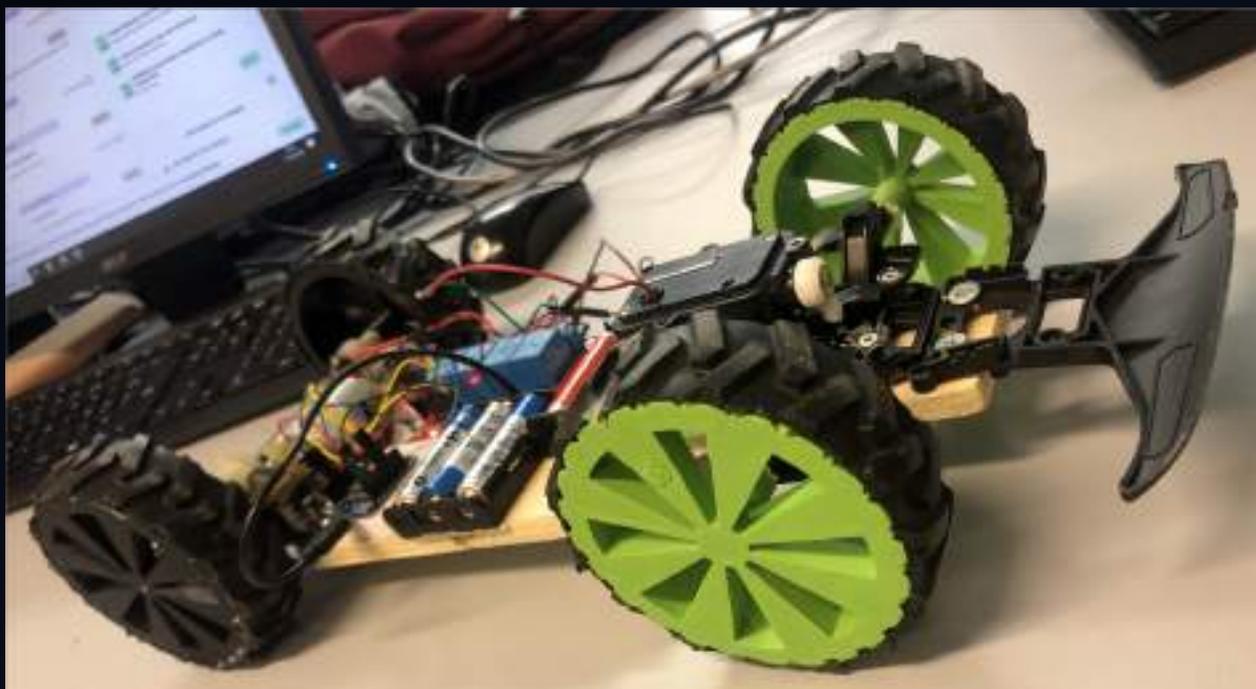


Voiture télécommandée



- Gabriel D.
- Bastien L.
- Noah B.
- Maxime F.
- Lucas L.

Sommaire :

- Notre projet

- Répartition des tâches

- Cahier des charges

- Les différentes pièces

- Calculs : vitesse et poids

- Modifications et problèmes rencontrés

- A-t-on réussi ?

Le projet :

Besoins initiaux :

Avoir une voiture que l'on peut contrôler depuis son téléphone portable est bien plus pratique, pas de télécommande superflue donc un coût de production moins important et une facilité d'utilisation supérieure.

Les problèmes techniques à résoudre :

- Mettre en place la propulsion de la voiture
- Diriger la voiture
- Un châssis permettant d'accueillir le système de direction, le moteur, les servomoteurs et les batteries d'alimentation
- Réaliser le programme qui contrôle la voiture à distance
- Une coque esthétique pour la voiture
- Aller à environ à 15 km/h

Répartition des tâches :

