

## Capítulo I – O Evento: Inovação, Aprendizado e Competição

**Art. 1º** O **PUC Minas RoboChallenge 2025** é mais que uma competição; é uma celebração da criatividade, da tecnologia e da colaboração. Organizado pelo curso de Engenharia de Computação, o evento convida todos os estudantes da PUC Minas a transformarem ideias em realidade, aprenderem na prática e se conectarem com o futuro da tecnologia.

**Art. 2º** São objetivos do evento:

- I. Estimular a inovação, criatividade e o trabalho em equipe;
- II. Promover a integração entre hardware, software e inteligência artificial;
- III. Incentivar soluções alinhadas aos Objetivos de Desenvolvimento Sustentável (ODS) da ONU;
- IV. Aplicar conhecimentos teóricos em desafios práticos e multidisciplinares;
- V. Incentivar a participação de alunos ingressantes e o compartilhamento de conhecimento aberto.

## Capítulo II – Dos Participantes e das Equipes

**Art. 3º** A participação é restrita a estudantes regularmente matriculados em qualquer curso de graduação da PUC Minas.

**Art. 4º** Cada equipe poderá inscrever um robô por modalidade, sendo permitido que uma mesma equipe participe de múltiplas modalidades

**Art. 5º** São permitidos 04 alunos(as) por equipe para as modalidades Sumô, Labirinto e Freestyle.

**Art. 6º** A modalidade Robocode poderá ser realizada individualmente ou em duplas.

**Art. 7º** São responsabilidades de cada equipe:

- I. Ler e compreender integralmente este regulamento.
- II. Garantir que seu projeto e robô estejam em conformidade com todas as regras aqui descritas.
- III. Zelar pelo espírito competitivo saudável e pelo fair play.

## Capítulo III – Das Inscrições

**Art. 8º** O período de inscrições será de **22 de setembro a 23 de outubro de 2025**.

**Art. 9º** As inscrições deverão ser realizadas através de formulário online disponível em <https://forms.office.com/r/U9cWDjz3mU>. No ato da inscrição, a equipe deverá fornecer:

- I. Nome da equipe.
- II. Nome completo, R.A. (Registro Acadêmico) e curso de todos os integrantes.
- III. Indicação das modalidades em que a equipe competirá.

## Capítulo IV – Dos Requisitos Técnicos Gerais

**Art. 10º** Os robôs devem ser desenvolvidos pelos alunos, não é permitida a utilização de robôs comerciais (vendidos em lojas). É permitida, entretanto, robôs montados a partir de kits.

**Art. 11º** Os projetos deverão ser publicados até o dia **31 de outubro de 2025** em repositório público no **GitHub Classroom**, contendo:

- a. Código-fonte funcional e comentado
- b. Documentação (README.md) clara
- c. Lista de materiais e esquema de montagem
  - i. Diagramas elétricos (se houver)
  - ii. Arquivos de modelagem 3D (se houver)
  - iii. Lista de materiais – BOM (se houver)
  - iv. Processo de construção
- d. Explicação da IA utilizada

**Art. 12º** Documentar e compartilhar seu trabalho em plataformas como o GitHub é uma prática padrão na indústria de tecnologia. Isso constrói seu portfólio e ensina habilidades essenciais de colaboração. A não publicação do código no Github implica na desclassificação da equipe.

## Capítulo V – Das Modalidades e Suas Regras Específicas

### Seção I – Modalidade: Sumô de Robôs (Classe 0,5kg Autônomo)

**Art. 13º** Dois robôs autônomos batalham em uma arena circular (Dohyo). O objetivo é empurrar o robô adversário para fora da arena.

**Art. 14º Arena (Dohyo):** Círculo de madeira ou MDF na cor preta, com 77 cm de diâmetro, elevado a 5 cm do chão, com uma borda branca de 5 cm de largura.

**Art. 15º** Especificações do Robô:

- I. Dimensões: No início da partida, o robô deve caber em uma caixa de 10 cm x 10 cm. Não há limite de altura. Após o sinal de início, o robô poderá se expandir.
- II. Peso: O peso máximo permitido é de 0,5 kg.
- III. Controle: O robô deve ser 100% autônomo. Nenhum tipo de controle remoto é permitido após o início da partida.

