

# Faculdade Impacta Tecnologia

Projeto de Software (5º semestre)

**Curso:** Análise e Desenvolvimento de Sistemas

**Turma:** 5A - EAD

**Matéria:** Software Product: Analysis, Specification, Project & Implementation

**Professor:** Antonio de Oliveira Dias

## **ALUNOS PARTICIPANTES**

Ana Carollyne Guimarães de Souza	<b>RA:</b> 2301929
Luana Ramos de Almeida	<b>RA:</b> 2302018
Natã Pedro	<b>RA:</b> 2301849
Pedro Pescarole	<b>RA:</b> 2301635

## **SOBRE O PROJETO**

**Nome:** Moveat

**Resumo:** Esse projeto busca oferecer uma experiência prática para aqueles que praticam atividades físicas. A plataforma permite que pessoais trainers e nutricionistas cadastrem seus usuários e associem planos de treino e alimentação personalizados.

Ao fazer login, os usuários encontram tudo o que precisam para seguir suas rotinas de treino e alimentação. O objetivo da plataforma é trazer praticidade, unindo tudo em um lugar só.

## **PROVA OFICIAL – FUNCIONALIDADE PLANO ALIMENTAR (CRUD) | DASHBOARD**

### **DO PACIENTE | REFINAMENTO**

**Objetivo:** A quarta e última sprint do projeto Moveat teve como objetivo a implementação das funcionalidades de gerenciamento de planos alimentares, dashboard do paciente e o refinamento do sistema.

Agora, além de gerenciar seus pacientes, o nutricionista pode:

- Criar planos alimentares personalizados para cada paciente;
- Editar o plano alimentar de um paciente;
- Remover o plano alimentar de um paciente.

O paciente consegue visualizar seu plano alimentar através do dashboard. Não é possível haver mais de um plano alimentar cadastrado.

**Nota importante:** o principal objetivo do nosso projeto era unir nutricionistas e personal trainers em uma única plataforma, para que eles pudessem gerenciar seus pacientes/alunos. Infelizmente, com o curto prazo de entrega e uma funcionalidade difícil como um plano alimentar, decidimos remover toda parte relacionada ao personal trainer.

O projeto Moveat foi **concluído** apenas para nutricionistas e pacientes.

**Link do board do projeto:**

<https://github.com/orgs/Moveat-Fit/projects/9>

**Link do vídeo apresentando a funcionalidade (Youtube):**

[https://www.youtube.com/watch?v=Am\\_IUT3QKMw](https://www.youtube.com/watch?v=Am_IUT3QKMw)

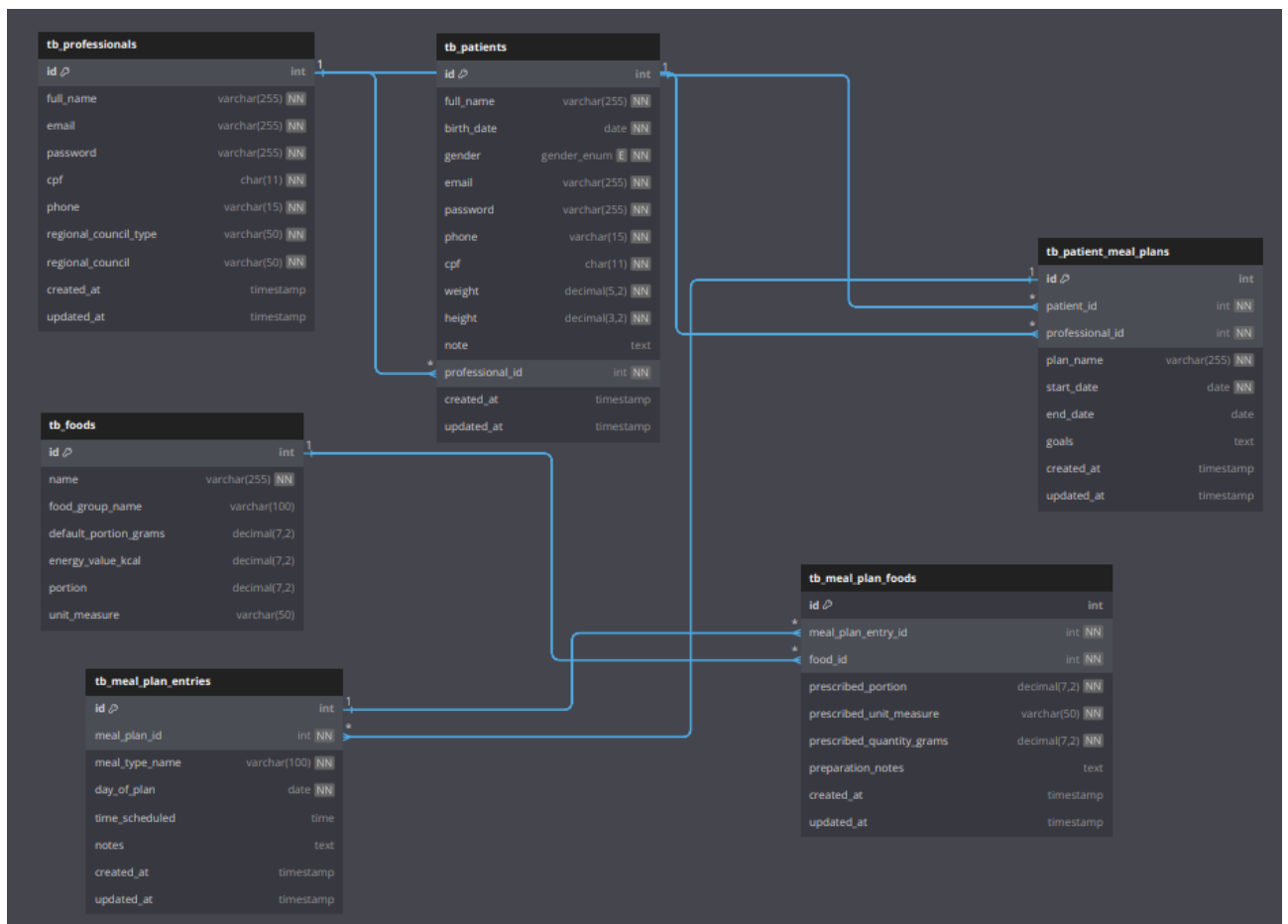
**Link secundário do vídeo (Google Drive):**

<https://drive.google.com/file/d/1WfYsi6VqFo93RsU9JcZKqnHhKOr5MNGe/view?usp=sharing>

**Link do código-fonte (GitHub):**

<https://github.com/orgs/Moveat-Fit/repositories>

## I. DIAGRAMA ENTIDADE RELACIONAMENTO (DER)



## II. REFINAMENTO DO SISTEMA

### 1. REFINAMENTOS REALIZADOS NO FRON-TEND

Foram feitos diversos aprimoramentos na estrutura do front-end para a entrega final do projeto:

- Organização dos serviços de API: As chamadas de API foram separadas em arquivos distintos, de acordo com o domínio (autenticação, paciente, profissional e plano alimentar), facilitando a manutenção e a escalabilidade do código.
- Tipagem completa: Todas as interfaces e funções passaram a utilizar tipagem TypeScript, garantindo maior segurança e clareza no desenvolvimento, algo que não estava presente nas primeiras versões.
- Componentização: Diversas telas que estavam concentradas em um único arquivo foram divididas em componentes menores e reutilizáveis, melhorando a organização e a reutilização do código.
- Uso de prop drilling: Foi implementado o prop drilling para passar dados entre componentes de forma mais controlada, substituindo abordagens anteriores menos estruturadas.
- Context API: A aplicação passou a utilizar contexto para armazenar dados globais, evitando chamadas desnecessárias à API e otimizando a experiência do usuário.

### 2. REFINAMENTOS REALIZADOS NO QA

Foram realizados diversos aprimoramentos na estrutura dos testes automatizados para a entrega final do projeto:

- Organização dos fluxos de teste: Os testes foram separados por fluxos distintos, abrangendo cenários do nutricionista, do paciente e validações específicas das regras de negócio, como a criação de senhas.
- Estrutura em Page Objects: Cada tela do sistema possui um page object dedicado, organizados em pastas temáticas (Autenticação,

DashboardNutricionista, DashboardPaciente e PlanoAlimentar), facilitando a manutenção e a reutilização dos comandos.

