

Projet de Science de l'ingénieur sur l'appareil de mesure : compte-rendu

Découverte projet 2020

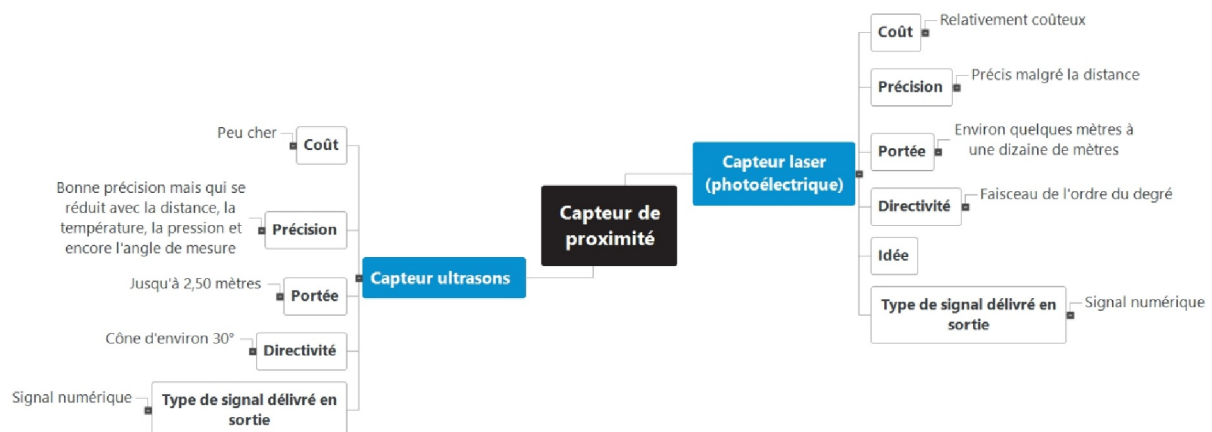
Etape 1 : Intérêt d'un appareil de mesure et des définitions de la mesure

Question 1 :

Grâce à l'organigramme et à nos différentes recherches, nous avons pu trouver 2 types de capteurs possibles : le capteur à ultrasons et le capteur laser (photoélectrique).

Question 2 :

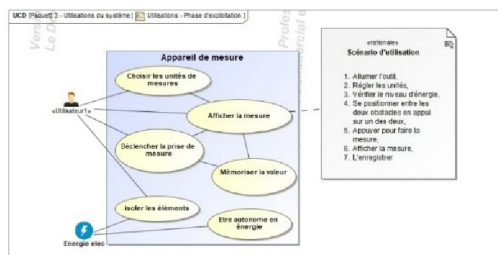
Ce diagramme (ci-dessous) nous montre les 2 types de capteurs (laser et à ultrasons) et leur différentes propriétés



Etape 2 : Comment réaliser l'appareil de mesure à distance

Question 4 :

Sur le diagramme des cas d'utilisation (ci-dessous)



Elève 1 :

Elève 2 :

Elève 3 :

1G3

On décrit les modes de fonctionnement de l'appareil en fonction des cas d'utilisation :

Cas d'utilisation	Mode de fonctionnement
Choisir les unités de mesures	Un bouton à côté d'un écran permettra de sélectionner l'unité de la mesure avant son déclenchement
Afficher les mesures	En appuyant sur un bouton positionner sur le côté, on déclenchera la mesure qui sera affichée sur un écran jusqu'à la prise d'une nouvelle mesure

Question 5 :

Le principal paramètre influent sur la certitude lors de la mesure à l'aide d'ultrason est la température, elle influe sur la compressibilité de l'air et donc sur la vitesse de propagation du son (cette vitesse peut-être déterminer par la formule suivant : $c = 20,05 \sqrt{T}$)

Question 6 : déterminer les critères à ajouter pour assurer une baisse de l'incertitude de la mesure.

