

GAUTROT Oriane

NINO-LAGADEC Timothée

DUPIED Benjamin

SALQUEBRE Valentin

TELECOM NANCY 1A FISE

Mai 2024

Projet Pluridisciplinaire d'Informatique Intégrative

Création d'un jeu vidéo en C avec SDL2

« TN Escape »

Genèse du projet :

L'aventure commence début mars 2024, après l'annonce du sujet de ce projet de deuxième semestre de première année étudiante à Telecom Nancy.

Le groupe se forme rapidement, et nous tombons tous rapidement d'accord sur la nature du jeu que nous souhaitons développer : un labyrinthe, dans un style retro, enrichi avec un système de combat contre des monstres.

Développement du jeu et méthodologie de travail :

Dès nos premières réunions début mars, nous nous mettons facilement d'accord sur la direction que prendra le jeu, mais nous sommes également conscient des difficultés que nous allons rencontrer :

- Nous avons une faible visibilité sur la décomposition en tâches nécessaire à la réalisation du projet.
- La répartition de la charge de travail dans le temps va être compliquée à arbitrer avec nos différentes contraintes.

Bien que nous soyons en accord sur l'aspect technique du projet, des différences de point de vue émergent rapidement sur la méthodologie à adopter, et sur le rythme de travail. Certains membres du groupe estiment qu'il faut directement commencer à coder chacun de son côté, alors que d'autres préféreraient une granularité plus fine dans la planification des objectifs (ce qui implique dans un premier temps de discuter davantage et de coder moins).

De premières tensions apparaissent. Tous les membres du groupe ne s'impliquent pas de la même manière, certains se mettent en route plus vite que d'autres. Tout cela a lieu durant une période de partiels, ce qui n'aide pas à apaiser la situation.

Malgré ces difficultés, nous nous retrouvons dans la volonté de rendre un beau projet fonctionnel. Nous sommes en conversation continue sur une application de messagerie, et nous efforçons de concilier nos efforts pour avancer au mieux. C'est ainsi durant le mois de mai que se concentrera la majorité du travail. Nous adoptons une méthode Agile : chacun essaie d'implémenter une ou plusieurs fonctionnalités de son côté, et des bilans sont faits très régulièrement, afin de pouvoir consulter les autres membres sur les obstacles rencontrés et la démarche à adopter.

Une autre difficulté majeure s'est imposée à nous : la complexité du travail à 4 personnes sur un même code. Plus le nombre de lignes de codes et de fonctions augmente, plus il devient laborieux d'ajouter une fonctionnalité sans interférer avec ce qui a été mis en place par un autre membre du groupe. Des points de vue divergents au sujet de l'utilisation des outils de GitLab ont généré d'autres tensions relationnelles. Sur la phase de finalisation du projet, énormément de temps et d'énergie ont été perdus à résoudre des problèmes de fusion de branche, ce qui a nuit grandement à la productivité du groupe ainsi qu'au climat de travail.

Ce travail de groupe aura globalement été particulièrement challengeant au niveau de la gestion de projet et de la cohésion d'équipe. Toutes les fonctionnalités que nous avons envisagées ne pourront pas être concrétisées. Pour autant, la visée pédagogique de ce projet étudiant a pris tout son sens : nous avons pu identifier des erreurs à ne pas reproduire, des comportements à éviter, et des stratégies efficaces pour surmonter les défis futurs. Nous en sortons grandis, avons gagné en compétences relationnelles et organisationnelles, et sommes confiants pour les projets futurs.

Reunion PPII 19 MARS

Membres présents : Oriane, Benjamin, Timothée, Valentin

Problématique :

- Répartition de la charge de travail - partiels
- Faible visibilité sur les tâches

idées : perso labyrinthe etc (cf mail)

Features possibles :

Vision autour du perso uniquement

Lanterne allumée si combat gagné ?

Idée : combat :

→ Interface à la Pokemon ?

→ Interface combat type l'ennemi nous tire dessus, et soit on doit aller le frapper au CAC, soit on lui tire dessus également en évitant ses projectiles

Idées :

UI barre de PV, gain de stats : nombre de cases de vision autour du personnage ?

