



UNIVERSIDADE FEDERAL DO CEARÁ
CURSO DE GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO

FRANCISCO VILMAR RODRIGUES DA SILVA

UTILIZANDO REACTJS PARA DESENVOLVIMENTO DE UMA INTERFACE
GRÁFICA PARA O SISTEMA WEB CEUA

SOBRAL

2023

FRANCISCO VILMAR RODRIGUES DA SILVA

UTILIZANDO REACTJS PARA DESENVOLVIMENTO DE UMA INTERFACE GRÁFICA
PARA O SISTEMA WEB CEUA

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia da Computação da Universidade Federal do Ceará, como requisito à obtenção do grau de bacharel em Engenharia da Computação.

Orientador: Prof. Dr. Iális Cavalcante de Paula Júnior.

SOBRAL

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

S58u Silva, Francisco Vilmar Rodrigues da.
Utilizando reactjs para desenvolvimento de uma interface gráfica para o sistema web
ceua / Francisco Vilmar Rodrigues da Silva. – 2023.
39 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus
de Sobral, Curso de Engenharia da Computação, Sobral, 2023.

Orientação: Prof. Dr. Iális Cavalcante de Paula Júnior.

1. Reactjs. 2. Interface gráfica. 3. CEUA. I. Título.

CDD 621.39

FRANCISCO VILMAR RODRIGUES DA SILVA

UTILIZANDO REACTJS PARA DESENVOLVIMENTO DE UMA INTERFACE GRÁFICA
PARA O SISTEMA WEB CEUA

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia da Computação da Universidade Federal do Ceará, como requisito à obtenção do grau de bacharel em Engenharia da Computação.

Aprovada em: 14/12/23

BANCA EXAMINADORA

Prof. Dr. Iális Cavalcante de Paula
Júnior (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Antonio Josefran de Oliveira Bastos
(Membro da banca)
Universidade Federal do Ceará (UFC)

Iago Magalhães de Mesquita (Membro da banca)
Tecnólogo em Mecatrônica

AGRADECIMENTOS

Primeiramente agradeço a Deus por ter trazido conforto e sabedoria.

Agradeço aos meus pais, **Francimar Luiz da Silva** e **Maria das Dores Rodrigues** por sempre terem me apoiado nas minhas decisões e me incentivaram a seguir meus sonhos.

Agradeço a meus irmãos, **Niltomar Rodrigues da Silva**, **Tamires Rodrigues da Silva** e **Mailton Rodrigues da Silva** e meu sobrinho **Heitor Victor da Silva Coutinho** por terem me incentivado e me auxiliado ao longo desses anos.

Agradeço aos meus amigos por terem feito parte dessa jornada, fazendo meus dias na universidade mais alegres e confortáveis.

Ao **Prof. Dr. Iális Cavalcante de Paula Júnior**, por ter me auxiliado na elaboração deste trabalho. E aos demais professores por terem contribuído com o desenvolvimento do meu conhecimento ao longo dos anos.

RESUMO

O processo de transformar a captação e armazenamento de dados dos processos da Comissão de Ética no Uso de Animais(CEUA) em um projeto web é de extrema necessidade para trazer melhorias para a equipe que gerencia, organiza e trata da documentação e solicitações que aparecem constantemente. As solicitações requerem a concentração de documentos e papeladas, o que precisa ser tratado de forma bem organizada para tornar o processo menos complexo. Tendo isso em mente, foi proposto o desenvolvimento de uma interface intuitiva e de fácil utilização, atendendo às demandas e feedbacks dos usuários, para que se possa ser acessada de forma organizada, tendo uma perspectiva quanto às etapas referentes ao desenvolvimento das pesquisas.

Palavras-chave: *Reactjs*; Interface gráfica; CEUA

ABSTRACT

The process of transforming the capture and storage of data from the Commission on Ethics in the Use of Animals (CEUA) processes into a web project is extremely necessary to bring improvements to the team that manages, organizes and handles the documentation and requests that constantly appear. Requests require a concentration of documents and paperwork, which needs to be handled in a well-organized way to make the process less complex. With this in mind, it was proposed to develop an intuitive and easy-to-use interface, meeting users' demands and feedback, so that it can be accessed in an organized way, having a perspective on the stages related to the development of research.

Keywords: Reactjs; Graphic interface; CEUA

LISTA DE ILUSTRAÇÕES

Figura 1 – Etapas de avaliações de projetos no sistema	13
Figura 2 – Lógica de exibição de telas	14
Figura 3 – Arquitetura da aplicação	20
Figura 4 – Requisição de Login	22
Figura 5 – Requisição para novo projeto	23
Figura 6 – Tabelas do banco de dados - níveis de usuários	24
Figura 7 – Tela de <i>login</i>	25
Figura 8 – Configurar dados de usuario logado	26
Figura 9 – Usuários cadastrados	26
Figura 10 – Cadastrar usuários	27
Figura 11 – Formulário de dados iniciais	28
Figura 12 – Formulário de colaboradores	28
Figura 13 – Formulário de modelo animal	29
Figura 14 – Formulário de animais	29
Figura 15 – Formulário de procedimentos experimentais	30
Figura 16 – Formulário de fármacos	30
Figura 17 – Formulário de pós-operatório	31
Figura 18 – Formulário de anexos	31
Figura 19 – Tabela de projetos	32
Figura 20 – Trecho de um formulário em revisão pelo parecerista	32

LISTA DE ABREVIATURAS E SIGLAS

CEUA	Comissão de Ética de Uso de Animais
UI	<i>User Interface</i>
DOM	<i>Document Object Model</i>
HTML	<i>Hyper Text Markup Language</i>
CSS	<i>Cascading Style Sheet</i>
JSON	<i>JavaScript Object Notation</i>
API	<i>Application Programming Interface</i>

