

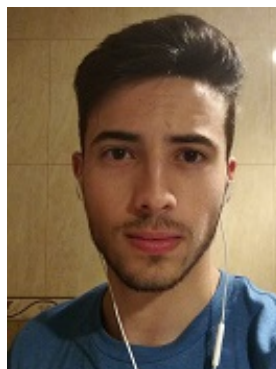


**Universidade do Minho**

## JavaFatura Grupo 40

José Fernandes (a82467)  
Marco Dantas (a81736)  
César Borges (a81644)

Maio 2018



# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>4</b>
<b>2</b>	<b>Arquitetura de classes</b>	<b>5</b>
2.1	Contribuinte . . . . .	5
2.1.1	ContribuinteIndividual . . . . .	5
2.1.2	ContribuinteEmpresarial . . . . .	5
2.2	BeneficioFiscal . . . . .	5
2.3	Contribuintes . . . . .	6
2.3.1	Metodos importantes . . . . .	6
2.4	Fatura . . . . .	6
2.5	Faturas . . . . .	6
2.5.1	Metodos importantes . . . . .	7
2.6	Morada . . . . .	7
2.7	Localidade . . . . .	8
2.7.1	LocalidadeLitoral e LocalidadeCentro . . . . .	8
2.8	Localidades . . . . .	8
2.9	AtividadesEconomicas . . . . .	8
2.10	Exceptions . . . . .	9
2.11	Comparators . . . . .	9
<b>3</b>	<b>Descrição da aplicação desenvolvida</b>	<b>10</b>
3.1	Menu inicial . . . . .	10
3.2	Registar um contribuinte . . . . .	10
3.3	Autenticação de contribuintes . . . . .	10
3.4	Menu do contribuinte empresarial . . . . .	11
3.4.1	Criar faturas associadas a despesas feitas por um contribuinte individual . . . . .	11
3.4.2	Obter as listagens das faturas por contribuinte num determinado intervalo de tempo . . . . .	11
3.4.3	Obter as listagens das faturas por valor decrescente de despesa . . . . .	11
3.4.4	Indicar o total faturado por uma empresa num determinado intervalo de tempo . . . . .	12
3.5	Menu do contribuinte individual . . . . .	12
3.5.1	Ver as despesas que foram emitidas ao contribuinte individual como cliente . . . . .	12
3.5.2	Ver montante de dedução fiscal acumulado do agregado familiar . . . . .	12
3.5.3	Associar classificação de uma atividade económica a um documento despesa . . . . .	12
3.5.4	Corrigir atividade económica de um documento despesa . . . . .	12
3.5.5	Obter as faturas de uma empresa ordenadas por data de emissão ou por valor . . . . .	13
3.5.6	Extra: Ver despesas pendentes . . . . .	13

3.6	Menu do administrador . . . . .	13
3.6.1	Determinar a coleção dos 10 contribuintes que mais gastam em todo o sistema . . . . .	13
3.6.2	Determinar a coleção de um número arbitrário de empresas que mais faturam no sistema e o montante de dedução fiscal que as empresas registam . . . . .	14
3.6.3	Determinar as correções feitas em faturas . . . . .	14
3.6.4	Extra: Adicionar localidades ao sistema . . . . .	14
3.7	Salv guarda do estado da aplicação num ficheiro . . . . .	14
<b>4</b>	<b>Como introduzir novos tipos de despesas e algoritmos de deduções fiscais</b>	<b>15</b>
<b>5</b>	<b>Conclusão</b>	<b>15</b>

# 1 Introdução

O enunciado deste projeto propôs a realização de uma aplicação que permite aos contribuinte interagirem com as suas faturas.

A aplicação desenvolvida deverá permitir a um contribuinte individual validar e consultar as suas faturas e verificar as deduções fiscais, por outro lado deve permitir a um contribuinte empresarial emitir faturas e associa-las a atividades económicas.

O projeto também terá de suportar uma interface para o administrador e fornecer métodos de consulta específicos.

## 2 Arquitetura de classes

Neste projeto utilizamos packages com o intuito de manter uma melhor organização do projeto daí não termos as classes todas agrupadas o que não permite obter uma imagem global do projeto no BlueJ.

### 2.1 Contribuinte

Classe que representa a principal entidade do programa, o contribuinte.

Agrega as características comuns a todos os tipos de contribuintes do sistema.

