



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

REDSON FARIAS BARBOSA FILHO

**APOIO CARIDADE: UMA APLICAÇÃO PARA INTERMÉDIO
ENTRE ORGANIZAÇÕES DE TERCEIRO SETOR E PESSOAS
PRESTATIVAS**

CAMPINA GRANDE - PB

2021

REDSON FARIAS BARBOSA FILHO

**APOIO CARIDADE: UMA APLICAÇÃO PARA INTERMÉDIO
ENTRE ORGANIZAÇÕES DE TERCEIRO SETOR E PESSOAS
PRESTATIVAS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de **Bacharel ou
Bacharela** em Ciência da Computação.**

Orientadora: Professora Francilene Procópio.

CAMPINA GRANDE - PB

2021

REDSON FARIAS BARBOSA FILHO

**APOIO CARIDADE: UMA APLICAÇÃO PARA INTERMÉDIO
ENTRE ORGANIZAÇÕES DE TERCEIRO SETOR E PESSOAS
PRESTATIVAS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de **Bacharel ou
Bacharela** em Ciência da Computação.**

BANCA EXAMINADORA:

Professora Francilene Procópio

Orientador – UASC/CEEI/UFCG

Professora Patrícia Machado

Examinador – UASC/CEEI/UFCG

Professor Tiago Lima Massoni

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 11 de abril de 2022.

CAMPINA GRANDE - PB

ABSTRACT

Third sector institutions are organizations committed to social problems, where they usually count on the support of helpful people who value their work and wish to help them through donations. However, even with the support of these people, many of these organizations face difficulties related to the scarcity of resources and materials necessary for their maintenance and continuity. Therefore, this work aims to develop a PWA (Progressive Web App) application to provide functionalities aimed at creating portfolios and publications of these organizations. In the development process, the agile methodology based on scrum was used. To assess its usability, the method based on the Likert scale was used. The results obtained in the evaluation of the application show a good degree of user satisfaction and its potential to serve as a digital environment for donors and third sector institutions.

Apoio Caridade: Uma aplicação para intermédio entre organizações de terceiro setor e pessoas prestativas

Redson Farias Barbosa Filho
redson.filho@ccc.ufcg.edu.br
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil

Francilene Procópio
garcia@computacao.ufcg.edu.br
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil

RESUMO

Instituições do terceiro setor são organizações comprometidas com problemas sociais, onde costumam contar com o apoio de pessoas prestativas que valorizam seus trabalhos e desejam ajudá-las através de doações. Porém, mesmo com o apoio dessas pessoas, muitas dessas organizações passam por dificuldades relacionadas com a escassez de recursos e materiais necessários para sua manutenção e continuidade. Diante disso, esse trabalho tem como objetivo o desenvolvimento de uma aplicação PWA (Progressive Web App)[3] para fornecer funcionalidades voltadas para criação de portfólio e publicações dessas organizações. No processo de desenvolvimento, foi utilizada a metodologia ágil baseada no scrum. Para avaliar sua usabilidade, foi utilizado o método baseado na escala Likert. Os resultados obtidos na avaliação da aplicação mostram um bom grau de satisfação dos usuários e seu potencial para servir como um ambiente digital para doadores e instituições do terceiro setor.

PALAVRAS-CHAVE

Desenvolvimento de software, terceiro setor, doação, caridade.

1 INTRODUÇÃO

O terceiro setor é a denominação dada para organizações como ONGs, Entidades Beneficentes e Institutos, que são organizações sem fins lucrativos e que contam com o apoio de uma rede de pessoas prestativas para se manter. A ideia principal dessas organizações é gerar impacto positivo na sociedade, através de diversas ações como assistência social, pesquisa científica e associações de voluntariado, com o intuito de torná-la menos desigual.

