Ciências da Computação

Programação Orientada aos Objetos

JavaFactura - Relatório

GRUPO 13

A82726 - Matias Abreu Capitão

A79346 - José António da Silva Oliveira

A77457 - Rafael Antunes Simões

Índice

1-Introdução

Nos dias de hoje, é possível que um contribuinte, tanto individual como empresarial, possa iniciar sessão numa plataforma e observar os movimentos efetuados relativos ao seu número fiscal. Os contribuintes individuais podem ver as faturas emitidas em seu nome, observar o valor total deduzido e também, classificar a atividade de uma determinada fatura por classificar.

Neste projeto procuramos desenvolver em Java um programa que fosse o mais próximo possível á realidade.

Neste caso específico o programa que nos foi proposto deveria permitir efetuar registos de contas e, posteriormente, Login com as credenciais relativas a cada utilizador, assim como, efetuar operações, após a sessão inicializada.

Mais abaixo será possível observar detalhadamente como procedemos na realização do projeto e também justificar as decisões tomadas conforme o avanço da escrita do código.

2-Descrição das classes

Neste trabalho foram criadas 5 classes sem contar com a principal denominada Projeto com o seguinte aspeto:

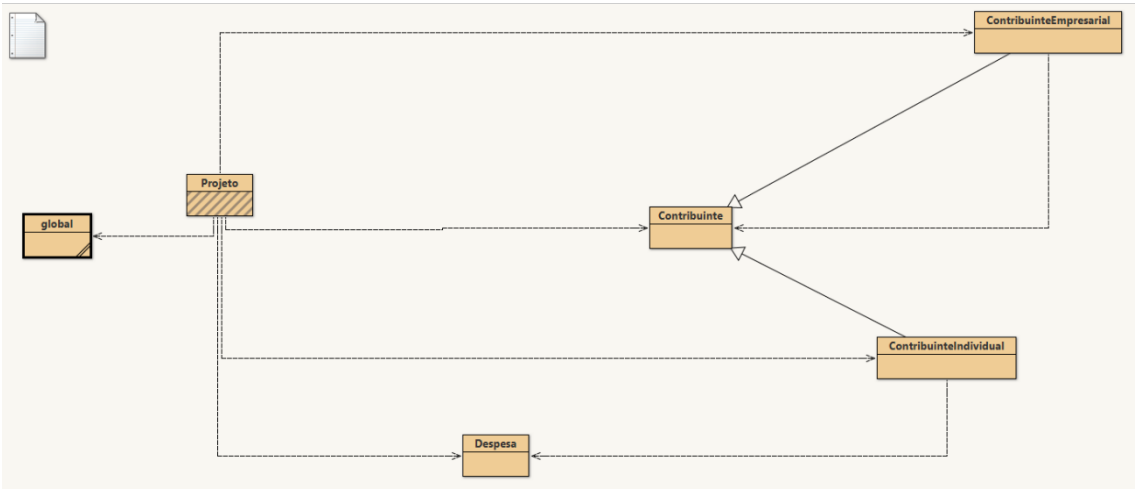

Figura 1 - Estrutura

Ou seja, temos a Classe Contribuinte que contém informações comuns tanto á classe Contribuinte Individual como á Contribuinte Empresarial. Estas duas últimas acrescentam mais alguma informação á classe anterior. De notar que todos os Contribuintes serão guardados em dois Maps:

- Map<String,ContribuinteEmpresarial>();
- Map<String,ContribuinteIndividual>();

Depois temos a classe Despesa que contém informações relativas a uma fatura, tais como, número fiscal emitente, designação do emitente, número fiscal do cliente, descrição da despesa, data da despesa, natureza da despesa e, por fim, o valor da despesa. Esta informação será gradualmente adicionada num ArrayList<Despesa> armazenado nos contribuintes envolvidos.

Finalmente, temos a classe global, na qual, existem dois Maps do mesmo tipo dos Contribuintes Individuais e Empresarias. A razão pela qual isto acontece é porque vamos usar as funções setEmp(Map x) e setInd(Map y), onde x é um determinado Map de Contribuintes Empresariais e y é um determinado Map de Contribuintes Individuais, para guardar a informação, de modo a que possa ser reutilizada num futuro próximo. Ora, esta última parte só pode ser concretizada com a ajuda das duas funções implementadas dentro da classe sendo, neste caso, chamadas guardaEstado(String nomeFicheiro) e leEstado(String nomeFicheiro);

3-Descrição da aplicação

Após a leitura do enunciado do projeto prático chegamos á conclusão que a solução ideal passaria por utilizar menus de modo que tornaria mais fácil a interpretação pelo utilizador. Assim sendo, começamos por criar um menu inicial com duas opções, Login e Registo. No entanto, previamente é feita uma condição, na qual, caso não exista qualquer utilizador em registo, o programa avança diretamente para a página de registo. Apesar de, neste último caso, não existir qualquer conta registada, esta situação não inteiramente verdadeira, uma vez que, existe sempre um utilizador em registo que neste caso é o Administrador ao qual associamos como nome de utilizador 'admin' e como password '12345';

```
Nada em registo!
****************************************
*               Bem vindo              *
****************************************


Nenhuma conta em registo.
Iremos encaminha-lo para a pagina de registo.

----------------------------------------
          Pagina de registo
----------------------------------------

 1-Contribuinte Individual
 2-Contribuinte Empresarial
Indique o tipo de Contribuinte:
```

Figura 3 - Página de registo

```
Escolha uma opcao:
 1-Login
 2-Registo

Opcao:
```

Figura 2 - Menu Login

Na página de registo são dadas duas opções, registar como Contribuinte Individual ou como Contribuinte Empresarial. O utilizador escolhe a opção que lhe convém e o programa avança para a área de registo, na qual, são pedidas as seguintes informações:

- Comuns a ambos:
  - Nome;
  - Morada;
  - Password;
  - Número Fiscal;
  - E-mail.

