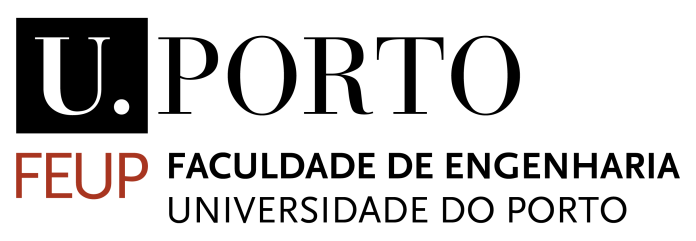
Faculdade de Engenharia da Universidade do Porto



Base de Dados

Hugo Gomes - [up202004343@edu.fe.up.pt](mailto:up202004343@edu.fe.up.pt)

João Moreira - [up202005035@edu.fe.up.pt](mailto:up202005035@edu.fe.up.pt)

Lia Vieira - [up202005042@edu.fe.up.pt](mailto:up202005042@edu.fe.up.pt)

Entrega 1

Relatório realizado no âmbito da

Licenciatura em Engenharia Informática e Computação

Professor: Michel Ferreira

Novembro 2021

Índice

1.Contexto 1

2.Esquema Relacional 3

3.Análise de Dependências Funcionais e Formas Normais 4

3.1.Decomposição 5

3.2.Teste à junção sem perdas 5

4. Restrições 6

## 1.Contexto

A entidade responsável pelo mercado imobiliário no distrito do Porto pretende criar um registo de todos os imóveis disponíveis para venda e arrendamento nesta zona. Para tal, desafiou os alunos da unidade curricular de Base de Dados do L.EIC para a criação de uma base de dados, tendo como base o site [Idealista](https://www.idealista.pt/). O Idealista consiste num site onde se apresenta toda a oferta imobiliária disponível no país, sendo possível filtrá-la através de diversas características.

De cada imóvel pretende-se armazenar o seu preço, estado (novo, seminovo, etc.), dimensão, referência e tipo de negócio (arrendar ou comprar). Estes imóveis estão tipificados em: apartamento, moradia, escritório, armazém, garagem e terreno. Em relação aos apartamentos é necessário saber qual o seu tipo, andar, tipologia (T0, T1, etc.) e número de casas de banho. O tipo, o número de pisos, a tipologia e o número de casas de banho, no que diz respeito às moradias. A distribuição, no que respeita aos escritórios. A vigilância no que toca às garagens e, por fim, o tipo (urbano, urbanizável e não urbanizável) dos terrenos. Em relação a todos os imóveis é necessário saber a sua localização, que consiste em: rua, concelho e número.

Cada imóvel pode ter mais que um proprietário, tal como um proprietário pode ser dono de vários imóveis. Sobre o proprietário é importante salientar que tanto pode ser um indivíduo, identificado através do seu número de identificação civil, ou uma instituição bancária, que é distinguida através da sua referência. Em ambos os casos, é necessário saber o seu nome, contacto telefónico e endereço de correio eletrónico.

Todos os imóveis são apresentados através de anúncios. Cada anúncio está associado a um só imóvel, mas, por outro lado, um imóvel pode estar ligado a vários anúncios. Os anúncios, os quais é necessário saber a data da sua criação, podem ser publicados de duas formas: através de uma imobiliária ou do(s) próprio(s) proprietário(s).

Em relação às agências imobiliárias é imperativo saber o seu nome, contacto e email. A cada imóvel podem estar associadas zero ou várias agências imobiliárias, mas, por outro lado, a cada agência imobiliária estão associados vários imóveis. Além disso, é necessário referir que a uma agência imobiliária estão associados vários vendedores, os quais possuem um nome, um contacto e um email. Cada vendedor trabalha exclusivamente para uma agência imobiliária.

