

ESCOLA TÉCNICA ESTADUAL - ETEC

PAULINO BOTELHO

Eletromecânica

**Cláudio Rafael Siqueira, Leonardo Luiz Zeviani, Lucas Durval e
Renato de Melo Figueiredo**

BRAÇO ROBÓTICO CONTROLADO POR ARDUÍNO

São Carlos - SP

2025

**Cláudio Rafael Siqueira, Leonardo Luiz Zeviani, Lucas Durval e
Renato de Melo Figueiredo**

BRAÇO ROBÓTICO CONTROLADO POR ARDUÍNO

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Eletromecânica da Etec Paulino Botelho, orientado pelos Prof. Luiz de Paula Júnior e Luiz Schiavone Neto, como requisito parcial para obtenção do título de técnico em Eletromecânica.

São Carlos - SP

2025

Agradecemos especialmente,

À Deus, que nos guiou em todos os momentos, mostrando os melhores caminhos e sempre nos abençoando. Aos nossos queridos e amados pais, muito obrigado pelo amor incondicional, paciência, educação, carinho, amizade e por tudo que proporcionaram para o nosso aprendizado.

À Escola Técnica Estadual “Paulino Botelho”, na presença de sua Diretora, Professora Dirlei Martins Franco. Ao Coordenador do curso de Eletromecânica, Prof. Dr. Luiz Vicente Vareda, pela dedicação, competência e empenho em todas as atividades. Aos nossos orientadores, Professor Luiz de Paula Junior e Prof. Luiz Schiavone Neto. Somos infinitamente gratos a todos os demais professores que participaram do nosso aprendizado. Sendo eles: Prof. Antônio Maurilo Barreiro Villas Boas, Prof. Célio Escobar, Prof. Eliezer Gilbertoni, Profa. Fernanda Remes Mattiuz, Prof. Frederico Jurgensen Junior, Prof. Guillermo Lourivalso Espíndola de Almeida, Prof. José Wagner Tavano Cavazin, Prof. Valter Cesár Govoni, Prof. Yan Masetto Nicolai e Claudio Torres Gonçalves.

“ A matemática é o alfabeto com o qual Deus escreveu o Universo”

*(Galileu Galilei)**

*"Eu asseguro que, se vocês tiverem fé e não duvidarem, poderão fazer não somente o que foi feito à figueira, mas também dizer a este monte: 'Levante-se e atire-se no mar', e assim será feito. E tudo o que pedirem em oração, se crerem, vocês receberão". (Mateus 21:21-22)***

* Galilei, Galileu. *Il Saggiatore* (O Ensaaiador). Roma: Accademia dei Lincei, 1623.

** Bíblia Sagrada. Barueri: Sociedade Bíblica do Brasil, 2018.

RESUMO

Este projeto tem como objetivo integrar e aplicar conhecimentos adquiridos no curso Técnico em Eletromecânica aplicado a um protótipo de braço robótico controlado por Arduino, utilizando MDF cortado a laser para confecção da estrutura. A escolha do Arduino como plataforma de controle proporcionou flexibilidade na programação e desenvolvimento de habilidades técnicas em automação. Este protótipo funcional serve como ferramenta didática para ensino e prática na área de mecatrônica.

Palavras-chave: Arduino, Braço Robótico, Automação, Protótipo.

ABSTRACT

This project aims to integrate and apply knowledge acquired in the Electromechanics Technical Course to a robotic arm prototype controlled by Arduino, using laser-cut MDF for the construction of the structure. The choice of Arduino as the control platform provided flexibility in programming and the development of technical skills in automation. This functional prototype serves as a didactic tool for teaching and practice in the field of mechatronics.

Keywords: Arduino, Robotic Arm, Automation, Prototype.

SUMÁRIO

1. INTRODUÇÃO	10
2. DESENVOLVIMENTO	10
2.1. Funcionamento	12
2.2. Componentes	13
2.2.1 Alimentação	13
2.2.2 Movimentação	13
2.2.3 Controle	14
3. PROJETO	18
3.1 Mecânico	18
3. 2 Eletrônico.....	48
3.3 Programação Digital	49
4. PROCESSO DE FABRICAÇÃO	60
4.1 Seleção de Materiais	60
4.2 Corte das Peças	60
4.3 Montagem do Braço Robótico	61
4.3.1 Montagem da Base com os apoios (pés).....	61
4.3.2 Montagem do motor I.....	62
4.3.3 Montagem da base com a montagem do motor I	63
4.3.4 Montagem do apoio da base na montagem 4.3.3.....	64
4.3.5 Montagem do motor IV	65
4.3.6 Montagem da articulação “1” na montagem do motor IV	66
4.3.7 Montagem do motor III.....	67
4.3.8 Montagem da articulação “6” na montagem do motor III	68
4.3.9 Montagem da estrutura “2”	69
4.3.10 Montagem da estrutura “3” com articulação “4”	70
4.3.11 Montagem das estruturas “2” com as demais montagens	71
4.3.12 Montagem da estrutura com o “suporte 3”	76
4.3.13 União da montagem da base com a montagem da estrutura central ...	78
4.3.14 Montagem do conjunto com a “Articulação 3”	79
4.3.15 Montagem do “Fêmur” e da “Articulação 8” no conjunto.....	80
4.3.16 Montagem da “Articulação 3” na outra extremidade do “Fêmur”	82
4.3.17 Montagem da “Articulação 3” na extremidade que falta do “Fêmur” ..	84
4.3.18 Montagem da “Haste 1”	86

4.3.19 Montagem do motor II.....	88
4.3.20 Montagem das Garras (direita e esquerda) na montagem do motor II .	91
4.3.21 Montagem da articulação nas garras e todo o conjunto do motor II	93
4.3.22 Montagem da garra no conjunto de montagens	95
4.4 Testes e Ajustes	98
5. APLICAÇÕES	106
6. CUSTOS TOTAIS DO PROJETO	107
7. CONSIDERAÇÕES FINAIS	108
8. REFERÊNCIAS.....	109
APÊNDICES:	110
ANEXOS:.....	110

LISTA DE FIGURAS

Figura 1 – Fonte 12V 1A	13
Figura 2 – Micro servomotor SG90	14
Figura 3 – Placa Arduino Uno R3	15
Figura 4 – Sensor Shield V5.0	16
Figura 5 – Módulo Joystick KY-023	18
Figura 6 – Montagem geral do braço robótico - perspectiva isométrica	19
Figura 7 – Apoio da base	20
Figura 8 – Apoio do motor	21
Figura 9 – Articulação I	22
Figura 10 – Articulação II	23
Figura 11 – Articulação III	24
Figura 12 – Articulação IV	25
Figura 13 – Articulação V	26
Figura 14 – Articulação VI	27
Figura 15 – Articulação VII	28
Figura 16 – Articulação VIII	29
Figura 17 – Articulação IX	30
Figura 18 – Base	31
Figura 19 – Controle	32
Figura 20 – Engrenagem - garra direita	33
Figura 21 – Engrenagem - garra esquerda	34
Figura 22 – Estrutura I	35
Figura 23 – Estrutura II	36
Figura 24 – Estrutura III	37
Figura 25 – Fêmur	38

Figura 26 – Haste I	39
Figura 27 – Apoios	40
Figura 28 – Suporte I	41
Figura 29 – Suporte II	42
Figura 30 – Suporte III	43
Figura 31 – Suporte IV	44
Figura 32 – Suporte V	45
Figura 33 – Suporte VI	46
Figura 34 – Suporte VII	47
Figura 35 – Projeto Eletrônico	48
Figura 36 – Base - Peças	61
Figura 37 – Base - Montagem dos pés	61
Figura 38 – Base - Montagem dos parafusos centrais	62
Figura 39 – Motor I - Peças	62
Figura 40 – Motor I - União das peças	63
Figura 41 – Base e motor I	63
Figura 42 – Base e motor I montados	64
Figura 43 – Apoio da base	64
Figura 44 – Motor IV - Peças	65
Figura 45 – Motor IV - União das peças	65
Figura 46 – Articulação I e Motor IV	66
Figura 47 – Articulação I e Motor IV - União das peças	66
Figura 48 – Motor III - Peças	67
Figura 49 – Motor III - União das peças	67
Figura 50 – Articulação VI e Motor III	68
Figura 51 – Articulação VI e Motor III - União das peças	68
Figura 52 – Estrutura II e Kit do Servo - Peças	69

Figura 53 – Estrutura II e Kit do Servo - União das peças	69
Figura 54 – Estrutura III e Articulação IV - Peças	70
Figura 55 – Estrutura III e Articulação IV - União das peças	70
Figura 56 – Montagem do Motor III, Suportes II, Montagem do Motor IV, Montagem da Estrutura III e Montagem da Estrutura II	71
Figura 57 – Montagem do Motor III e Suporte II - Peças	71
Figura 58 – Montagem do Motor III e Suporte II - Peças	72
Figura 59 – Montagem do Motor III e Suporte II - União das peças	72
Figura 60 – Montagem do Motor III, Suporte II e Montagem da Estrutura II - União das peças	73
Figura 61 – Montagem do Motor III, Suporte II, Montagem da Estrutura II e Montagem da Estrutura III- União das peças	73
Figura 62 - União do outro Suporte II	74
Figura 63 - Montagem em outro ângulo	74
Figura 64 - Montagem de todos os conjuntos e peças apresentado na figura 56	75
Figura 65 - Montagem de todos os conjuntos e peças apresentado na figura 56 - União	75
Figura 66 - Montagem do Suporte III no conjunto	76
Figura 67 - Parafusando o Suporte III no conjunto	76
Figura 68 - Parafusando o outro lado do Suporte III no conjunto	77
Figura 69 - Montagem do Suporte III concluída	77
Figura 70 - Montagem da Base e Montagem apresentada na Figura 69	78
Figura 71 - União dos conjuntos	78
Figura 72 - Fixação dos conjuntos por parafusos	79
Figura 73 - Conjunto e Articulação III	79
Figura 74 - Montagem da Articulação III na Articulação I do Motor IV	80
Figura 75 - Montagem geral, Articulação VIII e Fêmur	80
Figura 76 - Fixação da Articulação VIII e Fêmur na parte central do conjunto ..	81

Figura 77 – Montagem das peças concluída	81
Figura 78 – Outro ângulo da montagem	82
Figura 79 – Montagem geral, Articulação III e Apoio	82
Figura 80 – Montagem geral da Articulação III no conjunto	83
Figura 81 - Montagem da mesma Articulação III no Fêmur com o Apoio	83
Figura 82 - Montagem das peças concluída	84
Figura 83 - Montagem geral, Articulação III e Apoio	84
Figura 84 - Montagem da Articulação III no furo que falta do Fêmur	85
Figura 85 - Montagem das peças concluída	85
Figura 86 - Montagem geral, Haste I e Parafusos	86
Figura 87 - Fixação da Haste I no Conjunto Central	86
Figura 88 - Fixação da Haste I na Articulação III	87
Figura 89 - Montagem das peças concluída	87
Figura 90 - Motor II - Peças	88
Figura 91 – Posicionamento do Suporte I no Motor II	88
Figura 92 – Inserção da Articulação IX no Suporte I	89
Figura 93 – Posição final da Articulação IX no Suporte I	89
Figura 94 – Inserção da Suporte IV no Suporte I	90
Figura 95 – Fixação do Suporte V	90
Figura 96 – Outro ângulo da montagem	91
Figura 97 – Garra - Peças	91
Figura 98 – Montagem da garra direita	92
Figura 99 – Montagem da garra esquerda	92
Figura 100 - Articulação da Garra - Peças	93
Figura 101 - Montagem da Articulação na Garra	93
Figura 102 - Montagem da segunda Articulação	94
Figura 103 - Montagem das Articulações no Motor II	94

Figura 104 - Posicionamento e montagem do Suporte do Kit do Servo	95
Figura 105 - Conjunto geral do Braço robótico e Garra	95
Figura 106 - Fixação do lado direito da garra no Braço robótico	96
Figura 107 - Fixação concluída do lado direito da garra	96
Figura 108 - Fixação do ponto superior	97
Figura 109 - Montagem completa do Braço robótico	97
Figura 110 – Modelamento 3D das peças	98
Figura 111 - Montagem da base emborrachada nos joysticks e arduino	99
Figura 112 - Montagem dos potenciômetros no Thinkercad	100
Figura 113 – Ajuste de todas as alavancas dos servos	101
Figura 114 – Teste de pega de um pote R35	102
Figura 115 – Foto dos produtos (potes R35, jumpers, caixa, parafusos e camisa)	102
Figura 116 – Teste com rolhas de vinhos e revestimento emborrachado nas garras	103
Figura 117 – Aquisição de material de revestimento da caixa e peças de silicone	104
Figura 118 – Trecho do código onde foram inseridos valores do “Monitor Serial”...105	
Figura 119 – Conjunto completo do braço robótico	105
Figura 120 - Tabela com os custos totais do projeto	107

1. INTRODUÇÃO

Os braços robóticos desempenham papel fundamental na automação industrial desde o "Unimate" (1950). Com o avanço da eletrônica, projetos educacionais como o deste TCC se tornaram acessíveis.

Este nosso projeto tem como objetivo principal integrar e aplicar os conhecimentos adquiridos ao longo do curso de Técnico em Eletromecânica na Escola Etec Paulino Botelho. O grupo optou por desenvolver e fabricar um braço robótico controlado por Arduino, utilizando o MDF como material estrutural para a fabricação de todo o braço robótico. O foco do projeto é proporcionar aprendizado prático nas áreas de mecânica, eletroeletrônica e eletrônica digital e no futuro aplicarmos esses conhecimentos nos nossos trabalhos.

A escolha do Arduino como plataforma de controle oferece flexibilidade e baixo custo, enquanto a estrutura em MDF foi cortada a laser e montada com parafusos M3, permitindo um protótipo funcional e didático. O projeto visa aprimorar habilidades em projetos, montagem, programação e controle de sistemas mecatrônicos, preparando os integrantes para desafios profissionais na área da mecânica e automação.

2. DESENVOLVIMENTO

O braço robótico desenvolvido neste projeto possui quatro graus de liberdade, controlados por micro servomotores e operados por dois joysticks analógicos conectados a um microcontrolador Arduino. A ideia partiu da observação de sistemas automatizados utilizados em máquinas de ensaios de resistência de materiais, como as máquinas universais de ensaio. Grandes fabricantes, como a Instron, utilizam robôs industriais para realizar tarefas como coleta automática de amostras, leitura de código de barras, posicionamento na máquina e remoção do corpo de prova após o ensaio. Referência: (<https://www.youtube.com/watch?v=HTyq3R2Etnw>).

Com base nesse conceito, buscou-se na internet um projeto de baixo custo que permitisse aplicar os mesmos princípios de movimentação. O modelo escolhido foi obtido no site Thingiverse. Referência: (<https://www.thingiverse.com/thing:360108>), mas os arquivos estavam em polegadas e em formato inadequado para conversão direta. Por isso, foi necessário extrair manualmente as medidas e redesenhar cada peça no SolidWorks, adaptando o modelo para fabricação em MDF com corte a laser. Após o modelamento, a montagem virtual permitiu verificar interferências e realizar ajustes antes da construção do protótipo físico.

A estrutura funcional do braço é composta por:

Um servomotor na base, que realiza o movimento de rotação horizontal em um arco de 180°;

Um servomotor de elevação, que ajusta a altura do braço (movimento vertical);

Um servomotor de extensão, que permite o avanço e recuo da parte frontal do braço (movimento linear);

Um servomotor na garra, responsável pela abertura e fechamento para fixação de objetos.

O controle do sistema é feito por dois joysticks analógicos: o joystick esquerdo aciona a rotação da base e o movimento da garra, enquanto o joystick direito controla os movimentos de elevação e extensão. Os sinais analógicos gerados pelos joysticks são lidos pelas portas analógicas do Arduino e convertidos em sinais PWM, que controlam os servomotores conforme o movimento desejado.

Dessa forma, o projeto permite a simulação prática de um braço robótico industrial com controle manual, servindo como base para estudos em automação aplicada e possíveis adaptações para uso em ambientes produtivos reais.

2.1. Funcionamento

Funcionamento Geral:

O arduíno que é o cérebro do projeto foi programado e testado utilizando o site Tinkercad Referência: (<https://www.tinkercad.com/>). O braço robótico controlado por um Arduino Uno R3 possui quatro microservomotores, controlados por dois joysticks. A fonte de 12 V e 1 A alimenta o sistema através do Sensor Shield V5.0, que facilita a conexão dos componentes.

Descrição dos Movimentos e Controle:

1. Microservomotor I (Base - Movimento Rotacional):
Responsável por girar a base do braço robótico horizontalmente (esquerda e direita).
Controlado pelo eixo X do joysticks esquerdo.
2. Microservomotor II (Estrutura Central - Avanço e Retração):
Movimenta o braço para frente e para trás.
Controlado pelo eixo Y do joystick esquerdo.
3. Microservomotor III (Estrutura Central - Subida e Descida):
Realiza o movimento de elevação e descida do braço, permitindo o alcance de diferentes alturas.
Controlado pelo eixo Y do joystick direito.
4. Microservomotor IV (Garra - Abertura e Fechamento):
Controla a abertura e o fechamento da garra para agarrar ou soltar objetos.
Controlado pelo eixo X do joysticks direito.
5. Ao pressionar os dois botões centrais dos dois joysticks, mudará o sistema de movimentação do braço robótico entre manual e automático (e vice versa).

2.2. Componentes

2.2.1 Alimentação

Componente: 01 Fonte 12V 1ª.

Marca: ZX.

Modelo: Zx-1210.

Especificações técnicas: Entrada: 127/220 VAC e Saída: 12 V – 1 A.

Figura 1 – Fonte 12V 1A



2.2.2 Movimentação

Componente: 04 Micro servomotores SG90.

Marca: Tower Pro.

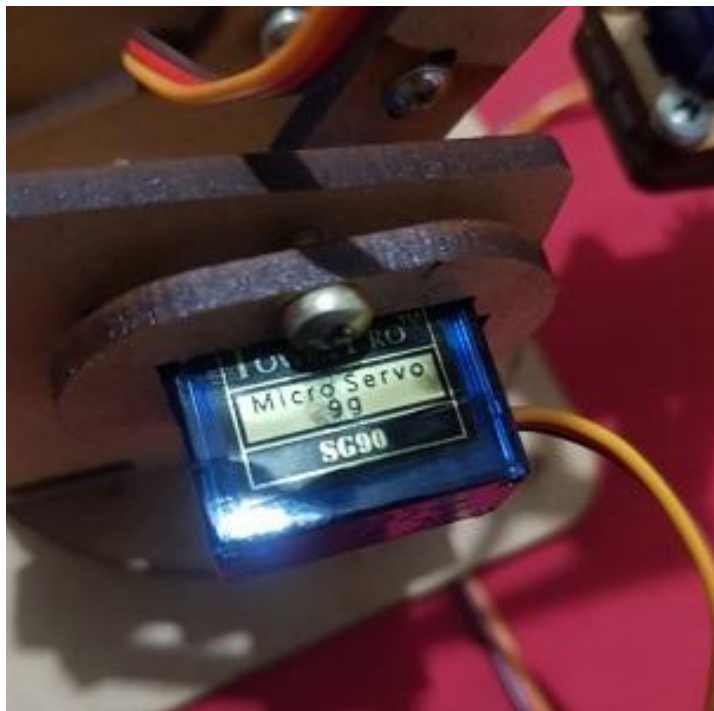
Modelo: 9g SG90.

Especificações técnicas:

– Voltagem de Operação: 3,0 – 7,2V.

- Ângulo de rotação: 180 graus.
- Velocidade: 0,12 seg/60Graus (4,8V) sem carga.
- Torque: 1,2 kg.cm (4,8V) e 1,6 kg.cm (6,0V).
- Temperatura de Operação: -30C ~ +60C.
- Tipo de Engrenagem: Nylon.

Figura 2 – Micro servomotor SG90



2.2.3 Controle

Componente: 01 Arduino Uno R3.

Características Gerais:

Modelo: Arduino Uno R3 (SMD).

Microcontrolador: ATmega328P (versão SMD).

Chip USB-Serial: CH340 (em vez do FTDI ou Atmega16U2 nas versões originais).

Tensão de Operação: 5V.

Tensão de Alimentação Recomendada: 7V a 12V.

Tensão de Alimentação Máxima: 6V a 20V (não recomendada continuamente).

Especificações Técnicas:

Microcontrolador: ATmega328P (32 KB de memória flash, 2 KB SRAM, 1 KB EEPROM).

Clock: 16 MHz.

Tensão de Entrada (recomendada): 7V a 12V.

Pinos Digitais de I/O: 14 (6 com PWM).

Pinos de Entrada Analógica: 6 (A0 a A5).

Corrente Máxima por Pino I/O: 40 mA.

Corrente Máxima no Pino 3,3V: 50 mA.

Memória Flash: 32 KB (0,5 KB utilizados pelo bootloader).

SRAM: 2 KB.

EEPROM: 1 KB.

Figura 3 – Placa Arduino Uno R3



Componente: 01 Sensor Shield V5.0.

Modelo: Sensor Shield V5.0.

Compatibilidade: Placas Arduino Uno, Mega, Leonardo e similares.

Função: Expansão de portas para facilitar a conexão de sensores, atuadores e módulos externos sem necessidade de solda.

Especificações Técnicas:

Portas Digitais: D0 a D13 — Conectores de 3 pinos (Sinal, VCC e GND).

Portas Analógicas: A0 a A5 — Conectores de 3 pinos (Sinal, VCC e GND).

Portas PWM: Compatíveis com os pinos PWM do Arduino Uno (D3, D5, D6, D9, D10, D11).

Conectores de Comunicação:

UART (Serial): Conector de 4 pinos (TX, RX, VCC e GND).

I2C: Pinos dedicados SDA e SCL (compatível com I2C do Arduino).

SPI: Compatível com os pinos do Arduino Uno (D10 a D13).

Porta de Alimentação: Bornes para alimentação externa de 5V a 12V.

Compatibilidade de Alimentação: 5V e 3.3V (via Arduino).

Figura 4 – Sensor Shield V5.0



Componente: 02 Joysticks KY-023.

Modelo: KY-023.

Tipo: Módulo de joystick analógico bidimensional com botão.

Compatibilidade: Placas Arduino, Raspberry Pi, ESP32, ESP8266 e outros microcontroladores.

Especificações Técnicas:

Tensão de Operação: 3,3V a 5V (compatível com níveis lógicos de 3,3V e 5V).

Corrente de Operação: Aproximadamente 10 mA.

Saídas:

2 saídas analógicas (eixo X e eixo Y).

1 saída digital (botão de pressão).

Tipo de Saída:

Analógica: Faixa de 0V a 5V (para uso com Arduino) ou 0V a 3,3V (para uso com ESP32, ESP8266).

Digital: Nível baixo (0V) ao pressionar o botão.

Resistência do Potenciômetro: 10 k Ω (para cada eixo X e Y).

Pinos de Conexão:

GND: Terra (negativo).

VCC: Alimentação (3,3V ou 5V).

VRx: Saída analógica do eixo X (leitura de 0 a 1023 no Arduino).

VRy: Saída analógica do eixo Y (leitura de 0 a 1023 no Arduino).

SW: Saída digital do botão (valor LOW quando pressionado).

Dimensões e Peso:

Dimensões: Aproximadamente 34 mm x 26 mm x 32 mm (altura considerando a alavanca).

Peso: Cerca de 12g.

Figura 5 – Módulo Joystick KY-023



3. PROJETO

3.1 Mecânico

Modelagem 3D realizada no SolidWorks, com detalhamento técnico de todas as peças.

