

**manual**

**versão 0.3**

**Português do Brasil (*Brazilian Portuguese*)**

**FunTester 0.7c**

**Versões deste documento**

0.3 – Melhoria da Introdução; Descritos detalhes sobre Instalação e Extensões de Arquivos; Descritos tópicos sobre Conexões e Consultas a Bancos de Dados; Descrito tópico sobre Expressões Regulares.

0.2 – Adicionadas informações sobre instalação e execução.

0.1 – Primeira versão.

Sumário

[1 Introdução 1](#_Toc395714368)

[1.1 Instalação 1](#_Toc395714369)

[1.1.1 Instalando drivers de bancos de dados 1](#_Toc395714370)

[1.2 Execução 2](#_Toc395714371)

[1.3 Extensões dos Arquivos 3](#_Toc395714372)

[2 Uso 4](#_Toc395714373)

[2.1 Configurando para Português do Brasil 4](#_Toc395714374)

[2.2 Criando um Software 4](#_Toc395714375)

[2.3 Criando um Ator 5](#_Toc395714376)

[2.4 Criando um Caso de Uso 5](#_Toc395714377)

[2.5 Criando Fluxos em um Caso de Uso 6](#_Toc395714378)

[2.5.1 Criando um Fluxo Principal 6](#_Toc395714379)

[2.5.2 Criando um Fluxo Alternativo 9](#_Toc395714380)

[2.6 Criando Conexões e Consultas a Bancos de Dados 10](#_Toc395714381)

[2.6.1 Criando Consultas 12](#_Toc395714382)

[2.7 Criando Expressões Regulares 13](#_Toc395714383)

[2.8 Detalhando Elementos e suas Regras de Negócio 14](#_Toc395714384)

[2.9 Gerando e Executando Testes 14](#_Toc395714385)

# Introdução

FunTester (acrônimo para *Functional Tester*) é uma ferramenta para testes baseados em modelos (*model-based testing*) que permite documentar a especificação de softwares usando casos de uso e utilizá-la para automaticamente gerar testes funcionais, executá-los e verificar se a implementação do software sob teste está de acordo com a sua especificação.

Os testes gerados pela ferramenta contêm dados e oráculos, que são criados a partir da especificação das regras de negócio do software, e visam simular tanto as condições normais de uso quanto explorar situações anômalas, visando expor defeitos. Essas regras de negócio podem descrever restrições sobre os dados informados pelo usuário quando ele interage com o sistema e a expectativa de comportamento do sistema quando esses dados não atendem às restrições estabelecidas.

Entras as possibilidades de descrição das restrições sobre os dados informados pelo usuário, encontra-se a possibilidade de descrever consultas a bancos de dados. Desta forma, um testador pode preparar uma massa de dados de teste contendo dados similares ao utilizados em ambiente de produção (em que o software é usado por seus usuários finais), para que os testes possam ser executados com dados realistas. Além disso, os demais dados de teste gerados pela ferramenta exploram condições limítrofes e outras propensas a descobrir defeitos, usando técnicas de análise de valor limite e particionamento de equivalência, visando aumentar a taxa de descoberta de defeitos. Assim, a ferramenta pode gerar rapidamente uma série de testes úteis e aumentar a produtividade dos testadores, além de garantir que diversas situações de execução são exercitadas no software sob teste, o que pode aumentar sua robustez.

**Este manual visa cobrir aspectos do uso da ferramenta, não da abordagem utilizada por ele. Em alguns momentos, entretanto, alguns aspectos de sua abordagem podem ser discutidos, para melhorar a compreensão de seu funcionamento.**

## Instalação

Para instalar o programa, basta extrair o arquivo ZIP para uma pasta de seu computador. É necessário ter instalado o [Java 6](http://www.java.com) (*Java Runtime Environment* 6) ou posterior, ter um dispositivo apontador (como um mouse), um teclado e um monitor com suporte a resolução 1280x720.

### Instalando drivers de bancos de dados

Para usar um bancos de dados com o FunTester, é necessário:

1. Colocar o arquivo de seu respectivo driver JDBC, que tem a extensão JAR, na pasta jdbc; e
2. Criar/configurar seus respectivos arquivos de modelos (*templates*), que possuem formato JSON e extensão DBT, de *database template*.