- Comando customizado de login: Foi criado um comando personalizado para login, evitando a repetição de lógica em diferentes arquivos de teste e tornando o processo mais eficiente.
- Geração de dados dinâmicos: A utilização da biblioteca Faker permite o cadastro de pacientes com dados aleatórios, prevenindo conflitos e garantindo maior robustez nos testes.
- Cobertura de cenários negativos: Foram implementados testes para validar o tratamento de erros e mensagens de campos obrigatórios, reforçando a confiabilidade das regras de negócio da aplicação.

### III. RELATÓRIO DO TESTE DE API

**Responsável QA:** Ana Carollyne Guimarães de Souza

**Ferramenta de Teste:** Postman (v11.36.3)

**Data dos testes:** 26/05/2025 a 28/05/2025

#### 3. ENDPOINTS TESTADOS (TESTE DE REGRESSÃO)

##### 3.1. AUTENTICAÇÃO

Endpoint	Método HTTP	Descrição	Status Esperado	Resultado
/register	POST	Registra um nutricionista	201 CREATED	Passou
/professional	POST	Loga um nutricionista	200 OK	Passou
/patient	POST	Loga um paciente	200 OK	Passou

### 3.2. DASHBOARD NUTRICIONISTA: GERENCIAMENTO DE PACIENTES

Endpoint	Método HTTP	Descrição	Status Esperado	Resultado
/register/patient	POST	Registra um paciente	201 CREATED	<b>Passou</b>
/register/patient/<patient_id>	PUT	Atualiza dados de um paciente	200 OK	<b>Passou</b>
/deletePatient/<patient_id>	DELETE	Remove registro de um paciente	200 OK	<b>Passou</b>
/patients/<id_nutricionista>	GET	Lista pacientes	200 OK	<b>Passou</b>
/patient/<patient_id>	GET	Lista dados de um paciente	200 OK	<b>Passou</b>


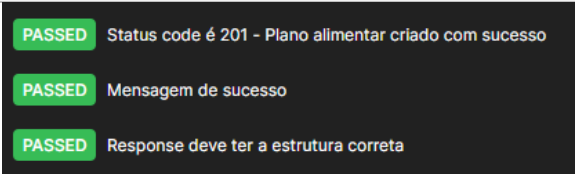
#### 4. ENDPOINTS TESTADOS (TESTES MANUAIS)

##### 4.1. DASHBOARD NUTRICIONISTA: PLANOS ALIMENTARES

Endpoint	Método HTTP	Descrição	Status Esperado	Resultado
/api/meal-plans	POST	Cria plano alimentar de um paciente	201 CREATED	<b>Passou</b>
/api/meal-plans/<patient_id>	GET	Detalha plano alimentar de um paciente	200 OK	<b>Passou</b>
/api/meal-plans/<id_plano_alimentar>	PUT	Atualiza plano alimentar de um paciente	200 OK	<b>Passou</b>
/api/meal-plans/<id_plano_alimentar>	DELETE	Remove plano alimentar de um paciente	200 OK	<b>Passou</b>
/api/foods	GET	Lista alimentos cadastrados	200 OK	<b>Passou</b>

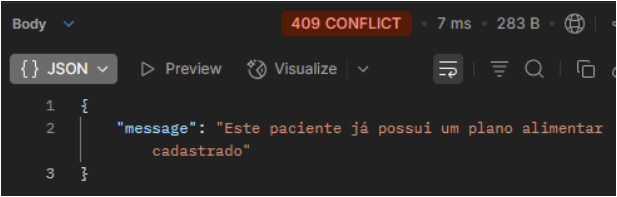
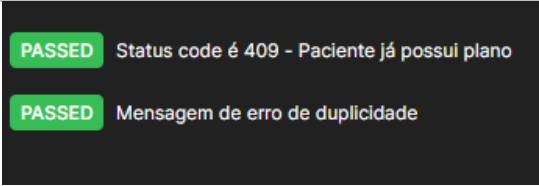
5. CRIAR PLANO ALIMENTAR (POST /api/meal-plans)

5.1. CENÁRIO POSITIVO: CRIA PLANO ALIMENTAR

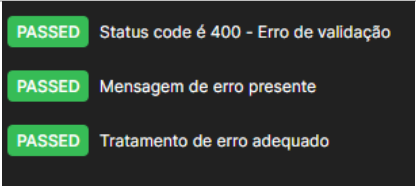
Status Code esperado	201	
Resultado esperado	Mensagem: “Plano alimentar criado com sucesso”	
Resultado obtido		
Corpo da requisição	<pre>{   "patient_id": 60,   "plan_name": "Plano Exemplo",   "start_date": "2024-06-01",   "end_date": "2024-06-07",   "goals": "Perder peso",   "entries": [     {       "meal_type_name": "Café da manhã",       "day_of_plan": "2024-06-01",       "time_scheduled": "08:00",       "notes": "Sem açúcar",       "foods": [         {           "food_name": "Pão francês",           "prescribed_quantity": 1.0,           "unit_measure": "unidade",           "energy_value_kcal": 150,           "preparation_notes": "Servir fresco"         },         {           "food_name": "Requeijão cremoso light",           "prescribed_quantity": 1.0,           "unit_measure": "colher de sopa",           "energy_value_kcal": 80         }       ]     }   ] }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	



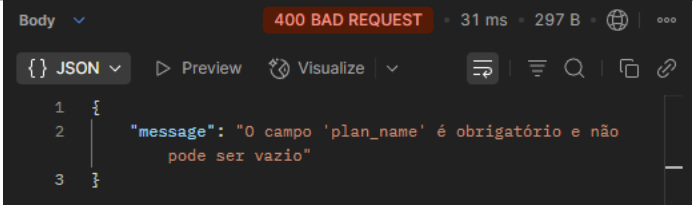
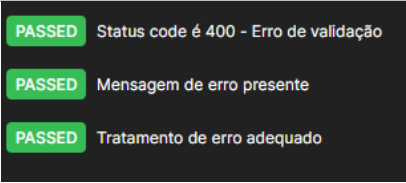
5.2. CENÁRIO NEGATIVO: TENTATIVA DE CRIAR OUTRO PLANO PARA O MESMO PACIENTE

Status Code esperado	409	
Resultado esperado	Mensagem: “Este paciente já possui um plano alimentar cadastrado”	
Resultado obtido		
Corpo da requisição	<pre>{   "patient_id": 60,   "plan_name": "Plano Exemplo II",   "start_date": "2024-06-01",   "end_date": "2024-06-07",   "goals": "Perder peso",   "entries": [     {       "meal_type_name": "Café da manhã",       "day_of_plan": "2024-06-01",       "time_scheduled": "08:00",       "notes": "Sem açúcar",       "foods": [         {           "food_name": "Pão francês",           "prescribed_quantity": 1.0,           "unit_measure": "unidade",           "energy_value_kcal": 150,           "preparation_notes": "Servir fresco"         },         {           "food_name": "Requeijão cremoso light",           "prescribed_quantity": 1.0,           "unit_measure": "colher de sopa",           "energy_value_kcal": 80         }       ]     }   ] }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	

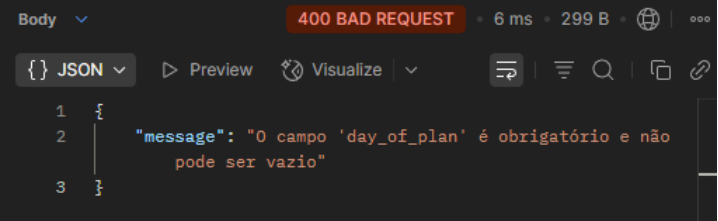
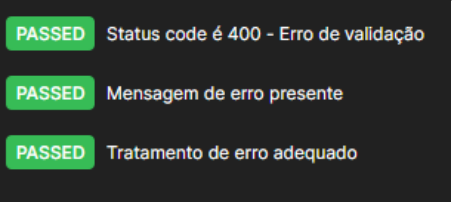
5.3. CENÁRIO NEGATIVO: TENTATIVA DE CRIAR UM PLANO COM ALIMENTO INVÁLIDO

Status Code esperado	400	
Resultado esperado	Mensagem: “Alimento <nome-alimento> não encontrado no banco de dados”	
Resultado obtido		
Corpo da requisição	<pre>{   "patient_id": 58,   "plan_name": "Plano de teste",   "start_date": "2024-06-01",   "end_date": "2024-06-07",   "goals": "Perder peso",   "entries": [     {       "meal_type_name": "Café da manhã",       "day_of_plan": "2024-06-01",       "time_scheduled": "08:00",       "notes": "Sem açúcar",       "foods": [         {           "food_name": "Alface", // alimento não cadastrado no banco           "prescribed_quantity": 1.0,           "unit_measure": "unidade",           "energy_value_kcal": 150,           "preparation_notes": "Servir fresco"         },         {           "food_name": "Requeijão cremoso light",           "prescribed_quantity": 1.0,           "unit_measure": "colher de sopa",           "energy_value_kcal": 80         }       ]     }   ] }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	

