



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

# GeriCare

---

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

---

## **DESCRIÇÃO DO DOCUMENTO**

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

## HISTÓRICO DE VERSÕES DO DOCUMENTO

# 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	DESCRÍÇÃO 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>ITENS-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)

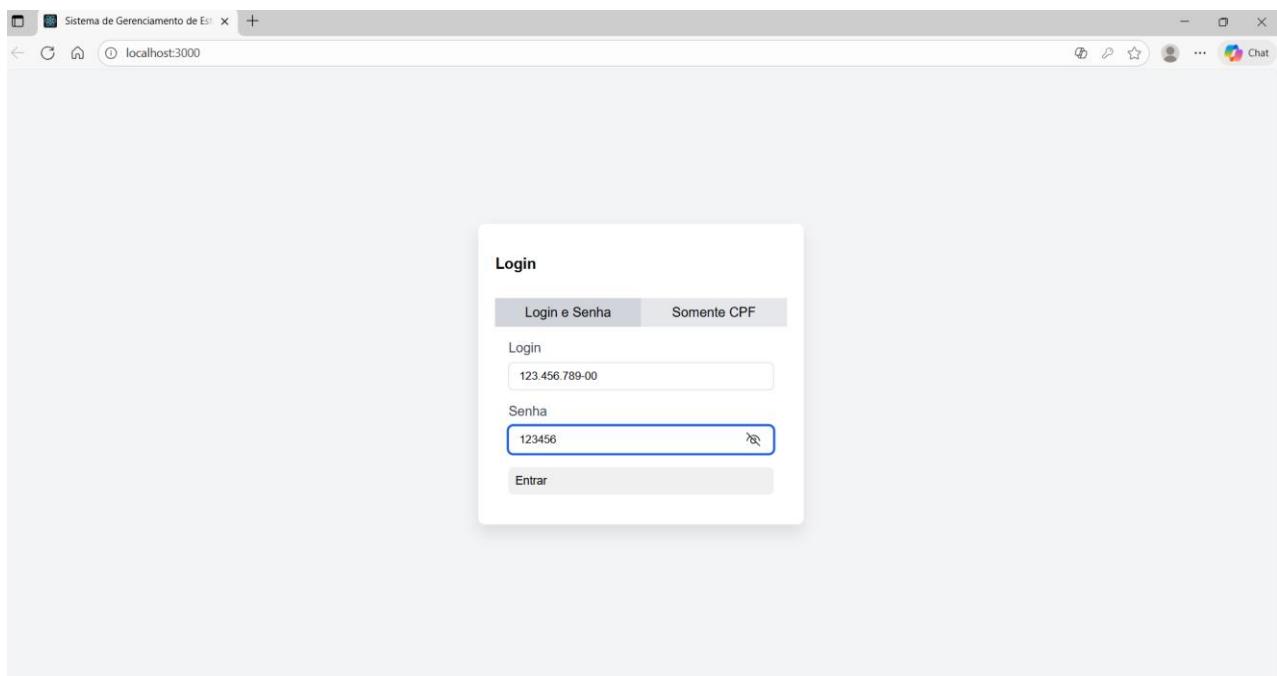
Test Suites: 1 passed, 1 total
Tests:       12 passed, 12 total
Snapshots:  0 total
Time:        3.779 s
Ran all test suites.