Figura 6 – Montagem geral do braço robótico - perspectiva isométrica

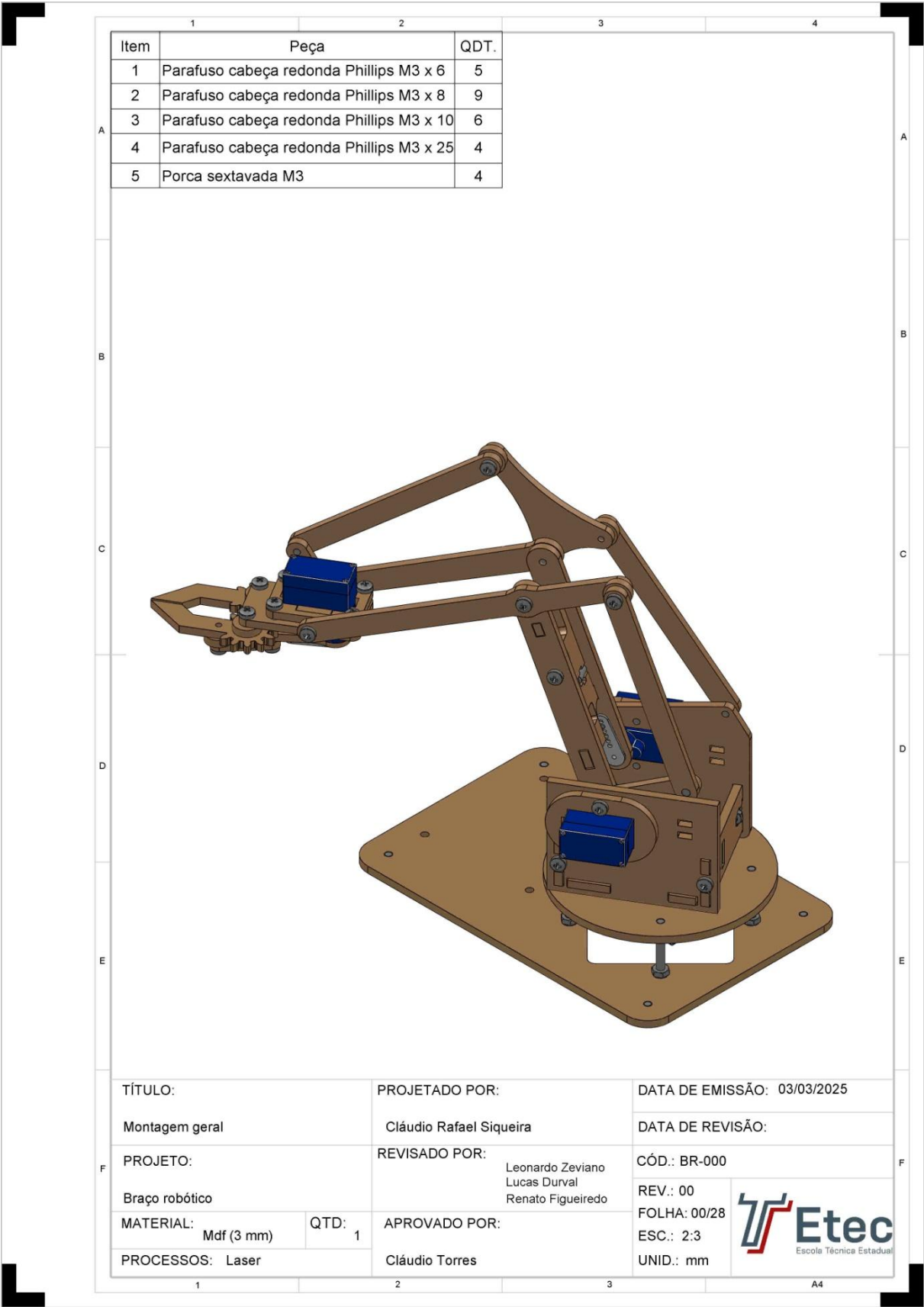


Figura 7 – Apoio da base

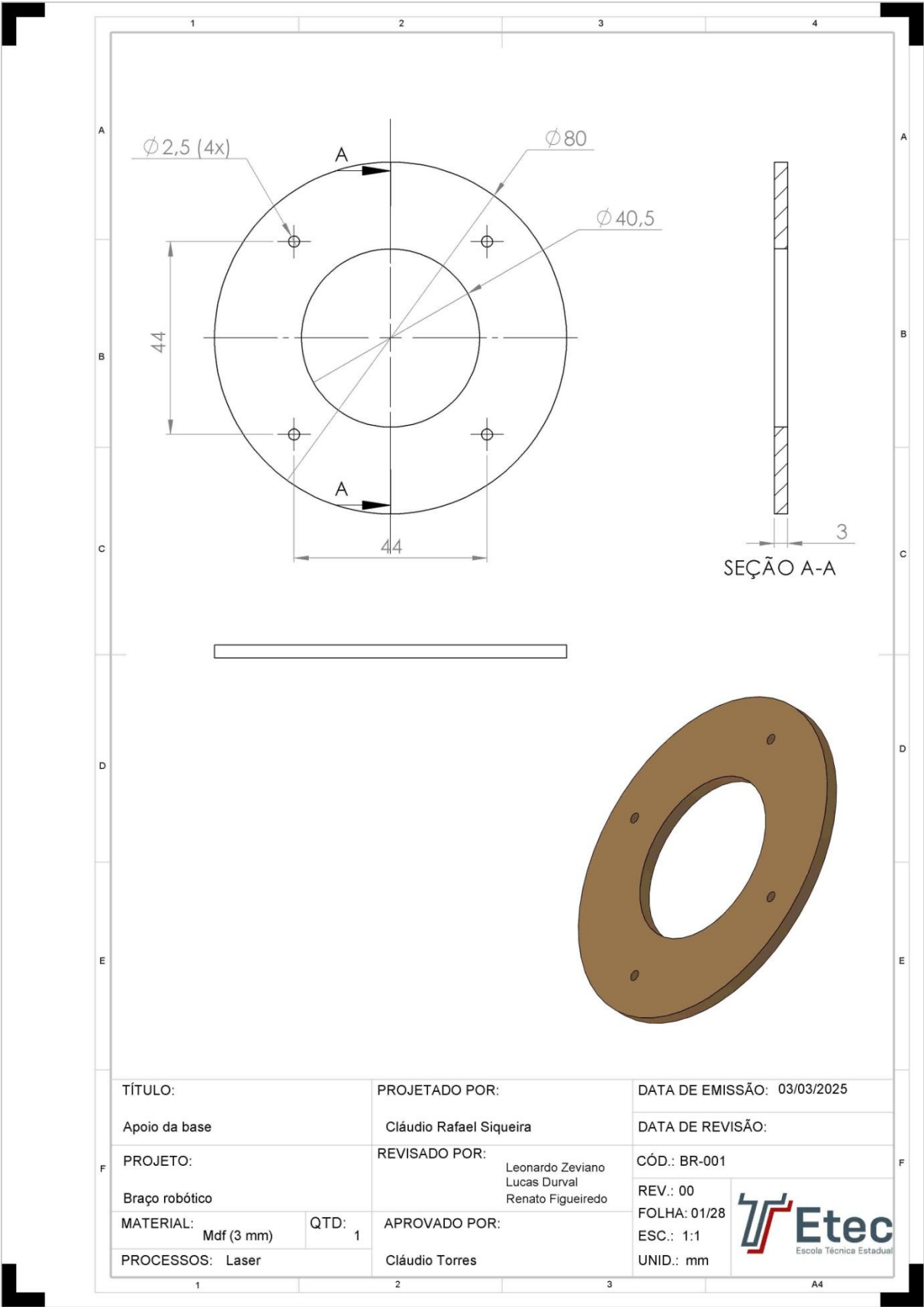


Figura 8 – Apoio do motor

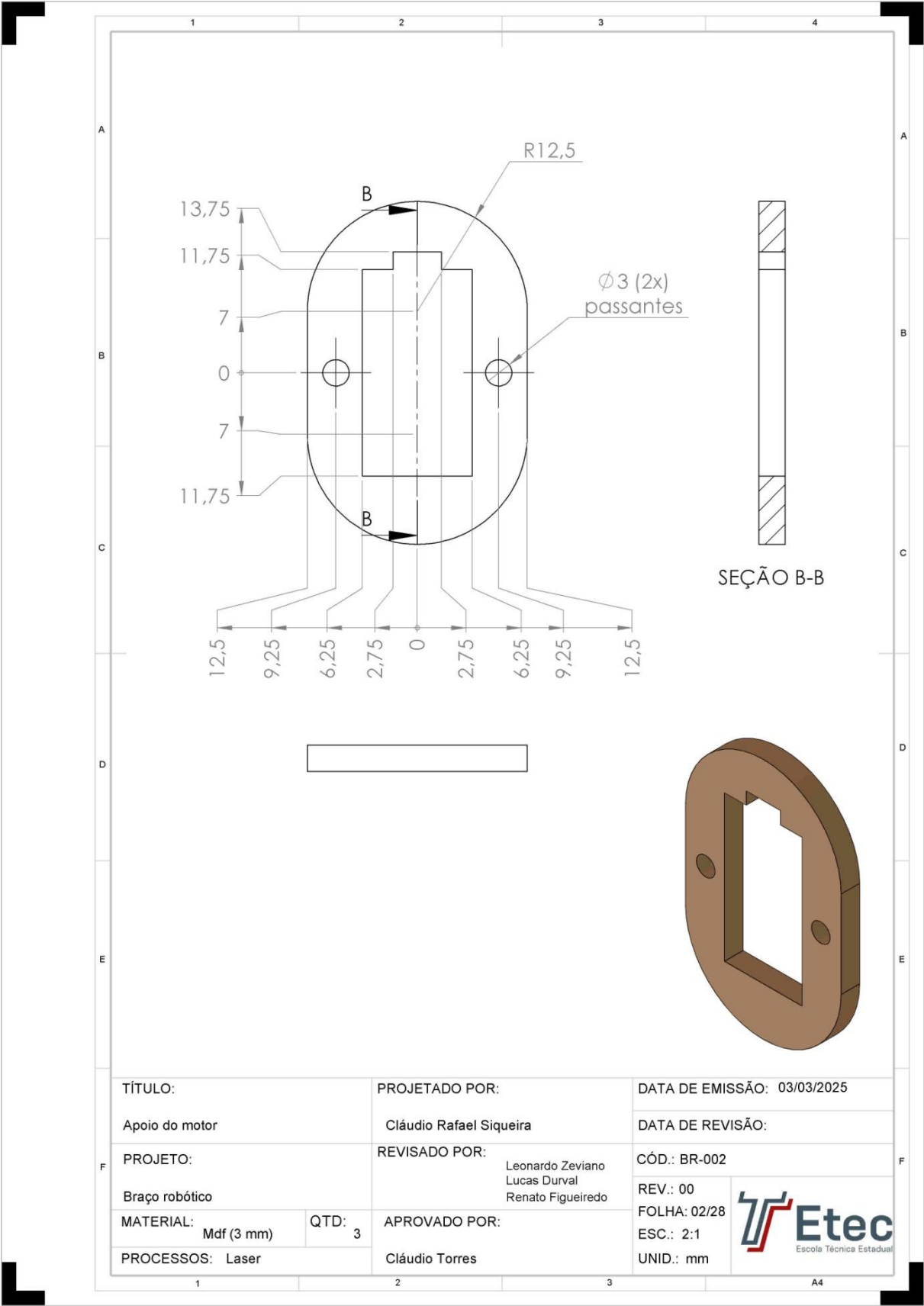


Figura 9 – Articulação I

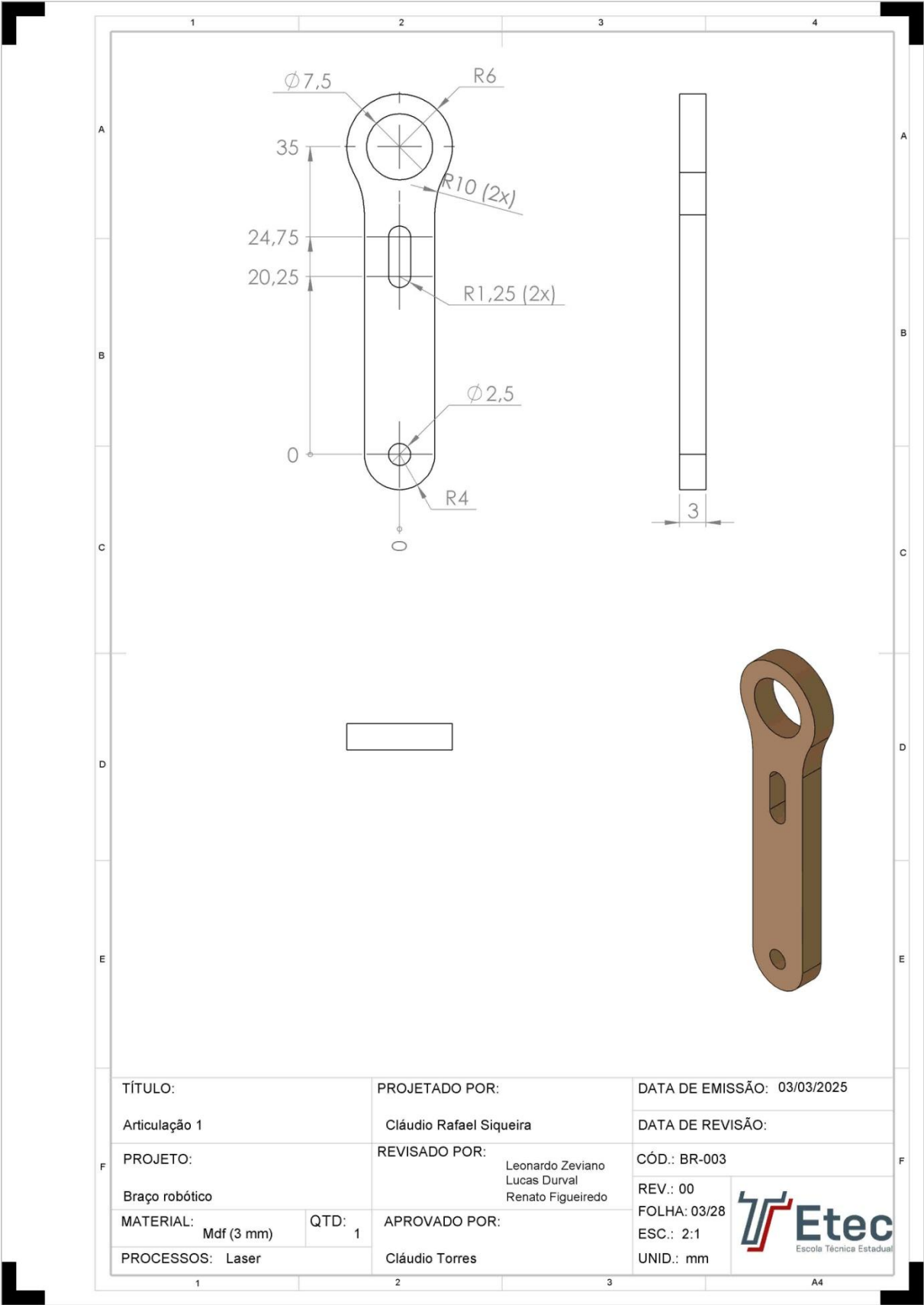


Figura 10 – Articulação II

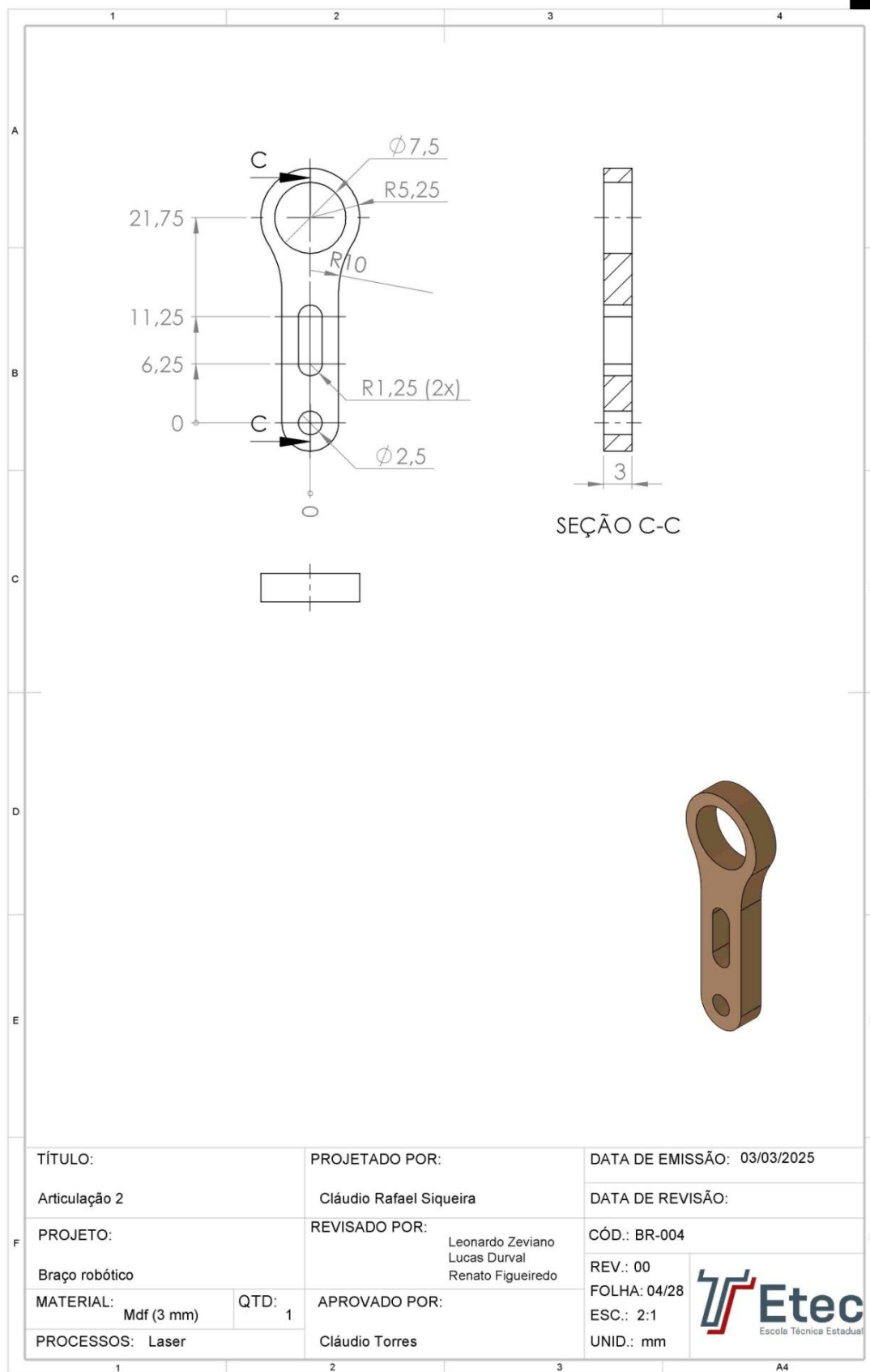


Figura 11 – Articulação III

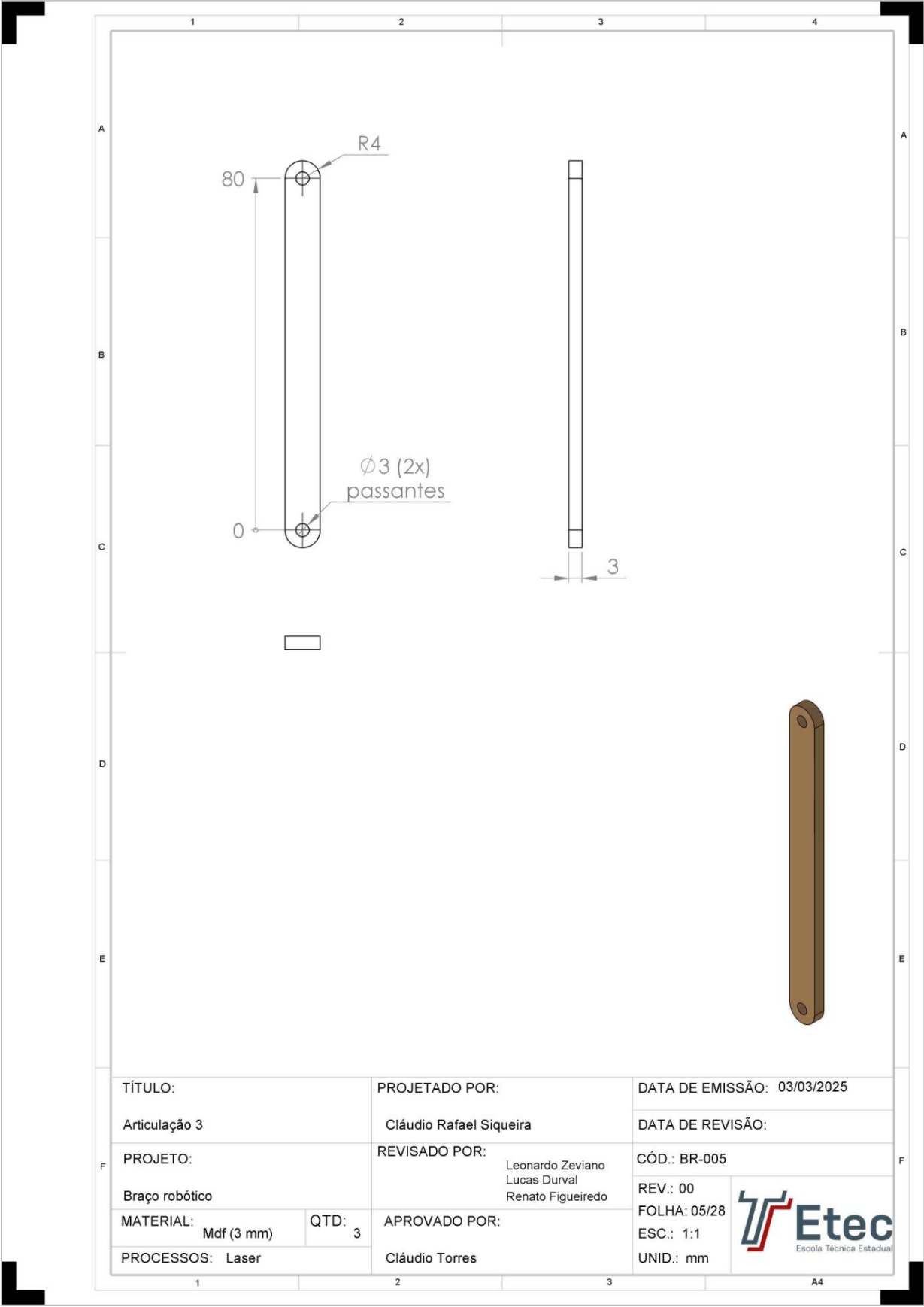


Figura 12 – Articulação IV

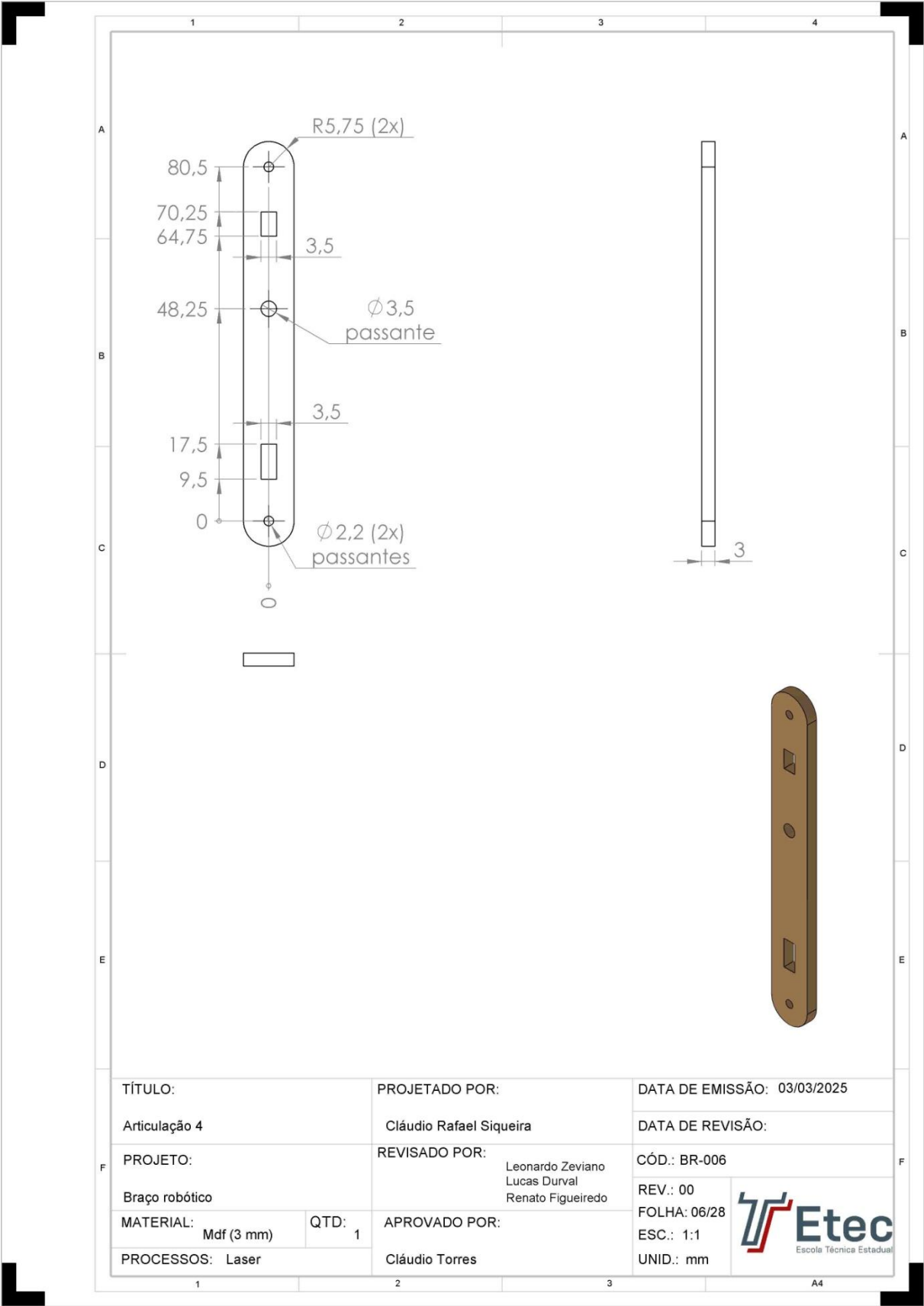


Figura 13 – Articulação V

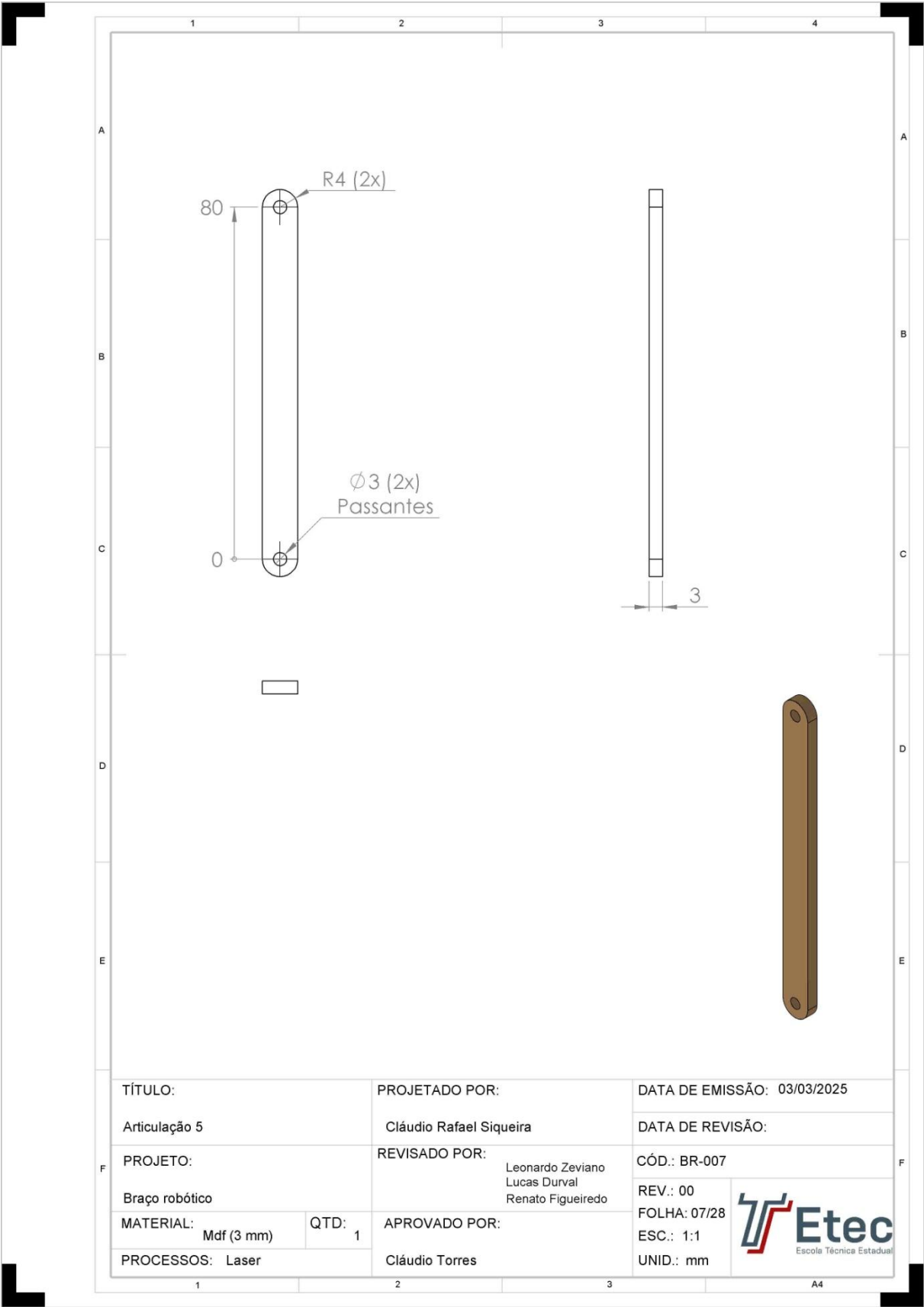


Figura 14 – Articulação VI

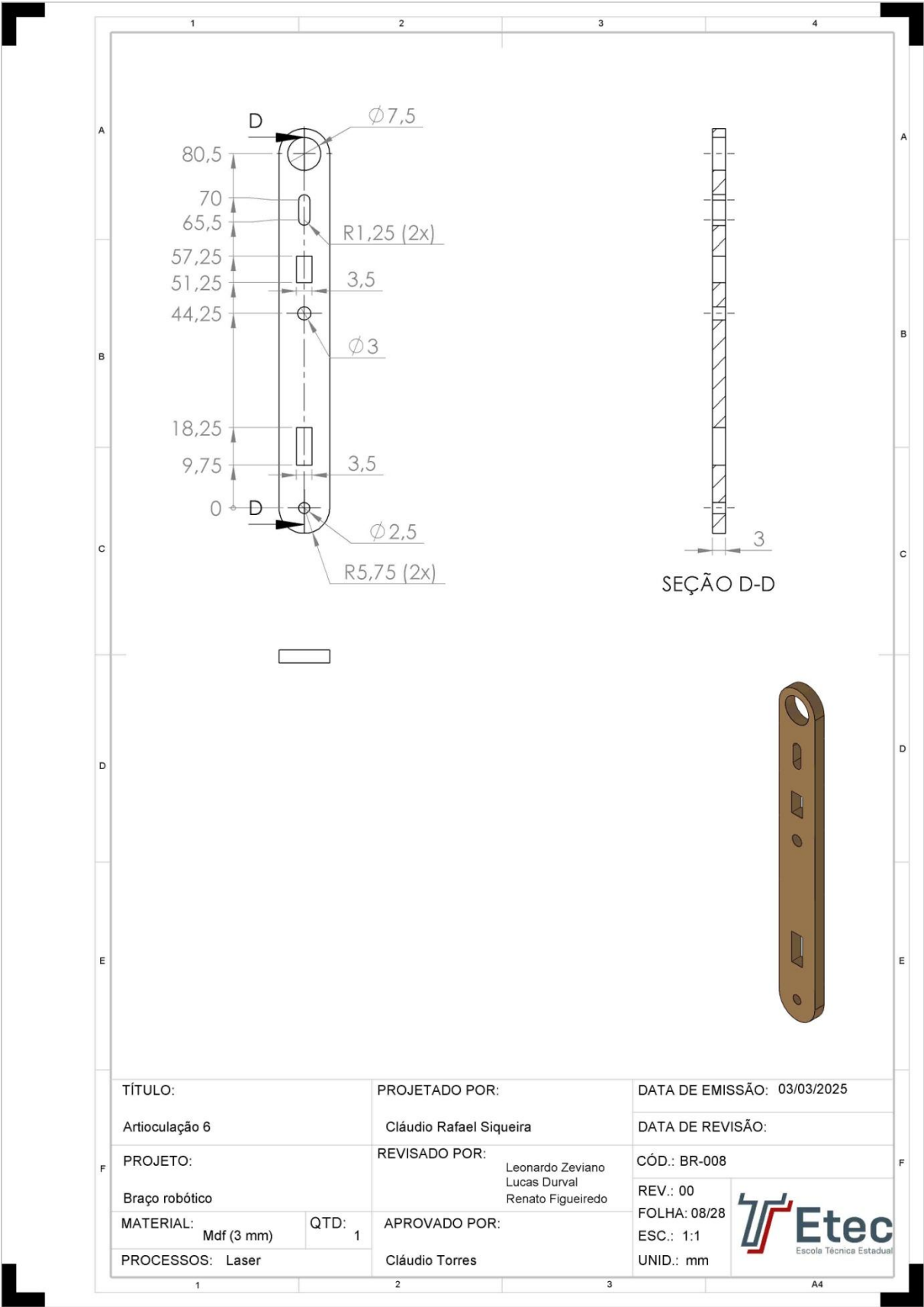


Figura 15 – Articulação VII

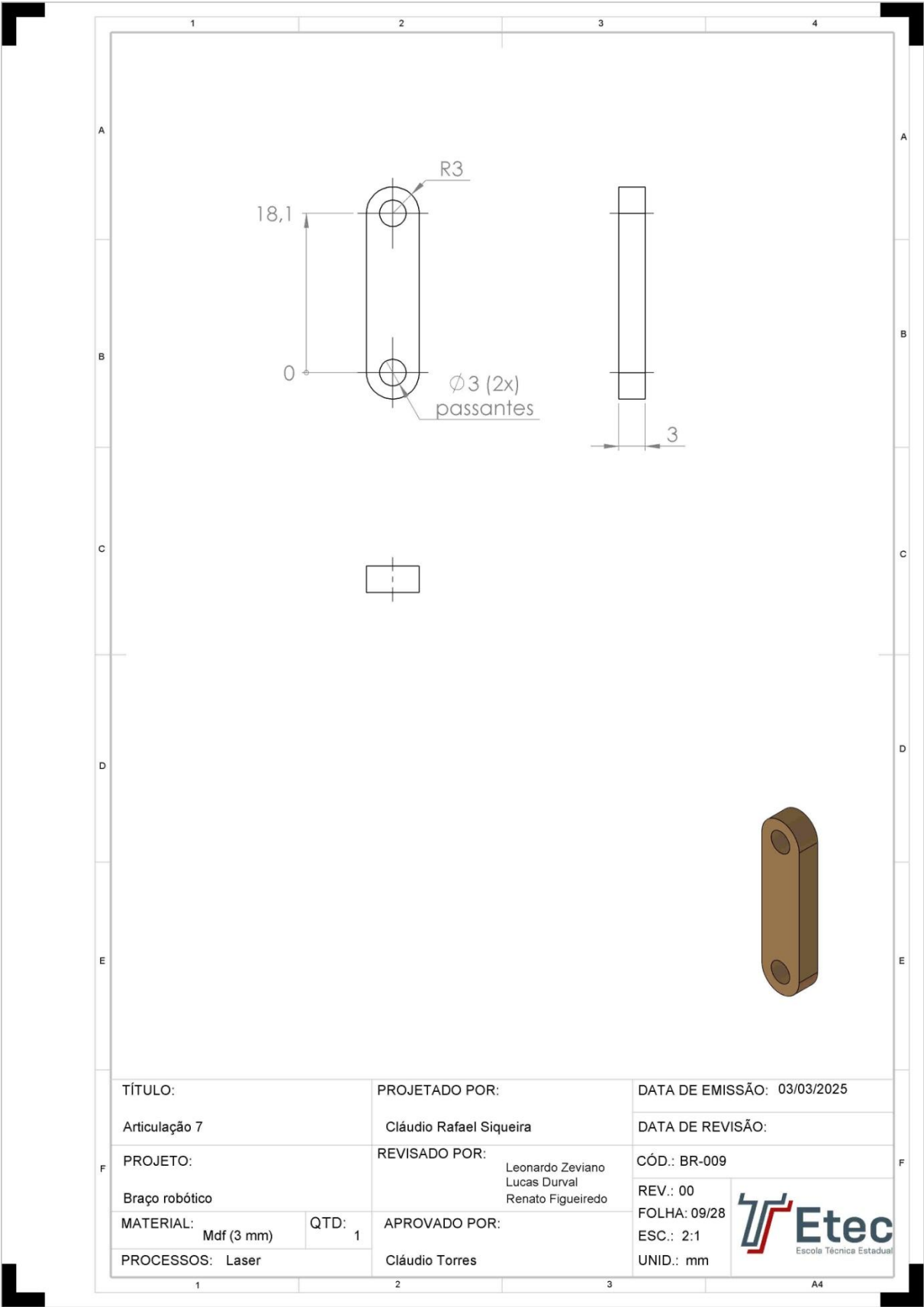


Figura 16 – Articulação VIII

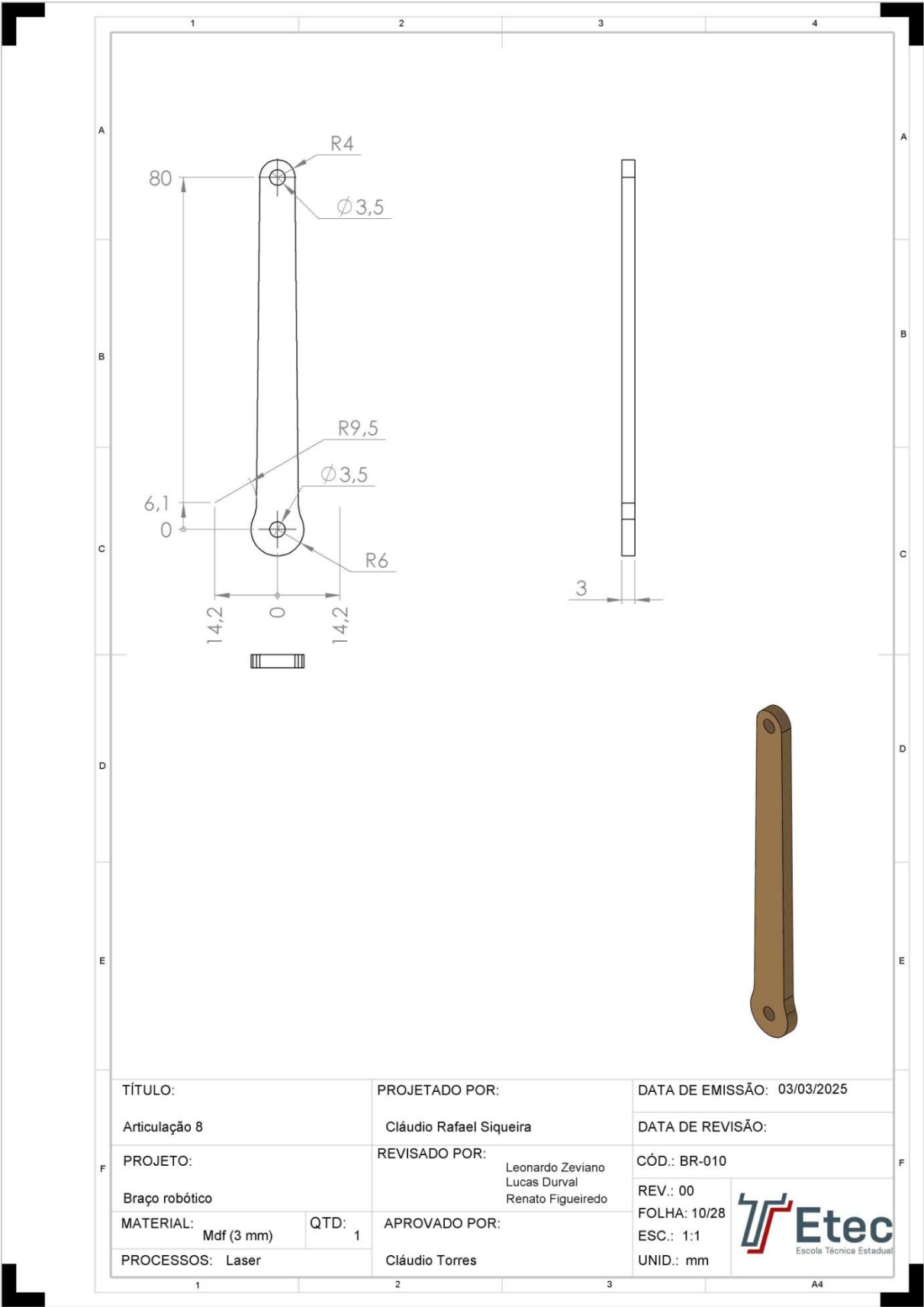


Figura 17 – Articulação IX

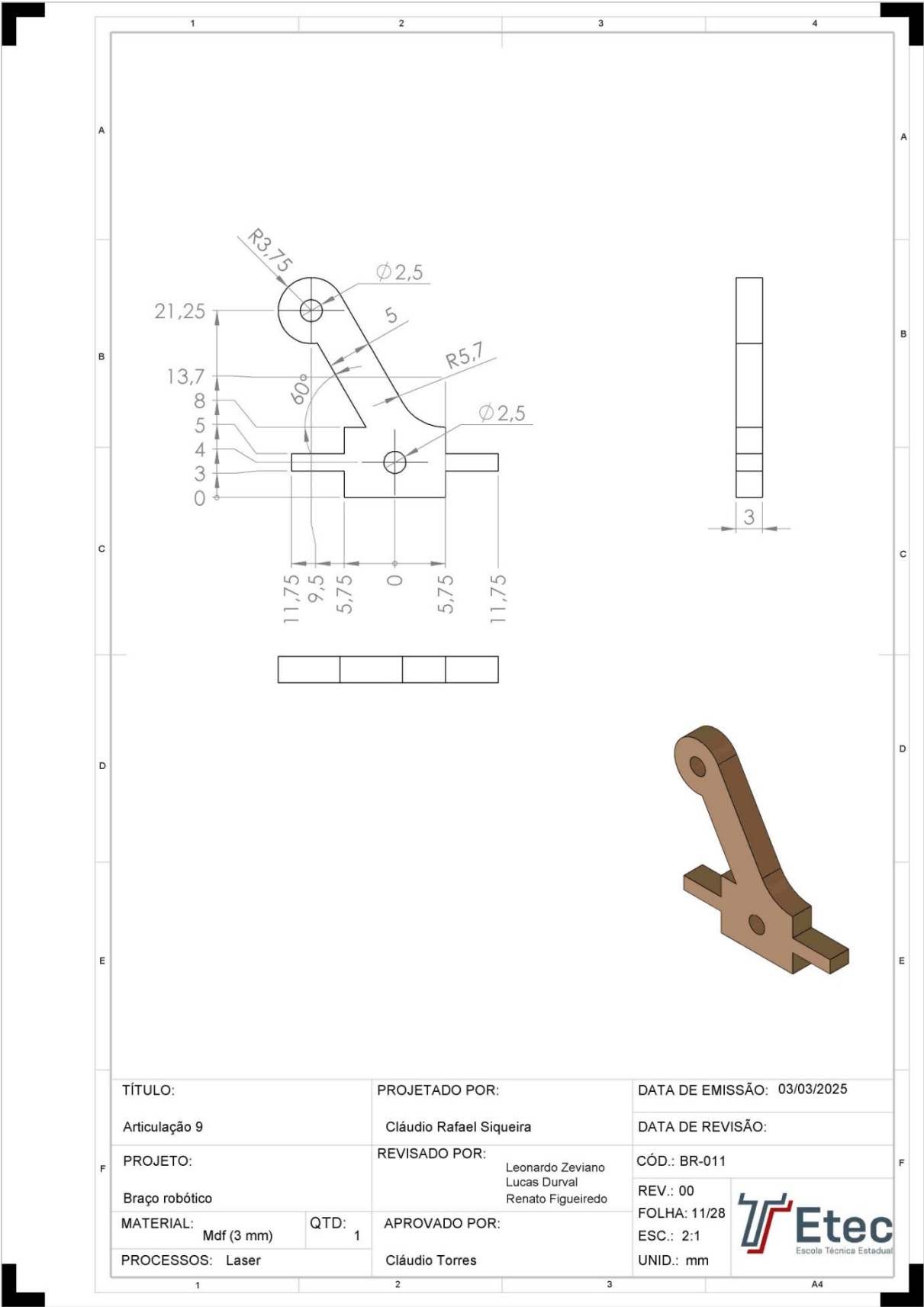


Figura 18 – Base

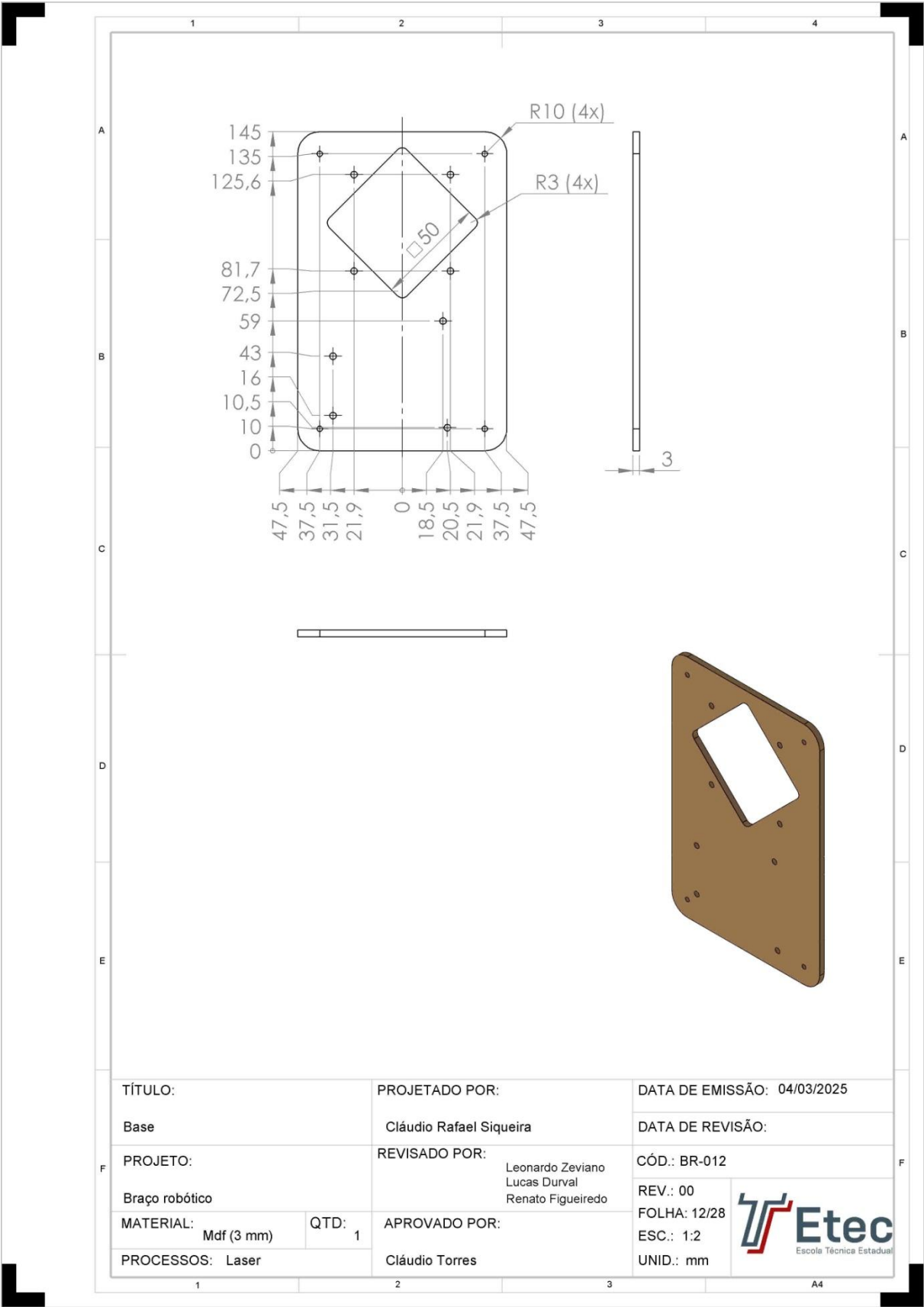


Figura 20 – Engrenagem - garra direita

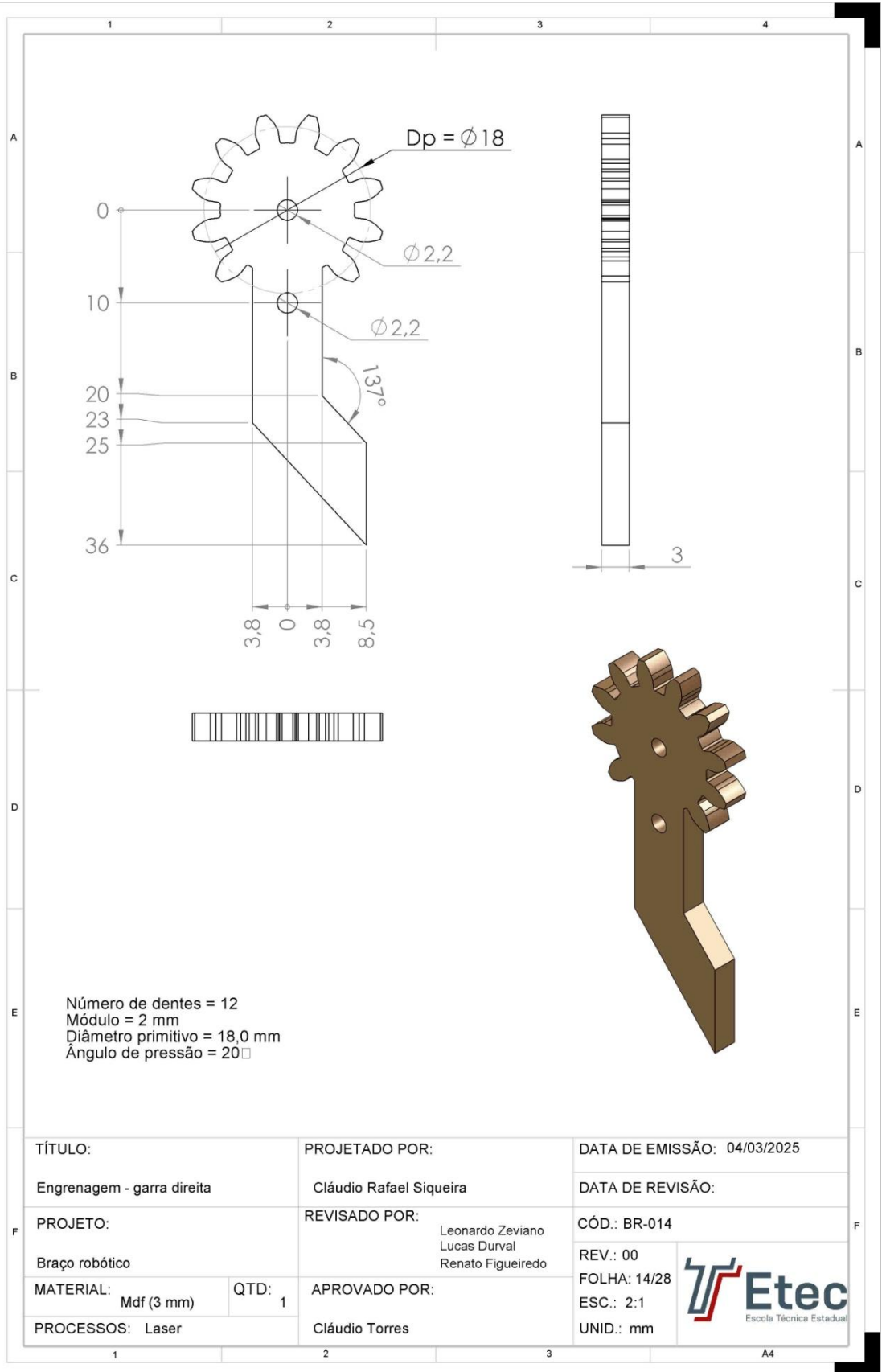


Figura 21 – Engrenagem - garra esquerda

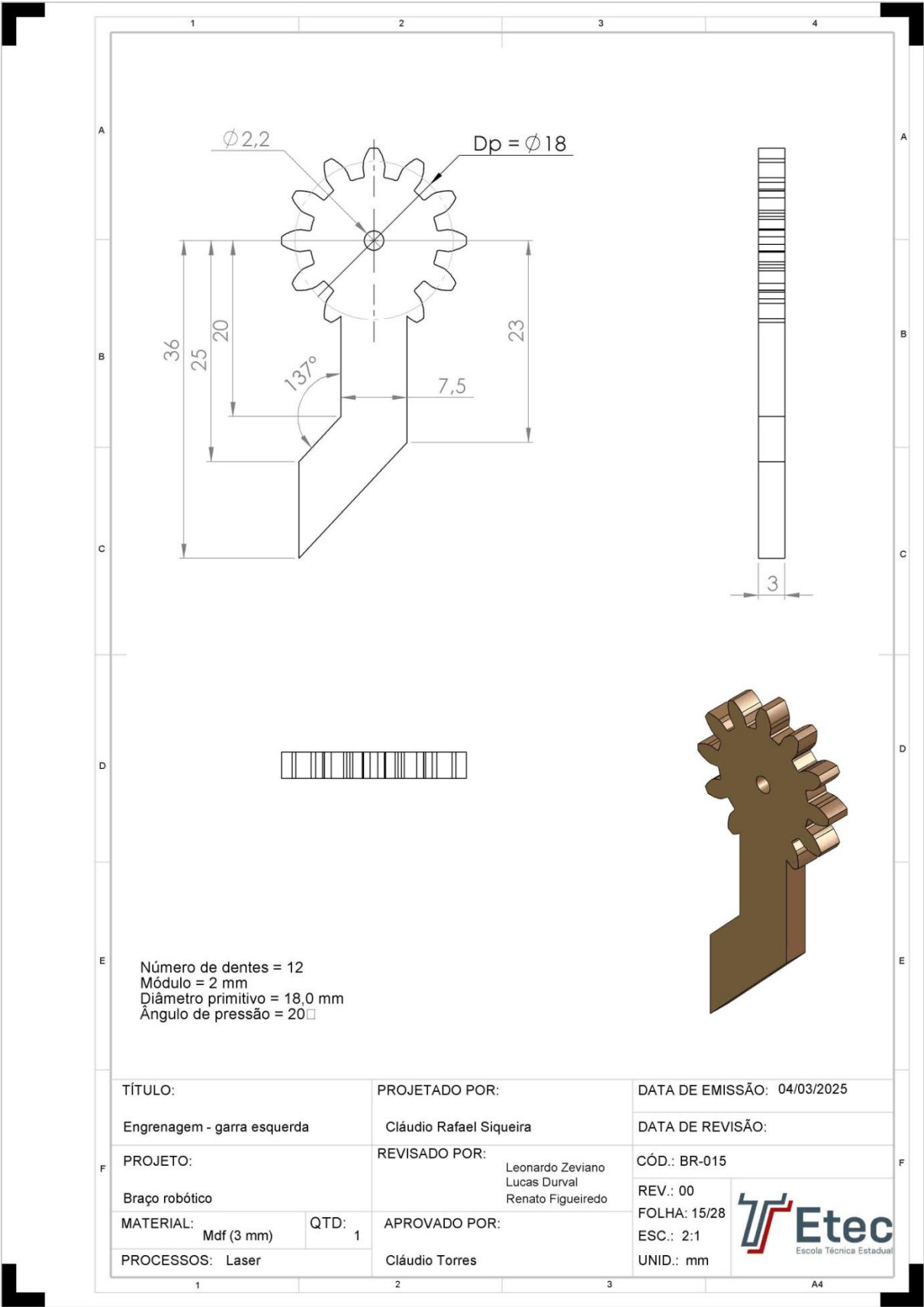


Figura 22 – Estrutura I

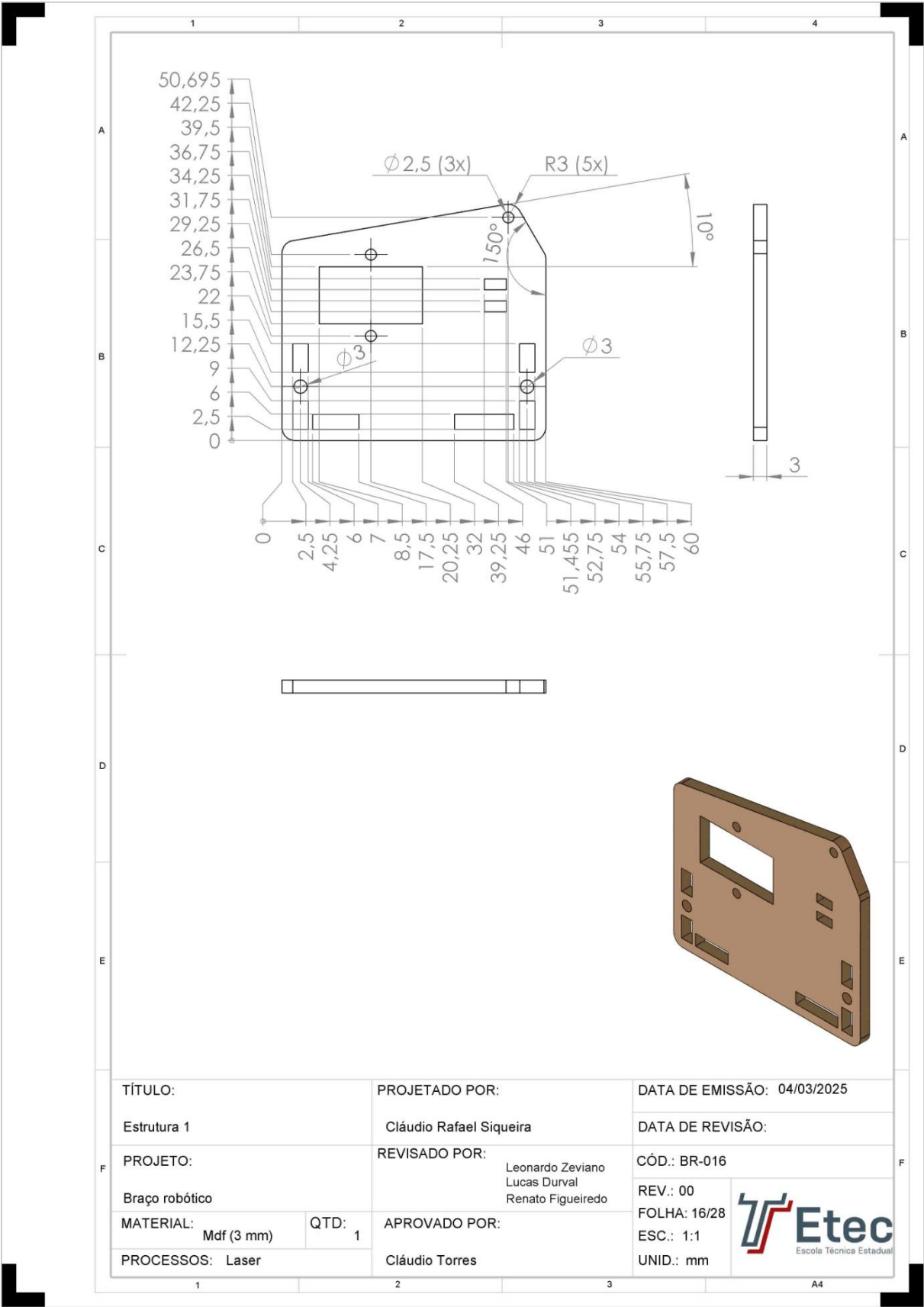


