

PROJETO FACILITA

Pedro Augusto Silva Ribeiro

Centro Universitário de Anápolis - UniEVANGÉLICA. pedro.ouito@gmail.com

Yuri Moura Marques

Centro Universitário de Anápolis - UniEVANGÉLICA. yuri.marques@aluno.unievangelica.edu.br

Rodrigo De Souza Costa

Centro Universitário de Anápolis - UniEVANGÉLICA. rodrigo.costa@aluno.unievangelica.edu.br

Lucas Carvalho Maracaípe

Centro Universitário de Anápolis - UniEVANGÉLICA. lucas.maracaípe@aluno.unievangelica.edu.br

Abiezer Sidônio José Odanaca Junior

Centro Universitário de Anápolis - UniEVANGÉLICA. abiezerodanacajosejunior@gmail.com

RESUMO

O problema social abordado neste projeto se refere a questão das ONGs. Muitas instituições de caridade passam por períodos de dificuldades com frequência. Dentre essas dificuldades, pode-se citar, a falta de apoio financeiro, carência de alimentos e vestimentas, dentre outras dificuldades recorrentes. O projeto facilita surgiu com a intenção de aproximar as pessoas que desejam doar as ONGs necessitadas. As doações poderão ser feitas tanto da forma online, através da plataforma Web Facilita, quanto da forma presencial, pois a plataforma disponibilizará o endereço dessas ONGs, as doações feitas de forma online serão apenas transições bancárias. Doações de roupas, alimentos, entre outros poderão ocorrer apenas de forma presencial, com o usuário doador se deslocando até a ONG desejada. Para a construção da plataforma foram utilizados a biblioteca *React* juntamente com a ajuda do *Bootstrap*. Nesta plataforma digital contará com *Leaflet*, uma biblioteca JavaScript para mapeamento Web, dessa maneira facilita encontrar a localização das ONGs próximas dos usuários doadores de maneira mais rápida e eficiente. O banco de dados desse projeto foi modelado utilizando o sistema de gerenciamento de banco de dados (SGBD) *PostgreSQL*. Para a construção da arquitetura da plataforma foi utilizado o *Spring Framework*, que é um *framework open source* criado para plataforma Java, com módulos voltados para desenvolvimento *Web*, juntamente com o *Spring Data JPA* para fazer a persistência de dados. O resultado esperado para o final do projeto é facilitar a interação dos usuários doadores com as ONGs que precisam de ajuda, levando assim mais doações e auxílio para as instituições necessitadas. Tendo em vista os aspectos observados, foi concluído e ressaltado a importância de ajudar estas instituições que passam por grandes dificuldades, principalmente em períodos de crise. O auxílio às instituições é de suma importância, pois as mesmas contribuem para o crescimento de bondade e empatia na sociedade.

Palavras-chave: Desenvolvimento Web; Organizações não governamentais; Sistema de Doações; Projeto de Software.

1. Introdução

Na atualidade, com a pandemia de Corona Vírus, diversas famílias tiveram sua condição financeira agravada. Portanto, a atividade de ONGs se torna cada vez mais importante. Ademais, não há uma plataforma que mapeia exclusivamente essas instituições e dessa maneira, as informações sobre como realizar doações se encontra de maneira dispersa. Sendo assim, visando essa dificuldade de realizar doações para ONGs, foi proposta a realização de um projeto para o desenvolvimento de uma aplicação Web, com intuito de intermediar doações a estas instituições, e mapear os seus endereços. Desta forma, este projeto tem o objetivo de facilitar e encurtar o trajeto para a realização de doações.

O desenvolvimento desta ideia ocorreu através de uma sequência reuniões, realizadas por meio do Discord, que foi a aplicação utilizada para realizar a comunicação do grupo durante a realização do projeto. Como resultado destas reuniões foi decidido o tema a ser abordado, e definido um projeto de natureza social. Sendo usado como inspiração para a construção da plataforma Facilita, diversos projetos Web, como por exemplo: Vakinha, Abacashi e Ajudaê, que oferecem serviços de doação similares ao que foi proposto, estes serviços tem o objetivo de ajudar as pessoas a receberem doações, porém o projeto citado neste relato é direcionado, principalmente, às ONGs e instituições de caridades.

