



**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS  
SISTEMAS DE INFORMAÇÃO**

# GeriCare

---

**Plano do Projeto de Testes**  
Versão: 1.0

---

## DESCRIÇÃO DO DOCUMENTO

OBJETIVO	Este documento tem como objetivo documentar os testes feitos na aplicação.
LOCALIZAÇÃO	Documento disponível no Repositório do GitHub <a href="#">Teste GeriCare</a> e na entrega do Canvas.

## HISTÓRICO DE VERSÕES DO DOCUMENTO

VERSÃO	OBJETIVO	DATA	RESPONSÁVEL
1.1	Preencher o plano de testes	03/12/2025	Ana Flavia
1.2	Preencher Produtos Liberados e Fluxo de Trabalho	04/12/2025	Ana Clara
1.3	Preencher Necessidades Ambientais	04/12/2025	Sabrina

## APROVAÇÕES INTERNAS

PERFIL	NOME	DATA
Analista de Sistemas	Otávio	06/12/205
Analista de Qualidade	João	07/12/2025

## ÍNDICE

---

<b>1.</b>	<b>VISÃO GERAL.....</b>	<b>3</b>
1.1	OBJETIVO .....	3
1.2	DESCRIÇÃO DO PRODUTO A SER TESTADO .....	3
<b>2.</b>	<b>ESCOPO .....</b>	<b>3</b>
2.1	PLANEJAMENTO DO ESCOPO .....	3
<b>3.</b>	<b>ÍTEM-ALVO DOS TESTES.....</b>	<b>3</b>
<b>4.</b>	<b>RESUMO DOS TESTES PLANEJADOS.....</b>	<b>5</b>

# 1. Visão geral

---

## 1.1 Objetivo

Este Plano de Testes tem como finalidade documentar e orientar toda a estratégia de testes aplicada ao sistema GeriCare, um software de gerenciamento de estoque voltado para instituições de cuidados a idosos.

O documento especifica os requisitos a serem testados, as técnicas aplicadas, os prazos, os critérios de aceitação e todos os casos de teste manuais e automatizados.

## 1.2 Descrição do produto a ser testado

O GeriCare é um sistema web voltado ao controle e gestão de estoque de itens utilizados por instituições, incluindo:

- 1) Itens de Cozinha
- 2) Materiais de Limpeza
- 3) Itens Pessoais
- 4) Itens Médicos
- 5) Cadastro de Residentes
- 6) Controle de Usuários

Funcionalidades principais:

- Controle de estoque por categorias
- Atualização de itens e quantidades
- Alerta de estoque baixo
- Dashboard com gráficos
- Login e gestão de usuários

Repositórios no GitHub:

[Trabalho Interdisciplinar](#)

[Teste GeriCare](#)

# 2. Escopo

---

## 2.1 Planejamento do escopo

Os testes ocorrerão no módulo de usuário mock (administrador), no qual vão ser empregados testes de unidade, testes funcionais automatizados (Selenium) e testes funcionais manuais.

### 3. Itens-alvo dos testes

---

#### 1 - UC1: Login

**01 - Descrição da história/necessidade** Como usuário, quero realizar login no sistema para acessar o gerenciamento de estoque e de pessoas do asilo.

##### 02 - O que deve ser feito

- Tela de login com CPF e senha.
- Validação de credenciais.
- Mensagens de erro em caso de falha.

##### 03 - Não faz parte da história

- Recuperação ou redefinição de senha.

#### 2 - UC2: Cadastro de Moradores

**01 - Descrição da história/necessidade** Como administrador, quero cadastrar, editar e remover moradores para manter o controle atualizado das pessoas que vivem no asilo.

##### 02 - O que deve ser feito

- Tela de cadastro de moradores.
- Campos para nome, CPF, idade e dados pessoais relevantes.
- Botão para salvar novo morador.
- Botão para editar morador existente.
- Botão para excluir morador.

##### 03 - Regras de negócio

- O sistema deve validar CPF antes de salvar.
- Não permitir cadastro duplicado de CPF.
- Apenas usuários com perfil administrador podem excluir registros.

#### 3 - UC3: Cadastro e Gerenciamento de Itens (Estoque)

**01 - Descrição da história/necessidade** Como responsável pelo estoque, quero cadastrar, editar e remover itens para controlar os recursos da cozinha e demais áreas do asilo.

## 02 - O que deve ser feito

- Tela de estoque com listagem de itens.
- Botão para adicionar novo item (nome, categoria, quantidade).
- Botão para editar item existente.
- Botão para excluir item.

## 03 - Regras de negócio

- O sistema deve validar que a quantidade seja um número inteiro positivo.
- Não permitir cadastro de item sem nome ou categoria.
- Apenas usuários logados podem acessar o estoque.

## 4 - UC4: Cadastro de Usuários

**01 - Descrição da história/necessidade** Como administrador, quero cadastrar, editar e remover usuários para controlar quem tem acesso ao sistema de gerenciamento do asilo.

## 02 - O que deve ser feito

- Tela de cadastro de usuários.
- Campos para nome, CPF, senha e perfil de acesso (administrador ou comum).
- Botão para salvar novo usuário.
- Botão para editar usuário existente.
- Botão para excluir usuário.

## 03 - Regras de negócio

- O sistema deve validar CPF antes de salvar.
- Não permitir cadastro duplicado de CPF.
- Apenas administradores podem cadastrar ou excluir usuários.

## 4. Resumo dos testes planejados

---

### Testes Unitários (automatizados):

CT01: Verificar login com CPF e senha corretos, garantindo geração de token e acesso.

CT02: Verificar falha ao logar com senha incorreta, exibindo mensagem de erro e sem gerar token.

CT03: Verificar criação de usuário, adicionando novo registro à lista de usuários.

CT04: Verificar atualização de usuário, confirmando alteração dos dados na lista.  
 CT05: Verificar remoção de usuário, garantindo exclusão do registro da lista.  
 CT06: Verificar criação de residente, adicionando novo registro à lista de residentes.  
 CT07: Verificar atualização de residente, confirmando alteração dos dados na lista.  
 CT08: Verificar remoção de residente, garantindo exclusão do registro da lista.  
 CT09: Verificar criação de item, adicionando novo registro à lista de itens.  
 CT10: Verificar exclusão de item, garantindo remoção do registro da lista.

#### **Testes funcionais (manuais e automatizados):**

CT01: Verificar login com CPF e senha válidos, garantindo acesso à plataforma.  
 CT02: Verificar login com senha incorreta, exibindo mensagem “CPF ou senha inválidos”.  
 CT03: Verificar login com CPF incorreto, exibindo mensagem “CPF ou senha inválidos”.  
 CT04: Verificar login com CPF e senha incorretos, exibindo mensagem “CPF ou senha inválidos”.  
 CT05: Verificar logout, garantindo retorno à tela de login.  
 CT06: Verificar edição de item no estoque (cozinha), confirmando atualização da quantidade para o novo valor.  
 CT07: Verificar exclusão de item no estoque (cozinha), garantindo que o item não apareça mais na lista.  
 CT08: Verificar adição de novo morador, confirmando exibição do morador na lista.  
 CT09: Verificar exclusão de morador, garantindo remoção do registro da lista.  
 CT10: Verificar cadastro de novo usuário, confirmando exibição do usuário na lista.

#### **Testes Unitários (Automatizados):**

<b>Caso de Teste</b>	CT01 – Login com CPF e senha corretos
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) O ator insere CPF e senha válidos. 2) O sistema valida as credenciais. 3) O sistema gera um token de autenticação. 4) O sistema permite o acesso.
<b>Dados de entrada</b>	CPF: 123.456.789-00, Senha: 123456
<b>Saída esperada</b>	Token de autenticação gerado
<b>Saída encontrada</b>	Token de autenticação gerado
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT02 – Falha ao logar com senha incorreta
<b>Pré-Condição</b>	Acessar a tela de login
<b>Procedimento</b>	1) O ator insere CPF válido e senha incorreta. 2) O sistema valida as credenciais. 3) O sistema exibe mensagem de erro. 4) O sistema não gera token.
<b>Dados de entrada</b>	CPF: 123.456.789-00, Senha: errado

<b>Saída esperada</b>	Mensagem de erro “CPF ou senha inválidos”
<b>Saída encontrada</b>	Mensagem de erro “CPF ou senha inválidos”
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT03 – Criar usuário
<b>Pré-Condição</b>	Acessar módulo de usuários
<b>Procedimento</b>	1) O ator preenche os dados do novo usuário. 2) O sistema adiciona o usuário à lista.
<b>Dados de entrada</b>	CPF, senha e nome
<b>Saída esperada</b>	Lista de usuários com novo registro
<b>Saída encontrada</b>	Lista de usuários com novo registro
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT04 - Atualizar usuário
<b>Pré-Condição</b>	Usuário existente
<b>Procedimento</b>	1) O ator edita os dados do usuário. 2) O sistema atualiza os dados na lista.
<b>Dados de entrada</b>	ID do usuário e novos dados
<b>Saída esperada</b>	Dados atualizados na lista
<b>Saída encontrada</b>	Dados atualizados na lista
<b>Prioridade</b>	Média
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT05 – Remover usuário
<b>Pré-Condição</b>	Usuário existente
<b>Procedimento</b>	1) O ator solicita exclusão do usuário. 2) O sistema remove o usuário da lista.
<b>Dados de entrada</b>	ID do usuário
<b>Saída esperada</b>	Lista sem o usuário removido
<b>Saída encontrada</b>	Lista sem o usuário removido
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT06 – Criar residente
<b>Pré-Condição</b>	Acessar módulo de residentes
<b>Procedimento</b>	1) O ator preenche os dados do residente. 2) O sistema adiciona o residente à lista.
<b>Dados de entrada</b>	Nome do residente



<b>Saída esperada</b>	Lista de residentes com novo registro
<b>Saída encontrada</b>	Lista de residentes com novo registro
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT07 – Atualizar residente
<b>Pré-Condição</b>	Residente existente
<b>Procedimento</b>	1) O ator edita os dados do residente. 2) O sistema atualiza os dados na lista.
<b>Dados de entrada</b>	ID do residente e novos dados
<b>Saída esperada</b>	Dados atualizados na lista
<b>Saída encontrada</b>	Dados atualizados na lista
<b>Prioridade</b>	Média
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT08 – Remover residente
<b>Pré-Condição</b>	Residente existente
<b>Procedimento</b>	1) O ator solicita exclusão do residente. 2) O sistema remove o residente da lista.
<b>Dados de entrada</b>	ID do residente
<b>Saída esperada</b>	Lista sem o residente removido
<b>Saída encontrada</b>	Lista sem o residente removido
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT10 – Criar item
<b>Pré-Condição</b>	Acessar módulo de estoque
<b>Procedimento</b>	1) O ator preenche os dados do item. 2) O sistema adiciona o item à lista.
<b>Dados de entrada</b>	Nome e categoria do item
<b>Saída esperada</b>	Lista de itens com novo registro
<b>Saída encontrada</b>	Lista de itens com novo registro
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

<b>Caso de Teste</b>	CT10 – Excluir item
<b>Pré-Condição</b>	Item existente

<b>Procedimento</b>	1) O ator solicita exclusão do item. 2) O sistema remove o item da lista.
<b>Dados de entrada</b>	ID do item
<b>Saída esperada</b>	Lista sem o item removido
<b>Saída encontrada</b>	Lista sem o item removido
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

