

# Troisième rapport de projet Arduino

## **EN COURS :**

Nous avons déterminé le design du menu « Equipe », qui permet au joueur de changer d'Arduimon en combat. Pour cela, il nous faut afficher à l'écran la totalité des Arduimon que possède le joueur ainsi que leurs caractéristiques : nom, niveau, point de vie, barre représentative de ces mêmes points de vie ainsi qu'une image miniature de l'Arduimon en question pour bien se les représenter.

Cela nous a pris beaucoup de temps, car d'un côté le dessin d'un tel menu est très minutieux quand Matéo s'en occupe, et de l'autre il nous fallait modifier une partie importante du code actuel pour intégrer ce menu de façon fluide aux combats, qui fonctionnaient totalement jusqu'alors.

Nous avons aussi récupéré un module de lecteur de musique que nous a remis notre enseignant ; nous l'utiliserons dans la semaine afin de déterminer s'il est simple de l'utiliser correctement ou non, et si la place ne manquera pas car rappelons-le : nous cherchons à créer une console portable.

## **HORS COURS :**

Accrochez-vous, 2 semaines de travail vont être résumées.

D'une part, nous avons résolu tous nos problèmes de hardware. Il nous reste encore une dizaine de pourcent d'espace libre dans la mémoire RAM de notre carte, sachant que le code n'est pas du tout optimisé (pour le moment). La carte a été réduite : nous sommes passés à un format de 400x400 pixels, ce qui nous a permis de gagner ce précieux espace dans la mémoire.

En parlant de carte, des choix de créations se sont imposés : puisque nous avons réduit la taille de notre carte, nous devons en faire plus, c'est-à-dire des cartes (ou tableaux) plus petites mais en plus grandes quantité. C'est un procédé simple dès lors que l'on peut lire et écrire depuis la carte SD dans nos variables, ce qui est déjà possible depuis le dernier rapport. Nous vous ferons une démonstration quand nous aurons un second tableau achevé, car en effet la première carte est terminée ! Cela a pris un temps monstrueux au graphiste du groupe (qui n'est pas moi), mais désormais nous pouvons nous mouvoir dans un monde plus vrai, qui respire la fraîcheur des îles.

Vous pouvez retrouver le fichier qui nous sert de carte dans les documents complémentaires de notre GitHub commun.

Il nous est également possible de modifier les collisions de la carte, fonction qui va de pair avec la modification du tableau actuel, mais de la même manière nous ferons une démonstration quand nous aurons la seconde et dernière partie du village qui sera achevée.

Les collisions seront peut être amenées à être modifiées, car elles nous servent actuellement de repère (quel objet ou individu est à quel emplacement). Or, nous pensons coder des classes représentant les objets à l'avenir, ce qui pourrait réduire les collisions à ce qu'elles devraient être, de simple différenciation de collision avec la carte ou non.

Les combats entre équipes sont également fonctionnels désormais ! S'il était avant possible de participer à des combats textuellement, nous pouvons désormais combattre graphiquement ! Cela nous a demandé un temps fou, vraiment, car l'ancien code n'était pas du tout adapté à ce genre de fonctionnalité graphique, et il a fallu recoder l'entièreté du système de combat de zéro, une partie de plaisir (ironie évidente). Il est possible d'attaquer et de fuir. L'utilisation d'objet n'est pas encore utilisable, puisque le principe même d'objet n'est pas encore codé, et le changement d'Arduimon en combat est une mécanique que nous sommes sur le point d'implanter (cf : **EN COURS** ).

Cet affichage de combat est d'ailleurs dynamique : les sprites des Arduimon et étudiants dépendant de l'adversaire, mais également leur niveau, nom, barre de vie. Il en va de même pour nos caractéristiques, à la seule différence que nos points de vies sont affichés, tout comme notre barre d'expérience (une caractéristique qui augmente quand on vient à bout d'un Arduimon, et qui permet la montée de niveau à un certain pallié. La fonction d'expérience est une fonction polynomiale de degré 3).

Enfin, une démonstration de cette fonctionnalité est disponible en privé sur youtube au lien suivant : [https://studio.youtube.com/video/VfMp\\_SocsCI/edit](https://studio.youtube.com/video/VfMp_SocsCI/edit).

Pour finir, j'aimerais simplement expliquer brièvement les difficultés rencontrées pour le code relatif au combat, et pourquoi le code est plus gros que ce qu'il pourrait être :

Une mécanique d'Arduino très intéressante est la fonction `loop()`, qui boucle dès qu'elle le peut, selon la fréquence du Hardware. Dans le cas de nos combats, outre les animations et les textes qui apparaissent, lorsque nous arrivons au choix de l'action le code n'est pas « bloqué » dans une fonction quelconque, mais toute la `loop()` boucle correctement. Seulement, aucune action n'est envisagée sans que nous ayons touchés aux boutons. Ce mécanisme est essentiel pour notre code, car s'il nous fallait ajouter quelque chose qui a besoin de s'appeler fréquemment à côté, il serait impensable que cela soit impossible à cause de la fonction `loop()` qui reste bloqué à attendre que nous lui envoyons une instruction. C'est pourquoi nous devons définir des variables et des timers, afin de ne pas avoir à attendre une instruction pour faire avancer le code.

Si vous avez la foi d'en découvrir plus, je pourrai partager mon code avec vous, mais le bon sens veut que je vous épargne cette torture. Retenez simplement que le code a dû être complexifié afin de permettre à cette fonction `loop()` de ne pas entrer dans des `while` ou autres méthodes « bloquant » un bout du code.

Nous avons aussi optimisé certaines fonctions et rendu d'autres moins lourdes pour la Teensy, notamment la lecture d'images depuis la carte SD qui n'est plus soumise à un `switch()` pour déterminer quelle image ouvrir, mais qui va dynamiquement créer la bonne chaîne de caractère relative au nom du bon fichier. Cela nous a permis de gagner en rapidité ainsi qu'en place, car chaque case : d'un `switch()` occupe la mémoire RAM d'un microcontrôleur sous Arduino.

## **BIBLIOGRAPHIE :**

Nous avons préparé plusieurs documents sources qui viennent appuyer nos rapports. Vous pouvez dès à présent jeter un coup d'œil à différentes photos et vidéos appuyant nos hauts faits sur la branche « feat » de notre GitHub, dans le dossier « documents supplémentaires ».

De plus, une vidéo illustrant la fonctionnalité de combat est présente à l'adresse suivante : [https://www.youtube.com/watch?v=VfMp\\_SocsCI&feature=youtu.be](https://www.youtube.com/watch?v=VfMp_SocsCI&feature=youtu.be).