

Gestão de Condomínios – Fase 2



Trabalho realizado pelos alunos do Grupo 404:

António Palmeirim, 63667

Duarte Inácio, 62397

Daniel Cruz, 62436

Turno Prático 4

Projeto desenvolvido no servidor da Oracle

Descrição detalhada do projeto

A nossa ideia é criar uma base de dados para gerir os condomínios de uma empresa, em que cada condomínio tem várias propriedades/unidades, funcionários, serviços e assembleias. Cada propriedade tem um morador/dono associado, um valor de quotas que tem de pagar mensalmente para o condomínio, sendo estas quotas a “receita” do condomínio e as despesas são os serviços e funcionários, ou seja, as quotas é o que faz aumentar o saldo da conta bancária do condomínio e os serviços e os funcionários são os que tiram saldo à conta bancária do condomínio. Para o cálculo do salário de um condomínio, decidimos em vez da criação de uma entidade relativa à conta bancária, fazemos a seguinte conta: a diferença entre a soma do valor de todas as quotas pagas e a soma do preço de todos os serviços realizados com o valor de todos os recibos de vencimento de todos os funcionários do condomínio.

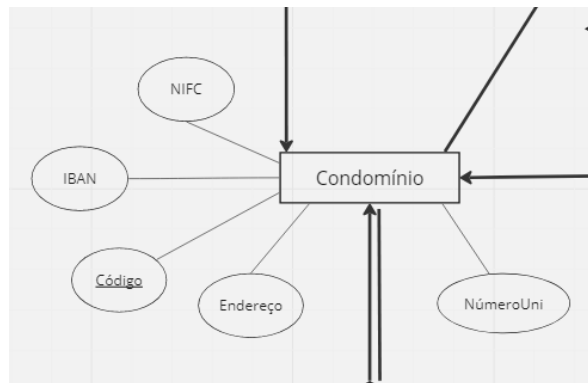
Cada pessoa pode ser um morador ou um funcionário. No caso do morador, tem a informação relativa à data de entrada e a data de saída, caso já tenha sido transmitida uma data de saída. No caso do funcionário, este tem a informação do salário, da função que desempenha, a data de entrada e uma data de saída, caso já tenha sido transmitida, como nos moradores. Cada funcionário pode ter ou não recibos de vencimento (No caso em que o funcionário acabou de ser contratado, não tem qualquer recibo de vencimento associado), mas cada recibo de vencimento apenas tem um funcionário. Os recibos de vencimento têm a informação da data do recibo e o valor que corresponde ao salário do funcionário. (Nos meses junho e dezembro o salário é multiplicado do 2 já que é o subsídio de férias)

Em cada assembleia são discutidos vários assuntos e cada assunto tem informação do id do assunto, o nome do assunto e quantos votos a favor desse assunto houve.

As empresas têm informação do nome e do número de telefone da empresa e cada empresa tem serviços disponíveis com a informação do nome do serviço. Caso seja realizado um serviço num condomínio por uma empresa, o serviço passa a ser, para além de disponível, serviço realizado, com a informação da data em que foi realizado, o preço do serviço e as observações feitas ao serviço.

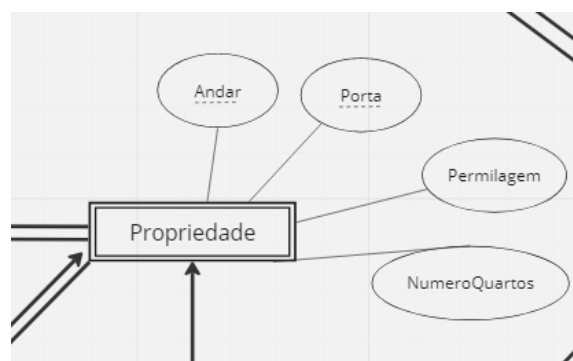
Há assim que ter em conta as seguintes entidades e as suas descrições:

- Condomínio: representa um prédio/edifício. Cada condomínio deve ter um código que o difere dos outros, como por exemplo, um condomínio na Rua das Flores porta 7, o código poderia ser RF7LX. Neste caso (caso mais específico, pois existe a rua das flores em Lisboa e no Porto) teríamos de conseguir diferenciar as Rua das Flores de Lisboa e do Porto já que uma empresa pode ter o “azar” de ter dois ou mais condomínios em ruas e portas iguais, logo uma ideia seria colocar uma abreviatura do concelho no final do código. O condomínio também vai ter um endereço/morada, número de propriedades/unidades, número fiscal e o IBAN.



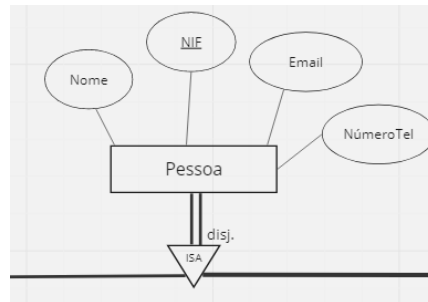
- **Propriedade**: representa uma unidade/apartamento/casa/loja individual dentro do condomínio. Cada propriedade tem informação sobre o andar e a porta do apartamento, número de quartos, pernilagem (valor representativo de cada

total do
edifício



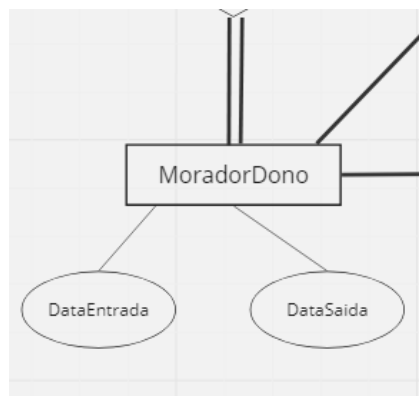
propriedade no
edifício (o total do
representa 1000)).

