



Universidade do Minho  
Escola de Ciências

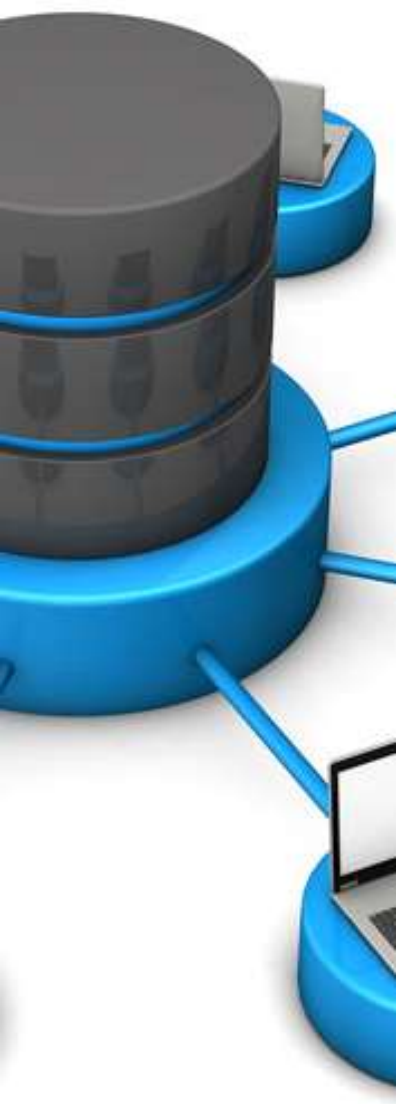
# Gestão de Eventos em Cidadela das Flores

---

BASES DE DADOS

Paulo Freitas  
& Miguel Martins  
& Wellyson Vieira

15 de janeiro de 2024



# Tópicos a abordar:

---

PT.1

- + Contextualização e Fundamentação,
- + Objetivos e Viabilidade,
- + Recursos, Equipa de trabalho e Plano de execução de trabalho,
- + Método de levantamento e organização de requisitos,
- + Modelo Conceptual, entidades e relacionamentos,



# Tópicos a abordar:

---

PT.2

- + Modelo Lógico,
- + Normalização de Dados,
- + Interrogações usando Álgebra Relacional,
- + Implementação Física,
- + Queries ( Interrogações, procedimentos e views),
- + Indexação,
- + Tamanho da base de dados e crescimento anual,
- + Segurança,
- + Conclusões



## Contextualização e Fundamentação

---

A Cidadela das Flores é uma cidade com bastante população jovem no Alto Minho. Possui vários eventos durante o ano, um deles sendo o festival de verão “Funk In Flores”. O Município vê-se com problemas em definir os eventos. A Dra. Ana Santos propôs o desenvolvimento de uma base de dados.





# Objetivos

---

## 🎯 Melhorar a Gestão de Eventos:

- Otimizar a agenda e organização dos eventos.

## 🎯 Aprimorar a Divulgação:

- Facilitar a divulgação de informações relevantes sobre os eventos futuros.

## 🎯 Conhecer o Público-Alvo:

- Coletar dados sobre o perfil dos participantes de eventos, de forma a melhorar a oferta mediante os gostos do público.

## 🎯 Eficiência de Recursos:

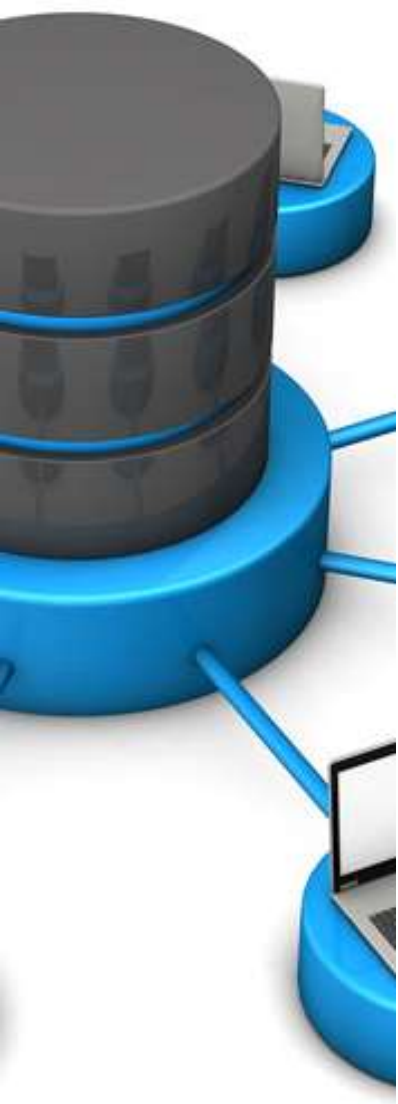
- Maximizar o uso de recursos, como locais e equipa técnica.