A classe é abstrata porque não queremos que sejam criadas instâncias de apenas Contribuinte uma vez que esta serve apenas para agrupar funcionalidades e variáveis comuns a todos os contribuintes.

#### 2.1.1 ContribuinteIndividual

Uma extensão de Contribuinte que junta as propriedades de um indivíduo.

Assumimos que os elementos dependentes do agregado familiar só podem ser filhos.

Foi utilizado um ArrayList para guardar uma lista com os NIF de cada elemento do agregado familiar.

É usado um HashMap para guardar as atividades económicas dedutíveis. Foi usado um HashMap porque há precisava-mos de manter uma correspondência de cada atividade a uma chave para uma procura mais fácil quando precisamos de determinar se uma atividade é dedutível ou não.

#### 2.1.2 ContribuinteEmpresarial

Uma extensão de Contribuinte que agrega as propriedades de uma empresa.

É utilizado um HashMap para agrupar as atividades económicas nas quais a empresa atua. É utilizado um HashMap para facilitar a inserção de atividades à empresa sem ter repetidos, uma vez que a ordem não nos interessa e porque temos uma maneira de obter chaves únicas para cada atividade.

### 2.2 BeneficioFiscal

Interface que é implementada nos Contribuintes individuais e empresariais, para implementar o método `reducaoImposto()` pedido nos novos requisitos da segunda fase.

Decidimos usar uma interface para não afetar muito o comportamento do programa e porque não fazia sentido na hierarquia inserir o método na classe Contribuinte.

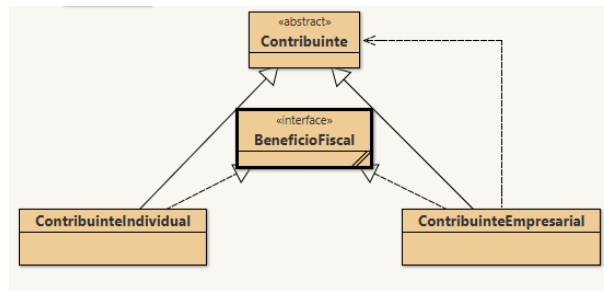


Figura 1: Hierarquia dos Contribuintes

## 2.3 Contribuintes

Junta todos os contribuintes numa estrutura de dados e gere-os

A estrutura é um HashMap cujas keys são NIFs de contribuintes e os valores objetos Contribuinte.

A classe contém métodos para aceder a utilizadores singulares ou para obter todos os utilizadores que manifestam determinadas características, também contém métodos para adicionar elementos ao conjunto dos agregados familiares.

### 2.3.1 Metodos importantes

**getXMostFaturado** Devolve a lista das N empresas que mais faturam no sistema.

**addNewContribuinteToAgregado** Cria um Contribuinte e adiciona-o a um agregado.

**addDependenteToAgregado** Adiciona n elementos dependentes a todos os elementos de um agregado familiar.

## 2.4 Fatura

Classe que representa uma Fatura. Optamos por guardar uma cópia dos contribuintes cliente e emitente no instante de tempo em que a fatura é criada para facilitar os cálculos de deduções de cada fatura, uma vez que as faturas são guardadas numa classe e os contribuintes são guardados noutra classe o calculo da dedução do cliente é feita na fatura. As faturas contêm um campo com o seu número de fatura, este campo é lhes fornecido pela classe faturas e apenas existe devido a uma necessidade de distinguir faturas.

## 2.5 Faturas

Classe que guarda e gere as faturas. As faturas são separadas em faturas pendentes e não

Permite através de métodos concretos corrigir a atividade económica de uma fatura e classifica-la como pendente ou não, através do modo como as guardamos.

Por fim também fornece a maioria dos métodos disponibilizados ao administrador e aos contribuintes relativos a faturas.

### 2.5.1 Metodos importantes

**getDespesasAndDFAcumulado** Devolve um par com a lista de faturas de um contribuinte assim como a sua despesa deduzida, juntamente com a do agregado.

**associaAtividadesEconomicas** Associa uma atividade económica a uma fatura caso ela não esteja já emitida.

**corrigeAtividadeEconomica** Corrige uma atividade económica, ou seja, atualiza a atividade nas faturas.

**getFaturasByDate e getFaturasByValor** Devolve uma lista de faturas ordenadas por data ou valor de despesa.

**getFaturasFromEmitenteBetweenDate** Apresenta a lista de todas as faturas entre duas datas.

**getFaturasByValorDecrescente** Devolve uma lista de faturas ordenadas por valor decrescente, mas também agrupadas por nif.

