

Curso Superior de Desenvolvimento de Software Multiplataforma

Matheus Tadao Momiy, RA: 3011392423038

Maicon Rodrigues dos Santos, RA: 3011392423036

Nicolas Katsuji Nagano, RA: 3011392423037

Projeto Interdisciplinar
Engenharia de Software II
Desenvolvimento Web II
Banco de Dados Relacional

Conecta Social

Orientadores: Prof^a Cristiane Palomar Mercado
Prof^o Tiago Vanderlei de Arruda
Prof^o Jones Artur Gonçalves

Votorantim
Junho, 2025

RESUMO

O presente projeto consiste em uma plataforma de gestão de dados para iniciativas sociais de pequeno porte. Através do sistema os responsáveis pelas iniciativas sociais poderão cadastrar e gerir eventos benficiares que serão realizados, voluntários , famílias parceiras à iniciativa social e famílias que recebem auxílio através de doações, assim como as doações arrecadadas para os eventos e necessitados. O Sistema também possuirá uma área pública, onde esses projetos sociais poderão contar mais sobre a sua história, seus propósitos, compartilhar suas histórias, divulgar os eventos e centralizar informações. Os usuários responsáveis pelo projeto social poderão acessar o sistema através de um login, para gerir os dados, mas a área pública será de acesso livre para os visitantes. A plataforma foi desenvolvida de forma que possa atender diferentes tipos de projetos, para isso foi construída uma estrutura que facilita alterações de identidade visual, para que assim possa ser publicado como independente dos outros projetos sociais que fazem uso.

LISTA DE FIGURAS

Figura 1 - Mapa do site	3
Figura 2 - Logomarca	3
Figura 3 - Diagrama de caso de uso 1	7
Figura 4 - Diagrama de caso de uso 2	8
Figura 5 - Diagrama conceitual	25
Figura 6 - Diagrama de lógico	26
Figura 7 - Diagrama de classe	27
Figura 8 - Diagrama de atividade	28
Figura 9 - Diagrama de sequência	29
Figura 10 - Interface - Home	30
Figura 11 - Interface - Sobre	30
Figura 12 - Interface - Fale conosco	31
Figura 13 - Interface - CRUD	31
Figura 14 - Interface - Dashboard 1	32
Figura 15 - Interface - Dashboard 2	32

LISTA DE QUADROS

Tabela 1 – Requisitos funcionais	6
Tabela 2 - Caso de uso "Gerenciar Voluntários "	9
Tabela 3 - Caso de uso "Criar/Editar Voluntários "	11
Tabela 4 - Caso de uso "Cancelar beneficiários"	12
Tabela 5 - Caso de uso "Cancelar família"	13
Tabela 6 - Caso de uso "Cancelar doações"	14
Tabela 7 - Caso de uso "Cadastrar/Cancelar Eventos"	15
Tabela 8 - Caso de uso "Editar Eventos"	16
Tabela 9 - Caso de uso "Cadastrar/Alterar Beneficiários"	17
Tabela 10 - Caso de uso "Cadastrar/Alterar Família"	18
Tabela 11 - Caso de uso "Cadastrar/Alterar Doações"	19
Tabela 12 - Caso de uso "Fazer Login"	20
Tabela 13 - Caso de uso "Atribuir doações a família"	21
Tabela 14 - Caso de uso "Visualizar eventos"	22
Tabela 15 – Requisitos não funcionais	23

SUMÁRIO

1. DESCRIÇÃO DO PROJETO	1
1.1. Proposta do Software	1
1.2. Justificativa	2
1.3. Mapa do Site	3
1.4. Logomarca	3
2. REQUISITOS DO PROJETO	4
2.1. Levantamento de Requisitos	4
2.2. Requisitos Funcionais	6
2.3. Diagrama de Caso de Uso	7
2.4. Descrição do Caso de Uso	9
2.5. Requisitos Não Funcionais	23
3. PROJETO DO SOFTWARE	24
3.1. Tecnologias Utilizadas	24
3.2. Modelo de dados	25
3.2.1 Modelo Conceitual	25
3.2.2 Modelo Lógico	26
3.3. Diagrama de Classe	27
3.4. Diagrama de Atividades	28
3.5. Diagrama de Sequência	29
3.6. Interfaces com o usuário	30
4. ESTRATÉGIA DE TESTES	33
4.1. Níveis de Testes aplicados	33
4.2. Tipos de Testes	33
5. PROJETO	35
5.1. Link do repositório github	35
6. CONCLUSÃO	36
7. REFERÊNCIAS	37
Protótipo funcional do Figma	38
Script - Banco de dados	38

1. DESCRIÇÃO DO PROJETO

1.1. Proposta do Software

Descrição do Projeto: Plataforma para pequenos projetos comunitários e Gestão de Eventos e Doações.

O projeto visa desenvolver uma plataforma online dedicada a promover a visibilidade de eventos e iniciativas sociais na comunidade. Este site servirá como uma rede centralizada onde os membros da comunidade poderão acompanhar e participar de eventos voltados para o desenvolvimento social, especialmente aqueles destinados a apoiar crianças carentes e outras populações vulneráveis.

Através da nossa plataforma, os organizadores de eventos poderão publicar informações detalhadas sobre suas iniciativas, incluindo data, local, descrição e formas de participação. O objetivo é facilitar a mobilização da comunidade em torno dessas atividades, promovendo um ambiente colaborativo e solidário.

Além disso, a plataforma incluirá um sistema de controle de doações, permitindo que as pessoas contribuam de forma transparente e segura. As doações poderão ser direcionadas a famílias cadastradas na comunidade, que necessitam de apoio. O sistema garantirá que os recursos sejam distribuídos de maneira justa e eficaz, oferecendo um suporte real às necessidades identificadas.

Com essa abordagem, buscamos fortalecer os laços comunitários, aumentar a participação ativa dos cidadãos e assegurar que pequenos projetos comunitários recebam a visibilidade e o apoio que merecem. Acreditamos que, juntos, podemos criar um ambiente mais solidário e inclusivo, promovendo a dignidade e o bem-estar de todos os membros da comunidade.

Este projeto está alinhado com vários Objetivos de Desenvolvimento Sustentável (ODS) da ONU, destacando-se na erradicação da pobreza (ODS 1) ao apoiar crianças carentes e famílias vulneráveis; na promoção de uma educação de qualidade (ODS 4) por meio de eventos educacionais; na redução

das desigualdades (ODS 10) ao assegurar a inclusão social de grupos marginalizados; e na criação de parcerias eficazes (ODS 17) que potencializam o impacto social. Dessa forma, nossa plataforma se torna uma ferramenta essencial para promover a solidariedade e o desenvolvimento sustentável na comunidade.

1.2. Justificativa

A ideia para este projeto surgiu a partir da experiência de um conhecido do grupo, que organiza e gerencia doações e dados de pessoas atendidas por meio de métodos bastante manuais — como cadernos de anotações, memória própria ou, em alguns casos, planilhas simples. Essa realidade é bem comum em pequenas comunidades e iniciativas solidárias, o que acaba tornando o processo desorganizado, com risco de informações pessoais importantes serem perdidas e dificuldade em dar continuidade às ações de forma eficiente e clara.

Diante disso, percebemos uma oportunidade de contribuir com algo que vá além da teoria: criar uma solução digital acessível, que possa ajudar essas iniciativas a centralizarem suas informações, organizar seus eventos solidários e os divulgar com mais facilidade. Nosso objetivo é facilitar o trabalho de quem já faz muito com poucos recursos e fortalecer o impacto social dessas ações voltadas às comunidades mais vulneráveis.

1.3. Mapa do Site

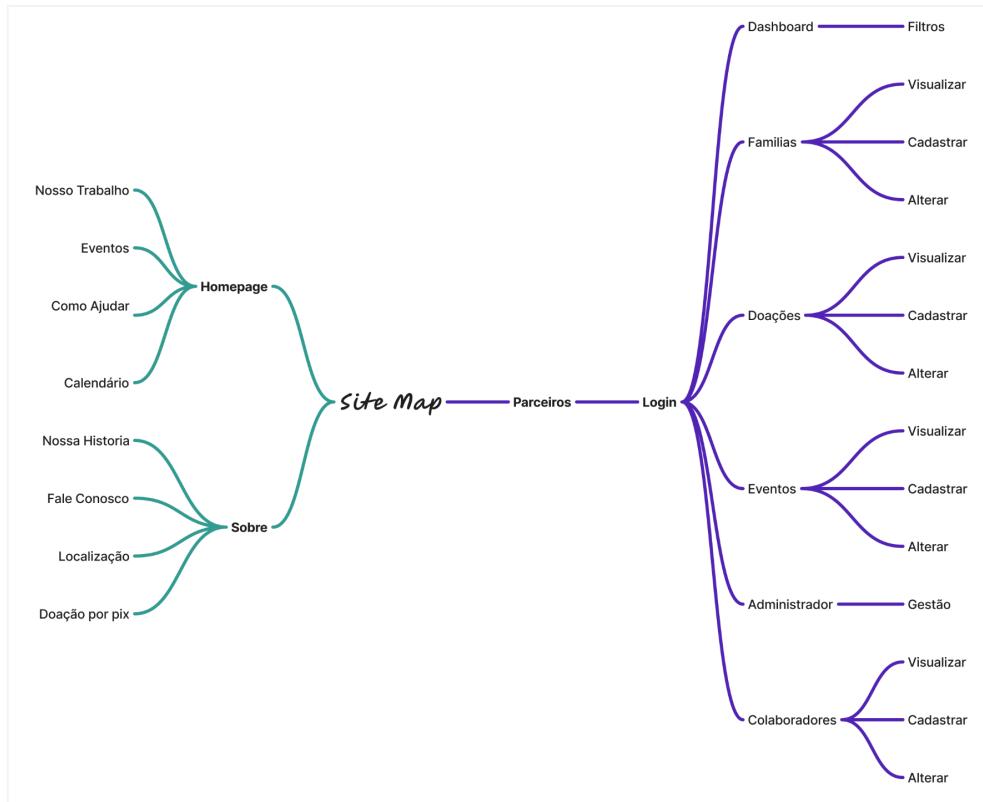


Figura 1 - Mapa do site

Fonte: Autoria Própria.

