

**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

**INTEGRAÇÃO DO BANCO DE DADOS VOLTADO PARA
O MODELO FÍSICO E DESENVOLVIMENTO DA
POPULAÇÃO**

Aluno: Marina Silene Jeronimo de Mello

Prontuário: 1620932

São João da Boa Vista – SP

2019

Resumo

Este trabalho demonstra o processo de desenvolvimento de um banco de dados de um software criado por alunos do ultimo ano de uma escola técnica, que possuem como objetivo criar um sistema que auxilie na administração de instituições de longa permanência. O banco de dados é uma parte indispensável para a elaboração de um software, e no decorrer do trabalho será apresentado a partir de etapas o desenvolvimento da integração do banco de dados do projeto citado acima. Inicia-se com levantamentos teóricos e apresentações e descrição do processo de desenvolvimento dos digramas, com alguns apontamentos de dificuldades. Por fim, todos os objetivos estabelecidos inicialmente foram cumpridos, visto que o banco de dados possui todas as funcionalidades essenciais. Além disso, apresenta recomendações para trabalhos futuros, colaborando posteriormente para outro projeto.

Palavras-chave: banco de dados, software, instituições de longa permanência.

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	Dado VS Informação	9
2.1.2	Banco de dados	9
2.1.3	SGBD.....	9
2.1.4	Modelo e dados.....	10
2.1.5	Inserção (INSERTS)	14
2.2	Etapas para o desenvolvimento da pesquisa.....	15
	2.2.1 Objetivos específicos 1: Apresentação do modelo conceitual e lógico do projeto gerações	15
	2.2.2 Objetivos específicos 2: Exibição das ferramentas utilizadas no desenvolvimento da atividade	34
	2.2.3 Objetivos específicos 3: Processo de desenvolvimento do modelo físico.....	36
	2.2.4 Objetivos específicos 4: Descrição do processo da população do banco de dados 50	
3	Conclusões e Recomendações	52
	Referências Bibliográficas	54

Índice de figuras

Figura 1: Divisão dos módulos do projeto 1	7
Figura 2: Divisão dos módulos do projeto 2	7
Figura 3 Exemplo do diagrama E_R [7]	11
Figura 4 Exemplo Modelo logico [8]	12
Figura 5 Exemplo de terceira tabela [9]	13
Figura 6 Exemplo de um modelo físico do projeto gerações	14
Figura 7 Exemplo de inserts (população) do banco de dados do projeto gerações	15
Figura 8 Modelo conceitual banco integrado	16
Figura 9 Parte 1 modelo conceitual	17
Figura 10 Parte 2 Modelo conceitual	18
Figura 11 Parte 3 Modelo conceitual	19
Figura 12 Parte 4 Modelo conceitual	20
Figura 13 Parte 5 Modelo conceitual	21
Figura 14 Parte 6 Modelo conceitual	22
Figura 15 Parte 7 Modelo conceitual	23
Figura 16 Parte 8 Modelo conceitual	23
Figura 17 Modelo lógico banco integrado	25
Figura 18 Parte 1 modelo logico	26
Figura 19 Parte 2 Modelo lógico	27
Figura 20 Parte 3 Modelo lógico	28
Figura 21 Parte 4 Modelo lógico	29
Figura 22 Parte 5 Modelo lógico	30
Figura 23 Parte 6 Modelo lógico	31
Figura 24 Parte 7 Modelo lógico	32
Figura 25 Parte 8 Modelo lógico	33
Figura 26 Parte 9 Modelo lógico	34
Figura 27 barra de tarefas do brModelo	35
Figura 28 Parte 1 modelo físico	37
Figura 29 Parte 2 modelo físico	38
Figura 30 Parte 3 modelo físico	39
Figura 31 Parte 4 Modelo físico	40
Figura 32 Parte 5 Modelo físico	41
Figura 33 Parte 6 Modelo Físico	42

Figura 34 Parte 7 Modelo físico	43
Figura 35 Parte 8 Modelo físico	44
Figura 36 Parte 9 Modelo físico	45
Figura 37 Parte10 Modelo físico	46
Figura 38 Parte 11 modelo físico	47
Figura 39 Parte 12 modelo físico	48
Figura 40 Parte 13 modelo físico	49
Figura 41 Parte 1 Inserts.....	50
Figura 42 Parte 2 inserts.....	51
Figura 43 Parte 3 inserts.....	51

1 Introdução

1.1 Contextualização/Motivação

São João da Boa Vista é uma cidade localizada no estado de São Paulo, com aproximadamente 84 mil habitantes segundo dados do IBGE, sido fundada em 24 de junho de 1821, com alto índice de desenvolvimento humano e boas condições econômicas, sociais, culturais e educacionais, sendo considerada a melhor cidade de até 100 mil habitantes para viver após os 60 anos de idade, possuindo grande parcela de sua população composta por idosos [1][2].

Nos últimos 10 anos o índice de desenvolvimento humano municipal (IDHM), comprovou que a expectativa de vida aumentou de 73 anos para 77 anos de idade na cidade citada anteriormente, e devido ao aumento da qualidade de vida, os investimentos aos idosos são constantes e necessários para seu bem-estar [2].

Consequentemente, existem diversas instituições que tem como objetivo cuidar dos idosos que não possuem condições de manter sua própria sobrevivência, um exemplo são as casas de longa permanência (Asilos) que tem como objetivo abrigarem as pessoas da terceira idade, oferecendo o necessário para viverem, preservando e cuidado de sua saúde, privacidade e outros aspectos básicos, proporcionando também lazer e incentivos às praticas de exercícios físicos. Atualmente, em São João da Boa Vista existem algumas dessas instituições que são ONGs ou privadas e podem ou não receberem auxilio familiar [3].

Contudo, os gerenciamentos dessas organizações não são eficazes, dificultando o trabalho dos profissionais presentes nos asilos, tendo todas as informações referentes aos idosos registradas manualmente, utilizando, por exemplo, a ferramenta EXCEL para esses registros, sem suporte de um programa que facilitasse a administração desses abrigos.

Ademais, existem instituições educacionais que possuem cursos técnicos e superiores, que promovem alguns projetos de pesquisas e softwares capazes de atender as necessidades da sociedade, tendo como um exemplo, o Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, possuindo um campus na cidade São João da Boa Vista, que tem como objetivo capacitar pessoas na área técnica e científica formando profissionais humanistas e inovadores tendo como vista, induzir o desenvolvimento econômico e social [4].

A partir desse intuito, existe o curso de informática integrado ao ensino médio que possui uma matéria denominada PDS (Pratica de desenvolvimento de sistemas), onde alunos do 4º ano elaboram juntamente com os professores das áreas técnicas softwares que visam garantir a melhoria da comunidade e desenvolvimento da cidade local.

Foi nesse contexto, que os alunos do 4º ano de 2019 do curso técnico em informática, iram desenvolver com as orientações dos professores da disciplina de PDS ao longo do ano, um portal que auxilie os cuidados diários dos idosos em uma instituição de longa permanência com intuito de melhorar e facilitar a vida das pessoas na melhor idade. Esse projeto foi nomeado Gerações e pela sua complexidade foi necessário ser dividido em nove módulos com suas respectivas responsabilidades e funções.

Figura 1: Divisão dos módulos do projeto 1

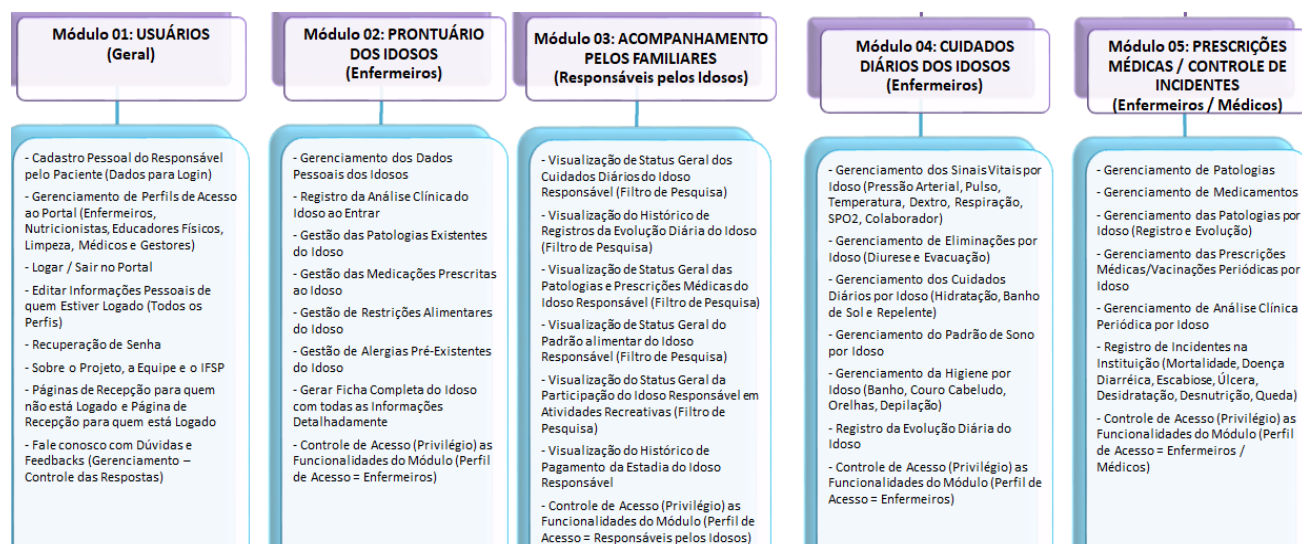
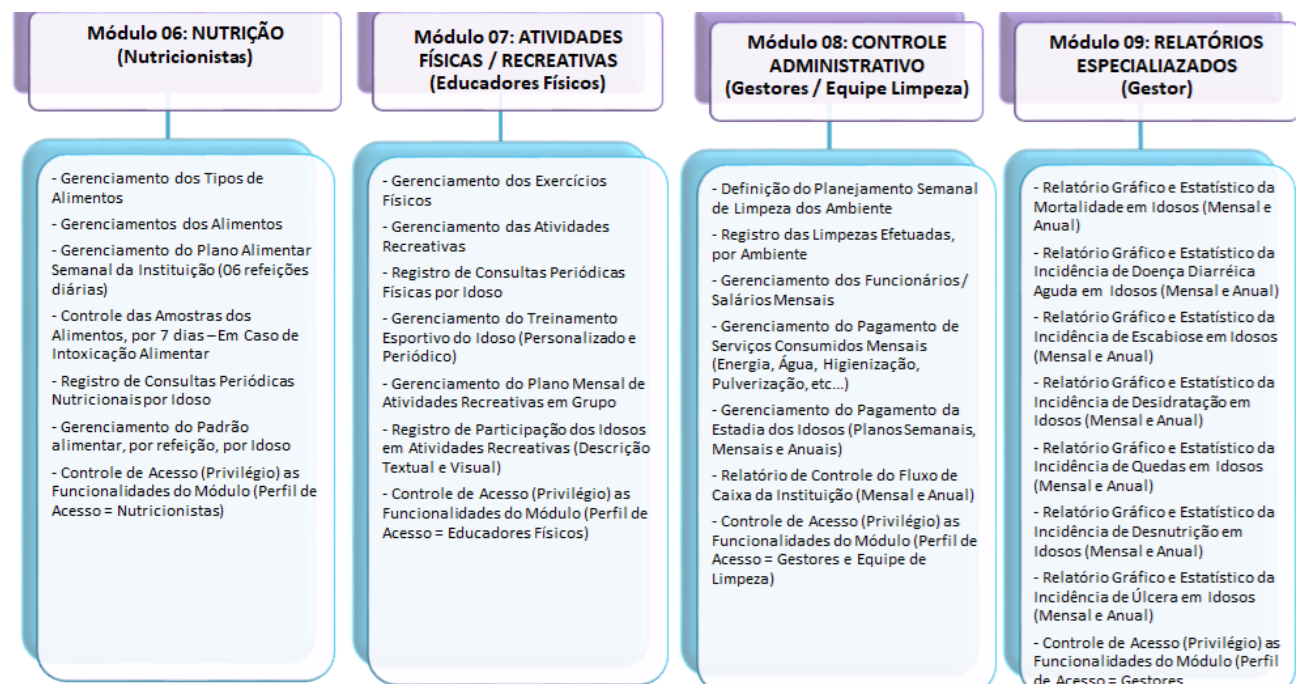


Figura 2: Divisão dos módulos do projeto 2



As imagens acima representam a separação dos módulos e quais são suas funcionalidades. O controle de atividades e desenvolvimento do projeto está sendo realizado através de duas ferramentas: Kanban e Redmine que devem ser atualizadas de acordo com a execução das tarefas.

Todo o desenvolvimento do software é armazenado em um repositório chamado SVN, onde os alunos que estão desenvolvendo o programa podem ter acesso às atividades do projeto como um todo.

Além disso, cada modulo contém de 6 a 7 alunos, que desempenham um papel específico, contendo dois analistas e testadores que estão voltados para a parte de documentação e análise dois administradores do banco de dados (DBAs) que realizam atividades relacionadas ao banco de dados (DataBase – DB) e dois a três desenvolvedores, que executam e manipulam a parte de códigos.

Os módulos devem ser integrados para a execução do software, e uma parte indispensável nesse processo, é a integração do banco de dados, onde é feita a junção do modelo conceitual das DBs de cada modulo, para a geração do modelo logico e físico integrado, sem essa etapa não é possível a realização funcional do sistema.

1.2 Objetivo geral da pesquisa

O objetivo desse trabalho é mostrar como foi o desenvolvimento da integração do banco de dados do projeto gerações, voltado para o modelo físico e o processo da população (Inserts).

1.3 Objetivos específicos

- Conceitos teóricos do banco de dados.
- Primeiramente, será apresentado o modelo conceitual e logico integrado.
- Logo após será exibido às ferramentas utilizadas no desenvolvimento da atividade.
- Em seguida será mostrado o desenvolvimento do modelo físico.
- Por fim, será descrito o processo da população do banco de dados.

1.4 Estrutura do documento

Quanto a estrutura da monografia, é contido o capítulo 1, apresenta a contextualização e motivação do trabalho. Capítulo 2, o desenvolvimento do objetivo geral da pesquisa, tais contendo conceitos teóricos. Capítulo 3, a conclusão do trabalho em si, com algumas recomendações. E por último as referências bibliográficas que serviram como base para o desenvolvimento do conteúdo desse documento.

2 Desenvolvimento

2.1 Levantamento bibliográfico

2.1.1 Dado VS Informação

Para a compreensão do tema desse trabalho, é essencial a conceitualização do que é o banco de dados, e tudo relacionado a ele.

A priori, é importante entender a diferença entre dado e informação: os dados são números, medidas ou valores que sozinhos não transmitem nenhum conhecimento, de maneira geral, são conteúdos quantificáveis e que por si só não transferem mensagens que possibilite o entendimento sobre determinada situação. Já a informação é a aglomeração e organização dos dados de forma a transmitir a compreensão dentro de um determinado conceito [5][6].

2.1.2 Banco de dados

Banco de dados, comumente chamado de DB (DataBase) é uma biblioteca de dados, onde é armazenado todas as informações referentes à uma ou mais organizações. Como por exemplo, um banco de dados de uma universidade pode conter informações e atividades de alunos, servidores, relacionamentos entre a matrícula do aluno com sua turma e os cursos ministrados pelos professores, entre outras coisas [5][6].

2.1.3 SGBD

Um Sistema de gerenciamento de Banco de Dados (SGBD) é um software capaz de auxiliar e manipular a manutenção da utilização dos conjuntos de dados armazenados em uma DB, permitindo ao usuário, o acesso a consultas e alterações desses dados, é responsável por salvar os dados no HD, manter em memória os mais acessados, ligar dados e metadados, disponibilizar uma interface para que programas e usuários externos acessem o banco de dados, controlar o acesso a informações, manter cópias dos dados para recuperação de uma possível falha, garantir transações na DB, enfim, sem o SGBD o banco de dados não funciona. Um Sistema de gerenciamento de Banco de Dados é projetado sobre um modelo de dados com apenas alguns construtores básicos, como o modelo relacional.

Exemplos de SGBDs são: Oracle, SQL Server, DB2, PostgreSQL, MySQL, entre outros [5][6].

2.1.4 Modelo de dados

Um modelo de dados são ferramentas conceituais usadas para a descrição, relacionamentos e semântica dos dados. Um SGBD possibilita que um usuário defina os dados a serem armazenados em termos de modelo de dados. É uma coleção de construtores de alto nível que são separados entre três grupos: Modelo entidade relacionamento (MER), modelo lógico e modelo físico [5][6].

2.1.4.1 Modelo Entidade – Relacionamento

O modelo de dados entidade – relacionamento (E_R) tem como base um conjunto de objetos básicos, chamados de entidades, e relacionamentos entre eles. Uma entidade é um objeto do mundo real que pode ser identificado por outros objetos, em outros termos, é tudo que possui um ou mais atributos. Por exemplo: Alunos é uma entidade que possui características, chamados de atributos, como nome, idade, CPF, etc. Professores também é uma entidade que possui certos atributos, que podem ser os mesmos que dos alunos, a diferença é que deve existir para cada entidade, um atributo identificador nomeado chave primaria que difere uma da outra e que seja única representada por uma “bolinha pintada” e relacionamentos são as ligações que uma entidade tem com a outra.

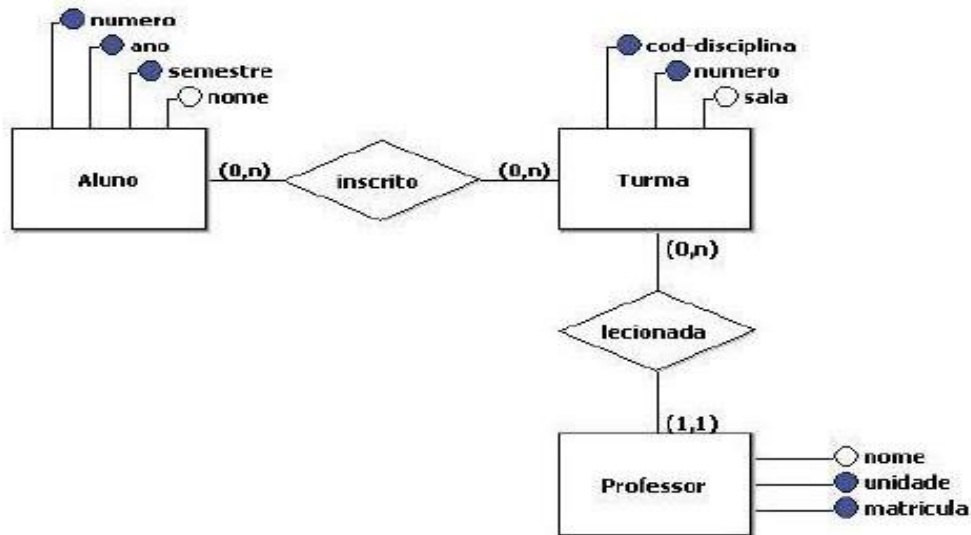
Além das entidades e relacionamentos, o modelo E_R apresenta algumas regras na qual o conteúdo do banco de dados deve respeitar. Uma das regras é o mapeamento das cardinalidades, a qual expressa o número de entidades às quais a outra entidade se relaciona.

Toda estrutura lógica do banco de dados pode ser expressa por meio do diagrama E_R, cujos construtores são:

- Retângulos, que representam os conjuntos de entidades;
- Elipses ou campos, que representam os atributos;
- Losangos, que representam os relacionamentos entre o conjunto de entidades;
- Linhas, que representam a união dos atributos às entidades e o conjunto de entidades aos seus relacionamentos.

Cada componente é nomeado de acordo com o nome da entidade ou relacionamento que representa [5][6].