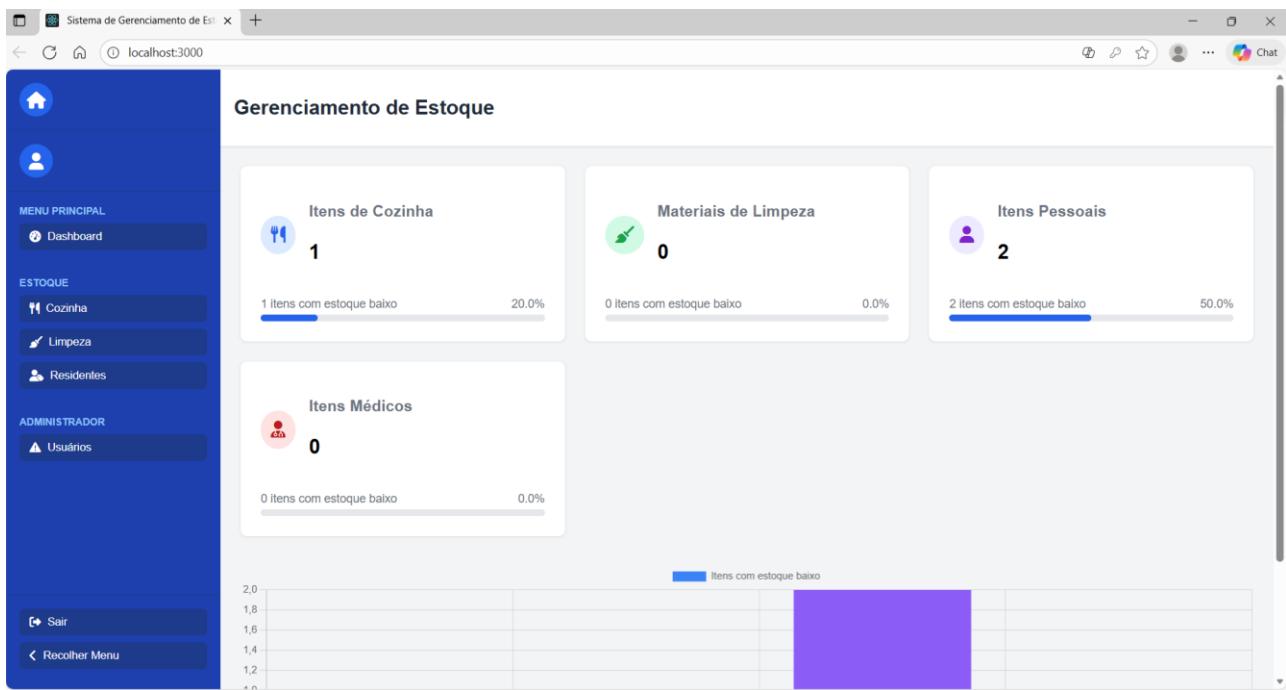
Watch Usage
→ Press f to run only failed tests
```

**Testes funcionais (manuais):**

<b>Caso de Teste</b>	CT01 – Realizar login com sucesso
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	<ol style="list-style-type: none"><li>1) Inserir CPF válido.</li><li>2) Inserir senha válida.</li><li>3) Clicar "Entrar".</li></ol>
<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

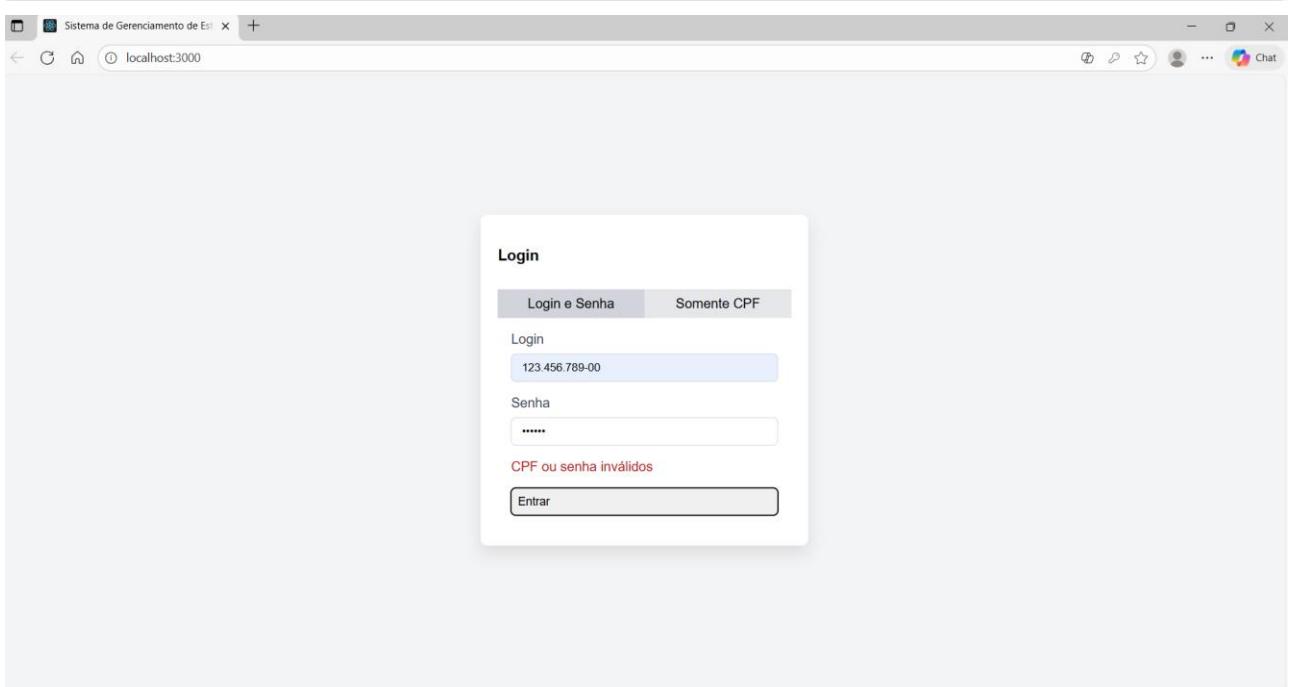
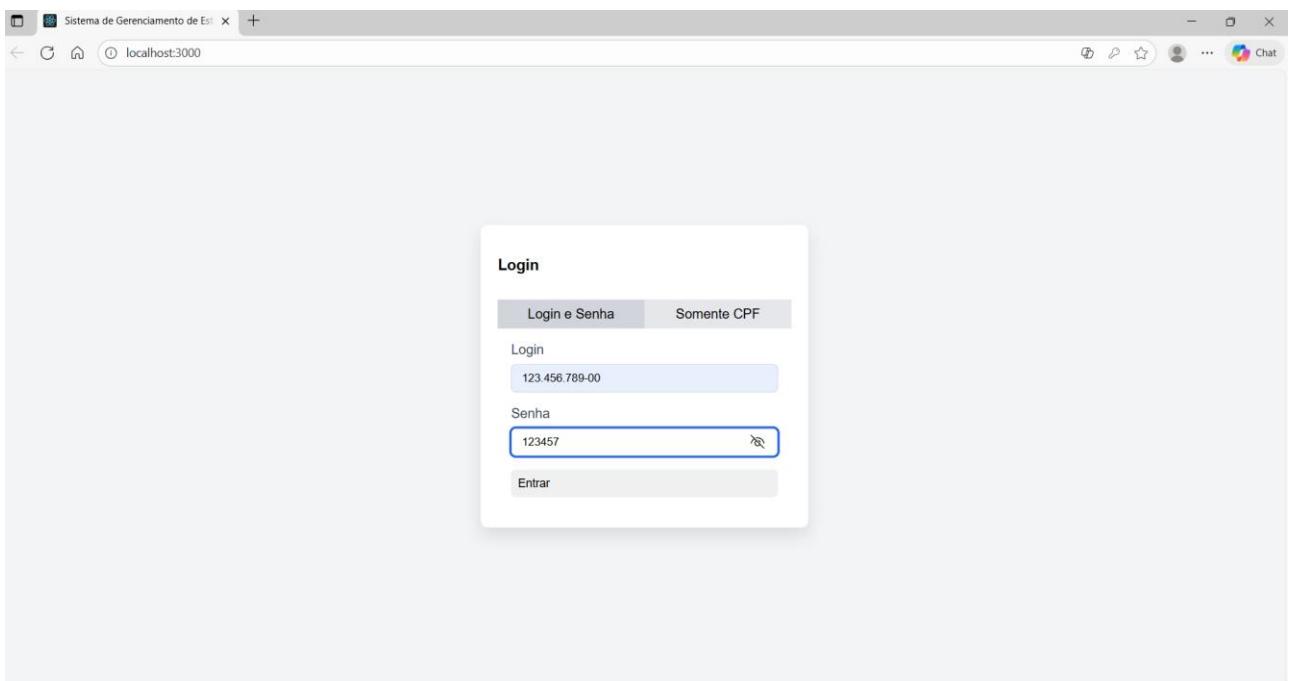
**Print do resultado:**





<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:

The image contains two screenshots of a web browser window titled "Sistema de Gerenciamento de Estoques". Both screenshots show a "Login" form with two tabs: "Login e Senha" (selected) and "Somente CPF".

**Top Screenshot (Successful Login):**

- Username input field: 123 456 789-01
- Password input field: 123456
- Submit button: Entrar

**Bottom Screenshot (Unsuccessful Login):**

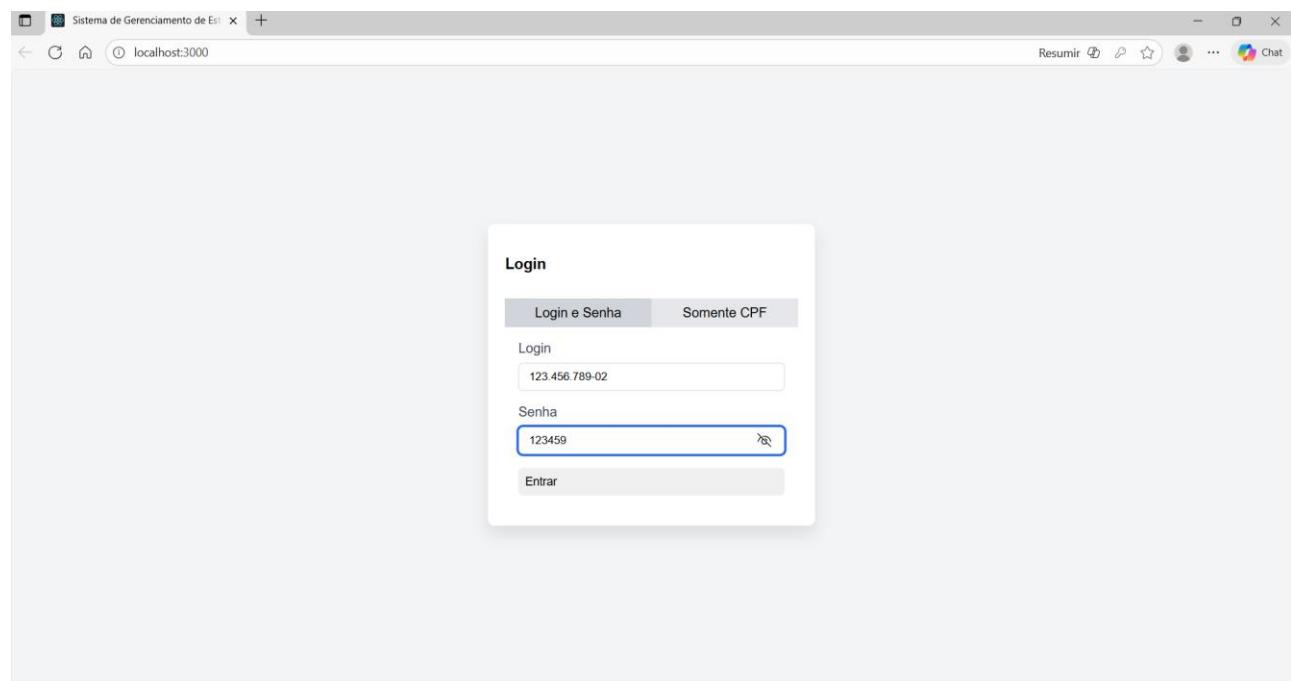
- Username input field: 123 456 789-01
- Password input field: ..... (redacted)
- Error message: CPF ou senha inválidos
- Submit button: Entrar

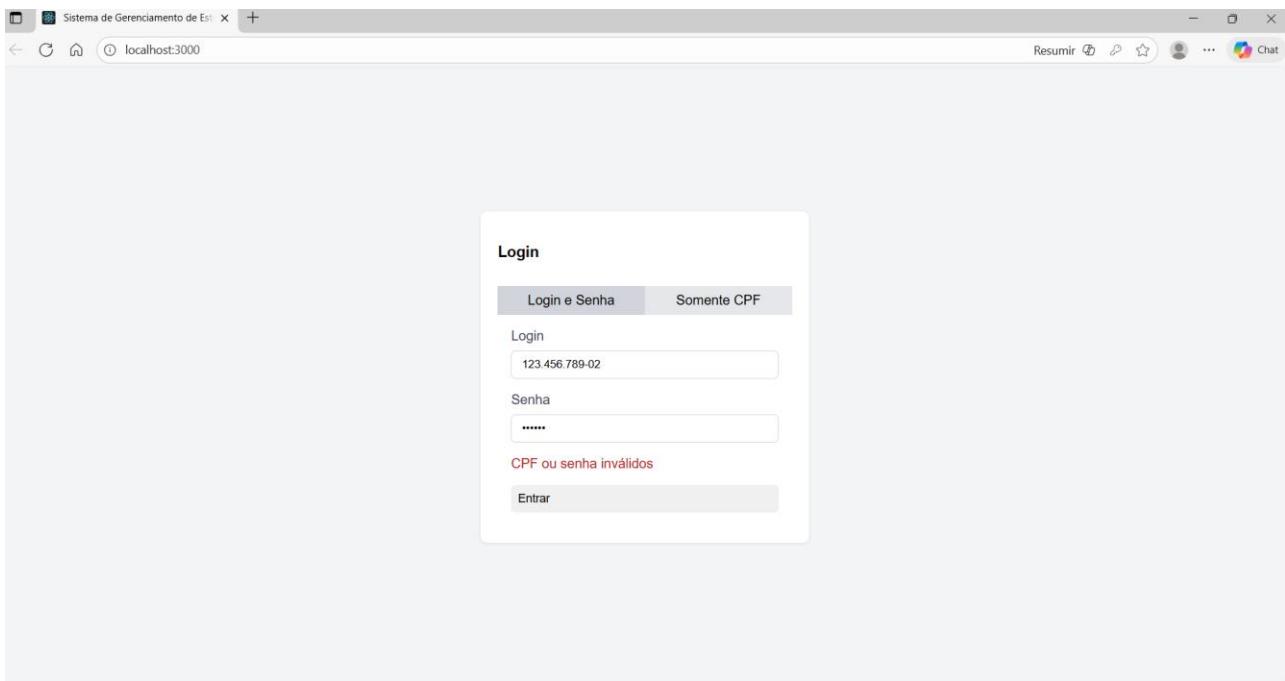
<b>Caso de Teste</b>	CT04 – Login com CPF e senha incorretos
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	1) Inserir CPF errado. 2) Inserir senha errada.

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

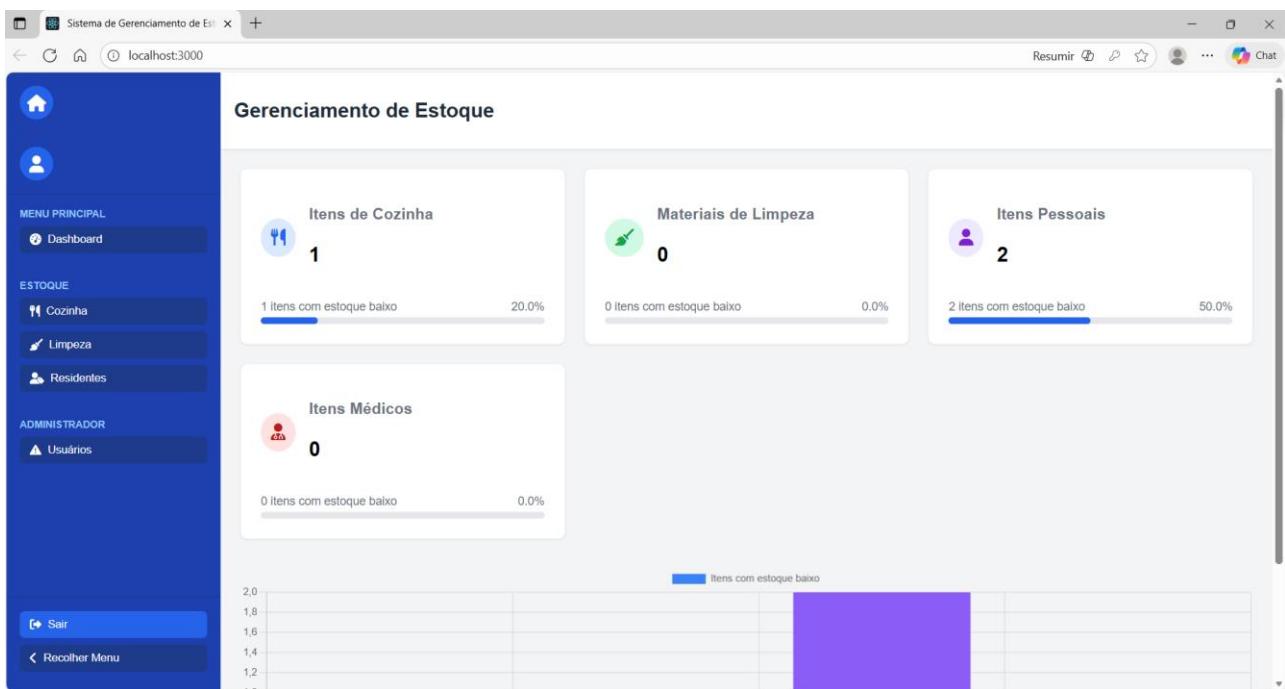
### Print do resultado:

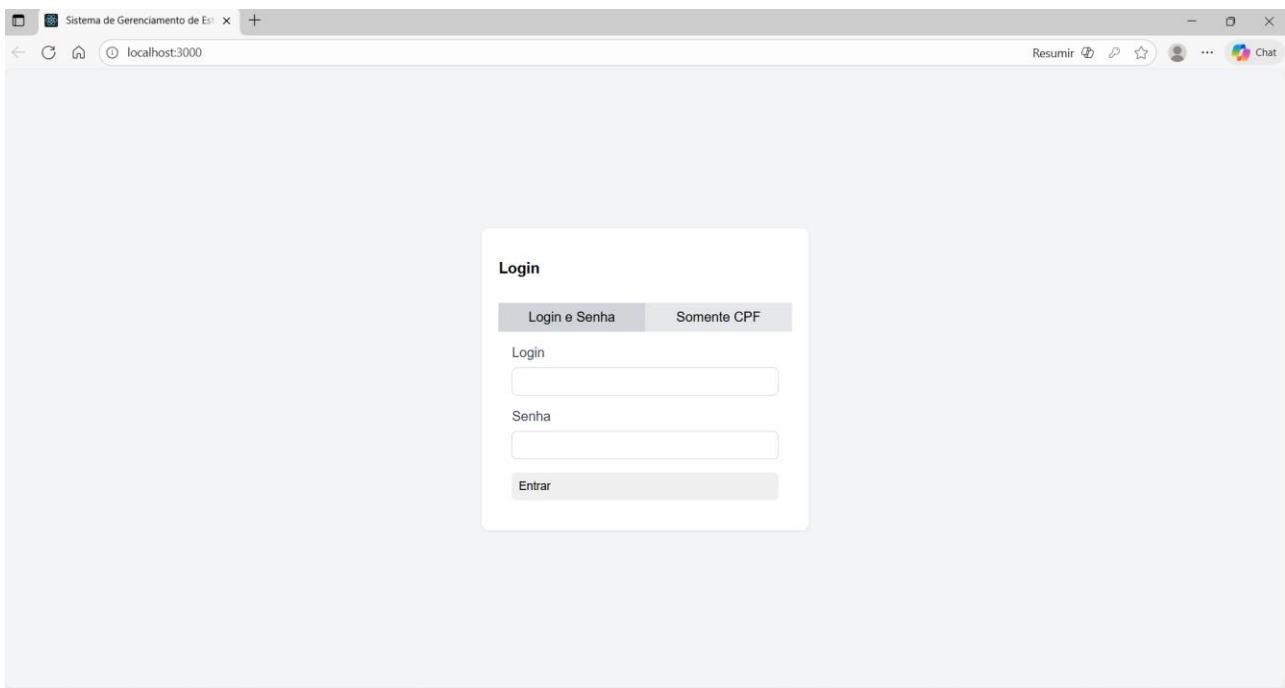




<b>Caso de Teste</b>	CT05 – Logout
<b>Pré-Condição</b>	Estar Logado.
<b>Procedimento</b>	1) Estar logado. 2) Clicar "Sair".
<b>Dados de entrada</b>	Clicar no botão "Sair".