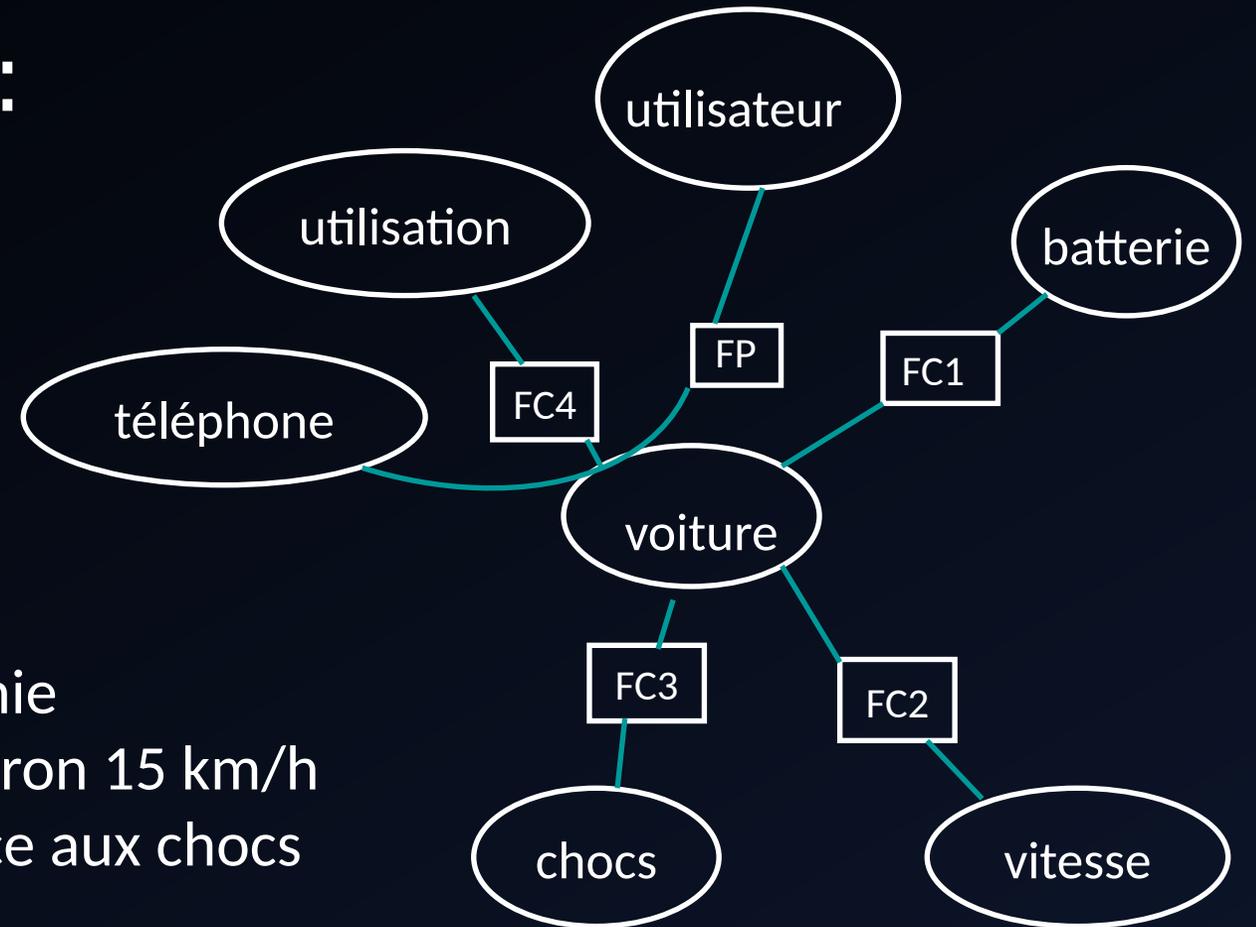
- Maxime : câblage, codage des commandes de la voiture
- Bastien : interface de la télécommande, modélisation et impression des pièces 3D, direction de la voiture (aide de Lucas)
- Gabriel et Lucas : usinage des pièces et montage de la voiture
- Noah : cahier des charges, documentation, calculs de la vitesse et du poids de la voiture

Cahier des charges :

Pieuvre :

