

# LEVANTAMENTO E ANÁLISE DE REQUISITOS

---

## Product Backlog – Plataforma de Dados Limnológicos INPE

---

### Epic 1: Visualização de Dados

#### US01 - Painel Unificado de Parâmetros (RF01)

**Como** usuário,

**Quero** visualizar todos os parâmetros armazenados em um painel unificado,

**Para** obter uma visão abrangente e centralizada dos dados disponíveis.

#### Critérios de Aceitação:

- Exibir todos os parâmetros armazenados em um painel único;
- Garantir atualização automática dos dados exibidos;
- Layout responsivo e acessível em diferentes dispositivos;

*Prioridade: Alta | Story Points: 8*

---

#### US02 - Filtragem e Indicadores de Qualidade (RF02)

**Como** usuário,

**Quero** filtrar parâmetros por tipo e visualizar indicadores de qualidade,

**Para** analisar apenas os dados relevantes e verificar sua confiabilidade.

#### Critérios de Aceitação:

- Permitir filtragem por tipo de parâmetro;
- Apresentar indicadores de qualidade dos dados (status de coleta, integridade);
- Atualizar a visualização de acordo com os filtros aplicados;

*Prioridade: Alta | Story Points: 5*

---

#### US03 - Consulta de Dados em Tabela (RF02)

**Como** usuário,

**Quero** consultar e visualizar os dados em formato tabular,

**Para** facilitar a análise e interpretação estruturada das informações apresentadas.

#### Critérios de Aceitação:

- Exibir dados em tabela responsiva;
- Incluir paginação para grandes volumes de registros;
- Permitir ordenação por colunas (crescente e decrescente);
- Incluir campo de busca para filtragem rápida;
- Mostrar metadados relevantes (instituição, reservatório, data de coleta);

Prioridade: *Alta* | Story Points: 5

---

#### US04 - Exportação de Dados em CSV (RF03)

**Como** usuário,

**Quero** consultar e exportar os dados no formato CSV,

**Para** possibilitar sua utilização em outras ferramentas e atividades correlatas.

#### Critérios de Aceitação:

- Permitir exportação de todos os registros em CSV;
- Garantir compatibilidade com softwares comuns (Excel, Libre Office, R, Python);
- Preservar a estrutura e integridade dos dados no arquivo exportado;
- Incluir cabeçalhos e metadados no arquivo gerado;
- Possibilitar exportação de dados filtrados da tabela;

Prioridade: *Alta* | Story Points: 3

---

#### US05 - Localização de Dados em Mapa Interativo (RF04)

**Como** usuário,

**Quero** consultar e localizar os dados em um mapa interativo,

**Para** compreender espacialmente as informações apresentadas de maneira mais intuitiva.

#### Critérios de Aceitação:

- Exibir dados georreferenciados em mapa interativo;
- Permitir navegação com zoom e movimentação do mapa;
- Exibir informações detalhadas ao clicar em um ponto (pop-up);
- Permitir filtragem espacial (por região, reservatório ou ponto de coleta);
- Garantir atualização automática conforme novos dados forem inseridos;

Prioridade: *Média* | Story Points: 8

---

#### US06 - Exibição de Séries Temporais em Gráficos (RF05)

**Como** desenvolvedor,

**Quero** implementar a exibição de dados de séries temporais em formato gráfico,

**Para** fornecer aos usuários uma interpretação mais clara e eficiente das informações analisadas.

**Critérios de Aceitação:**

- Exibir gráficos de séries temporais responsivos;
- Permitir seleção de intervalos de tempo específicos;
- Oferecer diferentes tipos de gráficos (linha, barra, dispersão);
- Possibilitar comparação entre múltiplos parâmetros no mesmo gráfico;
- Permitir exportação dos gráficos em imagem (PNG, JPEG);

Prioridade: Média | Story Points: 8

---

## Epic 2: Filtros e Busca

### US07 - Filtros Avançados (RF06)

**Como** usuário,  
**Quero** filtrar dados por instituição, reservatório, período e parâmetros,  
**Para** analisar apenas os dados relevantes para minha pesquisa.

**Critérios de Aceitação:**

- Dropdown para instituições disponíveis;
- Seleção múltipla de reservatórios e parâmetros;
- Date picker para intervalo de tempo;
- Aplicação de filtros em tempo real;
- Indicação clara dos filtros ativos;

Prioridade: Alta | Story Points: 8

---

## Epic 3: Infraestrutura e Performance

### US08 - Configuração de Ambiente Containerizado (RP04)

**Como** desenvolvedor,  
**Quero** um ambiente containerizado para todas as camadas,  
**Para** garantir consistência e facilitar o processo de deploy.

