

JFindMyFiles

Manual de Programador Projecto Informático n.º AP01 / 2007-

8

Patrícia Monteiro, n.º 12435

Sérgio Lopes, n.º 10635

Manual de Programador para o sistema JFindMyFiles submetido para avaliação parcial da unidade curricular de Projecto Informático, do curso de Engenharia Informática. Projecto sob a orientação da Professora Ana Filipa Nogueira, do Departamento de Engenharia Informática da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria.



Departamento de Engenharia Informática da
Escola Superior de Tecnologia e Gestão do
Instituto Politécnico de Leiria

22 de Julho de 2008

Índice

1	Manual do Programador	4
1.1	<i>Criação de Extensões.....</i>	<i>4</i>
1.2	<i>Criação de um Template de Exportação.....</i>	<i>5</i>

Índice de tabelas

Tabela 1.	Tabela de Marcadores para criação de Templates HTML.....	6
-----------	--	---

1 Manual do Programador

1.1 Criação de Extensões

O *software* permite a criação de extensões, que serão usados durante a leitura de ficheiros, para a obtenção de informações mais específicas sobre um determinado ficheiro. A criação de uma extensão passa pela sua programação, usando a linguagem Java.

A classe a ser usada como ponto de entrada deverá implementar a interface `de.berlios.jfindmyfiles.readingfiles.pluginapi.Reader`, definida na biblioteca `de-berlios-jfindmyfiles-readingfiles.jar`, contida na pasta de instalação da aplicação.

A interface define os seguintes métodos:

- `getName` – Método que devolve um valor do tipo `String` que representa o nome da extensão. O nome de uma extensão não necessita ser único, uma vez que é usado apenas na apresentação ao utilizador, mas deve manter-se constante durante todas as execuções da aplicação. Sugere-se a utilização de um atributo de classe, constante, para a definição do nome.
- `getAuthor` – Método que permite obter o nome do autor da extensão. Tal com o método anterior, o valor devolvido é do tipo `String` e, do mesmo modo se recomenda o uso de atributos de classe constantes.
- `pluginFor` – Método que permite determinar a extensão do ficheiro sobre o qual esta extensão irá actuar. Deve ser notado, no entanto, que a extensão de um ficheiro não terá significado em todos os sistemas operativos, mas esta é a forma de identificar uma extensão da aplicação inequivocamente. Este método deverá devolver um valor do tipo `String` que contenha a extensão comumente usada nos ficheiros que se pretendem ler, caso duas extensões da aplicação se registem para o mesmo tipo de ficheiro, apenas a última a ser registada prevalece, sendo que a ordem pela qual as extensões da aplicação são registadas, não pode ser garantida. Cabe ao utilizador, e não ao programador da extensão de aplicação, acautelar qualquer situação de conflito.
- `read` – Este é o método que efectua a leitura das informações de ficheiros. É-lhe passado um parâmetro, do tipo `File`, que representa o ficheiro a ler. A leitura de informação é síncrona, pelo que se espera que este método termine o mais rapidamente possível, não efectuando nada mais além do estritamente necessário para

a obtenção e devolução da informação. Toda a informação é encapsulada num objecto do tipo **DescriptionValues**, que é devolvido por este método no fim da sua execução. O objecto não deverá ser null, pelo que, caso seja, deverá ser devolvido um objecto criado com o construtor sem argumentos da classe **DescriptionValues**.

- **isActive** – Forma de determinar se uma extensão se encontra activo ou não. Este método deve ser implementado para reflectir o estado da extensão e não deve devolver sempre um valor predefinido. Ao implementar uma extensão, o programador não se deve preocupar com quem irá alterar o valor mas sim garantir que é possível alterar o estado da extensão, uma implementação típica será:

```
private boolean active;  
  
public boolean isActive() {  
    return active;  
}
```

- **setActive** – Permite definir o estado de uma extensão. Este método deverá ser implementado para que o método anterior, **isActive**, reflecta correctamente o estado da extensão.