**totalFaturado** Devolve o total faturado por um contribuinte.

**getMostSpenders** Devolve uma lista com os contribuintes com mais despesas.

**getDFEmpresa** Calcula o montante de deduções fiscais fornecidas de todas as despesas emitidas por uma empresa.

## 2.6 Morada

Contém a Localidade, código postal e numero de porta de uma residência.

Criamos uma classe para a morada porque precisávamos de uma solução mais rica do que uma string.

## 2.7 Localidade

Contém a informação da região onde se encontra o contribuinte.

Esta classe é abstrata e, neste caso, apenas temos duas classes que a herdam. Esta implementação permite-nos ter N classes que especifiquem uma determinada região.

A classe é abstrata porque não queremos que sejam criadas instâncias de apenas Localidade uma vez que esta serve apenas para agrupar funcionalidades comuns a todas as Localidades.

### 2.7.1 LocalidadeLitoral e LocalidadeCentro

Classes que especificam as localidades litoral e centro, contendo os benefícios específicos destas regiões, para termos de dedução.

## 2.8 Localidades

Classe que agrega as Localidades existentes no sistema.

Permite gerir as Localidades, não permitindo que haja engano ou fraude por parte de um contribuinte ao atribuir de forma errada a sua localidade, para depois ter o benefício fiscal correto.

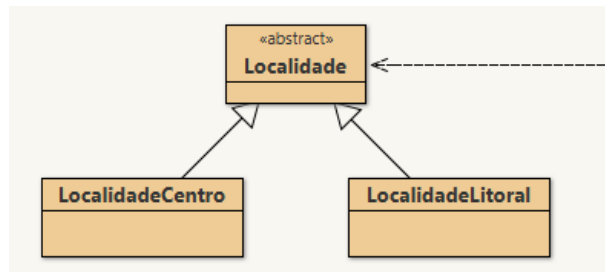


Figura 2: Hierarquia das Localidades

## 2.9 AtividadesEconomicas

Classe que contém o coeficiente fiscal de uma determinada atividade

Esta classe é abstrata porque não queremos que sejam criadas instâncias de AtividadesEconomicas diferentes uma vez que esta serve apenas para agrupar classes com funcionalidades comuns. Classes que a herdam:

- *Educacao*
- *Imoveis*
- *Lares*
- *PassesTranspostes*



- *ReparacaoManutencaoVeiculos*
- *ReparacaoManutencaoMotociclos*
- *Restauracao*
- *Saude*
- *Veterinario*
- *CabeleiroBeleza*

## 2.10 Exceptions

**ContribuinteDoesntExistException** Exceção para quando queremos aceder a um contribuinte que está guardado numa estrutura de dados e ele não existe.

**ContribuinteNaoIndividualException** Para quando queremos adicionar um contribuinte a um agregado, mas ele não é individual.

**FaturaPendenteException** Para sempre que quisermos adicionar ou corrigir a atividade económica de uma fatura e ela encontra-se pendente.

**FaturaNaoPendenteException** Para sempre que quisermos adicionar ou corrigir a atividade económica de uma fatura e ela encontra-se não pendente.

## 2.11 Comparators

**CompareFaturasByDate** Utilizado para ordenar as faturas por data. Usado em `getFaturasByDate(...)`

**CompareFatrurasByValor** Utilizado para ordenar as faturas por valor de despesa. Usado em `getFaturasByValor(...)`

**ComparePairDespesa** Utilizado para ordenar pares pelo valor do seu segundo elemento, que é a despesa. Usado em `getMostSpender(...)`.

## 3 Descrição da aplicação desenvolvida

### 3.1 Menu inicial

```
Welcome User  
1-Registar novo Contribuinte  
2-Log In  
0-Sair
```

O menu inicial da aplicação consiste em duas opções apenas, a de registar um novo contribuinte e de autenticar um utilizador já existente ou o administrador.