1.4. Logomarca



Figura 2 - Logomarca

Fonte: Autoria Própria.

2. REQUISITOS DO PROJETO

2.1. Levantamento de Requisitos

O levantamento de requisitos foi realizado utilizando as seguintes técnicas:

1. Entrevista com Possível Usuário

Realizamos entrevista qualitativa com a organizadora de eventos comunitários, membro da comunidade local, em particular a nossa parceira deste projeto. A entrevista buscou entender as dificuldades enfrentadas atualmente para divulgar eventos, gerenciar doações e mobilizar a comunidade.

2. Pesquisa de Mercado

Foi realizada uma análise de mercado para identificar plataformas semelhantes que oferecem funcionalidades de gestão de eventos ou sistemas de doação. O objetivo foi mapear as melhores práticas e identificar lacunas que o projeto pode preencher.

3. Análise de Casos de Uso Existentes

Ferramentas existentes como **GoFundMe**, **Kickstarter** e plataformas regionais de eventos comunitários foram analisadas. Cada uma delas serviu de referência para definir requisitos funcionais e não funcionais específicos, com base no que funciona bem e nas limitações observadas.

Plataformas Analisadas e Funcionalidades Relevantes

- **GoFundMe**¹:

Sistema transparente de doações e rastreamento de metas financeiras. Inspirou a funcionalidade de acompanhamento de arrecadações por meio de painéis interativos.

- **Eventbrite**²:

Gerenciamento de eventos, incluindo registro de participantes e

¹ Conforme disponível em: <<https://www.gofundme.com/>>. Acesso em: 21 Mar. 2025.

² Conforme disponível em: <<https://www.eventbrite.com/>>. Acesso em: 21 Mar. 2025.

lembretes automatizados. Adaptamos o modelo para incluir eventos sem fins lucrativos e maior detalhamento de impacto social.

- **Vakinha³ (Brasil):**

Interface simplificada para doações. Trouxe insights para desenvolver um sistema amigável e acessível, mesmo para usuários com pouca experiência digital.

- **Plataformas Comunitárias Locais:**

Aplicativos e sites regionais com foco em eventos locais mostraram a importância de integração com redes sociais e sistemas de geolocalização para aumentar a visibilidade das iniciativas.

³ Conforme disponível em: <<https://www.vakinha.com.br/>>. Acesso em: 21 Mar. 2025.

2.2. Requisitos Funcionais

Tabela 1 – Requisitos funcionais

REQUISITOS FUNCIONAIS	
RF01	CADASTRO DE VOLUNTÁRIOS : o administrador do sistema deverá ter a possibilidade de cadastrar outros usuários no sistema
RF02	O administrador deve ser cadastrado pelo time técnico no momento da entrega do software, pois não terá área pública de cadastro.
RF03	GESTÃO DE DOAÇÕES: área para registrar e acompanhar doações recebidas e alocar recursos conforme necessário.
RF04	GESTÃO DE EVENTOS: área para registrar, alterar, vincular doações e acompanhar os eventos que o projeto social está organizando, na área não logada serão exibidos os últimos eventos realizados e os próximos eventos do calendário.
RF05	GESTÃO DE FAMÍLIAS: área para registrar, alterar e visualizar as famílias para agrupar e vincular as pessoas ajudadas.