Cada anúncio é consultado por um visitante, identificado pelo seu IP. De cada consulta é guardado a data e a hora da mesma. Os visitantes podem ser apresentados como: anónimos ou utilizadores, dos quais se sabe o seu nome, e-mail e password. Os utilizadores têm vantagens em relação aos anónimos, uma vez que, podem guardar como favorito os anúncios que mais lhes interessam. Além disso, é-lhes possível marcar entrevistas contactando uma imobiliária e escolhendo um ou diversos imóveis. A estas entrevistas estão associadas uma data e uma hora

Diagram

Description automatically generated

# 2.Esquema Relacional

* **Proprietario**(idProprietario, Contacto, Email, Nome);
* **Localizacao**(idLocalizacao, Concelho, Zona, Freguesia, Rua, Numero);
* **AgenciaImobiliaria**(idAgenciaImobiliaria, Contacto, Nome, Email);
* **Anuncio**(idAnuncio, Data, idImovel -> Imovel, idAgenciaImobiliaria -> AgenciaImobiliaria);
* **Visitante**(idVisitante, IP);
* **Vendedor**(idVendedor, Nome, Contacto, Email, idAgenciaImobiliaria -> AgenciaImobiliaria);
* **Imovel**(idImovel, Preço, Estado, Dimensão, Referencia, TipoNegocio, idLocalizacao -> Localizacao);
* **Apartamento**(idImovel -> Imovel , Tipo, Andar, Tipologia, NumCasasBanho);
* **Moradia**(idImovel -> Imovel, Tipo, NumPisos, Tipologia, NumCasasBanho);
* **Escritorio**(idImovel -> Imovel, Distribuicao);
* **Armazem**(idImovel -> Imovel);
* **Garagem**(idImovel -> Imovel, Vigilancia);
* **Terreno**(idImovel -> Imovel, Tipo);
* **IndividuoProp**(idProprietario -> Proprietario, NIdentificacao);
* **BancoProp**(idProprietario -> Proprietario, ReferenciaBanco);
* **Utilizador**(idVisitante -> Visitante, Email, Password, Nome);
* **Anonimo**(idVisitante -> Visitante);
* **Visita**(idAnuncio -> Anuncio, idVisitante -> Visitante, Data, Hora);
* **Entrevista**(idVisitante -> Utilizador, idImovel -> Imovel, Data, Hora, idAgenciaImobiliaria-> AgenciaImobiliaria);
* **PropImovel**(idProprietario -> Proprietario, idImovel -> Imovel);
* **PropAnuncio**(idProprietario -> Proprietario, idAnuncio -> Anuncio);
* **ImovelAgencia**(idImovel -> Imovel, IdAgenciaImobiliaria -> AgenciaImobiliaria);
* **Favorita**(idAnuncio -> Anuncio, idVisitante -> Utilizador);

# 3.Análise de Dependências Funcionais e Formas Normais

Fruto do esquema relacional previamente apresentado, calculou-se o conjunto de dependências funcionais associado a cada relação e eventuais violações à Forma Normal Boyce-Codd (BCNF) e 3ª Forma Normal (3ª FN). Na seguinte tabela é apresentada cada relação, as respetivas dependências funcionais e o possível cumprimento ou não da Forma Normal Boyce-Codd e 3ª Forma Normal.

Graphical user interface, application, table

Description automatically generated

Como condição, para uma relação estar na 3ª Forma Normal e, consequentemente, na Forma Normal Boyce-Codd, é necessário que para cada dependência funcional não trivial -> , seja uma superchave ou consista apenas em atributos primos. Assim, analisando a tabela previamente apresentada, conclui-se então que a relação “Localizacao” é a única que não respeita quer a Forma Normal Boyce-Codd quer a 3ª Forma Normal, visto que, Zona e Concelho não são atributos primos e Freguesia não é uma superchave. Desta forma, como a relação não está na 3ª Forma Normal, também não está na Forma Normal Boyce-Codd, visto que esta ainda é mais restrita que a anteriormente apresentada. Assim, foi necessária a sua decomposição que é demonstrada a seguir.

## 3.1.Decomposição

**1ª Decomposição**

Violação da BCNF

Freguesia ->Zona, Concelho

Decomposição Localizacao

S1(Freguesia, Zona, Concelho)

S2(Freguesia, Rua, Numero, idLocalizacao)

Cálculo das FDs e chaves de S1

Freguesia -> Zona, Concelho

Chave: {Freguesia}

Cálculo das FDs e chaves de S2

idLocalizacao -> Freguesia, Rua, Numero

Chave: {idLocalizacao}

**Resultado**

S1(Freguesia, Zona, Concelho)

S2(idLocalizacao, Freguesia, Rua, Numero)

As tabelas S1 e S2, resultantes da decomposição BCNF, foram nomeadas Localizacao\_Geral e Localizacao\_Exata, respetivamente.

