

## Técnico Superior Profissional em Tecnologias Informáticas

### Fórum Coimbra – Base de Dados



Nome	Nº de Aluno	Email
Leonardo Carvalho	2023118977	a2023118977@alunos.estgoh.ipc.pt
Luis Hernández	2023135575	a2023135575@alunos.estgoh.ipc.pt
Luis Vieira	2023124946	a2023124946@alunos.estgoh.ipc.pt
Tiago Nogueira	2023123255	a2023123255@alunos.estgoh.ipc.pt

Coimbra

2022 / 2023

Escola Superior de Tecnologia e Gestão de Oliveira do Hospital

Curso Técnico Superior Profissional em Tecnologias Informáticas (Coimbra)

Base de dados

Fórum de Coimbra

Leonardo Carvalho | Luís Hernandez | Luís Vieira | Tiago Nogueira

Docente da aula  
Olga Coutinho

Coimbra

Ano Letivo 2023/2024

## Índice

Índice de tabelas.....	4
Índice de Figuras .....	5
Introdução .....	6
Objetivo .....	8
Situação atual .....	9
Problemas Encontrados/Possíveis Melhorias .....	10
Descrição da Solução Proposta .....	11
Análise de Dados .....	13
Entidades e atributos .....	13
Relacionamentos .....	18
Diagrama Entidade Relacionamento .....	20
Diagrama do Modelo Físico .....	21
Script de criação da Base de Dados .....	22
Anexo: Lista de Termos Técnicos .....	31
Conclusão .....	32
Webgrafia.....	33

## Índice de tabelas

Tabela 1 - Esforço .....	7
--------------------------	---

## Índice de Figuras

Figura 1 – Diagrama ER.....	20
Figura 2- Modelo Físico.....	21

## Introdução

Neste trabalho vamos abordar a criação da base de dados do fórum Coimbra, decidimos escolher este tema pois, para além de ser algo presente e que nos conhecemos, achamos que podia ser um projeto interessante e que consegue desenvolver o nosso conhecimento a nível de *teamwork*, brainstorming, e base de dados na criação de uma para o mesmo.

Para além disso escolher uma base de dados como esta seria um desafio muito maior que apenas uma base de dados convencional, daí este tópico fazer muito sentido para nós

## Distribuição do esforço:

Tarefa	Leonardo Carvalho	Luis Hernández	Luís Vieira	Tiago Nogueira
Objetivo	100%	0%	0%	0%
Descrição Solução	0%	100%	0%	0%
Problemas/Melhorias	0%	0%	100%	0%
Situação Atual	0%	0%	0%	100%
Entidades	15%	30%	40%	15%
Relacionamentos	20%	40%	20%	20%
Modelo Físico	25%	10%	15%	50%
Conclusão	40%	20%	25%	15%

Tabela 1 - Esforço

## Objetivo

O nosso objetivo com o trabalho de base de dados, na especificação inicial tem em conta o Shopping Fórum Coimbra.

Decidimos trabalhar sobre este contexto dado ao facto de ser um centro comercial muito movimento, e com recurso a melhoramento da sua base de dados atual, para que a sua gestão fique muito mais facilitada, e que se possa fazer todo o gerenciamento a partir desta.

Este vai poder gerenciar loja, funcionários, categoria, despesas, eventos, pisos...

Vamos ter que ter alguma prudência pois algum dos dados são confidenciais e daí o uso da base de dados ser tão importante, mas também corretamente utilizado com segurança.

A Nossa base de dados vai ser implementada utilizando um Sistema de Gestão de Base de Dados.



## Situação atual

Devido a esta época festiva e à sua ser antiga e lenta, o fórum decidiu contactar-nos pois desejavam ter uma base de dados mais rápida e atualizada pois várias das funcionalidades que a sua base de dados faz é muito lento para além de quererem colocar novas funcionalidades que tinham que ser feitas em papel de custava muito dinheiro em papel e custar muito tempo. Após entrarem em contacto connosco fizemos uma análise, começamos a pensar em como implementar o que nos foi solicitado e, depois de algum tempo, apresentamos uma solução para garantir que estivesse de acordo com as expectativas.