-Répéter plusieurs fois la mesure pour en faire une moyenne (pas top)

-Ajouter une incertitude de +/- 3cm sur l'écran

-Ajouter un laser sur au niveau du capteur pour faciliter la visée de l'utilisateur

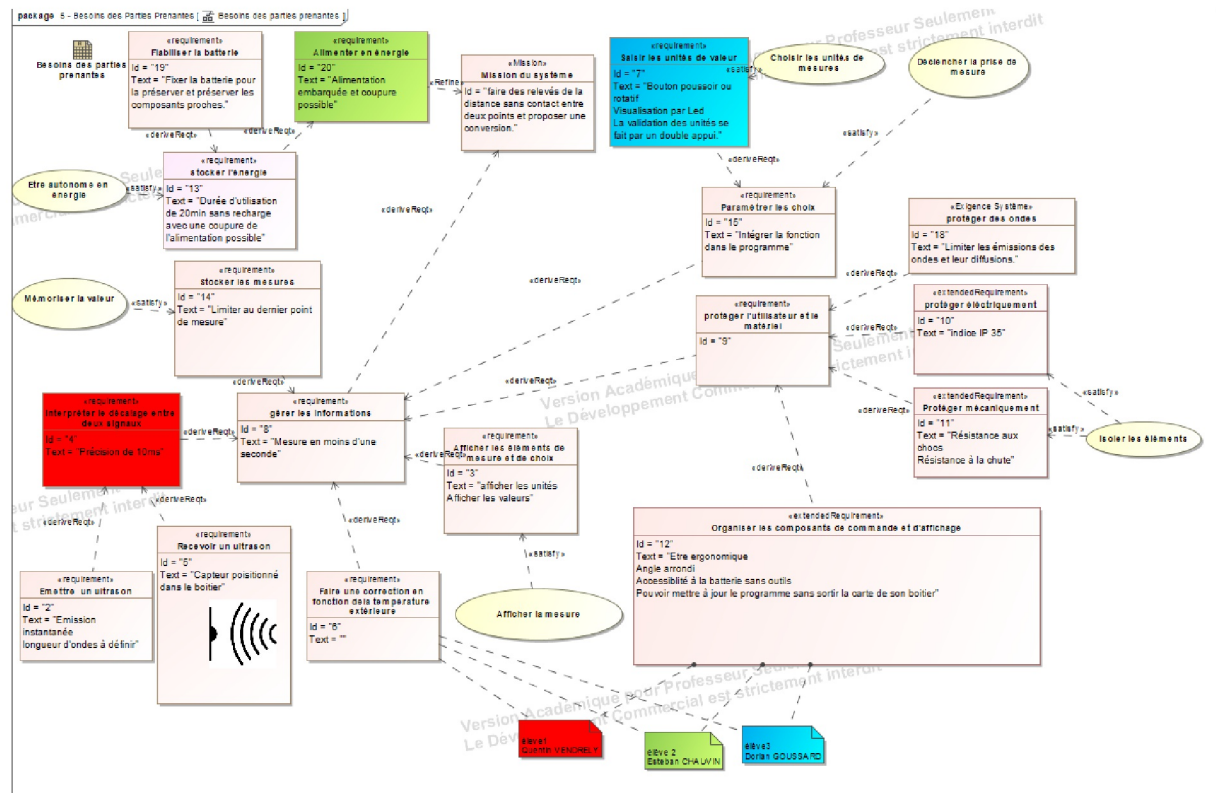
-Ajouter capteur de T° pour adapter la certitude des mesures en fonction de la température :

Influence de la température sur l'air			
θ en °C	c en m·s ⁻¹	ρ en kg·m ⁻³	Z en N·s·m ⁻³
- 10	325,4 325,4	1,341	436,5
- 5	328,5 328,5	1,316	432,4
0	331,5 331,5	1,293	428,3
+ 5	334,5 334,5	1,269	424,5
+ 10	337,5 337,6	1,247	420,7
+ 15	340,5 340,6	1,225	417,0
+ 20	343,4 343,6	1,204	413,5
+ 25	346,3 346,7	1,184	410,0
+ 30	349,2 349,7	1,164	406,6

Elève 1 :
 Elève 2 :
 Elève 3 :

Etapes 3 : Répartition des tâches et organisation temporelle

Question 7 :



Activité 1 : élève 1

Pour le projet de création d'un appareil de mesure de distance, nous avons décidé d'utiliser un capteur à ultrason.