- Pessoa:
que pode ser um
Funcionário.
informação sobre
o Email caso
telefone.



representa uma pessoa
MoradorDono ou um
Cada pessoa tem a
o NIF, o Nome da pessoa,
tenha, e o número de

- MoradorDono:
vive numa
Cada morador
telefone, email,
contribuinte
na propriedade e
tenha sido

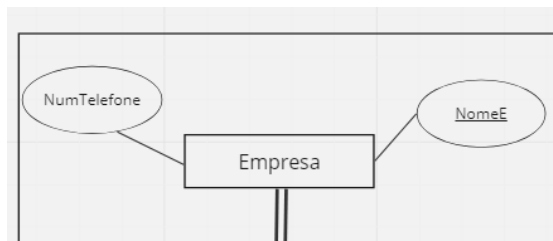


representa a pessoa que
unidade do condomínio.
tem o nome, número de
número fiscal de
(NIF), a data de entrada
a data de saída cajo já
transmitida.

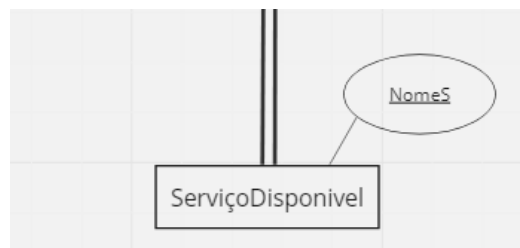
- Funcionário(s) (Não são os funcionários dos serviços, mas sim do condomínio):
representa a(s) pessoa(s) que trabalha(m) no condomínio, como por exemplo,
um porteiro, zelador (cuida da manutenção e do dia a dia do prédio). Cada

funcionário tem, para além da informação sobre a pessoa, a informação sobre a função, data de entrada como funcionário no condomínio e a data de saída como funcionário no condomínio, caso exista, e salário. (Despesa)

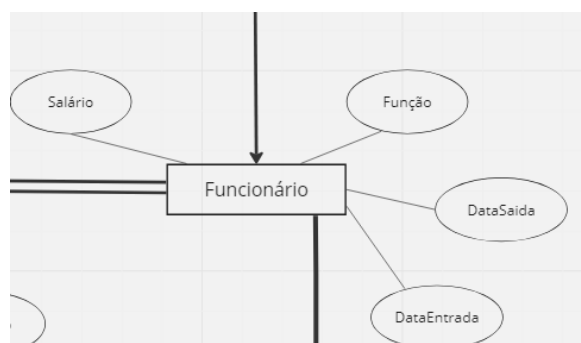
- Empresa: representa uma empresa que fornece serviços para os condomínios. Cada empresa tem a informação do seu nome e do número de telefone.



- ServiçoDisponível: representa um serviço que é oferecido aos moradores do condomínio, por exemplo, limpeza, segurança ou manutenção (elevadores, lâmpadas). Cada serviço tem um nome (nome do tipo de serviço prestado).



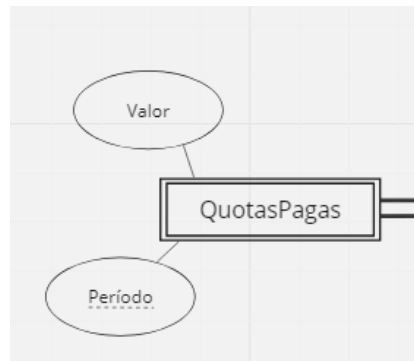
- ServiçoRealizado: representa um serviço que já foi realizado a um condomínio de entre os serviços disponíveis. Cada ServiçoRealizado tem a informação da data, do preço e a observação feita ao serviço. Os serviços realizados são entidade (Empresa, já que este realizado se disponível empresa.



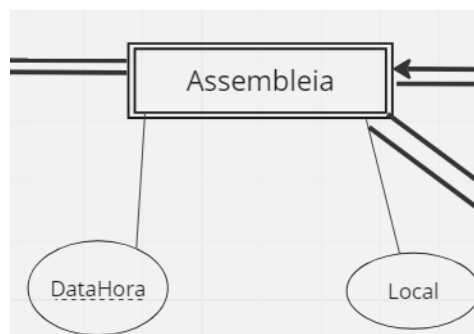
fraca da agregação
ServicoDisponível)
só poderá ser
for um serviço
fornecido pela

- QuotasPagas: representa as taxas que cada morador de uma determinada propriedade tem de pagar regularmente ao condomínio, que são usadas para cobrir as despesas comuns a todos os moradores, como por exemplo os serviços. As quotas pagas do condomínio têm informação sobre o valor e o período (mês da quota). Quotas pagas é entidade fraca de cada propriedade, pois cada propriedade tem uma quantidade de quotas pagas diferente das outras propriedades (Receita).

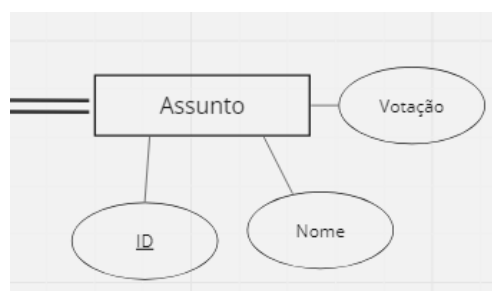




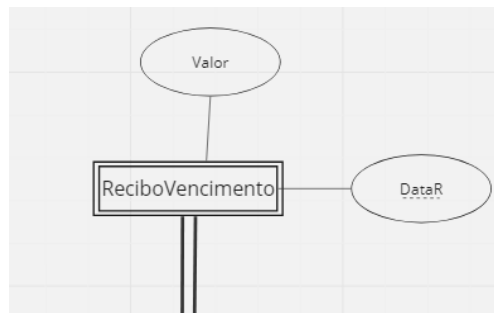
- **Assembleia**: representa as reuniões de condomínio. Cada assembleia inclui a data, a hora e o local.



