

BRUTIER Mélina  
HOUEL Lucie  
PICQUART Tom  
SERAPHIN Bradley



## Escape game : Storyteller

## Sommaire :

Introduction

I - Création du PCB

- I.1 Choix des composants
- I.3 KiCad
- I.2 Impression et soudage

II - Partie informatique

- II.1 Capteur RFID
- II.2 Ecran LED
- II.3 Servomoteur
- II.4 Programme principal

III - Impression 3D

IV - Partie narrative

- IV.1 Crédit de l'histoire
- IV.2 Les personnages
- IV.3 Les différentes pièces

V - Ressentie de chacun sur ce projet

Conclusion

# Introduction

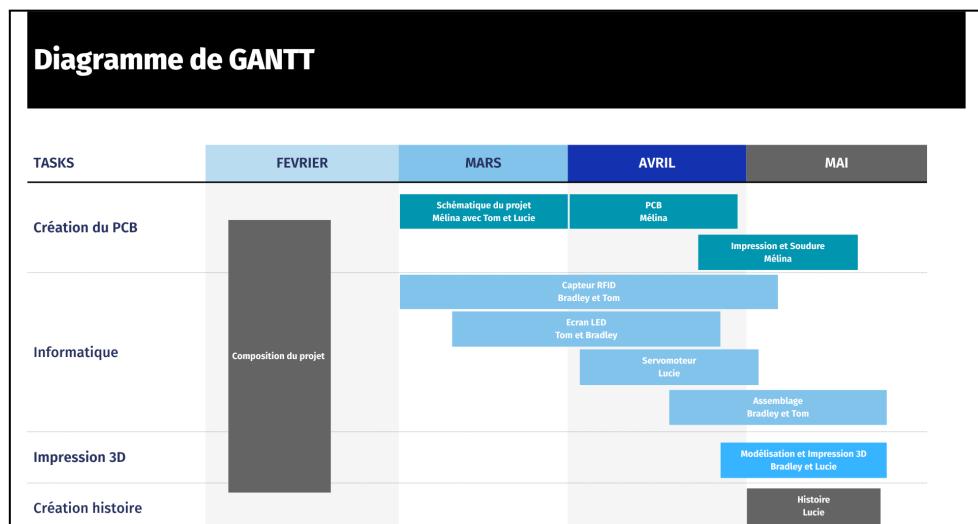
Dans le cadre de ce projet, nous devions concevoir une étape pour un escape game. Nous avons opté pour un jeu interactif mêlant narration et réflexion. Inspiré du jeu de logique *storyteller*, notre dispositif propose une enquête policière se déroulant dans un manoir, au fil de plusieurs scènes découpées en trois parties de difficulté croissante. À chaque étape de l'histoire, le joueur doit analyser les témoignages et indices, puis reconstituer la scène en plaçant les personnages dans les bonnes pièces de la maison.

Pour cela, chaque personnage est représenté par une carte illustrée équipée d'une puce RFID, dite carte RFID. Le joueur choisit sur un écran la pièce dans laquelle il pense que le personnage se trouve, puis présente successivement chaque carte devant un capteur RFID RC522. Après appuie sur le bouton VALIDATION, si tous les personnages sont bien placés, une lumière verte s'allume, signalant le passage à la scène suivante. Sinon, une lumière rouge s'allume, le jeux se reset et le joueur doit réessayer, s'il se trompe il peut appuyer sur le bouton RESET et recommencer

Le système repose sur plusieurs modules développés séparément : lecteur RFID, écran LED, gestion du jeu, et contrôle d'un servomoteur (prévu pour libérer une clé pour finir l'escape game complet). Une carte électronique (PCB) dédiée a également été conçue spécifiquement pour le projet, afin d'intégrer proprement les différents composants matériels.

Enfin une modélisation 3D du manoir a été faite pour rendre plus interactif et visuel notre jeu. Des éléments visuels (illustration de l'histoire, cartes, pièces du manoir) ont aussi été imprimés et pour être fixés de manière amovible, afin de permettre une évolution ou un remplacement simple.

Ce rapport présente les différentes étapes du développement du projet, du scénario narratif à l'assemblage technique, en passant par les choix de conception, la réalisation du PCB, les illustrations et les contraintes rencontrées.



# I - Création du PCB

## I.1 Choix des composants

Lors de la réalisation de notre PCB, nous avons dû utiliser différents matériaux et composants. Afin de limiter les coûts et les délais, nous avons privilégié les composants déjà disponibles à l'école, ce qui nous a permis d'éviter toute commande extérieure.

### **Microcontrôleur**

Initialement, nous avions prévu d'utiliser un STM32L021K4Tx. Cependant, ce microcontrôleur n'était plus disponible au moment de la soudure. Après avoir vérifié la compatibilité des broches, nous avons identifié un modèle équivalent : le STM32L412KBTx, que nous avons finalement utilisé sur le PCB.

### **Régulateur de tension**

Pour convertir la tension de 5V en 3,3V, nécessaire au bon fonctionnement de certains composants, nous avons utilisé un BU33SD5WG-TR.

### **Servomoteur**

Le servomoteur utilisé est un 6800 MGA Protronic, adapté à notre application en termes de couple et de dimensions.

### **Capteur RFID**

Nous avons utilisé le module RFID-RC522, couramment utilisé, que nous avons trouvé sur internet.

### **Connecteurs**

Pour relier les différents modules externes au PCB, nous avons utilisé des connecteurs à broches adaptés au nombre de signaux nécessaires. Nous les avons choisis en fonction du nombre de pins requis pour chaque composant.

## Autres composants

En plus des éléments principaux, notre PCB intègre également :

Un bouton-poussoir pour les interactions avec l'écran.

Un écran pour l'affichage des informations

Une LED utilisée comme indicateur pour le résultat de l'énigme.

Voici ci dessous les empreintes utilisés sur le KiCad :

