

Mademoiselle Borges: Um Sistema de Bases
de Dados para Gestão de Eventos em
Eventopolis
Grupo 06
Bases de Dados

Bruno Gião
A96544

João Pereira
A95375

Helena Salazar
A75635

Tiago Teixeira
A97666

15 de janeiro de 2024

1 INTRODUÇÃO

- Apresentação do Caso de Estudo

2 METODOLOGIA

- Definição do Sistema
- Definição de Requisitos
- Modelação Concetual
- Modelação Lógica
- Modelação Física

3 CONCLUSÕES E TRABALHO FUTURO

APRESENTAÇÃO DO CASO DE ESTUDO

Este trabalho consistirá na elaboração de um SBD que consiga, aptamente, ajudar Henrique Borges e a câmara municipal de Eventopolis a gerir e publicitar os seus eventos.

CONTEXTUALIZAÇÃO

Em Eventopolis, uma localidade remota no centro de uma densa floresta, a gestão dos eventos sempre foi baseada em *outsourcing* ou métodos manuais, devido à escassez de recursos humanos e à existência de um monopólio na área de Bases de Dados (BD). Este monopólio era controlado por uma seita de ocultistas tecnológicos, os quais praticavam preços exorbitantes e limitavam o acesso a uma parte significativa das informações nas suas bases de dados. Após uma revolta motivada pela insatisfação com a direção da empresa, alguns ex-membros, descontentes com a situação, optaram por adotar uma abordagem mais humanista e criar uma *start-up* de Engenharia de Software em Eventopolis.

Ao tomar conhecimento desta informação, o Professor Doutor Henrique Borges, responsável atual pela Gestão de Eventos na Câmara Municipal da cidade, prontamente identificou a oportunidade de mitigar os prejuízos significativos dos últimos anos ao estabelecer um contrato com a referida *start-up* para a implementação de um sistema de Bases de Dados *open-source*.

CONTEXTUALIZAÇÃO

O sistema de Bases de Dados seria batizado de “Mademoiselle Borges” em homenagem a Antoinette Borges, a antiga gestora de Eventos da Câmara Municipal de Eventopolis e esposa de Henrique Borges, que faleceu há alguns anos. Antoinette enfrentou uma pressão considerável ao depender da seita ou ao ser forçada a gerir manualmente os eventos com uma equipa de funcionários bastante limitada, desafios que foram fatores cruciais para o seu falecimento precoce.

Para Henrique Borges, este projeto tem então um significado profundamente pessoal. Além de simplificar o funcionamento dos eventos, diminuindo a mortalidade deste posto de trabalho, a criação deste Sistema também reflete a sua vontade de fomentar a promoção da arte e da cultura na sua pequena cidade, algo que era o maior sonho da sua falecida esposa. Antoinette queria ver a transformação da modesta e isolada cidade numa capital cultural, uma aspiração que, infelizmente, apenas se concretizaria após o seu falecimento.

CONTEXTUALIZAÇÃO

Durante todos os eventos aprovados pela Câmara a cidade será transformada num cenário requintado que exalta a estética do estilo *Art Nouveau*, o estilo artístico predileto da *Mademoiselle*, este estilo tira inspiração da vegetação exuberante, densa e colorida, característica das imensas florestas que rodeiam Eventopolis. O principal local de eventos será uma gigantesca estufa situada no parque central, construída no início do século anterior. Esta estrutura possui uma cúpula central e vitrais coloridos, com um esqueleto de ferro com linhas detalhadas e artísticas, que ao longo do tempo oxidaram e agora exibem uma tonalidade verde clássica.

FUNDAMENTAÇÃO

Considerando o modo prévio de gerir eventos em Eventopolis, onde o uso de serviços externos era considerado excessivamente dispendioso, e diante da escassez de recursos humanos para uma gestão manual, a única alternativa viável, na perspetiva de Henrique Borges, seria desenvolver um SBD interno.

OBJETIVOS

O Professor Doutor Henrique Borges acredita que a introdução de uma base de dados trará sucesso aos eventos.

Os objetivos mencionados abaixo são fundamentais para refletir este sucesso:

- Aumentar a capacidade de armazenamento de informações;
- Saber em tempo real qual a previsão de afluência de cada evento, sendo assim possível planejar os eventos com maior precisão;
- Perceber quais são os colaboradores com melhor desempenho nas vendas, permitindo o uso de incentivos para estimulá-los a alcançar novos patamares de vendas;
- Possibilitar uma gestão financeira mais abrangente e precisa;
- Garantir que é minimizada a possibilidade da capacidade do evento ser excedida;

OBJETIVOS

- Obter, em tempo real, um registro preciso das compras de cada participante, bem como identificar os itens mais vendidos tanto em eventos específicos quanto globalmente;
- Melhorar a organização de horários para cada evento;
- Promover a cidade no âmbito nacional e internacional;
- Estimular a economia local por meio de injeção de capital na região;

VIABILIDADE

O Professor Doutor Henrique Borges defende que ao implementar um sistema de controlo de eventos será possível:

- Recuperar, no final no primeiro semestre, 40% das perdas anteriores e cerca de 20% do investimento inicial;
- Aumentar a participação nos eventos em 20%;

RECURSOS A UTILIZAR

Recursos Humanos:

- Pessoal de limpeza;
- Equipa de segurança;
- Vendedores;
- Equipa de multimédia;
- Funcionários da empresa de desenvolvimento;
- Potenciais Voluntários.

Recursos Materiais:

- Hardware:
 - 1 servidor fornecido pela *start-up* com 128GiB;
 - 15 terminais “burros”;
 - 10 computadores pessoais.
- Software:
 - SGBD;
 - Aplicação de vendas e aprovisionamento;
 - Redes sociais para divulgar o calendários dos eventos.

EQUIPA DE TRABALHO

- **Pessoal Interno** Na equipa de gestão de eventos da Câmara Municipal de Eventopolis temos:
 - Professor Doutor Henrique Borges: O coordenador principal da equipa;
 - Maria Ivanovna Ivanova: Colaboradora com experiência em *marketing* e co-coordenadora da equipa;
 - Herr Otto Mustermann: Trabalhador *part-time*.
- **Pessoal Externo** Já o pessoal externo, consiste na equipa de desenvolvimento da “start-up”, que seria constituída por 4 engenheiros, nomeadamente:
 - Luke Bytespell
 - Aurelius Cibernético
 - Bella Firewall
 - Aurora Matrix

PLANO DE EXECUÇÃO DO TRABALHO

De forma a atempadamente desenvolver o SBD “Mademoiselle Borges”, Henrique Borges e a equipa de desenvolvimento juntaram-se numa reunião e elaboraram o seguinte esquema GANTT:

Implementação do Sistema de Bases de Dados Eventopolis

Início do Projeto: **seg, 9/18/2023**
Semana: **1**

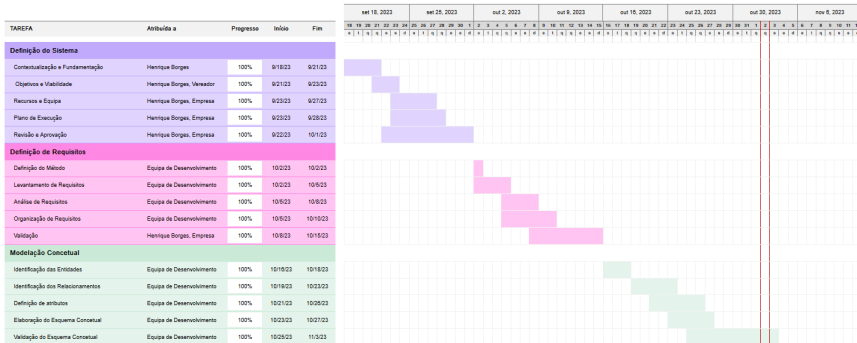


FIGURA: Diagrama de GANTT com conteúdos da primeira fase do Trabalho

PLANO DE EXECUÇÃO DO TRABALHO

Implementação do Sistema de Bases de Dados Eventopolis

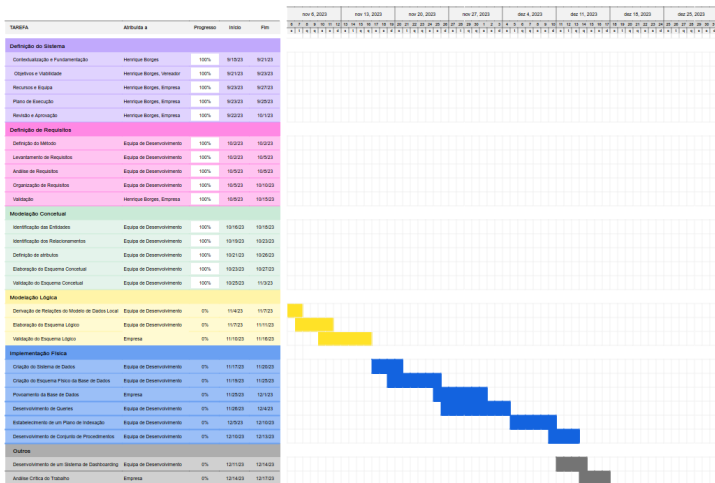
Início do Projeto: **seg, 9/18/2023**Semana: **8**

FIGURA: Diagrama de GANTT com conteúdos da segunda fase do Trabalho

MÉTODO DE LEVANTAMENTO E DE ANÁLISE DE REQUISITOS ADOTADO

METODOLOGIA

Com o objetivo de determinar os objetivos a serem alcançados pelo sistema de bases de dados, foram agendadas diversas reuniões com o Prof. Dr. Henrique Borges, onde foram discutidas várias questões pertinentes. No final destas reuniões, é previsto obter-se uma compreensão abrangente dos requisitos a serem implementados.

REQUISITOS DE DESCRIÇÃO

Nr	Hour	Description	Source	Analist	Type
1	12:29	Each event must have an unique identifier.	Henrique Borges	Aurora Matrix	DR
2	12:29	Each event must have an unique name	Henrique Borges	Aurora Matrix	DR
3	12:29	Each event must have a description	Henrique Borges	Aurora Matrix	DR
4	12:29	Each event must have a beginning and ending date	Henrique Borges	Aurora Matrix	DR
5	12:29	Each event must have a maximum capacity	Henrique Borges	Aurora Matrix	DR
6	12:30	There are no concurrent events	Henrique Borges	Aurora Matrix	DR
7	12:30	Each employee must have an unique identifier which is formatted as XXXXXYYYYY where XXXXX is a descriptor of rank or function and YYYYY is a number	Henrique Borges	Aurora Matrix	DR
8	12:30	Each employee must have a name	Henrique Borges	Aurora Matrix	DR
9	12:30	Each employee must have a VAT	Henrique Borges	Aurora Matrix	DR
10	12:30	Each employee must have a birth date	Henrique Borges	Aurora Matrix	DR
11	12:30	Each employee must have a list of emails	Henrique Borges	Aurora Matrix	DR
12	12:30	Each employee must have a list of phone numbers	Henrique Borges	Aurora Matrix	DR
13	12:30	Each employee must have an address	Henrique Borges	Aurora Matrix	DR
14	12:30	"An address is a composite of Street name, Locale and Postal Code"	Henrique Borges	Aurora Matrix	DR
15	12:31	Each sale must have an unique identifier	Henrique Borges	Aurora Matrix	DR
16	12:31	Each sale must have a total sale value which is the result of a certain calculation	Henrique Borges	Aurora Matrix	DR
17	12:31	Each sale must have the quantity of products	Henrique Borges	Aurora Matrix	DR
18	12:31	Each sale must have the date the sale closed	Henrique Borges	Aurora Matrix	DR
19	12:32	Each participant must have an unique identifier	"Henrique Borges"	Aurora Matrix	DR
20	12:32	Each participant must have a name	Henrique Borges	Aurora Matrix	DR

REQUISITOS DE DESCRIÇÃO

21	12:32	Each participant must have date of birth	Henrique Borges	Aurora Matrix	DR
22	12:32	Each participant must have a list of phone numbers	Henrique Borges	Aurora Matrix	DR
23	12:32	" Each participant must have, optionally, a list of emails"	Henrique Borges	Aurora Matrix	DR
24	12:32	" Each participant must have, optionally, their VAT"	Henrique Borges	Aurora Matrix	DR
25	12:32	" Each participant must have optionally, address"	Henrique Borges	Aurora Matrix	DR
26	12:33	Each product must have an unique identifier	Henrique Borges	Aurora Matrix	DR
27	12:33	Each product must have an unique name	Henrique Borges	Aurora Matrix	DR
28	12:33	Each product must have a current price	Henrique Borges	Aurora Matrix	DR
29	12:33	Each product must have a stock which is a number that represents the total quantity of a given product in storage	Henrique Borges	Aurora Matrix	DR
30	12:33	Each product must have a description	Henrique Borges	Aurora Matrix	DR
31	12:33	A ticket is a product who's name is the same as the event's name	Henrique Borges	Aurora Matrix	DR
32	12:34	Each supplier must have an unique identifier	Henrique Borges	Aurora Matrix	DR
33	12:34	Each supplier must have an unique name	Henrique Borges	Aurora Matrix	DR
34	12:34	Each supplier must have an IBAN	Henrique Borges	Aurora Matrix	DR
35	12:34	Each supplier must have an list of emails	Henrique Borges	Aurora Matrix	DR
36	12:34	Each supplier must have an list of cellphones	Henrique Borges	Aurora Matrix	DR
37	12:34	Each supplier must have an address	Henrique Borges	Aurora Matrix	DR
38	12:34	" The only type of identifier that should be manually inserted is the employee identifier, every other should be an automatic number"	Henrique Borges	Aurora Matrix	DR