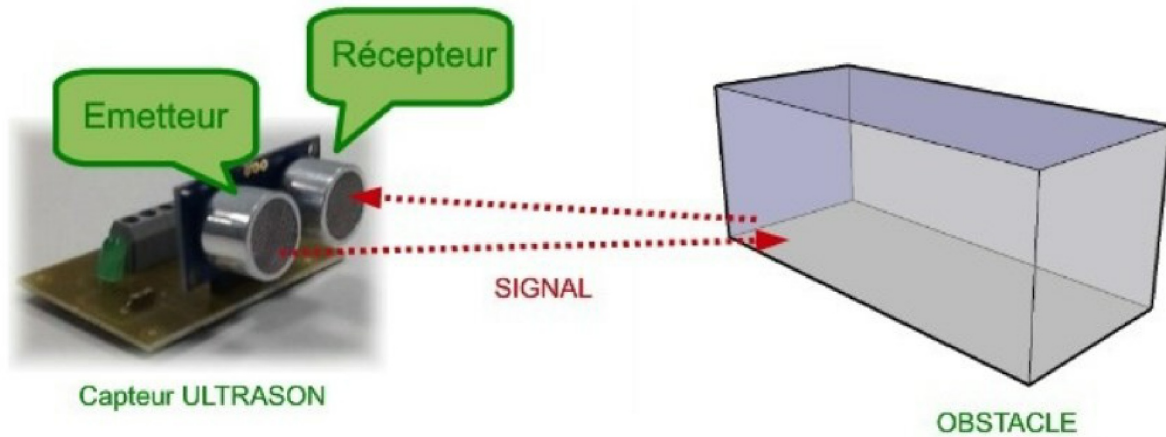
Elève 1 :

Elève 2 :

Elève 3 :

