



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFOMÁTICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
COORDENAÇÃO DE GRADUAÇÃO EM ENGENHARIA
ELÉTRICA**

Trabalho de Conclusão de Curso (TCC)

Projeto do simulador para treinamento de operadores: Módulo do banco de dados

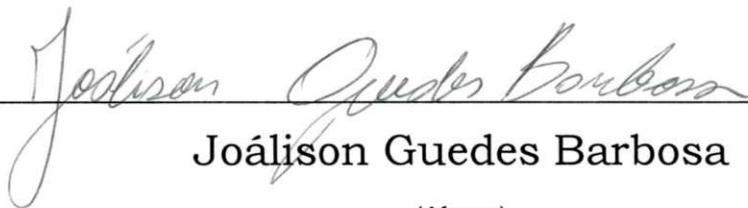
Joálison Guedes Barbosa
(Aluno)

Prof.^a Maria de Fátima Queiroz Vieira
(Orientadora)

Campina Grande – PB
Abril 2010

Projeto do simulador para treinamento de operadores: Módulo do banco de dados

*Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Engenharia Elétrica da
Universidade Federal de Campina Grande, em cumprimento
parcial às exigências para obtenção do grau de
Engenheiro Eletricista.*



Joálison Guedes Barbosa

(Aluno)



Maria de Fátima Queiroz Vieira

(Orientadora)

Campina Grande – PB
Abril 2010

DEDICATÓRIA

Dedico esse trabalho a meus pais, José Barbosa da Silva e Joana Guedes Barbosa.

A meus irmãos Joailson Guedes Barbosa e José Barbosa da Silva Júnior.

Aos amigos que me acompanharam ao longo da minha vida acadêmica.

Não quero aqui citar nomes para não cometer a gafe de esquecer algum.

AGRADECIMENTOS

Agradeço a professora Fátima por disponibilizado tempo e recursos do LIHM, essenciais para ter alcançado êxito nesse projeto.

A todo pessoal do LIHM por sua presteza, em especial a Raffael, Alves e Daniel.

LISTA DE FIGURAS

Figura 1: Arquitetura do Simulador	8
Figura 2: Simulador	9
Figura 3: MER do antigo BD	10
Figura 4: Arquitetura ANSI-SPARC para o simulador	12
Figura 5: Caso de uso	13
Figura 6: Visualização inteira do MER	19
Figura 7: Treinamento e resultados	20
Figura 8: Pessoal.....	20
Figura 9: Instalações	21
Figura 10: Eventos.....	22
Figura 11: Cenários e Estado da Subestação	23
Figura 12: Dispositivo	24
Figura 13: Detalhe de tabelas do novo BD.....	24

LISTA DE TABELAS

Tabela 1: Visões dos usuários	14
Tabela 2: Resposta do servidor antigo.....	16
Tabela 3: Apresenta painéis.....	27
Tabela 4: Mostra os dispositivos	28
Tabela 5: Lista de estados de uma planta	30
Tabela 6: Estado inicial da planta.....	31
Tabela 7: Dispositivos e seus relativos painéis	32
Tabela 8: Apresenta cenário	33
Tabela 9: Treinamento individual	33
Tabela 10: Treinamento coletivo.....	33

SUMÁRIO

LISTA DE FIGURAS	5
LISTA DE TABELAS	6
SUMÁRIO.....	7
1. INTRODUÇÃO.....	8
1.1. DESCRIÇÃO GERAL DO PROJETO.....	8
1.2. O BANCO DE DADOS	10
1.3. OBJETIVOS DO PROJETO.....	11
1.4. APRESENTAÇÃO DO TEXTO.....	11
2. A ARQUITETURA DO BD DO SIMULADOR.....	12
2.1 A ARQUITETURA DE UM SGBD	12
2.2 A ARQUITETURA EM MÓDULOS	13
3. O PROJETO DO BD DO SIMULADOR.....	15
3.1. ATIVIDADE APOIADA	15
3.2 VALIDAÇÃO DO BD ORIGINAL.....	15
3.3 MODELAGEM DOS DADOS	16
3.3.1 Tuplas	16
3.3.2 Dicionário de Dados	18
3.3.3 Apresentação do Modelo Entidade-Relacionamento	18
3.4 ARMAZENAMENTO DOS DADOS.....	24
3.4.1 O phpMyAdmin para WEB.....	24
3.4.2 Povoando as tabelas.....	24
3.4.3 Apresentação das tabelas	25
4. INTERLIGAÇÃO DO SGBD AO SIMULADOR	26
5. VALIDAÇÃO	27
6. CONSIDERAÇÕES FINAIS	34
ANEXO A – DICIONÁRIO DE DADOS	36
ANEXO B – TABELAS do BD (modelo físico)	47

1. INTRODUÇÃO

A necessidade de treinamento de operadores do setor energético deu origem ao projeto de um simulador construído em realidade virtual. Neste setor o treinamento de operadores originalmente era feito a partir de aulas teóricas e da utilização dos equipamentos no sistema real (Souza, 2008). Ambientes de simulação virtual surgem como mecanismos que facilitam outros métodos de treinamento.

A idéia original tornou-se mais ampla e genérica, passando a ser útil também para outros setores onde o treinamento virtual por meio tutor se torna cada vez mais necessário.

Essa ferramenta traz vantagens para o treinamento na rotina de trabalho do operador reduzindo a necessidade da presença do tutor no local do treinamento e dispensando a necessidade de um ambiente físico para realização de treinamento, sob o risco de desenergização da rede elétrica.

1.1. DESCRIÇÃO GERAL DO PROJETO

Há cerca de seis anos, a equipe do Laboratório de Interfaces Homem-Máquina (LIHM) do DEE UFGC vem desenvolvendo o projeto de um simulador para treinamento de operadores, com base na representação do ambiente físico de trabalho em realidade virtual e da representação do comportamento dos objetos (motor de simulação) construído no formalismo de redes de Petri. A arquitetura do simulador é ilustrada na figura 1.

A versão inicial do simulador foi desenvolvida em VRML¹ (ver figura 2).

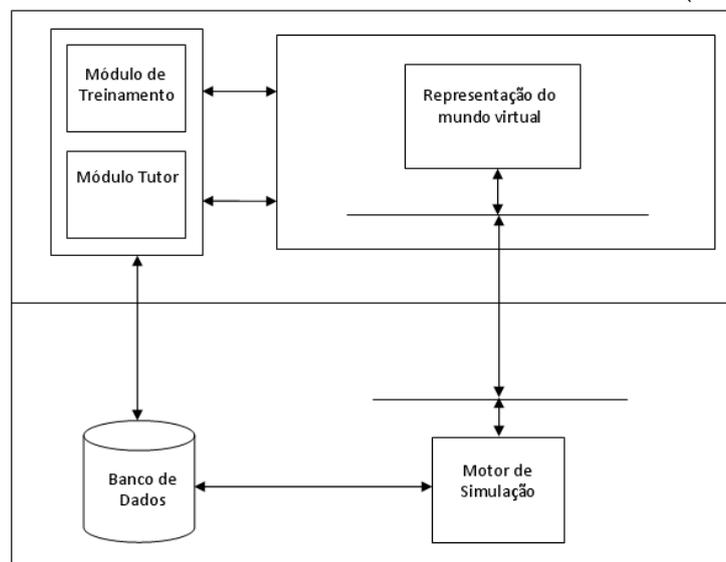


Figura 1: Arquitetura do Simulador

¹Virtual Reality Modeling Language



Figura 2: Simulador

No ambiente representado no simulador o usuário é livre para se mover e atuar nos pontos sensíveis dos painéis, isto é, nas chaves que representam o comando dos disjuntores e seccionadoras.

A camada do motor de simulação é composta de modelos construídos em rede de Petri Colorida (CPN), os quais se comunicam entre si e com a camada de visualização (Vieira et al 2010). Os modelos CPN representam a planta do sistema elétrico, a interface de operação via controle supervisão e a interface de operação via painéis. A comunicação entre os modelos se dá através da troca de mensagens, utilizando o protocolo TCP/IP².

O deslocamento no ambiente virtual pode ser feito com mudança de pontos de vista ou através da utilização do mouse. No primeiro caso utiliza-se o menu *Navigate/Next Viewpoint*, de forma que a próxima vista da seqüência será exibida, e isso é feito de maneira cíclica. Esta funcionalidade é inerente ao software de visualização *FreeWRL* (FreeWRL, 2010).

Para o deslocamento no ambiente virtual, com o auxílio do mouse, a interação se dá da seguinte forma:

- Posiciona-se o cursor do mouse no centro da tela.
- Mantem-se pressionado o botão esquerdo do mouse.
- Movimenta-se o cursor na tela de tal modo que:
 - Movimento do cursor para cima equivale a caminhar para frente;
 - Movimento do cursor para baixo é equivalente a caminhar para trás;
 - Movimento do cursor para a esquerda equivale a virar-se para esquerda;
 - Movimento do cursor para a direita equivale a virar-se para direita.

A abertura ou fechamento de chaves nos painéis do simulador é feita através do mouse seguindo os passos:

² TCP *Transmission Control Protocol* e o IP *Internet Protocol*

- Posiciona-se o cursor do mouse sobre a chave a ser fechada ou aberta.
- Clica-se com o botão esquerdo do mouse sobre a chave.

1.2. O BANCO DE DADOS

A primeira versão do banco de dados do simulador foi desenvolvida por (Souza, 2008) e ainda não havia sido integrado ao projeto do simulador. O BD inicial abrangeu o armazenamento de informações sobre os usuários do simulador (treinandos e tutores) e sobre a estruturação e resultados dos treinamentos.

O modelo de entidade relacionamento (MER) do banco de dados (BD) original é ilustrado na Figura 3.

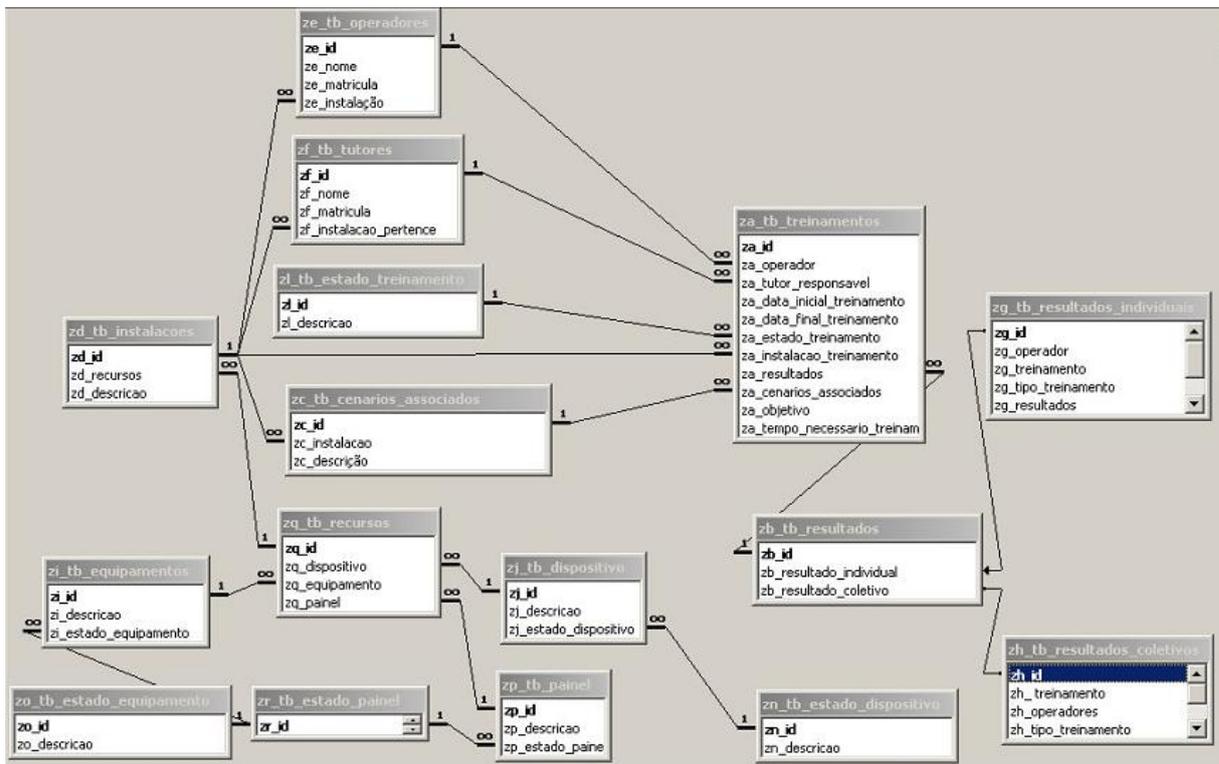


Figura 3: MER do antigo BD

Este banco de dados atende aos requisitos do módulo treinamento do simulador, contudo deixa uma lacuna no que se refere o módulo tutor.

