

SCC-0240 – Bases de Dados
Profa. Mirela Teixeira Cazzolato
Monitor: Leonardo Campos
1S / 2025

Projeto Final da Disciplina

1. Descrição do Problema

Uma instituição de ensino (escola) necessita de um sistema online para gerenciar suas atividades acadêmicas e administrativas, facilitando a interação entre alunos, professores e a administração. A escola oferece diversos cursos e disciplinas em diferentes níveis de ensino.

Este trabalho visa criar uma base de dados para armazenar as informações envolvidas nas atividades da escola. Dentre essas informações está a descrição dos Cursos oferecidos, que deve estar identificada por pelo menos um nome, um código único e o departamento ao qual pertence. Cada curso é classificado por um nível de ensino (ex: fundamental, médio, técnico, graduação) e sua descrição deve incluir a carga horária total, o número de vagas, e pode indicar ainda a ementa, os pré-requisitos (outros cursos ou disciplinas) e a sala de aula padrão onde é ministrado.

Cada Disciplina que compõe um curso deve indicar quantas aulas semanais ela possui e o material didático básico recomendado. Cada curso pode ter diversas Regras (ex: frequência mínima obrigatória, critérios de aprovação) e diversas Necessidades de Infraestrutura (ex: laboratório de informática, projetor, lousa digital).

A escola contempla três tipos principais de Usuários: o Professor, que é quem ministra as disciplinas; o Aluno, que se matricula nas disciplinas; e o Funcionário Administrativo, que gerencia as operações. Todos os usuários têm um nome, sobrenome, data de nascimento, endereço, sexo, número de telefone, endereço de e-mail e uma senha para acesso ao sistema. Assuma que não existem dois usuários com o mesmo nome, sobrenome e número de telefone.

Cada Matrícula é feita por um aluno em uma ou mais disciplinas de um curso para um determinado período letivo (ex: semestre ou ano). Cada matrícula deve incluir também uma data da matrícula (quando foi efetivada), o status da matrícula (ex: ativa, trancada, concluída, reprovada) e as notas obtidas pelo aluno naquela disciplina. Pode incluir também informações sobre bolsas de estudo ou descontos aplicados. Note-se que cada matrícula em uma disciplina se refere a um único aluno e a uma oferta de disciplina específica. Uma matrícula pode ser confirmada ou cancelada até uma data limite antes do início do período letivo, e se confirmada, pode envolver o pagamento de taxas pelo aluno.

Cada disciplina pode ter muitas matrículas, mas o número de alunos matriculados não pode exceder a capacidade da turma ou da sala de aula designada. Cada disciplina é de responsabilidade de um ou mais professores, e cada professor pode ter informações sobre sua área de especialização e titulação. A escola também mantém um registro de Departamentos Acadêmicos, onde cada departamento possui um nome, um código único e um professor designado como chefe do departamento. Um departamento só pode ter um chefe.

Além disso, cada curso, disciplina, aluno e professor estão associados a uma Unidade da Escola (campus ou localidade específica), que corresponde à cidade, estado e país, e possivelmente a um prédio ou bloco específico dentro da unidade.

A comunicação no sistema é feita por meio de Avisos e Mensagens. Um professor pode enviar 0 ou mais mensagens para seus alunos (individualmente ou para a turma) e vice-versa. A administração também pode enviar avisos gerais. Cada mensagem é identificada por quem a enviou, o destinatário (ou grupo de destinatários) e o timestamp de sua criação, e deve incluir o texto da mensagem.

Um aluno pode Avaliar zero ou mais disciplinas e/ou professores ao final de um período letivo. Cada avaliação pode incluir um comentário em texto, e classificações para didática do professor, material de apoio, relevância do conteúdo e infraestrutura da sala.

2. Tarefas

2.1. Modelagem Conceitual

Exercício 1: Descreva, em forma de texto corrido, todas as **conjuntos entidades** (CE) identificados na problemática e seus respectivos **atributos**. Em seguida, identifique todas as **conjuntos relaciona-**

mentos (CRs) entre as entidades, sempre explicando a lógica por trás da escolha da *multiplicidade* e participação de cada entidade na relação. Identifique também os conjuntos entidades fracos (CEF, se houver), juntamente com CRT que o define como fraco, atributos multivalorados, atributos compostos, atributos de CRs.

Exercício 2: Faça a Modelagem E-R da base de dados descrita para o sistema de gestão escolar, utilizando o Modelo Entidade-Relacionamento Estendido (ME-RX). Seu diagrama deve representar explicitamente pelo menos uma abstração de Generalização/Especialização e uma de Agregação, caso se apliquem naturalmente ao problema.

2.2 Mapeamento para o Modelo Relacional e Normalização

Exercício 3: Faça o mapeamento completo do Modelo Entidade-Relacionamento Estendido (ME-RX), obtido no Exercício 2, para o Modelo Relacional. Sua resposta deve incluir a definição de todas as relações, seus atributos com os respectivos domínios, as chaves primárias, chaves candidatas e chaves estrangeiras. Justifique as decisões de mapeamento para construções mais complexas (ex: generalizações/especializações, atributos multivalorados, diferentes multiplicidades).

Exercício 4: Popule as tabelas resultantes do mapeamento (Exercício 3) com dados criados manualmente ou de forma sintética. Cada tabela principal (ex: Aluno e Professor) deve conter no mínimo 20 registros, e as tabelas resultantes de CRs (ex: Matrícula) devem conter um volume de dados consistente que permita a execução das consultas futuras. Os dados devem ser entregues em formato de scripts SQL de inserção (.sql) junto à atividade.

