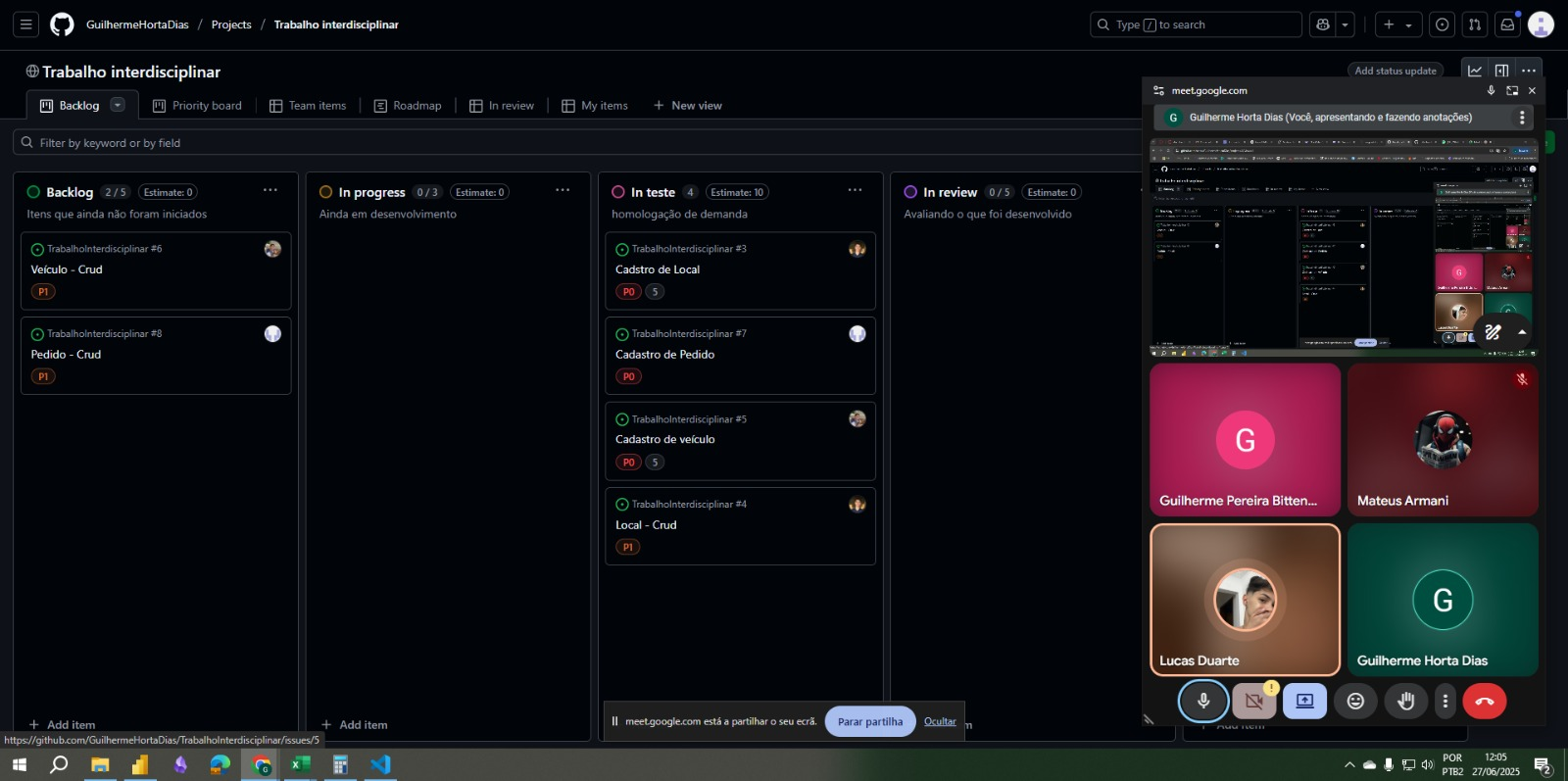
