



10º Encontro de Ensino Pesquisa e Extensão

Patrocínio, MG, outubro de 2023

ANÁLISE AVALIATIVA ACERCA DO USO DO REACT PARA PROGRAMAÇÃO WEB FRONT-END

João Augusto Marciano Silva; Cintia Carvalho Oliveira
Instituto Federal do Triângulo Mineiro – Campus Patrocínio
Pesquisa
Artigo Completo
PIBIC IFTM

Resumo:

Durante o desenvolvimento de uma página web, é de grande eficiência o uso de um framework para agilizar o trabalho a ser desenvolvido na área de *front-end* ou *back-end*, sendo assim economizando tempo e recursos. Entretanto, um problema que ocorre constantemente com os desenvolvedores é a dificuldade de encontrar um framework que se adeque perfeitamente ao projeto em que está trabalhando, isto pois, muitos desses *frameworks* possuem várias diferenças entre si em questão de documentação, idioma e frequência de atualizações. Logo, isso atrasa o desenvolvedor, fazendo-o perder tempo com *frameworks* que não se aplicam ao projeto por incompatibilidade ou por dificuldades com documentação, seja por falta desta ou pelo idioma em que se encontra. Nesse sentido, buscando facilitar a vida de um desenvolvedor web *front-end* que pretende utilizar o React em seu projeto, foi realizada uma pesquisa sobre esse framework para mostrar um panorama geral com vários pontos de críticas positivas e negativas. Com um estudo baseado na experiência do uso do React como framework de front-end, será possível demonstrar com clareza o melhor que este framework pode ofertar para cada tipo de situação, seja quando observa-se se possui a documentação em português, ou quando se deseja algo mais prático e simples, isto entre outros pontos.

Palavras-chave: Estudo avaliativo. Front-end. Programação Web. React Framework.

Introdução

Atualmente, o React está entre um dos *frameworks* mais utilizados na programação web tanto para *front-end*, quanto também para *back-end*. Entretanto, este também é um *framework* complicado de se entender e utilizar dependendo do nível de conhecimento em que se possui da área. Buscando facilitar um conhecimento prévio acerca do React, foi realizada esta pesquisa científica que se baseia em um experimento

próprio do autor, onde foi observado as dificuldades e facilidades no uso deste *framework*.

Durante o trabalho foram observadas várias questões para serem respondidas ao final da pesquisa, questões estas que avaliam o uso do React na visão de um programador web de *front-end* com conhecimento prévio sobre HTML (*HyperText Markup Language*), CSS (*Cascading Style Sheets*) e JS (JavaScript). Ao todo, estas informações levantadas servem para auxiliar qualquer profissional de informática que queira iniciar um projeto com o React.

Objetivo

Neste trabalho o objetivo principal é realizar um estudo avaliativo acerca do *framework* React voltado à área de *front-end* em JavaScript.

Como objetivos específicos temos:

- Pesquisar quais são os *frameworks* mais utilizados atualmente;
- Levantar informações básicas sobre o React;
- Desenvolver um blog em HTML;
- Desenvolver novamente o mesmo blog utilizando o React;
- Criar um quadro avaliativo acerca deste *framework*.

Metodologia

Por meio de pesquisa online foram realizadas várias buscas meticolosas de diversas fontes observando os sites com confiabilidade e veracidade em suas estatísticas a fim de construir uma base de conhecimento acerca do *framework* React e suas interações. Em seguida, com a metodologia experimental, foram criados blogs em HTML como material de estudo a fim de analisar os efeitos do React práticos neste projeto.

Referencial Teórico

Nesta seção serão apresentados os conceitos necessários para o entendimento deste trabalho.

Javascript

O JavaScript (JS) é, atualmente, uma das linguagens de programação mais utilizadas para o desenvolvimento web. Segundo Melo (2021), do site Tecnoblog, o JavaScript é uma linguagem de alto nível, o que significa que sua sintaxe é voltada para o entendimento humano, a qual permite a criação de páginas interativas com os usuários, possibilitando a utilização de elementos dinâmicos e melhorando a performance. Além disso, vale destacar que o JavaScript é uma linguagem interpretada, ao contrário de outras linguagens como C ou C++ as quais são compiladas.

