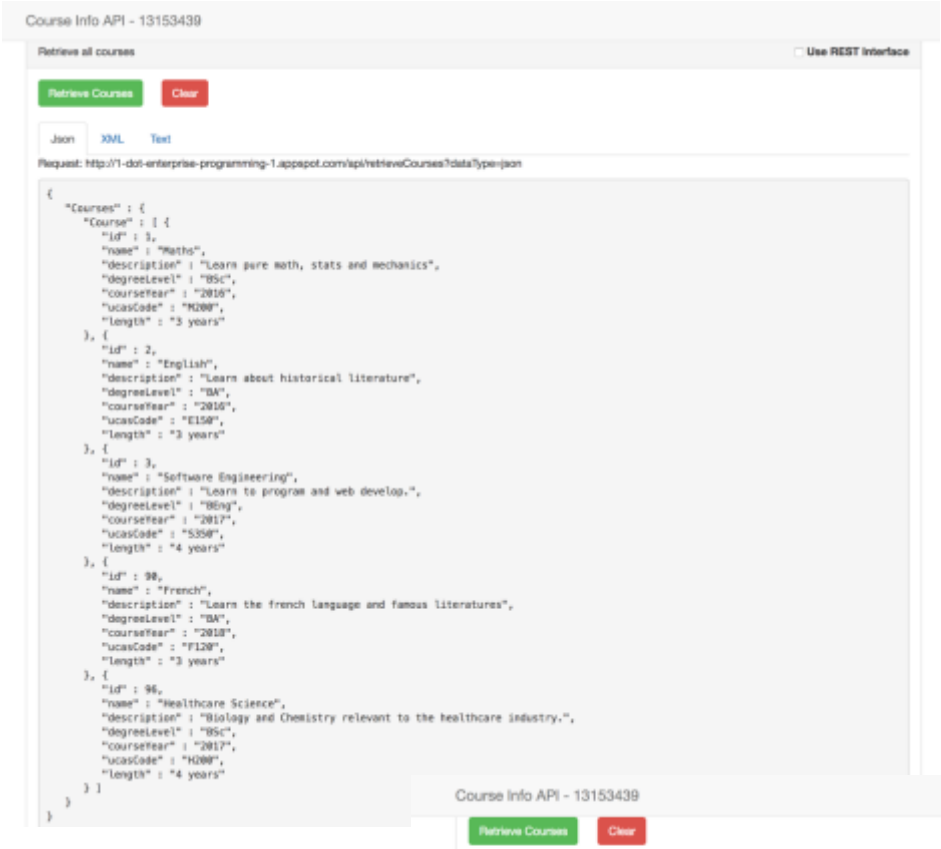
MMU SCMDT

ENTERPRISE PROGRAMMING REPORT

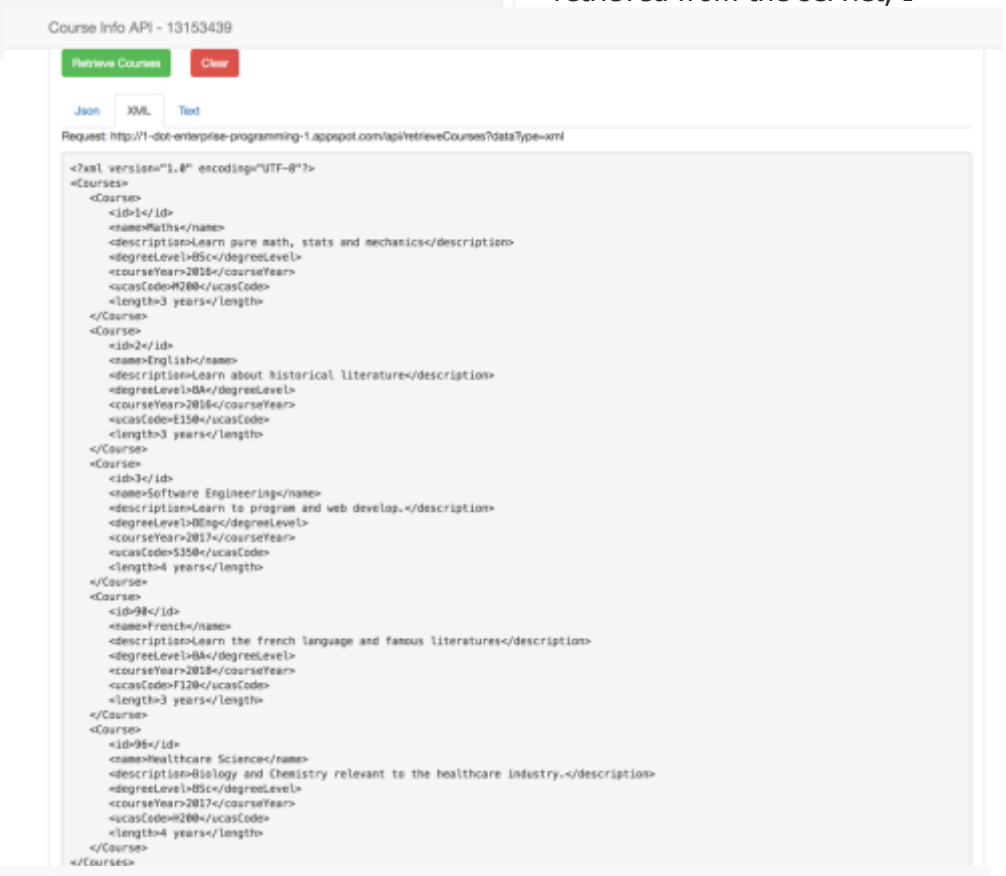
Huseyin Arpalikli

13153439

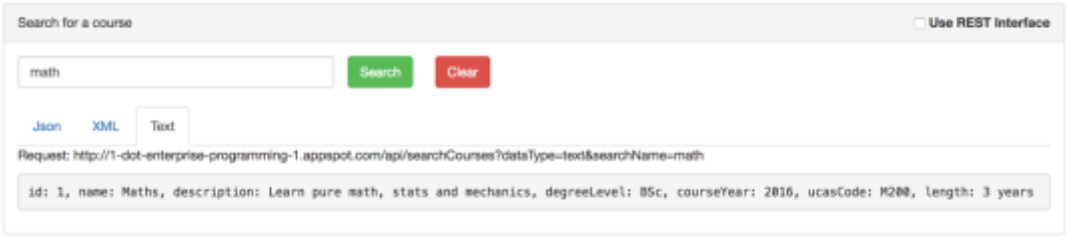
For this assignment, I have created a web service using various technologies. I was required to create a web service accessible in various formats including plain text, JSON and XML. I was then required to deploy a web service to the cloud, which would make it available for client use over the internet.