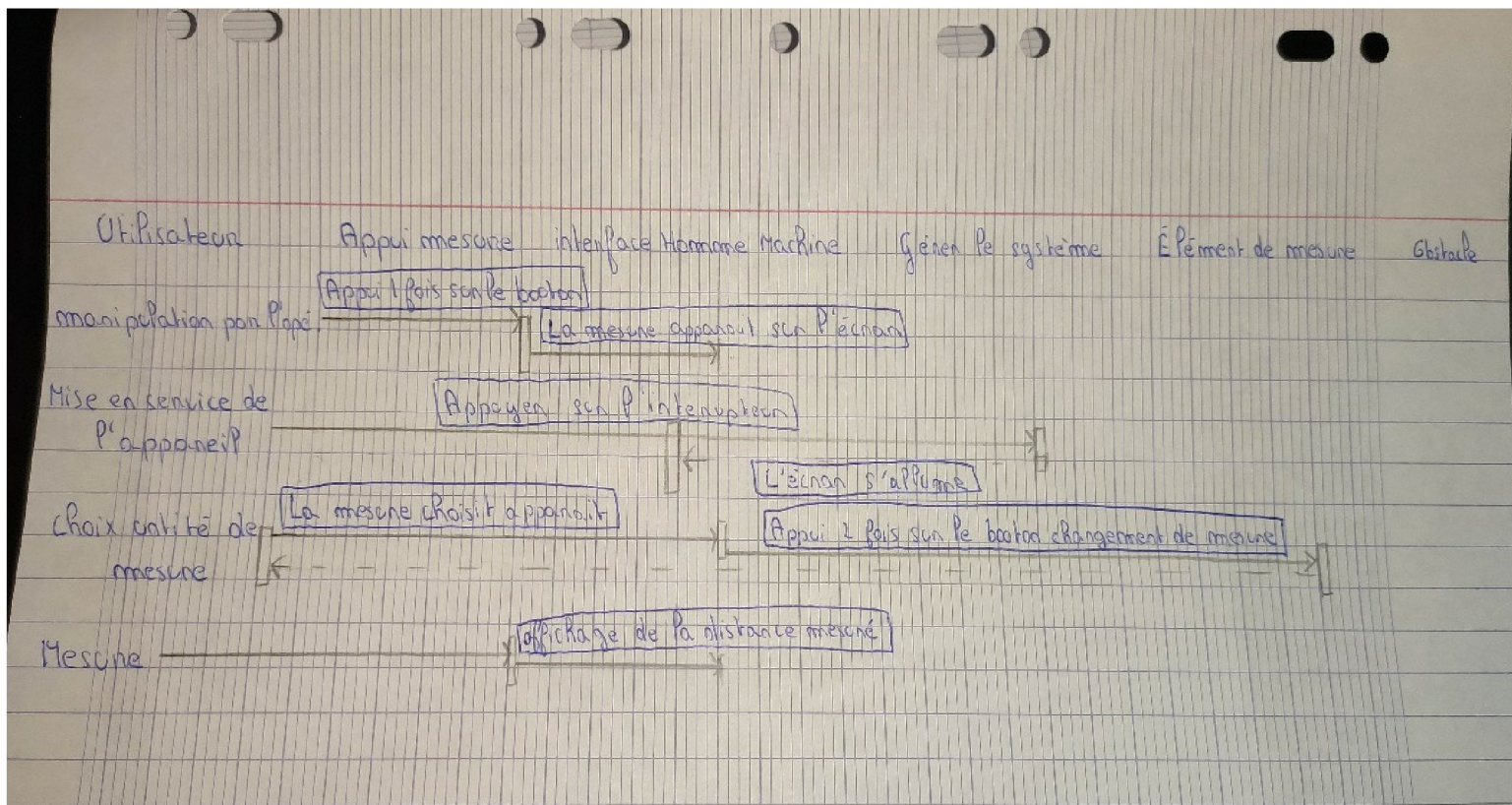
1G3

Il fonctionne sur un principe d'émission d'ultrasons par un émetteur et la réception des ondes par son récepteur, entre les 2 le son se répercutera sur un objet et grâce à un programme nous saurons la distance entre le capteur et l'objet.



