**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS  
Bacharelado em Sistemas de Informação**

**RELATÓRIO TÉCNICO**

**Trabalho Integrado: Aplicações para Sustentabilidade**

**EDIFI-CAD: Sistema de Cadastro**

**Autores:**

**Jardel Nunes,**

**Luis Otávio,**

**Luiz Arthur,**

**Matheus Rógeres e**

**Stephanie Santos**

| Histórico de Revisões | | | |
| --- | --- | --- | --- |
| Data | **Autor** | **Descrição** | **Versão** |
| 21/08/2023 | Jardel Nunes | Incluir descrição da ONG | 1 |
| 25/09/2023 | Stephanie Santos | Levantamento dos Requisitos | 1.1 |
| 10/10/2023 | Jardel Nunes | Revisões do documento conforme observações da Sprint II | 1.2 |
| 10/10/2023 | Jardel Nunes | Inclusão dos diagramas C4 | 1.3 |
| 10/10/2023 | Luiz Otávio | Revisões na descrição do cliente | 1.4 |
| 07/11/2023 | Jardel Nunes | Inclusão dos links do projeto / diagrama de classe | 1.5 |
| 05/12/2023 | Stephanie Santos | Revisão Final | 2.0 |

**SUMÁRIO**

[**1 Apresentação 3**](#_heading=h.gjdgxs)

[1.1 Cliente 3](#_heading=h.30j0zll)

[1.2 Problema 3](#_heading=h.1fob9te)

[1.3 Objetivo do trabalho 4](#_heading=h.3znysh7)

[1.4 Definições e Abreviaturas 4](#_heading=h.2et92p0)

[**2 Requisitos 4**](#_heading=h.tyjcwt)

[2.1 Lista de Atores 4](#_heading=h.3dy6vkm)

[2.2 Lista de Funcionalidades 4](#_heading=h.1t3h5sf)

[2.3 Requisitos Funcionais 5](#_heading=h.4d34og8)

[2.4 Requisitos Não Funcionais 5](#_heading=h.17dp8vu)

[2.5 Descrição Resumida dos Casos de Uso 5](#_heading=h.3rdcrjn)

[2.6 Restrições Arquiteturais 9](#_heading=)

[2.7 Mecanismos Arquiteturais 9](#_heading=h.lnxbz9)

[**3 Modelagem e diagramas arquiteturais: (Modelo C4) 9**](#_heading=h.35nkun2)

[3.1 Nível 1: Diagrama de Contexto 10](#_heading=h.1ksv4uv)

[3.2 Nível 2: Diagrama de Contêiner 11](#_heading=h.2jxsxqh)

[3.3 Nível 3: Diagrama de Componentes 12](#_heading=h.z337ya)

[3.4 Nível 4: Código 13](#_heading=h.3j2qqm3)

[**4 Avaliação Crítica dos Resultados 13**](#_heading=h.4i7ojhp)

[**5 Conclusão 14**](#_heading=h.2xcytpi)

[**6 Referências 15**](#_heading=h.1ci93xb)

[**7 APÊNDICES 16**](#_heading=h.3whwml4)

# Apresentação

Neste documento, será apresentado o processo de criação de uma solução tecnológica para a ONG Edificando Vidas, que desempenha um papel vital na comunidade de Belo Horizonte, Minas Gerais. Nos últimos anos, a organização acumulou uma quantidade significativa de informações essenciais sobre seus beneficiários, dependentes, voluntários e atividades financeiras devido aos projetos de apoio a famílias carentes existentes que já proporcionou à ONG um total superior a 3.000 famílias cadastradas e uma média de 50 cestas distribuídas mensalmente. No entanto, esses dados estão dispersos em diversos formatos, desde cadastros em papel até mensagens de WhatsApp. Essa falta de centralização e de uma solução tecnológica adequada tem dificultado uma gestão eficaz da organização.

## Cliente

Fundada há mais de 17 anos por Marcos Naves e Ana Naves, a ONG Edificando Vidas surgiu a partir de uma experiência vivida pelos fundadores com um casal que enfrentava dificuldades e necessidades. Rapidamente, eles receberam apoio solidário de várias pessoas, o que os ajudou a superar esse momento desafiador. A partir dessa ajuda, surgiu o desejo de retribuir esse gesto para outras famílias que enfrentam situações semelhantes. Desde então, a organização tem se dedicado incansavelmente a melhorar a vida de indivíduos em situação de vulnerabilidade em Belo Horizonte, oferecendo projetos de assistência social, distribuição de cestas básicas, aulas de idiomas, entre outros serviços.