REQUISITOS DE DESCRIÇÃO

65	13:01	The amount of tickets does not exceed the maximum capacity of an event	Henrique Borges	Aurora Matrix	DR
70	13:07	The value of a product in a sale is dependant on it's base price at the moment of purchase and the desired quantity	Henrique Borges	Aurora Matrix	DR
79	13:23	Administrator info must be stored in the database as employees.	Henrique Borges	Aurora Matrix	DR
81	13:24	The value of a sale is sum of the value times the quantity of all products in a sale.	Henrique Borges	Aurora Matrix	DR
85	13:28	It must be possible to differentiate completed deliveries from ongoing reservations	Henrique Borges	Aurora Matrix	DR
86	13:28	It must be possible to differentiate failed reservations from ongoing reservations or completed reservations	Henrique Borges	Aurora Matrix	DR

REQUISITOS DE MANIPULAÇÃO

Nr	Hour	Description	Source	Analist	Type
43	12:37	It must be possible to consult who is the manager of an employee.	Henrique Borges	Aurora Matrix	ER
44	12:39	It must be possible to consult products in a sale	Henrique Borges	Aurora Matrix	ER
45	12:41	It must be possible to consult every closed sale	Henrique Borges	Aurora Matrix	ER
46	12:42	It must be possible to consult all participants in a given event	Henrique Borges	Aurora Matrix	ER
47	12:43	It must be possible to consult all participants in the system	Henrique Borges	Aurora Matrix	ER
48	12:44	It must be possible to consult the participant associated with a given sale	Henrique Borges	Aurora Matrix	ER
49	12:45	It must be possible to consult all products in stock	Henrique Borges	Aurora Matrix	ER
50	12:46	It must be possible to consult all suppliers of a given product	Henrique Borges	Aurora Matrix	ER
51	12:46	It must be possible to consult only past suppliers of a given product	Henrique Borges	Aurora Matrix	ER
52	12:46	It must be possible to consult only eventual suppliers of a given product	Henrique Borges	Aurora Matrix	ER
53	12:47	It must be possible to consult all of a participant's purchases	Henrique Borges	Aurora Matrix	ER
54	12:48	It must be possible to consult all suppliers	Henrique Borges	Aurora Matrix	ER
55	12:49	It must be possible to consult all employees	Henrique Borges	Aurora Matrix	ER
56	12:50	It must be possible to consult all events	Henrique Borges	Aurora Matrix	ER
57	12:51	It must be possible to consult the value of sales in a particular day	Henrique Borges	Aurora Matrix	ER
58	12:51	It must be possible to consult the volume of sales in a particular day	Henrique Borges	Aurora Matrix	ER
59	12:52	It must be possible to determine who is the participant with highest volume of sales	Henrique Borges	Aurora Matrix	ER

REQUISITOS DE MANIPULAÇÃO

60	12:53	It must be possible to determine the event with the highest volume of sales.	Henrique Borges	Aurora Matrix	ER
61	12:54	It must be possible to determine the event with the highest rate of participation	Henrique Borges	Aurora Matrix	ER
62	12:55	"When the system closes, it must dump a sales report as a text file"	Henrique Borges	Aurora Matrix	ER
63	12:59	A participant is inserted into the database when they buy a ticket	Henrique Borges	Aurora Matrix	ER
64	13:00	If an event is open-entry the sale of a ticket is still registered with 0 value	Henrique Borges	Aurora Matrix	ER
68	13:05	It must be possible to consult what events occurred in a given timespan	Henrique Borges	Aurora Matrix	ER
69	13:06	It must be possible to consult which employee sold the most tickets for a given event.	Henrique Borges	Aurora Matrix	ER
73	13:16	It must be possible to consult the events in which someone participated in	Henrique Borges	Aurora Matrix	ER
74	13:17	It must be possible to insert new events into the database	Henrique Borges	Aurora Matrix	ER
75	13:18	It must be possible to insert new employees in to the database	Henrique Borges	Aurora Matrix	ER
76	13:19	A product, if it exists, must be updated as soon as the supplying is concluded	Henrique Borges	Aurora Matrix	ER
77	13:20	A product, if new, must be added to the database as soon as it is ordered	Henrique Borges	Aurora Matrix	ER
82	13:24	It must be possible to consult the employee managed by another employee.	Henrique Borges	Aurora Matrix	ER
83	13:25	It must be possible to determine the participant with highest value of sales.	Henrique Borges	Aurora Matrix	ER
84	13:27	It must be possible to determine the event with highest value in sales.	Henrique Borges	Aurora Matrix	ER
87	13:30	It must be possible to consult all sales made by an employee	Henrique Borges	Aurora Matrix	ER

REQUISITOS DE CONTROLO

Nr	Hour	Description	Source	Analist	Type
39	12:35	Henrique Borges is a System Administrator	Henrique Borges	Aurora Matrix	AR
40	12:36	Maria Ivanovna Ivanova is a System Administrator	Henrique Borges	Aurora Matrix	AR
41	12:36	Herr Mustermann is a System Administrator	Henrique Borges	Aurora Matrix	AR
42	12:36	There must be a special way of accessing the database called guest for anyone who wishes to view information on events	Henrique Borges	Aurora Matrix	AR
66	13:02	Access to the database is only available from 07:00 to 02:00	Henrique Borges	Aurora Matrix	AR
67	13:04	Database Administrators may revoke access to the database if provided with a suitable/legal reason.	Henrique Borges	Aurora Matrix	AR
71	13:08	An administrator has access to any and all information in the database	Henrique Borges	Aurora Matrix	AR
72	13:09	An administrator has access to all and any functionalities of the database	Henrique Borges	Aurora Matrix	AR
78	13:23	Only administrators may update information	Henrique Borges	Aurora Matrix	AR
80	13:23	Only administrators have access to the total value of sales	Henrique Borges	Aurora Matrix	AR
88	13:30	The usernames in the database correspond directly to the identifier	Henrique Borges	Aurora Matrix	AR

ANÁLISE E VALIDAÇÃO GERAL DOS REQUISITOS

Depois do levantamento dos requisitos, marcou-se uma reunião no intuito de o pessoal interno tomar conhecimento dos requisitos documentados. Esta reunião, por sua vez, foi realizada com sucesso, e o pessoal interno mostrou-se satisfeito com o progresso e nível de detalhe a que os membros da equipa de desenvolvimento de BD chegaram, especialmente o Prof. Dr. Henrique Borges, que viu muito potencial neste projeto.

IDENTIFICAÇÃO E CARACTERIZAÇÃO DAS ENTIDADES E DOS ATRIBUTOS DAS MESMAS

- **Evento:** Evento a ser gerido.

- ID: Chave Primária da entidade que estará no domínio INTEGER e será auto incrementável;
- Nome: O nome de um evento que estará no domínio VARCHAR(75).
- Descrição: A descrição de um elemento da tabela “Evento” será um breve texto que introduz o tema e qualquer tipo de subevento que possa estar inserido no evento. Sendo assim, este atributo estará no domínio TEXT;
- DataFim: A data do fim de um evento será uma data com as horas a qual o evento será dado por oficialmente terminado. Sendo assim, estará no domínio DATETIME;
- DataInicio: A data do inicio de um evento será uma data com as horas a qual o evento será dado por oficialmente iniciado. Sendo assim, estará no domínio DATETIME;
- Capacidade: É importante ter noção da quantidade máxima de participantes num dado evento. Então, este atributo será um elemento numérico e estará no domínio INTEGER.

