

Requisitos Funcionais

1. Gerenciamento de Alunos

- O sistema deve permitir o cadastro, consulta, atualização e exclusão de alunos.
- O sistema deve listar os alunos por semestre, ano ou disciplina.

2. Cálculo de Média Acadêmica

- O sistema deve calcular a média ponderada dos alunos em cada disciplina usando a fórmula:

3. Cálculo de Presença

- O sistema deve calcular a porcentagem de presença do aluno em cada disciplina.

4. Classificação de Risco de Evasão e Reprovação

- O sistema deve categorizar o risco de cada aluno em três níveis (alto, médio e baixo) com base em presença, média, participação em atividades de sala e extracurriculares.
- Exibir a classificação de risco para cada aluno.

5. Formação de Grupos de Estudos

- O sistema deve formar grupos de estudo de cinco alunos cada, misturando alunos em risco com os de bom desempenho.

6. Relatórios e Visualizações

- O sistema deve gerar relatórios sobre desempenho acadêmico, presença e nível de risco dos alunos.
- Deve permitir a visualização de dados filtrados por disciplina, semestre ou ano.

7. Autenticação e Controle de Acesso

- O sistema deve exigir login e senha para acesso, com diferentes níveis de permissão para administradores, professores e alunos.

8. Exportação de Dados

- Permitir a exportação de relatórios e dados em formatos como PDF ou CSV.

Requisitos Não Funcionais

1. Usabilidade

- A interface deve ser intuitiva, com navegação simples e organizada.
- A solução deve oferecer uma experiência de usuário responsiva, permitindo acesso em dispositivos móveis e desktop.

2. Desempenho

- O sistema deve responder às consultas em até 3 segundos para evitar lentidão na navegação e nos cálculos.
- Deve ser capaz de suportar o uso simultâneo de até 100 usuários sem perda significativa de desempenho.

3. Segurança

- As informações dos alunos devem ser protegidas, usando criptografia para dados sensíveis como senhas.
- Deve implementar controle de acesso com autenticação para garantir que apenas usuários autorizados possam acessar informações específicas.

4. Escalabilidade

- O sistema deve ser projetado para suportar o aumento de dados sem perda de desempenho, com a possibilidade de integrar novas funcionalidades no futuro.

5. Confiabilidade e Robustez

- O sistema deve ser robusto, com tratamento de erros para entradas inválidas e mecanismos para evitar falhas críticas.
- Deve ter uma taxa de disponibilidade de pelo menos 99,9%, garantindo acesso estável e contínuo.

6. Manutenibilidade

- O código deve ser modular e bem documentado para facilitar futuras manutenções e atualizações.
- Deve seguir boas práticas de desenvolvimento, como padrões de codificação e versionamento adequado.

7. Compatibilidade

- O sistema deve ser compatível com os principais navegadores (Chrome, Firefox, Edge) e sistemas operacionais.