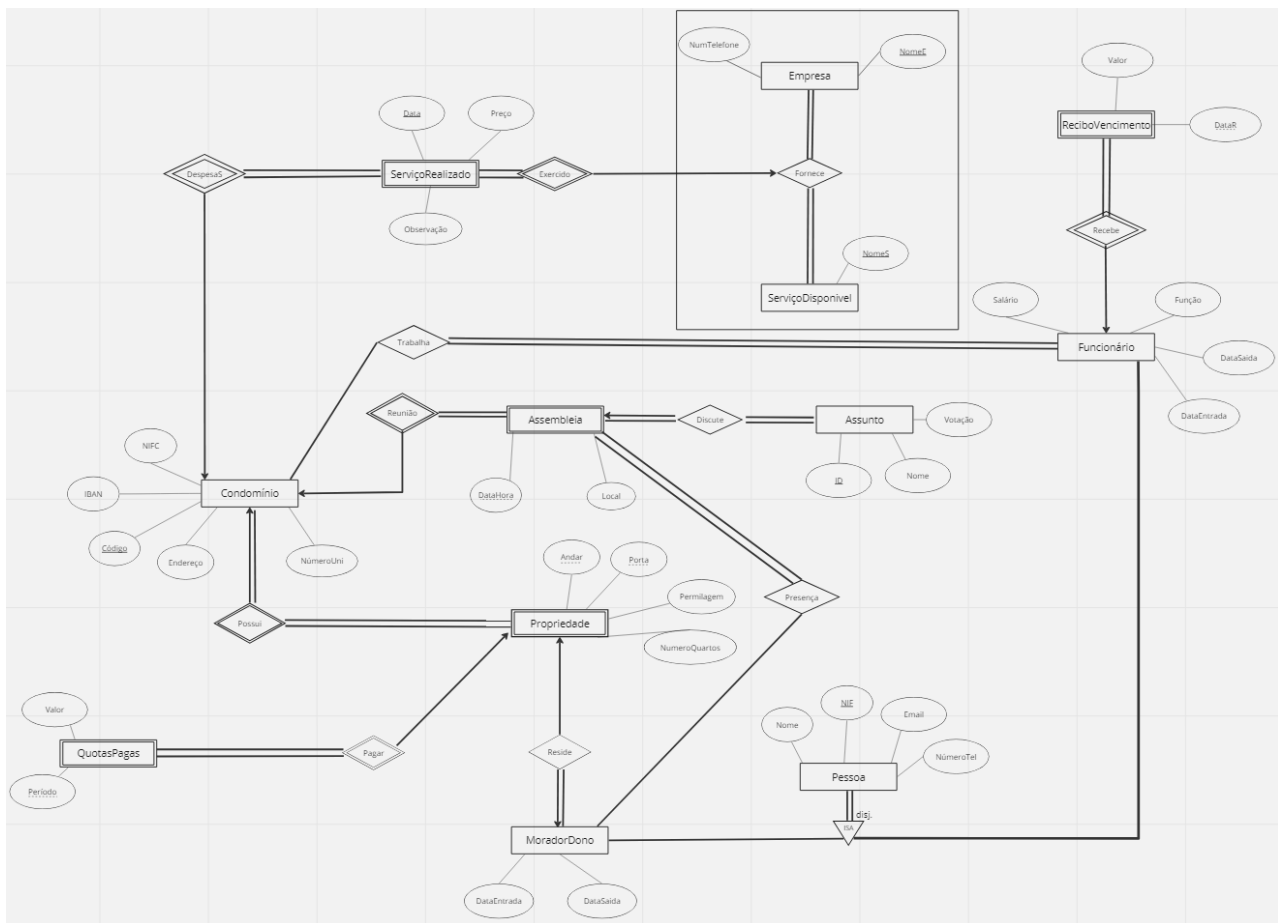
- **Assunto**: representa o assunto discutido numa assembleia. Cada assunto tem a informação do ID do assunto, o Nome do assunto (Tema) e o número de votos do assunto.



- **ReciboVencimento**: representa a prova de como foi efetuado o pagamento do salário a um determinado funcionário. Cada ReciboVencimento tem informação da data do recibo e o valor, que corresponde ao valor do salário do funcionário, exceto em junho e dezembro em que o valor é multiplicado por 2, já que estes meses normalmente correspondem aos meses em que os funcionários recebem os subsídios de férias.



Modelo ER Gestão de Condomínios



https://miro.com/welcomeonboard/aUFnaTE3c0dISDM2UG0wWVAydmkzWVdT1hMU1RTYmt5UTB5VnFha0IETFZ6RE1LVk5VMTdMTkZwTEM5ZzhRSHwzNDU4NzY0NTQ0MTYwODE2Mjc3fDI=?share_link_id=474356135350

Agora vamos explicar um pouco as relações utilizadas e detalhes do modelo ER:

- Decidimos fazer uma agregação entre Empresa e ServiçoDisponível, pois um serviço apenas é realizado caso seja um serviço disponível pela empresa.
- Fornece: é uma relação entre Empresa e ServiçoDisponível, em que todas as empresas têm pelo menos um serviço disponível, se não, não existiria sequer a empresa e todos os serviços disponíveis tem pelos menos uma empresa, já que o mesmo serviço pode ser prestado por empresas diferentes.
- Exercido: é uma relação entre ServiçoRealizado e a Agregação (Empresa e ServiçoDisponível), em que todos os serviços realizados tem uma e apenas uma agregação e toda a agregação pode ou não ter serviços realizados.

- DespesaS: é uma relação entre Condomínio e ServiçosRealizados, em que todos os serviços realizados tem apenas um condomínio, mas os condomínios podem ou não ter serviços realizados.
- Trabalha: é uma relação entre Funcionário e Condomínio, em que todos os funcionários têm pelo menos um condomínio, mas os condomínios podem ou não ter funcionários.
- Reunião: é uma relação entre Assembleia e Condomínio, em que todas as assembleias têm apenas um condomínio, mas os condomínios podem ou não ter assembleias.
- Discute: é uma relação entre Assunto e Assembleia, em que todos os assuntos têm apenas uma assembleia e todas as assembleias têm pelo menos um assunto.
- Presença: é uma relação entre MoradorDono e Assembleia, em que todas as assembleias têm pelo menos um morador, se não, não faria sentido haver uma assembleia e os moradores podem ou não estar presentes numa assembleia.
- Possui: é uma relação entre Condomínio e Propriedade, em que todos os condomínios possuem pelo menos uma propriedade e todas as propriedades estão associadas a apenas um condomínio.
- Pagar: é uma relação entre Propriedade e Quotaspagas, em que todas as quotas são pagas por apenas uma propriedade, mas uma propriedade pode ou não ter quotas pagas.
- Reside: é uma relação entre Propriedade e MoradorDono, em que todos os moradores residem em apenas uma propriedade e uma propriedade tem apenas um morador ou nenhum.
- Recebe: é uma relação entre Funcionário e ReciboVencimento, em que todos os recibos de vencimento tem apenas um funcionário, mas um funcionário pode ou não ter recibos de vencimento.
- Os serviços e os funcionários são despesas diferentes, pois os serviços podem ser cobrados diariamente/semanalmente por exemplo, e os funcionários têm, normalmente, um salário mensal fixo, por tanto decidimos separar estas despesas.