## Problemas Encontrados/Possíveis Melhorias

Após uma análise cuidada da situação atual do Shopping Fórum Coimbra, foi possível identificar alguns aspetos que podem ser melhorados:

- A gestão de dados do shopping é ineficiente e insegura, pois não existe uma base de dados centralizada e integrada que armazene e organize todos os dados relevantes ao shopping.
- Alguns dados são registados e guardados em formato de papel, o que aumenta o risco de perda, erro, falsificação e duplicação dos mesmos, para além do tempo consumido e custos adicionais nesta gestão.
- A base de dados atual foi desenvolvida há vários anos e, conseqüentemente, foram utilizadas tecnologias que não permitem uma melhoria fácil ou a adição de funcionalidades.

## Descrição da Solução Proposta

Para melhorar a gestão de dados do Shopping Fórum Coimbra, é sugerido construir uma nova base de dados relacional usando o MySQL como sistema de gestão de base de dados e o SQL como linguagem de manipulação de dados. A nova base de dados deve ser capaz de armazenar, organizar, consultar e atualizar toda a informação relacionada ao shopping, como:

Dados das lojas, como nome, categoria, piso, área, número de funcionários, facturamento, despesas, etc.

Dados dos funcionários, como nome, cargo, salário, horário, loja em que trabalham, etc.

Dados dos eventos, como nome, data, hora, local, descrição, etc.

Dados dos pisos, como número, área, número de lojas, número de elevadores, etc.

A integração da nova base de dados garante que os dados não sejam perdidos, assim como elimina redundâncias ou inconsistências. A nova base de dados deve ser segura para proteger informações confidenciais de acesso não autorizado e alterações indevidas.

Devem ser desenvolvidas aplicações que permitam aos usuários trabalhar com a nova base de dados e realizar as seguintes tarefas:

inserir, alterar e remover informações sobre lojas, funcionários, eventos e pisos.

Examinar e visualizar os dados de lojas, funcionários, eventos e pisos de acordo com os critérios definidos pelos usuários.

Gerar relatórios e gráficos sobre o desempenho do shopping, incluindo facturamento, despesas, ocupação e satisfação do cliente, entre outras informações.

As aplicações devem ter uma arquitetura cliente/servidor, em que o servidor gere a conexão com a base de dados e execute as operações solicitadas pelo cliente, enquanto os clientes interagem com os usuários e apresentam os resultados das

operações. intervalo de tempo compreendido entre duas datas, escolhidas pelo utilizador.

## Análise de Dados

O objetivo deste capítulo é apresentar o diagrama de Entidade-Relacionamento da base de dados, e fornecer uma descrição abrangente de todas as entidades e relações, que fazem parte da solução proposta.

### Entidades e atributos

Piso:

- idpiso (chave primária, int(11), não nulo)

Loja:

- idloja (chave primária, int(11), não nulo, incremento automático)
- idcontrato (chave estrangeira referenciando Contratos, int(11))
- idcategoria (chave estrangeira referenciando Categoria, int(11), não nulo)
- idpiso (chave estrangeira referenciando Piso, int(11), não nulo)
- nome (varchar(100), não nulo)
- donoloja (varchar(100), não nulo)

Funcionários:

- idfuncionario (chave primária, int(11), não nulo, incremento automático)
- idpiso (chave estrangeira referenciando Piso, int(11), não nulo)
- idcontrato (int(11))

Contratos:

- idcontrato (chave primária, int(11), não nulo, incremento automático)
- idcategoriapag (chave estrangeira referenciado CategoriaPagamentos, int(11), não nulo)
- idadmin (chave estrangeira referenciando Administradores, int(11), não nulo)
- datainicio (date, não nulo)
- datatermino (date, não nulo)
- valor (int(11), não nulo)