<b>Saída esperada</b>	Retorno à tela de login.
<b>Saída encontrada</b>	Retorno à tela de login.
<b>Prioridade</b>	Alta
<b>Técnica</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>	<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><li>5) Clicar no botão "Salvar".</li></ol>
<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:**

**Estoque da Cozinha**

Item	Status	Estoque	Mínimo
Leite em Pó	Essencial	5	15
Arroz	Essencial	20	10
Feijão	Essencial	18	10
Macarrão	Essencial	25	10
Óleo de soja	Essencial	12	6

**MENU PRINCIPAL**

- Dashboard
- ESTOQUE
  - Cozinha
  - Limpeza
- Residentes
- ADMINISTRADOR
  - Usuários

**Sair**

Sistema de Gerenciamento de Estoque

localhost:3000

MENU PRINCIPAL

- Dashboard
- ESTOQUE
  - Cozinha
  - Limpeza
  - Residentes
- ADMINISTRADOR
  - Usuários

← Sair

< Recolher Menu

Estoque da Cozinha

+ Adicionar

**Leite em Pó** Essencial

Estoque: 5 Minimo: 15

Unidade:

**Óleo de soja** Essencial

Estoque: 12 Minimo: 6

Unidade:

**Macarrão** Essencial

Estoque: 25 Minimo: 10

Unidade:

**Editar Item**

Categoria: Cozinha

Nome do Item: Leite em Pó

Descrição: Descrição detalhada do item

Quantidade: 20 Quantidade Mínima: 15

Unidade:  Código de Barras:

Tipo: Perecível

<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:**

Sistema de Gerenciamento de Estoque

localhost:3000

Estoque da Cozinha

+ Adicionar

MENU PRINCIPAL

- Dashboard
- ESTOQUE
- Cozinha
- Limpeza
- Residentes

ADMINISTRADOR

- Usuários

Sair

Recolher Menu

Leite em Pó

Essencial

Estoque: 5  
Mínimo: 15

Unidade:

Editar Excluir

Arroz

Essencial

Estoque: 20  
Mínimo: 10

Unidade:

Editar Excluir

Feijão

Essencial

Estoque: 18  
Mínimo: 10

Unidade:

Editar Excluir

Macarrão

Essencial

Estoque: 25  