Modelo relacional Gestão de Condomínios

Estas são as tabelas das entidades (as chaves primárias estão sublinhadas):

- Condomínio (Código, IBAN, Endereço, NúmeroUni, NIFC)
- Propriedade (Andar, Porta, Código, Permilagem, NúmeroQuartos)
Código é CE de Condomínio
- Assembleia (HoraData, Código, Local)
Código é CE de Condomínio
- Assunto (ID, Nome, Votação, HoraData, Código)
(HoraData, Código) é CE de Assembleia
- Pessoa (NIF, Nome, Email, NúmeroTel)
- MoradorDono (NIF, DataEntrada, DataSaida, Andar, Porta, Código)
NIF é CE de Pessoa
(Andar, Porta, Codigo) é CE de Propriedade
- Funcionário (NIF, Função, DataEntrada, DataSaída, Salário)
NIF é CE de Pessoa
- QuotasPagas (Período, Código, Andar, Porta, Valor)
(Código, Andar, Porta) é CE de Propriedade
- ReciboVencimento(Datar, NIF, Valor)
NIF é CE de Funcionário
- ServiçoDisponível (Nomes)
- ServiçoRealizado(Data, Nomes, Código, Nomee, Preço, Observação)
(Nomes, Nomee) é CE de Fornece
Código é CE de Condomínio
- Empresa (Nomee, Numtelefone)

E estas são as tabelas das relações (as chaves primárias estão sublinhadas):

- Presença (NIF, HoraData, Código)
NIF é CE de MoradorDono
(HoraData, Código) é CE de Assembleia
- Trabalha (Código, NIF)
Código é CE de Condomínio
NIF é CE de Funcionário
- Fornece (Nomee, Nomes)
Nomee é CE de Empresa
Nomes é CE de ServiçoDisponível

Views, Triggers e Funções

Views

```
CREATE OR REPLACE FORCE EDITIONABLE VIEW "DESPESAS_SERVICOS" ("CODIGO", "TOTAL_PRECO") AS
  SELECT CODIGO, nvl(SUM(PRECO),0) AS TOTAL_PRECO
FROM SERVICOREALIZADO
GROUP BY CODIGO
/
```

- DESPESAS_SERVICOS - view usada para calcular o valor total das despesas dos condomínios devido aos serviços realizados. Corresponde a uma tabela com o código do condomínio e o valor associado.

```
CREATE OR REPLACE FORCE EDITIONABLE VIEW "DESPESA_FUNC" ("CODIGO", "TOTAL_FUNC") AS
  SELECT CONDOMINIO.CODIGO, nvl(SUM(FUNRECEBNOME.VALOR),0)
FROM CONDOMINIO, TRABALHA, FUNCIONARIO, FUNRECEBNOME
WHERE CONDOMINIO.CODIGO = TRABALHA.CODIGO AND FUNRECEBNOME.NIF = FUNCIONARIO.NIF AND FUNCIONARIO.NIF = TRABALHA.NIF
GROUP BY CONDOMINIO.CODIGO
/
```

- DESPESA_FUNC - view usada para calcular o valor total das despesas dos condomínios devido aos funcionários que trabalham para certo condomínio. Corresponde a uma tabela com o código do condomínio e o valor associado.

```
CREATE OR REPLACE FORCE EDITIONABLE VIEW "FUNCSEMRECIBO" ("NIF") AS
  select nif from funcionario
minus
select nif from recibovencimento
/
```

- FUNCSEMRECIBO - view usada para ver os funcionários mais recentemente adicionados, pois estes ainda não tem recibos de vencimento.

```
CREATE OR REPLACE FORCE EDITIONABLE VIEW "NAMEFUNC" ("NOME", "NIF", "EMAIL", "NUMEROTEL", "FUNCAO", "SALARIO", "DATAENTRADA", "DATASAIIDA", "CODIGO") AS
select nome, funcionario.nif, email, numerotel, funcao, salario, dataentrada, datasaida, trabalha.codigo
FROM pessoa, funcionario, trabalha
where pessoa.nif = funcionario.nif and trabalha.nif = funcionario.nif
/
```

- NAMEFUNC - view auxiliar no trigger para a inserção de um funcionário, associando logo este a um condomínio, e o funcionário também é logo adicionado a tabela das pessoas.

```
CREATE OR REPLACE FORCE EDITIONABLE VIEW "NAMEMORADOR" ("NOME", "EMAIL", "NIF", "NUMEROTEL", "DATAENTRADA", "DATASAIIDA", "ANDAR", "PORTA", "CODIGO") AS
select nome, email, moradordono.nif, numerotel, dataentrada, datasaida,propriedade.andar,propriedade.porta, condominio.codigo
from pessoa, moradordono, propriedade, condominio
where pessoa.nif = moradordono.nif and moradordono.codigo = propriedade.codigo
and propriedade.codigo = condominio.codigo and moradordono.andar = propriedade.andar and
moradordono.porta = propriedade.porta
/
```

- NAMEMORADOR - view auxiliar no trigger para a inserção de um morador, associando logo este a uma propriedade, e o morador também é logo adicionado a tabela das pessoas.

```
CREATE OR REPLACE FORCE EDITIONABLE VIEW "NOMEDOFUNCIONARIO" ("NOME", "NIF") AS
select nome, funcionario.nif
FROM pessoa, funcionario
where pessoa.nif = funcionario.nif
/
```

- NOMEDOFUNCIONARIO - view auxiliar que corresponde a tabela dos nomes de todos os funcionários e os seus respectivos NIFS.

```
CREATE OR REPLACE FORCE EDITIONABLE VIEW "RENDIMENTOTOTALPROP" ("CODIGO", "TOTAL_REVENUE") AS
SELECT codigo, nvl(SUM(valor),0)
FROM quotaspagas
GROUP BY codigo
/
```

- RENDIMENTOTOTALPROP - view usada para calcular o valor total de rendimento dos condomínios devido ao pagamento de quotas. Corresponde a uma tabela com o código do condomínio com o valor associado.