Print do resultado:

```

at src/App.test.js:50:18

PASS src/App.test.js
  useAuthMock
    ✓ login com CPF e senha corretos (526 ms)
    ✓ falha ao logar com senha incorreta (572 ms)
    ✓ logout remove token (2 ms)
  useUsersMock
    ✓ cria usuário (2 ms)
    ✓ atualiza usuário (2 ms)
    ✓ remove usuário (2 ms)
  useResidents
    ✓ cria residente (2 ms)
    ✓ atualiza residente (1 ms)
    ✓ remove residente (2 ms)
  useItems
    ✓ cria item (2 ms)
    ✓ atualiza item (2 ms)
    ✓ exclui item (2 ms)

Test Suites: 1 passed, 1 total
Tests:       12 passed, 12 total
Snapshots:   0 total
Time:        3.779 s
Ran all test suites.

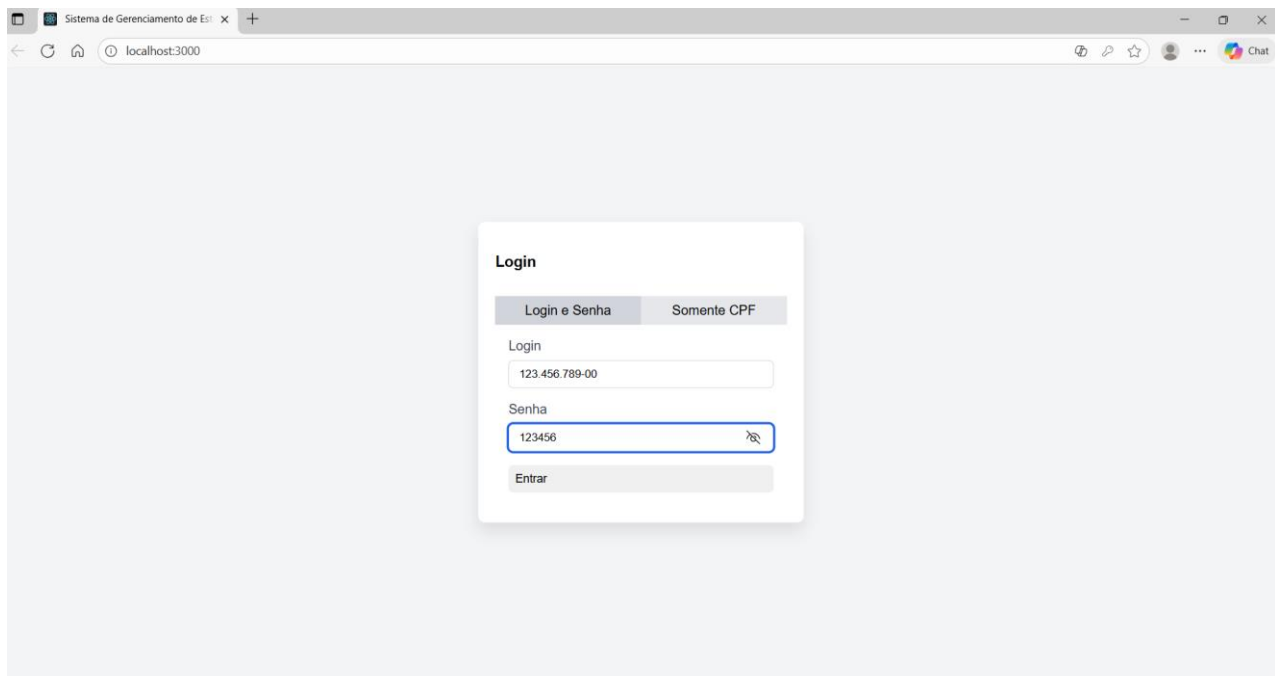
Watch Usage
> Press f to run only failed tests

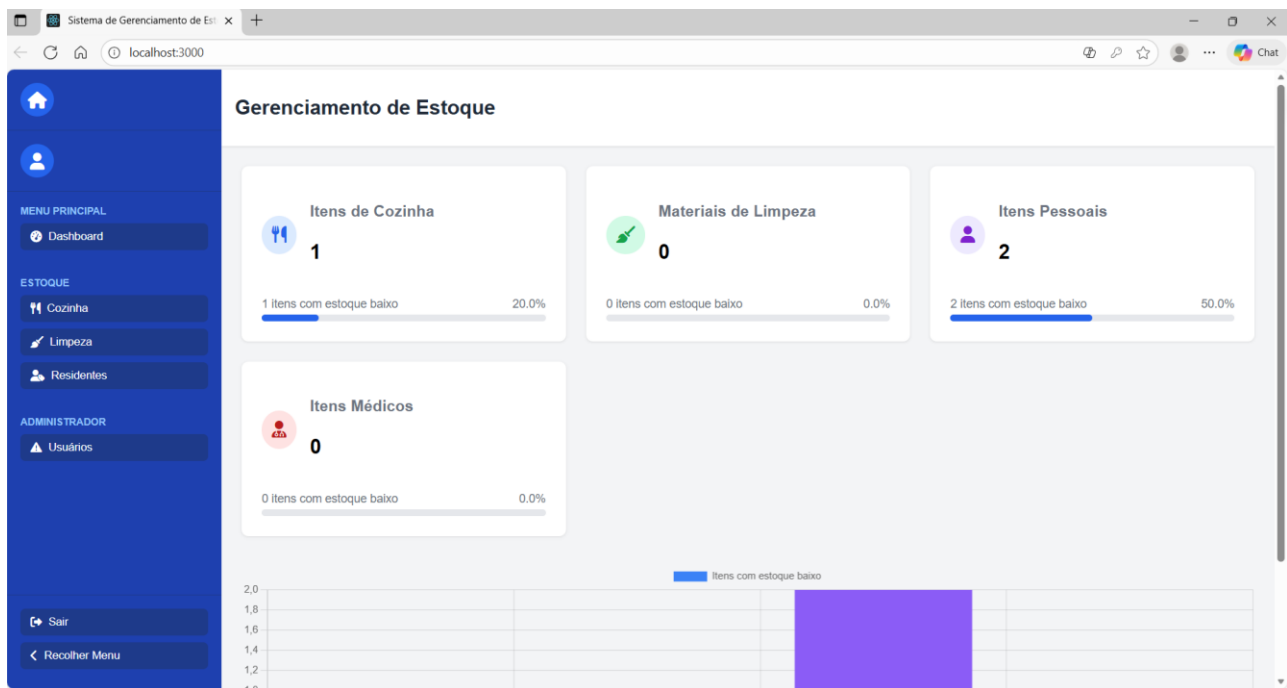
```

### Testes funcionais (manuais):

<b>Caso de Teste</b>	CT01 – Realizar login com sucesso
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF válido. 2) Inserir senha válida. 3) Clicar "Entrar".
<b>Dados de entrada</b>	Usuário e senha válidos.
<b>Saída esperada</b>	Entrar na plataforma.
<b>Saída encontrada</b>	Entrar na plataforma.
<b>Prioridade</b>	Alta
<b>Técnica</b>	Manual

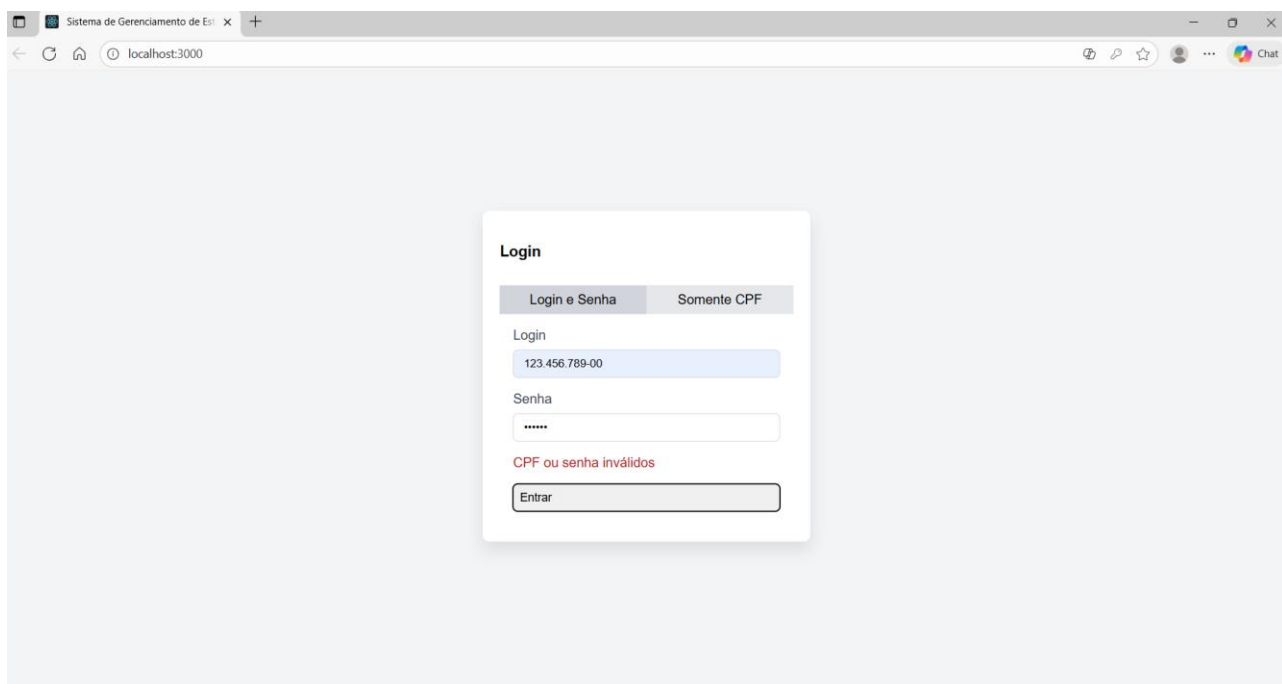
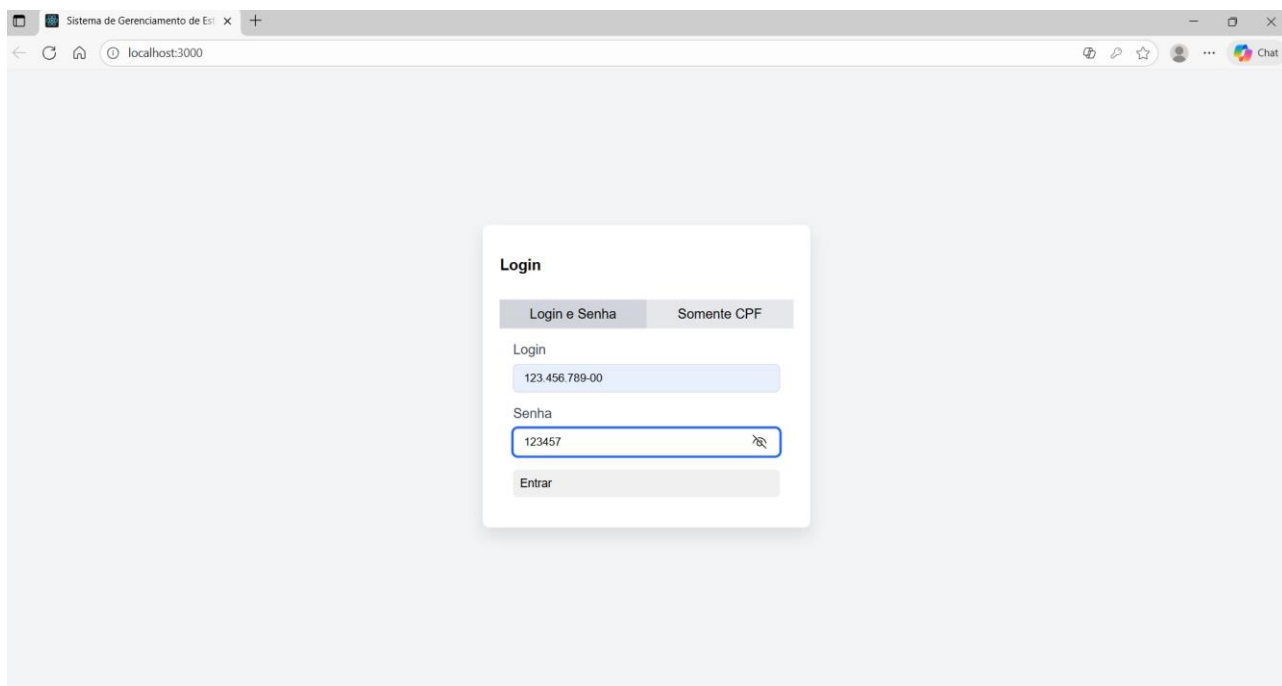
### Print do resultado:





<b>Caso de Teste</b>	CT02 – Login com senha incorreta
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF válido. 2) Inserir senha errada. 3) Clicar "Entrar".
<b>Dados de entrada</b>	Senha errada e CPF válido.
<b>Saída esperada</b>	“CPF ou senha inválidos”
<b>Saída encontrada</b>	“CPF ou senha inválidos”
<b>Prioridade</b>	Alta
<b>Técnica</b>	Manual

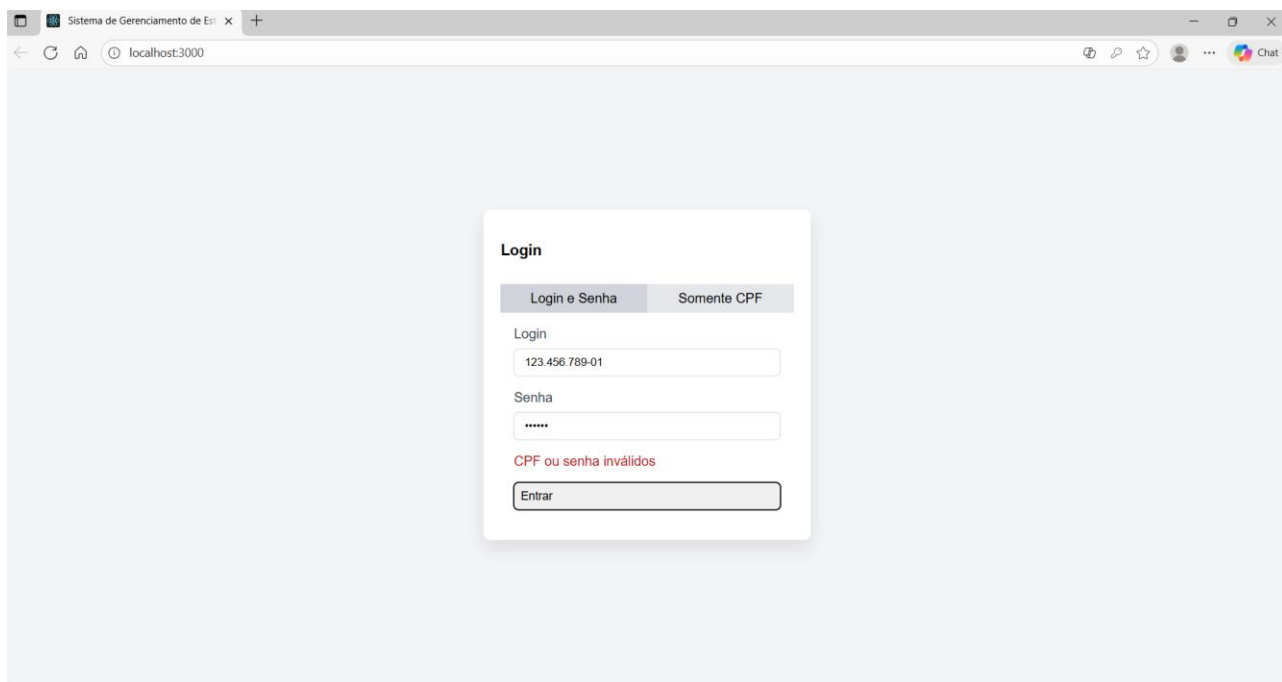
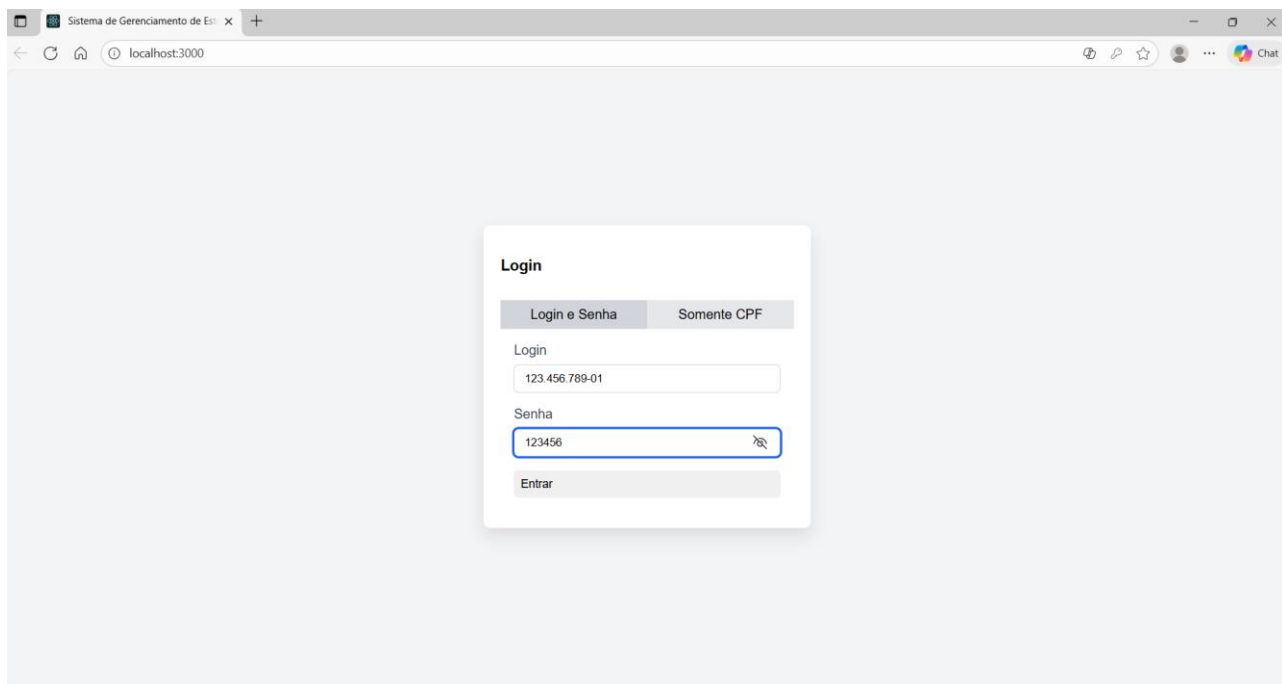
**Print do resultado:**



<b>Caso de Teste</b>	CT03 – Login com CPF incorreto
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF errado.

	2) Inserir senha válida.
	3) Clicar "Entrar".
<b>Dados de entrada</b>	CPF errado e senha válida.
<b>Saída esperada</b>	"CPF ou senha inválidos"
<b>Saída encontrada</b>	"CPF ou senha inválidos"
<b>Prioridade</b>	Alta
<b>Técnica</b>	Manual

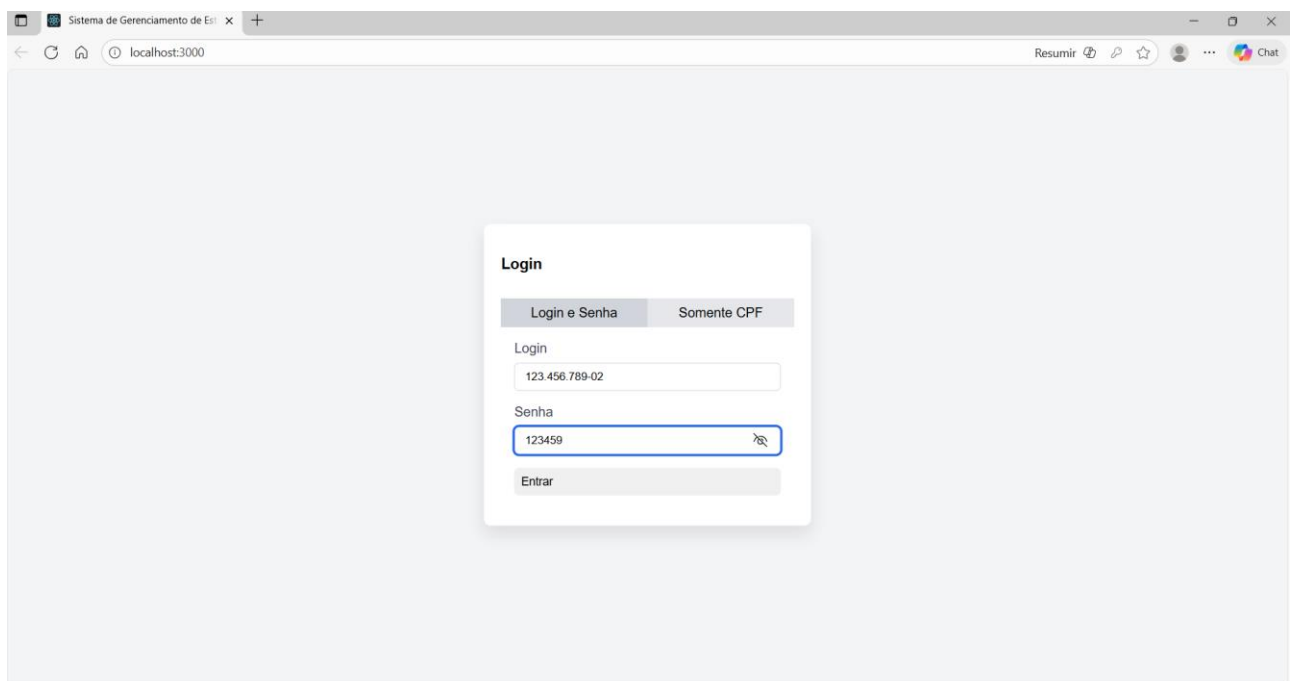
#### Print do resultado:

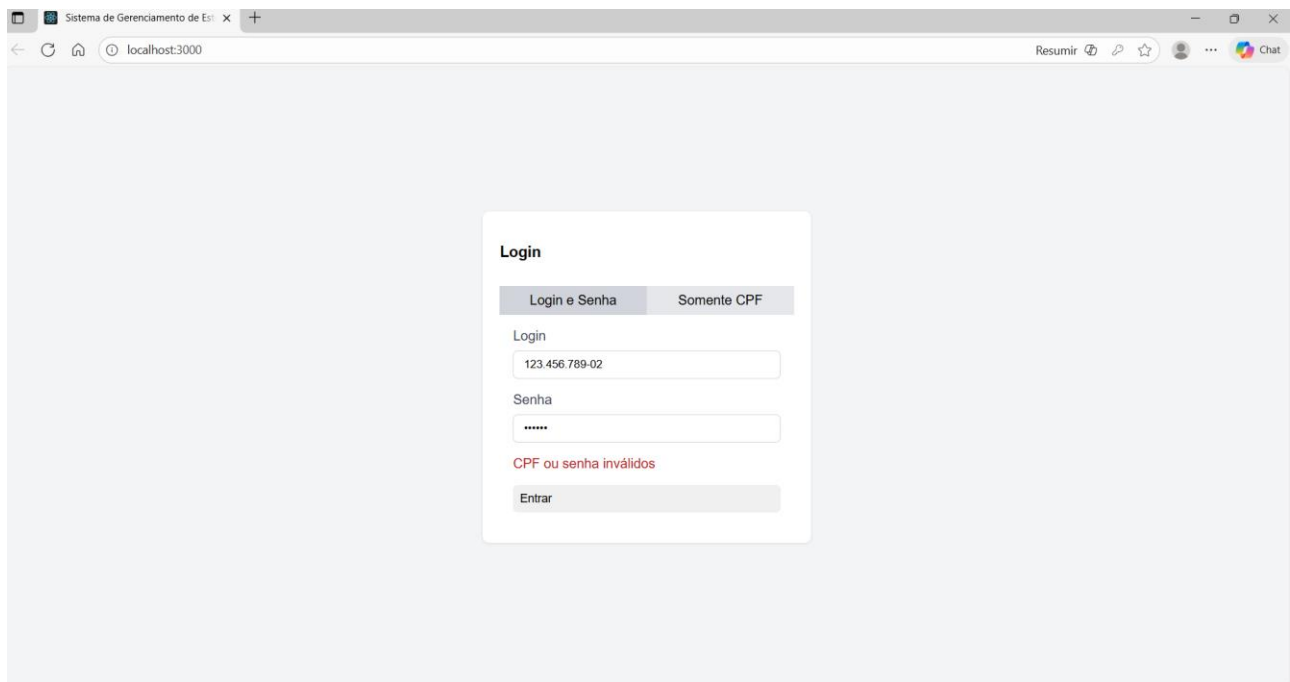


<b>Caso de Teste</b>	CT04 – Login com CPF e senha incorretos
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF errado. 2) Inserir senha errada.

	3) Clicar "Entrar ".
<b>Dados de entrada</b>	CPF e senha incorretos
<b>Saída esperada</b>	“CPF ou senha inválidos”
<b>Saída encontrada</b>	“CPF ou senha inválidos”
<b>Prioridade</b>	Alta
<b>Técnica</b>	Manual

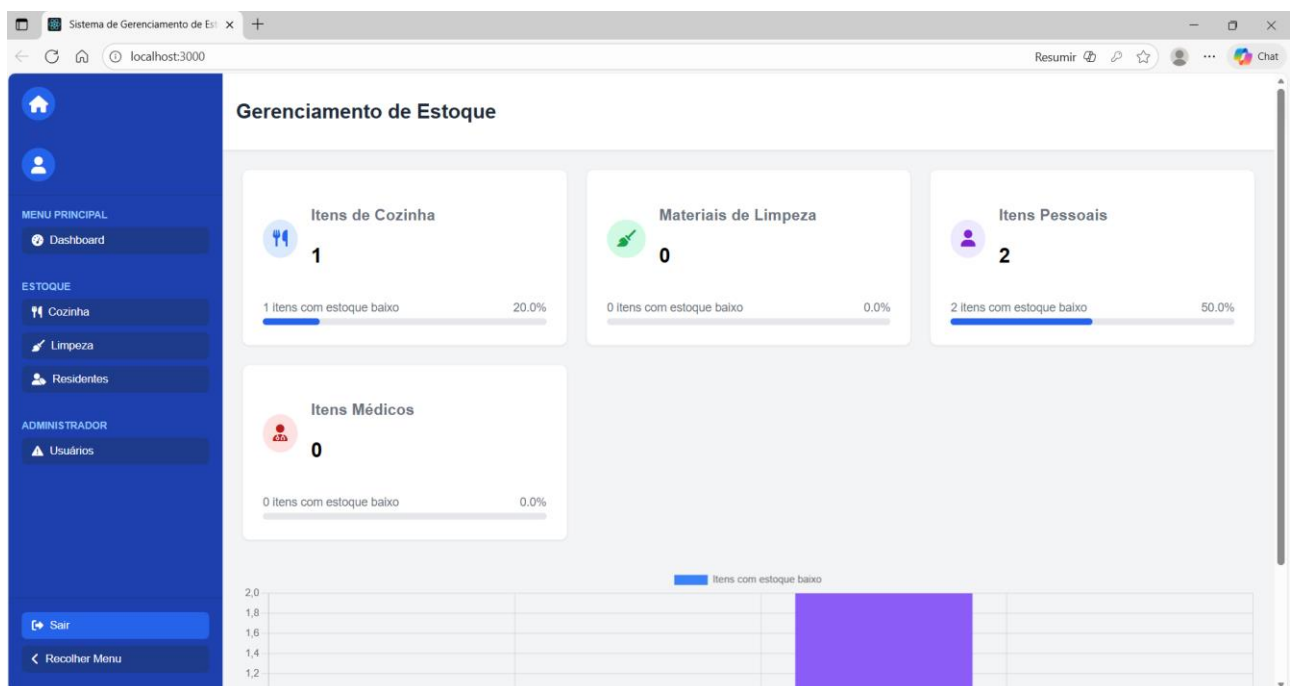
### Print do resultado:



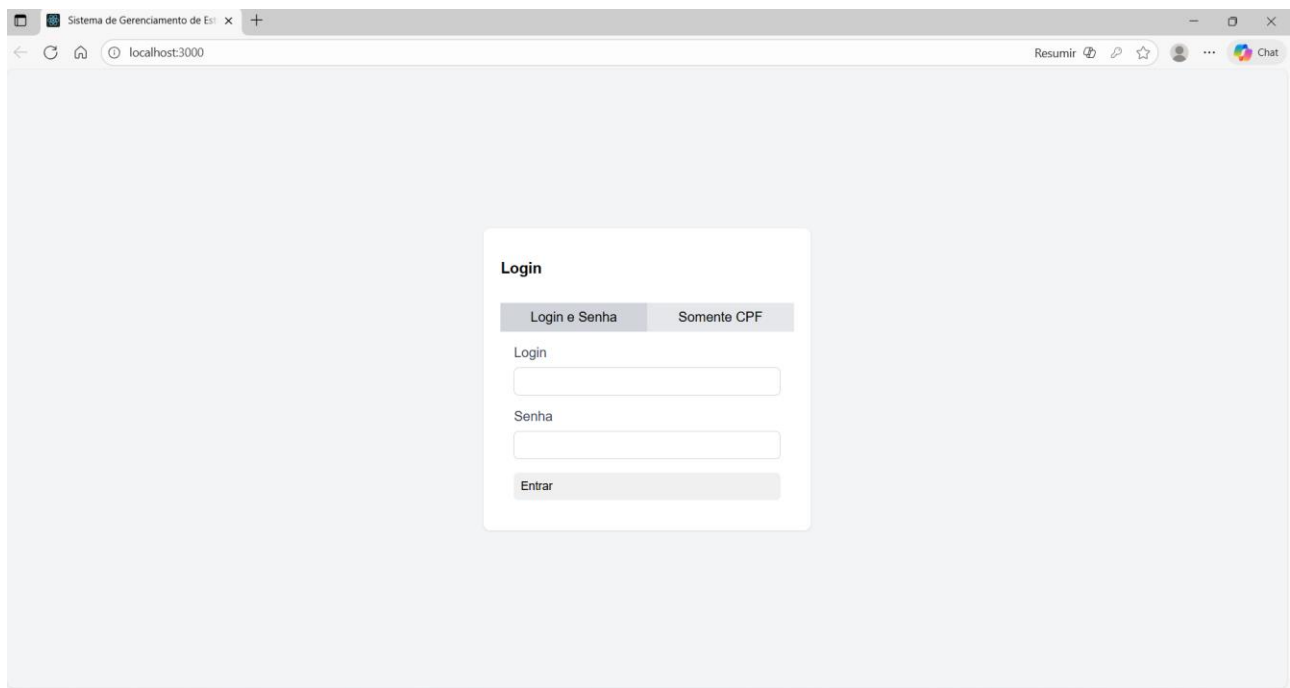


<b>Caso de Teste</b>	CT05 – Logout
<b>Pré-Condição</b>	Estar Logado.
<b>Procedimento</b>	1) Estar logado. 2) Clicar "Sair".
<b>Dados de entrada</b>	Clicar no botão "Sair".
<b>Saída esperada</b>	Retorno à tela de login.
<b>Saída encontrada</b>	Retorno à tela de login.
<b>Prioridade</b>	Alta
<b>Técnica</b>	Manual

