

Database Marketing

Base de Dados de Clientes

Luís Alberto Ribeiro Carreira

Fevereiro de 2011

Orientador: Professor Luís Cavique

Página em branco

Agradecimentos

Quero deixar aqui uma palavra de apreço ao Professor Luís Cavique. A sua receptividade para acolher a orientação deste trabalho fora do período normal de leccionamento e a sua total disponibilidade tornaram possível a execução deste projecto.

Agradeço igualmente a todos os outros professores que contribuíram com o seu trabalho e dedicação para este meu percurso pela Universidade Aberta.

Aos meus colegas de curso expresso também a minha gratidão. Com o vosso apoio as coisas ficaram sempre mais simples.

Por fim, um obrigado muito especial à minha família. Pelo vosso amparo e enorme paciência.

Página em branco

Índice

1. Introdução	8
2. Objectivos.....	9
2.1. Introdução.....	9
2.2. Objectivos	9
2.3. Estratégia adoptada	9
2.4. Enquadramento	10
2.5. Conclusão	10
3. Requisitos	11
3.1. Introdução.....	11
3.2. Preparação da Base de Dados.....	11
3.3. Requisitos para o funcionamento.....	13
3.4. Conclusão	13
4. RM e as métricas	14
4.1. Introdução.....	14
4.2. O critério de segmentação RM	14
4.3. Churn.....	Erro! Marcador não definido.
4.4. Taxa de retenção.....	15
4.5. LTV.....	15
4.6. LTV individual.....	16
4.7. Vantagens dos indicadores	17
4.8. Conclusão	17
5. Access e SQL.....	18
5.1. Introdução.....	18
5.2. As consultas	18
5.3. As tabelas	20

5.4.	Código SQL	22
5.5.	Conclusão	32
6.	O interface	33
6.1.	Introdução.....	33
6.2.	Os dashboards.....	33
6.3.	Ecrã inicial	35
6.4.	Painel Clientes (BD).....	37
6.5.	Painel RM	39
6.6.	Painel LTV.....	41
6.7.	Conclusão	43
7.	VBA	44
7.1.	Introdução.....	44
7.2.	Código VBA.....	44
7.3.	VBA para utilizador ver	44
7.4.	DAO	47
7.5.	ADO	48
7.6.	Diagrama de sequência.....	49
7.7.	Conclusão	51
8.	Manuais de utilização	52
8.1.	Introdução.....	52
8.2.	Produzir os tutoriais.....	52
8.3.	Tutorial do ecrã inicial.....	54
8.4.	Conclusão	60
9.	Conclusão	61
10.	Bibliografia.....	62

Índice de Ilustrações

<i>FIGURA 3 - 1: UMA BASE DE DADOS RELACIONAL DE CLIENTES.....</i>	<i>11</i>
<i>FIGURA 3 - 2: PROGRAMA FLOWHEATER PARA GERAÇÃO DE DADOS PARA TESTES</i>	<i>12</i>
<i>FIGURA 3 - 3: TABELAS DA BASE DE DADOS NECESSÁRIAS.....</i>	<i>13</i>
 <i>FIGURA 4 - 1: CÁLCULO DO LTV CONFORME HTTP://WWW.DBMARKETING.COM/ARTICLES/ART251A.HTM</i>	 <i>16</i>
 <i>FIGURA 5 - 1: LTV CALCULADO NO MS ACCESS.....</i>	 <i>19</i>
<i>FIGURA 5 - 2: TABELAS NA BASE DE DADOS</i>	<i>22</i>
 <i>FIGURA 6 - 1: PRODUTO PILOT DA SAP.....</i>	 <i>34</i>
<i>FIGURA 6 - 2: PRODUTO PILOT DA SAP (DASHBOARD)</i>	<i>34</i>
<i>FIGURA 6 - 3: DASHBOARD – ECRÃ INICIAL.....</i>	<i>35</i>
<i>FIGURA 6 - 4: DASHBOARD – ECRÃ INICIAL, PRONTO PARA UTILIZAÇÃO</i>	<i>37</i>
<i>FIGURA 6 - 5: DASHBOARD – PAINEL CLIENTES</i>	<i>38</i>
<i>FIGURA 6 - 6: PORMENOR DO PAINEL CLIENTES</i>	<i>38</i>
<i>FIGURA 6 - 7: DASHBOARD RM.....</i>	<i>40</i>
<i>FIGURA 6 - 8: DASHBOARD LTV.....</i>	<i>42</i>
<i>FIGURA 6 - 9: USERFORM COM OS DADOS PARA CÁLCULO DO LTV.....</i>	<i>42</i>
 <i>FIGURA 7 - 1: DIAGRAMA DE SEQUÊNCIA.....</i>	 <i>50</i>
 <i>FIGURA 8 - 1: PROGRAMA WINK</i>	 <i>53</i>
<i>FIGURA 8 - 2 A 12: TUTORIAL DO ECRÃ INICIAL</i>	<i>59</i>

1. Introdução

O tema deste trabalho foi escolhido entre três propostas apresentadas pelo Professor Luís Cavique.

A proposta inicial previa a demonstração do funcionamento de um novo critério de segmentação de clientes, o modelo RM (**R**ecentividade/**valor Monetário**), inspirado na combinação de critérios referidos na literatura, que para além da sua simplicidade, apresenta a vantagem de permitir criar um maior número de estratégias de comunicação. Devia de ser evidenciada a sua utilidade na manipulação dos dados existentes numa base de dados de clientes, de forma a permitir tomar decisões de gestão e prever quais os clientes que vão deixar de comprar.

Posteriormente percebeu-se que seria muito interessante proceder também ao cálculo de outras métricas e fazer uma comparação dos resultados obtidos com duas delas – RM e LTV individual (**L**ife **T**ime **V**alue).

2. Objectivos

2.1. Introdução

Neste capítulo vão ser explicados os objectivos do trabalho e a metodologia adoptada para os concretizar.

2.2. Objectivos

Em marketing relacional pretende-se associar ao cliente certo, o produto certo com a mensagem apropriada.

O objectivo deste trabalho é providenciar uma ferramenta capaz de encontrar os compradores que deixaram de ter interesse económico para a empresa, utilizando um inovador critério de segmentação de clientes, o modelo RM (**R**ecentricidade/**valor Monetário**), aplicado a uma base de dados de clientes existente. Este modelo, de simples aplicação, representa um enorme apoio no planeamento de estratégias de comunicação.

Pretende-se alargar ainda mais a ajuda à tomada de decisões relacionadas com os clientes, pelo que se calculam algumas das métricas mais utilizadas em marketing relacional, tais como o churn, a taxa de retenção, o LTV (**L**ife **T**ime **V**alue) global a três anos, e o LTV individual.

O gestor fica assim habilitado a saber quais os clientes que vão deixar de comprar, qual o lucro que, em média, pode esperar no futuro de cada um dos seus clientes e qual o lucro que cada um dos seus clientes representa individualmente.

Esta ferramenta deve ser de fácil adaptação e utilização, e possibilitar o máximo de interacção com o usuário de forma a ser apelativa e útil ao mesmo tempo.

2.3. Estratégia adoptada

Considerando os objectivos propostos optou-se por fazer a implementação de todas as pesquisas e preparação dos dados em SQL (Microsoft Office Access), sendo as análises, e a exibição dos respectivos resultados, processadas em Microsoft Office Excel, tomando assim partido das melhores capacidades de cada uma destas ferramentas.

A base de dados em uso tem o nome de “bd_teste.mdb”. Mas facilmente pode ser feita a alteração necessária para trabalhar com outro ficheiro, desde que reúna os requisitos indicados no ponto 3.2 da página 11.

O ficheiro de Excel, que é o que o utilizador vai usar efectivamente, chama-se “Projecto.xls”.

Para interligar as duas aplicações de forma dinâmica e em tempo real recorreu-se ao VBA.

2.4. Enquadramento

O presente trabalho enquadra-se nas seguintes Unidades Curriculares da Licenciatura em Informática:

- Fundamentos de Base de Dados
- Sistemas de Gestão de Base de Dados
- Desenvolvimento de Software
- Programação

2.5. Conclusão

Neste capítulo referiram-se quais são os objectivos do trabalho e qual a estratégia que foi seguida para a sua concretização, bem como as Unidades Curriculares em que se enquadra.

O capítulo 3 vai esclarecer o trabalho envolvido na preparação da base de dados de clientes e os requisitos para o funcionamento da aplicação.

3. Requisitos

3.1. Introdução

Neste capítulo clarifica-se como se obteve a base de dados de trabalho e como a mesma teve de ser preparada. Esclarece-se também os requisitos necessários para o funcionamento do aplicativo.

3.2. Preparação da Base de Dados

Toda a empresa que dependa das compras dos seus clientes deve criar e manter actualizada uma base de dados relacional que permita conhecer a sua carteira de clientes – quem são, onde estão, que produtos compraram, quando compraram e quais os contactos que houve com eles.

Na figura 3-1 apresenta-se uma possível estrutura com essa finalidade.

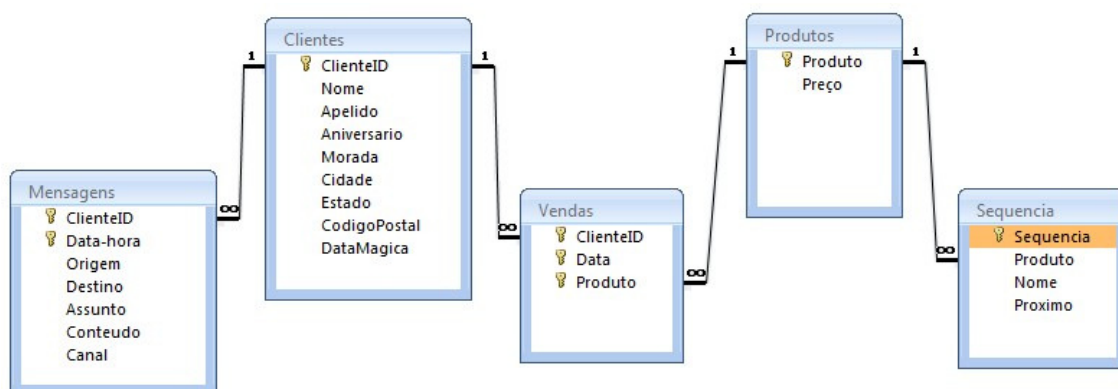


Figura 3 - 1: Uma Base de Dados relacional de clientes

Para a finalidade deste trabalho pretendeu-se obter uma base de dados que espelhasse o mais realisticamente possível a actividade de uma empresa, durante um espaço temporal mínimo de três anos.

A partir do site do Database Marketing Institute (<http://www.dbmarketing.com>), de Arthur Middleton Hughes, um dos maiores especialistas da matéria, fez-se o download das tabelas necessárias. No entanto foi necessário gerar ainda cerca de 12.448 registos com o programa de conversão e geração de dados FlowHeater, disponível em <http://www.flowheater.net>, pois os dados disponíveis não cobriam na totalidade os três anos requeridos. Desta forma os valores deixaram de ter a vertente tão realista que se gostava, mas para a finalidade do trabalho em si este facto não passa de um pequeno pormenor.

Para o funcionamento da aplicação desenvolvida, é necessário ter as tabelas Orders e Clientes, conforme figura 3-3.

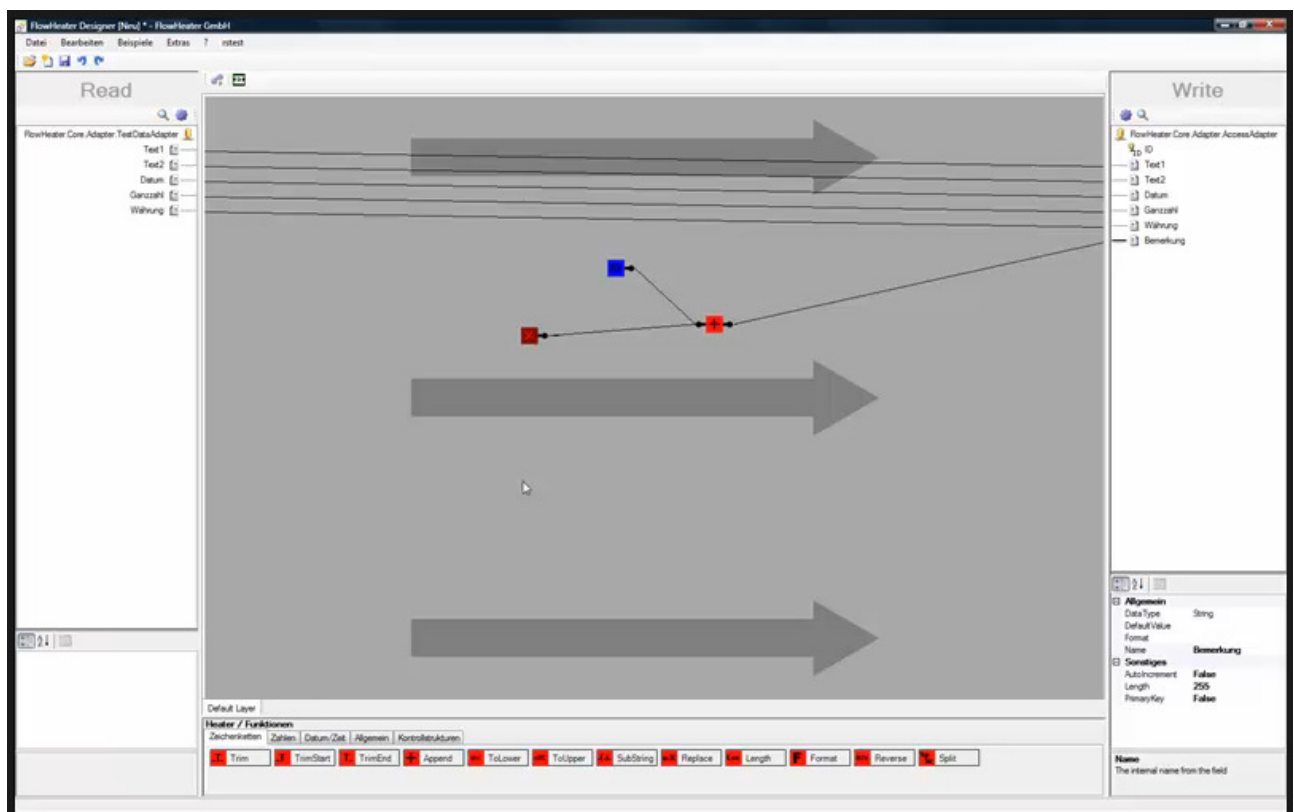


Figura 3 - 2: Programa FlowHeater para geração de dados para testes

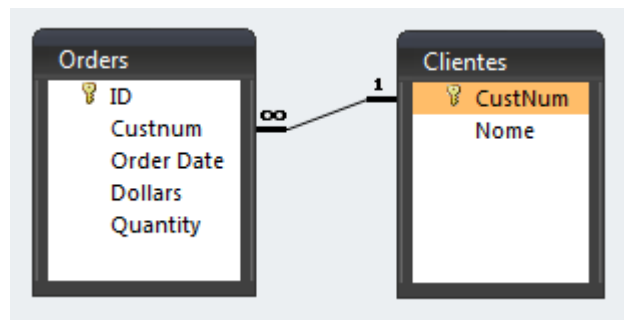


Figura 3 - 3: Tabelas da Base de Dados necessárias

3.3. Requisitos para o funcionamento

Tanto a base de dados do MS Access como o MS Excel possuem código em VBA para comunicarem e para controlo do interface de utilizador.

