

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
DE SÃO PAULO**

Campus São João da Boa Vista

Trabalho Final de Curso

4º ano – Curso Técnico em Informática

Prof. Breno Lisi Romano

**INDRA: DESENVOLVIMENTO DOS MODELOS DE BANCO DE  
DADOS (CONCEITUAL, LÓGICO E FÍSICO) DO MÓDULO DE  
RELATÓRIOS.**

Aluno: João Pedro Leocardio Jacomini Nascimento

Prontuário: 1320106

São João da Boa Vista – SP

2016

## **Resumo**

Texto do resumo...

## Sumário

1	Introdução .....	4
2	Desenvolvimento .....	8
2.1	Referencial Teórico. ....	8
2.1.1	Banco de Dados .....	8
2.1.2	Modelagem de Banco de Dados .....	11
2.2	Metodologia do Trabalho .....	23
2.2.1	Objetivos do Projeto: Indra.....	24
2.2.2	Indra: Necessidades de Banco de Dados .....	25
2.2.3	Idealizando o Banco de Dados .....	27
2.2.4	Aplicando o Banco de Dados .....	35
2.2.5	Integrando o Banco de Dados.....	35
3	Conclusões e Recomendações .....	36
4	Referências Bibliográficas .....	37

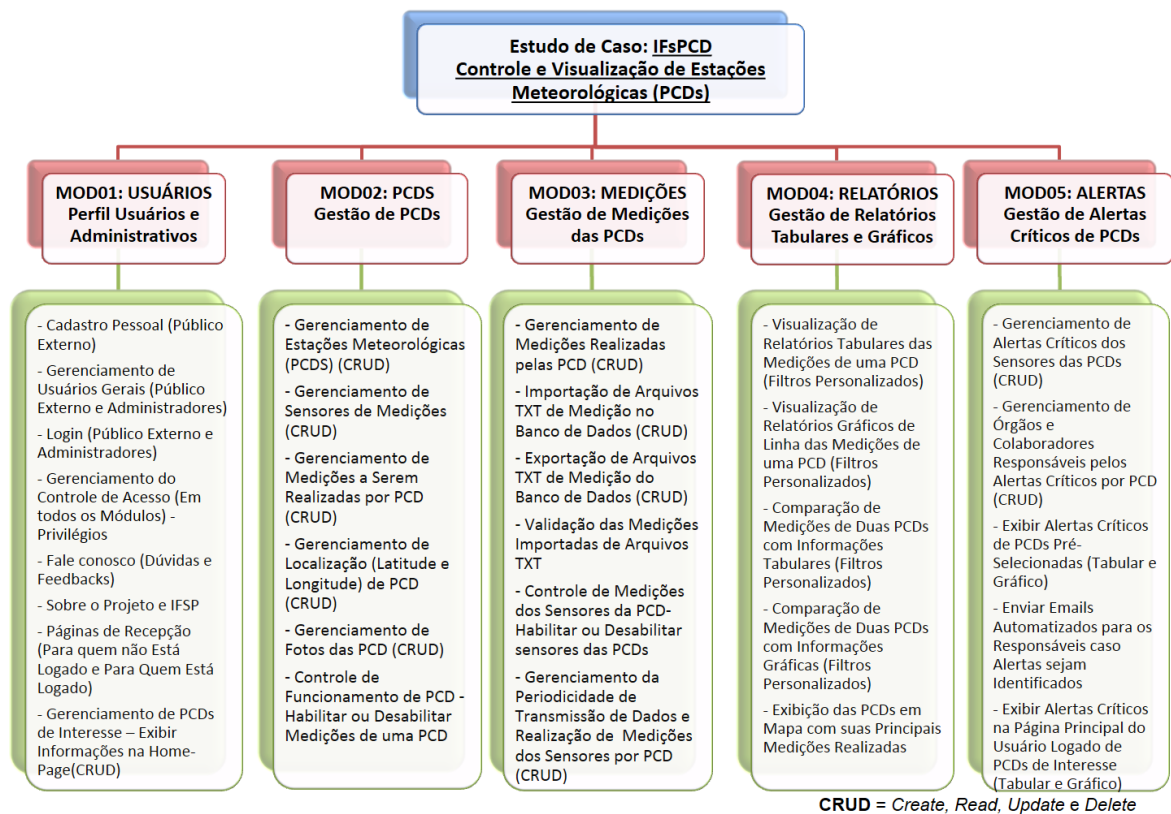
# 1 Introdução

Inicialmente, sob a equipe do Quarto Ano (4º Ano) de 2016 do curso Técnico Integrado de Informática, foi proposto o desenvolvimento de um projeto que agregaria todas as técnicas adquiridas através das disciplinas de Informática ao longo desses três anos de curso. Este projeto seria administrado através das aulas de Prática de Desenvolvimento de Sistemas (PDS), pelo Professor Breno Lisi Romano, o mesmo que definiu o tema do projeto.

O tema, primeiramente, deveria ser baseado nas especificações de uma Empresa existente, no entanto, a que viria a participar com a equipe foi declarada como desistente. Portanto, foi necessário que o Professor administrador da disciplina de PDS forjasse um tema particular. O assunto, então, foi baseado em receptar as mudanças climáticas presentes na região de São João da Boa Vista influenciado pelas constantes alterações súbitas do clima que ocorre habitualmente, e também devido aos alagamentos ocasionais que se originam nas regiões de córregos do município. O monitoramento do clima foi planejado para ser aplicado através de Plataforma de Coleta de Dados (PCD), constituídas por complexos dispositivos eletrônicos que agregaram informações do ambiente, que posteriormente seriam enviados para o sistema desenvolvido através deste projeto.

O Professor também definiu os setores pelos quais cada aluno deveria trabalhar, se diferenciando entre os aspectos que constituem o projeto, chamados de Módulos. Cada Módulo representa não apenas funcionalidades específicas, mas também ambientes diferenciados. Estes ambientes foram construídos através de páginas WEB, onde a população terá o acesso às informações. Portanto, foram definidos cinco Módulos diferentes, que se segue na **Figura 1**:

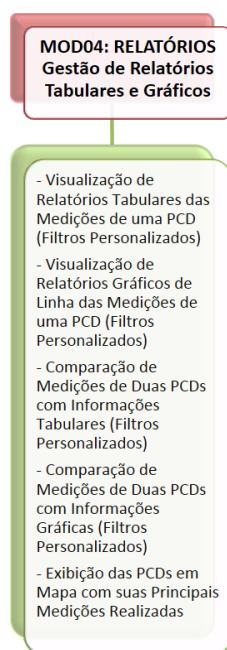
- Módulo 01: Usuários;
- Módulo 02: PCDS;
- Módulo 03: Medições;
- Módulo 04: Relatórios;
- Módulo 05: Alertas.



**Figura 1 Módulos do Projeto**

Como se trata de uma disciplina da grade obrigatório do curso, todos os alunos tiveram de participar do desenvolvimento. No entanto, o primeiro objetivo da equipe foi à divisão de seus pares dentre os Módulos disponíveis. Cada Módulo deveria ser composto por integrantes que exerceriam funções específicas: Desenvolvedor, Analista e Desenvolvedor de Banco de Dados. Os conjuntos, então, foram equilibrados dentre o número de seus integrantes e a quantidade da função que cada membro exerceria dentro do Módulo.

Este relatório, no entanto, procura se centralizar apenas nas tarefas de um único Módulo: o Módulo de Relatórios, que segue na **Figura 2**. Mais especificamente, na experiência do desenvolvimento dos Modelos de Banco de Dados específicos do conjunto. Estes modelos agregam desde ao conceito de Banco de Dados até sua execução física (virtual).



**Figura 2 Módulo 04: Relatórios**

É necessário enfatizar que o uso de um sistema de Bancos de Dados é extremamente necessário para este projeto, assim como qualquer outro em um mercado ou empresa, que se utiliza de sua base de dados virtuais para ter sempre disponível informações que auxiliem suas equipes gestoras e estratégicas a calcular riscos, otimizar processos e melhorar os resultados [1]. O projeto necessita de uma forma dinâmica para armazenar as informações adquiridas relacionadas ao clima e ao ambiente externo, pois através desses dados que é possível realizar processos e resultados para atender as funções que o sistema promete.

Dentro das funcionalidades do Módulo 04 Relatórios, estarão sendo tratadas informações referentes principalmente às características das PCDs dispostas na região e os Dados coletados por elas, pois este módulo possui o intuito de criar apresentações instrutivas, através das páginas WEB, de modo a divulgar as informações concebidas para a população, trazendo um novo meio para esta de se tornar ciente das variações climáticas da sua região e das possíveis catástrofes.

Então, se torna importante entender como a construção de Banco de Dados afeta no desempenho das funcionalidades de um sistema, desde seus protótipos até seu contínuo ciclo de manutenções. Os resultados dependem, basicamente, da manutenção e da atualização dos dados, portanto, a utilidade e os benefícios adquiridos são diretamente proporcionais à forma de armazenar, organizar e acessar esses dados [1], perante a estes motivos que os Modelos de Banco de Dados (**Figura 3**) devem ser trabalhados com atenção, pois através destes que se conseguirá adquirir um Banco de Dados funcional que atenda às expectativa do projeto.

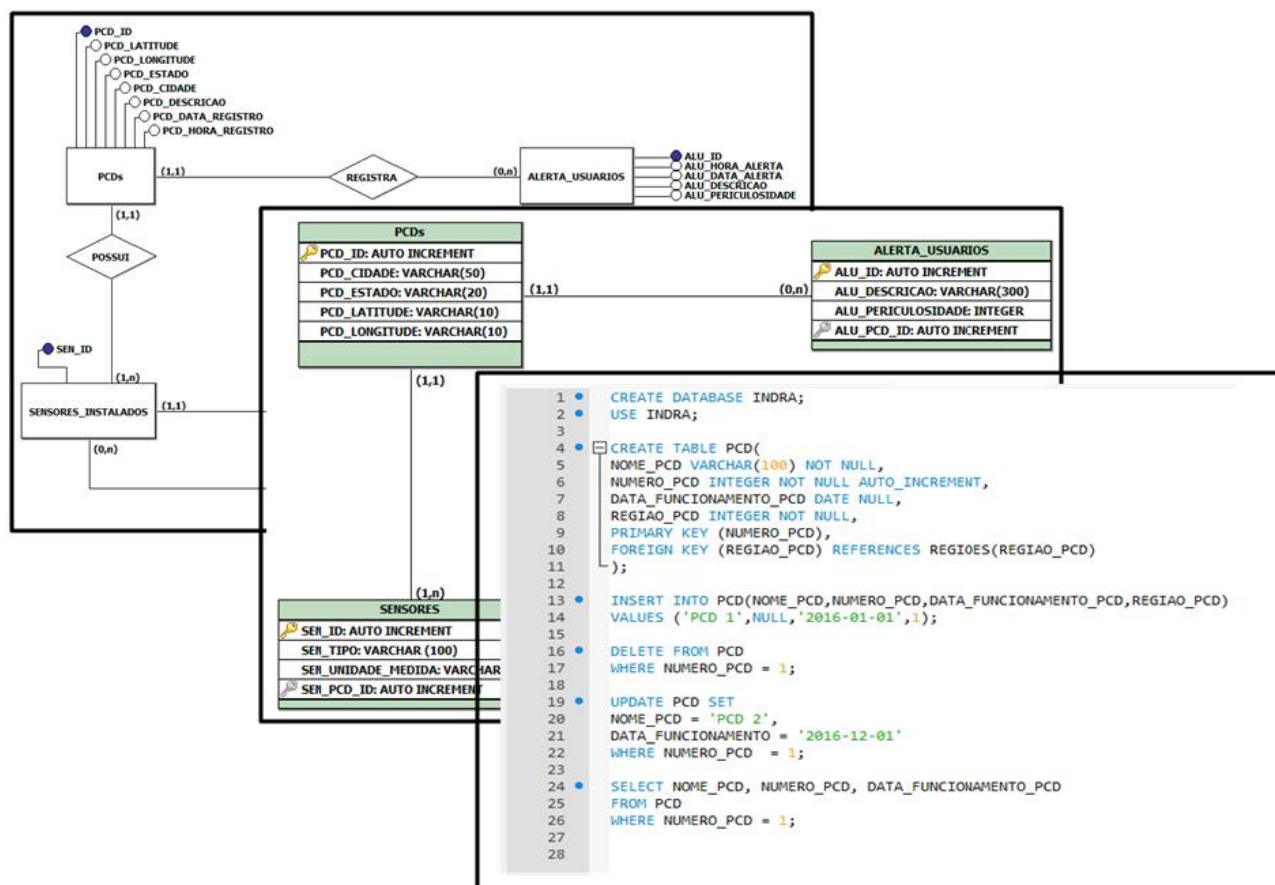


Figura 3 Modelos de Banco de Dados

Portanto, este trabalho possui o intuito de apresentar o propósito de se desenvolver Modelos de Banco de Dados, enfatizando quais impactos eles causam perante a construção do Banco. Cada Modelo possui uma estrutura e parâmetros diferenciados, que gradualmente se tornam mais complexos com a evolução deste. Para garantir um Banco funcional e com menores chances de falhas, é necessário entender as estruturas destes Modelos, que facilitam para o Administrador de Banco de Dados a entender o projeto da distribuição dos dados desde o princípio, identificando seus cenários, entidades e seus respectivos atributos, e, enfim, é sobre esse processo que este trabalho está se baseando.

## 2 Desenvolvimento

### 2.1 Referencial Teórico.

Para se desenvolver um novo Banco de Dados ou atualizar os que já se encontram muito ultrapassados, é necessário propor um projeto. Este projeto se utiliza da organização de informações e do uso de técnicas para que na finalização do banco, ele obtenha sua maior performance. O projeto também consiste em várias documentações, que podem auxiliar posteriormente na manutenção do Banco.

Para se desenvolver um projeto de Banco de Dados é recomendado seguir algumas instruções em relação á técnica dos Modelos de Banco de Dados, além de conhecer os conceitos utilizados nesta técnica.

#### 2.1.1 Banco de Dados

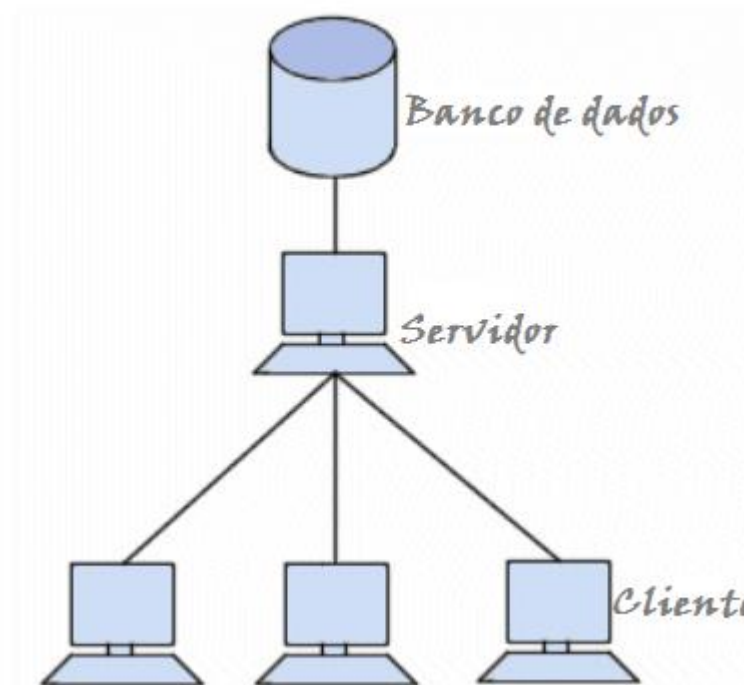
Os Bancos de Dados estão presentes em nosso cotidiano á muito tempo, através dos meios pelo qual a humanidade armazenou suas informações de forma estruturada, como a lista telefônica. No passado, empresas e pessoas guardavam seus dados através de formas materiais, isto é, eles os registravam em arquivos físicos, que ao se acumular, acabavam ocupando muito espaço dentro do ambiente de trabalho. O uso dos meios físicos, como os gabinetes, gavetas e pastas limitavam o armazenamento das informações, dificultava a organização dos dados e sua pesquisa, ocupavam grande espaço e também poderiam sofrer avarias devido à influência do ambiente externo, como umidade, incêndios, deterioração do material, etc.

Com o surgimento e evolução dos computadores, possibilitou as empresas a armazenar suas informações de forma digital. O ambiente virtual não ocupa espaço no meio físico, apenas em relação aos equipamentos que são usados para seu funcionamento, mas sua capacidade é muito superior ao registro material. Seus dados podem ser replicados a fim de manter sua integridade, impedindo que informações importantes se percam. Eles não sofrem avarias do meio externo e podem dispor de sistemas de segurança para mantê-los livre de qualquer ameaça virtual. E, com a chegada da Internet, foi possível conectá-los á rede e torná-los disponíveis a qualquer distância e a qualquer momento, transformando a administração de dados muito mais prática e dinâmica.

Um Banco de Dados são conjuntos de informações que se relacionam formando um sentido [2], ou seja, todo ou qualquer agrupamento de informações que se relacionem pelo mesmo assunto, podendo ser físico ou virtual. É uma entidade na qual há a possibilidade de se armazenar dados de maneira estruturada e com menor redundância. Estes dados podem poder ser utilizadas por



programas, por aplicações ou por diferentes usuários [3]. Portanto, ele está freqüentemente conectado á rede de Internet, possibilitando poder ser acessado por vários usuários, simultaneamente [3]. Isso é demonstrado na **Figura 4**.



**Figura 4** Comunicação de uma Banco de Dados [3]

O Banco de Dados dispõe informações aos usuários para que os mesmos possam efetuar atualização, inserções e consultas, de acordo com os privilégios que são atribuídos a ele. O acesso a essas informações pode ser através de uma máquina física usada pelo o usuário, ou pode ser acessada através da rede caso o armazenamento for compartilhado através de outras máquinas ou servidores.

#### 2.1.1.1 Dados e Informações

É comum assemelhar as informações como sendo dados, onde ambas são sinônimos, mas na verdade elas não possuem o mesmo significado.

Os Dados são fragmentos de informações, em sua forma mais simples. Muitas vezes, os dados não fazem sentido sozinhos [3]. A Informação consiste em um conjunto de dados organizados para fazer sentido, gerar conhecimento [3].

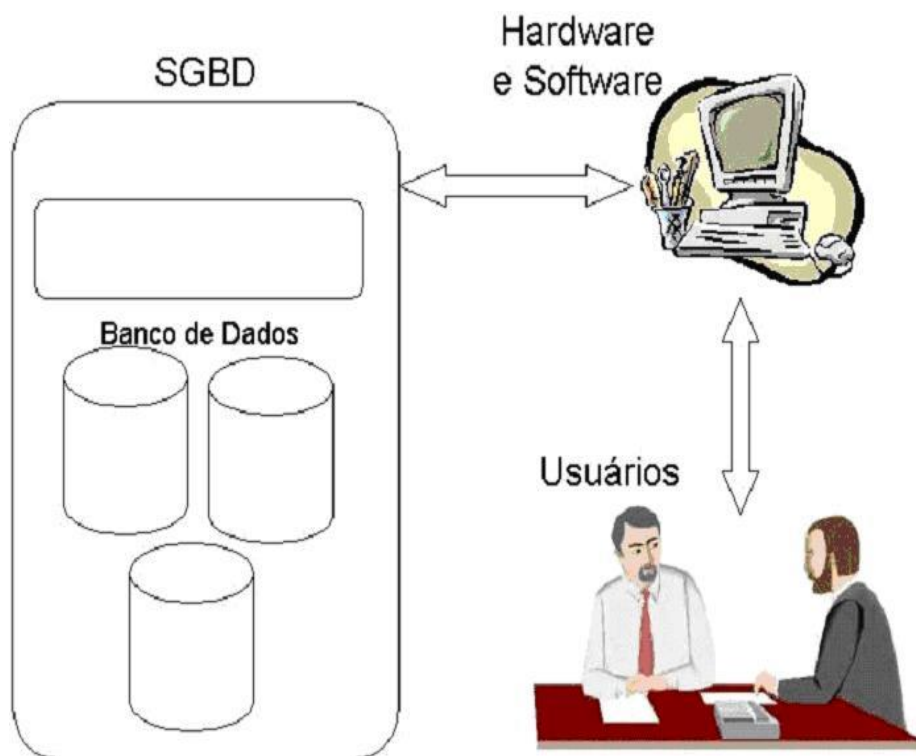
Portanto, o Banco é um conjunto de dados estruturados e organizados capaz de transmitir informação.

### 2.1.1.2 Sistema de Gerenciamento de Banco de Dados (SGBD)

Devido ao acúmulo de informações no Banco, torna-se necessário controlar os dados e o acesso dos usuários, portanto a aplicação de um software para a sua gestão é indispensável. O SGBD é um software que possui recursos capazes de manipular as informações armazenadas no banco de dados e interagir com os usuários [3]. Este programa permitir o acesso aos dados de maneira simples, autorizar um acesso às informações simultaneamente, manipular os dados presentes no banco de dados (adicionar, apagar, modificar) [4]. Exemplos de SGBDs são: Oracle, SQL Server, DB2, PostgreSQL, MySQL, etc.

### 2.1.1.3 Sistema de Banco de Dados

Um Sistema de Banco de Dados é formado por quatro componentes básicos: dados, hardware, software e usuários [3], como apresentado na **Figura 5**. Ele possui o propósito de isolar o usuário dos detalhes internos do banco de dados, isto é, promover a abstração de dados, e promover a independência dos dados em relação às aplicações, ou seja, tornar a forma de acesso e a forma de armazenamento independente da aplicação utilizada [3].



**Figura 5** Sistema de Banco de Dados [3]

#### 2.1.1.4 Abstração de Dados

O Sistema de Banco de Dados deve proporcionar um ambiente prático ao usuário, isto é, tornar acessível a ele somente às informações que lhe são pertinentes e úteis, abstraindo as informações que poderão prejudicar sua experiência e confundi-lo. Por isso é importante definir quais informações estarão acessíveis e quais estarão somente ao administrador, a fim de proteger a integridade do Banco de Dados e facilitar o acesso as informações pelo usuário.

#### 2.1.2 Modelagem de Banco de Dados

É chamada de Modelagem de Banco de Dados técnica usada para a especificação das regras de negócios e as estruturas de dados de um banco de dados. Ela participa do ciclo de desenvolvimento de um sistema de informação e é extremamente importante para o sucesso do projeto. Modelar dados consiste em esquematizar, através de desenhos, o sistema de informações do Banco de Dados, concentrando-se nas entidades lógicas e nas dependências lógicas entre essas entidades [7].

Envolve uma série de aplicações teóricas e práticas, visando construir um modelo de dados consistente, não redundante e aplicável em qualquer SGBD moderno [7].

A Modelagem envolve três divisões chamadas de Modelos de Banco de Dados. Os Modelos de Banco de Dados se referem a um Banco de Dados ainda não estruturado, e suas etapas se descrevem em diferentes níveis de abstração de informações e diferentes objetivos. Cada descrição possui um nome referente ao seu principal propósito.

##### 2.1.2.1 Modelo Conceitual

Representa as regras de negócio sem limitações tecnológicas ou conhecimento técnico, assim podendo ser trabalhado por qualquer tipo de usuário. [5] Ele é independente da forma de implementação do Banco de Dados, utilizado principalmente para se obter uma visão generalizada de todo o projeto de BD. Por representar uma visão simplificada esse modelo registra quais os tipos de dados que deverão ser armazenados, mas não registra como eles serão armazenados.

Devido ao fato não precisar de um conhecimento técnico para seu entendimento, pode ser utilizado com facilidade para a reflexão entre usuários e desenvolvedores de como o BD irá se comportar. E por se tratar de um Modelo simplificado, ele possui apenas as Entidades e os Atributos principais do BD, facilitando seu entendimento. [5]

##### 2.1.2.1.1 Entidades, Atributos e Relacionamentos

São os componentes principais dos Modelos de Banco de Dados, e são utilizados para definir os envolvidos no sistema e suas características.

Entidades podem representar tudo do mundo real, abstrato ou concreto, da qual se deseja armazenar informações [5], ou seja, é um objeto ou evento do mundo real sobre o qual desejamos manter um registro.

Ex.: Aluno, Carro, Produto, Vendedor, etc.

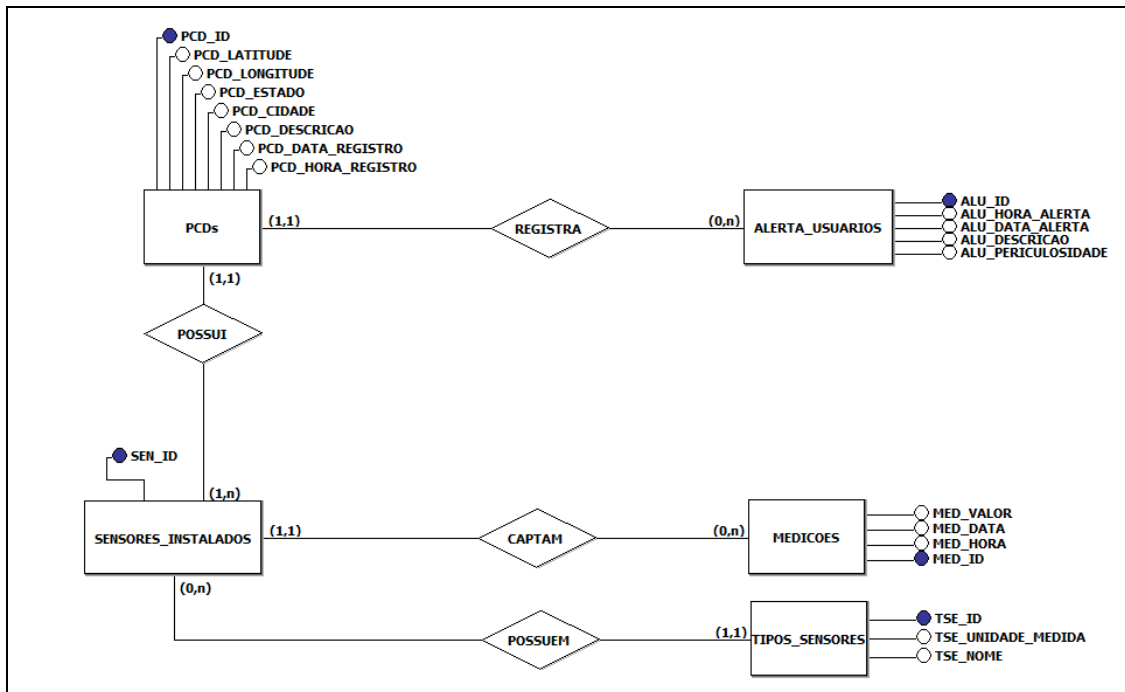
Atributos derivam das propriedades das entidades [6], isto é, é uma propriedade ou característica que descreve uma entidade. Também é chamado de campo.

Ex.: Atributos da entidade ALUNO: nome, data de nascimento, telefone, endereço, etc.

Relacionamentos são usados pós a definição das Entidades, a fim de indicar como elas irão se relacionar. Os relacionamentos em geral são nomeados com verbos ou expressões que representam a forma como as entidades interagem, ou a ação que uma exerce sobre a outra. O sentido dessa nomenclatura pode variar de acordo com a direção em que se lê o relacionamento [6].

Ex:   Pai– possui– Filho  
      Cliente – realiza – Pedido  
      Vendedor – vende – Produto

A técnica mais difundida de Modelagem de Banco de Dados é a abordagem Entidade Relacionamento (ER), a mesma pelo qual este projeto desenvolve seu Banco de Dados. Neste Modelo, em específico, essa abordagem se chama Modelo Entidade Relacionamento (MER), apresentada na **Figura 6**, é utilizada para descrever os objetos (entidades) envolvidos em um domínio de negócios, com suas características (atributos) e como elas se relacionam entre si (relacionamentos). Em geral, este modelo representa de forma simples e resumida a estrutura que o banco de dados do projeto irá possuir. O banco de dados poderá conter várias outros componentes também, tais como chaves e tabelas intermediárias [6].



**Figura 6 Modelo Entidade Relacionamento (MER)**

#### 2.1.2.2 Modelo Lógico

Representa uma descrição dos componentes e estruturas que serão armazenados no Banco de Dados, resultando em uma representação gráfica dos dados de maneira lógica. Este modelo considera os limites impostos para a tecnologia de Banco de Dados utilizada [6]. Sua exemplificação está na **Figura 7**.

Ele deriva do Modelo Conceitual, trazendo suas características de negócios e implementando novos tipos de relacionamentos, ou seja, se torna mais centralizado e detalhado. Nesta etapa, podem ser definidas as Chaves Primárias e Chaves Estrangeiras de suas Entidades e a padronização da nomenclatura de cada Entidade e Atributo. [5]

Este Modelo de Banco de Dados deve ser desenvolvido considerando as informações da modelagem de dados criada no Modelo Conceitual [7] e não deve descrever as características do armazenamento interno, apenas apresentando as relações e atributos de forma mais detalhada.

##### 2.1.2.2.1 Chaves Primárias e Chaves Estrangeiras

São atributos inseridos nas Entidades do Modelo Lógico que possuem um valor único e são utilizados principalmente para referenciar outras Entidades e permitir diferenciar suas instâncias.

Ex.: Matrícula, CPF, código, Renavam, Chassi, etc.

Chaves Primárias são obrigatórias em cada Entidade e possui a função de identificar exclusivamente cada ocorrência de uma entidade, diferenciando-as entre si [5].

Chaves Estrangeiras derivam das Chaves Primárias de outras Entidades e é descrita como a chave primária de uma entidade que aparece em outra entidade por consequência de seus relacionamentos [5].

Para este Modelo, é utilizada a técnica do Diagrama Entidade Relacionamento, que também é aplicada para este projeto. Enquanto o MER é um modelo conceitual, o Diagrama Entidade Relacionamento (Diagrama ER ou ainda DER) é a sua representação gráfica. O diagrama facilita ainda a comunicação entre os integrantes da equipe do projeto, pois oferece uma linguagem comum conhecida e utilizada tanto pelo analista, responsável por levantar os requisitos, quanto os desenvolvedores, responsáveis por aplicar aquilo que foi modelado [6].

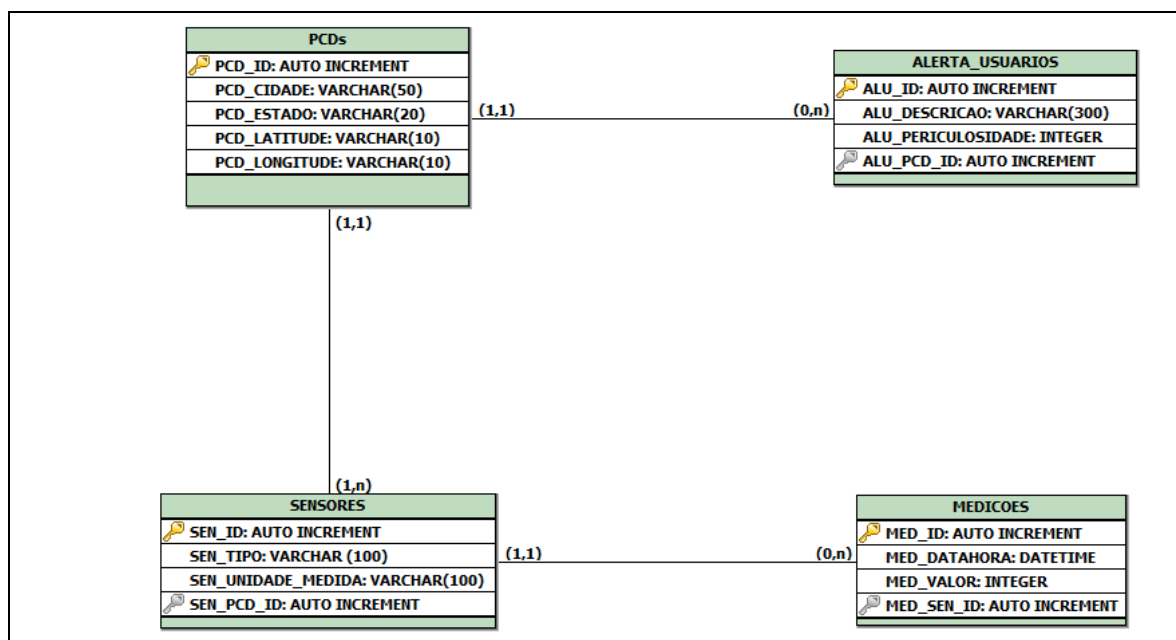


Figura 7 Diagrama Entidade Relacionamento (DER)

### 2.1.2.3 Modelo Físico

Este último Modelo é uma descrição da estrutura do Banco de Dados visto através do SGBD, manipulado por um usuário com a técnica em BD e na linguagem escolhida para seu desenvolvimento. Nesta etapa são detalhados os componentes da estrutura física do banco, como tabelas, campos, tipos de valores, índices, etc. Usando o SQGB escolhido para o projeto, é possível então criar o Banco de Dados propriamente dito. Portanto, ele considera as limitações do Sistema de Gerenciamento de Banco de Dados. [5] Na **Figura 7**, este modelo está sendo demonstrado através da utilização do serviço MySQL Workbench, uma das aplicações utilizadas para as construções de Bancos Físicos.

Ele é elaborado perante o Modelo Lógico e, portanto, traz consigo suas características informações. Suas modificações se baseiam na sintaxe de notação do SGBD, que pode proporcionar tabelas físicas e colunas físicas. [5]

#### 2.1.2.3.1 Criar Banco de Dados

No Modelo Físico, para se criar uma Base de Dados, é necessário adotar os comandos utilizados pelo SGBD escolhido. Neste caso, utilizando-se dos serviços MySQL, existem comandos específicos que efetuam cada procedimento da criação de uma Banco de Dados.

##### 2.1.2.3.1.1 CREATE DATABASE

O MySQL utiliza-se de diretórios para gravar as tabelas de suas Bases de Dados. Para se criar um Banco de Dados que não contenha nenhuma tabela, é utilizado o seguinte comando [9]:

```
CREATE DATABASE [NOME_DO_BANCO];
```

Esse comando criar um novo diretório sobre o diretório do MySQL. O nome do Banco de Dados pode conter até 64 caracteres, além de qualquer tipo de caractere, exceto Barra (/) e Ponto (.). Caso seja necessário excluir o BD, utilizasse o seguinte comando:

```
DROP DATABASE (IF EXIST) [NOME_DO_BANCO];
```

O parâmetro IF EXIST permite evitar erros na instrução MySQL caso o nome do Banco de Dados selecionado não exista. Para começar a utilizar o Banco de Dados, é necessário indicar qual estará em uso, portanto é preciso aplicar o seguinte comando:

```
USE [NOME_DO_BANCO];
```

##### 2.1.2.3.1.2 CREATE TABLE

Assim como nos Modelos anteriores, no Modelo Físico é necessário criar Tabelas, que aplicam o mesmo efeito das Entidades, porém estas irão conter todas as informações do Banco de Dados, segregando-as perante entre seus atributos. Para criar uma nova tabela no BD selecionado, utilizasse o comando [9], igual segue a **Figura 8**:

```
CREATE TABLE [NOME_DA_TABELA]( );
```

```
1· CREATE DATABASE INDRA;  
2· USE INDRA;
```

**Figura 8 Comando CREATE DATABASE**

Os nomes das tabelas não podem ser repetidos, ou seja, em uma BD, cada tabela deve ser única. Elas podem se basear nas tabelas e entidades existentes nos Modelos anteriores, para facilitar na construção do Modelo Físico. Para completar a criação da tabela, é necessário prosseguir com alguns parâmetros característicos, identificados como Atributos, que seguem uma sintaxe determinada [9], como na **Figura 9**:

*[NOME\_CAMPO] [TIPO\_DO\_CAMPO] [TAMANHO\_TIPO] (NULL | NOT NULL)  
(DEFAULT [VALOR\_PADRÃO]) (AUTO\_INCREMENT);*

Primeiramente, deve-se indicar o nome do novo atributo (NOME\_CAMPO), logo depois o seu tipo (TIPO\_DO\_CAMPO) e seu respectivo tamanho, que representa a quantidade de caracteres que podem ser armazenados naquele atributo, mas não é obrigatório em tipos já determinados ou em tipos inteiros. Para indicar se o preenchimento do campo é obrigatório ou não, é necessário completar com NULL ou NOT NULL, sendo que no primeiro caso o valor nulo é válido, ou seja, pode ser “preenchido” com vazio, ao contrário do segundo caso, que não permite o valor de vazio e exige o preenchimento com caracteres. O parâmetro DEFAULT não é obrigatório, e pode alojar um conteúdo padrão para o atributo, quando o mesmo não for preenchido, e caso não for indicado, seu valor padrão é NULL. O AUTO\_INCREMENT indica que o atributo será preenchido automaticamente por uma sequência numérica crescente, iniciando em um (1), e prosseguindo perante a quantidade de instâncias que a mesma tabela possui, mas só pode haver um campo AUTO\_INCREMENT por tabela, e o mesmo não pode ser NULL ou possuir valor DEFAULT.

#### 2.1.2.3.1.2.1 Tipos de Campo

Os tipos existentes atendem à maioria dos dados e informações que são inseridos nos Bancos, e se estendem entre [10]:

- Dados Numéricos Exatos (NUMERIC, DECIMAL, INTEGER, SMALLINT);
- Dados Numéricos Aproximados (FLOAT, REAL, e DOUBLE PRECISION);
- Dados Alfabéticos (CHAR, VARCHAR, TEXT, ENUM, SET, LONGTEXT);



- Dados Cronológicos (DATE, DATETIME, TIMESTAMP, TIME, YEAR);

Os tamanhos dentre os tipos podem variar, e em alguns a alocação do dado pode ser feita de forma dinâmica ou estática, dependendo do tipo utilizado. Os Tipos de Campo mais comuns são:

- VARCHAR: Armazena uma cadeia de longitude variável, isto é, seu tamanho depende da quantidade armazenada, mas ainda possui tamanho máximo;
- INTEGER: Número inteiro, de tamanho comum;
- FLOAT: Número de ponto flutuante de precisão simples (32 bits);
- DATE: Valor de Data no formato 'AAAA-MM-DD'.

No interior da tabela, assim como nos Modelos Conceituais e Lógicos, ainda é necessário indicar a Chave Primária, que deve referenciar outro campo já existente na Tabela, e funcionará como um referencial único para cada instância:

*PRIMARY KEY ([NOME\_CHAVE\_PRIMARIA])*

Também podemos indicar uma Chave Estrangeira, caso for necessário, mas, assim como na Chave Primária, a Chave Estrangeira também deve referenciar um campo já existente na Tabela, além de referenciar um Campo e Tabela de onde se originou:

*FOREIGN KEY ([NOME\_CHAVE\_ESTRANGEIRA])*

*REFERENCES [NOME\_BANCO\_ORIGEM] ([NOME\_CHAVE\_PRIMARIA\_ORIGEM])*

```

4 CREATE TABLE PCD(
5   NOME_PCD VARCHAR(100) NOT NULL,
6   NUMERO_PCD INTEGER NOT NULL AUTO_INCREMENT,
7   DATA_FUNCIONAMENTO_PCD DATE NULL,
8   REGIAO_PCD INTEGER NOT NULL,
9   PRIMARY KEY (NUMERO_PCD),
10  FOREIGN KEY (REGIAO_PCD) REFERENCES REGIOES(REGIAO_PCD)
11 );

```

**Figura 9 Comando CREATE TABLE**

Caso seja necessário, também é possível apagar qualquer tabela do Banco de Dados, porém, caso efetuado, perderá todas as instâncias daquela tabela, e muitas vezes esse comando pode ser interrompido por existir instâncias que usam atributos da Tabela que seria apagada. O comando segue uma sintaxe repetida:

```
DROP TABLE [NOME_TABELA];
```

#### 2.1.2.3.1.3 INSERT

O comando INSERT é utilizado para inserir instâncias de uma tabela em um Banco de Dados. Ele se aplica, como por exemplo, em momentos de efetuar cadastros ou criar novos objetos [8]. Seu comando segue a seguinte sintaxe:

```
INSERT INTO [NOME_TABELA] ([CAMPO_1], [CAMPO_2]..., [CAMPO_N])  
VALUES ([VALOR_1], [VALOR_2]..., [VALOR_N]);
```

Os valores inseridos logo após a cláusula VALUE devem corresponder à mesma ordenação dos campos. Neste comando, indicar a Chave Primária não é usual, no entanto, em casos que ela esteja presente, seu valor deve ser indicado como NULL. Caso houver a presença de campos de tipos alfabéticos, como VARCHAR e CHAR, os valores devem estar entre aspas simples, pois se trata de uma String (cadeia de caracteres). Enfim, na existência de um campo contendo valor DEFAULT, caso não for indicado, ele receberá o valor pré-definido pelo DEFAULT. A seguir, há uma imagem demonstrativa sobre a aplicação de um comando INSERT, na **Figura 10**:

```
13 • INSERT INTO PCD(NOME_PCD,NUMERO_PCD,DATA_FUNCIONAMENTO_PC  
14 VALUES ('PCD 1',NULL,'2016-01-01',1);
```

**Figura 10 Comando INSERT**

#### 2.1.2.3.1.4 DELETE

O comando DELETE permite excluir uma ou mais instâncias de uma mesma tabela. Basta indicar a tabela e filtrar as instâncias até selecionar especificamente quais deverão ser alteradas. E esta é sua estrutura:

```
DELETE FROM [NOME_TABELA]  
WHERE [CAMPO_WHERE] = [CONTEUDO_CAMPO_WHERE];
```

Portanto, o DELETE necessita da clausula WHERE para filtrar e indicar qual ou quais instâncias devem ser excluídas, caso ela não for especificada, todas as instâncias da tabela serão excluídas, por isso é necessário aplicar este comando com muita atenção [8]. Na **Figura 11**, segue um exemplo da aplicação desse comando:

```
16 • DELETE FROM PCD
17 WHERE NUMERO_PCD = 1;
```

**Figura 11** Comando DELETE

O comportamento do DELETE para tabelas com Chaves Estrangeiras pode variar, dependendo do tratamento que essas Chaves recebem.

#### 2.1.2.3.1.5 UPDATE

O comando UPDATE permite atualizar os atributos ou colunas de instâncias ou registros já existentes. Ele pode ser usado para alterar todas as colunas, ou apenas colunas específicas.

```
UPDATE [NOME_TABELA] SET
[CAMPO_1] = [VALOR_CAMPO_1],
[CAMPO_2] = [VALOR_CAMPO_2],
[CAMPO_N] = [VALOR_CAMPO_N]
WHERE [CAMPO_WHERE] = [CONTEUDO_CAMPO_WHERE];
```

Primeiramente, deve-se indicar a tabela no campo NOME\_TABELA, referenciando onde estão os atributos que se deseja alterar os valores. O SET indica o início da listagem de quais atributos devem ser alterados, seguidos de seu novo valor: CAMPO\_N = VALOR\_CAMPO\_N. Caso haja mais de um atributo a ser modificado, eles devem vir separados por vírgula, exceto o último. E atributos que forem determinados como Chaves Primárias ou AUTO\_INCREMENT não podem ser alterados. Por fim, utilizando-se da clausula WHERE é possível indicar quais dos registros existentes que se deseja alterar os valores, utilizando-se de comparações de dados, mas se ele não for especificado, o comando irá atualizar todas as instâncias da tabela [8]. Para uma melhor compreensão, a **Figura 12** demonstra esse comando:

```

19 • UPDATE PCD SET
20 NOME_PCD = 'PCD 2',
21 DATA_FUNCIONAMENTO = '2016-12-01'
22 WHERE NUMERO_PCD = 1;

```

Figura 12 Comando UPDATE

#### 2.1.2.3.1.6 SELECT

Este último comando, não menos importante, é utilizado para retornar registros de uma ou mais tabelas, ou seja, ele retorna os valores de atributos pertencentes á instâncias de tabelas. Sua sintaxe é simples:

```

SELECT [DISTINCT] [EXPRESSÃO_SELECT_1], [EXPRESSÃO_SELECT_N]
FROM [NOME_TABELA_1], [NOME_TABELA_2], [NOME_TABELA_3]
WHERE [CAMPO_WHERE] = [CONTEUDO_CAMPO_WHERE]
GROUP BY [EXPRESSÃO_SELECT_1], [EXPRESSÃO_SELECT_N]
ORDER BY [EXPRESSÃO_SELECT_1], [EXPRESSÃO_SELECT_N]
LIMIT [NUMERO_LIMITE];

```

Inicialmente, indicamos quais atributos queremos retornar, indicados pela EXPRESSÃO\_SELECT, que pode ser qualquer coluna pertencente a qualquer tabela existente. E caso houver mais de uma expressão, elas devem vir separadas por vírgula, exceto a última. Em casos onde é necessário retornar todos os atributos de alguma tabela, utiliza-se Asterisco (\*), que irá referenciar todas as colunas. O DISTINCT não é obrigatório, mas ao utilizá-lo, é especificado que todos os registros duplicados no conjunto de resultados devem ser removidos.

Depois de indicar quais serão os atributos, é necessário indicar de quais tabelas esses atributos se originam. Os nomes das tabelas devem estar indicados logo após a cláusula FROM, indicado no campo NOME\_TABELA. Em caso de mais de um nome de tabela, elas devem vir separadas por vírgula, exceto a última.

O uso do WHERE também não é obrigatório, no entanto, sem sua presença, o comando SELECT irá retornar todas as instâncias registradas nas tabelas selecionadas. A cláusula WHERE, se definida, especifica os filtros que deverão ser aplicados na seleção, reduzindo as ocorrências. A consulta tende a ficar sutilmente mais rápida quanto menor o número de ocorrências [8].

As clausulas a seguir não são obrigatórias em uma seleção, porém elas podem auxiliar na visualização e na filtragem dos dados:

- **GROUP BY:** Define o agrupamento que os dados devem adotar, indicado através dos atributos, representados pelo campo EXPRESSÃO\_SELECT.
- **ORDER BY:** Define a ordem que os dados devem ser apresentados, indicado através dos atributos, representados pelo campo EXPRESSÃO\_SELECT. Sob necessidade de ordenar inversamente, é necessário adicionar a notação DESC (descendente) ao nome da coluna na cláusula ORDER BY na qual você está ordenando. A ordem padrão é ascendente, ela pode ser especificada usando a notação ASC.
- **LIMIT:** Define o limite de linhas que devem ser retornadas na consulta, usando-se de até dois argumentos número e inteiros. Com apenas um argumento, é indicado o limite de linhas que o comando deve percorrer, iniciando do primeiro registro. Com dois argumentos, o primeiro indica qual é o índice, dentre os resultados, que a consulta deve iniciar, e o segundo indica o limite de linhas que ela deve retornar.

Utilizando diferentes combinações de cláusulas no comando SELECT, é possível determinar inúmeros resultados diferentes, inclusive é possível realizar consultas que se baseiam ou relacionam com outras consultas, fundindo estruturas de SELECT, chamando-se assim SELECT Aninhado. No entanto, este método é mais aprofundado e requer mais habilidade com a manipulação de Banco de Dados.

Enfim, este comando é muito utilizado para retornar os valores presentes no Banco de Dados, sendo assim um dos mais utilizados no Módulo de Relatórios (Módulo 04), presente no projeto Indra. A próxima imagem (**Figura 13**) contém a exemplificação desse comando:

```
24 • SELECT NOME_PCD, NUMERO_PCD, DATA_FUNCIONAMENTO_PCD
25 FROM PCD
26 WHERE NUMERO_PCD = 1;
```

**Figura 13 Comando SELECT**

#### 2.1.2.3.1.7 WHERE

A cláusula WHERE, é responsável pela filtragem dos registros através de qualquer modificação ou consulta no Banco de Dados (UPDATE, DELETE, SELECT), ou seja, sempre que você precisa acessar uma instância já existente no Banco. Sem sua presença, não é possível especificar as linhas requisitadas de uma ou mais tabelas, dificultando qualquer manipulação de informação, além de tornar o uso do Banco de Dados consideravelmente mais lento, eventualmente buscando resultados não desejados.

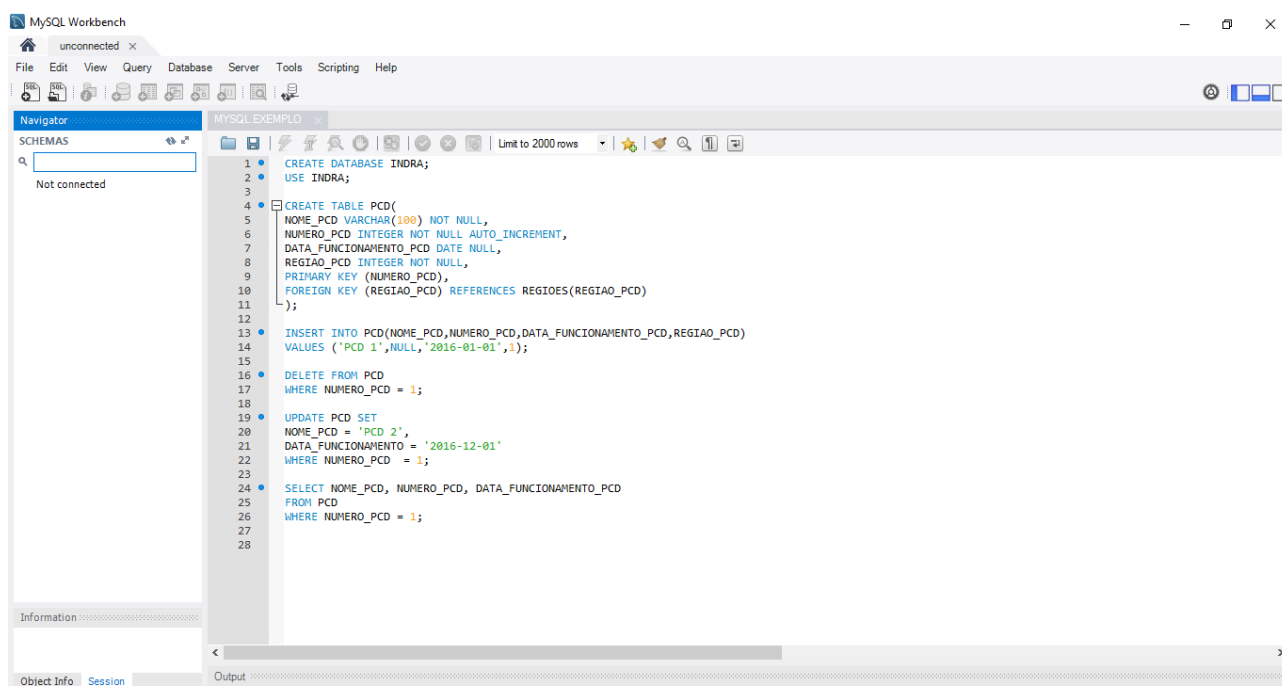
Na cláusula WHERE é possível ter mais de uma condição, portanto é necessário o uso das cláusulas AND (ambas as ocorrências precisam ser verdadeiras para retornar verdadeiro) e/ou OR (qualquer das ocorrências podem ser verdadeiras para retornar verdadeiro) para conjugar as condições, e em caso de agrupamento deve-se adotar o uso de parênteses [12].

Nesta cláusula é possível utilizar diferentes operadores para especificar a filtragem das informações [13]. A seguir, seguem alguns operadores, que também foram utilizados no projeto Indra:

OPERADOR	DESCRIÇÃO
=	Igual
<>	Diferente
<=	Menor e igual
>=	Menor e igual
<	Menor
>	Maior
BETWEEN	Entra faixa de valores
NOT BETWEEN	Fora de faixa de valores
LIKE	Parte do conteúdo
NOT LIKE	Não contenha parte do conteúdo
IS NULL	Conteúdo é NULL
IS NOT NULL	Conteúdo não é NULL
IN	Lista de valores
NOT IN	Fora da lista de valores

Estes são os comandos e cláusulas mais utilizados no sistema Indra para a construção do Modelo Físico de Banco de Dados. Nas aplicações que lidam com a manipulação de dados está incluso ao menos uma das funções do acrônimo CRUD (Create, Read, Update, Delete), referente às quatro operações básicas utilizadas em Bases de Dados.

No projeto em questão, será utilizado o Sistema de Gerenciamento de Banco de Dados MySQL, sistema pelo qual os envolvidos já haviam trabalhado nos anos anteriores do Ensino Médio.



**Figura 14 Modelo Físico MySQL Workbench**

Estes são os Modelos que foram implementados na Modelagem de Banco de Dados do sistema Indra. Mas é preciso enfatizar que estes são exemplos de aplicação, e não são regras de modelagem. Existem outras abordagens que podem ser utilizadas, mas neste caso, o projeto é composto pelas quais foram descritas nesse capítulo. Esses são os primeiros conceitos para se iniciar a construção de um projeto que aplica a estruturação de dados, usados para a documentação das etapas e da evolução do BD, obtendo assim uma ferramenta de apoio fundamental e para a implementação e manutenção do futuro sistema.

## 2.2 Metodologia do Trabalho

A Metodologia deste documento se baseia nas estratégias e técnicas utilizadas para o desenvolvimento do Banco de Dados do sistema Indra, utilizando-se da Modelagem de Dados, portanto, para não tangenciar o tema, alguns aspectos do projeto não serão abordados, a fim de não comprometer a objetividade deste Trabalho de Finalização de Curso. No entanto, há a necessidade de explicar alguns processos triviais que permitiram a equipe alcançar a construção do Sistema de Banco de Dados, portanto posteriormente serão apresentadas as atividades que antecederam a modelagem de dados, mas que contribuíram para a mesma.

### 2.2.1 Objetivos do Projeto: Indra

Indra, Deus dos Céus e das Tempestades na mitologia Hindu, é o nome do projeto voltado aos desastres naturais e suas conseqüências na cidade de São João da Boa Vista. Este projeto, composto pela equipe do 4º Ano do Ensino Médio, curso Técnico Integrado de Informática, tem o objetivo de prover uma plataforma WEB de meteorologia simplificada para os cidadãos, utilizando-se de dados climáticos recepcionados pelas Plataformas de Coletas de Dados (PCD). A fim de reduzir os impactos de catástrofes imprevisíveis sobre a cidade, há o tratamento dos dados meteorológicos coletados para análise e previsão de possíveis eventos climáticos e assim divulgar á população através da plataforma WEB, que será composta por inúmeras funcionalidades para consulta, contato e troca de informações.

Portanto, para que o objetivo do projeto se concretize, há a necessidade de aplicar técnicas de Informática, especificamente técnicas de Manipulação WEB e Administração de Banco de Dados. Usando-se de ferramentas de programação e um SGBD (Sistema de Gerenciamento de Banco de Dados), é possível implementar as funcionalidades que o sistema promete. As páginas WEB serão o meio pelo qual os usuários terão acesso ás informações que os administradores irão disponibilizar, referente á situação climática da região, inclusive alertar de possíveis riscos ou catástrofes. Este ambiente deve ser simples para sua manipulação, abstraindo todos os dados técnicos do site, mas permitindo o entendimento sucinto de cada funcionalidade. O SGBD deve permitir aos administradores o total controle das informações que serão utilizadas e armazenadas pelos sistema, a fim de garantir o acesso dos usuários, a proteção dos dados e o tratamento das informações climáticas.

No entanto, antes de iniciar o trabalho com as técnicas e ferramentas, foi necessário concluir outros processos para fixar os objetivos deste projeto.

#### 2.2.1.1 Atividades Primárias

Inicialmente, os alunos foram organizados para atender as primeiras tarefas triviais, formando equipes de Desenvolvedores (13 alunos) e equipes de Analistas (10 alunos) e Administradores de Banco de Dados (10 alunos), ignorando a formação inicial dos Módulos.

Os Desenvolvedores se responsabilizaram pelo Ícone (símbolo que representa o projeto), Nome (referente á suas funções) e pelo Modelo de Página WEB (padrão de página utilizada para todas as camadas do sistema). A equipe realizou algumas pesquisas para identificar símbolos e



nomes que relacionavam á Clima, uma das temáticas principais do projeto. Os integrantes que possuíam experiência ou técnica com a modelagem de imagens ficaram encarregados de representar os desenhos idealizados para o Ícone do projeto em plataformas de modelagem, onde a idéia poderia estar visível e assim poder se aprovada pelos outros integrantes. Os Desenvolvedores efetuaram algumas pesquisas e discutiram sobre o Nome que o projeto deveria adotar, vários integrantes contribuíram com idéias, e todas estavam relacionadas com o Clima. Então, foram definidas três propostas para o Nome e o Ícone, apresentadas posteriormente para todos os alunos votarem e aprovarem. Enfim, na **Figura 15** segue o ícone e o nome escolhido pela equipe para representar o projeto (o ícone possui a coloração originalmente branca, porém foi necessário alterar sua cor neste documento para destacá-lo):



**Figura 15 Ícone do Projeto**

Os Analistas e os Administradores de Banco de Dados trabalham em conjunto para construir o Termo de Abertura do projeto para formalizar e legalizar o desenvolvimento do mesmo, porém, apresentar outros documentos que tangenciam o tema deste trabalho não é o objetivo. Portanto, concluindo a apresentação do sistema Indra, os assuntos tratados a seguir serão referente á elaboração desenvolvimento do Banco de Dados desse sistema.

### 2.2.2 Indra: Necessidades de Banco de Dados

A utilização do Banco de Dados permite com que haja o controle das informações utilizadas pelo sistema. Em relação aos dados climáticos coletados pelas PDCs, eles são armazenados para que, posteriormente possa haver a sua consulta por diferentes usuários simultaneamente, atendo á função principal do sistema: conscientização da população sobre as informações meteorológicas da região. O gerenciamento do acesso dos usuários também é importante para atender a maior

variedade possível de pessoas ou entidades que possam usufruir das funções do sistema, portanto o Banco de Dados permite o gerenciamento desses usuários, garantindo a acessibilidade constante e, ao mesmo tempo, controlando a disposição das informações. Enfim, o Banco de Dados se tornou a técnica mais prática para o armazenamento de informações deste projeto, sendo assim adotado pela equipe em seus Modelos, e integrado às suas funcionalidades.

Ciente da importância do Banco de Dados para o projeto, tornou-se necessário que seu desenvolvimento tivesse início com antecedência, independente do momento em que o mesmo fosse aplicado no sistema, pois problemas poderiam surgir durante o seu processo de construção e assim atrasar o progresso da equipe. Portanto, os alunos foram orientados a desenvolvê-lo paralelamente à construção do sistema, delegando novas tarefas aos Administradores de Banco de Dados (DBA) de cada módulo, a fim de não concentrar tarefas em outros integrantes que já estavam trabalhando.

#### 2.2.2.1 Atividades: Modelagem de Dados

Os DBAs foram designados para planejar e desenvolver o Banco de Dados que seria utilizado pelo sistema. Mas, primeiramente, eles deveriam se concentrar em um Banco que atendesse apenas as funcionalidades de seus Módulos, respectivamente, tornando o trabalho mais fácil e permitindo com que cada Módulo identificasse quais eram suas necessidades em relação ao armazenamento de dados. O trabalho segmentado auxilia no tratamento dos detalhes da atividade, neste caso possibilitando com que cada Módulo trabalhasse individualmente nas partes que formariam o Banco de forma específica e precisa.

Durante o trabalho individual de cada Módulo, inicialmente os DBAs realizaram discussões para idealizar o Banco, construindo idéias generalizadas sobre as necessidades de entidades e atributos que iriam compor. Cada Módulo necessariamente iria tratar de funções diferentes dentro do projeto, conseqüentemente fazendo com que seus Bancos de Dados se diferenciasssem em sua lógica e em sua composição. No entanto, alguns Módulos precisavam dividir suas idéias quando alguma Entidade de seus Bancos era compartilhada entre outros Bancos, precisando-se então padronizar as características dessas Entidades dentre os Módulos que a utilizavam. Durante a idealização dos Bancos, os integrantes construíam os modelos MER e DER, para progredir em uma visão cada vez mais completa da Base de Dados.

Depois de planejar a estrutura geral, eles aplicavam o Banco de Dados nos Protótipos referentes às funções do sistema, produzidos pelos Desenvolvedores de cada Módulo, através do

Modelo Físico, a fim de testar a conexão desses protótipos com os Bancos e, caso necessário, corrigir possíveis problemas durante a aplicação. O Modelo Físico era herdado dos modelos anteriores, assimilando as características que os DBAs julgavam necessário.

Após os DBAs confirmarem que seus Bancos e Protótipos estavam funcionando em conjunto, a próxima atividade incluía a integração de todas as Bases de Dados, que até então estavam fragmentadas entre os Módulos. A integração precisou do envolvimento de todos os DBAs para formular novos Modelos MER, DER e Físico do novo Banco de Dados, pois agora envolvia as características de todos esses Módulos, por isso os DBAs de cada equipe deveriam estar presentes. Integrando o Banco em uma única estrutura, permitia que as funcionalidades do projeto operassem com apenas uma Base de Dados, facilitando na sua manutenção e no desenvolvimento dos protótipos. Enfim, a partir da integração, as atividades com o Banco se concentraram em sua constante manutenção, atendendo aos requisitos e necessidades dos outros integrantes, principalmente dos Desenvolvedores, que constantemente utilizavam-se de informações armazenadas.

Esses foram os métodos adotados para a construção do Banco de Dados do projeto Indra. Nos próximos tópicos serão abordados os mesmos métodos, porém explicando as técnicas e estratégias adotadas para aplicá-los, exemplificando-as nas atividades do Módulo 4 (Relatórios), composto por dois Administradores de Banco de Dados.

### 2.2.3 Idealizando o Banco de Dados

Para a elaboração da Base de Dados, as equipes de cada Módulo utilizaram-se das técnicas de Modelagem de Dados, pois esse foi o método estudado e praticado por todos os alunos no decorrer do curso, durante as aulas de Banco de Dados (BDD), através do programa brModelo. Portanto, aplicar o conhecimento já adquirido sob uma técnica reconhecida tornaria o processo de construção do Banco muito mais produtivo, além de que a Modelagem proporciona etapas de trabalho que podem assegurar o potencial da Base de Dados do início ao fim.

No começo das atividades, os Módulos reuniram seus integrantes para discutir e refletir quais características os seus Bancos de Dados deveriam conter para atender às suas funções, e assim compor seu primeiro Modelo Conceitual (MER), utilizando-se do programa brModelo. Na discussão, era necessário a participação de todos, independentemente do seu papel, pois todos os integrantes e suas atividades seriam afetados pelo Banco, portanto para que nenhum papel fosse comprometido, os integrantes deveriam incluir o seu ponto de vista.

É importante, em uma reflexão sobre a estrutura inicial do Banco de Dados, analisar o sistema de negócio por completo, focando nos aspectos mais comuns, como Entidades e Atributos, pois característica mais complexas, como os Relacionamentos, poderiam ser melhor adaptados posteriormente. Neste caso, o MER proporcionar as características básicas para se criar as informações que irão compor as regras do negócio, bem como os envolvidos, suas características e a forma como se relacionam.

Portanto, a técnica do Modelo Conceitual se tornou a base para a elaboração inicial dos Bancos de Dados, sendo adotada por todos os Módulos do projeto. O Módulo 4 (Relatórios) atendeu aos requisitos da atividade e inclui, em seu primeiro modelo, as entidades que supostamente iriam participar dos negócios para a consulta de relatórios e apresentações de diferentes dados sobre as PCDs e suas Medições.

#### 2.2.3.1 Módulo de Relatórios: Modelo Conceitual

O MER desenvolvido pela equipe do Módulo 4 (Relatórios) não apresentou uma estrutura complexa, como demonstra a **Figura 16**, pois o mesmo módulo concentrava funcionalidades para a consulta de informações, precisando-se apenas de se conectar com as entidades responsáveis pelo armazenamento dessas informações, que posteriormente seriam tratadas para tornar a consulta disponível na página WEB do projeto.

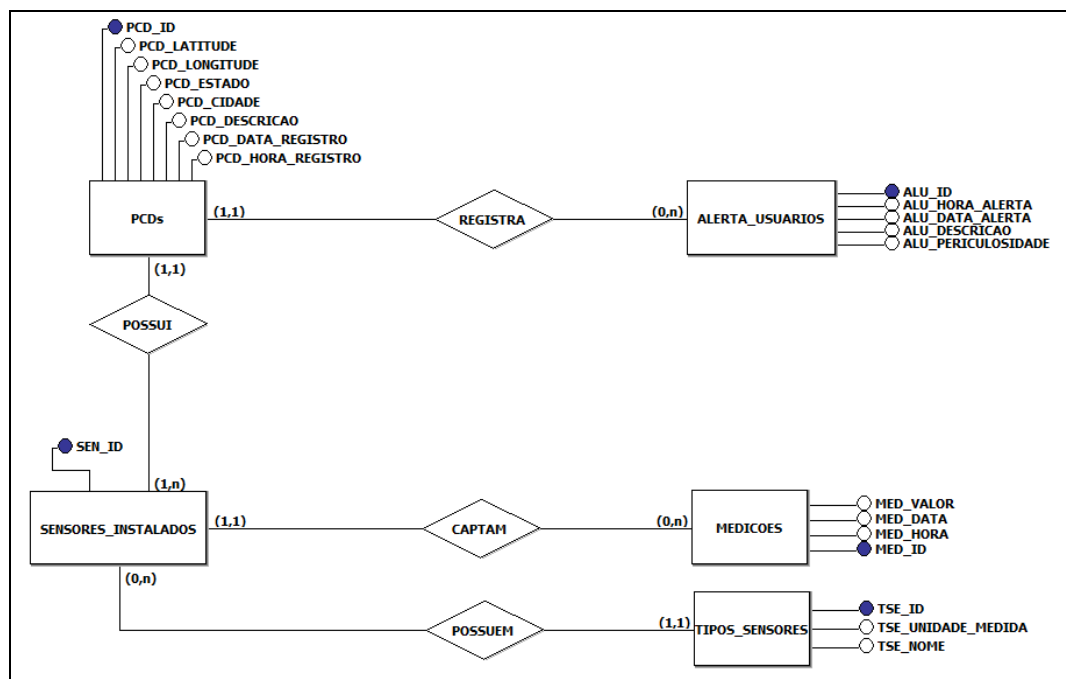


Figura 16 Modelo Conceitual do Módulo 4

#### 2.2.3.1.1 MOD04 - Modelo Conceitual: Entidades

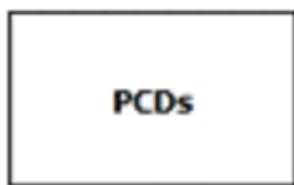
Inicialmente, com a equipe do módulo reunida, foram feitas trocas de ideias sobre quais entidades deveriam estar presentes no Banco para que a consulta seja bem-sucedida. Se tratando da consulta de Alertas Públicos, PCDs, Medições, Sensores e Tipos de Sensores, a elaboração do MER necessitava apresentar entidades que armazenassem as informações respectivas às essas consultas:

- [ALERTAS\_USUARIOS]: Esta Entidade (**Figura 17**) armazena os alertas que são criados pela própria população, com o propósito de permitir com que qualquer cidadão com acesso ao sistema pudesse notificar os administradores sobre perigos ou catástrofes que ocorreram e forem registradas. Esse feedback é importante para garantir que, em caso onde as PCDs não possam registrar os perigos climáticos, a população possa ainda ser notificada através de alertas criados por outros cidadãos, e também incentivar a participação constante da comunidade dentro do sistema.



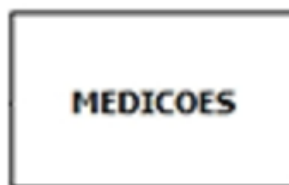
**Figura 17 Entidade ALERTA\_USUARIOS**

- [PCDs]: Esta Entidade (**Figura 18**) permite o registro das características das PCDs, tornando possível o controle do armazenamento de suas respectivas medições. As PCDs também podem ser consultadas pelos usuários, a fim de identificar quais estão presentes em sua região e sob quais ele possui permissão para consultar os dados.



**Figura 18 Entidade PCDs**

- [MEDICOES]: Esta Entidade (**Figura 19**) armazena os dados referentes as medições de cada PCD, proporcionando o registro constante de todos os valores identificados pelas plataformas em períodos de tempo contínuo. Desta forma, o acesso à material para consulta sempre estará disponível e atualizado.



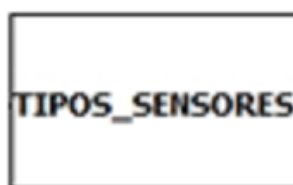
**Figura 19 Entidade MEDICOES**

- [SENSORES\_INSTALADOS]: Esta Entidade (**Figura 20**) representa o molde para o registro de sensores instalados em respectivas PCDs. Os sensores variam de acordo com a necessidade de medição, mas neste caso, em sua maioria eles estão direcionados a receptor dados referentes ao clima da região onde a PCD está posicionada. Ela é necessária para identificar quais sensores estão instalados nas PCDs e identificar a origem de suas medições.



**Figura 20 Entidade SENSORES\_INSTALADOS**

- [TIPOS\_SENSORES]: Esta Entidade (**Figura 21**) permite a listagem dos tipos de sensores existentes nas PCDs e assim identificar quais tipos de medições os sensores registram. Durante as consultas, é necessário que as informações estejam identificadas e fáceis de compreender, portanto, no momento de se apresentar os dados das medições de uma PCD, precisa-se definir o que representa cada dado em relação ao clima, portanto a entidade TIPOS\_SENSORES prove a identificação desses valores.



**Figura 21 Entidade TIPOS\_SENSORES**

#### 2.2.3.1.2 MOD04 - Modelo Conceitual: Atributos

No entanto, apesar da existência das Entidades, para o armazenamento de informações as mesmas precisaram ser descritas através de Atributos, a fim de definir quais seriam os dados que iriam compor as Entidades. Estes Atributos atendiam às características de suas Entidades, mas em alguns casos poderiam surgir Atributos iguais, representando a mesma informação, mas se diferenciando através dos seus nomes, constituído por uma Trígrama. A Trígrama é uma sigla composta por três letras que representam somente a respectiva Entidade ao qual o Atributo pertence:

- [ALERTAS\_USUARIOS]: Os Atributos da entidade ALERTAS\_USUARIOS possuem o objetivo de armazenar dados referentes ao período e propósito do alerta enviado pelo usuário:

##### Trígrama: ALU

Atributo	Descrição
ALU_ID	Código do alerta enviado pelo usuário. Chave primária da tabela.
ALU_HORA_ALERTA	Horário do alerta enviado pelo usuário.
ALU_DATA_ALERTA	Data do alerta enviado pelo usuário.
ALU_PERICULOSIDADE	Nível do alerta enviado pelo usuário, conforme sua periculosidade.
ALU_DESCRICAO	Descrição do alerta enviado pelo usuário, contendo previsão de algum desastre ocorrer.

- [PCDs]: Os Atributos pertencentes à entidade PCDs registram características essenciais que descrevem as PCDs catalogadas no Banco de Dados, assim como sua localização, descrição e horário de cadastro:

##### Trígrama: PCD

Atributo	Descrição
PCD_ID	Código de identificação de cada PCD. Chave primária da tabela.
PCD_DESCRICAÇÃO	Descrição básica da PCD conforme sua funcionalidade.
PCD_LATITUDE	Latitude da PCD cadastrada.
PCD_LONGITUDE	Longitude da PCD cadastrada.
PCD_ESTADO	Nome completo ou sigla do estado em que a PCD cadastrada está situada.
PCD_CIDADE	Nome completo da cidade em que a PCD cadastrada está situada.
PCD_HORA_REGISTRO	Hora de registro do cadastro da PCD.
PCD_DATA_REGISTRO	Data de registro do cadastro da PCD.

- [MEDICOES]: Os Atributos localizados na entidade MEDICOES armazenam os dados utilizados para tratamento e consulta das medições efetuadas pelas PCDs:

**Trigrama: MED**

Atributo	Descrição
MED_ID	Código da medição captada. Chave primária da tabela.
MED_TIME	Horário da medição captada pelo sensor.
MED_DATA	Data e hora da medição captada pelo sensor.
MED_VALOR	Valores de medição captados pelo sensor instalado.

- [SENSORES\_INSTALADOS]: A entidade SENSORES\_INSTALADOS é composta por apenas um Atributo, que representa sua Chave Primária, pois no momento da idealização desta Entidade no modelo MER não se acreditava que qualquer outro de seus possíveis Atributos que causariam qualquer impacto no Banco de Dados do Módulo 4 (Relatórios). No entanto, durante o desenvolvimento dos Modelos posteriores foi observado que esta Entidade na verdade pode afetar a consulta de dados de outras Entidades. Portanto este estudo sobre os Atributos necessários foi importante para identificar brevemente quais poderiam ser os possíveis equívocos do Banco de Dados Físico caso a Modelagem de Dados não fosse devidamente aplicada. Neste caso, há a presença apenas da Chave Primária:

**Trigrama: SEN**

Atributo	Descrição
SEN_ID	Código dos sensores instalados. Chave primária da tabela.

- [TIPOS\_SENSORES]: Os Atributos presentes na entidade TIPOS\_SENSORES, quando agrupados, formam a informação referente ao tipo de unidade de medição e o nome do seu respectivo sensor:

**Trigrama: TSE**

Atributo	Descrição
TSE_ID	Código do tipo de sensor instalado. Chave primária da tabela.
TSE_UNIDADE_MEDIDA	Tipo de conversão de unidade de medida da medição realizada pelo sensor instalado.
TSE_NOME	Nome do sensor instalado, conforme sua funcionalidade.



### 2.2.3.1.3 MOD04 - Modelo Conceitual: Relacionamentos

Relacionamentos...

[REGISTRA]: (Figura 22)



Figura 22 Relacionamento REGISTRA

[POSSUI]: (Figura 23)

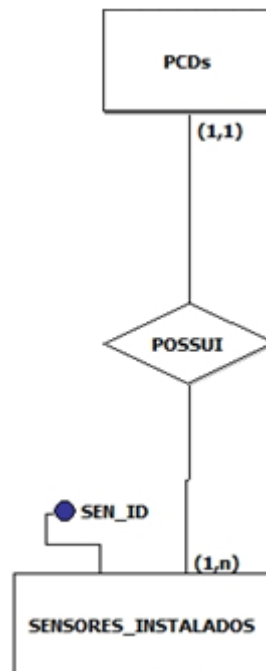


Figura 23 Relacionamento POSSUI

[CAPTAM]: (Figura 24)

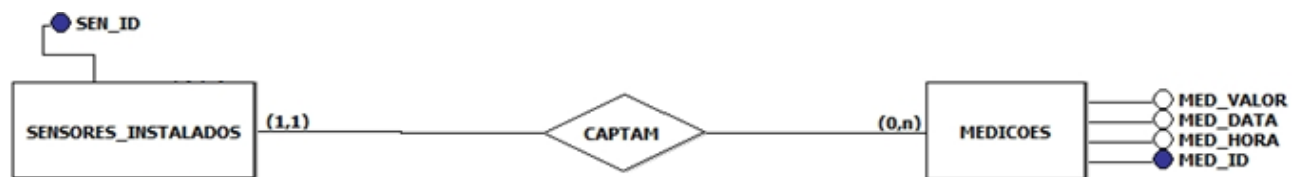


Figura 24 Relacionamento CAPTAM

[POSSUEM]: (Figura 25)

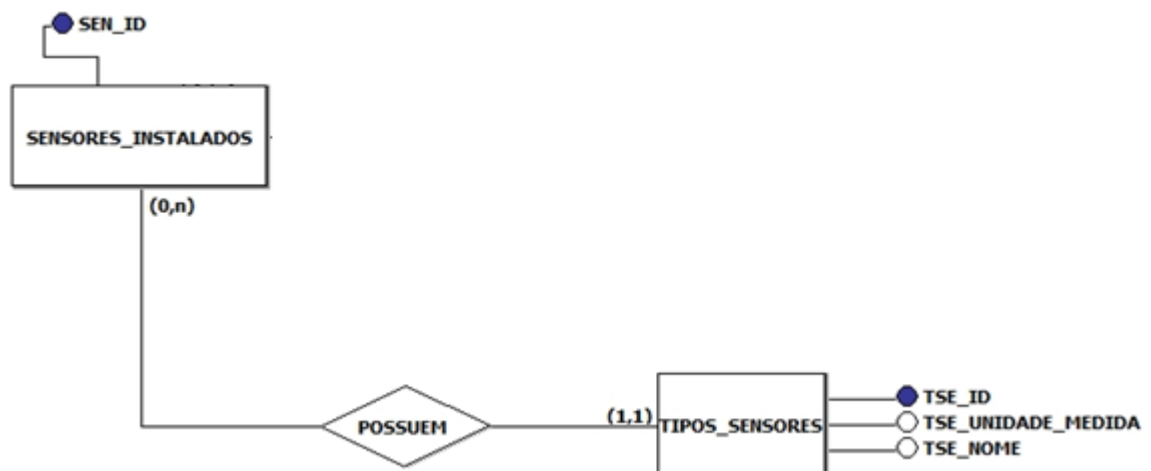
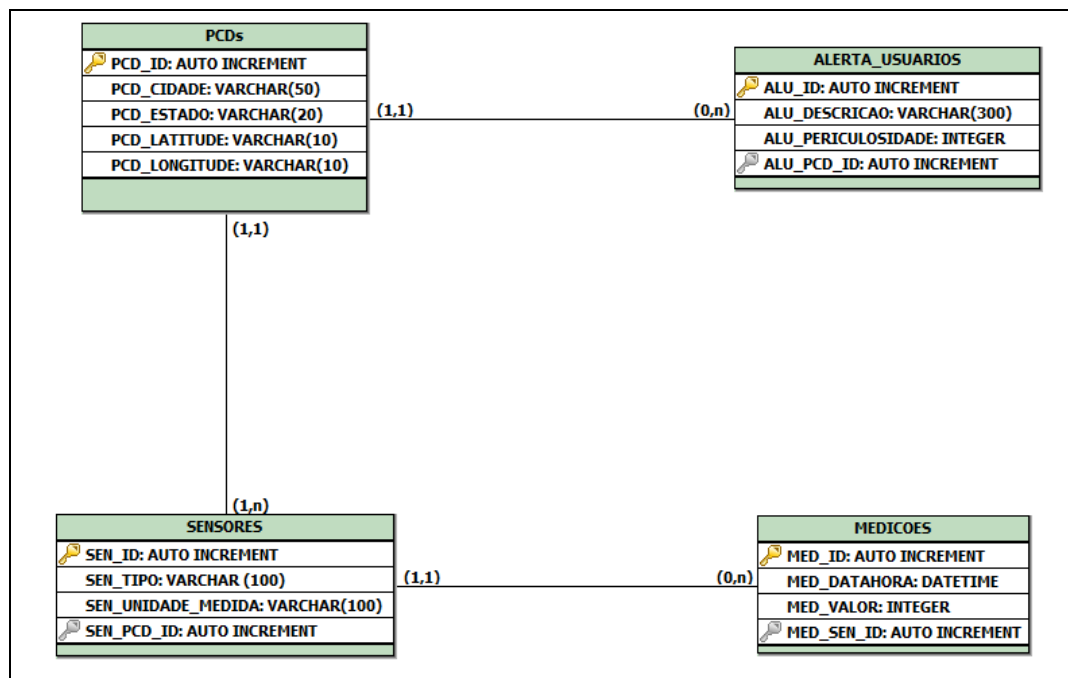


Figura 25 Relacionamento POSSUEM

### 2.2.3.2 Módulo de Relatórios: Modelo Lógico

DER...



Diferenças do DER para o MER....No entanto, apesar de incluir os aspectos mais simples do Banco, surgiam eventualmente problemas e incoerências durante a elaboração do Modelo Conceitual.

#### 2.2.3.3 Dificuldades para a Idealização

### 2.2.4 Aplicando o Banco de Dados

#### 2.2.4.1 Módulo de Relatórios: Modelo Físico

#### 2.2.4.2 Dificuldades para a Aplicação

### 2.2.5 Integrando o Banco de Dados

#### 2.2.5.1 Dificuldades para a Integração

### **3 Conclusões e Recomendações**

Texto...

## 4 Referências Bibliográficas

[1] Entenda a importância do banco de dados da sua empresa - Blog GMPE – 24/07/2016 – Disponível em: <<https://gmpe.com.br/blog/-entenda-a-importancia-do-banco-de-dados-da-sua-empresa-18.html>>. Acesso em: 26/08/2016.

[2] FURTADO, Gustavo – O que é Banco de Dados? – Dicas de Programação –26/03 – Disponível em: <<http://www.dicasdeprogramacao.com.br/o-que-e-um-banco-de-dados/>>. Acesso em: 24/09/2016.

[3] REZENDE, Ricardo – Conceitos Fundamentais de Banco de Dados – DEVMedia – 24/09/2016 – Disponível em: <<http://www.devmedia.com.br/conceitos-fundamentais-de-banco-de-dados/1649>>. Acesso em: 24/09/2016.

[4] O que é um banco de dados? – CCM – 07/2016 – Disponível em: <<http://br.ccm.net/contents/65-bancos-de-dados-introducao>>. Acesso em: 24/09/2016.

[5] MACORATTI, José Carlos – Conceitos Básicos de modelagem de dados – Macoratti – 24/09/2016 – Disponível em: <<http://www.macoratti.net/cbmd1.htm>>. Acesso em: 24/09/2016.

[6] RODRIGES, Joel – Modelo Entidade Relacionamento (MER) e Diagrama Entidade-Relacionamento (DER) – DEVMedia –24/09/2016 – Disponível em: <<http://www.devmedia.com.br/modelo-entidade-relacionamento-mer-e-diagrama-entidade-relacionamento-der/14332>>. Acesso em: 24/09/2016.

[7] Modelagem de dados: modelo conceitual, modelo lógico e físico – #luis.blog.br – 24/09/2016– Disponível em: <<http://www.luis.blog.br/modelagem-de-dados-modelo-conceitual-modelo-logico-e-fisico.aspx>>. Acesso em: 24/09/2016.

[8] Comandos básicos do MySQL (Insert, Delete, Update e Select) – RLSYSTEM – 10/10/2016– Disponível em: <<http://www.rlsystem.com.br/outros/noticia/select-insert-delete-update-mysql/>>. Acesso em: 10/10/2016.

[9] MySQL: comandos básicos – ELIAS PRACIANO - OS TUTORIAIS DO GEEK SEARCH – 19/02/2013 – Disponível em: <<https://elias.praciano.com/2013/02/mysql-comandos-basicos/>>. Acesso em: 10/10/2016.

[10] Mysql: tipos de dados – ELIAS PRACIANO - OS TUTORIAIS DO GEEK SEARCH – 11/01/2014 – Disponível em: <<https://elias.praciano.com/2014/01/mysql-tipos-de-dados/>>. Acesso em: 10/10/2016.

[11] Tipos de Campos no MySQL, saiba como escolher o tipo correto – RECOLI – 10/10/2016 – Disponível em: <<http://www.rcoli.com.br/2012/08/tipos-de-campos-no-mysql-saiba-como-escolher-o-tipo-correto/>>. Acesso em: 10/10/2016.

[12] Usando operadores aritméticos, lógicos e comparação em consultas SQL – #luis.blog.br – 23/10/2016– Disponível em: <<http://www.luis.blog.br/operadores-aritmeticos-logicos-e-comparacao.aspx>>. Acesso em: 23/10/2016.

[13] 6.3.1.2. Operadores de Comparação – MySQL – 24/09/2016– Disponível em: <<http://ftp.nchu.edu.tw/MySQL/doc/refman/4.1/pt/comparison-operators.html>>. Acesso em: 24/09/2016.