Figura 23 – Estrutura II

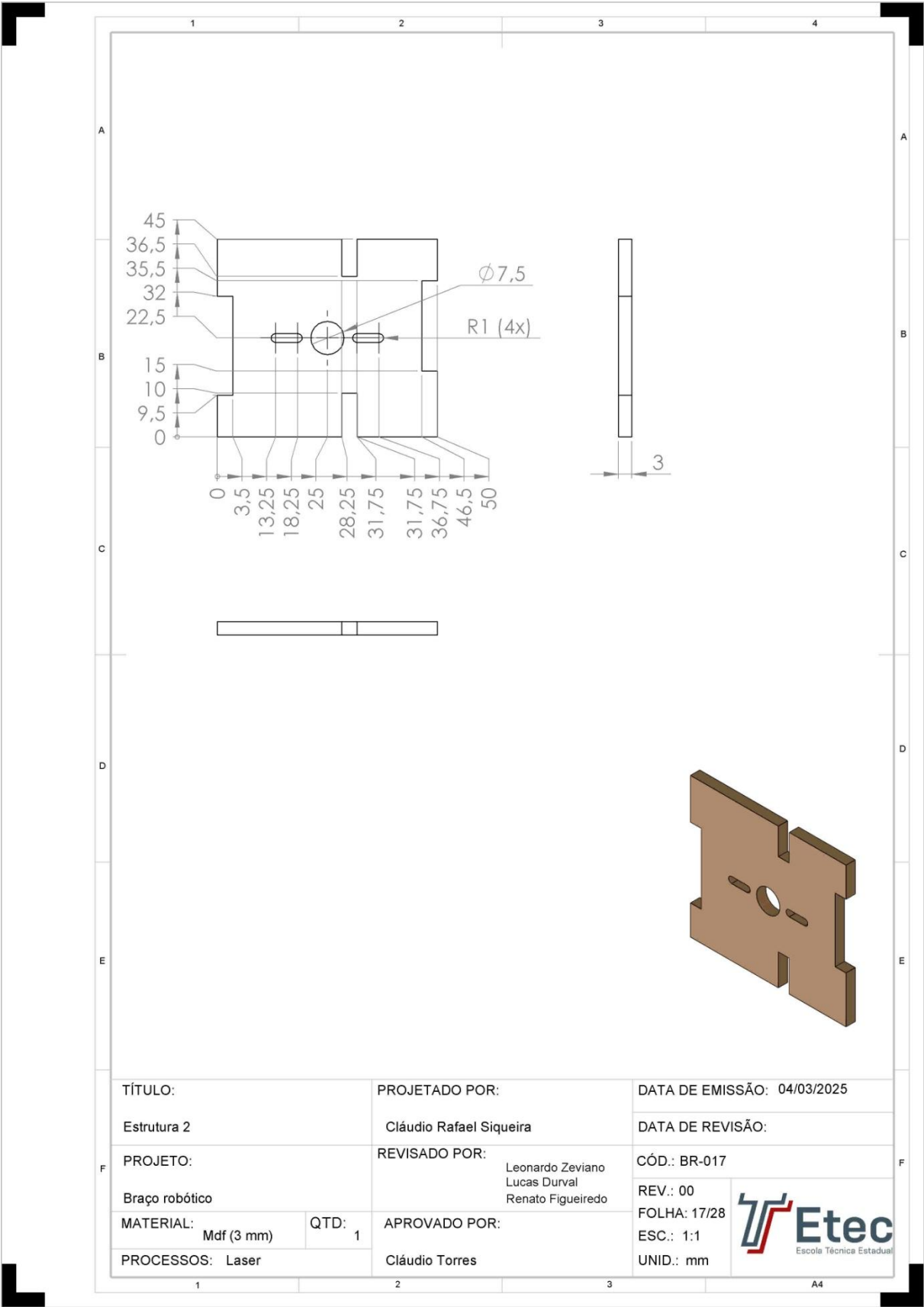


Figura 24 – Estrutura III

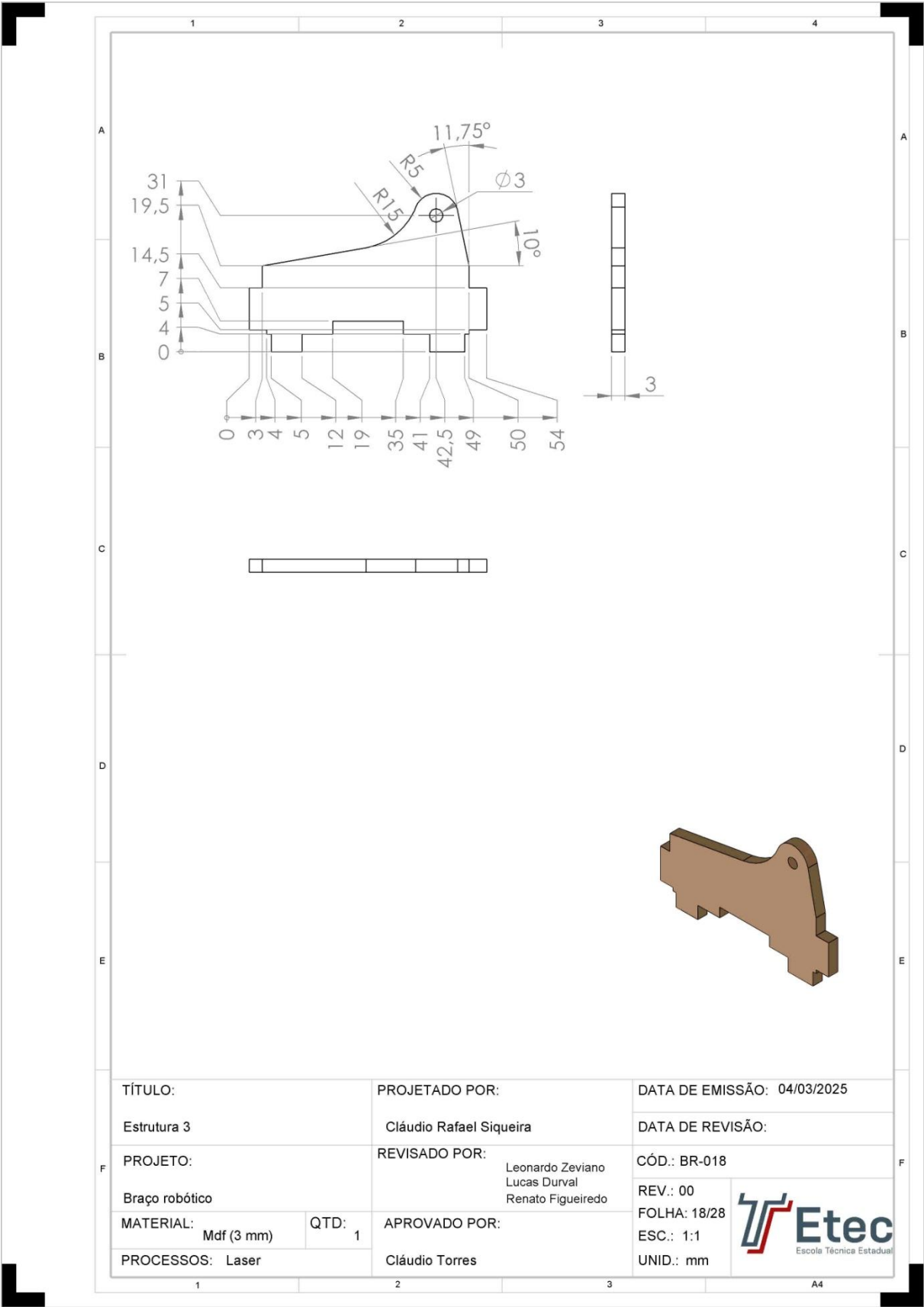


Figura 25 – Fêmur

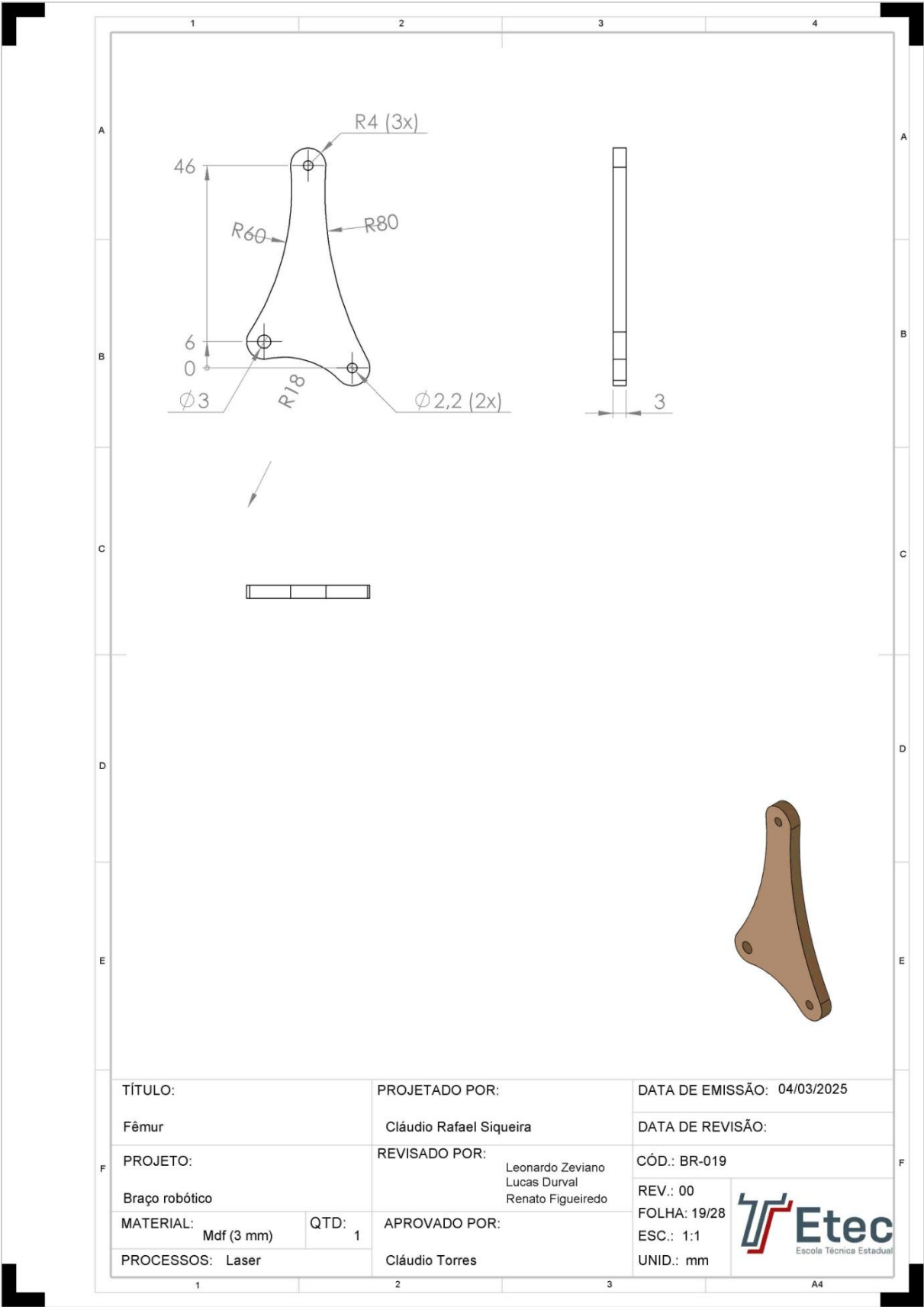


Figura 26 – Haste I

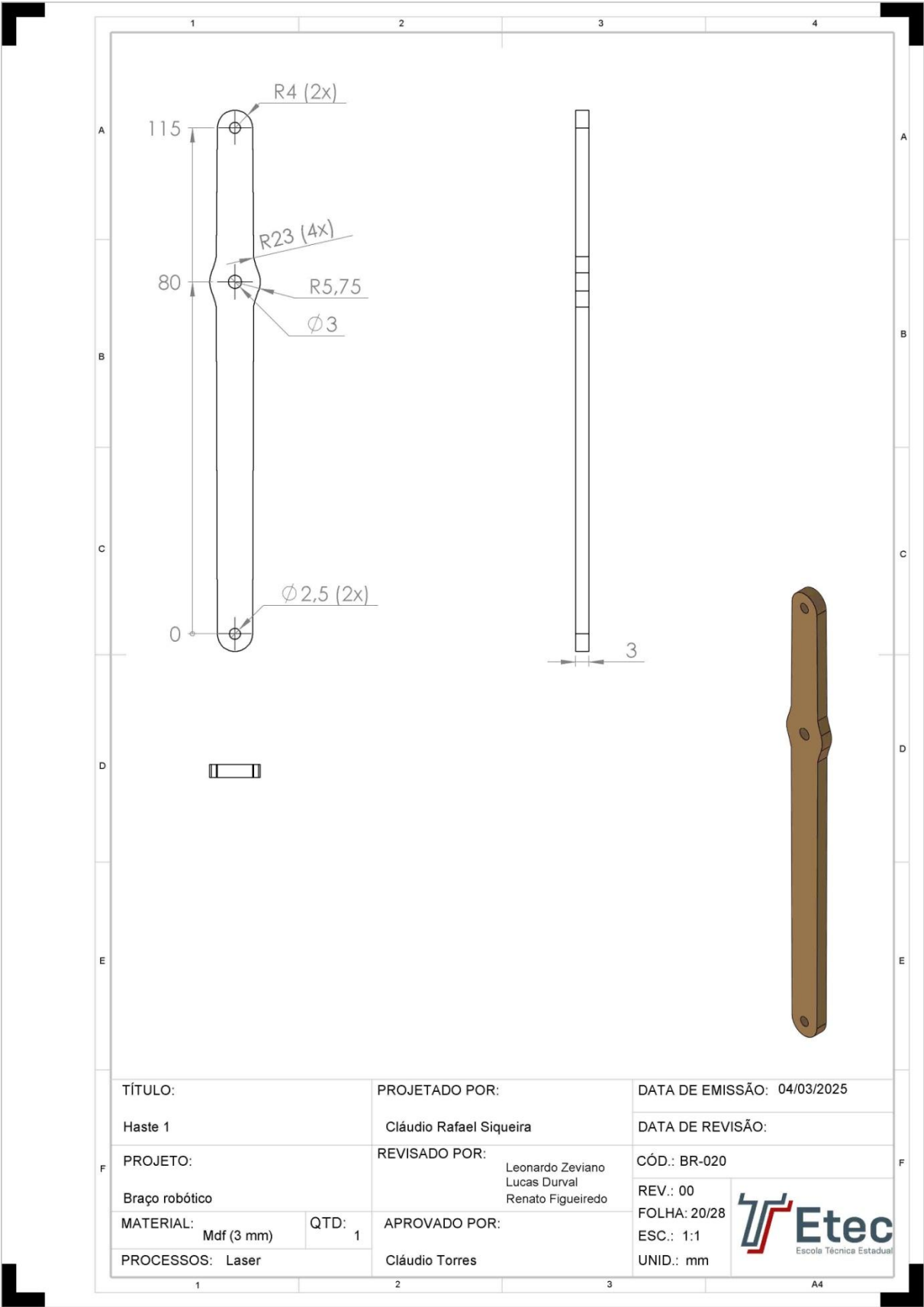


Figura 27 – Apoios

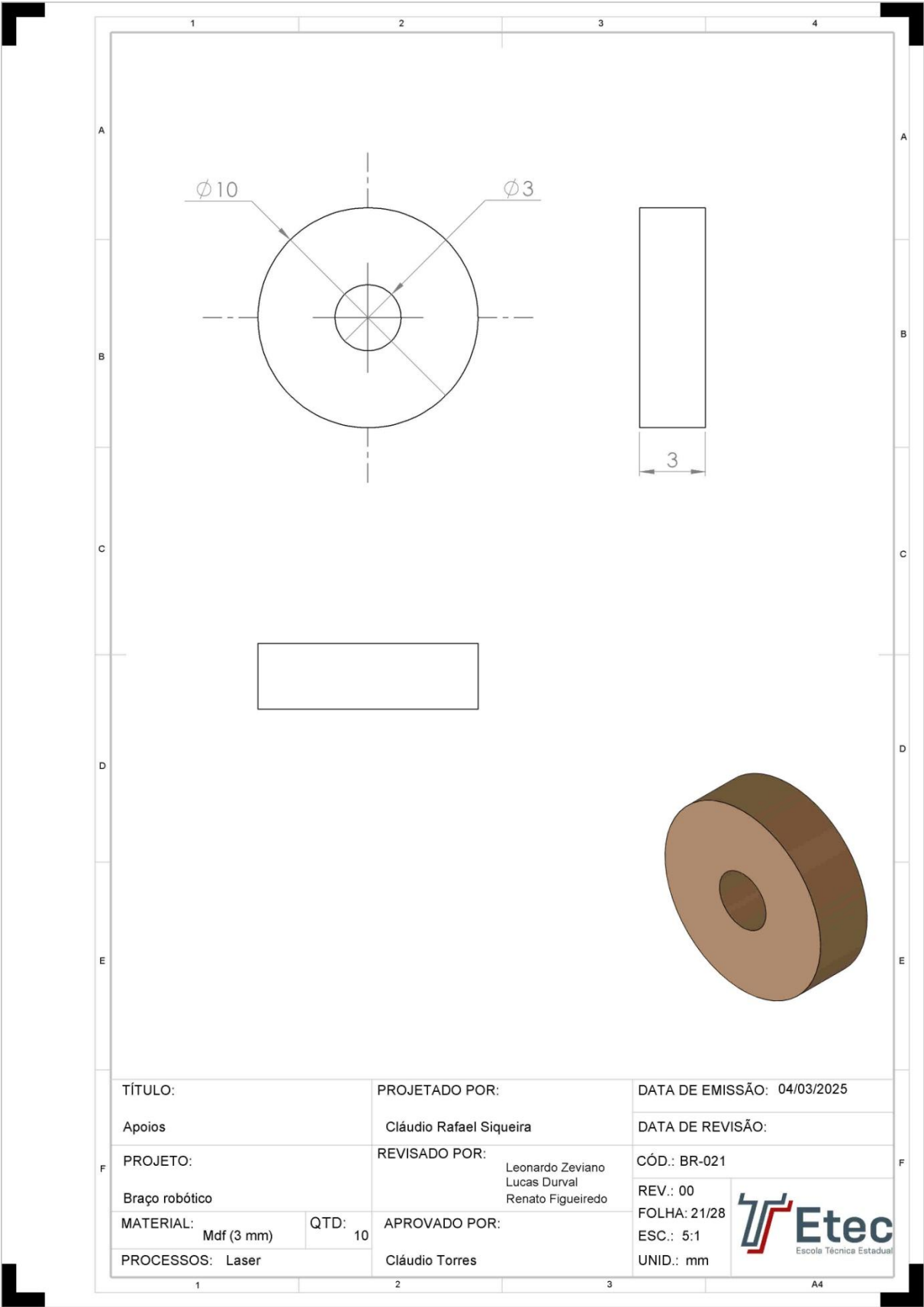


Figura 28 – Suporte I

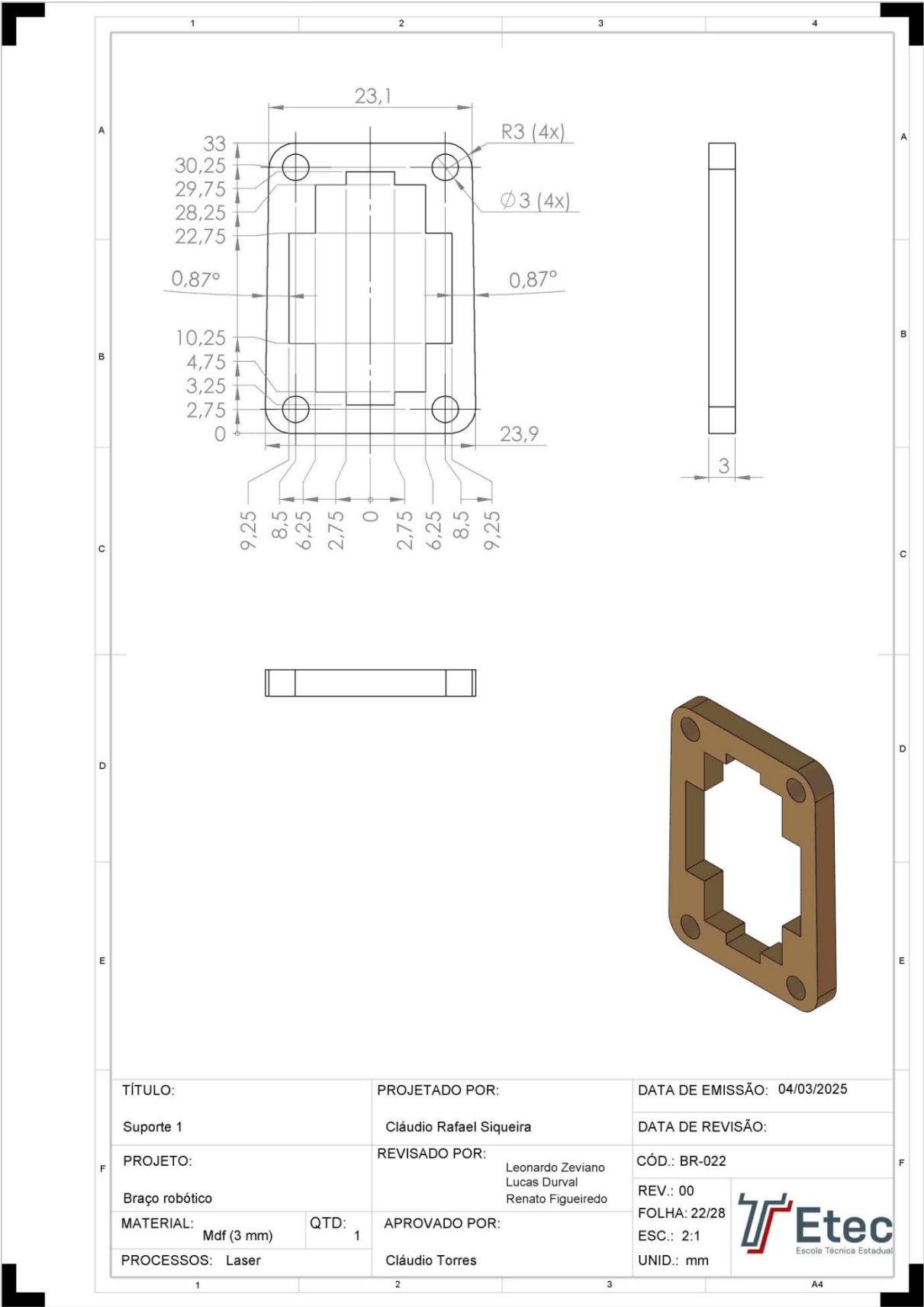


Figura 29 – Suporte II

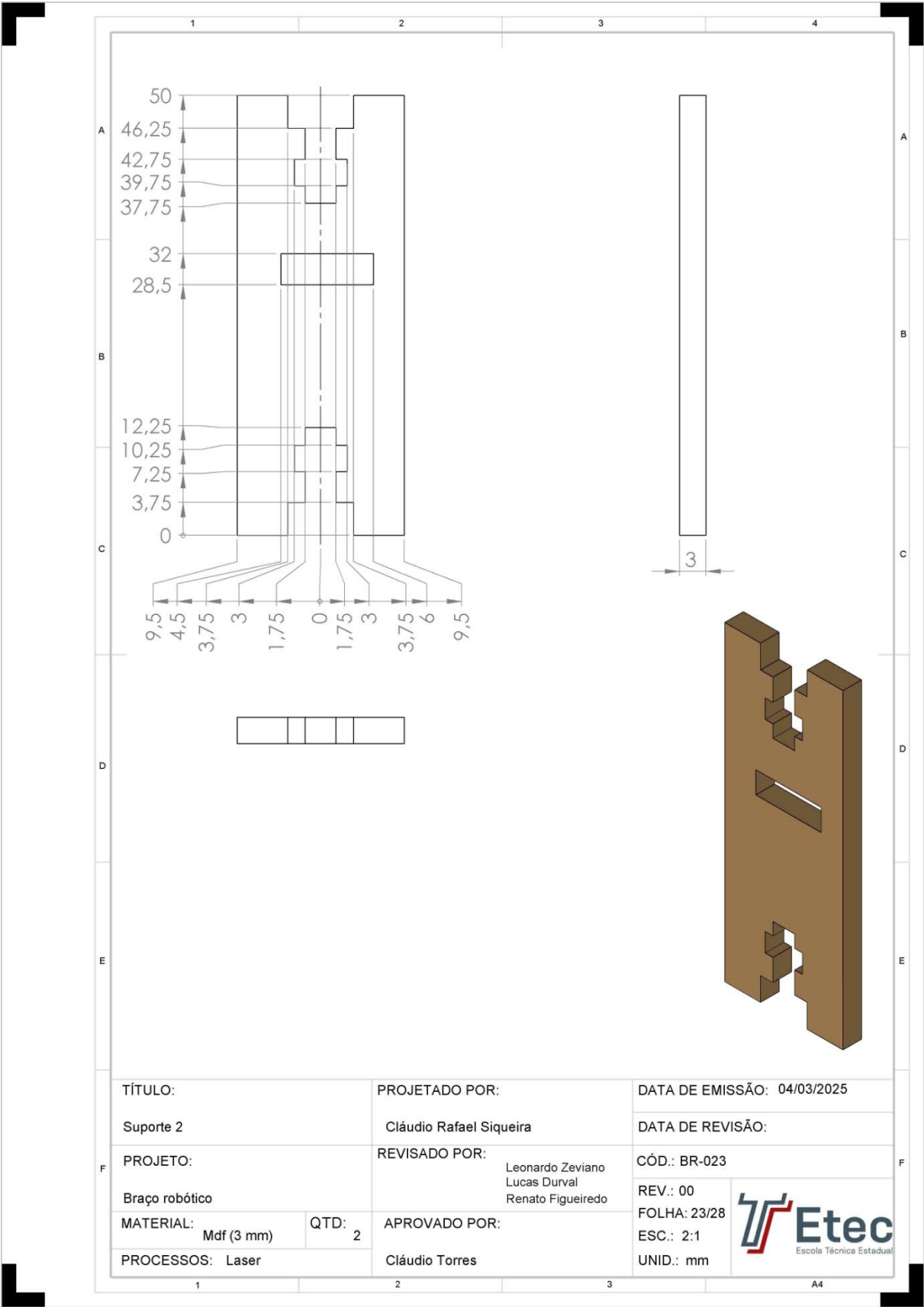


Figura 30 – Suporte III

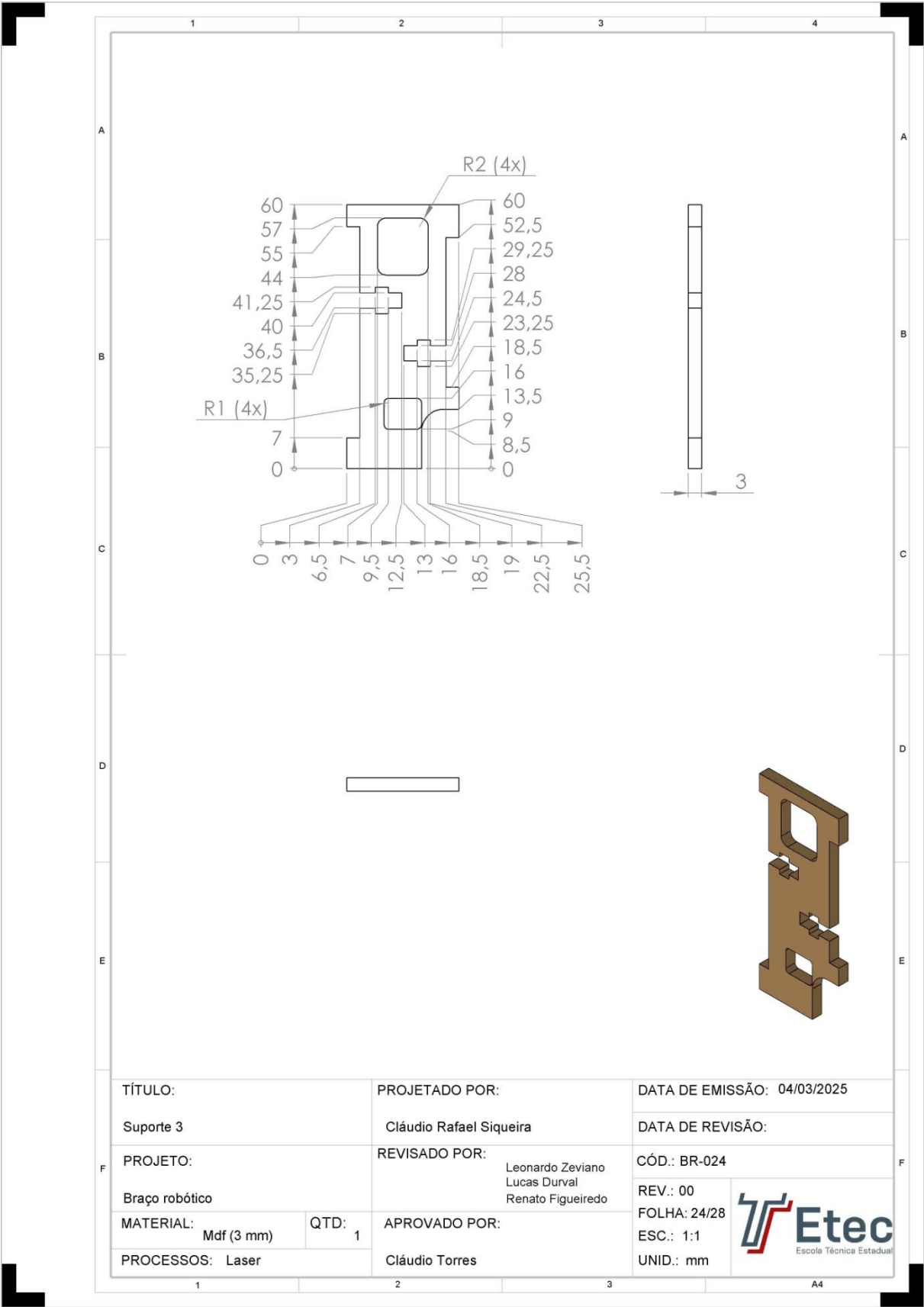


Figura 31 – Suporte IV

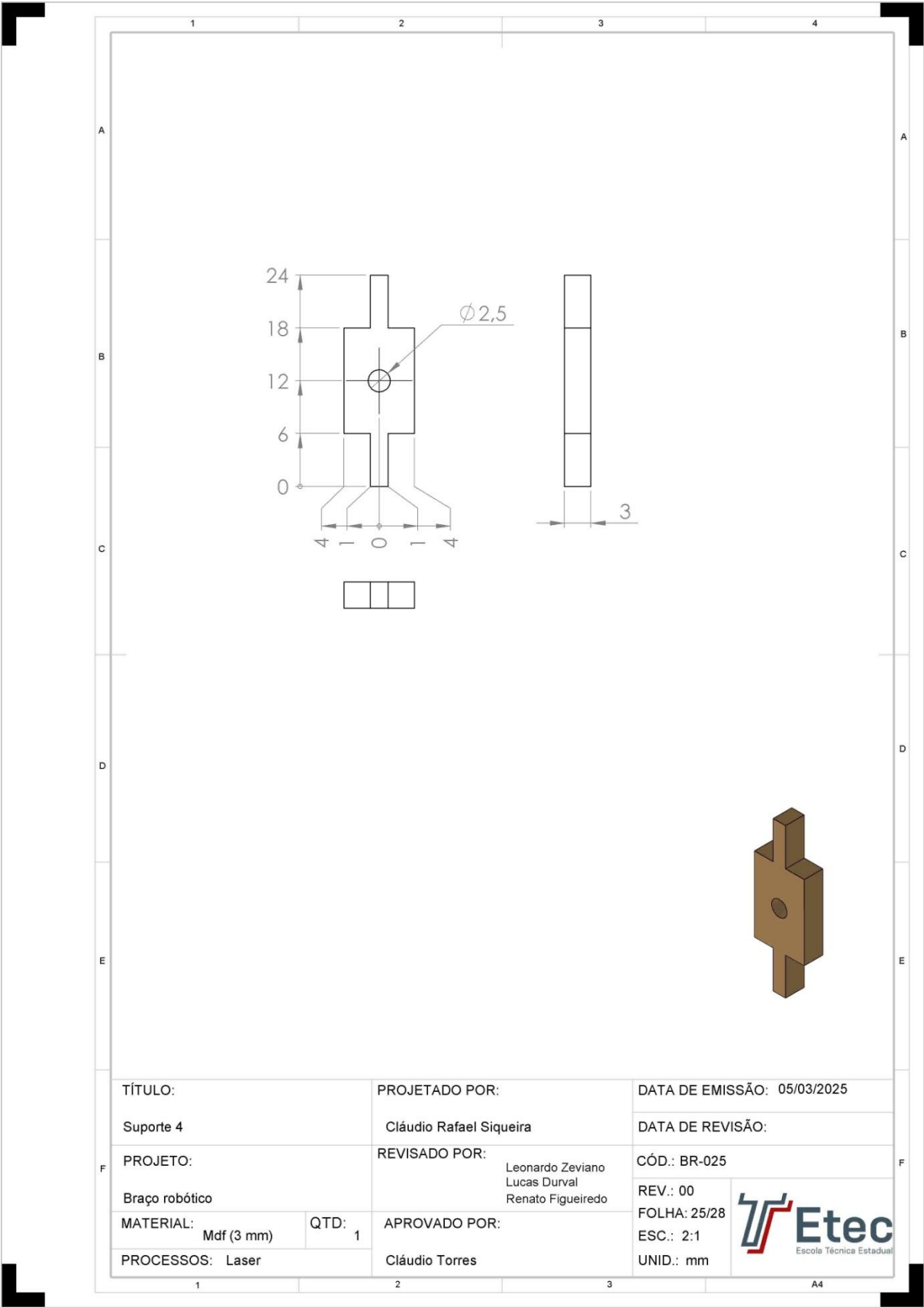


Figura 32 – Suporte V

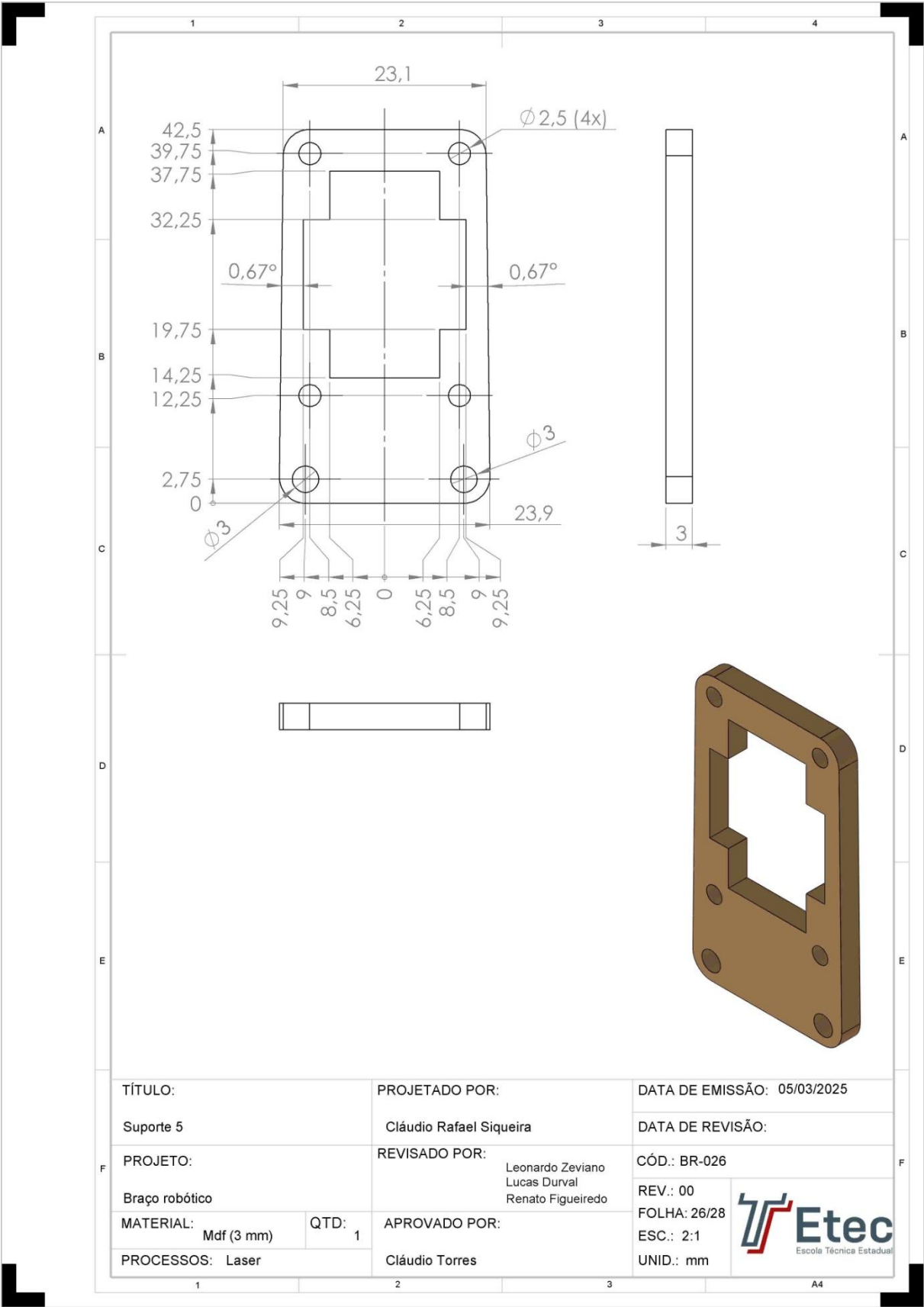


Figura 33 – Suporte VI

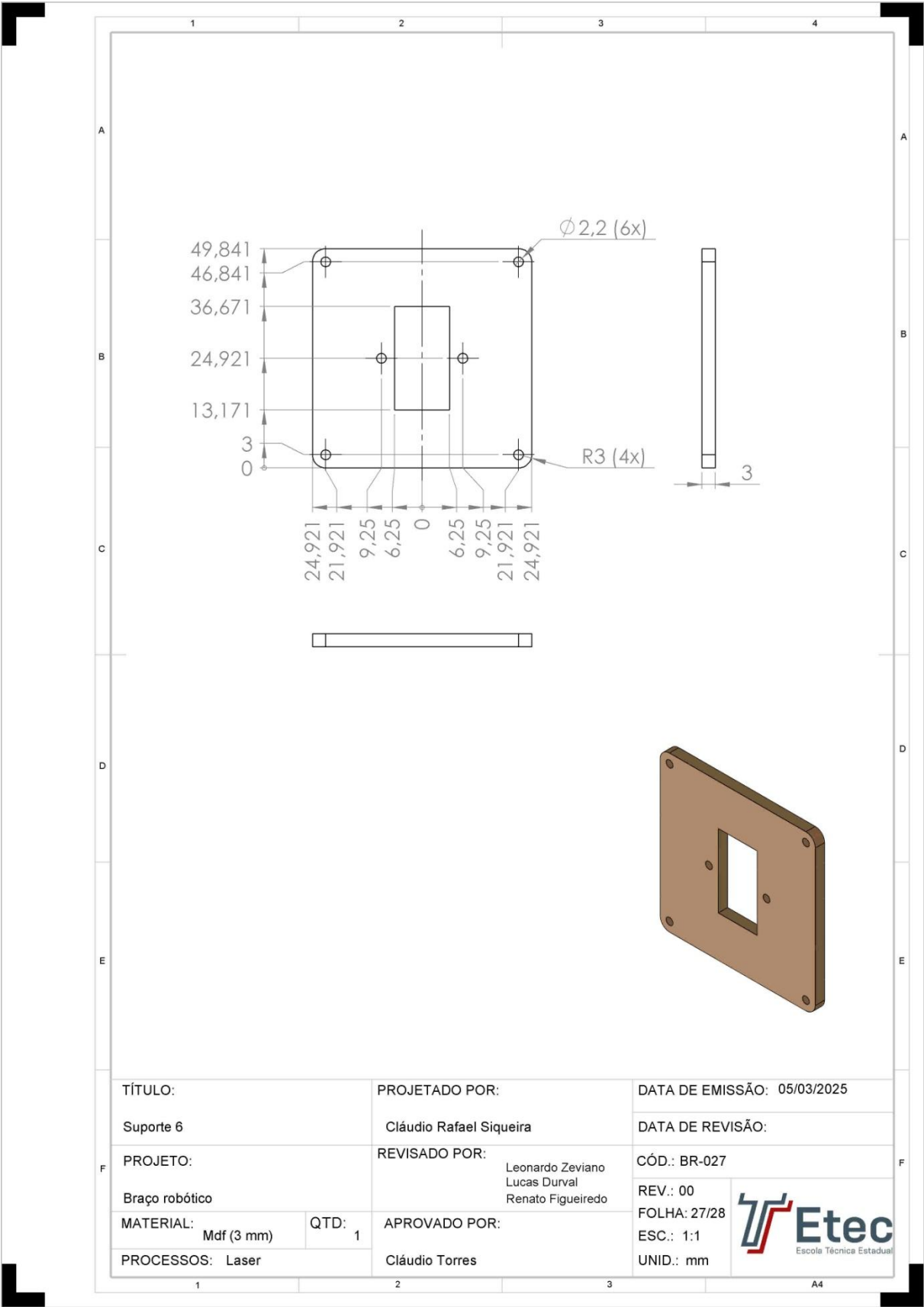
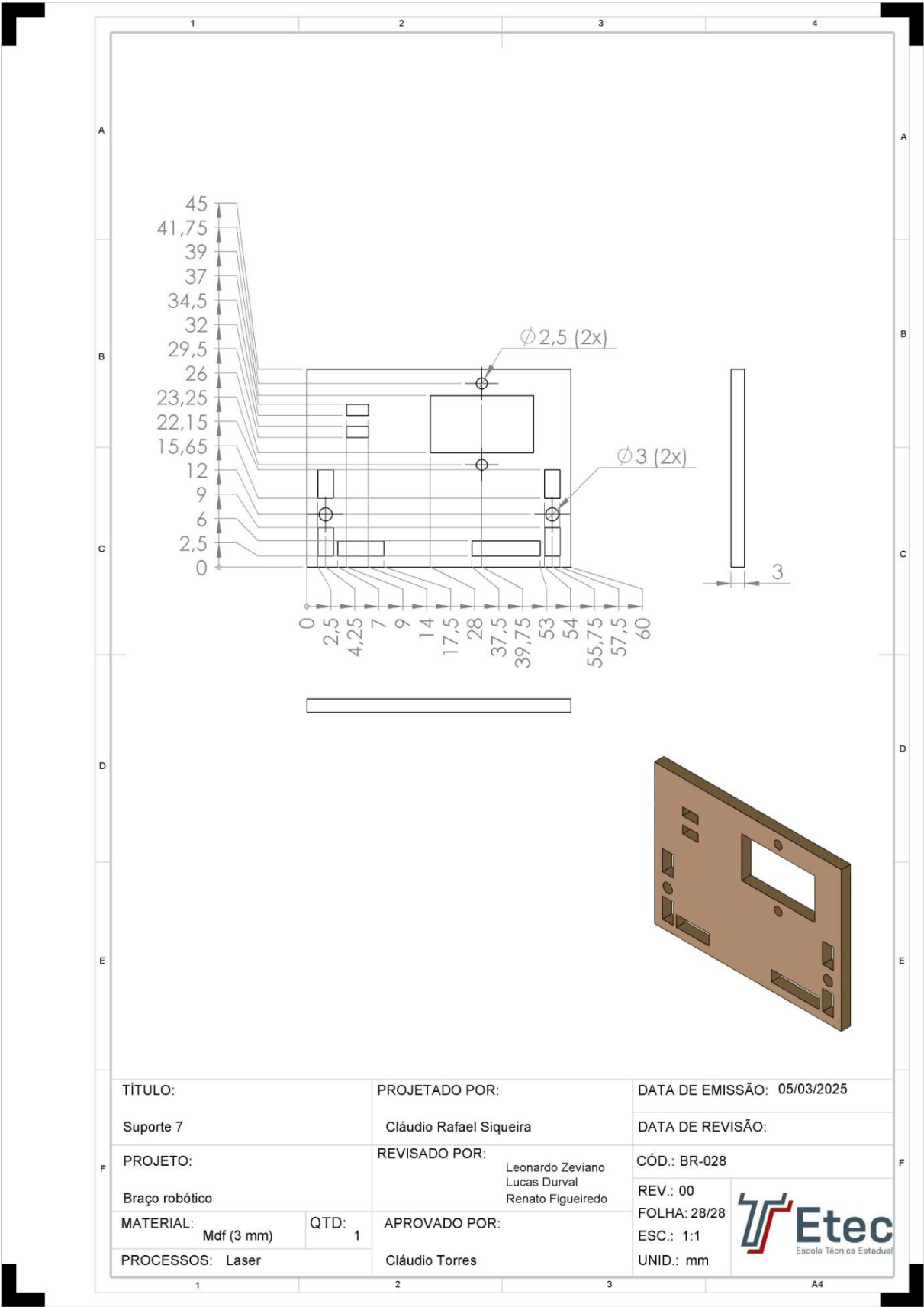


Figura 34 – Suporte VII



3. 2 Eletrônico

Os joysticks são conectados às portas analógicas do Arduino e os servos às portas digitais.

No código, os dois joysticks estão organizados da seguinte forma (recomendação física: coloque o joystick que controla Base+Extensão à esquerda e o que controla Altura+Garra à direita), para ficar coerente com os controles:

Joystick 1 (esquerdo) — controla a Base (eixo X) e a Extensão / Profundidade (eixo Y).

