

Gestão de um Hotel

Relatório do Projeto de Base de Dados

Alexandre Peres, nº 61615
Filipe Santo, nº 58388

Turno P2
Grupo 21

Índice

| | |
|--|----|
| 1. Tema e objetivos da Base de Dados..... | 2 |
| 2. Modelo de Dados..... | 4 |
| 2.1. Modelo de Dados..... | 4 |
| 2.2. Passagem do Modelo ER para o Modelo Relacional..... | 5 |
| 2.3. Modelo Relacional..... | 6 |
| 2.4. Consultas interessantes para serem efetuadas..... | 7 |
| 2.5. Limitações e opções tomadas para a implementação da base de dados | 8 |
| 2.6. Criação da base de dados..... | 10 |
| 2.6.1. Criação de tabelas..... | 10 |
| 2.6.2. Criação de sequências..... | 13 |
| 2.6.3. Criação de triggers..... | 14 |
| 3. Implementação das consultas interessantes em SQL e no Aplicação APEX | 16 |
| 4. Manual de utilizador da Aplicação APEX..... | 19 |
| 5. Considerações finais..... | 29 |

1.Tema e objetivos da Base de Dados

O tema do nosso projeto é a implementação de uma base de dados para a **gestão de um Hotel**, sendo utilizada para facilitar a organização do mesmo. Este sistema tem como objetivo gerir as **pessoas do hotel (clientes e funcionários)**, as **reservas** e os **quartos** do hotel.

As **pessoas** do hotel podem ser **funcionários** ou **clientes** (funcionários podem também ser clientes), cada **pessoa** tem guardado na base de dados o seu nome, o contacto e o número de cartão de cidadão, sendo o número de cartão de cidadão a forma utilizada para distinguir diferentes pessoas pois estas podem ter nomes iguais. Se a pessoa for um **cliente** tem a si associado um **número de cliente**, que é um valor incrementado por cada novo cliente. Se a pessoa for um **funcionário** tem guardado na base de dados um número de identificação de funcionário, o seu salário por hora, os dias de trabalho por semana e as horas que entra e sai do trabalho.

Os **funcionários** estão também divididos em três categorias: **funcionários de balcão**, **funcionários de limpeza** e **outros funcionários**. Os **funcionários de balcão** são os que atendem os clientes ao balcão para marcar as reservas, os funcionários de limpeza são funcionários que limpam os quartos e os **outros funcionários** são os funcionários com menos relevância para a utilização da base dados, mas que também trabalham no hotel, por exemplo os Gerente do Hotel e os técnicos de manutenção (eletricistas, canalizadores, etc...). A única coisa da base de dados guardada especificamente sobre cada um dos outros funcionários é o nome da sua função.

As **reservas** podem ser feitas online ou no balcão, a única diferença entre elas é que as **online** têm um email, de quem fez a reserva, associado à mesma, e as de **balcão** têm um id do funcionário (de balcão), que tratou dessa reserva. Cada reserva tem também associado os dias de check-in e de check-out, a data em que a reserva foi feita, o número de quartos pretendidos, e o id do cliente que fez a reserva, cada reserva é identificada por um id único.

Cada **reserva** tem a si associado um **pagamento** único, a **relação entre o pagamento e a reserva** guarda a data em que o mesmo foi feito, e o **pagamento** guarda o valor que foi/vai ser pago, o tipo de pagamento se é por dinheiro, multibanco ou cartão de crédito, estado do pagamento, se foi pago ou não, e guarda o número do pagamento que é uma de identificar cada pagamento.

Os **quartos** do hotel têm um número de identificação associado a cada um, o andar onde se situa o quarto, o número do quarto nesse andar e a descrição do quarto.

Cada **quarto** tem um **tipo** (individual, duplo, de casal, triplo, quádruplo...) em relação ao número de camas que tem e à sua descrição. Os tipos de quartos têm um id único para ser mais fácil a sua identificação.

Um quarto pode ter diferentes preços dependendo do dia, por isso também pretendemos guardar esses dados.

É necessário saber o **estado de um quarto**, se ele está pronto para utilizar ou não, e a descrição desse estado, por exemplo pode haver alguma avaria no quarto, ou o quarto pode estar à espera da próxima limpeza para poder se encontrar pronto a utilizar.

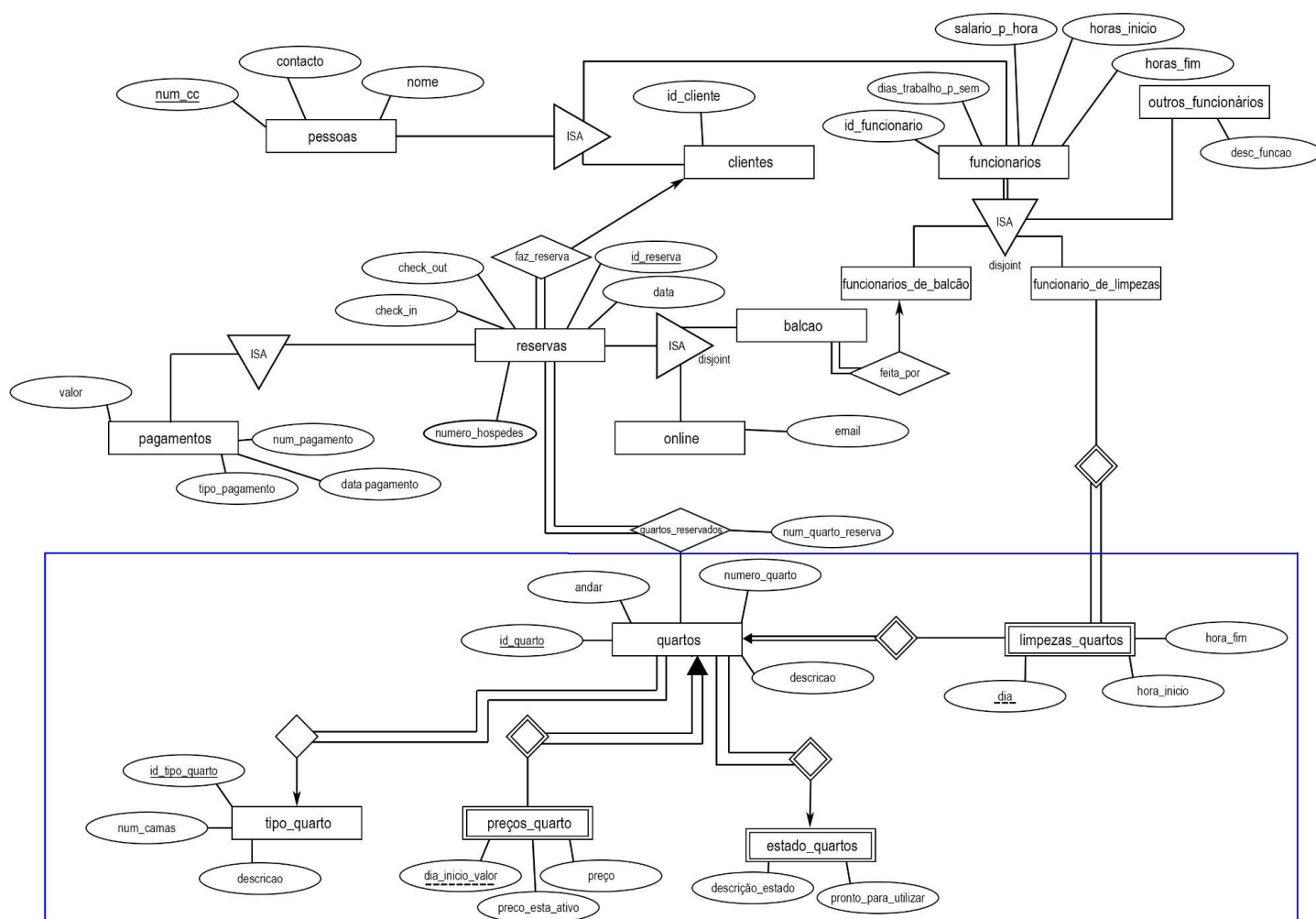
Cada **quarto** tem também associado as várias **limpezas** do mesmo durante o dia, por isso é guardado a hora do início e do fim de cada **limpeza** de cada quarto para todos os dias. Essa limpeza também guarda o/s número/s de identificação do/s funcionário/s de limpeza que a fizeram (visto que a limpeza pode ser feita por mais de uma pessoa ao mesmo tempo). Um funcionário só

pode estar a fazer uma limpeza dentro de um intervalo de tempo e esse funcionário não pode ter nenhuma limpeza fora do intervalo das suas horas de trabalho.