Symbole: Attribution Empreintes	
1	C1 -
2	C2 -
3	C3 -
4	C4 -
5	C5 -
6	C6 -
7	C7 -
8	C8 -
9	D1 -
10	D2 -
11	D3 -
12	D4 -
13	D5 -
14	D6 -
15	H1 - MountingHole_Pad : MountingHole:MountingHole_3.2mm_M3_DIN965_Pad
16	H2 - MountingHole_Pad : MountingHole:MountingHole_3.2mm_M3_DIN965_Pad
17	H3 - MountingHole_Pad : MountingHole:MountingHole_3.2mm_M3_DIN965_Pad
18	H4 - MountingHole_Pad : MountingHole:MountingHole_3.2mm_M3_DIN965_Pad
19	J1 - Conn_01x03_Pin : Connector_PinHeader_2.54mm:PinHeader_1x03_P2.54mm_Horizontal
20	J2 - Conn_02x07_Odd_Even : Connector_PinHeader_1.27mm:PinHeader_2x07_P1.27mm_Vertical_SMD
21	J3 - Conn_OLED_Display : Connector_JST:JST_NV_B04P-NV_1x04_P5.00mm_Vertical
22	J4 - Conn_01x02 : Connector_JST:JST_XH_B2B-XH-A_1x02_P2.50mm_Vertical
23	L1 -
24	M2 - RFID-RC522 : Connector_PinHeader_2.54mm:PinHeader_1x08_P2.54mm_Vertical
25	R1 -
26	R2 -
27	R3 -
28	R4 -
29	R5 -
30	R6 -
31	R7 -
32	R8 -
33	R9 -
34	R10 -
35	R11 -
36	SW1 -
37	SW2 -
38	SW3 -
39	SW4 -
40	U1 -
41	U2 -
	4u7 : Capacitor_SMD:C_0805_2012Metric_Pad1.18x1.45mm_HandSolder
	100n : Capacitor_SMD:C_0805_2012Metric_Pad1.18x1.45mm_HandSolder
	100n : Capacitor_SMD:C_0805_2012Metric_Pad1.18x1.45mm_HandSolder
	1u : Capacitor_SMD:C_0805_2012Metric_Pad1.18x1.45mm_HandSolder
	100n : Capacitor_SMD:C_0805_2012Metric_Pad1.18x1.45mm_HandSolder
	10n : Capacitor_SMD:C_0805_2012Metric_Pad1.18x1.45mm_HandSolder
	1uF : Capacitor_SMD:C_0805_2012Metric_Pad1.18x1.45mm_HandSolder
	1uF : Capacitor_SMD:C_0805_2012Metric_Pad1.18x1.45mm_HandSolder
	LED_Small : LED_SMD:LED_0805_2012Metric_Pad1.15x1.40mm_HandSolder
	MountingHole_Pad : MountingHole:MountingHole_3.2mm_M3_DIN965_Pad
	Conn_01x03_Pin : Connector_PinHeader_2.54mm:PinHeader_1x03_P2.54mm_Horizontal
	Conn_02x07_Odd_Even : Connector_PinHeader_1.27mm:PinHeader_2x07_P1.27mm_Vertical_SMD
	Conn_OLED_Display : Connector_JST:JST_NV_B04P-NV_1x04_P5.00mm_Vertical
	Conn_01x02 : Connector_JST:JST_XH_B2B-XH-A_1x02_P2.50mm_Vertical
	Inductor_SMD:L_0805_2012Metric_Pad1.15x1.40mm_HandSolder
	RFID-RC522 : Connector_PinHeader_2.54mm:PinHeader_1x08_P2.54mm_Vertical
	R_Small : Resistor_SMD:R_0805_2012Metric_Pad1.20x1.40mm_HandSolder
	SW_SPST : Button_Switch_SMD:SW_SPST PTS647_Sx50
	STM32L021K4Tx : Package_QFP:LQFP-32_7x7mm_P0.8mm
	BUG33SD5WG-TR : Package_TO_SOT_SMD:SOT-23-5

## I.3 KiCad

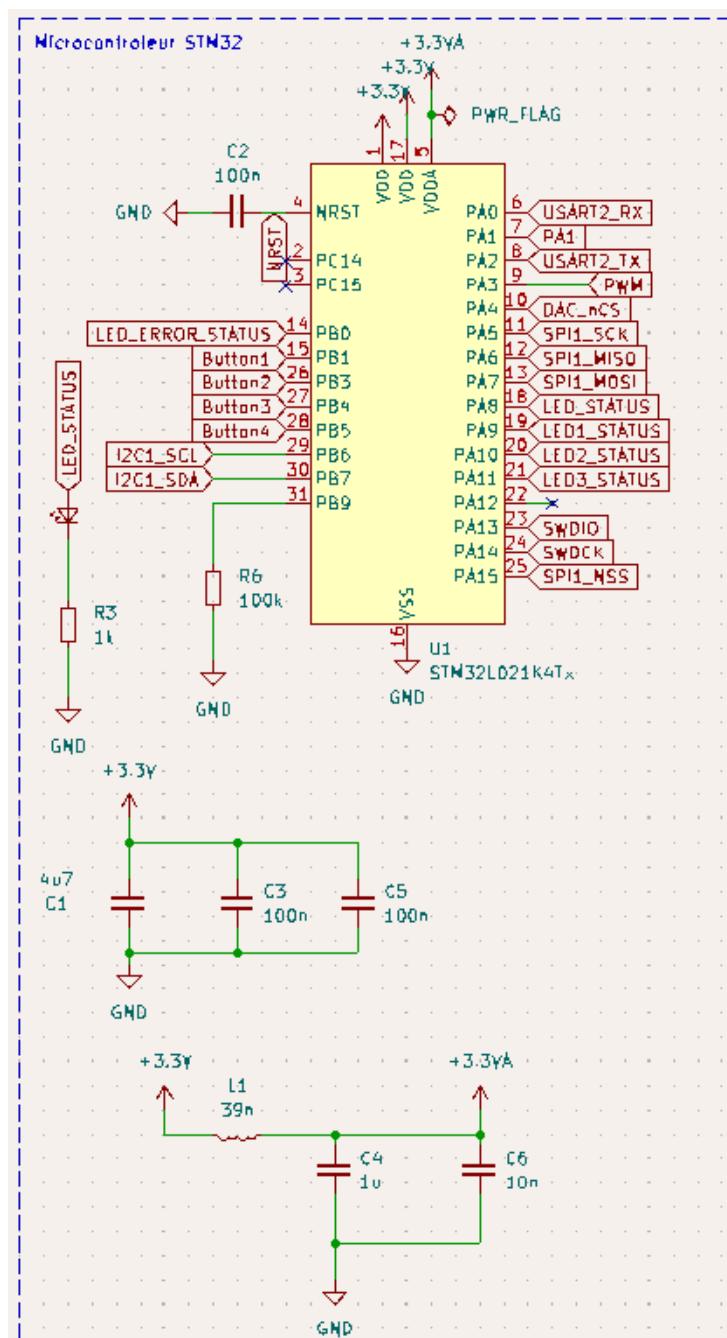
Le PCB a été créé sur KiCad à partir d'une carte microcontrôleur STM32 à laquelle on a branché tous les éléments nécessaires, c'est-à-dire : un régulateur, un moteur, un écran, un capteur RFID, des boutons, des LEDs et des connecteurs.