2.3. Diagrama de Caso de Uso

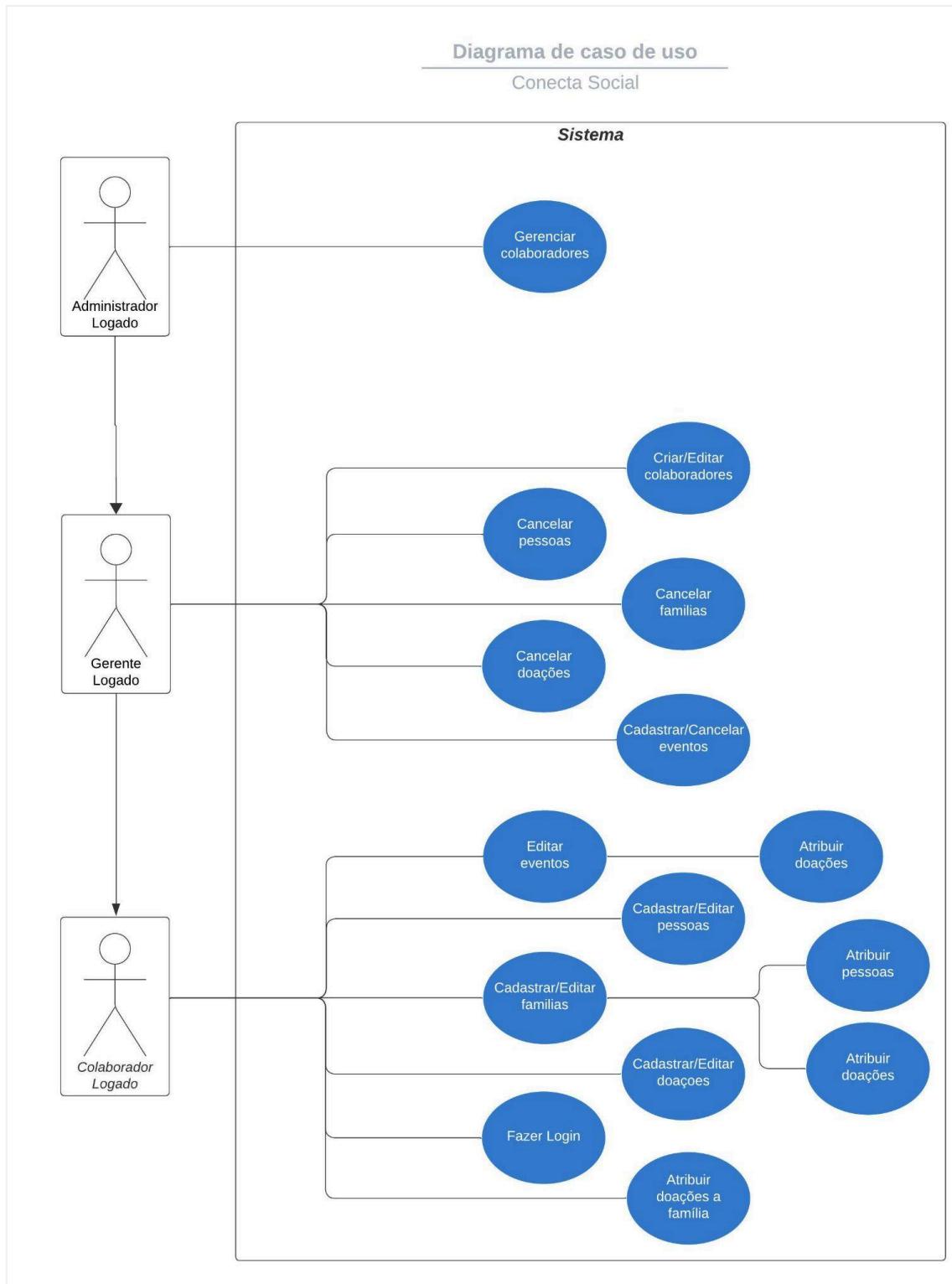


Figura 3 - Diagrama de caso de uso 1

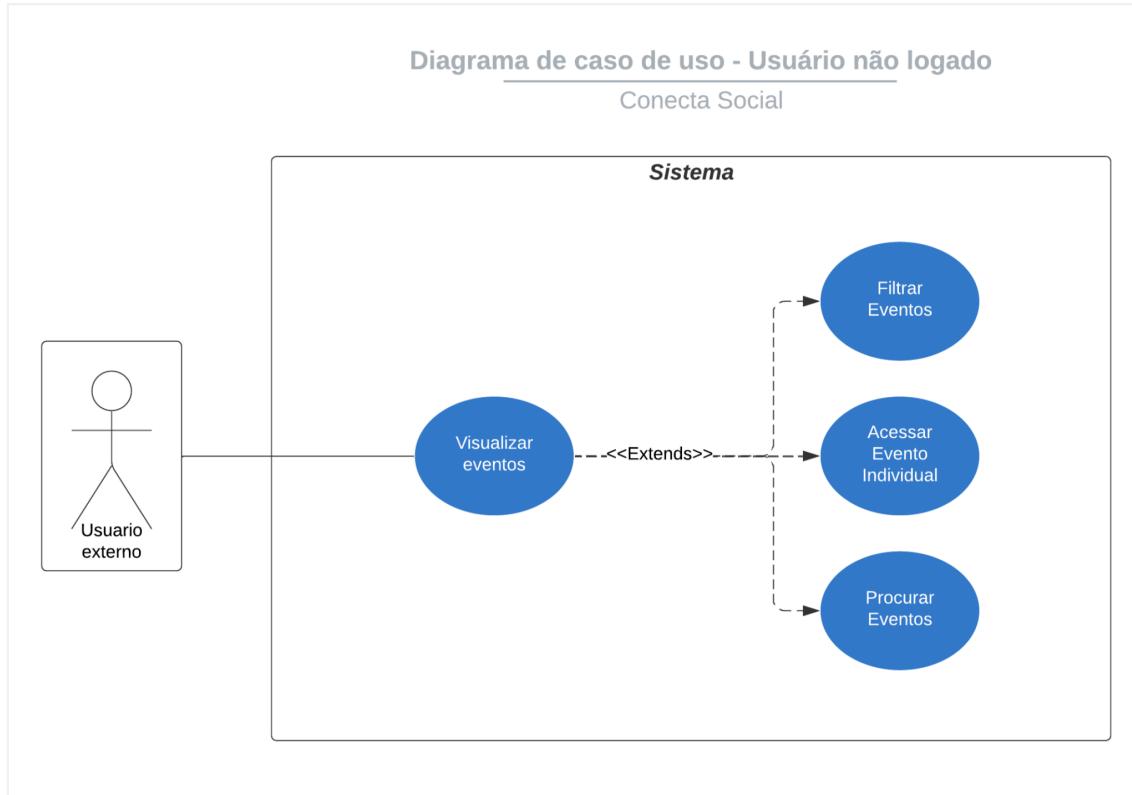


Figura 4 - Diagrama de caso de uso 2

2.4. Descrição do Caso de Uso

Gerenciar Usuário

Tabela 2 - Caso de uso "Gerenciar Voluntários "

Caso de uso	Gerenciar Voluntários
Autor Principal	Administrador
Autor Secundário	-
Pré-condições	Administrador possuir acesso ao sistema
Pós-condições	Voluntário cadastrado, alterado ou excluído do sistema
Fluxo Principal	
Ações do Ator Ações do Sistema	
CADASTRAR	
Acessar Tela de Cadastrar Voluntários	
	Sistema exibe o formulário
Gerente preenche as informações do voluntário	
	Sistema valida se todos campos estão preenchidos, retorna mensagem de sucesso e salva registro.
ALTERAR	
Acessar Tela de Voluntários	
	Sistema exibe lista de voluntários
Seleciona o voluntário que deseja alterar	
	O Sistema exibe as informações do voluntário no formato de formulário bloqueado para edição.
Seleciona a opção de editar	
	O Sistema habilita os campos para edição e exibe o botão de salvar.
Administrador edita as informações desejadas e seleciona para salvar.	
	O Sistema valida as informações, retorna mensagem de sucesso e salva o registro.
RESET DE SENHA	
Acessar Tela de Voluntários	

	Sistema exibe lista de voluntários
Seleciona um registro e clica em resetar senha	
	Sistema exibe tela de confirmação
Administrador confirma o <i>reset</i> de senha	
	O Sistema faz o <i>reset</i> da senha do voluntário e exibe a senha nova para o administrador.

EXCLUIR	
Acessar Tela de Voluntários	
	Sistema exibe lista de voluntários
Selecionar o registro desejado e clicar em excluir	
	Sistema exibe uma tela para confirmar a exclusão.
Gerente confirma a exclusão	
	Sistema cancela/oculta o registro e retorna mensagem de sucesso.

Fonte: Autoria própria

Criar/Editar Voluntários

Tabela 3 - Caso de uso "Criar/Editar Voluntários "

Caso de uso	Criar/Editar Voluntários	
Autor Principal	Gerente	
Autor Secundário	-	
Pré-condições	Gerente possuir acesso ao sistema	
Pós-condições	Voluntário cadastrado ou alterado no sistema	
Fluxo Principal		
Ações do Ator		Ações do Sistema
CADASTRAR		
Acessar Tela de Cadastrar Voluntários		
	Sistema exibe o formulário	
Gerente preenche as informações do voluntário		
	Sistema valida se todos campos estão preenchidos, retorna mensagem de sucesso e salva registro.	
ALTERAR		
Acessar Tela de Voluntários		
	Sistema exibe lista de voluntários	
Seleciona o voluntário que deseja alterar		
	O Sistema exibe as informações do voluntário no formato de formulário bloqueado para edição.	
Seleciona a opção de editar		
	O Sistema habilita os campos para edição e exibe o botão de salvar.	
Administrador edita as informações desejadas		
	O Sistema valida as informações, retorna mensagem de sucesso e salva o registro.	

Fonte: Autoria própria

Cancelar beneficiários

Tabela 4 - Caso de uso "Cancelar beneficiários"

Caso de uso	Cancelar beneficiários
Autor Principal	Gerente, Administrador
Autor Secundário	-
Pré-condições	Autor possuir acesso ao sistema, ter uma beneficiário cadastrada no sistema
Pós-condições	Pessoa com registro cancelado/oculto do sistema
Fluxo Principal	
Ações do Ator	Ações do Sistema
CANCELAR	
Acessar Tela de Beneficiários	Sistema exibe a lista de beneficiários
Seleciona o beneficiário que deseja “excluir” ou cancelar	Sistema exibe as informações de beneficiário no formato de formulário bloqueado
Seleciona a opção de excluir	O sistema exibe uma tela para confirmar a exclusão.
Autor confirma a exclusão de beneficiário	Sistema cancela/oculta o registro e retorna mensagem de sucesso.

Fonte: Autoria própria

Cancelar família

Tabela 5 - Caso de uso "Cancelar família"

Caso de uso	Cancelar família
Autor Principal	Gerente, Administrador
Autor Secundário	-
Pré-condições	Autor possuir acesso ao sistema, ter uma família cadastrada
Pós-condições	Família com registro cancelado/oculto do sistema
Fluxo Principal	
Ações do Ator	
Ações do Sistema	
CANCELAR	
Acessar Tela de Famílias cadastradas	
	Sistema exibe a lista de famílias
Seleciona a família que deseja “excluir” ou cancelar	
	O sistema exibe as informações da família no formato de formulário bloqueado para edição e os beneficiários pertencentes àquela família.
Seleciona a opção de excluir	
	O sistema exibe uma tela para confirmar a exclusão.
Autor confirma a exclusão da família	
	Sistema cancela/oculta o registro e retorna mensagem de sucesso.

Fonte: Autoria própria

Cancelar doações

Tabela 6 - Caso de uso "Cancelar doações"

Caso de uso	Cancelar doações
Autor Principal	Gerente, Administrador
Autor Secundário	-
Pré-condições	Autor possuir acesso ao sistema, doação estar registrada no sistema
Pós-condições	Doação cancelada/oculta no sistema
Fluxo Principal	
Ações do Ator	
Ações do Sistema	
CANCELAR	
Acessar Tela de doações cadastradas	
	Sistema exibe a lista de doações
Seleciona a doação que deseja “excluir” ou cancelar	
	O sistema exibe as informações da doação no formato de formulário bloqueado para edição.
Seleciona a opção de excluir	
	O sistema exibe uma tela para confirmar a exclusão.
Ator confirma a exclusão da doação	
	Sistema cancela/oculta o registro e retorna mensagem de sucesso.

Fonte: Autoria própria

Cadastrar/Cancelar Eventos

Tabela 7 - Caso de uso "Cadastrar/Cancelar Eventos"

Caso de uso	Cadastrar/Cancelar Eventos
Autor Principal	Gerente, Administrador
Autor Secundário	-
Pré-condições	Autor possuir acesso ao sistema
Pós-condições	Evento cadastrado ou alterado no sistema
Fluxo Principal	
Ações do Ator Ações do Sistema	
CADASTRAR	
Acessar Tela de Cadastrar Evento	
	Sistema exibe o formulário
Autor preenche as informações do Evento	
	Sistema valida se todos campos estão preenchidos, retorna mensagem de sucesso e salva registro.
CANCELAR	
Acessar Tela de eventos cadastrados	
	Sistema exibe a lista de doações
Seleciona o evento que deseja “excluir” ou cancelar	
	O sistema exibe as informações do evento no formato de formulário bloqueado para edição.
Seleciona a opção de excluir	
	O sistema exibe uma tela para confirmar a exclusão.
Autor confirma a exclusão do evento	
	Sistema cancela/oculta o registro e retorna mensagem de sucesso.