SUMÁRIO

1	INTRODUÇÃO	10
1.1	JUSTIFICATIVA	11
1.2	OBJETIVOS	11
<i>1.2.1</i>	<i>Objetivo geral</i>	<i>11</i>
<i>1.2.2</i>	<i>Objetivos específicos</i>	<i>11</i>
2	FUNDAMENTAÇÃO TEÓRICA	13
2.1	Níveis de acesso do sistema	13
2.2	Fluxo de projetos na aplicação	13
2.3	Lógica de exibição através das permissões de usuário	14
2.4	Tecnologias e recursos computacionais utilizados	15
<i>2.4.1</i>	<i>Reactjs</i>	<i>15</i>
<i>2.4.2</i>	<i>NextJs</i>	<i>15</i>
<i>2.4.3</i>	<i>TypeScript</i>	<i>15</i>
2.5	Recursos de desenvolvimento	16
<i>2.5.1</i>	<i>Material UI</i>	<i>16</i>
<i>2.5.2</i>	<i>Tailwind CSS</i>	<i>16</i>
<i>2.5.3</i>	<i>Formik</i>	<i>17</i>
<i>2.5.4</i>	<i>JSON</i>	<i>17</i>
<i>2.5.5</i>	<i>Visual studio code</i>	<i>18</i>
3	METODOLOGIA	19
3.1	Identificação do problema	19
3.2	Validação do problema	19
3.3	Etapas de desenvolvimento	19
<i>3.3.1</i>	<i>Análise do código antigo</i>	<i>19</i>
<i>3.3.2</i>	<i>Escolha das bibliotecas</i>	<i>19</i>
<i>3.3.3</i>	<i>Coleta de requisitos</i>	<i>20</i>
<i>3.3.4</i>	<i>Processo de conversão de arquivos do projeto</i>	<i>20</i>
<i>3.3.5</i>	<i>Atualização do front-end</i>	<i>20</i>
3.4	Arquitetura da aplicação	20
3.5	Estrutura de conexões	21

3.5.1	<i>Formato de dados em JSON - Login</i>	22
3.5.2	<i>Formato de dados em JSON - Formulário de novo projeto</i>	23
3.5.3	<i>Tabelas desenvolvidas para a aplicação</i>	24
4	RESULTADOS	25
4.1	Tela de login	25
4.2	Gerenciamento do perfil do usuário	26
4.3	Telas para cadastro e gerenciamento de usuários	26
4.4	Formulários de submissão de projetos	27
4.5	Tabela para gerenciamento de projetos	31
4.6	Formulários para correção e revisão	32
5	CONCLUSÕES E TRABALHOS FUTUROS	33
	REFERÊNCIAS	34

1 INTRODUÇÃO

Recentemente, o processo de transformação digital vem sendo cada vez mais presente em diferentes áreas e setores da sociedade, seja em atividades cotidianas, entretenimento e também na área acadêmica. Esse processo de modernização vem melhorando o fluxo de realizações de atividades, bem como tornando o processo de documentação e registro em uma forma cada vez mais organizada.

Na área de pesquisa científica, esse avanço tecnológico também se mostra essencial, como é o caso de projetos relacionados ao uso de animais, uma vez que o registro dos materiais relacionados à pesquisa, em grande parte se trata de forma manual, envolvendo documentação física e um processo burocrático desenvolvido em etapas. Esse processo manual pode gerar confusões, uma vez que o processo não seja padronizado, pode resultar em dificuldade para o acesso dessas informações, podendo atrasar o processo de análise bem como a aprovação ou rejeição dos projetos.

O processo de modernização surge como uma solução satisfatória para resolver essas limitações, organizando e otimizando os procedimentos de registro e armazenamento centralizado de informações que envolvam as pesquisas realizadas. A utilização de ferramentas digitais, como um sistema web, pode ser aproveitada de forma eficiente para que se tenha um controle e acompanhamento das etapas que permeiam o desenvolvimento de um projeto.

A utilização de um sistema web para o gerenciamento de projetos de pesquisas científicas traz várias vantagens. A primeira é ter um local em que todas as informações referentes à pesquisa estarão concentradas, podendo assim ser acessada por todos os participantes envolvidos no projeto de pesquisa de forma a ter uma visão detalhada das etapas, possibilitando um aprimoramento no desenvolvimento e avaliações. Além de possibilitar a padronização de etapas, permitindo que a informação seja acessada de forma clara e instantânea de qualquer lugar.

Para o desenvolvimento de um sistema, é necessário a utilização de ferramentas que possuem uma fácil utilização e adaptação, por isso é importante ter uma plataforma desenvolvida de forma eficiente e que possua uma interface de fácil utilização. A utilização de um framework (ferramenta de abstração de códigos) e bibliotecas podem ser usadas como aliadas, principalmente no quesito agilidade no tempo de produção. A biblioteca *Reactjs* é bastante utilizada no desenvolvimento de interface para usuários, por ser bastante popular, as comunidades de desenvolvimento estão em constantes atualizações, trazendo novidades e soluções.

1.1 JUSTIFICATIVA

Atualmente, a coleta de informações relacionadas às pesquisas está sendo feito utilizando o serviço de armazenamento em nuvem da *Google*, o *Google Drive*. No entanto, esta é uma plataforma de uso geral e que, por não ser padronizada, pode acabar trazendo confusões quanto ao armazenamento e disposição das informações. Além disso, por ser uma solução adaptada para esse propósito, poderá trazer possíveis inconvenientes posteriormente, como limitação de armazenamento e até mesmo perda de informações, levando a CEUA a ter perdas e prejuízos relacionados aos seus dados.

Com isso, o processo de desenvolvimento de uma plataforma sob medida e própria, trazendo soluções que se encaixem aos problemas particulares da CEUA, havia sido iniciado. No entanto, a versão do sistema desenvolvido não possuía uma aplicação funcional, apenas uma parte de interface foi entregue, não atendendo as necessidades reais atuais.

Dito isso, esse trabalho é de grande importância para que a plataforma possa ter continuidade e o funcionamento seja estabelecido de forma a atender as necessidades, trazendo uma solução eficiente para que os usuários possam utilizar de forma satisfatória com uma interface intuitiva e de fácil compreensão.

1.2 OBJETIVOS

1.2.1 *Objetivo geral*

Dessa forma, este trabalho tem como objetivo geral criar uma interface *front-end* para a plataforma WEBCEUA, seguindo uma aparência amigável e intuitiva aos usuários para que possam registrar e acompanhar o andamento e as correções dos projetos cadastrados.