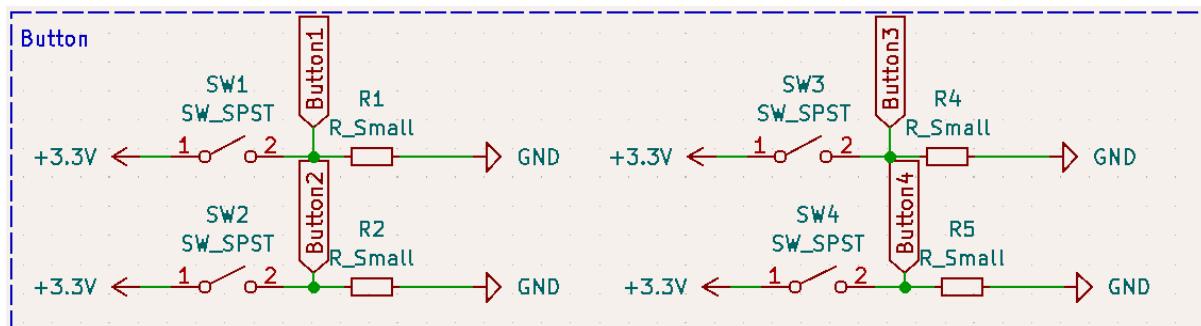
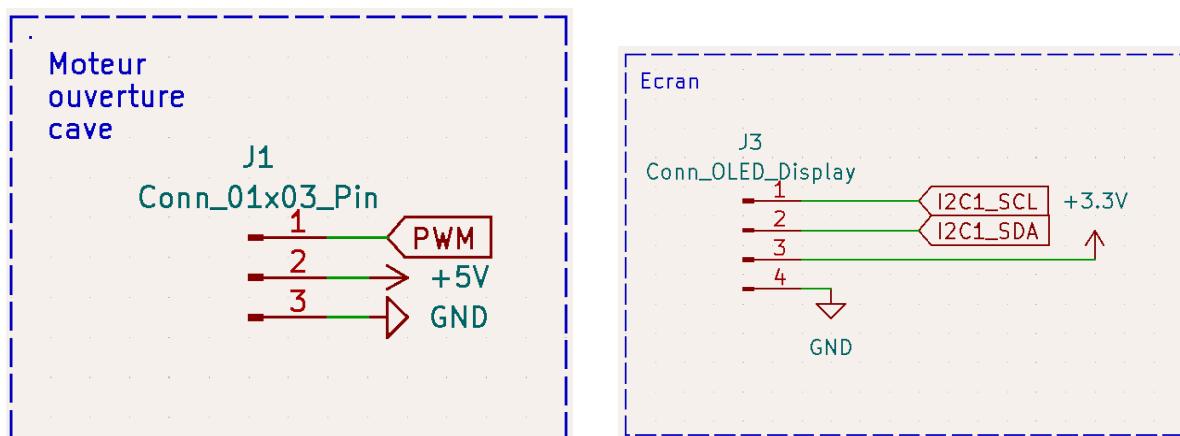
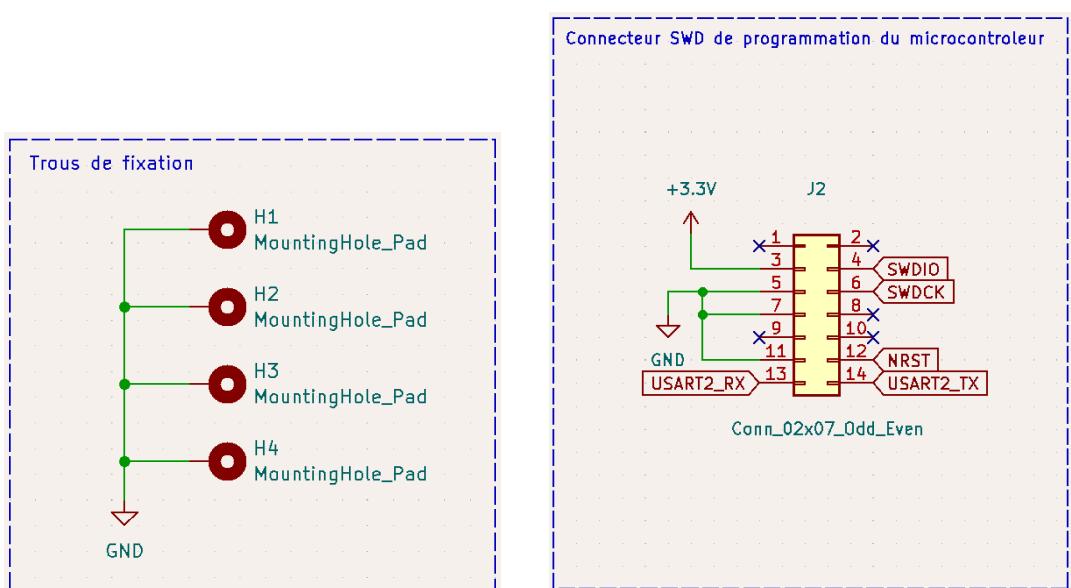
L'écran va permettre d'afficher les consignes du jeu. Et de choisir dans quelles pièces on souhaite mettre les personnages.

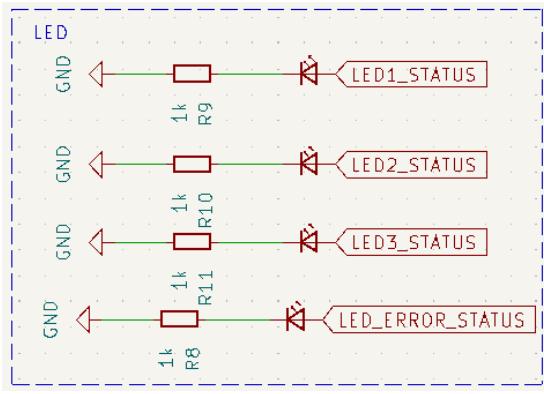
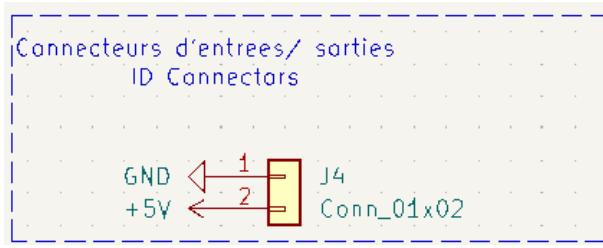
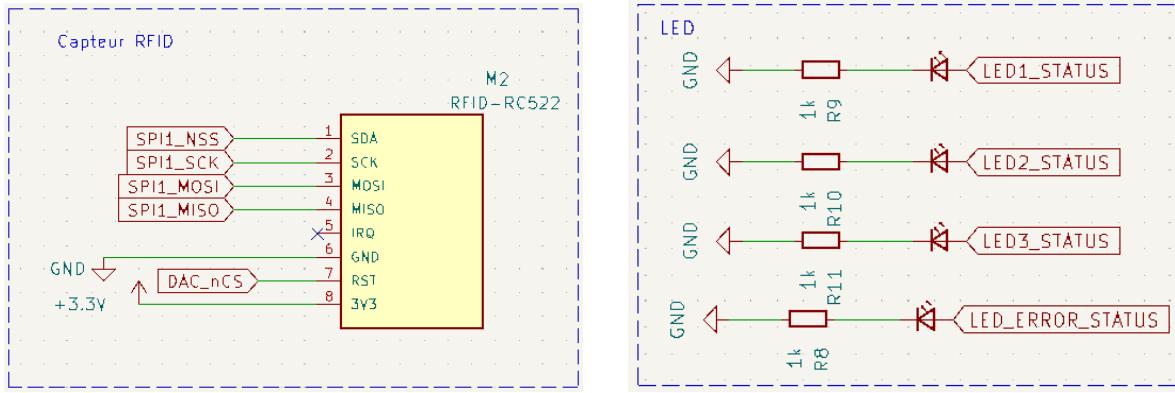
Les leds et les boutons serviront à valider et savoir si la réponse est correcte.

Une led verte s'allumera en cas de bonne réponse et une led rouge dans le cas contraire.

De plus, les fichiers Kicad sont présents sur le GitHub.