Evento:

- idevento (chave primária, int(11), não nulo, incremento automático)
- idloja (chave estrangeira referenciando Loja, int(11), não nulo)
- idpiso (chave estrangeira referenciando Piso, int(11), não nulo)
- nome (varchar(100), não nulo)
- dia (date, não nulo)
- hora (time, não nulo)

Categoria:

- idcategoria (chave primária, int(11), não nulo, incremento automático)
- nome (varchar(30), não nulo)

Técnicos:

- idtecnico (chave primária, int(11), não nulo)
- nome (varchar(100), não nulo)
- data\_nascimento (date, não nulo)
- telemovel (int(9), não nulo)
- NIF (int(9), não nulo)

Administradores:

- idadmin (chave estrangeira referenciando Funcionários, int(11), não nulo)
- nome (varchar(30), não nulo)
- data\_nascimento (date, não nulo)
- telemovel (int(9), não nulo)
- NIF (int(9), não nulo)

Atividades:

- idatividade (chave primária, int(11), não nulo, incremento automático)
- nome (varchar(100), não nulo)

Pagamentos:

- idpagamentos (chave primária, int(11), não nulo, incremento automático)

- idcontrato (chave estrangeira referenciando Contratos, int(11), não nulo)
- idfuncionario (chave estrangeira referenciando Funcionarios, int(11))
- idloja (chave estrangeira referenciando Loja, int(11))
- dia (date, não nulo)

#### Anúncios:

- idanuncio (chave primária, int(11), não nulo, incremento automático)
- idpiso (chave estrangeira referenciando Piso, int(11), não nulo)
- idloja (chave estrangeira referenciando Loja, int(11), não nulo)
- nome (varchar(100), não nulo)
- conteudo (varchar(100), não nulo)
- custo (int(11), não nulo)
- datapublicacao (date, não nulo)

#### PisoAtividade:

- idpisoatividade (chave primária, int(11), não nulo, incremento automático)
- idpiso (chave estrangeira referenciando Piso, int(10), não nulo)
- idatividade (chave estrangeira referenciando Atividades, int(10), não nulo)

#### TecnicoAtividade:

- idtecnicoatividade (chave primária, int(11), não nulo, incremento automático)
- idtecnico (chave estrangeira referenciando Técnicos, int(10), não nulo)



- idatividade (chave estrangeira referenciando Atividades, int(10), não nulo)
- dia (date, não nulo)
- horainicio (time, não nulo)
- horafim (time, não nulo)

#### CategoriaPagamentos

- idcategoriapag (chave primária, int(11), não nulo, incremento automático)
- nome (varchar(30), não nulo)

#### Despesas

- iddespesas (chave primária, int(11), não nulo, incremento automático)
- idcategoriapag (chave estrangeira referenciado CategoriaPagamentos, int(11), não nulo)
- valor(int(11), não nulo)
- descricao (varchar(255), não nulo)
- dia (date, não nulo)

