

Pré-Projeto

Nome do Projeto

Ruan Petrus - 21/1010459

¹Dep. Ciência da Computação – Universidade de Brasília (UnB)
Banco de Dados

211010459@aluno.unb.br

1. Introdução

Esse relatório foi criado com o intuito de aplicar os conhecimentos da matéria de banco de dados. Resolveremos um problema real, criando uma aplicação completa, no relatório será discutido apenas o banco de dados da aplicação.

O problema pode ser enunciado da seguinte maneira:

Para auxiliar os estudantes na escolha de disciplinas, você ficou encarregado de elaborar um sistema onde estudantes podem avaliar professores e disciplinas. O programa vai permitir que estudantes se cadastrem e postem avaliações de professores e turmas de diferentes semestres. Comentários ofensivos podem ser denunciados por usuários, que então são avaliados por administradores. Os administradores podem ignorar a denúncia, aceitá-la e remover o comentário ofensivo e até mesmo excluir uma conta de algum estudante. Os estudantes também podem editar e excluir seus próprios comentários.

Foi utilizado o *postgresql* como SGBD. O DER foi criado utilizando o site. E o MR foi criado utilizando a ferramenta ERD tools do PgADMIN4.

A seguir modelaremos o banco de dados e discutiremos o problema.

2. Diagrama Entidade Relacionamento

Podemos ver o diagrama entidade relacionamento na imagem 1. Para modelar os alunos, usuários e administradores, criaremos uma entidade com o atributo is admin chamada Usuário, dessa forma se o atributo for positivo ele será administrador, senão ele será apenas um estudante usuário.

Criaremos também as entidades **Departamento**, **Professor**, **Disciplina** e relacionaremos **Professor** e **Disciplina** com **Departamento**.

Criaremos a entidade **Turma** e relacionaremos ela tanto com **Professor** tanto com **Disciplina**.

Para as avaliações criaremos a entidade **Avaliacoes** que será uma generalização tanto de avaliações relacionadas a turma tanto com avaliações relacionadas a professor. Essa entidade terá ligação com o usuário que a escreveu, e suas especificações com **Turma** e **Professor** respectivamente.

E por último as denuncias seram uma entidade relacionada com a avaliação.

3. Modelo Relacional

Podemos ver o modelo relacional na imagem 2. Algumas informações interessantes sobre a modelagem é que eu decidi modelar a generalização da **Avaliação** em tabelas separadas

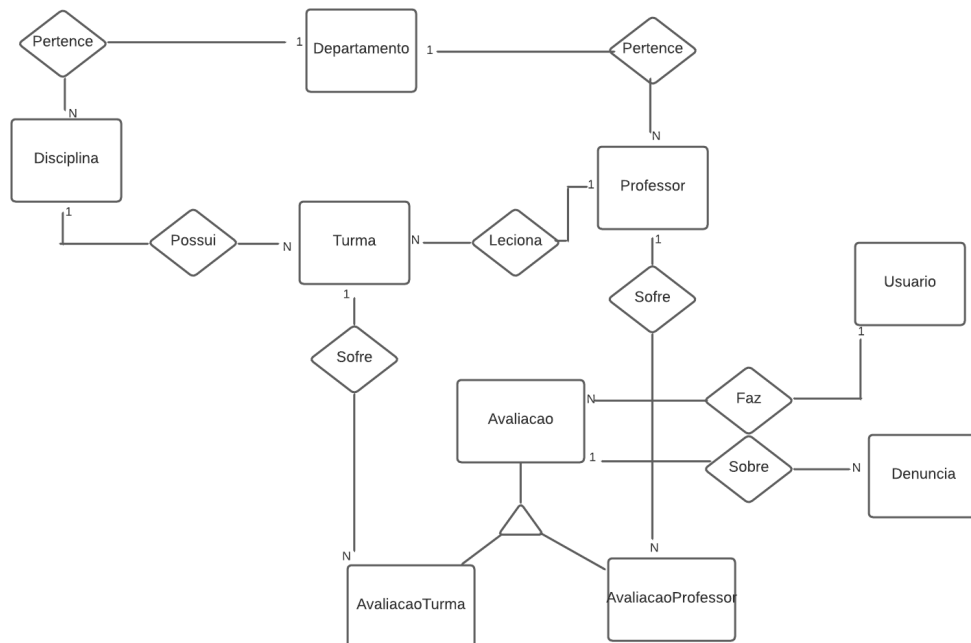


Figure 1. Diagrama Entidade Relacionamento

Avaliacoes e **Professores** **Avaliaco**es que correspondem as entidades do MER **Avaliacao-Turma** e **AvaliacaoProfessor** respectivamente. As denuncias também estão separadas em duas entidades.

E eu decidi adicionar atributos relacionado a notas na tabela de **Professor** que são modificados automaticamente quando é criado uma nova avaliação relacionada a aquele professor por meio de uma *PROCEDURE* e um *TRIGGER*.

Foi também

4. Formas normais

Faremos a avaliação das formas normais das tabelas **Turmas**, **Departamentos** e **Avaliaco**es. Podemos ver a tabela **Turmas** na imagem 3, a tabela **Departamentos** na imagem 4 e a tabela **Avaliaco**es na imagem 5. Como podemos ver não temos nenhum valor múltiplo dentro de uma célula na tabela e todas as linhas possuem uma chave primária única simples chamada *id*, satisfazendo assim a primeira forma normal. Em relação a segunda forma normal podemos ver que as chaves primárias não são compostas, dessa forma todos os atributos são funcionalmente dependentes somente de *id*, satisfazendo a segunda forma normal. E em relação a terceira forma normal podemos ver que os valores das tabelas são estritamente ligados somente a chave primária, de forma que eles não são dependentes do atributo não chave, satisfazendo assim a terceira forma normal.

5. Conclusão

Acredito que o projeto me ensinou muito sobre banco de dados principalmente porque as especificações foram mudando ao longo da implementação e o banco de dados teve que ser mudado ao mesmo tempo. Acredito que em um projeto real aconteça a mesma coisa.

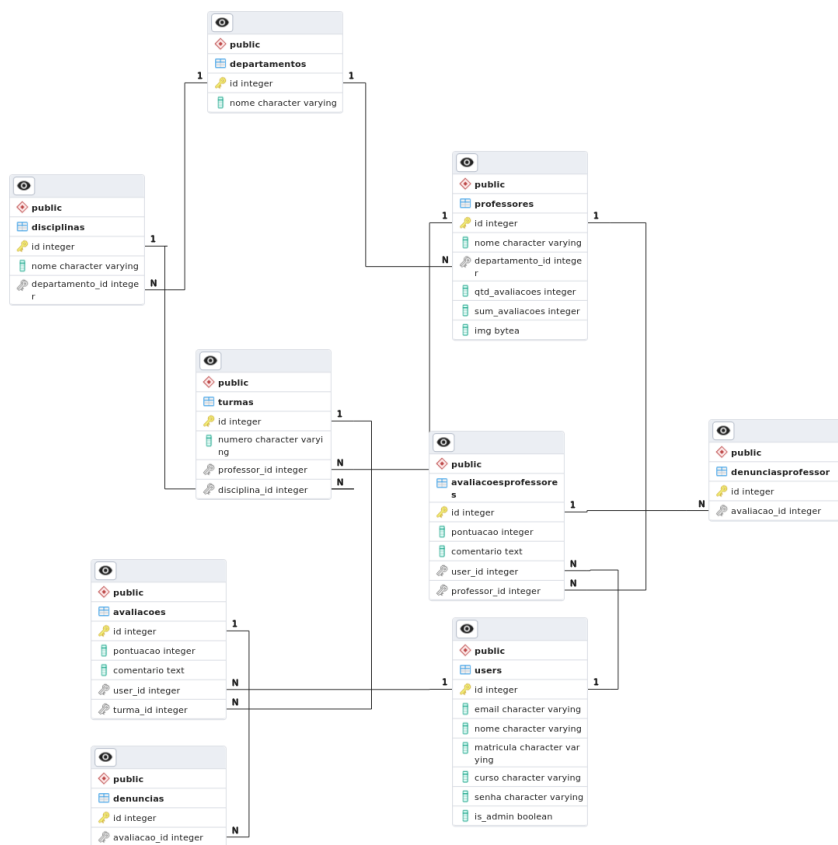


Figure 2. Modelo Relacional

Por causa das mudanças acredito que a arquitetura final dexou a desejar mas foi suficiente para a implementação da aplicação.

Link para a demonstração da implementação .

Link para o projeto no github.

	id [PK] integer	numero character varying	professor_id integer	disciplina_id integer
1	1	01	1	1
2	2	01	1	2
3	3	01	1	3
4	4	02	2	1
5	5	02	2	2
6	6	02	3	3
7	7	03	3	1

Figure 3. Tabela Turmas

	id [PK] integer	nome character varying
1	1	CIC
2	2	MAT
3	3	EST

Figure 4. Tabela Departamentos

	id [PK] integer	pontuacao integer	comentario text
1	1	5	Essa é com certeza a melhor matéria do curso, você aprende muita coisas sobre algoritmos que você jamais saberia caso não tivesse feito. Além disso o professor João é r
2	2	3	Essa matéria aí é de maluco, não vou usar nada disso na minha carreira profissional, e também não entendi nada que o professor João falou
3	3	5	Achei o professor Luan um fofo
4	4	3	Muito divertida a matéria
5	5	3	Matéria mais difícil do planeta terra
6	6	4	Eu achei haskell muito divertido
7	7	3	Achei que o professor falava muito lento e a prova muito difícil
8	8	4	Achei a materia OK
9	9	4	O professor explicava bem
10	10	4	Achei a materia interessante

Figure 5. Tabela Avaliacoes