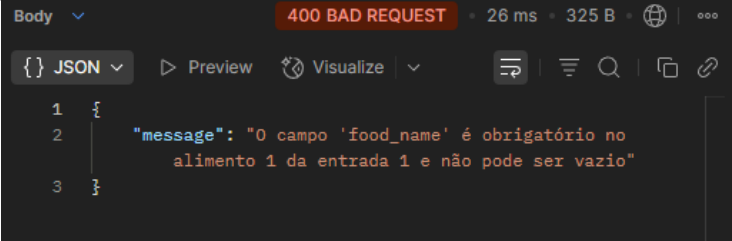
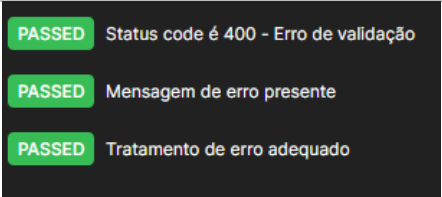
5.4. CENÁRIO NEGATIVO: TENTATIVA DE CRIAR UM PLANO COM CAMPOS OBRIGATÓRIOS EM BRANCO

Status Code esperado	400
Resultado esperado	Mensagem: “O campo <nome-campo> é obrigatório e não pode ser vazio”
Resultado obtido	
Corpo da requisição	<pre>{   "patient_id": 58,   "plan_name": "", // campo obrigatório vazio   "start_date": "2024-06-01",   "end_date": "2024-06-07",   "goals": "Perder peso",   "entries": [     {       "meal_type_name": "Café da manhã",       "day_of_plan": "2024-06-01",       "time_scheduled": "08:00",       "notes": "Sem açúcar",       "foods": [         {           "food_name": "Pão francês",           "prescribed_quantity": 1.0,           "unit_measure": "unidade",           "energy_value_kcal": 150,           "preparation_notes": "Servir fresco"         },         {           "food_name": "Requeijão cremoso light",           "prescribed_quantity": 1.0,           "unit_measure": "colher de sopa",           "energy_value_kcal": 80         }       ]     }   ] }</pre>
Evidência	
Implementação	Manual
Status do teste	Passou

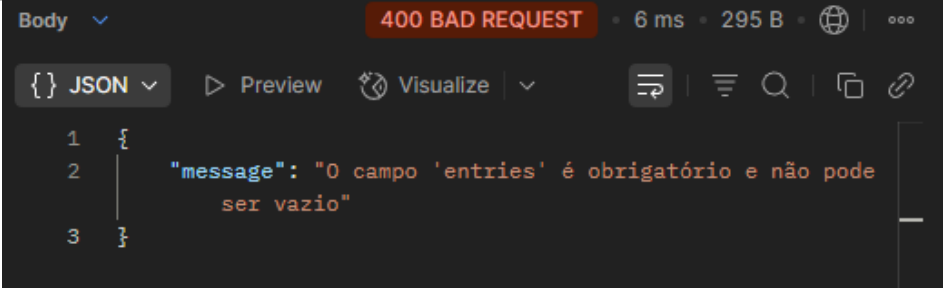
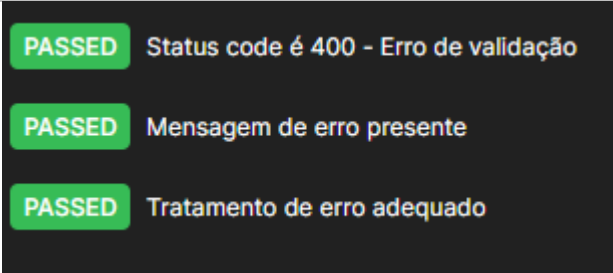
5.5. CENÁRIO NEGATIVO: TENTATIVA DE CRIAR UM PLANO COM QUALQUER CAMPO OBRIGATÓRIO DAS REFEIÇÕES NÃO PREENCHIDO

Status Code esperado	400	
Resultado esperado	Mensagem: “O campo <nome-campo> é obrigatório e não pode ser vazio”	
Resultado obtido		
Corpo da requisição	<pre>{   "patient_id": 58,   "plan_name": "Plano Exemplo",   "start_date": "2024-06-01",   "end_date": "2024-06-07",   "goals": "Perder peso",   "entries": [     {       "meal_type_name": "Café da manhã",       "day_of_plan": "", // Campo obrigatório       "time_scheduled": "08:00",       "notes": "Sem açúcar",       "foods": [         {           "food_name": "Pão francês",           "prescribed_quantity": 1.0,           "unit_measure": "unidade",           "energy_value_kcal": 150,           "preparation_notes": "Servir fresco"         },         {           "food_name": "Requeijão cremoso light",           "prescribed_quantity": 1.0,           "unit_measure": "colher de sopa",           "energy_value_kcal": 80         }       ]     }   ] }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	

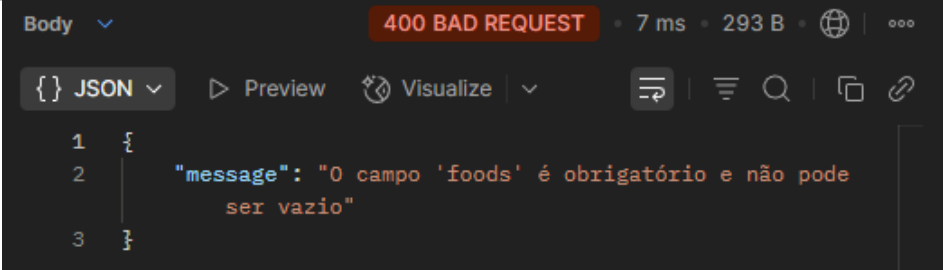
5.6. CENÁRIO NEGATIVO: TENTATIVA DE CRIAR UM PLANO COM QUALQUER CAMPO OBRIGATÓRIO DOS ALIMENTOS NÃO PREENCHIDO

Status Code esperado	400	
Resultado esperado	Mensagem: “O campo <nome-campo> é obrigatório e não pode ser vazio”	
Resultado obtido		
Corpo da requisição	<pre>{   "patient_id": 58,   "plan_name": "Plano Exemplo",   "start_date": "2024-06-01",   "end_date": "2024-06-07",   "goals": "Perder peso",   "entries": [     {       "meal_type_name": "Café da manhã",       "day_of_plan": "2024-06-01",       "time_scheduled": "08:00",       "notes": "Sem açúcar",       "foods": [         {           "food_name": "", // Campo obrigatório           "prescribed_quantity": 1.0,           "unit_measure": "unidade",           "energy_value_kcal": 150,           "preparation_notes": "Servir fresco"         },         {           "food_name": "Requeijão cremoso light",           "prescribed_quantity": 1.0,           "unit_measure": "colher de sopa",           "energy_value_kcal": 80         }       ]     }   ] }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	

5.7. CENÁRIO NEGATIVO: TENTATIVA DE CRIAR UM PLANO SEM ADICIONAR REFEIÇÕES

Status Code esperado	400	
Resultado esperado	Mensagem: “O campo <nome-campo> é obrigatório e não pode ser vazio”	
Resultado obtido		
Corpo da requisição	<pre>{   "patient_id": 58,   "plan_name": "Plano de Teste",   "start_date": "2024-06-01",   "end_date": "2024-06-07",   "goals": "Perder peso",   "entries": [] // lista obrigatória }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	

5.8. CENÁRIO NEGATIVO: TENTATIVA DE CRIAR UM PLANO SEM ADICIONAR UM ALIMENTO

Status Code esperado	400	
Resultado esperado	Mensagem: “O campo <nome-campo> é obrigatório e não pode ser vazio”	
Resultado obtido		
Corpo da requisição	<pre>{   "patient_id": 58,   "plan_name": "Plano de teste",   "start_date": "2024-06-01",   "end_date": "2024-06-07",   "goals": "Perder peso",   "entries": [     {       "meal_type_name": "Café da manhã",       "day_of_plan": "2024-06-01",       "time_scheduled": "08:00",       "notes": "Sem açúcar",       "foods": [] // lista vazia     }   ] }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	

