

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

2025

UniALFA Umuarama / Paraná

1. INTRODUÇÃO

A construção deste sistema surgiu da necessidade de modernizar os processos internos da instituição, 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 ou casa de passagem ficam dispersos em diferentes documentos físicos ou planilhas isoladas.

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.

Esperamos que este Sistema de TI contribua significativamente para a sociedade, para a organização e para o cliente final, 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.

Metodologia:

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 que foi passado para a equipe, em relação ao controle de estoque dos alimentos, material de higiene e limpeza, ao cadastro de acolhidos, controle de leitos e controle de patrimônio.

Resultados preliminares:

A fase de levantamento de requisitos, através da entrevista, proporcionou uma identificação dos processos manuais atualmente utilizados na gestão do albergue. Identificou-se uma grande dependência de registros em papel e planilhas.

Com base no levantamento de requisitos, foi possível identificar um conjunto de funcionalidades consideradas essenciais para o sistema.

- 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 de acolhimentos, informações sobre necessidades específicas (saúde, documentação, etc.)

- Controle de Patrimônio: Controlar itens patrimoniais e seu estado.

- Controle de estoque: alimento, material de higiene e limpeza: Buscar, visualizar e dar baixa nos produtos.

Dificuldades Encontradas:

Com sorte não tivemos dificuldade em encontrar uma Instituição ou sequer entrar em contato com os funcionários porque um dos 3 integrantes do grupo do TCC já tinha a Instituição como cliente a muito tempo da empresa onde o mesmo trabalha.

Cronograma:

Entrega parcial do framework contendo as telas de controle de estoque, controle de patrimônio, cadastro dos acolhidos e gestão de vagas e leitos. Feito parcialmente todos os critérios de avaliação na Documentação.

Próximos Passos:

Fazer uma análise Detalhada das Informações Coletadas: Organizar todas as informações que coletamos de forma clara e estruturada.

Priorizar as funcionalidades: Com base nas necessidades identificadas, vamos priorizar as funcionalidades que o sistema deve ter. Concentrar nas funcionalidades mais críticas e que trarão maior impacto para a casa de acolhimento, como as funcionalidades que já identificamos (Gestão de vagas e leitos, Cadastro Completo dos Acolhidos, Controle de Patrimônio, Controle de estoque).

Decidir qual tecnologia vamos usar na codificação do sistema e também qual banco de dados.

2. PROJETO DO SISTEMA

1.1. User Story – Controle de Estoque

Como, responsável pelo estoque,

Eu quero cadastrar e controlar os produtos,

Por quê assim consigo saber o que está em falta e evitar desperdícios .

1.1.1 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.

1.1.2 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.

1.1.3 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 produto.
 - **Resultado Esperado:** O produto é salvo com sucesso.
- **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:** Estar logado no sistema
 - Acessar a tela de pesquisa.
 - Selecionar “alimento” no filtro de tipo.
 - **Resultado Esperado:** O sistema vai exibir apenas os produtos do tipo “Higiene” .

1.1.4 Teste de Aceitação Negativo

- **CT03** – 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”.

1.1.5 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.

1.2. User Story – Gestão de Leitos

Como, coordenador do abrigo,

Eu quero controlar a ocupação dos leitos e ver quem está em cada um,

Por quê para garantir organização e evitar superlotação.

1.2.1 Critérios

- Cada leito pode ter no máximo 4 acolhidos.
- O sistema deve calcular automaticamente o status do leito (Livre, Parcial, Ocupado).
- A listagem dos leitos deve exibir ocupação e nomes dos acolhidos.

1.2.2 Regras de negócio:

- Um leito com 0 acolhidos é “Livre”.
- De 1 a 3 acolhidos, é “Parcial”.
- Com 4, é “Ocupado”.
- Não é permitido cadastrar acolhido em leito cheio.

1.2.3 Teste de Aceitação Positivo

- **CT01** – Atualização de Status do Leito.
 - **Descrição:** Verificar se o status muda corretamente com a entrada/saída de acolhidos.
 - **Pré-requisito:** Leito com acolhidos cadastrados.
 - **Passos:** Remover ou adicionar acolhidos.
 - **Resultado Esperado:** Status muda automaticamente para Livre, Parcial ou Ocupado.
- **CT02** - Exibir de forma correta do status do leito após alocação
 - **Descrição:** Verifica se o status do leito é atualizado corretamente após a entrada de um acolhido.
 - **Pré-requisito:** Leito parcialmente ocupado.
 - **Passos/ação:** acessar a tela de leitos
 - Acolher mais uma pessoa no mesmo leito.
 - Acessar a tela de leitos
 - **Resultado Esperado:** Verificar se o status muda para “Ocupado” se for o quarto acolhido, ou para “Parcial” se for entre 1 e 3.

1.2.4 Teste de Aceitação Negativo

- **CT03** – Tentativa de alocar acolhido em leito cheio.
 - **Descrição:** Verifica se o sistema impede a alocação de um acolhido em um leito que já está cheio.
 - **Pré-requisito:** Um leito já deve estar cheio, com 4 acolhido.
 - **Passos:** Acessar a tela de gestão de leitos.
 - Selecionar um leito que já está em uso.
 - Tentar alocar um novo acolhido nesse leito.
 - **Resultado Esperado:** O sistema exibe mensagem de erro: “O leito já está cheio, por favor selecione outro leito”.

1.2.5 Requisitos Não Funcionais

- Desempenho: Atualização de status deve ocorrer em tempo real ao salvar.
- Segurança: Só usuários autorizados podem remover acolhidos de leitos.
- Escalabilidade: Deve suportar até 100 leitos.
- Manutenibilidade: O módulo de leitos deve permitir novos campos no futuro.
- Compatibilidade: Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- Usabilidade: Tela clara com informações principais visíveis e botão “Ver detalhes” acessível.