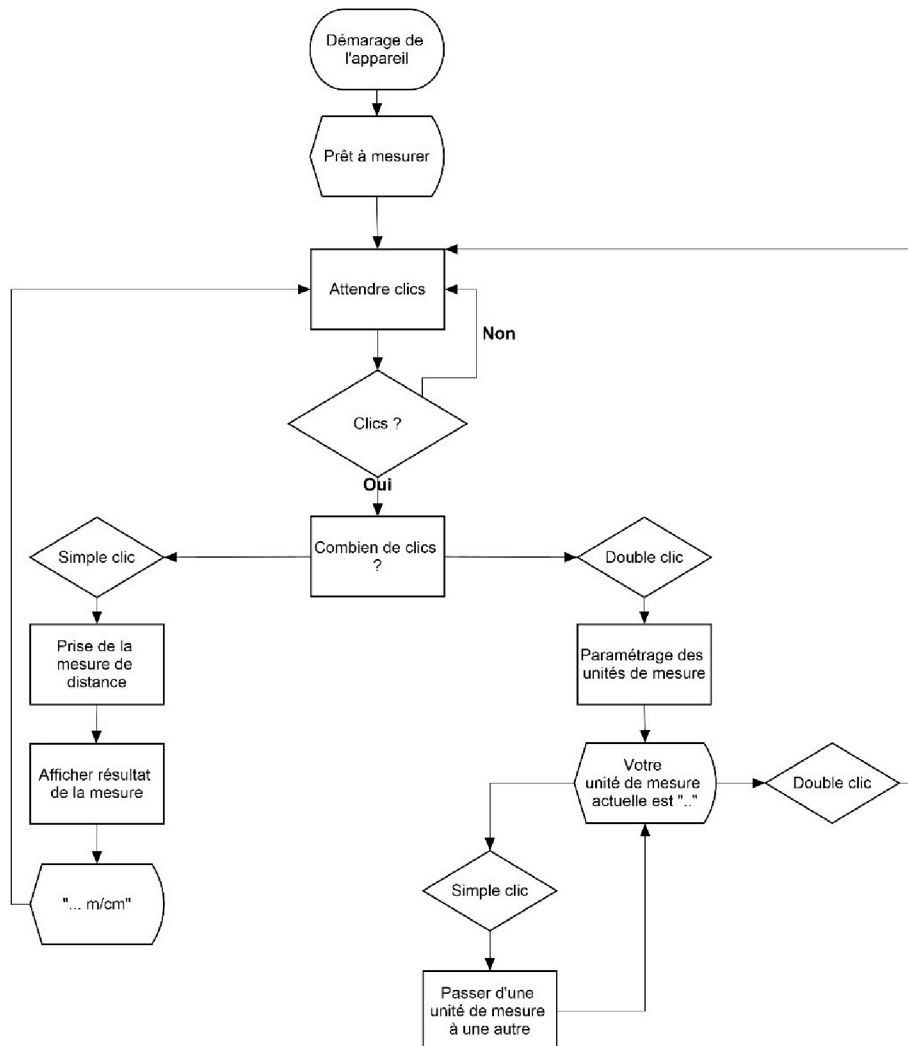
Activité 1 : élève 2

Nous avons réalisé l'algorithme de choix des unités et de la gestion du système (diagramme des tâches) pour définir la stratégie globale de fonctionnement :



Activité 1 : élève 3

Pour l'IHM, nous avons choisi un seul et unique bouton pour réaliser l'ensemble des tâches :



Activité 2 : élève 1

Ci-dessous est présenté le code du programme permettant d'effectuer la mesure de distance conformément à l'IHM établi:

```

#include <HCSR04.h>
#include <LiquidCrystal.h>

LiquidCrystal lcd(12, 11, 6, 5, 4, 3);
int pin_trig =10;
int pin_echo =9;
int pinBouton;
int capteurPin = 0;// Port analogique pour lire la température
int capteur;
float x;
long temp;
long lecture_echo;
long cm;
  
```

Elève 1 :
Elève 2 :
Elève 3 :

1G3

```
float vitesse;  
float temperature;  
  
void setup() {  
    //mise en place du nombre de colonne et de ligne du LCD  
    lcd.begin (16, 2);  
  
    //écriture du premier message  
    lcd.setCursor (0, 0);  
    lcd.print ("Bonjour");  
    delay (1000);  
    lcd.clear ();  
  
    //état des pin echo et trig du capteur  
    pinMode (pin_trig, OUTPUT);  
    digitalWrite (pin_trig, LOW);  
    pinMode (pin_echo, INPUT);  
  
    //configuration du bouton  
    pinBouton = 8;  
    pinMode (pinBouton, INPUT);  
  
    //message de confirmation  
    lcd.setCursor (0, 0);  
    lcd.print ("Pret");  
    delay (1000);  
    lcd.clear ();  
    //demande de mesure  
    lcd.setCursor (0, 0);  
    lcd.print ("Appuyez sur le");  
    lcd.setCursor (0, 1);  
    lcd.print ("bouton");  
}  
  
void loop() {  
  
    int lecture = analogRead(capteur); //lecture de la valeur numérique  
    float tension = lecture * 5.0;  
    tension = tension /1024.0; //conversion en tension  
    float temp = (tension-0.5) *100.0;  
  
    //lecture de l'état du bouton et stockage dans etatBouton  
    boolean etatBouton = digitalRead(pinBouton);  
  
    if (etatBouton==HIGH)//test si bouton appuyé  
    {  
        lcd.clear ();  
        //prise de mesure  
        lcd.setCursor (0, 0);  
        lcd.print ("Prise de mesure");  
        delay (2000);  
        lcd.clear ();  
        digitalWrite (pin_trig, HIGH);  
        delayMicroseconds(10);  
        digitalWrite(pin_trig, LOW);  
  
        lecture_echo = pulseIn(pin_echo,HIGH);  
        x = 1.40*287.053+(temp+273);  
    }
```