# Viabilidade

---

- 🎯 Recursos disponíveis,
- 🎯 Benefícios para a comunidade,
- 🎯 Proteção de dados,
- 🎯 Aceitação da comunidade,
- 🎯 Planeamento financeiro.



# Recursos e equipa de trabalho

---

## Ferramentas:

- 🌀 BrModelo,
- 🌀 MySQL Workbench e MySQL Server,

## Equipa:

- 🌀 3 arquitetos de SQL.
- 🌀 1 gestor de eventos - ( Leandro Barbosa)

## Outros recursos:

- 🌀 Hardware e Software,
- 🌀 Agentes de eventos ( limpeza, manutenção, organizadores, som, etc)
- 🌀 Artistas,
- 🌀 Clientes ( população e turistas).



# Plano de execução

Name	:	Sep 24, 2023							Oct 01, 2023							Oct 08, 2023				
		S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	
Definição do Sistema																				
Levantamento e Análise de Requisitos																				
Modelo Conceptual																				

ID	Name	Nov, 2023					Dec, 2023				
		01 Nov	05 Nov	12 Nov	19 Nov	26 Nov	03 Dec	10 Dec	17 Dec	24 Dec	
1	Modelo Lógico										
2	Implementação Física										
3	Conclusões										

Dias: 4 - 7 - 6 - 4 - 12 - 4





# Método de Levantamento de Requisitos

---

Os requisitos foram obtidos pelos seguintes métodos:

- Questionário,
- Reuniões,
- Testemunhos,
- Análise de documentos a partir dos questionários.

## Questionário - Eventos em Cidadela das Flores

Questionário sobre os eventos na cidade de Cidadela das Flores

Que tipo de evento estarias mais interessado?

Sua resposta

Qual a altura do ano achas melhor para eventos?

- ☐ Verão
- ☐ Primavera
- ☐ Outono
- ☐ Inverno

O que achaste da gestão de eventos passados?