## 3.2.Teste à junção sem perdas

Devido à decomposição, é necessário agora provar que, através da junção natural, é possível reconstruir a relação inicial. Desta forma levou-se a cabo o Teste à junção sem perdas apresentado a seguir.

S1(B,C,D)

S2(A,B,E,F)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A (idLocalizacao)** | **B (Freguesia)** | **C (Zona)** | **D (Concelho)** | **E (Rua)** | **F (Numero)** |
| A1 | B | C | D | E1 | F1 |
| A | B | C2 | D2 | E | F |

B-> C,D

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A (idLocalizacao)** | **B (Freguesia)** | **C (Zona)** | **D (Concelho)** | **E (Rua)** | **F (Numero)** |
| A1 | B | C | D | E1 | F1 |
| A | B | C | D | E | F |

Visto que a última tabela tem uma linha sem índices, conclui-se então que é uma decomposição sem perdas.

## 4. Restrições

De forma a assegurar a integridade da base de dados, foram adicionadas restrições limitando o tipo de dados inserido em cada tabela. Nos casos em que a restrição NOT NULL foi associada a um atributo, é obrigatório ser fornecido um valor não nulo. A restrição CHECK garante que todos os valores satisfaçam um determinado critério e a restrição UNIQUE impede a existência de valores iguais em tuplos diferentes. A restrição PRIMARY KEY é usada para identificar de forma única cada linha da relação. Foram, também, aplicadas FOREIGN KEYS a tabelas intrinsecamente relacionadas. Por último, foram adicionadas restrições no SQL do género ON DELETE CASCADE e ON UPDATE CASCADE, garantindo a integridade dos dados armazenados.

Em seguida, são apresentadas as restrições associadas a cada relação.

**AgenciaImobiliaria:**

* **PK\_AgenciaImobiliaria:** identificador da tabelaAgenciaImobiliaria, não podendo haver duas agências com o mesmo ID; restrição chave (primary key).
* **UNIQUE\_AgenciaImobiliaria\_Contacto:** não pode haver duas agências imobiliárias com o mesmo contacto; restrição chave (unique).
* **UNIQUE\_AgenciaImobiliaria\_Email:** não pode haver duas agências imobiliárias com o mesmo e-mail; restrição chave (unique).

**Localizacao\_Geral:**

* **PK\_ Localizacao\_Geral:** identificador da tabelaLocalizacao\_Geral, não podendo haver duas localizações gerais com a mesma freguesia; restrição chave (primary key).

**Localizacao\_Exata:**

* **PK\_Localizacao\_Exata:** identificador daLocalizacao\_Exata, não podendo haver duas localizações exatas com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_Freguesia:** chave estrangeira da tabela Localizacao\_Geral referente a Freguesia; integridade referencial (chave estrangeira).

**Imovel:**

* **PK\_Imovel:** identificador da tabela Imóvel, não podendo haver dois imóveis com o mesmo ID; restrição chave (primary key).
* **CHECK\_Imovel\_TipoNegocio:** restringe os dados inseridos no atributo TipoNegocio em 'Vender' ou 'Arrendar'; restrição CHECK.
* **UNIQUE\_Imovel\_Referencia:** não pode haver dois imóveis com a mesma referência; restrição chave (unique).
* **CHECK\_Imovel\_Estado:** restringe os dados inseridos no atributo Estado em 'Nova construção', 'Bom estado' ou 'Para recuperar'; restrição CHECK.
* **FOREIGNKEY\_idLocalizacao:** chave estrangeira da tabela Localizacao\_Exata referente ao idLocalizacao; integridade referencial (chave estrangeira).

**Apartamento:**

* **PK\_Apartamento:** identificador da tabela Apartamento, não podendo haver dois apartamentos com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idApart**: chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).
* **CHECK\_Apartamento\_Tipo:** restringe os dados inseridos no atributo Tipo em 'penthouse', 'duplex' ou 'apartamento'; restrição CHECK.
* **CHECK\_Apartamento\_Tipologia:** restringe os dados inseridos no atributo Tipologia em 'T0', 'T1', ‘T2’, ‘T3’ ou 'T4+'; restrição CHECK.

