

TRABALHO FINAL DE CURSO

tmn

*DESENVOLVIMENTO DE UMA
APLICAÇÃO INTERNA PARA A
PARAMETRIZAÇÃO DE
DESCONTOS SELECTIVOS A
CLIENTES EMPRESARIAIS*



Lisboa, 25 de Julho de 2007

Silvana Rute C. Raimundo

AGRADECIMENTOS

Gostaria de agradecer à TMN pela possibilidade de me deixar apresentar um trabalho feito em parceria com a Universidade Lusófona Humanidades e Tecnologias.

Quero Agradecer em particular à minha Chefia Eng^a Isabel Barbosa e ao meu Director Eng^o Eugénio Alves da Silva por todo o apoio e disponibilidade demonstrada no acompanhamento da realização do projecto.

O meu agradecimento em especial para o meu Orientador, o **Professor Pedro Malta**, pela sua disponibilidade, ajuda, paciência e aconselhamento.

Aos restantes professores orientadores da cadeira de Projecto.

Aos Professores do Curso, pelos métodos e matérias ensinadas, mas em especial o **Professor António Frazão**, que me ensinou a programar com linguagem OO, tão importante neste projecto, ao **Professor Tiago Barreiros** porque sem ele não teria aprendido nunca Matemática, e por fim ao **Professor Alexandre Pereira** pelos seus ensinamentos em Java e pela paciência demonstrada comigo em tantas outras cadeiras que tive com ele.

Aos colegas dos anos de 99 até agora ...

Ao **meu Marido**, pelo espírito critico, pelas saídas tardias da TMN e pelos muitos fins-de-semana que nos foram “roubados”.

E principalmente, aos **meus Pais**, pelo incentivo e porque finalmente me vêem a terminar, o que há tanto tempo se prolongava!

A Todos o meu Muito Obrigada,

Silvana Rute C. Raimundo

Lisboa, 25 de Julho de 2007

Descrição de Termos Técnicos:

DDSI – Direcção Desenvolvimentos Sistemas Informação

BD – Base Dados

DS – Descontos Selectivos

Mobilix – Base de Dados que contem todos os registos de um determinado cliente, desde nome, morada, tipo tarifário, nº conta, nº contribuinte, etc.

Msisdn – Número de 11 dígitos que identifica um cartão.

NIF – Número de Identificação Fiscal

Abstract

Until the present time the application that allow applying discounts on customers accounts was very limited. It allows many human mistakes and many complain of the customers.

Not only had those factors contributed to the development of this new application as also the need to go on the encountered of the needs of the customer manager. The actual process was to slow and not very effective. The actual process generates around 500 e-mails per month and it was growing.

For all of this, it becomes urgent the development of a new application. Do to competition of the market, its mandatory to give, in real time, answer to the customer needs.

In the guiding line of the company, TMN. – Telecomunicações Móveis Nacionais, this new application has to be developed in Web based interface.

Do to that the application was developed in Java language, regarding the interface of the company's local intranet, based on web interface. This interface allows, any of the customer managers, to access the Client information, any time and any place.

All of the parts of this paper are confidential. It is no allow showing or giving part, or all, of the application, as also the paper, to any one with out TMN – Telecomunicações Móveis Nacionais, rotten authorization.

ÍNDICE

1.	Introdução.....	1
2.	Enquadramento Organizacional do Projecto	2
3.	Análise de Requisitos	3
4.	Sistemas de Informação	5
4.1	Requisitos dos Sistemas de Informação	5
4.2	Interface <i>Mobilix</i> vs. Descontos Selectivos.....	6
5.	Arquitectura da Aplicação.....	7
5.1	Organização dos <i>Packages</i>	7
5.2	Estrutura do Ficheiro EAR.....	7
5.3	Arquitectura.....	8
6.	Migração dos Dados	9
7.	Descrição do Modelo Físico de Dados.....	10
7.1	Descrição das Tabelas	10
8.	Testes.....	18
9.	Formação aos Gestores e Manutenção da Aplicação	19
9.1	Formação	19
9.2	Manutenção da Aplicação	19
10.	Conclusão	20
	ANEXOS.....	21
	Anexo 1 – Schema de Criação da Nova BD – Descontos Selectivos	22
	Anexo 2 – Modelo Físico de Dados	32
	Anexo 3 – Modelo Relatório de Testes TMN – Mod_B0657_Rel_Testes	33
	Anexo 4 – Manual do Utilizador.....	34

1. Introdução

Até á presente data, a aplicação existente de descontos selectivos era rudimentar, limitada e sobretudo permitia uma incorrecta introdução de descontos a Clientes.

Não só estes factores contibuíram para que se desenvolvesse uma nova aplicação, como e o factor mais importante foi, satisfazer as necessidades dos Gestores, uma vez que o procedimento que estava em vigor, era moroso e ineficaz, para que este projecto avançasse o facto do volume de mails que mensalmente foi crescendo a um ritmo vertiginoso (cerca de 500 *mails* mensais), também foi um aspecto a ter em linha de conta.

Pelo que se tornou urgente, o desenvolvimento de uma nova aplicação para a atribuição de descontos selectivos - em ambiente *Web* - uma vez que face à concorrência no mercado das Telecomunicações, as necessidade dos gestores de conta terem uma resposta imediata para fornecer aos clientes é uma prioridade da Empresa.

Assim sendo, foi criada uma nova aplicação, desenvolvida em Java, tendo como interface gráfico a intranet local da empresa (ambiente *web*), onde é acedida a qualquer dia e a qualquer hora pelas pessoas que dela necessitam para o seu trabalho diário.

A informação contida neste documento é de caracter Confidencial, pelo que agradeço a não divulgação sem ordem expressa da TMN – Telecomunicações Movéis Nacionais.

2. Enquadramento Organizacional do Projecto

A **TMN – Telecomunicações Móveis Nacionais**, é a maior empresa a nível nacional de telecomunicações móveis. Detém actualmente, o maior número de clientes dos 3 operadores existentes no mercado. Pertence a um dos maiores Grupos Económicos Portugueses, e que representa uma grande fatia da economia de Portugal – a **Portugal Telecom**.

Tendo em conta o que foi referido atrás, a Direcção Desenvolvimentos Sistemas Informação (DDSI), e que é responsável por todos os desenvolvimentos na TMN, foi confrontada pelas Direcções Clientes, para que se desenvolvesse uma Nova Aplicação de Descontos Selectivos, que permitisse a introdução, alteração e remoção, de descontos da Carteira de Clientes de determinado Gestor.

Contudo, e até este momento, o processo era feito manualmente e da seguinte forma:

- ✓ Envio de um mail (de dia 1 até dia 20 do mês) para uma *mailbox* existente para o efeito;
- ✓ Parametrização por parte da DDSI dos respectivos descontos, solicitados pelos Gestores de Clientes;
- ✓ Mail de resposta com o que tinha sido efectuado.

Todo este processo era moroso, para além da alocação exclusiva de pelo menos 1 recurso humano para tais funções, ao longo do mês.

Assim sendo, e com a concordância das Direcções Cliente, da Direcção desenvolvimentos sistemas de informação e da Universidade Lusófona, foi criada uma nova aplicação, para que cada gestor possa aceder e ser responsável por todas as alterações que sejam feitas na sua carteira de clientes.

