Logotipo, nome da empresa

Descrição gerada automaticamente

**Fatec – Faculdade de Tecnologia de Araras**

**DSM – Desenvolvimento de Software Multiplataforma**

Projeto Interdisciplinar-3

Fome Zero e Agricultura Sustentável

Araras - SP

2023-1

**Fatec – Faculdade de Tecnologia de Araras**

**DSM – Desenvolvimento de Software Multiplataforma**

Projeto Interdisciplinar-3

Fome Zero e Agricultura Sustentável

Food Connect

Douglas Hugo, Daniel França, Felipe de Paula Vieira da Silva, Klayvert Ryan Alves, Luca Wyllian Beteghella

Araras - SP

2023-1

**Conteúdo**

**1 Introdução…………………………………………….4**

**2 Escopo ...................................................................5**

**2.1 EAP – Estrutura Analítica do Projeto – 6**

**2.2 Backlog – 7**

**3. Planejamento Preliminar.....................................12**

**3.1. Organização do Projeto e Matriz de Responsabilidades – 15**

**4. Desenvolvimento................................................15**

**4.1. Ferramentas – 15**

**4.2. Diagramas Utilizados – 16**

**4.3. Banco de dados – 18**

**4.4. Protótipo Visual – 23**

**5. Cronograma de execução e Orçamentação.....32**

**6. Gestão de Riscos e Problemas..........................34**

**7. Conclusão.............................................................37**

**8. Referências bibliográficas...................................39**

# 1. Introdução

Este documento descreve o PI-3, Projeto Integrador do 3º período do curso de Desenvolvimento de Software Multiplataforma, e integra os conhecimentos obtidos nas disciplinas Gestão Ágil de Projetos, Desenvolvimento Web III, Banco de Dados não Relacional e Interação Humano Computador. O tema selecionado para o projeto foi o 2º Objetivo de Desenvolvimento Sustentável da ONU – Fome Zero e Agricultura Sustentável. O desafio proposto foi o desenvolvimento de um produto de software com potencial para contribuir com o tema. O planejamento preliminar do projeto foi feito com o PMCanvas. Os requisitos foram tratados com histórias de usuário, modelagem de casos de uso e o diagrama de atividades da UML.

A aplicação foi desenvolvida usando Django, MongoDB, HTML, CSS e JavaScript.

Os requisitos obrigatórios estabelecidos foram: consumo de API, utilização de banco de dados não relacional, utilização dos conceitos e práticas da gestão ágil de projetos e de interface humano-computador.

Os artefatos gerados no desenvolvimento do projeto estão no repositório do projeto no GitHub: github.com/Klayvert2003/PI\_3Semestre\_2023.

Nesse documento foram organizadas as informações do planejamento, análise e construção do produto.

# 2. Escopo

Escopo do Projeto é o detalhamento de todo o trabalho necessário para entregar um produto, serviço ou resultado.

Este documento contém informações importantes, como: descrição e limites do projeto, objetivos, custos, prazos, responsabilidades, entregas, tarefas, restrições, premissas, além de critérios de validação. De forma geral, ele traz um foco para o projeto.

A linha de base do escopo é composta pela EAP – Estrutura Analítica do Projeto e o Backlog do Produto.

Nossa visão é criar uma plataforma online inovadora que seja uma referência no combate à fome, conectando de forma prática e eficiente instituições e indivíduos engajados na segurança alimentar. Através dessa plataforma, buscamos facilitar o processo de doações e trabalho voluntário, promovendo ações colaborativas e maximizando o impacto das iniciativas voltadas à alimentação.

Queremos estabelecer um ambiente digital que promova a conectividade entre instituições que necessitam de doações e pessoas dispostas a contribuir, seja com recursos financeiros, tempo ou habilidades. Ao facilitar esse encontro, nossa plataforma permitirá uma distribuição mais eficiente dos alimentos e aumentará o alcance das ações solidárias.

Nossa visão é impulsionada pela crença de que a conectividade e o compartilhamento de recursos são essenciais para combater efetivamente a fome. Queremos criar uma comunidade online na qual as instituições e os voluntários possam interagir de maneira transparente e confiável, possibilitando a troca de informações, solicitações de ajuda e o compartilhamento de boas práticas.

Ao fornecer uma solução prática e eficiente, queremos não apenas facilitar o processo de doações, mas também promover uma cultura de solidariedade e engajamento na sociedade. Através da nossa plataforma, esperamos aumentar a conscientização sobre a segurança alimentar e incentivar cada vez mais pessoas a se envolverem ativamente no combate à fome.

Nosso objetivo final é ser reconhecidos como a plataforma de referência, sendo um elo confiável e eficaz entre instituições e indivíduos comprometidos em promover a segurança alimentar. Ao atingir essa visão, esperamos contribuir significativamente para a redução da fome e o fortalecimento das comunidades, proporcionando uma plataforma que faça a diferença na vida das pessoas necessitadas.

## 2.1 EAP – Estrutura Analítica do Projeto

A EAP é a decomposição hierárquica do trabalho necessário para que os objetivos do projeto sejam atingidos. O trabalho é dividido em partes que podem ser entendidas, gerenciadas e estimadas mais facilmente. É uma estrutura hierárquica que assegura a visão completa do projeto em cada nível de análise. O nível mais baixo da EAP corresponde aos pacotes de trabalho. A decomposição em pacotes de trabalho permite obter estimativas mais precisas de esforço, duração e custos e definir os responsáveis pela execução. A EAP é a base para a elaboração da lista de atividades do projeto.

Diagrama

Descrição gerada automaticamente

Figura - EAP

## 2.2. Backlog Segue requisitos funcionais que foram desenvolvidas em nosso projeto, abaixo estão descritas como história de usuário:

**Tela de Login #3:**

* **ID:** US003
* **Prioridade:** Alta
* **Título:** Como usuário, desejo ter uma tela de login no sistema.
* **Descrição:** Eu, como usuário do sistema, gostaria de ter uma tela de login para inserir minhas credenciais e acessar as funcionalidades do sistema.
* **Critério de Aceitação:** Deve haver uma tela de login com campos para inserir nome de usuário e senha. O sistema deve validar as credenciais fornecidas pelo usuário. Se as credenciais forem inválidas, uma mensagem de erro adequada deve ser exibida.