**Moradia:**

* **PK\_Moradia:** identificador da tabela Moradia, não podendo haver duas moradias com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idMoradia:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).
* **CHECK\_Moradia\_Tipo:** restringe os dados inseridos no atributo Tipo em 'isoladas', 'geminadas' ou 'em banda'; restrição CHECK.
* **CHECK\_Moradia\_Tipologia:** restringe os dados inseridos no atributo Tipologia em 'T0', 'T1', ‘T2’, ‘T3’ ou 'T4+'; restrição CHECK.

**Escritorio:**

* **PK\_Escritorio:** identificador da tabela Escritório, não podendo haver dois escritórios com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idEscritorio:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).
* **CHECK\_Escritorio\_Distribuicao:** restringe os dados inseridos no atributo Distribuicao em 'Indiferente', 'Open Space' ou 'Com divisorias'; restrição CHECK.

**Armazem:**

* **PK\_Armazem:** identificador da tabela Armazem, não podendo haver dois armazéns com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idArmazem:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).

**Garagem:**

* **PK\_Garagem:** identificador da tabela Garagem, não podendo haver duas garagens com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idGaragem:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).

**Terreno:**

* **PK\_Terreno:** identificador da tabela Terreno, não podendo haver dois terrenos com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idTerreno:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).
* **CHECK\_Terreno\_Tipo:** restringe os dados inseridos no atributo Tipologia em 'Urbano, 'Urbanizavel' ou 'Não Urbanizavel'; restrição CHECK.

**Proprietario:**

* **PK\_Proprietario:** identificador da tabela Proprietário, não podendo haver dois proprietários com o mesmo ID; restrição chave (primary key).
* **UNIQUE\_Proprietario\_Contacto:** não pode haver dois proprietários com o mesmo contacto; restrição chave (unique).
* **UNIQUE\_Proprietario\_Email:** não pode haver dois proprietários com o mesmo e-mail; restrição chave (unique).

**IndividuoProp:**

* **PK\_IndividuoProp:** identificador da tabela IndividuoProp, não podendo haver dois tuplos com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idIndividuoProp:** chave estrangeira da tabela Proprietario referente ao idProprietario; integridade referencial (chave estrangeira).
* **UNIQUE\_IndividuoProp\_NIdentificacao:** não pode haver dois indivíduos proprietários com o mesmo número de identificação; restrição chave (unique).

**BancoProp:**

* **PK\_BancoProp:** identificador da tabela BancoProp, não podendo haver dois tuplos com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idBancoProp:** chave estrangeira da tabela Proprietario referente ao idProprietario; integridade referencial (chave estrangeira).
* **UNIQUE\_BancoProp\_Referencia:** não pode haver dois bancos proprietários com a mesma referência; restrição chave (unique).

**Visitante:**

* **PK\_Visitante:** identificador da tabela Visitante, não podendo haver dois visitantes com o mesmo ID; restrição chave (primary key).
* **UNIQUE\_Visitante\_IP:** não pode haver dois visitantes com o mesmo IP; restrição chave (unique).

**Anonimo:**

* **PK\_Anonimo:** identificador da tabela Anonimo, não podendo haver dois visitantes anónimos com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idAnonimo:** chave estrangeira da tabela Visitante referente ao idVisitante; integridade referencial (chave estrangeira).

**Utilizador:**

* **PK\_Utilizador:** identificador da tabela Utilizador, não podendo haver dois utilizadores com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_idVisitante:** chave estrangeira da tabela Visitante referente ao idVisitante; integridade referencial (chave estrangeira).
* **UNIQUE\_Utilizador\_Email:** não pode haver dois utilizadores com o mesmo e-mail; restrição chave (unique).

**Anuncio:**

* **PK\_Anuncio:** identificador da tabela Anuncio, não podendo haver dois anúncios com o mesmo ID; restrição chave (primary key).
* **FOREIGNKEY\_Anuncio\_idImovel:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).
* **FOREIGNKEY\_Anuncio\_idAgenciaImobiliaria:** chave estrangeira da tabela AgenciaImobiliaria referente ao idAgenciaImobiliaria; integridade referencial (chave estrangeira).

