

Banco de dados da Conecta Recycle

Levantamento e análise de requisitos

Iniciamos o desenvolvimento do banco de dados com o levantamento e a análise de requisitos onde observamos na documentação do software quais dados seriam indispensáveis, às exigências necessárias para o usuário solucionar um problema e alcançar objetivos.

Modelo conceitual

Após o levantamento e a análise de requisitos, começamos o modelo conceitual que é um tipo de modelo descritivo conceitual com a criação do MER (Modelo Entidade-Relacionamento). Ele é utilizado para descrever de forma abstrata os objetos que existirão no banco de dados. Dessa forma, descrevemos detalhadamente através da escrita as entidades, atributos e suas classificações. Para sua concepção utilizamos o Word.

Modelo Entidade - Relacionamento (MER) Plataforma Conecta Recycle

USUARIO(id_usuario, nome, data_nascimento, endereco, cpf, telefone, email, senha, pontuacao)
Entidade forte: USUARIO

Atributo identificador: id
Atributo composto: endereco
Atributos simples: pontuação
Atributo multivalorado: telefone
Atributos obrigatórios: email, senha, cpf, nome

GERENTE(id_gerente, cpf, nome, email, telefone, departamento, fk_id_empresa)
Entidade fraca: GERENTE

Atributo identificador: id_gerente
Atributos obrigatórios: cpf, nome, email, fk_id_empresa
Atributo multivalorado: telefone
Atributo descritivo: departamento

EMPRESA(id_empresa, nome, cnpj, cnae, endereco, email, telefone, rede_social, logo)
Entidade forte: EMPRESA

Atributo identificador: id_empresa
Atributos obrigatórios: nome, email, cnpj, cnae, logo
Atributo multivalorado: telefone
Atributo composto: endereco (logradouro, numero, bairro, cidade, estado)

COLETA(id_coleta, relatorio_produtos, data, hora)

Entidade: COLETA

Atributo identificador: id_coleta

Atributo descritivo: relatorio_produtos

Atributos simples: data, hora

PONTO_COLETA(id_ponto_coleta, fk_id_empresa, endereco)

Entidade fraca: PONTO_COLETA

Atributo identificador: id_ponto

Atributos obrigatórios: fk_id_empresa

Atributo composto: endereco

CONTRATO(id_contrato, codigo_data_inicio, data_final, fk_id_empresa)

Entidade fraca: CONTRATO

Atributo identificador: id_contrato

Atributos obrigatórios: nome, data, quantidade, fk_id_empresa

DESCARTE(id_descarte, data, hora, fk_id_ponto, fk_id_empresa, produto)

Entidade fraca: DESCARTE

Atributo identificador: id_descarte

Atributo simples: data, hora

Atributo obrigatório: id_ponto, fk_id_empresa, produto

Modelo Entidade - Relacionamento (MER)

Blog Conecta Recycle

ARTIGO(id_artigo, título, subtítulo, fk_id_autor, texto, data_postagem)

Entidade fraca: ARTIGO

Atributo identificador: id_artigo

Atributo simples: título, subtítulo, data_postagem, fk_id_autor

Atributo descritivo: texto

NOTICIA(id_noticia, titulo, subtítulo, texto, fk_id_autor, data_postagem)

Entidade fraca: NOTICIA

Atributo identificador: id_noticia

Atributo simples: subtítulo, título, data_postagem, fk_id_autor

Atributo descritivo: texto

AUTOR(id_autor, nome, data_nascimento, formacao)

