

# **Trabalho de Conclusão do Curso Tecnólogo em Sistemas para Internet**

Danilo Antonio Rosa RA 13721

Gabriel Paiva de Almeida RA 13631

Thiago da Silva Felipe RA 13628

**AlberguePro**

**Sistema de Albergue para Instituições de  
Acolhimento à Pessoas em Vulnerabilidade**

2025

UniALFA Umuarama / Paraná

Trabalho de Conclusão do Curso Tecnólogo em Sistemas para Internet.....	1
<b>AlberguePro.....</b>	<b>1</b>
Sistema de Albergue para Instituições de.....	1
Acolhimento à Pessoas em Vulnerabilidade.....	1
<b>1. INTRODUÇÃO.....</b>	<b>5</b>
<b>2. PROJETO DO SISTEMA.....</b>	<b>7</b>
2.1. User Story – Tela de Login.....	7
2.1.1. Critérios.....	7
2.1.2. Regras de negócio:.....	7
2.1.3. Teste de Aceitação Positivo.....	8
2.1.4. Teste de Aceitação Negativo.....	8
2.1.5. Requisitos Não Funcionais.....	8
2.2. User Story – Cadastro de Usuários.....	9
UniALFA Umuarama / Paraná2.2.1. Critérios.....	9
2.2.2. Regras de negócio:.....	10
2.2.3. Teste de Aceitação Positivo.....	10
2.2.4. Teste de Aceitação Negativo.....	10
2.2.5. Requisitos Não Funcionais.....	10
2.3. User Story - Gerência de Usuários.....	11
2.3.1. Critérios.....	12
2.3.2. Regras de negócio:.....	12
2.3.3. Teste de Aceitação Positivo.....	12
2.3.4. Teste de Aceitação Negativo.....	13
2.3.5. Requisitos Não Funcionais.....	13
2.4. User Story – Acesso às Funcionalidades do Sistema.....	14
2.4.1. Critérios.....	14
2.4.2. Regras de negócio.....	14
2.4.3. Teste de Aceitação Positivo.....	15
2.4.4. Teste de Aceitação Negativo.....	15
2.4.5. Requisitos Não Funcionais.....	15
2.5. User Story – Controle de Estoque.....	16
2.5.1. Critérios.....	16
2.5.2. Regras de negócio.....	16
2.5.3. Teste de Aceitação Positivo.....	16
2.5.4. Teste de Aceitação Negativo.....	17
2.5.5. Requisitos Não Funcionais.....	18
2.6. User Story – Busca de Produtos.....	18
2.6.1. Critérios.....	19
2.6.2. Regras de negócio.....	19
2.6.3. Teste de Aceitação Positivo.....	19
2.6.4. Teste de Aceitação Negativo.....	20
2.6.5. Requisitos Não Funcionais.....	20
2.7. User Story – Dar Baixa em Produtos.....	21
2.7.1. Critérios.....	21
2.7.2. Regras de negócio.....	21
2.7.3. Teste de Aceitação Positivo.....	22

2.7.4. Teste de Aceitação Negativo.....	22
2.7.5. Requisitos Não Funcionais.....	23
2.8. User Story – Cadastro de patrimônio.....	24
2.8.1. Critérios.....	24
2.8.2. Regras de negócio:.....	24
2.8.3. Teste de Aceitação Positivo.....	24
2.8.4. Teste de Aceitação Negativo.....	25
2.8.5. Requisitos Não Funcionais.....	25
2.9. User Story – Busca de Patrimônio.....	26
2.9.1. Critérios.....	26
2.9.2. Regras de negócio:.....	26
2.9.3. Teste de Aceitação Positivo.....	26
2.9.4. Teste de Aceitação Negativo.....	27
2.9.5. Requisitos Não Funcionais.....	27
2.10. User Story – Edição de Patrimônio.....	28
2.10.1. Critérios.....	28
2.10.2. Regras de negócio:.....	28
2.10.3. Teste de Aceitação Positivo.....	29
2.10.4. Teste de Aceitação Negativo.....	29
2.10.5. Requisitos Não Funcionais.....	30
2.11. User Story – Cadastro de Acolhidos.....	30
2.11.1. Critérios.....	31
2.11.2. Regras de negócio:.....	31
2.11.3. Teste de Aceitação Positivo.....	31
2.11.4. Teste de Aceitação Negativo.....	32
2.11.5. Requisitos Não Funcionais.....	32
2.12. User Story – Busca de Acolhidos.....	33
2.12.1. Critérios.....	33
2.12.2. Regras de negócio:.....	33
2.12.3. Teste de Aceitação Positivo.....	34
2.12.4. Teste de Aceitação Negativo.....	34
2.12.5. Requisitos Não Funcionais.....	35
2.13. User Story – Edição de Cadastro de Acolhidos.....	36
2.13.1. Critérios.....	36
2.13.2. Regras de negócio:.....	36
2.13.3. Teste de Aceitação Positivo.....	36
2.13.4. Teste de Aceitação Negativo.....	37
2.13.5. Requisitos Não Funcionais.....	37
2.14. User Story – Alocação de Acolhido em Leito.....	38
2.14.1. Critérios.....	38
2.14.2. Regras de negócio:.....	38
2.14.3. Teste de Aceitação Positivo.....	39
2.14.4. Teste de Aceitação Negativo.....	39
2.15. User Story – Pesquisa e Gestão de Leitos.....	40
2.15.1. Critérios.....	40
2.15.2. Regras de negócio:.....	40

2.15.3. Teste de Aceitação Positivo.....	41
2.15.4. Teste de Aceitação Negativo.....	41
2.15.5. Requisitos Não Funcionais.....	42
2.16. User Story – Edição e Desalocação de Leitos.....	42
2.16.1. Critérios.....	43
2.16.2. Regras de negócio:.....	43
2.16.3. Teste de Aceitação Positivo.....	43
2.16.4. Teste de Aceitação Negativo.....	44
2.16.5. Requisitos Não Funcionais.....	44
<b>3. ANEXOS.....</b>	<b>44</b>

# 1. INTRODUÇÃO

A Casa de Passagem que escolhemos para fazer o sistema, é uma instituição que oferece acolhimento temporário as pessoas em situação de rua. Seu objetivo principal é garantir abrigo, alimentação, higiene e apoio psicossocial a esses indivíduos enquanto são encaminhados para serviços mais permanentes, como reinserção familiar, programas sociais ou emprego.

A construção deste sistema surgiu da necessidade de modernizar os processos internos do Albergue e casa de passagem para moradores de rua em Umuarama, tornando-os mais eficientes, transparentes e capazes de fornecer uma visão abrangente da situação dos acolhidos. Percebemos que os dados sobre o histórico de cada pessoa em situação de rua que passa pelo albergue, ficam dispersos em diferentes documentos físicos ou planilhas isoladas, muitas vezes perdidas em salas e gavetas empoeiradas.

O objetivo, neste caso, é de criar uma plataforma centralizada que permita o registro e o acesso facilitado a informações relevantes sobre cada acolhido, desde o seu ingresso até o seu possível desligamento. Esse sistema conterá uma gestão de vagas e leitos, cadastro completo dos acolhidos, controle de patrimônio, controle de estoque: alimento, material de higiene e limpeza. Atualmente, a gestão do Albergue enfrenta alguns desafios que podem impactar diretamente a qualidade do serviço oferecido e a capacidade de reinserção social dos moradores de rua. O sistema "AlberguePro" surge para resolver ou mitigar os seguintes problemas e atender às seguintes necessidades. Falta de Dados Centralizados e Acessíveis: Hoje, muitas informações sobre os acolhidos (histórico, necessidades, acompanhamentos) estão dispersas em diferentes registros, grande maioria manuais. Isso dificulta a visão completa da pessoa atendida e a tomada de decisões rápidas e informadas. O "AlberguePro" centraliza todos os dados em um único local, de fácil acesso e atualizado. Ineficiência na Gestão de Vagas e Recursos: Sem um sistema, o controle de vagas e leitos é feito manualmente, geralmente por meio de planilhas ou cadernos. Isso impossibilita ter uma visão atualizada e precisa da disponibilidade. O sistema automatiza essa gestão, permitindo um controle em tempo real e um planejamento mais eficaz. Burocracia Excessiva nos Processos de Acolhimento: O registro de novos acolhidos, a entrada e saída de pessoas e a emissão de documentos podem ser processos demorados e burocráticos. O sistema agiliza esses procedimentos, reduzindo o tempo de espera e permitindo que a equipe dedique mais tempo ao atendimento direto.

Criamos este sistema pensando em ajudar as pessoas que cuidam de albergues e casas de passagem. Percebemos que, muitas vezes, é complicado organizar informações importantes, como quais camas estão livres, quem está hospedado e o que tem no estoque. A motivação principal foi desenvolver uma ferramenta que pudesse otimizar esses processos, facilitando o trabalho das instituições e, conseqüentemente, melhorando a qualidade do acolhimento oferecido às pessoas em situação de vulnerabilidade.

Esperamos que este Sistema de TI contribua significativamente para o Albergue e as pessoas em situação de rua. A implementação do sistema proporcionará uma gestão mais eficiente e organizada, otimizando o uso de recursos e facilitando a comunicação entre os diferentes setores.

Na questão das funcionalidades do sistema, as áreas que o mesmo cobre são, Gestão de vagas e leitos: Controlar quais leitos estão disponíveis, parcialmente ocupados ou totalmente ocupados, e vincular os acolhidos a cada leito. Cadastro Completo dos Acolhidos: Incluindo dados pessoais, histórico dos acolhidos, (saúde, documentação, escolaridade etc.). Controle de Patrimônio: Controlar os bens patrimoniais e seus estados. Controle de estoque: alimento, material de higiene e limpeza: Buscar, visualizar e dar entrada e baixa nos produtos. O Levantamento de Requisitos foi feito através de entrevista junto à equipe dos profissionais atuantes do albergue. O objetivo foi identificar as necessidades específicas da instituição, nos foi passado que o que mais precisava para um sistema seria o controle de estoque dos alimentos, material de higiene e limpeza, o cadastro de acolhidos, controle de leitos e controle de patrimônio. Após levantamento dos requisitos, analisamos as tecnologias que aprendemos no curso e discutimos qual seria a tecnologia que mais daria certo para o projeto, com isso decidimos usar a linguagem Java, usando o Framework Spring Boot que já integra tudo no mesmo, o Frontend e Backend, facilitando a codificação do mesmo. Sobre a questão de qual banco de dados usar, pensamos no mysql, por ser conhecido por sua confiabilidade, desempenho e facilidade de uso, sendo popular para desenvolvimento de aplicativos web e outras aplicações.

