

# Banco de Dados

## Projeto Final

Isabela Souza Sisnando De Araujo - 231018884

João Henrique Jácomo Lemes - 231018893

Luca Heringer Megiorin - 231003390

Universidade de Brasília

11 de Julho, 2025

## 1 Introdução

Para este projeto da disciplina de Banco de Dados, o grupo optou por realizar um sistema para facilitar a gestão de atividades de extensão da Universidade de Brasília, bem como o cadastro de alunos, participantes e coordenadores. O sistema permite uma análise do perfil de cada aluno, professor e profissionais externos à universidade, de modo que os professores organizadores poderão entender melhor sobre aqueles que desejam participar (contribuindo) de cada atividade, ou seja, não é para a inscrição em eventos de extensão como público, e sim como participante na organização da atividade.

Cada aluno possui uma matrícula de 9 dígitos que o identifica, bem como um nome, CPF, senha, índice de rendimento acadêmico, data de ingresso, semestre atual e pode ter vários emails. Os docentes também são identificados por uma matrícula (de 11 dígitos) e possuem um nome, CPF, senha e podem ter vários emails. Por fim, uma pessoa (externa à UnB) é identificada por um CPF, tem um nome, senha e pode ter um arquivo para o seu currículo, bem como ter vários emails.

Todo aluno está associado a apenas um curso, que tem um nome e é identificado por um código, e todo curso deve fazer parte de pelo menos um departamento, e pode ter vários alunos. Os departamentos são identificados por um código e possuem um nome. Eles podem ter vários cursos, várias matérias e vários professores (docentes). Toda matéria tem um nome e um código, e pertence a apenas um departamento e todo aluno participa de pelo menos alguma matéria em algum semestre, podendo ver qual foi a sua menção em cada matéria que já cursou (em alguns casos, pode cursar a mesma matéria em mais de um semestre).

Cada aluno, docente ou pessoa externa pode exercer uma função em vários projetos, dependendo se sua inscrição for aprovada ou não, enquanto o docente que exerce a função de coordenador sempre está como admitido em sua condição de inscrição.

Uma atividade de extensão é definida por um código, possuindo um título, um tipo de ação (evento, curso ou projeto), uma área temática, uma descrição, um público interno e externo estimado, bem como um público-alvo interno e externo. Ela possui uma data para o início de sua realização (quando foi criada) e de fim da sua realização. Essas atividades podem ter vários alunos, docentes e pessoas do público externo participando e exercendo funções diversas. Ela pode ter várias situações associadas a ela (data de análise, data de aprovação, data de real início das atividades, por exemplo). Uma situação é definida pela data em que ocorreu e pela atividade de extensão a ela relacionada.

Uma atividade de extensão também necessariamente tem um local (mesmo que seja apenas o local do departamento do professor coordenador). Este local é identificado por meio de um código, possui um nome, um tipo (escola, universidade, outro) e um endereço, com o estado, município, bairro e complemento. Várias fotos podem ser adicionadas a uma atividade de extensão, sendo que toda foto existente deve se relacionar a apenas uma atividade de extensão. Cada foto tem o seu código identificador, uma descrição e o arquivo de foto em si.

Por fim, uma extensão pode ter vários feedbacks de alunos, docentes ou pessoas externas. Um feedback tem um código para a sua identificação, bem como uma data de envio, nota (de 1 a 5) e um comentário. Todo feedback deve pertencer a apenas uma atividade de extensão e tem apenas um autor, seja esse um aluno, docente ou pessoa, os quais podem fazer diversos feedbacks.

Com base nas ideias do grupo, foram definidas as seguintes entidades:

- |                 |                          |
|-----------------|--------------------------|
| 1. Aluno        | 7. Extensão              |
| 2. Curso        | 8. Foto                  |
| 3. Departamento | 9. Local                 |
| 4. Matéria      | 10. Situação da Extensão |
| 5. Docente      | 11. Feedback             |
| 6. Pessoa       |                          |

Neste documento consta o Modelo Entidade Relacionamento para o Banco de Dados descrito acima (seção 2), o Modelo Relacional (seção 3), a forma normal de 5 tabelas do modelo (seção 4) e algumas consultas em álgebra relacional (seção 5). Os modelos foram feitos usando a ferramenta yEd e o script SQL foi escrito no PostgreSQL e podem ser vistos no repositório do GitHub (apêndice A).