### 3.2 Registar um contribuinte

```
Registando um novo Contribuinte  
1-Registar contribuinte Individual  
2-Registar contribuinte Empresarial  
3-Retroceder  
0-Sair
```

Para registar um novo contribuinte escolhe-se a opção 1 do menu inicial e de seguida, caso queira registar um contribuinte individual pressione 1, caso queira registar um contribuinte empresarial pressione 2. Se se enganou no menu pode utilizar a opção 3 para voltar ao menu inicial ou a opção 0 para sair da aplicação.

### 3.3 Autenticação de contribuintes

```
Introduza o seu NIF:  
Nif do Contribuinte  
Introduza a sua palavra-passe:  
Palavra-passe do Contribuinte
```

Para autenticar um contribuinte escolhe-se a opção 2 do menu inicial e após isto, o utilizador deve de inserir o seu Nif e a sua palavra-passe, a autenticação não é feita se um dos dados fornecidos esteja errado e o utilizador é retornado para o menu inicial, caso contrário, um novo menu aparecerá com novas opções. Ainda se consta que é aqui que um administrador possa de autenticar e que ele próprio não se encontra na estrutura de dados do menu.

### 3.4 Menu do contribuinte empresarial

```
Authentication sucessful
Em que lhe posso ajudar Hospital
1-Criar fatura
2-Ver faturas
3-Ver total faturado num intervalo de tempo
4-Log out
0-Sair
```

#### 3.4.1 Criar faturas associadas a despesas feitas por um contribuinte individual

Para uma empresa criar uma fatura é necessário que esta faça a sua autenticação primeiro.

Pressione 1 para criar uma nova fatura, serão pedidos o NIF do cliente, uma descrição da despesa e o seu valor. Caso o contribuinte empresarial que emita a fatura tenha mais que uma atividade económica possível, a fatura ficará pendente até que o cliente associe a atividade correta, caso tenha apenas uma atividade possível, a fatura é válida e terá essa mesma atividade.

Ainda se consta que é impossível haver um contribuinte empresarial com nenhuma atividade económica.

#### 3.4.2 Obter as listagens das faturas por contribuinte num determinado intervalo de tempo

```
1-Ver todas as faturas
2-Ver faturas por contribuinte num intervalo de tempo
3-Ver faturas por contribuinte por ordem decrescente
4-Retroceder
0-Sair
```

Para obter as faturas de um determinado contribuinte num intervalo de tempo é preciso selecionar a opção 2 no menu do contribuinte empresarial e de seguida a opção 2 do menu acima.

Será pedido as duas datas que delimitam as faturas e após isto aparecerá no ecrã as faturas.

#### 3.4.3 Obter as listagens das faturas por valor decrescente de despesa

Para obter as faturas por valor decrescente de despesa é preciso selecionar a opção 2 do menu do contribuinte empresarial e depois a opção 3 do menu acima.

Não é preciso introduzir nenhum dado para esta ação.

### 3.4.4 Indicar o total faturado por uma empresa num determinado intervalo de tempo

Para obter as faturas por valor decrescente de despesa é preciso selecionar a opção 3 do menu do contribuinte empresarial.

Terá que introduzir duas datas e após isto será mostrado no ecrã o total faturado pela empresa nesse intervalo de tempo.

## 3.5 Menu do contribuinte individual

```
Authentication sucessful
Em que lhe posso ajudar Kratos
1-Ver despesas
2-Ver despesas pendentes
3-Ver montante de dedução fiscal acumulado do agregado familiar
4-Associar uma atividade economica a uma despesa
5-Corrigir classificação de uma atividade economica
6-Ver lista de faturas de uma empresa
7-Log out
0-Sair
```

### 3.5.1 Ver as despesas que foram emitidas ao contribuinte individual como cliente

Para obter as despesas referentes ao contribuinte individual deve de pressionar a opção 1 do seu menu.

Não é necessário introduzir dados para esta ação.

### 3.5.2 Ver montante de dedução fiscal acumulado do agregado familiar

Para obter o montante de dedução fiscal acumulado do agregado familiar selecione a opção 3 do menu do contribuinte individual.

Não é necessário introduzir dados para esta ação.

### 3.5.3 Associar classificação de uma atividade económica a um documento despesa

Para associar uma atividade económica a um documento é preciso selecionar a opção 4 do menu do contribuinte individual.

Será pedido para o utilizador introduzir o número da fatura e a atividade que quer associar à fatura.

Apenas se pode associar uma atividade económica a uma fatura pendente.