Figura 3 Exemplo do diagrama E_R [7]

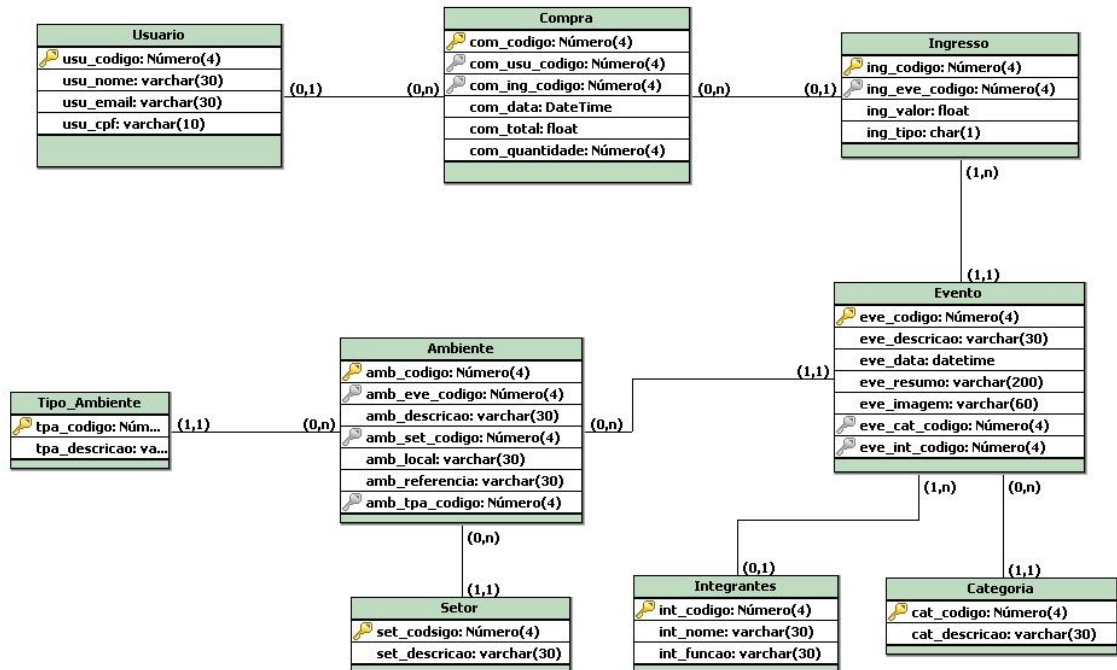


A imagem acima ilustra o exemplo citado anteriormente.

2.1.4.2 Modelo lógico

O modelo lógico, mais chamado de modelo relacional, é constituído por um conjunto de tabelas que representam tanto os dados como a relação entre eles. Cada tabela possui múltiplas linhas de atributos com nomes únicos. A diferença desse com o E_R, é que o relacional não precisa mais dos losangos para representar as relações, e é definido neste, o tamanho e o valor de cada atributo da tabela que será utilizado no modelo físico e no processo da população (INSERTS) [5][6].

Figura 4 Exemplo Modelo logico [8]

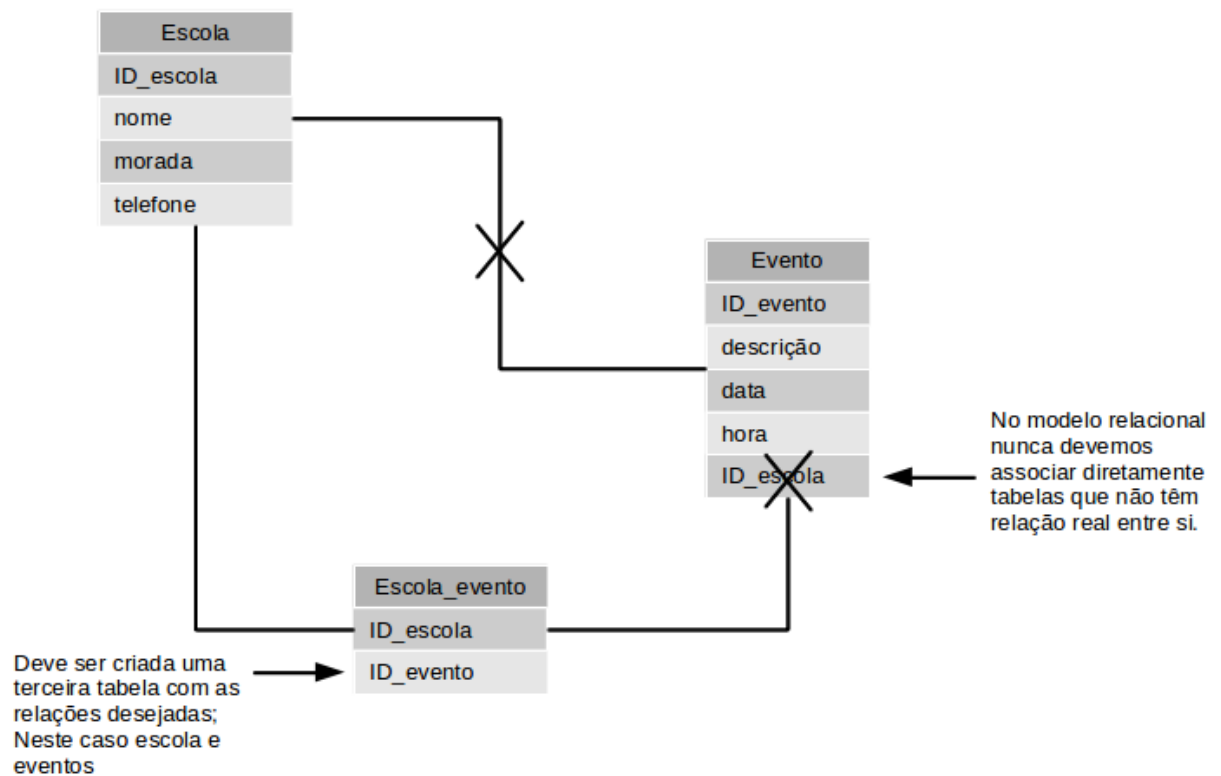


A imagem acima representa um exemplo de um digrama relacional, onde o número em parêntese identifica o tamanho de cada atributo.

Existe também nesse modelo, além da chave primaria, a chave estrangeira, ou *foreign key*, ela não diz respeito, especificamente, a uma tabela, mas sim a um relacionamento entre elas. De forma sucinta, a chave estrangeira é uma referência em uma tabela a uma chave primária de outra tabela. Por exemplo, no diagrama acima, duas tabelas: Compra e ingresso. Para montar-se um relacionamento entre elas, deve-se ter na tabela compra o campo ing_codigo fazendo referência à chave primária da tabela ingresso.

Quando existe uma relação de muitos para muitos entre duas entidades (0, n) e (0, n), cria-se uma terceira tabela, que referencia as chaves primárias das tabelas de ligação “n” tornando-se chaves estrangeiras em outra tabela [5][6].

Figura 5 Exemplo de terceira tabela [9]



Nessa imagem representa a terceira tabela gerada da relação de muitos para muitos das tabelas escola e evento.

2.1.4.3 Modelo Físico

O modelo físico é gerado a partir do modelo relacional, utilizando uma linguagem SQL, que suporta a criação, exclusão e modificação das tabelas, processo esse denominado Data Definition Language (DDL- Linguagem de definição de dados). Aqui são detalhados os componentes da estrutura física do banco, como tabelas, campos, tipos de valores, índices, etc. Nesse estágio é onde cria o banco de dados propriamente dito, usando o SGBD preferido [5][6].

Figura 6 Exemplo de um modelo físico do projeto gerações

```
CREATE TABLE CIDADES (  
    CID_NOME VARCHAR(128) NOT NULL,  
    CID_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    FK_ESTADOS_EST_ID INTEGER  
);  
  
CREATE TABLE ESTADOS (  
    EST_SIGLA VARCHAR(2) NOT NULL UNIQUE,  
    EST_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    EST_NOME VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE DOACOES (  
    DOA_DADOR VARCHAR(128),  
    DOA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    DOA_DATA_HORA DATETIME NOT NULL,  
    DOF_PRODUTO VARCHAR(128),  
    DOF_VALOR DOUBLE,  
    DOACOES_TIPO INT NOT NULL,  
    FK_GESTORES_USU_ID INTEGER  
);  
  
CREATE TABLE TIPOS_FEEDBACK (  
    TFE_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    TFE_TIPO VARCHAR(128) NOT NULL  
);
```

A imagem acima representa a criação SQL das tabelas definidas no modelo relacional, juntamente com o seu tipo e valor, podendo ser atualizadas e excluídas quando necessário.

2.1.5 Inserção (INSERTS)

INSERT é uma declaração SQL que adiciona um ou mais registros em qualquer tabela simples de um banco de dados relacional.

Para inserir dados em uma relação podemos especificar uma tupla a ser inserida ou escrever uma consulta cujo resultado é um conjunto de tuplas a inserir.

Tupla pode ser definido como a linha da tabela, ou o registro, como também é conhecido o conceito. Então, no banco de dados é juntar os dados necessários para preencher o que a tabela espera em apenas uma unidade conceitual.

Os valores dos atributos a serem inseridos, devem pertencer aos seus domínios, e as tuplas a serem inseridas devem possuir a ordem correta.

Figura 7 Exemplo de inserts (população) do banco de dados do projeto gerações

```
-- TABELA: TIPOS_MEDICAMENTOS
INSERT INTO `geracoes`.`tipos_medicamentos` (`TIP_DESCRICAO`) VALUES ('GENÉRICO');
INSERT INTO `geracoes`.`tipos_medicamentos` (`TIP_DESCRICAO`) VALUES ('SIMILAR');
INSERT INTO `geracoes`.`tipos_medicamentos` (`TIP_DESCRICAO`) VALUES ('ORIGINAL');
INSERT INTO `geracoes`.`tipos_medicamentos` (`TIP_DESCRICAO`) VALUES ('MANIPULADO');
INSERT INTO `geracoes`.`tipos_medicamentos` (`TIP_DESCRICAO`) VALUES ('FITOTERÁPICO');
INSERT INTO `geracoes`.`tipos_medicamentos` (`TIP_DESCRICAO`) VALUES ('ALOPÁTICO');
INSERT INTO `geracoes`.`tipos_medicamentos` (`TIP_DESCRICAO`) VALUES ('HOMEOPÁTICO');
INSERT INTO `geracoes`.`tipos_medicamentos` (`TIP_DESCRICAO`) VALUES ('DE REFERÊNCIA ');

-- TIPOS_PATOLOGIAS
INSERT INTO `geracoes`.`tipo_patologia` (`TIPO_PAT_DESCRICAO`) VALUES ('RENAL');
INSERT INTO `geracoes`.`tipo_patologia` (`TIPO_PAT_DESCRICAO`) VALUES ('CARDÍACA');
INSERT INTO `geracoes`.`tipo_patologia` (`TIPO_PAT_DESCRICAO`) VALUES ('CRÔNICA');
INSERT INTO `geracoes`.`tipo_patologia` (`TIPO_PAT_DESCRICAO`) VALUES ('RESPIRATÓRIA');
INSERT INTO `geracoes`.`tipo_patologia` (`TIPO_PAT_DESCRICAO`) VALUES ('MUSCULAR');
INSERT INTO `geracoes`.`tipo_patologia` (`TIPO_PAT_DESCRICAO`) VALUES ('REUMÁTICA');
INSERT INTO `geracoes`.`tipo_patologia` (`TIPO_PAT_DESCRICAO`) VALUES ('NEUROLÓGICA');

-- TABELA: TARJA_MEDICAMENTOS
INSERT INTO `geracoes`.`tarja_medicamentos` (`TAR_DESCRICAO`) VALUES ('PRETA');
INSERT INTO `geracoes`.`tarja_medicamentos` (`TAR_DESCRICAO`) VALUES ('VERMELHA');
INSERT INTO `geracoes`.`tarja_medicamentos` (`TAR_DESCRICAO`) VALUES ('AMARELA');
INSERT INTO `geracoes`.`tarja_medicamentos` (`TAR_DESCRICAO`) VALUES ('NÃO TARJADO');
```

A imagem acima representa a inserção dos dados das tabelas do modulo 5 – prescrição médica referente ao projeto gerações.

Inserir dados na DB é verificar se as tabelas estão inserindo corretamente os dados contidos nelas, e para fazer essa inserção é necessário seguir uma estrutura, sendo ela:

```
INSERT INTO `nome da tabela` (`coluna1`, `coluna 2`, `coluna n`)
VALUES ('valor1', 'valor2', 'valor n');
```

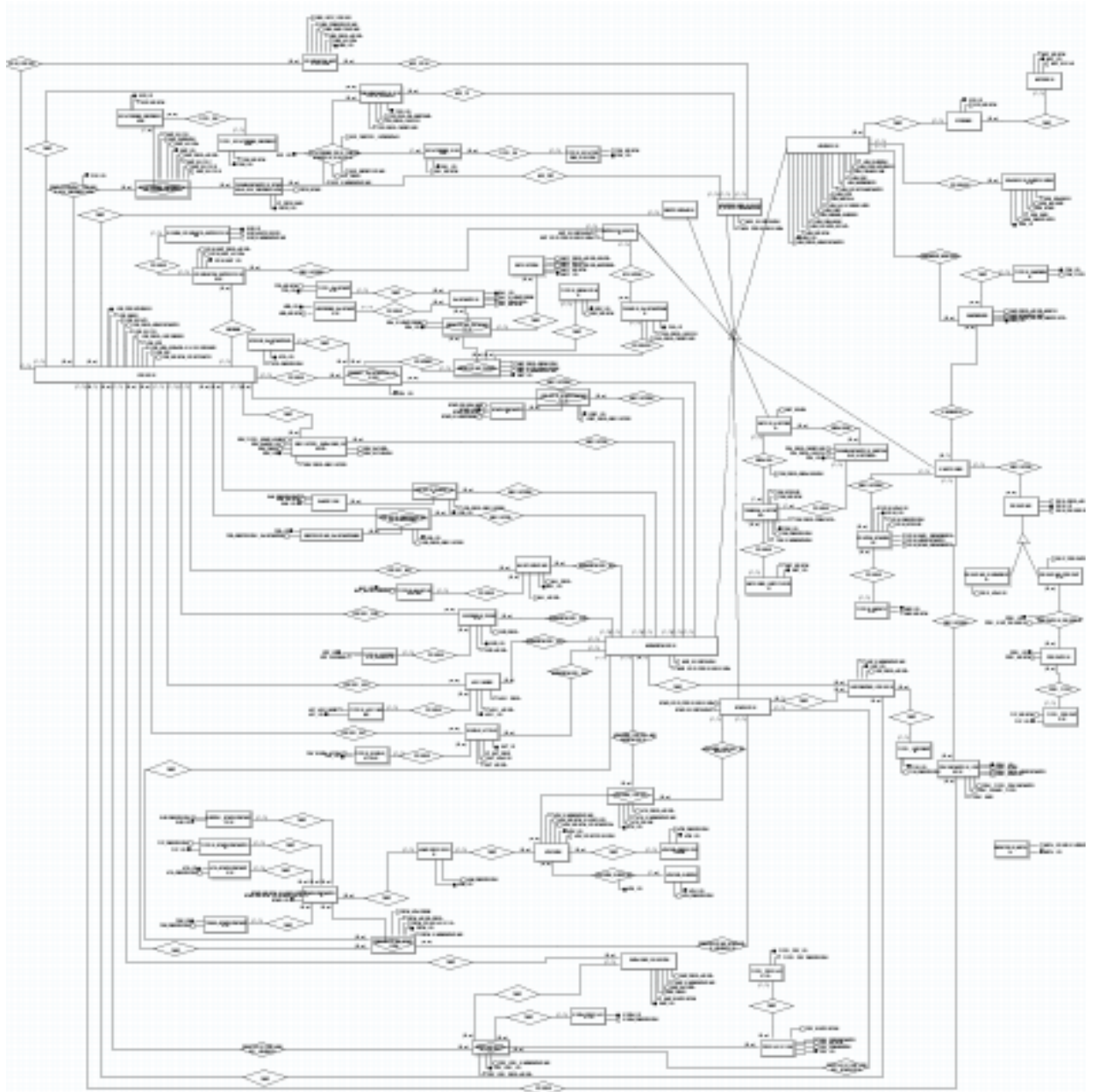
Não é necessário fazer a inserção de chaves primarias, pois frequentemente são AUTO_INCREMENT, ou seja, são preenchidas automaticamente ao passo que as informações das tabelas forem inseridas [5][6].

2.2 Etapas para o desenvolvimento da pesquisa

2.2.1 Objetivos específicos 1: Apresentação do modelo conceitual e lógico do projeto gerações

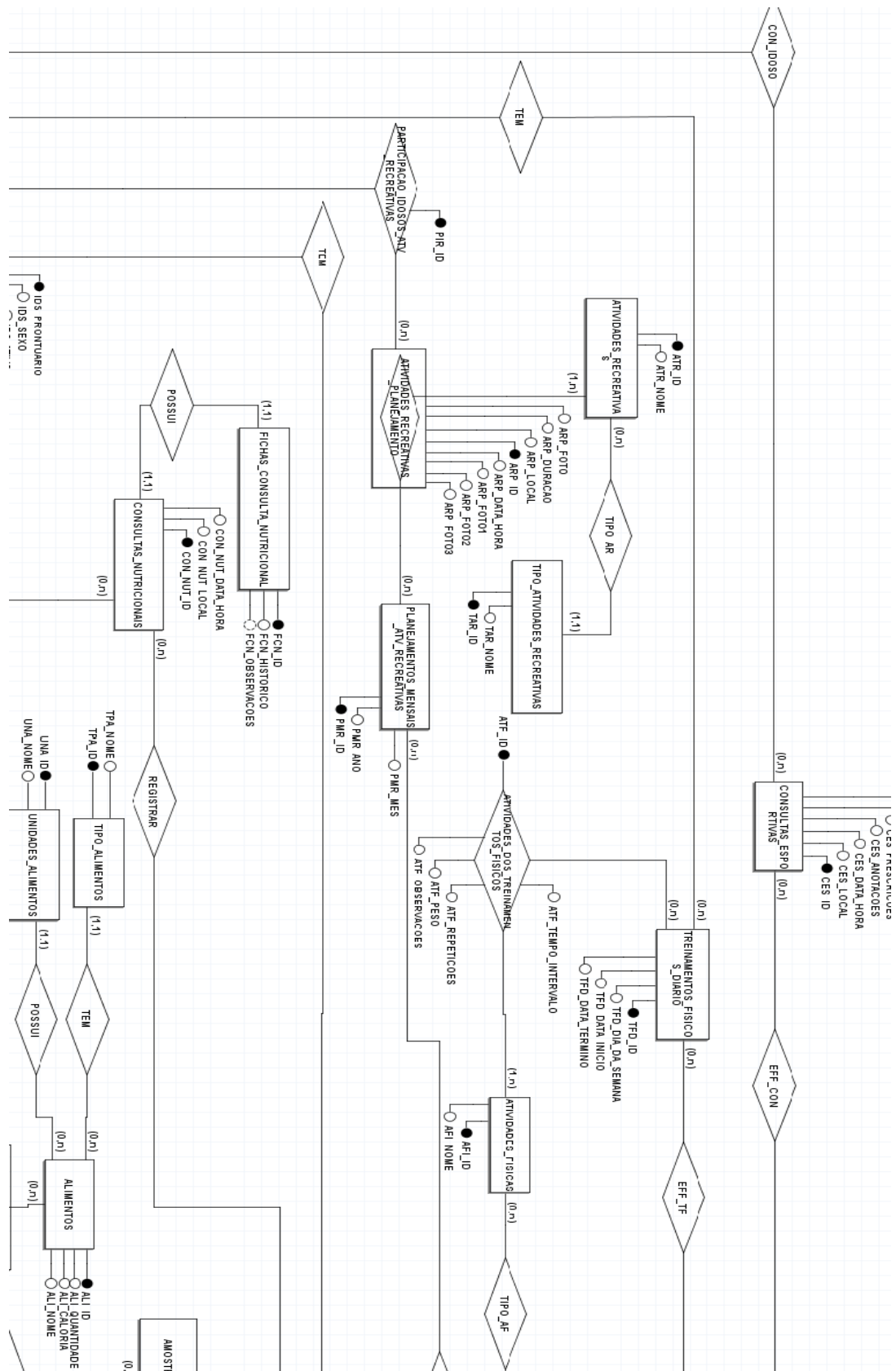
A seguir será apresentado o modelo conceitual do projeto gerações, logo após será descrito brevemente o seu processo de desenvolvimento.