De acordo com as informações fornecidas no Instagram da ONG, mais de 500 moradores de rua tiveram suas vidas transformadas, sendo auxiliados a retornar para suas famílias e serem reintegrados na sociedade. Atualmente, a ONG está presente nas redes sociais, contando com uma conta no Instagram e um canal no YouTube. Além disso, a organização possui um site no Wix, que está temporariamente desativado. A proposta deste projeto é reativar o site no Wix e implementar uma aplicação para facilitar parte do gerenciamento das atividades da ONG.

## Problema

O problema central enfrentado pelo parceiro é a falta de uma solução tecnológica que centralize os dados cadastrais dispersos, permita uma análise sistemática dos dados e reduza o consumo excessivo de papéis. Com isso, e gerencie eficazmente seus dados, incluindo informações sobre beneficiários, voluntários, atividades financeiras e aulas ministradas.

As principais questões incluem:

- Descentralização de Dados: Atualmente, os registros cadastrais de beneficiários, dependentes, voluntários e controle da distribuição de cestas estão dispersos em papel, planilhas e mensagens de WhatsApp, tornando difícil o acesso rápido e a análise de informações cruciais.

- Dificuldades na Gestão: A falta de um sistema adequado torna difícil a gestão de voluntários, beneficiados e aulas oferecidas pela organização;

- Dificuldades na Análise de Dados: A descentralização dos dados, torna impossível uma estruturação, tratamento ou análise dos dados cruciais para o funcionamento da instituição;

- Limitações no acesso a cursos: As aulas de idiomas oferecidas pela ONG são presenciais e não têm opção de gravação ou acesso online, o que limita o alcance e a disponibilidade para os beneficiados.

## Objetivo do trabalho

Este trabalho tem como principal objetivo propor e desenvolver um sistema de cadastro e distribuição de cestas que nos permita reduzir o consumo excessivo de papel. Para alcançar esse objetivo geral, os seguintes objetivos específicos serão seguidos:

- Desenvolvimento do sistema de cadastro e distribuição de cestas: Permitirá cadastrar beneficiários, dependentes, voluntários, cestas e processo de distribuição;

- Centralizar e digitalizar dados dispersos: Realizar a importação no sistema dos dados dispersos em arquivo físico, planilhas e WhatsApp.

Ao atingir esses objetivos, o projeto visa capacitar a ONG a continuar seu trabalho vital na comunidade de Belo Horizonte com maior eficácia, transparência e impacto positivo para o meio ambiente.

## Definições e Abreviaturas

ONG: ORGANIZAÇÃO NÃO GOVERNAMENTAL

# Requisitos

## Lista de Atores

Nesta seção, serão apresentados os usuários que utilizarão o sistema para gerenciar o cadastro de usuários (papel de administrador) e os usuários que irão operar o sistema, seja realizando cadastros referentes a processos ou o processo propriamente dito.

**Diretor**: Acesso a todos os recursos e cadastro de usuários que irão operar o sistema (perfil de administrador).

**Voluntários**: Usuários cadastrados conforme autorização do diretor e serão responsáveis por operar o sistema e realizar os processos oferecidos pela aplicação (perfil de operador).

## Lista de Funcionalidades

Nesta seção, serão apresentadas as funcionalidades que a aplicação contemplará. São elas:

* Cadastro de beneficiários;
* Cadastro de cestas:
* Cadastro de voluntários;
* Cadastro de dependentes;
* Controle e distribuição de cestas;
* Cadastro de operadores/usuários do sistema
* Vincular usuário/operador logado na distribuição de cestas;
* Buscar beneficiários, dependentes ou voluntários pelo CPF;
* Emitir relatório da distribuição de cestas;

## Requisitos Funcionais

| **ID** | **Descrição Resumida** | **Dificuldade (B/M/A)\*** | **Prioridade**  **(B/M/A)\*** |
| --- | --- | --- | --- |
| RF01 | O sistema deve permitir o cadastro de beneficiários; | B | A |
| RF02 | O sistema deve permitir o cadastro de cestas | B | A |
| RF03 | O sistema deve permitir o cadastro de voluntários; | B | A |
| RF04 | O sistema deve permitir o cadastro de dependentes; | M | A |
| RF05 | O sistema deve permitir o controle da distribuição de cestas; | B | A |
| RF06 | O sistema deve permitir o cadastro dos usuários que irão operar o sistema; | B | A |
| RF07 | O sistema deve possibilitar vincular o usuário no controle de cestas. | B | A |
| RF08 | O sistema deve possibilitar a busca do usuário pelo CPF. | M | M |
| RF09 | O sistema deve permitir gerar relatórios sobre a distribuição das cestas. | B | M |

\*B=Baixa, M=Média, A=Alta.