Mínimo: 10

Unidade:

Editar Excluir

Óleo de soja

Essencial

Estoque: 12  
Mínimo: 6

Unidade:

Editar Excluir

Sistema de Gerenciamento de Estoque

localhost:3000

Resumir

Estoque da Cozinha

+ Adicionar

MENU PRINCIPAL

- Dashboard
- ESTOQUE
- Cozinha
- Limpeza
- Residentes

ADMINISTRADOR

- Usuários

Sair

Recolher Menu

Leite em Pó

Essencial

Estoque: 5  
Mínimo: 15

Unidade:

Editar Excluir

Arroz

Essencial

Estoque: 20  
Mínimo: 10

Unidade:

Editar Excluir

Macarrão

Essencial

Estoque: 25  
Mínimo: 10

Unidade:

Editar Excluir

Óleo de soja

Essencial

Estoque: 12  
Mínimo: 6

Unidade:

Editar Excluir

<b>Pré-Condição</b>	Usuário deve estar logado no sistema; Tela Estoque → Residentes deve estar aberta;
<b>Procedimento</b>	<ol style="list-style-type: none"> <li>1) No menu lateral, clicar em Estoque → Residentes.</li> <li>2) Clicar no botão "+ Adicionar Morador".</li> <li>3) Preencher os dados do Novo Morador.</li> <li>4) Clicar no botão "Salvar".</li> </ol>
<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:

Moradores

+ Adicionar Morador

José da Sily

Adicionar Novo Morador

Morador do Asilo

CPF: 41330781015

Idade: 60

Entrada: 01/10/2015

Editar

Categoria: Moradores

Nome do Morador: Pedro

CPF: 41330781015

Data de Nascimento: 12/12/1955

Data de Entrada: 01/10/2015

Observações: Novo residente.

Cancelar Salvar

Sair

**Moradores**

**José da Silva**  
Morador do Asilo Feliz  
CPF: [REDACTED]  
Idade: [REDACTED]  
Entrada: [REDACTED]  
**Editar** **Detalhes**

**Maria Oliveira**  
Morador do Asilo Feliz  
CPF: [REDACTED]  
Idade: [REDACTED]  
Entrada: [REDACTED]  
**Editar** **Detalhes**

**Pedro**  
Novo residente.  
CPF: 41330781015  
Idade: 70  
Entrada: 2015-10-01  