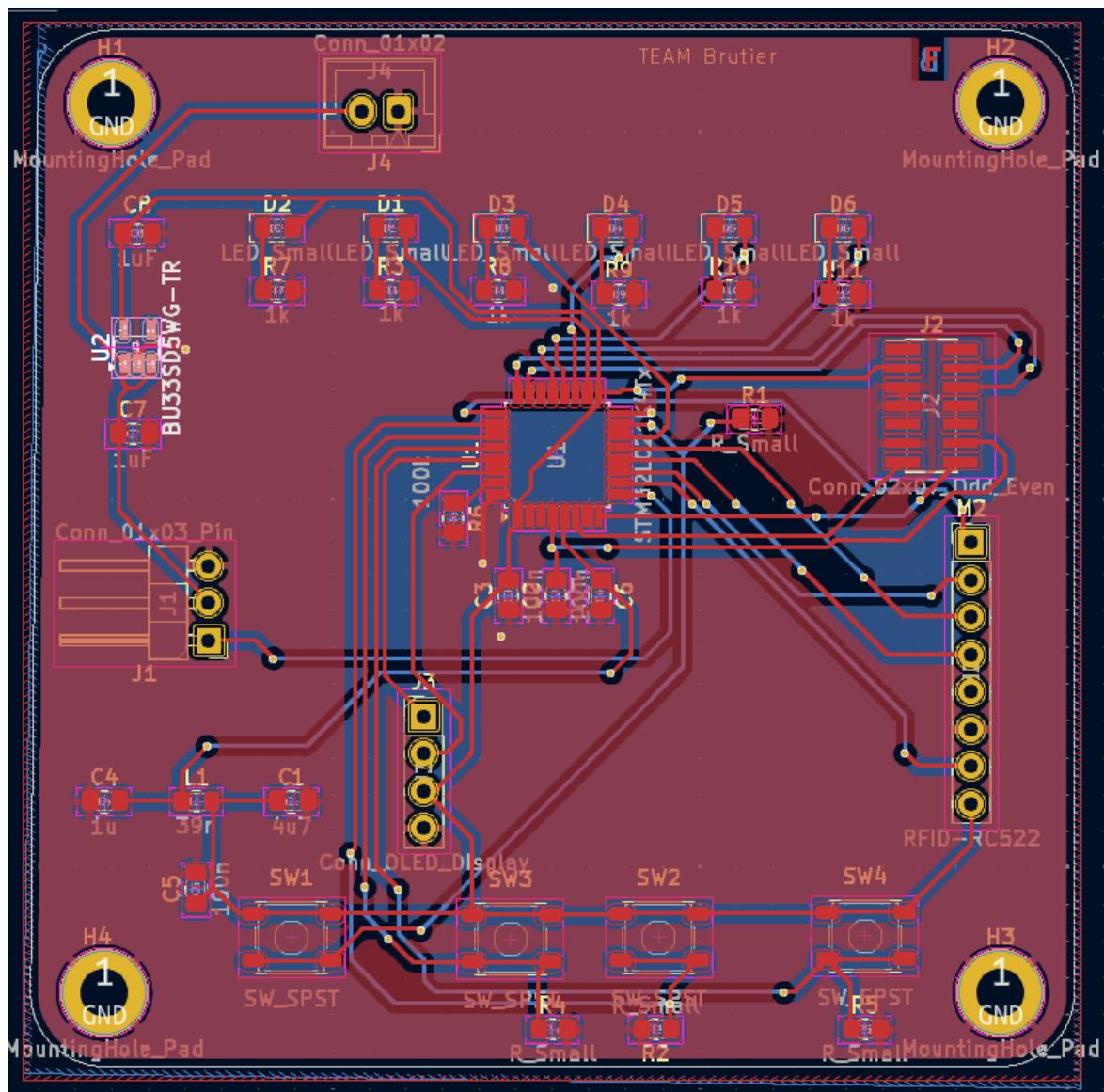


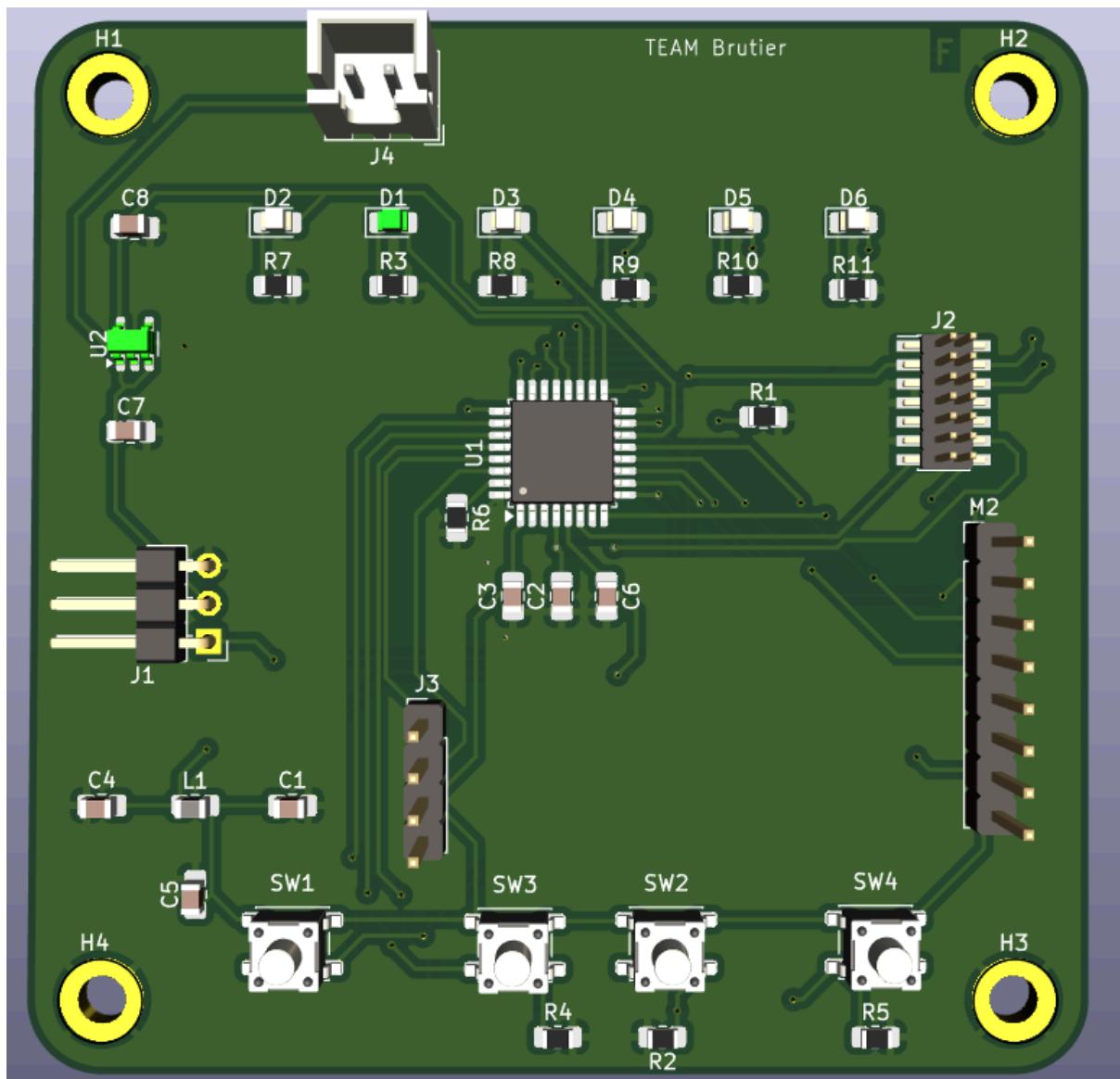


Après avoir créé toutes les liaisons entre les différents composants, nous réalisons le tracé sur le PCB qui fait 7,5 cm par 7,5 cm.