Para a criação da plataforma Web foram utilizadas diversas tecnologias que auxiliaram na sua construção. Com o progresso de desenvolvimento, passaram-se por diversas ferramentas que acrescentaram e direcionaram para o bom êxito do projeto. O grupo teve grande ajuda dos diagramas UML, cuja ajuda se demonstrou de extrema importância para a visualização de diversos aspectos do projeto. A UML é uma linguagem de notação que tem como intuito modelar e documentar qualquer desenvolvimento de sistemas e possui uma utilidade de grande importância, a mesma ajuda a equipe na compreensão do funcionamento do que está sendo desenvolvido.

Com a condução dos diagramas a equipe teve uma visão mais clara do que usar para o desenvolvimento, tomando assim, mais conhecimento sobre o uso de frameworks, bibliotecas, de diversas linguagens de programação, além SGBDs e da ajuda do open source. Frameworks são conjuntos de códigos prontos que auxiliam e ajudam no desenvolvimento de plataformas web ou em diversos softwares, em sua maioria possuem funções específicas para complemento do projeto. A equipe teve ajuda também das bibliotecas das linguagens de desenvolvimento utilizadas, as bibliotecas possuem o objetivo de entregar soluções utilizando-se de funções ou métodos. Dessa maneira, a aplicação web desenvolvida atinge maior nível de produtividade e qualidade, garantindo a boa performance da mesma.

Contudo, com a progressão do projeto, o grupo teve que se decidir qual SGBD utilizar, em um dos encontros marcado pelos membros foi decidido o uso do *PostgreSQL*. A escolha foi feita tendo como base o conhecimento do grupo sobre a ferramenta, conhecimento esse adquirido nas aulas de banco de dados que foram realizadas no mesmo período de construção desse projeto. O PostgreSQL é gerenciador de banco de dados open source. Ademais, o mesmo dispõe de recursos importantes como a capacidade de fazer consultas complexas, chaves estrangeiras, além da facilidade de acesso. O grupo também utilizou diversas linguagens para a construção da plataforma, focando sempre mais nas linguagens Java e JavaScript. O Java foi utilizado pelo conhecimento da equipe sobre a linguagem, sendo a mesma uma ferramenta muito útil e eficaz para construção da plataforma, juntamente com o JavaScript que como o *PostgreSQL*, a equipe aprendeu a utilizar essa ferramenta no decorrer do bimestre, mas dessa vez, nas aulas de Programação e Tecnologias Web.

2. Metodologia

A definição dos requisitos foi realizada através das reuniões organizadas por todos, onde foi definido como seria executado todo o trabalho desde o escopo até a sua forma final. As reuniões foram feitas

pelo aplicativo Discord onde o grupo se reunia para elaboração do projeto utilizando a técnica de Brainstorming. Dessa maneira, o projeto teve a colaboração de todos os membros e foi desenvolvido utilizando o conhecimento e as ideias do grupo como um todo. A idealização da plataforma foi atingida após todos os membros do grupo perceberem a dificuldade de uma pessoa achar ONGs próximas para realizar doações.

Durante o processo de desenvolvimento da plataforma foi utilizado diversas ferramentas para atingir realizar o projeto. Entre essas ferramentas estão o *React.js* juntamente com o *Bootstrap* para a construção do site, foi utilizado também o *Leaflet*, uma biblioteca JavaScript para mapeamento Web, para facilitar assim a localização das ONGs por parte do usuário. Para a construção do Banco de Dados foi utilizado o SGBD *PostgreSQL*, onde foi feita a modelagem do banco de dados.

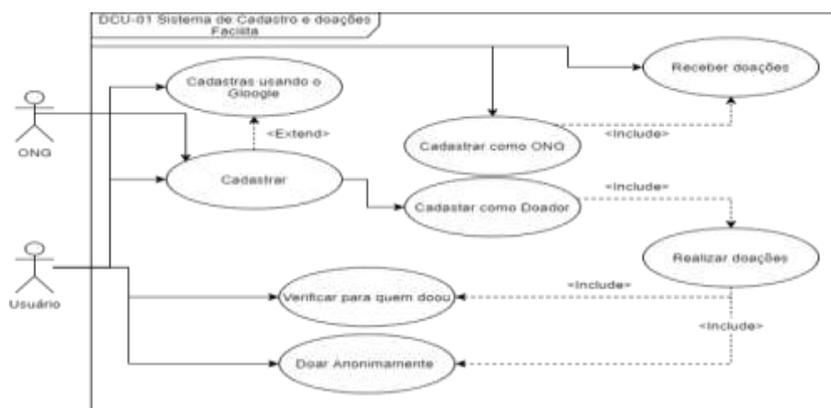
Para o desenvolvimento dos diagramas do projeto foi utilizado a ferramenta *Draw.io*, uma ferramenta utilizada com unanimidade pelos membros da equipe para a criação dos diagramas UML. Os diagramas foram feitos de forma conjunta por todos da equipe.

