

Laboratórios de Informática III

Sistema de Gestão de Vendas

Grupo 34



Ana Afonso A85762



Márcia Teixeira A80943



Pedro Silva A82522



Índice

Introdução	3
Include	4
Módulos.....	4
Source	4
Clientes.....	5
Produtos.....	5
SGV	6
View e Controller	7
Estruturas de Dados	7
Memória.....	7
Tempos de Execução.....	8
Conclusão	9



Introdução

O presente projeto enquadra-se na unidade curricular de Laboratórios de Informática III, na qual foi proposta a elaboração de um Sistema de Gestão de Vendas.

Recorrendo à aplicação de uma arquitetura baseada em Model, View e Controller, bem como à utilização de determinadas estruturas de dados, o sistema a desenvolver apresenta como principal objetivo o armazenamento de informações de vendas relacionando produtos, clientes e vendas.

Em foco, encontra-se ainda o encapsulamento de dados, que é de considerável importância, uma vez que visa impedir possíveis alterações efetuadas por agentes externos.

A fim de garantir o bom funcionamento do sistema torna-se necessário responder de forma ponderada e consciente às *queries* enunciadas, assegurando a eficiência do mesmo.

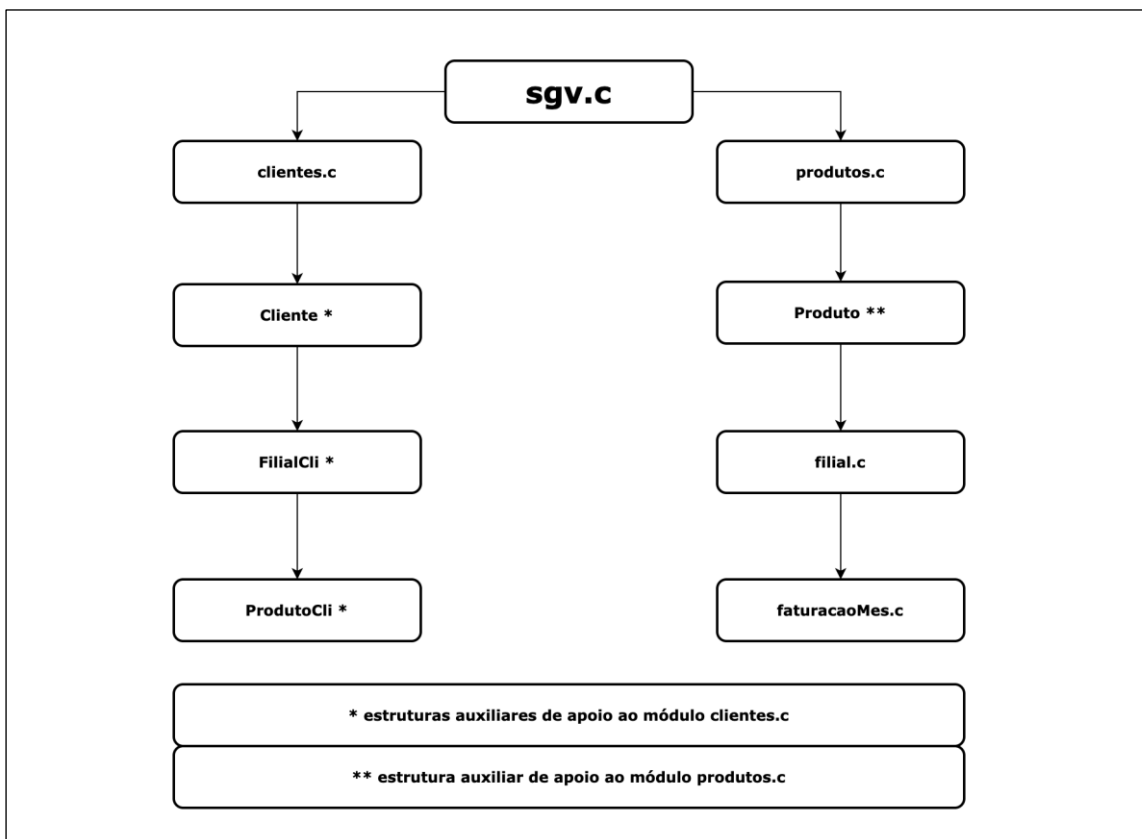


Include

Esta pasta contém todos os *header files* do projeto.

