

# Projeto de Desenvolvimento de Solução de Software para Comunidade Local

## **CCC** **Community Care Connection**

Tecnologia em Análise e Desenvolvimento de Sistemas  
1º Semestre de 2024

Alunos: Carlos Eduardo Alves de Souza - CP3031268  
Cleorbeth Oliveira Santos - CP3030989  
Giuliano De Lima Bengevenga - CP3031616

Coordenador Geral do Projeto de Extensão: Prof. Carlos Beluzo  
Coordenador do Curso: Diego Fernandes Gonçalves Martins  
Professores Envolvidos: Zady Castaneda Salazar  
Ricardo Barz Sovat

**Agosto de 2024**

## **SUMÁRIO**

1.	Introdução	3
2.	O Problema	4
3.	Objetivos	5
	Objetivo Geral	5
	Objetivos Específicos	5
4.	Justificativa	6
5.	Usuário Piloto	7
6.	Metodologia	8
7.	Tecnologias e Ferramentas	9
8.	Etapas do Desenvolvimento	10
9.	Meta 1: Elaborar Documentação de uma Prova de Conceito de um Programa de Computador (CMPALGP - Algoritmos e Programação)	11
10.	Meta 2: Implementar uma Prova de Conceito de um Programa de Computador (9. CMPLGP1 - Linguagem de Programação)	12
11.	Estrutura Analítica do Projeto (EAP)	13
12.	Dicionário de EAP	14
13.	Cronograma	15
14.	Considerações Finais	16

# 1.Introdução

O Community Care Connect surge para enfrentar um desafio crucial que afeta muitas comunidades: a dificuldade na gestão e no acesso a informações precisas sobre recursos essenciais de assistência social.

Em um cenário onde a fragmentação de dados e a ausência de uma plataforma centralizada comprometem a eficiência das operações das ONGs e projetos de caridade, surge a necessidade de uma solução integrada que facilite o rastreamento e a administração desses recursos.

Este projeto visa desenvolver uma ferramenta de desktop que centralize informações sobre estoques e doações, proporcionando uma visão consolidada e em tempo real sobre a disponibilidade e alocação de recursos. Ao facilitar a coordenação entre diferentes entidades e melhorar a transparência na gestão dos recursos, o Community Care Connect pretende não apenas otimizar as operações das organizações de assistência social, mas também garantir que a ajuda chegue de maneira mais precisa e eficaz às pessoas que mais necessitam.

Com isso, o projeto busca fortalecer o impacto positivo na comunidade, promovendo um sistema de assistência social mais organizado e eficiente, e contribuindo para a melhoria da qualidade de vida das populações vulneráveis.

## 2.O Problema

Atualmente, muitas comunidades enfrentam desafios significativos na gestão e no acesso a informações sobre os recursos disponíveis em seus estoques de assistência social. Esse problema é causado principalmente pela fragmentação das informações, que são frequentemente mantidas em sistemas distintos e não integrados, ou até mesmo em registros manuais. Como resultado, os dados sobre o que está disponível tornam-se desatualizados e imprecisos, dificultando a visão clara e precisa dos recursos em cada momento. Além disso, a falta de uma plataforma centralizada para gerenciar essas informações impede uma coordenação eficiente entre organizações e autoridades locais, resultando em uma gestão ineficaz dos recursos e na dificuldade em atender às necessidades emergentes.

Esse cenário afeta diretamente as organizações de assistência social, que enfrentam desafios na administração de seus recursos e na coordenação das atividades. A ausência de um sistema unificado para rastreamento e atualização dos estoques pode levar a decisões inadequadas sobre a distribuição de ajuda, comprometendo a eficácia das ações realizadas. Consequentemente, os beneficiários da assistência social também são impactados, pois a falta de alimentos e outros itens essenciais pode resultar em atrasos e insuficiências na ajuda recebida, agravando problemas de saúde e bem-estar na comunidade.