Figura 8 Modelo conceitual banco integrado



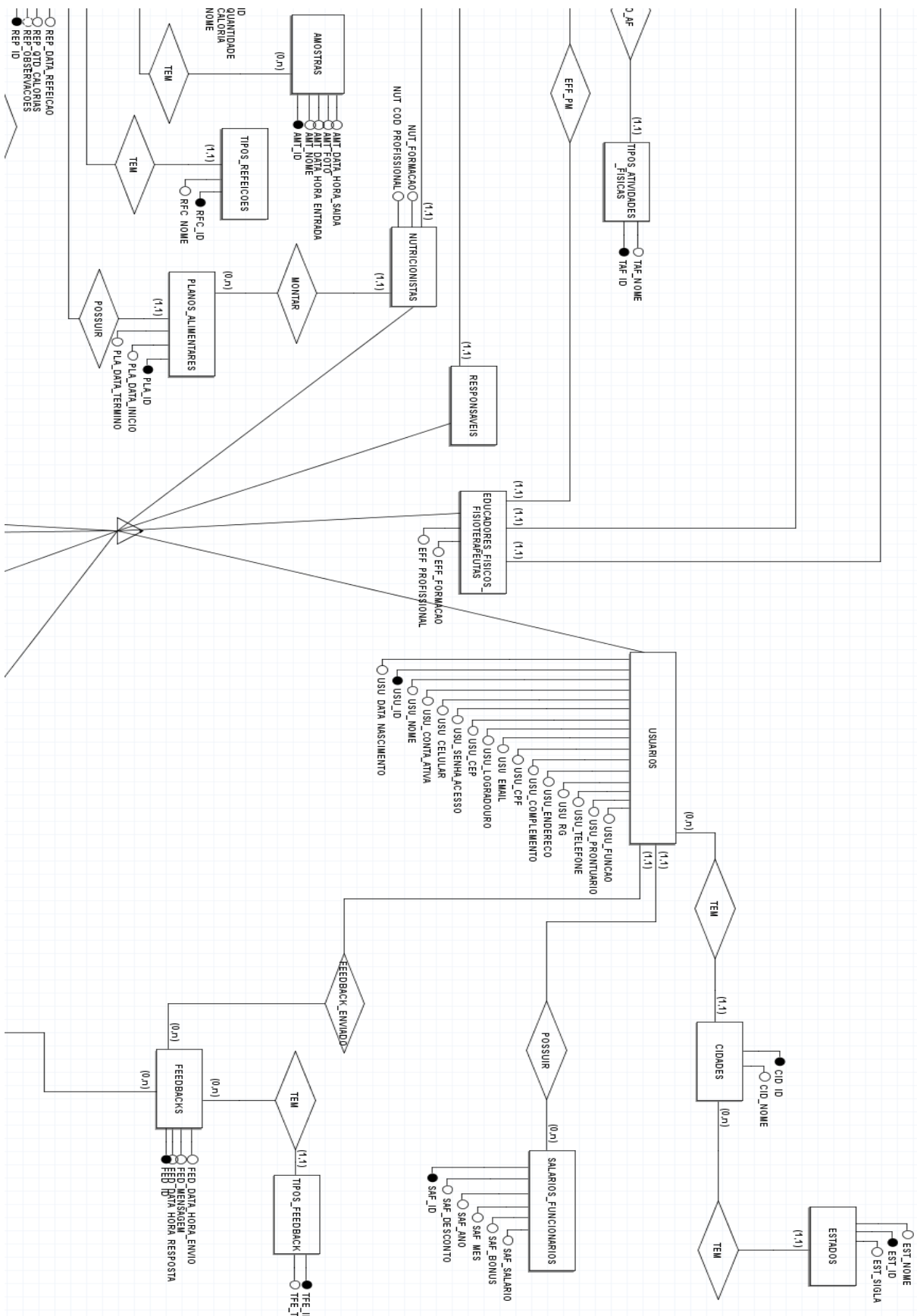
A imagem acima representa o modelo conceitual do banco integrado do projeto gerações, e a seguir será mostrado com ampliação esse diagrama, para uma melhor visualização.

Figura 9 Parte 1 modelo conceitual



A imagem apresenta em sua maioria entidades e relacionamentos relacionados ao banco de dados do módulo 7 – Atividades físicas e recreativas, e algumas do módulo 6 – Nutrição.

Figura 10 Parte 2 Modelo conceitual



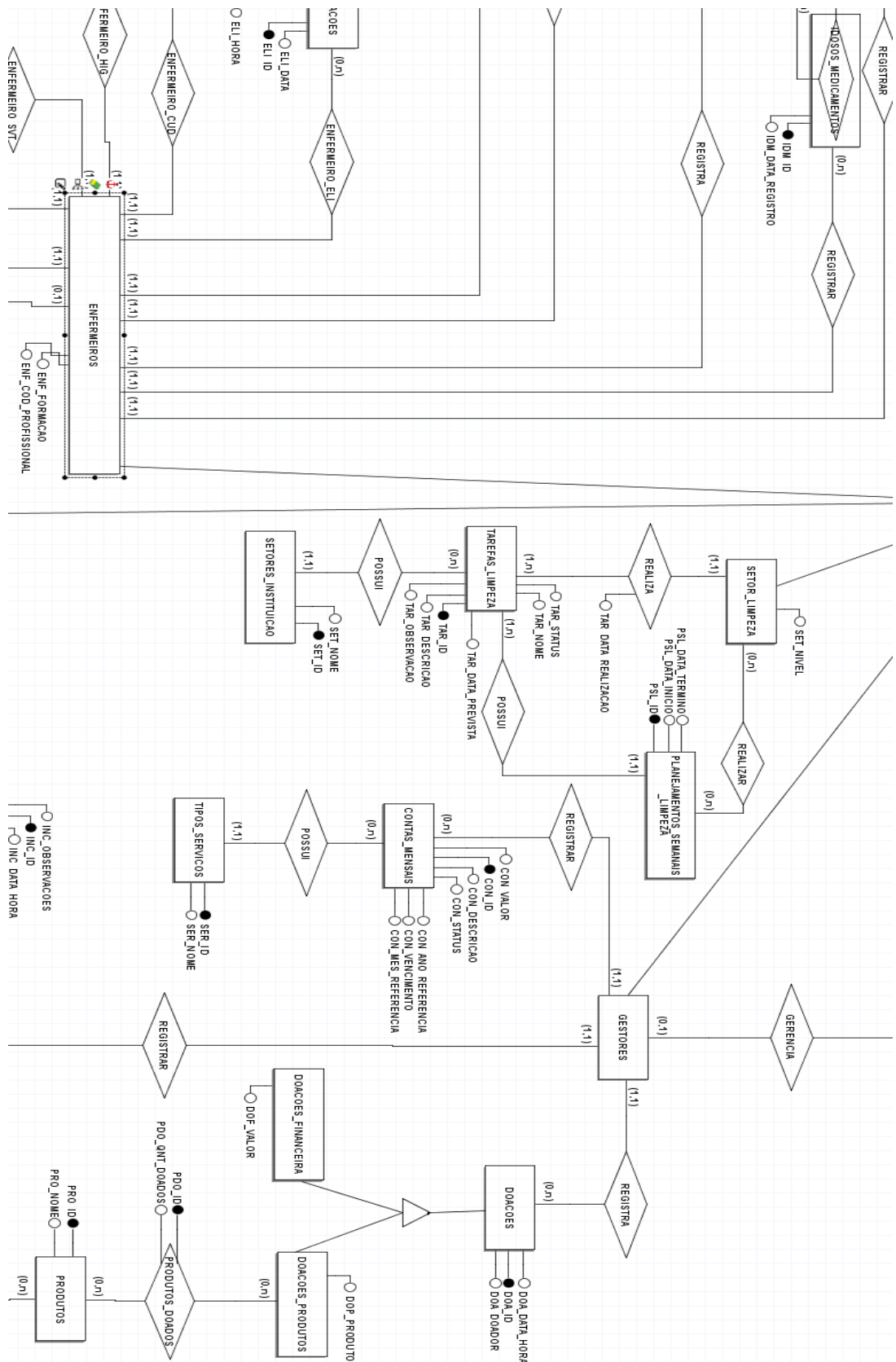
Essa imagem esta relacionada às entidades e relacionamentos do módulo 1 – Usuários, Módulo 6 – Nutrição e Módulo 8 – Controle administrativo.

The diagram is an Entity-Relationship (ER) model for a medical system. It includes the following components:

- Entities and Attributes:**
 - IDOSOS:** Attributes include IDOS_SEXO, IDOS_ATIVO, IDOS_DATA_MASCIMENTO, IDOS_FOTO, IDOS_DATA_INGRESSO, IDOS_CPF, IDOS_USA_FRALDA_OU_FORRINHO, IDOS_RG, and IDOS_NOME_COMPLETO.
 - UNIDADES_ALIMENTOS:** Attributes include UNA_NOME.
 - RESTRIÇÕES_ALIMENTARES:** Attributes include TRA_ID, TRA_DESCRICAO, and TRA_DATA_REGISTRO.
 - ELIMINACÕES:** Attributes include ELT_ID, ELT_DATA, ELT_HORA, and ELT_TIPO.
 - ALERTAS:** Attributes include ALE_ID, ALE_NOME, ALE_DATA, and ALE_TIPO.
 - RESTRIÇÕES_ALERGIAS:** Attributes include IDA_ID, IDA_DATA, and IDA_TIPO.
 - REGISTRO_ANALISES_CLINICA:** Attributes include RAC_TPO_SANGUE, RAC_ALERGIA, RAC_PECO, RAC_SITUACAO, and RAC_DATA_REGISTRO.
 - MEDICAMENTOS:** Attributes include MED_DOSAGEM, MED_ID, MED_NOME, MED_QUANTIDADE, and MED_DATA.
 - REFEICOES_PLANO:** Attributes include REF_DATA_REFEICAO, REF_QUANTIDADE, REF_CALORIAS, and REF_OBSERVACOES.
- Relationships (Diamonds):**
 - RECEBE:** Connects IDOSOS and UNIDADES_ALIMENTOS.
 - POSSUI:** Connects IDOSOS and RESTRIÇÕES_ALIMENTARES.
 - CONTEM:** Connects RESTRIÇÕES_ALIMENTARES and REFEICOES_PLANO.
 - REGISTRA:** Connects RESTRIÇÕES_ALIMENTARES and ELIMINACÕES.
 - REGISTRA:** Connects ELIMINACÕES and ENFERMEIROS.
 - REGISTRA:** Connects RESTRIÇÕES_ALERGIAS and ENFERMEIROS.
 - REGISTRA:** Connects REGISTRO_ANALISES_CLINICA and ENFERMEIROS.
 - REGISTRA:** Connects MEDICAMENTOS and ENFERMEIROS.
 - REGISTRA:** Connects REFEICOES_PLANO and ENFERMEIROS.
- Cardinalities:**
 - IDOSOS (1) to RECEBE (1) to UNIDADES_ALIMENTOS (1).
 - IDOSOS (1) to POSSUI (1) to RESTRIÇÕES_ALIMENTARES (1).
 - RESTRIÇÕES_ALIMENTARES (1) to CONTEM (1) to REFEICOES_PLANO (1).
 - RESTRIÇÕES_ALIMENTARES (1) to REGISTRA (1) to ELIMINACÕES (1).
 - ELIMINACÕES (1) to REGISTRA (1) to ENFERMEIROS (1).
 - RESTRIÇÕES_ALERGIAS (1) to REGISTRA (1) to ENFERMEIROS (1).
 - REGISTRO_ANALISES_CLINICA (1) to REGISTRA (1) to ENFERMEIROS (1).
 - MEDICAMENTOS (1) to REGISTRA (1) to ENFERMEIROS (1).
 - REFEICOES_PLANO (1) to REGISTRA (1) to ENFERMEIROS (1).

19

Figura 12 Parte 4 Modelo conceitual



A imagem apresenta entidades e relacionamentos do módulo 1 – Usuários, módulo 8 – controle administrativo e módulo 9 - relatórios que trabalham com doações.

Figura 13 Parte 5 Modelo conceitual

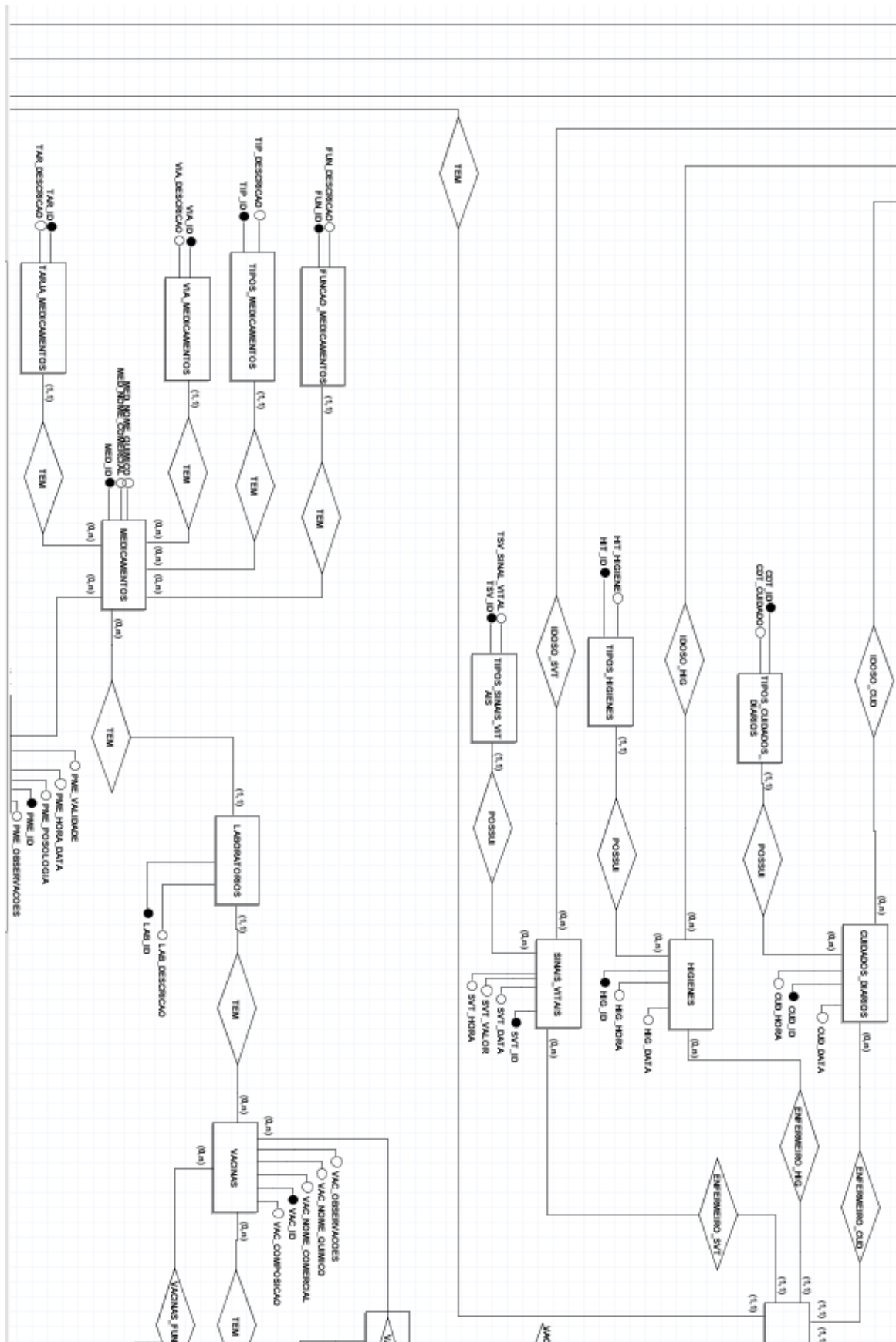
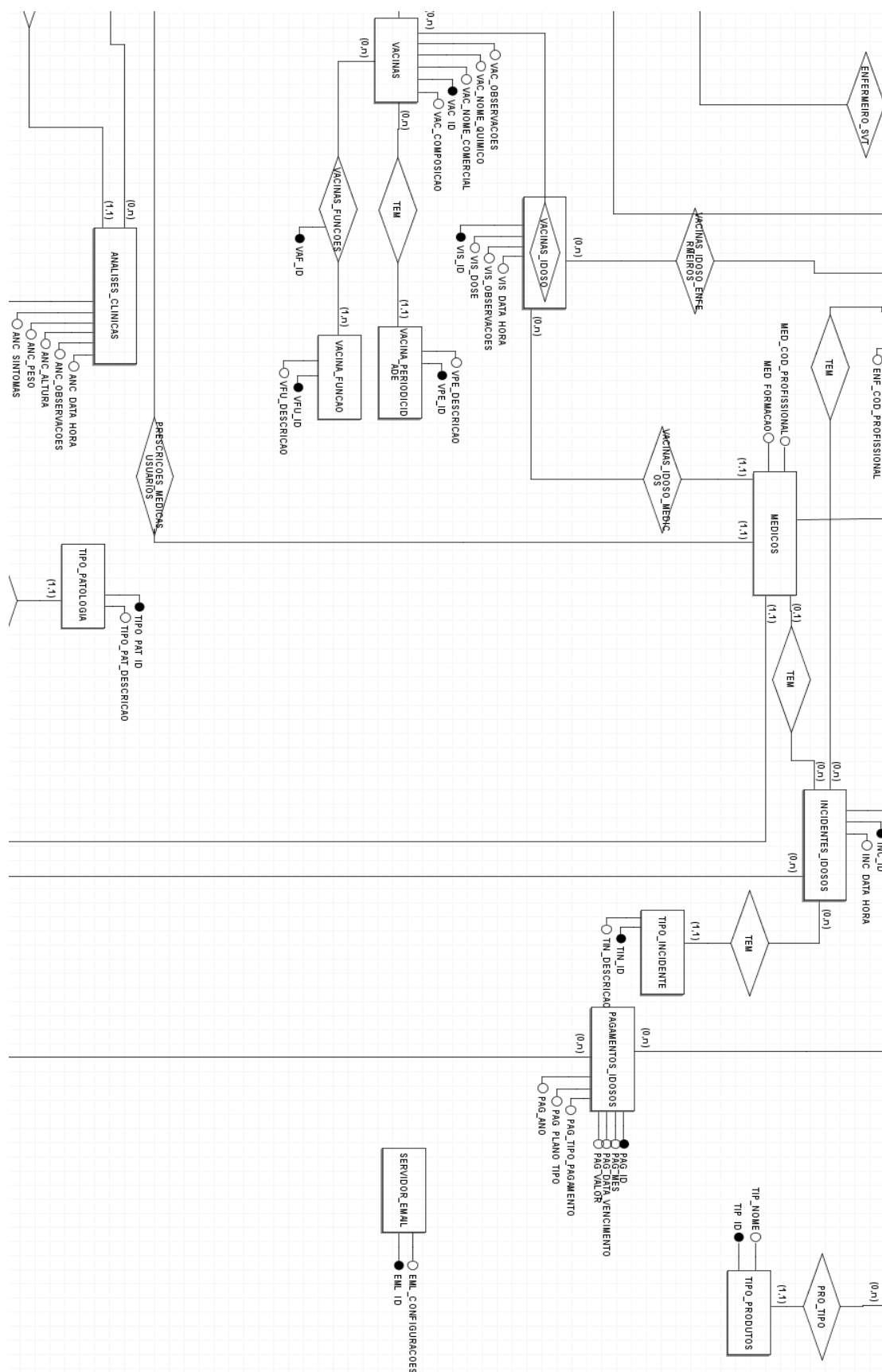


Figura 14 Parte 6 Modelo conceitual



Nessa imagem acima e nas duas imagens a seguir contem apenas entidades e relacionamentos relacionados ao módulo 5 – prescrições médicas.

