

**Relatório de Projeto – Fase 1**

Curso Técnico Superior Profissional em Tecnologias Informáticas

Alex Fernandes Torres

Gabriel Asafe de Jesus Martins

Tomás Oliveira Máximo

Duarte Bernardo Pedrosa

Leiria, março de 2023



**Relatório de Projeto – Fase 1**

Curso Técnico Superior Profissional em Tecnologias Informáticas

Alex Fernandes Torres

Gabriel Asafe de Jesus Martins

Tomás Oliveira Máximo

Duarte Bernardo Pedrosa

Projeto realizado sob a orientação do Professor Nuno Lopes Santos e sob supervisão do mesmo.

Leiria, março de 20

Contents

[Resumo 4](#_Toc130800711)

[Plano de Projeto 5](#_Toc130800712)

[Requisitos do Sistema 5](#_Toc130800713)

[Dados do Projeto 6](#_Toc130800714)

[Justificativa de algumas ligações entre entidades 6](#_Toc130800715)

[Entidades e Atributos 7](#_Toc130800716)

[Tablespaces 11](#_Toc130800717)

[Conclusão 12](#_Toc130800718)

# Resumo

O seguinte relatório é referente à primeira fase do projeto de Complementos de Base de Dados, implementado no segundo semestre do TESP TI no âmbito da disciplina de Complementos de Base de Dados, e visa ser o ponto inicial do processo de desenvolvimento de uma base de dados pelos alunos.

Os objetivos deste trabalho eram a criação dos modelos de entidade e relacionamento; A criação do modelo físico de dados; Criação dos tablespaces e por fim o DDL para a implementação das tabelas.

Se tratando de uma fase inicial de desenho e pré implementação, pudemos debater a melhor implementação com base em nossos conhecimentos, aspirando tornar a modelagem inicial a mais completa possível, evitando que no futuro, caso seja necessária alteração estrutural da base de dados, cause o mínimo possível de transtornos para a segunda fase.

# Plano de Projeto

O projeto visa a elaboração da base de dados de um banco, passando por todas as etapas de desenvolvimento, desde o planeamento à implementação, considerando suas agências, funcionários, clientes e respetivas contas e transações.

O trabalho contará com a criação do modelo de entidade de relacionamento, visando auxiliar a depuração futura do banco de dados relacional. Também será criado o modelo de dados físico com scripts de estruturas de dados, juntamente com a definição dos objetos necessários para garantir a integridade referencial do modelo de dados proposto. Será feito, também a criação da estrutura do tablespace juntamente com o datafile usado e a criação de scripts de DDL para criação e teste preliminar do modelo de dados de suporte à aplicação.

O projeto foi implementado pelos alunos Tomás Máximo, responsável pela elaboração do relatório do projeto; Alex Fernandes e Gabriel Martins, responsáveis pelo desenho do sistema, criação das tabelas e inserção dos dados para teste. Mesmo que cada membro fosse responsável por alguma tarefa, a elaboração contou com o auxílio de todos simultaneamente.

O cronograma inicial foi pensado de maneira que fosse semanalmente pensado em conjunto após as aulas, de maneira que houvesse uma melhor assimilação do conteúdo letivo, visando uma melhor implementação do projeto. Na primeira semana foi desenhado o modelo de entidades e relacionamentos. Na segunda foi feito o modelo de dados. Na terceira foi feita a inserção de dados de teste na tabela e por fim na última foi feito o relatório.

# Dados do Projeto

Aqui, apresentar-se-ão todas as decisões tomadas em relação a criação de tabelas, atributos e tablespaces, e explicar-se-á o porquê de se ter criado certos elementos, como algumas tabelas e tablespaces.

## Criação de tabelas

Vamos estar justificando a criação de algumas ligações e tabelas intermédias que fizemos com base na nossa interpretação do enunciado e não constam no diagrama de entidades e relacionamento.

**Transação**: Faz uma disjunção para transferência e movimento e com base no enunciado proposto acreditamos que deveria se ser dessa maneira pois as transferências precisam receber a conta de origem e a de destino, diferente de outras operações como um levantamento ou depósito.

**Conta Cliente e Conta Produto**: Nascem de uma ligação de muitos para muitos e não foi referenciada no modelo de entidade e relacionamento, mas nascem no modelo físico.

**Conta**: Apresenta uma disjunção pois segundo nossa interpretação do enunciado, só existem dois tipos de conta, sendo a conta a ordem a única que possuía cartões de crédito. Entretanto ambas tinha a possibilidade de ter produtos associados a elas.

