

# **Laudo de revisão**

## **Especificação dos requisitos:**

As especificações dos requisitos foram seguidas, as tecnologias selecionadas para o trabalho atenderam as necessidades do projeto. O Node.js juntamente com o Express.js funcionaram bem no backend do projeto e permitiram a implementação das features iniciais de cadastramento e armazenamento dos questionários. O MongoDB, o banco de dados utilizado, funcionou como esperado, entregando um serviço de banco de dados de uso simples e que conversou muito bem com o backend. O React.js foi utilizado no front-end e supriu as necessidades ao permitir que uma implementação inicial da interface tenha sido feita de forma simples. As ferramentas de documentação Jsdoc e o framework de testes Jest também atenderam as necessidades do projeto, e ofereceram a documentação de forma automática, e os testes também de forma automática, que contribuíram para o uso da metodologia TDD orientada a testes. Nenhuma escolha de tecnologia importante que foi utilizada foi modificada em relação a especificação de requisitos inicial.

## **Design do software**

O design do software teve enfoque no paradigma de orientação a objetos, tornando o usuário e os questionários duas entidades (classes) principais no início da implementação do projeto. O uso do padrão de design “repository pattern” ajudou a modularizar e simplificou a interface do banco de dados com o back-end, facilitando a implementação de novas features e impedindo que redundâncias em relação ao acesso ao banco de dados estejam presentes no código.

## **Fazer o código**

O código foi produzido com a ajuda da ferramenta de verificação estática de código, ESLint, que permitiu que erros de sintaxe fossem identificados rapidamente, facilitando a produção do código. A metodologia TDD também auxiliou o processo de produção do código, pois permite que novas features sejam implementadas e rapidamente testadas, para verificar a eficácia do código escrito.

## **Testar o código**

O teste do código foi realizado com a ajuda da ferramenta Jest, que automatiza o processo de teste do código escrito, permitindo que a metodologia TDD orientada a testes seja utilizada no trabalho de forma mais automática. O uso dessa ferramenta

permite acompanhar a eficiência do código escrito. Os testes foram verificados antes de serem utilizados para testar o código.

## **Depurar o código**

Algumas técnicas foram utilizadas para depurar o código, como o uso dos verificadores estáticos e dinâmicos de código, bem como o uso de breakpoints presentes no editor de código VSCode.