## Relacionamentos

- **Loja e Contratos** (Muitos para 1)
  - Cada loja só tem um contrato, mas um contrato pode ser usado para várias lojas.
- **Funcionários e Contratos** (Muitos para 1)
  - Um funcionário só tem um contrato, mas um contrato é usado para vários funcionários.
- **Contratos e Administradores** (Muitos para 1)
  - Cada contrato é gerido por um único administrador, mas um administrador pode gerir vários contratos.
- **Contratos e CategoriaPagamentos** (Muitos para 1)
  - Cada contrato está associado a uma única categoria de pagamento, mas vários contratos podem partilhar a mesma categoria.
- **Contratos e Pagamentos** (1 para Muitos)
  - Cada contrato pode ter vários pagamentos associados, mas cada pagamento pertence a um único contrato.
- **Pisos e Lojas** (Muitos para 1)
  - Cada loja está associada a um único piso, mas um piso pode ter várias lojas.
- **Categorias e Lojas** (Muitos para 1)
  - Cada loja pertence a uma única categoria, mas várias lojas podem partilhar a mesma categoria.
- **Eventos e Lojas** (Muitos para 1)
  - Cada loja pode ter vários eventos associados, mas cada evento pertence a uma única loja.
- **Lojas e Anúncios** (1 para Muitos)
  - Cada loja pode ter vários anúncios associados, mas cada anúncio pertence a uma única loja.
- **Lojas e Pagamentos** (1 para Muitos):
  - Cada loja pode ter vários pagamentos associados, mas cada pagamento pertence a uma única loja.
- **Eventos e Pisos** (Muitos para 1):
  - Cada evento está associado a um único piso, mas um piso pode ter vários eventos.
- **Anúncios e Pisos** (Muitos para 1):
  - Cada anúncio está associado a um único piso, mas um piso pode ter vários anúncios.
- **Funcionários e Pisos** (Muitos para 1):
  - Cada funcionário está associado a um único piso, mas um piso pode ter vários funcionários.
- **Pagamentos e Funcionários** (Muitos para 1):
  - Cada pagamento é feito a um único funcionário, mas um funcionário pode receber vários pagamentos
- **Despesas e CategoriaPagamentos** (1 para Muitos)
  - Cada despesa está associada a uma única categoria de pagamento (Eletricidade, água, etc), mas várias despesas vão partilhar a mesma categoria.

- **Atividades e Técnicos** (Muitos para Muitos)
  - Cada atividade pode ter vários técnicos associados, e cada técnico pode participar em várias atividades. (Para esta relação foi criada no modelo físico a tabela intermediária: TecnicoAtividade)
- **Funcionários** divididos em **Técnicos** e **Administradores**
  - A tabela funcionários é a entidade principal que contém informações compartilhadas para todos os funcionários.  
Dado que existem diferentes tipos de funcionários, nomeadamente técnicos e administradores, adotamos uma abordagem de herança.  
Dessa forma, cada registo na tabela funcionários pode ser associado a zero ou um registo nas tabelas especializadas, permitindo a inclusão de detalhes específicos para cada tipo de funcionário.