IDENTIFICAÇÃO E CARACTERIZAÇÃO DAS ENTIDADES E DOS ATRIBUTOS DAS MESMAS

- **Funcionário:** Colaborador da câmara municipal.
 - ID: Chave Primária da entidade que estará no domínio VARCHAR(10);
 - Nome: O nome legal ou social de um funcionário também será armazenado e estará no domínio VARCHAR(75);
 - NIF: O número de identificação fiscal é um número de 9 algarismos e, portanto estará no domínio VARCHAR(9);
 - DataNascimento: Data de nascimento que estará no domínio DATE;
 - Email: Email é um atributo multivalorado que estará no domínio VARCHAR(75);
 - NTelemovel: Números de telemóvel. Podem conter letras e números. É um atributo multivalorado no domínio VARCHAR(20);
 - Morada: Morada composta por rua, localidade, código-postal. Atributo Composto de VARCHAR(), onde rua será VARCHAR(50), localidade VARCHAR(30) e código-postal VARCHAR(15);

IDENTIFICAÇÃO E CARACTERIZAÇÃO DAS ENTIDADES E DOS ATRIBUTOS DAS MESMAS

- **Participante:** Pessoa que participa num evento(s). É registado no sistema quando efetua a compra do seu primeiro bilhete para um evento;
 - ID: Chave Primária da entidade que estará no domínio INTEGER;
 - Nome: O nome legal ou social da pessoa também será armazenado e estará no domínio VARCHAR(75);
 - NIF: O número de identificação fiscal é um número de 9 algarismos e, portanto estará no domínio VARCHAR(9). Este campo é opcional;
 - DataNascimento: Data de nascimento que estará no domínio DATE;
 - Email: Email do participante que estará no domínio. É um atributo multivalorado no domínio VARCHAR(75). Este campo é opcional;
 - NTelemovel: Números de telemóvel. Podem conter letras e números. É também um atributo multivalorado no domínio VARCHAR(20);
 - Morada: Morada composta por rua, localidade, código-postal. Atributo Composto de VARCHAR(), onde rua será VARCHAR(50), localidade VARCHAR(30) e código-postal VARCHAR(15). Este campo é opcional;

IDENTIFICAÇÃO E CARACTERIZAÇÃO DAS ENTIDADES E DOS ATRIBUTOS DAS MESMAS

- **Artigo:** Artigo que é possível estar numa venda;
 - ID: Chave Primária da entidade que estará no domínio INTEGER;
 - Nome: O nome do artigo que estará no domínio VARCHAR(75);
 - Descrição: A descrição de um elemento da tabela “Artigo” será um breve texto que descreve o produto em questão e qualquer medida extra necessária a ter com o mesmo. Sendo assim, este atributo estará no domínio TEXT;
 - Preço: Valor de um artigo, logo estará no domínio DECIMAL(5,2);
 - Stock: Quantidade de um artigo que está disponível, estará no domínio INTEGER;

IDENTIFICAÇÃO E CARACTERIZAÇÃO DAS ENTIDADES E DOS ATRIBUTOS DAS MESMAS

- **Venda:** Venda de artigo(s) a ser efetuada a um participante por parte de um funcionário;
 - ID: Chave Primária da entidade que estará no domínio INTEGER;
 - Valor: Valor total da venda que estará no domínio DECIMAL(5,2);
 - Quantidade: Valor total do número de artigos na venda que estará no domínio INTEGER;
 - Data: Data na qual a venda aconteceu, logo estará no domínio DATE;

IDENTIFICAÇÃO E CARACTERIZAÇÃO DAS ENTIDADES E DOS ATRIBUTOS DAS MESMAS

- **Fornecedor:** Quem fornece os artigos;
 - ID: Chave Primária da entidade que estará no domínio INTEGER;
 - Nome: O nome da empresa fornecedora que estará no domínio VARCHAR(75);
 - IBAN: Código de identificação de conta bancária para a qual devem ser feitos os pagamentos ao fornecedor. Está no domínio VARCHAR(50);
 - Email: Email do fornecedor que estará no domínio. É um atributo multivalorado no domínio VARCHAR(75);
 - Contacto: Pessoa que representa a empresa e o seu número de telemóvel, estará no domínio VARCHAR(50);
 - NTelemovel: Números de telemóvel. Podem conter letras e números. É também um atributo multivalorado no domínio VARCHAR(20);
 - Morada: Morada composta por rua, localidade, código-postal. Atributo Composto de VARCHAR(), onde rua será VARCHAR(50), localidade VARCHAR(30) e código-postal VARCHAR(15);

IDENTIFICAÇÃO E CARACTERIZAÇÃO DA ASSOCIAÇÃO DOS ATRIBUTOS COM AS ENTIDADES E RELACIONAMENTOS E DOS ATRIBUTOS DAS MESMAS

- **Evento emprega Funcionário:** Relacionamento entre Evento e Funcionário, representado quais funcionários estão alocados para quais eventos. Sendo a cardinalidade muitos para muitos;
- **Funcionário gere Funcionário:** Relacionamento entre Funcionário e Funcionário, representado qual funcionário gere qual/quais funcionários são geridos por um Funcionário. Sendo a cardinalidade um para muitos;
- **Funcionário realiza Venda:** Relacionamento entre Funcionário e Venda, representado qual funcionário efetuou uma Venda. Sendo a cardinalidade um para muitos.

IDENTIFICAÇÃO E CARACTERIZAÇÃO DA ASSOCIAÇÃO DOS ATRIBUTOS COM AS ENTIDADES E RELACIONAMENTOS E DOS ATRIBUTOS DAS MESMAS

- **Venda contém Artigo:** Relacionamento entre Venda e Artigo, representando quais artigos estão numa venda. Sendo a cardinalidade muitos para muitos.
 - Valor: Valor do artigo no momento da venda, estará então no domínio DECIMAL(5,2);
 - Quantidade: Valor total da quantidade de um dado artigo numa venda, estará então no domínio INTEGER;
- **Venda para Participante:** Relacionamento entre Venda e Participante, representando a que participante uma venda pertence. Sendo a cardinalidade 1 para muitos;

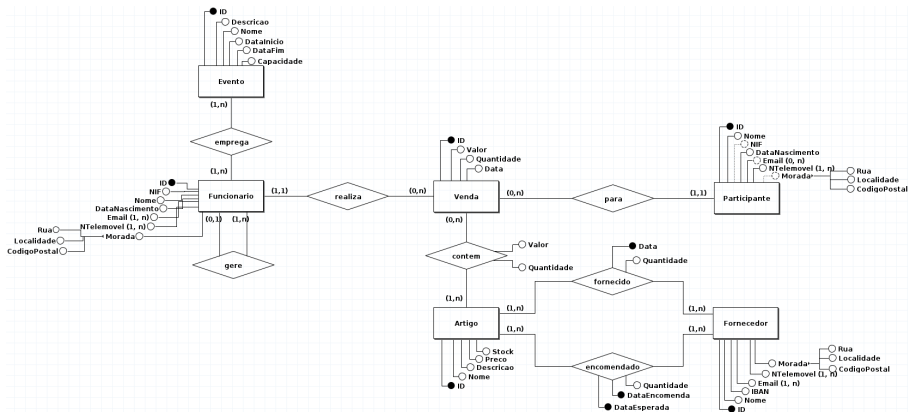
IDENTIFICAÇÃO E CARACTERIZAÇÃO DA ASSOCIAÇÃO DOS ATRIBUTOS COM AS ENTIDADES E RELACIONAMENTOS E DOS ATRIBUTOS DAS MESMAS