Assim é fundamental autorizar a execução de macros, na opção de Segurança das duas aplicações.

A base de dados e a pasta que contém os manuais de utilização têm de estar na mesma directoria do livro de Excel.

3.4. Conclusão

Era desejável trabalhar com valores realistas, mas a base de dados obtida não continha dados suficientes para três anos, pelo que houve necessidade de gerar os que faltavam.

Para o funcionamento do aplicativo é preciso autorizar a execução de código de programação e os ficheiros têm de estar todos na mesma directoria.

O capítulo 4 mostra o que é o critério RM e as outras métricas que usamos.

4. RM e as métricas

4.1. Introdução

Neste capítulo clarifica-se em que consiste o critério RM e o significado das outras métricas que são calculadas.

4.2. O critério de segmentação RM

O critério RM propõe-se a classificar cada cliente em classes, de acordo com a data da última compra (Recenticidade – R) e do valor global das compras (valor Monetário – M). A quantidade das classes deve ser inicialmente estipulada. Caso se escolham 5 classes para cada atributo, com os números 1 a 5, obtemos 25 classificações diferentes. Assim, um cliente 55 é um cliente muito recente e de alto valor de compras, enquanto que um cliente 11 é precisamente o contrário.

Neste trabalho esta classificação é atribuída da seguinte forma: para o R é feita a computação de quantos dias decorreram entre a data da última compra e a data actual. Para o efeito consideramos a data actual como sendo 01/01/2001. A tabela é depois ordenada de forma crescente por este valor. A seguir classifica-se os primeiros n/5 clientes com o número 5 (mais recentes), os segundos n/5 clientes com o número 4 e assim sucessivamente. Operação semelhante decorre para o M, só que desta vez calcula-se o valor médio das compras por cliente e ordena-se a partir de aí, sendo a classificação feita nos mesmos termos do R.

O utilizador tem a liberdade de escolher entre 2 a 5 segmentos.

4.3. Taxa de Abandono (churn rate)

O objectivo da taxa de abandono é medir a quantidade de clientes que deixaram de comprar.

Calcula-se com a seguinte fórmula:

$$\frac{\text{clientes iniciais} - \text{clientes finais}}{\text{clientes iniciais}}$$

Uma taxa de abandono negativo significa que a empresa ganhou mais clientes do que perdeu.

4.4. Taxa de retenção

Representa a percentagem de clientes que a empresa conseguiu segurar como compradores activos.

Calcula-se: $1 - \text{taxa abandono}$

4.5. LTV

LTV (Life Time Value) – o valor do tempo de vida, analisa o histórico dos clientes para calcular o lucro esperado das vendas a um cliente no futuro.

Neste trabalho reproduz-se o cálculo do LTV a 3 anos apresentado por Arthur Middleton Hughes conforme figura 4-1:

	Acquisition Year	Second Year	Third Year
Customers	100,000	60,000	42,000
Retention Rate	60%	70%	80%
Orders per Year	1.8	2.5	3
Avg Order Size	\$90	\$95	\$100
Total Revenue	\$16,200,000	\$14,250,000	\$12,600,000
Costs	70%	65%	65%
Cost of Sales	\$11,340,000	\$9,262,500	\$8,190,000
Acquisition/Mkt. Cost	\$55	\$20	\$20
Marketing Costs	\$5,500,000	\$1,200,000	\$840,000
Total Costs	\$16,840,000	\$10,462,500	\$9,030,000
Gross Profit	(\$640,000)	\$3,787,500	\$3,570,000
Discount Rate	1	1.16	1.35
Net Present Value	(\$640,000)	\$3,265,086	\$2,644,444
Cumulative NPV Profit	(\$640,000)	\$2,625,086	\$5,269,531
Customer LTV	(\$6)	\$26	\$53

Figura 4 - 1: Cálculo do LTV conforme <http://www.dbmarketing.com/articles/Art251a.htm>

4.6. LTV individual

Podemos calcular o LTV de cada cliente individualizado, a partir do LTV médio calculado anteriormente, através da seguinte fórmula:

$$LTV_{Cliente_i} = \frac{TotalCompras_i}{média\ global\ de\ vendas} * LTV_{3^{o}\ ano}$$

Neste trabalho procedeu-se à segmentação dos LTV individuais com a técnica já referida, para que se possa fazer uma comparação com o RM.

4.7. Vantagens dos indicadores

A utilização destes indicadores, para além de medir a fidelidade do cliente, também permite acompanhar o impacto das opções de gestão sobre o valor dos clientes.

Obtém-se assim uma boa base para a selecção de clientes e adaptação das estratégias de comunicação da empresa.

4.8. Conclusão

Apresentaram-se as técnicas e fórmulas para aplicação e cálculo do critério RM e das métricas de cliente que são usadas neste trabalho.

O capítulo 5 mostra o funcionamento das consultas e das tabelas da base de dados.

5. Access e SQL

5.1. Introdução

Este capítulo trata de mostrar a constituição da base de dados e como se implementou todo o processo a nível de consultas de modo a permitir obter os dados a serem mostrados ao utilizador.

5.2. As consultas

Para iniciar o processo só são indispensáveis as tabelas já anteriormente referidas: Orders e Clientes, e estas nunca são alteradas.

Toda a informação e outras tabelas necessárias são conseguidas através das seguintes consultas:

- **Contar Clientes global**
 - Conta o nº de clientes por cada ano
 - Utiliza a tabela “Orders”
- **Contar Vendas Ano tx ret cria tabela**
 - Cria/substitui a tabela “ContaValoresAno”
 - Recebe como parâmetro o ano actual
 - Utiliza as tabelas “Clientes” e “Orders”
- **LTV ind calculado**
 - Calcula, para cada cliente, o valor médio de vendas, e o valor médio global para cálculo do LTV individual; calcula também o valor máximo dos LTV individual e atribui o segmento respectivo
 - Recebe como parâmetros o LTV global do 3º ano e o nº de segmentos pretendidos
 - Utiliza a tabela “Orders”
- **LTV_Global**
 - Consulta união que implementa o cálculo do LTV global a 3 anos conforme o exemplo de Arthur Middleton Hughes (figura 4-1), como se mostra na figura 5-1:
 - Recebe como parâmetros, para cada ano, o custo de produção (%), custo médio de cada cliente para a empresa, taxa de juro de mercado, o risco da operação (entre 1 e 2) e o prazo de pagamento normal concedido aos clientes (em dias)
 - Calcula-se ainda a taxa de retenção e o churn a cada ano
 - Utiliza as tabelas “ContaValoresAno”, “MediaValoresAno” e “Tabela1”; esta última só é necessária para introduzir a taxa de desconto

Category	Ano1	Ano2	Ano3
Ano	1998	1999	2000
Cientes_Ano	3840	2579	4028
TxRet_Ano	0,9	0,67	1,56
Churn_Ano	0,098	0,328	-0,562
Enc_Cliente	11,1	5,8	3,1
Avg_Val_Enc	424,64	418,33	393,36
ReceitaAnual	18099855,36	6257463,81	4911807,65
CustoProd	1809985,54	1251492,76	1473542,3
OutrosCustos	130560	90265	217512
Tot_Custos	1940545,54	1341757,76	1691054,3
Lucro_Liq	16159309,82	4915706,05	3220753,35
TxDesconto	1	1,08	1,17
VAL	16159309,82	4551579,68	2752780,64
VAL_Acum	16159309,82	20710889,5	23463670,14
LTV	4208,15	5393,46	6110,33

Figura 5 - 1: LTV calculado no MS Access

- **LTVInd_RM**
 - Junta a informação dos segmentos devolvida pelas consultas “LTV ind calculado” e “RM calculado por segmento”, para comparação
 - Recebe como parâmetros o LTV global do 3º ano e o nº de segmentos pretendidos
- **MaxData**
 - Apresenta informação, por cliente, das datas (mínima e máxima) das compras, nº dias desde a última compra (R) e total gasto (M)
 - Utiliza a tabela “Orders”
- **MaxR_MaxM**
 - Calcula o máximo de dias (MaxR) e de valor (MaxM) dos clientes
 - Utiliza as tabelas “Clientes” e “Orders”
- **Media Vendas Ano**
 - Calcula o nº total de vendas e a média de valor por ano (últimos 3 anos)
 - Recebe como parâmetro o ano actual
 - Utiliza a tabela “Orders”
- **Media Vendas Ano cria tabela**
 - Calcula o nº total de vendas e a média de valor por ano (últimos 3 anos) e cria a tabela “MediaValoresAno”

- Recebe como parâmetro o ano actual
 - Utiliza a tabela “Orders”
- **OrdersTotalDia**
 - Calcula o valor de vendas por dia
 - Utiliza a tabela “Orders”
- **OrdersTotalMes**
 - Calcula o valor de vendas por mês
 - Utiliza a tabela “Orders”
- **ordersTotalMêsLista**
 - cria uma lista com os meses em que houve vendas
 - Utiliza a tabela “Orders”
- **Criar_Clientes_R**
 - Obtém os dados necessários para cálculo do R
 - Cria a tabela “Clientes_R”
- **Criar_Clientes_M**
 - Obtém os dados necessários para cálculo do M
 - Cria a tabela “Clientes_R”
- **Cria_NumSeq_R**
 - Acrescenta uma coluna à tabela “Clientes_R” com numeração sequencial, requisito para a segmentação
- **Cria_NumSeq_M**
 - Acrescenta uma coluna à tabela “Clientes_M” com numeração sequencial, requisito para a segmentação
- **Cria_Clientes_Seg_R**
 - Calcula os segmentos do R por cliente
 - Acrescenta uma coluna à tabela “Clientes_R” com o segmento e cria a tabela “Clientes_Seg_R”
- **Cria_Clientes_Seg_M**
 - Calcula os segmentos do M por cliente
 - Acrescenta uma coluna à tabela “Clientes_M” com o segmento e cria a tabela “Clientes_Seg_M”
- **RM calculado por segmento**
 - Faz a junção das tabelas “Clientes_Seg_R” e “Clientes_Seg_M” depois de segmentadas
- **RM calculado por segmento cria tabela**
 - Faz a junção das tabelas “Clientes_Seg_R” e “Clientes_Seg_M” depois de segmentadas
 - Cria tabela “TabelaRMporSegmento”

5.3. As tabelas

As seguintes tabelas ficam a constituir a base de dados depois de executadas todas as consultas:

- **Clientes**
 - *CustNum -> Nome*
- **Orders**
 - *ID -> Custnum, Order Date, Dollars, Quantity*
- **Tabela1**

- *Nome*
- Útil para a consulta união “LTV_Global”
- **ContaValoresAno**
 - Tabela criada/substituída pela consulta “Contar Vendas Ano tx ret cria tabela”
 - Contém informação sobre o nº total de clientes, e os valores dos últimos 3 anos (nº clientes, nº vendas, vendas/cliente, tx retenção e churn)
- **MediaValoresAno**
 - Tabela criada/substituída pela consulta “Media Vendas Ano cria tabela”
 - Contém o nº total de vendas e a média de valor por ano (últimos 3 anos)
- **Clientes_R**
 - Tabela criada/substituída pela consulta “Criar_Clientes_R”
 - Contém, por cliente, a data mais antiga e a mais recente das suas compras, a quantidade de compras efectuadas, e o nº dias que decorreram desde a última compra (R_Val)
- **Clientes_M**
 - Tabela criada/substituída pela consulta “Criar_Clientes_M”
 - Contém, por cliente, o valor médio de compras (M_Val)
- **Clientes_Seg_R**
 - Tabela criada/substituída pela consulta “Cria_Clientes_Seg_R”
 - É a reutilização da tabela “Clientes_R” mas já contém o segmento R
- **Clientes_Seg_M**
 - Tabela criada/substituída pela consulta “Cria_Clientes_Seg_M”
 - É a reutilização da tabela “Clientes_M” mas já contém o segmento M
- **TabelaRMporSegmento**
 - Tabela criada/substituída pela consulta “RM calculado por segmento cria tabela”
 - Contém, por cliente, a data mais antiga e a mais recente das suas compras, a quantidade de compras efectuadas, o valor médio gasto, o nº dias da compra menos recente de todos os clientes (MaxR), o valor médio de compras mais alto de todos os clientes (MaxM), e segmentos R e M
 - Os segmentos vão de 2 a 5 e são estipulados pelo utilizador