Framework

Um *framework* é basicamente uma estrutura que possui modelos pré-definidos que facilitam e agilizam a construção de projetos. De acordo com Bittencourt (2021), da plataforma Alura, a utilização de um *framework* permite que o foco e esforços estejam no desenvolvimento em si do projeto e não em configurações e padrões, o que automatiza as tarefas. Além disso, segundo a mesma autora, com a utilização de um *framework*, o código fica usualmente mais limpo e legível, facilitando o entendimento da implementação e posterior manutenção. Como exemplos de *frameworks* da linguagem JavaScript, pode-se citar o Angular, Vue.js e React JS.

Front-end

Front-end é a parte desenvolvedora que interage diretamente com o usuário através da construção de interfaces gráficas, diferentemente do *Back-end*, o qual comumente atua no *background* (por trás) do desenvolvimento de um projeto. De acordo com Souto (2023), da plataforma Alura, a complexidade dos sistemas atuais exigiu que as funcionalidades fossem separadas, criando assim desenvolvedores dos tipos *Front* e *Back-end*. Consoante o autor, as ferramentas base para um desenvolvedor *Front-end* são: HTML, CSS e JavaScript, contudo, este também pode contar com o auxílio de *frameworks* para a realização de seu trabalho.

React

O React é uma biblioteca JavaScript usada para criar interfaces de usuário interativas, e em pouco tempo, se tornou uma das tecnologias mais usadas no desenvolvimento web. Segundo Neves (2023), escritor da plataforma de ensino Alura, ele justifica essa ascensão do React ao fato de ele ser uma biblioteca de código aberto,

isto significa que há uma comunidade ativa de pessoas que contribuem com código e recursos para assim ajudar a melhorar a tecnologia cada vez mais.

Desenvolvimento e Resultados

Com as pesquisas realizadas a fim de saber quais os *frameworks* mais utilizados no momento foi observado que os 3 (três) mais citados por fontes como a Awari (2023), AppMaster (2023) em ordem classificatória são:

1. React;
2. Vue.js;
3. Angular.

Pelo grande número de indicações como o *framework front-end* mais utilizado, foi decidido que o React seria o *framework* a ser utilizado como objeto de estudo. A partir disso começou-se então a pensar em pontos em que poderiam ser observados durante o experimento, e os pontos principais foram:

- Qual o volume de documentação gratuita disponível?
- Em qual idioma se encontra o *framework*?
- Em qual idioma se encontra a maior parte da documentação?
- Com qual frequência é atualizado?
- Qual o conhecimento prévio requerido?
- Qual o nível de praticidade do *framework*?
- Qual o nível de complexidade do *framework*?
- O *framework* possui compatibilidade com quais outros *frameworks* usuais?
- O *framework* facilita a identificação de erros?
- O *framework* é recomendado para uso coletivo?

Com estes pontos a serem observados, deu-se início ao desenvolvimento do experimento em si. Foi escolhido como ideia de projeto um blog chamado “Notícias do PETi” sem fins lucrativos até o momento deste trabalho. A escolha desse projeto surgiu para possibilitar um futuro uso deste blog para o PETi Conexão Patrocínio, Projeto de Educação Tutorial Institucional que está localizado no Instituto Federal do Triângulo Mineiro de Ciência e Tecnologia - *Campus Patrocínio*.

Com a ideia pronta, começou então a primeira etapa do projeto: criação do blog simples. Um blog sem a utilização do React e apenas com HTML, CSS, JS e com o auxílio do Bootstrap e o Font Awesome, ambos para agilizar o processo do site, foi surgindo a primeira etapa. Entretanto, antes de começar de fato a programar o blog, é necessário idealizar o que será construído, por isso a primeira fase desta etapa foi identificar os elementos do site, requisitos e funcionalidades, criando-se então *wireframes* (desenhos técnicos) das páginas do blog, assim possibilitando uma visão prévia de o que seria o site e como ele funcionaria.

Após algumas ideias de nomes e logotipo, chegou-se a três opções conforme a Figura 1.