**Editar** **Detalhes**

<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.
<b>Técnica</b>	Manual.

**Print do resultado:**

**Moradores**

**José da Silva**  
Morador do Asilo Feliz

CPF: 41330781015  
Idade:  
Entrada:

**Editar**

**Editar Morador**

Categoria: Moradores

Nome do Morador: Pedro

CPF: 41330781015

Data de Nascimento: 12/12/1955

Data de Entrada: 01/10/2015

Observações: Novo residente.

**Excluir** **Cancelar** **Salvar**

**Pedro**  
Novo residente.  
CPF: 41330781015  
Idade: 70  
Entrada: 2015-10-01

**Editar** **Detalhes**

**Moradores**

**José da Silva**  
Morador do Asilo Feliz

CPF: 41330781015  
Idade:  
Entrada:

**Editar** **Detalhes**

**Maria Oliveira**  
Morador do Asilo Feliz

CPF: 41330781015  
Idade:  
Entrada:

**Editar** **Detalhes**

<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.
<b>Técnica</b>	Manual.

### Print do resultado:

The screenshot shows a web browser window titled "Sistema de Gerenciamento de Estoque" at "localhost:3000". The left sidebar has a blue theme with icons for home, users, dashboard, inventory, cleaning, and residents. Under "ADMINISTRADOR", "Usuários" is selected. The main content area is titled "Usuários" and shows a card for "Usuário Mock" with CPF: 123.456.789-00, with "Editar" and "Excluir" buttons. A blue button at the top right says "+ Adicionar Usuário".

This screenshot shows the same application interface after adding a new user. The main content area now displays two cards: "Usuário Mock" (CPF: 123.456.789-00) and "Helena Catarina" (CPF: 120.345.123-45), each with "Editar" and "Excluir" buttons. A green success message at the top right says "Usuário adicionado com sucesso!" and a blue button says "+ Adicionar Usuário".