Elève 1 :
Elève 2 :
Elève 3 :

1G3

```
vitesse = sqrt(x)*0.001;
cm = lecture_echo*vitesse/2;

//affichage mesure
lcd.setCursor (4,0);
lcd.print ("DISTANCE:");
lcd.setCursor (5,1);
lcd.print (cm);
lcd.setCursor (9,1);
lcd.print ("cm");
delay(5000);
lcd.clear();
}

if (etatBouton==LOW)//test si bouton levé
{
    digitalWrite (pin_trig, LOW);
}

lcd.setCursor (0, 0);
lcd.print ("Pour nouvelle");
lcd.setCursor (0, 1);
lcd.print ("mesure");
lcd.clear ();
lcd.setCursor (0, 0);
lcd.print ("Appuyez sur le");
lcd.setCursor (0, 1);
lcd.print ("bouton");
}
```

Activité 2 : élève 2

Les composants nécessaires à ce projet sont : un écran LCD, un boîtier, un Arduino, des câbles, un bouton, une pile 9V, un Shield Grove, un capteur à ultrasons et un laser permettant de viser plus précisément.

Voici le programme permettant de choisir l'unité de distance conformément à l'IHM établi :

```
#include <OneButton.h>    //librairies !
#include "rgb_lcd.h"
int simpleclick;
OneButton button(8, true); //Onebutton au pin D8
rgb_lcd lcd;
int unit;                  //variables
int const cm;              //constantes
int const m;
int const mm;

void setup() {
    // put your setup code here, to run once:
    lcd.begin(16, 2);
    button.attachDoubleClick(doubleclick); //définition du double clic
    button.attachClick(simpleclick);      //définition du simple clic
    unit = cm;                            //unité par défaut le cm
}
```


Elève 1 :
Elève 2 :
Elève 3 :

1G3

```
}

void loop() {

    button.tick();                //surveillance du bouton
    delay(10);
}

void doubleclick() {
    lcd.setCursor(0, 0);
    lcd.print("Votre unité de mesure est");    //doubleclic
    delay(1000);
    lcd.print("unit");

    if (simpleclick){

        lcd.print("Votre unité de mesure est");
        delay(1000);                    //si il y a un seul clic
        unit = m;
        lcd.print("unit");

        if (simpleclick){

            lcd.print("Votre unité de mesure est");
            delay(1000);                //si il y a encore un seul
clic
            unit = mm;
            lcd.print("unit");

            if (simpleclick){
                lcd.print("Votre unité de mesure est");
                delay(1000);            //si il y a encore un seul
clic
                unit = cm;
                lcd.print("unit");
            }

        }

    }

    if (simpleclick){
        loop();                        //si il y a encore un seul clic, retour à la
        ligne de code 35
    }
    if (doubleclick){
        return;                        //si il y a un double clic, revenir à void
        loop()
    }
}

}
```