- ☐ Excelente
- ☐ Bom
- ☐ Aceitável
- ☐ Má

## Organização de requisitos : Descrição

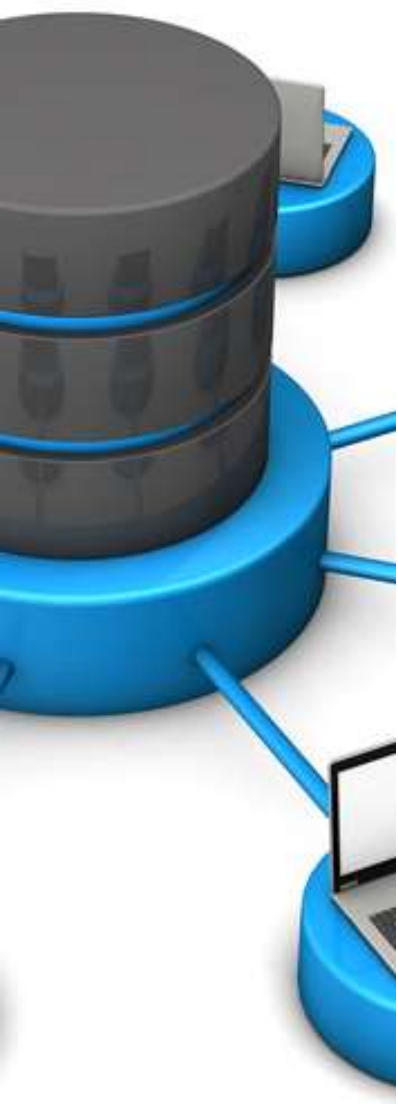
ID	DATA/HORA	REQUISITO
1	27/09/2023 16:18	Todos os eventos devem estar registados, juntamente com a data correspondente, local correspondente, artistas e clientes, utilizando um método de identificação exclusivo.
2	27/09/2023 16:19	Nenhum evento deve estar agendado para o mesmo dia, evitando conflitos de datas.
3	27/09/2023 16:20	Todos os usuários devem estar registados na base de dados incluindo informações de identificação, como e-mail, número de telefone, idade e nome. Assim como os agentes de evento devem estar registados na base de dados incluindo sua designação, o nome e informações de contacto como e-mail e número de telefone
4	27/09/2023 16:20	Cada evento terá uma designação, data, lotação e localização pré-definidos.
5	27/09/2023 16:26	Os custos totais, incluindo aluguer de locais e pagamento de artistas, pagamento dos clientes, entre outros, serão registados. Assim como o estado deste pagamento
6	27/09/2023 17:25	Todos os agentes de evento devem estar registados para trabalhar no evento
7	11/10/2023 14:00	Cada cliente pode estar associado a no máximo uma morada e cada morada pode estar associada a nenhum ou vários clientes
8	11/10/2023 14:05	Cada cliente pode estar associado a vários eventos e cada evento pode estar associado a vários clientes
9	27/09/2023 16:21	Serão realizados questionários para entender as preferências do público em relação aos eventos.
10	11/10/2023 14:15	Um evento está associado a apenas uma morada (localização) e uma morada está associada a nenhum ou vários eventos
11	11/10/2023 14:20	Um agente de evento pode estar associado a vários eventos e vice-versa

## Organização de requisitos : Exploração

ID	DATA/HORA	REQUISITO
1	11/10/2023 13:39	Deve ser possível listar todos os clientes que vão a um certo evento
2	11/10/2023 13:39	Deve ser possível enumerar todos os clientes que moram em uma localidade específica
3	11/10/2023 16:43	Deve ser possível enumerar todos os clientes que pagaram uma certa quantia de dinheiro
4	13/10/2023 15:50	Deve ser possível enumerar todos os agentes de evento em um certo evento
5	13/10/2023 15:43	Deve ser possível listar a morada com o preço de aluguer mais alto para um evento

## Organização de requisitos : Controlo

ID	DATA/HORA	REQUISITO
1	27/09/2023 16:21	A base de dados deve ser protegida de forma a evitar acesso não autorizado.
2	27/09/2023 16:25	Leandro Barbosa, Dra. Maria Santos e os engenheiros terão acesso a todo a base de dados (administrador), podendo adicionar eventos que são pedidos pelos clientes
3	27/09/2023 16:40	O sistema terá de funcionar apenas entre as 8h até às 20h
4	11/10/2023 13:43	Na base de dados apenas é inserido utilizadores que tenham idade acima de 18 anos