- **Artigo fornecido por Fornecedor:** Relacionamento entre Artigo e Fornecedor, representado o fornecedor pelo qual um artigo foi fornecido. Sendo a cardinalidade muitos para muitos;
 - Data: Data na qual o artigo foi entregue pelo fornecedor, estará então no domínio DATETIME;
 - Quantidade: Valor total da quantidade de um dado artigo numa entrega por parte de um fornecedor, estará então no domínio INTEGER;

IDENTIFICAÇÃO E CARACTERIZAÇÃO DA ASSOCIAÇÃO DOS ATRIBUTOS COM AS ENTIDADES E RELACIONAMENTOS E DOS ATRIBUTOS DAS MESMAS

- **Artigo encomendado do Fornecedor:** Relacionamento entre Artigo e Fornecedor, representado o fornecedor pelo qual um artigo foi encomendado. Sendo a cardinalidade muitos para muitos;
 - DataEncomenda: Data na qual o artigo foi encomendado ao fornecedor, estará então no domínio DATETIME;
 - DataEsperada: Data na qual o artigo é esperado que seja entregue, estará então no domínio DATETIME. DataEncomenda e DataEsperada compõe a chave primária do relacionamento;
 - Quantidade: Valor total da quantidade de um dado artigo numa encomenda por parte de um fornecedor, estará então no domínio INTEGER;

APRESENTAÇÃO E EXPLICAÇÃO DO DIAGRAMA ER PRODUZIDO



CONSTRUÇÃO E VALIDAÇÃO DO MODELO DE DADOS LÓGICO

Com a modelação conceptual completa e validada, a equipa de desenvolvimento procedeu com a modelação lógica do SBD, sendo que a mesma é uma tradução direta do modelo conceptual, introduzindo agora chaves estrangeiras.

NORMALIZAÇÃO DE DADOS

Devido à abordagem que adotámos no trabalho desde o modelo conceptual onde a leitura ocorre da esquerda para a direita e de cima para baixo, e considerando que atendemos a todos os requisitos das leis de normalização, os nossos dados encontram-se naturalmente normalizados na Terceira Forma Normal (3FN).

O facto de estarem normalizados em 3FN proporciona várias vantagens, incluindo a prevenção de anomalias como a redundância, e problemas de consistência decorrentes de processos de inserção, atualização ou remoção. Além disso, evita a atualização deficiente de um determinado registo com múltiplas ocorrências, onde nem todas são corretamente atualizadas.

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

Tradução de Entidades

● **Evento** → **EventCal**

- ID → EventID
- Nome → EventName
- Descricao → EventDescription
- DataInicio → EventStart
- DataFim → EventEnd
- Capacidade → Capacity

● **Funcionário** → **Employee**

- ID → EmployeeID
- Nome → EmployeeName
- NIF → EmployeeVAT
- DataNascimento → EmployeeBirthDate

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

Temos dois atributos multivalorados na entidade Funcionário que geram duas novas tabelas:

- **Email** → **EmployeeEmail**

- EmployeeID_eem: Chave estrangeira (chave primária e NOT NULL) que referencia a chave candidata EmployeeID na tabela Employee.
- Email: Atributo que representa um email tal como no modelo conceptual.

- **NTelemovei** → **EmployeePhone**

- EmployeeID_ep: Chave estrangeira (chave primária e NOT NULL) que referencia a chave candidata EmployeeID na tabela Employee.
- Phone: Atributo que representa um número de telemóvel como no modelo conceptual.

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

● Venda → Sale

- ID → ReceiptNO
- Valor → TotalValue
- Quantidade → TotalQuantity
- Data → DateOfSale
- EmployeeID_s: Chave estrangeira (NOT NULL) que referencia a chave candidata EmployeeID na tabela Employee.
- ParticipantID_s: Chave estrangeira (NOT NULL) que referencia a chave candidata ParticipantID na tabela Participant.

● Participante → Participant

- ID → ParticipantID
- Nome → ParticipantName
- NIF → ParticipantVAT
- DataNascimento → ParticipantBirthDate
- Rua → Street
- Localidade → Locale
- CodigoPostal → Postal

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

Temos dois atributos multivalorados na entidade Participante que geram duas novas tabelas:

- **NTelemovelel → ParticipantPhone**

- ParticipantID_pp: Chave estrangeira (chave primária e NOT NULL) que referencia a chave candidata ParticipantID na tabela Participant.
- Phone: Atributo que representa um número de telemóvel como no modelo conceptual.

- **Email → ParticipantEmail**

- ParticipantID_pem: Chave estrangeira (chave primária e NOT NULL) que referencia a chave candidata ParticipantID na tabela Participant.
- Email: Atributo que representa um email como no modelo conceptual.

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

● **Artigo** → **Produto**

- ID → ProductID
- Nome → ProductName
- Descricao → ProductDescription
- Preço → BasePrice
- Stock → QuantityInStock

● **Fornecedor** → **Supplier**

- ID → SupplierID
- Nome → SupplierName
- IBAN → IBAN
- Rua → Street
- Localidade → Locale
- CodigoPostal → Postal

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

Temos dois atributos multivalorados na entidade Artigo que geram duas novas tabelas:

- **NTelemovelel** → **SupplierPhone**

- SupplierID_sp: Chave estrangeira (chave primária e NOT NULL) que referencia a chave candidata SupplierID na tabela Supplier.
- Phone: Atributo que representa um número de telemóvel como no modelo conceptual.

- **Email** → **SupplierEmail**

- SupplierID_sem: Chave estrangeira (chave primária e NOT NULL) que referencia a chave candidata SupplierID na tabela Supplier.
- Email: Atributo que representa um email como no modelo conceptual.

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

Tradução de Relacionamentos

Os relacionamentos com cardinalidade n para n geram novas tabelas. Veremos como são traduzidos os mesmos.

- **Evento emprega Funcionário** → **EventEmployee**

- EventID_ee: Chave estrangeira (chave primária composta com EmployeeID_ee) que referencia a chave candidata EventID na tabela EventCal.
- EmployeeID_ee: Chave estrangeira (chave primária composta com EventID_ee) que referencia a chave candidata EmployeeID na tabela Employee.

