

# 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.

### **AC 03 – FUNCIONALIDADE DASHBOARD DO NUTRICIONISTA II**

**Objetivo:** A segunda sprint do projeto Moveat teve como objetivo a implementação das funcionalidades de atualização de dados (PUT) e exclusão (DELETE) de pacientes como um nutricionista autenticado na plataforma.

**Link do board do projeto:**

<https://github.com/orgs/Moveat-Fit/projects/8>

**Link do vídeo apresentando a funcionalidade:**

<https://www.youtube.com/watch?v=Wxwc42DQpKE>

**Link secundário do vídeo:**

<https://drive.google.com/drive/folders/1Dygz18fBlhQVpC7f1SXYizxsCEhfzbfR>

**Link do código-fonte (GitHub):**

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

# I. RELATÓRIO DO TESTE DE API

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

**Sistema Operacional:** Windows 11 Pro

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

**Data dos testes:** 26/04/2025 e 27/04/2025

## 1. OBJETIVO DOS TESTES DE API

Validar a funcionalidade dos endpoints da API, verificando as respostas HTTP, a estrutura do JSON das respostas e a segurança da API em relação à autorização de acesso. As validações foram feitas para cenários positivos e também cenários negativos. Todos os testes seguem a mesma estrutura:

```
/* TESTES PARA REGISTRO BEM SUCEDIDO */
if (pm.response.code === 201) {
  pm.test("Status code é 201 - Registro bem sucedido", function() {
    pm.response.to.have.status(201);
  });
  pm.test("Mensagem de sucesso", function() {
    pm.expect(jsonData.message).to.eql("Profissional registrado com sucesso");
  });
  pm.test("Response deve ter a estrutura correta", function() {
    pm.expect(jsonData).to.have.property("message");
    pm.expect(typeof jsonData.message).to.eql("string");
  });
}

/* TESTES PARA CAMPOS OBRIGATÓRIOS AUSENTES */
} else if (pm.response.code === 400) {
  pm.test("Status code é 400 - Campos obrigatórios ausentes", function() {
    pm.response.to.have.status(400);
  });
  pm.test("Mensagem de erro", function() {
    pm.expect(jsonData.message).to.eql("Campos obrigatórios ausentes ou inválidos");
  });
}

/* TESTES PARA EMAIL JÁ CADASTRADO */
} else if (pm.response.code === 409) {
  pm.test("Status code é 409 - Email, CPF ou telefone já cadastrado", function() {
    pm.response.to.have.status(409);
  });
  pm.test("Mensagem de erro", function() {
    pm.expect(jsonData.message).to.eql("Email, CPF ou número de telefone já registrado");
  });
}

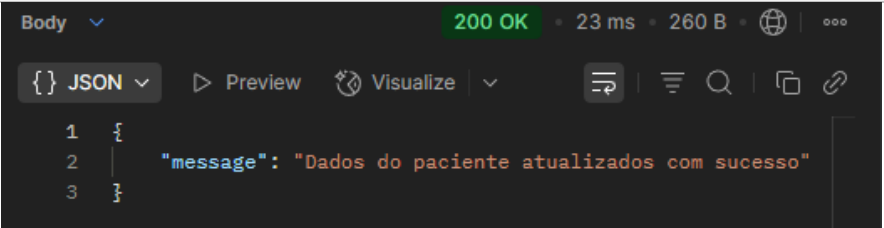
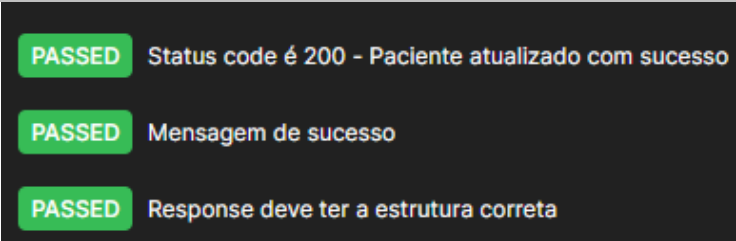
/* TESTES PARA ERRO INTERNO DO SERVIDOR */
} else if (pm.response.code === 500) {
  pm.test("Status code é 500 - Erro interno do servidor", function() {
    pm.response.to.have.status(500);
  });
  pm.test("Mensagem de erro", function() {
    pm.expect(jsonData.message).to.include("Erro interno do servidor");
  });
}
}
```