**Tela de Cadastro #4:**

* **ID:** US004
* **Prioridade:** Alta
* **Título:** Como usuário, desejo ter uma tela de cadastro no sistema.
* **Descrição:** Eu, como usuário do sistema, gostaria de ter uma tela de cadastro para criar uma nova conta no sistema.
* **Critério de Aceitação:** Deve haver uma tela de cadastro com campos para inserir nome, email, senha e outros dados relevantes. O sistema deve validar os campos obrigatórios e exibir mensagens de erro adequadas se necessário. Após preencher os campos e clicar em "Cadastrar", o sistema deve criar uma nova conta de usuário.

**Landing Page PF #5:**

* **ID:** US005
* **Prioridade:** Alta
* **Título:** Como usuário, desejo ter uma Landing Page para doadores pessoa física.
* **Descrição:** Eu, como usuário do sistema, gostaria de ter uma página inicial direcionada para doadores pessoa física, fornecendo informações sobre como contribuir e incentivar as doações.
* **Critério de Aceitação:** A Landing Page deve conter informações sobre a causa, impacto e como as doações são utilizadas. Deve haver um botão de chamada para ação que redirecione os doadores para a página de cadastro ou doação. A página deve ser visualmente atraente e responsiva, adaptando-se a diferentes dispositivos e tamanhos de tela.

**Landing Page Instituições#6:**

* **ID:** US006
* **Prioridade:** Alta
* **Título:** Como usuário, desejo ter uma Landing Page para instituições.
* **Descrição:** Eu, como usuário do sistema, gostaria de ter uma página inicial direcionada para instituições, fornecendo informações sobre como se cadastrar e os benefícios de fazer parte da plataforma.
* **Critério de Aceitação:** A Landing Page deve conter informações relevantes sobre o processo de cadastro e os recursos disponíveis para as instituições. Deve haver um botão de chamada para ação que redirecione as instituições para a página de cadastro. A página deve ser visualmente atraente e responsiva, adaptando-se a diferentes dispositivos e tamanhos de tela.

**Página Instituições #7:**

* **ID:** US007
* **Prioridade:** Média
* **Título:** Como usuário, desejo visualizar informações detalhadas sobre uma instituição.
* **Descrição:** Eu, como usuário do sistema, gostaria de ter acesso a uma página com informações detalhadas sobre uma instituição, incluindo missão, histórico, projetos e formas de contato.
* **Critério de Aceitação:** Deve haver uma página dedicada a cada instituição cadastrada no sistema. A página deve exibir informações como nome, descrição, missão, história, projetos em andamento e formas de contato da instituição. Os usuários devem poder acessar essa página a partir de uma lista de instituições.

**Sobre Nós #8:**

* **ID:** US008
* **Prioridade:** Média
* **Título:** Como usuário, desejo ter acesso a uma página "Sobre Nós".
* **Descrição:** Eu, como usuário do sistema, gostaria de ter acesso a uma página que forneça informações detalhadas sobre a organização responsável pelo sistema, incluindo sua missão, visão e valores.
* **Critério de Aceitação:** Deve haver uma página "Sobre Nós" que apresente informações sobre a organização responsável pelo sistema. A página deve incluir uma descrição da missão, visão, valores e objetivos da organização. As informações devem ser apresentadas de forma clara e atraente aos usuários.

**Página Contato #9:**

* **ID:** US09
* **Prioridade:** Baixa
* **Título:** Como usuário, desejo ter acesso a uma página de contato.
* **Descrição:** Eu, como usuário do sistema, gostaria de ter acesso a uma página que forneça informações de contato da organização responsável pelo sistema, para que eu possa entrar em contato quando necessário.
* **Critério de Aceitação:** Deve haver uma página de contato que exiba informações de contato, como endereço de email e formulário de contato. As informações devem ser apresentadas de forma clara e facilmente acessíveis aos usuários. Se houver um formulário de contato, ele deve funcionar corretamente e enviar as mensagens para o destino apropriado.

**Página detalhes do Doador/Voluntário #10:**

* **ID:** US010
* **Prioridade:** Média
* **Título:** Como usuário, desejo visualizar os detalhes de um doador/voluntário.
* **Descrição:** Eu, como usuário do sistema, gostaria de poder visualizar os detalhes de um doador/voluntário para saber mais sobre sua contribuição, interesses e experiências.
* **Critério de Aceitação:** Deve haver uma página dedicada para exibir os detalhes de um doador/voluntário. A página deve mostrar informações como nome, email e disponibilidade de doações/voluntariado, entre outros detalhes relevantes.

**Banco de Dados #11:**

* **ID:** US011
* **Prioridade:** Alta
* **Título:** Como usuário, desejo ter um banco de dados NoSQL para armazenar informações do sistema.
* **Descrição:** Eu, como usuário do sistema, gostaria que as informações do sistema, como dados de doadores, instituições, etc, sejam armazenadas em um banco de dados NoSQL para facilitar o acesso e a escalabilidade.
* **Critério de Aceitação:** O sistema deve ser configurado para usar um banco de dados NoSQL, como MongoDB. As informações do sistema devem ser corretamente armazenadas e recuperadas do banco de dados. A estrutura do banco de dados deve ser projetada de forma eficiente para suportar as necessidades do sistema.

**Consumo de API do Google Maps #12:**

* **ID:** US013
* **Prioridade:** Alta
* **Título:** Como usuário, desejo consumir a API do Google Maps para exibir informações geográficas relevantes.
* **Descrição:** Eu, como usuário do sistema, gostaria de usar a API do Google Maps para exibir informações geográficas relevantes, como localização de instituições, rotas para doações ou eventos relacionados à causa.
* **Critério de Aceitação:** O sistema deve ser integrado com a API do Google Maps para acessar funcionalidades como geocodificação, exibição de mapas e rotas. As informações geográficas relevantes, como endereços de instituições ou pontos de interesse, devem ser corretamente exibidas no mapa. Os usuários devem poder interagir com o mapa, como fazer zoom, obter direções ou visualizar detalhes adicionais.

**Integração para o recebimento de Emails#13:**

**• ID:** US013

**• Prioridade:** Média

**• Título:** Como usuário, desejo integrar um serviço de recebimento de emails no sistema de contato da página.

**• Descrição:** Eu, como usuário do sistema, gostaria de integrar um serviço de recebimento de emails, para receber mensagens enviadas através do formulário de contato da página. Dessa forma, eu poderia entrar em contato com a página por email e receber respostas ou notificações relacionadas ao meu contato.

**• Critério de Aceitação:**

O sistema deve ser integrado com um serviço de recebimento de emails, seguindo as melhores práticas de segurança e configuração adequada. Quando um usuário envia uma mensagem através do formulário de contato da página, essa mensagem deve ser entregue corretamente no endereço de email designado para receber os contatos.