O projeto relativo ao módulo tutor requer uma gama maior de configurações e informações a serem armazenadas. Dessa forma constatou-se a necessidade de incluir no modelo de dados as informações sobre cenários de treinamento, o que implica no armazenamento do estado inicial e estados intermediários e final de uma instalação do sistema elétrico (uma subestação).

Conclui-se de um estudo inicial sobre o BD original que seria necessário reorganizar os objetos que representam os componentes de uma instalação. As configurações dos cenários de treinamento devem permitir mudanças de acordo com o planejamento do tutor antes e durante o treinamento. O aspecto do treinamento remoto também foi incorporado ao projeto.

Assim, foi concebida uma extensão além de modificações no modelo do BD original. O novo banco acomoda a execução do simulador que foi constatada a partir da análise do roteiro de manobras da CHESF³.

1.3. OBJETIVOS DO PROJETO

O principal objetivo deste trabalho é o refinamento e ampliação de um banco de dados para dar suporte à execução de cenários de treinamento. O simulador é composto por dois módulos, o módulo de treinamento e o módulo tutor. O módulo de treinamento já existe, enquanto que o módulo tutor encontra-se em fase de desenvolvimento.

O banco de dados é um componente fundamental para o projeto do simulador, armazenando informações sobre os tipos de instalações, eventos associados a cenários, a configuração da instalação e o registro do *logs* das ações dos usuários.

No projeto do simulador, o tutor poderá verificar remotamente em tempo real (via WEB⁴) se o operador executa corretamente a rotina de trabalho, escolhe uma determinada atividade, indicar uma nova tarefa, ou até mesmo interferir, criando situações inesperadas a partir do disparo de eventos. Poderá acompanhar os treinandos sob diferentes pontos de vista, avaliando se adotaram os procedimentos corretos, se o fizeram no prazo prescrito e se responderam corretamente a eventos inesperados.

A primeira fase deste trabalho consistiu na validação do modelo de dados existente, testando os relacionamentos e verificando se a estrutura consegue representar adequadamente as funções necessárias ao módulo de treinamento.

Na seqüência foi realizado o refinamento e modelagem do novo banco de dados, consideradas as demandas do módulo tutor. E como última atividade foi previsto o acesso ao BD via WEB.

1.4. APRESENTAÇÃO DO TEXTO

Este texto está elaborado em seis capítulos. O capítulo 2 apresenta a arquitetura de banco de dados adotada e a estrutura modular do banco de dados desenvolvido. O capítulo 3 trata da modelagem, do armazenamento de dados, e das ferramentas utilizadas no projeto. No capítulo 4 o BD é integrado ao projeto do simulador. Enquanto o capítulo 5 descreve a validação do banco de dados. Em seqüência são tecidas as considerações finais.

³ Companhia Hidro Elétrica do São Francisco

⁴ World Wide Web

2. A ARQUITETURA DO BD DO SIMULADOR

Um SGBD (Sistema Gerenciador de Banco de Dados) é composto de uma coleção de arquivos inter-relacionados e de um conjunto de programas que permitem aos usuários fazer o acesso a estes arquivos e modificá-los, quando necessário.

2.1 A ARQUITETURA DE UM SGBD

Antes de tratar da arquitetura propriamente dita é importante destacar em qual nível da arquitetura se situa este trabalho, para tanto será apresentada uma breve revisão sobre arquiteturas de sistemas de bancos de dados.

O objetivo de um sistema de banco de dados é prover os usuários uma visão dos dados, próxima à realidade. Isto é, o sistema oculta detalhes de como os dados são armazenados. Uma vez que muitos dos usuários de banco de dados não têm familiaridade com a computação, a complexidade deve ser embutida em níveis de abstração que simplificam sua interação com o sistema.

A maioria dos SGBD é baseada no sistema ANSI-SPARC⁵. Sua arquitetura conceitual consiste em três níveis (A figura 4 apresenta a arquitetura do BD do simulador):

- O nível externo (lógico do usuário);
- O nível conceitual (lógico comunitário, ou apenas indireto);
- O nível interno (físico).

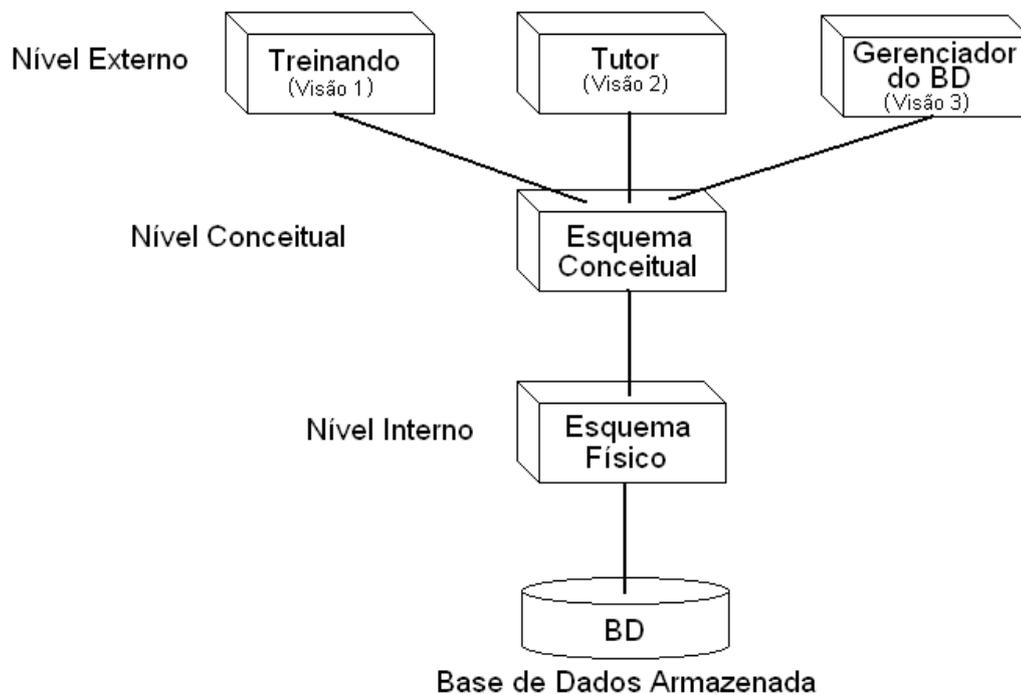


Figura 4: Arquitetura ANSI-SPARC para o simulador

⁵ American National Standards Institute, Standards Planning And Requirements Committee

O **nível de visões** (nível externo) é o mais alto nível de abstração. Descreve apenas parte do banco de dados. Muitos usuários do sistema de banco de dados não estarão interessados em todas as informações, ao invés disso precisam de apenas uma parte do banco de dados. O nível de abstração das visões de dados é definido para simplificar esta interação com o sistema, que pode fornecer múltiplas visões para o mesmo banco de dados

O **nível conceitual** descreve quais dados estão armazenados no banco de dados e as relações que existem entre eles. Aqui o banco de dados em sua totalidade é descrito em termos de sua estrutura. Embora a implementação da estrutura do BD no nível conceitual possa corresponder a estruturas complexas no nível físico, o usuário do nível conceitual não precisa desse conhecimento para acessar o BD. O nível conceitual de abstração também é usado por administradores de banco de dados, que podem decidir quais informações devem ser mantidas no BD.

O **nível físico** é o nível mais baixo de abstração. Descreve como os dados estão realmente armazenados. No nível físico, as estruturas de dados armazenam a informação propriamente dita.

2.2 A ARQUITETURA EM MÓDULOS

Um caso de uso geral concebido para este projeto corresponde à visão do usuário tutor (módulo tutor) e do usuário treinando (módulo treinamento). Essas duas visões distintas são o ponto de partida para traçar um esquema conceitual para o projeto (figura 5).

O tutor elabora o cenário do treinamento, e pode alterá-lo adicionando eventos ao simulador durante o treinamento. Enquanto o treinando (operador) fica limitado a resolver os problemas que lhe são apresentados.

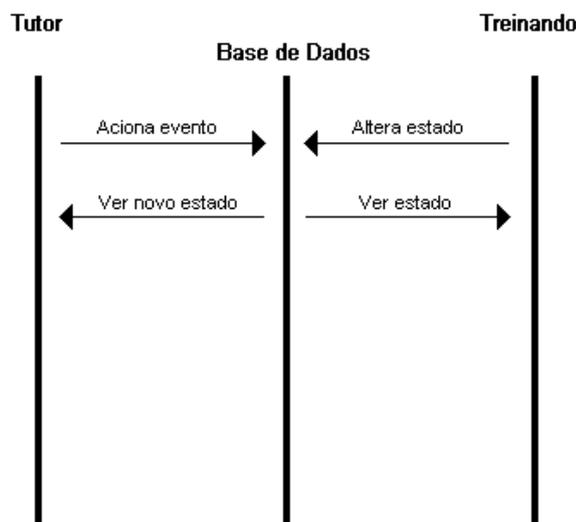


Figura 5: Caso de uso

Cada usuário interage a partir de sua visão. O treinando interage com o BD a partir da interface com o simulador que repassa as consultas ao BD. Por sua vez o tutor acessa o BD de forma mais direta, preparando os cenários de treinamento a partir da interface com o simulador.

Na visão do treinando, o operador interage em uma sala virtual que representa o ambiente de treinamento no qual deve realizar tarefas programadas pelo tutor. O ambiente de treinamento é o de uma sala de comando de subestação representado em realidade virtual. O estado da subestação é representado nos painéis e em uma tela de supervisorio virtuais. O operador interage com o ambiente virtual, modificando o estado de objetos tais como chaves e botoeiras e verifica o estado resultante em mostradores. A sua atuação deve ser gravada em um *log* de dados para análise posterior. Durante o treinamento o tutor pode gerar eventos além daqueles que já havia programado para o cenário de treinamento.

A estrutura do banco é cliente-servidor (ver Figura 5). A integração do banco de dados ao projeto do simulador não faz parte do escopo deste projeto. No entanto, como parte deste trabalho foram construídas as visões dos usuários do simulador (tutor e treinando) as quais são detalhadas na tabela 1.

	Dispositivo	Painel	SE	Planta	Evento	Tarefa	Instalação
Operador	C-estado A-estado M-estado	C-marcação	C-estado M-estado	C-estado		C-tudo	C-tudo
Tutor	C-estado A-estado M-estado	C-marcação M-marcação	C-estado M-estado	C-estado A-estado M-estado E-estado	C-estado A-estado M-estado E-estado	C-tudo	C-tudo
Gerenciador do BD	C-estado A-tipo M-tipo E-tipo	C-marcação A-tudo M-tudo E-tudo	C-tudo A-tudo M-tudo E-tudo	C-marcação A-tudo M-tudo E-tudo	C-tudo A-tudo M-tudo E-tudo	C-tudo A-tudo M-tudo E-tudo	C-tudo A-tudo M-tudo E-tudo

Legenda:
 C => Consultar
 A => Adicionar
 M => Modificar
 E => Excluir

Tabela 1: Visões dos usuários

A visão para o treinando está associada ao módulo de treinamento o qual já se encontra em fase de protótipo funcional. No entanto o módulo tutor ainda está em desenvolvimento. Das visões descritas foi desenvolvido o novo⁶ modelo conceitual do SGBD.

⁶ O antigo modelo conceitual foi descrito no trabalho (Souza, 2008)

3. O PROJETO DO BD DO SIMULADOR

O simulador é composto de uma descrição em Realidade Virtual da sala de comando, compreendendo tanto painéis de controle como uma representação do sistema supervisor. Sua descrição do mundo virtual pode ser visualizada através do *software* FreeWRL e está ligada a um motor de simulação construído em Java , ou em redes de Petri coloridas.

3.1. ATIVIDADE APOIADA

Para esta implementação foi desenvolvido um banco de dados para armazenar informações sobre: objetos do cenário, estado dos objetos da planta, informações sobre treinandos e tutores, cenários de treinamento, além do histórico das sessões de treinamento (*log*) para avaliação do desempenho dos treinandos.

No ambiente virtual do simulador são representados os objetos utilizados pelo operador da subestação elétrica durante a realização de uma tarefa. Dentre estes objetos se encontram os painéis com suas chaves e mostradores para atuação sobre equipamentos em campo. Acrescentando a capacidade ao usuário de se deslocar se livremente pelo ambiente virtual e interagir com os dispositivos ali representados, obtemos um ambiente fiel à realidade e capaz de ser usado para treinamento de operadores em diversos cenários.

Cada cenário necessita conter as informações de estado da planta simulada. Também é importante armazenar o estado da planta antes e após algum evento.