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

● **Venda contém Artigo → SaleProduct**

- ReceiptNO_sp: Chave estrangeira (chave primária composta com ProductID_sp) que referencia a chave candidata ReceiptNO na tabela Sale.
- ProductID_sp: Chave estrangeira que (chave primária composta com ReceiptNO_sp) referencia a chave candidata ProductID na tabela Product.
- Valor → CurrentValue

● **Artigo fornecido por Fornecedor → ProductSupplierPast**

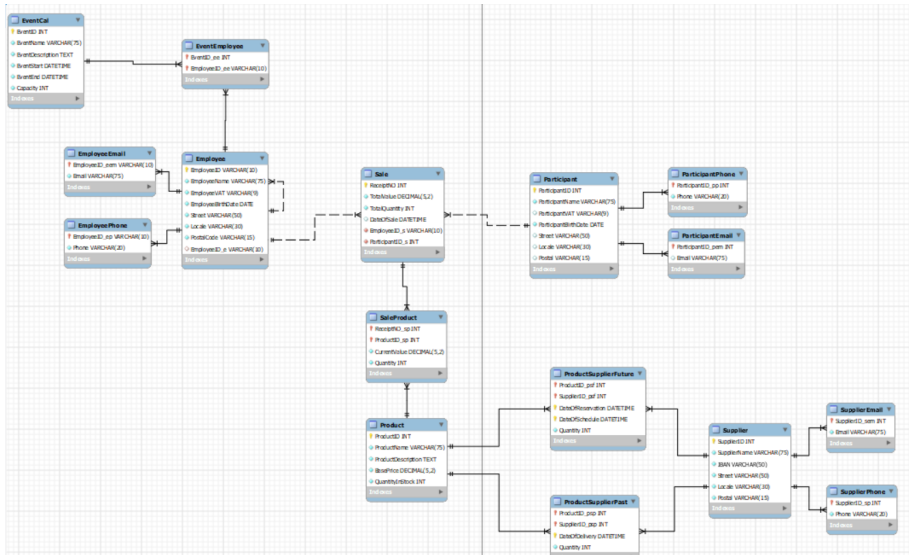
- ProductID_psp: Chave estrangeira (chave primária composta com SupplierID_psp e DateOfDelivery) que referencia a chave candidata ProductID na tabela Product.
- SupplierID_psp: Chave estrangeira (chave primária composta com ProductID_psp e DateOfDelivery) que referencia a chave candidata SupplierID na tabela Supplier.
- Data → DateOfDelivery: Chave primária composta com ProductID_psp e SupplierID_psp.
- Quantidade → Quantity

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO

● **Artigo encomendado a Fornecedor** → **ProductSupplierFuture**

- ProductID_psf: Chave estrangeira (chave primária composta com SupplierID_psf, DateOfReservation e DateOfSchedule) que referencia a chave candidata ProductID na tabela Product.
- SupplierID_psf: Chave estrangeira (chave primária composta com ProductID_psf, DateOfReservation e DateOfSchedule) que referencia a chave candidata SupplierID na tabela Supplier.
- DataEncomenda → DateOfReservation: Chave primária composta com ProductID_psf, SupplierID_psf e DateOfSchedule.
- DataEsperada → DateOfSchedule: Chave primária composta com ProductID_psf, SupplierID_psf e DateOfReservation.
- Quantidade → Quantity

APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO PRODUZIDO



VALIDAÇÃO DO MODELO COM INTERROGAÇÕES DO UTILIZADOR

- Saber qual evento teve a maior taxa de participação de todos (RM61)

$$\tau_{rate} DESC \gamma_{ID, Name; rate} \pi_{\delta}((\rho_{EV} EventCal) \bowtie (\rho_{SP} SaleProduct) \bowtie_{\alpha \wedge \beta} (\rho_P Product))$$

Sabendo que:

$\delta : EV.EventID \rightarrow ID, EV.EventName \rightarrow Name, SUM(SP.Quantity)/EV.Capacity \rightarrow rate.$

$\alpha : P.ProductID = SP.ProductID_sp$

$\beta : P.ProductName = EV.EventName$

VALIDAÇÃO DO MODELO COM INTERROGAÇÕES DO UTILIZADOR

- Saber qual é o evento com maior valor de vendas (RM94)

$$\tau_{TotVal} DESC \gamma_{ID, Name; TotVal} \pi_{\delta}((\rho_{EV} EventCal) \bowtie_{\alpha} (\rho_S Sale))$$

Onde:

$\alpha : EV.EventStart < S.DateOfSale < EV.EventEnd$

$\delta : EV.EventID \rightarrow ID, EV.EventName \rightarrow$

$Name, SUM(S.TotalValue) \rightarrow TotVal$

TRADUÇÃO DO ESQUEMA LÓGICO PARA O SISTEMA DE GESTÃO DE BASES DE DADOS ESCOLHIDO

Esta tradução é direta, sendo que até se pode utilizar uma funcionalidade do *MySQL Workbench* para a realizar. Listam-se exemplos de criação de tabelas.

CRIAÇÃO DA TABELA SALEPRODUCT

Esta tabela representa o relacionamento Venda contém Artigo, onde indicamos todos os atributos como pedido no modelo lógico, bem como a identificação de chave primária composta com ReceiptNO_sp e ProductID_sp.

```
CREATE TABLE SaleProduct (  
    ReceiptNO_sp INTEGER NOT NULL,  
    ProductID_sp INTEGER NOT NULL,  
    CurrentValue DECIMAL(5,2) NOT NULL,  
    Quantity INTEGER NOT NULL,  
    PRIMARY KEY (ReceiptNO_sp , ProductID_sp),  
    FOREIGN KEY (ReceiptNO_sp)  
REFERENCES Sale (ReceiptNO),  
    FOREIGN KEY (ProductID_sp)  
REFERENCES Product (ProductID)  
);
```

CRIAÇÃO DA TABELA EventCal

No caso da tabela EventCal, a tradução é ainda mais simples, visto que não tem chaves estrangeiras. Sendo assim, só tivemos que indicar os atributos como pedido e indicar que a chave primária é EventID.

```
CREATE TABLE EventCal (  
    EventID INTEGER AUTO_INCREMENT,  
    EventName VARCHAR(75) NOT NULL UNIQUE,  
    EventDescription TEXT NOT NULL,  
    EventStart DATETIME NOT NULL,  
    EventEnd DATETIME NOT NULL,  
    Capacity INTEGER NOT NULL,  
    PRIMARY KEY (EventID)  
);
```

TRADUÇÃO DAS INTERROGAÇÕES DO UTILIZADOR PARA SQL

Neste ponto, traduziremos alguns requisitos de manipulação em *queries* de MySQL, assim como as tabelas que resultam da execução das mesmas, tendo em conta o povoamento que usamos, que se encontra nos Anexos deste relatório.