# 3. Planejamento Preliminar

O planejamento preliminar do projeto foi elaborado com a técnica PMCanvas – Project Model Canvas, que consiste no preenchimento de um quadro com as principais informações sobre o plano do projeto. O quadro é composto por 13 quadrantes organizados em 5 colunas. As informações de cada coluna contribuem para responder as principais questões relacionadas com o plano do projeto: 1 – Por quê?, 2- O que?, 3- Quem?, 4- Como?, 5- Quando e Quanto?  
  
Calendário

Descrição gerada automaticamente

Figura - PMCanvas

Segue uma Breve descrição dos quadrantes do PMCanvas:  
  
**Por quê? (Why?)**

Problema: Descrição do problema ou necessidade que o projeto pretende resolver.

Objetivo: Declaração clara dos objetivos do projeto e os resultados esperados.

Benefícios: Enumeração dos benefícios ou valor agregado que o projeto proporcionará.

**O que? (What?)**

Escopo: Descrição detalhada do escopo do projeto, incluindo os produtos, resultados ou entregas esperadas.

Requisitos: Lista dos requisitos do projeto, incluindo requisitos funcionais e não funcionais.

Critérios de Sucesso: Indicação dos critérios que serão utilizados para determinar o sucesso do projeto.

**Quem? (Who?)**

Patrocinadores: Identificação dos patrocinadores do projeto, que são as partes interessadas responsáveis pelo suporte financeiro e político.

Stakeholders: Lista das partes interessadas envolvidas no projeto, incluindo clientes, usuários, equipes internas, fornecedores, etc.

Equipe: Descrição dos papéis e responsabilidades das pessoas envolvidas no projeto, incluindo líderes e membros da equipe.

**Como? (How?)**

Metodologia: Descrição da metodologia ou abordagem de gerenciamento de projetos que será utilizada.

Atividades-Chave: Enumeração das principais atividades que precisam ser realizadas para alcançar os resultados do projeto.

Recursos: Identificação dos recursos necessários, como recursos humanos, materiais, financeiros, tecnológicos, etc.

**Quando e Quanto? (When and How Much?)**

Cronograma: Definição do cronograma do projeto, incluindo as principais etapas, marcos e prazos.

Orçamento: Estimativa de custos e orçamento necessário para executar o projeto.

Indicadores de Desempenho: Definição de métricas e indicadores para avaliar o desempenho e o progresso do projeto.

## 3.1. Organização do Projeto e Matriz de Responsabilidades

A equipe responsável pelo projeto "Food Connect - Combate à Fome e Agricultura Sustentável" é composta pelos seguintes membros:

- Daniel França: Frontend / Figma

- Douglas Hugo: Frontend / Documentação

- Felipe Vieira: Backend / Testes

- Klayvert Alves: Backend / Integração

- Luca Willian Beteghella: Documentação / Testes

Cada membro da equipe será responsável por suas tarefas designadas, colaborando e se comunicando regularmente para garantir o progresso do projeto.

# 4. Desenvolvimento

## 4.1. Ferramentas:

As ferramentas utilizadas no desenvolvimento foram:

Django: Django é um framework de desenvolvimento web em Python. Ele fornece uma estrutura robusta e eficiente para criar aplicativos web, incluindo recursos como gerenciamento de rotas, ORM (Object-Relational Mapping), autenticação de usuários, entre outros. O Django foi escolhido como a base para o desenvolvimento da aplicação devido à sua facilidade de uso, segurança e escalabilidade.

MongoDB: O MongoDB é um banco de dados NoSQL orientado a documentos. Ele permite armazenar e gerenciar dados de forma flexível e escalável, com consultas poderosas e desempenho otimizado para aplicativos web. O MongoDB foi escolhido como o banco de dados da aplicação devido à sua capacidade de lidar com grandes volumes de dados e à sua integração bem-sucedida com o Django.

HTML: HTML (Hypertext Markup Language) é a linguagem de marcação utilizada para estruturar o conteúdo das páginas web. No desenvolvimento da aplicação, o HTML foi utilizado para criar a estrutura das diferentes páginas e elementos visuais.

CSS: CSS (Cascading Style Sheets) é uma linguagem de estilo utilizada para definir a aparência e o layout das páginas web. O CSS foi utilizado para estilizar os elementos HTML, fornecendo cores, fontes, posicionamento e outros atributos visuais para garantir uma experiência de usuário agradável e consistente.

JavaScript: JavaScript é uma linguagem de programação utilizada principalmente para adicionar interatividade e comportamento dinâmico às páginas web. No desenvolvimento da aplicação, o JavaScript foi utilizado para implementar funcionalidades interativas, validação de formulários e integração com APIs.

JIRA: O JIRA é uma plataforma de gerenciamento de projetos e problemas amplamente utilizada no desenvolvimento de software. Ele fornece recursos para rastrear tarefas, atribuir responsabilidades, monitorar o progresso do projeto e facilitar a colaboração entre os membros da equipe.

Lucidchart: O Lucidchart é uma ferramenta de criação de diagramas online. Foi utilizada no projeto para criar diagramas de fluxo, organogramas e outros tipos de representações visuais que ajudam a visualizar a arquitetura e o fluxo de trabalho da aplicação.

Essas ferramentas desempenharam papéis importantes no desenvolvimento da aplicação "Food Connect", permitindo a criação de um sistema web funcional, atraente e eficiente, que atende aos objetivos e requisitos definidos no escopo do projeto.  
  
**4.2. Diagramas utilizados**

Diagrama de Atividade: Um *Diagrama de Atividade* é uma representação gráfica que descreve a sequência de atividades, decisões, fluxos de controle e paralelismo em um processo ou fluxo de trabalho. Ele é usado para modelar a lógica de negócios, o comportamento do sistema e as interações entre os diferentes participantes em um sistema.

Para o nosso projeto, decidimos utilizar o *Diagrama de Atividade* como uma das representações gráficas desenvolvidas pelo grupo para ilustrar o fluxo de atividades e interações entre as telas do site. Esse diagrama oferece uma visão mais detalhada e sequencial das etapas e decisões envolvidas no processo, permitindo uma compreensão abrangente da lógica e das interações do sistema.

Segue abaixo o diagrama desenvolvido pelo grupo.

Diagrama

Descrição gerada automaticamente

Diagrama de Casos de Uso: *O diagrama de Casos de Uso* representa uma função específica ou um conjunto de ações que o sistema realiza em resposta a uma solicitação do usuário (ator externo). Ele descreve as interações entre o usuário e o sistema, mostrando o fluxo de eventos e as possíveis ações que podem ocorrer.