## Requisitos Não Funcionais

| **ID** | **Descrição** | **Prioridade**  **B/M/A** |
| --- | --- | --- |
| RNF01 | O sistema deve ser acessível nas plataformas web e móvel | A |
| RNF02 | O sistema deve exigir autenticação dos usuários | A |
| RNF03 | O sistema deve executar nos principais navegadores (Chrome, Firefox, Safari e Edge) | A |
| RNF04 | O sistema deve estar disponível 99,9% do tempo, excluindo janelas de manutenção programadas. | A |
| RNF05 | O sistema deve validar se o CPF indicado é válido. | M |

## Descrição Resumida dos Casos de Uso

| **UC01 – Cadastro de Beneficiários** | |
| --- | --- |
| **Descrição** | O sistema deve permitir o cadastro, consulta, edição e exclusão das pessoas que serão beneficiadas com as cestas básicas (beneficiários) |
| **Atores** | Voluntário |
| **Prioridade** | Alta |
| **Requisitos associados** | RF01, RNF02 |
| **Fluxo Principal** | Tela de cadastro:  - O usuário acessa o item de menu “Cadastro > Beneficiários”;  - O sistema exibe a tela “Consultando Beneficiários”;  - O usuários clica sobre a opção “Novo”;  - O sistema exibe a tela de cadastro de beneficiários;  - O usuário preenche os campos obrigatórios do formulário;  - O usuário submete o formulário clicando sobre o botão “Salvar”;  - O sistema valida as informações submetidas no formulário;  - O sistema persiste as informações na base de dados;  - O sistema exibe a mensagem “Beneficiário cadastrado com sucesso”;  - O caso de uso é encerrado. |

| **UC02 – Cadastro de Cestas** | |
| --- | --- |
| **Descrição** | O sistema deve permitir o cadastro das cestas básicas que serão distribuídas aos beneficiários. |
| **Atores** | Voluntários |
| **Prioridade** | Alta |
| **Requisitos associados** | RNF02, RF02 |
| **Fluxo Principal** | - O usuário acessa o item de menu “Cadastro > Cestas Básica”;  - O sistema exibe a tela de consulta;  - O usuário clica na opção “Novo”;  - O sistema exibe a tela de cadastro de cestas;  - O usuário informa os campos obrigatórios;  - nome;  - descrição;  - quantidade em estoque;  - O usuário clica sobre a opção “Salvar”;  - O sistema valida as informações submetidas no formulário;  - O sistema persiste as informações na base de dados;  - O sistema exibe a mensagem “Cesta cadastrado com sucesso”;  - O caso de uso é encerrado. |

| **UC03 – Cadastro de Voluntários** | |
| --- | --- |
| **Descrição** | O sistema deve manter dados referentes aos Voluntários, aquelas pessoas que se voluntariam a apoiar a ONG sem retorno financeiro. |
| **Atores** | Diretor |
| **Prioridade** | Média |
| **Requisitos associados** | RF03, RNF02 |
| **Fluxo Principal** | - O usuário acessa o item de menu “Configuração > Voluntários”;  - O sistema exibe a tela “Consultando Voluntários”;  - O usuário clica sobre a opção “Novo”;  - O sistema exibe a tela de cadastro;  - O usuário preenche os campos obrigatórios do formulário;  - O usuário submete o formulário ao clicar sobre a opção “Salvar”;  - O sistema valida as informações submetidas no formulário;  - O sistema persiste as informações na base de dados;  - O sistema exibe a mensagem “Voluntário cadastrado com sucesso”;  - O caso de uso é encerrado; |

| **UC04 – Cadastro de Dependentes** | |
| --- | --- |
| **Descrição** | O sistema deve permitir o cadastro, consulta, edição e exclusão de Dependentes, pessoas dependentes de um beneficiário; |
| **Atores** | Voluntário |
| **Prioridade** | Alta |
| **Requisitos associados** | RF04, RNF02 |
| **Fluxo Principal** | - O usuário acessa o item de menu “Cadastro > Dependentes” e clica sobre a opção “Novo” ou à partir da tela “Beneficiário”, clica sobre a opção “Novo Dependente”:  - O sistema exibe a tela de cadastro;  - O usuário preenche os campos obrigatórios do formulário:  - Beneficiário (campo de seleção com lista dos beneficiários cadastrados)  - O usuário submete o formulário ao clicar sobre a opção “Salvar”;  - O sistema valida as informações submetidas no formulário;  - O sistema persiste as informações na base de dados;  - O sistema exibe a mensagem “Dependente cadastrado com sucesso”;  - O caso de uso é encerrado; |
|  |  |