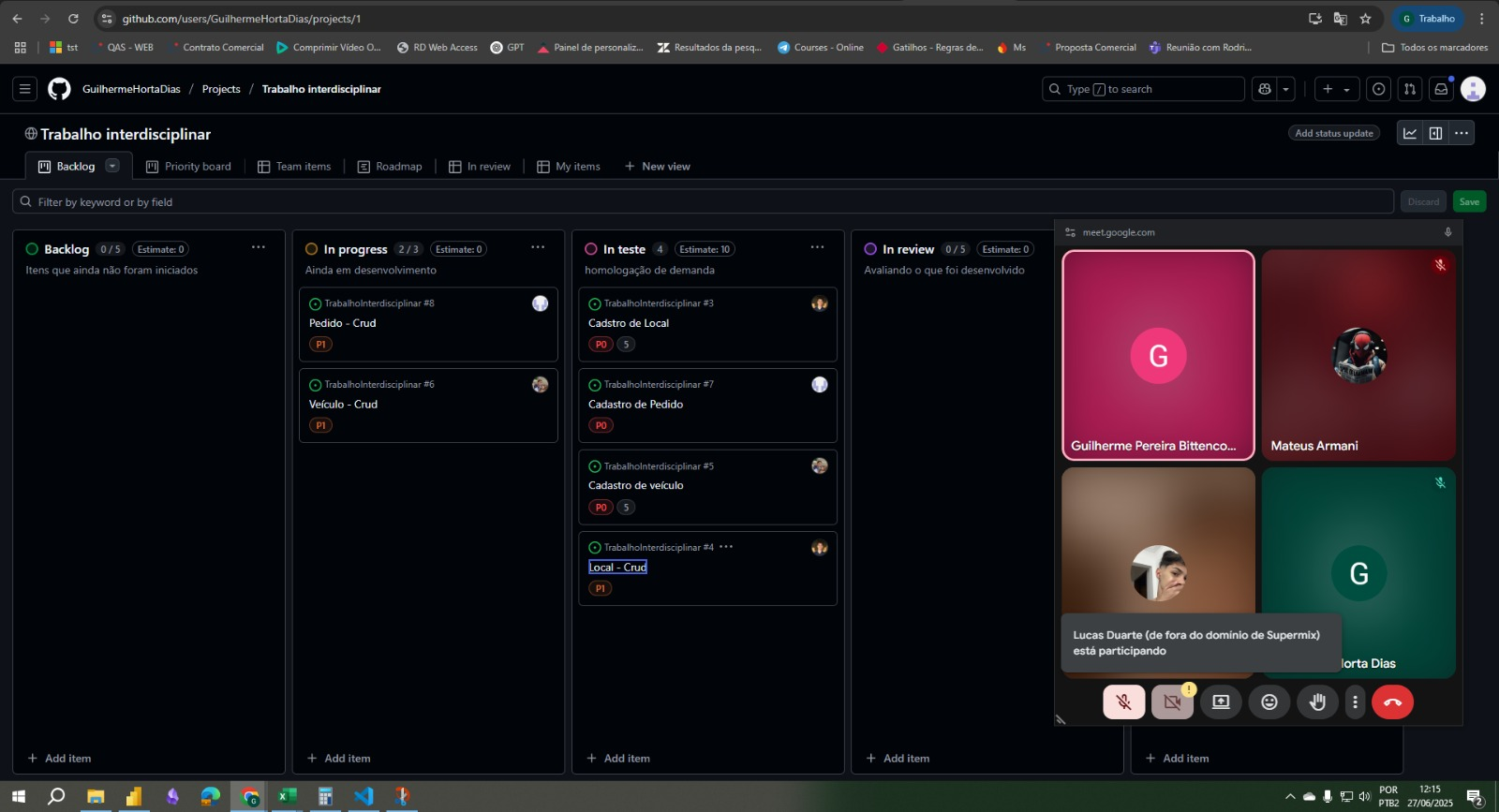