Para obter o driver de seu banco de dados, consulte seu respectivo site oficial. A lista a seguir contém links site de driver JDBC de alguns bancos de dados conhecidos:

* [Firebird](http://www.firebirdsql.org/en/jdbc-driver/)
* [HSQLDB](http://hsqldb.org/)
* [MySQL](http://dev.mysql.com/downloads/connector/j/)
* [Oracle](http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html)
* [PostgreSQL](http://jdbc.postgresql.org/)
* [SQL Server](http://msdn.microsoft.com/en-us/sqlserver/aa937724.aspx)
* [SQLite](https://bitbucket.org/xerial/sqlite-jdbc/downloads)

A versão 0.7a (e posteriores) do FunTester vem com os modelos (templates) mysql.dbt, firebird.dbt e postgres.dbt. O código a seguir mostra o conteúdo do arquivo mysql.dbt, que possui formato JSON:

{

"name" : "MySQL JDBC Driver",

"type" : "mysql",

"driverFile" : "mysql-connector-java-5.1.28-bin.jar",

"driverClass" : "com.mysql.jdbc.Driver",

"defaultPort" : 3306,

"defaultUser" : "root",

"defaultPassword" : ""

}

O arquivo contém as seguintes propriedades:

* **name**: Nome do modelo/driver;
* **type**: Tipo do banco de dados. Deve ser informado sem espaços e em letras minúsculas, conforme o tipo de conexão JDBC;
* **driverFile**: Nome do driver utilizado pelo modelo. Este driver deve existir na pasta jdbc;
* **driverClass**: Nome da classe do driver que deve ser acessada para instanciar uma conexão do banco de dados. O valor dessa propriedade é comumente encontrado no site oficial do respectivo driver;
* **defaultPort**: Porta padrão de conexão com o banco de dados;
* **defaultUser**: Usuário padrão de conexão com o banco de dados;
* **defaultPassword**: Senha padrão de conexão com o banco de dados.

Basta editá-las, segundo as informações do driver e do banco de dados utilizado.

## Execução

Para executar o programa pela interface gráfica:

1. No **Windows** (XP ou superior), dê um duplo clique sobre o arquivo funtester.jar. Também é possível executá-lo pelo arquivo funtester.bat, clicando duas vezes sobre ele;
2. No **Linux** ou no **MacOS X**, dê um duplo clique sobre o arquivo funtester.sh.

Para executar o programa pela linha de comando:

java –jar funtester.jar

## Extensões dos Arquivos

FunTester armazena seus dados em arquivos JSON,[[1]](#footnote-1) de forma que você pode mantê-los junto ao código-fonte de seu sistema e controlá-los com seu programa de controle de versão preferido (como Subversion,[[2]](#footnote-2) Git,[[3]](#footnote-3) Mercurial,[[4]](#footnote-4) etc.).

Apesar dos arquivos terem formato JSON, suas extensões variam, para que possam ser reconhecidas mais facilmente:

|  |  |
| --- | --- |
| **Extensão** | **Conteúdo/Significado** |
| *cfg* | Configurações básicas do FunTester. |
| *dbt* | Contém um modelo (*template*) de configuração de *driver* de banco de dados. |
| *fat* | Contém os testes abstratos, gerados pelo FunTester. |
| *fp* | Contém um perfil, utilizado para gerar os testes abstratos. |
| *fs* | Informações do software criado no projeto (*.fun*). |
| *ftr* | Contém os resultados da execução dos scripts de teste. Este arquivo é gerado por um *plug-in* de execução de testes. |
| *fun* | Projeto do FunTester. Ao criar um software, um arquivo de projeto é criado. |
| *fv* | Contém um vocabulário, utilizado pelo projeto para descrever o software. |
| *pcfg* | Configuração de *plug-in* para o projeto (*.fun*). |
| *plg* | Descreve um *plug-in* utilizado pelo FunTester. |

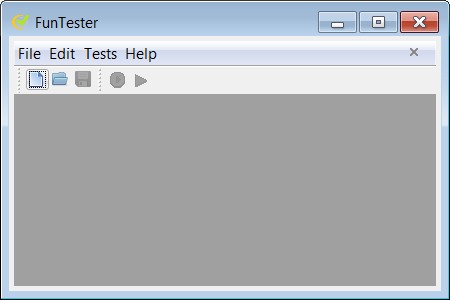
As extensões de arquivo comumente manipuladas por um projeto de software são **fat**, **ftr**, **fs**, **fun** e **pcfg**.