Uma **reserva** como pode ter vários **quartos reservados** precisa de uma forma de geri-los por isso existe uma **relação entre as reservas e os quartos**, e nessa relação existe também o número do quarto dentro da reserva por exemplo: uma reserva tem os quartos 242, 602 e 54, na relação de quartos reservados esses quartos são identificados respectivamente por 1,2,3.

2. Modelo de Dados

2.1. Modelo ER



2.2. Passagem do Modelo ER para o Modelo relacional

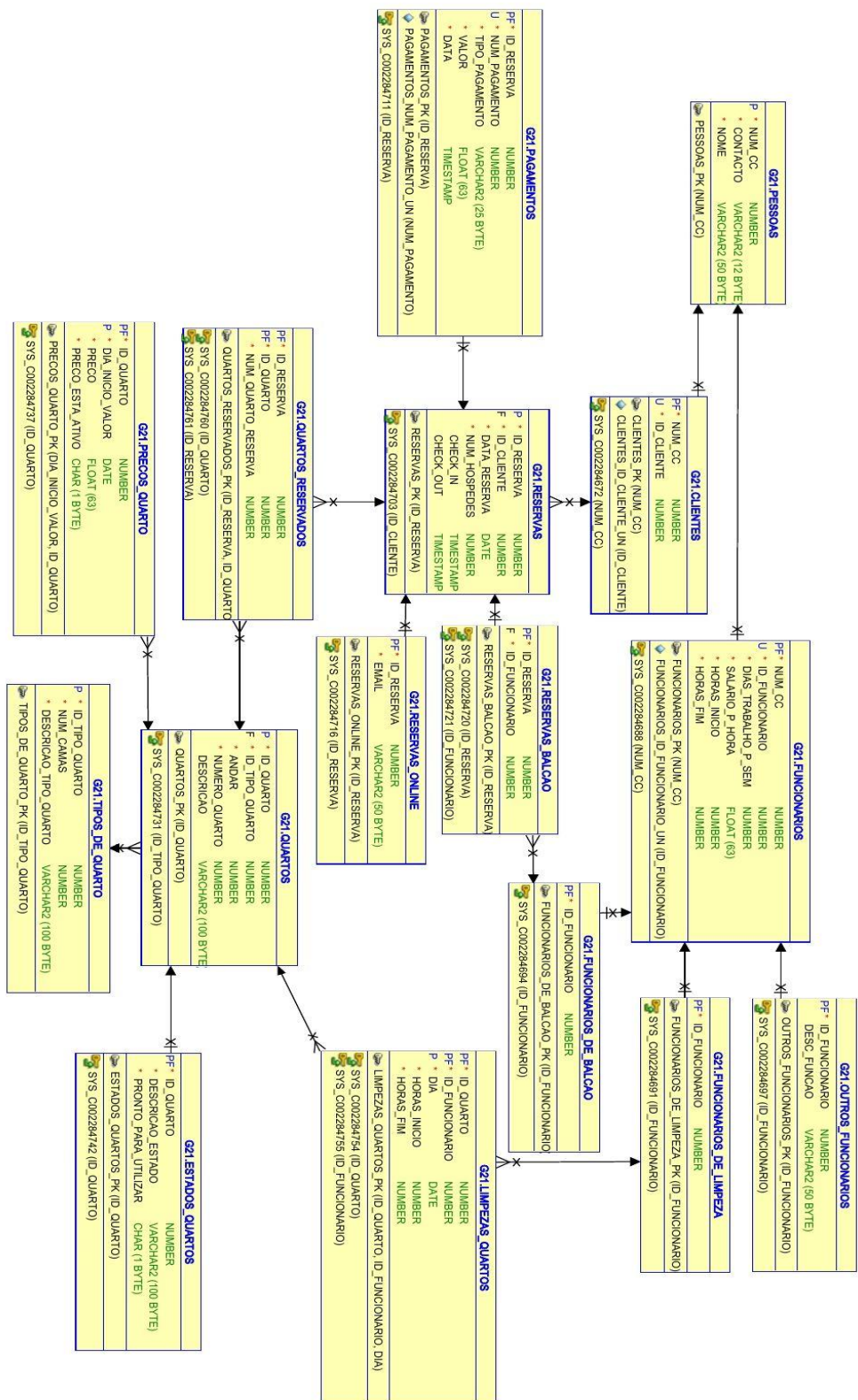
Entidades

- *peessoas* (num_cc, contacto, nome)
- *clientes* (num_cc, id_cliente)
 - num_cc é a chave estrangeira de *peessoas*
- *funcionarios* (num_cc, id_funcionario, dias_trabalho_p_semana, salario_p_hora, horas_inicio, horas_fim)
 - num_cc é a chave estrangeira de *peessoas*
- *funcionarios_de_limpeza* (id_funcionario)
 - id_funcionario é a chave estrangeira de *funcionarios*
- *funcionarios_de_balcao* (id_funcionario)
 - id_funcionario é a chave estrangeira de *funcionarios*
- *outros_funcionarios* (id_funcionario, desc_funcao)
 - id_funcionario é a chave estrangeira de *funcionarios*
- *reservas* (id_reserva, id_cliente, data_reserva, check_in, check_out, num_hospedes)
 - id_cliente é a chave estrangeira de *reservas*
- *pagamentos* (id_reserva, num_pagamento, estado, tipo_pagamento, valor)
 - id_reserva é a chave estrangeira de *reservas*
- *reserva_online* (id_reserva, email)
 - id_reserva é a chave estrangeira de *reservas*
- *reserva_balcao* (id_reserva, id_funcionario, email)
 - id_reserva é a chave estrangeira de *reservas*
 - id_funcionario é a chave estrangeira de *funcionarios_de_balcao*
- *quartos* (id_quarto, id_tipo_quarto, andar, numero_quarto, descricao)
 - id_tipo_quarto é a chave estrangeira de *tipo_quarto*
- *preços_quarto* (id_quarto, dia_inicio_valor, preço, preco_esta_ativo)
 - id_quarto é a chave estrangeira de *quartos*
 - dia_inicio_valor é chave parcial
- *estado_quarto* (id_quarto, descricao_estado, pronto_para_utilizar)
 - id_quarto é a chave estrangeira de *quartos*
- *tipo_quarto* (id_tipo_quarto, num_camadas, descricao_tipo_quarto)
- *limpezas_quarto* (id_quarto, id_funcionario, dia, hora_inicio, hora_fim)
 - id_quarto é a chave estrangeira de *quartos*
 - id_funcionario é a chave estrangeira de *funcionarios_de_limpeza*
 - dia é chave parcial

Relações

- *quartos_reservados* (id_reserva, id_quarto, num_quarto_reserva)
 - id_reserva é a chave estrangeira de *reservas*
 - id_quarto é a chave estrangeira de *quartos*

2.3. Modelo Relacional



2.4. Consultas Interessantes para serem efetuadas

1. Quantos funcionários estão empregados no hotel?
2. Quantas pessoas estão neste momento hospedados?
3. Quais os clientes que têm mais reservas no hotel?
4. Qual é a média de preço dos diferentes tipos de quartos?
5. Qual é o funcionário de balcão que atendeu mais clientes?
6. Quais são os clientes que também são funcionários?

