

# Tapestry5-TestWithTapestry-Spring

You need a new version of `org.apache.tapestry5.test.PageTester` which creates the [SpringContext](#) itself.

```
package mypackage.test;

import org.apache.tapestry5.test.PageTester;
import org.apache.tapestry5.internal.spring.SpringModuleDef;
import org.apache.tapestry5.ioc.def.ModuleDef;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class PageTesterSpring extends PageTester {

    public PageTesterSpring(String appPackage, String appName) {
        super(appPackage, appName);
    }

    public PageTesterSpring(String appPackage, String appName, String contextPath, Class... moduleClasses){
        super(appPackage, appName, contextPath, moduleClasses);
    }

    protected ModuleDef[] provideExtraModuleDefs() {
        ApplicationContext springContext = new ClassPathXmlApplicationContext(new String[]{"applicationContext.xml"});
        return new ModuleDef[]{ new SpringModuleDef(springContext) };
    }
}
```

Then you can use your [PageTesterSpring](#) just like the [PageTester](#):

```
package mypackage.test.frontend.pages;

import org.apache.tapestry5.dom.Document;
import mypackage.PageTesterSpring;
import org.testng.Assert;
import org.testng.annotations.Test;

public class Index extends Assert {

    @Test
    public void test1() {
        String appPackage = "mypackage.frontend";
        String appName = "app";
        PageTesterSpring tester = new PageTesterSpring(appPackage, appName, "WebContent");

        Document doc = tester.renderPage("Index");
        assertEquals(doc.getElementById("id1").getChildMarkup(), "hello");
    }
}
```