[redacted] Back End Developer Sample Project

Instructions

Please create a web application (war) project based on the following requirements, and provide a zip file containing your source code to [redacted] at your earliest convenience. You are free to use any reference materials and tools not specifically restricted by the Technical Requirements below.

Evaluation

In addition to reviewing your source code, we will be compiling your project using Eclipse, and deploying it as a war file to a Java application server (Jetty, Tomcat, JBoss, etc.) to test that it functions as expected. We will review your code with you as part of your interview and you will be expected to explain your design choices and how your code is intended to function. We are looking to see how effectively you can code, meet defined requirements, and how easily you can pick up new technologies, APIs, etc. that you may not already be familiar with.

Functional Requirements

Create a Java project that compiles into a war file that provides a simple JAX-RS restful web service. This web service should accept HTTP GET requests to this url: <localhost:port>/<appname>/rest/number/pick/{x}, where {x} is any valid Java long value. This service should return an XML document as follows:

```
<stats>
        <average>77</average>
        <you-picked>10</you-picked>
        <last-pick>40</last-pick>
        <standard-deviation>115</standard-deviation>
</fstats>
```

The values above should be calculated as follows:

average: The mean of the last 100 numbers submitted to the service (by any requester), including the number defined in the url of the current request (The {x} value in the url described

above).

you-picked: The long value defined in the url of the current request (The {x} value in the url described above).

last-pick: The previous number most recently requested by the service.

standard-deviation: The standard deviation of the last 100 numbers submitted to the service (by any requester), including the number defined in the url of the current request (The {x} value in the url described above).

This service should be written to support concurrent requests without exception or data corruption.

Technical Requirements

- Project must be compatible with java 1.6
- JAX-RS implementation you should use is RESTEasy.
- You are also free to use compile and/or run-time dependencies provided in the distribution of JBoss 7.1.1.Final (Java EE 6 APIs etc.)
- No other external libraries or dependencies should be used.
- Feel free to use a build tool (ant, maven, etc.) or rely on Eclipse for compiling and deployment.
- Project should import into Eclipse (Juno).
- There should be no database/persistence component of this project.

Example Input/Output

Assuming that the following 4 requests were submitted in order, they should produce these associated responses:

Request 1:
request: http://localhost:8080/[redacted]/rest/number/pick/20
response:
<stats>
<average>20</average>
<you-picked>20</you-picked>
<standard-deviation>0</standard-deviation>
</stats>

Request 2:
request: http://localhost:8080/[redacted]/rest/number/pick/18
response:
<stats>
<average>19</average>
<you-picked>18</you-picked>
<last-pick>20</last-pick>
<standard-deviation>1</standard-deviation>
</stats>

Request 3:
request: http://localhost:8080/[redacted]/rest/number/pick/5
response:
<stats>
<average>14</average>
<you-picked>5</you-picked>
<last-pick>18</last-pick>
<standard-deviation>6</standard-deviation>
</stats>

Request 4:
request: http://localhost:8080/[redacted]/rest/number/pick/87
response:
<stats>
<average>32</average>
<you-picked>87</you-picked>
<last-pick>5</last-pick>
<standard-deviation>31</standard-deviation>
</stats>

Resources

Feel free to use any resources available to you to complete this project (although we expect all the code--aside from boilerplate generated by the IDE--to be your own). If you have never used RESTful web services, the JAX-RS specification or RESTEasy before, this is a good place to start:

Building Restful Web Services with JAX-RS
http://docs.oracle.com/javaee/6/tutorial/doc/giepu.html

RESTEasy Project Home:

http://www.jboss.org/resteasy

If you have any questions about the requirements, or get stuck, you may also feel free to contact [redacted]