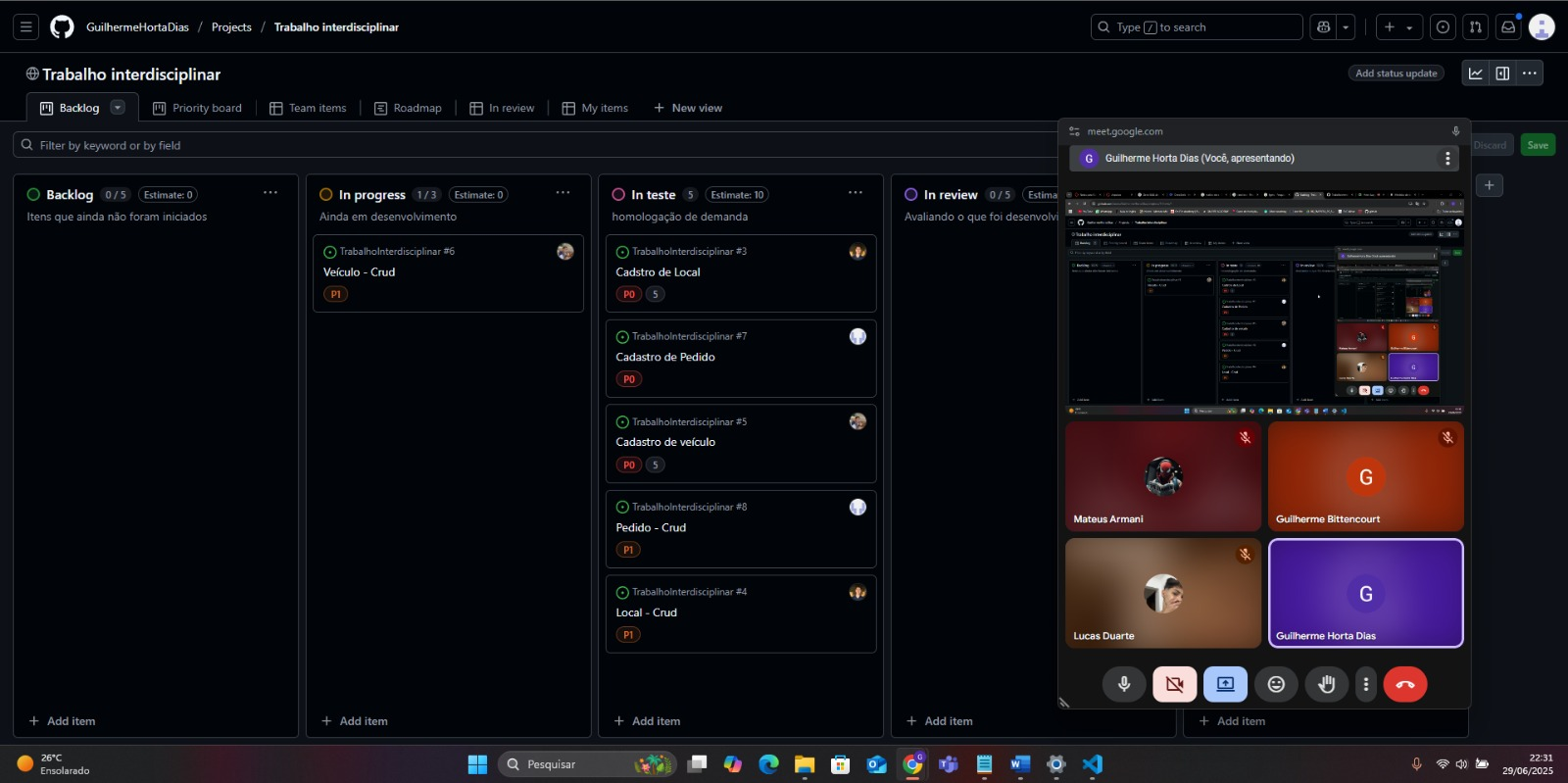
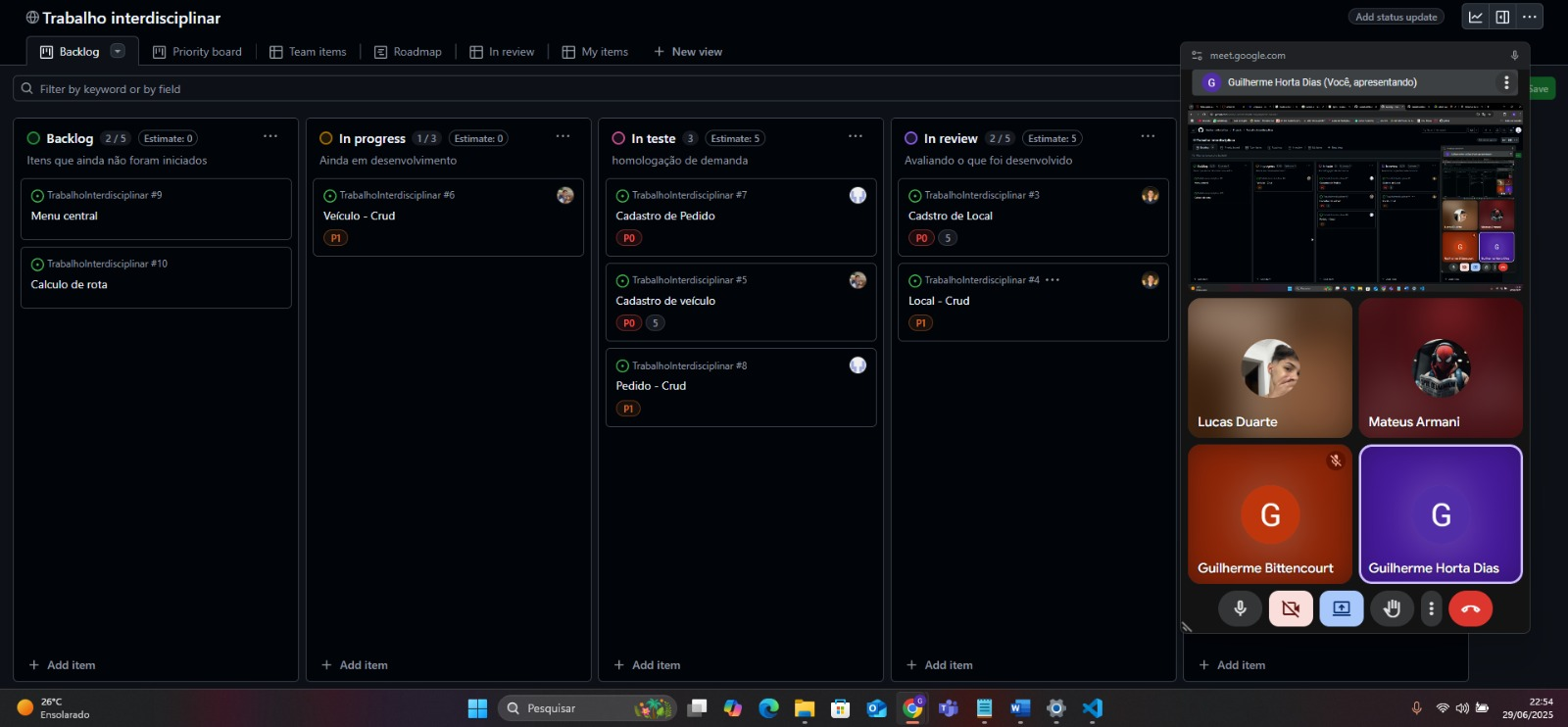
Trabalho de AEDS + FUNDAMENTOS  
  
