

**CURSO SUPERIOR DE TECNOLOGIA EM
MANUFATURA AVANÇADA**

**PROJETO EXECUTIVO DE DESENVOLVIMENTO
INTEGRADO DE PRODUTO 0004 – AITEC (CONTROLE)**

JOSÉ EDUARDO TEIXEIRA DOS REIS

HELOÍSA CLARA RODRIGUES

**São José dos Campos
2025**

ÍNDICE DE FIGURAS

Figura 1 - Relação de materiais.	5
Figura 2 - Representação 3D (case).	6
Figura 3 - Representação 3D (tampa da case).	6
Figura 4 - Desenho técnico - Tampa da case.	7
Figura 5 - Desenho técnico - Case.	8
Figura 6 - Tampa da case Ultimaker.	9
Figura 7 - Case no Ultimaker.	10
Figura 8 - Case montada.	11
Figura 9 - Teste inicial dos componentes.	12
Figura 10 - Diagrama elétrico do controle.	13
Figura 11 - Circuito finalizado.	14

SUMÁRIO

1. INTRODUÇÃO	4
2. PROJETO EXECUTIVO	4
2.1 Materiais necessários	4
3. MONTAGEM	5
3.1 Montagem (case)	5
3.1.1 Impressão das peças	9
3.2 Montagem eletrônica.....	11
3.3 Programação.....	14

1. INTRODUÇÃO

O objetivo deste projeto é o desenvolvimento de uma arena para competição de robôs. A arena de robôs será projetada para competições de robôs remoto controladas, incluindo um placar para marcação de pontos, bem como um cronômetro para controle das partidas. Os robôs serão diferenciados por cor para identificação e divisão de times. Haverá um controle remoto para cada robô. O placar será controlado por botões inseridos na lateral da mesa. As regras serão estabelecidas para uma melhor organização das partidas e controladas por um juiz que será da nossa equipe.

2. PROJETO EXECUTIVO

Esta seção aborda as etapas de montagem do controle por gestos, contendo o material necessário para o desenvolvimento do produto como um todo, desde a eletrônica a parte mecânica. Ele é composto por materiais adquiridos, como os componentes eletrônicos, e materiais desenvolvidos pela equipe, como a case para a proteção eletrônica e ergonomia do operador, feita por impressão 3D.

2.1 Materiais necessários

Esta subseção aborda não os materiais que serão utilizados para a montagem de cada etapa do controle, a figura (1) apresenta a relação completa dos materiais que são necessários para o funcionamento do produto e seus custos. O custo total para desenvolvimento do controle fica em torno de R\$ 329,78 reais.

Figura 1 - Relação de materiais.

Item	Descrição	Quantidade	Unidade	Valor total
1	Arduino Nano Compatível V3 Atmega328 Ch340 Pino Sold	1	PÇ	R\$31,91
2	Módulo Bluetooth Hc-05 - Rs232/bluetooth - Master / Slave	1	PÇ	R\$33,22
3	Kit 2 Bateria Recarregável 18650 3.7v 4.2v Li-ion	2	PÇ	R\$60,00
4	1 pçs 5.2x8.9cm padrão permanente placa de ensaio solda placa pcb protótipo placa protoboard diy protótipo eletrônica	1	PÇ	R\$13,90
5	Barra De Pinos Macho - 1x40 - 2,54mm Cor Preto	1	PÇ	R\$09,90
6	Kit Cabo Jumper Mxm + Mxf + Fxf 120 Unidades 20cm Arduino	1	PÇ	R\$26,00
7	Fita Auto Adesiva com material Macho + Fêmea 25mm Fixação De Pedais - 1 Metro PRETO	1	MT	R\$28,50
8	Suporte para 2 Baterias 18650	1	PÇ	R\$05,90
9	Peças geradas por impressão 3D	2	PÇ	R\$00,00
10	Kit com 40 LEDs Diversos 5mm Difusos - 4 CORES (Amarelo, Azul, Verde e Vermelho)	1	PÇ	R\$16,53
11	Acelerômetro E Giroscópio 3 Eixos Mpu-6050 Gy-521	1	PÇ	R\$14,90

Fonte: Próprios autores (2025).

