**Product Backlog - Sistema de Gestão Escolar (EscolaSystem)**

**Grupo:** Nefi / Lucas

**Visão Geral:** Este Product Backlog define as funcionalidades essenciais para um sistema web simplificado de gestão escolar, focado em cadastro de alunos, turmas, frequência e relatórios básicos. O sistema visa demonstrar a aplicação de conceitos de desenvolvimento web e persistência de dados local (LocalStorage).

**1. Gestão de Alunos**

* **História de Usuário:** Como administrador, quero cadastrar e gerenciar as informações dos alunos (nome, dados de contato, etc.) para manter um registro organizado e acessível.
* **Critérios de Aceitação:**
  + Permitir o cadastro, edição e exclusão de alunos.
  + Validar dados obrigatórios (nome).
  + Exibir uma lista de alunos com opção de busca.
* **Estimativa de Esforço:** 5 pontos

**2. Gestão de Turmas**

* **História de Usuário:** Como administrador, quero criar e gerenciar turmas, associando alunos a elas, para organizar as classes.
* **Critérios de Aceitação:**
  + Permitir a criação, edição e exclusão de turmas.
  + Permitir a associação de alunos a turmas.
  + Exibir uma lista de turmas com os alunos associados.
* **Estimativa de Esforço:** 5 pontos

**3. Registro de Frequência**

* **História de Usuário:** Como professor, quero registrar a frequência dos alunos por turma e data, para acompanhar a presença.
* **Critérios de Aceitação:**
  + Permitir a seleção de turma e data para o registro.
  + Exibir a lista de alunos da turma para marcar presença/ausência.
  + Salvar o registro de frequência.
* **Estimativa de Esforço:** 8 pontos

**4. Relatórios Básicos**

* **História de Usuário:** Como administrador, quero gerar relatórios simples para visualizar informações sobre alunos e turmas.
* **Critérios de Aceitação:**
  + Gerar uma lista de alunos por turma.
  + Gerar um resumo de frequência (ex: total de presentes/ausentes por aluno/turma).
* **Estimativa de Esforço:** 5 pontos

**5. Interface e Navegação**

* **História de Usuário:** Como usuário, quero uma interface clara e fácil de usar para acessar as funcionalidades do sistema.
* **Critérios de Aceitação:**
  + Utilizar Bootstrap para um layout responsivo.
  + Implementar uma sidebar para navegação entre os módulos.
  + Apresentar mensagens claras para o usuário.
* **Estimativa de Esforço:** 3 pontos

**6. Persistência de Dados (LocalStorage)**

* **História de Usuário:** Como desenvolvedor, preciso que os dados inseridos sejam mantidos mesmo após o fechamento do navegador, para simular um banco de dados.
* **Critérios de Aceitação:**
  + Utilizar LocalStorage para armazenar os dados de alunos, turmas e frequência.
  + Garantir que os dados sejam carregados corretamente ao abrir o sistema novamente.
* **Estimativa de Esforço:** 3 pontos

**Priorização:**

As funcionalidades foram priorizadas com base na sua importância para a demonstração do sistema e na facilidade de implementação inicial. A persistência de dados (LocalStorage) é fundamental para simular um sistema funcional, mesmo que em um ambiente local.