### Print do resultado:

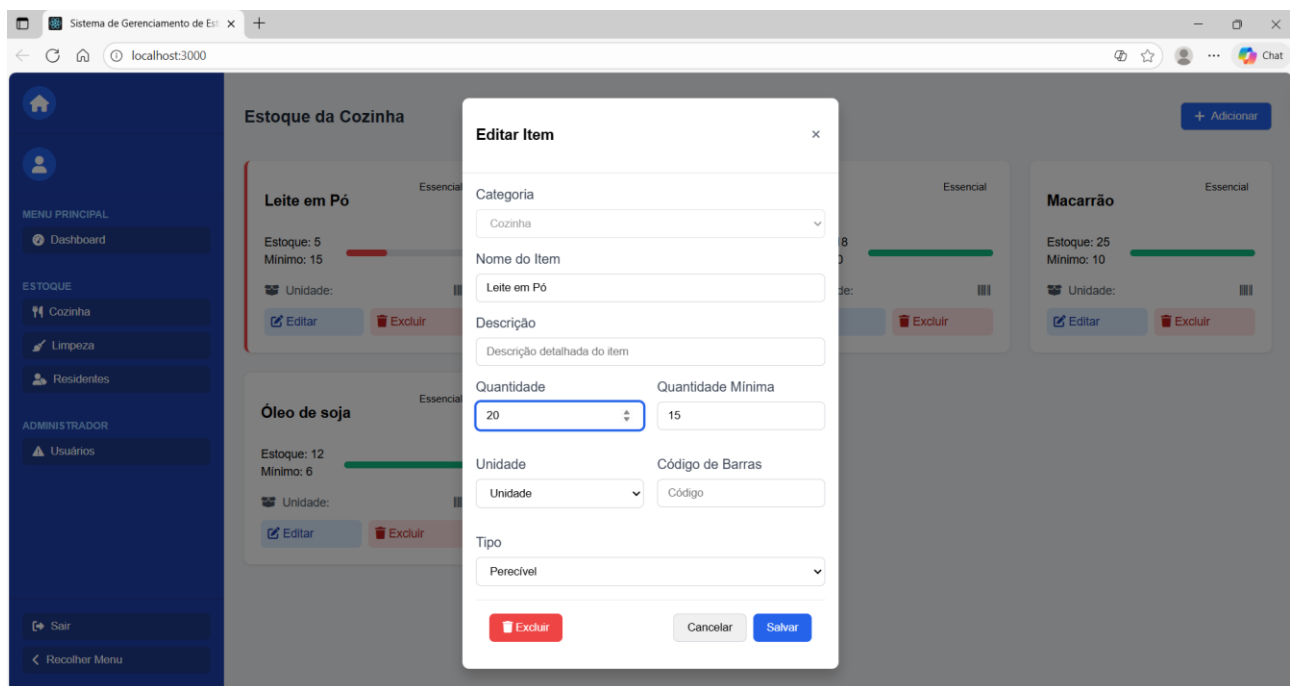
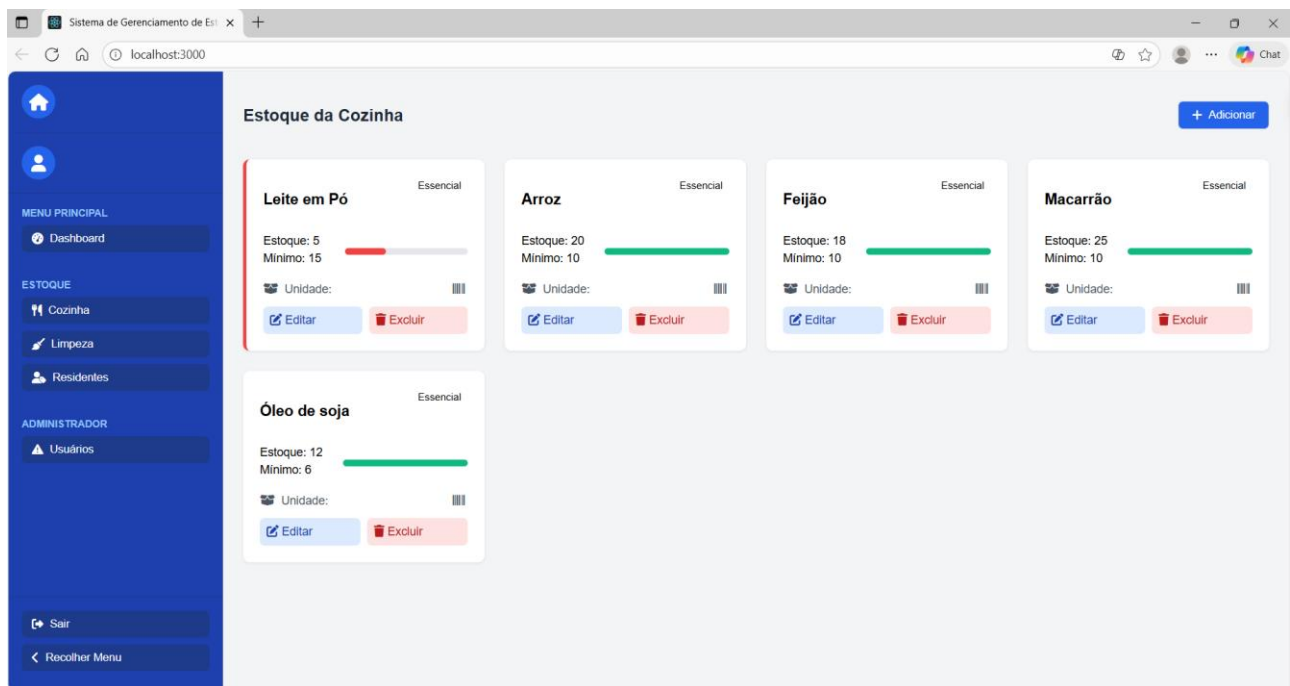


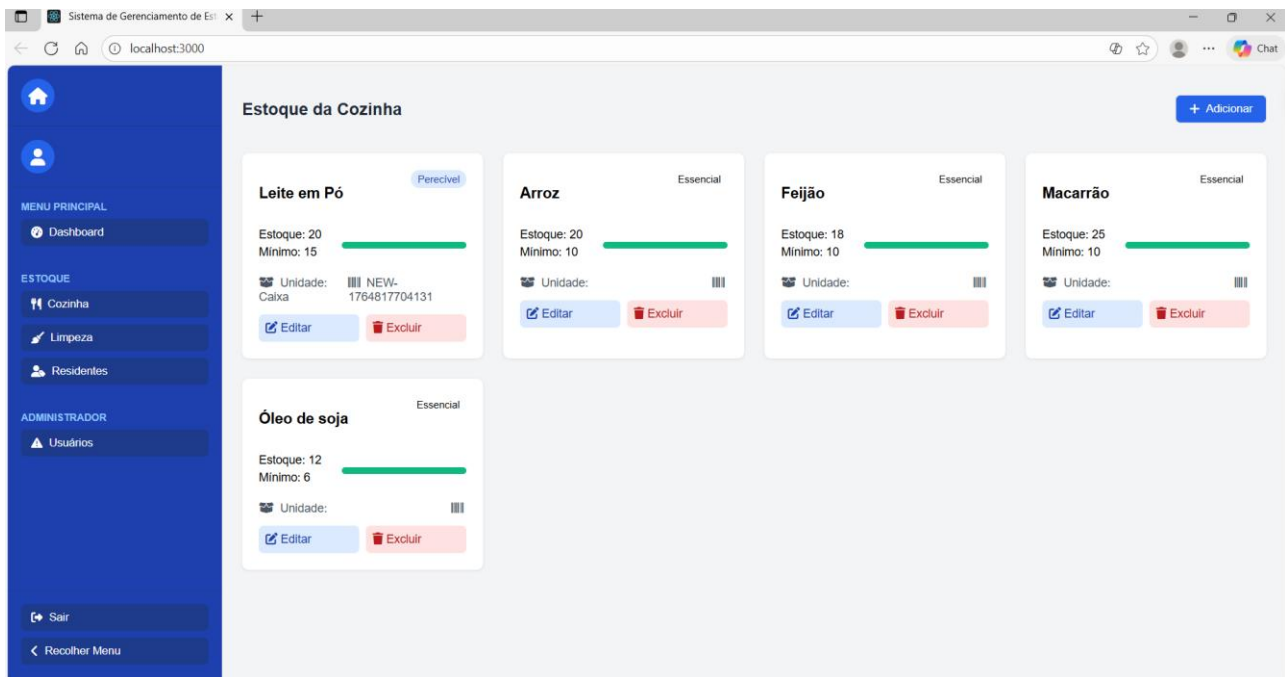




<b>Caso de Teste</b>	CT06 – Editar Estoque (Cozinha)
<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Cozinha deve estar aberta; Deve existir pelo menos um item listado.
<b>Procedimento</b>	1) No menu lateral, clicar em Estoque → Cozinha. 2) Identificar um item listado. 3) Clicar no botão "Editar " no item. 4) No modal de edição, alterar o campo Quantidade para um novo valor. 5) Clicar no botão "Salvar".
<b>Dados de entrada</b>	Quantidade 20.
<b>Saída esperada</b>	A tabela de itens deve atualizar e mostrar a quantidade 20.
<b>Saída encontrada</b>	O valor do item atualizou de 5 para 20.
<b>Prioridade</b>	Alta
<b>Técnica</b>	Manual

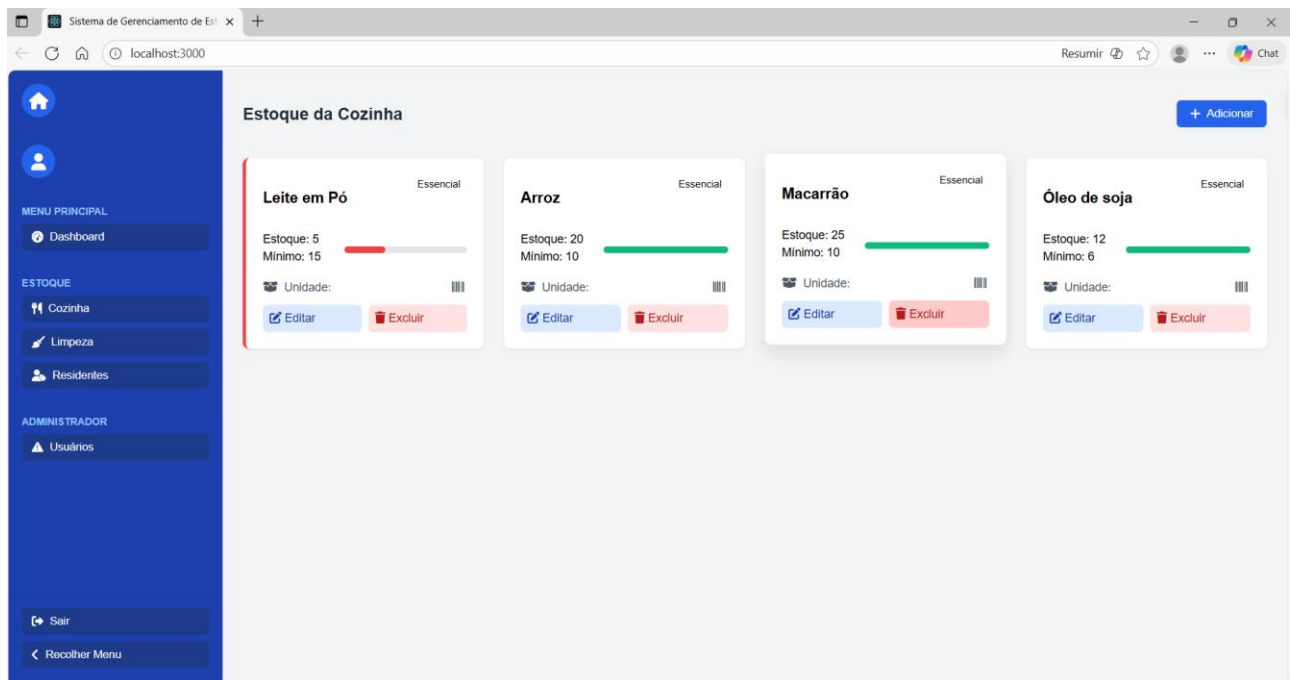
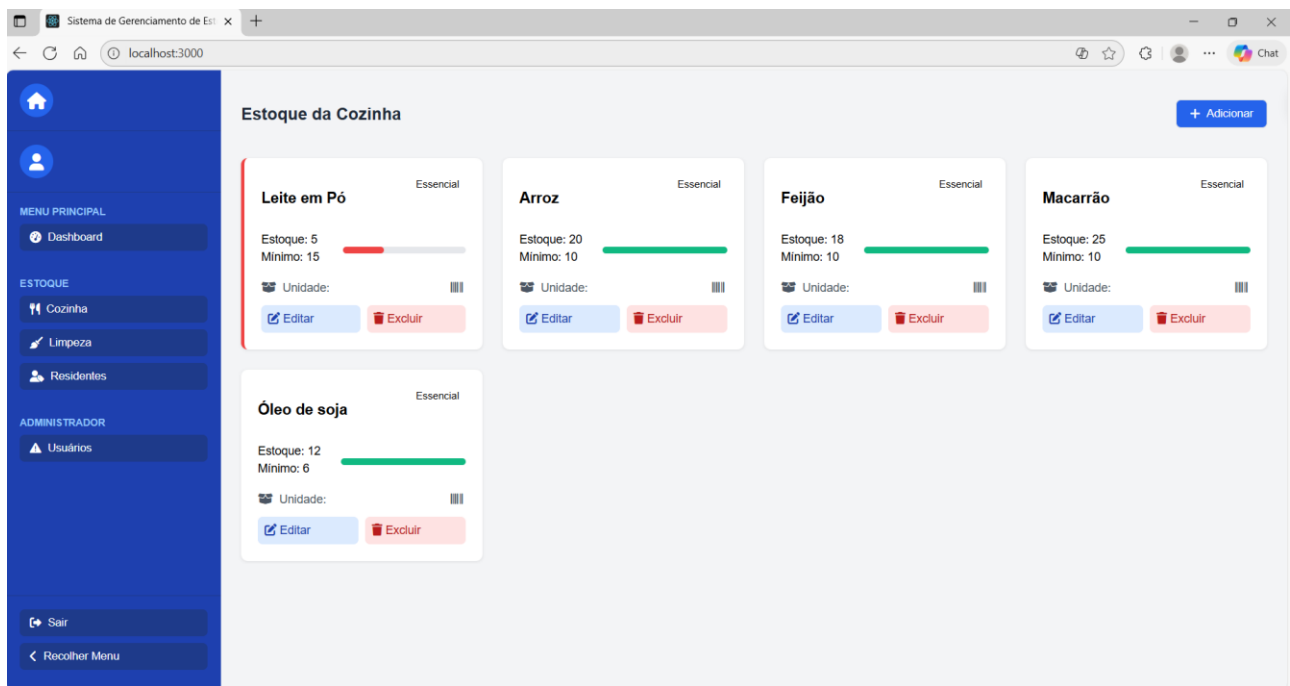
**Print do resultado:**