| **UC05 – Controle de Distribuição de cestas** | |
| --- | --- |
| **Descrição** | O sistema deve permitir cadastro das distribuições das cestas aos Beneficiários. |
| **Atores** | Voluntário |
| **Prioridade** | Alta |
| **Requisitos associados** | RF05 e RNF02 |
| **Fluxo Principal** | - O usuário clica sobre o item de menu “Distribuição de Cestas”  - O sistema exibe a tela de consulta;  - O usuário clica sobre a opção “Registrar nova distribuição”;  - O sistema exibe o formulário referente à funcionalidade;  - O usuário preenche todos os campos obrigatórios:  - Beneficiário(campo de seleção);  - Cesta (campo de seleção);  - Data/Hora (inicializado com data/hora atual);  - O usuário submete o formulário ao clicar sobre a opção “Confirmar”;  - O sistema valida os dados do formulário;  - O sistema associa o usuário autenticado ao registro;  - O sistema persiste as informações na base de dados;  - O sistema desconta 1(uma) cesta na quantidade em estoque;  - O sistema exibe a mensagem de sucesso “Distribuição registrada com sucesso para o beneficiário {nome\_beneficiario}”;  - O caso de uso é encerrado. |

| **UC06 – Cadastro de Usuários/operadores** | |
| --- | --- |
| **Descrição** | O sistema deve permitir o cadastro de usuários que irão operar o sistema |
| **Atores** | Diretor |
| **Prioridade** | Alta |
| **Requisitos associados** | RNF02, RF06 |
| **Fluxo Principal** | - O usuário acessa o item de menu “Configuração > Usuários”;  - O sistema exibe a tela de consulta de usuários;  - O usuário clica na opção “Novo”;  - O sistema exibe o formulário de cadastro;  - O usuário informa os campos obrigatórios;  - O usuário clica sobre a opção “Salvar”;  - O sistema valida as informações submetidas no formulário;  - O sistema persiste as informações na base de dados;  - O sistema exibe a mensagem “Usuário cadastrado com sucesso”;  - O caso de uso é encerrado. |

| **UC07 – Vincular Usuário no Controle de Cestas** | |
| --- | --- |
| **Descrição** | O sistema deve permitir que o funcionário associe um usuário específico ao controle de cestas, o que é necessário para acompanhar e gerenciar as cestas atribuídas a esse usuário. |
| **Atores** | Funcionário |
| **Prioridade** | Alta |
| **Requisitos associados** | RF07, RFN05 |
| **Fluxo Principal** | - O funcionário acessa a funcionalidade de vinculação de usuário no controle de cestas.  - O sistema exibe a interface para realizar a vinculação.  - O funcionário insere o CPF do beneficiário que deseja vincular.  - O sistema verifica se o CPF é válido e se o beneficiário está cadastrado.  - Se o CPF for válido e o beneficiário estiver cadastrado, o sistema vincula o beneficiário ao controle de cestas.  - O usuário salva a inclusão;  - O caso de uso é encerrado. |

| **UC08 – Buscar informações atreladas ao CPF buscado.** | |
| --- | --- |
| **Descrição** | O sistema deve permitir que o usuário encontre os beneficiários cadastrados por CPF |
| **Atores** | Funcionário |
| **Prioridade** | Alta |
| **Requisitos associados** | RF08 |
| **Fluxo Principal** | - O funcionário acessa o cadastro de beneficiários ou o cadastro de cestas;  - O sistema deve possuir um campo de busca;  - O funcionário insere o CPF do beneficiário que deseja pesquisar;  - O sistema retorna as informações vinculadas ao CPF pesquisado;  - O caso de uso é encerrado. |

| **UC09 – Relatórios sobre a distribuição das cestas** | |
| --- | --- |
| **Descrição** | O sistema deve permitir gerar relatórios mensais sobre a distribuição das cestas, assim como relatórios sobre a distribuição referentes a um beneficiário selecionado. |
| **Atores** | Voluntário, Diretor |
| **Prioridade** | Média |
| **Requisitos associados** | RF09, RNF02 |
| **Fluxo Principal** | - O usuário acessa o item de menu “Distribuição de Cestas”;  - O sistema exibe o formulário da tela de consulta;  - O usuário preenche os campos desejados;  - O usuário submete o formulário clicando sobre a opção “Pesquisar”;  - O sistema busca as informações na base de dados conforme filtros no formulário;  - O sistema exibe as informações na tabela de resultados;  - O usuário clica sobre a opção “Relatório > PDF”;  - O sistema gera um relatório das informações exibidas na tabela;  - O sistema exibe o link de download do relatório;  - O usuário salva o arquivo;  - O caso de uso é encerrado;  - O usuário acessa a funcionalidade de relatório na tela de controle de cestas;  - O usuário seleciona o período desejado para o relatório;  - O usuário confirma ou cancela a ação;  - O caso de uso é encerrado. |