Para o nosso projeto, decidimos utilizar *o Diagrama de Casos de Uso* como uma das representações gráficas desenvolvidas pelo grupo para ilustrar as interações entre os atores externos e o sistema. Esse diagrama oferece uma visão mais detalhada e abrangente das funcionalidades e interações do sistema, permitindo uma compreensão sequencial das ações realizadas pelos atores e suas relações com o sistema.

Segue abaixo o diagrama desenvolvido pelo grupo.

Diagrama

Descrição gerada automaticamente

**4.3. Banco de Dados**

A plataforma possui funcionalidades que abrangem operações de criação, leitura, atualização e exclusão (CRUD) relacionadas aos cadastros de usuários e instituições. A seguir, são fornecidos exemplos de como essas operações são realizadas no código:

CREATE:

Existem duas classes responsáveis pela criação de cadastros de usuários: DetalhesInstituicao e DetalhesUsuario. Essas classes armazenam os dados do formulário e, após a validação desses dados, utilizam o método save para registrar o usuário na base de dados.

READ:

Há exemplos de operações de leitura em duas classes: ListaContatos e InstituicoesView. Na classe ListaContatos, é gerada uma tabela interativa que expande conforme mais usuários são registrados no sistema, exibindo informações essenciais de contato e disponibilidade dos usuários autenticados. Na classe InstituicoesView, são gerados cards com informações resumidas de instituições cadastradas. Ao clicar no botão desses cards, o usuário é redirecionado para a classe CardMapView, onde é exibido um mapa com o endereço da instituição, juntamente com mais detalhes disponíveis no registro do banco de dados.

UPDATE:

No código fornecido, a primeira função trata uma requisição GET para carregar os dados do usuário autenticado a partir do banco de dados. Esses dados são armazenados em um dicionário e passados como contexto para o HTML, permitindo que o usuário visualize suas informações cadastradas anteriormente. A segunda função trata uma requisição POST, realizando uma operação de atualização (UPDATE). Os dados inseridos pelo usuário no HTML são capturados, armazenados em um dicionário e usados para atualizar os dados do usuário no banco de dados.

DELETE:

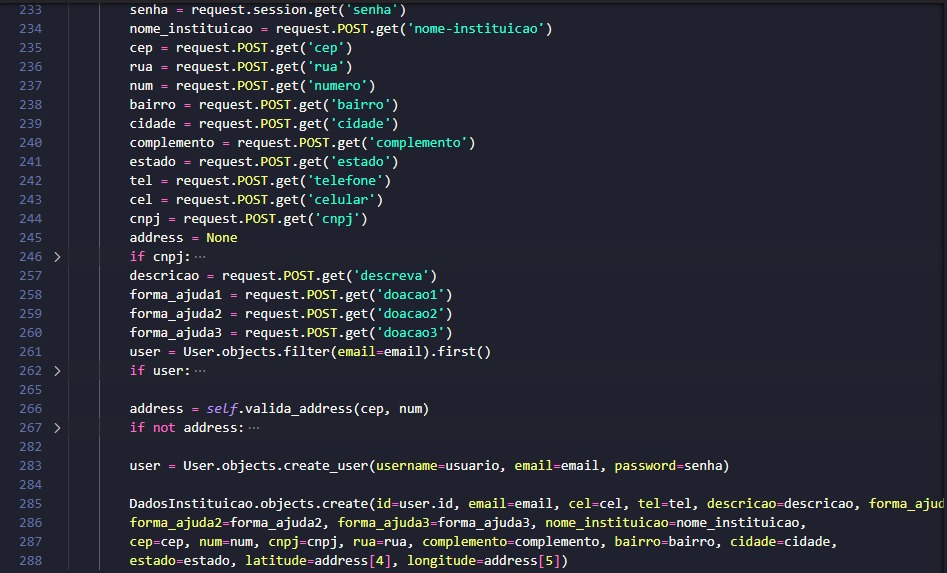
No código fornecido, a função trata uma requisição GET e, após a autenticação do usuário, obtém o ID da sessão, que corresponde ao ID do usuário cadastrado no banco de dados. Se o usuário clicar no botão de exclusão, o registro correspondente é removido com base nesse ID.

AGREGAÇÃO:

Foi estabelecida uma relação entre os modelos do projeto (as coleções core\_dadosusuarios e core\_dadosinstituicao), em que o ID de autenticação do usuário do Django é armazenado como um campo no modelo de usuário personalizado. Essa associação permite obter a ligação entre os usuários dos dois modelos, criando uma espécie de "relação" entre as coleções, similar a um banco de dados relacional.

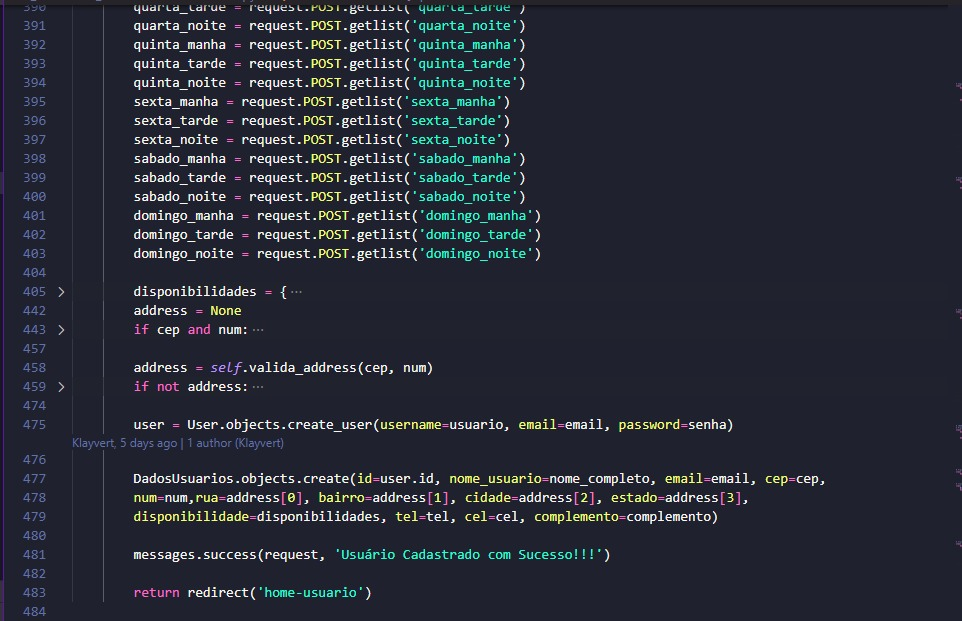
DESCRIÇÃO DE IMAGENS:

CADASTRO DE INSTITUIÇÃO



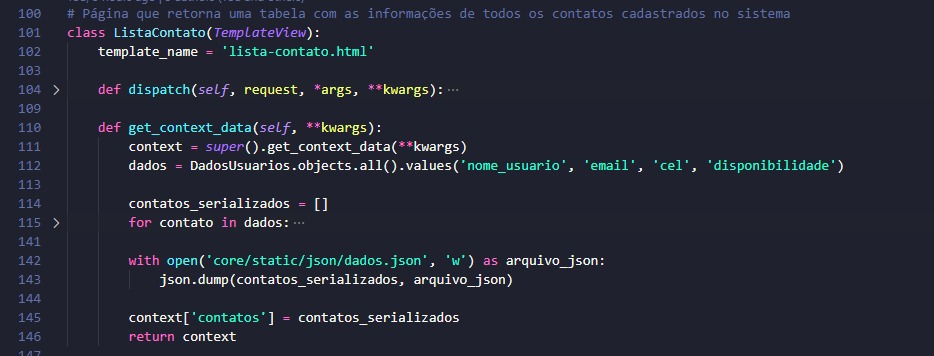
O código acima é responsável por estar cadastrando os dados das instituições no banco e no site, fazendo conexão direta com a tela de Cadastro de dados detalhados das instituições.

CADASTRO DE USUÁRIO



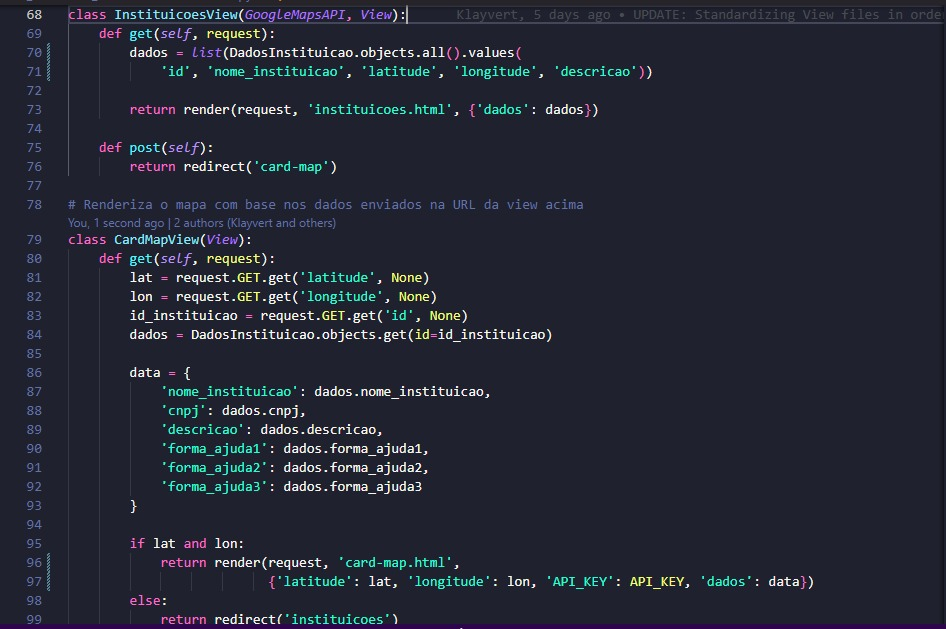
O código acima é responsável por estar cadastrando os dados dos usuários no banco e no site, fazendo conexão direta com a tela de Cadastro de dados detalhados dos usuários.

LISTA DE VOLUNTÁRIOS



Esta parte do código é responsável por estar retornando na tela de “Lista de Voluntários” os dados de todos os usuários voluntariados cadastrados no site. Puxando esses dados através do URL da view acima.

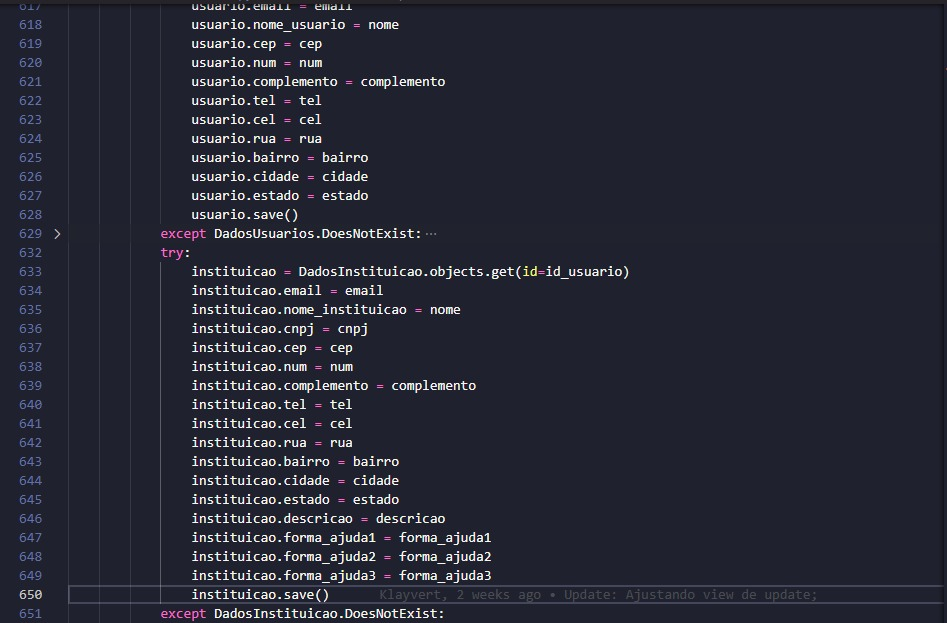
DADOS E MAPA DE INSTITUIÇÕES CADASTRADAS