Fonte: Autoria própria

Editar Eventos

Tabela 8 - Caso de uso "Editar Eventos"

Caso de uso	Editar Eventos	
Autor Principal	Voluntário, Gerente, Administrador	
Autor Secundário	-	
Pré-condições	Ator possuir acesso ao sistema, ter um evento cadastrado	
Pós-condições	Evento alterado no sistema	
Fluxo Principal		
Ações do Ator		Ações do Sistema
ALTERAR		
Acessar Tela de Eventos		
	O Sistema exibe a lista de eventos	
Seleciona a doação que deseja alterar		
	O Sistema exibe as informações do evento no formato de formulário bloqueado para edição.	
Seleciona a opção de editar		
	O Sistema habilita os campos para edição e exibe o botão de salvar.	
O ator edita as informações desejadas e clicar para salvar.		
	O Sistema valida as informações, retorna mensagem de sucesso e salva a alteração.	

Fonte: Autoria própria

Cadastrar/Alterar Beneficiários

Tabela 9 - Caso de uso "Cadastrar/Alterar Beneficiários"

Caso de uso	Cadastrar/Alterar Beneficiários
Autor Principal	Voluntário, Gerente, Administrador
Autor Secundário	-
Pré-condições	Autor possuir acesso ao sistema
Pós-condições	Pessoa cadastrada ou alterada no sistema
Fluxo Principal	
Ações do Ator	Ações do Sistema
CADASTRAR	
Acessar Tela de Cadastrar Pessoa	Sistema exibe o formulário
Autor preenche as informações do beneficiário	Sistema valida se todos campos estão preenchidos, retorna mensagem de sucesso e salva registro.
ALTERAR	
Acessar Tela de Beneficiários	O Sistema exibe a lista de beneficiários registradas
Seleciona o beneficiário que deseja alterar	O Sistema exibe as informações de beneficiário no formato de formulário bloqueado para edição.
Seleciona a opção de editar	O Sistema habilita os campos para edição e exibe o botão de salvar.
O autor edita as informações desejadas e clica para salvar.	O Sistema valida as informações, retorna mensagem de sucesso e salva a alteração.

Fonte: Autoria própria

Cadastrar/Alterar Família

Tabela 10 - Caso de uso "Cadastrar/Alterar Família"

Caso de uso	Cadastrar/Alterar Família
Autor Principal	Voluntário, Gerente, Administrador
Autor Secundário	-
Pré-condições	Autor possuir acesso ao sistema
Pós-condições	Família cadastrada ou alterada no sistema
Fluxo Principal	
Ações do Ator	Ações do Sistema
CADASTRAR	
Acessar Tela de Cadastrar Família	
	Sistema exibe o formulário
Autor preenche as informações da família e seleciona os beneficiários que farão parte.	
	Sistema valida se todos campos estão preenchidos, retorna mensagem de sucesso e salva registro.
ALTERAR	
Acessar Tela de Família	
	O Sistema exibe a lista de famílias registradas
Seleciona a família que deseja alterar	
	O Sistema exibe as informações da família no formato de formulário bloqueado para edição.
Seleciona a opção de editar	
	O Sistema habilita os campos para edição e exibe o botão de salvar.
O autor edita as informações desejadas e clicar para salvar.	
	O Sistema valida as informações, retorna mensagem de sucesso e salva a alteração.

Fonte: Autoria própria

Cadastrar/Alterar Doações

Tabela 11 - Caso de uso "Cadastrar/Alterar Doações"

Caso de uso	Cadastrar/Alterar Doações
Autor Principal	Voluntário, Gerente, Administrador
Autor Secundário	-
Pré-condições	Autor possuir acesso ao sistema
Pós-condições	Doação cadastrada ou alterada no sistema
Fluxo Principal	
Ações do Ator	Ações do Sistema
CADASTRAR	
Acessar Tela de Cadastrar Doação	Sistema exibe o formulário
Autor preenche as informações da família e seleciona os beneficiários que farão parte.	Sistema valida se todos campos estão preenchidos, retorna mensagem de sucesso e salva registro.
ALTERAR	
Acessar Tela de Doações	O Sistema exibe a lista de doações registradas
Seleciona a doação que deseja alterar	O Sistema exibe as informações da doação no formato de formulário bloqueado para edição.
Seleciona a opção de editar	O Sistema habilita os campos para edição e exibe o botão de salvar.
O autor edita as informações desejadas e clicar para salvar.	O Sistema valida as informações, retorna mensagem de sucesso e salva a alteração e atualiza a quantidade atual da doação se necessário.

Fonte: Autoria própria

Fazer Login

Tabela 12 - Caso de uso "Fazer Login"

Caso de uso	Fazer Login	
Autor Principal	Voluntário, Gerente, Administrador	
Autor Secundário	-	
Pré-condições	-	
Pós-condições	Usuário consegue acessar o sistema	
Fluxo Principal		
Ações do Ator	Ações do Sistema	
LOGAR		
Acessa a tela de login		
	O sistema exibe os campos de login para preencher	
Insere as credenciais de login (usuário e e-mail)		
	O sistema faz a autenticação das informações.	

Fonte: Autoria própria

Atribuir doações a família

Tabela 13 - Caso de uso "Atribuir doações a família"

Caso de uso	Atribuir doações a família	
Autor Principal	Voluntário, Gerente, Administrador	
Autor Secundário	-	
Pré-condições	Ter família e doação cadastradas previamente no sistema	
Pós-condições	O sistema registra a atribuição da doação para a família	
Fluxo Principal		
Ações do Ator	Ações do Sistema	
ATRIBUIR DOAÇÃO		
Usuário acessa a tela de informações da doação ou informações da família.		
	O sistema exibe os dados da doação ou da família.	
Usuário seleciona a opção de atribuir doação		
	O sistema abre o formulário de preenchimento para atribuição de doação e família.	
O usuário preenche as informações para atribuir a doação e seleciona para salvar.		
	O sistema exibe a mensagem de confirmação e salva o registro	

Fonte: Autoria própria

Visualizar eventos

Tabela 14 - Caso de uso "Visualizar eventos"

Caso de uso	Visualizar eventos	
Autor Principal	Usuário externo	
Autor Secundário	-	
Pré-condições	Ter eventos cadastrados no sistema	
Pós-condições	O ator visualiza os eventos públicos	
Fluxo Principal		
Ações do Ator		Ações do Sistema
VISUALIZAR EVENTOS		
Usuário externo acessa home do site do projeto social, filtra eventos, acessa evento Individual e procura eventos.		
	O site exibe as informações sobre o projeto e os eventos passados e futuros.	

Fonte: Autoria própria

2.5. Requisitos Não Funcionais

Tabela 15 – Requisitos não funcionais

REQUISITOS NÃO FUNCIONAIS	
RNF01	A interface deverá ter layout responsivo e navegação intuitiva.
RNF02	Os usuários deverão se cadastrar na aplicação, e o login deverá ser feito utilizando um e-mail e senha. pelo ‘bcrypt’ criptografar senhas e jwt-passport.
RNF03	O sistema deve processar pelo menos 10 requisições por minuto.
RNF04	o sistema deverá ter alta disponibilidade, 99% do tempo
RNF05	O sistema deverá seguir a Lei Geral de Proteção de Dados (LGPD).
RNF06	O produto deve ser web responsivo
RNF07	O sistema deverá se comunicar com o banco de dados MySQL.
RNF08	O sistema deverá ser desenvolvido utilizando a linguagem Javascript.

3. PROJETO DO SOFTWARE

3.1. Tecnologias Utilizadas

Para o desenvolvimento do Front-end, utilizaremos React⁴ com Next.js⁵, juntamente com TailwindCSS⁶ para a estilização.

No Back-end, serão empregados Node.js⁷ e Express.js⁸.

Tanto no Front-end quanto no Back-end, usaremos JavaScript⁹ como linguagem principal, com TypeScript¹⁰ como superconjunto para tipagem e estruturação de código.

Ferramentas utilizadas: Figma¹¹ para criar o protótipo do projeto, Brandmark.io¹² para criar a logomarca do time, Lucidchart¹³ para criar os diagramas e modelos, Phosphor Icon¹⁴ para utilizar os ícones do projeto.

⁴ Conforme disponível em: <<https://react.dev>>. Acesso em: 10 mai. 2025.

⁵ Conforme disponível em: <<https://nextjs.org>>. Acesso em: 10 mai. 2025.

⁶ Conforme disponível em: <<https://tailwindcss.com>>. Acesso em: 10 mai. 2025.

⁷ Conforme disponível em: <<https://nodejs.org/pt>>. Acesso em: 10 mai. 2025.

⁸ Conforme disponível em: <<https://expressjs.com>>. Acesso em: 10 mai. 2025.

⁹ Conforme disponível em: <<https://developer.mozilla.org/pt-BR>>. Acesso em: 10 mai. 2025.

¹⁰ Conforme disponível em: <<https://www.javascript.com>>. Acesso em: 10 mai. 2025.

¹¹ Conforme disponível em: <<https://www.figma.com>>. Acesso em: 10 mai. 2025.

¹² Conforme disponível em: <<https://www.brandmark.io>>. Acesso em: 10 mai. 2025.

