DOCUMENTAÇÃO DA API RESTFUL – SISTEMA DE CHAMADAS

**Finalidade**

Nossa aplicação tem o objetivo de fazer autenticação JWT e validação dos dados inseridos pelo usuário, fazendo uma conexão com o banco de dados MongoDB para tal, mantendo a segurança em todos os estágios do processo e um bom fluxo do processo.

Ferramentas utilizadas:

## Programas

Visual Studio Code – É uma IDE (Integrated Development Environment) em que utilizamos para rodar alguns programas que são possíveis de compilações e outros que precisam ser interpretados.

Node.JS – Node.JS é um programa que executa código Javascript e o transforma em linguagem de máquina, sem que haja a necessidade de usar um navegador.

Git – É um programa de versionamento de arquivos.

## Frameworks

Vue.JS – É um framework Javascript que tem como objetivo a criação de interfaces de usuário e aplicativos de página única.

Express – Express é um pacote do NPM que define e gerencia a estrutura que uma requisição encontrará ao ser processada. Ele facilita a criação de servidores web no Node.js, permitindo configurar rotas, middlewares e respostas de forma eficiente e flexível.

Mongoose – Mongoose é um pacote do NPM que irá fazer a conexão da nossa aplicação com o MongoDB. Ele fornece uma camada de abstração, permitindo a definição de esquemas, validação, consultas e operações no banco de dados. Com ele, é possível trabalhar com dados do MongoDB utilizando uma interface orientada a objetos.

## Banco de dados

MongoDB – É um banco de dados não relacional (NoSQL), ele utiliza estruturas JSON como esquemas.

body-parser –  É um middleware que converte o body da requisição para n formatos, inclusive JSON.

## Middlewares

Morgan – O Morgan é um middleware de logging para **Node.js**, utilizado para registrar as requisições HTTP feitas ao servidor. Ele ajuda a monitorar as requisições e suas respostas.

body-parser –  É um middleware que converte o body da requisição para n formatos, inclusive JSON.

## Modulos do node

CORS – CORS é um mecanismo de segurança que filtra quem pode enviar requisições para o seu servidor.

Bcrypt – É uma bliblioteca que transforma senhas em HASH.

Bcryptjs – É uma implementação Javascript do Bcrypt para o Node.js

Dotenv – É um pacote NPM que irá esconder nossa conexão do banco de dados, afim de assegurar nossa aplicação.

## Padrao de token

Jsonwebtoken – JWT é uma autenticação que usa de Headers, Payload e Signatures.

## Extensões VSCODE

.editorconfig – É uma extensão do VSCODE que permite a melhor estruturação do nosso código e padronização.

Eslint friendly formatter – É um linter que permite a melhor estruturação do nosso código de acordo com padrões estabelecidos.

Nodemon – É um pacote que roda nossa aplicação “ao vivo”.

## Modulos da parte do frontend

vue-jwt-decode

axios

bootstrap