

# MARCELO SOARES

## Desenvolvedor Full-stack



[www.marcelo-soares-codes.vercel.app/](http://www.marcelo-soares-codes.vercel.app/)



[/marcelo-soares-codes](https://www.linkedin.com/in/marcelo-soares-codes)



(75) 999867597-7890



[marcelo.soares1070@gmail.com](mailto:marcelo.soares1070@gmail.com)

## SOBRE MIM

Sou formado como Técnico em Informática após três anos de estudo técnico. Durante esse período, foquei em desenvolvimento back-end em Python, trabalhando em diversos projetos. Atualmente, como freelancer em programação web, estou constantemente aprimorando minhas habilidades nas tecnologias que domino e buscando aprender novas para atender às demandas dos meus clientes. Tenho experiência tanto em front-end quanto em back-end, proporcionando soluções completas aos projetos.

## TECNOLOGIAS

- PYTHON
- TYPESCRIPT
- NODE
- REACT
- TAILWIND
- STYLED COMPONENTS
- STORYBOOK
- POSTGRASQL
- MONGODB
- DOCKER
- GIT / GITHUB

## FORMAÇÃO

- CENTRO TERRITORIAL DE EDUCAÇÃO  
PROFISSIONAL DA BACIA DO JACUÍPE  
[ 2019 - 2022 ]  
CURSO: TECNOLOGIA DA INFORMAÇÃO E  
COMUNICAÇÃO

## EXPERIÊNCIAS

### FREELANCER - BAHIA

#### DESENVOLVEDOR FRONT-END & BACK-END

JUNHO 2022 - PRESENTE

- **MEU MOMENTO - ( MARÇO 2024 - ABRIL 2024 )**

Meu Momento é um aplicativo desenvolvido para capturar replays em quadra durante atividades esportivas, oferecendo aos jogadores a oportunidade de revisitar momentos específicos do jogo posteriormente.

**No front-end**, utilizei React para a interface do usuário, React Router DOM para navegação, Axios para requisições HTTP, FontAwesome para ícones e Tailwind CSS para estilos. Para desempenho, empreguei Image-compressor.js e JS Cookie.

**No back-end**, desenvolvi a API com Node e Express, integrando Prisma para o banco de dados postgresql. Para segurança, usei Bcrypt e JSON Web Token. Além disso, incluí nodemailer e nodemailer-mailgun-transport para e-mails, Yup para validação e Cors para integridade dos recursos.

[Acesse o site aqui](#)

- **T.R.F.S.E - ( JUNHO 2022 - NOVEMBRO 2022 )**

Utilizando a linguagem de programação Python, o sistema reconhece os rostos dos alunos e funcionários registrados na escola. Ao chegar à portaria, o sistema identifica a pessoa e libera a entrada apenas para aqueles autorizados, garantindo um ambiente seguro.

**No Software**. O projeto utiliza Python 3.8 como linguagem principal. Para a interface do usuário, são usados os frameworks Kivy (v2.10) e KivyMD (v1.0.2). A comunicação com a placa Arduino é feita via pyFirmata (v1.1.0). O MongoDB (v4.2.0) é o banco de dados integrado. Para processamento e reconhecimento de rostos, são empregadas as bibliotecas opencv-python (v4.6.0.66) e opencv-contrib-python (v4.6.0.66). A manipulação de imagens é feita com Pillow (v9.0.1). A Arduino IDE é usada para desenvolvimento na placa Arduino Nano.

[Acesse o Repositório aqui](#)