## Restrições Arquiteturais

A aplicação deste projeto, contará com as seguintes restrições arquiteturais:

* O back-end do software deverá ser desenvolvido em Java com Spring Boot.
* A API desenvolvida deve seguir o padrão ReSTful.
* O sistema deve utilizar o SGBD PostgreSQL.

## Mecanismos Arquiteturais

| **Análise** | **Design** | **Implementação** |
| --- | --- | --- |
| Persistência | ORM | Spring JPA |
| Front end | UI | React |
| Back end | MVC | Java / Spring Boot |
| Logs | Logging | Apache Commons Log |
| Exceptions | Tratamento de exceções | Spring Boot Handle Exceptions |
| Teste de Software | Unit Test | JUnit |
| Deploy | CI | Railway |

# Modelagem e diagramas arquiteturais: (Modelo C4)

As principais interações entre usuário e sistema ou entre partes do sistema consiste em:

1. O usuário interage com a aplicação web por meio de seu navegador, enviando solicitações e recebendo como resposta páginas html.

2. A aplicação web faz solicitações JSON/HTTP para a API Spring Boot quando precisa acessar ou modificar dados.

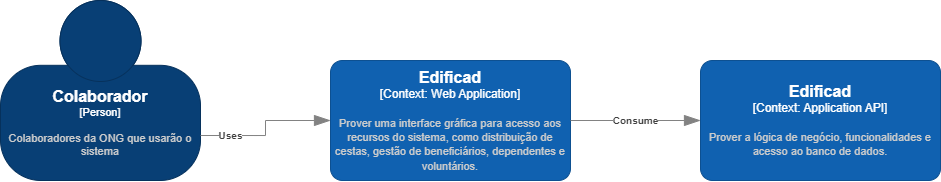
3. A API processa as solicitações da aplicação web, interage com o banco de dados conforme necessário e retorna as respostas requisitadas.

4. O banco de dados armazena os dados conforme solicitado pela API.



**Figura 1 - Visão Geral**

## Nível 1: Diagrama de Contexto



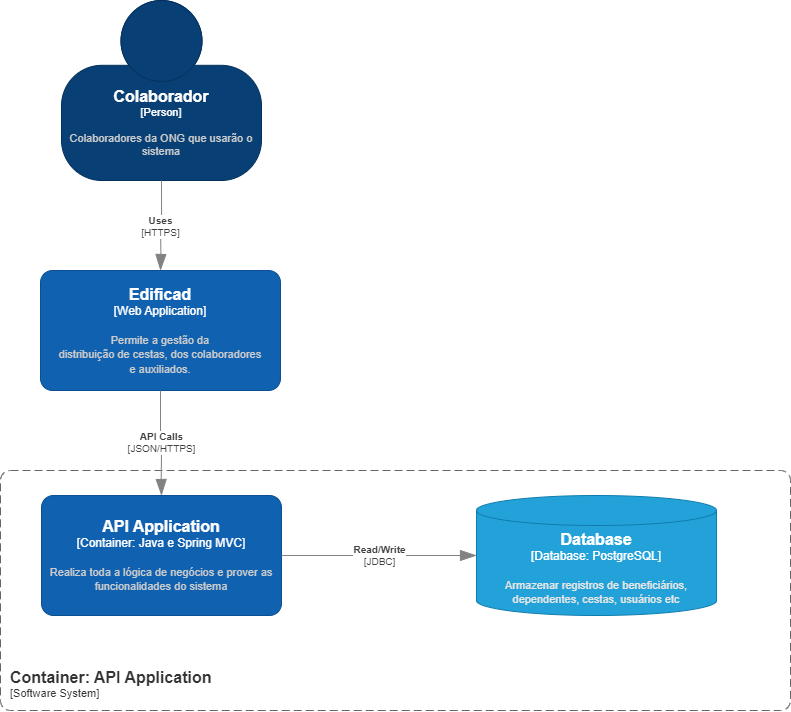
**Figura 2: Diagrama de Contexto**

1. Person(usuários do sistema). Esse é o usuário final que interage com o sistema e irá utilizar a aplicação web.

2. Web Application: Interface que o usuário utiliza para interagir com o sistema e é responsável por apresentar informações e funcionalidades ao Colaborador. Essa camada irá se comunicar com a API e obter ou enviar informações de acordo com as operações realizadas pelo usuário.