Em um cenário um evento pode ser desencadeado: quando o tutor deseja construir um novo cenário, estabelecendo o momento exato de sua ocorrência, após decorrer um prazo pré-estabelecido, ou em consequência da atuação sobre algum dispositivo na planta.

3.2 VALIDAÇÃO DO BD ORIGINAL

A estrutura do banco de dados original (Souza, 2008) foi mantida e a sua validação consistiu em povoar o BD e realizar consultas relacionadas ao comportamento do simulador. Essa validação consistiu em instanciar um roteiro de manobras da empresa CHESF (Nascimento, 2004).

Os testes do BD foram feitos através de consultas em SQL⁷ submetidas ao banco instalado no servidor do LIHM. Segue um exemplo de consulta em SQL, cujo resultado é ilustrado na Tabela 2:

```
SELECT `zf_matricula`,`zf_nome`,`zd_descricao`  
FROM zd_tb_instalacoes, zf_tb_tutores
```

⁷ *Structured Query Language*, ou Linguagem de Consulta Estruturada

←T→		
zf_matricula	zf_nome	zd_descricao
32837	Ambrosini	Campina Grande II
32837	Ambrosini	Campina Grande I
32837	Ambrosini	Tacaimbó
75844	Clarence	Campina Grande II
75844	Clarence	Campina Grande I
75844	Clarence	Tacaimbó
57689	Maldini	Campina Grande II
57689	Maldini	Campina Grande I
57689	Maldini	Tacaimbó

Tabela 2: Resposta do servidor antigo

Dos testes realizados ficou evidente a necessidade de refinar o projeto do BD. Foi então construída uma cópia manual já que não havia um arquivo com extensão apropriada para a modelagem visual de classes, usando *software* livre e aberto.

A proposta consistiu em estender o banco, visando armazenar a descrição de um cenário de treinamento em uma subestação elétrica, utilizando apenas software livre e aberto.

3.3 MODELAGEM DOS DADOS

O projeto de um sistema de banco de dados adotou o modelo Entidade Relacionamento, que representa o projeto conceitual do sistema, e se encontra descrito a seguir.

3.3.1 Tuplas

O modelo de dados relacional representa os dados da base de dado como uma coleção de relações. Cada relação pode ser entendida como uma tabela ou um simples arquivo de registros.

Cada linha da tabela é formada por uma lista ordenada de colunas que representa um registro, ou tupla. Os registros não precisam conter informações em todas as colunas, podendo assumir valores nulos quando assim for necessário. Na seqüência é apresentado o conjunto de relações base do projeto deste BD, na forma de tuplas, onde PK (*Primary Key*) são chaves primárias e FK (*Foreing Key*) são chaves estrangeiras.

zaa_tb_forma_treinamento (zaa_id(PK), zaa_tb_forma_treinamento);

zab_tb_objetivo_treinamento (zab_id(PK), zab_tb_objetivo_treinamento);

zac_tb_tipo_treinamento (zac_id(PK), zac_tb_tipo_treinamento);

zad_tb_estado_treinamento (id_zad(PK), zad_estado_treinamento);

za_tb_treinamentos (za_id(PK), za_operador(FK), za_tutor(FK),
za_cenarios_associados(FK), za_resultados(FK), za_tipo_treinamento(FK),
za_estado_treinamento(FK), za_objetivo_treinamento(FK), za_forma_treinamento(FK),
za_data_treinamento, za_tempo_necessario_treinamento, za_numero_operadores, za_turma);

zb_tb_resultados (zb_id(PK), zb_resultados_coletivos(FK), zb_resultados_individuais(FK));

zca_tb_tarefas (zca_id(PK), zca_identificacao, zca_edicao, zca_data, zca_motivo,
zca_observacao);

zc_tb_cenarios_associados (zc_id(PK), zc_tarefas(FK), zc_eventos(FK), zc_estado_se(FK),
zc_cenario);

zd_tb_instalacoes (zd_id(PK), zd_recurso, zd_instalacao, zd_descricao);

zd_tb_instalacoes_tem_zc_tb_cenarios_associados (zd_tb_instalacoes_zd_id(FK),
zc_tb_cenarios_associados_zc_id(FK));

ze_tb_operadores (ze_id(PK), ze_instalacoes(FK), ze_operador, ze_matricula,
ze_treinamento_anterior, ze_lentes_corretivas, ze_data_nascimento, ze_experiencia_anterior,
ze_anos_na_funcao);

zf_tb_tutores (zf_id(PK), zf_instalacao(FK), zf_tutor, zf_matricula);

zg_tb_resultados_individuais (zg_id(PK), zg_nota, zg_descricao, zg_numero_erros);

zh_tb_resultados_coletivos (zh_id(PK), zh_nota, zh_descricao, zh_numero_erros);

zi_tb_estado_se (zi_id(PK), zi_estado_instalacao);

zja_tb_tipo_dispositivo (zja_id(PK), zja_tipo_dispositivo);

zjb_tb_categoria_dispositivo (zjb_id(PK), zjb_categoria_dispositivo);

zjc_tb_especifica_dispositivo (zjc_id(PK), zjc_especifica_dispositivo);

zj_tb_dispositivo (zj_id(PK), zj_tipo_dispositivo(FK), zj_categoria_dispositivo(FK),
zj_especifica_dispositivo(FK));

zka_tb_lista_eventos (zka_id(PK), zka_evento);

zkb_tb_estado_planta (zkb_id(PK), zkb_lista_estado(FK), zkb_dispositivo(FK));

zkc_tb_estado_atuador (zkc_id(PK), zkc_dispositivo(FK), zkc_lista_estado(FK));

zkd_tb_planta (zkd_id(PK), zkd_estado);

zkd_tb_planta_tem_zkb_tb_estado_planta (zkd_planta(FK), zkb_estado_planta(FK));

zk_tb_eventos (zk_id(PK), zk_planta(FK), zk_lista_eventos(FK), zk_estado_atuador(FK),
zk_pelo_tempo, zk_por_dispositivo, zk_pelo_tutor);

zk_tb_eventos_tem_zl_tb_estado_dispositivo (zl_estado_dispositivo(FK), zk_eventos(FK));

zla_tb_lista_estado (zla_id, zla_estado);

zl_tb_estado_dispositivo (zl_id(PK), zl_instalacao, zl_dispositivo, zl_momento);

zpa_tb_tipo_painel (zap_id(PK), zja_tipo_painel);

zpb_tb_aviso_painel (zpb_id(PK), zpb_tipo_aviso, zpb_aviso);

zp_tb_painel (zp_id(PK), zp_aviso_painel(FK), zp_tipo_painel(FK), zp_codigo_painel, zp_marcacao);

zp_tb_painel_tem_zj_tb_dispositivo (zj_dispositivo(FK), zp_painel(FK));

zqa_objetos_interacao (zqa_id(PK), objetos_interacao);

zq_tb_recurso (zq_id(PK), zq_objetos_interacao(FK), zq_nome);

zq_tb_recurso_tem_zp_tb_painel (zq_tb_recurso_zq_id(FK), zp_tb_painel_zp_id(FK));

A terminologia utilizada mantém o padrão do banco de dados original, de modo a facilitar uma consulta ao BD.

3.3.2 Dicionário de Dados

Além do modelo de Entidade e Relacionamento, é necessário armazenar um documento com a descrição dos objetos do BD. Este documento, denominado Dicionário de Dados, contém informações sobre todos os objetos do BD, de forma textual.

O intuito é padronizar a semântica do BD; e a representação dos dados, de acordo com os tipos (inteiro, caracter, entre outros). O dicionário de dados deste projeto é apresentado no anexo A.

3.3.3 Apresentação do Modelo Entidade-Relacionamento

Nesta etapa foi utilizada a ferramenta DBDesigner (DBDesigner, 2010), que oferece uma interface de usuário simples além de seu desenvolvimento para código aberto MySQL⁸.

O diagrama MER elaborado no DBDesigner foi muito extenso para uma visualização satisfatória em apenas uma página desse documento. Para superar essa limitação o modelo foi seccionado em 6 setores:

- Treinamento e resultados;
- Pessoal;

⁸ SGBD que utiliza a linguagem SQL

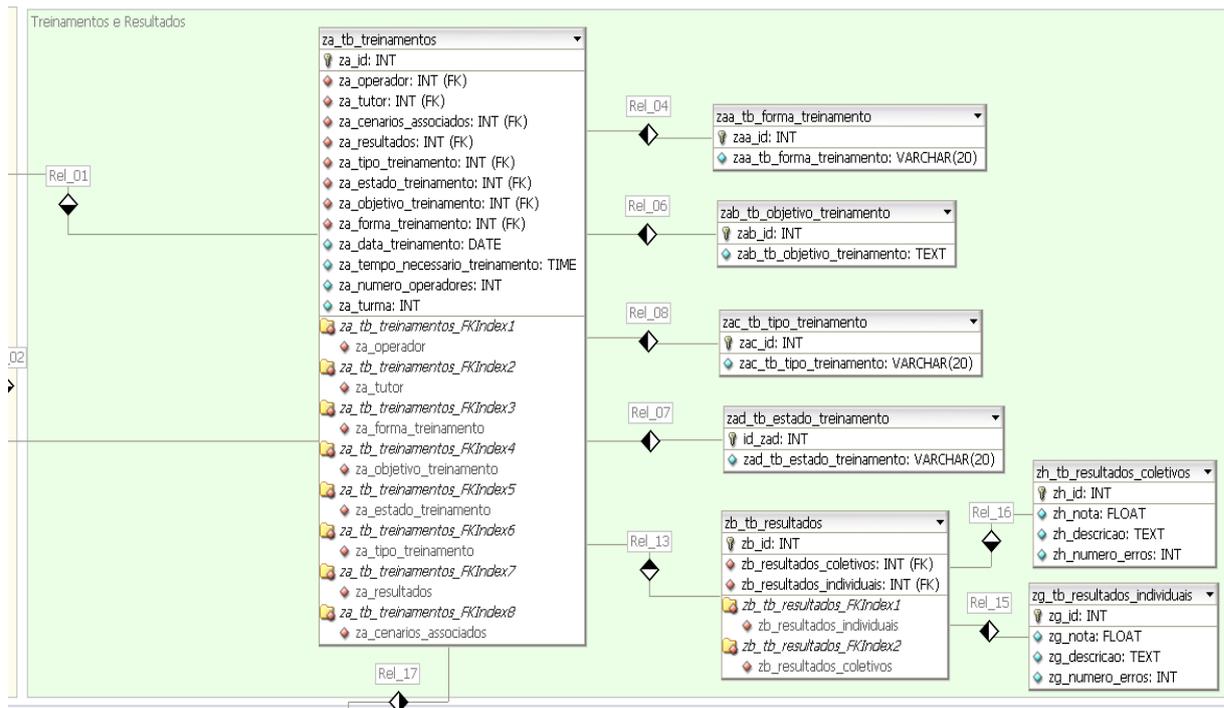


Figura 7: Treinamento e resultados

O setor de **Pessoal** (ver Figura 8) é composto pelas entidades operadores e tutores. Como citado anteriormente, ambas entidades possuem relacionamento 1:n com a entidade treinamento. Além disso, cada um (operadores e tutores) está lotado em uma única instalação.

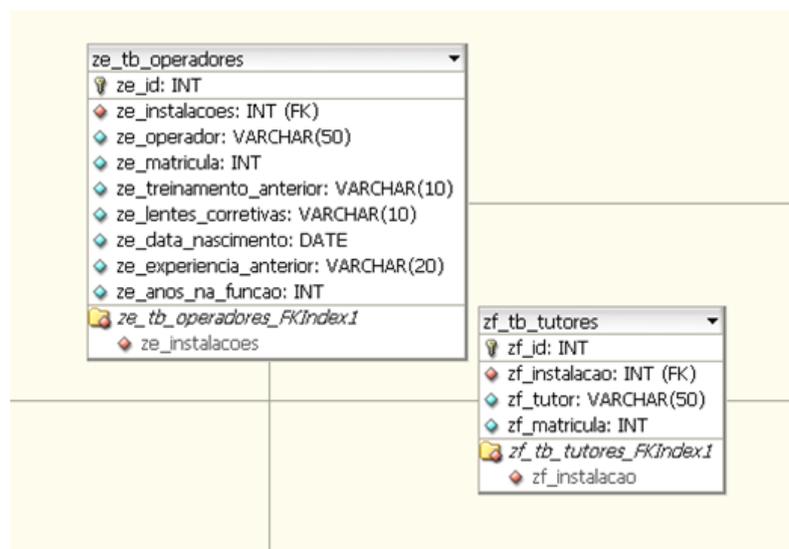


Figura 8: Pessoal

Na seção **Instalações** (ver Figura 9) há as entidades: instalações, recursos e painel. As demais tabelas são oriundas de chaves estrangeiras obedecendo as regras de normalização.

