

UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO.

BRUNO BINELLI RIBEIRO
BRUNO GABRIEL CARBONI

RELATÓRIO TRABALHO FINAL DISCIPLINA DE BANCO DE DADOS -
Prof. Alexandre L. Gonçalves.

Araranguá, 2024

SUMÁRIO

1. DESCRIÇÃO DO OBJETIVO GERAL	03
2. DESCRIÇÃO DETALHADA	03
3. MODELO CONCEITUAL	04
4. MODELO LÓGICO	04
5. DDL	04
6. CONSULTAS	07
7. APLICAÇÃO	13

1. Descrição do objetivo geral do sistema:

Com a finalidade de reduzir a desordem e potencializar os ganhos de uma grande construtora, criaremos um projeto cujo objetivo central é desenvolver um sistema eficiente que auxilie a empresa no gerenciamento de seus projetos de construção civil. O software deve permitir o controle completo, desde o cadastro dos projetos, com suas respectivas equipes de atuação, conjunto de materiais que serão utilizados com seus respectivos fornecedores até o registro do contrato feito com um cliente

2. Descrição detalhada:

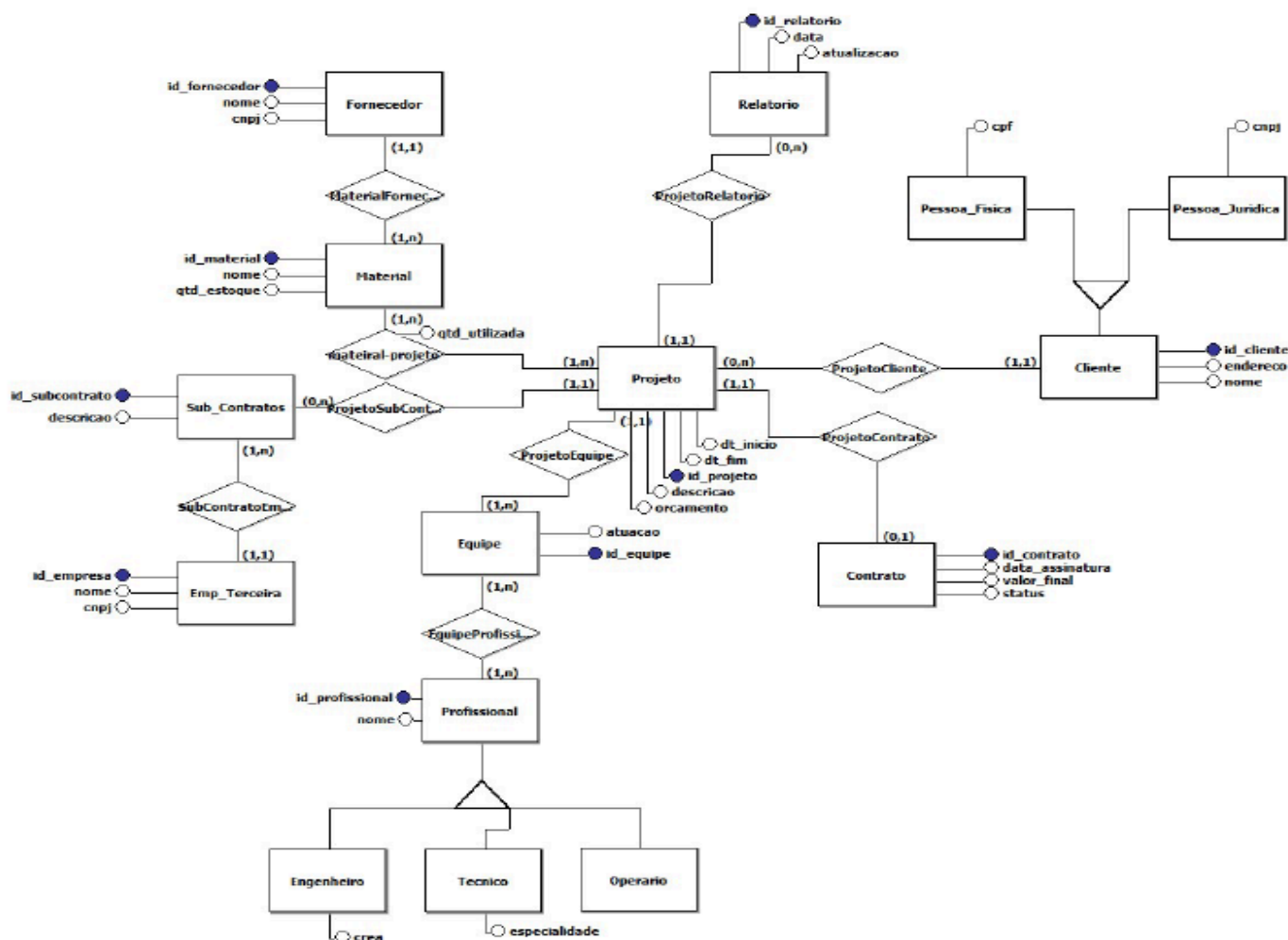
Com a finalidade de prover as informações necessárias para o sistema, os seguintes requisitos devem ser levados em consideração:

1. O sistema deve permitir o registro de projetos. Cada registro deve conter um identificador, uma descrição e as previsões de datas (início e fim) para a realização do projeto;
2. O software deve permitir o cadastro da equipe profissional que vai trabalhar no projeto, sendo que a empresa conta com as seguintes 3 classes de profissionais: Engenheiro, Técnico, Operário. Cada equipe possui um identificador e uma atuação. Todos os profissionais dispõem de um identificador e um nome, no entanto o Engenheiro possui ainda o cadastro do seu CREA e o Técnico uma descrição da sua especialidade;
3. Na realização de um projeto são utilizados inúmeros materiais. Para que haja um controle, o sistema deve permitir os cadastros dos materiais e sua quantidade utilizada em cada projeto. Cada material terá um id e sua quantidade disponível em estoque. Um material é adquirido de um fornecedor, que deve ser cadastrado, mas cada fornecedor pode prover mais de um material. Fornecedores possuem id, nome e CNPJ.
4. Alguns dos serviços imprescindíveis para a realização de um projeto (ex. pintura) não são providos diretamente pela empresa, portanto são terceirizados. Logo, o software deve permitir o registro dos contratos (“subcontratos” levando em consideração que são uma categoria abaixo do contrato final de obra) realizados com as empresas contratadas, que devem ser cadastradas, para a realização desses serviços. Esses subcontratos possuem um id e uma descrição. As empresas, id, nome e cnpj.
5. O sistema deve também permitir o cadastro de clientes. Clientes podem ser pessoas físicas ou jurídicas. Em seu registro deve possuir um id, seu nome e cpf/cnpj.
6. Cada cliente ao entrar em contato com a empresa estará interessado em realizar um projeto. Portanto, em resultado a esse contato, um orçamento será gerado. Essa informação de orçamento deve ser armazenada junto com as informações de projeto já descritas anteriormente.
7. Se o cliente resolver fechar o projeto, um contrato deve ser gerado a partir do que foi orçado entre o cliente e a empresa. Esse contrato terá um identificador, um status (se foi ou não assinado), data de assinatura e o valor final, que pode ou não ser igual ao valor do orçamento.

8. Os projetos podem possuir ainda relatórios que registram o andamento do projeto. Por ele, tanto o cliente quanto a equipe terão informações detalhadas do momento da obra. Logo, o sistema deve permitir o cadastro desses relatórios que contêm um identificador, uma data e uma atualização (descrição do andamento do projeto).