6. DETALHAR PLANO ALIMENTAR (GET /api/meal-plans/<patient\_id>)

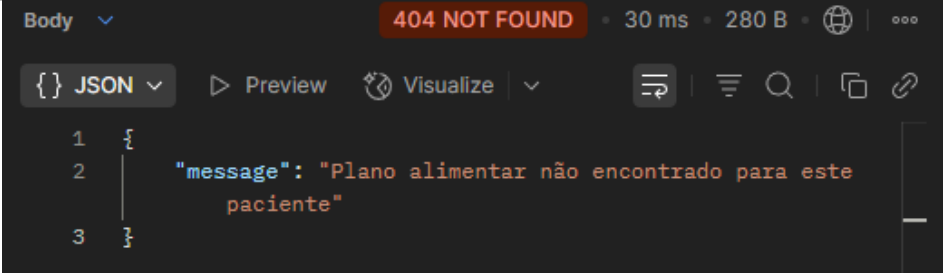
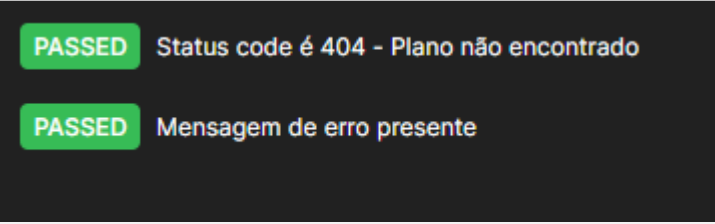
6.1. CENÁRIO POSITIVO: DETALHAR PLANO ALIMENTAR DE UM PACIENTE

Status Code esperado	200
Resultado esperado	Plano alimentar detalhado
Resultado obtido	<div><div>Body</div><div>200 OK • 26 ms • 1.52 KB</div><div><div>{ } JSON</div><div>Preview</div><div>Visualize</div></div><div><pre>1  { 2    "meal_plan": { 3      "id": 22, 4      "patient_id": 60, 5      "professional_id": 24, 6      "plan_name": "Plano Exemplo", 7      "start_date": "2024-06-01", 8      "end_date": "2024-06-07", 9      "goals": "Perder peso", 10     "created_at": "2025-05-27 23:44:36", 11     "updated_at": "2025-05-27 23:44:36", 12     "entries": [ 13       { 14         "id": 20, 15         "meal_type_name": "Café da manhã", 16         "day_of_plan": "2024-06-01", 17         "time_scheduled": "08:00", 18         "notes": "Sem açúcar", 19         "foods": [ 20           { 21             "id": 1, 22             "food_id": 6, 23             "prescribed_quantity_grams": 1.0, 24             "prescribed_portion": 1.0, 25             "prescribed_unit_measure": 26               "unidade", 27             "preparation_notes": "Servir 28               fresco" 29           }, 30           { 31             "id": 2, 32             "food_id": 26, 33             "prescribed_quantity_grams": 1.0, 34             "prescribed_portion": 1.0, 35             "prescribed_unit_measure": "colher 36               de sopa", 37             "preparation_notes": null 38           } 39         ] 40       } 41     ] 42   } 43 }</pre></div></div>



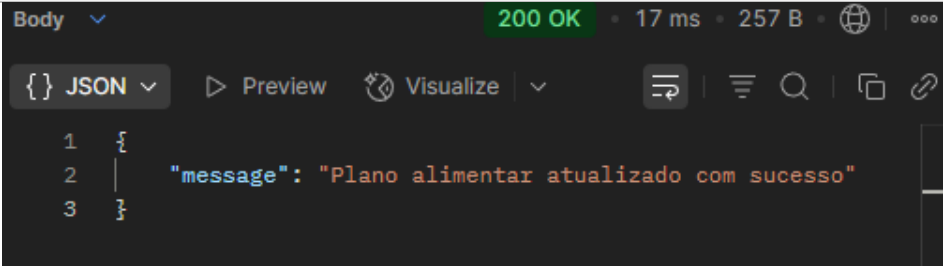
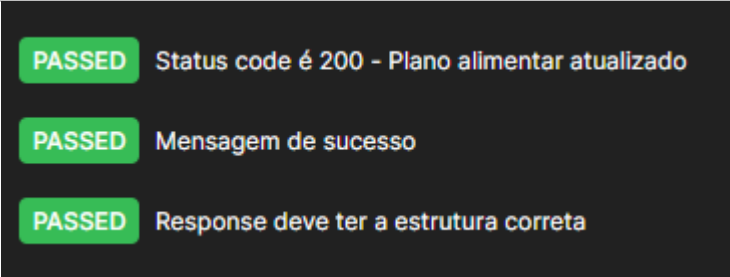
Evidência	<div><div>PASSED</div> Status code é 200 - Plano alimentar encontrado</div> <div><div>PASSED</div> 'meal_plan' deve ter a estrutura correta</div> <div><div>PASSED</div> 'entries' e 'foods' devem ter a estrutura correta</div>
Implementação	Manual
Status do teste	Passou

6.2. CENÁRIO NEGATIVO: TENTATIVA DE DETALHAR UM PLANO ALIMENTAR INEXISTENTE

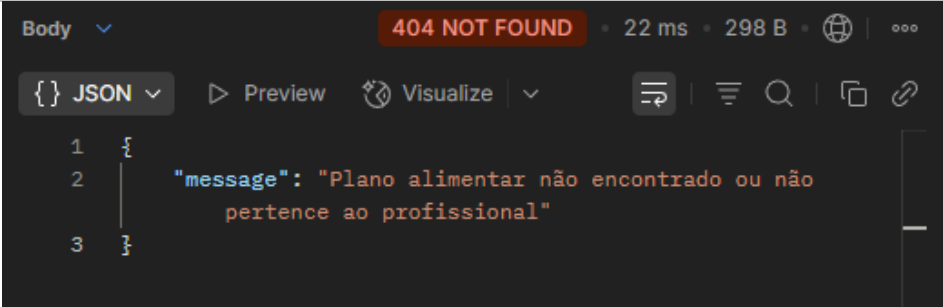
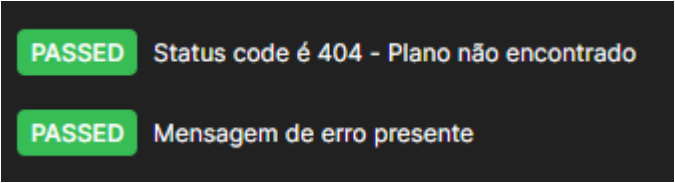
Status Code esperado	400
Resultado esperado	Mensagem: “Plano alimentar não encontrado para este paciente”
Resultado obtido	 <p>The screenshot shows a REST client interface with a dropdown menu set to 'JSON'. The response status is '404 NOT FOUND' with a duration of '30 ms' and a size of '280 B'. The JSON body is displayed as follows:</p> <pre>1 { 2   "message": "Plano alimentar não encontrado para este 3   paciente"</pre>
Evidência	 <p>The screenshot shows two test results, both marked as 'PASSED' in green boxes:</p> <ul style="list-style-type: none"><li>PASSED Status code é 404 - Plano não encontrado</li><li>PASSED Mensagem de erro presente</li></ul>
Implementação	Manual
Status do teste	Passou

7. ATUALIZAR PLANO ALIMENTAR (PUT /api/meal-plans/<id\_plano>)

7.1. CENÁRIO POSITIVO: ATUALIZAÇÃO BEM-SUCEDIDA

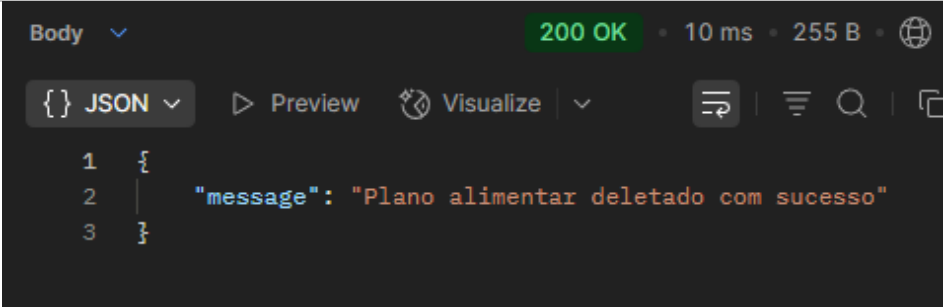
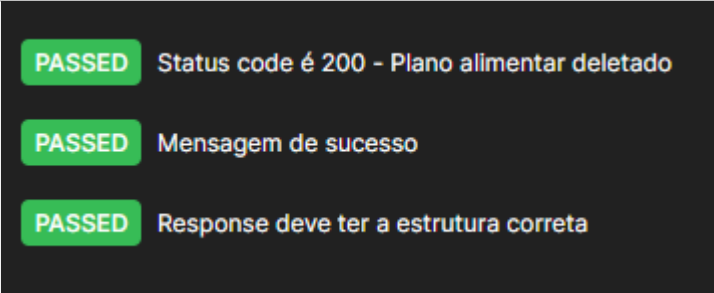
Status Code esperado	200	
Resultado esperado	Mensagem: “Plano alimentar atualizado com sucesso”	
Resultado obtido		
Corpo da requisição	<pre>{   "plan_name": "Plano Atualizado",   "start_date": "2025-10-03",   "end_date": "2026-02-02",   "goals": "Novas metas" }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	