#### Art. 16º Regras da Batalha:

- I. Uma partida consiste em 3 rounds de 1 minuto cada. A primeira equipe a vencer 2 rounds vence a partida.
- II. Um robô vence um round ao forçar o oponente a tocar qualquer parte fora da arena.
- III. Se nenhum robô sair da arena em 1 minuto, a vitória do round será decidida por *Yusei* (vantagem), a critério do juiz, para o robô que demonstrou maior combatividade.

#### Seção II – Modalidade: Labirinto (Classe Micromouse)

**Art. 17º Objetivo:** O robô autônomo deve encontrar o caminho do ponto de partida até o ponto de chegada de um labirinto no menor tempo possível.

**Art. 18º Labirinto:** Construído em uma base de 16x16 células, com paredes de altura e largura padronizadas. O layout exato do labirinto será desconhecido pelas equipes antes do início da competição.

#### Art. 19º Especificações do Robô:

- I. Dimensões: O robô deve ter no máximo 25 cm de largura e 25 cm de comprimento. Não há limite de altura ou tamanho mínimo.
- II. Peso: Não há limite de peso, mas é autorregulado (um robô pesado jamais será rápido).
- III. Controle: 100% autônomo.

#### Art. 20º Regras da Competição:

- I. O tempo é cronometrado do início ao fim do percurso.
- II. O robô pode realizar uma ou mais corridas de reconhecimento para mapear o labirinto, seguidas por uma corrida de velocidade (speed run). O tempo que vale é o da speed run mais rápida.
- III. Penalidades de tempo podem ser aplicadas em caso de intervenção manual da equipe.

#### Seção III – Modalidade: Categoria Livre (Freestyle)

**Art. 21º** Apresentar um robô inovador que se destaque pela tecnologia embarcada, design e, principalmente, por sua capacidade de ajudar pessoas ou atender a pelo menos um dos Objetivos de Desenvolvimento Sustentável (ODS) da ONU.

**Art. 22º** Cada equipe terá 5 minutos para uma apresentação ("pitch") de seu projeto a uma banca de jurados, com 3 minutos demonstração do robô e 3 minutos para perguntas e respostas.

#### Art. 23º Critérios de Avaliação (Total: 100 Pontos):

| Critério                        | Descrição   | Pontuação |
|---------------------------------|---|-----------|
| Inovação e Complexidade Técnica | Qualidade da implementação de IA, uso de sensores e atuadores avançados, complexidade do projeto. | 30 pts    |

|                                  |  |        |
|----------------------------------|--|--------|
| Impacto Social/Aderência aos ODS | Clareza na solução de um problema real e alinhamento com os ODS da ONU             | 30 pts |
| Design e Construção              | Estética, robustez, acabamento e originalidade do projeto mecânico e eletrônico    | 20 pts |
| Apresentação e Documentação      | Clareza do pitch, qualidade da demonstração e excelência da documentação no GitHub | 20 pts |

## Seção IV – Modalidade: Batalha de Rôbos com RoboCode

**Art. 24º** Objetivo: Programar em um encontro em laboratório presencial, em local e horário estabelecido, um tanque de guerra para batalha virtual visando competir contra outros robôs em uma arena (Combate 1:1).

A competição RoboCode é individual ou dupla, em um laboratório especialmente preparado e sem acesso à Internet. O vencedor é determinado por um sistema de pontuação do próprio RoboCode, que recompensa o dano causado e a sobrevivência.

Os alunos deverão acessar a página oficial do Robocode, baixar o simulador, aprender a usar e treinar. Link para acesso: <https://robocode.sourceforge.io/>.

**Art. 25º** Especificações do Robô:

1. O robô é uma entidade puramente de software, controlada por um código desenvolvido pela equipe.
2. A linguagem de programação oficial para esta modalidade é Java, utilizando o ambiente de desenvolvimento Robocode.
3. O robô é composto por três partes que podem ser controladas independentemente: o veículo (chassi), o canhão e o radar.

**Art. 26º** Regras da Competição:

1. Esta é uma categoria individual com um único aluno.
2. As batalhas ocorrem em uma arena virtual de dimensões pré-definidas.
3. Os terão 90 minutos para desenvolver o seu robô em laboratório, sem consulta a internet.
4. Uma partida consiste em múltiplos rounds, e a pontuação final é o somatório dos pontos de cada round
5. A pontuação a ser adotada será a do próprio sistema do Robocode que é calculada com base nos seguintes critérios:
  - I. **Dano por Projétil:** Pontos são concedidos por cada ponto de dano infligido a um oponente.
  - II. **Bônus por Destruição:** Um bônus é concedido ao destruir um oponente.
  - III. **Dano por Colisão:** Danos causados por colisões (ramming) também geram pontos.