Eixo X (Base) → A0

Eixo Y (Extensão) → A1

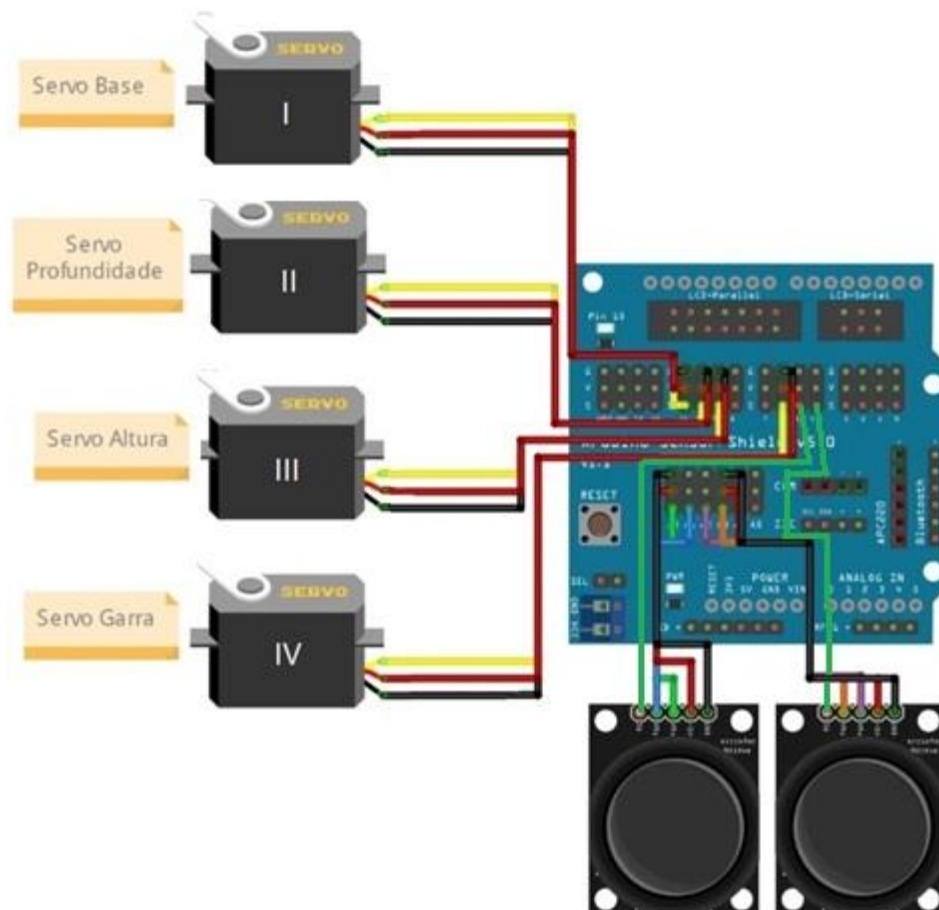
Joystick 2 (direito) — controla a Altura (eixo X) e a Garra (eixo Y).

Eixo X (Altura) → A2

Eixo Y (Garra) → A3

Os dois módulos de joystick também devem ter suas conexões de +5V e GND ligadas ao Arduino (ou ao Sensor Shield), conforme o diagrama.

Figura 35 – Projeto Eletrônico



Abaixo a descrição correta de cada pino conectado (conforme `servoX.attach(...)` no código):

Servo Base (servo1) = D11

Servo Extensão / Profundidade (servo2) = D10

Servo Altura (servo3) = D9

Servo Garra (servo4) = D6

Botões (SW) dos joysticks (para alternar modo automático/manual):

SW1 = D4

SW2 = D5

No Sensor Shield, os fios vermelhos (+VCC) dos joysticks devem ser ligados aos pinos V e os fios pretos/marrons (GND) aos pinos G correspondentes.

Observação sobre orientações físicas (visão por trás do braço):

O servo da Altura é o ligado em D9 (servo3 no código).

O servo da Profundidade/Extensão é o ligado em D10 (servo2 no código).

Alimentação:

Para a alimentação de todo o conjunto do Braço Robótico, usamos uma fonte de alimentação de 12V com a corrente mínima de 1 A.