1.2.2 *Objetivos específicos*

Através das análises feitas no projeto inicial existente, avaliações de diferentes recursos de *Reactjs*, dos *feedbacks* e interações junto aos responsáveis pela CEUA e em conjunto com o *back-end* foram feitas propostas para aplicações no sistema:

- Desenvolver uma tela inicial de *login*;
- Definir logicamente níveis de acesso;
- Cadastrar pesquisadores, parecerista e coordenadores;

- Desenvolver Formulários;
- Coletar respostas objetivas e subjetivas através dos formulários;
- Coletar arquivos através dos formulários;
- Visualizar e gerenciar projetos cadastrados(de acordo com nível de permissão), bem como a etapa de avaliação que o mesmo se encontra;

2 FUNDAMENTAÇÃO TEÓRICA

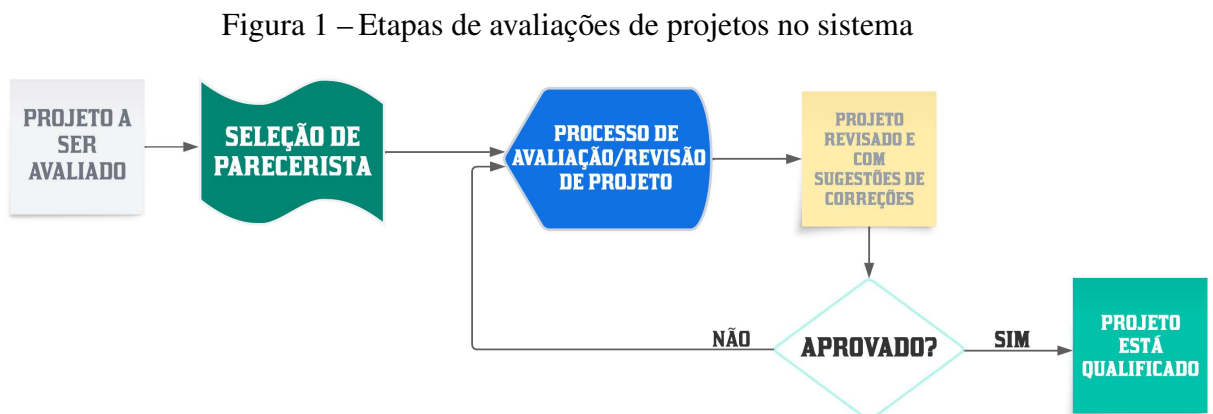
Nessa seção, iremos descrever os conceitos e características para o desenvolvimento da aplicação, bem como as tecnologias e os recursos escolhidos. Os conceitos, definições e características descritas ao longo dessa seção podem ser consideradas um resumo de um apurado de informações coletadas em diferentes fontes, como a própria documentação da tecnologia e também fragmentos de textos em sites voltados para a área de desenvolvimento.

2.1 Níveis de acesso do sistema

Para seguir uma funcionalidade de gerenciamento e uso do sistema, foram definidos níveis de permissões para os usuários, onde cada nível de permissão possui privilégios de acesso. Os níveis foram classificados da seguinte forma:

- Pesquisador: usuário com nível de permissão para enviar projetos de pesquisas com propósito de serem avaliados pela a CEUA;
- Parecerista: usuário com nível de permissão para acessar e dar parecer sobre os projetos de pesquisas na qual lhe foram alocados;
- Coordenador/CEUA: usuário com nível de permissão absoluta, podendo fazer as principais configurações do sistema, cadastrar e manipular usuários cadastrados, fazer encaminhamentos dos projetos para os pareceristas, gerar certificados dos projetos e etc;

2.2 Fluxo de projetos na aplicação



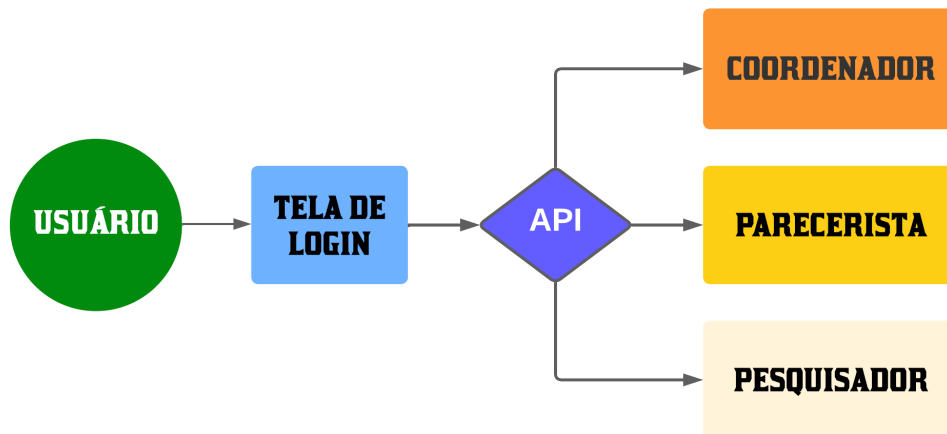
Fonte: elaborada pelo autor(2023).

O Fluxograma mostrado acima, descreve de forma resumida o processo em que um projeto passa ao ser examinado pela CEUA. As etapas obedecem a seguinte ordem:

- Um pesquisador entra com um projeto para ser avaliado e revisado;
- O sistema apresenta os projetos recém cadastrados para a coordenação, essa irá ficar responsável por encaminhar esses projetos para os pareceristas (pessoas que avaliaram o projeto);
- O projeto começa a ser analisado. O processo de análise pode ser repetido por algumas vezes, uma vez que o projeto pode conter informações insuficientes ou com alguns dados que possam ter sofrido alterações no decorrer da pesquisa, ou mesmo alguma etapa do projeto que precisou ser refeita;
- Depois de revisado e todas as informações necessárias reunidas, o projeto passará para o processo de aprovação para então receber certificação;

2.3 Lógica de exibição através das permissões de usuário

Figura 2 – Lógica de exibição de telas



Fonte: elaborada pelo autor(2023).

No diagrama exibido acima, podemos ter uma noção de como a API faz o reconhecimento das permissões do usuário que está fazendo o *login* e direciona as telas que para ele foram projetadas.

2.4 Tecnologias e recursos computacionais utilizados

2.4.1 *Reactjs*

O *ReactJs* é uma das bibliotecas de front-end mais populares (AWARI, 2023). Ela é usada para construir Interface de Usuário (UI), oferecendo uma resposta excelente para o usuário adicionar comandos usando um novo método de renderizar sites.

Os componentes dessa ferramenta foram desenvolvidos pelo Facebook. Ela foi lançada em 2013 como uma ferramenta *javascript* de código aberto. Atualmente, ela permanece na frente das suas principais competidoras, como a *Angular* e a *bootstrap*, as duas bibliotecas *javascript* mais bem avaliadas.

Por que usar *react*?