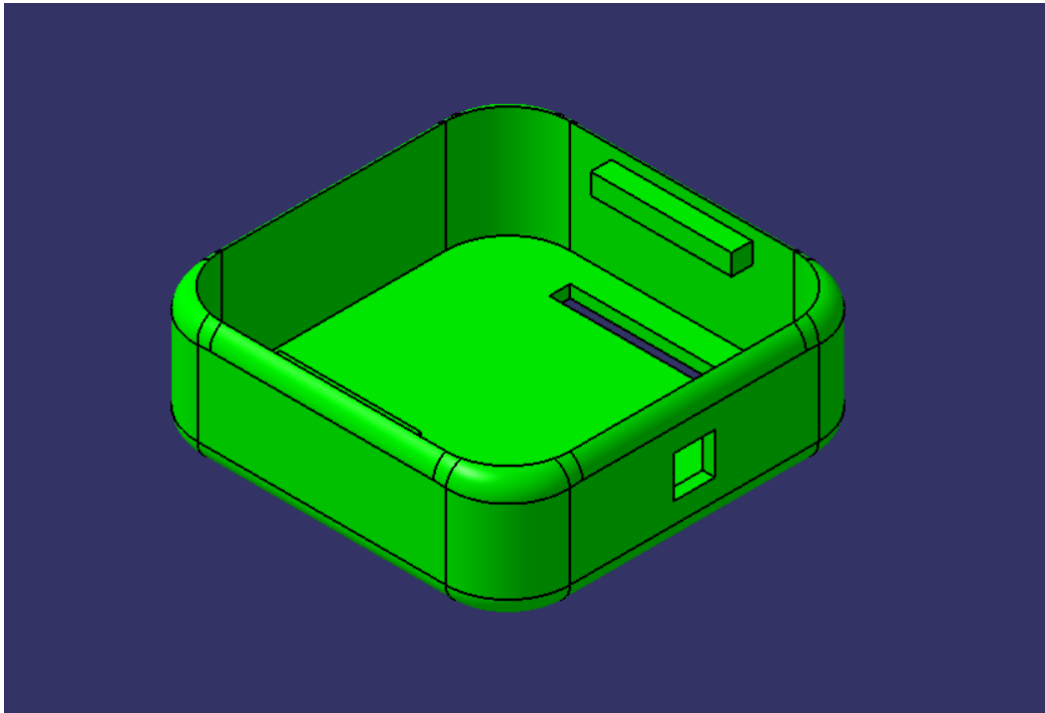
3. MONTAGEM

Esta seção aborda a montagem por completo do controle, da parte eletrônica e da parte mecânica do produto.

3.1 Montagem (case)

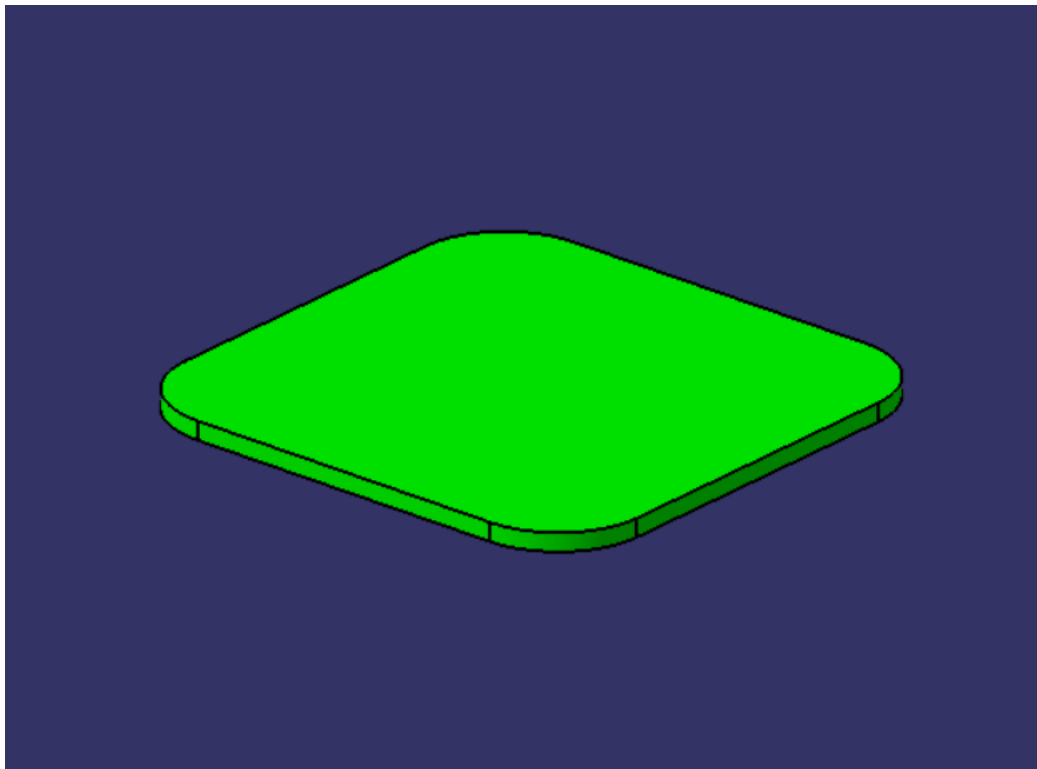
Esta subseção apresenta as etapas de montagem da case, desde o desenvolvimento do desenho técnico (CAD) da solução, a impressão das peças e o resultado final da solução. A imagens a seguir, de dois (2) a cinco (5), representam as peças desenvolvidas no software de desenho assistido por computador CATIA, no caso, a case para a parte eletrônica e sua tampa.