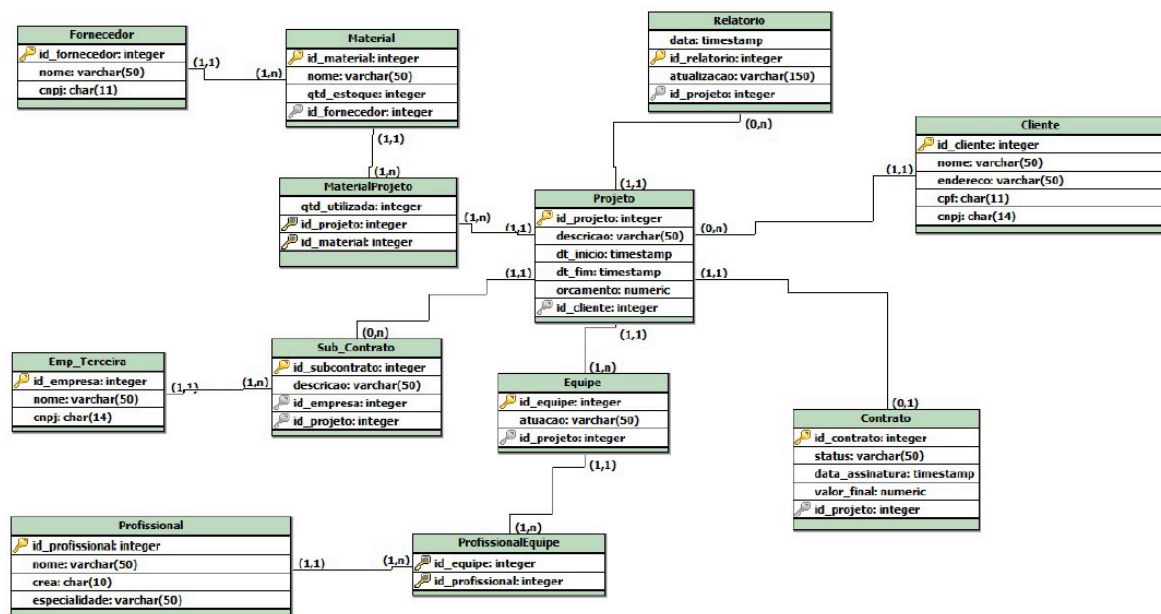
3. Modelo conceitual:

Disponível também no zip do projeto em arquivo no path [./modelos./tfinal_conceitual.brM.](#)



4. Modelo lógico:

Disponível também no zip do projeto em arquivo no path [./modelos./tfinal_logico.brM.](#)



5. DDL:

```

create table pfinal.cliente (
  id_cliente integer primary key,
  nome varchar(50) not null,
  endereco varchar(50),
  cpf char(11) unique,
  cnpj char(14) unique,
  constraint chk_cpf_cnpj check(
    (cpf is not null and cnpj is null) or
    (cpf is null and cnpj is not null)
  )
);

create table pfinal.projeto (
  id_projeto integer primary key,
  descricao varchar(50),
  dt_inicio timestamp,
  dt_fim timestamp,
  orcamento numeric,
  id_cliente integer,
  foreign key (id_cliente) references pfinal.cliente (id_cliente),
  constraint chk_dt_valida check (dt_inicio is null or dt_fim is null or dt_inicio < dt_fim)
);

create table pfinal.relatorio (
  data timestamp not null,
  id_relatorio integer primary key,
  atualizacao varchar(150),
  id_projeto integer,
  foreign key (id_projeto) references pfinal.projeto (id_projeto)
);
  
```

```

create table pfinal.contrato (
id_contrato integer primary key,
status varchar(50) not null constraint chk_status_contrato check (status in ('Assinado', 'Não Assinado')),
dataassinatura timestamp,
valor_final numeric default 0 constraint chk_valor_positivo check ((valor_final >= 0)),
id_projeto integer,
foreign key (id_projeto) references pfinal.projeto (id_projeto)
);
create table pfinal.equipe (
id_equipe integer primary key,
atuacao varchar(50),
id_projeto integer,
foreign key (id_projeto) references pfinal.projeto (id_projeto)
);
create table pfinal.profissional (
id_profissional integer primary key,
nome varchar(50) not null,
crea char(10),
especialidade varchar(50),
foreign key (id_equipe) references pfinal.equipe (id_equipe)
);
create table pfinal.profissional_equipe (
id_equipe integer,
id_profissional integer,
primary key (id_equipe, id_profissional),
foreign key (id_equipe) references pfinal.equipe (id_equipe),
foreign key (id_profissional) references pfinal.profissional (id_profissional)
);
create table pfinal.fornecedor (
id_fornecedor integer primary key,
nome varchar(50) not null,
cnpj char(14) not null
);
create table pfinal.material (
id_material integer primary key,
nome varchar(50) not null,
qtd_estoque integer not null constraint chk_estoque_positivo check (qtd_estoque >= 0),
id_fornecedor integer,
foreign key (id_fornecedor) references pfinal.fornecedor (id_fornecedor)
);
create table pfinal.material_projeto (
qtd_utilizada integer not null constraint chk_quantidade_positiva check(qtd_utilizada >= 0),
id_projeto integer,