Entidade forte: AUTOR

Atributo identificador: id_autor

Atributo simples: nome, formacao, data_nascimento

Em seguida, demos início a criação do DER (Diagrama Entidade-Relacionamento), que também é utilizado para representar de forma abstrata a modelagem do banco de dados, entretanto graficamente, através de formas e símbolos. Aqui identificamos que o cadastro gera três tipos de usuários, o descartador, a empresa parceira e a empresa coletora, entretanto os dados serão armazenados apenas em uma entidade, enquanto a cardinalidade é um por um (1:1). O usuário poderá efetuar login uma vez e no máximo muitos (1:N). O login poderá ser feito nenhuma ou muitas vezes pelo usuário (1:N). A entidade empresa pode ter pontos de coleta, a cardinalidade um por muitos (1:N). Ela assina contrato uma vez (1:1), precisa de um gerente (1:1). A empresa coletora faz a coleta no ponto de coleta uma ou muitas vezes (1:1). O usuário realiza descarte no ponto de coleta nenhuma ou muitas vezes (0:N). A entidade blog tem autor dos textos sendo que ele fará nenhuma ou muitas atualizações (0:N). Na formulação do DER utilizamos o BRModelo na versão 3.31.

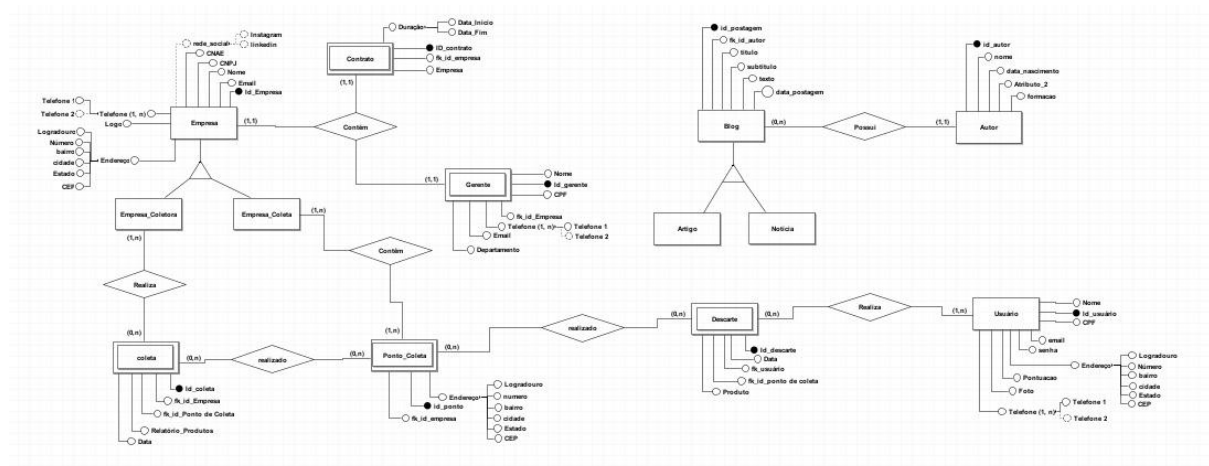


Imagem 1: Diagrama de Entidade-relacionamento

Modelo Lógico

Posteriormente ao modelo conceitual, elaboramos o modelo lógico que trata da definição da estrutura do banco de dados onde serão armazenados os dados, descrevendo detalhadamente suas tabelas, campos e relacionamentos. Desse modo, definimos detalhadamente a apresentação da arquitetura do banco de dados, melhoramos o entendimento da estrutura técnica e tomamos decisões que beneficiem o projeto. Em sua produção utilizamos também o BRModelo.