Figura 2 - Representação 3D (case).



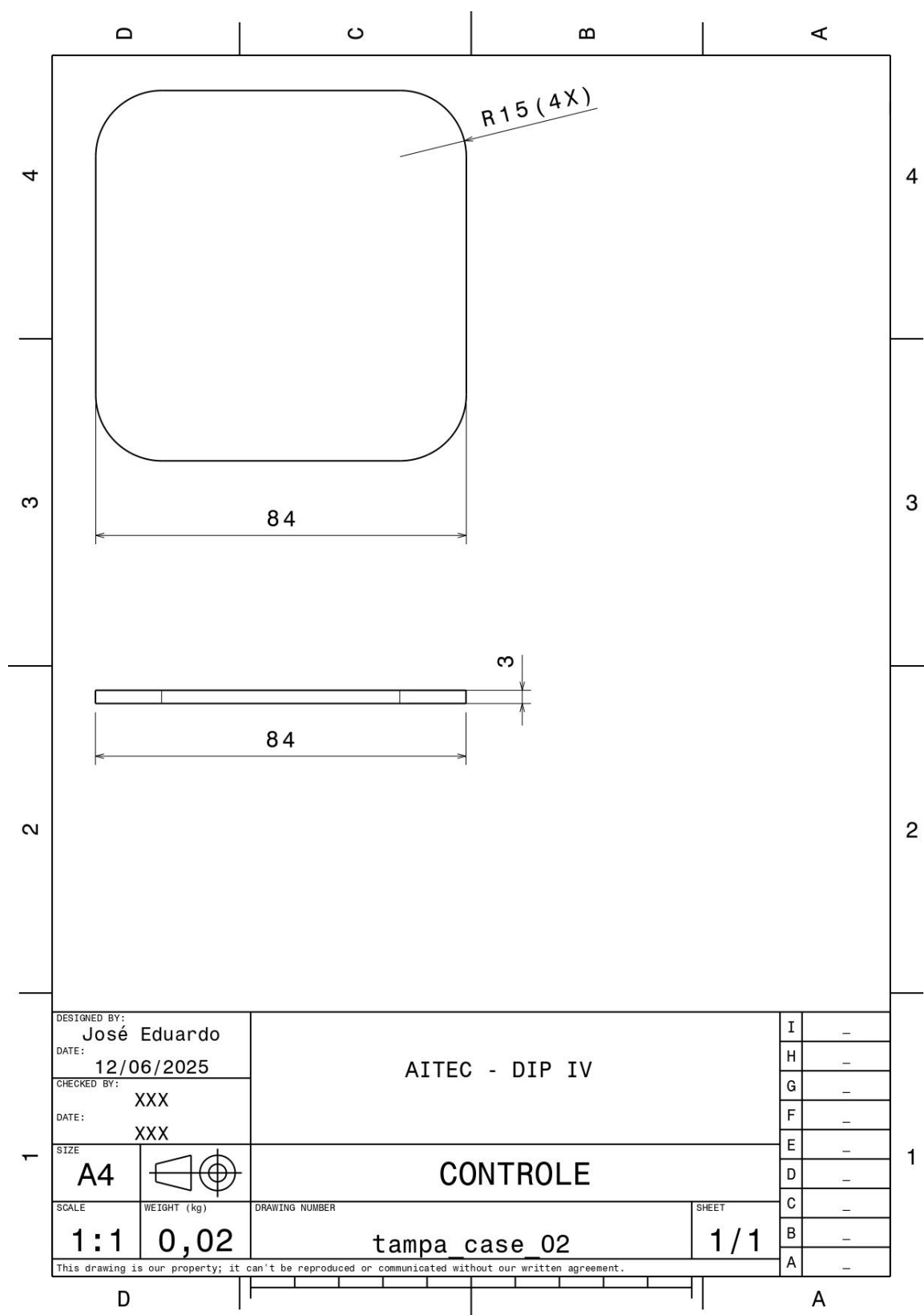
Fonte: Próprios autores (2025).

Figura 3 - Representação 3D (tampa da case).



