



Relatório de Projeto – Fase 1

Curso Técnico Superior Profissional em Tecnologias Informáticas

Alex Fernandes Torres

Gabriel Asafe de Jesus Martins

Tomás Oliveira Máximo

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

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

Leiria, março de 2023

Índice

Resumo.....	4
Plano de Projeto	5
Dados do Projeto	6
Entidades e Atributos	6
TableSpaces	12
Matriz de Requisitos e Plano de Tarefas	13
Procedures e Functions.....	14
Procedures	14
Functions.....	15
Conclusão.....	16

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.

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_Nome: É um atributo do tipo varchar2, que guarda o nome do funcionário;

DTA_dtaNascimento_ É um atributo do tipo date, que guarda a data de nascimento do funcionário.

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;

NB_FUNCIONARIO_NB_NSuperior: É um atributo do tipo do tipo number que representa o número do funcionário superior ao funcionário em questão;

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;

NB_FUNCIONARIO_NB_NGerente: É um atributo do tipo number, que representa o funcionário que é o gerente da agência em questão;

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

NB_CodPostal: 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;

VC_Conselho: É um atributo do tipo varchar2, que representa o conselho em que se situa o endereço;

VC_Distrito: É um atributo do tipo varchar2, que representa o distrito em que se situa o endereço;

VC_Pais: É um atributo do tipo varchar2, que representa o país em que se situa o endereço;

CLIENTE:

NB_NCliente: É um atributo do tipo number, que é gerado automaticamente, representa o número único de cada funcionário, sendo a chave primária da tabela;

DT_DataNascimento: 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_Nome: 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_AGENCIA_NB_NAgencia: É um atributo do tipo number, que representa a que agência o cliente está associado, sendo a chave estrangeira da tabela agência;

CONTA:

NB_IBAN: É um atributo do tipo number, que é gerado automaticamente, começando em 10000000000000000 e aumentando de 1 em 1, representando o número único de cada conta, o IBAN, sendo a chave primária da tabela;

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

NB_AGENCIA_NB_NAgencia: É um atributo do tipo number que representa a agência à qual a conta está associada, sendo também a chave estrangeira da tabela AGENCIA;

NB_Tipo: É um atributo do tipo number, que representa o tipo de conta que é, podendo ser a prazo ou à ordem;

NB_PRODUTO_NB_numPorduto: É um atributo do tipo number que representa que produto está associado à conta, sendo também a chave estrangeira da tabela PRODUTO;

CARTAO:

NB_numeroCartao: 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;

NB_TITULAR_NB_IBAN: É um atributo do tipo number, que representa o IBAN da conta à qual o cartão está associado, sendo, em conjunto com o atributo NB_TITULAR_NB_NCliente, a chave estrangeira da tabela TITULAR;

NB_TITULAR_NB_NCliente: É um atributo do tipo number, que representa o cliente que é dono do cartão, sendo, em conjunto com o atributo NB_TITULAR_NB_IBAN, a chave estrangeira da tabela TITULAR;

PRODUTO:

NB_numProduto: 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 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, representando 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’;

NB_categoria: É um atributo do tipo number, que representa a categoria da transação que foi feita, sendo também a chave estrangeira da tabela categoria;

NB_operacao: É um atributo do tipo number, que representa o número da operação associado à transação, sendo também a chave estrangeira da tabela OPERACAO;

NB_TITULAR_NB_NCliente: É um atributo do tipo number, que representa o número do cliente associado à transação em questão, sendo também a chave estrangeira da tabela TITULAR, em conjunto com o atributo NB_TITULAR_NB_IBAN;

NB_TITULAR_NB_IBAN: É um atributo do tipo number, que representa o IBAN da conta associada à transação, sendo também a chave estrangeira da tabela TITULAR, em conjunto com o atributo NB_TITULAR_NB_NCliente;

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

DTA_dataTransacao: É um atributo do tipo date, que representa a data em que foi efetuada a transação;

NB_CONTA_NB_IBANRecetor: É um atributo do tipo number, que representa o IBAN da conta recetora da transação, sendo a chave estrangeira da tabela CONTA;

CATEGORIA:

NB_categoria: É um atributo do tipo number, e representa o número 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;

MORADACLIENTE:

NB_CLIENTE_NB_NumeroCliente: É um atributo do tipo number, que representa o número do cliente associado ao endereço, sendo também a chave primária da tabela, em conjunto com o atributo NB_ENDERECO_nb_NEndereco, e a chave estrangeira da tabela CLIENTE;

NB_ENDERECO_nb_NEndereco: É um atributo do tipo number que representa o número do endereço que está associado ao cliente, sendo a chave primária da tabela, em conjunto com o atributo NB_CLIENTE_NB_NCliente, e a chave estrangeira da tabela ENDERECO;