- é fácil de usar;
- suporte a componente reusável em *javascript*;
- componente fácil de escrever;
- melhor desempenho com virtual *Document Object Model* (DOM) - Modelo de Documento por Objetos, usado internamente nos navegadores;

2.4.2 *NextJs*

O *NextJs* é um framework *React* criado em 2016 pela *Vercel* e sob a liderança do MIT (GIZELE SILVA, 2023). Hoje ele está disponível em código aberto e pode contribuir para a criação de interfaces do usuário junto com *React*.

Entre as principais características do *NextJs*, temos:

- Renderização estática e pelo lado do servidor;
- Suporte também ao *Typescript* (permitindo dar mais segurança a aplicação desenvolvida);
- Serviço de tratamento de rotas;
- Permite a instalação de *plugins*;
- Permite apenas o carregamento dos elementos necessário;

2.4.3 *TypeScript*

TypeScript é um superconjunto de *JavaScript*, ou seja, um conjunto de ferramentas e formas mais eficientes de escrever código *JavaScript*, adicionando recursos que não estão

presentes de maneira nativa na linguagem (DIEGO MELO, 2020). Outra diferença entre o *JavaScript* e o *TypeScript* é a tipagem de dados de forma estática no *TypeScript*, enquanto que o *JavaScript* é uma linguagem dinâmica. Entre as principais características do *TypeScript*, temos:

- Potencial de detecção de erros durante o desenvolvimento de projetos devido a tipagem estática;
- Possibilidade de incluir inteligência (*IntelliSense*) no ambiente de desenvolvimento, tornando o ambiente mais ágil e seguro enquanto o usuário está construindo o código da aplicação;
- Ecossistema maduro: possui um ecossistema de bibliotecas e frameworks crescente, oferecendo suporte a várias tecnologias populares;

2.5 Recursos de desenvolvimento

2.5.1 *Material UI*

A *Material UI* é uma biblioteca de componentes *ReactJS* de código aberto que implementa o design do *Google* (MUI, 2023). Ele inclui uma coleção abrangente de componentes pré-construídos que estão prontos para uso na produção imediata.

Características:

- A sua forma de uso baseia-se em utilização de componentes autossuficientes, não dependendo assim de uma estilização utilizando *CSS* em uma página separada;
- Ele segue diretrizes, componentes e ferramentas de *material design* do *Google* para tornar os aplicativos da web mais rápidos;

2.5.2 *Tailwind CSS*

É um framework *CSS* visando a estilização, fornecendo várias classes utilitárias, opinativas e de propósito único, que podem ser usadas diretamente dentro do bloco de código para projetar características a um elemento. O propósito geral da criação desse framework é aumentar a velocidade para o desenvolvimento, agilizando os processos de criação de seus web sites.

Vantagens:

- Processo de desenvolvimento mais rápido;
- Ajuda você a praticar mais o seu *CSS*, pois as utilidades são similares;

- Todas as utilidades e componentes são facilmente personalizáveis;
- O tamanho total do arquivo para a produção é geralmente pequeno;
- Fácil de aprender se você já conhece o CSS;
- Boa documentação para o aprendizado;

Desvantagens:

- Sua marcação pode parecer desorganizada para grandes projetos, porque todos os estilos estão nos arquivos HTML;
- Não é fácil de aprender se você não entende bem o CSS;
- Você é forçado a construir tudo do zero, incluindo seus elementos de entrada. Quando você instala o Tailwind CSS, ele remove todos os estilos padrão de CSS;

2.5.3 *Formik*

O *Formik* é uma biblioteca de código aberto popular que fornece os componentes *Form*, *Field* e *ErrorMessage* para que possamos construir formulários, campos de formulários e mensagens de erro de exibição, respectivamente.

Vantagens:

- Simples de usar;
- Fácil de obter os valores dentro e fora do estado do formulário;
- Super leve;
- Fornece alguns mecanismos que nos permitem validar os campos do formulário, rastrear o estado de envio do formulário e lidar com erros;

Desvantagem:

- Controle do formulário acontece fora de seu escopo (porém é intencional);

2.5.4 *JSON*

É um formato de arquivo de texto leve, compacto, utilizado para estruturar dados. É comumente usado para transmitir dados em aplicações web.

A linguagem é bastante usada, por oferecer simplicidade, legibilidade, portabilidade e suporte amplo, o que torna uma escolha assertiva para a troca de informações na web e em outros ambientes. Além de ser usada para transferências de dados, o *JSON* também é utilizado como uma "linguagem de configuração", na qual são guardados parâmetros e outras opções para configurar ou personalizar programas e aplicativos, seja no *desktop*, *mobile* ou servidores.

2.5.5 *Visual studio code*

O *visual Studio Code* é um editor de código aberto desenvolvido pela *Microsoft* disponível para os sistemas operacionais mais conhecidos, leve e destinado a diferentes linguagens de programação.

Entre as vantagens, podemos citar que é um programa gratuito de código aberto, utilizado em multiplataforma e por linguagens de multiníveis. Possui também as características de ser altamente personalizável, possui diversas extensões e atalhos, além de possuir a possibilidade de depuração de códigos.

3 METODOLOGIA

Ao longo desta seção, serão descritos e detalhados os passos que resultaram no desenvolvimento da aplicação frontend para o sistema WEBCEUA.

3.1 Identificação do problema

O processo para identificar os principais problemas, o funcionamento da CEUA e as perspectivas de evolução para o projeto foi feito em diferentes momentos. O principal meio para nos trazer esclarecimentos se deu através de reuniões constantes pelo *Google Meet*, onde os responsáveis pelo projeto nos trouxeram os principais desafios, entre eles o gerenciamento dos projetos que, no momento, acontecia de forma manual através de ferramentas como *Gmail* e *Google Drive*.

3.2 Validação do problema

Com a identificação do problema, a equipe do desenvolvimento passou a procurar soluções para trazer a implementação e aperfeiçoamento do sistema para que pudesse suprir as necessidades pendentes para a CEUA.

3.3 Etapas de desenvolvimento

3.3.1 *Análise do código antigo*

Inicialmente foi feito uma análise do código do projeto já existente, mas que não estava em funcionamento, buscando entender o conceito inicial e procurando uma forma reaproveitar a estrutura da melhor maneira possível com a finalidade de desenvolver uma solução em um período razoável.

3.3.2 *Escolha das bibliotecas*

Buscando uma maneira de desenvolver uma interface intuitiva e de fácil manipulação, foram escolhidas as bibliotecas *ReactJs* e *Material UI* devido a sua popularidade e fóruns com movimentações constantes, trazendo sugestões e soluções de rápido desenvolvimento e aparência amigável.