**Critérios de Aceitação:**

- Docker Compose com PostgreSQL, Node.js e React;
- Variáveis de ambiente configuráveis;
- Scripts de inicialização de dados incluídos;
- Documentação de setup;

Prioridade: Alta | Story Points: 5

---

## US09a - API REST Backend Básica (RP02)

**Como** desenvolvedor,

**Quero** disponibilizar uma API REST inicial em Node.js/TypeScript,

**Para** permitir consultas básicas ao banco PostgreSQL de forma padronizada na MVP.

### Critérios de Aceitação:

- Endpoints RESTful básicos documentados (ex.: /dados, /parâmetros);
- Validação mínima de parâmetros de entrada;
- Tratamento inicial de erros (respostas HTTP claras);
- Deploy funcional e acessível via container básico;

*Prioridade: Alta | Story Points: 3*

---

## US10 - API REST Backend Completa (RP02)

**Como** desenvolvedor,

**Quero** disponibilizar uma API REST robusta e documentada,

**Para** garantir acesso seguro e auditável aos dados do PostgreSQL.

### Critérios de Aceitação:

- Endpoints RESTful completos com documentação detalhada (Swagger ou equivalente);
- Validação avançada de parâmetros (tipos, ranges, formatos);
- Tratamento consistente de erros com mensagens padronizadas;
- Logs de auditoria detalhados (requisições, respostas, falhas);
- Testes automatizados para os endpoints principais;

*Prioridade: Alta | Story Points: 8*

---

## US11 - Interface Responsiva Básica (RP03, RNF01)

**Como** usuário,

**Quero** acessar a plataforma por meio de uma interface simples e responsiva,

**Para** consultar dados em diferentes dispositivos sem dificuldades de navegação.

### Critérios de Aceitação:

- Implementação inicial em React/TypeScript;
- Layout responsivo para desktop e mobile;
- Navegação funcional entre páginas principais;

- Performance aceitável em dispositivos comuns;

*Prioridade: Alta | Story Points: 5*

---

## **US12 - Interface Responsiva Completa (RP03, RNF01, RNF02)**

**Como** usuário,

**Quero** utilizar uma interface intuitiva, consistente e otimizada,

**Para** ter uma experiência fluida e agradável em qualquer dispositivo.

### **Critérios de Aceitação:**

- Aplicação de design system consistente (cores, tipografia, componentes);
- Testes de usabilidade aprovados com usuários reais;
- Performance otimizada (RNF02) com carregamento rápido;
- Acessibilidade conforme padrões WCAG 2.1;
- Ajustes de refinamento baseados em feedback;

*Prioridade: Alta | Story Points: 8*

---

## **US13 - Otimização de Performance (RNF02)**

**Como** usuário,

**Quero** que os dados carreguem rapidamente,

**Para** garantir uma experiência fluida na plataforma.

### **Critérios de Aceitação:**

- Tempo de carregamento inferior a 3 segundos;
- Lazy loading implementado;
- Cache para dados mais acessados;
- Compressão de assets aplicada;

*Prioridade: Média | Story Points: 8*

---

## **Epic 4: Identidade Visual e UX**

### **US14 - Aplicação de Padrões Visuais INPE (RNF03)**

**Como** usuário,

**Quero** que a interface siga os padrões visuais do INPE,

**Para** manter consistência institucional e identidade visual.

### **Critérios de Aceitação:**

- Aplicação de cores e tipografia oficiais;
- Logotipos posicionados corretamente;
- Style guide documentado;
- Aprovação formal do cliente;

Prioridade: *Média* | Story Points: 5

---

## Epic 5: Funcionalidades Avançadas

### US15 - Comparação de Parâmetros

**Como** usuário,

**Quero** comparar parâmetros entre diferentes reservatórios,

**Para** realizar análises comparativas aprofundadas.

#### Critérios de Aceitação:

- Seleção múltipla de reservatórios;
- Visualização lado a lado dos dados;
- Gráficos comparativos integrados;
- Opção de exportar comparações;

Prioridade: *Baixa* | Story Points: 13

---

## Epic 6: Segurança e Autenticação

### US16 - Monitoramento de Erros e Alertas (RNF05)

**Como** desenvolvedor,

**Quero** coletar e monitorar logs de erro detalhados,

**Para** identificar falhas rapidamente e melhorar a estabilidade do sistema.

#### Critérios de Aceitação:

- Captura centralizada de erros (frontend e backend);
- Tratamento consistente de erros com mensagens claras ao usuário;
- Alertas automáticos (e-mail/Slack) em falhas críticas;
- Dashboard com estatísticas de erros recorrentes;

Prioridade: *Média* | Story Points: 8

---

## Epic 7: Testes e Qualidade

### US17 - Testes Automatizados de Backend (RNF06)

**Como** desenvolvedor,  
**Quero** testes automatizados para o backend,  
**Para** garantir a consistência e confiabilidade da API.

**Critérios de Aceitação:**

- Cobertura mínima de 80% do código;
- Testes unitários para serviços e controladores;
- Testes de integração com PostgreSQL;
- Execução automática em pipeline CI/CD;

*Prioridade: Alta | Story Points: 8*

---

**US18 - Testes de Interface e Usabilidade (RNF07)**

**Como** usuário,  
**Quero** que a interface seja validada por meio de testes automatizados e de usabilidade,  
**Para** ter uma experiência consistente e livre de falhas,

**Critérios de Aceitação:**

- Testes automatizados de UI com Cypress/Playwright;
- Validação de responsividade em diferentes dispositivos;
- Testes de acessibilidade (WCAG 2.1);
- Relatórios de usabilidade validados por grupo de teste;

*Prioridade: Média | Story Points: 5*