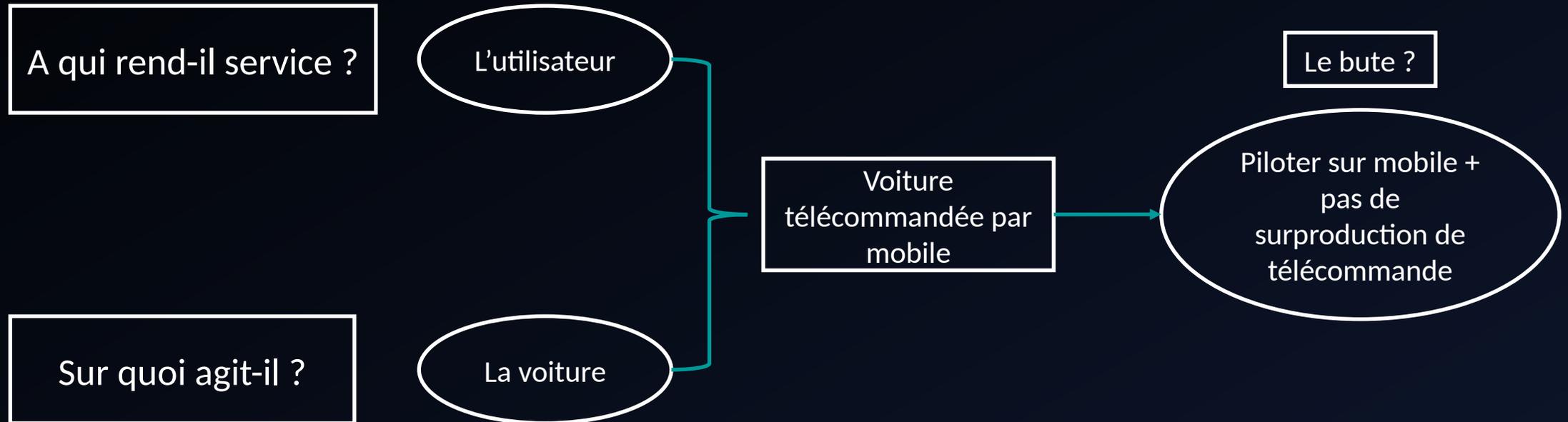
La voiture doit être capable de :

- FC1 : Avoir une certaine autonomie
- FC2 : Atteindre une vitesse d'environ 15 km/h
- FC3 : Avoir une certaine résistance aux chocs
- FC4 : Avoir une utilisation simple



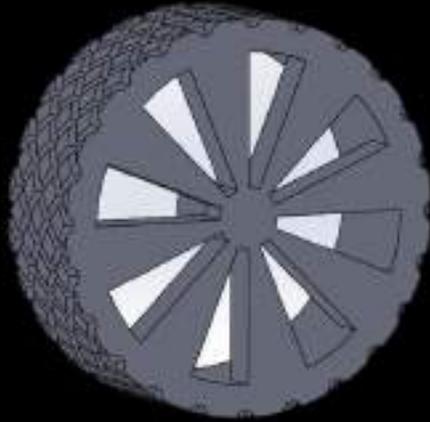
FP : Elle doit rendre son utilisation plus simple à l'utilisateur avec la télécommande directement sur le téléphone portable.

Bête à corne :

