Application Installation Guide

**Choose host server**

Our choice was to install in an Amazon Web Services (AWS) instance as host server. Along with that, we purchased a domain [www.rxeffex.com](http://www.rxeffex.com) and a trusted SSL certificate.

**Install and/or configure a web server/application server**

The application can run on any web server or application server. See the documentation for the particular web server/application server of your choice. We installed an Apache Tomcat application server on the AWS instance. One can easily download and install the application server from <http://tomcat.apache.org> .

**Install application files**

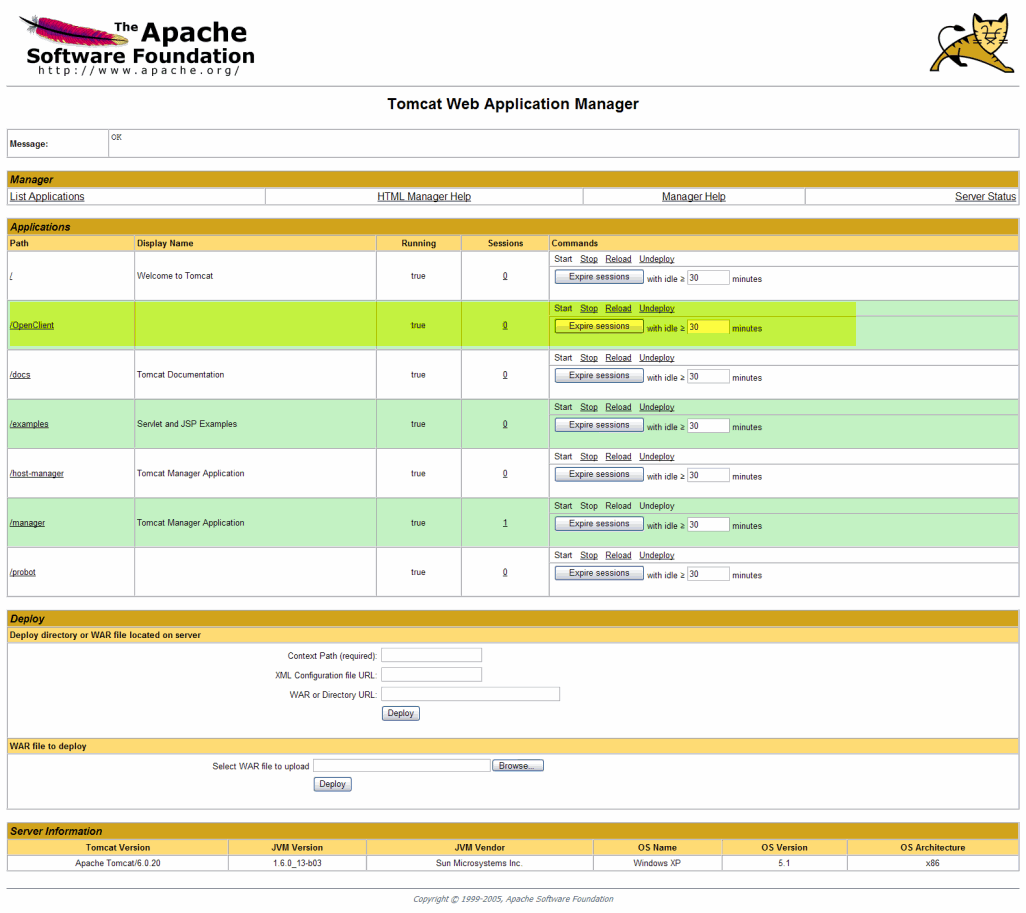
The application consists of basic HTML and JavaScript components. Use whatever means are appropriate to deploy the application files. Often times the application files can just as easily be copied to the appropriate web server directory.

However to perform web server deployments an application container is preferable. The method we used was web application archive (war) file. This was done to allow for future functionality that might be incorporated into the application. War file provides convenient means for installation in an application server.

To create a war file the application can be loaded into an integrated development environment (IDE) project such as Eclipse. See Eclipse documentation for the various methods projects can be imported.

One such method is to create a new Dynamic Web Project and load the contents of the 18f prototype project Git repository Web Content folder. The code is available at in the Git repository at the following address:

<https://github.com/amathur09/imc18f/tree/master/WebContent>

Then use a build tool such as Maven or the native Eclipse functionality to export the project as a war file. Then to deploy to the Apache Tomcat application server, one can perform a deployment through the Web Application Manager shown below. See Deploy section, enter war file location, and click “deploy”.

Tomcat in this example will provide the path to the deployed application. For custom path settings and other information, see Apache documentation for such any additional installation instructions.

**Configure to utilize SSL connection**

See the documentation with the web server/application server of your choice. As well as any instructions to set the appropriate port settings, certificate installation, and URL rewrite rules for the application SSL connection.

In our Tomcat installation we configured two ports for example in the server.xml:

<Connector port="80" … />

and

<Connector port="443" …/>

**Access the application and Test**

Use the url path you’ve configured to pull up the application in a browser of your choice.

The site should look like the image below. You can validate that the site represents the latest build by checking the build number in the footer of the main site page