### Resource Action

Người viết: Phạm Văn Nam

Mục lục

[1. Định nghĩa resource và action 2](#_Toc294039132)

[1.1. Định nghĩa Resource: 2](#_Toc294039133)

[1.2. Định nghĩa Permission: 2](#_Toc294039134)

[2. Tạo resource và action 2](#_Toc294039135)

[2.1. Cấu hình resource-actions 2](#_Toc294039136)

[2.2. Xây dựng class kiểm tra permission trên resource 7](#_Toc294039137)

[2.3. Xây dựng các phương thức cần kiểm tra quyền trong class {Entity}ServiceImpl 10](#_Toc294039138)

[3. Phân loại resource 13](#_Toc294039139)

Định nghĩa resource và action

* 1. Định nghĩa Resource:

Resource là một thuật ngữ chung cho bất kỳ đối tượng nào được đại diện trong Portal.

Ví dụ: Các resource

* Porlet: Message Boards, Calendar, v.v.
* Java classes: Message Board Topics, Calendar Events, v.v.
* File: documents, images, v.v.
  1. Định nghĩa Permission:

Là hành động tác động lên tài nguyên (resource).

Ví dụ như: Xem porlet calendar cũng được xem như là một quyền (permission).

Tạo resource và action

Trong phần này sẽ hướng dẫn bạn cấu hình resource-actions phân quyền cho ứng dụng quản lý kho hàng dưới dạng portlet trên Liferay Portal với các chức năng thêm, xóa , sửa.

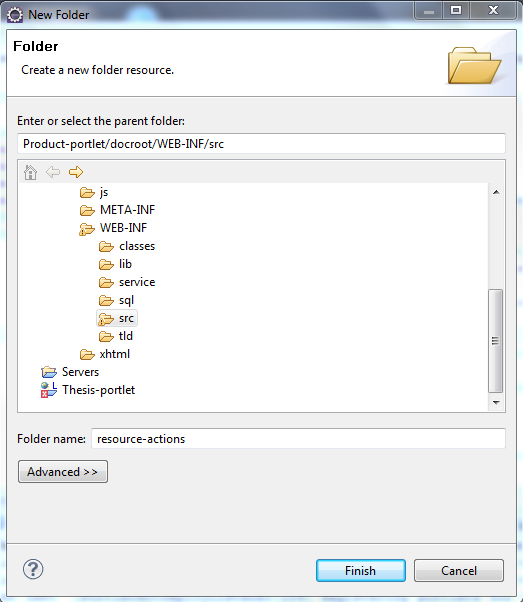
* 1. Cấu hình resource-actions

Mặc định liferay hỗ trợ 2 quyền view và configuration permission nên khi cấu hình resource-actions sẽ không cấu hình 2 quyền cho porlet.

Sau đây là các bước cấu hình resource-actions.

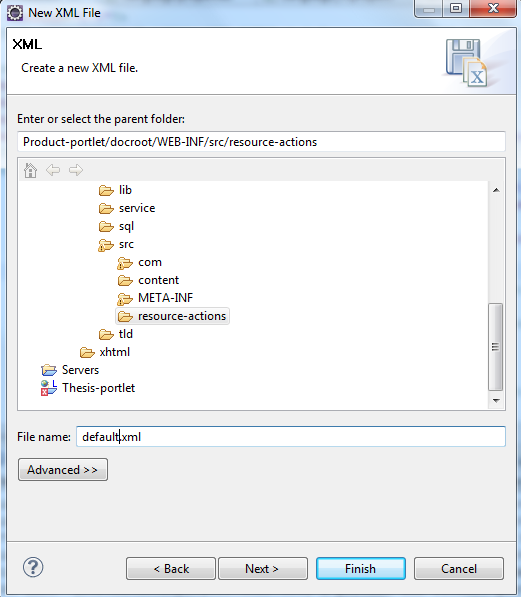
* Bước 1: Tạo folder resource-actions trong /docroot/WEB-INF/src/

+ Thư mục resource-actions là thư mục chưa các file cấu hình vê quyên cho portlet



Đường dẫn đến thư mục resource-action

* Bước 2: Tạo file default.xml trong thư mục resource-actions.



Đường dẫn đến file default.xml

+ default.xml là file lưu đường dẫn đến các file cấu hình quyền hạn cho portlet

+ Nội dung file default.xml.