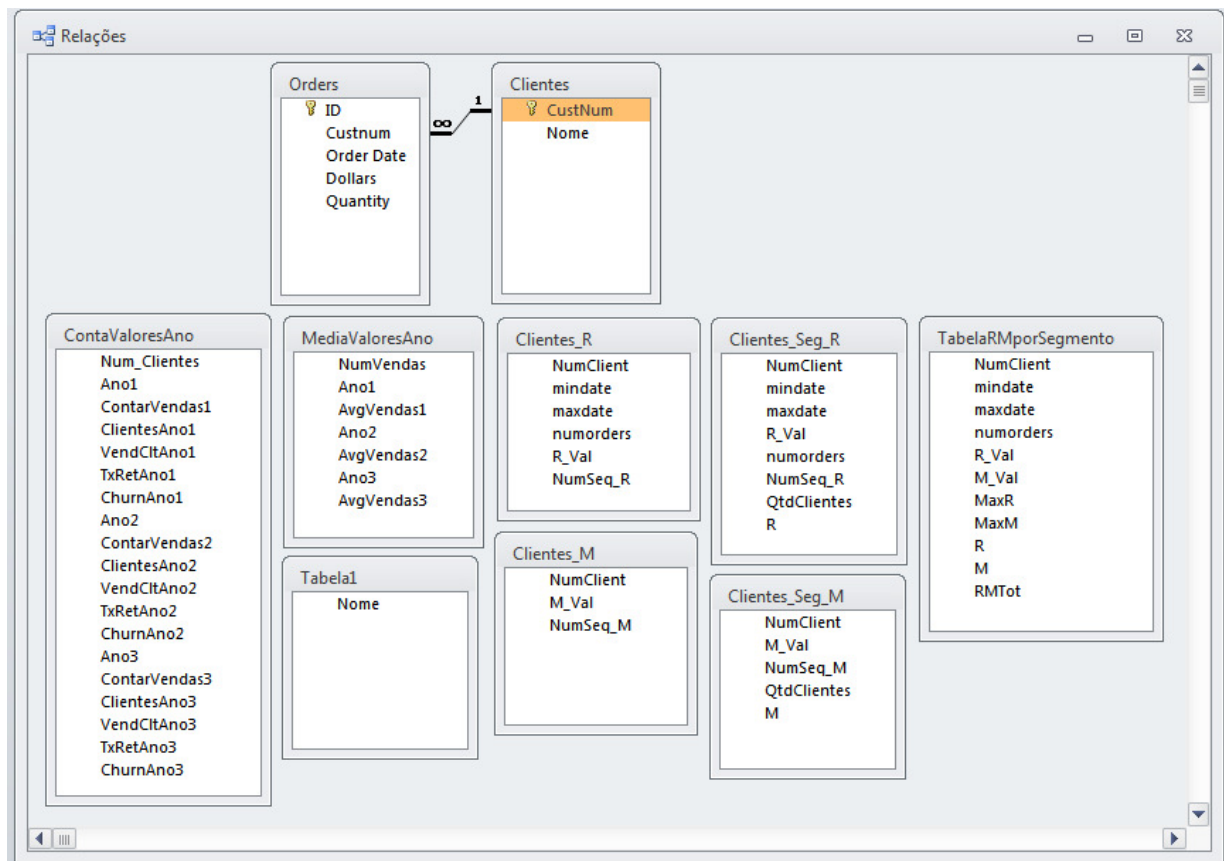


Figura 5 - 2: Tabelas na Base de Dados

5.4. Código SQL

A estrutura de cada uma das consultas em SQL é a seguinte:

Contar Clientes global

```
SELECT count(Expr1) AS N, Ano
FROM [SELECT DISTINCT (Orders.Custnum) AS Expr1, Year(Orders.[Order Date]) AS Ano
FROM Orders
GROUP BY Year(Orders.[Order Date]), Orders.Custnum]. AS C
GROUP BY Ano
ORDER BY Ano DESC;
```

Contar Vendas Ano tx ret cria tabela

```
SELECT Count(Clientes.CustNum) AS Num_Clientes,

(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order Date])=[AnoActual]-3) AS Ano1,
```

```
(SELECT Count(Orders.ID) AS ContarVendas1 FROM Orders group by Year([Order Date]) having
Year([Order Date])=[AnoActual]-3) AS ContarVendas1,
```

```
(SELECT count(*) AS N FROM (SELECT DISTINCT (Custnum) AS QtdClient, Year([Order Date]) AS
AnoCompra FROM Orders GROUP BY Year([Order Date]), Custnum HAVING (Year(Orders.[Order
Date]))=[AnoActual]-3)) AS ClientesAno1,
round(ContarVendas1/ClientesAno1,1) AS VendCltAno1,
round(1-((Num_Clientes-ClientesAno1)/Num_Clientes),2) AS TxRetAno1,
round((Num_Clientes-ClientesAno1)/Num_Clientes,3) AS ChurnAno1,
```

```
(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order
Date])=[AnoActual]-2) AS Ano2,
```

```
(SELECT Count(Orders.ID) AS ContarVendas1 FROM Orders group by Year([Order Date]) having
Year([Order Date])=[AnoActual]-2) AS ContarVendas2,
```

```
(SELECT count(*) AS N FROM (SELECT DISTINCT (Custnum) AS QtdClient, Year([Order Date]) AS
AnoCompra FROM Orders GROUP BY Year([Order Date]), Custnum HAVING (Year(Orders.[Order
Date]))=[AnoActual]-2)) AS ClientesAno2,
round(ContarVendas2/ClientesAno2,1) AS VendCltAno2,
round(1-((ClientesAno1-ClientesAno2)/ClientesAno1),2) AS TxRetAno2,
round((ClientesAno1-ClientesAno2)/ClientesAno1,3) AS ChurnAno2,
```

```
(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order
Date])=[AnoActual]-1) AS Ano3,
```

```
(SELECT Count(Orders.ID) AS ContarVendas1 FROM Orders group by Year([Order Date]) having
Year([Order Date])=[AnoActual]-1) AS ContarVendas3,
```

```
(SELECT count(*) AS N FROM (SELECT DISTINCT (Custnum) AS QtdClient, Year([Order Date]) AS
AnoCompra FROM Orders GROUP BY Year([Order Date]), Custnum HAVING (Year(Orders.[Order
Date]))=[AnoActual]-1)) AS ClientesAno3,
round(ContarVendas3/ClientesAno3,1) AS VendCltAno3,
round(1-((ClientesAno2-ClientesAno3)/ClientesAno2),2) AS TxRetAno3,
round((ClientesAno2-ClientesAno3)/ClientesAno2,3) AS ChurnAno3 INTO ContaValoresAno
FROM Clientes;
```

LTV ind calculado

```
SELECT Custnum AS NumClient,
Round(Avg(Dollars),0) AS M_Val,
```

```
(SELECT Avg(Dollars) AS MediaDeVendas FROM Orders) AS AvgGlobal,
Round((M_Val/AvgGlobal)*LTV,2) AS LTV_Ind,
```

```
(SELECT Max(LTV_Ind)
FROM
```

```
(SELECT Custnum, Round(Avg(Dollars),0) AS M_Val,
```

```
(SELECT Avg(Dollars) AS MediaDeVendas FROM Orders) AS AvgGlobal,
Round((M_Val/AvgGlobal)*LTV,2) AS LTV_Ind FROM Orders group by Custnum)) AS MaxLTV,
```

```
Switch(
[segmentos:]=2 And [LTV_Ind]<[MaxLTV]/2,1,
[segmentos:]=2 And [LTV_Ind]>=[MaxLTV]/2,2,
[segmentos:]=3 And [LTV_Ind]<[MaxLTV]/3,1,
[segmentos:]=3 And [LTV_Ind]>=[MaxLTV]/3 And [LTV_Ind]<2*[MaxLTV]/3,2,
[segmentos:]=3 And [LTV_Ind]>=2*[MaxLTV]/3,3,
[segmentos:]=4 And [LTV_Ind]<[MaxLTV]/4,1,
[segmentos:]=4 And [LTV_Ind]>=[MaxLTV]/4 And [LTV_Ind]<2*[MaxLTV]/4,2,
[segmentos:]=4 And [LTV_Ind]>=2*[MaxLTV]/4 And [LTV_Ind]<3*[MaxLTV]/4,3,
[segmentos:]=4 And [LTV_Ind]>=3*[MaxLTV]/4,4,
[segmentos:]=5 And [LTV_Ind]<[MaxLTV]/5,1,
[segmentos:]=5 And [LTV_Ind]>=[MaxLTV]/5 And [LTV_Ind]<2*[MaxLTV]/5,2,
[segmentos:]=5 And [LTV_Ind]>=2*[MaxLTV]/5 And [LTV_Ind]<3*[MaxLTV]/5,3,
[segmentos:]=5 And [LTV_Ind]>=3*[MaxLTV]/5 And [LTV_Ind]<4*[MaxLTV]/5,4,
[segmentos:]=5 And [LTV_Ind]>=4*[MaxLTV]/5,5) AS Seg_LTV
```

```
FROM Orders
GROUP BY Orders.Custnum;
```

LTV_Global

```
SELECT "Ano" as Category, Ano1, Ano2, Ano3
FROM ContaValoresAno
```

```
UNION ALL
select "Clientes_Ano", ClientesAno1, ClientesAno2, ClientesAno3
FROM ContaValoresAno
```

```
UNION ALL
select "TxRet_Ano", TxRetAno1, TxRetAno2, TxRetAno3
FROM ContaValoresAno
```

```
UNION ALL
select "Churn_Ano", ChurnAno1, ChurnAno2, ChurnAno3
FROM ContaValoresAno
```

```
UNION ALL
select "Enc_Cliente", VendCltAno1, VendCltAno2, VendCltAno3
FROM ContaValoresAno
```

```
UNION ALL
select "Avg_Val_Enc", AvgVendas1, AvgVendas2, AvgVendas3
FROM MediaValoresAno
```

```
UNION ALL
```



```

select "ReceitaAnual", Round(ClientesAno1*VendCltAno1*AvgVendas1,2),
Round(ClientesAno2*VendCltAno2*AvgVendas2,2), Round(ClientesAno3*VendCltAno3*AvgVendas3,2)
FROM ContaValoresAno, MediaValoresAno

```

UNION ALL

```

select "CustoProd", Round(Round(ClientesAno1*VendCltAno1*AvgVendas1,2)*[CustoProd1]/100,2),
Round(Round(ClientesAno2*VendCltAno2*AvgVendas2,2)*[CustoProd2]/100,2),
Round(Round(ClientesAno3*VendCltAno3*AvgVendas3,2)*[CustoProd3]/100,2)
FROM ContaValoresAno, MediaValoresAno

```

UNION ALL

```

select "OutrosCustos", Round(ClientesAno1*CustoCl1,2), Round(ClientesAno2*CustoCl2,2),
Round(ClientesAno3*CustoCl3,2)
FROM ContaValoresAno

```

UNION ALL

```

select "Tot_Custos", sum(a), sum(b), sum(c)
from(
select
Round(Round(ClientesAno1*VendCltAno1*AvgVendas1,2)*[CustoProd1]/100,2)+Round(ClientesAno1*CustoCl1,2) as a,
Round(Round(ClientesAno2*VendCltAno2*AvgVendas2,2)*[CustoProd2]/100,2)+Round(ClientesAno2*CustoCl2,2) as b,
Round(Round(ClientesAno3*VendCltAno3*AvgVendas3,2)*[CustoProd3]/100,2)+Round(ClientesAno3*CustoCl3,2) as c
FROM ContaValoresAno, MediaValoresAno)

```

UNION ALL

```

select "Lucro_Liq", sum(a), sum(b), sum(c)
from(
select Round(ClientesAno1*VendCltAno1*AvgVendas1,2)-
(Round(Round(ClientesAno1*VendCltAno1*AvgVendas1,2)*[CustoProd1]/100,2))-
(Round(ClientesAno1*CustoCl1,2)) as a, Round(ClientesAno2*VendCltAno2*AvgVendas2,2)-
Round(Round(ClientesAno2*VendCltAno2*AvgVendas2,2)*[CustoProd2]/100,2))-
(Round(ClientesAno2*CustoCl2,2)) as b, Round(ClientesAno3*VendCltAno3*AvgVendas3,2)-
Round(Round(ClientesAno3*VendCltAno3*AvgVendas3,2)*[CustoProd3]/100,2)) -
(Round(ClientesAno3*CustoCl3,2)) as c
FROM ContaValoresAno, MediaValoresAno)

```

UNION ALL

```

select "TxDesconto",round((1+(txJuro1/100)*risco1)^(0+(prazoPag1/365)),2),
round((1+(txJuro2/100)*risco2)^(1+(prazoPag2/365)),2),
round((1+(txJuro3/100)*risco3)^(2+(prazoPag3/365)),2) from Tabela1

```

UNION ALL

```

select "VAL", round(sum(a),2), round(sum(b),2), round(sum(c),2)
from(
select (Round(ClientesAno1*VendCltAno1*AvgVendas1,2)-
(Round(Round(ClientesAno1*VendCltAno1*AvgVendas1,2)*[CustoProd1]/100,2))-
(Round(ClientesAno1*CustoCl1,2))) / (round((1+(txJuro1/100)*risco1)^(0+(prazoPag1/365)),2)) as a,

```

```

(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)-
Round(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)*[CustoProd2]/100,2)-
(Round(ClientesAno2*CustoCl2,2))) / (round((1+(txJuro2/100)*risco2)^(1+(prazoPag2/365)),2)) as b,
(Round(ClientesAno3*VendClItAno3*AvgVendas3,2)-
Round(Round(ClientesAno3*VendClItAno3*AvgVendas3,2)*[CustoProd3]/100,2) -
(Round(ClientesAno3*CustoCl3,2))) / (round((1+(txJuro3/100)*risco3)^(2+(prazoPag3/365)),2)) as c
FROM ContaValoresAno, MediaValoresAno)