3.3.3 Coleta de requisitos

Através de reuniões semanais, os requisitos foram levantados e ajustados junto com os responsáveis pelo projeto, fazendo com que a plataforma fosse desenvolvida de forma a suprir as demandas existentes.

3.3.4 Processo de conversão de arquivos do projeto

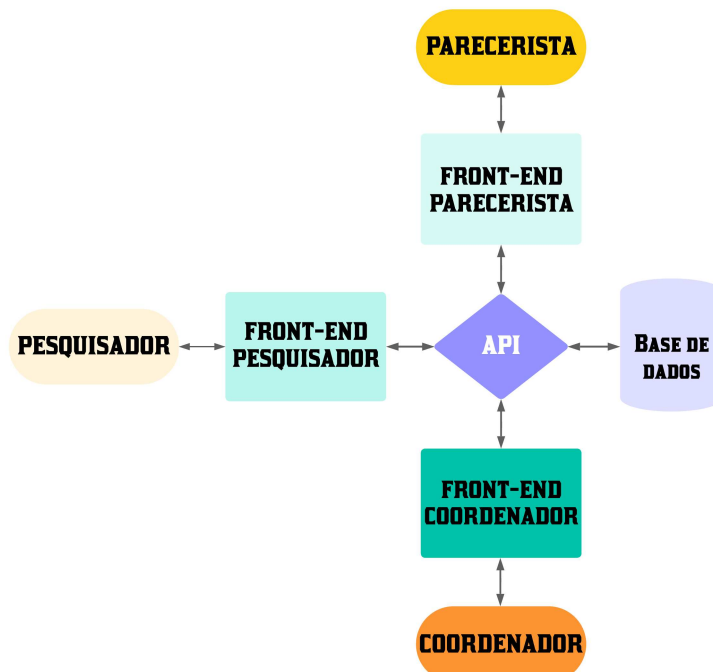
Nessa etapa, optamos pela conversão de alguns arquivos *JavaScript* para *TypeScript*, dentre os motivos da escolha dessa ferramenta está a questão da segurança e manipulação de dados.

3.3.5 Atualização do front-end

Através dos requisitos levantados com os responsáveis, foram feitas as implementações das melhorias e correções sugeridas no *front-end*, para que fosse feita uma comunicação eficaz com o *back-end*, trazendo resultados satisfatórios.

3.4 Arquitetura da aplicação

Figura 3 – Arquitetura da aplicação



Fonte: elaborada pelo autor(2023).

Podemos ver acima que a ideia é termos uma interface de *front-end* de acordo com os níveis de permissão do usuário para que possa consumir a aplicação *back-end* que está sendo desenvolvida em conjunto. Vale dizer que essas ações seguirão algumas regras de negócio, por exemplo, um usuário pesquisador não pode realizar uma ação que um usuário coordenador pode.

A ideia para criar os *front-ends* separados, é para poder organizar de forma mais detalhadas os processos que envolvem o desenvolvimento dos projetos de pesquisa, levando em consideração a submissão dos formulários bem como os arquivos necessários.

O desenvolvimento do *front-end* para essas telas foi feito basicamente com *reactjs* e *typescript*, além de algumas bibliotecas como *Formik*, *Material UI*, *Tailwind* que nos facilitou no desenvolvimento dos principais componentes utilizados para apresentações de tabelas e formulários.

A elaboração de componentes com *reactJs*, nos permite que, uma vez construído, esse mesmo componente possa ser utilizado quantas vezes o desenvolvedor precisar, facilitando na parte de manutenção e atualização na aparência, além de reduzir a tempo gasto no desenvolvimento de elementos visuais que irão se repetir em diferentes telas.

A utilização das bibliotecas *Material UI* e *Tailwind*, nos permite desenvolver a parte da aparência e layout de forma rápida e intuitiva devido trazer em sua própria construção estilos e formatações padronizadas, mas que através de breve estudo em sua documentação permite ao desenvolvedor fazer a personalização necessária de forma rápida para atender as necessidades visuais e de aparência agradável. Tem também a facilidade de acesso a responsividade de desenvolvimento, permitindo que os elementos visuais desenvolvidos possam se adaptar aos diferentes tamanhos de telas disponíveis no mercado.

Para desenvolver a parte de formulários, a equipe optou pela utilização do *Formik* e *TypeScript*, entre os principais fatores está a parte de autenticação de dados e sobre a segurança na manipulação de dados. Além de serem ferramentas que trazem um certo nível de segurança, elas são bem reconhecidas pela comunidade de desenvolvimento, possibilitando assim que, se bem utilizada e estruturada, a manutenção ocorra de forma intuitiva.

3.5 Estrutura de conexões

Para que ocorresse a comunicação entre o *front-end* e o *back-end* desenvolvido pela equipe, optamos pela utilização do padrão de dados *JSON*, entre alguns fatores está pelo fato de ser um formato extremamente leve e de fácil entendimento tanto no *front-end* quanto no

back-end e manipulações por parte da API.

A seguir, demonstraremos algumas conexões que acontecem entre o *front-end* e *back-end* para que a aplicação retorne, de forma satisfatória, os resultados planejados para o funcionamento da aplicação. As imagens apresentadas abaixo são apenas alguns exemplos dos resultados das requisições por parte do sistema, há algumas requisições, que por retornar uma quantidade de dados tão grande, optamos por não colocar aqui, mas que poderemos ter uma noção do funcionamento na seção de resultados.

3.5.1 Formato de dados em JSON - Login

Figura 4 – Requisição de Login



```
POST .base_url/signin
JSON
Auth
Query
Headers
Docs
1 {
2   "email": " ",
3   "password": " "
4 }
```

Fonte: elaborada pelo autor(2023).