Activité 2 : élève 3

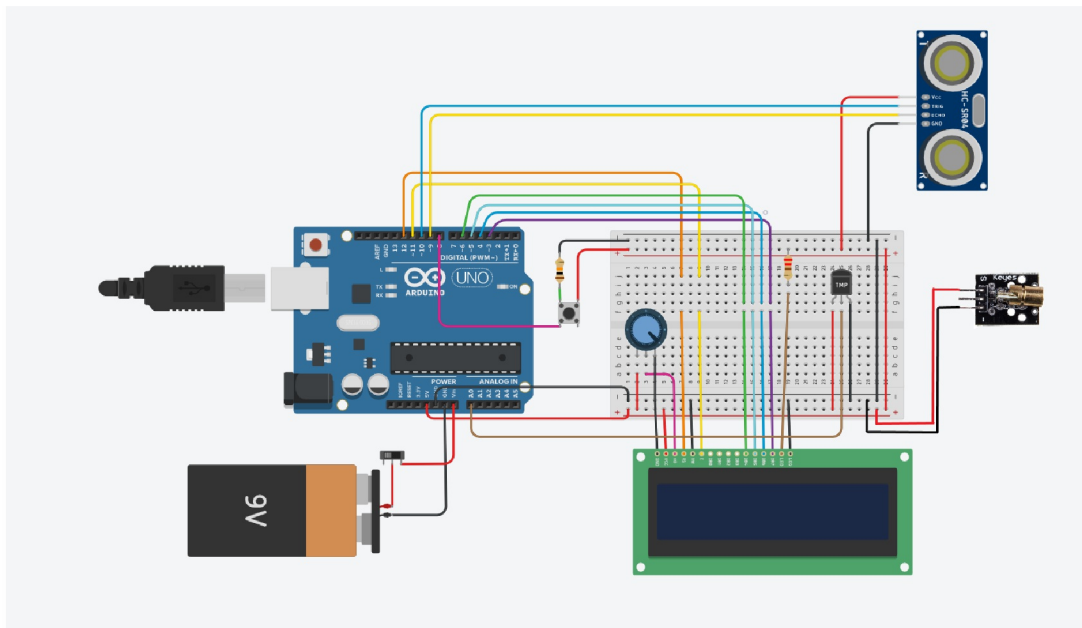
L'appareil de mesure de distance possède comme alimentation autonome une pile Duracell 9V avec pour capacité 565 mAh (c'est-à-dire qu'elle peut fournir 565 mAh pendant 1 heure): il suffit donc de faire le total de la consommation de l'appareil et de diviser la capacité de la pile par le total de la consommation de l'appareil :

Consommation système	mA
Arduino uno	38
Laser	40
Capteur ultrason	8
Ecran	60
Capteur T°	Négligeable
TOTAL	146

Capacité de la pile Duracell, 9V, alcaline):	565	mAh
Autonomie de le l'appareil (arrondi au dixième près):	232,2	Minutes
Soit en format 24H: 3H et 52 minutes environ.		

Activité 3 : câblage de l'appareil et schématisation de l'implantation des composants

Câblage des éléments: la carte Arduino sera câblée ainsi avec ses différences éléments :

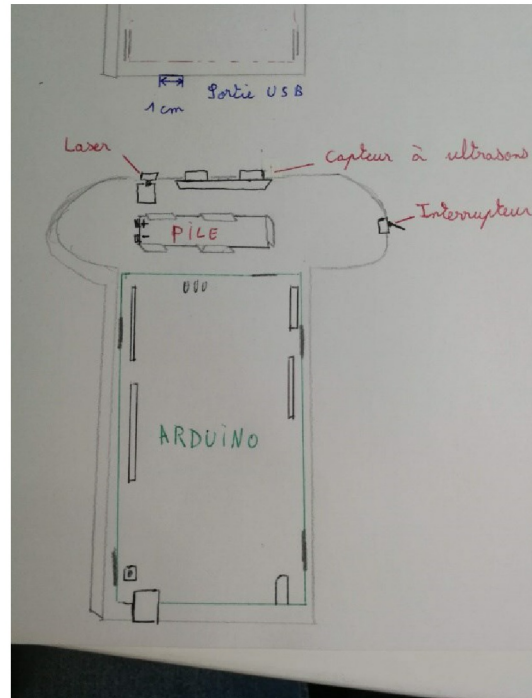
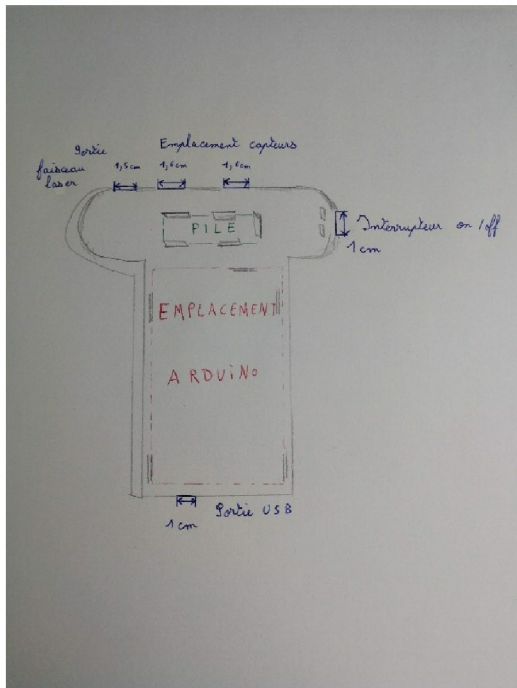