```

UNION ALL

```

select "VAL_Acum", round(sum(a),2), round(sum(b),2), round(sum(c),2)
from(
Select (Round(ClientesAno1*VendClItAno1*AvgVendas1,2)-
(Round(Round(ClientesAno1*VendClItAno1*AvgVendas1,2)*[CustoProd1]/100,2))-
(Round(ClientesAno1*CustoCl1,2))) / (round((1+(txJuro1/100)*risco1)^(0+(prazoPag1/365)),2)) as a,
(Round(ClientesAno1*VendClItAno1*AvgVendas1,2)-
(Round(Round(ClientesAno1*VendClItAno1*AvgVendas1,2)*[CustoProd1]/100,2))-
(Round(ClientesAno1*CustoCl1,2))) / (round((1+(txJuro1/100)*risco1)^(0+(prazoPag1/365)),2)) +
(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)-
Round(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)*[CustoProd2]/100,2)-
(Round(ClientesAno2*CustoCl2,2))) / (round((1+(txJuro2/100)*risco2)^(1+(prazoPag2/365)),2)) as b,
(Round(ClientesAno1*VendClItAno1*AvgVendas1,2)-
(Round(Round(ClientesAno1*VendClItAno1*AvgVendas1,2)*[CustoProd1]/100,2))-
(Round(ClientesAno1*CustoCl1,2))) / (round((1+(txJuro1/100)*risco1)^(0+(prazoPag1/365)),2)) +
(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)-
Round(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)*[CustoProd2]/100,2)-
(Round(ClientesAno2*CustoCl2,2))) / (round((1+(txJuro2/100)*risco2)^(1+(prazoPag2/365)),2)) +
(Round(ClientesAno3*VendClItAno3*AvgVendas3,2)-
Round(Round(ClientesAno3*VendClItAno3*AvgVendas3,2)*[CustoProd3]/100,2) -
(Round(ClientesAno3*CustoCl3,2))) / (round((1+(txJuro3/100)*risco3)^(2+(prazoPag3/365)),2)) as c
FROM ContaValoresAno, MediaValoresAno)

```

UNION ALL

```

select "LTV", round(sum(a),2), round(sum(b),2), round(sum(c),2)
from(
select ((Round(ClientesAno1*VendClItAno1*AvgVendas1,2)-
(Round(Round(ClientesAno1*VendClItAno1*AvgVendas1,2)*[CustoProd1]/100,2))-
(Round(ClientesAno1*CustoCl1,2))) / (round((1+(txJuro1/100)*risco1)^(0+(prazoPag1/365)),2))) /
ClientesAno1 as a, ((Round(ClientesAno1*VendClItAno1*AvgVendas1,2)-
(Round(Round(ClientesAno1*VendClItAno1*AvgVendas1,2)*[CustoProd1]/100,2))-
(Round(ClientesAno1*CustoCl1,2))) / (round((1+(txJuro1/100)*risco1)^(0+(prazoPag1/365)),2)) +
(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)-
Round(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)*[CustoProd2]/100,2)-
(Round(ClientesAno2*CustoCl2,2))) / (round((1+(txJuro2/100)*risco2)^(1+(prazoPag2/365)),2))) /
ClientesAno1 as b, ((Round(ClientesAno1*VendClItAno1*AvgVendas1,2)-
(Round(Round(ClientesAno1*VendClItAno1*AvgVendas1,2)*[CustoProd1]/100,2))-
(Round(ClientesAno1*CustoCl1,2))) / (round((1+(txJuro1/100)*risco1)^(0+(prazoPag1/365)),2)) +
(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)-
Round(Round(ClientesAno2*VendClItAno2*AvgVendas2,2)*[CustoProd2]/100,2)-
(Round(ClientesAno2*CustoCl2,2))) / (round((1+(txJuro2/100)*risco2)^(1+(prazoPag2/365)),2)) +
(Round(ClientesAno3*VendClItAno3*AvgVendas3,2)-

```

```

Round(Round(ClientesAno3*VendCltAno3*AvgVendas3,2)*[CustoProd3]/100,2) -
(Round(ClientesAno3*CustoCl3,2))) / (round((1+(txJuro3/100)*risco3)^(2+(prazoPag3/365)),2))) /
ClientesAno1 as c
FROM ContaValoresAno, MediaValoresAno);
  
```

LTVInd_RM

```

SELECT [LTV ind calculado].NumClient, [LTV ind calculado].Seg_LTV, [RM calculado por segmento].R,
[RM calculado por segmento].M, [RM calculado por segmento].RMTot
FROM [LTV ind calculado] INNER JOIN [RM calculado por segmento] ON [LTV ind calculado].NumClient
= [RM calculado por segmento].NumClient;
  
```

MaxData

```

SELECT Orders.Custnum, Min(Orders.[Order Date]) AS mindate, Max(Orders.[Order Date]) AS maxdate,
Count(*) AS numorders, DateDiff("d",maxdate,#1/1/2001#) AS R, Round(Sum([Dollars]),0) AS M
FROM Orders
GROUP BY Orders.Custnum
ORDER BY Orders.Custnum;
  
```

MaxR_MaxM

```

SELECT Max(R) AS MaxR,

(SELECT Max(M) from(SELECT Clientes.CustNum as NumClient, round(Avg(Dollars),0) AS M FROM
Clientes INNER JOIN Orders ON Clientes.CustNum = Orders.Custnum GROUP BY Clientes.CustNum
ORDER BY Clientes.CustNum)) AS MaxM

FROM
(SELECT Clientes.CustNum as NumClient,Max(Orders.[Order Date]) AS maxdate,
DateDiff("d",maxdate,#1/1/2001#) AS R FROM Clientes INNER JOIN Orders ON Clientes.CustNum =
Orders.Custnum
GROUP BY Clientes.CustNum
ORDER BY Clientes.CustNum)AS MaxR;
  
```

Media Vendas Ano

```

SELECT Count(ID) AS NumVendas,

(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order
Date])=[AnoActual]-3) AS Ano1,

(SELECT round(Avg(Dollars),2) AS MediaDeDollars FROM Orders GROUP BY Year([Order Date])
HAVING (Year([Order Date]))=[AnoActual]-3) AS AvgVendas1,
  
```

```
(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order Date])=[AnoActual]-2) AS Ano2,
```

```
(SELECT round(Avg(Dollars),2) AS MediaDeDollars FROM Orders GROUP BY Year([Order Date])  
HAVING (Year([Order Date]))=[AnoActual]-2) AS AvgVendas2,
```

```
(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order Date])=[AnoActual]-1) AS Ano3,
```

```
(SELECT round(Avg(Dollars),2) AS MediaDeDollars FROM Orders GROUP BY Year([Order Date])  
HAVING (Year([Order Date]))=[AnoActual]-1) AS AvgVendas3  
FROM Orders;
```

Media Vendas Ano cria tabela

```
SELECT Count(ID) AS NumVendas,
```

```
(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order Date])=[AnoActual]-3) AS Ano1,
```

```
(SELECT round(Avg(Dollars),2) AS MediaDeDollars FROM Orders GROUP BY Year([Order Date])  
HAVING (Year([Order Date]))=[AnoActual]-3) AS AvgVendas1,
```

```
(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order Date])=[AnoActual]-2) AS Ano2,
```

```
(SELECT round(Avg(Dollars),2) AS MediaDeDollars FROM Orders GROUP BY Year([Order Date])  
HAVING (Year([Order Date]))=[AnoActual]-2) AS AvgVendas2,
```

```
(SELECT Year([Order Date]) FROM Orders group by Year([Order Date]) having Year([Order Date])=[AnoActual]-1) AS Ano3,
```

```
(SELECT round(Avg(Dollars),2) AS MediaDeDollars FROM Orders GROUP BY Year([Order Date])  
HAVING (Year([Order Date]))=[AnoActual]-1) AS AvgVendas3
```

```
INTO MediaValoresAno  
FROM Orders;
```

OrdersTotalDia

```
SELECT Format([Order Date],'mmm/yyyy') AS mesData, cdate(Format([Order Date],'dd/mm/yyyy')) AS  
diaData, Sum(Orders.Dollars) AS SomaDeDollars
```

```
FROM Orders  
GROUP BY Format([Order Date],'mmm/yyyy'), cdate(Format([Order Date],'dd/mm/yyyy'))  
ORDER BY cdate(Format([Order Date],'dd/mm/yyyy'));
```

ordersTotalMêsLista

```

SELECT Format([Order Date],'mmm/yyyy') AS mesData
FROM Orders
GROUP BY Format([Order Date],'mmm/yyyy')
ORDER BY cdate(Format([Order Date],'mmm/yyyy'));
```

Criar_Clientes_R

```

SELECT H.NumClient, H.mindate, H.maxdate, H.numorders, H.R_Val INTO Clientes_R
FROM (SELECT CustNum AS NumClient, Min(Orders.[Order Date]) AS mindate, Max(Orders.[Order Date])
AS maxdate, Count(*) AS numorders, DateDiff("d",maxdate,#1/1/2001#) AS R_Val FROM Orders GROUP
BY CustNum ORDER BY CustNum) AS H
ORDER BY H.R_Val;
```

Criar_Clientes_M

```

SELECT H.NumClient, H.M_Val INTO Clientes_M
FROM (SELECT CustNum AS NumClient, round(Avg(Dollars),0) AS M_Val FROM Orders GROUP BY
CustNum ORDER BY CustNum) AS H
ORDER BY H.M_Val;
```

Cria_NumSeq_R

```

ALTER TABLE Clientes_R ADD COLUMN NumSeq_R COUNTER
```

Cria_NumSeq_M

```

ALTER TABLE Clientes_M ADD COLUMN NumSeq_M COUNTER
```

Cria_Clientes_Seg_R

```

SELECT Clientes_R.NumClient, Clientes_R.mindate, Clientes_R.maxdate, Clientes_R.R_Val,
Clientes_R.numorders, Clientes_R.NumSeq_R, (SELECT count(*) from (select distinct Clientes.CustNum

FROM Clientes INNER JOIN Orders ON Clientes.CustNum = Orders.Custnum)) AS QtdClientes,

Switch(
[segmentos:]=2 And [NumSeq_R]<[QtdClientes]/2,2,
[segmentos:]=2 And [NumSeq_R]>=[QtdClientes]/2,1,
[segmentos:]=3 And [NumSeq_R]<[QtdClientes]/3,3,
[segmentos:]=3 And [NumSeq_R]>=[QtdClientes]/3 And [NumSeq_R]<2*[QtdClientes]/3,2,
[segmentos:]=3 And [NumSeq_R]>=2*[QtdClientes]/3,1,
[segmentos:]=4 And [NumSeq_R]<[QtdClientes]/4,4,
[segmentos:]=4 And [NumSeq_R]>=[QtdClientes]/4 And [NumSeq_R]<2*[QtdClientes]/4,3,
```

```
[segmentos:]=4 And [NumSeq_R]>=2*[QtdClientes]/4 And NumSeq_R<3*[QtdClientes]/4,2,
[segmentos:]=4 And [NumSeq_R]>=3*[QtdClientes]/4,1,
[segmentos:]=5 And [NumSeq_R]<[QtdClientes]/5,5,
[segmentos:]=5 And [NumSeq_R]>=[QtdClientes]/5 And [NumSeq_R]<2*[QtdClientes]/5,4,
[segmentos:]=5 And [NumSeq_R]>=2*[QtdClientes]/5 And NumSeq_R<3*[QtdClientes]/5,3,
[segmentos:]=5 And [NumSeq_R]>=3*[QtdClientes]/5 And NumSeq_R<4*[QtdClientes]/5,2,
[segmentos:]=5 And [NumSeq_R]>=4*[QtdClientes]/5,1) AS R
```

```
INTO Clientes_Seg_R
FROM Clientes_R;
```

Cria_Clientes_Seg_M

```
SELECT Clientes_M.NumClient, Clientes_M.M_Val, Clientes_M.NumSeq_M,

(SELECT count(*) from (select distinct Clientes.CustNum
FROM Clientes INNER JOIN Orders ON Clientes.CustNum = Orders.Custnum)) AS QtdClientes,

Switch(
[segmentos:]=2 And [NumSeq_M]<[QtdClientes]/2,2,
[segmentos:]=2 And [NumSeq_M]>=[QtdClientes]/2,1,
[segmentos:]=3 And [NumSeq_M]<[QtdClientes]/3,3,
[segmentos:]=3 And [NumSeq_M]>=[QtdClientes]/3 And [NumSeq_M]<2*[QtdClientes]/3,2,
[segmentos:]=3 And [NumSeq_M]>=2*[QtdClientes]/3,1,
[segmentos:]=4 And [NumSeq_M]<[QtdClientes]/4,4,
[segmentos:]=4 And [NumSeq_M]>=[QtdClientes]/4 And [NumSeq_M]<2*[QtdClientes]/4,3,
[segmentos:]=4 And [NumSeq_M]>=2*[QtdClientes]/4 And NumSeq_M<3*[QtdClientes]/4,2,
[segmentos:]=4 And [NumSeq_M]>=3*[QtdClientes]/4,1,
[segmentos:]=5 And [NumSeq_M]<[QtdClientes]/5,5,
[segmentos:]=5 And [NumSeq_M]>=[QtdClientes]/5 And [NumSeq_M]<2*[QtdClientes]/5,4,
[segmentos:]=5 And [NumSeq_M]>=2*[QtdClientes]/5 And NumSeq_M<3*[QtdClientes]/5,3,
[segmentos:]=5 And [NumSeq_M]>=3*[QtdClientes]/5 And NumSeq_M<4*[QtdClientes]/5,2,
[segmentos:]=5 And [NumSeq_M]>=4*[QtdClientes]/5,1) AS M
```

```
INTO Clientes_Seg_M
FROM Clientes_M;
```