Figura 15 Parte 7 Modelo conceitual

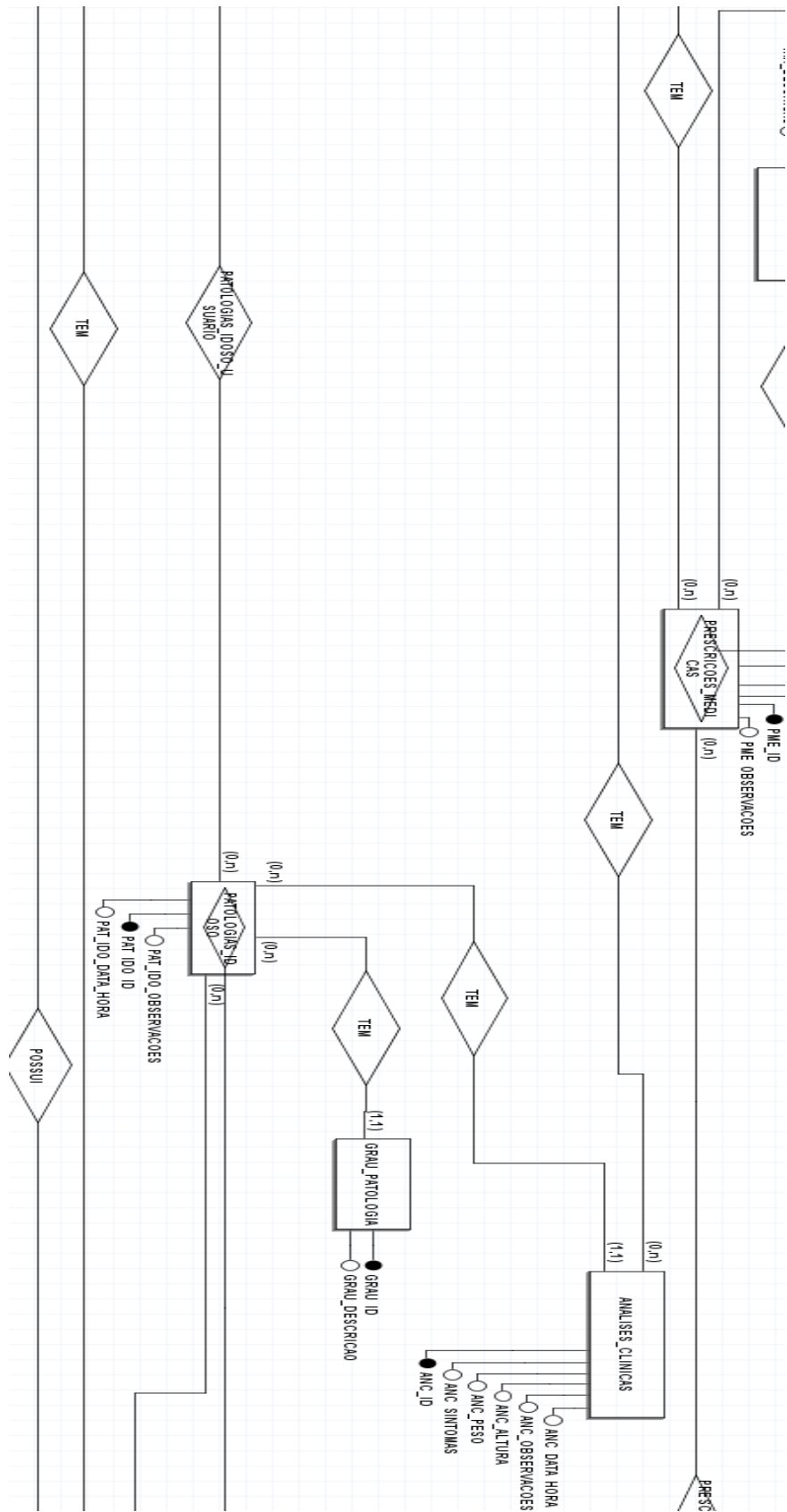
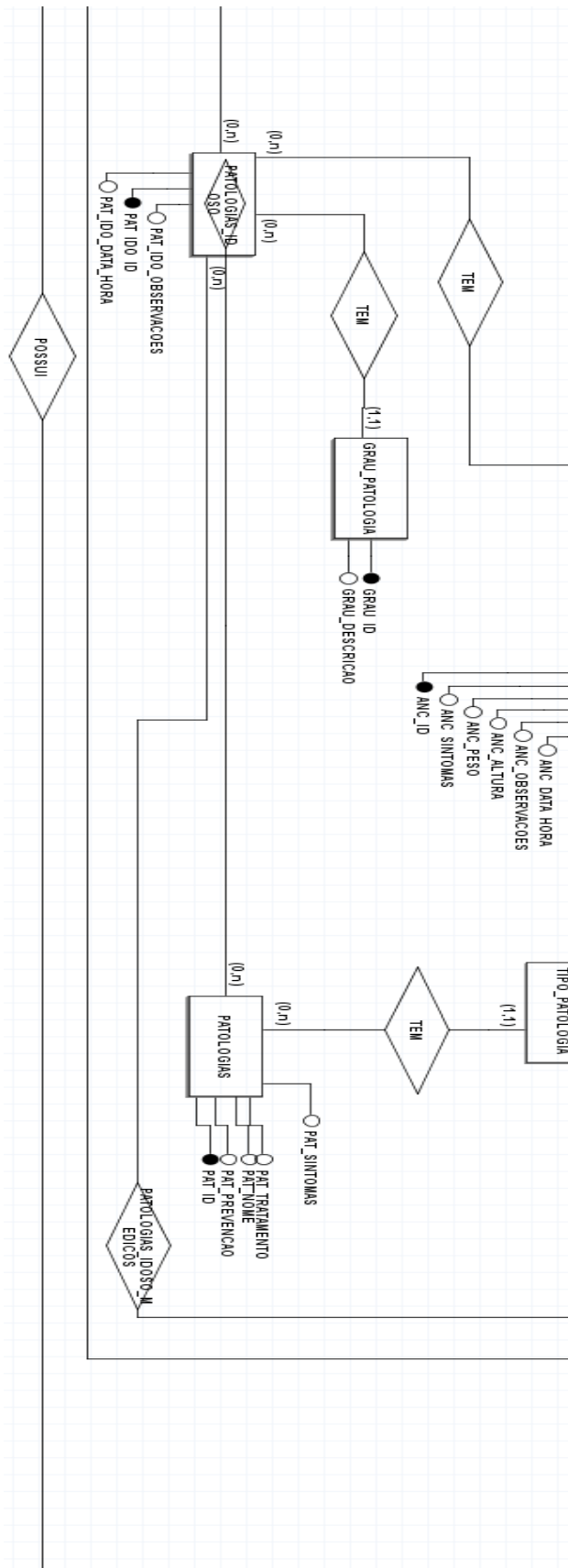


Figura 16 Parte 8 Modelo conceitual



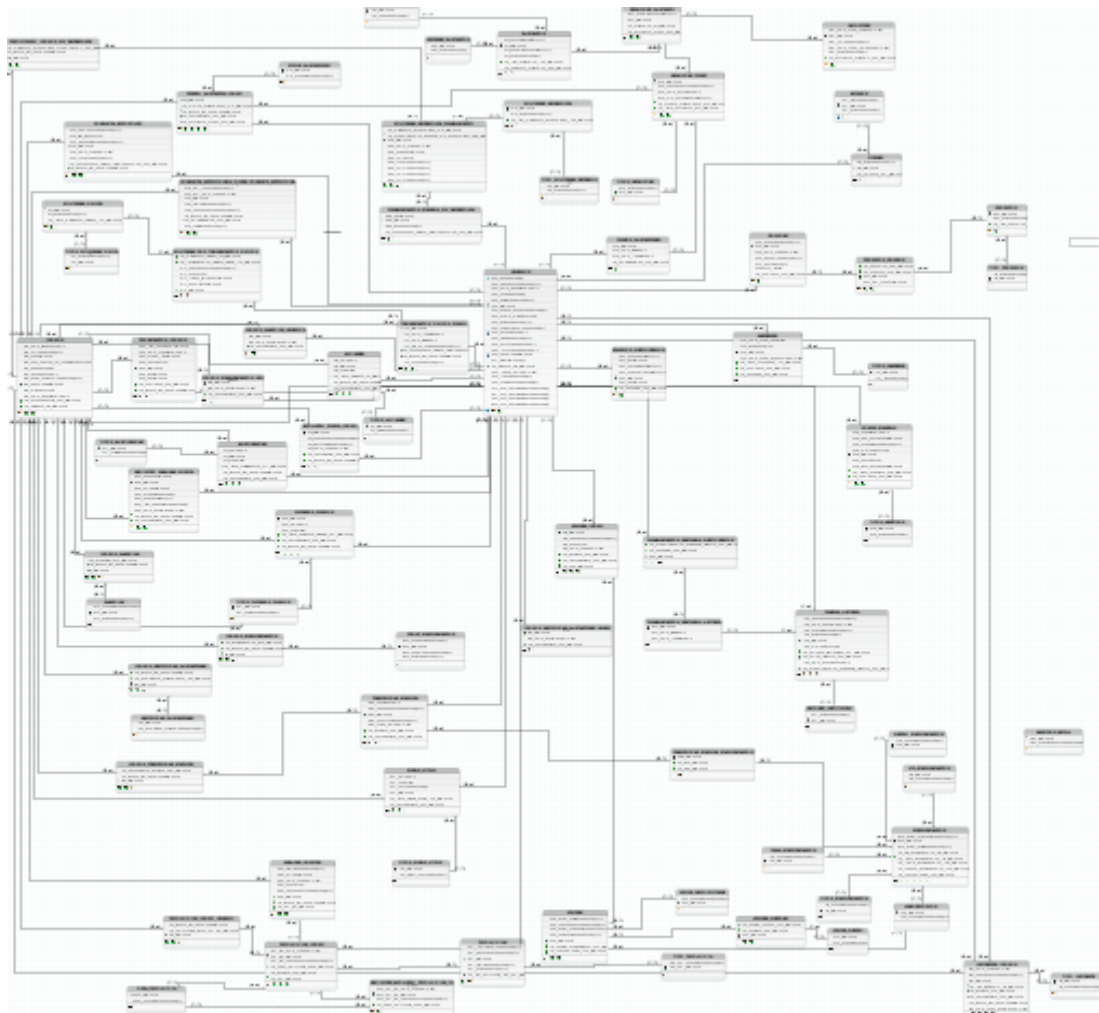
Foi necessário dividi-lo em oito(8) partes pois se trata de um diagrama complexo, que engloba a união de todas as BDs de cada módulo do projeto gerações. O processo de desenvolvimento da integração do banco de dados foi longo, principalmente do modelo conceitual mostrado acima pelo fato de ter sido unido manualmente, havendo a necessidade de correção de acordo com as tabelas semelhantes entre os diagramas.

Para desenvolver essa integração, foi necessário a priori, que todos os módulos estivessem com seu banco de dados finalizados e funcionando corretamente, para que assim os BDAs do projeto gerações, elaborassem em conjunto o banco de dados integrado. Foi utilizado a ferramenta BrModelo para esse processo que será melhor explicado no tópico 2.2.2.

Depois de finalizado a integração do modelo conceitual, foi marcado um encontro com o professor da matéria de PDS para revisar o diagrama a fim de minimizar os erros, depois foi elaborado através desse o modelo relacional (Lógico) a partir de uma execução do brModelo, que gerava os diagramas seguintes a partir do modelo entidade e relacionamento.

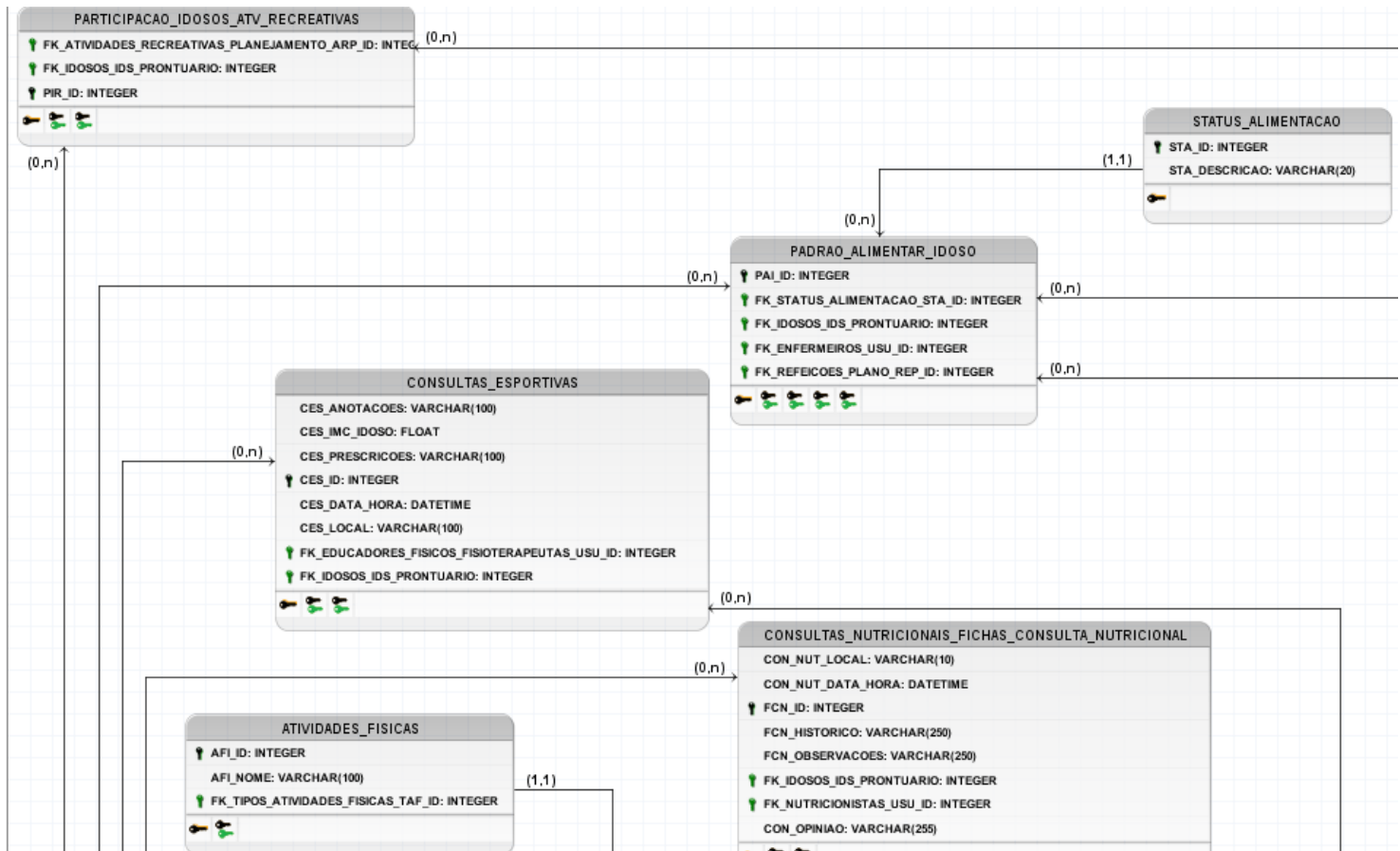
Abaixo será mostrado o atual modelo relacional do banco de dados do projeto gerações.

Figura 17 Modelo lógico banco integrado



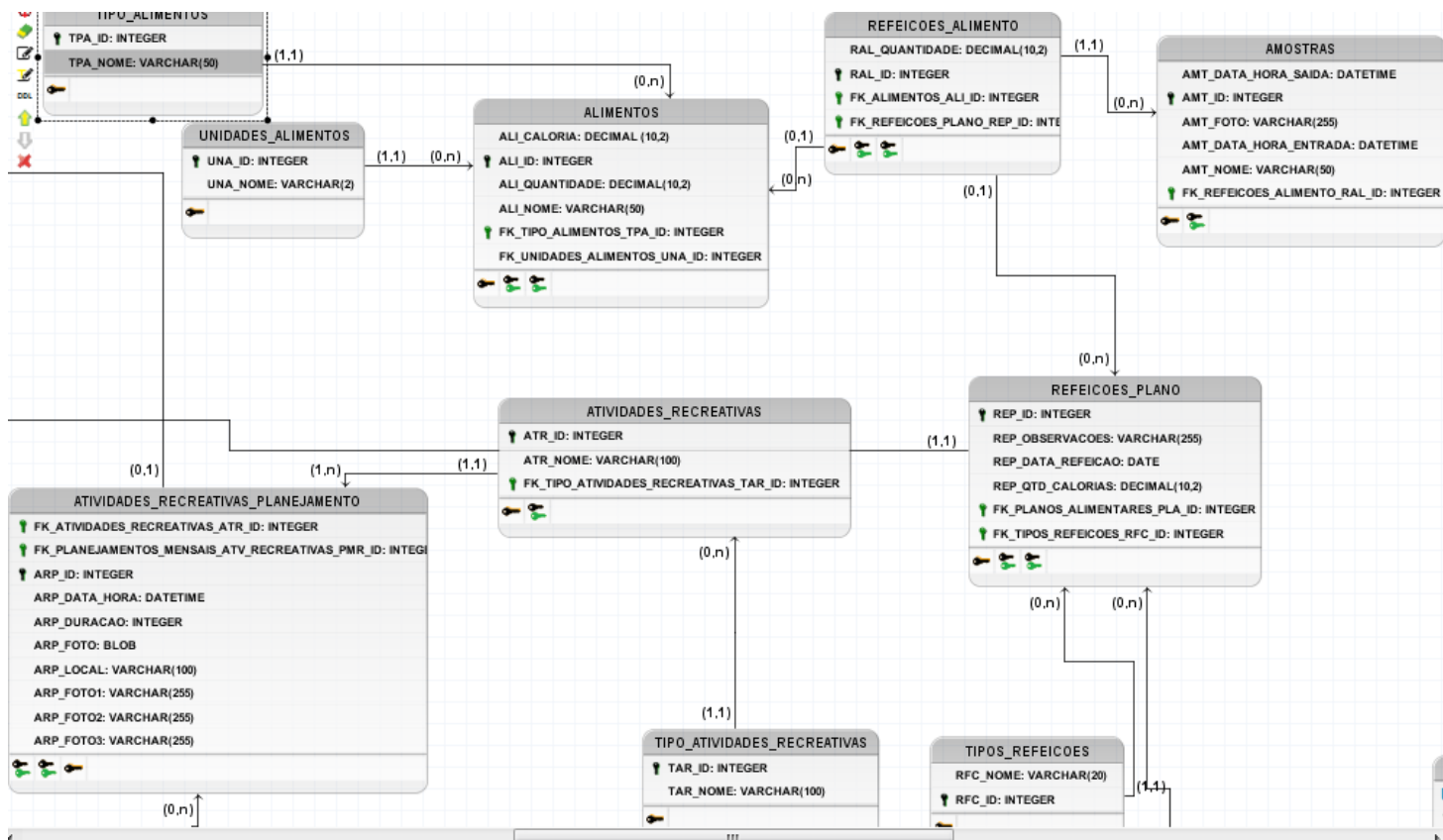
A imagem acima representa o modelo relacional (logico) do banco integrado do projeto gerações, e a seguir será mostrado com ampliação esse diagrama, para uma melhor visualização.

