A blue and black logo

Description automatically generated

|  |
| --- |
| Complementos de Bases de Dados 2023/2024  Licenciatura em Engª. Informática |
| 1ª Fase Relatório Técnico |

Turma: 2

Horário de Laboratório: 10:30 ao 12:30

Docente: João Portelinha

Grupo

Nº202100728, André Rolo  
Nº202100744, Tomás Gonçalves

# Introdução

*No presente relatório tem como objetivo explicar o que o grupo com dois elementos (André Rolo e Tomás Gonçalves) fizeram no projeto de fase 1 da Unidade Curricular de Complementos de Bases de Dados. Neste Projeto elaborámos com lógica a criação de novas tabelas a uma base de dados (usámos a migração de dados para facilitar a criação das novas tabelas e as suas relações), onde foi fornecida tabelas já criadas em Excel. Tivemos de aplicar os conhecimentos em aula. Uma das matérias que aplicámos foi os metadados e outros conhecimentos adquiridos em outras disciplinas relacionadas a criação de base de dados (Sistemas de Gestão de Bases de Dados).*

*Nós esperamos que este projeto ajude-nos a melhorar os nossos conhecimentos de criação de tabelas eficientes e com um sistema de segurança, para isso declaramos funções e procedimentos, também usamos as declaramos de views, with e trigger, onde iremos explicar com mais detalhes no decorrer do relatório. Para além disso nosso objetivo é melhorar a capacidade de trabalhar em equipe e aprendermos a resolver problemas que possam aparecer no mundo do trabalho.*

# Especificação de Requisitos

*Nós ao vermos o enunciado dado pelo professor, foi nos propostos fazer uma base de dados em que os produtos podiam ter categorias e subcategorias, onde fomos ter um discussão em grupo de decido-mos declarar uma tabela para categorias e outra para subcategorias. Como pode ver no modelo relacional e no modelo Entidade Relação*

|  |  |  |
| --- | --- | --- |
| ID | Descrição | Implementado  (S/N) |
| *R01* | *Este R01 é o mesmo que o MER mas com a diferença de ter mais especificado as chaves primarias e estrangeiras mostrando as suas colunas.* | *S* |
| *MER01* | *Criamos um sistema que irá permitir adicionar clientes , produtos dar categorias aos produtos e nas categorias estão a usar subcategorias, assim sendo mais fácil procurar por um produto existente no banco de dados. Também foi adicionado o sendEmail para enviar um email ao cliente e assim mostrando possíveis alterações na sua conta (Ex.: Password alterada)* | *S* |