RM calculado por segmento

```
SELECT Clientes_Seg_R.NumClient, Clientes_Seg_R.mindate, Clientes_Seg_R.maxdate,
Clientes_Seg_R.numorders, Clientes_Seg_R.R_Val, Clientes_Seg_M.M_Val,

(SELECT Max(R) from

(SELECT CustNum as NumClient, Max(Orders.[Order Date]) AS maxdate, DateDiff("d",maxdate,#1/1/2001#)
AS R
FROM Orders
GROUP BY CustNum
```

```

ORDER BY CustNum)) AS MaxR,

(SELECT Max(M) from

(SELECT CustNum as NumClient, round(Avg(Dollars),0) AS M
FROM Orders
GROUP BY CustNum
ORDER BY CustNum))

AS MaxM, Clientes_Seg_R.R, Clientes_Seg_M.M, R*10+M AS RMTot

FROM Clientes_Seg_R INNER JOIN Clientes_Seg_M ON
Clientes_Seg_R.NumClient=Clientes_Seg_M.NumClient

ORDER BY Clientes_Seg_M.M_Val;
  
```

RM calculado por segmento cria tabela

```

SELECT Clientes_Seg_R.NumClient, Clientes_Seg_R.mindate, Clientes_Seg_R.maxdate,
Clientes_Seg_R.numorders, Clientes_Seg_R.R_Val, Clientes_Seg_M.M_Val,

(SELECT Max(R) from (SELECT CustNum as NumClient, Max(Orders.[Order Date]) AS maxdate,
DateDiff("d",maxdate,#1/1/2001#) AS R

FROM Orders
GROUP BY CustNum
ORDER BY CustNum)) AS MaxR,

(SELECT Max(M) from

(SELECT CustNum as NumClient, round(Avg(Dollars),0) AS M
FROM Orders GROUP BY CustNum
ORDER BY CustNum))

AS MaxM, Clientes_Seg_R.R, Clientes_Seg_M.M, R*10+M AS RMTot

INTO TabelaRMporSegmento
FROM Clientes_Seg_R INNER JOIN Clientes_Seg_M ON
Clientes_Seg_R.NumClient=Clientes_Seg_M.NumClient

ORDER BY Clientes_Seg_M.M_Val;
  
```

5.5. Conclusão

O tratamento dos dados da maneira pretendida só é possível através das tabelas que se criam e das consultas respectivas. Apresentaram-se essas tabelas e o que elas contêm e mostrou-se como se processam as várias consultas.

No próximo capítulo vai ser mostrado a interface que foi implementado em Excel.

6. O interface

6.1. Introdução

Este capítulo vai fazer a apresentação da interface criado em Excel para interacção com o utilizador.

Pretendeu-se desenvolver ecrãs de aspecto agradável mas ricos em informação. Os dados mostrados não deviam ser estáticos mas sim possibilitar o máximo de flexibilidade para o utilizador poder interagir livremente, sem no entanto perder a eficácia desejada.

6.2. Os "dashboards"

A opção escolhida foi a preparação de “dashboards” – painéis de indicadores, que foram apetrechados com variados gráficos e pequenas indicações visuais que permitem mais facilmente e de forma rápida entender o sentido de determinada informação numérica.

Um bom exemplo da implementação de dashboards é o produto Pilot da SAP (<http://www.sap.com/portugal/press.epx?pressid=7430>):



Figura 6 - 1: produto Pilot da SAP

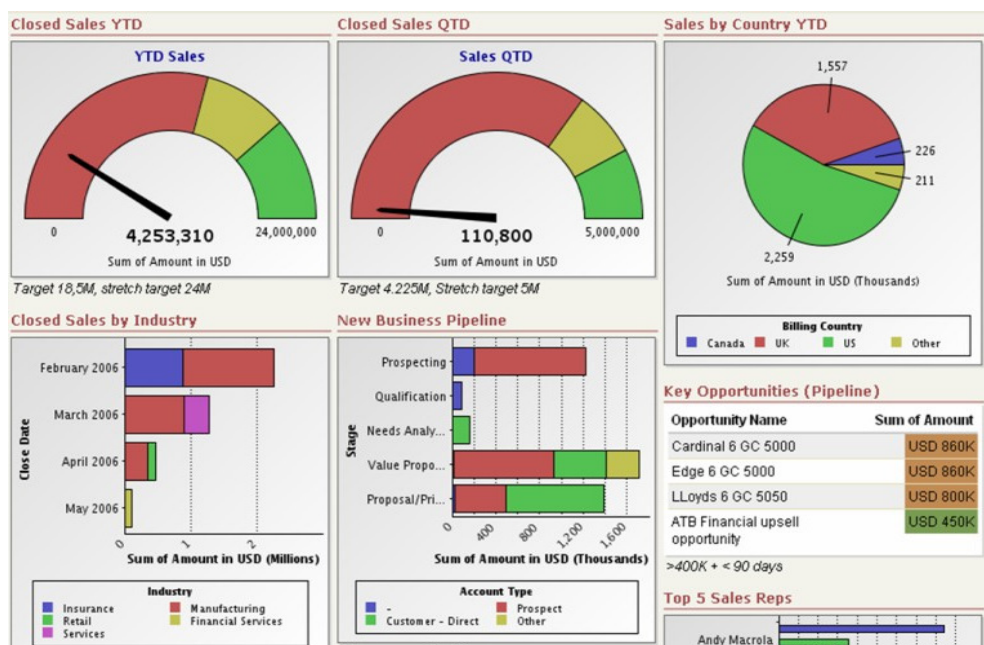


Figura 6 - 2: produto Pilot da SAP (dashboard)

6.3. Ecrã inicial

Quando se abre o livro “Projecto.xls” a aplicação passa para exibição em ecrã inteiro e automaticamente é mostrado o ecrã inicial com revelação de um splash screen identificando o que está a acontecer em segundo plano para manter o utilizador informado e interessado.

Trata-se de um userform ao qual, através do acesso à Windows API (Application Programming Interface), são retiradas a moldura e a barra de título. Ao ser fechado tem um efeito de desvanecimento por manipulação da sua propriedade transparência.

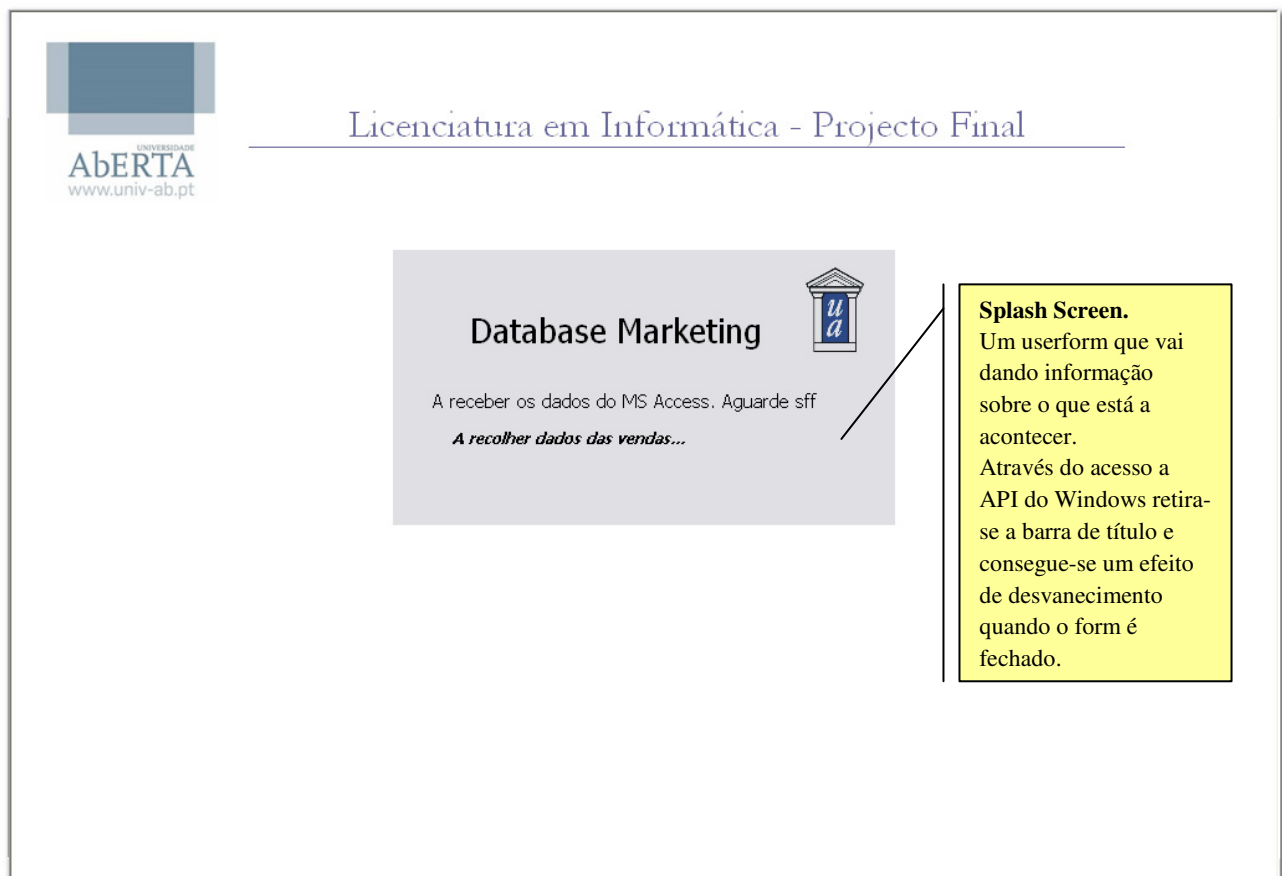


Figura 6 - 3: Dashboard – Ecrã Inicial

São executados os seguintes procedimentos para importação de dados do Access e ajuste ao livro do Excel:

Module 3

- inicio() – ajusta a folha de apresentação, corre splash screen (userform2) depois finaliza a apresentação;
- preparaDados() - procedimento chamado pelo userform2, tem como função preparar todos os dados enquanto actualiza informação ao utilizador no splashscreen

Module 1 – informação para painel de clientes

- VendasPorDia() - Lista das orders por dia;
- ListaMeses() - Lista dos meses;
- ContaClientes() – obtém nº clientes;
- ClientesAno() - Lista clientes que compram por ano;
- MediaVendasAno() - Lista média de vendas por ano;
- ResumoVendas() - Lista max dias e max valor de vendas;
- VendasPorMes() - Lista vendas por mês;
- MaxDataValor() - Lista valor total e dias decorridos por cliente

Module 2

dados para painel RM

- preparaRM() – executa procedimentos do Access para preparar RM
- buscaRM() - Lista dados RM conforme nº segmentos estipulado;
- clientesRM() – obtém clientes para “esquecer”;

dados para painel LTV

- iniciaLTV() - Lista dados LTV conforme nº segmentos estipulado em RM;

Depois de terminadas as tarefas de comunicação com o Access o utilizador pode escolher qual dos outros dashboards deseja ver através de botões de navegação que estão sempre disponíveis:

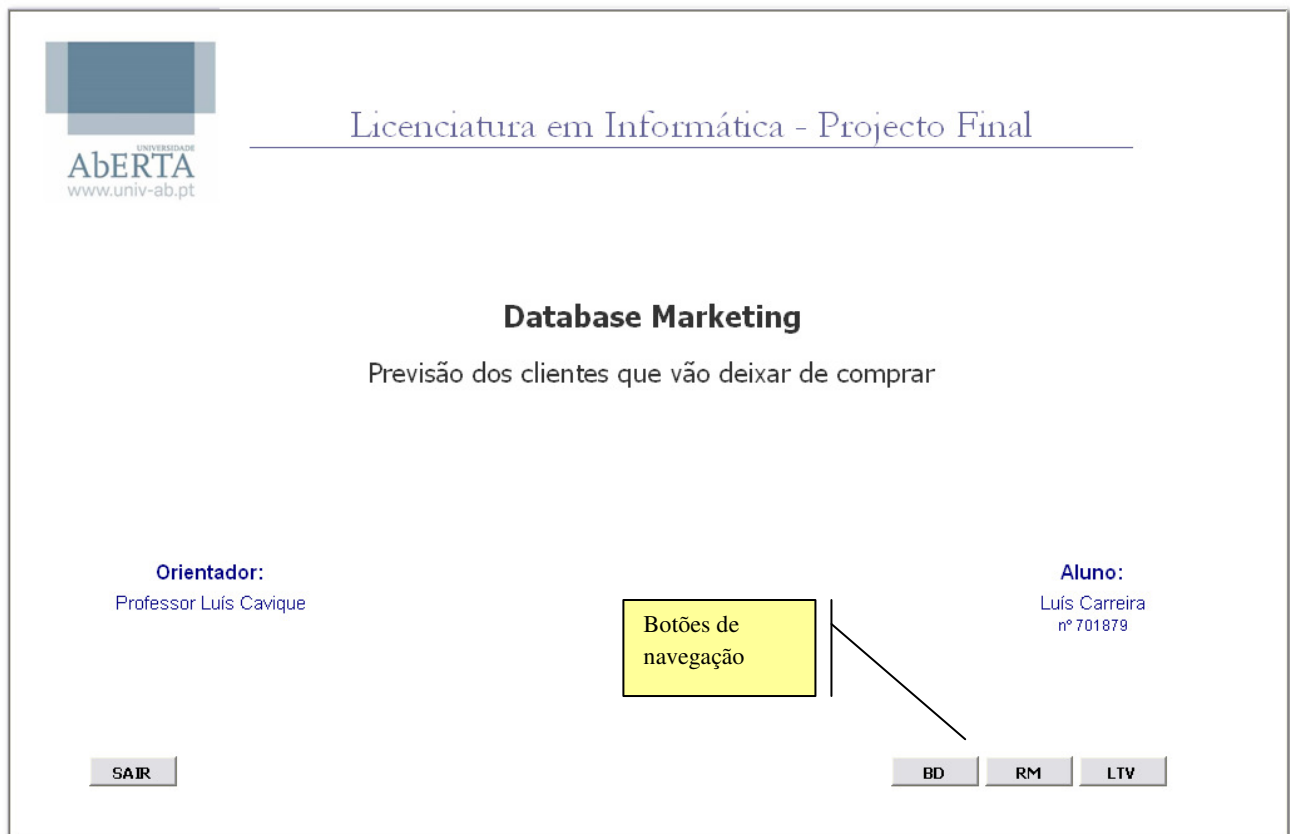


Figura 6 - 4: Dashboard – Ecrã Inicial, pronto para utilização

6.4. Painel Clientes (BD)

O dashboard de clientes exibe informação geral da base de dados:

- quantos clientes existem
- médias das compras
- nº de vendas registadas com indicação da menos recente e da de maior valor
- 2 gráficos de vendas mensais, um é interactivo. Um gráfico com a evolução das vendas e da média
- informação individualizada por cliente dos dias que decorreram desde a última encomenda e o valor comprado. Aqui são mostrados indicadores visuais para mais fácil leitura. A informação pode ser ordenada por cada uma destas colunas.