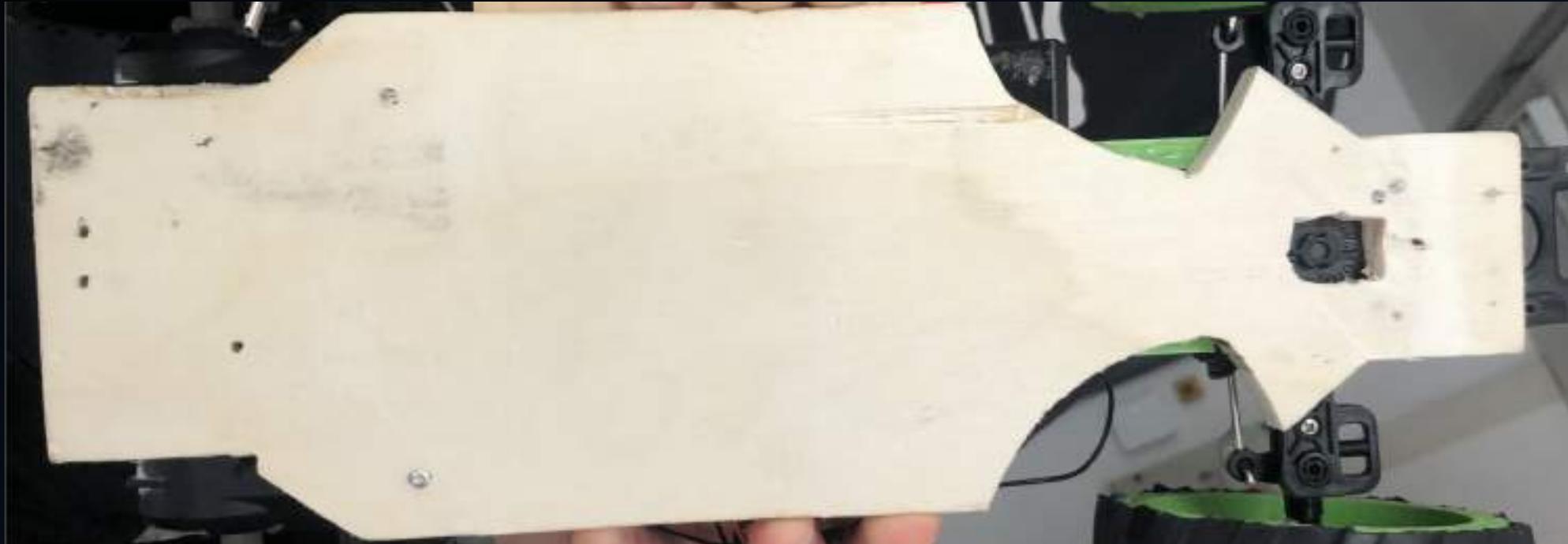


Les différentes pièces :

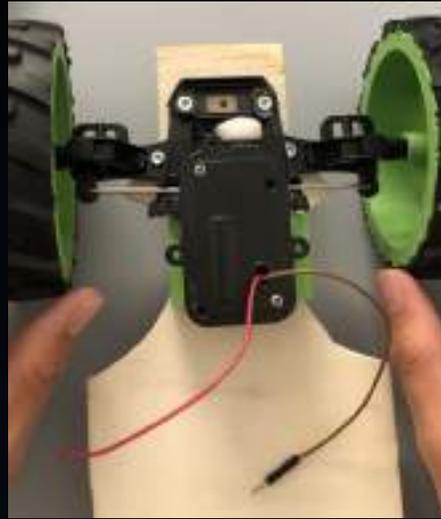
Solidworks :



Châssis :



Systeme
de direction :



Axe arriere :



Pneus :



Câblage :