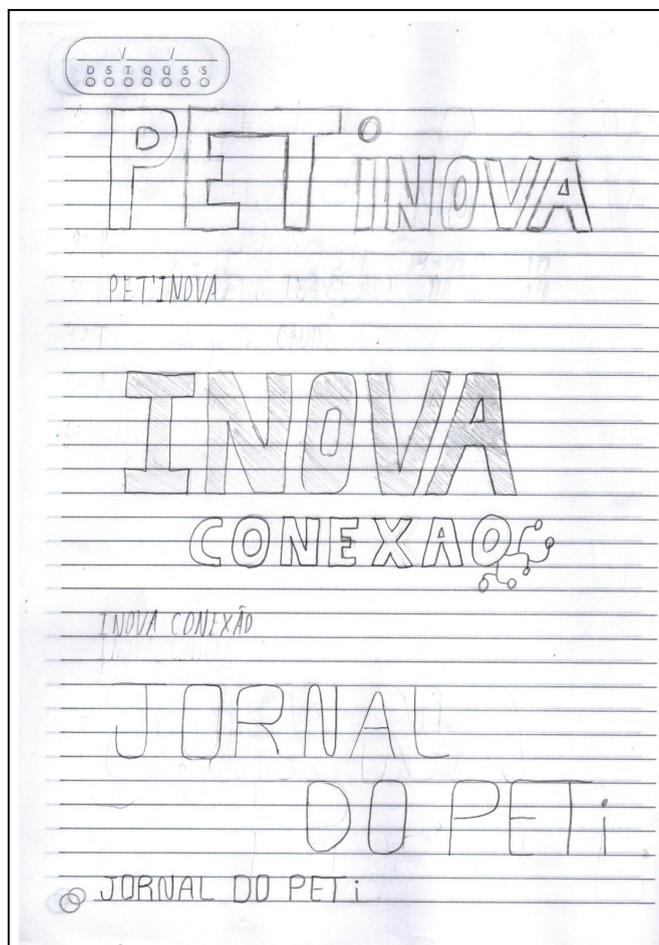


Figura 1 - Ideias de nome e logotipo

Dentre esses três nomes: PET'inoa, Inova Conexão e Jornal do PETi. O que demonstrava mais intuitivo e mais fácil de entender e recordar foi o “Jornal do PETi”, que futuramente se tornou “Notícias do PETi” no qual teve como logotipo a Figura 2.



Figura 2 - Logo Notícias do PETi

Já nos desenhos técnicos das páginas do blog, todas buscaram a mesma base de cabeçalho, rodapé e menu de navegação como pode ser observado na Figura 3 onde mostra o *wireframe* da página inicial do site. O cabeçalho conta com a logomarca do blog, uma barra de pesquisa com a possibilidade de utilizar um filtro avançado e também algumas opções de acessibilidade para o site. No menu lateral tem um guia de navegação e a ideia deste menu é que ele seja retrátil para melhorar a visualização limpa e organizada do site. Por fim, no rodapé tem alguns *links* de acesso rápido e também informações de contato. O corpo central do *wireframe* é a única coisa que altera de uma página para outra, por isso apenas era necessário idealizar a base do site e a página inicial, pois as próximas páginas seguirão o mesmo padrão.

