

CS 548—Fall 2018

Enterprise Software Architecture and Design

Assignment Eight—REST Web Service Client

In Assignment 7, you developed a REST Web service for your clinical information system. In this assignment, you will use the `curl` command-line tool to test your application, which should be deployed in AWS (EC2). To test that your REST Web service is properly set up, you can ping it for the clinic name¹:

```
curl -X GET -D headers \  
http://domain-name:8080/clinic-rest/resources/patient/site
```

This puts the result headers in the file called `headers`. You can query a patient resource as follows:

```
curl -X GET -D headers \  
http://domain-name:8080/clinic-rest/resources/patient/17
```

This returns an XML representation of the patient resource, of the form:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<patient-rep  
  xmlns:ns2="http://cs548.stevens.edu/clinic/service/web/rest/data">  
  <ns2:id  
    url="http://domain-name:8080/clinic-rest/resources/patient/17"  
    relation="http://cs548.stevens.edu/clinic/rest/relations/patient"  
    mediaType="application/xml"/>  
  <ns2:patient-id>12345678</ns2:patient-id>  
  <ns2:name>Joe Blow</ns2:name>  
  <ns2:dob>1983-10-04-04:00</ns2:dob>  
  <ns2:age>30</ns2:age>  
</patient-rep>
```

You can query a treatment for a patient in a similar way:

```
curl -X GET -D headers \  
http://domain-name:8080/clinic-rest/resources/patient/17/treatment/35
```

This returns a treatment representation, e.g. of the form:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<treatment-rep>...</treatment-rep>
```

You can add a patient resource as follows:

```
curl -X POST -D headers -H "Content-Type: application/xml" \  
-d @new-patient.xml \  

```

¹ The backslash ‘\’ is just the Unix way of breaking a shell command across several lines.

<http://domain-name:8080/clinic-rest/resources/patient>

The file `new-patient.xml` contains the representation for the patient that is being added. It is similar to the patient representation above. The response headers in the file headers have the following form:

```
HTTP/1.1 201 Created
X-Powered-By: Servlet/3.1 JSP/2.3 (GlassFish Server Open Source Edition 4.1
Java/Oracle Corporation/1.7)
Server: GlassFish Server Open Source Edition 4.1
Location: http://domain-name:8080/clinic-rest/resources/patient/3
Date: Sun, 16 Mar 2015 22:59:44 GMT
Content-Length: 0
```

Use curl to test the other operations, including the operations for adding providers and treatments, and for querying provider and treatment resources.

Submission

Your solutions should be developed for Java EE 8. You should use Java 8, Payara (Glassfish 5.0) and EclipseLink 2.5.x, and Eclipse Oxygen. In addition, record short mpeg or Quicktime videos of a demonstration of your assignment working (deploy the app in Amazon EC2 and run your testing, showing both input and output for client-side testing). Make sure that your name appears at the beginning of the video. *Do not provide private information such as your email or cwid in the video.*

Your solution should be uploaded via the Canvas classroom, as a zip file. This zip file should have the same name as your Canvas userid. It should unzip to a folder with this same name, which should contain the files and subfolders with your submission.

It is important that you provide a completed rubric that documents your submission, included as a PDF document in your submission root folder. As part of your submission, export your Eclipse projects to your file system, and then include those folders as part of your archive file. Your submission should also include any input test files that you used for testing. Your test videos should show input and output files, where appropriate.