Triggers

```
CREATE OR REPLACE EDITIONABLE TRIGGER "ADICIONAFUNCIONARIO"  
INSTEAD OF INSERT ON NAMEFUNC  
FOR EACH ROW  
BEGIN  
INSERT INTO PESSOA (NIF, NOME, EMAIL, NUMEROTEL) VALUES (:NEW.NIF, :NEW.NOME, :NEW.EMAIL, :NEW.NUMEROTEL);  
INSERT INTO FUNCIONARIO (NIF, FUNCAO, DATAENTRADA, DATASAIDA, SALARIO) VALUES (:NEW.NIF, :NEW.FUNCAO, :NEW.DATAENTRADA, :NEW.DATASAIDA, :NEW.SALARIO);  
INSERT INTO TRABALHA (CODIGO, NIF) VALUES (:NEW.CODIGO, :NEW.NIF);  
END;
```

- ADICIONAFUNCIONARIO - trigger usado para adicionar um funcionário. Usando o INSTEAD OF, recebemos toda a informação da view FUNCIONARIOCONDOMINIO e inserimos os dados recebidos nas tabelas respectivas, sendo essas, a tabela das pessoas (com os dados: NIF, Nome, Email, Numerotel), a tabela dos funcionários (com os dados: NIF, Funcao, Dataentrada, Datasaida, Salário) e a tabela da relação entre os funcionários e os condomínios (com os dados: Codigo, NIF).

```
CREATE OR REPLACE EDITIONABLE TRIGGER "ADICIONAMORADOR"  
INSTEAD OF INSERT ON NAMEMORADOR  
FOR EACH ROW  
BEGIN  
INSERT INTO PESSOA (NIF, NOME, EMAIL, NUMEROTEL) VALUES (:NEW.NIF, :NEW.NOME, :NEW.EMAIL, :NEW.NUMEROTEL);  
INSERT INTO MORADORONHO (NIF, DATAENTRADA, DATASAIDA, ANDAR, PORTA, CODIGO) VALUES (:NEW.NIF, :NEW.DATAENTRADA, :NEW.DATASAIDA, :NEW.ANDAR, :NEW.PORTA, :NEW.CODIGO);  
END;
```

- ADICIONAMORADOR - trigger usado para adicionar um morador. Usando o INSTEAD OF, recebemos toda a informação da view NAMEMORADOR e inserimos os dados recebidos nas tabelas respectivas, sendo essas, a tabela das pessoas (com os dados: NIF, Nome, Email, Numerotel), a tabela dos moradores (com os dados: NIF, Dataentrada, Datasaida, Andar, Porta, Codigo).

```
CREATE OR REPLACE EDITIONABLE TRIGGER "ID"  
BEFORE INSERT ON ASSUNTO  
FOR EACH ROW  
DECLARE  
    id number;  
BEGIN  
    SELECT create_id.nextval  
    INTO id  
    FROM dual;  
    :new.id := id;  
END;
```

- ID - trigger usado para gerar um código sequencial e único para os assuntos.

```

CREATE OR REPLACE EDITIONABLE TRIGGER "LIMITEPROP"
BEFORE INSERT ON propriedade
FOR EACH ROW
DECLARE
    nprop NUMBER;
    numero_unitotal NUMBER;
BEGIN
    nprop := propriedade_count(:new.codigo);
    SELECT numerouni INTO numero_unitotal
    FROM condominio
    WHERE codigo = :new.codigo;

    IF nprop = numero_unitotal THEN
        RAISE_APPLICATION_ERROR(-20100, 'O limite de propriedades no condomínio foi atingido.');
```

- LIMITEPROP - trigger usado para verificar se o limite de propriedades de um certo condomínio foi atingido ou não, pois não é possível inserir mais propriedades num condomínio do que o número estipulado inicialmente.

```

CREATE OR REPLACE EDITIONABLE TRIGGER "RECEBFUN"
BEFORE INSERT ON RECIBOVENCIMENTO
FOR EACH ROW
DECLARE
    valorR number;
    month number;
BEGIN
    SELECT EXTRACT(MONTH FROM :new.datar) INTO month FROM DUAL;
    SELECT salario INTO valorR
    FROM funcionario
    WHERE nif = :new.nif;
    IF month = 6 OR month = 12 THEN
        :new.valor := 2 * valorR;
    ELSE
        :new.valor := valorR;
    END IF;
END;
```

- RECEBFUN - trigger usado para inserir o valor do recibo de vencimento de um funcionário a partir da informação da data e do NIF do funcionário. A partir do NIF obtemos o salário, e a partir da data verificamos se o mês é um mês de subsídio (Junho e Dezembro), se for, o valor é o valor do salário duplicado, caso contrário o valor é apenas o valor do salário.

```

CREATE OR REPLACE EDITIONABLE TRIGGER "VERIFICA_PERMILAGEM"
BEFORE INSERT ON PROPRIEDADE
FOR EACH ROW
DECLARE
    TOTAL_PERMILAGEM NUMBER(4,0);
    NPROPRIEDADE NUMBER(2,0);
    NUMEROUNI_CONDOMINIO NUMBER(2,0);
BEGIN
    NPROPRIEDADE := PROPRIEDADE_COUNT(:NEW.CODIGO);
    TOTAL_PERMILAGEM := SUM_PERMILAGEM(:NEW.CODIGO) + :NEW.PERMILAGEM;
    SELECT NUMEROUNI INTO NUMEROUNI_CONDOMINIO
    FROM CONDOMINIO
    WHERE CODIGO = :NEW.CODIGO;
    IF NPROPRIEDADE = NUMEROUNI_CONDOMINIO - 1 AND TOTAL_PERMILAGEM <> 1000 THEN
        DECLARE
            FATOR NUMBER;
        BEGIN
            FATOR := 1000/TOTAL_PERMILAGEM;
            UPDATE PROPRIEDADE
            SET PERMILAGEM = PERMILAGEM * FATOR
            WHERE CODIGO = :NEW.CODIGO;
            :NEW.PERMILAGEM := :NEW.PERMILAGEM * FATOR;
        END;
    END IF;
END;