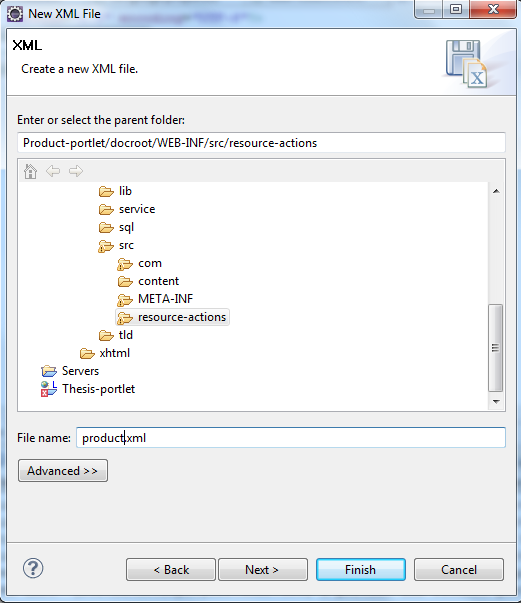
<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<resource-action-mapping>

<resource file=*"resource-actions/product.xml"* />

</resource-action-mapping>

* Bước 3: Tạo file product.xml trong thư mục resource-actions



Đường dẫn đến file product.xml

+ Product.xml là file định nghĩa các từ khóa tương ứng với các chức năng cho portlet.

+ Các thẻ sử dụng:

<supports>: Định nghĩa các từ khóa tương ứng các chức năng cho portlet.

<guest-defaults>: Gán quyền mặc định cho guest.

<guest-unsupported>: Các quyền hạn không hỗ trợ cho guest.

<owner-defaults>: Quyền hạn mặc định cho owner.

+ Nội dung file product.xml.

