# **Arquitetura tecnológica – SANEM**

## **1. Levantamento de Tecnologias**

### **Back-end**

**Opção 1: Node.js (Express)**

* **Nível de dificuldade:** Baixo a intermediário para a equipe.
* **Justificativa da escolha:** Linguagem JavaScript já é familiar, ampla comunidade, boa performance e integração com plataformas modernas de hospedagem e banco de dados. Escolhida como tecnologia principal.

**Opção 2: Python (Flask)**

* **Nível de dificuldade:** Intermediário.
* **Justificativa alternativa:** Ideal para APIs simples, ótima documentação, mas requer mais configurações manuais em relação a Node.js.

### **Front-end**

**Opção 1: React + TypeScript**

* **Nível de dificuldade:** Intermediário.
* **Justificativa da escolha:** Oferece estrutura moderna, componente reutilizável, alta interatividade. TypeScript traz segurança adicional na tipagem. Escolhida como tecnologia principal.

**Opção 2: HTML + Bootstrap**

* **Nível de dificuldade:** Baixo.
* **Justificativa alternativa:** Boa para prototipagem rápida, mas não oferece flexibilidade nem escalabilidade para aplicações ricas em interação.

### **Banco de Dados**

**Opção 1: PostgreSQL**

* **Nível de dificuldade:** Intermediário.
* **Justificativa da escolha:** SGBD robusto e relacional, ideal para estrutura de dados consistente e rastreável como no sistema da SANEM. Escolhido como principal.

**Opção 2: MongoDB**

* **Nível de dificuldade:** Intermediário.
* **Justificativa alternativa:** NoSQL, indicado para dados sem estrutura fixa, mas menos adequado ao tipo de controle transacional necessário ao projeto.

## **2. Pesquisa de Ambientes de Hospedagem**

### **Front-end**

**Opção 1: Netlify**

* **Gratuito:** Sim.
* **Limitações:** Até 100GB de bandwidth/mês.
* **Facilidade de integração:** Deploy contínuo via GitHub, ótimo para projetos em React.
* **Escolha principal.**

**Opção 2: Vercel**

* **Gratuito:** Sim.
* **Limitações:** Uso contínuo pode exigir upgrade.
* **Facilidade de integração:** Integração nativa com GitHub, GitLab, Bitbucket.
* **Alternativa viável.**

### **Back-end**

**Opção 1: Railway**

* **Gratuito:** Sim (até 500 horas/mês e 1GB RAM).
* **Limitações:** Após o limite, plano pago necessário.
* **Facilidade de integração:** Interface moderna, integração rápida com banco de dados.
* **Escolha principal.**

**Opção 2: Render**

* **Gratuito:** Sim (até 750h/mês).
* **Limitações:** “Sleep mode” em planos gratuitos.
* **Facilidade de integração:** Boa documentação, deploy via Git.
* **Alternativa viável.**

### **Banco de Dados**

**Opção 1: Railway (PostgreSQL)**

* **Gratuito:** Sim (até 1GB).
* **Limitações:** Armazenamento limitado; plano pago a partir de US$5/mês.
* **Facilidade de integração:** Altíssima, mesma plataforma do back-end.
* **Escolha principal.**

**Opção 2: Supabase (PostgreSQL)**

* **Gratuito:** Sim (500MB de dados).
* **Limitações:** Recursos limitados nos planos gratuitos.
* **Facilidade de integração:** Interface semelhante ao Firebase, fácil de usar.
* **Alternativa viável.**

**3. Comparativo Resumido de Hospedagem**

| **Camada** | **Opção** | **Gratuidade** | **Limitações** | **Facilidade Integração** |
| --- | --- | --- | --- | --- |
| Front-end | Netlify | Sim | 100GB bandwidth/mês | Excelente com Git |
| Front-end | Vercel | Sim | Upgrade necessário para uso intenso | Muito boa |
| Back-end | Railway | Sim (500h/mês) | 1GB RAM, 1GB DB | Muito fácil |
| Back-end | Render | Sim (750h/mês) | Sleep em inatividade | Boa |
| Banco de dados | Railway (PostgreSQL) | Sim (1GB) | Armazenamento limitado | Totalmente integrada |
| Banco de dados | Supabase | Sim (500MB) | Limitação de espaço e recursos | Excelente documentação |

## **4. Stack Tecnológica e Plano de Hospedagem Final**

### **Tecnologias Escolhidas**

* **Back-end:** Node.js (Express)
* **Front-end:** React + TypeScript
* **Banco de dados:** PostgreSQL

### **Provedores de Hospedagem**

* **Front-end:** Netlify
* **Back-end:** Railway
* **Banco de dados:** Railway (PostgreSQL)

## **Custos Previstos**

* **Fase inicial:** Gratuito, dentro dos limites dos planos free dos provedores.
* **Escalonamento futuro:** Caso necessário, planos pagos estimados entre **US$5 a US$10/mês** por serviço, dependendo da escala e uso.

## **Justificativa Final**

A stack escolhida oferece o equilíbrio ideal entre facilidade de desenvolvimento, custo acessível e escalabilidade. O uso de React com TypeScript permite construção de interfaces modernas, enquanto Node.js com PostgreSQL garante robustez no back-end e integridade dos dados.

Os provedores selecionados (Netlify e Railway) são amplamente utilizados por projetos profissionais e acadêmicos, oferecem planos gratuitos robustos, documentação acessível, integração simplificada e fácil escalabilidade.

A arquitetura tecnológica atende aos requisitos funcionais e não funcionais do sistema da SANEM, como controle de estoque, cadastro de beneficiários, emissão de relatórios, segurança dos dados e compatibilidade com a LGPD.

Essa proposta proporciona um ambiente de desenvolvimento ágil, com baixo custo inicial e capacidade de adaptação conforme o crescimento do projeto.