# Uso

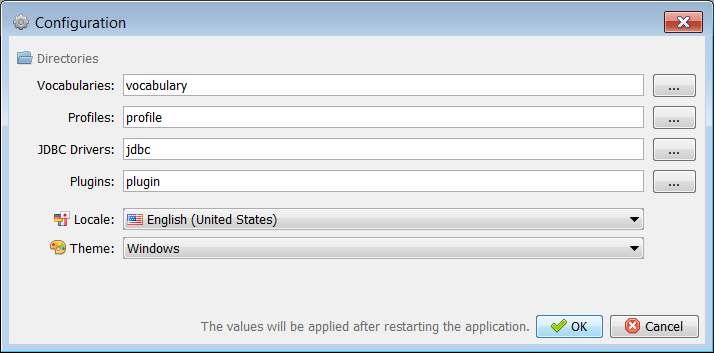
Esta seção descreve os principais aspectos do uso da ferramenta.

## Configurando para Português do Brasil

Ao iniciar o programa, será exibida uma tela inicial como a abaixo:



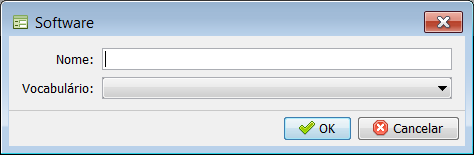
Para configurar o programa para Português do Brasil, acesse o menu Edit 🡪 Configuration.



Mude o Locale para Portuguese (Brazil) e clique em OK. **Reinicie o programa para aplicar as configurações**.

## Criando um Software

Para criar um Software, acesse o menu Arquivo 🡪 Novo.

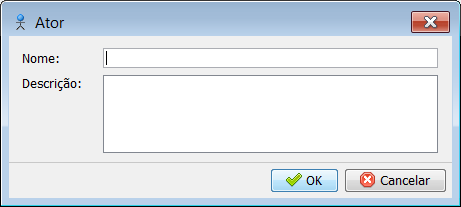


Digite o Nome, escolha um Vocabulário e clique em OK. Um Vocabulário é um conjunto de palavras disponível para preencher os casos de uso do sistema, num idioma específico. Ações como "clica", "digita", "move", etc. são exemplos de palavras que podem fazer parte de um vocabulário.

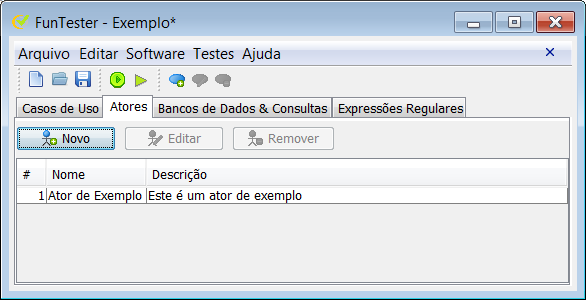
Ao criar o Software, serão exibidos os menus Software e Testes, bem como as abas Casos de Uso, Atores, Bancos de Dados & Consultas e Expressões Regulares.

## Criando um Ator

Um Ator é uma pessoa ou dispositivo que interage com o sistema. Para criar um Ator, acesse Software 🡪 Novo Ator.

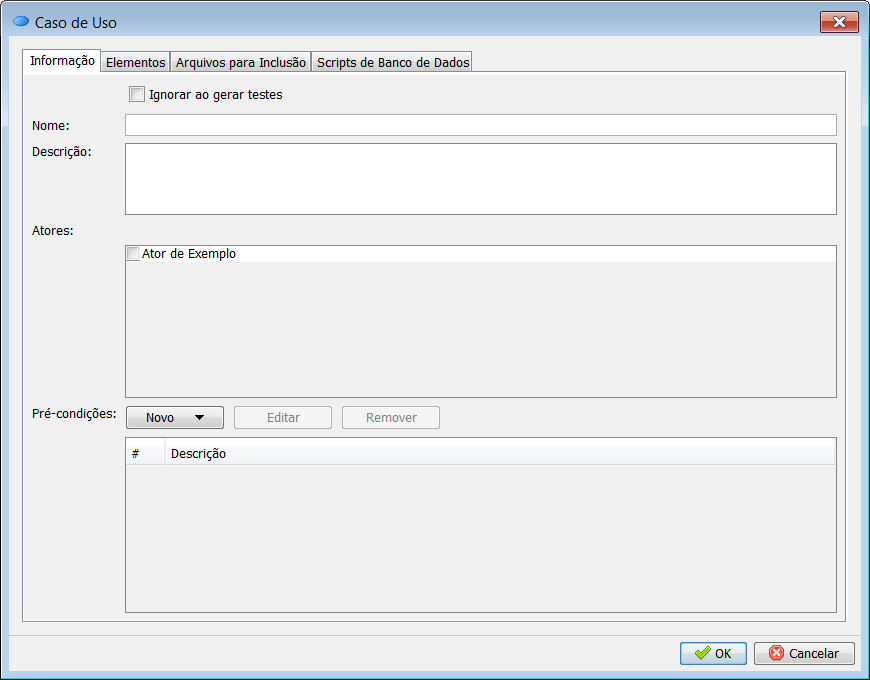


Informe o Nome e a Descrição do Ator e clique em OK. Após cria-lo, você pode ver o Ator na aba Atores:



## Criando um Caso de Uso

Para criar um Caso de Uso, acesse o menu Software 🡪 Novo Caso de Uso, ou pressione Ctrl + U.