2.5. Limitações e opções tomadas para a implementação da base de dados

Foi necessário criar algumas restrições para cada tabela existente da implementação da base de dados, para garantir o seu funcionamento e integridade.

Na tabela **peessoas** declaramos os atributos *num_cc*, *contacto* e *nome* como **NOT NULL**, por serem campos obrigatórios que em cada registo de cada pessoas é preciso ficarem guardados. A chave primária é *num_cc*.

Na tabela **clientes** declaramos os atributos *num_cc* e *id_cliente* como **NOT NULL**, e *id_cliente* como **UNIQUE**, por cada cliente tem de se guardar o seu número de CC, e no seu registo vai ser atribuído um *id_cliente* que só é usado para um cliente, não há repetição de *id_cliente*. A chave primária é *num_cc*.

Na tabela **funcionarios** declaramos os atributos *num_cc*, *id_funcionario*, *dias_trabalho_p_sem*, *salario_p_hora*, *horas_inicio* e *horas_fim* como **NOT NULL**, e *id_funcionario* como **UNIQUE**, por cada cliente ter de guardar o seu número de CC, e as suas respetivas informações de trabalho (o salário, horário etc), e no seu registo vai ser atribuído um *id_funcionario* que só é usado para um funcionário, não há repetição de *id_funcionario*, também usamos **CHECK** para termos a certeza que as horas do fim do trabalho são depois das horas do início do trabalho. A chave primária é *num_cc*.

Nas tabelas **funcionarios_de_limepza**, **outros_funcionarios** e **funcionarios_balcao** declaramos o atributo *id_funcionario* como **NOT NULL**, por ser um campo obrigatório que em cada registo de cada funcionários nas três listas diferentes é preciso ficar guardado. A chave primária de cada uma delas é *id_funcionario*.

Na tabelas **reservas** declaramos os atributos *id_reserva*, *id_cliente*, *data_reserva* e *num_hospedes* como **NOT NULL**, por serem campos obrigatórios no registo de cada reserva, já *check_in* e *check_out* são campos que podem estar vazios pois o dono da reserva pode ainda não ter entrado e/ou saído do hotel. A chave primária é *id_reserva*.

Na tabelas **pagamentos** declaramos os atributos *id_reserva*, *num_pagamento*, *tipo_pagamento*, *valor* e *data* como **NOT NULL**, por serem campos obrigatórios no pagamento de cada reserva, também declaramos *num_pagamento* como **UNIQUE** pois cada *num_pagamento* tem um valor único. A chave primária é *id_reserva*.

Na tabelas **reservas_online** e **reservas_balcao** declaramos em ambas os *id_reserva* como **NOT NULL**, por ser obrigatório para identificar a reserva.

Na **reservas_online** declaramos também o *email* como **NOT NULL**, por ser a identificação do utilizador que fez essa reserva, também é garantido que *email* é uma estrutura de email válida utilizando uma restrição com **REGEX**.

Na **reservas_balcao** declaramos também o *id_funcionario* como **NOT NULL**, por ser a identificação do funcionário que tratou desta reserva.

As chaves primários são *id_reserva*, em ambas as tabelas.

Na tabela **tipos_de_quarto** declaramos *id_tipo_quarto*, *num_camras* e *descricao_tipo_quarto* como **NOT NULL**, por serem campos obrigatórios em cada tipo de quarto. A chave primária é *id_tipo_quarto*.

Na tabela **quartos** declaramos *id_quarto*, *id_tipo_quarto*, *andar* e *numero_quarto* como **NOT NULL**, por serem campos obrigatórios em cada quarto. A chave primária é *id_quarto*.

Na tabela **precos_quarto** declaramos *id_quarto*, *dia_inicio_valor*, *preco* e *preco_esta_ativo* como **NOT NULL**, por serem campos obrigatórios em cada preço de um quarto. A chave primária é *id_quarto* e *dia_inicio_valor* é chave parcial.

Na tabela **estados_quartos** declaramos *id_quarto*, *descricao_estado* e *pronto_para_utilizar* como **NOT NULL**, por serem campos obrigatórios em cada estado de um quarto. A chave primária é *id_quarto*.

Na tabela **limpezas_quartos** declaramos *id_quarto*, *id_funcionario*, *dia*, *horas_inicio* e *horas_fim* como **NOT NULL**, por serem campos obrigatórios em cada limpeza de um quarto. As chaves primárias são *id_quarto*, *id_funcionários* e *dia* é chave parcial.

Na tabela **quartos_reservados** declaramos *id_quarto*, *num_quarto_reserva* e *id_reserva* como **NOT NULL**, por serem campos obrigatórios em cada um dos quartos reservados. As chaves primárias são *id_quarto* e *id_reserva*.

2.6. Criação da base de dados

A base de dados foi criada na versão 21.4.3.063.0100 do SQL Developer

2.6.1. Criação de tabelas

DROP TABLE pessoas **CASCADE CONSTRAINTS**;

```
CREATE TABLE pessoas (  
    num_cc NUMBER NOT NULL,  
    contacto VARCHAR(12) NOT NULL,  
    nome VARCHAR(50) NOT NULL,  
    PRIMARY KEY(num_cc)  
);
```

DROP TABLE clientes **CASCADE CONSTRAINTS**;

```
CREATE TABLE clientes (  
    num_cc NUMBER NOT NULL,  
    id_cliente NUMBER NOT NULL,  
    UNIQUE(id_cliente),  
    PRIMARY KEY(num_cc),  
    FOREIGN KEY(num_cc) REFERENCES pessoas ON DELETE CASCADE  
);
```

DROP TABLE funcionarios **CASCADE CONSTRAINTS**;

```
CREATE TABLE funcionarios (  
    num_cc NUMBER NOT NULL,  
    id_funcionario NUMBER NOT NULL,  
    dias_trabalho_p_sem NUMBER NOT NULL CHECK(dias_trabalho_p_sem>=0),  
    salario_p_hora REAL NOT NULL CHECK(salario_p_hora>=0),  
    horas_inicio NUMBER NOT NULL CHECK (horas_inicio >= 0) CHECK (horas_inicio <= 23),  
    horas_fim NUMBER NOT NULL CHECK (horas_fim >= 0) CHECK (horas_fim <= 23),  
    CONSTRAINT chk_hours CHECK(horas_inicio < horas_fim),  
    UNIQUE(id_funcionario),  
    PRIMARY KEY(num_cc),  
    FOREIGN KEY(num_cc) REFERENCES pessoas ON DELETE CASCADE  
);
```

DROP TABLE funcionarios_de_limpeza **CASCADE CONSTRAINTS**;

```
CREATE TABLE funcionarios_de_limpeza (  
    id_funcionario NUMBER NOT NULL,  
    PRIMARY KEY(id_funcionario),  
    FOREIGN KEY(id_funcionario) REFERENCES funcionarios(id_funcionario) ON DELETE CASCADE  
);
```