Cada *header file* inclui as declarações e respetivas descrições sumárias das funções implementadas nos *source files*.

Módulos



Source

Esta pasta inclui todos os módulos (*source files*).



Cientes

Este módulo visa responder à necessidade da existência de um espaço que contenha todos os clientes de um sistema. Para esse efeito, foi criada uma *hashtable* que permite, de forma eficiente, o armazenamento e eventual pesquisa de cada cliente.

Cliente

Esta estrutura auxiliar foi criada devido à inevitabilidade da existência de uma estrutura capaz de incluir as informações de cada cliente. Como mencionado anteriormente, cada cliente encontra-se armazenado na *hashtable* existente no módulo Cientes, uma vez que é necessário existir um identificador de cliente (*key*) e um array de filiais que contém informação relativa às compras de cada cliente (*value*).

FilialCli

Esta estrutura auxiliar foi gerada com vista ao arquivo de informação relativa a cada filial onde os clientes efetuam compras. A fim de facilitar o processo de resposta relativo a determinadas *queries* foi incluída a quantidade (por unidade) de compras efetuadas por cliente numa dada filial, bem como um *array* de *hashtables* em que em cada posição estão os produtos comprados em cada mês.

ProdutoCli

Esta estrutura auxiliar foi concebida para guardar conhecimento útil subjacente a certas *queries*, nomeadamente o identificador de produto, a quantidade (em unidades) comprada daquele produto e a faturação do mesmo. Tudo isto apenas referente a compras de um produto num dado mês.

Produtos

Este módulo foi elaborado com o intuito de ser viável o acesso e armazenamento, de forma eficiente, de todos os produtos que coexistem no sistema. A ideia concebida passa também pela utilização das vantagens inerentes às *hashtables*, assim foi criado um *array* que em cada posição tem guardada uma *hashtable* de produtos de uma determinada letra, de modo a que esteja presente uma ordem lexicográfica.



Produto

Este módulo foi gerado para armazenar a informação de cada produto, designadamente o seu identificador e as filiais nas quais foi vendido, estas últimas organizadas num *array* de filiais.

Filial

Este módulo foi desenvolvido para guardar a informação relativa à faturação mensal existente em cada filial. Para efeitos de eficiência na pesquisa e resposta a determinadas *queries*, a informação encontra-se armazenada num *array* em que cada posição possui a faturação mensal.

FaturacaoMes

Este módulo foi estabelecido com a finalidade de abarcar um total de faturação promocional e/ou normal, bem como um total de vendas promocional e/ou normal. Desta forma, tornou-se necessário armazenar as vendas efetuadas no mês em causa, sendo que para tal, foram utilizados dois *pointer arrays* capazes de distinguir vendas promocionais de vendas normais.

Vendas

Este módulo foi elaborado para guardar a informação relativa a cada venda, nomeadamente o identificador de produto comprado, o preço unitário, a quantidade (por unidade) comprada, o tipo de compra, o identificador de cliente que procedeu à compra, o mês e a filial nos quais a compra foi realizada.

SGV

Este módulo agrega todos os módulos acima descritos, apresentando uma estrutura que inclui o Catálogo de Produtos e o Catálogo de Clientes.

Aqui é também estabelecida a ligação entre todos os módulos internos e o exterior, sendo que é a este que a main (controlador e *view*) faz pedidos e tem métodos capazes de responder às *queries*.