# Validação de Requisitos

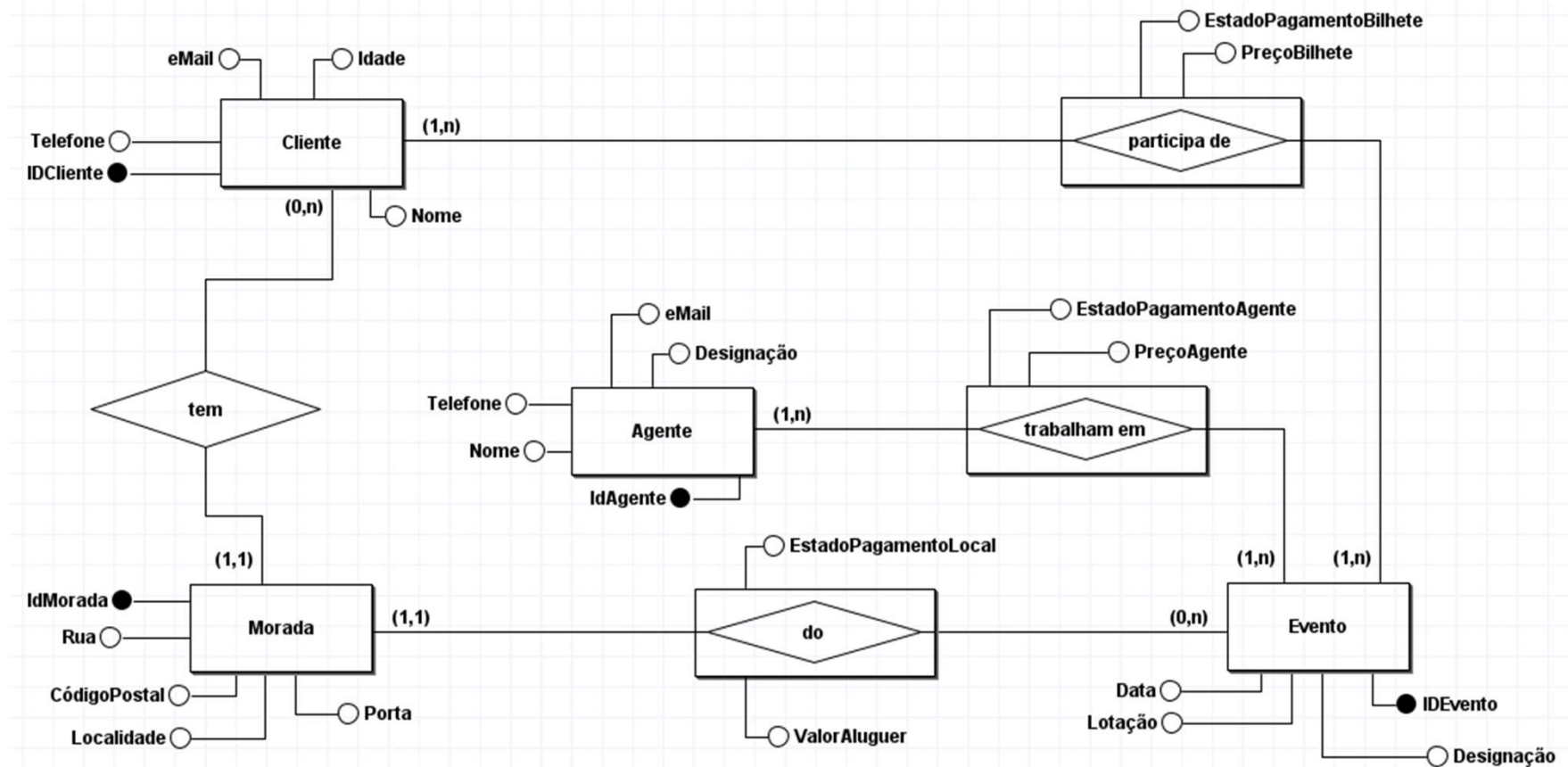
---

Com os requisitos de descrição é possível entender o funcionamento do sistema e de que forma ele está organizado.

Com os requisitos de exploração conseguimos identificar as funcionalidades interações com o usuário que o sistema deve ser capaz de realizar.

Com os requisitos de controle define a proteção do sistema, e os usuários administradores.

# Modelo Conceptual



# Modelo Lógico : Construção

---

## Cliente

- Primary Key: IDCliente (INT);
- Atributos: Nome (VARCHAR(75));  
Telefone (VARCHAR(20));  
Email (VARCHAR(50));  
Idade (INT);
- Foreign Key: iDMorada (INT);

## Agente

- Primary Key: IDAgente (INT);
- Atributos: Nome (VARCHAR(50));  
Email (VARCHAR(50));  
Designação (VARCHAR(50));  
Telefone (VARCHAR(20));

## Evento

- Primary Key: IDEvento (INT);
- Atributos: Designação (VARCHAR(50));  
Data (DATETIME);  
Lotacao (INT);  
ValorAluguer (INT);  
EstadoPagamento (BINARY(1));
- Foreign Key: idMorada (INT);