CodPostal_Ext – É uma tabela standalone que não tem relação com nenhuma tabela, servindo apenas como tabela de verificação:

NB_CodPostal: É um atributo do tipo number, que representa o código postal de um endereço;

VC_Cidade: É um atributo do tipo varchar2, que representa a cidade de um endereço;

TITULAR:

CONTA_NB_IBAN: É um atributo do tipo number que representa o IBAN da conta associada ao titular em questão;

CLIENTE_NB_NCliente: É um atributo do tipo number que representa o número do cliente que é titular da conta em questão;

DTA_dtaRegistro: É um atributo do tipo date que representa a data na qual o titular foi associado à conta;

VC_OrdemTitularidade: É um atributo do tipo varchar2, que representa a ordem de titularidade do cliente na conta em questão;

OPERACAO:

NB_Operacao: É um atributo do tipo number que representa o número da operação, sendo a chave primária da tabela;

VC_nome_operacao: É um atributo do tipo varchar2, que representa o nome da operação;

TableSpaces

Foram criados 4 tablespaces, dividindo as tabelas entre si da seguinte forma:

- **transacao:** Este tablespace foi criado especificamente para armazenar os dados das transações, tendo 350M de espaço e um *AUTOEXTEND ON NEXT 350M*;
- **PAC_gerais:** Este tablespace foi criado para armazenar os dados das tabelas: produto, agência e categoria, tendo 1M de espaço e um *AUTOEXTEND ON NEXT 1M*;
- **FE3C_gerais:** Este tablespace foi criado para armazenar os dados das tabelas: funcionários, cliente, conta, endereço e cartão, tendo 72M de espaço e um *AUTOEXTEND ON NEXT 72M*;
- **otm_gerais:** Este tablespace foi criado para armazenar os dados das tabelas: operação, titular e moradacliente, tendo 1M de espaço e um *AUTOEXTEND ON NEXT 1M*;

cidadeexternal: Este tablespace foi criado para guardar os dados da external table, tendo 100M de espaço e um *AUTOEXTEND ON NEXT 100M*.

Matriz de Requisitos e Plano de Tarefas

Os dados da matriz de requisitos estão disponíveis para análise no ficheiro de nome [matriz_de_Requesítosl.xlsx](#).

Os dados do plano de tarefas estão disponíveis para análise no ficheiro de nome [plano_de_tarefas.xlsx](#).

Procedures e Functions

Procedures

No âmbito deste projeto, foram realizados diversos Procedures, sendo estes:

- **INSERT_LOGLOG:** Este procedimento adiciona um registo de erro na tabela de logs quando ocorre algum erro;
- **DELETE_SPECIFIC_REGISTER:** Este procedimento recebe o nome da tabela e o id do item a ser apagado e apaga-o da base de dados, utilizando a tabela de metadados;
- **UPDATE_SPECIFIC_REGISTER:** Este procedimento recebe o nome da tabela, o id do item a ser atualizado, o campo que será atualizado e o novo valor, e atualiza-o utilizando a tabela de metadados;
- **SP_ENDERECO_INSERT:** Este procedimento insere um endereço na tabela ENDERECO;
- **SP_CONTA_INSERT:** Este procedimento insere uma conta na tabela CONTA;
- **SP_TITULAR_INSERT:** Este procedimento tem como objetivo associar uma conta a um cliente;
- **SP_CLIENTE_INSERT:** Este procedimento insere um cliente na tabela CLIENTE;
- **SP_CARTAO_INSERT:** Este procedimento insere um cartão na tabela CARTAO;
- **SP_FUNCIONARIO_INSERT:** Este procedimento insere um funcionário na tabela FUNCIONARIO;
- **SP_TRANSACAO_ALEATORIO:** Este procedimento adiciona um número recebido por parâmetro de registos aleatórios na tabela TRANSACAO;
- **SP_AGENCIA_INSERT:** Este procedimento adiciona um endereço na tabela ENDERECO, para além de adicionar uma agencia na tabela AGENCIA;
- **SP_OPERACAO_INSERT:** Este procedimento, insere dados na tabela OPERACAO;
- **SP_PRODUTO_INSERT:** Este procedimento insere dados na tabela PRODUTO;
- **SP_CATEGORIA_INSERT:** Este procedimento insere dados na tabela CATEGORIA;

Functions

No âmbito deste projeto, foram realizadas diversas Functions, sendo estas:

- tamanhoNumero: Esta função devolve o tamanho de uma String enviada por parâmetro;
- titularOrdem: Esta função manipula a ordem de titularidade de uma conta;
- numeroaleatorio: Esta função retorna um número aleatório entre um número mínimo e um número máximo, entregues por parâmetro.

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 aspecto fundamental para o bom desenvolvimento de um trabalho em grupo.