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<resource-action-mapping>

<portlet-resource>

<portlet-name>Product</portlet-name>

<permissions>

<supports>

<action-key>ADD\_TO\_PAGE</action-key>

<action-key>PREFERENCES</action-key>

<action-key>CONFIGURATION</action-key>

<action-key>VIEW</action-key>

</supports>

<community-defaults>

<action-key>VIEW</action-key>

</community-defaults>

<guest-defaults>

<action-key>VIEW</action-key>

</guest-defaults>

<guest-unsupported>

<action-key>ADD\_TO\_PAGE</action-key>

<action-key>PREFERENCES</action-key>

<action-key>CONFIGURATION</action-key>

<action-key>VIEW</action-key>

</guest-unsupported>

</permissions>

</portlet-resource>

<model-resource>

<model-name>com.nss.portlet.product</model-name>

<portlet-ref>

<portlet-name>Product</portlet-name>

</portlet-ref>

<permissions>

<supports>

<action-key>ADD\_PRODUCT</action-key>

<action-key>DELETE</action-key>

<action-key>PERMISSIONS</action-key>

<action-key>UPDATE\_PRODUCT</action-key>

</supports>

<community-defaults />

<guest-defaults />

<guest-unsupported>

<action-key>ADD\_PRODUCT</action-key>

<action-key>DELETE</action-key>

<action-key>PERMISSIONS</action-key>

<action-key>UPDATE\_PRODUCT</action-key>

</guest-unsupported>

<owner-defaults>

<action-key>DELETE</action-key>

<action-key>PERMISSIONS</action-key>

<action-key>UPDATE\_PRODUCT</action-key>

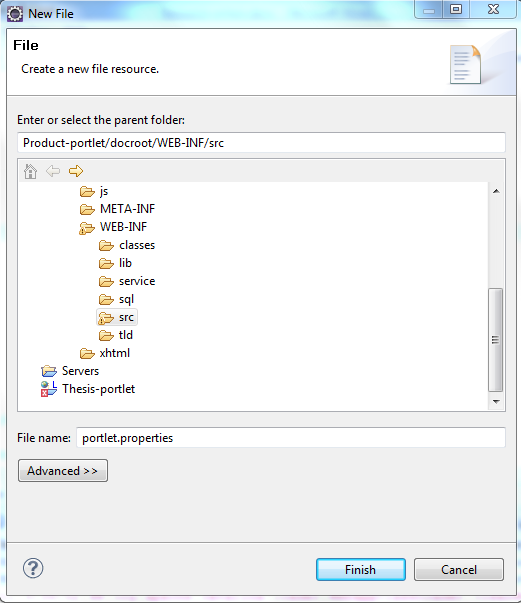
</owner-defaults>

</permissions>

</model-resource>

</resource-action-mapping>

* Bước 4: Tạo file portlet.properties trong /src



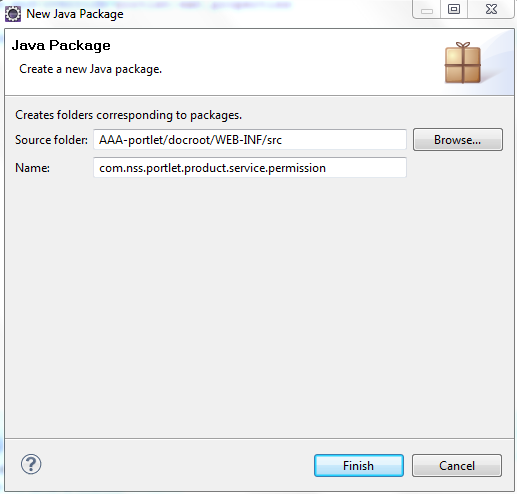
Đường dẫn đến file portlet.properties

+ Nội dung file portlet.properties.

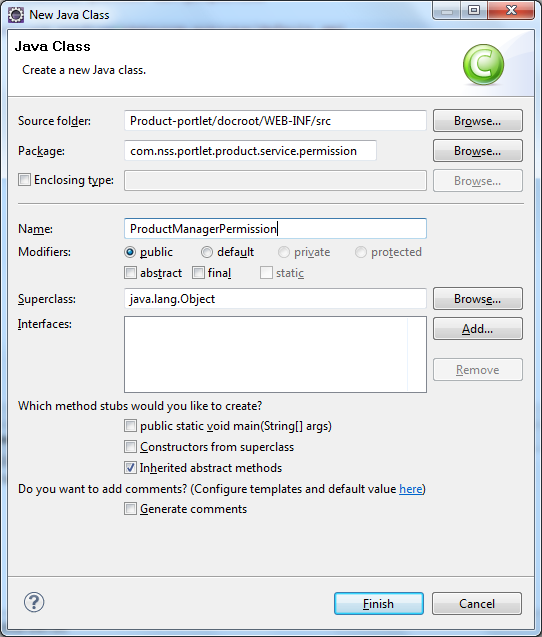
include-and-override=portlet-ext.properties

resource.actions.configs=resource-actions/default.xml

* 1. Xây dựng class kiểm tra permission trên resource
* Bước 1: Tao package com.nss.portlet.product.service.permission



* Bước 2: Tao lớp ProductManagerPermission trong package com.nss.portlet.product.service.permission



+ Sử dụng đối tượng PermissionChecker để kiểm tra quyền truy xuất của người dùng

+ Nội dung lớp ProductManagerPermission.

**package** com.nss.portlet.product.service.permission;

**import** com.liferay.portal.kernel.exception.PortalException;

**import** com.liferay.portal.security.auth.PrincipalException;

**import** com.liferay.portal.security.permission.PermissionChecker;

**public** **class** ProductManagerPermission {

**private** **static** **final** String *\_CLASS\_NAME* =

"com.nss.portlet.product";

**public** **static** **void** check(

PermissionChecker permissionChecker, **long**

groupId, String actionId)

**throws** PortalException {

**if** (!*contains*(permissionChecker, groupId,

actionId)) {

**throw** **new** PrincipalException();

}

}

**public** **static** **boolean** contains(

PermissionChecker permissionChecker, **long**

groupId, String actionId) {

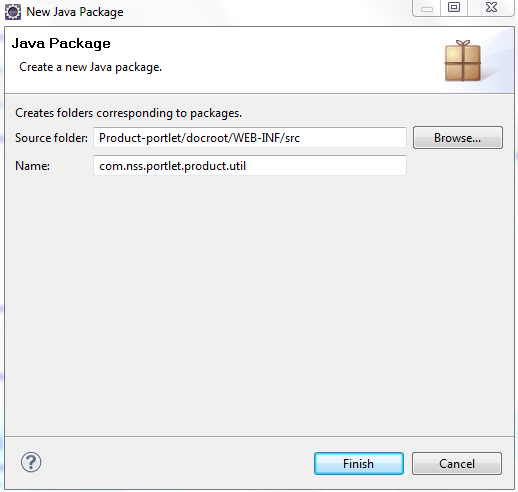
**return** permissionChecker.hasPermission(

groupId, *\_CLASS\_NAME*, groupId, actionId);