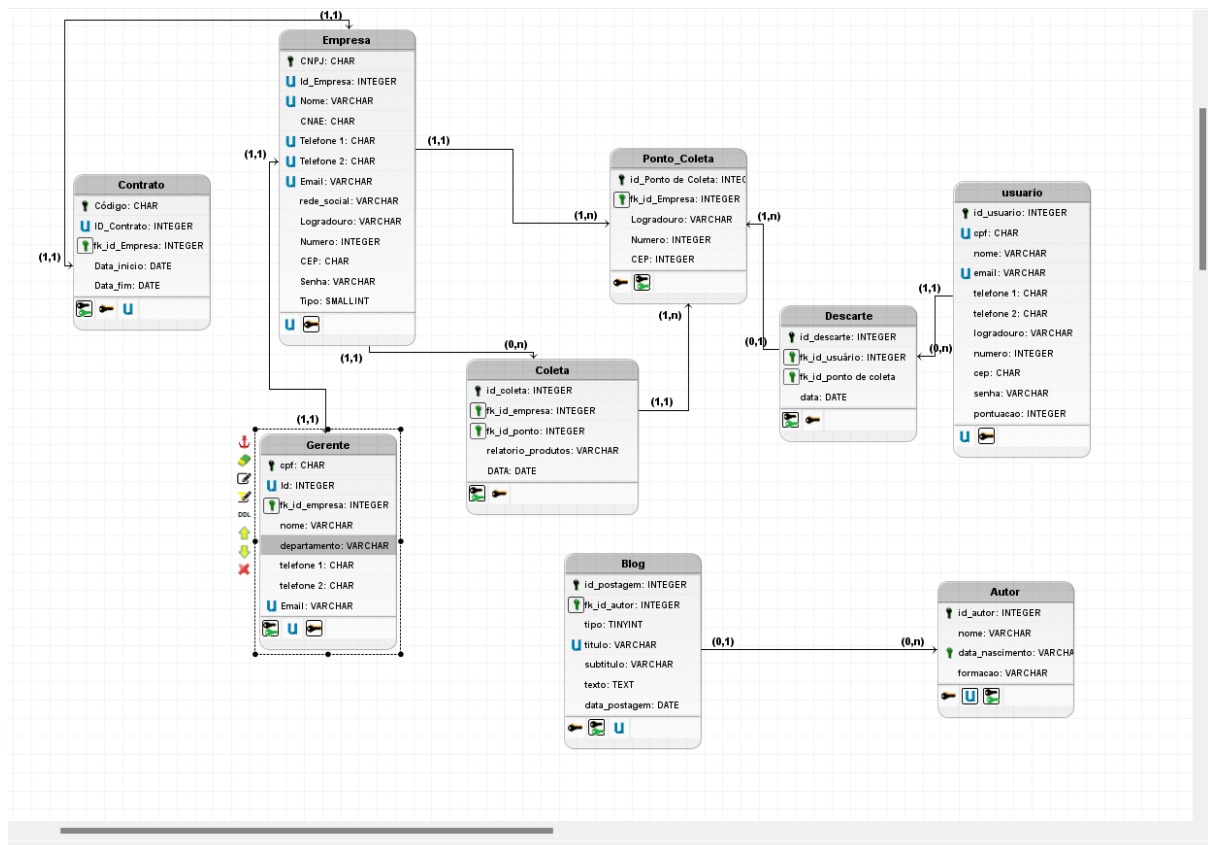


Imagem 2: modelo lógico

Modelo físico

Por fim, realizamos a implementação do projeto através do modelo físico que descreve, específica com base no modelo lógico. E ajuda a gerar esquemas com detalhes mais específicos como o tipo de atributos, nomes das tabelas e dos campos delas. O SGBD(Sistema Gerenciador de Banco de Dados) que aplicamos para este modelo foi o MySQL em sua versão 8.0.