## 2. ENDPOINTS TESTADOS


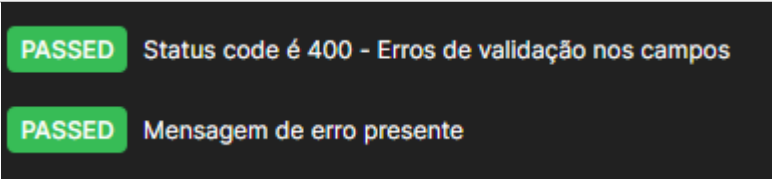
Endpoint	Método HTTP	Descrição	Status Esperado
/patient/<patient_id>/update	PUT	Atualiza dados de um paciente	200 OK
/deletePatient/<patient_id>	DELETE	Deleta um paciente	200 OK
/patients/<professional_id>	GET	Lista pacientes do nutricionista	200 OK
/patient/<patient_id>	GET	Lista os dados de um paciente específico	200 OK

**3. ATUALIZAR DADOS DO PACIENTE (PUT /patient/<patient\_id>/update) |**  
**Apenas nutricionistas autenticados podem editar dados dos seus pacientes)**

**3.1. CENÁRIO POSITIVO: ATUALIZAÇÃO DE DADOS BEM-SUCEDIDA**

<b>Status Code esperado</b>	<b>200</b>
<b>Resultado esperado</b>	Mensagem: “Dados do paciente atualizados com sucesso”
<b>Resultado obtido</b>	
<b>Corpo da requisição</b>	<pre>{   "full_name": "Novo Nome Completo",   "birth_date": "1990-01-01",   "gender": "M",   "email": "novoemail@example.com",   "mobile": "11987654321",   "cpf": "12345678901",   "weight": 70.5,   "height": 1.75,   "note": "Nota atualizada" }</pre>
<b>Evidência</b>	
<b>Implementação</b>	Manual
<b>Status do teste</b>	<b>Passou</b>

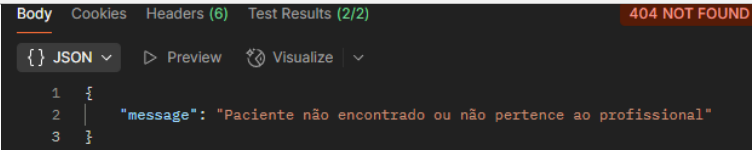
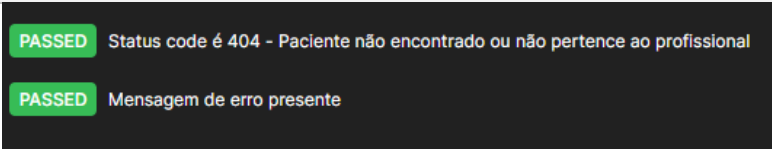
3.2. CENÁRIO NEGATIVO: TENTATIVA DE ATUALIZAÇÃO DE DADOS COM CAMPOS AUSENTES

Status Code esperado	400	
Resultado esperado	Erros de validação nos campos	
Resultado obtido		
Corpo da requisição	<pre>{   "full_name": "",   "birth_date": "2001-09-11",   "gender": "M",   "email": "novoemail@teste.com",   "mobile": "11987654321",   "cpf": "12345678901",   "weight": 70.5,   "height": 1.75,   "note": "Nota atualizada" }</pre>	
Evidência		
Implementação	Manual	
Status do teste	Passou	

3.3. CENÁRIO NEGATIVO: TENTATIVA DE ATUALIZAÇÃO DE DADOS COM OUTRO EMAIL, TELEFONE OU CPF JÁ CADASTRADO

Status Code esperado	409
Resultado esperado	Mensagem: “Email, CPF ou telefone já registrado”
Resultado obtido	<div><pre>1  { 2       "message": "Email já registrado" 3  }</pre></div> <div><pre>1  { 2       "message": "Número de telefone já registrado" 3  }</pre></div> <div><pre>1  { 2       "message": "CPF já registrado" 3  }</pre></div>
Corpo da requisição	<pre>{   "full_name": "Novo Nome Completo",   "birth_date": "1990-01-01",   "gender": "M",   "email": // email já cadastrado,   "mobile": // telefone já cadastrado,   "cpf": // CPF já cadastrado,   "weight": 70.5,   "height": 1.75,   "note": "Nota atualizada" }</pre>
Evidência	<div><div>PASSED</div> Status code é 409 - Dados já cadastrados</div> <div><div>PASSED</div> Mensagem de erro apropriada</div>
Implementação	Manual
Status do teste	REPROVADO (27/04/2025) CORRIGIDO E VALIDADO (27/04/2025)