3. Application API: Camada de back end do sistema e será responsável por gerenciar a lógica de negócios, acessar e manipular dados e responder às solicitações/requisições da aplicação web.

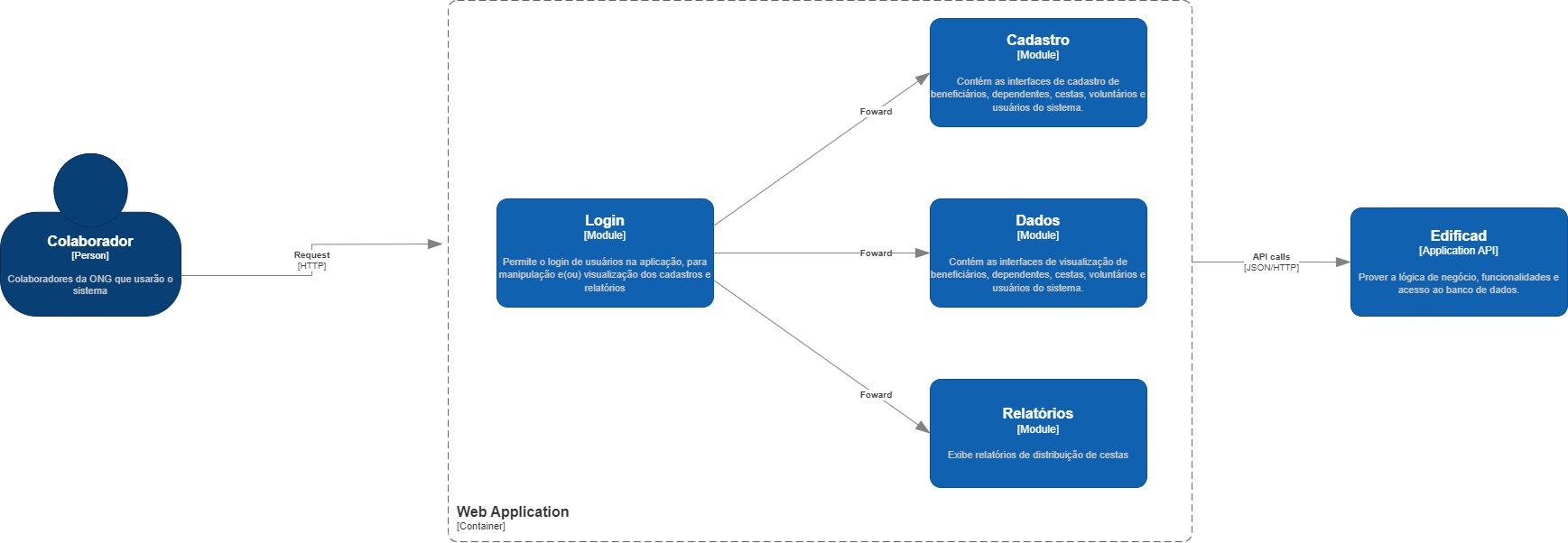
## Nível 2: Diagrama de Contêiner



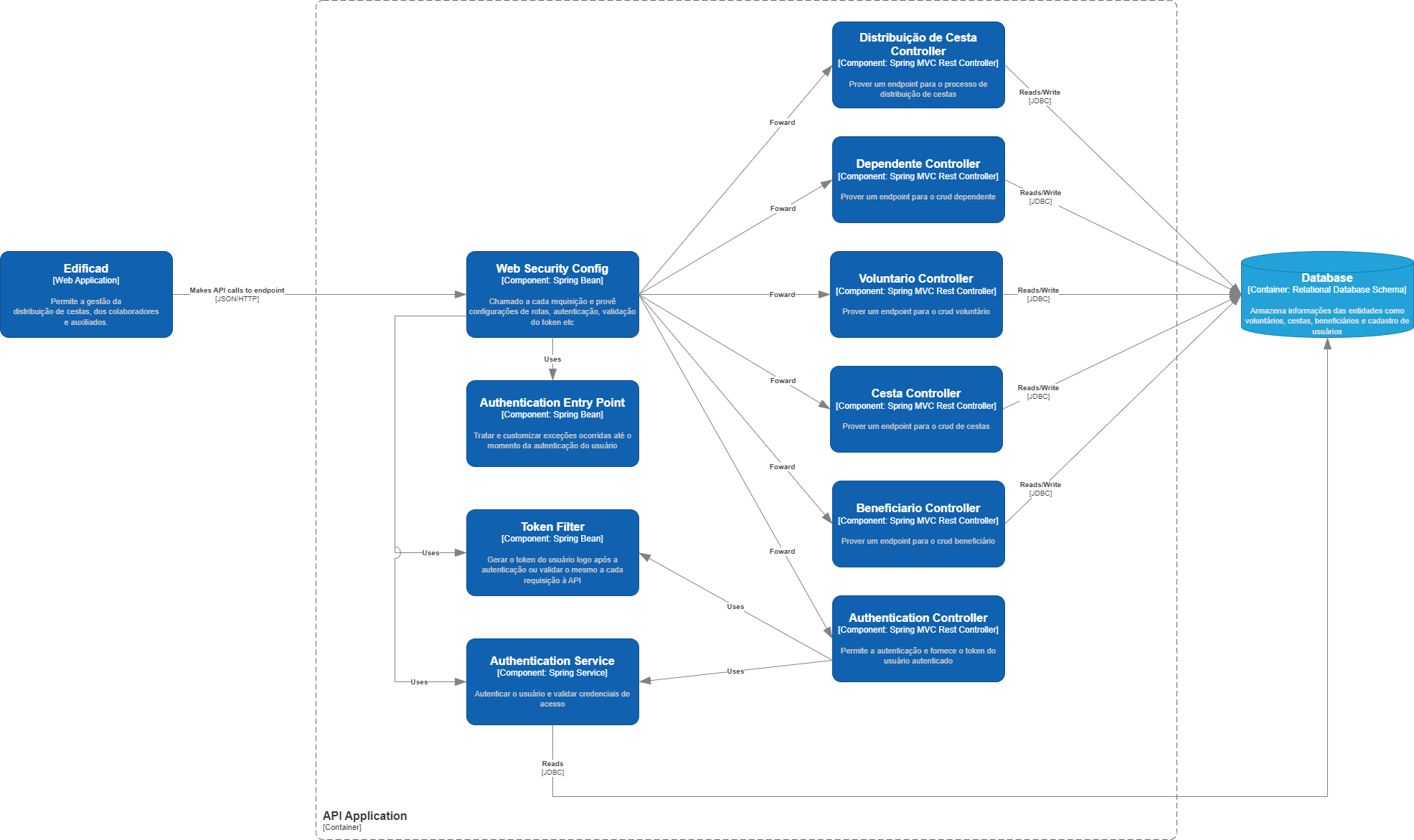
**Figura 3 – Diagrama de Contêiner**

1. Container - Application API: API que atuará como back-end e realizará toda a lógica de negócio, controle de usuários além de acessar e manipular dados em um banco PostgreSQL conforme requisições da view.