A comunidade em geral sofre com a desigualdade na distribuição de recursos, o que pode aumentar a vulnerabilidade de famílias e indivíduos que dependem das ajudas fornecidas. A desorganização e a falta de acesso a informações precisas contribuem para uma gestão ineficiente e um desperdício de recursos, afetando a capacidade das organizações de atender às necessidades de forma adequada. Da mesma forma, as organizações enfrentam dificuldades em suas operações e na prestação de contas aos financiadores e stakeholders, o que pode comprometer suas metas e a eficácia das suas ações.

## 3. Objetivos

### Objetivo Geral

O objetivo principal do Community Care Connect é desenvolver uma aplicação de desktop que permita às comunidades acompanhar e gerenciar eficientemente os recursos disponíveis em suas áreas locais. Especificamente, o projeto visa facilitar o acesso dos membros da comunidade a informações sobre recursos como centros comunitários, pontos de coleta de alimentos e clínicas de saúde. Promover a colaboração entre organizações locais e autoridades para otimizar a distribuição de recursos Fortalecer a coesão comunitária ao fornecer uma ferramenta útil para os residentes ajudarem uns aos outros.

Aumentar a conscientização sobre os recursos disponíveis na comunidade e como acessá-los.

### Objetivos Específicos

- Validação dos stakeholders
- Desenvolvimento do cronograma
- Desenvolvimento da aplicação
- Versão de testes
- Estabelecimento do canal de comunicação para feedback
- Entrega final

## 4. Justificativa

O Community Care Connect é um projeto inovador dedicado ao desenvolvimento de uma ferramenta de desktop avançada, projetada para auxiliar ONGs e projetos de caridade na gestão eficiente e no monitoramento detalhado de seus recursos. A necessidade deste projeto é fundamentada na crescente demanda por soluções que melhorem o acesso a recursos essenciais, como alimentos, e roupas, que são cruciais para o bem-estar e a qualidade de vida das comunidades vulneráveis.

Ao oferecer uma plataforma integrada e intuitiva, o Community Care Connect visa não apenas simplificar o acompanhamento dos recursos disponíveis, mas também fortalecer a coordenação entre diferentes organizações e facilitar uma resposta mais ágil e eficaz às necessidades emergentes. Através desta ferramenta, esperamos promover uma gestão mais transparente e eficiente, melhorar a alocação dos recursos e, conseqüentemente, contribuir para uma maior equidade e suporte às populações que mais necessitam.

## 5. Usuário Piloto

usuário piloto do projeto chama-se Fernando Keller, membro da comunidade Paróquia Nossa Senhora Auxiliadora, foi apresentado a proposta e tomou a frente como usuário piloto do projeto, por ser um membro ativo da comunidade, durante o processo de desenvolvimento do projeto sempre auxiliou com feedbacks a equipe. A expectativa é de que o projeto atenda as demandas relatadas e que haja uma melhora significativa de como são gerenciados as doações.

## 6. Metodologia

Para se desenvolver o projeto foi utilizada a seguinte metodologia:

- Levantamento inicial de problemas que justificassem o projeto
- Reunião entre as partes interessadas
- Definição de função na equipe
- Planejamento inicial
- Escolha das tecnologias (linguagem C, IDE CodeBlocks)
- Desenvolvimento do programa



## 7. Tecnologias e Ferramentas

O projeto será desenvolvido inteiramente com a linguagem de programação C, utilizando as bibliotecas `stdio.h`, `stdlib.h`, `string.h`, `locale.h`, `ctype.h` e a IDE que será utilizada para o desenvolvimento da aplicação será o programa CodeBlocks.

## 8. Etapas do Desenvolvimento

### 1. Planejamento Detalhado

#### 1.1 Definição do Escopo:

- Descrição do Projeto: Detalhar as funcionalidades da aplicação, metas do projeto e entregáveis.
- Requisitos do Projeto: Documentar requisitos funcionais e não funcionais, como a necessidade de rastreamento de recursos, colaboração entre organizações e suporte ao usuário.