3. Análise de Requisitos

De seguida é apresentada a Análise de Requisitos, e a viabilidade de implementação dos mesmos. Foram solicitados os seguintes requisitos, para a Nova Aplicação de Descontos Selectivos pelas áreas comerciais da empresa:

1. Possibilidade de inserir, alterar ou apagar um conjunto de descontos, ou um só desconto (iguais ou diferentes), através de um ficheiro, que seria automaticamente inserido na Base de Dados de Descontos Selectivos;
2. Apagar todos os descontos existentes de um Contribuinte, independentemente de eles se encontrarem atribuídos ao contribuinte, à conta, ou ao telefone;
3. Inserir, alterar ou apagar um tipo desconto (existente na BD) para todas as contas de um contribuinte (por exemplo, apagar o código 3 para todas as contas do contribuinte x);
4. Permitir através de um template a introdução, alteração ou remoção dos descontos;
5. A Aplicação deverá reportar através de ficheiros Excel os descontos existentes num determinado contribuinte/conta/Telefone.
6. A Aplicação deverá lançar um alerta sempre que existam inconformidades, isto é, sempre que exista um contribuinte inválido, ou uma conta inválida, ou um Telefone errado a aplicação deverá lançar uma mensagem de alerta, avisando assim a pessoa que se encontra a parametrizar os descontos;
7. Apenas um *key user* autorizado poderá inserir descontos de 100%, os outros utilizadores, deverão remeter todos estes pedidos, acompanhados da devida autorização, para ele;
8. Um cliente não pode ter descontos por conta, por contribuinte e por telefone em simultâneo, a não ser que se possa definir uma regra hierárquica na atribuição do desconto, ou seja, se existirem descontos por contribuinte, esse desconto é aplicado a todas as contas e telefones que não tenham descontos selectivos definidos;

9. Validar o desconto introduzido, validar a existência das contas, dos Telefones e dos Nifs na BD;
10. Registar o utilizador que realizou qualquer operação na aplicação (*logs*);
11. A Aplicação deverá possibilitar a inclusão de Descontos Selectivos ao Telefone, validando sempre a existência de descontos na Conta ou no Nif. A atribuição de descontos ao telefone não permitirá a atribuição de descontos à conta e ao NIF, em simultâneo.
12. Verificar a incompatibilidade entre descontos, isto é, se um cliente tiver descontos no TMN-TMN, o mesmo não poderá ter descontos no TMN – TMN (Off + Virtual On Net), pois se assim não fosse os descontos seriam cumulativos;
13. Será desejável a definição de três opções na Base de Dados de Descontos Selectivos, consoante o *login* da pessoa:
 - ✓ Introdução/Alteração;
 - ✓ Consulta;
 - ✓ Controller de Direcção.
14. Deverá ser possível definir o perfil dos logins de acesso de acordo com a informação acima;
15. A atribuição de descontos selectivos de 100%, bem como do desconto 184 – Desconto Comunicações TMN-TMN (Entre Cartões da mesma Conta), deve ser restrita a determinados *logins* (em acréscimo deverá ter a autorização do director respectivo). Devem ser geradas listagens mensais dos descontos selectivos atribuídos com esta percentagem.
16. Possibilidade de retirar listagens de descontos: por Código Desconto, telefone, conta, NIF, nome Cliente, data inicio, data fim;
17. Possibilidade retirar informação com perspectiva a produção de indicadores. A produção de indicadores terá de ser definida pela entidade cliente (Direcções Cliente). Por exemplo, Quantidade de cartões/contas/NIF com Descontos Selectivos (DS) atribuído; Quantidade de item's com DS atribuídos; Quantidade de item's com DS atribuídos por cartão/conta/NIF, Quantidade de DS por intervalo percentual, por cartão/conta/NIF; TOP X de descontos máximos por cartão/conta/NIF. Estes Relatórios deverão ser definidos: em termos globais; por direcção gestora; por tarifário; por tipo de conta; etc...

4. Sistemas de Informação

Foi solicitado à DDSI, pelas Direcções Cliente, que a nova Aplicação de Descontos Selectivos ficasse disponível na intranet da Empresa, com um ambiente gráfico, semelhante às aplicações existentes, uma vez que a TMN é uma Empresa certificada, e por tal os layouts encontram-se uniformizados.

Como tal, a DDSI, entendeu desenvolver a aplicação com base em standards J2EE, que possuirá uma interface WEB, consistente com as normas e padrões usuais nas Aplicações da Intranet da Empresa.

De seguida, são apresentados os Requisitos relativos aos sistemas de informação.

4.1 Requisitos dos Sistemas de Informação

A solução proposta, para esta nova aplicação, foi desenvolvida utilizando a linguagem de programação JAVA, de acordo com a especificação J2EE e fará uso das seguintes tecnologias:

- *JavaServer Pages*, para a construção da interface da aplicação;
- *Enterprise JavaBeans*, para implementação das regras de negócio;
- *JDBC*, para acesso às bases de dados.

A aplicação desenvolvida está suportada pelo servidor aplicacional *IBM WebSphere*.

A Base de Dados para o acesso à informação é o *SQL* da *Informix*.

Para além da Base de Dados Descontos Selectivos, é também necessário aceder à Base de Dados *Mobilix* (Base de Dados responsável por toda a informação de um Cliente, desde Nome, N° de Conta, N° Telemóvel, Morada, NIF, etc.), de forma a consultar os dados do cliente.

Esta Base Dados será só de consulta, a nova aplicação não irá guardar nomes de Clientes, isto é, sempre que um NIF/Conta/Telemóvel é inserido, a BD DS, consulta através de uma *Stored Procedure* a BD *Mobilix*, e esta retorna a informação do respectivo Cliente em questão.

4.2 Interface *Mobilix* vs. Descontos Selectivos

Neste sub capítulo, são descritas as interfaces necessárias entre as aplicações *Mobilix* e Descontos Selectivos.

Sempre que for mencionado o campo *Msisdn*, este refere-se sempre ao cartão de voz.

Sempre que for mencionado o campo *Nr_Conta*, este refere-se sempre à conta principal.

Interfaces	Dados de Input	Dados de Output
Interface 1	Msisdn	Nr_Conta Nome do Cliente (par Cliente, Msisdn mais recente).
Interface 2	Nr_conta	Nif Nome do Cliente Código Tipo de Conta (Agrupamento)
Interface 3	Nif	Nome do Cliente (Max(Nr_Conta))
Interface 4	Cod_Campanha	Descrição da Campanha
Interface 5	Cod_Campanha	Lista de Msisdn
Interface 6	Nif	Lista de Contas
Interface 7	Nr_Conta	Lista de Msisdn

Figura 1 – Tabela com o tipo de dados de entrada e saída para a *Stored Procedure*

5. Arquitectura da Aplicação

Neste ponto é descrita a arquitectura da nova aplicação de Descontos Selectivos.

