d'ennemis finissant le chemin le joueur perd, son but est d'aller le plus loin possible.

Le joueur possède de l'argent capable d'ajouter des tours sur le terrain ou d'améliorer celles déjà présentes.

L'argent est gagné après chaque ennemi vaincu.

Les ennemis apparaissent sur le chemin en vague qui sont chacune plus forte que les autres.

Des bonus peuvent être lâchés par les ennemis:

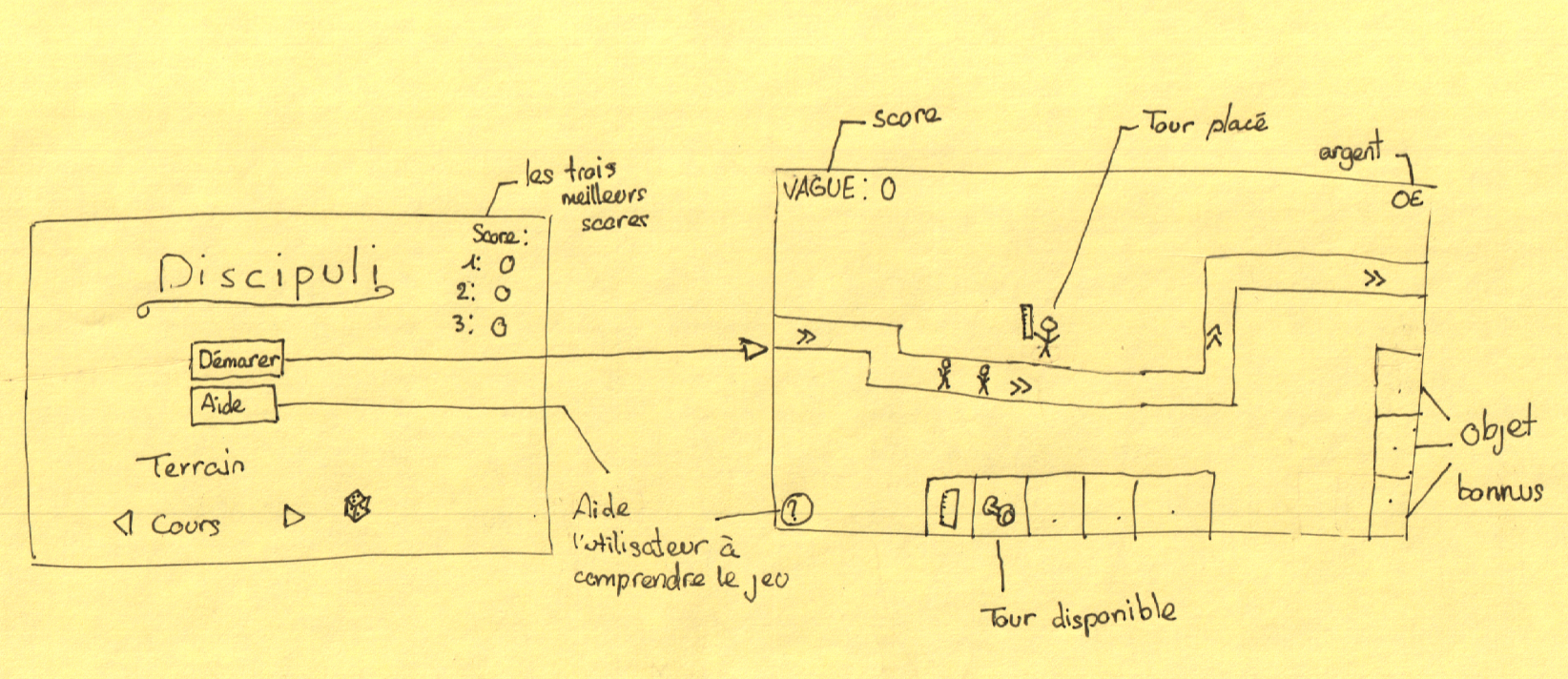
1 augmente la vitesse d'attaque des tours

2 baisse la vitesse des ennemis

3 Rajoute de l'argent

Chaque vague a une chance de devenir plus dur que le normal ( dans le contexte du jeu appelé "période d'examens").

**Détail de l’interface et des niveaux du jeu**



**3. Contraintes**

- Le jeu sera développé en C/C++ sous Linux

- Les librairies utilisées seront SDL2 et une librairie texte pour une version alpha

- Le code respectera le standard suivant : code indenté, variable ayant du sens, ...