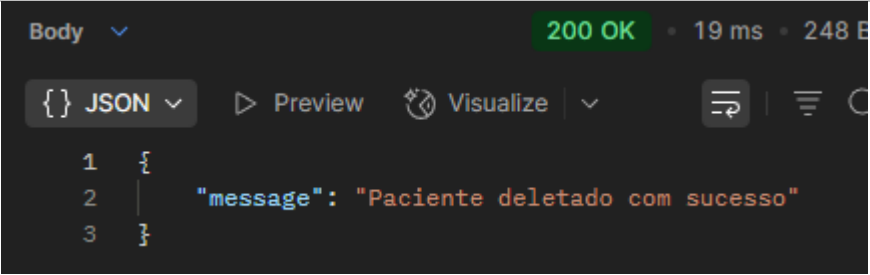
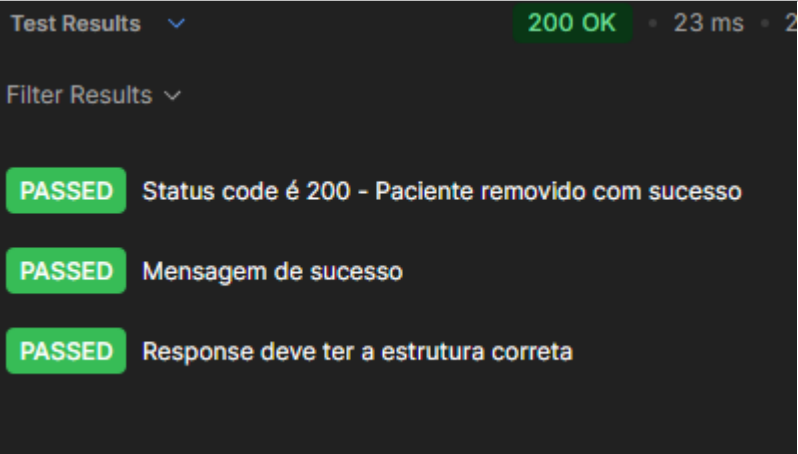
3.4. CENÁRIO NEGATIVO: TENTATIVA DE ATUALIZAÇÃO DE DADOS COM ACESSO NÃO AUTORIZADO (PACIENTE NÃO PERTENCE AO NUTRICIONISTA LOGADO)

Status Code esperado	404
Resultado esperado	Mensagem: “Paciente não encontrado ou não pertence ao profissional”
Resultado obtido	
Corpo da requisição	<pre>/* Deve estar logado como outro usuário nutricionista Paciente do: teste@nutricionista.com Logado como: teste2@nutricionista.com  */ {   "full_name": "Novo Nome Completo",   "birth_date": "1990-01-01",   "gender": "M",   "email": "novoemail@example.com",   "mobile": "11987654321",   "cpf": "12345678901",   "weight": 70.5,   "height": 1.75,   "note": "Nota atualizada" }</pre>
Evidência	
Implementação	Manual
Status do teste	Passou

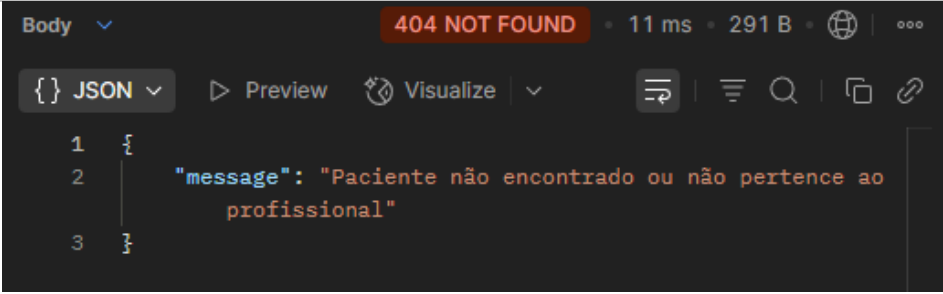


4.      **REMOVER PACIENTE (DELETE /deletePatient/<patient\_id> | Apenas nutricionistas autenticados podem remover seus pacientes)**