# Modelo Relacional (*Modelo de dados*)

## Diagrama do Modelo Entidade Relação

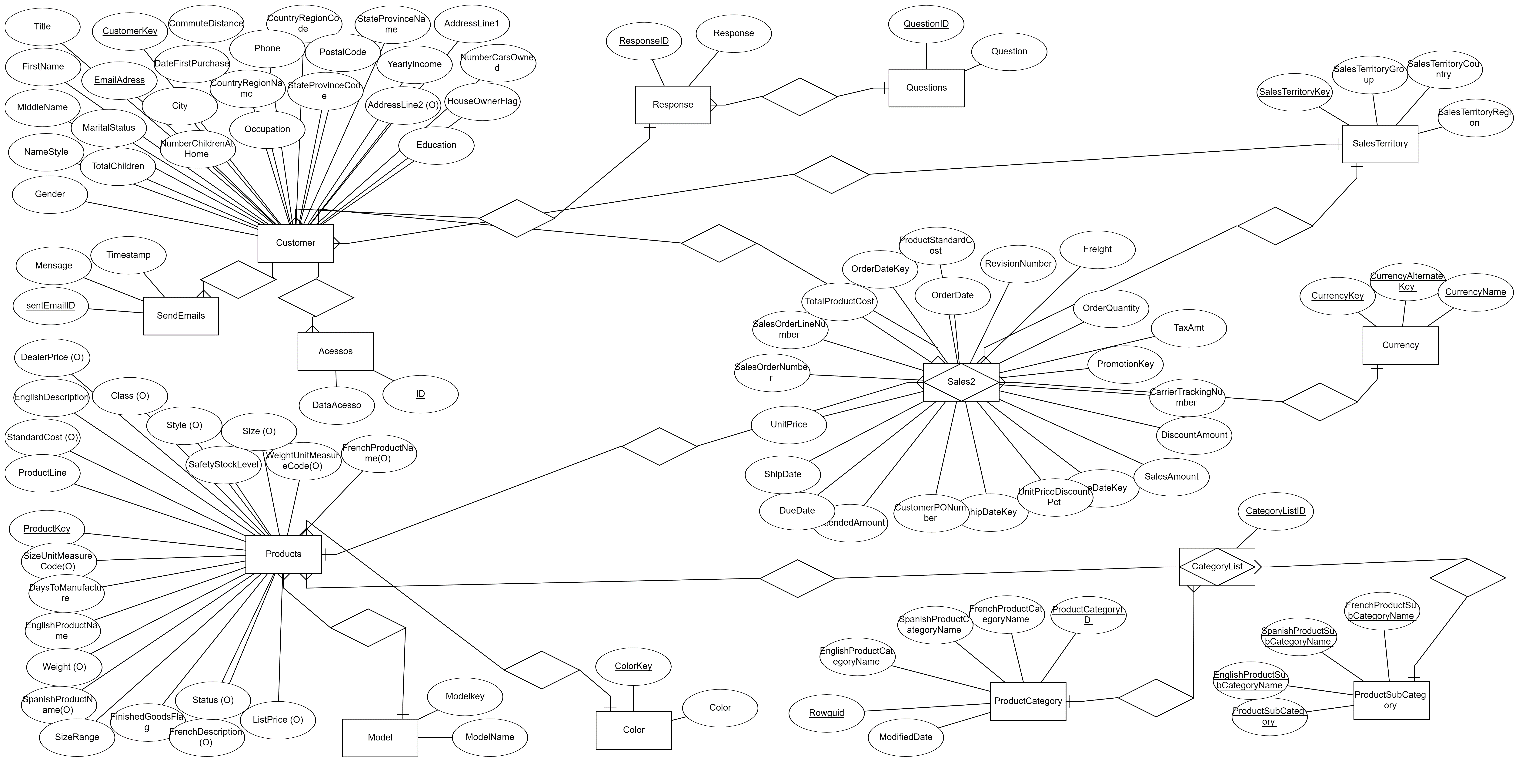
**

Figura 1 - Diagrama de Modelo Entidade Relação

## Diagrama do Modelo Relacional

Uma imagem com texto, captura de ecrã, diagrama, Paralelo

Descrição gerada automaticamente

Figura 2 - Modelo Relacional

# Definição do Layout

*No nosso Layout começamos por criar a possibilidade de ver os espaços de cada tabela dando a informação do numero de linhas e espaço que foi reservado para a tabela e seu espaço usado, para além de mostrar os espaço que esta a ser ocupado pelos índices da tabela (são partes para melhorar o desempenho), e mostro ainda o espaço que ainda pode ser usado. Ainda fazemos um select em que mostra os nomes de todas as tabelas e exibe o tamanho máximo, número de registos e espaço total. Com base no select de cima geramos uma tabela temporária onde ira mostrar a taxa de crescimento da tabela e ainda colocamos um gatilho em cada tabela para não crascar para quando se faz insert, update ou delete numa tabela.*

## Identificação do espaço ocupado por tabela

|  |  |  |
| --- | --- | --- |
| Nome Tabela | Dimensão do Registo | Nº de Registos (inicial/final) |
| *Acessos* |  | *0* |
| *CategoryList* |  | *37* |
| *Currency* |  | 105 |
| *Customer* |  | 18484 |
| *ErrorLog* |  | 9 |
| *EstatisticasTabelas* |  | 18 |
| *MetadadosTabelas* |  | 139 |
| *Model* |  | 119 |
| *ProductCategory* |  | 4 |
| *Products* |  | 397 |
| *ProductSubCategory* |  | 37 |
| *Questions* |  | 2 |
| *Response* |  | 1 |
| *Sales2* |  | 6072 |
| *SalesTerritory* |  | 11 |
| *SentEmails* |  | 65 |
| *sysdiagrams* |  | 1 |

## Especificação dos FIlegroups

|  |  |  |
| --- | --- | --- |
| Nome Filegroup | Tabelas associadas | Parâmetros |
| *PRIMARY* | *Sales2, Customer, ErrorLog, EstatisticasTabelas, MetadadosTabelas, Model, ProductCategory, Products, ProductSubCategory, Questions, Response, SalesTerritory, SentEmails* | *Nome, Localização de Ficheiros* |