5.1 Organização dos Packages

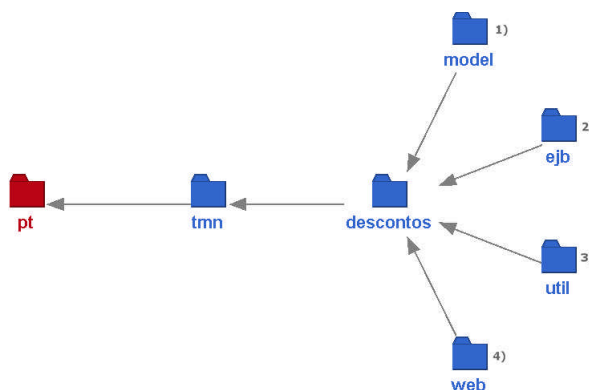


Figura 2 – Organização dos Packages

- 1) Package que contém os objectos trocados entre a camada *WEB* e os *EJBs*;
- 2) Package que contém os *EJBs*;
- 3) Package que contém as classes utilitárias utilizadas em toda a aplicação;
- 4) Package que contém as classes usadas na camada de apresentação (interface *WEB*).

5.2 Estrutura do Ficheiro EAR

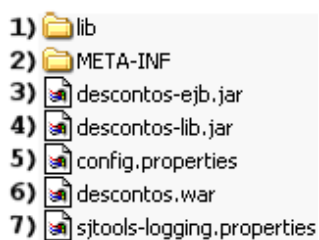


Figura 3 – Estrutura dos Ear`s

- 1) Directório que contém os *JARs* dos quais a aplicação depende;
- 2) Directório que contém os ficheiros específicos para *deploy* de uma aplicação *J2EE*;
- 3) Ficheiro *JAR* que contém os *EJBs*;
- 4) Ficheiro *JAR* que contém as classes usadas pelos *EJBs* e pelas aplicações *WEB*.
- 5) Ficheiro de configuração da aplicação;
- 6) Ficheiro *WAR* que contém a aplicação destinada à Internet;
- 7) Ficheiro de configuração do *log*.

5.3 Arquitectura

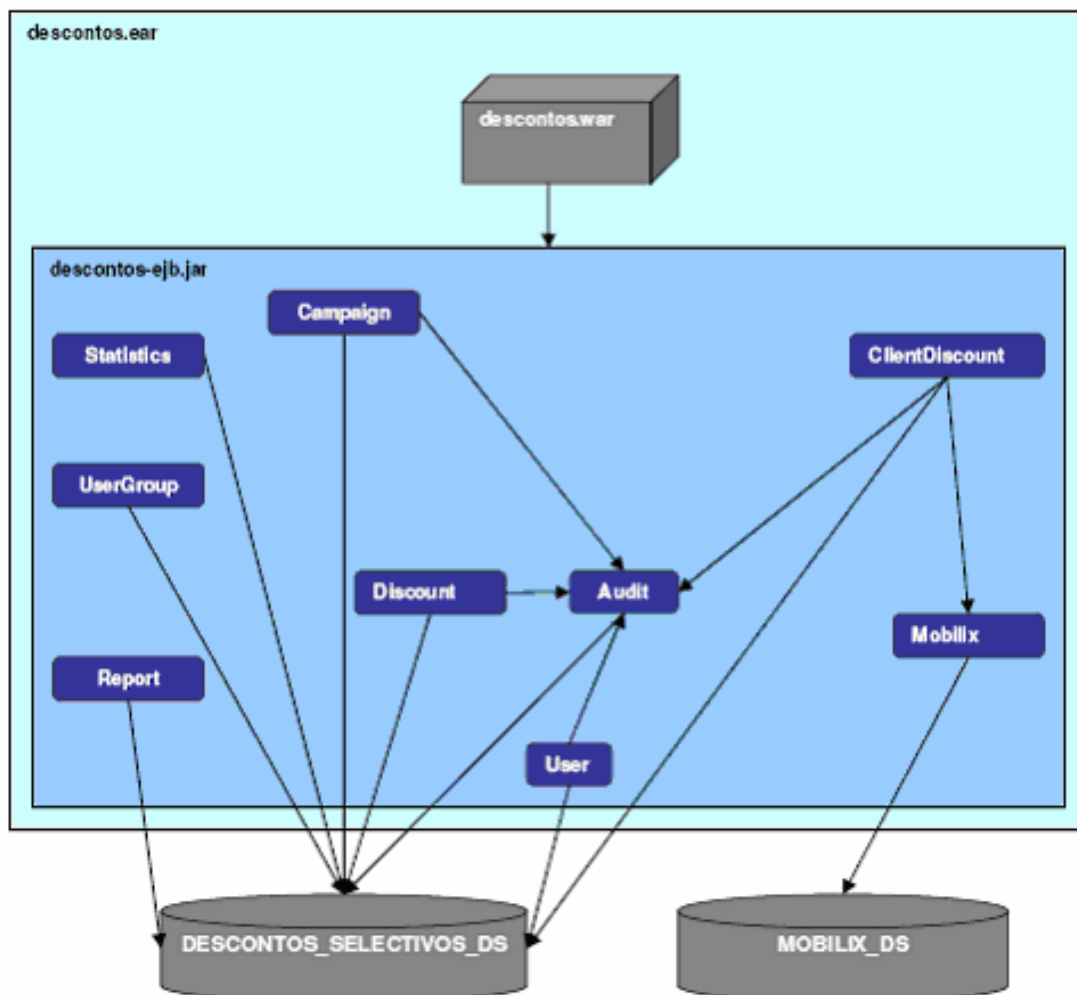


Figura 3 – Arquitectura da Nova Aplicação

6. Migração dos Dados

Como foi referido anteriormente, todos os processos que envolvam a consulta /alteração de dados é feita através da Linguagem *SQL (Strutured Query Language)*.

O Objectivo deste ponto do documento é fazer a descrição das tabelas de suporte à nova aplicação de DS, que se encontram actualmente em vigor.

Esta descrição foi elaborada, segundo a análise às tabelas que se encontram na máquina Melga, motor gateway, Base Dados descontoselect, pois é onde a aplicação está residente ao nível aplicacional.

A criação das tabelas base da nova aplicação foi feita mediante um *script* de configuração (Ver Anexo 1), bem como os dados existentes na antiga aplicação para a actual Base Dados. Essa transição teve de ocorrer num curto espaço de tempo, pois as aplicações (tanto a antiga, como a nova aplicação) não podiam estar paradas por mais de 24 horas, para isso foram seguidos vários passos para que se procedesse à migração sem que ocorressem percas ou falhas de informação, tendo o processo sido testado exaustivamnete.

O processo consistiu em 3 passos:

- 1) Foram feitos unload das tabelas existentes da aplicação antiga;
- 2) Os ficheiros de unload foram tratados de forma a terem a mesma configuração que as tabelas actuais, atrvés de *scripts sql*;
- 3) Foram feitos *loads* para as novas tabelas, com os dados que migraram da antiga aplicação, depois de devidamente tratados.

7. Descrição do Modelo Físico de Dados

O objectivo deste ponto é fazer a Descrição das Tabelas da Base de Dados DescontosSelect e que suportam a aplicação de gestão e configuração de Descontos Selectivos.

Este documento, foi elaborado segundo a análise às tabelas que se encontram na máquina Melga, em descontoselect@gateway, que é onde está residente a Base de Dados.