No cenário atual da pandemia, essas instituições estão tendo dificuldades para se manter, devido aos recursos adquiridos através das doações não serem suficientes para lidar com as altas demandas sociais, fazendo com que muitas dessas instituições tenham que reduzir suas atividades ou até mesmo tenham que fechar. Com isso, observa-se como grande parte da sociedade não conhece os trabalhos e as causas dessas organizações.

Diante disso, é de grande importância uma aplicação que sirva como ponte entre a sociedade e as instituições de terceiro setor, onde as pessoas podem conhecer tanto as instituições quanto os projetos e as causas que cada uma defende, fazendo com que haja maior envolvimento entre esses dois grupos e aumente o número de doações.

2 SOLUÇÃO

Esse projeto tem como objetivo desenvolver uma aplicação PWA que permita a criação de conteúdo das instituições de terceiro setor, para que elas possam mostrar seus projetos e necessidades.

Sendo assim, as instituições poderão criar seu portfólio, a fim de mostrar as causas que defendem e as informações que acharem mais importantes. Além disso, poderão criar publicações do modo que lhes for mais útil.

2.1 Funcionalidades

A aplicação conta com dois tipos de usuários, a Instituição e o Doador. O cadastro de ambos é feito de forma distinta, logo, o tipo de usuário é definido no momento do cadastro.

2.1.1 Funcionalidades do doador. Visualizar instituições, onde o doador tem acesso a todas as instituições da aplicação, filtrando por estado, cidade e tipo de instituição, onde podem selecionar uma instituição para ver suas informações e suas publicações.

Feed de publicações, onde o doador pode ver todas as publicações de todas as instituições por meio de um feed de publicações, filtrando por estado, cidade e tipo de instituição.



Figura 1: Listagem das instituições.



Figura 2: Feed de publicações.



Figura 4: Configurações da instituição

2.1.2 *Funcionalidades da instituição.* Editar informações, onde a instituição consegue gerenciar suas próprias informações e montar um perfil completo e customizável.

Gerenciar publicações, onde a instituição pode criar, visualizar, editar ou excluir publicações da maneira que desejar.



Figura 3: Adicionar publicação.

3 METODOLOGIA

Durante o levantamento de requisitos da aplicação, a experiência de desenvolvimento adquirida ao decorrer do curso foi muito importante, onde foram definidas funcionalidades voltadas para exibição e filtragem de conteúdo no caso de usuários do tipo doador, e funcionalidades de criação e gerenciamento para usuários do tipo instituição.

Para ajudar na construção do design da interface, foi utilizada a dribbble, uma comunidade utilizada por designers para compartilhar seus trabalhos. Onde foi possível usar vários designs diferentes como inspiração para a interface da aplicação.

Pensando no tipo de aplicação, foi escolhido o modelo PWA, um modelo de aplicação que se comporta de forma semelhante a um aplicativo nativo. Esse modelo oferece mais praticidade para o desenvolvimento, onde uma única base de código abrange várias plataformas, além disso, a PWA se comporta de maneira menos burocrática na visão do usuário, onde o mesmo tem acesso a aplicação através de uma url e ainda possui a opção de adicionar o aplicativo à tela inicial com apenas alguns toques.

4 ARQUITETURA

A arquitetura adotada nesta aplicação foi o modelo cliente-servidor[5], onde o cliente faz uma requisição ao servidor que retorna uma resposta com os dados solicitados. A aplicação é dividida em duas partes: O frontend, que é a interface da aplicação e assume o papel de cliente da arquitetura, e o backend, que é responsável por aplicar as regras de negócio e persistir os dados e assume o papel de servidor da arquitetura. A comunicação entre ambas é dada pelo modelo API REST.

4.1 Backend

4.1.1 Tecnologias do Backend. O backend foi desenvolvido utilizando o NodeJS, uma plataforma que permite a execução de códigos JavaScript do lado do servidor de forma escalável e eficiente. Essa tecnologia foi escolhida por sua alta flexibilidade e fácil manutenção, garantindo um processo de desenvolvimento mais prático e produtivo.