```

- VERIFICA_PERMILAGEM - trigger usado para ajustar os valores das permilagens de todas as propriedades de um condomínio. Vai ser calculado o valor total da permilagem de cada propriedade, um fator (1000/valor total de permilagem de todas as propriedades) e multiplicado este fator por todas as permilagens do condomínio, ficando assim todas as permilagem do condomínio ajustadas à permilagem total do condomínio que é 1000.

```

create or replace TRIGGER REMOVEFUNCIONARIO
BEFORE DELETE ON PESSOA
FOR EACH ROW
BEGIN
    DELETE FROM TRABALHA where nif = :old.nif;
    DELETE FROM FUNCIONARIO where nif = :old.nif;
    DELETE FROM RECIBOVENCIMENTO where nif = :old.nif;
END;

```

- REMOVEFUNCIONARIO - trigger usado para quando removemos uma pessoa que é um funcionário, removemos também esse funcionário da relação TRABALHA, da entidade FUNCIONARIO e da entidade RECIBOVENCIMENTO. Assim removemos a pessoa dos funcionários, tiramos todos os recibos associados ao funcionário e tiramos o funcionário do(s) condomínio(s) a quais ele estava associado.

Funções

```
create or replace FUNCTION PROPRIEDADE_COUNT(VCODE VARCHAR2)
RETURN NUMBER
IS
    PROPRIEDADE_COUNT NUMBER;
BEGIN
    SELECT COUNT(*) INTO PROPRIEDADE_COUNT
    FROM PROPRIEDADE
    WHERE CODIGO = VCODE;

    RETURN PROPRIEDADE_COUNT;
END;

/
```

- PROPRIEDADE_COUNT - Função auxiliar para calcular a quantidade de propriedade é que existem num certo condomínio.

```
create or replace FUNCTION SUM_PERMILAGEM(VCODE VARCHAR2)
RETURN NUMBER
IS
    PROPRIEDADE_SUM NUMBER;
BEGIN
    SELECT SUM(PERMILAGEM) INTO PROPRIEDADE_SUM
    FROM PROPRIEDADE
    WHERE CODIGO = VCODE;

    RETURN PROPRIEDADE_SUM;
END;

/
```

- SUM_PERMILAGEM - Função auxiliar para calcular a permilagem total de um certo condomínio.

```
create or replace FUNCTION verificarEstadoAssunto
RETURN VARCHAR2
IS
    vResultado VARCHAR2(15);
    vRandomNumber NUMBER;
BEGIN
    vRandomNumber := FLOOR(DBMS_RANDOM.VALUE(0, 3));
    CASE vRandomNumber
        WHEN 0 THEN
            vResultado := 'Reprovado';
        WHEN 1 THEN
            vResultado := 'Aprovado';
        WHEN 2 THEN
            vResultado := 'Em Discussão';
    END CASE;
    RETURN vResultado;
END;
```

- VERIFICARESTADOASSUNTO - Função que confere aleatoriamente ao assunto criado, o seu estado. Este estado pode tomar três valores diferentes, “Aprovado”, “Reprovado” e “Em discussão”.

Limitações/Opções para a implementação da BD

O nosso grupo tomou algumas decisões para otimizar a base de dados, e para simplificar alguns aspetos que ficariam desnecessariamente complicadas.

Inicialmente no caso da relação entre as propriedades e os moradores, um morador está apenas associado, isto é, reside apenas numa e obrigatoriamente numa propriedade. Para o caso da propriedade, uma propriedade tem apenas um morador associado a ele, mas também pode não ter nenhum.

Continuando, para o caso das pessoas, uma é obrigatoriamente um funcionário ou um morador, não podendo ser os dois ao mesmo tempo. Optamos por esta opção para simplificar na inserção e remoção de um morador ou de um funcionário visto que estas duas entidades funcionam de forma independente uma da outra.

Nos assuntos, para otimizar a criação e alteração dos assuntos de uma assembleia, ele é independente da entidade assembleia pois tem um ID único gerada de forma sequencial. Mas como tem participação total na relação com a "Assembleia", podemos garantir que todos os assuntos com ID distinto estão associados a uma e uma só assembleia.

Um serviço realizado é entidade fraca do "Condomínio" e da agregação "Fornece" que representa todos os serviços disponíveis. Assim sendo, podemos garantir que todos os serviços realizados estejam associados a apenas um condomínio e que existe um serviço disponível com uma certa empresa associada, sendo por isso, que também foi criada para a agregação "Fornece". Além disso, a agregação foi criada para que uma empresa possa disponibilizar vários serviços, e para que um serviço possa ser realizado por várias empresas.

Por final, a entidade que representa os recibos de vencimento de um funcionário funciona como entidade fraca de um funcionário pois para existir um recibo tem de estar associado a um funcionário. O recibo de vencimento tem o atributo valor, mas não é possível alterá-lo nem dar valor, pois este está dependente do salário do funcionário e da data em que o recibo é criado.

Consultas interessantes

- Consulta relativa à soma das quotas de todas as propriedades, o valor total de todos os serviços realizados e a soma do valor de todos os recibos de vencimento dos funcionários de um condomínio. Esta informação é importante para a consulta do salário de um determinado condomínio, tal como está nesta SQL Query.

```
select condominio.codigo, nvl(total_fun,0), nvl(total_preco,0), nvl(total_revenue,0),  
nvl(total_revenue,0) - nvl(total_fun,0) - nvl(total_preco,0)  
FROM CONDOMINIO  
FULL JOIN DESPESA_FUNC ON CONDOMINIO.codigo = DESPESA_FUNC.codigo  
FULL JOIN DESPESAS_SERVICOS ON CONDOMINIO.codigo = DESPESAS_SERVICOS.codigo  
FULL JOIN RENDIMENTOTOTALPROP ON CONDOMINIO.codigo = RENDIMENTOTOTALPROP.codigo
```

- Consulta relativa aos funcionários que tenham acabado de ser contratados, ou seja os que ainda não possuem qualquer recibo de vencimento em seu nome. Utilizamos para além da view FUNCSEMRECIBO a seguinte SQL Query.

```
select FUNCSEMRECIBO.nif, nome  
from FUNCSEMRECIBO, NOMEDEFUNCIONARIO  
WHERE FUNCSEMRECIBO.NIF = NOMEDEFUNCIONARIO.NIF  
  
CREATE OR REPLACE FORCE EDITIONABLE VIEW "FUNCSEMRECIBO" ("NIF") AS  
select nif from funcionario  
minus  
select nif from recibovencimento  
/
```

