
Table of Contents

Foreword	ii
1. About, Prereqs, and Dependencies	3
1.1. Prerequisites	3
1.2. Dependencies	3
Compile/Test Time	3
RunTime	4
2. Configuration, Compilation, and Deployment	5
2.1. Configuration	5
2.2. Compilation	5
2.3. Deploying the portlet to Tomcat	5
Installing the Portlet (uPortal only)	5
3. Customizing the Portlet	7
3.1. CSS	7
3.2. JSP	8

Foreword

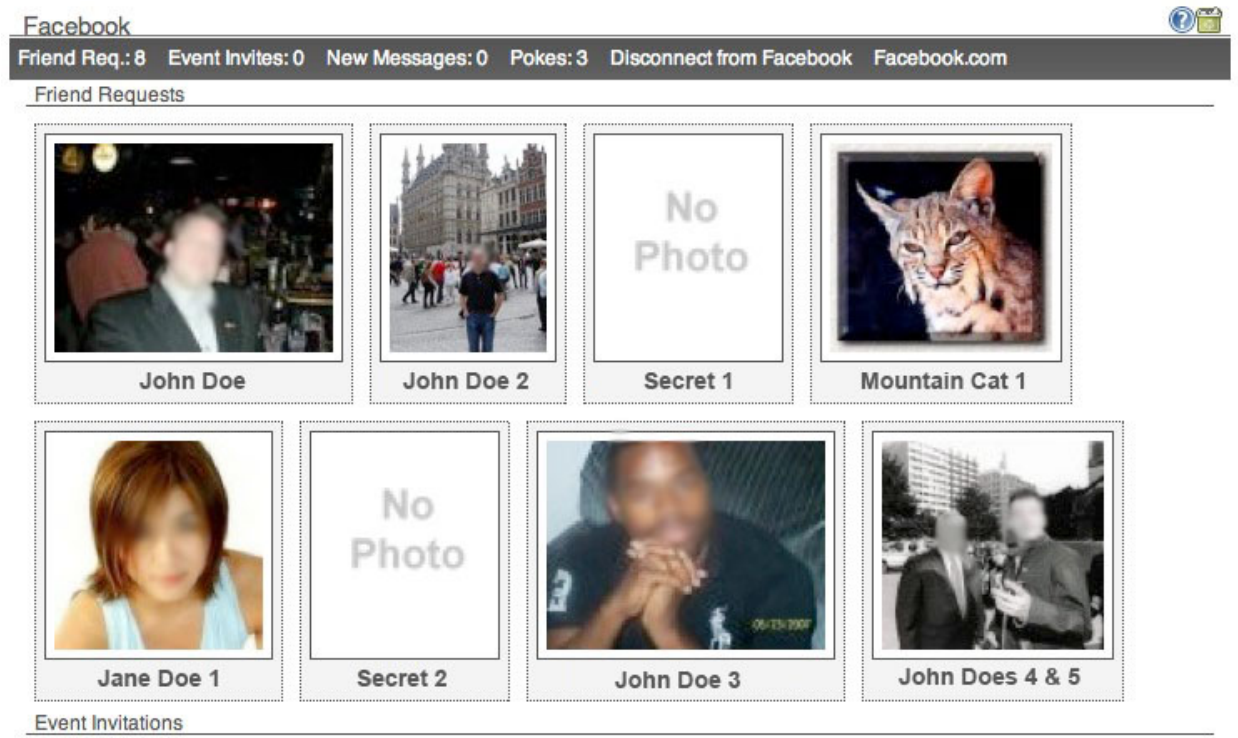


The Internet Frameworks Services division of Duke University's Office Of Information Technology (OIT) has created a number of portlets used in the student portal, DukePass. We have released the source code for these portlets under the JA-SIG License [<http://www.uportal.org/license.html>].

We encourage anyone with an interest to download, install, and use these portlets. Our portal installation uses uPortal version 2.5.3, but we have tried wherever possible to create JSR-168 compliant portlets that should run in any portlet container.

1. About, Prereqs, and Dependencies

The SocialNet Portlet currently contains Facebook [<http://www.facebook.com>] integration, but we intend to expand use to other Social Networking sites as their APIs and usage policies permit.



1.1. Prerequisites

Before you begin, you should know how to deploy portlets into your portal environment. We developed and tested this portlet with uPortal v2.5.3, but it may work with other version of uPortal. If you discover a shortcoming in our code that we can correct to make the portlet more JSR-168 compliant, please let us know.

You will need to have ant and a Java 1.5+ compiler installed.

1.2. Dependencies

Where permissible, all dependencies are included in the /dep directory. These are separated into those dependencies required for runtime, and those required only for building (such as unit testing). Dependant libraries are listed below.

Compile/Test Time

- JUnit 3.x
- Servlet API

- Portlet API
- Spring Mock Objects

RunTime

- Spring Framework 2.x
- Spring Portlet Support
- Commons Logging
- Log4j

2. Configuration, Compilation, and Deployment

2.1. Configuration

Unzip the distribution .zip file to a working directory that will be used to build and configure the portlet. Change to the working directory. All paths referenced below are relative to the working directory.

At minimum, you must obtain an API key and a Secret Key from <http://developer.facebook.com> [<http://developer.facebook.com>]. These should be installed in `/src/facebook.properties`

This portlet has a Spring-based configuration, so you should be able to change all relevant parameters in text files rather than changing the source code. However, you are welcome to change the source code to suit your needs if you are comfortable doing so.