View e Controller

Com o objetivo último de facilitar a pesquisa de *queries* foi criado um módulo que unifica o controlador e a *view*, este recebe os inputs do utilizador e chama os métodos existentes no SGV para responder aos pedidos do mesmo, fazendo uma impressão genérica de estruturas.

Estruturas de Dados

Numa primeira fase, utilizaram-se *arrays* para armazenar os dados, o que se traduziu em tempos de leitura e validação de ficheiros de aproximadamente seis minutos.

Numa segunda fase, utilizaram-se *GHashTables*, *GLists* e *GPtrArrays* (decisão tomada após diversas abordagens deliberadas à problemática) o que se traduziu numa diminuição significativa do tempo, para meros segundos. Dado que esta resultou num melhor tempo de execução, considerou-se esta solução como a ótima.

Memória

A ferramenta *Valgrind*, permitiu verificar que o programa corre sem *memory leaks*, excetuando alguns provocados pela biblioteca *glib*.

```
==324837==
==324837== HEAP SUMMARY:
==324837==    in use at exit: 18,612 bytes in 6 blocks
==324837== total heap usage: 27,306,787 allocs, 27,306,781 frees, 1,163,446,428 bytes allocated
==324837==
==324837== LEAK SUMMARY:
==324837==    definitely lost: 0 bytes in 0 blocks
==324837==    indirectly lost: 0 bytes in 0 blocks
==324837==    possibly lost: 0 bytes in 0 blocks
==324837==    still reachable: 18,612 bytes in 6 blocks
==324837==    suppressed: 0 bytes in 0 blocks
==324837== Reachable blocks (those to which a pointer was found) are not shown.
==324837== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==324837==
==324837== For lists of detected and suppressed errors, rerun with: -s
==324837== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```



Tempos de Execução

	Vendas 1 milhão	Vendas 3 milhões	Vendas 5 milhões
Query 1 Time	3.691577	8.529476	13.634581
Query 2 Time	0.001517	0.001640	0.001645
Query 3 Time	0.000013	0.000012	0.000016
Query 4 Time	0.155543	0.093921	0.073680
Query 5 Time	0.004853	0.004902	0.005165
Query 6 Time	0.130629	0.095232	0.073805
Query 7 Time	0.000020	0.000028	0.000096
Query 8 Time	0.157763	0.158705	0.163804
Query 9 Time	0.000018	0.000021	0.000028
Query 10 Time	0.000047	0.000055	0.000042
Query 11 Time	0.678762	0.677011	0.684826
Query 12 Time	0.000164	0.000768	0.001936

Os tempos de execução apresentados encontram-se em segundos



Conclusão

Em virtude do que foi mencionado, o grupo considera que o sistema gerado é capaz de responder, de uma forma eficiente, a uma panóplia de *queries* que visam o bom funcionamento deste.

Durante a realização do projeto, a principal inquietação esteve presente na busca incessante pela eficiência, servindo esta de fio condutor para diversas decisões.

Ao longo da implementação de todos os módulos foram sendo respeitados os requisitos propostos, apresentando-se como maior dificuldade a organização e encapsulamento de informação que melhor respondesse a todas as *queries* da forma mais eficiente possível. Para contornar esta adversidade optou-se por elaborar um planeamento que fosse útil para um maior número de *queries*.

Como aspetos a melhorar, seria principal foco diminuir os tempos de execução, bem como realizar mudanças na estrutura, de modo a torná-la mais flexível ao acréscimo de novas *queries* ou possíveis alterações ao sistema noutras vertentes.

Este projeto revelou-se bastante enriquecedor para todos os elementos do grupo, uma vez que incentivou não só a aprendizagem e manuseamento relativo a estruturas de dados, como também mitigou a capacidade de autonomia nas decisões e divisões equitativas do trabalho.

Em suma, compreende-se a necessidade da existência de Sistemas de Gestão de Vendas nomeadamente para empresas, sendo uma forma extremamente rápida de pesquisar por todo e qualquer género de informação que aí se encontre.