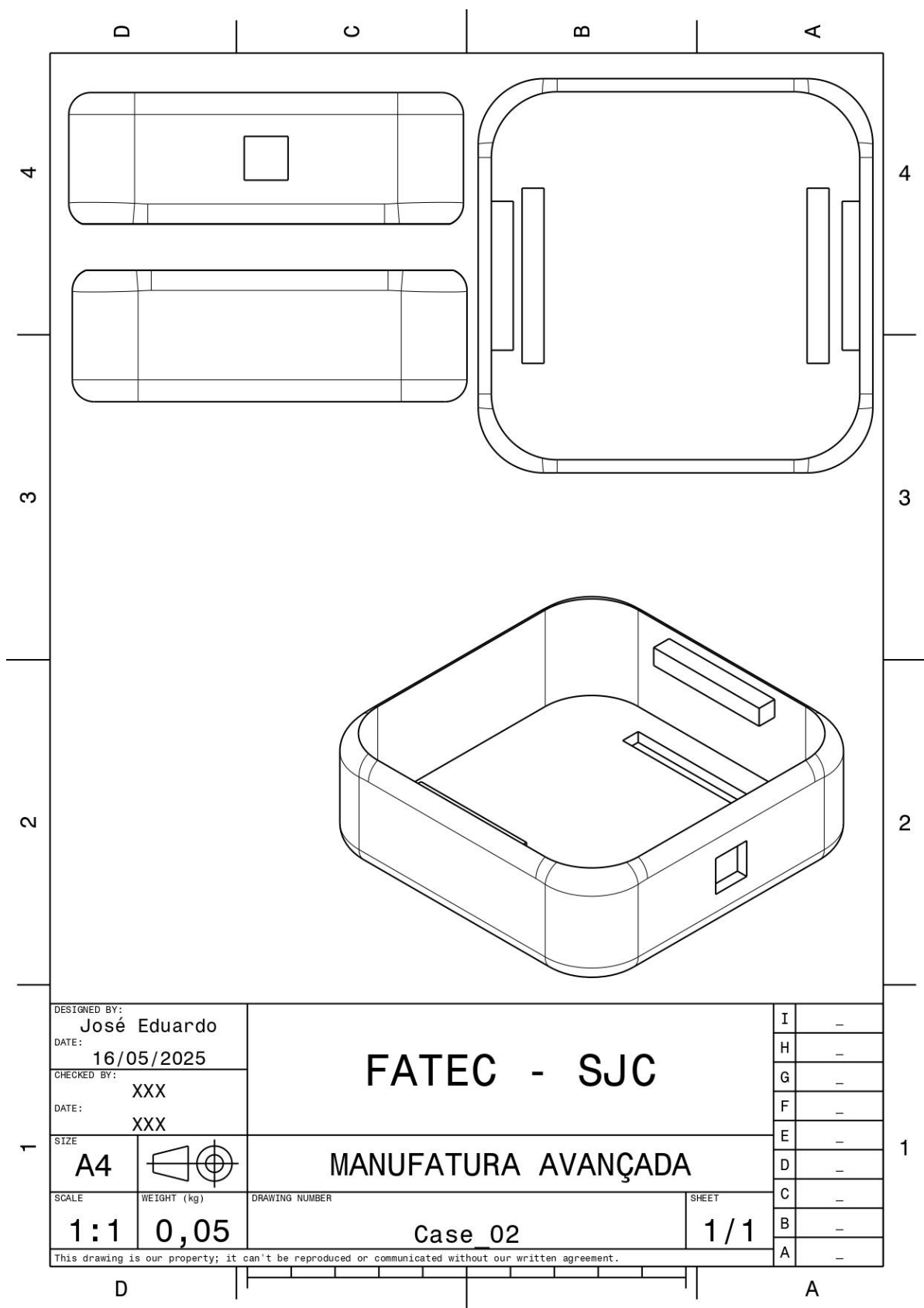
Fonte: Próprios autores (2025).

Figura 4 - Desenho técnico - Tapa da case.



Fonte: Próprios autores (2025).

Figura 5 - Desenho técnico - Case.



Fonte: Próprios autores (2025).