- IV. **Sobrevivência:** Robôs que sobrevivem por mais tempo recebem pontos à medida que outros são eliminados.
- V. **Bônus para o Último Sobrevivente:** O último tanque na arena recebe um bônus de pontos significativo.

## Capítulo V – Do Dia da Competição

**Art. 27º** O evento ocorrerá nos dias **29 e 30 de outubro de 2025**, no Campus Lourdes, no mezanino do 2º Andar do Prédio 1. Detalhes dos locais das batalhas e instruções serão enviadas para os inscritos e disponibilizadas no site oficial da competição.

**Art. 28º** Todas as equipes deverão se apresentar no início do evento para a **Homologação**, onde os robôs serão inspecionados quanto ao cumprimento das regras de peso e dimensão. Robôs fora das especificações não poderão competir.

**Art. 29º** Para que uma modalidade seja realizada, são necessárias pelo menos 04 equipes inscritas em cada uma delas e 08 alunos(as) inscritos na modalidade RoboCode.

## Capítulo VI – Da Comissão Julgadora

**Art. 30º** A Comissão Organizadora e a Banca de Jurados são soberanas em suas decisões, não cabendo recurso aos resultados e às interpretações deste regulamento.

## Capítulo VII – Da Premiação e Reconhecimento

**Art. 31º** Serão premiadas com troféus e/ou medalhas as equipes classificadas em 1º, 2º e 3º lugar em cada uma das modalidades.

**Art. 32º** Prêmios Especiais: Para celebrar todos os aspectos dos talentos demonstrados, a Comissão Organizadora, mediante disponibilidade de recursos de patrocinadores, poderá conceder distinções e premiações especiais.

## Capítulo VIII – Do Espírito do Evento e Disposições Gerais

**Art. 33º** Espírito do Evento e Fair Play: O RoboChallenge é um ambiente para competir, mas acima de tudo, para aprender e colaborar. Espera-se que todas as equipes ajudem umas às outras, compartilhem conhecimento e mantenham uma atitude respeitosa. Lembre-se: seu concorrente de hoje pode ser seu colega de equipe amanhã.

**Art. 34º** Condutas antidesportivas, desrespeitosas ou qualquer violação intencional das regras por parte de qualquer membro de uma equipe poderão resultar na desclassificação sumária dela.

**Art. 35º** Ao se inscreverem, as equipes autorizam a PUC Minas a utilizar fotos e vídeos do evento, dos projetos e de seus integrantes para fins de divulgação acadêmica e institucional.

## Capítulo IX – Cronograma

| Data               | Atividade  |
|--------------------|--|
| 22/09/2025         | Lançamento do Edital e Início das Inscrições     |
| 23/10/2025         | Encerramento das Inscrições                      |
| 29 e 30 de Outubro | Dias da Competição                               |
| 31 de Outubro      | Prazo máximo para publicação do código no Github |
| 04/11/2025         | Premiação no EngComp Day                         |

## Capítulo X – Contato e Informações

**Art. 36º** Dúvidas sobre este regulamento podem ser enviadas ao e-mail oficial do evento: [robochallenge@pucminas.br](mailto:robochallenge@pucminas.br).

Belo Horizonte, 22 de setembro de 2025.

Comissão Organizadora

PUC Minas RoboChallenge 2025

Curso de Engenharia de Computação – PUC Minas



## Anexo A – Sugestão de Kit Básico para Iniciantes

Este anexo não é uma regra, mas um guia para ajudar as equipes que estão começando. Para equipes que buscam um ponto de partida, a organização sugere um conjunto de componentes versáteis e de baixo custo.

- **Microcontrolador:** ESP32 ou Raspberry Pi Pico/2040.
- **Chassi:** Placas de acrílico ou MDF cortadas a laser, ou um chassi impresso em 3D.
- **Motores:** 2 ou 4 motores DC com caixa de redução ("motor amarelo") e rodas compatíveis.
- **Driver de Motor:** Ponte H L298N (para robôs leves) ou DRV8833.
- **Sensores:** Kit básico de sensores contendo módulo ultrassônico (HC-SR04) e seguidores de linha infravermelho (TCRT5000).
- **Alimentação:** Suporte para 4 pilhas AA ou 2 baterias de lítio 18650 com circuito de gerenciamento.