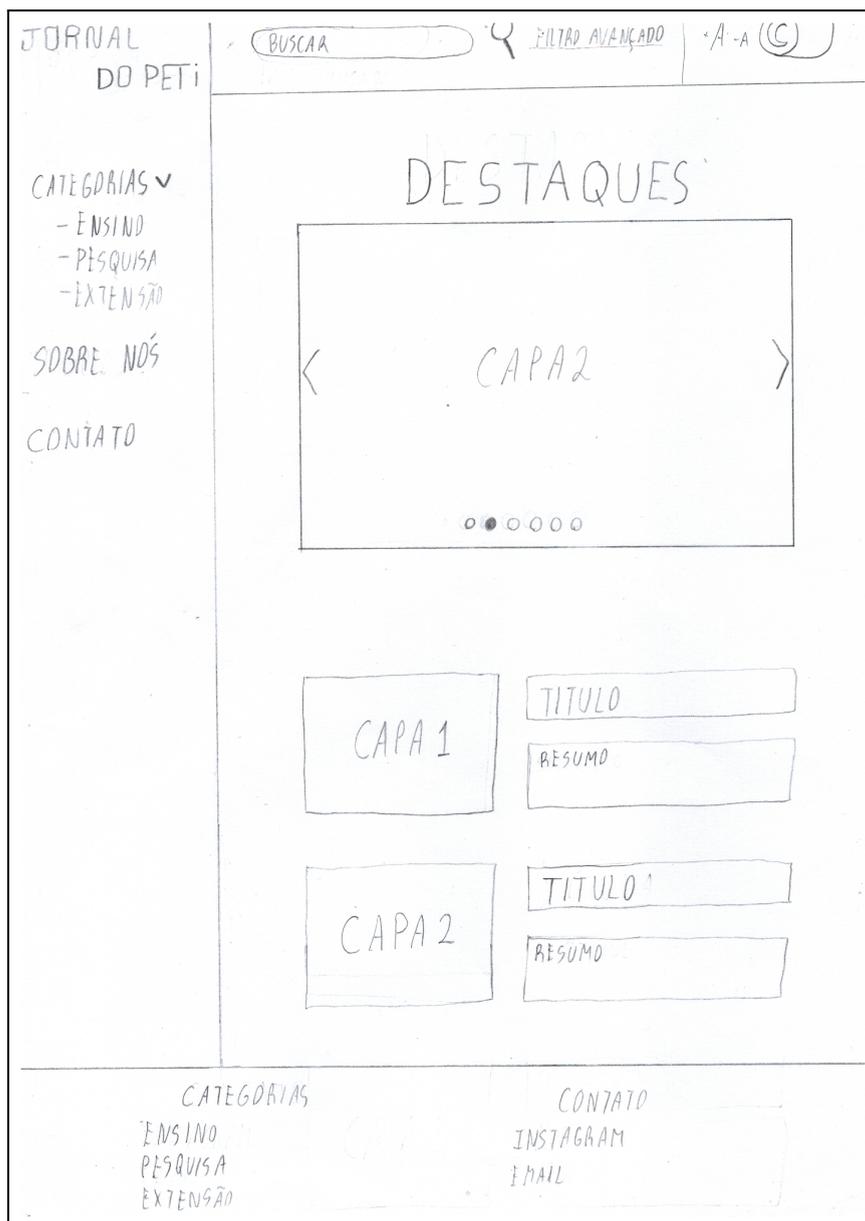


Figura 3- *Wireframe* da Página Inicial

Tendo em mãos as ideias e requisitos definidos, já se podia dar início à segunda fase da primeira etapa, onde começa-se a desenvolver o site em código. Nesta fase, além do Bootstrap e do Font Awesome, foram também utilizados o Git e o Github para fazer o versionamento do site e também para manter um controle do desenvolvimento do blog. Após o desenvolvimento, junto com alguns ajustes e complementações na ideia inicial do site, obteve-se como resultado três páginas: a página inicial (Figura 4), a página com todas as notícias e por último uma página com informações sobre o site e contatos.