### **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=.item-ação-botão	Entrar
7 Asserto do texto	css=h1	Gerenciamento de Estoque
8 //		

Runs: 0 Falhas: 0

Tora Referência

- 2. clique em id=login OKEY
- 3. digite no id=login com o valor 123.456.789-00 OKEY
- 4. Clique em ID=senha OKEY
- 5. Digite em id=senha com valor 123456 OKEY
- 6. clicar em css=.item-ação-botão com valor Entrar OKEY
- 7. assertText on css=h1 com valor Gerenciamento de Estoque OKEY

'CT01 - Realizar Login com Sucesso' concluído com sucesso

<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	Comando	Alvo	Valor
CT01 - Realizar Login com Sucesso*	abrir	/	
CT02 - Login com senha incorreta	clique	id=login	
	digite	id=login	123.456.789-00
	clique	id=senha	123457
	clique em	css=.item-ação-botão	Entrar
	Asserto de texto	css=.modal-field:nth-child(3)	CPF ou senha inválidos

Comando

Alvo

Valor

Descrição

Tora Referência

2. clique em id=login OKEY  
 3. digite no id=login com o valor 123.456.789-00 OKEY  
 4. Clique em ID=senha OKEY  
 5. Digite em id=senha com valor 123457 OKEY  
 6. clicar em css=.item-ação-botão com valor Entrar OKEY  
 7. Tentando encontrar css=.modal-field:nth-child(3)... OKEY

‘CT02 - Login com senha incorreta’ concluído com sucesso

<b>Caso de Teste</b>	CT03 – Login com CPF incorreto
<b>Pré-Condição</b>	Tela de login aberta
<b>Procedimento</b>	<ol style="list-style-type: none"> <li>1) Inserir CPF errado.</li> <li>2) Inserir senha válida.</li> <li>3) Clicar "Entrar".</li> </ol>
<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:

**Testes** + | **Testes de busca...** | **CT03 - Login com CPF incorreto**

**Projeto:** Testes GeriCare\*

Comando	Alvo	Valor
1 <i>abrir</i>	/	
