

## **Tecnologias a serem utilizadas**

### **Linguagens**

#### **HTML – HiperText Markup Language**

Criada pelo britânico Tim Berners-Lee, HTML (HiperText Markup Language), em português, Linguagem de Marcação de Hipertexto, acaba por ser tornar o principal elemento para a inserção e estruturação básica em seu website. Usado para organização e informações em sua tela, o HTML sem dúvida é indispensável para a criação de seu website, pois é a partir dele que o navegador consegue exibir os textos, elementos, imagens, programados pelo desenvolvedor. Está tecnologia foi escolhida em nosso projeto pois ela será o grande pilar para a criação de nosso website, a mesma ficará responsável pela estruturação e inserção de estilizações em nosso site, fazendo com que fique conforme o tema elaborado pela equipe.

#### **CSS – Cascading Style Sheets**

Criado em 1996 pelo W3C (World Wide Web Consortium), CSS (Cascading Style Sheets), em tradução para português, Folhas de Estilo em Cascata, é uma linguagem de estilo que apresenta para o usuário documentos escritos tanto em HTML quanto XML com formatações visuais em sua página. Podendo mudar a cor de se texto, background, espaçamentos, fontes estilizadas e outras infinidades de estilos que o desenvolvedor pode codificar. O CSS foi escolhido para nosso projeto pelo fato de as estilizações serem feitas a partir dele, podendo mudar o tamanho e cores de textos, aumentar e diminuir imagens e outras infinidades de atributos que podemos inserir para uma melhor apresentação de nosso website.

## **JS – JavaScript**

Js ou JavaScript, foi criado em 1996 pelo programador pelo Brendan Eich, é uma linguagem de programação de comportamento, podendo criar e manipular criação de conteúdos dinâmicos. Focando na interatividade com o usuário, o JavaScript é de difícil manuseio, categorizando-se como uma linguagem de alto-nível. Com o decorrer dos anos, o mesmo vem sofrendo diversas alterações em quesito funções e aplicabilidades, sempre ficando mais versátil e completo a cada atualização que recebe. O JavaScript foi escolhido para nosso projeto, pois ele ficará responsável por toda a interação do usuário com a plataforma, ou seja, é através desta tecnologia que teremos os efeitos, alertas, entre outros tipos de interatividade.

## **MySQL**

Escrito e desenvolvido em C e C++ pelos suecos David Axmark, Allan Larsson e o finlandês Michael Widenius a partir de 1980, O MySQL trata-se de um Banco de Dados relacional (RDBMS – Relational Database Management Systems), seguindo uma modelo de banco modelo-cliente. Por se tratar de um RDBMS, o mesmo possui uma linguagem SQL (Structured Query Language) como principal pilar. Escolhemos essa tecnologia pelo fato de ser bastante conhecida e por ser uma linguagem de simples entendimento, podendo ser compreendida de maneira rápida e eficaz, tendo assim uma fácil usabilidade. Outro grande ponto que a torna ideal para nosso projeto, é possibilidade de o MySQL ser compatível com vários sistemas operacionais, nos possibilitando a testá-lo em qualquer ambiente.

## **Bibliotecas**

### **Dontenv.js**

Dotenv.js é uma biblioteca JavaScript escrita em Typescript e que roda no node.js. É responsável por buscar informações armazenadas nas variáveis de ambiente declaradas no arquivo “process.env”. Atualmente é muito utilizado para diferenciar o ambiente de produção e o ambiente de desenvolvimento, além de guardar informações sensíveis como o usuário e senha do banco de dados. Essa

tecnologia foi escolhida para aumentar a segurança do projeto, armazenando as senhas do banco de dados e fazendo a diferenciação entre o ambiente de desenvolvimento e produção, em que, no ambiente de desenvolvimento o Back-End irá se comunicar com um banco de dados feito exclusivamente para testes.

## **MySQL2.js**

O MySQL2.js é uma biblioteca JavaScript do Node.js responsável por fazer a conexão com o banco de dados MySQL. Foi criado em parceria com a equipe desenvolvedora de outra biblioteca, o MySQL.js, isso ocorreu devido a lentidão de desempenho que a antiga tecnologia apresentava. Essa tecnologia foi escolhida pois apresenta maior facilidade de uso e alto desempenho, além de ter diversas opções para combater o SQL injection, com ela é possível integrar o Back-End com o banco de dados de forma segura.

## **Cookie Parse**

É uma biblioteca middleware para o Express.js, desenvolvido com o propósito de facilitar o uso dos cookies enviados pelo cliente em uma requisição http para um servidor node. Essa biblioteca foi escolhida para armazenar informações sensíveis, com a principal sendo a credencial do usuário, que é enviada automaticamente ao Back-End com as requisições feitas pelo usuário. Com essa tecnologia é possível dar respostas que vão de acordo com o seu nível de acesso dentro do sistema.

# **Frameworks**

## **Express.js**

De acordo com o site oficial, o Express.js é um framework minimalista desenvolvido para otimizar nas construções de aplicações web e API's. Foi criado inicialmente em maio de 2010, por TJ Holowaychuk, StrongLoop e entre outros, desde então é o mais utilizado para definir as rotas das aplicações web, tanto por iniciantes, como por plenos e seniores da área. Vale ressaltar que o mesmo possui código aberto sobre a licença MIT e utiliza de node para a execução do JavaScript como linguagem

de Back-End. Essa tecnologia foi escolhida em nosso projeto para otimizar a construção de aplicações web e API's, utilizando o Node.js para execução do JavaScript como linguagem de Back-End.

