

Tapestry5RunViaMain

Create a Jetty Run Main Class

This method lets you create a class in your project that runs your application. This has a number of benefits

- Runs directly from compile output directory - so no waiting for copying files
- Works in all IDE's exactly the same
- Debugging/Profiling/Running work perfectly

This assumes you are running with Maven.

To make this work

1. Add a dependency on Jetty to your POM at provided scope. Use the jetty-server-dependencies and jetty-provided-apis artifacts
2. Turn off J2EE deployment in module properties
3. Add a class like the following to your src/test/resources

```
public class RunTapestryApp {

    public static void main(String[] args) throws Exception {
        Server server = new Server();

        Connector connector = new SelectChannelConnector();
        connector.setPort(Integer.getInteger("jetty.port", 8080));
        server.setConnectors(new Connector[]{connector});

        String jetty_home = System.getProperty("jetty.home", ".");

        WebAppContext webapp = new WebAppContext();
        webapp.setContextPath("/");
        webapp.setWar(jetty_home + "src/main/webapp");
        webapp.setDefaultsDescriptor(jetty_home + "src/test/resources/webdefault.xml");

        // Remove slf4j from list of classes not exposed to webapp
        webapp.setServerClasses(new String[] {"-org.mortbay.jetty.plus.jaas.", "org.mortbay.jetty."});

        server.setHandler(webapp);

        server.start();
        server.join();
    }
}
```

This might be a bad idea if you want to use Tapestry-test since it depends on Jetty 5 at present.