## Diagrama Entidade Relacionamento

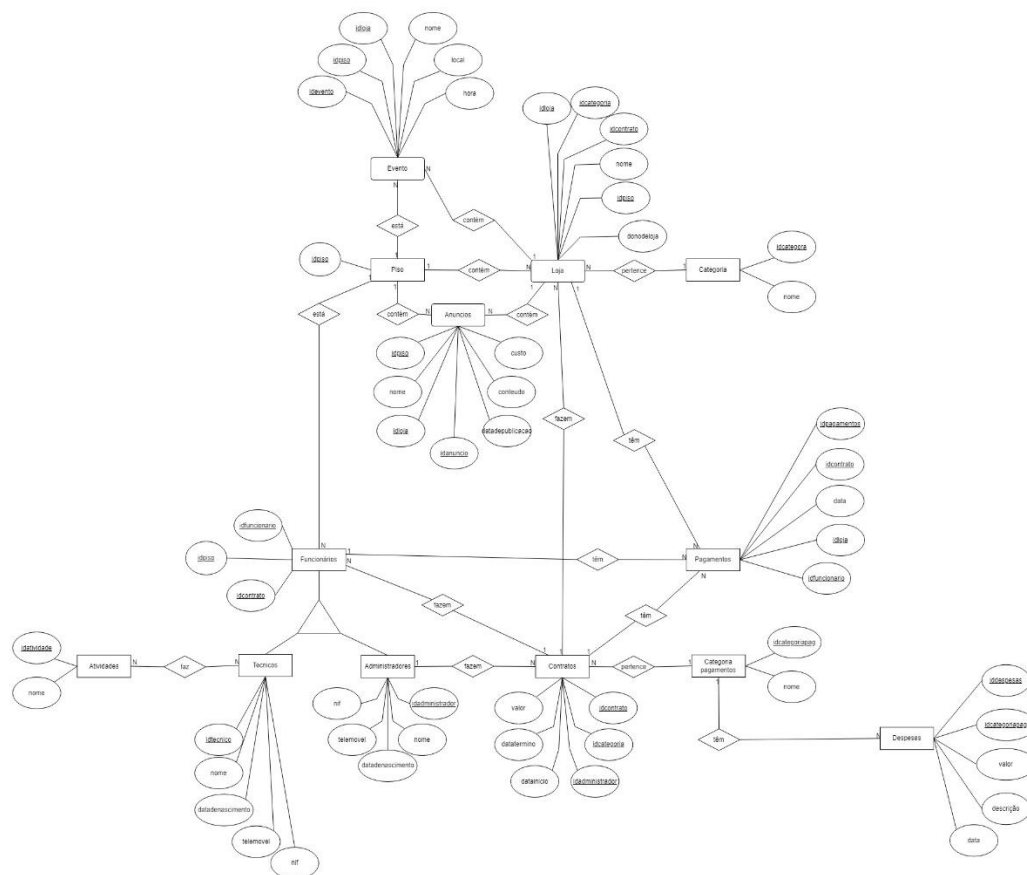


Figura 1 – Diagrama ER

## Diagrama do Modelo Físico

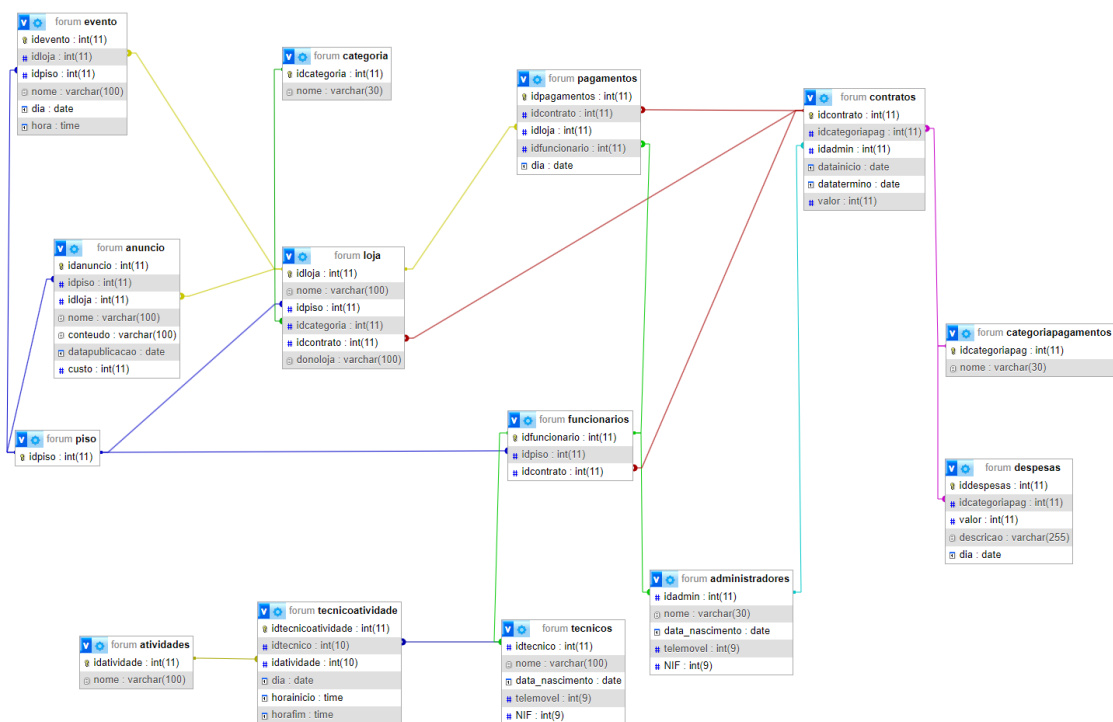


Figura 2- Modelo Físico

## Script de criação da Base de Dados

De seguida apresenta-se o código necessário para a criação da Base de Dados do Shopping Fórum Coimbra.