#### 1.2 Planejamento dos Recursos:

- Equipe: Confirmar a equipe envolvida e suas responsabilidades. Ajustar conforme necessário.
- Recursos Financeiros: Estabelecer o orçamento e alocar recursos financeiros para cada fase do projeto.

#### 1.3 Planejamento de Cronograma:

- Cronograma: Criar um cronograma detalhado com marcos importantes e datas de entrega.
- Gerenciamento de Tempo: Utilizar ferramentas de gerenciamento de projetos para monitorar o progresso e ajustar o cronograma conforme necessário.

#### 1.4 Planejamento de Comunicação:

- Plano de Comunicação: Definir como a comunicação será gerida entre a equipe do projeto, stakeholders e usuários finais.
- Relatórios: Estabelecer a frequência e o formato dos relatórios de progresso.

### 2. Desenvolvimento do Protótipo

#### 2.1 Análise de Requisitos:

- Requisitos Funcionais: Identificar e documentar todas as funcionalidades essenciais da aplicação, como cadastro e rastreamento de recursos.
- Requisitos Não Funcionais: Definir requisitos de desempenho, segurança e usabilidade.

#### 2.2 Design da Aplicação:

- Design de Interface (UI/UX): Criar mockups e wireframes para garantir uma interface amigável e intuitiva.

#### 2.3 Desenvolvimento do Protótipo Funcional:

- Desenvolvimento Inicial: Codificar as funcionalidades básicas e integrar componentes.
- Testes Unitários e de Integração: Realizar testes para garantir que cada módulo funciona corretamente e interage bem com outros módulos.

### 3. Lançamento da Versão Beta

#### 3.1 Desenvolvimento de Funcionalidades Avançadas:

- Gestão de Estoques: Implementar funcionalidades avançadas, como gestão detalhada de estoques e relatórios.

#### 3.2 Testes de Usabilidade:

- Testes com Usuários Reais: Conduzir testes com um grupo selecionado de ONGs e usuários para obter feedback sobre a usabilidade e funcionalidade da aplicação.

### 3.3 Implementação de Feedback:

- Ajustes e Melhorias: Incorporar feedback dos testes para melhorar a aplicação e corrigir problemas identificados.

### 3.4 Lançamento da Versão Beta:

- Distribuição: Disponibilizar a versão beta para um grupo mais amplo de usuários e monitorar o uso e a eficácia.

## 4. Programa de Suporte ao Usuário

### 4.1 Desenvolvimento de Material de Suporte:

- FAQ e Documentação: Criar uma seção de perguntas frequentes e manuais do usuário para ajudar na navegação e uso da aplicação.

### 4.2 Estabelecimento do Canal de Atendimento ao Cliente:

- Suporte: Implementar suporte por e-mail e telefone para resolver problemas e dúvidas dos usuários.

### 4.3 Treinamento de Usuários:

- Sessões de Treinamento: Realizar sessões de treinamento para ONGs e projetos de caridade sobre como utilizar a aplicação.

## 5. Implementação e Manutenção

### 5.1 Implementação na Comunidade:

- Distribuição e Configuração: Disponibilizar a aplicação para as comunidades e ajudar na configuração inicial.

### 5.2 Monitoramento e Avaliação:

- Coleta de Feedback: Monitorar o uso da aplicação e coletar feedback contínuo para identificar áreas de melhoria.

### 5.3 Atualizações e Manutenção:

- Correções e Atualizações: Realizar manutenção regular, corrigir bugs e atualizar funcionalidades conforme necessário.

## 6. Encerramento do Projeto

### 6.1 Revisão do Projeto:

- Avaliação: Rever o projeto para garantir que todos os objetivos foram alcançados e documentar as lições aprendidas.

### 6.2 Documentação Final:

- Relatórios: Finalizar toda a documentação do projeto e garantir que está disponível para referência futura.

### 6.3 Relatório de Lições Aprendidas:

- Análise: Documentar as lições aprendidas durante o projeto para melhorar futuros projetos.