Figura 18 Parte 1 modelo logico



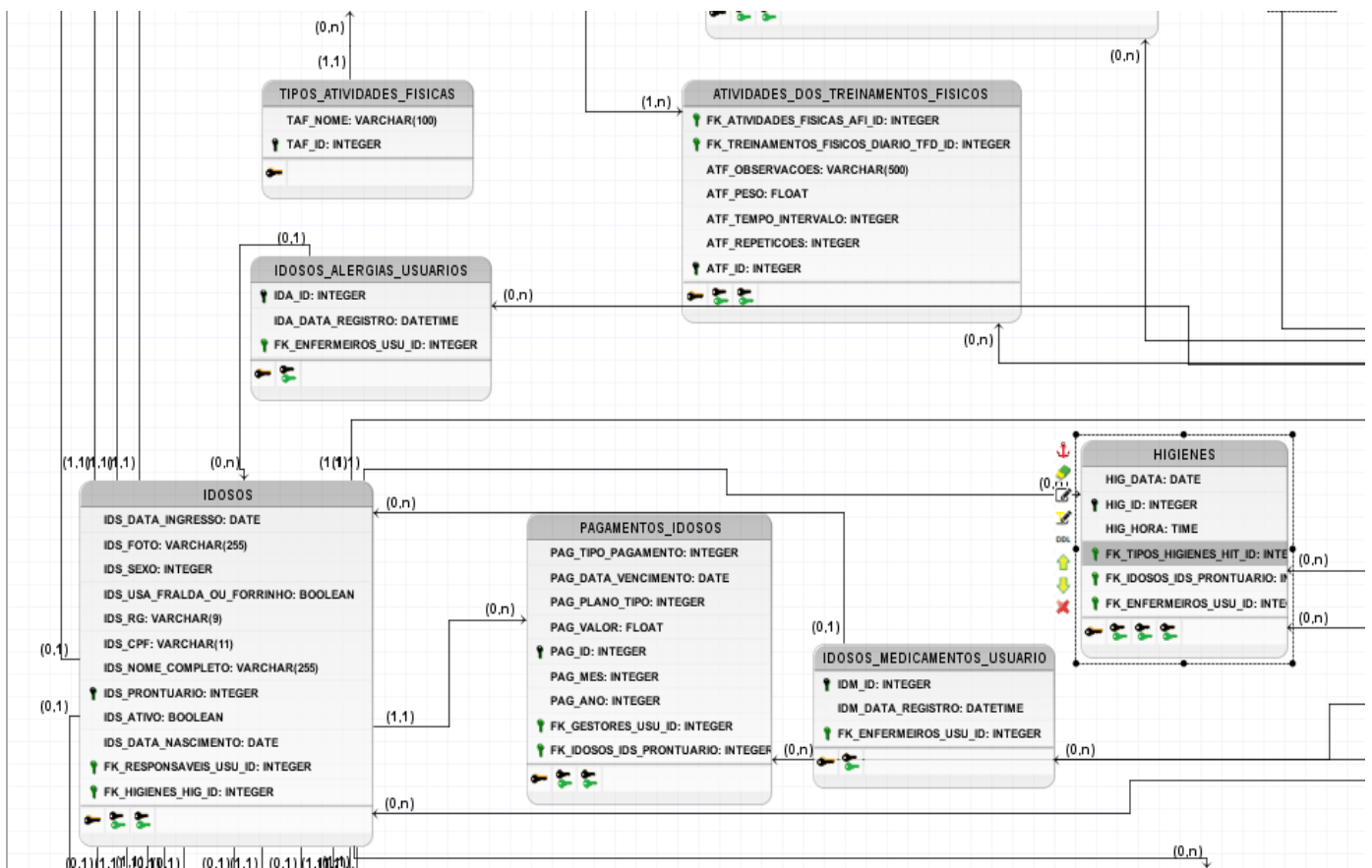
A imagem acima representa as tabelas referentes às atividades físicas e recreativas e nutrição, pertencentes aos módulos 6 e 7.

Figura 19 Parte 2 Modelo lógico



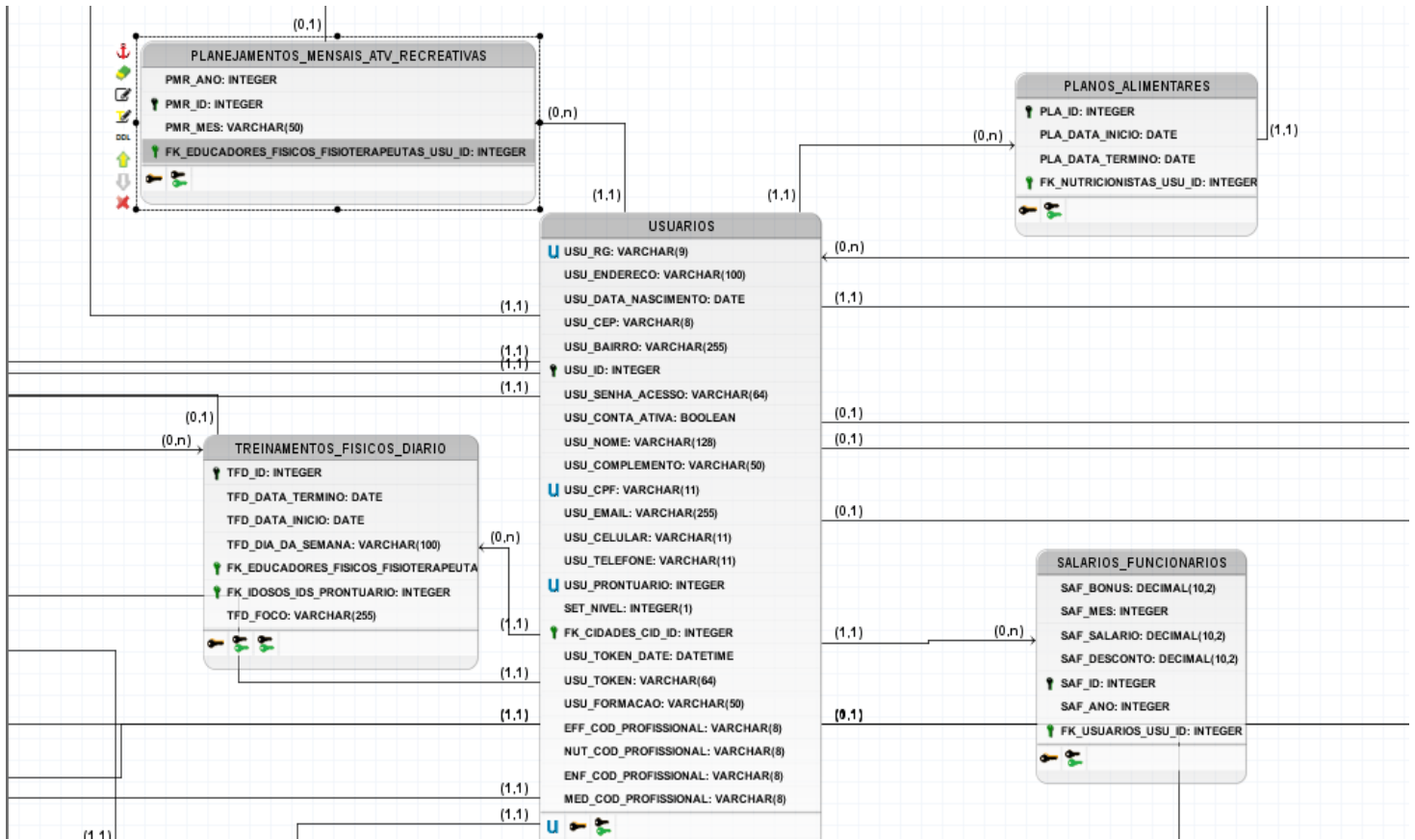
A imagem acima representa outras tabelas referentes às atividades físicas e recreativas e nutrição, pertencentes aos módulos 6 e 7.

Figura 20 Parte 3 Modelo lógico



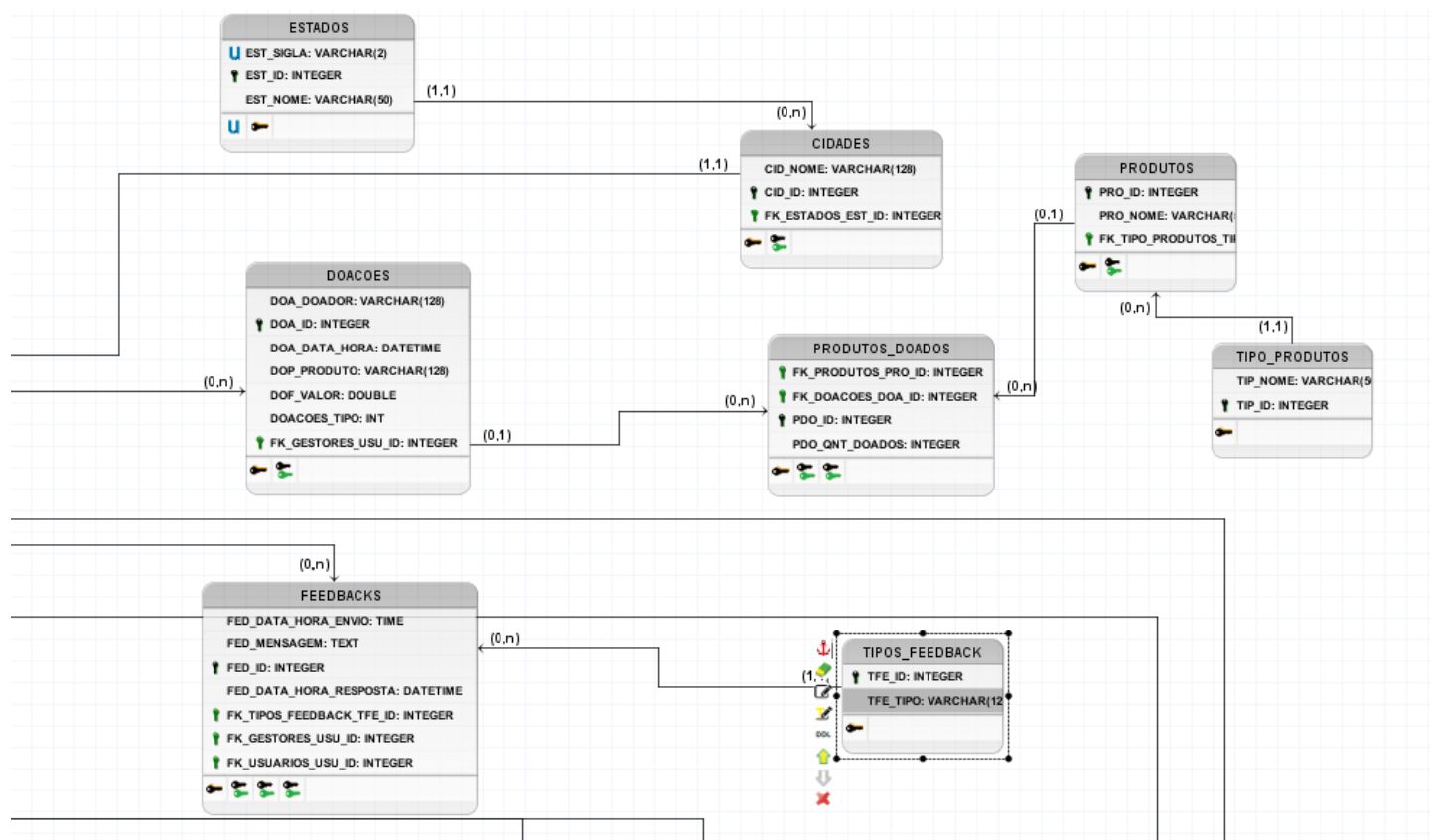
A imagem acima representa as tabelas referentes às atividades físicas e recreativas, prontuários dos idosos, cuidados diários e controle administrativo, pertencentes aos módulos 2,4,5 e 7.

Figura 21 Parte 4 Modelo lógico



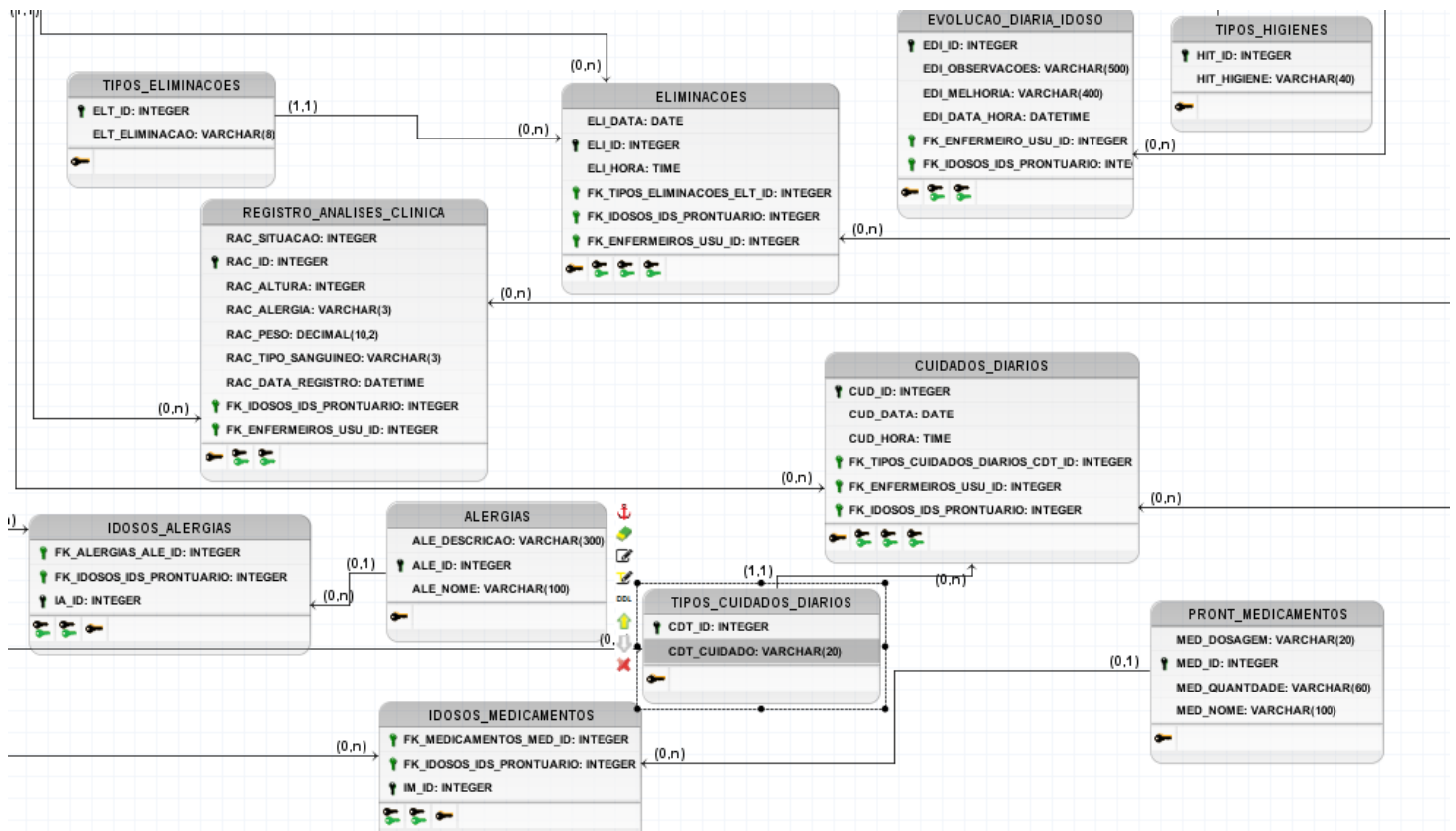
A imagem acima representa as tabelas referentes às atividades físicas e recreativas, nutrição, controle administrativo e usuários pertencentes aos módulos 1, 7 e 8.

Figura 22 Parte 5 Modelo lógico



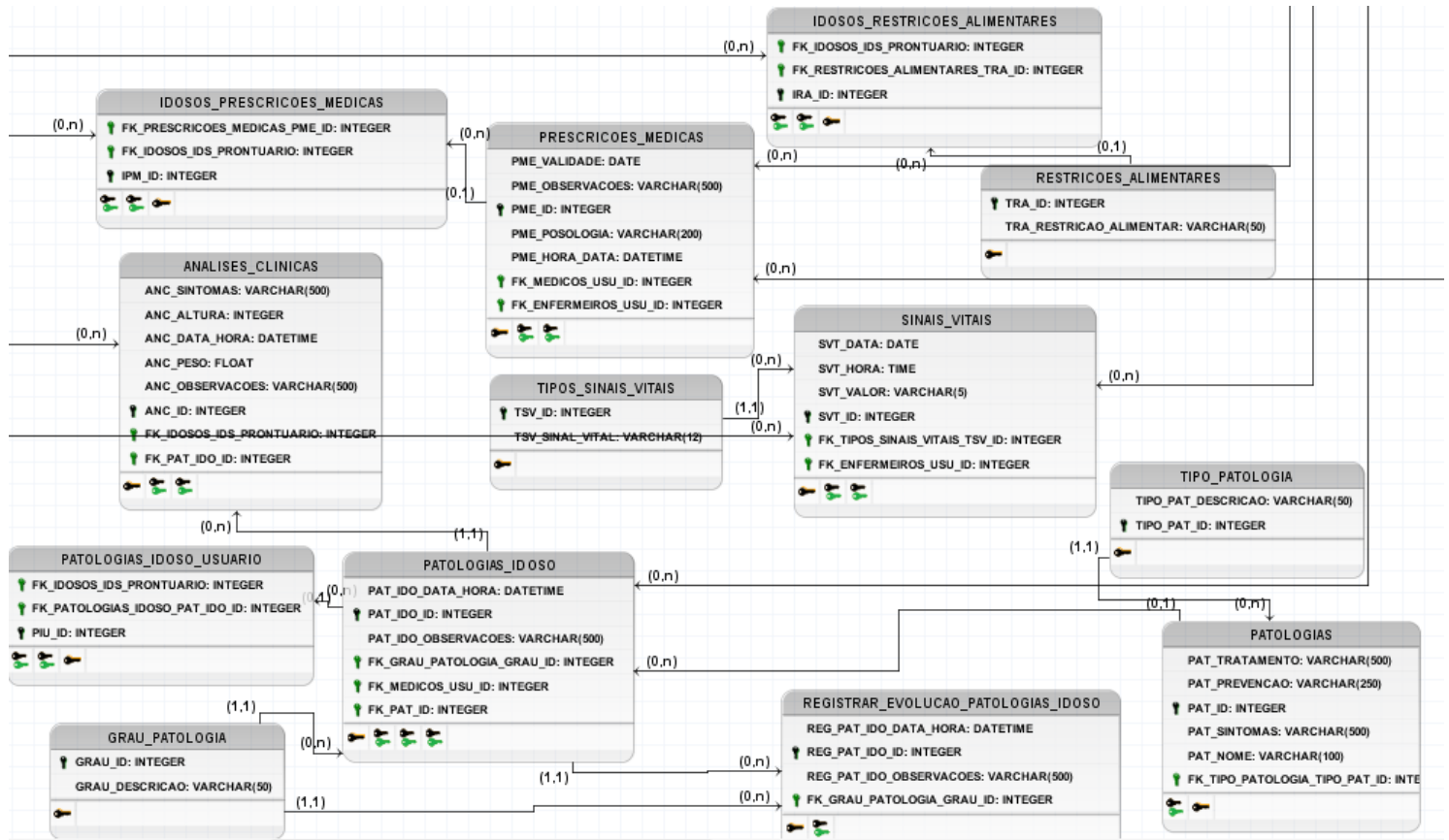
A imagem acima representa as tabelas referentes aos usuários (cadastros) e relatórios pertencentes aos módulos 1 e 9.