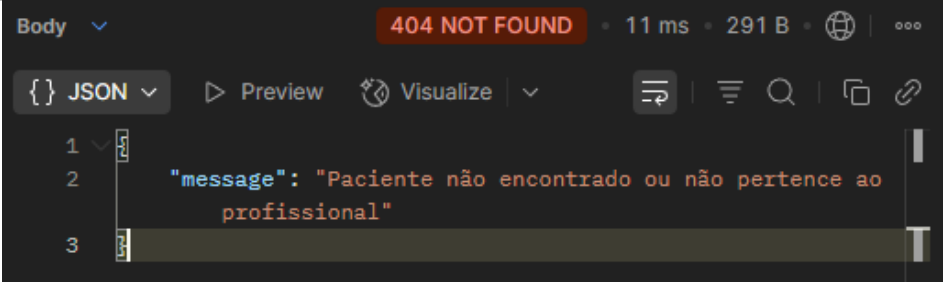
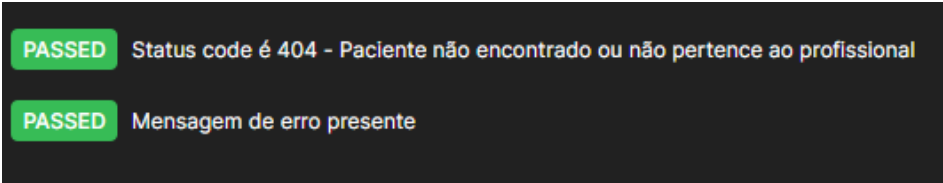
4.1.    CENÁRIO POSITIVO: DELETAR PACIENTE

Status Code esperado	200
Resultado esperado	Mensagem: “Paciente deletado com sucesso”
Resultado obtido	
Request	DELETE {{base_url}}/deletePatient/{{patient_id}}
Evidência	
Implementação	Manual
Status do teste	Passou

4.2. CENÁRIO NEGATIVO: TENTATIVA DE REMOVER UM PACIENTE INEXISTENTE

Status Code esperado	404
Resultado esperado	Mensagem: “Paciente não encontrado ou não pertence ao profissional”
Resultado obtido	 A screenshot of a REST client interface. At the top, it shows 'Body' with a dropdown arrow, '404 NOT FOUND' in a red box, '11 ms', '291 B', and a globe icon. Below this is a toolbar with a JSON icon, a 'Preview' button, a 'Visualize' button, and several other icons. The main area shows a JSON body with three lines: 1 {, 2   "message": "Paciente não encontrado ou não pertence ao", 3   "profissional"}. The text is color-coded: { is blue, "message": is orange, "Paciente não encontrado ou não pertence ao" is orange, "profissional" is orange, and } is blue.
Request	DELETE {{base_url}}/deletePatient/0
Evidência	 A screenshot of test results. It shows two green boxes with the word 'PASSED' in white. The first box is followed by the text 'Status code é 404 - Paciente não encontrado ou não pertence ao profissional'. The second box is followed by the text 'Mensagem de erro presente'.
Implementação	Manual
Status do teste	Passou

4.3. CENÁRIO NEGATIVO: TENTATIVA DE REMOVER UM PACIENTE DE OUTRO  
PROFISSIONAL

Status Code esperado	404
Resultado esperado	Mensagem: “Paciente não encontrado ou não pertence ao profissional”
Resultado obtido	
Request	DELETE {{base_url}}/deletePatient/{{patient_id}}
Evidência	
Implementação	Manual
Status do teste	Passou