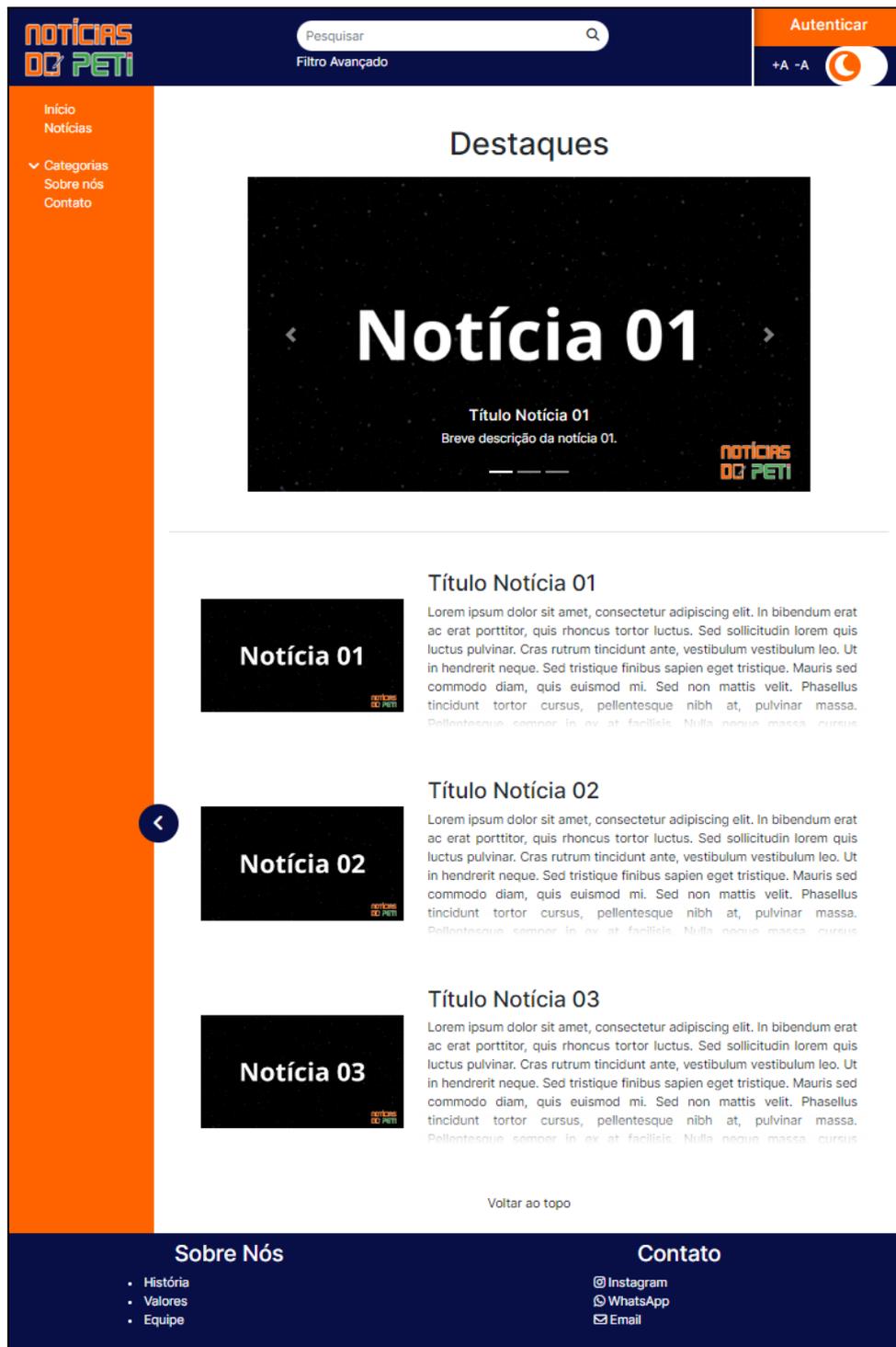


Figura 4 - Página Inicial do Blog

Com o site sem a utilização do React pronto, finalizou-se a primeira etapa e este primeiro blog pode ser visitado através do seguinte *link* do GitHub Pages: <https://joaoaugusto2020.github.io/noticias-do-peti/index.html>. Em seguida deu-se início a segunda etapa do projeto que é a construção deste mesmo site porém desta vez com o uso do React, para assim observar os 12 pontos anotados anteriormente.

Nesta etapa foram observados muita dificuldade para entender a estrutura do *framework* e como ele funciona, por este motivo foi pausado o desenvolvimento do site para que o React pudesse ser estudado de forma prática através de um site modelo com acompanhamento. Nesse sentido, para fins de estudos, foi criado um site de lista de tarefas com o React, também chamado de “*To Do List*”, isto acompanhado com a vídeo-aula do Mateus Battisti, do canal Hora de Acordar. Matheus diz em sua aula que:

O projeto é uma ótima maneira de aprender React ou aprimorar seus conhecimentos, apresentando diversas funcionalidades úteis, como a criação de tarefas, remoção de tarefas, alteração de status da tarefa (completa ou incompleta), filtro por status e ordenação alfabética, além da pesquisa de tarefas! (BATTISTI. M., 2023)

Terminado os estudos sobre o React e entendido sua estrutura, foi possível identificar de maneira mais clara as suas funcionalidades e como implementá-las. Então foi retomado o desenvolvimento da segunda etapa, onde o site da primeira etapa seria refeito utilizando o React.

Depois de toda a construção do site e adaptações necessárias, tais como o uso do Bootstrap e o Font Awesome. Foi finalizado o desenvolvimento do segundo blog até a sua página inicial, pois esta é a principal e que contém maior parte das funcionalidades. Com a segunda etapa concluída podia-se dar início a terceira e última etapa, e nesta seria necessário tratar os dados recolhidos durante as etapas anteriores, e tudo isto havia sido anotado em um arquivo *online* para que não se perca nenhuma informação. Desta forma foi criado um quadro avaliativo (Tabela 1) onde apresenta respostas para todas as 10 perguntas que foram observadas no decorrer do desenvolvimento.