3. Resultados e discussão

O projeto começou de uma maneira complicada, pois o grupo havia realizado um projeto no semestre passado e queria dar continuidade, porém esse projeto não tinha relação com o tema meio ambiente, sustentabilidade e intervenção humana ou Acessibilidade e inclusão social. Com isso, outro projeto deveria ser pensado. Cada integrante do grupo ficou responsável por pensar em uma ideia e essas ideias foram debatidas, por meio de um aplicativo de voz chamado Discord. Uma ideia em que todo pessoal do grupo aprovou foi de criar um site que lista todas as instituições perto do usuário e ter um serviço para realizar pagamentos para essas instituições de modo virtual. Com a ideia em mente, faltava escolher o nome do projeto, várias sugestões foram cogitadas como: OnGreen, Growing, Ajudei, Financiei e Facilita. Facilita foi a escolha.

Com a ideia inicial do projeto em mente, faltava definir os requisitos. Uma reunião foi realizada para definir os requisitos e fazer a prototipação, porém foi um desastre, pois haviam dúvidas em relação a ideia inicial, que não ficou clara para alguns integrantes e outra dificuldade foi utilizar a ferramenta de prototipação. Foi decidido que a prototipação apenas seria realiza após definição dos requisitos e após isso a definição dos requisitos foi feita combinado a alguns diagramas utilizando a linguagem de modelagem unificada (UML), principalmente o diagrama de casos de uso, como é mostrado na Figura 1.

Figura 1 – Diagrama de casos de uso.



Fonte: Elaborado pelo autor.

Depois que os requisitos foram definidos, outra reunião foi realizada para fazer a prototipação do site. Essa reunião foi outro desastre, pois novamente faltava conhecimento sobre a ferramenta de prototipação, faltava noções de estilização e também não havia sido estabelecido um Produto Mínimo Viável (MVP). Com esses problemas não foi possível concluir a prototipação, mas um MVP foi definido.

Após a definição do MVP, ficou decidido que não ocorreria uma prototipação antes da parte do desenvolvimento. Sem ter feito um planejamento prévio do layout e da estilização da página, os resultados não ficaram tão agradáveis visualmente como esperados, como é mostrado nas Figura 2, Figura 3, Figura 4. Respectivamente as figuras mostram: a página de cadastro, formulário de cadastro de usuário e formulário para encontrar as coordenadas. Essas páginas estão em uso atualmente, mas mudanças nessas páginas ocorrerão.

Figura 2 – Pagina de Cadastro



Fonte: Elaborado pelo autor.

Figura 3 – Formulário de Cadastro de usuário



Fonte: Elaborado pelo autor

Figura 4 – Formulário para encontrar coordenadas



Fonte: Elaborado pelo autor.

Na parte do desenvolvimento foi utilizado dois editores de código: *Visual Studio Code* e *IntelliJ IDEA*. O *Visual Studio Code* foi utilizado para desenvolver a interface gráfica usando *React JS* e o *IntelliJ IDEA* foi utilizado para desenvolver a Interface de programação de aplicações (API) usando *Spring Boot*. Em ambos os editores de código foi utilizado um plugin chamado *Live Share*, que possibilita o desenvolvimento em conjunto.