3.5.2 Formato de dados em JSON - Formulário de novo projeto

Figura 5 – Requisição para novo projeto

```

POST ..base_url/newProject

JSON Auth Query Headers 1 Docs

1 {
2   "title": "teste 41",
3   "researcherId": 1,
4   "accommodation": "",
5   "animalsBed": "",
6   "animalsPerCase": "",
7   "conditions": "",
8   "conditionsAlimentation": "",
9   "conditionsExhaustionSN": "",
10  "conditionsFonteAgua": "",
11  "localConditions": "",
12  "conditionsLotacao": "",
13  "species": "beta",
14  "invasiveness": "",
15  "justification": "",
16  "methodsCapture": "",
17  "modifiedNumCQB": "",
18  "modifiedSN": "",
19  "originBioterioLocation": "",
20  "originBioterioName": "",
21  "originBioterioSN": "",
22  "wildAnimal": "",
23  "wildNumRequest": "",
24  "wildOther": "",
25  "wildWhat": "",
26  "wildSN": "",
27  "usageMaterial": "",
28  "usageMaterialProtocol": "",
29  "usageMaterialSN": "",
30  "animals": [],
31  "attachments": [],
32  "drugs": [],
33  "foodJustification": "",
34  "food": "",
35  "sameActSurgery": "",
36  "SurgeryName": "",
37  "surgery": "",
38  "typeSurgery": "",
39  "stressJustification": "",
40  "mobilization": "",
41  "mobilizationTime": "",
42  "mobilizationType": "",
43  "procedureStress": "",
44  "procedureStressType": "",
45  "procedure": "",
46  "typeProcedure": "",
47  "restrictionDuration": "",
48  "restrictionJustification": "teste",
49  "restriction": ""
50 }
51
Beautify JSON

```

Fonte: elaborada pelo autor(2023).

3.5.3 Tabelas desenvolvidas para a aplicação

A seguir serão exibidas na figura 6 algumas tabelas desenvolvidas para a aplicação. Devida a quantidade de tabelas, exibir os relacionamentos entre elas em uma única página fica confuso, por isso optou-se por não exibir os relacionamentos e demonstrar apenas algumas dessas tabelas.

Figura 6 – Tabelas do banco de dados - níveis de usuários

Users	researchers	coordinators	reviewers
Id	Id	Id	Id
name	user	userid	user
institution	project	user	project
departament	userid		userid
lattes			
email			
bond			
fone			
password			
firstAccess			
isValidatedEmail			
createdAt			
updatedAt			
Researcher			
researcherid			
Coordination			
coordinationid			
Reviewer			
reviewerid			

Fonte: elaborada pelo autor(2023).

Na imagem exibida acima, podemos visualizar a estrutura dos usuários e as definições de níveis de usuários, que foi mencionados anteriormente. Para uma melhor apresentação das tabelas e seus relacionamentos, optou-se por colocar o diagrama do banco de dados em anexo (Ver Anexo A - Tabelas desenvolvidas para a aplicação).

4 RESULTADOS

Neste capítulo serão apresentadas as telas desenvolvidas para a plataforma WEB-CEUA, seguindo o propósito de ter uma aparência amigável e intuitiva, onde através delas os usuários conseguissem registrar e acompanhar o andamento e as correções dos projetos cadastrados. Através das ferramentas utilizadas, como *reactjs*, *tailwind* e *material ui*, o desenvolvimento aconteceu de forma clara e organizada, trazendo elementos funcionais.

4.1 Tela de login

Figura 7 – Tela de *login*



Endereço de email

Senha

Esqueceu sua senha?

ENTRAR

Fonte: elaborada pelo autor(2023).

A tela de *login* utiliza os campos de E-mail e Senha para permitir o acesso. Os níveis de permissão de acesso são: Coordenador(CEUA)/Secretaria, Pesquisador e Parecerista.

4.2 Gerenciamento do perfil do usuário

Figura 8 – Configurar dados de usuário logado

Alguns dos dados a seguir são inseridos automaticamente no formulário de submissão de um novo projeto

Meus dados




Foto padrão

Nome

Telefone

Instituição/Unidade

Departamento/Disciplina

Link do currículo lattes

Vínculo com a instituição

Docente/Pesquisador
 Técnico Nível Superior
 Jovem Pes./Pes. Visitante

Salvar alterações

Fonte: elaborada pelo autor(2023).

4.3 Telas para cadastro e gerenciamento de usuários

Figura 9 – Usuários cadastrados

Usuarios
Caio
CEUA

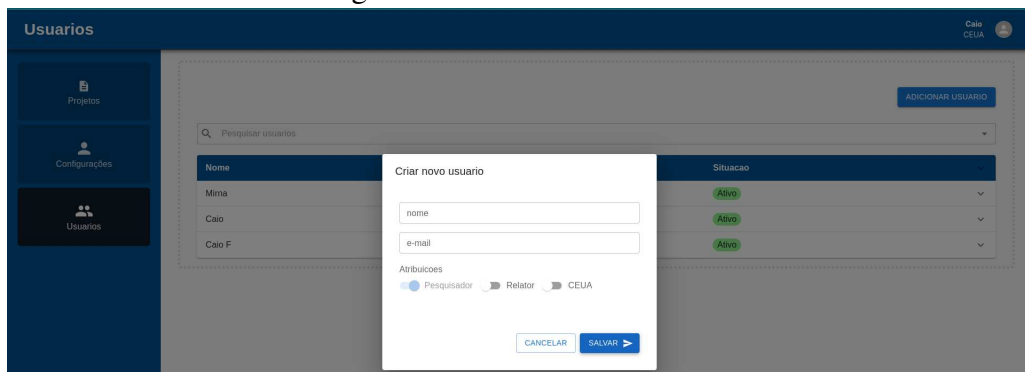
Projetos
Configurações
Usuarios

ADICIONAR USUARIO

Nome	Atribuições	Situação
Mima		Ativo
Caio	Parecerista, CEUA	Ativo
Caio F	Parecerista	Ativo

Fonte: elaborada pelo autor(2023).

Figura 10 – Cadastrar usuários



Fonte: elaborada pelo autor(2023).

4.4 Formulários de submissão de projetos

Os formulários para submissão de projetos no sistema para análise da CEUA está dividido em:

1. Formulário de dados iniciais;
2. Formulário de colaboradores;
3. Formulário de modelo animal;
4. Formulário de animais;
5. Formulário de procedimentos experimentais;
6. Formulário de fármacos;;
7. Formulário de pós-operatório;
8. Formulário de anexos;

Figura 11 – Formulário de dados iniciais

Dados iniciais
Nesta aba do formulário de submissão, serão preenchidos alguns dados sobre a área acadêmica e alguns outros para contato

[Página inicial](#)

Prazo

Início: dd/mm/aaaa Término: dd/mm/aaaa

Título do Projeto

Título em português
O modelo animal deve estar descrito no título

Título em inglês
O modelo animal deve estar descrito no título

Área de conhecimento ([Link externo](#))

Responsável

Nome completo
Mirna

Salvar projeto

Fonte: elaborada pelo autor(2023).

Figura 12 – Formulário de colaboradores

Colaboradores
Preencha os campos caso seja necessário a adição de algum colaborador.