7.1 Descrição das Tabelas

Tabela Cliente_Desconto

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Cliente_Desconto	cliente_id tipo_cliente_id desconto_id data_inicio	cliente_id tipo_cliente_id desconto_id data_inicio data_fim tipo_valor valor	integer char(1) integer date date varchar(20,0) float

Tabela que armazena os descontos existentes (em vigor) de determinado Cliente. O campo cliente_id aceita apenas inteiros, a tipificação dos descontos é feita através do campo tipo_cliente_id, que é um campo do tipo char(1), e apenas aceita as letras N para Nif, C para Conta e M para Msisdn e V para VPN_ID.

Tabela Cliente_Desconto_1

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Cliente_Desconto_1	id	id cliente_id tipo_cliente_id desconto_id data_inicio data_fim tipo_valor valor processo_id operacao data utilizador	serial integer char(1) integer date date varchar(20,0) float integer char(1) datetime year to fraction(3) varchar(20,0)

Tabela que armazena os logs referentes a quem alterou, inseriu ou apagou os descontos de determinado cliente. Esta tabela funciona como o histórico de toda a aplicação.

Tabela Campanha_Desconto

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Campanha_Desconto	campanha_id desconto_id	campanha_id desconto_id tipo_valor valor	varchar(4,0) integer varchar(20,0) float

Tabela onde estão armazenadas as condições de uma campanha em vigor, isto é, sempre que surgirem novas Campanhas, para um universo de clientes, é nesta tabela que as condições ficam registadas e associadas a determinado código campanha.

Tabela Campanha_Descont_I

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Campanha_Descont_I	id	id campanha_id desconto_id tipo_valor valor operacao data utilizador	serial varchar(4,0) integer varchar(20,0) float char(1) datetime year to fraction(3) varchar(20,0)

Tabela que armazena os logs referentes a quem alterou, inseriu ou apagou as campanhas que se encontram em vigor. Esta tabela funciona como o histórico de todas as campanhas que já não estão activas.

Tabela Desconto

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Desconto	desconto_id	desconto_id descricao data_inicio data_fim tipo_desconto_id	integer varchar(50,0) datetime year to fraction(3) datetime year to fraction(3) integer

Tabela responsável pelo armazenamento dos códigos desconto e respectiva descrição dos mesmos. É nesta tabela que são parametrizados os novos códigos de desconto. O campo desconto_id é atribuído automaticamente e é sequencial.

Tabela Desconto_l

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Desconto_l	id	id desconto_id descricao data_inicio data_fim tipo_desconto_id operacao data utilizador	serial integer varchar(50,0) datetime year to fraction(3) datetime year to fraction(3) fraction(3) integer char(1) datetime year to fraction(3) varchar(20,0)

Tabela que armazena os logs referentes a quem alterou, inseriu ou apagou os códigos de desconto que se encontram em vigor. Esta tabela funciona como o histórico de os códigos que existem e que já existiram.

Tabela Tipo_Desconto

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Tipo_Desconto	tipo_desconto_id	tipo_desconto_id descricao	integer varchar(50,0)

Tabela responsável pela descodificação dos Tipo de Desconto. Existe, apenas para que o Facturix possa interpretar cada tipo de desconto, para que a atribuição dos mesmos seja feita segundo estes parâmetros.

Ex:

1 DESCONTO SELECTIVO EM COMUNICACOES
 3 DESCONTO SELECTIVO EM ASSINATURAS
 6 DESCONTO SELECTIVO NO TOTAL DA FACTURA
 7 DESCONTO SELECTIVO ON-NET (Fact.Repartida)
 8 DESCONTO CLIENTE ESPECIAL (CPPLAN)

Tabela Desconto_Linha

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Desconto_Linha	desconto_id bil_sel_input	desconto_id bil_sel_input bil_sel_output	integer integer integer

Tabela responsável pela associação dos billings de factura a cada código de desconto.

Tabela Desconto_Item

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Desconto_Item	item_id	item_id descricao tipo_item	integer varchar(50,0) char(1)

Tabela que contém a descrição dos itens de factura (item_id), necessários à tabela Desconto_Linha.

Ex.:

item_id	descricao	tipo_item
11	PLANO TMN 15X4	I
12	INDEMNIZACAO POR INCUMPRIMENTO CONTRATUAL	I
13	PLANO TMN 30X4	I
14	TAXA FACTURACAO DETALHADA	I
15	PLANO TMN 30	I
16	PLANO TMN 15	I

NOTA: A coluna tipo_item permite classificar cada item como sendo um item de Input ou de Output

Input: Itens que são considerados para a determinação do valor/base de incidência sobre o qual são aplicadas as condições do desconto.

Output: Itens com que são “facturados” os descontos atribuídos, isto é, item com o qual é apresentado na factura/recibo o desconto selectivo atribuído.

Para uma melhor percepção das tabelas e suas ligações, segue como Anexo 2 o Modelo Físico de Dados.

Tabela Desconto_Intervalo

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Desconto_Intervalo	cliente_id tipo_cliente_id desconto_id data_inicio limite_inferior	cliente_id tipo_cliente_id desconto_id data_inicio limite_inferior limite_superior valor	integer char(1) integer date float float float

Tabela que possibilita a configuração de intervalos/patamares de condições (percentagem/valor) dos descontos a atribuir.

As condições do desconto a atribuir podem variar para os vários intervalos/patamares. São obtidas por comparação dos campos de Limite com o valor da base de incidência do desconto, este ultimo obtido pela soma do valor facturado nos itens que compões/são válidos para o desconto que se está a atribuir.

NOTA: Quando o intervalo/patamar de atribuição de desconto é único, isto é, não existe alteração das condições de desconto em função do valor atingido, esta tabela não possui qualquer ocorrência para o Cliente/Desconto a atribuir.

Tabela Desconto_Conta

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Desconto_Conta	desconto_id tipo_conta_id	desconto_id tipo_conta_id	integer integer

Tabela que possibilita a limitação de códigos de descontos a tipos de conta específicos. Por exemplo, contas VPN apenas podem receber descontos do tipo VPN.

Tabela Desconto_Grupo

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Desconto_Grupo	desconto_id grupo_id	desconto_id grupo_id	integer integer

Tabela que permite associar grupos de utilizadores a determinados tipos de desconto.

Por exemplo, para o desconto 246 – Tráfego TMN – TMN fidelizado, apenas utilizadores que se encontram associados ao grupo 8 (que é o Grupo da Intraconta Fidelizado) tem permissões para inserir descontos neste código.

Tabela Grupo

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Grupo	grupo_id	grupo_id descricao tipo	serial varchar(50,0) char(1)

Tabela onde são criados os grupos de utilizadores e sua descrição.

Ex:

- 1 Administrador
- 2 Indicadores
- 3 Intro/Alteração
- 4 Campanhas
- 5 Consulta
- 6 Descontos 100%
- 8 Desconto IntraConta Fidelizado

Tabela Grupo_Login

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Grupo_Login	grupo_id username	grupo_id username	integer varchar(20,0)

Tabela que possibilita a associação de logins aos vários grupos de utilizadores, isto é, permite definir quais os logins que podem pertencer a cada um dos grupos, como por exemplo Administrador, Consulta, Consulta e Alteração, Campanhas, etc.