7.2. CENÁRIO NEGATIVO: TENTATIVA DE ATUALIZAR UM PLANO INEXISTENTE

Status Code esperado	404
Resultado esperado	Mensagem: “Plano alimentar não encontrado ou não pertence ao profissional”
Resultado obtido	
Corpo da requisição	<pre>{   "plan_name": "Plano Atualizado",   "start_date": "2025-10-03",   "end_date": "2026-02-02",   "goals": "Novas metas" }</pre>
Evidência	
Implementação	Manual
Status do teste	Passou

8. REMOVER PLANO ALIMENTAR (DELETE /api/meal-plans/< id\_plano>)

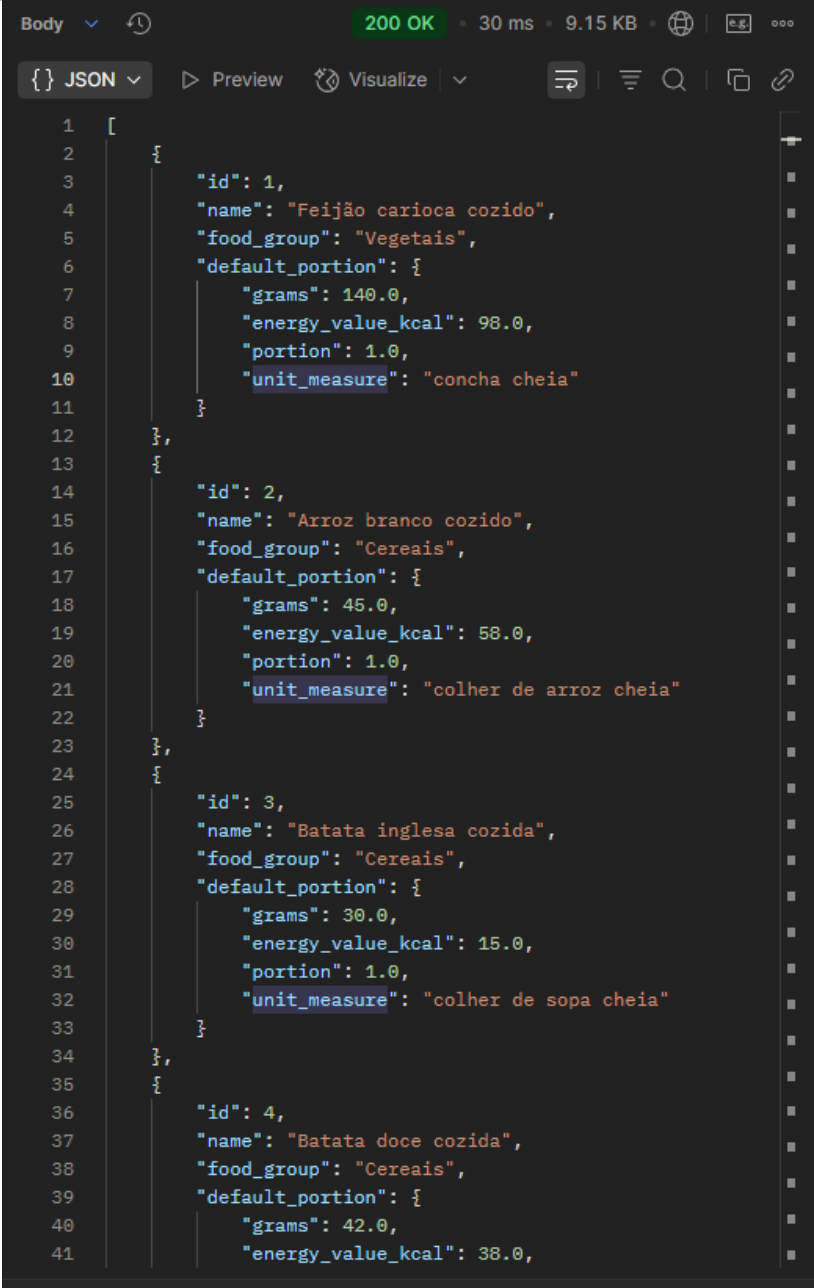
8.1. CENÁRIO POSITIVO: REMOVER PLANO ALIMENTAR DE UM PACIENTE

Status Code esperado	200	
Resultado esperado	Mensagem: “Plano alimentar deletado com sucesso”	
Resultado obtido		
Evidência		
Implementação	Manual	
Status do teste	Passou	

8.2. CENÁRIO NEGATIVO: TENTATIVA DE REMOVER UM PLANO ALIMENTAR INEXISTENTE

Status Code esperado	404
Resultado esperado	Mensagem: “Plano alimentar não encontrado ou não pertence ao profissional”
Resultado obtido	
Evidência	
Implementação	Manual
Status do teste	Passou

9. LISTAR ALIMENTOS CADASTRADOS NO BANCO (GET /api/foods)

Status Code esperado	200
Resultado esperado	Mensagem: “Plano alimentar deletado com sucesso”
Resultado obtido	 <pre>1  [ 2    { 3      "id": 1, 4      "name": "Feijão carioca cozido", 5      "food_group": "Vegetais", 6      "default_portion": { 7        "grams": 140.0, 8        "energy_value_kcal": 98.0, 9        "portion": 1.0, 10       "unit_measure": "concha cheia" 11     } 12   }, 13   { 14     "id": 2, 15     "name": "Arroz branco cozido", 16     "food_group": "Cereais", 17     "default_portion": { 18       "grams": 45.0, 19       "energy_value_kcal": 58.0, 20       "portion": 1.0, 21       "unit_measure": "colher de arroz cheia" 22     } 23   }, 24   { 25     "id": 3, 26     "name": "Batata inglesa cozida", 27     "food_group": "Cereais", 28     "default_portion": { 29       "grams": 30.0, 30       "energy_value_kcal": 15.0, 31       "portion": 1.0, 32       "unit_measure": "colher de sopa cheia" 33     } 34   }, 35   { 36     "id": 4, 37     "name": "Batata doce cozida", 38     "food_group": "Cereais", 39     "default_portion": { 40       "grams": 42.0, 41       "energy_value_kcal": 38.0,</pre>
Implementação	Manual
Status do teste	Passou

## 10. BUGS ENCONTRADOS DURANTE OS TESTES DE API

TESTES DE API		
ID	DESCRIÇÃO	STATUS
<a href="#">BUG-API-001</a>	Endpoint /register/patient não retorna o id	Corrigido
<a href="#">BUG-API-002</a>	Endpoint /register/patient retorna id como 0	Corrigido
<a href="#">BUG-API-003</a>	Endpoint /patient/<patient_id>/update com mensagem de erro incorreta	Corrigido
<a href="#">BUG-API-004</a>	Endpoint /patients/<professional_id> não retorna array adequado	Corrigido



## IV. RELATÓRIO DO TESTE DE SISTEMA (E2E)

**Responsável QA:** Ana Carollyne Guimarães de Souza

**Sistema Operacional:** Windows 11 Pro

**Data dos testes:** 04/06/2025 – 08/06/2025

### 11. OBJETIVO

Validar que o sistema permite que o **nutricionista** crie e gerencie os planos alimentares de seus pacientes.

Validar que o sistema permite que o **paciente** visualize seu plano alimentar.

### 12. PRÉ-CONDIÇÕES

- O nutricionista está autenticado no sistema e com pacientes cadastrados;
- O paciente possui um plano alimentar criado pelo seu nutricionista.

### 13. CASOS DE TESTE EXECUTADOS

Para melhor organização e rastreabilidade, os casos de teste foram criados na plataforma **Qase**. Todos foram executados manualmente e automatizados em seguida.