### 3.5.4 Corrigir atividade económica de um documento despesa

Para corrigir uma atividade de um documento despesa seleciona-se a opção 5 do menu do contribuinte individual.

Será pedido para o utilizador introduzir o número da fatura e a nova atividade, que irá substituir a antiga.

Esta ação deixa um registo que pode ser vista por um administrador.

### 3.5.5 Obter as faturas de uma empresa ordenadas por data de emissão ou por valor

```
1-Ver faturas de uma empresa ordenada por data
2-Ver faturas de uma empresa ordenada por valor
3-Retroceder
0-Sair
```

Para usar esta opção deve de seleccionar a opção 6 do menu do contribuinte individual.

Apenas se pede ao utilizador para introduzir o NIF da empresa da qual quer ver as faturas.

Se o utilizador quiser ver as faturas ordenadas por data de emissão deve pressionar 1 ou caso queira ver-las ordenadas por valor deve pressionar 2 do menu acima.

### 3.5.6 Extra: Ver despesas pendentes

Para ver as despesas pendentes é preciso seleccionar a opção 2 do menu do contribuinte individual.

Não é necessário introduzir qualquer dado.

Decidimos colocar esta opção extra para tornar mais simples e clara a interação do contribuinte com a opção para associar uma atividade a uma despesa.

## 3.6 Menu do administrador

```
Welcome Administrator
1-Ver os 10 contribuintes que gastam mais no sistema
2-Ver as empresas que faturam mais e as sua dedução fiscal
3-Ver correcoes de faturas
4-Adicionar localidades
5-Log out
0-Sair
```

### 3.6.1 Determinar a coleção dos 10 contribuintes que mais gastam em todo o sistema

Para usar esta ação tem de seleccionar a opção 1 do menu do administrador. Não são pedidos dados para esta ação.

### **3.6.2 Determinar a coleção de um número arbitrário de empresas que mais faturam no sistema e o montante de dedução fiscal que as empresas registam**

Para ver as empresas que faturam mais no sistema é preciso selecionar a opção 2 do menu do administrador.

Será pedido para introduzir o número de empresas que quer ver.

Caso esse número seja maior que o número de empresas existentes no sistema aparecerão todas as empresas.

### **3.6.3 Determinar as correções feitas em faturas**

Para ver as correções feitas em faturas pelos contribuintes individuais deve de selecionar a opção 3 do menu do administrador.

Não é preciso introduzir qualquer dado.

### **3.6.4 Extra: Adicionar localidades ao sistema**

Para adicionar localidades ao sistema tem de selecionar a opção 4 do menu do administrador.

Será pedido para indicar o nome da localidade e de seguida se é uma localidade do litoral ou do centro, caso seja do centro, ainda é pedido que introduza o benefício dessa empresa.

Nós decidimos colocar esta opção extra para impedir que um novo contribuinte ao se registar não possa explorar a aplicação e definir o seu próprio benefício fiscal, pois depende da sua morada. Desta forma um novo contribuinte apenas introduz a sua localidade e o seu benefício fiscal é lhe atribuído automaticamente.

## **3.7 Salvaguarda do estado da aplicação num ficheiro**

A salvaguarda do estado da aplicação é feita quando o utilizador sai da aplicação, ou seja, quando escolhe a opção 0 de qualquer menu, sendo possível, ao iniciar novamente a aplicação, recuperar todos os dados.

## 4 Como introduzir novos tipos de despesas e algoritmos de deduções fiscais

Para introduzir novos tipos de despesas criam-se subclasses de `AtividadeEconómica`.

Para introduzir novos algoritmos de deduções fiscais criam-se subclasses de `Fatura` e das classes que implementam `BeneficioFiscal` e alteram-se os seguintes métodos:

- **`reducaoImposto`** nas classes que implementam `BeneficioFiscal`
- **`getValorDeduzivel`** na classe `Fatura`.

Também poderiam ser criados subclasses mais elaboradas de `Localidade` de modo a aumentar o número de variáveis que poderiam entrar nos cálculos das deduções.

## 5 Conclusão

Conseguimos resolver o problema proposto pelo enunciado sem grandes dificuldades. Um dos maiores problemas que encontramos foi ao perceber as questões do enunciado relacionados com as deduções, os problemas variavam entre o algoritmo e a dificuldades a perceber enunciado.