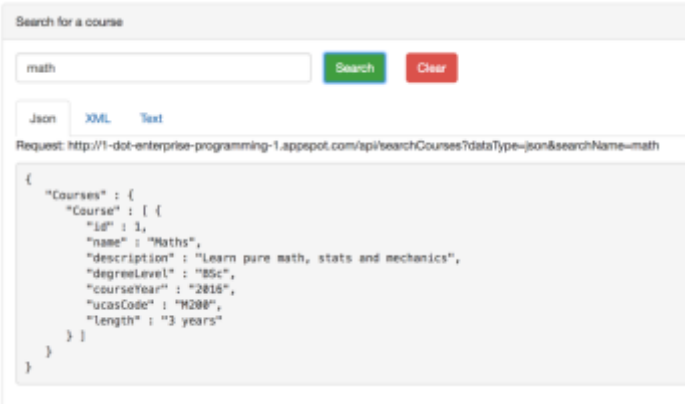


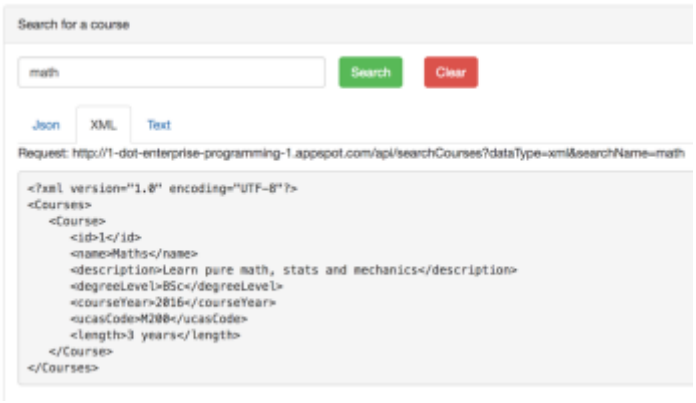
In the three screenshots on this page, I have retrieved data from a MySQL Database, using my Data Accessor Object (DAO) in Java, which uses my Courses POJO (Plain Old Java Object) to create the model that stores the objects to be sent to the servlet. Once the data has been retrieved from the servlet, I have used the JAXB library to convert the object data into JSON and XML. For the text format, I have simply created a string which concatenates the object variables and returns the values. After returning the data to the servlets, I have then used JQuery AJAX requests to return the data into the website HTML page. To implement my search function, I have used the same process described above, with the only difference being that I have used a returned parameter between the front-end GUI and the database, where the parameter is sent via the servlet to the DAO



and Database to retrieve used the SQL 'Like' function and SQL '%' parameter so signify 'any'. The method to return the data has been implemented using regular HTTP Servlets and in a RESTful interface. Both retrieval and searching courses use the HTTP GET method.