2 <i>clique</i>	id=login	
3 <i>tipo</i>	id=login	123.456.789-01
4 <i>clique</i>	id=senha	
5 <i>tipo</i>	id=senha	123456
6 <i>clique em</i>	css= item-ação-botão	Entrar
7 <i>Asserto do texto</i>	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>	<ol style="list-style-type: none"> <li>1) Inserir CPF errado.</li> <li>2) Inserir senha errada.</li> </ol>

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 - + D: http://localhost:3000

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\*

Teste	Comando	Alvo	Valor
CT01 - Realizar Login com Sucesso*	abrir	/	
CT02 - Login com senha incorreta	clique	id=login	
CT03 - Login com CPF incorreto	digite	id=login	123.456.789-00
CT04 - Login com CPF e senha incorretos	clique	id=senha	
CT05 - Realizar Logout	digite	id=senha	123456
CT06 - Editar Estoque (Cozinha)	clique em	css=item-ação-botão	Entrar
CT07 - Excluir Estoque (Cozinha)			
CT08 - Adicionar Novo Morador	clique	css= sidebar-nav-item nth-child(1)	Sair
CT09 - Excluir um Morador	Aserto do texto	css=h3	Login
CT10 – Cadastrar Novo Usuário			

Comando:

Alvo:

Valor:

Descrição:

Tora 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:

The screenshot shows the Selenium IDE interface with a test log for 'CT06 - Editar Estoque (Cozinha)'. The log details the steps taken to edit the stock quantity for an item in the kitchen. The steps include logging in, navigating to the kitchen stock page, clicking to edit the quantity, entering the value 20, and saving the changes. The log also shows assertions for the expected stock value of 20.

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 nh-child(1) span nh-child(1)	
14 Asserto do texto	css= item-card nh-child(1) span nh-child(1)	Estoque: 20

**Tora Referência**