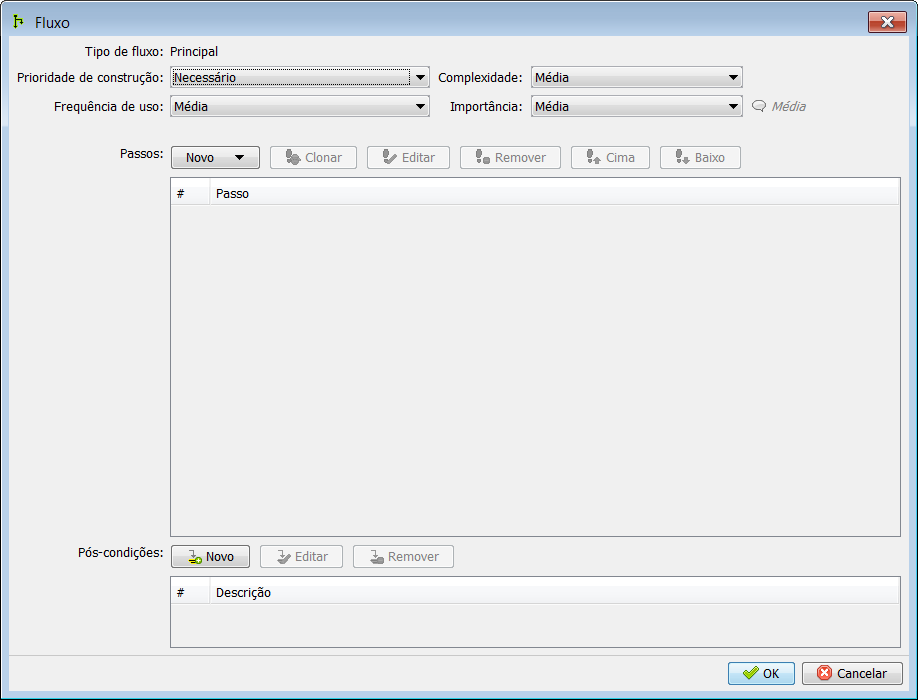
Informe o Nome e a Descrição, selecione o(s) Ator(es) que participa(m) do caso de uso e informe as possíveis Pré-condições existentes para que o caso de uso possa ser executado. É comum que a pós-condição de um caso de uso sirva como pré-condição outro um caso de uso. Quando você tiver esse caso, cadastre a Pós-condição pelo menu Novo 🡪 Nova Pré-condição de uma Pós-condição. Do contrário, cadastre-a pelo menu Novo 🡪 Nova Pré-condição.

## Criando Fluxos em um Caso de Uso

Um fluxo é a descrição da interação de um ator com o caso de uso, através de uma série de passos. Um caso de uso possui um Fluxo Principal (ou Fluxo Básico) e, possivelmente, um ou mais Fluxos Alternativos. O Fluxo Principal é o caminho de execução mais frequente, enquanto os Fluxos Alternativos são outras possibilidades de execução a partir de o Fluxo Principal ou de outros Fluxos Alternativos.

### Criando um Fluxo Principal

O primeiro fluxo a ser criado no caso de uso é o **Fluxo Principal**. Para fazer isso, selecione o caso de uso e acesse o botão Novo e a opção Novo Fluxo Principal.



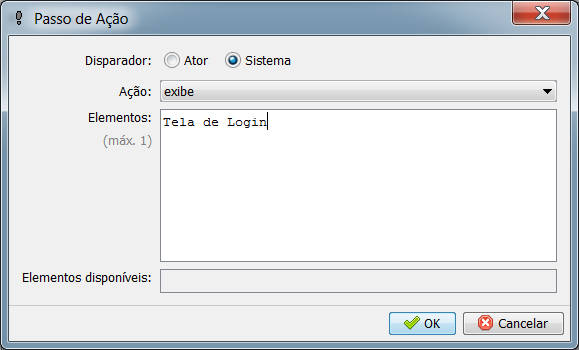
Escolha a Prioridade de construção, a Complexidade, a Frequência de uso e a Importância do fluxo. Os três primeiros valores são usados para calcular a **sugestão de importância** do fluxo, exibida ao lado de um pequeno balão (). O valor da Importância de um fluxo é usado no calculo da importância de um **cenário de execução**, [[5]](#footnote-5) que por sua vez é usado para calcular a importância de um **caso de teste**. Assim, cada **teste abstrato** gerado possui um valor de **importância**, que pode ser usado pelos *plug-ins* que os transformam em **script de teste** (código-fonte), embutindo-os no código para **categorizar** os testes. Então, ao executar os testes, pode-se parametrizar qual categoria de testes – ou seja, qual *importância* dos testes – se deseja executar.[[6]](#footnote-6)