## Morada

- Primary Key: IDMorada (INT);
- Atributos: Rua (VARCHAR(50));  
Porta (VARCHAR(50));  
CodigoPostal (VARCHAR(10));  
Localidade (VARCHAR(50));



## Modelo Lógico : Construção (Cont.)

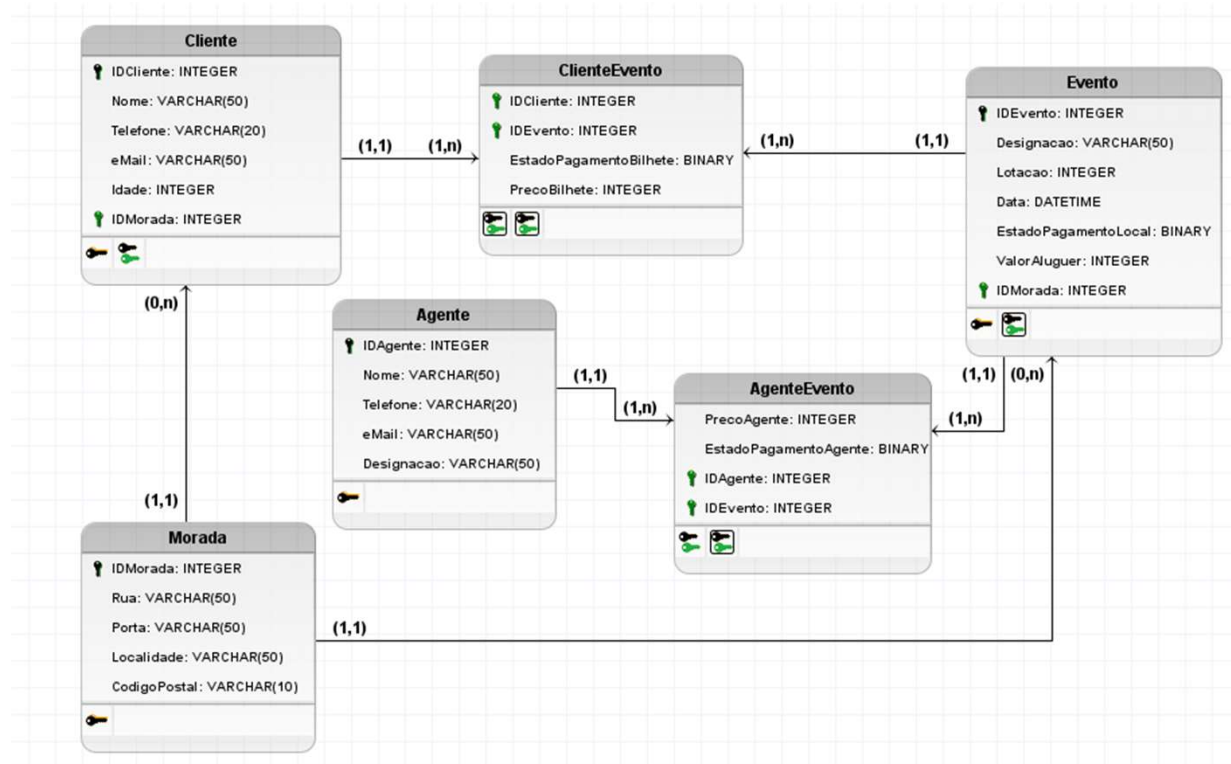
### AgenteEvento

- Primary Key: IDAgente (INT);  
IDEvento (INT);
- Atributos: PrecoAgente (DECIMAL(8,2))  
EstadoPagamento (BINARY(1));
- Foreign Key: IDAgente (INT);  
IDEvento (INT);

### ClienteEvento

- Primary Key: IDCliente (INT);  
IDEvento (INT);
- Atributos: PrecoPagamentoBilhete (DECIMAL(8,2))  
EstadoPagamento (BINARY(1));
- Foreign Key: IDEvento (INT);  
IDEvento (INT);

# Modelo Lógico





# Normalização de dados

---

No presente projeto, as tabelas encontram-se na 3FN. Isto é, apenas existem valores atômicos, todos os atributos que não são chave primária em cada tabela, dependem da mesma. Por fim, não há dependência transitiva entre atributos não primos e a chave primária.