DROP TABLE funcionarios_de_balcao **CASCADE CONSTRAINTS**;

```
CREATE TABLE funcionarios_de_balcao (  
    id_funcionario NUMBER NOT NULL,  
    PRIMARY KEY(id_funcionario),  
    FOREIGN KEY(id_funcionario) REFERENCES funcionarios(id_funcionario) ON DELETE CASCADE  
);
```

DROP TABLE outros_funcionarios **CASCADE CONSTRAINTS**;

```
CREATE TABLE outros_funcionarios (  
    id_funcionario NUMBER NOT NULL,  
    desc_funcao VARCHAR(50),
```

```

PRIMARY KEY(id_funcionario),
FOREIGN KEY(id_funcionario) REFERENCES funcionarios(id_funcionario) ON DELETE CASCADE
);

```

```

DROP TABLE reservas CASCADE CONSTRAINTS;

```

```

CREATE TABLE reservas (
  id_reserva NUMBER NOT NULL,
  id_cliente NUMBER NOT NULL,
  data_reserva DATE NOT NULL,
  num_hospedes NUMBER NOT NULL,
  check_in TIMESTAMP ,
  check_out TIMESTAMP ,
  PRIMARY KEY(id_reserva),
  FOREIGN KEY(id_cliente) REFERENCES clientes(id_cliente) ON DELETE CASCADE
);

```

```

DROP TABLE pagamentos CASCADE CONSTRAINTS;

```

```

CREATE TABLE pagamentos (
  id_reserva NUMBER NOT NULL,
  num_pagamento NUMBER NOT NULL,
  tipo_pagamento VARCHAR(25) NOT NULL,
  valor REAL NOT NULL,
  data TIMESTAMP NOT NULL,
  UNIQUE(num_pagamento),
  PRIMARY KEY(id_reserva),
  FOREIGN KEY(id_reserva) REFERENCES reservas ON DELETE CASCADE
);

```

```

DROP TABLE reservas_online CASCADE CONSTRAINTS;

```

```

CREATE TABLE reservas_online (
  id_reserva NUMBER NOT NULL,
  email VARCHAR(50) NOT NULL CHECK (
    REGEXP_LIKE ( email, '[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,4}$' )
  ),
  PRIMARY KEY(id_reserva),
  FOREIGN KEY(id_reserva) REFERENCES reservas ON DELETE CASCADE
);

```

```

DROP TABLE reservas_balcao CASCADE CONSTRAINTS;

```

```

CREATE TABLE reservas_balcao (
  id_reserva NUMBER NOT NULL,
  id_funcionario NUMBER NOT NULL,
  PRIMARY KEY(id_reserva),
  FOREIGN KEY(id_reserva) REFERENCES reservas ON DELETE CASCADE,
  FOREIGN KEY(idDcos__funcionario) REFERENCES funcionarios_de_balcao ON DELETE CASCADE
);

```

```

DROP TABLE tipos_de_quarto CASCADE CONSTRAINTS;

```

```

CREATE TABLE tipos_de_quarto(
  id_tipo_quarto NUMBER NOT NULL,
  num_camras NUMBER NOT NULL,
  descricao_tipo_quarto VARCHAR(100) NOT NULL,
  PRIMARY KEY(id_tipo_quarto)
);

```

```

DROP TABLE quartos CASCADE CONSTRAINTS;
CREATE TABLE quartos (
    id_quarto NUMBER NOT NULL,
    id_tipo_quarto NUMBER NOT NULL,
    andar NUMBER NOT NULL,
    numero_quarto NUMBER NOT NULL,
    descricao VARCHAR(100),
    PRIMARY KEY(id_quarto),
    FOREIGN KEY(id_tipo_quarto) REFERENCES tipos_de_quarto ON DELETE CASCADE
);

```

```

DROP TABLE precos_quarto CASCADE CONSTRAINTS;
CREATE TABLE precos_quarto(
    id_quarto NUMBER NOT NULL,
    dia_inicio_valor DATE NOT NULL,
    preco REAL NOT NULL,
    preco_esta_ativo CHAR(1) NOT NULL, --Y if yes N if no
    PRIMARY KEY(dia_inicio_valor,id_quarto),
    FOREIGN KEY(id_quarto) REFERENCES quartos ON DELETE CASCADE
);

```

```

DROP TABLE estados_quartos CASCADE CONSTRAINTS;
CREATE TABLE estados_quartos(
    id_quarto NUMBER NOT NULL,
    descricao_estado VARCHAR(100) NOT NULL,
    pronto_para_utilizar CHAR(1) NOT NULL, --Y if yes N if no
    PRIMARY KEY(id_quarto),
    FOREIGN KEY(id_quarto) REFERENCES quartos ON DELETE CASCADE
);

```

```

DROP TABLE limpezas_quartos CASCADE CONSTRAINTS;
CREATE TABLE limpezas_quartos(
    id_quarto NUMBER NOT NULL,
    id_funcionario NUMBER NOT NULL,
    dia DATE NOT NULL,
    horas_inicio NUMBER NOT NULL CHECK (horas_inicio >= 0) CHECK (horas_inicio <= 23),
    horas_fim NUMBER NOT NULL CHECK (horas_fim >= 0) CHECK (horas_fim <= 23),
    CONSTRAINT chk_hours_limpeza CHECK(horas_inicio < horas_fim),
    PRIMARY KEY (id_quarto,id_funcionario,dia),
    FOREIGN KEY(id_quarto) REFERENCES quartos ON DELETE CASCADE,
    FOREIGN KEY(id_funcionario) REFERENCES funcionarios_de_limpeza ON DELETE CASCADE
);

```

```

DROP TABLE quartos_reservados CASCADE CONSTRAINTS;
CREATE TABLE quartos_reservados(
    id_reserva NUMBER NOT NULL,
    id_quarto NUMBER NOT NULL,
    num_quarto_reserva NUMBER NOT NULL,
    PRIMARY KEY(id_reserva,id_quarto),
    FOREIGN KEY(id_quarto) REFERENCES quartos ON DELETE CASCADE,
    FOREIGN KEY(id_reserva) REFERENCES reservas ON DELETE CASCADE
);

```

2.6.2. Criação de sequências

As *Sequences* que criámos são utilizadas para dar valores únicos e incrementais às Chaves Primárias das seguintes tabelas **quartos**, **reservas**, **tipos_de_quarto** (nos atributos id_quarto, id_reserva, id_tipo_quarto respetivamente) também para atributos únicos de outras tabelas, clientes, funcionarios e pagamentos (nos atributos id_cliente, id_funcionario, num_pagamento respetivamente):

```
DROP SEQUENCE cliente_seq;  
CREATE SEQUENCE cliente_seq;
```

```
DROP SEQUENCE quartos_seq;  
CREATE SEQUENCE quartos_seq;
```

```
DROP SEQUENCE funcionarios_seq;  
CREATE SEQUENCE funcionarios_seq;
```

```
DROP SEQUENCE reservas_seq;  
CREATE SEQUENCE reservas_seq;
```

```
DROP SEQUENCE pagamentos_seq;  
CREATE SEQUENCE pagamentos_seq;
```

```
DROP SEQUENCE tipo_quarto_seq;  
CREATE SEQUENCE tipo_quarto_seq;
```

2.6.3. Criação de triggers

O *Trigger cliente_bir* é usado no momento de inserção de um cliente para o id_cliente ser incrementado em cada inserção.

```
CREATE OR REPLACE TRIGGER cliente_bir
BEFORE INSERT ON clientes
FOR EACH ROW
BEGIN
    SELECT cliente_seq.NEXTVAL
    INTO :new.id_cliente
    FROM dual;
END;
/
```