**Vendedor:**

* **PK\_Vendedor:** identificador da tabela Vendedor, não podendo haver dois vendedores com o mesmo ID; restrição chave (primary key).
* **UNIQUE\_Vendedor\_Email:** não pode haver dois vendedores com o mesmo e-mail; restrição chave (unique).
* **FOREIGNKEY\_Vendedor\_idAgenciaImobiliaria:** chave estrangeira da tabela AgenciaImobiliaria referente ao idAgenciaImobiliaria; integridade referencial (chave estrangeira).

**Visita:**

* **PK\_Visita\_idAnuncio\_idVisitante:** identificador da tabela Visita, não podendo haver duas visitas com o mesmo idVisitante e idAnuncio simultaneamente; restrição chave (primary key).
* **FOREIGNKEY\_Visita\_idAnuncio:** chave estrangeira da tabela Anuncio referente ao idAnuncio; integridade referencial (chave estrangeira).
* **FOREIGNKEY\_Visita\_idVisitante:** chave estrangeira da tabela Visitante referente ao idVisitante; integridade referencial (chave estrangeira).

**PropImovel:**

* **PK\_PropImovel\_idProp\_idImovel:** identificador da tabela PropImovel, não podendo haver dois tuplos com o mesmo idProprietario e idImovel simultaneamente; restrição chave (primary key).
* **FOREIGNKEY\_PropImovel\_idProp:** chave estrangeira da tabela Proprietario referente ao idProprietario; integridade referencial (chave estrangeira).
* **FOREIGNKEY\_PropImovel\_idImov:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).

**ImovelAgencia:**

* **PK\_ImovelAgencia\_idImovel\_idAgenciaImobiliaria:** identificador da tabela ImovelAgencia, não podendo haver dois tuplos com o mesmo idImovel e idAgenciaImobiliaria simultaneamente; restrição chave (primary key).
* **FOREIGNKEY\_ImovelAgencia\_idImovel:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).
* **FOREIGNKEY\_ImovelAgencia\_idAgenciaImobiliaria:** chave estrangeira da tabela AgenciaImobiliaria referente ao idAgenciaImobiliaria; integridade referencial (chave estrangeira).

**PropAnuncio:**

* **PK\_PropAnuncio\_idImovel\_idAnuncio:** identificador da tabela PropAnuncio, não podendo haver dois tuplos com o mesmo idImovel e idAnuncio simultaneamente; restrição chave (primary key).
* **FOREIGNKEY\_PropAnun\_idProp:** chave estrangeira da tabela Proprietario referente ao idProprietario; integridade referencial (chave estrangeira).
* **FOREIGNKEY\_PropAnun\_idAnuncio:** chave estrangeira da tabela Anuncio referente ao idAnuncio; integridade referencial (chave estrangeira).

**Favorita:**

* **PK\_Favorita\_idAnuncio\_idVisitante:** identificador da tabela Favorita, não podendo haver dois tuplos com o mesmo idAnuncio e idVisitante simultaneamente; restrição chave (primary key).
* **FOREIGNKEY\_Favorita\_idAnuncio:** chave estrangeira da tabela Anuncio referente ao idAnuncio; integridade referencial (chave estrangeira).
* **FOREIGNKEY\_Favorita\_idVisitante:** chave estrangeira da tabela Utilizador referente ao idVisitante; integridade referencial (chave estrangeira).

**Entrevista:**

* **PK\_Entrevista\_idVisitante\_idImovel:** identificador da tabela Entrevista, não podendo haver duas entrevistas com o mesmo idVisitante e idImovel simultaneamente; restrição chave (primary key).
* **FOREIGNKEY\_Entrevista\_idVisitante:** chave estrangeira da tabela Utilizador referente ao idVisitante; integridade referencial (chave estrangeira).
* **FOREIGNKEY\_Entrevista\_idImovel:** chave estrangeira da tabela Imovel referente ao idImovel; integridade referencial (chave estrangeira).
* **FOREIGNKEY\_Entrevista\_idAgenciaImobiliaria:** chave estrangeira da tabela AgenciaImobiliaria referente ao idAgenciaImobiliaria; integridade referencial (chave estrangeira).