

## Module 17: Spring and Hibernate

### Exercise 17.2 – Spring Open Session in View

#### *The Setup:*

The goal of this exercise is to test Spring's open session in view support. Start by copying the **exercise-16-2** project, and add the spring context, tx and orm dependencies (the same Spring dependencies as in **Exercise-17-1**). Also add the following spring web dependency:

```
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-web</artifactId>
  <version>4.0.2.RELEASE</version>
</dependency>
```

In essence the exercise will add Spring onto this Hibernate web application. If you want, you can try it on your own or follow the steps in the exercise section below.

#### *The Exercise:*

Update the web.xml file to use Spring's open session in view filter and be sure to also add the spring ContextLoaderListener. You can delete our own OpenSessionInViewFilter class.

Create a springconfig.xml file inside the WEB-INF directory and create beans for the studentService and studentDao classes (setup dependency injection between them).

Also configure Hibernate inside the springconfig.xml file and inject the sessionFactory into the StudentDao. You can remove the hibernate.cfg.xml file and the HibernateUtil class.

Update the StudentCourseServlet class to retrieve the studentService instance from Spring instead of creating a new instance.

Annotate the appropriate methods on the StudentService and StudentDAO classes as @Transactional and remove the programmatic transactional demarcation.