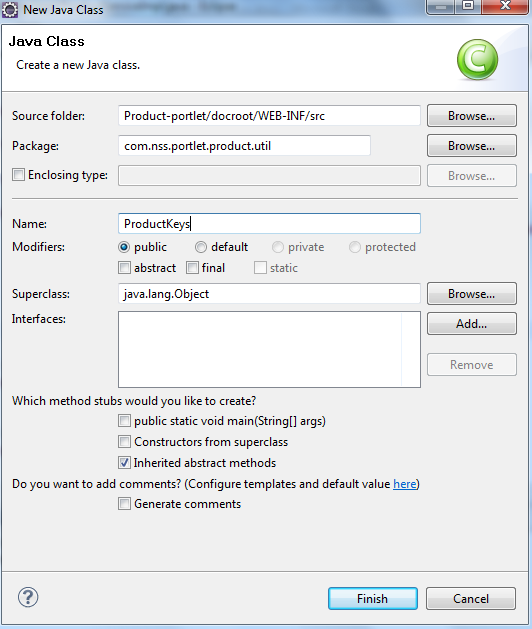
}

}

* 1. Xây dựng các phương thức cần kiểm tra quyền trong class {Entity}ServiceImpl
* Bước 1: Tạo package com.nss.portlet.product.util



* Bươc 2: Tạo lớp ProductKeys.java



+ Định nghĩa các từ khóa tương ứng các chức năng của portlet.

+ Nội dung lớp ProductKeys.java

**package** com.nss.portlet.product.util;

**public** **interface** ProductKeys {

**public** **static** **final** String *ADD\_PRODUCT* = "ADD\_PRODUCT";

**public** **static** **final** String *DELETE* = "DELETE";

**public** **static** **final** String *PERMISSIONS* = "PERMISSIONS";

**public** **static** **final** String *PORTLET\_ID* = "sample";

**public** **static** **final** String *UPDATE\_PRODUCT* =

"UPDATE\_PRODUCT";

}

* Bước 3: Chỉnh sửa nội dung lớp ProductServiceImpl.java.

Sử dụng ProductManagerPermission dể kiểm tra quyền cho các phương thức

addProduct, delete, updateProduct tương ứng với các quyền của portlet.

**package** com.nss.portlet.product.service.impl;

**import** com.liferay.portal.kernel.exception.PortalException;

**import** com.liferay.portal.kernel.exception.SystemException;

**import** com.liferay.portal.security.permission.ActionKeys;

**import** com.liferay.portal.service.permission.PortletPermissionUtil;

**import** com.nss.portlet.product.model.Product;

**import** com.nss.portlet.product.service.base.ProductServiceBaseImpl;

**import** com.nss.portlet.product.service.permission.ProductManagerPermission;

**import** com.nss.portlet.product.util.ProductKeys;

**public** **class** ProductServiceImpl **extends** ProductServiceBaseImpl {

**public** Product addProduct(

String productName, String productCode, String description, **double** price)**throws** SystemException, PortalException {

System.*out*.println(getUserId());

ProductManagerPermission.*check*(

getPermissionChecker(), getUser().getGroup().getGroupId(),

ProductKeys.*ADD\_PRODUCT*);

**return** productLocalService.addProduct(

productName, productCode, description, price);

}

**public** **void** delete(**long** productId)

**throws** SystemException, PortalException {

productLocalService.deleteProduct(productId);

}

**public** Product getProduct(**long** productId)

**throws** PortalException, SystemException {

PortletPermissionUtil.*check*(

getPermissionChecker(), ProductKeys.*PORTLET\_ID*, ActionKeys.*VIEW*);

**return** productLocalService.getProduct(productId);

}

**public** Product updateProduct(

**long** productId, String productName, String productCode,

String description, **double** price)

**throws** SystemException, PortalException {

ProductManagerPermission.*check*(

getPermissionChecker(), getUser().getGroup().getGroupId(),

ProductKeys.*UPDATE\_PRODUCT*);

**return** productLocalService.updateProduct(

productId, productName, productCode, description, price);

}

}

Phân loại resource

Resource có 2 loại: Portlet resource và Model resource

* Portlet resource sử dụng phân quyền cho toàn bộ portlet.
* Model resource sử dụng phân quyền cho các thành phân bên trong portlet như package, class

Permission checker sử dụng kiểm tra quyền truy suất của người dụng vào trong hệ thống.