- Consultas relativas à soma do preço de todos os serviços, à soma do valor de todos os recibos de vencimento de todos os funcionários e o valor total das quotas pagas de todas as propriedades de um determinado condomínio.

```
CREATE OR REPLACE FORCE VIEW "DESPESAS_SERVICOS" ("CODIGO", "TOTAL_PRECO") AS  
SELECT CODIGO, nvl(SUM(PRECO),0) AS TOTAL_PRECO  
FROM SERVICOREALIZADO  
GROUP BY CODIGO  
/
```

```
CREATE OR REPLACE FORCE VIEW "DESPESA_FUNC" ("CODIGO", "TOTAL_FUN") AS  
SELECT CONDOMINIO.CODIGO, nvl(SUM(FUNRECEBNOME.VALOR),0)  
FROM CONDOMINIO, TRABALHA, FUNCIONARIO, FUNRECEBNOME  
WHERE CONDOMINIO.CODIGO = TRABALHA.CODIGO AND FUNRECEBNOME.NIF = FUNCIONARIO.NIF AND FUNCIONARIO.NIF = TRABALHA.NIF  
GROUP BY CONDOMINIO.CODIGO  
/
```

```
CREATE OR REPLACE FORCE VIEW "RENDIMENTOTOTALPROP" ("CODIGO", "TOTAL_REVENUE") AS  
SELECT codigo, nvl(SUM(valor),0)  
FROM quotaspagas  
GROUP BY codigo  
/
```

Descrição interface

Página “Condomínios”:

Esta é a página principal dos condomínios. Existem três regiões Interactive Grid. Uma delas mostra os condomínios presentes na base de dados, uma mostra informações sobre o condomínio que pode ser selecionado no primeiro, e a última região mostra as propriedades que fazem parte do condomínio selecionado. Nesta última região existe um botão para adicionar quotas pagas de uma certa propriedade.

Página “Serviços”:

Esta é a página principal dos serviços. Aqui, temos uma Static Content denominada “Condomínios” que tem uma LOV dos condomínios associada. Ao escolher um condomínio, aparece num Interactive Report, os serviços que foram realizados. Aqui, podemos também adicionar estes serviços. Por fim, numa outra região, denominada “Total gasto”, que é um Classic Report, temos a informação sobre o dinheiro gasto em serviços.

Página “Empresas”:

Em “Empresas”, apenas temos um Interactive Report que mostra o nome e o número de telefone das empresas que realizam os serviços.

Página “Serviços Disponíveis”:

Nesta página, vemos os serviços disponíveis a serem utilizados por um condomínio. Existe a opção de criar estes serviços, com a restrição de que o nome da empresa tem de ser de uma empresa disponível na base de dados.

Página “Assembleias”:

Aqui podemos adicionar e editar assembleias. Existem dois Interactive Grid que estão implementados com um Master Detail, ou seja, selecionando um condomínio na região “Condomínios”, vão aparecer todas as assembleias que estão associadas a esse condomínio. Ao clicar no botão “Criar Assembleia”, podemos criar assembleias. Somos então redirecionados para a página “Adicionar Assembleia”.

Página “Assunto”:

Na página “Assunto” temos toda a informação sobre os assuntos discutidos nas assembleias, tal como o seu id, o seu estado e até mesmo o código de condomínio que está associado. Esta informação está num simples "Interactive Report". É possível criar um assunto, clicando no botão “Adicionar Assunto”.

Página “Adicionar Assembleia”:

Para adicionar uma assembleia, é necessário preencher a data e hora respetiva, e o código de condomínio.

Página “Pessoas”:

Nesta página, para além de um Interactive Report que mostra as pessoas que estão na base de dados, existem também quatro botões, onde dois deles servem para ver os funcionários e os moradores e, os outros dois, são para adicionar as respectivas pessoas (funcionários ou moradores) à base de dados.

Página “Moradores”:

Na página “Pessoas”, ao clicar no botão “Moradores” somos redirecionados para aqui. É visível as informações sobre os moradores, tal como o seu nif, data de entrada, data de saída (quando aplicável) ...etc

Página “Adicionar Morador”:

Na página “Pessoas”, ao clicar no botão “Adicionar Moradores” somos redirecionados para aqui. Ao preencher o formulário, é adicionado um novo morador na base de dados.

Página “Funcionários”:

Um Detalhe Condicional onde é nos apresentado os funcionários ativos que estão associados a um ou mais condomínios, escolhido através de uma LOV ligada a uma Select List. A partir desta página é também possível verificar os novos funcionários do condomínio, pois são aqueles que não têm recibos associados.

Página “Recibos de Vencimento”:

Um simples Report onde é apresentado os recibos associados aos funcionários. Estes recibos podem ser criados a partir do botão “Adicionar Recibo”.

Página “Adicionar Funcionário ao Condomínio”:

Depois do funcionário ser criado, ao entrar nesta página, é possível adicioná-lo a um novo condomínio, indicando o condomínio de inserção e o respetivo funcionário. Naturalmente, não vai ser possível adicioná-lo aos condomínios em que ele faz parte.

Página “Adicionar Funcionário”:

Form necessário para se adicionar um funcionário na base de dados.

Página “Funcionários ativos sem recibo”:

Um Report para consultar os funcionários ativos que não tem recibos de vencimento associados. São mostrados o seu NIF e o nome do funcionário.