Para armazenar os dados, foi escolhido o MongoDB, um banco de dados NoSQL que armazena e gerencia informações que são orientadas a documentos, tornando o gerenciamento mais prático pela aplicação.

4.1.2 Estrutura do Backend. A organização das pastas do backend foi feita tendo como base boas práticas adquiridas em experiências de projetos passados, de modo que são separadas em 5 camadas principais: *config*, *controllers*, *models*, *middlewares* e *routes*.

Os *controllers* são responsáveis por realizar a lógica de manipulação de dados e aplicar a regra de negócio da aplicação.

Os *models* definem as estruturas das entidades da aplicação que ficam armazenadas no banco de dados.

Os *middlewares* são um conjunto de métodos utilizados como interceptores, eles têm acesso aos objetos de requisição e resposta da chamada, facilitando o tratamento das requisições.

As *routes* são o conjunto de endpoints da API, que são responsáveis por fazer a ligação entre o frontend e os serviços do backend.

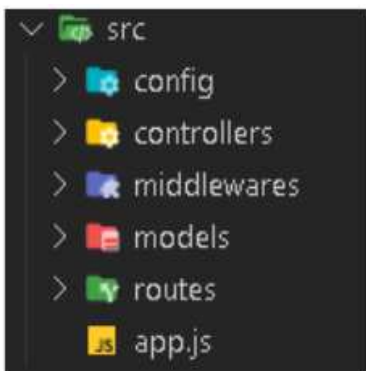


Figura 5: Estrutura do backend.

4.2 Frontend

4.2.1 Tecnologias do Frontend. O frontend foi implementado utilizando o React, uma biblioteca JavaScript para criar interfaces de usuário. O React é baseado em componentes, o que torna o desenvolvimento do sistema bastante flexível e modularizado, proporcionando boa reutilização de código. Essa tecnologia foi escolhida por possuir grande afinidade com a mesma.

Para realizar a comunicação com o servidor foi utilizado o Axios, uma biblioteca de código aberto, que pode ser utilizada tanto no navegador quanto no NodeJS. Ela é um cliente HTTP que consegue se comunicar com servidores REST através de requisições HTTP.

4.2.2 Estrutura do Frontend. A estrutura do frontend foi dividida em 3 camadas principais: *components*, *pages* e *services*, e em 5 camadas auxiliares: *animation*, *constants*, *context*, *routes* e *util*.

Os *components* representam os componentes presentes nas *pages*, eles possuem a finalidade de reutilizar o código de elementos que se repetem ao longo da aplicação.

As *pages* representam todas as páginas presentes na interface da aplicação, onde a cada uma é atribuída uma rota em uma das *routes*.

Os *services* são responsáveis por fazer a interação entre a aplicação e os serviços fornecidos pela API da aplicação.

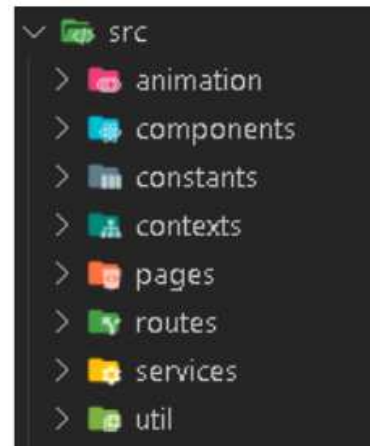


Figura 6: Estrutura do frontend.

4.3 Autenticação

Para realizar a autenticação do sistema, foi utilizado o modelo JWT (Json Web Token)[2], que permite compartilhar informações de forma compacta e segura através de um objeto JSON. Essa biblioteca codifica um token com as informações desejadas, no caso da autenticação é utilizado para codificar os dados do usuário que está fazendo a requisição.

4.3.1 Fluxo de autenticação. O usuário passa o email e a senha, que faz com que o backend verifique se existe um usuário que possua os mesmos dados, em caso de sucesso, é gerado um token de acesso com informações do usuário e retornado como resposta para o frontend, que armazena o token recebido para enviar no campo Authorization dos headers das requisições que necessitam de autenticação.