Tabela Login

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Login	id	id username nome email perfil_id operacao data utilizador	serial varchar(20,0) varchar(50,0) varchar(50,0) integer char(1) datetime year to fraction(3) varchar(20,0)

Tabela onde são inseridos todos os utilizadores da aplicação, e suas permissões de acesso à mesma.

Tabela Login_I

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Login_I	id	id username nome email perfil_id operacao data utilizador	serial varchar(20,0) varchar(50,0) varchar(50,0) integer char(1) datetime year to fraction(3) varchar(20,0)

Tabela que armazena os logs referentes a quem alterou, inseriu ou apagou os utilizadores da aplicação. Esta tabela funciona como o histórico de todas as operações sobre utilizadores que já não estão activos.

Tabela Relatorio_Grupo

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Relatorio_Grupo	relatorio_id grupo_id	relatorio_id grupo_id	integer integer

Tabela que relaciona determinados relatórios (feitos à medida do utilizador) com os grupos de utilizadores que devem ter acesso a esse relatório.

Tabela Relatorio

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Relatorio	relatorio_id	relatorio_id descricao query	serial varchar(50,0) char(1000)

Tabela onde é possível parametrizar relatórios (sob a forma de query) para a realização de indicadores.

NOTA: Apenas os logins do grupo Administração têm privilégios para configurar relatórios.

Tabela Relatorio_P_IN

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Relatorio_P_IN	relatorio_id ordem	relatorio_id ordem parametro tipo_dados	integer integer varchar(50,0) varchar(20,0)

Tabela onde ficam residentes os inputs que a aplicação recebe para a realização de determinado query, com fim à realização de um relatório.

Tabela Relatorio_P_OUT

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Relatorio_P_OUT	relatorio_id ordem	relatorio_id ordem parametro label tipo_dados	integer integer varchar(50,0) varchar(20,0) varchar(20,0)

Tabela onde ficam residentes os outputs que a aplicação devolve com a execução de determinado query, com fim à realização de um relatório.

Tabela Processo_I

Nome da Tabela	Chaves Primárias	Campos da Tabela	Tipo Dados
Processo_I	id	id descricao data utilizador	serial varchar(50,0) datetime year to fraction(3) varchar(20,0)

Tabela que regista o nº sequencial dos vários processos de carregamento/alteração/remoção de descontos selectivos em massa.

Acessível através de: Menu Principal → Processo em Massa

8. Testes

Antes da Entrada em Produção, qualquer aplicação que seja, carece de uma exaustiva e extensa fase de testes, pois como já foi mencionado anteriormente, a TMN é uma Empresa certificada, como tal, cada processo é feito de acordo com o Manual de Qualidade instituído na Empresa.

Assim sendo, e para esta aplicação em específico, foram efectuados 142 testes, todos eles com sucesso. Segue como Anexo 3, apenas 18 exemplos desses testes, dos 142 efectuados, (se existir interesse poderei facultar a totalidade dos mesmos) para que possam visualizar a metodologia utilizada pela TMN, sempre que existe uma nova aplicação, ou uma alteração que envolva novas funcionalidades numa aplicação já existente e em funcionamento.

Nenhum código, por mais pequena que seja a alteração, entra em Produção sem que se realizem os designados Testes de Aceitação, que são realizados por uma equipa específica (equipa de Testes) responsável pelos mesmos.

9. Formação aos Gestores e Manutenção da Aplicação

De seguida é apresentada a forma como foi dada a formação e como vai ser feita a manutenção da aplicação no futuro.

9.1 Formação

A formação aos utilizadores foi dada nos dias 2 e 3 de Julho em Lisboa, no Porto no dia 4 de Julho, na Madeira no dia 5 de Julho e por fim nos Açores no dia 6 de Julho, todas estas formações foram dadas nas instalações da TMN e ministradas por mim, através de um documento Power Point e através da própria aplicação on-line em rede. Esta formação teve como principal objectivo, colocar as cerca de 200 pessoas, em contacto directo com a aplicação e tentar transmitir as potencialidades da mesma.

Foi-lhes fornecido, via e-mail, um Manual da Aplicação, onde é explicado cada menú exhaustivamente, esse Manual do Utilizador segue como documento Anexo 4.

9.2 Manutenção da Aplicação

A aplicação será mensalmente assistida pela Direcção de Exploração de Sistema de Informação, que é a Direcção que coloca em Produção as novas alterações às aplicações existentes na TMN. Esta aplicação terá uma manutenção baixa, isto é, uma vez por mês é corrido um script (fornecido pela DDSI) com os novos tarifários criados ao longo do mês, para que os códigos de desconto relativos às mensalidades possam englobar estes novos tarifários, bem como novos tipos de chamadas, tarifas, etc.).

Todo este processo é como disse anteriormente exhaustivamente testado.

10. Conclusão

Este Projecto, em conjunto com as Direcções Clientes, teve como propósito a passagem da parametrização dos descontos de clientes, feita pela DDSI para os próprios.

Com esta passagem, os Gestores tornam-se independentes, podendo gerir o seu tempo como lhes convém, sem ter de estar dependentes de terceiros para que os seus clientes possam usufruir de melhores condições em termos de descontos.

O objectivo foi cumprido, e a Nova Aplicação é já um sucesso, uma vez que é a primeira aplicação desenvolvida em WebSphere de raiz.

Existem pois, já algumas aplicações a correr nesse ambiente, contudo são aplicações que já existiam e que migraram para este ambiente.

O feed-back das direcções cliente, têm sido excelentes, tendo sido este projecto considerado como exemplo futuro a seguir.

A aplicação não tem qualquer limitação, pelo que quer a nível de requisitos quer a nível de prazos os mesmos foram cumpridos com *SUCESSO*.

Como Gestora de Projecto, na Direcção de Desenvolvidos de Sistemas de Informação (DDSI) da TMN, este Projecto em que fui responsável, foi uma aposta muito forte e que me fortaleceu ainda mais em termos técnicos e pessoais.

A nível técnico, o conhecimento com uma ferramenta que está a ser fortemente implementada na TMN, que é o WebSphere (RAD) e que está numa fase muito inicial ainda, mas que prevê-se que a curto prazo todas as aplicações migrem para esta tecnologia.

A nível pessoal, muito gratificante, pois a minha Direcção, e em específico o meu Departamento levou os Parabéns, quer dos Utilizadores para quem a mesma foi desenhada e concebida, quer das outras Direcções, uma vez que este projecto foi pioneiro em termos da tecnologia atrás mencionada.