¹³ Conforme disponível em: <<https://www.lucidchart.com>>. Acesso em: 10 mai. 2025.

¹⁴ Conforme disponível em: <<https://phosphoricons.com>>. Acesso em: 10 mai. 2025.

3.2. Modelo de dados

3.2.1 Modelo Conceitual

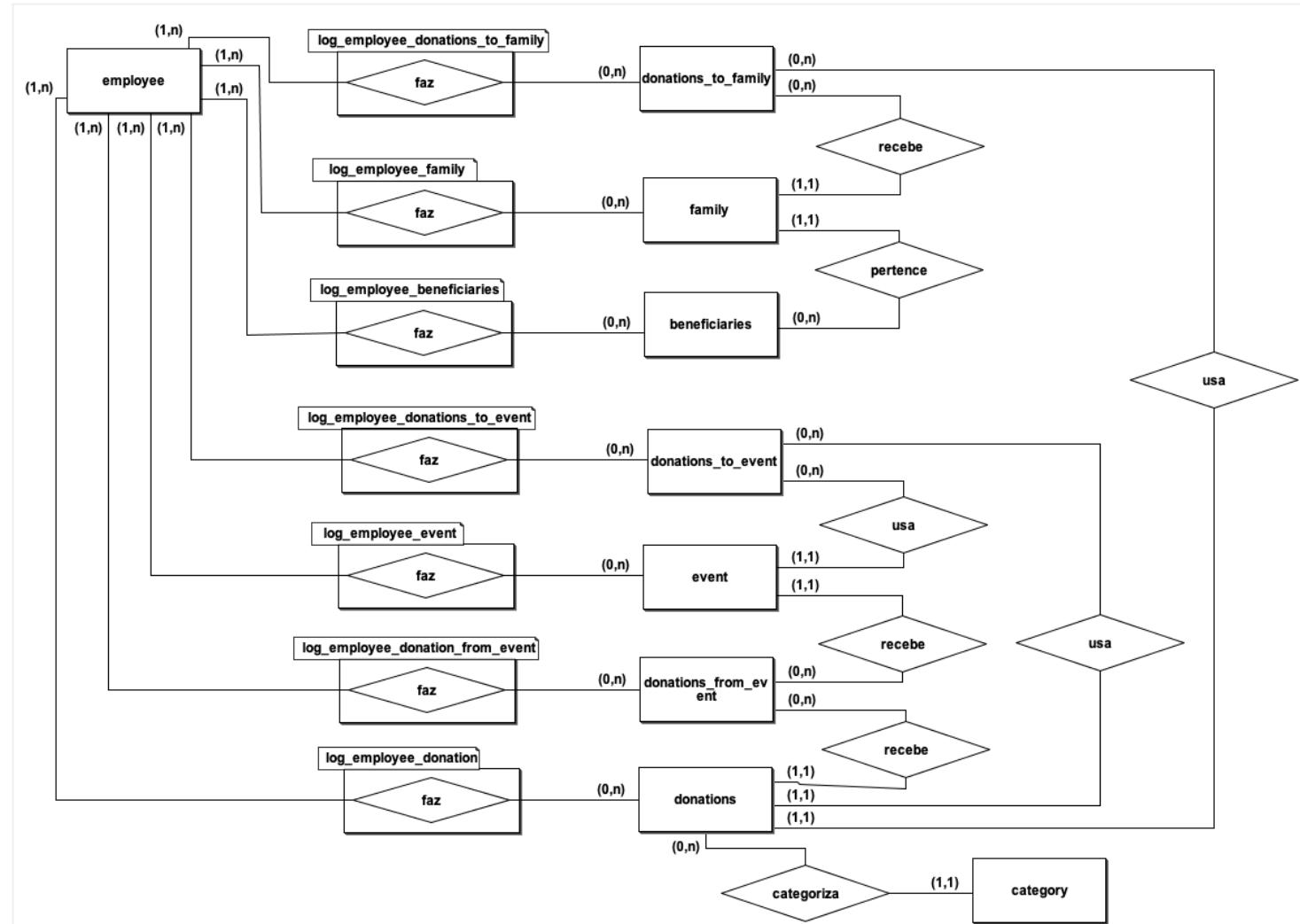


Figura 5 - Diagrama conceitual

3.2.2 Modelo Lógico

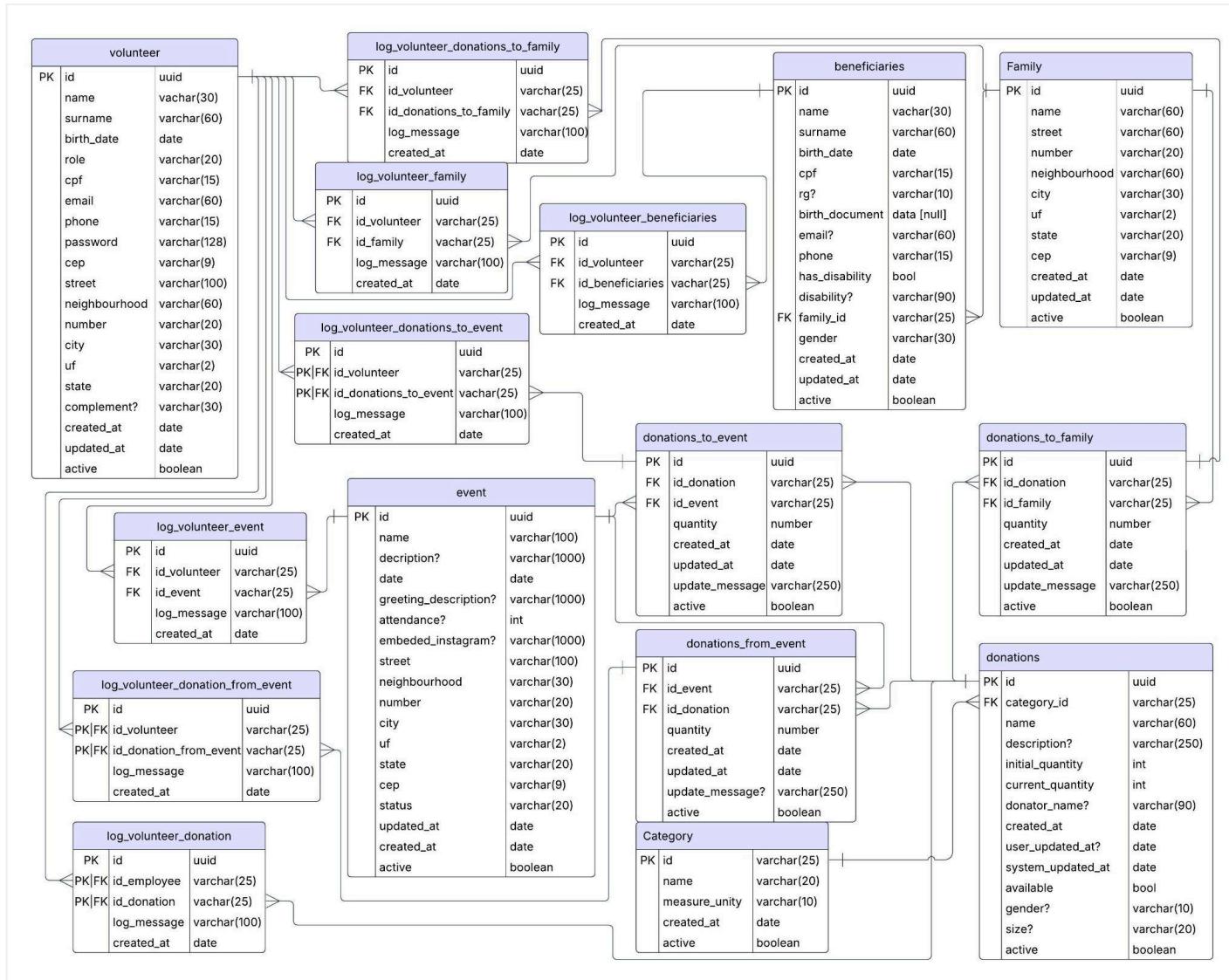


Figura 6 - Diagrama de lógico

3.3. Diagrama de Classe



Figura 7 - Diagrama de classe

3.4. Diagrama de Atividades

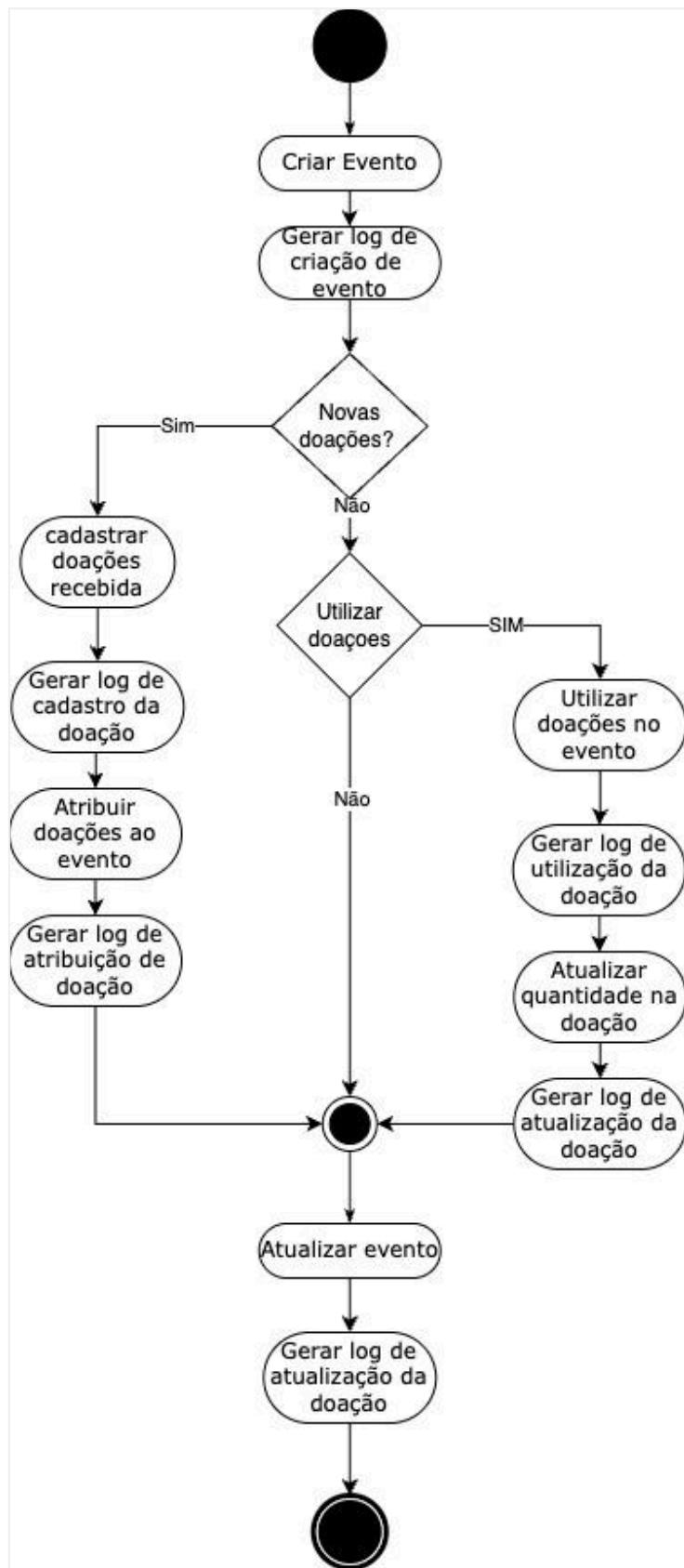


Figura 8 - Diagrama de atividade

3.5. Diagrama de Sequência

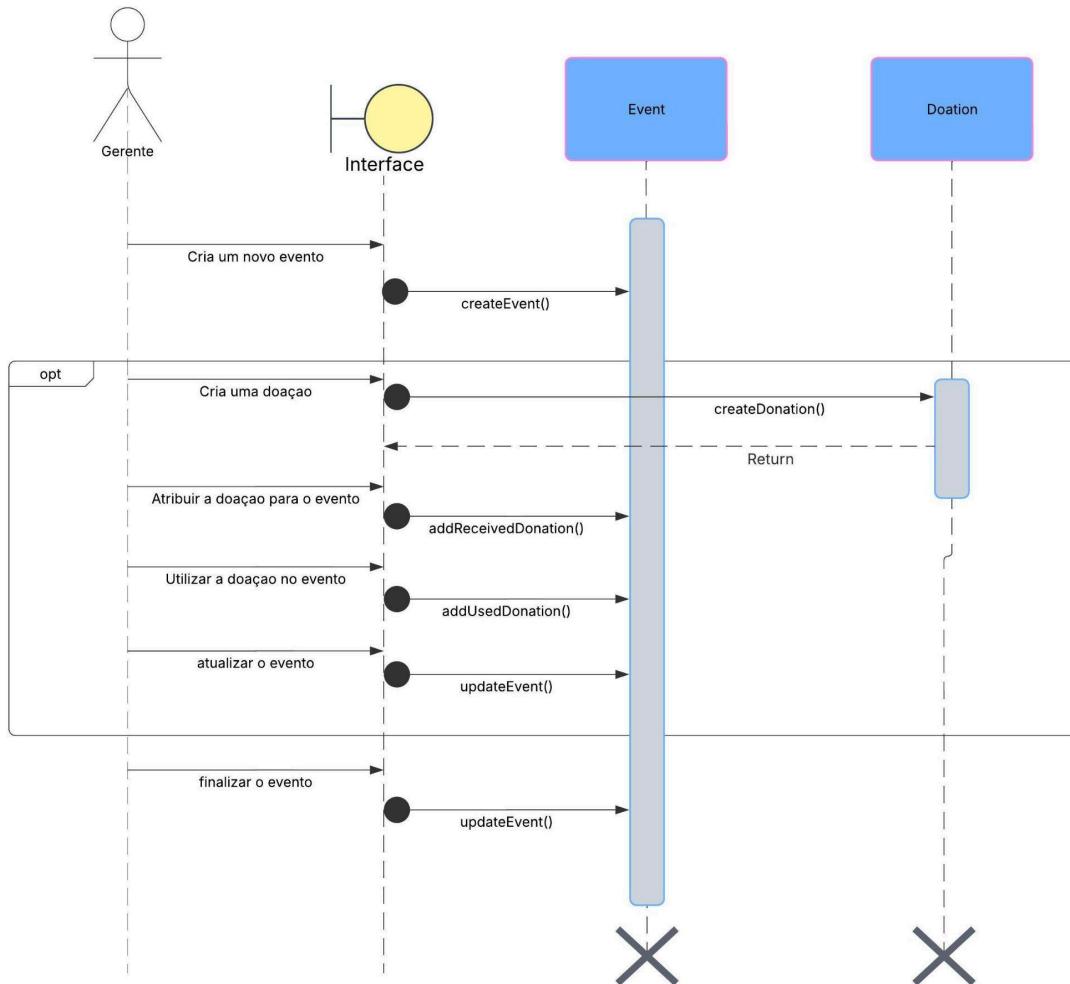


Figura 9 - Diagrama de sequência

3.6. Interfaces com o usuário

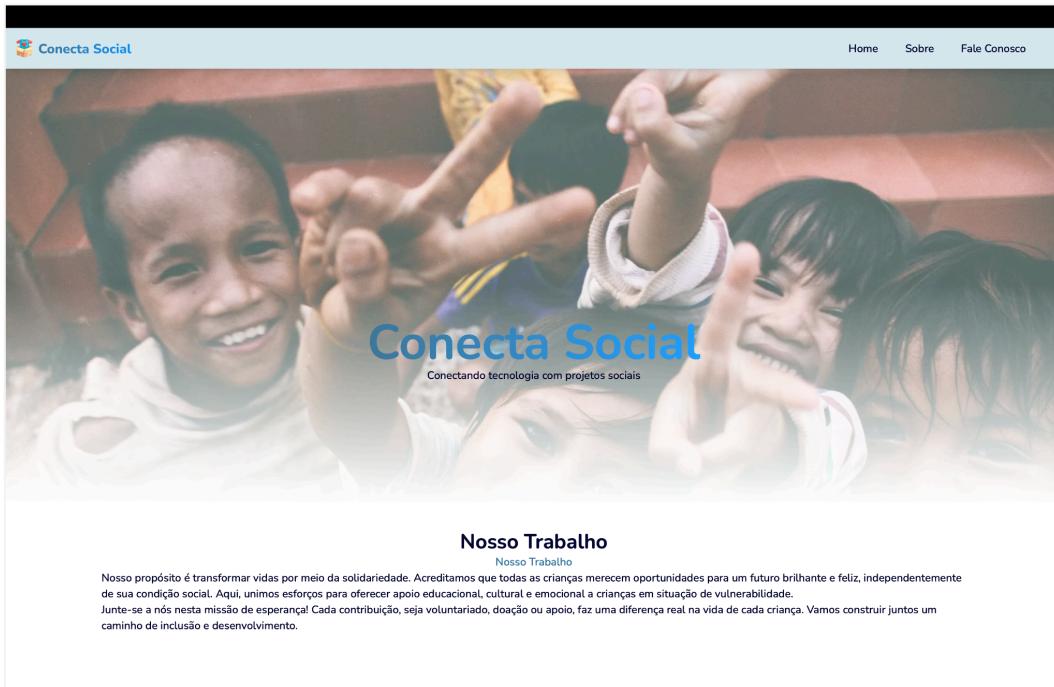


Figura 10 - Interface - Home



Figura 11 - Interface - Sobre

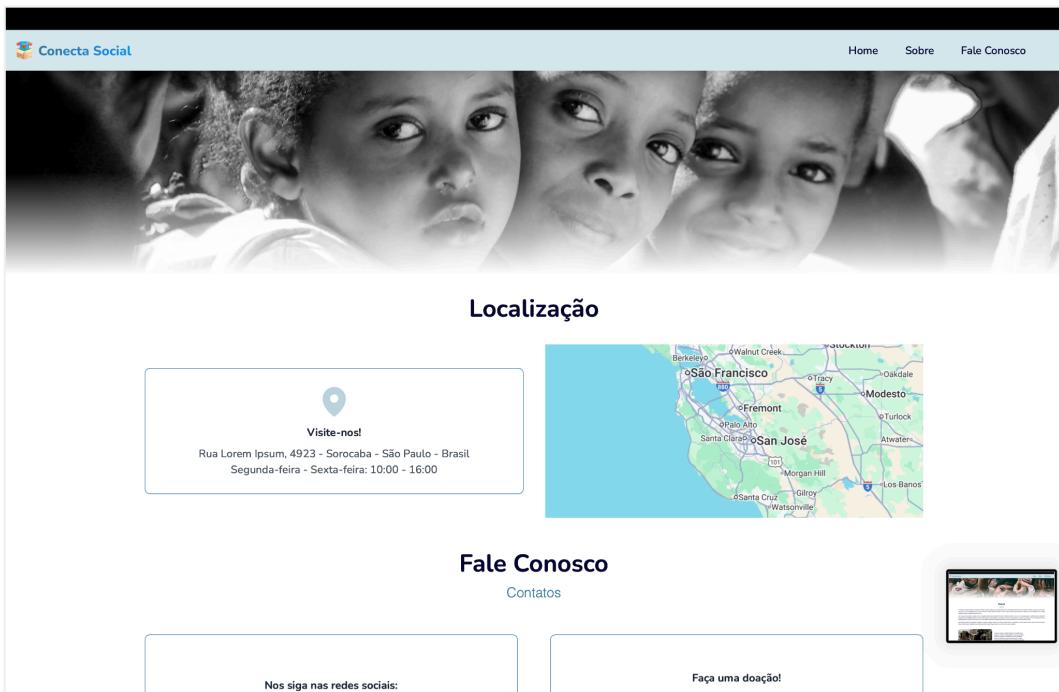


Figura 12 - Interface - Fale conosco

Figura 13 - Interface - CRUD

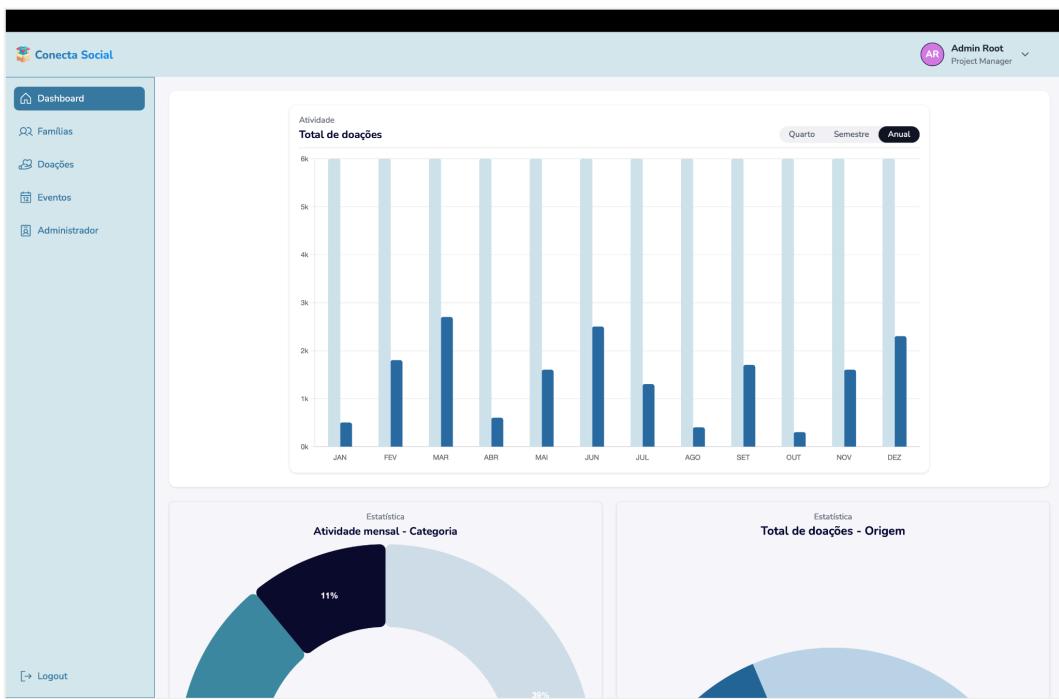


Figura 14 - Interface - Dashboard 1

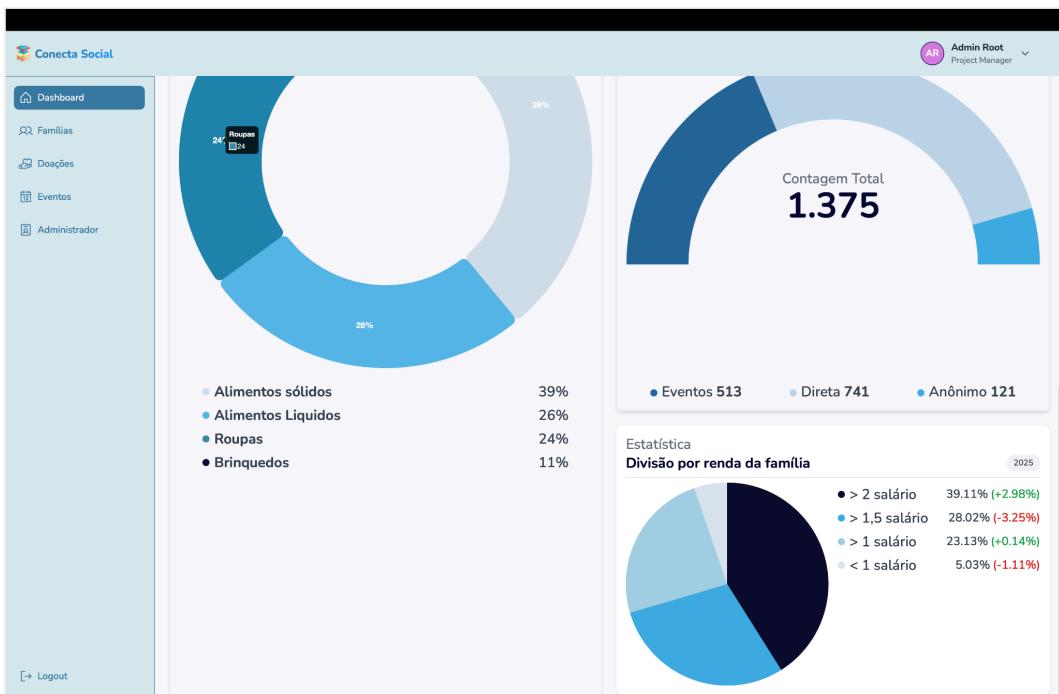


Figura 15 - Interface - Dashboard 2

4. ESTRATÉGIA DE TESTES