BDForum Coimbra.

```
CREATE DATABASE forum;
```

```
-----  
CREATE TABLE categoria(  
    idcategoria int PRIMARY KEY AUTO_INCREMENT,  
    nome varchar(30) NOT NULL  
);
```

```
CREATE TABLE piso(  
    idpiso int PRIMARY KEY  
);
```

```
CREATE TABLE funcionarios(  
    idfuncionario int PRIMARY KEY AUTO_INCREMENT  
);
```

```
CREATE TABLE `tecnicos` (  
    `idtecnico` int(11) NOT NULL,  
    `nome` varchar(100) NOT NULL,  
    `data_nascimento` date NOT NULL,  
    `telemovel` int(9) NOT NULL,  
    `NIF` int(9) NOT NULL,  
    FOREIGN KEY (idtecnico) REFERENCES funcionarios(idfuncionario)  
);
```

```
CREATE TABLE `administradores` (  
  `idadmin` int(11) NOT NULL,  
  `nome` varchar(30) NOT NULL,  
  `data_nascimento` date NOT NULL,  
  `telemovel` int(9) NOT NULL,  
  `NIF` int(9) NOT NULL,  
  FOREIGN KEY (idadmin) REFERENCES funcionarios(idfuncionario)  
);
```

```
CREATE TABLE categoriapagamentos(  
  
  idcategoriapag int PRIMARY KEY AUTO_INCREMENT,  
  nome varchar(30)  
)
```

```
CREATE TABLE despesas(  
  iddespesas int PRIMARY KEY AUTO_INCREMENT,  
  idcategoriapag int NOT NULL,  
  valor int NOT NULL,  
  descricao varchar(255) NOT NULL,  
  FOREIGN KEY (idcategoriapag) REFERENCES  
categoriapagamentos(idcategoriapag)  
)
```

```
CREATE TABLE contratos(  
  idcontrato int PRIMARY KEY AUTO_INCREMENT,  
  idcategoriapag int NOT NULL,  
  idadmin int NOT NULL,  
  datainicio date NOT NULL,  
  datatermino date NOT NULL,  
  valor int NOT NULL,
```

FOREIGN KEY (idcategoriapag) REFERENCES  
categoriapagamentos(idcategoriapag),

FOREIGN KEY (idadmin) REFERENCES administradores(idadmin)  
)

CREATE TABLE atividades(  
    idatividade int PRIMARY KEY AUTO\_INCREMENT,  
    nome varchar(100) NOT NULL  
)

CREATE TABLE tecnicoatividade (  
    idtecnicoatividade int(11) NOT NULL PRIMARY KEY AUTO\_INCREMENT,  
    idtecnico int(10) NOT NULL,  
    idatividade int(10) NOT NULL,  
    dia date NOT null,  
    horainicio time NOT null,  
    horafim time not null,  
    FOREIGN KEY (idtecnico) REFERENCES tecnicos(idtecnico),  
    FOREIGN KEY (idatividade) REFERENCES atividades(idatividade)  
)

ALTER TABLE funcionarios ADD COLUMN idpisso int NOR NULL

ALTER TABLE funcionarios ADD COLUMN idcontrato int

ALTER TABLE funcionarios ADD FOREIGN KEY (idpisso) REFERENCES piso(idpisso)

ALTER TABLE funcionarios ADD FOREIGN KEY (idcontrato) REFERENCES  
contratos(idcontrato)