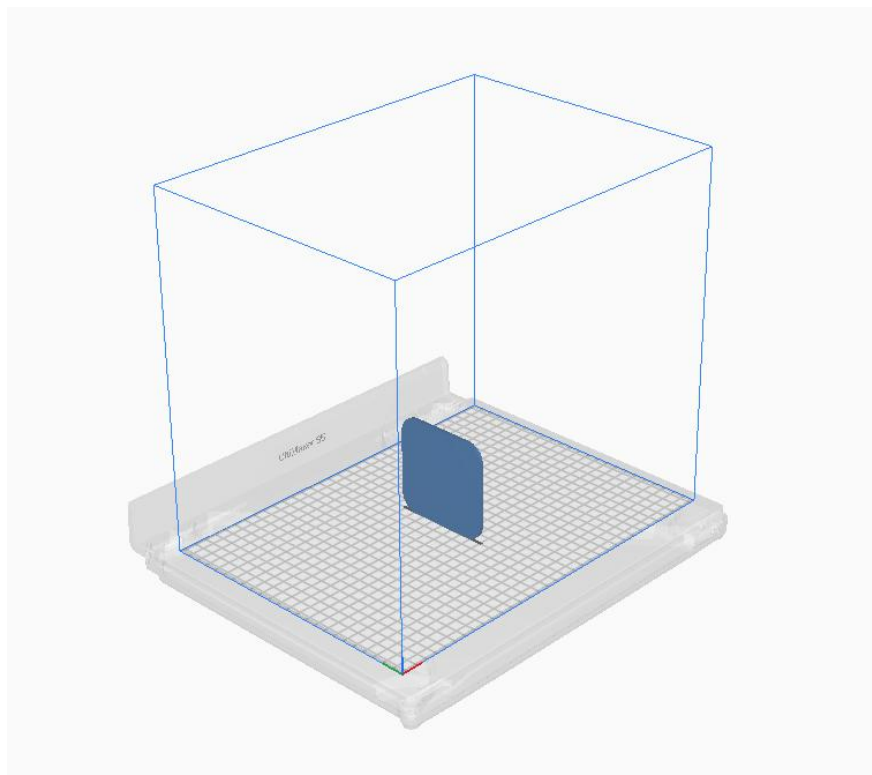
3.1.1 Impressão das peças

Esta subseção apresenta as peças que devem ser desenvolvidas por impressão 3D, com desenhos técnicos e as peças no formato STL. Na hora da impressão deve-se levar em consideração os seguintes itens:

- Modelo da impressora;
- O tipo da impressora e o material (preferência ao PLA);
- Não é necessário o uso de suportes ou moldes para impressão;
- Resolução 0.28mm;
- Densidade do filamento 15%.

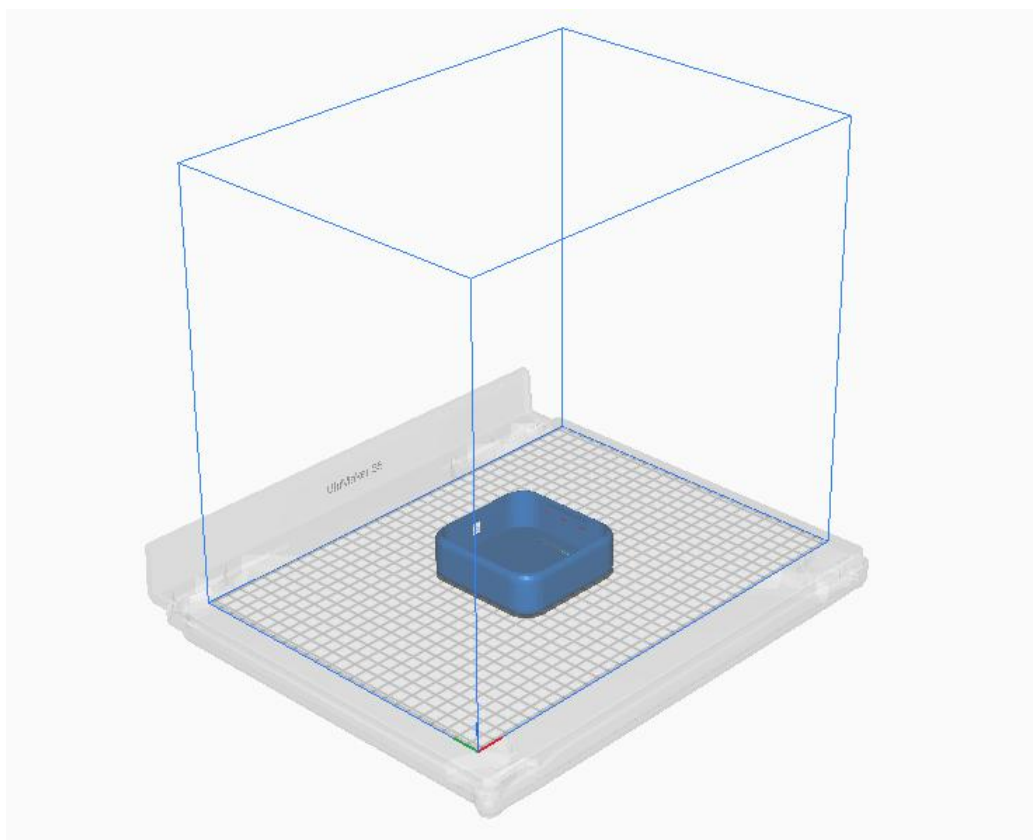
O software de fatiamento utilizado para o desenvolvimento das peças foi o **UltiMaker**, que é disponível para uso gratuitamente. Após a impressão, verifique a parte dos furos e realize uma limpeza simples na peça. As imagens de seis (6) e sete (7) apresentam as peças no software e impressas.

Figura 6 - Tampa da case Ultimaker.



Fonte: Próprios autores (2025).