Facebook requires you to enter the list of IP addresses that are valid for using your credentials. Remember to include the IP addresses of the machines running unit tests, or they will fail because the connection will be denied.

There is no need to modify `/resources/portlet/socialnetportlet.xml`, the Spring configuration file, to make the portlet work. This is, however, the right place to swap in other view technologies or replace controllers or services where appropriate.

2.2. Compilation

Once the configuration is complete, you're ready to compile and create a deployable version of the portlet. By default, the portlet will be compiled in a format readily deployable for uPortal, a pluto-based portlet container. Alternate deployments will require modification of the `portlet.xml` file, the `web.xml` file, and possibly the `build.xml` file.

```
ant build
```

This will compile the application, run the unit tests, and generate a .war file that is ready to go into your portal environment.

If the build succeeds, you will find the .war file in `/dist/socialnetportlet.war`

2.3. Deploying the portlet to Tomcat

Deploying the portlet only requires copying the portlet .war file to your servlet's webapp directory.

```
cp dist/socialnetportlet.war $TOMCAT_HOME/webapps
```

Installing the Portlet (uPortal only)

Every portlet container has a different method for installing a new portlet once it is deployed as a web application. Instructions are included here for installing the portlet for use with uPortal.

3. Customizing the Portlet

3.1. CSS

This section details the CSS styles used to lay out the portlet. You'll need to change these to match the look and feel of your portal's themes and skins. As the standards evolve in CSS styling for portlets, we'll be converting these specific names to more general CSS classes as defined by the specs.

Here is the section of CSS we use to style the portlet at Duke. We chose to include all of our CSS in one .css file that's downloaded as a part of the portal page's template rather than distribute a .css file with each portlet.

```
div.socialNetwork {}
div.socialNetwork div.stats {color:#FFF; padding:3px; background:#036 url(../slices
div.socialNetwork div.stats ul {margin:0; padding:0; list-style:none;}
div.socialNetwork div.stats ul li {display:inline; margin:0; padding:0 8px 0 2px;
div.socialNetwork div.stats ul li span.count {font-weight:bold;}
div.socialNetwork div.stats ul li span.friendReqs {}
div.socialNetwork div.stats ul li span.friendReqs span.count {}
div.socialNetwork div.stats ul li span.invites {}
div.socialNetwork div.stats ul li span.invites span.count {}
div.socialNetwork div.stats ul li span.unread {}
div.socialNetwork div.stats ul li span.unread span.count {}
div.socialNetwork div.stats ul li span.pokes {}
div.socialNetwork div.stats ul li span.pokes span.count {}
div.socialNetwork div.stats ul li a {color:#FFF; margin:0; padding:0;}
div.socialNetwork div.stats span#faceBookMore {}
div.socialNetwork div.stats span#faceBookMore a {display:block; color:#FFF;}
div#ssExtendedContent {clear:both;}

div.snContactRequests {margin:1px 0 0 0; clear:both; padding:0 10px;}
div.snContactRequests span.snContactRequestsTitle {display:block; color:#036; text
div.snContactRequests div.contact {float:left;}
div.snContactRequests div.contact a {display:block; margin:5px; padding:5px 5px 0 !
div.snContactRequests div.contact a:hover {background-color:#C9D4EA;}
div.snContactRequests div.contact a img {height:125px; padding:5px; border:1px sol
div.snContactRequests div.contact a span.snContactName {display:block; clear:both;

div.snEventInvites {margin:1px 0 0 0; clear:both; padding:0 10px;}
div.snEventInvites span.snEventInvitesTitle {display:block; color:#036; text-indent
div.snEventInvites div.invite {float:left;}
div.snEventInvites div.invite a {display:block; min-width:270px; margin:5px; paddin
div.snEventInvites div.invite a:hover {background-color:#C9D4EA;}
div.snEventInvites div.invite a img {float:left; height:100px; margin: 0 5px 0 0; p
div.snEventInvites div.invite a span.snInviteDescription {display:block; float:left
div.snEventInvites div.invite a span.snEventTitle {display:block; font-weight:bold
div.snEventInvites div.invite a span.snEventHost {display:block; line-height:1.3;}
div.snEventInvites div.invite a span.snEventDate {display:block; line-height:1.3;}
div.snLinks {clear:both; margin:1px 0 0 0;}
div.snLinks span.snLogoutLink {}
```

```
div.snLinks span.snLogoutLink a {}
div.snLinks span.snLogoutLink a:hover {}

div.socialNetwork span.snLogoutLink {}
div.socialNetwork span.snLoginLink a {display:block; padding:.5em .5em; color:#FF

div.socialNetwork span.snLogoutLink2 a {display:block; padding:.5em .5em; color:#FF
div.socialNetwork span.snLogoutLink span#facebookLogoutInfo {display:block; clear:l
```

Example 3.1 SocialNet CSS

3.2. JSP

We've used CSS strategies that will flow wherever possible, however, our JSP assumes a fairly wide layout. Therefore you may want to re-tool how the data are laid out on your page. We've used Spring WebMVC with standard JSTL tags. There are, however, text strings embedded in the JSP. We'll be working to extract these strings to make internationalization a more straightforward procedure.