## 

## Entidades e Atributos

Cada tabela tem atributos que a compõem, sendo eles:

FUNCIONARIO:

NB\_NFuncionario: É um atributo do tipo number, que é gerado automaticamente, começando em 1000 e aumentando de 1 em 1, representando o número único de cada funcionário, sendo a chave primária da tabela;

VC\_Gerente: É um atributo do tipo varchar2, que verifica se o funcionário tem a função de gerente ou não, tendo uma constraint que limita as escolhas a ‘S’ ou ‘N’;

NB\_AGENCIA\_NB\_NAgencia: É um atributo do tipo number, e representa o número da agencia a que cada funcionário pertence, sendo também a chave estrangeira da tabela agencia;

AGENCIA:

NB\_NAgencia: É um atributo do tipo number, que é gerado automaticamente, começando em 1000 e aumentando de 1 em 1, representando o número único de cada agência, sendo a chave primária da tabela;

NB\_ENDERECO\_NEndereco: É um atributo do tipo number, que representa o endereço de cada agência, sendo também a chave estrangeira da tabela endereço;

ENDERECO:

NB\_NEndereco: É um atributo do tipo number, que é gerado automaticamente, começando em 1000 e aumentando de 1 em 1, representando o número único de cada endereço, sendo a chave primária da tabela;

VC\_Cidade: É um atributo do tipo varchar2, que representa a cidade onde se situa esse endereço.

VC Ruã: É um atributo do tipo varchar2, que representa a rua onde se situa o endereço;

NB\_CodPos: É um atributo do tipo number, que representa o código postal onde se situa o endereço;

NB\_NumPorta: É um atributo do tipo number, que representa o número da porta do endereço;

CLIENTE:

NB\_iban: É um atributo do tipo number, que é gerado automaticamente, começando em 1000000000000000 e aumentando de 1 em 1, representando o número único de cada funcionário, sendo a chave primária da tabela;

DT\_dataNasc: É um atributo do tipo date e representa a data de nascimento do cliente;

VC\_profissao: É um atributo do tipo varchar2, e representa a profissão do cliente;

VC\_nomeCliente: É um atributo do tipo varchar2, e representa o nome do cliente;

NB\_nif: É um atributo do tipo number, e representa o NIF do cliente;

VC\_email: É um atributo do tipo varchar2, e representa o email do cliente;

VC\_password: É um atributo do tipo varchar2, e representa a password do cliente;

NB\_idade: É um atributo do tipo number, e representa a idade do cliente;

NB\_ENDERECO\_NEndereco: É um atributo do tipo number, que representa o endereço do cliente, sendo a chave estrangeira da tabela endereco;

NB\_AGENCIA\_NB\_AGENCIA: É um atributo do tipo number, que representa a que agência o cliente está associado, sendo a chave estrangeira da tabela agência;

CONTA:

NB\_numeroConta: É um atributo do tipo number, que é gerado automaticamente, começando em 1000000000000000 e aumentando de 1 em 1, representamdo o número único de cada conta, sendo a chave primária da tabela;

NB\_saldo: É um atributo do tipo number, e representa o saldo de cada conta;´

NB\_titular: É um atributo do tipo number, e representa o IBAN do cliente que é o primeiro titular da conta;

ORDEM:

NB\_CONTA\_numeroConta: É um atributo do tipo number, que representa o número da conta à ordem, sendo não só a chave primária da tabela, mas também, a chave estrangeira da tabela conta;

PRAZO:

NB\_CONTA\_numeroConta: É um atributo do tipo number, que representa o número da conta à ordem, sendo não só a chave primária da tabela, mas também, a chave estrangeira da tabela conta;

CARTAO:

NB\_numero: É um atributo do tipo number, que é gerado automaticamente, começando em 1000 e aumentando de 1 em 1, representando o número único de cada cartão, sendo a chave primária da tabela;

NB\_ORDEM\_CONTA\_numeroConta: É um atributo do tipo number, que representa o número da conta à qual o cartão está associado, sendo a chave estrangeira da tabela ordem;

NV\_cvv: É um atributo do tipo number, e representa o CVV de cada cartão;

NB\_pin: É um atributo do tipo number, e representa o código pin de cada cartão;

VC\_validade: É um atributo do tipo varchar2, e representa a validade do cartão;

PRODUTO:

NB\_numProduto: É um atributo do tipo number, que é gerado automaticamente, começando em 1000 e aumentando de 1 em 1, representamdo o número único de cada produto, sendo a chave primária da tabela;

VC\_tipoProduto: É um atributo do tipo varchar2, e representa o tipo de cada produto

TRANSACAO:

NB\_numeroTransacao: É um atributo do tipo number, que é gerado automaticamente, começando em 1000 e aumentando de 1 em 1, representamdo o número único de cada transação, sendo a chave primária da tabela;

VC\_plataforma: É um atributo do tipo varchar2, que representa em que plataforma em que foi feita a transação, tendo uma constraint que limita as opções a ‘ATM’ e ‘WEB’;

VC\_CATEGORIA\_abrev\_categoria: É um atributo do tipo varchar2, que representa a categoria da transação que foi feita, sendo também a chave estrangeira da tabela categoria;

NB\_CONTACLIENTE\_Cliente\_IBAN: É um atributo do tipo number, que representa o IBAN do cliente que realizou a transação, sendo em conjunto com o atributo CONTACLIENTE\_CONTA\_numeroConta, a chave estangeira da tabela CONTACLIENTE.