### 6.4 Entrega Final:

- Entrega Formal: Concluir a entrega do projeto e transferir a propriedade e suporte para a equipe de manutenção.

## 9. Meta 1: Elaborar Documentação de uma Prova de Conceito de um Programa de Computador (CMPALGP - Algoritmos e Programação)

A documentação da prova de conceito do Community Care Connection detalhou a implementação e avaliação do sistema desenvolvido em C, destacando suas principais funcionalidades, como cadastro e login, gerenciamento de doações e geração de relatórios. O objetivo foi validar a eficácia das funções principais, a usabilidade da interface e a eficiência do armazenamento em arquivos de texto. Foram estabelecidos critérios de sucesso, incluindo a operação correta das funcionalidades, a adequação dos menus, a criação de um canal de comunicação com usuários e a qualidade da documentação e relatórios gerados. O processo forneceu uma base sólida para ajustes futuros e confirmou a utilidade da aplicação na gestão de recursos de assistência social.

Prova de Conceito:

<https://github.com/Drd-Crls/CCC-Community-Care-Connection/blob/577ca698e7df501be67550d178ec1d08dfd4bd00/Documentos%20Preliminares/Meta1/Prova%20de%20Conceito.pdf>

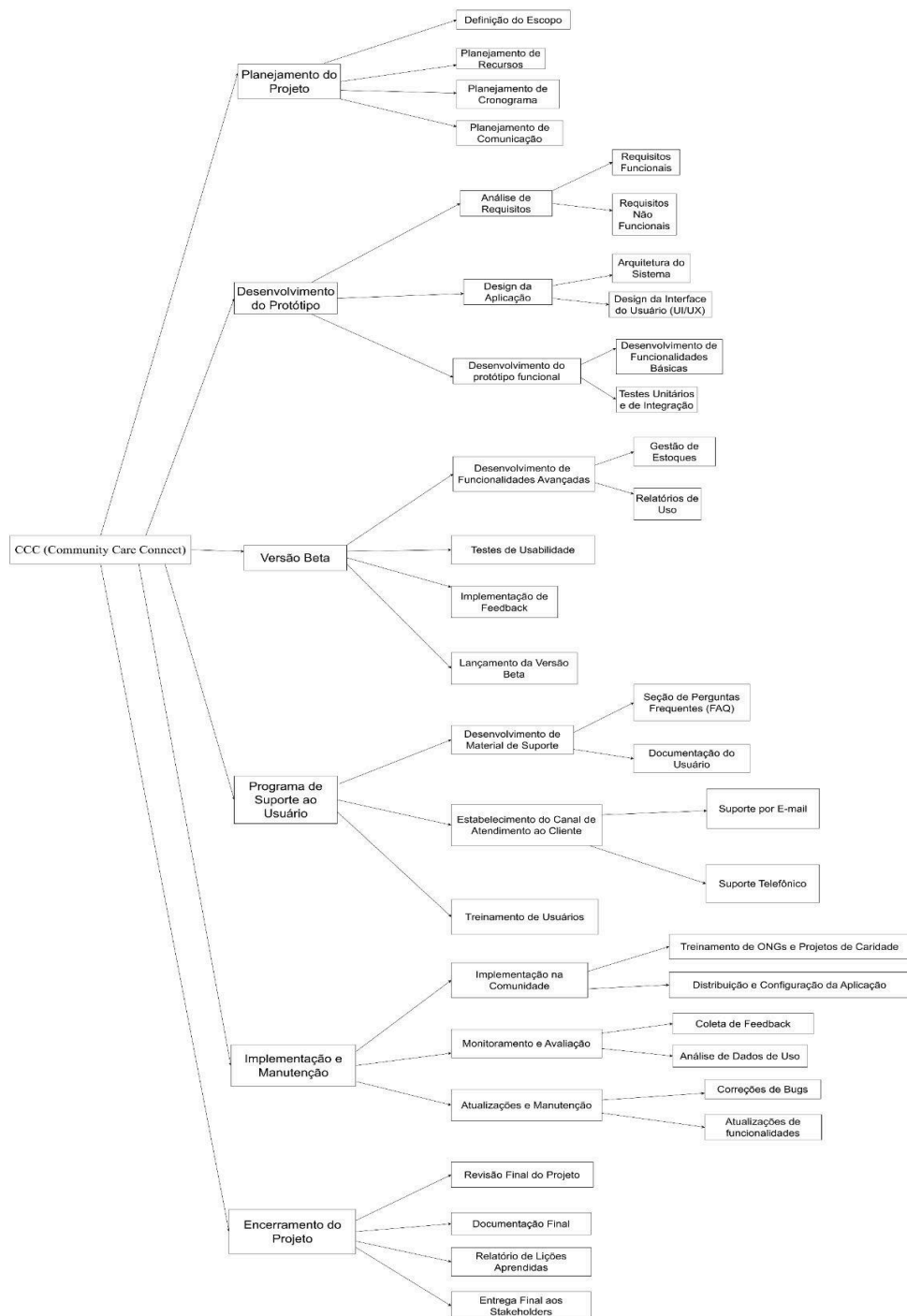
## 10. Meta 2: Implementar uma Prova de Conceito de um Programa de Computador (9. CMPLGP1 - Linguagem de Programação)