## 2. PROJETO DO SISTEMA



### 2.1. User Story – Tela de Login

**Como** usuário autorizado,

**Eu quero** acessar o sistema informando meu usuário e senha,

**Por quê** ter acesso às funcionalidades internas do sistema com segurança.

#### 2.1.1. Critérios

- O sistema deve validar se todos os campos estejam preenchidos.
- O login deve ser feito apenas se as credenciais estiverem corretas.
- Caso os dados estejam incorretos, o sistema deve exibir uma mensagem de erro clara.
- O campo de senha deve ocultar os caracteres digitados.

#### 2.1.2. Regras de negócio:

- Todos os campos devem ser obrigatórios (usuário e senha).
- O login só será permitido caso o usuário e senha estejam corretos.

### 2.1.3. Teste de Aceitação Positivo

- **CT01** – Login bem sucedido
  - **Descrição:** O sistema deve permitir o acesso com credenciais corretas.
  - **Pré-requisito:** Usuário já deve estar cadastrado no sistema.
  - **Passos:** Digitar usuário e senha corretos.
    - Clicar no botão "Logar".
  - **Resultado Esperado:** Sistema redireciona o usuário para a tela principal.

### 2.1.4. Teste de Aceitação Negativo

- **CT02** – Tentativa de login com senha incorreta
  - **Descrição:** Verificar se o sistema bloqueia o acesso com senha inválida.
  - **Pré-requisito:** Usuário já deve estar cadastrado no sistema.
  - **Passos:** Digitar usuário correto e senha incorreta.
    - Clicar no botão "Logar".
  - **Resultado Esperado:** Sistema exibe mensagem de erro ("Usuário ou senha inválidos").
- **CT03** – Tentativa de login com campos vazios
  - **Descrição:** Verificar se o sistema impede login com algum campo vazio
  - **Pré-requisito:** O usuário está na tela de login.
  - **Passos:** Deixar um campo vazio.
    - Clicar no botão "Logar".
  - **Resultado Esperado:** Sistema exibe mensagem de erro ("Todos os campos são obrigatórios").
- **CT04** – Validar que o botão não envia múltiplas requisições.
  - **Descrição:** Verificar se o sistema impede múltiplas requisições.
  - **Pré-requisito:** O usuário está na tela de login.
  - **Passos:** Deixar um campo vazio.
    - Clicar várias vezes o botão "Logar".
  - **Resultado Esperado:** Apenas uma requisição é enviada e processada.

### 2.1.5. Requisitos Não Funcionais

- **Desempenho:** A autenticação deve ser concluída em até 2 segundos após o envio dos dados.



- **Segurança:** O sistema deve bloquear após 3 tentativas falhas consecutivas por um período de 10 minutos.
- **Escalabilidade:** O sistema deve suportar múltiplos acessos simultâneos sem queda de performance.
- **Manutenibilidade:** O código da autenticação deve ser modular e permitir fácil atualização do mecanismo de segurança.
- **Compatibilidade:** A tela de login deve funcionar corretamente em navegadores modernos (Chrome, Firefox, Edge).
- **Usabilidade:** A interface deve ser simples, com feedback visual em tempo real (ex: campos inválidos destacados).

The screenshot shows a web browser window with the address bar displaying 'http://www.alberguepro.com.br/admin'. The page title is 'AlberguePro'. The main content area is titled 'Administrar Usuarios - Cadastro'. It contains a form with the following elements:

- A label 'Usuario:' followed by a text input field.
- A label 'Tipo:' followed by two radio buttons: 'Usuario' and 'Admin'.
- A label 'Senha:' followed by a password input field.
- Two buttons at the bottom: 'Salvar' and 'Voltar'.

## 2.2. User Story – Cadastro de Usuários

**Como** administrador do sistema,

**Eu quero** cadastrar novos usuários e definir seus tipos (Usuário ou Admin),

**Por quê** preciso gerenciar o acesso ao sistema e garantir que as pessoas certas tenham as permissões adequadas.

### 2.2.1. Critérios

- Deve haver campos para preencher o nome de usuário e a senha.
- Deve ser possível selecionar o tipo de usuário (Usuário ou Admin) através de botões de rádio.
- O botão "Salvar" deve efetivar o cadastro do novo usuário no

sistema

- O botão "Voltar" deve permitir ao administrador retornar à tela anterior sem salvar.

#### 2.2.2. Regras de negócio:

- Os campos "Usuário", "Tipo" e "Senha" são obrigatórios.
- O "Usuário" (nome de usuário) deve ser único no sistema.
- O campo "Tipo" deve ter uma opção padrão pré-selecionada (ex: "Usuário").

#### 2.2.3. Teste de Aceitação Positivo

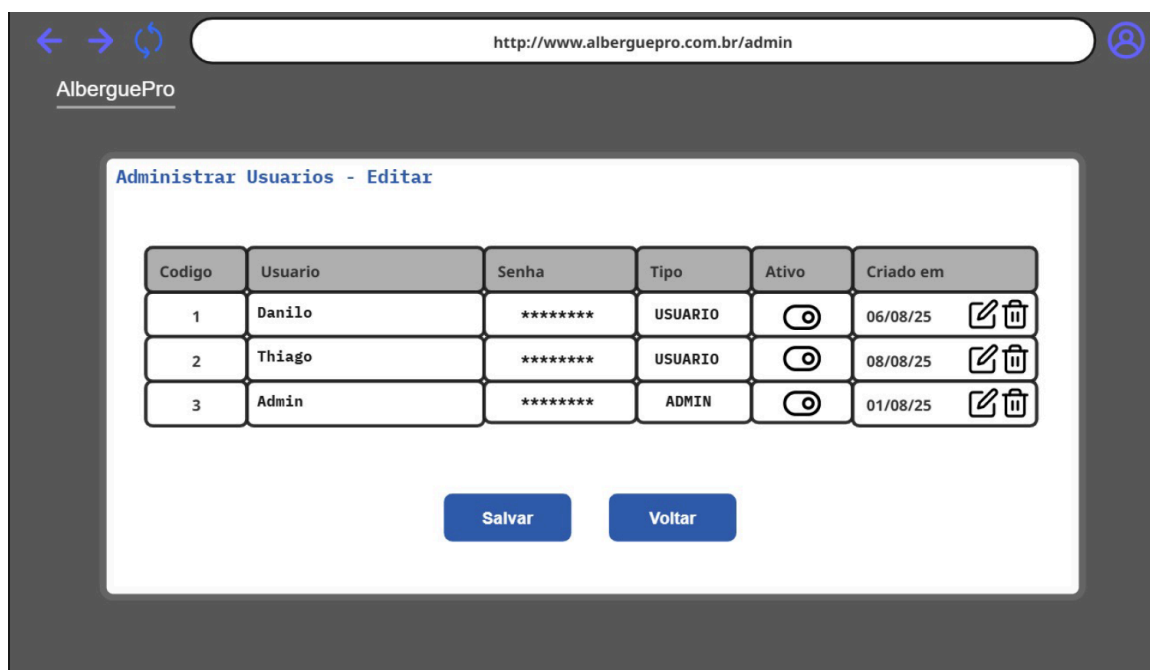
- **CT01** – Cadastro de usuário com sucesso.
  - **Descrição:** Verificar se o sistema permite cadastrar um novo usuário com todas as informações preenchidas corretamente.
  - **Pré-requisito:** Acesso à tela de cadastro de usuários com permissões de administrador.
  - **Passos:** Preencher todos os campos obrigatórios
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** O usuário é cadastrado com sucesso.

#### 2.2.4. Teste de Aceitação Negativo

- **CT02** – Cadastro com campos obrigatórios vazios.
  - **Descrição:** Verificar se o sistema impede o cadastro de um novo usuário com campos obrigatórios vazios.
  - **Pré-requisito:** Acesso à tela de cadastro de usuários.
  - **Passos:** Deixar o campo "Usuário" ou "Senha" vazio.
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** O sistema exibe uma mensagem de erro: "Preencha todos os campos obrigatórios".
- **CT03** – Cadastro com nome de usuário duplicado.
  - **Descrição:** Verificar se o sistema impede o cadastro de um usuário com um nome de usuário já existente.
  - **Pré-requisito:** Ter um usuário cadastrado no sistema.
  - **Passos:** Preencher um nome de usuário que já existe.
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** Sistema exibe mensagem de erro "O nome de usuário já existe".

#### 2.2.5. Requisitos Não Funcionais

- **Desempenho:** A tela de cadastro de usuários deve carregar em no máximo 1 segundo.
- **Segurança:** Apenas usuários com perfil de administrador podem acessar a tela de cadastro. A senha deve ser armazenada de forma segura (criptografada).
- **Escalabilidade:** O sistema deve ser capaz de gerenciar um grande volume de usuários sem degradação do desempenho no login ou no gerenciamento de perfis.
- **Manutenibilidade:** O módulo de administração de usuários deve ser modular para fácil adaptação a novas regras de permissão ou campos no futuro.
- **Compatibilidade:** A tela de login deve funcionar corretamente em navegadores modernos (Chrome, Firefox, Edge).
- **Usabilidade:** O formulário deve ser claro e intuitivo, com mensagens de erro e sucesso amigáveis.



### 2.3. User Story - Gerência de Usuários

**Como** administrador do sistema,

**Eu quero** Gerenciar usuários cadastrados no sistema,

**Por quê** preciso manter o controle de acesso ao sistema e garantir que apenas pessoas autorizadas utilizem a aplicação.

### 2.3.1. Critérios

- O sistema deve exibir uma lista de usuários com as informações: código, nome de usuário, senha (oculta), tipo(Usuário/Admin), status(ativo/inativo) e data de criação.
- Deve ser possível editar, excluir dados dos usuários.
- O botão "Salvar" deve aplicar as alterações feitas.
- O botão "Voltar" deve retornar à tela anterior sem aplicar alterações.

### 2.3.2. Regras de negócio:

- Apenas usuários com perfil Admin podem acessar essa tela.
- O campo de senha deve ser exibido de forma oculta (asteriscos).
- A alteração do status "Ativo/Inativo" deve refletir imediatamente na permissão de login do usuário.