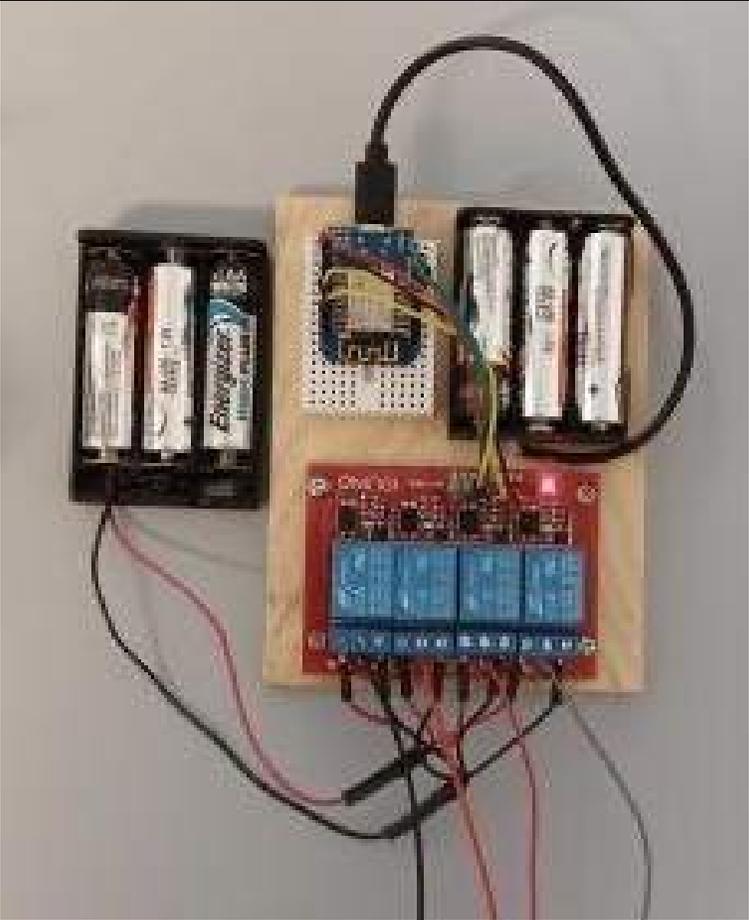
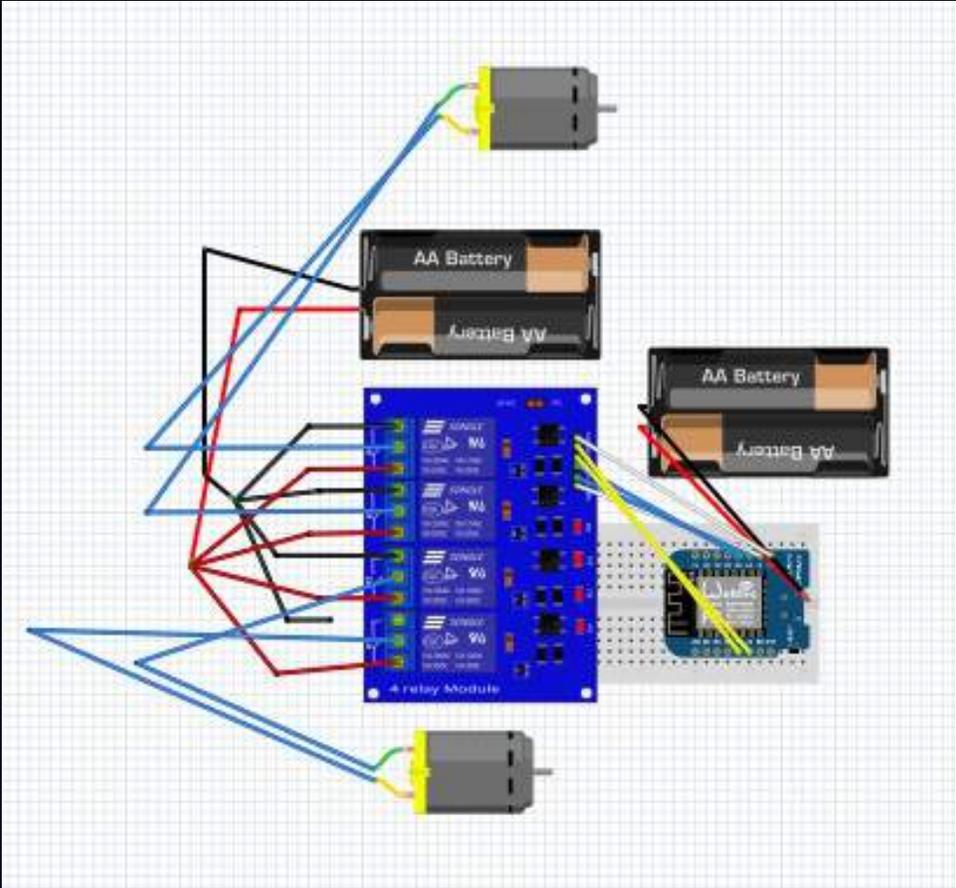


Schéma :



Calculs : vitesse et poids :