## Nível 3: Diagrama de Componentes



**Figura 4 - Diagrama de Componentes (Web Application)**



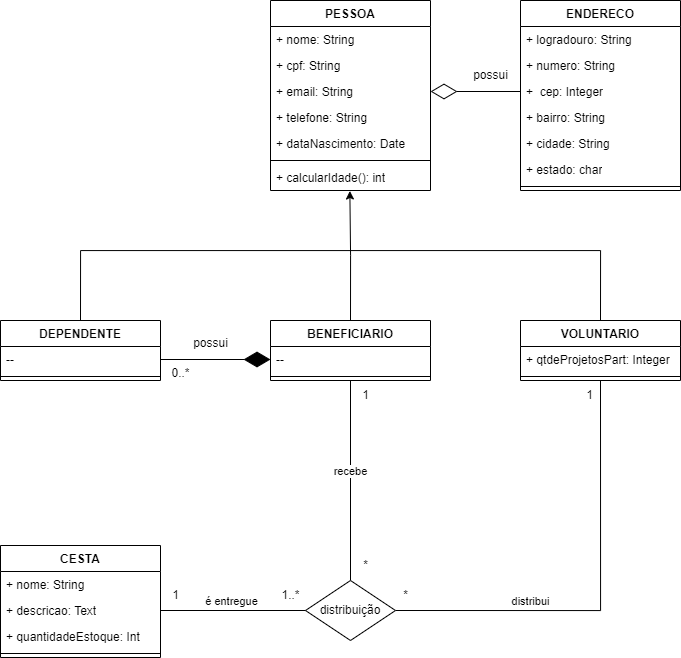
**Figura 5 – Diagrama de Componentes (API)**

1. Web Security: Uma camada de segurança chamada a cada requisição a um endpoint ou a qualquer ‘path’ da API. Responsável por definir as rotas da aplicação, quais rotas requer autenticação, chamar o serviço responsável pela validação do token e encaminhar a requisição para o controller correspondente ao endpoint requisitado.