Implementação das funcionalidades da interface APEX

- A página de entrada na nossa aplicação é o “Início” a partir da qual temos acesso a todas as páginas subsequentes.
- Na página dos “Funcionários Ativos” na coluna do “Nome do funcionário” aparece o nome dos funcionários em vez dos seus respetivos NIF.
- Na página dos “Condomínios” quando selecionamos um condomínio aparece o “Saldo Total” do condomínio que é um cálculo feito a partir das despesas resultantes dos funcionários e dos serviços realizados e o rendimento a partir das quotas.
- Na página das “Pessoas” é possível adicionar pessoas, a partir dos botões “Adicionar Funcionário” e “Adicionar Morador” que além de adicionar a tabela dos funcionários ou moradores também insere na tabela das pessoas. É possível também atualizar dados sobre a pessoa, como o número de telemóvel ou o email. Além disso, é possível apagar uma pessoa da base de dados.
- Quando inserimos um morador, temos uma select list para selecionar o condomínio da propriedade do morador. Usamos uma LOV para apresentar os endereços dos condomínios em vez dos seus respetivos códigos. A partir da escolha do condomínio são disponibilizados uma escolha restrita de valores para o andar e a porta dependente do condomínio escolhido para o morador. (eg. se escolhermos um condomínio com 5 andares, só vai ser permitido escolher no select list dos andares um valor entre 1 a 5. Este comportamento é idêntico no caso da seleção da porta.)
- Em todas as páginas existem breadcrumbs. Estes são apresentados de forma de fácil interpretação para uma navegação pela aplicação mais simples para o utilizador.
- Na página dos “Serviços” podemos encontrar dois reports interligados, onde ao escolher um condomínio temos informação detalhada sobre os serviços realizados sobre esse condomínio. A partir deste segundo report temos a opção de adicionar um serviço realizado.
- Nesta mesma página, ao escolher um condomínio também podemos encontrar o total do valor gasto em todos os serviços realizados para esse condomínio.

- Na página “Assembleias” podemos encontrar um form master-detail, onde ao escolher um condomínio temos acesso a todas as assembleias marcadas, a decorrer ou já terminadas. Na parte detail, podemos inserir, alterar e remover assembleias de um condomínio. Para facilitar a inserção de uma assembleia existe um botão para “Criar Assembleia”. Quando uma assembleia é marcada não podemos alterar a sua hora, nem data, mas sim o seu local, caso haver algum imprevisto sobre o local inicialmente definido.

Manual do utilizador

Este manual de utilizador foi desenvolvido para ajudar a compreender e a utilizar a nossa base de dados. A base de dados foi desenvolvida especialmente para gestores e/ou auxiliares de condomínios, fornecendo informações importantes, maximizando os recursos e funcionalidades e simplificando a gestão eficiente dos vários condomínios que os gestores possam ter na sua posse.

As entidades que se destacam na nossa base de dados são: Condomínio e Pessoa. A partir destas duas tabelas, conseguimos aceder às mais diversas informações sobre os condomínios. A base de dados guarda informações tais como: as várias assembleias e assuntos que os condomínios possam ter, todos os serviços e, dentro destes serviços, os serviços disponíveis e também as empresas que executam os serviços e também toda a informação sobre as pessoas, sejam elas moradores ou funcionários.

Entrar na base de dados "Gestão de Condomínios": User - UBD404 ; Password - 404

Entrando na base de dados, ao clicar na subsequência “Condomínios” do menu de navegação, vemos todos os condomínios existentes tal como todas as informações do mesmo. A partir desta página conseguimos também adicionar quotas pagas de uma certa propriedade. Dentro da subsequência anterior, temos outras duas páginas denominadas “Serviços” e “Assembleias”. Para adicionar serviços, primeiro o serviço tem de estar disponível. Para isso, clicando na seta para ver mais na subsequência “Serviços”, temos uma página “Serviços Disponíveis” onde podemos adicionar serviços disponíveis para serem realizados. Ao adicionar um serviço disponível, dentro de “Serviços” conseguimos adicionar serviços realizados e ver estes serviços por cada condomínio. Além disso, podemos ver as empresas disponíveis para a criação de serviços, em “Empresas”. Para adicionar assembleias, podemos, dentro de “Assembleias”, clicar no botão “Criar Assembleia” na região “Assembleias”. Ao

clicar para ver as restantes subsequências de “Assembleias”, podemos clicar em “Assuntos” para ver todos os assuntos discutidos, o seu estado e também criá-los.

Existe a opção de adicionar um Morador/Dono ou um Funcionário. Para isso, clicamos na página “Pessoas” e, dentro da mesma, temos 2 botões, um para adicionar Moradores e outro para adicionar funcionários. Nesta mesma página podemos também ver as pessoas existentes, no botão “Moradores”, os moradores e no botão “Funcionários”, os funcionários. Depois de um funcionário já pertencer à base de dados, pode ser adicionado a um certo condomínio seguindo esta sequência a partir do menu de navegação: “Pessoas”->“Funcionários”->“Adicionar Funcionários ao Condomínio”. Os funcionários vão ter recibos associados, e por isso, para os adicionar, segue-se a sequência “Pessoas”->“Funcionários”->“Adicionar Recibos”.

Alterações ao esquema da base de dados em relação à primeira entrega

- Eliminámos a entidade “ContaBancária” pois era desnecessário no nosso caso. Para calcular o saldo do condomínio, utilizamos as despesas dos serviços realizados, através da relação “DespesaS”, onde liga “Serviço Realizado” a “Condomínio” e as despesas dos funcionários (salários que recebem), através de “ReciboVencimento” que é uma entidade fraca de Funcionário e também da relação “Trabalha” que liga o “Funcionário” ao “Condomínio”. Em relação ao lucro, é obtido através das quotas pagas pelas propriedades.
- Adicionámos esta nova entidade “QuotasPagas” que é entidade fraca de “Propriedade” e assim o Condomínio tem informações das quotas pagas de cada propriedade.
- Foi adicionada uma nova entidade “Assunto” que caracteriza os assuntos discutidos na assembleia.
- Foram criadas 3 novas entidades que constituem os serviços. “Empresas”, “ServiçoDisponível” e “ServiçoRealizado”. “Empresa” é ligado a “ServiçoDisponível” através da relação “Fornece”. Esta ligação é importante para termos informações das empresas dos serviços da base de dados. Através da entidade dos serviços disponíveis, a entidade “ServiçoRealizado” diz-nos todos os serviços realizados em determinado condomínio, através da relação

“Exercido” que é ligada a “ServiçoDisponível” e da relação “DespesaS” que é ligado a “Condomínio”.

- Criou-se uma entidade denominada “ReciboVencimento” que é ligada a “Funcionário” através da relação “Recebe”. Esta nova entidade é importante pois é através dela que os funcionários recebem o seu salário e permite calcular qual os salários já pagos aos funcionários.