TRADUÇÃO DAS INTERROGAÇÕES DO UTILIZADOR PARA SQL

Saber qual evento teve a maior taxa de participação de todos (RM61)

```
SELECT EV.EventID ,  
        EV.EventName ,  
        SUM(SP.Quantity) / EV.Capacity * 100 AS rate  
FROM EventCal as EV INNER JOIN SaleProduct as SP  
      INNER JOIN Product AS P  
        ON P.ProductID = SP.ProductID_sp AND  
        P.ProductName = EV.EventName  
GROUP BY EV.EventID , EV.EventName  
      ORDER BY rate DESC  
LIMIT 1;
```

#	EventID	EventName	rate
1	5	Puppet Show: 'Red Ridding Hood'	7.5000

FIGURA: Resultado RM61

TRADUÇÃO DAS INTERROGAÇÕES DO UTILIZADOR PARA SQL

Saber qual é o evento com maior valor de vendas (RM94)

```
SELECT E.EmployeeID , E.name, SUM(S.Val) AS totVal  
FROM EventCal AS E INNER JOIN sale as S  
    ON S.DateOfSale BETWEEN EV.EventStart  
    AND EV.EventFin  
GROUP BY E.EmployeeID , E.name  
ORDER BY totVal DESC  
LIMIT 1;
```

#	EventID	EventName	totVal
1	6	Arts and Crafts Fair	16.00

FIGURA: Resultado RM94

DEFINIÇÃO E CARACTERIZAÇÃO DAS VISTAS DE UTILIZAÇÃO EM SQL

Apresentemos então algumas vistas que se definiu ao longo do desenvolvimento da BD. Vistas estas que restringem o utilizador, de obter informação além do que as vistas permitem.

DEFINIÇÃO E CARACTERIZAÇÃO DAS VISTAS DE UTILIZAÇÃO EM SQL

```
CREATE VIEW product_supplier AS  
  SELECT P.ProductID AS ProductID ,  
        PSP.SupplierID_psp AS PastSupplierID ,  
        PSF.SupplierID_psf AS FutureSupplierID  
  FROM Product AS P  
        INNER JOIN ProductSupplierPast AS PSP  
  ON P.ProductID = PSP.ProductID_psp  
        INNER JOIN ProductSupplierFuture AS PSF  
  ON P.ProductID = PSF.ProductID_psf  
GROUP BY P.ProductID ,  
        PSP.SupplierID_psp ,  
        PSF.SupplierID_psf ;
```

DEFINIÇÃO E CARACTERIZAÇÃO DAS VISTAS DE UTILIZAÇÃO EM SQL

```
CREATE VIEW BestSellersEmployee AS  
  SELECT EV.EventID AS EventID ,  
        S.EmployeeID_s AS EmployeeID ,  
        SUM(SP.Quantity) AS Quantity  
FROM EventCal AS EV  
INNER JOIN EventEmployee AS EE  
  ON EV.EventID = EE.EventID_ee  
INNER JOIN Employee AS E  
  ON EE.EmployeeID_ee = E.EmployeeID
```


DEFINIÇÃO E CARACTERIZAÇÃO DAS VISTAS DE UTILIZAÇÃO EM SQL

```
INNER JOIN Sale AS S  
    ON E.EmployeeID = S.EmployeeID_s  
INNER JOIN SaleProduct AS SP  
    ON S.ReceiptNO = SP.ReceiptNO_sp  
INNER JOIN Product AS P  
    ON P.ProductName = EV.EventName  
GROUP BY EV.EventID , S.EmployeeID_s  
ORDER BY SUM(SP.Quantity) DESC;
```

CÁLCULO DO ESPAÇO DA BASE DE DADOS

De forma a computar o espaço utilizado pela Base de Dados podemos utilizar a seguinte *query* de MariaDB/MySQL:

SELECT

```
    table_schema
      AS 'Database-Name',
  SUM(data_length + index_length)
      AS 'Size-in-Bytes',
  ROUND(SUM(data_length + index_length)
        / 1024 / 1024, 2) Database Name
      AS 'Size-in-MiB'
FROM information_schema.tables
WHERE table_schema = 'mademoiselle_borges';
```

CÁLCULO DO ESPAÇO DA BASE DE DADOS

Ou, alternativamente, recorrendo a uma ferramenta, como “MySQL Workbench”, no menu contexto esquerdo, sob *schemas*, clicando no símbolo ‘i’, conseguimos ver o espaço ocupado pelo nosso SBD. Sendo assim, a nossa Base de Dados ocupa um total de 0.53 *Mebibytes* após o povoamento inicial. Como podemos ver na tabela resultante da *query* apresentada:

Database Name	Size in Bytes	Size in MebiBytes
mademoiselle_borges	557056	0.53

Após fazermos o cálculo do tamanho que a Base de Dados vai ocupar, tendo em conta o povoamento inicial, podemos concluir, assumindo que a BD tem um crescimento de 10% anual, que o seu tamanho previsto para o próximo será de 612762 bytes (0.58 MiB).

INDEXAÇÃO DO SISTEMA DE DADOS

Apesar da indexação do sistema de dados, em geral, poder vir a trazer benefícios de *performance* optamos por não implementar este processo. Devido ao tamanho reduzido da base de dados e ao facto de usarmos maioritariamente os ID's de cada tuplo nas condições *WHERE* e *GROUP BY* o ganho de *performance* seria mínimo podendo até resultar em perda de *performance* em *queries* de atualização de tabelas. No futuro, com o crescimento da base de dados, se a *performance* baixar, a indexação de dados poderá voltar a ser considerada.

PROCEDIMENTOS IMPLEMENTADOS

Os procedimentos permitem ao utilizador encapsular uma série de *queries* numa só unidade. Para além disso, permite modularidade e reutilização. Veremos agora alguns procedimentos que implementamos. No caso, estes têm o intuito de povoar a base de dados.

PROCEDIMENTOS IMPLEMENTADOS

O procedimento a seguir listado permite ao utilizador registar um novo fornecedor na base de dados.

DELIMITER &&

```
CREATE PROCEDURE register_supplier  
(IN s_name VARCHAR(75),  
  iban VARCHAR(50),  
  street VARCHAR(50),  
  locale VARCHAR(30),  
  postal VARCHAR(15),  
  email VARCHAR(75),  
  phone VARCHAR(20))
```

PROCEDIMENTOS IMPLEMENTADOS

BEGIN**DECLARE** last_ins **INTEGER**;**START TRANSACTION**;**INSERT INTO** Supplier (SupplierName ,
IBAN ,
Street ,
Locale ,
Postal)**VALUES** (s_name , iban , street , locale , postal);

PROCEDIMENTOS IMPLEMENTADOS

```
IF ROW_COUNT() = 0 THEN  
    ROLLBACK;  
END IF;  
SELECT SupplierID INTO last_ins  
    FROM Supplier  
ORDER BY SupplierID DESC LIMIT 1;  
CALL register_supplier_phone(last_ins , phone);
```


PROCEDIMENTOS IMPLEMENTADOS

```
IF ROW_COUNT() = 0 THEN  
    ROLLBACK;  
END IF;  
CALL register_supplier_email(last_ins , email);
```

```
IF ROW_COUNT() = 0 THEN  
    ROLLBACK;  
END IF;  
COMMIT;  
END &&
```

PROCEDIMENTOS IMPLEMENTADOS

Este procedimento permite criar um novo carrinho de compras de um participante novo. O participante será registado na base de dados e a compra será registada quando a compra for concluída.