Les routages ont été tracés en 0,3 mm pour que le PCB puisse être imprimé en interne à l'école.

La plupart des composants sont des composants CMS (Composants Montés en Surface) aussi appelés SMD (Surfaces Mount Devices), c'est-à -dire qu'ils sont non traversants. Cela a permis de réduire la taille des circuits imprimés, seuls les connecteurs sont traversants.

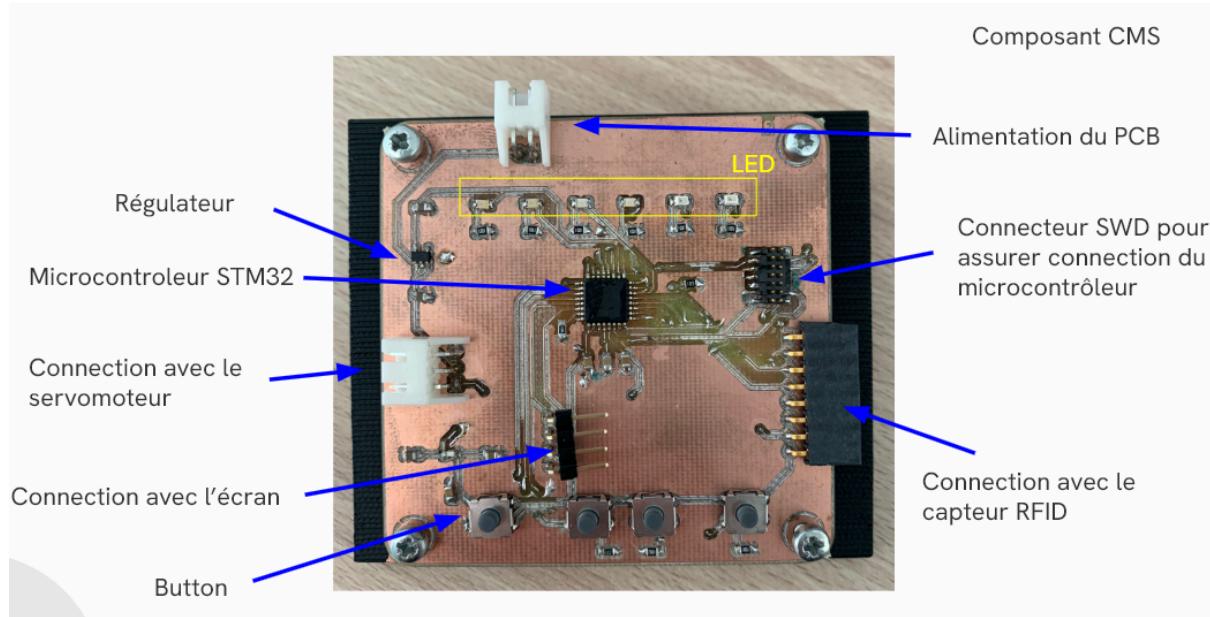




## I.2 Impression et soudage

Lorsqu'on a voulu réaliser l'impression du PCB, la machine d'impression de l'école présentait des dysfonctionnements au niveau de la métallisation des vias. En conséquence, nous avons été contraints de souder manuellement tous les vias sur le PCB, ce qui explique le temps important nécessaire pour réaliser la soudure.

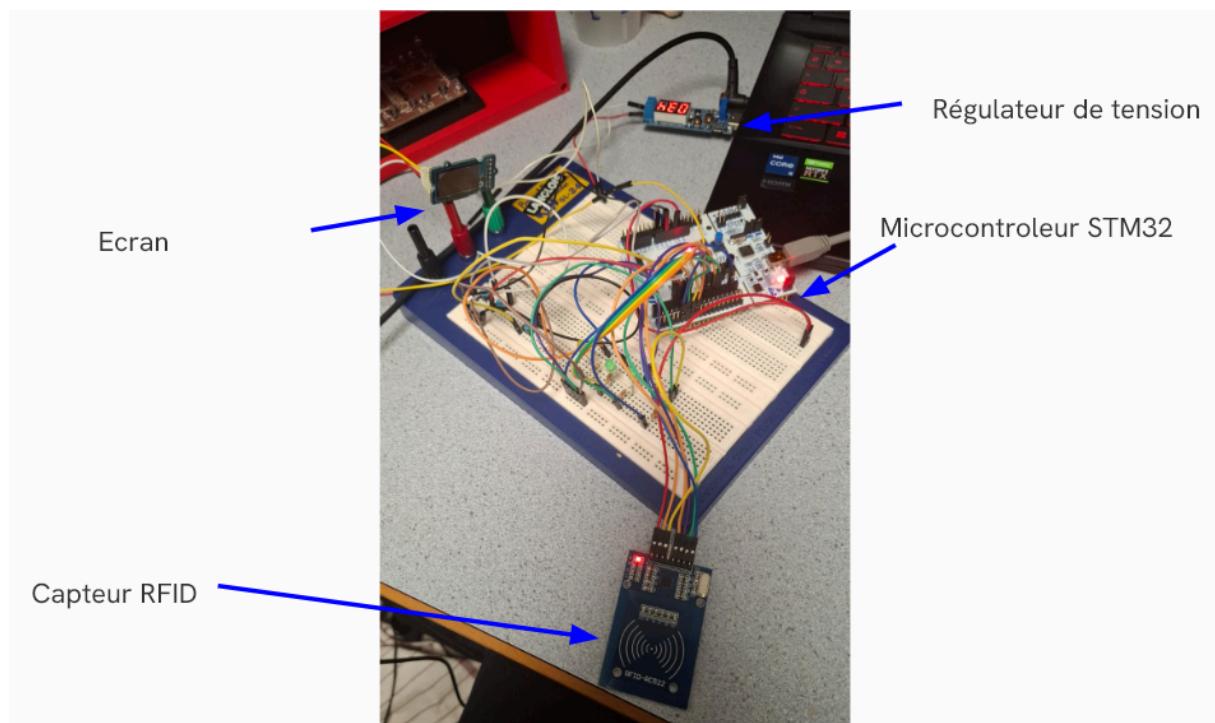
Vous trouverez ci-dessous une photo de notre PCB une fois imprimé et soudé.



Cependant, lorsqu'on a voulu essayer d'envoyer le code informatique sur le PCB, un message d'erreur s'affichait : no device found target.

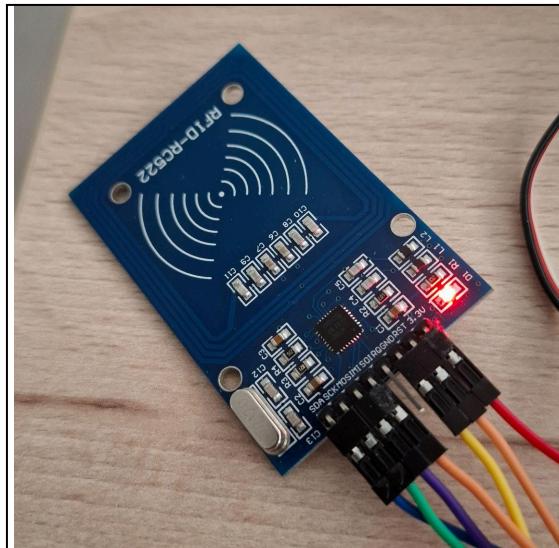
N'ayant pas pu régler ce problème à temps, nous avons donc utilisé un breadboard. On a pu connecter tous nos composants et mettre aussi les LED de différentes couleurs.