Figura 23 Parte 6 Modelo lógico



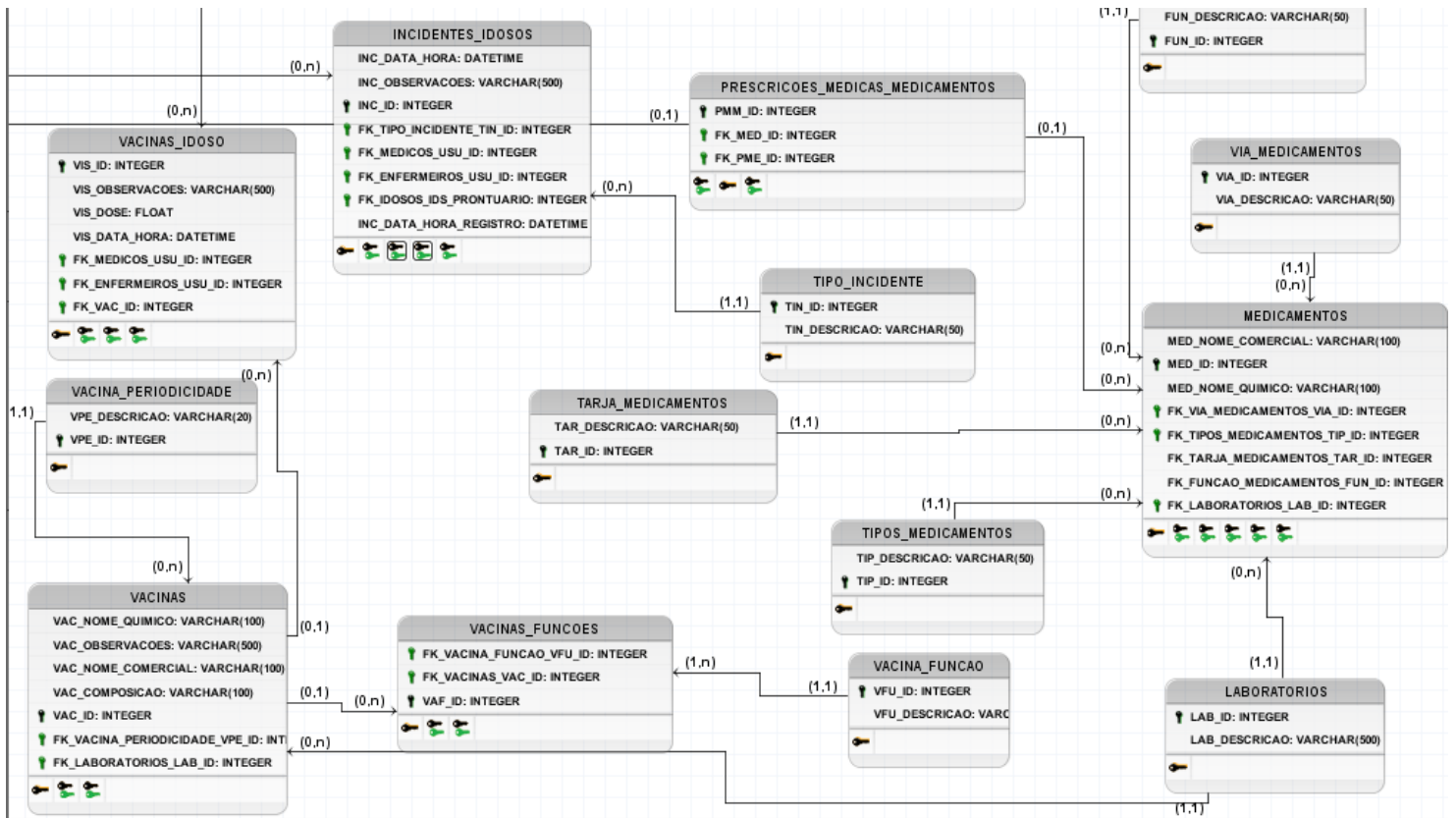
A imagem acima representa as tabelas referentes aos prontuários dos idosos, prescrição médica e cuidados diários pertencentes aos módulos 2, 4 e 5.

Figura 24 Parte 7 Modelo lógico



A imagem acima representa as tabelas referentes aos prontuários dos idosos e prescrição médica pertencente aos módulos 2, 4 e 5.

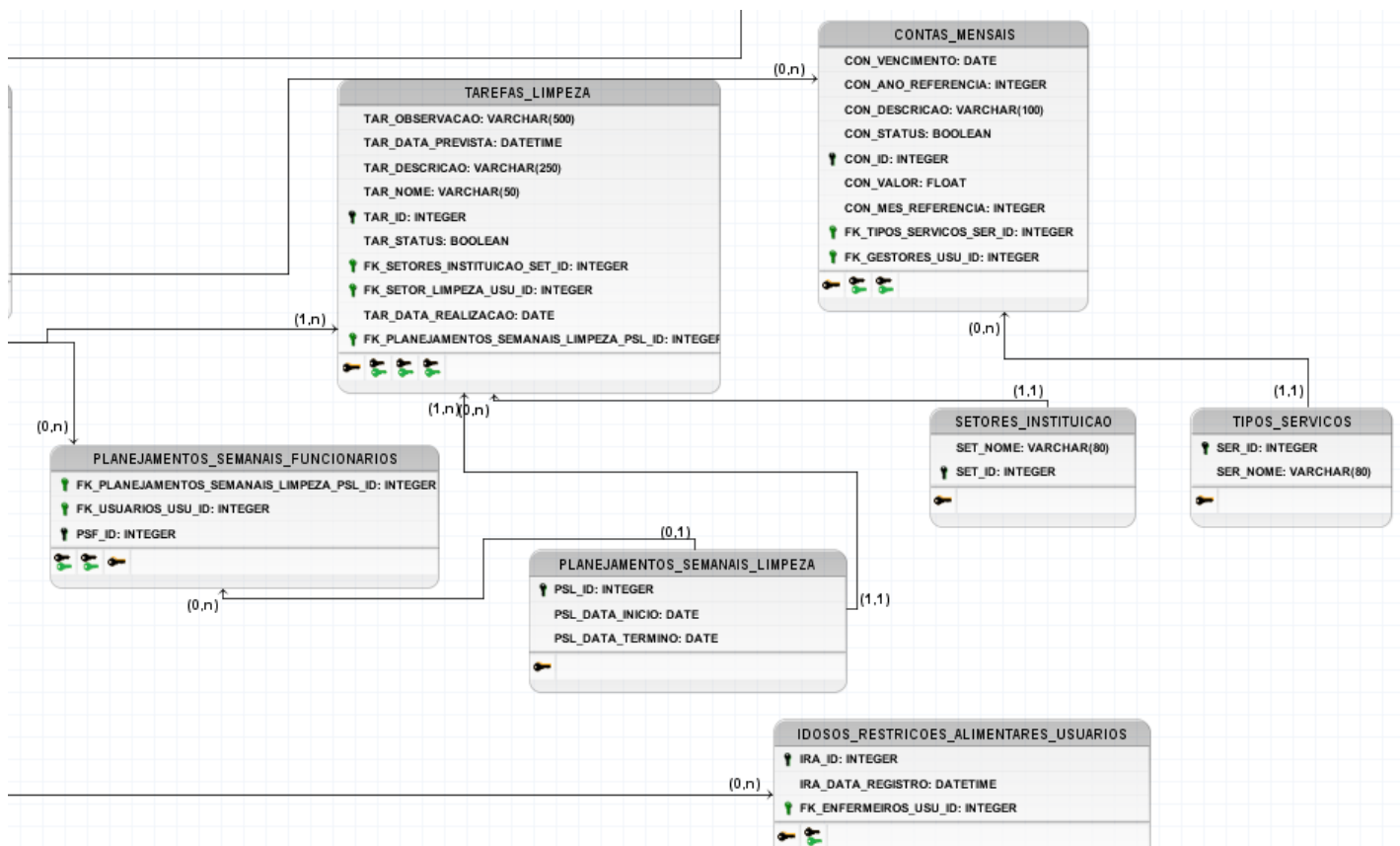
Figura 25 Parte 8 Modelo lógico



A imagem acima representa as tabelas referente à prescrição médica pertencentes ao módulo

5.

Figura 26 Parte 9 Modelo lógico



A imagem acima representa as tabelas referentes ao controle administrativo pertencentes ao módulo 8.

No processo do modelo relacional acima, foi preciso estipular um valor adequado para cada atributo, para assim finalizá-lo e conseguir construir o modelo físico. Quando são necessárias alterações de acordo com o desenvolvimento do projeto, é corrigido no modelo exposto para depois modificar o Script e os inserts.

Foi preciso mostrar o modelo conceitual e lógico para assim poder explicar o processo de desenvolvimento do modelo físico, pois ele só é construído quando os dois diagramas anteriores estiverem finalizados e com seus requisitos preenchidos.

2.2.2 Objetivos específicos 2: Exibição das ferramentas utilizadas no desenvolvimento da atividade

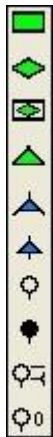
As ferramentas utilizadas no processo de desenvolvimento do banco de dados foram: brModelo, MySQL Workbench e Xampp.

Para os diagramas (conceitual e relacional) foi utilizado o brModelo que é uma ferramenta criada para facilitar a criação de modelos relacionais de um banco de dados. A ferramenta apresenta uma interface muito simples, mas extremamente funcional. A pequena barra de tarefas colocada do lado esquerdo da tela possui todas as ferramentas necessárias para criar entidades, relacionamentos, atributos, etc.

Figura 27 barra de tarefas do brModelo

Essa ferramenta como citado anteriormente, possui uma funcionalidade que permite, a partir de um esquema criado, gerar também outro tipo de modelo. Por exemplo, pode-se criar apenas o modelo conceitual e, a partir dele, gerar um esquema lógico, com todos os atributos, nomes dos relacionamentos e ligações corretamente posicionados.

A grande vantagem em utilizar o brModelo é o fato desse ser extremamente compacto e leve, podendo assim rodar diretamente de pendrives ou qualquer outro dispositivo portátil. Além disso, não é necessário instalá-lo, basta fazer o download e usar [10].



Já para o modelo físico e inserts foi utilizado o MySQL Workbench, é um sistema gerenciador de banco de dados relacional de código aberto usado na maioria das aplicações gratuitas para gerir suas bases de dados. O serviço utiliza a linguagem SQL (Structure Query Language – Linguagem de Consulta Estruturada), que é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados [11].

E o Xampp é um pacote com os principais servidores de código aberto do mercado, incluindo FTP, banco de dados MySQL e Apache com suporte as linguagens PHP e Perl. Com ele, é possível rodar sistemas como WordPress e Drupal localmente, o que facilita e agiliza o desenvolvimento. Como o conteúdo estará armazenado numa rede local, o acesso aos arquivos é realizado instantaneamente. A partir dele que se inicia o apache e o Mysql para o último poder criar uma conexão e funcionar corretamente [12].

2.2.3 Objetivos específicos 3: Processo de desenvolvimento do modelo físico

Nesta etapa definiu-se detalhes técnicos da implementação do banco de dados, por exemplo, a forma como os dados foram armazenados, os scripts para a criação dos objetos no banco de dados (tabelas, visões, colunas, funções,...).

O processo de desenvolvimento desse modelo foi realizado após gerar o script a partir do modelo relacional, utilizando uma ferramenta do brModelo que permite a criação desse modelo, posteriormente, foi criada uma conexão no programa Mysql, com o Xampp ligado, e criou-se um esquema para processar o arquivo sql. Logo após em uma ação da ferramenta que executava o Script, criou-se as tabelas no banco de dados. Esta etapa é fortemente ligada ao SGBD que foi utilizado. A otimização de desempenho do banco de dados foi trabalhada nesta fase do projeto.

Para o bom funcionamento dessa etapa, foi necessário que o modelo lógico estivesse com suas informações corretas, para no modelo físico criar-se as tabelas sem erros que prejudicariam o desenvolvimento do projeto.

A seguir será mostrado o atual modelo físico do banco de dados integrado do projeto gerações.

Figura 28 Parte 1 modelo físico

```
/* Lógico_ATUALIZADO: */
CREATE DATABASE IF NOT EXISTS geracoes;
USE geracoes;

DROP DATABASE IF EXISTS geracoes;
CREATE DATABASE geracoes;

USE geracoes;

CREATE TABLE USUARIOS (
    USU_RG VARCHAR(9) NOT NULL,
    USU_ENDERECO VARCHAR(100) NOT NULL,
    USU_DATA_NASCIMENTO DATE NOT NULL,
    USU_CEP VARCHAR(8) NOT NULL,
    USU_BAIRRO VARCHAR(255) NOT NULL,
    USU_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    USU_SENHA_ACESSO VARCHAR(64) NOT NULL,
    USU_CONTA_ATIVA BOOLEAN NOT NULL,
    USU_NOME VARCHAR(128) NOT NULL,
    USU_COMPLEMENTO VARCHAR(50) NOT NULL,
    USU_CPF VARCHAR(11) NOT NULL,
    USU_EMAIL VARCHAR(255) NOT NULL,
    USU_CELULAR VARCHAR(11),
    USU_TELEFONE VARCHAR(11),
    USU_PRONTUARIO INTEGER NOT NULL,
    USU_FUNCAO INTEGER NOT NULL,
    SET_NIVEL INTEGER(1) NULL,
    FK_CIDADES_CID_ID INTEGER,
    USU_TOKEN_DATE DATETIME NULL,
    USU_TOKEN VARCHAR(64) NULL,
    USU_FORMACAO VARCHAR(256),
    EFF_COD_PROFISSIONAL VARCHAR(8) NULL,
    NUT_COD_PROFISSIONAL VARCHAR(8) NULL,
    ENF_COD_PROFISSIONAL VARCHAR(8) NULL,
    MED_COD_PROFISSIONAL VARCHAR(8) NULL,
    UNIQUE (USU_RG, USU_CPF, USU_PRONTUARIO, MED_COD_PROFISSIONAL, ENF_COD_PROFISSIONAL, NUT_COD_PROFISSIONAL)
);

CREATE TABLE CIDADES (
    CID_NOME VARCHAR(128) NOT NULL,
    CID_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FK_ESTADOS_EST_ID INTEGER
);

CREATE TABLE ESTADOS (
    EST_SIGLA VARCHAR(2) NOT NULL UNIQUE,
    EST_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    EST_NOME VARCHAR(50) NOT NULL
);
```

Essa imagem representa as primeiras tabelas do script do banco de dados integrado, todas referentes ao módulo 1 – usuários.

Figura 29 Parte 2 modelo físico

```

CREATE TABLE DOACOES (
    DOA_DOADOR VARCHAR(128),
    DOA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    DOA_DATA_HORA DATETIME NOT NULL,
    DOP_PRODUTO VARCHAR(128),
    DOF_VALOR DOUBLE,
    DOACOES_TIPO INT NOT NULL,
    FK_GESTORES_USU_ID INTEGER
);

CREATE TABLE TIPOS_FEEDBACK (
    TFE_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    TFE_TIPO VARCHAR(128) NOT NULL
);

CREATE TABLE FEEDBACKS (
    FED_DATA_HORA_ENVIO TIME NOT NULL,
    FED_MENSAGEM TEXT,
    FED_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FED_DATA_HORA_RESPOSTA DATETIME NOT NULL,
    FK_TIPOS_FEEDBACK_TFE_ID INTEGER,
    FK_GESTORES_USU_ID INTEGER,
    FK_USUARIOS_USU_ID INTEGER
);

CREATE TABLE IDOSOS (
    IDS_DATA_INGRESSO DATE NOT NULL,
    IDS_FOTO VARCHAR(255) NOT NULL,
    IDS_SEXO INTEGER NOT NULL,
    IDS_USA_FRALDA_OU_FORRINHO BOOLEAN NOT NULL,
    IDS_RG VARCHAR(9) NOT NULL,
    IDS_CPF VARCHAR(11) NOT NULL,
    IDS_NOME_COMPLETO VARCHAR(255) NOT NULL,
    IDS_PRONTUARIO INTEGER AUTO_INCREMENT PRIMARY KEY,
    IDS_ATIVO BOOLEAN NOT NULL,
    IDS_DATA_NASCIMENTO DATE NOT NULL,
    FK_RESPONSAVEIS_USU_ID INTEGER NOT NULL,
    FK_HIGIENES_HIG_ID INTEGER
);

CREATE TABLE VACINA_PERIODICIDADE (
    VPE_DESCRICAO VARCHAR(20),
    VPE_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE VACINAS_IDOSO (
    VIS_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    VIS_OBSERVACOES VARCHAR(500),
    VIS_DOSE FLOAT NOT NULL,
    VIS_DATA_HORA DATETIME NOT NULL,
    FK_MEDICOS_USU_ID INTEGER,
    FK_ENFERMEIROS_USU_ID INTEGER,
    FK_VAC_ID INTEGER
);

```

A imagem acima apresenta tabelas de módulos distintos como o módulo 5 – Prescrições médicas, módulo 2 - prontuários dos idosos, modulo 1 – usuários e módulo 9 – relatórios.

Figura 30 Parte 3 modelo físico

```
CREATE TABLE VACINA_FUNCAO (  
    VFU_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    VFU_DESCRICAO VARCHAR(50)  
);  
  
CREATE TABLE VACINAS (  
    VAC_NOME_QUIMICO VARCHAR(100) NOT NULL,  
    VAC_OBSERVACOES VARCHAR(500),  
    VAC_NOME_COMERCIAL VARCHAR(100) NOT NULL,  
    VAC_COMPOSICAO VARCHAR(100) NOT NULL,  
    VAC_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    FK_VACINA_PERIODICIDADE_VPE_ID INTEGER,  
    FK_LABORATORIOS_LAB_ID INTEGER  
);  
  
CREATE TABLE LABORATORIOS (  
    LAB_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    LAB_DESCRICAO VARCHAR(500) NOT NULL  
);  
  
CREATE TABLE VIA_MEDICAMENTOS (  
    VIA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    VIA_DESCRICAO VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE TIPOS_MEDICAMENTOS (  
    TIP_DESCRICAO VARCHAR(50) NOT NULL,  
    TIP_ID INTEGER AUTO_INCREMENT PRIMARY KEY  
);  
  
CREATE TABLE MEDICAMENTOS (  
    MED_NOME_COMERCIAL VARCHAR(100) NOT NULL,  
    MED_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    MED_NOME_QUIMICO VARCHAR(100) NOT NULL,  
    FK_VIA_MEDICAMENTOS_VIA_ID INTEGER,  
    FK_TIPOS_MEDICAMENTOS_TIP_ID INTEGER,  
    FK_TARJA_MEDICAMENTOS_TAR_ID INTEGER,  
    FK_FUNCAO_MEDICAMENTOS_FUN_ID INTEGER,  
    FK_LABORATORIOS_LAB_ID INTEGER  
);  
  
CREATE TABLE FUNCAO_MEDICAMENTOS (  
    FUN_DESCRICAO VARCHAR(50) NOT NULL,  
    FUN_ID INTEGER AUTO_INCREMENT PRIMARY KEY  
);
```

Essa imagem representa a criação de tabelas referentes ao módulo 5 – prescrições médicas.

Figura 31 Parte 4 Modelo físico

```
CREATE TABLE TARJA_MEDICAMENTOS (  
    TAR_DESCRICAO VARCHAR(50) NOT NULL,  
    TAR_ID INTEGER AUTO_INCREMENT PRIMARY KEY  
);  
  
CREATE TABLE INCIDENTES_IDOSOS (  
    INC_DATA_HORA DATETIME NOT NULL,  
    INC_OBSERVACOES VARCHAR(500),  
    INC_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    FK_TIPO_INCIDENTE_TIN_ID INTEGER,  
    FK_MEDICOS_USU_ID INTEGER,  
    FK_ENFERMEIROS_USU_ID INTEGER,  
    FK_IDOSOS_IDS_PRONTUARIO INTEGER  
);  
  
CREATE TABLE TIPO_INCIDENTE (  
    TIN_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    TIN_DESCRICAO VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE PRESCRICOES_MEDICAS (  
    PME_VALIDADE DATE NOT NULL,  
    PME_OBSERVACOES VARCHAR(500),  
    PME_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    PME_POSOLOGIA VARCHAR(200) NOT NULL,  
    PME_HORA_DATA DATETIME NOT NULL,  
    FK_MEDICOS_USU_ID INTEGER,  
    FK_ENFERMEIROS_USU_ID INTEGER  
);  
  
CREATE TABLE PATOLOGIAS_IDOSO (  
    PAT_IDO_DATA_HORA DATETIME NOT NULL,  
    PAT_IDO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    PAT_IDO_OBSERVACOES VARCHAR(500),  
    FK_GRAU_PATOLOGIA_GRAU_ID INTEGER,  
    FK_MEDICOS_USU_ID INTEGER,  
    FK_ANALISES_CLINICAS_ANC_ID INTEGER,  
    FK_PAT_ID INTEGER  
);  
  
CREATE TABLE GRAU_PATOLOGIA (  
    GRAU_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    GRAU_DESCRICAO VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE PATOLOGIAS (  
    PAT_TRATAMENTO VARCHAR(500) NOT NULL,  
    PAT_PREVENCAO VARCHAR(250),  
    PAT_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    PAT_SINTOMAS VARCHAR(500) NOT NULL,  
    PAT_NOME VARCHAR(100) NOT NULL,  
    FK_TIPO_PATOLOGIA_TIPO_PAT_ID INTEGER  
);
```

A imagem acima representa a criação de tabelas referente ao módulo 5 – prescrições médicas.