### 2.3.3. Teste de Aceitação Positivo

- **CT01** – Visualização da lista de usuários
  - **Descrição:** Verificar se a lista de usuários é exibida corretamente para um usuário com perfil Admin.
  - **Pré-requisito:** Usuário já deve estar cadastrado no sistema.
  - **Passos:** Estar logado como Admin.
    - Estar na tela gerência de usuários
  - **Resultado Esperado:** Lista completa é exibida com código, nome de usuário, senha oculta, tipo, status e data de criação.
- **CT02** – Alterar status de usuário com sucesso
  - **Descrição:** Verificar se é possível ativar ou desativar um usuário.
  - **Pré-requisito:** Estar logado como Admin
  - **Passos:** Estar logado como Admin.
    - Clicar no botão de "Alterar" do usuário desejado
    - Clicar no botão "Salvar"
  - **Resultado Esperado:** O status é atualizado e salvo no sistema.
- **CT03** – Edição de usuário com sucesso
  - **Descrição:** Verificar se é possível editar os dados de um usuário.
  - **Pré-requisito:** Estar logado como Admin
  - **Passos:** Estar logado como Admin.
    - Clicar no ícone de lápis do usuário.
    - Modificar os dados
    - Clicar no botão "Salvar"
  - **Resultado Esperado:** Os dados são atualizados com sucesso.

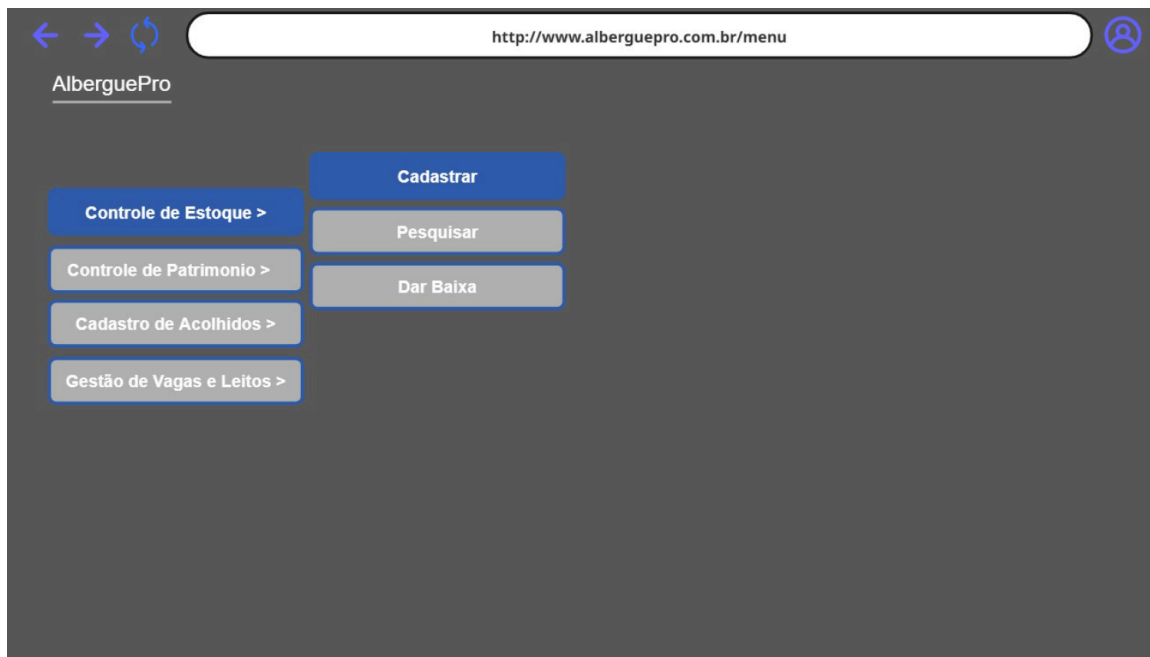
- **CT04** – Exclusão de usuário com sucesso
  - **Descrição:** Verificar se é possível excluir um usuário.
  - **Pré-requisito:** Estar logado como Admin
  - **Passos:** Estar logado como Admin.
    - Clicar no ícone de lixeira de um usuário.
    - Confirmar a exclusão.
  - **Resultado Esperado:** O usuário é removido com sucesso.

#### 2.3.4. Teste de Aceitação Negativo

- **CT04** – Tentativa de editar sem preencher campos obrigatórios
  - **Descrição:** Verificar se o sistema impede a edição se algum campo obrigatório for deixado em branco.
  - **Pré-requisito:** Estar na tela gerência de usuários.
  - **Passos:** Estar logado como Admin.
    - Deixar um campo em branco.
    - Clicar no botão “Salvar”.
  - **Resultado Esperado:** O sistema exibe mensagem de erro: “Todos os campos são obrigatórios”.
- **CT05** – Tentativa de excluir o Admin logado
  - **Descrição:** Verificar se o sistema impede a exclusão do usuário Admin atualmente logado.
  - **Pré-requisito:** Estar na tela gerência de usuários.
  - **Passos:** Estar logado como Admin.
    - Clicar na lixeira do próprio usuário logado.
    - Confirmar a exclusão.
  - **Resultado Esperado:** O sistema exibe mensagem de erro: “Não é possível excluir o usuário logado.”

#### 2.3.5. Requisitos Não Funcionais

- **Desempenho:** A tela deve carregar e exibir os dados em até 2 segundos.
- **Segurança:** Somente usuários com permissão Admin podem acessar essa funcionalidade.
- **Escalabilidade:** Deve suportar mais de 100 usuários sem travamentos.
- **Manutenibilidade:** A lógica de edição, exclusão e ativação deve ser modular para fácil atualização futura.
- **Compatibilidade:** A tela deve funcionar em navegadores modernos (Chrome, Firefox, Edge).
- **Usabilidade:** Os ícones devem ter tooltips explicativos (ex: “Editar”, “Excluir”), e feedback visual após cada ação (ex: mensagens de sucesso ou erro).



## 2.4. User Story – Acesso às Funcionalidades do Sistema

**Como**, usuário do sistema.

**Eu quero** navegar facilmente entre os diferentes módulos e funcionalidades do AlberguePro.

**Porque** assim consigo acessar rapidamente as tarefas que preciso realizar.

### 2.4.1. Critérios

- Deve haver um menu lateral com as principais categorias de funcionalidades (Controle de Estoque, Controle de Patrimônio, Cadastro de Acolhidos, Gestão de Vagas e Leito).
- Ao clicar em uma categoria (ex: Controle de Estoque), as sub-opções relacionadas (Cadastrar, Buscar, Dar baixa) devem ser exibidas.
- Ao clicar em uma sub-opção (ex: Cadastrar de Controle de Estoque), a tela correspondente àquela funcionalidade deve ser carregada.

### 2.4.2. Regras de negócio

- As sub-opções de cada módulo (Cadastrar, Buscar, Dar baixa, etc.) só devem ser visíveis ou clicáveis se o usuário possuir a devida permissão de acesso àquela funcionalidade.
- A página inicial, ao ser carregada, deve exibir o menu e, por padrão, o logo do sistema centralizado.

#### 2.4.3. Teste de Aceitação Positivo

- **CT01** - Navegação para a tela de Cadastro de Estoque
  - **Descrição:** Verificar se é possível navegar do menu principal para a tela de "Controle de Estoque - Cadastrar".
  - **Pré-requisito:** Usuário logado no sistema e na tela inicial do AlberguePro.
  - **Passos/ação:** Clicar na opção "Controle de Estoque" no menu lateral. Clicar na sub-opção "Cadastrar".
  - **Resultado Esperado:** O sistema exibe a tela de "Controle de Estoque - Cadastro".

#### 2.4.4. Teste de Aceitação Negativo.

- **CT02** – Tentativa de acesso a funcionalidade sem permissão
  - **Descrição:** Verificar se o sistema impede o acesso a uma funcionalidade para a qual o usuário não possui permissão.
  - **Pré-requisito:** Usuário logado no sistema com perfil que não tem permissão para "Dar baixa" no estoque.
  - **Passos:** Clicar na opção "Controle de Estoque" no menu lateral. Tentar clicar na sub-opção "Dar baixa".
  - **Resultado Esperado:** A sub-opção "Dar baixa" deve estar desabilitada ou, se clicada, o sistema deve exibir uma mensagem de erro "Você não tem permissão para acessar esta funcionalidade".

#### 2.4.5. Requisitos Não Funcionais

- **Desempenho:** A tela inicial deve carregar em <1s; transição entre telas do menu em <1s.
- **Segurança:** A visibilidade e acesso às opções do menu devem ser controlados por permissões de usuário.
- **Escalabilidade:** O menu deve suportar a adição de novos módulos e sub-opções sem impacto no desempenho.
- **Manutenibilidade:** A estrutura do menu deve ser modular, facilitando a adição/remoção de funcionalidades.
- **Compatibilidade:** Funcionar em navegadores modernos (Chrome, Edge, Firefox) e ser responsivo.
- **Usabilidade:** Menu intuitivo, claro e de fácil navegação para o usuário.

← → ↺ http://www.alberguepro.com.br/cecadastro

AlberguePro

Controle de Estoque - Cadastrar

Tipo: ☐ Alimento ☐ Higiene ☐ Limpeza

Nome:

Quantidade:

Unidade:

Data de Vencimento:

Salvar Voltar

## 2.5. User Story – Controle de Estoque

**Como**, responsável pelo estoque,

**Eu quero** cadastrar e controlar os produtos,

**Porque** assim consigo saber o que está em falta e evitar desperdícios.

### 2.5.1. Critérios

- O formulário não deve permitir salvar com campos vazios.
- A quantidade deve ser um número positivo.
- O tipo de produto deve ser selecionado entre: alimento, higiene, e limpeza.
- Deve ser possível buscar produtos por nome.
- Deve ser possível filtrar os produtos por tipo.
- Deve ser permitido dar baixa em produtos com quantidade positiva.

### 2.5.2. Regras de negócio

- Não deve ser permitido dar baixa maior do que a quantidade disponível em estoque.
- Não deve ser permitido cadastrar produtos com nome duplicado para o mesmo tipo.

### 2.5.3. Teste de Aceitação Positivo

- **CT01** - Cadastro válido de produto



- **Descrição:** Verificar se é possível cadastrar um produto com os campos preenchidos válidos.
- **Pré-requisito:** Usuário está na tela de cadastro de produtos.
- **Passos/ação:** preencher todos os campos corretamente.
  - Clicar no botão “Salvar”.
- **Resultado Esperado:** O produto é salvo com sucesso.