Figura 7 - Case no Ultimaker.



Fonte: Próprios autores (2025).

Para a montagem, basta encaixar a tampa na case, que será presa por uma fita de velcro nas hastes da case, e na superfície inferior da tampa, assim a parte eletrônica estará protegida e o controle pronto para uso. A imagem oito (8) representa a case montada e pronta para uso.

Figura 8 - Case montada.



Fonte: autoria própria (2025).

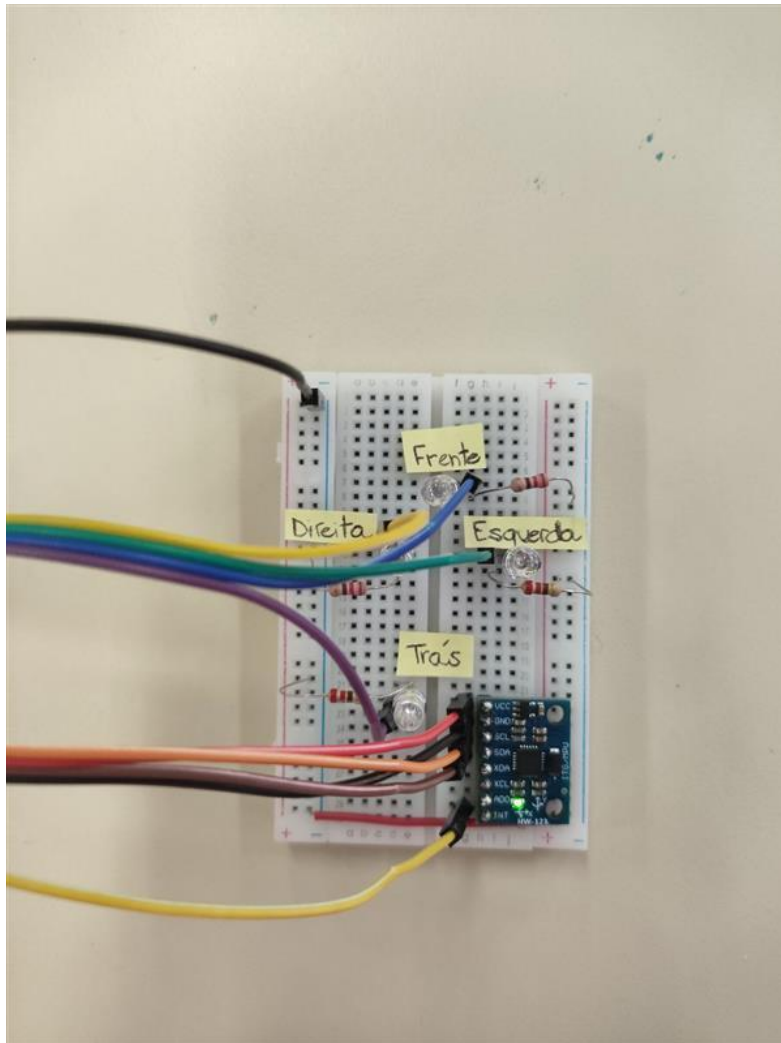
3.2 Montagem eletrônica

Esta subseção apresenta a montagem eletrônica do controle, com a relação de itens necessários e as etapas de montagem. Segue a lista de materiais necessários para a montagem:

- Arduino nano (1x);
- Acelerômetro e Giroscópio MPU-6050 (1x);
- Jumpers (15X);
- Bateria 3,7V 18500 (1x);
- Módulo Bluetooth Hc-05 (1x);
- Barra de pinos macho (1x);
- Placa de ensaio solda placa pcb (1x);
- Módulo step up mt3608 (1x).

Antes de realizar a montagem, é necessário realizar testes de funcionamento de cada componente e um teste de integração e funcionamento da programação, A imagem nove (9) apresenta o teste realizado para o desenvolvimento da solução.