O projeto Community Care Connect foi desenvolvido utilizando a linguagem de programação C#. A aplicação incluirá funcionalidades essenciais como cadastro e login de usuários, permitindo a autenticação segura e o acesso controlado às informações. Além disso, o sistema permitirá a gestão de doações, possibilitando que os usuários registrem e rastreiem todas as contribuições de forma prática e organizada. Todos os dados serão armazenados em arquivos de texto (.txt), garantindo simplicidade na manipulação e integridade das informações. Para assegurar que a aplicação atenda plenamente às demandas do projeto, ela será submetida a testes rigorosos conduzidos por um usuário piloto, que fornecerá feedback crucial sobre a funcionalidade e usabilidade do sistema. Esses testes ajudarão a validar se a aplicação cumpre os requisitos estabelecidos e permitirá ajustes necessários para aprimorar o desempenho e a eficácia da ferramenta na gestão de recursos de assistência social.

Programa funcional:

<https://github.com/Drd-Crls/CCC-Community-Care-Connection/tree/5e5f37dbd8ed25fcf29a2a48ef925b1a41cdef15/Documentos%20Preliminares/Meta%201>

# 11. Estrutura Analítica do Projeto (EAP)



## 12. Dicionário de EAP

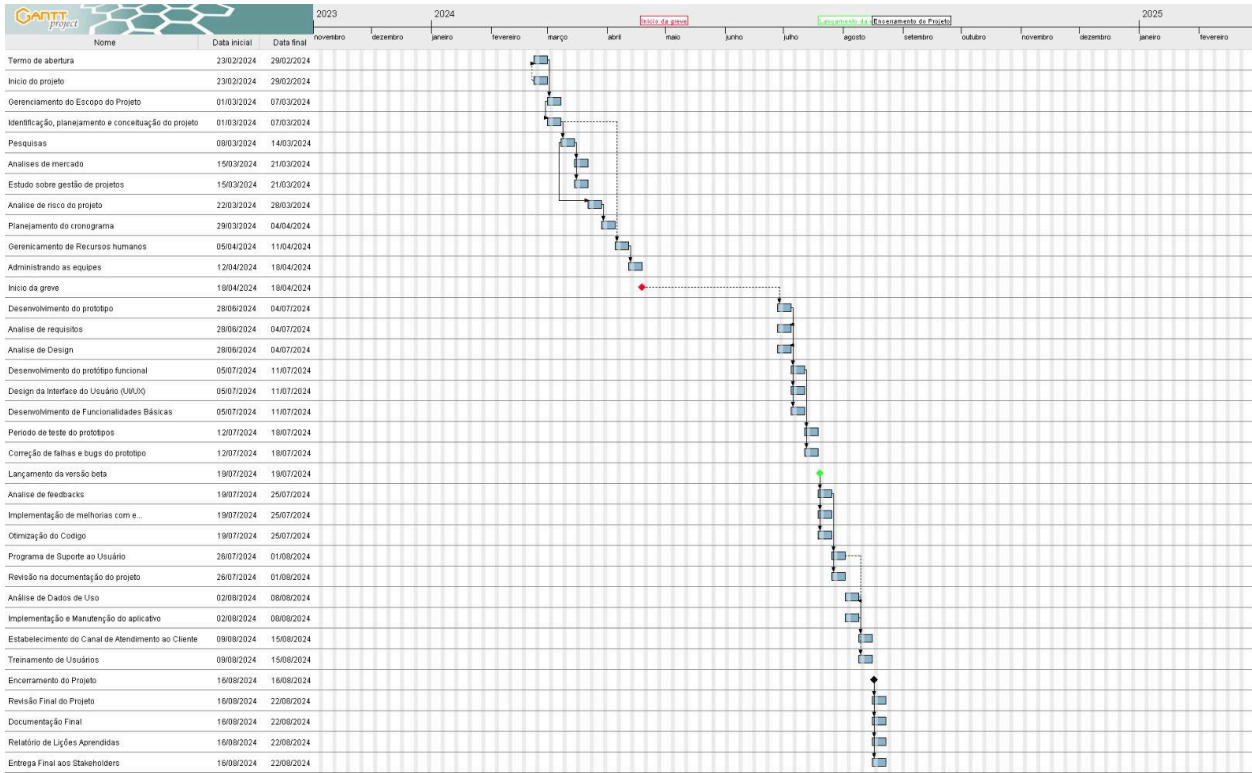
- 1.1 Planejamento do Projeto: Fase inicial do projeto que define o escopo, recursos, cronograma, riscos e comunicação.