#### 2.5.4. Teste de Aceitação Negativo.

- **CT02 – Cadastro com quantidade negativa**
  - **Descrição:** Verifica se o sistema impede o cadastro com quantidade negativa.
  - **Pré-requisito:** Acesso à tela de cadastro.
  - **Passos:** Acessar a tela de cadastro.
    - Preencher nome e tipo, e colocar quantidade -1.
    - Clicar em “Salvar Produto”.
  - **Resultado Esperado:** O sistema exibe mensagem de erro: “Não é permitido salvar produtos com unidades negativas”.
- **CT03 – Cadastro com campos vazios**
  - **Descrição:** Verificar se o sistema impede o cadastro de um produto com campos obrigatórios vazios.
  - **Pré-requisito:** Usuário estar na tela de cadastro de produtos.
  - **Passos:** Deixar um campo obrigatório vazio.
    - Clicar no botão “Salvar”.
  - **Resultado Esperado:** O sistema exibe mensagem de erro: “Preencha todos os campos obrigatórios”.
- **CT04 – Cadastro de produto com o mesmo nome de um produto já cadastrado**
  - **Descrição:** Verificar se o sistema impede o cadastro de um produto com o mesmo de um produto já cadastrado.
  - **Pré-requisito:** Ter um produto já cadastrado (ex: nome: arroz).
  - **Passos:** Na tela de cadastro.
    - Preencher o campo nome: arroz
    - Clicar no botão “Salvar”.
  - **Resultado Esperado:** O sistema exibe mensagem de erro: “Já existe um produto com este nome”.
- **CT05 – Cadastro com data de vencimento inválida**
  - **Descrição:** O sistema deve impedir o cadastro de produtos com data de vencimento anterior à data atual.

- **Pré-requisito:** Usuário estar na tela de cadastro de produtos
- **Passos:** Preencher os campos normalmente.
  - Informar uma data de vencimento passada.
  - Clicar no botão “Salvar”.
- **Resultado Esperado:** O sistema exibe mensagem de erro: “A data de vencimento deve ser futura”.

#### 2.5.5. Requisitos Não Funcionais

- **Desempenho:** A tela de listagem deve carregar em até 2 segundos.
- **Segurança:** A ação “Dar Baixa” só pode ser feita por usuários autorizados.
- **Escalabilidade:** Deve suportar centenas de produtos sem perda de desempenho.
- **Manutenibilidade:** O sistema deve ter código modular para facilitar atualizações.
- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** interface simples, com botões claros e instruções visíveis.

AlberguePro

Controle de Estoque - Pesquisar

Pesquisar por tipo: ☐ Alimento ☐ Higiene ☐ Limpeza

Nome:

Tipo	Nome	Quantidade	Unidade	Data de Vencimento
Alimento	feijão	56	5kg	15/03/26
Alimento	arroz	36	10kg	23/06/27
Alimento	açúcar	44	1kg	02/08/26

Buscar Editar

#### 2.6. User Story – Busca de Produtos

**Como,** responsável pelo estoque,

**Eu quero** buscar produtos por nome ou tipo,

**Porque** assim consigo encontrar rapidamente os itens que preciso.

#### 2.6.1. Critérios

- Deve ser possível buscar produtos por nome.
- Deve ser possível filtrar os produtos por tipo (Alimento, Higiene, Limpeza)
- A busca e o filtro devem exibir uma lista de produtos correspondentes com suas informações (Tipo, Nome, Quantidade, Unidade de medida e Data de Vencimento).

#### 2.6.2. Regras de negócio

- A busca por nome deve ser case-insensitive (não diferenciar maiúsculas de minúsculas).
- O filtro por tipo deve exibir apenas os produtos do tipo selecionado.

#### 2.6.3. Teste de Aceitação Positivo

- **CT01** - Busca de produto por nome
  - **Descrição:** Verificar se a busca por nome retorna os produtos corretos.
  - **Pré-requisito:** Na tela de busca, selecionar "Filtro por nome" e digitar o nome do produto no campo de busca.
  - **Passos/ação:** Preencher todos os campos corretamente.
    - Clicar no botão "Buscar".
  - **Resultado Esperado:** A tabela exibe apenas os produtos com o nome digitado e suas informações.
- **CT02** - Aplicação do filtro por tipo na listagem
  - **Descrição:** Verificar se o filtro funciona corretamente.
  - **Pré-requisito:** Ter pelo menos um produto cadastrado de cada tipo.
  - **Passos/ação:** Usuário estar na tela de buscar produtos
    - Selecionar "alimento" no filtro de tipo.
    - Clicar no botão "Buscar"
  - **Resultado Esperado:** O sistema vai exibir apenas os produtos do tipo "Alimento".

#### 2.6.4. Teste de Aceitação Negativo.

- **CT03** – Buscar produto com nome não existente
  - **Descrição:** Verificar se buscar o produto por nome não existente retorna nenhum resultado.
  - **Pré-requisito:** Ter produtos cadastrados no estoque.
  - **Passos:** Usuário estar na tela de buscar produtos
    - Selecionar "Filtro por nome" e digitar um nome de produto que não está cadastrado
    - Clicar no botão "Buscar"
  - **Resultado Esperado:** O sistema exibe a mensagem: "Nenhum produto encontrado com este nome".
- **CT04** – Aplicar o filtro por tipo sem produtos correspondentes
  - **Descrição:** Verificar se o filtro por tipo funciona corretamente quando não há produtos do tipo selecionado.
  - **Pré-requisito:** Ter produtos cadastrados, mas garantir que não haja nenhum produto de um tipo específico.
  - **Passos:** Usuário estar na tela de buscar produtos
    - Selecionar "Limpeza" no filtro de tipo.
    - Clicar no botão "Buscar"
  - **Resultado Esperado:** O sistema exibe mensagem: "Nenhum produto do tipo Limpeza encontrado".
- **CT05** – Busca com campo de nome vazio e filtro de tipo desmarcado
  - **Descrição:** Verificar se a busca funciona corretamente quando o campo de busca por nome está vazio e nenhum filtro por tipo é selecionado, exibindo todos os produtos.
  - **Pré-requisito:** Ter produtos cadastrados.
  - **Passos:** Usuário estar na tela de buscar produtos
    - Deixar o campo de busca por nome vazio e nenhum tipo selecionado.
  - **Resultado Esperado:** O sistema exibe todos os produtos cadastrados.

#### 2.6.5. Requisitos Não Funcionais

- **Desempenho:** A tela de listagem deve carregar em até 2 segundos.
- **Segurança:** A funcionalidade de consulta de produtos só deve estar disponível para usuários autenticados no sistema.
- **Escalabilidade:** Deve suportar a busca e filtragem de centenas de produtos sem perder desempenho.
- **Manutenibilidade:** O sistema deve ter código modular para facilitar atualizações.

- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** Interface de busca deve ser clara e os filtros intuitivos.

AlberguePro

Controle de Estoque - Dar Baixa

Pesquisar por tipo: ☐ Alimento ☐ Higiene ☐ Limpeza

Nome:

Tipo	Nome	Quantidade	Unidade	Data de Vencimento	
Alimento	feijão	56	5kg	15/03/26	
Alimento	arroz	36	10kg	23/06/27	
Alimento	açúcar	44	1kg	02/08/26	

Salvar Voltar

## 2.7. User Story – Dar Baixa em Produtos

**Como**, responsável pelo estoque,

**Eu quero** dar baixa em produtos,

**Porque** assim consigo registrar a saída de itens e manter o estoque atualizado.

### 2.7.1. Critérios

- Deve ser possível dar baixa em produtos com quantidade positiva.
- Deve ser possível buscar produtos por nome ou filtrar por tipo (Alimento, Higiene ou Limpeza).
- A ação de "Dar Baixa" deve ser ativada ao clicar no botão "Editar" ao lado do item.
- Ao clicar no botão "Salvar", a quantidade do produto deve ser atualizada.

### 2.7.2. Regras de negócio

- Não deve ser permitido dar baixa maior do que a quantidade disponível em estoque.

### 2.7.3. Teste de Aceitação Positivo

- **CT01** – Dar baixa em produto com quantidade menor que a disponível
  - **Descrição:** Verificar se é possível dar baixa em uma quantidade válida de um produto.
  - **Pré-requisito:** Ter um produto cadastrado com quantidade positiva
  - **Passos:** Usuário estar na tela de dar baixar
    - Clicar no botão “Editar”
    - Inserir uma quantidade menor do que a quantidade do estoque.
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** A quantidade do produto deve ser atualizada de acordo com a quantidade que foi dada baixa.
- **CT02** – Filtrar por tipo de produto
  - **Descrição:** Verificar se o sistema exibe corretamente os produtos ao selecionar um tipo.
  - **Pré-requisito:** Ter produtos cadastrados nos três tipos (Alimento, Higiene, Limpeza).
  - **Passos:** Usuário estar na tela de dar baixar
    - Clicar no botão “Editar”
    - Seleciona o tipo "Higiene".
  - **Resultado Esperado:** Apenas os produtos do tipo "Higiene" devem ser exibidos na tabela.