3.3 Programação Digital

```
/* =====
```

PROJETO: BRAÇO ROBÓTICO CONTROLADO POR ARDUINO

DESCRIÇÃO:

- O braço possui 4 servo motores:

Servo 1 → Base (rotação lateral)

Servo 2 → Extensão (movimento para frente e trás)

Servo 3 → Altura (sobe e desce)

Servo 4 → Garra (abre e fecha)

- Possui dois modos de operação:

1. Modo Manual → Controlado pelos dois joysticks analógicos.

2. Modo Automático → Executa uma sequência programada de movimentos.

- A mudança de modo ocorre ao pressionar simultaneamente os dois botões (SW1 e SW2).

```
===== */
```

```
#include "VarSpeedServo.h" // Biblioteca que permite controlar a velocidade dos servos
```

```
/* -----
```

DECLARAÇÃO DOS OBJETOS DOS SERVOS

```
----- */
```

```
VarSpeedServo servo1; // Servo da base
```

```
VarSpeedServo servo2; // Servo da extensão do braço
```

```
VarSpeedServo servo3; // Servo da altura
```

```
VarSpeedServo servo4; // Servo da garra
```

```
/* -----
```

PINOS DOS JOYSTICKS E BOTÕES

```
----- */
```

```
int potpin1 = A0; // Eixo X do joystick 1 (base)
```

```
int potpin2 = A1; // Eixo Y do joystick 1 (extensão)
```

```
int potpin3 = A2; // Eixo X do joystick 2 (altura)
```

```
int potpin4 = A3; // Eixo Y do joystick 2 (garra)
```

```
int sw1 = 4; // Botão do joystick 1
```

```
int sw2 = 5; // Botão do joystick 2
```

```
/* -----
```

LIMITES DE MOVIMENTO DE CADA SERVO (em graus)

```
----- */
```

```

int pos4min = 25, pos4max = 60; // Limites da garra
int pos3min = 50, pos3max = 178; // Limites da altura
int pos2min = 2, pos2max = 130; // Limites da extensão
int pos1min = 5, pos1max = 175; // Limites da base

/* -----
   VARIÁVEIS DE POSIÇÃO ATUAL DOS SERVOS
   ----- */

float pos1 = 90.0; // Base (float para permitir incrementos suaves)
int pos2 = 50;    // Extensão
int pos3 = 95;    // Altura
int pos4 = 48;    // Garra

/* -----
   CONTROLE DE MODOS E ESTADOS
   ----- */

bool modoAutomatico = false; // Inicia no modo manual
bool botaoAnterior = false; // Armazena estado anterior dos botões

/* -----
   CONTROLE DE TEMPO (para suavizar o movimento)
   ----- */

unsigned long ultimoMovBase = 0;    // Tempo do último movimento da base
const unsigned long intervaloBase = 80; // Intervalo entre movimentos da base (ms)
unsigned long ultimoMovOutros = 0;   // Tempo dos outros servos
const unsigned long intervaloOutros = 20;

/* -----
   CONTROLE DA FREQUÊNCIA DE IMPRESSÃO SERIAL

```

```

----- */

unsigned long ultimoPrint = 0;

const unsigned long intervaloPrint = 300; // Atualiza as posições a cada 300ms

/* =====

FUNÇÃO SETUP()

Executa apenas uma vez quando o Arduino é ligado.

===== */

void setup() {
  Serial.begin(9600); // Inicia a comunicação serial

  pinMode(sw1, INPUT_PULLUP); // Configura botão 1 com resistor interno
  pinMode(sw2, INPUT_PULLUP); // Configura botão 2 com resistor interno

  // Define os pinos de controle dos servos
  servo1.attach(11);
  servo2.attach(10);
  servo3.attach(9);
  servo4.attach(6);

  // Posiciona os servos na posição inicial
  servo1.write((int)round(pos1));
  servo2.write(pos2);
  servo3.write(pos3);
  servo4.write(pos4);

  delay(800); // Pausa inicial para estabilizar os servos
}

/* =====

```

FUNÇÃO LOOP()

Executa continuamente enquanto o Arduino estiver ligado.

```

===== */
void loop() {
    // Verifica se ambos os botões estão pressionados ao mesmo tempo
    bool swPressionado = (digitalRead(sw1) == LOW && digitalRead(sw2) == LOW);

    // Detecta a troca de modo apenas quando o estado muda (debounce)
    if (swPressionado && !botaoAnterior) {
        delay(200); // Evita leituras falsas por ruído
        if (digitalRead(sw1) == LOW && digitalRead(sw2) == LOW) {

            // Alterna entre manual ↔ automático
            modoAutomatico = !modoAutomatico;

            // Se entrou no modo automático:
            if (modoAutomatico) {
                moverParaInicial(); // Move o braço para posição neutra
                while (modoAutomatico) { // Mantém no modo automático até sair
                    executarSequencia();
                }
            }

            // Espera até os botões serem soltos
            while (digitalRead(sw1) == LOW || digitalRead(sw2) == LOW);
            delay(200);
        }
    }

    botaoAnterior = swPressionado; // Atualiza estado anterior dos botões
}

```

```

// Se estiver no modo manual, executa o controle pelos joysticks
if (!modoAutomatico) {
    controleManual();
    imprimirPosicoes(); // Mostra as posições no monitor serial
}
}

/* =====
FUNÇÃO: controleManual()
Controla os servos de acordo com os valores dos joysticks.
===== */

void controleManual() {
    unsigned long agora = millis(); // Tempo atual

    // Lê os valores dos 4 eixos analógicos dos joysticks
    int leituraBase = analogRead(potpin1);
    int leituraOut1 = analogRead(potpin2);
    int leituraOut2 = analogRead(potpin3);
    int leituraOut3 = analogRead(potpin4);

    const int centro = 512; // Valor central do joystick
    const int zonaMorta = 60; // Margem para evitar vibrações no centro

    /* ----- CONTROLE DA BASE ----- */
    if (agora - ultimoMovBase >= intervaloBase) {
        if (leituraBase > centro + zonaMorta) {
            pos1 += 1;
            if (pos1 > pos1max) pos1 = pos1max;

```

```

servo1.write((int)round(pos1));
} else if (leituraBase < centro - zonaMorta) {
    pos1 -= 1;
    if (pos1 < pos1min) pos1 = pos1min;
    servo1.write((int)round(pos1));
}
ultimoMovBase = agora;
}

/* ----- CONTROLE DOS DEMAIS SERVOS ----- */

if (agora - ultimoMovOutros >= intervaloOutros) {

    // EXTENSÃO (movimento invertido)
    if (leituraOut1 > centro + zonaMorta) {
        pos2--;
        if (pos2 < pos2min) pos2 = pos2min;
        servo2.write(pos2);
    } else if (leituraOut1 < centro - zonaMorta) {
        pos2++;
        if (pos2 > pos2max) pos2 = pos2max;
        servo2.write(pos2);
    }

    // ALTURA (para cima e para baixo)
    if (leituraOut2 > centro + zonaMorta) {
        pos3++;
        if (pos3 > pos3max) pos3 = pos3max;
        servo3.write(pos3);
    } else if (leituraOut2 < centro - zonaMorta) {

```

```

    pos3--;
    if (pos3 < pos3min) pos3 = pos3min;
    servo3.write(pos3);
}

// GARRA (abrir e fechar)
if (leituraOut3 > centro + zonaMorta) {
    pos4--;
    if (pos4 < pos4min) pos4 = pos4min;
    servo4.write(pos4);
} else if (leituraOut3 < centro - zonaMorta) {
    pos4++;
    if (pos4 > pos4max) pos4 = pos4max;
    servo4.write(pos4);
}

ultimoMovOutros = agora; // Atualiza tempo do último movimento
}
}

/* =====
FUNÇÃO: imprimirPosicoes()
Exibe no Monitor Serial as posições atuais de cada servo.
===== */

void imprimirPosicoes() {
    unsigned long agora = millis();
    if (agora - ultimoPrint >= intervaloPrint) {
        Serial.print("Base: "); Serial.print(pos1);
        Serial.print(" | Extensao: "); Serial.print(pos2);
    }
}

```

```

    Serial.print(" | Altura: "); Serial.print(pos3);

    Serial.print(" | Garra: "); Serial.println(pos4);

    ultimoPrint = agora;
}
}

/* =====
FUNÇÃO: moverParaInicial()
Retorna o braço robótico à posição inicial.
===== */

void moverParaInicial() {
    servo1.slowmove(90, 20);
    servo2.slowmove(50, 20);
    servo3.slowmove(95, 20);
    servo4.slowmove(48, 20);

    pos1 = 90.0; pos2 = 50; pos3 = 95; pos4 = 48;
    delay(800);
}

/* =====
FUNÇÃO: checarSairAutomatico()
Permite sair do modo automático pressionando os dois botões.
===== */

bool checarSairAutomatico() {
    if (digitalRead(sw1) == LOW && digitalRead(sw2) == LOW) {
        delay(200);
        if (digitalRead(sw1) == LOW && digitalRead(sw2) == LOW) {
            while (digitalRead(sw1) == LOW || digitalRead(sw2) == LOW);
        }
    }
}

```

```

    delay(200);

    modoAutomatico = false;

    return true;

}

}

return false;

}

/* =====

FUNÇÃO: executarSequencia()

Executa a sequência automática para pegar as 4 peças:

- Rosa

- Verde

- Branca

- Azul

===== */

void executarSequencia() {

/* ----- PEÇA ROSA ----- */

servo2.slowmove(17, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(41, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(52, 18); if (checarSairAutomatico()) return; delay(1000);
servo4.slowmove(31, 18); if (checarSairAutomatico()) return; delay(1000);
servo3.slowmove(169, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(19, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(156, 20); if (checarSairAutomatico()) return; delay(2000);
servo4.slowmove(58, 18); if (checarSairAutomatico()) return; delay(1000);
servo4.slowmove(31, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(90, 18); if (checarSairAutomatico()) return; delay(1000);

```

```
moverParaInicial();
```

```
/* ----- PEÇA VERDE ----- */
```

```
servo4.slowmove(58, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(17, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(27, 20); if (checarSairAutomatico()) return; delay(1000);
servo3.slowmove(115, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(75, 18); if (checarSairAutomatico()) return; delay(1000);
servo4.slowmove(31, 18); if (checarSairAutomatico()) return; delay(1000);
servo3.slowmove(160, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(19, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(156, 20); if (checarSairAutomatico()) return; delay(2000);
servo4.slowmove(58, 18); if (checarSairAutomatico()) return; delay(1000);
servo4.slowmove(31, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(90, 18); if (checarSairAutomatico()) return; delay(1000);
moverParaInicial();
```

```
/* ----- PEÇA BRANCA ----- */
```

```
servo2.slowmove(17, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(55, 20); if (checarSairAutomatico()) return; delay(1000);
servo3.slowmove(117, 18); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(74, 18); if (checarSairAutomatico()) return; delay(1000);
servo4.slowmove(31, 18); if (checarSairAutomatico()) return; delay(1000);
servo3.slowmove(169, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(19, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(156, 20); if (checarSairAutomatico()) return; delay(2000);
servo4.slowmove(58, 18); if (checarSairAutomatico()) return; delay(1000);
servo4.slowmove(31, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(90, 18); if (checarSairAutomatico()) return; delay(1000);
```

```

moverParaInicial();

/* ----- PEÇA AZUL ----- */

servo4.slowmove(58, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(17, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(43, 20); if (checarSairAutomatico()) return; delay(1000);
servo3.slowmove(146, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(99, 18); if (checarSairAutomatico()) return; delay(1000);
servo4.slowmove(31, 18); if (checarSairAutomatico()) return; delay(1000);
servo3.slowmove(160, 20); if (checarSairAutomatico()) return; delay(1000);
servo2.slowmove(19, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(156, 20); if (checarSairAutomatico()) return; delay(2000);
servo4.slowmove(58, 18); if (checarSairAutomatico()) return; delay(1000);
servo4.slowmove(31, 20); if (checarSairAutomatico()) return; delay(1000);
servo1.slowmove(90, 18); if (checarSairAutomatico()) return; delay(1000);
moverParaInicial();
}

```

4. PROCESSO DE FABRICAÇÃO

4.1 Seleção de Materiais

Escolhemos o MDF devido à facilidade de corte. Optamos por um material com baixo custo, baixo peso para não sobrecarregar os servos e baixa espessura (3 milímetros) para ser facilmente cortado.

4.2 Corte das Peças

Precisão garantida pelo corte a laser.

4.3 Montagem do Braço Robótico

4.3.1 Montagem da Base com os apoios (pés)

Figura 36 – Base - Peças

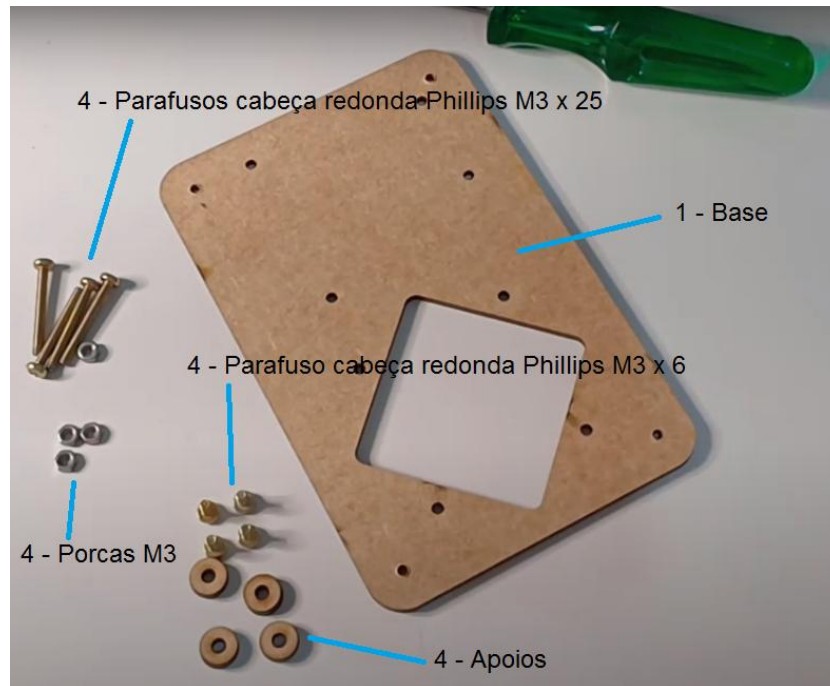


Figura 37 – Base - Montagem dos pés

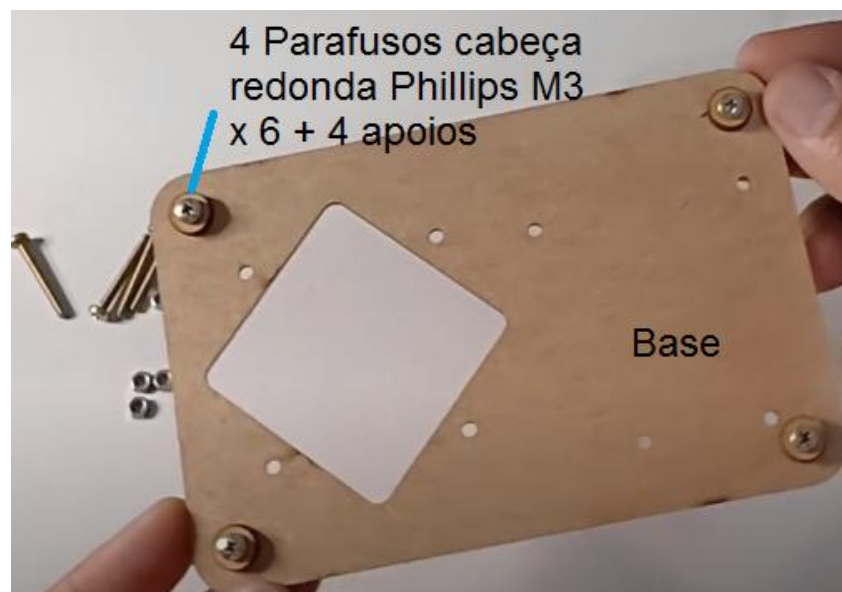
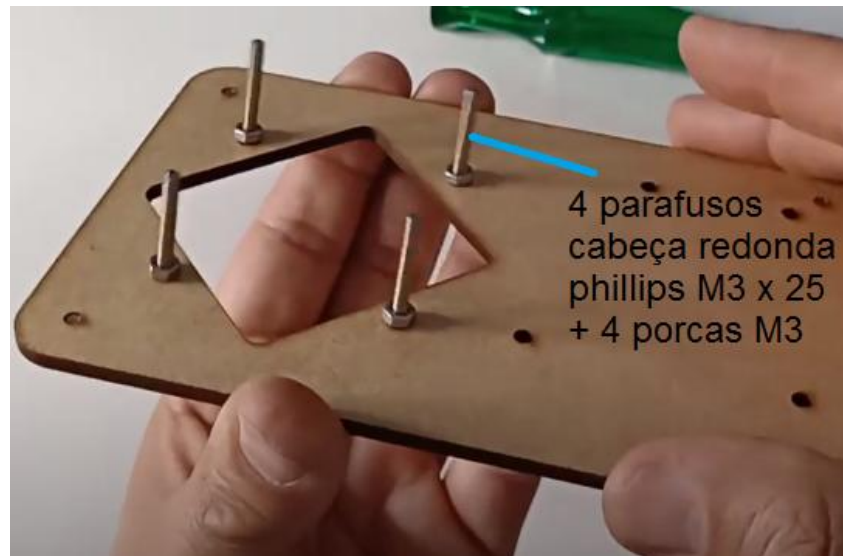


Figura 38 – Base - Montagem dos parafusos centrais



4.3.2 Montagem do motor I

Figura 39 – Motor I - Peças

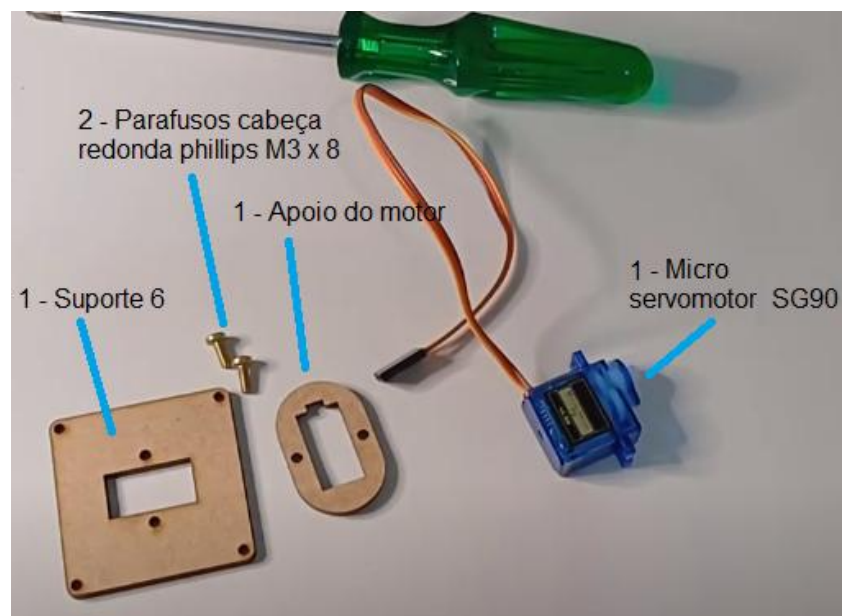
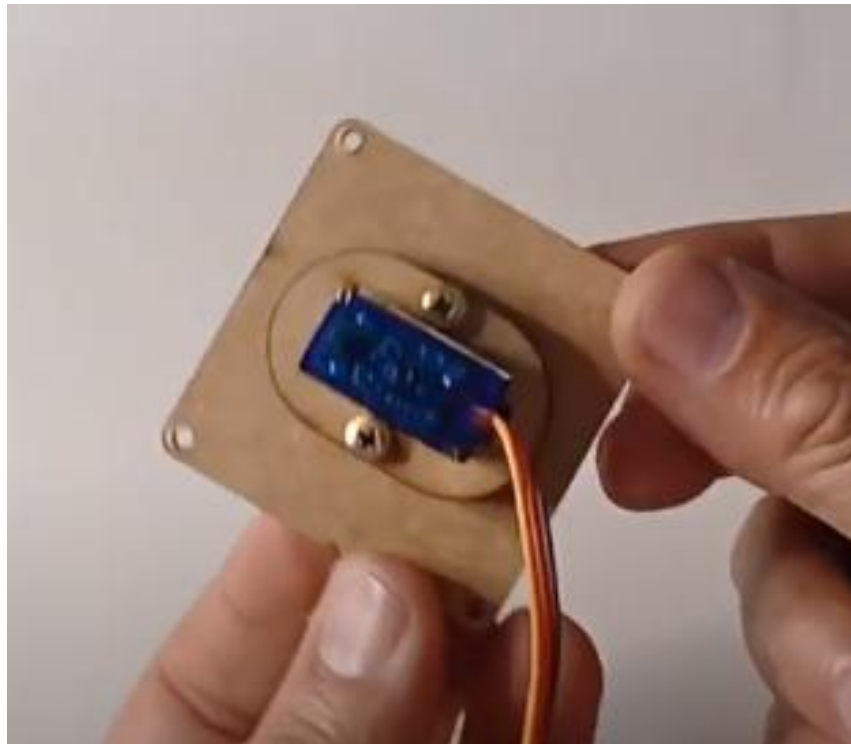


Figura 40 – Motor I - União das peças

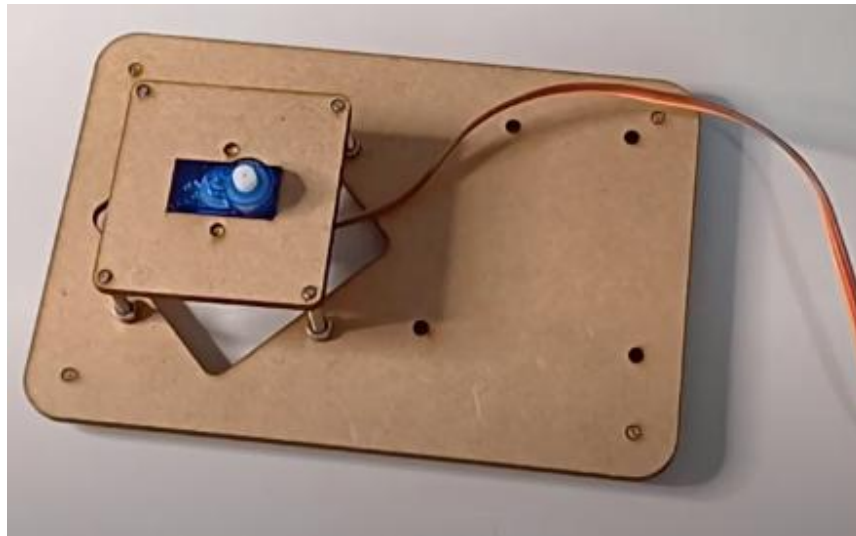


4.3.3 Montagem da base com a montagem do motor I

Figura 41 – Base e motor I



Figura 42 – Base e motor I montados



4.3.4 Montagem do apoio da base na montagem 4.3.3

Figura 43 – Apoio da base



4.3.5 Montagem do motor IV

Figura 44 – Motor IV - Peças

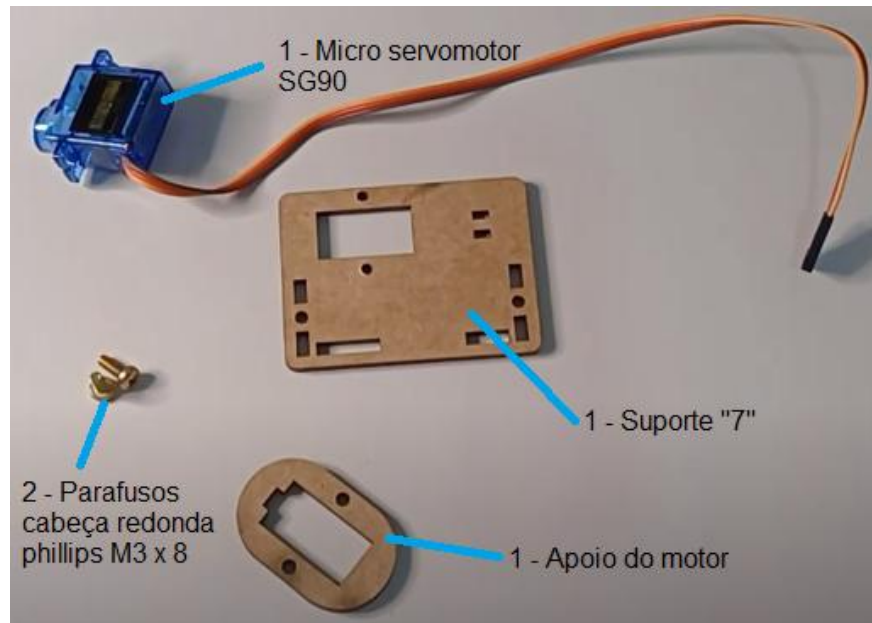
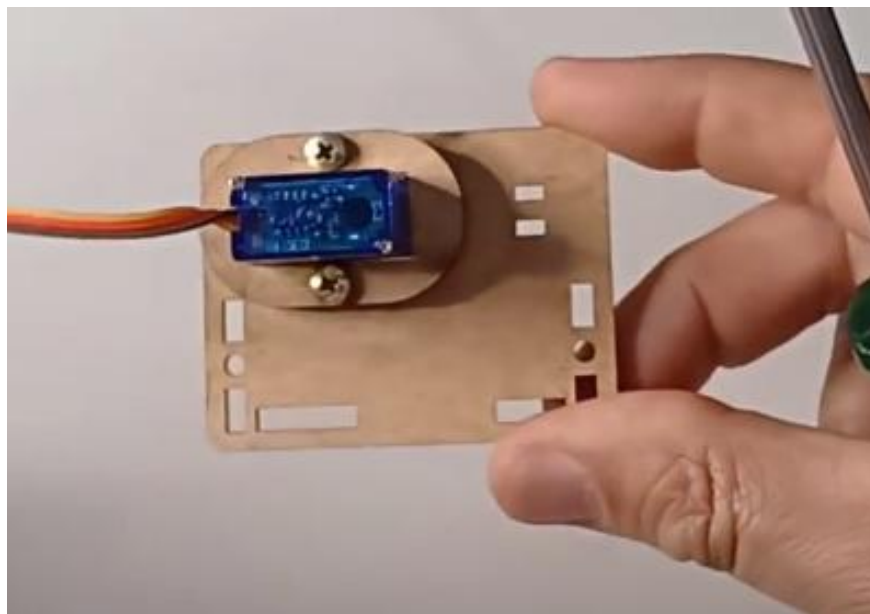


Figura 45 – Motor IV - União das peças



4.3.6 Montagem da articulação “1” na montagem do motor IV

Figura 46 – Articulação I e Motor IV



Figura 47 – Articulação I e Motor IV - União das peças



4.3.7 Montagem do motor III

Figura 48 – Motor III - Peças

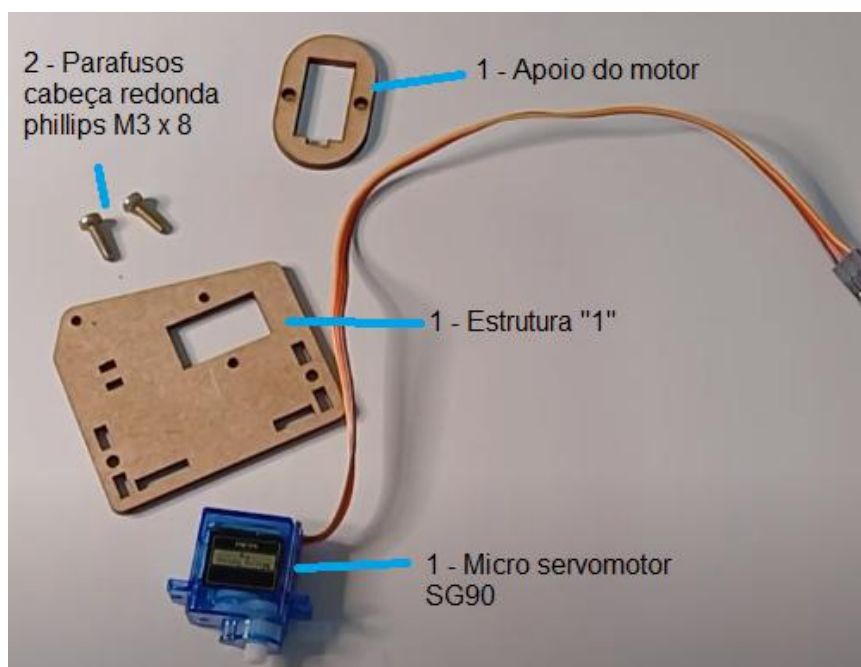
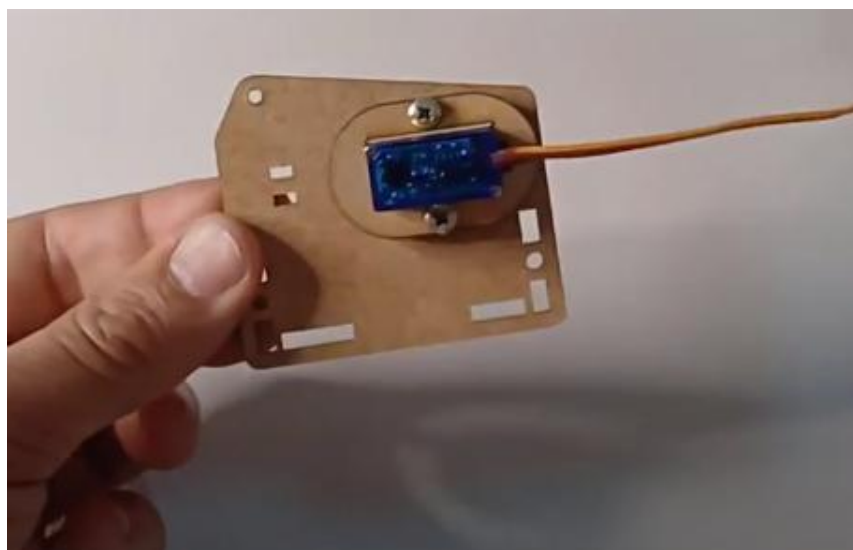


Figura 49 – Motor III - União das peças



4.3.8 Montagem da articulação “6” na montagem do motor III

Figura 50 – Articulação VI e Motor III

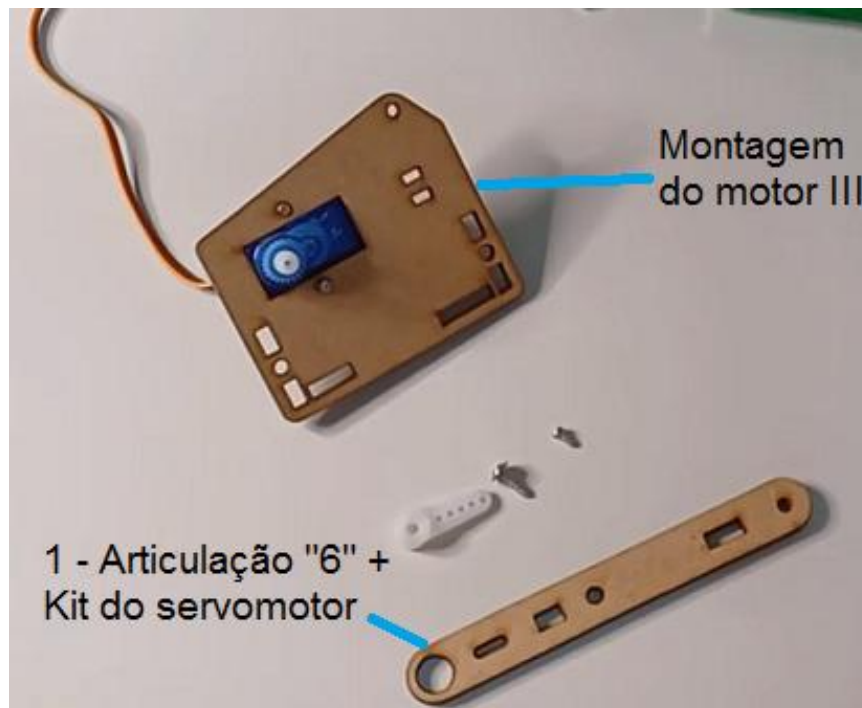


Figura 51 – Articulação VI e Motor III - União das peças



4.3.9 Montagem da estrutura "2"

Figura 52 – Estrutura II e Kit do Servo - Peças

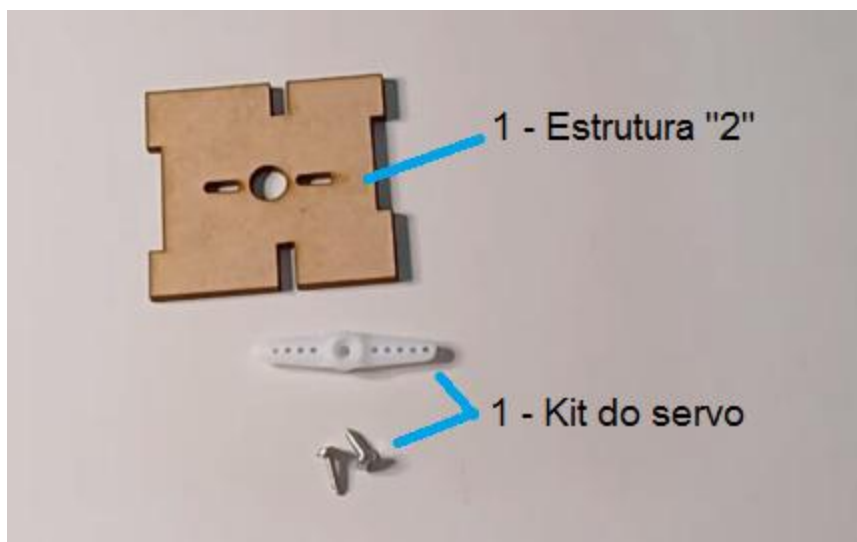
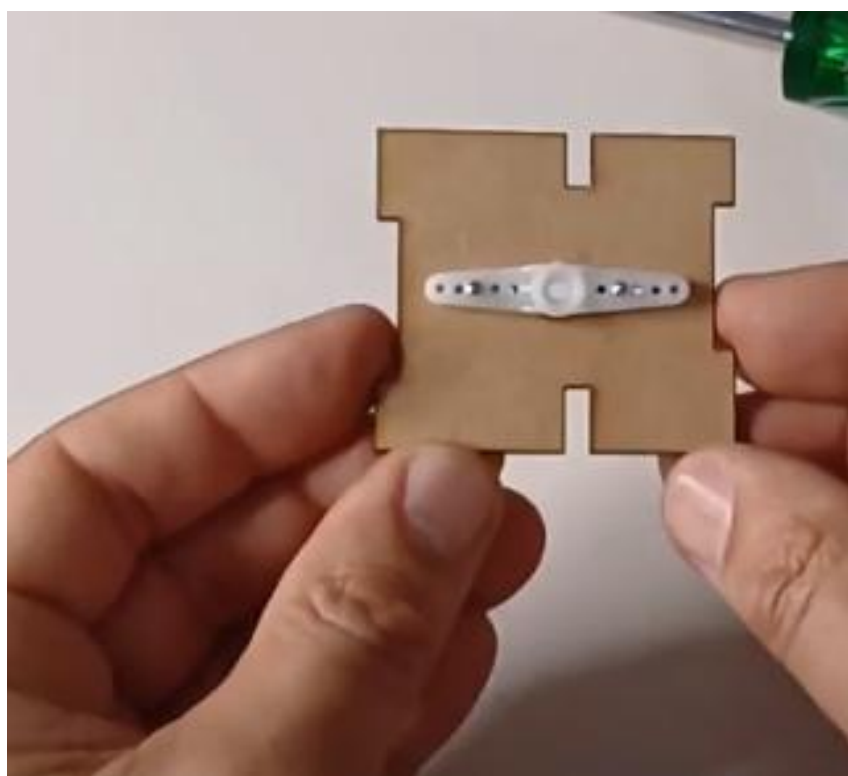


Figura 53 – Estrutura II e Kit do Servo - União das peças



4.3.10 Montagem da estrutura "3" com articulação "4"

Figura 54 – Estrutura III e Articulação IV - Peças

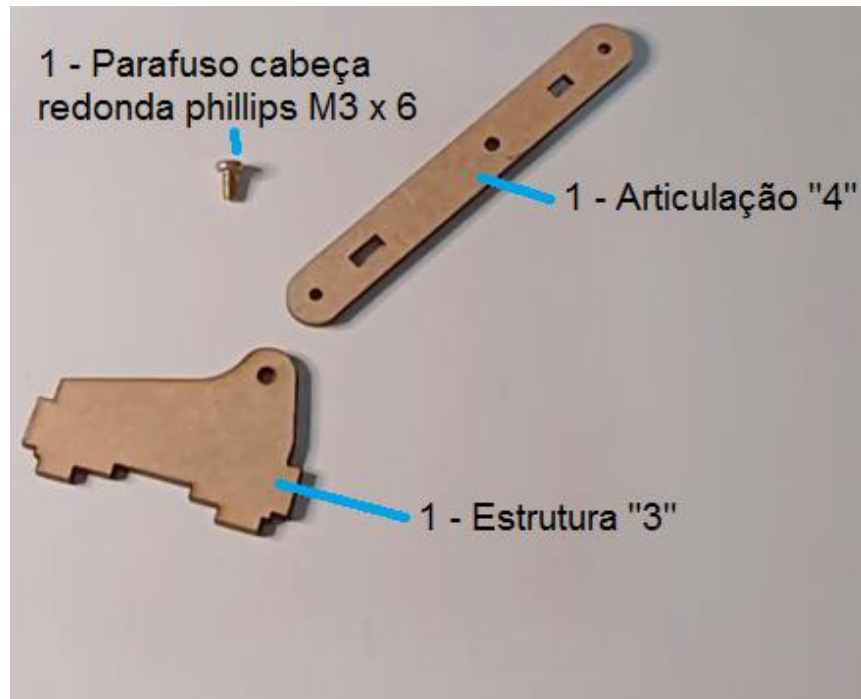


Figura 55 – Estrutura III e Articulação IV - União das peças



4.3.11 Montagem das estruturas "2" com as demais montagens

Figura 56 – Montagem do Motor III, Suportes II, Montagem do Motor IV, Montagem da Estrutura III e Montagem da Estrutura II