- 1.1.1 Definição do Escopo: Estabelecimento dos limites e objetivos do projeto.
- 1.1.2 Planejamento de Recursos: Identificação e alocação dos recursos necessários.
- 1.1.3 Planejamento de Cronograma: Criação de um cronograma para todas as fases do projeto.
- 1.1.4 Identificação de Riscos: Avaliação e planejamento para mitigar riscos potenciais.
- 1.1.5 Planejamento de Comunicação: Estruturação do plano de comunicação com stakeholders.
- 1.2 Desenvolvimento do Protótipo: Construção e testes iniciais do protótipo funcional da aplicação.
- 1.2.1 Análise de Requisitos: Definição dos requisitos do sistema.
- 1.2.1.1 Requisitos Funcionais: Funcionalidades essenciais que o sistema deve ter.
- 1.2.1.2 Requisitos Não Funcionais: Requisitos de performance, segurança, etc.
- 1.2.2 Design da Aplicação: Planejamento da estrutura e interface do sistema.
- 1.2.2.1 Arquitetura do Sistema: Desenho da estrutura técnica do sistema.
- 1.2.2.2 Design da Interface de Usuário (UI/UX): Criação da interface e experiência do usuário.
- 1.2.3 Desenvolvimento do Protótipo Funcional: Codificação e construção do protótipo.
- 1.2.3.1 Desenvolvimento de Funcionalidades Básicas: Implementação das funções principais.
- 1.2.3.2 Testes Unitários e de Integração: Testes para assegurar o funcionamento das partes e do todo.
- 1.3 Versão Beta: Desenvolvimento e lançamento de uma versão preliminar do software para testes.