## 2 Modelo Entidade Relacionamento

Abaixo consta o modelo entidade relacionamento desenvolvido para o sistema de atividades de extensão. O modelo também pode ser visto no repositório do GitHub (apêndice A).

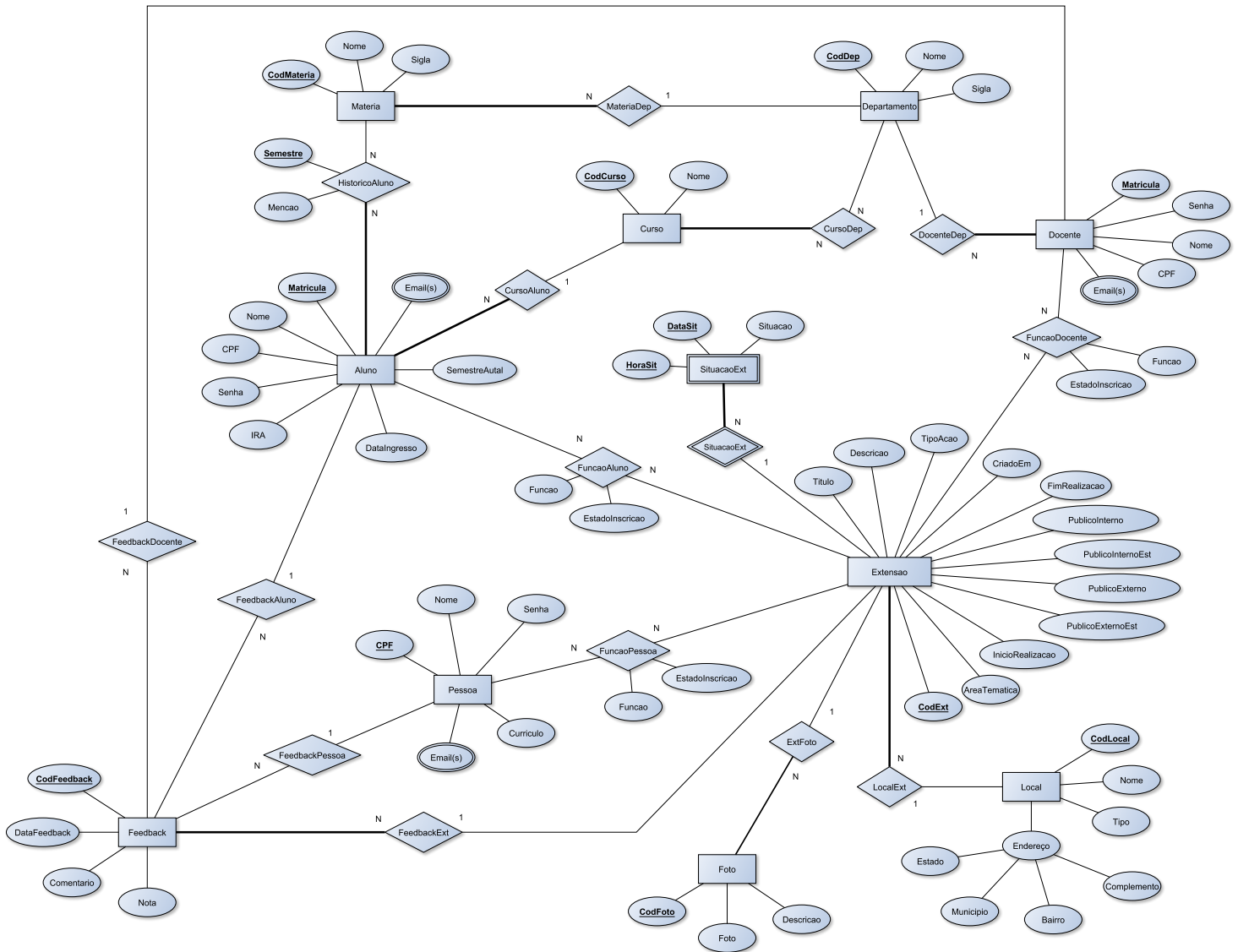


Figura 1: Modelo Entidade Relacionamento do Banco de Dados do projeto

### 3 Modelo Relacional

Abaixo consta o modelo relacional para o sistema de atividades de extensão. O modelo também pode ser visto no repositório do GitHub (apêndice A).