Por meio da URL da view acima, o código retorna as informações e o mapa

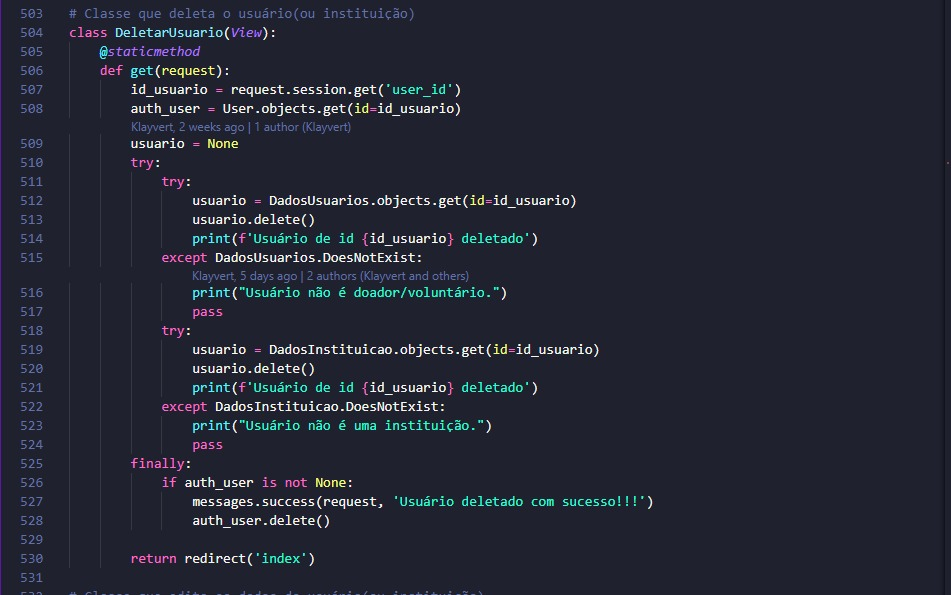
da instituição selecionada na tela de exibição do site.

ATRIBUINDO VALORES ÀS PROPRIEDADES



Aqui estamos atribuindo os respectivos valores às propriedades dos objetos usuário e instituição, para que possam realizar updates automáticos entre o site e o banco.

DELETAR USUÁRIO OU INSTITUIÇÃO



Esta etapa do código é responsável por permitir que o usuário exclua seus dados da base de dados do site e das telas atribuídas, possibilitando que o usuário tenha maior controle e liberdade sobre suas informações.

**4.4. Protótipo Visual**

Optamos pelo uso da plataforma Figma para desenvolver o protótipo visual das interfaces do site devido às suas diversas vantagens. Com o Figma, pudemos colaborar em tempo real, acessar e compartilhar o projeto de forma fácil, além de contar com recursos robustos de design. Também pudemos criar protótipos interativos, facilitar a manutenção e atualização, garantindo uma experiência de design eficiente, colaborativa e de alta qualidade.

Abaixo estão as telas do site com uma breve descrição sobre cada uma.

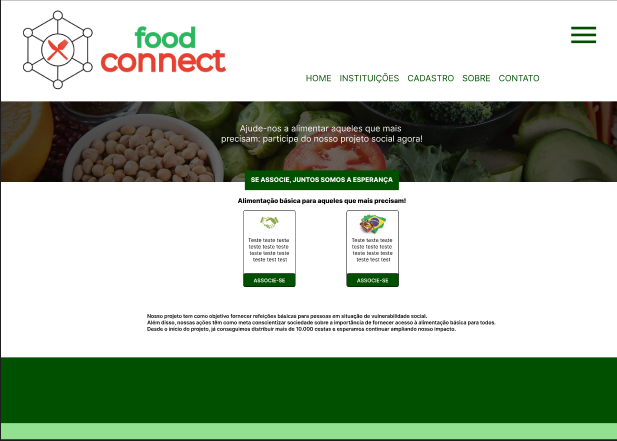
HOME

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

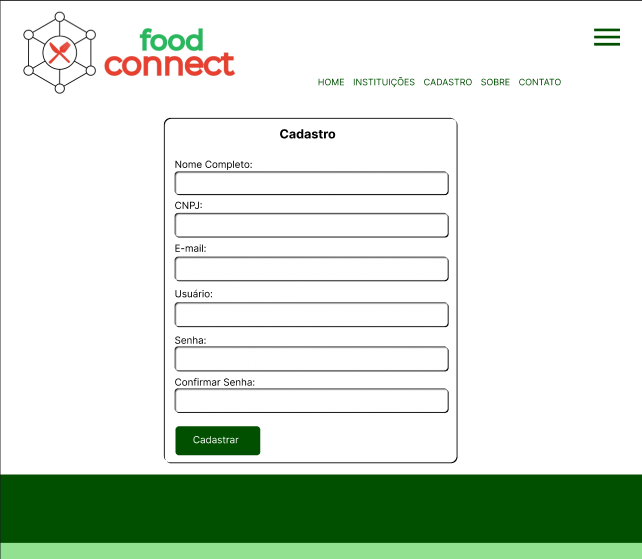
Ao entrar no site, os usuários serão direcionados para esta tela inicial, que apresenta duas opções: Doador ou Instituição. Dependendo da escolha do usuário, a página será adaptada para o tipo de usuário que está acessando-a.

LANDING-PAGE-INSTITUIÇÃO



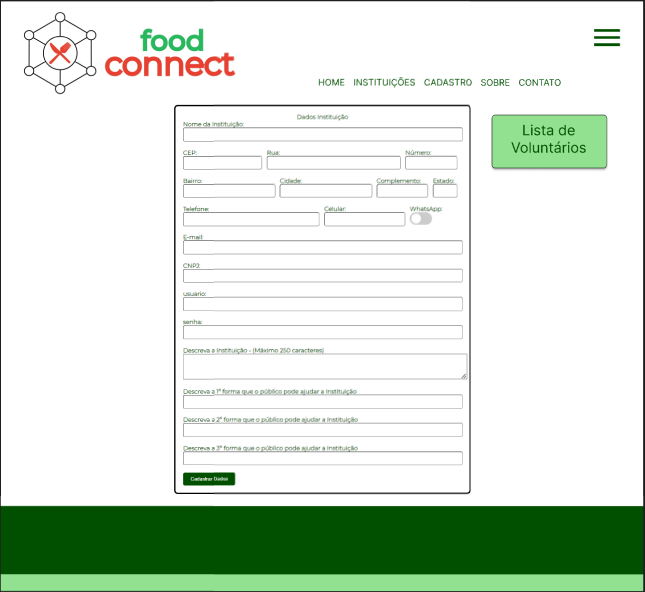
Caso o usuário escolha a opção “INSTITUIÇÃO” na tela HOME, esta será a tela seguinte que irá aparecer à ele, nesta tela contém breves textos e cards incentivando os usuários a cadastrarem sua organizações no site. Para cadastrar, basta ele clicar nas opções *“*ASSOCIE-SE*”* ou clicar diretamente em “CADASTRO”.

TELA DE CADASTRO DE INSTITUIÇÃO



Esta é a primeira Tela de Cadastro das Instituições, é aqui onde os usuários jurídicos poderão estar adicionando suas informações para poderem estar posteriormente cadastrando suas organizações.

TELA PARA ADICIONAR DADOS DA INSTITUIÇÃO



Aqui os usuários poderão estar especificando, editando ou excluindo os dados de cadastro de suas instituições.