Figura 32 Parte 5 Modelo físico

```

CREATE TABLE ANALISES_CLINICAS (
    ANC_SINTOMAS VARCHAR(500) NOT NULL,
    ANC_ALTURA INTEGER NOT NULL,
    ANC_DATA_HORA DATETIME NOT NULL,
    ANC_PESO FLOAT NOT NULL,
    ANC_OBSERVACOES VARCHAR(500),
    ANC_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FK_IDSOS_IDS_PRONTUARIO INTEGER,
    FK_PAT_IDO_ID INTEGER
);

CREATE TABLE TIPO_PATOLOGIA (
    TIPO_PAT_DESCRICAO VARCHAR(50) NOT NULL,
    TIPO_PAT_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE SINAIS_VITAIS (
    SVT_DATA DATE NOT NULL,
    SVT_HORA TIME NOT NULL,
    SVT_VALOR VARCHAR(5) NOT NULL,
    SVT_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FK_TIPOS_SINAIS_VITAIS_TSV_ID INTEGER,
    FK_IDSOS_IDS_PRONTUARIO INTEGER,
    FK_ENFERMEIROS_USU_ID INTEGER
);

CREATE TABLE TIPOS_SINAIS_VITAIS (
    TSV_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    TSV_SINAL_VITAL VARCHAR(12) NOT NULL
);

CREATE TABLE TIPOS_HIGIENES (
    HIT_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    HIT_HIGIENE VARCHAR(40) NOT NULL
);

CREATE TABLE HIGIENES (
    HIG_DATA DATE NOT NULL,
    HIG_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    HIG_HORA TIME NOT NULL,
    FK_TIPOS_HIGIENES_HIT_ID INTEGER,
    FK_IDSOS_IDS_PRONTUARIO INTEGER,
    FK_ENFERMEIROS_USU_ID INTEGER
);

CREATE TABLE TIPOS_CUIDADOS_DIARIOS (
    CDT_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    CDT_CUIDADO VARCHAR(20) NOT NULL
);

```

A imagem acima apresenta tabelas referentes ao módulo 4 – cuidados diários e ao módulo 5 – prescrições médicas.

Figura 33 Parte 6 Modelo Físico

```

CREATE TABLE CUIDADOS_DIARIOS (
    CUD_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    CUD_DATA DATE NOT NULL,
    CUD_HORA TIME NOT NULL,
    FK_TIPOS_CUIDADOS_DIARIOS_CDT_ID INTEGER,
    FK_ENFERMEIROS_USU_ID INTEGER,
    FK_IDSOS_IDS_PRONTUARIO INTEGER
);

CREATE TABLE TIPOS_ELIMINACOES (
    ELT_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    ELT_ELIMINACAO VARCHAR(8) NOT NULL
);

CREATE TABLE ELIMINACOES (
    ELI_DATA DATE NOT NULL,
    ELI_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    ELI_HORA TIME NOT NULL,
    FK_TIPOS_ELIMINACOES_ELT_ID INTEGER,
    FK_IDSOS_IDS_PRONTUARIO INTEGER,
    FK_ENFERMEIROS_USU_ID INTEGER
);

CREATE TABLE RESTRICOES_ALIMENTARES (
    TRA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    TRA_RESTRICAO_ALIMENTAR VARCHAR(50) NOT NULL
);

CREATE TABLE IDSOS_RESTRICOES_ALIMENTARES_USUARIOS (
    IRA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    IRA_DATA_REGISTRO DATETIME NOT NULL,
    FK_ENFERMEIROS_USU_ID INTEGER
);

CREATE TABLE ALERGIAS (
    ALE_DESCRICAO VARCHAR(300),
    ALE_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    ALE_NOME VARCHAR(100) NOT NULL
);

CREATE TABLE IDSOS_ALERGIAS_USUARIOS (
    IDA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    IDA_DATA_REGISTRO DATETIME NOT NULL,
    FK_ENFERMEIROS_USU_ID INTEGER
);

CREATE TABLE REGISTRO_ANALISES_CLINICA (
    RAC_SITUACAO INTEGER NOT NULL,
    RAC_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    RAC_ALTURA INTEGER NOT NULL,
    RAC_ALERGIA VARCHAR(3),
    RAC_PESO DECIMAL(10,2) NOT NULL,
    RAC_TIPO_SANGUINEO VARCHAR(3) NOT NULL,
    RAC_DATA_REGISTRO DATETIME,
    FK_IDSOS_IDS_PRONTUARIO INTEGER,
    FK_ENFERMEIROS_USU_ID INTEGER
);

```

A imagem acima apresenta tabelas referentes ao módulo 4 – cuidados diários e ao módulo 2 – prontuário dos idosos.

Figura 34 Parte 7 Modelo físico

```

CREATE TABLE PRONT_MEDICAMENTOS (
    MED_DOSAGEM VARCHAR(20) NOT NULL,
    MED_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    MED_QUANTIDADE VARCHAR(60) NOT NULL,
    MED_NOME VARCHAR(100) NOT NULL
);

CREATE TABLE IDOSOS_MEDICAMENTOS_USUARIO (
    IDM_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    IDM_DATA_REGISTRO DATETIME NOT NULL,
    FK_ENFERMEIROS_USU_ID INTEGER
);

CREATE TABLE CONSULTAS_NUTRICIONAIS_FICHAS_CONSULTA_NUTRICIONAL (
    CON_NUT_LOCAL VARCHAR(10),
    CON_NUT_DATA_HORA DATETIME NOT NULL,
    FCN_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FCN_HISTORICO VARCHAR(250) NOT NULL,
    FCN_OBSERVACOES VARCHAR(250),
    FK_IDOSOS_IDS_PRONTUARIO INTEGER,
    FK_NUTRICIONISTAS_USU_ID INTEGER,
    CON_OPINIAO VARCHAR(255)
);

CREATE TABLE PADRAO_ALIMENTAR_IDOSO (
    PAI_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FK_STATUS_ALIMENTACAO_STA_ID INTEGER,
    FK_IDOSOS_IDS_PRONTUARIO INTEGER,
    FK_ENFERMEIROS_USU_ID INTEGER,
    FK_REFEICOES_PLANO_REP_ID INTEGER
);

CREATE TABLE STATUS_ALIMENTACAO (
    STA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    STA_DESCRICAO VARCHAR(20) NOT NULL
);

CREATE TABLE REFEICOES_PLANO (
    REP_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    REP_OBSERVACOES VARCHAR(255),
    REP_DATA_REFEICAO DATE NOT NULL,
    REP_QTD_CALORIAS DECIMAL(10,2),
    FK_PLANOS_ALIMENTARES_PLA_ID INTEGER,
    FK_TIPOS_REFEICOES_RFC_ID INTEGER
);

CREATE TABLE PLANOS_ALIMENTARES (
    PLA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    PLA_DATA_INICIO DATE NOT NULL,
    PLA_DATA_TERMINO DATE NOT NULL,
    FK_NUTRICIONISTAS_USU_ID INTEGER
);

```

A imagem acima apresenta tabelas referentes ao módulo 2 – prontuários dos idosos e ao módulo 6 – Nutrição.

Figura 35 Parte 8 Modelo físico

```
CREATE TABLE ALIMENTOS (  
    ALI_CALORIA DECIMAL (10,2) NOT NULL,  
    ALI_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    ALI_QUANTIDADE DECIMAL(10,2) NOT NULL,  
    ALI_NOME VARCHAR(50) NOT NULL,  
    FK_TIPO_ALIMENTOS_TPA_ID INTEGER,  
    FK_UNIDADES_ALIMENTOS_UNA_ID INTEGER  
);  
  
CREATE TABLE TIPO_ALIMENTOS (  
    TPA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    TPA_NOME VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE UNIDADES_ALIMENTOS (  
    UNA_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    UNA_NOME VARCHAR(2) NOT NULL  
);  
  
CREATE TABLE TIPOS_REFEICOES (  
    RFC_NOME VARCHAR(20) NOT NULL,  
    RFC_ID INTEGER AUTO_INCREMENT PRIMARY KEY  
);  
  
CREATE TABLE AMOSTRAS (  
    AMT_DATA_HORA_SAIDA DATETIME NOT NULL,  
    AMT_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    AMT_FOTO VARCHAR(255) NOT NULL,  
    AMT_DATA_HORA_ENTRADA DATETIME NOT NULL,  
    AMT_NOME VARCHAR(50) NOT NULL,  
    FK_REFEICOES_ALIMENTO_RAL_ID INTEGER  
);  
  
CREATE TABLE CONTAS_MENSAIS (  
    CON_VENCIMENTO DATE NOT NULL,  
    CON_ANO_REFERENCIA INTEGER NOT NULL,  
    CON_DESCRICAO VARCHAR(100),  
    CON_STATUS BOOLEAN NOT NULL,  
    CON_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    CON_VALOR FLOAT NOT NULL,  
    CON_MES_REFERENCIA INTEGER NOT NULL,  
    FK_TIPOS_SERVICOS_SER_ID INTEGER,  
    FK_GESTORES_USU_ID INTEGER  
);  
  
CREATE TABLE TIPOS_SERVICOS (  
    SER_ID INTEGER AUTO_INCREMENT PRIMARY KEY,  
    SER_NOME VARCHAR(80) NOT NULL  
);
```

A imagem acima apresenta tabelas referentes ao módulo 6 – Nutrição e ao módulo 8 controle administrativo.

Figura 36 Parte 9 Modelo físico

```
CREATE TABLE PAGAMENTOS_IDOSOS (
    PAG_TIPO_PAGAMENTO INTEGER NOT NULL,
    PAG_DATA_VENCIMENTO DATE NOT NULL,
    PAG_PLANO_TIPO INTEGER NOT NULL,
    PAG_VALOR FLOAT NOT NULL,
    PAG_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    PAG_MES INTEGER NOT NULL,
    PAG_ANO INTEGER NOT NULL,
    FK_GESTORES_USU_ID INTEGER,
    FK_IDOSOS_IDS_PRONTUARIO INTEGER
);

CREATE TABLE SALARIOS_FUNCIONARIOS (
    SAF_BONUS DECIMAL(10,2),
    SAF_MES INTEGER,
    SAF_SALARIO DECIMAL(10,2) NOT NULL,
    SAF_DESCONTO DECIMAL(10,2),
    SAF_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    SAF_ANO INTEGER NOT NULL,
    FK_USUARIOS_USU_ID INTEGER
);

CREATE TABLE TAREFAS_LIMPEZA (
    TAR_OBSERVACAO VARCHAR(500),
    TAR_DATA_PREVISTA DATETIME NOT NULL,
    TAR_DESCRICAO VARCHAR(250),
    TAR_NOME VARCHAR(50) NOT NULL,
    TAR_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    TAR_STATUS BOOLEAN NOT NULL,
    FK_SETORES_INSTITUICAO_SET_ID INTEGER,
    FK_SETOR_LIMPEZA_USU_ID INTEGER,
    TAR_DATA_REALIZACAO DATE NOT NULL,
    FK_PLANEJAMENTOS_SEMANAIS_LIMPEZA_PSL_ID INTEGER
);

CREATE TABLE SETORES_INSTITUICAO (
    SET_NOME VARCHAR(80) NOT NULL,
    SET_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE PLANEJAMENTOS_SEMANAIS_LIMPEZA (
    PSL_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    PSL_DATA_INICIO DATE NOT NULL,
    PSL_DATA_TERMINO DATE NOT NULL
);

CREATE TABLE TIPO_PRODUTOS (
    TIP_NOME VARCHAR(50) NOT NULL,
    TIP_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE PRODUTOS (
    PRO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    PRO_NOME VARCHAR(50) NOT NULL,
    FK_TIPO_PRODUTOS_TIP_ID INTEGER
);
```

A imagem acima apresenta tabelas referentes apenas ao módulo 8 - controle administrativo.

Figura 37 Parte10 Modelo físico

```

CREATE TABLE PLANEJAMENTOS_MENSAIS_ATV_RECREATIVAS (
    PMR_ANO INTEGER NOT NULL,
    PMR_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    PMR_MES VARCHAR(50) NOT NULL,
    FK_EDUCADORES_FISICOS_FISIOTERAPEUTAS_USU_ID INTEGER
);

CREATE TABLE TIPO_ATIVIDADES_RECREATIVAS (
    TAR_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    TAR_NOME VARCHAR(100) NOT NULL
);

CREATE TABLE ATIVIDADES_RECREATIVAS (
    ATR_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    ATR_NOME VARCHAR(100) NOT NULL,
    FK_TIPO_ATIVIDADES_RECREATIVAS_TAR_ID INTEGER
);

CREATE TABLE TREINAMENTOS_FISICOS_DIARIO (
    TFD_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    TFD_DATA_TERMINO DATE NOT NULL,
    TFD_DATA_INICIO DATE NOT NULL,
    TFD_DIA_DA_SEMANA VARCHAR(100) NOT NULL,
    FK_EDUCADORES_FISICOS_FISIOTERAPEUTAS_USU_ID INTEGER,
    FK_IDSOS_IDS_PRONTUARIO INTEGER,
    TFD_FOCO VARCHAR(255)
);

CREATE TABLE ATIVIDADES_FISICAS (
    AFI_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    AFI_NOME VARCHAR(100) NOT NULL,
    FK_TIPOS_ATIVIDADES_FISICAS_TAF_ID INTEGER
);

CREATE TABLE TIPOS_ATIVIDADES_FISICAS (
    TAF_NOME VARCHAR(100) NOT NULL,
    TAF_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE CONSULTAS_ESPORTIVAS (
    CES_ANOTACOES VARCHAR(100),
    CES_IMC_IDOSO FLOAT NOT NULL,
    CES_PRESCRICOES VARCHAR(100) NOT NULL,
    CES_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    CES_DATA_HORA DATETIME NOT NULL,
    CES_LOCAL VARCHAR(100) NOT NULL,
    FK_EDUCADORES_FISICOS_FISIOTERAPEUTAS_USU_ID INTEGER,
    FK_IDSOS_IDS_PRONTUARIO INTEGER
);

CREATE TABLE REFEICOES_ALIMENTO (
    RAL_QUANTIDADE DECIMAL(10,2) NOT NULL,
    RAL_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FK_ALIMENTOS_ALI_ID INTEGER,
    FK_REFEICOES_PLANO_REP_ID INTEGER
);

```

A imagem acima apresenta tabelas referentes ao módulo 7 – atividades físicas e recreativas e ao módulo 6 – nutrição.

Figura 38 Parte 11 modelo físico

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

CREATE TABLE IDOSOS_PRESCRICOES_MEDICAS (
    FK_PRESCRICOES_MEDICAS_PME_ID INTEGER,
    FK_IDOSOS_IDS_PRONTUARIO INTEGER,
    IPM_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE PATOLOGIAS_IDOSO_USUARIO (
    FK_IDOSOS_IDS_PRONTUARIO INTEGER,
    FK_PATOLOGIAS_IDOSO_PAT_IDO_ID INTEGER,
    PIU_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE PLANEJAMENTOS_SEMANAIS_FUNCIONARIOS (
    FK_PLANEJAMENTOS_SEMANAIS_LIMPEZA_PSL_ID INTEGER,
    FK_USUARIOS_USU_ID INTEGER,
    PSF_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE PRODUTOS_DOADOS (
    FK_PRODUTOS_PRO_ID INTEGER,
    FK_DOACOES_DOA_ID INTEGER,
    PDO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    PDO_QNT_DOADOS INTEGER NOT NULL
);

CREATE TABLE PARTICIPACAO_IDOSOS_ATV_RECREATIVAS (
    FK_ATIVIDADES_RECREATIVAS_PLANEJAMENTO_ARP_ID INTEGER,
    FK_IDOSOS_IDS_PRONTUARIO INTEGER,
    PIR_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);
```

Nessa imagem apresenta tabelas referentes ao modulo 5 - prescrição médica, ao módulo 2 – prontuários dos idosos, ao módulo 7 – atividades físicas e recreativas e ao módulo 9 – relatórios.

Figura 39 Parte 12 modelo físico

```
CREATE TABLE ATIVIDADES_DOS_TREINAMENTOS_FISICOS (
    FK_ATIVIDADES_FISICAS_AFI_ID INTEGER,
    FK_TREINAMENTOS_FISICOS_DIARIO_TFD_ID INTEGER,
    ATF_OBSERVACOES VARCHAR(500),
    ATF_PESO FLOAT NOT NULL,
    ATF_TEMPO_INTERVALO INTEGER NOT NULL,
    ATF_REPETICOES INTEGER NOT NULL,
    ATF_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE IDOSOS_RESTRICOES_ALIMENTARES (
    FK_IDOSOS_IDS_PRONTUARIO INTEGER,
    FK_RESTRICOES_ALIMENTARES_TRA_ID INTEGER,
    IRA_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE IDOSOS_ALERGIAS (
    FK_ALERGIAS_ALE_ID INTEGER,
    FK_IDOSOS_IDS_PRONTUARIO INTEGER,
    IA_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);

CREATE TABLE IDOSOS_MEDICAMENTOS (
    FK_MEDICAMENTOS_MED_ID INTEGER,
    FK_IDOSOS_IDS_PRONTUARIO INTEGER,
    IM_ID INTEGER AUTO_INCREMENT PRIMARY KEY
);
```

Nessa imagem apresenta tabelas referentes ao módulo 2 - prontuários dos idosos, ao módulo 7 – atividades físicas e recreativas.

Figura 40 Parte 13 modelo fisico

```
CREATE TABLE ATIVIDADES_RECREATIVAS_PLANEJAMENTO (
    FK_ATIVIDADES_RECREATIVAS_ATR_ID INTEGER,
    FK_PLANEJAMENTOS_MENSAIS_ATV_RECREATIVAS_PMR_ID INTEGER,
    ARP_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    ARP_DATA_HORA DATETIME,
    ARP_DURACAO INTEGER,
    ARP_FOTO BLOB,
    ARP_LOCAL VARCHAR(100),
    ARP_FOTO1 VARCHAR(255),
    ARP_FOTO2 VARCHAR(255),
    ARP_FOTO3 VARCHAR(255)
);

CREATE TABLE PRESCRICOES_MEDICAS_MEDICAMENTOS (
    PMM_ID INTEGER PRIMARY KEY,
    FK_MED_ID INTEGER,
    FK_PME_ID INTEGER
);