## Anexo B- Barema dos Resultados

### B.1. Barema de Avaliação – Modalidade: Sumô de Robôs

Nesta modalidade, a avaliação é direta e baseada no resultado das batalhas.

#### Súmula de Partida – Sumô de Robôs (Classe 0,5kg)

| Partida Nº       |                  | Fase   |   | Juiz:               |
|------------------|------------------|--|---|---------------------|
|                  |                  | <input type="checkbox"/> Oitavas de Final <input type="checkbox"/> Quartas de final<br><input type="checkbox"/> Semifinal <input type="checkbox"/> Final |   |                     |
| <b>Equipe A:</b> |                  | <b>Equipe B:</b>   |   |                     |
| Round            | Duração<br>mm:ss | Vencedor<br>(A ou B)   | Método da Vitória   | Observações do Juiz |
| 1                | (ex: 0:25s)      |  | <input type="checkbox"/> Forçou o oponente para fora<br><input type="checkbox"/> Yusei (Vantagem) |                     |
| 2                |                  |  | <input type="checkbox"/> Forçou o oponente para fora<br><input type="checkbox"/> Yusei (Vantagem) |                     |
| 3*               |                  |  | <input type="checkbox"/> Forçou o oponente para fora<br><input type="checkbox"/> Yusei (Vantagem) |                     |

\*O Round 3 só ocorre se houver empate nos dois primeiros rounds

#### Resultado Final da Partida:

**Vitória da Equipe A**

**Vitória da Equipe B**

#### Instruções para o Juiz:

1. **Vitória por Ponto (Yuko):** Uma equipe marca um ponto (vence o round) se qualquer parte do robô adversário tocar o chão fora da arena.
2. **Vitória por Vantagem (Yusei):** Se o tempo de 1 minuto expirar sem um vencedor claro, o juiz deve conceder a vitória ao robô que demonstrou maior combatividade, controle da arena ou que esteve mais próximo de empurrar o oponente.
3. **Vencedor da Partida:** A equipe que vencer 2 rounds primeiro é declarada vencedora da partida e avança na chave do torneio.



## B. 2. Barema de Avaliação – Modalidade: Labirinto (Micromouse)

Nesta modalidade, o critério principal é o tempo. O barema é uma **Tabela de Classificação** baseada no menor tempo registrado.

### Tabela de Classificação – Labirinto (Micromouse)

| Posição | Equipe | Melhor Tempo | Penalidades | Tempo Final |
|---------|--------|--------------|-------------|-------------|
| 1º      |        |              |             |             |
| 2º      |        |              |             |             |
| 3º      |        |              |             |             |
| ...     |        |              |             |             |

### Instruções para Cronometragem e Avaliação:

- Corridas (Runs):** Cada equipe tem um tempo total (ex: 7 minutos) na arena para realizar quantas corridas (runs) desejar.
- Mapeamento vs. Speed Run:** As primeiras corridas são geralmente para mapear o labirinto. A equipe deve informar ao juiz quando uma corrida é uma "Speed Run" (corrida de velocidade).
- Tempo Válido:** O tempo que conta para a classificação é o da "**Speed Run**" mais rápida completada com sucesso.
- Penalidades:**
  - Toque no Robô (Intervenção Manual):** Adicionar **+5 segundos** ao tempo daquela corrida. Se o robô for retirado e colocado no início, a corrida atual é invalidada.
  - Não Completar o Percurso:** A corrida é invalidada (DNF - Did Not Finish).
- Tempo Final:** O Tempo Final é calculado como: Tempo Final = Melhor Tempo (Speed Run) + Soma das Penalidades (daquela corrida).
- Classificação:** As equipes são ordenadas do menor para o maior Tempo Final.

