

## Tecnologias utilizadas:

- **NodeJs:**

Foi decidido que será utilizada para o desenvolvimento back-end por mostrar uma boa performance e aceitação da tecnologia em diversos projetos do mesmo tipo. Além disso, dispõe de uma vasta biblioteca e comunidade o que auxilia no processo de desenvolvimento, melhorando assim a qualidade e diminuindo o tempo de produção.

- **Flutter:**

Foi escolhida esta tecnologia para o desenvolvimento do front-end do software, pois é sabido que o framework é aceito por diferentes sistemas operacionais sem necessitar de muitas mudanças em seu código. Além disso, apresenta um bom desempenho nas suas aplicações, o que o torna a melhor opção no momento.

- **PostgreSQL:**

Foi escolhido como o sistema de banco de dados a ser utilizado, pois permite a utilização de estruturas de dados, tipagens e métodos criados pelos desenvolvedores, desta forma permite que o desenvolvimento e a manutenção do código sejam mais fáceis. Além disso, permite criar funções específicas que otimizam o processamento de informações utilizando os Stored Procedures em um sistema modular.

[RNF001] Portabilidade
O sistema deverá apresentar compatibilidade com diferentes sistemas operacionais mobile. *** COLOCAR SISTEMAS OPERACIONAIS ***
<b>Prioridade:</b> <input checked="" type="checkbox"/> Essencial <input type="checkbox"/> Importante <input type="checkbox"/> Desejável

### **[RNF002] Alta disponibilidade**

O sistema deverá estar disponível em 100% do tempo, de maneira que quando necessário realizar a suspensão temporária dos seus serviços para manutenção ou atualizações, seja definido o momento que gere menor impacto aos usuários do sistema.

**Prioridade:**      ☒ Essencial                      ☐ Importante                      ☐ Desejável

#### **RNF002.:**

Será utilizada uma redundância de servidores controlados por um gerenciador de carga, no qual fará a alternância entre os servidores caso algum deles apresente falhas.

### **[RNF003] Segurança de armazenamento dos dados**

Os dados gerados através do presente sistema deverão ser salvos em uma base de dados segura, e deve-se tentar ao máximo bloquear o acesso de pessoas não autorizadas.

**Prioridade:**      ☒ Essencial                      ☐ Importante                      ☐ Desejável

#### **RNF003.:**

Será utilizado a autenticação JWT para consultas do usuário ao servidor, com criptografia MD5 e Base64 das credenciais para solicitação do token e protocolo https para comunicação.

#### **[RNF004] Confiabilidade durante execução**

O sistema deverá ser capaz de lidar com diferentes exceções, se recuperando de comportamentos inesperados sem que haja perda de dados previamente armazenados, nem a necessidade de reiniciar o sistema ou realizar ações paliativas para a restauração do comportamento normal do sistema.

**Prioridade:**      ☒ Essencial              ☐ Importante              ☐ Desejável

#### **RNF004.: \*\*\* ALTERAR RESPOSTA \*\*\***

Todos os erros devem ser testados e tratados da melhor forma para que o risco a falhar seja o mínimo possível. Além disso, apenas no lado cliente será feita uma persistência de dados local para caso haja falha na comunicação com o servidor, os dados já cadastrados pelo usuário não seja perdido e seja enviado quando a conexão for restabelecida.

#### **[RNF005] Segurança dos dados**

O sistema de autenticação de usuário deverá ser criptografado, de modo que apenas usuários com credenciais válidas possuam acesso ao mesmo. Garantindo que os dados armazenados através do sistema estejam devidamente seguros.

**Prioridade:**      ☒ Essencial              ☐ Importante              ☐ Desejável

#### **RNF005 - Solução:**

Será necessário enviar o token de autenticação a cada requisição onde o servidor fará a validação e enviará uma resposta com os dados caso o token seja válido ou retornará uma exceção de usuário não autenticado.

### **[RNF006] Escalabilidade**

O sistema deve permitir o acesso simultâneo de diferentes usuários sem a perda de qualidade e informações durante a execução. A quantidade prevista de usuários para o sistema é de pelo menos 150 mil.

**Prioridade:**      ☒ Essencial                      ☐ Importante                      ☐ Desejável

**RNF006 - Solução: \*\* MELHORAR RESPOSTA - DÁ A ENTENDER QUE TODO O SISTEMA SERÁ ASSÍNCRONA \*\***

O sistema fará o processamento dos dados de maneira assíncrona, garantindo que seja possível atender um maior número de usuários ao mesmo tempo sem que o tráfego de dados seja congestionado.

### **[RNF007] Usabilidade de interface**

O sistema deverá apresentar uma interface simplificada, de forma que todas as funcionalidades sejam de fácil acesso, proporcionando aos usuários do sistema agradável experiência.

**Prioridade:**      ☒ Essencial                      ☐ Importante                      ☐ Desejável

### [RNF008] Manutenibilidade do Sistema

O sistema deverá possuir um código que facilite a manutenção do código ao longo do tempo e facilitando a escalabilidade do mesmo.

**Prioridade:**      ☒ Essencial              ☐ Importante              ☐ Desejável

#### RNF008 - Solução:

O sistema será modular e terá seu back-end baseado em microsserviços, facilitando a manutenibilidade e permitindo que a manutenção seja feita em pequenos pedaços sem necessidade de interromper todo o sistema para pequenas atualizações.

O back-end seguirá o padrão de Model-Controller em seus módulos, tendo em vista que esse padrão é amplamente usado e têm demonstrado sucesso nos projetos que utilizam NodeJs.

No aplicativo Flutter será utilizado os conceitos de SOLID, Clean Code e Clean Architecture, com uso extensivo de testes automatizados, para garantir qualidade do código e facilitar a leitura de código para a equipe de manutenção, esses conceitos já vem sendo utilizados e amplamente recomendados pelos desenvolvedores mais experientes nesta tecnologia.

Para mantermos a qualidade do projeto ficou acordado entre os desenvolvedores que será utilizado o package MobX para fazer a gerência de estados das Views que necessitarem de mudanças de estados e o package Flutter Modular para injeções de dependências e navegação entre telas, ambos já foram amplamente testados pela comunidade e apresentam uma excelente qualidade, atendendo nossas demandas com confiabilidade.