

## Pontos críticos de segurança do sistema

Os principais pontos de atenção do sistema são:

- Acesso ao banco de dados, que precisa ser controlado.
- Armazenamento de informações de alunos e professores.
- Manipulação de cursos e matrículas, que deve evitar erros ou duplicações.

## Estratégias aplicadas (ex: autenticação, controle de acesso, logs)

Para reforçar a segurança e a estabilidade do sistema, foram aplicadas algumas estratégias:

- Uso do padrão Singleton para manter uma única conexão com o banco de dados.
- Tratamento básico de exceções com `try/catch` nas operações JDBC.
- Validação simples de dados para evitar registros incompletos ou repetidos.
- Planejamento para futuras melhorias, como autenticação e controle de acesso por tipo de usuário (aluno, professor e administrador).

## Aplicação de princípios SOLID

- **Responsabilidade Única:** cada classe tem uma função específica (ex.: `AlunoDAO` lida apenas com dados de alunos).
- **Aberto/Fechado:** o código pode ser ampliado sem precisar alterar o que já funciona.
- **Separação de Interfaces e Dependências:** o sistema foi pensado para facilitar futuras expansões e mudanças sem afetar o restante do código.

## Separação clara de responsabilidades entre módulos

- **Model:** classes principais (Aluno, Professor, Curso).
- **DAO:** acesso e manipulação dos dados no MySQL.
- **Controller:** controle da lógica e ligação entre as partes.
- **View:** camada de apresentação (a ser desenvolvida em versões futuras).