## **Plataformas**

### **Node.js**

O Node.js foi criado por Ryan Dahl em 2009, após analisar as formas como as páginas webs funcionavam em seu processo de codificação. Ryan, analisou que as aplicações utilizadas no desenvolvimento de web sites consumia uma quantidade de recursos de processamento acima da média do que era necessário e chegou à conclusão que as aplicações eram mal utilizadas e ineficientes. O Node.js é uma tecnologia que permite desenvolvedores de diversos sistemas executar códigos JavaScript fora do navegador, o mesmo permite a construção de aplicações web em geral, até API's e microsserviços. O benefício desta tecnologia permite a conexão e a facilitação entre o navegador (Google Chrome, por exemplo) com o V8 interpretador de código Java Script, e a execução do JavaScript criada pelo Node JS. Sendo uma tecnologia Back-End, ou seja, que é escrita e compreendida no servidor, o Node.JS irá possuir a função em nosso projeto de rodar JavaScript do lado do servidor, possibilitando executar aplicações desenvolvidas com a linguagem de forma autônoma, sem depender do browser.

### **Express React Views**

Segundo os desenvolvedores, Express React Views é um motor de templates desenvolvido com base no React, que busca substituir outras engines com o Ejs, Handlebars e o Jade. Não se sabe quando foi desenvolvido, mas seu repositório no GitHub conta com mais de 2.7 mil estrelas. Essa engine foi escolhida para a renderização das páginas no servidor, retornando diferentes páginas de acordo com o login do usuário e seu nível de acesso.

## **Ambientes de Programação**

### **Visual Studio Code**

Visual Studio Code ou VS Code, é uma multiplataforma voltada para edição de códigos-fontes, disponível tanto para Windows, quanto para Mac OS e Linux. Seu diferencial é a capacidade de atender enormes gamas de projetos, não apenas ASP.NET, como também Node.js, adicionalmente, o editor possui suporte à sintaxe de diversas linguagens como Python, Ruby, C++ e demais outras linguagens.

### **Sublime Text 3**

Trata-se de um editor de códigos-fontes, desenvolvido em Python, o Sublime foi criado para ser uma ferramenta leve e eficiente, se destaca quando o assunto é produtividade, com seus ambientes robustos de desenvolvimento e muitos deles gratuitos. Além de aceitar diversas linguagens de programação, o Sublime possui muitos recursos a serem explorados, sempre focando na facilitação de manuseio do usuário.

### **MySQL Workbench**

O MySQL Workbench foca na disponibilidade de ferramentas visuais para o usuário, permitindo que o mesmo possa desenvolver ou arquitetar seu Banco de Dados de forma totalmente livre, podendo modelar e gerenciá-lo visualmente da forma que quiser. Com ele é possível executar e otimizar consultas SQL, dando ao usuário a opção de mudar sintaxes de cores, exibir histórico de execuções e várias outras funcionalidades que a ferramenta disponibiliza.

## Fontes:

### VS CODE

<https://www.devmedia.com.br/introducao-ao-visual-studio-code/34418>

### Sublime

<https://www.devmedia.com.br/sublime-text-ide-introducao-a-melhor-ide-para-desenvolvimento/34117>

### MySQL Workbench

<https://www.mysql.com/products/workbench/>

### HTML

<https://www.homehost.com.br/blog/tutoriais/o-que-e-html/>

### CSS

<https://developer.mozilla.org/pt-BR/docs/Web/CSS>

<https://www.hostinger.com.br/tutoriais/o-que-e-css-guia-basico-de-css>

### JS

<https://www.hostinger.com.br/tutoriais/o-que-e-javascript>

### Node.js

<https://www.devmedia.com.br/guia/node-js/40312>

<https://www.x-apps.com.br/node-js>

## Express.js

<https://www.treinaweb.com.br/blog/o-que-e-o-express-js>

<https://expressjs.com/pt-br/>

## MySQL

<https://www.hostinger.com.br/tutoriais/o-que-e-mysql>

<https://www.infoescola.com/informatica/mysql/>

## MySQL2.js

<https://www.npmjs.com/package/mysql2#history-and-why-mysql2>

[mysql2 - npm \(npmjs.com\)](#)

[sidorares/node-mysql2: piloto mysqljs/mysql compatível com mysql para nó.js \(github.com\)](#)

## Dotenv.js

<https://www.npmjs.com/package/dotenv>

[dotenv - npm \(npmjs.com\)](#)

[motdotla/dotenv: Loads environment variables from .env for nodejs projects. \(github.com\)](#)

## Cookie Parse

[cookie-parse - npm \(npmjs.com\)](#)

[jkohrman/cookie-parse: análise e serialização de cookies do servidor HTTP \(github.com\)](#)

## Express React Views

[express-react-views - npm \(npmjs.com\)](#)

[reactjs/express-react-views: Este é um mecanismo de exibição expresso que renderiza componentes React no servidor. Ele torna a marcação estática e \\*não\\* suporta a montagem dessas visualizações no cliente. \(github.com\)](#)