ANEXOS

Anexo 1 – Schema de Criação da Nova BD – Descontos Selectivos

DATABASE descontoselect@gateway;

BEGIN work;

```
CREATE TABLE GRUPO (  
    GRUPO_ID SERIAL NOT NULL,  
    DESCRICAO VARCHAR(50) NOT NULL,  
    TIPO CHAR(1) NOT NULL,  
    PRIMARY KEY (GRUPO_ID)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON GRUPO FROM PUBLIC;  
GRANT SELECT ON GRUPO TO PUBLIC;  
GRANT ALL ON GRUPO TO tm004651;  
GRANT ALL ON GRUPO TO tm008181;  
GRANT ALL ON GRUPO TO ds_dsv;
```

```
CREATE TABLE LOGIN (  
    USERNAME VARCHAR(20) NOT NULL,  
    NOME VARCHAR(50) NOT NULL,  
    EMAIL VARCHAR(50) NOT NULL,  
    PERFIL_ID INTEGER NOT NULL,  
    PRIMARY KEY (USERNAME)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON LOGIN FROM PUBLIC;  
GRANT SELECT ON LOGIN TO PUBLIC;  
GRANT ALL ON LOGIN TO tm004651;  
GRANT ALL ON LOGIN TO tm008181;  
GRANT ALL ON LOGIN TO ds_dsv;
```

```
ALTER TABLE LOGIN ADD CONSTRAINT FOREIGN KEY (PERFIL_ID) REFERENCES  
GRUPO (GRUPO_ID);
```

```
CREATE TABLE LOGIN_L (  
    ID SERIAL NOT NULL,  
    USERNAME VARCHAR(20),  
    NOME VARCHAR(50),  
    EMAIL VARCHAR(50),  
    PERFIL_ID INTEGER,  
    OPERACAO CHAR(1) NOT NULL,
```

```
DATA DATETIME year TO fraction(3) NOT NULL,
UTILIZADOR VARCHAR(20) NOT NULL,
PRIMARY KEY (ID)
) LOCK MODE ROW;

REVOKE ALL ON LOGIN_L FROM PUBLIC;
GRANT SELECT ON LOGIN_L TO PUBLIC;
GRANT ALL ON LOGIN_L TO tm004651;
GRANT ALL ON LOGIN_L TO tm008181;
GRANT ALL ON LOGIN_L TO ds_dsv;

CREATE TABLE GRUPO_LOGIN (
    GRUPO_ID INTEGER NOT NULL,
    USERNAME VARCHAR(20) NOT NULL,
    PRIMARY KEY (GRUPO_ID, USERNAME)
) LOCK MODE ROW;

REVOKE ALL ON GRUPO_LOGIN FROM PUBLIC;
GRANT SELECT ON GRUPO_LOGIN TO PUBLIC;
GRANT ALL ON GRUPO_LOGIN TO tm004651;
GRANT ALL ON GRUPO_LOGIN TO tm008181;
GRANT ALL ON GRUPO_LOGIN TO ds_dsv;

ALTER TABLE GRUPO_LOGIN ADD CONSTRAINT FOREIGN KEY (GRUPO_ID)
REFERENCES GRUPO (GRUPO_ID);
ALTER TABLE GRUPO_LOGIN ADD CONSTRAINT FOREIGN KEY (USERNAME)
REFERENCES LOGIN (USERNAME);

CREATE TABLE DESCONTO_L (
    ID SERIAL NOT NULL,
    DESCONTO_ID INTEGER,
    DESCRICAO VARCHAR(50),
    DATA_INICIO DATETIME year TO fraction(3),
    DATA_FIM DATETIME year TO fraction(3),
    TIPO_DESCONTO_ID INTEGER,
    OPERACAO CHAR(1) NOT NULL,
    DATA DATETIME year TO fraction(3) NOT NULL,
    UTILIZADOR VARCHAR(20) NOT NULL,
    PRIMARY KEY (ID)
) LOCK MODE ROW;

REVOKE ALL ON DESCONTO_L FROM PUBLIC;
GRANT SELECT ON DESCONTO_L TO PUBLIC;
GRANT ALL ON DESCONTO_L TO tm004651;
GRANT ALL ON DESCONTO_L TO tm008181;
```