Vitesse :

$$=(B1*B3)/(B2*B4)$$

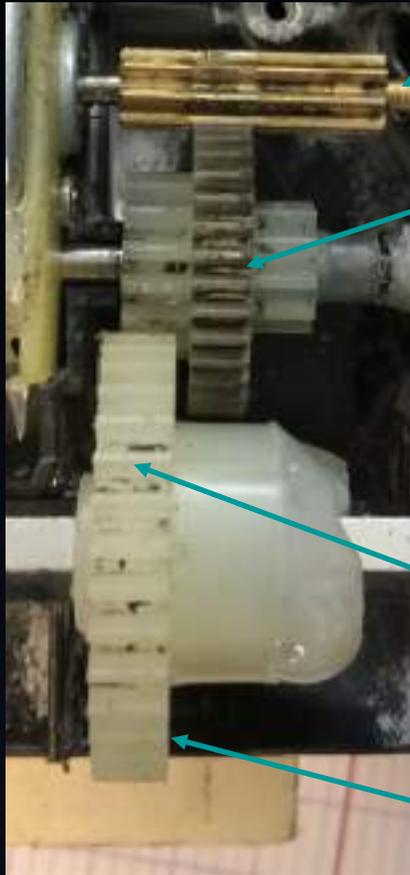
$$=3,1416*D1$$

	A	B	C	D	E	F
1	z1	8	diamètre roues (en m)	0,09	vitesse moteur (tr/min)	5750
2	z2	34	périmètre	0,282744		
3	z3	15				
4	z4	36				
5						
6	Rg	0,09803922				
7						
8	vitesse essieux (tr/min)	563,72549				
9	vitesse voiture (km/h)	9,5614877				

$$=F1*B6$$

$$=B8*D2/16,67$$

Schéma Rapport Global :

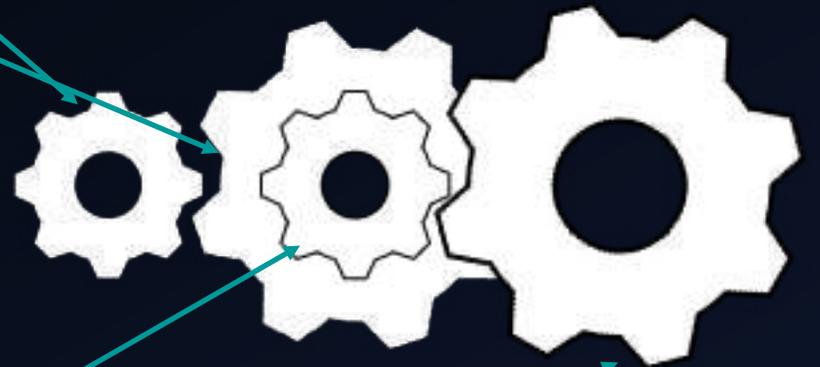


z_1

z_2

z_3

z_4



Masse totale :

	A	B	C	D	E	F	G	H
1	composants :	gentes	moteurs	axe arrière	batterie	châssis	carte wemos d1	moteur P
2	masse en grammes:	11	35	57	92	229	45	86
3								
4								
5								
6								
7	masse total (en g):	588						

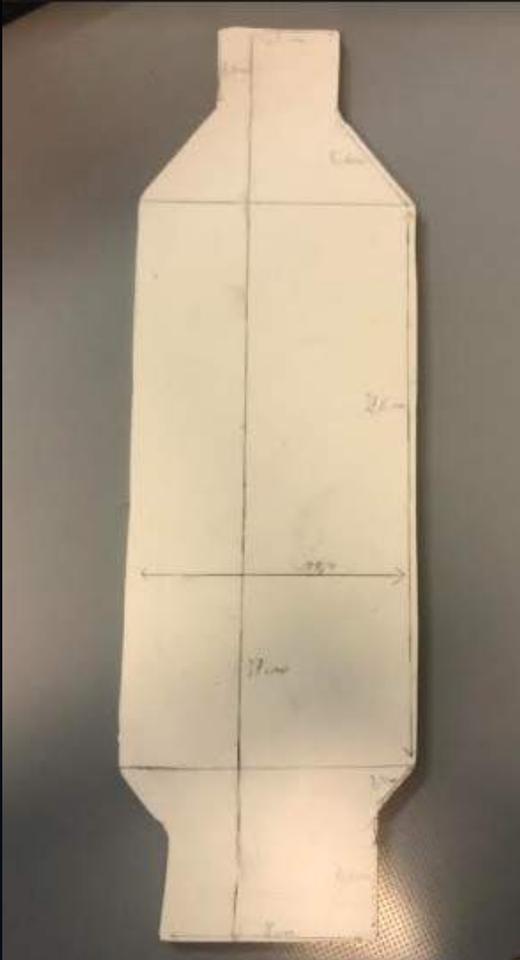

$$=B2*4+C2+D2+E2+F2+G2+H2$$

Modifications et problèmes rencontrés :

