

# Relatório Projeto Final

## Projeto: Banco de Dados para Gestão de uma Escola

Alexsander Correa de Oliveira – 19/0023562

João Vítor Siqueira de Araujo – 19/0031026

### Introdução

O projeto consiste na implementação de um banco de dados para a gestão de um sistema acadêmico de uma escola. A seguir será apresentado a especificação do modelo utilizado para a construção do Banco de Dados para essa escola:

*Um estudante é identificado pela sua matrícula. Deve-se saber o nome do estudante, sua data de nascimento, seu sexo e a mensalidade cobrada para esse estudante. Além disso, também é necessário que o estudante possua uma foto sua armazenada no registro do banco. Todo estudante possui um ou mais responsáveis e cada responsável responde por um ou mais estudantes. O responsável é identificado pelo seu CPF, e tem como dados interessantes para serem armazenados o seu nome, seu estado civil e os telefones para contato.*

*Temos também que um estudante pode realizar nenhuma ou várias atividades extracurriculares, que por sua vez uma atividade extracurricular pode ter nenhum ou vários estudantes fazendo-a (uma atividade recém-criada não precisa já ter estudantes cadastros a realizando). Cada atividade é determinada unicamente pelo seu código, e possui um horário em que acontece (exemplo: 16:00:00) e uma descrição dizendo do que se trata a atividade (aula de língua estrangeira, clube de esporte, banda da escola, clube de teatro, conselho estudantil, etc.). Toda atividade acontece em um local específico, mas temos locais em que não é realizada nenhuma atividade extracurricular.*

*Os locais são definidos por um código, além disso, deve-se ter ciência da categoria daquele local (se é uma sala, uma quadra, um laboratório, etc.), podendo ter uma descrição para caracterizar a categoria (por exemplo, para salas teria: “de aula”, “de reunião”, “de detenção”, etc.). Também é necessário que se tenha registrado os itens pertencentes àquele local, informando a quantidade e o nome do item, podendo conter também uma descrição adicional para dar detalhes sobre o item e uma ilustração (imagem) do mesmo.*

*Todo estudante é matriculado em uma única turma, que por sua vez uma turma pode ter nenhum ou vários estudantes matriculados (uma turma recém-criada não necessariamente precisa de estudantes matriculados). Cada turma é definida pelo seu código de identificação e é preciso saber também o seu turno do dia (matutino, vespertino). Cada turma tem aula em um único local, porém um local não necessariamente é usado por uma turma para ter aula, além disso, um mesmo local pode ser usado por várias turmas para terem aula. Toda turma tem aula com um ou mais professores e um professor pode dar aula para nenhuma ou várias turmas (caso*

*o professor seja substituto ou seja recém-contratado, não precisa obrigatoriamente dar aula a uma turma).*

*Um professor é identificado pela sua matrícula, sendo também descrito pelo seu nome, seu salário, a sua data de nascimento e de contratação e uma formação acadêmica. Além disso, assim como os estudantes, é preciso que uma foto do professor seja guardada dentro do banco de dados. Um professor pode ou não supervisionar algumas atividades extracurriculares, entretanto, toda atividade extracurricular deve ser supervisionada por um professor. Ademais, um professor pode ou não ter dependentes. Cada dependente precisa estar associado a um professor para ser identificado, sendo interessante que os dados acerca do nome do dependente, o parentesco com o professor e a data de nascimento sejam armazenadas no banco.*

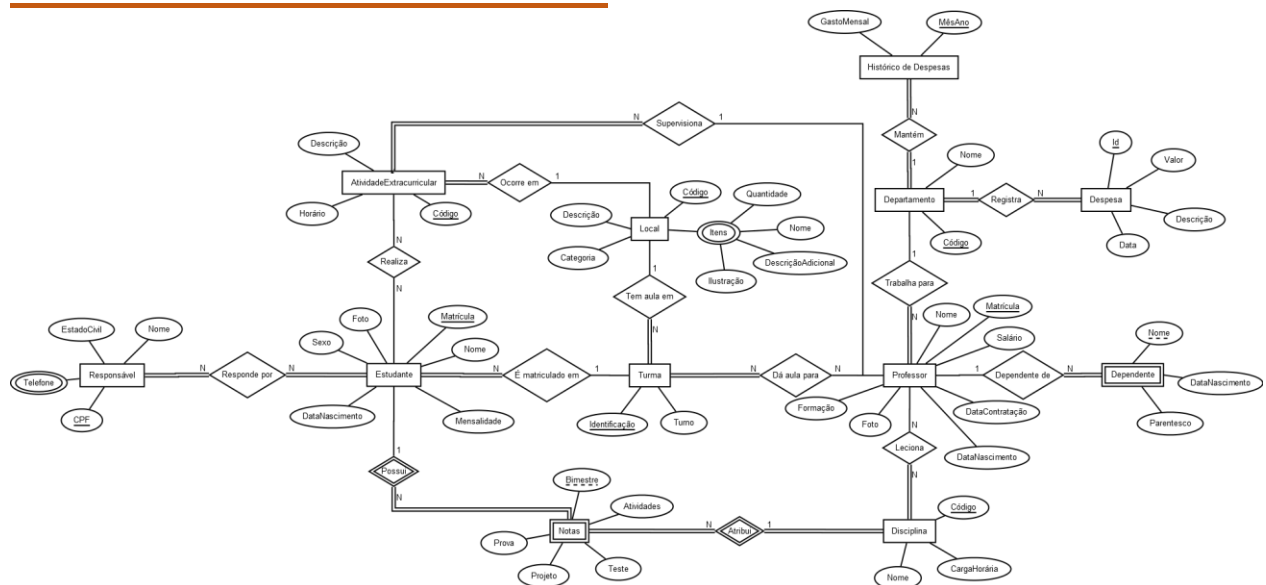
*Todo professor trabalha para um único departamento e um departamento pode ou não ter professores trabalhando para ele (no caso de um departamento que acabou de ser criado, ainda não há professores contratados para o mesmo). Cada departamento é descrito por um código e um nome e deve registrar as despesas mensais em uma tabela, onde devem ser salvos o identificador da despesa, o valor do gasto, a data em que ocorreu e a descrição da despesa (luz, água, manutenção, etc.). Cada despesa é associada a apenas um departamento. Além disso, cada departamento deve manter um ou mais históricos de despesas, porém um histórico só pode estar associado a um único departamento. O histórico é composto pelo mês/ano (por exemplo: 2021/08/00) e pelo gasto mensal total, sendo identificado pelo mês/ano.*

*Um professor pode ou não dar aula de uma ou mais disciplinas (mesmo caso do professor substituto e recém-contratado citado anteriormente) e toda disciplina é dada por um ou mais professores. A disciplina é determinada por um código e é formada pelo seu nome e sua carga horária. Cada disciplina atribui um conjunto de notas, e um conjunto de notas é identificado por uma única disciplina. O conjunto de notas contém o valor total das notas das provas, dos testes, dos projetos e das atividades do estudante naquela disciplina, assim como o bimestre em que essas notas foram registradas.*

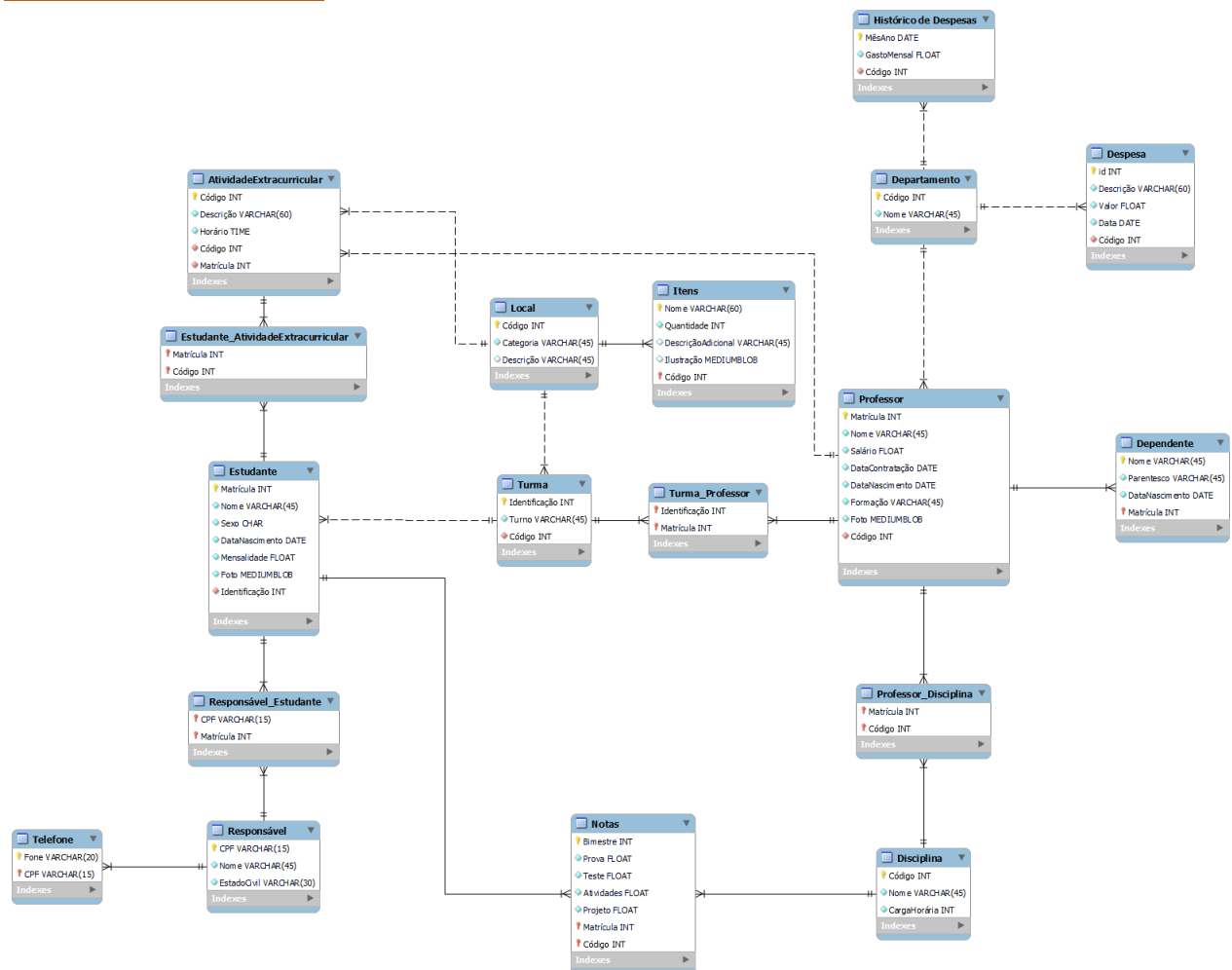
*Por fim, um estudante pode ou não possuir notas (um estudante do maternal, por exemplo, não tem notas). Porém, toda nota é associada a um estudante, pois as notas são identificadas unicamente pelo estudante que as possui e a disciplina que as atribuiu.*

Dado a especificação usada para implementar o Banco de Dados, a seguir serão mostrados o **Modelo Entidade Relacionamento** e o **Modelo Relacional** do Banco de Dados proposto. Posteriormente, serão feitas 5 consultas usando **Álgebra Relacional** nas tabelas do banco e também será feita uma análise das **Formas Normais** para 5 tabelas do nosso Banco de Dados. Por fim, será mostrado o **Diagrama da Camada de Mapeamento** juntamente com o link para o repositório do GitHub contendo todos os códigos usados para implementação do Banco de Dados.

## Modelo Entidade Relacionamento



## Modelo Relacional



(\*) Caso não esteja difícil de visualizar os modelos, no repositório do projeto estará o .png e .pdf dos modelos. Talvez seja melhor ver por lá.

## Álgebra Relacional

Para facilitar a visualização das consultas em Álgebra Relacional, forma utilizadas tabelas temporárias ( $T_1$ ,  $T_2$  e  $T_3$ ), como pode ser visto nas equações abaixo

1) A equação abaixo trata-se de um produto cartesiano entre as tabelas *Estudante*, *Turma* e *Local*. É feito em seguida uma operação de seleção, para selecionar apenas os registros em que as 3 tabelas se relacionam. A tabela resultante é formada pelas colunas da tabela *Estudante* seguidas pelas colunas da tabela *Turma*, por fim, as colunas da tabela *Local*.

$$T_1 \leftarrow (Estudante \times Turma \times Local)$$

$$\sigma_{Estudante.Identificacao=Turma.Identificacao \text{ and } Turma.Codigo=Local.Codigo}(T_1)$$

2) A próxima equação utiliza da operação de renomeação para facilitar as demais operações da equação. De resto, é feito um produto cartesiano entre as tabelas *Estudante*, *Turma*, *Turma\_Professor* e *Professor*, similar ao que foi feito na primeira consulta e, por fim, é projetado dessa tabela o nome do estudante, o turno em que ele tem aula e o nome de seu professor.

$$T_1 \leftarrow (\rho_E(Estudante) \times \rho_T(Turma) \times \rho_{TP}(Turma\_Professor) \times \rho_P(Professor))$$

$$T_2 \leftarrow \sigma_{E.Identificacao=T.Identificacao \text{ and } T.Identificacao=TP.Identificacao \text{ and } TP.Matricula=P.Matricula}(T_1) \\ \pi_{E.nome, T.turno, P.nome}(T_2)$$

3) A terceira consulta faz exatamente a mesma coisa que a primeira, porém ela é feita de uma outra forma. Ao invés de fazer o produto cartesiano entre *Estudante*, *Turma* e *Local* e em seguida fazer uma seleção para unir as *Foreign Keys* das tabelas, é utilizado apenas a operação de **Join**, que já realiza essas duas operações de uma vez só. No caso, renomeamos as tabelas *Estudante* e *Turma* para facilitar a operação, e aplicamos um *Join* nelas e, posteriormente, é feito outro *Join* dessa tabela resultante com a tabela *Local*. O resultado final é o mesmo apresentado da primeira consulta.

$$T_1 \leftarrow \rho_E(Estudante) \bowtie_{E.Identificacao=T.Identificacao} \rho_T(Turma)$$

$$T_1 \bowtie_{T.Codigo=Local.Codigo} Local$$

4) A consulta a seguir realiza uma junção natural das tabelas *Estudante*, *Notas* e *Disciplina*, renomeadas para E, N e D, respectivamente. Da tabela resultante, é projetado o nome do Estudante, o nome da Disciplina e as notas da Prova e do Projeto.

$$\pi_{E.nome, D.nome, N.Prova, N.Projeto}(\rho_E(Estudante) * \rho_N(Notas) * \rho_D(Disciplina))$$

5) A última consulta é um pouco mais extensa que as demais. É realizado o produto cartesiano entre as tabelas *Estudante*, *AtividadeExtracurricular* e *EstudanteAtividadeExtracurricular*, que sofreram operações de renomeação para facilitar a manipulação das mesmas. É feita uma seleção, para ligar as *Foreign Keys* das tabelas. Dessa tabela resultante, é feita outra operação de seleção na qual selecionamos apenas as estudantes que participam de um clube de esporte. Por fim, dessas estudantes, selecionamos aquelas que têm mensalidade superior a 1000,00 reais e projetamos seu nome e o valor da mensalidade.

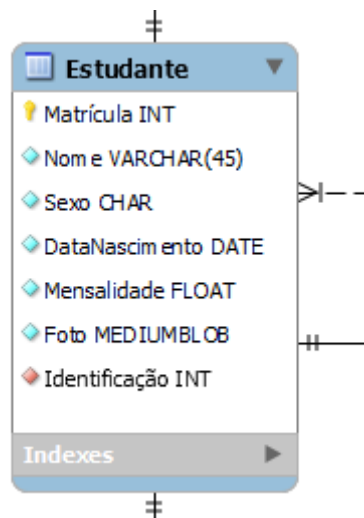
$$T_1 \leftarrow (\rho_E(\text{Estudante}) \times \rho_{AE}(\text{AtividadeExtracurricular}) \times \rho_{EAE}(\text{Estudante\_AtividadeExtracurricular}))$$

$$T_2 \leftarrow \sigma_{E.Matricula=EAE.Matricula \text{ and } EAE.Codigo=AE.Codigo}(T_1)$$

$$T_3 \leftarrow \sigma_{E.Sexo=F \text{ and } AE.Descricao="clube de esporte"}(T_2)$$

$$\pi_{E.nome, E.mensalidade}(\sigma_{E.Mensalidade > 1000}(T_3))$$

## Formas Normais



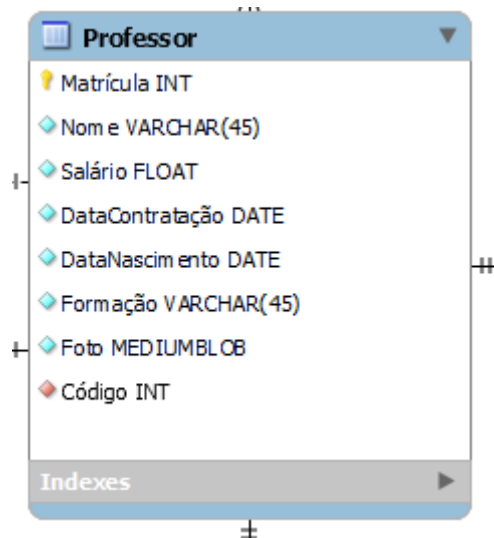
### 1) Tabela Estudante

**1FN:** A tabela se encontra na primeira forma normal, pois podemos perceber que todos os seus atributos são atômicos, afinal um estudante só pode ter uma matrícula, um nome, um sexo (masculino ou feminino, nunca os dois), uma data de nascimento e uma mensalidade. Para esse modelo, não faz sentido que o estudante tenha mais de uma foto salva no banco, pois ela só serve para descrição visual do mesmo. Por fim, o atributo identificação, referente à identificação da Turma que o estudante pertence, também é atômico, pois um estudante só pertence a uma única turma.

**2FN:** A tabela também se encontra na segunda forma normal, pois está na primeira forma normal e, além disso, todos os atributos do complemento da chave (nesse caso, a matrícula), que são o nome, o sexo, a data de nascimento, a mensalidade, a foto e a identificação da turma que o estudante pertence, são todos totalmente funcionalmente dependentes da matrícula do estudante.

**3FN:** Por fim, a tabela *Estudante* também se encontra na terceira forma normal, pois ela está na segunda forma normal e todos os seus atributos não-chaves são dependentes não-transitivos da chave primária (matrícula) pois:

- Estudantes com o mesmo nome não necessariamente nasceram na mesma data, nem possuem a mesma mensalidade, nem o mesmo sexo em alguns casos (por exemplo, o nome Alex pode ser tanto masculino como feminino). A foto provavelmente também será diferente, e não necessariamente farão parte da mesma turma.
- Estudantes do mesmo sexo podem ter nome totalmente diferentes um dos outros, terem nascido em datas diferentes, terem mensalidades diferentes, a foto no banco provavelmente é diferente, e não necessariamente fazem parte da mesma turma.
- Estudantes que nasceram na mesma data, também não necessariamente tem o mesmo nome, sexo nem mensalidade. A foto também não precisa ser a mesma, e podem fazer parte de turmas diferentes.
- Estudantes com mensalidades iguais não garante que esses terão o mesmo nome, sexo e foto armazenada nem que nasceram na mesma data ou que pertencem a mesma turma.
- Estudantes da mesma turma, obviamente não possuem obrigatoriamente o mesmo nome, sexo, mensalidade, foto nem nasceram no mesmo dia.
- Para o atributo foto temos uma situação interessante. Pode ocorrer que dois estudantes tenham a mesma foto, isso pode acontecer se por engano uma mesma foto tenha sido armazenada para estudantes gêmeos, ou ainda pode acontecer de dois estudantes terem tirado uma foto juntos e ambos submeteram essa foto para ser armazenada. Nesses casos, teríamos estudantes com fotos iguais, mas totalmente diferentes, o que mostra que uma mesma foto não necessariamente faz com que os estudantes tenham o mesmo nome, mensalidade ou sexo (no caso, a foto pode ser de um casal, por exemplo) nem que pertençam a mesma turma muito menos terem nascido no mesmo dia.



## 2) Tabela Professor

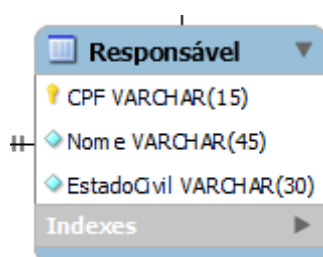
**1FN:** A tabela se encontra na primeira forma normal, pois todos os seus atributos são atômicos de modo que cada um pode ser expresso por um único valor. Um professor obviamente só tem uma matrícula, um salário, uma data de nascimento e de contratação. No nosso modelo, cada professor também só pode possuir uma formação acadêmica cadastrada e uma foto armazenada. Por fim, o atributo código, referente ao código do departamento, também é atômico, pois um professor só pode trabalhar para um único departamento.

**2FN:** A tabela também está na segunda forma normal, pois está na primeira forma normal e, além disso, todos os atributos do complemento da chave (nesse caso, a matrícula do professor), que são o nome, o salário, a data de nascimento, a data de contratação, a formação acadêmica, a foto e o código do departamento em que trabalha, são totalmente funcionalmente dependentes da chave (matrícula).

**3FN:** Por fim, a tabela *Professor* também se encontra na terceira forma normal, pois está na segunda forma normal e todos os seus atributos não-chave são dependentes não-transitivos da chave primária (matrícula) pois:

- Professores com o mesmo nome, podem ter salários diferentes, terem nascido e serem contratados em datas diferentes, podem também ter formações diferentes, assim como suas fotos no banco serem diferentes e não necessariamente precisam trabalhar no mesmo departamento.
- Professores com o mesmo salário também não precisam possuir o mesmo nome, foto e formação acadêmica, nem terem sido contratados ou terem nascido na mesma data, muito menos trabalharem para o mesmo departamento.
- Professores contratados na mesma data, não precisam ter o mesmo nome, data de nascimento, salário, formação, foto nem trabalharem para o mesmo departamento.
- Professores que nasceram no mesmo dia é análogo ao caso anterior.
- Professores com a mesma formação acadêmica podem ter nomes, salários, datas de nascimento e contratação e fotos diferentes, e também podem trabalhar para departamentos diferentes.

- Professores podem ter a mesma foto armazenada no banco de dados, como visto anteriormente na análise do estudante, para professores, o mesmo pode acontecer. Entretanto, terem a mesma foto não quer dizer que os nomes, os salários, as datas de nascimento e contratação e a formação sejam iguais para eles, muito menos significa que esses professores trabalham para o mesmo departamento.
- Por fim, professores que trabalham no mesmo departamento obviamente não precisam ter o mesmo nome, salário, data de nascimento, data de contratação, formação ou foto.



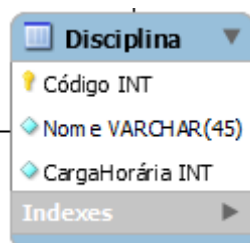
### 3) Tabela Responsável

**1FN:** A tabela se encontra na primeira forma normal, pois todos os seus atributos são atômicos de forma que cada um pode ser expresso por um único valor. No caso, um responsável tem obviamente um único CPF, um único nome e também um único estado civil.

**2FN:** A tabela também está na segunda forma normal, pois está na primeira forma normal e, além disso, todos os atributos do complemento da chave (nesse caso, o CPF), que são o nome e o estado civil, são totalmente funcionalmente dependentes do CPF.

**3FN:** Por fim, a tabela *Responsável* também se encontra na terceira forma normal, pois está na segunda forma normal e todos os seus atributos não-chave são dependentes não-transitivos da chave primária (CPF) pois:

- Responsáveis com o mesmo nome, podem ter estados civis diferentes (um pode ser casado, enquanto o outro é solteiro, por exemplo).
- Responsáveis com o mesmo estado civil, obviamente não precisam ter obrigatoriamente o mesmo nome.



### 4) Tabela Disciplina

**1FN:** A tabela se encontra na primeira forma normal, pois todos os seus atributos são atômicos de modo que cada um pode ser expresso por um único valor.

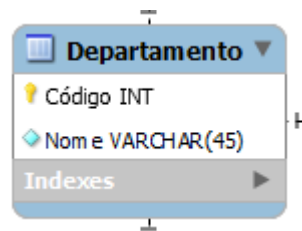


No caso, uma disciplina possui um único código que a identifica, um único nome e uma única carga horária.

**2FN:** A tabela também está na segunda forma normal, pois está na primeira forma normal e, além disso, todos os atributos do complemento da chave (nesse caso, o Código), que são o nome e a carga horária, são totalmente funcionalmente dependentes do Código da disciplina.

**3FN:** Por fim, a tabela *Disciplina* também se encontra na terceira forma normal, pois está na segunda forma normal e todos os seus atributos não-chave são dependentes não-transitivos da chave primária (código) pois:

- Disciplinas com o mesmo nome (por exemplo, a Matemática do Ensino Fundamental e a Matemática do Ensino Médio) podem possuir carga horária diferente.
- Disciplinas com a mesma carga horária não necessariamente possuem o mesmo nome.



## 5) Tabela Departamento

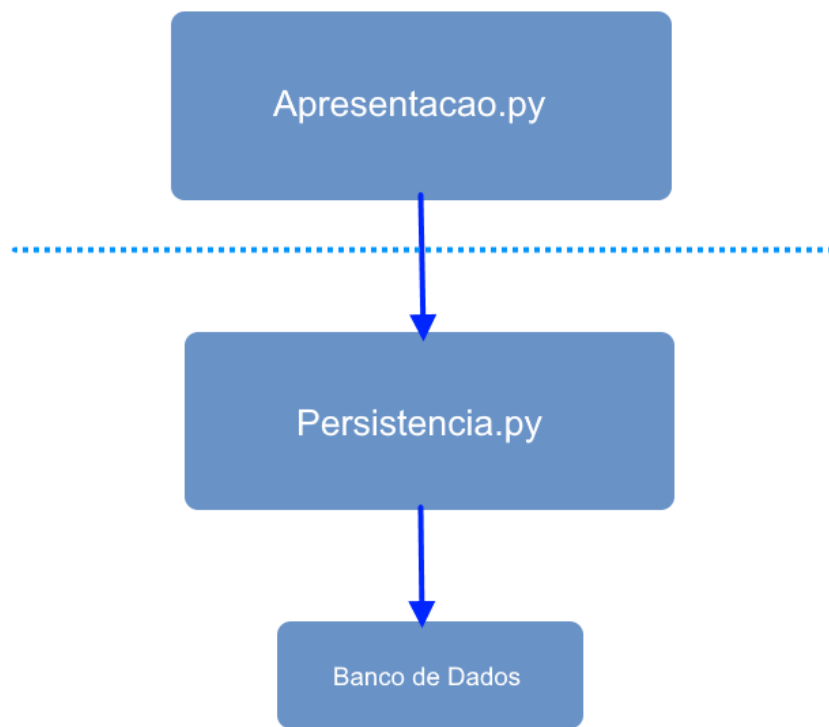
**1FN:** A tabela se encontra na primeira forma normal, pois todos os seus atributos são atômicos de forma que cada um pode ser expresso por um único valor. No caso, um departamento possui um único código que o identifica e um único nome.

**2FN:** A tabela também está na segunda forma normal, pois está na primeira forma normal e, além disso, o único atributo do complemento da chave (nesse caso, o Código), que é o nome, é totalmente funcionalmente dependente do Código do departamento.

**3FN:** Por fim, a tabela *Departamento* também se encontra na terceira forma normal, pois está na segunda forma normal e como existe um único atributo não-chave nessa tabela, ele é necessariamente dependente não-transitivo da chave primária (código).

## Diagrama da Camada de Mapeamento

A imagem a seguir demonstra como é feita a comunicação do nosso projeto com o Banco de Dados. Todas as informações são captadas pelo programa *Apresentacao.py* e encaminhadas para o programa *Persistencia.py*. Este por sua vez realiza a conexão com o Banco de Dados que efetivamente processa os dados.



**Link para o GitHub:** <https://github.com/KvotheKS/EscolaBD>