```

```

id_material integer,
primary key(id_projeto,id_material),
foreign key (id_projeto) references pfinal.projeto (id_projeto),
foreign key (id_material) references pfinal.material (id_material)
);
create table pfinal.emp_terceira (
id_empresa integer primary key,
nome varchar(50) not null,
cnpj char(14) not null
);
create table pfinal.sub_contrato (
id_subcontrato integer primary key,
descricao varchar(50),
id_empresa integer,
id_projeto integer,
foreign key (id_projeto) references pfinal.projeto (id_projeto),
foreign key (id_empresa) references pfinal.emp_terceira (id_empresa)
);

```

6. Consultas:

Foram realizadas no total cinco consultas, sendo as três primeiras as principais e as duas últimas consultas que eram principais, mas decidimos, posteriormente, substituí-las. Todavia, achamos pertinente incluí-las no relatório, visto que já havíamos criado elas e apresentaram um resultado satisfatório.

6.1 - Consulta 01:

Descrição: Obter quantidade utilizada de cada um dos materiais para projetos iniciados entre 2020 e 2024.

Script:

```

select
    m1.id_material,
    m1.nome,
    sum(qtd_m2.qtd_utilizada) as quantidade_utilizada
from
    pfinal.material m1
full join
    (select
        m2.id_material,
        m2.nome,
        mp.qtd_utilizada
    from
        pfinal.material m2
    join
        pfinal.material_projeto mp on mp.id_material = m2.id_material

```

```

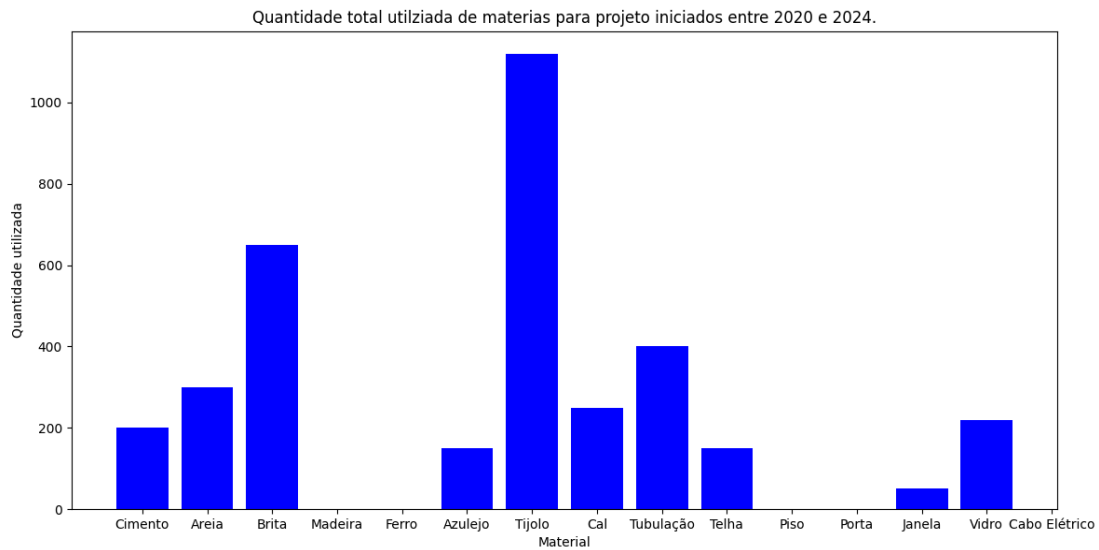
join
    pfinal.projeto p on p.id_projeto = mp.id_projeto
where
    p.dt_inicio between '2020-01-01' and '2024-12-01') qtd_m2 on
qtd_m2.id_material = m1.id_material
group by
    m1.id_material
order by
    m1.id_material

```

Resultado da consulta:

	id_material [PK] integer	nome character varying (50)	quantidade_utilizada bigint
1	1	Cimento	200
2	2	Areia	300
3	3	Brita	650
4	4	Madeira	[null]
5	5	Ferro	[null]
6	6	Azulejo	150
7	7	Tijolo	1120
8	8	Cal	250
9	9	Tubulação	400
10	10	Telha	150
11	11	Piso	[null]
12	12	Porta	[null]
13	13	Janela	50
14	14	Vidro	220
15	15	Cabo Elétrico	[null]

Gráfico gerado a partir do resultado da consulta:



6.2 - Consulta 02:

Descrição: Obter diferença do valor orçado e do valor final para projetos assinados entre 2019 e meio de 2021.