Para criar passos para o fluxo, clique no botão Novo e escolha um dos tipos de passo disponíveis:

* Passo de Ação: É um passo realizado por um Ator ou pelo Sistema em que uma ação é executada. Esta ação deve ser verificável, isto é, deve poder ser observada através de um *teste funcional de interface de usuário*. Por exemplo, ao definir o passo "*Sistema exibe a Janela de Cadastro de Usuário*", esta ação de "exibir a janela" deve ser verificável. As ações disponíveis para um Passo de Ação podem variar de acordo com o Vocabulário escolhido ao criar o Software (veja seção 0);
* Passo de Documentação: É um passo realizado por um Ator ou pelo Sistema em que a ação executada é descrita de forma livre, sem o intuito que seja verificável através de testes funcionais de interface de usuário;
* Passo de Chamada a Caso de Uso: É um passo realizado pelo Sistema, em que um caso de uso anteriormente especificado é chamado;
* Passo de Oráculo: Na versão atual, é um passo realizado pelo Sistema para verificar as regras de negócio de elementos de interface de usuário (*widgets*) editáveis, isto é, que podem receber entrada de dados do usuário.[[7]](#footnote-7)

#### Passo de Ação

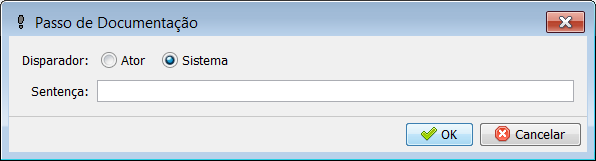
Ao criar um Passo de Ação, escolha o Disparador da ação (Ator ou Sistema), a Ação, informe um ou mais Elementos de interface que serão os alvos da Ação, **separando-os com vírgula**. Então, clique em OK para concluir.



Fornecer vários Elementos para uma Ação em um Passo pode ser mais produtivo que criar vários Passos. Enquanto algumas ações admitem somente um elemento (como "exibe"), outras admitem mais de um elemento (como "digita" e "clica"). Isto dependerá do Perfil utilizado pelo Vocabulário, que é quem determina o número máximo de Elementos aceitos por uma Ação. Você será avisado caso tente fornecer um número de Elementos maior que o permitido em uma Ação.

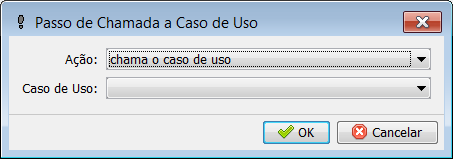
#### Passo de Documentação

Ao criar um Passo de Documentação, escolha o Disparador da ação (Ator ou Sistema), informe (livremente) a Sentença de o que esse passo realiza e clique em OK.



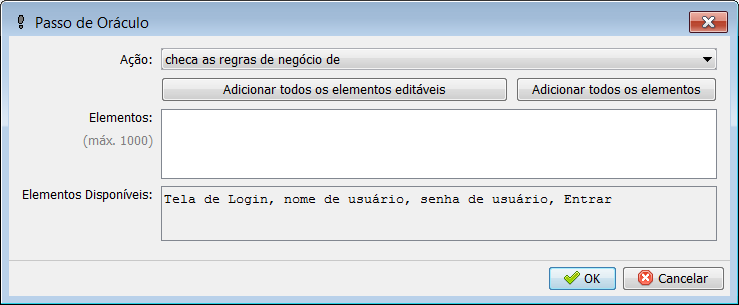
#### Passo de Chamada a Caso de Uso

Ao criar um Passo de Chamada a Caso de Uso, defina a Ação, escolha o Caso de Uso a ser chamado e clique em OK.



#### Passo de Oráculo

Ao criar um Passo de Oráculo, define a Ação, informe os Elementos separando-os por vírgula e clique em OK.