SPRINT 4: PLANO ALIMENTAR | DASHBOARD DO PACIENTE | REFINAMENTO + ✎ 📄 🗑

+ Create quick test

Gerenciamento de planos alimentares (Nutricionista) + ✎ 📄 🗑

+ Create quick test

Criar plano alimentar + ✎ 📄 🗑

- ↑ 🧑 MOVEAT-29 Criar plano alimentar de um paciente
- ↑ 🧑 MOVEAT-30 Validar tratamento de erro para plano alimentar sem campos obrigatórios preenchidos
- ↑ 🧑 MOVEAT-31 Validar tratamento de erro para plano alimentar sem refeições
- ↑ 🧑 MOVEAT-32 Validar tratamento de erro para refeição sem alimentos no plano alimentar

+ Create quick test

Remover plano alimentar + ✎ 📄 🗑

- ↑ 🧑 MOVEAT-33 Remover plano alimentar de um paciente
- 🧑 MOVEAT-34 Cancelar a remoção do plano alimentar
- ↓ 🧑 MOVEAT-35 Validar tratamento de erro ao tentar remover um plano alimentar inexistente

+ Create quick test

Editar plano alimentar + ✎ 📄 🗑

- 🧑 MOVEAT-36 Editar plano alimentar de um paciente
- ↑ 🧑 MOVEAT-37 Validar tratamento de erro ao tentar salvar com campos obrigatórios
- ↑ 🧑 MOVEAT-38 Validar tratamento de erro ao tentar salvar sem refeições
- ↑ 🧑 MOVEAT-39 Validar tratamento de erro ao tentar salvar uma refeição sem alimentos

+ Create quick test

Visualizar plano alimentar (Paciente) + ✎ 📄 🗑

- ↑ 🧑 MOVEAT-40 Visualizar plano alimentar
- ↑ 🧑 MOVEAT-41 Validar tratamento de erro ao tentar editar o plano como paciente logado
- ↑ 🧑 MOVEAT-42 Validar tratamento de erro ao tentar acessar sem um plano cadastrado

## 14. EXECUÇÃO DOS TESTES

Para garantir o funcionamento correto do sistema, os testes foram divididos em suítes e executados separadamente. Depois de corrigidas e retestadas, todas as suítes de teste foram executadas em uma única *run* e de forma automatizada.

### 14.1. SUÍTE DE TESTE: CRIAR PLANO ALIMENTAR ([Veja o relatório aqui](#))

#### 👤 Teste Funcional: Sprint 4 | Criar plano alimentar (04/06/2025)

Test cases

Defects

Team stats

Timeline

Search...

By all fields

Add filter

Assign

Assign to me

Unassign

Select all

0 cases

0 suites

Criar plano alimentar

SPRINT 4: PLANO ALIMENTAR | DASHBOARD DO PACIENTE | REFINAMENTO > Gerenciamento de planos alimentares (Nutricionista) > Criar plano alimentar

	ID	MEMBER	STATUS	TITLE
<div><div><div></div><div>↑</div><div></div></div></div>	29	Ana Carolyn	Passed	Criar plano alimentar de um paciente
<div><div><div></div><div>↑</div><div></div></div></div>	30	Ana Carolyn	Passed	Validar tratamento de erro para plano alimentar sem campos obrigatórios preenchidos
<div><div><div></div><div>↑</div><div></div></div></div>	31	Ana Carolyn	Passed	Validar tratamento de erro para plano alimentar sem refeições
<div><div><div></div><div>↑</div><div></div></div></div>	32	Ana Carolyn	Passed	Validar tratamento de erro para refeição sem alimentos no plano alimentar

### 14.2. SUÍTE DE TESTE: VISUALIZAR PLANO ALIMENTAR ([Veja o relatório aqui](#))

#### - 👤 Teste Funcional: Sprint 4 | Visualizar plano alimentar (05/06/2025)

est cases

Defects

Team stats

Timeline

Search...

By all fields

▼

Add filter

Assign

Assign to me

Select all

0 cases

|

0 suites

▼

Visualizar plano alimentar (Paciente)

SPRINT 4: PLANO ALIMENTAR | DASHBOARD DO PACIENTE | REFINAMENTO > Visualizar plano alimentar (Paciente)

	ID	MEMBER	STATUS	TITLE
<div><div></div><div>↑</div><div><div></div><div></div></div></div>	40	<div><div></div><div>Ana Carolyn</div></div>	<div>Passed</div>	Visualizar plano alimentar
<div><div></div><div>↑</div><div><div></div><div></div></div></div>	42	<div><div></div><div>Ana Carolyn</div></div>	<div>Passed</div>	Validar tratamento de erro ao tentar acessar sem um plano cadastrado

14.3. SUÍTE DE TESTE: ATUALIZAR PLANO ALIMENTAR ([Veja o relatório aqui](#))

✓

Editar plano alimentar

6m 45s

SPRINT 4: PLANO ALIMENTAR | DASHBOARD DO PACIENTE | REFINAMENTO > Gerenciamento de planos alimentares (Nutricionista) > Editar plano

	ID	MEMBER	STATUS	TITLE
<div><div></div><div></div></div>	36	Ana Carolyn	Passed	Editar plano alimentar de um paciente
<div><div></div><div></div></div>	37	Ana Carolyn	Passed	Validar tratamento de erro ao tentar salvar com campos obrigatórios
<div><div></div><div></div></div>	38	Ana Carolyn	Passed	Validar tratamento de erro ao tentar salvar sem refeições
<div><div></div><div></div></div>	39	Ana Carolyn	Passed	Validar tratamento de erro ao tentar salvar uma refeição sem alimentos

14.4. SUÍTE DE TESTE: REMOVER PLANO ALIMENTAR ([Veja o relatório aqui](#))

✓

Remover plano alimentar

1h 55m

SPRINT 4: PLANO ALIMENTAR | DASHBOARD DO PACIENTE | REFINAMENTO > Gerenciamento de planos alimentares (Nutricionista) > Remover plano

	ID	MEMBER	STATUS	TITLE
<div><div></div><div></div></div>	33	Ana Carolyn	Passed	Remover plano alimentar de um paciente
<div><div></div><div></div></div>	34	Ana Carolyn	Passed	Cancelar a remoção do plano alimentar
<div><div></div><div></div></div>	35	Ana Carolyn	Passed	Validar tratamento de erro ao tentar remover um plano alimentar inexistente

## 15. MELHORIAS IDENTIFICADAS DURANTE OS TESTES DE SISTEMA

TESTES DE SISTEMA		
ID	DESCRIÇÃO	STATUS
<a href="#">UPGRADE-CAD.PLANO-001</a>	Limitar uso de " , " no campo "Porção"	Melhoria implementada
<a href="#">UPGRADE-CAD.PLANO-002</a>	Redirecionar o nutricionista após criar um plano alimentar	Melhoria implementada
<a href="#">UPGRADE-VISU.PLANO-001</a>	Nenhuma mensagem é exibida para paciente sem plano alimentar	Melhoria implementada
<a href="#">UPGRADE-VISU.PLANO-002</a>	Nome incompleto do paciente logado	Melhoria implementada

## 16. BUGS IDENTIFICADOS DURANTE OS TESTES DE SISTEMA

TESTES DE SISTEMA		
ID	DESCRIÇÃO	STATUS
<a href="#">BUG-VISU.PLANO-001</a>	Nome do nutricionista é genérico	Corrigido

## V. RELATÓRIO DO TESTE DE REGRESSÃO

**Responsável QA:** Ana Carollyne Guimarães de Souza

**Sistema Operacional:** Windows 11 Pro

**Ferramenta de teste:** Cypress

**Data dos testes:** 07/06/2025

### 17. OBJETIVO

Validar que o sistema construído anteriormente não tenha “quebrado” após as mudanças realizadas. Todos os casos de teste da sprint de **autenticação, dashboard do nutricionista (cadastro e leitura de pacientes) e dashboard do nutricionista (alterar e deletar pacientes)** foram automatizados com cypress.