Grâce à ce breadboard nous avons réussi à faire fonctionner l'énigme et ainsi pouvoir faire jouer les joueurs.



## II - Partie informatique

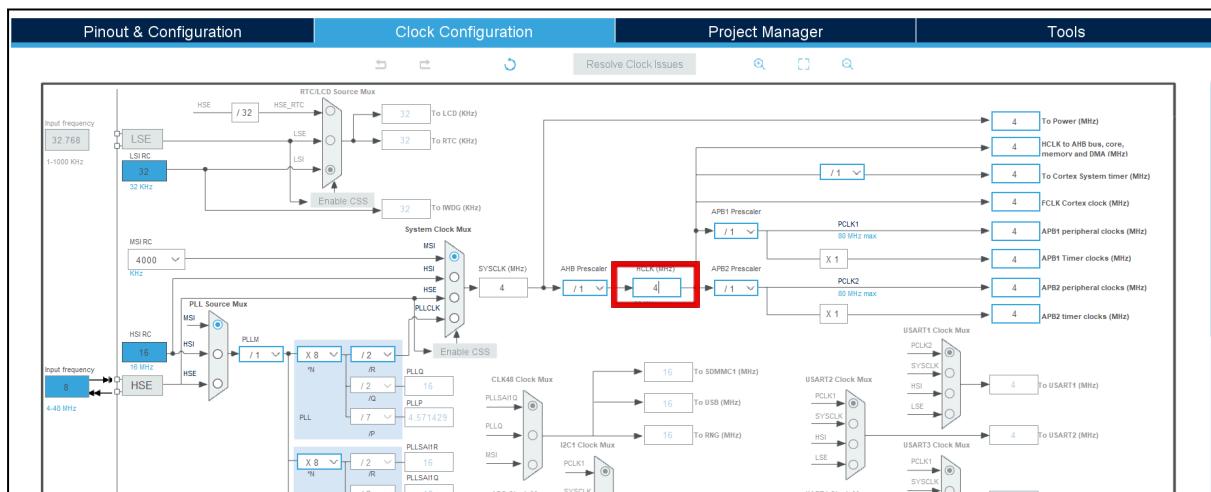
## II.1 Capteur RFID RC522



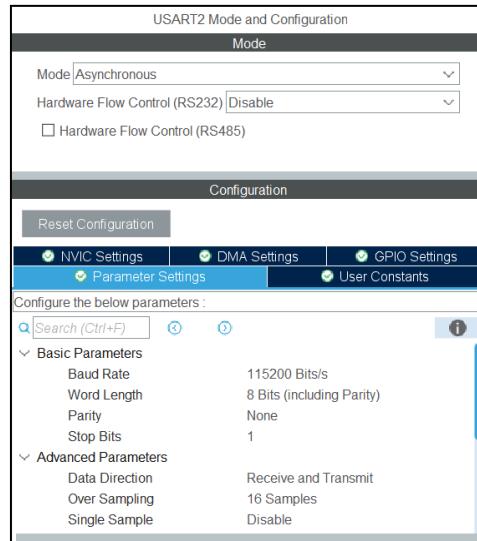
Pour commander le capteur RFID RC522 nous avons suivis les instructions d'une vidéo sur Youtube, [cliquez sur le lien](#).

Voici les instructions pour utiliser le capteur RFID RC522 avec la STM32L476RG que nous utilisons en TP:

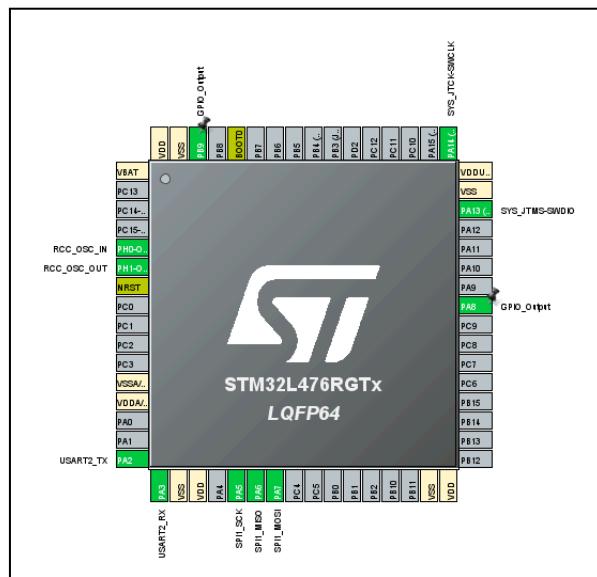
- Créer un nouveau projet, pas d'un projet existant.
  - Dans l'IOC, vérifier que tous les pin sont reset.
  - Dans System Core>RCC>HSE: Mettre Crystal/Ceramic Resonator
  - System Core>SYS>Debug: Serial Wire
  - Vérifier que HCLK vaut 80MHz.



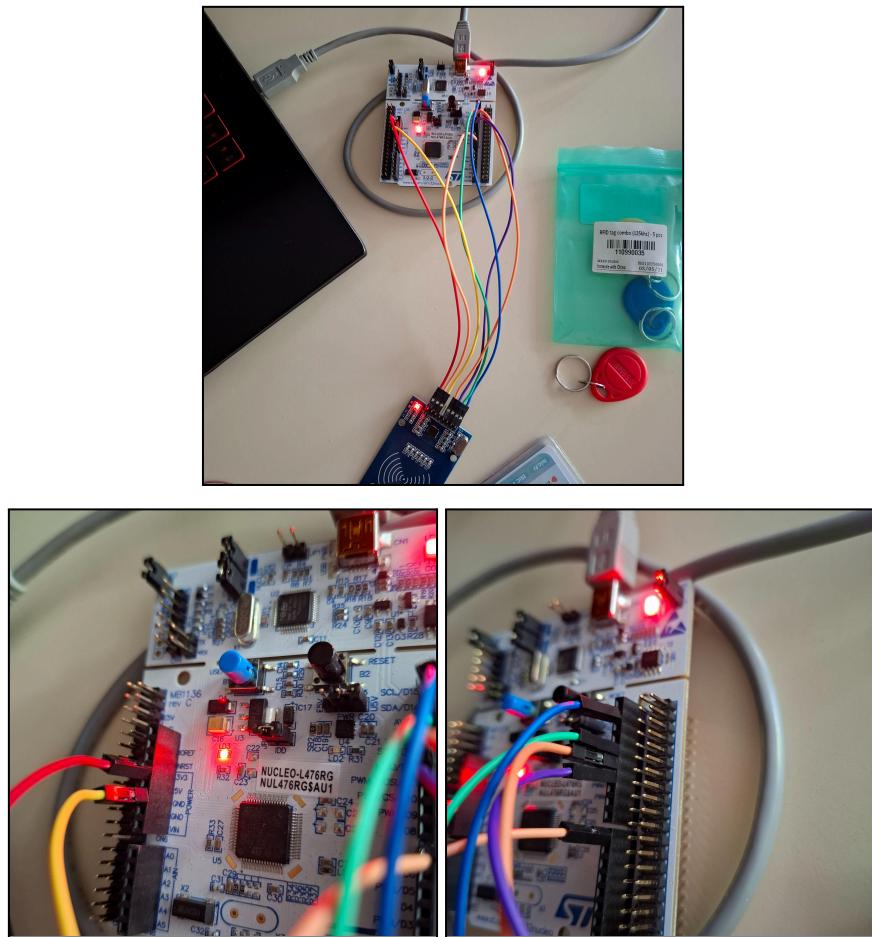
- Connectivity>USART2>Mode: Asynchronous + Vérifier le baud rate de 115200 Bits/s