É comum que os Elementos informados nesse tipo de passo sejam **editáveis**, isto é, que possam receber valores informados pelo ator (usuário). Logo, para não precisar digitá-los, simplesmente clique no botão Adicionar todos os elementos editáveis. No exemplo acima, que se refere a um caso de uso de *login* (Efetuar Login), os elementos editáveis são "nome de usuário" e "senha de usuário". Logo, é possível digitá-los (ação que possui auxílio do recurso auto-completar) ou adicioná-los diretamente através desse botão.

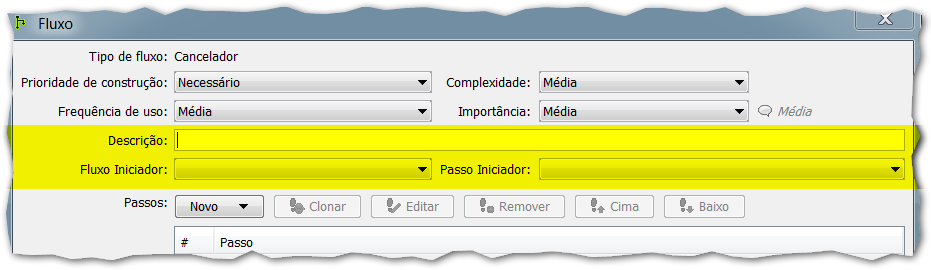
Após criar os passos, é possível cloná-los (), alterá-los (), removê-los () e movê-los para cima () e para baixo () na lista.

### Criando um Fluxo Alternativo

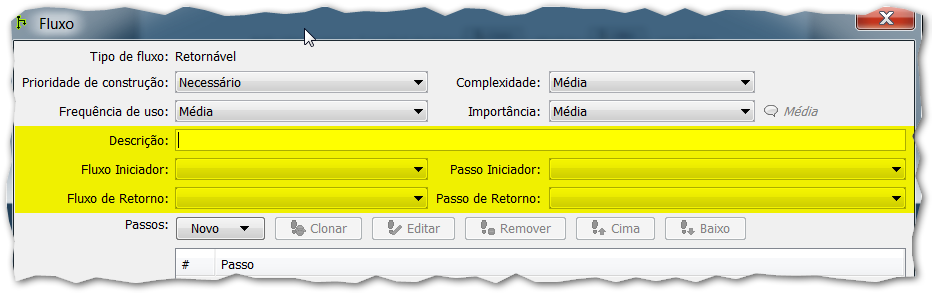
Após criar um Fluxo Principal, você poderá criar um Fluxo Alternativo, escolhendo um dos tipos disponíveis:

* **Fluxo Terminador**: Fluxo alternativo que leva ao término do caso de uso com sucesso, possivelmente gerando uma ou mais pós-condições;
* **Fluxo Retornável**: Fluxo alternativo em que o fluxo de execução do caso de uso retorna para o caso de uso ao qual teve início, após executar alguns passos;
* **Fluxo Cancelador**: Fluxo alternativo em que a execução do caso de uso é cancelada, possivelmente não gerando pós-condições.

Ao criar um **Fluxo Terminador** ou um **Fluxo Cancelador**, há, adicionalmente aos dados do Fluxo Principal, a Descrição do fluxo, o Fluxo Iniciador – que é a indicação do fluxo do qual o fluxo atual é disparado – e o Passo Iniciador – que é o passo do Fluxo Iniciador em que o fluxo atual é disparado.

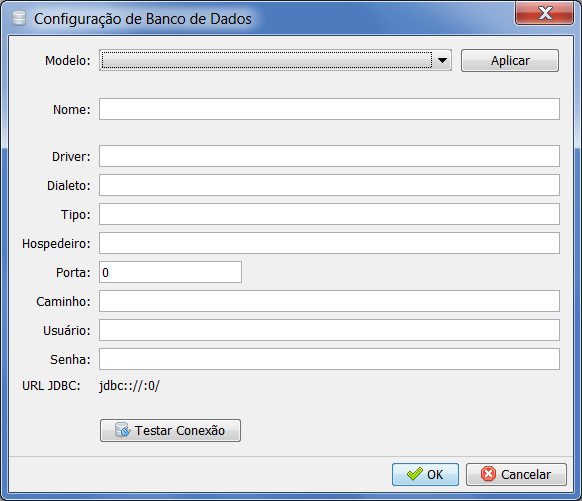


Ao criar um **Fluxo Retornável**, há, adicionalmente a um **Fluxo Terminador** ou a um **Fluxo Cancelador**, o Fluxo de Retorno – que é o fluxo para o qual o fluxo atual retorna – e o Passo de Retorno – que é o passo em que o fluxo atual é retorna.