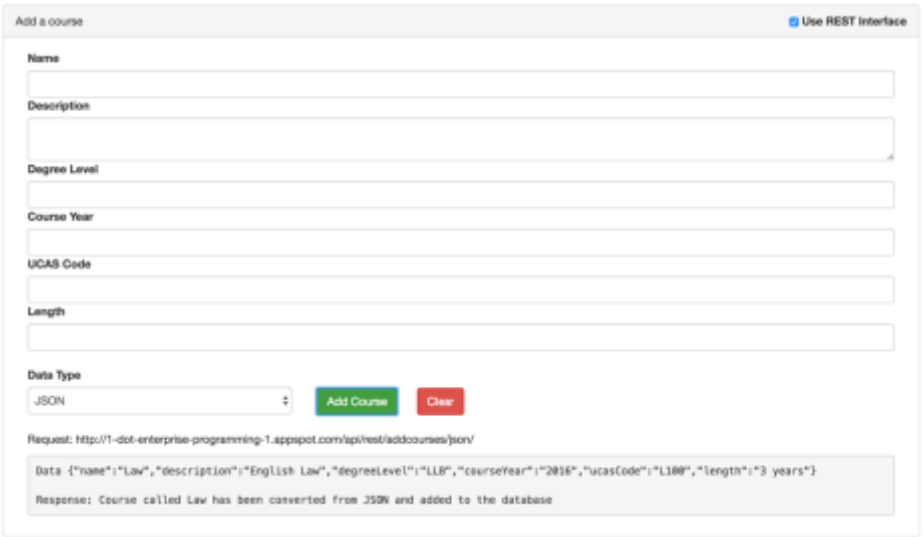






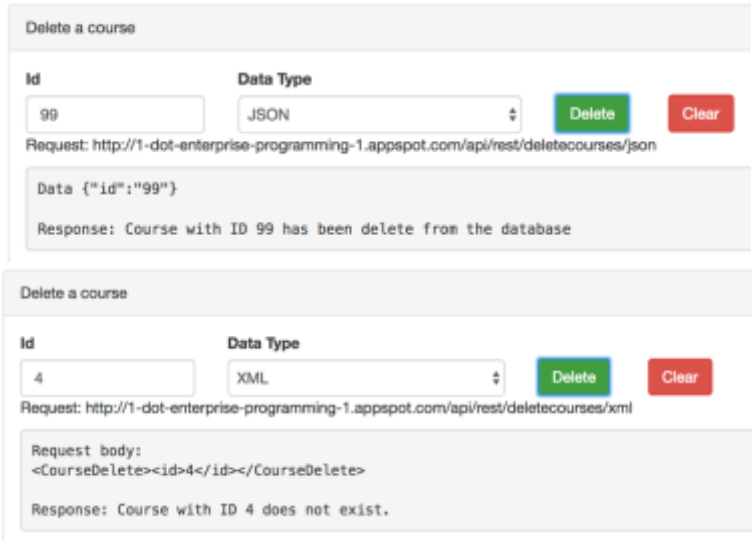
For course searching, I have searched by name and retrieved data in a similar method to retrieving all courses.

As part of this assignment I have used the four method of HTTP requests. These include GET, which I have used for searches and retrieval of data (see above); POST, which I have used for adding unique data (of new courses) from the front-end form to the database; PUT, which has been used for updating data which already exists inside of the database; and DELETE, which is used to delete data from the database. POST and GET are most commonly used in general but as an API PUT and DELETE functions are just as essential for manipulation functions within the API.

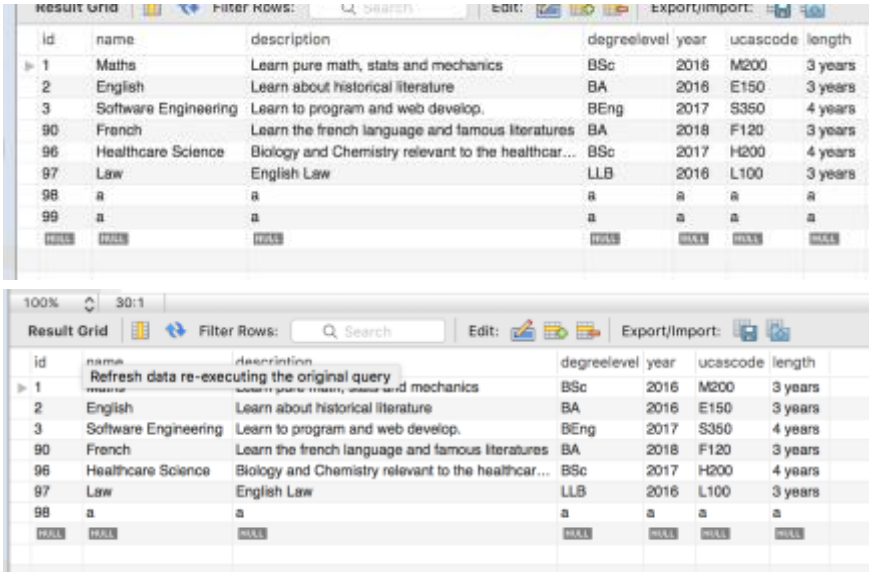


The screenshot on the left shows form data that has been sent via JSON to the servlet and

is then parsed using the GSON library (and JAXB library in the case of the XML data being parsed) and then sent via the DAO to the database.



To delete course data, I have implemented the different formats for transferring data, but as well as this I have made good use of handing errors and validating using if statements. In my servlet and DAO, I have carried out checks using if statements to make sure that errors messages are returned when data is not added due to non-existent values or empty text boxes. I have done this check in both the update and delete implementations.



These screenshots show the data before and after record 99 is deleted by the GUI.

I have made sure that my web service is persistent in its ability to store data in a Google MySQL instance. The Model View Controller pattern has been used in the assignment, however, I have not used JSP as I felt this was unnecessary.

My web service has been deployed to Google App Engine and I have created an instance of a cloud database my web service can be accessed from http://1-dot-enterprise-programming-1.appspot.com/