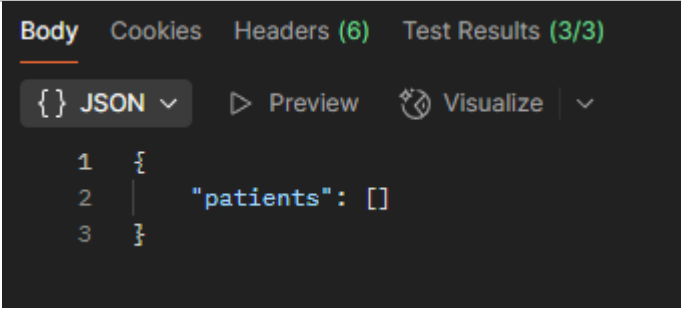
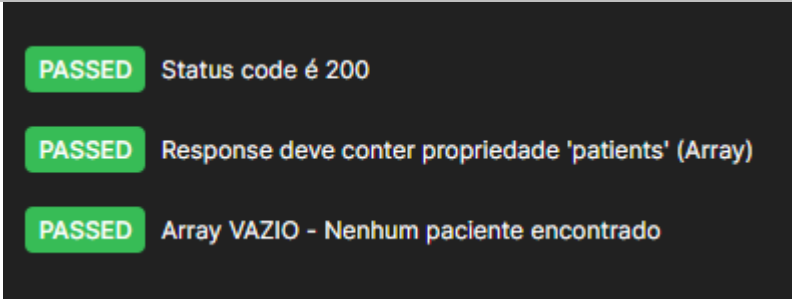
5. LISTAR PACIENTES DE UM NUTRICIONISTA

(GET /patients/<professional\_id>)

5.1. CENÁRIO POSITIVO: PACIENTES ENCONTRADOS

Status Code esperado	200
Resultado esperado	Array preenchido com os dados dos pacientes cadastrados
Resultado obtido	<pre>{   "patients": [     {       "id": 6,       "full_name": "Paciente de Teste",       "birth_date": "2001-09-11",       "gender": "F",       "email": "paciente@teste.com",       "mobile": "11996696799",       "cpf": "50262230830",       "weight": "50.00",       "height": "1.50",       "note": "Nada",       "professional_id": 1,       "created_at": "2025-03-31 23:06:16",       "updated_at": "2025-03-31 23:06:16"     },     {       "id": 8,       "full_name": "Paciente de Teste 2",       "birth_date": "2001-09-11",       "gender": "F",       "email": "paciente2@teste.com",       "mobile": "11996690099",       "cpf": "50260030830",       "weight": "50.00",       "height": "1.50",       "note": "Diabético",       "professional_id": 1,       "created_at": "2025-04-01 18:51:11",       "updated_at": "2025-04-01 18:51:11"     }   ] }</pre>
Request	GET {{base_url}} /patients/{{id_nutricionistaComPaciente}}
Evidência	<div><div>PASSED</div>Status code é 200</div> <div><div>PASSED</div>Response deve conter propriedade 'patients' (Array)</div> <div><div>PASSED</div>Array PREENCHIDO - Valida estrutura do paciente</div>
Implementação	Manual
Status do teste	Passou

5.2. CENÁRIO NEGATIVO: NENHUM PACIENTE ENCONTRADO

Status Code esperado	200
Resultado esperado	Array vazio
Resultado obtido	
Request	GET {{base_url}} /patients/{{id_nutricionistaComPaciente}}
Evidência	
Implementação	Manual
Status do teste	REPROVADO (01/05/2025) CORRIGIDO E VALIDADO (02/05/2025)

6. LISTAR DADOS CADASTRAIS DE UM PACIENTE ESPECÍFICO  
(GET /patient/<patient\_id>)

6.1. CENÁRIO POSITIVO: PACIENTE ENCONTRADO

Status Code esperado	200
Resultado esperado	Array preenchido com os dados de um paciente específico
Resultado obtido	<div><div>BodyCookiesHeaders (6)Test Results (3/3)</div><div><div>{ }JSON ▾▶ Preview🔄 Visualize ▾</div><pre>1 { 2   "patient": [ 3     { 4       "id": 19, 5       "full_name": "Lucas Guimarães", 6       "birth_date": "2012-06-06", 7       "gender": "M", 8       "email": "lucas@gmail.com", 9       "mobile": "11996052188", 10      "cpf": "30265587410", 11      "weight": "80.00", 12      "height": "1.57", 13      "note": "", 14      "professional_id": 6, 15      "created_at": "2025-04-06 17:24:31", 16      "updated_at": "2025-04-06 17:24:31" 17    } 18  ] 19 }</pre></div></div>
Request	GET {{base_url}}/patient/{{id_pacienteCadastrado}}
Evidência	<div><div>PASSEDStatus code é 200</div><div>PASSEDResponse deve conter propriedade 'patient' (Array)</div><div>PASSEDArray PREENCHIDO - Valida estrutura do paciente</div></div>
Implementação	Manual
Status do teste	Passou