A estratégia de teste visa garantir que a plataforma Conecta Social atenda aos requisitos funcionais e não funcionais definidos anteriormente, promovendo um sistema confiável, seguro e de fácil usabilidade para seus usuários. A abordagem adotada será baseada em boas práticas da engenharia de software apresentadas na norma ISO/IEC 25010 e nas recomendações de Pressman para WebApps.

Devido ao atual nível do projeto não será aplicado testes como Unidade e Integração.

4.1. Níveis de Testes aplicados

4.1.1. Teste de sistema

Avaliar o comportamento do sistema como um todo, confrontando os resultados esperados com os reais a partir dos requisitos do projeto.

4.1.2. Teste de aceitação

Verificar a conformidade do sistema com as expectativas dos usuários finais (usuários e administradores), focando principalmente na experiência e funcionalidades críticas.

4.1.3. Teste de regressão

Será aplicado após ajustes ou atualizações, assegurando que novas implementações não afetem funcionalidades já validadas.

4.2. Tipos de Testes

4.2.1. Funcionalidade

Confirmar se funcionalidades como cadastro, login, publicação de eventos e doações funcionam corretamente.

4.2.2. Interface

Verificação dos elementos visuais e interativos da interface (usabilidade, responsividade e estética).

4.2.3. Navegação

Testar a coerência dos fluxos de navegação entre páginas (ex: do menu inicial até a visualização de um evento).

4.2.4. Usabilidade

Avaliação da facilidade de uso por diferentes perfis de usuário, especialmente públicos com menor familiaridade digital.

4.2.5. Desempenho

Análise de tempo de resposta e estabilidade da aplicação em situações normais e de pico (ex: eventos com grande número de acessos simultâneos).

4.2.6. Segurança

Testes para validar proteção contra acesso indevido, como tentativa de SQL Injection e falhas de autenticação/autorização.

5. PROJETO

Os arquivos do projeto estão disponíveis no repositório do GitHub nos links abaixo. Os arquivos de documentação estão no seguinte link: [link](#). Neste [link](#), você encontrará o arquivo da **documentação em PDF**, a **pasta zipada com os códigos-fonte do sistema**, o **vídeo de apresentação do produto**. O **script do banco de dados** do projeto está no apêndice.