Alter TABLE despesas ADD COLUMN dia date NOT NULL



```
CREATE TABLE loja(  
    idloja int PRIMARY KEY AUTO_INCREMENT,  
    nome varchar(100) NOT NULL,  
    idpiso int NOT NULL,  
    idcategoria int NOT NULL,  
    idcontrato int,  
    donoloja varchar(100) NOT NULL,  
    FOREIGN KEY (idpiso) REFERENCES piso(idpiso),  
    FOREIGN KEY (idcategoria) REFERENCES categoria(idcategoria),  
    FOREIGN KEY (idcontrato) REFERENCES contratos(idcontrato)  
)
```

```
CREATE TABLE evento(  
    idevento int PRIMARY key AUTO_INCREMENT,  
    idloja int NOT NULL,  
    idpiso int NOT NULL,  
    nome varchar(100) NOT NULL,  
    dia date NOT NULL,  
    hora time NOT NULL,  
    FOREIGN KEY (idpiso) REFERENCES piso(idpiso),  
    FOREIGN KEY (idloja) REFERENCES loja(idloja)  
)
```

```
CREATE TABLE anuncio(  
    idanuncio int PRIMARY KEY AUTO_INCREMENT,  
    idpiso int NOT NULL,  
    idloja int NOT NULL,  
    nome varchar(100) NOT NULL,  
    conteudo varchar(100) NOT NULL,  
    datapublicacao date NOT NULL,  
    custo int NOT NULL,
```

```
FOREIGN KEY (idpiso) REFERENCES piso(idpiso),  
FOREIGN KEY (idloja) REFERENCES loja(idloja)  
)
```

```
CREATE TABLE pagamentos(  
    idpagamentos int PRIMARY KEY AUTO_INCREMENT,  
    idcontrato int NOT NULL,  
    idloja int,  
    idfuncionario int,  
    dia date NOT NULL,  
    FOREIGN KEY (idcontrato) REFERENCES contratos(idcontrato),  
    FOREIGN KEY (idloja) REFERENCES loja(idloja),  
    FOREIGN KEY (idfuncionario) REFERENCES funcionarios(idfuncionario)  
)
```

-----

```
INSERT INTO `categoria`(`nome`) VALUES
```

```
('Acessorios de Moda e Bijuteria'),  
( 'Agencias de Viagens'),  
( 'Artigos de Desporto'),  
( 'Bancos'),  
( 'Brinquedos'),  
( 'Cabeleireiro e Estetica'),  
( 'Cafes e Pastelarias'),  
( 'Cinema'),  
( 'Cosmetica'),  
( 'Decoracao'),  
( 'Doces e Chocolates'),  
( 'Electrodomésticos'),  
( 'Farmacia'),  
( 'Fotografia'),
```

('Gelatarias'),  
('Hipermercado'),  
('Infantil e Pre-Mama'),  
('Informatica'),  
('Joalharia'),  
('Lingerie e Roupa Interior'),  
('Livraria e Papelaria'),  
('Malas e Artigos de Viagem'),  
('Optica'),  
('Ourivesaria'),  
('Perfumaria'),  
('Prendas'),  
('Produtos Naturais'),  
('Relojoaria'),  
('Restauração'),  
('Sapataria'),  
('Serviços Diversos'),  
('Som e Video'),  
('Tabacaria'),  
('Telecomunicacoes'),  
('Vestuario')

-----  
  
INSERT INTO ` piso`(` idpiso`) VALUES (0),(1),(2)

-----  
  
INSERT INTO `funcionarios`(` idfuncionario`,` idpiso`,` idcontrato`) VALUES (NULL, '1', NULL);

```
INSERT INTO `funcionarios` (`idfuncionario`, `idpiso`, `idcontrato`) VALUES (NULL, '1', NULL);
```

-----

```
INSERT INTO `tecnicos` (`idtecnico`, `nome`, `data_nascimento`, `telemovel`, `NIF`) VALUES ('1', 'Ricardo Antunes', '2000-10-04', '911085984', '227516338');
```

```
INSERT INTO `administradores` (`idadmin`, `nome`, `data_nascimento`, `telemovel`, `NIF`) VALUES ('2', 'Jose Silva', '1995-06-08', '92384097', '123456789');
```

