

Algoritmos Chave & Lógica de Negócio

1. Fluxo de Autenticação do Usuário

1. O usuário insere o e-mail/número de matrícula e a senha.
2. O Frontend envia as credenciais para o endpoint /api/auth/login.
3. O Backend verifica as credenciais na tabela users.
4. Em caso de sucesso, um JSON Web Token (JWT) é gerado contendo o user_id e o role.
5. O JWT é enviado de volta para o cliente e armazenado de forma segura (ex: em um cookie HttpOnly).
6. Para todas as requisições de API subsequentes, o JWT é incluído no cabeçalho de Autorização.
7. Um middleware no backend verifica o JWT em cada rota protegida para autorizar o usuário e identificar seu papel.

2. Algoritmo de Distribuição de Livros (Lógica Atualizada)

Este processo é acionado quando um professor finaliza um class_program. A lógica prioriza livros que o aluno ainda não leu.

Função: distribuirLivrosParaPrograma(id_do_programa)

1. **Buscar Detalhes do Programa:** Obter o class_id, genre_id e a lista de alunos para o id_do_programa fornecido.
2. **Buscar Livros Candidatos:**
 - Consultar a tabela books para todos os livros onde:
 - genre_id corresponde ao gênero do programa.
 - status é 'disponível'.
 - Armazenar esses livros em uma lista, chamada livrosCandidatos.
3. Iterar Sobre os Alunos: Para cada aluno na turma:
 - a. Obter Dados do Aluno: Calcular a idade do aluno e buscar seu histórico de leitura dos últimos 12 meses na tabela reading_history. Armazenar os IDs dos livros lidos em um conjunto livrosLidos12Meses.
 - b. Filtrar por Idade e Leitura: Criar duas listas a partir de livrosCandidatos:
 - livrosNaoLidos: Livros apropriados para a idade do aluno e cujo ID NÃO está em livrosLidos12Meses.
 - livrosLidosRecentes: Livros apropriados para a idade do aluno e cujo ID ESTÁ em livrosLidos12Meses.
 - c. Tentar Atribuir Livro Não Lido (Prioridade 1):
 - Se a lista livrosNaoLidos não estiver vazia, selecionar aleatoriamente um livro dela. Este é o cenário ideal.
 - d. Tentar Atribuir Livro Lido Mais Antigo (Prioridade 2):

- Se livrosNaoLidos estiver vazia, significa que o aluno já leu todos os livros disponíveis no gênero nos últimos 12 meses.
- Nesse caso, consultar o reading_history para encontrar qual livro na lista livrosLidosRecentes foi lido há mais tempo (ou seja, tem a completion_date mais antiga). Atribuir esse livro.
- e. Finalizar Atribuição:
 - Se um livro foi selecionado no passo (c) ou (d):
 - Criar um novo registro na tabela program_assignments com os IDs do aluno, livro e programa.
 - Atualizar o status do livro na tabela books para 'atribuido'.
 - Remover o livro da lista livrosCandidatos para que não seja atribuído a outro aluno.
- f. Lidar com Falha na Atribuição:
 - Se nenhum livro puder ser atribuído (nenhum livro disponível se encaixa nos critérios), registrar o evento e notificar o professor para que ele possa fazer uma atribuição manual.

3. Lógica de Atribuição Manual pelo Professor

Para garantir flexibilidade, os professores podem sobreescriver a lógica automática a qualquer momento.

1. **Interface:** Na interface do professor, haverá uma opção para "Atribuir Manualmente" para cada aluno em um programa ativo.
2. **Seleção:** O professor poderá buscar e selecionar qualquer livro do acervo da biblioteca, independentemente de gênero, status ou histórico de leitura.
3. **Confirmação:** O sistema exibirá um aviso caso o livro esteja fora da faixa etária recomendada para o aluno ou se o aluno já o leu recentemente, mas permitirá que o professor confirme a sua decisão.
4. **Execução:** Após a confirmação, o sistema:
 - o Se houver um livro já atribuído ao aluno, seu status voltará a ser 'disponível'.
 - o O novo livro selecionado terá seu status alterado para 'atribuido'.
 - o O registro em program_assignments será criado ou atualizado para refletir a atribuição manual.

4. Processo de Devolução de Livros (para Empréstimos e Programas)

1. Um Bibliotecário escaneia ou insere o ISBN/ID do livro.
2. O sistema encontra o registro ativo de rental ou program_assignment para aquele livro.
3. **Atualizar Status do Livro:** O status do livro na tabela books é alterado de volta para 'disponível'.
4. **Registrar no Histórico:** Uma nova entrada é criada na tabela reading_history com o ID do aluno, o ID do livro, a data atual e a origem ('emprestimo' ou 'programa').
5. **Finalizar Empréstimo:**

- Para um rental, a return_date é definida, e quaisquer multas por atraso são calculadas e sinalizadas.
- Para uma program_assignment, o registro é simplesmente marcado como concluído.