Consommables ?

Avoir un temps limité pour sortir ? Développer le lore ?

Prochaine réunion : vendredi 22 mars à midi

REU vendredi 22 mars :

Membres présents : Oriane, Benjamin, Timothée, Valentin

Réflexion sur le thème graphique : caverne/médiéval

Types d'armes possibles : arc, boomerang ...

Questions à poser au enseignants : Quel logiciel pour les sprites personnage ? MAP ? Droit de récupérer des assets ?

Essayer d'avoir les images au plus vite pour pouvoir commencer à coder

Réflexions sur la stratégie pour avoir le cache de vision. MAP fixe ou fenêtre glissante ?
→ On part sur map fixe

Pour la prochaine fois :

- Créer la map : Oriane + Valentin
- Créer les sprites persos : Benjamin + Timothée

PPII 1AS2

Réunion Discord du jeudi 16 mai (17h-18h)

Membres présents : Oriane, Benjamin, Valentin

Ordre du jour :

- Point sur l'avancement du projet (Passage en revue de certaines fonctions dans le code)
- Concertation et hiérarchisation des objectifs de la prochaine phase
- Discussion des problèmes passés et à venir
- Affectation des tâches

Idées Validées Importantes/Urgentes : (avec répartition des tâches)

- Correction obstacles : [Benjamin]
- Pouvoir cliquer sur les items du menu, ou alors afficher les raccourcis à l'écran [Valentin]
- Page commandes/règles [Valentin]
- Corriger sprites pour que le perso bouge [Benjamin]
- Position des monstres : déterministe : Coder le nécessaire pour placer les monstres sur la map [Oriane]
- Comportement monstres, projectiles [Valentin]
- Sprites boules de feu [Benjamin]
- Boules de feu des monstres : on la voit par-dessus le noir [Valentin]
- Torches : Après avoir vaincu un monstre, elle s'allume et le reste [Oriane]
- Mécanique combat : si on met un coup d'épée on tue le monstre au cac, sinon il nous tue. Le monstre nous tue également avec projectiles [Timothée]
- Implémenter un chrono ? Possibilités débloquées : enregistrer les meilleurs temps (petite bdd, avec choix de nom de pseudo ou de partie) – avoir un boost de visibilité pendant un certain temps ou un certain nombre de cases [Le premier à en avoir besoin s'en occupe]
- Ajouter des couleurs pour les sprites perso (le perso devient rouge quand il se fait toucher puis meurt -> Game over. Idem pour le blob. Durée du rouge : 1 seconde (à voir pour le timer)) [Timothée]

Idées Validées Non Importantes/Urgentes :

- Cases périphériques ou autre style de cases pour que le perso ne puisse pas sortir de l'écran, il arrive genre par une porte, et sort par une autre porte. [Valentin]
- Ajouter des maps, chacune avec une nouvelle feature ? [Valentin]
- Musiques/sons [Benjamin]
- Interface/UI
- Consommables, bonus sur la map ? (invulnérabilité, gain de vision)

Idées Rejetées :

- Unifier le déplacement par cases ? → Non, on laisse comme ça
- Capacité combat du perso (cac ? PDV ? dégâts ?) → On part sur une mécanique de OS donc pas besoin
- revoir la résolution pour être en plein écran ?

Problématiques :

- Manque d'Algorithmes poussés, idem structures de données (idées : des monstres qui nous courent après / stocker l'historique des scores des parties dans une BDD)
- Tests unitaires [Timothée]
- Documentation à faire [Valentin]

Stand-by :

- Proposer de passer la distance/vitesse de déplacement à 1 ?