O *Trigger funcionarios_bir* é usado no momento de inserção de um funcionário para o id_funcionario ser incrementado em cada inserção.

```
CREATE OR REPLACE TRIGGER funcionarios_bir
BEFORE INSERT ON funcionarios
FOR EACH ROW
BEGIN
    SELECT funcionarios_seq.NEXTVAL
    INTO :new.id_funcionario
    FROM dual;
END;
/
```

O *Trigger reservas_bir* é usado no momento de inserção de uma reserva para o id_reserva ser incrementado em cada inserção.

```
CREATE OR REPLACE TRIGGER reservas_bir
BEFORE INSERT ON reservas
FOR EACH ROW
BEGIN
    SELECT reservas_seq.NEXTVAL
    INTO :new.id_reserva
    FROM dual;
END;
/
```

O *Trigger pagamentos_bir* é usado no momento de inserção de um pagamento para o num_pagamento ser incrementado em cada inserção.

```
CREATE OR REPLACE TRIGGER pagamentos_bir
BEFORE INSERT ON pagamentos
FOR EACH ROW
BEGIN
    SELECT pagamentos_seq.NEXTVAL
    INTO :new.num_pagamento
    FROM dual;
END;
/
```

O *Trigger* tipo_quarto_bir é usado no momento de inserção de um tipo de quarto para o id_tipo_quarto ser incrementado em cada inserção.

```
CREATE OR REPLACE TRIGGER tipo_quarto_bir  
BEFORE INSERT ON tipos_de_quarto  
FOR EACH ROW  
BEGIN  
    SELECT tipo_quarto_seq.NEXTVAL  
    INTO :new.id_tipo_quarto  
    FROM dual;  
END;  
/
```

O *Trigger* quartos_bir é usado no momento de inserção de um quarto para o id_quarto ser incrementado em cada inserção.

```
CREATE OR REPLACE TRIGGER quartos_bir  
BEFORE INSERT ON quartos  
FOR EACH ROW  
BEGIN  
    SELECT quartos_seq.NEXTVAL  
    INTO :new.id_quarto  
    FROM dual;  
END;  
/
```


3.Implementação das consultas interessantes em SQL e no Aplicação APEX

1. Quantos funcionários estão empregados no hotel?

SQL:

```
SELECT COUNT(id_funcionario)  
FROM funcionarios;
```

Apex:



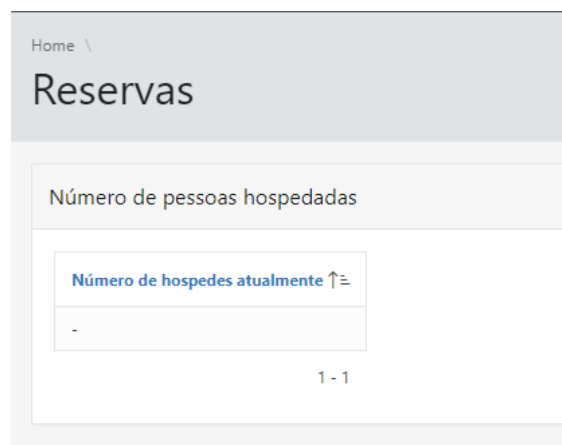
| |
|----------------------|
| Home \ |
| Funcionários |
| Total Funcionários |
| Total Funcionários ↑ |
| 10 |
| 1 - 1 |

2. Quantas pessoas estão neste momento hospedados?

SQL:

```
SELECT SUM(num_hospedes)  
FROM reservas  
WHERE check_in IS NOT NULL AND check_out IS NULL;
```

Apex:



| |
|---------------------------------|
| Home \ |
| Reservas |
| Número de pessoas hospedadas |
| Número de hospedes atualmente ↑ |
| - |
| 1 - 1 |

3. Quais os clientes que têm mais reservas no hotel?

SQL:

```
SELECT nome,COUNT(id_reserva) FROM reservas
INNER JOIN clientes USING(id_cliente)
INNER JOIN pessoas USING(num_cc)
GROUP BY nome
ORDER BY COUNT(id_reserva) DESC;
```

Apex:

Home \

TOP Clientes Hotel

Top Clientes Hotel

| Nome | Número de estadias |
|------------------|--------------------|
| Manuela da Cunha | 1 |
| Igor dias | 1 |
| Esther da Mota | 1 |
| Valetina da Mota | 1 |

[Download](#)

1 - 4

4. Qual é a média de preço dos diferentes tipos de quartos?

SQL:

```
SELECT descricao_tipo_quarto,AVG(preco) FROM precos_quarto
INNER JOIN quartos USING(id_quarto)
INNER JOIN tipos_de_quarto USING (id_tipo_quarto)
GROUP BY descricao_tipo_quarto
WHERE preco_esta_ativo = 'Y'
;
```

Apex:

Tipos de Quarto

Tipo quarto

Média preço

Média de preço: ↑=

29

1 - 1

5. Qual é o funcionário de balcão que atendeu mais clientes?

SQL:

```
SELECT nome,COUNT(id_reserva) FROM reservas_balcao
INNER JOIN funcionarios USING(id_funcionario)
INNER JOIN pessoas USING(num_cc)
GROUP BY nome
;
```

Apex:

| TOP Funcionários que atenderam mais clientes | |
|--|------------------------------|
| Nome ↑ | Número de clientes atendidos |
| Stella Carvalho | 1 |
| 1 - 1 | |

6. Quais são os clientes que também são funcionários?

SQL:

```
SELECT * FROM
  (SELECT num_cc FROM clientes
   INTERSECT
   SELECT num_cc FROM funcionarios)
INNER JOIN pessoas USING(num_cc)
;
```

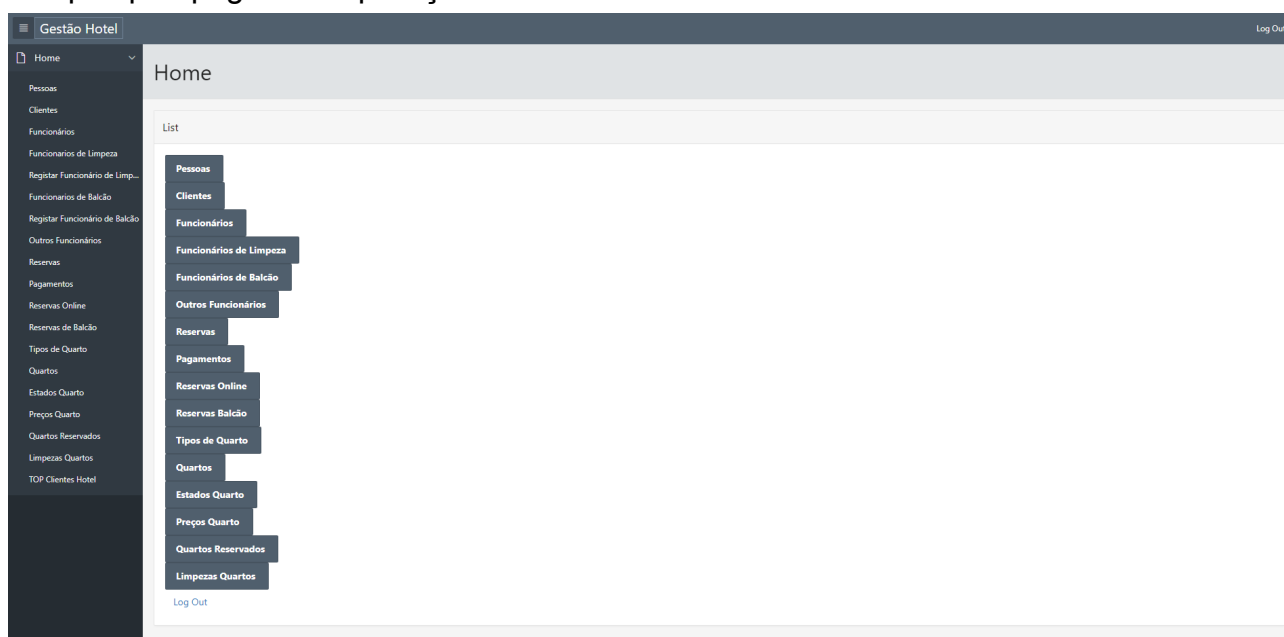
Apex:

| Clientes que são funcionários | | |
|-------------------------------|--------------|--------------------|
| Num cc ↑ | Contacto | Nome |
| 43534543 | 351976543246 | Benjamin Gonçalves |
| 1 - 1 | | |

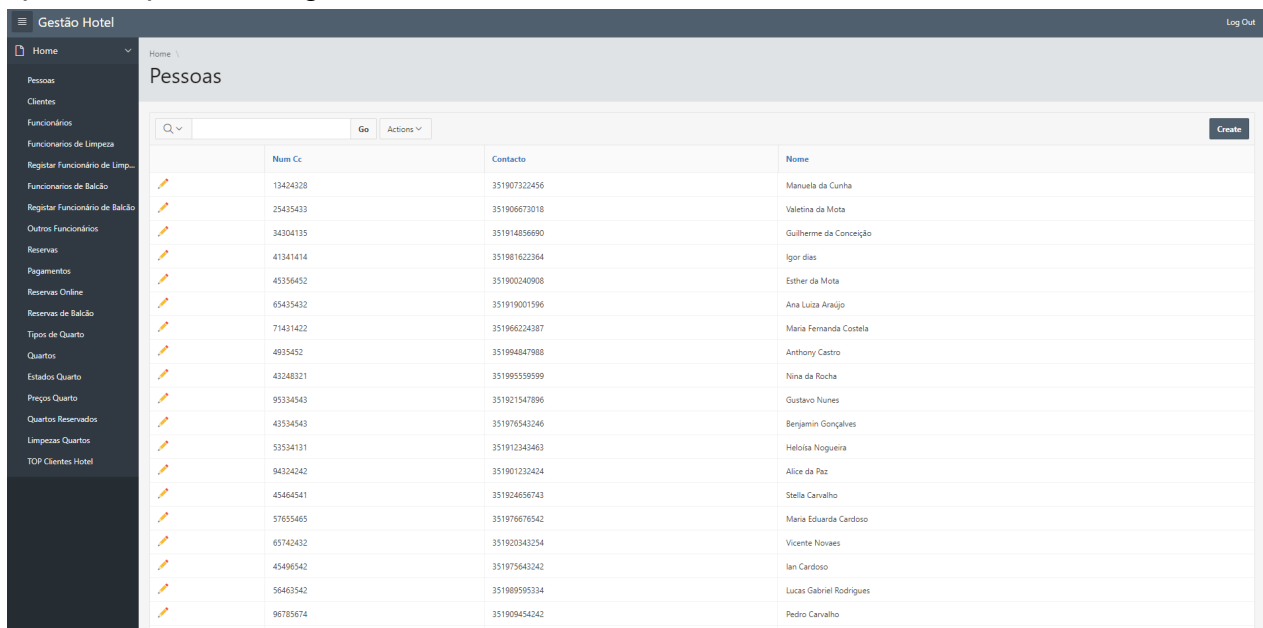
4.Manual de utilizador da Aplicação APEX

Foi criada uma interface gráfica utilizando a ferramenta APEX para uma melhor utilização da base de dados para utilizadores com pouco ou mesmo nenhum conhecimento em SQL.

Quando se entra na aplicação a pagina inicial onde entramos é esta, que lista as outras páginas da aplicação, tendo capacidade de navegar por elas, também é possível navegar pelas páginas usando a lista de páginas à esquerda que está sempre presente em qualquer página da aplicação.

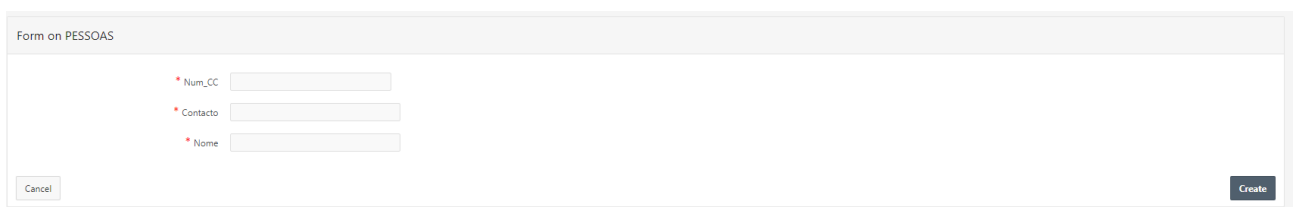
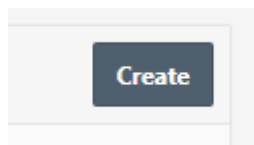


Ao clicar em **Pessoas** somos redirecionados para esta página que mostra um report das pessoas registadas na base de dados.



| | Num Cc | Contacto | Nome |
|--|----------|--------------|-------------------------|
| | 13424328 | 351907322456 | Manuela da Cunha |
| | 25435433 | 351906673018 | Valetina da Mota |
| | 34304135 | 351914856690 | Guilherme da Conceição |
| | 41341414 | 351981622364 | Igor dias |
| | 45356452 | 351900340908 | Esther da Mota |
| | 65435432 | 351919001596 | Ana Luiza Araújo |
| | 71431422 | 351966324387 | Maria Fernanda Costela |
| | 48935452 | 351994847988 | Anthony Castro |
| | 43248321 | 351995559599 | Nina da Rocha |
| | 93334543 | 351921547896 | Gustavo Nunes |
| | 43534543 | 351976543246 | Benjamin Gonçalves |
| | 53534131 | 351912343463 | Heloisa Nogueira |
| | 94324242 | 351901232424 | Alice da Paz |
| | 45464541 | 351924656743 | Stella Carvalho |
| | 57655465 | 351976676542 | Maria Eduarda Cardoso |
| | 65742432 | 351920343254 | Vicente Noveas |
| | 45496542 | 351975643242 | Ian Cardoso |
| | 56463542 | 351989959534 | Lucas Gabriel Rodrigues |
| | 96785674 | 351909454242 | Pedro Carvalho |

Se clicarmos neste botão, tanto nesta como em outras páginas temos a possibilidade de adicionar novos dados utilizando um formulário como o da imagem (no manual de utilizador não vai ser mostrado todas as páginas de create, pois são bastante semelhantes entre si, e têm basicamente a mesma funcionalidade que é adicionar novos dados à tabela).



Form on PESSOAS

* Num_CC

* Contacto

* Nome

Cancel Create

Ao clicar em **Clientes** somos redirecionados para esta página que mostra um report dos clientes registados na base de dados. Esta página tem também uma opção de create e também um report (não atualizável) de clientes que também são funcionários do hotel.