<b>Caso de Teste</b>	CT07 – Excluir Estoque (Cozinha)
<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Cozinha deve estar aberta; Deve existir pelo menos um item listado.
<b>Procedimento</b>	<ol style="list-style-type: none"> <li>1) No menu lateral, clicar em Estoque → Cozinha.</li> <li>2) Identificar um item listado.</li> <li>3) Clicar no botão Excluir no item.</li> </ol>
<b>Dados de entrada</b>	Clicar no botão Excluir.
<b>Saída esperada</b>	O item deve não aparecer mais na tela.
<b>Saída encontrada</b>	O item não aparece mais na tela.
<b>Prioridade</b>	Alta
<b>Técnica</b>	Manual

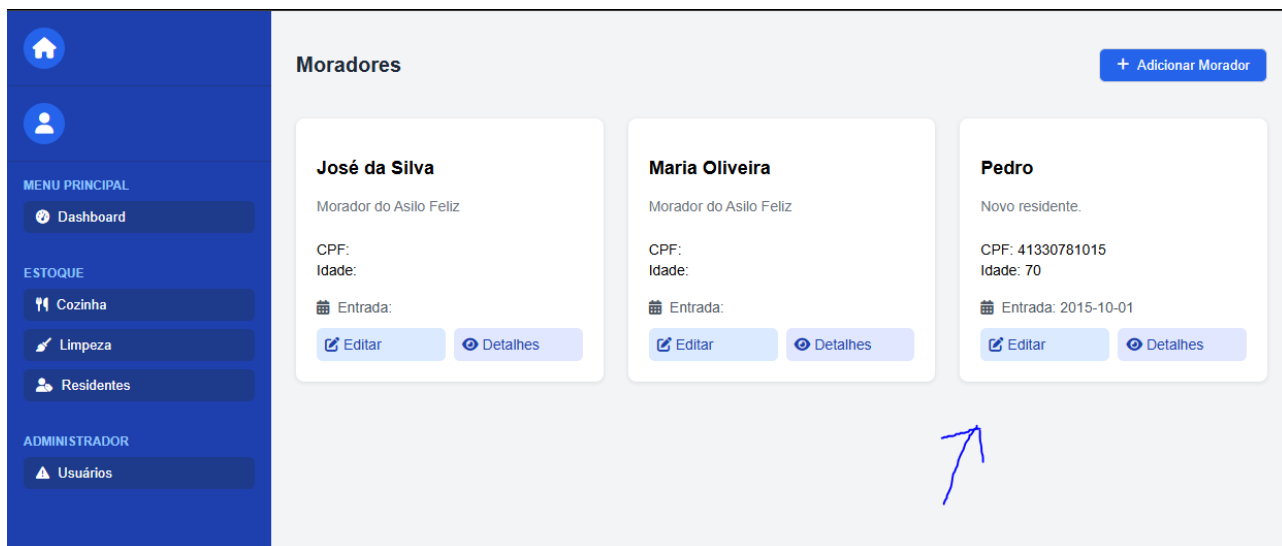
**Print do resultado:**



<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Residentes deve estar aberta;
<b>Procedimento</b>	1) No menu lateral, clicar em Estoque → Residentes. 2) Clicar no botão "+ Adicionar Morador". 3) Preencher os dados do Novo Morador. 4) Clicar no botão "Salvar".
<b>Dados de entrada</b>	Dados do Novo Morador.
<b>Saída esperada</b>	Exibição do Novo Morador na lista.
<b>Saída encontrada</b>	Novo Morador exibido.
<b>Prioridade</b>	Alta.
<b>Técnica</b>	Manual.

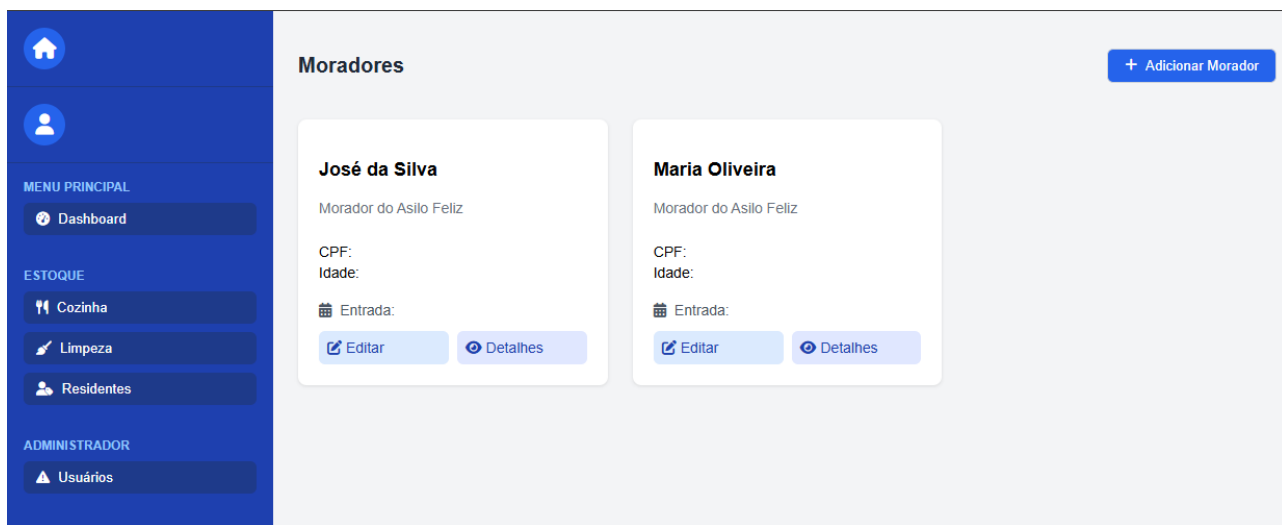
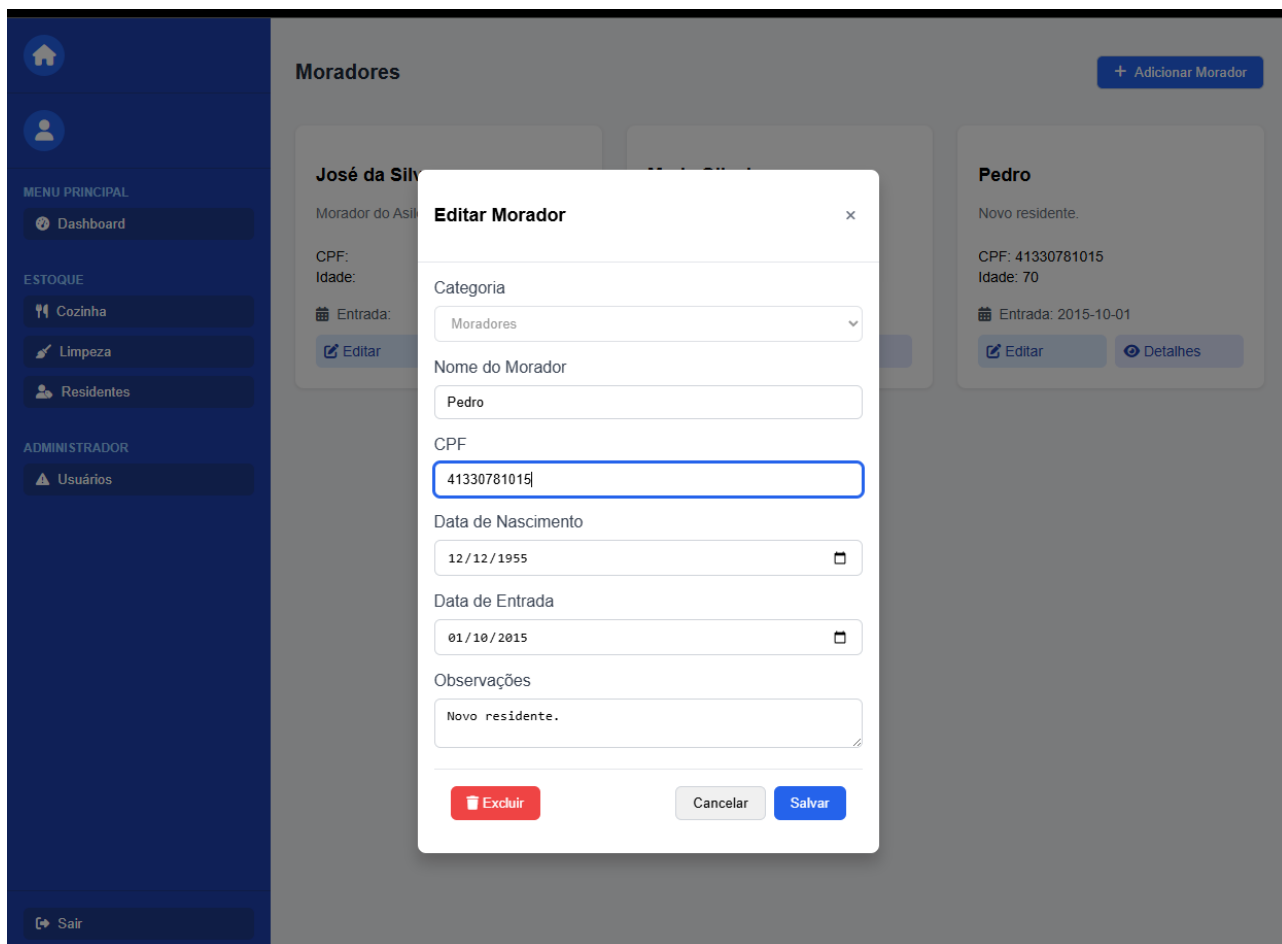
#### Print do resultado:

The screenshot displays a web application interface. On the left is a dark blue sidebar menu with icons for home, user profile, and a 'MENU PRINCIPAL' section containing 'Dashboard'. Below this is an 'ESTOQUE' section with 'Cozinha', 'Limpeza', and 'Residentes'. At the bottom of the sidebar is an 'ADMINISTRADOR' section with 'Usuários' and a 'Sair' button. The main content area is titled 'Moradores' and features a '+ Adicionar Morador' button in the top right. A modal window titled 'Adicionar Novo Morador' is open in the center, containing the following fields: 'Categoria' (dropdown menu with 'Moradores' selected), 'Nome do Morador' (text input with 'Pedro'), 'CPF' (text input with '41330781015'), 'Data de Nascimento' (date picker with '12/12/1955'), 'Data de Entrada' (date picker with '01/10/2015'), and 'Observações' (text area with 'Novo residente.'). At the bottom of the modal are 'Cancelar' and 'Salvar' buttons. In the background, a card for 'José da Silva' is visible, showing fields for 'CPF:', 'Idade:', and 'Entrada:' with an 'Editar' button.