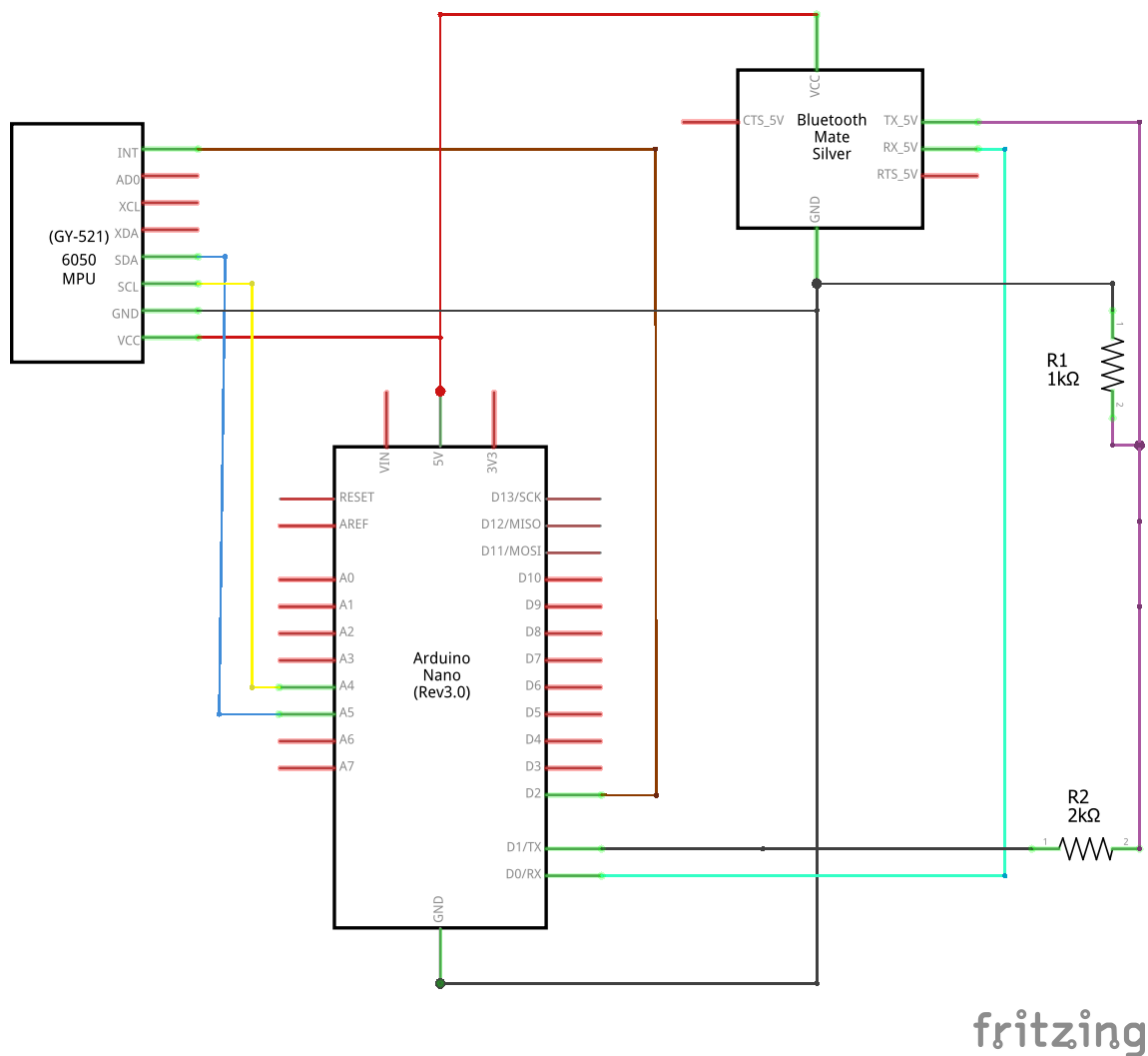
Figura 9 - Teste inicial dos componentes.



Fonte: Próprios autores (2025).

Após realizar o teste inicial, e verificar que todos os componentes estão funcionando corretamente, a montagem do circuito final pode ser realizada de acordo com o diagrama elétrico a seguir, representado pela imagem dez (10).

Figura 10 - Diagrama elétrico do controle.