Script:

```
select
    prj.id_projeto,
    prj.descricao,
    sum(prj.orcamento - cnt.valor_final) as saldo_restante
from
    pfinal.projeto prj
left join
    pfinal.sub_contrato sent on sent.id_projeto = prj.id_projeto
join
    pfinal.contrato cnt on cnt.id_projeto = prj.id_projeto
where
    cnt.dataassinatura between '2019-01-01' and '2021-06-01'
group by
    prj.id_projeto,
    prj.descricao
```

Resultado da consulta:

	id_projeto [PK] integer	descricao character varying (50)	saldo_restante numeric
1	5	Reforma Residencial	10000
2	6	Construção de Galpão Industrial	30000
3	7	Construção de Piscina Pública	10000
4	8	Ampliação de Escola	20000
5	9	Pavimentação de Rua	10000
6	10	Construção de Parque Urbano	10000
7	11	Reforma de Prédio Comercial	10000
8	12	Construção de Praça Pública	10000

Gráfico gerado a partir do resultado da consulta:



6.3 - Consulta 03:

Descrição: Encontrar média dos orçamentos anuais realizados pela empresa.

Script:

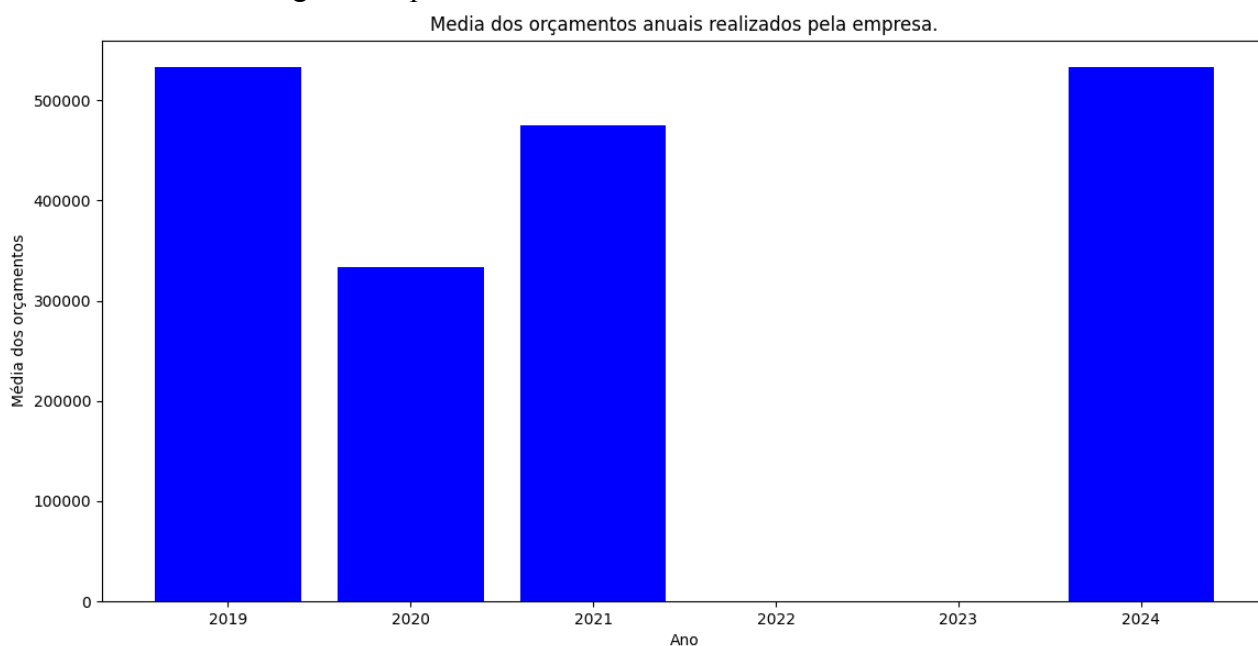
```
select
    extract(year from prj.dt_inicio) as ano,
    avg(prj.orcamento) as orcamento_medio
from
    pfinal.projeto prj
join
    pfinal.cliente cli on cli.id_cliente = prj.id_cliente
join
    pfinal.contrato cnt on cnt.id_projeto = prj.id_projeto
where
```

```
prj.dt_inicio between '2019-01-1' and '2025-01-1'
group by
ano
```

