

Rio De Janeiro, Rio De Janeiro.
heberttatferreira.com@gmail.com
(+55) 21 97668-5649

Hebertt Da Silva Nascimento

Desenvolvedor Web Front-End

Objetivo Profissional

Atualmente, estou buscando oportunidades que me permitam contribuir com a empresa e ganhar experiência na área através do desenvolvimento e promoção de tecnologias consolidadas e/ou inovadoras. Atualmente também estou estudando stacks back-end como node.js e Mysql.

Formação

- Cursando bacharel em Engenharia De Software
- Desenvolvimento Web Full Stack - Trybe, será concluído em julho de 2023.
O programa conta com mais de 1.500 horas de aulas e aborda introdução ao desenvolvimento de software, front-end, back-end, ciência da computação, engenharia de software, metodologias ágeis e habilidades comportamentais.

Experiências Profissionais

Lojas Americanas

Auxiliar

Responsável por requisições de cartões de crédito –
setembro de 2016 / agosto de 2017

Auxiliar De Escritório – Rabelo e Rebouças Consultoria(Status).

maio 2021 / Janeiro 2022

Responsável por consultas e cadastros de clientes e atendimento ao cliente.

Principais tecnologias

- HTML5
- CSS3
- Javascript
- GitHub
- React
- Redux
- context API e hooks,
- Scss
- TypeScript
- Material Ui
- Docker
- Jest
- Test RTL

Idiomas

- Inglês – Fluente
- Francês - Intermediário

Principais projetos

App de Receita: Aplicação de receita.

[Link para o GitHub e deploy](#)

- Tecnologias Utilizadas: React, context API e Hooks, Css.
- Feito em grupo utilizando metodologia ágeis e scrum.

Projeto TribeWallet: App de controlador de despesas.

[Link para GitHub](#)

- Tecnologias Utilizadas: React, Redux e Css.

Projeto Trivia: app de Quiz.

[Link para o GitHub](#)

- Tecnologias Utilizadas: React, Context Api e Css.
- Feito em grupo utilizando metodologia ágeis e scrum.

Projeto Page Films: site que mostra descrição de animes.

[Link para o GitHub](#)

- Tecnologias Utilizadas: React, context API e Hooks, Css.

Projeto Trybe Tunes: app de música.

[Link para o GitHub](#)

- Tecnologias Utilizadas: React e usando gerenciador de estado Redux