Exercício 5: Analise o esquema relacional obtido no Exercício 3 e justifique a forma normal (ex: 1FN, 2FN, 3FN, BCNF) em que cada tabela se encontra, logo após a modelagem. Depois, com base nas dependências funcionais identificadas, normalize toda a base até a BCNF (considerando as que ainda não se encontram nas FNs). Ao aplicar o processo de normalização, detalhe os passos seguidos e apresente o esquema relacional final, normalizado.

2.3 Implementação (SQL) e Demonstração

Exercício 6: Desenvolva scripts SQL para:

- Criar a estrutura do banco de dados (tabelas, restrições de integridade como chaves, etc.) no SGBD relacional PostgreSQL.
- Inserir os dados criados no Exercício 4.
- Implementar um conjunto de pelo menos 7 consultas SQL significativas que recuperem informações relevantes e demonstrem a capacidade do sistema escolar (ex: listar todos os alunos matriculados em uma disciplina específica em um determinado período; calcular a média de notas de um aluno; listar os professores de um departamento com suas respectivas disciplinas; encontrar cursos que não tiveram alunos matriculados no último período letivo; etc.).

Exercício 7: Escolha e crie pelo menos 3 índices em sua base de dados (de diferentes tipos):

- Justifique as escolhas para cada índice criado;
- Consulte informações sobre o plano de consulta, gerado com e sem o índice criado. Dica: para mostrar o uso do índice e o impacto dele sobre as consultas, recomenda-se fazer nova carga nos dados, aumentando significativamente a cardinalidade das relações, para refletir a diferença nos tempos de consulta. Não serão aceitas justificativas de falta de diferença no tempo das consultas devido ao baixo volume dos dados. A definição de cada tamanho faz parte da resolução do exercício.

Exercício 8: Crie pelo menos 3 visões em sua base de dados. Apresente as consultas utilizadas, que obrigatoriamente devem possuir junções entre (pelo menos) duas outras relações.

Exercício 9: Grave um vídeo (máximo 5 minutos) realizando a execução das consultas / comandos SQL implementados nos Exercícios 6-8. O vídeo deve mostrar claramente as consultas / comandos sendo

executados no ambiente do banco de dados e os resultados obtidos. Disponibilize o vídeo no YouTube (não listado ou público) e inclua o link no PDF final enviado.

2.4 Reflexão

Exercício 10: Descreva, de forma sucinta e coesa, quais foram as principais dificuldades encontradas durante a construção deste trabalho e quais observações ou aprendizados você julga importantes ao finalizar o projeto.

Instruções de entrega (IMPORTANTE)

1. Todos os grupos devem usar o Draw.io (<https://app.diagrams.net>) para as tarefas de modelagem E-R. A entrega da modelagem (Exercício 2) deve ser em uma única página no arquivo .pdf final. Se entregue de outra forma, será considerada a nota 0 (zero) para esta parte específica.
2. O grupo deverá disponibilizar um link para um repositório no GitHub público, contendo todos os scripts SQL e um arquivo README.md com instruções claras para configurar e executar o banco de dados e as consultas.
3. O trabalho completo deve ser entregue como um **único arquivo** em formato .pdf contendo todas as atividades textuais (em forma de relatório), e os links requeridos (GitHub, YouTube). Os scripts SQL e arquivos de dados devem estar no repositório GitHub.
4. Cada etapa deve ser acompanhada de uma explicação detalhada, primariamente em forma de texto corrido. O uso excessivo de tópicos em substituição a parágrafos explicativos não é permitido; no entanto, listas curtas e numeradas podem ser usadas para enumerar itens (como atributos, regras, passos de um processo) de forma clara dentro da explicação principal, visando a clareza da argumentação. A falta de uma explanação textual coesa e bem fundamentada poderá levar à penalização da questão ou do trabalho.
5. Trabalhos com seções ou diagramas não legíveis (baixa resolução, texto muito pequeno, etc.) serão penalizados, podendo inclusive serem zerados.
6. Cópias totais ou parciais de trabalhos entre grupos serão zeradas para todos os grupos envolvidos. O mesmo se aplica para partes do trabalho identificadas como utilização de IA Generativa (como ChatGPT, Gemini, dentre outros).
7. A não entrega ou a entrega deficiente de uma das partes principais do projeto (Modelagem Conceitual, Mapeamento e Normalização, Implementação SQL) comprometerá a avaliação das demais questões, podendo levar à anulação completa do trabalho.
8. Atrasos na entrega não serão permitidos.

Prazo e avaliação

Prazo de entrega: 22/6/2025, até 23h59.

Entrega: A entrega será via e-disciplinas, em tarefa dedicada a esse projeto. Na entrega do material, deverão ser indicados os links do GitHub e YouTube, conforme detalhado nos itens 2 e 3.

A correção do trabalho levará em consideração os commits do GitHub, as datas dos commits, e os autores dos mesmos. Atualizações no GitHub feitas depois do prazo de entrega não serão aceitas, e poderão levar ao cancelamento da avaliação daquela questão para o grupo todo.

Avaliação: A entrega do projeto é por grupo (somente o representante deve fazer a entrega).

A nota final do projeto é individual: somente terão nota diferente de zero aqueles que realizarem a apresentação.