Ao preencher este formulário, as informações de suas organizações serão exibidas para os demais usuários do site através de outra página.

Vale lembrar que esta tela de cadastro possui a opção “LISTA DE VOLUNTÁRIOS”, que será acessível somente para os usuários jurídicos.

LISTA DE USUÁRIOS VOLUNTÁRIOS

Tabela

Descrição gerada automaticamente

Nesta tela serão exibidas para as instituições as informações sobre doadores voluntários que desejam colaborar.

LANDING-PAGE-USUÁRIO COMUM



Caso o usuário escolha a opção “DOADOR” na tela HOME, esta será a tela seguinte que irá aparecer à ele, nesta tela contém textos informativos e um vídeo curto incentivando os usuários a se tornarem doadores voluntários do site. Para se cadastrar, ao clicar na opção *“*QUERO COLABORAR*”,* o usuário será direcionado a outra interface, contendo dados das instituições cadastradas.E para se cadastrar, basta clicar diretamente em “CADASTRO”.

TELA DE CADASTRO DO USUÁRIO

Tabela

Descrição gerada automaticamente

Esta é a tela onde os usuários poderão estar criando seu cadastro no site para posteriormente estarem se tornando doadores voluntários em potencial.

CADASTRO DE DOADOR VOLUNTÁRIO

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Aqui o usuário após se cadastrar, poderá estar adicionando informações sobre sua disponibilidade para doações, esses dados ficarão visíveis para as organizações na tela “LISTA DE USUÁRIOS VOLUNTÁRIOS”. Vale lembrar que os usuários também poderão estar editando ou excluindo seus dados a qualquer momento.

LISTA DE INSTITUIÇÕES CADASTRADAS

Uma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamente

Esta tela é visível para todos os tipos de usuários do site, é aqui onde todas as instituições cadastradas serão exibidas.

Ao clicar em “Ver Mapa” os usuários serão designados para outra página contendo dados mais detalhados sobre a empresa selecionada.

DETALHES DA INSTITUIÇÃO

Interface gráfica do usuário, Site

Descrição gerada automaticamente

Ao clicar em “Ver Mapa” na tela anterior, é aqui onde os usuários serão direcionados. Esta tela contém os principais dados sobre as instituições cadastradas no site, informações como: Nome da empresa, Pix e horários serão exibidas aqui.

TELA DE LOGIN

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Esta tela permite que os usuários tanto comuns quanto jurídicos, realizem um login rápido para acessar suas informações pessoais caso já possuam um cadastro, permitindo-os a editarem seus dados a respeito de suas instituições ou dados pessoais sobre disponibilidade para doar.

CONTATO

Interface gráfica do usuário, Texto, Aplicativo, Site

Descrição gerada automaticamente

Aqui é onde todos os usuários poderão estar enviando mensagens de sugestões ou dúvidas para os gerenciadores do site

SOBRE

Interface gráfica do usuário, Site

Descrição gerada automaticamente

Esta tela também está visível para todos os usuários, é onde as informações sobre os objetivos do site se encontram.

# 

# 5 Cronograma de execução e Orçamentação

O cronograma contém todas as atividades do projeto e é construído pela técnica de decomposição da EAP.

Decompõe-se cada entrega da EAP, em atividades sumárias que são decompostas em atividades e assim por diante até obter-se o nível de detalhe desejado.

Para cada atividade identificada, identificam-se os recursos necessários e suas respectivas quantidades, que irão compor o orçamento do projeto.  
  
O cronograma do projeto foi estabelecido utilizando a ferramenta de gestão de projetos JIRA. O JIRA foi escolhido devido à sua capacidade de acompanhar o progresso das atividades, atribuir responsabilidades, priorizar tarefas e gerenciar prazos de forma eficiente.

Para a organização das atividades, foram utilizados quadros Kanban no JIRA. Cada atividade foi representada como uma tarefa dentro do quadro, com colunas que refletiam o fluxo de trabalho, como "A fazer", "Em andamento" e "Concluído". As tarefas foram movidas de uma coluna para outra à medida que progrediam.

Além disso, o JIRA permitiu o acompanhamento das atividades por meio de sprints. O projeto foi dividido em quatro sprints, cada uma com uma duração de 10 dias corridos. No entanto, devido a alguns imprevistos, ocorreram atrasos em algumas atividades.

Prazo das Sprints:

Sprint #1: 06/05 – 16/05

Sprint #2: 16/05 – 26/05

Sprint #3: 26/05 – 04/06

Sprint #4: 04/06 – 14/06

Durante cada sprint, as atividades foram acompanhadas de perto no JIRA, com atualizações regulares sobre o progresso e qualquer atraso ou problema identificado. A equipe utilizou ferramentas de comunicação, como WhatsApp e Discord, para manter uma comunicação eficaz e compartilhar atualizações sobre o andamento das atividades e possíveis atrasos.

A combinação do uso do JIRA como ferramenta de gestão de projetos e a comunicação constante entre a equipe por meio de plataformas de mensagens instantâneas permitiu um acompanhamento adequado das atividades, mesmo diante dos atrasos imprevistos, garantindo que o projeto continuasse avançando e que a equipe estivesse ciente dos desafios e ajustes necessários.

Embora algumas atividades de detalhes não tenham sido executadas devido ao prazo, elas foram registradas e colocadas em backlog para futuras considerações. Essas demandas foram reconhecidas como itens pendentes que podem ser abordados em atualizações ou iterações futuras da plataforma.

Ao colocar essas demandas em backlog, a equipe reconhece sua importância e a intenção de abordá-las posteriormente para aprimorar ainda mais a experiência do usuário. Essa abordagem permite que as funcionalidades principais sejam implementadas dentro do prazo, enquanto os aprimoramentos e detalhes adicionais são tratados em momentos oportunos no desenvolvimento contínuo da plataforma.

Link do quadro (Cronograma) no JIRA:

<https://fatecometto.atlassian.net/jira/software/projects/FC/boards/1/backlog>

**Estimativa de custos:**

Os custos foram estimados principalmente com base no tempo e na mão de obra necessária para o desenvolvimento do projeto. Considerando um prazo de 40 dias, é necessário levar em conta os salários/honorários dos membros da equipe envolvidos. Essa estimativa pode variar dependendo dos valores praticados no mercado e das habilidades dos membros da equipe.