5 AVALIAÇÃO

5.1 Deploy

Para dar início ao processo de testes, a aplicação foi hospedada no Heroku, uma plataforma que permite hospedagem, configuração, testagem e publicação de projetos virtuais na nuvem. Por se tratar de uma versão de teste do sistema, essa plataforma foi escolhida por disponibilizar recursos gratuitos que atendem as necessidades da aplicação mesmo possuindo algumas limitações de espaço de armazenamento e tempo de resposta.

5.2 Metodologia de Avaliação

Para a avaliar o Apoio Caridade foi utilizado um formulário do Google contendo perguntas baseadas no método da escala Likert [4], que é uma das metodologias mais populares do mundo quando se trata de pesquisas de satisfação. Essa metodologia oferece uma opção de resposta em uma escala de pontos com descrições verbais, com isso, foram utilizados 5 níveis de intensidade: concordo totalmente, concordo, nem concordo nem discordo, discordo e discordo totalmente.

O formulário contou com 6 perguntas, onde foi possível analisar os seguintes pontos da aplicação: usabilidade, utilidade, interface, comportamento, desempenho e satisfação geral.

5.3 Resultados

Os resultados obtidos representam o nível de satisfação de 25 pessoas de perfil acadêmico que se interessaram em conhecer a aplicação do Apoio Caridade.

Usabilidade: representa a capacidade da aplicação em fazer com que o usuário consiga utilizar suas funcionalidades de maneira fácil e eficiente.



Figura 7: Gráfico de satisfação de usabilidade.

Utilidade: representa o nível de serventia da aplicação em meio ao contexto proposto.



Figura 8: Gráfico de satisfação de utilidade.

Interface: representa o conjunto de elementos visuais da aplicação e a forma como são estruturados.

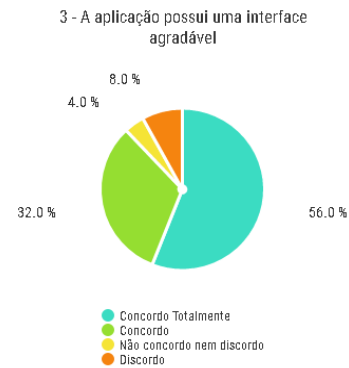


Figura 9: Gráfico de satisfação da interface.

Comportamento: representa a forma como a aplicação reage às ações do usuário.

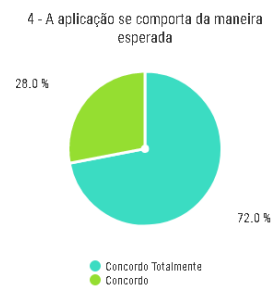


Figura 10: Gráfico de satisfação do comportamento.

Desempenho: representa a capacidade da aplicação em processar dados e realizar ações de forma rápida e eficiente.

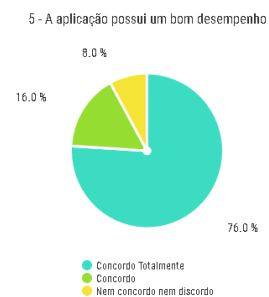


Figura 11: Gráfico de satisfação do desempenho.

Satisfação geral: representa o nível de satisfação que a aplicação trouxe para os usuários que a utilizaram.

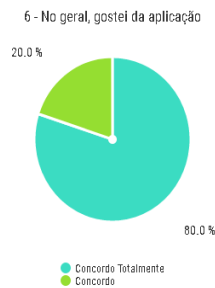


Figura 12: Gráfico de satisfação da satisfação geral.

Convertendo os valores descritivos da escala Likert em uma escala numérica com intervalo de 1 à 5, conseguimos obter as médias dos valores de cada ponto da aplicação, onde pode se notar com mais facilidade como os resultados obtidos tiveram um grau de satisfação muito positivo.