- Le code sera géré et archivé sur GitLab

- 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

- Le code sera fourni à la société InfogamesPro

**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 1er prototype mode texte (1 type d'ennemi, 1 type de tour)**

Durée : 3 semaines

Tâche 2.1 : écriture et test module Vecteur (Vecteur.h /.cpp)

Membres impliqués : Ludovic

Tâche 2.2 : écriture et test module TableauDynamique (TableauDynamique.h /.cpp)

Membres impliqués : Evan

Tâche 2.3 : écriture et test module Ennemi (Ennemi.h /.cpp)

Membres impliqués : Ludovic

Tâche 2.4 : écriture et test module Tour (Tour.h /.cpp)

Membres impliqués : Deux

Tâche 2.5 : écriture et test module Terrain (Terrain.h/ .cpp)

Membres impliqués : Evan

Tâche 2.6 écriture et test module jeuModeTexte (jeuModeTexte.h /.cpp)

Membres impliqués : tous

**Tâche 3 : développement 2eme prototype mode graphique (implémentation de SDL2)**

Membres impliqués : tous

Durée : 2 semaines

**Tâche 4 : développement 3eme prototype mode graphique**

Durée : 2 semaines

Tâche 4.1 : écriture et test sytème d'argent (implémenté dans module Terrain)

Membres impliqués : Evan

Tâche 4.2 : écriture et test sytème d'upgrade de tour (implémenté dans module Tour)

Membres impliqués : Deux

Tâche 4.3 : écriture et test module Objet (Objet.h / .cpp)

Membres impliqués : Ludovic

**Tâche 5 : développement 4eme prototype mode graphique**

Durée : 2 semaines

Tâche 5.1 : écriture et test nouveaux types d'ennemis (implémenté dans module Ennemi)

Membres impliqués : Ludovic

Tâche 5.2 : écriture et test nouveaux types de tours (implémenté dans module Tour)

Membres impliqués : Deux

Tâche 5.3 : écriture et test sytème de sprites

Membres impliqués : Evan

**Tâche 6 : développement 6eme prototype mode graphique**

Durée : 2 semaines

Tâche 6.1 : écriture et test système de trajet aléatoire (implémenté dans module Terrain)

Membres impliqués : Ludovic

Tâche 6.2 : écriture et test module Menu (Menu.h /.cpp)

Membres impliqués : Deux

Tâche 6.3 : écriture et test sytème de choix de map (implémenté dans module Menu)

Membres impliqués : Deux

Tâche 6.4 : écriture et test évenements aléatoires (implémenté dans module Terrain)

Membres impliqués : Evan

**Tâche 7 : développement 7eme prototype mode graphique (implémentation de musique ?)**

Membres impliqués : Deux

Durée : 2 semaines

**5. Diagramme de Gantt**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Semaine 1 | Semaine 2 | Semaine 3 | Semaine 4 | Semaine 5 | Semaine 6 | Semaine 7 | Semaine 8 |
| Tâche 0 |  |  |  |  |  |  |  |  |
| Tâche 1 |  |  |  |  |  |  |  |  |
| Tâche 2.1 |  |  |  |  |  |  |  |  |
| Tâche 2.2 |  |  |  |  |  |  |  |  |
| Tâche 2.3 |  |  |  |  |  |  |  |  |
| Tâche 2.4 |  |  |  |  |  |  |  |  |
| Tâche 2.5 |  |  |  |  |  |  |  |  |
| Tâche 2.6 |  |  |  |  |  |  |  |  |
| Tâche 3 |  |  |  |  |  |  |  |  |
| Tâche 4.1 |  |  |  |  |  |  |  |  |
| Tâche 4.2 |  |  |  |  |  |  |  |  |
| Tâche 4.3 |  |  |  |  |  |  |  |  |
| Tâche 5.1 |  |  |  |  |  |  |  |  |
| Tâche 5.2 |  |  |  |  |  |  |  |  |
| Tâche 5.3 |  |  |  |  |  |  |  |  |
| Tâche 6.1 |  |  |  |  |  |  |  |  |
| Tâche 6.2 |  |  |  |  |  |  |  |  |
| Tâche 6.3 |  |  |  |  |  |  |  |  |
| Tâche 6.4 |  |  |  |  |  |  |  |  |
| Tâche 7 |  |  |  |  |  |  |  |  |

Ludovic, Evan , Deux

**6. Diagramme des classes (version simplifée)**