DELIMITER &&

```
CREATE PROCEDURE add_prod_new_shop_new_part(  
IN e_id VARCHAR(10), pd_id INTEGER,  
part_name VARCHAR(75), part_vat VARCHAR(9),  
street VARCHAR(50), locale VARCHAR(30),  
postal VARCHAR(15), part_bd DATE,  
quant INTEGER, phone VARCHAR(20),  
email VARCHAR(75))
```

PROCEDIMENTOS IMPLEMENTADOS

BEGIN

```
    DECLARE last_ins INTEGER;  
    DECLARE last_sale_id INTEGER;  
    DECLARE cur_val DECIMAL(5,2);
```

```
START TRANSACTION;
```

```
INSERT INTO Participant ( ParticipantName ,  
                           ParticipantVAT ,  
                           ParticipantBirthDate ,  
                           Street , Locale , Postal )  
VALUES ( part_name , part_vat ,  
          part_bd , street , locale , postal );
```

PROCEDIMENTOS IMPLEMENTADOS

```
IF ROW_COUNT() = 0 THEN
    ROLLBACK;
END IF;

SELECT ParticipantID INTO last_ins
    FROM Participant
ORDER BY ParticipantID DESC LIMIT 1;

INSERT INTO ParticipantPhone
    VALUES (last_ins , phone);
```

PROCEDIMENTOS IMPLEMENTADOS

```
IF ROW_COUNT() = 0 THEN
    ROLLBACK;
END IF;
IF email IS NOT NULL THEN
    INSERT INTO ParticipantEmail
    VALUES (last_ins , email);
    IF ROW_COUNT() = 0 THEN
        ROLLBACK;
    END IF;
END IF;
```

PROCEDIMENTOS IMPLEMENTADOS

```
INSERT INTO Sale(TotalValue , TotalQuantity ,  
DateOfSale , EmployeeID_s ,  
ParticipantID_s)  
VALUES (" 0.00" , " 0" , NULL, e_id , last_ins );  
IF ROW_COUNT() = 0 THEN  
    ROLLBACK;  
END IF ;
```

PROCEDIMENTOS IMPLEMENTADOS

```
SELECT ReceiptNO INTO last_sale_id FROM Sale
ORDER BY ReceiptNO DESC LIMIT 1;
SELECT BasePrice INTO cur_val
FROM Product
WHERE ProductID = pd_id;
INSERT INTO SaleProduct(ReceiptNO_sp ,
ProductID_sp , CurrentValue ,
Quantity)
VALUES (last_sale_id , pd_id , cur_val , quant);
```

PROCEDIMENTOS IMPLEMENTADOS

```
IF ROW_COUNT() = 0 THEN
    ROLLBACK;
END IF;
UPDATE Product
SET QuantityInStock = QuantityInStock - quant
WHERE ProductID = pd_id;
IF ROW_COUNT() = 0 THEN
    ROLLBACK;
END IF;
COMMIT;
END &&
```


PROCEDIMENTOS IMPLEMENTADOS

O procedimento permite registar um novo funcionário na base de dados.

DELIMITER &&

```
CREATE PROCEDURE register_new_employee (  
    IN e_id VARCHAR(10),  
    e_name VARCHAR(75),  
    vat VARCHAR(9),  
    bd DATE,  
    street VARCHAR(50),  
    locale VARCHAR(30),  
        postal VARCHAR(15),  
    manager VARCHAR(10),  
    phone VARCHAR(20),  
    email VARCHAR(75))
```

PROCEDIMENTOS IMPLEMENTADOS

BEGIN**START TRANSACTION;****INSERT INTO** Employee**VALUES** (e_id , e_name ,

vat , bd ,

street , locale ,

postal , manager);

IF ROW_COUNT() = 0 THEN**ROLLBACK;****END IF;****CALL** register_employee_email(e_id , email);

PROCEDIMENTOS IMPLEMENTADOS

```
IF ROW_COUNT() = 0 THEN
    ROLLBACK;
END IF;
CALL register_employee_phone(e_id , phone);
IF ROW_COUNT() = 0 THEN
    ROLLBACK;
END IF;
COMMIT;
END &&
```

PLANO DE SEGURANÇA E RECUPERAÇÃO DE DADOS

De forma a assegurar a possibilidade de Recuperação de Dados no casos de falhas, podemos utilizar o seguinte comando numa *shell* **POSIX** como a de sistemas operativos GNU-Linux/UNIX:

```
$ mysqldump -h localhost -u root -p mademoiselle_borges >\
$WORK_DIR/.backup/mademoiselle_backup.sql
```

Ou, em sistemas MS-DOS/Windows 3.x/Windows 9.x/Windows NT:

```
C:\MYSQL\BIN> MYSQLDUMP.EXE -u ROOT -p mademoiselle_borges ^
> %BACKUPDIR%\MADEMOISELLE_BACKUP.SQL
```

CONCLUSÕES E TRABALHO FUTURO

O desenvolvimento e implementação de uma base de dados relacional (em MySQL) envolveu uma abordagem sistemática, desde a definição do sistema até à sua implementação física. O trabalho iniciou-se com a definição do sistema, durante a qual procedemos à contextualização, fundamentação, estabelecimento de objetivos e avaliação de viabilidade, delineando de seguida um plano de execução e identificando a equipa de trabalho responsável. Estes passos forneceram uma base sólida para o desenvolvimento subsequente.

Na fase de definição de requisitos, optou-se por um método presencial para levantar e analisar as necessidades do cliente. Realizamos reuniões diretas com intuito de descobrir de forma eficaz as exigências e expectativas do mesmo, e procedemos a organização dos requisitos obtidos em categorias distintas - descrição, exploração e controlo.

CONCLUSÕES E TRABALHO FUTURO

A modelação conceptual proporcionou uma visão abstrata do sistema, destacando as entidades, os relacionamentos e os atributos. O diagrama ER resultante tornou-se uma ferramenta valiosa para visualizar a estrutura de uma forma clara e compreensível.

Ao avançarmos para a modelação lógica, a construção do modelo de dados, juntamente com a sua normalização, garantiu a eficiência e integridade do sistema. A implementação física e a subsequente criação de *Queries* e a definição de *Views* evidenciaram a aplicabilidade prática do modelo desenvolvido.

CONCLUSÕES E TRABALHO FUTURO

Para um trabalho futuro, será fundamental refletir sobre os desafios enfrentados neste projeto e identificar áreas específicas para aprimoramento. Considerando as áreas onde tivemos maiores dificuldades, como no controle de acesso no contexto *Grant* e *Revoke*, e nas funcionalidades de gestão e estatística, é imperativo traçar estratégias para melhorar estes pontos específicos. Ao refletirmos sobre estas dificuldades, esperamos, com base na experiência adquirida, estar mais preparados para enfrentar desafios futuros.

Mademoiselle Borges: Um Sistema de Bases
de Dados para Gestão de Eventos em
Eventopolis
Grupo 06
Bases de Dados

Bruno Gião
A96544

João Pereira
A95375

Helena Salazar
A75635

Tiago Teixeira
A97666

15 de janeiro de 2024