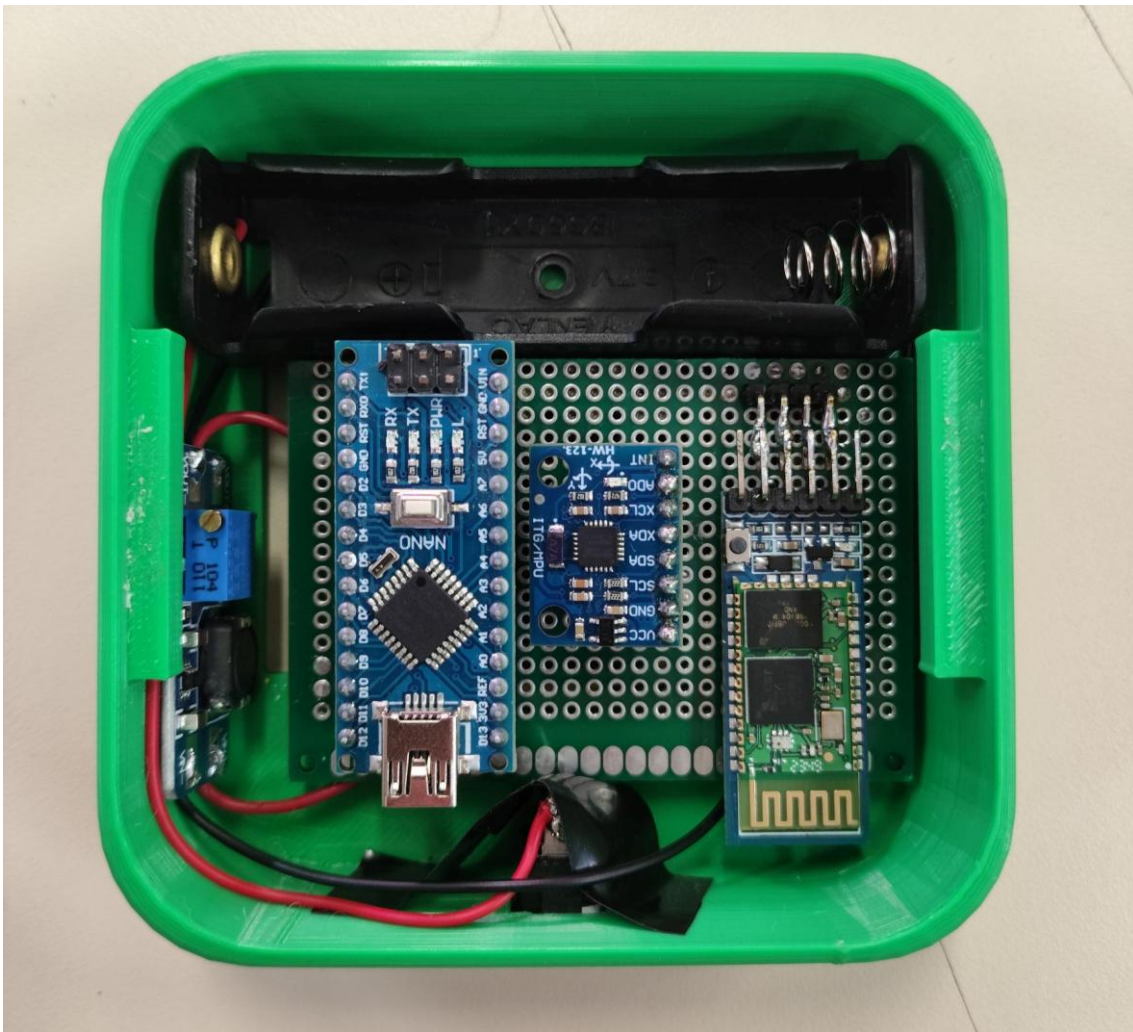
Fonte: Próprios autores (2025).

Para realizar a montagem, é necessário seguir estas etapas:

- Separar todos os componentes e materiais necessários;
- Alinhar a pinagem de comunicação e alimentação para inicial a solda;
- Alinhar os componentes na placa PCB;
- Soldar todos os componentes de acordo com o diagrama.

A solda serve para evitar o mal contato entre os componentes e garantir um funcionamento efetivo do circuito, a imagem onze (11) apresenta o circuito finalizado.

Figura 11 - Circuito finalizado.



Fonte: Próprios autores (2025).

3.3 Programação

Esta subseção apresenta o desenvolvimento da programação do controle, ela foi feita na linguagem C++ na plataforma arduino IDE. Segue a programação do controle por completo a seguir.

```
#include <Wire.h>
```

```
#include <SoftwareSerial.h>
```

```
#include <MPU6050.h>
```

```
MPU6050 mpu;
```

```
SoftwareSerial bt(10, 11); // TX = 10, RX = 11 para o HC-05/HC-05
```

```

int16_t ax, ay, az;

void setup() {

  Wire.begin();      // Inicia comunicação I2C com o MPU6050

  mpu.initialize();  // Inicializa o sensor MPU6050

  bt.begin(9600);     // Inicia comunicação com módulo Bluetooth

  delay(1000);       // Aguarda 1 segundo para estabilizar tudo
}

void loop() {

  mpu.getAcceleration(&ax, &ay, &az); // Lê valores brutos de aceleração dos
eixos X, Y e Z

  // Mapeia a inclinação no eixo Y para um valor de velocidade (entre 80 e 200)

  int velocidade = map(abs(ay), 2000, 16000, 80, 200);

  velocidade = constrain(velocidade, 0, 200); // Garante que não ultrapasse os
limites

  // Movimento para frente

  if (ay > 2500) {

    bt.print('F');

    bt.write(velocidade);

  }

  // Movimento para trás

  else if (ay < -2500) {

    bt.print('B');

    bt.write(velocidade);

  }

  // Curva rápida para a direita (mais sensível)

```

```
else if (ax > 4200) {  
    bt.print('E');  
}  
  
// Curva rápida para a esquerda (mais sensível)  
else if (ax < -4200) {  
    bt.print('Q');  
}  
  
// Parado  
else {  
    bt.print('S');  
}  
  
delay(150); // Delay para estabilidade  
}
```