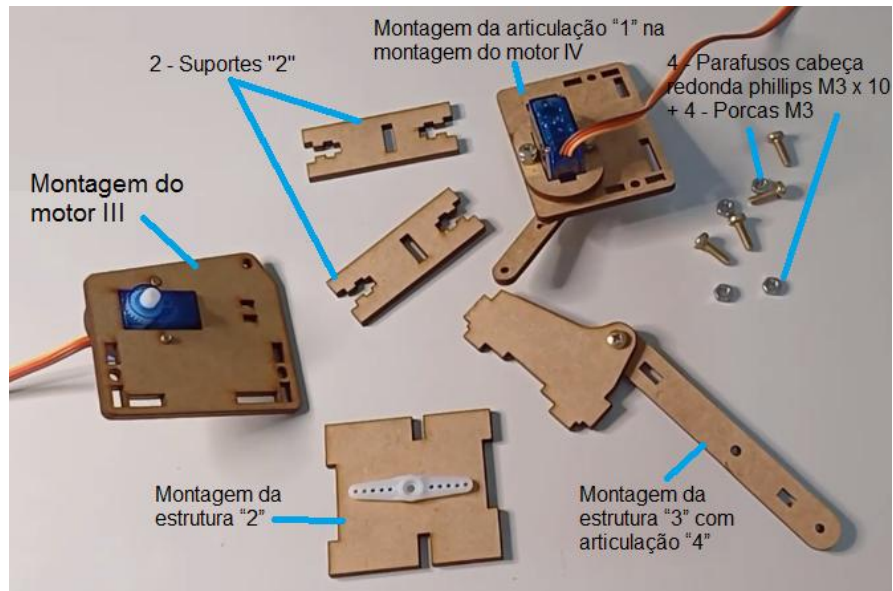


Figura 57 – Montagem do Motor III e Suporte II - Peças

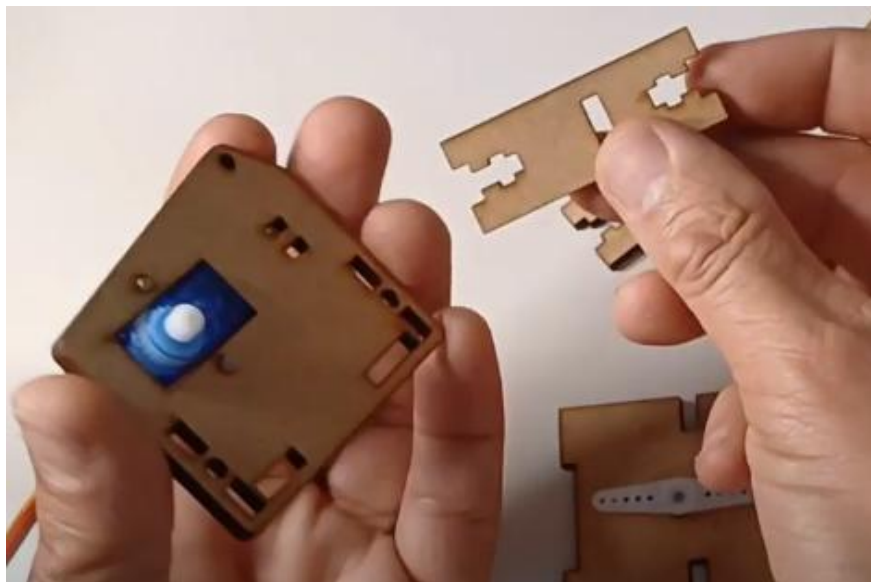


Figura 58 – Montagem do Motor III e Suporte II - Peças

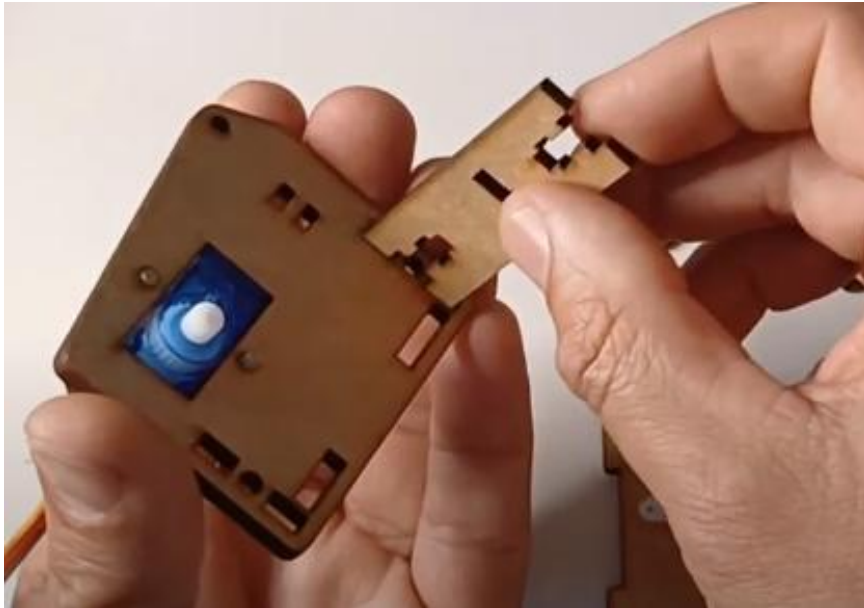


Figura 59 – Montagem do Motor III e Suporte II - União das peças

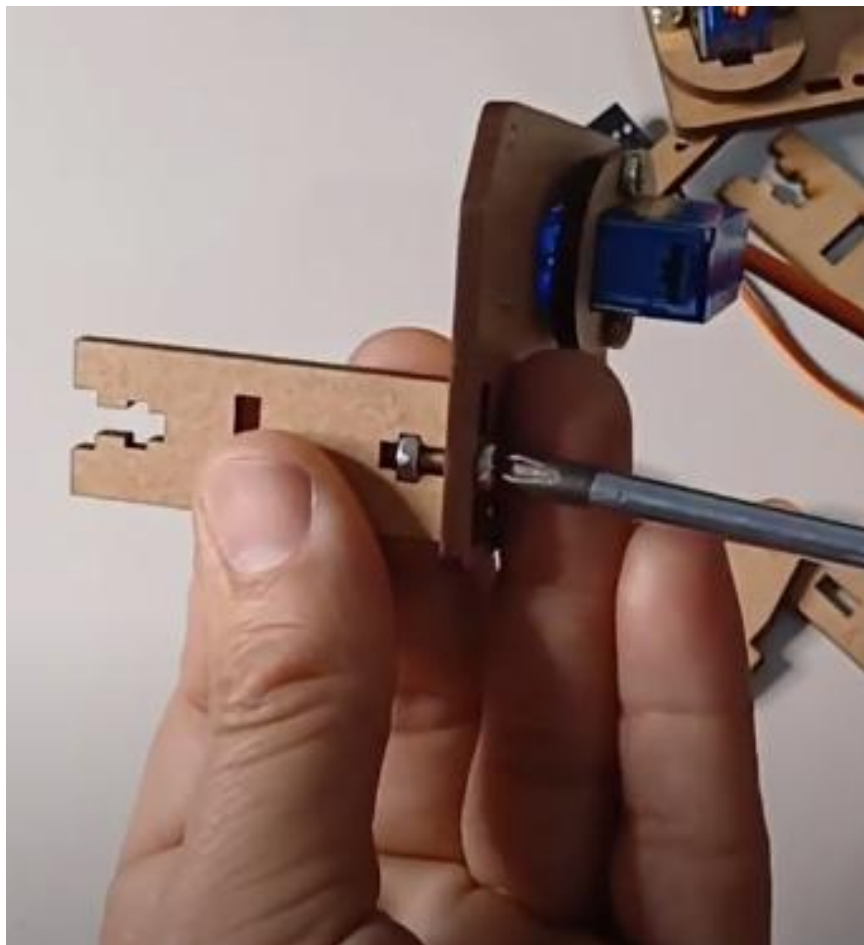


Figura 60 – Montagem do Motor III, Suporte II e Montagem da Estrutura II - União das peças

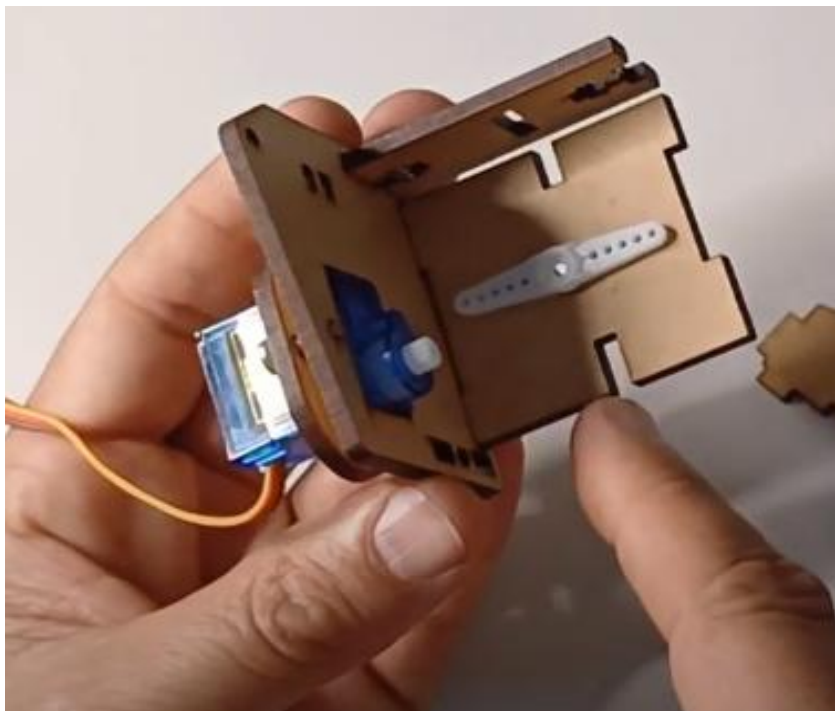


Figura 61 – Montagem do Motor III, Suporte II, Montagem da Estrutura II e Montagem da Estrutura III- União das peças

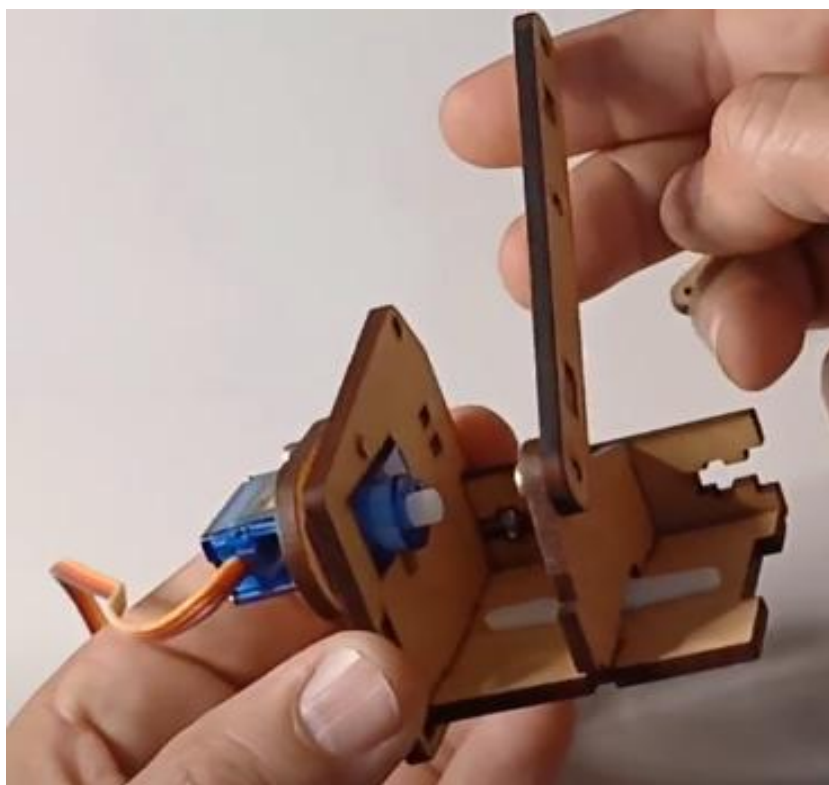


Figura 62 - União do outro Suporte II

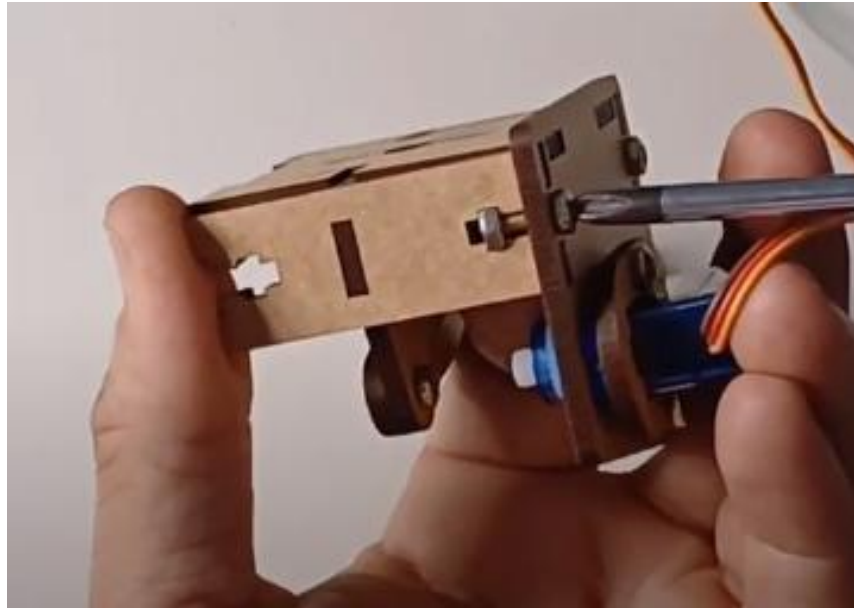


Figura 63 - Montagem em outro ângulo

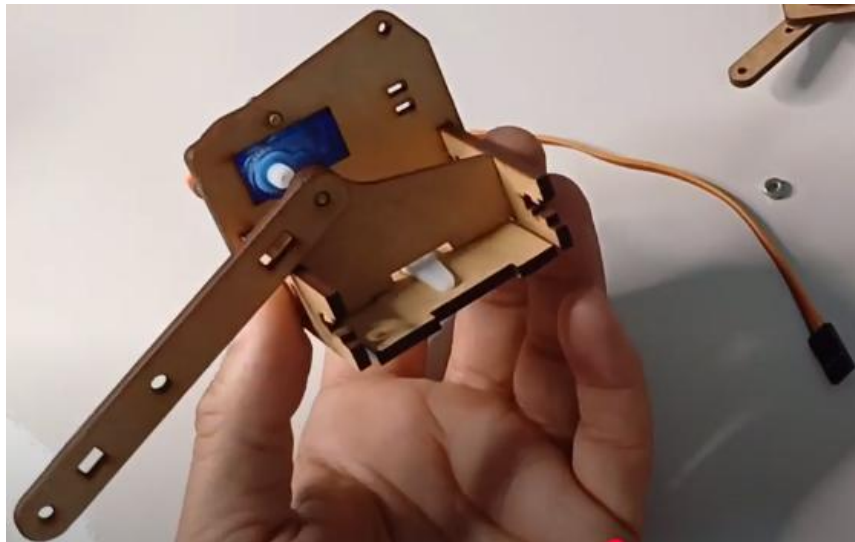


Figura 64 - Montagem de todos os conjuntos e peças apresentado na figura 56

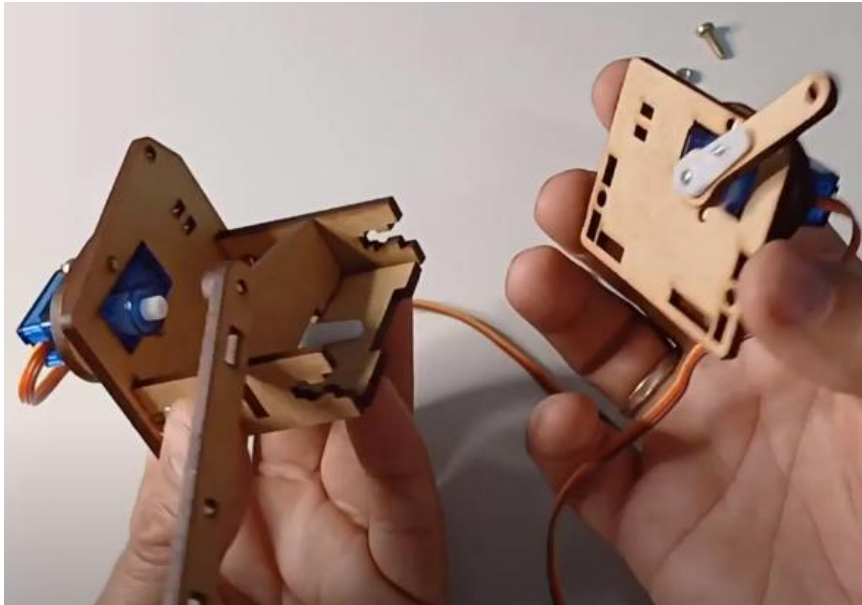
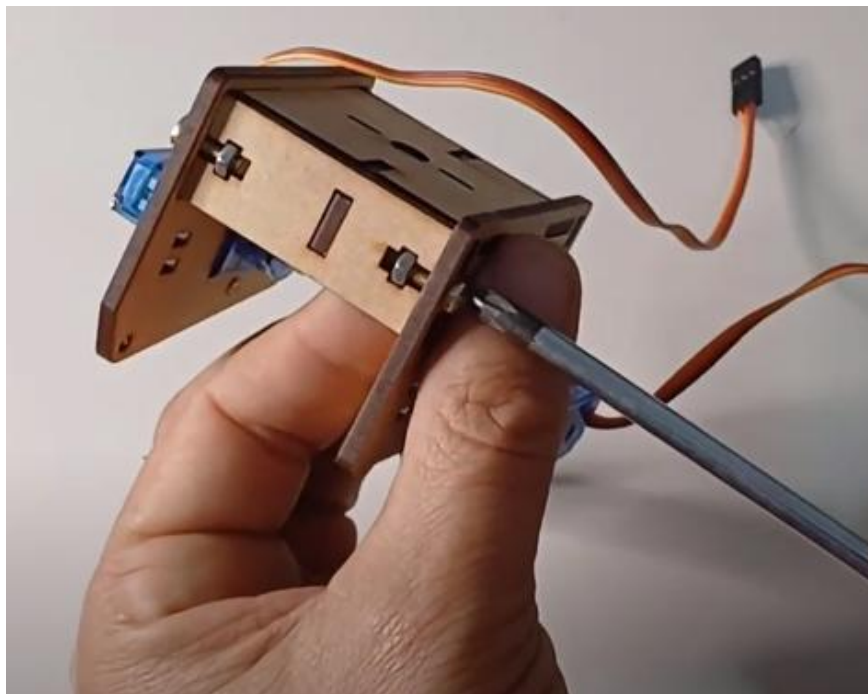


Figura 65 - Montagem de todos os conjuntos e peças apresentado na figura 56 - União



4.3.12 Montagem da estrutura com o “suporte 3”

Figura 66 - Montagem do Suporte III no conjunto

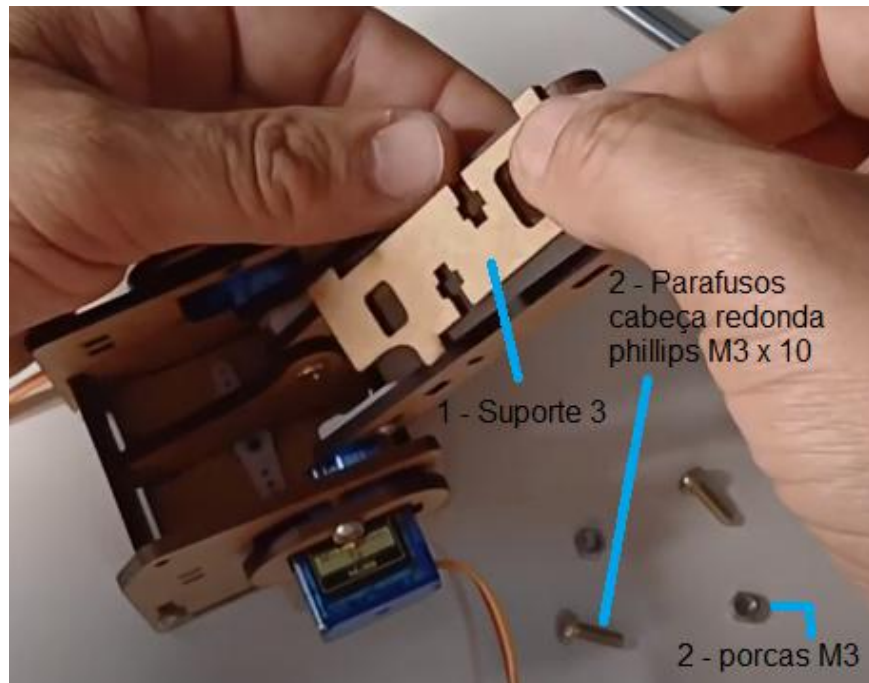


Figura 67 - Parafusando o Suporte III no conjunto

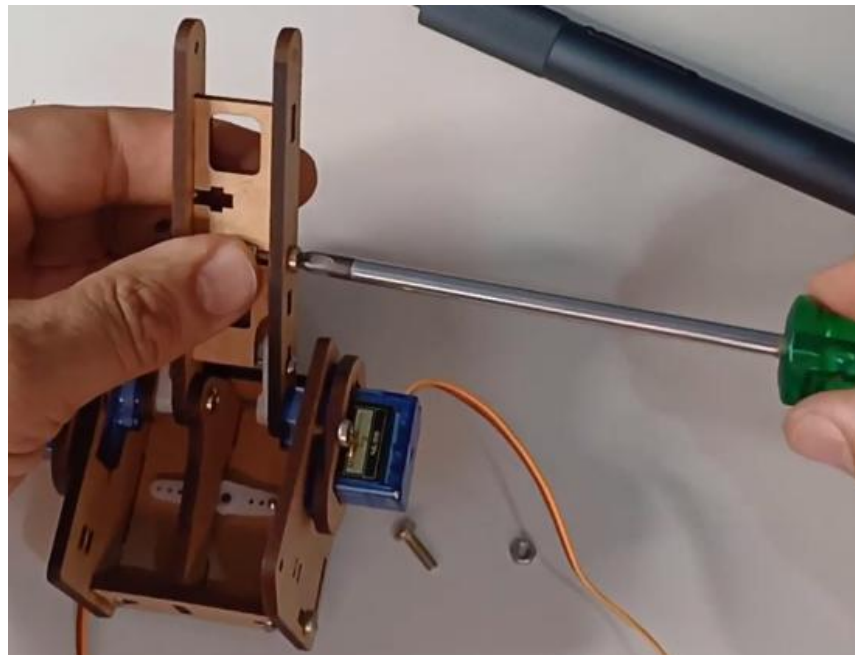


Figura 68 - Parafusando o outro lado do Suporte III no conjunto

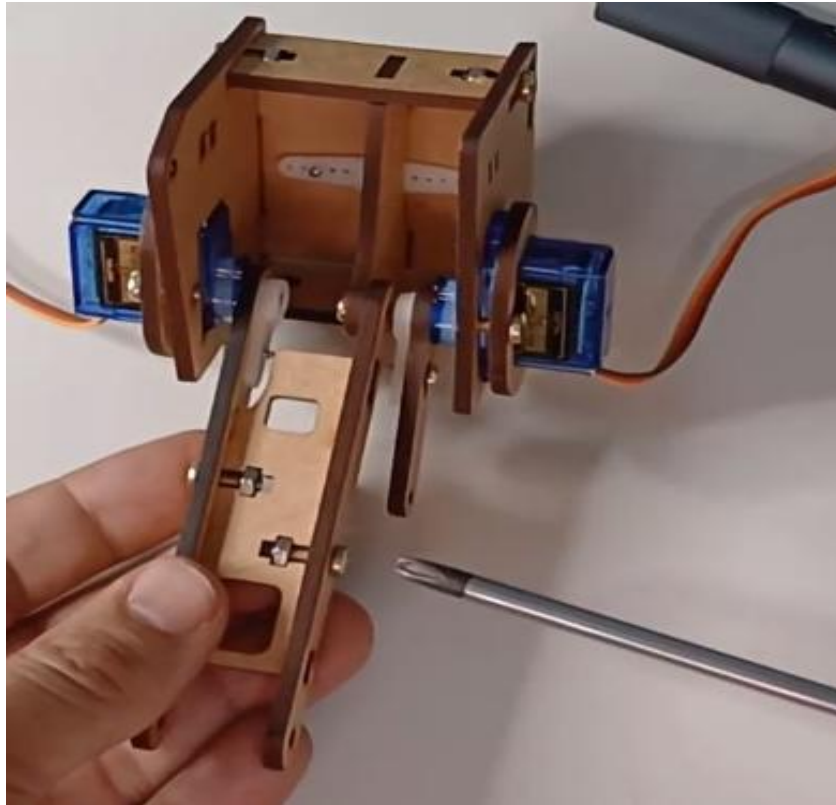


Figura 69 - Montagem do Suporte III concluída

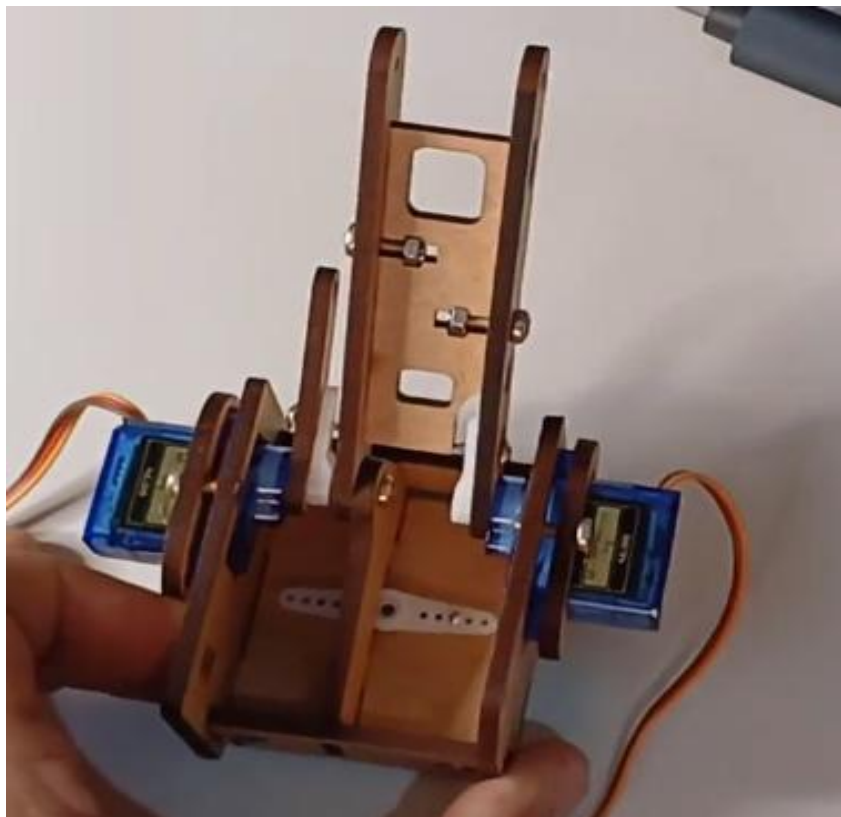
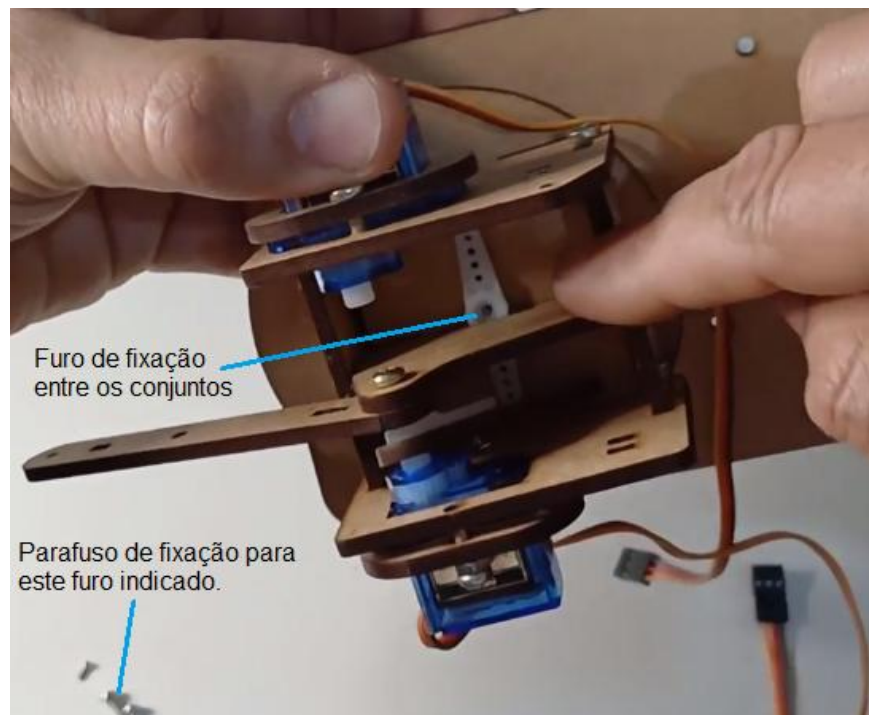