## Schemas

|  |  |
| --- | --- |
| Nome | Descrição |
| *sp\_spaceused* | *Usado para obter rapidamente informações resumidas sobre o espaço ocupado por uma tabela. Ele fornece dados como o número de linhas, espaço total alocado, espaço usado para dados e índices, e a diferença entre o espaço reservado e o espaço realmente utilizado* |
| *sys.tables* | *Fornece uma lista de todas as tabelas no banco de dados, incluindo detalhes como o nome da tabela, a data de criação, a última data de modificação, e outras propriedades relacionadas às tabelas* |
| *sys.indexes* | *oferece uma lista de todos os índices no banco de dados, incluindo detalhes como o nome do índice, o tipo de índice, a tabela à qual o índice pertence, e informações sobre a fragmentação do índice* |
| *sys.partitions* | *fornece informações sobre as partições de tabelas e índices, incluindo detalhes como o nome do objeto, a função da partição, o tipo de índice, o número de linhas na partição e informações sobre o armazenamento de dados* |
| *sys.allocation\_units* | *oferece informações resumidas sobre como o espaço físico é alocado para objetos, como tabelas e índices, no banco de dados. Ela inclui detalhes sobre o tipo de alocação, objeto associado, quantidade de espaço alocado e localização física* |
| *sys.data\_spaces* | *oferece informações resumidas sobre a organização de espaços de dados no banco. Fornece dados como o nome do espaço de dados, tipo (filegroup ou partition scheme), e local de armazenamento* |
| *sys.filegroups* | *fornece informações resumidas sobre os filegroups no banco de dados, incluindo detalhes como nome, tipo e localização física* |
| *sys.database\_files* | *fornece informações resumidas sobre os arquivos físicos de um banco de dados, incluindo detalhes como nome, tipo, tamanho e localização.* |
| *sys.columns* | *fornece informações resumidas sobre as colunas em tabelas e exibições, incluindo nome, tipo de dados e outras propriedades.* |
| *sys.default\_constraints* | *fornece informações sobre restrições padrão em colunas de tabelas, incluindo nome da restrição, tabela, coluna e o valor padrão.* |
| *sys.types* | *fornece informações resumidas sobre os tipos de dados disponíveis no banco de dados, incluindo tipos padrão e definidos pelo utilizador.* |

# Verificação da migração de dados

## Consultas sobre a base de dados original

## Consultas sobre a nova base de dados

# Programação

## Views

|  |  |
| --- | --- |
| Nome | Descrição |
| *CustomerDetailsView* | *Esta view permite obter a lista de clientes* |
| *CustomerPurchasesView* | *Esta View permite obter a lista de os produtos que os clientes compraram* |
| *HistoricoComprasCliente* | *Esta View permite obter a lista de todos os clientes que fizeram compras* |
| *VendasPorTerritorioView* | *Esta Views permite obter a lista de locais onde foram mais compradas* |

## Functions

|  |  |  |  |
| --- | --- | --- | --- |
| Nome | Atributos | Requisito | Descrição |
| *dbo.udf\_getUtilizador* | *@id\_user INT* | *R0#* | *Permite obter informação detalhada sobre um utilizador* |
|  |  |  |  |

## Stored procedures

|  |  |  |  |
| --- | --- | --- | --- |
| Nome | Atributos | Requisito | Descrição |
| *dbo.setUtilizador* | *@nome VARCHAR(50)* | *R0#* | *Permite adicionar um novo utilizador* |
|  |  |  |  |

## Triggers

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome | Tipo | Tabela | Requisito | Descrição |
| *dbo.tr\_utlizador\_historico* | *AFTER UPDATE* | *dbo.utilizador* | *R0#* | *Guarda o histórico de alterações sobre o utilizador* |
|  |  |  |  |  |

# Catálogo/Metadados

## Monitorização

|  |  |  |
| --- | --- | --- |
| Nome | Atributos | Descrição |
|  |  |  |
|  |  |  |

# Descrição da Demonstração

## Script de demonstração

*Sequência de execução de código (scripts, consultas, chamadas a SPs,…) que permita verificar o correto funcionamento da base de dados face aos requisitos.*

# Conclusões