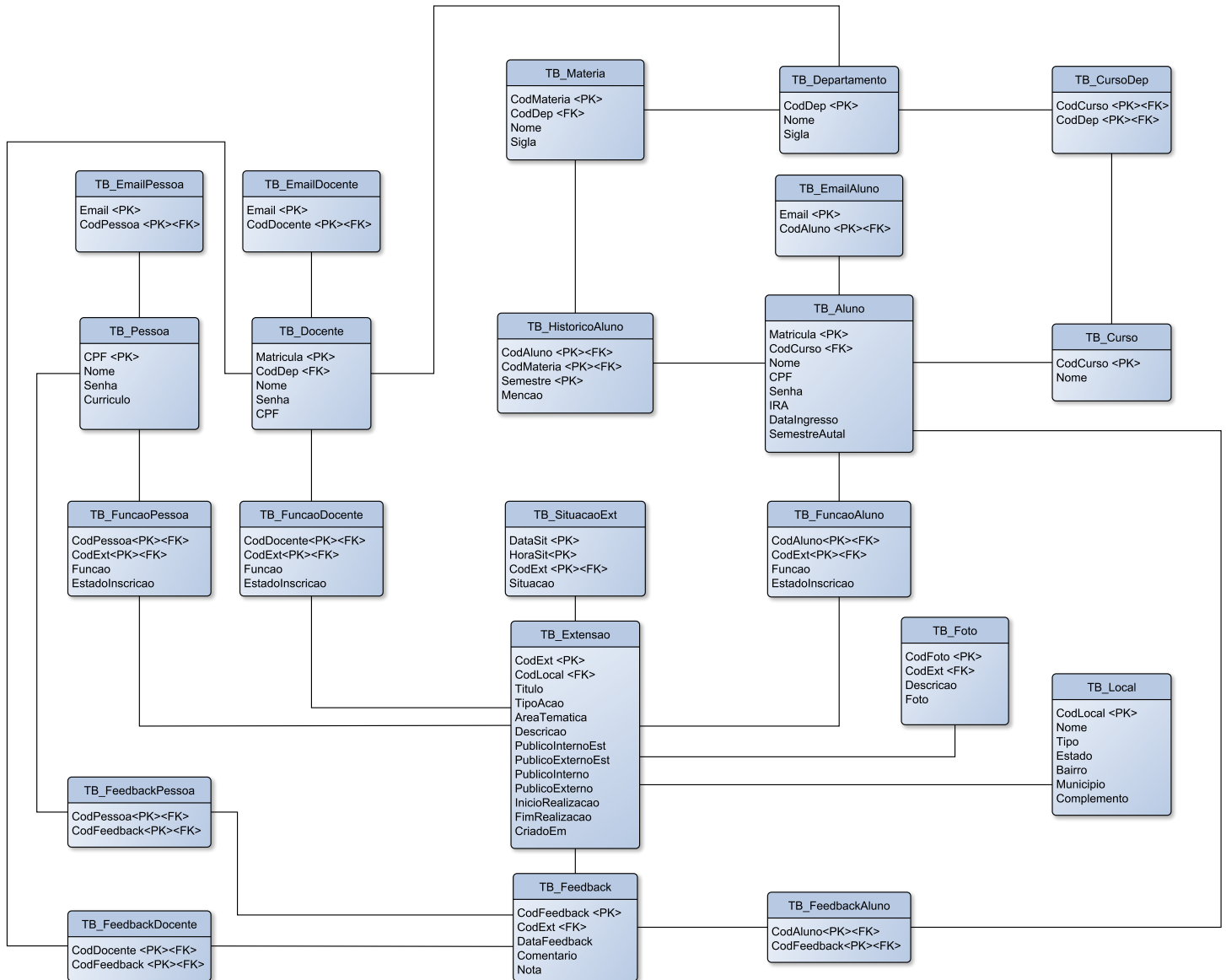


Figura 2: Modelo Relacional do Banco de Dados do projeto

### 4 Formas normais

Com o objetivo de verificar a normalização do banco de dado, foram escolhidas 5 tabelas do modelo acima. Essas tabelas foram desnormalizadas e, posteriormente, normalizadas novamente para provar que já estavam na terceira forma normal.