GRANT ALL ON DESCONTO_L TO ds_dsv;

```
CREATE TABLE TIPO_DESCONTO (  
    TIPO_DESCONTO_ID INTEGER NOT NULL,  
    DESCRICAO VARCHAR(50) NOT NULL,  
    PRIMARY KEY (TIPO_DESCONTO_ID)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON TIPO_DESCONTO FROM PUBLIC;  
GRANT SELECT ON TIPO_DESCONTO TO PUBLIC;  
GRANT ALL ON TIPO_DESCONTO TO tm004651;  
GRANT ALL ON TIPO_DESCONTO TO tm008181;  
GRANT ALL ON TIPO_DESCONTO TO ds_dsv;
```

```
CREATE TABLE DESCONTO_ITEM (  
    ITEM_ID INTEGER NOT NULL,  
    DESCRICAO VARCHAR(50) NOT NULL,  
    TIPO_ITEM CHAR(1) NOT NULL,  
    PRIMARY KEY (ITEM_ID)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON DESCONTO_ITEM FROM PUBLIC;  
GRANT SELECT ON DESCONTO_ITEM TO PUBLIC;
```

```
GRANT ALL ON DESCONTO_ITEM TO tm004651;  
GRANT ALL ON DESCONTO_ITEM TO tm008181;  
GRANT ALL ON DESCONTO_ITEM TO ds_dsv;
```

```
CREATE TABLE CLIENTE_DESCONTO_L (  
    ID SERIAL NOT NULL,  
    CLIENTE_ID INTEGER,  
    TIPO_CLIENTE_ID CHAR(1),  
    DESCONTO_ID INTEGER,  
    DATA_INICIO DATE,  
    DATA_FIM DATE,  
    TIPO_VALOR VARCHAR(20),  
    VALOR FLOAT,  
    PROCESSO_ID INTEGER,  
    OPERACAO CHAR(1) NOT NULL,  
    DATA DATETIME year TO fraction(3) NOT NULL,  
    UTILIZADOR VARCHAR(20) NOT NULL,  
    PRIMARY KEY (ID)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON CLIENTE_DESCONTO_L FROM PUBLIC;
```

```
GRANT SELECT ON CLIENTE_DESCONTO_L TO PUBLIC;  
GRANT ALL ON CLIENTE_DESCONTO_L TO tm004651;  
GRANT ALL ON CLIENTE_DESCONTO_L TO tm008181;  
GRANT ALL ON CLIENTE_DESCONTO_L TO ds_dsv;
```

```
CREATE TABLE RELATORIO (  
    RELATORIO_ID SERIAL NOT NULL,  
    DESCRICAO VARCHAR(50) NOT NULL,  
    QUERY CHAR(1000) NOT NULL,  
    PRIMARY KEY (RELATORIO_ID)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON RELATORIO FROM PUBLIC;  
GRANT SELECT ON RELATORIO TO PUBLIC;  
GRANT ALL ON RELATORIO TO tm004651;  
GRANT ALL ON RELATORIO TO tm008181;  
GRANT ALL ON RELATORIO TO ds_dsv;
```

```
CREATE TABLE RELATORIO_P_IN (  
    RELATORIO_ID INTEGER NOT NULL,  
    ORDEM INTEGER NOT NULL,  
    PARAMETRO VARCHAR(50) NOT NULL,  
    TIPO_DADOS VARCHAR(20) NOT NULL,  
    PRIMARY KEY (RELATORIO_ID, ORDEM)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON RELATORIO_P_IN FROM PUBLIC;  
GRANT SELECT ON RELATORIO_P_IN TO PUBLIC;  
GRANT ALL ON RELATORIO_P_IN TO tm004651;  
GRANT ALL ON RELATORIO_P_IN TO tm008181;  
GRANT ALL ON RELATORIO_P_IN TO ds_dsv;
```

```
ALTER TABLE RELATORIO_P_IN ADD CONSTRAINT FOREIGN KEY  
(RELATORIO_ID) REFERENCES RELATORIO (RELATORIO_ID);
```

```
CREATE TABLE RELATORIO_P_OUT (  
    RELATORIO_ID INTEGER NOT NULL,  
    ORDEM INTEGER NOT NULL,  
    PARAMETRO VARCHAR(50) NOT NULL,  
    LABEL VARCHAR(20) NOT NULL,  
    TIPO_DADOS VARCHAR(20) NOT NULL,  
    PRIMARY KEY (RELATORIO_ID, ORDEM)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON RELATORIO_P_OUT FROM PUBLIC;
```

```
GRANT SELECT ON RELATORIO_P_OUT TO PUBLIC;
GRANT ALL ON RELATORIO_P_OUT TO tm004651;
GRANT ALL ON RELATORIO_P_OUT TO tm008181;
GRANT ALL ON RELATORIO_P_OUT TO ds_dsv;
```

```
ALTER TABLE RELATORIO_P_OUT ADD CONSTRAINT FOREIGN KEY
(RELATORIO_ID) REFERENCES RELATORIO (RELATORIO_ID);
```

```
CREATE TABLE RELATORIO_GRUPO (
    RELATORIO_ID INTEGER NOT NULL,
    GRUPO_ID INTEGER NOT NULL,
    PRIMARY KEY (RELATORIO_ID, GRUPO_ID)
) LOCK MODE ROW;
```

```
REVOKE ALL ON RELATORIO_GRUPO FROM PUBLIC;
GRANT SELECT ON RELATORIO_GRUPO TO PUBLIC;
GRANT ALL ON RELATORIO_GRUPO TO tm004651;
GRANT ALL ON RELATORIO_GRUPO TO tm008181;
GRANT ALL ON RELATORIO_GRUPO TO ds_dsv;
ALTER TABLE RELATORIO_GRUPO ADD CONSTRAINT FOREIGN KEY
(RELATORIO_ID) REFERENCES RELATORIO (RELATORIO_ID);
ALTER TABLE RELATORIO_GRUPO ADD CONSTRAINT FOREIGN KEY (GRUPO_ID)
REFERENCES GRUPO (GRUPO_ID);
```

```
CREATE TABLE CAMPANHA_DESCONT_L (
    ID SERIAL NOT NULL,
    CAMPANHA_ID VARCHAR(4),
    DESCONTO_ID INTEGER,
    TIPO_VALOR VARCHAR(20),
    VALOR FLOAT,
    OPERACAO CHAR(1) NOT NULL,
    DATA DATETIME year TO fraction(3) NOT NULL,
    UTILIZADOR VARCHAR(20) NOT NULL,
    PRIMARY KEY (ID)
) LOCK MODE ROW;
```

```
REVOKE ALL ON CAMPANHA_DESCONT_L FROM PUBLIC;
GRANT SELECT ON CAMPANHA_DESCONT_L TO PUBLIC;
GRANT ALL ON CAMPANHA_DESCONT_L TO tm004651;
GRANT ALL ON CAMPANHA_DESCONT_L TO tm008181;
GRANT ALL ON CAMPANHA_DESCONT_L TO ds_dsv;
```

```
CREATE TABLE PROCESSO_L (
    ID SERIAL NOT NULL,
    DESCRICAO VARCHAR(50),
```

```
DATA DATETIME year TO fraction(3) NOT NULL,
UTILIZADOR VARCHAR(20) NOT NULL,
PRIMARY KEY (ID)
) LOCK MODE ROW;

REVOKE ALL ON PROCESSO_L FROM PUBLIC;
GRANT SELECT ON PROCESSO_L TO PUBLIC;
GRANT ALL ON PROCESSO_L TO tm004651;
GRANT ALL ON PROCESSO_L TO tm008181;
GRANT ALL ON PROCESSO_L TO ds_dsv;

CREATE TABLE DESCONTO (
    DESCONTO_ID INTEGER NOT NULL,
    DESCRICAO VARCHAR(50) NOT NULL,
    DATA_INICIO DATETIME year TO fraction(3) NOT NULL,
    DATA_FIM DATETIME year TO fraction(3),
    TIPO_DESCONTO_ID INTEGER NOT NULL,
    PRIMARY KEY (DESCONTO_ID)
) LOCK MODE ROW;

REVOKE ALL ON DESCONTO FROM PUBLIC;
GRANT SELECT ON DESCONTO TO PUBLIC;
GRANT ALL ON DESCONTO TO tm004651;
GRANT ALL ON DESCONTO TO tm008181;
GRANT ALL ON DESCONTO TO ds_dsv;

ALTER TABLE DESCONTO ADD CONSTRAINT FOREIGN KEY (TIPO_DESCONTO_ID)
REFERENCES TIPO_DESCONTO (TIPO_DESCONTO_ID);

CREATE TABLE DESCONTO_GRUPO (
    DESCONTO_ID INTEGER NOT NULL,
    GRUPO_ID INTEGER NOT NULL,
    PRIMARY KEY (DESCONTO_ID, GRUPO_ID)
) LOCK MODE ROW;

REVOKE ALL ON DESCONTO_GRUPO FROM PUBLIC;
GRANT SELECT ON DESCONTO_GRUPO TO PUBLIC;
GRANT ALL ON DESCONTO_GRUPO TO tm004651;
GRANT ALL ON DESCONTO_GRUPO TO tm008181;
GRANT ALL ON DESCONTO_GRUPO TO ds_dsv;

ALTER TABLE DESCONTO_GRUPO ADD CONSTRAINT FOREIGN KEY
(DESCONTO_ID) REFERENCES DESCONTO (DESCONTO_ID);
ALTER TABLE DESCONTO_GRUPO ADD CONSTRAINT FOREIGN KEY (GRUPO_ID)
REFERENCES GRUPO (GRUPO_ID);
```



```
CREATE TABLE DESCONTO_CONTA (  
    DESCONTO_ID INTEGER NOT NULL,  