1.2 Criação de um Template de Exportação

A opção de exportação para HTML, que permite a criação de um ficheiro HTML contendo todo o catálogo, num formato facilmente transportável e apresentável, usa marcadores especiais em ficheiros HTML base, os *templates*, de forma a colocar os dados nas posições correctas.

Para criar um *template* HTML é apenas necessário criar o ficheiro e colocar nos locais apropriados, os marcadores que serão usados. Após a criação, os ficheiros que compõem o *template*, tipicamente um ficheiro HTML, as imagens e algum ficheiro CSS, devem ser colocados dentro da pasta definida como local de pesquisa de *templates*.

Dado que o sistema permite apresentar uma imagem, com uma pré-visualização do aspecto final, é conveniente que seja criado um ficheiro de imagem, com extensão JPG. Esta imagem será apresentada ao utilizador na janela de diálogo de exportação para HTML.

Tabela 1. Tabela de Marcadores para criação de Templates HTML

Marcador	Descrição
«CATALOG_DESCRIPTION»	Descrição do catálogo
«CATALOG_DISKNUMBER»	Número de discos catalogados
«CATALOG_CREATION»	Data de criação do catálogo
«CATALOG_FILES»	Número de ficheiros existentes no catálogo
«CATALOG_FOLDERS»	Número de pastas existentes
«CATALOG_SIZE»	Tamanho total do catálogo
«DISKGROUP_START»	Início da secção de listagem de grupos de disco
«DISKGROUP_NAME»	Nome do grupo de discos
«DISKGROUP_DESCRIPTION»	Descrição do grupo de discos
«DISKGROUP_CAPACITY»	Capacidade total do grupo de discos
«DISKGROUP_END»	Fim da secção de listagem de grupos de discos
«MEDIA_START»	Início da secção de apresentação de discos
«MEDIA_NAME»	Nome do disco
«MEDIA_DESCRIPTION»	Descrição do disco
«MEDIA_CAPACITY»	Capacidade do disco
«MEDIA_FREESPACE»	Espaço livre no disco

«MEDIA_LASTMODIFIED»	Data da última alteração que o disco sofreu
«MEDIA_LOCATION»	Localização do disco
«MEDIA_END»	Fim da secção de apresentação dos discos
«LOAN_CREATION_DATE»	Data de criação empréstimo
«LOAN_RETURNED_DATE»	Data de término do empréstimo
«LOAN_USER»	Utilizador do empréstimo
«LOAN_MEDIA»	Disco emprestado
«LOAN_START»	Início da lista de empréstimos
«LOAN_END»	Fim da lista de empréstimos
«USER_START»	Início da lista de utilizadores
«USER_FIRSTNAME»	Nome do utilizador
«USER_SURNAME»	Apelido do utilizador
«USER_LOAN_LIST»	Início da lista de empréstimos deste utilizador
«USER_END»	Fim da lista de utilizadores
«FILE_START»	Início da lista de ficheiros
«FILE_END»	Fim da lista de ficheiros
«FILE_NAME»	Nome do ficheiro
«FILE_PARENT»	Pasta do ficheiro, caso se aplique

«FILE_DISK»	Disco do ficheiro
«FILE_CHILDREN_START»	Início da lista de ficheiros contidos, caso se aplique
«FILE_CHILDREN_ITEM»	Ficheiro contido, caso se aplique
«FILE_CHILDREN_END»	Fim da lista de ficheiros contidos, caso se aplique
«FILE_DESCRIPTION»	Descrição do ficheiro
«FILE_SIZE»	Tamanho do ficheiro
«FILE_SHA1»	<i>Hash</i> SHA-1 do ficheiro
«FILE_ISHIDDEN»	Indicação de ficheiro escondido
«FILE_EXTENSION»	Extensão do ficheiro
«FILE_LASTMODIFIED»	Data da última modificação do ficheiro
«FILE_ABSOLUTPATH»	Caminho absoluto para o ficheiro