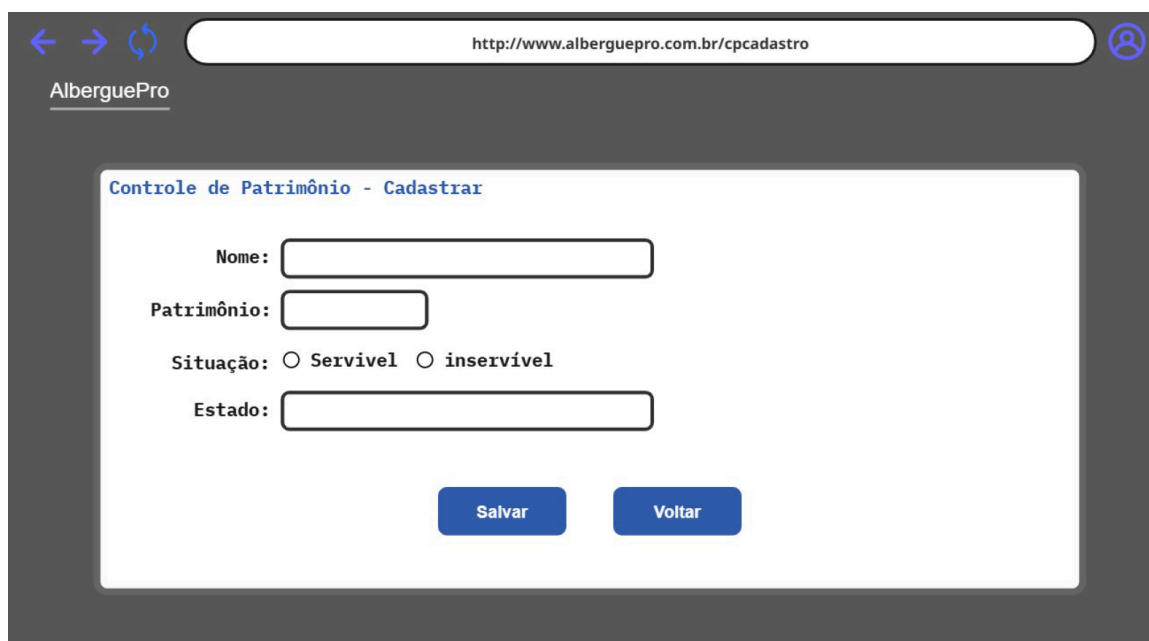
### 2.7.4. Teste de Aceitação Negativo

- **CT03** – Dar baixa em produto com quantidade negativa
  - **Descrição:** Verificar se o sistema impede a baixa de uma quantidade negativa..
  - **Pré-requisito:** Ter um produto cadastrado com quantidade positiva
  - **Passos:** Usuário estar na tela de dar baixar.
    - Inserir "-10" na quantidade a ser baixada.
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** O sistema deve exibir uma mensagem de erro " quantidade para baixa deve ser um número positivo".
- **CT04** – Dar baixa em produto com quantidade maior que a disponível

- **Descrição:** Verificar se o sistema impede a baixa de uma quantidade maior do que a disponível em estoque.
- **Pré-requisito:** Ter um produto cadastrado com quantidade positiva
- **Passos:** Usuário estar na tela de dar baixa.
  - Inserir uma quantidade maior do que a quantidade do estoque.
  - Clicar no botão "Salvar".
- **Resultado Esperado:** O sistema deve exibir uma mensagem de erro "Não é permitido dar baixa maior do que a quantidade disponível em estoque"

#### 2.7.5. Requisitos Não Funcionais

- **Desempenho:** A atualização do estoque após a baixa deve ser processada em menos de 1 segundo.
- **Segurança:** A ação de "Dar Baixa" só pode ser executada por usuários com permissão de gerência de estoque.
- **Escalabilidade:** O sistema de baixa de estoque deve ser capaz de lidar com um grande número de produtos sem perda de desempenho.
- **Manutenibilidade:** O sistema deve ter código modular para facilitar atualizações.
- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** A interface para dar baixa deve ser clara, com campos de entrada bem definidos e mensagens de erro compreensíveis.



The screenshot shows a web browser window with the URL <http://www.alberguepro.com.br/cpcadastro>. The page title is "AlberguePro". The main content area is titled "Controle de Patrimônio - Cadastrar" and contains the following form elements:

- Nome:** A text input field.
- Patrimônio:** A text input field.
- Situação:** Two radio buttons labeled "Servível" and "inservível".
- Estado:** A text input field.
- Buttons:** Two blue buttons at the bottom, labeled "Salvar" and "Voltar".

## 2.8. User Story – Cadastro de patrimônio

**Como** usuário do sistema,

**Eu quero** cadastrar itens patrimoniais com nome, estado e situação,

**Porque** preciso manter o controle do patrimônio da instituição.

### 2.8.1. Critérios

- O sistema deve permitir o cadastro de patrimônio com os campos obrigatórios: Nome, Estado e Situação (Servível/Inservível).
- O botão “Salvar” deve validar se todos os campos obrigatórios estão preenchidos.
- O botão “Voltar” deve redirecionar para a tela anterior sem salvar as alterações.

### 2.8.2. Regras de negócio:

- O campo Nome deve aceitar no máximo 100 caracteres.
- O campo Estado deve aceitar apenas letras e espaços, com até 50 caracteres.
- A Situação do patrimônio deve ser obrigatoriamente uma das opções: Servível ou Inservível.
- Não pode haver campos obrigatórios em branco.
- Apenas usuários autenticados podem acessar essa funcionalidade.

### 2.8.3. Teste de Aceitação Positivo

- **CT01** – Cadastro de item patrimonial servível.
  - **Descrição:** Verifica se o sistema permite cadastrar um item com dados válidos e o estado servível selecionado.
  - **Pré-requisito:**
  - **Passos:** Usuário estar na tela de cadastro de patrimônio.
    - Preencher todos os campos corretamente.
    - Clicar no botão “Salvar”
  - **Resultado Esperado:** O item é salvo e listado corretamente.
- **CT02** – Cadastro de item patrimonial com estado Inservível
  - **Descrição:** Verifica se o sistema permite cadastrar um item com dados válidos e o estado inservível selecionado.
  - **Pré-requisito:**
  - **Passos:** Usuário estar na tela de cadastro de patrimônio.
    - Preencher todos os campos corretamente.
    - Clicar no botão “Salvar”



- **Resultado Esperado:** O item é salvo e listado corretamente

#### 2.8.4. Teste de Aceitação Negativo

- **CT03** – Cadastro com campos vazios
  - **Descrição:** Verificar se o sistema impede o cadastro de um item com campos obrigatórios vazios.
  - **Pré-requisito:** Acesso à tela de cadastro patrimônio
  - **Passos:** Acessar a tela de cadastro de patrimônio
    - Deixar um campo obrigatório vazio.
    - Clicar no botão “Salvar”.
  - **Resultado Esperado:** O sistema exibe uma msg de erro: “Preencha todos os campos obrigatórios”.
- **CT04** – Cadastro com nome duplicado
  - **Descrição:** Verificar se o sistema impede o cadastro de um item com um nome que já existe.
  - **Pré-requisito:** Ter um item já cadastrado com o nome "Mesa".
  - **Passos:** Na tela de cadastro.
    - Preencher o campo nome: Mesa
    - Clicar no botão “Salvar”.
  - **Resultado Esperado:** O sistema exibe mensagem de erro: “Já existe um item com este nome”.
- **CT05** – Estado inválido
  - **Descrição:** Verifica se o sistema impede salvar sem selecionar o estado.
  - **Pré-requisito:** Tela de cadastro.
  - **Passos:** Deixar sem selecionar o estado
    - Clicar no botão “Salvar”.
  - **Resultado Esperado:** O sistema exibe erro: “O Estado deve ser selecionado”.

#### 2.8.5. Requisitos Não Funcionais

- **Desempenho:** A tela deve carregar em até 1 segundos.
- **Segurança:** Apenas usuários autenticados podem alterar o estado.
- **Escalabilidade:** Gerência de muitos itens sem perda de desempenho.
- **Manutenibilidade:** Módulo de patrimônio modular para fácil depuração e atualizações.
- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.

- **Usabilidade:** Campos claros, uso de dropdown para evitar erros.

AlberguePro

http://www.alberguepro.com.br/cpesquisar

Controle de Patrimônio - Pesquisar

Nome:

Nome	Patrimonio	Situação	Estado
ventilador	8343	<input type="radio"/> Servível <input type="radio"/> inservível	bom estado
cama	5689	<input type="radio"/> Servível <input type="radio"/> inservível	ótimo estado
bebedouro	9234	<input type="radio"/> Servível <input type="radio"/> inservível	bom estado

## 2.9. User Story – Busca de Patrimônio

**Como** administrador do sistema,

**Eu quero** buscar itens patrimoniais por nome,

**Porque** consigo localizar rapidamente os bens para consulta ou edição.

### 2.9.1. Critérios

- Deve ser possível buscar itens patrimoniais pelo "Nome".
- A busca deve exibir uma lista de itens correspondentes com suas informações: "Nome", "Patrimônio", "Situação" e "Estado".

### 2.9.2. Regras de negócio:

- A busca por nome deve ser case-insensitive (não diferenciar maiúsculas de minúsculas).
- A busca deve retornar todos os itens que contenham o texto digitado no campo "Nome".

### 2.9.3. Teste de Aceitação Positivo

- **CT01** – Busca de item por nome existente
  - **Descrição:** Verificar se a busca por nome retorna os itens corretos.
  - **Pré-requisito:** Ter itens patrimoniais cadastrados no sistema.
  - **Passos:** Usuário estar na tela Buscar Patrimônio
    - Digitar o nome do item já cadastrado no campo de busca.
    - Clicar no botão "Buscar".
  - **Resultado Esperado:** A tabela exibe o(s) item(ns) com o nome correspondente.
  
- **CT02** – Busca sem preencher o campo de pesquisa
  - **Descrição:** Verificar se o sistema exibe todos os itens patrimoniais cadastrados quando o campo de busca está vazio e o botão "Buscar" é clicado.
  - **Pré-requisito:** Ter itens patrimoniais cadastrados no sistema.
  - **Passos:** Usuário estar na tela Buscar Patrimônio
    - Deixar o campo de busca vazio.
    - Clicar no botão "Buscar".
  - **Resultado Esperado:** A tabela exibe todos os itens patrimoniais com suas respectivas informações.

#### 2.9.4. Teste de Aceitação Negativo

- **CT03** – Busca de item patrimonial por nome não existente
  - **Descrição:** Verificar a busca por nome de item patrimonial que não existe.
  - **Pré-requisito:** Ter itens patrimoniais cadastrados no sistema.
  - **Passos:** Usuário estar na tela Buscar Patrimônio
    - Digitar um nome de item inexistente.
    - Clicar no botão "Buscar".
  - **Resultado Esperado:** O sistema exibe uma mensagem: "Nenhum item patrimonial encontrado com este nome".

#### 2.9.5. Requisitos Não Funcionais

- **Desempenho:** A busca e o carregamento dos resultados devem ser concluídos em menos de 1 segundo.
- **Segurança:** O acesso à tela de busca e edição deve ser restrito a usuários autenticados..
- **Escalabilidade:** O sistema deve ser capaz de gerenciar e exibir uma lista com muitos itens patrimoniais sem degradação de desempenho.
- **Manutenibilidade:** Módulo de patrimônio modular para fácil depuração e atualizações.

- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** A interface de busca deve ser clara e intuitiva, com o campo de pesquisa e o botão "Buscar" facilmente identificáveis.