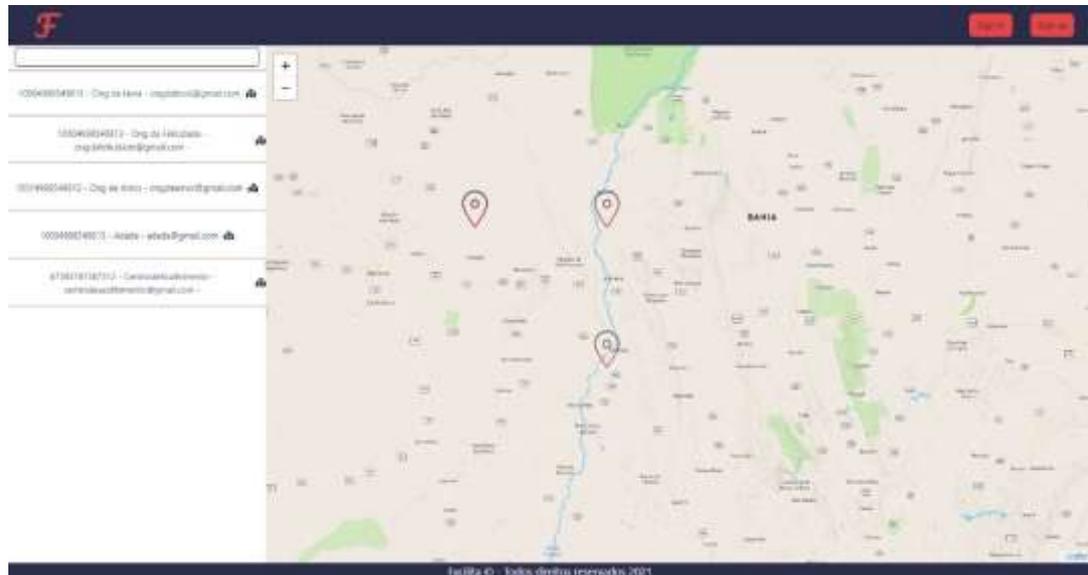
Para fazer a interface gráfica era necessário utilizar o framework *React Js*. Muitos integrantes do grupo não tinham conhecimento sobre esse *framework*, com exceção de um integrante que sabia um pouco. Esse integrante ajudou o restante do grupo.

Durante a criação da interface iria ser utilizado duas APIs do *Google*, a *Maps JavaScript API* e a *Google Places*. A *Maps JavaScript API* seria utilizada para mostrar o mapa na tela e a *Google Places* seria utilizada para encontrar as instituições que estavam próximas do usuário, mesmo sem o cadastro das instituições em nosso sistema, porém para utilizar a API *Google Places* era necessário pagar por uma quantidade de requisições.

Não sendo possível consumir os dados das API do *Google* gratuitamente, foi decidido que apenas seria possível listar as instituições cadastradas que estão próximas do usuário para mostrar o mapa na tela foi decidido usar outra API, a *Leaflet* que é gratuita e consegue suprir as necessidades do projeto.

Outra dificuldade do projeto era conseguir as coordenadas do usuário utilizando o endereço como parâmetro, foi procurado alguns serviços que faziam isso, mas não foi encontrado nenhum serviço gratuito e com qualidade. Assim os requisitos teriam que ser alterados novamente, o usuário teria que passar as coordenadas no cadastro. Para facilitar o usuário nessa parte é disponibilizado um mapa em que o usuário marca onde é a localização atual e essas coordenadas são salvas, como é mostrado na Figura 5.

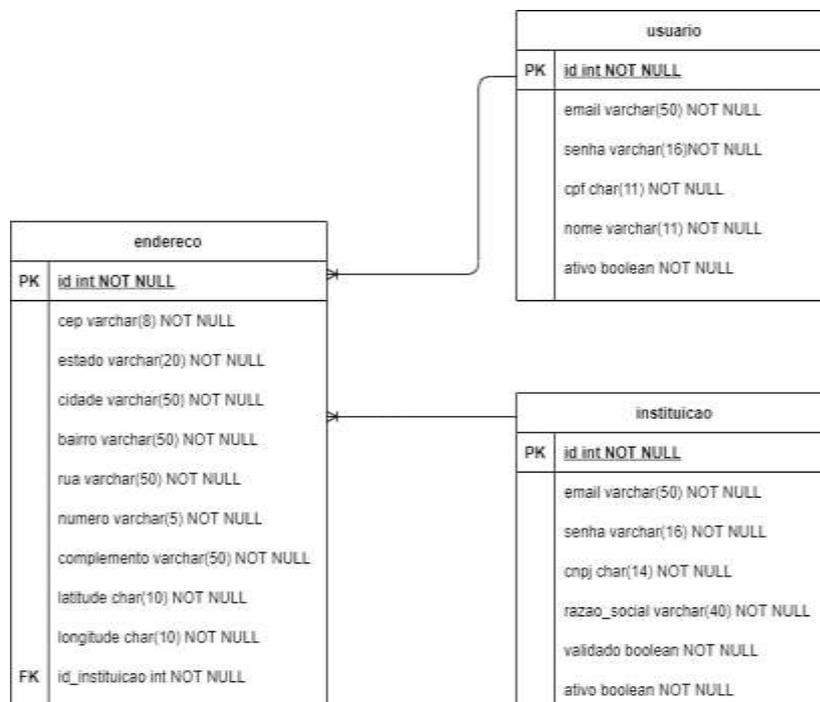
Figura 5 – Mapa Dentro do Cadastro de ONG.