NBCONTACLIENTE\_CONTA\_numeroConta: É um atributo do tipo number, que representa o número da conta do cliente que realizou a transação, sendo em conjunto com o atributo CONTACLIENTE\_Cliente\_IBAN, a chave estangeira da tabela CONTACLIENTE.

NB\_valor: É um atributo do tipo number, que representa o valor da transação feita;

CATEGORIA:

VC\_abrev\_categoria: É um atributo do tipo varchar2, e representa a abreviatura de cada categoria, sendo a chave primária da tabela;

VC\_nome\_categoria: É um atributo do tipo varchar2, e representa o nome de cada categoria;

TRANSFERENCIA:

NB\_TRANSACAO\_numeroTransacao: É um atributo do tipo number, e representa o número da transação associada à transferência, sendo não só a chave primária da tabela, mas também a chave estrangeira da tabela transacao;

NB\_CONTA\_numeroConta: É um atributo do tipo number, que representa o número da conta associada à transferência feita, sendo a chave estrangeira da tabela conta;

MOVIMENTO:

NB\_TRANSACAO\_numeroTransacao: É um atributo do tipo number, e representa o número da transação associada à transferência, sendo não só a chave primária da tabela, mas também a chave estrangeira da tabela transacao;

VC\_operacao: É um atributo do tipo varchar2, que justifica se o movimento foi um depósito ou um levantamento, tendo uma constraint que limita as escolhas a ‘L’ ou ‘D’;

CONTACLIENTE:

NB\_CLIENTE\_IBAN: É um atributo do tipo number, que representa o IBAN do cliente associado, sendo a chave estrangeira da tabela cliente, e, em conjunto com o atributo NB\_CONTA\_numeroConta, é a chave primária da tabela;

NB\_CONTA\_numeroConta: É um atributo do tipo number, que representa o número da conta que está associada, sendo a chave estrangeira da tabela conta, e, em conjunto com o atributo NB\_CLIENTE\_IBAN, é a chave primária da tabela;

PRODUTOCONTA:

NB\_PRODUTO\_NB\_numProduto: É um atributo do tipo number, que representa o número do produto associado, sendo não só a chave estrangeira da tabela PRODUTO, como também, em conjunto com o atributo NB\_CONTA\_numeroConta, a chave primária da tabela;

NB\_CONTA\_numeroConta: É um atributo do tipo number, que representa o número da conta associada, sendo não só a chave estrangeira da tabela CONTA, como também, em conjunto com o atributo NB\_PRODUTO\_NB\_numProduto, a chave primária da tabela;

# TableSpaces

Para a criação do tablespace fizemos a estimativa com 30 funcionários pensando que cada agência teria pelo menos três deles. Estimamos 10 agências, 60 endereços, 50 clientes e cada um com duas contas com pelo menos um produto e dois cartões de credito e com cada um fazendo duas transferências

Tendo todos esses dados inseridos na base de dados, estimamos uma média de 4 Mega Bytes para os acomodar todos. Por motivos pessoais, sempre usamos números ímpares, então ficamos com 5 Mega Bytes. Os cálculos feitos estão descritos a seguir.

Funcionário = 30 \* (15) = 450

Agência = 10 \* (14) = 140

Cliente = 50 \* (99) = 4950

Conta Cliente = 50 \* 2 \* (32) = 3200

Endereços = 60 \* (46) = 2760

Transações do tipo transferência com uma categoria = 50 \* 2 \* (95) = 9500

Todas as contas à ordem têm dois cartões = 50 \* 2 \* (51) = 5100

Conta tem pelo menos um produto, e estão referenciados em produto conta = 50 \* (92) = 4600

Ao somarmos os valores, dividi-los por 8 e posteriormente por 1024 teremos 3,8 M

Em seguida arredondamos o resultado para 4 e subimos para 5, para criarmos uma margem de segurança, caso no futuro seja necessário adicionar mais dados do que o previsto.

# Conclusão

Esta primeira fase do projeto é, sem dúvidas, a parte mais importante por contar com o desenho inicial do sistema. Por ser a base de todo nosso trabalho futuro, é importante que toda interpretação das nuances que esse trabalho apresenta tenham sido bem interpretadas e isso requer experiência para que não ocorram erros grandes o suficiente para representarem uma alteração estrutural drástica.

Entretanto acreditamos que os resultados obtidos inicialmente vão ser bons e serão suficientes para iniciar uma segunda fase livre de problemas.

As demais questões que surgiram no decorrer do projeto foram sanadas com a comunicação em equipe, outro aspeto fundamental para o bom desenvolvimento de um trabalho em grupo.