### 18. RESULTADO DOS TESTES

Os testes automatizados passaram por uma pequena manutenção na captura de botões e validações das mensagens de erro, que foram atualizadas nessa sprint. O teste de regressão confirmou que nenhuma mudança afetou as funcionalidades já desenvolvidas anteriormente.

update-fluxo-nutricionista

tsucarol

Ran 10 mins ago

02:11

main

None

# 13

Overview

Test Results 26


Specs 7

Errors 0

UI Coverage

Accessibility

Properties

 cypress\e2e\Fluxo nutricionista\0-cadastroNutricionista.cy.js

3

00:11

E2E

Windows 10.0.26100


Chrome 137

View Output

Cadastro do nutricionista > Cadastro com dados válidos

Cadastro do nutricionista > Tentativa de cadastro com email existente

Cadastro do nutricionista > Tentativa de cadastro com dados em branco

 cypress\e2e\Fluxo nutricionista\1-loginNutricionista.cy.js

3

00:06

E2E

Windows 10.0.26100


Chrome 137

View Output

Tela de Login > Login com credenciais válidas

Tela de Login > Tentativa de login com email não cadastrado

Tela de Login > Tentativa de login com senha incorreta

 cypress\e2e\Fluxo nutricionista\2-cadastroPaciente.cy.js

6

00:34

E2E

Windows 10.0.26100

Chrome 137

View Output

Dashboard do Nutricionista > Tentativa de cadastro com CPF inválido

Dashboard do Nutricionista > Tentativa de cadastro com email inválido

Dashboard do Nutricionista > Tentativa de cadastro com CPF já cadastrado

Dashboard do Nutricionista > Tentativa de cadastro com email já cadastrado

Dashboard do Nutricionista > Tentativa de cadastro com campos não preenchidos

Dashboard do Nutricionista > Cadastro com dados válidos

📁 cypress\e2e\Fluxo nutricionista\3-leituraPaciente.cy.js

✓ 3 • 00:08 • E2E • Windows 10.0.26100 • Chrome 137 • [View Output](#)

- ✓ Dashboard do Nutricionista - Leitura de pacientes > Pesquisa pelo nome completo **MODIFIED**
- ✓ Dashboard do Nutricionista - Leitura de pacientes > Pesquisa pelo nome parcialmente
- ✓ Dashboard do Nutricionista - Leitura de pacientes > Tentativa de pesquisa por nome inexistente

📁 cypress\e2e\Fluxo nutricionista\4-removerPaciente.cy.js

✓ 2 • 00:05 • E2E • Windows 10.0.26100 • Chrome 137 • [View Output](#)

- ✓ Dashboard do Nutricionista - Remover pacientes > Remove um paciente aleatório
- ✓ Dashboard do Nutricionista - Remover pacientes > Cancela a exclusão de um paciente

📁 cypress\e2e\Fluxo nutricionista\5-atualizarDadosPaciente.cy.js

✓ 5 • 00:16 • E2E • Windows 10.0.26100 • Chrome 137 • [View Output](#)

- ✓ Dashboard do Nutricionista - Atualizar dados cadastrais dos pacientes > Atualiza dados cadastrais de um paciente aleatório
- ✓ Dashboard do Nutricionista - Atualizar dados cadastrais dos pacientes > Cancela a atualização dos dados
- ✓ Dashboard do Nutricionista - Atualizar dados cadastrais dos pacientes > Validar tratamento de erro para email inválido
- ✓ Dashboard do Nutricionista - Atualizar dados cadastrais dos pacientes > Validar tratamento de erro para email já cadastrado
- ✓ Dashboard do Nutricionista - Atualizar dados cadastrais dos pacientes > Validar tratamento de erro para campos em branco

📁 cypress\e2e\Outras validações\validacoesSenha.cy.js

✓ 4 • 00:04 • E2E • Windows 10.0.26100 • Chrome 137 • [View Output](#)

- ✓ Validando tratamento de senha > Deve exibir erro quando a senha não tiver caracteres especiais
- ✓ Validando tratamento de senha > Deve exibir erro quando a senha não tiver letras maiúsculas
- ✓ Validando tratamento de senha > Deve exibir erro quando as senhas não coincidirem
- ✓ Validando tratamento de senha > Deve exibir erro quando a senha tiver menos de 8 caracteres



## VI. RELATÓRIO DO TESTE AUTOMATIZADO (E2E)

**Responsável QA:** Ana Carollyne Guimarães de Souza

**Sistema Operacional:** Windows 11 Pro

**Ferramenta de teste:** Cypress

**Data dos testes:** 08/06/2025

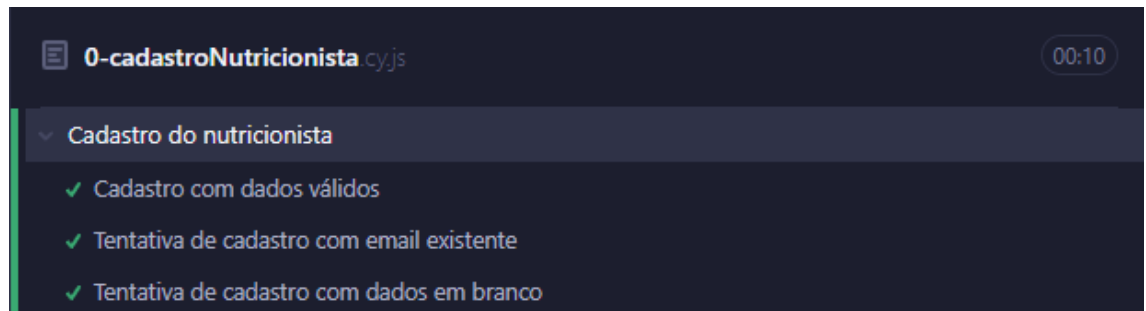
### 19. OBJETIVO

Validar o funcionamento do sistema como um todo através dos testes automatizados. Todas as funcionalidades foram testadas

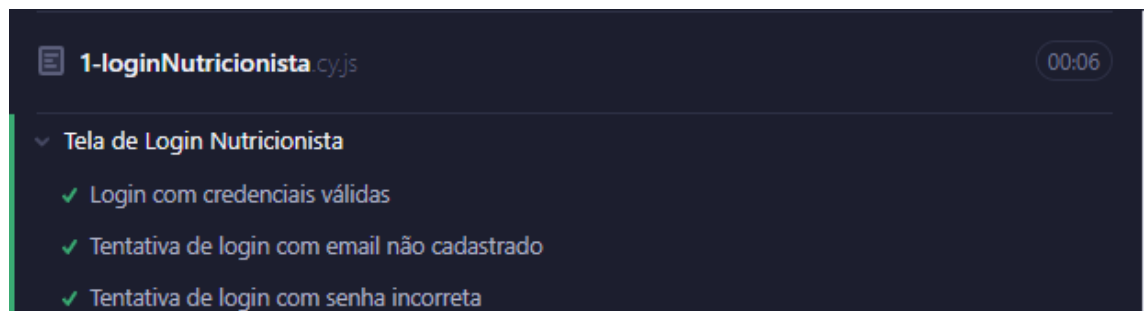
### 20. EXECUÇÃO DOS TESTES

▼	📁	cypress\e2e
▼	📁	Fluxo nutricionista
	📄	0-cadastroNutricionista.cyjs
	📄	1-loginNutricionista.cyjs
	📄	2-cadastroPaciente.cyjs
	📄	3-leituraPaciente.cyjs
	📄	4-removerPaciente.cyjs
	📄	5-atualizarDadosPaciente.cyjs
	📄	6-criarPlanoAlimentar.cyjs
	📄	7-removerPlanoAlimentar.cyjs
	📄	8-editarPlanoAlimentar.cyjs
▼	📁	Fluxo paciente
	📄	1-loginPaciente.cyjs
	📄	2-visualizarPlanoAlimentar.cyjs
▼	📁	Outras validações
	📄	validacoesSenha.cyjs