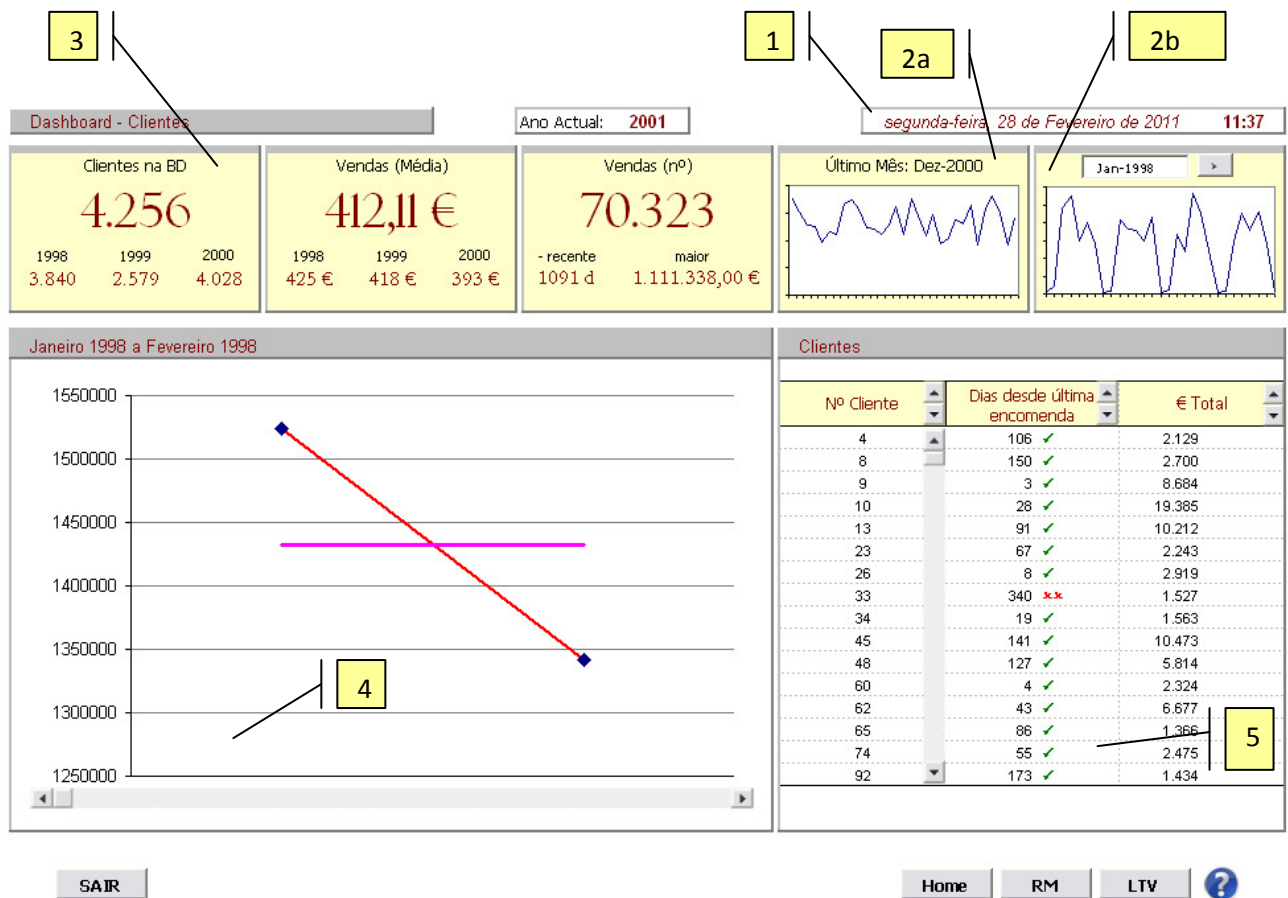


Figura 6 - 5: Dashboard – Painel Clientes

- 1 – Data e hora; o relógio é actualizado a cada 30 s (ajustável)
- 2a – Vendas do último mês na BD; 2b – vendas de qualquer mês, seleccionável
- 3 – Informações gerais
- 4 – Gráfico interactivo, mostra a evolução das vendas com indicação da média
- 5 – Informação individualizada por cliente, com indicações visuais. Colunas ordenáveis.

Nº Cliente	Dias desde última encomenda	€ Total
13164	342 ✗✗	1.111.338
8147	4 ✓	1.107.110
3526	180 ✗	561.165
13002	101 ✓	529.867
2151	24 ✓	351.502
14294	572 ✗✗✗✗	289.44

Figura 6 - 6: Pormenor do Painel Clientes

A interactividade dos gráficos consegue-se através da actualização de tabela dinâmica (2b) e de intervalos dinâmicos (4).

6.5. Painel RM

Neste painel é dada a possibilidade ao utilizador de seleccionar em quantas classes deseja segmentar os clientes.

Em tempo real a informação é actualizada e exibida uma matriz com a quantidade de clientes que se enquadram em cada classe. As classes dos clientes que vão deixar de comprar são indicadas a vermelho.

Há um gráfico circular que se ajusta automaticamente e exhibe o peso dos segmentos.

É mostrada a informação individual dos clientes “para esquecer”, com indicação das datas da primeira e da última compra, número de encomendas efectuadas, quantos dias passaram desde a última compra (R), valor médio das compras (M) e o segmento. A informação pode ser ordenada por cada uma destas colunas.

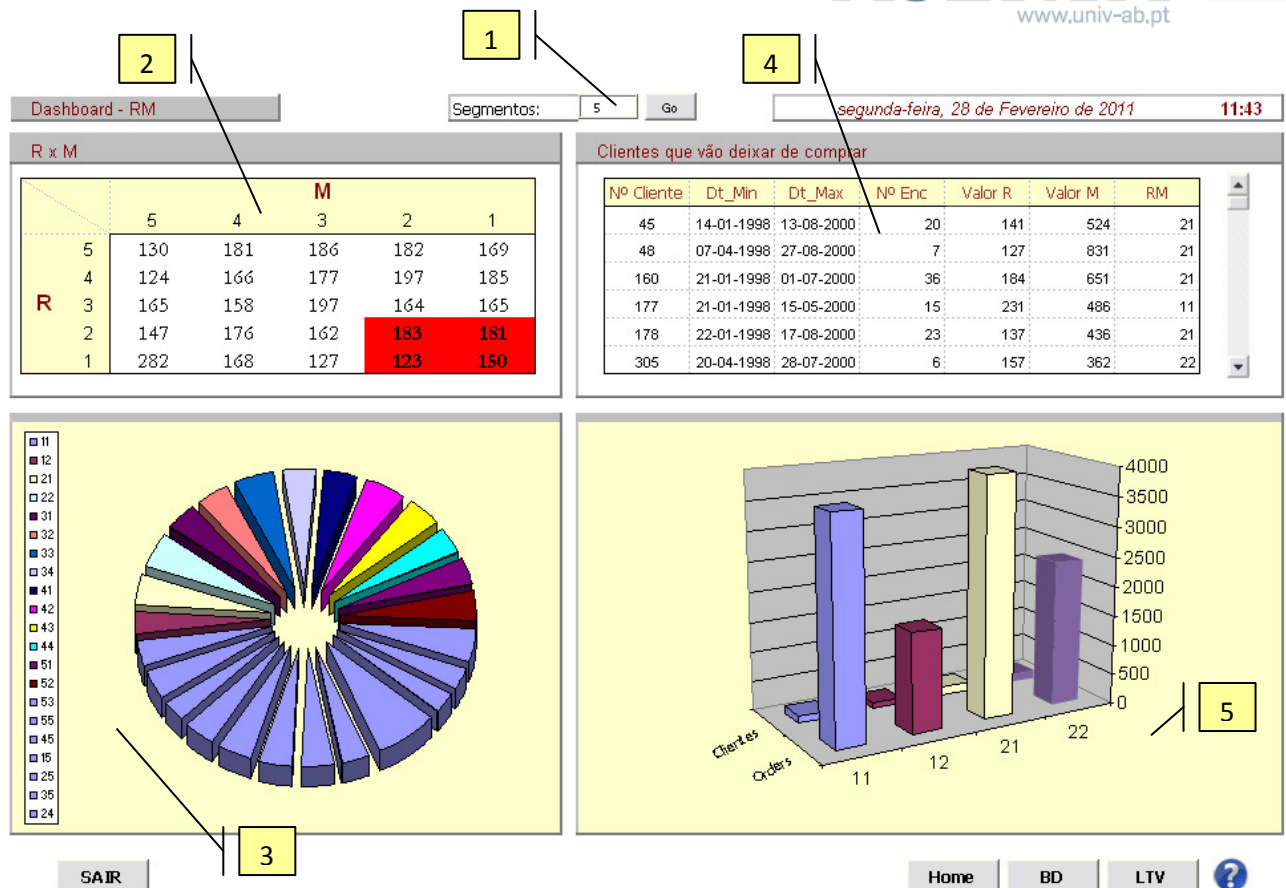


Figura 6 - 7: Dashboard RM

- 1 – Selecção dos segmentos (2 a 5)
- 2 – Matriz RxM com indicação a cores dos segmentos de clientes que vão deixar de comprar
- 3 – Peso de cada segmento
- 4 – Informação individualizada dos clientes que vão deixar de comprar. Colunas ordenáveis.
- 5 – Display gráfico dos clientes que vão deixar de comprar.

Principais procedimentos executados:

Module 2

- buscaRM() - Lista dados RM conforme nº segmentos estipulado;
- clientesRM() – obtém clientes para “esquecer”;

A interactividade dos gráficos consegue-se através da actualização de tabelas dinâmicas.

6.6. Painel LTV

É neste painel que é feita a exibição do LTV global a três anos, importado directamente do Access (ver figura 5-1). Os dados só são adaptados ao quadro com alguma formatação, mas nenhum cálculo suplementar é efectuado.

O LTV individual de cada cliente é mostrado no quadro ao lado, com indicação do seu valor e segmento e depois procede-se à comparação com os segmentos do RM. Há uma indicação visual (verde ou vermelho) caso o LTV individual seja igual ao M ou não.

Os segmentos são ajustados ao número estipulado pelo utilizador no dashboard do RM.

Um gráfico de linhas 3D exhibe essa comparação.

Para cálculo do LTV são necessários alguns dados que não se podem obter pela base de dados, a saber:

- Custo produção (%)
- Custo com os clientes (u.m.)
- Taxa de juro (%)
- Risco da actividade (1 a 2)
- Prazo de pagamento concedido aos clientes

Inicialmente calcula-se o LTV baseado em valores exemplo.

Neste painel está sempre visível um userform não modal, exibindo os dados por defeito que foram usados para o seu cálculo, e permitindo ao utilizador alterá-los para estudar o seu impacto no LTV.

Este form valida os dados – só são aceites valores numéricos e o “.” (ponto) é substituído por “,” (vírgula).