Fonte: Elaborado pelo autor.

Para fazer a API, primeiramente era necessário fazer a modelagem do banco, foi bastante simples, pois não era necessário armazenar muitos dados, a Figura 6 mostra a modelagem do banco em um diagrama UML. Como SGBD foi utilizado o *PostgreSQL*, pois todos os integrantes estavam habituados com ele.

Figura 6 – Modelagem do Banco



Fonte: Elaborado pelo Autor

Após o banco ter sido criado, houve uma tentativa de criar a API, mas ninguém no grupo sabia nem ao menos começar, com isso, uma pausa de 1 semana aconteceu, para os integrantes estudarem e depois se reunirem e desenvolver a API. Depois da pausa para os estudos, quase tudo ocorreu perfeitamente, o *framework Spring* deixa tudo muito fácil de ser desenvolvido. Também foi usado o *software insomnia* para testar a API, mandando requisições de todos os tipos.

Os poucos problemas que ocorreram foram o de não entender muito bem do *framework* no começo e de fazer um relacionamento um para muitos. Houve uma tentativa de relacionar uma instituição a vários endereços, mas sempre ocorria algum erro, depois de uma série de pesquisas foi descoberto que havia a necessidade de usar uma notação específica, mas quando esse problema foi resolvido surgiu outro, quando uma requisição de consultar era efetuada a API retornava dados infinitos, pois instituição tem endereço e endereço tem instituição, sendo assim a API retornava dados infinitos. Para resolver esse problema foi preciso utilizar um objeto de transferência de dados (DTO), assim a API apenas retorna o que era preciso.

Nesse ponto o MVP, já havia sido alcançado, o usuário e a instituição podem cadastrar e o usuário pode visualizar as instituições cadastradas que estão em volta. Tem mais um requisito que o grupo queria implementar, mas não deu tempo que era o do usuário conseguir realizar doações em dinheiro para uma instituição. Seria utilizado o serviço de pagamento do pagar.me, pois é disponibilizado uma API de teste gratuitamente.

4. Conclusão

Foi concluído pelo grupo a importância do projeto facilita no âmbito social, principalmente em tempos difíceis. Na qual muitas ONGs e entidades de caridade estão passando por muita dificuldade, falta de dinheiro, alimento, entre outros recursos. É esperado pelo grupo que a plataforma *Web Facilita*, seja de grande ajuda e um excelente apoio para essas instituições que passam por necessidades no país.

Na questão técnica este projeto foi muito interessante para o grupo, pois poderão trabalhar com tecnologias que nunca tinham ouvido falar, por exemplo: criar uma *API REST* com *Spring*, trabalhar com APIs externas e com análise e projeto de *software* com a elaboração de diagramas.

5. Referências

- [1] LARMAN, Craig. Utilizando UML e Padrões. **Uma Introdução à Análise e ao Projeto Orientados a Objetos e ao Desenvolvimento Iterativo**. 3ª ed. Bookman, 2016.
- [2] OMG. **OMG Unified Modeling Language: OMG UML**. Versão 2.5.1. OMG, 2017.
- [3] BEZERRA, Eduardo. **Princípios de análise e projeto de sistemas com UML**. Rio de Janeiro, RJ, Brasil: Elsevier, 2007. 369 p.
- [4] LUCKOW, Décio Heinzemann; et al. **Programação Java para a Web. 2ª edição**. 1. ed. São Paulo: Novatec, 2015. 640 p.
- [5] FACEBOOK IN. **React – uma biblioteca JavaScript para criar interfaces de usuários**. 2021. Disponível em: <https://pt-br.reactjs.org>. Acesso em: 13 jun. 2021.
- [6] THE POSTGRESQL GLOBAL DEVELOPMENT GROUP. **PostgreSQL: The world's most advanced open source database**. 2021. Disponível em: <https://www.postgresql.org>. Acesso em: 13 jun. 2021.
- [7] ORACLE. **Overview Java Platform SE 8**. 2021. Disponível em: <https://docs.oracle.com/javase/8/docs/api>. Acesso em: 13 jun. 2021.