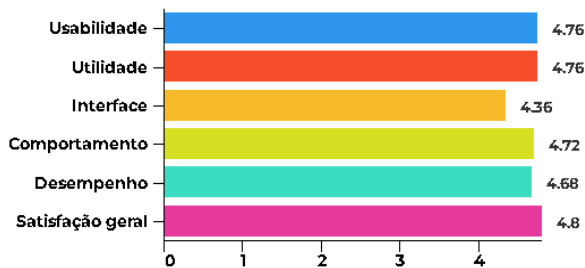


Figura 13: Gráfico de médias da avaliação.

6 EXPERIÊNCIA

6.1 Processo de desenvolvimento

No processo de desenvolvimento da aplicação foi adotado uma metodologia ágil baseada nos princípios do Scrum[1], uma metodologia baseada em ciclos que permitem o desenvolvimento das funcionalidades de forma progressiva, onde foram aproveitados pontos importantes para um desenvolvimento individual.

Essa metodologia adaptada é composta por ciclos que são chamados de Sprints, intervalos curtos de tempo onde serão executadas as funcionalidades definidas. Todas as funcionalidades desenvolvidas na aplicação foram definidas no Product Backlog, uma lista de todos os requisitos que se tem conhecimento de que precisam estar no produto.

Com isso, foi definido o Product Backlog da aplicação com suas funcionalidades onde cada uma possuía um nível de prioridade, em seguida foram definidas Sprints com duração de 2 semanas de modo a distribuir as funcionalidades dentro delas levando em conta seu nível de prioridade.

Inicialmente, foi feita a prototipação das telas da aplicação através do Figma, uma ferramenta de design de interface. Com isso, foi possível refinar algumas funcionalidades do Product Backlog, além de facilitar o processo de desenvolvimento da interface.

O processo de desenvolvimento se iniciou no backend, com a implementação das regras de negócio e das funcionalidades definidas no Product Backlog. Após isso, se deu início a implementação do frontend, onde foi desenvolvida a interface do usuário responsável por consumir os serviços criados no backend.

6.2 Principais Desafios

Dentre os desafios enfrentados, o que mais se destacou foi a implementação do upload de imagens do lado do backend. Por ter mais afinidade com frontend, acabou sendo minha primeira experiência com upload de arquivos em um servidor AWS, com isso houve uma certa dificuldade para entender a lógica necessária e conseguir aplicá-la no backend.

6.3 Trabalhos Futuros

O Apoio Caridade possui um escopo simples, mas que representa uma base muito sólida e escalável para um ambiente com grande interação entre a sociedade e as instituições do terceiro setor, sendo assim, em trabalhos futuros podem ser adotadas novas funcionalidades, como:

- Seguir instituições.
- Adicionar comentários nas publicações.
- Usuários do tipo Instituição terem acesso a outras instituições e ao feed de publicações.

REFERÊNCIAS

- [1] Marcio Roberto Andrade. 2021. Metodologia Scrum. Retrieved 18 de Março de 2022 from <https://blog.contaazul.com/metodologia-scrum>
- [2] Aylan. 2019. Como o JWT funciona. Retrieved 16 de Março de 2022 from <https://www.devmedia.com.br/como-o-jwt-funciona/40265>
- [3] Rock Content. [n. d.]. Entenda o que são Progressive Web Apps (PWAs) e veja os melhores exemplos do mercado. Retrieved 9 de Abril de 2022 from <https://rockcontent.com/br/blog/progressive-web-apps/>
- [4] Rafaela Frankenthal. 2022. Entenda a escala Likert e saiba como aplicá-la em sua pesquisa. Retrieved 16 de Março de 2022 from <https://mindminers.com/blog/entenda-o-que-e-escala-likert/>
- [5] Canal TI. 2018. Arquitetura cliente-servidor. Retrieved 18 de Março de 2022 from <https://www.canalti.com.br/arquitetura-de-computadores/arquitetura-cliente-servidor/>