[Página inicial](#)

Colaboradores

Selecione o usuário pelo email

Nome completo:

Instituição/Unidade

Nível acadêmico:

Experiência prévia(em anos):

Experiência Prévia no modelo a ser desenvolvido neste projeto:
 Sim
 Não

Treinamento (especificar):

Telefone:

Salvar projeto

Fonte: elaborada pelo autor(2023).

Figura 13 – Formulário de modelo animal

The screenshot shows a web interface for creating an animal model. At the top, there is a navigation bar with icons for home, users, a selected profile, notifications, a list, a folder, a pencil, and a link. The main content area is titled 'Modelo animal' and contains the following sections:

- Modelo animal** (with a help icon):
 - Text: 'Nesta aba do formulário de submissão serão preenchidos os dados que fazem referência aos animais.'
 - Link: 'Página inicial'
- Modelo animal** (with a help icon):
 - Section: 'Espécie ou grupo taxonômico'
 - Text: 'Nome vulgar, se existir'
 - Section: 'Justificativa do uso do modelo'
 - Text input field
- Procedência** (with a help icon):
 - Section: 'Biotério'
 - Radio buttons: Sim Não
 - Section: 'Animal silvestre'
 - Radio buttons: Sim Não
 - Section: 'O animal é geneticamente modificado?'
 - Radio buttons: Sim Não
- Métodos de Captura** (with a help icon):
 - Text input field

A 'Salvar projeto' button is located at the bottom right of the form.

Fonte: elaborada pelo autor(2023).

Figura 14 – Formulário de animais

The screenshot shows a web interface for managing animals. At the top, there is a navigation bar with icons for home, users, a selected profile, notifications, a list, a folder, a pencil, and a link. The main content area is titled 'Animais' and contains the following sections:

- Animais** (with a help icon):
 - Text: 'Preencha os campos com os animais correspondentes ao projeto.'
 - Link: 'Página inicial'
- Tipo e Características** (with a help icon):
 - Section: 'Seleciono o animal'
 - Dropdown menu
 - Section: 'Linhagem'
 - Text input field
 - Section: 'Idade'
 - Text input field with value '0'
 - Section: 'Peso aproximado'
 - Text input field with value '0'
 - Section: 'Quantidade fêmea(s)'
 - Text input field with value '0'
 - Section: 'Quantidade macho(s)'
 - Text input field with value '0'
 - Section: 'Subtotal (M+F)'
 - Text input field with value '0'

A 'Salvar projeto' button is located at the bottom right of the form.

Fonte: elaborada pelo autor(2023).

Figura 15 – Formulário de procedimentos experimentais

Procedimentos experimentais
Nesta aba do formulário de submissão, serão preenchidos os dados que fazem referência a experiências feitas aos animais

[Página inicial](#)

Procedimentos Experimentais
Procedimentos experimentais do projeto
 Sim Não

Estresse
Realização de procedimento de estresse?
 Sim Não

Imobilização
Imobilização do animal
 Sim Não

Condições Alimentares
Jejum
 Sim Não
Restrição hídrica
 Sim Não

Salvar projeto

Fonte: elaborada pelo autor(2023).

Figura 16 – Formulário de fármacos

Fármacos
Preencha os campos caso seja necessário a utilização de algum fármaco.

[Página inicial](#)

Tipo e Características
Selecione o tipo de fármaco

Fármaco

Dose (UI ou mg/kg)

Via de administração

Frequência

Justificativa em caso de não uso

Remover fármaco

Salvar projeto

Fonte: elaborada pelo autor(2023).

Figura 17 – Formulário de pós-operatório

Fonte: elaborada pelo autor(2023).

Figura 18 – Formulário de anexos

Fonte: elaborada pelo autor(2023).

4.5 Tabela para gerenciamento de projetos

Logo após ser submetido um projeto por parte dos pesquisadores, cabe a coordenadora da CEUA escolher um parecerista(uma pessoa responsável por fazer uma análise minuciosa das informações postadas pelo pesquisador). A seguir, esse parecerista fará uma revisão e devolverá ao pesquisador um parecer constando a análise, bem como os dados pendentes para conclusão e aprovação do projeto.

Figura 19 – Tabela de projetos

Titulo	Relator	Situacao
formulario 05 editado		EmAnalise
projeto teste		Submetido
projeto animais teste		Submetido

Fonte: elaborada pelo autor(2023).

Na imagem acima, podemos visualizar a disposição de projetos submetidos, com a possibilidade de realizar o filtro de acordo com o status na qual os projetos se encontram. Esse é um dos principais intuitos do projeto, permitir que a CEUA possa acompanhar e administrar os projetos nas suas etapas até a sua aprovação.

4.6 Formulários para correção e revisão

Para que os trabalhos de pesquisas submetidos para análise por parte da CEUA passasse por revisões, os formulários foram, de certa forma, reaproveitadas utilizando campos adicionais para a inserção de observações por parte dos corretores pareceristas. Os formulários para correção são praticamente os mesmos.

Figura 20 – Trecho de um formulário em revisão pelo parecerista

Experiência do pesquisador responsável no modelo animal proposto:

Adequado?

Sim Não

Comentários:

Fonte: elaborada pelo autor(2023).

Na imagem acima, podemos observar o trecho de um formulário onde o parecerista poderá colocar a sua opinião acerca dos dados preenchidos pelo pesquisador.

5 CONCLUSÕES E TRABALHOS FUTUROS

Ao longo do desenvolvimento deste trabalho, buscamos trazer soluções para o projeto WEBCEUA, trazendo uma interface de simples entendimento e com recursos funcionais através da utilização de tecnologias atuais no mercado. A utilização de bibliotecas, *frameworks* e outras tecnologias (como o *ReactJs*) foram nossas aliadas para entregar nossas soluções desenvolvidas em curto tempo, e isso aconteceu principalmente através de amplos meios de consultas para nos auxiliarmos na implementação em tempo hábil.

Diante do exposto neste trabalho, podemos chegar em um resultado satisfatório mediante as necessidades propostas como requisitos por parte dos integrantes do projeto CEUA. À medida em que avançou-se no desenvolvimento, podemos criar soluções viáveis para torná o projeto funcional, no entanto, novas perspectivas e melhorias poderão ser vistas e aperfeiçoadas através dos constantes feedbacks.