- Referente ao Contribuinte Empresarial
  - Áreas de atividade.
- Referente ao Contribuinte Individual
  - Número de dependentes do agregado fiscal;
  - Números fiscais dos dependentes do agregado familiar.

```
---------- Informacoes ----------


Insira as suas info's
E-mail: exemplo@uminho.pt
Nome: exemplo exemplo exemplo
Morada: Rua do exemplo
Numero de dependentes do agregado familiar: 3

Numeros fiscais do agregado familiar:
Numero fiscal (9 digitos) da entidade 1:  xxxxxxxxx
Numero Fiscal Invalido
Por favor verifique o NIF: 111111111

Numero fiscal (9 digitos) da entidade 2:  222222222

Numero fiscal (9 digitos) da entidade 3:  333333333
```

Figura 4 - Registo Individual

```
---------- Informacoes ----------

Insira as suas info's
E-mail: exemplo@uminho.pt
Nome: exemplo
Morada: exemplo
Em quantas atividades opera: 2
Tabela de codigos deduziveis:

 ------------------------------------------------------------------------------------------------------------------------
| 0000 -                           Saude                           | 0001 -        Premios de seguros de saude          |
| 0010 - Juros de emprestimos para habitacao propria e permanente | 0011 - Rendas de imoveis para habitacao permanente |
| 0100 -           Encargos com a reabilitacao de imoveis         | 0001 -        Premios de seguros de saude          |
| 0101 -                           Educacao                        | 0110 -              Despesas Gerais                |
| 0111 -                       IVA de faturas                      | 1000 -                  Lares                      |
| 1001 -                   Pensoes de alimentos                    | 1010 -         PPR e fundos de pensoes             |
| 1011 -            Regime publico de capitalizacao                | 1100 -                 Donativos                   |
 ------------------------------------------------------------------------------------------------------------------------



Os codigos da lista acima servem apenas para casos de deducao fiscal.
Se o tipo de atividade praticado da estiver identificado acima deve inserir o codigo que tem.

Insira o codigo de atividade 1: 0000
Insira o codigo de atividade 2: 0010
```

Figura 5 - Registo Empresarial

De salientar que, ao longo do projeto, fomos aplicando filtros sempre que necessário. Por exemplo, sempre que seja necessário indicar um número

de contribuinte, este deverá ser composto por nove números(int) e verificar a validez no caso deste já estar registado. Outros exemplos serão também, as passwords (mínimo de 5 carateres) e a validez do e-mail (existência de '@');

Após o registo o programa regressa ao menu inicial e depois do utilizador iniciar sessão com as credenciais será aberto um novo menu que será diferente caso o Contribuinte seja Individual ou Empresarial e, também, caso as credenciais sejam as do administrador.

No menu do Contribuinte Individual o utilizador poderá aceder a faturas, ver o total deduzido, o total gasto, o total deduzido pelo seu agregado familiar e, também, classificar faturas que não têm tipo de atividade definido. No campo de classificação de faturas, é emitido um aviso mal o utilizador entra no sistema de quantas faturas tem por classificar.

No menu do Contribuinte Empresarial o utilizador poderá aceder a faturas, por valor, por data e por contribuinte, associar faturas a um determinado contribuinte e observar o total faturado, tanto na história da empresa, como num determinado intervalo de datas.

No menu do Administrador é possível obter a relação entre os dez contribuintes que mais gastam no sistema e a relação das X empresas que mais faturas emitiram e o valor deduzido por todas essas faturas. Neste menu é também apresentado um aviso de quantos contribuintes estão em registo.

```
Tem 5 contribuintes individuais e 1 contribuintes empresariais em registo.

0-Voltar
1-Relacao dos 10 contribuintes que mais gastam
2-Relacao dos X contribuintes que mais gastam
Opcao:
```

Figura 6 - Menu de Administrador

```
0-Voltar
1-Aceder a faturas
2-Associar faturas
3-Faturado
Opcao:
```

Figura 8 - Menu Empresarial

```
0-Voltar
1-Aceder a faturas
2-Agregado familiar
Opcao:
```

Figura 7 - Menu Individual

Discussão

No nosso trabalho em específico, apesar da forma que nós utilizamos para identificar os tipos de despesa não nos parecer a mais fiável á primeira vista, foi aquela com a qual nos sentimos mais confortáveis para a realização dessa parte do trabalho. Assim sendo, e, tendo em conta a forma como atuamos, para adicionar outros tipos de despesas teríamos apenas de adicionar no Map criado a tal suposta despesa.

Falando agora através de outro ponto de vista, talvez a melhor forma fosse através de classes de tipos de atividades, uma vez que, tornaria o código mais leve, menos confuso e mais fácil de trabalhar.

Uma possibilidade de algoritmo de despesa seria criar uma classe que tinha duas variáveis:

- Um Map com os valores de gastos e códigos;
- Uma Double que seria o valor deduzido.

Nesta classe, tínhamos a função setFaturas(Map x) para gravarmos as faturas do contribuinte e depois teríamos uma função para cada código que retornava o valor deduzido da fatura em questão, e por fim tínhamos uma função que faria a soma das deduções fiscais de todas as faturas e associava á tal Double definida no inicio da classe. Para se saber o valor da dedução fiscal sempre que se precisasse criávamos outra função (double getDeducao()) que retornava o valor.