Cada instalação possui muitos recursos mas, um dispositivo é único numa instalação. Entre instalação e cenários associados é estabelecida uma relação do tipo

n:m. Uma relação de muitos para muitos é estabelecida entre recurso e painel e entre painel e dispositivo.

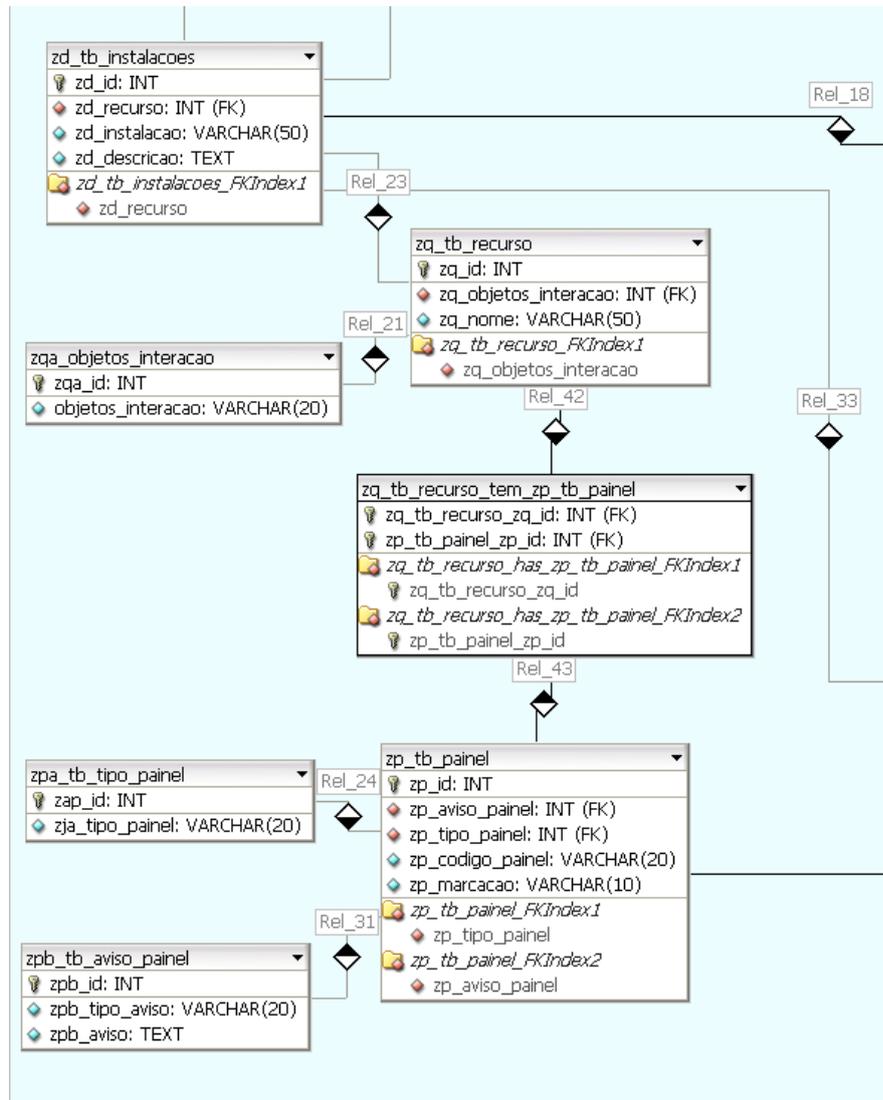


Figura 9: Instalações

O setor de **Eventos** (Figura 10) é de grande importância para o módulo tutor. Nesse setor encontram-se as entidades: eventos, planta, estado da planta e estado do atuador.

A entidade eventos possui uma lista de eventos, está associada a muitas plantas e a muitos estados dos atuadores, e está associada a um cenário. Além de possuir uma relação n:m com estado do dispositivo.

A planta possui uma relação n:m com o estado da planta. O estado do atuador e o estado da planta, possuem uma relação n:1 com a lista de estados e com os dispositivos.

Há uma tabela na parte superior da Figura 10 indicando uma relação n:m entre instalações e cenários associados.

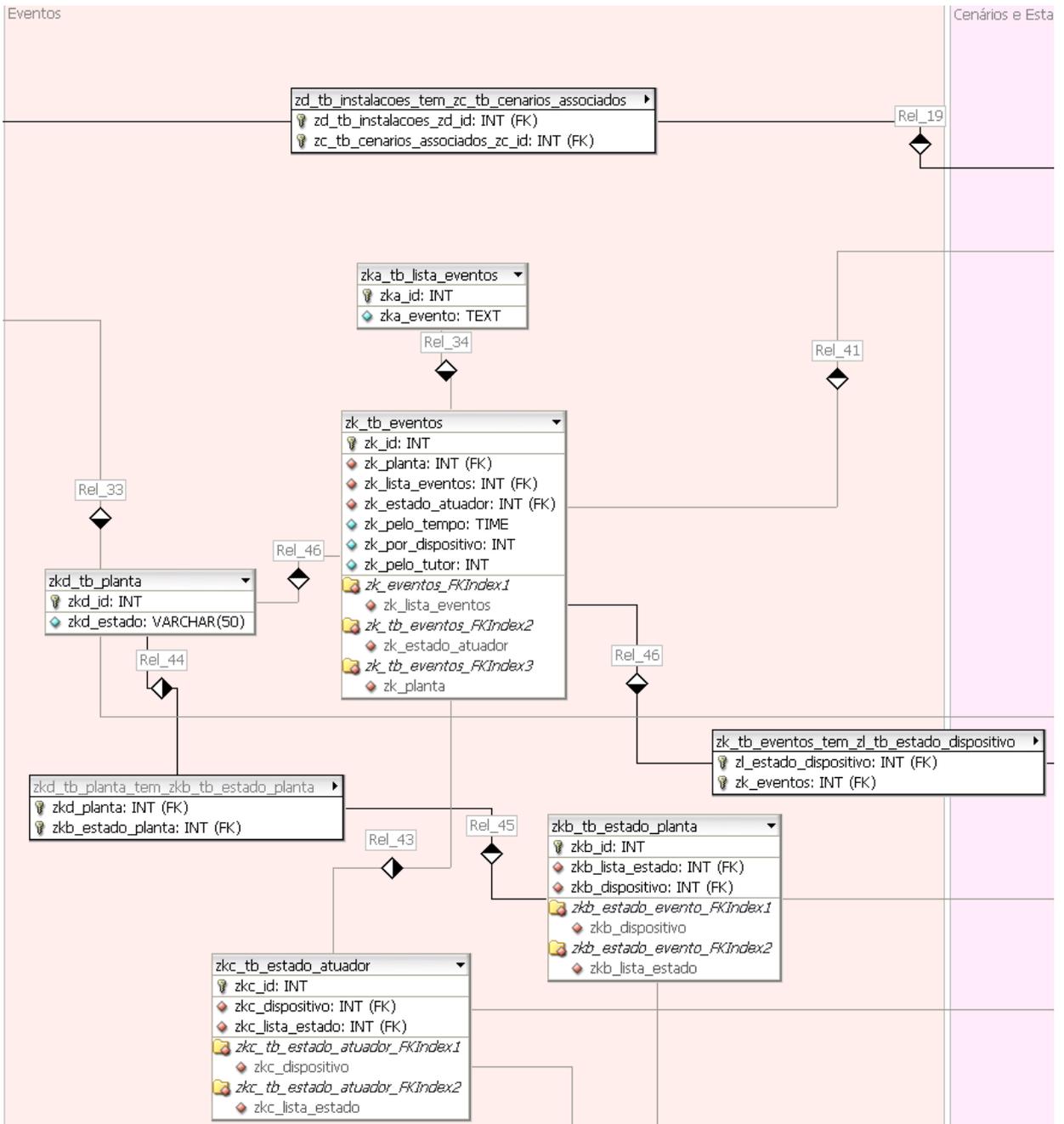


Figura 10: Eventos

Na área de **Cenários e Estados da Subestação (Figura 11)** há as entidades: cenários associados; estado da SE; estado do dispositivo; lista de estados; e tarefas.

Um cenário associado está associado a: n estados da SE, n tarefas e n eventos. Um Cenário associado está em um treinamento, e tem uma relação n:m com instalações.

O Estado do dispositivo é único em uma instalação, e tem relação n:m com estado da SE e eventos. A entidade possui n lista de estados e n dispositivos.

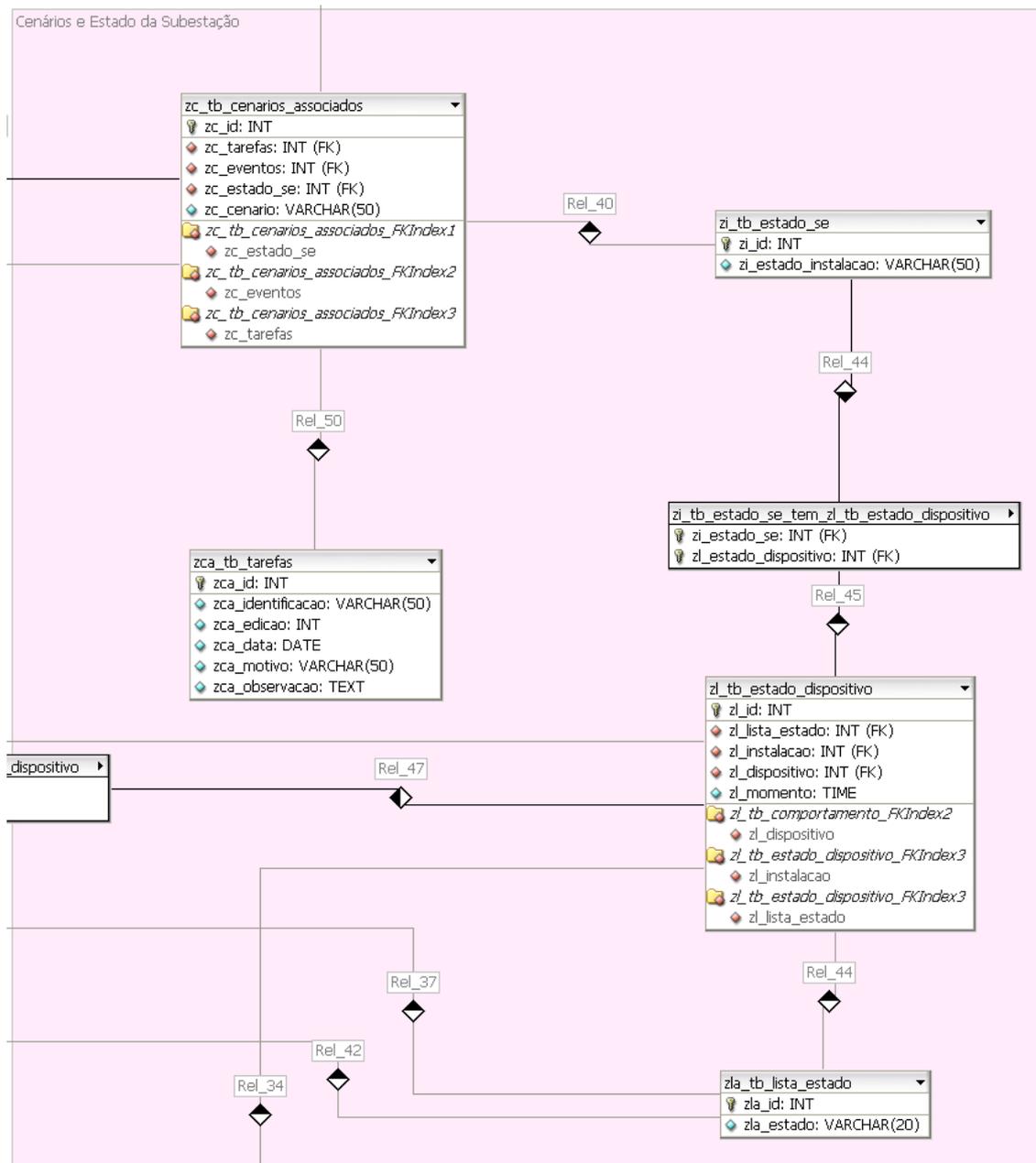


Figura 11: Cenários e Estado da Subestação

No setor **Dispositivo** (Figura 12) encontram-se o dispositivo e suas classes, em ordem crescente de nível há: tipo do dispositivo, categoria do dispositivo e específica dispositivo. Todos os níveis tem uma relação n:1 com dispositivo.

Dispositivo está em estado do dispositivo, em estado do atuador e em estado da planta. Uma relação n:m é estabelecida entre dispositivo e painel.