CREATE TABLE REGISTRAR_EVOLUCAO_PATOLOGIAS_IDOSO (
    REG_PAT_IDO_DATA_HORA DATETIME,
    REG_PAT_IDO_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    REG_PAT_IDO_OBSERVACOES VARCHAR(500),
    FK_GRAU_PATOLOGIA_GRAU_ID INTEGER
);
```

A imagem acima apresenta tabelas referentes ao módulo 7 – atividades físicas e recreativas e ao módulo 5 – Prescrições médicas.

Não foi citado o módulo 3 – Acompanhamento familiar, pelo fato de apresentar informações contidas em outros módulo pois na união das Dbs, tabelas iguais viraram uma, sem a necessidade de repetição, portanto o modulo 3 está sendo referenciado indiretamente com os módulos 4 e 5 as quais herdam dados.

2.2.4 Objetivos específicos 4: Descrição do processo da população do banco de dados

A priori, para o desenvolvimento da população do banco de dados (INSERTS) foi dividido para os BDAs as tabelas referentes ao seu módulo, para realizar a inserção com valores reais. Após a divisão foi necessário inserir informações apenas das tabelas que não possuem chave estrangeira, para posteriormente realizar a inserção das que possuem “FK” pela necessidade de “pegarem” informações de outras tabelas que necessariamente precisam já estar inseridas no banco de dados, para não acarretar em erros nos inserts.

Foi um processo longo pelo fato citado anteriormente, de que uma tabela só pode ser realizada a inserção, se a outra já estiver inserida, e por conta desse processo houve dificuldades na junção de todos os inserts por conta das divergências dos valores atribuídos para as tabelas com chaves estrangeiras, não possuindo uma ligação e, por conseguinte gerando erros e não inserindo no banco de dados.

Felizmente o problema foi resolvido, finalizando a atividade. A seguir será mostrado alguns inserts do projeto gerações.

Figura 41 Parte 1 Inserts

```
-- TABELA: PRONTI_MEDICAMENTO
INSERT INTO `geracoes`.`pront_medicamentos` (`MED_DOSAGEM`, `MED_QUANTIDADE`, `MED_NOME`) VALUES ('300mg', '1 por dia', 'Propafenona');
INSERT INTO `geracoes`.`pront_medicamentos` (`MED_DOSAGEM`, `MED_QUANTIDADE`, `MED_NOME`) VALUES ('30mg', '1 por dia', 'Atenolol');
INSERT INTO `geracoes`.`pront_medicamentos` (`MED_DOSAGEM`, `MED_QUANTIDADE`, `MED_NOME`) VALUES ('500mg', '1 a cada 6 horas', 'Quetiapina');
INSERT INTO `geracoes`.`pront_medicamentos` (`MED_DOSAGEM`, `MED_QUANTIDADE`, `MED_NOME`) VALUES ('10mg', '2 por dia', 'Dipirona');
INSERT INTO `geracoes`.`pront_medicamentos` (`MED_DOSAGEM`, `MED_QUANTIDADE`, `MED_NOME`) VALUES ('50mg', '1 por dia', 'Bipropena');

-- TABELA: ALERGIAS
INSERT INTO `geracoes`.`alergias` (`ALE_DESCRICAO`, `ALE_NOME`)
VALUES ('Essa forma de dermatite é uma inflamação crônica que surge em áreas da pele que contêm grande número de glândulas sebáceas.', 'Dermatite atópica');
INSERT INTO `geracoes`.`alergias` (`ALE_DESCRICAO`, `ALE_NOME`)
VALUES ('A dermatite atópica, também chamada de eczema atópico, é uma lesão de pele, tipo rash, muito comum em crianças', 'Dermatite seborreica na face. ');
INSERT INTO `geracoes`.`alergias` (`ALE_DESCRICAO`, `ALE_NOME`)
VALUES ('O angioedema é uma quadro perigoso que pode levar à dificuldade respiratória por edema de glote ou asma grave', 'Angioedema');
INSERT INTO `geracoes`.`alergias` (`ALE_DESCRICAO`, `ALE_NOME`)
VALUES ('A urticária é uma lesão de pele, avermelhada, em placas e com relevo, que provoca intensa coceira. ', 'Urticária');
INSERT INTO `geracoes`.`alergias` (`ALE_DESCRICAO`, `ALE_NOME`)
VALUES ('O sintoma mais habitual uma pequena elevação avermelhada na pele com intensa coceira.', 'Picadas de mosquito');

-- TABELA: RESTRICOES_ALIMENTARES
INSERT INTO `geracoes`.`restricoes_alimentares` (`TRA_RESTRICAO_ALIMENTAR`) VALUES ('Ovo');
INSERT INTO `geracoes`.`restricoes_alimentares` (`TRA_RESTRICAO_ALIMENTAR`) VALUES ('Carne');
INSERT INTO `geracoes`.`restricoes_alimentares` (`TRA_RESTRICAO_ALIMENTAR`) VALUES ('Carboidratos');
INSERT INTO `geracoes`.`restricoes_alimentares` (`TRA_RESTRICAO_ALIMENTAR`) VALUES ('Laticínio');
INSERT INTO `geracoes`.`restricoes_alimentares` (`TRA_RESTRICAO_ALIMENTAR`) VALUES ('Peixe');
```

A imagem acima apresenta inserts relacionados ao módulo 2 – Prontuário dos idosos.

Figura 42 Parte 2 inserts

```
-- TABELA: IDOSOS
INSERT INTO `geracoes`.`idosos` (`IDS_DATA_INGRESSO`, `IDS_FOTO`, `IDS_SEXO`, `IDS_USA_FRALDA_OU_FORRINHO`, `IDS_RG`, `IDS_CPF`, `IDS_NOME_COMPLETO`, `IDS_ATIVO`, `IDS_DATA_NASCIMENTO`, `FK_RESPONSAVEIS_USU_ID`) VALUES ('2018-05-14', 'images/Camila1.jpg', 0, 0, '165755696', '76850402633', 'Camila Stefany Yasmin Dias', true, '1943-07-22', 7);
INSERT INTO `geracoes`.`idosos` (`IDS_DATA_INGRESSO`, `IDS_FOTO`, `IDS_SEXO`, `IDS_USA_FRALDA_OU_FORRINHO`, `IDS_RG`, `IDS_CPF`, `IDS_NOME_COMPLETO`, `IDS_ATIVO`, `IDS_DATA_NASCIMENTO`, `FK_RESPONSAVEIS_USU_ID`) VALUES ('2018-01-10', 'images/Catarina2.jpg', 0, 0, '229827007', '29519519300', 'Catarina Jennifer Regina Silva', false, '1947-05-27', 7);
INSERT INTO `geracoes`.`idosos` (`IDS_DATA_INGRESSO`, `IDS_FOTO`, `IDS_SEXO`, `IDS_USA_FRALDA_OU_FORRINHO`, `IDS_RG`, `IDS_CPF`, `IDS_NOME_COMPLETO`, `IDS_ATIVO`, `IDS_DATA_NASCIMENTO`, `FK_RESPONSAVEIS_USU_ID`) VALUES ('2017-02-20', 'images/Thomas3.jpg', 1, 0, '483528651', '82715968108', 'Thomas Luis Carvalho', true, '1949-09-22', 7);
INSERT INTO `geracoes`.`idosos` (`IDS_DATA_INGRESSO`, `IDS_FOTO`, `IDS_SEXO`, `IDS_USA_FRALDA_OU_FORRINHO`, `IDS_RG`, `IDS_CPF`, `IDS_NOME_COMPLETO`, `IDS_ATIVO`, `IDS_DATA_NASCIMENTO`, `FK_RESPONSAVEIS_USU_ID`) VALUES ('2018-09-29', 'images/Raul4.jpg', 1, 1, '196802039', '79725834470', 'Raul Marcos Gonçalves', false, '1949-01-11', 7);
INSERT INTO `geracoes`.`idosos` (`IDS_DATA_INGRESSO`, `IDS_FOTO`, `IDS_SEXO`, `IDS_USA_FRALDA_OU_FORRINHO`, `IDS_RG`, `IDS_CPF`, `IDS_NOME_COMPLETO`, `IDS_ATIVO`, `IDS_DATA_NASCIMENTO`, `FK_RESPONSAVEIS_USU_ID`) VALUES ('2018-09-11', 'images/Francisco5.jpg', 1, 1, '309600832', '08019203702', 'Francisco Erick Julio Novaes', true, '1949-09-03', 7);

-- TABELA: IDOSOS_ALERGIAS_USUARIOS
INSERT INTO `geracoes`.`idosos_alergias_usuarios` (`IDA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-05-14 10:15:11', 3);
INSERT INTO `geracoes`.`idosos_alergias_usuarios` (`IDA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-01-10 11:12:23', 3);
INSERT INTO `geracoes`.`idosos_alergias_usuarios` (`IDA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2017-02-20 17:21:13', 3);
INSERT INTO `geracoes`.`idosos_alergias_usuarios` (`IDA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-09-29 20:15:20', 3);
INSERT INTO `geracoes`.`idosos_alergias_usuarios` (`IDA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-09-11 15:31:44', 3);

-- TABELA: IDOSOS_MEDICAMENTOS_USUARIO
INSERT INTO `geracoes`.`idosos_medicamentos_usuario` (`IDM_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-05-14 10:15:11', 3);
INSERT INTO `geracoes`.`idosos_medicamentos_usuario` (`IDM_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-01-10 11:12:23', 3);
INSERT INTO `geracoes`.`idosos_medicamentos_usuario` (`IDM_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2017-02-20 17:21:13', 3);
INSERT INTO `geracoes`.`idosos_medicamentos_usuario` (`IDM_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-09-29 20:15:20', 3);
INSERT INTO `geracoes`.`idosos_medicamentos_usuario` (`IDM_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-09-11 15:31:44', 3);
```

A imagem acima apresenta inserts relacionado ao módulo 2 – Prontuário dos idosos.

Figura 43 Parte 3 inserts

```
-- TABELA: REGISTRO_ANALISES_CLINICA
INSERT INTO `geracoes`.`registro_analises_clinica` (`RAC_SITUACAO`, `RAC_ALTURA`, `RAC_ALERGIA`, `RAC_PESO`, `RAC_TIPO_SANGUINEO`, `RAC_DATA_REGISTRO`, `FK_IDSOS_IDS_PRONTUARIO`, `FK_ENFERMEIROS_USU_ID`) VALUES (1, 168, 'Sim', 65.2, 'A+', '2018-05-14 10:15:11', 1, 3);
INSERT INTO `geracoes`.`registro_analises_clinica` (`RAC_SITUACAO`, `RAC_ALTURA`, `RAC_ALERGIA`, `RAC_PESO`, `RAC_TIPO_SANGUINEO`, `RAC_DATA_REGISTRO`, `FK_IDSOS_IDS_PRONTUARIO`, `FK_ENFERMEIROS_USU_ID`) VALUES (0, 174, 'Não', 73.4, 'A-', '2018-01-10 11:12:23', 2, 3);
INSERT INTO `geracoes`.`registro_analises_clinica` (`RAC_SITUACAO`, `RAC_ALTURA`, `RAC_ALERGIA`, `RAC_PESO`, `RAC_TIPO_SANGUINEO`, `RAC_DATA_REGISTRO`, `FK_IDSOS_IDS_PRONTUARIO`, `FK_ENFERMEIROS_USU_ID`) VALUES (1, 189, 'Sim', 90.1, 'B+', '2017-02-20 17:21:13', 3, 3);
INSERT INTO `geracoes`.`registro_analises_clinica` (`RAC_SITUACAO`, `RAC_ALTURA`, `RAC_ALERGIA`, `RAC_PESO`, `RAC_TIPO_SANGUINEO`, `RAC_DATA_REGISTRO`, `FK_IDSOS_IDS_PRONTUARIO`, `FK_ENFERMEIROS_USU_ID`) VALUES (0, 173, 'Não', 75.9, 'O+', '2018-09-29 20:15:20', 4, 3);
INSERT INTO `geracoes`.`registro_analises_clinica` (`RAC_SITUACAO`, `RAC_ALTURA`, `RAC_ALERGIA`, `RAC_PESO`, `RAC_TIPO_SANGUINEO`, `RAC_DATA_REGISTRO`, `FK_IDSOS_IDS_PRONTUARIO`, `FK_ENFERMEIROS_USU_ID`) VALUES (1, 172, 'Sim', 68.7, 'O-', '2018-09-11 15:31:44', 5, 3);

-- TABELA: IDOSOS_RESTRICAOES_ALIMENTARES_USUARIOS
INSERT INTO `geracoes`.`idosos_restricoes_alimentares_usuarios` (`IRA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-05-14 10:15:11', 3);
INSERT INTO `geracoes`.`idosos_restricoes_alimentares_usuarios` (`IRA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-01-10 11:12:23', 3);
INSERT INTO `geracoes`.`idosos_restricoes_alimentares_usuarios` (`IRA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2017-02-20 17:21:13', 3);
INSERT INTO `geracoes`.`idosos_restricoes_alimentares_usuarios` (`IRA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-09-29 20:15:20', 3);
INSERT INTO `geracoes`.`idosos_restricoes_alimentares_usuarios` (`IRA_DATA_REGISTRO`, `FK_ENFERMEIROS_USU_ID`) VALUES ('2018-09-11 15:31:44', 3);
```

As três imagens apresentam todos os inserts do módulo 2 – Prontuário dos idosos, contendo relações de chave estrangeira e terceira tabela, conceitos que foram explicados no tópico 2.1.4.2.

Para a visualização dos inserts de todas as tabelas disponibiliza-se o link abaixo do repositório onde está contida a população do banco de dados.

<https://svn.sbv.ifsp.edu.br/svn/pds2019vespertino/documentacao/comum/BANCO%20DE%20DADOS%20INTEGRADO>

3 Conclusões e Recomendações

Esse trabalho teve como objetivo mostrar o desenvolvimento do projeto gerações elaborado ao longo do ano de 2019 pelo 4º ano integrado a informática do Instituto Federal de São Paulo da cidade São João da Boa Vista, a fim de mostrar os benefícios que um software desenvolvido traria para uma instituição de longa permanência que não possui um sistema que auxilie na administração, com a intenção de ajudar a melhorar a qualidade de vida das pessoas na melhor idade.

O trabalho teve como foco principal explicar o processo de desenvolvimento da integração do banco de dados, voltado para o modelo físico e processo da população da DB, uma parte que foi fundamental para o bom funcionamento das funcionalidades do software, e para isso foi preciso elaborar etapas para desenvolver essa atividade.

Para a primeira etapa fez-se necessário descrever conceitos relacionados ao banco de dados, como qual a diferença entre dados e informação, o que é banco de dados, o que são modelos de dados, quais são esse modelos, e outras definições. Foi essencial para a compreensão do trabalho.

A próxima etapa para o desenvolvimento da atividade foi mostrar o modelo conceitual e relacional em partes e seu processo de desenvolvimento, nos quais foram necessários para explicar o objetivo principal.

Na realização da atividade foram utilizadas ferramentas específicas para facilitar no desenvolvimento, e na terceira etapa, foram explicados quais foram esses programas e suas funções, sendo eles: brModelo, Mysql Workbench e Xampp.

Após a terceira etapa, foi descrito um dos objetivos específicos, o processo de desenvolvimento do modelo físico, onde foi explicado o seu processo de desenvolvimento e reforçado com imagens do Script oficial.

Por fim, a última etapa exemplificou a população do banco de dados e mostrou algumas dificuldades encontradas, disponibilizou também, o link onde está todos os inserts do projeto gerações. Essa foi à etapa final na elaboração do banco de dados integrado

Vale ressaltar os pontos negativos que foram encontrados no decorrer do desenvolvimento da atividade. Um dos pontos e o mais prejudicial foi a ineficiência da ferramenta brModelo, seu objetivo de facilitar na modelagem de dados falhou em alguns momentos, quando por exemplo, foi perdido a primeira versão finalizada do modelo conceitual pelo fato do programa não ter salvado as alterações realizadas em nenhuma plataforma, onde acarretou em um atraso da tarefa.

Outro ponto foi à falta de comunicação entre alguns alunos, que prejudicou de certa forma o desenvolvimento do projeto, mas vale lembrar que apesar dos problemas a elaboração do banco de dados integrado foi concluída com dificuldades, mas alcançou o sucesso.

Para trabalhos futuros recomenda-se que reveja ferramentas para serem utilizadas na elaboração de modelagem de dados para não haver perdas ou qualquer outra dificuldade que atrase o desenvolvimento da atividade. Outrossim, seria interessante que os professores da matéria de PDS, para os próximos anos, elaborassem atividades de dinâmicas para os papéis dentro do projeto interagirem entre si, a fim de diminuir a falta de comunicação entre os alunos.

Referências Bibliográficas

- [1] IBGE - São João da Boa Vista, 2017. Disponível em: <<https://cidades.ibge.gov.br/brasil/sp/sao-joao-da-boa-vista/panorama>> Acesso em: 16 de agosto de 2019.
- [2] São João é melhor cidade para idosos, 2017. Disponível em: <http://saojoao.sp.gov.br/home/ler_noticia.php?id=2312> Acesso em: 16 de agosto de 2019.
- [3] Ministério da Saúde, Resolução da diretoria colegiada – rdc nº 283, 2005. Disponível em: <https://svn.sbv.ifsp.edu.br/svn/pds2019vespertino/documentacao/comum/Documentos%20de%20Institui%C3%A7%C3%B5es%20de%20Longa%20Perman%C3%Aancia%20-%20Estudo/04%20-%20RDC_283_2005_COMP.pdf> Acesso em: 25 de agosto de 2019.
- [4] Missão e Visão - Institutos Federais Por Estado. Disponível em: <<http://pdi.ifpe.edu.br/wp-content/uploads/2014/08/Refer%C3%Aancias-de-Miss%C3%A3o-e-Vis%C3%A3o-de-outros-Institutos-Federais.pdf>> Acesso em: 25 de agosto de 2019.
- [5] SILBERSCHATZ, A ; KORTH, H.F e SUDARSHAN, S. Sistema de banco de dados – Terceira edição. São Paulo: Pearson Makron Books,1999.
- [6] RAMAKRISHNAN, R; GEHRKE, J. Sistema de gerenciamento de banco de dados – Terceira edição. São Paulo: McGraw-Hill, 2008.
- [7] Disponível em:<https://www.google.com/url?sa=i&source=images&cd=&ved=2ahUKEwiNi7nM0OTkAhX4ILkGHcHLAo8QjRx6BAgBEAQ&url=https%3A%2F%2Fwww.questaocerta.com.br%2Fquestoes%2Fassunto%2Fdiagrama-de-entidade-e-relacionamento%3Fimprimir%3Dtrue&psig=AOvVaw1qhpBAGmv_crQ3Vfu5T3mG&ust=1569248397038575> Acesso em: 22 de setembro de 2019.
- [8] Disponível em:<https://www.google.com/url?sa=i&source=images&cd=&ved=2ahUKEwiaiOze0-TkAhUqKrKGHwVgDIwQjRx6BAgBEAQ&url=http%3A%2F%2Ffueltonmeira2012.blogspot.com%2F2012%2F10%2F&psig=AOvVaw22JbOVL1ZebMk_Akte_mxV&ust=1569249399370389> Acesso em: 22 de setembro de 2019.
- [9] Disponível em: <<https://www.google.com/imgres?imgurl=https%3A%2F%2Fabutresmacrobios.files.wordpress.com%2F2013%2F06%2F13-uml-cruzamento-de-informac3a7c3a3o.png&imgrefurl=https%3A%2F%2Fabutresmacrobios.wordpress.com%2Fmodelo-relacional-parte-02%2F&docid=s9K3dBVtCSEZ6M&tbnid=9uKwmysXIz9PUM%3A&vet=10ahUKEwjNgr-A3OTkAhUSDrkGHVJIBssQMwg6KAYwBg..i&w=714&h=461&bih=657&biw=1360&q=exemplo%20de%20terceira%20tabela%20modelo%20relacional&ved=0ahUKEwjNgr->>

[A3OTkAhUSDrkGHVJIBssQMwg6KAYwBg&iact=mrc&uact=8](#)> Acesso em: 22 de setembro de 2019.

[10] Carlos Henrique Candido - brModelo, 2018. Disponível em:<<http://www.sis4.com/brModelo/>> Acesso em: 11 de outubro de 2019.

[11] Andrei L - O Que É MySQL? Guia Para Iniciantes, 2019. Disponível em: <<https://www.hostinger.com.br/tutoriais/o-que-e-mysql/>> Acesso em: 11 de outubro de 2019.

[12] Paulo Higa - O que é XAMPP e para que serve, 2012. Disponível em: <techtudo.com.br/dicas-e-tutoriais/noticia/2012/02/o-que-e-xampp-e-para-que-serve.html> Acesso em: 11 de outubro de 2019.