```

9. clique em css= critical-stock_edit com valor Editar OKEY
10. clique em id=itemQuantidade OKEY
11. tipo em id=itemQuantidade com valor 20 OKEY
12. clique em css= save com valor Salvar OKEY
13. Clique em CSS= Item-Card Nh-Child(1) Span Nh-Child(1) OKEY
14. assertText em css= item-card nth-child(1) span:nth-child(1) com valor Estoque: 20 OKEY
'CT06 - Editar Estoque (Cozinha)' concluído com sucesso
  
```

<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> </ol>

	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:

The screenshot shows the Selenium IDE interface with a test run for 'CT07 - Excluir Estoque (Cozinha)'. The test steps are listed in the left pane, and the corresponding commands, targets, and values are shown in the right pane. The test was successful.

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

**Tora 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. clique 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>	<ol style="list-style-type: none"> <li>1) No menu lateral, clicar em Estoque → Residentes.</li> <li>2) Clicar no botão "+ Adicionar Morador".</li> <li>3) Preencher os dados do Novo Morador.</li> <li>4) Clicar no botão "Salvar".</li> </ol>
<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:

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	Está precisando de cuidados + que especiais.
25 tipo	id=residentesObservacoes	
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 0002-12-01 OKEY
23. digite no id=residenteEntrada com valor 2025-12-01 OKEY
24. clique em id=residentesObservacoes OKEY
25. type on id=residentesObservacoes com valor Está precisando de cuidados + que especiais. OKEY
26. Clique em CSS= save OKEY
27. assertText on css= item-card:nth-child(3) h3 com valor Ana Flavia OKEY
'CT08 - Adicionar Novo Morador' concluido com sucesso
  
```

<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:

The screenshot shows the Selenium IDE interface with the following details:

- Projecto:** Testes GeriCare\*
- Testes:** A dropdown menu.
- Execute teste de corrente Ctrl+R:** A button.
- Table (Teste de busca...):**

	Comando	Alvo	Valor
CT01 - Realizar Login com Sucesso*	1 abrir	/	
CT02 - Login com senha incorreta	2 clique	id=login	
CT03 - Login com CPF incorreto	3 tipo	id=login	123.456.789-00
CT04 - Login com CPF e senha incorretos	4 clique	id=senha	
CT05 - Realizar Logout	5 tipo	id=senha	123456
CT06 - Editar Estoque (Cozinha)	6 clique em	css=.item-ação-botão	Entrar
CT07 - Excluir Estoque (Cozinha)	7 //	css=.sidebar-nav-item:nth-child(4) > span	
CT08 - Adicionar Novo Morador	8 clique	css=.item-card:nth-child(2).edit	Editar
CT09 - Excluir um Morador	9 clique	css=.delete	Excluir
CT10 - Cadastrar Novo Usuário	10 clique	css=.item-card-header	Maria Oliveira e Morador do Asilo Feliz
	11 assertNotText	css=.item-card-header	
- Form Fields:** Comando, Alvo, Valor, Descrição.
- Log:** Shows a list of executed steps with their status and execution time.

<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>	<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.

**Print do resultado:**

The screenshot shows the Selenium IDE interface with the following details:

- Project:** Testes GeriCare\*
- Test:** CT10 – Cadastrar Novo Usuário
- Status:** Concluído com sucesso
- Test Steps:**
  - 7. // clique
  - 8. clique
  - 9. clique
  - 10. clique
  - 11. tipo
  - 12. clique
  - 13. tipo
  - 14. clique
  - 15. tipo
  - 16. clique
  - 17. Asserto do texto
- Alvo (Target):** css= sidebar-nav-group nth-child(3) span, css= section-button, id=userCPF, id=userCPF, id=userPassword, id=userPassword, id=userName, id=userName, css= save, css= item-card nth-child(2) h3
- Valor (Value):** 120.345.123-45, 123456, Helena Catarina, Helena Catarina
- Log:**
  - 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