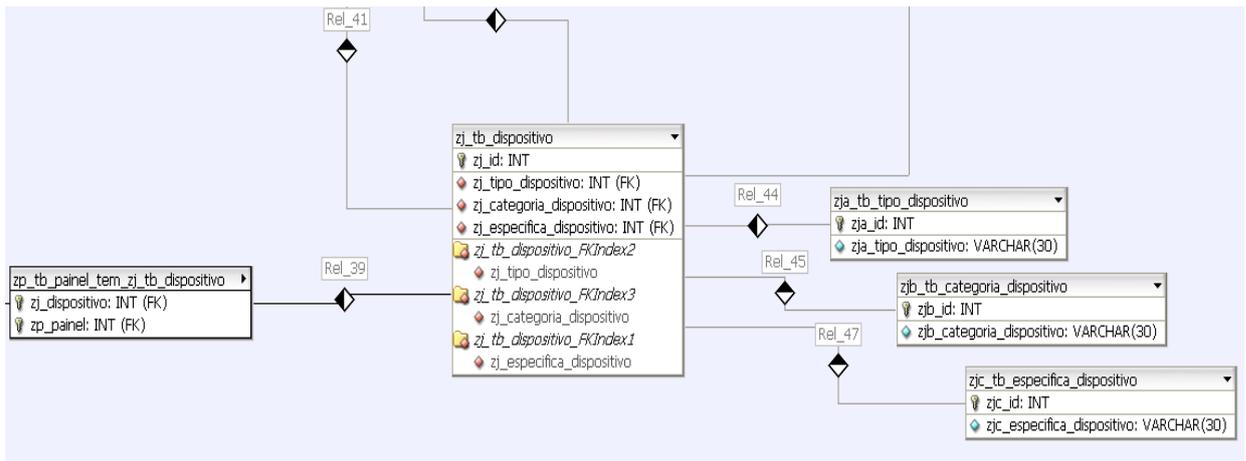


Figura 12: Dispositivo

A modelagem no DBDesigner gera uma DDL⁹ em SQL.

3.4 ARMAZENAMENTO DOS DADOS

Uma vez de posse do *script* gerado na modelagem, foi realizada a construção do nível físico do banco de dados, no servidor do LIHM, acessado através do sitio do laboratório, utilizando o phpMyAdmin (phpMyAdmin, 2010).

3.4.1 O phpMyAdmin para WEB

O phpMyAdmin é uma ferramenta de software livre escrita em PHP¹⁰ destinada a lidar com a administração do MySQL sobre a WEB. O phpMyAdmin suporta uma ampla gama de operações com o MySQL.

O phpMyAdmin instala um servidor Apache no PC e disponibiliza o banco de dados *on-line*.

3.4.2 Povoando as tabelas

O banco de dados criado no servidor do LIHM foi disponibilizado e para acesso local e remoto no endereço <http://150.165.61.62/phpMyAdmin/>.

O script em SQL foi importado e o banco de dados povoado. No detalhe de uma captura de tela são ilustradas tabelas (Figura 13). Ao clicar em  é possível inserir dados na tabela.

Estrutura SQL Procurar Procurar por exemplo Exportar Importar Operações								
Privilégios Eliminar								
	Tabela ▾	Ação				Registros	Tipo	Collation
<input type="checkbox"/>	zaa_tb_forma_treinamento					1	InnoDB	utf8_general_ci
<input type="checkbox"/>	zab_tb_objetivo_treinamento					4	InnoDB	utf8_general_ci
<input type="checkbox"/>	zac_tb_tipo_treinamento					3	InnoDB	utf8_general_ci

Figura 13: Detalhe de tabelas do novo BD

⁹ Data Definition Language ou Linguagem de Definição de Dados

¹⁰ Um acrônimo recursivo para "PHP: Hypertext Preprocessor"

Foram inseridos dados nas 36 tabelas do BD, de modo a representar os dados relacionados a uma manobra no sistema real, de acordo com um roteiro de manobras da empresa CHESF e com base no trabalho realizado no LIHM por (Guerrero, 2006).

3.4.3 Apresentação das tabelas

O modelo físico do sistema consiste em uma estrutura de tabelas no SGBD. As tabelas povoadas são apresentadas no anexo B.

4. INTERLIGAÇÃO DO SGBD AO SIMULADOR

Quando um acesso ao SGBD é solicitado pelo usuário através do simulador, este estabelece uma conexão com o SGBD instalado no servidor. Uma vez a conexão criada, o programa cliente pode se comunicar com o SGBD.

Um padrão denominado Conectividade Base de Dados Aberta (Open DataBase Connectivity - ODBC) provê uma Interface para Programação de Aplicações (API) que permite que os programas do lado cliente possam chamar o SGBD, desde que as máquinas clientes assim como o servidor tenham o software específico instalado.

Um SGBD tem um driver específico para seu sistema; assim um programa cliente pode se conectar a diversos SGBDs e enviar requisições de consultas e transações usando a API, as quais são processadas nos servidores.

Após o processamento de uma chamada de função (com uma cadeia de caracteres ou programas armazenados), o resultado é fornecido pelo servidor de dados através de tabelas armazenadas na memória. Os resultados das consultas são enviados para o programa cliente, que pode processá-los ou visualizá-los conforme a necessidade.

A linguagem Java (Java, 2010) permite o acesso a bancos de dados relacionais através das funcionalidades definidas no pacote `java.sql` e associados, que definem o "produto JDBC". JDBC é uma API para execução e manipulação de resultados a consultas SQL através de Java. Para desenvolver uma aplicação com Java e bancos de dados relacionais, é necessário:

- O pacote JDBC (padrão na distribuição da plataforma de desenvolvimento Java, desde sua versão 1.1);
- Acesso ao servidor de banco de dados relacional, ou seja, o sistema gerenciador de banco de dados phpMyAdmin;
- E um *driver* JDBC adequado ao SGBD utilizado.

Uma vez que esses recursos estejam disponíveis, a aplicação Java tem acesso ao banco de dados relacional executando os seguintes passos:

1. Habilitar o *driver* JDBC a partir da aplicação cliente;
2. Estabelecer uma conexão entre a aplicação cliente e servidor do banco de dados;
3. Montar e executar a consulta SQL desejada;
4. Processar no cliente o resultado da consulta.

Essa interligação já ocorre na prática com um banco de dados primário, contendo apenas uma tabela para registro de *log*. O banco de dados desenvolvido passa a suprir essa demanda. Cabe apenas ajustar o novo BD à estrutura existente.

5. VALIDAÇÃO

A validação preliminar do banco de dados consistiu na realização de várias consultas. Uma validação mais rigorosa só será possível quando o BD for integrado ao código do simulador.

A seguir são apresentadas as consultas ao servidor do LIHM que foram usadas nesta etapa de validação e as respectivas respostas.

Lista de consultas do módulo treinando (simulador):

1. *Apresenta os painéis utilizados em um cenário no treinamento 1:*

```
SELECT zc_cenario, zd_instalacao, zp_codigo_painel
FROM za_tb_treinamentos a, zc_tb_cenarios_associados c, zd_tb_instalacoes d,
zp_tb_painel p
WHERE a.za_id =1
```

zc_cenario	zd_instalacao	zp_codigo_painel
Cenário 1	SE Campina Grande II (CGD)	LT 02J5
Cenário 1	SE Campina Grande II (CGD)	DJ 12D1

Tabela 3: Apresenta painéis

2. *Consulta para mostrar os dispositivos:*

```
SELECT zja_tipo_dispositivo, zjb_categoria_dispositivo, zjc_especifica_dispositivo
FROM zja_tb_tipo_dispositivo, zjb_tb_categoria_dispositivo,
zjc_tb_especifica_dispositivo, zj_tb_dispositivo
WHERE zj_tipo_dispositivo = zja_id
AND zj_categoria_dispositivo = zjb_id
AND zj_especifica_dispositivo = zjc_id
ORDER BY zj_id
```

zja_tipo_dispositivo	zjb_categoria_dispositivo	zjc_especifica_dispositivo
Mostradores Digitais	Corrente	02J5
Anunciador	Botão	QS-02J5
Anunciador	Botão	QL-02J5
Anunciador	Botão	TL-02J5
Anunciador	Botão	TF-02J5
Anunciador	Botão	RE-02J5
Anunciador	Texto	02J5
Chave	Botoeira	12D1
Chave	Religamento Automático	02J5
Chave	43	12J5
Chave	CLT	12J5
Chave	GPG	32J5-4
Chave	GPG	32J5-5
Chave	Comando seccionadora	32J5-6
Chave	CLT	32J5-7
Mostradores Digitais	Corrente	12D1
Anunciador	Botão	QS-12D1
Anunciador	Botão	QL-12D1
Anunciador	Botão	TL-12D1
Anunciador	Botão	TF-12D1
Anunciador	Botão	RE-12D1
Anunciador	Texto	12D1
Chave	GPG	12D1
Chave	GPG	12D1-1
Chave	GPG	12D1-2
Anunciador	Alarme	02J5
NULL	NULL	NULL

Tabela 4: Mostra os dispositivos

3. Consulta para obter o estado de uma planta:

```

SELECT zja_tipo_dispositivo, zjb_categoria_dispositivo, zjc_especifica_dispositivo,
zla_estado
FROM zkb_tb_estado_planta z, zj_tb_dispositivo zj, zja_tb_tipo_dispositivo zja,
zjb_tb_categoria_dispositivo zjb, zjc_tb_especifica_dispositivo zjc, zla_tb_lista_estado
zla
WHERE z.zkb_lista_estado = zla.zla_id
AND z.zkb_dispositivo = zj.zj_id
AND zj.zj_tipo_dispositivo = zja.zja_id
AND zj.zj_categoria_dispositivo = zjb.zjb_id
AND zj.zj_especifica_dispositivo = zjc.zjc_id
ORDER BY zkb_id

```

zja_tipo_dispositivo	zjb_categoria_dispositivo	zjc_especifica_dispositivo	zla_estado
Mostradores Digitais	Corrente	02J5	Ativado
Anunciador	Botão	QS-02J5	Desativado
Anunciador	Botão	QL-02J5	Desativado
Anunciador	Botão	TL-02J5	Desativado
Anunciador	Botão	TF-02J5	Desativado
Anunciador	Botão	RE-02J5	Desativado
Anunciador	Texto	02J5	Desativado
Chave	Botoeira	12D1	Não Pressionado
Chave	Religamento Automático	02J5	Desativado
Chave	43	12J5	Posição '0'
Chave	CLT	12J5	Posição 'TEL'
Chave	GPG	32J5-4	Fechado
Chave	GPG	32J5-5	Fechado
Chave	Comando seccionadora	32J5-6	Aberto
Chave	CLT	32J5-7	Aberto
Mostradores Digitais	Corrente	12D1	Ativado
Anunciador	Botão	QS-12D1	Desativado
Anunciador	Botão	QL-12D1	Desativado
Anunciador	Botão	TL-12D1	Desativado
Anunciador	Botão	TF-12D1	Desativado
Anunciador	Botão	RE-12D1	Desativado
Anunciador	Texto	12D1	Desativado
Chave	GPG	12D1	Aberto
Chave	GPG	12D1-1	Fechado
Chave	GPG	12D1-2	Fechado
Mostradores Digitais	Corrente	02J5	Desativado
Anunciador	Botão	QS-02J5	Desativado
Anunciador	Botão	QL-02J5	Desativado
Anunciador	Botão	TL-02J5	Desativado
Anunciador	Botão	TF-02J5	Desativado

zja_tipo_dispositivo	zjb_categoria_dispositivo	zjc_especifica_dispositivo	zla_estado
Anunciador	Botão	RE-02J5	Desativado
Anunciador	Texto	02J5	Desativado
Chave	Botoeira	12D1	Não Pressionado
Chave	Religamento Automático	02J5	Desativado
Chave	43	12J5	Posição 'T'
Chave	CLT	12J5	Posição 'LOC'
Chave	GPG	32J5-4	Aberto
Chave	GPG	32J5-5	Aberto
Chave	Comando seccionadora	32J5-6	Fechado
Chave	CLT	32J5-7	Aberto
Mostradores Digitais	Corrente	12D1	Desativado
Anunciador	Botão	QS-12D1	Desativado
Anunciador	Botão	QL-12D1	Desativado
Anunciador	Botão	TL-12D1	Desativado
Anunciador	Botão	TF-12D1	Desativado
Anunciador	Botão	RE-12D1	Desativado
Anunciador	Texto	12D1	Desativado
Chave	GPG	12D1	Fechado
Chave	GPG	12D1-1	Fechado
Chave	GPG	12D1-2	Fechado