O processo de desenvolvimento, apesar de ter um início, meio e fim, também passa por constantes melhorias e adaptações no intuito de torná o software ou aplicação cada vez mais eficaz e seguro utilizando os meios atuais do mercado. Com isso em mente, entregou-se o software com os principais recursos funcionais, mas com espaço para aprimoramentos futuros.

Como trabalhos futuros, podemos ter em mente trazer melhorias na parte visual e também na abordagem no funcionamentos e interações do usuários para com as telas e funcionalidades desenvolvidas. Outras funcionalidades também poderão ser inseridas, tornando o sistema mais complexo ao ponto de torná-lo sólido permitindo a venda de licenças para outras instituições e empresas.

REFERÊNCIAS

EDUARDO RABELO. **Usando esm module(esm) em node.js: um guia prático - parte 1. 2023.** Disponível em: <https://dev.to/oieduardorabelo/usando-es-modules-esm-em-node-js-um-guia-pratico-part-1-3bjp>. Acesso: 12 de Nov. 2023.

REACT. **Getting Started.** Disponível em: <https://legacy.reactjs.org/docs/getting-started.html>. Acesso: 27 de Nov. 2023.

AWARI. **As Melhores Bibliotecas De Front-End Em Javascript Para Aprimorar Suas Habilidades.** Disponível em: <https://awari.com.br/as-melhores-bibliotecas-de-front-end-em-javascript-para-aprimorar-suas-habilidades/>. Acesso: 18 de Dez. 2023.

ANDREI L. **O que é react e como funciona?.** Disponível em: <https://www.hostinger.com.br/tutoriais/o-que-e-react-javascript>. Acesso: 12 de Nov. 2023.

AKIRA HANASHIRO, **Vs code - o que é e por que você deve usar?.** Disponível em: <https://www.treinaweb.com.br/blog/vs-code-o-que-e-e-por-que-voce-deve-usar?>. Acesso: 12 de Nov. 2023.

SOHAM DE ROY. **O que é tailwind css? um guia para iniciantes.** Disponível em: <https://www.freecodecamp.org/portuguese/news/o-que-e-tailwind-css-um-guia-para-iniciantes/>. Acesso: 14 de Nov. 2023.

TAILWINDCSS. **Get started with Tailwind CSS.** Disponível em: <https://tailwindcss.com/docs/installation>. Acesso: 27 de Nov. 2023.

IHECHIKARA ABBA. **Como usar o tailwind css para desenvolver rapidamente os sites da snazzy.** Disponível em: <https://kinsta.com/pt/blog/tailwind-css/>. Acesso: 14 de Nov. 2023.

ANDREYNA CARVALHO. **Saiba como aplicar a validação de campos no react no seu projeto.** Disponível em: <https://coodesh.com/blog/candidates/saiba-como-aplicar-a-validacao-de-campos-no-react-no-seu-projeto/>. Acesso: 14 de Nov. 2023.

BRAD HICK. **Formik:react forms sem dor.** Disponível em: <https://bradhick.medium.com/formik-react-forms-sem-dor-12271acd61da>. Acesso: 14 de Nov. 2023.

FORMIK. **Overview**. Disponível em: <https://formik.org/docs/overview>. Acesso: 27 de Nov. 2023.

ASCOM. **Quais são os tipos de pesquisa e abordagem em um trabalho acadêmico**.

Disponível em: <http://www.uesb.br/noticias/quais-sao-os-tipos-de-pesquisa-e-abordagem-em-um-trabalho-academico>. Acesso: 14 de Nov. 2023.

MADEINWEB. **Por que investir em um sistema web?**. Disponível em:

<https://madeinweb.com.br/por-que-investir-em-um-sistema-web/>. Acesso: 14 de Nov. 2023.

FAGNER RODRIGUES. **Vale a pena desenvolver seu próprio sistema de gestão?**. Disponível em: <https://www.conveniar.com.br/blog/vale-a-pena-desenvolver-seu-proprio-sistema/>. Acesso: 14 de Nov. 2023.

GIZELE SILVA. **O que é Next.js?**. Disponível em:

<https://coodesh.com/blog/dicionario/o-que-e-next-js>. Acesso: 14 de Nov. 2023.

DIEGO MELO. **O que é TypeScript? [Guia para iniciantes]**. Disponível em:

<https://tecnoblog.net/responde/o-que-e-typescript-guia-para-iniciantes/>. Acesso: 12 de Nov. 2023.

TYPESCRIPT. **What is TypeScript?**. Disponível em: <https://www.typescriptlang.org/>. Acesso: 27 de Nov. 2023.

JORGE NASCIMENTO. **Introdução ao TypeScript**. Disponível em:

<https://dev.to/nascimento/introducao-ao-typescript-3ekh/>. Acesso: 12 de Nov. 2023.

PABLO HENRIQUE AGUIAR CAVALCANTE. **Introdução a Typescript: o que é e como começar?**. Disponível em: <https://blog.geekhunter.com.br/introducao-a-typescript/>. Acesso: 12 de Nov. 2023.

JORGE NASCIMENTO. **React - O que são Componentes**. Disponível em:

<https://dev.to/nascimento/react-o-que-sao-componentes-40f0>. Acesso: 15 de Nov. 2023.

BRUNO BRAGA. **Melhores práticas para organização de código em TypeScript**. Disponível em: <https://orango.dev/melhores-praticas-para-organizacao-de-codigo-em-typescript/>.

Acessado em: 15 de Nov. 2023.

MUI. **Material UI - Overview**. Disponível em: <https://mui.com/material-ui/getting-started/>. Acessado em: 20 de Dez. 2023.

GABRIEL LEITE. **Saiba o que é JSON e como utilizar**. Disponível em: <https://www.alura.com.br/artigos/o-que-e-json>. Acessado em: 15 de Nov. 2023.

DIEGO MELO. **O que é JSON? [Guia para Iniciantes]**. Disponível em: <https://tecnoblog.net/responde/o-que-e-json-guia-para-iniciantes/>. Acessado em: 15 de Nov. 2023.

MDN WEB DOCS. **Trabalhando com JSON**. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/Objects/JSON>. Acessado em: 15 de Nov. 2023.

IVAN DE SOUZA. **Afinal, o que é JSON e para que ele serve? Descubra agora!**. Disponível em: <https://rockcontent.com/br/blog/json/>. Acessado em: 15 de Nov. 2023.

JSON. **Introducing JSON**. Disponível em: <https://www.json.org/json-en.html>. Acesso: 27 de Nov. 2023.

ANEXO A – TABELAS DESENVOLVIDAS PARA A APLICAÇÃO