Tabela 1 - Quadro Avaliativo

QUADRO AVALIATIVO ACERCA DO REACT	
QUESTÃO	RESULTADO
Qual o volume de documentação gratuita disponível?	Média, depende da plataforma, a maior quantidade encontrada é no Youtube, no próprio site do React e em sites de ensino com por exemplo o W3schools.
Em qual idioma se encontra o <i>framework</i> ?	Inglês.
Em qual idioma se encontra a maior parte da documentação?	Inglês e Português (Brasil).
Com qual frequência é atualizado?	Média, pode variar de 1 a 12 meses a depender da quantidade de adições.
Qual o conhecimento prévio requerido?	HTML, CSS e JS.
Qual o nível de praticidade do <i>framework</i> ?	Alto, como a atualização das páginas ocorrem de forma síncrona, o próprio sistema identifica atualizações e já realiza sugestões que torna mais prático conforme o tempo.
Qual o nível de complexidade do <i>framework</i> ?	Médio, no início pode-se parecer muito complexo, mas com pouco tempo de uso torna-se mais simples.
O <i>framework</i> possui compatibilidade com quais outros <i>frameworks</i> usuais?	Sim, dentre eles o Bootstrap e o Vue.js.
O <i>framework</i> facilita a identificação de erros?	Sim, devido a organização que se encontra a estrutura do React, torna-se mais rápida a identificação de erros.
O <i>framework</i> é recomendado para uso coletivo?	Sim, uso coletivo e individual.

Conclusão e Considerações Finais

Com a pesquisa feita, pode-se concluir que o React é um ótimo *framework front-end* que possui suas vantagens e dentre elas pode-se destacar os seguintes pontos:

- O React é muito interessante para particionar o site, isso ajuda a manter um foco em uma estrutura por vez, além de facilitar a identificação de bugs e erros e a manutenção destes;

- Com o React é possível implementar estruturas prontas, isso pode ser muito útil em um trabalho em equipe onde cada membro se responsabiliza por uma estrutura em específica, e conforme forem sendo finalizadas podem ser implementadas sem interferência com as outras partes do código;
- O React é muito organizado, depois que se entende a sua lógica de organização de diretórios ou quando se cria a própria lógica de diretórios, se torna muito mais simples, rápido e fácil de encontrar determinada parte do código.

Entretanto, o React também possui algumas desvantagens, e estas não diminuem a efetividade dos pontos positivos já citados, mas apontam possíveis melhorias para o React. Dentre estas desvantagens pode-se perceber que:

- A organização do React é bem confusa no início, é bem complicado entender as conexões entre as pastas;
- Não é viável realizar o *upload* do projeto inteiro direto para o Github, é necessário particioná-lo para que funcione corretamente devido aos seus módulos e grande quantidade de arquivos;
- Não há uma forma padrão para entender as extensões (Módulos e *Hooks*) que o React tem e isto para um iniciante pode ser complexo;
- Por mais que o React suporte outros *frameworks*, ainda sim é complicado utilizar o React junto com outro *framework*, por exemplo o Bootstrap, isto por ele ser particionado, e no caso do Bootstrap, ele geralmente depende de outras estruturas para que uma outra funcione corretamente.

Por fim, através deste trabalho, foi possível entender na prática o uso do React observando o ponto de vista de um programador web *front-end* e assim foi apresentado vários pontos que avaliam o React de uma maneira geral e também o suficiente para sanar dúvidas de um desenvolvedor na hora de escolher o React como um *framework* a ser usado em seu projeto.

Referências

APPMASER (2023). **Os 10 principais frameworks de front-end populares para usar em 2023.** Disponível em: <https://appmaster.io/pt/blog/frameworks-front-end-populares>. Acesso em: 12 de jun de 2023.

AWARI (2023). **As Melhores Frameworks Front-end De 2022.** Disponível em: <https://awari.com.br/as-melhores-frameworks-frontend-de-2022>. Acesso em 12 de jun de 2023

BITTENCOURT. J. **Framework: o que é e pra que serve essa ferramenta?** Alura (2021). Disponível em: <https://www.alura.com.br/artigos/framework-o-que-e-pra-que-serve-essa-ferramenta>. Acesso em: 22 de fev de 2023.

MATHEUS BATTISTI - HORA DE CODAR. **PROJETO de React JS para INICIANTES - Faça uma To Do List do zero!** YouTube, 2023. Disponível em: <https://www.youtube.com/watch?v=YVEVrigByKY>. Acesso em: 03 de jul de 2023.

MELO. D. **O que é JavaScript? [Guia para iniciantes].** Tecnoblog (2021). Disponível em: <https://tecnoblog.net/responde/o-que-e-javascript-guia-para-iniciantes/>. Acesso em: 22 de fev de 2023.

NEVES. V. **React: o que é, como funciona e um Guia dessa popular ferramenta JS.** Alura (2023). Disponível em: <https://www.alura.com.br/artigos/react-js>. Acesso em: 12 de jun de 2023.

SOUTO. M. **Front-end, Back-end e Full Stack.** Alura (2023). Disponível em: <https://www.alura.com.br/artigos/o-que-e-front-end-e-back-end>. Acesso em: 22 de fev de 2023.