## 4.1 Primeira Forma Normal (1FN)

Juntando 5 tabelas (Extensao, Feedback, SituacaoExt, Foto e Local), percebe-se que a tabela resultante está, de fato, na primeira forma normal, dado que nenhum dos atributos presentes nela são atributos multivalorados, ou seja, o valor de uma coluna de uma tabela é indivisível.

Contanto, ainda existem problemas de redundância de dados, inclusão, atualização, e remoção, sendo necessário aplicar o próximo passo, a segunda forma normal.

## 4.2 Segunda Forma Normal

Estando na primeira forma normal, pode-se aplicar as regras da segunda forma normal. Neste caso, tem-se que o menor número de chaves candidatas é dado pelas chaves CodExt, CodFoto, CodFeedback, DataSit e HoraSit. A partir disso, sendo ( $\rightarrow$ ) o símbolo que indica a dependência funcional, temos que:

CodExt  $\rightarrow$  Titulo, TipoAcao, AreaTematica, Descricao, PublicoInternoEst, PublicoExternoEst, PublicoInterno, PublicoExterno, InicioRealizacao, FimRealizacao, CriadoEm, CodLocal, Nome, Tipo, Estado, Bairro, Municipio, Complemento

CodFeedback  $\rightarrow$  DataFeedback, Comentario, Nota

CodFoto  $\rightarrow$  CodExt, Foto

{CodExt, DataSit, HoraSit}  $\rightarrow$  Situacao

TB_1FN
CodExt <PK>
Titulo
TipoAcao
AreaTematica
Descricao
PublicoInternoEst
PublicoExternoEst
PublicoInterno
PublicoExterno
InicioRealizacao
FimRealizacao
CriadoEm
CodLocal
Nome
Tipo
Estado
Bairro
Municipio
Complemento
DataSit <PK>
HoraSit <PK>
Situacao
CodFoto <PK>
Descricao
Foto
CodFeedback <PK>
DataFeedback
Comentario
Nota

Figura 3: Tabelas desnormalizadas para a primeira forma normal

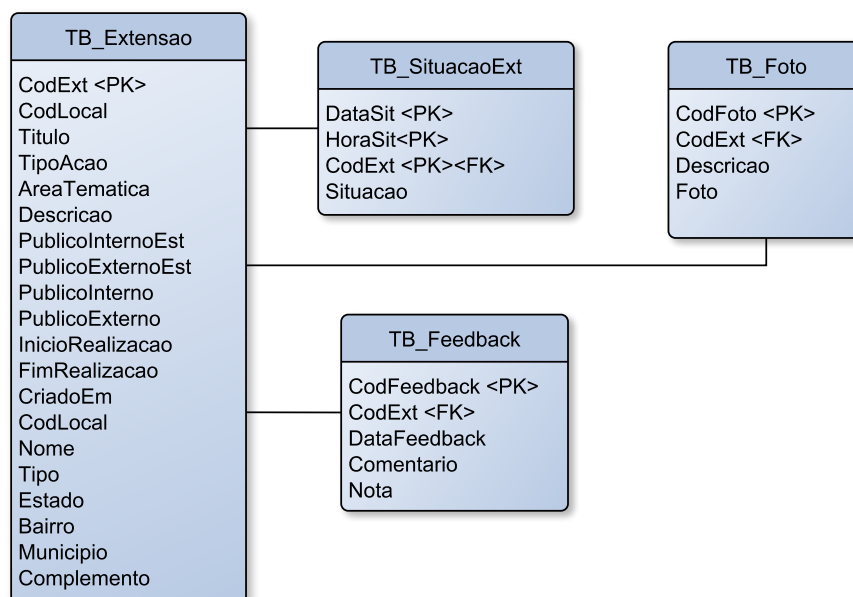


Figura 4: Tabelas na segunda forma normal

Com as definições acima, foram obtidas as tabelas as tabelas SituacaoExt, Foto e Feedback e Extensao, todas na segunda forma normal, pois todos os componentes complementares às chaves candidatas dependem totalmente funcionalmente de alguma combinação delas, ou seja, em cada tabela, todos os complementos das chaves primárias dependem totalmente funcionalmente das chaves primárias daquelas tabelas. Porém, ainda há alguns problemas de normalização, sendo necessário aplicar regras da terceira forma normal para terminar o processo.

### 4.3 Terceira Forma Normal

Analisando as tabelas obtidas, percebe-se um problema de transitividade na tabela Extensao. A sua chave (CodExt) define funcionalmente os seguintes atributos:

CodExt → Titulo, TipoAcao, AreaTematica, Descricao, PublicoInternoEst, PublicoExternoEst, PublicoInterno, PublicoExterno, InicioRealizacao, FimRealizacao, CriadoEm, CodLocal, Nome, Tipo, Estado, Bairro, Municipio, Complemento

Todavia, o atributo CodLocal é capaz de definir funcionalmente os seguintes atributos:

CodLocal → Nome, Tipo, Estado, Bairro, Municipio, Complemento

Assim, percebe-se que Nome, Tipo, Estado, Bairro, Municipio, Complemento são dependentes transitivos de CodExt, o que não é permitido para a terceira forma normal. Para isso, separa-se a tabela Extensao nas tabelas Local e Extensao. O resultado são as tabelas abaixo, que estão na terceira forma normal, pois estavam na segunda forma normal e todos os atributos não-chave são dependentes não-transitivos das chaves primárias de suas respectivas tabelas:

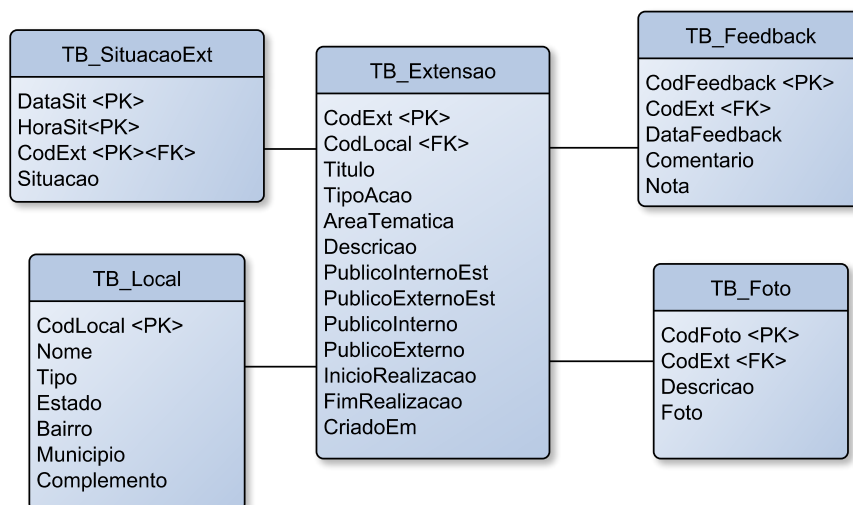


Figura 5: Tabelas na terceira forma normal

## 5 Álgebra Relacional

Conforme as especificações do projeto, seguem cinco consultas de álgebra relacional, envolvendo pelo menos 3 tabelas presentes no banco de dados em cada uma:

### 5.1 Consulta dos nomes dos alunos participantes em extensões e dos nomes das extensões em que participam

$$\begin{aligned} \text{ALUNOS\_FUNC\_EXT} &\leftarrow \text{TB\_Extensao} \bowtie_{\text{TB\_Extensao.CodExt}=\text{TB\_FuncaoAluno.CodExt}} \text{TB\_FuncaoAluno} \\ &\pi_{\text{Nome, Titulo}}(\text{TB\_Aluno} \bowtie_{\text{Matricula}=\text{CodAluno}} \text{ALUNOS\_FUNC\_EXT}) \end{aligned}$$

### 5.2 Consulta dos nomes das extensões e suas respectivas situações e docentes coordenadores

$$\begin{aligned} \text{SIT\_EXT} &\leftarrow \text{TB\_Extensao} \bowtie_{\text{TB\_Extensao.CodExt}=\text{TB\_SituacaoExt.CodExt}} \text{TB\_SituacaoExt} \\ \text{FUNCDOC\_SIT\_EXT} &\leftarrow \text{TB\_FuncaoDocente} \bowtie_{\text{TB\_FuncaoDocente.CodExt}=\text{SIT\_EXT.CodExt}} \text{SIT\_EXT} \\ \text{COORD\_SIT\_EXT} &\leftarrow \text{TB\_Docente} \bowtie_{\text{TB\_FuncaoDocente.CodExt}=\text{SIT\_EXT.CodExt}} \text{FUNCDOC\_SIT\_EXT} \\ &\pi_{\text{Titulo, Nome, Situacao}}(\sigma_{\text{Funcao}='Coordenador(a)}(\text{COORD\_SIT\_EXT})) \end{aligned}$$

### 5.3 Consulta dos nomes dos alunos e de seus respectivos cursos e departamentos

$$\begin{aligned} \text{AlunoCurso} &\leftarrow \text{TB\_Aluno} \bowtie_{\text{TB\_Aluno.CodCurso}=\text{TB\_Curso.CodCurso}} \text{TB\_Curso} \\ \text{CursoDepAluno} &\leftarrow \text{TB\_CursoDep} \bowtie_{\text{TB\_CursoDep.CodCurso}=\text{AlunoCurso.CodCurso}} \text{AlunoCurso} \\ \text{DepCursoAluno} &\leftarrow \text{TB\_Departamento} \bowtie_{\text{TB\_Departamento.CodDep}=\text{CursoDepAluno.CodDep}} \text{CursoDepAluno} \\ &\pi_{\text{TB\_Departamento.Nome, TB\_Curso.Nome, TB\_Aluno.Nome}}(\text{DepCursoAluno}) \end{aligned}$$

## 5.4 Consulta dos nomes dos docentes que participam em extensão, seus departamentos e suas extensões

$$\begin{aligned}\text{FuncDocExt} &\leftarrow \text{TB\_Extensao} \bowtie_{\text{TB\_Extensao.CodExt}=\text{TB\_FuncaoDocente.CodExt}} \text{TB\_FuncaoDocente} \\ \text{DocExt} &\leftarrow \text{TB\_Docente} \bowtie_{\text{Matricula}=\text{CodDocente}} \text{FuncDocExt} \\ \text{DepDocExt} &\leftarrow \text{TB\_Departamento} \bowtie_{\text{TB\_Departamento.CodDep}=\text{DocExt.CodDep}} \text{DocExt} \\ \pi_{\text{TB\_Departamento.Nome}, \text{TB\_Docente.Nome}, \text{Titulo}}(\text{DepDocExt})\end{aligned}$$

## 5.5 Consulta dos detalhes dos feedbacks de um docente coordenador em 2025 (com o nome do docente e da extensão ao qual o feedback foi direcionado)

$$\begin{aligned}\text{FuncDocExt} &\leftarrow \text{TB\_Extensao} \bowtie_{\text{TB\_Extensao.CodExt}=\text{TB\_FuncaoDocente.CodExt}} \text{TB\_FuncaoDocente} \\ \text{DocExt} &\leftarrow \text{TB\_Docente} \bowtie_{\text{Matricula}=\text{CodDocente}} \text{FuncDocExt} \\ \text{CoordExt} &\leftarrow \sigma_{\text{Funcao}='Coordenador'}(\text{DocExt}) \\ \text{FBCoordExt} &\leftarrow \text{TB\_Feedback} \bowtie_{\text{TB\_Feedback.CodExt}=\text{CoordExt.CodExt}} \text{CoordExt} \\ \text{FBDocCoordExt} &\leftarrow \text{TB\_FeedbackDocente} \bowtie_{\text{CodDocente}=\text{Matricula}} \text{FBCoordExt} \\ \text{FB2025CoordExt} &\leftarrow \sigma_{\text{DataFeedback}='2025-xx-xx'}(\text{FBDocCoordExt}) \\ \pi_{\text{TB\_Docente.Nome}, \text{Titulo}, \text{TB\_Extensao.CodExt}, \text{TB\_Feedback.Comentario}, \text{TB\_Feedback.Nota}}(\text{FB2025CoordExt})\end{aligned}$$

## A Arquivos Do Relatório

Os arquivos dos modelos deste relatório, bem como o código em SQL para a geração das tabelas do banco de dados estão presentes no link abaixo. Vale notar que foi desenvolvido para o SGBD PostgreSQL.

- Repositório: <https://github.com/Luke0133/ProjetoBancoDados/tree/main/Relatorio>.

O código, também presente no github, foi desenvolvido em python (testado no Windows) e pode ser executado com `python main.py`. Para isso, deve-se configurar um banco de dados no PostgreSQL com o nome BDExtensao e senha "adminext".

- Repositório: <https://github.com/Luke0133/ProjetoBancoDados/tree/main/App>.