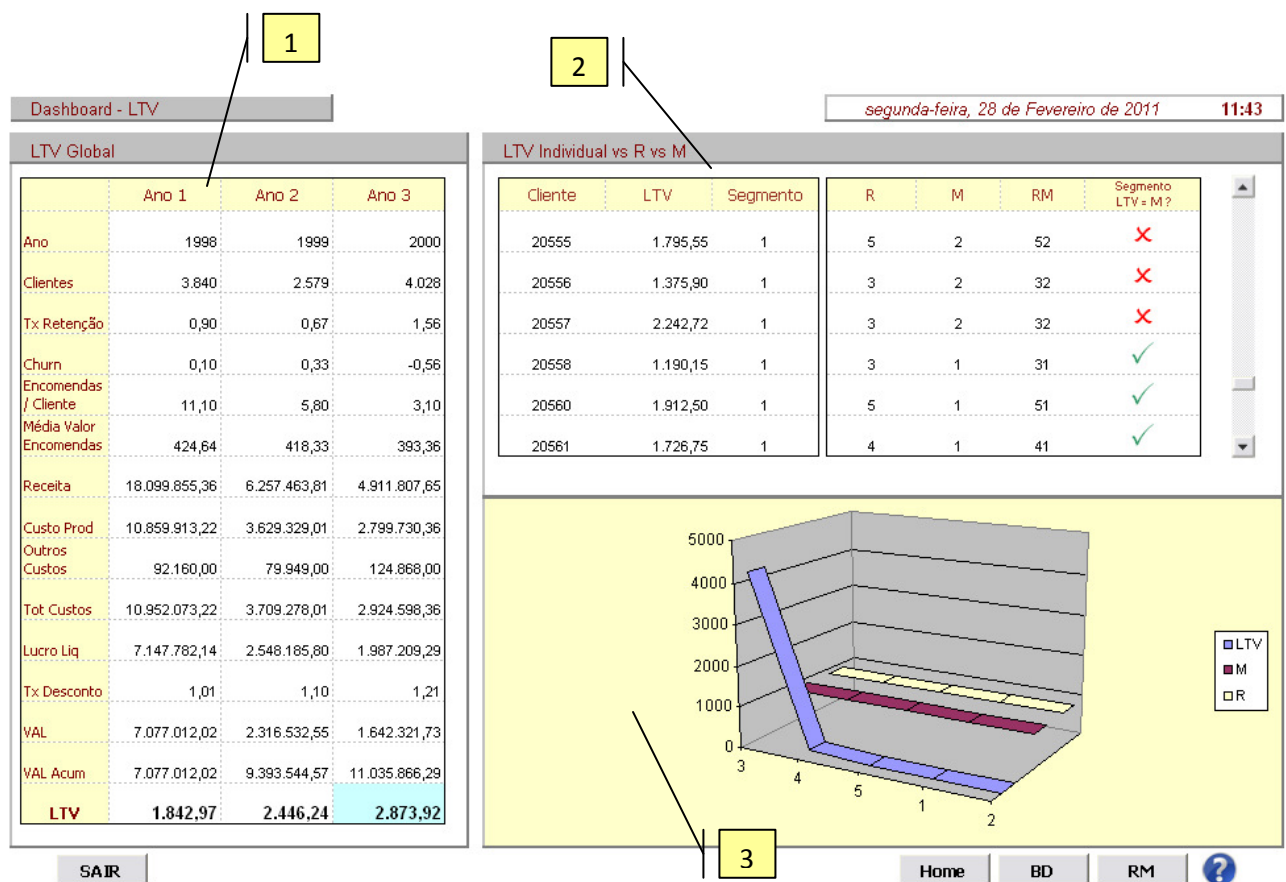


Figura 6 - 8: Dashboard LTV

Dados para cálculo do LTV

	Ano 1	Ano 2	Ano 3
Custo Prod (%)	60	58	57
Custo Cliente	24	31	31
Tx Juro %	6	6	6
Risco (1 a 2)	1,6	1,6	1,6
Prazo Pag (d)	30	30	30

Go

Figura 6 - 9: Userform com os dados para cálculo do LTV

1 – LTV

2 – LTV individualizado e comparação com o RM; indicação visual se LTV = M

3 – Display gráfico da comparação com o RM

4 – Form para alteração dos dados para cálculo do LTV; form não modal para poder estar sempre visível; só são aceites valores numéricos e o “.” é substituído por “,” para mais fácil utilização

Principais procedimentos executados:

- CommandButton1_Click() do Userform1 - Lista dados LTV conforme nº segmentos estipulado em RM e dados de LTV individual, mas só funciona se não houver campos vazios.

6.7. Conclusão

O interface implementado assume a forma de dashboards (painéis de indicadores). Existem um ecrã inicial e três painéis – Clientes (BD), RM e LTV com os quais se pode interagir e o utilizador movimenta-se entre eles com botões de navegação.

O próximo capítulo descreve em pormenor o código VBA programado e como os procedimentos se interligam

7. VBA

7.1. Introdução

Nas próximas secções é descrito o código VBA programado e como todos os processos são interligados de forma a permitir o devido funcionamento do aplicativo.

7.2. Código VBA

A produção do código VBA teve duas grandes finalidades: proporcionar uma experiência mais rica ao utilizador e tornar possível a comunicação entre o Excel e o Access.

Na primeira enquadra-se o código executado para:

- Preparar o ecrã inicial e executar o splash screen;
- Apresentar os manuais de utilização;
- Actualizar o relógio;
- Permitir a selecção do gráfico mensal;
- Ordenar as colunas dos quadros;
- Validar os valores introduzidos no userform do LTV;

Relativamente à segunda finalidade havia a escolha entre o uso de ligação via DAO (Data Access Objects) ou ADO (ActiveX Data Objects). Basicamente, DAO é uma interface para acesso a bases de dados. ADO é mais recente e é constituído por um conjunto de objectos para acesso a dados.

Descobriu-se que o recurso a DAO é mais funcional quando se trata de ter acesso a consultas gravadas no Access e obter os dados devolvidos por estas.

ADO pelo contrário, revelou-se ideal para obter dados do Access por envio da própria instrução de SQL.

Nas secções 7.4 e 7.5 apresenta-se um exemplo de cada uma destas duas ligações.

7.3. VBA para utilizador ver

Apresentam-se dois exemplos de código desenvolvido com o sentido de cativar o utilizador.

O primeiro a ser destacado é a criação do splash screen. Quando o userform é activado, retira-se-lhe a moldura e a barra de título, de acordo com a versão do Excel que está a ser executada. De seguida é preparado o processo para mudar a transparência e o form fica em execução.

```
Private Sub UserForm_Initialize()
'retira a moldura e a barra de titulo do form
If Val(Application.Version) >= 9 Then
    wHandle = FindWindow("ThunderDFrame", Me.Caption)
Else
    wHandle = FindWindow("ThunderXFrame", Me.Caption)
End If
If wHandle = 0 Then Exit Sub
lStyle = GetWindowLong(wHandle, GWL_STYLE)
Me.Caption = ""
lStyle = lStyle And Not WS_SYSMENU
lStyle = lStyle And Not WS_MAXIMIZEBOX
lStyle = lStyle And Not WS_MINIMIZEBOX
lStyle = lStyle And Not WS_CAPTION
SetWindowLong wHandle, -20, frm
SetWindowLong wHandle, GWL_STYLE, lStyle
DrawMenuBar wHandle
'torna o form visivel
Transparency = 120
Call SemiTransparent(100)
DoEvents
End Sub
```

A visibilidade é mantida enquanto a variável *Running* é *true*. Portanto, ao fechar o form basta mudar a variável para *false* e o form é gradualmente tornado transparente.

```
Private Sub Transparency()
'faz o fade out
Dim MyTimer As Double
DoEvents
MyTimer = Timer
Do
    Do
        Loop While Timer - MyTimer < 0.07
        MyTimer = Timer
        Transparency = Transparency - 3
    If Transparency < 0 Then
        Unload Me
    Else
        Call SemiTransparent(Application.WorksheetFunction.Min(Transparency, 100))
    End If
Loop
```

```
End If
DoEvents
Loop While Running
End Sub
```

```
Private Sub SemiTransparent(ByVal intLevel As Integer)
'ajusta a transparência
Dim lngWinIdx As Long
    hwnd = GetActiveWindow
    lngWinIdx = GetWindowLong(hwnd, GWL_EXSTYLE)
    SetWindowLong hwnd, GWL_EXSTYLE, lngWinIdx Or WS_EX_LAYERED
    SetLayeredWindowAttributes hwnd, 0, (255 * intLevel) / 100, LWA_ALPHA
End Sub
```

Outro exemplo que se destaca é um dos procedimentos com a função de validar os valores que o utilizador escreve no userform1 (dados para cálculo do LTV).

O procedimento verifica se os dígitos inscritos nas caixas de texto são numéricos, exibindo uma mensagem de erro e mudando a cor da caixa de texto para vermelho em caso de engano.

Para ser mais fácil de utilizar este form, o procedimento troca a inserção de um ponto “.” Por uma vírgula “,”.

```
Private Sub TextBox1_Change()
Dim valorcx As String
Dim conta As Integer
Dim caractEmUso As String
Dim verificacao As Boolean
    valorcx = TextBox1.Text
    conta = TextBox1.TextLength
    If valorcx <> "" Then
        caractEmUso = Mid(valorcx, conta, 1)
        verificacao = IsNumeric(valorcx)
        If verificacao = False Then
            TextBox1.BackColor = RGB(255, 0, 0)
            hls_MsgBalloonvalor
            valorcx = ""
            TextBox1.Text = ""
            TextBox1.BackColor = &H80000005
        End If
        If caractEmUso = "." Then
            Application.SendKeys ("{bs}")
            Application.SendKeys (",")
        End If
    End If
```

```
End If
End Sub
```

7.4. DAO

O primeiro procedimento a usar DAO é o *VendasPorDia()* cuja finalidade é importar uma lista do valor das vendas por cada dia.

Começa por identificar a base de dados e a consulta (query) que se pretende executar. De seguida abre o conjunto dos registos (recordset) devolvidos pela consulta.

Os registos são então copiados para as células destino. Para se obterem também os cabeçalhos é necessário ler o nome dos campos e copiá-los para o destino.

Depois é preciso fechar as ligações novamente.

```
Sub VendasPorDia()
'Lista das encomendas por dia, vem da tabela orders, vai para DadosBD.A1
'Declara variaveis
Dim MyDatabase As DAO.Database
Dim MyQueryDef As DAO.QueryDef
Dim MyRecordset As DAO.Recordset
Dim i As Integer
'Identifica a BD e a query
Set MyDatabase = DBEngine.OpenDatabase(BDaAbrir)
Set MyQueryDef = MyDatabase.QueryDefs("OrdersTotalDia")
'Abre query
Set MyRecordset = MyQueryDef.OpenRecordset
'Limpa conteúdo destino
Sheets("DadosBD").Select
ActiveSheet.Range("A:C").ClearContents
'Copia registos para Excel
ActiveSheet.Range("A2").CopyFromRecordset MyRecordset
'Acrescenta cabeçalho
For i = 1 To MyRecordset.Fields.Count
    ActiveSheet.Cells(1, i).Value = MyRecordset.Fields(i - 1).Name
Next i
With ActiveSheet.Range("A1:C1")
    .EntireColumn.AutoFit
End With
'fecha ligação
MyRecordset.Close
MyQueryDef.Close
'limpeza
```

```
Set MyDatabase = Nothing
Set MyQueryDef = Nothing
Set MyRecordset = Nothing
End Sub
```

7.5. ADO

Um dos procedimentos mais interessantes a utilizar ADO é o *clientesRM()*, que vai obter os clientes que vão deixar de comprar, para exibir no quadro respectivo.

Começa por abrir uma ligação à base de dados e depois define a própria instrução SQL. Desta maneira pode-se ajustar o SQL às necessidades do momento, o que se faz neste caso com o número de segmentos pretendidos pelo utilizador. De seguida abre o recordset e processa a consulta, obtém os dados e copia-os para o destino. Aqui não há a possibilidade de obter os nomes dos cabeçalhos pelo que têm de ser indicados no próprio procedimento. A ligação é depois fechada.

```
Sub clientesRM()
'lista os clientes com a segmentação
'RM individual já atribuída
'variaveis
Dim MyConnect As String
Dim MyRecordset As ADODB.Recordset
Dim MySQL As String
Dim param As Integer
'Identifica a BD
MyConnect = "Provider=Microsoft.ACE.OLEDB.12.0;" & _
    "Data Source=" & BDaAbrir
'SQL
MySQL = "SELECT NumClient, mindate," & _
    " maxdate, numorders," & _
    " R_Val, M_Val," & _
    " RMTot" & _
    " FROM TabelaRMporSegmento "
'ajusta SQL ao nº segmentos
param = Sheets("DadosRM").Range("C2").Value
Select Case param
Case Is > 3
    MySQL = MySQL & "WHERE (RMTot=22) " & _
        "OR (RMTot=21) " & _
        "OR (RMTot=12) " & _
        "OR (RMTot=11)"
Case Else
```



```

        MySQL = MySQL & "WHERE (RMTot=11)"
    End Select
    MySQL = MySQL & " ORDER BY NumClient;"
'Abre registos
    Set MyRecordset = New ADODB.Recordset
    MyRecordset.Open MySQL, MyConnect, adOpenStatic, adLockReadOnly
'copia para excel
    Sheets("DadosRM").Select
    ActiveSheet.Range("Y:AE").ClearContents
    ActiveSheet.Range("Y2").CopyFromRecordset MyRecordset
'Step 6: Add column labels
    With ActiveSheet.Range("Y1:AE1")
        .Value = Array("NumClient", "mindate", "maxdate", _
            "numorders", "R_Val", "M_Val", "RMTot")
        .EntireColumn.AutoFit
    End With
'actualiza pivot table
    ActiveSheet.PivotTables("TabDinCIFora").PivotCache.Refresh
'fecha ligação e limpa
    MyRecordset.Close
    Set MyRecordset = Nothing
End Sub

```

7.6. Diagrama de sequência

Apresenta-se de seguida o diagrama de sequência dos procedimentos envolvidos. Existem mais procedimentos para além destes, cuja função é executar tarefas menores como ordenar colunas, navegar entre os painéis ou despoletar a execução de outros procedimentos. Estes não são retratados neste diagrama por não terem importância de maior.

Indicam-se os parâmetros que são passados entre Excel e Access, que podem ser SQL ou outros.