Figura 72 - Fixação dos conjuntos por parafusos



4.3.14 Montagem do conjunto com a “Articulação 3”

Figura 73 - Conjunto e Articulação III

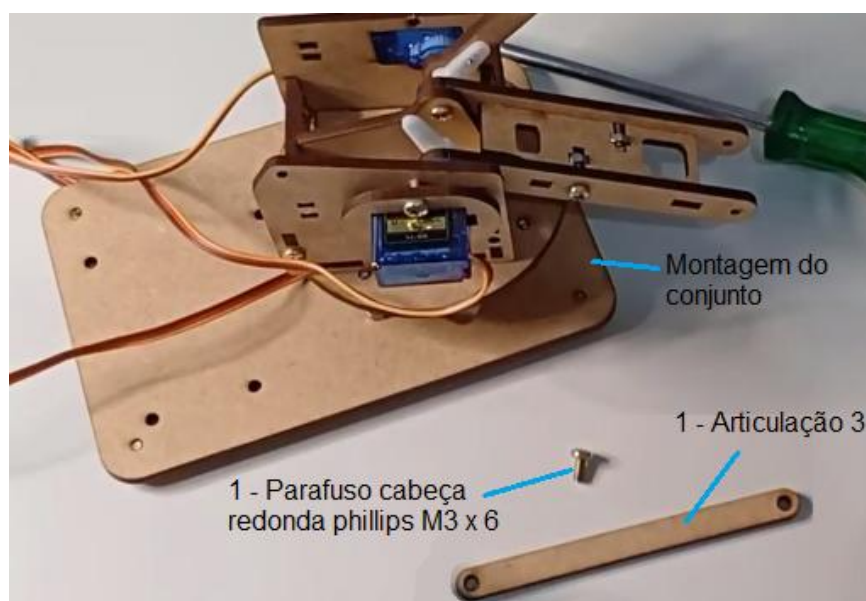


Figura 74 - Montagem da Articulação III na Articulação I do Motor IV



4.3.15 Montagem do “Fêmur” e da “Articulação 8” no conjunto

Figura 75 - Montagem geral, Articulação VIII e Fêmur

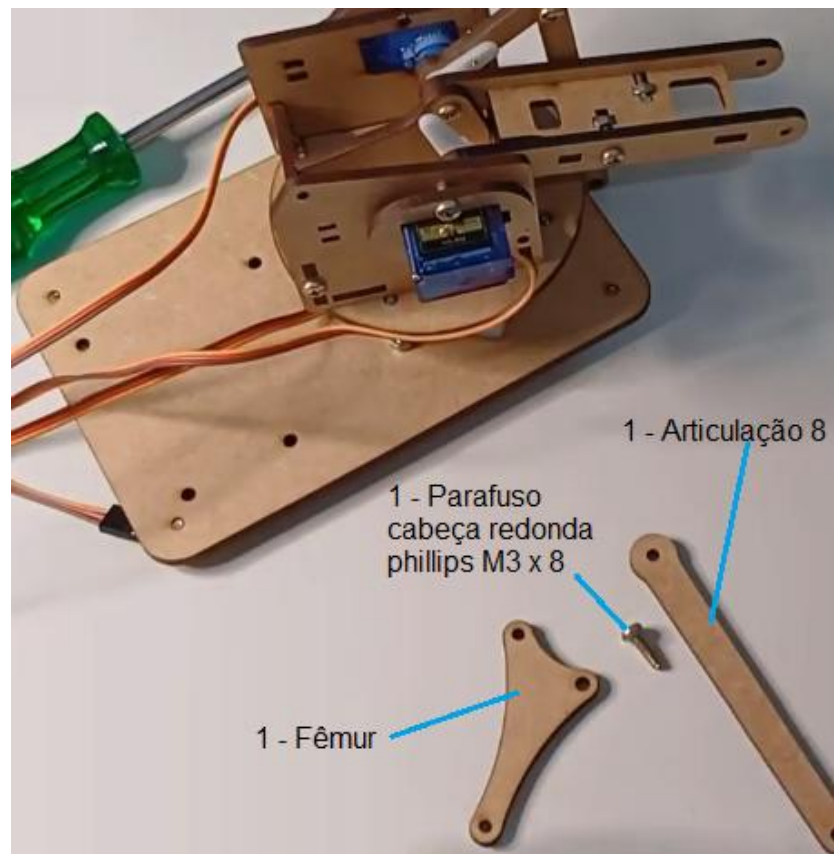


Figura 76 - Fixação da Articulação VIII e Fêmur na parte central do conjunto



Figura 77 – Montagem das peças concluída

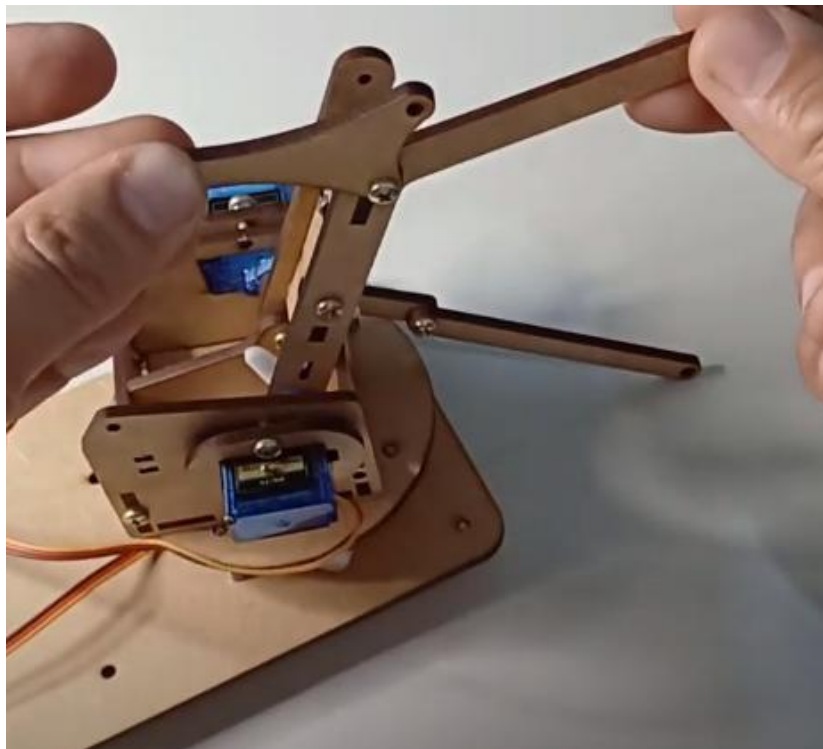
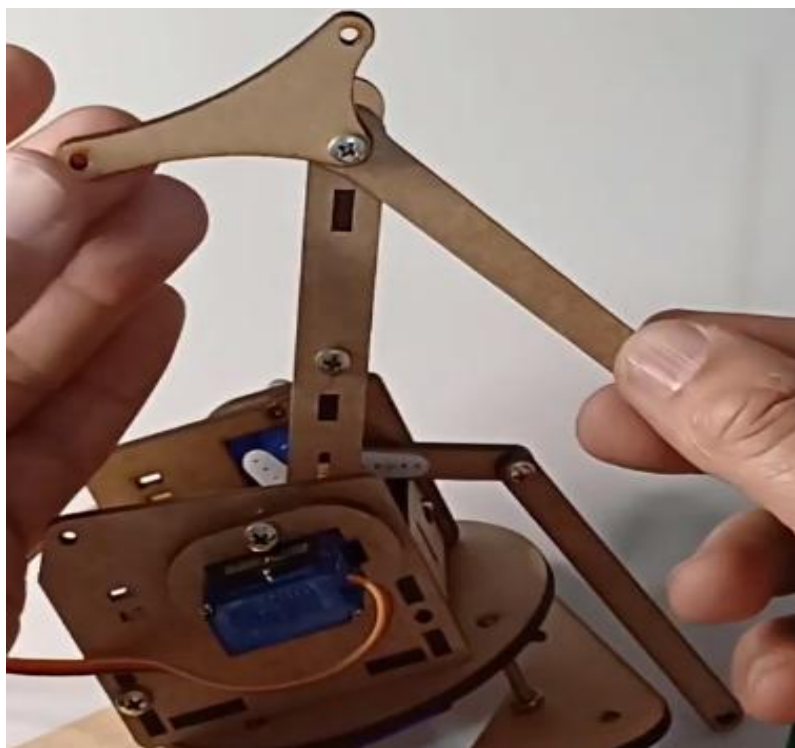


Figura 78 – Outro ângulo da montagem



4.3.16 Montagem da “Articulação 3” na outra extremidade do “Fêmur”

Figura 79 – Montagem geral, Articulação III e Apoio

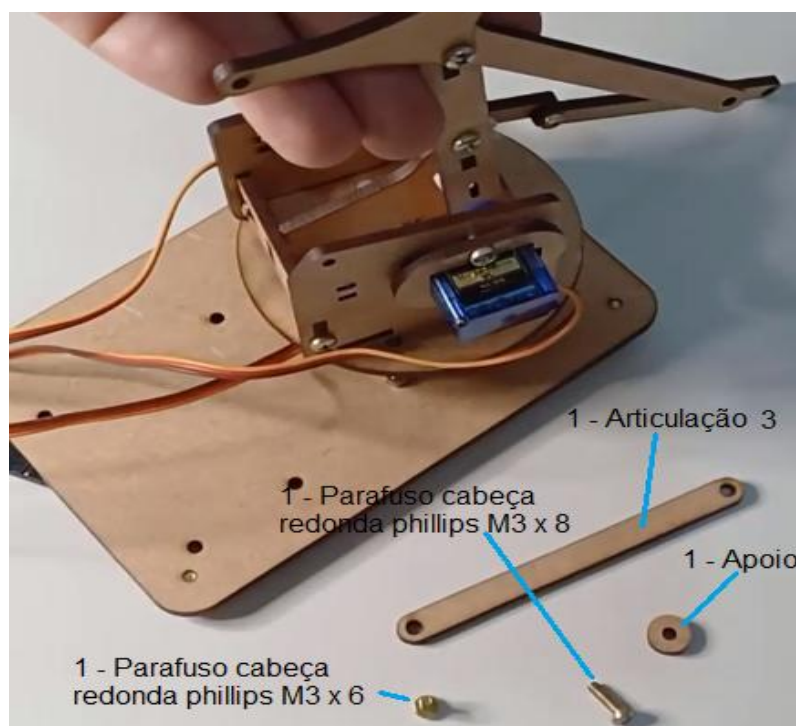


Figura 80 – Montagem geral da Articulação III no conjunto

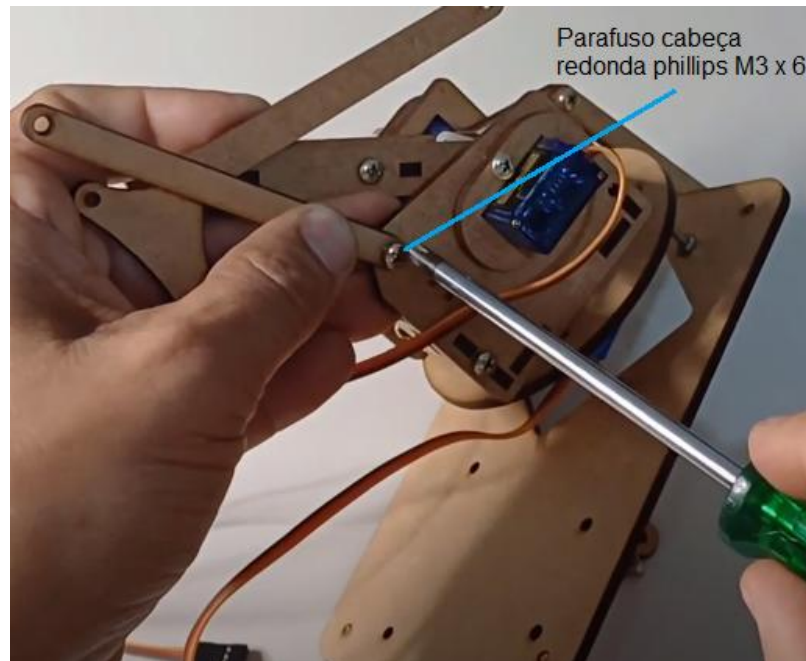


Figura 81 - Montagem da mesma Articulação III no Fêmur com o Apoio

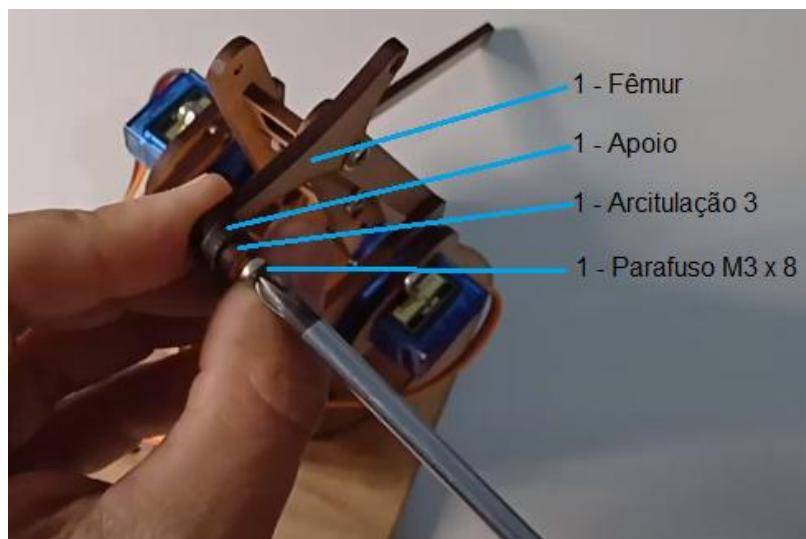
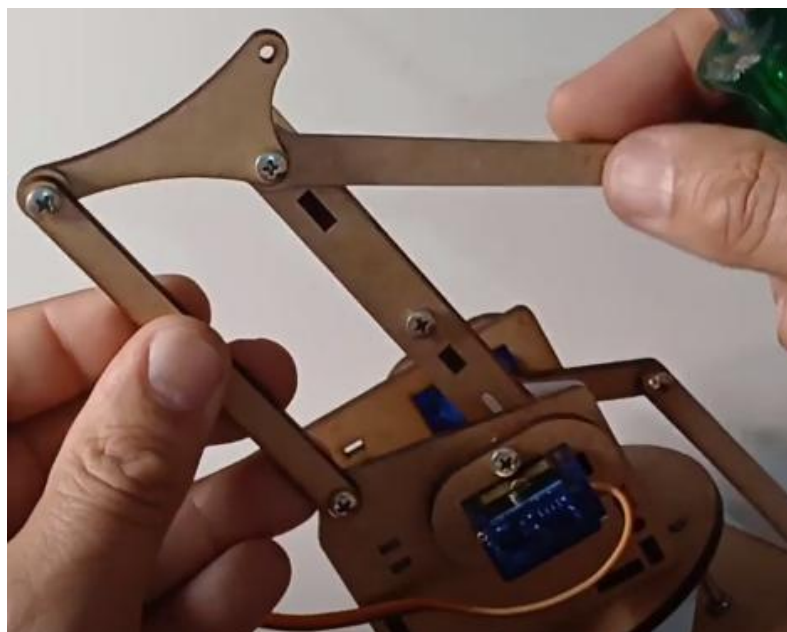


Figura 82 - Montagem das peças concluída



4.3.17 Montagem da “Articulação 3” na extremidade que falta do “Fêmur”

Figura 83 - Montagem geral, Articulação III e Apoio

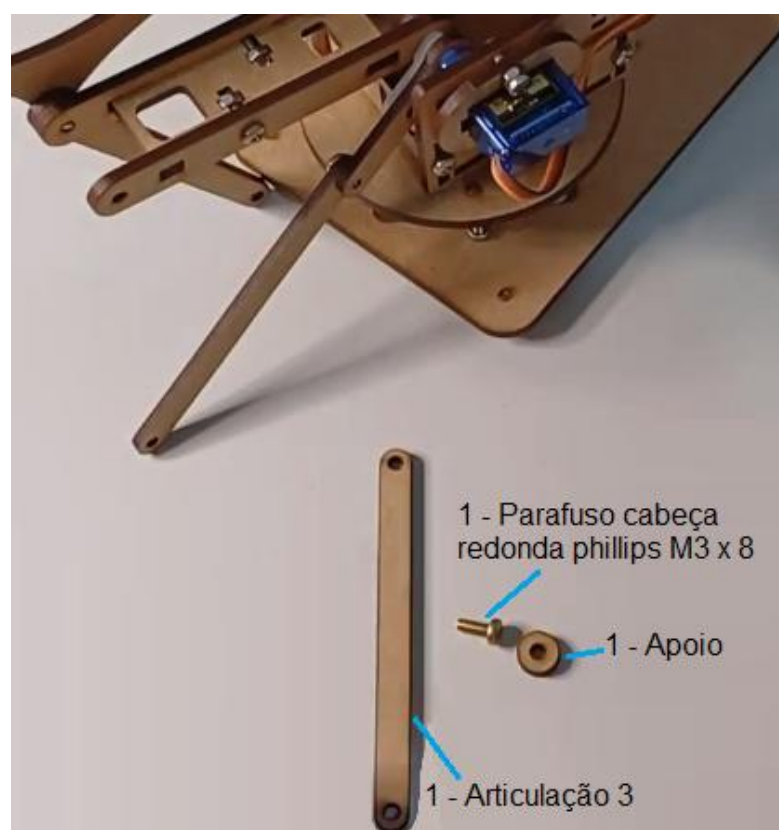


Figura 84 - Montagem da Articulação III no furo que falta do Fêmur

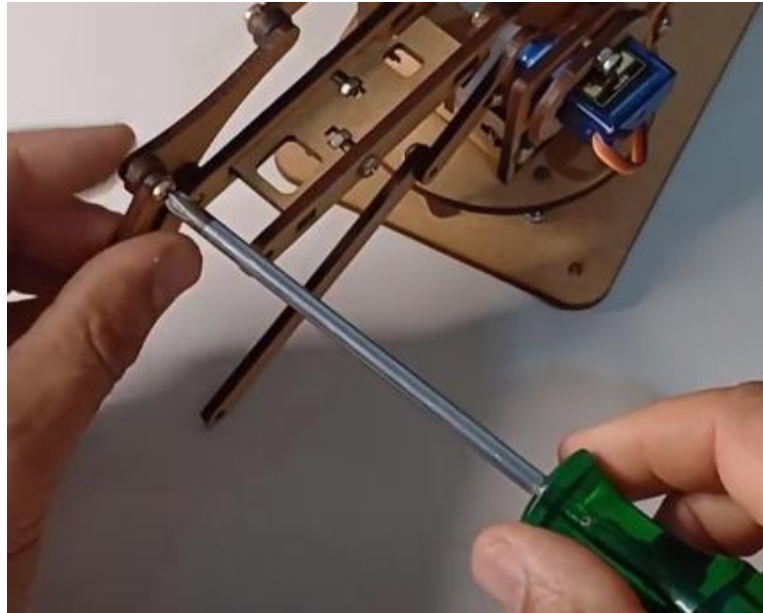


Figura 85 - Montagem das peças concluída



4.3.18 Montagem da “Haste 1”

Figura 86 - Montagem geral, Haste I e Parafusos

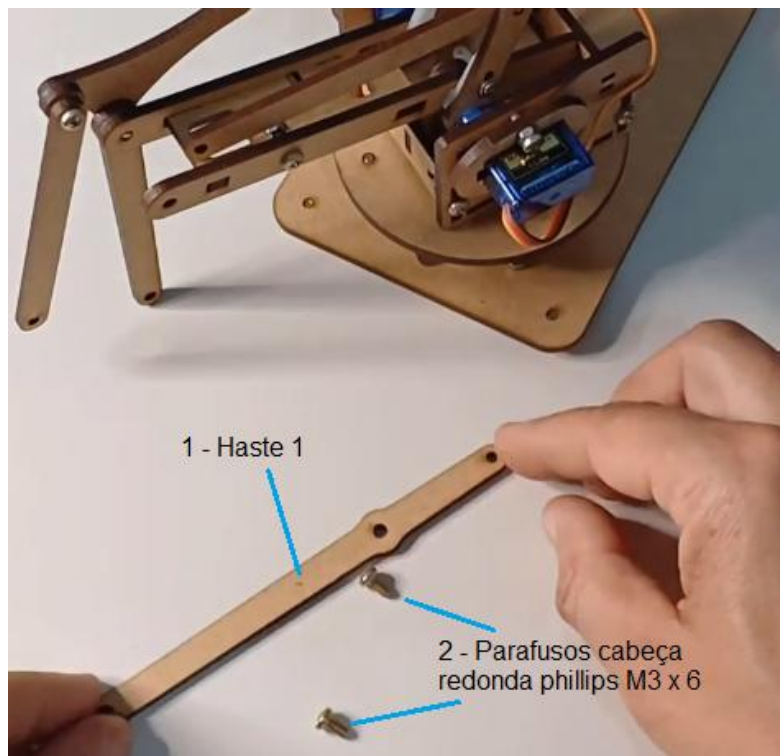


Figura 87 - Fixação da Haste I no Conjunto Central

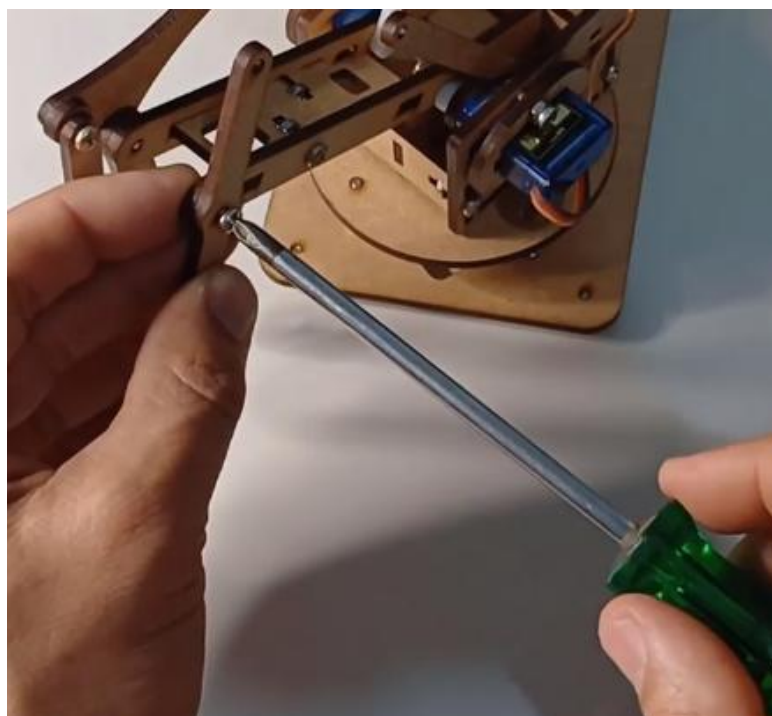


Figura 88 - Fixação da Haste I na Articulação III

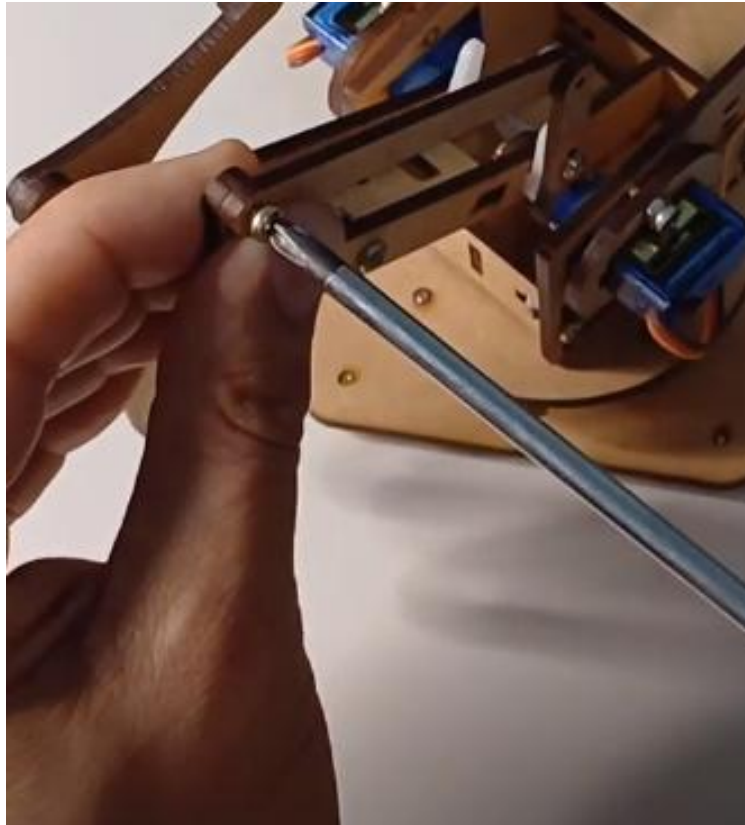
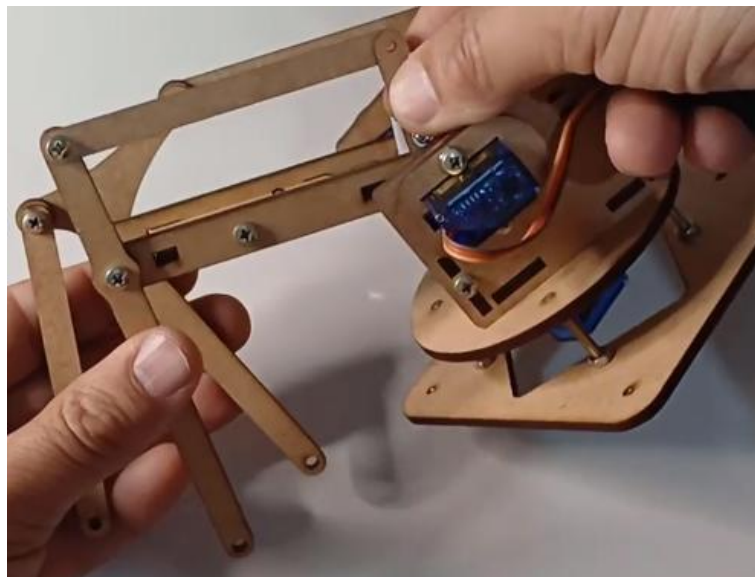


Figura 89 - Montagem das peças concluída



4.3.19 Montagem do motor II

Figura 90 - Motor II - Peças

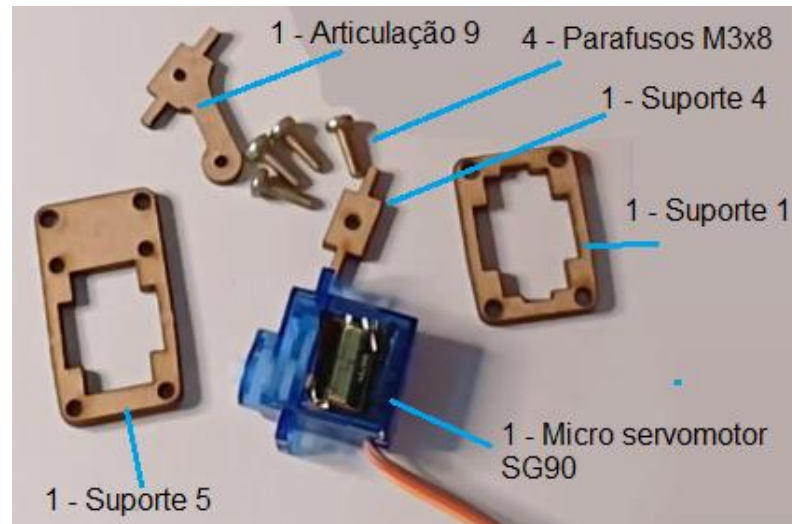


Figura 91 – Posicionamento do Suporte I no Motor II



Figura 92 – Inserção da Articulação IX no Suporte I

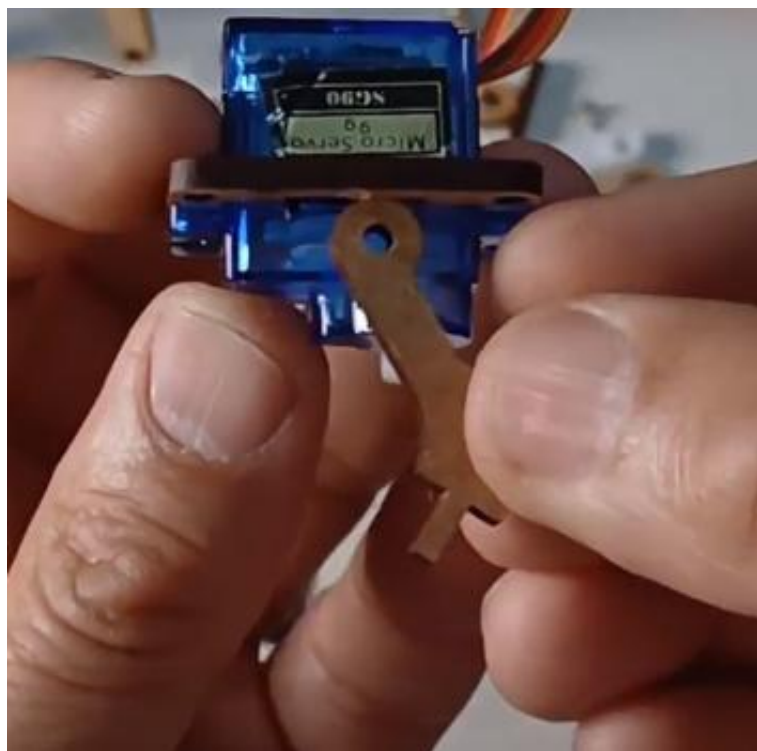


Figura 93 – Posição final da Articulação IX no Suporte I



Figura 94 – Inserção da Suporte IV no Suporte I



Figura 95 – Fixação do Suporte V

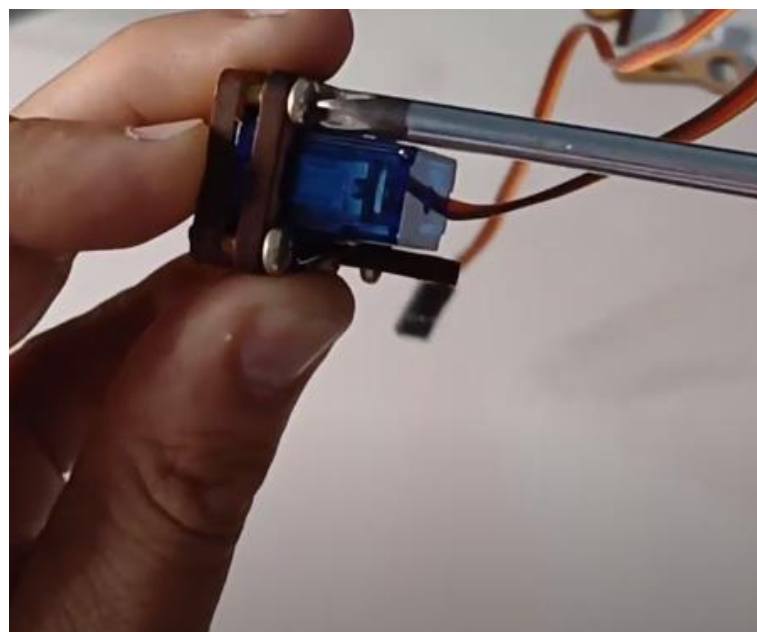


Figura 96 – Outro ângulo da montagem



4.3.20 Montagem das Garras (direita e esquerda) na montagem do motor II

Figura 97 – Garra - Peças

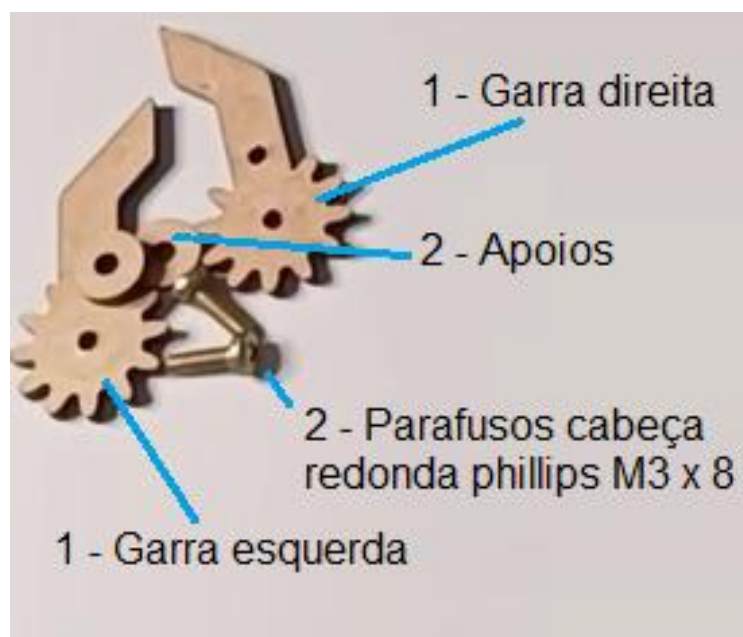
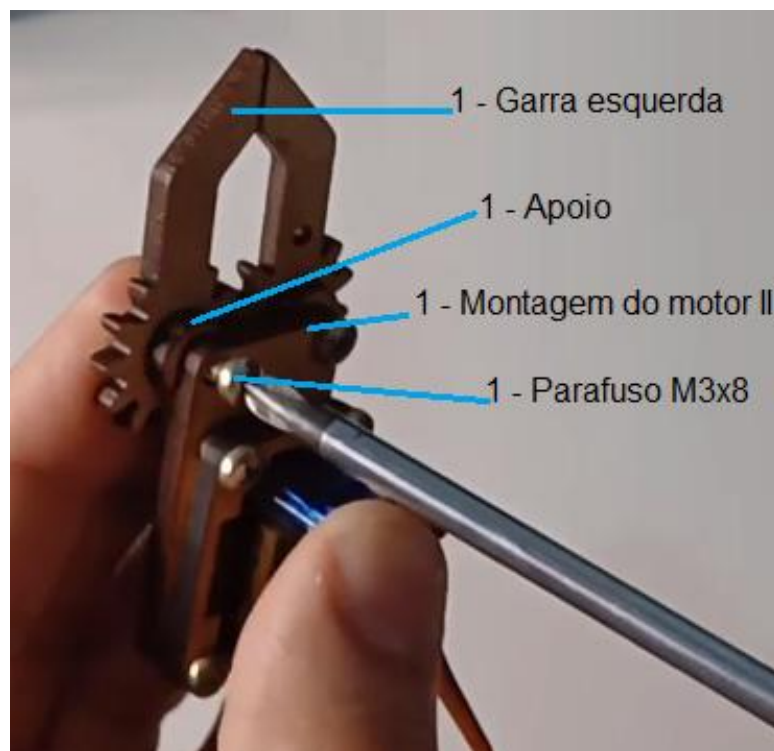