Elève 1 :
 Elève 2 :
 Elève 3 :

1G3

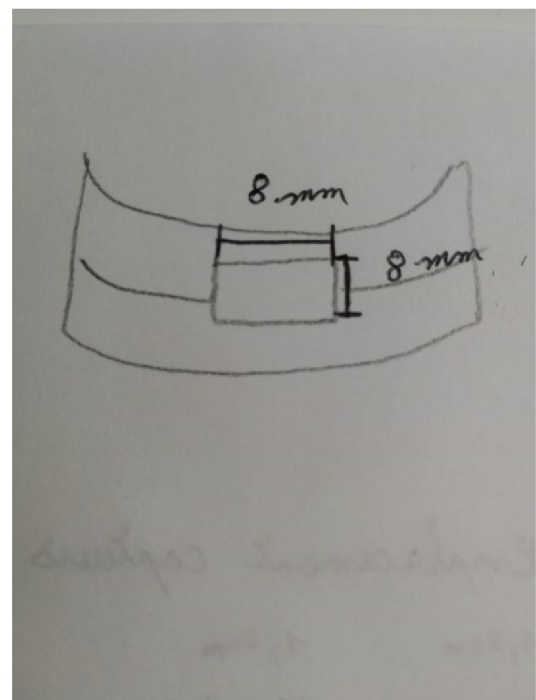
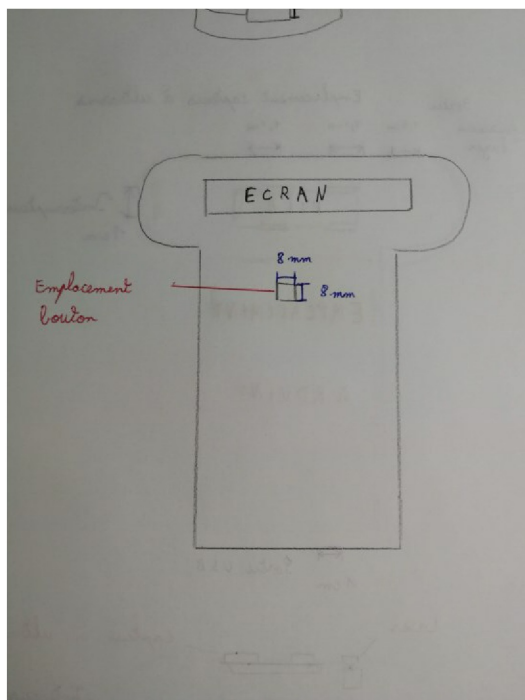
Note : L'écran sur le schéma n'étant pas le même que celui utilisé, certains câbles et composants (le potentiomètre par exemple) ne seront pas présents dans la réalité.

Schématisation des composants dans le boîtier :



Vue intérieure du boîtier sans les composants

Vue intérieure du boîtier avec les composants



Vue de la façade

Vue côté droit de l'intégration de l'interrupteur on/off