- Enfin, sélectionner PA8 et PB9, et définissez les en GPIO\_Output



- A l'aide du mapping de la STM32L476RG, on réalise les branchements



- Pour le code, récupérer le fichier .c et .h RC522 sur le GitHub [ci-joint](#) et insérez-les dans votre projet. Puis dans le main.c.
- Dans Private include :

```
#include "RC522.h"
#include "string.h"
```

- Dans Private variables :

```
uint8_t status;
uint8_t str[16];
uint8_t sNum[5];
```

- Dans les initialisations :

```
MFRC522_Init();
status1=MFRC522_Request(PICC_REQIDL,str);
status2=MFRC522_Anticoll(str);
memcpy(sNum,str,5);
```

- Avec un printf vous pouvez lire les différentes valeurs de str soit 5 valeurs et ainsi savoir les informations des cartes. A vérifier avec une application sur mobile

## II.2 Ecran LED

Les personnages et leur positions sont gardés dans des listes:

```
char* pieces[6]={"salon","cuisine","bibliot","bureau","chambre","couloir"};
char* personages[6]={","","","","","",""};
```

Pour faciliter l'affichage, le personnage le plus récent est gardé dans:

```
char* perso;
```

De même le nombre de personnage dans chaque pièce est gardé dans:

```
int nb_personages[6]={0,0,0,0,0,0};
```

L'écran LED sert à afficher la pièce ainsi que les personnages présents dans celle-ci. Il ne se met à jour que si un nouveau personnage est ajouté ou si la pièce est changée. Ces cas sont vérifiés par les variables Card\_Passed et button\_pressed.

```
if (Card_Passed==1) {
    Card_Passed=0;
    char buffer[16]={0};
    sprintf(buffer,16,"%s ",(pieces[i]));
    ssd1315_Draw_String(0,0,buffer,&Font_11x18); // la pièce est
    affichée en haut et en gros
    sprintf(buffer,16,"%s ",perso); // L'écran n'étant réinitialisé que lors d'un
    changement de pièce il suffit d'ajouter le personnage le plus récent

ssd1315_Draw_String(0,10+(10*nb_personages[i]),buffer,&Font_7x10); // les
personnages sont affichés les uns en dessous des autres
ssd1315_Refresh();}
```

Lors d'un changement de pièce, il faut réinitialiser l'écran et afficher les personnages déjà présents dans la pièce:

```
if (button_pressed==1) {
    char buffer[16]={0};
    ssd1315_Clear(SSD1315_COLOR_BLACK);
    sprintf(buffer,16,"%s ",(pieces[i]));
    ssd1315_Draw_String(0,0,buffer,&Font_11x18);
    int len=strlen(personages[i]);
    for (int n=0;n<len;n++){ // l'affichage n'est pas idéal (échec
    d'écriture similaire au cas Card_Passed, ici les personnages sont affichés
    sur une seule ligne)
        sprintf(buffer,16,"%c ",(personages[i][n]));
        ssd1315_Draw_String(7*n, pos, buffer, &Font_7x10); // la
variable pos est un reste des tests pour afficher l'écran correctement, elle
aurait été incrémentée lors du passage d'un personnage à un autre dans la
liste des personnages
        ssd1315_Refresh();
        printf("%c \n\r", personages[i][n]);
    }
    ssd1315_Refresh();
}
```

## II.3 Servomoteur

Le servomoteur utilisé dans le projet est un 6800 MGA Protronic. En l'absence de documentation technique, son angle maximal de rotation a été déterminé expérimentalement : des tests progressifs ont permis d'identifier une plage fonctionnelle d'environ 0° à 120°, au-delà de laquelle des blocages mécaniques ou une instabilité apparaissaient.

Le servomoteur est piloté par un signal PWM généré via le canal 1 du Timer 2. Le code suivant calcule la largeur de l'impulsion du signal PWM à appliquer en fonction de l'angle souhaité :

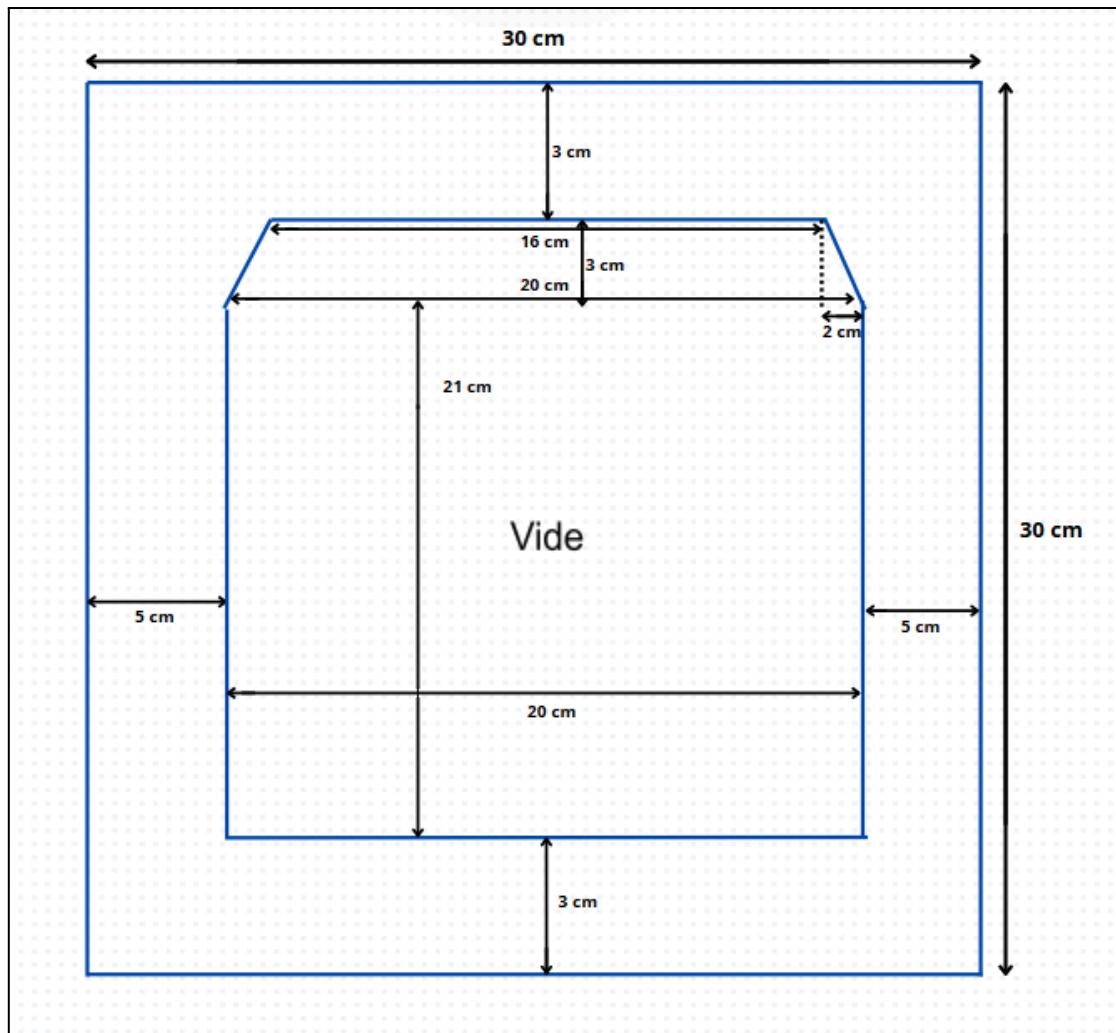
```
void setServo(int angle){  
    int pulsemax = 1900; // Durée de l'impulsion pour l'angle maximal  
(en µs)  
    int pulsemin = 1500; // Durée de l'impulsion pour l'angle  
minimal (en µs)  
    int anglemax = 120; // Angle maximal déterminé expérimentalement  
(en degrés)  
    float r = (float)angle/anglemax;  
    int pulse = pmin+(int)((pulsemax - pulsemin)*r);  
    TIM2->CCR1 = pulse; // Mise à jour du canal 4 du timer 2 avec la  
valeur calculée  
}
```