Figura 98 – Montagem da garra direita



Figura 99 – Montagem da garra esquerda



4.3.21 Montagem da articulação nas garras e todo o conjunto do motor II

Figura 100 - Articulação da Garra - Peças

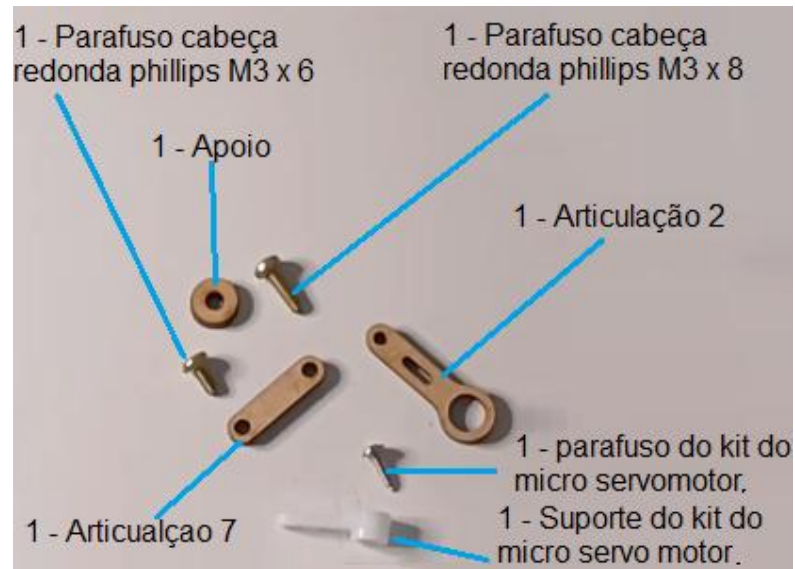


Figura 101 - Montagem da Articulação na Garra



Figura 102 - Montagem da segunda Articulação

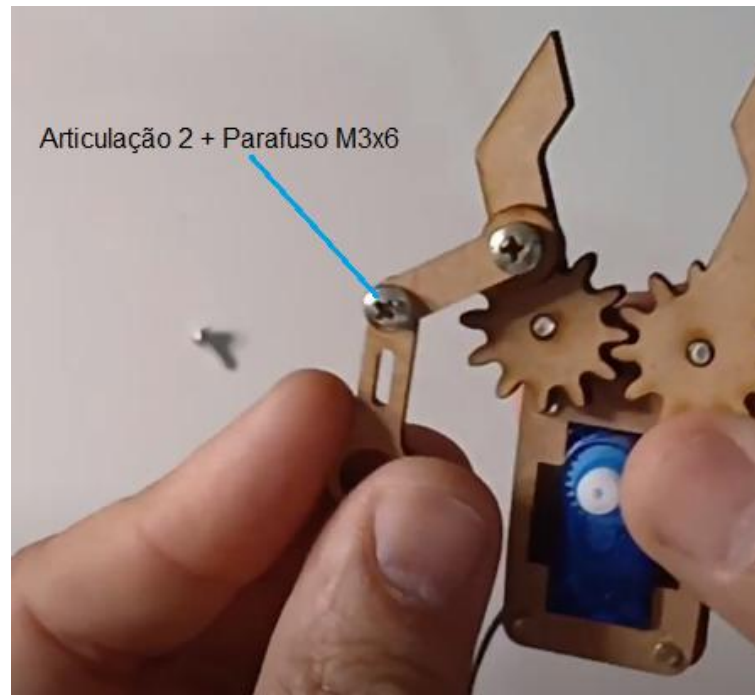


Figura 103 - Montagem das Articulações no Motor II



Figura 104 - Posicionamento e montagem do Suporte do Kit do Servo



4.3.22 Montagem da garra no conjunto de montagens

Figura 105 - Conjunto geral do Braço robótico e Garra

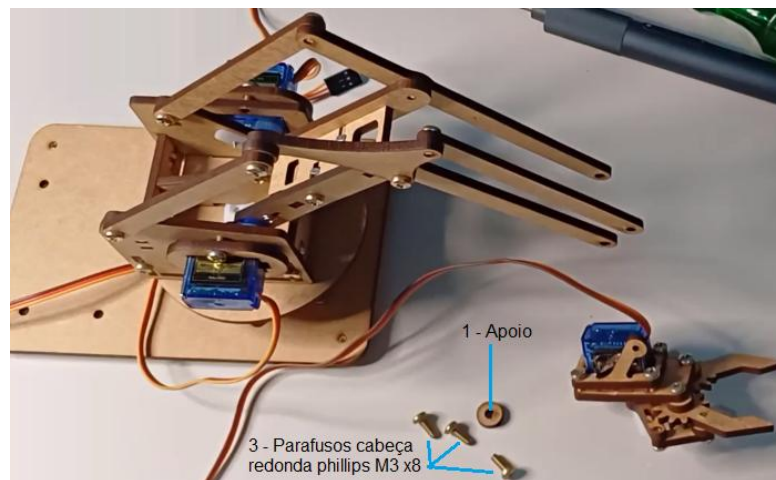


Figura 106 - Fixação do lado direito da garra no Braço robótico

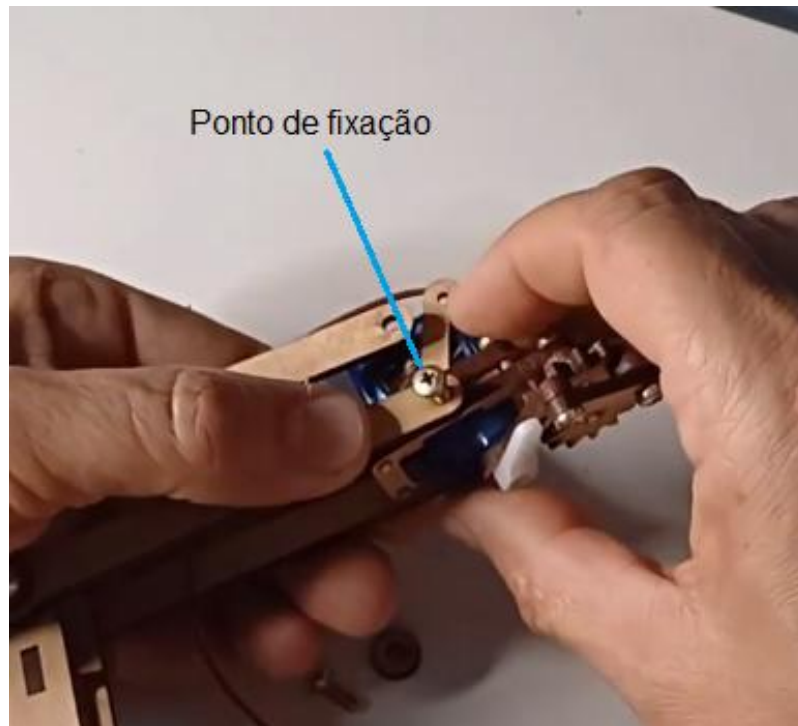


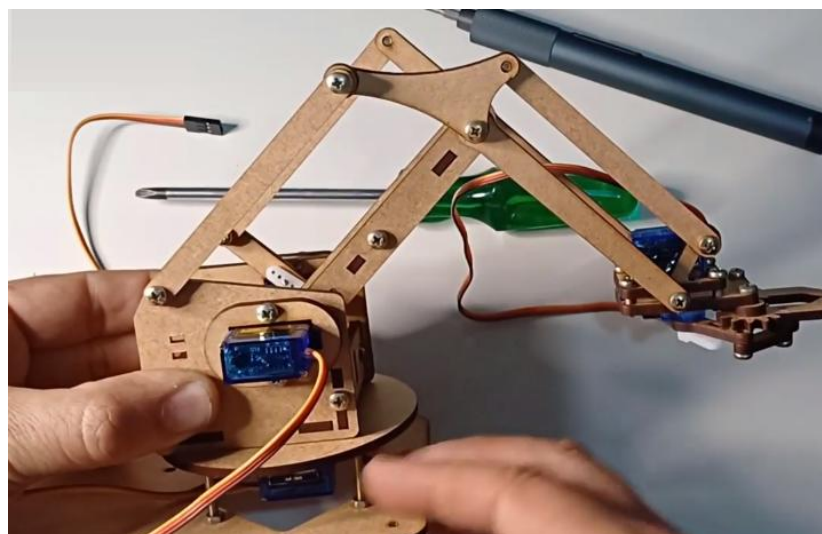
Figura 107 - Fixação concluída do lado direito da garra



Figura 108 - Fixação do ponto superior



Figura 109 - Montagem completa do Braço robótico

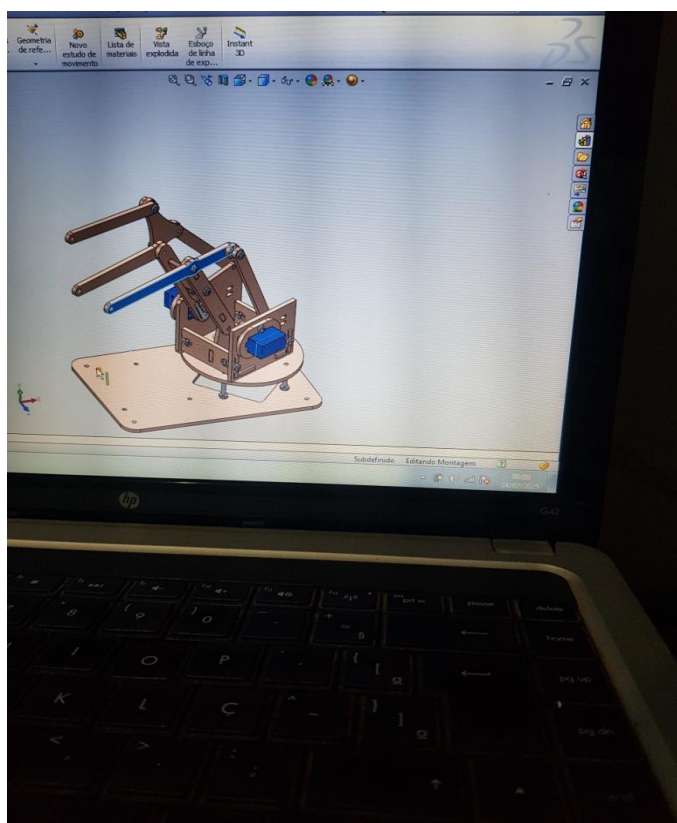


4.4 Testes e Ajustes

O projeto inicial do braço robótico foi obtido em formato DWG, com todas as medidas em polegadas. Optamos por extrair cada dimensão e redesenhar o robô em milímetros no SolidWorks, de modo a padronizar as unidades e garantir maior precisão no processo de modelagem.

Ao todo, foram modeladas 28 peças, que posteriormente foram montadas no próprio software. Durante essa etapa, realizamos ajustes dimensionais necessários para que o robô pudesse ser corretamente montado e apresentasse movimentação adequada dentro do ambiente virtual.

Figura 110 – Modelamento 3D das peças

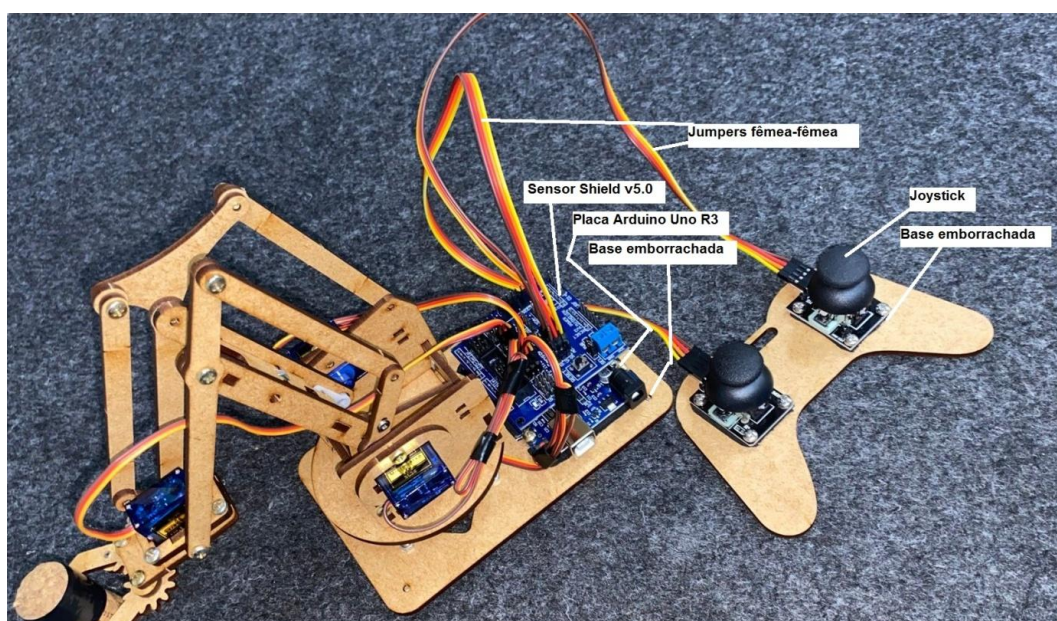


Após a conclusão da modelagem digital, as peças foram cortadas a laser em MDF, conforme o projeto redesenhado. Na montagem física, não foram necessários

ajustes significativos, uma vez que todas as medidas e encaixes apresentaram conformidade com o projeto.

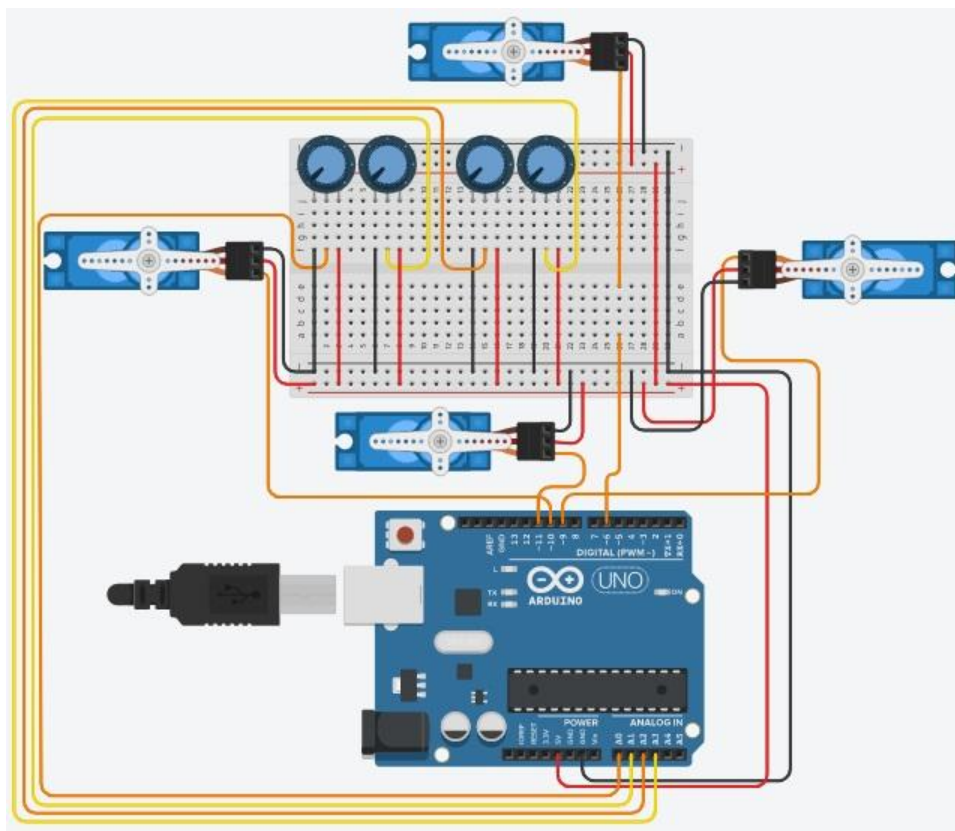
Durante a fixação da placa Arduino no corpo do robô, foi necessário utilizar uma base de mouse pad como isolante, devido à presença de pontos de solda e componentes na parte inferior da placa, que não poderiam entrar em contato direto com o MDF. O mesmo procedimento foi adotado para os joysticks, que foram fixados no controle, cada um com sua base recortada do mouse pad.

Figura 111 - Montagem da base emborrachhada nos joysticks e arduino



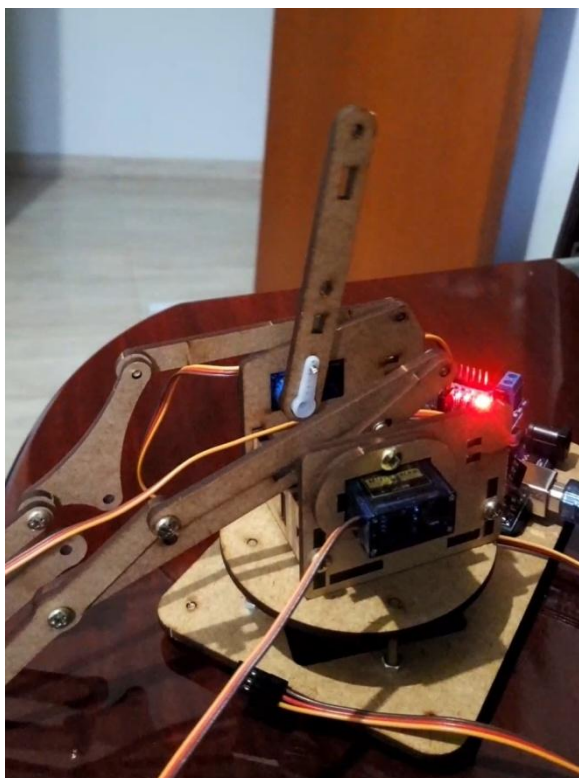
Com a estrutura mecânica finalizada, iniciamos o processo de programação do Arduino. Como o curso técnico em Eletromecânica não contemplava a disciplina de programação, realizamos um curso EAD específico de Arduino para adquirir o conhecimento necessário.

Durante o desenvolvimento do código, surgiram diversos desafios. Um deles foi a ausência de joysticks no simulador Tinkercad, o que exigiu a adaptação de potenciômetros (um para cada eixo do joystick) para simular seus sinais. A principal dificuldade consistiu em criar uma lógica de controle que compensasse o retorno automático do joystick ao ponto central quando liberado, de modo que o robô respondesse adequadamente aos comandos.

Figura 112 - Montagem dos potenciômetros no Thinkercad

Durante essa fase, também foi necessário realizar um ajuste mecânico fundamental: o reposicionamento das alavancas dos servomotores. Cada servo foi desmontado individualmente para que a alavanca fosse reinstalada em uma posição correta, correspondente ao ângulo inicial esperado na programação. Esse processo exigiu a leitura contínua dos valores de cada servo no monitor serial, de modo a alinhar corretamente os ângulos lógicos (programados) com as posições físicas do robô. Esse procedimento foi trabalhoso e demandou várias repetições até que todos os eixos se movimentassem de forma sincronizada e precisa.

Figura 113 – Ajuste de todas as alavancas dos servos



Foram desenvolvidos diversos códigos e realizadas inúmeras pesquisas até alcançar uma movimentação manual fluida e precisa.

Durante os testes de movimentação, surgiu a ideia de utilizar potes de medicamentos R35 como objetos de transporte do robô. Contudo, observamos que o sistema de garra não conseguia fixar adequadamente esses potes, o que comprometia a execução das tarefas. Optamos, então, por rolhas de vinho, que apresentavam peso reduzido e superfície mais porosa, oferecendo melhor aderência.

Figura 114 – Teste de pega de um pote R35

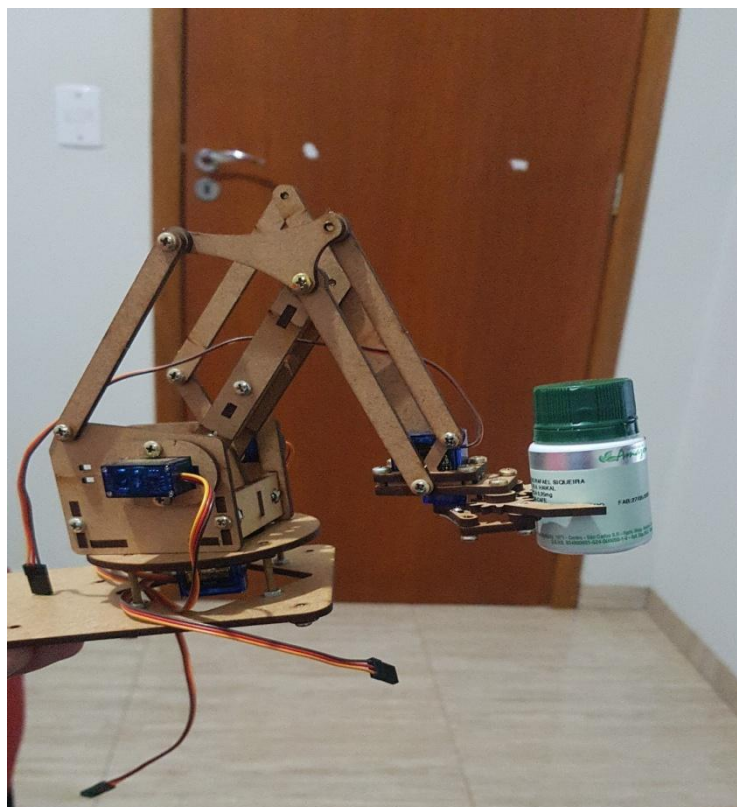


Figura 115 – Foto dos produtos (potes R35, jumpers, caixa, parafusos e camisa)

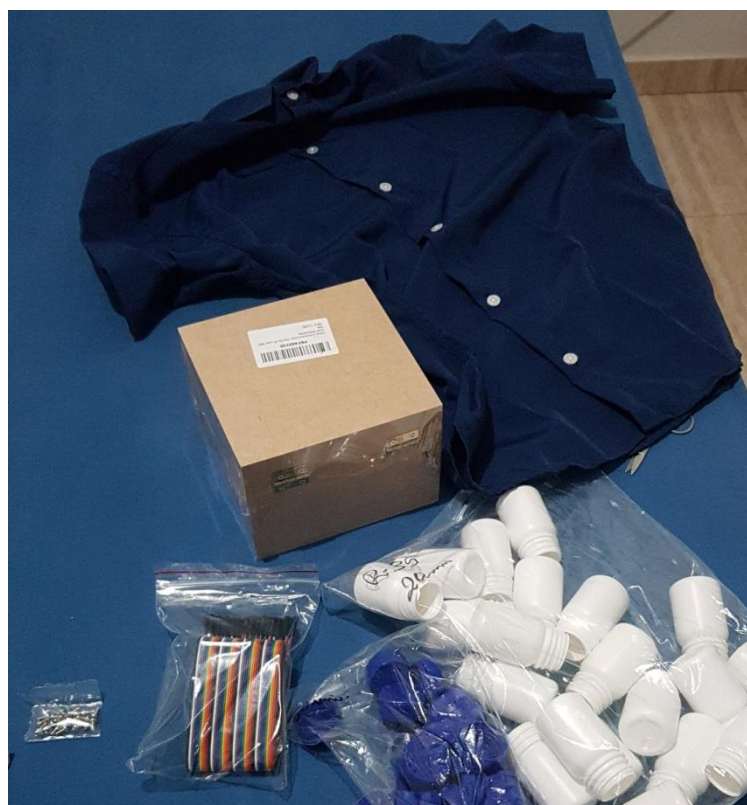


Figura 116 – Teste com rolhas de vinhos e revestimento emborrachado nas garras



Ainda assim, para aprimorar o desempenho da garra, recortamos novamente o mouse pad e o utilizamos como revestimento nas extremidades das garras, aumentando o atrito e melhorando a fixação. Embora os testes com as rolhas tenham obtido êxito, o resultado estético não foi satisfatório.

Posteriormente, encontramos em uma loja de utilidades pequenos objetos de silicone utilizados como protetores de parede para portas e janelas. Esses itens se mostraram ideais, pois possuíam saliências que facilitavam a pega e apresentavam melhor aparência estética, além de serem coloridos, o que conferiu um aspecto mais visualmente agradável ao conjunto.

Em seguida, identificamos a necessidade de o robô operar em uma posição fixa, a fim de evitar desvios de referência durante a execução dos movimentos automáticos entre os pontos de coleta (A) e entrega (B). Após testar algumas bases de madeira, optamos por utilizar uma caixa de papelão reforçada, revestida com papel contact preto, o que proporcionou melhor aparência e versatilidade, permitindo inclusive transportar o robô dentro da própria caixa durante as apresentações.

Figura 117 – Aquisição de material de revestimento da caixa e peças de silicone



Concluída a fase de movimentação manual, passamos à programação automática. Na apresentação final, o grupo demonstrará inicialmente o controle manual do robô via joysticks e, em seguida, ao pressionar os dois botões centrais dos controladores, o robô entrará em modo automático, realizando o transporte sequencial de quatro peças até a caixa de destino.

Durante essa etapa, foram realizados ajustes no código para torná-lo versátil e intercambiável, além de utilizar o Monitor Serial para a leitura e calibração das coordenadas de cada posição. Essa metodologia permitiu determinar com precisão as coordenadas de cada ponto de coleta e entrega, resultando em uma execução estável e previsível do sistema.

Por exemplo, na linha 261: `servo2.slowmove(17, 20); if (checarSairAutomatico())`
`return; delay(1000);`

O comando “slowmove” pertence a biblioteca “VarSpeedServo” adicionada por nós para controlar melhor os movimentos de cada servo. Nela podemos ver dois números, onde o 17 corresponde à angulação que o servo se posicionará e 20 corresponde à velocidade de 1 a 255 que o servo se moverá.

Figura 118 – Trecho do código onde foram inseridos valores do “Monitor Serial”

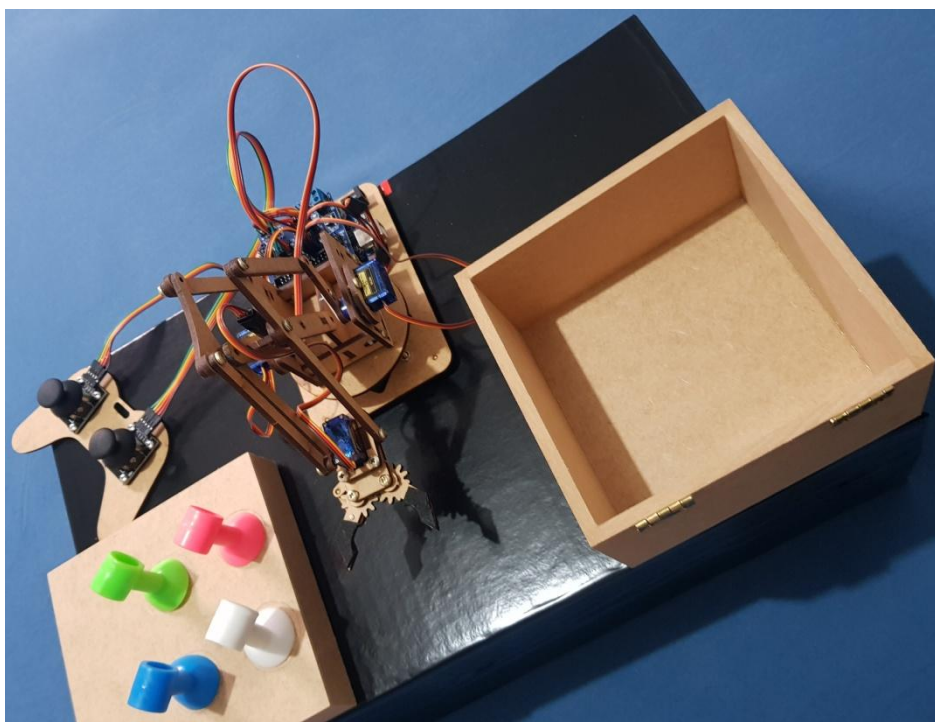
```
braco-robotico-arduino-3 | Arduino 1.8.19
Arquivo Editar Sketch Ferramentas Ajuda

braco-robotico-arduino-3 $

259
260 /* ----- PEÇA ROSA ----- */
261 servo2.slowmove(17, 20); if (checarSairAutomatico()) return; delay(1000);
262 servo1.slowmove(41, 20); if (checarSairAutomatico()) return; delay(1000);
263 servo2.slowmove(52, 18); if (checarSairAutomatico()) return; delay(1000);
264 servo4.slowmove(31, 18); if (checarSairAutomatico()) return; delay(1000);
265 servo3.slowmove(169, 20); if (checarSairAutomatico()) return; delay(1000);
266 servo2.slowmove(19, 20); if (checarSairAutomatico()) return; delay(1000);
267 servo1.slowmove(156, 20); if (checarSairAutomatico()) return; delay(2000);
268 servo4.slowmove(58, 18); if (checarSairAutomatico()) return; delay(1000);
269 servo4.slowmove(31, 20); if (checarSairAutomatico()) return; delay(1000);
270 servo1.slowmove(90, 18); if (checarSairAutomatico()) return; delay(1000);
271 moverParaInicial();
272
273 /* ----- PEÇA VERDE ----- */
274 servo4.slowmove(58, 20); if (checarSairAutomatico()) return; delay(1000);
275 servo2.slowmove(17, 20); if (checarSairAutomatico()) return; delay(1000);
276 servo1.slowmove(27, 20); if (checarSairAutomatico()) return; delay(1000);
277 servo3.slowmove(115, 20); if (checarSairAutomatico()) return; delay(1000);
278 servo2.slowmove(75, 18); if (checarSairAutomatico()) return; delay(1000);
279 servo4.slowmove(31, 18); if (checarSairAutomatico()) return; delay(1000);
280 servo3.slowmove(160, 20); if (checarSairAutomatico()) return; delay(1000);
281 servo2.slowmove(19, 20); if (checarSairAutomatico()) return; delay(1000);
282 servo1.slowmove(156, 20); if (checarSairAutomatico()) return; delay(2000);
283 servo4.slowmove(58, 18); if (checarSairAutomatico()) return; delay(1000);
284 servo4.slowmove(31, 20); if (checarSairAutomatico()) return; delay(1000);
285 servo1.slowmove(90, 18); if (checarSairAutomatico()) return; delay(1000);
286 moverParaInicial();
---
```

Todos os testes e ajustes descritos ocorreram conforme o planejado, culminando em um funcionamento satisfatório e coerente com os objetivos propostos do projeto.

Figura 119 – Conjunto completo do braço robótico



5. APLICAÇÕES

O protótipo desenvolvido, embora de caráter didático e experimental, apresenta potencial para diversas aplicações práticas em contextos industriais, educacionais e de pesquisa. Seu controle baseado em Arduino confere flexibilidade, baixo custo e facilidade de reprogramação, características altamente desejáveis em ambientes onde a automação e a adaptabilidade são essenciais.

No setor industrial, esse tipo de braço robótico pode ser empregado em tarefas de manipulação de pequenos componentes, como alimentação de máquinas, separação de peças em linhas de produção de baixa escala, ou em processos de inspeção visual com câmeras embarcadas. A estrutura modular e o controle preciso dos movimentos tornam o sistema adequado para simular operações que envolvem repetibilidade e precisão, como testes de resistência mecânica, área na qual se insere a empresa citada na conclusão — potencializando o uso do protótipo como ferramenta auxiliar em ensaios automatizados.

No meio acadêmico, o braço robótico pode ser utilizado como plataforma de ensino para disciplinas de automação, mecatrônica e programação embarcada, permitindo que alunos experimentem na prática conceitos de cinemática, controle de sistemas e integração de sensores. Sua construção com materiais acessíveis, como MDF e componentes eletrônicos amplamente disponíveis, também favorece sua replicabilidade em instituições de ensino técnico e superior.

Além disso, há espaço para aplicação em projetos de acessibilidade e reabilitação, como sistemas assistivos para pessoas com mobilidade reduzida, onde o braço robótico pode ser adaptado para executar tarefas simples do dia a dia. Com integração a interfaces como controle por voz, aplicativos móveis ou sensores mioelétricos, abre-se um campo promissor para inovação social.

Por fim, com os devidos aprimoramentos estruturais e eletrônicos, este tipo de sistema pode ser a base para o desenvolvimento de soluções mais robustas, como braços robóticos colaborativos em ambientes industriais inteligentes (Indústria 4.0), ou mesmo como ferramenta para prototipagem de dispositivos em laboratórios de inovação tecnológica.

6. CUSTOS TOTAIS DO PROJETO

Figura 120 - Tabela com os custos totais do projeto

ITENS	QTD	VALOR UNIT (R\$)	VALOR TOTAL (R\$)	OBS
Camisa social para apresentação	4	R\$ 58,33	R\$ 233,32	
Banner 1,2 x 0,9 m	1	R\$ 120,00	R\$ 120,00	
Monografia	1	R\$ 111,00	R\$ 111,00	
Micro servomotor SG90	4	R\$ 11,93	R\$ 47,72	
Módulo Joystick KY-023	2	R\$ 23,00	R\$ 46,00	
Placa Arduino Uno R3	1	R\$ 42,56	R\$ 42,56	
Colas e Fitas Adesivas	1	R\$ 25,40	R\$ 25,40	
Recipientes para transporte do robô	1	R\$ 24,90	R\$ 24,90	
Jumpers fêmea-fêmea	1	R\$ 23,90	R\$ 23,90	
Fonte 12V 1A	1	R\$ 23,63	R\$ 23,63	
Sensor Shield V5.0	1	R\$ 23,00	R\$ 23,00	
Parafusos M3 x 6	1	R\$ 19,90	R\$ 19,90	
Caixa de madeira	1	R\$ 19,88	R\$ 19,88	
Papel contact preto	1	R\$ 18,00	R\$ 18,00	
Pinos coloridos para transporte do robô	4	R\$ 2,99	R\$ 11,96	
Corte a laser do MDF	1	R\$ -	R\$ -	Doação
Material MDF	1	R\$ -	R\$ -	Doação
TOTAL			R\$ 791,17	

7. CONSIDERAÇÕES FINAIS

A realização deste trabalho de conclusão de curso proporcionou uma experiência prática extremamente enriquecedora, que integrou conhecimentos de diferentes áreas da Eletromecânica. O desenvolvimento de um braço robótico controlado por Arduino exigiu habilidades em mecânica — para o projeto, modelamento e montagem das peças —, em eletroeletrônica — para a elaboração do circuito e diagrama elétrico —, e em programação, área na qual foi necessário um esforço autodidata, uma vez que o curso não ofereceu formação específica para o controle de microcontroladores.

A construção do protótipo em MDF, com peças cortadas a laser e montadas com parafusos e porcas M3, demonstrou ser uma solução viável para testes e validações iniciais. A escolha do Arduino como plataforma de controle permitiu uma implementação funcional e didática, tornando o projeto acessível tanto do ponto de vista técnico quanto econômico.

Além do aprendizado técnico, o projeto reforçou a importância da iniciativa própria na busca por conhecimento, especialmente quando se deseja aplicar a teoria em soluções reais. A expectativa é que as competências desenvolvidas aqui possam ser aplicadas no ambiente industrial, contribuindo para a automação de processos e o desenvolvimento de soluções inovadoras. Um exemplo concreto de aplicação futura seria a utilização desses conhecimentos na empresa onde o membro Cláudio atua, voltada ao desenvolvimento de máquinas para ensaios de resistência mecânica, área intimamente relacionada ao escopo deste trabalho.

Portanto, este TCC não apenas consolidou o aprendizado adquirido ao longo do curso, como também abriu portas para futuras aplicações profissionais e aprimoramentos técnicos, marcando um passo importante na formação dos integrantes como técnicos em Eletromecânica.

8. REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 6023: Informação e documentação — Referências — Elaboração. Rio de Janeiro, 2018.

AUTODESK. Tinkercad [recurso eletrônico]. Disponível em: [\[https://www.tinkercad.com\]](https://www.tinkercad.com) (<https://www.tinkercad.com>). Acesso em: mar. 2025.

CENTRO PAULA SOUZA. Manual para a Elaboração do Trabalho de Conclusão de Curso. São Paulo, 2015.

CENTRO PAULA SOUZA. Manual para a Elaboração do Trabalho de Conclusão de Curso nas ETEC's. São Paulo, 2022.

CENTRO PAULA SOUZA. Planejamento e Desenvolvimento do TCC. São Paulo, 2011.

DASSAULT SYSTÈMES. SolidWorks 2012 [programa de computador]. Waltham: Dassault Systèmes, 2012.

INSTRON. Instron 6800 Series Universal Testing Systems. YouTube, 24 jan. 2020. Disponível em: [\[https://www.youtube.com/watch?v=HTyq3R2Etnw\]](https://www.youtube.com/watch?v=HTyq3R2Etnw) (<https://www.youtube.com/watch?v=HTyq3R2Etnw>). Acesso em: mar. 2025.

MARTY33. Parametric Servo Gripper [modelo 3D]. Thingiverse, 25 jan. 2014. Disponível em: [\[https://www.thingiverse.com/thing:360108\]](https://www.thingiverse.com/thing:360108) (<https://www.thingiverse.com/thing:360108>). Acesso em: mar. 2025.

APÊNDICES:

APÊNDICE A – CÓDIGO-FONTE UTILIZADO NO ARDUINO	49 - 60
--	---------

ANEXOS:

ANEXO A – DATASHEET DA FONTE 12V 1A	13
ANEXO B – DATASHEET DO SERVOMOTOR SG90	13 - 14
ANEXO C – DATASHEET DA PLACA ARDUINO UNO R3	14 - 15
ANEXO D – DATASHEET DO SENSOR SHIELD V5.0	16
ANEXO E – MÓDULO JOYSTICK KY-023	17 - 18