

Spécifications détaillées projet autonome PA 2018-2019

1. On veut quoi ?

Un « Tiny » Serious Game. Représenté par un jeu Memory, à destination de la classe senior.

2. Comment ?

Le langage utilisé sera le C#, développé avec le moteur de jeu Unity.

3. But du jeu

Le Serious Game reprendra le contexte du Memory. Le but étant de retrouver toutes les paires de cartes. Dans le cas de ce projet, le jeu se joue tout seul. Le joueur devra trouver toutes les paires de cartes le plus rapidement possible. Plusieurs niveaux seront disponibles. Des cartes seront placées face cachée sur un tableau de jeu. L'utilisateur devra cliquer sur 2 d'entre elles pour checker si ces dernières sont les mêmes. Si c'est le cas, les cartes ne se retournent pas. Sinon, elles se replacent face cachée sur le *board*. Une fois que le joueur a trouvé toutes les paires de cartes, son score final est calculé en fonction du temps utilisé pour finir la partie.

Diverses améliorations seront ajoutées durant le jeu. Plusieurs niveaux seront disponibles en choisissant les images et le nombre de cartes (16, 20 ou 24 cartes).

4. À réaliser

a. Plateforme

i. Tableau de jeu

Choisir une image de fond. Cette image sera posée sur un GameObject représentant la table sur laquelle sera posée les cartes.

ii. Cartes

Une carte sera représentée comme un prefab. Ce dernier sera donc réutilisable et aura son propre script lié. Les cartes auront 2 faces, la face *back* et *front*, respectivement le dos de carte et l'image de la carte.

iii. Actions

Les actions seront simples. Cliquer sur 2 cartes pour checker si elles sont identiques et sélectionner différentes options/niveaux par le biais du menu.

b. Logique du jeu

La logique du jeu consiste au bon déroulement d'une partie. Checker que 2 cartes retournées soient identiques. Effectuer diverses actions en fonction du résultat. Mettre à jour le temps et le score en temps réel. Finir la partie quand toutes les cartes sont retournées.

c. Niveaux

i. Nombre de cartes

Le nombre de cartes sera statique. Il est possible de jouer avec 16, 20 ou 24 cartes sur le board.

i. Images

Le jeu de base inclut 2 niveaux. Ces niveaux seront différenciés par les images représentées sur les cartes. Les images seront spécifiques à chaque niveau et, afin de viser la classe senior, demanderont une certaine perception visuelle afin de les différencier.

b. Menu

Menu de démarrage. Sélection du niveau et du nombre de cartes.

d. Améliorations

- i.** Ajout de modification dans le menu de démarrage
- ii.** Ajout d'un autre niveau
- iii.** Effet pour le retournement de la carte
- iv.** Ajout de musique
- v.** Changement de dos de carte par le joueur
- vi.** Sauvegarde du score

5. Objectifs principaux :

- a.** Création tableau de jeu et carte
- b.** Implémentation script carte
- c.** Implémentation de la logique du jeu
- d.** Menu
- e.** Fin de jeu
- f.** Au moins 2 niveaux

6. Objectifs secondaires :

- a.** Modification du menu
- b.** Ajout de niveau(x) supplémentaire(s)
- c.** Effet retournement
- d.** Musiques
- e.** Amélioration graphique
- f.** Sauvegarde
- g.** Utilisation de la réalité virtuelle

7. Points chauds :

Les points chauds sont les points sur lesquels je pense rencontrer le plus de difficulté. Le plus difficile sera probablement la logique du jeu. Comment checker que 2 cartes soient identiques. Un autre point chaud serait dans l'ajout de niveaux. Est-il possible de développer l'application de façon à faciliter l'ajout de niveaux ?

Aussi, je n'ai jamais développé un projet utilisant un menu de démarrage. Est-ce simple de gérer les actions faites par l'utilisateur ?

- a.** Logique du jeu
- b.** Ajout niveau(x)
- c.** Effet *flip* cartes
- d.** Menu du jeu
- e.** Sauvegarde du score