Resultado da consulta:

	ano numeric	orcamento_medio numeric
1	2019	533333.33333333333333
2	2020	333333.33333333333333
3	2021	475000.00000000000000
4	2024	533333.33333333333333

Gráfico gerado a partir do resultado da consulta:



6.4 - Consulta 04:

Descrição: Exibir quantidade de diferentes materiais utilizados por projetos com contrato assinado entre 2019 e 2023.

Script:

```
select
    prj.id_projeto as codigo,
    prj.descricao as projeto,
    count(mprj.id_projeto) as num_material
from
    pfinal.projeto prj
join
    pfinal.material_projeto mprj on mprj.id_projeto = prj.id_projeto
join
    pfinal.contrato cnt on cnt.id_projeto = prj.id_projeto
where
```

```

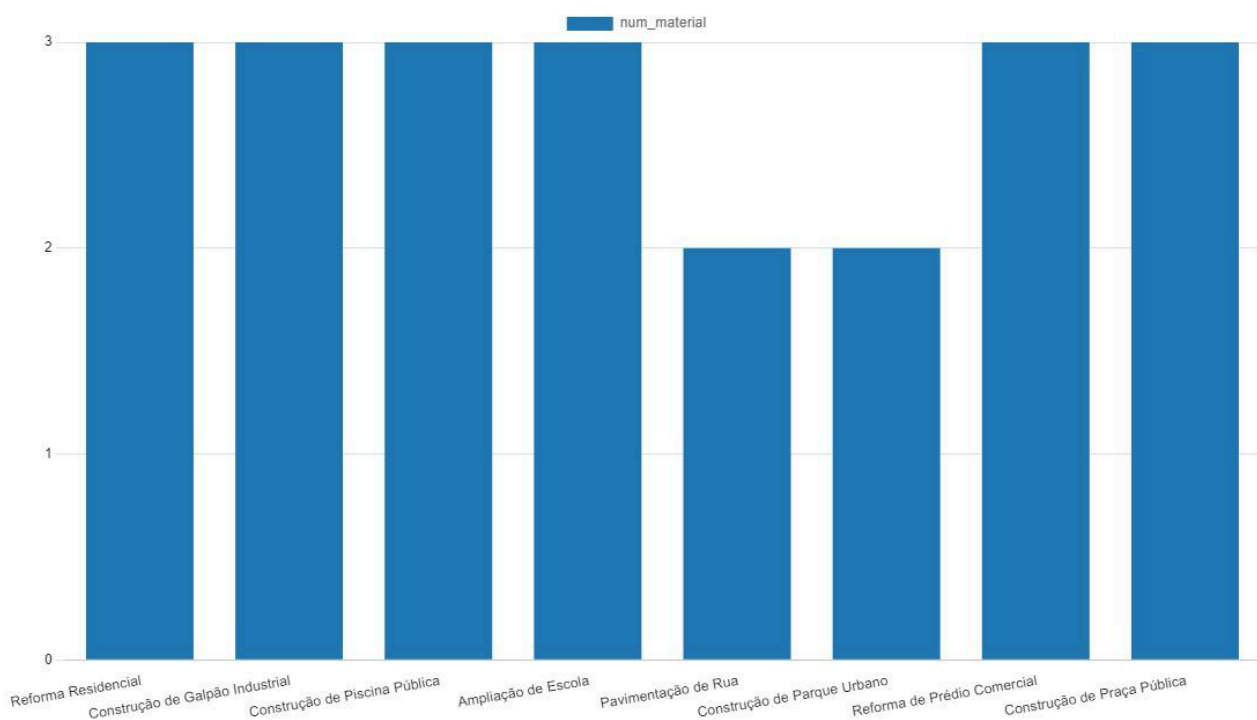
cnt.data_assinatura between '2019-01-01' and '2022-12-31'
group by
    prj.id_projeto,
    prj.descricao