alunos:   
Lucas Batista Duarte  
Mateus de Moura Armani

Guilherme Horta Dias

Guilherme Pereira Bittencourt  
  
github:  
<https://github.com/GuilhermeHortaDias/TrabalhoInterdisciplinar>  
https://github.com/users/GuilhermeHortaDias/projects/1  
  
Ritos:   
O acompanhamento das sprints de acordo com as histórias também está disponível no github projects:  








Assinatura das funções:  
  
  
1. bool criar\_local(const string& nome, float x, float y)

Cria um novo local no sistema.

Parâmetros:

nome: nome do local

x, y: coordenadas do local

Retorno:

true: local criado com sucesso

false: já existe um local com esse nome ou erro de gravação

2. void listar\_locais()

Lista todos os locais ativos no sistema.

Parâmetros: nenhum

Retorno: void

3. bool atualizar\_local(const string& nome\_atual, const string& novo\_nome, float x, float y)

Atualiza os dados de um local existente.

Parâmetros:

nome\_atual: nome do local a ser atualizado

novo\_nome: novo nome desejado

x, y: novas coordenadas

Retorno:

true: atualizado com sucesso

false: local não encontrado ou nome já em uso

4. bool deletar\_local(const string& nome)

Desativa um local com base no nome.

Parâmetros:

nome: nome do local a ser desativado

Retorno:

true: local desativado

false: local não encontrado

5. vector<Local> get\_todos\_locais()

Retorna todos os locais ativos cadastrados.

Parâmetros: nenhum

Retorno: vetor de Local

6. Local get\_local\_by\_nome(const string& nome)

Busca e retorna um local pelo nome.

Parâmetros:

nome: nome do local

Retorno:

Local encontrado

Local inativo (default) se não encontrado

Pedidos:

7. bool criar\_pedido(const string& origem, const string& destino, float peso)

Cria um novo pedido.

Parâmetros:

origem: local de origem

destino: local de entrega

peso: peso da carga

Retorno:

true: criado com sucesso

false: erro ao gravar

8. bool atualizar\_pedido(int id, const string& nova\_origem, const string& novo\_destino, float novo\_peso)

Atualiza um pedido existente.

Parâmetros:

id: ID do pedido

nova\_origem, novo\_destino, novo\_peso: novos dados

Retorno:

true: atualizado com sucesso

false: pedido não encontrado

9. bool deletar\_pedido(int id)

Desativa (deleta) um pedido.

Parâmetros:

id: ID do pedido

Retorno:

true: deletado

false: não encontrado

10. void listar\_pedidos()

Lista todos os pedidos ativos.

Parâmetros: nenhum

Retorno: void

11. vector<Pedido> get\_todos\_pedidos()

Retorna todos os pedidos ativos.

Parâmetros: nenhum

Retorno: vetor de Pedido

12. Pedido get\_pedido\_by\_id(int id)

Retorna um pedido pelo ID.

Parâmetros:

id: identificador do pedido

Retorno:

Pedido encontrado

Pedido vazio se não encontrado

Veículos:

13. bool criar\_veiculo(const string& tipo, const string& placa, float capacidade)

Cadastra um novo veículo.

Parâmetros:

tipo, placa: dados de identificação

capacidade: capacidade máxima de carga

Retorno:

true: criado

false: placa já existe ou erro de gravação

14. bool atualizar\_veiculo(const string& placa\_atual, const string& novo\_tipo, const string& nova\_placa, float nova\_capacidade)

Atualiza dados de um veículo.

Parâmetros:

placa\_atual: placa atual do veículo

novo\_tipo, nova\_placa, nova\_capacidade: novos dados

Retorno:

true: atualizado

false: não encontrado ou nova placa já existe

15. bool deletar\_veiculo(const string& placa)

Desativa um veículo.

Parâmetros:

placa: placa do veículo

Retorno:

true: deletado

false: não encontrado

16. void listar\_veiculos()

Lista veículos ativos.

Parâmetros: nenhum

Retorno: void

17. vector<Veiculo> get\_todos\_veiculos()

Retorna todos os veículos ativos.