Respeita a 1FN, a 2FN e a 3FN.



# Interrogações para Validação

---

- **O identificador, o nome e a idade de todos os clientes:**

$\Pi \text{ IDCLIENTE, NOME, IDADE (CLIENTE)}$

- **O nome, o identificador do evento e a idade de todos os clientes que vão a um certo evento e têm uma idade acima de 24 anos:**

$\Pi \text{ NOME, IDEVENTO, IDADE } (\sigma \text{ IDEVENTO}=2 \wedge \text{IDADE}>24 \text{ (CLIENTE} \\ \bowtie \text{(CLIENTE.IDCLIENTE = CLIENTEEVENTO.IDCLIENTE) CLIENTEEVENTO))$

## Implementação Física: Construção de Tabelas

---

-----  
-- CRIACAO TABELA - MORADA  
-----

CREATE TABLE IF NOT EXISTS MORADA(  
IDMORADA INT NOT NULL,  
RUA VARCHAR(50) NOT NULL,  
PORTA VARCHAR(50) NOT NULL,  
CODIGOPOSTAL VARCHAR(50) NOT NULL,  
LOCALIDADE VARCHAR(50) NOT NULL,  
PRIMARY KEY(IDMORADA)  
);

-----  
-- CRIACAO TABELA - CLIENTE  
-----

CREATE TABLE IF NOT EXISTS CLIENTE(  
IDCLIENTE INT NOT NULL,  
NOME VARCHAR(75) NOT NULL,  
EMAIL VARCHAR(50),  
IDADE INT NOT NULL,  
IDMORADA INT NOT NULL,  
TELEFONE VARCHAR(20) NOT NULL,  
PRIMARY KEY (IDCLIENTE),  
FOREIGN KEY(IDMORADA) REFERENCES MORADA(IDMORADA)  
);

# Implementação Física: Povoamento de Tabelas

---

```
-----  
-- POVOAMENTO MORADA  
-----
```

```
insert into MORADA values(1,'Rua das Lágrimas','34','4566-224','Cidadela das Flores');  
insert into MORADA values(2,'Rua das Acácias','25','4221-525','Porto');  
insert into MORADA values(3,'Avenida de São Romeo','42','4624-367','Braga');  
insert into MORADA values(4,'Rua das Andorinhas','12','3115-312','Cidadela das Flores');  
insert into MORADA values(5,'Avenida dos Pandas','16','4114-323','Cidadela das Flores');  
insert into MORADA values(6,'Rua dos sapos','156','4762-578','Cidadela das Flores');  
insert into MORADA values(7,'Rua Comendador Paulo Freitas','2','4127-254','Bragança');  
insert into MORADA values(8,'Rua da Maria das Dores','564','4234-691','Localidade da Maria das Dores');  
insert into MORADA values(9,'Rua do Pontinho','12','4982-393','Arouca');  
insert into MORADA values(10,'Rua da Dona Acácia','68','4652-271','Cidadela das Flores');  
insert into MORADA values(11,'Avenida das Máquinas','2','4989-898','Cidadela das Flores');  
insert into MORADA values(12,'Rua da Madeira','87','4234-567','Cidadela das Flores');
```

```
-----  
-- POVOAMENTO CLIENTE  
-----
```

```
insert into CLIENTE values(1,'Irenilde da Silva Costa','irenilde123@gmail.com',45,1,'912345092');  
insert into CLIENTE values(2,'Richarlyson Vieira','richas24@outlook.com',19,2,'914551234');  
insert into CLIENTE values(3,'Jasinto Leite','jasleite@hotmail.com',37,3,'943323011');  
insert into CLIENTE values(4,'Jeremias Fonseca','wells@sapo.pt',52,7,'961961961');  
insert into CLIENTE values(5,'Ana Maria das Dores','anamariadores08@gmail.com',71,8,'928008543');  
insert into CLIENTE values(6,'Fábio Martins','masterpt2000@yahoo.com',18,9,'914487245');
```