```

Resultado da consulta:

	codigo integer	projeto character varying (50)	num_material bigint
1	5	Reforma Residencial	3
2	6	Construção de Galpão Industrial	3
3	7	Construção de Piscina Pública	3
4	8	Ampliação de Escola	3
5	9	Pavimentação de Rua	2
6	10	Construção de Parque Urbano	2
7	11	Reforma de Prédio Comercial	3
8	12	Construção de Praça Pública	3

Gráfico gerado a partir do resultado da consulta:



6.5 - Consulta 05:

Descrição: Obter a soma total dos orçamentos anuais dos projetos de projetos em andamento entre 2019 e 2024.

Script:

```

select
    extract(year from prj.dt_inicio) as ano,
    sum(prj.orcamento) as orcamento_total

```

```

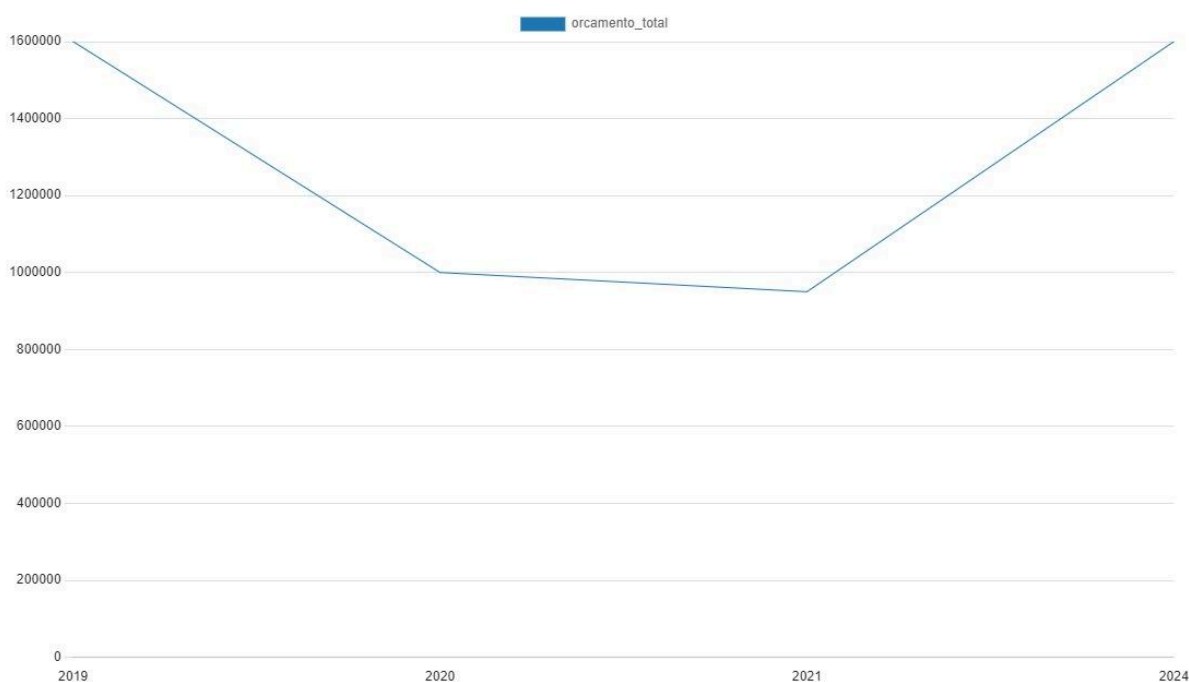
from
  pfinal.projeto prj
join
  pfinal.cliente cli on cli.id_cliente = prj.id_cliente
join
  pfinal.contrato cnt on cnt.id_projeto = prj.id_projeto
where
  prj.dt_inicio between '2019-01-1' and '2025-01-1'
group by
  ano

```

Resultado da consulta:

	ano numeric 🔒	orcamento_total numeric 🔒
1	2019	1600000
2	2020	1000000
3	2021	950000
4	2024	1600000

Gráfico gerado a partir do resultado da consulta:



7. Aplicação:

Disponível na pasta application no zip do projeto.