NetUI Idea50

Creating an IntelliJ IDEA 5.0 Project using NetUI

This page describes the steps necessary to get a the netui-blank webapp running inside of IntelliJ IDEA 5.0. These steps are based upon the 1.0 release of Beehive and the documentation provided for that release. Currently, this points to the nightly documentation. These steps also assume that Tomcat is the target Servlet container. It should be simple to adopt these instruction for older versions of IDEA such as 4.5.

[NOTE: Once we do the 1.0 release, we need to point the links to the official 1.0 documentation.]

Create the Project

These steps are the preliminary steps that creates a new Beehive NetUI project. These steps are found in the documentation for NetUI.

1. First you must setup your environment. Complete all of the steps in the "Beehive Setup and Installation" topic.
2. Complete the sets to setup a new webapp. These instruction are found in the "NetUI Project Template" topic. With this step, do an **ant build** in the project root to confirm that the steps have worked. This will also create the **build** directory which is the expanded webapp that would be deployed to the server. At this point you should have the following three directories below the project root.
   - build – The directory that is the expanded webapp
   - src – The directory containing the source files
   - web – The directory containing the webapp content(.JSPs, Images, etc) plus the WEB-INF directory. **Note:** at the moment the root page flow controller **Controller.java** is found in this directory.

Create a new IDEA Project

This set of sets will create the IntellJ IDEA 5.0 project. These steps follow the pages of the wizard. A few of the page do not require modification, but are mentioned.

1. Open up IDEA and create a new project (File -> New Project).

2. The first page of the dialog should come up. Enter a name for the project and a place for the project file location. Then hit [Next].

3. Select a 1.5 JDK from your list of JDKs. Then hit [Next].

4. Select 'Create single-module project'. Then hit [Next].

5. Select the module type **Web Module**. Then hit [Next].

6. The module name should be the same as the project name. For the **Module content root**, select the root of your project. This is the directory that contains the **build**, **src**, and **web** directories along with the **build.xml** and **build.properties** files. **NOTE:** in the rest of this document, the project root is identified as **$project-root$** Then hit [Next].

7. Select an application server, in this case select "Tomcat". Then hit [Next].

8. On the deployment descriptor page, modify the **Web Module Deployment Descriptor** by selecting it and hitting the [Edit] button. First change where the Deployment descriptor is located. You should select the **$project-root$\web\WEB-INF\web.xml** deployment descriptor. Second change the **Deployment descriptor version** to **2.4**. Once you've changed the deployment descriptor hit [Next].

9. Set your **Web resource directory** to be **$project-root$\web**. Then hit [Next].

10. Set your **exploded directory** to be **$project-root$\build**. Then hit [Next].

11. Java Source is found in both the **src** and the **web** directories because the **Controller.java** is currently found in the web content. Hit [Next].

12. Compiler output should be directed to **$project-root\build\WEB-INF\classes**. Hit [Finish]. At this point your project has been created.

Project Compilation

In order to get your project to compile correctly and run inside IDEA, you need to do the following within the project. You won't actually compile the project using IDEA, instead you need to use the ant files. This step is necessary so that you don't recieve errors in your Java source as you develop page flows and supporting classes.

1. Open the project settings and select the **Libraries (Classpath)** tab.

2. Hit the **Add Jar/Directory...** button.
3. Add all of the .jar files from the **$project-root$/build/WEB-INF/lib** directory to the project.

4. Hit [OK] and exit the Project Settings page.

At this point, you should be able to build the project without any errors. You can't do an actual build from IDEA, but instead must use the ant commands found in the ant **build.xml** file.

Setup Ant For Build

The following steps will setup ant to run correctly inside your IDEA project.

1. Open the 'Ant Build' view and add [+] the **$project-root$/build.xml** file. You should see a set of commands including:

   - build
   - clean
2. Right click the ant file and pull up the Build File Properties dialog.

3. On the **Properties** tab, enter the following property (**name** = 'os.CATALINA_HOME', **value**='Tomcat-Location') where 'Tomcat-Location' is the root directory for you Tomcat installation. Hit [OK] on the dialog.

Building the Project

At this point you should be able to build the project using ant. To confirm this, run the ant **clean** command followed by the **build** command. The results may contains warning but shouldn't have any errors.

Running the Project

In order to run/debug in your new project do the following:

1. Create a new configuration [run -> edit configurations].

2. Select the **Tomcat Server** tab

3. Hit the [+] button and select 'local'

4. Give the configuration a **Name**.
5. Select the Tomcat Application Server
6. Uncheck the **Make Module before running/debugging/reloading** because we this project assume that the project is built using ant.

7. Press [OK] to save the new configuration.

At this point you should be able to run the project by hitting the [debug] or [run] buttons. When you run it, the server should start and a browser window will open showing you the default page flow.

Iterative Dev

In order to do iterative dev within the page flow, you do the following:

1. Make some modification to a page flow, for example add a new action.
2. Make a modification to a .JSP file, for example add a <newui:anchor> to call the action
3. Using ant, do another **build**
4. In the **Run - $Configuration$** window hit the **Deploy All** button which will deploy the changes to the server
5. In your browser window hit refresh. You should see the change to the page and if you press the link the action should run.