Parâmetros: nenhum

Retorno: vetor de Veiculo

18. Veiculo get\_veiculo\_by\_placa(const string& placa)

Busca veículo pela placa.

Parâmetros:

placa: placa a ser buscada

Retorno:

Veículo encontrado

Veículo vazio se não encontrado

Rotas:

19. void gerar\_rotas\_otimizadas()

Gera e salva rotas otimizadas com base em pedidos e veículos.

Parâmetros: nenhum

Retorno: void

20. void listar\_rotas\_com\_detalhes()

Exibe as rotas salvas com detalhes de trajeto, peso e veículo.

Parâmetros: nenhum

Retorno: void

21. void limpar\_rotas\_antigas()

Remove todas as rotas do arquivo.

Parâmetros: nenhum

Retorno: void

22. vector<Rota> get\_todas\_rotas()

Retorna todas as rotas ativas.

Parâmetros: nenhum

Retorno: vetor de Rota  
  
  
  
Casos de teste:   
 **criar\_local(nome, x, y)**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| nome: string | "Loja A" | Local criado com sucesso (true) | string vazia, nome já existente, nome muito longo | Local não criado (false) |
| x, y: float | 10.0, 20.0 | Local salvo com coordenadas corretas | letras, NaN, valores extremamente grandes ou negativos | Local não criado (false) |

## **atualizar\_local(nome\_atual, novo\_nome, x, y)**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| nome\_atual: string | "Loja A" | Local atualizado (true) | nome não existente | Mensagem de erro, retorna false |
| novo\_nome: string | "Loja B" | Nome alterado | nome já usado por outro local | Erro de nome duplicado |
| x, y: float | 15.0, 25.0 | Coordenadas atualizadas | NaN, strings, números inválidos | Atualização falha |

## **deletar\_local(nome)**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| nome: string | "Loja B" | Local desativado (true) | nome inexistente, string vazia | Mensagem de erro, retorna false |

## **criar\_pedido(origem, destino, peso)**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| origem/destino: str | "Garagem", "Loja B" | Pedido criado com ID novo | locais inexistentes, strings vazias | Erro, retorna false |
| peso: float | 30.0 | Pedido salvo | peso negativo, zero, valor muito alto ou não numérico | Erro ao salvar |

## **deletar\_pedido(id)**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| id: int | 1 | Pedido desativado (true) | id inexistente, negativo | Pedido não encontrado (false) |

## **criar\_veiculo(tipo, placa, capacidade)**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| tipo: string | "Caminhão" | Tipo registrado | string vazia, caracteres especiais | Veículo não criado |
| placa: string | "ABC1234" | Placa registrada | placa já existente, inválida | Veículo rejeitado |
| capacidade: float | 100.0 | Capacidade atribuída | capacidade zero, negativa ou não numérica | Erro, veículo não salvo |

## **atualizar\_veiculo(placa\_atual, novo\_tipo, nova\_placa, nova\_capacidade)**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| placa\_atual: string | "ABC1234" | Veículo encontrado | placa inexistente | Erro, não atualiza |
| nova\_placa: string | "XYZ9999" | Placa alterada | placa já existente | Erro, não permite duplicidade |
| nova\_capacidade: float | 80.0 | Capacidade atualizada | valores negativos, letras | Erro na atualização |

## **gerar\_rotas\_otimizadas()**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| Dados do sistema | pedidos e veículos ativos suficientes | Rotas geradas e salvas | ausência de pedidos, veículos ou locais ativos | Erro: dados insuficientes para rotas |

## **listar\_rotas\_com\_detalhes()**

| **Entradas - Classes** | **Válidas** | **Resultado Esperado** | **Inválidas** | **Resultado Esperado** |
| --- | --- | --- | --- | --- |
| Arquivo de rotas | rotas salvas existentes | Detalhes da rota exibidos | arquivo vazio ou inexistente | Mensagem: Nenhuma rota encontrada |

– todos os casos de teste deram certo, e o script de teste de cada está presente na pasta de tests