AlberguePro

Controle de Patrimônio - Editar

Nome:

Nome	Patrimonio	Situação	Estado
ventilador	8343	<input type="radio"/> Servível <input type="radio"/> inservível	bom estado
cama	5689	<input type="radio"/> Servível <input type="radio"/> inservível	otimo estado
bebedouro	9234	<input type="radio"/> Servível <input type="radio"/> inservível	bom estado

## 2.10. User Story – Edição de Patrimônio

**Como** administrador do sistema,

**Eu quero** editar as informações de itens patrimoniais,

**Porque** consigo atualizar dados como nome, quantidade, detalhes e o estado.

### 2.10.1. Critérios

- Deve ser possível buscar itens patrimoniais pelo nome para edição.
- Deve ser possível atualizar a "Situação" (Servível/Inservível).
- Deve ser possível atualizar o "Estado" (Ex: bom estado, ótimo estado).
- O botão "editar" ao lado de cada item deve permitir modificação inline (na mesma linha).
- O botão "Salvar" deve manter as alterações realizadas na tabela.

### 2.10.2. Regras de negócio:

- A busca por nome deve ser case-insensitive(não diferenciar

maiúsculas de minúsculas).

- A quantidade editada deve ser um número inteiro e positivo.

#### 2.10.3. Teste de Aceitação Positivo

- **CT01** – Alterar situação de um item para "Inservível"
  - **Descrição:** Verificar se é possível alterar o estado de um item patrimonial de "Servível" para "Inservível".
  - **Pré-requisito:** Ter um item patrimonial cadastrado como "Servível".
  - **Passos:** Usuário estar na tela de Editar Patrimônio.
    - Localizar o item servível.
    - Clicar no botão "Editar".
    - Marcar "Inservível".
    - .Clicar no botão "Salvar".
  - **Resultado Esperado:** O estado do item é alterado para "Inservível" e o item é listado na tabela.
- **CT02** – Alterar estado de um item para "ótimo estado"
  - **Descrição:** Verificar se é possível alterar o estado de um item patrimonial.
  - **Pré-requisito:** Ter um item patrimonial cadastrado.
  - **Passos:** Usuário estar na tela de Editar Patrimônio.
    - Localizar um item patrimonial qualquer.
    - Clicar no botão "Editar".
    - Alterar o campo "Estado" para "ótimo estado"
    - .Clicar no botão "Salvar".
  - **Resultado Esperado:** O estado do item é atualizado corretamente.

#### 2.10.4. Teste de Aceitação Negativo

- **CT03** – Preencher campo Estado com valor inválido (ex: número)
  - **Descrição:** Verificar se o sistema impede salvar um estado inválido.
  - **Pré-requisito:** Ter um item patrimonial cadastrado.
  - **Passos:** Usuário estar na tela de Editar Patrimônio.
    - Clicar no botão "editar".
    - Digitar "12345" no campo "Estado".
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** Sistema exibe uma mensagem de erro: "Estado inválido".

- **CT04** – Salvar sem selecionar situação
  - **Descrição:** Verificar se o sistema impede salvar sem definir a situação.
  - **Pré-requisito:** Ter um item patrimonial cadastrado.
  - **Passos:** Usuário estar na tela de Editar Patrimônio.
    - Clicar no botão "editar".
    - Não selecionar nenhuma opção de situação.
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** O sistema exibe uma mensagem de erro: "Selecione uma situação (Servível ou Inservível)".

#### 2.10.5. Requisitos Não Funcionais

- **Desempenho:** A tela deve carregar em até 1 segundo.
- **Segurança:** Apenas usuários autenticados com perfil de administrador podem editar os itens patrimoniais.
- **Escalabilidade:** Deve suportar edição de milhares de itens.
- **Manutenibilidade:** Código modular para facilitar manutenção e expansão.
- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** Interface clara, com campos visíveis e botões bem localizados.

The screenshot shows a web browser window with the URL `http://www.alberguepro.com.br/cacadastrar`. The page title is "AlberguePro" and the form is titled "Cadastro de Acolhidos - Cadastrar". The form fields are as follows:

- Nome: \_\_\_\_\_
- Sexo: ☐ Masc ☐ Fem
- Data de Nascimento: \_\_\_\_\_
- Naturalidade: \_\_\_\_\_
- Idade: \_\_\_\_\_
- Profissão: \_\_\_\_\_
- Cor: \_\_\_\_\_
- RG: \_\_\_\_\_
- CPF: \_\_\_\_\_
- Certidão de Nascimento: \_\_\_\_\_
- Filiação: \_\_\_\_\_
- Estado Civil: \_\_\_\_\_
- ☐ Solteiro/a ☐ Casado/a ☐ Divorciado/a ☐ União Estável ☐ Viúvo/a
- Nome do Cônjuge: \_\_\_\_\_
- Filhos: ☐ Sim ☐ Não Quantos: \_\_\_\_\_
- Endereço: \_\_\_\_\_
- Telefone para Contato Familiar: \_\_\_\_\_
- Data do Ingresso: \_\_\_\_\_
- Horário: \_\_\_\_\_
- Data de Saída: \_\_\_\_\_
- Horário: \_\_\_\_\_
- Pode Retornar: Resposta automática do sistema

At the bottom of the form, there are two buttons: "Salvar" and "Voltar".

#### 2.11. User Story – Cadastro de Acolhidos

**Como** assistente social,  
**Eu quero** cadastrar os acolhidos com todos os dados relevantes,  
**Porque** preciso acompanhar sua situação, histórico e permanência na instituição.

#### 2.11.1. Critérios

- Deve ser possível preencher e registrar todas as informações pessoais do acolhido, incluindo dados de identificação.
- Deve ser possível registrar dados familiares, como Estado Civil, Nome do Cônjuge e informações sobre filhos, incluindo o número de filhos.
- Deve ser possível informar dados de contato e localização, incluindo o endereço do acolhido e um telefone para contato familiar.
- Deve ser possível registrar as datas e horários de entrada e saída do acolhido.
- O sistema deve preencher automaticamente o campo "Pode retornar:".
- O botão "Salvar" deve efetivar o registro de todas as informações do acolhido no sistema.

#### 2.11.2. Regras de negócio:

- Campos como "Nome", "Data de Nascimento", "Sexo", "Estado Civil", "Endereço" e "Data do Ingresso" são obrigatórios.
- A "Idade" deve ser calculada automaticamente pela data de nascimento.
- A quantidade de filhos é obrigatória e deve ser um número positivo se o campo "Filhos: Sim" for selecionado.
- A "Data de Saída", se informada, deve ser posterior ou igual à "Data do Ingresso".
- A resposta em "Pode retornar:" é gerada automaticamente pelo sistema e não pode ser editada pelo usuário.
- "CPF" e "RG" devem ser únicos no cadastro.

#### 2.11.3. Teste de Aceitação Positivo

- **CT01** – Cadastro completo e válido de um acolhido.
  - **Descrição:** Verificar se o sistema permite cadastrar um acolhido com todas as informações preenchidas corretamente.
  - **Pré-requisito:** Todos os campos preenchidos corretamente.
  - **Passos:** Usuário estar na tela de cadastro de acolhidos.

- Preencher todos os campos obrigatórios.
- Preencher alguns campos opcionais.
- Clicar no botão "Salvar".
- **Resultado Esperado:** : O acolhido é cadastrado com sucesso. Uma mensagem de confirmação é exibida. O campo "Pode retornar:" deve ter uma resposta automática preenchida.

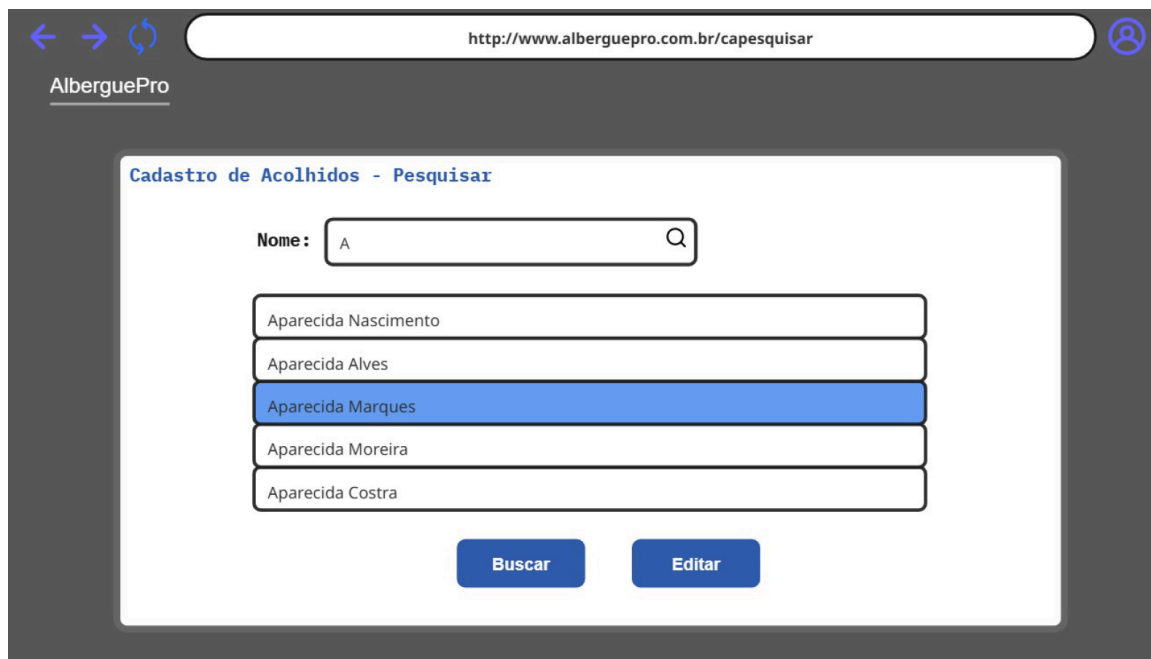
#### 2.11.4. Teste de Aceitação Negativo

- **CT02** – Cadastro com campos obrigatórios vazios.
  - **Descrição:** Verificar se o sistema impede o cadastro de um acolhido com campos obrigatórios vazios.
  - **Pré-requisito:** Acesso à tela de cadastro de acolhidos.
  - **Passos:** Usuário estar na tela de cadastro de acolhidos.
    - Deixar o campo um campo obrigatório vazio.
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** O sistema exibe uma mensagem de erro: "Preencha todos os campos obrigatórios."
- **CT03** – Cadastro com "Data de Saída" anterior à "Data d o Ingresso".
  - **Descrição:** Verificar se o sistema impede o cadastro quando a Data de Saída é anterior à Data de Ingresso.
  - **Pré-requisito:** Usuário na tela de cadastro de acolhidos.
  - **Passos:** Preencher "Data do Ingresso: 10/06/2025"
    - Preencher a ""Data de Saída: 09/06/2025".
  - **Resultado Esperado:** O sistema exibe uma mensagem de erro: "A Data de Saída deve ser posterior a Data do Ingresso."

