SOFTENG 325 Software Architecture
Assignment 1 (main)

Ian Warren

July 29, 2016

This is the main part of assignment 1 for SOFTENG 325. It is supported by parts 1 and 2, due in weeks 2 and 6 respectively. You should work on the main assignment throughout weeks 3 to 8 of the course. The main part of the assignment is worth 13%, and thus represents a significant proportion of this course's assessment. This is indicative of the amount of work required to do well in the assignment.

The objective of this part of the assignment is for you to design and develop a REST Web service that persists data using ORM. More specifically, you should use Java's JAX-RS, JAXB and JPA specifications in developing your service.

You have complete freedom in the functionality of your Web service. For example, it could be a blogging service that allows users to post blog entries, query blog entries and to be notified when entries of interest are posted. It could be a shop service allowing users to browse products and submit orders. Another option is a news service that allows reporters to send in news reports and readers to search for and retrieve news articles; the service might also allow readers to subscribe to news feeds. Other options include services to support University clubs and societies.

In selecting a particular application domain for your Web Service, be creative! Also, be guided by the assessment criteria – ultimately your Web service is a vehicle to demonstrate that you meet the criteria.

Assessment criteria

The assessment criteria is organised around three grades – A to C. In general, the higher grades subsume the lower grades. So, to be awarded a B, grade you will need to demonstrate that you have satisfied all criteria for both C and B grades. For an A grade, you will need to satisfy the criteria for both the B and C grades in addition to the requirements for grade A.

Grade A*

For an A* submission, you should show that you have read beyond the lecture materials in some way and that you have applied what you have learned independently in developing your Web service. Possible suggestions include, but aren't limited to:

- *Provisioning for data formats other than XML and Java Serialization.* In addition to the formats covered in class, your Web service might include support for JSON, for example, and implement content negotiation.

- *Developing an AJAX client.* REST Web services are often consumed by browser-based applications. You might develop a Web page with part of its content being generated/updated by your Web service.

- *Making creative use of an interceptor.* In the second supporting assignment, a Web service interceptor is used for auditing purposes. You could devise a feature (other than auditing) that can be appropriately implemented using an interceptor.

Grade A

- *Provisioning for an asynchronous processing use case.* As discussed in class, there are a number of uses cases that require support for asynchronous processing. Your Web service should appropriately implement support for one such use case. The implementation should also be free of any concurrency-related issues, e.g. race conditions and deadlocks.

Grade B

- *Sufficiently complex domain model.* Your service's domain model should be sufficiently complex – including a minimum of 5 classes and a mix of different kinds of relationships between classes. As a guide, it should have a similar level of complexity to that of the full Parolee application (see slide 11 of the Web Applications lecture from Tuesday 26 July).

- *Leveraging the HTTP protocol.* The HTTP protocol has many features (e.g. cookies, link headers to support HATEOAS, query parameters, caching support etc.) that can be exploited in developing REST services. Your service should make appropriate use of at least 3 such features.

Grade C

To gain a pass mark for the assignment, you will need to develop a functioning Web service. Particular criteria that need to be satisfied are as follows:

- *Conformance to REST design principles.* Your service should be designed according to the following 4 fundamental principles of REST: addressable resources, uniform constrained interface, representation-oriented and stateless communication. You should take care in designing the service's URI scheme, and associating particular URIs with HTTP methods.

- *Development of a domain model.* The service must include a domain model. For a C grade submission, the model can be relatively simple involving a few classes and relationships.

- *Use of XML for resource representation.* Clients and the Web service must exchange XML-based representations of resources. The JAXB framework should be used to automate marshalling and unmarshalling between objects and XML.

- *ORM-based persistence.* Application data must be persisted using the JPA. The domain model should be appropriately decorated with persistence metadata (annotations).

- *Development of test cases.* JUnit test cases should be implemented to demonstrate operation of the Web service. The tests needn't be exhaustive, but must be sufficient to show operation of the service's fundamental features. The tests should use the JAX-RS Client API.

- *Use of a logging framework.* SLF4J should be configured and used to handle logging in your implementation.

- *Use of Maven for building the project.* You must use Maven to build, deploy and test your Web service. Deployment should involve using the embedded Jetty and H2 database resources. The project's POM file should specify all necessary dependencies and plugin configuration.

Submission

Submit a zip file that contains **all** code and resources that are necessary to build, deploy and test your Web service.

In preparing the zip file, run Maven's `clean` goal on your project to clear all generated code. Zip up the project's directory structure to include the `src` directory and `pom.xml` file. Upload your Zip file to the Assignment Drop Box (https://adb.auckland.ac.nz).

If you wish, you may include an optional `README` file to be stored in the same directory as the `pom.xml` file. Include a `README` file if it's necessary or helpful to explain any aspect of your assignment.

The deadline for submitting your work to the ADB is Wednesday 21 September at 18:00.