

Rapport

Cy-Fighter



Démarrage et mise en place

Nous sommes un groupe de 3 dont les membres sont Dhahak Yassine, Charbel Elias et Joseph Kouto-messan. Le choix du sujet s'est fait rapidement et nous nous sommes orientés vers le Cy-fighter, influencés par nos expériences communes sur plusieurs jeux de combat tels que Street Fighter ou encore Brawlhalla, dont on s'est inspiré dans certains aspects comme l'ajout d'armes. Nous avons ensuite créé nos comptes GitHub en même temps afin d'obtenir par la suite l'URL de dépôt de notre projet. En revanche, après

répartition des tâches, nous codions chacun de notre côté sur des supports différents (VS Code, compilateur gdb, etc.).

Avancement et organisation

Avant même de commencer à coder, nous avons d'abord parlé à l'oral des quelques idées que chacun avait, notamment sur l'aspect graphique de notre futur jeu ou d'autres fonctionnalités. C'est d'ailleurs durant cette discussion qu'est venue l'idée de rajouter des armes comme c'est le cas dans le jeu Brawlhalla cité plus haut. Forcément, certaines idées partaient un peu trop loin vis-à-vis de nos connaissances, du temps, mais surtout du cahier des charges que l'on risquait de ne pas respecter si on ne faisait pas attention. La seconde étape a donc été de mettre en place un plan le plus détaillé possible de ce dont nous avons besoin pour la mise en place du programme (les variables les plus évidentes, les fonctions, les fichiers, les structures, etc.). L'un de nous était donc chargé d'élaborer le plan le plus détaillé possible tout en sachant qu'il serait changé au fur et à mesure. Les deux autres membres du groupe se sont dans le même temps divisé les tâches entre commencer à coder le menu de présentation du jeu et la construction de nos personnages et de leurs techniques spéciales (d'abord dans un fichier Excel pour avoir une vue globale sur les caractéristiques). Nous nous sommes ensuite réunis pour valider le menu de démarrage, puis pour voir en détail les personnages et leurs attaques spéciales. Au départ, nous avons oublié la fonctionnalité "effet spécial" présente dans le cahier des charges. Alors nous étions d'abord partis sur l'idée de faire pour chaque personnage 2 attaques spéciales, une servant seulement à faire des dégâts et une deuxième pouvant faire des dégâts ou non, mais qui inflige un bonus ou malus de stats à la cible ou au personnage lui-même. Nous abandonnerons cette idée plus tard pour finalement garder une attaque spéciale par personnage, qui fait des dégâts et déclenche un effet spécial. Nous avons ensuite mis en place, à l'aide du plan détaillé, la manière dont on allait stocker les personnages, donc à l'aide de structures qui seraient remplies en début

de jeu par les infos contenues dans des fichiers. Il nous aura fallu une discussion avec notre chargé de TD pour éviter de faire une erreur au niveau des statistiques des personnages. Au départ, nous avons seulement une structure pour les statistiques, mais, puisqu'on allait ajouter des effets spéciaux qui allaient modifier ses statistiques temporairement, il fallait qu'on puisse garder les statistiques d'origine. Nous avons au départ créé deux structures distinctes pour les statistiques de bases et les statistiques temporaires. Puis on a réalisé que seulement une seule structure suffisait et que dans la structure personnage où elle serait ajoutée, on pourrait simplement mettre une variable comme stats temporaire et une autre stats de base, mais qui seraient toutes les deux des structures de structure du même type. C'est ainsi qu'on a poursuivi le projet en mettant en place toutes les bases dont on avait besoin ensemble pour partir du même point. Nous n'avons pas encore fait la partie lecture de fichier qu'on a laissée pour plus tard et nous avons initialisé nos personnages de manière statique dans un tableau de structure pour pouvoir manipuler plus facilement notre code au départ. Nous avons donc mis en place notre logique de combat avec la gestion des esquives, de la vitesse pour rejouer, la limitation pour les attaques spéciales, etc. Lorsque quelqu'un faisait une modification, il l'envoyait aux autres pour qu'ils puissent le tester, et on la mettait seulement sur GitHub une fois que c'était validé et que ça compilait bien. On faisait le maximum pour commenter pour les autres, mais nous avons également effectué des appels sur Discord pour certaines modifications ou explications quand ce n'était pas clair. Au fur et à mesure que le code avançait, il fallait sécuriser au maximum ce qu'on a essayé de faire dès que l'on pouvait, pour ne pas se retrouver avec trop de bugs à corriger en même temps, pour ne pas se perdre.

Conclusion

Ainsi, durant les semaines consacrées à l'élaboration du projet, l'idée de base a évolué et d'autres idées ont fait surface, nous permettant de simplifier ou de rendre plus robuste le programme au bout du temps. Le plus dur est parfois de se méfier de ne pas s'éloigner du cahier des charges, car on s'est parfois rendu compte qu'on oubliait certaines consignes importantes.