<b>Caso de Teste</b>	CT09 – Excluir um Morador.
<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Residentes deve estar aberta; Deve existir pelo menos um Morador cadastrado.
<b>Procedimento</b>	1) No menu lateral, clicar em Estoque → Residentes. 2) Identificar um Residente. 3) Clicar no botão "Editar". 4) Clicar no botão "Excluir".
<b>Dados de entrada</b>	Clicar no botão "Excluir".
<b>Saída esperada</b>	Exclusão do Morador.
<b>Saída encontrada</b>	Morador excluído.
<b>Prioridade</b>	Média.
<b>Técnica</b>	Manual.

Print do resultado:



### Caso de Teste

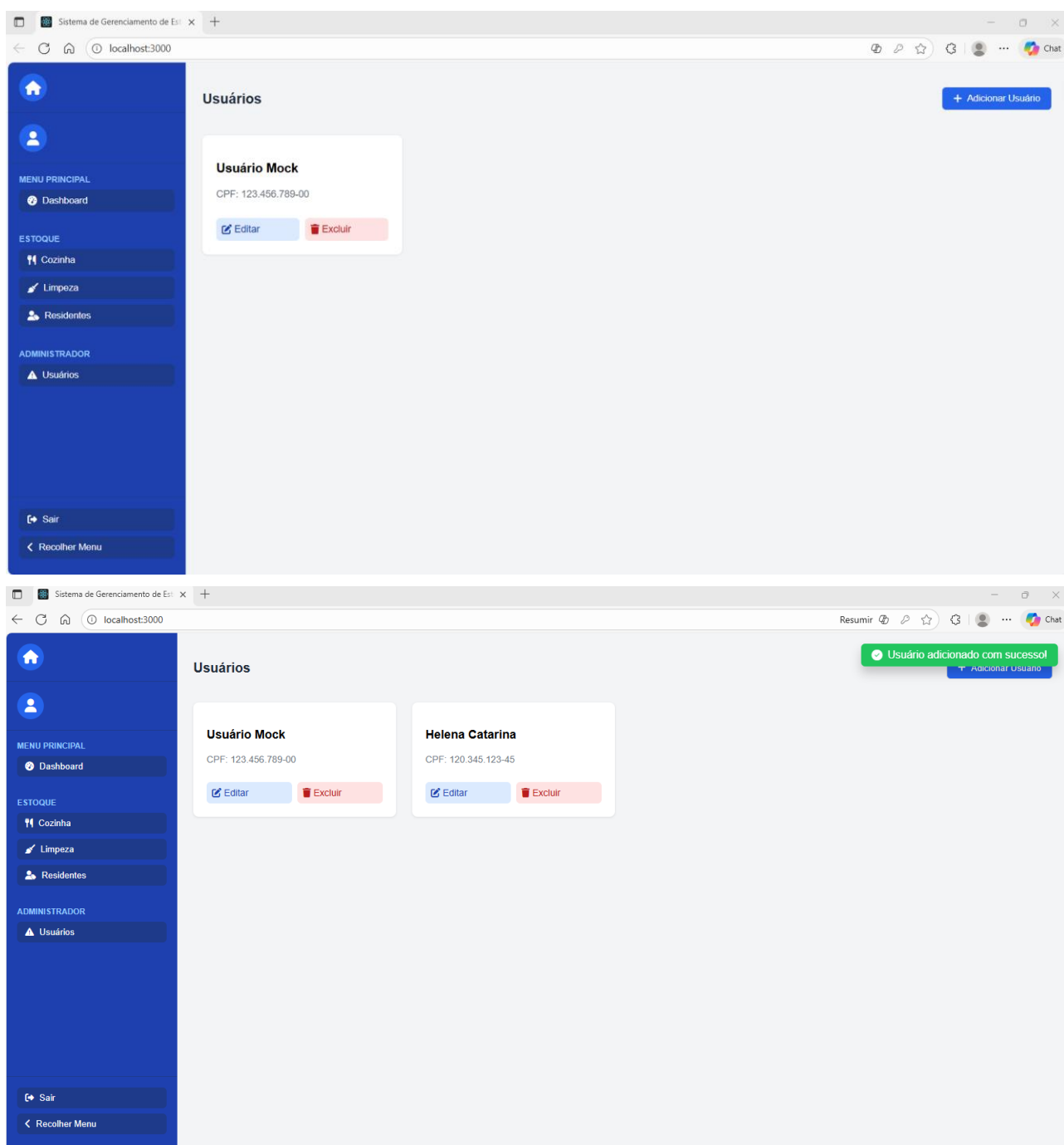
CT10 – Cadastrar Novo Usuário

### Pré-Condição

Usuário deve estar logado no sistema; Tela Administrador  
→ Usuários deve estar aberta;

<b>Procedimento</b>	<ol style="list-style-type: none"> <li>1) No menu lateral, clicar em Administrador → Usuários.</li> <li>2) Clicar no botão "+ Adicionar Usuário".</li> <li>3) Preencher os dados do Novo Usuário.</li> <li>4) Clicar no botão "Salvar".</li> </ol>
<b>Dados de entrada</b>	Dados do novo Usuário.
<b>Saída esperada</b>	Exibição do novo Usuário na lista.
<b>Saída encontrada</b>	Exibição do novo Usuário.
<b>Prioridade</b>	Média.
<b>Técnica</b>	Manual.

## Print do resultado:





### Testes funcionais (Automatizados):

<b>Caso de Teste</b>	CT01 – Realizar login com sucesso
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF válido. 2) Inserir senha válida. 3) Clicar "Entrar".
<b>Dados de entrada</b>	Usuário e senha válidos.
<b>Saída esperada</b>	Entrar na plataforma.
<b>Saída encontrada</b>	Entrar na plataforma.
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

### Print do resultado:

Selenium IDE - Testes GeriCare\*

Projeto: Testes GeriCare\*

Execução ▾

CT01 - Realizar Login com Sucesso\*

http://localhost:3000

	Comando	Alvo	Valor
1.	abrir	/	
2.	clique	id=login	
3.	tipo	id=login	123.456.789-00
4.	clique	id=senha	
5.	tipo	id=senha	123456
6.	clique em	css= ítem-ação-botão	Entrar
7.	Asserto do texto	css=h1	Gerenciamento de Estoque
8.	//		

Comando  #

Alvo

Valor

Descrição

Runs: 0 Falhas: 0

Tela	Referência	
2. clique em id=login	OKEY	11:01:38
3. digite no id=login com o valor 123.456.789-00	OKEY	11:01:38
4. Clique em id=senha	OKEY	11:01:38
5. Digite em id=senha com valor 123456	OKEY	11:01:39
6. clicar em css= ítem-ação-botão com valor Entrar	OKEY	11:01:39
7. assertText on css=h1 com valor Gerenciamento de Estoque	OKEY	11:01:39
CT01 - Realizar Login com Sucesso* concluído com sucesso		11:01:39

<b>Caso de Teste</b>	CT02 – Login com senha incorreta
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF válido. 2) Inserir senha errada. 3) Clicar "Entrar".
<b>Dados de entrada</b>	Senha errada e CPF válido.
<b>Saída esperada</b>	“CPF ou senha inválidos”
<b>Saída encontrada</b>	“CPF ou senha inválidos”
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

## Print do resultado:

Selenium IDE - Testes GeriCare\*

Projeto: Testes GeriCare\*

Testes de busca...

http://localhost:3000

Comando	Alvo	Valor
1 abrir	/	
2 clique	id=login	
3 tipo	id=login	123.456.789-00
4 clique	id=senha	
5 tipo	id=senha	123457
6 clique em	css= item-ação-botão	Entrar
7 Asserto do texto	css= modal-field:nth-child(3)	CPF ou senha inválidos

Comando

Alvo

Valor

Descrição

Tora	Referência	
2. clique em id=login	OKEY	11.03.25
3. digite no id=login com o valor 123.456.789-00	OKEY	11.03.25
4. Clique em id=senha	OKEY	11.03.25
5. Digite em id=senha com valor 123457	OKEY	11.03.25
6. clicar em css= item-ação-botão com valor Entrar	OKEY	11.03.25
7. Tentando encontrar css= modal-field:nth-child(3)	OKEY	11.03.26
*CT02 - Login com senha incorreta* concluído com sucesso		11.03.26

<b>Caso de Teste</b>	CT03 – Login com CPF incorreto
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF errado. 2) Inserir senha válida. 3) Clicar "Entrar".
<b>Dados de entrada</b>	CPF errado e senha válida.
<b>Saída esperada</b>	“CPF ou senha inválidos”
<b>Saída encontrada</b>	“CPF ou senha inválidos”
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

## Print do resultado:

Selenium IDE - Testes GeriCare\*

Projeto: Testes GeriCare\*

Testes de busca...

http://localhost:3000

Comando	Alvo	Valor
1. abrir	/	
2. clique	id=login	
3. tipo	id=login	123.456.789-01
4. clique	id=senha	
5. tipo	id=senha	123456
6. clique em	css=.item-ação-botão	Entrar
7. Asserto do texto	css=.modal-field:nth-child(3)	CPF ou senha inválidos

Comando

Alvo

Valor

Descrição

Tora Referência

2. clique em id=login OKEY 11:04:09

3. Digite no ID=Login com o valor 123.456.789-01 OKEY 11:04:09

4. Clique em ID=senha OKEY 11:04:09

5. Digite em id=senha com valor 123456 OKEY 11:04:10

6. clicar em css=.item-ação-botão com valor Entrar OKEY 11:04:10

7. Tentando encontrar css=.modal-field:nth-child(3)... OKEY 11:04:10

\*CT03 - Login com CPF incorreto\* concluído com sucesso 11:04:10

<b>Caso de Teste</b>	CT04 – Login com CPF e senha incorretos
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF errado. 2) Inserir senha errada.

3) Clicar "Entrar ".	
<b>Dados de entrada</b>	CPF e senha incorretos
<b>Saída esperada</b>	“CPF ou senha inválidos”
<b>Saída encontrada</b>	“CPF ou senha inválidos”
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

## Print do resultado:

**Selenium IDE - Testes GeriCare\***

Projeto: Testes GeriCare\*

Testes de busca...

CT01 - Realizar Login com Sucesso\*

CT02 - Login com senha incorreta

CT03 - Login com CPF incorreto

**CT04 - Login com CPF e senha incorretos**

CT05 - Realizar Logout

CT06 - Editar Estoque (Cozinha)

CT07 - Excluir Estoque (Cozinha)

CT08 - Adicionar Novo Morador

CT09 - Excluir um Morador

CT10 - Cadastrar Novo Usuário

Comando	Alvo	Valor
1. abrir	/	
2. clique	id=login	
3. tipo	id=login	123.456.789-02
4. clique	id=senha	
5. tipo	id=senha	123459
6. clique em	css= item-ação-botão	Entrar
7. Asserto do texto	css= modal-field:nth-child(3)	CPF ou senha inválidos

Comando:  #

Alvo:

Valor:

Descrição:

Tora Referência

2. clique em id=login OKEY 11:04:48

3. Digite no ID=Login com o valor 123.456.789-02 OKEY 11:04:48

4. Clique em ID=senha OKEY 11:04:48

5. Digite em id=senha com valor 123459 OKEY 11:04:48

6. clicar em css= item-ação-botão com valor Entrar OKEY 11:04:48

7. Tentando encontrar css= modal-field:nth-child(3). OKEY 11:04:49

CT04 - Login com CPF e senha incorretos concluído com sucesso 11:04:49

<b>Caso de Teste</b>	CT05 – Logout
<b>Pré-Condição</b>	Estar Logado.
<b>Procedimento</b>	1) Estar logado. 2) Clicar "Sair ".
<b>Dados de entrada</b>	Clicar no botão "Sair".
<b>Saída esperada</b>	Retorno à tela de login.