-----

```
INSERT INTO `loja` (`idloja`, `nome`, `idpiso`, `idcategoria`, `idcontrato`, `donoloja`) VALUES (NULL, 'Continente', '1', '16', NULL, 'João Lopes');
```

-----

```
INSERT INTO `categoriapagamentos` (`idcategoriapag`, `nome`) VALUES (NULL, 'Loja');
```

```
INSERT INTO `categoriapagamentos` (`idcategoriapag`, `nome`) VALUES (NULL, 'Funcionario');
```

-----

```
INSERT INTO `contratos` (`idcontrato`, `idcategoriapag`, `idadmin`, `datainicio`, `datatermino`, `valor`) VALUES (NULL, '1', '2', '2021-02-01', '2024-05-09', '5000');
```

```
INSERT INTO `contratos` (`idcontrato`, `idcategoriapag`, `idadmin`, `datainicio`, `datatermino`, `valor`) VALUES (NULL, '2', '2', '2023-12-08', '2024-08-07', '890');
```

-----

```
UPDATE `loja` SET `idcontrato` = '1' WHERE `loja`.`idloja` = 1;
```

-----

```
UPDATE `funcionarios` SET `idcontrato` = '2' WHERE `funcionarios`.`idfuncionario` = 1;
```

-----

```
INSERT INTO `atividades` (`idatividade`, `nome`) VALUES (NULL, 'Verificar as luzes');
```

-----

```
INSERT INTO `tecnicoatividade` (`idtecnicoatividade`, `idtecnico`, `idatividade`, `dia`, `horainicio`, `horafim`) VALUES (NULL, '1', '1', '2023-12-01', '14:42:30', '19:50:00');
```

-----

```
INSERT INTO `categoriapagamentos` (`idcategoriapag`, `nome`) VALUES (NULL, 'Luz');
```

-----

```
INSERT INTO `despesas` (`iddespesas`, `idcategoriapag`, `valor`, `descricao`, `dia`) VALUES (NULL, '3', '2500', 'Luz', '2023-12-30');
```

-----

```
INSERT INTO `anuncio` (`idpiso`, `idloja`, `nome`, `conteudo`, `datapublicacao`, `custo`) VALUES (2, 1, 'Para um bem maior', 'Venda de canetas para a caridade', '2022-12-17', 1000)
```

-----

```
INSERT INTO `evento` (`idloja`, `idpiso`, `nome`, `dia`, `hora`) VALUES (1, 2, 'Evento Natal', '2023-12-23', '16:00:00')
```

-----

```
INSERT INTO `pagamentos`(`idcontrato`, `idloja`, `idfuncionario`, `dia`) VALUES (2,  
NULL, 1, '2023-12-30')
```

## Anexo: Lista de Termos Técnicos

Base de dados – sítio onde se guarda informação sobre determinado assunto

Mysql – sistema de gestão de base de dados

Sql – linguagem de gestão de base de dados

Teamwork – trabalho em equipa

## Conclusão

Com este trabalho, conseguimos analisar estatísticas e chegar a um consenso que permitisse levar uma solução otimizada para o nosso cliente, o Fórum Coimbra.

A colaboração em equipa permitiu-nos concluir as tarefas de forma mais rápida e eficiente.

A utilização prática dos conceitos estudados em sala de aula, combinada com a investigação e assimilação de novos dados, levou a um progresso considerável no nosso aprendizado. Em conclusão, este projeto não só expandiu a nossa perceção sobre bases de dados, mas também sublinhou a importância do trabalho colaborativo.



## Webgrafia

[Base de dados - ícones de eletrônicos grátis \(flaticon.com\)](https://flaticon.com)

[Forum Coimbra \(forum-coimbra.com\)](https://forum-coimbra.com)