#### 2.11.5. Requisitos Não Funcionais

- **Desempenho:** A tela carrega em 1s.
- **Segurança:** Acesso a dados de acolhidos restritos a usuários autorizados e autenticados.
- **Escalabilidade:** Capaz de gerenciar muitos registros de acolhidos sem degradação.
- **Manutenibilidade:** Módulo de cadastro modular, fácil de adaptar a novas leis ou campos.
- **Compatibilidade:** Funciona em navegadores modernos como Chrome, Edge e Firefox.
- **Usabilidade:** Formulário intuitivo, campos claros e mensagens de erro e sucesso amigáveis.





## 2.12. User Story – Busca de Acolhidos

**Como** assistente social,  
**Eu quero** buscar acolhidos por nome,  
**Porque assim** consigo localizar rapidamente seus registros para consulta ou edição.

### 2.12.1. Critérios

- Deve ser possível buscar acolhidos pelo nome em um campo de pesquisa.
- A busca deve exibir uma lista de acolhidos correspondentes.
- A listagem de resultados deve exibir o nome do acolhido.

### 2.12.2. Regras de negócio:

- A busca por nome deve ser case-insensitive (não diferenciar maiúsculas de minúsculas).
- A busca deve retornar todos os acolhidos cujo nome contenha o texto digitado.
- Se o campo de busca estiver vazio e o botão "Buscar" for clicado, todos os acolhidos cadastrados devem ser listados.

### 2.12.3. Teste de Aceitação Positivo

- **CT01** – Busca de acolhido por nome existente.
  - **Descrição:** Verificar se a busca por nome retorna os acolhidos corretos.
  - **Pré-requisito:** Ter acolhidos cadastrados no sistema.
  - **Passos:** Usuário estar na tela de Buscar Acolhidos.
    - Digitar um nome no campo de busca.
    - Clicar no botão "Buscar".
  - **Resultado Esperado:** A lista exibe apenas os nomes correspondentes ao da pesquisa.
  
- **CT02** – Busca sem preencher o campo de pesquisa.
  - **Descrição:** Verificar se o sistema exibe todos os acolhidos cadastrados quando o campo de busca está vazio.
  - **Pré-requisito:** Ter vários acolhidos cadastrados.
  - **Passos:** Usuário estar na tela de Buscar Acolhidos.
    - Deixar o campo de busca vazio.
    - Clicar no botão "Buscar".
  - **Resultado Esperado:** O sistema exibe todos os acolhidos cadastrados no sistema.
  
- **CT03** – Seleção e clique no botão "Editar".
  - **Descrição:** Verificar se o botão "Editar" funciona corretamente ao selecionar um acolhido.
  - **Pré-requisito:** Ter acolhidos cadastrados.
  - **Passos:** Usuário estar na tela de Buscar Acolhidos.
    - Selecionar um acolhido na lista, dando um clique no nome do acolhido.
    - Clicar no botão "Editar".
  - **Resultado Esperado:** O sistema direciona para a tela de edição do acolhido selecionado.

### 2.12.4. Teste de Aceitação Negativo

- **CT04** – Busca de acolhido por nome não existente.
  - **Descrição:** Verificar se a busca por um nome de acolhido que não existe retorna nenhum resultado.
  - **Pré-requisito:** Ter acolhidos cadastrados no sistema.
  - **Passos:** Usuário estar na tela de Buscar Acolhidos.
    - Digitar um nome inexistente no campo de busca.
    - Clicar no botão "Buscar".
  - **Resultado Esperado:** O sistema exibe uma mensagem de erro: “Nenhum acolhido encontrado com esse nome.”

- **CT05** – Tentar editar sem selecionar um acolhido.
  - **Descrição:** Verificar se o sistema impede a edição se nenhum acolhido for selecionado na lista.
  - **Pré-requisito:** Ter acolhidos cadastrados.
  - **Passos:** Usuário estar na tela de Buscar Acolhidos.
    - Não selecionar nenhum nome na lista.
    - Clicar no botão "Editar".
  - **Resultado Esperado:** O sistema exibe uma mensagem de erro: "Nenhum acolhido selecionado para edição."

#### 2.12.5. Requisitos Não Funcionais

- **Desempenho:** A tela carrega em até 2 segundos.
- **Segurança:** Acesso à tela de busca e edição restrito a usuários autenticados e autorizados.
- **Escalabilidade:** Capaz de gerenciar e buscar entre muitos de registros de acolhidos sem perda de desempenho.
- **Manutenibilidade:** Módulo de busca modular para fácil manutenção e futuras expansões.
- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** Interface clara, busca intuitiva, e fácil seleção/edição de acolhidos.

← → ↺ http://www.alberguepro.com.br/caeditar

AlberguePro

Cadastro de Acolhidos - Editar

Nome: Aparecida Marques Sexo: ☐ Masc ☐ Fem

Data de Nascimento: \_\_\_\_\_ Naturalidade: \_\_\_\_\_

Idade: \_\_\_\_\_ Profissão: \_\_\_\_\_ Cor: \_\_\_\_\_ RG: \_\_\_\_\_ CPF: \_\_\_\_\_

Certidão de Nascimento: \_\_\_\_\_ Filiação: \_\_\_\_\_

Estado Civil: \_\_\_\_\_ ☐ Solteiro/a ☐ Casado/a ☐ Divorciado/a ☐ União Estável ☐ Viúvo/a

Nome do Cônjuge: \_\_\_\_\_ Filhos: ☐ Sim ☐ Não Quantos: \_\_\_\_\_

Endereço: \_\_\_\_\_ Telefone para Contato Familiar: \_\_\_\_\_

Data do Ingresso: \_\_\_\_\_ Horário: \_\_\_\_\_ Data de Saída: \_\_\_\_\_ Horário: \_\_\_\_\_

Pode Retornar: Resposta automática do sistema

Salvar Voltar

## 2.13. User Story – Edição de Cadastro de Acolhidos

**Como** assistente social,  
**Eu quero**, editar as informações de um acolhido existente,  
**Porque assim** consigo manter os dados atualizados e corrigir eventuais erros no registro.

### 2.13.1. Critérios

- Deve ser possível visualizar todos os campos do cadastro do acolhido previamente preenchidos.
- O campo "Pode retornar:" deve exibir a resposta automática do sistema e não deve ser editável.
- O botão "Salvar" deve persistir as alterações realizadas no registro do acolhido.

### 2.13.2. Regras de negócio:

- Validação de campos obrigatórios se aplica também na edição.
- "Quantos" filhos é obrigatório e positivo se "Filhos: Sim" for selecionado.
- A Data de Saída deve ser igual ou posterior à Data do Ingresso.
- CPF e RG devem permanecer únicos no sistema após edição.
- Nome do Cônjuge se torna obrigatório para Estado Civil "Casado/a" ou "União Estável".
- Campo "Pode retornar:" não é editável, é gerado automaticamente pelo sistema.

### 2.13.3. Teste de Aceitação Positivo

- **CT01** – Edição bem-sucedida de informações gerais do acolhido.
  - **Descrição:** Verificar se é possível editar e salvar corretamente as informações de um acolhido.
  - **Pré-requisito:** Ter um acolhido cadastrado e ter acesso a tela de edição.
  - **Passos:** Usuário estar na tela de Editar Acolhidos.
    - Alterar algum campo editável.
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** As informações do acolhido são editadas no sistema.

#### 2.13.4. Teste de Aceitação Negativo

- **CT02** – Edição com campo obrigatório vazio.
  - **Descrição:** Verificar se o sistema impede o salvamento de edições se um campo obrigatório for deixado em branco.
  - **Pré-requisito:** Ter um e ter acesso a tela de edição.
  - **Passos:** Usuário estar na tela de Editar Acolhidos.
    - Apagar um conteúdo de campo obrigatório.
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** O sistema exibe mensagem de erro: "Preencha todos os campos obrigatórios."
- **CT03** – Edição com "Data de Saída" anterior à "Data do Ingresso"
  - **Descrição:** Verificar se o sistema impede a alteração se a Data de Saída for anterior à Data do Ingresso.
  - **Pré-requisito:** Ter um acolhido acessado na tela de edição.
  - **Passos:** Usuário estar na tela de Editar Acolhidos.
    - Alterar "Data do Ingresso" para "10/01/2025" e "Data de Saída" para "10/12/2024".
    - Clicar no botão "Salvar".
  - **Resultado Esperado:** O sistema exibe uma mensagem de erro: "A Data de Saída não pode ser anterior à Data do Ingresso."

#### 2.13.5. Requisitos Não Funcionais

- **Desempenho:** A tela carrega em até 2 segundos.
- **Segurança:** Acesso à tela de edição restrito a usuários autenticados e autorizados.
- **Escalabilidade:** Capaz de gerenciar a edição de grande volume de registros sem degradação.
- **Manutenibilidade:** Módulo de busca modular para fácil manutenção e futuras expansões.
- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** Campos pré-preenchidos, validações claras e feedback visual ao usuário.

← → ↺ http://www.alberguepro.com.br/gvlicadastrar

AlberguePro

Gestão de Vagas e Leitos - Cadastrar

Nome:  Q

Quarto:  ▼

Leito:  ▼

Data de Entrada:

Data de Saída:

Salvar Voltar

## 2.14. User Story – Alocação de Acolhido em Leito

**Como**, coordenador do abrigo,

**Eu quero** adicionar um novo acolhido a um leito,

**Por quê** consigo registrar a ocupação de uma vaga.

### 2.14.1. Critérios

- Deve ser possível selecionar o acolhido por nome, buscando em uma lista de acolhidos já cadastrados.
- Deve ser possível selecionar o quarto e o leito disponíveis para o acolhido..
- Deve ser possível informar a data de entrada do acolhido no leito.
- A data de saída deve ser um campo preenchível ou gerado automaticamente e o usuário pode editar, se necessário.
- Ao clicar em "Salvar", a alocação do acolhido no quarto e leito selecionados deve ser persistida e o status do leito deve ser atualizado.

### 2.14.2. Regras de negócio:

- Não é permitido alocar um acolhido em um leito cheio.
- Não deve ser possível salvar a alocação sem preencher os campos "Nome", "Quarto", "Leito" e "Data de Entrada".
- A data de saída deve ser igual ou posterior à data de entrada.