Rédigé par : Valentin

PPII 1AS2

Réunion Discord du vendredi 24 mai (14h30-15h)

Membres présents : Oriane, Benjamin, Valentin, Timothée

Ordre du jour :

- Point sur l'avancement du projet
- Concertation et hiérarchisation des objectifs de la phase finale
- Affectation des tâches
- Mise en place d'une trame claire jusqu'à la fin du projet

Idées Validées Importantes/Urgentes :

- Comportement monstres [Benjamin]
- Projectiles : Boules de feu des monstres : on la voit par-dessus le noir [Valentin]
- Mécanique combat : si on met un coup d'épée on tue le monstre au cac, sinon il nous tue. Le monstre nous tue également avec projectiles [Timothée]
- Ajouter des couleurs pour les sprites perso (le perso devient rouge quand il se fait toucher puis meurt -> Game over. Idem pour le blob. Durée du rouge : 1 seconde [Timothée])
- Torches : Après avoir vaincu un monstre, elle s'allume et le reste : Idée pour les torches : une fois tué, une torche apparaît à la place du blob [Oriane]
- Transition portes : apparition sur la 1ère, et quand on passe la dernière on a gagné. En profiter pour afficher un truc à l'écran en cas de victoire ou défaite [Valentin]
- Interactions retour menu depuis le jeu ? Pause du jeu ou arrière plan ? Ne pas confondre avec nouvelle partie après mort [Valentin]
- Tests unitaires [Oriane]
- Documentation technique du projet [Benjamin]
- Documentation Gestion de projet [Valentin]

Améliorations possibles :

- Son sur les événements
- Ajouter des maps, chacune avec une nouvelle feature ?
- Consommables, bonus sur la map ? (invulnérabilité, gain de vision)
- Mettre un peu de rng dans le choix des illustrations des tuiles avec des ptites variantes
- Avoir un mob qui nous court après ou un truc du genre pour algorithme de plus court chemin ? Pour la dernière map par exemple ?

A faire à la fin :

- Refaire les pages règles et commandes selon l'état du jeu. Améliorations graphiques possibles.
- S'assurer que la mémoire est libérée quand nécessaire
- Nettoyer le code et les fichiers du git

Rédigé par : Valentin

À ces réunions s'ajoutent de nombreuses discussions et échanges en flux tendu, amenant à des petites décisions incrémentales successives.

Documentation

Description du jeu :

Le joueur doit traverser 3 labyrinthes en moins de 10 minutes pour gagner la partie. Seule une petite partie de la carte est visible autour du personnage, le reste est en noir. Les labyrinthes possèdent chacun une porte d'entrée et une porte de sortie. Lorsque le joueur atteint la porte de sortie, il est téléporté automatiquement vers la carte suivante.

Sur les 2^e et 3^e cartes, il y a des monstres (Blobs) à éviter et/ou tuer. Ils tirent des boules de feu en direction du personnage. Elles sont visibles dans le noir mais elles ne traversent pas les murs. Le joueur peut se défendre en tirant des balles avec son pistolet. Les balles tuent les Blobs en un seul coup. Le joueur meurt s'il est touché par un Blob ou une boule de feu.

Le jeu s'arrête si le personnage meurt, s'il atteint la dernière porte ou si le chronomètre est à 0.

Structure du jeu :

- main.c : Fichier principal contenant la boucle de jeu et la gestion des événements.
- font.c : Fichier contenant les fonctions liées à la gestion des labyrinthes.
 - Génération et chargement des 3 labyrinthes.
 - Affichage de la carte avec une visibilité limitée autour du personnage.
- move.c : Fichier contenant les fonctions liées au personnage.
 - Gestion des déplacements du personnage.
 - Gestion des animations du personnage.
 - Gestion de la vie et de la mort du personnage.
- monster.c : Fichier contenant les fonctions liées aux monstres (Blobs).
 - Gestion des déplacements et des animations des Blobs.
 - Gestion des collisions entre les Blobs et le personnage.
 - Gestion de la vie et de la mort des Blobs.
- projectile.c : Fichier contenant les fonctions liées aux projectiles (balles et boules de feu).
 - Gestion du tir des balles par le personnage.
 - Gestion du tir des boules de feu par les Blobs.
 - Gestion des déplacements et des collisions des projectiles.
 - Gestion de l'affichage des projectiles.
 - Une structure de liste chaînée a été choisie pour les projectiles. Cela permet de les parcourir facilement et efficacement, tout en ayant une facilité d'insertion et de suppression.

Autres éléments :

- Timer : Gestion du chronomètre de 10 minutes.
- Musique de fond et différents bruits quand le joueur tire, meurt ou gagne.