Suponhamos uma equipe composta por 5 membros com os seguintes valores horários:

Daniel França: R$ 30/hora

Douglas Hugo: R$ 20/hora

Felipe Vieira: R$ 30/hora

Klayvert Alves: R$ 30/hora

Luca Willian Beteghella: R$ 20/hora

Considerando uma média de 6 horas diárias de trabalho por membro, o custo diário da equipe seria:

Custo diário = (30 + 20 + 30 + 30 + 20) \* 6 = R$ 780,00

Multiplicando esse valor pelo prazo de 40 dias, obtemos o custo total estimado do projeto:

Custo total estimado = R$ 780,00 \* 40 = R$ 31.200  
  
\*Observação: No nosso PMCanvas ficou divergente o custo do projeto pois erramos ao não incluir todos os integrantes.

# 6. Gestão de Riscos e Problemas

**Riscos identificados no projeto:**

Limitação do prazo: Com um prazo de apenas 2 meses para o desenvolvimento da aplicação, existe o risco de não conseguir concluir todas as funcionalidades dentro do tempo estipulado. Isso pode resultar em um produto incompleto ou com funcionalidades essenciais ausentes.

Pouco conhecimento nas linguagens usadas: Se a equipe de desenvolvimento tiver pouca experiência ou conhecimento nas linguagens e tecnologias utilizadas no projeto, pode haver dificuldades na implementação correta das funcionalidades, o que pode levar a erros, retrabalho e atrasos.

**Problemas ocorridos no projeto:**

Durante o desenvolvimento do projeto "Food Connect", a equipe enfrentou alguns desafios decorrentes dos riscos identificados:

Restrição do prazo: O prazo limitado de 2 meses exigiu uma gestão cuidadosa do tempo e dos recursos disponíveis. A equipe teve que priorizar as funcionalidades mais essenciais e realizar um planejamento detalhado para garantir que os objetivos principais do projeto fossem alcançados dentro do prazo estipulado.

Pouco conhecimento nas linguagens usadas: A falta de experiência prévia com as linguagens e tecnologias utilizadas na aplicação pode ter levado a dificuldades técnicas e atrasos na implementação. A equipe teve que se dedicar a aprender e familiarizar-se rapidamente com as ferramentas para garantir um desenvolvimento eficiente e de qualidade.

**Ações necessárias para alcançar os objetivos:**

Para superar os problemas enfrentados e alcançar os objetivos do projeto, as seguintes ações foram necessárias:

Planejamento detalhado: Foi essencial realizar um planejamento cuidadoso, identificando as funcionalidades prioritárias e estabelecendo um cronograma realista. Isso permitiu que a equipe se concentrasse nas áreas mais importantes do projeto e otimizasse o tempo disponível.

Treinamento e capacitação: Diante da limitação de conhecimento nas linguagens utilizadas, a equipe precisou buscar recursos de aprendizagem, como tutoriais, documentação e treinamentos online, para aprimorar suas habilidades e adquirir o conhecimento necessário para o desenvolvimento eficiente da aplicação.

Comunicação e colaboração: A comunicação efetiva entre os membros da equipe foi fundamental para superar os desafios. Manter um diálogo aberto, compartilhar conhecimento, esclarecer dúvidas e colaborar ativamente ajudou a mitigar os riscos e a solucionar problemas de forma mais eficiente.

Gerenciamento de expectativas: Foi importante realizar uma comunicação clara e transparente com os stakeholders envolvidos no projeto, informando sobre o prazo limitado e os desafios enfrentados. Manter as expectativas realistas e alinhadas com as restrições de tempo e conhecimento ajudou a garantir uma compreensão adequada dos resultados esperados.

Embora os problemas e riscos possam ter apresentado desafios durante o projeto, as ações acima mencionadas, combinadas com o comprometimento da equipe, permitiram que os objetivos fossem alcançados dentro das limitações impostas. A superação desses obstáculos contribuiu para o desenvolvimento de uma plataforma funcional e eficiente, capaz de conectar instituições e indivíduos engajados no combate à fome.

# 7. Conclusão

Em conclusão, o projeto "Food Connect " foi resultado de um esforço coletivo e comprometido da equipe de desenvolvimento. Ao longo do processo, cada membro contribuiu com suas habilidades e conhecimentos nas áreas de frontend, backend, documentação e testes, trabalhando de forma colaborativa e mantendo uma comunicação regular.

Através do uso de metodologias ágeis, como a gestão de projetos ágil, a equipe foi capaz de adaptar-se às mudanças, responder rapidamente aos desafios e entregar um produto atingindo todos os requisitos estabelecidos. A utilização de ferramentas adequadas, como o JIRA e o Lucidchart, facilitou a organização das tarefas, o acompanhamento do progresso e a visualização da arquitetura do sistema.

Além disso, enfrentamos um grande desafio ao optar por não utilizar um framework de frontend neste projeto. A equipe assumiu a responsabilidade de criar manualmente a estrutura e a estilização das páginas utilizando HTML, CSS e JavaScript. Essa decisão exigiu um esforço adicional em termos de desenvolvimento e design, mas permitiu um maior controle sobre a personalização e adaptabilidade da interface do usuário. A utilização de um banco de dados não relacional, em particular, mostrou-se fundamental para lidar com grandes volumes de dados e garantir a eficiência da plataforma.

O sucesso alcançado no desenvolvimento do projeto "Food Connect" não teria sido possível sem o comprometimento e a dedicação de cada membro da equipe. O esforço conjunto e à busca contínua pela excelência, resultou em um produto funcional, confiável e alinhado com as necessidades do público-alvo.

O desenvolvimento do projeto não se limitou apenas à implementação do software, mas também envolveu a criação de uma visão clara e uma estratégia para enfrentar o desafio da fome e promover a segurança alimentar. A equipe demonstrou habilidades de planejamento, análise e construção, garantindo a entrega de um produto que busca fazer a diferença na vida das pessoas necessitadas.

Em suma, o esforço coletivo e o comprometimento da equipe de desenvolvimento foram fundamentais para o sucesso do projeto "Food Connect". O trabalho árduo, a colaboração e a paixão pelo tema resultaram em uma plataforma online inovadora e eficiente, capaz de promover a segurança alimentar e combater a fome, tornando-se uma referência na área. O projeto é um testemunho do potencial transformador da tecnologia e do trabalho em equipe na busca por um mundo mais justo e sustentável.

# 8 Referências bibliográficas

<https://www.lucidchart.com/pages/pt>

<https://docs.djangoproject.com/>

<https://docs.python.org/pt-br/3/tutorial/>

<https://www.mongodb.com/docs/>

<https://www.djongomapper.com/get-started/>

<https://www.euax.com.br/2018/12/gerenciamento-escopo-do-projeto/>

https://www.flowup.me/blog/project-model-canvas/