Código modelo físico

```
create database recycle;
use recycle;
```

```
create table Empresa(
id_empresa int auto_increment not null unique,
nome_emp varchar(50) not null,
cnpj char(14) primary key not null,
cnae char(7),
email varchar(256) unique not null,
senha varchar(30) not null,
telefone1 varchar(13) unique not null,
telefone2 varchar(13),
```

```
logradouro varchar(100) not null,  
numero varchar(5) not null,  
bairro varchar(100) not null,  
cidade varchar(100) not null,  
Estado varchar(100) not null,  
cep char(8) not null,  
instagram varchar(300) unique,  
linkedin varchar(300) unique,  
tipo smallint not null,  
logo varchar(500) not null  
);
```

```
create table gerente(  
id_gerente int auto_increment unique,  
cpf char(14) not null primary key,  
fk_cnpj_empresa int not null unique,  
nome_ger varchar(100) not null,  
departamento varchar(50) not null,  
email varchar(256) not null,  
telefone1 varchar(13) unique not null,  
telefone2 varchar(13)  
);
```

```
create table contrato(  
id_contrato int auto_increment not null unique,  
fk_cnpj_empresa int not null,  
data_inicio date not null,  
data_fim date not null  
);
```

```
create table coleta(  
id_coleta int auto_increment not null unique,  
fk_id_ponto int not null,  
`data` datetime not null,  
relatorio text not null  
);
```

```
create table ponto_coleta(  
id_ponto int auto_increment not null primary key,  
fk_cnpj_empresa int not null,  
logradouro varchar(100) not null,  
numero varchar(5) not null,  
bairro varchar(50) not null,  
cidade varchar(50) not null,  
Estado varchar(50) not null,  
cep char(8) not null  
);
```

```
create table descarte(  
id_descarte int auto_increment not null primary key,  
fk_cpf_usuario int not null,  
fk_id_ponto int not null,  
`data` datetime not null  
);
```

```
create table usuario(  
id_usuario int auto_increment not null unique,  
cpf char(14) primary key not null,  
nome_usu varchar(100) not null,  
data_nascimento date not null,  
email varchar(256) not null unique,  
telefone1 varchar(13) not null unique,  
telefone2 varchar(13),  
logradouro varchar(50) not null,  
numero varchar(5),  
bairro varchar(50),  
cidade varchar(50),  
estado varchar(50),  
cep char(8) not null,  
pontuacao int not null);
```

```
insert into usuario values(1,8787878787878787, "Akyssa", 20020418, "akyssa@gmail.com",  
819827389, 239283746, "Rua senac",  
22, "Senac", "Recife", "Pernambuco", 38297188,0 );
```

```
alter table usuario add column senha varchar(100) not null;  
select*from usuario;
```

```
alter table gerente add foreign key (fk_cnpj_empresa) references Empresa(id_empresa);  
alter table contrato add foreign key (fk_cnpj_empresa) references Empresa(id_empresa);  
alter table ponto_coleta add foreign key (fk_cnpj_empresa) references  
Empresa(id_empresa);  
alter table coleta add foreign key (fk_cnpj_empresa) references Empresa(id_empresa);  
alter table coleta add foreign key (fk_id_ponto) references ponto_coleta(id_ponto);  
alter table descarte add foreign key (fk_cpf_usuario) references usuario(id_usuario);
```

```
alter table descarte add foreign key (fk_id_ponto) references ponto_coleta(id_ponto);
```

```
create table blog(  
id_postagem int auto_increment primary key,  
fk_id_autor int not null,  
tipo tinyint not null,  
titulo varchar(100) not null unique,  
subtitulo varchar(100) not null,  
texto text not null,  
data_postagem date not null);
```

```
create table autor(  
id_autor int auto_increment primary key,  
nome varchar(100) not null,  
data_nascimento date not null,  
formacao varchar(30));
```

```
alter table blog add foreign key (fk_id_autor) references autor(id_autor);
```

Lei Geral de Proteção de Dados Pessoais (LGPD)

A LGPD é a lei de nº13.709/2018 que foi promulgada para proteger os direitos fundamentais de liberdade e de privacidade e a livre formação da personalidade de cada indivíduo. E fala sobre o tratamento de dados pessoais, dispostos em meio físico ou digital. Segundo ela os dados pessoais devem ter um tratamento de boa-fé e princípios como finalidade, adequação, necessidade, livre acesso, qualidade dos dados, transparência, segurança, prevenção, não discriminação, responsabilização e prestação de contas.

A Conecta Recycle irá seguir os princípios da LGPD para manter todos os dados de seus usuários em total segurança e para isto irá utilizar criptografia em seu banco de dados e no tráfego de dados. As pessoas que estão responsáveis pela segurança e administração dos dados estarão sempre utilizando se dos melhores métodos e práticas de segurança pois isto será um requisito da própria empresa.