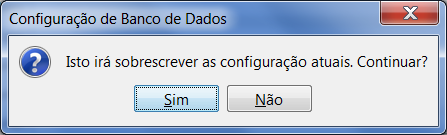


## Criando Conexões e Consultas a Bancos de Dados

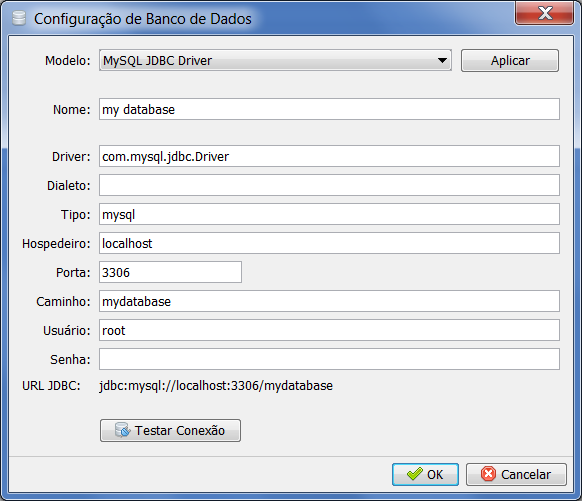
Para configurar uma conexão com um banco de dados, acesse a aba Bancos de Dados & Consultas e clique no botão Novo ().



Escolha um dos modelos de banco de dados disponíveis e tente aplicá-lo, através do botão Aplicar. Será apresentada a mensagem abaixo. Ao clicar em Sim, as configurações atuais serão sobrescritas com as configurações padrão do modelo de banco de dados escolhido.



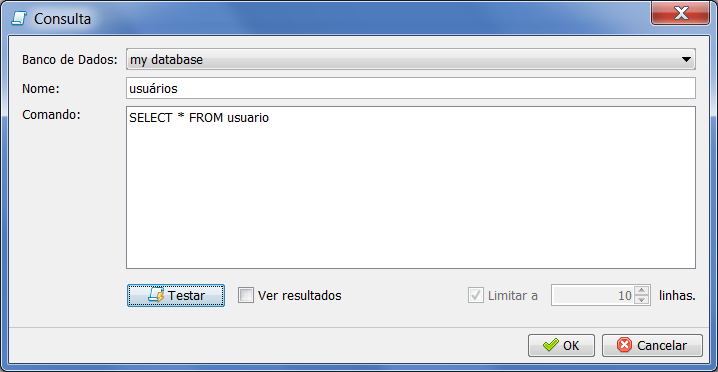
Então, preencha os campos para configurar a conexão. A figura abaixo mostra um exemplo de configuração para um banco de dados MySQL chamado "my database":



O botão Testar Conexão () permite verificar se a conexão com o banco de dados pode ser estabelecida. Após finalizar a configuração, clique em OK para confirmá-la.

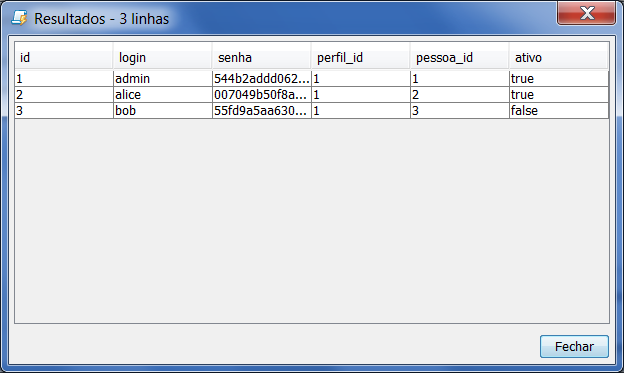
### Criando Consultas

Para criar uma consulta, clique no botão Novo (), localizado no centro da tela. Escolha o Banco de Dados, informe o Nome e o Comando. A figura abaixo mostra um exemplo de consulta.



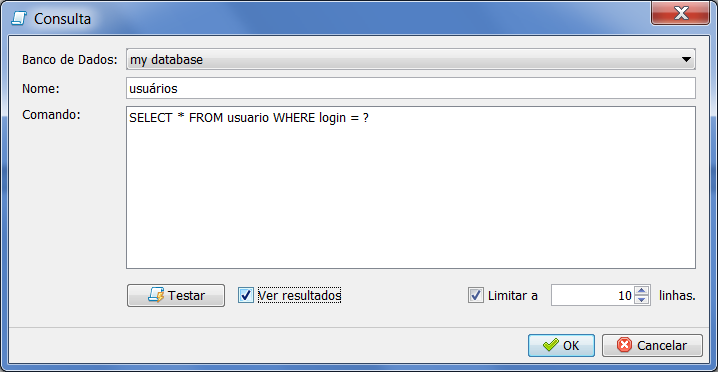
Clique em Testar para verificar se a consulta está correta. Caro queira ver os resultados da consulta, marque a opção Ver resultados e, então, clique em Testar. Por padrão, uma consulta é limitada em 10 linhas (registros). Você pode remover esse limite desmarcando a opção Limitar a ou modificar o valor limite, informando o respectivo número de linhas.