Home \

Cientes

Cientes que são funcionários

| Num cc ↑↓ | Contacto | Nome |
|-----------|--------------|--------------------|
| 43534543 | 351976543246 | Benjamin Gonçalves |

1 - 1

Q ▾

Go

Actions ▾

Create

| | Num Cc | Nome |
|--|----------|------------------------|
| | 13424328 | Manuela da Cunha |
| | 25435433 | Valetina da Mota |
| | 34304135 | Guilherme da Conceição |
| | 41341414 | Igor dias |
| | 45356452 | Esther da Mota |
| | 71431422 | Maria Fernanda Costela |
| | 4935452 | Anthony Castro |
| | 43248321 | Nina da Rocha |
| | 95334543 | Gustavo Nunes |
| | 43534543 | Benjamin Gonçalves |

1 - 10

Ao clicar em **Funcionários** somos redirecionados para esta página que mostra um report dos funcionarios registados na base de dados. Esta página tem também uma opção de create e tem no topo o número de total de funcionários do hotel.

Home \

Funcionários

Total Funcionários

Total Funcionários ↑↓

11

1 - 1

Q ▾

Go

Actions ▾

Create

| | Id Funcionario | Nome | Número cc | Dias trabalho p/sem | Salário p/hora | Horas início do trabalho | Horas fim do trabalho | Salário p/dia |
|--|----------------|-------------------------|-----------|---------------------|----------------|--------------------------|-----------------------|---------------|
| | 21 | Benjamin Gonçalves | 43534543 | 3 | 12 | 10 | 13 | 36 |
| | 4 | Stella Carvalho | 45464541 | 4 | 20 | 7 | 20 | 260 |
| | 7 | Ian Cardoso | 45496542 | 7 | 20 | 9 | 20 | 220 |
| | 10 | Mariane Ferreira | 46353453 | 7 | 13 | 11 | 21 | 130 |
| | 1 | Heloisa Nogueira | 53534131 | 5 | 15 | 7 | 16 | 135 |
| | 8 | Lucas Gabriel Rodrigues | 56463542 | 7 | 20 | 14 | 16 | 40 |
| | 5 | Maria Eduarda Cardoso | 57655465 | 5 | 20 | 17 | 23 | 120 |
| | 6 | Vicente Novaes | 65742432 | 3 | 15 | 10 | 18 | 120 |
| | 3 | Rebeca Pinto | 69545435 | 4 | 13 | 8 | 14 | 78 |
| | 2 | Alice da Paz | 94324242 | 5 | 17 | 8 | 18 | 170 |
| | 9 | Pedro Carvalho | 96785674 | 7 | 20 | 15 | 19 | 80 |

1 - 11

Também é possível aceder **Funcionarios de Balcão** e **Funcionarios de Limpeza**, que mostra um report de cada uma destas tabelas de funcionários, o primeiro destes mostra também a Top de funcionários que já atenderam mais clientes. As páginas com formulários para inserir novos dados não estão disponíveis dentro das próprias páginas mas estão no menu à esquerda e são chamadas **Registar Funcionário de Balcão** e **Registar Funcionário de Limpeza** respetivamente.

Home \ Funcionários \

Funcionarios de Balcão

| TOP Funcionários que atenderam mais clientes | |
|--|------------------------------|
| Nome ↑ | Número de clientes atendidos |
| Stella Carvalho | 1 |
| 1 - 1 | |

| Q ▾ Go Actions ▾ | | |
|------------------|----------|-----------------------|
| Id Funcionario | Num cc | Nome |
| 4 | 45464541 | Stella Carvalho |
| 5 | 57655465 | Maria Eduarda Cardoso |
| 6 | 65742432 | Vicente Novees |
| 1 - 3 | | |

Home \ Funcionários \




Funcionarios de Limpeza

| Q ▾ Go Actions ▾ | | |
|------------------|----------|------------------|
| Id Funcionario | Num cc | Nome |
| 1 | 53534131 | Heloisa Nogueira |
| 2 | 94324242 | Alice da Paz |
| 3 | 69545435 | Rebeca Pinto |
| 1 - 3 | | |

Existe também a página de **Outros Funcionários** também com uma opção de create para adicionar novos dados.

Home \ Funcionários \

Outros Funcionários

| Q ▾ Go Actions ▾ Create | | | | |
|---|----------------|-------------|----------|-------------------------|
| | Id Funcionario | Desc Funcao | Num cc | Nome |
|  | 7 | Diretor | 45496542 | Ian Cardoso |
|  | 8 | Canalizador | 56463542 | Lucas Gabriel Rodrigues |
|  | 9 | Eletricista | 96785674 | Pedro Carvalho |
| 1 - 3 | | | | |

Temos a página de **Reservas** que tem um report de todas as reservas, também no topo tem o número de hóspedes atualmente no hotel.

Home \ Reservas

Número de pessoas hospedadas

Número de hóspedes atualmente ↑

1 - 1

Q Go Actions Create

| | Id Reserva | Data Reserva | Num Hospedes | Check In | Check Out | Nome do cliente |
|--|------------|--------------|--------------|------------------------------|------------------------------|------------------|
| | 1 | 31-MAY-22 | 2 | - | - | Manuela da Cunha |
| | 2 | 25-MAY-22 | 1 | - | - | Esther da Mota |
| | 3 | 25-MAY-22 | 2 | - | - | Valetina da Mota |
| | 4 | 20-FEB-22 | 2 | 22-FEB-22 04:24:00.000000 PM | 26-FEB-22 11:10:00.000000 AM | Igor dias |

1 - 4

Esta também tem uma opção create que redireciona para este formulário para adicionar novos dados, e dentro deste formulário em vez de ser necessário adicionar o ID de um cliente para a reserva é só preciso escolher pelo o nome de clientes já registados, como mostra a segunda imagem

Form on RESERVAS

* Identificação

* Data Reserva

* Num Hospedes

Check In

Check Out

Cancel Create

* Identificação

* Data Reserva

* Num Hospedes

Check In

Check Out

- Anthony Castro
- Benjamin Gonçalves
- Esther da Mota
- Guilherme da Conceição
- Gustavo Nunes
- Igor dias
- Manuela da Cunha
- Maria Fernanda Costela
- Nina da Rocha
- Valetina da Mota

Existe a página **Pagamentos** que mostra o report de todos os pagamentos feitos, tem uma opção create para ser possível adicionar novos dados.

Home \ Pagamentos

Q Go Actions Create

| | Id Reserva | Num Pagamento | Tipo Pagamento | Valor | Data |
|--|------------|---------------|----------------|-------|------------------------------|
| | 4 | 1 | Multibanco | 55 | 26-FEB-22 11:30:00.000000 AM |


1 - 1

Existem as páginas **Reservas Online** e **Reservas de Balcão** que são bastante parecidas só alteram as colunas dos reports para tipo de reserva, tendo ambas a possibilidade de criar novos dados.

Home \ Reservas \

Reservas Online

Q ▾ Go Actions ▾ Create


| | Id Reserva | Email | Nome cliente |
|---|------------|------------------|------------------|
|  | 3 | abc123@gmail.com | Valetina da Mota |

1 - 1

Home \ Reservas \

Reservas de Balcão

Q ▾ Go Actions ▾ Create

| | Id Reserva | Nome do funcionário |
|---|------------|---------------------|
|  | 2 | Stella Carvalho |

1 - 1

Na criação de dados para **Reservas de Balcão** o utilizador pode escolher o funcionário a partir de lista em vez de inserir o Id do funcionário.