### B.3. Barema de Avaliação – Modalidade: Categoria Livre (Freestyle)

Nome da Equipe: \_\_\_\_\_ Nome do Projeto: \_\_\_\_\_ Jurado: \_\_\_\_\_

| Critério de Avaliação                         | Max        | Descrição e Níveis de Avaliação   | Pontos | Observações do Jurado |
|---|------------|---|--------|-----------------------|
| <b>1. Inovação e Complexidade Tecnológica</b> | <b>30</b>  | <p><b>Excelente (25-30):</b> Uso avançado e criativo de IA. Integração complexa de múltiplos sensores/atuadores. Solução genuinamente inovadora.</p> <p><b>Bom (18-24):</b> Aplicação funcional de IA. Boa integração de componentes. O projeto demonstra um desafio técnico considerável.</p> <p><b>Regular (10-17):</b> Uso básico de IA (ex: desvio de obstáculos simples). Pouca complexidade na integração. Projeto tecnicamente simples.</p> <p><b>Insuficiente (0-9):</b> IA não demonstrada ou trivial. Projeto muito simples ou não funcional.</p>                     |        |                       |
| <b>2. Impacto Social e Relevância (ODS)</b>   | <b>30</b>  | <p><b>Excelente (25-30):</b> O projeto aborda um problema real de forma clara e eficaz. O alinhamento com um ou mais ODS é profundo e bem justificado. Potencial de impacto real é evidente.</p> <p><b>Bom (18-24):</b> O problema abordado é relevante e a solução é plausível. O alinhamento com os ODS é claro.</p> <p><b>Regular (10-17):</b> O problema é pouco definido ou a solução é pouco prática. O alinhamento com os ODS é superficial.</p> <p><b>Insuficiente (0-9):</b> O projeto não tem um propósito claro de impacto social ou não se alinha a nenhum ODS.</p> |        |                       |
| <b>3. Design e Construção</b>                 | <b>20</b>  | <p><b>Excelente (16-20):</b> Design original, esteticamente agradável e funcional. Construção robusta, com ótimo acabamento e organização interna (cabos, etc.).</p> <p><b>Bom (11-15):</b> Design funcional. Construção sólida, com bom acabamento.</p> <p><b>Regular (6-10):</b> Design genérico ou pouco otimizado. Construção funcional, mas com falhas no acabamento ou robustez.</p> <p><b>Insuficiente (0-5):</b> Design pobre. Construção frágil, desorganizada ou inacabada.</p>   |        |                       |
| <b>4. Apresentação e Documentação</b>         | <b>20</b>  | <p><b>Excelente (16-20):</b> Apresentação (pitch) clara, envolvente e dentro do tempo. Demonstração do robô sem falhas. Documentação no GitHub exemplar.</p> <p><b>Bom (11-15):</b> Apresentação clara e bem estruturada. Demonstração funcional. Documentação no GitHub completa.</p> <p><b>Regular (6-10):</b> Apresentação confusa ou incompleta. Demonstração com falhas. Documentação no GitHub básica.</p> <p><b>Insuficiente (0-5):</b> Apresentação ruim. Robô não funciona na demonstração. Documentação inexistente ou muito pobre.</p>                               |        |                       |
| <b>PONTUAÇÃO TOTAL</b>                        | <b>100</b> |   |        |                       |

#### B.4. Barema de Avaliação – Modalidade: Robocode

| Equipe / Robô | Desempenho em Batalha<br>(40%) | Estratégia e Comportamento<br>(20%) | Eficiência do Código<br>(15%) | Criatividade<br>(15%) | Documentação<br>(10%) | Nota Final (%) | Observações               |
|---------------|--------------------------------|-------------------------------------|-------------------------------|-----------------------|-----------------------|----------------|---------------------------|
| Time 1        | 35                             | 18                                  | 12                            | 13                    | 9                     | 87             | Boa evasão e mira precisa |
| Time 2        | ...                            | ...                                 | ...                           | ...                   | ...                   | ...            | ...                       |

#### Fórmula para cálculo da nota final:

**Nota Final (%)** = (Batalha × 0.4) + (Estratégia × 0.2) + (Código × 0.15) + (Criatividade × 0.15) + (Documentação × 0.1)

**Exemplo:** (35×0.4) + (18×0.2) + (12×0.15) + (13×0.15) + (9×0.1) = 87

#### Dicas para preenchimento

- **Desempenho em batalha:** use a pontuação automática do Robocode (normalmente de 0 a 100).
- **Estratégia e comportamento:** avalie movimentação, mira, radar, evasão.
- **Eficiência do código:** clareza, modularidade, uso da API.
- **Criatividade:** soluções únicas, robôs com personalidade ou estilo.
- **Documentação:** comentários no código, README explicativo.