Tabela 5: Lista de estados de uma planta

4. Consulta para obter o estado inicial da planta:

```

SELECT zkb_id, zkd_estado, zja_tipo_dispositivo, zjb_categoria_dispositivo,
zjc_especifica_dispositivo, zla_estado
FROM zkb_tb_estado_planta z, zkd_tb_planta zk, zj_tb_dispositivo zj,
zja_tb_tipo_dispositivo zja, zjb_tb_categoria_dispositivo zjb,
zjc_tb_especifica_dispositivo zjc, zla_tb_lista_estado zla,
zkd_tb_planta_tem_zkb_tb_estado_planta zkdb
WHERE zk.zkd_id =1
AND zkdb.zkd_planta = zk.zkd_id
AND zkdb.zkb_estado_planta = z.zkb_id
AND z.zkb_lista_estado = zla.zla_id
AND z.zkb_dispositivo = zj.zj_id
AND zj.zj_tipo_dispositivo = zja.zja_id
AND zj.zj_categoria_dispositivo = zjb.zjb_id
AND zj.zj_especifica_dispositivo = zjc.zjc_id
ORDER BY zkb_id

```

zkb_id ▲	zkd_estado	zja_tipo_dispositivo	zjb_categoria_dispositivo	zjc_especifica_dispositivo	zla_estado
1	Inicial	Mostradores Digitais	Corrente	02J5	Ativado
2	Inicial	Anunciador	Botão	QS-02J5	Desativado
3	Inicial	Anunciador	Botão	QL-02J5	Desativado
4	Inicial	Anunciador	Botão	TL-02J5	Desativado
5	Inicial	Anunciador	Botão	TF-02J5	Desativado
6	Inicial	Anunciador	Botão	RE-02J5	Desativado
7	Inicial	Anunciador	Texto	02J5	Desativado
8	Inicial	Chave	Botoeira	12D1	Não Pressionado
9	Inicial	Chave	Religamento Automático	02J5	Desativado
10	Inicial	Chave	43	12J5	Posição 'D'
11	Inicial	Chave	CLT	12J5	Posição 'TEL'
12	Inicial	Chave	GPG	32J5-4	Fechado
13	Inicial	Chave	GPG	32J5-5	Fechado
14	Inicial	Chave	Comando seccionadora	32J5-6	Aberto
15	Inicial	Chave	CLT	32J5-7	Aberto
16	Inicial	Mostradores Digitais	Corrente	12D1	Ativado
17	Inicial	Anunciador	Botão	QS-12D1	Desativado
18	Inicial	Anunciador	Botão	QL-12D1	Desativado
19	Inicial	Anunciador	Botão	TL-12D1	Desativado
20	Inicial	Anunciador	Botão	TF-12D1	Desativado
21	Inicial	Anunciador	Botão	RE-12D1	Desativado
22	Inicial	Anunciador	Texto	12D1	Desativado
23	Inicial	Chave	GPG	12D1	Aberto
24	Inicial	Chave	GPG	12D1-1	Fechado
25	Inicial	Chave	GPG	12D1-2	Fechado

Tabela 6: Estado inicial da planta

5. Consulta sobre dispositivos e seus relativos painéis, por recurso:

```

SELECT zp_codigo_painel, zja_tipo_dispositivo, zjb_categoria_dispositivo,
zjc_especifica_dispositivo
FROM zja_tb_tipo_dispositivo, zjb_tb_categoria_dispositivo,
zjc_tb_especifica_dispositivo, zj_tb_dispositivo, zp_tb_painel,
zp_tb_painel_tem_zj_tb_dispositivo
WHERE zj_id = zj_dispositivo
AND zp_painel =1
AND zj_tipo_dispositivo = zja_id
AND zj_categoria_dispositivo = zjb_id
AND zj_especifica_dispositivo = zjc_id
ORDER BY zp_codigo_painel

```

zp_codigo_painel ▲	zja_tipo_dispositivo	zjb_categoria_dispositivo	zjc_especifica_dispositivo
DJ 12D1	Mostradores Digitais	Corrente	02J5
DJ 12D1	Anunciador	Botão	QS-02J5
DJ 12D1	Anunciador	Botão	QL-02J5
DJ 12D1	Anunciador	Botão	TL-02J5
DJ 12D1	Anunciador	Botão	TF-02J5
DJ 12D1	Anunciador	Botão	RE-02J5
DJ 12D1	Anunciador	Texto	02J5
DJ 12D1	Chave	Botoeira	12D1
DJ 12D1	Chave	Religamento Automático	02J5
DJ 12D1	Chave	43	12J5
DJ 12D1	Chave	CLT	12J5
DJ 12D1	Chave	GPG	32J5-4
DJ 12D1	Chave	GPG	32J5-5
DJ 12D1	Chave	Comando seccionadora	32J5-6
DJ 12D1	Chave	CLT	32J5-7
LT 02J5	Mostradores Digitais	Corrente	02J5
LT 02J5	Anunciador	Botão	QS-02J5
LT 02J5	Anunciador	Botão	QL-02J5
LT 02J5	Anunciador	Botão	TL-02J5
LT 02J5	Anunciador	Botão	TF-02J5
LT 02J5	Anunciador	Botão	RE-02J5
LT 02J5	Anunciador	Texto	02J5
LT 02J5	Chave	Botoeira	12D1
LT 02J5	Chave	Religamento Automático	02J5
LT 02J5	Chave	43	12J5
LT 02J5	Chave	CLT	12J5
LT 02J5	Chave	GPG	32J5-4
LT 02J5	Chave	GPG	32J5-5
LT 02J5	Chave	Comando seccionadora	32J5-6
LT 02J5	Chave	CLT	32J5-7

Tabela 7: Dispositivos e seus relativos painéis

Lista de consultas do módulo tutor (elaboração do cenário):

Consulta a um cenário:

```

SELECT zc_cenario, zca_identificacao, zka_evento, zi_estado_instalacao
FROM zc_tb_cenarios_associados z, zca_tb_tarefas zc, zi_tb_estado_se zi,
zk_tb_eventos zk, zka_tb_lista_eventos zka
WHERE z.zc_tarefas = zc.zca_id
AND z.zc_estado_se = zi.zi_id
AND z.zc_eventos = zk.zk_id
AND zk.zk_lista_eventos = zka.zka_id

```

zc_cenario	zca_identificacao	zka_evento	zi_estado_instalacao
Cenário 1	RTM-CGD-12J5	Nenhum evento ocorreu.	Normal de Operação

Tabela 8: Apresenta cenário

Lista de consultas do módulo tutor (resultado do treinamento):

1. *Consulta a nota resultante do treinamento individual:*

```
SELECT DISTINCT ze_operador, za_data_treinamento, zh_nota
FROM za_tb_treinamentos z, zb_tb_resultados zb, zh_tb_resultados_coletivos zh,
zg_tb_resultados_individuais zg, zc_tb_cenarios_associados zc, zf_tb_tutores zf,
ze_tb_operadores ze
WHERE zb.zb_resultados_coletivos = zh.zh_id
AND zb.zb_resultados_individuais = zg.zg_id
AND z.za_cenarios_associados = zc.zc_id
AND z.za_tutor = zf.zf_id
AND z.za_operador = ze.ze_id
```

ze_operador	za_data_treinamento	zh_nota
Marcelo Ramos	2010-01-25	8
Rivaldo José	2010-01-22	8

Tabela 9: Treinamento individual

2. *Consulta a nota resultante do treinamento coletivo:*

```
SELECT DISTINCT ze_operador, za_data_treinamento, za_turma, zg_nota
FROM za_tb_treinamentos z, zb_tb_resultados zb, zh_tb_resultados_coletivos zh,
zg_tb_resultados_individuais zg, zc_tb_cenarios_associados zc, zf_tb_tutores zf,
ze_tb_operadores ze
WHERE zb.zb_id =1
AND zb.zb_resultados_coletivos = zh.zh_id
AND zb.zb_resultados_individuais = zg.zg_id
AND z.za_cenarios_associados = zc.zc_id
AND z.za_tutor = zf.zf_id
AND z.za_operador = ze.ze_id
```

ze_operador	za_data_treinamento	za_turma	zg_nota
Marcelo Ramos	2010-01-25	1	9
Rivaldo José	2010-01-22	1	9

Tabela 10: Treinamento coletivo

6. CONSIDERAÇÕES FINAIS

A partir da validação preliminar foi possível constatar que o banco de dados atende os requisitos básicos do projeto, tanto do ponto de vista de armazenamento da informação quanto de sua disponibilização via WEB.

No que tange à interligação com o projeto do simulador, neste trabalho foi apenas investigado como seria realizá-la. Uma API (o JDBC) é usada para execução e manipulação de resultados a consultas SQL através de Java.

O armazenamento de informações para o módulo de treinamento foi validado a partir de consultas específicas deste módulo, submetidas diretamente ao BD. Em relação ao módulo tutor o BD, o mesmo processo de validação foi adotado.

Como proposta de trabalho futuro recomenda-se uma validação mais rigorosa do BD, a qual só será possível quando o BD for integrado ao código do simulador e aos demais módulos.

BIBLIOGRAFIA

(Souza, 2008) Souza, A. B. Banco de dados para simulador de cursos da CHESF - Relatório do Trabalho de Conclusão de Curso, Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, Dezembro de 2008.

(DBDesingner, 2010) DBDesingner home page, disponível em <http://www.fabforce.net/dbdesigner4/> acessado em março de 2010.

(FreeWRL, 2010) FreeWRL home page, disponível em <http://freewrl.sourceforge.net/index.html> acessado em março de 2010.

(Guerrero, 2006) Guerrero, C. V. S. Modelo Conceitual de Cenários de Acidentes causados pelo Erro Humano em Sistemas Industriais Críticos com Foco na Concepção de Interfaces Ergonômicas - Tese de Doutorado, Pós-Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, maio de 2006.

(Java, 2010) Java home page, disponível em http://www.java.com/pt_BR/ acessado em março de 2010.

(Nascimento, 2010) Nascimento, J. A. N. Processo para Concepção de Estratégias para Prevenção do Erro na Operação de Sistemas Elétricos - Tese de Doutorado, Pós-Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, fevereiro de 2010.

(phpMyAdmin, 2010) home page, disponível em http://www.phpmyadmin.net/home_page/index.php acessado em março 2010.

(Vieira et al., 2010) VIEIRA, M. F. Q. ; NASCIMENTO NETO, José Alves Do ; SCAICO, A. ; SANTONI, Charles ; MERCANTINI, Jean Marc . A Model Based Operator Training Simulator to support Human Behavior Studies. Transactions of the Society for Computer Simulation, v. 86, p. 41-51, 2010.

ANEXO A - DICIONÁRIO DE DADOS

O dicionário de dados apresentado possibilita a identificação da descrição do campo e das tabelas. Também fornece a estrutura de tipo de variável e exibe se o campo pode ser nulo, e seu respectivo padrão.

Estrutura da tabela **zaa_tb_forma_treinamento**

Indica se o treinamento está ocorrendo de forma remota ou local.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zaa_id</i>	int(11)	Não		ID da forma de treinamento. Números inteiros auto incrementados
<i>zaa_tb_forma_treinamento</i>	varchar(20)	Sim	NULL	Descreve a forma de treinamento

Estrutura da tabela **zab_tb_objetivo_treinamento**

Descreve sob qual equipamento será desenvolvido o treinamento.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zab_id</i>	int(11)	Não		ID do objetivo de treinamento. Números inteiros auto incrementados
<i>zab_tb_objetivo_treinamento</i>	text	Sim	NULL	Informa o objetivo do treinamento

Estrutura da tabela **zac_tb_tipo_treinamento**

Indica qual tipo de treinamento será realizado.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zac_id</i>	int(11)	Não		ID do tipo de treinamento. Números inteiros auto incrementados
<i>zac_tb_tipo_treinamento</i>	varchar(20)	Sim	NULL	Indica o tipo de treinamento

Estrutura da tabela **zad_estado_treinamento**

Possui a informação sobre o estado do treinamento.

Campo	Tipo	Nulo	Padrão	Descrição
<i>id_zad</i>	int(11)	Não		ID para estado do treinamento
zad_estado_treinamento	varchar(20)	Sim	NULL	Explicita o estado do treinamento

Estrutura da tabela **za_tb_treinamentos**

Possui dos elementos envolvidos em um treinamento. Informando data, tempo e pessoal envolvido, bem como o cenário ambientado e seu resultado.