1.3. User Story – Controle Patrimonial

Como administrador do sistema,

Eu quero cadastrar bens patrimoniais com estado servível ou inservível,

Por quê preciso controlar os itens físicos da instituição para manutenção e substituição.

1.3.1 Critérios

- Deve ser possível cadastrar nome, quantidade, estado e detalhes do item.
- Deve permitir editar o estado, quando necessário.
- Listar os itens em uma tabela.

1.3.2 Regras de negócio:

- Itens inservíveis não podem ser alocados a novos usos.
- O sistema deve manter um histórico de alteração de estado.

1.3.3 Teste de Aceitação Positivo

- **CT01** – Cadastro de item patrimonial;
 - **Descrição:** Verifica se o sistema permite cadastrar um item com dados válidos.

- **Pré-requisito:** Acesso à tela de cadastro.
 - **Passos:** Acessar tela de bens patrimoniais.
 - Preencher os campos obrigatórios.
 - Clicar em “Salvar”.
 - **Resultado Esperado:** O item é salvo e listado corretamente.
-
- **CT02** - Alterar estado de bem patrimonial
 - **Descrição:** Verifica se é possível editar o estado de um item de “Servível” para “Inservível”.
 - **Pré-requisito:** Um item cadastrado como “servível”.
 - **Passos/ação:** Acessar a tela de listagem de patrimônio.
 - Clicar no botão de editar estado.
 - Alterar o item para “Inservível” e confirmar.
 - **Resultado Esperado:** O sistema deve salvar e listar o item atualizado na tela de listagem.

1.3.4 Teste de Aceitação Negativo

- **CT03** –Estado inválido
 - **Descrição:** Verifica se o sistema impede salvar sem selecionar o estado.
 - **Pré-requisito:** Tela de cadastro.
 - **Passos:** Preencher nome, quantidade e detalhes.
 - Deixar o campo “Estado” vazio.
 - Clicar em “Salvar”.
 - **Resultado Esperado:** O sistema exibe erro e impede o cadastro.

1.3.5 Requisitos Não Funcionais

- Desempenho: Deve carregar a listagem em até 2 segundos.
- Segurança: Apenas usuários autenticados podem alterar o estado.
- Escalabilidade: Possibilidade de adicionar categorias no futuro.
- Manutenibilidade: Estrutura em módulos separados.
- Compatibilidade: Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- Usabilidade: Campos claros, uso de dropdown para evitar erros.

1.4. User Story – Cadastro de Acolhidos

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

1.4.1 Critérios

- O sistema deve permitir o preenchimento de todos os campos obrigatórios.
- campos condicionais devem aparecer somente quando necessário (ex:

- nome do cônjuge se for casado).
- Deve registrar a data e hora de entrada e saída.

1.4.2 Regras de negócio:

- A idade deve ser calculada automaticamente com base na data de nascimento.
- Não é permitido salvar o cadastro com campos obrigatórios em branco.
- Apenas os acolhidos com vínculo familiar informado devem preencher “com quem e cidade”.

1.4.3 Teste de Aceitação Positivo

- **CT01** – Cadastro completo do acolhido
 - **Descrição:** Verifica se é possível cadastrar um acolhido com dados válidos.
 - **Pré-requisito:** Estar na tela de cadastro de acolhido.
 - **Passos:** Preencher todos os campos obrigatórios.
 - Clicar em “Salvar”.
 - **Resultado Esperado:** Acolhido salvo com sucesso e registrado no sistema.
- **CT02** - Cálculo automático do campo “Pode Retornar?”
 - **Descrição:** Verificar se o campo “Pode Retornar?” é preenchido automaticamente baseado nas regras.
 - **Pré-requisito:** Cadastro com o preenchimento automático e correto das regras.
 - **Passos/ação:** Estar logado no sistema.
 - Acessar a página de cadastro de acolhidos.
 - Preencher os dados de entrada, histórico e vínculo familiar.
 - Clicar em “Salvar”.
 - **Resultado Esperado:** O sistema deve definir o campo pode entrar?. Automaticamente de acordo com as regras.

1.4.4 Teste de Aceitação Negativo

- **CT03** – Campo obrigatório não preenchido
 - **Descrição:** Verifica se o sistema impede o cadastro sem preencher “data de nascimento”.
 - **Pré-requisito:** Tela de cadastro.
 - **Passos:** Deixar o campo “data de nascimento” sem preencher.
 - Clicar em “Salvar”.
 - **Resultado Esperado:** O sistema exibe mensagem de erro: “Todos os campos obrigatórios devem ser preenchidos”.

1.4.5 Requisitos Não Funcionais

- **Desempenho:** O sistema deve responder a ações do usuário em no máximo 2 segundos.
- **Segurança:** Os dados dos acolhidos devem ser armazenados com criptografia.

- Escalabilidade: Permitir integração futura com outros serviços sociais.
- Manutenibilidade: Campos controlados por componentes reusáveis.
- Compatibilidade: Funcionar nos navegadores modernos, como o Chrome, Edge e Firefox.
- Usabilidade: Formulários dinâmicos com campos condicionais e validação visual.

1.5. 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.

1.5.1 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.

1.5.2 Regras de negócio:

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

1.5.3 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 em "Logar".
 - **Resultado Esperado:** Sistema redireciona o usuário para a tela principal.

1.5.4 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 em "Logar".
 - **Resultado Esperado:** Sistema exibe mensagem de erro ("Usuário ou senha inválidos").

○

1.5.5 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).

3. ANEXOS

- Link do repositório Git Hub : <https://github.com/gabriel7419/TCCSistemas>
- .pdf das telas de Wireframes ou Protótipos.