**Definição das Tarefas Técnicas-Sistema de Gestão Escolar (EscolaSystem)**

**Grupo:** Nefi / Lucas

**Objetivo:** Este documento detalha as tarefas técnicas necessárias para a implementação de cada funcionalidade do EscolaSystem, considerando o uso de HTML, CSS (Bootstrap) e JavaScript (com LocalStorage para persistência de dados).

**1. Módulo: Gestão de Alunos**

* **1.1. Estrutura de Dados de Alunos:**
  + Definir a estrutura JSON para armazenar dados de alunos no LocalStorage.
  + Implementar funções para salvar, carregar, atualizar e deletar dados de alunos do LocalStorage.
* **1.2. Interface de Cadastro/Edição de Alunos:**
  + Criar formulário HTML com campos para: Nome, Matrícula, CPF, Data de Nascimento, Turma (seleção), Status.
  + Implementar validação básica dos campos (ex: campo obrigatório).
* **1.3. Lógica de Negócio de Alunos:**
  + Função para gerar matrícula única automaticamente (simples).
  + Função para adicionar novo aluno (persistindo no LocalStorage).
  + Funções para editar e excluir alunos (persistindo no LocalStorage).
* **1.4. Listagem e Busca de Alunos:**
  + Criar tabela HTML para exibir a lista de alunos.
  + Implementar funcionalidade de busca por nome/matrícula na lista.

**2. Módulo: Gestão de Turmas**

* **2.1. Estrutura de Dados de Turmas:**
  + Definir a estrutura JSON para armazenar dados de turmas no LocalStorage.
  + Implementar funções para salvar, carregar, atualizar e deletar dados de turmas do LocalStorage.
* **2.2. Interface de Cadastro/Edição de Turmas:**
  + Criar formulário HTML com campos para: Nome da Turma, Série.
* **2.3. Lógica de Negócio de Turmas:**
  + Função para adicionar nova turma.
  + Funções para editar e excluir turmas.
* **2.4. Associação Aluno-Turma:**
  + Modificar formulário de aluno para incluir seleção de turma existente.
  + Garantir que a associação seja salva com os dados do aluno.
* **2.5. Listagem de Turmas:**
  + Criar tabela/lista para exibir as turmas cadastradas.

**3. Módulo: Registro de Frequência**

* **3.1. Estrutura de Dados de Frequência:**
  + Definir a estrutura JSON para armazenar registros de frequência (associados a turma e data).
  + Implementar funções para salvar e carregar dados de frequência do LocalStorage.
* **3.2. Interface de Seleção e Exibição:**
  + Criar seletores (dropdown) para Turma e campo de data.
  + Implementar carregamento dinâmico da lista de alunos com base na turma selecionada.
  + Criar tabela HTML para lista de chamada com opções "Presente/Ausente" (radio buttons) e campo para observações por aluno.
* **3.3. Lógica de Registro:**
  + Função para capturar os dados de presença/ausência e observações.
  + Função para salvar o registro de frequência para a turma e data selecionadas.
  + Atualizar contadores de Presentes/Ausentes dinamicamente na interface.

**4. Módulo: Relatórios Básicos**

* **4.1. Listagem de Alunos por Turma:**
  + Função para filtrar e exibir todos os alunos de uma turma selecionada.
* **4.2. Resumo de Frequência:**
  + Função para calcular e exibir totais de presentes e ausentes para uma turma em uma data específica.

**5. Módulo: Autenticação (Acadêmica)**

* **5.1. Estrutura de Dados de Usuários:**
  + Definir estrutura JSON para armazenar usuários (login, senha simples, papel) no LocalStorage.
* **5.2. Interface de Login:**
  + Criar formulário HTML para login (e-mail/usuário e senha).
* **5.3. Lógica de Autenticação:**
  + Função para validar credenciais de login contra os dados no LocalStorage.
  + Implementar redirecionamento após login bem-sucedido.
* **5.4. Controle de Acesso Simplificado:**
  + Lógica para exibir/ocultar links ou funcionalidades com base no papel do usuário (professor/administrador).

**6. Infraestrutura e Componentes Comuns**

* **6.1. Layout e Navegação:**
  + Estruturar o HTML com a sidebar e o conteúdo principal.
  + Aplicar estilos Bootstrap para um design responsivo.
  + Configurar links de navegação da sidebar.
* **6.2. Gerenciamento de Dados Locais:**
  + Funções utilitárias para localStorage.setItem() e localStorage.getItem() para cada tipo de entidade (alunos, turmas, frequência, usuários).
  + Inicialização de dados mockados se o LocalStorage estiver vazio.
* **6.3. Scripts JavaScript Globais:**
  + Carregamento dos scripts Bootstrap JS.
  + Scripts para inicialização e coordenação das funcionalidades entre as páginas.