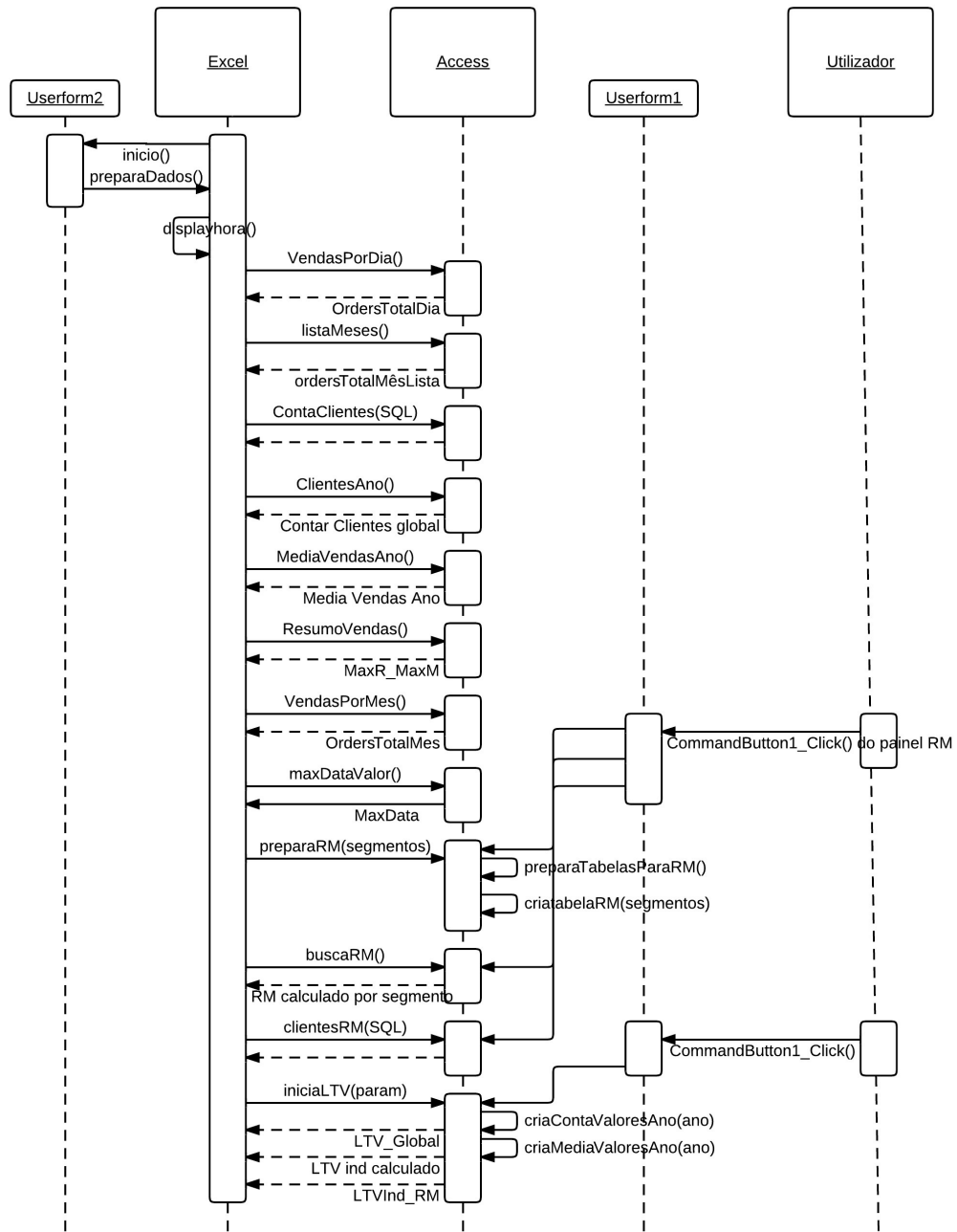


Figura 7 - 1: Diagrama de sequência

7.7. Conclusão

Enumeraram-se as finalidades do código VBA da aplicação, tendo sido destacados alguns exemplos mais importantes.

Mostrou-se o que são e como se utilizou as ligações DAO e ADO para estabelecer as comunicações entre Excel e Access, e apresentou-se um diagrama de sequência.

O capítulo 8 faz a apresentação dos manuais de utilizador implementados.

8. Manuais de utilização

8.1. Introdução

Pretendeu-se fugir aos manuais “tradicionais” em papel ou pdf, onde somente se procede à leitura e visualização de algumas figuras. Em vez disso utilizou-se uma ferramenta para criar tutoriais em filme no formato flash, o que permite que o utilizador “veja” a explicação a acontecer ao mesmo tempo em que são apresentados os esclarecimentos necessários.

8.2. Produzir os tutoriais

Optou-se pelo programa *Wink*, freeware disponível em <http://www.debugmode.com/wink/>.

Este programa permite fazer a captura de tela em frames, trabalhá-las, enriquecê-las com caixas de diálogo em vários formatos, botões de navegação e várias outras formas para além de áudio. Toda a acção que decorreu quando foi feita a captura de tela é reproduzida, incluindo os movimentos do rato.

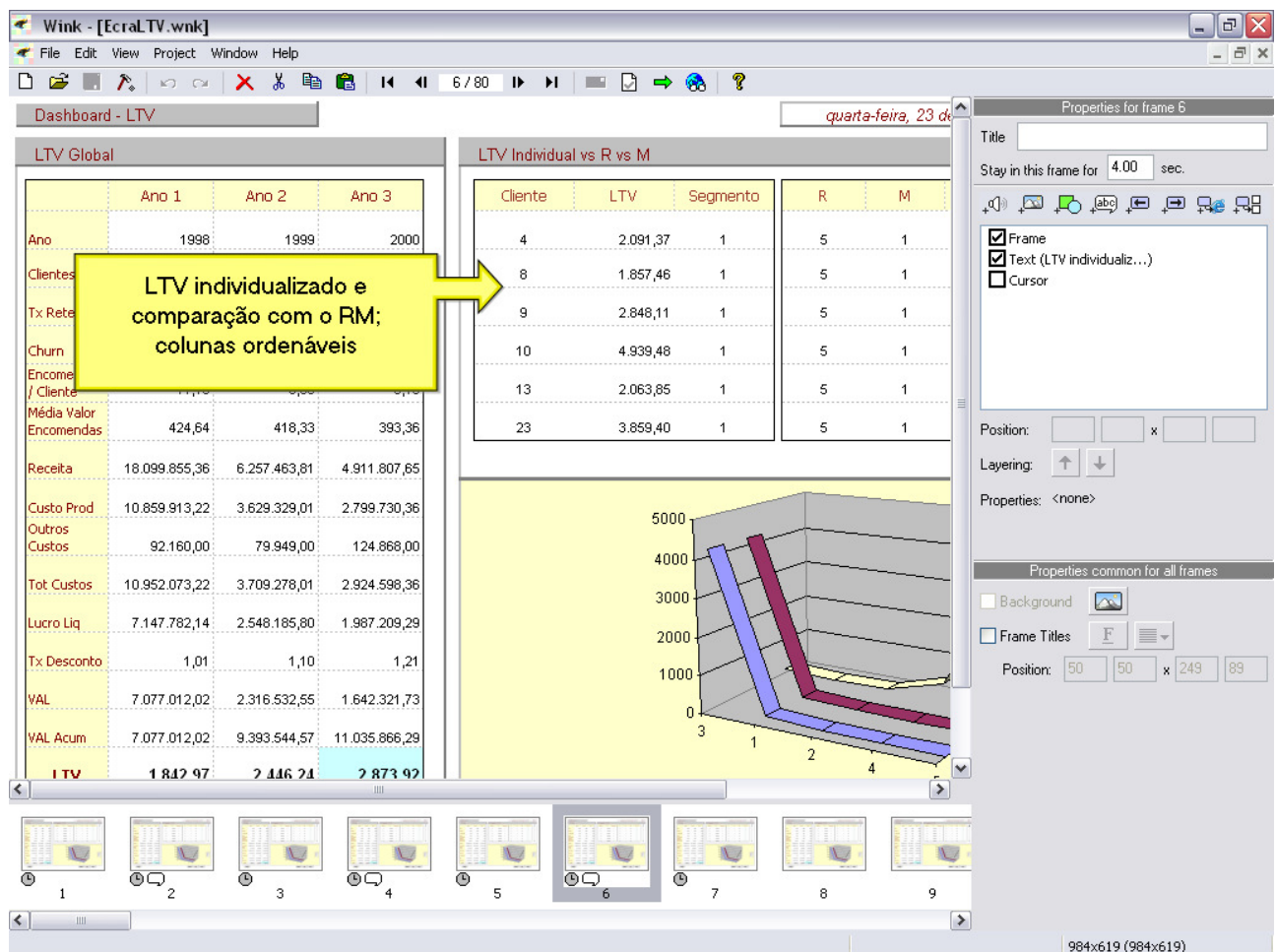



Figura 8 - 1: Programa Wink

Depois o programa converte o trabalho em filme no formato flash e até cria uma página html para a sua exibição.

Procedeu-se assim à feitura de quatro tutoriais, um para cada dashboard.

Os tutoriais são chamados pelos botões de ajuda em cada ecrã “”. Ao clicar, é executado o procedimento *tutorial()*, que vai determinar em que dashboard se encontra o utilizador e a partir daí chamar a página html respectiva, o que vai iniciar o browser pré-definido no sistema e exibir o filme.

8.3. Tutorial do ecrã inicial

Os tutoriais são bastante extensos, o do ecrã de clientes tem 114 frames, o que torna a sua reprodução neste espaço inviável. Demonstra-se unicamente o do ecrã inicial, feito com 11 frames e perfeito para exemplificar o trabalho executado:



Licenciatura em Informática - Projecto Final

Database Marketing

Previsão dos clientes que vão deixar de comprar

Orientador:

Professor Luís Cavique

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

Ecrã inicial

Este aplicativo pretende determinar os clientes que vão deixar de comprar, com base nos dados existentes em base de dados do MS Access.

Pre

mprar

Orientador:

Professor Luís Cavique

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

Apresenta-se o critério de segmentação de clientes, modelo RM, com indicação do enquadramento de cada cliente e identificação dos que já não têm interesse para a empresa.

Pre

mprar

Orientador:

Professor Luís Cavique

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

São calculadas outras metricas, tais como o churn, a taxa de retenção, o LTV (Lifetime Value) global a três anos, e LTV individual.

Pre mprar

Orientador:

Professor Luís Cavique

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

Pode ainda ver-se uma comparação entre RM/LTV individual, depois de aplicado o mesmo critério de segmentação a este último.

Pre mprar

Orientador:

Professor Luís Cavique

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

Database Marketing

Previsão dos clientes que vão deixar de comprar

Orientador:

Professor Luís Cavique

Pode seleccionar o
painel que pretende
visitar através dos botões
de navegação.

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

Database Marketing

Previsão dos clientes que vão deixar de comprar

Orientador:

Professor Luís Cavique

Pode seleccionar o
painel que pretende
visitar através dos botões
de navegação.

Aluno:

Luís Carreira
nº 701879

SAIR



BD

RM

LTV

Licenciatura em Informática - Projecto Final

Database Marketing

Previsão dos clientes que vão deixar de comprar

Orientador:

Professor Luís Cavique

Painel BD:

- contém informação geral dos
clientes que constam na BD

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

Database Marketing

Previsão dos clientes que vão deixar de comprar

Orientador:

Professor Luís Cavique

Painel RM:

- onde é mostrado como os
segmentos (de 2 a 5) afectam a
selecção dos clientes a "esquecer"

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

Database Marketing

Previsão dos clien

Painel LTV:

- exibe o LTV de 3 anos e dá a possibilidade de alterar os parâmetros para ver o seu impacto nos valores finais
- exibe informação do LTV individual e a sua comparação (depois de segmentado) com o RM

Orientador:

Professor Luís Cavique

SAIR

BD

RM

LTV

Licenciatura em Informática - Projecto Final

Database Marketing

Previsão dos clientes que vão deixar de comprar

Ver outra vez



Orientador:

Professor Luís Cavique

Aluno:

Luís Carreira
nº 701879

SAIR

BD

RM

LTV

Figura 8 - 2 a 12: Tutorial do ecrã inicial

8.4. Conclusão

Quanto aos manuais de utilização optou-se por produzir filmes em formato flash utilizando o programa Wink. Explica-se o processo de produção e representa-se o tutorial produzido para o ecrã inicial.

O capítulo 9 apresenta as conclusões finais.

9. Conclusão

A tarefa de desenvolvimento deste projecto acabou por ser muito mais trabalhosa e morosa do que inicialmente se pensava. A auto-exigência de que o tratamento e a preparação dos dados seriam efectuados totalmente em SQL obrigou à criação de uma quantidade razoável de consultas, por vezes bastante complexas. O desenvolvimento de uma interface atraente, a sua automatização e a conjugação de todo o sistema necessitou de vastas pesquisas e da leitura de alguns livros sobre a matéria. A fase de testes foi também algo lenta, fruto da quantidade de código VBA envolvido.

O resultado final é uma aplicação muito funcional, agradável de usar e de fácil leitura. O usuário não necessita de ter conhecimentos de estatística nem de Microsoft Office – basta iniciar o aplicativo, fazer selecções simples e analisar os resultados exibidos.

Globalmente, conseguiu-se aproveitar o que o Excel e o Access têm de melhor, comprovou-se que estes oferecem realmente enormes possibilidades e que foram a escolha acertada para este trabalho.

Quanto ao RM confirmou-se que a sua aplicação é de grande simplicidade e que é verdadeiramente eficaz. O LTV, o "churn" e a taxa de retenção são mostrados com clareza e o utilizador pode livremente fazer as suas projecções. A comparação entre o LTV individual e o RM é útil para assistir à tomada de decisões. Assim a ferramenta desenvolvida impõe-se como uma excelente ajuda à gestão dos clientes da empresa.

Os objectivos traçados foram plenamente atingidos, se não mesmo ultrapassados, mas haverá sempre lugar a melhorias pois uma aplicação destas nunca está realmente acabada. Os aperfeiçoamentos mais imediatos seriam talvez a possibilidade de impressão dos dados obtidos e dos respectivos relatórios, a possibilidade de personalização dos gráficos e a automatização da importação das tabelas principais para permitir uma flexibilidade ainda maior.

10. Bibliografia

Cavique L. (2003), Micro-Segmentação de Clientes com Base em Dados de Consumo: Modelo RM-Similis, Revista Portuguesa e Brasileira de Gestão, volume 2, nº3, pp. 72-77.

Arthur Middleton Hughes, <http://www.dbmarketing.com>

Kumar V. (2008), Managing Customers for Profit: Strategies to Increase Profits and Build Loyalty, Wharton School Publishing

Vercellis C. (2009), Business Intelligence - Data Mining and Optimization for Decision Making, Wiley

Alexander M. (2008), Excel 2007 Dashboards & Reports For Dummies, Wiley

Dagleish D. (2006), Excel Pivot Tables Recipe Book - A Problem-Solution Approach, Apress

Alexander M. (2010), The Excel Analyst's Guide to Access, Wiley

Linoff G. (2008), Data Analysis Using SQL and Excel, Wiley

Allison C., Berkowitz N. (2005), SQL for Microsoft Access, Wordware Publishing

Steve Saunders, Jeff Webb (2006), Programming Excel with VBA and .NET, O'Reilly