Problèmes rencontrés :

- Problèmes pour que les roues tournent à cause du châssis et de la direction
- Problèmes de la taille du trou des roues pour l'axe
- Problèmes de commandes de direction

Le châssis :



On a creusé sur les côtés pour que les roues puissent tourner



Taille du trou des roues pour l'axe :



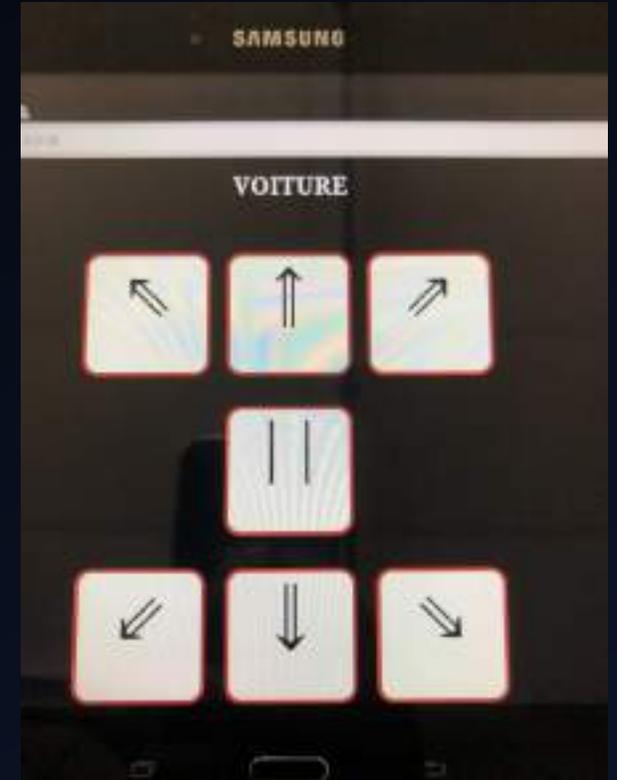
La taille de l'axe était trop petite, on l'a donc redimensionné sur solidworks avec des bagues d'axes de taille différentes.

Commandes de direction :

On a modifié le script ci-dessous :

```
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println("");
client.println("<!DOCTYPE HTML>");
client.println("<html>");
client.println("<head><meta charset=\"utf-8\"></head><style>body {text-align: center;background-color: black;}a
client.println("<body>");
client.println("<h1>VOITURE</h1>");
client.println("<br></br>");
client.println("<a href=\"/LED=ON\"> ⌂ </a><a href=\"/LED=V\"> ↑ </a><a href=\"/LED=OFF\"> ↗ </a>");
client.println("<br></br>");
client.println("<a href=\"/LED=B\">||</a>");
client.println("<br></br>");
client.println("<a href=\"/LED=C\"> ⌘ </a> <a href=\"/LED=X\"> ↓ </a> <a href=\"/LED=W\"> ⌘ </a>");
client.println("</body>");
client.println("</html>");
```

A-t-on réussi ?



La voiture n'atteint pas les 15 km/h mais 9 et son esthétique laisse à désirer, mais sinon tout fonctionne 👍

Voiture en marche :



Script (Annexe):

- #include <ESP8266WiFi.h>
- const char* ssid = "WEMOS_D1";
- const char* password = "987654321";
- IPAddress local_IP(193,168,0,222);
- IPAddress gateway(192,168,4,9);
- IPAddress subnet(255,255,255,0);
- WiFiServer server(80);
- int etat_led = 0;
-
- void setup() {
 - pinMode(SCL, OUTPUT);
 - pinMode(SDA, OUTPUT);
 - pinMode(MOSI, OUTPUT);
 - pinMode(MISO, OUTPUT);