Campo	Tipo	Nulo	Padrão	Descrição
<i>za_id</i>	int(11)	Não		ID do treinamento. São números inteiros auto incrementados
za_operador	int(11)	Não		Operador do treinamento
za_tutor	int(11)	Não		Tutor responsável
za_cenarios_associados	int(11)	Não		Cenário associado ao treinamento
za_resultados	int(11)	Não		Resultado do treinamento
za_tipo_treinamento	int(11)	Não		Tipo do treinamento
za_estado_treinamento	int(11)	Não		Estado do treinamento
za_objetivo_treinamento	int(11)	Não		Objetivo do treinamento
za_forma_treinamento	int(11)	Não		Forma do treinamento
za_data_treinamento	date	Sim	NULL	Data do treinamento
za_tempo_necessario_treinamento	time	Sim	NULL	Tempo necessário para realização desse treinamento
za_numero_operadores	int(11)	Sim	NULL	Quantidade de operadores envolvidos no treinamento
za_turma	int(11)	Sim	NULL	Turma submetida ao treinamento

Estrutura da tabela **zb_tb_resultados**

Relaciona os resultados individuais e os coletivos de um operador.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zb_id</i>	int(11)	Não		ID do resultado
zb_resultados_coletivos	int(11)	Não		Relaciona o resultado coletivo ao resultado geral
zb_resultados_individuais	int(11)	Não		Relaciona o resultado individual ao resultado geral

Estrutura da tabela **zca_tb_tarefas**

Identifica o conjunto de passos da tarefa indicado de um roteiro de manobras. O operador tem a sua disposição no PDA as tarefas a serem seguidas.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zca_id</i>	int(11)	Não		ID da tarefa
zca_identificacao	varchar(50)	Sim	NULL	Campo para identificação da tarefa
zca_edicao	int(11)	Sim	NULL	Campo para indicar a edição da tarefa
zca_data	date	Sim	NULL	Data da elaboração da tarefa
zca_motivo	varchar(50)	Sim	NULL	Motivo da tarefa
zca_observacao	text	Sim	NULL	Campo para observação

Estrutura da tabela **zc_tb_cenarios_associados**

Constitui uma das principais tabelas do banco de dados. Caracteriza o cenário com suas tarefas, eventos e o estado da subestação.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zc_id</i>	int(11)	Não		ID do cenário
zc_tarefas	int(11)	Não		Relaciona as tarefas para um cenário
zc_eventos	int(11)	Não		Relaciona eventos envolvidos no cenário
zc_estado_se	int(11)	Não		Relaciona o estado da subestação ao cenário
zc_cenario	varchar(50)	Sim	NULL	Dar nome ao cenário

Estrutura da tabela **zd_tb_instalacoes**

Descreve e indica os recursos disponíveis em uma instalação.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zd_id</i>	int(11)	Não		ID da instalação
<i>zd_recurso</i>	int(11)	Não		Relaciona recurso de uma instalação
<i>zd_instalacao</i>	varchar(50)	Sim	NULL	Nome da instalação
<i>zd_descricao</i>	text	Sim	NULL	Descreve a instalação

Estrutura da tabela **zd_tb_instalacoes_tem_zc_tb_cenarios_associados**

Tabela que relaciona uma instalação a um cenário.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zd_tb_instalacoes_zd_id</i>	int(11)	Não		Relaciona ID da instalação
<i>zc_tb_cenarios_associados_zc_id</i>	int(11)	Não		Relaciona ID do cenário

Estrutura da tabela **ze_tb_operadores**

Expressa características ao operador.

Campo	Tipo	Nulo	Padrão	Descrição
<i>ze_id</i>	int(11)	Não		ID do operador.
<i>ze_instalacoes</i>	int(11)	Não		Instalação do operador
<i>ze_operador</i>	varchar(50)	Sim	NULL	Nome
<i>ze_matricula</i>	int(11)	Sim	NULL	Matrícula
<i>ze_treinamento_anterior</i>	varchar(10)	Sim	NULL	Campo que indica se o operador possui treinamento anterior
<i>ze_lentes_corretivas</i>	varchar(10)	Sim	NULL	Campo que indica se o operador possui lentes corretivas
<i>ze_data_nascimento</i>	date	Sim	NULL	Data de nascimento
<i>ze_experiencia_anterior</i>	varchar(20)	Sim	NULL	Campo que indica se o operador possui experiência anterior
<i>ze_anos_na_funcao</i>	int(11)	Sim	NULL	Campo que indica quantos ano o operador atua na função

Estrutura da tabela **zf_tb_tutores**

Identifica o tutor.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zf_id</i>	int(11)	Não		ID do tutor
<i>zf_instalacao</i>	int(11)	Não		Instalação ao qual o tutor pertence
<i>zf_tutor</i>	varchar(50)	Sim	NULL	Nome do tutor
<i>zf_matricula</i>	int(11)	Sim	NULL	Matrícula do tutor

Estrutura da tabela **zg_tb_resultados_individuais**

Indica o resultado e descreve o erro individual.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zg_id</i>	int(11)	Não		ID do resultado individual
<i>zg_nota</i>	float	Sim	NULL	Nota obtida pelo operador
<i>zg_descricao</i>	text	Sim	NULL	Descrição dos erros do operador
<i>zg_numero_erro</i>	int(11)	Sim	NULL	Número de erros no treinamento

Estrutura da tabela **zh_tb_resultados_coletivos**

Indica o resultado e descreve o erro coletivo.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zh_id</i>	int(11)	Não		ID do resultado coletivo
<i>zh_nota</i>	float	Sim	NULL	Nota obtida pelo operador
<i>zh_descricao</i>	text	Sim	NULL	Descrição dos erros dos operadores
<i>zh_numero_erro</i>	int(11)	Sim	NULL	Número de erros no treinamento

Estrutura da tabela **zi_tb_estado_se**

Indica o estado da subestação. Pode ser para um determinado cenário ou apenas para o estado normal de operação ou atual.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zi_id</i>	int(11)	Não		ID do estado da subestação
<i>zi_estado_instalacao</i>	varchar(50)	Sim	NULL	Tipo do estado em que se encontra a subestação

Estrutura da tabela **zi_tb_estado_se_tem_zl_tb_estado_dispositivo**

Estabelece para cada subestação o estado de seus dispositivos.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zi_estado_se</i>	int(11)	Não		Relaciona um ID estado da subestação a um dispositivo
<i>zl_estado_dispositivo</i>	int(11)	Não		Relaciona ID estado do dispositivo a um estado da subestação

Estrutura da tabela **zja_tb_tipo_dispositivo**

Refere-se à classe de nível mais alto dos dispositivos.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zja_id</i>	int(11)	Não		ID do tipo de dispositivo
<i>zja_tipo_dispositivo</i>	varchar(30)	Sim	NULL	Indica o tipo do dispositivo

Estrutura da tabela **zjb_tb_categoria_dispositivo**

Referente à classe intermediária de composição do dispositivo.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zjb_id</i>	int(11)	Não		ID categoria do dispositivo
<i>zjb_categoria_dispositivo</i>	varchar(30)	Sim	NULL	Categoria a qual o dispositivo pertence

Estrutura da tabela **zjc_tb_especifica_dispositivo**

Classe de nível mais baixo do dispositivo.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zjc_id</i>	int(11)	Não		ID da especificação do dispositivo
<i>zjc_especifica_dispositivo</i>	varchar(30)	Sim	NULL	Especifica o dispositivo

Estrutura da tabela **zj_tb_dispositivo**

Reúne as informações sobre o dispositivo. Essas informações são obtidas através de suas três classes, respectivamente do maior nível para o menor: tipo de dispositivo, categoria do dispositivo, especificação do dispositivo. Dessa forma garantimos que o dispositivo será único numa instalação.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zj_id</i>	int(11)	Não		ID de um dispositivo
<i>zj_tipo_dispositivo</i>	int(11)	Não		Faz referência ao tipo do dispositivo
<i>zj_categoria_dispositivo</i>	int(11)	Não		Faz referência a categoria do dispositivo
<i>zj_especifica_dispositivo</i>	int(11)	Não		Referencia a especificação do dispositivo

Estrutura da tabela **zka_tb_lista_eventos**

Nomeia os eventos que podem ocorrer em um cenário.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zka_id</i>	int(11)	Não		ID do evento
<i>zka_evento</i>	text	Sim	NULL	Descreve o evento

Estrutura da tabela **zkb_tb_estado_planta**

Define se na planta um estado para cada dispositivo envolvido no evento.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zkb_id</i>	int(11)	Não		ID dos estados dos dispositivos na planta
<i>zkb_lista_estado</i>	int(11)	Não		Define um estado
<i>zkb_dispositivo</i>	int(11)	Não		Apresenta um dispositivo

Estrutura da tabela **zkc_tb_estado_atuador**

Descreve o estado de um dispositivo, informando se irá atuar em uma planta. Fato que poderá alterar o estado da mesma.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zkc_id</i>	int(11)	Não		ID do estado do atuador
<i>zkc_dispositivo</i>	int(11)	Não		Faz referência ao dispositivo atuador
<i>zkc_lista_estado</i>	int(11)	Não		Informa o estado do dispositivo

Estrutura da tabela **zkd_tb_planta**

Informa se o estado da planta é inicial ou final. Podendo ser acrescentado mais algum outro estado.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zkd_id</i>	int(11)	Não		ID estado da planta
<i>zkd_estado</i>	varchar(50)	Sim	NULL	Indica o estado a qual a planta se encontra

Estrutura da tabela **zkd_tb_planta_tem_zkb_tb_estado_planta**

Relaciona um estado da planta ao conjunto de dispositivos que a compõe, com o seu respectivo estado.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zkd_planta</i>	int(11)	Não		Aponta para o estado da planta
<i>zkb_estado_planta</i>	int(11)	Não		Referencia um conjunto de dispositivos com estado

Estrutura da tabela **zk_tb_eventos**

Tabela que reúne informações sobre eventos ocorridos para um cenário. Possui informações sobre a planta, descrição do evento. Informa ainda o motivo do desencadeamento do evento, podendo ter sido por um dispositivo atuador, pelo tempo, ou pelo tutor.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zk_id</i>	int(11)	Não		ID do evento
<i>zk_planta</i>	int(11)	Não		Referencia o estado da planta
<i>zk_lista_eventos</i>	int(11)	Não		Faz referência a descrição do evento
<i>zk_estado_atuador</i>	int(11)	Não		Indica o estado e o atuador caso o evento seja desencadeado pelo mesmo
<i>zk_pelo_tempo</i>	time	Sim	NULL	Campo para indicação se o evento é pelo tempo
<i>zk_por_dispositivo</i>	int(11)	Sim	NULL	Campo para indicar se o evento é pelo dispositivo atuador
<i>zk_pelo_tutor</i>	int(11)	Sim	NULL	Campo para indicar se o evento é desencadeado pelo tutor

Estrutura da tabela **zk_tb_eventos_tem_zl_tb_estado_dispositivo**

Tabela que relaciona evento a estados do dispositivos de uma instalação.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zl_estado_dispositivo</i>	int(11)	Não		Estado do dispositivo
<i>zk_eventos</i>	int(11)	Não		Faz referência a evento

Estrutura da tabela **zla_tb_lista_estado**

Tabela que guarda todos estados possíveis de um dispositivo.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zla_id</i>	int(11)	Não		ID da lista de estados
<i>zla_estado</i>	varchar(20)	Sim	NULL	Estado de um dispositivo

Estrutura da tabela **zl_tb_estado_dispositivo**

Tabela que indica para um determinado dispositivo de uma instalação, qual seu estado em certo momento.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zl_id</i>	int(11)	Não		ID do estado do dispositivo
<i>zl_instalacao</i>	int(11)	Não		Instalação do dispositivo
<i>zl_lista_estado</i>	int(11)	Não		Estado do dispositivo
<i>zl_dispositivo</i>	int(11)	Não		Referencia o dispositivo
<i>zl_momento</i>	time	Sim	NULL	Indica o momento para o estado

Estrutura da tabela **zpa_tb_tipo_painel**

Indica o tipo de equipamento ou linha ao qual o painel se refere.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zap_id</i>	int(11)	Não		ID do tipo de painel
<i>zja_tipo_painel</i>	varchar(20)	Sim	NULL	Indica o tipo de painel

Estrutura da tabela **zpb_tb_aviso_painel**

Tabela que armazena o tipo de avisos possíveis e o texto nele contido.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zpb_id</i>	int(11)	Não		ID para aviso
<i>zpb_tipo_aviso</i>	varchar(20)	Sim	NULL	Tipo de aviso
<i>zpb_aviso</i>	text	Sim	NULL	Texto do aviso

Estrutura da tabela **zp_tb_painel**

Tabela que possui as características do painel num armário.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zp_id</i>	int(11)	Não		ID para o painel
<i>zp_aviso_painel</i>	int(11)	Não		Referencia o aviso
<i>zp_tipo_painel</i>	int(11)	Não		Aponta para o tipo do painel
<i>zp_codigo_painel</i>	varchar(20)	Sim	NULL	Define o código do painel
<i>zp_marcacao</i>	varchar(10)	Sim	NULL	Informa se o painel está marcado no simulador

Estrutura da tabela **zp_tb_painel_tem_zj_tb_dispositivo**

A tabela relaciona para cada painel os dispositivos que o mesmo possui.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zj_dispositivo</i>	int(11)	Não		Faz referência ao dispositivo
<i>zp_painel</i>	int(11)	Não		Faz referência ao painel

Estrutura da tabela **zqa_objetos_interacao**

Guarda os objetos que podem interagir com o operador.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zqa_id</i>	int(11)	Não		ID do objeto de interação
<i>objetos_interacao</i>	varchar(20)	Sim	NULL	Nome do objeto de interação

Estrutura da tabela **zq_tb_recurso**

Tabela que possui uma lista de recursos possíveis e o objeto de interação.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zq_id</i>	int(11)	Não		ID do recurso
<i>zq_objetos_interacao</i>	int(11)	Não		Objeto de interação de um recurso
<i>zq_nome</i>	varchar(50)	Sim	NULL	Nome do recurso

Estrutura da tabela **zq_tb_recurso_tem_zp_tb_painel**

A tabela relaciona para cada recurso os painéis disponíveis.