<b>Saída encontrada</b>	Retorno à tela de login.
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

## Print do resultado:

Selenium IDE - Testes GeriCare\*

Projeto: Testes GeriCare\*

Testes de busca...

http://localhost:3000

Comando	Alvo	Valor
1. abrir	/	
2. clique	id=login	
3. tipo	id=login	123.456.789-00
4. clique	id=senha	
5. tipo	id=senha	123456
6. clique em	css= item-ação-botão	Entrar
7. //		
8. clique	css= sidebar-nav-item:nth-child(1)	Sair
9. Asserto do texto	css=h3	Login

Comando:  #

Alvo:

Valor:

Descrição:

Tela Referência

3. digite no id=login com o valor 123.456.789-00 OKEY 11.06.01

4. Clique em ID=senha OKEY 11.06.01

5. Digite em id=senha com valor 123456 OKEY 11.06.01

6. clicar em css= item-ação-botão com valor Entrar OKEY 11.06.01

8. Tentando encontrar css= sidebar-nav-item:nth-child(1) ... OKEY 11.06.01

9. assertText on css=h3 com valor Login OKEY 11.06.02

CT05 - Realizar Logout concluído com sucesso 11.06.02

<b>Caso de Teste</b>	CT06 – Editar Estoque (Cozinha)
<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Cozinha deve estar aberta; Deve existir pelo menos um item listado.
<b>Procedimento</b>	<ol style="list-style-type: none"> <li>1) No menu lateral, clicar em Estoque → Cozinha.</li> <li>2) Identificar um item listado.</li> <li>3) Clicar no botão "Editar " no item.</li> <li>4) No modal de edição, alterar o campo Quantidade para um novo valor.</li> </ol>

	5) Clicar no botão "Salvar".
<b>Dados de entrada</b>	Quantidade 20.
<b>Saída esperada</b>	A tabela de itens deve atualizar e mostrar a quantidade 20.
<b>Saída encontrada</b>	O valor do item atualizou de 5 para 20.
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

## Print do resultado:

Selenium IDE - Testes GeriCare\*

Projeto: Testes GeriCare\*

Testes de busca...

http://localhost:3000

Comando	Alvo	Valor
4 clique	id=senha	
5 tipo	id=senha	123456
6 clique em	css=.item-ação-botão	Entrar
7 //		
8 clique	css=.sidebar-nav-group:nth-child(2) > .sidebar-nav-item:nth-child(2)	
9 clique	css=.critical-stock .edit	Editar
10 clique	id=itemQuantidade	
11 tipo	id=itemQuantidade	20
12 clique	css=.save	Salvar
13 clique	css=.item-card:nth-child(1) span:nth-child(1)	
14 Asserto do texto	css=.item-card:nth-child(1) span:nth-child(1)	Estoque: 20

Comando: #

Alvo: [X] [Q]

Valor:

Descrição:

Tora Referência

9. clique em css=.critical-stock .edit com valor Editar OKEY 11:06:49

10. clique em id=itemQuantidade OKEY 11:06:49

11. tipo em id=itemQuantidade com valor 20 OKEY 11:06:49

12. clique em css=.save com valor Salvar OKEY 11:06:49

13. Clique em CSS=.item-Card-Nh-Child(1) Span-Nh-Child(1) OKEY 11:06:49

14. assertText em css=.item-card:nth-child(1) span:nth-child(1) com valor Estoque: 20 OKEY 11:06:49

CT06 - Editar Estoque (Cozinha) concluído com sucesso 11:06:49

<b>Caso de Teste</b>	CT07 – Excluir Estoque (Cozinha)
<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Cozinha deve estar aberta; Deve existir pelo menos um item listado.
<b>Procedimento</b>	1) No menu lateral, clicar em Estoque → Cozinha. 2) Identificar um item listado.

	3) Clicar no botão Excluir no item.
<b>Dados de entrada</b>	Clicar no botão Excluir.
<b>Saída esperada</b>	O item deve não aparecer mais na tela.
<b>Saída encontrada</b>	O item não aparece mais na tela.
<b>Prioridade</b>	Alta
<b>Técnica</b>	Automatizada

## Print do resultado:

Selenium IDE - Testes GeriCare\*

Projeto: Testes GeriCare\*

Testes de busca... http://localhost:3000

Comando	Alvo	Valor
1. abrir	/	
2. clique	id=login	
3. tipo	id=login	123.456.789-00
4. clique	id=senha	
5. tipo	id=senha	123456
6. clique em	css= item-ação-botão	Entrar
7. //		
8. clique	css= sidebar-nav-group:nth-child(2) > sidebar-nav-item:nth-child(2)	
9. clique	css= item-card:nth-child(3) delete	
10. assert not text	css= item-card:nth-child(3) h3	Feijão

Comando:  #

Alvo:

Valor:

Descrição:

Tela	Referência	
4.	Clique em ID=senha OKEY	11:07:41
5.	Digite em id=senha com valor 123456 OKEY	11:07:41
6.	clicar em css= item-ação-botão com valor Entrar OKEY	11:07:41
8.	Tentando encontrar css= sidebar-nav-group:nth-child(2) > sidebar-nav-item:nth-child(2) OKEY	11:07:41
9.	clicar em CSS= item-card:nth-child(3) delete OKEY	11:07:42
10.	assertNotText on css= item-card:nth-child(3) h3 com valor Feijão OKEY	11:07:42
*CT07 - Excluir Estoque (Cozinha) concluído com sucesso		11:07:42

<b>Caso de Teste</b>	CT08 – Adicionar Novo Morador
<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Residentes deve estar aberta;
<b>Procedimento</b>	1) No menu lateral, clicar em Estoque → Residentes. 2) Clicar no botão "+ Adicionar Morador". 3) Preencher os dados do Novo Morador. 4) Clicar no botão "Salvar".
<b>Dados de entrada</b>	Dados do Novo Morador.
<b>Saída esperada</b>	Exibição do Novo Morador na lista.
<b>Saída encontrada</b>	Novo Morador exibido.

<b>Prioridade</b>	Alta.
<b>Técnica</b>	Automatizada

**Print do resultado:**

**Selenium IDE - Testes GeriCare\***

Projeto: Testes GeriCare\*

Testes de busca...

Comando | Alvo | Valor

17	tipo	id=residenteNascimento	1995-12-12
18	clique	id=residentEntry	
19	clique	id=residentEntry	
20	tipo	id=residentEntry	0002-12-01
21	tipo	id=residentEntry	0020-12-01
22	tipo	id=residentEntry	0202-12-01
23	tipo	id=residentEntry	2025-12-01
24	clique	id=residentesObservacoes	
25	tipo	id=residentesObservacoes	Está precisando de cuidados + que especiais.
26	clique	css= save	
27	Asserto do texto	css= .item-card:nth-child(3) h3	Ana Flavia

Comando:  #

Alvo:

Valor:

Descrição:

Tora Referência

22. type on id=residentEntry com valor 0202-12-01 OKEY 11:08:21

23. digite no id=residenteEntrada com valor 2025-12-01 OKEY 11:08:21

24. clique em id=residenteObservacoes OKEY 11:08:21

25. type on id=residentObservacoes com valor Está precisando de cuidados + que especiais. OKEY 11:08:21

26. Clique em CSS= save OKEY 11:08:22

27. assertText on css= .item-card:nth-child(3) h3 com valor Ana Flavia OKEY 11:08:22

CT08 - Adicionar Novo Morador concluído com sucesso 11:08:22

<b>Caso de Teste</b>	CT09 – Excluir um Morador.
<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Residentes deve estar aberta; Deve existir pelo menos um Morador cadastrado.
<b>Procedimento</b>	<ol style="list-style-type: none"> <li>1) No menu lateral, clicar em Estoque → Residentes.</li> <li>2) Identificar um Residente.</li> <li>3) Clicar no botão "Editar".</li> <li>4) Clicar no botão "Excluir".</li> </ol>
<b>Dados de entrada</b>	Clicar no botão "Excluir".
<b>Saída esperada</b>	Exclusão do Morador.
<b>Saída encontrada</b>	Morador excluído.
<b>Prioridade</b>	Média.



## Print do resultado:

Selenium IDE - Testes GeriCare\*

Projeto: Testes GeriCare\*

Testes de busca... Execute teste de corrente Ctrl+R

Comando	Alvo	Valor
1. abrir	/	
2. clique	id=login	
3. tipo	id=login	123.456.789-00
4. clique	id=senha	
5. tipo	id=senha	123456
6. clique em	css= ítem-ação-botão	Entrar
7. //		
8. clique	css= sidebar-nav-ítem-nth-child(4) > span	
9. clique	css= ítem-card-nth-child(2).edit	Editar
10. clique	css= delete	Excluir
11. assert not text	css= ítem-card-header	Maria Oliveira e Morador do Asilo Feliz

Comando:  Alvo:  Valor:  Descrição:

Tora Referência

5. Digite em id=senha com valor 123456 OKEY 11:08:52

6. clicar em css= ítem-ação-botão com valor Entrar OKEY 11:08:52

8. Tentando encontrar css= sidebar-nav-ítem-nth-child(4) > span... OKEY 11:08:53

9. clique em css= ítem-card-nth-child(2).editar com valor Editar OKEY 11:08:53

10. clique em css= delete com o valor Excluir OKEY 11:08:53

11. assertNotText on css= ítem-card-header com valor Maria Oliveira e Morador do Asilo Feliz OKEY 11:08:53

CT09 - Excluir um Morador\* concluído com sucesso 11:08:54

<b>Caso de Teste</b>	CT10 – Cadastrar Novo Usuário
<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Administrador → Usuários deve estar aberta;
<b>Procedimento</b>	1) No menu lateral, clicar em Administrador → Usuários. 2) Clicar no botão "+ Adicionar Usuário". 3) Preencher os dados do Novo Usuário. 4) Clicar no botão "Salvar".
<b>Dados de entrada</b>	Dados do novo Usuário.
<b>Saída esperada</b>	Exibição do novo Usuário na lista.
<b>Saída encontrada</b>	Exibição do novo Usuário.
<b>Prioridade</b>	Média.

## Print do resultado:

Selenium IDE - Testes GeriCare\*

Projeto: Testes GeriCare\*

Testes ▾

Testes de busca...

7 //

8 clique css=sidebar-nav-group:nth-child(3) span

9 clique css=section-button

10 clique id=userCPF

11 tipo id=userCPF 120.345.123-45

12 clique id=userPassword

13 tipo id=userPassword 123456

14 clique id=userName

15 tipo id=userName Helena Catarina

16 clique css=save

17 Asserto do texto css=.item-card:nth-child(2) h3 Helena Catarina

Comando

Alvo

Valor

Descrição

Tora Referência

12. clique em id=userPassword OKEY 11:09:28

13. escreva em id=userPassword com o valor 123456 OKEY 11:09:28

14. clique em id=nome do usuário OKEY 11:09:28

15. type on id=userName com valor Helena Catarina OKEY 11:09:28

16. Clique em CSS= save OKEY 11:09:28

17. assertText on css=.item-card:nth-child(2) h3 com valor Helena Catarina OKEY 11:09:28

CT10 - Cadastrar Novo Usuário concluído com sucesso 11:09:29