## Traduções das interrogações para SQL

```
SELECT IDCLIENTE,NOME,IDADE FROM CLIENTE;
```

IDCLIENTE	NOME	IDADE
1	Irenilde da Silva Costa	45
2	Richarlyson Vieira	19
3	Jasinto Leite	37
4	Jeremias Fonseca	52
5	Ana Maria das Dores	71
6	Fábio Martins	18
NULL	NULL	NULL

```
SELECT NOME,IDEVENTO,IDADE  
FROM CLIENTE AS C  
INNER JOIN CLIENTEEVENTO AS CE  
ON C.IDCLIENTE = CE.IDCLIENTE  
WHERE IDEVENTO = 2 AND IDADE > 24;
```

NOME	IDEVENTO	IDADE
Jasinto Leite	2	37



# Definição de Vistas

```
CREATE VIEW Clientes_PBA
AS
    SELECT Nome, Idade
    FROM CLIENTE c
    INNER JOIN MORADA m ON c.IdMorada = m.IdMorada
    WHERE Localidade = 'Porto' OR Localidade = 'Braga' OR Localidade = 'Arouca'
    ORDER BY Idade ASC;

SELECT * FROM Clientes_PBA;
```

	Nome	Idade
▶	Fábio Martins	18
	Richarlyson Vieira	19
	Jasinto Leite	37

```
CREATE VIEW Eventos
AS
    SELECT Designacao AS "Nome",
           Lotacao,
           Data
    FROM EVENTO
    ORDER BY Data DESC;

SELECT * FROM Eventos;
```

	Nome	Lotacao	Data
▶	Véspera de ano novo	5200	2022-12-31 18:00:00
	Halloween	2005	2022-10-31 18:00:00
	Festival de Verão	3900	2022-08-02 15:30:00
	Festival das Flores	2500	2022-07-24 14:00:00
	Funk in Flores	4700	2022-06-15 15:00:00
	Festival de Rancho	1630	2022-01-09 09:00:00

# Cálculo do espaço da base de dados

Atributos Agente	Tipo de Dados	Tamanho máximo
IDAgente	INT	4 bytes
Nome	VARCHAR(75)	76 bytes
Email	VARCHAR(50)	51 bytes
Designação	VARCHAR(50)	51 bytes
Telefone	VARCHAR(20)	21 bytes

Atributos Cliente	Tipo de Dados	Tamanho máximo
IDCliente	INT	4 bytes
Nome	VARCHAR(75)	76 bytes
Email	VARCHAR(50)	51 bytes
Idade	INT	4 bytes
IDMorada	INT	4 bytes
Telefone	VARCHAR(20)	21 bytes

Atributos Evento	Tipo de Dados	Tamanho máximo
IDEvento	INT	4 bytes
Designação	VARCHAR(50)	51 bytes
Data	DATETIME	5 bytes
Lotação	INT	4 bytes
ValorAluguer	DECIMAL(8,2)	8 bytes
EstadoPagamentoLocal	BINARY(1)	1 byte
IDMorada	INT	4 bytes

Atributos AgenteEvento	Tipo de Dados	Tamanho máximo
IDAgente	INT	4 bytes
IDEvento	INT	4 bytes
PreçoAgente	DECIMAL(8,2)	8 bytes
EstadoPagamentoAgente	BINARY(1)	1 byte

Atributos ClienteEvento	Tipo de Dados	Tamanho máximo
IDCliente	INT	4 bytes
IDEvento	VARCHAR(75)	76 bytes
PreçoBilhete	VARCHAR(50)	51 bytes
EstadoPagamentoBilhete	VARCHAR(50)	51 bytes

Atributos Morada	Tipo de Dados	Tamanho máximo
IDMorada	INT	4 bytes
Rua	VARCHAR(50)	51 bytes
Porta	VARCHAR(50)	51 bytes
CódigoPostal	VARCHAR(50)	51 bytes
Localidade	VARCHAR(50)	51 bytes

- Tamanho máximo inicial: 847 bytes
- Suponha-se que no primeiro ano foram adicionados 300 registos
- Considerando que existiam 1500 registos no início, a taxa de crescimento do 1º ano será de 20%

A 3D illustration of a database cylinder, rendered in blue and grey, with several blue lines representing data connections extending from its base. It is positioned on the left side of the slide.