- Serial.begin(115200);
- Serial.println();
- Serial.print("Setting soft-AP configuration ... ");
- Serial.println(WiFi.softAPConfig(local_IP, gateway, subnet) ? "Ready" : "Failed!");
- Serial.print("Setting soft-AP ... ");
- Serial.println(WiFi.softAP(ssid,password) ? "Ready" : "Failed!");
- Serial.print("Soft-AP IP address = ");
- Serial.println(WiFi.softAPIP());
-
- // Démarrer le serveur
- server.begin();
- Serial.println("Serveur démarré");
- digitalWrite(SCL, HIGH);
- }
-
- void loop() {

- // Vérifier si le client est connecté
- WiFiClient client = server.available();
- if (!client) {
- return;
- }
-
- // Attendre jusqu'à ce que le client envoie des données
- Serial.println("nouveau client");
- while(!client.available()){
- delay(100);
- }
-
- // Lire la première ligne de la requête
- String request = client.readStringUntil('\r');
- Serial.println(request);
- client.flush();

- // Vérifier la requête
- if (request.indexOf("LED=ON") != -1) {
- digitalWrite(SCL, HIGH);
- digitalWrite(SDA, LOW);
- digitalWrite(MISO,LOW);
- digitalWrite(MOSI,HIGH<<);
- etat_led = 1;
- }
-
- if (request.indexOf("LED=OFF") != -1) {
- digitalWrite(SDA, HIGH);
- digitalWrite(SCL, LOW);
- digitalWrite(MISO,LOW);
- digitalWrite(MOSI,HIGH);

- `etat_led = 0;`
- `}`
- `if (request.indexOf("LED=W") != -1) {`
- `digitalWrite(SCL, LOW);`
- `digitalWrite(SDA, HIGH);`
- `digitalWrite(MISO,HIGH);`
- `digitalWrite(MOSI,LOW);`
- `etat_led = 1;`
- `}`
- `if (request.indexOf("LED=X") !=-1) {`
- `digitalWrite(MISO,HIGH);`
- `digitalWrite(MOSI,LOW);`
- `digitalWrite(SCL,LOW);`
- `digitalWrite(SDA,LOW);`
- `etat_led = 1;`
- `}`

- `if (request.indexOf("LED=C") != -1) {`
- `digitalWrite(MISO,HIGH);`
- `digitalWrite(MOSI,LOW);`
- `digitalWrite(SCL,HIGH);`
- `digitalWrite(SDA,LOW);`
- `etat_led = 1;`
- `}`
- `if (request.indexOf("LED=V") != -1) {`
- `digitalWrite(MISO,LOW);`
- `digitalWrite(MOSI,HIGH);`
- `digitalWrite(SCL,LOW);`
- `digitalWrite(SDA,LOW);`
- `etat_led = 1;`
- `}`
- `if (request.indexOf("LED=B") != -1) {`

- `digitalWrite(SCL,LOW);`
- `digitalWrite(SDA,LOW);`
- `digitalWrite(MISO,LOW);`
- `digitalWrite(MOSI,LOW);`
- `etat_led = 1;`
- `}`
-
- `// Retourner la réponse`
- `client.println("HTTP/1.1 200 OK");`
- `client.println("Content-Type: text/html");`
- `client.println("");`
- `client.println("<!DOCTYPE HTML>");`
- `client.println("<html>");`

- `client.println("<head><meta charset=\"utf-8\"></head><style>body {text-align: center;background-color: black;}a {width: 150px;height: 150px;text-decoration: none;background-color: white;color: black;margin: 10px;border: 5px solid brown;border-radius: 20px;display: inline-block;font-size: 80px}h1{color: white;text-align: center;}b {width: 500px;height: 150px;text-decoration: none;background-color: white;color: black;margin: 10px;border: 5px solid brown;border-radius: 20px;display: inline-block;font-size: 80px}h1{color: white;text-align: center;}</style> ");`
- `client.println("<body>");`
- `client.println("<h1>VOITURE</h1>");`
- `client.println("
</br>");`
- `client.println(" ⚡ ⬆ ⬇ ");`
- `client.println("
</br>");`
- `client.println(" | |");`
- `client.println("
</br>");`
- `client.println(" ⚡ ⬇ ⚡ ");`
- `client.println("</body>");`
- `client.println("</html>");`
- `delay(10);`
- `Serial.println("Client disconnected");`
- `Serial.println("");`