Timothy Yong
Edward Choi
Shina Kolisetti
Shahan Akhter

11/4/15

Distributed Systems
Assignment 3

-POST Operation

We add basic information about the student to the database. We decoded an input JSON, then we took that object and added it into the database.

-GET Operation

The GET operation queries the students information. We queried the database for the first match, and then wrote it.

-DELETE Operation

The delete operation looks for specific year and take those Id’s in order to delete years that are less than the specific year.

-UPDATE Operation

The update operation gets the database and loop through once in order to get all grades. Get the average and the loop through a second time comparing grade to average in order to determine rating.

-LIST Operation

The list operation takes all students and lists them in JSON format.

Main.go runs the actual server, containing several sections which separate the different functionality.

To build the server, type: $ go build

then to run the server, type: $ ./417_hw3

REST API:

Post: to insert a new student, send over a json encoded POST request to /Student, where the data contains a dictionary with a fields “NetID”, “Name”, “Major”, “Year”, “Grade”, and “Rating” to http://url/Student

Example:

```
curl -H "Content-Type: application/json" -d '{"NetID":"147001235",
"Name":"Paul","Major":"Computer Science","Year":2015,"Grade":90,"Rating":"D"}'
http://localhost:1234/Student
```

Get: to get a student’s data, send over a query based url with a GET request to /Student/getstudent, with the student name in the field “name”

Example:

```
curl http://localhost:1234/Student/getstudent?name=Paul
```

Update: call update to url /Student with PUT request

Example:

```
curl -X "PUT" http://localhost:1234/Student
```

List: to get all the student’s data, send over a GET request to /Student/listall

Example:

```
curl http://localhost:1234/Student/listall
```

Delete: send a year with json encoded data DELETE request for the database to query based on that value. It then deletes all “students” based on what it finds

Example:

```
curl -X "DELETE" -H "Content-Type: application/json" -d '{"year":2016}'
http://localhost:1234/Student
```