# Indexação

---

```
CREATE INDEX index_morada ON MORADA(IDMORADA);  
CREATE INDEX index_evento ON EVENTO(IDEVENTO);
```

# Procedimentos implementados

## Listar a morada com o preço de aluguer mais alto para um evento

```
DELIMITER $$
CREATE PROCEDURE GetMoradaMaisCara()
) BEGIN
    SELECT LOCALIDADE,RUA,PORTA,VALORALUGUER FROM MORADA AS M
    INNER JOIN EVENTO AS E ON M.IDMORADA = E.IDMORADA
    ORDER BY E.VALORALUGUER DESC
    LIMIT 1;
END
$$

CALL GetMoradaMaisCara();
```

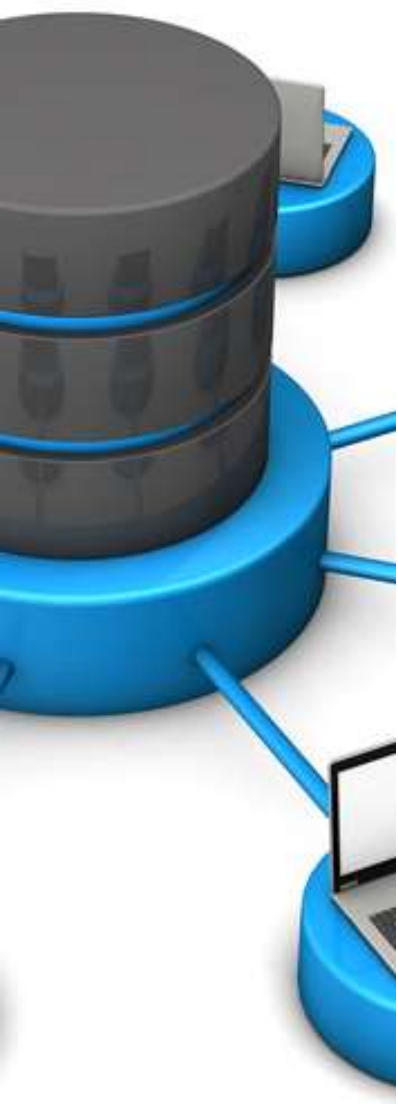
LOCALIDADE	RUA	PORTA	VALORALUGUER
Cidadela das Flores	Avenida das Máquinas	2	7025.00

## Enumerar todos os clientes que pagaram uma certa quantia ou menor

```
DELIMITER $$
CREATE PROCEDURE GetClientesPrecoBilhete(IN quantia INT)
BEGIN
    SELECT NOME,PRECIBILHETE
    FROM CLIENTE AS C
    LEFT JOIN CLIENTEEVENTO AS E
    ON C.IDCLIENTE = E.IDCLIENTE
    WHERE PRECOBILHETE <= quantia
    ORDER BY PRECOBILHETE ASC;
END
$$

CALL GetClientesPrecoBilhete(200);
```

NOME	PRECIBILHETE
Richarlyson Vieira	120.00
Jeremias Fonseca	190.00
Jasinto Leite	200.00



# Plano de segurança e recuperação de dados

---

Este sistema está sujeito a uma perda de dados ou até mesmo à corrupção dos mesmos, proveniente de erros que possam ocorrer na base de dados.

Realiza backups incrementais diários e segue o método “3-2-1”. Um backup fica no servidor, outro é enviado para a Cloud e outro vai para um dispositivo físico não conectado à rede.

Realiza backups completos mensais guardados em um dispositivo físico não conectado à rede.



# Conclusão

---

De modo geral, conseguimos desenvolver uma base de dados adequada ao problema;

Existe erros no cálculo do tamanho da base de dados;

Não conseguimos desenvolver soluções para os requisitos de controlo.



Universidade do Minho  
Escola de Ciências

# Gestão de Eventos em Cidadela das Flores

---

BASES DE DADOS

Paulo Freitas  
& Miguel Martins  
& Wellyson Vieira

15 de janeiro de 2024