    TIPO_CONTA_ID INTEGER NOT NULL,  
    PRIMARY KEY (DESCONTO_ID, TIPO_CONTA_ID)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON DESCONTO_CONTA FROM PUBLIC;  
GRANT SELECT ON DESCONTO_CONTA TO PUBLIC;  
GRANT ALL ON DESCONTO_CONTA TO tm004651;  
GRANT ALL ON DESCONTO_CONTA TO tm008181;  
GRANT ALL ON DESCONTO_CONTA TO ds_dsv;
```

```
ALTER TABLE DESCONTO_CONTA ADD CONSTRAINT FOREIGN KEY  
(DESCONTO_ID) REFERENCES DESCONTO (DESCONTO_ID);
```

```
CREATE TABLE DESCONTO_LINHA (  
    DESCONTO_ID INTEGER NOT NULL,  
    BIL_SEL_INPUT INTEGER NOT NULL,  
    BIL_SEL_OUTPUT INTEGER NOT NULL,  
    PRIMARY KEY (DESCONTO_ID, BIL_SEL_INPUT)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON DESCONTO_LINHA FROM PUBLIC;  
GRANT SELECT ON DESCONTO_LINHA TO PUBLIC;  
GRANT ALL ON DESCONTO_LINHA TO tm004651;  
GRANT ALL ON DESCONTO_LINHA TO tm008181;  
GRANT ALL ON DESCONTO_LINHA TO ds_dsv;
```

```
ALTER TABLE DESCONTO_LINHA ADD CONSTRAINT FOREIGN KEY  
(DESCONTO_ID) REFERENCES DESCONTO (DESCONTO_ID);  
ALTER TABLE DESCONTO_LINHA ADD CONSTRAINT FOREIGN KEY  
(BIL_SEL_INPUT) REFERENCES DESCONTO_ITEM (ITEM_ID);  
ALTER TABLE DESCONTO_LINHA ADD CONSTRAINT FOREIGN KEY  
(BIL_SEL_OUTPUT) REFERENCES DESCONTO_ITEM (ITEM_ID);
```

```
CREATE TABLE CLIENTE_DESCONTO (  
    CLIENTE_ID INTEGER NOT NULL,  
    TIPO_CLIENTE_ID CHAR(1) NOT NULL,  
    DESCONTO_ID INTEGER NOT NULL,  
    DATA_INICIO DATE NOT NULL,  
    DATA_FIM DATE,  
    TIPO_VALOR VARCHAR(20) NOT NULL,  
    VALOR FLOAT,
```

```
PRIMARY KEY (CLIENTE_ID, TIPO_CLIENTE_ID, DESCONTO_ID,  
DATA_INICIO)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON CLIENTE_DESCONTO FROM PUBLIC;  
GRANT SELECT ON CLIENTE_DESCONTO TO PUBLIC;
```

```
GRANT ALL ON CLIENTE_DESCONTO TO tm004651;  
GRANT ALL ON CLIENTE_DESCONTO TO tm008181;  
GRANT ALL ON CLIENTE_DESCONTO TO ds_dsv;
```

```
ALTER TABLE CLIENTE_DESCONTO ADD CONSTRAINT FOREIGN KEY  
(DESCONTO_ID) REFERENCES DESCONTO (DESCONTO_ID);
```

```
CREATE TABLE DESCONTO_INTERVALO (  
    CLIENTE_ID INTEGER NOT NULL,  
    TIPO_CLIENTE_ID CHAR(1) NOT NULL,  
    DESCONTO_ID INTEGER NOT NULL,  
    DATA_INICIO DATE NOT NULL,  
    LIMITE_INFERIOR FLOAT NOT NULL,  
    LIMITE_SUPERIOR FLOAT NOT NULL,  
    VALOR FLOAT NOT NULL,  
    PRIMARY KEY (CLIENTE_ID, TIPO_CLIENTE_ID, DESCONTO_ID,  
DATA_INICIO, LIMITE_INFERIOR)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON DESCONTO_INTERVALO FROM PUBLIC;  
GRANT SELECT ON DESCONTO_INTERVALO TO PUBLIC;  
GRANT ALL ON DESCONTO_INTERVALO TO tm004651;  
GRANT ALL ON DESCONTO_INTERVALO TO tm008181;  
GRANT ALL ON DESCONTO_INTERVALO TO ds_dsv;
```

```
ALTER TABLE DESCONTO_INTERVALO ADD CONSTRAINT FOREIGN KEY  
(CLIENTE_ID, TIPO_CLIENTE_ID, DESCONTO_ID, DATA_INICIO)  
REFERENCES CLIENTE_DESCONTO (CLIENTE_ID, TIPO_CLIENTE_ID,  
DESCONTO_ID, DATA_INICIO);
```

```
CREATE TABLE CAMPANHA_DESCONTO (  
    CAMPANHA_ID VARCHAR(4) NOT NULL,  
    DESCONTO_ID INTEGER NOT NULL,  
    TIPO_VALOR VARCHAR(20) NOT NULL,  
    VALOR FLOAT,  
    PRIMARY KEY (CAMPANHA_ID, DESCONTO_ID)  
) LOCK MODE ROW;
```

```
REVOKE ALL ON CAMPANHA_DESCONTO FROM PUBLIC;
```

```
GRANT SELECT ON CAMPANHA_DESCONTO TO PUBLIC;  
GRANT ALL ON CAMPANHA_DESCONTO TO tm004651;  
GRANT ALL ON CAMPANHA_DESCONTO TO tm008181;  
GRANT ALL ON CAMPANHA_DESCONTO TO ds_dsv;
```

```
ALTER TABLE CAMPANHA_DESCONTO ADD CONSTRAINT FOREIGN KEY  
(DESCONTO_ID) REFERENCES DESCONTO (DESCONTO_ID);
```

```
insert into tipo_desconto values (1,"DESCONTO SELECTIVO EM COMUNICACOES");  
insert into tipo_desconto values (3,"DESCONTO SELECTIVO EM ASSINATURAS");  
insert into tipo_desconto values (6,"DESCONTO SELECTIVO NO TOTAL DA FACTURA");  
insert into tipo_desconto values (7,"DESCONTO SELECTIVO ON-NET (Fact.Repartida)");  
insert into tipo_desconto values (8,"DESCONTO CLIENTE ESPECIAL (CPPLAN)");
```

```
insert into grupo values (0,"Administrador","I");  
insert into grupo values (0,"Indicadores ","I");  
insert into grupo values (0,"Intro/Alteração","I");  
insert into grupo values (0,"Campanhas ","I");  
insert into grupo values (0,"Consulta","I");  
insert into grupo values (0,"Descontos 100%","A");
```

```
commit work;
```

```
begin work;  
--Tabela Desconto  
load from desconto_2.unl insert into desconto;  
commit work;
```

```
begin work;  
--Tabela Desconto_Itens  
load from descontoitens_2.unl insert into desconto_item;  
commit work;
```

```
begin work;  
--Tabela Desconto_Linha  
load from descontolinha_2.unl insert into desconto_linha;  
commit work;
```

```
--Tabela Cliente_Desconto por Cliente, Nif e Telefone, por Percentagem e Valor  
begin work;  
lock table cliente_desconto in exclusive mode;  
load from conta_percentagem.unl insert into cliente_desconto;  
commit work;
```

```
begin work;  
lock table cliente_desconto in exclusive mode;  
load from conta_valor.unl insert into cliente_desconto;  
commit work;
```

```
begin work;  
lock table cliente_desconto in exclusive mode;  
load from nif_percentagem.unl insert into cliente_desconto;  
commit work;
```

```
begin work;  
lock table cliente_desconto in exclusive mode;  
load from nif_valor.unl insert into cliente_desconto;  
commit work;
```

```
begin work;  
lock table cliente_desconto in exclusive mode;  
load from telefone_percentagem.unl insert into cliente_desconto;  
commit work;
```

```
begin work;  
lock table cliente_desconto in exclusive mode;  
load from telefone_valor.unl insert into cliente_desconto;  
commit work;
```

```
begin work;  
--Tabela Login  
load from Tipo_Acesso_BD_DS_SR.unl insert into login;  
commit work;
```

```
update statistics for table Desconto;  
update statistics for table Desconto_item;  
update statistics for table Desconto_linha;  
update statistics for table cliente_desconto;  
update statistics for table Login;
```

Anexo 2 – Modelo Físico de Dados

Anexo 3 – Modelo Relatório de Testes TMN – Mod_B0657_Rel_Testes

Anexo 4 – Manual do Utilizador