# Tecnologias empregadas no Back-end.

## Node.Js

Ambiente de execução JavaScript que permite executar aplicações desenvolvidas com a linguagem de forma autônoma, sem depender de um navegador. Com ele é possível criar praticamente qualquer tipo de aplicações web, desde servidores para sites estáticos e dinâmicos, até APIs e sistemas baseados em microserviços.

## Express

Framework amplamente utilizado com o Node.Js devido às diversas características que facilitam o desenvolvimento, como possuir um sistema de rotas completo, possibilitar o tratamento de exceções dentro da aplicação, permitir a integração de vários sistemas de templates que facilitam a criação de páginas web para suas aplicações, gerenciar diferentes requisições HTTP com seus mais diversos verbos, etc.

## Bcrypt

biblioteca para criptografia de senha. Seu algoritmo é um dos mais seguros na atualidade, pois permite adicionar um código aleatório ao hash original, ou aumentar a dificuldade de decifrar o hash por adicionar um salt, permitindo utilizar o poder computacional para criptografar os dados.

## body-parser

biblioteca utilizada para converter o corpo de uma requisição em um determinado formato (utilizaremos o JSON).

## cookie-parser

Middleware analisador de cookies, usado para analisar os cookies anexados à solicitação feita pelo cliente ao servidor.

## Cors

Mecanismo usado para adicionar cabeçalhos HTTP que são informados aos navegadores para permitir que uma aplicação Web seja executada em uma origem e acesse recursos de outra origem diferente.

## Dotenv

Pacote utilizado para gerenciar as variáveis de ambiente dentro de um projeto Node.js. Essa ferramenta armazena a configuração dessas variáveis em um ambiente separado do código da aplicação.

## Helmet

Conjunto de funções de middlewares que pode ajudar a proteger a aplicação de algumas vulnerabilidades da web bastante conhecidas, configurando os cabeçalhos HTTP adequadamente.

## Joi

Módulo para validação de dados em endpoints.

## Knex

Query builder para NodeJs que é capaz de conectar com diversos tipos de bancos de dados. Com o Knex obetemos controle de fluxo assíncrono respostas padronizadas entre diferentes bancos. Conta também com um sistema de migrations.

## knex-stringcase

Utilizado para converter o padrão de nomenclatura utilizada no banco de dados para a aplicação Node.Js.

## Moment

Biblioteca utilizada para a manipulação de datas.

## sqlite3 e mysql

Bibliotecas que permitem a conexão com o banco de dados MySql.

# Tecnologias empregadas no Front-end.

## Vue.Js

Framework JavaScript open source para a criação de aplicações web. Possibilita criar aplicações de forma reativa. O Vue.js faz a utilização de um DOM virtual, o que faz com que seja extremamente performático na maioria das situações. Além disso, conta com uma arquitetura muito bem estruturada por meio da criação de componentes reusáveis. É muito utilizado para criar aplicações SPA (Single Page Applications).

## vue-datepicker

Componente personalizado que implementa um calendário selecionável.

## Axios

Client HTTP que funciona tanto do lado do Cliente (browser), quanto do lado do Servidor (NodeJS), possibilitando a manipulação dos dados vindos de uma API e as operações de CRUD.

## Bootstrap

Framework front-end que fornece estruturas de CSS para a criação de sites e aplicações responsivas de forma rápida e simples. Além disso, pode lidar com sites de desktop e páginas de dispositivos móveis da mesma forma.

## bootstrap-icons

Biblioteca de ícones padrão do Bootstrap.

## Moment

Biblioteca utilizada para a manipulação de datas.

## sass e sass-loader

Utilizando a mesma lógica do CSS (seletores, regras etc), mas de uma maneira mais organizada, intuitiva e com trechos de código facilmente reutilizáveis.

## v-money

Biblioteca para inserção de máscara para campos monetários.

## vue-the-mask

Biblioteca para inserção de máscaras em campos, de acordo com padrões estabelecidos pelo desenvolvedor.

## Vuex

Padrão de gerenciamento de estado + biblioteca para aplicações Vue.js. Ele serve como um store centralizado para todos os componentes em uma aplicação, com regras garantindo que o estado só possa ser mutado de forma previsível.