Home \ Reservas \ Reservas de Balcão \

Registar Reserva de Balcão

Form on RESERVAS_BALCAO

* ID Reserva

* Nome Funcionário

▾

Stella Carvalho

Maria Eduarda Cardoso

Vicente Novaes

Cancel

Temos o report de **Quartos** onde é possível escolher o tipo de quarto que apenas queremos visualizar, mostra também a média de preços desse tipo de quarto, havendo possibilidade de criar novos quartos.

| Quartos | | | | | |
|---|-----------|-------|---------------|-----------------------------|--|
| Tipos de Quarto | | | | | |
| Tipo quarto Quarto com cama de casal e uma casa de banho | | | | | |
| Média preço | | | | | |
| <div>Média de preços: ↑ %</div> <div>29</div> <div>1 - 1</div> | | | | | |
| <div><input type="text"/> <input type="button" value="Go"/> <input type="button" value="Actions"/></div> <div><input type="button" value="Create"/></div> | | | | | |
| | Id Quarto | Andar | Numero Quarto | Descrição | Descrição tipo quarto |
| | 1 | 2 | 25 | Quarto adequado para casais | Quarto com cama de casal e uma casa de banho |
| | 2 | 2 | 27 | Quarto adequado para casais | Quarto com cama de casal e uma casa de banho |
| | 3 | 3 | 35 | Quarto adequado para casais | Quarto com cama de casal e uma casa de banho |
| 1 - 3 | | | | | |

No formulários de criação de novos quartos podemos logo escolher a partir da lista de tipos de quarto em vez de inserirmos o id do tipo de quarto.

Form on QUARTOS

* Tipo de Quarto

* Andar

* Numero Quarto

Descricao

Quarto com cama de casal e uma cama individual , e com duas casas de banho

Quarto com cama de casal e uma casa de banho

Quarto com cama individual e uma casa de banho

Quarto com duas camas individuais, uma mini-cozinha e uma casa de banho

Cancel

Temos o report de **Tipos de Quartos** onde mostra todos os tipos de quartos e há a possibilidade de adicionar novos tipos de quartos. Também existe um link na descrição de tipo de quarto que redireciona para o report de **Quartos** para um tipo de quarto específico.

Home \

Tipos de Quarto

Q ▾ Go Actions ▾ Create

| | Id Tipo Quarto | Num Camas | Descricao Tipo Quarto |
|---|----------------|-----------|---|
|  | 1 | 2 | Quarto com duas camas individuais, uma mini-cozinha e uma casa de banho |
|  | 2 | 1 | Quarto com cama de casal e uma casa de banho |
|  | 3 | 1 | Quarto com cama individual e uma casa de banho |
|  | 4 | 2 | Quarto com cama de casal e uma cama individual, e com duas casas de banho |










1 - 4

Existem os reports dos **Preços do Quarto** e **Estados do Quarto**, onde mostra os dados de cada uma dessas tabelas havendo a possibilidade de alterar ou adicionar novos valores.

Quartos \

Estados Quarto

Q ▾ Go Actions ▾ Create











| | Id Quarto | Descricao Estado | Pronto Para Utilizar |
|---|-----------|-----------------------------|----------------------|
|  | 1 | Ocupado | N |
|  | 2 | Casa de banho com problemas | N |
|  | 3 | À espera de limpeza | N |
|  | 4 | Limpo | Y |
|  | 5 | Limpo | Y |
|  | 6 | Limpo | Y |
|  | 7 | Limpo | Y |
|  | 8 | Limpo | Y |
|  | 9 | Limpo | Y |

1 - 9

Quartos \

Preços Quarto

Q ▾ Go Actions ▾ Create

| | Id Quarto | Data Inicio Valor | Preco | Preco Esta Ativo |
|---|-----------|-------------------|-------|------------------|
|  | 1 | 25-MAY-22 | 25 | Y |
|  | 1 | 12-FEB-22 | 30 | N |
|  | 2 | 14-APR-22 | 30 | Y |
|  | 3 | 15-APR-22 | 32 | Y |
|  | 4 | 14-APR-22 | 30 | Y |
|  | 5 | 14-APR-22 | 30 | Y |
|  | 6 | 29-NOV-20 | 22 | Y |
|  | 7 | 14-APR-22 | 30 | Y |
|  | 8 | 01-APR-21 | 30 | Y |
|  | 9 | 01-APR-21 | 30 | Y |

1 - 10

Temos o report de **Quartos Reservados** onde mostra para cada reserva qual o/os quarto/s que essa reserva tem, havendo também possibilidade de adicionar novos dados com o create.

Home \ Reservas \

Quartos Reservados

Q v Go Actions Create

| | Id Reserva | Id Quarto | Num Quarto Reserva | Nome | Andar | Numero quarto |
|--|------------|-----------|--------------------|-----------|-------|---------------|
| | 4 | 6 | 1 | Igor dias | 2 | 27 |
| | 4 | 7 | 2 | Igor dias | 2 | 29 |

1 - 2

Existe também um report com mais detalhes sobre as **Limpezas de Quartos** em que mostrada cada limpeza de cada quarto , nome do funcionário que a vai fazer e também as horas em que vai decorrer.

Home \

Limpezas Quartos

Limpezas Quartos Create

Q v Search: All Text Columns Go Actions Reset

| | Id Quarto | Id Funcionario | Dia | Horas Inicio | Horas Fim | Nome |
|--|-----------|----------------|-----------|--------------|-----------|------------------|
| | 1 | 1 | 14-APR-22 | 10 | 11 | Heloisa Nogueira |
| | 1 | 2 | 15-APR-22 | 9 | 10 | Alice da Paz |
| | 1 | 3 | 17-APR-22 | 10 | 11 | Rebeca Pinto |
| | 2 | 3 | 24-MAY-22 | 11 | 12 | Rebeca Pinto |
| | 3 | 2 | 24-MAY-22 | 11 | 12 | Alice da Paz |
| | 1 | 1 | 02-JUN-22 | 12 | 14 | Heloisa Nogueira |

1 rows selected Total 6

Há também a possibilidade de criar novas limpezas, e para adicionar na tabela da base de dados em vez de se usar o ID do funcionário escolhe-se o nome do funcionário desta lista com os nomes de todos os funcionários de limpeza.

Home \ Limpezas Quartos \

Inserir Limpeza de Quarto

Inserir Limpeza de Quarto

* ID Quarto

Funcionário

Dia

* Horas Inicio

* Horas Fim

Cancel

Por fim existe também a página TOP Clientes Hotel que mostra o ranking de pessoas com mais estadias no hotel.

Home \

TOP Clientes Hotel

Top Clientes Hotel

| Nome | Número de estadias |
|------------------|--------------------|
| Manuela da Cunha | 1 |
| Igor dias | 1 |
| Esther da Mota | 1 |
| Valetina da Mota | 1 |

Download

1 - 4

5. Considerações Finais

Consideramos ter implementado uma base de dados, apesar de básica e com algumas imperfeições, o que é suposto uma base de dados de um hotel ter.

O trabalho foi bastante educativo, pois deu-nos a possibilidade de perceber como funciona a criação de uma base de dados completa e como é possível geri-la.

Houve alguns percalços durante o trabalho, um deles foi que ficamos com menos um membro no grupo e por isso o projeto ficou mais básico do que inicialmente imaginamos e propusemos, mas pensamos que as partes mais importantes foram feitas com sucesso.

A utilização do SQLDeveloper e do APEX , apesar de um pouco complicadas de aprender a manusear ao início, foi bastante agradável e de futuro esperamos ter a possibilidade de as utilizar mais vezes.

Concluindo, com este projeto todo grupo aprendeu mais sobre criação de bases de dados consistentes, simples, eficazes e redundantes.