5.1. Link do repositório github

[Front-end](#)

[Back-end / API](#)

6. CONCLUSÃO

O desenvolvimento do projeto Conecta Social representou mais do que a aplicação de conteúdos vistos em sala de aula, foi uma oportunidade de concretizar a documentação do nosso projeto e enxergar como a tecnologia pode ser aliada na promoção do bem-estar social. Ao longo da disciplina de Engenharia de Software 2, enfrentamos desafios técnicos e organizacionais.

Cada etapa do projeto, desde validar os requisitos até os testes, contribuiu para consolidar nosso entendimento sobre a importância de planejar, modelar e desenvolver sistemas com propósito e responsabilidade. Mais do que entregar uma solução funcional, buscamos criar algo que possa, de fato, fazer diferença na vida de pessoas e comunidades que mais precisam de visibilidade e apoio.

O Conecta Social é o reflexo do nosso comprometimento em aliar conhecimento técnico que temos com o comprometimento com a sociedade. Encerramos esta etapa conscientes de que teremos mais desafios, não se limitando ao código: vai envolver empatia, colaboração e um olhar atento para os problemas do mundo real.

7. REFERÊNCIAS

As referências sem autoria, representadas apenas por uma URL foram apresentadas ao longo do texto, em notas de rodapé.

SOMMERVILLE, Ian. **Engenharia de Software**. 8^a ed. Pearson, 2007

Apêndice

Protótipo funcional do Figma

Script - Banco de dados

```
-- Criação do banco de dados
CREATE DATABASE conecta_social;
GO

-- Seleção do banco para uso
USE conecta_social;
GO

-- Tabela de categorias de doações
CREATE TABLE Category
(
    id VARCHAR(25) PRIMARY KEY,
    name VARCHAR(60),
    measure_unity VARCHAR(20),
    created_at DATE,
    active BIT
);

-- Tabela de doações
CREATE TABLE donations
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    category_id VARCHAR(25),
    name VARCHAR(60),
```

```
description VARCHAR(1000) NULL,  
initial_quantity INT,  
current_quantity INT,  
donator_name VARCHAR(90) NULL,  
created_at DATE,  
user_updated_at DATE NULL,  
system_updated_at DATE NULL,  
available BIT,  
gender VARCHAR(10) NULL,  
size VARCHAR(20) NULL,  
active BIT,  
FOREIGN KEY (category_id) REFERENCES Category(id)  
) ;  
  
-- Tabela de famílias  
CREATE TABLE Family  
(  
    id UNIQUEIDENTIFIER PRIMARY KEY,  
    name VARCHAR(60),  
    street VARCHAR(60),  
    number VARCHAR(20),  
    neighbourhood VARCHAR(60),  
    city VARCHAR(30),  
    uf VARCHAR(2),  
    state VARCHAR(20),  
    cep VARCHAR(9),  
    updated_at DATE,  
    active BIT  
) ;  
  
-- Tabela de beneficiários
```

```
CREATE TABLE beneficiaries
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    name VARCHAR(30),
    surname VARCHAR(60),
    birth_date DATE,
    cpf VARCHAR(15),
    rg VARCHAR(20) NULL,
    birth_document DATE,
    email VARCHAR(60) NULL,
    phone VARCHAR(15),
    has_disability BIT,
    disability VARCHAR(90) NULL,
    family_id VARCHAR(25),
    gender VARCHAR(30),
    created_at DATE,
    updated_at DATE,
    active BIT,
    FOREIGN KEY (family_id) REFERENCES Family(id)
);

-- Tabela de funcionários
CREATE TABLE volunteer
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    name VARCHAR(30),
    surname VARCHAR(60),
    birth_date DATE,
    role VARCHAR(20),
    cpf VARCHAR(15),
    email VARCHAR(60),
```

```
phone VARCHAR(15),
password VARCHAR(128),
cep VARCHAR(9),
street VARCHAR(60),
neighbourhood VARCHAR(60),
number VARCHAR(30),
city VARCHAR(30),
uf VARCHAR(2),
state VARCHAR(20),
complement VARCHAR(30) NULL,
created_at DATE,
updated_at DATE,
active BIT
);

-- Tabela de eventos
CREATE TABLE event
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    name VARCHAR(60),
    description VARCHAR(1000) NULL,
    date DATE,
    greeting_description VARCHAR(1000) NULL,
    attendance INT NULL,
    embeded_instagram VARCHAR(1000) NULL,
    street VARCHAR(60),
    neighbourhood VARCHAR(30),
    number VARCHAR(20),
    city VARCHAR(30),
    state VARCHAR(20),
    cep VARCHAR(9),
```

```
status VARCHAR(20),  
updated_at DATE,  
created_at DATE,  
active BIT  
) ;  
  
-- Tabela de doações para famílias  
CREATE TABLE donations_to_family  
(  
    id UNIQUEIDENTIFIER PRIMARY KEY,  
    id_donation VARCHAR(25),  
    id_family VARCHAR(25),  
    quantity INT,  
    created_at DATE,  
    updated_at DATE,  
    update_message VARCHAR(250),  
    FOREIGN KEY (id_donation) REFERENCES donations(id),  
    FOREIGN KEY (id_family) REFERENCES Family(id)  
) ;  
  
-- Tabela de doações para eventos  
CREATE TABLE donations_to_event  
(  
    id UNIQUEIDENTIFIER PRIMARY KEY,  
    id_donation VARCHAR(25),  
    id_event VARCHAR(25),  
    quantity INT,  
    created_at DATE,  
    updated_at DATE,  
    update_message VARCHAR(250),  
    FOREIGN KEY (id_donation) REFERENCES donations(id),
```

```
    FOREIGN KEY (id_event) REFERENCES event(id)
);

-- Tabela de doações recolhidas de eventos
CREATE TABLE donations_from_event
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    id_event VARCHAR(25),
    id_donation VARCHAR(25),
    quantity INT,
    created_at DATE,
    updated_at DATE,
    update_message VARCHAR(250) NULL,
    active BIT,
    FOREIGN KEY (id_event) REFERENCES event(id),
    FOREIGN KEY (id_donation) REFERENCES donations(id)
);

-- Logs de funcionário com famílias
CREATE TABLE log_volunteer_family
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    id_volunteer VARCHAR(25),
    id_family VARCHAR(25),
    log_message VARCHAR(100),
    created_at DATE,
    FOREIGN KEY (id_volunteer) REFERENCES volunteer(id),
    FOREIGN KEY (id_family) REFERENCES Family(id)
);

-- Logs de funcionário com beneficiários
```

```

CREATE TABLE log_volunteer_beneficiaries
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    id_volunteer VARCHAR(25),
    id_beneficiaries VARCHAR(25),
    log_message VARCHAR(100),
    created_at DATE,
    FOREIGN KEY (id_volunteer) REFERENCES volunteer(id),
    FOREIGN KEY (id_beneficiaries) REFERENCES
beneficiaries(id)
);

-- Logs de funcionário com eventos
CREATE TABLE log_volunteer_event
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    id_volunteer VARCHAR(25),
    id_event VARCHAR(25),
    log_message VARCHAR(100),
    created_at DATE,
    FOREIGN KEY (id_volunteer) REFERENCES volunteer(id),
    FOREIGN KEY (id_event) REFERENCES event(id)
);

-- Logs de funcionário com doações
CREATE TABLE log_volunteer_donation
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    id_volunteer VARCHAR(25),
    id_donation VARCHAR(25),
    log_message VARCHAR(100),
)

```

```
created_at DATE,  
FOREIGN KEY (id_volunteer) REFERENCES volunteer(id),  
FOREIGN KEY (id_donation) REFERENCES donations(id)  
) ;  
  
-- Logs de funcionário com doações para famílias  
CREATE TABLE log_volunteer_donations_to_family  
(  
    id UNIQUEIDENTIFIER PRIMARY KEY,  
    id_volunteer VARCHAR(25),  
    id_donations_to_family VARCHAR(25),  
    log_message VARCHAR(100),  
    created_at DATE,  
    FOREIGN KEY (id_volunteer) REFERENCES volunteer(id),  
        FOREIGN KEY (id_donations_to_family) REFERENCES  
donations_to_family(id)  
) ;  
  
-- Logs de funcionário com doações para eventos  
CREATE TABLE log_volunteer_donations_to_event  
(  
    id UNIQUEIDENTIFIER PRIMARY KEY,  
    id_volunteer VARCHAR(25),  
    id_donations_to_event VARCHAR(25),  
    log_message VARCHAR(100),  
    created_at DATE,  
    FOREIGN KEY (id_volunteer) REFERENCES volunteer(id),  
        FOREIGN KEY (id_donations_to_event) REFERENCES  
donations_to_event(id)  
) ;
```

```
-- Logs de funcionário com doações recolhidas de evento
CREATE TABLE log_volunteer_donation_from_event
(
    id UNIQUEIDENTIFIER PRIMARY KEY,
    id_volunteer VARCHAR(25),
    id_donation_from_event VARCHAR(25),
    log_message VARCHAR(100),
    created_at DATE,
    FOREIGN KEY (id_volunteer) REFERENCES volunteer(id),
        FOREIGN KEY (id_donation_from_event) REFERENCES
donations_from_event(id)
);
```