6.2. CENÁRIO NEGATIVO: PACIENTE NÃO ENCONTRADO

Status Code esperado	404
Resultado esperado	Mensagem de erro “Paciente não encontrado”
Resultado obtido	<div><div>BodyCookiesHeaders (6)Test Results (2/2)</div><div><div>{ }JSON ▾▶ Preview🔄 Visualize ▾</div><div><div>1 {</div><div>2  </div><div>3 }</div><div>"message": "0 paciente com id 0 não foi encontrado."</div></div></div></div>
Request	GET {{base_url}}/patient/{{id_pacienteInexistente}}
Evidência	<div><div>PASSEDStatus code é 404 - Paciente não encontrado</div><div>PASSEDMensagem de erro presente</div></div>
Implementação	Manual
Status do teste	Passou

## 7. 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



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

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

**Sistema Operacional:** Windows 11 Pro

**Data dos testes:** 03/05/2025 – 04/05/2025

**Relatório completo:** [CLIQUE AQUI](#)

### 8. OBJETIVO























Validar que o sistema permite que o nutricionista atualize os dados de um paciente já cadastrado ou remova-o de sua lista.

### 9. PRÉ-CONDIÇÕES

- O nutricionista está autenticado no sistema;
- O nutricionista possui pacientes já cadastrados no sistema

### 10. CASOS DE TESTE EXECUTADOS

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

SPRINT 3: DASHBOARD NUTRICIONISTA (EDITAR E EXCLUIR PACIENTES) +    			
+ Create quick test			
Excluir paciente +    			
		MOVEAT-23	Deletar paciente com sucesso
		MOVEAT-24	Cancelar exclusão de paciente
+ Create quick test			
Editar dados do paciente +    			
		MOVEAT-25	Atualizar dados com sucesso
		MOVEAT-26	Validar tratamento de erro para e-mail inválido
		MOVEAT-27	Validar tratamento de erro para e-mail já cadastrado
		MOVEAT-28	Validar tratamento de erro para campos não preenchidos
+ Create quick test			

**11. BUGS ENCONTRADOS DURANTE OS TESTES DE SISTEMA**

TESTES DE SISTEMA		
ID	DESCRIÇÃO	STATUS
<a href="#">BUG-DASHBOARD-001</a>	Botões incorretos no formulário de atualização dos dados do paciente (Prioridade Alta)	Corrigido
<a href="#">BUG-DASHBOARD-002</a>	Mensagem de erro e de sucesso ao mesmo tempo (Prioridade Média)	Corrigido
<a href="#">BUG-DASHBOARD-003</a>	Os campos de peso e altura não seguem o mesmo padrão de mensagem de erro (Prioridade Média)	Corrigido

### III. RELATÓRIO DO TESTE DE REGRESSÃO

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

**Sistema Operacional:** Windows 11 Pro

**Ferramenta de teste:** Cypress

**Data dos testes:** 04/05/2025

#### 12. OBJETIVO

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

#### 13. 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 loginNutricionista	
tsucarol · Ran 7 mins ago · 02:04 · main · None · # 9	
Overview	Test Results 19
Specs 7	Errors 0
UI Coverage	Acces
Execution order	Status
Flaky Tests	Last Modified
cypress\e2e\Fluxo nutricionista\0-cadastroNutricionista.cy.js	
3 · 00:13 · E2E · Windows 10.0.26100 · Chrome 136 · 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 · 00:07 · E2E · Windows 10.0.26100 · Chrome 136 · 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	

## update loginNutricionista

tsucarol · Ran 7 mins ago · 02:04 · main · None · #9

Overview **Test Results 19** Specs 7 Errors 0 UI Coverage Accessibility Properties

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

✓ 6 · 00:45 · E2E · Windows 10.0.26100 · Chrome 136 · View Output

- ✓ Dashboard do Nutricionista > Cadastro com dados válidos
- ✓ 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

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

✓ 3 · 00:08 · E2E · Windows 10.0.26100 · Chrome 136 · View Output

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

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

✓ 4 · 00:06 · E2E · Windows 10.0.26100 · Chrome 136 · View Output

- ✓ Validando tratamento de senha > Deve exibir erro quando a senha tiver menos de 8 caracteres
- ✓ 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