A figura abaixo mostra um exemplo de visualização dos resultados da consulta exemplificada anteriormente:

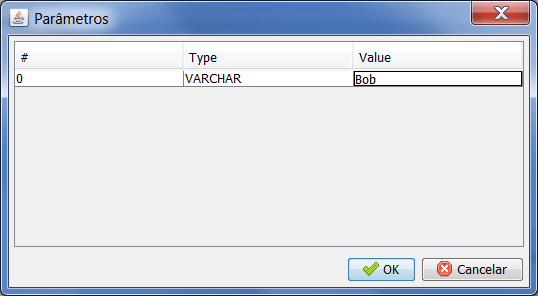


#### Especificando Parâmetros numa Consulta

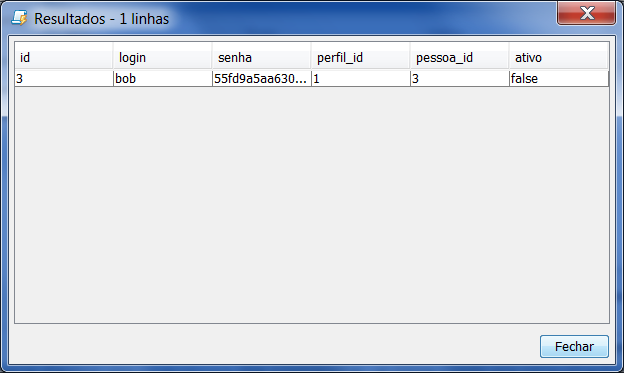
Ao cadastrar uma consulta, você pode usar **parâmetros anônimos**, isto é, parâmetros que não são nomeados. Esses parâmetros devem ser representados pelo símbolo de interrogação (?). Ao testar este tipo de consulta, usando o botão Testar, será solicitado que você informe o valor de cada parâmetro. Por exemplo, considere a consulta abaixo, que selecione registros da tabela "usuario" contendo um "login" fornecido no momento de sua execução, e cujo comando em SQL é "SELECT \* FROM usuario WHERE login = ?":



Ao clicar em Testar, será exibida uma tela solicitando o valor do parâmetro da consulta:



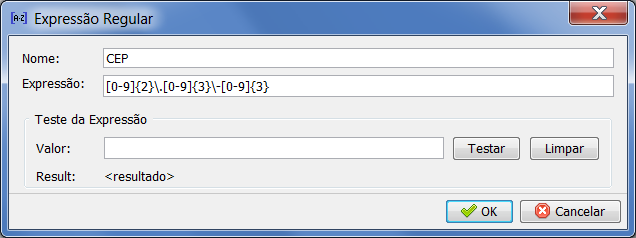
Ao informar o valor respectivo ao parâmetro e clicar em OK, a consulta será avaliada. Caso tenha marcado para exibir os registros da consulta, a tela com os resultados da consulta será exibida:



Ao final, para confirmar a criação da consulta (na tela de criação da Consulta), clique em OK.

## Criando Expressões Regulares

Para criar uma expressão regular, acesse a aba Expressões Regulares e clique no botão Novo (). Informe o Nome e a Expressão e, caso queira testá-la, clique no botão Testar.



Para confirmar a criação da expressão regular, clique em OK.

## Detalhando Elementos e suas Regras de Negócio

TO-DO

## Gerando e Executando Testes

TO-DO

1. http://json.org/ [↑](#footnote-ref-1)
2. http://subversion.apache.org/ [↑](#footnote-ref-2)
3. http://git-scm.com/ [↑](#footnote-ref-3)
4. http://mercurial.selenic.com/ [↑](#footnote-ref-4)
5. Um cenário de execução é um caminho realizado por um ator ao interagir com o caso de uso, podendo ser composto de um ou mais fluxos (sempre participando o fluxo principal) [↑](#footnote-ref-5)
6. Essa parametrização depende do framework de testes utilizado e pode ser fornecida na linha de comando que executa os testes. [↑](#footnote-ref-6)
7. Futuramente um Passo de Oráculo poderá também verificar propriedades de *widgets*, sendo mais flexível como oráculo de interface de usuário. [↑](#footnote-ref-7)