Ce code permet un positionnement angulaire proportionnel dans la plage admissible du servomoteur.

## II.4 Programme principal

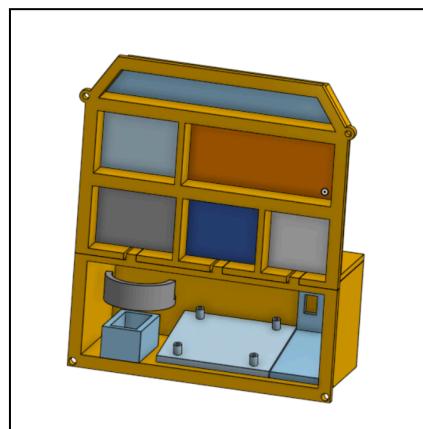
Le programme se trouve dans le GitHub dans un fichier compressé, téléchargez le fichier Projet1A5.0.zip et une fois extrait, utilisez-le avec STM32 IDE.

### III - Impression 3D

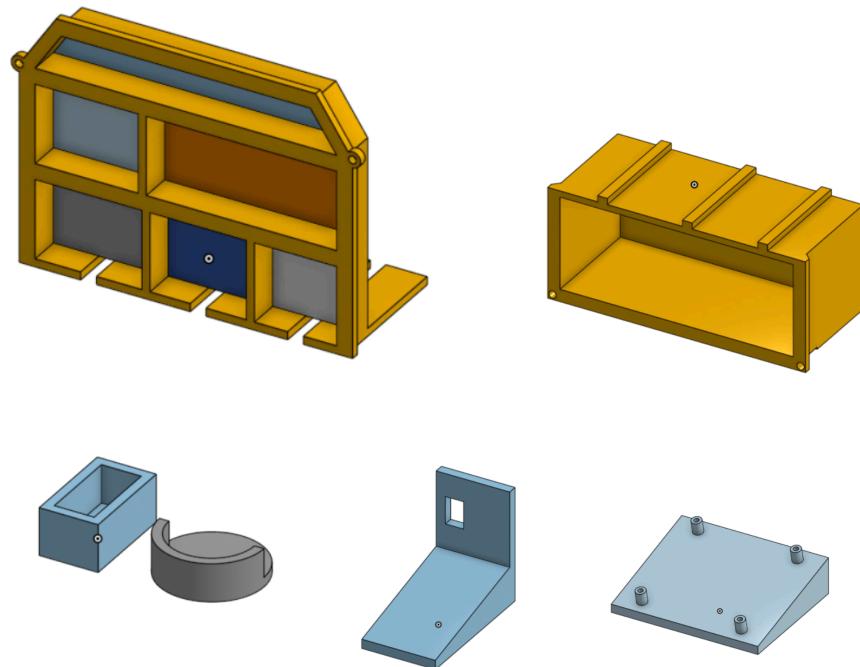


*Face du cube*

Pour l'impression 3D plusieurs parties étaient nécessaires dont les supports pour les composants et la maison en elle-même.



*Assemblage*



### *Toutes les pièces*

Les pièces sont aux format stl dans le [GitHub](#), nous avons utilisé le logiciel Bambu Studio pour les imprimer avec les imprimantes associées.

## IV - Partie narrative

### IV.1 Crédit de l'histoire

L'histoire est construite en trois parties, avec une difficulté croissante pensée pour faire de plus en plus appel à la réflexion des joueurs. La première partie, volontairement simple, permet de prendre ses repères : on découvre les personnages, le cadre, les premiers témoignages. Elle pose les bases du jeu. La seconde, plus difficile, oblige à confronter les différentes versions, à déceler les incohérences, à faire appel à l'observation et à la logique. La dernière partie, enfin, est la plus complexe : les témoignages se nuancent, des objets retrouvés donnent de faux indices, et il devient difficile de distinguer qui ment pour trouver le/la coupable.

### IV.2 Les personnages

Chaque personnage a sa propre carte, comportant son nom, son lien avec la victime, une courte description et un portrait. Ces cartes ont été pensées pour être à la fois lisibles et immersives, en accord avec l'ambiance de l'enquête. Elles sont simplement collées sur leur support, ce qui permet de les modifier ou remplacer facilement en cas de mise à jour du scénario ou d'amélioration graphique.



### IV.3 Les différentes pièces

L'histoire prend place dans un manoir, lui aussi représenté visuellement par une illustration. À l'origine, une maquette 3D du lieu avait été envisagée, mais par manque de temps, elle n'a pas été imprimée, simplement modélisée. L'illustration de la maison a été tout de même été faite, avec les pièces importantes bien identifiées : bureau, bibliothèque, salon, couloirs, chambres... Comme pour les cartes des personnages, cette illustration doit être fixée de façon à pouvoir être modifiée ou remplacée si nécessaire.



*Illustration de la maison*

## V - Ressentie de chacun sur ce projet / Rapport d'étonnement

### Ressentie :

- Mélina : J'ai beaucoup appris durant ce projet notamment sur la création du PCB sur lequel j'ai rencontré de nombreuses difficultés. Cela me permettra ainsi de ne pas reproduire ces erreurs lors des prochains projets.
- Tom : J'ai découvert les complexités liées à la réalisation d'un projet ainsi que l'importance d'une bonne organisation dans un groupe.
- Bradley : Que ce soit en code, modélisation 3D et management j'ai beaucoup appris, ce qui me permettra de mieux entreprendre mes futurs projets.
- Lucie : Ce projet m'a permis de développer mes compétences techniques tout en laissant une place importante à la créativité, ce qui était très ludique.

### Rapport d'étonnement :

On a appris sur le temps de production et sur les deadlines à respecter. L'organisation et l'anticipation sont les aspects majeurs du projet à respecter pour qu'il puisse être fonctionnel à temps.

## Conclusion

Plusieurs points d'amélioration sont possibles :

- Ajout d'un bouton "Admin" pour choisir les histoires ou simplement appuie simultané de plusieurs boutons.
- Intégrer plusieurs histoires, le code de vérification étant simple et l'ajout de nouvelle cartes est défini par leur valeur.
- D'autres actions pour les cartes, par exemple action se laver associé à la pièce salle de bain et qui s'afficherait sur l'écran en plus du personnage.