2. Authentication Controller: Controller responsável por autenticar o usuário e retornar um token que será utilizado para acesso a todos os endpoints da API.. O endpoint correspondente a esse controller é o único que não requer autenticação no Web Security.

3. Controller: Controladores correspondentes aos endpoints da API e responsáveis por chamar os serviços(que concentram toda regra de negócio). Os serviços, além de executar as regras de negócio, são os responsáveis por chamar os repositórios para operações de leitura e escrita na base de dados.

## Nível 4: Código



**Figura 6 – Diagrama de Classe**

Diagrama de classe que representa a estrutura estática do sistema proposto, descrevendo os elementos fundamentais do processo de gestão da organização parceira e tem como objetivo gerenciar a distribuição de cestas, registros de voluntários, beneficiários e informações dos usuários.

# Avaliação Crítica dos Resultados

| **Ponto avaliado** | **Descrição** |
| --- | --- |
| React | Documentação rasa, sem detalhatamentos de navegação em SPA |
| TypeScript | Mensagens de erros genéricas, dificultam a compressão |
| Componentização | Complexidade de configuração com bibliotecas externas |
| Spring Boot | Facilidade no tratamento de exceções |
| JPA | Facilita a portabilidade do código entre diferentes sistemas de gerenciamento de banco de dados (SGBD |
| Spring Boot | Complexidade reduzida, aumento da produtividade |

# Conclusão

***5.1 Visão do Grupo***

O desenvolvimento do projeto proposto no presente documento foi concluído e o objetivo geral atingido. Criou-se um sistema web de cadastro, para atender a necessidade da ONG Edificando Vidas. Com a criação desse projeto a ONG conseguirá controlar estoque, cadastro de beneficiários e voluntários, assim como controlar a distribuição de cestas, por beneficiários e seus dependentes.

***5.1 Visão do Cliente***

O cliente demonstrou sua gratidão pelo grupo criado como plataforma de contato, com o seguinte feedback, através de uma mensagem de texto: “Sistema muito prático, de fácil utilização e que vai ajudar a ONG Edificando Vidas no cadastro de cestas, voluntários e assistidos pelo nosso projeto de distribuição de cestas. Obrigado ao grupo EDIFICAD.”

Também é possível ouvir o áudio de agradecimento enviado pelo cliente através do link: <https://drive.google.com/file/d/1UIxOvwyRt6xbh1_LDw0eFO35d_7fcGO6/view?usp=drive_link>

Dessa forma, conclui-se que o projeto foi finalizado e as necessidades foram atendidas.

# Referências

Spring Web Applications: <https://spring.io/web-applications> <Acesso em: 15 de outubro de 2023>

Thymeleaf - Integração com Spring: <https://www.thymeleaf.org/doc/tutorials/3.0/thymeleafspring.html> <Acesso em: 28 de novembro de 2023>

Apache POI - Java API para processamento de documentos do Microsoft Office: <https://poi.apache.org> <Acesso em: 10 de outubro de 2023>

Chart.js - Biblioteca JavaScript para criar gráficos interativos: <https://www.chartjs.org> <Acesso em: 5 de novembro de 2023>

React. Disponível em: https://react.dev. Acesso em: 10 de outubro de 2023.

Tailwind CSS. Disponível em: https://tailwindcss.com. Acesso em: 12 de outubro de 2023.

Material UI. Disponível em: https://mui.com. Acesso em: 18 de outubro de 2023.

ReactJS Popup (npm). Disponível em: https://www.npmjs.com/package/reactjs-popup. Acesso em: 22 de outubro de 2023.

Radix UI. Disponível em: https://www.radix-ui.com. Acesso em: 27 de outubro de 2023.

Phosphor React. Disponível em: https://www.radix-ui.com. Acesso em: 4 de novembro de 2023.

React-bootstrap. Disponível em: https://react-bootstrap.netlify.app. Acesso em: 9 de novembro de 2023.

# APÊNDICES

Trello: <https://trello.com/b/MyqyilyG/ti-sustentabilidade>

Back-end: <https://github.com/jarnunes/edificad>

Front-end: <https://github.com/GuildNerd/edifi-cad>

OBS.: Possivelmente, até a próxima sprint o back e front estarão em um único repositório.