- 1.3.1 Desenvolvimento de Funcionalidades Avançadas: Implementação de funções adicionais.
- 1.3.1.1 Gestão de Estoques: Ferramenta para gerenciar os recursos disponíveis.
- 1.3.1.2 Relatórios de Uso: Relatórios gerados com base no uso da aplicação.
- 1.3.2 Testes de Usabilidade: Avaliação da experiência do usuário.
- 1.3.3 Implementação de Feedback: Ajustes e melhorias baseados em feedback de usuários.
- 1.3.4 Lançamento da Versão Beta: Disponibilização da versão de teste para um grupo seletivo de usuários.
- 1.4 Programa de Suporte ao Usuário: Desenvolvimento de recursos para apoiar os usuários do sistema.
- 1.4.1 Desenvolvimento de Material de Suporte: Criação de documentação e tutoriais.
- 1.4.1.1 Seção de Perguntas Frequentes (FAQ): Respostas às perguntas mais comuns.
- 1.4.1.2 Documentação do Usuário: Guia completo sobre o uso da aplicação.
- 1.4.2 Estabelecimento do Canal de Atendimento ao Cliente: Criação de canais de suporte.
- 1.4.2.1 Suporte por E-mail: Atendimento através de e-mails.
- 1.4.2.2 Suporte Telefônico: Atendimento através de chamadas telefônicas.
- 1.4.3 Treinamento de Usuários: Capacitação dos usuários para utilizar a aplicação.
- 1.5 Implementação e Manutenção: Colocação da aplicação em uso real e sua manutenção contínua.
- 1.5.1 Implementação na Comunidade: Distribuição e adaptação do sistema para os usuários.
- 1.5.1.1 Treinamento de ONGs e Projetos de Caridade: Capacitação específica para as organizações.
- 1.5.1.2 Distribuição e Configuração da Aplicação: Entrega e ajuste da ferramenta nos ambientes de uso.



- 1.5.2 Monitoramento e Avaliação: Acompanhamento e avaliação do desempenho da aplicação
- 1.5.2.1 Coleta de Feedback: Recolhimento de informações dos usuários para melhorias.
- 1.5.2.2 Análise de Dados de Uso: Estudo dos dados gerados pelo uso da aplicação.
- 1.5.3 Atualizações e Manutenção: Manutenção contínua e melhoria do sistema.
- 1.5.3.1 Correções de Bugs: Resolução de problemas técnicos.
- 1.5.3.2 Atualizações de Funcionalidades: Adição de novas funcionalidades e melhorias.
- 1.6 Encerramento do Projeto: Finalização e entrega do projeto com toda a documentação e revisão necessária.
- 1.6.1 Revisão Final do Projeto: Avaliação final do projeto para assegurar que todos os objetivos foram cumpridos.
- 1.6.2 Documentação Final: Preparação e entrega de toda a documentação relacionada ao projeto.
- 1.6.3 Relatório de Lições Aprendidas: Relatório detalhado sobre as lições aprendidas durante o projeto.
- 1.6.4 Entrega Final aos Stakeholders: Entrega formal do projeto aos interessados.

# 13. Cronograma



## 14. Considerações Finais

O projeto, Community Care Connect, visa criar uma aplicação de desktop que melhora a eficiência na coordenação de doações e gerenciamento de estoques, abordando a fragmentação e falta de transparência enfrentadas pelas ONGs e entidades de apoio. Durante o desenvolvimento, foram abordados aspectos cruciais como a criação de um sistema de cadastro e login, o gerenciamento eficaz de doações e a geração de relatórios detalhados. A prova de conceito demonstrou que a aplicação pode atender a essas necessidades com sucesso, proporcionando uma visão consolidada e em tempo real dos recursos disponíveis. A implementação foi avaliada positivamente pelo usuário piloto, que destacou a interface intuitiva e a melhoria na gestão. No entanto, foram identificadas áreas para aprimoramento, como a integração com servidores web para maior confiabilidade dos dados. O projeto, ao seguir o cronograma detalhado e incorporar o feedback recebido, promete transformar a gestão de recursos de assistência social, contribuindo para uma distribuição mais eficiente e equitativa de apoio, e reforçar o impacto positivo na comunidade.

Assinaturas: