

**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 e Prof. Luiz Angelo Valota Francisco

**Processo de criação do Banco de Dados do Módulo de Prescrições  
Médicas e Controle de Incidentes**

Aluno: Caroline Manzotti Rabello

Prontuário: 1620371

São João da Boa Vista – SP

2019

## **Resumo**

As Instituições de Longa Permanência em São João da Ba Vista têm, pelos mais diversos motivos, uma deficiência em relação a um software que auxilie a realização de algumas de suas funções. Por este motivo, os alunos do 4º ano do Integrado em Informática do Instituto Federal de Ciências e Tecnologia aceitaram o desafio de desenvolver um software que atendesse essas necessidades. O projeto, desenvolvido na disciplina de PDS (Prática de Desenvolvimento de Sistemas), conta com 9 módulos que são específicos para cada macro função. São desenvolvidos desde a documentação, prototipação, banco de dados e testes. Este trabalho tem como meta apresentar o processo de criação do Banco de Dados do Módulo de Prescrições Médicas e Controle de Incidentes, iniciando nos Casos de Uso, Modelo Conceitual, Modelo Lógico e por último o Modelo Físico. O primeiro capítulo compreende a mostrar o Diagrama de Casos de Uso e mostrar suas especificações. O segundo capítulo trata de demonstrar de maneira exemplificada a passagem dos Casos de Uso para o primeiro passo do Banco de Dados efetivamente, o Diagrama Entidade-Relacionamento, que compreende ao Modelo Conceitual. O terceiro capítulo apresenta o Diagrama Entidade-Relacionamento do Modelo Conceitual e explica cada uma de suas relações. O quarto capítulo continua a criação do Banco, mostrando o Modelo Lógico, gerado pela ferramenta brModelo a partir do Modelo Conceitual, além disso, explica algumas de suas particularidades, como chaves estrangeiras. O último capítulo engloba o código em SQL, também conhecido como o Modelo Físico do Banco de Dados. Ao final deste, conclui-se a criação do Banco de Dados do Modelo de Prescrições Médicas e Controle de Incidentes.

Palavras Chaves: Banco de Dados; Instituição de Longa Permanência; Projeto; Casos de Uso; Modelo Conceitual; Modelo Lógico; Modelo Físico.

## Sumário

1	Introdução .....	6
1.1	Contextualização/Motivação .....	6
1.2	Objetivo Geral da Pesquisa .....	8
1.3	Objetivos Específicos .....	8
1.4	Estrutura do Documento.....	8
2	Desenvolvimento .....	9
2.1	Levantamento Bibliográfico.....	9
2.1.1	SBGD (Sistema Gerenciador de Bando de Dados) .....	9
2.1.2	Modelo Conceitual .....	9
2.1.3	Modelo Lógico Relacional .....	11
2.1.4	Modelo Físico .....	12
2.2	Etapas para o Desenvolvimento da Pesquisa .....	14
2.2.1	Casos de Uso Documento de Casos de Uso .....	14
2.2.2	Passagem dos Casos de Uso para o Modelo Conceitual .....	18
2.2.3	Modelo Conceitual .....	19
2.2.4	Modelo Lógico .....	28
2.2.5	Modelo Físico .....	34
3	Conclusões e Recomendações .....	41
4	Referências Bibliográficas .....	43

## Sumário de Figuras

Figura 1 - Especificação dos módulos da divisão de Acesso Externo .....	7
Figura 2 - Especificação dos módulos da divisão de Gerenciamento dos Idosos.....	7
Figura 3 - Especificação dos módulos da divisão de Controle Gerencial.....	7
Figura 4 - Exemplo de Diagrama Entidade-Relacionamento .....	9
Figura 5 - Estrutura de funções SQL .....	13
Figura 6 - Diagrama de Casos de Uso.....	14
Figura 7 - Recorte 1 do Diagrama de Casos de Uso .....	15
Figura 8 - Recorte 2 do Diagrama de Casos de Uso .....	15
Figura 9 - Recorte 3 do Diagrama de Casos de Uso .....	16
Figura 10 - Recorte 4 do Diagrama de Casos de Uso .....	16
Figura 11 - Recorte 5 do Diagrama de Casos de Uso .....	17
Figura 12 - Recorte 6 do Diagrama de Casos de Uso .....	17
Figura 13 - Recorte 7 do Diagrama de Casos de Uso .....	17
Figura 14 - Exemplo da passagem dos Casos de Uso para o Modelo Conceitual .....	18
Figura 15 - Exemplo de entidades construtivas para o Modelo Conceitual.....	18
Figura 16 - Diagrama Entidade-Relacionamento.....	19
Figura 17 - Recorte 1 do Diagrama Entidade-Relacionamento .....	20
Figura 18 - Recorte 2 do Diagrama Entidade-Relacionamento .....	21
Figura 19 - Recorte 3 do Diagrama Entidade-Relacionamento .....	22
Figura 20 - Recorte 4 do Diagrama Entidade-Relacionamento .....	22
Figura 21 - Recorte 5 do Diagrama Entidade-Relacionamento .....	23
Figura 22 - Recorte 6 do Diagrama Entidade-Relacionamento .....	24
Figura 23 - Recorte 7 do Diagrama Entidade-Relacionamento .....	24
Figura 24 - Recorte 8 do Diagrama Entidade-Relacionamento .....	25
Figura 25 - Recorte 9 do Diagrama Entidade-Relacionamento .....	25
Figura 26 - Recorte 10 do Diagrama Entidade-Relacionamento .....	26
Figura 27 - Recorte 11 do Diagrama Entidade-Relacionamento .....	27
Figura 28 - Recorte 12 do Diagrama Entidade-Relacionamento .....	27
Figura 29 - Recorte 13 do Diagrama Entidade-Relacionamento .....	28
Figura 30 - Modelo Lógico .....	29
Figura 31 - Recorte 1 do Modelo Lógico.....	29
Figura 32 - Recorte 2 do Modelo Lógico.....	30
Figura 33 - Recorte 3 do Modelo Lógico.....	31

Figura 34 - Recorte 4 do Modelo Lógico.....	31
Figura 35 - Recorte 5 do Modelo Lógico.....	32
Figura 36 - Recorte 6 do Modelo Lógico.....	33
Figura 37 - Recorte 7 do Modelo Lógico.....	33

# 1 Introdução

## 1.1 Contextualização/Motivação

A cidade de São João da Boa Vista encontra-se no interior do estado de São Paulo, que atualmente conta com uma população estimada de 90.637 pessoas [1], e com o passar dos anos sua taxa de envelhecimento (razão entre a população de 65 anos ou mais de idade em relação à população total) vem crescendo [2]. Desse modo, há um grande número de idosos na região. Em uma pesquisa realizada pelo Instituto de Longevidade Mongeral Aegon em parceria com a FVG, das 348 cidades brasileiras com população entre 50 mil e 100 mil habitantes, onde São João da Boa Vista encontra-se em primeiro lugar para oferecer melhores condições de vida para essas pessoas. [3]

O município disponibiliza algumas Casas de Apoio para Idosos gratuitas para a população. Além disso, há alguns particulares. Em pesquisa feita na cidade, percebeu-se que em nenhuma delas há um sistema que auxilia na execução das tarefas da Instituição. A fim de contribuir para a sociedade, o Instituto Federal de Ciências e Tecnologias desenvolveu um projeto de caráter social para atender a essas necessidades de uma parcela tão grande da população.

O IF surgiu como instituição federal no ano de 2007, e desde então é um dos polos de educação da região, onde se desenvolve diversas pesquisas e projetos [4], além de oferecer Cursos Técnicos, Bacharelados, Engenharias, Licenciaturas, Pós-Graduação e Tecnologias. Um dos fatores que diferencia o Instituto Federal das demais instituições de ensino da região é a preocupação com a formação pessoal do aluno, através de atividades extracurriculares que ajudam a olhar para a realidade de maneiras diferentes. [5]

A fim de melhorar o desempenho dos Lares de Idosos através da inserção de tecnologias para agilizar os processos, os alunos do 4º ano do Curso Técnico Integrado em Informática, do Instituto Federal Campus São João da Boa Vista, propuseram a criação de um software que conseguisse desempenhar essas funções, que é desenvolvido na disciplina de Projeto de Desenvolvimento de Sistemas. Os mesmos tomaram como um desafio de conclusão de curso criar essa aplicação. [6]

O projeto chamado Gerações é gerido por dois professores, que auxiliam e direcionam o andamento do desenvolvimento. A dimensão dos trabalhos é vasta, logo foi necessário dividi-lo em nove módulos, cada um com suas especificações. Os alunos formam uma equipe de 57 pessoas, dividam entre os nove módulos. Os módulos de agrupam de maneira geral em 3 grandes grupos: Acesso Externo, Gerenciamento dos Idosos e Controle Remoto.

Abaixo encontra-se uma série de imagem que exemplifica as funções básicas desenvolvidas por cada módulo dentro do projeto:



Figura 1 - Especificação dos módulos da divisão de Acesso Externo

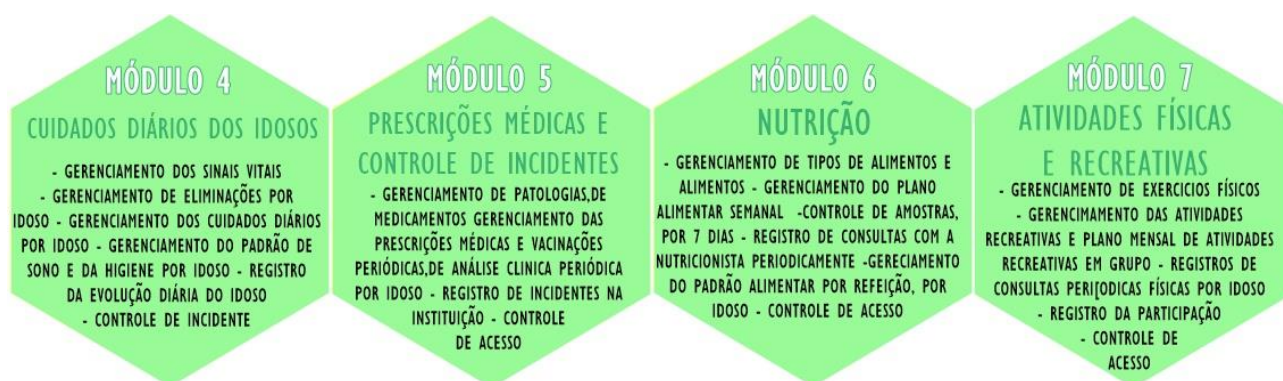


Figura 2 - Especificação dos módulos da divisão de Gerenciamento dos Idosos

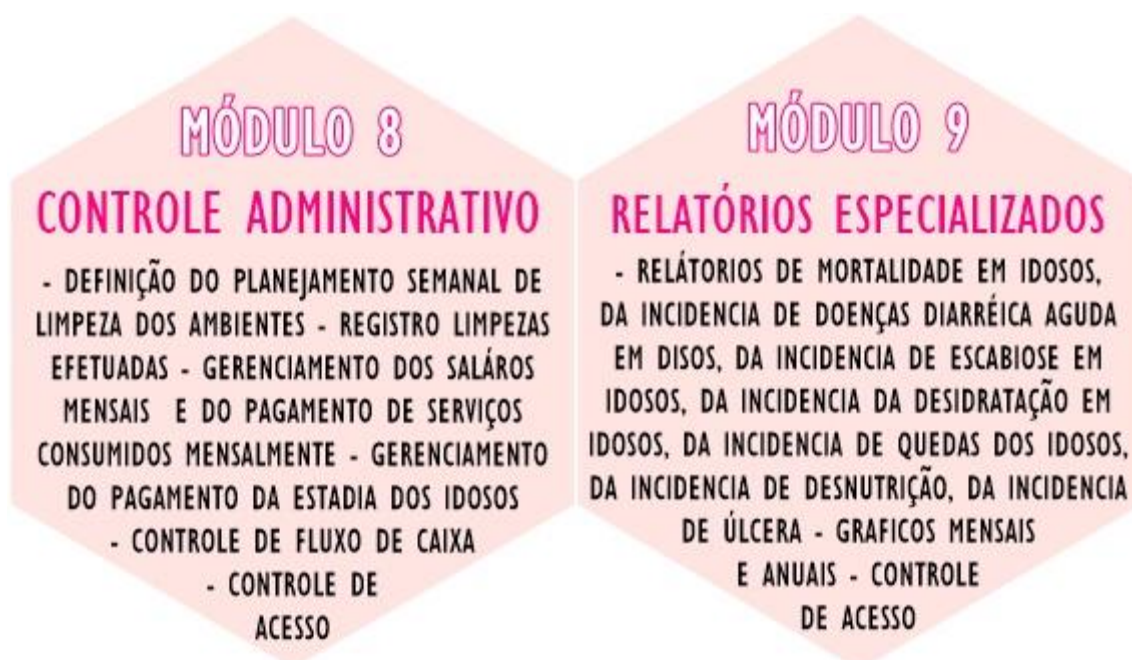


Figura 3 - Especificação dos módulos da divisão de Controle Gerencial

A meta é desenvolver algo que traga mais rapidez para a organização dos Idosos, além de contribuir para um melhor desempenho das funções e maior segurança e confiabilidade, visto que todos os dados estarão salvos em caso de consulta.

As especificações dos módulos vão desde documentação, prototipação, codificação, desenvolvimento de Banco de Dados e aplicação das funcionalidades. Dentre todos os ramos do projeto, este trabalho irá tomar como foco o desenvolvimento do Banco de Dados feito pelo Módulo de Prescrições Médicas e Controle de Incidentes.

## **1.2 Objetivo Geral da Pesquisa**

O propósito deste documento é mostrar como foi feito o Banco de Dados do Módulo de Prescrições Médicas e Controle de Incidentes, iniciando com os Casos de Uso, e posteriormente a aplicação dos mesmos para criação dos Modelos de Banco, sendo eles o Modelo Conceitual, Modelo Lógico e Modelo Físico.

## **1.3 Objetivos Específicos**

Os objetivos gerais dessa pesquisa são:

- Conceituar termos relacionados ao Banco de Dados;
- Apresentar os Casos de Uso, a fim de direcionar a elaboração dos Modelo Conceitual;
- Explicar como foi passado os Casos de Uso para o Modelo Conceitual;
- Exibir o Modelo Conceitual do Módulo de Prescrições Médicas e Controle de Incidentes e explicar suas relações;
- Exibir o Modelo Lógico, gerado a partir do Modelo Conceitual e especificar suas particularidades;
- Exibir o Modelo Físico, gerado em formato SQL.

## **1.4 Estrutura do Documento**

O capítulo 1 abordará a contextualização de todo o Projeto, falando sobre a cidade, o impacto de todo o trabalho da equipe para a população sanjoanense.

O capítulo 2 desenvolverá o tema proposto, analisando as particularidades encontradas nos Modelos Entidade-Relacionamento, Modelo Lógico e Modelo Físico, próprias do Módulo de Prescrições Médicas e Controle de Incidentes.

O capítulo 3 englobará todas as conclusões tiradas a partir da análise apresentada no capítulo 2, buscando reforçar os principais pontos.

O capítulo 4 irá referenciar todas as fontes usadas como base para o desenvolvimento desse trabalho.



## 2 Desenvolvimento

### 2.1 Levantamento Bibliográfico

Este capítulo foi baseado nos livros referenciados no Capítulo 4, correspondendo aos itens [8], [9] e [10]

#### 2.1.1 SBGD (Sistema Gerenciador de Bando de Dados)

Significa de maneira simples um software que inclui as funções de definição, recuperação e alteração de dados que estão inseridos em um determinado bando de dados.

#### 2.1.2 Modelo Conceitual

##### 2.1.2.1 Modelo Conceitual

Representa a realidade do ambiente do problema, constituindo uma visão global dos dados e dos relacionamentos (macrodefinição), sem se preocupar como estes serão armazenados a nível SGBD. É representado por um diagrama chamado diagrama entidade-relacionamento.

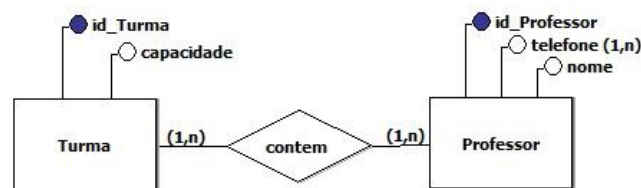


Figura 4 - Exemplo de Diagrama Entidade-Relacionamento

##### 2.1.2.2 Diagrama Entidade-Relacionamento (DER)

Definido por Peter Chen em 1976 e teve como base a teoria relacional criada por E. F. Codd (1970). Segundo ele, a visão de uma dada realidade baseia-se no relacionamento entre entidades, o qual retrata os fatos que governam essa mesma realidade, e cada um pode possuir atributos.

##### 2.1.2.3 Entidade

É definida como um objeto pertencente ao mundo real, com uma identificação distinta e significado próprio. Sua representação se dá através de uma Classe de dados, formando um conjunto de dados com as mesmas características.

Representa um conjunto de objetos da realidade modelada sobre os quais deseja-se manter informações no banco de dados.

Para identificar uma entidade, faz-se uma abstração da realidade em nível macro de informações. Isso é encontrado vendo-se os objetos presentes no ambiente. Normalmente, o nome de uma entidade será um substantivo. Sua representação no Modelo Entidade-Relacionamento é feita através de um retângulo com o nome da mesma no interior. Ela representa uma tabela de dados, onde cada linha dessa tabela representará uma instância.

#### **2.1.2.4 Atributos**

Todo objeto, para ser uma entidade, precisa ter propriedades que descrevem e dão qualidades a ele. Essas propriedades são chamadas de atributos, que conterão valores específicos para cada instancia da entidade.

Atributos são dados associados a cada ocorrência de uma entidade.

Podem ser classificados como:

*Atributo Simples*: receberá um valor único e não é o identificador.

*Atributo Composto*: é formado por vários valores menores que o compõem.

*Atributo Identificador (ou determinante)*: identifica de forma única uma instancia da entidade, não podendo haver repetições.

#### **2.1.2.5 Entidade Associativa**

Uma entidade associativa é na verdade uma redefinição de um relacionamento, que passa a ser tratado como se fosse uma entidade.

#### **2.1.2.6 Relacionamento**

Fato que associa dois objetos, dão sentido a existência deles e mostra suas inter-relações. Relacionamentos são representados através de um verbo que contextualizará a relação daquelas duas ou mais entidades.

No Modelo Entidade-Relacionamento, é mostrado através de um losango entre as duas entidades, e conectado em ambas.

#### **2.1.2.7 Grau de Relacionamento**

Diz respeito a quantidade de ocorrências da relação entre duas entidades. Deve-se lembrar que sempre é visto em ambos os sentidos, pois os relacionamentos não são unidirecionais. Podem ser divididas em 3 tipos de relacionamentos:

*Relacionamento de 1:1* – cada elemento da entidade pode se relacionar com somente um elemento da outra entidade.

*Relacionamento de 1: muitos (n)* - a entidade 1 se relaciona com somente um elemento da entidade 2, enquanto a entidade 2 se relaciona com vários elementos da entidade 1.

*Relacionamento muitos (n): muitos (n)* - ambas as entidades se relacionam com vários elementos em relação a outra. Esse tipo de relacionamento possui atributos, pois possuem dados inerentes ao fato e não as entidades.

#### **2.1.2.8 Valor Nulo**

Quando não há nada dentro de um campo. Não é obrigatório ser informado, pois se caso não seja informado nenhum dado, ele automaticamente atribui o valor nulo a campo.

### **2.1.3 Modelo Lógico Relacional**

#### **2.1.3.1 Modelo Lógico**

É gerado a partir do Modelo Conceitual. Diferente do anterior, depende do tipo de SGBD que está sendo usado, já que o Bando de Dados é descrito no nível de abstração pelo usuário de SGBD.

#### **2.1.3.2 12 Regras de Codd**

Para se criar o Modelo Lógico Relacional foi estabelecida 12 regras:

1. Toda informação num Bando de Dados Relacional é apresentada em nível lógico por valores em tabelas.
2. Todo dado em um bando de dados relacional tem garantia de ser logicamente acessível, recorrendo-se a uma combinação do nome da tabela, um valor de chave e o nome da coluna.
3. Tratamento sistemático de valores nulos (ausência de dados).
4. O dicionário de dados relacional ativo é baseado no modelo relacional
5. O SGBD relacional deve ter uma linguagem para definição, detalhamento e manipulação de dados.
6. Tratamento da atualização e visão dos dados
7. Tratamento de alto nível para inserção, atualização e exclusão de dados
8. Independência dos dados físicos (mudança na memória e no método de acesso)
9. Independência de dados lógicos (mudanças de qualquer tipo as tabelas básicas, por exemplo, divisão de uma tabela por linha ou coluna)
10. Independência das restrições de integridade
11. Independência de distribuição

12. Não subversão das regras de integridade ou restrições quando se utiliza uma linguagem de baixo nível.

#### **2.1.3.3 Chave Primária**

Atributo que identifica de maneira única um elemento da tabela. Normalmente assume um valor inteiro, para não haver dualidade.

#### **2.1.3.4 Chave Estrangeira**

São o elo entre as tabelas. Quando estão conectadas, a chave primária da tabela muitos começa a fazer parte da tabela com cardinalidade 1, tornando-se assim uma chave estrangeira nessa tabela. Isso forma uma ligação lógica entre as duas tabelas, onde uma dependerá de informações da outra.

#### **2.1.3.5 Características**

Uma tabela pode ser acessada através de qualquer campo, sendo ele uma chave ou não.

O relacionamento entre as tabelas não existe fisicamente, pois é um conceito lógico representado através das chaves estrangeiras.

#### **2.1.3.6 Tabela**

A tabela é um conjunto de linhas que contém uma série de campos. O conjunto de campos alinhados de todas as linhas correspondem a colunas.

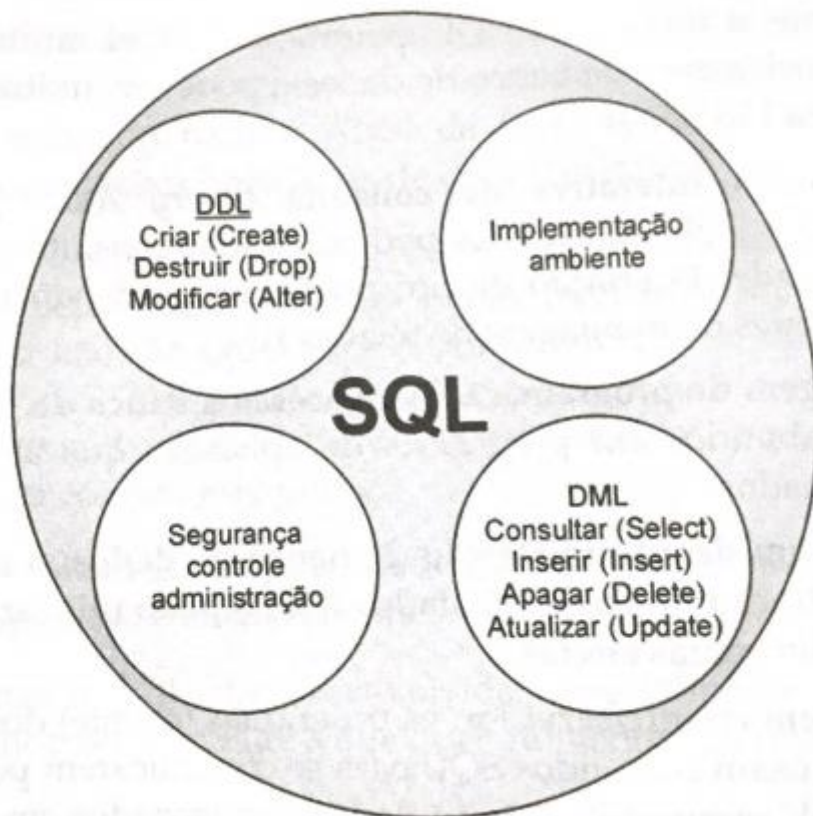
### **2.1.4 Modelo Físico**

#### **2.1.4.1 Modelo Físico**

Gerado a partir do Modelo Lógico. Nele será especificado as estruturas físicas de armazenamento de dados, como tamanho de campos, índices, tipo de variável, nomenclaturas etc. Segue-se os requisitos de processamento. Detalha os métodos de acesso ao SGBD para os dados descritos nos Modelo Conceitual e Modelo Lógico. Aqui será usada a Linguagem de Definição de Dados SGBD (DDL) para sua montagem.

#### **2.1.4.2 Linguagem SQL**

SQL significa Structured Query Language – Linguagem Estruturada de Pesquisa. É uma linguagem desenvolvida especialmente para ambientes relacionais. Pode ser utilizada em linguagens hospedeiras, como Java, PHP, Cobol, C, dentre outras.



**Figura 5 - Estrutura de funções SQL**

*Definição de Dados (DDL):* estruturação e organização dos dados armazenados e das relações entre eles.

*Manipulação de Dados (DML):* inclusão, exclusão, seleção e atualização de dados armazenados.

*Controle de acesso:* proteção dos dados armazenados.

*Integração dos dados:* auxílio no processo de integridade de dados, evitando corrupções, inconsistências e falhas no sistema.

#### **2.1.4.3 Variáveis**

É o atributo vindo dos modelos anteriores, no formato SQL, podendo ser manipulado de acordo com a necessidade do usuário.

## 2.2 Etapas para o Desenvolvimento da Pesquisa

### 2.2.1 Casos de Uso Documento de Casos de Uso

Os Casos de Uso fazem parte de um dos primeiros passos da criação do Bando de Dados. Eles surgem a partir das especificações dos requisitos, e é eles que direcionam as necessidades de armazenamento do Banco.

No Diagrama de Casos de Uso do Módulo de Prescrições Médicas e Controle de Incidentes é possível perceber as funções que o sistema irá realizar. Abaixo encontra-se o Diagrama integralmente.

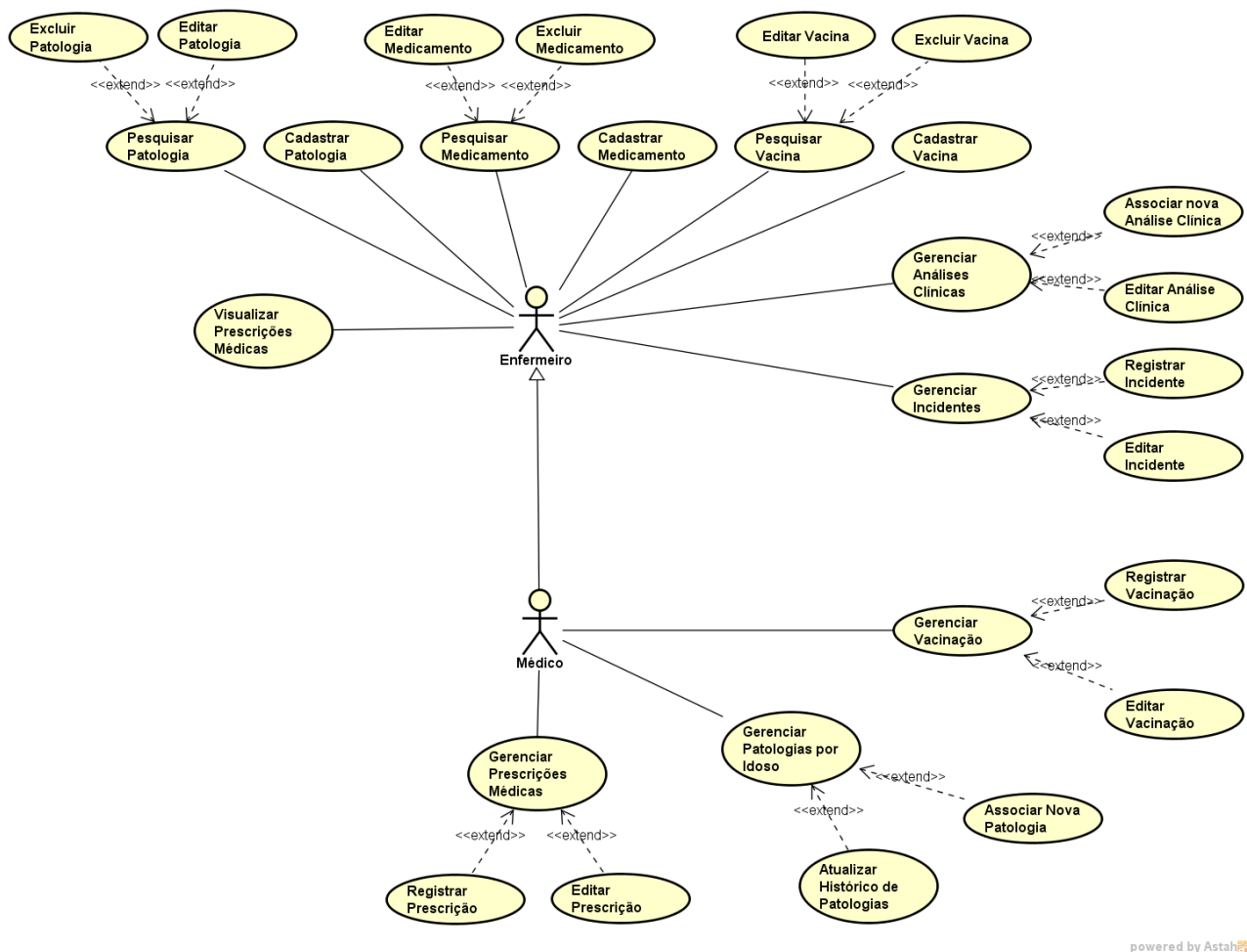


Figura 6 - Diagrama de Casos de Uso

Analisando as relações entre os Casos de Uso e entre os atores, tem-se principalmente a estrutura de um CRUD (em inglês, Creat, Read, Update e Delete), sendo possível a criação, exclusão, visualização e edição das informações:

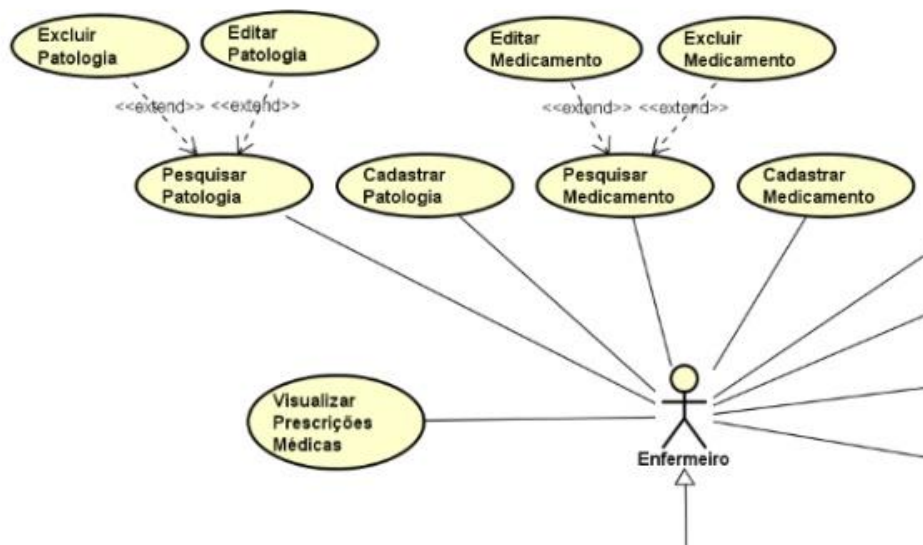


Figura 7 - Recorte 1 do Diagrama de Casos de Uso

O enfermeiro é responsável por Pesquisar e Cadastrar Patologias. Há uma dependência das ações quando vemos excluir e o editar, que só serão possíveis de acontecer depois de já ter pesquisado a patologia.

O mesmo se aplica a Medicamentos, onde é de responsabilidade do enfermeiro Cadastrar-lo e cuidar de sua exclusão e edição.

Uma ação que também é possível ao enfermeiro é a de visualização das Prescrições Médicas feitas pelo Médico. É importante ressaltar que nesta etapa, o enfermeiro só é capaz de visualizá-las, e não de fazer, excluir ou editar elas.

Ao ver o lado direito dos diagramas, temos a mesma linha de raciocínio, aplicada a outras funções:

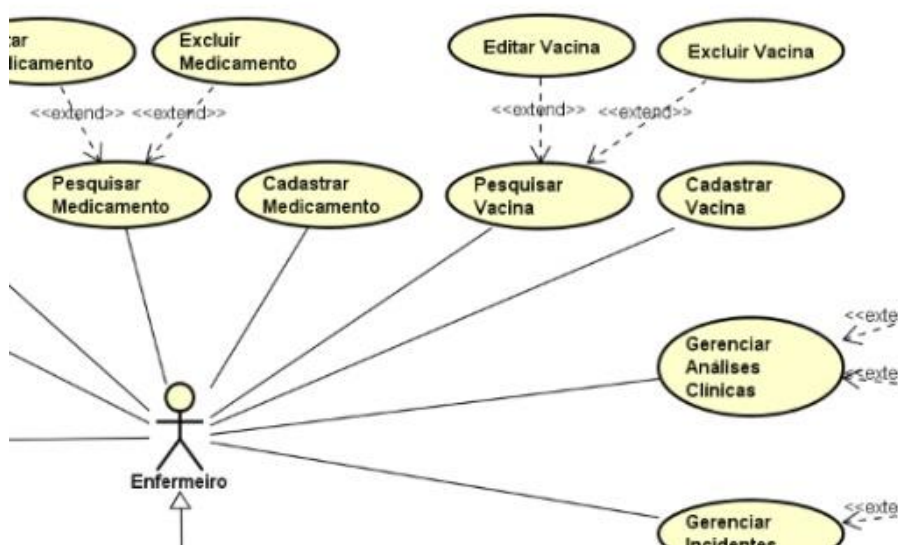
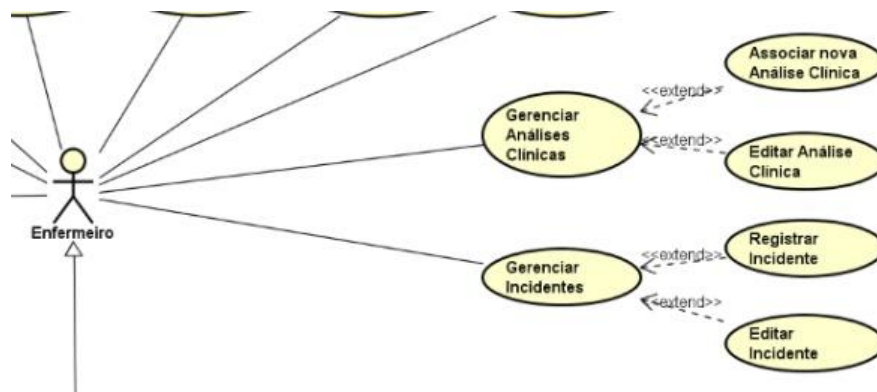


Figura 8 - Recorte 2 do Diagrama de Casos de Uso

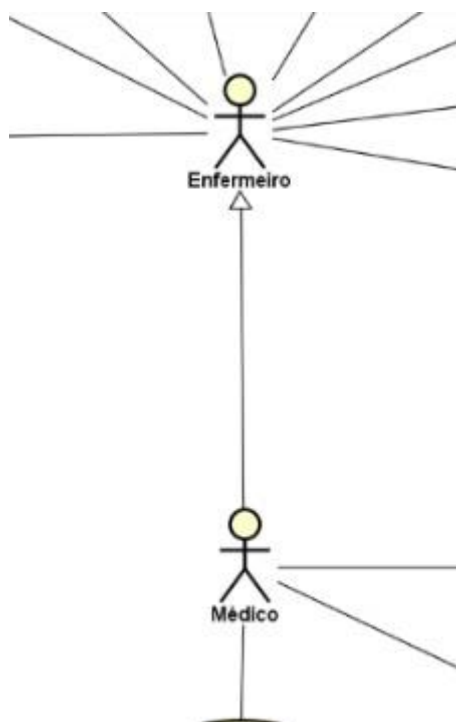
O Enfermeiro cadastra as vacinas, e tem-se uma dependência para excluí-las e editá-las, pois é preciso pesquisá-las antes.



**Figura 9 - Recorte 3 do Diagrama de Casos de Uso**

Neste pedaço do Diagrama o CRUD continua fazendo parte. Tanto para Gerenciar Análises Clínicas, quanto para Gerenciar Incidentes, é possível fazer sua exclusão e edição.

Uma interação relevante, é a ligação dos dois atores, mostrando que tudo o enfermeiro faz, médico pode fazer também, mas tudo que está ligado ao ator Médico é exclusivo dele.



**Figura 10 - Recorte 4 do Diagrama de Casos de Uso**

A partir deste momento, muda-se o ator, os Casos de Uso a seguir só são permitidas ao médico:



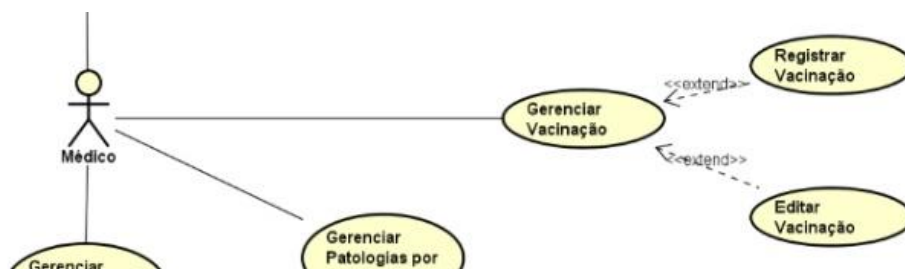


Figura 11 - Recorte 5 do Diagrama de Casos de Uso

O médico poderá Registrar Vacinação e editá-la, com isso ele obrigatoriamente estará Gerenciando uma Vacinação ao idoso.

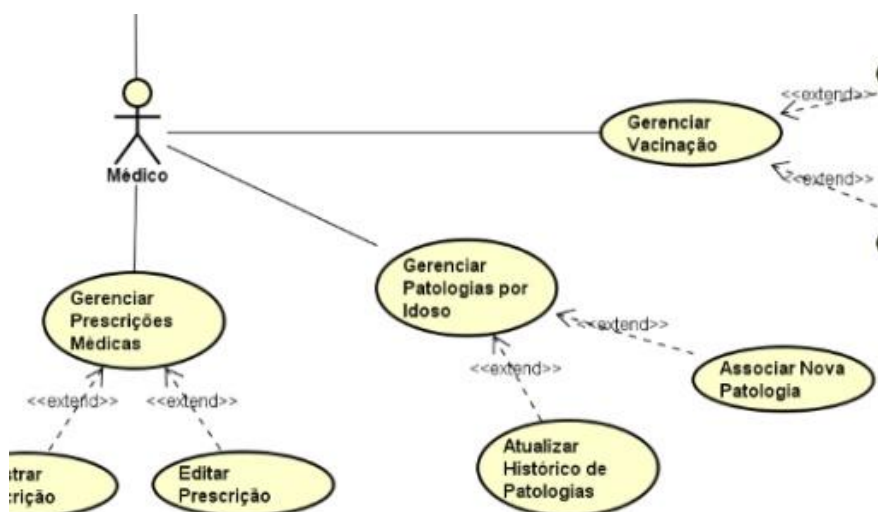


Figura 12 - Recorte 6 do Diagrama de Casos de Uso

Assim como o anterior, o médico terá acesso a atualizar o histórico de patologias do Idoso, além de associá-lo a uma nova, e ao fazer isso, estará obrigatoriamente gerenciando as patologias relacionadas a aquele idoso.

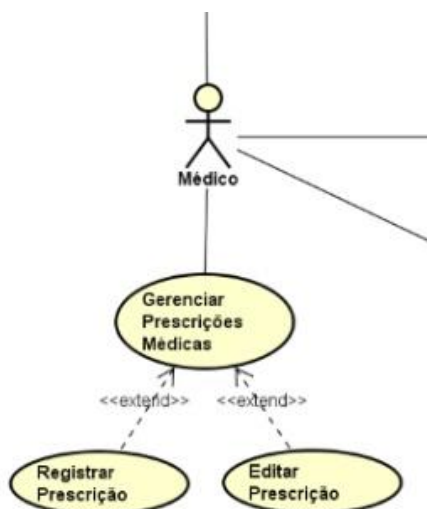


Figura 13 - Recorte 7 do Diagrama de Casos de Uso

Para finalizar, o médico pode registrar uma prescrição e editá-la, e ao realizar essas funções, estará gerenciando as prescrições médicas do idoso.

### 2.2.2 Passagem dos Casos de Uso para o Modelo Conceitual

A fim de modelar o Bando de Dados, foi usado os Casos de Uso como meio de início. Como mostrado no tópico anterior, através dele é possível identificar as necessidades do sistema. Ao completá-lo, é preciso pensar sobre o que será preciso guardar dentro de uma base de dados, quais são as entidades, ou seja, aquilo que carrega consigo qualidades e ações.

A partir desse pensamento, foi feita outra análise, e a transformação foi, muitas vezes, óbvia, como no exemplo abaixo:

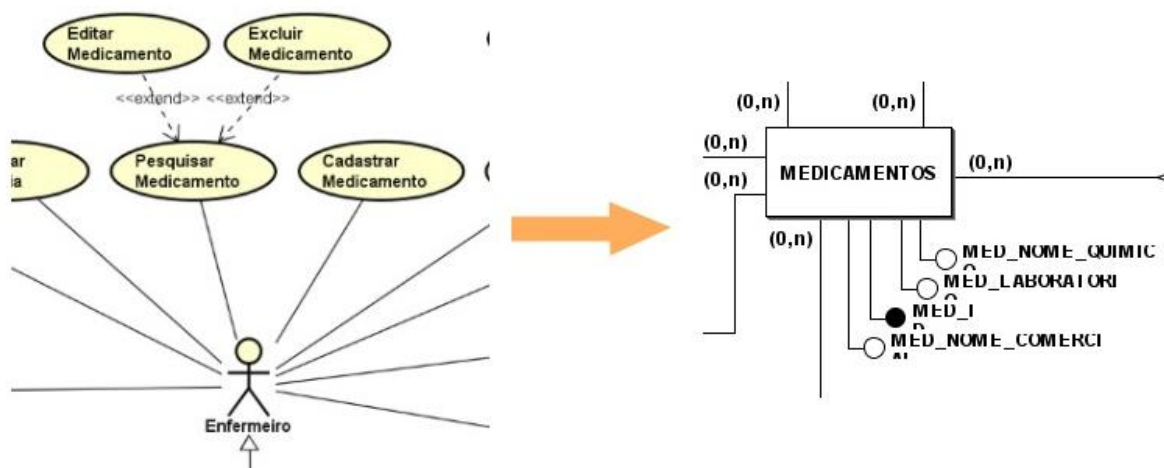


Figura 14 - Exemplo da passagem dos Casos de Uso para o Modelo Conceitual

Aqui é possível observar a necessidade de manipular informações sobre medicamentos. Para isso, foi criada uma entidade que tem por função armazenar os dados sobre os medicamentos. Há tabelas conectadas a ela, como mostra a figura abaixo:

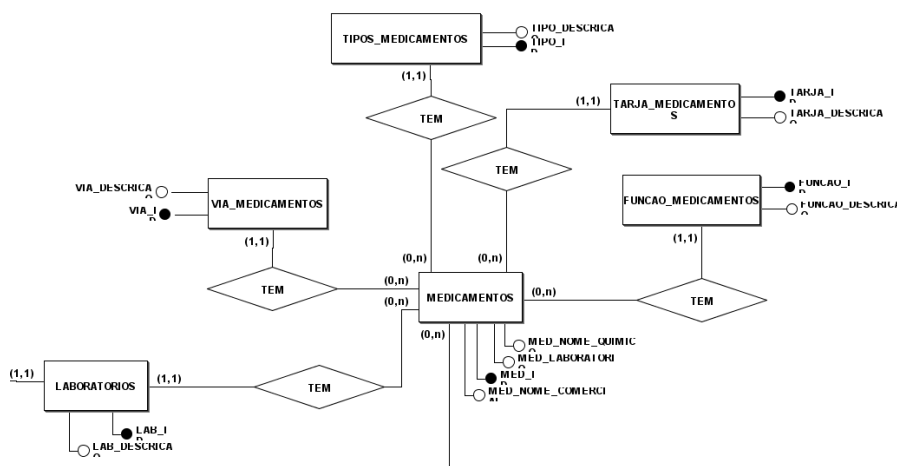


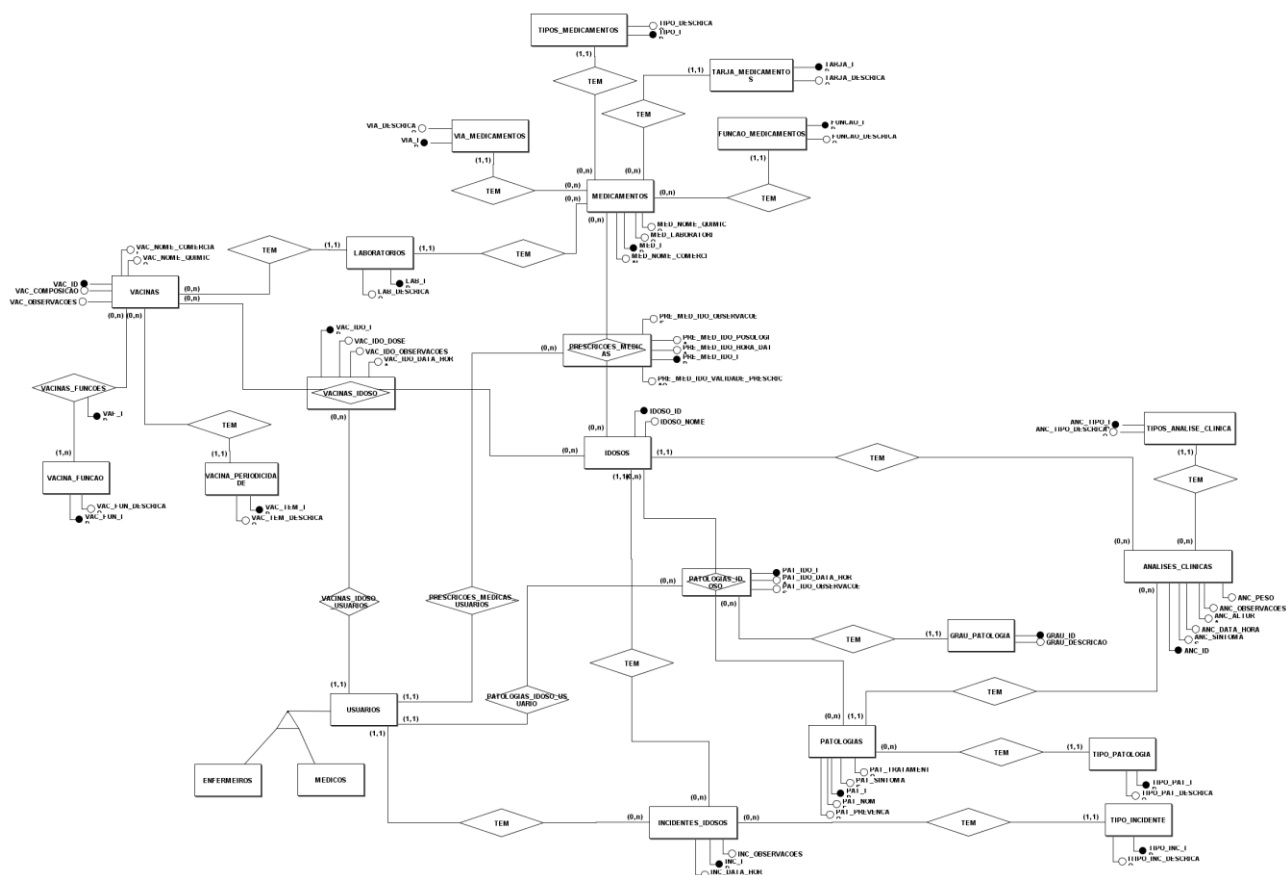
Figura 15 - Exemplo de entidades construtivas para o Modelo Conceitual

Essas entidades (LABORATORIOS, VIA\_MEDICAMENTOS,

Esse processo foi feito com todos os Casos de Uso, visualizando quais seriam as necessidades azenados para que fosse necessário executa-los.

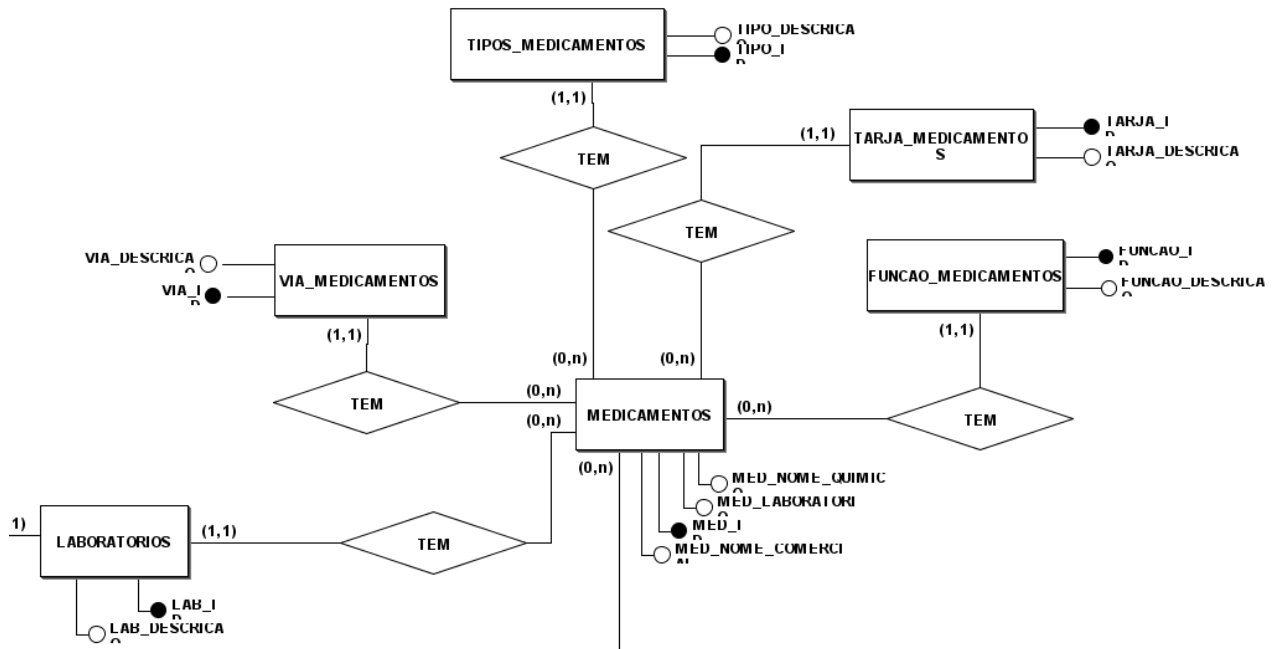
### 2.2.3 Modelo Conceitual

Nesta etapa, o Banco começa a ser modelado definitivamente. O Diagrama Entidade-Relacionamento do Módulo de Prescrições Médicas e Controle de Incidentes conta com 20 entidades, 3 entidades associativas e 19 relacionamentos. O mesmo pode ser visto na íntegra pela imagem abaixo:



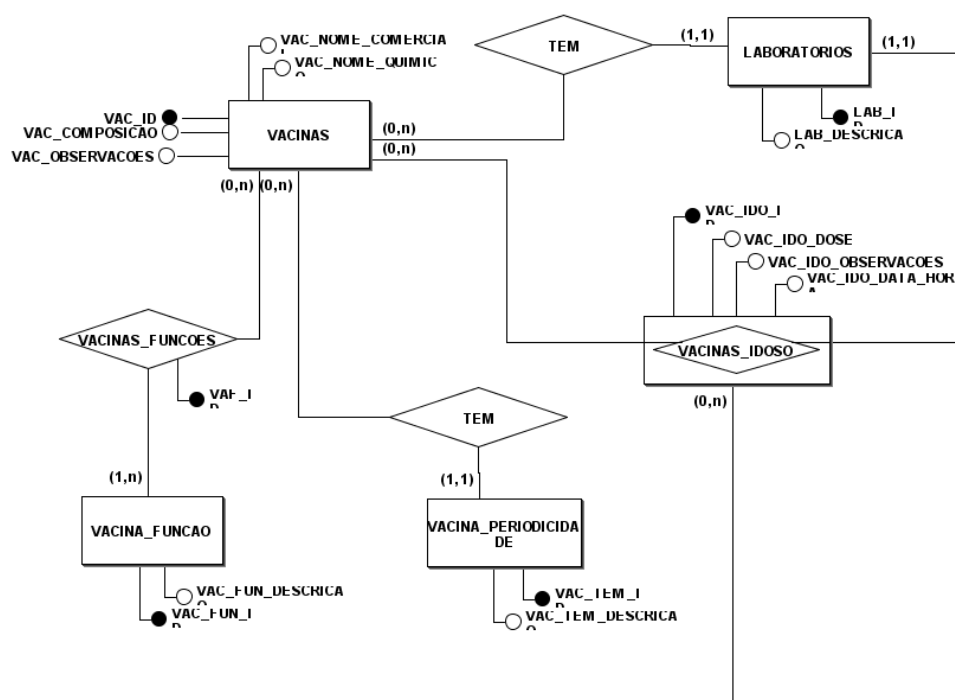
**Figura 16 - Diagrama Entidade-Relacionamento**

O diagrama mostra como as entidades interagem entre si, mas não especifica o fluxo dos dados. Todos os atributos ID's das entidades são os identificadores. Analisando de maneira mais profunda ele, temos que:



**Figura 17 - Recorte 1 do Diagrama Entidade-Relacionamento**

Neste recorte do diagrama, temos a entidade **MEDICAMENTOS**, que tem seus atributos (**MED\_NOME\_COMERCIAL**, **MED\_NOME\_QUIMICO**, **MED\_LABORATORIO** e **MED\_ID**). Esta entidade está ligada com **VIA\_MEDICAMENTOS** (que possui como atributos **VIA\_DESCRICA** e **VIA\_ID**), já que um medicamento pode ser tomado somente por uma via, mas uma via pode estar associada a nenhum ou vários medicamentos. O mesmo acontece com as demais entidades mostradas no recorte, sendo **TIPO\_MEDICAMENTO** (com atributos de **TIPO\_DESCRICA** e **TIPO\_ID**), onde será armazenado dados como analgésico, antidepressivo, antibiótico, dentre outros de acordo com as necessidades, **FUNCAO\_MEDICAMENTOS** (com atributos **FUNCAO\_DESCRICA** e **FUNCAO\_ID**), e por último **LABORATORIOS** (contendo os atributos **LAB\_DESCRICA** e **LAB\_ID**).



**Figura 18 - Recorte 2 do Diagrama Entidade-Relacionamento**

Neste segundo recorte do diagrama, temos as ligações da entidade VACINAS (com atributos VAC\_NOME\_COMERCIAL, VAC\_NOME\_QUIMICO, VAC\_ID, VAC\_COMPOSICAO e VAC\_OBSERVACOES), que se liga com LABORATORIOS, onde vacina pode ter somente um laboratório, e um laboratório pode não estar associado a nenhuma vacina ou a várias. Seguindo o mesmo pensamento, VACINAS conecta-se com VACINA\_PERIODICIDADE (sendo seus atributos VAC\_TEMP\_ID e VAC\_TEM\_DESCRICAÇÃO).

Com a entidade VACINA\_FUNCAO (atributos sendo VAC\_FUN\_DESCRICAÇÃO e VAC\_FUN\_ID), a cardinalidade é diferente das demais até aqui. Nesta relação, uma vacina deve ter no mínimo uma função, mas também pode ter várias funções, e uma função pode não estar associada a nenhuma vacina ou a várias. Outra novidade nesta relação (VACINAS\_FUNCOES) é a presença de um atributo nela, pois como se trata de uma relação n:n, resultando em outra tabela no banco de dados, logo sendo necessário a presença de um atributo identificador, no caso VAF\_ID. Todas as relações n:n serão melhor especificadas no tópico seguinte.

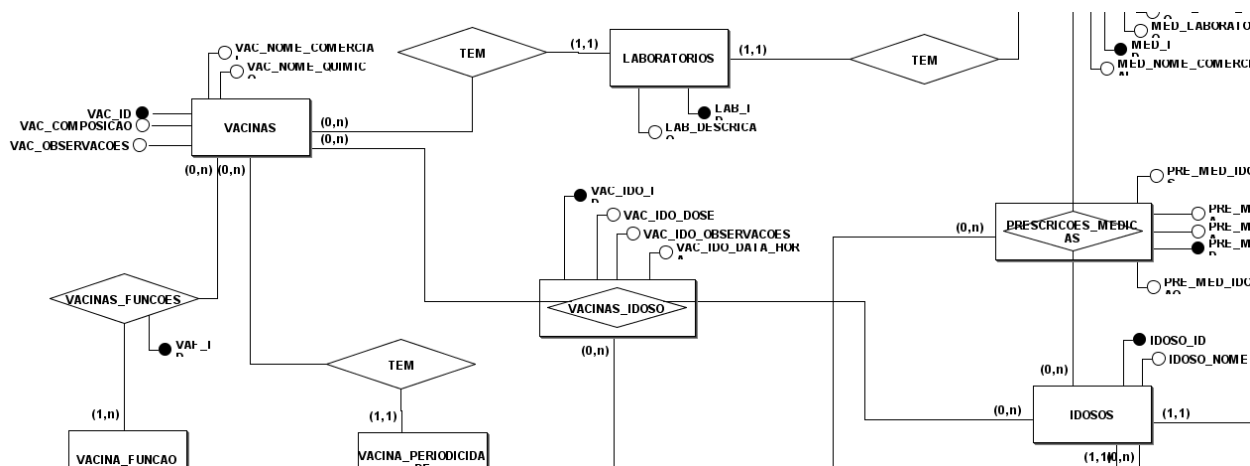


Figura 19 - Recorte 3 do Diagrama Entidade-Relacionamento

Aqui tem-se outra novidade. No Diagrama conta com algumas entidades associativas. A ligação entre VACINAS e IDOSOS (a entidade IDOSO não foi de responsabilidade do módulo de Prescrições Médicas e Controle de Incidentes, por esse motivo, seus atributos são restritos a IDOSO\_NOME e IDOSO\_ID, somente com a intenção de representar a inserção de dados) gerou uma entidade associativa, pois apesar de ser uma relação, ela possuía características específicas e atributos, como VAC\_IDO\_ID, VAC\_IDO\_DOSE, VAC\_IDO\_OBSERVACOES e VAC\_IDO\_DATA\_HORA.

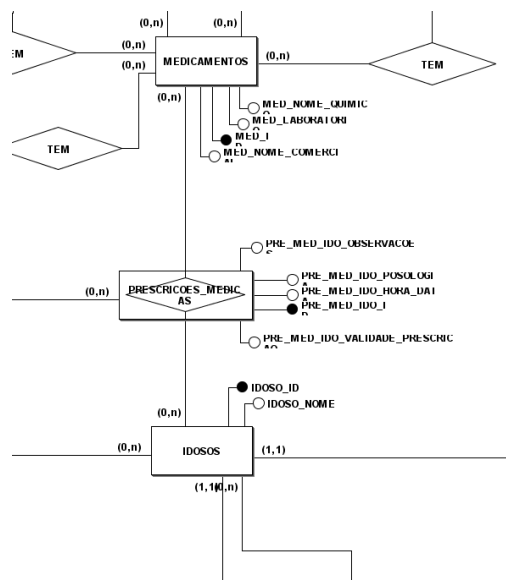
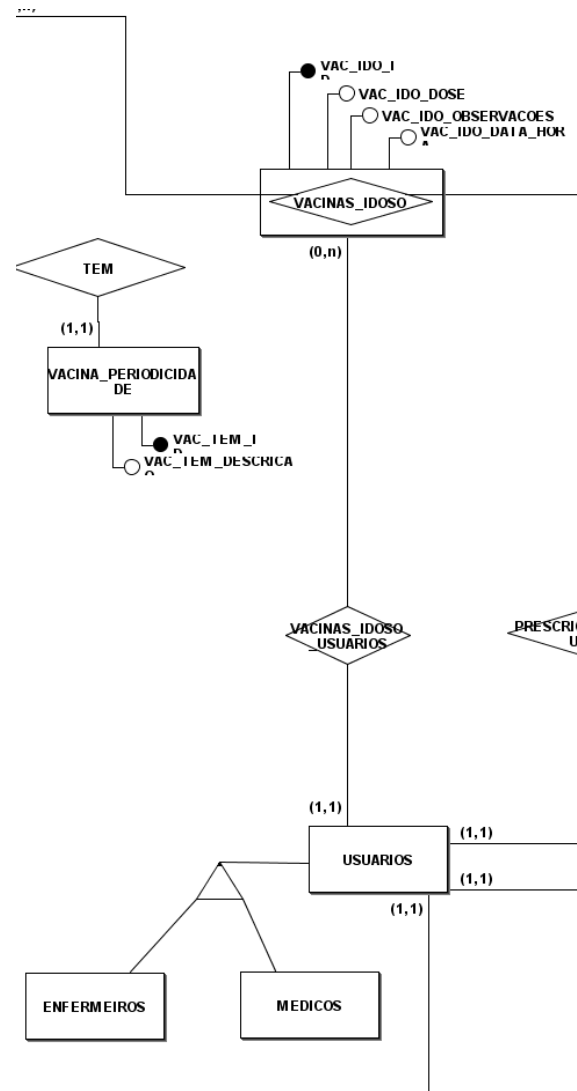


Figura 20 - Recorte 4 do Diagrama Entidade-Relacionamento

O mesmo mostrado na figura anterior acontece com este caso. A relação entre IDOSOS e MEDICAMENTOS gerou uma entidade associativa chamada PRESCRICOES\_MEDICAS, com atributos PRE\_MED\_IDO\_OOBSERVACOES, PRE\_MED\_IDO\_POSOLOGIA, PRE\_MED\_IDO\_HORA\_DATA, PRE\_MED\_IDO\_ID e

PRE\_MED\_IDO\_VALIDADE\_PRESCRICAO. Idosos podem estar associados a nenhum ou a vários medicamentos, enquanto um medicamento pode não estar associado a nenhum idoso ou a vários.



**Figura 21 - Recorte 5 do Diagrama Entidade-Relacionamento**

A entidade associativa VACINAS\_IDOSO tem também uma ligação com a tabela USUARIOS, a fim de manter salvo quem foi o responsável por receitar e aplicar a vacina no idoso. A tabela USUARIOS não era de responsabilidade do módulo de Prescrições Médicas e Controle de Incidentes, mas para representação foi colocada no diagrama por ser uma parte fundamental no funcionamento de armazenamento dos dados. Na entidade USUARIOS há também uma especialização, onde as entidades ENFERMEIROS e MEDICOS herdam tudo que a entidade USUARIOS contém. A vacina do idoso deve ser de responsabilidade de somente um usuário, mas um usuário pode ter como responsabilidade uma ou mais vacinas.

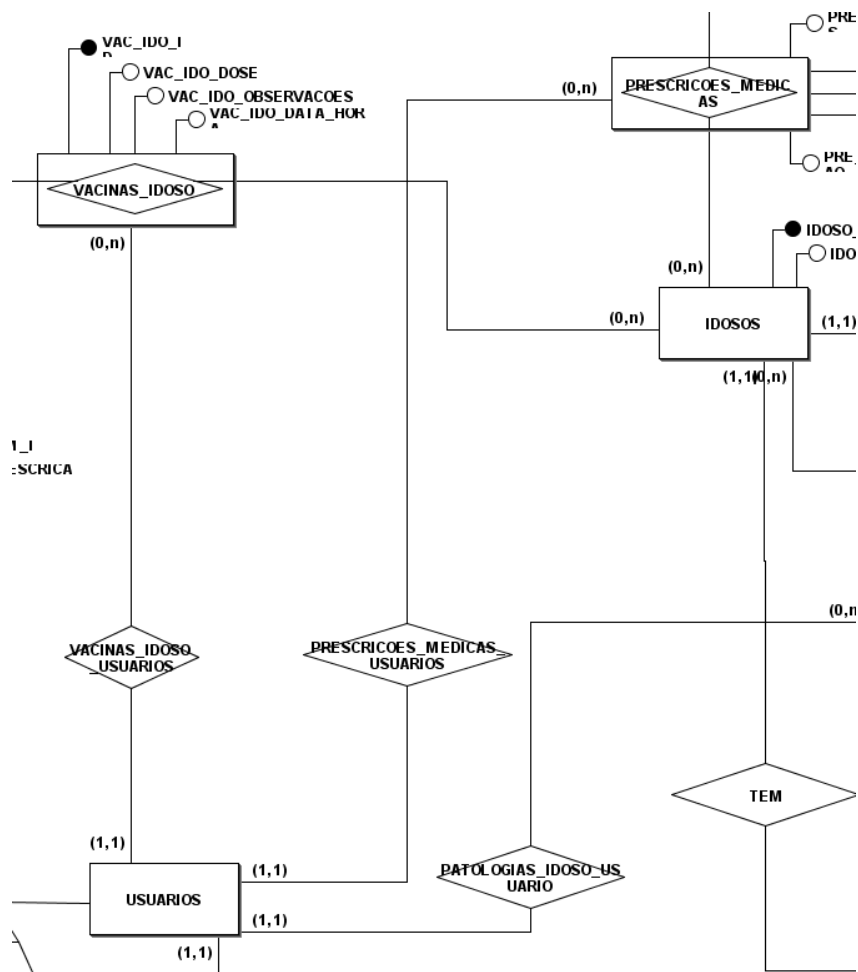


Figura 22 - Recorte 6 do Diagrama Entidade-Relacionamento

Ainda ligado a tabela USUARIOS, há a entidade associativa PRESCRICOES\_MEDICAS, com a mesma finalidade de registrar qual usuário prescreveu os medicamentos. Assim, um usuário pode não estar associado a nenhuma prescrição médica ou a várias, e uma prescrição médica tem que estar associado a somente um usuário.

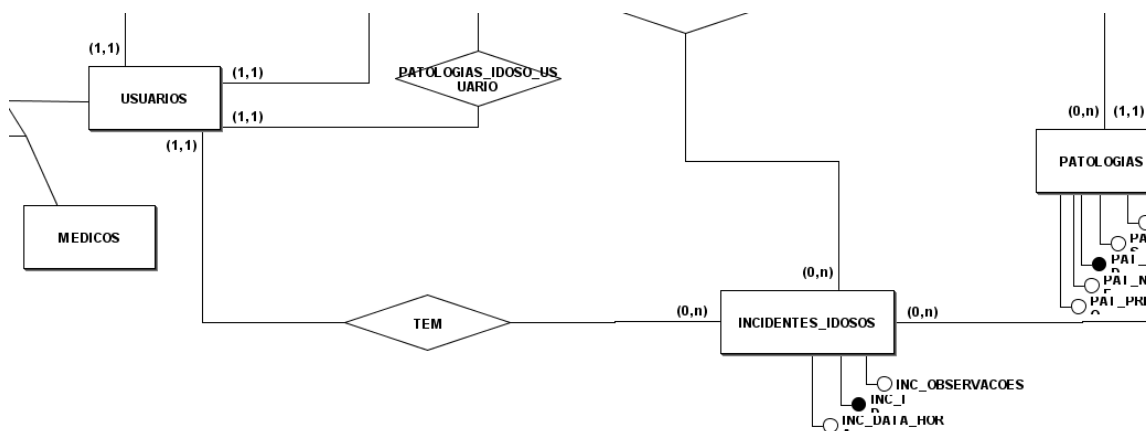


Figura 23 - Recorte 7 do Diagrama Entidade-Relacionamento



USUARIOS também está ligado a INCIDENTES\_IDOSOS (que conta com atributos INC\_OBSERVACOES, INC\_ID e INC\_DATA\_HORA). Nesta ligação, um incidente deve estar associado a somente um usuário, e um usuário pode estar associado a nenhum ou vários incidentes.

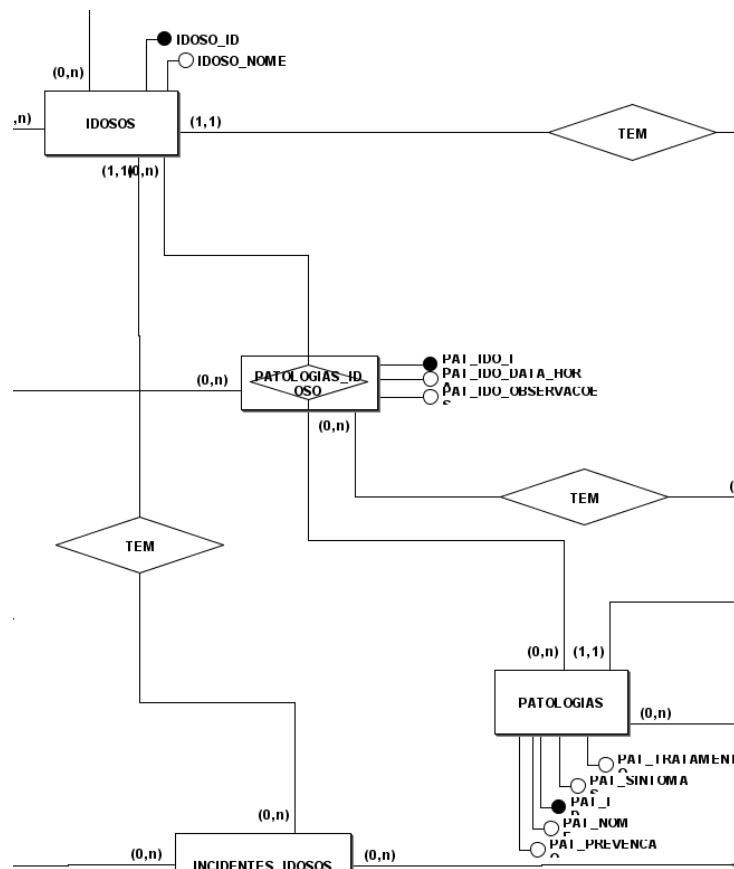


Figura 24 - Recorte 8 do Diagrama Entidade-Relacionamento

A relação entre IDOSOS e PATOLOGIAS (que tem como atributos PAT\_TRATAMENTO, PAT\_SINTOMAS, PAT\_ID, PAT\_NOME e PAT\_PREVENCAO) gerou uma entidade associativa também, já que possuía características próprias. A associação chama-se PATOLOGIAS\_IDOSO, e há como atributos PAT\_IDO\_ID, PAT\_IDO\_DATA\_HORA e PAT\_IDO\_OBSERVACOES. Uma patologia pode estar associada a nenhum ou vários idosos, enquanto um idoso por não estar associado a nenhuma patologia ou a várias.

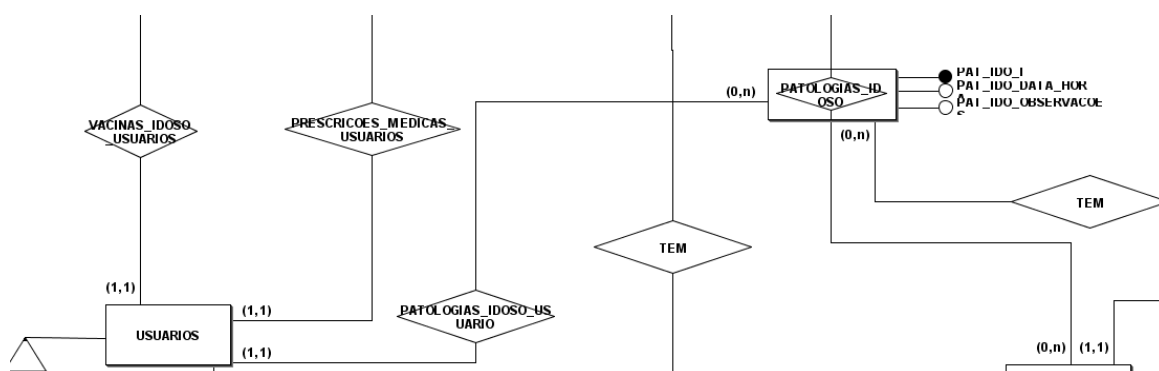
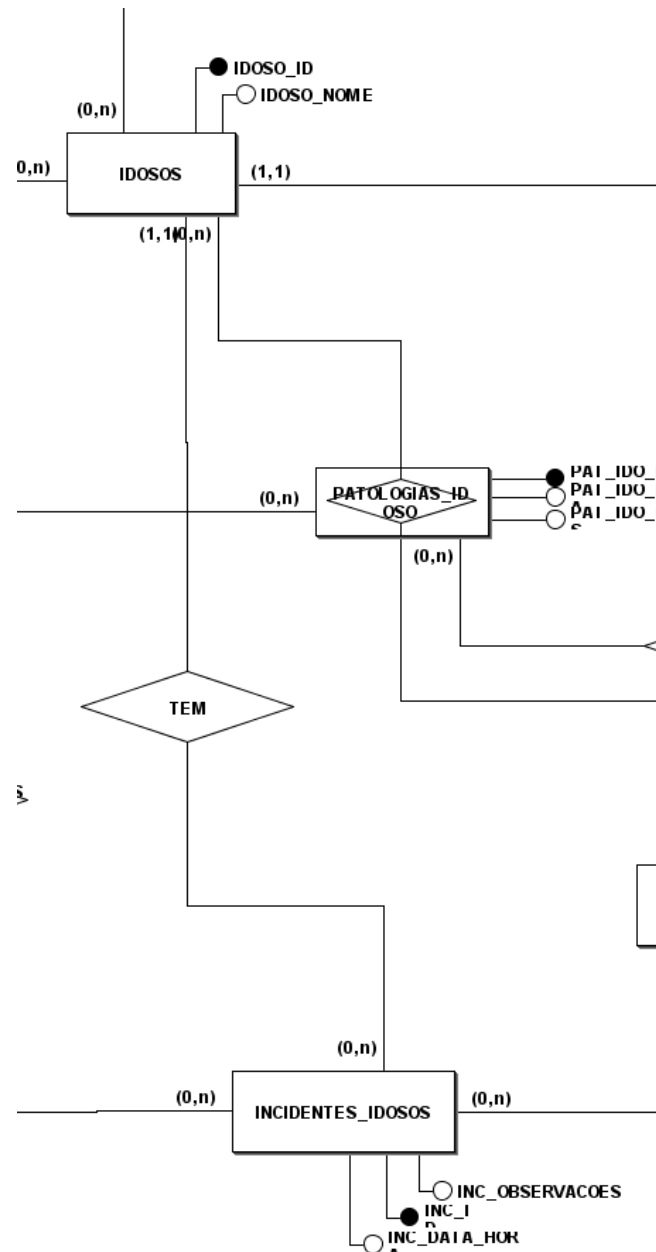


Figura 25 - Recorte 9 do Diagrama Entidade-Relacionamento

Com a mesma finalidade das demais, a ligação entre USUARIOS e PATOLOGIAS\_IDOSO busca armazenar o responsável por diagnosticar as patologias e associá-las ao idoso. A entidade PATOLOGIA\_IDOSO. Um usuário pode não estar associado a nenhuma a várias patologias dos idosos, enquanto a associação dela ao idoso tem que ser de responsabilidade de um único usuário.



**Figura 26 - Recorte 10 do Diagrama Entidade-Relacionamento**

Este recorte mostra uma ligação simples entre duas entidades, IDOSOS e INCIDENTES\_IDOSO. Um idoso pode não ter nenhum ou vários acidentes, mas um acidente que está associado a ele deve estar somente a ele.

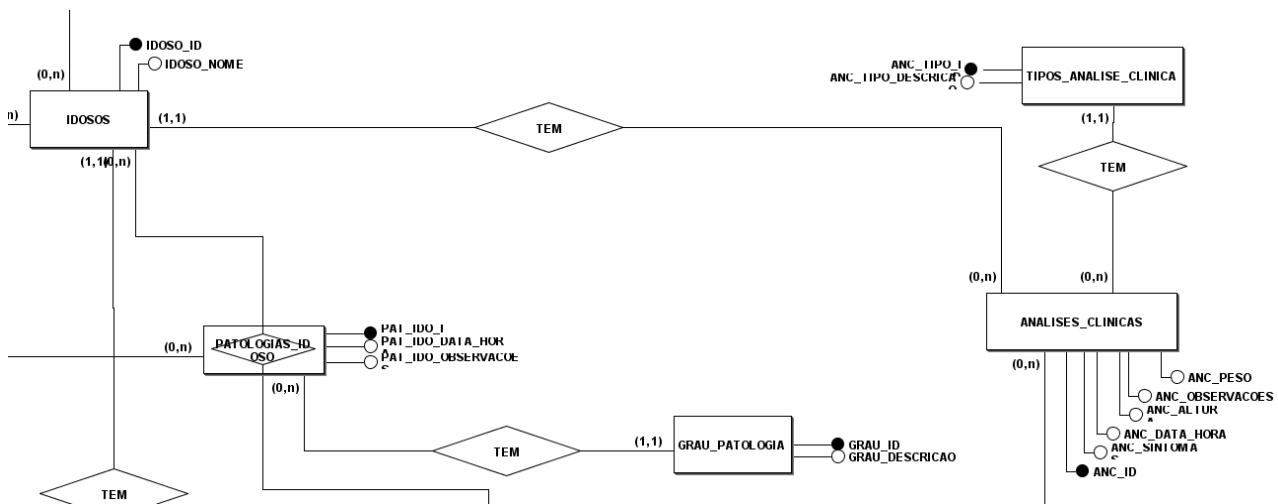


Figura 27 - Recorte 11 do Diagrama Entidade-Relacionamento

PATOLOGIAS\_IDOSO também tem uma ligação com a entidade GRAU\_PATOLOGIA (atributos sendo GRAU\_ID e GRAU\_DESCRICAO). Isso serve para especificar para cada idoso qual nível aquela patologia se encontra, como avançado, estágio inicial, estágio terminal, e outros dependendo das necessidades da organização. Um grau pode não ser associado a nenhuma ou várias patologias do idoso, mas uma patologia do idoso deve estar associada a somente um grau.

A última ligação feita na entidade IDOSOS é com a entidade ANALISES\_CLINICAS (sendo seus atributos ANC\_PESO, ANC\_OBSERVACOES, ANC\_ALTURA, ANC\_DATA\_HORA, ANC\_SINTOMAS e ANC\_ID). A análise clínica é associada a somente um idoso, e um idoso pode passar por nenhuma ou várias análises clínicas.

A entidade ANALISES\_CLINICAS também faz liga-se a entidade TIPO\_ANALISE\_CLINICA, que tem como atributos ANC\_TIPO\_ID e ANC\_TIPO\_DESCRICAO. Uma análise clínica só pode estar associada a um tipo, porém um tipo pode não estar associado a nenhuma análise clínica ou a várias.

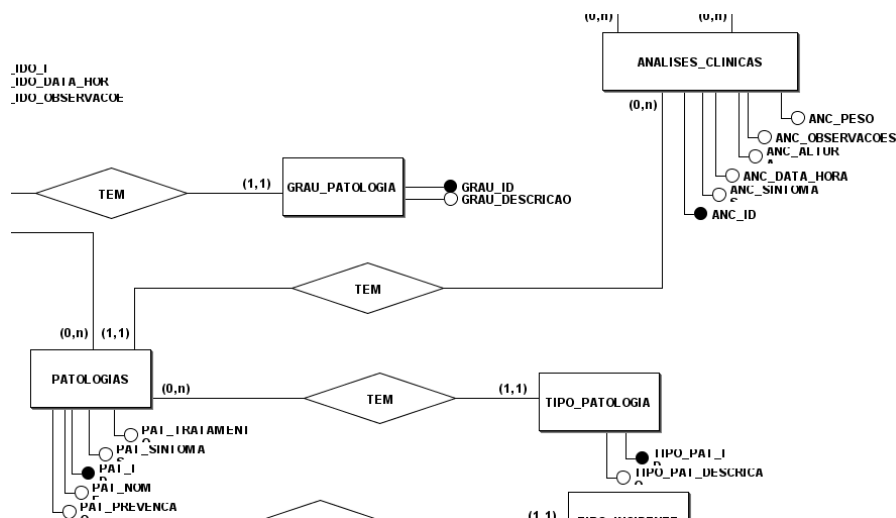
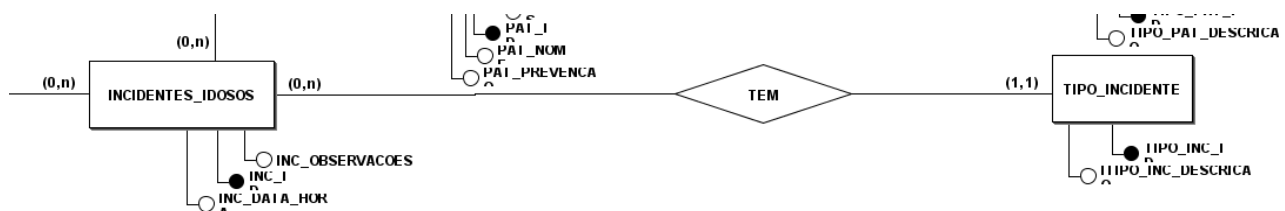


Figura 28 - Recorte 12 do Diagrama Entidade-Relacionamento

A entidade ANALISES\_CLINICAS também se conecta com a entidade PATOLOGIA, visto que quando feita uma análise clínica, pode haver a necessidade de se diagnosticar uma patologia. Uma análise clinica deve haver somente uma patologia associada a ela, enquanto uma patologia pode ser associada a nenhuma ou a várias análises clinicas.

O último vínculo da entidade PATOLOGIAS é a entidade TIPO\_PATOLOGIAS (com atributos TIPO\_PAT\_ID e TIPO\_PAT\_DESCRICAÇÃO). Trata-se de uma ligação onde patologias é associada a somente um tipo, e um tipo pode estar associado a nenhuma patologia ou a várias.



**Figura 29 - Recorte 13 do Diagrama Entidade-Relacionamento**

A última delimitação do diagrama mostra a ligação entre INCIDENTES\_IDOSOS e TIPO\_INCIDENTES (atributos sendo TIPO\_INC\_ID e TIPO\_INC\_DESCRICAÇÃO), trata-se de uma relação simples onde o tipo do incidente pode estar associado a nenhum a vários incidentes do idoso, e o incidente do idoso tem somente um tipo.

#### 2.2.4 Modelo Lógico

Depois da análise do Modelo Conceitual, foi gerado o Modelo Lógico do Banco de Dados, usando as próprias funções do aplicativo brModelo. Este Modelo é importante pois engloba de maneira lógica a relação entre as tabelas. Nele é mostrado quais tabelas estão conectadas através da sua chave identificadora, que se torna uma chave estrangeira na tabela que ela possui ligação.

O modelo com todas as suas tabelas encontra-se na íntegra na imagem abaixo:

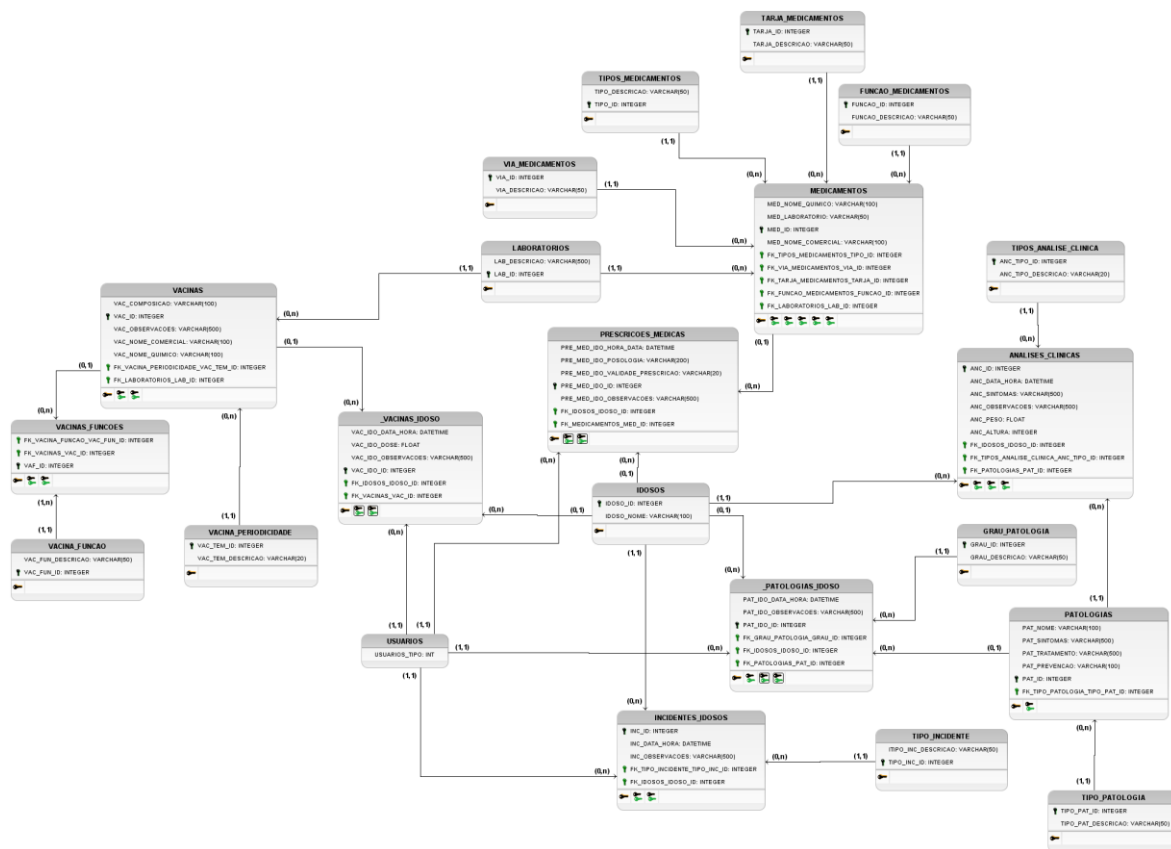


Figura 30 - Modelo Lógico

Este diagrama além de mostrar as chaves estrangeiras, também especifica o tipo de cada variável. Além disso, todas as entidades associativas mostradas no tópico anterior se transformaram em uma tabela a mais no modelo. Abaixo, é possível ver mais claramente essas especificações:

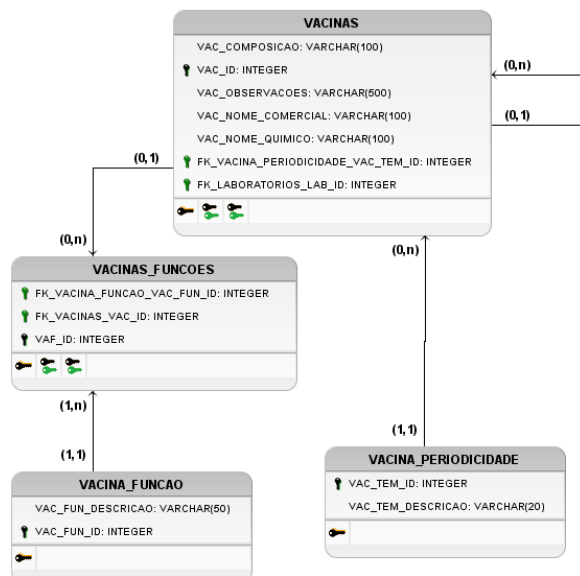


Figura 31 - Recorte 1 do Modelo Lógico

Na imagem acima nota-se a presença de 2 chaves estrangeiras na tabela VACINAS, sendo elas FK\_VACINA\_PERIODICIDADE\_VAC\_TEMP\_ID e FK\_LABORATORIOS\_LAB\_ID. Com elas, mesmo se não existisse as linhas ligando as tabelas, seria possível saber que a tabela VACINA se liga com as tabelas VACINA\_PERIODICIDADE e VACINA\_FUNCOES.

Ademais, tem-se a criação de uma tabela terciária, VACINAS\_FUNCOES, já que ela foi gerada de uma relação n:n entre as tabelas VACINA\_FUNCAO E VACINAS. Ela conta com 2 chaves estrangeiras (uma vinda de cada tabela de sua criação, FK\_VACINA\_FUNCAO\_VAC\_FUN\_ID e VAC\_VACINAS\_VAC\_ID e uma chave identificadora, VAF\_ID).

Como complemento, é preciso lembrar que é possível através dele saber todos os tipos das variáveis.

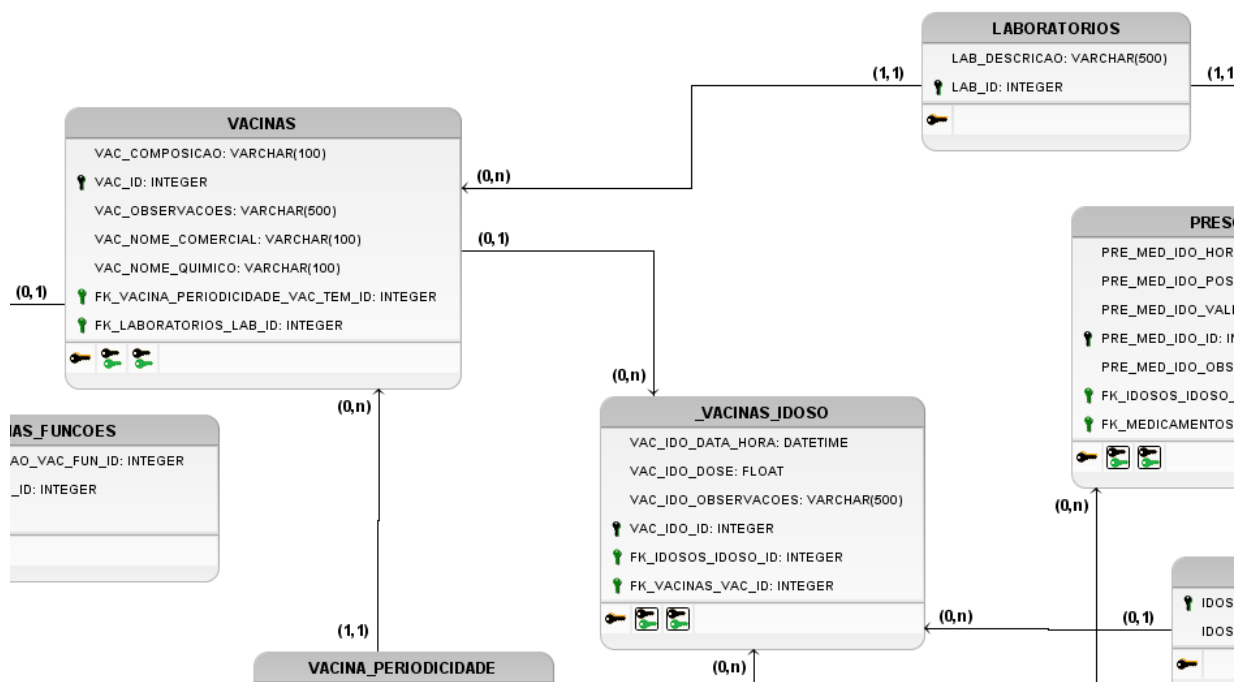


Figura 32 - Recorte 2 do Modelo Lógico

Nesta segunda imagem mostrada acima, temos a primeira tabela gerada a partir da entidade associativa, VACINAS\_IDOSO. Nela temos todas as chaves estrangeiras das tabelas que a originaram, sendo elas FK\_IDOSOS\_IDOSO\_ID e FK\_VACINAS\_VAC\_ID.

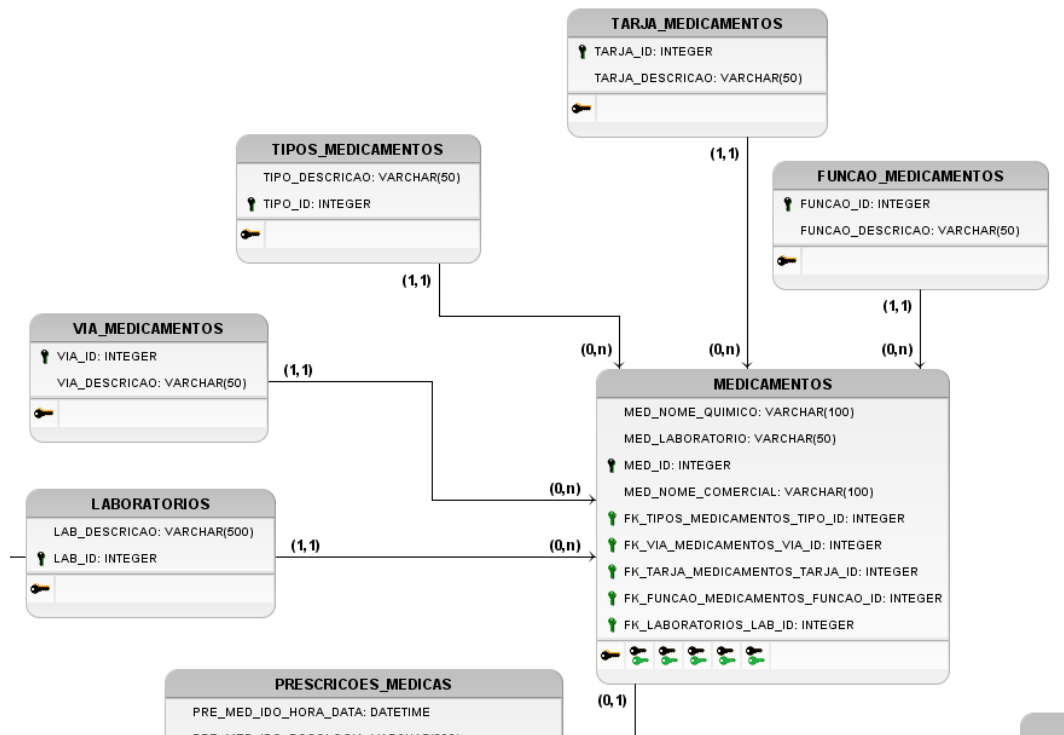


Figura 33 - Recorte 3 do Modelo Lógico

A tabela MEDICAMENTOS é uma que traz consigo muitas chaves estrangeiras. Elas têm origem nas tabelas LABORATORIOS, VIA\_MEDICAMENTOS, TIPOS\_MEDICAMENTOS, TARJA\_MEDICAMENTOS, FUNCAO\_MEDICAMENTOS, sendo respectivamente representadas na tabela pelas chaves FK\_LABORATORIOS\_ID, FK\_VIA\_MEDICAMENTO\_VIA\_ID, FK\_TIPOS\_MEDICAMENTOS\_TIPO\_ID, FK\_TARJA\_MEDICAMENTOS\_TARJA\_ID e FK\_FUNCAO\_MEDICAMENTOS\_FUNCAO\_ID.

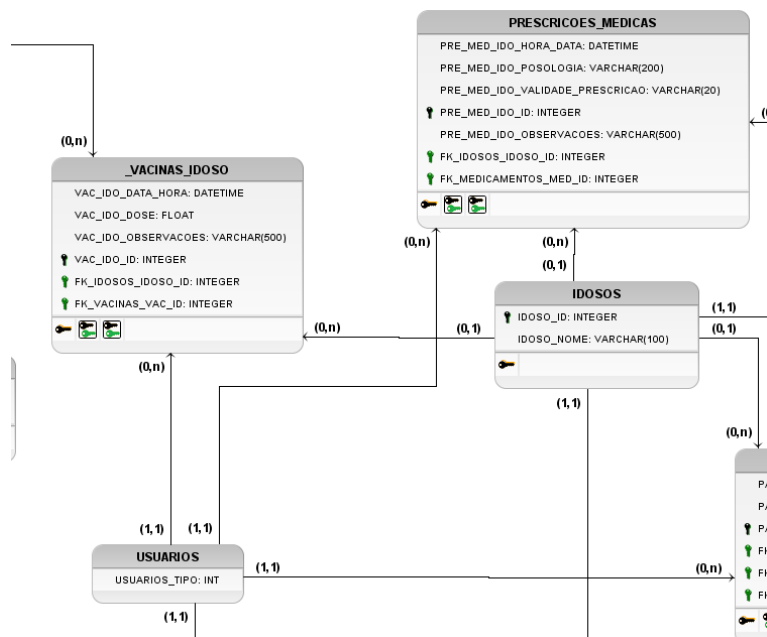
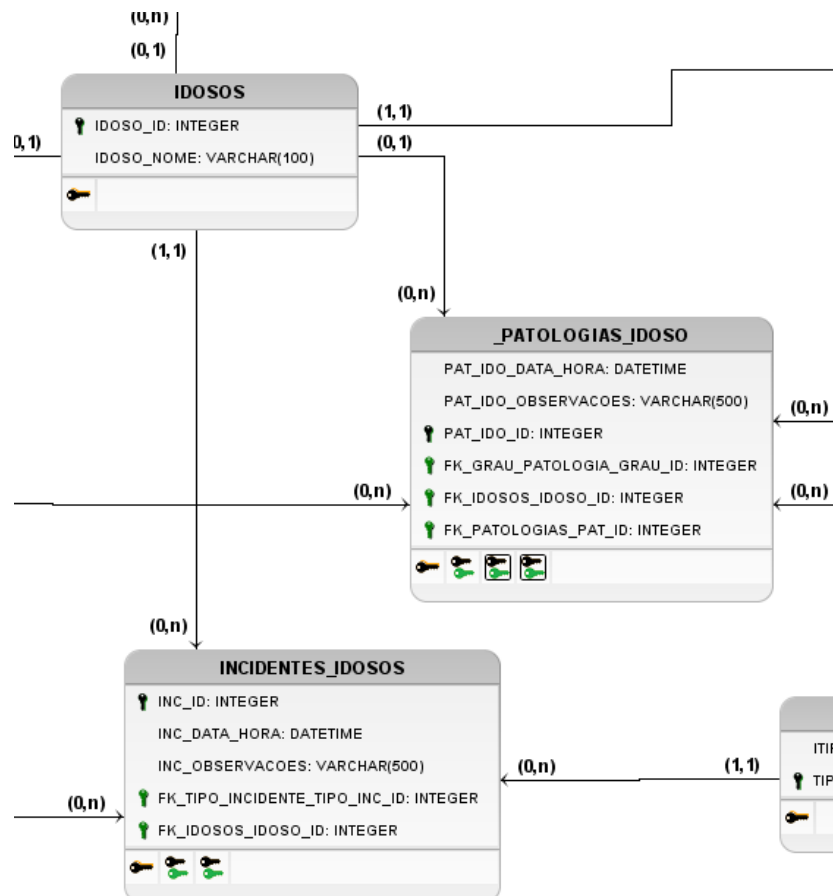


Figura 34 - Recorte 4 do Modelo Lógico

Eis que há mais uma tabela gerada da entidade associativa PRESCRICOES\_MEDICAS. Nela há 2 chaves estrangeiras sendo elas FK\_IDOSOS\_IDOSO\_ID e FK\_MEDICAMENTOS\_MED\_ID, vindas das tabelas IDOSO e MEDICAMENTOS, respectivamente.



**Figura 35 - Recorte 5 do Modelo Lógico**

A tabela INCIDENTES\_IDOSOS recebe as chaves das tabelas TIPO\_INCIDENTE e IDOSOS, sendo elas FK\_TIPO\_INCIDENTE\_TIPO\_INC\_ID e FK\_IDOSOS\_IDOSO\_ID.

Aqui há mais uma tabela vinda de uma entidade associativa, PATOLOGIA\_IDOSO. As chaves estrangeiras estão representadas na imagem pela cor verde.



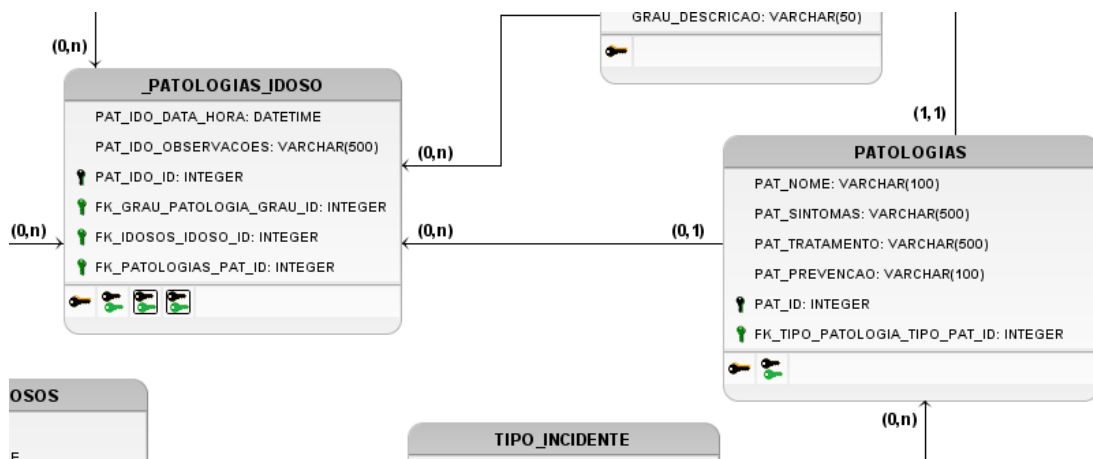


Figura 36 - Recorte 6 do Modelo Lógico

A tabela PATOLOGIAS conta com somente uma chave estrangeira, vinda da tabela TIPO\_PATOLOGIA, e chamando-se FK\_TIPO\_PATOLOGIA\_TIPO\_PAT\_ID.

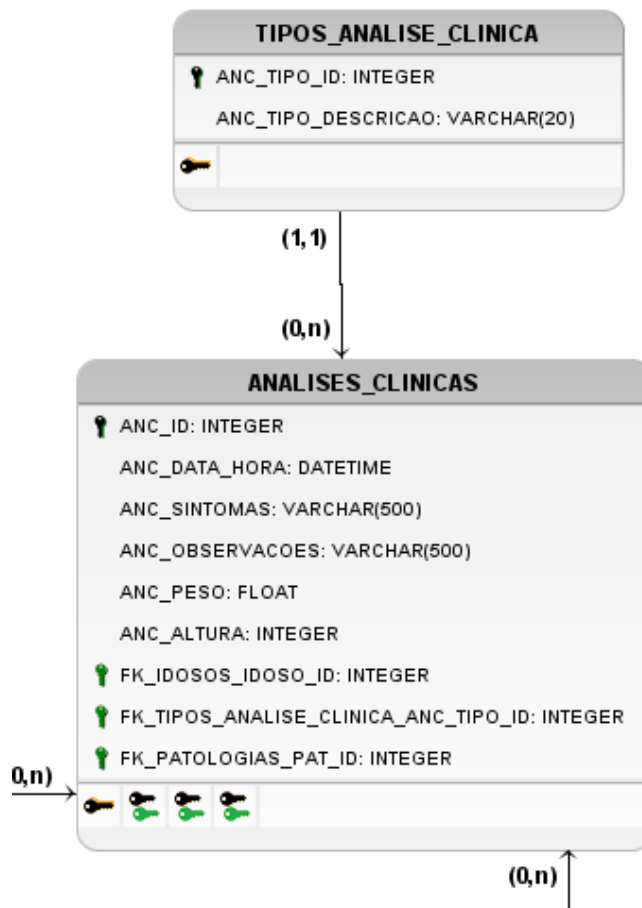


Figura 37 - Recorte 7 do Modelo Lógico

Nesta última parte, a tabela ANALISES\_CLINICAS conta com 3 chaves estrangeiras, FK\_IDSOS\_IDSOS\_ID, FK\_TIPOS\_ANALISE\_CLINICA\_ANC\_TIPO\_ID e FK\_PATOLOGIAS\_PAT\_ID.

Um ponto a ser mencionado é a padronização das nomeações das chaves estrangeiras. Por padrão, todas começam com FK, abreviação vinda da expressão em inglês ‘Foreign Key’, que traduzida significa justamente ‘Chave Estrangeira’.

### 2.2.5 Modelo Físico

Após gerado o Modelo Lógico, também com o auxílio da ferramenta brModelo, foi gerado o Modelo Físico, que se trata da codificação em linguagem SQL de todos os processos mostrados até aqui.

Abaixo é mostrado o código inteiro:

```
CREATE TABLE PATOLOGIAS (  
    PAT_NOME VARCHAR(100),  
    PAT_SINTOMAS VARCHAR(500),  
    PAT_TRATAMENTO VARCHAR(500),  
    PAT_PREVENCAO VARCHAR(100),  
    PAT_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    FK_TIPO_PATOLOGIA_TIPO_PAT_ID INTEGER  
);  
  
CREATE TABLE MEDICAMENTOS (  
    MED_NOME_QUIMICO VARCHAR(100),  
    MED_LABORATORIO VARCHAR(50),  
    MED_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    MED_NOME_COMERCIAL VARCHAR(100),  
    FK_TIPOS_MEDICAMENTOS_TIPO_ID INTEGER,  
    FK_VIA_MEDICAMENTOS_VIA_ID INTEGER,  
    FK_TARJA_MEDICAMENTOS_TARJA_ID INTEGER,  
    FK_FUNCAO_MEDICAMENTOS_FUNCAO_ID INTEGER,  
    FK_LABORATORIOS_LAB_ID INTEGER  
);  
  
CREATE TABLE TIPO_PATOLOGIA (  
    TIPO_PAT_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    TIPO_PAT_DESCRICAO VARCHAR(50)  
);  
  
CREATE TABLE VIA_MEDICAMENTOS (  
    VIA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    VIA_DESCRICAO VARCHAR(50)
```

);

```
CREATE TABLE VACINAS (  
    VAC_COMPOSICAO VARCHAR(100),  
    VAC_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    VAC_OBSERVACOES VARCHAR(500),  
    VAC_NOME_COMERCIAL VARCHAR(100),  
    VAC_NOME_QUIMICO VARCHAR(100),  
    FK_VACINA_PERIODICIDADE_VAC_TEM_ID INTEGER,  
    FK_LABORATORIOS_LAB_ID INTEGER  
);
```

```
CREATE TABLE TIPOS_MEDICAMENTOS (  
    TIPO_DESCRICAO VARCHAR(50),  
    TIPO_ID INTEGER AUTO_INCREMENT PRIMARY KEY  
);
```

```
CREATE TABLE TARJA_MEDICAMENTOS (  
    TARJA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    TARJA_DESCRICAO VARCHAR(50)  
);
```

```
CREATE TABLE FUNCAO_MEDICAMENTOS (  
    FUNCAO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    FUNCAO_DESCRICAO VARCHAR(50)  
);
```

```
CREATE TABLE VACINA_FUNCAO (  
    VAC_FUN_DESCRICAO VARCHAR(50),  
    VAC_FUN_ID INTEGER AUTO_INCREMENT PRIMARY KEY  
);
```

```
CREATE TABLE VACINA_PERIODICIDADE (  
    VAC_TEM_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    VAC_TEM_DESCRICAO VARCHAR(20)  
);
```

```
CREATE TABLE IDOSOS (  
    IDOSO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    IDOSO_NOME VARCHAR(100)  
);
```

```
CREATE TABLE USUARIOS (
    USUARIOS_TIPO INT
);
```

```
CREATE TABLE _PATOLOGIAS_IDOSO (
    PAT_IDO_DATA_HORA DATETIME,
    PAT_IDO_OBSERVACOES VARCHAR(500),
    PAT_IDO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FK_GRAU_PATOLOGIA_GRAU_ID INTEGER,
    FK_IDOSOS_IDOSO_ID INTEGER,
    FK_PATOLOGIAS_PAT_ID INTEGER
);
```

```
CREATE TABLE _VACINAS_IDOSO (
    VAC_IDO_DATA_HORA DATETIME,
    VAC_IDO_DOSE FLOAT,
    VAC_IDO_OBSERVACOES VARCHAR(500),
    VAC_IDO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FK_IDOSOS_IDOSO_ID INTEGER,
    FK_VACINAS_VAC_ID INTEGER
);
```

```
CREATE TABLE _PRESCRICOES_MEDICAS (
    PRE_MED_IDO_HORA_DATA DATETIME,
    PRE_MED_IDO_POSOLOGIA VARCHAR(200),
    PRE_MED_IDO_VALIDADE_PRESCRICAO VARCHAR(20),
    PRE_MED_IDO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    PRE_MED_IDO_OBSERVACOES VARCHAR(500),
    FK_IDOSOS_IDOSO_ID INTEGER,
    FK_MEDICAMENTOS_MED_ID INTEGER
);
```

```
CREATE TABLE TIPOS_ANALISE_CLINICA (
    ANC_TIPO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    ANC_TIPO_DESCRICAO VARCHAR(20)
);
```

```
CREATE TABLE INCIDENTES_IDOSOS (
    INC_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    INC_DATA_HORA DATETIME,
```

```

    INC_OBSERVACOES VARCHAR(500),
    FK_TIPO_INCIDENTE_TIPO_INC_ID INTEGER,
    FK_IDOSOS_IDOSO_ID INTEGER
);

CREATE TABLE TIPO_INCIDENTE (
    ITIPO_INC_DESCRICAO VARCHAR(50),
    TIPO_INC_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE LABORATORIOS (
    LAB_DESCRICAO VARCHAR(500),
    LAB_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE ANALISES_CLINICAS (
    ANC_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    ANC_DATA_HORA DATETIME,
    ANC_SINTOMAS VARCHAR(500),
    ANC_OBSERVACOES VARCHAR(500),
    ANC_PESO FLOAT,
    ANC_ALTURA INTEGER,
    FK_IDOSOS_IDOSO_ID INTEGER,
    FK_TIPOS_ANALISE_CLINICA_ANC_TIPO_ID INTEGER,
    FK_PATOLOGIAS_PAT_ID INTEGER
);

CREATE TABLE GRAU_PATOLOGIA (
    GRAU_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    GRAU_DESCRICAO VARCHAR(50)
);

CREATE TABLE VACINAS_FUNCOES (
    FK_VACINA_FUNCAO_VAC_FUN_ID INTEGER,
    FK_VACINAS_VAC_ID INTEGER,
    VAF_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

```

Como é possível observar, este modelo também atrás particularidades. Nas chaves primárias das tabelas, há o comando “AUTO\_INCREMENT”. Ele é responsável por fazer a contagem automática das inserções, impedindo assim a duplicidade de chaves primárias.

Após a criação das tabelas de acordo com o mostrado, surgiu a necessidade da alteração de alguns detalhes. Essas mudanças podem ser vistas através dos comandos abaixo:

```
ALTER TABLE PATOLOGIAS ADD CONSTRAINT FK_PATOLOGIAS_2
    FOREIGN KEY (FK_TIPO_PATOLOGIA_TIPO_PAT_ID)
    REFERENCES TIPO_PATOLOGIA (TIPO_PAT_ID)
    ON DELETE CASCADE;
```

```
ALTER TABLE MEDICAMENTOS ADD CONSTRAINT FK_MEDICAMENTOS_2
    FOREIGN KEY (FK_TIPOS_MEDICAMENTOS_TIPO_ID)
    REFERENCES TIPOS_MEDICAMENTOS (TIPO_ID)
    ON DELETE CASCADE;
```

```
ALTER TABLE MEDICAMENTOS ADD CONSTRAINT FK_MEDICAMENTOS_3
    FOREIGN KEY (FK_VIA_MEDICAMENTOS_VIA_ID)
    REFERENCES VIA_MEDICAMENTOS (VIA_ID)
    ON DELETE CASCADE;
```

```
ALTER TABLE MEDICAMENTOS ADD CONSTRAINT FK_MEDICAMENTOS_4
    FOREIGN KEY (FK_TARJA_MEDICAMENTOS_TARJA_ID)
    REFERENCES TARJA_MEDICAMENTOS (TARJA_ID)
    ON DELETE CASCADE;
```

```
ALTER TABLE MEDICAMENTOS ADD CONSTRAINT FK_MEDICAMENTOS_5
    FOREIGN KEY (FK_FUNCAO_MEDICAMENTOS_FUNCAO_ID)
    REFERENCES FUNCAO_MEDICAMENTOS (FUNCAO_ID)
    ON DELETE CASCADE;
```

```
ALTER TABLE MEDICAMENTOS ADD CONSTRAINT FK_MEDICAMENTOS_6
    FOREIGN KEY (FK_LABORATORIOS_LAB_ID)
    REFERENCES LABORATORIOS (LAB_ID)
    ON DELETE CASCADE;
```

```
ALTER TABLE VACINAS ADD CONSTRAINT FK_VACINAS_2
    FOREIGN KEY (FK_VACINA_PERIODICIDADE_VAC_TEM_ID)
    REFERENCES VACINA_PERIODICIDADE (VAC_TEM_ID)
    ON DELETE CASCADE;
```

```
ALTER TABLE VACINAS ADD CONSTRAINT FK_VACINAS_3
    FOREIGN KEY (FK_LABORATORIOS_LAB_ID)
```

```

REFERENCES LABORATORIOS (LAB_ID)
ON DELETE CASCADE;

ALTER TABLE _PATOLOGIAS_IDOSO ADD CONSTRAINT FK_PATOLOGIAS_IDOSO_2
FOREIGN KEY (FK_GRAU_PATOLOGIA_GRAU_ID)
REFERENCES GRAU_PATOLOGIA (GRAU_ID)
ON DELETE CASCADE;

ALTER TABLE _PATOLOGIAS_IDOSO ADD CONSTRAINT FK_PATOLOGIAS_IDOSO_3
FOREIGN KEY (FK_IDOSOS_IDOSO_ID)
REFERENCES IDOSOS (IDOSO_ID);

ALTER TABLE _PATOLOGIAS_IDOSO ADD CONSTRAINT FK_PATOLOGIAS_IDOSO_4
FOREIGN KEY (FK_PATOLOGIAS_PAT_ID)
REFERENCES PATOLOGIAS (PAT_ID);

ALTER TABLE _VACINAS_IDOSO ADD CONSTRAINT FK_VACINAS_IDOSO_2
FOREIGN KEY (FK_IDOSOS_IDOSO_ID)
REFERENCES IDOSOS (IDOSO_ID);

ALTER TABLE _VACINAS_IDOSO ADD CONSTRAINT FK_VACINAS_IDOSO_3
FOREIGN KEY (FK_VACINAS_VAC_ID)
REFERENCES VACINAS (VAC_ID);

ALTER TABLE _PRESCRICOES_MEDICAS ADD CONSTRAINT FK_PRESCRICOES_MEDICAS_2
FOREIGN KEY (FK_IDOSOS_IDOSO_ID)
REFERENCES IDOSOS (IDOSO_ID);

ALTER TABLE _PRESCRICOES_MEDICAS ADD CONSTRAINT FK_PRESCRICOES_MEDICAS_3
FOREIGN KEY (FK_MEDICAMENTOS_MED_ID)
REFERENCES MEDICAMENTOS (MED_ID);

ALTER TABLE INCIDENTES_IDOSOS ADD CONSTRAINT FK_INCIDENTES_IDOSOS_2
FOREIGN KEY (FK_TIPO_INCIDENTE_TIPO_INC_ID)
REFERENCES TIPO_INCIDENTE (TIPO_INC_ID)
ON DELETE CASCADE;

ALTER TABLE INCIDENTES_IDOSOS ADD CONSTRAINT FK_INCIDENTES_IDOSOS_3
FOREIGN KEY (FK_IDOSOS_IDOSO_ID)
REFERENCES IDOSOS (IDOSO_ID)
ON DELETE CASCADE;

```

```
ALTER TABLE ANALISES_CLINICAS ADD CONSTRAINT FK_ANALISES_CLINICAS_2
FOREIGN KEY (FK_IDOSOS_IDOSO_ID)
REFERENCES IDOSOS (IDOSO_ID)
ON DELETE CASCADE;
```

```
ALTER TABLE ANALISES_CLINICAS ADD CONSTRAINT FK_ANALISES_CLINICAS_3
FOREIGN KEY (FK_TIPOS_ANALISE_CLINICA_ANC_TIPO_ID)
REFERENCES TIPOS_ANALISE_CLINICA (ANC_TIPO_ID)
ON DELETE CASCADE;
```

```
ALTER TABLE ANALISES_CLINICAS ADD CONSTRAINT FK_ANALISES_CLINICAS_4
FOREIGN KEY (FK_PATOLOGIAS_PAT_ID)
REFERENCES PATOLOGIAS (PAT_ID)
ON DELETE CASCADE;
```

```
ALTER TABLE VACINAS_FUNCOES ADD CONSTRAINT FK_VACINAS_FUNCOES_2
FOREIGN KEY (FK_VACINA_FUNCAO_VAC_FUN_ID)
REFERENCES VACINA_FUNCAO (VAC_FUN_ID)
ON DELETE RESTRICT;
```

```
ALTER TABLE VACINAS_FUNCOES ADD CONSTRAINT FK_VACINAS_FUNCOES_3
FOREIGN KEY (FK_VACINAS_VAC_ID)
REFERENCES VACINAS (VAC_ID)
ON DELETE SET NULL;
```

A partir desta etapa, o Banco de Dados do Módulo de Prescrições Médicas e Controle de Incidentes está criado e pronto para ser integrado aos demais módulos.



### 3 Conclusões e Recomendações

A meta ao desenvolver este conteúdo era trazer mais rapidez para a organização dos Idosos, além de contribuir para um melhor desempenho das funções e maior segurança e confiabilidade, visto que todos os dados estarão salvos em caso de consulta.

Este documento teve por objetivo mostrar como foi feito o Bando de Dados do Módulo de Prescrições Médicas e Controle de Incidentes, iniciando com os Casos de Uso, e posteriormente a aplicação dos mesmos para criação dos Modelos de Banco, sendo eles o Modelo Conceitual, Modelo Lógico e Modelo Físico.

A primeira etapa deste trabalho era conceituar alguns termos que seriam importantes ao desenvolvimento. Esse levantamento de informações foi feito utilizando-se de três livros acadêmicos que se encontram referenciados no capítulo 4.

A segunda etapa consistia-se de apresentar e interpretar os Casos de Uso do Módulo de Prescrições Médicas e Controle de Incidentes. O diagrama de Casos de Uso foi feito usando a ferramenta Astah, e conta com 2 atores e 28 casos de uso.

A terceira etapa englobou-se efetivamente a primeira modelagem do Banco, através da apresentação e interpretação do Diagrama Entidade-Relacionamento. Esta modelagem foi feita utilizando a ferramenta brModelo, e o resultado final conta com 20 entidades, 3 entidades associativas e 19 relacionamentos.

A penúltima etapa buscava dar continuidade a geração dos modelos essenciais ao Banco. Nesta etapa, abrangeu-se o Modelo Lógico. Nela, foram mostradas as principais especificações deste modelo e suas diferenças em relação ao anterior. Assim como o anterior a este, foi criado através da ferramenta brModelo e conta com 22 tabelas.

A última etapa tratou-se de expor o resultado final que será usado efetivamente na aplicação do Bando de Dados, em linguagem SQL. Esta etapa da criação do Banco chama-se Modelo Físico. Ele foi gerado a partir do Modelo Lógico, também com o auxílio da ferramenta brModelo.

Em virtude de todas as etapas desenvolvidas, pode-se concluir que todos os objetivos deste trabalho foram concluídos.

Pode-se considerar alguns pontos positivos e negativos. Como ponto negativo pode-se citar a ferramenta brModelo, que devido a alguns problemas dificultou bastante a execução dos trabalhos. Além disso, uma das demais ferramentas usadas para o desenvolvimento do projeto como um todo também dificultou em alguns momentos o andamento do projeto foi o SVN. Em contrapartida, contou-se com uma equipe excelente de alunos que fizeram todo o projeto ser possível, incluindo também os mentores do projeto.

A partir da conclusão deste trabalho, foi possível realizar a integração do Banco de Dados com os demais dos outros módulos do projeto. Isso gerou um Banco de Dados completo para atender não somente as necessidades sobre prescrições médicas e incidentes, mas também sobre todas as áreas, como nutrição, alimentação, atividades físicas, prontuários, administração, dentre outras existentes em uma Instituição de Longa Permanência.

## 4 Referências Bibliográficas

- [1] IBGE. **IBGE São João da Boa Vista**. Instituto Brasileiro de Geografia e Estatísticas; 2017. Disponível em: <<https://cidades.ibge.gov.br/brasil/sp/sao-joao-da-boa-vista/panorama>>. Acesso em: 16/ago/2019.
- [2] ATLAS. **São João da Boa Vista, SP**. Atlas do Desenvolvimento Humano do Brasil. Disponível em: < [http://www.atlasbrasil.org.br/2013/pt/perfil\\_m/sao-joao-da-boa-vista\\_sp](http://www.atlasbrasil.org.br/2013/pt/perfil_m/sao-joao-da-boa-vista_sp)>. Acesso em: 16/ago/2019.
- [3] PRADO, Ana Laura. **As 40 melhores pequenas cidades para envelhecer**. EXAME. Disponível em: <<https://exame.abril.com.br/brasil/as-40-melhores-pequenas-cidades-para-envelhecer/>>. Acesso em: 16/ago/2019.
- [4] IFSP Campus São João da Boa Vista. Sobre o Campus. INSTITUTO FEDERAL DE SÃO PAULO. Disponível em: <<https://www.sbv.ifsp.edu.br/sobre-campus>>. Acesso em: 16/ago/2019.
- [5] IFSP Campus São João da Boa Vista. **Cursos Técnicos**. INSTITUTO FEDERAL DE SÃO PAULO. Disponível em: <<https://www.sbv.ifsp.edu.br/cursos-tecnicos>>. Acesso em: 16/ago/2019.
- [6] IFSP Campus São João da Boa Vista. **Técnico Integrado em Informática**. INSTITUTO FEDERAL DE SÃO PAULO. Disponível em: <<https://www.sbv.ifsp.edu.br/index.php/component/content/article/64-ensino/cursos/168-tecnico-integrado-informatica>>. Acesso em: 16/ago/2019.
- [8] MACHADO; Felipe, ABREU; Maurício. PROJETO DE BANCO DE DADOS. 17ª Edição. Editora Érica Ltda. Acessado em: 25/set/2019.
- [9] HEUSER; Carlos A. Projeto de Banco de Dados. 6ª Edição. BookMan. Acesso em: 25/set/2019.
- [10] ELMASRI; Ramez, NAVATHE; Shamkant B. SISTEMA DE BANCO DE DADOS. 4ª Edição. Person Education do Brasil Ltda. Acesso em: 25/set/2019.