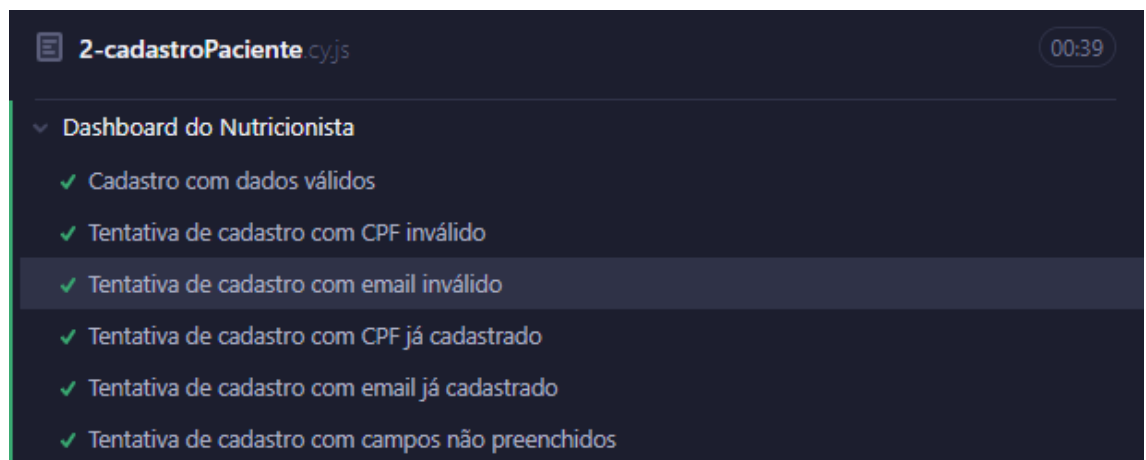
## 20.1. CADASTRAR UM NUTRICIONISTA



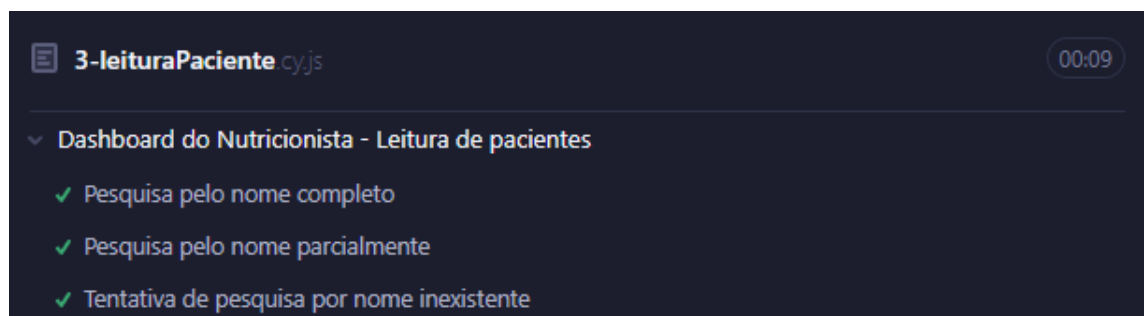
## 20.2. LOGAR UM NUTRICIONISTA



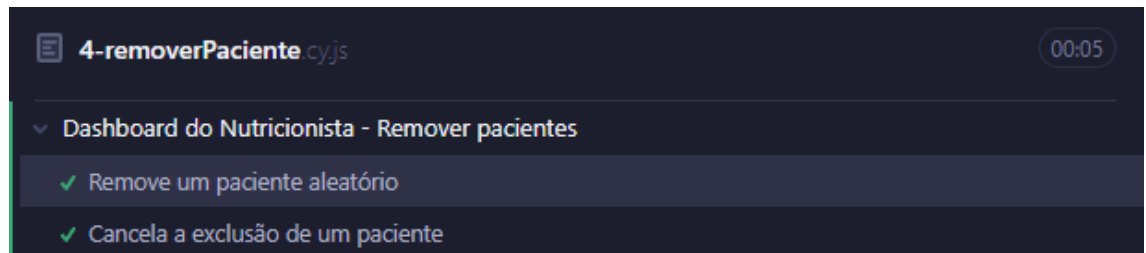
## 20.3. CADASTRAR UM PACIENTE



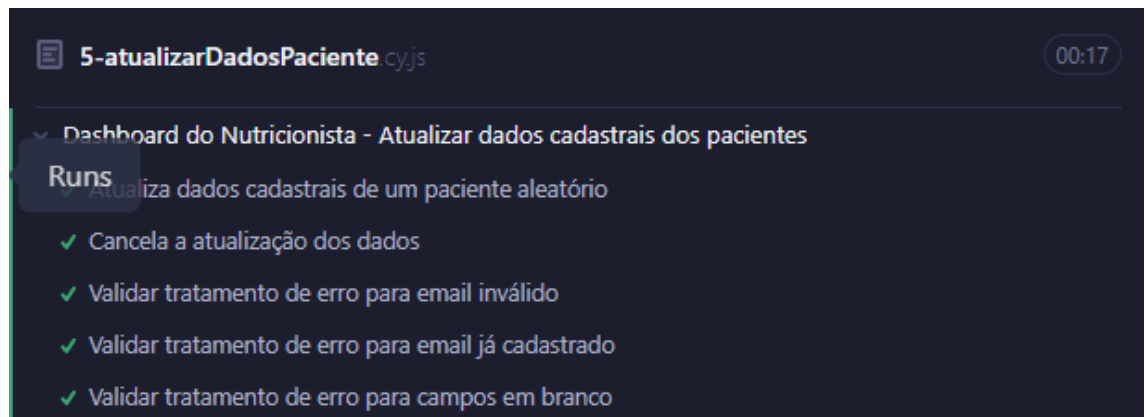
## 20.4. PESQUISAR UM PACIENTE



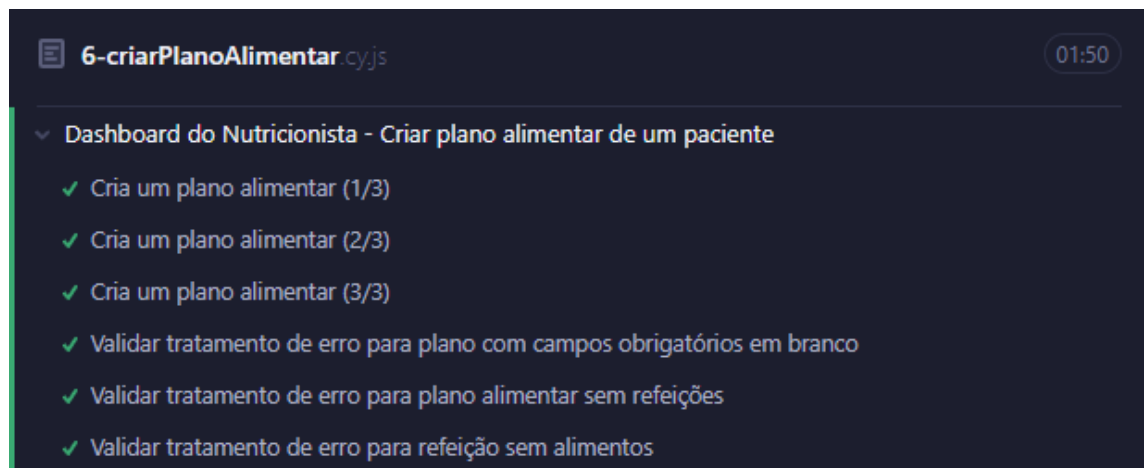
## 20.5. EXCLUIR UM PACIENTE




## 20.6. ATUALIZAR DADOS CADASTRAIS DE UM PACIENTE



## 20.7. CRIAR PLANO ALIMENTAR PARA UM PACIENTE




## 20.8. REMOVER PLANO ALIMENTAR DE UM PACIENTE

 **7-removerPlanoAlimentar** cyjs

Dashboard do Nutricionista - Remover plano alimentar de um paciente

- ✓ Remover plano alimentar
- ✓ Cancelar a exclusão do plano
- ✓ Validar tratamento de erro ao tentar remover um plano inexistente


## 20.9. EDITAR DADOS DE UM PLANO ALIMENTAR

 **8-editarPlanoAlimentar** cyjs 00:13

Dashboard do Nutricionista - Editar plano alimentar de um paciente

- ✓ Editar plano alimentar
- ✓ Validar erro ao tentar salvar edição com campos obrigatórios em branco
- ✓ Validar erro ao tentar salvar edição sem refeições
- ✓ Validar erro ao tentar salvar edição sem alimentos

## 20.10. LOGAR UM PACIENTE

 **1-loginPaciente** cyjs 00:06


Tela de Login Paciente

Runs

com credenciais válidas

- ✓ Tentativa de login com email não cadastrado
- ✓ Tentativa de login com senha incorreta


## 20.11. VISUALIZAR PLANO ALIMENTAR CRIADO POR UM NUTRICIONISTA

 **2-visualizarPlanoAlimentar** cyjs 00:04

Dashboard do Paciente - Visualizar plano alimentar

- ✓ Visualizar plano alimentar
- ✓ Validar tratamento de erro ao tentar acessar sem um plano cadastrado

## 20.12. VALIDAÇÕES DE SENHA

 **validacoesSenha** cyjs

00:05

---

✓ Validando tratamento de senha

- ✓ Deve exibir erro quando a senha tiver menos de 8 caracteres
- ✓ Deve exibir erro quando a senha não tiver caracteres especiais
- ✓ Deve exibir erro quando a senha não tiver letras maiúsculas
- ✓ Deve exibir erro quando as senhas não coincidirem