Campo	Tipo	Nulo	Padrão	Descrição
<i>zq_tb_recurso_zq_id</i>	int(11)	Não		ID do recurso
<i>zp_tb_painel_zp_id</i>	int(11)	Não		ID do painel

ANEXO B - TABELAS do BD (modelo físico)

Extraindo dados da tabela **zaa_tb_forma_treinamento**:

<u>zaa_id</u>	<u>zaa_tb_forma_treinamento</u>
1	Local
2	Remoto

Extraindo dados da tabela **zab_tb_objetivo_treinamento**:

<u>zab_id</u>	<u>zab_tb_objetivo_treinamento</u>
1	Liberação do 12J5
2	Desenergizar linha 230 KV 4637
3	Desenergizar Transformador 3457
4	Energizar transformador TT45

Extraindo dados da tabela **zac_tb_tipo_treinamento**:

<u>zac_id</u>	<u>zac_tb_tipo_treinamento</u>
1	Operação
2	Segurança
3	Reciclagem

Extraindo dados da tabela **zad_estado_treinamento**:

<u>id_zad</u>	<u>zad_estado_treinamento</u>
1	Finalizado
2	Em execução
3	Em espera

Extraindo dados da tabela **za_tb_treinamentos**:

<u>za_id</u>	<u>za_operador</u>	<u>za_tutor</u>	<u>za_cenarios_associados</u>	<u>za_resultados</u>	<u>za_tipo_treinamento</u>
1	1	1	1	1	1
2	2	1	1	2	1

<u>za_id</u>	<u>za_estado_treinamento</u>	<u>za_objetivo_treinamento</u>	<u>za_forma_treinamento</u>	<u>za_data_treinamento</u>
1	1	1	1	2010-01-25
2	1	1	1	2010-01-22

<u>za_id</u>	<u>za_tempo_necessario_treinamento</u>	<u>za_numero_operadores</u>	<u>za_turma</u>
1	00:00:30	1	1
2	00:00:30	1	1

Extraindo dados da tabela **zb_tb_resultados**:

zb_id	zb_resultados_coletivos	zb_resultados_individuais
1	1	1
2	1	2

Extraindo dados da tabela **zca_tb_tarefas**:

zca_id	zca_identificacao	zca_edicao	zca_data	zca_motivo	zca_observacao
1	RTM-CGD-12J5	7	2006-09-13	Padronização do RTM	- A chave 43 - 12J5 na posição 'T', transfere o trip das proteções do 12J5 inclusive o esquema de subfrequência para o 12D1, e desativa trip das proteções próprias do 12D1. - Caso necessário liberação simultânea do 12J5 e proteções associadas proceder conforme ICP - CGD.

Extraindo dados da tabela **zc_tb_cenarios_associados**:

zc_id	zc_tarefas	zc_eventos	zc_estado_se	zc_cenario
1	1	1	1	Cenário 1

Extraindo dados da tabela **zd_tb_instalacoes**:

zd_id	zd_recurso	zd_instalacao	zd_descricao
1	1	SE Campina Grande (CGD)	II Atende a região de Campina Grande

Extraindo dados da tabela **zd_tb_instalacoes_tem_zc_tb_cenarios_associados**:

zd_tb_instalacoes_zd_id	zc_tb_cenarios_associados_zc_id
1	1

Extraindo dados da tabela **ze_tb_operadores**:

ze_id	ze_instalacoes	ze_operador	ze_matricula	ze_treinamento_anterior
1	1	Marcelo Ramos	2	Não
2	1	Rivaldo José	10	Sim
3	1	Rubens M	3	Sim
4	1	José	7	Não
5	1	Jonas da Silva	2	Sim

ze_id	ze_lentes_corretivas	ze_data_nascimento	ze_experiencia_anterior	ze_anos_na_funcao
1	Não	1957-03-07	Sim	33
2	Não	1959-02-01	Não	30
3	Sim	1958-08-02	Não	27
4	Sim	1950-10-08	Sim	27
5	Não	1958-03-29	Não	27

Extraindo dados da tabela **zf_tb_tutores**:

zf_id	zf_instalacao	zf_tutor	zf_matricula
1	1	Alves	1
2	2	Yuska	2

Extraindo dados da tabela **zg_tb_resultados_individuais**:

zg_id	zg_nota	zg_descricao	zg_numero_erro
1	9	Esqueceu passo 2	1
2	8.5	Demorou excessivamente	1

Extraindo dados da tabela **zh_tb_resultados_coletivos**:

zh_id	zh_nota	zh_descricao	zh_numero_erro
1	8	Não interagiu com a turma e executou manobra errada.	2
2	10	Corretamente	0

Extraindo dados da tabela **zi_tb_estado_se**:

zi_id	zi_estado_instalacao
1	Normal de Operação
2	Inicial do Cenário
3	Atual

Extraindo dados da tabela **zi_tb_estado_se_tem_zl_tb_estado_dispositivo**:

zi_estado_se	zl_estado_dispositivo
1	1

Extraindo dados da tabela **zja_tb_tipo_dispositivo**

zja_id	zja_tipo_dispositivo
1	Mostradores Analógicos
2	Mostradores Digitais
3	Anunciador
4	Chave
5	NULL

Extraindo dados da tabela **zjb_tb_categoria_dispositivo**

zjb_id	zjb_categoria_dispositivo
1	Potência
2	Tensão
3	Corrente
4	Botão
5	Texto
6	Alarme
7	GPG
8	Comando seccionadora
9	CLT
10	43
11	Botoeira
12	LRM
13	Ajuste de tensão
14	Temperatura da água
15	Teste de buzina
16	Religamento Automático
17	Automático / Manual
18	Pressão Sinótico
19	CS
20	Normal By-Pass
21	Ajuste de tensão
22	Tensão (chave)
23	NULL

Extraindo dados da tabela zjc_tb_especifica_dispositivo

<u>zjc_id</u>	<u>zjc_especifica_dispositivo</u>
1	02J5
2	12D1
3	QS-02J5
4	QL-02J5
5	TL-02J5
6	TF-02J5
7	RE-02J5
8	QS-12D1
9	QL-12D1
10	TL-12D1
11	TF-12D1
12	RE-12D1
13	12J5
14	12D1
15	32J5-4
16	32J5-5
17	32J5-6
18	32J5-7
19	12D1
20	12D1-1
21	12D1-2
22	NULL

Extraindo dados da tabela **zj_tb_dispositivo**

zj_id	zj_tipo_dispositivo	zj_categoria_dispositivo	zj_especifica_dispositivo
1	2	3	1
2	3	4	3
3	3	4	4
4	3	4	5
5	3	4	6
6	3	4	7
7	3	5	1
8	4	11	19
9	4	16	1
10	4	10	13
11	4	9	13
12	4	7	15
13	4	7	16
14	4	8	17
15	4	9	18
16	2	3	2
17	3	4	8
18	3	4	9
19	3	4	10
20	3	4	11
21	3	4	12
22	3	5	2
23	4	7	14
24	4	7	20
25	4	7	21
26	3	6	1
27	5	23	22

Extraindo dados da tabela **zka_tb_lista_eventos:**

zka_id	zka_evento
1	Nenhum evento ocorreu.
2	Acionar alarme do painel 02J5.
3	Liberação do 12J5

Extraindo dados da tabela **zkb_tb_estado_planta**:

zkb_id	zkb_lista_estado	zkb_dispositivo			
1	1	1	27	2	2
2	2	2	28	2	3
3	2	3	29	2	4
4	2	4	30	2	5
5	2	5	31	2	6
6	2	6	32	2	7
7	2	7	33	14	8
8	14	8	34	2	9
9	2	9	35	10	10
10	8	10	36	11	11
11	12	11	37	6	12
12	7	12	38	6	13
13	7	13	39	7	14
14	6	14	40	6	15
15	6	15	41	2	16
16	1	16	42	2	17
17	2	17	43	2	18
18	2	18	44	2	19
19	2	19	45	2	20
20	2	20	46	2	21
21	2	21	47	2	22
22	2	22	48	7	23
23	6	23	49	7	24
24	7	24	50	7	25
25	7	25			
26	2	1			

Extraindo dados da tabela **zkc_tb_estado_atuador**:

zkc_id	zkc_dispositivo	zkc_lista_estado
1	27	2
2	26	1

Extraindo dados da tabela **zkd_tb_planta**:

zkd_id	zkd_estado
1	Inicial
2	Final

Extraindo dados da tabela **zkd_tb_planta_tem_zkb_tb_estado_planta**:

zkd_planta	zkb_estado_planta		
1	1	2	27
1	2	2	28
1	3	2	29
1	4	2	30
1	5	2	31
1	6	2	32
1	7	2	33
1	8	2	34
1	9	2	35
1	10	2	36
1	11	2	37
1	12	2	38
1	13	2	39
1	14	2	40
1	15	2	41
1	16	2	42
1	17	2	43
1	18	2	44
1	19	2	45
1	20	2	46
1	21	2	47
1	22	2	48
1	23	2	49
1	24	2	50
1	25		

Extraindo dados da tabela **zk_tb_eventos**:

zk_id	zk_planta	zk_lista_eventos	zk_estado_atuador	zk_pelo_tempo	zk_por_dispositivo	zk_pelo_tutor
1	1	1	1	NULL	NULL	NULL
2	2	3	1	NULL	NULL	NULL

Extraindo dados da tabela **zk_tb_eventos_tem_zl_tb_estado_dispositivo**:

zl_estado_dispositivo	zk_eventos
1	3

Extraindo dados da tabela **zla_tb_lista_estado**:

zla_id	zla_estado
1	Ativado
2	Desativado
3	Aceso
4	Apagado
5	Piscando
6	Aberto
7	Fechado
8	Posição '0'
9	Posição 'ET'
10	Posição 'T'
11	Posição 'LOC'
12	Posição 'TEL'
13	Pressionado
14	Não Pressionado
15	By pass
16	Controle
17	Posição 'L'
18	Posição 'R'
19	Posição 'M' (manut.)
20	Tensão X
21	Tensão Y
22	Tensão Z
23	Nenhuma tensão
24	Alarme 1
25	Alarme 2
26	Alarme 3
27	Posição 'M' (manual)
28	Posição 'A'
29	Posição 'D'
30	Aumentando
31	Repouso
32	Diminuindo

Extraindo dados da tabela **zpa_tb_tipo_painel**:

zap_id	zja_tipo_painel
1	Equipamentos
2	Linha
3	Banco de Capacitores
4	Transformadores
5	Disjuntores
6	Compensador Estático

Extraindo dados da tabela **zl_tb_estado_dispositivo**:

zl_id	zl_instalacao	zl_lista_estado	zl_dispositivo	zl_momento
1	1	1	26	09:15:00

Extraindo dados da tabela **zpb_tb_aviso_painel**:

zpb_id	zpb_tipo_aviso	zpb_aviso
1	NULL	NULL

Extraindo dados da tabela **zp_tb_painel**:

zp_id	zp_aviso_painel	zp_tipo_painel	zp_codigo_painel	zp_marcacao
1	1	2	LT 02J5	sim
2	1	5	DJ 12D1	sim

Extraindo dados da tabela **zp_tb_painel_tem_zj_tb_dispositivo**

zj_dispositivo	zp_painel
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1
15	1
16	2
17	2
18	2
19	2
20	2
21	2
22	2
23	2
24	2
25	2
26	2

Extraindo dados da tabela **zqa_objetos_interacao:**

zqa_id	objetos_interacao
1	Rádio
2	Telefone
3	PDA

Extraindo dados da tabela **zq_tb_recurso**:

<u>zq_id</u>	<u>zq_objetos_interacao</u>	<u>zq_nome</u>
1	3	Recurso 1

Extraindo dados da tabela **zq_tb_recurso_tem_zp_tb_painel**:

<u>zq_tb_recurso_zq_id</u>	<u>zp_tb_painel_zp_id</u>
1	1
1	2