- O campo "Nome" deve ser um campo de busca que se baseia nos nomes dos acolhidos já cadastrados.

#### 2.14.3. Teste de Aceitação Positivo

- **CT01** – Adição de um novo acolhido a um leito vazio ou parcial
  - **Descrição:** Verificar se é possível adicionar um novo acolhido a um leito livre, e se a data de saída é preenchida automaticamente.
  - **Pré-requisito:** Ter um quarto com status vazio ou parcial.
  - **Passos:** Na tela de busca gestão de vagas e leito.
  - Clicar no botão "Adicionar".
  - Preencher os dados corretamente
  - CLicar no botão “Salvar”
  - **Resultado Esperado:** Acolhido é adicionado à lista de acolhidos do Quarto, o sistema deve mudar o status do leito se necessário e deve exibir a "Data de Saída" do acolhido automaticamente

#### 2.14.4. Teste de Aceitação Negativo

- **CT02** – Tentativa de adicionar acolhido com campos obrigatórios vazios.
  - **Descrição:** Verificar se o sistema impede a adição de um acolhido com um ou mais campos obrigatórios vazios.
  - **Pré-requisito:** Leito deve ter vagas.
  - **Passos:** Usuário estar na tela de adicionar gestão de leitos.
    - Deixar os campos vazios.
    - Clicar no botão “Salvar”
  - **Resultado Esperado:** O sistema exibe mensagem de erro: “Preencha todos os campos obrigatórios”.
- **CT03** – Tentativa de alocar em um leito ocupado.
  - **Descrição:** Verificar se o sistema impede a alocação de um acolhido em um leito que já está ocupado.
  - **Pré-requisito:** Ter um leito ocupado.
  - **Passos:** Usuário estar na tela de alocação de leitos.
    - Selecionar um "Quarto" e um "Leito" que já estão ocupados.
    - Clicar no botão “Salvar”
  - **Resultado Esperado:** O sistema exibe mensagem de erro: “O leito selecionado já está ocupado”.

AlberguePro

Gestão de Vagas e Leitos - Pesquisar

Nome:

**Acolhidos**

Código	Nome	Quarto	Leito	Data Entrada	Data Saida
1	Aparecida Nascimento	02	01	06/01/25	06/06/25
2	Aparecida Alves	05	04	08/02/25	n/a

## 2.15. User Story – Pesquisa e Gestão de Leitos

**Como**, coordenador do abrigo,

**Eu quero** pesquisar e visualizar a alocação de acolhidos em leitos,

**Por quê** preciso consultar a ocupação atual e ter a possibilidade de editar os registros.

### 2.15.1. Critérios

- Deve ser possível buscar acolhidos por nome através de um campo de pesquisa.
- Deve ser possível selecionar um registro de acolhido na tabela para realizar uma ação de edição..
- Deve haver botões "Buscar" e "Editar" para executar as ações na tela.
- A data de saída deve ser um campo preenchível ou gerado automaticamente e o usuário pode editar, se necessário.
- Ao clicar em "Salvar", a alocação do acolhido no quarto e leito selecionados deve ser persistida e o status do leito deve ser atualizado.

### 2.15.2. Regras de negócio:

- busca por nome deve ser case-insensitive (não diferenciar maiúsculas de minúsculas).



- O botão "Editar" só deve ser funcional se um acolhido estiver selecionado na tabela.

#### 2.15.3. Teste de Aceitação Positivo

- **CT01** – Busca por nome existente.
  - **Descrição:** Verificar se a busca por nome retorna os acolhidos corretos, filtrando a tabela.
  - **Pré-requisito:** Ter acolhidos alocados cadastrados no sistema.
  - **Passos:** Usuário estar na tela de Pesquisa de Vagas e Leitos.
  - Digitar um nome existente no campo de busca (ex: "Aparecida").
  - Preencher os dados corretamente
  - Clicar no botão "Buscar".
  - **Resultado Esperado:** A tabela exibe apenas os registros de acolhidos cujos nomes correspondem ao da pesquisa.
- **CT02** – Seleção e clique no botão "Editar".
  - **Descrição:** Verificar se a seleção de um acolhido e o clique no botão "Editar" direcionam para a tela de edição correta.
  - **Pré-requisito:** Ter acolhidos alocados cadastrados.
  - **Passos:** Usuário estar na tela de Pesquisa de Vagas e Leitos.
  - Selecionar um acolhido na tabela (ex: "Aparecida Marques").
  - Clicar no botão "Editar".
  - **Resultado Esperado:** O sistema direciona para a tela de edição da alocação do acolhido selecionado.

#### 2.15.4. Teste de Aceitação Negativo

- **CT03** – Busca de acolhido por nome não existente..
  - **Descrição:** Verificar se a busca por um nome que não existe não retorna nenhum resultado.
  - **Pré-requisito:** Ter acolhidos alocados cadastrados no sistema.
  - **Passos:** Usuário estar na tela de Pesquisa de Vagas e Leitos.
    - Digitar um nome inexistente no campo de busca (ex: "José").
    - Clicar no botão "Buscar".
  - **Resultado Esperado:** O sistema exibe mensagem de erro: "Nenhum acolhido encontrado com esse nome".
- **CT04** – Tentar editar sem selecionar um acolhido.
  - **Descrição:** Verificar se o sistema impede a edição se nenhum acolhido for selecionado na tabela.
  - **Pré-requisito:** Ter acolhidos alocados cadastrados..
  - **Passos:** Usuário estar na tela de Pesquisa de Vagas e Leitos.

- Não selecionar nenhum acolhido na tabela.
- Clicar no botão "Editar".
- **Resultado Esperado:** O sistema exibe uma mensagem de erro: "Nenhum acolhido selecionado para edição."

#### 2.15.5. Requisitos Não Funcionais

- **Desempenho:** A busca e a exibição da tabela devem ser rápidas, levando no máximo 2 segundos para carregar e exibir os resultados.
- **Segurança:** O acesso a esta tela é restrito a usuários autenticados e com a devida autorização.
- **Escalabilidade:** O sistema deve ser capaz de gerenciar e buscar entre um grande volume de registros de alocação de leitos sem degradação do desempenho.
- **Manutenibilidade:** O módulo de pesquisa e gestão deve ser modular, permitindo fácil manutenção e futuras expansões, como a adição de mais colunas.
- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** A interface deve ser clara e intuitiva, com a busca e a tabela de resultados facilmente compreensíveis pelo usuário.

AlberguePro

Gestão de Vagas e Leitos - Editar

Nome:

**Acolhidos**

Codigo	Nome	Quarto	Leito	Data Entrada	Data Saida
1	Aparecida Nascimento	02	01	06/01/25	06/06/25
2	Aparecida Alves	05	04	08/02/25	n/a

Salvar Voltar

#### 2.16. User Story – Edição e Desalocação de Leitos

**Como**, coordenador do abrigo,

**Eu quero** editar e desalocar acolhidos dos leitos, de forma massiva ou individual,

**Por quê** preciso manter o controle da ocupação e ajustar as alocações em tempo real.

#### 2.16.1. Critérios

- Deve ser possível visualizar a lista de acolhidos alocados em uma tabela que exibe o Código, Nome, Quarto, Leito, Data de Entrada e Data de Saída.
- Deve haver um campo de busca por nome para filtrar a lista de acolhidos exibida na tabela.
- O botão "Salvar" deve persistir em todas as alterações feitas na tela (edições e deslocações)..
- O botão "Voltar" deve permitir ao usuário retornar à tela anterior sem salvar as alterações.

#### 2.16.2. Regras de negócio:

- busca por nome deve ser case-insensitive (não diferenciar maiúsculas de minúsculas).
- Ao clicar no ícone de lápis, o usuário deve ser capaz de editar as informações de alocação (quarto, leito, datas).
- A "Data de Saída", se alterada, deve ser igual ou posterior à "Data de Entrada".

#### 2.16.3. Teste de Aceitação Positivo

- **CT01** – Edição bem-sucedida de alocação.
  - **Descrição:** Verificar se é possível editar a alocação de um acolhido (por exemplo, a data de saída) e salvar a alteração.
  - **Pré-requisito:** Ter um acolhido alocado e acesso à tela de edição.
  - **Passos:** Usuário estar na tela de Edição de Leitos.
  - Clicar no ícone de lápis de um acolhido.
  - Alterar algum campo.
  - Clicar no botão "Salvar".
  - **Resultado Esperado:** A alteração é salva e a tabela é atualizada.
- **CT02** – Desalocação bem-sucedida de um acolhido.
  - **Descrição:** Verificar se é possível desalocar um acolhido de um

leito.

- **Pré-requisito:** Ter um acolhido alocado e acesso à tela de edição.
- **Passos:** Usuário estar na tela de Edição de Leitos.
- Clicar no ícone de lixeira de um acolhido.
- **Resultado Esperado:** O sistema deve exibir uma mensagem de confirmação: “Deseja mesmo desalocar o acolhido (sim) (Não)”

#### 2.16.4. Teste de Aceitação Negativo

- **CT03** – Tentar editar sem selecionar nada.
- **Descrição:** Verificar se o que acontece se clicar no botão "Salvar" sem que nenhuma alteração tenha sido feita.
- **Pré-requisito:** Ter acolhidos alocados cadastrados no sistema.
- **Passos:** Usuário estar na tela de Pesquisa de Vagas e Leitos.
  - Não fazer nenhuma alteração na tabela.
  - Clicar no botão "Salvar".
- **Resultado Esperado:** O sistema exibe mensagem de erro: “Nenhuma alteração feita para salvar”.

#### 2.16.5. Requisitos Não Funcionais

- **Desempenho:** A tela deve carregar e responder às ações de busca e edição em no máximo 2 segundos.
- **Segurança:** As ações de edição e remoção (desalocação) são restritas a usuários autenticados e autorizados.
- **Escalabilidade:** O sistema deve ser capaz de gerenciar e exibir milhares de registros na tabela de forma eficiente e sem degradação perceptível do desempenho.
- **Manutenibilidade:** O módulo de gestão deve ser projetado para permitir a inclusão de novas colunas ou funcionalidades de edição no futuro.
- **Compatibilidade:** Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- **Usabilidade:** A interface deve ser clara e intuitiva, com a busca e a tabela de resultados facilmente compreensíveis pelo usuário.

### 3. ANEXOS

- Link do repositório Git Hub : <https://github.com/gabriel7419/TCCSistemas>

- .pdf das telas de Wireframes e/ou Protótipos.
- Sebrae Canva: → [Acesso](#) ←